Thai Journal of Mathematics : (2018) 125-147
Special Issue (ACFPTO2018) on : Advances in fixed point theory towards real world optimization problems

http://thaijmath.in.cmu.ac.th
Online ISSN 1686-0209

A New Hybrid Spectral Gradient Projection Method for Monotone System Equations with Convex Constraints

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

[†]KMUTTFixed Point Research Laboratory, Department of Mathematics, Room SCL 802, Fixed Point Laboratory, 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 : nuttapol.pakka@gmail.com (N. Pakkaranang)
[‡] 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 : Spectral gradient methods and projection technique have motivated many numerical methods for solving monotone equations. In this work, we proposed a hybrid spectral gradient algorithm for system of nonlinear monotone equations with convex constraints. The method is a combination of a convex combination of two different positive spectral parameters and the projection technique. The global convergence of the method was established under the assumptions of

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

monotonicity and Lipschitz continuity. Numerical results presented by means of comparative experiments with a similar method shows the proposed method is very efficient.

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

1 Introduction

Consider the constrained monotone nonlinear equations of the form

$$F(x) = 0, \quad x \in \Omega, \tag{1.1}$$

where $\mathbf{F} : \Omega \subset \mathbb{R}^n \to \mathbb{R}^n$ is continuous and monotone, that is $\langle F(x) - F(y), x - y \rangle \geq 0, \ \ \forall x, y \in \Omega$, Ω is a nonempty closed and convex set and $\mathbb{R}^n$ is the $n-$dimensional Euclidean space. Provided these conditions hold, the solution set of problem (1.1) is convex [21]. There are many practical applications of problem (1.1), for example monotone equations are used as subproblems in the generalized proximal algorithms with Bregman distance [9]. Some monotone variational inequality problems can be converted into system of nonlinear monotone equations by means of fixed point map or normal map [31]. Recently, monotone equations were used in signal and image recovery problems [26].

Different methods for solving unconstrained problem, where $\Omega = \mathbb{R}^n$ have been developed. Newton's method, quasi-Newton methods, and Lagrangian global method and their variants (see in [1, 4, 5, 10, 15, 25, 27, 33]) are particularly crucial because of their local quadratic and superlinear convergence property. However, they are typically unattractive for large-scale nonlinear systems of equations because they need to solve a linear system using the Jacobian matrix of $F(x)$ or an approximation of it in each iteration.

Methods for solving unconstrained minimization problems such as conjugate gradient (CG) methods, spectral gradient methods and spectral CG methods are very attractive due their low storage requirements. The main attractive feature of these methods is that the search direction do not require the computation of the Jacobian matrix and therefore a low computational effort per iteration is required. Thus, motivated by the projection technique proposed by Solodov and Svaiter [23], some researchers extended these methods to solve large-scale nonlinear equations and also unconstrained nonlinear monotone equations (see, [7, 13, 14, 17, 18, 20, 22, 30], for details).

In recent time, how to find the solution of the constrained monotone equations (1.1) has received much attentions. For example, Wang et al. [24] proposed method for solving nonlinear monotone equations with convex constraints. The proposed method is global convergent and numerically robust and effective. Motivated by the projection technique [23], Yu et al. [28] extended the work of Zhang and Zhou [30] to solve monotone nonlinear equations with convex constraints. A relevant property of the method is that computation of the iteration sequence does not require the solution of any subproblem. The global convergent of the method was discussed under some mild assumptions. Another algorithm for solving convex constrained nonlinear monotone equations was proposed by Zheng [32]. The algorithm was a combination of proximal point and projection methodology which was introduced based on Armijo-type line search procedure. In [18], Liu et al. discussed two frameworks of some sufficient descent CG methods. Combined with the hyperplane projection technique, they applied the methods to solve convex constrained nonlinear monotone equations. Numerical results showed the two methods are efficient. Again, based on the spectral gradient parameter and the projection scheme, Yuan [29] extended the well-known CGD method to solve nonlinear monotone equations with convex constraints. Preliminary numerical results showed the proposed method worked well. Very recently, Liu and Feng [16] proposed a derivative free projection method to solve convex constrained monotone nonlinear equations. The method was a modification of the DY CG method [6]. The modification improved the numerical performance of the DY method and the global convergence was also established.

Inspired the by the contributions of Yu et al. [28] and Mohammad and Abubakar [20] together with the projection technique [23], we present a hybrid spectral gradient (HSG) method to solve system of monotone nonlinear equations with convex constraints. To do so, we take a convex combination of the spectral gradient parameter in [3] and the positive spectral coefficient [7] to define the search direction of our algorithm. The direction satisfies the sufficient descent condition and the global convergence theorem under suitable assumptions. We present some preliminary numerical experiments to illustrate the efficiency of the proposed method.

The remaining part of this paper is organized as follows. In section 2, we described the proposed method and its algorithm. The global convergence is established in section 3 and we report numerical experiments in section 4.

2 Motivation and proposed method

In this section, we first give some preliminaries . Throughout this paper, $\|\cdot\|$ denotes the 2−norm. Let $\Omega \subset \mathbb{R}^n$ be a nonempty closed and convex set. The projection operator is a map $P_\Omega : \mathbb{R}^n \to \Omega$, which is defined as

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

An interesting property is that this operator $P_\Omega(\cdot)$ is nonexpansive, that is,

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

Consequently, we have

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

Next, we recall the Barzilai and Borwein [3] spectral gradient method for unconstrained optimization problem $\min f(x), \quad x \in \mathbb{R}^n$, where $f : \mathbb{R}^n \to \mathbb{R}$ is a smooth nonlinear function. Starting from a given initial guess, the method generates sequence of iterates $\{x_k\}$ using the following iterative formula:

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

where $\alpha_k > 0$ is the step length obtained via some suitable line search and the search direction d_k is defined by

$$d_k = \begin{cases} -g_k, & \text{if } k = 0, \\ -\lambda_k g_k, & \text{if } k > 0, \end{cases} \tag{2.3}$$

where

$$\lambda_k = \frac{\langle s_{k-1}, s_{k-1}\rangle}{\langle y_{k-1}, s_{k-1}\rangle},$$

$y_{k-1} = g_k - g_{k-1}$, $s_{k-1} = x_k - x_{k-1}$ and g_k is the gradient of f at x_k.
The gradient g is a function from $\mathbb{R}^n$ to $\mathbb{R}^n$ which can be viewed as our function F. However, one disadvantage of this method is that λ_k may be negative for nonconvex functions. Dai et al. [7] proposed a remedy by adopting a positive spectral gradient parameter $\gamma_k = \frac{\|s_{k-1}\|}{\|y_{k-1}\|}$ and used the approach to solve symmetric linear system of equations.
On the other hand, Amini et al. [2] proposed a modified CG method to for unconstrained optimization and the CG parameter was given by

$$\beta_k = \frac{\langle g_k, y_{k-1}\rangle}{\langle y_{k-1}, d_{k-1}\rangle}\theta_k - \gamma\left(\frac{\|y_{k-1}\|\theta_k}{\langle y_{k-1}, d_{k-1}\rangle}\right)^2 \langle g_k, d_{k-1}\rangle, \tag{2.4}$$

where

$$\theta_k = 1 - \langle g_k, d_{k-1}\rangle^2 / \|g_k\|^2 \|d_{k-1}\|^2,$$

$\gamma > 1/4$. We are interested in the parameter θ_k because by Cauchy-Schwatz inequality, we have $|\langle g_k, d_{k-1}\rangle| \leq \|g_k\|\|d_{k-1}\|$. This means that θ_k is a sequence in the interval $[0, 1]$. That is, $0 \leq \theta_k \leq 1 \ \forall \ k \geq 0$. Therefore we can use θ_k for our convex combination.
For the purpose of this paper, we need the following assumptions

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

$$\|F(x) - F(y)\| \leq L\|x - y\|, \quad \forall x, y \in \mathbb{R}^n. \tag{2.5}$$

(Aii). The solution set of (1.1) is nonempty and is denoted by Λ.

We now formally state the steps of our algorithm as follows:

Algorithm 1 (HSG)

Step 0. Given $x_0 \in \Omega$, $\rho \in (0,1)$, $\sigma, \kappa > 0$, $r > 0$ stopping tolerance $\epsilon \geq 0$. Set $k = 0$.
Step 1. Compute F_k. If $\|F_k\| \leq \epsilon$, stop.
Step 2. Compute $d_k = -\tau_k F(x_k)$, $d_0 = -F(x_0)$, where

$$\tau_k = (1-\theta_k)\lambda_k + \theta_k \gamma_k, \quad \theta_k = 1 - \frac{\langle F(x_k), d_{k-1}\rangle^2}{\|F(x_k)\|^2 \|d_{k-1}\|^2},$$

$$\lambda_k = \frac{\langle s_{k-1}, s_{k-1}\rangle}{\langle \nu_{k-1}, s_{k-1}\rangle}, \;\; \gamma_k = \frac{\|s_{k-1}\|}{\|\nu_{k-1}\|}, \;\; \nu_{k-1} = y_{k-1} + r s_{k-1}, \;\; \text{and} \;\; y_{k-1} = F(x_k) - F(x_{k-1}).$$

Step 3. Determine $\alpha_k = \max\{\kappa\rho^i : i = 0, 1, 2, \cdots\}$ such that

$$-\langle F(x_k + \alpha_k d_k), d_k\rangle \geq \sigma \alpha_k \|d_k\|^2. \tag{2.6}$$

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

$$x_{k+1} = P_\Omega\left[x_k - \mu_k F(z_k)\right], \quad \text{where} \quad \mu_k = \frac{\langle F(z_k), x_k - z_k\rangle}{\|F(z_k)\|^2}. \tag{2.7}$$

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

Remark 2.1. *By the definition of ν_{k-1} and the monotonicity of F, we have*

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

This implies λ_k is positive for all k. Moreover, by the assumption (Ai), we have

$$\|\nu_{k-1}\| = \|F_k - F_{k-1} + r s_{k-1}\| \leq (L+r)\|s_{k-1}\|, \tag{2.9}$$

which means $\gamma_k = \frac{\|s_{k-1}\|}{\|\nu_{k-1}\|} \geq \frac{1}{L+r} > 0$. Therefore, the spectral gradient τ_k in step 2 of Algorithm 1 is strictly positive for all $k = 0, 1, 2, \cdots$.

Remark 2.2. *Since $\tau_k > 0$, $\forall\ k$, it is not difficult to see that the search direction d_k defined in step 2 of Algorithm 1 satisfies the sufficient descent property $\langle F(x_k), d_k\rangle \leq -c\|F(x_k)\|^2$, for all k and $c > 0$.*

Remark 2.3. *The search direction defined in step 2 of Algorithm 1 is different from the one in [20]. The main difference was the choice of the parameter θ_k. Moreover, the algorithm in [20] was used to solve unconstrained monotone equations while our algorithm was used for monotone equations with convex constraints.*

3 Convergence Analysis

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

Lemma 3.1. *Suppose the sequence $\{x_k\}$ and $\{z_k\}$ are generated by Algorithm 1. Let the assumptions (Ai)-(Aii) hold, then the sequences $\{x_k\}$, and $\{z_k\}$ are bounded. In addition, we have*

$$\lim_{k\to\infty} \|x_k - z_k\| = 0, \tag{3.1}$$

and

$$\lim_{k\to\infty} \|x_{k+1} - x_k\| = 0. \tag{3.2}$$

Proof. For any $\overline{x} \in \Lambda$, using the monotonicity of F, we have

$$\langle F(z_k), x_k - \overline{x}\rangle \geq \langle F(z_k), x_k - z_k\rangle. \tag{3.3}$$

By the definition of z_k and the line search (2.6) it follows that

$$\langle F(z_k), x_k - z_k\rangle \geq \sigma\alpha_k^2\|d_k\|^2 \geq 0. \tag{3.4}$$

Now, from the nonexpansiveness of the projection operator $P_\Omega(\cdot)$, it holds

$$\begin{aligned}
\|x_{k-1} - \overline{x}\|^2 &= \|P_\Omega[x_k - \mu_k F(z_k)] - \overline{x}\|^2 \\
&\leq \|x_k - \mu_k F(z_k) - \overline{x}\|^2 \\
&= \|x_k - \overline{x}\|^2 - 2\mu_k\langle F(z_k), x_k - \overline{x}\rangle + \mu_k^2\|F(z_k)\|^2 \\
&\leq \|x_k - \overline{x}\|^2 - 2\mu_k\langle F(z_k), x_k - z_k\rangle + \mu_k^2\|F(z_k)\|^2 \\
&= \|x_k - \overline{x}\|^2 - \left(\frac{\langle F(z_k), x_k - z_k\rangle}{\|F(z_k)\|}\right)^2 \\
&\leq \|x_k - \overline{x}\|^2.
\end{aligned} \tag{3.5}$$

We can easily see that the sequence $\{\|x_k - \overline{x}\|\}$ is decreasing and convergent which implies that the sequence $\{x_k\}$ is bounded. From (3.5), it holds that

$$\|x_k - \overline{x}\| \leq \|x_0 - \overline{x}\|, \quad \forall\, k \geq 0. \tag{3.6}$$

Therefore, by assumption (Ai), we have

$$\|F(x_k)\| = \|F(x_k) - F(\overline{x})\| \leq L\|x_k - \overline{x}\| \leq L\|x_0 - \overline{x}\| = \omega. \tag{3.7}$$

Using the definition of z_k and the monotonicity of F together with equation (3.4), we have

$$\sigma\|x_k - z_k\| = \frac{\sigma\|\alpha_k d_k\|^2}{\|x_k - z_k\|} \leq \frac{\langle F(z_k), x_k - z_k\rangle}{\|x_k - z_k\|} \leq \frac{\langle F(x_k), x_k - z_k\rangle}{\|x_k - z_k\|} \leq \|F(x_k)\|, \tag{3.8}$$

where the last inequality follows from the Cauchy-Schwarz inequality. Hence, from the boundedness of $\{x_k\}$ and equations (3.7)-(3.8), the sequence $\{z_k\}$ is bounded. Since the sequence $\{z_k\}$ is bounded, it follows that for any $\overline{x} \in \Omega$, the sequence $\{\|z_k - \overline{x}\|\}$ is also bounded, that is, there exists $\vartheta > 0$ such that $\|z_k - \overline{x}\| \leq \vartheta$. Combining it together with (2.5), we have

$$\|F(z_k)\| = \|F(z_k) - F(\overline{x})\| \leq L\|z_k - \overline{x}\| \leq L\vartheta.$$

Then it follows from (3.5) that

$$\frac{\sigma^2}{(L\vartheta)^2}\sum_{k=0}^{\infty}\|x_k - z_k\|^4 \leq \sum_{k=0}^{\infty}\left(\frac{\langle F(z_k), x_k - z_k\rangle}{\|F(z_k)\|}\right)^2 \leq \sum_{k=0}^{\infty}(\|x_k-\overline{x}\|^2 - \|x_{k+1}-\overline{x}\|^2) < +\infty.$$

By the property of convergence series, it implies $\lim_{k\to\infty}\|x_k - z_k\| = 0$.
Then using (2.1), the definition of μ_k and Cauchy-Schwarz inequality, we have

$$\begin{aligned}\|x_{k+1} - x_k\| &= \|P_\Omega[x_k - \mu_k F(z_k)] - x_k\| \\ &\leq \|x_k - \mu_k F(z_k) - x_k\| \\ &= \|\mu_k F(z_k)\| \\ &\leq \|x_k - z_k\|,\end{aligned} \tag{3.9}$$

which implies

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

□

The following lemma is proved in a similar way as in [20].

Lemma 3.2. *Suppose assumption (Ai) holds. Let $\{d_k\}$ be the sequence of directions generated by Algorithm 1, then there exists a positive constant M such that $\|d_k\| \leq M$ for all $k = 0, 1, 2, \cdots$.*

Proof. Since $z_k = x_k + \alpha_k d_k$, it follows from lemma (3.1) that

$$\lim_{k\to\infty}\alpha_k\|d_k\| = \lim_{k\to\infty}\|x_k - z_k\| = 0. \tag{3.10}$$

From equation (3.7) and *step 2* of *Algorithm 1*, we have

$$\|d_0\| = \|F(x_0)\| \leq \omega,$$

and

$$\begin{aligned}\|d_k\| &= |\tau_k|\|F(x_k)\| \\ &\leq |\tau_k|\omega.\end{aligned}$$

Equation (3.10) implies, there exists a positive integer k_0 such that $\alpha_{k-1}\|d_{k-1}\| \leq \epsilon_0, \forall\, k > k_0$, for an arbitrary constant ϵ_0. Taking $M := \max\{\|d_0\|, \|d_1\|, \cdots, \|d_{k_0}\|, |\tau_k|\omega\}$, we have

$$\|d_k\| \leq M, \quad \forall \;\; k = 0, 1, 2, \cdots. \tag{3.11}$$

□

The following theorem establish the global convergence of Algorithm 1.

Theorem 3.3. *Let $\{x_k\}$ and $\{z_k\}$ be sequences generated by Algorithm 1. Then*

$$\lim_{k\to\infty} \inf \|F(x_k)\| = 0. \tag{3.12}$$

Proof. Suppose $\liminf_{k\to\infty} \|d_k\| = 0$, then by the sufficient descent property, we have $\liminf_{k\to\infty} \|F(x_k)\| = 0$. Since F is continuous and the sequence $\{x_k\}$ is bounded, then there exists some accumulation point $\tilde{x}$ such that $F(\tilde{x}) = 0$. Since $\{\|x_k - \tilde{x}\|\}$ converges and $\tilde{x}$ is an accumulation point of the sequence $\{x_k\}$, it follows that $\{x_k\}$ converges to $\tilde{x}$.
If $\liminf_{k\to\infty} \|d_k\| > 0$, we have $\liminf_{k\to\infty} \|F(x_k)\| > 0$. From (3.10), we have $\lim_{k\to\infty} \alpha_k = 0$. From the line search defined by equation (2.6), for the step-size $\rho^{-1}\alpha_k$, we have

$$-\langle F(x_k + \rho^{-1}\alpha_k d_k), d_k\rangle < \sigma\rho^{-1}\alpha_k\|d_k\|^2. \tag{3.13}$$

By the boundedness of $\{x_k\}$, there exists an accumulation point $\tilde{x}$ and an infinite index set K such that $\lim_{k\to\infty} x_k = \tilde{x}$, for $k \in K$.
Again, because of the boundedness of $\{d_k\}$, there also exists an infinite set $K_1 \subset K$ and an accumulation point $\tilde{d}$ such that $\lim_{k\to\infty} d_k = \tilde{d}$, for $k \in K_1$.
Therefore, by taking limit as $k \to \infty$ on both sides of (3.13) for $k \in K_1$ results to

$$\langle F(\tilde{x}), \tilde{d}\rangle > 0. \tag{3.14}$$

On the other hand, allowing $k \to \infty$ on both sides of (2.6) for $k \in K_1$, we have

$$\langle F(\tilde{x}), \tilde{d}\rangle \leq 0. \tag{3.15}$$

It is clear that inequalities (3.14) and (3.15) cannot hold concurrently. Therefore, it is impossible to have $\lim_{k\to\infty} \inf \|F(x_k)\| > 0$ and this completes the proof. □

4 Numerical Experiments

This section gives a performance comparison between our proposed algorithm HSG and the method proposed by Yu et al. [28] (for simplicity, we denote the method by (SGP)) for solving convex constrained nonlinear monotone equations (1.1). We set the following parameters for the implementation of HSG algorithm $r = \sigma = 0.001$, $\kappa = 1$ and $\rho = 0.9$ except for problem 8 where we choose $\rho = 0.7$. The parameters in SGP algorithm are chosen as in [28]. 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 8 constrained test problems with 8 different initial starting points (ISP) (See table 1). We used 4 different dimensions (DIM) which are 5,000, 10,000, 50,000 and 100,000. The iteration is terminated whenever the inequality $\|F(x_k)\| \leq 10^{-6}$ or $\|F(z_k)\| \leq 10^{-6}$ is satisfied.

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	$(\frac{1}{n}, \frac{2}{n}, \frac{3}{n}, \cdots, 1)^T$

In Tables 2-9, we present the results of the following information: the number of iterations (ITER) needed to converge to an approximate solution, the CPU time in seconds (TIME), the number of functions evaluation (FEVAL) and the norm of the objective function F at the approximate solution x^* (NORM). The symbol $'-'$ indicates the failure of a method when the number of iterations exceeds 1,000 and no solution is reached.

We use the following test problems where $F(x) = (f_1(x), f_2(x), \cdots, f_n(x))^T$, and $x = (x_1, x_2, \cdots, x_n)^T$:

Problem 1[12]

$$f_1(x) = e^{x_1} - 1$$
$$f_i(x) = e^{x_i} + x_{i-1} - 1,$$

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

Problem 2[12]

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

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

Problem 3 [11]

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

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

Problem 4 [11]

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

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

Problem 5 [34]

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

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

Problem 6 [19]

$$f_1(x) = hx_1 + x_2 - 1$$
$$f_i(x) = x_{i-1} + hx_i + x_{i+1} - 1$$
$$f_n(x) = x_{n-1} + hx_n - 1,$$

where $h = 2.5$ *and* $\Omega = \mathbb{R}^n_+$.
Problem 7 [11]

$$
\begin{aligned}
f_1(x) &= x_1 - e^{\cos(h(x_1+x_2))} \\
f_i(x) &= x_i - e^{\cos(h(x_{i-1}+x_i+x_{i+1}))} \text{ for } i = 2, 3, ..., n-1 \\
f_n(x) &= x_n - e^{\cos(h(x_{n-1}+x_n))},
\end{aligned}
$$

where $h = \frac{1}{n+1}$ and $\Omega = \mathbb{R}^n_+$.
Problem 8

$$
\begin{aligned}
f_1(x) &= 2x_1 + x_2 + e^{x_1} - 1 \\
f_i(x) &= -x_{i-1} + 2x_i - x_{i+1} + e^{x_i} - 1 \\
f_n(x) &= -x_{n-1} + 2x_n + e^{x_n} - 1.
\end{aligned}
$$

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

The Dolan and Moré [8] performance profiles was adopted to compare the performance of the proposed HSG method and SGP method (See figures 1-3). The comparison was based on the followings: the number of iterations; number of functions evaluation; and CPU time. From the graphs in figures 1-3, it can be observed that all the curves with respect to our proposed method stays longer on the vertical axis. This means that the percentage won by our algorithm in the numerical experiment is greater than that of SGP algorithm. To be specific, the graphs showed our method won about 75%, 59% and 56% of the experiments in terms of number of iterations, number of function evaluations and CPU time(s) respectively. Moreover, from tables 2-9, it can be seen that in most of the experiments, our method HSG recorded less number of iterations, the number of function evaluation and CPU time(s) to reach a solution compared to SGP method. It is worth noting that, our method HSG converges to exact solutions of problems 2 and 5 with some of the initial starting point (ISP), this can be seen from table 3 and 6.

Table 2: Numerical Comparison of HSG and SGP methods for problem 1

		HSG				SGP			
DIM	ISP	ITER	FEVAL	TIME	NORM	ITER	FEVAL	TIME	NORM
1000	x_1	9	36	0.80881	3.53E-07	470	1410	0.112033	9.84E-07
	x_2	8	32	0.027333	2.07E-07	453	1360	0.139607	9.83E-07
	x_3	8	31	0.028298	2.62E-07	443	1330	0.14377	9.98E-07
	x_4	9	35	0.015175	1.25E-07	523	1568	0.111548	9.92E-07
	x_5	114	1017	0.1063	9.34E-07	465	1395	0.154678	9.85E-07
	x_6	86	746	0.090239	8.62E-07	521	1562	0.144145	9.99E-07
	x_7	86	744	0.091275	8.63E-07	466	1399	0.133845	9.98E-07
	x_8	87	745	0.067804	8.7E-07	529	1586	0.157304	9.48E-07
10000	x_1	9	36	0.055203	3.79E-07	479	1436	0.61838	9.99E-07
	x_2	8	32	0.017467	3.3E-07	471	1413	0.558702	9.86E-07
	x_3	8	31	0.051548	2.42E-07	467	1401	0.570495	9.94E-07
	x_4	9	35	0.091194	1.85E-07	509	1526	0.591647	9.91E-07
	x_5	114	1017	0.355824	9.88E-07	534	1601	0.621983	9.81E-07
	x_6	86	746	0.285963	8.62E-07	529	1586	0.619876	9.92E-07
	x_7	86	744	0.268787	8.63E-07	520	1559	0.604896	9.83E-07
	x_8	87	745	0.287959	8.7E-07	511	1532	0.578868	9.81E-07
50000	x_1	9	36	0.132714	7.59E-07	526	1577	2.388438	9.99E-07
	x_2	8	32	0.112479	6.92E-07	474	1422	2.171568	9.93E-07
	x_3	8	31	0.101503	4.74E-07	514	1542	2.407779	9.96E-07
	x_4	9	35	0.103286	3.85E-07	504	1511	2.312448	9.83E-07
	x_5	114	1017	1.542272	9.89E-07	555	1663	2.576463	9.82E-07
	x_6	86	746	1.072495	8.62E-07	554	1660	2.52902	9.96E-07
	x_7	86	744	1.09343	8.63E-07	530	1589	2.404612	9.97E-07
	x_8	87	745	1.06465	8.7E-07	535	1604	2.471612	9.88E-07
100000	x_1	10	39	0.269472	1.06E-07	543	1627	5.943249	9.88E-07
	x_2	8	32	0.179234	9.7E-07	472	1416	5.245323	9.99E-07
	x_3	8	31	0.176154	6.59E-07	529	1587	5.663519	9.97E-07
	x_4	9	35	0.20924	5.39E-07	511	1532	5.636506	9.83E-07
	x_5	114	1017	3.773009	9.89E-07	557	1669	6.012137	9.89E-07
	x_6	86	746	2.666463	8.62E-07	558	1672	6.222271	9.91E-07
	x_7	86	744	2.640047	8.63E-07	530	1589	5.87635	9.89E-07
	x_8	87	745	2.672406	8.7E-07	552	1654	6.170925	9.94E-07

Table 3: Numerical Comparison of HSG and SGP methods for problem 2

		HSG				SGP			
DIM	**ISP**	**ITER**	**FEVAL**	**TIME**	**NORM**	**ITER**	**FEVAL**	**TIME**	**NORM**
1000	x_1	2	5	0.066364	0	2	5	0.002165	0
	x_2	7	23	0.01322	6.62E-07	2	5	0.003532	0
	x_3	8	27	0.013886	2.32E-07	2	5	0.003295	0
	x_4	2	5	0.003453	0	2	5	0.001926	0
	x_5	2	5	0.001373	0	2	5	0.004897	0
	x_6	2	5	0.004464	0	2	5	0.003875	0
	x_7	2	5	0.00149	0	2	5	0.003558	0
	x_8	2	5	0.003822	0	2	5	0.003515	0
10000	x_1	2	5	0.015688	0	2	5	0.008306	0
	x_2	8	26	0.051861	2.1E-07	2	5	0.014822	0
	x_3	8	27	0.04483	7.32E-07	2	5	0.012704	0
	x_4	2	5	0.011583	0	2	5	0.0125	0
	x_5	2	5	0.005028	0	2	5	0.012828	0
	x_6	2	5	0.010684	0	2	5	0.013202	0
	x_7	2	5	0.020179	0	2	5	0.012223	0
	x_8	2	5	0.014288	0	2	5	0.012893	0
50000	x_1	2	5	0.039591	0	2	5	0.036832	0
	x_2	8	26	0.10289	4.7E-07	2	5	0.027083	0
	x_3	9	30	0.099963	1.65E-07	2	5	0.043481	0
	x_4	2	5	0.040087	0	2	5	0.036792	0
	x_5	2	5	0.041857	0	2	5	0.036762	0
	x_6	2	5	0.044677	0	2	5	0.036646	0
	x_7	2	5	0.042205	0	2	5	0.038985	0
	x_8	2	5	0.043862	0	2	5	0.04021	0
100000	x_1	2	5	0.058072	0	2	5	0.06599	0
	x_2	8	26	0.135272	6.65E-07	2	5	0.06366	0
	x_3	9	30	0.277955	2.34E-07	2	5	0.067043	0
	x_4	2	5	0.037094	0	2	5	0.059717	0
	x_5	2	5	0.04062	0	2	5	0.070502	0
	x_6	2	5	0.041596	0	2	5	0.067963	0
	x_7	2	5	0.054348	0	2	5	0.065861	0
	x_8	2	5	0.062998	0	2	5	0.047998	0

Table 4: Numerical Comparison of HSG and SGP methods for problem 3

		HSG				SGP			
DIM	ISP	ITER	FEVAL	TIME	NORM	ITER	FEVAL	TIME	NORM
1000	x_1	8	26	0.044326	8.37E-07	6	14	0.003597	1.12E-07
	x_2	7	22	0.004395	3.34E-07	5	12	0.00493	1.97E-08
	x_3	7	22	0.004274	6.68E-07	5	12	0.00525	6.94E-08
	x_4	8	26	0.00947	3.74E-07	5	12	0.00878	9.66E-07
	x_5	9	32	0.012987	5.99E-07	6	14	0.007826	4.62E-07
	x_6	9	33	0.004161	8.52E-07	6	14	0.007127	1.7E-07
	x_7	10	37	0.014264	1.35E-07	6	14	0.008164	1.22E-08
	x_8	9	34	0.015282	8.04E-07	8	19	0.012343	2.82E-08
10000	x_1	9	29	0.028185	2.67E-07	6	14	0.030145	3.55E-07
	x_2	8	25	0.031928	1.07E-07	5	12	0.022725	6.23E-08
	x_3	8	25	0.049419	2.13E-07	5	12	0.024885	2.2E-07
	x_4	9	29	0.017637	1.19E-07	6	14	0.027459	3.02E-08
	x_5	10	35	0.051295	1.91E-07	7	16	0.031669	1.45E-08
	x_6	10	36	0.026258	2.72E-07	6	14	0.028225	5.36E-07
	x_7	10	37	0.043225	4.27E-07	6	14	0.031406	3.85E-08
	x_8	10	37	0.057086	2.57E-07	8	19	0.035062	8.91E-08
50000	x_1	9	29	0.107741	5.97E-07	6	14	0.065336	7.95E-07
	x_2	8	25	0.07691	2.38E-07	5	12	0.043543	1.39E-07
	x_3	8	25	0.098828	4.77E-07	5	12	0.06823	4.91E-07
	x_4	9	29	0.087619	2.67E-07	6	14	0.075736	6.76E-08
	x_5	10	35	0.104122	4.27E-07	7	16	0.08076	3.24E-08
	x_6	10	36	0.110021	6.08E-07	7	16	0.079123	1.19E-08
	x_7	10	37	0.094977	9.56E-07	6	14	0.051578	8.61E-08
	x_8	10	37	0.148641	5.74E-07	8	19	0.084988	1.99E-07
100000	x_1	9	29	0.18666	8.44E-07	7	16	0.127112	1.11E-08
	x_2	8	25	0.151972	3.37E-07	5	12	0.10093	1.97E-07
	x_3	8	25	0.139031	6.74E-07	5	12	0.087968	6.94E-07
	x_4	9	29	0.165162	3.78E-07	6	14	0.113096	9.56E-08
	x_5	10	35	0.202669	6.04E-07	7	16	0.105682	4.58E-08
	x_6	10	36	0.206742	8.6E-07	7	16	0.071571	1.68E-08
	x_7	11	40	0.205453	1.36E-07	6	14	0.113583	1.22E-07
	x_8	10	37	0.209855	8.11E-07	8	19	0.120498	2.82E-07

Table 5: Numerical Comparison of HSG and SGP methods for problem 4

		HSG				SGP			
DIM	**ISP**	**ITER**	**FEVAL**	**TIME**	**NORM**	**ITER**	**FEVAL**	**TIME**	**NORM**
1000	x_1	56	663	0.258644	9.92E-07	71	144	0.049273	9.77E-07
	x_2	54	665	0.124756	9.4E-07	67	135	0.046013	9.78E-07
	x_3	55	665	0.195229	9.35E-07	68	137	0.094601	9.88E-07
	x_4	56	664	0.18581	9.49E-07	69	139	0.104496	9.95E-07
	x_5	57	666	0.186467	9.73E-07	72	147	0.086641	9.77E-07
	x_6	57	667	0.223628	9.59E-07	72	147	0.092224	9.82E-07
	x_7	57	666	0.225285	9.73E-07	72	147	0.097597	9.78E-07
	x_8	57	667	0.199422	9.7E-07	71	145	0.078072	9.86E-07
10000	x_1	74	897	0.765182	9.52E-07	115	232	0.32434	9.94E-07
	x_2	71	886	0.799422	9.7E-07	111	223	0.315119	9.94E-07
	x_3	72	886	0.758446	9.67E-07	113	227	0.316302	9.81E-07
	x_4	73	885	0.753975	9.76E-07	114	229	0.322942	9.85E-07
	x_5	74	887	0.773131	9.91E-07	116	235	0.356975	9.94E-07
	x_6	74	888	0.75477	9.82E-07	116	235	0.311852	9.97E-07
	x_7	74	887	0.792405	9.91E-07	116	235	0.349416	9.94E-07
	x_8	74	888	0.82702	9.89E-07	115	233	0.305334	9.99E-07
50000	x_1	92	1131	4.187495	9.66E-07	165	332	1.81074	9.95E-07
	x_2	89	1120	4.173089	9.79E-07	161	323	1.743016	9.95E-07
	x_3	90	1120	4.165075	9.76E-07	162	325	1.742095	9.99E-07
	x_4	91	1119	4.19101	9.83E-07	164	329	1.756766	9.89E-07
	x_5	92	1121	4.215128	9.94E-07	166	335	1.804617	9.95E-07
	x_6	92	1122	4.212077	9.87E-07	166	335	1.791506	9.97E-07
	x_7	92	1121	4.251761	9.94E-07	166	335	1.823336	9.95E-07
	x_8	92	1122	4.170212	9.93E-07	165	333	1.80167	9.99E-07
100000	x_1	102	1261	10.21122	9.69E-07	194	390	4.263049	9.9E-07
	x_2	99	1250	10.5986	9.8E-07	190	381	4.195978	9.9E-07
	x_3	100	1250	10.28765	9.79E-07	191	383	4.229144	9.94E-07
	x_4	101	1249	10.87119	9.84E-07	192	385	4.175026	9.96E-07
	x_5	102	1251	10.25577	9.94E-07	195	393	4.244361	9.9E-07
	x_6	102	1252	10.65662	9.88E-07	195	393	4.259356	9.92E-07
	x_7	102	1251	10.12021	9.94E-07	195	393	4.282953	9.9E-07
	x_8	102	1252	10.0277	9.93E-07	194	391	4.220459	9.93E-07

Table 6: Numerical Comparison of HSG and SGP methods for problem 5

		HSG				SGP			
DIM	**ISP**	**ITER**	**FEVAL**	**TIME**	**NORM**	**ITER**	**FEVAL**	**TIME**	**NORM**
1000	x_1	9	33	0.02175	6.09E-07	6	14	0.003196	6.51E-07
	x_2	7	23	0.00694	7.65E-07	5	12	0.004627	3.18E-07
	x_3	7	22	0.008987	7.32E-07	6	14	0.00575	2.23E-08
	x_4	8	27	0.003655	8.14E-07	6	14	0.007134	4.06E-07
	x_5	1	14	0.007605	0	7	17	0.008116	9.64E-07
	x_6	1	14	0.005733	0	9	22	0.011394	6.68E-08
	x_7	1	14	0.003686	0	8	20	0.011024	1.89E-07
	x_8	1	14	0.004478	0	10	25	0.013326	2.7E-08
10000	x_1	10	36	0.042983	1.94E-07	7	16	0.026687	2.04E-08
	x_2	8	26	0.038262	2.44E-07	6	14	0.022402	9.95E-09
	x_3	8	25	0.03576	2.34E-07	6	14	0.025002	7.05E-08
	x_4	9	30	0.049849	2.6E-07	7	16	0.028925	1.27E-08
	x_5	1	14	0.018574	0	8	19	0.030818	3.02E-08
	x_6	1	14	0.019756	0	9	22	0.037613	2.11E-07
	x_7	1	14	0.019674	0	8	20	0.03446	5.98E-07
	x_8	1	14	0.012966	0	10	25	0.03914	8.53E-08
50000	x_1	10	36	0.096474	4.35E-07	7	16	0.029361	4.56E-08
	x_2	8	26	0.05228	5.46E-07	6	14	0.060396	2.22E-08
	x_3	8	25	0.077648	5.22E-07	6	14	0.065584	1.58E-07
	x_4	9	30	0.064128	5.81E-07	7	16	0.047721	2.85E-08
	x_5	1	14	0.058528	0	8	19	0.055061	6.75E-08
	x_6	1	14	0.05805	0	9	22	0.093499	4.72E-07
	x_7	1	14	0.059652	0	9	22	0.084222	1.32E-08
	x_8	1	14	0.03414	0	10	25	0.087145	1.91E-07
100000	x_1	10	36	0.15789	6.15E-07	7	16	0.106841	6.45E-08
	x_2	8	26	0.121342	7.72E-07	6	14	0.091209	3.14E-08
	x_3	8	25	0.127373	7.39E-07	6	14	0.10202	2.23E-07
	x_4	9	30	0.119771	8.21E-07	7	16	0.081254	4.02E-08
	x_5	1	14	0.062876	0	8	19	0.12956	9.55E-08
	x_6	1	14	0.10068	0	9	22	0.138084	6.68E-07
	x_7	1	14	0.087004	0	9	22	0.111249	1.87E-08
	x_8	1	14	0.076061	0	10	25	0.135734	2.7E-07

Table 7: Numerical Comparison of HSG and SGP methods for problem 6

		HSG				SGP			
DIM	**ISP**	**ITER**	**FEVAL**	**TIME**	**NORM**	**ITER**	**FEVAL**	**TIME**	**NORM**
1000	x_1	27	130	0.110869	8.46E-07	87	355	0.038727	8.5E-07
	x_2	27	152	0.03359	3.84E-07	98	417	0.063436	4.97E-07
	x_3	26	189	0.013507	5.26E-07	94	412	0.065027	9.02E-07
	x_4	22	114	0.030913	9.57E-07	85	357	0.068865	8.28E-07
	x_5	23	135	0.031662	9.8E-07	71	275	0.071683	9.62E-07
	x_6	32	200	0.054317	9.7E-07	83	329	0.067108	9.39E-07
	x_7	32	194	0.043469	5.64E-07	101	413	0.067194	4.45E-07
	x_8	28	167	0.022114	9.77E-07	89	357	0.061408	8.87E-07
10000	x_1	30	180	0.122853	4.46E-07	80	336	0.17351	9.97E-07
	x_2	30	188	0.128457	2.27E-07	90	375	0.211582	9.84E-07
	x_3	31	208	0.124282	5.45E-07	85	366	0.231805	9.91E-07
	x_4	26	138	0.109561	9.79E-07	91	376	0.21884	8.3E-07
	x_5	32	193	0.131256	3.07E-07	71	273	0.147586	9.87E-07
	x_6	28	164	0.10831	8E-07	79	317	0.189958	9.58E-07
	x_7	28	167	0.081835	7.96E-07	87	352	0.200406	8.28E-07
	x_8	33	192	0.109065	8.95E-07	86	348	0.207741	9.64E-07
50000	x_1	29	166	0.292088	2.59E-07	78	318	0.545327	9.77E-07
	x_2	29	175	0.277888	4E-07	97	397	0.647195	9.88E-07
	x_3	30	204	0.340097	2.58E-07	92	385	0.709076	9.97E-07
	x_4	33	174	0.34449	6.66E-07	92	387	0.628403	9.51E-07
	x_5	30	159	0.30115	2.61E-07	71	272	0.455039	9.96E-07
	x_6	37	229	0.367311	2.09E-07	79	313	0.577814	9.52E-07
	x_7	30	175	0.274331	5.1E-07	86	340	0.569439	9.85E-07
	x_8	32	168	0.277867	5.23E-07	100	417	0.677655	6.21E-07
100000	x_1	31	190	0.815499	1.51E-07	87	352	1.50231	8.48E-07
	x_2	33	204	0.948442	3.26E-07	93	391	1.661453	7.38E-07
	x_3	26	174	0.700978	7.82E-07	82	348	1.464409	9.86E-07
	x_4	25	118	0.647404	9.76E-07	86	362	1.528764	5.54E-07
	x_5	27	141	0.610669	7.7E-07	71	272	1.189805	9.3E-07
	x_6	32	182	0.871517	3.68E-07	86	337	1.496047	8.65E-07
	x_7	28	149	0.66063	7.67E-07	85	336	1.443586	8.61E-07
	x_8	25	133	0.59816	3.78E-07	92	368	1.564383	9.44E-07

Table 8: Numerical Comparison of HSG and SGP methods for problem 7

		HSG				SGP			
DIM	ISP	ITER	FEVAL	TIME	NORM	ITER	FEVAL	TIME	NORM
1000	x_1	8	25	0.120801	5.78E-07	21	63	0.009614	6.19E-07
	x_2	8	25	0.015304	8.81E-07	22	66	0.02953	4.76E-07
	x_3	8	25	0.016758	8.47E-07	21	63	0.032447	9.07E-07
	x_4	8	25	0.004738	7.46E-07	21	63	0.031462	7.99E-07
	x_5	8	25	0.014963	2.42E-07	20	60	0.033661	5.12E-07
	x_6	7	22	0.013375	7.28E-07	18	54	0.022337	6.1E-07
	x_7	7	22	0.00937	9.4E-07	7	19	0.012794	5.57E-07
	x_8	8	25	0.015453	2.63E-07	7	18	0.012037	3.03E-08
10000	x_1	9	28	0.05657	1.85E-07	20	59	0.10127	7.6E-08
	x_2	9	28	0.031323	2.81E-07	21	62	0.098426	5.85E-08
	x_3	9	28	0.045735	2.71E-07	19	56	0.082544	2.21E-07
	x_4	9	28	0.069502	2.38E-07	23	68	0.058387	1.26E-08
	x_5	8	25	0.036801	7.65E-07	18	53	0.04099	1.25E-07
	x_6	8	25	0.056165	2.32E-07	18	53	0.084984	3.79E-08
	x_7	8	25	0.053186	3E-07	5	12	0.039959	1.35E-07
	x_8	8	25	0.033629	8.32E-07	6	15	0.044568	1.9E-07
50000	x_1	9	28	0.124372	4.13E-07	19	56	0.204414	3.36E-07
	x_2	9	28	0.131859	6.29E-07	18	52	0.188405	1.99E-08
	x_3	9	28	0.177138	6.05E-07	20	59	0.194514	2.49E-07
	x_4	9	28	0.140612	5.33E-07	18	53	0.195097	8.6E-07
	x_5	9	28	0.096043	1.73E-07	17	50	0.216363	5.52E-07
	x_6	8	25	0.14649	5.2E-07	15	43	0.181963	1.29E-08
	x_7	8	25	0.139724	6.71E-07	6	15	0.110283	1.53E-07
	x_8	9	28	0.132728	1.88E-07	7	18	0.104996	2.14E-07
100000	x_1	9	28	0.271453	5.84E-07	19	56	0.562865	4.76E-07
	x_2	9	28	0.260537	8.89E-07	17	49	0.358332	5.58E-08
	x_3	9	28	0.267964	8.55E-07	17	49	0.346541	5.36E-08
	x_4	9	28	0.237759	7.54E-07	19	55	0.408642	1.2E-08
	x_5	9	28	0.268326	2.44E-07	17	50	0.463142	7.8E-07
	x_6	8	25	0.223161	7.35E-07	14	40	0.307303	3.61E-08
	x_7	8	25	0.307605	9.48E-07	5	12	0.19482	4.28E-07
	x_8	9	28	0.265437	2.66E-07	7	18	0.166348	3.03E-07

Table 9: Numerical Comparison of HSG and SGP methods for problem 8

		HSG				SGP			
DIM	**ISP**	**ITER**	**FEVAL**	**TIME**	**NORM**	**ITER**	**FEVAL**	**TIME**	**NORM**
1000	x_1	59	291	0.022342	7.09E-07	74	270	0.03314	8.83E-07
	x_2	45	220	0.04485	8.58E-07	71	266	0.064066	8.78E-07
	x_3	45	214	0.032972	8.44E-07	81	304	0.088796	8.04E-07
	x_4	50	236	0.018667	5.23E-07	80	299	0.078271	9.14E-07
	x_5	45	212	0.039955	7.55E-07	88	336	0.034955	6.87E-07
	x_6	57	279	0.025319	5.92E-07	85	320	0.066573	9.12E-07
	x_7	60	293	0.038769	4.87E-07	79	300	0.073987	8.34E-07
	x_8	58	296	0.024319	6.51E-07	89	334	0.052072	9.12E-07
10000	x_1	61	292	0.142279	6.51E-07	86	319	0.212828	6.12E-07
	x_2	54	275	0.161509	8.9E-07	76	285	0.233712	6.58E-07
	x_3	44	200	0.100478	9.68E-07	88	332	0.221482	8.97E-07
	x_4	55	252	0.131647	6.99E-07	83	311	0.230731	8.36E-07
	x_5	75	485	0.233953	7.7E-07	104	410	0.269581	8.76E-07
	x_6	86	563	0.265828	7.37E-07	96	368	0.289794	7.38E-07
	x_7	70	330	0.174515	7.02E-07	107	431	0.266218	9.88E-07
	x_8	73	422	0.238606	5.83E-07	89	334	0.276688	7.57E-07
50000	x_1	65	387	0.718651	7.64E-07	105	400	0.917389	7.37E-07
	x_2	53	253	0.588221	7.68E-07	82	311	0.681641	9.23E-07
	x_3	59	295	0.592014	8.74E-07	83	311	0.726733	8.04E-07
	x_4	56	256	0.513466	6.96E-07	104	384	0.933497	9.88E-07
	x_5	212	1964	3.418223	6.73E-07	151	743	1.530829	9.95E-07
	x_6	127	871	1.587047	5.58E-07	130	526	1.126783	8.4E-07
	x_7	85	469	0.898776	5.56E-07	133	541	1.140028	9.23E-07
	x_8	100	731	1.326527	5.27E-07	119	472	1.044182	9.99E-07
100000	x_1	93	694	2.9377	8.14E-07	108	426	2.153943	7.76E-07
	x_2	60	300	1.438234	8.29E-07	79	294	1.556848	8.93E-07
	x_3	56	279	1.301234	9.25E-07	101	386	1.965254	8.53E-07
	x_4	62	307	1.427271	9.27E-07	124	470	2.439437	7.8E-07
	x_5	186	1663	6.861447	5.53E-07	167	739	3.638497	8.5E-07
	x_6	169	1228	5.663522	6.33E-07	131	542	2.689844	7.67E-07
	x_7	104	637	3.048335	7.01E-07	154	649	3.234712	7.99E-07
	x_8	140	1192	5.539761	9.41E-07	142	574	2.872369	7.79E-07

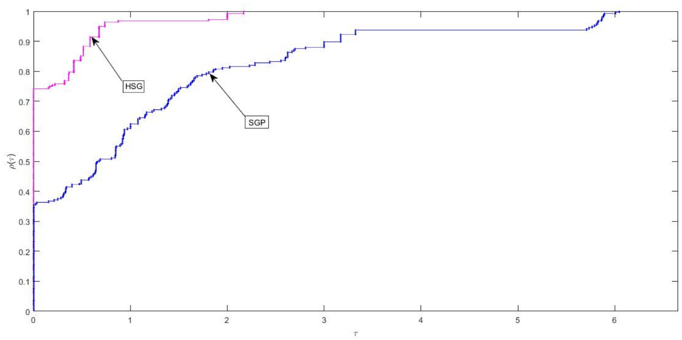

Figure 1: Performance profile with respect to number of iterations

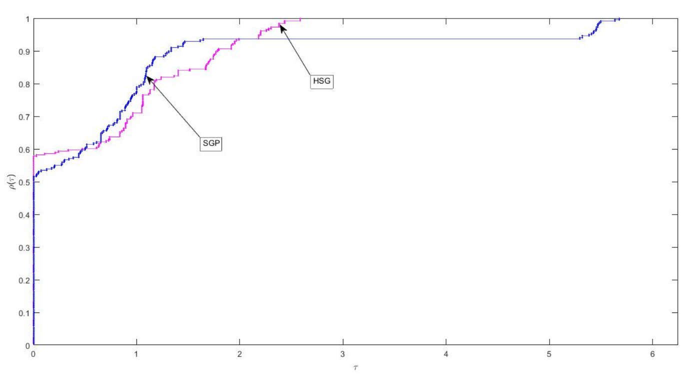

Figure 2: Performance profile with respect to number of functions evaluation

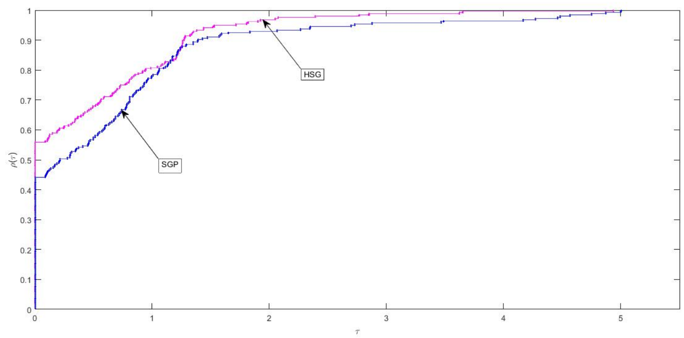

Figure 3: Performance profile with respect to CPU time

5 Conclusions

In this paper, we proposed a new hybrid positive spectral gradient projection method for monotone system of nonlinear equations with convex constraints. The search direction satisfies the sufficient descent property. The proposed method can be applied to solve nonsmooth equations as well as large scale equations because it does not require the Jacobian information of the nonlinear equation or it storage space. The global convergent under some suitable assumptions was established. Preliminary numerical experiments show that our proposed methods is efficient and promising.

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.

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] Y. Dai and Y. Yuan. A nonlinear conjugate gradient method with a strong global convergence property. *SIAM Journal on Optimization*, 10(1):177–182, 1999.

[7] Y. H. Dai, M. Al-Baali, and X. Yang. A positive Barzilai–Borwein-like stepsize and an extension for symmetric linear systems. In *Numerical Analysis and Optimization*, pages 59–75. Springer, 2015.

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

[9] N. A. Iusem and V. M. Solodov. Newton-type methods with generalized distances for constrained optimization. *Optimization*, 41(3):257–278, 1997.

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

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

[12] 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.

[13] 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.

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

[15] 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.

[16] J. Liu and Y. Feng. A derivative-free iterative method for nonlinear monotone equations with convex constraints. *Numerical Algorithms*, pages 1–18, 2018.

[17] 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.

[18] S. Y. Liu, Y. Y. Huang, and H. W. Jiao. Sufficient descent conjugate gradient methods for solving convex constrained nonlinear monotone equations. In *Abstract and Applied Analysis*, volume 2014. Hindawi, 2014.

[19] L. Min. A Liu-Storey-Type method for solving large scale nonlinear monotone equations. *Numerical Functional Analysis and Optimization*, 35(3):310–322, 2014.

[20] 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.

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

[22] M. Raydan. On the Barzilai and Borwein choice of steplength for the gradient method. *IMA Journal of Numerical Analysis*, 13(3):321–326, 1993.

[23] 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.

[24] C. Wang, Y. Wang, and C. Xu. A projection method for a system of nonlinear monotone equations with convex constraints. *Mathematical Methods of Operations Research*, 66(1):33–46, 2007.

[25] 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.

[26] Y. Xiao, Q. Wang, and Q. Hu. Non-smooth equations based methods for l_1-norm problems with applications to compressed sensing. *Nonlinear Analysis: Theory, Methods and Applications*, 74(11):3570–3577, 2011.

[27] 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.

[28] 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.

[29] N. Yuan. A derivative-free projection method for solving convex constrained monotone equations. *SCIENCE ASIA*, 43(3):195–200, 2017.

[30] 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.

[31] 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.

[32] Lian Zheng. A new projection algorithm for solving a system of nonlinear equations with convex constraints. *Bulletin of the Korean Mathematical Society*, 50(3):823–832, 2013.

[33] 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.

[34] 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 14 August 2018)
(Accepted 22 November 2018)