MATEMATIKA, MJIAM, Volume 41, Number 1, 123–134

Derivative-Free Hybrid Conjugate Gradient Method for Constrained Nonlinear Equations with Projection in Line Searches

[1]Nor Deanna Syahirah Mohd Sham and [2]Nur Syarafina Mohamed*

[1]Department of Mathematical Sciences, Faculty of Science, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia.

*Corresponding author: nursyarafina@utm.my

Article history
Received: 26 October 2024
Received in revised form: 5 March 2025
Accepted: 12 March 2025
Published online: 20 April 2025

Abstract Robustness, efficiency, and accuracy are qualities that excellent algorithms should have. Due to the simplicity and minimal storage requirements, Conjugate gradient (CG) methods are useful for solving large-scale, unconstrained optimization problems. Despite that, it has a few drawbacks. Even if they have high numerical performance, certain approaches lack global convergence properties; therefore, the solutions might not be the most accurate. Various methods and modifications have been done. Some formulations would be difficult to comprehend and apply, and would lead to high CPU time. The proving process would also be impacted by the complex formulations. Over the past years, researchers have developed various globally convergent CG methods, but with a complicated algorithm, it rather hampered the implementation. Therefore, new CG methods with a derivate-free approach that have good convergence properties and outperform the existing CG coefficients in terms of number of iterations (NOI), number of function evaluations (NFE), and central processing time per unit (CPU time) are proposed. The proposed method will employ a non-derivative approach. This approach should make the algorithm's processing time as minimal as possible. The comparison for derivative-free tools among the existed derivative-free CG. The proposed approach was chosen because it combines the strengths of the CG method with derivate-free optimization to optimize complicated objective functions without explicitly computing derivatives. This paper will show the derivate-free CG, which was proven to fulfil both convergence analysis and numerical performance.

Keywords Conjugate Gradient Method, Derivative-Free, Hybrid, Nonlinear Equations, Projection Method.

Mathematics Subject Classification 65K05, 65K10, 90C25, 90C26, 49M30, 93E99.

1 Background/Objective and Goals

Developing a hybrid CG approach that is derivative-free and effectively optimizes nonlinear equations without requiring derivative information is a challenge. In order to obtain rapid convergence and robust performance across a variety of optimization problems, this approach tries to combine the adaptability of derivative-free optimization techniques with the computing efficiency of CG approaches. Large-scale problems can be effectively optimized using classical CG methods, but it usually need derivative information, which isn't always available or accurate. While derivative-free optimization techniques present an alternative approach, it might not possess the same efficiency and convergence properties as CG methods. A hybrid approach that combines the strengths of each might offer a powerful derivative-free optimization solution. In accordance with its derivative-free approach, the coefficient based on hybrid CG methods was given, which has strong convergence qualities, validity, reliability, accuracy, and outperforms the current CG coefficients in terms of NOI and CPU time [1, 2]. This approach should make the algorithm's processing time as minimal as possible.

The goal of this study is to discover ways to enhance the hybrid CG method to answer the aforementioned problems, which are complicated formulations, a high number of CPUs, and NOI. In order to demonstrate their effectiveness and highlight the benefits for the fourth industrial revolution, we will investigate how the suggested approach may be applied to image restoration problem. This study aims to propose a derivative-free approach in hybrid CG method, specifically using the Hybrid-Syarafina-Mustafa-Rivaie (HSMR) method [3]. HSMR method is a combination of Rivaie, Mustafa, Ismail, and Leong (RMIL) method [4] and Syarafina, Mustafa, Rivaie (SMR) [5] while using under exact line search. The HSMR method has outperformed the other hybrid CG methods in terms of NOI and CPU time. A modified hybrid CG method that neglects the computation of function gradients is called the derivative-free hybrid CG method. This renders it particularly useful in cases when computing the gradient is either complicated or costly. The comparison for derivative-free tools is concentrated on the derivative-free SMR (DF-SMR) method [2], and the proposed method of this study. The proposed method was chosen because it combines the strengths of the CG method with derivative-free optimization to optimize complicated objective functions without explicitly computing derivatives.

A graphical representation of a different optimization algorithm's performance on a set of problems is called a performance profile. It also compares the effectiveness of different algorithms and determining which one works best for a given problem may be done using this helpful method [6]. Plotting the performance profiles of different algorithms on a single graph is the method of performance profile comparison. The performance ratio, which is the ratio of the algorithm's time to the best algorithm's time, is shown by the x-axis of the graph. The y-axis is showing the proportion of problems that the algorithm resolves within a certain tolerance range. When evaluating the performance of different algorithms on a variety of problems using the performance profile, the best algorithm for a given problem may be identified. The best algorithm is the one that has the highest ratio of performance and the highest percentage of problems resolved within the tolerance limit.

This study aims to improve the HSMR CG method by modifying the CG coefficient, which derives from the CG formula, which is well-known for its good convergence features. Two components are necessary to obtain high overall performance: restart property at the numerator [7, 8] and simple expression at the denominator in the form of descent direction or gradient vector [4, 9]. Furthermore, the derivative-free HSMR (DF-HSMR) is chosen where the non-derivative approach is required. The proposed method is used to solve large-scale nonlinear equations with convex constraints. The aim is to improve hybrid CGs performance in solving larger scale problems while also saving a significant amount of time due to its derivative-free algorithm. The approach is designed to be more efficient in tackling constrained optimization test problems as well as real-life problems. Moreover, as demonstrated theoretically and numerically, the method must possess sufficient descent and global convergence properties. Therefore, the effectiveness of the proposed method in resolving constrained optimization problems, its convergence properties, and its applicability for real-life applications in image restoration [10, 11].

2 Methods

A numerical approach for solving large-scale nonlinear systems of equations without use of function derivatives is the derivative-free CG methods [12], and it is established on the notion of minimizing a quadratic model of the function along a conjugate direction [13]. The goal of this technique is to overcome the obstacles in calculating or approximating derivatives that could be expensive, inaccurate, or unavailable for certain problems [13], and the suitability of this method lies in its ability to handle nonsmooth or noisy functions that cannot be handled by derivative-requiring methods. When using this method, one should specify an initial point along with a stopping criterion while also considering relevant parameters for both the line-search and conjugate gradient techniques. Under some mild assumptions, the method generates a sequence of iterates that converge to a solution of the nonlinear system [12]. One of the main benefits of using this method is that it is simple to apply without requiring any derivatives. Furthermore, it guarantees global convergence. Efficient and robust handling of large-scale problems is another capability it has [14].

The performances of each solver are often shown and compared in a table based on the CPU time, NOI, and NFE. Because there are a large number of data sets, this method of interpreting the results sometimes led to a point of disagreement. In order to solve this problem, various tools for analyzing the data had been studied. Billups, Dirkse, and Ferris developed the first performance comparison method [15]. The ratio of one solver's runtime to the best runtime is used to compare solvers. The solvers are ranked according to the percentage of problems for which their time is either very competitive or competitive.

The SMR method has been proposed by Mohamed *et al.* in 2016 [5]. The denominator of β_k^{SMR} maintains the same form as that of β_k^{RMIL}, and the method proposed is based on the idea of the RMIL method. Also changed from the β_k^{RMIL} is the numerator. Presented below is the coefficient of SMR:

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

where β_k is the CG coefficient, g_k is the gradient, and d_k is the search direction. According to the equation above, the maximum feature causes the value of β_k^{SMR} to automatically revert to zero when the second expression tends to the negative values. The aim of this equation is to enhance the method by avoiding the negative values that occurred in RMIL. The classical version of the CG method, known as SMR, is an extension of RMIL. The hybrid CG method was proposed as an enhancement of the SMR method [16]. β_k^{RMIL} [4] and β_k^{SMR} [5] are combined to form HSMR. In order to prevent jamming, the idea is to combine different CG algorithms to create a new hybrid convex-combination method [17]. The SMR method has good computational properties under exact line search, whereas RMIL has strong convergence properties [4, 5, 18]. A more improved and useful method is produced by combining all the desirable parameters. A hybrid CG that combines RMIL and SMR is suggested.

$$\beta_k^{HSMR} = max\left\{0, min\left\{\beta_k^{SMR}, \beta_k^{RMIL}\right\}\right\}$$

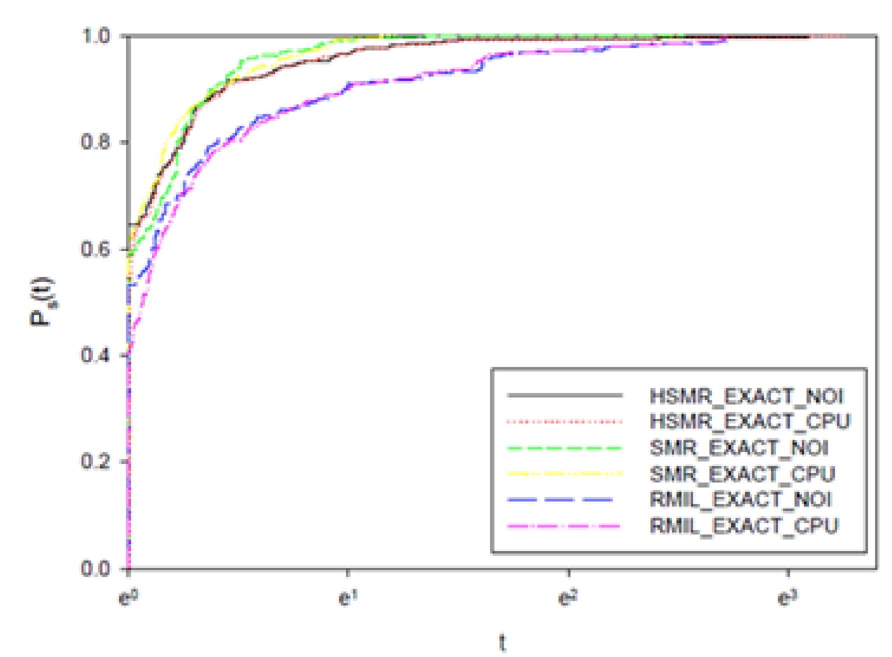

Figure 1: Performance Profile for SMR, HSMR, and RMIL.

As can be seen in Figure 1, the performance profile compares each coefficient performance under exact line search in terms of NOI and CPU time. The top right curve shows the coefficient's ability to solve the test functions utilized, while the top left curve shows the coefficient's speed in reaching the solution point. These curves demonstrate that, for exact line search, SMR and HSMR perform better than RMIL, and the outcomes demonstrate that the hybrid version performs better as it inherited the good characteristics from both SMR and RMIL.

Large-scale nonlinear systems of equations can be solved numerically using the derivative-free CG method [12]. When the function's derivative or gradient is unavailable or too costly to compute, this approach is especially helpful. The CG method is a best option for handling large-scale unconstrained optimization problems due to its simplicity and does not require a large amount of storage space [19]. The CG method and the projection method's benefits are combined in this derivative-free approach. The derivative-free HSMR method is an enhancement of the HSMR method [3]. Apply β_k^{EHSMR} in place of β_k^{ESMR} in the DF-SMR method's search direction [2]. Using the same algorithm, it can be used to distinguish between DF-SMR method and the DF-HSMR method. Nonlinear equation systems with convex constraints can be expressed mathematically as

$$v(u) = 0, u \in \Omega, \tag{1}$$

where $v : \Omega \longrightarrow \mathbb{R}$ is a continuous nonlinear mapping and $\Omega \subseteq \mathbb{R}^{\ltimes}$ is a closed convex set.

The HSMR conjugate gradient method, which will be reviewed below, is the base of the proposed method. Mohamed *et al.* [3] suggested using the HSMR method to solve the following unconstrained optimization problem

$$min \{z(u) : u \in \mathbb{R}^{\ltimes}\} \tag{2}$$

where $z : \mathbb{R}^{\ltimes} \longrightarrow \mathbb{R}$ is a continuous differentiable function. Let the gradient of z at u_t be represented by $\nabla z(u_t)$. A sequence of iterates $\{u_t\}$ is produced using the [3] approach using the recursive formula below

$$u_{t+1} = u_t + s_t d_t, t \geq 0 \tag{3}$$

where u_0 is specified to be the sequence's starting point and u_t is the current iterative point. According to (3), $s_t > 0$ is referred to the step size and d_t is the search direction defined by the rule:

$$d_t = \begin{cases} -\nabla Z(u_t), & \text{if } t = 0, \\ -\nabla Z(u_t) + \beta_t^{HSMR} d_{t-1}, & \text{if } t > 0, \end{cases} \tag{4}$$

where β_t^{HSMR} , the conjugate gradient parameter, is defined as

$$\beta_t^{HSMR} = max \left\{0, min \left\{\beta_t^{SMR}, \beta_t^{RMIL}\right\}\right\}. \tag{5}$$

As can be seen, not all t are likely to descend in the search direction d_t, which is indicated by (4). Select a vector from a subspace $\mu_t = \left\{p | v_t^T p = 0\right\}$ to substitute for the second phrase $\beta_t^{HSMR} d_{t-1}$ of the direction (4) in order to maintain decency. Therefore, it obtains

$$d_t = -v_t + p, p \in \mu_t. \tag{6}$$

This is undoubtedly inspired by the Gram-Schmidt (MGS) procedure, where the direction used in [20, 21].

Definition 1: Let $\Omega \subseteq \mathbb{R}^{\ltimes}$ be a closed convex set that is not empty. The projection of any given $y \in \mathbb{R}^{\ltimes}$ onto Ω, represented by $P_\Omega[y]$, is thus defined by

$$P_\Omega[y] = \arg\min\{\|y - x\| : x \in \Omega\}. \tag{7}$$

One well-known characteristic of the projection operator P_Ω, $x \in \mathbb{R}^{\ltimes}$ is that the following nonexpansive property holds for any y

$$\|P_\Omega[y] - P_\Omega[x]\| \leq \|y - x\|. \tag{8}$$

The new algorithm of the DF-HSMR method is given as follows:

Algorithm

Input. Set up an initial point $u_0 \in \Omega$,
the positive constants: $Tol > 0$
$\rho \in (0,1), x \in (0,2), a > 0, \sigma > 0$.
Set t=0.

Step 0. Compute v_t. If $v_t \leq Tol$, stop.
Alternatively, generate the search direction d_t using the following

$$d_t = \begin{cases} -v_t, & \text{if } t = 0, \\ -v_t + \beta_t^{EHSMSR}\left(d_{t-1} - \frac{v_t^T d_{t-1}}{\|v_t\|^2} j_t\right), & \text{if } t > 0, \end{cases} \tag{9}$$

where β_t^{EHSMSR} is computed by (5).

Step 1. Determine the step size $s_t = max\{a\rho^j | j \geq 0\}$ such that

$$v(u_t + s_t d_t)^T d_t \geq \sigma s_t ||d_t||^2 \tag{10}$$

Step 2. Compute $r_t = u_t + s_t d_t$, where r_t is a trial point.

Step 3. If $r_t \in \Omega$ and $v(r_t) = 0$, stop.
Alternatively, compute the next iteration by

$$u_{t+1} = P_\Omega\left[u_t - x\frac{v(r_t)^T(u_t - r_t)}{\|v(r_t)\|^2} v(r_t)\right],$$

Step 4. Finally, set $t = t + 1$ and return to step 1.

3 Numerical Experiments

In this section shows the numerical performance of the proposed method, DF-HSMR, compared to DF-SMR, using the Dolan and More performance profile [22]. The NOI, NFE, and CPU time are among the metrics that are taken into consideration using the Dolan and More performance profiles. The same algorithm that was proposed in [2] is applied in this study. Ten test problems were chosen, and for each test problem, seven initial starting points were picked. All codes were coded using MATLAB R2024b and executed on a desktop running Windows 11 and equipped with an AMD Ryzen 5 5600H processor, 16.0 GB of RAM, and 3.30 GHz CPU. The following elements are considered in the experiments:

- Parameters for derivative-free HSMR,pick $a = 1, \rho = 0.8, \sigma = 10^{-4}, x = 1.2, Tol = 10^{-6}$ same as derivative-free SMR [2].
- Dimensions: 1 000, 5 000, 10 000, 50 000, 100 000.
- Initial points:
  $u_1 = (0.1, 0.1, \ldots, 0.1)^T, u_2 = (0.2, 0.2, \ldots, 0.2)^T$,
  $u_3 = (0.5, 0.5, \ldots, 0.5)^T, u_4 = (1.2, 1.2, \ldots, 1.2)^T, u_5 = (1.5, 1.5, \ldots, 1.5)^T$,
  $u_6 = (2, 2, \ldots, 2)^T, u_7 = rand(0, 1)$.

The test problems with $v = (v_1, v_2, \dots, v_n)$ are listed below:
Problem 3.1 [23].Exponential function:

$$v(u) = e^{u_1} - 1, v_i(u) = e^{u_i} + u_i - 1, \quad \text{for} \quad i = 2, 3, \dots, n, \quad \text{and} \quad \Omega = \mathbb{R}^n_+.$$

Problem 3.2 [23].Modified logarithmic function:

$$v_i(u) = \ln(u_i + 1) - \frac{u_i}{n}, \quad \text{for } i = 1, 2, 3, \dots, n,$$
$$\Omega = \left\{ u \in \mathbb{R}^n : \sum_{i=1}^{n} u_i \leq n,\ u_i > -1,\ i = 1, 2, \dots, n \right\}.$$

Problem 3.3 [24].The elements of the function $v(i)$:

$$v_i(u) = 2u_i - \sin(u_i), \text{ for } i = 1, 2, \dots, n.$$

Problem 3.4 [25].Discrete boundary value:

$$\begin{aligned} v_1(u) &= 2u_1 + 0.5m^2(u_1 + m)^3 - u_2, \\ v_i(u) &= 2u_i + 0.5m^2(u_i + mi)^3 - u_{i-1} + u_{i+1}, \quad \text{for } i = 2, 3, \dots, n-1, \\ v_n(u) &= 2u_n + 0.5m^2(c_n + mn)^3 - u_{n-1}, \\ m &= \frac{1}{(n+1)}. \end{aligned}$$

Problem 3.5 [23].Strictly convex function I:

$$v_i(u) = e^{u_i} - 1, \text{ for } i = 1, 2, \dots, n.$$

Problem 3.6 [23].Strictly convex function II:

$$v_i(u) = \frac{i}{10} e^{u_i} - 1, \text{ for } i = 1, 2, \dots, n.$$

Problem 3.7 [26].Tridiagonal exponential function:

$$\begin{aligned} v_1(u) &= u_1 - e^{\cos(m(u_1 + u_2))}, \\ v_i(u) &= u_i - e^{\cos(m(u_{i-1} + u_i + u_{i+1}))}, \quad \text{for } i = 2, \dots, n-1, \\ v_n(q) &= u_n - e^{\cos(m(u_{n-1} + u_n))}, \\ m &= \frac{1}{n+1}. \end{aligned}$$

Problem 3.8 [27].Nonsmooth function:

$$v_i(u) = u_i - \sin|u_i - 1|, \quad \text{for } i = 1, 2, 3, \dots, n,$$
$$\Omega = \left\{ u \in \mathbb{R}^n : \sum_{i=1}^{n} u_i \leq n,\ u_i \geq -1,\ i = 1, 2, \dots, n \right\}.$$

Problem 3.9 [23].Trigexp function:

$$\begin{aligned} v_1(u) &= 3u_1^3 + 2u_2 - 5 + \sin(u_1 - u_2)\sin(u_1 + u_2), \\ v_i(u) &= 3u_i^3 + 2u_{i+1} - 5 + \sin(u_i - u_{i+1}) + 4u_i - u_{i-1}e^{(u_{i-1}-u_i)} - 3, \\ &\quad \text{for } i = 2, \ldots, n-1, \\ v_n(q) &= -u_{n-1} - e^{(u_{n-1}-u_n)} + 4u_n - 3. \end{aligned}$$

Problem 3.10 [28].Penalty I:

$$t_i = \sum_{i=1}^{n} u_i^2, r = 10^{-5}, v_i(u) = 2r(u_i - 1) + 4(t_i - 0.25)u_i, \text{for} i = 1, 2, 3, ..., n.$$

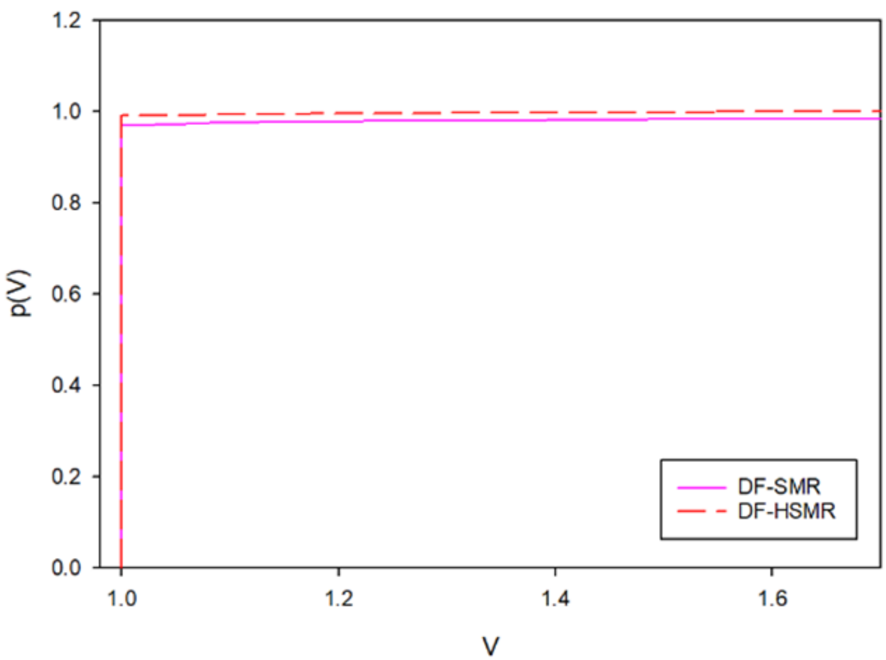

Figure 2: Performance profiles based on number of iterations (NOI).

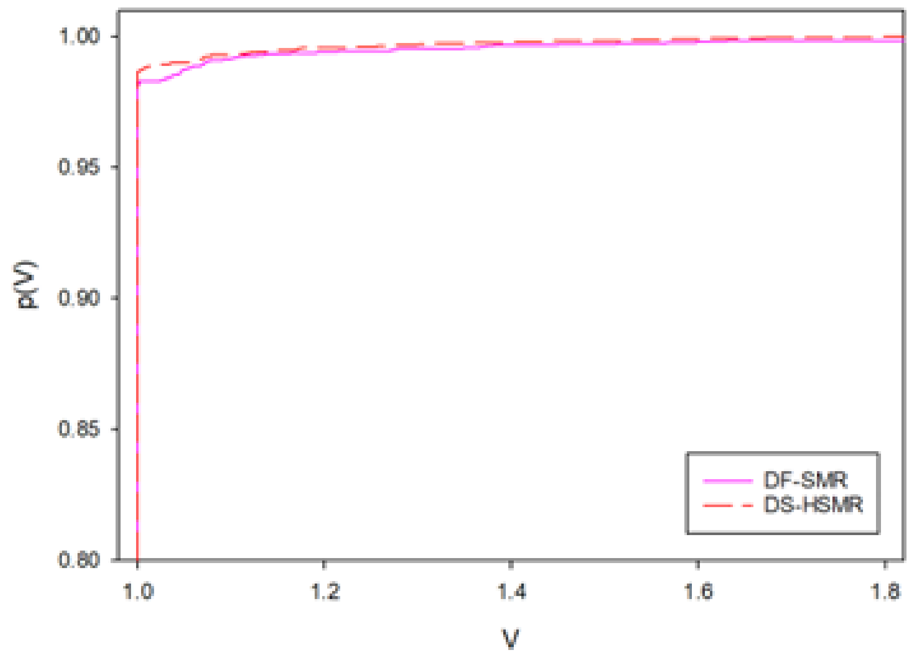

Figure 3: Performance profiles based on number of function evaluations (NFE).

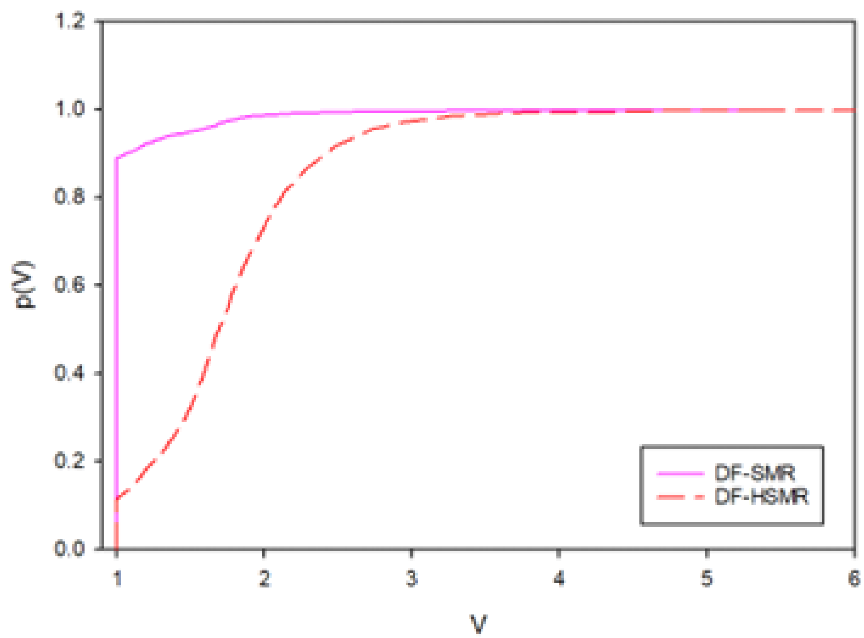

Figure 4: Performance profiles based on CPU time (in seconds).

In the performance profile, the p(V) represents the fraction of problems where the method performs within a factor V. Figure 2-4 shows the plotting of the DF-SMR and DF-HSMR with comparison to NOI, NFE, and CPU time. DF-HSMR has considerably outperformed DF-SMR in terms of NOI, NFE, and CPU time. It is evident from Figure 2, the NOI performance profile, that DF-HSMR performs better than DF-SMR, since DF-HSMR can solve 99% of the test problems with fewer iterations, while DF-SMR can solve 96% of the test problems with fewer iterations. Since DF-HSMR solves approximately 99% of the test problems better than DF-SMR, and DF-SMR solves approximately 98% of the test problems better, it is clear from Figure 3 that DF-HSMR also performs better than DF-SMR in terms of NFE. The performance profile as determined by CPU time is shown in Figure 4. The top curve represents the DF-SMR that resolved the most problems within a factor V of the optimal time. Specifically, DF-SMR gets around 88% of the test problems solved with the least CPU time, whereas DF-HSMR gets about 11%. Even though DF-SMR is faster than DF-HSMR, regarding to the ability to solve the test problems, DF-HSMR is better than DF-SMR.

4 Conclusion

In this paper, the hybrid CG methods and derivative-free methods are all useful tools for optimization problems, each with its own set of advantages and uses. The derivative-free HSMR method is chosen to be explored and improved for dealing with complicated optimization issues, which contributes to the improvement of optimization algorithms. Since this is a new approach on derivative-free using the hybrid CG method, there is still potential for improvement in using this method.

5 Future Work

The proposed method should be tested for global convergence and further modifications for real-life applications in image restoration.

Acknowledgments

This work was supported by Universiti Teknologi Malaysia (UTM) for the UTM Fundamental Research (UTMFR) Grant with vot number Q.J130000.3854.22H10.

References

[1] Dauda, M., Mamat, M., Mohamad, F.S., Magaji, A.S. and Waziri, M. Derivative Free Conjugate Gradient Method via Broyden's Update for solving symmetric systems of nonlinear equations. *Journal of Physics*. Conference Series, 2019. 1366(1): p. 012099.

[2] Ibrahim, A.H., Muangchoob, K., Mohamed, N.S. and Abubakar, A.B. Derivative-free SMR conjugate gradient method for con-straint nonlinear equations. *Journal of Mathematics and Computer Science*, 2022. 24(2): p. 147–164.

[3] Mohamed, N.S., Mamat, M., Rivaie, M. and Shaharudin, S.M. A new hyhbrid coefficient of conjugate gradient method. *Indonesian Journal of Electrical Engineering and Computer Science*, 2020. 18(3): p. 1454.

[4] 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*, 2012. 218(22): p. 11323–11332.

[5] Mohamed, N.S., Mamat, M., Mohamad, F.S. and Rivaie, M. new coefficient of conjugate gradient methods for nonlinear unconstrained optimization. *J. Teknol*, 2016. 78(6-4): p. 131–136.

[6] Sofi, A., Mamat, M., Mohid, S.Z., Ibrahim, M.A.H. and Khalid, N. Performance profile comparison using matlab. in *Proceedings of International Conference on Information Technology & Society*. 2015.

[7] Hestenes, M.R. and Stiefel, E. Methods of conjugate gradients for solving linear systems. *Journal of research of the National Bureau of Standards*, 1952. 49: p. 409–436

[8] Polak, E. and Ribiere, G. Note sur la convergence de mthodes de directions conjugues. Revue franaise d'informatique et de recherche oprationnelle. *Srie rouge*, 1969. 3(16): p. 35–43.

[9] Fletcher, R. and Reeves, C.M. Function minimization by conjugate gradients. *The computer journal*, 1964. 7(2): p. 149–154.

[10] Hassan Ibrahim, A., Kumam, P., Hassan, B.A., Abubakar, A.B. and J. Abubakar, A derivative-free three-term HestenesStiefel type method for constrained nonlinear equations and image restoration. *International Journal of Computer Mathematics*, 2022. 99(5): p. 1041-1065.

[11] Sulaiman, I.M., Malik, M., Awwal, A.M., Kumam, P., Mamat, M. and S. Al-Ahmad, On three-term conjugate gradient method for optimization problems with applications on COVID-19 model and robotic motion control. *Advances in Continuous and Discrete Models, 2022*. 2022(1): p. 1.

[12] Fang, X. and Ni, Q. A new derivative-free conjugate gradient method for large-scale nonlinear systems of equations. *Bulletin of the Australian Mathematical Society*, 2017. 95(3): p. 500–511.

[13] Waziri, M.Y. and Sabi'u, J. A derivative-free conjugate gradient method and its global convergence for solving symmetric nonlinear equations. *International Journal of Mathematics and Mathematical Sciences*, 2015. 2015: p. 1–8.

[14] Wright, S.J., Numerical optimization. 2006, New York: Springer.

[15] Billups, S.C., Dirkse, S.P. and Ferris, M.C. A comparison of large scale mixed complementarity problem solvers. *Computational Optimization and Applications*, 1997. 7: p. 3–25.

[16] Mohamed, N.S., Mamat, M., Rivaie, M. and Shaharudin, S.M. A comparison on classical-hybrid conjugate gradient method under exact line search. *International Journal of Advances in Intelligent Informatics*, 2019. 5(2): p. 150–168.

[17] Andrei, N., Hybrid conjugate gradient algorithm for unconstrained optimization. *Journal of Optimization Theory and Applications*, 2009. 141: p. 249–264

[18] Dong, X.L., Liu, H.W., He, Y.B. and Yang, X.M. A modified HestenesStiefel conjugate gradient method with sufficient descent condition and conjugacy condition. *Journal of Computational and Applied Mathematics*, 2015. 281: p. 239–249.

[19] Ibrahim, A.H., Alshahrani, M. and Al-Homidan, S. Two classes of spectral three-term derivative-free method for solving nonlinear equations with application. *Numerical Algorithms*, 2023: p. 1–21.

[20] Feng, D., Sun, M. and Wang, X. A family of conjugate gradient methods for large-scale nonlinear equations. *Journal of Inequalities and Applications*, 2017. 2017(1): p. 236.

[21] Sun, M. and Liu, J. Three derivative-free projection methods for nonlinear equations with convex constraints. *Journal of Applied Mathematics and Computing*, 2015. 47(1): p. 265–276.

[22] Dolan, E.D. and Mor, J.J. Benchmarking optimization software with performance profiles. *Mathematical programming*, 2002. 91: p. 201–213.

[23] Cruz, W. and Raydan, M. Nonmonotone Spectral Methods for Large-Scale Nonlinear Systems. *Optimization Methods & Software - OPTIM METHOD SOFTW*, 2003. 18: p. 583–599.

[24] Zhou, W. and Li, D. Limited memory BFGS method for nonlinear monotone equations. *Journal of Computational Mathematics*, 2007. 25(1): p. 89–96.

[25] More, J.J., Garbow, B.S. and Hillstrom, K.E. Testing unconstrained optimization software. *ACM Transactions on Mathematical Software*, 1981. 7(1): p. 17–41.

[26] Yang, B. and Lin, G. An Efficient Implementation of Merrill's Method for Sparse or Partially Separable Systems of Nonlinear Equations. *SIAM Journal on Optimization*, 1991. 1(2): p. 206–221.

[27] Yu, Z., Lin, J., Sun, J., Xiao, Y., Liu, L. and Li, Z. Spectral gradient projection method for monotone nonlinear equations with convex constraints.*Applied Numerical Mathematics*, 2009. 59(10): p. 2416–2423.

[28] Ding, Y., Xiao, Y. and Li, J. A class of conjugate gradient methods for convex constrained monotone equations. *Optimization*, 2017. 66(12): p. 2309–2328.