A comparison on classical-hybrid conjugate gradient method under exact line search

Nur Syarafina Mohamed [a,1,*], Mustafa Mamat [b,2], Mohd Rivaie [c,3], Shazlyn Milleana Shaharudin [d,4]

[a] Universiti Kuala Lumpur, Malaysian Institute of Industrial Technology, Johor Bharu, Malaysia
[b] Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin (Unisza), Terengganu, Malaysia
[c] Department of Computer Sciences and Mathematics, Universiti Teknologi MARA (UiTM), Terengganu, Malaysia
[d] Department of Mathematics, Faculty of Science and Mathematics, Universiti Pendidikan Sultan Idris, Malaysia
[1] nursyarafina@unikl.edu.my; [2] must@unisza.edu.my; [3] rivaie75@yahoo.com; [4] shazlyn@fsmt.upsi.edu.my
* corresponding author

ARTICLE INFO

Article history
Received May 12, 2019
Revised June 17, 2019
Accepted June 17, 2019
Available online July 31, 2019

Keywords
Exact line search
Conjugate gradient
Hybrid conjugate gradient
Classical conjugate gradient

ABSTRACT

One of the popular approaches in modifying the Conjugate Gradient (CG) Method is hybridization. In this paper, a new hybrid CG is introduced and its performance is compared to the classical CG method which are Rivaie-Mustafa-Ismail-Leong (RMIL) and Syarafina-Mustafa-Rivaie (SMR) methods. The proposed hybrid CG is evaluated as a convex combination of RMIL and SMR method. Their performance are analyzed under the exact line search. The comparison performance showed that the hybrid CG is promising and has outperformed the classical CG of RMIL and SMR in terms of the number of iterations and central processing unit per time.

1. Introduction

Consider the following unconstrained optimization objective function;

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

where $f : R^n \to R$ is a continuously differentiable function. R^n is denoted as $n-$dimensional Euclidean space [1]. With any $x_0 \in R^n$ as an initial guess, generally, a sequence of $\{x_k\}$ is generated by employing the CG iterative method given by;

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

where x_k is the k^{th} iterative point and $\alpha_k > 0$ is a step size while the search direction d_k is defined by;

$$d_k = \begin{cases} -g_k & \text{if} \quad k = 0 \\ -g_k + \beta_k d_{k-1} & \text{if} \quad k \geq 1 \end{cases}. \tag{3}$$

The scalar β_k is called the CG coefficient and g_k is the gradient of f at point x_k. Step size $\alpha_k > 0$ is the stepsize determined by using exact line search, given as;

$$f(x_k + \alpha_k d_k) = \min_{\alpha \geq 0} f(x_k + \alpha d_k). \tag{4}$$

Exact line search is used because of its well-known ability to provide the optimal step size [2]. Recently, studies have shown that the newer technologies with faster processors and better equipment have successfully eliminated the speed problems often suffered by exact line search, as demonstrated by Rivaie *et al.* [1]. This motivates plenty of its applications for solving unconstrained optimization problems. From [3], [4], a classical CG method introduced were;

$$\beta_k^{RMIL} = \frac{g_k^T (g_k - g_{k-1})}{\|d_{k-1}\|^2} \tag{5}$$

$$\beta_k^{SMR} = \max\left\{0, \frac{\|g_k\|^2 - |g_k^T g_{k-1}|}{\|d_{k-1}\|^2}\right\} \tag{6}$$

where the g_k and g_{k-1} are the abbreviations of $g(x_k)$ and $g(x_{k-1})$. They are the gradients of $f(x)$ at points x_k and x_{k-1} respectively. Euclidean norm of the vectors is denoted by $\|.\|$. These corresponding methods are known as RMIL(Rivaie-Mustafa-Ismail-Leong) [3] and SMR (Syarafina-Mustafa-Rivaie) [4]. Different CG method yield different performance of the CG algorithm due to the different choices for calculating the CG coefficients. Some well known CG formulas are;

$$\beta_k^{HS} = \frac{g_k^T (g_k - g_{k-1})}{(g_k - g_{k-1})^T d_{k-1}} \tag{7}$$

$$\beta_k^{FR} = \frac{g_k^T g_k}{\|g_{k-1}\|^2} \tag{8}$$

$$\beta_k^{PRP} = \frac{g_k^T (g_k - g_{k-1})}{\|g_{k-1}\|^2} \tag{9}$$

$$\beta_k^{CD} = -\frac{g_k^T g_k}{d_{k-1}^T g_{k-1}} \tag{10}$$

$$\beta_k^{DY} = \frac{g_k^T g_k}{(g_k - g_{k-1})^T d_{k-1}} \tag{11}$$

$$\beta_k^{LS} = \frac{g_k^T (g_k - g_{k-1})}{-d_{k-1}^T g_{k-1}} \tag{12}$$

These corresponding methods are known as HS (Hestenes and Steifel [5]), FR (Fletcher and Reeves [6]), PRP (Polak and Ribiere [7]), CD (Conjugate Descent by Fletcher [8]), DY (Dai and Yuan [9]), and LS (Liu-Storey [10]), respectively. Due to the strictly convex quadratic function $f(x)$, all of these methods (7-12) have finite convergence properties under exact line search. From all the mentioned methods, CG coefficient in (8), (10) and (11) have strong convergence properties but not excellent in practical performance due to the jamming problem. Meanwhile, methods in (7), (9) and (12) have better numerical performance though lacking in convergence properties [11]. Conjugate gradient method can be classified into three different groups; classical, scaled, and hybrid CG method [12]. Methods mentioned in (7-12) are called classical CG due to their simple approaches. Detailed discussions are available in [13]–[24]. Meanwhile, some well known hybrid conjugate algorithms are;

$$\beta_k^{HDY} = \max\{0, \min\{\beta_k^{HS}, \beta_k^{DY}\}\} \tag{13}$$

$$\beta_k^{HHUS} = \max\{0, \min\{\beta_k^{PRP}, \beta_k^{FR}\}\} \tag{14}$$

$$\beta_k^{HLSCD} = \max\{0, \min\{\beta_k^{LS}, \beta_k^{CD}\}\} \tag{15}$$

$$\beta_k^{HJHJ} = \frac{\|g_k\|^2 - \max\{0, \frac{\|g_k\|}{\|g_{k-1}\|} g_k^T g_{k-1}\}}{\max\{\|g_{k-1}\|^2, d_{k-1}^T (g_k - g_{k-1})\}} \tag{16}$$

From (13)-(16) HDY is a modification effort from Dai and Yuan [9] combining its algorithm with Hestenes and Steifel [5], HHUS was introduced by Hu and Storey, HLSCD is the combination of LS [10] and CD [8], and HJHJ is a hybrid CG from Jinbao-Han-Jiang [25]. In this paper, the main objective is to compare the performance of classical and hybrid CG methods. The idea is to combine different conjugate algorithms to use the projections to form a new hybrid convex-combination algorithm in order to avoid jamming [11] and compare with the original coefficients used for its hybridization. Hence, SMR and RMIL are combined in order to introduce a hybrid CG and its performance is compared between SMR and RMIL. SMR has good computational properties while RMIL has strong convergence properties [1], [2] both under exact line search. The combinations of all the good criteria of SMR and RMIL are used in order to obtain a better practical algorithm both in numerical and convergence analysis. Section two will discuss the motivation of the algorithm and the new hybrid conjugate gradient algorithm. Section three presents the convergence analysis. Numerical experiments are discussed in section four, and the last section concludes all the works in this paper.

2. Method

Hybrid-Syarafina-Mustafa-Rivaie (HSMR) method is the idea of combining SMR and RMIL methods together. HSMR method is introduced as Hybrid-Syarafina-Mustafa-Rivaie method and is known as;

$$\beta_k^{HSMR} = \max\{0, \min\{\beta_k^{SMR}, \beta_k^{RMIL}\}\} \tag{17}$$

The idea was initiated by using the restart strategy proposed in Jinbao *et al.* [25]. If the value tends to be negative values, it is preferable to set $\beta_k = 0$, which implies a restart along g_k. If $\beta_k < 0$, the search direction d_k from (3) tend to almost opposite to d_{k-1}. Then, the conjugate gradient (CG) coefficient to $\beta_k \geq 0$, the two consecutive search directions are prevented from tending to be almost opposite [26]. The new algorithm of β_k^{HSMR} is given as in Algorithm 1 (Fig. 1).

Algorithm 1: Conjugate gradient method

Step 1: Initialization. Set $k = 0$ and select $x_0 \in \Re^n$, $d_0 = -g_0$, if $g_0 = 0$, stop.

Step 2: Compute β_k^{HSMR} based on (17).

Step 3: Compute search directions d_k based on (3). If $\|g_k\| \leq \varepsilon$, then stop.
Otherwise, go to the next step.

Step 4: Compute for α_k based on exact line search (4).

Step 5: Updating new initial point using (2).

Step 6: Convergence test and stopping criteria. If $f(x_{k+1}) < f(x_k)$ and $\|g_k\| \leq \varepsilon$ then, stop.
Otherwise go to Step 2 with $k = k+1$.

Fig. 1. Conjugate gradient algorithm

3. Results and Discussion

3.1. Convergence Analysis

In this section, the convergence analysis for HSMR based on exact line search in (4) is analysed. An algorithm has to possess both sufficient descent condition and global convergence properties for a method to be converged in order to have a good practical algorithm. The convergence analysis for SMR and RMIL can be reached out in Rivai *et al.* [3] and Mohamed *et al.* [4].

3.1.1. Sufficient Descent Condition

Sufficient descent condition holds when

$$g_k^T d_k \le -C\|g_k\|^2 \text{ for } k \ge 0 \text{ and } C > 0 \tag{18}$$

Theorem 1: Consider a CG method with search direction (3) and β_k^{HSMR} defined as (17), then, condition (18) will hold for all $k \ge 0$.

Proof: From (3), know that $g_0^T d_0 = -C\|g_0\|^2$. Hence, condition (18) holds. In order to show condition (18) also hold for $k \ge 1$, multiply (3) by g_k^T. Then,

$$g_k^T d_k = -g_k^T g_k + \beta_k^{HSMR} g_k^T d_{k-1} = -\|g_k\|^2 + \beta_k^{HSMR} g_k^T d_{k-1}. \tag{19}$$

For $k = 1$, it is easy to know that $g_1^T d_1 = -\|g_k\|^2 < 0$. Assume that $g_{k-1}^T d_{k-1} < 0$ holds for $k-1$ and $k \ge 2$. To prove that $g_k^T d_k < 0$ for all k, consider these two cases:

CASE I: If $\beta_k^{RMIL} = \dfrac{g_k^T(g_k - g_{k-1})}{\|d_{k-1}\|^2} < 0$, automatically β_k^{HSMR} return to 0 such that $\beta_k^{SMR} \ge 0$. When β_k^{HSMR} =0, it is known that $g_k^T d_k = -\|g_k\|^2 < 0$.

CASE II: If $\beta_k^{RMIL} = \dfrac{g_k^T(g_k - g_{k-1})}{\|d_{k-1}\|^2} \ge 0$, then, $\beta_k^{HSMR} \ne 0$ and can either be β_k^{SMR} or β_k^{RMIL}.

Since the line search is exact, it is known that $g_k^T d_{k-1} = 0$. Thus, $g_k^T d_k = -\|g_k\|^2$ implying d_k is a sufficient descent direction. Hence, the descent condition holds i.e., $g_k^T d_k \le -C\|g_k\|^2$. The proof is completed.

3.1.2. Global Convergence Properties

From Rivai *et al.* [3] and Mohamed *et al.* [4], it is known that β_k^{SMR} and β_k^{RMIL} can be simplified to

$$0 \le \beta_{k+1} \le \frac{\|g_{k+1}\|^2}{\|d_k\|^2}. \tag{20}$$

Case II stated that β_k^{HSMR} and can either be β_k^{SMR} or β_k^{RMIL} if $\beta_k^{RMIL} = \dfrac{g_k^T(g_k - g_{k-1})}{\|d_{k-1}\|^2} \ge 0$. Then, from Rivai *et al.* [3], Mohamed *et al.* [4], and Rivaie *et al.* [19], in the analysis of global convergence properties, the following assumption is needed.

Assumption 1

- f is bounded below on the level set R^n and is continuous and differentiable in a neighborhood N of the level set $\ell = \{x \in R^n \mid f(x) \le f(x_0)\}$ at the initial point x_0.

- The gradient $g(x)$ is Lipschitz continuous in N, so, there exists a constant $L>0$ such that; $\|g(x)-g(y)\| \le L\|x-y\|$ for any $x, y \in N$.

Under this assumption, the following lemma is obtained, which was proved by Zoutendijk [27].

Lemma 1: Suppose that Assumption 1 holds. Consider any CG methods of the form (3) where d_k is a descent search direction and α_k satisfies the exact minimization rules. Then the following conditions known as Zoutendijk conditions hold; $\sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} < \infty$.

The proof of this lemma can be seen from Dai and Yuan [28]. By using this lemma; the following convergence theorem of the conjugate gradient method can be obtained by using (20).

Theorem 1: Suppose that Assumption 1 holds. Consider any CG methods in the form of (3) and (2) where α_k is obtained by the exact minimization rules. Also, suppose that Assumption 1 and the descent condition hold. Then either $\lim_{k\to\infty}\|g_k\| = 0$ or $\sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} < \infty$.

Proof: By induction, if Theorem 1 is not true then, there exists a constant $c>0$ such that

$$\|g_k\| \ge c \tag{21}$$

Rewriting (3) and squaring both sides, we get

$$\|d_{k+1}\|^2 = (\beta_{k+1})^2\|d_k\|^2 - 2g_{k+1}^T d_{k+1} - \|g_{k+1}\|^2 \tag{22}$$

Using $\left(g_{k+1}^T d_{k+1}\right)^2$ and dividing both sides, $\frac{\|d_{k+1}\|^2}{\left(g_{k+1}^T d_{k+1}\right)^2} = \frac{(\beta_{k+1})^2\|d_k\|^2}{\left(g_{k+1}^T d_{k+1}\right)^2} - \frac{2}{g_{k+1}^T d_{k+1}} - \frac{\|g_{k+1}\|^2}{g_{k+1}^T d_{k+1}}$.

By completing the square, the equation becomes

$$= \frac{(\beta_{k+1})^2\|d_k\|^2}{\left(g_{k+1}^T d_{k+1}\right)^2} - \left(\frac{1}{\|g_{k+1}\|} + \frac{\|g_{k+1}\|}{g_{k+1}^T d_{k+1}}\right)^2 + \frac{1}{\|g_{k+1}\|^2} \le \frac{(\beta_{k+1})^2\|d_k\|^2}{\left(g_{k+1}^T d_{k+1}\right)^2} + \frac{1}{\|g_{k+1}\|^2} .$$

Applying (20), yields

$$= \left(\frac{\|g_{k+1}\|^2}{\|d_k\|^2}\right)^2 \frac{\|d_k\|^2}{\left(g_{k+1}^T d_{k+1}\right)^2} + \frac{1}{\|g_{k+1}\|^2} = \frac{\|g_{k+1}\|^2}{\|d_k\|^2\|d_{k+1}\|^2} + \frac{1}{\|g_{k+1}\|^2} \le \frac{1}{\|d_k\|^2} + \frac{1}{\|g_{k+1}\|^2} \tag{23}$$

From (23), noting that

$$\frac{1}{\|d_0\|^2} = \frac{1}{\|g_0\|^2} ,$$

then,

$$\frac{\|d_k\|^2}{(g_k d_k)^2} = \frac{1}{\|g_0\|^2} + \frac{1}{\|g_k\|^2}.$$

Hence,

$$\frac{\|d_k\|^2}{(g_k d_k)^2} \le \sum_{i=0}^{k} \frac{1}{\|g_i\|^2} \text{ and } \frac{(g_k d_k)^2}{\|d_k\|^2} \ge \frac{c^2}{k}.$$

Then, from (21) and (23), it follows; $\sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} = \infty$. This contradicts the Zoutendijk [27] condition in Lemma 1. Therefore, the proof is completed.

Corollary 1: If $\sum_{k=0}^{\infty} \|d_k\|^2 = 0$, then $\sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} < \infty$ holds.

Proof: By using contradiction, assume that $\|g_k\| \ge c$ and $\sum_{k=0}^{\infty} \|d_k\|^2 = \infty$. For $\|g_k\| \to \infty$, then, $\frac{1}{\|g_k\|} \to 0$.

From (23),

$$\frac{\|d_k\|^2}{\left(g_k^T d_k\right)^2} \le \frac{1}{\|d_k\|^2}$$

then,

$$\|d_k\|^2 \le \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2}.$$

Which leads to, $\sum_{k=0}^{\infty} \|d_k\|^2 \le \sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2}$ and $\infty \le \sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2}$.

This contradicts the Zoutendijk [27] conditions. Hence the corollary holds.

3.2. Discussion

This section shows the numerical performance of the new coefficient HSMR compared with its classical combination in (5) and (6). In this research, artificial problems are chosen from the small to large-scale twenty-one test functions as listed in Table 1. The list is considered from Andrei [29]. A large-scaled problem is chosen in order to detect a cynical observer preventing the algorithm being tuned in particular functions [29]. For each of the test functions selected, random initial points are chosen in order to test the global convergence properties along with the robustness of the methods. On the other hand, for each of the random initial points chosen, they are tested on various dimensions from two to thousand dimensions [30], [31]. Points chosen can also be used to test the global convergence properties and the robustness of the methods. Results analyses are based on MATLAB subroutine program on workstation Intel Core i7, 2.2 GHz tested on the number of iterations and central processing time per unit. The stopping criterion is set to $\|g_k\| \le 10^{-6}$, where $\varepsilon = 10^{-6}$. Performance profiles based on Dolan and More [32] are shown graphically in figures below. The results summaries for all SMR, RMIL, and HSMR methods are shown in Table 2.

Table 1. List of test functions

Function	Initial Points
Camel Function- Three Hump	(-1,1), (1,-1), (-2,2), (2,-2).
Camel Function- Six Hump	(8,8), (-8,-8), (10,10), (-10,-10).
Booth Function	(10,10), (25,25), (50,50), (100,100).
Dixon and Price Function	(2,-2), (5,-5), (10,-10), (13,-13).
Treccani Function	(5,5), (10,10), (50,50), (100,100).
Zettl Function	(5,5), (10,10), (20,20), (50,50).
Leon Function	(5,5), (10,10), (-5,-5), (-10,-10).
Quartic Function	(2,2,2,2), (5,5,5,5), (8,8,8,8), (10,10,10,10).
Extended Freudstein and Roth Function	(1,...,1), (3,...,3), (5,...,5), (7,...,7)
Extended Himmelblau Function	(10,...,10), (50,...,50), (100,...,100), (200,...,200)
Extended Rosenbrock Function	(13,...,13), (16,...,16), (20,...,20), (30,...,30)
Extended Denschnb Function	(5,...,5), (8,...,8), (13,...,13), (25,...,25)
Extended White and Holst Function	(-3,...,-3), (3,...,3), (6,...,6), (9,...,9)
Extended Beale Function	(5,...,5), (10,...,10), (2,...,2), (8,...,8)
Cube Function	(-3,...,-3), (3,...,3), (6,...,6), (9,...,9)
Extended Tridiagonal 1 Function	(10,...,10), (12,...,12), (17,...,17), (20,...,20)
Shallow Function	(10,..,10), (25,...,25), (50,...,50), (100,...,100)
Generalized Quartic Function	(10,..,10), (50,...,50), (100,...,100), (200,...,200)
Fletchcr Function	(3,...,3), (5,...,5), (2,...,2), (9,...,9)
Extended Maratos Function	(10,..,10), (50,...,50), (100,...,100), (150,...,150)
Diagonal 4	(10,..,10), (50,...,50), (100,...,100), (200,...,200)

Performance comparison involving tables are very hard to interpret in order to benchmark methods' efficiency. Thus, the best way to discuss the performance of each method is by using the performance profile as introduced by Dolan [33]. The $P_s(t)$ from the performance profile is the fraction of the problem with a ratio performance t. A solver has higher efficiency when its value $P_s(t)$ is higher. In a set of problem P and a set of optimization solver S, a performance comparison of the problem $p \in P$ by a particular algorithm $s \in S$ is measured. Let, $t_{p,s}$ be the number of iterations or CPU time required when solving a problem $p \in P$ with the solver $s \in S$. The performance ratio is defined by $r_{p,s} = \frac{t_{p,s}}{\min\{t_{p,s} : s \in S\}}$. From this expression, it is assumed that $r_{p,s} \in [1, r_M]$, where $r_M \geq r_{p,s}$ and $r_{p,s} = r_M$ only when the problem p is not solved by the solver s. Then, graphically, a graph of $P_s(t)$ versus $t \in [1, r_M]$ is plotted.

Table 2. Results summaries of SMR, RMIL, and HSMR

Method	Total NOI	*Total CPU time*	*Total CPU/ Total NOI*
RMIL	50811	257.1624	0.0051
SMR	195746	345.2154	0.0018
HSMR	29791	1784.5037	0.0599

From Fig. 2, the graph shows that HSMR, SMR, and RMIL are quite identical to each other in terms of their competitiveness and HSMR slightly reach 1 at the top faster than SMR and HSMR. In a graph of performance profile, the smallest performance ratio is 1 and it will be located at the most left of t - axis, hence, the top curve represents the most efficient method. Though the values displayed in Table 2 may be used to describe the methods' efficiency, this way of comparison is not very fair since the presence of 'NAN' in some problems are not taken into account.

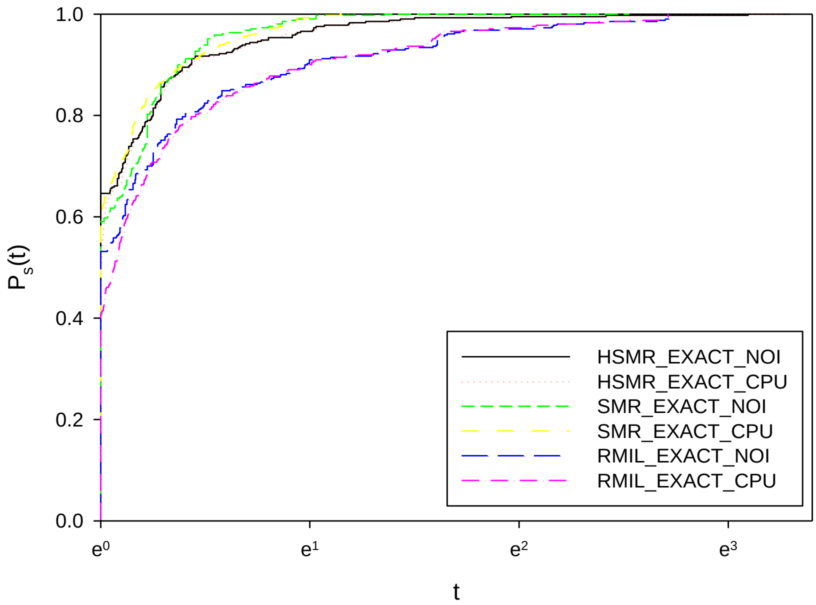

Fig. 2. Performance Profile for SMR, HSMR, and RMIL under Exact Line Search

From Fig. 2, the performance profile showed indicates the comparison of each of the coefficient performances in terms of Number of Iterations (NOI) and Central Processing Time per Unit (CPU time) under exact line search. The top left curve indicates the fastest coefficient reaching the solution point while the top right curve shows the ability of the coefficient can solve the test functions used. Based on these curves, it is shown that SMR and HSMR outperform the RMIL under exact line search.

4. Conclusion

In this paper, a hybrid CG method of HSMR method is introduced as a combination of SMR and RMIL. All three methods are compared and the results show that the hybrid version has better performance due to inherit the good characteristics from both SMR and RMIL. The proofs showed that the new method fulfills the convergence properties. The application for these methods can be seen in [33]–[35]. For future works, these methods should be done for inexact line search approach.

Acknowledgment

The authors are very grateful to the editors and the anonymous reviewers for their valuable comments and suggestions which improved this paper substantially. The author is very grateful to Universiti Kuala Lumpur to make the publication possible.

References

[1] M. Rivaie, M. Mamat, and A. Abashar, "A new class of nonlinear conjugate gradient coefficients with exact and inexact line searches," *Appl. Math. Comput.*, vol. 268, pp. 1152–1163, Oct. 2015, doi: 10.1016/j.amc.2015.07.019.

[2] M. A. Hery, M. Ibrahim, and L. W. June, "BFGS method: a new search direction," *Sains Malaysiana*, vol. 43, no. 10, pp. 1591–1597, 2014, available at: Google Scholar.

[3] M. Rivaie, M. Mamat, L. W. June, and I. Mohd, "A new class of nonlinear conjugate gradient coefficients with global convergence properties," *Appl. Math. Comput.*, vol. 218, no. 22, pp. 11323–11332, Jul. 2012, doi: https://doi.org/10.1016/j.amc.2012.05.030.

[4] N. S. Mohamed, M. Mamat, F. S. Mohamad, and M. Rivaie, "A new coefficient of conjugate gradient methods for nonlinear unconstrained optimization," *J. Teknol.*, vol. 78, no. 6–4, pp. 131–136, 2016, doi: https://doi.org/10.11113/jt.v78.8988.

[5] M. R. Hestenes and E. Stiefel, "Methods of conjugate gradients for solving linear systems," *J. Res. Natl. Bur. Stand. (1934).*, vol. 49, no. 6, p. 409, Dec. 1952, doi: 10.6028/jres.049.044.

[6] R. Fletcher, "Function minimization by conjugate gradients," *Comput. J.*, vol. 7, no. 2, pp. 149–154, Feb. 1964, doi: 10.1093/comjnl/7.2.149.

[7] E. Polak and G. Ribiere, "Note sur la convergence de méthodes de directions conjuguées," *Rev. française d'informatique Rech. opérationnelle. Série rouge*, vol. 3, no. 16, pp. 35–43, May 1969, doi: 10.1051/m2an/196903R100351.

[8] R. Fletcher, *Practical Methods of Optimization: Unconstrained optimization*, no. v. 1. J. Wiley, 1980, available at: https://books.google.co.id/books?id=igc8AQAAMAAJ.

[9] Y. H. Dai and Y. Yuan, "A Nonlinear Conjugate Gradient Method with a Strong Global Convergence Property," *SIAM J. Optim.*, vol. 10, no. 1, pp. 177–182, Jan. 1999, doi: 10.1137/S1052623497318992.

[10] Y. Liu and C. Storey, "Efficient generalized conjugate gradient algorithms, part 1: Theory," *J. Optim. Theory Appl.*, vol. 69, no. 1, pp. 129–137, Apr. 1991, doi: 10.1007/BF00940464.

[11] N. Andrei, "Hybrid Conjugate Gradient Algorithm for Unconstrained Optimization," *J. Optim. Theory Appl.*, vol. 141, no. 2, pp. 249–264, May 2009, doi: 10.1007/s10957-008-9505-0.

[12] N. Andrei, "Open problems in nonlinear conjugate gradient algorithms for unconstrained optimization," *Bull. Malaysian Math. Sci. Soc.*, vol. 34, no. 2, 2011, available at: Google Scholar.

[13] S. Shoid, M. Rivaie, and M. Mamat, "A modification of classical conjugate gradient method using strong Wolfe line search," 2016, p. 020071, doi: 10.1063/1.4952551.

[14] A. Abashar, M. Mamat, M. Rivaie, I. Mohd, and O. Omer, "The proof of sufficient descent condition for a new type of conjugate gradient methods," 2014, pp. 296–303, doi: 10.1063/1.4882502.

[15] N. H. A. Ghani, M. Rivaie, and M. Mamat, "A modified form of conjugate gradient method for unconstrained optimization problems," 2016, p. 020076, doi: 10.1063/1.4952556.

[16] N. 'Aini, M. Rivaie, and M. Mamat, "A modified conjugate gradient coefficient with inexact line search for unconstrained optimization," 2016, p. 080019, doi: 10.1063/1.4968158.

[17] N. Hajar, M. Mamat, M. Rivaie, and Z. Salleh, "A combination of Polak-Ribiere and Hestenes-Steifel coefficient in conjugate gradient method for unconstrained optimization," *Appl. Math. Sci.*, vol. 9, pp. 3131–3142, 2015, doi: 10.12988/ams.2015.53242.

[18] M. Hamoda, M. Mamat, M. Rivaie, and Z. Salleh, "A conjugate gradient method with strong Wolfe-Powell line search for unconstrained optimization," *Appl. Math. Sci.*, vol. 10, pp. 721–734, 2016, doi: 10.12988/ams.2016.56449.

[19] M. Rivaie, A. Abashar, M. Mamat, and I. Mohd, "The convergence properties of a new type of conjugate gradient methods," *Appl. Math. Sci.*, vol. 8, pp. 33–44, 2014, doi: 10.12988/ams.2014.310578.

[20] N. S. Mohamed, M. Mamat, and M. Rivaie, "Solving a large scale nonlinear unconstrained optimization with exact line search direction by using new coefficient of conjugate gradient methods," 2016, p. 080018, doi: 10.1063/1.4968157.

[21] N. Shapiee, M. Rivaie, and M. Mamat, "A new classical conjugate gradient coefficient with exact line search," 2016, p. 020082, doi: 10.1063/1.4952562.

[22] Z. Z. Abidin, M. Mamat, M. Rivaie, and I. Mohd, "A new steepest descent method," 2014, pp. 273–278, doi: 10.1063/1.4882499.

[23] A. Abashar, "A new type of conjugate gradient methods with sufficient descent property for unconstrained optimization," Terengganu: Universiti Malaysia Terengganu, 2014, available at: Google Scholar.

[24] I. Jusoh, "New family of conjugate gradient methods with sufficient descent condition and global convergence for unconstrained optimization," Universiti Sultan Zainal Abidin, 2015.

[25] J. Jian, L. Han, and X. Jiang, "A hybrid conjugate gradient method with descent property for unconstrained optimization," *Appl. Math. Model.*, vol. 39, no. 3–4, pp. 1281–1290, Feb. 2015, doi: 10.1016/j.apm.2014.08.008.

[26] M. J. D. Powell, "Nonconvex minimization calculations and the conjugate gradient method," 1984, pp. 122–141, doi: 10.1007/BFb0099521.

[27] G. Zoutendijk, "Nonlinear programming, computational methods," *Integer nonlinear Program.*, pp. 37–86, 1970, available at: Google Scholar.

[28] Y. H. Dai and Y. Yuan, "An Efficient Hybrid Conjugate Gradient Method for Unconstrained Optimization," *Ann. Oper. Res.*, 2001, doi: 10.1023/A:1012930416777.

[29] N. Andrei, "An unconstrained optimization test functions collection," *Adv. Model. Optim*, vol. 10, no. 1, pp. 147–161, 2008, available at: Google Scholar.

[30] J. J. Moré, B. S. Garbow, and K. E. Hillstrom, "Testing Unconstrained Optimization Software," *ACM Trans. Math. Softw.*, vol. 7, no. 1, pp. 17–41, Mar. 1981, doi: 10.1145/355934.355936.

[31] K. E. Hillstrom, "A Simulation Test Approach to the Evaluation of Nonlinear Optimization Algorithms," *ACM Trans. Math. Softw.*, vol. 3, no. 4, pp. 305–315, Dec. 1977, doi: 10.1145/355759.355760.

[32] E. D. Dolan and J. J. Moré, "Benchmarking optimization software with performance profiles," *Math. Program.*, vol. 91, no. 2, pp. 201–213, Jan. 2002, doi: 10.1007/s101070100263.

[33] N. S. Mohamed *et al.*, "Estimating the Number of Paediatric Patients Using Least Square and Conjugate Gradient Methods," *J. Eng. Appl. Sci.*, vol. 12, no. 12, pp. 3068–3071, 2017, doi: 10.3923/jeasci.2017.3068.3071.

[34] N. Syarafina Mohamed, M. Mamat, M. Rivaie, N. Hamizah Abdul Ghani, N. Zull, and S. Syoid, "Estimating the unemployment rate using least square and conjugate gradient methods," *Int. J. Eng. Technol.*, vol. 7, no. 2.15, p. 94, Apr. 2018, doi: 10.14419/ijet.v7i2.15.11360.

[35] S. Shoid *et al.*, "The application of new conjugate gradient methods in estimating data," *Int. J. Eng. Technol.*, vol. 7, no. 2.14, p. 25, Apr. 2018, doi: 10.14419/ijet.v7i2.14.11147.