Performance Analysis of New Spectral and Hybrid Conjugate Gradient Methods for Solving Unconstrained Optimization Problems

Maulana Malik*, *Member, IAENG,* Mustafa Mamat, Siti Sabariah Abas, Ibrahim Mohammed Sulaiman and Sukono, *Member, IAENG*

***Abstract*—The spectral and hybrid conjugate gradient methods are part of the conjugate gradient methods. Conjugate gradient methods are among the iterative method for solving unconstrained optimization problems. In this paper, a new spectral and hybrid conjugate gradient methods are proposed. Based on some assumptions and strong Wolfe line search, the new spectral conjugate gradient method satisfies the global convergence properties. As well as the hybrid conjugate gradient method fulfill the global convergence properties under an exact line search. We also prove that the proposed methods fulfill the sufficient descent condition. Finally, based on some test problems, the numerical results of the proposed methods are very competitive and most efficient.**

***Index Terms*—Strong Wolfe line search, spectral conjugate gradient method, global convergence properties, hybrid conjugate gradient method, exact line search, sufficient descent condition.**

I. Introduction

THE new method in this paper is designed to solve unconstrained optimization problems, in which problems modeled as minimization problems:

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}), \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a smooth objective function and its gradient is available. The conjugate gradient method are an iterative method with generates a sequence $\{\mathbf{x}_k\}$ by formula [1]

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k, \quad k = 0, 1, 2, ..., \tag{2}$$

where $\mathbf{x}_0$ is initial point, $\mathbf{x}_k$ is point at kth iteration, α_k is a positive step length, and $\mathbf{d}_k$ is a search direction defined by:

$$\mathbf{d}_k = \begin{cases} -\mathbf{g}_k, & k = 0 \\ -\mathbf{g}_k + \beta_k \mathbf{d}_{k-1}, & k \geq 1 \end{cases}. \tag{3}$$

Manuscript is received on July 22, 2020; revised on Sept 9, 2020.

*Maulana Malik is a Lecturer at Department of Mathematics, Universitas Indonesia (UI), Depok 16424, Indonesia and a Ph.D. student in Mathematics at Universiti Sultan Zainal Abidin (UniSZA), Terengganu 22200, Malaysia. (e-mail: m.malik@sci.ui.ac.id)

Mustafa Mamat is a Professor of Applied Mathematics at Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin (UniSZA), Terengganu 22200, Malaysia. (e-mail: must@unisza.edu.my)

Siti Sabariah Abas is a Lecturer at Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin (UniSZA), Terengganu 22200, Malaysia. (e-mail: sabariahabas@unisza.edu.my)

Ibrahim Mohammed Sulaiman is a post-doctoral researcher at Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin (UniSZA), Terengganu 22200, Malaysia. (e-mail: sulaimanib@unisza.edu.my)

Sukono is an Associate Professor at Department of Mathematics, Universitas Padjadjaran (Unpad), Jatinangor 45363, Indonesia. (e-mail: sukono@unpad.ac.id)

Need to know that $\mathbf{g}_k = \mathbf{g}(\mathbf{x}_k) = \nabla f(\mathbf{x}_k)$ is a gradient of f at point $\mathbf{x}_k$, and β_k is a scalar known as the conjugate gradient coefficient [2]. Some well-known formulas for coefficients of the conjugate gradient method are:

$$\begin{aligned}
\beta_k^{HS} &= \frac{\mathbf{g}_k^T(\mathbf{g}_k - \mathbf{g}_{k-1})}{\mathbf{d}_{k-1}^T(\mathbf{g}_k - \mathbf{g}_{k-1})}, \\
\beta_k^{FR} &= \frac{\|\mathbf{g}_k\|^2}{\|\mathbf{g}_{k-1}\|^2}, \\
\beta_k^{CD} &= -\frac{\|\mathbf{g}_k\|^2}{\mathbf{d}_{k-1}^T\mathbf{g}_{k-1}}, \\
\beta_k^{LS} &= -\frac{\mathbf{g}_k^T(\mathbf{g}_k - \mathbf{g}_{k-1})}{\mathbf{d}_{k-1}^T\mathbf{g}_{k-1}}, \\
\beta_k^{DY} &= \frac{\|\mathbf{g}_k\|^2}{\mathbf{d}_{k-1}^T(\mathbf{g}_k - \mathbf{g}_{k-1})}, \\
\beta_k^{PRP} &= \frac{\mathbf{g}_k^T(\mathbf{g}_k - \mathbf{g}_{k-1})}{\|\mathbf{g}_{k-1}\|^2}, \\
\beta_k^{WYL} &= \frac{\mathbf{g}_k^T(\mathbf{g}_k - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|}\mathbf{g}_{k-1})}{\|\mathbf{g}_{k-1}\|^2}, \\
\beta_k^{NPRP} &= \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|}\left|\mathbf{g}_k^T\mathbf{g}_{k-1}\right|}{\|\mathbf{g}_{k-1}\|^2},
\end{aligned} \tag{4}$$

where $\|.\|$ represent the Euclidean norm of vectors and $\mathbf{g}_k^T$ is transpose $\mathbf{g}_k$. The proper naming of the methods above are Hestenes-Steifel (HS) method [3], Fletcher-Reeves (FR) method [4], Conjugate Descent (CD) method [5], Liu-Storey (LS) method [6], Dai-Yuan (DY) method [7], Polak-Ribiére-Polyak (PRP) method [8], Wei-Yao-Liu (WYL) method [9], and modified Wei-Yao-Liu (NPRP) method [10].

The value of step length α_k can obtained by using any type of line search such as exact line search and inexact line search. The formula of exact line search is defined as:

$$f(\mathbf{x}_k + \alpha_k \mathbf{d}_k) = \min_{\alpha\geq 0} f(\mathbf{x}_k + \alpha_k \mathbf{d}_k), \tag{5}$$

and inexact line search that are; the strong Wolfe line search defined as follows:

$$\begin{cases} f(\mathbf{x}_k + \alpha_k \mathbf{d}_k) \leq f(\mathbf{x}_k) + \delta\alpha_k \mathbf{g}_k^T \mathbf{d}_k \\ \left|\mathbf{g}\left(\mathbf{x}_k + \alpha_k \mathbf{d}_k\right)^T \mathbf{d}_k\right| \leq -\sigma \mathbf{g}_k^T \mathbf{d}_k \end{cases}, \tag{6}$$

where $0 < \delta < \sigma < 1$ and Armijo line search where the step size α_k is obtained by $\alpha_k = \max\{\rho^j, j = 0, 1, 2, ...\}$ satisfying

$$f(\mathbf{x}_k + \alpha_k \mathbf{d}_k) \leq f(\mathbf{x}_k) - \delta\alpha_k^2 \|\mathbf{d}_k\|^2$$

with the constants $\rho, \delta \in (0, 1)$ [11].

It is possible to classify the conjugate gradient method into many different types; the standard, hybrid, spectral, and three-term conjugate gradient methods. The methods mentioned above (HS, FR, CD, LS, DY, PRP, WYL, NPRP) are the standard conjugate gradient method.

The Spectral method was originally introduced by Barzilai and Borwein in 1988 [12] and Raydan developed the specrtal method to solve the optimization problems [13]. In addition, Birgin and Martinez [14] suggested three kinds of spectral methods that are a mixture of spectral and gradient conjugate methods with the search direction as follows

$$\mathbf{d}_k = -\theta_k \mathbf{g}_k + \beta_k \mathbf{s}_{k-1},$$

where θ_k is the spectral gradient parameter, $\mathbf{s}_{k-1} = \alpha_{k-1}\mathbf{d}_{k-1}$, and the coefficient β_k is determined by

$$\begin{aligned}\beta_k &= \frac{\left(\theta_k \mathbf{y}_{k-1} - \mathbf{s}_{k-1}\right)^T \mathbf{g}_k}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}},\\ \beta_k &= \frac{\theta_k \mathbf{g}_k^T \mathbf{y}_{k-1}}{\alpha_{k-1}\theta_{k-1}\mathbf{g}_{k-1}^T \mathbf{g}_{k-1}},\\ \beta_k &= \frac{\theta_k \mathbf{g}_k^T \mathbf{g}_k}{\alpha_{k-1}\theta_{k-1}\mathbf{g}_{k-1}^T \mathbf{g}_{k-1}},\end{aligned}$$

where

$$\mathbf{y}_{k-1} = \mathbf{g}_k - \mathbf{g}_{k-1}\ ,\ \theta_k = \frac{\mathbf{s}_{k-1}^T \mathbf{s}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}}.$$

Zhang et al. [15] suggested modifying the FR method in 2006 with the name of the modified FR method (MFR). The MFR method's search direction is defined as follows:

$$\mathbf{d}_k = \begin{cases} -\mathbf{g}_k, & k = 0 \\ -\theta_k \mathbf{g}_k + \beta_k \mathbf{d}_{k-1}, & k \geq 1 \end{cases}, \tag{7}$$

where

$$\beta_k = \beta_k^{FR}\ ,\ \theta_k = \frac{\mathbf{d}_{k-1}^T \mathbf{y}_{k-1}}{\|\mathbf{g}_{k-1}\|^2}.$$

It could easily be that the search direction of the MFR method can be written as follows:

$$\mathbf{d}_k = -\left(1 + \beta_k^{FR}\frac{\mathbf{g}_k^T \mathbf{d}_{k-1}}{\|\mathbf{g}_k\|^2}\right)\mathbf{g}_k + \beta_k^{FR}\mathbf{d}_{k-1}. \tag{8}$$

The main difference between standard and spectral conjugate gradient methods lies in the compute of the search direction $\mathbf{d}_k$. The search direction of the standard conjugate gradient method using formula (3), but the search direction of the spectral conjugate gradient method using formula (7).

As well as Liu and Jiang [16] proposed a spectral conjugate gradient method which is called the SCD method. The SCD method is run using the conjugate gradient coefficient (β_k) and spectral gradient parameter (θ_k) as follows:

$$\beta_k = \begin{cases} \beta_k^{CD}, & \text{if } \mathbf{g}_k^T \mathbf{d}_{k-1} \leq 0 \\ 0, & \text{else} \end{cases},$$

$$\theta_k = 1 - \frac{\mathbf{g}_k^T \mathbf{d}_{k-1}}{\mathbf{g}_{k-1}^T \mathbf{d}_{k-1}}.$$

In 2020, Jian et al. [17] introduced a new approach for spectral gradient parameter formula, the formula was inspired by the θ_k of the SCD method, which is written in form:

$$\theta_k^{JYJLL} = 1 + \frac{\left|\mathbf{g}_k^T \mathbf{d}_{k-1}\right|}{-\mathbf{g}_{k-1}^T \mathbf{d}_{k-1}},$$

and conjugate gradient coefficient formula as follows:

$$\beta_k^{JYJLL} = \frac{\|\mathbf{g}_k\|^2 - \frac{\left(\mathbf{g}_k^T \mathbf{d}_{k-1}\right)^2}{\|\mathbf{d}_{k-1}\|^2}}{\max\left\{\|\mathbf{g}_{k-1}\|^2, \mathbf{d}_{k-1}^T\left(\mathbf{g}_k - \mathbf{g}_{k-1}\right)\right\}}.$$

The sufficient descent condition and global convergence properties are also investigated by several researchers while proposing new methods of standard, spectral, and hybrid conjugate gradient methods. The search direction in conjugate gradient method satisfies the sufficient descent condition if there exist a constant $c > 0$ such that

$$\mathbf{g}_k^T \mathbf{d}_k \leq -c\|\mathbf{g}_k\|^2, \text{ for all } k \geq 0, \tag{9}$$

and the conjugate gradient method is global convergence if

$$\lim_{k\to\infty} \inf \|\mathbf{g}_k\| = 0 \tag{10}$$

(see [18]).

The MFR method fulfill the descent property ($\mathbf{g}_k^T \mathbf{d}_k < 0$) and with Armijo line search satisfies the global convergence properties even if the objective function is nonconvex. The SCD and JYJLL method satisfies the sufficient descent condition without depending any line search, and under strong Wolfe line search, the method fulfill the global convergence properties. For the NPRP method, Zhang has proven that method fulfill the sufficient descent condition with the strong Wolfe line search and converges globally for nonconvex minimization.

Besides the spectral conjugate gradient method, the hybrid conjugate gradient method can also be used to solve the problem (1). The hybrid conjugate gradient coefficient is a mixture of different parts of the standard conjugate gradient coefficient to give better performance.

Several hybrid conjugate gradient approaches have also been proposed in the literature. The most popular for hybrid conjugate gradient method are Touati-Ahmed and Storey (TS) method [19], Hu and Storey (HuS) method [20], Gilbert and Nocedal (GN) method [21], Dai and Yuan (hDY and LS-CD) method [22], Li and Zhao (P-W) method [23], and Hybrid-Jinbao, Han and Jiang (HJHJ) method [24]:

$$\begin{aligned}\beta_k^{TS} &= \begin{cases} \beta_k^{PRP}, & \text{if } 0 \leq \beta_k^{PRP} \leq \beta_k^{FR} \\ \beta_k^{FR}, & \text{otherwise} \end{cases},\\ \beta_k^{HuS} &= \max\left\{0, \min\left\{\beta_k^{PRP}, \beta_k^{FR}\right\}\right\},\\ \beta_k^{GN} &= \max\left\{-\beta_k^{FR}, \min\left\{\beta_k^{PRP}, \beta_k^{FR}\right\}\right\},\\ \beta_k^{hDY} &= \max\left\{0, \min\left\{\beta_k^{HS}, \beta_k^{DY}\right\}\right\},\\ \beta_k^{LS-CD} &= \max\left\{0, \min\left\{\beta_k^{LS}, \beta_k^{CD}\right\}\right\},\\ \beta_k^{P-W} &= \max\left\{\beta_k^{PRP}, \beta_k^{WYL}\right\},\\ \beta_k^{HJHJ} &= \frac{\|\mathbf{g}_k\|^2 - \max\left\{0, \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|}\mathbf{g}_k^T \mathbf{g}_{k-1}\right\}}{\max\left\{\|\mathbf{g}_{k-1}\|^2, \mathbf{d}_{k-1}^T(\mathbf{g}_k - \mathbf{g}_{k-1})\right\}}.\end{aligned}$$

The convergence properties and performance computational of the above methods have been studied by authors. The hybrid TS and HuS methods are known to fulfill the descent property and global convergence under the strong Wolfe line search, and computational results are more efficient than the FR and PRP methods. The hybrid GN method can be negative, since β_k^{FR} is always nonnegative, and when the HuS method jams, then the hybrid GN method is used instead. The hDY method is a mixture of HS and DY methods in which the global convergence of the method was identified in the rules of the Wolfe line search. The hybrid LS-CD approach used for the exact line search has comparable performance to the HuS method. Under certain line search, the hybrid P-W approach has been shown to be a global convergent. Under the Wolfe line search, the HJHJ method fulfills the global convergence properties.

For good references for studies about the conjugate gradient method can be seen in [25]-[33].

Inspired by the work of Zhang et al. [15] and Zhang [10], we further propose and analyze a new spectral conjugate gradient method to solve the unconstrained optimization problems. The sufficient descent condition of the new method will be presented, and under some assumptions, the global convergence properties are established using the strong Wolfe line search.

In this paper, we also propose a new hybrid conjugate gradient coefficient in which the sufficient descent condition and global convergence properties were proven under exact line search and performance computational compared with the HuS, GN, HDY, LS-CD, and HJHJ methods.

The next part of the paper is structured as follows. We are giving a new parameter for the spectral and a new coefficient for the hybrid conjugate gradient methods in Section II. In Section III, we present the global convergence analysis of the new spectral conjugate gradient method and in Section IV we provide the convergence analysis for the new hybrid conjugate gradient method. In Section V, the numerical results are presented to illustrate the performance of our new methods. The conclusion in this paper is presented in Section VI.

II. New Parameter and Coefficient

Recently, Malik et al. [34] propose a new β_k which its inspired by the formula β_k^{NPRP}. The coefficient β_k is defined as follows:

$$\beta_k^{MMSIS} = \begin{cases} X & ,\text{if } Y \\ 0 & ,\text{otherwise} \end{cases} \tag{11}$$

where

$$\begin{aligned} X &= \frac{\|\mathbf{g}_k\|^2 - \frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|}\left|\mathbf{g}_k^T\mathbf{g}_{k-1}\right| - \left|\mathbf{g}_k^T\mathbf{g}_{k-1}\right|}{\|\mathbf{d}_{k-1}\|^2}, \\ Y &= \|\mathbf{g}_k\|^2 > \left(\frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} + 1\right)\left|\mathbf{g}_k^T\mathbf{g}_{k-1}\right|. \end{aligned}$$

Based on formula (8), we propose a new spectral gradient parameter as follows:

$$\theta_k^{MMSIS} = 1 + \beta_k^{MMSIS}\frac{\mathbf{g}_k^T\mathbf{d}_{k-1}}{\|\mathbf{g}_k\|^2}, \tag{12}$$

where MMSIS is denotes Malik, Mustafa, Sabariah, Ibrahim, Sukono.

In the following, we establish the new algorithm of the spectral MMSIS (SpMMSIS) method for solving unconstrained optimization problems.

Algorithm 1. *(SpMMSIS Method)*

Step 1. Given any an initial point $\mathbf{x}_0 \in \mathbb{R}^n$, stopping criteria $\epsilon > 0$, parameters σ, and δ. Suppose that $\mathbf{d}_0 = -\mathbf{g}_0$, set $k := 1$.

Step 2. Calculate $\|\mathbf{g}_k\|$, if $\|\mathbf{g}_k\| \leq \epsilon$ then stop. Otherwise, go to Step 3.

Step 3. Calculate β_k^{MMSIS} using (11).

Step 4. Calculate θ_k^{MMSIS} using (12).

Step 5. Calculate search direction $\mathbf{d}_k$ using (7).

Step 6. Calculate step length α_k using the strong Wolfe line search (6).

Step 7. Set $k := k+1$ and calculate the next iterate $\mathbf{x}_{k+1}$ using (2). Go to Step 2.

Secondly, we propose the new hybrid coefficient β_k which is known as β_k^{HMMSIS}. The new β_k^{HMMSIS} is motivated from HuS method, where the β_k^{FR} in HuS method substituted by β_k^{MMSIS}, β_k^{PRP} is retained and expanded by multiplying with constant μ. Hence the proposed new coefficient is defined as follows:

$$\beta_k^{HMMSIS} = \max\{0, \mu \min\{\beta_k^{PRP}, \beta_k^{MMSIS}\}\} \tag{13}$$

where $\mu = 6$ and HMMSIS denotes Hybrid MMSIS.

We now present our new HMMSIS algorithm.

Algorithm 2. *(HMMSIS Method)*

Step 1. Given $\mathbf{x}_0 \in \mathbb{R}^n$, stopping tolerance $\epsilon > 0$, set $k = 0$.

Step 2. Compute $\|\mathbf{g}_k\|$. If $\|\mathbf{g}_k\| \leq \epsilon$ then stop. Else, go to Step 3.

Step 3. Calculate β_k using (13).

Step 4. Calculate $\mathbf{d}_k$ using (3).

Step 5. Calculate α_k using the exact line search (5).

Step 6. Set $k := k+1$ and use (2) to compute the next iteration of $\mathbf{x}_{k+1}$. Just go to Step 2.

III. Global Convergence Analysis New Spectral Conjugate Gradient Method

In this section, the sufficient descent condition and global convergence properties of the SpMMSIS method will be presented.

The following theorem shows that the SpMMSIS method possesses the sufficient descent condition without depending any line search.

Theorem 1. *Suppose that the sequences $\{\mathbf{g}_k\}$ and $\{\mathbf{d}_k\}$ be generated by Algorithm 1, and let the step length α_k be calculated by any line search, then*

$$\mathbf{g}_k^T\mathbf{d}_k = -\|\mathbf{g}_k\|^2 < 0$$

holds for any $k \geq 0$.

Proof: We first prove for $k = 0$. From (7), we have $\mathbf{d}_0 = -\mathbf{g}_0$. Further we obtain $\mathbf{g}_0^T\mathbf{d}_0 = -\mathbf{g}_0^T\mathbf{g}_0 = -\|\mathbf{g}_0\|^2 < 0$. Now, we prove for $k \geq 1$. From (7), we get

$$\mathbf{d}_k = -\theta_k\mathbf{g}_k + \beta_k\mathbf{d}_{k-1}.$$

Substituting θ_k by θ_k^{MMSIS} and β_k by β_k^{MMSIS}, then we have

$$\mathbf{d}_k = -\theta_k^{MMSIS}\mathbf{g}_k + \beta_k^{MMSIS}\mathbf{d}_{k-1}. \tag{14}$$

The proof is split into two cases based on the value of β_k^{MMSIS} as follows.

- Case 1. If $\|\mathbf{g}_k\|^2 > \left(\frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|} + 1\right)\left|\mathbf{g}_k^T\mathbf{g}_{k-1}\right|$, then from (11) and (14), we obtain

$$\mathbf{d}_k = -\mathbf{g}_k.$$

Multiply both sides of equation above by $\mathbf{g}_k^T$, we get $\mathbf{g}_k^T\mathbf{d}_k = -\|\mathbf{g}_k\|^2 < 0$. Hence, the sufficient descent condition holds.

- Case 2. If $\|\mathbf{g}_k\|^2 \leq \left(\frac{\|\mathbf{g}_k\|}{\|\mathbf{g}_{k-1}\|}+1\right)\left|\mathbf{g}_k^T\mathbf{g}_{k-1}\right|$, then from (11), (12), and (14), we have

$$\mathbf{d}_k = -\left(1 + X\frac{\mathbf{g}_k^T\mathbf{d}_{k-1}}{\|\mathbf{g}_k\|^2}\right)\mathbf{g}_k + X\mathbf{d}_{k-1}.$$

Multiply both sides by $\mathbf{g}_k^T$, we obtain

$$\begin{aligned}\mathbf{g}_k^T\mathbf{d}_k &= -\left(1 + X\frac{\mathbf{g}_k^T\mathbf{d}_{k-1}}{\|\mathbf{g}_k\|^2}\right)\mathbf{g}_k^T\mathbf{g}_k + X\mathbf{g}_k^T\mathbf{d}_{k-1}\\ &= -\|\mathbf{g}_k\|^2 - X\mathbf{g}_k^T\mathbf{d}_{k-1} + X\mathbf{g}_k^T\mathbf{d}_{k-1}\\ &= -\|\mathbf{g}_k\|^2 < 0.\end{aligned}$$

Hence, the sufficient descent condition holds for $k \geq 1$. The proof is completed. ■

The next lemma is needed to prove the global convergence properties of the SpMMSIS method.

Lemma 1. β_k^{MMSIS} *satisfies*

$$0 \leq \beta_k^{MMSIS} \leq \frac{\|\mathbf{g}_k\|^2}{\|\mathbf{d}_{k-1}\|^2}, \forall\, k \geq 0.$$

Proof: See [34]. ■

In the global convergence analysis of the conjugate gradient methods, we will assume that

Assumption 1. *(A1) The level set $\mathcal{T} = \{\mathbf{x} : f(\mathbf{x}) \leq f(\mathbf{x}_0)\}$ is bounded. (A2) Let $\mathcal{M}$ be some neighborhood of $\mathcal{T}$, then f is continuous and differentiable, and its gradient $\mathbf{g}(\mathbf{x})$ is Lipschitz continuous on $\mathcal{M}$ with Lipschitz constant $L > 0$; i.e.,*

$$\|\mathbf{g}(\mathbf{x}) - \mathbf{g}(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|, \quad \text{for all } \ \mathbf{x}, \mathbf{y} \in \mathcal{M}.$$

The following lemma often called the Zoutendijk condition, is used to prove the global convergence properties of the conjugate gradient method, which has been proven by Zoutendijk in [35].

Lemma 2. *Suppose that Assumption 1 holds. Consider any conjugate gradient method with (2), where $\mathbf{d}_k$ satisfies the descent condition such that $\mathbf{g}_k^T\mathbf{d}_k < 0$, and step length α_k determined by the exact line search (5) or strong Wolfe line search (6). Then*

$$\sum_{k=0}^{\infty}\frac{(\mathbf{g}_k^T\mathbf{d}_k)^2}{\|\mathbf{d}_k\|^2} < \infty.$$

The next theorem establishes the global convergence properties of SpMMSIS method under strong Wolfe line search.

Theorem 2. *Suppose that Assumption 1 holds, and let the sequence $\{\mathbf{x}_k\}$ be generated by Algorithm 1, where step length α_k be calculated by the strong Wolfe line search (6). Then*

$$\lim_{k\to\infty}\inf\|\mathbf{g}_k\| = 0. \tag{15}$$

Hence, the SpMMSIS method is global convergence.

Proof: Suppose by contradiction that (15) is not true. Then there exist constant W such that $\|\mathbf{g}_k\| \geq W$, for all $k \geq 0$, further we have

$$\frac{1}{\|\mathbf{g}_k\|^2} \leq \frac{1}{W^2}. \tag{16}$$

By rewriting (14), we get

$$\mathbf{d}_k + \theta_k^{MMSIS}\mathbf{g}_k = \beta_k^{MMSIS}\mathbf{d}_{k-1}.$$

Squaring both sides yields:

$$\begin{aligned}\|\mathbf{d}_k\|^2 &= (\beta_k^{MMSIS})^2\|\mathbf{d}_{k-1}\|^2 - 2\theta_k^{MMSIS}\mathbf{g}_k^T\mathbf{d}_k - \\ &\quad \left(\theta_k^{MMSIS}\right)^2\|\mathbf{g}_k\|^2.\end{aligned}$$

Dividing both sides by $(\mathbf{g}_k^T\mathbf{d}_k)^2$, then we obtain

$$\begin{aligned}\frac{\|\mathbf{d}_k\|^2}{(\mathbf{g}_k^T\mathbf{d}_k)^2} &= \frac{\left(\beta_k^{MMSIS}\right)^2\|\mathbf{d}_{k-1}\|^2}{(\mathbf{g}_k^T\mathbf{d}_k)^2} - \frac{2\theta_k^{MMSIS}}{\mathbf{g}_k^T\mathbf{d}_k} - \\ &\quad \frac{\left(\theta_k^{MMSIS}\right)^2\|\mathbf{g}_k\|^2}{(\mathbf{g}_k^T\mathbf{d}_k)^2}.\end{aligned}$$

From Theorem 1, we have $\mathbf{g}_k^T\mathbf{d}_k = -\|\mathbf{g}_k\|^2$. So the above equation becomes

$$\begin{aligned}\frac{\|\mathbf{d}_k\|^2}{(\mathbf{g}_k^T\mathbf{d}_k)^2} &= \left(\beta_k^{MMSIS}\right)^2\frac{\|\mathbf{d}_{k-1}\|^2}{\|\mathbf{g}_k\|^4} + \frac{2\theta_k^{MMSIS}}{\|\mathbf{g}_k\|^2}\\ &\quad -\frac{\left(\theta_k^{MMSIS}\right)^2}{\|\mathbf{g}_k\|^2}\\ &= \left(\beta_k^{MMSIS}\right)^2\frac{\|\mathbf{d}_{k-1}\|^2}{\|\mathbf{g}_k\|^4} - \\ &\quad \frac{1}{\|\mathbf{g}_k\|^2}\left(\left(\theta_k^{MMSIS}\right)^2 - 2\theta_k^{MMSIS}\right)\\ &= \left(\beta_k^{MMSIS}\right)^2\frac{\|\mathbf{d}_{k-1}\|^2}{\|\mathbf{g}_k\|^4} - \\ &\quad \frac{1}{\|\mathbf{g}_k\|^2}\left(\left(\theta_k^{MMSIS} - 1\right)^2 - 1\right)\\ &= \left(\beta_k^{MMSIS}\right)^2\frac{\|\mathbf{d}_{k-1}\|^2}{\|\mathbf{g}_k\|^4} - \frac{\left(\theta_k^{MMSIS} - 1\right)^2}{\|\mathbf{g}_k\|^2}\\ &\quad +\frac{1}{\|\mathbf{g}_k\|^2}\\ &\leq \left(\beta_k^{MMSIS}\right)^2\frac{\|\mathbf{d}_{k-1}\|^2}{\|\mathbf{g}_k\|^4} + \frac{1}{\|\mathbf{g}_k\|^2}.\end{aligned}$$

By Applying Lemma 1, we obtain

$$\begin{aligned}\frac{\|\mathbf{d}_k\|^2}{(\mathbf{g}_k^T\mathbf{d}_k)^2} &\leq \left(\frac{\|\mathbf{g}_k\|^2}{\|\mathbf{d}_{k-1}\|^2}\right)^2\frac{\|\mathbf{d}_{k-1}\|^2}{\|\mathbf{g}_k\|^4} + \frac{1}{\|\mathbf{g}_k\|^2}\\ &\leq \frac{1}{\|\mathbf{d}_{k-1}\|^2} + \frac{1}{\|\mathbf{g}_k\|^2}.\end{aligned}$$

Since $\frac{1}{\|\mathbf{d}_k\|^2} < \frac{4}{\|\mathbf{g}_k\|^2}$ (see equation (41) in [34]), and from (16), then we have

$$\begin{aligned}\frac{\|\mathbf{d}_k\|^2}{(\mathbf{g}_k^T\mathbf{d}_k)^2} &\leq \frac{1}{\|\mathbf{d}_{k-1}\|^2} + \frac{1}{\|\mathbf{g}_k\|^2}\\ &< \frac{4}{\|\mathbf{g}_{k-1}\|^2} + \frac{1}{\|\mathbf{g}_k\|^2}\\ &< \frac{4}{W^2} + \frac{1}{W^2} = \frac{5}{W^2}.\end{aligned}$$

That implies

$$\frac{(\mathbf{g}_k^T\mathbf{d}_k)^2}{\|\mathbf{d}_k\|^2} > \frac{W^2}{5}. \tag{17}$$

Furthermore from (17), we can obtain

$$\sum_{k=0}^{n}\frac{(\mathbf{g}_k^T\mathbf{d}_k)^2}{\|\mathbf{d}_k\|^2} > \left(\sum_{k=0}^{n}\frac{W^2}{5} = \frac{W^2}{5}(n+1)\right).$$

Hence,

$$\sum_{k=0}^{\infty}\frac{(\mathbf{g}_k^T\mathbf{d}_k)^2}{\|\mathbf{d}_k\|^2} > \left(\lim_{n\to\infty}\frac{W^2}{5}(n+1)=\infty\right).$$

This contradicts Zoutendijk condition in Lemma 2. Therefore, (15) is true. Furthermore, based on (10), the SpMMSIS method fulfill the global convergence properties. The proof is completed. ■

IV. Global Convergence Analysis New Hybrid Conjugate Gradient Method

In this section, the sufficient descent condition and global convergence properties of the new hybrid conjugate gradient method will be discussed.

Therefore, we must attention to the following lemma.

Lemma 3. *The β_k^{HMMSIS} satisfies one of the following condition:*

1) If $0<\beta_k^{MMSIS}<\beta_k^{PRP}$, then

$$\beta_k^{HMMSIS}=6\beta_k^{MMSIS}\le 6\frac{\|\mathbf{g}_k\|^2}{\|\mathbf{d}_{k-1}\|^2}. \qquad (18)$$

2) If $\beta_k^{MMSIS}>\beta_k^{PRP}>0$, then

$$\beta_k^{HMMSIS}=6\beta_k^{PRP+}\le 6\frac{\|\mathbf{g}_k\|^2}{\|\mathbf{g}_{k-1}\|^2}. \qquad (19)$$

3) If $\beta_k^{PRP}<0$ or $\beta_k^{PRP}=\beta_k^{MMSIS}=0$, then

$$\beta_k^{HMMSIS}=0. \qquad (20)$$

Proof:

1) For $0<\beta_k^{MMSIS}<\beta_k^{PRP}$, then based on (13), we have $\beta_k^{HMMSIS}=6\beta_k^{MMSIS}$. Applying Lemma 1, we obtain $\beta_k^{HMMSIS}=6\beta_k^{MMSIS}\le 6\frac{\|\mathbf{g}_k\|^2}{\|\mathbf{d}_{k-1}\|^2}$.
2) For $\beta_k^{MMSIS}>\beta_k^{PRP}>0$, then based on (13), we have $\beta_k^{HMMSIS}=6\beta_k^{PRP+}$. From (4), we obtain $\beta_k^{HMMSIS}=6\beta_k^{PRP+}\le 6\frac{\|\mathbf{g}_k\|^2}{\|\mathbf{g}_{k-1}\|^2}$.
3) For $\beta_k^{PRP}<0$ or $\beta_k^{PRP}=\beta_k^{MMSIS}=0$, then based on (13),we have $\beta_k^{HMMSIS}=0$.

The proof is completed. ■

First, we will show that for the HMMSIS method the sufficient descent condition will be fulfilled.

Theorem 3. *Let the sequences $\{\mathbf{g}_k\}$ and $\{\mathbf{d}_k\}$ be generated by Algorithm 2 under the exact line search. Then the sufficient descent condition holds.*

Proof: If $k=0$, then $\mathbf{d}_0=-\mathbf{g}_0$, and we get $\mathbf{g}_0^T\mathbf{d}_0=-\mathbf{g}_0^T\mathbf{g}_0=-\|\mathbf{g}_0\|^2$. Therefore, condition (9) holds true. We also need to proof that for $k\ge 1$, condition (9) will also hold true. From (3), multiply both side by $\mathbf{g}_k^T$, then

$$\mathbf{g}_k^T\mathbf{d}_k=-\mathbf{g}_k^T\mathbf{g}_k+\beta_k^{HMMSIS}\mathbf{g}_k^T\mathbf{d}_{k-1}.$$

For the exact line search, we know that $\mathbf{g}_k^T\mathbf{d}_{k-1}=0$. Thus,

$$\mathbf{g}_k^T\mathbf{d}_k=-\mathbf{g}_k^T\mathbf{g}_k=-\|\mathbf{g}_k\|^2, \qquad (21)$$

which implies that the sufficient descent condition holds true for $k\ge 1$. Hence, for the HMMSIS method, the sufficient descent condition under exact line search holds. ■

We need the following lemma to prove the global convergence properties of the HMMSIS method under the exact line search.

Lemma 4. *Suppose that any conjugate gradient method in the form (2) and (3), where α_k is calculated by exact line search (5). Then the following relation holds:*

$$\frac{1}{\|\mathbf{d}_k\|^2}\le\frac{1}{\|\mathbf{g}_k\|^2},\ \forall k\ge 0. \qquad (22)$$

Proof: Note that we have the following relation:

$$\|\mathbf{g}_k+\mathbf{d}_k\|^2=\|\mathbf{g}_k\|^2+2\mathbf{g}_k^T\mathbf{d}_k+\|\mathbf{d}_k\|^2.$$

By applying (21) to equation above, we have

$$\|\mathbf{g}_k+\mathbf{d}_k\|^2+\|\mathbf{g}_k\|^2=\|\mathbf{d}_k\|^2.$$

Furthermore,

$$\|\mathbf{g}_k\|^2\le\|\mathbf{d}_k\|^2,$$

which means,

$$\frac{1}{\|\mathbf{d}_k\|^2}\le\frac{1}{\|\mathbf{g}_k\|^2},\ \forall k\ge 0.$$

The proof is finished. ■

The next theorem establishes the global convergence properties of the HMMSIS method under the exact line search.

Theorem 4. *Suppose that Assumption 1 holds. Assume the conjugate gradient method in the form (2) and (3), where α_k is calculated by the exact line search (5) and β_k is calculated by β_k^{HMMSIS}. Also, consider the sufficient descent condition (9) holds. Then*

$$\lim_{k\to\infty}\inf\|\mathbf{g}_k\|=0. \qquad (23)$$

Hence, the HMMSIS method is global convergence.

Proof: Assume that (23) does not hold. Then there exist a constant $H>0$ such that $\|\mathbf{g}_k\|\ge H,\forall k\ge 0$, it becomes

$$\frac{1}{\|\mathbf{g}_k\|^2}\le\frac{1}{H^2}. \qquad (24)$$

Rewriting (3) as

$$\mathbf{d}_k+\mathbf{g}_k=\beta_k^{HMMSIS}\mathbf{d}_{k-1},$$

and squaring both side of the equation, we obtain

$$\|\mathbf{d}_k\|^2=\left(\beta_k^{HMMSIS}\right)^2\|\mathbf{d}_{k-1}\|^2-2\mathbf{g}_k^T\mathbf{d}_k-\|\mathbf{g}_k\|^2.$$

Dividing both sides by $\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2$, we get

$$\begin{aligned}\frac{\|\mathbf{d}_k\|^2}{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2} &= \frac{\left(\beta_k^{HMMSIS}\right)^2\|\mathbf{d}_{k-1}\|^2}{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2}-\frac{2}{\mathbf{g}_k^T\mathbf{d}_k}- \\ &\quad \frac{\|\mathbf{g}_k\|^2}{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2} \\ &= \frac{\left(\beta_k^{HMMSIS}\right)^2\|\mathbf{d}_{k-1}\|^2}{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2}- \\ &\quad \left(\frac{1}{\|\mathbf{g}_k\|}+\frac{\|\mathbf{g}_k\|}{\mathbf{g}_k^T\mathbf{d}_k}\right)^2+\frac{1}{\|\mathbf{g}_k\|^2} \\ &\le \frac{\left(\beta_k^{HMMSIS}\right)^2\|\mathbf{d}_{k-1}\|^2}{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2}+\frac{1}{\|\mathbf{g}_k\|^2}. \qquad (25)\end{aligned}$$

Note that, from Lemma 3 there are three cases for value β_k^{HMMSIS}. So that we have three cases for the inequality (25) above.

- Case 1. For $\beta_k^{HMMSIS} \leq 6\frac{\|\mathbf{g}_k\|^2}{\|\mathbf{d}_{k-1}\|^2}$, together with (21), then the equation (25) is going to be
$$\frac{\|\mathbf{d}_k\|^2}{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2} \leq \frac{36}{\|\mathbf{d}_{k-1}\|^2} + \frac{1}{\|\mathbf{g}_k\|^2}. \tag{26}$$
By using (22) and (24) , we obtain
$$\begin{aligned}\frac{\|\mathbf{d}_k\|^2}{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2} &\leq \frac{36}{\|\mathbf{g}_{k-1}\|^2} + \frac{1}{\|\mathbf{g}_k\|^2}\\ &\leq \frac{36}{H^2} + \frac{1}{H^2} = \frac{37}{H^2}.\end{aligned}$$
Then we get
$$\frac{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2}{\|\mathbf{d}_k\|^2} \geq \frac{H^2}{37}.$$
This implies,
$$\sum_{k=0}^{n}\frac{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2}{\|\mathbf{d}_k\|^2} \geq \sum_{k=0}^{n}\frac{H^2}{37} = (n+1)\frac{H^2}{37}.$$
Furthermore, if $n \to \infty$, we obtain
$$\sum_{k=0}^{\infty}\frac{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2}{\|\mathbf{d}_k\|^2} \geq \lim_{n\to\infty}(n+1)\frac{H^2}{37} = \infty.$$
This contradicts the Zoutendijk condition in Lemma 2. Therefore, (23) holds. So, the HMMSIS method is global convergence.
- Case 2. For $\beta_k^{HMMSIS} \leq 6\frac{\|\mathbf{g}_k\|^2}{\|\mathbf{g}_{k-1}\|^2}$, together with inequality (25) and (21), we obtain
$$\frac{\|\mathbf{d}_k\|^2}{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2} \leq \frac{36\|\mathbf{d}_{k-1}\|^2}{\|\mathbf{g}_{k-1}\|^2} + \frac{1}{\|\mathbf{g}_k\|^2}.$$
Since $\mathbf{d}_0 = -\mathbf{g}_0$, then we have $\frac{\|\mathbf{d}_0\|^2}{(\mathbf{g}_0^T\mathbf{d}_0)^2} = \frac{1}{\|\mathbf{g}_0\|^2}$, furthermore by using (24), we get
$$\begin{aligned}\frac{\|\mathbf{d}_k\|^2}{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2} &\leq \frac{36\|\mathbf{d}_{k-1}\|^2}{\|\mathbf{g}_{k-1}\|^2} + \frac{1}{\|\mathbf{g}_k\|^2}\\ &\leq \frac{36^2\|\mathbf{d}_{k-2}\|^2}{\|\mathbf{g}_{k-2}\|^2} + \frac{1}{\|\mathbf{g}_{k-1}\|^2} + \frac{1}{\|\mathbf{g}_k\|^2}\\ &\leq \ldots \leq \frac{36^k}{\|\mathbf{g}_0\|^2} + \sum_{i=1}^{k}\frac{1}{\|\mathbf{g}_i\|^2}\\ &\leq \frac{36^k}{\|\mathbf{g}_0\|^2} + \frac{k}{H^2} = U,\end{aligned}$$
where $U > 0$ is arbitrary constant. So, we get $\frac{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2}{\|\mathbf{d}_k\|^2} \geq \frac{1}{U}$. Furthermore, we have a relation
$$\sum_{k=0}^{n}\frac{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2}{\|\mathbf{d}_k\|^2} \geq \sum_{k=0}^{n}\frac{1}{U} = \frac{n+1}{U}.$$
Take $n \to \infty$, we get
$$\sum_{k=0}^{\infty}\frac{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2}{\|\mathbf{d}_k\|^2} \geq \lim_{n\to\infty}\frac{n+1}{U} = \infty.$$
This contradicts with Zoutendijk condition in Lemma 2. Hence, the condition (23) holds and the HMMSIS method is global convergence.
- Case 3. For $\beta_k^{HMMSIS} = 0$, then the inequality (25) is going to be
$$\frac{\|\mathbf{d}_k\|^2}{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2} \leq \frac{1}{\|\mathbf{g}_k\|^2}.$$
By applying (24), we obtain
$$\frac{\|\mathbf{d}_k\|^2}{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2} \leq \frac{1}{\|\mathbf{g}_k\|^2} \leq \frac{1}{H^2}.$$
Take summation, we have
$$\sum_{k=0}^{n}\frac{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2}{\|\mathbf{d}_k\|^2} \geq \sum_{k=0}^{n}H^2 = (n+1)H^2.$$
If $n \to \infty$, we get
$$\sum_{k=0}^{\infty}\frac{\left(\mathbf{g}_k^T\mathbf{d}_k\right)^2}{\|\mathbf{d}_k\|^2} \geq \lim_{n\to\infty}(n+1)H^2 = \infty.$$
This contradicts the Zoutendijk condition in Lemma 2. Hence, the condition (23) holds and the HMMSIS method is global convergence.

The proof is finished. ■

V. Numerical Results

In this section, the computational performance of the SpMMSIS and HMMSIS methods are analyzed. We compare the performance of the number of iterations (NOI) and the central processing unit (CPU) time. The performance profile under strong Wolfe line search, we use parameter $\sigma = 0.001$ and $\delta = 0.0001$. The performance of the SpMMSIS method will be compared with MFR, SCD, JYJLL, and NPRP method. Meanwhile, the HMMSIS method will be compared with the HuS, GN, HDY, LS-CD, and HJHJ method. The stopping criterion $\|\mathbf{g}_k\|^2 \leq 10^{-6}$, where $\epsilon = 10^{-6}$.

To find the performance of the SpMMSIS, MFR, SCD, JYJLL, NPRP, HMMSIS, HuS, GN, HDY, LS-CD, and HJHJ methods, we use some test functions together initial point considered by Andrei [36]. In addition, we use various dimensions of each test function, i.e., 2, 3, 4, 10, 50, 100, 500, 1,000, 5,000, and 10,000, as in the Malik et al. [37], [38], [39], [40].

In this paper, we list the test functions together with the initial points in Table I, and implemented MATLAB software with personal laptop; Intel Core i7 processor, 16 GB RAM, 64 bit Windows 10 Pro operating system. Note that we are changing the initial point in problem 10 with (-1,...,-1) and problem 24 with (2,...,2) for comparing the hybrid method. The numerical result of each method is presented in Table II and Table III.

From Table II, we can see that the SpMMSIS method is successful in solving all problems, whereas the MFR method is only 93%, the SCD method 96%, the JYJLL method 94%, and the NPRP method 96%.

From Table III, we can see that the HMMSIS method is successful in solving all problems, whereas the HuS method is only 97%, the HDY method 96%, the GN, LS-CD, and HJHJ method 98%.

TABLE I: The list of the test functions, dimension, and initial point.

Problem	Test Function	Dimension	Initial point	Problem	Test Function	Dimension	Initial point
1	Ext. White & Holst	1000	(-1.2,1,...,-1.2,1)	50	Ext. Maratos	10	(-1,...,-1)
2	Ext. White & Holst	1000	(10,...,10)	51	Six hump camel	2	(-1,2)
3	Ext. White & Holst	10000	(-1.2,1,...,-1.2,1)	52	Six hump camel	2	(-5,10)
4	Ext. White & Holst	10000	(5,...,5)	53	Three hump camel	2	(-1,2)
5	Ext. Rosenbrock	1000	(-1.2,1,...,-1.2,1)	54	Three hump camel	2	(2,-1)
6	Ext. Rosenbrock	1000	(10,...,10)	55	Booth	2	(5,5)
7	Ext. Rosenbrock	10000	(-1.2,1,...,-1.2,1)	56	Booth	2	(10,10)
8	Ext. Rosenbrock	10000	(5,...,5)	57	Trecanni	2	(-1,0.5)
9	Ext. Freudenstein & Roth	4	(0.5,-2,0.5,-2)	58	Trecanni	2	(-5,10)
10	Ext. Freudenstein & Roth	4	(5,5,5,5)	59	Zettl	2	(-1,2)
11	Ext. Beale	1000	(1,0.8,...,1,0.8)	60	Zettl	2	(10,10)
12	Ext. Beale	1000	(0.5,...,0.5)	61	Shallow	1000	(0,...,0)
13	Ext. Beale	10000	(-1,...,-1)	62	Shallow	1000	(10,...,10)
14	Ext. Beale	10000	(0.5,...,0.5)	63	Shallow	10000	(-1,...,-1)
15	Ext. Wood	4	(-3,-1,-3,-1)	64	Shallow	10000	(-10,...,-10)
16	Ext. Wood	4	(5,5,5,5)	65	Generalized Quartic	1000	(1,...,1)
17	Raydan 1	10	(1,...,1)	66	Generalized Quartic	1000	(20,...,20)
18	Raydan 1	10	(10,...,10)	67	Quadratic QF2	50	(0.5,...,0.5)
19	Raydan 1	100	(-1,...,-1)	68	Quadratic QF2	50	(30,...,30)
20	Raydan 1	100	(-10,...,-10)	69	Leon	2	(2,2)
21	Ext. Tridiagonal 1	500	(2,...,2)	70	Leon	2	(8,8)
22	Ext. Tridiagonal 1	500	(10,...,10)	71	Gen. Tridiagonal 1	10	(2,...,2)
23	Ext. Tridiagonal 1	1000	(1,...,1)	72	Gen. Tridiagonal 1	10	(10,...,10)
24	Ext. Tridiagonal 1	1000	(-10,...,-10)	73	Gen. Tridiagonal 2	4	(1,1,1,1)
25	Diagonal 4	500	(1,...,1)	74	Gen. Tridiagonal 2	4	(10,10,10,10)
26	Diagonal 4	500	(-20,...,-20)	75	POWER	10	(1,...,1)
27	Diagonal 4	1000	(1,...,1)	76	POWER	10	(10,...,10)
28	Diagonal 4	1000	(-30,...,-30)	77	Quadratic QF1	50	(1,...,1)
29	Ext. Himmelblau	1000	(1,...,1)	78	Quadratic QF1	50	(10,...,10)
30	Ext. Himmelblau	1000	(20,...,20)	79	Quadratic QF1	500	(1,...,1)
31	Ext. Himmelblau	10000	(-1,...,-1)	80	Quadratic QF1	500	(-5,...,-5)
32	Ext. Himmelblau	10000	(50,...,50)	81	Ext.quad.pen.QP2	100	(1,...,1)
33	FLETCHCR	10	(0,...,0)	82	Ext.quad.pen.QP2	100	(10,...,10)
34	FLETCHCR	10	(10,...,10)	83	Ext.quad.pen.QP2	500	(10,...,10)
35	Ext. Powell	100	(3,-1,0,1,...,1)	84	Ext.quad.pen.QP2	500	(50,...,50)
36	Ext. Powell	100	(5,...,5)	85	Ext.quad.pen.QP1	4	(1,1,1,1)
37	NONSCOMP	2	(3,3)	86	Ext.quad.pen.QP1	4	(10,10,10,10)
38	NONSCOMP	2	(10,10)	87	Quartic	4	(10,10,10,10)
39	Ext. DENSCHNB	10	(1,...,1)	88	Quartic	4	(15,15,15,15)
40	Ext. DENSCHNB	10	(10,...,10)	89	Matyas	2	(1,1)
41	Ext. DENSCHNB	100	(10,...,10)	90	Matyas	2	(20,20)
42	Ext. DENSCHNB	100	(-50,...,-50)	91	Colville	4	(2,2,2,2)
43	Ext. Penalty	10	(1,2,3,...,10)	92	Colville	4	(10,10,10,10)
44	Ext. Penalty	10	(-10,...,-10)	93	Dixon and Price	3	(1,1,1)
45	Ext. Penalty	100	(5,...,5)	94	Dixon and Price	3	(10,10,10)
46	Ext. Penalty	100	(10,...,10)	95	Sphere	5000	(1,...,1)
47	Hager	10	(1,...,1)	96	Sphere	5000	(10,...,10)
48	Hager	10	(-10,...,-10)	97	Sum Squares	50	(0,1,...,0,1)
49	Ext. Maratos	10	(1.1,0.1)	98	Sum Squares	50	(10,...,10)

TABLE II: Numerical results of the SpMMSIS, JYJLL, MFR, SCD, and NPRP methods.

Problem	SpMMSIS		JYJLL		MFR		SCD		NPRP	
	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU
1	12	0.0498	40	0.0933	49	0.1059	15	0.0625	17	0.0574
2	53	0.1931	202	0.8989	210	0.9704	105	0.252	55	0.2006
3	13	0.331	41	0.7117	50	0.8048	15	0.389	17	0.4105
4	40	1.199	291	6.5962	130	4.4795	382	5.0925	42	1.1005
5	23	0.0442	57	0.1101	59	0.1179	5954	3.6831	35	0.0587
6	33	0.0603	104	0.1882	163	0.2156	150	0.1317	25	0.0392
7	13	0.3435	57	0.8398	59	0.8258	6014	41.4951	36	0.3775
8	25	0.2982	171	2.7871	194	3.1419	1058	6.8776	23	0.303
9	8	5.90E-04	21	0.0066	21	0.0029	9	0.0183	9	0.005
10	11	8.22E-04	fail	fail	21	0.1658	fail	fail	fail	fail
11	14	0.0445	75	0.1352	75	0.1322	17	0.0732	26	0.0575
12	12	0.0433	81	0.1376	81	0.1436	335	0.4867	25	0.0639
13	14	0.289	87	1.2204	87	1.2208	14	0.32	28	0.4719
14	12	0.257	87	1.1868	87	1.2353	322	4.0799	25	0.4269
15	270	0.0116	fail	fail	fail	fail	922	0.0647	281	0.0111

(Continued on next page)

TABLE II – *Continued*

Problem	SpMMSIS		JYJLL		MFR		SCD		NPRP	
	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU
16	454	0.0198	fail	fail	fail	fail	2967	0.136	1427	0.0488
17	21	0.0015	19	0.0065	19	0.0047	20	0.015	27	0.002
18	34	0.0026	2350	0.1437	2620	0.1278	50	0.0087	29	0.0027
19	102	0.0279	93	0.0311	95	0.0243	253	0.0509	97	0.03
20	164	0.0433	801	0.349	fail	fail	390	0.0782	134	0.0424
21	12	0.0298	452	0.3883	452	0.4074	19	0.0373	22	0.0473
22	8	0.0231	9	0.0169	9	0.0225	15	0.0258	8	0.0243
23	12	0.0437	517	0.7546	517	0.767	19	0.0458	22	0.0695
24	8	0.0374	8	0.023	9	0.0265	17	0.0391	8	0.0372
25	2	0.002	2	0.0021	2	0.0017	5	0.0116	2	0.002
26	2	0.0021	2	0.002	2	0.0021	3	0.0022	2	0.0025
27	2	0.0033	2	0.0035	2	0.0025	4	0.0057	2	0.003
28	2	0.0029	2	0.0034	2	0.0031	4	0.0034	2	0.0032
29	9	0.011	14	0.0199	15	0.0208	21	0.042	13	0.0192
30	6	0.0124	9	0.0164	9	0.0211	10	0.0149	10	0.0177
31	11	0.1177	22	0.2385	17	0.2096	15	0.1613	15	0.1302
32	7	0.0865	13	0.1242	13	0.1156	18	0.1403	10	0.1028
33	85	0.0051	1142	0.0615	1208	0.0461	153	0.0217	85	0.0057
34	94	0.0061	403	0.0373	299	0.0214	148	0.0105	134	0.0087
35	239	0.0748	5487	1.1151	5589	1.0281	fail	fail	fail	fail
36	156	0.0564	6066	1.2068	6019	1.0706	fail	fail	fail	fail
37	12	6.97E-04	86	0.0081	156	0.0052	28	0.0071	15	0.000822
38	14	8.59E-04	88	0.0049	93	0.0043	22	0.0025	15	0.0013
39	7	4.90E-04	9	0.0011	9	0.000734	10	0.0105	10	0.000661
40	9	6.59E-04	11	0.0014	11	0.0014	21	0.0027	9	0.0011
41	10	0.0034	11	0.0026	11	0.0045	22	0.0087	9	0.0035
42	7	0.0047	63	0.0132	63	0.0099	11	0.0055	13	0.0052
43	20	0.0012	10	8.84E-04	11	0.0011	45	0.0151	14	0.000943
44	9	6.88E-04	19	0.0024	19	0.0014	54	0.0058	14	0.0012
45	9	0.0034	28	0.0087	28	0.0109	10	0.0044	10	0.0043
46	9	0.0038	fail	fail	28	0.1552	17	0.0121	10	0.0067
47	12	6.98E-04	11	0.0039	11	0.0031	12	0.0107	12	0.0014
48	18	0.0012	96	0.0096	97	0.0076	18	0.0081	19	0.0015
49	44	0.0038	3527	0.5159	fail	fail	1229	0.0732	48	0.0129
50	33	0.0029	165	0.0245	128	0.0121	47	0.0084	37	0.1736
51	7	4.52E-04	27	0.0027	27	8.09E-04	13	0.0021	10	0.0006
52	10	6.35E-04	264	0.0181	536	0.0154	13	0.0019	10	0.0009
53	13	0.0027	11	0.0033	11	0.0024	13	0.0032	12	0.0028
54	12	0.0024	11	0.0024	12	0.0039	13	0.0065	8	0.0017
55	2	2.02E-04	2	2.63E-04	2	2.63E-04	2	2.93E-04	2	0.000179
56	2	1.57E-04	2	2.03E-04	2	1.29E-04	2	0.0041	2	0.00018
57	1	1.32E-04	1	1.82E-04	1	1.64E-04	1	2.40E-04	1	0.000166
58	5	3.45E-04	7	8.31E-04	7	4.69E-04	5	0.0083	7	0.000465
59	11	6.17E-04	11	0.0081	11	6.34E-04	105	0.0064	9	0.000574
60	12	6.17E-04	16	0.0022	16	9.00E-04	68	0.0081	13	0.000806
61	8	0.0132	18	0.0348	18	0.0261	10	0.0109	14	0.0215
62	10	0.022	78	0.0692	96	0.076	50	0.0393	35	0.0451
63	9	0.0917	47	0.3093	47	0.2884	18	0.1548	26	0.2308
64	9	0.0945	10	0.0883	9	0.0753	11	0.1134	12	0.1087
65	5	0.044	7	0.0345	7	0.0384	5	0.0311	6	0.0529
66	10	0.0541	16	0.0646	40	0.2119	10	0.0477	14	0.0763
67	96	0.0163	116	0.017	116	0.0106	200	0.0256	86	0.013
68	79	0.0166	1306	0.3108	fail	fail	187	0.0315	109	0.0185
69	22	0.0015	180	0.0157	194	0.0062	8117	0.3526	26	0.0019
70	49	0.0044	735	0.0449	736	0.0229	4249	0.1963	57	0.0046
71	24	0.002	27	0.002	27	0.0027	30	0.0046	23	0.002
72	30	0.003	43	0.0038	43	0.005	34	0.0111	29	0.0031

(Continued on next page)

TABLE II – *Continued*

Problem	SpMMSIS		JYJLL		MFR		SCD		NPRP	
	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU
73	4	2.84E-04	5	5.72E-04	5	3.36E-04	5	6.25E-04	5	0.000321
74	11	9.83E-04	4710	0.205	6315	0.1435	11	0.0086	14	0.001
75	97	0.0043	10	0.0013	10	6.97E-04	77	0.0089	10	0.000656
76	104	0.0073	10	0.0012	10	6.85E-04	105	0.0102	10	0.00064
77	71	0.0097	38	0.0072	38	0.0054	67	0.0087	38	0.0048
78	74	0.0097	40	0.0041	40	0.0056	75	0.015	40	0.0059
79	317	0.1454	131	0.0532	131	0.0504	234	0.0883	639	0.2417
80	399	0.1821	137	0.0615	137	0.0489	255	0.1001	716	0.265
81	33	0.0231	255	0.141	388	0.1794	46	0.0189	50	0.0216
82	33	0.0269	3690	0.997	490	0.1991	40	0.0205	50	0.0349
83	58	0.1142	1149	3.5478	1217	3.475	87	0.1231	75	0.1262
84	61	0.1237	1763	4.102	1132	3.246	120	0.1653	76	0.14
85	10	5.93E-04	20	0.0022	20	0.001	8	8.67E-04	19	0.0011
86	9	5.52E-04	51	0.0056	51	0.0025	12	0.0092	10	0.0136
87	131	0.008	272	0.0166	272	0.0111	4334	0.2208	1234	0.0551
88	152	0.0106	273	0.0181	273	0.0138	1230	0.0778	1198	0.063
89	1	1.55E-04	1	2.58E-04	1	0.0011	1	0.0011	1	0.000621
90	1	1.75E-04	1	0.003	1	0.0015	1	0.0056	1	0.0085
91	405	0.0225	fail	fail	fail	fail	4295	0.2079	1293	0.0402
92	122	0.0085	33	0.0029	33	0.0015	1346	0.0801	578	0.039
93	15	9.47E-04	16	0.0015	16	8.83E-04	55	0.006	14	0.0015
94	27	0.0033	24	0.0027	25	0.002	105	0.0119	47	0.067
95	1	0.0068	1	0.0083	1	0.0075	1	0.0052	1	0.0095
96	1	0.0062	1	0.0072	1	0.0043	1	0.0151	1	0.1767
97	53	0.0076	25	0.005	25	0.0054	45	0.0065	25	0.0059
98	72	0.0142	41	0.0048	41	0.0055	77	0.0181	41	0.8284

TABLE III: Numerical results of the HMMSIS, HuS, GN, HDY, LS-CD and HJHJ methods.

Problem	HMMSIS		HuS		GN		HDY		LS-CD		HJHJ	
	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU
1	16	0.4548	99	2.4769	99	2.51	100	2.7059	100	2.6619	22	0.604
2	39	1.061	32	0.8199	318	7.636	37	0.9531	116	2.8378	50	1.3286
3	16	4.0989	107	26.9841	107	27.0696	108	27.4512	108	27.097	23	5.8905
4	22	6.4689	61	15.2301	74	18.3691	57	14.2066	62	15.4045	33	8.3382
5	16	0.0923	44	0.178	63	0.2525	44	0.1853	44	0.2029	82	0.3409
6	33	0.1686	32	0.1302	28	0.1191	31	0.1275	32	0.1271	67	0.2699
7	16	0.2904	45	0.7621	64	1.073	45	0.7681	45	0.75	88	1.5143
8	23	0.4214	27	0.4756	52	0.8988	31	0.5576	27	0.4566	39	0.6935
9	8	0.0457	9	0.0316	15	0.053	9	0.0384	9	0.0369	9	0.048
10	8	0.0449	9	0.0457	fail	fail	fail	fail	fail	fail	20	0.1996
11	13	0.3884	105	2.9226	105	2.9519	105	2.9001	105	3.0178	102	2.8863
12	12	0.3969	81	2.1649	81	2.177	81	2.1602	81	2.1578	81	2.3531
13	14	3.829	87	24.5308	87	24.6839	87	24.5671	87	24.0999	87	23.5478
14	12	3.3952	87	23.4729	87	23.5523	87	23.5355	87	23.5402	87	23.6365
15	35	0.1022	91	0.2026	91	0.1935	91	0.1892	91	0.1956	107	0.2716
16	153	0.3601	131	0.2781	210	0.4461	175	0.3667	268	0.5812	234	0.5488
17	17	0.0635	19	0.0475	19	0.0504	19	0.0479	19	0.0464	19	0.0839
18	42	0.1343	39	0.1017	36	0.0916	38	0.0966	39	0.097	35	0.1138
19	72	0.2311	74	0.237	74	0.2315	73	0.2475	74	0.2358	72	0.2345
20	121	0.4071	163	0.4696	163	0.4662	165	0.485	165	0.4638	158	0.446
21	68	1.0224	676	9.7927	676	9.7493	676	9.988	676	9.8306	544	8.0939
22	171	2.4796	44	0.6683	616	8.9069	44	0.6395	44	0.6697	58	0.874
23	88	2.4783	782	23.3245	782	23.8477	782	35.4964	782	25.0315	644	17.6239
24	92	2.9348	782	21.2006	782	21.136	782	21.2476	782	21.092	644	17.2066
25	3	0.0188	5	0.0302	5	0.031	5	0.0332	5	0.0297	5	0.0394

(Continued on next page)

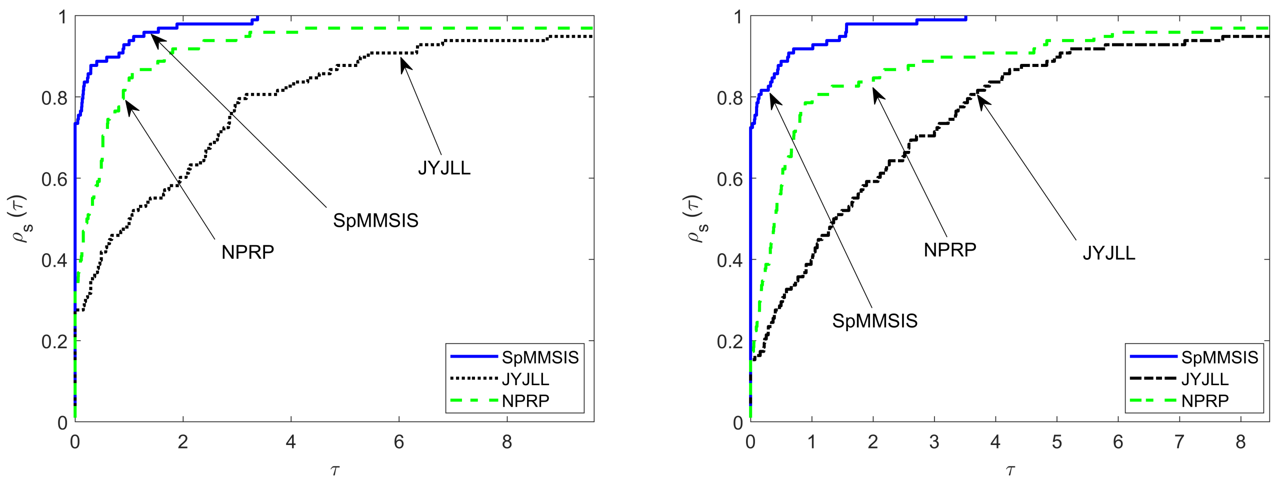
TABLE III – *Continued*

Problem	HMMSIS		HuS		GN		HDY		LS-CD		HJHJ	
	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU
26	3	0.0185	5	0.0252	5	0.033	5	0.0307	5	0.0316	5	0.0334
27	3	0.0239	5	0.0303	5	0.0366	5	0.0369	5	0.035	5	0.0297
28	3	0.0235	5	0.0359	5	0.0326	5	0.0348	5	0.0368	5	0.0357
29	7	0.0481	16	0.0937	16	0.0961	16	0.094	16	0.1695	13	0.0782
30	6	0.0412	6	0.0471	6	0.0466	6	0.0433	6	0.0447	8	0.0491
31	10	0.2142	9	0.1859	10	0.2068	9	0.1933	9	0.1868	13	0.262
32	7	0.1528	17	0.3097	17	0.3248	17	0.3247	17	0.3325	17	0.3301
33	42	0.1321	52	0.1152	52	0.1214	52	0.1219	52	0.1196	52	0.1484
34	29	0.0897	30	0.0741	30	0.0772	30	0.0764	30	0.0846	30	0.0915
35	188	0.9336	5892	44.9293	5892	37.8607	5892	36.9626	5892	37.674	5594	23.747
36	202	0.9828	5047	31.6929	6399	40.8155	4998	31.9683	5054	37.4891	5048	21.5379
37	9	0.042	28	0.0648	28	0.0639	28	0.0646	28	0.064	12	0.0521
38	16	0.066	36	0.0888	14	0.0362	36	0.0867	36	0.0882	74	0.2007
39	5	0.0222	9	0.0246	9	0.024	9	0.026	9	0.0242	9	0.0463
40	8	0.0366	11	0.03	11	0.032	11	0.0327	11	0.0301	10	0.0435
41	9	0.0441	12	0.0456	12	0.0332	12	0.0442	12	0.0352	11	0.0444
42	11	0.0503	9	0.0295	9	0.0273	9	0.0279	9	0.0291	11	0.0511
43	17	0.0655	11	0.0279	17	0.0428	11	0.0271	11	0.0297	13	0.0607
44	7	0.0338	14	0.0462	14	0.0476	14	0.0402	14	0.0393	14	0.0608
45	9	0.0449	fail	fail	14	0.2003	14	0.0467	13	0.04	fail	fail
46	10	0.0512	fail	fail	28	0.0754	fail	fail	27	0.0728	30	0.2636
47	12	0.0516	11	0.0305	11	0.03	11	0.0289	11	0.0351	11	0.0525
48	18	0.0746	18	0.0477	18	0.054	18	0.0478	18	0.0474	19	0.0918
49	56	0.1644	39	0.1533	23	0.0835	fail	fail	39	0.1206	21	0.0871
50	25	0.0865	41	0.1501	28	0.0917	29	0.103	28	0.1594	30	0.0932
51	7	0.0348	7	0.0341	9	0.0406	7	0.0328	7	0.0327	10	0.0529
52	6	0.0301	6	0.0301	6	0.0329	6	0.0201	6	0.03	9	0.0436
53	9	0.0441	9	0.0389	8	0.0395	9	0.0425	9	0.043	9	0.0491
54	11	0.0526	15	0.0713	8	0.0395	11	0.0542	15	0.0631	15	0.0639
55	3	0.0126	3	0.0152	3	0.014	3	0.0144	3	0.0145	3	0.0176
56	3	0.0155	3	0.0152	3	0.0159	3	0.016	3	0.0155	3	0.0183
57	1	0.0051	1	0.0054	1	0.0062	1	0.0059	1	0.0082	1	0.0121
58	5	0.0258	5	0.0261	5	0.0261	5	0.0259	5	0.0258	9	0.0497
59	11	0.0513	9	0.0417	20	0.0877	9	0.0405	9	0.0402	12	0.0567
60	11	0.0518	12	0.056	12	0.0553	12	0.0568	12	0.0491	12	0.0557
61	7	0.0484	18	0.1009	18	0.0868	18	0.099	18	0.1238	18	0.0943
62	11	0.0695	12	0.082	21	0.1773	12	0.0668	12	0.0634	15	0.0777
63	8	0.1656	47	0.8346	47	0.8379	47	0.9283	47	0.8251	47	0.8137
64	9	0.1741	43	0.7445	43	0.7624	43	0.7771	43	0.738	43	0.7219
65	5	0.0367	5	0.0213	6	0.0248	5	0.0204	5	0.0228	6	0.0516
66	7	0.0461	7	0.0305	9	0.0385	7	0.0355	7	0.0318	9	0.0671
67	71	0.1942	71	0.1631	71	0.1612	71	0.1603	71	0.1718	71	0.2199
68	65	0.1777	60	0.1482	60	0.139	60	0.1393	60	0.1406	62	0.1872
69	20	0.0657	9	0.0327	9	0.0218	12	0.0307	9	0.0328	61	0.1743
70	19	0.0678	49	0.1113	83	0.1776	29	0.0713	59	0.1331	36	0.123
71	22	0.0794	22	0.0851	22	0.0893	23	0.0932	23	0.0899	23	0.0905
72	24	0.0874	27	0.1212	27	0.107	27	0.1005	27	0.1135	27	0.106
73	4	0.0182	5	0.0243	5	0.0203	5	0.0241	5	0.0238	5	0.0351
74	10	0.0429	11	0.2056	15	0.0541	11	0.0491	11	0.0444	14	0.0571
75	24	0.0778	23	0.0692	23	0.132	20	0.0827	20	0.0751	23	0.0811
76	25	0.0796	23	0.1203	23	0.0933	22	0.0738	21	0.0862	23	0.0902
77	38	0.1117	38	0.0961	38	0.0957	38	0.0909	38	0.0967	39	0.1376
78	41	0.1224	41	0.1137	41	0.1017	41	0.102	41	0.1051	41	0.1303
79	313	1.8818	131	0.6837	131	0.6259	131	0.5597	131	0.5944	144	1.017
80	342	2.2016	137	0.5641	137	0.5707	137	0.6007	150	0.9507	151	0.8641
81	27	0.1164	55	0.173	35	0.126	16	0.0599	73	0.2517	64	0.2531
82	39	0.1577	27	0.0996	60	0.1715	32	0.1077	63	0.201	58	0.2245

(Continued on next page)

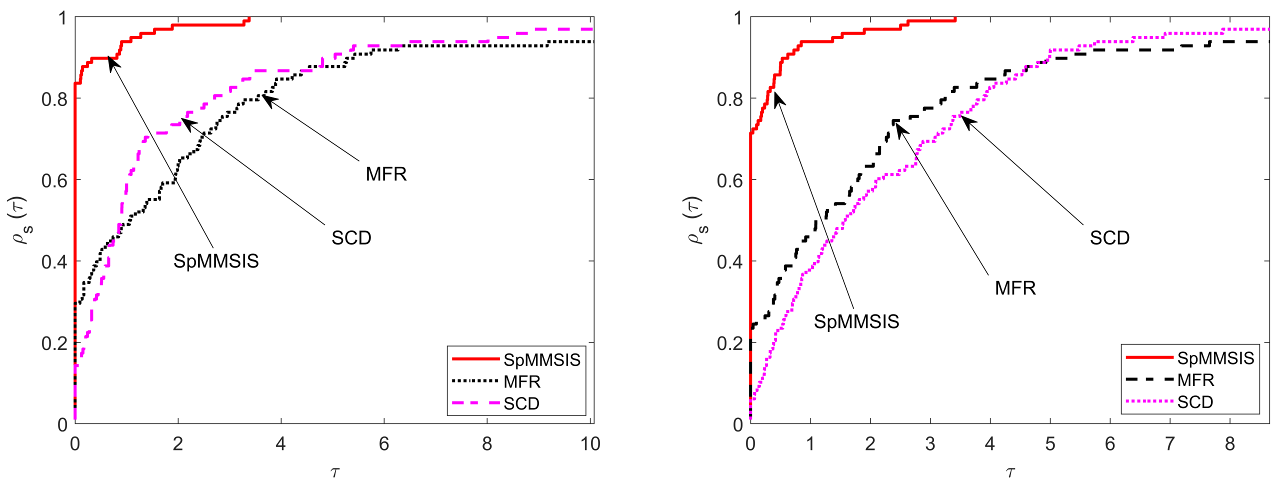
TABLE III – *Continued*

Problem	HMMSIS		HuS		GN		HDY		LS-CD		HJHJ	
	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU
83	73	0.6299	174	1.9825	93	0.7327	47	0.4501	48	0.4327	74	0.673
84	74	0.6557	66	0.8412	53	0.7323	49	0.5871	41	0.3818	109	0.9309
85	7	0.0328	20	0.0868	20	0.0912	20	0.0573	20	0.0984	20	0.0734
86	9	0.0432	19	0.0557	19	0.0706	19	0.079	19	0.0816	19	0.0754
87	48	0.1518	271	1.4919	271	1.7813	271	2.1317	271	1.8957	271	0.7489
88	60	0.1857	266	0.8987	266	0.9015	266	1.5651	266	2.1936	273	0.7322
89	1	0.0051	1	0.006	1	0.0047	1	0.0057	1	0.005	1	0.01
90	1	0.0058	1	0.0058	1	0.0055	1	0.007	1	0.0049	1	0.0059
91	173	0.4015	200	0.6743	277	1.3688	151	0.7055	330	1.5364	145	0.3672
92	27	0.0845	56	0.1771	63	0.2003	56	0.1891	56	0.1973	49	0.2294
93	10	0.042	10	0.0453	14	0.0639	10	0.0371	10	0.0452	14	0.0635
94	29	0.0902	21	0.0703	31	0.1	21	0.0725	21	0.0748	31	0.0913
95	1	0.0181	1	0.0316	1	0.0227	1	0.0226	1	0.0219	1	0.0367
96	1	0.0169	1	0.0183	1	0.0194	1	0.0192	1	0.0189	1	0.0204
97	26	0.0832	26	0.0829	26	0.0896	26	0.1235	26	0.3061	26	0.0835
98	42	0.1252	42	0.169	42	0.1519	41	0.1606	42	0.1695	45	0.1368



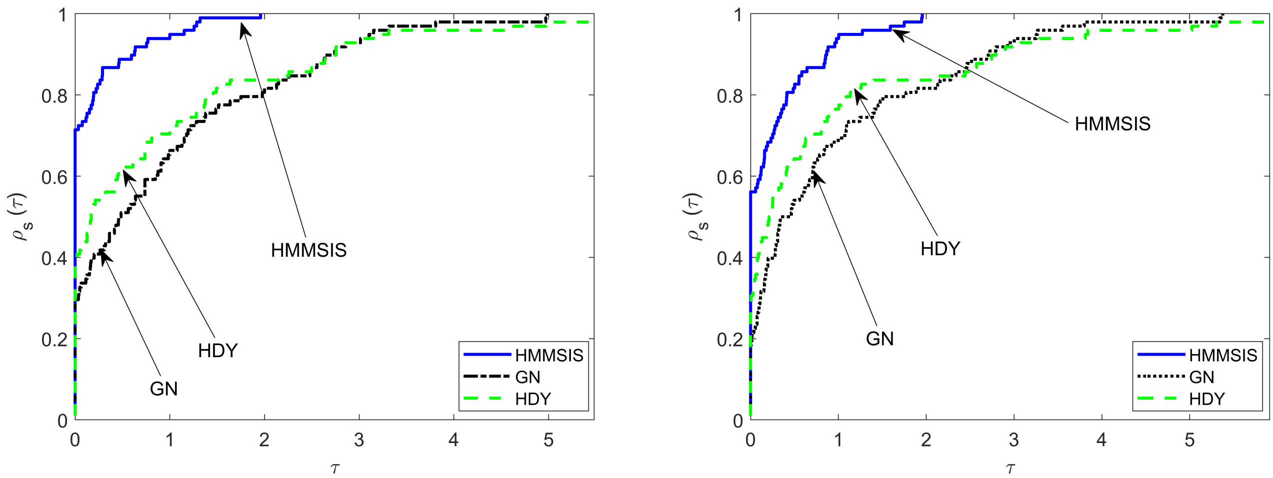
(a) Based on Number of Iterations. (b) Based on CPU time.

Fig. 1: Comparison performance of SpMMSIS, JYJLL, and NPRP methods.



(a) Based on Number of Iterations. (b) Based on CPU time.

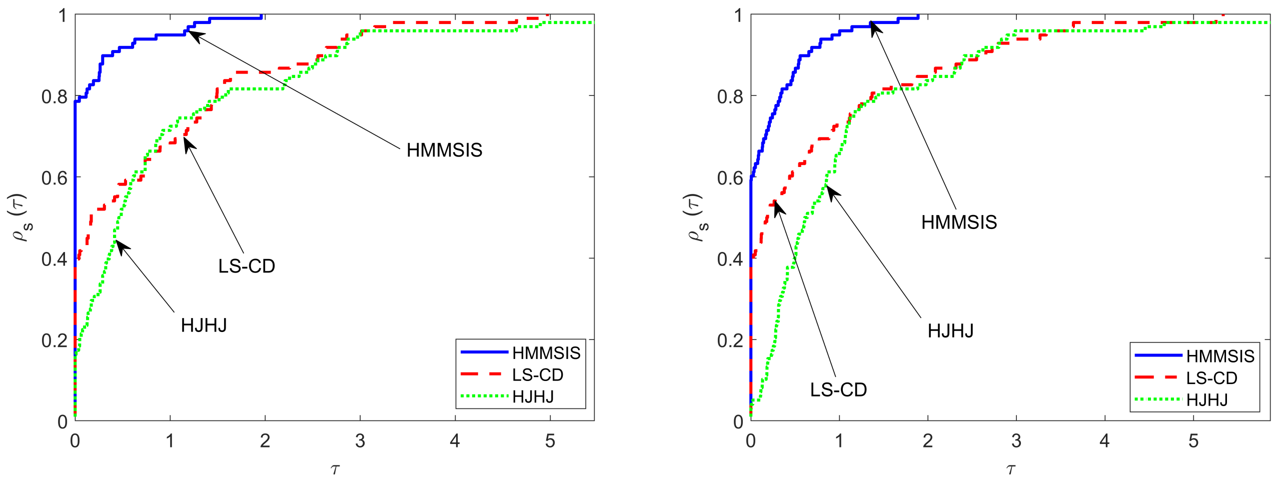
Fig. 2: Comparison performance of SpMMSIS, MFR, and SCD methods.



(a) Based on Number of Iterations.

(b) Based on CPU time.

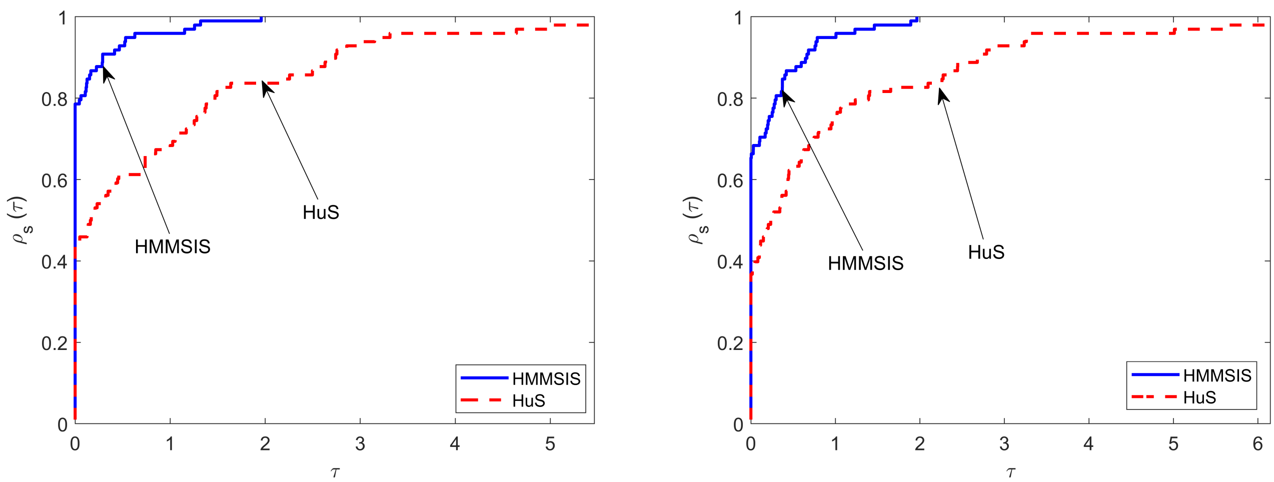
Fig. 3: Comparison performance of HMMSIS, GN, and HDY methods.



(a) Based on Number of Iterations.

(b) Based on CPU time.

Fig. 4: Comparison performance of HMMSIS, LS-CD, and HJHJ methods.



(a) Based on Number of Iterations.

(b) Based on CPU time.

Fig. 5: Comparison performance of HMMSIS and HuS methods.

Based on the numerical results in Table II and Table III, we can illustrate the performance profile curves of the all method, in this case we will use the performance profile proposed by Dolan and Moré [41]. We plot the performance profile curve using the formula as follows:

$$r_{p,s} = \frac{a_{p,s}}{\min\{a_{p,s} : p \in P \text{ and } s \in S\}},$$

$$\rho_s(\tau) = \frac{1}{n_p} size\{p \in P : r_{p,s} \leq \tau\},$$

where $r_{p,s}$ is the performance profile ratio used to compare the s solver performance method with the best performance for any p problem solver. $\rho_s(\tau)$ is the probability that the best possible ratio is a consideration for solvers.

The performance profile curve shows that the percentage of test issues that are successfully solved by each system is the right side of 98 problems; the left side of the figure shows the percentage of test problems for which the procedure is the fastest. The best solver is usually represented by the solver whose output profile plot is on the top right.

From Fig. 1 and Fig. 2 we can see that the proposed SpMMSIS method performs more efficient than the JYJLL, NPRP, MFR, and SCD methods both in terms of number of iterations and CPU time.

Meanwhile, the performance profile results of HMMSIS method are presented for both NOI and CPU time in Fig. 3 to Fig. 5. For each figure, the plot shows that the proposed HMMSIS method performed more efficient than the GN, HDY, LS-CD, HJHJ, and HuS methods both in terms of number of iterations and CPU time.

VI. Conclusion

In this article, we have proposed the new parameter of the spectral conjugate gradient method (SpMMSIS method) and a new coefficient for the hybrid conjugate gradient method (HMMSIS method). The new spectral conjugate gradient method satisfies the global convergence properties by using the strong Wolfe line search and has the required descent condition without relying on any line search. Similarly, the new hybrid conjugate gradient approach uses an exact line search to satisfy the sufficient descent condition and global convergence properties. The numerical results based on NOI and CPU time of 98 problems shows that the new method both SpMMSIS and HMMSIS conjugate gradient methods are very competitive and most efficient compared with other methods.

References

[1] E. Polak, "Optimization: Algorithms and Consistent Approximations," Springer, Berlin, 1997.

[2] J. Nocedal, and S. J. Wright, "Numerical Optimization," Springer Science & Business Media, New York, 2006.

[3] M. R. Hestenes, and E. Stiefel, "Methods of Conjugate Gradients for Solving Linear Systems," Journal of Research of the National Bureau of Standards, vol. 49, no. 6, pp. 409-436, 1952.

[4] R. Fletcher and C. M. Reeves, "Function minimization by conjugate gradients," The Computer Journal, vol. 7, no. 2, pp. 149-154, 1964.

[5] R. Fletcher, "Practical methods of optimization," john wiley & sons, 2013.

[6] Y. Liu, and C. Storey, "Efficient Generalized Conjugate Gradient Algorithms, Part 1: Theory," Journal of Optimization Theory and Applications, vol. 69, no. 1 ,pp. 129-137, 1991.

[7] Y. H. Dai, and Y. Yuan, "A Nonlinear Conjugate Gradient Method with A Strong Global Convergence Property, SIAM Journal on Optimization, vol. 10, no. 1, 177-182, 1999.

[8] E. Polak, and G. Ribiere, "Note sur la convergence de méthodes de directions conjuguées," ESAIM: Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique, vol. 3, no. R1, pp. 35-43, 1969.

[9] Z. Wei, S. Yao, and L. Liu, "The convergence properties of some new conjugate gradient methods," Applied Mathematics and Computation, vol. 183, no. 2, pp. 1341-1350, 2006.

[10] L. Zhang, "An improved Wei-Yao-Liu nonlinear conjugate gradient method for optimization computation," Applied Mathematics and Computation, vol. 215, no. 6, pp. 2269-2274, 2009.

[11] R. Pytlak, "Conjugate gradient algorithms in nonconvex optimization," Springer Science & Business Media, vol. 89, 2008.

[12] J. Barzilai, and J. M. Borwein, "Two point step size gradient methods," IMA journal of numerical analysis, vol. 8, no. 1, pp. 141-148, 1988.

[13] M. Raydan, "The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem," SIAM Journal on Optimization, vol. 7, no. 1. pp. 26-33, 1997.

[14] E. G. Birgin, and J. M. Martinez, "A spectral conjugate gradient method for unconstrained optimization," Applied Mathematics and optimization, vol. 43, no. 2, pp. 117-128, 2001.

[15] L. Zhang, W. Zhou, and D. Li, "Global convergence of a modified Fletcher-Reeves conjugate gradient method with Armijo-type line search," Numerische Mathematik, vol. 104, no. 4, pp. 561-572, 2006.

[16] J. Liu, and Y. Jiang, "Global convergence of a spectral conjugate gradient method for unconstrained optimization," Abstract and Applied Analysis, vol. 2012, Article ID 758287, 2012.

[17] J. Jian, L. Yang, X. Jiang, P. Liu, and M. Liu, "A Spectral Conjugate Gradient Method with Descent Property," Mathematics, vol. 8, no. 2, p. 280, 2020.

[18] Y. H. Dai, "Nonlinear conjugate gradient methods," Wiley Encyclopedia of Operations Research and Management Science, 2010.

[19] D. Touati-Ahmed, and C. Storey, "Efficient Hybrid Conjugate Gradient Techniques," Journal of Optimization Theory and Applications, vol. 64, no. 2, pp. 379-397, 1990.

[20] Y. F. Hu, and C. Storey, "Global Convergence Result for Conjugate Gradient Methods," Journal of Optimization Theory and Applications, vol. 71, no. 2, pp. 399-405, 1991.

[21] J. C. Gilbert, and J. Nocedal, "Global Convergence Properties of Conjugate Gradient Methods for Optimization," SIAM Journal on Optimization, vol. 2, pp. 21-42, 1992.

[22] Y. H. Dai, and Y. X. Yuan, "An Efficient Hybrid Conjugate Gradient Method for Unconstrained Optimization," Annals of Operations Research, vol. 103, no. 1-4, pp. 33-47, 2001.

[23] X. Li, and X. Zhao, "A Hybrid Conjugate Gradient Method for Optimization Problems," Natural Science, vol. 03, no. 01, pp. 85-90, 2011.

[24] J. Jian, L. Han, and X. Jiang, "A Hybrid Conjugate Gradient Method with Descent Property for Unconstrained Optimization," Applied Mathematical Modelling, vol. 39, no. 3-4,pp. 1281-1290, 2015.

[25] Issam A. R. Moghrabi, "A New Preconditioned Conjugate Gradient Method for Optimization," IAENG International Journal of Applied Mathematics, vol. 49, no.1, pp29-36, 2019.

[26] Omid Kardani, Andrei V. Lyamin, and Kristian Krabbenhoft, "A Comparative Study of Preconditioning Techniques for Large Sparse Systems Arising in Finite Element Limit Analysis," IAENG International Journal of Applied Mathematics, vol. 43, no. 4, pp195-203, 2013.

[27] Dharminder Kumar, Sangeeta Gupta, and Parveen Sehgal, "Improved Training of Predictive ANN with Gradient Techniques," Lecture Notes in Engineering and Computer Science: Proceedings of The International MultiConference of Engineers and Computer Scientists 2014, IMECS 2014, 12-14 March, 2014, Hong Kong, pp394-399.

[28] Z. Wan, S. Zhang, and Y. Wang, "Penalty algorithm based on conjugate gradient method for solving portfolio management problem," Journal of Inequalities and Applications, vol. 2009, pp. 1-16, 2009.

[29] J. Cao, and J. Wu, "A conjugate gradient algorithm and its applications in image restoration," Applied Numerical Mathematics, vol. 152, pp. 243-252, 2020.

[30] K. Yang, Gh. Jiang, Q. Qu, Hf. Peng, and Xw. Gao, "A new modified conjugate gradient method to identify thermal conductivity of transient non-homogeneous problems based on radial integration boundary element method," International Journal of Heat and Mass Transfer, vol. 133, pp. 669-676, 2019.

[31] J. K. Liu, Y. M. Feng, and L. M. Zou, "A spectral conjugate gradient method for solving large-scale unconstrained optimization," Computers & Mathematics with Applications, vol. 77, no. 3, pp. 731-739, 2019.

[32] M. Malik, M. Mamat, S. S. Abas, I. M. Sulaiman, Sukono, and A. T. Bon, "Solving Unconstrained Minimization Problems with a New Hybrid Conjugate Gradient Method," Proceedings of the 5th

NA International Conference on Industrial Engineering and Operations Management Detroit, Michigan, USA, ID 497, 2020.
[33] M. Malik, M. Mamat, S. S. Abas, I. M. Sulaiman, Sukono, and A. T. Bon, “Comparison of Conjugate Gradient Method on Solving Unconstrained Optimization Problems,” Proceedings of the 5th NA International Conference on Industrial Engineering and Operations Management Detroit, Michigan, USA, ID 500, 2020.
[34] Maulana Malik, Mustafa Mamat, Siti Sabariah Abas, Ibrahim Mohammed Sulaiman, and Sukono, “A New Coefficient of the Conjugate Gradient Method with the Sufficient Descent Condition and Global Convergence Properties,” Engineering Letters, vol. 28, no.3, pp704-714, 2020.
[35] G. Zoutendijk, “Nonlinear programming, computational methods,” Integer and nonlinear programming, pp. 37-86, 1970.
[36] N. Andrei, “An unconstrained optimization test functions collection,” Adv. Model. Optim, vol. 10, no. 1, pp. 147-161, 2008.
[37] M. Malik, M. Mamat, S. S. Abas, and Sukono, “Convergence analysis of a new coefficient conjugate gradient method under exact line search,” International Journal of Advanced Science and Technology, vol. 29, no. 5, pp. 187-198, 2020.
[38] M. Malik, S. S. Abas, M. Mamat, Sukono, and I. S. Mohammed, “A new hybrid conjugate gradient method with global convergence properties,” International Journal of Advanced Science and Technology, vol. 29, no. 5, pp. 199-210, 2020.
[39] M. Malik, M. Mamat, S. S. Abas, I. M. Sulaiman, and Sukono, “A new spectral conjugate gradient method with descent condition and global convergence property for unconstrained optimization,” J. Math. Comput. Sci., vol. 10, No. 5, pp. 2053-2069, 2020.
[40] M. Malik, M. Mamat, S. S. Abas, I. M. Sulaiman, and Sukono, “A new modification of NPRP conjugate gradient method for unconstrained optimization ,” Advances in Mathematics: Scientific Journal, vol. 9, No. 7, pp. 4955-4970, 2020.
[41] E. D. Dolan, and J. J. Moré, “Benchmarking optimization software with performance profiles,” Mathematical Programming, vol. 91, no. 2, pp. 201-213, 2002.

Maulana Malik (Member), is currently a Lecturer at the Department of Mathematics, Universitas Indonesia, Indonesia since 2016. He received his B.Sc. (2009) and M.Sc. (2012) in Mathematics from Universitas Indonesia (UI), Indonesia and he is currently (2019-present) a Ph.D. student at Universiti Sultan Zainal Abidin (UniSZA), Terengganu, Malaysia. His current research focuses on optimization include the conjugate gradient (CG), hybrid CG, spectral CG, and three-term CG methods.

Mustafa Mamat is currently a Professor in Faculty of Informatics and Computing at the Universiti Sultan Zainal Abidin since 2013. He obtained his Ph.D. from the UMT in 2007 with specialization in optimization. He was appointed as a Senior Lecturer in 2008 and the as an Associate Professor in 2010 also the UMT. To date, he has published more than 407 research paper in various international journals and conferences. His research interest in applied mathematics, with a field of concentration of optimization include conjugate gradient, steepest descent methods, Broydens family and quasi-Newton methods.

Siti Sabariah Abas is a Lecturer at Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin (UniSZA) Malaysia. She obtained her Ph.D. from the Universiti Sains Malaysia (USM) in 2016 with field in numerical analysis include the fluid dynamics.

Ibrahim Mohammed Sulaiman is currently a post-doctoral researcher at Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin (UniSZA) Malaysia, from 2019 till date. He obtained his Ph.D. from UniSZA in 2018 specializing in the field of fuzzy systems. He has published research papers in various international journals and attended international conferences. His research interest includes numerical research, Fuzzy nonlinear systems and unconstrained optimization.

Sukono (Member) is a Lecturer in the Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Padjadjaran. Masters in Actuarial Sciences at Institut Teknologi Bandung, Indonesia in 2000, and Ph.D. in Financial Mathematics at the Universitas Gadjah Mada, Yogyakarta Indonesia in 2011. Currently serves as Head of Master Program in Mathematics, the field of applied mathematics, with a field of concentration of financial mathematics and actuarial sciences.