**Thai Journal of Mathematics : 181-199**
**Special Issue: Annual Meeting in Mathematics 2018**
**http://thaijmath.in.cmu.ac.th**
Online ISSN 1686-0209

# A Projection Hestenes-Stiefel-Like Method for Monotone Nonlinear Equations with Convex Constraints

**Aliyu Muhammed Awwal[†,§], Poom Kumam[†,‡,1] Auwal Bala Abubakar[†,¶], Adamu Wakili[£]**

[†]KMUTTFixed Point Research Laboratory, Department of Mathematics,
Room SCL 802.,Fixed Point Laboratory, Science Laboratory Building,
King Faculty of Science, Mongkut's University of Technology Thonburi (KMUTT),
126 Pracha-Uthit Road, Bang Mod, Thrung Khru, Bangkok 10140, Thailand.
[‡] KMUTT-Fixed Point Theory and Applications Research Group, Theoretical and Computational Science Center (TaCS),Science Laboratory Building, Faculty of Science,
King Mongkut's University of Technology Thonburi (KMUTT),
126 Pracha-Uthit Road, Bang Mod, Thrung Khru, Bangkok 10140, Thailand.
e-mail : poom.kumam@mail.kmutt.ac.th (P. Kumam)
[§]Department of Mathematics, Faculty of Science, Gombe State University,
Gombe, Nigeria. e-mail : aliyumagsu@gmail.com (A.M. Awwal)
[¶]Department of Mathematical Sciences, Faculty of Physical Sciences, Bayero University,
Kano, Nigeria. e-mail : ababubakar.mth@buk.edu.ng (A.B. Abubakar)
[£]Department of Mathematical Sciences, Faculty of Science, Federal University,
Lokoja, Nigeria. e-mail : adamou_wakili@yahoo.com (A. Wakili)

**Abstract :** The Hestenes-Stiefel (HS) conjugate gradient (CG) method is generally regarded as one of the most efficient methods for large-scale unconstrained optimization problems. In this paper, we extend a modified Hestenes-Stiefel conjugate gradient method based on the projection technique and present a new projection method for solving nonlinear monotone equations with convex constraints. The search direction obtained satisfies the sufficient descent condition. The method can be applied to solve nonsmooth monotone problems because it is derivative free. Under appropriate assumptions, the method is shown to be glob-

[1]Corresponding author email: poom.kumam@mail.kmutt.ac.th (Poom Kumam)

ally convergent. Preliminary numerical results show that the proposed method works well and is very efficient.

**Keywords :** spectral gradient method; nonlinear monotone equations; projection method; global convergence
**2010 Mathematics Subject Classification :** 90C30; 90C06; 90C56

# 1 Introduction

Nonlinear conjugate gradient (CG) algorithms are well suited for large scale problems due to their low memory requirements as well as strong global convergence properties. Let $\mathbf{F} : \Omega \subset \mathbb{R}^n \to \mathbb{R}^n$ be continuous, monotone and nonlinear mapping, $\Omega$ is a nonempty closed and convex set and $\mathbb{R}^n$ is the $n-$dimensional Euclidean space. Monotonicity means for any $x, y \in \Omega$, we have $\langle F(x)-F(y), x-y\rangle \geq 0$. In this paper, we use conjugate gradient methods to find a vector $x^* \in \Omega$ for which

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

This problem has many applications, such as the ballistic trajectory computation and power flow equation [16, 22]. It can also be applied to some variational inequality problems which can be converted into (1.1) by means of fixed point maps or normal maps if the underlying function satisfies some coercive conditions [28]. A lot of computational methods have been proposed to solve unconstrained nonlinear monotone problem with $\Omega = \mathbb{R}^n$. Among these methods, Newton's method, the quasi-Newton methods, and their variants are very popular because of their respective local quadratic and local superlinear convergence (see in [1, 4, 5, 7, 12, 23, 29]). However, these methods are not suitable for large scale nonlinear monotone equations because they need to solve a linear system of equations using the Jacobian matrix of $F(x)$ or an approximation of the Jacobian matrix at each iteration. Among the very popular methods for solving (1.1) is the Levenberg-Marquardt type methods [20, 25] whose superlinear convergence rate can be established under an error bound estimation instead of the nonsingularity assumption.

Spectral gradient method is another efficient algorithm to solve large-scale unconstrained optimization problems,

$$\min f(x), \quad x \in \mathbb{R}^n, \tag{1.2}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a smooth nonlinear function, because of its simplicity and low storage requirements. It was proposed by Barzilai and Borwein [3] and the search direction is given as

$$d_k = -\lambda_k g(x_k), \quad d_0 = -g(x_0),$$

where $\lambda_k = \langle s_{k-1}, s_{k-1}\rangle / \langle y_{k-1}, s_{k-1}\rangle$, $y_{k-1} = g(x_k) - g(x_{k-1})$, $s_{k-1} = x_k - x_{k-1}$ and $g(x_k)$ is the gradient of $f(x_k)$. Thus some researchers have extended the spectral gradient methods to solve unconstrained nonlinear monotone equations (see in [10, 11, 13, 15, 17, 27]). Yu et al. [26] used the spectral gradient method together with the projection technique [18] to solve convex constrained nonlinear monotone equations. The method was an extension of the work in [27]. The global convergent of the method was discussed under some mild assumptions. Wang and Wang [21] proposed a modified version of the method for solving variational inequalities [19]. Theoretical analysis of the modification guarantees that the current iterate is closer to the solution set than the preceding iterate. Xiao and Zhu [24] extended the very popular CG_DESCENT method to solve monotone equations with convex constraints based on the projection techniques. Preliminary numerical results showed that the proposed method is promising. However, Liu and Li [14] observed that the CG parameter in the search direction of [24] may approach infinity if the number of iteration is sufficiently large enough. This observation may affect the numerical performance of the method. Consequently, they proposed some modifications which ensures that the CG parameter is well-defined throughout the iteration process. The numerical results reported showed that the modified method is more effective compared to CGD method in [24].

This inspired our idea to consider another modifications which we believed it will improve the numerical performance.
In this paper, we are interested in combining the projection technique with the modification of a Hestenes-Stiefel-like conjugate gradient method [2] to solve convex constrained monotone equations (1.1). Our modification improves the numerical performance and still retains the nice properties of the original method. The choice of the CG parameter $\beta_k$ in addition to the spectral gradient parameter to compute each iterate is what differentiate our method with the one in [14]. The remaining part of this paper is organized as follows. In section 2, we described the proposed algorithm. The global convergence is establish in section 3 and we report numerical experiments to show the efficiency of our method in section 4. Throughout this paper, $\|.\|$ denotes the Euclidean norm unless otherwise stated.

# 2 Proposed Algorithm

In this section, we give detail of our algorithm step by step. We use a projection operator $P_\Omega(\cdot)$ to describe our method. Let $P_\Omega(\cdot)$ be a mapping from $\mathbb{R}^n$ to $\Omega$ defined as

$$P_\Omega(x) = \text{argmin}\{\|x - y\| : y \in \Omega\}, \quad \text{for all } x \in \mathbb{R}^n.$$

An impressive property of this operator $P_\Omega(\cdot)$ is that it is nonexpansive, namely,

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

and as a result, we have

$$\|P_\Omega(x) - y\| \leq \|x - y\|, \quad \forall x, y \in \Omega.$$

In light of this, we now state the steps of the projection Hestenes-Stiefel (PHS) algorithm and discuss its nice properties. For convenience, we denote $F(x_k)$ by $F_k$.

**Algorithm 1 (PHS)**

**Step 0.** Select an initial point $x_0 \in \Omega$ and choose the constants $\rho \in (0,1)$, $\sigma, r, \xi > 0$, stopping tolerance $\varepsilon \geq 0$. Set $k = 0$.
**Step 1.** Compute $\|F_k\|$. If $\|F_k\| \leq \varepsilon$, stop.
**Step 2.** Calculate the search direction $d_k$ by

$$d_k = \begin{cases} -F_k, & \text{if } \ k = 0, \\ -\lambda_k F_k + \beta_k^{PHS} d_{k-1}, & \text{if } \ k > 0, \end{cases} \tag{2.1}$$

where

$$\beta_k^{PHS} = \max\left\{0, \frac{\langle F_k, \nu_{k-1}\rangle}{\langle w_{k-1}, d_{k-1}\rangle}\theta_k - 2\left(\frac{\|\nu_{k-1}\|\theta_k}{\langle w_{k-1}, d_{k-1}\rangle}\right)^2 \langle F_k, d_{k-1}\rangle\right\}, \tag{2.2}$$

$$\theta_k = 1 - \frac{\langle F_k, d_{k-1}\rangle^2}{\|F_k\|^2\|d_{k-1}\|^2}, \quad \nu_{k-1} = y_{k-1} + r s_{k-1}, \quad y_{k-1} = F_k - F_{k-1}, \tag{2.3}$$

$$\lambda_k = \frac{\langle s_{k-1}, s_{k-1}\rangle}{\langle \nu_{k-1}, s_{k-1}\rangle}, \ \ w_{k-1} = \nu_{k-1} + t_{k-1} d_{k-1} \ \text{ and } \ t_{k-1} = 1 + \max\left\{0, -\frac{\langle d_{k-1}, \nu_{k-1}\rangle}{\|d_{k-1}\|^2}\right\}. \tag{2.4}$$

**Step 3.** Set $z_k = x_k + \alpha_k d_k$ where the step-size $\alpha_k = \max\{\xi\rho^i : i = 0, 1, 2, \cdots\}$ such that

$$-\langle F(x_k + \alpha_k d_k), d_k\rangle \geq \sigma\alpha_k\|d_k\|^2. \tag{2.5}$$

**Step 4.** If $z_k \in \Omega$ and $\|F(z_k)\| \leq \varepsilon$, stop. Otherwise, compute the next iterate by

$$x_{k+1} = P_\Omega\left[x_k - \tau_k F(z_k)\right], \quad \text{where} \quad \tau_k = \frac{\langle F(z_k), x_k - z_k\rangle}{\|F(z_k)\|^2}. \tag{2.6}$$

**Step 5.** Set $k := k + 1$ and go to step 1.

In this article, we always assume the followings. The assumptions are very vital in proving the global convergence of our methods.

**Assumption (Ai)** The mapping $F : \Omega \to \mathbb{R}^n$ is Lipschitz continuous, i.e., there exists a positive constant $L$ such that

$$\|F(x) - F(y)\| \leq L\|x - y\|, \quad \forall x, y \in \Omega. \tag{2.7}$$

**Assumption (Aii)** The solution set of (1.1) is nonempty and is denoted by $\Gamma$.

**Remark (Ri)** It was noted that the search direction $d_k$ defined in *step 2 of Algorithm 1* is different from those in [2, 14].

**Remark (Rii)** By the monotonicity of $F$ and the definition of $\nu_{k-1}$, we have

$$\langle \nu_{k-1}, s_{k-1} \rangle = \langle y_{k-1}, s_{k-1} \rangle + r\|s_{k-1}\|^2 \geq r\|s_{k-1}\|^2 > 0. \tag{2.8}$$

On the other hand, since (2.8) holds, then by Lipschitz continuity of $F$ it holds

$$\langle \nu_{k-1}, s_{k-1} \rangle = \langle y_{k-1}, s_{k-1} \rangle + r\|s_{k-1}\|^2 \leq (L + r)\|s_{k-1}\|^2. \tag{2.9}$$

Therefore, the $\lambda_k$ defined in (2.4) is always positive for all $k \geq 0$ and satisfies

$$a \leq \lambda_k \leq b, \tag{2.10}$$

where $a := 1/(L + r)$ and $b := 1/r$.

**Remark (Riii)** By the Lipschitz continuity of $F$ and the definitions $\nu_{k-1}$, $w_{k-1}$ and $t_{k-1}$ in *step 2 of Algorithm 1*, the following inequalities hold

$$\langle w_{k-1}, d_{k-1} \rangle \geq \langle \nu_{k-1}, d_{k-1} \rangle + \|d_{k-1}\|^2 - \langle \nu_{k-1}, d_{k-1} \rangle = \|d_{k-1}\|^2 > 0. \tag{2.11}$$

$$\|\nu_{k-1}\| \leq \|F_k - F_{k-1}\| + r\|s_{k-1}\|^2 \leq (L + r)\alpha_{k-1}\|d_{k-1}\|. \tag{2.12}$$

The last inequality holds from $s_{k-1} = \alpha_{k-1}d_{k-1} = x_k - x_{k-1}$. The equations (2.10) and (2.11) show that the CG parameter $\beta_k^{PHS}$ is well-defined.

**Remark (Riv)** By Cauchy-Schwarz inequality, the $\theta_k$ defined in (2.3) satisfies $0 \leq \theta_k \leq 1$.

From the above remarks, we state the following Lemma.

**Lemma 2.1.** *Let the sequence of search directions $\{d_k\}$ be generated by Algorithm 1, then for every $k \geq 0$, there exists a positive constant $c$ such that*

$$\langle F_k, d_k \rangle \leq -c\|F_k\|^2, \quad \textit{where} \quad c = a - 1/8 \quad \textit{and} \quad a > 1/8. \tag{2.13}$$

*Proof.* If $\beta_k^{PHS} = 0$, then it clearly hold that $\langle F_k, d_k \rangle = -\lambda_k\|F_k\|^2 \leq -a\|F_k\|^2$, for all $k \geq 0$.
On the other hand, if $\beta_k^{PHS} \neq 0$, since (2.10) and (Riv) hold, then it follows from Lemma 2.1 in [14] that $\langle F_k, d_k \rangle \leq -(a - \frac{1}{8})\|F_k\|^2$, for all $k \geq 0$ and $a > 1/8$. □

# 3 Convergence Analysis

In this section, we establish the global convergence of our method.

**Lemma 3.1.** *Suppose that assumption (Ai) holds, there exists a step-size $\alpha_k$ satisfying the line search (2.5) for any $k \geq 0$.*

*Proof.* Suppose on the contrary that there exists a constant $k_0 \geq 0$ for which (2.5) does not hold, i.e.,

$$-\langle F(x_{k_0} + \xi\rho^i d_{k_0}), d_{k_0}\rangle < \sigma\xi\rho^i \|d_{k_0}\|^2, \quad \text{for any} \quad i = 0, 1, 2, \cdots .$$

Allowing $i \to \infty$, by Lipschitz continuity, we have

$$-\langle F(x_{k_0}), d_{k_0}\rangle \leq 0. \tag{3.1}$$

It follows from (2.13) that

$$-\langle F(x_{k_0}), d_{k_0}\rangle \geq c\|F(x_{k_0})\|^2 > 0. \tag{3.2}$$

Hence (3.1) and (3.2) cannot hold at the same time and the proof is complete. □

The above Lemma (3.1) indicates that *Algorithm 1* is well-defined.

**Lemma 3.2.** *[24] Suppose assumptions (Ai)-(Aii) hold. The sequences $\{x_k\}$ and $\{z_k\}$ generated by Algorithm 1 are bounded. Furthermore, we have*

$$\lim_{k\to\infty} \|x_k - z_k\| = 0; \tag{3.3}$$

$$\lim_{k\to\infty} \|x_{k+1} - x_k\| = 0. \tag{3.4}$$

From the above Lemma (3.2), we can deduce the followings

- Since $\{x_k\}$ is bounded and $F$ is Lipschitz continuous, then there exists a positive constant $\gamma$ such that

$$\|F_k\| \leq \gamma, \quad \forall\ k \geq 0. \tag{3.5}$$

- From the definition of $z_k$ and (3.3), it holds that

$$\lim_{k\to\infty} \alpha_k \|d_k\| = 0. \tag{3.6}$$

The following theorem establish the global convergence of *Algorithm 1.*

**Theorem 3.3.** *Suppose that assumptions (Ai)-(Aii) hold, and $\{x_k\}$ is the sequence generated by Algorithm 1, then*

$$\lim_{k\to\infty} \inf \|F(x_k)\| = 0. \tag{3.7}$$

*Proof.* If (3.7) does not hold, then there exists a positive constant $\epsilon$ for which

$$\|F_k\| \geq \epsilon, \quad \forall\ k \geq 0. \tag{3.8}$$

From *step 2 of Algorithm 1*, and *Remarks (Rii)-(Riv)*, we have

$$\begin{aligned}
\|d_k\| &= \left\|-\lambda_k F_k + \beta_k^{PHS} d_{k-1}\right\| \\
&\leq \lambda_k \|F_k\| + \left|\beta_k^{PHS}\right| \|d_{k-1}\| \\
&\leq b\|F_k\| + \left[\frac{|\langle F_k, \nu_{k-1}\rangle|}{\langle w_{k-1}, d_{k-1}\rangle}\theta_k + 2\left(\frac{\|\nu_{k-1}\|\theta_k}{\langle w_{k-1}, d_{k-1}\rangle}\right)^2 |\langle F_k, d_{k-1}\rangle|\right] \|d_{k-1}\| \\
&\leq b\|F_k\| + \left[\frac{\|F_k\|\|\nu_{k-1}\|}{\langle w_{k-1}, d_{k-1}\rangle} + \frac{2\|\nu_{k-1}\|^2}{\langle w_{k-1}, d_{k-1}\rangle^2}\|F_k\|\|d_{k-1}\|\right] \|d_{k-1}\| \\
&\leq b\gamma + \left[\frac{(L+r)\alpha_{k-1}\gamma\|d_{k-1}\|}{\|d_{k-1}\|^2} + \frac{2(L+r)^2\alpha_{k-1}^2\|d_{k-1}\|^2}{\|d_{k-1}\|^4}\gamma\|d_{k-1}\|\right] \|d_{k-1}\| \\
&= b\gamma + (L+r)\alpha_{k-1}\gamma + 2(L+r)^2\alpha_{k-1}^2\gamma \\
&\leq b\gamma + (L+r)\xi\gamma + 2(L+r)^2\xi^2\gamma.
\end{aligned}$$

The last inequality applies the definition of $\alpha_k$ in *step 3 of Algorithm 1.* Let $M := b\gamma + (L+r)\xi\gamma + 2(L+r)^2\xi^2\gamma$, then we have

$$\|d_k\| \leq M, \quad \forall\ k \geq 0, \tag{3.9}$$

which implies the search direction is bounded.
If $\alpha_k \neq \xi$, then by the definition of $\alpha_k$, $\rho^{-1}\alpha_k$ does not satisfy the line search (2.5), i.e.,

$$-\langle F(x_k + \rho^{-1}\alpha_k d_k), d_k\rangle < \sigma\rho^{-1}\alpha_k\|d_k\|^2.$$

Applying Cauchy-Schwarz inequality and using (2.7) and (2.13), we have

$$\begin{aligned}
c\|F_k\|^2 &\leq -\langle F_k, d_k\rangle \\
&= \langle F(x_k + \rho^{-1}\alpha_k d_k) - F_k, d_k\rangle - \langle F(x_k + \rho^{-1}\alpha_k d_k), d_k\rangle \\
&\leq L\rho^{-1}\alpha_k\|d_k\|^2 + \sigma\rho^{-1}\alpha_k\|d_k\|^2.
\end{aligned}$$

This together with (3.8) and (3.9) imply

$$\begin{aligned}
\alpha_k\|d_k\| &\geq \frac{c\rho}{(L+\sigma)} \cdot \frac{\|F_k\|^2}{\|d_k\|} \\
&\geq \frac{c\rho\epsilon^2}{(L+\sigma)M},
\end{aligned} \tag{3.10}$$

which contradicts (3.6). Thus, (3.7) holds and the proof is complete. □

# 4 Numerical Experiments

In this section, we report the results of some numerical experiments and compare the performance of the PHS method with that of the PCG method in [14]. For our PHS algorithm, the parameters were set as follows $\sigma = 0.0001$, $\rho = 0.55$, $\xi = 1$ and $r = 0.01$. The parameters in the PCG method come from [14]. All algorithms terminate whenever $\|F(x_k)\|_\infty \leq 10^{-6}$ or the number of iterations exceeds 1,000. A failure is reported and denoted by the symbol $'-'$ if any of the tested algorithms fails to satisfy the stopping criterion. All codes were written in MATLAB R2017a and run on a PC with intel Core(TM) i5-8250u processor with 4GB of RAM and CPU 1.60GHZ. We solved 6 constrained test problems (See, Appendix5.1) using 8 different initial starting points (ISP) (See, Table 1). The numerical results are reported in Tables $2-7$ for number of iterations (ITER), number of function evaluation (FEVAL), CPU time (TIME) and the norm of the residual function $F$ at the approximate solution (NORM).

The performance of the two methods was evaluated using the Dolan and Moré

Table 1: The initial points used for the test problems

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

[6] performance profiles. That is, we plotted the fraction $\rho(\tau)$ of the test problems for which each of the methods was within a factor $\tau$ of the best solver. Figures $1-3$ presented the performance profile referring to the number of iterations, the CPU time and number of function evaluations respectively. It can be observed from the Figures $1-3$ that our proposed PHS method wins higher percentage, of the numerical experiment, than the PCG method.

Numerical results listed in Tables $2-7$ show that the proposed method is efficient for solving problems (1.1). Based on the information presented in the Tables $2-7$, it could be seen that our PHS method reached the solutions (or approximate solutions) of all the test problems considered. The PCG method failed to reach to the solution of problem 2 with all the given initial guess; as well as the problems 3 and 7 with some given initial guess. Though all the failures were as a result of the number iterations exceeding 1,000. Therefore, in general, if we consider the number of wins in terms of ITER, TIME and FEVAL, our proposed PHS method performed better than the PCG method.

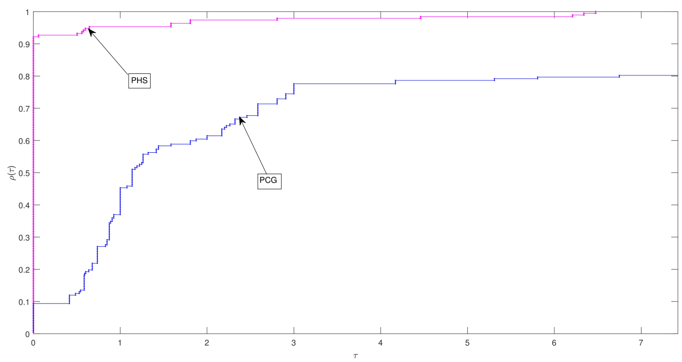


**Figure 1: Dolan and Moré performance profile with respect to number of iterations**

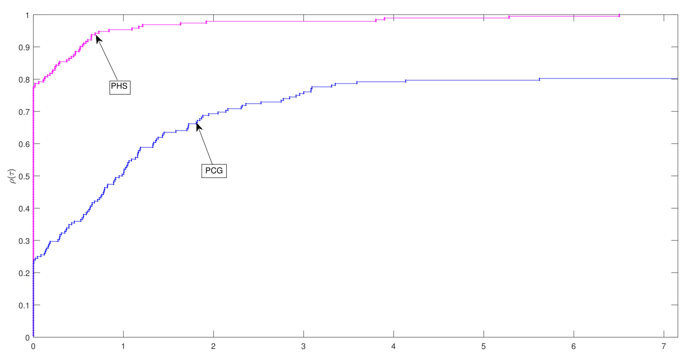


**Figure 2: Dolan and Moré performance profile with respect to CPU time**

Table 2: Experimental Results of PHS and PCG methods for problem 1

| | | PHS | | | | PCG | | | |
|---|---|---|---|---|---|---|---|---|---|
| **DIM** | **ISP** | **ITER** | **FEVAL** | **TIME** | **NORM** | **ITER** | **FEVAL** | **TIME** | **NORM** |
| 1000 | $x_1$ | 6 | 14 | 0.0045 | 7.42E-08 | 9 | 20 | 0.0056 | 1.9E-07 |
| | $x_2$ | 5 | 12 | 0.0018 | 1.75E-08 | 8 | 18 | 0.0036 | 2.7E-07 |
| | $x_3$ | 5 | 12 | 0.0060 | 5.97E-08 | 11 | 26 | 0.0134 | 1.83E-07 |
| | $x_4$ | 5 | 12 | 0.0022 | 7.52E-07 | 23 | 62 | 0.0169 | 8.6E-07 |
| | $x_5$ | 6 | 14 | 0.0061 | 1.29E-07 | 14 | 32 | 0.0075 | 6.13E-07 |
| | $x_6$ | 6 | 14 | 0.0054 | 1.25E-08 | 11 | 26 | 0.0061 | 2.18E-07 |
| | $x_7$ | 7 | 17 | 0.0087 | 5.28E-07 | 14 | 32 | 0.0041 | 6.13E-07 |
| | $x_8$ | 7 | 17 | 0.0087 | 7.48E-07 | 14 | 32 | 0.0078 | 6.12E-07 |
| 10000 | $x_1$ | 6 | 14 | 0.0544 | 2.35E-07 | 9 | 20 | 0.0144 | 6E-07 |
| | $x_2$ | 5 | 12 | 0.0234 | 5.52E-08 | 8 | 18 | 0.0101 | 8.52E-07 |
| | $x_3$ | 5 | 12 | 0.0257 | 1.89E-07 | 11 | 26 | 0.0547 | 1.83E-07 |
| | $x_4$ | 6 | 14 | 0.0255 | 2.35E-08 | 27 | 73 | 0.0842 | 9.15E-07 |
| | $x_5$ | 6 | 14 | 0.0084 | 4.08E-07 | 15 | 34 | 0.0216 | 9.14E-07 |
| | $x_6$ | 6 | 14 | 0.0276 | 3.95E-08 | 11 | 26 | 0.0304 | 2.17E-07 |
| | $x_7$ | 8 | 19 | 0.0350 | 1.65E-08 | 15 | 34 | 0.0384 | 9.14E-07 |
| | $x_8$ | 8 | 19 | 0.0400 | 2.34E-08 | 15 | 34 | 0.0267 | 9.14E-07 |
| 50000 | $x_1$ | 6 | 14 | 0.0627 | 5.25E-07 | 10 | 22 | 0.0682 | 1.44E-07 |
| | $x_2$ | 5 | 12 | 0.0380 | 1.23E-07 | 9 | 20 | 0.0764 | 2.05E-07 |
| | $x_3$ | 5 | 12 | 0.0606 | 4.22E-07 | 11 | 26 | 0.0909 | 1.83E-07 |
| | $x_4$ | 6 | 14 | 0.0577 | 5.27E-08 | 27 | 77 | 0.2537 | 4.11E-07 |
| | $x_5$ | 6 | 14 | 0.0699 | 9.12E-07 | 16 | 36 | 0.1353 | 3.82E-07 |
| | $x_6$ | 6 | 14 | 0.0748 | 8.82E-08 | 11 | 26 | 0.1161 | 2.17E-07 |
| | $x_7$ | 8 | 19 | 0.0902 | 3.7E-08 | 16 | 36 | 0.0794 | 3.82E-07 |
| | $x_8$ | 8 | 19 | 0.0924 | 5.23E-08 | 16 | 36 | 0.1215 | 3.82E-07 |
| 100000 | $x_1$ | 6 | 14 | 0.1135 | 7.42E-07 | 10 | 22 | 0.2489 | 2.04E-07 |
| | $x_2$ | 5 | 12 | 0.0783 | 1.75E-07 | 9 | 20 | 0.1466 | 2.9E-07 |
| | $x_3$ | 5 | 12 | 0.0841 | 5.97E-07 | 11 | 26 | 0.1452 | 1.83E-07 |
| | $x_4$ | 6 | 14 | 0.0766 | 7.45E-08 | 28 | 77 | 0.3792 | 5.62E-07 |
| | $x_5$ | 7 | 16 | 0.1171 | 1.28E-08 | 16 | 36 | 0.2204 | 5.4E-07 |
| | $x_6$ | 6 | 14 | 0.1024 | 1.25E-07 | 11 | 26 | 0.2028 | 2.17E-07 |
| | $x_7$ | 8 | 19 | 0.1925 | 5.23E-08 | 16 | 36 | 0.1584 | 5.4E-07 |
| | $x_8$ | 8 | 19 | 0.2452 | 7.4E-08 | 16 | 36 | 0.2110 | 5.4E-07 |

Table 3: Experimental Results of PHS and PCG methods for problem 2

| | | PHS | | | | PCG | | | |
|---|---|---|---|---|---|---|---|---|---|
| **DIM** | **ISP** | **ITER** | **FEVAL** | **TIME** | **NORM** | **ITER** | **FEVAL** | **TIME** | **NORM** |
| 1000 | $x_1$ | 71 | 144 | 0.0569 | 9.71E-07 | - | - | - | - |
| | $x_2$ | 67 | 135 | 0.0874 | 9.78E-07 | - | - | - | - |
| | $x_3$ | 68 | 137 | 0.0458 | 9.88E-07 | - | - | - | - |
| | $x_4$ | 69 | 139 | 0.0440 | 9.95E-07 | - | - | - | - |
| | $x_5$ | 70 | 142 | 0.0989 | 9.74E-07 | - | - | - | - |
| | $x_6$ | 64 | 130 | 0.1001 | 9.83E-07 | - | - | - | - |
| | $x_7$ | 64 | 130 | 0.0837 | 9.99E-07 | - | - | - | - |
| | $x_8$ | 65 | 132 | 0.0492 | 9.77E-07 | - | - | - | - |
| 10000 | $x_1$ | 115 | 232 | 0.3366 | 9.9E-07 | - | - | - | - |
| | $x_2$ | 111 | 223 | 0.3259 | 9.94E-07 | - | - | - | - |
| | $x_3$ | 113 | 227 | 0.3278 | 9.81E-07 | - | - | - | - |
| | $x_4$ | 114 | 229 | 0.3139 | 9.85E-07 | - | - | - | - |
| | $x_5$ | 114 | 230 | 0.3135 | 9.92E-07 | - | - | - | - |
| | $x_6$ | 108 | 218 | 0.2975 | 9.97E-07 | - | - | - | - |
| | $x_7$ | 109 | 220 | 0.2679 | 9.87E-07 | - | - | - | - |
| | $x_8$ | 109 | 220 | 0.2801 | 9.94E-07 | - | - | - | - |
| 50000 | $x_1$ | 165 | 332 | 1.9240 | 9.93E-07 | - | - | - | - |
| | $x_2$ | 161 | 323 | 1.8299 | 9.95E-07 | - | - | - | - |
| | $x_3$ | 162 | 325 | 1.8498 | 9.99E-07 | - | - | - | - |
| | $x_4$ | 164 | 329 | 1.8878 | 9.89E-07 | - | - | - | - |
| | $x_5$ | 164 | 330 | 1.8161 | 9.94E-07 | - | - | - | - |
| | $x_6$ | 158 | 318 | 1.7401 | 9.97E-07 | - | - | - | - |
| | $x_7$ | 159 | 320 | 1.8226 | 9.9E-07 | - | - | - | - |
| | $x_8$ | 159 | 320 | 1.8222 | 9.95E-07 | - | - | - | - |
| 100000 | $x_1$ | 193 | 388 | 4.4117 | 9.99E-07 | - | - | - | - |
| | $x_2$ | 190 | 381 | 4.3067 | 9.9E-07 | - | - | - | - |
| | $x_3$ | 191 | 383 | 4.3943 | 9.94E-07 | - | - | - | - |
| | $x_4$ | 192 | 385 | 4.2546 | 9.96E-07 | - | - | - | - |
| | $x_5$ | 193 | 388 | 4.3012 | 9.89E-07 | - | - | - | - |
| | $x_6$ | 187 | 376 | 4.2144 | 9.92E-07 | - | - | - | - |
| | $x_7$ | 187 | 376 | 4.1895 | 9.97E-07 | - | - | - | - |
| | $x_8$ | 188 | 378 | 4.2385 | 9.9E-07 | - | - | - | - |

Table 4: Experimental Results of PHS and PCG methods for problem 3

| | | PHS | | | | PCG | | | |
|---|---|---|---|---|---|---|---|---|---|
| DIM | ISP | ITER | FEVAL | TIME | NORM | ITER | FEVAL | TIME | NORM |
| 1000 | $x_1$ | 2 | 5 | 0.0024 | 0 | 9 | 19 | 0.0189 | 5.69E-07 |
| | $x_2$ | 2 | 5 | 0.0030 | 0 | 7 | 15 | 0.0061 | 2.55E-07 |
| | $x_3$ | 2 | 5 | 0.0032 | 0 | 11 | 23 | 0.0184 | 4.47E-07 |
| | $x_4$ | 2 | 5 | 0.0019 | 0 | 215 | 431 | 0.0953 | 4.67E-07 |
| | $x_5$ | 2 | 5 | 0.0027 | 0 | 14 | 29 | 0.0088 | 6.07E-07 |
| | $x_6$ | 2 | 5 | 0.0035 | 0 | 12 | 25 | 0.0129 | 8.2E-07 |
| | $x_7$ | 2 | 5 | 0.0040 | 0 | 14 | 29 | 0.0147 | 6.07E-07 |
| | $x_8$ | 2 | 5 | 0.0035 | 0 | 14 | 29 | 0.0250 | 6.08E-07 |
| 10000 | $x_1$ | 2 | 5 | 0.0169 | 0 | 10 | 21 | 0.0457 | 1.82E-07 |
| | $x_2$ | 2 | 5 | 0.0169 | 0 | 7 | 15 | 0.0557 | 7.55E-07 |
| | $x_3$ | 2 | 5 | 0.0057 | 0 | 12 | 25 | 0.0561 | 2.18E-07 |
| | $x_4$ | 2 | 5 | 0.0088 | 0 | - | - | - | - |
| | $x_5$ | 2 | 5 | 0.0085 | 0 | 15 | 31 | 0.0157 | 6.33E-07 |
| | $x_6$ | 2 | 5 | 0.0154 | 0 | 12 | 25 | 0.0387 | 3.42E-07 |
| | $x_7$ | 2 | 5 | 0.0155 | 0 | 15 | 31 | 0.0558 | 6.33E-07 |
| | $x_8$ | 2 | 5 | 0.0133 | 0 | 15 | 31 | 0.0468 | 6.33E-07 |
| 50000 | $x_1$ | 2 | 5 | 0.0191 | 0 | 10 | 21 | 0.0953 | 4.04E-07 |
| | $x_2$ | 2 | 5 | 0.0143 | 0 | 8 | 17 | 0.0429 | 1.8E-07 |
| | $x_3$ | 2 | 5 | 0.0429 | 0 | 12 | 25 | 0.0860 | 2.77E-07 |
| | $x_4$ | 2 | 5 | 0.0366 | 0 | - | - | - | - |
| | $x_5$ | 2 | 5 | 0.0418 | 0 | 16 | 33 | 0.0719 | 4.11E-07 |
| | $x_6$ | 2 | 5 | 0.0138 | 0 | 12 | 25 | 0.1042 | 3.4E-07 |
| | $x_7$ | 2 | 5 | 0.0147 | 0 | 16 | 33 | 0.1242 | 4.11E-07 |
| | $x_8$ | 2 | 5 | 0.0281 | 0 | 16 | 33 | 0.0917 | 4.11E-07 |
| 100000 | $x_1$ | 2 | 5 | 0.0786 | 0 | 10 | 21 | 0.1751 | 5.71E-07 |
| | $x_2$ | 2 | 5 | 0.0670 | 0 | 8 | 17 | 0.1099 | 2.55E-07 |
| | $x_3$ | 2 | 5 | 0.0473 | 0 | 12 | 25 | 0.2114 | 2.84E-07 |
| | $x_4$ | 2 | 5 | 0.0804 | 0 | - | - | - | - |
| | $x_5$ | 2 | 5 | 0.0736 | 0 | 16 | 33 | 0.1523 | 5.83E-07 |
| | $x_6$ | 2 | 5 | 0.0797 | 0 | 12 | 25 | 0.1159 | 3.39E-07 |
| | $x_7$ | 2 | 5 | 0.0393 | 0 | 16 | 33 | 0.2692 | 5.83E-07 |
| | $x_8$ | 2 | 5 | 0.0336 | 0 | 16 | 33 | 0.1723 | 5.83E-07 |

Table 5: Experimental Results of PHS and PCG methods for problem 4

| | | PHS | | | | PCG | | | |
|---|---|---|---|---|---|---|---|---|---|
| **DIM** | **ISP** | **ITER** | **FEVAL** | **TIME** | **NORM** | **ITER** | **FEVAL** | **TIME** | **NORM** |
| 1000 | $x_1$ | 5 | 12 | 0.0044 | 2.34E-07 | 12 | 28 | 0.0100 | 4.79E-07 |
| | $x_2$ | 5 | 12 | 0.0050 | 3.56E-07 | 12 | 28 | 0.0053 | 5.04E-07 |
| | $x_3$ | 5 | 12 | 0.0085 | 3.43E-07 | 11 | 25 | 0.0151 | 6.45E-07 |
| | $x_4$ | 5 | 12 | 0.0040 | 3.02E-07 | 24 | 57 | 0.0166 | 6.41E-07 |
| | $x_5$ | 5 | 12 | 0.0089 | 9.81E-08 | 12 | 28 | 0.0069 | 3.4E-07 |
| | $x_6$ | 5 | 12 | 0.0117 | 2.98E-08 | 11 | 25 | 0.0244 | 3.21E-07 |
| | $x_7$ | 5 | 12 | 0.0036 | 3.86E-08 | 12 | 28 | 0.0092 | 3.4E-07 |
| | $x_8$ | 5 | 12 | 0.0023 | 1.07E-07 | 12 | 28 | 0.0196 | 3.4E-07 |
| 10000 | $x_1$ | 5 | 12 | 0.0241 | 7.43E-07 | 10 | 22 | 0.0245 | 9.35E-07 |
| | $x_2$ | 6 | 14 | 0.0403 | 1.12E-08 | 11 | 25 | 0.0404 | 5.58E-07 |
| | $x_3$ | 6 | 14 | 0.0255 | 1.08E-08 | 11 | 25 | 0.0582 | 5.13E-07 |
| | $x_4$ | 5 | 12 | 0.0396 | 9.59E-07 | 26 | 61 | 0.0479 | 2.76E-07 |
| | $x_5$ | 5 | 12 | 0.0377 | 3.11E-07 | 11 | 25 | 0.0635 | 3.02E-07 |
| | $x_6$ | 5 | 12 | 0.0381 | 9.44E-08 | 11 | 25 | 0.0598 | 4.67E-07 |
| | $x_7$ | 5 | 12 | 0.0332 | 1.22E-07 | 11 | 25 | 0.0206 | 3.02E-07 |
| | $x_8$ | 5 | 12 | 0.0227 | 3.38E-07 | 11 | 25 | 0.0256 | 3.02E-07 |
| 50000 | $x_1$ | 6 | 14 | 0.1080 | 1.65E-08 | 10 | 22 | 0.0782 | 3.82E-07 |
| | $x_2$ | 6 | 14 | 0.0519 | 2.51E-08 | 10 | 22 | 0.0796 | 5.81E-07 |
| | $x_3$ | 6 | 14 | 0.1098 | 2.41E-08 | 10 | 22 | 0.0752 | 6.03E-07 |
| | $x_4$ | 6 | 14 | 0.0642 | 2.12E-08 | 22 | 51 | 0.1677 | 2.55E-07 |
| | $x_5$ | 5 | 12 | 0.0416 | 6.95E-07 | 10 | 22 | 0.1065 | 6.3E-07 |
| | $x_6$ | 5 | 12 | 0.0928 | 2.11E-07 | 10 | 22 | 0.1246 | 6.02E-07 |
| | $x_7$ | 5 | 12 | 0.0709 | 2.72E-07 | 10 | 22 | 0.0794 | 6.3E-07 |
| | $x_8$ | 5 | 12 | 0.0798 | 7.56E-07 | 10 | 22 | 0.1093 | 6.3E-07 |
| 100000 | $x_1$ | 6 | 14 | 0.2056 | 2.33E-08 | 10 | 22 | 0.2702 | 5.37E-07 |
| | $x_2$ | 6 | 14 | 0.2159 | 3.55E-08 | 10 | 22 | 0.2228 | 8.18E-07 |
| | $x_3$ | 6 | 14 | 0.2181 | 3.41E-08 | 10 | 22 | 0.4043 | 8.5E-07 |
| | $x_4$ | 6 | 14 | 0.1575 | 3E-08 | 27 | 63 | 0.6067 | 5.58E-07 |
| | $x_5$ | 5 | 12 | 0.1756 | 9.82E-07 | 10 | 22 | 0.2583 | 7.13E-07 |
| | $x_6$ | 5 | 12 | 0.1430 | 2.99E-07 | 10 | 22 | 0.2421 | 8.5E-07 |
| | $x_7$ | 5 | 12 | 0.2185 | 3.85E-07 | 10 | 22 | 0.2182 | 7.13E-07 |
| | $x_8$ | 6 | 14 | 0.1625 | 1.06E-08 | 10 | 22 | 0.2701 | 7.13E-07 |

Table 6: Experimental Results of PHS and PCG methods for problem 5

| | | PHS | | | | PCG | | | |
|---|---|---|---|---|---|---|---|---|---|
| **DIM** | **ISP** | **ITER** | **FEVAL** | **TIME** | **NORM** | **ITER** | **FEVAL** | **TIME** | **NORM** |
| 1000 | $x_1$ | 6 | 14 | 0.0043 | 5.72E-08 | 8 | 18 | 0.0039 | 2.2E-07 |
| | $x_2$ | 5 | 12 | 0.0036 | 2.57E-07 | 8 | 18 | 0.0024 | 2.07E-07 |
| | $x_3$ | 6 | 14 | 0.0058 | 1.69E-08 | 11 | 26 | 0.0041 | 2.37E-07 |
| | $x_4$ | 6 | 14 | 0.0033 | 2.34E-07 | 2 | 16 | 0.0117 | 0 |
| | $x_5$ | 6 | 15 | 0.0021 | 2.13E-07 | 16 | 38 | 0.0058 | 5.89E-07 |
| | $x_6$ | 8 | 20 | 0.0066 | 1.69E-07 | 12 | 30 | 0.0061 | 8.61E-07 |
| | $x_7$ | 9 | 23 | 0.0059 | 3.33E-07 | 16 | 38 | 0.0038 | 5.89E-07 |
| | $x_8$ | 8 | 21 | 0.0089 | 1.4E-07 | 16 | 38 | 0.0057 | 1.27E-07 |
| 10000 | $x_1$ | 6 | 14 | 0.0234 | 1.81E-07 | 8 | 18 | 0.0344 | 6.97E-07 |
| | $x_2$ | 5 | 12 | 0.0129 | 8.14E-07 | 8 | 18 | 0.0127 | 6.55E-07 |
| | $x_3$ | 6 | 14 | 0.0260 | 5.33E-08 | 11 | 26 | 0.0194 | 2.37E-07 |
| | $x_4$ | 6 | 14 | 0.0085 | 7.4E-07 | 2 | 16 | 0.0574 | 0 |
| | $x_5$ | 6 | 15 | 0.0301 | 6.75E-07 | 18 | 41 | 0.0193 | 6.89E-07 |
| | $x_6$ | 8 | 20 | 0.0365 | 5.35E-07 | 12 | 30 | 0.0268 | 8.79E-07 |
| | $x_7$ | 10 | 25 | 0.0424 | 1.04E-08 | 18 | 41 | 0.0469 | 6.89E-07 |
| | $x_8$ | 8 | 21 | 0.0346 | 4.42E-07 | 18 | 41 | 0.0592 | 7E-07 |
| 50000 | $x_1$ | 6 | 14 | 0.0595 | 4.04E-07 | 9 | 20 | 0.0571 | 1.67E-07 |
| | $x_2$ | 6 | 14 | 0.0428 | 1.8E-08 | 9 | 20 | 0.0487 | 1.57E-07 |
| | $x_3$ | 6 | 14 | 0.0487 | 1.19E-07 | 11 | 26 | 0.0422 | 2.37E-07 |
| | $x_4$ | 7 | 16 | 0.0733 | 1.64E-08 | 2 | 16 | 0.0942 | 0 |
| | $x_5$ | 7 | 17 | 0.0463 | 1.49E-08 | 19 | 43 | 0.1266 | 5.36E-07 |
| | $x_6$ | 9 | 22 | 0.0363 | 1.19E-08 | 12 | 30 | 0.0628 | 8.81E-07 |
| | $x_7$ | 10 | 25 | 0.1080 | 2.33E-08 | 19 | 43 | 0.0781 | 5.36E-07 |
| | $x_8$ | 8 | 21 | 0.0698 | 9.89E-07 | 19 | 43 | 0.1430 | 5.39E-07 |
| 100000 | $x_1$ | 6 | 14 | 0.1261 | 5.72E-07 | 9 | 20 | 0.1067 | 2.37E-07 |
| | $x_2$ | 6 | 14 | 0.1055 | 2.55E-08 | 9 | 20 | 0.1295 | 2.23E-07 |
| | $x_3$ | 6 | 14 | 0.1446 | 1.69E-07 | 11 | 26 | 0.0874 | 2.37E-07 |
| | $x_4$ | 7 | 16 | 0.1476 | 2.32E-08 | 2 | 16 | 0.1907 | 0 |
| | $x_5$ | 7 | 17 | 0.1085 | 2.11E-08 | 19 | 43 | 0.1562 | 7.71E-07 |
| | $x_6$ | 9 | 22 | 0.1520 | 1.68E-08 | 12 | 30 | 0.1062 | 8.81E-07 |
| | $x_7$ | 10 | 25 | 0.2142 | 3.3E-08 | 19 | 43 | 0.1507 | 7.71E-07 |
| | $x_8$ | 9 | 23 | 0.1512 | 1.39E-08 | 19 | 43 | 0.2290 | 7.71E-07 |

Table 7: Experimental Results of PHS and PCG methods for problem 6

| | | PHS | | | | PCG | | | |
|---|---|---|---|---|---|---|---|---|---|
| **DIM** | **ISP** | **ITER** | **FEVAL** | **TIME** | **NORM** | **ITER** | **FEVAL** | **TIME** | **NORM** |
| 1000 | $x_1$ | 3 | 19 | 0.0059 | 0 | 4 | 32 | 0.0093 | 0 |
| | $x_2$ | 79 | 327 | 0.0301 | 7.71E-07 | 120 | 533 | 0.0604 | 8.08E-07 |
| | $x_3$ | 81 | 324 | 0.0789 | 6.86E-07 | 55 | 262 | 0.0721 | 9.14E-07 |
| | $x_4$ | 74 | 294 | 0.0833 | 8.04E-07 | 1 | 3 | 0.0021 | 0 |
| | $x_5$ | 3 | 20 | 0.0100 | 0 | 119 | 552 | 0.1024 | 9.58E-07 |
| | $x_6$ | 75 | 298 | 0.0884 | 8.39E-07 | 72 | 337 | 0.0600 | 9.31E-07 |
| | $x_7$ | 85 | 340 | 0.0629 | 9.91E-07 | 119 | 552 | 0.1006 | 9.58E-07 |
| | $x_8$ | 81 | 329 | 0.0971 | 8.43E-07 | 122 | 567 | 0.0975 | 8.14E-07 |
| 10000 | $x_1$ | 3 | 19 | 0.0506 | 0 | 3 | 19 | 0.0283 | 0 |
| | $x_2$ | 81 | 331 | 0.2181 | 8.62E-07 | 118 | 527 | 0.2777 | 9.48E-07 |
| | $x_3$ | 82 | 335 | 0.2285 | 7.88E-07 | 55 | 262 | 0.1737 | 9.14E-07 |
| | $x_4$ | 81 | 327 | 0.2253 | 6.59E-07 | 1 | 3 | 0.0162 | 0 |
| | $x_5$ | 3 | 20 | 0.0272 | 0 | 168 | 928 | 0.4777 | 8.34E-07 |
| | $x_6$ | 109 | 445 | 0.4070 | 8.21E-07 | 72 | 337 | 0.1808 | 9.33E-07 |
| | $x_7$ | 84 | 308 | 0.2183 | 8.2E-07 | 168 | 928 | 0.4956 | 8.34E-07 |
| | $x_8$ | 88 | 369 | 0.2223 | 9.88E-07 | 163 | 882 | 0.4958 | 8.27E-07 |
| 50000 | $x_1$ | 3 | 19 | 0.0941 | 0 | 3 | 19 | 0.0929 | 0 |
| | $x_2$ | 81 | 332 | 0.7562 | 9.73E-07 | 126 | 563 | 1.1681 | 8.73E-07 |
| | $x_3$ | 86 | 353 | 0.8052 | 7.38E-07 | 55 | 262 | 0.6576 | 9.14E-07 |
| | $x_4$ | 89 | 363 | 1.6590 | 7.64E-07 | 1 | 3 | 0.0183 | 0 |
| | $x_5$ | 3 | 20 | 0.0980 | 0 | - | - | - | - |
| | $x_6$ | 4 | 23 | 0.0566 | 0 | 72 | 337 | 0.6821 | 9.33E-07 |
| | $x_7$ | 60 | 183 | 0.5077 | 8.25E-07 | - | - | - | - |
| | $x_8$ | 4 | 24 | 0.0578 | 0 | - | - | - | - |
| 100000 | $x_1$ | 3 | 19 | 0.1246 | 0 | 3 | 19 | 0.1046 | 0 |
| | $x_2$ | 86 | 351 | 2.4291 | 8.89E-07 | 124 | 556 | 2.9668 | 9.34E-07 |
| | $x_3$ | 78 | 313 | 1.7048 | 8.64E-07 | 55 | 262 | 1.4446 | 9.14E-07 |
| | $x_4$ | 7 | 38 | 0.2504 | 0 | 1 | 3 | 0.0807 | 0 |
| | $x_5$ | 3 | 20 | 0.1796 | 0 | 5 | 48 | 0.3172 | 0 |
| | $x_6$ | 4 | 23 | 0.2120 | 0 | 72 | 337 | 1.7967 | 9.33E-07 |
| | $x_7$ | 110 | 473 | 4.4783 | 7.25E-07 | 5 | 48 | 0.3004 | 0 |
| | $x_8$ | 4 | 24 | 0.1521 | 0 | 6 | 50 | 0.3814 | 0 |

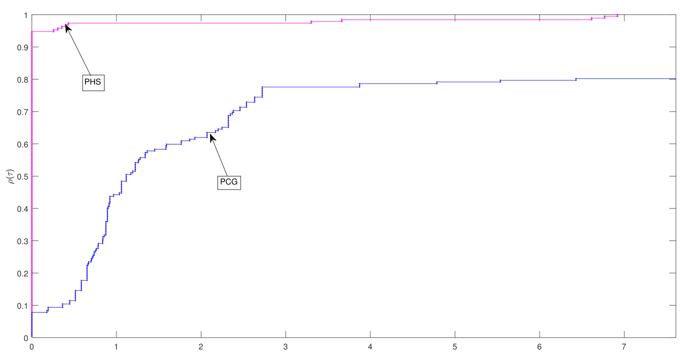


**Figure 3: Dolan and Moré performance profile with respect to number of function evaluations**

# 5 Conclusions

We proposed a projection conjugate gradient method for solving nonlinear monotone equations with convex constraints. The proposed method is suitable for large-scale monotone equations due to its low memory requirements and the global convergent was established under some suitable assumptions. The numerical results presented indicate that the proposed PHS methods effectively solved all the test problems considered using all the given initial points. The new method is competitive and performed better, than the PCG method [14] compared with, in most cases.

**Acknowledgements:** We would like to thank the referee for his comments and suggestions which provide an improvement on the manuscript. The authors acknowledge the financial support provided by King Mongkut's University of Technology Thonburi through the "KMUTT 55th Anniversary Commemorative Fund". The first author was supported by the Petchra Pra Jom Klao Doctoral Scholarship Academic for Ph.D. Program at KMUTT. This project was supported by the Theoretical and Computational Science (TaCS) Center under Computational and Applied Science for Smart Innovation Research Cluster (CLASSIC), Faculty of Science, KMUTT.

## 5.1 Appendix

In this section we list the test problems used for the numerical experiments. The mapping $F$ is taking as $F(x) = (f_1(x), f_2(x), \cdots, f_n(x))^T$, and $x = (x_1, x_2, \cdots, x_n)^T$.
**Problem 1** [8]

$$f_i(x) = 2x_i - \sin|x_i|,$$

where $\Omega = \mathbb{R}^n_+$.
**Problem 2** [8]

$$f_i(x) = \min[\min(|x_i|, x_i^2), \max(|x_i|, x_i^3)],$$

where $\Omega = \mathbb{R}^n_+$.
**Problem 3**[9]

$$f_i(x_i) = \log(|x_i| + 1) - \frac{x_i}{n},$$

where $\Omega = \mathbb{R}^n_+$.
**Problem 4** [8]

$$\begin{aligned} f_1(x) &= x_1 - e^{\cos(h(x_1+x_2))} \\ f_i(x) &= x_i - e^{\cos(h(x_{i-1}+x_i+x_{i+1}))} \\ f_n(x) &= x_n - e^{\cos(h(x_{n-1}+x_n))}, \end{aligned}$$

for $i = 2, 3, \cdots, n-1$, where $h = \frac{1}{n+1}$ and $\Omega = \mathbb{R}^n_+$.
**Problem 5** [30]

$$f_i(x) = e^{x_i} - 1,$$

where $\Omega = \mathbb{R}^n_+$.

**Problem 6**

$$f_1(x) = 2x_1 + x_2 + e^{x_1} - 1$$
$$f_i(x) = -x_{i-1} + 2x_i - x_{i+1} + e^{x_i} - 1$$
$$f_n(x) = -x_{n-1} + 2x_n + e^{x_n} - 1,$$

for $i = 2, 3, \cdots, n-1$, where $\Omega = \mathbb{R}^n_+$.

# References

[1] M. Al-Baali, E. Spedicato, and F. Maggioni. Broyden's quasi-Newton methods for a nonlinear system of equations and unconstrained optimization: a review and open problems. *Optimization Methods and Software*, 29(5):937–954, 2014.

[2] K. Amini, P. Faramarzi, and N. Pirfalah. A modified Hestenes–Stiefel conjugate gradient method with an optimal property. *Optimization Methods and Software*, pages 1–13, 2018.

[3] J. Barzilai and J. M. Borwein. Two-point step size gradient methods. *IMA Journal of Numerical Analysis*, 8(1):141–148, 1988.

[4] C. G. Broyden. A class of methods for solving nonlinear simultaneous equations. *Math Comput*, 19:577–593, 1965.

[5] A. Cordero and J. R. Torregrosa. Variants of Newton's method for functions of several variables. *Appl Math Comput*, 183:199–208, 2006.

[6] E. D. Dolan and J. J. Moré. Benchmarking optimization software with performance profiles. *Math. Program., Ser*, 91:201–213, 2002.

[7] C. T. Kelly. *Iterative Methods for Linear and Nonlinear Equations.* SIAM, 1995.

[8] W. La Cruz. A spectral algorithm for large-scale systems of nonlinear monotone equations. *Numerical Algorithms*, 76(4):1109–1130, 2017.

[9] W. La Cruz, J. M. Martínez, and M. Raydan. Spectral residual method without gradient information for solving large-scale nonlinear systems: theory and experiments. 2004.

[10] W. La Cruz, J. M. Martínez, and M Raydan. Spectral residual method without gradient information for solving large-scale nonlinear systems of equations. *Mathematics of Computation*, 75(255):1429–1448, 2006.

[11] W. La Cruz and M. Raydan. Nonmonotone spectral methods for large-scale nonlinear systems. *Optimization Methods and Software*, 18(5):583–599, 2003.

[12] Wah June Leong, Malik Abu Hassan, and Muhammad Waziri Yusuf. A matrix-free quasi-newton method for solving large-scale nonlinear systems. *Computers and Mathematics with Applications*, 62(5):2354–2363, 2011.

[13] J. Liu and S. Li. Spectral DY-type projection method for nonlinear monotone systems of equations. *Journal of Computational Mathematics*, 33(4):341–355, 2015.

[14] J. K. Liu and S. J. Li. A projection method for convex constrained monotone nonlinear equations with applications. *Computers & Mathematics with Applications*, 70(10):2442–2453, 2015.

[15] H. Mohammad and A. B. Abubakar. A positive spetral gradient-like method for large-scale nonlinear monotone equations. *Bull. Comput. Appl. Math.*, 5(1):97–113, 2017.

[16] J. M. Ortega and W. C. Rheinboldt. *Iterative solution of nonlinear equations in several variables*, volume 30. SIAM, 1970.

[17] A. Padcharoen, P. Kumam, and Y. J. Cho. Split common fixed point problems for demicontractive operators. *Numerical Algorithms*, pages 1–12, 2018.

[18] M. V. Solodov and B. F. Svaiter. A globally convergent inexact Newton method for systems of monotone equations. In *Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods*, pages 355–369. Springer, 1998.

[19] M. V. Solodov and B. F. Svaiter. A new projection method for variational inequality problems. *SIAM Journal on Control and Optimization*, 37(3):765–776, 1999.

[20] X. J. Tong and L. Qi. On the convergence of a trust-region method for solving constrained nonlinear equations with degenerate solutions. *Journal of optimization theory and applications*, 123(1):187–211, 2004.

[21] C. W. Wang and Y. J. Wang. A superlinearly convergent projection method for constrained systems of nonlinear equations. *J. Glob. Optimization*, 40:283–296, 2009.

[22] Y. Wang, L. Caccetta, and G. Zhou. Convergence analysis of a block improvement method for polynomial optimization over unit spheres. *Numerical Linear Algebra with Applications*, 22:1056–1076, 2015.

[23] M. Y. Waziri, W. J. Leong, M. A. Hassan, and M. Monsi. A new Newton's method with diagonal jacobian approximation for systems of nonlinear equations. *Journal of Mathematics and Statistics*, 6(3):246, 2010.

[24] Y. Xiao and 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):310–319, 2013.

[25] N. Yamashita and M. Fukushima. On the rate of convergence of the Levenberg-Marquardt method. In *Topics in numerical analysis*, pages 239–249. Springer, 2001.

[26] Z. Yu, J. Lin, J. Sun, Y. Xiao, L. Liu, and Z. Li. Spectral gradient projection method for monotone nonlinear equations with convex constraints. *Applied numerical mathematics*, 59(10):2416–2423, 2009.

[27] L. Zhang and W. Zhou. Spectral gradient projection method for solving nonlinear monotone equations. *Journal of Computational and Applied Mathematics*, 196(2):478–484, 2006.

[28] Y. B. Zhao and D. Li. Monotonicity of fixed point and normal mappings associated with variational inequality and its application. *SIAM Journal on Optimization*, 11(4):962–973, 2001.

[29] G. Zhou and K. C. Toh. Superlinear convergence of a Newton-type algorithm for monotone equations. *Journal of optimization theory and applications*, 125(1):205–221, 2005.

[30] W. Zhou and D. Shen. An inexact PRP conjugate gradient method for symmetric nonlinear equations. *Numer Funct Anal Optim*, 35(3):370–388, 2014.

(Received 24 August 2018)
(Accepted 22 November 2018)