Mathematics

http://www.aimspress.com/journal/Math

AIMS Mathematics, 6(8): 8078–8106.
DOI: 10.3934/math.2021469
Received: 21 March 2021
Accepted: 08 May 2021
Published: 24 May 2021

Research article

An efficient DY-type spectral conjugate gradient method for system of nonlinear monotone equations with application in signal recovery

Sani Aji[1,4], Poom Kumam[2,3,*], Aliyu Muhammed Awwal[2,4], Mahmoud Muhammad Yahaya[1] and Kanokwan Sitthithakerngkiet[5]

[1] KMUTT Fixed Point Research Laboratory, Room SCL 802 Fixed Point Laboratory, Science Laboratory Building, Department of Mathematics, Faculty of Science, King Mongkut's University of Technology Thonburi (KMUTT), 126 Pracha-Uthit Road, Bang Mod, Thung Khru, Bangkok 10140, Thailand.

[2] Center of Excellence in Theoretical and Computational Science (TaCS-CoE), Faculty of Science, King Mongkut's University of Technology Thonburi (KMUTT), 126 Pracha Uthit Rd., Bang Mod, Thung Khru, Bangkok 10140, Thailand.

[3] Department of Medical Research, China Medical University Hospital, China Medical University, Taichung 40402, Taiwan.

[4] Department of Mathematics, Faculty of Science, Gombe State University, Gombe 760214, Nigeria.

[5] Intelligent and Nonlinear Dynamic Innovations Research Center, Department of Mathematics, Faculty of Applied Science, King Mongkut's University of Technology North Bangkok (KMUTNB), 1518, Wongsawang, Bangsue, Bangkok 10800, Thailand

* **Correspondence:** Email: poom.kum@kmutt.ac.th; Tel: +6624708994;
Fax: +6624284025.

Abstract: Many problems in engineering and social sciences can be transformed into system of nonlinear equations. As a result, a lot of methods have been proposed for solving the system. Some of the classical methods include Newton and Quasi Newton methods which have rapid convergence from good initial points but unable to deal with large scale problems due to the computation of Jacobian matrix or its approximation. Spectral and conjugate gradient methods proposed for unconstrained optimization, and later on extended to solve nonlinear equations do not require any computation of Jacobian matrix or its approximation, thus, are suitable to handle large scale problems. In this paper, we proposed a spectral conjugate gradient algorithm for solving system of nonlinear equations where the operator under consideration is monotone. The search direction of the proposed algorithm is constructed by taking the convex combination of the Dai-Yuan (DY) parameter and a modified conjugate descent (CD) parameter. The proposed search direction is sufficiently descent and under some suitable assumptions, the global convergence of the proposed algorithm is proved. Numerical experiments on some test problems are presented to show the efficiency of the proposed algorithm in comparison with an existing one. Finally, the algorithm is successfully applied in signal recovery problem arising from compressive sensing.

Keywords: nonlinear monotone equations; conjugate gradient method; spectral conjugate gradient method and large-scale problems
Mathematics Subject Classification: 65K05, 65L09, 90C30

1. Introduction

Consider a nonempty closed convex set $\Omega \subset \mathbb{R}^n$ and a mapping $F : \mathbb{R}^n \to \mathbb{R}^n$ which is continuous and monotone. In this work, our interest is on finding the solution $x \in \Omega$ such that

$$F(x) = 0. \tag{1.1}$$

System of nonlinear equation (1.1) arises in many practical applications such as in controlling the motion of a planar robot manipulator [1], economic equilibrium problems [2], power flow equations [3] and chemical equilibrium systems [4]. Additionally, in mathematics, subproblems in the generalized proximal algorithms with Bregman distance [5] and monotone variational inequality problems by using fixed point map or normal map [6,7] can all be transformed into finding the solution of (1.1). Recently, algorithms for solving system of nonlinear monotone equations are proved to be efficient in signal and image recovery [8].

Due to the existence of nonlinear equations in various fields and their wide range of applications, a lot of methods have been proposed to find their solution. Some of the early and popular iterative methods are Newton method, Quasi-Newton method, Levenberg-Marquardt method and their variants [9–13]. These early methods are characterised by their advantage of rapid convergence from good initial points. However, they require solving linear systems using a Jacobian matrix or its approximation at every iteration. This problem affects their suitability to solve large-scale systems of nonlinear equations.

On the other hand, conjugate gradient methods, spectral gradient methods, and spectral conjugate gradient methods are class of methods for solving large scale unconstrained optimization problems. Among the advantages of these methods are their simplicity in implementation and low storage requirements. These advantages stimulate researchers to extend these methods in order to solve nonlinear equations. For example, the projection technique proposed by Solodov and Svaiter [14] motivated many researchers to extend conjugate gradient methods from solving unconstrained optimization problem to solve system of nonlinear equations. Inspired by the work of Solodov and Svaiter in [14], Wang et al. [15] proposed a projection method for solving system of nonlinear monotone equations. In their algorithm, a linear system of equations is solved approximately, at each iteration, to obtain a trial point and then a line search strategy is performed along the search direction determined by the current point and the trial point with the aim of getting a predictor-corrector point. Hence, the algorithm computes its next iterate by projection. They proved the global convergence of the proposed method and presented some numerical experiments in order to show the performance of the algorithm. In [16], Cheng combined the classical PRP method with hyperplane projection method to solve nonlinear monotone equations. Xiao and Zhou [17] extended the well-known CG_Descent for unconstrained minimization problems to solve large scale convex constraint nonlinear monotone equations. They achieved this by combining the CG_Descent with the projection technique in [14]. They proved the global convergence of this method and showed the numerical performance. Liu and Li [18] modified the work in [17] and proposed another extension of the CG_descent method to solve nonlinear monotone equations. Based on the popular Dai-Yuan (DY) conjugate gradient parameter [19], Liu [20] proposed a spectral DY-type method for solving nonlinear monotone equations. The method can be viewed as a combination of the DY conjugate gradient method, spectral gradient method and the projection technique. They showed that the method converges globally and presented some numerical experiments. Later on, Liu and Li [21] developed another DY-type algorithm, which is a multivariate spectral method for solving (1.1). In their work, the direction uses a combination of the multivariate spectral gradient method and DY conjugate gradient parameter. The numerical experiments of the method is reported and the global convergence is also proved. However, restriction is imposed on the lipschitz constant $L < 1 - r$ with $r \in (0, 1)$ before proving the global convergence. Motivated by this work, Liu and Feng [22] proposed another spectral conjugate gradient method for solving (1.1). Their work improved the computational effect of the DY conjugate gradient method and under some assumptions both the global and linear convergence of the method is proved. Most recently, a lot of algorithms have been developed for solving (1.1). Some of these algorithms can be found in [23–32].

In this paper, motivated by the work of Liu and Feng [22] on the modification of DY conjugate gradient method, we propose an efficient spectral conjugate gradient algorithm for solving systems of nonlinear monotone equations with convex constraint. The search direction in our proposed approach uses a convex combination of the DY parameter and a modified CD parameter. Specifically, this paper gives the following contributions:

- We propose an efficient spectral conjugate gradient algorithm for solving systems of nonlinear monotone equations with convex constraint by taking the convex combination of the DY parameter and a modified CD parameter.
- This algorithm can be viewed as an extension of the work proposed by Yu et al in [33].
- The global convergence of the proposed algorithm is proved under some suitable assumptions.
- The proposed algorithm is applied to recover a distorted signal.

The organization of the paper is as follows: In the next section, we introduce the details of the algorithm, some important definitions and prove global convergence. In the third section, we provide numerical experiments of the proposed algorithm and compare it performance with an existing one. Finally, we apply the algorithm in signal recovery, and give conclusion in the last section.

2. Algorithm: Motivation and convergence result

In this section, projection map, its properties, and some important assumptions needed for the convergence analysis are introduced.

Definition 2.1. *Let $\Omega \subset \mathbb{R}^n$ be a nonempty, closed and convex set. The projection of any $x \in \mathbb{R}^n$ onto*

Ω *is*

$$P_\Omega(x) = \arg\min\{\|x - y\| \ : y \in \Omega\}.$$

The projection map has the following property:

$$\|P_\Omega(x) - y\| \leq \|x - y\|, \quad \forall x, y \in \mathbb{R}^n. \tag{2.1}$$

Spectral and conjugate gradient algorithms generate sequence of iterates using the following formula:

$$x_{k+1} = x_k + \alpha_k d_k, \tag{2.2}$$

where α_k is called the step length, and the direction d_k defined in spectral and conjugate gradient method respectively as:

$$d_k = \begin{cases} -F_k, & \text{if } k = 0, \\ -\nu_k F_k, & \text{if } k \geq 1, \end{cases} \tag{2.3}$$

and

$$d_k = \begin{cases} -F_k, & \text{if } k = 0, \\ -F_k + \beta_k d_{k-1}, & \text{if } k \geq 1. \end{cases} \tag{2.4}$$

Different spectral and conjugate gradient directions are developed using different choice of the parameters ν_k and β_k respectively. To ensure the global convergence of these methods, the direction d_k needs to satisfy the sufficient decent property. That is:

$$F_k^T d_k \leq -\tau\|F_k\|, \tag{2.5}$$

where $\tau > 0$.

One of the well-known parameter proposed in this direction is the DY conjugate gradient parameter in [19] defined as

$$\beta_k^{DY} = \frac{\|F_k\|^2}{Y_{k-1}^T d_{k-1}}, \tag{2.6}$$

such that the direction in (2.4) becomes:

$$d_k = \begin{cases} -F_k, & \text{if } k = 0, \\ -F_k + \frac{\|F_k\|^2}{Y_{k-1}^T d_{k-1}} d_{k-1}, & \text{if } k \geq 1, \end{cases} \tag{2.7}$$

where $Y_{k-1} = F_k - F_{k-1}$. Unfortunately, (2.7) does not satisfy the decency property (2.5). As a result, Liu and Feng [22] modified the work in [19] and proposed a spectral conjugate gradient algorithm for solving (1.1). The direction in [22] satisfies (2.5) and thus, the global convergence is proved successfully under some appropriate assumptions.

Motivated by the work in [22], and due to the limited number of DY-type conjugate gradient methods in literature, we proposed a new spectral DY-type conjugate gradient algorithm for solving (1.1). Interestingly, the proposed direction satisfies the sufficient descent property (2.5), and uses a convex combination of the DY parameter and a modified CD parameter as follows:

$$d_k = \begin{cases} -F_k, & \text{if } k = 0, \\ -\nu_k F_k + [(1 - \theta_k)\beta_k^{DY} + \theta_k \tilde{\beta}_k] d_{k-1}, & \text{if } k \geq 1, \end{cases} \tag{2.8}$$

where

$$\nu_k = \frac{s_{k-1}^T s_{k-1}}{s_{k-1}^T y_{k-1}},\ y_{k-1} = F(x_k) - F(x_{k-1}) + r s_{k-1},\ s_{k-1} = x_k - x_{k-1}, \tag{2.9}$$

β_k^{DY} is defined as (2.6), $\tilde{\beta}_k = \frac{\|F_k\|^2}{\max\{-F_k^T d_{k-1}, \gamma\|d_{k-1}\|\}}$ and $\theta_k \in (0,1)$. Substituting the value of β_k^{DY} and $\tilde{\beta}_k$ in (2.8) we get

$$d_k = \begin{cases} -F_k, & \text{if } k = 0 \\ -\nu_k F_k + \left[(1-\theta_k)\frac{\|F_k\|^2}{Y_{k-1}^T d_{k-1}} + \theta_k \frac{\|F_k\|^2}{\max\{-F_k^T d_{k-1}, \gamma\|d_{k-1}\|\}}\right] d_{k-1} & \text{if } k \geq 1. \end{cases} \tag{2.10}$$

Throughout this work, the following assumptions are made.

- ($Assumption_1$) The mapping F is monotone, that is,

$$(F(x) - F(y))^T (x - y) \geq 0, \quad \forall x, y \in \mathbb{R}^n.$$

- ($Assumption_2$) The mapping F is Lipschitz continuous, that is there exists $L > 0$ such that

$$\|F(x) - F(y)\| \leq L\|x - y\|,\ \forall x, y \in \mathbb{R}^n.$$

- ($Assumption_3$) The solution set of (1.1), denoted by Ω, is nonempty.

We state the steps of our proposed algorithm as follows:

Algorithm 2.2. ***Step 0.*** *Choose initial point* $x_0 \in \Omega$, $\theta_k \in (0,1), \kappa \in (0,1], \beta \in (0,1)$ $\mu > 1$, σ, $\gamma > 0$, $\delta \in (0,2)$, *and* $Tol > 0$. *Set* $k := 0$.
Step 1. *If* $\|F_k\| \leq Tol$, *stop, otherwise proceed with Step 2.*
Step 2. *Compute* $d_k = -\|F_k\|$, $k = 0$ *and*

$$d_k = \begin{cases} -\nu_k F_k, & \text{if } Y_{k-1}^T d_{k-1} \leq \mu\|F_k\|\|d_{k-1}\|, \\ -\nu_k F_k + \left[(1-\theta_k)\frac{\|F_k\|^2}{Y_{k-1}^T d_{k-1}} + \theta_k \frac{\|F_k\|^2}{\max\{-F_k^T d_{k-1}, \gamma\|d_{k-1}\|\}}\right] d_{k-1}, & \text{otherwise}. \end{cases} \tag{2.11}$$

Step 3. *Compute* $\Lambda_k = \max\{\kappa\beta^i \ : i = 0, 1, 2, \ldots\}$ *such that*

$$-\langle F(x_k + \kappa\beta^i d_k), d_k\rangle \geq \sigma\kappa\beta^i \|d_k\|^2 \min\left\{1,\ \|F(x_k + \kappa\beta^i d_k)\|^{\frac{1}{c}}\right\},\ \ c \geq 1. \tag{2.12}$$

Step 4. *Set* $z_k = x_k + \Lambda_k d_k$. *If* $\|F(z_k)\| = 0$, *stop. Else compute*

$$x_{k+1} = P_\Omega\left[x_k - \delta\frac{F(z_k)^T(x_k - z_k)}{\|F(z_k)\|^2} F(z_k)\right].$$

Step 5. *Let* $k = k + 1$ *and go to Step 1.*

Remark 2.3. *It is worth noting that when* $Y_{k-1}^T d_{k-1} \leq \mu\|F_k\|\|d_{k-1}\|$, *our proposed search direction reduces to that of Yu et al. proposed in [33]. Thus, as a contribution, this work can be viewed as an extension of the work of Yu et al. [33].*

We now state and proof the following Lemmas and Theorem for the convergence.

Lemma 2.4. *The parameter ν_k given by* (2.9) *is well defined, and $\forall k \geq 0$, d_k satisfies*

$$F_k^T d_k \leq -\tau ||F_k||^2. \tag{2.13}$$

Proof. Since F is monotone, then

$$\langle F(x_k) - F(x_{k-1}), x_k - x_{k-1} \rangle \geq 0,$$

which yields

$$\langle y_{k-1}, s_{k-1} \rangle \geq r||s_{k-1}||^2. \tag{2.14}$$

Again, by Lipschitz continuity, we have

$$\langle y_{k-1}, s_{k-1} \rangle = \langle F(x_k) - F(x_{k-1}), s_{k-1} \rangle + r||s_{k-1}||^2 \leq (L + r)||s_{k-1}||^2. \tag{2.15}$$

From (2.14) and (2.15) we get

$$\frac{1}{(L + r)} \leq \nu_k \leq \frac{1}{r}. \tag{2.16}$$

Now, to show (2.13), for $k = 0$, $F_k^T d_k = -||F_k||^2$, thus $\tau = 1$ and the result holds. When $k \neq 0$, If $Y_{k-1}^T d_{k-1} \leq \mu ||F_k|| ||d_{k-1}||$, then from (2.11),

$$F_k^T d_k = -\nu_k ||F_k||^2,$$

using (2.16), we have

$$F_k^T d_k \leq -\frac{1}{(L + r)} ||F_k||^2,$$

and (2.13) holds by taking $\tau = \frac{1}{L+r}$.

On the other hand, if $Y_{k-1}^T d_{k-1} > \mu ||F_k|| ||d_{k-1}||$, multiplying (2.11) by F_k^T we obtain

$$\begin{aligned}
F_k^T d_k =& -\nu_k ||F_k||^2 + \left[(1 - \theta_k) \frac{||F_k||^2}{Y_{k-1}^T d_{k-1}} + \theta_k \frac{||F_k||^2}{\max\{-F_k^T d_{k-1}, \gamma ||d_{k-1}||\}} \right] F_k^T d_{k-1} \\
&\leq -\nu_k ||F_k||^2 + (1 - \theta_k) \frac{||F_k||^2}{Y_{k-1}^T d_{k-1}} F_k^T d_{k-1} + \theta_k \frac{||F_k||^2 F_k^T d_{k-1}}{-F_k^T d_{k-1}} \\
&= -\nu_k ||F_k||^2 + (1 - \theta_k) \frac{||F_k||^2}{Y_{k-1}^T d_{k-1}} F_k^T d_{k-1} - \theta_k ||F_k||^2 \\
&\leq -\nu_k ||F_k||^2 + (1 - \theta_k) \frac{||F_k||^2}{Y_{k-1}^T d_{k-1}} F_k^T d_{k-1} \\
&\leq -\nu_k ||F_k||^2 + \frac{||F_k||^2}{Y_{k-1}^T d_{k-1}} F_k^T d_{k-1} \\
&= -\nu_k ||F_k||^2 + \frac{F_k^T d_{k-1}}{Y_{k-1}^T d_{k-1}}]||F_k||^2 \\
&\leq -\nu_k ||F_k||^2 + \frac{||F_k|| ||d_{k-1}||}{Y_{k-1}^T d_{k-1}} ||F_k||^2 \text{(by cauchy-schwarz inequality)} \\
&< -\nu_k ||F_k||^2 + \frac{||F_k|| ||d_{k-1}||}{\mu ||F_k|| ||d_{k-1}||} ||F_k||^2 \\
&\leq -\left[\frac{1}{(L + r)} - \frac{1}{\mu} \right] ||F_k||^2.
\end{aligned} \tag{2.17}$$

The last inequality follows from (2.16). By letting $\tau = (\frac{1}{L+r} - \frac{1}{\mu}) > 0$, the required result holds. $\square$

Lemma 2.5. *Let* $\{d_k\}$ *be given by* (2.11)*, then there are some constants* $p_1 > 0$, $m_1 > 0$ *and* $m_2 > 0$ *for which*

$$\|d_k\| \le \begin{cases} p_1\|F_k\|, & \text{if } Y_{k-1}^T d_{k-1} \le \mu\|F_k\|\|d_{k-1}\|, \\ m_1\|F_k\| + m_2\|F_k\|^2, & \text{otherwise}. \end{cases} \tag{2.18}$$

Proof. If $Y_{k-1}^T d_{k-1} \le \mu\|F_k\|\|d_{k-1}\|$,

$$\|d_k\| = \nu_k\|F_k\|.$$

Using (2.16), we have

$$\|d_k\| \le p_1\|F_k\|,$$

where $p_1 = \dfrac{1}{r}$. However, if $Y_{k-1}^T d_{k-1} > \mu\|F_k\|\|d_{k-1}\|$, then

$$\begin{aligned} \|d_k\| &= \nu_k\|F_k\| + (1-\theta_k)\frac{\|F_k\|^2\|d_{k-1}\|}{|Y_{k-1}^T d_{k-1}|} + \theta_k\frac{\|F_k\|^2\|d_{k-1}\|}{\max\{-F_k^T d_{k-1}, \gamma\|d_{k-1}\|\}} \\ &\le \nu_k\|F_k\| + \frac{\|F_k\|^2\|d_{k-1}\|}{|Y_{k-1}^T d_{k-1}|} + \frac{\|F_k\|^2\|d_{k-1}\|}{\gamma\|d_{k-1}\|} \\ &\le \nu_k\|F_k\| + \frac{\|F_k\|^2\|d_{k-1}\|}{\mu\|F_k\|\|d_{k-1}\|} + \frac{\|F_k\|^2}{\gamma} \\ &\le (\nu_k + \frac{1}{\mu})\|F_k\| + \frac{1}{\gamma}\|F_k\|^2 \\ &\le (p_1 + \frac{1}{\mu})\|F_k\| + \frac{1}{\gamma}\|F_k\|^2 \\ &\le m_1\|F_k\| + m_2\|F_k\|^2, \end{aligned} \tag{2.19}$$

where $m_1 = p_1 + \frac{1}{\mu}$, $p_1 = \frac{1}{r}$ and $m_2 = \frac{1}{\gamma}$. $\square$

Lemma 2.6. *Suppose (Assumption$_1$) - (Assumption$_3$) hold, then the sequences* $\{x_k\}$ *and* $\{z_k\}$ *generated by Algorithm 2.2 are bounded. Also,*

$$\lim_{k\to\infty} \Lambda_k\|d_k\| = 0, \tag{2.20}$$

and

$$\lim_{k\to\infty} \|x_{k+1} - x_k\| = 0. \tag{2.21}$$

Proof. Let $\tilde{x}$ be a solution of problem (1.1), using monotonicity from Assumption 1 we have

$$\begin{aligned} \langle F(z_k),\ x_k - \tilde{x}\rangle &= \langle F(z_k),\ x_k - z_k + z_k - \tilde{x}\rangle \\ &= \langle F(z_k),\ x_k - z_k\rangle + \langle F(z_k) - F(\tilde{x}),\ z_k - \tilde{x}\rangle \\ &\ge \langle F(z_k),\ x_k - z_k\rangle. \end{aligned} \tag{2.22}$$

Using the above Eq (2.22) and x_{k+1} we obtain

$$\|x_{k+1} - \tilde{x}\|^2 = \left\| P_\Omega\left[x_k - \delta\frac{\langle F(z_k),\ x_k - z_k\rangle}{\|F(z_k)\|^2}F(z_k)\right] - \tilde{x}\right\|^2$$

$$
\begin{aligned}
&\leq \left\| x_k - \tilde{x} - \delta \frac{\langle F(z_k),\ x_k - z_k \rangle}{\|F(z_k)\|^2} F(z_k) \right\|^2 \\
&= \|x_k - \tilde{x}\|^2 - 2\delta \frac{\langle F(z_k),\ x_k - z_k \rangle}{\|F(z_k)\|^2} \langle F(z_k),\ x_k - \tilde{x} \rangle + \delta^2 \frac{\langle F(z_k),\ x_k - z_k \rangle^2}{\|F(z_k)\|^2} \\
&\leq \|x_k - \tilde{x}\|^2 - 2\delta \frac{\langle F(z_k),\ x_k - z_k \rangle}{\|F(z_k)\|^2} \langle F(z_k),\ x_k - z_k \rangle + \delta^2 \frac{\langle F(z_k),\ x_k - z_k \rangle^2}{\|F(z_k)\|^2} \\
&= \|x_k - \tilde{x}\|^2 - \delta(2-\delta) \frac{\langle F(z_k),\ x_k - z_k \rangle^2}{\|F(z_k)\|^2} \qquad (2.23) \\
&\leq \|x_k - \tilde{x}\|^2.
\end{aligned}
$$

Showing that $\|x_k - \tilde{x}\| \leq \|x_0 - \tilde{x}\|$ for all k and hence $\{x_k\}$ is bounded and $\lim\limits_{k\to\infty} \|x_k - \tilde{x}\|$ exists. Since $\{x_k\}$ is bounded, and F is Lipschitz continuous,

$$\|F(x_k)\| \leq p,\ \ p > 0. \tag{2.24}$$

Using this and (2.18), we have

$$\|d_k\| \leq \begin{cases} n_1, & \text{if } \ Y_{k-1}^T d_{k-1} \leq \mu F_k^T d_{k-1}, \\ n_2, & \text{otherwise}, \end{cases} \tag{2.25}$$

where $n_1 = p_1 p$ and $n_2 = m_1 p + m_2 p^2$ and taking $M = \min\{n_1, n_2\}$, we have that the direction d_k is bounded. That is

$$\|d_k\| \leq M,\ \ M > 0. \tag{2.26}$$

To prove that $\{z_k\}$ is bounded, we know that

$$z_k - x_k = \Lambda_k d_k,$$

and since we have proved that d_k is bounded. This implies $\{z_k\}$ is also bounded. Again, by Lipschitz continuity,

$$\|F(z_k)\| \leq n,\ \ n > 0. \tag{2.27}$$

Now from our line search (2.12), let $\min\{1,\ \|F(x_k + \kappa\beta^i d_k)\|^{\frac{1}{c}}\} = \|F(x_k + \kappa\beta^i d_k)\|^{\frac{1}{c}}$, squaring from both sides of (2.12) we get

$$\sigma^2 \Lambda_k^4 \|d_k\|^4 \|F(z_k)\|^{\frac{2}{c}} \leq \langle F(z_k),\ \Lambda_k d_k \rangle^2. \tag{2.28}$$

Also, since $0 < \delta < 2$, then from (2.23) we have

$$\langle F(z_k),\ x_k - z_k \rangle^2 \leq \frac{\|F(z_k)\|^2 (\|x_k - \tilde{x}\|^2 - \|x_{k+1} - \tilde{x}\|^2)}{\delta(2-\delta)}. \tag{2.29}$$

This together with (2.28) gives

$$\sigma^2 \Lambda_k^4 \|d_k\|^4 \|F(z_k)\|^{\frac{2}{c}} \leq \frac{\|F(z_k)\|^2 (\|x_k - \tilde{x}\|^2 - \|x_{k+1} - \tilde{x}\|^2)}{\delta(2-\delta)}. \tag{2.30}$$

Since $\lim\limits_{k\to\infty} \|x_k - \tilde{x}\|$ exists and that (2.27) holds, taking the limit as $k \to \infty$ on both sides of (2.30) we have

$$\sigma^2 \lim_{k\to\infty} \Lambda_k^4 \|d_k\|^4 \|F(z_k)\|^{\frac{2}{c}} = 0, \tag{2.31}$$

but $\|F(z_k)\| \neq 0$, therefore,

$$\lim_{k\to\infty} \Lambda_k \|d_k\| = 0. \tag{2.32}$$

Note that if the $\min\{1, \|F(x_k + \Lambda_k d_k)\|^{\frac{1}{c}}\} = 1$, then, (2.30) becomes

$$\sigma^2 \Lambda_k^4 \|d_k\|^4 \leq \frac{\|F(z_k)\|^2 (\|x_k - \tilde{x}\|^2 - \|x_{k+1} - \tilde{x}\|^2)}{\delta(2-\delta)}, \tag{2.33}$$

and thus (2.32) holds.

Using this and the definition of z_k, we obtain

$$\lim_{k\to\infty} \|z_k - x_k\| = 0. \tag{2.34}$$

From the definition of projection operation, we get

$$\begin{aligned}
\lim_{k\to\infty} \|x_{k+1} - x_k\| &= \lim_{k\to\infty} \left\| P_\Omega \left[x_k - \delta \frac{\langle F(z_k),\ x_k - z_k \rangle}{\|F(z_k)\|^2} F(z_k) \right] - x_k \right\| \\
&\leq \lim_{k\to\infty} \left\| x_k - \delta \frac{\langle F(z_k),\ x_k - z_k \rangle}{\|F(z_k)\|^2} F(z_k) - x_k \right\| \\
&\leq \delta \lim_{k\to\infty} \|x_k - z_k\| \\
&= 0.
\end{aligned} \tag{2.35}$$

□

Lemma 2.7. *Suppose (Assumption$_2$) holds, and the sequences $\{x_k\}$ and $\{z_k\}$ are generated by Algorithm 2.2. Then*

$$\Lambda_K \geq \max\left\{\kappa, \frac{\tau\beta\|F_k\|^2}{(L+\sigma)\|d_k\|^2}, \frac{\tau\beta\|F_k\|^2}{(L + \sigma\|F(x_k + \kappa\beta^{i-1}d_k)\|^{\frac{1}{c}})\|d_k\|^2}\right\}. \tag{2.36}$$

Proof. From (2.12), if $\Lambda_k \neq \kappa$, then $\widehat{\Lambda}_k = \Lambda_k\beta^{-1}$ does not satisfy (2.12), that is,

$$-\langle F(x_k + \widehat{\Lambda}_k d_k), d_k\rangle < \sigma\|d_k\|^2 \widehat{\Lambda}_k \min\{1, \|F(x_k + \widehat{\Lambda}_k d_k)\|^{\frac{1}{c}}\}.$$

Now let the $\min\{1, \|F(x_k + \widehat{\Lambda}_k d_k)\|^{\frac{1}{c}}\} = \|F(x_k + \widehat{\Lambda}_k d_k)\|^{\frac{1}{c}}$. Using (2.13) and (*Assumption*$_2$), we have

$$\begin{aligned}
\tau\|F_k\|^2 &\leq -F_k^T d_k \\
&= (F(x_k + \widehat{\Lambda}_k d_k) - F_k)^T d_k - \langle F(x_k + \widehat{\Lambda}_k d_k), d_k\rangle \\
&\leq \|F(x_k + \widehat{\Lambda}_k d_k) - F(x_k)\|\|d_k\| - \langle F(x_k + \widehat{\Lambda}_k d_k), d_k\rangle \\
&\leq L\|x_k + \widehat{\Lambda}_k d_k - x_k\|\|d_k\| + \sigma\widehat{\Lambda}_k\|d_k\|^2\|F(x_k + \widehat{\Lambda}_k d_k)\|^{\frac{1}{c}} \\
&\leq \widehat{\Lambda}_k L\|d_k\|^2 + \sigma\widehat{\Lambda}_k\|d_k\|^2\|F(x_k + \widehat{\Lambda}_k d_k)\|^{\frac{1}{c}} \\
&\leq \widehat{\Lambda}_k\|d_k\|^2(L + \sigma\|F(x_k + \widehat{\Lambda}_k d_k)\|^{\frac{1}{c}}).
\end{aligned}$$

Therefore,

$$\widehat{\Lambda}_k \geq \frac{\tau||F_k||^2}{(L+\sigma||F(x_k+\widehat{\Lambda}_k d_k)||^{\frac{1}{c}})||d_k||^2}, \tag{2.37}$$

substituting $\widehat{\Lambda}_k = \Lambda_k\beta^{-1}$ and solving for Λ_k we get

$$\Lambda_k \geq \frac{\tau\beta||F_k||^2}{(L+\sigma||F(x_k+\kappa\beta^{i-1}d_k)||^{\frac{1}{c}})||d_k||^2}. \tag{2.38}$$

□

On the other hand, if $\min\{1, ||F(x_k+\widehat{\Lambda}_k d_k)||^{\frac{1}{c}}\} = 1$, then (2.38) reduces to

$$\Lambda_k \geq \frac{\tau\beta||F_k||^2}{(L+\sigma)||d_k||^2}. \tag{2.39}$$

Combining (2.38) and (2.39), we get

$$\Lambda_K \geq \max\left\{\kappa, \frac{\tau\beta||F_k||^2}{(L+\sigma)||d_k||^2}, \frac{\tau\beta||F_k||^2}{(L+\sigma||F(x_k+\kappa\beta^{i-1}d_k)||^{\frac{1}{c}})||d_k||^2}\right\}. \tag{2.40}$$

Theorem 2.8. *Suppose that (Assumption$_1$-Assumption$_3$) hold and let the sequence $\{x_k\}$ be generated by Algorithm 2.2, then*

$$\liminf_{k\to\infty} ||F(x_k)|| = 0. \tag{2.41}$$

Proof. We prove by contradiction. Suppose (2.41) is not satisfied, then there exists $\alpha > 0$ such that $\forall k \geq 0$,

$$||F(x_k)|| \geq \alpha. \tag{2.42}$$

From Eqs (2.13) and (2.42), we obtain $\forall k \geq 0$,

$$||d_k|| \geq \tau\alpha. \tag{2.43}$$

We multiply $||d_k||$ on both sides of (2.36), and from (2.26) and (2.42), we get

$$\begin{aligned}\Lambda_k||d_k|| &\geq \max\left\{\kappa, \frac{\tau\beta||F_k||^2}{(L+\sigma)||d_k||^2}, \frac{\tau\beta||F_k||^2}{(L+\sigma||F(x_k+\kappa\beta^{i-1}d_k)||^{\frac{1}{c}})||d_k||^2}\right\}||d_k|| \\ &\geq \max\left\{\alpha, \frac{\tau\beta\alpha^2}{(L+\sigma)M}, \frac{\tau\beta\alpha^2}{(L+\sigma||F(x_k+\kappa\beta^{i-1}d_k)||^{\frac{1}{c}})M}\right\}.\end{aligned} \tag{2.44}$$

Taking limit as $k \to \infty$ on both sides, we obtain

$$\lim_{k\to\infty} \Lambda_k||d_k|| > 0, \tag{2.45}$$

which contradicts Eq (2.32). Therefore,

$$\liminf_{k\to\infty} ||F(x_k)|| = 0. \tag{2.46}$$

□

3. Numerical experiments

In this section, we give the numerical experiments in order to depict the advantages and the performance of our proposed algorithm (MDY) in comparison with the projected Dai-Yuan derivative-free algorithm (PDY) by Liu and Feng [22]. All codes are written on Matlab R2019b and run on a PC of corei3-4005U processor, 4 GB RAM and 1.70 GHZ CPU.

In MDY, the parameters are choosen as follows: $r = 0.001$, $\theta_k = 1/(k+1)$, $\mu = 1.9$, $\gamma = 0.9$, $\sigma = 0.02$, $c = 2$, $\kappa = 1, \beta = 0.70$ and $\delta = 1.1$. The parameters in the PDY algorithm are maintained as exactly as they are reported in [22]. Based on this setting, we consider nine test problems with eight different initial points and tested them on five different dimensions, $n = 1000, n = 5000, n = 10000, n = 50000$ and $n = 100000$. We used $\|F_k\| < 10^{-6}$ as stopping criteria and denoted failure by "-" whenever the number of iterations exceeds 1000 and the stopping criterion is not satisfied. The test problems are listed below, where the function F is taken as $F(x) = (f_1(x), f_2(x), \dots, f_n(x))^T$.

Problem 1 [26].

$$f_1(x) = e^{x_1} - 1$$
$$f_i(x) = e^{x_i} + x_i - 1, \text{ for } i = 1, 2, ..., n, \text{ and } \quad \Omega = \mathbb{R}^n_+.$$

Problem 2 [34] Modified Logarithmic Function.

$$f_i(x) = \ln(x_i + 1) - \frac{x_i}{n}, \text{ for } i = 2, 3, ..., n,$$
$$\text{and } \Omega = \{x \in \mathbb{R}^n : \sum_{i=1}^{n} x_i \leq n, x_i > -1, i = 1, 2, \dots, n\}.$$

Problem 3 [35] Nonsmooth Function.

$$f_i(x) = 2x_i - \sin|x_i|, \ i = 1, 2, 3, ..., n,$$
$$\text{and } \Omega = \{x \in \mathbb{R}^n : \sum_{i=1}^{n} x_i \leq n, x_i \geq 0, i = 1, 2, \dots, n\}.$$

Problem 4 [36]

$$f_i(x) = \min\left\{\min\{|x_i|, x_i^2\}, \max\{|x_i|, x_i^3\}\right\}$$
$$\text{for } i = 1, 2, 3, ..., n \quad \text{and} \quad \Omega = \mathbb{R}^n_+.$$

Problem 5 [37] Strictly Convex Function.

$$f_i(x) = e^{x_i} - 1, \text{ for } i = 1, 2, ..., n,$$
$$\text{and } \Omega = \mathbb{R}^n_+.$$

Problem 6

$$f_i(x) = \frac{i}{n} e^{x_i} - 1, \text{ for } i = 1, 2, ..., n,$$
$$\text{and } \Omega = \mathbb{R}^n_+.$$

Problem 7 [38] Tridiagonal Exponential Function

$$
\begin{aligned}
f_1(x) &= x_1 - e^{\cos(h(x_1+x_2))},\\
f_i(x) &= x_i - e^{\cos(h(x_{i-1}+x_i+x_{i+1}))}, \text{ for } i = 2, ..., n-1,\\
f_n(x) &= x_n - e^{\cos(h(x_{n-1}+x_n))},\\
h &= \frac{1}{n+1} \text{ and } \Omega = \mathbb{R}^n_+.
\end{aligned}
$$

Problem 8 [22]

$$
\begin{aligned}
f_1(x) &= \frac{5}{2}x_1 + x_2 - 1,\\
f_i(x) &= x_{i-1} + \frac{5}{2}x_i + x_{i+1} - 1, \text{ for } i = 1, 2, ..., n,\\
f_n(x) &= x_{n-1} + \frac{5}{2}x_n - 1 \text{ and } \Omega = \mathbb{R}^n_+.
\end{aligned}
$$

Problem 9 [26]

$$
\begin{aligned}
&f_i(x) = e^{{x_i}^2} + 1.5\sin(2x_i) - 1, \text{ for } i = 1, 2, ..., n,\\
&\text{and } \Omega = \mathbb{R}^n_+.
\end{aligned}
$$

The results of our experiments are shown in Tables 1–9 based on the number of iterations denoted as (ITER), number of function evaluations (FVAL), CPU time (TIME), and the norm of the function (NORM) when the solution was obtained. Looking at the reported results, it can be observed that the proposed MDY algorithm outperformed the PDY algorithm in most of the problems by having the least ITER, FVAL and TIME.

Table 1. Numerical results of the PDY and MDY algorithms on Problem 1 with given initial points and dimensions.

		PDY				MDY			
DIMENSION	INITIAL POINT	ITER	FVAL	TIME	NORM	ITER	FVAL	TIME	NORM
1000	x1	26	54	0.56086	2.43E-07	1	2	0.0171	0.00E+00
	x2	36	74	0.11293	3.49E-07	3	4	0.0523	0.00E+00
	x3	42	85	0.091373	8.84E-07	6	7	0.0311	1.99E-14
	x4	51	104	0.11197	9.89E-07	9	10	0.0293	0.00E+00
	x5	24	50	0.13649	7.08E-07	9	10	0.0392	0.00E+00
	x6	40	82	0.061757	3.16E-07	11	12	0.0155	0.00E+00
	x7	51	104	0.096776	9.89E-07	9	10	0.0157	0.00E+00
	x8	33	68	0.065298	7.09E-07	8	9	0.0179	0.00E+00
5000	x1	34	70	0.61141	9.63E-09	1	2	0.0561	0.00E+00
	x2	27	56	0.34817	1.30E-08	4	5	0.0586	0.00E+00
	x3	42	85	1.2759	8.84E-07	6	7	0.0296	1.99E-14
	x4	53	107	0.2958	7.98E-07	8	9	0.0441	0.00E+00
	x5	25	52	0.37603	7.92E-07	8	9	0.0369	0.00E+00
	x6	36	74	0.35124	3.44E-07	11	12	0.0472	0.00E+00
	x7	52	106	1.2879	7.98E-07	8	9	0.1098	0.00E+00
	x8	30	62	0.13964	7.92E-07	8	9	0.0776	0.00E+00
10000	x1	33	68	1.0889	6.22E-09	1	2	0.0153	0.00E+00
	x2	29	60	0.23701	2.14E-08	8	9	0.0538	4.54E-07
	x3	42	85	1.305	8.84E-07	6	7	0.0678	1.99E-14
	x4	47	95	1.5416	5.62E-07	8	9	0.0616	0.00E+00
	x5	26	54	0.23998	5.60E-07	8	9	0.0622	0.00E+00
	x6	41	84	0.38904	1.10E-07	11	12	0.0844	0.00E+00
	x7	47	95	1.2676	5.62E-07	8	9	0.3386	0.00E+00
	x8	29	60	0.27526	5.60E-07	8	9	0.1665	0.00E+00
50000	x1	37	76	1.4006	7.60E-09	1	2	0.0489	0.00E+00
	x2	28	58	0.83132	8.48E-08	3	4	0.0982	0.00E+00
	x3	42	85	2.0342	8.84E-07	6	7	0.1496	1.99E-14
	x4	51	104	1.9564	6.27E-07	8	9	0.2306	0.00E+00
	x5	27	56	2.007	6.26E-07	8	9	0.2302	0.00E+00
	x6	45	92	1.4682	2.84E-07	11	12	0.8319	0.00E+00
	x7	51	104	1.7938	6.27E-07	8	9	0.2874	0.00E+00
	x8	27	56	0.79777	6.26E-07	8	9	0.2378	0.00E+00
100000	x1	32	66	1.8828	3.23E-07	1	2	0.0875	0.00E+00
	x2	21	44	2.7454	7.70E-07	3	4	0.2164	0.00E+00
	x3	42	85	2.4577	8.84E-07	6	7	0.3210	1.99E-14
	x4	51	104	3.228	7.08E-07	8	9	0.4317	0.00E+00
	x5	28	58	1.9717	7.07E-07	8	9	1.0072	0.00E+00
	x6	32	66	1.6766	7.38E-07	11	12	0.6324	0.00E+00
	x7	51	104	2.8174	7.08E-07	8	9	0.4355	0.00E+00
	x8	28	58	1.9569	7.07E-07	8	9	0.4106	0.00E+00

Table 2. Numerical results of the PDY and MDY algorithms on Problem 2 with given initial points and dimensions.

		PDY				**MDY**			
DIMENSION	INITIAL POINT	ITER	FVAL	TIME	NORM	ITER	FVAL	TIME	NORM
1000	x1	4	10	0.0685	3.60E-08	9	10	0.0090	7.50E-07
	x2	2	6	0.0092	5.17E-07	6	7	0.0062	4.70E-08
	x3	18	38	0.0273	4.14E-07	11	12	0.0199	2.22E-07
	x4	27	56	0.0351	1.81E-07	13	14	0.0205	5.83E-08
	x5	27	56	0.0233	1.81E-07	13	14	0.0262	5.83E-08
	x6	20	42	0.0763	8.06E-07	11	12	0.0159	2.39E-07
	x7	27	56	0.0610	1.81E-07	13	14	0.0317	5.83E-08
	x8	23	48	0.0350	4.78E-07	13	14	0.0121	6.03E-08
5000	x1	4	10	0.1548	6.26E-09	9	10	0.0423	6.55E-07
	x2	2	6	0.1020	1.75E-07	8	9	0.0420	1.54E-07
	x3	30	62	0.3021	1.54E-07	11	12	0.0947	9.29E-08
	x4	22	46	0.4080	8.16E-07	15	16	0.0675	1.14E-07
	x5	22	46	0.5070	8.16E-07	15	16	0.0639	1.14E-07
	x6	18	38	0.0891	5.88E-08	11	12	0.0546	5.43E-07
	x7	22	46	0.8183	8.16E-07	15	16	0.0676	1.14E-07
	x8	22	46	0.1619	7.35E-07	15	16	0.0547	1.15E-07
10000	x1	4	10	0.0435	3.62E-09	8	9	0.0727	3.82E-07
	x2	2	6	0.0229	1.21E-07	8	9	0.1028	3.83E-07
	x3	28	58	0.8568	1.05E-07	11	12	0.0785	1.12E-07
	x4	25	52	1.1158	9.00E-07	16	17	0.1164	7.17E-07
	x5	25	52	0.1869	9.00E-07	16	17	0.1788	7.17E-07
	x6	15	32	0.1591	5.56E-07	11	12	0.0731	4.53E-07
	x7	25	52	0.3560	9.00E-07	16	17	0.1178	7.17E-07
	x8	27	56	1.0754	2.32E-07	16	17	0.0988	7.27E-07
50000	x1	5	12	0.1702	9.31E-09	8	9	0.3421	4.49E-07
	x2	2	6	0.1391	6.32E-08	8	9	0.6900	7.60E-07
	x3	21	44	0.9208	6.18E-10	11	12	0.3741	1.32E-07
	x4	23	48	1.5209	1.82E-07	14	15	0.3856	9.11E-07
	x5	23	48	0.5798	1.82E-07	14	15	0.8571	9.11E-07
	x6	15	32	0.5612	2.59E-07	10	11	0.3134	4.57E-07
	x7	23	48	1.4190	1.82E-07	14	15	0.3989	9.11E-07
	x8	23	48	0.8190	1.99E-07	14	15	0.6855	9.12E-07
100000	x1	6	14	0.3599	1.10E-09	8	9	0.4518	4.95E-07
	x2	2	6	0.1567	5.40E-08	6	7	0.3181	3.56E-07
	x3	29	60	2.2696	6.52E-08	11	12	0.8592	1.35E-07
	x4	20	42	1.6957	3.99E-07	15	16	0.9457	4.77E-07
	x5	20	42	1.5530	3.99E-07	15	16	0.8649	4.77E-07
	x6	15	32	1.0689	2.34E-07	10	11	0.4211	4.56E-07
	x7	20	42	1.6885	3.99E-07	15	16	1.2478	4.77E-07
	x8	20	42	1.7092	3.97E-07	15	16	1.2242	4.77E-07

Table 3. Numerical results of the PDY and MDY algorithms on Problem 3 with given initial points and dimensions.

		PDY				MDY			
DIMENSION	INITIAL POINT	ITER	FVAL	TIME	NORM	ITER	FVAL	TIME	NORM
1000	x1	1	3	0.0285	0.00E+00	1	2	0.0562	0.00E+00
	x2	1	3	0.0032	0.00E+00	1	2	0.0028	0.00E+00
	x3	1	3	0.0042	0.00E+00	1	2	0.0026	0
	x4	1	3	0.0029	0.00E+00	1	2	0.0023	0.00E+00
	x5	1	3	0.0043	0.00E+00	1	2	0.0070	0.00E+00
	x6	1	3	0.0029	0.00E+00	1	2	0.0027	0
	x7	1	3	0.0037	0.00E+00	1	2	0.0134	0.00E+00
	x8	1	3	0.0045	0.00E+00	1	2	0.0023	0.00E+00
5000	x1	1	3	0.0124	0.00E+00	1	2	0.0056	0.00E+00
	x2	1	3	0.0238	0.00E+00	1	2	0.0064	0.00E+00
	x3	1	3	0.0103	0.00E+00	1	2	0.0071	0
	x4	1	3	0.0246	0.00E+00	1	2	0.0053	0.00E+00
	x5	1	3	0.0158	0.00E+00	1	2	0.0053	0.00E+00
	x6	1	3	0.0194	0.00E+00	1	2	0.0085	0
	x7	1	3	0.0054	0.00E+00	1	2	0.0182	0.00E+00
	x8	1	3	0.0188	0.00E+00	1	2	0.0092	0.00E+00
10000	x1	1	3	0.0159	0.00E+00	1	2	0.0111	0.00E+00
	x2	1	3	0.0389	0.00E+00	1	2	0.0103	0.00E+00
	x3	1	3	0.0707	0.00E+00	1	2	0.0086	0
	x4	1	3	0.0514	0.00E+00	1	2	0.0178	0.00E+00
	x5	1	3	0.0095	0.00E+00	1	2	0.0207	0.00E+00
	x6	1	3	0.0700	0.00E+00	1	2	0.0116	0
	x7	1	3	0.1447	0.00E+00	1	2	0.0085	0.00E+00
	x8	1	3	0.0666	0.00E+00	1	2	0.0131	0.00E+00
50000	x1	1	3	0.0354	0.00E+00	1	2	0.0339	0.00E+00
	x2	1	3	0.0306	0.00E+00	1	2	0.0296	0.00E+00
	x3	1	3	0.0452	0.00E+00	1	2	0.0314	0
	x4	1	3	0.0981	0.00E+00	1	2	0.0288	0.00E+00
	x5	1	3	0.2769	0.00E+00	1	2	0.0285	0.00E+00
	x6	1	3	0.0522	0	1	2	0.0567	0
	x7	1	3	0.0429	0.00E+00	1	2	0.0677	0.00E+00
	x8	1	3	0.0304	0.00E+00	1	2	0.0317	0.00E+00
100000	x1	1	3	0.1861	0.00E+00	1	2	0.0528	0.00E+00
	x2	1	3	0.0571	0.00E+00	1	2	0.0665	0.00E+00
	x3	1	3	0.2492	0	1	2	0.0574	0
	x4	1	3	0.1440	0.00E+00	1	2	0.0518	0.00E+00
	x5	1	3	0.2074	0.00E+00	1	2	0.1099	0.00E+00
	x6	1	3	0.1527	0	1	2	0.0913	0
	x7	1	3	0.1820	0.00E+00	1	2	0.0632	0.00E+00
	x8	1	3	0.2676	0.00E+00	1	2	0.0525	0.00E+00

Table 4. Numerical results of the PDY and MDY algorithms on Problem 4 with given initial points and dimensions.

		PDY				**MDY**			
DIMENSION	INITIAL POINT	ITER	FVAL	TIME	NORM	ITER	FVAL	TIME	NORM
1000	x1	1	2	0.0370	0.00E+00	1	2	0.0315	0.00E+00
	x2	1	3	0.0192	0	1	2	0.0041	0
	x3	1	3	0.0051	0	1	2	0.0035	0
	x4	1	3	0.0163	0.00E+00	1	2	0.0028	0.00E+00
	x5	1	3	0.0208	0.00E+00	1	2	0.0056	0.00E+00
	x6	1	3	0.0895	0	1	2	0.0076	0
	x7	1	3	0.0048	0.00E+00	1	2	0.0045	0.00E+00
	x8	1	3	0.0036	0.00E+00	1	2	0.0043	0.00E+00
5000	x1	1	2	0.0186	0.00E+00	1	2	0.0174	0.00E+00
	x2	1	3	0.0506	0	1	2	0.0124	0
	x3	1	3	0.0075	0	1	2	0.0077	0
	x4	1	3	0.0136	0.00E+00	1	2	0.0266	0.00E+00
	x5	1	3	0.0700	0.00E+00	1	2	0.0216	0.00E+00
	x6	1	3	0.0195	0	1	2	0.0153	0
	x7	1	3	0.0145	0.00E+00	1	2	0.0098	0.00E+00
	x8	1	3	0.0121	0.00E+00	1	2	0.0141	0.00E+00
10000	x1	1	2	0.0070	0.00E+00	1	2	0.0119	0.00E+00
	x2	1	3	0.1034	0	1	2	0.0196	0
	x3	1	3	0.0135	0	1	2	0.0158	0
	x4	1	3	0.0361	0.00E+00	1	2	0.0332	0.00E+00
	x5	1	3	0.0544	0.00E+00	1	2	0.0394	0.00E+00
	x6	1	3	0.0638	0	1	2	0.1124	0
	x7	1	3	0.0252	0.00E+00	1	2	0.0225	0.00E+00
	x8	1	3	0.0146	0.00E+00	1	2	0.0371	0.00E+00
50000	x1	1	2	0.0343	0.00E+00	1	2	0.0865	0.00E+00
	x2	1	3	0.1957	0	1	2	0.0819	0
	x3	1	3	0.0459	0	1	2	0.0964	0
	x4	1	3	0.2370	0.00E+00	1	2	0.0841	0.00E+00
	x5	1	3	0.1080	0.00E+00	1	2	0.0739	0.00E+00
	x6	1	3	0.0633	0	1	2	0.0821	0
	x7	1	3	0.2133	0.00E+00	1	2	0.1969	0.00E+00
	x8	1	3	0.1406	0.00E+00	1	2	0.0873	0.00E+00
100000	x1	1	2	0.1164	0.00E+00	1	2	0.1440	0.00E+00
	x2	1	3	0.2909	0	1	2	0.2257	0
	x3	1	3	0.1400	0	1	2	0.0981	0
	x4	1	3	0.3750	0.00E+00	1	2	0.1590	0.00E+00
	x5	1	3	0.4534	0.00E+00	1	2	0.1982	0.00E+00
	x6	1	3	0.2308	0	1	2	0.1476	0
	x7	1	3	0.4785	0.00E+00	1	2	0.1296	0.00E+00
	x8	1	3	0.2340	0.00E+00	1	2	0.2338	0.00E+00

Table 5. Numerical results of the PDY and MDY algorithms on Problem 5 with given initial points and dimensions.

		PDY				MDY			
DIMENSION	INITIAL POINT	ITER	FVAL	TIME	NORM	ITER	FVAL	TIME	NORM
1000	x1	22	46	0.0222	9.21E-07	9	10	0.0444	1.48E-07
	x2	21	44	0.0430	6.82E-07	2	3	0.0172	0.00E+00
	x3	19	39	0.0233	7.03E-07	9	10	0.0142	6.20E-07
	x4	23	48	0.0597	5.48E-07	11	12	0.0127	4.08E-07
	x5	23	48	0.0573	5.48E-07	11	12	0.0116	4.08E-07
	x6	19	40	0.0357	8.45E-07	8	9	0.0112	1.64E-07
	x7	23	48	0.0509	5.48E-07	11	12	0.0105	4.08E-07
	x8	23	48	0.0198	5.48E-07	11	12	0.0121	4.73E-07
5000	x1	24	50	0.3187	5.15E-07	3	4	0.0191	0.00E+00
	x2	22	46	0.0826	7.62E-07	8	9	0.0300	7.90E-07
	x3	19	39	0.0502	7.03E-07	9	10	0.0263	6.20E-07
	x4	24	50	0.6055	6.13E-07	12	13	0.0365	9.32E-08
	x5	24	50	0.0659	6.13E-07	12	13	0.0433	9.32E-08
	x6	19	40	0.0562	8.45E-07	8	9	0.0263	1.79E-07
	x7	24	50	0.2314	6.13E-07	12	13	0.0359	9.32E-08
	x8	24	50	0.2175	6.13E-07	12	13	0.0715	9.55E-08
10000	x1	24	50	0.9088	7.28E-07	3	4	0.0217	0.00E+00
	x2	23	48	1.0124	5.39E-07	8	9	0.1157	7.49E-07
	x3	19	39	0.0770	7.03E-07	9	10	0.0461	6.20E-07
	x4	24	50	0.1673	8.66E-07	12	13	0.0581	1.97E-07
	x5	24	50	0.3174	8.66E-07	12	13	0.0515	1.97E-07
	x6	19	40	0.1640	8.45E-07	8	9	0.1526	1.81E-07
	x7	24	50	0.4775	8.66E-07	12	13	0.0701	1.97E-07
	x8	24	50	0.1529	8.66E-07	12	13	0.0624	2.00E-07
50000	x1	1000	2001	57.2051	-	3	4	0.0639	0.00E+00
	x2	24	50	0.6099	6.03E-07	3	4	0.0577	0.00E+00
	x3	19	39	0.3208	7.03E-07	9	10	0.1544	6.20E-07
	x4	26	54	1.4338	7.40E-07	13	14	0.2169	1.20E-07
	x5	26	54	0.5082	7.40E-07	13	14	0.2429	1.20E-07
	x6	19	40	0.4547	8.45E-07	8	9	0.2335	1.83E-07
	x7	26	54	1.4782	7.40E-07	13	14	0.2546	1.20E-07
	x8	26	54	0.4505	7.40E-07	13	14	0.2212	1.21E-07
100000	x1	1000	2001	111.2323	-	3	4	0.1011	0.00E+00
	x2	24	50	0.7284	0.00E+00	8	9	0.2502	7.91E-07
	x3	19	39	0.9047	7.03E-07	9	10	0.5091	6.20E-07
	x4	27	56	0.8660	5.23E-07	13	14	0.5304	1.63E-07
	x5	27	56	1.4732	5.23E-07	13	14	0.5062	1.63E-07
	x6	19	40	1.0220	8.45E-07	8	9	0.2425	1.83E-07
	x7	27	56	1.6349	5.23E-07	13	14	0.4260	1.63E-07
	x8	27	56	1.3935	5.23E-07	13	14	0.5942	1.63E-07

Table 6. Numerical results of the PDY and MDY algorithms on Problem 6 with given initial points and dimensions.

		PDY				**MDY**			
DIMENSION	INITIAL POINT	ITER	FVAL	TIME	NORM	ITER	FVAL	TIME	NORM
1000	x1	44	90	0.0658	1.18E-11	19	20	0.0445	5.53E-07
	x2	29	60	0.0381	7.81E-09	17	18	0.0228	6.66E-08
	x3	26	54	0.0379	6.89E-07	16	17	0.0188	2.58E-07
	x4	31	64	0.0327	6.25E-07	16	17	0.0149	3.89E-08
	x5	33	68	0.0308	5.56E-08	29	30	0.0539	3.99E-07
	x6	27	56	0.0306	7.55E-07	15	16	0.0155	2.42E-07
	x7	31	64	0.0422	6.25E-07	16	17	0.0183	3.89E-08
	x8	27	56	0.0372	8.60E-07	28	29	0.0614	8.56E-07
5000	x1	31	64	0.1616	5.94E-07	21	22	0.1470	9.91E-08
	x2	34	70	0.4522	8.16E-07	23	24	0.1008	2.64E-07
	x3	27	56	0.7265	8.45E-08	23	24	0.0947	8.84E-07
	x4	27	56	0.1009	5.59E-07	18	19	0.0651	1.23E-07
	x5	34	70	0.4300	3.51E-08	37	38	0.1812	5.97E-08
	x6	41	84	0.1106	7.52E-07	23	24	0.0756	1.19E-07
	x7	27	56	0.1036	5.59E-07	18	19	0.0748	1.23E-07
	x8	34	70	0.3258	2.98E-08	37	38	0.3811	5.11E-08
10000	x1	31	64	1.0425	1.08E-07	20	21	0.1020	7.15E-08
	x2	28	58	0.1353	6.94E-07	21	22	0.1216	8.05E-07
	x3	28	58	0.6867	6.64E-07	22	23	0.1175	4.11E-08
	x4	27	56	0.1326	6.74E-07	28	29	0.2012	5.77E-07
	x5	39	80	0.1576	9.58E-07	47	48	0.4850	2.16E-07
	x6	25	52	0.5139	3.29E-07	22	23	0.1242	7.33E-07
	x7	27	56	1.3498	6.74E-07	28	29	0.4814	5.77E-07
	x8	39	80	0.2082	9.54E-07	52	53	0.5935	1.68E-07
50000	x1	38	78	1.2228	3.81E-07	21	22	0.5432	3.12E-07
	x2	32	66	2.0428	9.69E-07	25	26	0.5454	3.09E-07
	x3	33	68	1.2210	7.92E-07	20	21	0.4772	8.52E-08
	x4	29	60	0.6295	7.67E-07	33	34	1.7709	1.82E-08
	x5	43	88	1.2302	5.80E-07	54	55	2.5580	3.97E-08
	x6	33	68	0.6304	5.82E-07	23	24	0.5140	4.08E-07
	x7	29	60	1.6810	7.67E-07	33	34	1.0103	1.82E-08
	x8	43	88	0.7733	5.80E-07	54	55	3.3048	4.32E-08
100000	x1	43	88	1.8763	6.08E-07	21	22	0.8796	5.05E-07
	x2	34	70	1.9474	6.98E-07	20	21	0.6977	7.14E-07
	x3	34	70	1.9860	8.90E-07	19	20	0.7883	1.35E-07
	x4	33	68	1.5214	5.29E-07	32	33	2.9599	3.60E-07
	x5	45	92	2.1469	5.72E-07	64	65	6.0691	4.87E-07
	x6	35	72	1.6323	7.19E-07	22	23	0.9286	5.29E-07
	x7	33	68	1.7901	5.29E-07	32	33	2.0214	3.60E-07
	x8	45	92	2.9727	5.72E-07	63	64	6.9672	8.49E-07

Table 7. Numerical results of the **PDY** and **MDY** algorithms on Problem 7 with given initial points and dimensions.

		PDY				**MDY**			
DIMENSION	INITIAL POINT	ITER	FVAL	TIME	NORM	ITER	FVAL	TIME	NORM
1000	x1	25	52	0.0393	8.08E-07	9	10	0.0634	7.69E-07
	x2	26	54	0.1552	6.16E-07	9	10	0.0105	2.29E-07
	x3	26	54	0.5694	6.39E-07	10	11	0.0251	5.26E-07
	x4	26	54	0.1853	5.26E-07	10	11	0.0182	2.35E-07
	x5	26	54	0.0600	5.26E-07	10	11	0.0177	2.35E-07
	x6	26	54	0.0420	6.38E-07	10	11	0.0207	7.79E-08
	x7	26	54	0.0874	5.26E-07	10	11	0.0162	2.35E-07
	x8	26	54	0.0620	5.26E-07	10	11	0.0250	2.34E-07
5000	x1	26	54	0.1716	9.05E-07	9	10	0.0751	6.03E-07
	x2	27	56	0.2699	6.90E-07	10	11	0.0637	2.10E-07
	x3	27	56	0.3134	7.16E-07	10	11	0.0550	2.40E-07
	x4	27	56	0.1879	5.89E-07	11	12	0.0595	7.07E-08
	x5	27	56	0.8405	5.89E-07	11	12	0.0716	7.07E-08
	x6	27	56	0.1697	7.16E-07	10	11	0.0543	2.77E-07
	x7	27	56	0.1786	5.89E-07	11	12	0.1336	7.07E-08
	x8	27	56	0.3122	5.89E-07	11	12	0.0701	7.07E-08
10000	x1	27	56	0.9604	6.40E-07	10	11	0.1228	1.71E-07
	x2	28	58	0.2988	7.32E-07	10	11	0.1194	3.88E-07
	x3	29	60	0.3063	5.70E-07	10	11	0.0914	3.91E-07
	x4	28	58	0.3465	6.25E-07	11	12	0.1108	5.35E-07
	x5	28	58	0.8089	6.25E-07	11	12	0.2015	5.35E-07
	x6	29	60	1.0545	5.69E-07	10	11	0.1050	3.93E-07
	x7	28	58	1.5376	6.25E-07	11	12	0.1228	5.35E-07
	x8	28	58	0.5480	6.25E-07	11	12	0.1011	5.35E-07
50000	x1	29	60	1.6703	0.00E+00	7	8	0.2341	8.40E-07
	x2	32	66	1.9397	0.00E+00	10	11	0.3769	3.12E-07
	x3	33	68	3.8015	0.00E+00	10	11	0.9105	3.07E-07
	x4	31	64	2.8722	0.00E+00	10	11	0.4259	7.09E-07
	x5	31	64	1.5377	0.00E+00	10	11	0.3419	7.09E-07
	x6	33	68	1.7936	0.00E+00	10	11	0.3309	3.07E-07
	x7	31	64	1.5968	0.00E+00	10	11	0.3624	7.09E-07
	x8	31	64	1.4277	0.00E+00	10	11	0.8288	7.09E-07
100000	x1	31	64	3.0553	0.00E+00	10	11	0.9843	3.12E-07
	x2	34	70	3.2251	0.00E+00	10	11	0.7217	7.84E-08
	x3	35	72	3.3118	0.00E+00	9	10	0.7526	5.66E-07
	x4	33	68	3.1050	0.00E+00	10	11	1.2116	2.42E-07
	x5	33	68	3.0781	0.00E+00	10	11	0.8509	2.42E-07
	x6	35	72	3.4062	0.00E+00	9	10	0.7296	5.67E-07
	x7	33	68	3.1780	0.00E+00	10	11	0.7927	2.42E-07
	x8	33	68	3.0914	0.00E+00	10	11	0.7982	2.42E-07

Table 8. Numerical results of the PDY and MDY algorithms on Problem 8 with given initial points and dimensions.

		PDY				MDY			
DIMENSION	INITIAL POINT	ITER	FVAL	TIME	NORM	ITER	FVAL	TIME	NORM
1000	x1	187	376	0.4150	9.74E-07	22	23	0.0312	4.76E-07
	x2	225	452	0.2766	9.68E-07	23	24	0.0282	2.75E-07
	x3	205	412	0.2474	9.84E-07	23	24	0.0273	3.97E-07
	x4	204	410	0.3133	9.77E-07	24	25	0.0712	8.64E-07
	x5	204	410	0.4370	9.77E-07	24	25	0.0553	8.64E-07
	x6	226	454	0.8938	9.59E-07	26	27	0.0550	9.85E-07
	x7	204	410	0.4095	9.77E-07	24	25	0.0283	8.64E-07
	x8	204	410	0.3274	9.77E-07	24	25	0.0349	8.65E-07
5000	x1	191	384	1.3117	9.76E-07	23	24	0.3638	4.85E-07
	x2	208	418	2.3451	9.98E-07	21	22	0.1202	5.49E-07
	x3	208	418	1.3535	9.66E-07	22	23	0.1095	8.12E-07
	x4	205	412	1.4452	9.48E-07	23	24	0.1518	4.57E-07
	x5	205	412	1.3220	9.48E-07	23	24	0.1192	4.57E-07
	x6	202	406	1.1897	9.74E-07	24	25	0.1190	5.90E-07
	x7	205	412	1.4644	9.48E-07	23	24	0.1242	4.57E-07
	x8	205	412	0.7725	9.48E-07	23	24	0.1392	4.57E-07
10000	x1	180	362	1.6384	9.95E-07	24	25	0.4626	5.22E-07
	x2	219	440	3.2108	9.71E-07	21	22	0.2472	7.44E-07
	x3	200	402	1.9769	9.98E-07	24	25	0.3573	2.77E-07
	x4	197	396	1.9286	9.86E-07	24	25	0.2675	4.32E-07
	x5	197	396	1.7271	9.86E-07	24	25	0.2505	4.32E-07
	x6	222	446	1.9256	9.60E-07	24	25	0.3133	4.85E-07
	x7	197	396	1.7646	9.86E-07	24	25	0.7206	4.32E-07
	x8	197	396	1.9013	9.86E-07	24	25	0.3670	4.33E-07
50000	x1	178	358	5.6703	9.89E-07	23	24	1.0011	2.43E-07
	x2	204	410	6.3962	9.76E-07	23	24	0.9521	5.05E-07
	x3	201	404	6.5874	9.87E-07	29	30	1.8089	2.98E-07
	x4	196	394	6.1046	9.62E-07	24	25	2.0486	4.07E-07
	x5	196	394	5.8055	9.62E-07	24	25	1.1312	4.07E-07
	x6	198	398	5.9883	9.96E-07	23	24	1.2455	9.10E-07
	x7	196	394	5.8655	9.62E-07	24	25	1.8848	4.07E-07
	x8	196	394	6.2188	9.62E-07	24	25	1.0326	4.07E-07
100000	x1	181	364	11.8715	9.65E-07	22	23	2.0797	8.40E-07
	x2	210	422	13.7964	9.90E-07	192	193	18.2612	9.34E-07
	x3	215	432	14.2830	1.00E-06	151	152	13.9538	7.09E-07
	x4	212	426	13.6333	9.92E-07	22	23	2.1214	7.18E-07
	x5	212	426	13.8869	9.92E-07	22	23	2.1095	7.18E-07
	x6	211	424	13.8529	9.60E-07	24	25	2.3077	8.51E-07
	x7	212	426	13.8288	9.92E-07	22	23	2.0365	7.18E-07
	x8	212	426	13.5868	9.92E-07	22	23	2.0444	7.17E-07

Table 9. Numerical results of the PDY and MDY algorithms on Problem 9 with given initial points and dimensions.

		PDY				MDY			
DIMENSION	INITIAL POINT	ITER	FVAL	TIME	NORM	ITER	FVAL	TIME	NORM
1000	x1	16	34	0.0253	7.23E-07	1	2	0.0417	0.00E+00
	x2	16	34	0.0268	8.17E-07	2	3	0.0086	0.00E+00
	x3	15	31	0.0695	7.97E-07	5	6	0.0130	3.41E-09
	x4	19	40	0.0565	4.77E-07	12	13	0.0170	0.00E+00
	x5	19	40	0.0995	4.77E-07	12	13	0.0214	0.00E+00
	x6	17	36	0.0403	4.98E-07	11	12	0.0149	0.00E+00
	x7	19	40	0.0750	4.77E-07	12	13	0.0322	0.00E+00
	x8	19	40	0.0627	4.83E-07	12	13	0.0355	0.00E+00
5000	x1	17	36	0.9394	6.06E-07	1	2	0.0172	0.00E+00
	x2	17	36	0.3665	6.85E-07	3	4	0.0420	0.00E+00
	x3	15	31	0.1176	7.97E-07	5	6	0.0259	3.41E-09
	x4	20	42	0.2467	4.02E-07	12	13	0.0782	0.00E+00
	x5	20	42	0.1706	4.02E-07	12	13	0.4931	0.00E+00
	x6	17	36	0.2692	4.99E-07	4	5	0.0382	0.00E+00
	x7	20	42	0.3417	4.02E-07	12	13	0.0810	0.00E+00
	x8	20	42	0.2955	4.03E-07	12	13	0.0718	0.00E+00
10000	x1	17	36	0.7680	8.57E-07	1	2	0.0211	0.00E+00
	x2	17	36	0.1949	9.69E-07	3	4	0.0487	0.00E+00
	x3	15	31	0.8021	7.97E-07	5	6	0.0414	3.41E-09
	x4	20	42	0.2540	5.69E-07	12	13	0.2491	0.00E+00
	x5	20	42	0.7110	5.69E-07	12	13	0.3815	0.00E+00
	x6	17	36	0.3225	4.99E-07	4	5	0.0871	0.00E+00
	x7	20	42	0.7992	5.69E-07	12	13	0.1418	0.00E+00
	x8	20	42	1.2002	5.70E-07	12	13	0.1433	0.00E+00
50000	x1	21	44	1.2440	9.52E-07	1	2	0.0817	0.00E+00
	x2	18	38	1.3985	8.13E-07	4	5	0.1564	0.00E+00
	x3	15	31	0.4610	7.97E-07	5	6	0.1678	3.41E-09
	x4	20	42	1.9300	7.76E-07	12	13	0.9732	0.00E+00
	x5	20	42	0.8548	7.76E-07	12	13	0.6015	0.00E+00
	x6	17	36	1.0072	4.99E-07	4	5	0.1133	0.00E+00
	x7	20	42	1.2552	7.76E-07	12	13	0.5539	0.00E+00
	x8	20	42	1.1311	7.76E-07	12	13	0.9251	0.00E+00
100000	x1	23	48	2.2913	3.79E-07	1	2	0.2783	0.00E+00
	x2	19	40	1.6232	4.31E-07	3	4	0.3007	0.00E+00
	x3	15	31	1.1872	7.97E-07	5	6	0.3306	3.41E-09
	x4	21	44	2.1965	4.11E-07	11	12	0.9774	0.00E+00
	x5	21	44	2.0734	4.11E-07	11	12	1.6032	0.00E+00
	x6	17	36	1.4438	4.99E-07	4	5	0.2584	0.00E+00
	x7	21	44	2.0290	4.11E-07	11	12	0.8957	0.00E+00
	x8	21	44	1.9219	4.11E-07	11	12	0.9805	0.00E+00

In addition, to further visualize the comparison of the MDY algorithm with the PDY algorithm graphically, we adopt the well- known Dolan and Morè performance profile [39] as reported in Figures

1–3. From Figures 1 and 2, it can be seen that the MDY algorithm outperformed the PDY algorithm significantly. The MDY algorithm solves about 93% of the problems considered with least ITER and 99% of the problems with least FVAL as opposed to the PDY method with about 28% and 3% problems for the ITER and FVAL respectively. Moreover, in terms of TIME, Figure 3 indicated that the MDY algorithm still performs much better than the PDY algorithm by solving around 88% of the problems in lesser time. From these figures, we can conclude that the numerical performance of the MDY algorithm has great advantage when compared with the existing PDY algorithm.

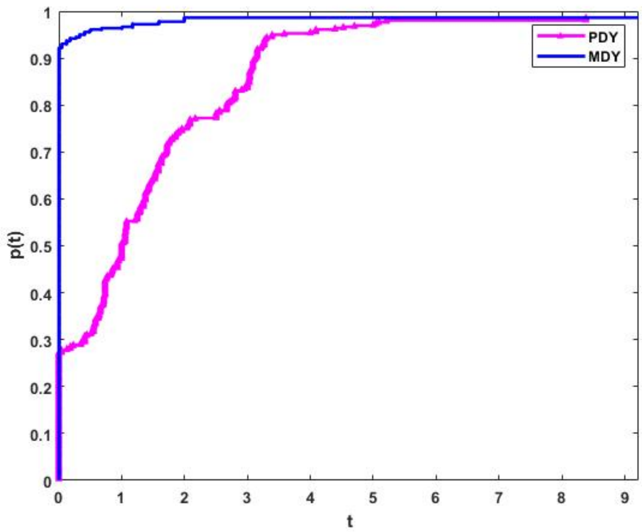

Figure 1. Performance profile on number of iterations.

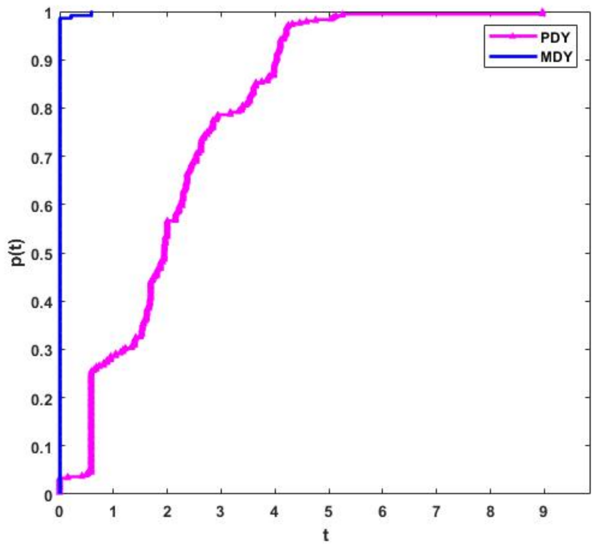

Figure 2. Performance profile on function evaluations.

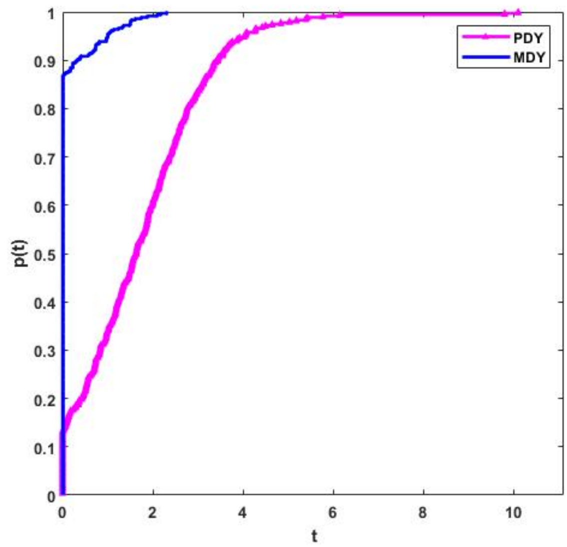

Figure 3. Performance profile on CPU time.

4. Application in compressive sensing

The problem of sparse signal reconstruction has attracted the attention of many researchers in the field of signal processing, machine learning and computer vision. This problem involves solving minimization of an objective function containing quadratic ℓ_2 error term and a sparse ℓ_1 regularization term as follows

$$\min_{x} \frac{1}{2}\|h - Ax\|_2^2 + \rho\|x\|_1, \tag{4.1}$$

where $x \in R^n$, $h \in R^m$, $A \in R^{m\times n}$ $(m << n)$ is a linear operator, $\rho \geq 0$, $\|x\|_2$ is the Euclidean norm of x and $\|x\|_1 = \sum_{i=1}^n |x_i|$ is the ℓ_1-norm of x.

A lot of methods have been developed for solving (4.1) some of which can be found in [40–45]. Figueiredo et al. [43] consider reformulating (4.1) into a quadratic problem by expressing $x \in \mathbb{R}^n$ into two parts as

$$x = t - y, \qquad t \geq 0, \quad y \geq 0,$$

where $t_i = (x_i)_+$, $y_i = (-x_i)_+$ for all $i = 1, 2, ..., n$, and $(.)_+ = \max\{0, .\}$. Also, we have $\|x\|_1 = e_n^T t + e_n^T y$, where $e_n = (1, 1, ..., 1)^T \in R^n$. From this reformulation, we can write (4.1) as

$$\min_{t,y} \frac{1}{2}\|h - A(t-y)\|_2^2 + \rho e_n^T t + \rho e_n^T y, \qquad t \geq 0, \quad y \geq 0, \tag{4.2}$$

from [43], Eq (4.2) can be written as

$$\min_{z} \frac{1}{2} z^T E z + c^T z, \qquad \text{such that} \quad z \geq 0, \tag{4.3}$$

where $z = \begin{pmatrix} t \\ y \end{pmatrix}$, $\quad c = \omega e_{2n} + \begin{pmatrix} -a \\ a \end{pmatrix}$, $\quad a = A^T h$, $\quad E = \begin{pmatrix} A^T A & -A^T A \\ -A^T A & A^T A \end{pmatrix}$.

It is not difficult to see that E is a positive semi-definite showing that problem (4.3) is quadratic programming problem.

Xiao et al [17] further translated (4.3) into a linear variable problem, equivalently, a linear complementary problem and the variable z solves the linear complementary problem provided that it solves the nonlinear equation:

$$A(z) = \min\{z, Ez + c\} = 0, \tag{4.4}$$

where A is a vector-valued function. In [8, 46], it is proved that the function $A(z)$ is continuous and monotone. Thus, problem (4.1) is equivalent to problem (1.1). Therefore, the algorithm we proposed in this work to solve (1.1) can efficiently solve (4.1) as well.

As an application, we consider applying our proposed algorithm in reconstructing a sparse signal of length n from k observations using mean squared error (MSE) as a metric for assessing quality reconstruction. The (MSE) is defined as

$$MSE = \frac{1}{n}\|s - \tilde{s}\|^2,$$

where s represents the original signal and $\tilde{s}$ the restored signal. We choose $n = 2^{12}$, $k = 2^{10}$ to be the size of the signal and the original signal contains 2^7 randomly nonzero elements. The measurement y contains noise, $y = As + \omega$, where ω is the Gaussian noise distributed as $N(0, 10^{-4})$ and A is the Gaussian matrix generated by command $randn(m, n)$, in Matlab.

We compared the performance of our proposed algorithm (MDY) with SGCS proposed in [8]. The parameters in SGCS are maintained as they are in [8], while in MDY we choose $r = 0.001$, $\theta = 1/(k+1)^2$, $\mu = 1.1$, $\gamma = 0.1$, $\sigma = 0.01$, $\Lambda = 1$, and $\beta = 0.65$. Each code is run with same initial point and continuation technique on parameter μ. We only focused on the convergence behaviour of each method to obtain a solution with similar accuracy. We initialized the experiments by $x_0 = A^T y$ and terminated the iteration when the relative change in the objective function satisfies

$$\left|\frac{f(x_k) - f(x_{k-1})}{f(x_{k-1})}\right| < 10^{-5}.$$

The performance of both MDY and SGCS are shown in Figures 4 and 5. Figure 4 shows that both the MDY and the SGCS methods recover the signal. However, looking at the reported metrics of each method, it can be observed that the MDY method is more efficient since it has a lesser MSE, and its recovery has fewer number of iterations and CPU time. To show the performance of both methods graphically, we plotted four graphs (see Figure 5) demonstrating the convergence behaviour of the MDY method and SGCS method based on the MSE, objective function, number of iterations and CPU time. From Figure 5, it can be observed that the proposed MDY method has faster convergence rate compared to the SGCS method. This shows that the MDY method can be a good alternative solver for signal recovery problems.

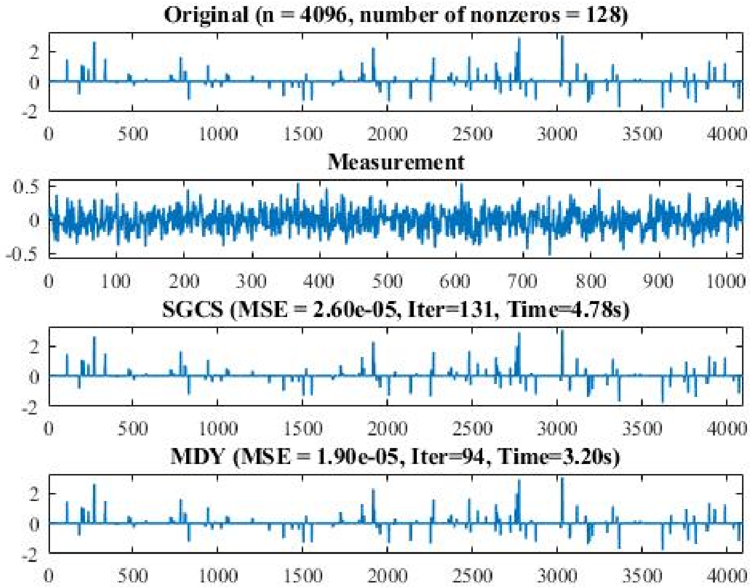

Figure 4. From top to bottom: the original signal, the measurement, and the recovery signals by SGCS and MDY methods.

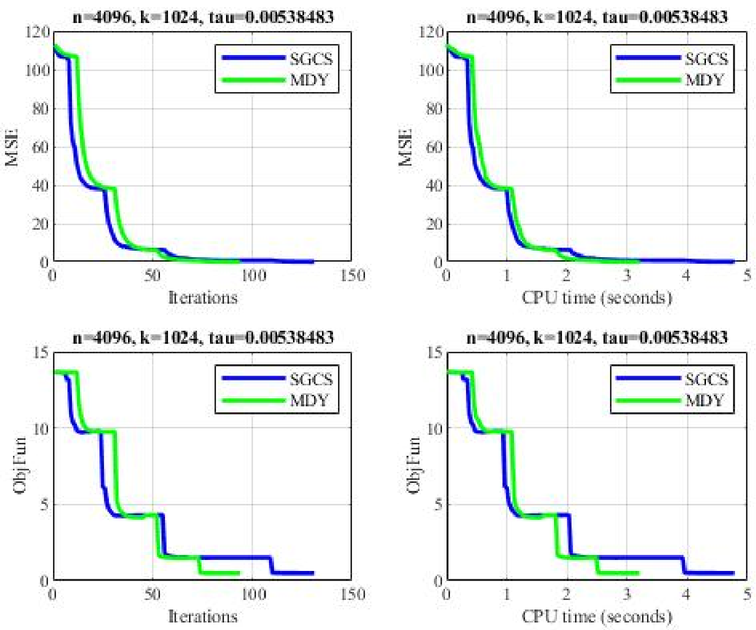

Figure 5. Comparison of SGCS and MDY methods based on MSE, number of iterations, objective function and CPU time.

5. Conclusions

In the work, a spectral conjugate gradient algorithm is proposed. The search direction uses a convex combination of the well known DY conjugate gradient parameter and a modified conjugate descent parameter. The search direction is suffiently descent, and global convergence of the proposed algorithm is proved under some assumptions. Numerical experiments are reported to show the efficiency of the algorithm in comparison with the PDY algorithm proposed in [22]. In addition, an application of the proposed algorithm is shown in signal recovery and the result is compared with SGCS algorithm proposed in [8]. Based on the results obtained, it can be observed that the proposed algorithm has a better performance than the PDY and SGCS algorithms in numerical and signal recovery experiments respectively. Future work include applying the new proposed algorithm to solve 2D robotic motion control as presented in [47].

Acknowledgments

The authors acknowledge the financial support provided by the Center of Excellence in Theoretical and Computational Science (TaCS-CoE), KMUTT and Center under Computational and Applied Science for Smart Innovation research Cluster (CLASSIC), Faculty of Science, KMUTT. This research was funded by Thailand Science Research and Innovation Fund, and King Mongkut's University of Technology North Bangkok with Contract no. KMUTNB-BasicR-64-22. The first author was supported by the Petchra Pra Jom Klao Ph.D. research Scholarship from King Mongkut's University of Technology Thonburi (KMUTT) Thailand (Grant No. 19/2562). Also, Aliyu Muhammed Awwal would like to thank the Postdoctoral Fellowship from King Mongkut's University of Technology Thonburi (KMUTT), Thailand. Moreover, this research project is supported by Thailand Science Research and Innovation (TSRI) Basic Research Fund: Fiscal year 2021 under project number 64A306000005.

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. M. Sun, J. Liu, Y. Wang, Two improved conjugate gradient methods with application in compressive sensing and motion control, *Math. Probl. Eng.*, **2020**, (2020), 9175496.
2. S. P. Dirkse, M. J. Ferris, A collection of nonlinear mixed complementarity problems, *Optim. Method. Softw.*, **5** (1995), 319–345.
3. A. J. Wood, B. F. Wollenberg, G. B. Sheblé, *Power generation, operation, and control*, John Wiley & Sons, 2013.
4. K. Meintjes, A. P. Morgan, A methodology for solving chemical equilibrium systems, *Appl. Math. Comput.*, **22** (1987), 333–361.
5. N. A. Iusem, V. M. Solodov, Newton-type methods with generalized distances for constrained optimization, *Optimization*, **41** (1997), 257–278.

6. M. Fukushima, Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems, *Math. program.*, **53** (1992), 99–110.
7. Y. B. Zhao, D, Li, Monotonicity of fixed point and normal mappings associated with variational inequality and its application, *SIAM J Optimization*, **11** (2001), 962–973.
8. Y. H. Xiao, Q. Y. Wang, Q. J. Hu, Non-smooth equations based method for ℓ_1-norm problems with applications to compressed sensing, *Nonlinear Anal.: Theory*, **74** (2011), 3570–3577.
9. J. E. Dennis, J. J. Moré, A characterization of superlinear convergence and its application to quasi-newton methods, *Math. Comput.*, **28** (1974), 549–560.
10. D. H. Li, M. Fukushima, A globally and superlinearly convergent gauss–newton-based bfgs method for symmetric nonlinear equations, *SIAM J. Numer, Anal.*, **37** (1998), 152–172.
11. A. M. Awwal, P. Kumam, A. B. Abubakar, Spectral modified Polak–Ribiére–Polyak projection conjugate gradient method for solving monotone systems of nonlinear equations, *Appl. Math. Comput.*, **362** (2019), 124514.
12. G. L. Zhou, K. C. Toh, Superlinear convergence of a newton-type algorithm for monotone equations, *J. Optimiz. Theory App.*, **125** (2005), 205–221.
13. W. J. Zhou, D. H. Li, A globally convergent bfgs method for nonlinear monotone equations without any merit functions, *Math. Comput.*, **77** (2008), 2231–2240.
14. M. V. Solodov, B. F. Svaiter, A globally convergent inexact newton method for systems of monotone equations, In: *Reformulation: Nonsmooth, piecewise smooth, semismooth and smoothing methods*, 1998, 355–369.
15. C. W. Wang, Y. J. Wang, C. L. Xu, A projection method for a system of nonlinear monotone equations with convex constraints, *Math. Meth. Oper. Res.*, **66** (2007), 33–46.
16. W. Y. Cheng, A PRP type method for systems of monotone equations, *Math. Comput. Model.*, **50** (2009), 15–20.
17. Y. H. Xiao, H. Zhu, A conjugate gradient method to solve convex constrained monotone equations with applications in compressive sensing, *J. Math. Anal. Appl.*, **405** (2013), 310–319.
18. J. K. Liu, S. J. Li, A projection method for convex constrained monotone nonlinear equations with applications, *Comput. Math. Appl.*, **70** (2015), 2442–2453.
19. Y. H. Dai, Y. X. Yuan. A nonlinear conjugate gradient method with a strong global convergence property, *SIAM J. Optimiz.*, **10** (1999), 177–182.
20. J. K. Liu, S. J. Li, Spectral DY-type projection method for nonlinear monotone systems of equations, *J. Comput. Math*, **33** (2015), 341–355.
21. J. K. Liu, S. J. Li, Multivariate spectral DY-type projection method for convex constrained nonlinear monotone equations, *J. Ind. Manag. Optim.*, **13** (2017), 283–295.
22. J. K. Liu, Y. M. Feng. A derivative-free iterative method for nonlinear monotone equations with convex constraints, *Numer Algorithms*, **82** (2019), 245–262.
23. A. M. Awwal, P. Kumam, L. Wang, S. Huang, W. Kumam, Inertial-based derivative-free method for system of monotone nonlinear equations and application, *IEEE Access*, **8** (2020), 226921–226930.

24. J. Guo, Z. Wan, A modified spectral PRP conjugate gradient projection method for solving large-scale monotone equations and its application in compressed sensing, *Math. Probl. Eng.*, **2019** (2019), 5261830.

25. A. M. Awwal, P. Kumam, H. Mohammad, W. Watthayu, A. B. Abubakar, A Perry-type derivative-free algorithm for solving nonlinear system of equations and minimizing ℓ_1 regularized problem, *Optimization*, **70** (2021), 1231–1259.

26. A. M. Awwal, L. Wang, P. Kumam, H. Mohammad, W. Watthayu, A projection Hestenes–Stiefel method with spectral parameter for nonlinear monotone equations and signal processing, *Math. Comput. Appl.*, **25** (2020), 27.

27. A. B. Abubakar, P. Kumam, H. Mohammad, A. M. Awwal, A Barzilai-Borwein gradient projection method for sparse signal and blurred image restoration, *J. Franklin I.*, **357** (2020), 7266–7285.

28. A. B. Abubakar, P. Kumam, H. Mohammad, A note on the spectral gradient projection method for nonlinear monotone equations with applications, *Comp. Appl. Math.*, **39** (2020), 129.

29. S. Aji, P. Kumam, P. Siricharoen, A. B. Abubakar, M. M. Yahaya, A modified conjugate descent projection method for monotone nonlinear equations and image restoration, *IEEE Access*, **8** (2020), 158656–158665.

30. S. Aji, P. Kumam, A. M. Awwal, M. M. Yahaya, W. Kumam. Two hybrid spectral methods with inertial effect for solving system of nonlinear monotone equations with application in robotics. *IEEE Access*, **9** (2021), 30918–30928.

31. A. H. Ibrahim, P. Kumam, A. B. Abubakar, U. B. Yusuf, S. E. Yimer, K. O. Aremu, An efficient gradient-free projection algorithm for constrained nonlinear equations and image restoration, *AIMS Mathematics*, **6** (2020), 235–260.

32. A. M. Awwal, L. Wang, P. Kumam, H. Mohammad, A two-step spectral gradient projection method for system of nonlinear monotone equations and image deblurring problems, *Symmetry*, **12** (2020), 874.

33. Z. S. Yu, J. Lin, J. Sun, Y. H. Xiao, L. Y. Liu, Z. H. Li, Spectral gradient projection method for monotone nonlinear equations with convex constraints, *Appl. Numer. Math.*, **59** (2009), 2416–2423.

34. A. M. Awwal, P. Kumam, A. B. Abubakar, A modified conjugate gradient method for monotone nonlinear equations with convex constraints, *Appl. Numer. Math.*, **145** (2019), 507–520.

35. W. J. Zhou, D. H. Li. A globally convergent BFGS method for nonlinear monotone equations without any merit functions, *Math. Comput.*, **77** (2008), 2231–2240.

36. W. La Cruz, A spectral algorithm for large-scale systems of nonlinear monotone equations, *Numer. Algor.*, **76** (2017), 1109–1130.

37. W. La Cruz, J. M. Martínez, M. Raydan, Spectral residual method without gradient information for solving large-scale nonlinear systems of equations, *Math. Comput.*, **75** (2006), 1429–1448.

38. Y. Bing, G. Lin, An efficient implementation of merrills method for sparse or partially separable systems of nonlinear equations, *SIAM J. Optim.*, **1** (1991), 206–221.

39. E. D. Dolan, J. J. Moré, Benchmarking optimization software with performance profiles. *Math. Program.*, **91** (2002), 201–213.

40. M. A. T. Figueiredo, R. D. Nowak, An EM algorithm for wavelet-based image restoration, *IEEE T. Image Processing*, **12** (2003), 906–916.
41. E. T. Hale, W. T. Yin, Y. Zhang, A fixed-point continuation method for ℓ_1-regularized minimization with applications to compressed sensing, *CAAM Technical Report TR07-07, Rice University*, **43** (2007), 44.
42. A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, *SIAM J. Imaging Sci.*, **2** (2009), 183–202.
43. M. A. T. Figueiredo, R. D. Nowak, S. J. Wright, Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems, *IEEE J.-STSP*, **1** (2007), 586–597.
44. E. Van Den Berg, M. P. Friedlander, Probing the pareto frontier for basis pursuit solutions, *SIAM J. Sci. Comput.*, **31** (2009), 890–912.
45. E. G. Birgin, J. M. Martínez, M. Raydan, Nonmonotone spectral projected gradient methods on convex sets, *SIAM J. Optimiz.*, **10** (2000), 1196–1211.
46. J. S. Pang, Inexact Newton methods for the nonlinear complementarity problem, *Math, Program*, **36** (1986), 54–71.
47. M. M. Yahaya, P. Kumam, A. M. Awwal, S. Aji, A structured quasi–Newton algorithm with nonmonotone search strategy for structured NLS problems and its application in robotic motion control, *J. Comput. Appl. Math.*, **395** (2021), 113582.