An Extension of Polak-Ribière-Polyak Method using Exact Line Search

Mahmoud Dawahdeh, Mustafa Mamat, Mohd Rivaie, Mohammad Afendee Mohamed, Puspa Liza Ghazali

ABSTRACT—Lately, many large-scale unconstrained optimization problems rely upon nonlinear conjugate gradient (CG) methods. Many areas such as engineering, computer science have benefited because of its simplicity, fast and low memory requirements. Many modified coefficients have appeared recently, all of which to improve these methods. This paper considers an extension conjugate gradient method of Polak-Ribière-Polyak using exact line search to show that it holds for some properties such as sufficient descent and global convergence. A set of 113 test problems is used to evaluate the performance of the proposed method and get compared to other existing methods using the same line search.

I. INTRODUCTION

The problem of unconstrained optimization is represented as a function for which the nonlinear CG methods are used to minimize. Generally, the form of the optimization problem is:

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

such that $$f : \mathbb{R}^n \rightarrow \mathbb{R}$$ is continuously differentiable. The gradient of the function $$f(x)$$ is available and denoted by $$g(x) = \nabla f(x)$$. Now, the iterative formula of CG method and the search direction $$d_k$$ are respectively defined by:

$$x_{k+1} = x_k + \alpha_k d_k$$, $$k = 0, 1, 2, 3, \ldots$$ \hspace{1cm} (2)

$$d_k = \begin{cases} -g_k & , k = 0 \\ -g_k + \beta_k d_{k-1} & , k \geq 1 \end{cases} \hspace{1cm} (3)$$

where $$\alpha_k > 0$$ is the step length which can be obtained through some methods (line search, etc.) and $$\beta_k$$ a CG coefficient that different CG methods are defined based on it. In addition, $$g_k$$ a gradient at point $$x_k$$ of $$f(x)$$. There exist two types of line searches namely exact and inexact line search. We use exact line search in this study as a tool to compute the step length, and it can be

$$f(x_k + \alpha_k d_k) = \min f(x_k + \alpha d_k)$$, $$\alpha \geq 0$$ \hspace{1cm} (4)

This method is not preferred to be studied by many researchers, because they believe that it is quite slow. They prefer to use the other line search which is inexact. But when fast computer processors are used with our method, it gives good numerical results and so, this will be an advantage for exact line search method. The most common $$\beta_k$$ formulas are:

- Hestenes-Stiefel (HS) \[1\], Polak-Ribière-Polyak (PRP) \[2\], Gilbert and Nocedal (PRP+) \[3\], Fletcher-Reeves (FR) \[4\], Liu-Storey (LS) \[5\], Dai-Yuan (DY) \[6\], Wei et al. (WYL) \[7\], Conjugate Descent (CD) \[8\] which are given by:

$$\beta_k^{HS} = \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}$$, \hspace{1cm} (5)

$$\beta_k^{PRP} = \frac{g_k^T y_{k-1}}{g_{k-1}^T g_{k-1}}$$, \hspace{1cm} (6)

$$\beta_k^{PRP+} = \max \{ \beta_k^{PRP}, 0 \}$$, \hspace{1cm} (7)

$$\beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}$$, \hspace{1cm} (8)

$$\beta_k^{LS} = \frac{g_k^T y_{k-1}}{g_{k-1}^T g_{k-1}}$$, \hspace{1cm} (9)

$$\beta_k^{DY} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}$$, \hspace{1cm} (10)

$$\beta_k^{WYL} = \frac{\|g_k\|^2 - \|g_{k-1}\|^2}{\|g_{k-1}\|^2}$$, \hspace{1cm} (11)

$$\beta_k^{CD} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}$$, \hspace{1cm} (12)

In 1952, Hestenes-Stiefel \[1\] proposed the earliest formula for solving the quadratic functions. Accordingly, a formula by Fletcher and Reeves \[4\] came later in 1964 for nonlinear functions. Based on FR method, some investigations on convergence properties were initiated by Zoutendijk \[9\], subsequently followed by Al-Baali \[10\] with the proof of global convergence using SWP line search having $$\sigma < 1/2$$. After that, Guanghui et al. \[11\] made an extension to the result with $$\sigma \leq 1/2$$. With an exact line search, Elijah and Ribiere \[2\] showed that PRP method finally globally converged. However, Powell \[12\] disproved the global convergence property of PRP method by showing that there exists a nonconvex function. As a result, Powell introduced nonnegative PRP method. Later, Gilbert and Nocedal \[3\] proved the global convergence property of PRP+ for $$\beta_k^{PRP+} = \max \{ \beta_k^{PRP}, 0 \}$$ using some line search. Unfortunately, using SWP line search with general nonlinear functions, PRP+ failed to converge. Moreover, Wei et al. \[7\] discovered a new positive CG method that behaves very similar to the old PRP method. This is followed by several modifications which appeared in \[13\], \[14\] much later.

Revised Manuscript Received on July 10, 2019.

Mahmoud Dawahdeh, Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Terengganu, Malaysia.

Mustafa Mamat, Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Terengganu, Malaysia.

Mohd Rivaie, Department of Computer Science and Mathematics, Universiti Teknologi Mara (UiTM) Terengganu, Kuala Terengganu Campus, Malaysia.

Mohammad Afendee Mohamed, Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Terengganu, Malaysia.

Puspa Liza Ghazali, Faculty of Economics and Management Sciences, Universiti Sultan Zainal Abidin, Terengganu, Malaysia.
AN EXTENSION OF POLAK-RIBIÈRE-POLYAK METHOD USING EXACT LINE SEARCH

\[
\beta_{VHS} = \frac{\|g_k\|^2 - \|g_{k-1}\|^2}{\|g_{k-1}\|^2} \tag{13}
\]

\[
\beta_{DRP} = \frac{\|g_k\|^2 - \|g_{k-1}\|^2}{\|g_{k-1}\|^2} \tag{14}
\]

where \( w \geq 1 \).

Zhang [15] introduced another method, a variant of \( \beta_{VHS} \).

\[
\beta_{NPRP} = \frac{\|g_k\|^2 - \|g_{k-1}\|^2}{\|g_{k-1}\|^2} \tag{15}
\]

Generally, CG methods are aiming at robustness and efficiency [16]–[19]. This paper is structured as follows. Our proposed CG method is presented in section 2. Then, the convergence analysis presented in section 3. Numerical results are shown in Section 4. The Last Section is 5, where the conclusion is.

II. NEWLY PROPOSED FORMULA AND ALGORITHM

In this section, our new \( \beta_k \) is presented which is known as \( \beta_{DMAR} \) and it is extended to \( \beta_{VWL} \) and \( \beta_{NPRP} \) method. It is known as (Dawahdeh, Mamat, and Rivaie) DMAR method. The idea of the newly proposed formula mainly comes from Zhang [15] and is given in (16).

\[
\beta_{DMAR} = \begin{cases} 
\|g_k\|^2 - \|g_{k-1}\|^2, & \|g_k\|^2 \geq \mu_k \|g_{k-1}\|^2 \\
0, & \text{otherwise}
\end{cases} \tag{16}
\]

where \( \| \cdot \| \) is the Euclidean norm. We define \( \mu_k \) as

\[
\mu_k = \frac{\|g_k\|^2}{\|g_{k-1}\|^2} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2} \cdot
\]

Noteworthy

\[
0 \leq \frac{\|g_k\|^2 - \|g_{k-1}\|^2}{\|g_{k-1}\|^2} \leq \frac{\|g_k\|^2}{\|g_{k-1}\|^2} = \beta_{FR}^k
\]

and so,

\[
0 \leq \beta_{DMAR} \leq \beta_{FR}^k \tag{17}
\]

Hence, according to the argument presented in [3], \( \beta_{DMAR} \) will inherit all properties belong to \( \beta_{FR}^k \).

Algorithm 1

Step 1: Choose an initial point \( x_0 \), and set initial search direction \( d_0 = -g_0 \) and \( \alpha_0 = 0 \).

Step 2: Compute \( d_k \) from (3) using (19).

Step 3: Compute \( \alpha_k \) using (4).

Step 4: Update \( x_{k+1} \) following to (2).

Step 5: If \( \|g_k\| \leq 10^{-6} \) then stop; otherwise let \( k = k + 1 \) and go to Step 2.

III. CONVERGENCE ANALYSIS

We begin the study of convergence properties of \( \beta_{DMAR} \) with sufficient descent condition which will be needed later to prove the global convergence property. In CG methods, one important rule is that of descent condition. By proving that

\[
g_k^T d_k \leq 0
\]

then, the expression \( f(x_{k+1}) < f(x_k) \) holds. We could extend to the definition of sufficient descent condition by the following.

\[
g_k^T d_k \leq -C\|g_k\|^2, \text{ when } k \geq 0, C > 0. \tag{18}
\]

Prior proceeding to the proof of sufficient descent direction condition for DMAR, noteworthy \( \beta_{DMAR} \) satisfies

\[
0 \leq \beta_{DMAR} \leq \frac{\|g_k\|^2}{\|g_{k-1}\|^2} \tag{19}
\]

Theorem 1. If \( g_k \neq 0 \), suppose the direction \( d_k \) is generated by Algorithm 1, then we have the condition (18) holds for any \( k \geq 0 \).

Proof. Clearly, the result is true for \( k = 0, d_0 = -g_0 \). Now for \( k \geq 1 \), from (3), and (19) we have

\[
d_k = -g_k + \beta_{DMAR} d_{k-1} \tag{20}
\]

Multiplying both side by \( g_k^T \), then we get

\[
g_k^T d_k = -\|g_k\|^2 + \beta_{DMAR} g_k^T d_{k-1}.
\]

Specific to exact line search, we have \( g_k^T d_{k-1} = 0 \). Hence,

\[
g_k^T d_k = -\|g_k\|^2.
\]

And so, (18) holds true which implies that \( d_k \) is a sufficient descent direction. This completes the proof.

Now, we will show that our CG method with \( \beta_{DMAR} \) converges globally. The following basic assumption is frequently required for completeness.

Assumption 1.

(i) The level set \( L = \{x \in \mathbb{R}^n: f(x) \leq f(x_0)\} \) is bounded.

(ii) For some neighborhood \( N \) of \( x_0 \), \( f(x) \) is continuously differentiable function, having Lipschitz continuous gradient, and there exists a constant \( K > 0 \) such that

\[
\|g(x) - g(y)\| \leq K\|x - y\|, \forall x, y \in N.
\]

Lemma 1. Let Assumption 1 hold. \( x_k \) is generated by Algorithm 1, then we have

\[
\sum_{k=1}^{\infty} \frac{\|g_k\|^4}{\|g_k\|^2} < \infty. \tag{21}
\]

Go back to [9] for the proof of Lemma 1. The following theorem proves the global convergence property of DMAR, depending on the above Assumption and Lemma.

Theorem 2. Consider the sequence \( x_k \) is generated by Algorithm 1, and suppose that Assumption 1 holds. Then, we obtain

\[
\lim_{k \to \infty} \inf \|g_k\| = 0. \tag{22}
\]
Proof. Suppose the contradiction, there exists a positive constant \( \epsilon > 0 \) such that
\[
\|g_k\| \geq \epsilon, \text{ for all } k \geq 0.
\] (23)
which means
\[
\frac{1}{\|g_k\|^2} \leq \frac{1}{\epsilon^2}, \text{ for all } k \geq 0 \text{ and } \|g_k\| \neq 0.
\] (24)
Rewriting (20) as \( d_k + g_k = \beta_k^{DMAR} d_{k-1} \), and squaring both sides we obtain
\[
\|d_k\|^2 + \|g_k\|^2 + 2g_k^T d_k = (\beta_k^{DMAR})^2 \|d_{k-1}\|^2,
\]
then
\[
\|d_k\|^2 = -\|g_k\|^2 - 2g_k^T d_k + (\beta_k^{DMAR})^2 \|d_{k-1}\|^2.
\] (25)
Dividing both sides of (25) by \((g_k^T d_k)^2\), we have
\[
\frac{\|d_k\|^2}{\|g_k\|^4} = \frac{\|d_k\|^2}{(g_k^T d_k)^2} = (\beta_k^{DMAR})^2 \frac{\|d_{k-1}\|^2}{(g_k^T d_k)^2} - \frac{2g_k^T d_k}{(g_k^T d_k)^2} - \frac{\|g_k\|^2}{(g_k^T d_k)^2}.
\]
By applying the condition that, \( g_k^T d_k = -\|g_k\|^2 \), we find that:
\[
\frac{\|d_k\|^2}{\|g_k\|^4} = \frac{\|d_k\|^2}{(g_k^T d_k)^2} = (\beta_k^{DMAR})^2 \frac{\|d_{k-1}\|^2}{\|g_k\|^4} - \frac{2(-\|g_k\|^2)}{\|g_k\|^4} - \frac{\|g_k\|^2}{\|g_k\|^4}
\]
\[
= (\beta_k^{DMAR})^2 \frac{\|d_{k-1}\|^2}{\|g_k\|^4} + \frac{1}{\|g_k\|^4}
\]
\[
= (\beta_k^{DMAR})^2 \frac{\|d_{k-1}\|^2}{\|g_k\|^4} + \frac{1}{\|g_k\|^4}
\]
By applying (19) yields:
\[
\frac{\|d_k\|^2}{\|g_k\|^4} = \left(\frac{\|g_k\|^2}{\|g_{k-1}\|^2}\right)^2 \frac{\|d_{k-1}\|^2}{\|g_{k-1}\|^4} + 1
\]
\[
\frac{\|d_k\|^2}{\|g_k\|^4} = \left(\frac{\|g_k\|^2}{\|g_{k-1}\|^2}\right)^2 \frac{\|d_{k-1}\|^2}{\|g_{k-1}\|^4} + 1
\]
\[
\leq \left(\frac{\|d_{k-1}\|^2}{\|g_{k-1}\|^4} + \frac{1}{\|g_{k-1}\|^2}\right)
\]
Note that \((\|d_0\|^2 = \|g_0\|^2)\), so we get:
\[
\frac{\|d_k\|^2}{(g_k^T d_k)^2} \leq \frac{1}{\|g_0\|^2} + \frac{1}{\|g_k\|^2}
\]
Then, we have
\[
\frac{\|d_k\|^2}{(g_k^T d_k)^2} \leq \sum_{i=0}^{k-1} \frac{1}{\|g_i\|^2} \leq \frac{k}{\epsilon^2}
\]
And so, we get
\[
\sum_{k=1}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} \geq \epsilon^2 \sum_{k=1}^{\infty} \frac{1}{k} = +\infty,
\]
which is contradictory to Lemma 1, and hence complete the proof.

IV. NUMERICAL RESULTS

To analyze the performance of our method, we use some standard test functions as given in Table 1, which is adopted from Adorio and Diliman [20], Andrei [21], and CUTEr [22]. We conducted comparison studies between our method DMAR against other existing CG methods such as WYL [7] and NPRP [15] methods which are from the most well-known conventional and recently CG methods. Measurements are based on CPU time and the number of iterations.

The tolerance \( \epsilon \) is selected to be \( 10^{-6} \), it is used to study the efficiency of these methods towards the optimal. Moreover, we chose \( \|g_k\| \leq 10^{-6} \) as the stopping criteria. In case the iteration counts grows bigger than 1000 times, the method is considered as failing.

All methods were simulated on MATLAB R2015a subroutine running on a PC with Intel R Core TM, i5-2410 having 3 GB RAM. Fig. 1 and 2 respectively show the measurement taken for both parameters. The performance measurements were based on that of Dolan and Mor’ [23]. Observably, the best method composes of high values of \( p(t) \) and is normally located on the upper most right section with within both figures.
Table 1: Test cases

<table>
<thead>
<tr>
<th>N Function</th>
<th>Dimension</th>
<th>Initial Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 Extended Beale</td>
<td>2</td>
<td>(1.8, 1.8)</td>
</tr>
<tr>
<td>2 Perturbed Quadratic</td>
<td>2</td>
<td>(0.5, 0.5)</td>
</tr>
<tr>
<td>3 Raydan 1</td>
<td>2</td>
<td>(1, 1, (10, 10)</td>
</tr>
<tr>
<td>4 Diagonal 4</td>
<td>2</td>
<td>(1, 1, 10, 10)</td>
</tr>
<tr>
<td>5 Extended Himmelblau</td>
<td>2, 500, 1000, 5000, 10000</td>
<td>(1, 1, ..., 1) (10, 10)</td>
</tr>
<tr>
<td>6 Extended Powell</td>
<td>2, 500, 1000, 5000, 10000</td>
<td>(1, 1, ..., 1) (10, 10)</td>
</tr>
<tr>
<td>7 FLETCHCR</td>
<td>2</td>
<td>(0.5, 0.5)</td>
</tr>
<tr>
<td>8 NONSCOMP</td>
<td>2</td>
<td>(3, 3) (10, 10)</td>
</tr>
<tr>
<td>9 Extended DENSCHNB</td>
<td>2</td>
<td>(1, 1, ... 1) (10, 10)</td>
</tr>
<tr>
<td>10 Extended Quadratic Penalty QP1</td>
<td>2</td>
<td>(1, 1)</td>
</tr>
<tr>
<td>11 Extended Penalty</td>
<td>2</td>
<td>(1, 1, 2, 2)</td>
</tr>
<tr>
<td>12 Hager</td>
<td>2</td>
<td>(1, 1, 10, 10)</td>
</tr>
<tr>
<td>13 BIGGSB1</td>
<td>2</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>14 Six Hump</td>
<td>2</td>
<td>(10, 10)</td>
</tr>
<tr>
<td>15 Three Hump</td>
<td>2</td>
<td>(1, 1, -1, -1)</td>
</tr>
<tr>
<td>16 Booth</td>
<td>2</td>
<td>(1, 1, 3, 3)</td>
</tr>
<tr>
<td>17 Zettl</td>
<td>2</td>
<td>(0, 0, 1, 1)</td>
</tr>
<tr>
<td>18 Shallow</td>
<td>2</td>
<td>(-2, -2)</td>
</tr>
<tr>
<td>19 Generalized Quartic</td>
<td>2, 500, 1000, 5000, 10000</td>
<td>(1, 1, ..., 1) (10, 10)</td>
</tr>
<tr>
<td>20 Quadratic QF2</td>
<td>2</td>
<td>(0.5, 0.5), (1, 1)</td>
</tr>
<tr>
<td>21 Price</td>
<td>2</td>
<td>(-10, -10), (1, 1)</td>
</tr>
<tr>
<td>22 Generalized Tridiagonal 1</td>
<td>500</td>
<td>(2, 2, ..., 2)</td>
</tr>
<tr>
<td>23 Generalized Tridiagonal 2</td>
<td>2</td>
<td>(1, 1)</td>
</tr>
<tr>
<td>24 Quadratic QF1</td>
<td>2</td>
<td>(1, 1)</td>
</tr>
<tr>
<td>25 Extended Quadratic Penalty QP2</td>
<td>500</td>
<td>(1, 1, ..., 1)</td>
</tr>
<tr>
<td>26 Extended White and Holst</td>
<td>2</td>
<td>(3, 3), (9, 9)</td>
</tr>
<tr>
<td>27 ARWHEAD</td>
<td>2</td>
<td>(1, 1), (10, 10)</td>
</tr>
<tr>
<td>28 QUARTC</td>
<td>2</td>
<td>(2, 2, ..., 2)</td>
</tr>
<tr>
<td>29 Diagonal 4</td>
<td>2, 500, 1000, 5000, 10000</td>
<td>(1, 1, ..., 1)</td>
</tr>
<tr>
<td>30 Extended Block Diagonal BD1</td>
<td>2, 500, 1000</td>
<td>(0.1, 0.1, ..., 0.1)</td>
</tr>
<tr>
<td>31 Extended Three Exponential Terms</td>
<td>2</td>
<td>(0.1, 0.1, (1, 1)</td>
</tr>
<tr>
<td>32 Extended Cliff</td>
<td>2</td>
<td>(0.0, ...)</td>
</tr>
<tr>
<td>33 DE jongs</td>
<td>2</td>
<td>(10, 10, ..., 10)</td>
</tr>
<tr>
<td>34 BIGGSB1</td>
<td>2, 500, 1000, 5000, 10000</td>
<td>(0, 0)</td>
</tr>
<tr>
<td>35 POWER</td>
<td>2</td>
<td>(1, 1)</td>
</tr>
<tr>
<td>36 Dixon and Price</td>
<td>2</td>
<td>(1, 1)</td>
</tr>
<tr>
<td>37 Sphere</td>
<td>2</td>
<td>(1, 1, ..., 1)</td>
</tr>
<tr>
<td>38 Sum Squares</td>
<td>2, 500, 1000, 5000, 10000</td>
<td>(1, 1)</td>
</tr>
<tr>
<td>39 TRIDIA</td>
<td>2</td>
<td>(1, 1)</td>
</tr>
<tr>
<td>40 DIXON3DQ</td>
<td>2</td>
<td>(-1, -1)</td>
</tr>
<tr>
<td>41 Generalized Quartic GQ1</td>
<td>2</td>
<td>(1, 1), (10, 10)</td>
</tr>
<tr>
<td>42 Generalized Quartic GQ2</td>
<td>2</td>
<td>(0.2, 0.2), (1, 1)</td>
</tr>
<tr>
<td>43 Extended Trigonometric</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

Each method is measure by the percentage of success in solving the test problem, represented by, \( p_s(t) \) on vertical axis, whereas, the horizontal axis determines the fastest methods.
This paper considers the CG methods that have been famously applied in many areas such as computer science, and engineering [24–29]. By proposing a new formula for CG method, the numerical simulation shows that the proposed coefficient exhibited superb performance in comparison to WYL and NPRP. Also, the global convergence is proved for DMAR method with the exact line search besides, the sufficient descent property.

ACKNOWLEDGMENT

This research is funded by the Malaysian Ministry of Education via Fundamental Research Grant Scheme (FRGS/1/2017/STG06/Unisza/01/1).

REFERENCES