Comparison of Conjugate Gradient Method on Solving Unconstrained Optimization Problems

Maulana Malik
University of Indonesia
40 PUBLICATIONS **113** CITATIONS

Mustafa Mamat
Universiti Sultan Zainal Abidin
404 PUBLICATIONS **2,840** CITATIONS

Siti Sabariah Abas
Universiti Sultan Zainal Abidin | UniSZA
16 PUBLICATIONS **58** CITATIONS

Sulaiman Mohammed Ibrahim
Universiti Utara Malaysia
95 PUBLICATIONS **331** CITATIONS

Comparison of Conjugate Gradient Method on Solving Unconstrained Optimization Problems

Maulana Malik*
Department of Mathematics
Universitas Indonesia (UI)
Depok 16424, Jawa Barat, Indonesia
m.malik@sci.ui.ac.id

Mustafa Mamat, Siti Sabariah Abas, Ibrahim Mohammed Sulaiman
Faculty of Informatics and Computing
Universiti Sultan Zainal Abidin (UniSZA)
Besut 22200, Terengganu, Malaysia
must@unisza.edu.my, sabariahabas@unisza.edu.my, sulaimanib@unisza.edu.my

Sukono
Department of Mathematics
Universitas Padjadjaran (Unpad)
Sumedang 45363, Jawa Barat, Indonesia
sukono@unpad.ac.id

Abdul Talib Bon
Department of Production and Operations,
Universiti Tun Hussein Onn Malaysia (UTHM)
Parit Raja 86400, Johor, Malaysia
talibon@gmail.com

Abstract

Conjugate gradient (CG) method approaches have been instrumental in solving unconstrained optimization problems. In 2020, Malik et al. have proposed a new hybrid coefficient (H-MS2), a combination of the RMIL coefficient and the new coefficient. In this paper, we propose the new method, which takes the new coefficients from H-MS2. Also, we will compare the new method and some of the classic methods that already based on the number of iterations and central processing unit (CPU) time. The new method fulfills the sufficient descent condition and global convergence properties, and it's tested on a set functions under exact line search. The numerical results show that the new CG method has the best efficiency between all the methods tested.

Keywords
Conjugate gradient method, unconstrained optimization problems, sufficient descent condition, global convergence properties, exact line search

1. Introduction

Consider the following an unconstrained optimization problem in the form of

$$\min_{x \in \mathbb{R}^n} f(x) \qquad (1)$$

where $f:\mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable function, and $\mathbb{R}^n$ denotes an n-dimensional Euclidean space. The conjugate gradient method is an iterative method, that is well-suited to solving problems (1) of the large scale (Polak, 1997). The iterative formula is given by

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k \,, k = 0,1,2,\ldots \tag{2}$$

where $\boldsymbol{x}_k$ is the current iterate point, $\boldsymbol{d}_k$ is a direction of f at $\boldsymbol{x}_k$, and $\alpha_k > 0$ is step-size obtained by a one-dimensional line search. Step size α_k is obtained using several forms of line search, i.e., exact line search (Nocedal and Wright, 2006) as follows:

$$f(\boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k) = \min_{\alpha \geq 0} f(\boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k) \tag{3}$$

or the strong Wolfe line search:

$$f(\boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k) \leq f(\boldsymbol{x}_k) + \delta \alpha_k \boldsymbol{g}_k^T \boldsymbol{d}_k,\ |\boldsymbol{g}(\boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k)^T \boldsymbol{d}_k| \leq -\sigma \boldsymbol{g}_k^T \boldsymbol{d}_k \tag{5}$$

with $0 < \delta < \sigma < 1$ (Wolfe, 1969).
The search direction $\boldsymbol{d}_k$ on this gradient conjugate method uses the following rules:

$$\boldsymbol{d}_k = \begin{cases} -\boldsymbol{g}_k\,, & k = 0 \\ -\boldsymbol{g}_k + \beta_k \boldsymbol{d}_{k-1}\,, & k \geq 1 \end{cases} \tag{6}$$

where $\boldsymbol{g}_k = \nabla f(\boldsymbol{x}_k)$ is gradient f at $\boldsymbol{x}_k$, and $\beta_k \in \mathbb{R}$ is a scalar parameter known as the conjugate gradient coefficient.

Many formulas have been proposed to calculate the β_k. The most well-known are Hestenes-Stiefel (HS) (Hestenes and Stiefel, 1952), Fletcher-Reeves (FR) (Fletcher and Reeves, 1964), Polak-Ribiere-Polyak (PRP) (Polak and Ribiere, 1969), Conjugate Descent (CD) (Fletcher, 1987), Dai-Yuan (DY) (Dai and Yuan, 1999) and WYL (Wei-Yao-Liu) (Wei et al., 2006).

$$\beta_k^{HS} = \frac{\boldsymbol{g}_k^T(\boldsymbol{g}_k - \boldsymbol{g}_{k-1})}{\boldsymbol{d}_{k-1}^T(\boldsymbol{g}_k - \boldsymbol{g}_{k-1})}, \beta_k^{FR} = \frac{\|\boldsymbol{g}_k\|^2}{\|\boldsymbol{g}_{k-1}\|^2}, \beta_k^{PRP} = \frac{\boldsymbol{g}_k^T(\boldsymbol{g}_k - \boldsymbol{g}_{k-1})}{\|\boldsymbol{g}_{k-1}\|^2}, \beta_k^{CD} = -\frac{\|\boldsymbol{g}_k\|^2}{\boldsymbol{d}_{k-1}^T \boldsymbol{g}_{k-1}},$$
$$\beta_k^{DY} = \frac{\|\boldsymbol{g}_k\|^2}{\boldsymbol{d}_{k-1}^T(\boldsymbol{g}_k - \boldsymbol{g}_{k-1})}, \beta_k^{WYL} = \frac{\boldsymbol{g}_k^T\left(\boldsymbol{g}_k - \frac{\|\boldsymbol{g}_k\|}{\|\boldsymbol{g}_{k-1}\|}\boldsymbol{g}_{k-1}\right)}{\|\boldsymbol{g}_{k-1}\|^2}$$

where $\|\,.\,\|$ denotes the Euclidean norm of vectors.

The global convergence properties are the most well-studied properties of CG methods. The HS method is one of the most efficient CG methods, which has good performance but no fulfill the global convergence properties under traditional line search. The FR method was developed from the HS method, which has a global convergent under exact and strong Wolfe line search (Al-Baali, 1985). The CD method has descent direction under the strong Wolfe line search (Liu et al., 2012) and fulfills the sufficient descent condition under the strong Wolfe line search. Dai and Yuan proposed the DY method, which has convergent globally under the strong Wolfe line search and Armijo line search (Zhang, 2007). Wei et al. proposed a new CG coefficient, which is a modification of the β_k^{FR}. This method known to fulfill the sufficient descent condition and global convergence properties under exact line search, Grippo-Lucidi line search, and strong Wolve line search. For good references to studies that have described recent CG methods with important results, see Rivaie et al., 2015, Yousif, 2020, Basri and Mamat, 2018, Waziri et al., 2019, Yuan et al., 2020, Liu, 2013, and Babaie-Kafaki, 2014.

In this paper, we will present our new CG algorithm whose performance compared with classical formulas of the FR, CD, DY, and WYL. Section 2 presents a new CG algorithm to solve unconstrained optimization problems. In Section 3, we shall present the sufficient descent condition and the global convergence. Significant numerical results and discussions will be presented in Section 4. Finally, the conclusions are presented in Section 5.

2. A New CG Algorithm

Recently, Malik et al., 2020, proposed the new hybrid conjugate gradient coefficient. The coefficient defined as follows:

$$\beta_k^{H-MS2} = \max\{0, \min\{\beta_k^{RMIL}, \beta_k^*\}\},$$

where $\beta_k^{RMIL} = \frac{\boldsymbol{g}_k^T(\boldsymbol{g}_k - \boldsymbol{g}_{k-1})}{\|\boldsymbol{d}_{k-1}\|^2}$ (Rivaie et al., 2012) and

$$\beta_k^* = \frac{\boldsymbol{g}_k^T\left(\boldsymbol{g}_k - \frac{\|\boldsymbol{g}_k\|}{\|\boldsymbol{g}_{k-1}\|}\boldsymbol{g}_{k-1} - \boldsymbol{g}_{k-1}\right)}{\boldsymbol{g}_{k-1}^T(\boldsymbol{g}_k - \boldsymbol{d}_{k-1})}. \tag{7}$$

In this paper, we propose to compute β_k using only β_k^*.

The algorithm is given as follows:

Algorithm 1
Step 1. Initialization. Given $\boldsymbol{x}_0 \in \mathbb{R}^n$, set $k \coloneqq 0$ and stopping criteria $\varepsilon > 0$.
Step 2. If $\|\boldsymbol{g}_k\| \leq \varepsilon$, then stop; otherwise, go to Step 3.
Step 3. Calculate the step size β_k using (7).
Step 4. Calculate $\boldsymbol{d}_k$ using (6).
Step 5. Calculate α_k using (3).
Step 6. Set $k \coloneqq k + 1$, and calculate the next iterate, go to step 3.

3. Convergence Analysis

This section will show the sufficient descent condition and the global convergent properties of the new method using the exact line search.

3.1 Sufficient Descent Condition

Sufficient descent condition holds when

$$\boldsymbol{g}_k^T\boldsymbol{d}_k \leq -C\|\boldsymbol{g}_k\|^2, \text{ for } k \geq 0 \text{ and } C > 0. \tag{8}$$

Theorem 1. Consider a CG method with search direction $\boldsymbol{d}_k$ (6), β_k^* given as equation (7), then, the condition (8) will hold for all $k \geq 0$.

Proof. If $k = 0$, then $\boldsymbol{d}_0 = -\boldsymbol{g}_0$, so $\boldsymbol{g}_0^T\boldsymbol{d}_0 = \boldsymbol{g}_0^T(-\boldsymbol{g}_0) = -\left(\sqrt{\boldsymbol{g}_0^T\boldsymbol{g}_0}\right)^2 = -\|\boldsymbol{g}_0\|^2 < 0$. Hence, condition (8) holds true for $k = 0$. Next, we will show that for $k \geq 1$, condition (8) will hold true.
Multiply (6) by $\boldsymbol{g}_k^T$, then

$$\boldsymbol{g}_k^T\boldsymbol{d}_k = -\boldsymbol{g}_k^T\boldsymbol{g}_k + \beta^*\boldsymbol{g}_k^T\boldsymbol{d}_{k-1} = -\|\boldsymbol{g}_k\|^2 + \beta^*\boldsymbol{g}_k^T\boldsymbol{d}_{k-1}.$$

We know that for exact line search, $\boldsymbol{g}_k^T\boldsymbol{d}_{k-1} = 0$. Thus,

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

Hence, condition (8) holds for $k \geq 1$. The proof is finished.

3.2 Global Convergence Properties

Next, we will show that a new CG method with coefficient β_k^* fulfill the convergence properties. In analyzing the global convergence properties of CG methods, the following basic assumptions are often required.

Assumption 1. (A1) The level set $\Omega = \{\boldsymbol{x} \in \mathbb{R}^n : f(\boldsymbol{x}) \le f(\boldsymbol{x}_0)\}$ is bounded, where $\boldsymbol{x}_0$ is a given starting point. (A2) In an open convex set Ω_0 that contains Ω, f is continuous and differentiable, and its gradient is Lipschitz continuous; that is, for any $\boldsymbol{x}, \boldsymbol{y} \in \Omega_0$, there exists a constant $L > 0$ such that $\|\boldsymbol{g}(\boldsymbol{x}) - \boldsymbol{g}(\boldsymbol{y})\| \le L\|\boldsymbol{x} - \boldsymbol{y}\|$.

This Assumption yields the following lemma, which Zoutendijk had proved (Zoutendijk, 1970).

Lemma 1. Suppose that Assumptions 1 hold, let $\boldsymbol{x}_k$ be generated by Algorithm 1, where $\boldsymbol{d}_k$ is a descent search direction, and α_k is obtained by (3), then the following condition, known as the Zoutendijk condition, holds

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

By using Lemma 1, the following convergent theorem for the new CG method.

Theorem 2. Suppose Assumption 1 hold, $\{\boldsymbol{x}_k\}$ generated by Algorithm 1, where the step size α_k is determined by the exact line (3), then:

$$\lim_{k\to\infty} \inf \|\boldsymbol{g}_k\| = 0. \tag{9}$$

Proof. Proof by contradiction. Suppose (9) is not correct, then there is a constant $M > 0$ such that

$$\|\boldsymbol{g}_k\| \ge M, \text{ for every } k \ge 0. \tag{10}$$

Rewriting (6),

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

and squaring both sides, we obtain

$$\|\boldsymbol{d}_k + \boldsymbol{g}_k\|^2 = \|\beta_k^* \boldsymbol{d}_{k-1}\|^2$$

$$\Leftrightarrow \|\boldsymbol{d}_k\|^2 + \|\boldsymbol{g}_k\|^2 + 2\boldsymbol{g}_k^T \boldsymbol{d}_k = (\beta_k^*)^2 \|\boldsymbol{d}_{k-1}\|^2$$

$$\Leftrightarrow \|\boldsymbol{d}_k\|^2 = (\beta_k^*)^2 \|\boldsymbol{d}_{k-1}\|^2 - 2\boldsymbol{g}_k^T \boldsymbol{d}_k - \|\boldsymbol{g}_k\|^2.$$

Using $(\boldsymbol{g}_k^T \boldsymbol{d}_k)^2$ and dividing both sides, we get

$$\frac{\|\boldsymbol{d}_k\|^2}{\left(\boldsymbol{g}_k^T \boldsymbol{d}_k\right)^2} = \frac{\left(\beta_k^*\right)^2 \|\boldsymbol{d}_{k-1}\|^2}{\left(\boldsymbol{g}_k^T \boldsymbol{d}_k\right)^2} - \frac{2}{\boldsymbol{g}_k^T \boldsymbol{d}_k} - \frac{\|\boldsymbol{g}_k\|^2}{\left(\boldsymbol{g}_k^T \boldsymbol{d}_k\right)^2} = \frac{\left(\beta_k^*\right)^2 \|\boldsymbol{d}_k\|^2}{\left(\boldsymbol{g}_k^T \boldsymbol{d}_k\right)^2} - \left(\frac{1}{\|\boldsymbol{g}_k\|} + \frac{\|\boldsymbol{g}_k\|}{\boldsymbol{g}_k^T \boldsymbol{d}_k}\right)^2 + \frac{1}{\|\boldsymbol{g}_k\|^2} \le \frac{\left(\beta_k^*\right)^2 \|\boldsymbol{d}_k\|^2}{\left(\boldsymbol{g}_k^T \boldsymbol{d}_k\right)^2} + \frac{1}{\|\boldsymbol{g}_k\|^2} \le \frac{1}{\|\boldsymbol{g}_k\|^2}.$$

So that,

$$\frac{\|\boldsymbol{d}_k\|^2}{(\boldsymbol{g}_k^T \boldsymbol{d}_k)^2} \le \frac{1}{\|\boldsymbol{g}_k\|^2}$$

or

$$\frac{(\boldsymbol{g}_k^T \boldsymbol{d}_k)^2}{\|\boldsymbol{d}_k\|^2} \ge \|\boldsymbol{g}_k\|^2.$$

Applying (10), then

$$\frac{(\boldsymbol{g}_k^T \boldsymbol{d}_k)^2}{\|\boldsymbol{d}_k\|^2} \ge M^2.$$

Take summation, we have

$$\sum_{k=0}^{n}\frac{(\boldsymbol{g}_k^T\boldsymbol{d}_k)^2}{\|\boldsymbol{d}_k\|^2}\geq\sum_{k=0}^{n}M^2=(n+1)M^2.$$

Hence,

$$\sum_{k=0}^{\infty}\frac{(\boldsymbol{g}_k^T\boldsymbol{d}_k)^2}{\|\boldsymbol{d}_k\|^2}=\lim_{n\to\infty}\sum_{k=0}^{n}\frac{(\boldsymbol{g}_k^T\boldsymbol{d}_k)^2}{\|\boldsymbol{d}_k\|^2}\geq\lim_{n\to\infty}(n+1)M^2=\infty.$$

This contradicts the Zoutendijk condition in Lemma 1. Therefore, the proof is completed.

4. Numerical Experiments

In this section, we will show the new method's numerical results compared with the other method of the FR, CD, DY, and WYL. We will use some of the test problems considered in (Andrei, 2008) under low, medium, and high dimensions as in (Malik et al., 2020), namely, 2, 3, 4, 10, 50, 100, 500, 1000 and 10,000 to show the efficiency. The function used is an artificial function. Artificial functions are used to see algorithmic behavior in different situations such as the length of the narrow valleys, unimodal functions, and functions with large number of significant local optimal.

There are thirty-one nonlinear functions to be tested in this paper, as listed in Table 1. Furthermore, for each test dimension of the test function, one of which is the initial point that suggested by Andrei (Andrei, 2008). The comparison each method is based on the number of iterations (NOI) and the time in seconds required for running each of the test problems (CPU). The tests' evaluation was based on Nocedal-Wright line search algorithm for exact condition and coded in MATLAB under stopping criterion is set to $\|\boldsymbol{g}_k\|\leq 10^{-6}$. The test was carried out on a laptop with processor intel ® Core ™ i7-8550U CPU @ 1.80GHz (8CPUs), ~2.0 GHz, 16 GB for RAM and Windows 10 Professional 64bit operating system.

Table 1. List of the Test Functions

No	Test Function	No	Test Function
1	Extended White & Holst	17	Six hump camel
2	Extended Rosenbrock	18	Booth
3	Extended Freudenstein & Roth	19	Trecanni
4	Extended Beale	20	Zettl
5	Extended Wood	21	Shallow
6	Raydan 1	22	Generalized Quartic
7	Extended Tridiagonal 1	23	Quadratic QF2
8	Diagonal 4	24	Generalized Tridiagonal 1
9	Extended Himmelblau	25	Generalized Tridiagonal 2
10	FLETCHCR	26	POWER
11	Extended Powell	27	Quadratic QF1
12	NONSCOMP	28	Extended quadratic penalty QP2
13	Extended DENSCHNB	29	Extended quadratic penalty QP1
14	Extended Penalty	30	Matyas
15	Hager	31	Dixon and Price
16	Extended Maratos		

Table 2. Comparison Between the New Method, FR, CD, DY, and WYL

Funct.	Dim.	Point	New		FR		CD		DY		WYL	
			NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU
1	1000	(-1.2,1)	23	0.959	26	0.8585	26	0.982	26	1.0212	1251	29.7987
		(10,10)	39	1.4218	293	9.7359	222	7.54	278	9.6918	2227	53.518

Funct.	**Dim.**	**Point**	**New**		**FR**		**CD**		**DY**		**WYL**	
			NOI	**CPU**	**NOI**	**CPU**	**NOI**	**CPU**	**NOI**	**CPU**	**NOI**	**CPU**
	10000	(-1.2,1)	23	6.7855	30	8.0132	30	7.9301	30	8.0372	1180	281.6603
		(5,5)	42	11.3446	180	48.6315	193	53.4613	145	39.8922	2923	704.4778
2	1000	(-1.2,1)	24	0.14	211	0.8497	100	0.407	221	0.8766	1283	3.7323
		(10,10)	29	0.1483	56	0.2329	56	0.2491	56	0.3524	1157	3.3508
	10000	(-1.2,1)	23	0.426	227	3.8702	99	1.6556	230	3.9131	415	9.1311
		(5,5)	37	0.6434	206	3.438	219	3.5928	224	3.6501	922	17.902
3	4	(0.5, -2)	8	0.0325	15	0.0676	15	0.0639	15	0.0778	560	1.032
		(5, 5,)	5	0.0214	7	0.0334	7	0.0346	7	0.0348	435	0.8117
4	1000	(1, 0.8)	15	0.5828	75	2.2823	75	2.2307	75	2.4375	315	8.4078
		(0.5, 0.5)	11	0.4071	81	2.526	81	2.5458	81	2.4948	263	7.1318
	10000	(-1, -1)	15	4.1158	87	24.0085	87	24.0719	87	24.0302	111	29.9674
		(0.5,0.5)	12	3.2948	87	23.9661	87	23.9937	87	23.9873	450	121.5545
5	4	(-3, -1)	235	0.5687	45545	258.8326	10820	62.6266	21279	118.3625	990	1.8304
6	10	(1,1)	17	0.0882	19	0.0449	19	0.0452	19	0.0484	24	0.0748
		(10,10)	72	0.174	19827	42.4045	13849	30.5942	20043	43.2475	208	0.4272
	100	(-1, -1)	93	0.3123	99	0.3171	90	0.2834	96	0.306	249	0.6654
		(-10, -10)	165	0.5009	985	2.8474	751	2.0725	fail	fail	376	0.9673
7	500	(2, 2)	333	5.4871	453	7.2773	452	7.2176	453	7.2673	3133	46.1204
		(10,10)	343	5.5663	7	0.1176	90	1.493	90	1.4984	2667	38.5006
	1000	(1,1)	423	18.8404	517	15.361	517	15.4591	517	20.1142	4381	118.1998
		(-10, -10)	443	16.9384	8	0.303	32	1.1196	78	2.9221	3651	97.6139
8	500	(1,1)	3	0.0132	5	0.0347	5	0.0311	5	0.0312	46	0.1409
		(-20, -20)	4	0.0277	5	0.0346	5	0.0349	5	0.0308	46	0.1427
	1000	(1,1)	3	0.0248	5	0.0337	5	0.0343	5	0.0387	46	0.1559
		(-30, -30)	4	0.0301	5	0.0397	5	0.0367	5	0.0393	64	0.2108
9	1000	(1,1)	8	0.0518	15	0.0893	15	0.0832	15	0.0802	20	0.083
		(20, 20)	6	0.0378	8	0.0496	8	0.0548	8	0.0507	10	0.0483
	10000	(-1, -1)	9	0.1947	27	0.5326	26	0.5417	27	0.5198	12	0.2166
		(50, 50)	7	0.1619	17	0.3447	17	0.3584	17	0.3445	20	0.3414
10	10	(0, 0)	53	0.1566	1214	3.1757	973	2.5102	1208	3.0949	198	0.3852
		(10,10)	28	0.1053	30	0.1165	30	0.1232	30	0.0959	48	0.1204
11	100	(3, -1,0,1)	516	8.2287	5644	87.6384	5640	89.2627	5653	90.6282	29913	131.7058
12	2	(3,3)	12	0.0582	50	0.1546	50	0.16	50	0.1612	118	0.2481
		(10, 10)	15	0.0692	915	4.7677	241	0.6989	1755	4.8267	109	0.2265
13	10	(1,1)	5	0.0236	9	0.0414	9	0.0415	9	0.0388	10	0.0427
		(10,10)	10	0.0483	13	0.056	13	0.059	13	0.0608	13	0.0366
	100	(10,10)	10	0.0462	13	0.0645	13	0.062	13	0.0675	14	0.0474
		(-50, -50)	10	0.0532	77	0.4324	77	0.2332	77	0.2346	14	0.0441
14	10	(1, …, 10)	16	0.0705	13	0.0614	13	0.0562	13	0.0595	48	0.1293
		(-10, -10)	8	0.0339	14	0.0639	14	0.0686	14	0.0657	39	0.0994
	100	(5, 5)	18	0.0702	13	0.0647	13	0.0735	14	0.0619	fail	fail
		(-10, -10)	12	0.0716	31	0.1023	39	0.1576	fail	fail	fail	fail
15	10	(1,1)	12	0.1513	11	0.0304	11	0.03	11	0.0315	13	0.1578
		(-10, -10)	18	0.0457	97	0.2343	99	0.2295	97	0.2283	21	0.0535
16	10	(1.1, 0.1)	71	0.1773	4820	8.5759	1022	1.9205	4096	7.5708	fail	fail
17	2	(-1,2)	7	0.0193	24	0.0564	24	0.0549	24	0.055	7	0.0419
		(-5, 10)	6	0.019	11	0.0389	11	0.031	11	0.0405	9	0.0336
18	2	(5, 5)	3	0.0178	3	0.0144	3	0.0151	3	0.0161	5	0.0184
		(10, 10)	3	0.0145	3	0.0187	3	0.0182	3	0.0205	4	0.0145
19	2	(-1, 0.5)	1	0.022	1	0.0084	1	0.0066	1	0.0075	1	0.0084
		(-5, 10)	7	0.0353	14	0.0524	14	0.062	14	0.0619	13	0.038
20	2	(-1, 2)	10	0.0508	11	0.0445	11	0.0495	11	0.0382	51	0.1497
		(10, 10)	11	0.0518	10	0.0456	10	0.0476	10	0.0474	73	0.1813
21	1000	(0, 0)	6	0.0454	18	0.103	18	0.1154	18	0.0851	76	0.2866
		(10, 10)	13	0.0755	175	0.6468	175	0.7126	175	0.6634	94	0.3217
	10000	(-1, -1)	10	0.1982	47	0.875	47	0.8105	47	0.8397	73	1.2541
		(-10, -10)	11	0.2213	43	0.7564	43	0.7327	43	0.7296	68	1.1735
22	1000	(1,1)	6	0.0472	6	0.0469	6	0.0488	6	0.0508	6	0.0427
		(20, 20)	12	0.0817	12	0.0732	12	0.0737	13	0.0976	9	0.0462
23	50	(0.5, 0.5)	86	0.2698	116	0.3075	117	0.3051	116	0.3219	151	0.3246
		(30, 30)	81	0.2209	125	0.4089	124	0.3247	126	0.3264	153	0.3462
24	10	(2, 2)	22	0.096	27	0.1314	27	0.1189	27	0.107	28	0.085
		(10, 10)	27	0.1119	43	0.1707	43	0.169	43	0.169	41	0.1117
25	4	(1, 1)	4	0.0214	5	0.0239	5	0.0235	5	0.023	5	0.0252

Funct.	Dim.	Point	New		FR		CD		DY		WYL	
			NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU	NOI	CPU
26	10	(1, 1)	24	0.186	20	0.0486	20	0.0487	21	0.0483	146	0.3292
		(10, 10)	25	0.0625	24	0.0546	24	0.0689	23	0.0659	157	0.3125
27	50	(1, 1)	38	0.0887	38	0.0847	38	0.0844	38	0.0863	126	0.2668
		(10, 10)	40	0.1072	41	0.1004	41	0.0959	41	0.0932	129	0.268
	500	(1, 1)	310	2.2523	131	0.58	131	0.546	131	0.5904	532	3.0803
		(-5, -5)	394	1.9249	137	0.5643	150	0.8401	137	0.5979	525	1.8863
28	100	(1, 1)	54	0.2979	189	0.9002	231	0.7706	141	0.7473	598	1.5307
		(10, 10)	52	0.2002	2758	7.671	101	0.4509	fail	fail	567	1.4758
29	4	(1, 1)	8	0.0339	20	0.0558	20	0.0462	20	0.048	17	0.0531
		(10, 10)	10	0.037	19	0.0574	19	0.0577	19	0.0562	21	0.0673
30	2	(1, 1)	1	0.0069	1	0.005	1	0.005	1	0.0083	1	0.0054
		(20, 20)	1	0.0059	1	0.0045	1	0.0048	1	0.0052	1	0.0099
31	3	(1, 1)	20	0.0559	15	0.0457	15	0.0343	15	0.0341	74	0.1607
		(10, 10)	22	0.0518	29	0.0668	29	0.0676	29	0.0693	76	0.167

The numerical results are combined using the profile results described in Dolan and More (Dolan and More, 2002). The profile results are illustrated in Figures 1 and 2. The results in Figure 1 and Figure 2 are obtained in the following way:

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

where $r_{p,s}$ is performance ratio, $a_{p,s}$ is the number of iterations or CPU time, P is set to test, and S is set of solvers on the test set P. Overall profile results can be obtained in the following ways:

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

The function $\rho_s(\tau)$ is the distribution function for the performance ratio, and $\rho_s(\tau)$ is the probability for solver $s \in S$, that a performance ratio $r_{p,s}$ is within a factor $\tau \in \mathbb{R}$ of the best possible ratio, and n_p is the number of functions.

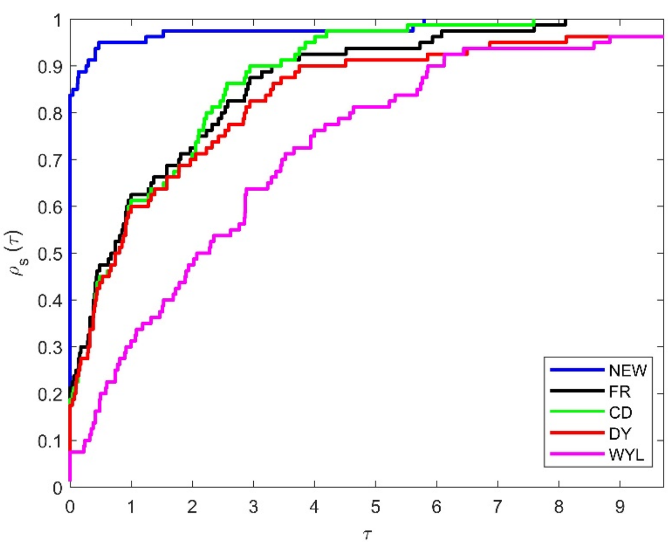

Figure 1. Performance Profile Based on Number of Iterations

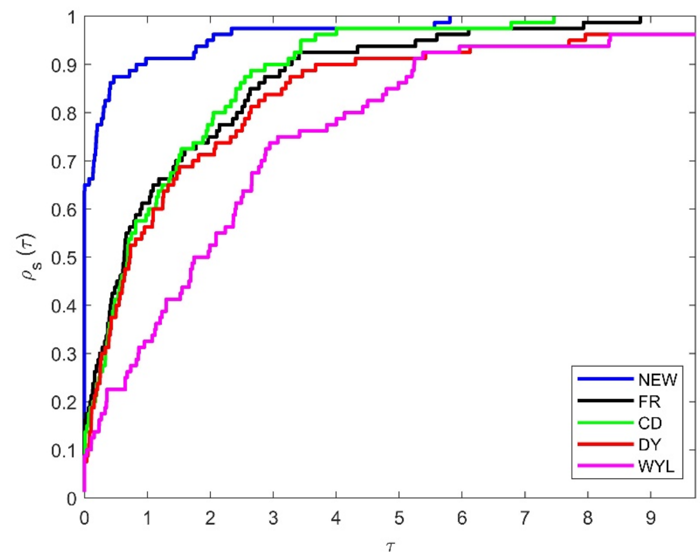

Figure 2. Performance Profile Based on CPU Time

Based on performance profiles plots in Figure 1 and Figure 2, where Figure 1 shows that the new method at the top curve, and it also in Figure 2. Corresponding in Table 2, the numerical results based on the number of iterations and CPU time shows that, the DY only reaches 96.25%, the WYL 96.25% and the FR, CD, and NEW method 100%. Since the curve of the new method on the top curve and reaches all the test function, then the new method is more efficient compared with other methods.

5. Conclusion

In this paper, we propose a new conjugate gradient method. Under some assumptions, the global convergent and the sufficient descent condition of the new conjugate gradient method when it is applied under exact line search have been established. A numerical experiment has shown the new method under exact line search can be used successfully in the practical computations and have the best performance compared with the other CG methods, namely FR, CD, DY, and WYL methods.

Acknowledgments

We would like to thank the reviewer for their suggestions and comments. This work is supported the Ph.D. mathematics study group on optimization field in Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin (UniSZA), Terengganu, Malaysia

References

Al-Baali, M., Descent property and global convergence of the Fletcher-Reeves method with inexact line searach, IMA Journal of Numerical Analysis, vol. 5, pp. 121-124, 1985.

Andrei, N., An Unconstrained Optimization Test Functions Collection, *Advanced Modelling and Optimization*, vol. 10, no. 1, pp. 147–161, 2008.

Babaie-Kafaki, S., Two modified scaled nonlinear conjugate gradient methods, *Journal of Computational and Applied Mathematics*, vol. 261, pp. 172-182, 2014.

Basri, S., and Mamat, M., A new class of nonlinear conjugate gradient with global convergence properties, *in Material Today: Proceedings*, 2018.

Dai, Y. H., and Yuan, Y., A nonlinear conjugate gradient method with a strong global convergence property, *SIAM Journal on optimization*, vol. 10, no. 1, pp. 177-182, 1999.

Dolan, E. D., and More, J. J., Benchmarking optimization software with performance profiles, *Mathematical programming*, vol. 91, no. 2, pp. 201-213, 2002.
Fletcher, R., *Practical methods of optimization*, John Wiley & Sons, New York, 1987.
Fletcher, R. and Reeves, C. M., Function minimization by conjugate gradients, *The computer journal*, vol. 7, no. 2, pp. 149-154, 1964.
Hestenes, M. R., and Stiefel, E., 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.
Liu, J., Convergence properties of a class of nonlinear conjugate gradient methods, *Computers and Operations Research*, vol. 40, no. 11, pp. 2656-2661, 2013.
Liu, J., Xianglin, D., and Kairong, W., A mixed spectral CD-DY conjugate gradient method, *Journal of Applied Mathematics*, pp. 1-10, 2012.
Malik, M., Mamat, M., Abas, S. S., 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.
Malik, M., Abas, S. S., Mamat, M., Sukono, and Mohammed, I. S., 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.
Nocedal, J. and Wright, S., *Numerical optimization*, Springer Science & Business Media, 2006.
Polak, E., *Optimization: Algorithms and Consistent Approximations*, Springer, Berlin, 1997.
Polak, E., and Ribiere, G., 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.
Rivaie, M., Mamat, M., June, L. W., and Mohd, I., A new class of nonlinear conjugate gradient coefficients with global convergence properties, *Applied Mathematics and Computation*, vol. 218, no. 22, pp. 11323-11332, 2012.
Rivaie, M., Mamat, M., and Abashar, A., A new class of nonlinear conjugate gradient coefficients with exact and inexact line searches, *Applied Mathematics and Computation*, vol. 268, pp. 1152-1163, 2015.
Waziri, M. Y., Ahmed, K., and Sabi'u, J., A family of Hager–Zhang conjugate gradient methods for system of monotone nonlinear equations, *Applied Mathematics and Computation*, vol. 361, pp. 645-660, 2019.
Wei, Z., Yao, S. and Liu, L., The convergence properties of some new conjugate gradient methods, *Applied Mathematics and Computation*, vol. 183, no. 2, pp. 1341-1350, 2006.
Wolfe, P., Convergence conditions for ascent methods, *SIAM Review*, vol. 11, no. 2, pp. 226-235, 1969.
Yousif, O.O.O., The convergence properties of RMIL+ conjugate gradient method under the strong Wolfe line search, *Applied Mathematics and Computation*, vol. 367, p. 124777, 2020.
Yuan, G., Li, T., and Hu, W, A conjugate gradient algorithm for large-scale nonlinear equations and image restoration problems, *Applied Numerical Mathematics*, vol. 147, no. 11661009, pp. 129-141, 2020.
Zhang, L., Zhou, W., and Li, D., Global convergence of the DY conjugate gradient method with Armijo line search for unconstrained optimization problems, *Optimization Methods and Software*, vol. 22, no. 3, pp. 511-517, 2007.
Zoutendijk, G., Nonlinear programming, computational methods, *Integer and nonlinear programming*, pp. 37-86, 1970.

Biographies

Maulana Malik is currently a Lecturer at the Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Indonesia, Indonesia, since 2016. He received his Bachelor of Science (2009), and Master of Science (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, and spectral CG methods. Maulana Malik is a member of Indonesian Mathematical Society (IndoMS), and in IAENG is a new member (236191) has been received in April 2019.

Mustafa Mamat is currently a Professor in the Faculty of Informatics and Computing at the Universiti Sultan Zainal Abidin (UniSZA) Malaysia since 2013. He was first appointed as a Lecturer at the Universiti Malaysia Terengganu (UMT), in 1999. He obtained his Ph.D. from the UMT in 2007 with a specialization in optimization. He was appointedas a Senior Lecturer in 2008 and as an Associate Professor in 2010 also the UMT. To date, he has published more than 380 research papers in various international journals and conferences. His research interest in applied mathematics, with a field of concentration of optimization, includes conjugate gradient methods, steepest descent methods, Broyden's family, and quasi-Newton methods.

Siti Sabariah Abas is a lecturer at the 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 a field in the numerical analysis include fluid dynamics.

Ibrahim Mohammed Sulaiman is currently a post-doctoral researcher at the 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 was born in Ngawi, East Java, Indonesia, on April 19, 1956. Master's in Actuarial Sciences at Institut Teknologi Bandung, Indonesia in 2000, and Ph.D. in Financial Mathematics at the Universitas Gajah Mada, Yogyakarta Indonesia, in 2011. The current work is the Chairman of the Master Program in Mathematics, Universitas Padjadjaran, Bandung Indonesia. The research is in the field of financial mathematics and actuarial science. Dr. Sukono is a member of Indonesian Mathematical Society (IndoMS), a member of Indonesian Operations Research Association (IORA), and in IAENG is a new member was received in February 2016.

Abdul Talib Bon is a professor of Production and Operations Management in the Faculty of Technology Management and Business at the Universiti Tun Hussein Onn Malaysia since 1999. He has a Ph.D. in Computer Science, which he obtained from the Universite de La Rochelle, France, in the year 2008. His doctoral thesis was on topic Process Quality Improvement on Beltline Moulding Manufacturing. He studied Business Administration in the Universiti Kebangsaan Malaysia, for which he was awarded the MBA in the year 1998. He's a bachelor's degree and a diploma in Mechanical Engineering, which his obtained from the Universiti Teknologi Malaysia. He received his postgraduate certificate in Mechatronics and Robotics from Carlisle, United Kingdom in 1997. He had published more 150 International Proceedings and International Journals and eight books. He is a member of MSORSM, IIF, IEOM, IIE, INFORMS, TAM, and MIM.