J. Ris. & Ap. Mat. Vol. 5, No. 1 (2021), pp. 47–59.
Jurnal Riset dan Aplikasi Matematika
e-ISSN 581-0154
URL: journal.unesa.ac.id/index.php/jram

A New Hybrid PRP-MMSIS Conjugate Gradient Method and Its Application in Portfolio Selection

SINDY DEVILA[1*], MAULANA MALIK[1], WED GIYARTI[2]

[1]Department of Mathematics, Universitas Indonesia, Depok 16424, Indonesia
[2]Mathematics Education Study Program, Universitas Islam Negeri Sunan Kalijaga, Yogyakarta 55281, Indonesia

*Corresponding author: sindy@sci.ui.ac.id

ABSTRACT

In this paper, we propose a new hybrid coefficient of conjugate gradient method (CG) for solving unconstrained optimization model. The new coefficient is combination of the PRP (Polak-Ribiére-Polyak) [1, 2] and a part of MMSIS (Malik-Mustafa-Sabariah-Ibrahim-Sukono) [3] coefficients. Under exact line search, the search direction of new method satisfies the sufficient descent condition and based on certain assumption, we establish the global convergence properties. Using some test functions, numerical results show that the proposed method is efficient than MMSIS method. Besides, the new method can be used to solve portfolio selection problem .

Keywords: Conjugate gradient method, Exact line search, Sufficient descent condition, Global convergence, Portfolio selection

1 Introduction

In this paper, we present a new hybrid coefficient of conjugate gradient (CG) method for solving unconstrained optimization problem

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

where $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable function and its gradient is defined by $g(x) = \nabla f(x)$. CG methods are among the effective methods for solving large-scale problems.

The conjugate gradient method works by constructing sequence $\{x_k\}$ with iterative formula

$$x_{k+1} = x_k + \alpha_k d_k, \ k = 0, 1, 2, ..., \tag{2}$$

where α_k is the step size which in this paper, we use the rule of exact line search

$$f(x_k + \alpha_k d_k) := \min_{\alpha \geq 0} f(x_k + \alpha d_k) \tag{3}$$

and d_k is the search direction formulated by

$$d_k := \begin{cases} -g_k, & \text{if } k = 0, \\ -g_k + \beta_k d_{k-1}, & \text{if } k > 0, \end{cases} \tag{4}$$

2000 Mathematics Subject Classification: 65K10, 90C52, 90C26.
Submitted: 13-02-21, reviewed: 11-03-21, accepted: 30-04-21

where β_k is the gradient conjugation coefficient which the researchers are currently making modifications to as a computational improvement of the existing method [4]. Some of the well-known conjugate gradient coefficients are the Hestenes-Stiefel (HS) [5], Polak-Ribiére-Polyak (PRP) [1, 2], Liu-Storey (LS) [6], Fletcher-Reeves (FR) [7], conjugate descent (CD) [8], and Dai-Yuan (DY) [9]. These coefficients are defined by the following formulas:

$$\beta_k^{HS} = \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}, \quad \beta_k^{PRP} = \frac{g_k^T y_{k-1}}{\|g_{k-1}\|^2}, \quad \beta_k^{LS} = \frac{g_k^T y_{k-1}}{-g_{k-1}^T d_{k-1}},$$

$$\beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}, \quad \beta_k^{CD} = \frac{\|g_k\|^2}{-d_{k-1}^T g_{k-1}}, \quad \beta_k^{DY} = \frac{\|g_k\|^2}{d_{k-1}^T y_{k-1}},$$

where $y_{k-1} = g_k - g_{k-1},\ \ g_k = g(x_k)$ and $\|\cdot\|$ is the Euclidean norm. One of the variants of this CG method is the hybrid CG method, which is defined as the coefficient is a combination of the existing CG coefficients. The popular for hybrid conjugate gradient method are Touati-Ahmed and Storey (TS) method [10], Hu and Storey (HuS) method [11], Gilbert and Nocedal (GN) method [12], and Dai and Yuan (hDY and LS-CD) method [13]:

$$\begin{aligned}
\beta_k^{TS} &= \begin{cases} \beta_k^{PRP}, & \text{if } 0 \le \beta_k^{PRP} \le \beta_k^{FR} \\ \beta_k^{FR}, & \text{otherwise} \end{cases}, \\
\beta_k^{HuS} &= \max\left\{0, \min\left\{\beta_k^{PRP}, \beta_k^{FR}\right\}\right\}, \\
\beta_k^{GN} &= \max\left\{-\beta_k^{FR}, \min\left\{\beta_k^{PRP}, \beta_k^{FR}\right\}\right\}, \\
\beta_k^{hDY} &= \max\left\{0, \min\left\{\beta_k^{HS}, \beta_k^{DY}\right\}\right\}, \\
\beta_k^{LS-CD} &= \max\left\{0, \min\left\{\beta_k^{LS}, \beta_k^{CD}\right\}\right\}.
\end{aligned}$$

When proposing new methods, the researchers also show the sufficient descent condition and global convergence properties. This properties are characteristics of good computational. A method is said to fulfill the sufficient descent condition, if there exists a constant $c > 0$ such that for all k:

$$g_k^T d_k \le -c\|g_k\|^2, \tag{5}$$

and satisfies the global convergence properties, if

$$\liminf_{k\to\infty} \|g_k\| = 0.$$

For the FR method, Zoutendijk has proved the global convergence properties under the exact line search [14] and Al-Baali also established the global convergence properties under inexact line search [15]. The hybrid TS and HuS methods satisfies the descent condition and global convergence property under the inexact line search, and computational results are superior than the FR and PRP methods. For a description of other methods, we can see it in [3] and [16].

Recently, Malik et.al [3] have proposed the new coefficient of CG method, which it is modification of NPRP coefficient [17]. The new coefficient is symbolized by β_k^{MMSIS} and defined as follows:

$$\beta_k^{MMSIS} = \begin{cases} \dfrac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|}\left|g_k^T g_{k-1}\right| - \left|g_k^T g_{k-1}\right|}{\|d_{k-1}\|^2} & \text{, if } \|g_k\|^2 > \left(\frac{\|g_k\|}{\|g_{k-1}\|} + 1\right)\left|g_k^T g_{k-1}\right|, \\ 0 & \text{, otherwise.} \end{cases} \tag{6}$$

For the MMSIS method, the sufficient descent condition is satisfied under exact and strong line searches. Likewise, the MMSIS method satisfies the global convergence properties under exact and strong Wolfe line searches with parameter $\sigma \in (0,1/8)$. Numerical experiments shows that the MMSIS method is efficient than FR, CD, and DY methods. For other references about the CG method can refer to [16, 18, 19, 20, 21, 22, 23, 24].

Motivated by the MMSIS and GN methods, we propose a new hybrid CG coefficient for solving problem (1). The new coefficient is a combination of a part the MMSIS and PRP coefficients. Furthermore, we will establish the sufficient descent condition and global convergence properties under exact line search. Numerical experiments is also presented to compare the efficiency computational and the application of new method is used for solving portfolio selection problem. In the next section, we will present the formula of new coefficient, algorithm, sufficient descent condition, and global convergence properties. In Section 3, numerical experiments is provided and in Section 4, we show an application in portfolio selection. Finally, the conclusion is presented in Section 5.

2 Algorithm and Convergence Analysis

In this section, we formulate a new hybrid coefficient and establish the sufficient descent condition, and global convergence properties under exact line search. The new coefficient is a combination of part the MMSIS and PRP coefficients which formulated as follows:

$$\beta_k^{HDMG} = \max\{\beta_k^{PRP}, \beta_k^{MMSIS*}\}, \tag{7}$$

where $\beta_k^{MMSIS*} = \dfrac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|}\left|g_k^T g_{k-1}\right| - \left|g_k^T g_{k-1}\right|}{\|d_{k-1}\|^2}$, and HDMG denotes Hybrid-Devila-Malik-Giyarti. The following algorithm describe the HDMG method.

Algorithm 1: (HDMG Method)
Step 1: Given initial point $x_0 \in \mathbb{R}^n$, $d_0 = -g_0$, stopping criteria ε, and set $k := 0$.
Step 2: If $\|g_k\| \le \varepsilon$, then stop. x_k is optimal point. Otherwise, go to next step.
Step 3: Compute $\beta_k^{HDMG} = \max\{\beta_k^{PRP}, \beta_k^{MMSIS*}\}$.
Step 4: Compute the search direction $d_k = -g_k + \beta_k^{HDMG} d_{k-1}$.
Step 5: Compute the step size α_k by using exact line search (3).
Step 6: Update new point for $k := k+1$ by formula (2) and go to Step 2.

The following lemma show that the search direction d_k under exact line search satisfies the sufficient descent condition.

Lemma 2.1. *Suppose that a CG method with search direction* (4)*, α_k is computed by using exact line search* (3)*, and β_k is computed by using* (7)*, then, for all $k \ge 0$ the condition* (5) *is satisfied.*

PROOF. According to (4), we have $d_0 = -g_0$, furthermore $g_0^T d_0 = -g_0 g_0 = -\|g_0\|^2$. Thus, for $k = 0$ the condition (5) fulfill. Now, for $k \ge 1$, we will show the condition (5) is satisfied. By multiplying (4) with g_k^T, we obtain

$$g_k^T d_k = -g_k^T g_k + \beta_k^{HDMG} g_k^T d_{k-1} = -\|g_k\|^2 + \beta_k^{HDMG} g_k^T d_{k-1}.$$

Since α_k is computed by exact line search, it implies $g_k^T d_{k-1} = 0$. Thus, we have $g_k^T d_k = -\|g_k\|^2$. Hence, the condition (5) fulfill. The proof is completed. □

To establish the global convergence properties, we need to simplify the β_k^{HDMG}. See the following lemma.

Lemma 2.2. *The value of β_k^{HDMG} must be one of $\beta_k^{HDMG} \le \dfrac{\|g_k\|^2}{\|d_{k-1}\|^2}$ or $\beta_k^{HDMG} \le \dfrac{\|g_k\|^2}{\|g_{k-1}\|^2}$ or $\beta_k^{HDMG} = 0$.*

PROOF. From (7), we have three cases.

- **Case 1**: if $\beta_k^{PRP} < \beta_k^{MMSIS*}$, we obtain

$$\beta_k^{HDMG} = \beta_k^{MMSIS*} = \frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|}\left|g_k^T g_{k-1}\right| - \left|g_k^T g_{k-1}\right|}{\|d_{k-1}\|^2} \leq \frac{\|g_k\|^2}{\|d_{k-1}\|^2}.$$

- **Case 2**: if $\beta_k^{PRP} > \beta_k^{MMSIS*}$, we obtain

$$\beta_k^{HDMG} = \beta_k^{PRP} = \frac{g_k^T(g_k - g_{k-1})}{\|g_{k-1}\|^2} = \frac{\|g_k\|^2 - g_k^T g_{k-1}}{\|g_{k-1}\|^2} \leq \frac{\|g_k\|^2}{\|g_{k-1}\|^2}.$$

- **Case 3**: if $\beta_k^{PRP} = \beta_k^{MMSIS*} = 0$, we obtain

$$\beta_k^{HDMG} = 0.$$

The proof is finished. □

The following assumption is needed to establish the convergence properties of HDMG method.

Assumption 2.3. *(A1) The level set $\mathbb{Y} = \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}$ at x_0 is bounded. (A2) In any neighborhood $\mathbb{H}_0$ of $\mathbb{H}$, the objective function f is differentiable and continuous, and its gradient $g(x)$ is Lipschitz continuous in $\mathbb{H}_0$, so, there exist a constant $L > 0$ such that $\|g(x) - g(y)\| \leq L\|x - y\|$, for all $x, y \in \mathbb{H}_0$.*

Based on the assumption, Zoutendijk [14] has proven the following lemma which is necessary to prove the global convergence.

Lemma 2.4. *Suppose that Assumption 2.3 hold. Consider any conjugate gradient method of the form* (2) *and* (4)*, where α_k satisfy the exact line search* (3)*. Then the following conditions, so, called Zoutendijk conditions hold:*

$$\sum_{k=0}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < \infty.$$

The following theorem is global convergence theorem for HDMG method.

Theorem 2.5. *Suppose that the sequence $\{x_k\}$ is generated by Algorithm 1. Assume that Assumption 2.3 hold. Then we have*

$$\liminf_{k\to\infty} \|g_k\| = 0. \tag{8}$$

PROOF. Assume the opposite, i.e, (8) is not true, hence there exists a constant $z > 0$ such that

$$\|g_k\| \geq z, \ \forall k \geq 0,$$

it means that

$$\frac{1}{\|g_k\|^2} \leq \frac{1}{z^2}, \ \forall k \geq 0, \ \|g_k\| \neq 0. \tag{9}$$

From (4), we know that

$$d_k + g_k = \beta_k^{HDMG} d_{k-1}.$$

By squaring both sides of the equation, we have

$$\|d_k\|^2 = \left(\beta_k^{HDMG}\right)^2 \|d_{k-1}\|^2 - 2\beta_k^{HDMG} g_k^T d_k - \|g_k\|^2. \tag{10}$$

Dividing both sides of (10) by $(g_k^T d_k)^2$, we obtain

$$\begin{aligned}\frac{\|d_k\|^2}{(g_k^T d_k)^2} &= \frac{\left(\beta_k^{HDMG}\right)^2 \|d_{k-1}\|^2}{(g_k^T d_k)^2} - \frac{2}{g_k^T d_k} - \frac{\|g_k\|^2}{(g_k^T d_k)^2} \\ &= \frac{\left(\beta_k^{HDMG}\right)^2 \|d_{k-1}\|^2}{(g_k^T d_k)^2} - \left(\frac{1}{\|g_k\|} - \frac{\|g_k\|}{g_k^T d_k}\right)^2 + \frac{1}{\|g_k\|^2} \\ &\leq \frac{\left(\beta_k^{HDMG}\right)^2 \|d_{k-1}\|^2}{(g_k^T d_k)^2} + \frac{1}{\|g_k\|^2}. \end{aligned} \tag{11}$$

According to Lemma 2.2, we have three cases:

- **Case 1**. if $\beta_k^{HDMG} \leq \frac{\|g_k\|^2}{\|d_{k-1}\|^2}$, then from (11) and Lemma 2.1, we obtain

$$\frac{\|d_k\|^2}{(g_k^T d_k)^2} \leq \frac{\|g_k\|^4}{\|d_{k-1}\|^4} \frac{\|d_{k-1}\|^2}{(g_k^T d_k)^2} + \frac{1}{\|g_k\|^2} = \frac{1}{\|d_{k-1}\|^2} + \frac{1}{\|g_k\|^2}.$$

We know that $\frac{1}{\|d_k\|^2} \leq \frac{1}{\|g_k\|^2}$ (see Lemma 4 in [16]), then we get

$$\frac{\|d_k\|^2}{(g_k^T d_k)^2} \leq \frac{1}{\|g_{k-1}\|^2} + \frac{1}{\|g_k\|^2}.$$

From (9) and the inequality above, we have

$$\frac{\|d_k\|^2}{(g_k^T d_k)^2} \leq \frac{1}{z^2} + \frac{1}{z^2} = \frac{2}{z^2}.$$

Furthermore,

$$\sum_{k=0}^{n} \frac{(g_k^T d_k)^2}{\|d_k\|^2} \geq \sum_{k=0}^{n} \frac{z^2}{2} = \frac{n+1}{2} z^2.$$

By Taking $n \to \infty$, we get

$$\sum_{k=0}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} \geq \lim_{n\to\infty} \frac{n+1}{2} z^2 = +\infty.$$

This contradicts the Zoutendijk condition in Lemma 2.4. Hence, the HDMG method is global convergence.

- **Case 2**. if $\beta_k^{HDMG} \leq \frac{\|g_k\|^2}{\|g_{k-1}\|^2}$, then from (11) and Lemma 2.1, we obtain

$$\frac{\|d_k\|^2}{(g_k^T d_k)^2} \leq \frac{\|g_k\|^4}{\|g_{k-1}\|^4} \frac{\|d_{k-1}\|^2}{(g_k^T d_k)^2} + \frac{1}{\|g_k\|^2} = \frac{\|d_{k-1}\|^2}{\|g_{k-1}\|^4} + \frac{1}{\|g_k\|^2}. \tag{12}$$

By utilizing (12) recursively, we get

$$\frac{\|d_k\|^2}{(g_k^T d_k)^2} \leq \sum_{i=0}^{k} \frac{1}{\|g_i\|^2}.$$

Furthermore, from (9), we have

$$\frac{(g_k^T d_k)^2}{\|d_k\|^2} \geq \frac{z^2}{k+1}.$$

By taking summation of both sides, we obtain

$$\sum_{k=0}^{n} \frac{(g_k^T d_k)^2}{\|d_k\|^2} \geq \sum_{k=0}^{n} \frac{z^2}{k+1} = z^2 \sum_{k=0}^{n} \frac{1}{k+1}.$$

This implies,

$$\sum_{k=0}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} \geq +\infty.$$

This contradicts the Zoutendijk condition in Lemma 2.4. Hence, the HDMG method is global convergence.

- **Case 3**. if $\beta_k^{HDMG} = 0$, then from (11) and (9), we obtain

$$\frac{\|d_k\|^2}{(g_k^T d_k)^2} \leq \frac{1}{\|g_k\|^2} \leq \frac{1}{z^2}.$$

Therefore,

$$\frac{(g_k^T d_k)^2}{\|d_k\|^2} \geq z^2.$$

Thus,

$$\sum_{k=0}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} \geq +\infty.$$

This contradicts the Zoutendijk condition in Lemma 2.4. Hence, the HDMG method is global convergence. □

3 Numerical Experiments

In this section, we report the numerical experiments of HDMG method to compare with MMSIS method. The comparing done by using some test functions considered by Andrei [25], and Jamil and Yang [26]. Every test function, we use several initial points, and dimensions from 2 to 10,000. Most of the starting points used were considered by Andrei [25] and the rest were randomly. The numerical results are presented in Table 1 and obtained with the MATLAB code R2019a, and run using personal laptop; Intel Core i7 processor, 16 GB RAM, 64 bit Windows 10 Pro operating system. The stopping criterion $\|\mathbf{g}_k\| \leq \varepsilon$, where $\varepsilon = 10^{-6}$.

According to the numerical results in Table 1, we can compare between methods by illustrating the performance profile curves, in this paper we will use the performance profile proposed by Dolan and Moré [27]. We plot the performance profile curve using the formula as follows:

$$r_{p,s} = \frac{a_{p,s}}{\min\{a_{p,s} : p \in P \text{ and } s \in S\}}, \rho_s(\tau) = \frac{1}{n_p} size\{p \in P : \log_2 r_{p,s} \leq \tau\},$$

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. Generally, the best method is represented on the top curve.

Table 1: Numerical results for the MMSIS and HDMG methods.

Test Functions	Dimensions	Initial Points	MMSIS		HDMG	
			NOI	CPU	NOI	CPU
Ext White & Holst	1,000	(-1.2, 1,...,-1.2,1)	16	0.4396	11	0.2952
Ext White & Holst	1,000	(10,..,10)	30	0.804	37	1.0595
Ext White & Holst	10,000	(-1.2,1,..,-1.2,1)	17	4.363	12	3.0189
Ext White & Holst	10,000	(5,...,5)	25	6.3978	18	4.561
Ext Rosenbrock	1,000	(-1.2, 1,...,-1.2,1)	16	0.0754	21	0.0827
Extended Rosenbrock	1,000	(10,...,10)	30	0.1338	21	0.0716
Ext Rosenbrock	10,000	(-1.2,1,...,-1.2,1)	16	0.2979	21	0.3561
Ext Rosenbrock	10,000	(5,...,5)	26	0.4768	11	0.2092
Ext Freudenstein & Roth	4	(0.5,-2,0.5,-2)	9	0.0502	8	0.0277
Ext Freudenstein & Roth	4	(-5,-5,-5,-5)	7	0.0348	5	0.0164
Ext Beale	1,000	(1,0.8,...,1,0.8)	13	0.4102	10	0.2804
Ext Beale	1,000	(0.5,...,0.5)	12	0.3768	10	0.2736
Ext Beale	10,000	(-1,...,-1)	14	3.8664	9	2.5379
Ext Beale	10,000	(0.5,...,0.5)	12	3.3274	10	2.844
Ext Wood	4	(-3,-1,-3,-1)	203	0.4727	158	0.3283
Ext Wood	4	(5,5,5,5)	272	0.6224	278	0.5681
Raydan 1	10	(1,...,1)	21	0.0777	17	0.0435
Raydan 1	10	(10,...,10)	75	0.2111	39	0.101
Raydan 1	100	(-1,...,-1)	118	0.4247	73	0.2121
Raydan 1	100	(-10,...,-10)	194	0.6187	170	0.4862
Ext Tridiagonal 1	500	(2,...,2)	12	0.215	13	0.2004
Ext Tridiagonal 1	500	(10,...,10)	139	2.0098	16	0.2573
Ext Tridiagonal 1	1,000	(1,...,1)	12	0.3666	13	0.3797
Ext Tridiagonal 1	1,000	(-10,...,-10)	198	5.2889	15	0.4877
Diagonal 4	500	(1,...,1)	5	0.04	3	0.0185
Diagonal 4	500	(-20,...,-20)	5	0.0293	4	0.0273
Diagonal 4	1,000	(1,...,1)	5	0.0347	3	0.0183
Diagonal 4	1,000	(-30,...,-30)	5	0.0386	4	0.0303
Ext Himmelblau	1,000	(1,...,1)	9	0.0654	7	0.0453
Ext Himmelblau	1,000	(20,...,20)	6	0.0429	6	0.0452
Ext Himmelblau	10,000	(-1,...,-1)	10	0.227	9	0.1886
Ext Himmelblau	10,000	(50,...,50)	7	0.173	6	0.1399
FLETCHCR	10	(0,...,0)	80	0.2188	56	0.1301
FLETCHCR	10	(10,...,10)	39	0.1233	30	0.083
Ext Powel	100	(3,-1,0,1,...)	810	3.7825	3307	14.6059
Ext Powel	100	(5,...,5)	264	1.3266	3088	14.4368
NONSCOMP	2	(3,3)	8	0.0442	9	0.0238
NONSCOMP	2	(10,10)	15	0.0628	14	0.0405
Extended DENSCHNB	10	(1,...,1)	7	0.0368	5	0.0143
Extended DENSCHNB	10	(10,...,10)	10	0.0489	9	0.0264
Extended DENSCHNB	100	(10,...,10)	11	0.0461	9	0.0292
Extended DENSCHNB	100	(-50,...,-50)	11	0.0564	8	0.027
Extended Penalty	10	(1,2,...,10)	22	0.0824	27	0.0687
Extended Penalty	10	(-10,...,-10)	8	0.0377	7	0.0228
Extended Penalty	100	(5,...,5)	13	0.0613	7	0.0246

(Continued on next page)

Table 1 – *Continued*

Test Functions	Dimensions	Initial Points	MMSIS		HDMG	
			NOI	CPU	NOI	CPU
Extended Penalty	100	(-10,...,-10)	10	0.0401	9	0.0442
Hager	10	(1,...,1)	13	0.0552	12	0.0353
Hager	10	(-10,...,-10)	18	0.0746	18	0.051
Extended Maratos	10	(1.1, 0.1,...,1.1,0.1)	53	0.1465	35	0.1071
Extended Maratos	10	(-1,...,-1)	22	0.0764	12	0.0537
Six Hump Camel	2	(-1,2)	7	0.0247	6	0.0284
Six Hump Camel	2	(-5,10)	6	0.0207	6	0.0314
Three Hump Camel	2	(-1,2)	9	0.0293	9	0.0433
Three Hump Camel	2	(2,-1)	11	0.0325	12	0.059
Booth	2	(5,5)	4	0.0135	3	0.0145
Booth	2	(10,10)	4	0.0155	3	0.0164
Trecanni	2	(-1,0.5)	1	0.0064	1	0.0056
Trecanni	2	(-5,10)	5	0.0175	5	0.0264
Zettl	2	(-1,2)	11	0.0375	10	0.0457
Zettl	2	(10,10)	11	0.0303	8	0.0382
Shallow	1,000	(0,...,0)	8	0.042	7	0.0453
Shallow	1,000	(10,...,10)	11	0.0525	9	0.0613
Shallow	10,000	(-1,...,-1)	9	0.1707	8	0.2012
Shallow	10,000	(-10,...,-10)	9	0.194	9	0.1779
Generalized Quartic	1,000	(1,...,1)	5	0.0231	6	0.0342
Generalized Quartic	1,000	(20,...,20)	6	0.0365	10	0.0409
Quadratic QF2	50	(0.5,...,0.5)	87	0.1943	71	0.1626
Quadratic QF2	50	(30,...,30)	78	0.1845	64	0.1541
Leon	2	(2,2)	25	0.0608	11	0.0294
Leon	2	(8,8)	18	0.0446	33	0.0812
Generalized Tridiagonal 1	10	(2,...,2)	24	0.0679	22	0.0868
Generalized Tridiagonal 1	10	(10,...,10)	29	0.0829	27	0.108
Generlized Tridiagonal 2	4	(1,...,1)	4	0.013	4	0.0182
Generalized Tridiagonal 2	4	(10,...,10)	11	0.0363	10	0.0432
POWER	10	(1,...,1)	102	0.2045	21	0.0867
POWER	10	(10,...,10)	129	0.2674	25	0.0855
Quadratic QF1	50	(1,...,1)	69	0.1599	38	0.094
Quadratic QF1	50	(10,...,10)	85	0.1955	41	0.1126
Quadratic QF1	500	(1,...,1)	240	1.3077	131	0.585
Quadratic QF1	500	(-5,...,-5)	424	2.4118	137	0.6383
Ext Quad Penalty QP2	100	(1,...,1)	41	0.1438	26	0.0882
Ext Quad Penalty QP2	100	(10,...,10)	36	0.1196	26	0.0958
Ext Quad Penalty QP2	500	(10,...,10)	94	0.7527	33	0.2865
Ext Quad Penalty QP2	500	(50,...,50)	96	0.8037	26	0.2048
Ext Quad Penalty QP1	4	(1,1,1,1)	9	0.0251	6	0.0191
Ext Quad Penalty QP1	4	(10,10,10,10)	9	0.0354	9	0.0259
Quartic	4	(10,10,10,10)	114	0.2794	365	0.9105
Quartic	4	(15,15,15,15)	118	0.3283	197	0.4841
Matyas	2	(1, 1)	1	0.0039	1	0.0065
Matyas	2	(20, 20)	1	0.006	1	0.0049
Colville	4	(2,2,2,2)	357	0.6761	204	0.4159

(Continued on next page)

Table 1 – *Continued*

Test Functions	Dimensions	Initial Points	MMSIS		HDMG	
			NOI	CPU	NOI	CPU
Colville	4	(10,10,10,10)	58	0.1358	98	0.2037
Dixon and Price	3	(1, 1, 1)	15	0.0403	13	0.042
Dixon and Price	3	(10, 10, 10)	18	0.0482	49	0.116
Sphere	5,000	(1,...,1)	1	0.0169	1	0.0114
Sphere	5,000	(10,...,10)	1	0.0164	1	0.013
Sum Squares	50	(0,1,...,0,1)	49	0.1473	26	0.0694
Sum Squares	50	(10,...,10)	80	0.2309	42	0.1037

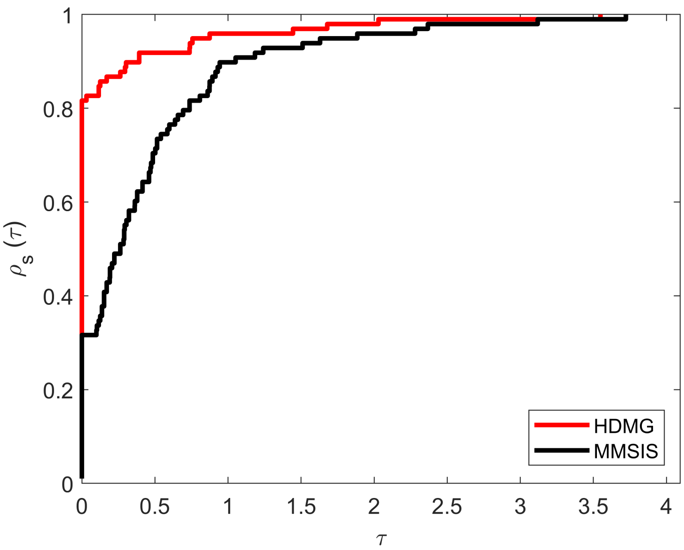

Figure 1: Performance Profile Based on Number of Iterations

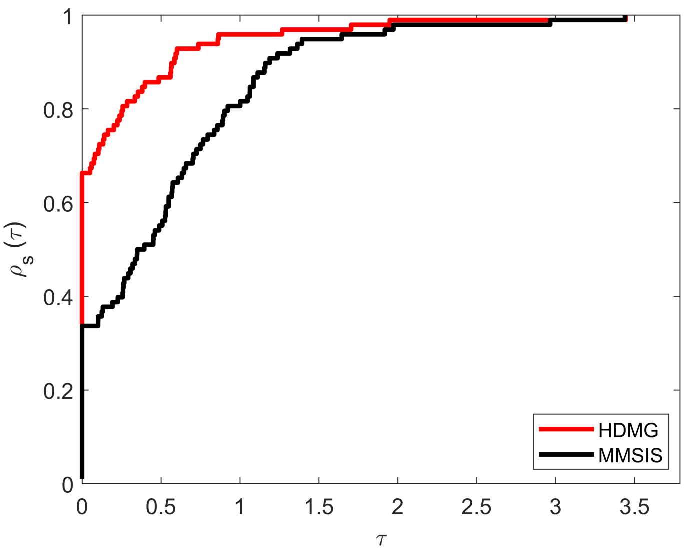

Figure 2: Performance Profile Based on CPU Time

In Fig. 1 and Fig. 2, the HDMG method is represented with red color, meanwhile, the MMSIS method is represented with black color. We can see that the curve of HDMG method always on the top MMSIS curve, so, based on Dolan and Moré rule, the HDMG method performs efficient than the MMSIS method both in terms of number of iterations and CPU time.

4 Application in Portfolio Selection

In this section, we present the application of CG method for solving portfolio selection problem. Consider there are M assets with return $r_1, \ldots, r_M$. Assume that expected return of asset denotes as $\mu^T = (\mu_1, \ldots, \mu_M)$ with $\mu_i = E[r_i]$, $i = 1, \ldots, M$, and covariance matrix denotes as $V = (\sigma_{ij})$ with $\sigma_{ij} = Cov(r_i, r_j)$, $i, j = 1, \ldots, M$. If proportional of asset is symbolized by $X^T = (x_1, x_2, \ldots, x_m)$, with subject to $\sum_{i=1}^{M} = 1$, then, the expected return of portfolio is defined as follows:

$$\mu_p = E[r_p] = \mu^T X,$$

and variance of portfolio is formulated by

$$\sigma_p^2 = Var(r_p) = X^T V X.$$

In portfolio theory, many investors want to maximum returns or minimal risk or even both. Moreover, there are also extreme investors who only care about maximizing return (ignoring risk) or minimizing risk (ignoring expected returns) [28]. In this article we only consider minimizing the risks and using only two stocks from the database http://finance.yahoo.com, over a period of 3 years (Jan 1, 2018 - Dec 31, 2020), i.e PT Bank Rakyat Indonesia (Persero) Tbk (BBRI), and PT Telekomunikasi Indonesia Tbk (TLKM). We just take the weekly closing price data and the return of each stock is defined as follows:

$$R_t = \frac{P_t - P_{t-1}}{P_{t-1}},$$

where P_t is the stock prices at time t and P_{t-1} is the stock prices at time $t-1$. According to the data of return, we can plot the movement price as in Figure 3.

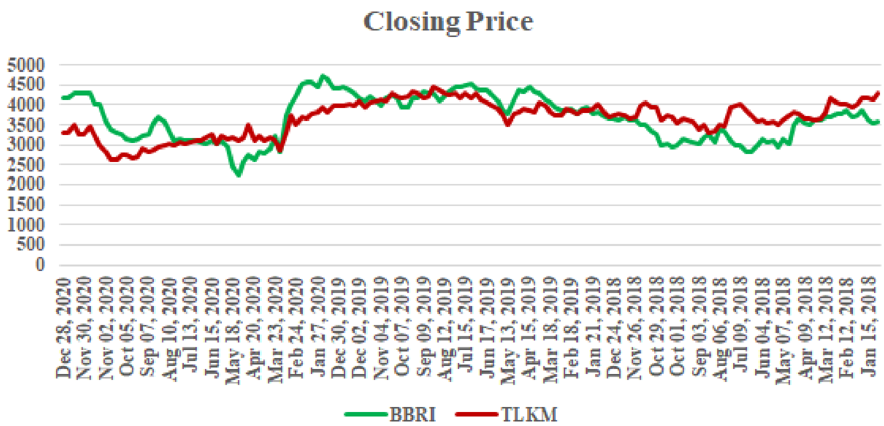

Figure 3: Closing Price of BBRI and TLKM in Currency IDR

The risk of our portfolio is defined as variance of the portfolio's return [28], so that the our problem can be written as:

$$\begin{cases} \text{minimize}: \ \sigma_p^2 = X^T V X, \\ \text{subject to}: \ \sum_{j=1}^{2} x_j = 1. \end{cases} \tag{13}$$

We need to change the problem (13) into an unconstrained optimization problem. Suppose that $x_2 = 1 - x_1$, then the problem (13) is an unconstrained problem as follows:

$$\min_{x_1 \in \mathbb{R}} (x_1 \quad 1 - x_1)^T V (x_1 \quad 1 - x_1). \tag{14}$$

The value of mean, variance, and covariance for BBRI and TLKM stocks are presented in Table 2.

Table 2: Mean, Variance and Covariance

Stocks	Mean	Variance	Covariance	BBRI	TLKM
BBRI	0.00033	0.00273	BBRI	0.00273	0.00091
TLKM	0.00247	0.00166	TLKM	0.00091	0.00166

Based on Table 2, we can be compute the objective function of (14) as follows:

$$f(x_1) = (0.00182x_1 + 0.00091)x_1 + (-0.00075x_1 + 0.00166)(1 - x_1)$$

Now, we solve this function by using HDMG CG method with any initial points, then, we obtain $x_1 = 0.2916$. Furthermore, the value of risk is $\sigma_p^2 = 0.00144$. Finally, We found that to minimize the risk we have to invest $x_1 = 29.16\%$ of the BBRI stock, and $x_2 = 70.84\%$ of the TLKM stock. The portfolio risk is 0.00144 and the expected portfolio return is 0.0018.

5 Conclusion

In this article, we presented a new hybrid CG method which is combination of PRP and a part of MMSIS coefficients. The new method satisfies the sufficient descent condition and global convergence properties under exact line search. Based on the numerical experiments, the new hybrid method is more efficient and robust than MMSIS method. Finally, the practical applicability of the hybrid method is also explored in risk optimization in portfolio selection problem.

References

[1] E. Polak and G. Ribiere, "Note sur la convergence de méthodes de directions conjuguées," *Revue française d'informatique et de recherche opérationnelle. Série rouge*, vol. 3, no. 16, pp. 35–43, 1969.

[2] B. T. Polyak, "The conjugate gradient method in extremal problems," *USSR Computational Mathematics and Mathematical Physics*, vol. 9, no. 4, pp. 94–112, 1969.

[3] M. Malik, M. Mamat, S. S. Abas, I. M. 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, pp. 704–714, 2020.

[4] J. Nocedal and S. Wright, *Numerical optimization*. Springer Science & Business Media, 2006.

[5] 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.

[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] R. Fletcher and C. M. Reeves, "Function minimization by conjugate gradients," *The computer journal*, vol. 7, no. 2, pp. 149–154, 1964.

[8] R. Fletcher, *Practical methods of optimization*. John Wiley & Sons, 2013.

[9] Y. Dai and Y. X. Yuan, "A nonlinear conjugate gradient method with a strong global convergence property," *SIAM Journal on optimization*, vol. 10, no. 1, pp. 177–182, 1999.

[10] 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.

[11] Y. 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.

[12] J. C. Gilbert and J. Nocedal, "Global convergence properties of conjugate gradient methods for optimization," *SIAM Journal on optimization*, vol. 2, no. 1, pp. 21–42, 1992.

[13] Y.-h. Dai and Y. Yuan, "An efficient hybrid conjugate gradient method for unconstrained optimization," *Annals of Operations Research*, vol. 103, no. 1, pp. 33–47, 2001.

[14] G. Zoutendijk, "Nonlinear programming, computational methods," *Integer and nonlinear programming*, pp. 37–86, 1970.

[15] M. Al-Baali, "Descent property and global convergence of the fletcher—reeves method with inexact line search," *IMA Journal of Numerical Analysis*, vol. 5, no. 1, pp. 121–124, 1985.

[16] M. Malik, M. Mamat, S. S. Abas, I. M. Sulaiman, and Sukono, "Performance analysis of new spectral and hybrid conjugate gradient methods for solving unconstrained optimization problems.," *IAENG International Journal of Computer Science*, vol. 48, no. 1, pp. 66–79, 2021.

[17] 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.

[18] I. Sulaiman, M. Mamat, M. Waziri, U. Yakubu, and M. Malik, "The convergence properties of a new hybrid conjugate gradient parameter for unconstrained optimization models," in *Journal of Physics: Conference Series*, vol. 1734, p. 012012, IOP Publishing, 2021.

[19] M. Malik, M. Mamat, S. S. Abas, *et al.*, "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.

[20] M. Malik, M. Mamat, S. S. Abas, I. M. Sulaiman, *et al.*, "A new modification of nprp conjugate gradient method for unconstrained optimization," *Advances in Mathematics: Scientific Journal*, vol. 9, no. 7, pp. 4955–4970, 2020.

[21] M. Malik, S. S. Abas, M. Mamat, I. S. Mohammed, *et al.*, "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.

[22] M. Malik, M. Mamat, S. S. Abas, I. M. Sulaiman, *et al.*, "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.

[23] I. Sulaiman, M. Mamat, M. Waziri, U. Yakubu, and M. Malik, "The performance analysis of a new modification of conjugate gradient parameter for unconstrained optimization models," *Mathematics and Statistics*, vol. 9, no. 1, pp. 16–23, 2021.

[24] A. B. Abubakar, P. Kumam, M. Malik, P. Chaipunya, and A. H. Ibrahim, "A hybrid fr-dy conjugate gradient algorithm for unconstrained optimization with application in portfolio selection," *AIMS Mathematics*, vol. 6, no. 6, pp. 6506–6527, 2021.

[25] N. Andrei, *Nonlinear Conjugate Gradient Methods for Unconstrained Optimization*. Springer, 2020.

[26] M. Jamil and X.-S. Yang, "A literature survey of benchmark functions for global optimisation problems," *International Journal of Mathematical Modelling and Numerical Optimisation*, vol. 4, no. 2, pp. 150–194, 2013.

[27] E. D. Dolan and J. J. Moré, "Benchmarking optimization software with performance profiles," *Mathematical Programming*, vol. 91, no. 2, pp. 201–213, 2002.

[28] S. Roman, *Introduction to the mathematics of finance: from risk management to options pricing*. Springer Science & Business Media, 2004.