A conjugate gradient method with descent properties under strong Wolfe line search

N Zull¹, N ‘Aini², S Shoid³, N H A Ghani⁴, N S Mohamed⁵, M Rivaïe⁶ and M Mamat⁷
¹,²,³,⁴,⁵,⁷Faculty of Informatics & Computing, Universiti Sultan Zainal Abidin (Kampus Gong Badak), 21300 Kuala Terengganu, Terengganu, Malaysia
⁶Department of Computer Science & Mathematics, Universiti Teknologi MARA (Kampus Kuala Terengganu), 21080 Kuala Terengganu, Terengganu, Malaysia

E-mail: lindazullpakkal@gmail.com

Abstract. The conjugate gradient (CG) method is one of the optimization methods that are often used in practical applications. The continuous and numerous studies conducted on the CG method have led to vast improvements in its convergence properties and efficiency. In this paper, a new CG method possessing the sufficient descent and global convergence properties is proposed. The efficiency of the new CG algorithm relative to the existing CG methods is evaluated by testing them all on a set of test functions using MATLAB. The tests are measured in terms of iteration numbers and CPU time under strong Wolfe line search. Overall, this new method performs efficiently and comparable to the other famous methods.

1. Introduction
The general unconstrained optimization problem is defined by the following rule,

$$\min_{x \in \mathbb{R}^n} f(x),$$  \hspace{1cm} (1.1)

where \( f : \mathbb{R}^n \rightarrow \mathbb{R} \) is continuously differentiable and \( \mathbb{R}^n \) denotes an \( n \)-dimensional Euclidean space. It is commonly solved by iterative method defined as follows

$$x_{k+1} = x_k + \alpha_k d_k, \quad k = 1, 2, \ldots, \hspace{1cm} (1.2)$$

where \( x_k \) is the current iteration point and \( \alpha_k > 0 \) is the stepsize. There are two methods in order to solve the stepsize which are exact and inexact line searches. At the present time, most researchers tend to use the inexact line search since it converges faster. As stated in [16], inexact line search are powerful in practical computational. The condition of inexact line search, strong Wolfe is as follow:

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \mu \alpha_k g^T_k d_k, \hspace{1cm} (1.3)$$

$$g^T_k d_k \leq -\sigma g^T_k d_k, \hspace{1cm} (1.4)$$

where \( 0 < \mu < \sigma < 1 \).
The search direction of conjugate gradient (CG) method, \( d_k \), is described as,
\[
d_k = \begin{cases} 
-g_k & \text{if } k = 0 \\
-g_k + \beta_k d_{k-1} & \text{if } k \geq 1,
\end{cases}
\] (1.5)
for which we have \( g_k = \nabla f(x_k) \) at the point \( x_k \) while \( \beta_k \in R \) is a scalar known as the CG coefficient.

The father of CG, Hestenes and Stiefel (HS) has proposed the classical CG in 1952. Another CG method has been established by Fletcher and Reeves (FR) in 1964. Few years later, a new CG method has been presented by Polak and Ribiere (PR) in 1969. In 1987, Fletcher again introduced Conjugate Descent (CD). In recent times, there are many improvements on the classical conjugate gradient methods are made by researchers. Besides, the spectral CG and hybrid CG also have been studied more extensively. These methods below have been used in this study for performance comparison,
\[
\beta^{\text{HN}}_k = \frac{g_k^T (g_k - g_{k-1})}{\|d_k^T - g_k\|^2},
\] (1.8)
where Hestenes and Stiefel (HS), Conjugate Descent (CD), Wei, Yao and Liu (WYL) and Rivaie, Mustafa, Ismail and Leong (RMIL). The methods are chosen from both previous and recent researches in order to certify the study. All of the methods can be read through [1, 4–8, 10–15, 17–19].

In Section 2, a new CG coefficient will be introduced with its algorithm. The proof is established based on the strong Wolfe technique using inexact line search are presented in Section 3. Section 4 consists of the set of chosen test functions, numerical results and discussion. Lastly, the paper is concluded in Section 5.

2. New conjugate gradient method
In this section, a new method has been proposed and is motivated from (1.8). This new method is known as \( \beta^{\text{LAMR}}_k \), where LAMR denotes Linda, Aini, Mustafa and Rivaie as below
\[
\beta^{\text{LAMR}}_k = \frac{\frac{1}{\|d_k\|^2} g_k^T \left( \frac{\|d_{k-1}\|^2}{\|d_{k-1} - g_k\|^2} d_{k-1} - g_k - g_{k-1} \right)}{\frac{1}{\|d_{k-1} - g_k\|^2} \|d_{k-1} - g_k\|^2},
\] (2.1)
The algorithm of new CG method used in this paper is given as follow:

**Step 1:** Initialization. Given \( x_k \), set \( k = 0 \).

**Step 2:** Compute \( \beta_k \) based on (1.6 - 1.9 & 2.1).

**Step 3:** Compute \( d_k \) based on (1.5). If \( \|g_k\| = 0 \), then stop.

**Step 4:** Compute \( \alpha_k \) by using inexact line search.
Step 5: Update a new point based on the iterative formula (1.2).

Step 6: Convergence test and stopping criteria.

If \( f(x_{k+1}) < f(x_k) \) and \( \|g_{k+1}\| \leq \epsilon \), then stop.

Otherwise go to Step 1 with \( k = k + 1 \).

3. Convergence analysis

A good algorithm should fulfill the sufficient descent and global convergence properties. Thus, the sufficient descent properties of this new method will be further studied in this section. From (2.1), we know that

\[
\beta_{i+1}^{\text{LMR}} = \frac{g_i^T (d_{i+1} - g_{i+1})}{d_i^T (d_i - g_i)} = \frac{\|d_i\|}{\|d_i - g_i\|} \frac{\|g_{i+1}\|^2 - g_i^T g_{i+1}}{\|d_i - g_i\|^2}, \quad \beta_{i+1}^{\text{LMR}} \leq \frac{\|d_i\|}{\|d_i - g_i\|} \frac{\|g_{i+1}\|^2}{\|d_i - g_i\|^2}.
\]

Hence, we obtain

\[
\beta_{i+1}^{\text{LMR}} \leq \frac{\|g_{i+1}\|}{\|d_i\|}.
\] (3.1)

The subsequent discussion indicates the proof of sufficient descent condition for LAMR method by strong Wolfe line search.

Assumption 1. Assume that \( g_k \neq 0 \) for all \( k \) or else, the stationary point has been found. The sufficient descent property is given as \( g_i^T d_k < 0 \).

Theorem 1. Consider LAMR method with the search direction (1.5) is determined by (1.3) and (1.4), with \( \sigma < \frac{1}{5} \), then for all \( k \geq 0 \), it becomes

\[
\frac{\|g_k\|}{\|d_k\|} < \frac{3}{2}.
\] (3.2)

Proof. Theorem 2 is completed by induction. The condition (3.1) holds for \( k = 0 \) as \( \frac{\|g_0\|}{\|d_0\|} = 1 < \frac{3}{2} \).

Besides, the condition (3.2) is also true for some \( k \geq 0 \).

Multiplying \( g_i^T \) to (1.5) gives

\[
g_{i+1}^T d_{i+1} = -\|g_i\|^2 + \beta_{i+1} g_{i+1}^T d_{i+1}.
\] (3.3)

By implementing the strong Wolfe condition as in (1.3) and (1.4),

\[
\|g_{i+1}\| \leq |g_{i+1}^T d_{i+1}| + \sigma |g_{i+1}^T g_{i+1}|.
\]

By using Cauchy inequalities, substitute (3.1) into (3.4),

\[
\|g_{i+1}\| \leq \frac{\|g_{i+1}\|}{\|d_{i+1}\|} \|d_{i+1}\| + \sigma \frac{\|g_{i+1}\|}{\|d_{i+1}\|} \|g_{i+1}\|.
\] (3.4)

That indicates

\[
\|g_{i+1}\| \leq \frac{\|g_{i+1}\|}{\|d_{i+1}\|} \|d_{i+1}\| + \sigma \frac{\|g_{i+1}\|}{\|d_{i+1}\|} \|g_{i+1}\|.
\]
Applying the induction hypothesis in (3.2),
\[ \left\| g_{k,i} \right\| \leq \left\| g_{k,i} \right\| \left\| d_{k,i} \right\| + \frac{3}{2} \sigma \left\| g_{k,i} \right\| , \]
\[ \left\| g_{k,i} \right\| \left( 1 - \frac{3}{2} \sigma \right) \leq \left\| g_{k,i} \right\| \left\| d_{k,i} \right\| , \]
Thus, \[ \frac{\left\| g_{k,i} \right\|}{\left\| d_{k,i} \right\|} \leq \frac{2}{2 - 3 \sigma} . \]

Therefore, if \( \sigma < \frac{1}{3} \), then \[ \frac{\left\| g_{k,i} \right\|}{\left\| d_{k,i} \right\|} \leq \frac{3}{2} . \] Hence (3.2) is true for \( k + 1 \).

Theorem 2. Suppose that Assumption 1 holds and that \( g_{i}^{*}d_{i} < 0 \) and \( d_{k,i} \) is generated by formula (1.5), \( \beta_{i} \) as (2.1) with \( \sigma < \frac{1}{3} \) and \( \alpha_{i} \) by using strong Wolfe line search, then for all \( k \geq 0 \), the relation
\[ \frac{-4}{4 - 9 \sigma} \frac{g_{i}^{*}d_{i}}{\left\| g_{i} \right\|} \leq \sigma \frac{g_{i}^{*}d_{i}}{\left\| g_{i} \right\|} \leq \frac{18 \sigma - 4}{4 - 9 \sigma} \] (3.5)

still holds. Therefore, the property of sufficient descent, \( g_{i}^{*}d_{i} \leq -c \left\| g_{i} \right\| \) for \( \forall k \geq 0 \) and \( c > 0 \) holds as long as \( g_{i} \neq 0 \).

Proof. By induction, the result holds for \( k = 0 \), where (3.5) is true for some \( k \geq 0 \). From (3.3),
\[ g_{k,i}^{*}d_{k,i} = -\left\| g_{k,i} \right\| + \beta_{k,i}^{LAMR} g_{k,i}^{*}d_{k} \]
Dividing (3.5) by \( \left\| g_{k,i} \right\| \),
\[ \frac{g_{k,i}^{*}d_{k,i}}{\left\| g_{k,i} \right\|} = -1 + \beta_{k,i}^{LAMR} \frac{g_{k,i}^{*}d_{k,i}}{\left\| g_{k,i} \right\|} \]
From strong Wolfe, we have
\[ \beta_{k,i}^{LAMR} g_{k,i}^{*}d_{k,i} \leq -\sigma \beta_{k,i}^{LAMR} g_{k,i}^{*}d_{k} \]
(3.6)

By considering the absolute value properties, the inequality is attained.
\[ -1 + \sigma \beta_{k,i}^{LAMR} \frac{g_{k,i}^{*}d_{k,i}}{\left\| g_{k,i} \right\|} \leq -1 - \sigma \beta_{k,i}^{LAMR} \frac{g_{k,i}^{*}d_{k,i}}{\left\| g_{k,i} \right\|} \]
As (3.5) is true for \( k \geq 0 \) and \( \beta_{k,i}^{LAMR} \leq \frac{\left\| g_{k,i} \right\|}{\left\| d_{k,i} \right\|} \), then
\[ -1 + \sigma \frac{\left\| g_{k,i} \right\|}{\left\| d_{k,i} \right\|} \frac{g_{k,i}^{*}d_{k,i}}{\left\| g_{k,i} \right\|} \leq -1 - \sigma \frac{\left\| g_{k,i} \right\|}{\left\| d_{k,i} \right\|} \frac{g_{k,i}^{*}d_{k,i}}{\left\| g_{k,i} \right\|} \]
\[ -1 + \sigma \frac{\left\| g_{k,i} \right\|}{\left\| d_{k,i} \right\|} \frac{g_{k,i}^{*}d_{k,i}}{\left\| g_{k,i} \right\|} \leq -1 - \sigma \frac{\left\| g_{k,i} \right\|}{\left\| d_{k,i} \right\|} \frac{g_{k,i}^{*}d_{k,i}}{\left\| g_{k,i} \right\|} \]
Based on (3.2) and (3.5),
\[ -1 + \sigma \frac{\left\| g_{k,i} \right\|}{\left\| d_{k,i} \right\|} \frac{-4}{4 - 9 \sigma} \frac{g_{k,i}^{*}d_{k,i}}{\left\| g_{k,i} \right\|} \leq -1 - \sigma \frac{\left\| g_{k,i} \right\|}{\left\| d_{k,i} \right\|} \frac{-4}{4 - 9 \sigma} \]
\[ -1 + \frac{9 \sigma}{4 - 9 \sigma} \leq \frac{g_{k,i}^{*}d_{k,i}}{\left\| g_{k,i} \right\|} \leq -1 + \frac{9 \sigma}{4 - 9 \sigma} . \]
Therefore, the proof is completed. \( \blacksquare \)
4. Numerical results and discussion

As in the Table 1 below, the test problems used to test the effectiveness and robustness of this new method are listed for the numerical experiments. These test problems are chosen by referring to [2] and considered for both small and large scale problems. Different variables which is ranging from 2 to 10000 have been tested to each test problem. In this experiment, four different initial points have been selected, starting from a point which is closer to the solution point to the point far away from the solution point. By using the random number generator, the initial points are selected randomly. For additional reading, refer to [9]. A comparison with the other CG methods which are CD, HS, RMIL and WYL (1.6 – 1.9) are made by inexact line search, strong Wolfe. All of these algorithms are considered $\varepsilon = 10^{-6}$. As the stopping criteria, $\|\varepsilon\| < 10^{-6}$ has been fulfilled, the calculation is terminated. All the problems listed below are calculated by MatlabR2012 subroutine programming. The iteration number and CPU time are evaluated as the best method requires fewer iteration numbers and less CPU time.

Table 1. A list of test problems.

| No. | Test Problems     | Variable/s | Initial Points                      |
|-----|-------------------|------------|-------------------------------------|
| 1   | Booth             | 2          | (3,3),(6,6),(9,9),(12,12)           |
| 2   | Six Hump          | 2          | (1,1),(3,3),(5,5),(9,9)             |
| 3   | Trecanni          | 2          | (5,5),(18,18),(26,26),(36,36)       |
| 4   | Dixon & Price     | 2,4        | (6,6,6,6),(10,10,10),(14,14,14),(26,26,26) |
| 5   | Power             | 2,4        | (5,5,5,5),(10,10,10,10),(15,15,15,15),(20,20,20,20) |
| 6   | Generalized Quartic | 2,4,10  | (4,...,4),(8,...,8),(12,...,12),(16,...,16) |
| 7   | Extended Penalty  | 2,4,10,100 | (3,...,3),(10,...,10),(23,...,23),(30,...,30) |
| 8   | Hager             | 2,4,10,100 | (1,...,1),(2,...,2),(3,...,3),(4,...,4) |
| 9   | Raydan            | 2,4,10,100 | (1,...,1),(2,...,2),(3,...,3),(4,...,4) |
| 10  | ARWHEAD           | 2,4,10,100,500,1000 | (2,...,2),(5,...,5),(11,...,11),(14,...,14) |
| 11  | Diagonal 4        | 2,4,10,100,500,1000,5000,10000 | (10,...,10),(20,...,20),(30,...,30),(40,...,40) |
| 12  | Extended DENSCHNB | 2,4,10,100,500,1000,5000,10000 | (3,...,3),(8,...,8),(15,...,15),(19,...,19) |
| 13  | Extended Himmelblau | 2,4,10,100,500,1000,5000,10000 | (2,...,2),(6,...,6),(14,...,14),(25,...,25) |
| 14  | NONDIA            | 2,4,10,100,500,1000,5000,10000 | (4,...,4),(10,...,10),(13,...,13),(31,...,31) |
| 15  | Sphere            | 2,4,10,100,500,1000,5000,10000 | (8,...,8),(16,...,16),(24,...,24),(32,...,32) |

By using the performance profile introduced by [3], the performance results as shown in Fig. 1 and Fig. 2 based on the iteration numbers and CPU time are described respectively. The performance of the method can be analyzed by referring to these graphs. The effectiveness of a method can be verified through its convergence rate. The robustness of a method is validated by the number of test problems it solved. The most robust method can be determined from the curve that is located to top right of the figure while the method with fastest convergence can be deduced from the top curve at the left side of the figure.

In both figures, the position of the curves for all tested CG methods are similar, meaning that the numerical results in terms of CPU time corresponds to those in iteration number and vice versa. The LAMR method is clearly better compared to RMIL and CD. It is also comparable to HS in term of efficiency but LAMR and WYL are more robust compared to HS. Based on these results, LAMR is suitable for further application and studies in solving optimization functions.
5. Conclusion

In this paper, LAMR has been introduced and it is proven that the method fulfils the sufficient descent condition under inexact line search. Based on the numerical results, the LAMR method is only the second fastest method in terms of iteration number and CPU time, but it ties with WYL as the most robust ones. Also, by numerical performance, it is shown to possess sufficient descent and global convergence properties. Overall, the results of LAMR are encouraging for further application and testing in optimization.

Acknowledgments

The authors would like to thank the editors and the referees for their suggestions and comments. The authors are also thankful to The Ministry of Higher Education of Malaysia (MOHE) for the sponsorship of this research via MyPhd Scheme.

References

[1] ‘Aini N, Rivaie M and Mamat M 2016 Aip Conf. Proc. 1787 080019
[2] Andrei N 2011 Bull. Malays. Math. Sci. Soc. 34 319-330.
[3] Dolan E and More J J 2002 Math. Programming 91 201-213
[4] Fletcher R 1987 John Wiley & Sons 1 1987.
[5] Fletcher R and Reeves C 1994 Comput. Journal 7 149-154.
[6] Ghani N H A, Rivaie M and Mamat M 2016 Aip Conf. Proc. 1739 020076.
[7] Hajar N, Rivaie M, and Mamat M, 2016 Aip Conf. Proc. 1739 020089.
[8] Hestenes M R and Steifel E 1952 J. Res. Nat. Bur. Stand. 49 409-436.
[9] Hilstrom K E 1977 A.C.M. Trans. Maths. Softw. 3 4 305-315.
[10] Khadijah W, Rivaie M, Mamad M, and Jusoh I 2016 Aip Conf. Proc. 1739 020072.
[11] Mohamed N S, Mamad M, Mohammad F S and Rivaie M 2016 Jurnal Teknologi Science & Engineering 78 131-136
[12] Polak E and Ribiere G 1969 Rev. Franaise Informat Recherche Operationelle 3 35-43.
[13] Rivaie M, Mamad M and Abashar A 2015 Appl. Math. and Comput. 268 1152–1163
[14] Rivaie M, Mamad M, Ismail M and Leong W J 2012 Appl. Math. and Comput 218 11323-11332.
[15] Shapiee N, Rivaie M, and Mamat M 2016 Aip Conf. Proc. 1739 020082.
[16] Shi Z J and Guo J 2009 J. Comput. Appl. Math. 224 444-457.
[17] Shoid S, Rivaie M and Mamad M 2016 Aip Conf. Proc. 1739 020071
[18] Wei Z, Yao S, and Liu L 2006 Appl. Math. and Comput 183 1341-1350
[19] Zull N, Rivaie M, Mamad M, Salleh Z and Amani Z 2015 Applied Math. Science 9 3105-3117.