MJFAS
Malaysian Journal of Fundamental and Applied Sciences

Luyoh *et al.* | Malaysian Journal of Fundamental and Applied Sciences, Vol. 22 (2026) 928–939

RESEARCH ARTICLE

Hybrid Conjugate Gradient Method with Exact Line Search for Grey Wolf Optimization Algorithm

Hannah Telen Luyoh[a], Ainur Madihah Mohd Nordin[b], Muhammad Najhan Abdul Rahaman[b], Aleya Maisarah Mohd Hakim[b], Norhaslinda Zullpakkal[b,*], Nur Syarafina Mohamed[c]

[a]Datamicron Systems Sdn Bhd 11th Floor, Suite 11-1, Wisma UOA 2, 21, Jalan Pinang, Kuala Lumpur, 50450 Kuala Lumpur, Federal Territory of Kuala Lumpur; [b]Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA Cawangan Terengganu Kampus Kuala Terengganu, 21080 Kuala Terengganu, Terengganu, Malaysia[c]Department of Mathematical Sciences, Faculty of Science, Universiti Teknologi Malaysia, 81310 Johor Bharu, Malaysia

Abstract Optimization is a process of obtaining optimal solution in solving objective function. Conjugate Gradient (CG) method is known for solving unconstrained optimization problems due to its simplicity, low memory storage and global convergence properties. Over the years, researchers improved classical CG method and introduced hybrid, three term and modified CG method. The hybrid conjugate gradient (HCG) method is the combination of two or more CG coefficients where the rapid computational efficiency of one method is paired with the robust global convergence of another. These CG methods have been implemented in various applications such as data fitting, robotic motion control and Grey Wolf Optimization (GWO) algorithm. GWO is a metaheuristic optimization algorithm that has proven effective in solving multiple optimization problems. However, GWO is said to have poor population diversity, slow convergence in later stages and prone to get stuck in local optimums where it proves the imbalance of exploration and exploitation search. Thus, this research aims to find the best HCG method under exact line search that can be implemented in GWO to improve search process through the hybrid GWO-CG algorithm. The performance of five HCG methods, Linda-Aini-Mustafa-Rivaie (LAMR) coefficient hybrid with Kamil-Mustafa-Aini-Rivaie (KMAR), Norrlaili-Rivaie-Mustafa-Ismail (NRMI), Hestenes–Stiefel (HS), Polak–Ribière–Polyak (PRP) and Liu–Storey (LS) coefficients respectively are tested using 15 standard test functions with different initial points and variables ranging from 2 to 10,000. The numerical results are computed based on number of iteration (NOI) and CPU time. The results are plotted using performance profile to evaluate its efficiency and robustness. Numerically, LAMR-PRP outperforms other methods by solving all test functions with least NOI and CPU time. Lastly, LAMR-PRP is implemented in GWO and resulting effective implementation as it performs better that original GWO.

Keywords: Optimization, Hybrid Conjugate Gradient, Exact Line Search, Grey Wolf Optimization.

***For correspondence:**
lindazullpakkal@uitm.edu.my

Received: 22 November 2025
Accepted: 26 August 2026

© Copyright Luyoh. This article is distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use and redistribution provided that the original author and source are credited.

Introduction

Optimization problems often involve multi-objective functions and constraints have become increasingly complex over time [1]. There are two types of optimization problems which are constrained and

unconstrained problems. The solution to solve the problem is via mathematical calculation of the maximum and minimum values of functions that are mathematically expressed as:

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

where x is element of real numbers, the function is continuously differentiable. Conjugate Gradient (CG) method is the best method to solve large unconstrained optimization problem as it is an iterative method that is known for its simplicity, low memory requirement and global convergence [2]. CG iterative method is expressed as:

$$x_{k+1} = x_k + \alpha_k d_k, \ k = 0,1,2,3,... \tag{2}$$

where x_k represents current iteration point and $\alpha_k > 0$ is the step size where it is determined using line search. The line search can be classified into two which are exact and inexact line search. CG coefficients exhibit different behaviour in different line search. Although inexact line search is widely used since the early days of computational optimization, exact line search has also increased in adaptation in various applications in recent years due to faster computer processors [3]. The exact line search condition is expressed as follow.

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

The d_k is the search direction given by

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

where g_k is the value of gradient and β_k is the CG coefficient. Over the decades, many CG coefficients have been developed and evolved. According to Zull Pakkal et al. [4], CG method can be divided into several classifications such as classical, modified, hybrid, three terms and spectral CG methods. The examples of CG coefficients by Linda-Aini-Mustafa-Rivaie (LAMR), Kamilu-Mustafa-Abdelrahman-Rivaie (KMAR), Norrlaili-Rivaie-Mustafa-Ismail (NRMI), Hestenes and Stiefel (HS), Polak, Ribiére and Polyak (PRP) and Liu and Storey (LS) [5-11] are listed respectively as below:

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

$$\beta_k^{KMAR} = \frac{g_k^T (g_k - g_{k-1})}{g_{k-1}^T (g_k + g_{k-1})} \tag{6}$$

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

$$\beta_k^{HS} = \frac{g_k^T (g_k - g_{k-1})}{d_{k-1}^T (g_k - g_{k-1})} \tag{8}$$

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

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

Hybrid method is a common approach to improve the flaws of certain method [12]. In context of CG method, HCG was first introduced by Touati-Ahmed and Storey [13] where Fletcher-Reeves (FR) and Polak-Ribière (PR) are combined. Then, with the significant improvement from Touati-Ahmed and Storey, numerous research study on HCG has been conducted. Yahaya and Kumam [14] proposed a HCG method by combining LS and Dai-Yuan (DY) coefficients. Then, Yahaya et al. [15] introduced another HCG method by integrating HS and Conjugate Descent (CD) coefficients.

CG methods have been successfully applied in various applications including optimizing metaheuristic algorithms. For instance, the successful development of CG in Whale Optimization Algorithm (WOA) by Shalal and Mitras [16] where CG-WOA gives better result than WOA. This proof motivates on the study of hybrid CG into Grey Wolf Optimization (GWO) algorithm as it shown poor population diversity, slow convergence in later stages and susceptibility to local optima [17]. Thus, GWO-CG is expected to improve the convergence of the original GWO.

Methodology

The comparison study of hybrid LAMR-KMAR, LAMR-NRMI, LAMR-HS, LAMR-PRP and LAMR-LS has been conducted. The algorithm of the CG method is given as follows:

Step 1: Initialize initial point, x_0 and set $k = 0$.

Step 2: Compute HCG methods, LAMR-KMAR, LAMR-NRMI, LAMR-HS, LAMR-PRP and LAMR-LS.

Table 1. List of Formula for each HCG Method.

No.	HCG Method	Formula
1	LAMR-KMAR	$\beta_k^{LAMR-KMAR} = \begin{cases} \beta_k^{LAMR} & 0 \le \beta_k^{LAMR} \le \beta_k^{KMAR} \\ \beta_k^{KMAR} & otherwise \end{cases}$
2	LAMR-NRMI	$\beta_k^{LAMR-NRMI} = \begin{cases} \beta_k^{LAMR} & 0 \le \beta_k^{LAMR} \le \beta_k^{NRMI} \\ \beta_k^{NRMI} & otherwise \end{cases}$
3	LAMR-HS	$\beta_k^{LAMR-HS} = \begin{cases} \beta_k^{LAMR} & 0 \le \beta_k^{LAMR} \le \beta_k^{HS} \\ \beta_k^{HS} & otherwise \end{cases}$
4	LAMR-PRP	$\beta_k^{LAMR-PRP} = \begin{cases} \beta_k^{LAMR} & 0 \le \beta_k^{LAMR} \le \beta_k^{PRP} \\ \beta_k^{PRP} & otherwise \end{cases}$
5	LAMR-LS	$\beta_k^{LAMR-LS} = \begin{cases} \beta_k^{LAMR} & 0 \le \beta_k^{LAMR} \le \beta_k^{LS} \\ \beta_k^{LS} & otherwise \end{cases}$

Step 3: Compute the search direction, d_k based on (4) and stop if $\|g_k\| = 0$.

Step 4: Compute the step size, α_k by using exact line search based on (3).

Step 5: Update new point based on iterative formula in (2).

Step 6: Test convergence and check stopping criterion. Stop the iteration if it satisfies both $f(x_{k+1}) < f(x_k)$ and $\|g_k\| \le 10^{-6}$. Otherwise, return to step 1 with $k = k+1$.

Results and Discussion

The HCG methods are tested numerically and applied in the GWO algorithm.

Numerical Results

Each HCG method is compared using fifteen test functions with random initial points and varies dimensions of 2, 4, 10, 100, 1000, 10000. The numerical result are visualized using performance profile by Dolan and Moré [18]. The performance profile is used to evaluate the efficiency and robustness of the tested methods.

Table 2. List of Test Functions with Initial Points and Dimensions.

No.	Test Functions	Initial Point	Dimensions
1	Six Hump	(3,3), (5,5), (9,9), (13,13)	2
2	Raydan 1	(1,1), (2,2), (3,3), (4,4)	2, 4, 10, 100
3	FLETCHCR	(5,5), (8,8), (15,15), (23,23)	2, 4, 10, 100
4	Hager	(1,1), (3,3), (5,5), (7,7)	2, 4, 10, 100
5	Peturbed Quadratic	(5,5), (10,10), (20,20), (30,30)	2, 4, 10, 100, 1000
6	Diagonal 4	(10,10), (20,20), (30,30), (40,40)	2, 4, 10, 100, 1000
7	Extended Quadratic Penalty QP2	(4,4), (9,9), (14,14), (22,22)	2, 4, 10, 100, 1000
8	Extended White & Holst	(2,2), (9,9), (11,11), (12,12)	2, 4, 10, 100, 1000, 10000
9	Extended Rosenbrock	(4,4), (8,8), (14,14), (22,22)	2, 4, 10, 100, 1000, 10000
10	Extended Freudenstein & Roth	(2,2), (3,3), (4,4), (5,5)	2, 4, 10, 100, 1000, 10000
11	Extended Beale	(1,1), (2,2), (3,3), (4,4)	2, 4, 10, 100, 1000, 10000
12	Extended Himmelblau	(2,2), (6,6), (14,14), (25,25)	2, 4, 10, 100, 1000, 10000
13	Shallow	(2,2), (13,13), (28,28), (39,39)	2, 4, 10, 100, 1000, 10000
14	Extended DENSCHNB	(4,4), (7,7), (16,16), (20,20)	2, 4, 10, 100, 1000, 10000
15	Sphere	(7,7), (13,13), (23,23), (31,31)	2, 4, 10, 100, 1000, 10000

The performance profiles of the number of iterations (NOI) and CPU time are shown in Figure 1 and Figure 2 respectively.

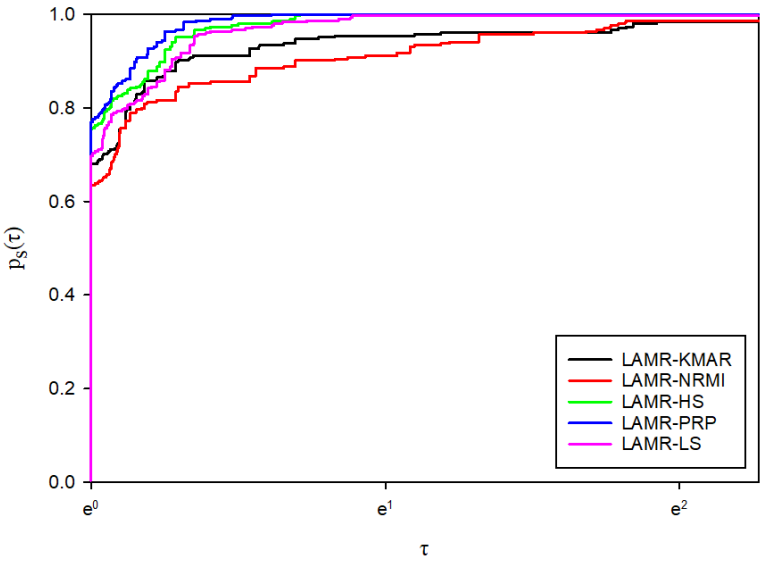


Figure 1. Performance Profile of NOI.

Figure 1 illustrates the performance profile of NOI. Based on the left side of the graph, it displays the efficiency of each method in terms of NOI. The coefficient with the highest curve on y-axis solves more problems with the least NOI. Thus, Figure 1 shows LAMR-PRP is the best method in displaying efficiency with the least NOI.

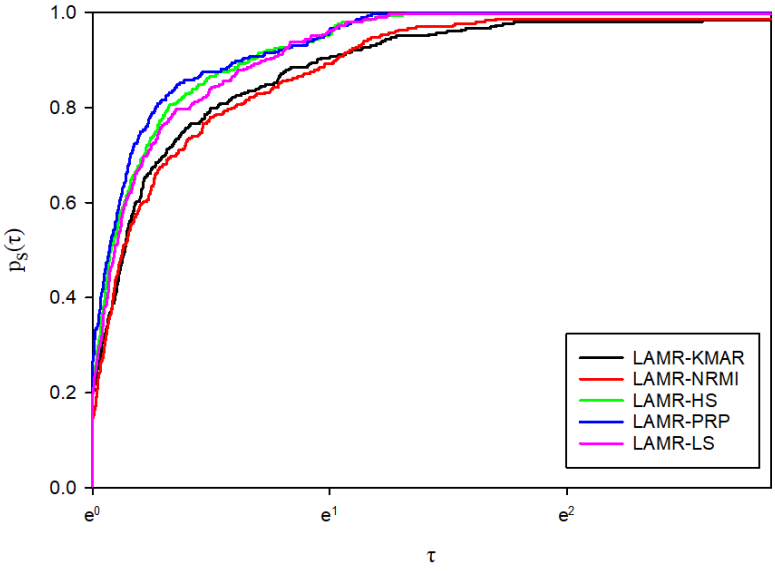


Figure 2: Performance Profile of CPU time.

On the other hand, the left side of the graph in Figure 2 displays the efficiency of the method in terms of CPU time. The coefficient with the highest curve on y-axis solves more problems with the least CPU time. Hence, Figure 2 shows LAMR-PRP is the best method in displaying efficiency with the least CPU time. The upper-right side of the graph in Figure 1 and Figure 2 displays the robustness of tested methods. Both graphs yield the same curve, and the result is tabulated as in Table 2.

Table 3. List of Success Rate for each HCG Method.

No.	HCG Method	Success Rate
1	LAMR-KMAR	98.36%
2	LAMR-NRMI	98.68%
3	LAMR-HS	100%
4	LAMR-PRP	100%
5	LAMR-LS	99.67%

In term of robustness, LAMR-PRP and LAMR-HS outperforms other by solving all the test functions while LAMR-KMAR, LAMR-NRMI and LAMR-LS are able to solve 98.36%, 98.68% and 99.67% of the test functions respectively. The test function is recorded as a failure for the tested method if the number of iterations (NOI) exceeded the maximum threshold of 10,000 or the convergence tolerance $\|g_k\| \leq 10^{-6}$ is not satisfied. These percentages of efficiency and robustness can be calculated manually. As an example, for LAMR-PRP,

i) Determine the minimum value of CPU time for each test function. As an example, six hump function,

$$\min\{0.4633, 0.0814, 0.0728, 0.0265, 0.0346\} = 0.0265$$

ii) Calculate the performance ratio,

$$r_{p,s} = \frac{t_{p,s}}{\min\{t_{p,s} : s \in S\}} = \frac{\text{CPU time}}{\text{min value}}$$

$$r_{p,s} = \frac{0.0265}{0.0265} = 1$$

iii) Calculate the performance ratio by using the formula below to find the most efficient method,

$$\rho_s(\tau) = \frac{size\left\{p \in P : r_{p,t} \le t\right\}}{size\left\{p \in P\right\}} = \frac{\text{Total test function with least CPU time}}{\text{Total Test Function}}$$

$$\rho_s(1) = \frac{80}{304} = 0.2632$$

It shows that LAMP-PRP is able to solve 26.32% of test function with least CPU time.

iv) Calculate the percentage of functions successfully solved by each method to identify the most robust method,

$$\rho_s(\tau) = \frac{304}{304} \times 100 = 100\%$$

Grey Wolf Optimization with Hybrid Conjugate Gradient

The process in GWO-CG is separated into two phases. The exploration phase is the first phase where GWO form random community and handle global search. Then, the second phase, exploitation is handled by CG where the best hybrid LAMR coefficient is applied for local search. This approach aims to balance exploration and exploitation to find the optimal solution. The algorithm of GWO-CG is listed below,

Step 1: Initialize grey wolf population, $x_i (i = 5,10,15)$ and max iteration $MaxIter = 500$.

Step 2: Set iteration counter, $t = 0$ and initialize leading wolves $\alpha, \beta, \delta = \infty$.

Step 3: Evaluate the objective function, $f(x_k)$, apply boundary constraints $X_i = \min(\max(X_i, LB), UB)$ and update α, β, δ positions.

Step 4: Apply the best HCG method to improve the grey wolf population before the position update.
$X_i^* = HCG(X_i)$

Step 5: Update the control parameter using $\alpha = 2 - t\left(\frac{2}{MaxIter}\right)$.

Step 6: Compute coefficient vectors for each α, β, δ using
$A_m = 2a \cdot r_1 - a$
$C_m = 2r_2$, where $m = 1,2,3$

Step 7: Calculate the distance vectors,
$D_k = \left|C_m \cdot X_k - X\right|$, where $k = \alpha, \beta, \delta$

Step 8: Update each grey wolf position using
$X_m = X_k - A_m \cdot D_k$
$X(t+1) = \frac{\sum X_m}{3}$

Step 9: Check the stopping criterion. If $t < MaxIter$, set $t = t + 1$ and return to Step 3. Otherwise, stop the algorithm.

Step 10: Output of the best solution and corresponding objective function values as the result.

Convergence Plot and Test Functions

The performance of GWO and GWO-CG are compared using convergence plot. Both performances are visualized by plotting the best function value found against the NOI. The convergence plot of both

algorithms is compared for 10 test functions which include 30 dimensions from unimodal and multimodal functions. Each function is solved by 5, 10 and 15 search agents over 500 executions in order to enhance the reliability of the results. The different search agents are selected to maintain consistent performance across different population sizes. Thus, minimized value of the objective function indicates the best solution obtained in 500 executions.

Table 4. List of Test Functions for GWO-CG.

No.	Test Functions
1	F_1 = Sphere
2	F_2 = Schwefel's P2.22
3	F_3 = Quadric
4	F_4 = Rosenbrock
5	F_5 = Step
6	F_6 = Ackley
7	F_7 = Generalized Penalized
8	F_8 = Penalized 2
9	F_9 = Complex Exponential-Cosine
10	F_{10} = Sine-Exponential

Comparison Result of GWO and GWO-CG for 5 search agents

The comparison result of GWO and GWO-CG for 5 search agents is shown in Table 4. The result shows that GWO-CG gives smaller value of objective functions compared to GWO.

Table 5. Comparison objective functions result of GWO and GWO-CG using 5 search agents.

No.	GWO	GWO-CG
F_1	3.0161e-09	1.0069e-27
F_2	1.0826e-05	1.3993e-11
F_3	0.54747	2.0343e-14
F_4	28.6387	1.6070e-09
F_5	3.5239	6.9712e-16
F_6	5.9904e-06	1.4321e-11
F_7	0.39322	2.9222e-16
F_8	2.3113	4.3032e-15
F_9	1.7357e-128	4.0196e-140
F_{10}	3.9235e-14	3.1617e-14

The convergence plot of every test function is illustrated in Figure 3 to show comparison between GWO and GWO-CG for 5 search agents.

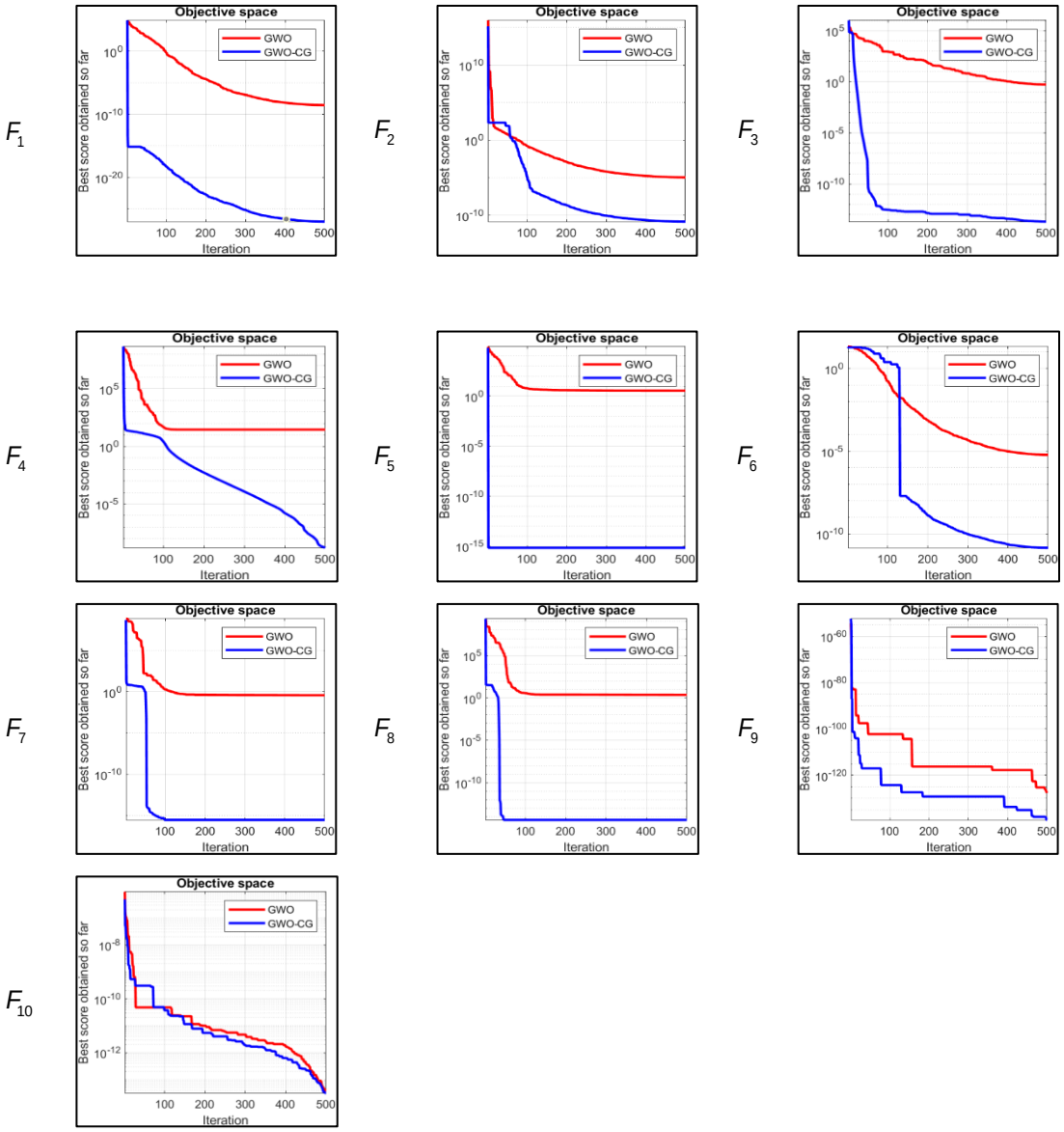


Figure 3. Convergence curve of test function of GWO and GWO-CG for 5 search agents.

Comparison Result of GWO and GWO-CG for 10 search agents

Table 5 shows the comparison result of GWO and GWO-CG for 10 search agents. The result also shows that GWO-CG give smaller value of objective functions compared to GWO.

Table 6. Comparison objective functions result of GWO and GWO-CG using 10 search agents.

No.	GWO	GWO-CG
F_1	7.7656e-16	1.7502e-32
F_2	4.6403e-10	8.81e-17
F_3	0.20076	3.3885e-15
F_4	28.6312	2.3308e-11
F_5	2.1494	6.1447e-16
F_6	4.9608e-09	6.0796e-13

F_7	0.20005	2.2433e-16
F_8	1.4539	3.0431e-15
F_9	1.1235e-128	3.5676e-167
F_{10}	6.399e-15	5.6191e-15

The convergence plot of every test functions is displayed in Figure 4 to show comparison between GWO and GWO-CG for 10 search agents.

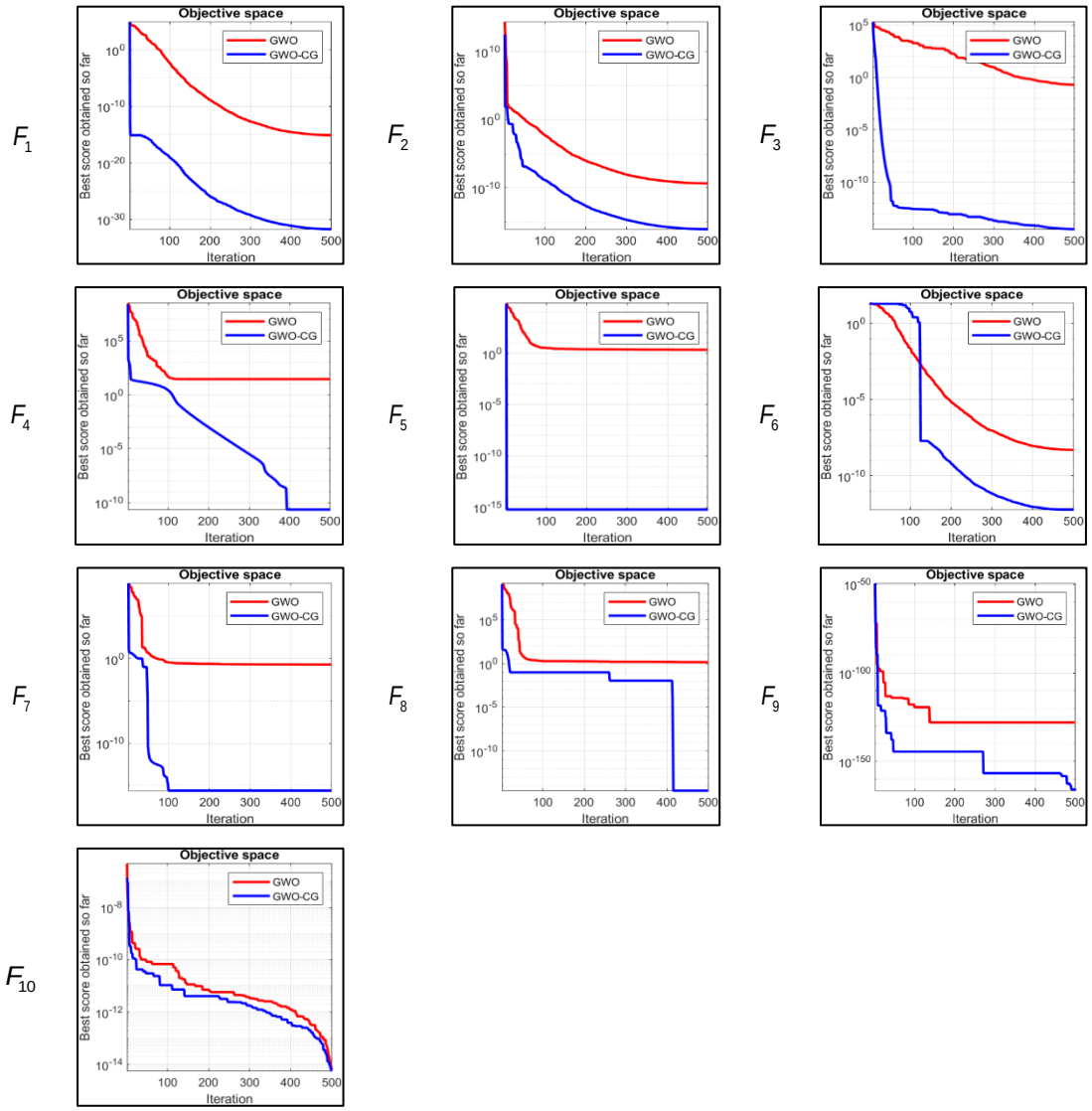


Figure 4. Convergence curve of test function of GWO and GWO-CG for 10 search agents.

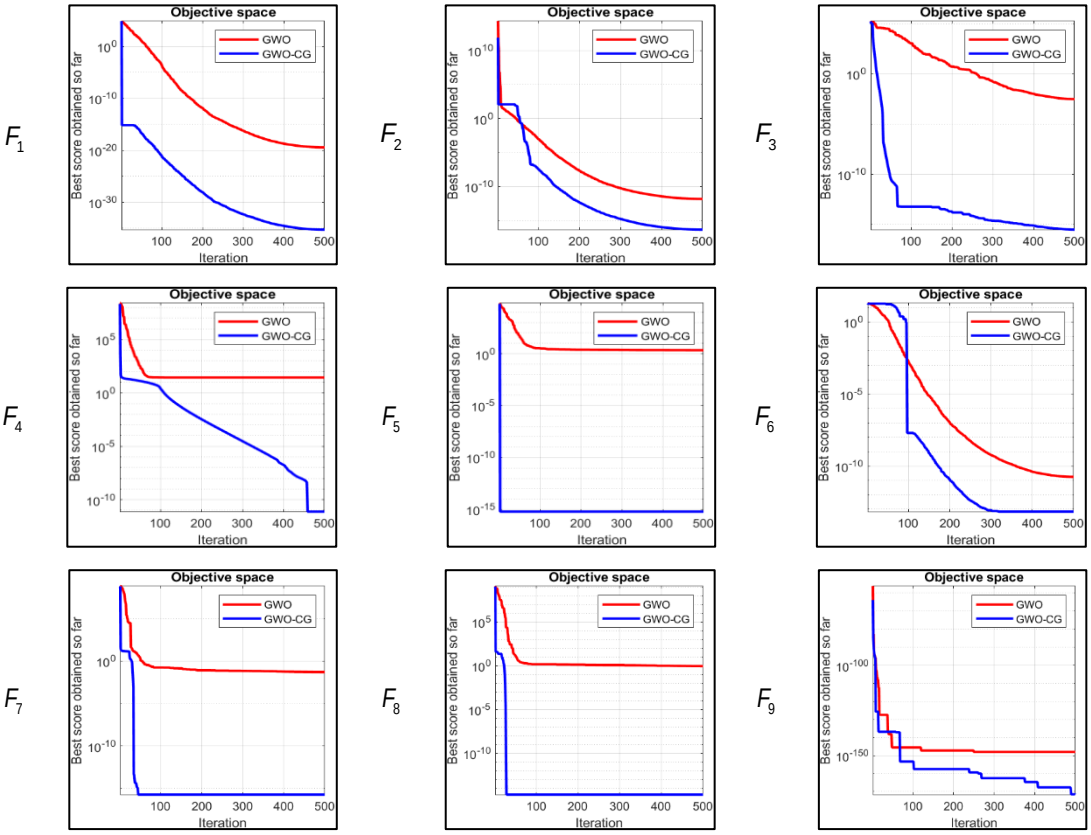
Comparison Result of GWO and GWO-CG for 15 search agents

The comparison result of GWO and GWO-CG for 15 search agents is shown in Table 6. The result also shows that GWO-CG give smaller value of objective functions compared to GWO, the same for 5 and 10 search agents.

Table 7. Comparison objective functions result of GWO and GWO-CG using 15 search agents.

No.	GWO	GWO-CG
F_1	3.5584e-20	5.0516e-36
F_2	1.4355e-12	4.3017e-17
F_3	0.0031245	2.8889e-16
F_4	27.972	7.5272e-12
F_5	1.0035	3.6626e-16
F_6	1.7814e-11	6.7946e-14
F_7	0.049093	1.9593e-16
F_8	0.88769	1.821e-15
F_9	9.3379e-149	2.0109e-172
F_{10}	2.3841e-15	2.1241e-15

Figure 5 illustrates the convergence plot of each test function to show comparison between GWO and GWO-CG for 15 search agents.



F_{10}

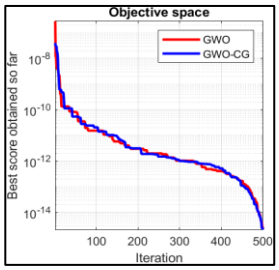


Figure 5. Convergence curve of test function of GWO and GWO-CG for 15 search agents.

Based on Table 4, Table 5 and Table 6, GWO-CG has the least value of objective function for each test function compared to GWO. The smaller value of objective function is the optimal value. This indicates that GWO-CG performs better in achieving the optimal value. On the other hand, Figure 3, Figure 4 and Figure 5, GWO-CG displays the best convergence performance on all test functions. Besides, the figures indicate as the number of search agents increases, the most optimal value decreases due to more extensive exploration in search space. This is resulting in a higher probability of finding better solutions and potentially reaching new optimal results with more search agents. Thus, the result of optimal function is decreasing as the number of search agents increase because the algorithm is converging to better solution.

Conclusions

As a conclusion, LAMR-PRP method under exact line search is the most robust method by solving all the test functions and the most efficient method by taking the vital aspects of NOI and CPU time. The implementation of LAMR-PRP in GWO is a success where GWO-CG performs better than the original GWO algorithm. Therefore, LAMR-PRP can be applied in other application such as image restoration, signal recovery and robotic motion. Besides, this method can be further studied by using inexact line searches and necessary to delve into other HCG combinations like LAMR-RMIL or PRP-CD to assess effectiveness of the method.

Conflicts of Interest

The author(s) declare(s) that there is no conflict of interest regarding the publication of this paper.

Acknowledgment

The authors would like to express their sincere gratitude to Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA Cawangan Terengganu Kampus Kuala Terengganu for providing the necessary resources and support throughout this research. Special thanks are extended to colleagues and mentors for their invaluable assistance and guidance, as well as to any funding organizations that contributed to this project.

References

[1] P. K. Mandal, *Results Control Optim.* 10 (2023) 100315.
[2] N. Salihu, P. Kumam, I. M. Sulaiman, I. Arzuka, W. Kumam, *Math. Comput. Simul.* 226 (2024) 354.
[3] M. Rivaie, M. Mamat, A. Abashar, *Appl. Math. Comput.* 268 (2015) 1152. [4] Zull Pakkal, N., Shapiee, N., Husin, S. F., Khadijah, W., Salahudin, N. A., Mohd Zokri, S., and Rashidi, M. N. The implementation of conjugate gradient methods for data fitting. *Data Analytics and Applied Mathematics (DAAM).* 2023: 23-29.
[4] N. Zull Pakkal, N. Shapiee, S. F. Husin, W. Khadijah, N. A. Salahudin, S. Mohd Zokri, M. N. Rashidi, *Data Anal. Appl. Math.* 2023 (2023) 23.
[5] N. Zull Pakkal, N. Aini, S. Shoid, N. Abdul Ghani, N. Mohamed, M. Rivaie, M. Mamat, in: *Journal of Physics: Conference Series*, IOP Publishing, Bristol, UK, 2017, p. 012105.
[6] K. Uba Kamfa, M. Mamat, A. Abashar, M. Rivaie, P. Ghazali, Z. Salleh, *Appl. Math. Sci.* 9 (2015) 1833.
[7] N. Shapiee, M. Rivaie, M. Ali, M. Mamat, Z. Salleh, A New Simple Conjugate Gradient Coefficient for Unconstrained Optimization, 2015.
[8] M. R. Hestenes, E. Stiefel, *J. Res. Natl. Bur. Stand.* 49 (1952) 409.
[9] E. Polak, G. Ribiere, *Rev. Fr. Inf. Rech. Oper. Ser. Rouge* 3 (1969) 35.

[10] B. T. Polyak, *USSR Comput. Math. Math. Phys.* 9 (1969) 94.
[11] Y. Liu, C. Storey, *J. Optim. Theory Appl.* 69 (1991) 129.
[12] E. Eriksson, E. M. Gray, *Appl. Energy* 202 (2017) 348.
[13] D. Touati-Ahmed, C. Storey, *J. Optim. Theory Appl.* 64 (1990) 379.
[14] J. Yahaya, P. Kumam, *Comput. Appl. Math.* 44 (2025) 163.
[15] J. Yahaya, P. Kumam, M. Yahaya, *Oper. Res. Forum* 6 (2025) 1.
[16] O. D. Shalal, B. A. Mitras, *Samarra J. Pure Appl. Sci.* 6 (2024) 239.
[17] J. Chai, L. Zhang, T. Liu, The Development and Application Research of Grey Wolf Optimization Algorithm, 2024.
[18] E. D. Dolan, J. J. Moré, *Math. Program.* 91 (2002) 201.