New Iterative Conjugate Gradient Method for Nonlinear Unconstrained Optimization Using Homotopy Technique

Salah Gazi Shareef and Hussein Ageel Khatab
Dept. of Mathematics, Faculty of Science, University of Zakho, Kurdistan Region-Iraq

Abstract: A new hybrid conjugate gradient method for unconstrained optimization by using homotopy formula, we computed the parameter $\beta_k$ as a convex combination of $\beta^{PR}$ (Polak-Ribiere (PR))\cite{9} and $\beta^{BA}$ (Al-Bayati and Al-Assady (BA))\cite{1}.

Keywords: Unconstrained optimization, line search, conjugate gradient method, homotopy formula.

I. Introduction

Consider the nonlinear following unconstrained optimization problem

$$\min f(x)$$

$$x \in \mathbb{R}^n$$

Where $f: \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable function whose gradient is denoted by $g(x)$. For solving this problem, starting from an initial vector $x_0 \in \mathbb{R}^n$, a nonlinear gradient algorithms generates a sequence $\{x_k\}$ as:

$$x_{k+1} = x_k + \alpha_k d_k \quad k=0,1,\ldots,n$$

Where $\alpha_k > 0$ is obtained by line search, the direction $d_k$ is generated by

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

and $d_0 = -g_0$.

and $\beta_k \in \mathbb{R}$ is a parameter which determines the different conjugate gradient methods. There are some well-known formulas which are given as follows:

$$\beta^{PR} = \frac{g_{k+1}^T (g_{k+1} - g_k)}{||g_k||^2}$$

$$\beta^{HS} = \frac{g_{k+1}^T (g_{k+1} - g_k)}{(g_{k+1} - g_k)^T d_k}$$

$$\beta^{CD} = \frac{g_{k+1}^T g_{k+1}}{d_k^T g_k}$$

$$\beta^{BA} = \frac{\|g_k\|^2}{d_k^T g_k}$$

Where $g_{k-1}$ and $g_k$ are gradients $\nabla f(x_{k-1})$ and $\nabla f(x_k)$ of $f(x)$ at the point $x_{k-1}$ and $x_k$, respectively. $\|\|$ denotes the Euclidian norm of vectors. The line search in the conjugate gradient algorithms often is based on the standard Wolfe conditions:

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \delta \alpha_k g_k^T d_k$$

$$g(x_k + \alpha_k d_k)^T d_k \geq \sigma g_k^T d_k$$

where $0 < \delta < \sigma < 1$.

The strong Wolfe line search corresponds to: that

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \delta \alpha_k g_k^T d_k$$

$$|g(x_k + \alpha_k d_k)^T d_k| \leq \sigma |g_k^T d_k|$$

where $0 < \delta < \sigma < 1$\cite{10}.
II. Hybrid Conjugate Gradient Algorithms

The hybrid conjugate gradient algorithms are combinations of different conjugate gradient algorithms, mainly with the purpose of avoiding the jamming phenomenon.[2].

The methods of (FR)[5], (DY) [3] and (CD)[4] have strong convergence properties, but they may have modest practical performance due to jamming.

On the other hand, the methods of (PR)[9], (HS)[6] and (LS)[8] may not always be convergent, but the often have better computational performances.[2]

III. New Hybrid conjugate gradient algorithm

The iterates \( x_0 , x_1, x_2, \ldots \) of new hybrid conjugate gradient algorithm computed by means of the recurrence \( x_{k+1} = x_k + \alpha_k d_k \), where the stepsize \( \alpha_k > 0 \) is determined according to the Wolf line search condition (1.10) and (1.11), and the directions \( d_k \) are computed by the rule:

\[
d_{k+1} = -\gamma_{k+1} d_{k+1}^{NEW} d_k,
\]

Where

\[
\beta_{k}^{NEW} = (1 - \theta_k) \beta_{k}^{PR} + \theta_k \beta_{k}^{BA},
\]

\[
= (1 - \theta_k) \frac{\langle g_k, y_k \rangle}{\|g_k\|^2} + \theta_k \frac{\|y_k\|^2}{d_k^T y_k}
\]

And \( \theta_k \) is a scalar parameter satisfying \( 0 \leq \theta_k \leq 1 \) which will be determined in a specific way to be described later. Observe that if \( \theta_k = 0 \), then \( \beta_k^{NEW} = \beta_k^{PR} \) and if \( \theta_k = 1 \), then \( \beta_k^{NEW} = \beta_k^{BA} \). On the other hand if \( 0 < \theta_k < 1 \), then \( \beta_k^{NEW} \) is a convex combination of \( \beta_k^{PR} \) and \( \beta_k^{BA} \).

The parameter \( \theta_k \) is selected in such a way that at every iteration the conjugacy condition is satisfied independently of the line search.

Obviously

\[
d_{k+1} = -\gamma_{k+1} (1 - \theta_k) \frac{\langle g_k, y_k \rangle}{\|g_k\|^2} d_k + \theta_k \frac{\|y_k\|^2}{d_k^T y_k} d_k,
\]

So, multiply both sides of above equation by \( y_k \) and we use conjugacy condition\( d_{k+1}^T y_k = 0 \)

\[
0 = -\gamma_{k+1} \frac{\langle g_k, y_k \rangle}{\|g_k\|^2} d_k^T y_k + \theta_k \frac{\|y_k\|^2}{d_k^T y_k} d_k^T y_k,
\]

Implies that

\[
g_{k+1}^T y_k - \frac{\langle g_k, y_k \rangle}{\|g_k\|^2} d_k^T y_k = -\theta_k \frac{\langle g_k, y_k \rangle}{\|g_k\|^2} d_k^T y_k + \theta_k \|y_k\|^2
\]

Finally, we have

\[
\theta_k = \frac{\|g_k\|^2 - \|g_k^T y_k\|}{\|g_k\|^2 - \|g_k^T y_k\|}
\]

**Theorem 3.1:** Assume that \( d_k \) is descent direction and \( \alpha_k \) in the algorithm (1.2),(3.2) and (3.5) determined by the wolfe line search (1.10) and (1.11). If \( 0 < \theta_k < 1 \), then the \( d_{k+1} \) is given by (3.3) is a descent direction.

**Proof:**

Multiply both sides of (2.3.3) by \( g_{k+1} \), we have

\[
d_{k+1}^T g_{k+1} = -\|g_{k+1}\|^2 + (1 - \theta_k) \frac{\langle g_k, y_k \rangle}{\|g_k\|^2} d_k^T g_{k+1} + \theta_k \frac{\|y_k\|^2}{d_k^T y_k} d_k^T g_{k+1}
\]

Since \( 0 < \theta_k < 1 \), then

\[
d_{k+1}^T g_{k+1} = -\|g_{k+1}\|^2 + \frac{\|y_k\|^2}{d_k^T y_k} d_k^T g_{k+1} + \frac{\|g_k^T y_k\|}{\|g_k\|^2} d_k^T g_{k+1}
\]

This prove is complete if the step length \( \alpha_k \) is chosen by an exact line search which requires \( d_k^T g_{k+1} = 0 \).

If the step length \( \alpha_k \) is chosen by an inexact line search which requires \( d_k^T g_{k+1} \neq 0 \).

We know that the first two terms of equation (3.6) are less than or equal to zero because the algorithm of (PR) is satisfies the descent condition (i.e)

\[
\|g_{k+1}\|^2 + \frac{\|y_k\|^2}{d_k^T y_k} d_k^T g_{k+1} \leq 0
\]

Now, we will explain the third term is less than or equal to zero

We know that

\[
\sigma d_k^T g_{k+1} \leq d_k^T y_k
\]
Where \( \sigma \) is positive, multiply both sides by \((-1)\), we have

\[
-\sigma d_k^T g_k \geq -d_k^T y_k
\]

Implies that

\[
\|g_k\|^2 \geq -\frac{1}{\sigma} d_k^T y_k
\]  

(3.8)

Then

\[
\frac{(g_{k+1}^T y_k)}{\|g_k\|^2} d_k^T g_{k+1} \leq -\frac{(g_{k+1}^T y_k)}{d_k^T y_k} d_k^T g_{k+1} \leq -\sigma \|g_{k+1}\|^2 \leq 0.
\]

Because \((g_{k+1}^T y_k)

\[
d_k^T g_{k+1} \leq \|g_{k+1}\|^2
\]

Then the proof is completed.\(\blacksquare\)

### 3.1 The global convergence

The following assumption are often needed to prove the convergence of the nonlinear conjugate gradient method (see [7]).

**Assumption:**

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

ii. In some neighborhood \( N \) of \( L \), the objective function \( f \) is continuously differentiable, and it is gradient is Lipschitz continuous, i.e., there exists a constant \( k > 0 \) such that

\[
\|\nabla f(x) - \nabla f(x)\| \leq k \|x - \bar{x}\|,
\]

for all \( x, \bar{x} \in N \).

Under above Assumption, there exists a constant \( \tau \) such that \( \|\nabla f(x)\| \leq \tau \), for all \( x \in L \),

**Lemma:** Let assumption (i) and (ii) hold and consider any conjugate gradient method (3.2) and (3.5), where \( d_k \) is a descent direction and \( \alpha_k \) is obtained by the strong Wolfe line search. If

\[
\sum_{k=1}^{\infty} \frac{1}{\|d_k\|^2} = \infty,
\]

Then

\[
\lim_{k \to \infty} \|g_k\| = 0. \tag{3.1.2}
\]

For uniformly convex function which satisfy the above assumptions, we can prove that the norm of \( d_{k+1} \) given by (3) is bounded above. Assume that the function \( f \) is uniformly convex function, i.e., there exists a constant \( \Gamma \geq 0 \), such that for all \( x, y \in L \),

\[
(x - \bar{x})^T (x - \bar{x}) \geq \Gamma \|x - \bar{x}\|^2
\]

and the step length \( \alpha_k \) is given by the strong Wolfe line search.

\[
f(x_{k+1}) \leq f(x_k) + \alpha_k g_k^T d_k,
\]

(3.14)

\[
|g_{k+1}^T d_k| \leq -\sigma_2 g_k d_k.
\]

(3.15)

Using lemma the following result can be proved.

**Theorem 3.2:** Suppose that the assumption (i) and (ii) hold. Consider the algorithm (1.2), (3.2) and (3.5), where \( 0 \leq \theta_k \leq 1 \) and \( \alpha_k \) is obtained by the strong Wolfe line search (1.12) and (1.13). If \( d_k \) tends to zero and there exists nonnegative constants \( \delta_1 \) and \( \delta_2 \) such that

\[
\|g_k\|^2 \geq \delta_1 \|v_k\|^2, \quad \text{and} \quad \|g_{k+1}\|^2 \leq \delta_2 \|v_k\|^2
\]

And \( f \) is a uniformly convex function, then

\[
\lim_{k \to \infty} g_k = 0. \tag{3.1.6}
\]

**Proof:**

From (3.1.3) it follows that

\[
y_k^T v_k \geq \Gamma \|v_k\|^2.
\]

Since \( 0 \leq \theta_k \leq 1 \), from uniform convexity and (3.2) we have

\[
|\beta_k^N| = \frac{(g_{k+1}^T y_k)}{\|g_k\|^2} \leq \frac{(g_{k+1}^T y_k)}{d_k^T y_k} \leq \frac{|y_k|^2}{d_k^T y_k}
\]

\[
|\beta_k^N| \leq \frac{|\|g_k\|^2 + \|y_k\|^2}{\delta_1 \|v_k\|^2} \quad \text{and} \quad \frac{|\|g_k\|^2 + \|y_k\|^2}{\delta_2 \|v_k\|^2}
\]

(3.17)

From Lipschitz Condition

We get

\[
|y_k| \leq K \|v_k\|.
\]

Then (3.1.6) gives
New Iterative Conjugate Gradient Method for Nonlinear Unconstrained Optimization Using ….

\[ |\beta_k^{\text{NEW}}| \leq \frac{\mu K}{\delta_1} + \frac{K^2}{\Gamma} \]  
\hfill (3.1.8)

Hence
\[ \|d_{k+1}\| \leq \|g_{k+1}\| + |\beta_k^{\text{NEW}}| \|d_k\| \]

Then
\[ \|d_{k+1}\| \leq \mu + \frac{\alpha \mu K}{\delta_1} + \frac{K^2}{\Gamma} \alpha \omega \omega \]  
\hfill (3.1.9)

Since \( \|v_k\| \rightarrow 0 \) as \( \rightarrow \infty \), then there exist a constant \( \omega > 0 \) such that
\[ \|v_k\| \leq \omega \]

Hence (3.1.9) gives
\[ \|d_{k+1}\| \leq \mu + \frac{\alpha \mu K}{\delta_1} + \frac{K^2}{\Gamma} \alpha \omega \omega \]

Which implies that (3.1.1) is true. So, by Lemma we have (3.1.2), which for uniformly convex functions is equivalent to (3.1.6).

3.2 Algorithm of New hybrid Conjugate Gradient coefficient:
step (1) : set \( k=0 \), select the initial point \( x_k \).

step (2) : \( g_k = \nabla f(x_k) \). If \( g_k = 0 \) then stop .

else

set \( d_k = -g_k \).

step (3) : compute \( \alpha_k > 0 \) satisfying the wolfe line search condition to minimize \( f(x_{k+1}) \).

step (4) : \( x_{k+1} = x_k + \alpha_k d_k \).

step (5) : \( g_{k+1} = \nabla f(x_{k+1}) \). If \( g_{k+1} = 0 \) then stop .

step (6) : compute \( \theta_k \) as in (3.5).

Step (7) : if \( 0 < \theta_k < 1 \), then compute \( \beta_k^{\text{NEW}} \) as in (3.2). If \( \theta_k \geq 1 \), then set \( \beta_k^{\text{NEW}} = \beta^{\text{BA}} \). If \( \theta_k \leq 0 \), then set \( \beta_k^{\text{NEW}} = \beta^{\text{PR}} \).

step (8) : \( d_{k+1} = g_{k+1} + \beta_k^{\text{NEW}} d_k \).

step (9) : if \( k=n \) then go to step 2, otherwise set \( k=k+1 \) and go to step 3.

3.3 Numerical Results:-
This section is devoted to test the implementation of the new methods. We compare the hybrid algorithm first suggestion with standard Polak-Ribiere (PR) and Al-Bayati (BA) ,the comparative tests involve well-known nonlinear problems (standard test function) with different dimension \( 4 \leq n \leq 5000 \), all programs are written in FORTRAN95 language and for all cases the stopping condition is \( \|g_{k+1}\|_{\infty} \leq 10^{-5} \). The results are given in table is specifically quote the number of functions NOF and the number of iteration NOI.

Experimental results in table confirm that the new CG method is superior to standard CG method with respect to the NOI and NOF.

<table>
<thead>
<tr>
<th>Test fun.</th>
<th>Standard formula (PR)</th>
<th>Standard formula (BA)</th>
<th>New formula</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NOI</td>
<td>NOF</td>
<td>NOI</td>
</tr>
<tr>
<td>Powell</td>
<td>4</td>
<td>60</td>
<td>162</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>119</td>
<td>252</td>
</tr>
<tr>
<td></td>
<td>500</td>
<td>505</td>
<td>1028</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>1002</td>
<td>2009</td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>2531</td>
<td>5067</td>
</tr>
<tr>
<td></td>
<td>5000</td>
<td>2817</td>
<td>5639</td>
</tr>
<tr>
<td>Wood</td>
<td>4</td>
<td>33</td>
<td>74</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>103</td>
<td>213</td>
</tr>
<tr>
<td></td>
<td>500</td>
<td>108</td>
<td>223</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>110</td>
<td>227</td>
</tr>
<tr>
<td></td>
<td>3000</td>
<td>188</td>
<td>282</td>
</tr>
<tr>
<td></td>
<td>5000</td>
<td>130</td>
<td>267</td>
</tr>
</tbody>
</table>
New Iterative Conjugate Gradient Method for Nonlinear Unconstrained Optimization Using.....

<table>
<thead>
<tr>
<th>Cubic</th>
<th>4</th>
<th>16</th>
<th>42</th>
<th>25</th>
<th>66</th>
<th>15</th>
<th>43</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>15</td>
<td>40</td>
<td>46</td>
<td>109</td>
<td>14</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>500</td>
<td>15</td>
<td>40</td>
<td>46</td>
<td>109</td>
<td>14</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>15</td>
<td>40</td>
<td>46</td>
<td>109</td>
<td>14</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>3000</td>
<td>15</td>
<td>40</td>
<td>49</td>
<td>105</td>
<td>14</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>5000</td>
<td>15</td>
<td>40</td>
<td>49</td>
<td>105</td>
<td>14</td>
<td>40</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rosen</th>
<th>4</th>
<th>22</th>
<th>63</th>
<th>68</th>
<th>247</th>
<th>23</th>
<th>66</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>22</td>
<td>61</td>
<td>113</td>
<td>248</td>
<td>17</td>
<td>52</td>
<td></td>
</tr>
<tr>
<td>500</td>
<td>22</td>
<td>61</td>
<td>484</td>
<td>990</td>
<td>17</td>
<td>52</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>22</td>
<td>61</td>
<td>502</td>
<td>1026</td>
<td>17</td>
<td>52</td>
<td></td>
</tr>
<tr>
<td>3000</td>
<td>22</td>
<td>61</td>
<td>526</td>
<td>1074</td>
<td>17</td>
<td>52</td>
<td></td>
</tr>
<tr>
<td>5000</td>
<td>22</td>
<td>61</td>
<td>542</td>
<td>1106</td>
<td>17</td>
<td>52</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Beel</th>
<th>4</th>
<th>10</th>
<th>23</th>
<th>25</th>
<th>59</th>
<th>9</th>
<th>22</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>9</td>
<td>23</td>
<td>40</td>
<td>89</td>
<td>9</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>500</td>
<td>9</td>
<td>23</td>
<td>40</td>
<td>89</td>
<td>9</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>9</td>
<td>23</td>
<td>40</td>
<td>89</td>
<td>9</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>3000</td>
<td>9</td>
<td>23</td>
<td>40</td>
<td>89</td>
<td>9</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>5000</td>
<td>9</td>
<td>23</td>
<td>40</td>
<td>89</td>
<td>9</td>
<td>23</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Non-Diagonal</th>
<th>4</th>
<th>23</th>
<th>59</th>
<th>40</th>
<th>115</th>
<th>23</th>
<th>61</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>22</td>
<td>62</td>
<td>105</td>
<td>462</td>
<td>22</td>
<td>60</td>
<td></td>
</tr>
<tr>
<td>500</td>
<td>24</td>
<td>68</td>
<td>466</td>
<td>222</td>
<td>22</td>
<td>59</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>222</td>
<td>22</td>
<td>59</td>
<td></td>
</tr>
<tr>
<td>3000</td>
<td>*</td>
<td>*</td>
<td>84</td>
<td>203</td>
<td>22</td>
<td>59</td>
<td></td>
</tr>
<tr>
<td>5000</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>22</td>
<td>59</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>EGR</th>
<th>4</th>
<th>6</th>
<th>16</th>
<th>9</th>
<th>24</th>
<th>6</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>6</td>
<td>16</td>
<td>33</td>
<td>70</td>
<td>6</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>500</td>
<td>6</td>
<td>16</td>
<td>35</td>
<td>74</td>
<td>6</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>6</td>
<td>16</td>
<td>36</td>
<td>76</td>
<td>6</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>3000</td>
<td>6</td>
<td>16</td>
<td>37</td>
<td>78</td>
<td>6</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>5000</td>
<td>6</td>
<td>16</td>
<td>37</td>
<td>78</td>
<td>6</td>
<td>15</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Extended PSC1</th>
<th>4</th>
<th>36</th>
<th>78</th>
<th>66</th>
<th>220</th>
<th>34</th>
<th>77</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>36</td>
<td>78</td>
<td>66</td>
<td>220</td>
<td>34</td>
<td>77</td>
<td></td>
</tr>
<tr>
<td>500</td>
<td>38</td>
<td>83</td>
<td>68</td>
<td>222</td>
<td>36</td>
<td>82</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>42</td>
<td>88</td>
<td>68</td>
<td>222</td>
<td>36</td>
<td>82</td>
<td></td>
</tr>
<tr>
<td>3000</td>
<td>42</td>
<td>88</td>
<td>68</td>
<td>222</td>
<td>36</td>
<td>82</td>
<td></td>
</tr>
<tr>
<td>5000</td>
<td>42</td>
<td>88</td>
<td>68</td>
<td>222</td>
<td>36</td>
<td>82</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Generalized PSC1</th>
<th>4</th>
<th>60</th>
<th>166</th>
<th>120</th>
<th>352</th>
<th>60</th>
<th>166</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>104</td>
<td>276</td>
<td>124</td>
<td>358</td>
<td>102</td>
<td>272</td>
<td></td>
</tr>
<tr>
<td>500</td>
<td>106</td>
<td>278</td>
<td>124</td>
<td>358</td>
<td>104</td>
<td>276</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>108</td>
<td>282</td>
<td>124</td>
<td>358</td>
<td>104</td>
<td>276</td>
<td></td>
</tr>
<tr>
<td>3000</td>
<td>108</td>
<td>282</td>
<td>124</td>
<td>358</td>
<td>104</td>
<td>276</td>
<td></td>
</tr>
<tr>
<td>5000</td>
<td>108</td>
<td>282</td>
<td>124</td>
<td>358</td>
<td>104</td>
<td>276</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Q-Central</th>
<th>4</th>
<th>25</th>
<th>163</th>
<th>45</th>
<th>227</th>
<th>25</th>
<th>148</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>19</td>
<td>115</td>
<td>*</td>
<td>*</td>
<td>20</td>
<td>132</td>
<td></td>
</tr>
<tr>
<td>500</td>
<td>23</td>
<td>172</td>
<td>*</td>
<td>*</td>
<td>20</td>
<td>132</td>
<td></td>
</tr>
<tr>
<td>1000</td>
<td>23</td>
<td>172</td>
<td>*</td>
<td>*</td>
<td>21</td>
<td>147</td>
<td></td>
</tr>
<tr>
<td>3000</td>
<td>23</td>
<td>172</td>
<td>*</td>
<td>*</td>
<td>21</td>
<td>147</td>
<td></td>
</tr>
<tr>
<td>5000</td>
<td>23</td>
<td>172</td>
<td>*</td>
<td>*</td>
<td>21</td>
<td>147</td>
<td></td>
</tr>
</tbody>
</table>

### IV. Conclusion

In this paper we have presented a new hybrid conjugate gradient method in which a famous parameter $\beta_k$ is computed as a convex combination of $\beta_k^{PR}$ and $\beta_k^{BA}$, and comparative numerical performances of a number of well known conjugate gradient algorithms Polak-Ribiare-Polyak (PR) and Al-Bayati and Al-Assady (BA). We saw that the performance profile of our method was higher than those of the well established conjugate gradient algorithms PR and BA.

### References


