BANGMOD INTERNATIONAL JOURNAL OF MATHEMATICAL & COMPUTATIONAL SCIENCE
ISSN: 2408-154X (Print)
ISSN: 3057-0557 (Online)
https://bangmodjmcs.com
Volume 12 (2026)
Pages 58–80



Spectral Algorithms with Inertial-strategy for Constrained System of Nonlinear Equations

Sani Aji[1*], **Hisham M. Khudur**[2], **Muhammad Dawaki**[3], **Semiu Oladipupo Oladejo**[1], **Karan Arora**[4]

[1] *Department of Mathematics, Faculty of Science, Gombe State University (GSU), Gombe, 760214, Nigeria*
E-mails: ajysani@yahoo.com (S. Aji), abdsemiu@yahoo.com
[2] *College of Computer Science and Mathematics, University of Mosul, Iraq*
E-mails: hisham892020@uomosul.edu.iq (H.M. Khudur)
[3] *Department of Computer Science, Faculty of Science, Gombe State University (GSU), Gombe, 760214, Nigeria*
E-mails: dawakimuhammad@gmail.com (M. Dawaki)
[4] *Indian Institute of Technology Jodhpur Karwar, Jodhpur - 342037 Rajasthan, India*
E-mails: karan36k@gmail.com (K. Arora)

**Corresponding author.*

Received: 14 April 2025 / Accepted: 7 March 2026

Abstract In this work, we proposed two modified spectral algorithms for solving a system of nonlinear monotone equations with convex constraints. We have introduced an inertial step in these algorithms to investigate its effect on the iteration process. We have shown that the two proposed algorithms are sufficiently descent and bounded, and consequently, we have established the global convergence under some suitable suppositions. We conducted some numerical experiments and compared the results with four existing methods, one of which, is an inertial-based method, and the other three with no inertia. Our numerical experiments indicated that incorporating the inertial step in a nonlinear system of equations algorithms has a significant advantage in accelerating the iteration process. Interestingly, we have applied one of the two proposed algorithms in motion control involving a 2D joint planar robotic manipulator.

MSC: 90C56, 90C30, 49J52, 90C90
Keywords: Derivative-Free Method; Spectral Gradient Method; Inertial Step; Nonlinear Monotone Equations

Published online: 31 March 2026
© 2026 By TaCS-CoE, All rights reserve.

Published by Center of Excellence in Theoretical and Computational Science (TaCS-CoE)

Please cite this article as: S. Aji et al., Spectral Algorithms with Inertial-strategy for Constrained System of Nonlinear Equations, Bangmod Int. J. Math. & Comp. Sci., Vol. 12 (2026), 58–80. https://doi.org/10.58715/bangmodjmcs.2026.12.4

1. Introduction

Let $C \subset \mathbb{R}^n$ be a nonempty closed and convex set, consider a continuous mapping $H : \mathbb{R}^n \to \mathbb{R}^n$. System of nonlinear equations can be defined as

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

It is considered system of nonlinear monotone equations when the mapping H satisfies

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

System of nonlinear equations arise in many practical domains ranging from economic equilibrium problems [1], power flow equations [2], motion control [3], chemical equilibrium systems [4], signal and image recovery [5–7].

Although, traditional methods such as Newton and Quasi Newton methods for solving the system have advantage of rapid convergence from good initial points, problems such as Jacobian matrix computation and storage or an approximate of such a matrix made them impractical for large-scale systems [8–13]. Alternatively, methods such as spectral and conjugate gradient methods do not require those computations, and storage is handy and, thus, more suitable for large-scale problems.

Spectral and conjugate gradient methods were proposed to solve unconstrained optimization problems, that is, problems of the form $\min_{x\in\mathbb{R}^n} f(x)$, and $f : \mathbb{R}^n \to \mathbb{R}$ is a smooth nonlinear function. The sequence of approximations in these methods is generated using:

$$x_{k+1} := x_k + \Theta_k d_k, \tag{1.2}$$

in which Θ_k is known as step-size or step-length and d_k is the search direction. More details on spectral and conjugate gradient methods for unconstrained optimization problems can be found in [14–19]. Motivated by the work of Barzilai and Borwein (BB) [20], La Cruz and Raydan [21] extended the spectral gradient method from solving unconstrained optimization problems to solve system of nonlinear equations. They proposed the spectral approach for nonlinear equations (SANE) algorithm, proved the global convergence based on nonmonotone line search techniques and presented some numerical experiments. Their work was a great inspiration for solving nonlinear system of equations using spectral gradient approach. Currently, many spectral, conjugate and spectral conjugate gradient approaches for dealing with system of nonlinear equations exist in literature, some of these can be found in [22–35].

The hyperplane projection technique proposed by Solodov and Svaiter in [36] motivated many researchers to develop iterative methods for solving (1.1). In [37], Zhang and Zhou proposed a spectral gradient projection method (SGPM) for solving nonlinear system of equations. To prove the global convergence of the method, suppositions that the mapping under consideration is Lipschitz continuous and monotone are imposed. Numerical results presented in the work indicated that their method has some advantages when compared with the inexact Newton method (INM) proposed in [36] and the SANE method proposed in [21]. Moreover, Yu et al. [38] consider SGPM developed in [37] to handle convex-constrained monotone problems which are equally nonlinear.

They defined the search direction in their work as

$$d_k := \begin{cases} -H(x_k), & \text{if } k = 0, \\ -\tilde{\theta}_k H(x_k), & \text{if } k \geq 1, \end{cases} \tag{1.3}$$

where $\tilde{\theta}_k$ is given by

$$\tilde{\theta}_k = \frac{\|s_{k-1}\|^2}{s_{k-1}^T y_{k-1}},$$

in which $s_{k-1} = x_k - x_{k-1}$ and $y_{k-1} = H(x_k) - H(x_{k-1}) + rs_{k-1}$. Awwal et al. [39] proposed another proposed approach for solving (1.1). In their work, they introduced a bounded parameter $\hat{\theta}_k$ in the search direction and defined it as

$$\hat{\theta}_k = 1 - \frac{(H(x_k)^T d_{k-1})^2}{\|H(x_k)\|^2 \|d_{k-1}\|^2}.$$

Inspired by the work of Yu et al. [38] and Awwal et al. [39], Zheng et al. [40] proposed another method for solving (1.1). The search direction in their method is defined as

$$d_k := \begin{cases} -H(x_k), & \text{if } k = 0, \\ -\lambda_k H(x_k), & \text{if } k \geq 1, \end{cases} \tag{1.4}$$

where

$$\lambda_k = \gamma_k \cdot \theta_k, \tag{1.5}$$

$$\gamma_k = \frac{\|s_{k-1}\|^2}{s_{k-1}^T y_{k-1}}, \tag{1.6}$$

$$\theta_k = 1 - \mu \cdot \frac{(H(x_k)^T d_{k-1})^2}{\overline{H(x_k)}^2 \|d_{k-1}\|^2}, \tag{1.7}$$

where

$$s_{k-1} = z_{k-1} - x_{k-1}, y_{k-1} = H(z_{k-1}) - H(x_{k-1}) + rs_{k-1},$$
$$\overline{H(x_k)} = \max\{\|H(x_{k-1})\|, \|H(x_k)\|\},$$

r and μ are constants. It is worth noting that setting $\mu = 0$ and replacing z_{k-1} by x_{k-1}, the parameter θ_k and $\hat{\theta}_k$ in [39] are the same.

In recent times, many researchers are inspired to introduce inertial step proposed by Polyak [41] into nonlinear system algorithms. Using two previous iterates to compute the current iterate, the inertial step proved to speed up methods for solving equilibrium problems such as the proximal point algorithm [42], auxiliary problem principle [43], etc. The works of Awwal et al. [44] and Aji et al. [45] have shown that, not only in equilibrium problems, but also, the inertial step has improved numerical performance in their algorithms proposed for solving nonlinear system of equations.

In this work, to further investigate the effect of the inertial step in algorithms for solving system of nonlinear equations, we proposed a modification of the work in [40], and developed two inertial-based spectral algorithms for solving system of nonlinear monotone equations with convex constraints. The two proposed algorithms are based on the following points:

- For the first algorithm, we have introduced the inertial step in the spectral parameter of the direction (1.4). The modified spectral parameter is given as

$$\overline{\lambda}_k = \overline{\gamma}_k \cdot \widehat{\theta}_k, \tag{1.8}$$

where

$$\overline{\gamma}_k = \frac{\|w_{k+1} - w_k\|^2}{(w_{k+1} - w_k)^T(H(w_{k+1}) - H(w_k) + r(w_{k+1} - w_k))}, \tag{1.9}$$

$$\widehat{\theta}_k = 1 - \mu \cdot \frac{(H(x_k)^T(w_{k+1} - w_k))^2}{\overline{H(x_k)}^2 \|w_{k+1} - w_k\|^2}, \tag{1.10}$$

$\overline{H(x_k)} = \max\{\|H(x_{k-1})\|, \|H(x_k)\|\}$, μ, r are constants, and $w_{k+1} \neq w_k$. Here, w_k is defined as $w_k := x_k + \alpha_k(x_k - x_{k-1})$ and is called the inertial step, where $\alpha_k \in (0, 1)$.

- For the second algorithm, we have defined a new spectral parameter $\widehat{\lambda}_k$

$$\widehat{\lambda}_k = \widehat{\gamma}_k \cdot \widehat{\theta}_k, \tag{1.11}$$

where

$$\widehat{\gamma}_k = \frac{(w_{k+1} - w_k)^T(H(w_{k+1}) - H(w_k) + r(w_{k+1} - w_k))}{\|H(w_{k+1}) - H(w_k) + r(w_{k+1} - w_k)\|^2}, \tag{1.12}$$

and $\widehat{\theta}_k$ is given as (1.10). This corresponds to a new search direction for solving (1.1).

It can be observed that the parameters $\overline{\gamma}_k$ and $\widehat{\gamma}_k$ are respectively modified popular BB1 and BB2 parameters proposed by Barzilai and Borwein [20]. The contribution of this paper include the following:

- We have proposed two inertial-based algorithms for solving system of nonlinear equations with convex constraint.
- We have proved that the search directions in these algorithms are sufficiently descent and bounded, and consequently, we proved the global convergence.
- We have performed some numerical experiments and compared the performance of our algorithms with some exisiting ones namely: the DAIS1 algorithm proposed by Awwal et al. [44], the MSGP algorithm proposed by Zheng et al. [40], the SGPM algorithm proposed by Zhang and Zhou [37] and the Limited memory BFGS algorithm (LBFGS) proposed by Zhou and Li [12].
- We have applied one of the two proposed algorithms in motion control involving a 2D joint planar robotic manipulator.

We have organized the remaining part of the work as follows: In the next section, we have given some definitions, details of the proposed methods, and their global convergence. In section three, we have presented some numerical experiments with the proposed algorithms on some test problems. Then, we showed a comparison with some existing algorithms proposed in the literature. Finally, we verified the applicability of the proposed algorithm in robotics.

2. Developed algorithms and some convergence result

This section begins with some important definitions stated as follows:

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

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

The projection map satisfies the following property

$$\|P_C(x) - y\| \leq \|x - y\|, \quad \forall y \in C. \tag{2.1}$$

Definition 2.2. A function H is said to be Lipschitz continuous if there exists $L > 0$ such that

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

We now proceed with the steps of our proposed algorithms below:

Algorithm 1: Inertial-Based Spectral Gradient Projection Method (IMSGP1)

Input : Initial parameters x_{-1}, $x_0 \in C$, $\sigma, \mu, \varsigma \in (0,1)$, $\Gamma \in (0,2)$, $r, \kappa > 0$, $\alpha_k \in (0,1)$ such that $\lim_{k\to\infty} \alpha_k = 0$ and $Tol > 0$.

Step 0: Set $k = 0$, compute $d_0 := -H(x_0)$ and $w_0 := x_0 + \alpha_0(x_0 - x_{-1})$.

Step 1: If $\|H(x_k)\| \leq Tol$, stop, else move on with **Step 2**.

Step 2: Set

$$z_k := x_k + \Theta_k d_k, \ \Theta_k = \kappa\varsigma^i, \tag{2.2}$$

in which $i \in \mathbb{Z}^+$ such that

$$-\langle H(x_k + \kappa\varsigma^i d_k), d_k\rangle \geq \sigma\kappa\varsigma^i\|d_k\|^2 \min\{1, \ \|H(x_k + \kappa\varsigma^i d_k)\|^{\frac{1}{c}}\}, \ c \geq 1. \tag{2.3}$$

Step 3: If $\|H(z_k)\| = 0$, stop. Otherwise, evaluate

$$x_{k+1} := P_C\left[x_k - \Gamma\frac{H(z_k)^T(x_k - z_k)}{\|H(z_k)\|^2}H(z_k)\right]. \tag{2.4}$$

Step 4: Compute $w_{k+1} := x_{k+1} + \alpha_k(x_{k+1} - x_k)$.

Step 5: Set $k := k + 1$, update the search direction and repeat the process from **Step 1.**

$$d_k := -\overline{\lambda}_k H(x_k), \ k = 1, 2, \ldots, \tag{2.5}$$

where $\overline{\lambda}_k$ is given by (1.8).

Algorithm 2: Inertial-Based Spectral Gradient Projection Method II (IMSGP2)

Input : Choose the parameters as in **Step 1** of Algorithm 1, and continue with the steps.

Update the search direction in **Step 5** of Algorithm 1 with the following

$$d_k := -\widehat{\lambda}_k H(x_k), \ k = 1, 2, \ldots, \tag{2.6}$$

where $\widehat{\lambda}_k$ is given by (1.11).

Remark 2.3. The computation of the step size using (2.3) was proposed by Awwal et al [44] and more details about it can be found in the references [31, 44].

The global convergence of IMSGP1 and IMSGP2 relies on the following suppositions on the mapping H:

(Q_1) Monotonic supposition of H,
(Q_2) Its Lipschitz continuity supposition, and
(Q_3) Non-emptiness of the solution set for (1.1).

Lemma 2.4. *Let the suppositions (Q_1)-(Q_3) hold, and that $\{x_k\}$ and $\{w_k\}$ are generated by Algorithm 1, then we have the following results,*

(i) $\lim_{k\to\infty} \|x_k - \widehat{x}\|$ *exists.*

(ii) $\{x_k\}$, $\{w_k\}$ *and* $\|H(x_k)\|$ *are bounded.*

(iii) d_k *is bounded, i.e.*

$$\|d_k\| \leq M, \ M > 0. \tag{2.7}$$

(iv) d_k *also satisfies*

$$H(x_k)^T d_k \leq -t_1 \|H(x_k)\|^2, \ t_1 > 0. \tag{2.8}$$

(v)

$$\lim_{k\to\infty} \Theta_k \|d_k\| = 0. \tag{2.9}$$

Proof. (i) Suppose $\widehat{x}$ is a solution of (1.1), then using supposition (Q_1), we can have

$$\begin{aligned} &\langle H(z_k), \ x_k - \widehat{x}\rangle \\ &= \langle H(z_k), \ x_k - z_k + z_k - \widehat{x}\rangle \\ &= \langle H(z_k), \ x_k - z_k\rangle + \langle H(z_k) - H(\widehat{x}), \ z_k - \widehat{x}\rangle \\ &\geq \langle H(z_k), \ x_k - z_k\rangle. \end{aligned} \tag{2.10}$$

Now, utilizing the above equation (2.10) and x_{k+1} gives

$$\begin{aligned} &\|x_{k+1} - \widehat{x}\|^2 \\ &= \left\| P_\Omega \left[x_k - \Gamma \frac{\langle H(z_k), \ x_k - z_k\rangle}{\|H(z_k)\|^2} H(z_k) \right] - \widehat{x} \right\|^2 \qquad (2.11) \\ &\leq \left\| x_k - \widehat{x} - \Gamma \frac{\langle H(z_k), \ x_k - z_k\rangle}{\|H(z_k)\|^2} H(z_k) \right\|^2 \\ &= \|x_k - \widehat{x}\|^2 - 2\Gamma \frac{\langle H(z_k), \ x_k - z_k\rangle}{\|H(z_k)\|^2} \langle H(z_k), \ x_k - \widehat{x}\rangle + \Gamma^2 \frac{\langle H(z_k), \ x_k - z_k\rangle^2}{\|H(z_k)\|^2} \\ &\leq \|x_k - \widehat{x}\|^2 - 2\Gamma \frac{\langle H(z_k), \ x_k - z_k\rangle}{\|H(z_k)\|^2} \langle H(z_k), \ x_k - z_k\rangle + \Gamma^2 \frac{\langle H(z_k), \ x_k - z_k\rangle^2}{\|H(z_k)\|^2} \\ &= \|x_k - \widehat{x}\|^2 - \Gamma(2 - \Gamma) \frac{\langle H(z_k), \ x_k - z_k\rangle^2}{\|H(z_k)\|^2} \qquad (2.12) \\ &\leq \|x_k - \widehat{x}\|^2, \end{aligned}$$

showing that for all k, $\|x_k - \widehat{x}\| \leq \|x_0 - \widehat{x}\|$ and $\lim_{k\to\infty} \|x_k - \widehat{x}\|$ exists.

(ii) We already have for all k, $\|x_k - \widehat{x}\| \leq \|x_0 - \widehat{x}\|$, thus $\{x_k\}$ is bounded. For the boundedness of $\{w_k\}$, we know that $\alpha_k \in (0, 1)$ and $\{x_k\}$ is bounded, then $\{w_k\}$ is also bounded.

Additionally, with Lipschitz continuity supposition on H and the bound on $\{x_k\}$, then a positive constant say, $m_1 > 0$ can be found where

$$\|H(x_k)\| \leq m_1. \tag{2.13}$$

(iii) To establish the boundedness of the search direction formulated in (2.5) we first require to show that the parameter $\overline{\lambda}_k$ defined by (1.8) is bounded. That is, we need to

show $\overline{\gamma}_k$ and $\widehat{\theta}_k$ are bounded.
From supposition (Q_1), the mapping H is monotone. This give us

$$\langle H(w_{k+1}) - H(w_k),\ w_{k+1} - w_k\rangle \geq 0.$$

Considering the denominator of (1.9), we have

$$\begin{aligned}&\langle H(w_{k+1}) - H(w_k) + r(w_{k+1} - w_k),\ w_{k+1} - w_k\rangle \\ &= \langle H(w_{k+1}) - H(w_k),\ w_{k+1} - w_k\rangle + r\|w_{k+1} - w_k\|^2 \\ &\geq r\|w_{k+1} - w_k\|^2.\end{aligned} \tag{2.14}$$

Furthermore, we will use the supposition (Q_2) on H and with Cauchy Schwartz inequality to establish as follows:

$$\begin{aligned}&\langle H(w_{k+1}) - H(w_k) + r(w_{k+1} - w_k),\ w_{k+1} - w_k\rangle \\ &= \langle H(w_{k+1}) - H(w_k),\ w_{k+1} - w_k\rangle + r\|w_{k+1} - w_k\|^2 \\ &\leq (L + r)\|w_{k+1} - w_k\|^2.\end{aligned} \tag{2.15}$$

Thus, (2.14) and (2.15) implies

$$\begin{aligned}r\|w_{k+1} - w_k\|^2 &\leq \langle H(w_{k+1}) - H(w_k) + r(w_{k+1} - w_k),\ w_{k+1} - w_k\rangle \\ &\leq (L + r)\|w_{k+1} - w_k\|^2\end{aligned} \tag{2.16}$$

dividing through by $\|w_{k+1} - w_k\|^2$, and inverting the fraction gives

$$\frac{1}{L + r} \leq \frac{\|w_{k+1} - w_k\|^2}{\langle H(w_{k+1}) - H(w_k) + r(w_{k+1} - w_k),\ w_{k+1} - w_k\rangle} \leq \frac{1}{r} \tag{2.17}$$

showing that $\overline{\gamma}_k$ is bounded.

Now, to prove $\widehat{\theta}_k$ defined by (1.10) is bounded, we begin by discussing the parameter $\widehat{\theta}_k$ in two ways. First, if $\|H(x_{k-1})\| \leq \|H(x_k)\|$, we have by Cauchy-Schwarz inequality and the definition of $\overline{H(x_k)}$

$$0 \leq \frac{(H(x_k)^T(w_{k+1} - w_k))^2}{\|\overline{H(x_k)}\|^2\|w_{k+1} - w_k\|^2} = \frac{(H(x_k)^T(w_{k+1} - w_k))^2}{\|H(x_k)\|^2\|w_{k+1} - w_k\|^2} \leq 1. \tag{2.18}$$

Again, if $\|H(x_{k-1})\| \geq \|H(x_k)\|$, we have

$$\begin{aligned}0 &\leq \frac{(H(x_k)^T(w_{k+1} - w_k))^2}{\|\overline{H(x_k)}\|^2\|w_{k+1} - w_k\|^2} \\ &= \frac{(H(x_k)^T(w_{k+1} - w_k))^2}{\|H(x_{k-1})\|^2\|w_{k+1} - w_k\|^2} \\ &\leq \frac{(H(x_k)^T(w_{k+1} - w_k))^2}{\|H(x_k)\|^2\|w_{k+1} - w_k\|^2} \\ &\leq 1.\end{aligned} \tag{2.19}$$

From (1.10), (2.18) and (2.19) we have

$$1 - \mu \leq \widehat{\theta}_k \leq 1. \tag{2.20}$$

From (1.8), (2.17) and (2.20), we have

$$t_1 \leq \overline{\lambda}_k \leq t_2, \tag{2.21}$$

where $t_1 = \frac{1-\mu}{L+r}$ and $t_2 = \frac{1}{r}$.
From the definition of the search direction (2.5), (2.13) and (2.21), we have

$$\|d_k\| \leq M, \tag{2.22}$$

where $M = t_2 m_1$. Showing that the search direction is bounded.

(iv) From (2.5) and (2.21) we have

$$\begin{aligned} H(x_k)^T d_k &= -\overline{\lambda}_k \|H(x_k)\|^2 \\ &\leq -t_1 \|H(x_k)\|^2. \end{aligned} \tag{2.23}$$

(v) Using the boundedness of $\{x_k\}$, (2.22) and the definition of z_k in (2.2), we obtain $\{z_k\}$ is bounded. Also, using the supposition that H is Lipschitz continuous, we get

$$\|H(z_k)\| \leq n_1, \ n_1 > 0. \tag{2.24}$$

Since $\min\left\{1, \|H(x_k + \Theta_k d_k)\|^{\frac{1}{c}}\right\} \leq 1$, then squaring both sides of (2.3) yields

$$\sigma^2 \Theta_k^4 \|d_k\|^4 \leq \langle H(z_k), \ \Theta_k d_k \rangle^2. \tag{2.25}$$

Furthermore, since $0 < \Gamma < 2$, then from (2.12) we obtain

$$\Gamma(2-\Gamma)\langle H(z_k), \ x_k - z_k\rangle^2 \leq \|H(z_k)\|^2(\|x_k - \hat{x}\|^2 - \|x_{k+1} - \hat{x}\|^2), \tag{2.26}$$

$$\langle H(z_k), \ x_k - z_k\rangle^2 \leq \frac{\|H(z_k)\|^2(\|x_k - \hat{x}\|^2 - \|x_{k+1} - \hat{x}\|^2)}{\Gamma(2-\Gamma)}. \tag{2.27}$$

This together with (2.25) gives

$$\sigma^2 \Theta_k^4 \|d_k\|^4 \leq \frac{\|H(z_k)\|^2(\|x_k - \hat{x}\|^2 - \|x_{k+1} - \hat{x}\|^2)}{\Gamma(2-\Gamma)}. \tag{2.28}$$

Recall that $\|H(z_k)\|$ is bounded by n_1 (see (2.24)), we obtain

$$\begin{aligned} \sigma^2 \Theta_k^4 \|d_k\|^4 &\leq \frac{\|H(z_k)\|^2\big(\|x_k - \hat{x}\|^2 - \|x_{k+1} - \hat{x}\|^2\big)}{\Gamma(2-\Gamma)} \\ &\leq \frac{n_1^2\big(\|x_k - \hat{x}\|^2 - \|x_{k+1} - \hat{x}\|^2\big)}{\Gamma(2-\Gamma)}. \end{aligned} \tag{2.29}$$

Since $\lim_{k\to\infty} \|x_k - \hat{x}\|$ exists, taking the limit as $k \to \infty$ on both sides of (2.29) gives

$$\sigma^2 \lim_{k\to\infty} \Theta_k^4 \|d_k\|^4 = 0, \tag{2.30}$$

which implies

$$\lim_{k\to\infty} \Theta_k \|d_k\| = 0. \tag{2.31}$$

∎

Lemma 2.5. *Suppose that the suppositions (Q_1)-(Q_3) hold, and the search direction $\{d_k\}$ is generated by Algorithm 2, then, we have the following*

(i) *The search direction is bounded, i.e.*

$$\|d_k\| \leq M_2, \ M_2 > 0. \tag{2.32}$$

(ii) *The search direction satisfies*

$$H(x_k)^T d_k \leq -t_3 \|H(x_k)\|^2, t_3 > 0. \tag{2.33}$$

(iii)

$$\lim_{k\to\infty} \Theta_k \|d_k\| = 0. \tag{2.34}$$

Proof. Observe that

$$\begin{aligned}
&\|H(w_{k+1}) - H(w_k) + r(w_{k+1} - w_k)\|^2 \\
&= \langle H(w_{k+1}) - H(w_k) + r(w_{k+1} - w_k), H(w_{k+1}) - H(w_k) + r(w_{k+1} - w_k)\rangle \\
&= \|H(w_{k+1}) - H(w_k)\|^2 + 2\langle H(w_{k+1}) - H(w_k), w_{k+1} - w_k\rangle + r^2\|w_{k+1} - w_k\|^2 \\
&\geq \|H(w_{k+1}) - H(w_k)\|^2 + r^2\|w_{k+1} - w_k\|^2 \\
&\geq r^2\|w_{k+1} - w_k\|^2.
\end{aligned} \tag{2.35}$$

Therefore,

$$\|H(w_{k+1}) - H(w_k) + r(w_{k+1} - w_k)\| \geq r\|w_{k+1} - w_k\|. \tag{2.36}$$

Using Lipschitz continuity, we have

$$\|H(w_{k+1}) - H(w_k) + r(w_{k+1} - w_k)\| \leq (L + r)\|w_{k+1} - w_k\|. \tag{2.37}$$

This implies

$$\|H(w_{k+1}) - H(w_k) + r(w_{k+1} - w_k)\|^2 \leq (L + r)^2\|w_{k+1} - w_k\|^2. \tag{2.38}$$

Combining (2.35) and (2.38) gives

$$r^2\|w_{k+1} - w_k\|^2 \leq \|H(w_{k+1}) - H(w_k) + r(w_{k+1} - w_k)\|^2 \leq (L+r)^2\|w_{k+1} - w_k\|^2. \tag{2.39}$$

Using (2.16) and (2.39) we have

$$\frac{r}{(L + r)^2} \leq \frac{\langle H(w_{k+1}) - H(w_k) + r(x_k - w_k),\ x_k - w_k\rangle}{\|H(w_{k+1}) - H(w_k) + r(x_k - w_k)\|^2} \leq \frac{(L + r)}{r^2}. \tag{2.40}$$

Showing that the parameter $\widehat{\gamma}_k$ is bounded. That is

$$\frac{r}{(L + r)^2} \leq \widehat{\gamma}_k \leq \frac{(L + r)}{r^2}. \tag{2.41}$$

We also have $\widehat{\theta}_k$ bounded from (2.20). Thus, combining (2.20) and (2.41) we have $\widehat{\lambda}_k$ defined by (1.11) as follows

$$t_3 \leq \widehat{\lambda}_k \leq t_4, \tag{2.42}$$

where $t_3 = \frac{r(1-\mu)}{(L+r)^2}$ and $t_4 = \frac{(L+r)}{r^2}$. Hence, using (2.13)and (2.42) we have the search direction defined by (2.6) as follows

$$\|d_k\| = \widehat{\lambda}_k\|H(x_k)\| \leq t_4 m_1 \tag{2.43}$$

and taking $M_2 = t_4 m_1$, (i) holds i.e,

$$\|d_k\| \leq M_2. \tag{2.44}$$

(ii) From (2.6) and (2.42) we have

$$\begin{aligned} H(x_k)^T d_k &= -\widehat{\lambda}_k \|H(x_k)\|^2 \\ &\leq -t_3 \|H(x_k)\|^2. \end{aligned} \tag{2.45}$$

The proof of (iii) follows directly from the same arguments in Lemma 2.4. ∎

Remark 2.6. In order to prove that the step length Θ_k which satisfies (2.3) exists, we assume that at a point say k_0 the line search (2.3) does not hold for any $i = 0, 1, ...,$ meaning

$$\begin{aligned} &-\langle H(x_{k_0} + \kappa\varsigma^i d_{k_0}), d_{k_0}\rangle \\ &< \sigma\kappa\varsigma^i \|d_{k_0}\|^2 \min\{1,\ \|H(x_{k_0} + \kappa\varsigma^i d_{k_0})\|^{\frac{1}{c}}\} \\ &\leq \sigma\kappa\varsigma^i \|d_{k_0}\|^2. \end{aligned} \tag{2.46}$$

Remember, the mapping H is continuous and $\varsigma \in (0, 1)$, thus, if we let $i \to \infty$ we obtain

$$-H(x_{k_0})^T d_{k_0} \leq 0.$$

This contradicts (2.23). Thus, we can conclude that the step size exists.

Lemma 2.7. *Let the Supposition (Q_2) holds and suppose the sequences $\{x_k\}$ and $\{z_k\}$ are generated using IMSCP1 Algorithm 1. Then we have*

$$\Theta_k \geq \max\left\{\kappa, \frac{t_1\varsigma\|H_k\|^2}{(L+\sigma)\|d_k\|^2}\right\}. \tag{2.47}$$

Proof. From (2.3), if $\Theta_k \neq \kappa$, then $\tilde{\Theta}_k = \Theta_k\varsigma^{-1}$ violates (2.3), that is,

$$-\langle H(x_k + \tilde{\Theta}_k d_k), d_k\rangle < \sigma\|d_k\|^2\tilde{\Theta}_k \min\{1, \|H(x_k + \varsigma^i d_k)\|^{\frac{1}{c}}\}.$$

Since we have $\min\{1, \|H(x_k + \varsigma^i d_k)\|^{\frac{1}{c}}\} \leq 1$. Consequently, we have from relation (2.8) and supposition (Q_2), we have the following

$$\begin{aligned} t_1\|H_k\|^2 &\leq -H_k^T d_k \\ &= (H(x_k + \tilde{\Theta}_k d_k) - H_k)^T d_k - \langle H(x_k + \tilde{\Theta}_k d_k), d_k\rangle \\ &\leq \|H(x_k + \tilde{\Theta}_k d_k) - H(x_k)\|\|d_k\| - \langle H(x_k + \tilde{\Theta}_k d_k), d_k\rangle \\ &\leq L\|x_k + \tilde{\Theta}_k d_k - x_k\|\|d_k\| + \sigma\tilde{\Theta}_k\|d_k\|^2 \\ &\leq \tilde{\Theta}_k L\|d_k\|^2 + \sigma\tilde{\Theta}_k\|d_k\|^2 \\ &\leq \tilde{\Theta}_k(L+\sigma)\|d_k\|^2. \end{aligned}$$

Therefore,

$$\tilde{\Theta}_k \geq \frac{t_1\|H_k\|^2}{(L+\sigma)\|d_k\|^2}, \tag{2.48}$$

substituting $\tilde{\Theta}_k = \Theta_k\varsigma^{-1}$ in (2.48) and solving for Θ_k we get

$$\Theta_k \geq \frac{t_1\varsigma\|H_k\|^2}{(L+\sigma)\|d_k\|^2}. \tag{2.49}$$

Thus, we have

$$\Theta_k \geq \max\left\{\kappa, \frac{t_1\varsigma\|H_k\|^2}{(L+\sigma)\|d_k\|^2}\right\}. \tag{2.50}$$

∎

Theorem 2.8. *Suppose that the suppositions ((Q_1)-(Q_3)) hold. Let the sequence $\{x_k\}$ be generated by Algorithm 1, then*

$$\liminf_{k\to\infty} \|H(x_k)\| = 0. \tag{2.51}$$

Proof. We prove by contradiction. Suppose (2.51) does not hold, then there exists $p > 0$ such that $\forall k \geq 0$,

$$\|H(x_k)\| \geq p. \tag{2.52}$$

From equation (2.8) and (2.52), we get $\forall k \geq 0$,

$$\|d_k\| \geq t_1 p. \tag{2.53}$$

Multiplying $\|d_k\|$ both sides of expression (2.47), and then by using (2.22) and (2.52), we obtain

$$\begin{aligned} \Theta_k \|d_k\| &\geq \max\left\{\kappa, \frac{t_1 \varsigma \|H_k\|^2}{(L+\sigma)\|d_k\|^2}\right\} \|d_k\| \\ &\geq \max\left\{\kappa t_1 p, \frac{t_1 \varsigma p^2}{(L+\sigma)M}\right\}. \end{aligned} \tag{2.54}$$

Taking the limit on both of the above expression as $k \to \infty$, we have

$$\lim_{k\to\infty} \Theta_k \|d_k\| > 0, \tag{2.55}$$

which contradicts (2.9). Hence, $\liminf_{k\to\infty} \|H(x_k)\| = 0$. ∎

Remark 2.9. Suppose that the suppositions ((Q_1)-(Q_3)) hold and let the sequence $\{x_k\}$ be generated by Algorithm 2, then by Lemma 2.5, the results in Lemma 2.7 and Theorem 2.8 hold.

3. Numerical experiments

In this section, we have discussed and presented the numerical experiments performed in this work. Recall that we proposed two algorithms, firstly, by introducing the inertial effect to the MSGP algorithm proposed by Zheng et al. [40], and the second algorithm is also inertial-based modified MSGP algorithm. Thus, we first compared the performances of our two proposed algorithms (IMSGP1 and IMSGP2) with three different algorithms: Inertial-based algorithm named DAISI proposed by Awwal et al. [44] for solving system of nonlinear-equations, the MSGP algorithm proposed by Zheng et al. [40] and the spectral gradient projection method (SGPM) proposed by Zhang and Zhou in [37]. 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 our proposed IMSGP1 and IMSGP2 algorithms, we choose $\varsigma = 0.47$, $\alpha_k = 1/(k+1)^2$, $\Gamma = 1.85$, $\mu = 0.59$, $\sigma = 0.01$, $c = 2$, and $\kappa = 1$. While in DAIS1, MSGP and SGPM, we maintained the parameters as reported in [44], [40] and [37] respectively. Moreover, we set $\|H_k\| < 10^{-6}$ as stopping criteria. With this setting, the experiment is performed on seven problems, five different dimensions: $1000, 5000, 10000, 50000, 100000$, and eight initial guesses shown in Table 1 below.

TABLE 1. Initial guesses used for the problems

Initial guess	Values
x_1	$(1, 1, 1, \cdots, 1)^T$
x_2	$(0.1, 0.1, 0.1, \cdots, 0.1)^T$
x_3	$(\frac{1}{2}, \frac{1}{2^2}, \frac{1}{2^3}, \cdots, \frac{1}{2^n})^T$
x_4	$(1 - \frac{1}{n}, 1 - \frac{2}{n}, 1 - \frac{3}{n}, \ldots, 0)^T$
x_5	$(0, \frac{1}{n}, \frac{2}{n}, \cdots, \frac{n-1}{n})^T$
x_6	$(1, \frac{1}{2}, \frac{1}{3}, \cdots, \frac{1}{n})^T$
x_7	$(\frac{n-1}{n}, \frac{n-2}{n}, \frac{n-3}{n}, \cdots, 0)^T$
x_8	$(\frac{1}{n}, \frac{2}{n}, \frac{3}{n}, \cdots, 0)^T$

The description of the considered problems with their associated constraints can be found in references [25], [39],[46], [47], [48].

We have performed two experiments to compare each of the IMSGP1 and IMSGP2 algorithms independently with the DAIS1, MSGP, and SGPM algorithms. The experiments are based on four metrics: ITER, which represents the number of iterations, FVAL for a number of function evaluations, TIME for CPU time; and NORM, which represents the norm of the function when an approximate solution was obtained. The details of the results of comparison between the IMSGP1 algorithm and the DAIS1, MSGP and SGPM are tabulated and can be found in the following link https://github.com/sani-aji/IMSGP/blob/main/EXPERIMENT(1).xlsx, and the results of comparison for the IMSGP2 algorithm with the other three algorithms is given in the link https://github.com/sani-aji/IMSGP/blob/main/EXPERIMENT(2).xlsx.

It can be observed from experiment1 and experiment2 that our proposed IMSGP1 and IMSGP2 algorithms respectively outperformed the DAIS1, MSGP and SGPM algorithms significantly by having less number of iterations, function evaluations and CPU time. We have also compared the IMSGP1 & IMSGP2 algorithms with the other three algorithms in terms of average error and CPU time across all the problems. In most cases, the two proposed algorithms have much better error and CPU time compared to the other three algorithms, as reported in Table 2.

TABLE 2. Comparison of Average Error and CPU Time for our proposed IMSGP1 & IMSGP2 with DAIS1, MSGP & SGPM

Problem	**IMSGP 1**		**IMSGP 2**		**DAIS 1**		**MSGP**		**SGPM**	
	Avg. Error	CPU Time	Avg. Error	CPU Time	Avg. Error	CPU Time	Avg. Error	CPU Time	Avg. Error	CPU Time
Problem 1	**1.24E-16**	**0.1398**	**1.24E-16**	**0.1116**	2.36E-07	0.1980	9.17E-07	1.4066	8.11E-07	0.5315
Problem 2	**2.22E-07**	**0.3562**	**6.08E-08**	**0.3328**	2.48E-07	0.3757	9.31E-07	1.7338	7.49E-07	0.8676
Problem 3	**0.00E+00**	**0.0848**	**0.00E+00**	**0.0512**	2.66E-07	0.3046	9.08E-07	1.3898	8.22E-07	0.4940
Problem 4	5.19E-17	0.1075	5.19E-17	0.0511	1.85E-07	0.0896	1.21E-07	0.1793	**0.00E+00**	**0.0284**
Problem 5	**1.78E-07**	0.4624	**1.78E-07**	**0.1938**	2.05E-07	0.2659	8.88E-07	1.5622	7.53E-07	0.7017
Problem 6	**3.93E-17**	**0.1447**	**3.93E-17**	**0.1474**	1.06E-07	0.2977	8.54E-07	1.5749	6.32E-07	0.6159
Problem 7	**4.73E-07**	**0.6452**	**4.70E-07**	2.0378	6.19E-07	0.6524	8.93E-07	2.4698	7.79E-07	0.6993

In addition, we have also adopted the well-known Dolan and Morè performance profile [49] to show the performance of the IMSGP1 and IMSGP2 algorithms graphically in comparison with the DAIS1, MSGP and SGPM algorithms. The comparison of IMSGP1

algorithm with the other three are shown in Figure 1, and for IMSGP2 is presented in Figure 2. It can be observed from Figure 1 (A) and (B) that the IMSGP1 outperformed the DAIS1, MSGP and SGPM algorithms by solving around 68 % of the problems with less number of iterations and function evaluations as compared to the DAIS1, MSGP and SGPM with around 35%, 18% and 16% respectively. In terms of time, Figure 1 (C) shows that IMSGP1 algorithm solved around 50% of the problems with less CPU time in comparison with the other three aglorithms. Similarly, the performance of the IMSGP2 algorithm in comparison with the DAIS1, MSGP and SGP algorithms can be observed from Figure 2.

From these figures, the proposed IMSGP1 and IMSGP2 algorithms have shown better performance in terms of number of iterations, number of function evaluations and CPU time as compared with the DAIS1, MSGP and SGPM algorithms. Thus, not only that the proposed inertial step algorithms have better performance than the MSGP and SGPM algorithms with no inertia, but also, the proposed algorithms outperformed DAIS1 algorithm which is among the recent inertial algorithms proposed in literature.

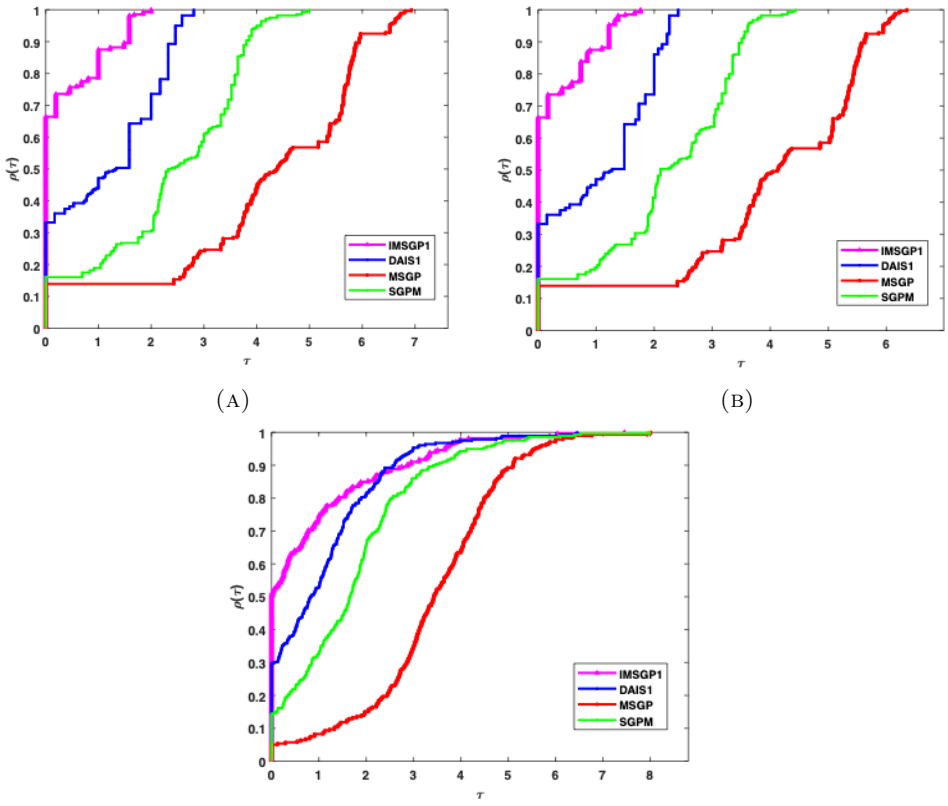


Figure 1. Performance profiles of IMSGP1, DAIS1, MSGP and SGPM on Number of Iterations, Function Evaluations and CPU Time: (A) Performance on Number of Iterations (B) Performance on Number of Function Evaluations (C) Performance on CPU Time

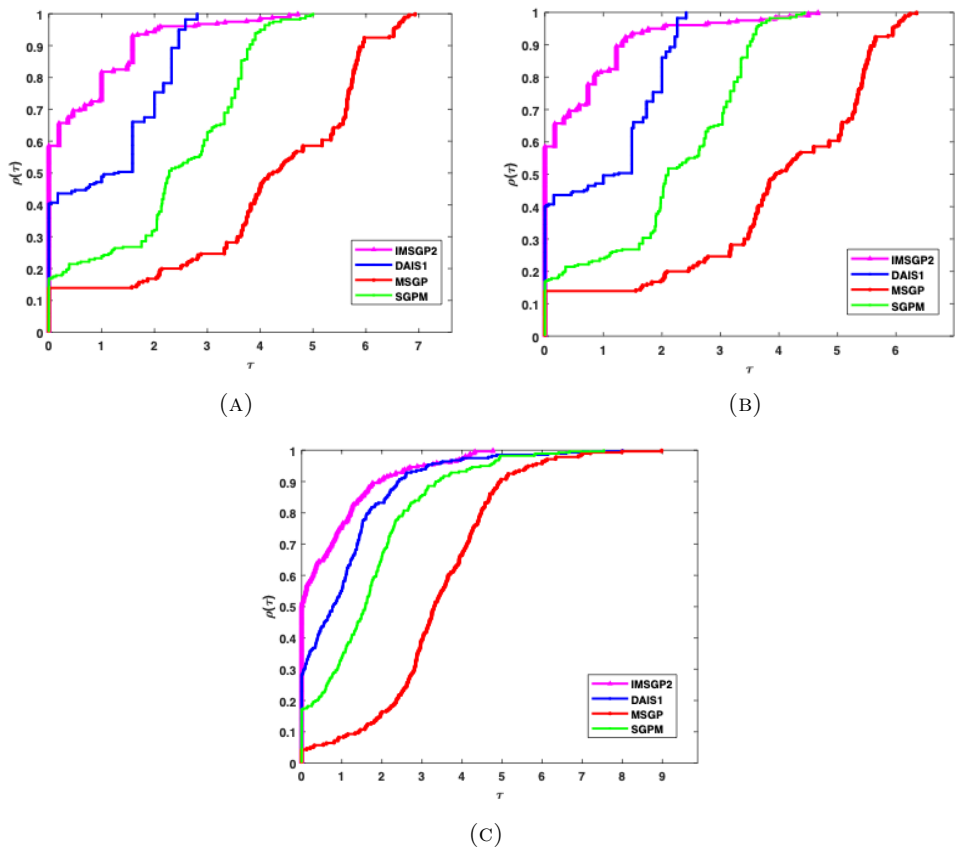


Figure 2. Performance Profiles of IMSGP2, DAIS1, MSGP and SGPM on Number of Iterations, Function Evaluations and CPU Time: (A) Performance on Number of Iterations (B) Performance on Number of Function Evaluations (C) Performance on CPU Time

To further depict the advantages of our two proposed algorithms, we consider comparing them with a Quasi-Newton method which is one of the classical approach for solving (1.1). Specifically, we further compared the two proposed algorithms with a limited memory BFGS method (LBFGS) proposed by Zhou and Li in [12]. For the proposed algorithms, all parameters are maintained as in the previous experiments, and for the LBFGS algorithm, the parameters are maintained as reported in [12]. We tested the three algorithms on the seven test problems listed previously. However, unlike the IMSGP 1 and IMSGP 2 algorithms which are matrix-free, the LBFGS method has a limited memory. Consequently, we limited the highest dimension of the problems to 3000, thus, the five dimensions considered in this case are: $500, 1000, 1500, 2000$ and 3000. The results of this experiment are tabulated in seven tables, each for one problem. We show the performance of the three algorithms on problem 1 with the different dimensions and initial points in Table 3, however, the whole tables can be found at the link: https://github.com/sani-aji/IMSGP/blob/main/IMSGPLBFGS.xlsx.

TABLE 3. Numerical Results of the **IMSGP 1**, **IMSGP 2** and **LBFGS** Algorithms on Problem 1 with given initial points and dimensions

		IMSGP 1				IMSGP 2				LBFGS			
DIMENSION	INITIAL POINT	ITER	FVAL	TIME	NORM	ITER	FVAL	TIME	NORM	ITER	FVAL	TIME	NORM
500	x1	1	3	0.3363	0.00E+00	1	3	0.0089	0.00E+00	32	95	1.5270	8.24E-05
	x2	1	3	0.0341	0.00E+00	1	3	0.0103	0.00E+00	42	125	1.7389	8.90E-05
	x3	2	5	0.0451	9.93E-16	2	5	0.0111	9.93E-16	22	66	0.8993	8.21E-05
	x4	2	5	0.0564	0.00E+00	2	5	0.0349	0.00E+00	31	92	1.4252	8.46E-05
	x5	2	5	0.0247	0.00E+00	2	5	0.0147	0.00E+00	32	95	2.1461	7.40E-05
	x6	2	5	0.0068	0.00E+00	3	7	0.0056	0.00E+00	23	68	0.8711	7.75E-05
	x7	2	5	0.0084	0.00E+00	2	5	0.0039	0.00E+00	31	92	1.3794	8.46E-05
	x8	2	5	0.0066	0.00E+00	2	5	0.0043	0.00E+00	32	95	2.4766	7.42E-05
1000	x1	1	3	0.0585	0.00E+00	1	3	0.0038	0.00E+00	33	98	8.3951	7.95E-05
	x2	1	3	0.0042	0.00E+00	1	3	0.0034	0.00E+00	44	131	9.3471	8.77E-05
	x3	2	5	0.0142	9.93E-16	2	5	0.0075	9.93E-16	22	66	5.3692	8.21E-05
	x4	2	5	0.0291	0.00E+00	2	5	0.0046	0.00E+00	32	95	11.4861	8.28E-05
	x5	2	5	0.0037	0.00E+00	2	5	0.0080	0.00E+00	33	98	10.2816	7.17E-05
	x6	2	5	0.0056	0.00E+00	3	7	0.0076	0.00E+00	23	69	5.6058	7.76E-05
	x7	2	5	0.0054	0.00E+00	2	5	0.0054	0.00E+00	32	95	7.4386	8.28E-05
	x8	2	5	0.0134	0.00E+00	2	5	0.0094	0.00E+00	33	98	7.2683	7.18E-05
1500	x1	1	3	0.0144	0.00E+00	1	3	0.0058	0.00E+00	33	98	21.5713	9.72E-05
	x2	1	3	0.0250	0.00E+00	1	3	0.0046	0.00E+00	44	131	27.7636	9.28E-05
	x3	2	5	0.0063	9.93E-16	2	5	0.0058	9.93E-16	22	66	15.0275	8.21E-05
	x4	2	5	0.0064	0.00E+00	2	5	0.0082	0.00E+00	33	98	19.9229	6.96E-05
	x5	2	5	0.0083	0.00E+00	2	5	0.0069	0.00E+00	33	98	18.9370	8.78E-05
	x6	2	5	0.0063	0.00E+00	3	7	0.0101	0.00E+00	23	68	19.0803	7.76E-05
	x7	2	5	0.0055	0.00E+00	2	5	0.0051	0.00E+00	33	98	27.9438	6.96E-05
	x8	2	5	0.0065	0.00E+00	2	5	0.0065	0.00E+00	33	98	19.0171	8.79E-05
2000	x1	1	3	0.0310	0.00E+00	1	3	0.0057	0.00E+00	34	101	44.4713	7.67E-05
	x2	1	3	0.0047	0.00E+00	1	3	0.0035	0.00E+00	44	131	67.1381	9.79E-05
	x3	2	5	0.0062	9.93E-16	2	5	0.0061	9.93E-16	22	66	30.2062	8.21E-05
	x4	2	5	0.0072	0.00E+00	2	5	0.0063	0.00E+00	33	98	46.6778	8.09E-05
	x5	2	5	0.0080	0.00E+00	2	5	0.0101	0.00E+00	34	101	47.5215	6.94E-05
	x6	2	5	0.0072	0.00E+00	3	7	0.0071	0.00E+00	23	68	32.5572	7.76E-05
	x7	2	5	0.0082	0.00E+00	2	5	0.0061	0.00E+00	33	98	45.6811	8.09E-05
	x8	2	5	0.0062	0.00E+00	2	5	0.0288	0.00E+00	34	101	47.3020	6.95E-05
3000	x1	1	3	0.0366	0.00E+00	1	3	0.0087	0.00E+00	34	101	153.0266	9.39E-05
	x2	1	3	0.0087	0.00E+00	1	3	0.0070	0.00E+00	45	134	207.9230	9.01E-05
	x3	2	5	0.0130	9.93E-16	2	5	0.0208	9.93E-16	22	66	99.6501	8.21E-05
	x4	2	5	0.0130	0.00E+00	2	5	0.0115	0.00E+00	33	99	150.5104	9.94E-05
	x5	2	5	0.0123	0.00E+00	2	5	0.0105	0.00E+00	34	101	182.8506	8.50E-05
	x6	2	5	0.0115	0.00E+00	3	7	0.0112	0.00E+00	23	68	116.4740	7.77E-05
	x7	2	5	0.0116	0.00E+00	2	5	0.0102	0.00E+00	33	99	164.9006	9.94E-05
	x8	2	5	0.0131	0.00E+00	2	5	0.0085	0.00E+00	34	101	158.9259	8.51E-05

It can be observed that for all the metrics, that is, the ITER, FVAL and TIME, our two proposed algorithms significantly outperformed the LBFGS by having less number of iterations, number of function evaluations and CPU time. This can be visualized in the following graphs which are plotted using the same Dolan and Morè performance profile [49].

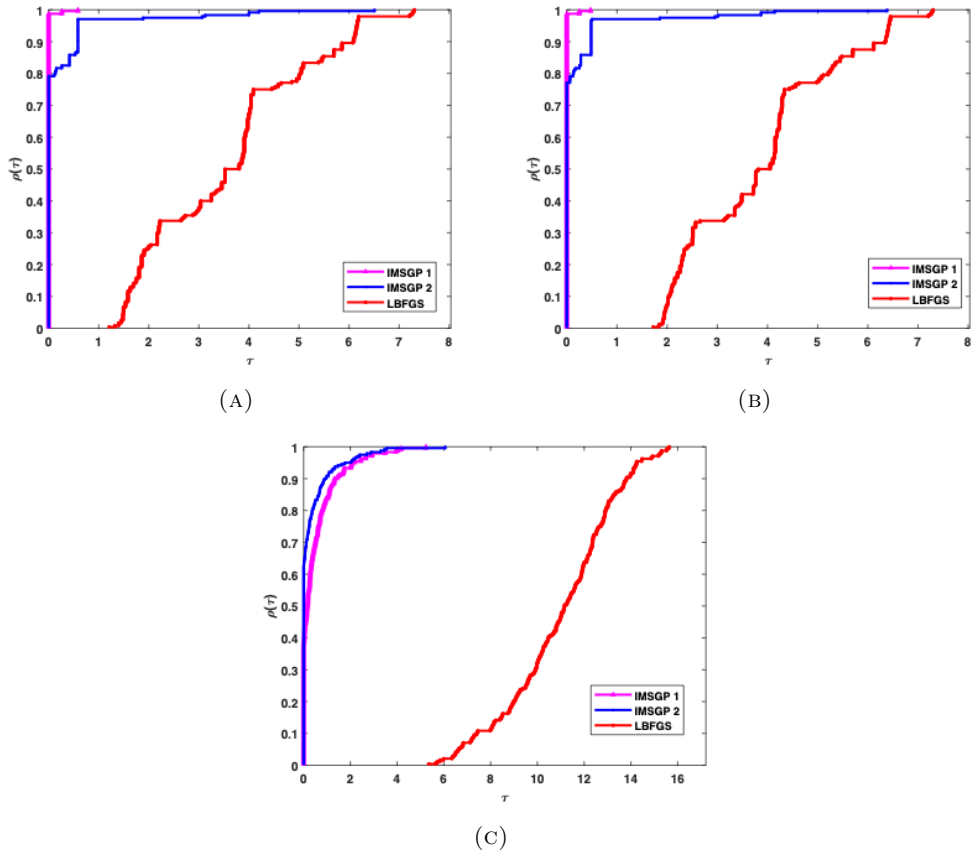


Figure 3. Performance profiles of IMSGP 1, IMSGP 2, and LBFGS on Number of Iterations, Function Evaluations and CPU Time: (a) Performance on Number of Iterations (b) Performance on Number of Function Evaluations (c) Performance on CPU Time

It is also worth mentioning that, although while comparing with LBFGS method, the highest dimension of the problems is limited to 3000, in the previous experiments, the proposed IMSGP 1 and IMSGP 2 algorithms were able to handle problems with up to 100000 dimension. This, in addition to other advantages of low storage requirement, proved their suitability to handle large scale problems.

4. Robotic motion control

Manipulators and effectors play a crucial role in robotics, as these components enable robots to interact with objects by executing various tasks such as picking and placing items into containers. Achieving stability and precision in the movement of robots involves careful consideration of the motor dynamics characteristics, which serve as actuators in both n-link and 1-link robot systems [50, 51]. Addressing the tracking control problem in nonlinear systems, it is essential for the motor dynamics to meet the condition that the system's actual output accurately follows the desired trajectory with minimal error [52].

A lot of methods have been developed to deal with the tracking control problems of nonlinear systems including proportional-integral-derivative (PID) control [53, 54], feedback linearization [55, 56], and optimal output tracking control by using approximation approach [57]. Our aim in this section is to apply the IMSGP1 algorithm in motion control involving 2D joint planar robotic manipulator.

First, we have modified the IMSGP1 algorithm to fit the motion control model which involves solving problem of the form:

$$\min_{x\in\mathbb{R}^n} h(x), \tag{4.1}$$

where $h:\mathbb{R}^n\to\mathbb{R}$ a smooth and convex function. The modification is given in the steps below:

Algorithm 3: Modified IMSGP1

Input : Let $\alpha_k = 0$ for $k \geq 0$ in the IMSGP1 algorithm (Algorithm 1). Consider the same inputs as in IMSGP1, and let $H(x_k) = \nabla h(x_k)$. Replace Step 2 and Step 3 of IMSGP1 with the following:

Step 2: Compute

$$\Theta_k = \kappa\varsigma^i,$$

where i is the smallest non–negative integer such that

$$h(x_k + \kappa\varsigma^i d_k) - h(x_k) \leq \sigma\kappa\varsigma^i H(x_k)^T d_k. \tag{4.2}$$

Step 3: Update the next iterate via

$$x_{k+1} := x_k + \Theta_k d_k. \tag{4.3}$$

Remark 4.1. We assumed that the lower bound to (4.1) and the level set $\{x : h(x) \leq h(x_0)\}$ is bounded for all $x \in \mathbb{R}^n$. By using supposition Q_2 and Theorem 2 of [3], we conclude that Algorithm 3 converges. That is $\liminf_{k\to\infty} \|H(x_k)\| = 0$.

As reported in [58] and [59], the discrete-time kinematics equation of two-joint planar robot manipulator at the position level can be expressed as:

$$\psi(\theta_k) = p_k \tag{4.4}$$

where $\psi(\cdot)$ is the kinematics mapping given as

$$\psi(\theta) = \begin{bmatrix} l_1\cos(\theta_1) + l_2\cos(\theta_1+\theta_2) \\ l_1\sin(\theta_1) + l_2\sin(\theta_1+\theta_2) \end{bmatrix}. \tag{4.5}$$

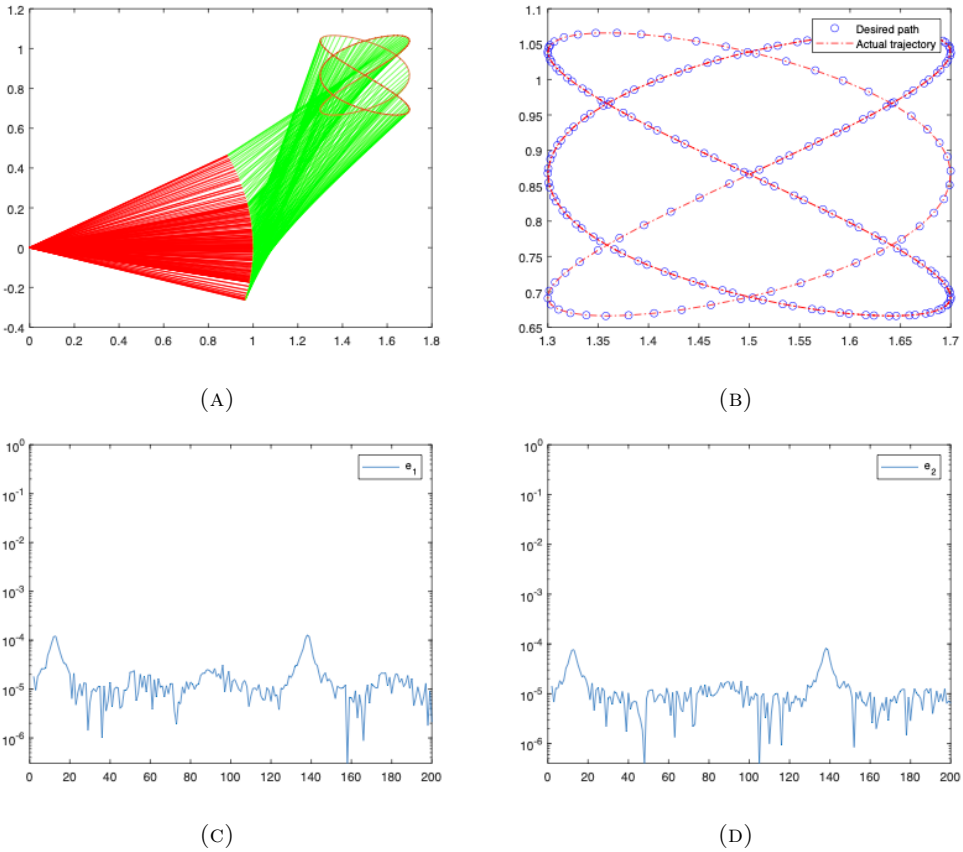
In this case, l_1 and l_2 are the length of the rod links, and $\theta_k \in \mathbb{R}^2$ is the joint angle vector. Moreover, $p_k \in \mathbb{R}^2$ is the end effector vector position. For the motion control problem, it is required to solve the following minimization problem at each computational time interval $t_k \in [0, t_f]$:

$$\min_{p_k\in\mathbb{R}^2} \frac{1}{2}\|p_k - r_{uk}\|_2^2, \tag{4.6}$$

where $p_k \in \mathbb{R}^2$ is controlled to track the following curve:

$$r_{uk} = \begin{bmatrix} \frac{3}{2} + \frac{1}{5}\sin(3t_k) \\ \frac{\sqrt{3}}{2} + \frac{1}{5}\sin(2t_k) \end{bmatrix}. \tag{4.7}$$

It is clear that Algorithm 3 successfully performed the experiments using the following parameters: initial joint angle vector $\theta_0 = [0, \frac{\pi}{3}]^T$, $h_1 = h_2 = 1$, $\tau = 4.2$, $\varsigma = 0.2$ and $\sigma = 0.08$. The time duration task is divided in the interval $[0, 10]$ into 200 equal parts.



(A) (B)

(C) (D)

Figure 4. These figures showcase the simulated result obtained from Algorithm 3 where the Fig. (A) Robot trajectories Fig. (B) End effector trajectory and desired path Fig. (C) Residual error on x-axis Fig. (D) Residual error on y-axis

From Figure 4, we can see the graphical results of the expeiments. Figure 4 (A) represents the robot trajectories synthesized by Algorithm 3. Figure 4 (B) shows the end effector and the desired path. Figure 4 (C) and 4 (D) show the performance errors on x and y axis respectively. These figures indicate that Algorithm 3 is able to perform the task of controlling the movement of the robots to follow the desired path with an error of around 10^{-5} on both x and y-axis.

5. Conclusion

This paper proposes two algorithms (IMSGP1 and IMSGP2) for solving system of nonlinear monotone equations with convex constraint. Both of the algorithms are modifications of the work of Zheng et al. [40]. In the first one, we have modified the MSGP algorithm in [40] and introduced the inertial step to it. The second one is also inertial-based algorithm defined using the modified BB 2 parameter. Both of these algorithms are proved to be globally convergent, and their numerical performance is compared with that of the DAIS1, MSGP and SGPM algorithms proposed in [44], [40] and [37] respectively. The IMSGP1 and IMSGP2 are reported to have better numerical performance in comparison with the DAIS1, MSGP and SGPM algorithms. We further performed another experiment to compare our proposed algorithms with a limited memory BFGS method given in [12]. Here also, our algorithms proved to be more efficient in solving the problems with less number of iterations, number of function evaluations and CPU time. In the last section, the IMSGP1 is further remodified to fit the motion control problem. This has made it possible to apply it in motion control involving 2D joint planar robotic manipulator.

Statements and Declarations

Conflict of interest The authors declare that they have no competing interest.

Data availability Not applicable.

References

[1] S.P. Dirkse, M.C. Ferris, MCPLIB: A collection of nonlinear mixed complementarity problems, Optimization Methods and Software 5 (4) (1995) 319–345. Available from: https://doi.org/10.1080/10556789508805619.

[2] A.J. Wood, B.F. Wollenberg, G.B. Sheblé, Power Generation, Operation, and Control, John Wiley & Sons, 2013.

[3] M. Sun, J. Liu, Y. Wang, Two improved conjugate gradient methods with application in compressive sensing and motion control, Mathematical Problems in Engineering 2020 (2020) Article ID 3058345. Available from: https://doi.org/10.1155/2020/9175496.

[4] K. Meintjes, A.P. Morgan, A methodology for solving chemical equilibrium systems, Applied Mathematics and Computation 22 (3) (1987) 333–361.

[5] M.A. Figueiredo, R.D. Nowak, S.J. Wright, Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems, IEEE Journal of Selected Topics in Signal Processing 1 (4) (2007) 586–597. Available from: https://doi.org/10.1109/JSTSP.2007.910281.

[6] Y. Xiao, Q. Wang, Q. Hu, Non-smooth equations based method for ℓ_1-norm problems with applications to compressed sensing, Nonlinear Analysis: Theory, Methods & Applications 74 (11) (2011) 3570–3577. Available from: https://doi.org/10.1016/j.na.2011.02.040.

[7] Y. Xiao, H. Zhu, A conjugate gradient method to solve convex constrained monotone equations with applications in compressive sensing, Journal of Mathematical Analysis and Applications 405 (1) (2013) 310–319. Available from: https://doi.org/10.1016/j.jmaa.2013.04.017.

[8] G. Zhou, K.C. Toh, Superlinear convergence of a Newton-type algorithm for monotone equations, Journal of Optimization Theory and Applications 125 (1) (2005) 205–221. Available from: https://doi.org/10.1007/s10957-005-1908-0.
[9] C.G. Broyden, A class of methods for solving nonlinear simultaneous equations, Mathematics of Computation 19 (92) (1965) 577–593.
[10] J.E. Dennis, J.J. Moré, A characterization of superlinear convergence and its application to quasi-Newton methods, Mathematics of Computation 28 (126) (1974) 549–560.
[11] B. van de Rotten, S.V. Lunel, A limited memory Broyden method to solve high-dimensional systems of nonlinear equations, In EQUADIFF 2003, World Scientific (2005) 196–201. Available from: https://doi.org/10.1142/9789812702067_0021.
[12] W. Zhou, D. Li, Limited memory BFGS method for nonlinear monotone equations, Journal of Computational Mathematics 25 (1) (2007) 89–96.
[13] C.T. Kelley, Iterative Methods for Linear and Nonlinear Equations, SIAM, 1995.
[14] M. Raydan, The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem, SIAM Journal on Optimization 7 (1) (1997) 26–33.
[15] E.G. Birgin, J.M. Martínez, A spectral conjugate gradient method for unconstrained optimization, Applied Mathematics and Optimization 43 (2) (2001) 117–128. Available from: https://doi.org/10.1007/s00245-001-0003-0.
[16] Y.H. Dai, M. Al-Baali, X. Yang, A positive Barzilai–Borwein-like stepsize and an extension for symmetric linear systems, In Numerical Analysis and Optimization, Springer, (2015) 59–75. Available from: https://doi.org/10.1007/978-3-319-17689-5_3.
[17] M. Al-Baali, Y. Narushima, H. Yabe, A family of three-term conjugate gradient methods with sufficient descent property for unconstrained optimization, Computational Optimization and Applications 60 (1) (2015) 89–110. Available from: https://doi.org/10.1007/s10589-014-9662-z.
[18] J. Yahaya, I. Arzuka, M. Isyaku, Descent modified conjugate gradient methods for vector optimization problems, Bangmod International Journal of Mathematical and Computational Science 9 (2023) 72–91. Available from: https://doi.org/10.58715/bangmodjmcs.2023.9.6.
[19] N. Salihu, H.A. Babando, I. Arzuka, S. Salihu, A hybrid conjugate gradient method for unconstrained optimization with application, Bangmod International Journal of Mathematical and Computational Science 9 (2023) 24–44. Available from: https://doi.org/10.58715/bangmodjmcs.2023.9.3.
[20] J. Barzilai, J.M. Borwein, Two-point step size gradient methods, IMA Journal of Numerical Analysis 8 (1) (1988) 141–148.
[21] W. La Cruz, M. Raydan, Nonmonotone spectral methods for large-scale nonlinear systems, Optimization Methods and Software 18 (5) (2003) 583–599. Available from: https://doi.org/10.1080/10556780310001610493.
[22] A.B. Abubakar, P. Kumam, A.M. Awwal, A modified self-adaptive conjugate gradient method for solving convex constrained monotone nonlinear equations with applications to signal recovery problems, Bangmod International Journal of Mathematical and Computational Science 5 (2019) 1–26.
[23] A.H. Ibrahim, P. Kumam, A.B. Abubakar, W. Jirakitpuwapat, J. Abubakar, A hybrid conjugate gradient algorithm for constrained monotone equations with application in compressive sensing, Heliyon 6 (3) (2020) e03466. Available from:

https://doi.org/10.1016/j.heliyon.2020.e03466.
[24] A.B. Abubakar, P. Kumam, A descent Dai-Liao conjugate gradient method for nonlinear equations, Numerical Algorithms 81 (1) (2019) 197–210. Available from: https://doi.org/10.1007/s11075-018-0541-z.
[25] 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, Mathematical and Computational Applications 25 (2) (2020) 27. Available from: https://doi.org/10.3390/mca25020027.
[26] 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. Available from: https://doi.org/10.1109/ACCESS.2020.3020334.
[27] S. Aji, P. Kumam, A.M. Awwal, M.M. Yahaya, K. Sitthithakerngkiet, An efficient DY-type spectral conjugate gradient method for system of nonlinear monotone equations with application in signal recovery, AIMS Mathematics 6 (8) (2021) 8078–8106. Available from: https://doi.org/10.3934/math.2021468.
[28] A.B. Abubakar, P. Kumam, A.H. Ibrahim, P. Chaipunya, S.A. Rano, New hybrid three-term spectral-conjugate gradient method for finding solutions of nonlinear monotone operator equations with applications, Mathematics and Computers in Simulation 187 (2021) 1–19. Available from: https://doi.org/10.1016/j.matcom.2021.07.005.
[29] A.A. Muhammed, P. Kumam, A.B. Abubakar, A. Wakili, A projection Hestenes-Stiefel-like method for monotone nonlinear equations with convex constraints, Thai Journal of Mathematics 16 (2) (2018) 395–410.
[30] Z. Dai, H. Zhu, A modified Hestenes-Stiefel-type derivative-free method for large-scale nonlinear monotone equations, Mathematics 8 (2) (2020) 168. Available from: https://doi.org/10.3390/math8020168.
[31] S. Aji, P. Kumam, A.M. Awwal, M.M. Yahaya, A.M. Bakoji, A new inertial-based method for solving pseudomonotone operator equations with application, Computational and Applied Mathematics 42 (2023) 1. Available from: https://doi.org/10.1007/s40314-022-02135-3.
[32] S. Aji, P. Kumam, W. Kumam, Some first-order-like methods for solving systems of nonlinear equations, In Soft Computing, CRC Press, (2023) 67–85.
[33] S. Aji, A.B. Abubakar, A.I. Kiri, A. Ishaku, A spectral conjugate gradient-like method for convex constrained nonlinear monotone equations and signal recovery, Nonlinear Convex Analysis and Optimization 1 (1) (2022) 1–23.
[34] A.K. Hassan Ibrahim, P. Kumam, A.B. Abubakar, A. Adamu, Accelerated derivative-free method for nonlinear monotone equations with an application, Numerical Linear Algebra with Applications 29 (3) (2022) e2424. Available from: https://doi.org/10.1002/nla.2424.
[35] A. Adamu, D. Kitkuan, T. Seangwattana, An accelerated Halpern-type algorithm for solving variational inclusion problems with applications, Bangmod International Journal of Mathematical and Computational Science 8 (2022) 37–55. Available from: https://doi.org/10.58715/bangmodjmcs.2022.8.4.
[36] 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, Springer, (1998) 355–369.

[37] L. Zhang, W. Zhou, Spectral gradient projection method for solving nonlinear monotone equations, Journal of Computational and Applied Mathematics 196 (2) (2006) 478–484. Available from: https://doi.org/10.1016/j.cam.2005.10.002.

[38] Z. Yu, J. Lin, J. Sun, Y. Xiao, L. Liu, Z. Li, Spectral gradient projection method for monotone nonlinear equations with convex constraints, Applied Numerical Mathematics 59 (10) (2009) 2416–2423. Available from: https://doi.org/10.1016/j.apnum.2009.04.004.

[39] A.M. Awwal, P. Kumam, A.B. Abubakar, A modified conjugate gradient method for monotone nonlinear equations with convex constraints, Applied Numerical Mathematics 145 (2019) 507–520. Available from: https://doi.org/10.1016/j.apnum.2019.05.012.

[40] L. Zheng, L. Yang, Y. Liang, A modified spectral gradient projection method for solving nonlinear monotone equations with convex constraints and its application, IEEE Access 8 (2020) 92677–92686. Available from: https://doi.org/10.1109/ACCESS.2020.2990298.

[41] B.T. Polyak, Some methods of speeding up the convergence of iteration methods, USSR Computational Mathematics and Mathematical Physics 4 (5) (1964) 1–17. Available from: https://doi.org/10.1016/0041-5553(64)90137-5.

[42] A. Moudafi, Proximal point algorithm extended to equilibrium problems, Journal of Natural Geometry 15 (1999) 91–100.

[43] G. Mastroeni, On auxiliary principle for equilibrium problems, In Equilibrium Problems and Variational Models, Springer, (2003) 289–298. Available from: https://doi.org/10.1007/978-1-4613-0239-1_15.

[44] 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. Available from: https://doi.org/10.1109/ACCESS.2020.3045493.

[45] 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. Available from: https://doi.org/10.1109/ACCESS.2021.3056567.

[46] W.J. Zhou, D.H. Li, A globally convergent BFGS method for nonlinear monotone equations without any merit functions, Mathematics of Computation 77 (264) (2008) 2231–2240. Available from: https://doi.org/10.1090/S0025-5718-08-02121-2.

[47] W. La Cruz, J. Martínez, M. Raydan, Spectral residual method without gradient information for solving large-scale nonlinear systems of equations, Mathematics of Computation 75 (255) (2006) 1429–1448. Available from: https://doi.org/10.1090/S0025-5718-06-01840-0.

[48] Z. 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, Applied Numerical Mathematics 59 (10) (2009) 2416–2423. Available from: https://doi.org/10.1016/j.apnum.2009.04.004.

[49] E.D. Dolan, J.J. Moré, Benchmarking optimization software with performance profiles, Mathematical Programming 91 (2) (2002) 201–213. Available from: https://doi.org/10.1007/s101070100263.

[50] Y. Qiang, F. Jing, J. Zeng, Z. Hou, Dynamic modeling and vibration mode analysis for an industrial robot with rigid links and flexible joints, In Proceedings of the

24th Chinese Control and Decision Conference (2012) 3317–3321. Available from: https://doi.org/10.1109/CCDC.2012.6244526.

[51] Y. Zhang, W. Li, B. Qiu, Y. Ding, D. Zhang, Three-state space reformulation and control of MD-included one-link robot system using direct-derivative and Zhang-dynamics methods, In Proceedings of the 29th Chinese Control and Decision Conference (2017) 3724–3729. Available from: https://doi.org/10.1109/CCDC.2017.7979152.

[52] L. Sun, G.Y. Tang, D.X. Gao, B.L. Zhang, Successive Approximation Procedure of Optimal Control for Nonlinear Interconnected Large-Scale Systems with Time-Delays, In Proceedings of the 6th World Congress on Intelligent Control and Automation (2006) 1176–1180. Available from: https://doi.org/10.1109/WCICA.2006.1712532.

[53] E.M. Jafarov, M.N.A. Parlakci, Y. Istefanopulos, A new variable structure PID-controller design for robot manipulators, IEEE Transactions on Control Systems Technology 13 (1) (2004) 122–130. Available from: https://doi.org/10.1109/TCST.2004.838558.

[54] V. Parra-Vega, S. Arimoto, Y.H. Liu, G. Hirzinger, P. Akella, Dynamic sliding PID control for tracking of robot manipulators: theory and experiments, IEEE Transactions on Robotics and Automation 19 (6) (2003) 967–976. Available from: https://doi.org/10.1109/TRA.2003.819600.

[55] J. Na, X. Ren, D. Zheng, Adaptive control for nonlinear pure-feedback systems with high-order sliding mode observer, IEEE Transactions on Neural Networks and Learning Systems 24 (3) (2013) 370–382. Available from: https://doi.org/10.1109/TNNLS.2012.2225845.

[56] B.S. Chen, H.J. Uang, C.S. Tseng, Robust tracking enhancement of robot systems including motor dynamics: a fuzzy-based dynamic game approach, IEEE Transactions on Fuzzy Systems 6 (4) (1998) 538–552.

[57] G.Y. Tang, Y.D. Zhao, B.L. Zhang, Optimal output tracking control for nonlinear systems via successive approximation approach, Nonlinear Analysis: Theory, Methods & Applications 66 (6) (2007) 1365–1377. Available from: https://doi.org/10.1016/j.na.2006.01.021.

[58] Y. Zhang, L. He, C. Hu, J. Guo, J. Li, Y. Shi, General four-step discrete-time zeroing and derivative dynamics applied to time-varying nonlinear optimization, Journal of Computational and Applied Mathematics 347 (2019) 314–329. Available from: https://doi.org/10.1016/j.cam.2018.08.017.

[59] 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, Journal of Computational and Applied Mathematics 2021 Article 113582. Available from: https://doi.org/10.1016/j.cam.2021.113582.