Journal of Mathematical Modeling
Vol. 11, No. 1, 2023, pp. 35–54. Research Article

JMM

Applications of the proximal difference-of-convex algorithm with extrapolation in optimal correction

Samira Shahsavari*, **Saeed Ketabchi**

Department of Applied Mathematics, Faculty of Mathematical Sciences University of Guilan, Rasht, Iran
Email(s): łsamira.tmoalem@gmail.com, sketabchi@guilan.ac.ir

Abstract. This paper proposes a proximal difference-of-convex algorithm with extrapolation ($PDCA_e$) based on Dinkelbach's approach for the optimal correction of two types of piecewise linear systems, classical obstacle problems and equilibrium problems, and linear inequalities. Using Dinkelbach's theorem leads to getting the roots of two single-variable functions. Considering the non-convex and level-bounded properties of the obtained problems, we use a proximal difference-of-convex algorithm programming to solve them. The experimental results on several randomly generated test problems show that the $PDCA_e$-generalized Newton method outperforms other methods for both feasible and infeasible cases.

Keywords: Proximal difference-of-convex, extrapolation, classical obstacle problem, equilibrium problems, linear inequalities, nonconvex, level-bounded.
AMS Subject Classification 2010: 90C26, 90C32, 90C25, 34A34, 15A39.

1 Introduction

In applied sciences, various factors, such as optimistic goals, noise, error in data, and lack of interaction between groups in charge of modeling (see [27]), may create an infeasible system. Reformulation or finding errors in the existing model may be time-consuming and costly, and the new system still may be infeasible. Hence, correcting systems have gained increasing attention in recent years (e.g., [1, 5, 11, 12, 15, 16, 18, 20–22, 25, 27, 28]).

This paper proposes a new method for correcting classical obstacle problems, equilibrium problems, and linear inequality systems. The classical obstacle problem arises by describing the equilibrium position of the elastic membrane above the obstacle; that is, minimizing the functional elastic energy by adding a constraint that represents the obstacle. In mathematical objects, the obstacle problem is related

*Corresponding author.
Received: 15 June 2022/ Revised: 21 August 2022/ Accepted: 11 September 2022
DOI: 10.22124/JMM.2022.22498.1986

to the study of differential calculus and partial differential equations. Its other applications can also be seen in optimal control, financial mathematics, and finite heat [14]. Equilibrium problems in numerical methods for free-surface hydrodynamics guarantee nonnegative water depths for each time step (see [4]). It can be shown that these two problems are equivalent to the absolute value equations (AVE).

For the optimal correction of these problems, first, the changes are applied simultaneously in the entries of the coefficient matrices and the right-hand side vectors. So to correct these problems, we obtain unconstrained quadratic fractional problems that are not necessarily convex (see [18]). Then, the Tikhonov regularization is used to control the norm of solutions. Next, the Dinkelbachs theorem is applied instead of solving the quadratic fractional problems, and the roots of two single-variable functions are found. This process requires solving optimization problems with non-Lipschitz, level-bounded, non-convex, and non-smooth objective functions. As the obtained problems are non-convex and level-bounded, the difference-of-convex (DC) programming is used to solve these problems. First, their objective functions are rewritten as the sum of smooth convex functions with the Lipschitz gradient, proper closed convex functions, and continuous concave functions. Then, a proximal DC algorithm with extrapolation ($PDCA_e$) is proposed to optimize the problems.

The remainder of this work is organized as follows. In Section 2, notations and preliminaries are provided. In Section 3, the $PDCA_e$-generalized Newton method is introduced for the optimal correction of infeasible piecewise linear systems and linear inequality systems, and the preliminaries of their optimal correction are described. In Section 4, numerical experiments are provided. Finally, conclusions are drawn in Section 5.

2 Notations and preliminaries

The notations used in this study are as follows. $|.|$, $\|.\|$ and $\|.\|_\infty$ denote the absolute value, 2-norm, and infinity-norm, respectively. The identity matrix of size n is shown by I. The transpose of $x \in \mathbb{R}^n$ is represented by x^t, and $x^t y$ displays the inner product of two vectors x and y, in $\mathbb{R}^n$. $(x)_+$ indicates a vector of elements $\max(0, x_i)$ for $i = 1, \ldots, n$, and $(x)_-$ represents a vector of elements $\min(0, x_i)$ for $i = 1, \ldots, n$. $sign(x)$ denotes a vector with components of $1, 0, -1$ depending on whether the corresponding component of x is positive, zero, or negative.

Let $f : \mathbb{R}^n \longrightarrow \mathbb{R} \cup \{+\infty\}$. If f is differentiable at point $x \in \mathbb{R}^n$, then the gradient of this function at x is denoted by $\triangledown f(x)$. Otherwise, $\partial f(x)$ denotes the generalized gradient of f at x, which is defined by

$$f(y) \geq f(x) + \partial f(x)^t (y - x), \quad \forall y \in dom(f).$$

For the two-variable function $f(x,t)$, $\triangledown_x f(x,t)$ $(\partial_x f(x))$ denotes the gradient (the generalized gradient) of the function $f(x,t)$ for the variable x. f is a closed proper convex function if its epigraph, i.e., the set $epi(f) = \{(x,q) \in \mathbb{R}^n \times \mathbb{R} : f(x) \leq q\}$, is a nonempty closed convex set, and the effective domain of this function is non-empty; in other words, $dom(f) = \{x \in \mathbb{R}^n : f(x) < +\infty\} \neq \emptyset$.

The proximal operator $prox_f : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ of a closed proper convex function f at point $y \in \mathbb{R}^n$ is defined by

$$prox_f(y) = \arg\min_{x \in \mathbb{R}^n} f(x) + \frac{1}{2}\|x - y\|^2.$$

The function minimized on the righthand side is strongly convex and not everywhere infinite, so it has a unique minimizer for every $y \in \mathbb{R}^n$ [24]. In this work, the proximal operator of the scaled function σf is used, where $\sigma > 0$, which is called the proximal operator of f with parameter σ and expressed as follows

$$prox_{\sigma f}(y) = arg\min_{x\in\mathbb{R}^n} f(x) + \frac{\sigma}{2}\|x-y\|^2.$$

The DC programming has been presented for the non-convex continuous optimization problems (see [32]). There is a class of DC optimization problems with an objective function as follows

$$\min_{x\in\mathbb{R}^n} f(x) + p(x) - g(x), \tag{1}$$

where f is a smooth convex function with a Lipschitz continuous gradient with modulus $L > 0$, p is a proper closed convex function, and function g is a continuous convex. Problem (1) can be solved by the classical DC algorithm (DCA), but the main drawback of this algorithm is its dependence on DC decomposition (see [31]). Goethe et al. [10] proposed the proximal DCA method to alleviate this shortcoming, although this algorithm may take a lot of iterations [31]. Wen et al. [31] employed extrapolation techniques to accelerate the proximal DCA and presented the proximal DCA with extrapolation (i.e., the $PDCA_e$ method). In each iteration of this algorithm, the extrapolation point is calculated by

$$y_k = x_k + \beta_k(x_k - x_{k-1}),$$

where $\beta_k \in (0,1]$, and the next iteration is estimated by

$$x_{k+1} = prox_{Lp}\left(y_k - \frac{1}{L}(\bigtriangledown f(y_k) - \partial g(x_k))\right).$$

3 $PDCA_e$-generalized Newton method

In this section, the $PDCA_e$-generalized Newton ($PDCA_e - GN$) method is presented for the optimal correction of infeasible systems of two piecewise linear systems and linear inequalities. This algorithm is a combination of the generalized Newton method for identifying the root of a function and the $PDCA_e$ method for solving an optimization problem.

In the optimal correction of infeasible systems, the following fractional problem is observed frequently

$$\min_{x\in\mathbb{R}^n} \Pi(x) = \frac{N(x)}{M(x)}, \tag{2}$$

where N and M are continuous, real-valued functions on $\mathbb{R}^n$, and $M(x) \neq 0$ for all $x \in \mathbb{R}^n$. The Tikhonov regularization is used to control the norm of the solutions

$$\min_{x\in\mathbb{R}^n} \overline{\Pi}(x) = \frac{N(x)}{M(x)} + \rho\|x\|^2, \tag{3}$$

where $\rho > 0$ is the regularization parameter. According to the Dinkelbach's theorem [8], instead of solving problem (3), the root of the following function is identified

$$F(t) = \min_{x\in\mathbb{R}^n} N(x) + \rho\|x\|^2 M(x) - tM(x). \tag{4}$$

The properties of the function $F(t)$ are proven by Dinkelbach [8] and stated in the following lemma.

Lemma 1. *The function $F(t)$ is concave, strictly monotonic decreasing, continuous on $\mathbb{R}$, and has a unique root.*

Theorem 1. *Consider the convex function $G(t) = -F(t)$ and assume that x_t is an optimal solution to the optimization problem in (4). Then, $G(t)$ is subdifferentiable, and $M(x_t)$ is its subdifferential at t.*

Proof. Since $G(t)$ is a convex function, it has at least one subgradient at t (see [2,7]). For all $s \in \mathbb{R}$,

$$G(t) - G(s) = -N(x_t) - \rho\|x_t\|^2 M(x_t) + tM(x_t) + N(x_s) + \rho\|x_s\|^2 M(x_s) - sM(x_s).$$

Since x_s is a solution to the problem

$$\min_{x\in\mathbb{R}^n} N(x) + \rho\|x\|^2 M(x) - sM(x),$$

we get

$$-G(s) = N(x_s) + \rho\|x_s\|^2 M(x_s) - sM(x_s) \leq N(x_t) + \rho\|x_t\|^2 M(x_t) - sM(x_t).$$

Thus,

$$\begin{aligned} G(t) - G(s) &\leq -N(x_t) - \rho\|x_t\|^2 M(x_t) + tM(x_t) + N(x_t) + \rho\|x_t\|^2 M(x_t) - sM(x_t) \\ &= M(x_t)(t-s), \end{aligned}$$

and so

$$G(s) \geq G(t) + M(x_t)(s-t),$$

which completes the proof. □

It should be noted that t^* is the root of $F(t)$ if and only if $t^* = \min_{x\in\mathbb{R}^n} \Pi(x)$ (see [21]). Now, as $G(t)$ is subdifferentiable, t^* can be obtained using the generalized Newton method

$$t_{i+1} = t_i - \frac{G(t_i)}{\partial G(t_i)}, \qquad \forall i \geq 0. \tag{5}$$

The function $G(t)$ contains an optimization problem. We consider it as a DC programming problem

$$G(t) = \min_{x\in\mathbb{R}^n} f(x) + p(x) - g(x,t), \tag{6}$$

where f is a smooth convex function with a Lipschitz-continuous gradient with a modulus $L > 0$, p is a proper closed convex function, and g is a continuous convex function for the variable x. This way, the $PDCA_e$ method can be used to solve it and calculate $x_{t_{i+1}}$ as follows:

$$\begin{aligned} x_{t_{i+1}} &= prox_{Lp}(x_k - \frac{1}{L}(\partial f(y_k) - \partial_x g(x_k,t_i))) \\ &= \arg\min_{x\in\mathbb{R}^n} p(x) + (\partial f(y_k) - \partial_x g(x_k,t_i))^t x + \frac{L}{2}\|x - y_k\|^2, \end{aligned} \tag{7}$$

where $k \geq 0$ and $x_0 = x_{t_i}$, $y_k = x_{k+1} - \beta_k(x_{k+1} - x_k)$, $\beta_k \in (0,1]$. The subproblem (7) is strongly convex, therefore, has a unique solution and can be easily solved with convex optimization methods.

Here, the algorithm is outlined based on the above process.

Algorithm 1 *$PDCA_e$*-GN method

1. **Choose** $\rho > 0$, accuracy parameter $\varepsilon_1 > 0$ and a starting point t_0.
2. **Set** $i = 0$.
3. **While** $|G(t_i)| \geq \varepsilon_1$ **do**
   Calculate $x(t_i)$ using the following algorithm:
   (a) **Choose** accuracy parameter $\varepsilon_2 > 0$ and $\{\beta_k\} \subseteq [0,1)$ whit $\sup_k \beta_k < 1$ and the starting point $x_0 \in dom(p)$.
   (b) **Set** $x_{-1} = x_0$ and $k = 0$.
   (c) **While** $\|C\|_\infty > \varepsilon_2$ **do**
   Calculate $\partial_x g(x_k, t_i)$;
   Set $y_k = x_k + \beta_k(x_k - x_{k-1})$; and calculate $\partial f(y_k)$;
   Calculate x_{k+1} by
$$\min_{x\in\mathbb{R}^n} p(x) + (\partial f(y_k) - \partial_x g(x_k, t_i))^t x + \frac{L}{2}\|x - y_k\|^2; \tag{8}$$
   Set $C = x_{k+1} - x_k$; and $k = k+1$;
   (d) **end;**

   Calculate $G(t_i)$;
   Set $t_{i+1} = t_i - \frac{G(t_i)}{\partial G(t_i)}$; and $i = i+1$;
4. **end**

In Algorithm 1, the parameter extrapolation β_k for $k \geq 0$ is considered as follows

$$\beta_k = \frac{\theta_{k-1} - 1}{\theta_k}, \text{ with } \theta_{k+1} = \frac{1 + \sqrt{1 + 4\theta_k^2}}{2}, \tag{9}$$

where $\theta_0 = \theta_{-1} = 1$ (see more details in [31]). The following theorem shows the convergence of Algorithm 1.

Theorem 2. *Suppose $\{x_{t_i}\}$ is a sequence generated by the $PDCA_e$-GN method. Then, any accumulation point of this sequence is convergent to a stationary point of* Π.

Proof. The *$PDCA_e$*-GN method in steps c-d (i.e., *$PDCA_e$* method) solves the following optimization problem at the ith iteration:

$$\min_{x\in\mathbb{R}^n} N(x) + \rho\|x\|^2 M(x) - t_i M(x). \tag{10}$$

Since the objective function of problem (10) is level-bounded, and the extrapolation parameters $\{\beta_k\}$ satisfy $\sup \beta_k < 1$ and $\{\beta_k\} \subseteq [0,1)$, any accumulation point of sequence generated by the *$PDCA_e$* method (steps c-d) is a stationary point of the problem (10) (see [31] for more details). On the other hand, $G(t)$ is a convex function by Theorem 1, and the generalized Newton method has a finite termination property (see [13]). Thus, any accumulation point of the sequence generated by the $PDCA_e$-GN method is a stationary point of Π. □

3.1 $PDCA_e$-GN method for optimal correction of piecewise linear and linear inequality systems

In this subsection, first, the following classical obstacle problem is considered

$$(x)_- + T(x)_+ = r, \tag{11}$$

where $T \in \mathbb{R}^{n\times n}$ and $r \in \mathbb{R}^n$. It appears in the semi-implicit methods for the numerical simulation of free-surface hydrodynamics and the numerical solutions to obstacle problems (see [33]).

Since $(x)_- = \frac{x-|x|}{2}$ and $(x)_+ = \frac{x+|x|}{2}$, systems (11) can be reformulated as the following absolute value equations

$$(T+I)x + (T-I)|x| = 2r. \tag{12}$$

Next, the following equilibrium problems are discussed, which are a correct formulation of numerical methods for free-surface hydrodynamics that guarantees nonnegative water depths for any time step (see [4])

$$Qx + (x)_+ = q, \tag{13}$$

where $Q \in \mathbb{R}^{n\times n}$ and $q \in \mathbb{R}^n$. If $(x)_+$ is replaced by $\frac{|x|+x}{2}$, system (13) is equivalent to the following absolute value equations

$$(2Q+I)x + |x| = 2q. \tag{14}$$

When the above problems are infeasible, the optimal correction of the following absolute value equations is considered

$$Ax + B|x| = b, \tag{15}$$

where $A, B \in \mathbb{R}^{n\times n}$, $b \in \mathbb{R}^n$. For the optimal correction of the infeasible system (15), the following fractional problem is solved (see [16] and [11])

$$\min_{x\in\mathbb{R}^n} \quad \frac{\|Ax+B|x|-b\|^2}{1+\|x\|^2}. \tag{16}$$

The above problem is not coercive, and it is possible that solving problem (16) leads to solutions with very large norms. Hence, the Tikhonov regularization method is used to turn this problem into a coercive problem and avoid ending up with a large norm solution:

$$\min_{x\in\mathbb{R}^n} \quad H(x) = \frac{\|Ax+B|x|-b\|^2}{1+\|x\|^2} + \rho\|x\|^2, \tag{17}$$

where $\rho \in \mathbb{R}$ is a positive parameter. The following function is obtained from Dinkelbach's theorem [8]

$$F(t) = \min_{x\in\mathbb{R}^n} \quad \|Ax+B|x|-b\|^2 + \rho\|x\|^2(1+\|x\|^2) - t(1+\|x\|^2). \tag{18}$$

The function F has the properties described in Lemma 1 and has a unique root in the interval $[0, \|b\|^2]$ (see [11]). Also, the subdifferential of the function $G(t) = -F(t)$ is $(1+\|x_t\|^2)$ at point t where x_t is a solution for the following problem (see [11, 21]):

$$\min_{x\in\mathbb{R}^n} \|Ax+B|x|-b\|^2+\rho\|x\|^2(1+\|x\|^2)-t(1+\|x\|^2). \tag{19}$$

So, the problem (19) must be solved to obtain the root of function F. Therefore, Proposition 1 demonstrates that problem (19) can be rewritten as a class of DC optimization problems.

Proposition 1. *There is $\gamma > 0$, so that $s(x) = \frac{\gamma}{2}\|x\|^2 + 2x^tA^tB|x| + \|B|x|\|^2$ is a convex function. Problem (19) is equivalent to a DC programming problem, so its objective function is the sum of a smooth convex function with Lipschitz-continuous gradient, a proper closed convex function, and a continuous concave function.*

The following notes are required to prove this proposition (also see [29]).

Note 1. *Let $x \in \mathbb{R}^n$,*

$$D_x = diag(sign(x)) = \begin{bmatrix} sign(x_1) & 0 & 0 & \cdots & 0 \\ 0 & sign(x_2) & 0 & \cdots & 0 \\ \vdots & & & \ddots & \vdots \\ 0 & 0 & \cdots & & sign(x_n) \end{bmatrix}, \tag{20}$$

where

$$sign(x_i) = \begin{cases} -1, & if \quad x_i < 0, \\ 0, & if \quad x_i = 0, \\ 1, & if \quad x_i > 0. \end{cases}$$

Therefore,

$$(sign(x_i))^2 = \begin{cases} 1, & if \quad x_i < 0 \quad or \quad x_i > 0, \\ 0, & if \quad x_i = 0, \end{cases} \tag{21}$$

and, since $|x| = D_x x$, thus

$$\||x|\|^2 = \|D_x x\|^2 = (\sum_{i=1}^{n}(sign(x_i)x_i)^2)^2 = \|x\|^2.$$

Therefore, $\|D_x x\|^2$ is a convex function.

Note 2. *Consider the function of two variables $M(x,y) = \delta(x^2+y^2)+\alpha x^2+\eta y^2+cxy$, where $\alpha, \eta, c \in \mathbb{R}$ are constant. This function can be reformulated as $M(x,y) = [(\delta+\overline{\alpha})x^2+px]+[(\delta+\overline{\eta})y^2+qy]$, where $\overline{\alpha}, \overline{\eta}, p, q \in \mathbb{R}$ (see [30]). For sufficiently large values of $\delta > 0$, $h(x) = (\delta+\overline{\alpha})x^2+px$ and $f(y) = (\delta+\overline{\eta})y^2+qy$ are convex functions. Then, $M(x,y)$ is a convex function. Now, consider the function $\phi(x_1,x_2) = \frac{\gamma}{2}(x_1^2+x_2^2)-(a_{11}sign(x_1))x_1^2+a_{22}sign(x_2))x_2^2)-(a_{12}sign(x_1)+a_{21}sign(x_2))x_1x_2$. Since $sign(x_i)$ =(-1) or 1 or 0, $i = 1,2$, $a_{ij}sign(x_i) \in \mathbb{R}$, $i, j = 1,2$. So, there exist $\alpha_1, \alpha_2, \lambda_1, \lambda_2 \in \mathbb{R}$ so that $\phi(x_1,x_2) = ((\frac{\gamma}{2}+\alpha_1)x_1^2+\lambda_1x_1)+((\frac{\gamma}{2}+\alpha_2)x_2^2+\lambda_2x_2)$. Therefore, for sufficiently large values of $\gamma > 0$, $\phi(x_1,x_2)$ is a convex function.*

Further details are provided below for the proof of Proposition 1.

Proof. It is proven that for sufficiently large $\gamma_1 > 0$, the function

$$s_1(x) = \frac{\gamma_1}{2}\|x\|^2 + 2x^t A^t B|x|,$$

is convex. Therefore,

$$s_1(x) = \frac{\gamma_1}{2}\|x\|^2 + 2x^t A^t BD_x x = x^t(\frac{\gamma_1}{2}I + ((A^t B)D_x + D_x(A^t B)^t))x.$$

Also, suppose that q_{ij}, $i, j = 1, \dots, n$ are the entries of the matrix $A^t B$. Then,

$$(A^t B)D_x = \begin{bmatrix} q_{11}sign(x_1) & q_{12}sign(x_2) & \cdots & q_{1n}sign(x_n) \\ q_{21}sign(x_1) & q_{22}sign(x_2) & \cdots & q_{2n}sign(x_n) \\ \vdots & \vdots & & \vdots \\ q_{n1}sign(x_1) & q_{n2}sign(x_2) & \cdots & q_{nn}sign(x_n) \end{bmatrix}.$$

So,

$$\frac{\gamma_1}{2}I + ((A^t B)D_x + D_x(A^t B)^t) =$$

$$\begin{bmatrix} \frac{\gamma_1}{2} + 2q_{11}sign(x_1) & (q_{12}sign(x_2) + q_{21}sign(x_1)) & \cdots & (q_{1n}sign(x_n) + q_{n1}sign(x_1)) \\ (q_{12}sign(x_2) + q_{21}sign(x_1)) & \frac{\gamma_1}{2} + 2q_{22}sign(x_2) & \cdots & (q_{2n}sign(x_n) + q_{n2}sign(x_2)) \\ \vdots & \vdots & \ddots & \vdots \\ (q_{1n}sign(x_n) + q_{n1}sign(x_1)) & (q_{2n}sign(x_n) + q_{n2}sign(x_2)) & \cdots & \frac{\gamma_1}{2} + 2q_{nn}sign(x_n) \end{bmatrix}.$$

Therefore

$$s_1(x) = \sum_{i=1}^{n}((\frac{\gamma_1}{2} + 2q_{ii}sign(x_i))x_i^2) + 2\sum_{i=1}^{n}\sum_{\substack{j=1 \\ j\neq i}}^{n}(q_{ij}sign(x_i) + q_{ji}sign(x_j))x_i x_j.$$

Since $sign(x_i) = (-1)$ or 1 or 0, $i = 1, \dots, n$, $q_{ij}sign(x_i) \in \mathbb{R}$, $i, j = 1, \dots, n$. According to Note 2, there exist $\alpha_i, \lambda_i \in \mathbb{R}$, $i = 1, \dots, n$, so that $s_1(x)$ is converted to

$$s_1(x) = \sum_{i=1}^{n}((\frac{\gamma_1}{2} - \alpha_i)x_i^2 + \lambda_i x_i).$$

Therefore, for sufficiently large values of $\gamma_1 > 0$, $s_1(x)$ is convex function. Similarly, it can be shown that there exists $\gamma_2 > 0$ so that the following function is convex

$$s_2(x) = \frac{\gamma_2}{2}\|x\|^2 + \|B|x|\|^2.$$

So, by choosing $\gamma = \max\{\gamma_1, \gamma_2\}$, function $s(x) = s_1(x) + s_2(x)$ is convex. Also, Problem (19) is equivalent to the following DC programming problem

$$\min_{x\in\mathbb{R}^n} f(x) + p(x) - g(x,t), \tag{22}$$

where

$$f(x) = \rho\|x\|^2, \quad p(x) = \|Ax-b\|^2 + \|B|x|\|^2 + 2x^tA^tB|x| + 2(-b^tB)_+|x| + \frac{\gamma}{2}\|x\|^2 + \rho\|x\|^4,$$
$$g(x,t) = t(1+\|x\|^2) + 2(b^tB)_+|x| + \frac{\gamma}{2}\|x\|^2.$$

Here, the function f is smooth convex with a Lipschitz-continuous gradient and modulus $L = 2\rho$. The effective domain of the function p is nonempty, so it is a proper closed convex, and the function g is a continuous convex for the variable x. □

In implementing the $PDCA_e$-GN method to correct infeasible AVE (15), the subgradient of the function g for the variable x at the point (x_k, t_i) and the gradient of the function f at the point $y_k = x_k + \beta_k(x_k - x_{k-1})$ are calculated as below

$$\begin{cases} \partial_x g(x_k,t_i) = (2t+\gamma)x_k + 2D_k(b^tB)^t_+, \\ \bigtriangledown f(y_k) = 2\rho y_k, \end{cases}$$

where $D_k = diag(sign(x_k))$. Instead of problem (8), the following convex optimization problem is solved

$$\min_{x\in\mathbb{R}^n} \|Ax-b\|^2 + \|B|x|\|^2 + 2x^tA^tB|x| + 2(-b^tB)_+|x| + \frac{\gamma}{2}\|x\|^2$$
$$+ \rho\|x\|^4 + (\bigtriangledown f(y_k) - \partial_x g(x_k,t_i))^t x + \frac{L}{2}\|x - y_k\|^2. \tag{23}$$

Remark 1. *For optimal correction of system* (15) *when B=-I, i.e., infeasible system*

$$Ax - |x| = b, \tag{24}$$

according to Note 1, the function $\|x\|^2$ is smooth. Therefore, the following DC problem can be considered instead of problem (22)

$$\min_{x\in\mathbb{R}^n} \overline{f}(x) + \overline{p}(x) - \overline{g}(x,t), \tag{25}$$

where

$$\overline{f}(x) = (1+\rho)\|x\|^2, \quad \overline{p}(x) = \|Ax-b\|^2 - 2x^tA^t|x| + 2(b)^t_+|x| + \frac{\gamma}{2}\|x\|^2 + \rho\|x\|^4,$$
$$\overline{g}(x,t) = t(1+\|x\|^2) + 2(-b)^t_+|x| + \frac{\gamma}{2}\|x\|^2,$$

in which the function $\overline{f}$ is smooth convex with a Lipschitz-continuous gradient with modulus $L = 2(1+\rho)$, the function $\overline{p}$ is proper closed convex, and the function $\overline{g}$ is a continuous convex for the variable x. The effectiveness of the $PDCA_e$-GN method for problem (25) *is confirmed via the results obtained in Section 4.*

Remark 2. *It should be noted that AVE* (15) *is converted to the linear equality system $Ax = b$ when $B = 0$. Hence, the linear equality system is a special case of AVE* (15)*, and the approach discussed in this subsection can be used to correct the infeasible linear equality system.*

Now, the infeasible system of linear inequalities is considered for all $A \in \mathbb{R}^{m\times n}$ and $b \in \mathbb{R}^m$

$$Ax \leq b, \tag{26}$$

The aim is to correct system (26) using changes in the matrix A entries and the right side vector b. It is equivalent to solving the following fractional problem (see [1])

$$\min_{x\in\mathbb{R}^n} \frac{\|(Ax-b)_+\|^2}{1+\|x\|^2}. \tag{27}$$

The Tikhonov regularization method is applied to control the norm of the solutions

$$\min_{x\in\mathbb{R}^n} \frac{\|(Ax-b)_+\|^2}{1+\|x\|^2} + \rho\|x\|^2, \tag{28}$$

where ρ is a positive parameter. Applying Dinkelbachs theorem results in

$$F(t) = \min_{x\in\mathbb{R}^n} \|(Ax-b)_+\|^2 + \rho\|x\|^2(1+\|x\|^2) - t(1+\|x\|^2). \tag{29}$$

The function $F(t)$ is a strictly decreasing and concave function with a unique root in the interval $[0, \|(-b)_+\|^2]$ (see [12,28]). Therefore, the function $F(t)$ has the properties of Lemma 1. Also, Theorem 1 is true for the function $G(t) = F(t)$ with subgradient $1+\|x_t\|^2$ at t, where x_t is a solution of

$$\min_{x\in\mathbb{R}^n} \|(Ax-b)_+\|^2 + \rho\|x\|^2(1+\|x\|^2) - t(1+\|x\|^2). \tag{30}$$

Proposition 2 proves that the problem (30) is equivalent to a form of the DC programming problem (1).

Proposition 2. *The problem* (30) *is equivalent to a DC programming problem so that the objective function of the problem is the sum of a smooth convex function with a Lipschitz-continuous gradient with modulus $L = 2\rho$, a proper closed convex function, and a continuous concave function.*

Proof. By choosing the following functions

$$f(x) = \rho\|x\|^2, \qquad p(x) = \|(Ax-b)_+\|^2 + \rho\|x\|^4, \qquad g(x,t) = t(1+\|x\|^2), \tag{31}$$

it is clear that problem (30) is equivalent to the following DC problem

$$\min_{x\in\mathbb{R}^n} f(x) + p(x) - g(x,t) \tag{32}$$

where f is a smooth convex function with a Lipschitz-continuous gradient with modulus $L = 2\rho$, p is a proper closed convex function, and g is a continuous convex function for the variable x. □

The results of this subsection support the conclusion that the $PDCA_e$-GN method can be used to solve problem (27). So, the two gradients are calculated as follows

$$\begin{cases} \bigtriangledown f(y_k) = 2\rho y_k, \\ \bigtriangledown_x g(x_k, t_i) = 2t_i x_k. \end{cases} \tag{33}$$

Also, problem (8) is equivalent to the following convex optimization problem

$$\min_{x\in\mathbb{R}^n} \|(Ax-b)_+\|^2 + \rho\|x\|^4 + (\bigtriangledown f(y_k) - \bigtriangledown_x g(x_k,t_i))^t x + \frac{L}{2}\|x-y_k\|^2. \tag{34}$$

Remark 3. *In many applications, the linear inequality systems have non-negative variables; that is, the following system is observed*

$$\begin{cases} \overline{A}x \leq \overline{b}, \\ x \geq 0, \end{cases} \tag{35}$$

where $\overline{A} \in \mathbb{R}^{n\times n}$ *and* $\overline{b} \in \mathbb{R}^n$*. If*

$$A = \begin{bmatrix} \overline{A} \\ -I \end{bmatrix}, \qquad b = \begin{bmatrix} \overline{b} \\ 0 \end{bmatrix},$$

then, system (35) *turns into the system* (26).

4 Numerical experiments

In this section, three examples are randomly generated to demonstrate the effectiveness of the $PDCA_e$-GN method. The first and second examples involve AVE problems (15) and (24), respectively. The third example considers the system of linear inequalities (26). In each of these examples, first, it is shown that the proposed algorithm is valid for optimal correction; so, the algorithm is valid

- For AVE problems (15)

$$r^* = \frac{\|Ax^* + B|x^*| - b\|^2}{1 + \|x^*\|^2} \approx 0, \quad \text{and} \quad E^* = -r^* x^{*^t} \approx 0,$$

  where x^* is an optimal solution to the fractional programming problem (16) (see [16]).

- For linear inequalities

$$r^* = \frac{(Ax - b)_+}{1 + \|x\|^2} \approx 0, \quad \text{and} \quad E^* = -r^* x^{*^t} \approx 0,$$

  where x^* is a solution to the problem (27) (see [27]).

Therefore, problems (15) and (27) are feasible whenever

$$\tfrac{1}{2}(\|r^*\|^2 + \|E^*\|^2) \approx 0.$$

Then, the PDCAe-GN method is compared with other algorithms for infeasible problems. The numerical results obtained in this section are calculated using a personal computer (CORE i5, CPU 2.50 GHz, 4 GB memory, MATLAB 2013). In the tables provided in this section, every random system with identical dimensions is generated ten times simultaneously, and the average of the numerical results is presented. The first column indicates the size of matrix A, and $Time(s)$ is the average calculated CPU times. The sign "-" indicates a $Time(s) > 600$, and

$$M_{rE} = \frac{1}{10}\sum_{i=1}^{10}\left(\frac{1}{2}(\|r_i^*\|_\infty^2 + \|E_i^*\|_\infty^2)\right),$$

where r_i^* and E_i^* are the approximations obtained of the algorithm in step i.

Example 1. Here, the system of AVE (15) is considered. Two numerical experiments are performed to demonstrate the effectiveness of the $PDCA_e$-GN method in various randomly generated feasible and infeasible problems of the AVE (15). In the tables, $G(t^*)$ is defined as

$$G(t^*) = |\frac{1}{10}\sum_{i=1}^{10}(\|Ax_i^* + B|x_i^*| - b\|^2 + \rho\|x_i^*\|^2(1+\|x_i^*\|^2) - t_i^*(1+\|x_i^*\|^2))|,$$

where x_i^* and t_i^* are the solutions obtained from the algorithm in step i, and $Me = \frac{1}{10}\sum_{i=1}^{10}(\|(A+E_i^*)x_i^* + B|x_i^*| - (b+r_i^*)\|_\infty)$. First, the feasible AVE (15) is considered, which is randomly generated by the following MATLAB code

Code 3. A randomly generated feasible system of absolute value equations

```
n = input('Enter n :');
A = 100 * ((rand(n,n) - rand(n,n)));
x = (rand(n,1) - rand(n,1));
B = spdiags(x,0,n,n);
b = A * x + B * abs(x);
```

The numerical results from the implementation of the $PDCA_e$-GN method for Code 3 are summarized in Table 1. The results of this table show that the proposed algorithm has a desirable performance on a feasible system of AVE (15) and is valid for correcting an infeasible AVE since the expectations are $\frac{1}{2}(\|r^*\|_\infty^2 + \|E^*\|_\infty^2) \approx 0$.

Table 1: Numerical results of the $PDCA_e$-GN method for the feasible AVE (15).

n	Me	$G(t^*)$	M_{rE}	$Time(s)$
3000	1.7440e-11	7.7812e-11	1.0033e-12	6.59
4000	2.3351e-11	1.7474e-13	2.0037e-12	13.95
5000	2.8581e-11	4.2826e-11	1.0680e-10	29.22

Second, for an infeasible AVE, the proposed algorithm is compared with the DC-Newton method from Hashemi and Ketabchi [11]. They used Dinkelbachs approach to transform the fractional programming problem (16) into a single variable equation. They then applied four smoothing functions for $|t|$

$$\phi_1(t,\mu) = \mu\left[\ln(1+e^{-\frac{t}{\mu}}) + \ln(1+e^{\frac{t}{\mu}})\right].$$

$$\phi_2(t,\mu) = \begin{cases} t, & if \quad t \geq \frac{\mu}{2}, \\ (\frac{t^2}{\mu} + \frac{\mu}{4}), & if \quad -\frac{\mu}{2} < t < \frac{\mu}{2}, \\ -t, & if \quad t \leq -\frac{\mu}{2}. \end{cases}$$

$$\phi_3(t,\mu) = \sqrt{4\mu^2 + t^2}.$$

$$\phi_4(t,\mu) = \begin{cases} \frac{t^2}{2\mu}, & if \quad |t| \leq \mu, \\ |t| - \frac{\mu}{2}, & if \quad |t| > \mu, \end{cases}$$

where μ is a positive constant. Instead of the problem (19), they obtained a smoothing DC programming problem as below

$$\min_{x\in\mathbb{R}^n} k(x) - h(x),$$

where

$$k(x) = \|Ax - \Phi_i(x) - b\|^2 + \overline{\gamma}\|x\|^2 + \rho\|x\|^2(1+\|x\|^2), \quad h(x) = t(1+\|x\|^2) + \overline{\gamma}\|x\|^2,$$

in which $\overline{\gamma} > 0$ and $\Phi_i(x) = \big(\phi_i(x_1,\mu), \phi_i(x_2,\mu), \ldots, \phi_i(x_n,\mu)\big)^\top$ (see [11] for more details). They indicated that ϕ_4 was the optimal choice of the smoothing function for applying the DC-Newtown method. So, the proposed algorithm was compared with the DC-Newtown method with a smoothing function ϕ_4 (i.e., $DC-N(\phi_4)$ method). The infeasible AVE is generated based on the following lemma (see [21]).

Lemma 2. *If* $\{x \in \mathbb{R}^n | (A-B)x - b \geq 0\} = \emptyset$ *or* $\{x \in \mathbb{R}^n | (A+B)x - b \geq 0\} = \emptyset$*, then equation* (15) *is infeasible.*

Farkas lemma is used to generate the MATLAB code of the infeasible AVE. It is supposed that $\{x \in \mathbb{R}^n | (A-B)x - b \geq 0\} = \emptyset$. Then, by Farkas lemma, a vector $u \in \mathbb{R}^n$ exists such that $(A-B)^t u = 0$, $b^t u > 0$ and $u \geq 0$. Hence, the MATLAB code for the infeasible AVE (15) is written as

Code 4. A randomly generated infeasible system of absolute value equations

```
n = input('Enter n :');
pl = inline('(abs(x)+x)/2'); % plusfunction
u = 10*(rand(n,1) - rand(n,1));
u = pl(u);
k = null(u');
k = [k, zeros(n,1)];
x = spdiags(rand(n;1), 0, n, n)*(rand(n,1) - rand(n,1));
x = spdiags(ones(n,1) - sign(x), 0, n, n)*10*(rand(n,1) - rand(n,1));
B = spdiags(x, 0, n, n) :
A = k + B;    % (A-B)'u = 0
b = u;    % b'u > 0
```

Table 2 compares the performance of the $PDCA_e$-GN method and the $DC-N(\phi_4)$ method. It indicates that the proposed algorithm has a better performance than the $DC-N(\phi_4)$ method in terms of time and error, optimal correction, and accuracy. Therefore, the $PDCA_e$-GN method is more effective for the optimal correction of an infeasible AVE system (15).

Table 2: Comparison of the $PDCA_e - GN$ and $DC - N(\phi_4)$ methods for the infeasible AVE (15).

n	method	Me	$G(t^*)$	M_{rE}	$Time(s)$
1000	$PDCA_e - GN$	1.8037e-15	5.8309e-11	4.9066e+01	0.56
	$DC - N(\phi_4)$	1.7493e-14	3.3120e-09	5.1274e+01	3.62
2000	$PDCA_e - GN$	2.0989e-15	5.8862e-10	4.9252e+01	2.57
	$DC - N(\phi_4)$	2.1066e-14	1.4814e-09	4.9010e+01	23.04
3000	$PDCA_e - GN$	2.0922e-15	2.9939e-09	4.9386e+01	6.85
	$DC - N(\phi_4)$	3.0598e-14	1.5942e-08	4.9297e+01	67.58
4000	$PDCA_e - GN$	2.0378e-15	9.5014e-09	4.9404e+01	14.09
	$DC - N(\phi_4)$	3.0366e-14	4.8726e-08	4.9471e+01	155.97
5000	$PDCA_e - GN$	2.1897e-15	2.4548e-08	4.9409e+01	25.75
	$DC - N(\phi_4)$	3.4903e-14	2.0384e-08	4.9454e+01	303.86
5500	$PDCA_e - GN$	2.2204e-15	8.6579e-09	4.9600e+01	31.93
	$DC - N(\phi_4)$	3.8431e-14	3.4386e-08	4.9557e+01	429.58
6000	$PDCA_e - GN$	2.0527e-15	1.2704e-08	4.9440e+01	46.58
	$DC - N(\phi_4)$	3.9549e-14	5.7378e-08	4.9288e+01	540.97

Example 2. In this example, the efficiency of the $PDCA_e$-GN method is investigated for two cases of feasible and infeasible AVE (24). In the implementation of the $PDCA_e$-generalized Newton method, the algorithm with $PDCA_e$-GN is used for the problem (22), and the algorithm with $PDCA_e$-GN(I) is used for the problem (25). Table 3 presents the numerical results of these two methods for feasible AVE (24). In Table 4, the numerical results of these two algorithms for the infeasible AVE (24) are compared with the $DC - N(\phi_4)$ method. The symbols of the Example 4 tables are used, the Codes MATLAB 3 and 4 are considered, and we put $B = -I$.

Table 3: Comparison of the $PDCA_e - GN(I)$ and $PDCA_e - GN$ methods for the feasible AVE (24).

n	method	Me	$G(t^*)$	M_{rE}	$Time(s)$
3000	$PDCA_e - GN(I)$	1.7099e-11	3.9926e-11	1.6898e-12	27.43
	$PDCA_e - GN$	1.5416e-11	9.4360e-16	1.6932e-12	29.53
4000	$PDCA_e - GN(I)$	2.2101e-11	5.5005e-11	1.9502e-12	59.98
	$PDCA_e - GN$	2.2567e-11	1.7124e-11	1.9478e-12	68.00
5000	$PDCA_e - GN(I)$	2.7467e-11	1.9553e-10	1.0042e-12	111.97
	$PDCA_e - GN$	2.7603e-11	1.0870e-11	9.9920e-13	138.06

The results presented in Table 3 demonstrate the validity of the $PDCA_e$-GN(I) method. The last column of the table shows that the $PDCA_e$-GN(I) method has higher speed in solving a feasible AVE (24) than the $PDCA_e$-GN method.

Table 4 presents the results of the implementation of the $PDCA_e$-GN(I), $PDCA_e$-GN, and $DC - N(\phi_4)$ methods for the correction of the infeasible AVE (24). It is seen that the error of optimal correction (Me)

in $PDCA_e$-GN(I) and $PDCA_e$-GN methods are nearly similar, but the $DC-N(\phi_4)$ method has higher error. Also, the $PDCA_e$-GN(I) method is faster than the other two methods, and the $PDCA_e$-GN method is faster than $DC-N(\phi_4)$.

Table 4: Comparison of the $PDCA_e-GN(I)$, $PDCA_e-GN$, and $DC-N(\phi_4)$ methods for the infeasible AVE (24).

n	method	Me	$G(t^*)$	M_{rE}	$Time(s)$
1000	$PDCA_e-GN(I)$	1.3975e-15	7.9019e-11	4.8923e+01	0.62
	$PDCA_e-GN$	2.0561e-15	3.5738e-11	4.8923e+01	1.34
	$DC-N(\phi_4)$	1.4663e-14	1.0550e-11	4.9118e+01	7.28
2000	$PDCA_e-GN(I)$	1.8363e-15	3.5056e-10	4.9166e+01	4.41
	$PDCA_e-GN$	2.0849e-15	5.8088e-10	4.9166e+01	8.48
	$DC-N(\phi_4)$	2.3228e-14	2.2344e-09	5.1218e+01	39.370
3000	$PDCA_e-GN(I)$	2.2720e-15	6.6551e-10	4.9302e+01	24.91
	$PDCA_e-GN$	2.1450e-15	8.4954e-10	4.9302e+01	28.72
	$DC-N(\phi_4)$	2.5938e-14	-8.2946e-10	4.9254e+01	119.59
4000	$PDCA_e-GN(I)$	2.1285e-15	4.3001e-10	4.9750e+01	51.05
	$PDCA_e-GN$	2.1764e-15	2.4628e-09	4.9323e+01	58.82
	$DC-N(\phi_4)$	2.9635e-14	1.4495e-08	4.9323e+01	277.91
5000	$PDCA_e-GN(I)$	2.1640e-15	3.9784e-10	4.9387e+01	123.60
	$PDCA_e-GN$	2.2189e-15	5.4887e-09	4.9387e+01	110.39
	$DC-N(\phi_4)$	3.5053e-14	2.0955e-10	4.9551e+01	599.23
5500	$PDCA_e-GN(I)$	2.1991e-15	4.8793e-10	4.9256e+01	220.89
	$PDCA_e-GN$	2.5689e-15	3.6224e-08	4.9256e+01	166.30
	$DC-N(\phi_4)$	-	-	-	-
6000	$PDCA_e-GN(I)$	2.1662e-15	6.2699e-10	4.9707e+01	422.15
	$PDCA_e-GN$	2.4778e-15	1.2065e-08	4.9707e+01	473.30
	$DC-N(\phi_4)$	-	-	-	-

The proximal subgradient method (PSM) by Shahsavari and Ketabchi [29] is compared with the exact Douglas-Rachford splitting method (DRs) by Chen et al. [6] for the feasible AVE (24). Table 5 shows the numerical results, where $Time(s)$ is the average elapsed CPU times, and

$$f^* = \frac{1}{10}\sum_{i=1}^{10}(\|Ax_i^* + B|x_i^*| - (b)\|_\infty),$$

where x_i^*, is the solution obtained from the algorithms. Numerical results indicate that the PSM method is superior to the DRs method in terms of accuracy and time.

Example 3. In this example, numerical experiments are presented for the optimal correction of an infeasible system in linear inequality on various randomly generated problems. In the first column of the following tables, d represents the density,

$$G(t^*) = |\frac{1}{10}\sum_{i=1}^{10}(\|(Ax_i^* - b)_+\|^2 + \rho\|x_i^*\|^2(1+\|x_i^*\|^2) - t_i^*(1+\|x_i^*\|^2))|,$$

Table 5: Comparison of the $PDCA_e - GN(I)$, $PDCA_e - GN$ and DRs methods for the feasible AVE (24).

n	method	f^*	$Time(s)$
3000	*PSM*	3.4062e-12	5.76
	DRs	4.5747e-09	218.98
4000	*PSM*	6.3699e-12	12.98
	DRs	8.7296e-10	554.39
5000	*PSM*	1.6139e-11	29.096
	DRs	-	-
6000	*PSM*	2.6191e-11	55.99
	DRs	-	-
7000	*PSM*	8.7411e-11	74.29
	DRs	-	-
8000	*PSM*	5.1587e-12	183.68
	DRs	-	-

where x_i^*, and t_i^* are the solutions obtained from the algorithm in step i, and

$$Me = \frac{1}{10}\sum_{i=1}^{10}(\|((A+E_i^*)x_i^* - (b+r_i^*))_+\|_\infty).$$

The following MATLAB code is considered for the feasible system of linear inequalities, and the numerical results are presented in Table 6.

Code 5. A randomly generated feasible system of linear inequalities

$m = input(Enter\ m:);$

$n = input(Enter\ n:);$

$d = input(Enter\ d\ in\ (0,1]:);$

$A = sprand(m,n,d);$

$A = 100*(A - .5*spones(A));$

$x = spdiags(rand(n,1),0,n,n)*(rand(n,1) - rand(n,1));$

$b = A*x + spdiags((rand(m,1)),0,m,m)*10*ones(m,1);$

Table 6: Numerical results of the $PDCA_e$-GN method for the feasible system of linear inequalities.

n, d	Me	$G(t^*)$	M_{rE}	$Time(s)$
3000, 0.1	3.2486e-12	8.8436e-09	1.0071e-13	23.42
4000, 0.1	1.6747e-12	3.4764e-09	2.3555e-16	49.58
5000, 0.1	1.7991e-12	1.4799e-08	2.2092e-12	85.39

As shown in Table 6, the values of $\frac{1}{2}(\|r^*\|_\infty^2 + \|E^*\|_\infty^2) \approx 0$. Therefore, the $PDCA_e$-GN method is valid for correcting the system of linear inequalities.

For the infeasible system of linear inequalities, the proposed algorithm was compared with the l_p-norm regularization method (l_p-NR) proposed by Hashemi and Ketabchi [12] for the optimal correction of this system. They used the l_p-NR method to solve the fractional minimization problem (27), then applied Dinkelbachs theorem and presented a DC smoothing function to approximate the non-differentiable objective function. Combining the DC programming and smoothing quadratic regularization approaches with the fast quasi-Newton method produces an efficient method for solving this problem (in this paper, it is assumed that $p = 2$). For this comparison, the following MATLAB code is considered (see [12]) based on Farkas lemma. Since the system of linear inequalities (26) is inconsistent, a vector $u \in \mathbb{R}^m$ exists such that $A^t u = 0$, $b^t u < 0$, and $u \geq 0$.

Code 6. A randomly generated infeasible system of linear inequalities

```
m = input(Enter m : );
u1 = rand(m,1);    u2 = rand(m,1);
u12 = [u1;u2];    %u = u12, u >= 0
A1 = null(u1');    A2 = null(u2');
A = 100 * [A1; -A2];    %A'u = 0
uu = rand(2 * m,1);
b = 5 * (0.8 * uu - u12);    %b'u < 0
```

Table 7 shows the results of the calculations. As can be seen, the $PDCA_e$-GN method has superior accuracy and speed to the l_p-NR method. The high accuracy and speed of the $PDCA_e$-GN method in large-scale infeasible problems indicate the efficiency of the proposed algorithm for correcting infeasible linear inequalities.

5 Conclusions

This paper proposed like Dinkelbach approach for the optimal correction of the linear inequalities and piecewise linear systems based on a proximal difference-of-convex algorithm with extrapolation ($PDCA_e$). This method resulted in the reduction of the non-convex fractional problems to simple univariate equations on closed intervals. The convergence of the proposed method was shown under suitable conditions. The experimental results on several randomly generated problems indicated that the PDCAe-generalized Newton method is superior to the other methods in both feasible and infeasible large-scale problems.

References

[1] P. Amaral, P. Barahona, *A framework for optimal correction of inconsistent linear constraints*, Constraints **10** (2005) 67–86.

[2] M.-S. Bazaraa, H.-D. Sherali, C.-M. Shetty, Nonlinear programming: theory and algorithms, John Wiley and Sons, 2013.

Table 7: Comparison of the $PDCA_e - GN$ and $l_p - NR$ methods for the infeasible system of linear inequalities.

$m \times n$, d	method	Me	$G(t^*)$	M_{rE}	$Time(s)$
2000×999, .1	$PDCA_e - GN$	1.4627e-15	1.1084e-13	2.0399e+01	1.113
	$l_p - NR$	9.2287e-16	7.6921e-09	1.8398e+01	73.80
3000×1499,.01	$PDCA_e - GN$	1.6737e-15	3.1264e-14	3.3002e+01	4.04
	$l_p - NR$	1.1102e-15	6.7212e-11	2.6664e+01	282.92
4000×1999,1	$PDCA_e - GN$	2.2787e-15	7.9581e-14	5.2097e+01	8.30
	$l_p - NR$	-	-	-	-
5000×2499,1	$PDCA_e - GN$	2.6701e-15	2.9559e-13	7.8520e+01	18.00
	$l_p - NR$	-	-	-	-
6000×2999,1	$PDCA_e - GN$	2.5258e-15	2.3306e-13	1.1234e+02	30.29
	$l_p - NR$	-	-	-	-
7000×3499,1	$PDCA_e - GN$	3.1641e-15	-3.6380e-13	1.5659e+02	61.42
	$l_p - NR$	-	-	-	-
8000×3999,1	$PDCA_e - GN$	3.6693e-15	-2.3874e-13	2.1098e+02	92.69
	$l_p - NR$	-	-	-	-
9000×4499,1	$PDCA_e - GN$	3.5250e-15	4.3201e-13	2.7372e+02	143.26
	$l_p - NR$	-	-	-	-
10000×4999,1	$PDCA_e - GN$	4.5741e-15	5.3433e-13	3.6156e+02	170.60
	$l_p - NR$	-	-	-	-
11000×5499 ,1	$PDCA_e - GN$	4.4187e-15	3.6380e-13	4.5113e+02	386.70
	$l_p - NR$	-	-	-	-

[3] L. Brugnano, A. Sestini, *Iterative solution of piecewise linear systems for the numerical solution of obstacle problems*, arXiv preprint: https://arxiv.org/abs/0912.3222.

[4] L. Brugnano, V. Casulli, *Iterative solution of piecewise linear systems*, SIAM J. Sci. Comput. **30** (2008) 463–472.

[5] D. Carp, C. Popa, C. Serban, *Iterative solution of inconsistent systems of linear inequalities*, PAMM **13** (2013) 407–408.

[6] C. Chen, D. Yu, D. Han, *Exact and inexact DouglasRachford splitting methods for solving large-scale sparse absolute value equations*, IMA J. Nume. Anal. **43** (2023) 1036–1060.

[7] J.P. Crouzeix, J.A. Ferland, *Algorithms for generalized fractional programming*, Math. Program. **52** (1991) 191–207.

[8] W. Dinkelbach, *On nonlinear fractional programming*, Manage. Sci. **13** (1967) 492–498.

[9] E.-D. Dolan, J.-J. Mor, *Benchmarking optimization software with performance profiles*, Math. Program. **91** (2002) 201–213.

[10] J.-Y. Gotoh, A. Takeda, K. Tono, *DC formulations and algorithms for sparse optimization problems*, Math. Program. **169** (2018) 141–176.

[11] F. Hashemi, S. Ketabchi, *Numerical comparisons of smoothing functions for optimal correction of an infeasible system of absolute value equations*, Numer. Algebra, Control. Optim. **10** (2020) 13.

[12] F. Hashemi, S. Ketabchi, l_p *Norm regularization method* $(0 < p < 1)$ *and DC Programming for correction system of inconsistency linear inequalities*, Bull. Iran. Math. Soc. **45** (2019) 865–882.

[13] C. Kanzow, H. Qi, L. Qi, *On the minimum norm solution of linear programs*, J. Optim. Theory Appl. **116** (2003) 333–345.

[14] A. Karppinen, M. Lee, *Hölder continuity of the minimizer of an obstacle problem with generalized Orlicz growth*, arXiv preprint: https://arxiv.org/abs/2006.08244.

[15] S. Ketabchi, H. Moosaei, *An efficient method for optimal correcting of absolute value equations by minimal changes in the right hand side*, Comput. Math. Appl. **64** (2012) 1882–1885.

[16] S. Ketabchi, H. Moosaei, S. Fallahi, 2013. *Optimal error correction of the absolute value equation using a genetic algorithm*, Math. Comput. Model. **57** (2013) 2339–2342.

[17] S. Ketabchi, H. Moosaei, *Minimum norm solution to the absolute value equation in the convex case*, J. Optim. Theory Appl. **154** (2012) 1080-1087.

[18] S. Ketabchi, H. Moosaei, *Optimal error correction and methods of feasible directions*, J. Optim. Theory Appl. **154** (2012) 209–216.

[19] O.-L. Mangasarian, R.-R. Meyer, *Absolute value equations*, Linear Algebra Appl. **419** (2006) 359–367.

[20] H. Moosaei, S. Ketabchi, M. Hladik, *Optimal correction of the absolute value equations*, J. Glob. Optim. **79** (2021) 645–667.

[21] H. Moosaei, S. Ketabchi, P.-M. Pardalos, *Tikhonov regularization for infeasible absolute value equations*, Optimization **65** (2016) 1531–1537.

[22] H. Moosaei, S. Ketabchi, *Optimal correcting of absolute value equations by using smoothing techniques*, J. Interdiscip. Math. **22** (2019) 531–538.

[23] S. Nakayama, J.-Y. Gotoh, *On the superiority of PGMs to PDCAs in nonsmooth nonconvex sparse regression*, Optim. Lett. **15** (2021) 28312860.

[24] N. Parikh, S. Boyd, *Proximal algorithms*, Found. Trends Syst. Control. **1** (2014) 127–239.

[25] C. Popa, C. Serban, *Han-type algorithms for inconsistent systems of linear inequalitiesA unified approach*, Appl. Math. Comput. **246** (2014) 247–256.

[26] M. Radons, S.-M. Rump, *Convergence results for some piecewise linear solvers*, Optim. Lett. 16 (2022) 1663–1673.

[27] M. Salahi, *Optimal correction of infeasible systems in the second order conic linear setting*, Compt. Sci. J. Moldova, **55** (2011) 72–79.

[28] M. Salahi, S. Ketabchi, *Correcting an inconsistent set of linear inequalities by the generalized Newton method*, Optim. Meth. Software **25** (2010) 457–465.

[29] S. Shahsavari, S. Ketabchi, *The proximal methods for solving absolute value equation*, Numer. Algebra Control. Optim. **11** (2021) 449.

[30] G.-B. Thomas, R.-L. Finney, Calculus, Addison-Wesley Publishing Company, 1961.

[31] B. Wen, X. Chen, T.-K. Pong, *A proximal difference-of-convex algorithm with extrapolation*, Comput. Optim. Appl. **69** (2018) 297–324.

[32] L. Yang, L. Wang, *A class of semi-supervised support vector machines by DC programming*, Adv. Data Anal. Classif. **7** (2013) 417–433.

[33] X.-T. Yuan, S. Yan, *Nondegenerate piecewise linear systems: A finite newton algorithm and applications in machine learning*, Neural Comput. **24** (2012) 1047–1084.