A variant of hybrid conjugate gradient methods based on the convex combination for optimization

Basim A. Hassan¹, Ahmed O. Owaid², Zena T. Yasen³

¹Department of Mathematics, College of Computers Sciences and Mathematics, University of Mosul, Iraq
²Department of studies and planning, Presidency of Mosul University, University of Mosul, Iraq

ABSTRACT

On some studies a conjugate parameter plays an important role for the conjugate gradient methods. In this paper, a variant of hybrid is provided in the search direction based on the convex combination. This search direction ensures that the descent condition holds. The global convergence of the variant of hybrid is also obtained. Our strong evidence is a numerical analysis showing that the proposed variant of hybrid method is efficient than the Hestenes and Stiefel method.

Keywords:
Descent property
Global convergence property
Hybrid conjugate gradient

1. INTRODUCTION

The conjugate gradient [1] is an adaptation of the optimization method to large-scale problems. Here, we matter with the following problem:

\[ \text{Min } f(x), \quad x \in \mathbb{R}^n \]  

(1)

to denote the objective function, and \( g(x) \) denote the gradient \( f(x) \), respectively.

Many conjugate gradient methods for (1) produce a sequence \( \{x_k\} \) by the recurrence:

\[ x_0 \in \mathbb{R}^n, \quad x_{k+1} = x_k + \alpha_k d_k \]  

(2)

where \( d_k \) is a search direction and \( \alpha_k \) is a step size which was calculated by applying line search rules, the most important and famous is the Wolfe conditions:

\[ f(x_k + \alpha_k d_k) \leq f(x_k) + \delta \alpha_k g_k^T d_k \]  

(3)

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

(4)

Journal homepage: http://ijeecs.iaescore.com
where $0 < \delta < \sigma$. For details see [2].

Therefore, this study will focus on the CG method whose direction of search is written as:

$$d_{k+1} = -g_{k+1} + \beta_k d_k,$$

where $\beta_k$ is a scalar known as the CG update. The suitable choice for $\beta_k$ leads to improve numerical performance conjugate gradient methods. There are some well known and effective conjugate gradient methods, such as, [3-6].

The two most popular formulas known in conjugate gradient methods are as follows:

$$\beta_k^{DY} = \frac{\|g_{k+1}\|^2}{d_k^T y_k}, \quad \beta_k^{HS} = \frac{g_{k+1}^T y_k}{d_k^T y_k}$$

where $y_k = g_{k+1} - g_k$. The first is better in terms of convergence and the second is better in numerical performance. More details can be found in [7, 8].

Yuan [9], claimed that the parameter conjugate gradient method has best numerical performance is given by:

$$\beta_k^Y = \frac{g_{k+1}^T y_k}{(f(x_k) - f(x_{k+1}))/\alpha_k - g_k^T d_k/2}$$

Newly, in [10], presented a modified Yuan method has good convergence property is given by:

$$\beta_k^B = \frac{g_{k+1}^T g_{k+1}}{(f(x_k) - f(x_{k+1}))/\alpha_k - g_k^T d_k/2}.$$  

The global convergence property is an important property of conjugate gradient methods, and has as many results as we can see in [1, 11].

The urge of this paper is to get great algorithms we combine the benefits of the two formulas $\beta_k^Y$ and $\beta_k^B$. Provide novel algorithms have fantastic numerical performance with good better convergence and a discussion for the theoretical properties and the numerical results.

2. A CONVEX COMBINATION OF $\beta_k^B$ AND $\beta_k^Y$ METHODS

In vision of the gentle convergence property of the $\beta_k^B$ method, and the efficient performance of the $\beta_k^Y$ method, we paying attention on designing new methods which possess the above properties simultaneously. The parameter $\beta_k$ of the hybrid conjugate gradient method of $\beta_k^B$ and $\beta_k^Y$ is formulized as:

$$\beta_k^{HYB} = (1 - \theta_k) \beta_k^Y + \theta_k \beta_k^B$$

So, we get:

$$d_{k+1}^{HYB} = -g_{k+1} + \beta_k^{HYB} s_k, \quad s_k = x_{k+1} - x_k$$

To be determined the scalar parameter $\theta_k$, the see it : if $\theta_k = 0$, then $\beta_k^{HYB} = \beta_k^Y$ and $\theta_k = 1$, then $\beta_k^{HYB} = \beta_k^B$. If $0 < \theta_k < 1$, then $\beta_k^{HYB}$ is a proper convex combination of the parameters is $\beta_k^Y$ and $\beta_k^B$. 

Theorem 2.1:
If the relationships (9) and (10) holds, then:
\[
d_{k+1}^{HYB} = (1 - \gamma_k)d_k^Y + \gamma_k d_{k+1}^B
\]
(11)

Proof:
Possessing perspective relationships \( \beta_k^b \) and \( \beta_k^r \), the relationship (9) becomes:
\[
\beta_k^{HYB} = \frac{(1-\gamma_k)g_{k+1}^Ty_k + \gamma_k g_{k+1}^Tg_{k+1}}{(f(x_k) - f(x_{k+1}))/\alpha_k - g_i^Td_k/2}
\]
(12)

So, the relation (10) becomes:
\[
d_0^{HYB} = -g_0, \quad d_{k+1}^{HYB} = -g_{k+1} + \frac{(1-\gamma_k)g_{k+1}^Ty_k + \gamma_k g_{k+1}^Tg_{k+1}}{(f(x_k) - f(x_{k+1}))/\alpha_k - g_i^Td_k/2} s_k
\]
(13)

In the other mind of a relationship (13) we have:
\[
d_{k+1}^{HYB} = -(\gamma_k g_{k+1} + (1 - \gamma_k)g_{k+1}) + \beta_k^{HYB} s_k,
\]
(14)

Implies that:
\[
d_{k+1}^{HYB} = -(\gamma_k g_{k+1} + (1 - \gamma_k)g_{k+1}) + ((1 - \gamma_k)\beta_k^r + \gamma_k \beta_k^b) s_k
\]
(15)

The last relation yields:
\[
d_{k+1}^{HYB} = \gamma_k(-g_{k+1} + \beta_k^b s_k) + (1 - \gamma_k)(-g_{k+1} + \beta_k^r s_k)
\]
(16)

From (16) we finally conclude:
\[
d_{k+1}^{HYB} = (1 - \gamma_k)d_k^Y + \gamma_k d_{k+1}^B
\]
(17)

Apply conjugacy condition to find value \( \gamma_k \) in our method:
\[
y_k^T d_{k}^{HYB} = 0
\]
(18)

Multiplying (13) by \( y_k^T \) and applying (18) we get:
\[
y_k^T \begin{bmatrix} -g_{k+1} + \frac{(1-\gamma_k)g_{k+1}^Ty_k + \gamma_k g_{k+1}^Tg_{k+1}}{(f(x_k) - f(x_{k+1}))/\alpha_k - g_i^Td_k/2} s_k \end{bmatrix} = 0
\]
(19)

\[
y_k^T \begin{bmatrix} -g_{k+1} + \frac{(1-\gamma_k)g_{k+1}^Ty_k + \gamma_k g_{k+1}^Tg_{k+1}(y_k^T s_k)}{(f(x_k) - f(x_{k+1}))/\alpha_k - g_i^Td_k/2} s_k \end{bmatrix} = 0,
\]
(20)

So,
\[
y_k^T g_{k+1} = \frac{g_{k+1}^Ty_k}{(f(x_k) - f(x_{k+1}))/\alpha_k - g_i^Td_k/2} (y_k^T s_k) = \gamma_k \frac{g_{k+1}^T g_{k+1} (y_k^T s_k) - g_{k+1}^T y_k (y_k^T s_k)}{(f(x_k) - f(x_{k+1}))/\alpha_k - g_i^Td_k/2}
\]
(21)
\[
\frac{(f(x_k) - f(x_{k+1}))/a_1 - g_k^T d_k / 2 - y_k^T s_k)}{(f(x_1) - f(x_{k+1}))/a_1 - g_k^T d_k / 2} (y_k^T g_{k+1}) = \delta_k \frac{g_k^T g_k}{(f(x_1) - f(x_{k+1}))/a_1 - g_k^T d_k / 2} (y_k^T s_k)
\]  

(22)

Finally,

\[
\delta_k = \frac{(f(x_k) - f(x_{k+1}))/\alpha_k - g_k^T d_k / 2 - y_k^T s_k)(y_k^T g_{k+1})}{g_k^T g_k (y_k^T s_k)}
\]  

(23)

The appropriate value for \( \delta_k \) is in the interval \([0, 1]\). We get \( \beta_k^{\text{HYB}} = \beta_k^B \) if \( \delta_k \leq 0 \), then set \( \delta_k = 0 \) in (9), and we get \( \beta_k^{\text{HYB}} = \beta_k^Y \) if \( \delta_k \geq 1 \), then set \( \delta_k = 1 \) in (9). If it's a value of \( \delta_k \) is specified, \( \beta_k^{\text{HYB}} \) it is combines the properties of the Y and the B algorithms in a convex way.

Now we formally state our algorithm and call by HYB:

**Algorithm (HYB):**

Stage 1. Initialization: \( x_1 \in \mathbb{R}^n \), \( 0 < \delta_1 < \delta_2 < 1 \), \( d_1 = -g_1 \) and \( \alpha_1 = 1/\|g_1\| \).

Stage 2. If \( \|g_{k+1}\| \leq 10^{-6} \), then stop.

Stage 3. Compute: \( \alpha_k \) by using the Wolfe condition and let \( x_{k+1} = x_k + \alpha_k d_k \).

Stage 4. Computation \( \delta_k \), if \( g_k^T g_k (y_k^T s_k) = 0 \), then set \( \delta_k = 0 \), else set \( \delta_k \) as in (23) respectively.

Stage 5. Compute: \( \beta_k^{\text{HYB}} \) as in (9). 

Stage 6. Compute: \( d_{k+1} = -g_{k+1} + \beta_k s_k \). Put \( k = k + 1 \) go to step 2.

3. CONVERGENCE ANALYSIS

We adopt the following assumption used often in the literature to analyze the global convergence.

**Assumption 3.1:**

suppose that the gradient is “Lipschitz continuous”. Then for some positive constant \( L \), we have:

\[
\|g(n) - g(m)\| \leq L\|n - m\|, \quad \forall \, n, m \in U.
\]  

(24)

where we assume the level set:

\[
S = \{x \in \mathbb{R}^n \mid f(x) \leq f(x_0)\}
\]  

(25)

is bounded set. Under these assumptions of \( f(x) \), there exists a constant \( \Gamma \geq 0 \) such that:

\[
\|x_{k+1}\| \leq \|x_k\| \leq \Gamma
\]  

(26)

More details can be found in [12, 13].

Now, we will focus the proof of the descent property.

**Theorem (3.1)**

Assume that (24) and (26) hold and let Wolfe conditions hold. Also, let \( \|s_k\| \) tend to zero, and consider there exist two positive constants \( \bar{c}_1, \bar{c}_2 \) we satisfies:

\[
(f(x_k) - f(x_{k+1})) / \alpha_k - g_k^T d_k / 2 \geq \bar{c}_1 \|s_k\|^2.
\]  

(27)
\[
\|g_{k+1}\|^2 \leq \frac{\varepsilon_2}{\varepsilon_1} \|s_k\| 
\]  
(28)

then \(d_k^{\text{HYB}}\) satisfies the \(g_{k+1}^Td_k^{\text{HYB}} \leq 0\), for all \(k\).

**Proof:**

It fistis \(d_0 = -g_0\). If \(k = 0\), it holds \(g_0^Td_0 = -\|g_0\|^2 < 0\). Multiplying (11) by \(g_{k+1}^T\), we obtained:

\[
g_{k+1}^Td_{k+1}^{\text{HYB}} = (1 - \theta_k)g_{k+1}^Td_{k+1}^{\text{Y}} + \theta_k g_{k+1}^Td_{k+1}^{B}
\]  
(29)

If \(\theta_k = 0\), the relation (29) becomes:

\[
g_{k+1}^Td_{k+1}^{\text{HYB}} = g_{k+1}^Td_{k+1}^{\text{Y}}
\]  
(30)

So, if \(\theta_k = 0\), the hybrid method satisfies the descent condition, if it fistis for \(\text{Y}\) method. By using the conditions of Theorem 3.1, we can prove the descent for \(\text{Y}\) method. It fistis:

\[
d_{k+1}^{\text{Y}} = -g_{k+1} + \beta_k^Y s_k
\]  
(31)

Multiplying (31) by \(g_{k+1}^T\), we get:

\[
g_{k+1}^Td_{k+1}^{\text{Y}} = -g_{k+1}^Tg_{k+1} + \beta_k^Y g_{k+1}^Ts_k
\]  
(32)

Using definition \(\beta_k^Y\), we get:

\[
g_{k+1}^Td_{k+1}^{\text{Y}} = -g_{k+1}^Tg_{k+1} + \beta_k^Y \left(\frac{g_{k+1}^Ty_k}{(f(x_k) - f(x_{k+1}))} / \alpha_k - g_{k}^Tg_k / 2 \right) g_{k+1}^Ts_k
\]  
(33)

From (33) we get:

\[
g_{k+1}^Td_{k+1}^{\text{Y}} \leq -\|g_{k+1}\|^2 + \frac{\|g_{k+1}\| \|y_k\|}{(f(x_k) - f(x_{k+1})) / \alpha_k - g_{k}^Tg_k / 2} \|s_k\|
\]  
(34)

From Lipschitz condition we have \(\|y_k\| \leq L \|s_k\|\) so:

\[
g_{k+1}^Td_{k+1}^{\text{Y}} \leq -\|g_{k+1}\|^2 + \frac{\|g_{k+1}\| L \|s_k\|}{(f(x_k) - f(x_{k+1})) / \alpha_k - g_{k}^Tg_k / 2}
\]  
(35)

But, using (27)-(28) we get:

\[
g_{k+1}^Td_{k+1}^{\text{Y}} \leq -\|g_{k+1}\|^2 + \frac{\varepsilon_2}{\varepsilon_1}  L \|s_k\|
\]  
(36)

But, benefit from \(\|s_k\| \Rightarrow 0\), the second part in (36) tends to zero, so there exists a \(0 < \delta \leq 1\), such that:

\[
\frac{1}{\varepsilon_1} \varepsilon_2  L \|s_k\| \leq \delta \|g_{k+1}\|^2
\]  
(37)
Now, from (36) we have:
\[ g^T_{k+1}d^Y_{k+1} \leq -\|g_{k+1}\|^2 + \delta \|g_{k+1}\|^2, \]  
(38)
i.e.
\[ g^T_{k+1}d^Y_{k+1} \leq -(1 - \delta)\|g_{k+1}\|^2 < 0 \]  
(39)
If either \( \delta_k = 1 \), the relationship (29) we have:
\[ g^T_{k+1}d^{HYB}_{k+1} = g^T_{k+1}d^B_{k+1} \]  
(40)
But, the descent property of B-method is proved in [10], by using Wolfe conditions.
Now, let \( 0 < \delta_k < 1 \) and from (29), we get:
\[ g^T_{k+1}d^{HYB}_{k+1} \leq (1 - \delta_k)g^T_{k+1}d^Y_{k+1} + \delta_k g^T_{k+1}d^B_{k+1} \]  
(41)
We obviously can conclude now:
\[ g^T_{k+1}d^{HYB}_{k+1} \leq 0 \]  
(42)
We adopt the next theorem used commonly in the research literatures.

**Theorem (3.2)**

Let the iterative method of the form (2) and (5), where \( d_k \) satisfies a \( g^T_kd_k < 0 \) and \( \alpha_k \) satisfies strong Wolfe conditions. If (24) holds, then either,
\[ \lim_{k \to \infty} \inf \|g_k\| = 0 \]  
(43)
Or,
\[ \sum_{k \geq 1} \|g_{k+1}\|^2 < \infty \]  
(44)
It was originally given by Zoutendijk [14].

**Theorem (3.3)**

Let conditions of Theorem 3.2 holds. Then either \( g_k = 0 \) for some \( k \), or,
\[ \lim_{k \to \infty} \inf \|g_k\| = 0 \]  
(45)
**Proof:**

Let \( g_k \neq 0, \forall k \). Using, a contrary to prove (45), that there exists a number \( c > 0 \), such that:
\[ g_{k+1} \geq c, \forall k \]  
(46)
From (11) we get:
\[
\|d_{k+1}^{HYB}\| \leq \|d_{k+1}^Y\| + \|d_{k+1}^B\| \tag{47}
\]

Next, it holds:
\[
\|d_{k+1}^B\| \leq \|g_{k+1}\| + \beta_k^B \|s_k\| \tag{48}
\]

From (9), (26)-(28), and (48) we get:
\[
\|d_{k+1}^B\| \leq \Psi + \frac{\xi_2}{\xi_1} \tag{49}
\]

Also,
\[
\|d_{k+1}^Y\| \leq \|g_{k+1}\| + \beta_k^Y \|s_k\| \tag{50}
\]

Using (26)-(28) and (50) we get:
\[
\|d_{k+1}^Y\| \leq \Psi + \frac{\Psi L}{\xi_1} \tag{51}
\]

So, using (47), (49) and (51) we get:
\[
\|d_{k+1}^{HYB}\| \leq 2\Psi + \frac{\Psi L}{\xi_1} + \frac{\xi_2}{\xi_1} \tag{52}
\]

We obtained:
\[
\frac{\|g_{k+1}\|^4}{\|d_{k+1}\|^2} \geq \frac{c^4}{\left(2\Psi + \frac{\Psi L}{\xi_1} + \frac{\xi_2}{\xi_1}\right)^2} \tag{53}
\]

where from:
\[
\sum_{k=1}^{\infty} \frac{\|g_{k+1}\|^4}{\|d_{k+1}\|^2} = \infty \tag{54}
\]

Applying the Theorem 3.2, we obtain a contradiction. The proof is finish.

4. NUMERICAL RESULTS AND DISCUSSION

In this section, we report some numerical results with the proposed method and HS-method. We test the performance of Algorithm 2 on the following (15) problems with various sizes. Using Fortran 90 to encrypt these methods. In our application, we choose the following parameters : \( \delta_1 = 0.001 \) and \( \delta_2 = 0.9 \) are selected examination problems of references [15]. Stop state is \( \|g_{k+1}\| \leq 10^{-6} \).

It has been reported numerical results in Table 1. Represents the first column and the name of the second problem and its dimensions in [15], respectively. Other test functions have been used in various
REFERENCES

Mathematics for their provided facilities, which helped to improve the quality of this work.

ACKNOWLEDGMENT

The authors are very grateful to the University of Mosul / College of Computers Sciences and Mathematics for their provided facilities, which helped to improve the quality of this work.

REFERENCES


5. CONCLUSIONS

The variant of hybrid of the conjugate gradient direction is based on the convex combination. The search direction in the variant of hybrid satisfies as a descent condition and the global convergence is also obtained. Show that variant of hybrid method are the best for solving unconstrained optimization problems.

ACKNOWLEDGMENT

The authors are very grateful to the University of Mosul / College of Computers Sciences and Mathematics for their provided facilities, which helped to improve the quality of this work.

Table 1. Comparing different conjugate gradient methods with different test functions

<table>
<thead>
<tr>
<th>P. No.</th>
<th>n</th>
<th>NI</th>
<th>NR</th>
<th>NF</th>
<th>NI</th>
<th>NR</th>
<th>NF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trigonometric</td>
<td>100</td>
<td>19</td>
<td>10</td>
<td>35</td>
<td>20</td>
<td>11</td>
<td>37</td>
</tr>
<tr>
<td>Perturbed Quadratic</td>
<td>100</td>
<td>102</td>
<td>33</td>
<td>155</td>
<td>19</td>
<td>26</td>
<td>148</td>
</tr>
<tr>
<td>Extended PSC</td>
<td>100</td>
<td>326</td>
<td>91</td>
<td>509</td>
<td>137</td>
<td>92</td>
<td>309</td>
</tr>
<tr>
<td>Q. Dragonal Perturbed</td>
<td>100</td>
<td>26</td>
<td>25</td>
<td>505</td>
<td>7</td>
<td>5</td>
<td>15</td>
</tr>
<tr>
<td>Extended Wood</td>
<td>100</td>
<td>47</td>
<td>6</td>
<td>84</td>
<td>53</td>
<td>9</td>
<td>95</td>
</tr>
<tr>
<td>ARWHEAD (CUTE)</td>
<td>100</td>
<td>188</td>
<td>29</td>
<td>335</td>
<td>173</td>
<td>24</td>
<td>309</td>
</tr>
<tr>
<td>NONDIA (CUTE)</td>
<td>100</td>
<td>17</td>
<td>9</td>
<td>34</td>
<td>13</td>
<td>7</td>
<td>26</td>
</tr>
<tr>
<td>Partial Perturbed Q.</td>
<td>100</td>
<td>260</td>
<td>57</td>
<td>429</td>
<td>259</td>
<td>65</td>
<td>429</td>
</tr>
<tr>
<td>LIARWHD (CUTE)</td>
<td>100</td>
<td>21</td>
<td>12</td>
<td>40</td>
<td>19</td>
<td>11</td>
<td>34</td>
</tr>
<tr>
<td>DENSCHNF (CUTE)</td>
<td>100</td>
<td>32</td>
<td>13</td>
<td>60</td>
<td>34</td>
<td>11</td>
<td>64</td>
</tr>
<tr>
<td>SINCOS</td>
<td>100</td>
<td>28</td>
<td>11</td>
<td>54</td>
<td>28</td>
<td>9</td>
<td>49</td>
</tr>
<tr>
<td>Generalized quartic GQ2</td>
<td>100</td>
<td>55</td>
<td>34</td>
<td>10</td>
<td>55</td>
<td>31</td>
<td>7</td>
</tr>
<tr>
<td>Diagonal 2</td>
<td>100</td>
<td>12</td>
<td>12</td>
<td>45</td>
<td>19</td>
<td>11</td>
<td>44</td>
</tr>
<tr>
<td>Extended Three Expo Terms</td>
<td>100</td>
<td>20</td>
<td>17</td>
<td>36</td>
<td>18</td>
<td>17</td>
<td>32</td>
</tr>
<tr>
<td>Quadratic QF1</td>
<td>100</td>
<td>22</td>
<td>17</td>
<td>41</td>
<td>18</td>
<td>16</td>
<td>33</td>
</tr>
<tr>
<td>Total</td>
<td>100</td>
<td>309</td>
<td>173</td>
<td>684</td>
<td>1985</td>
<td>648</td>
<td>387</td>
</tr>
</tbody>
</table>

From Table 1, it is completed that the more efficient algorithm in terms of the number of iterations is our method, being the best for 6% of the problems, followed by number of function evaluations, that was the efficient for nearly 22% of the problems and 6% in terms of the number of restart calls.

Table 2. Relativa efficiency of the variant of hybrid algorithm

<table>
<thead>
<tr>
<th></th>
<th>NI</th>
<th>NR</th>
<th>NF</th>
</tr>
</thead>
<tbody>
<tr>
<td>HS algorithm</td>
<td>100 %</td>
<td>100 %</td>
<td>100 %</td>
</tr>
<tr>
<td>HYB algorithm</td>
<td>94.74 %</td>
<td>94.73 %</td>
<td>77.94 %</td>
</tr>
</tbody>
</table>

Table 2. Relativa efficiency of the variant of hybrid algorithm

NI | NR | NF |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>HS algorithm</td>
<td>100 %</td>
<td>100 %</td>
</tr>
<tr>
<td>HYB algorithm</td>
<td>94.74 %</td>
<td>94.73 %</td>
</tr>
</tbody>
</table>
A Variant of Hybrid Conjugate Gradient Methods Based on the Convex Combination… (Basim A. Hassan)