Two-versions of descent conjugate gradient methods for large-scale unconstrained optimization

Hawraz N. Jabbar¹, Basim A. Hassan²
¹Department of Mathematics, College of Sciences, University of Kirkuk, Iraq
²Department of Mathematics, College of Computers Sciences and Mathematics, University of Mosul, Iraq

ABSTRACT

The conjugate gradient methods are noted to be exceedingly valuable for solving large-scale unconstrained optimization problems since it needn’t the storage of matrices. Mostly the parameter conjugate is the focus for conjugate gradient methods. The current paper proposes new methods of parameter of conjugate gradient type to solve problems of large-scale unconstrained optimization. A Hessian approximation in a diagonal matrix form on the basis of second and third-order Taylor series expansion was employed in this study. The sufficient descent property for the proposed algorithm are proved. The new method was converged globally. This new algorithm is found to be competitive to the algorithm of fletcher-reaves (FR) in a number of numerical experiments.

1. INTRODUCTION

The problem of unconstrained optimization is generally formulated as:

\[ \min \{ f(x) \mid x \in \mathbb{R}^n \} \]  

where \( f: \mathbb{R}^n \rightarrow \mathbb{R}^1 \) is a function that is continuously differentiable. Numerous famous techniques are found for solving (1); however, the conjugate gradient (CG) techniques are the mainly characterized ones. Newton technique is famous if the gradient matrix is non-negative definite, for more details see [1]. These CG-techniques are in the variety of iterations known by:

\[ x_0 \in \mathbb{R}^n, x_{k+1} = x_k + \lambda_k S_k \]  

where \( \lambda_k \) is the step length, as a rule obtained by the Wolfe line search:

\[ f(x_k) - f(x_k + \lambda_k S_k) \geq -\alpha \lambda_k \eta_k^T S_k \]  
\[ g(x_k + \lambda_k S_k)^T S_k \geq \sigma \eta_k^T S_k \]  

This is an open access article under the CC BY-SA license.
where $0 < \alpha < \sigma < 1$. The iterative searcher directions $S_k$ CG-technique are calculated as:

$$d_0 = -g_0 d_{k+1} = -g_{k+1} + \beta_k d_k$$  

(5)

at this point $\beta_k$ is a scalar given as the parameter of conjugate gradient, $\eta_{k+1}$ denotes gradient of $f(x_{k+1})$ at the points $x_{k+1}, s_k = x_{k+1} - x_k$ and $y_k = \eta_{k+1} - \eta_k$. The next sufficient descent state (6):

$$\eta_{k+1}^T S_{k+1} \leq -c\|\eta_{k+1}\|^2$$  

(6)

A lot used for analyzing the worldwide CG-technique convergence in mixture with inexact techniques of line search [2]. In the technique of quasi-Newton (QN), the direction of search is calculated using an approximation of the Hessian matrix inverse. In meticulous (5) is changed by:

$$S_{k+1} = -B_{k+1}^{-1} \eta_{k+1}$$  

(7)

where by the Hessian matrix $B_{k+1} = G_{k+1} = \nabla^2 f(x_{k+1})$ is updated during the iterations. More details can be found in [3], [4]. In modern years, a diversity of CG-formulas was known, majorly, differences are in the parameter $\beta_k$, the work by discussed details on some CG-techniques with special emphasis on their worldwide convergence. Furthermore, the design of CG-techniques had been studied by many of researchers for archetype refer to [5]-[10].

In this paper, the new proposed method is solved by second and third-order Taylor-series. The subsequent sections of study are organized in this way: the second section presents the outlines of the new algorithm and the deriving a new formula. Some interesting the convergence analysis of the new algorithm presented in the third section. Results of the current numerical experiments are presented in the fourth section by using the test problems found in [11]. Finally, the fifth section presents some obvious findings.

2. A NEW CONJUGATE GRADIENT METHOD

This section develops a new CG-method on the basis of approximating the Hessian with a symmetric positive-definite matrix. Now, the second and third-order Taylor-series approximation is employed to $f$ at the point $x_k$ can be written as by following the same approaches as in [12] as:

$$f(x) = f(x_{k+1}) - \eta^T_{k+1} r_k + \frac{1}{2} r_k^T G_{k+1} r_k, f_k = f_{k+1} - \eta^T_{k+1} r_k + \frac{1}{2} r_k^T G_{k+1} r_k + \frac{1}{2} r_k^T \gamma_{k+1} r_k$$  

(8)

where $T_{k+1}$ is the tensor of $f$ at the point $x_{k+1}$. Then, by using a $\eta^T_{k+1} S_k = 0$ in second -order Taylor-series, the next relation (9) is obtained:

$$r_k^T G_{k+1} r_k = 2(f(x_k) - f(x_{k+1}))$$  

(9)

the relation (10) is obtained by third-order Taylor-series expressions:

$$r_k^T G_{k+1} r_k = y_k^T r_k + 6(f_k - f_{k+1}) + 3(\eta_{k+1} + \eta_k)^T r_k$$  

(10)

the step size $\lambda_k$ is determined by many algorithms. In exact line search the step length $\lambda_k$ is selected as (11).

$$\lambda_k = -\frac{\eta^T S_k}{\eta_k^T G_k}$$  

(11)

From some algebra, the (12) is obtained:

$$r_k^T G_{k+1} r_k = f(x_k) - f(x_{k+1}) - \frac{\lambda_k \eta^T S_k}{2} r_k^T G_{k+1} r_k = \frac{1}{2} y_k^T r_k + 3(f_k - f_{k+1}) + \frac{3}{2} \eta^T_{k+1} r_k + \eta^T_k r_k$$  

(12)

by (12), (13) is derived and denote by $G_{k+1}^Q$ and as follows:

$$G_{k+1}^Q = \frac{f(x_k) - f(x_{k+1})}{r_k^T r_k} - \frac{\lambda_k \eta^T S_k}{2} I_{n \times n}, G_{k+1} = \frac{1}{2} \frac{y_k^T r_k + 3(f_k - f_{k+1}) + \frac{3}{2} \eta^T_{k+1} r_k + \eta^T_k r_k}{r_k^T r_k} I_{n \times n}$$  

(13)
then, it can be written as:

\[ S_{k+1}^\Theta = -
\left(\frac{r_k^Tr_k}{f(x_k) - f(x_{k+1}) - \lambda_k\eta_kS_k/2}\right) \eta_{k+1},
\quad S_{k+1}^\Gamma = -\left(\frac{r_k^Tr_k}{2y_k^T\eta_k + 3(f_k - f_{k+1}) + \frac{3}{2}\eta_{k+1}^2 + \eta_k^2r_k^Tr_k}\right) \eta_{k+1} \quad (14) \]

by use the conjugacy condition \( S_{k+1}^\eta y_k = 0 \) due to the conjugacy of Newton directions with exact line searches.

\[ S_{k+1}^\Theta y_k = -\left(\frac{r_k^Tr_k}{f(x_k) - f(x_{k+1}) - \lambda_k\eta_kS_k/2}\right) \eta_{k+1}y_k = 0 \]
\[ S_{k+1}^\Gamma y_k = -\left(\frac{r_k^Tr_k}{2y_k^T\eta_k + 3(f_k - f_{k+1}) + \frac{3}{2}\eta_{k+1}^2 + \eta_k^2r_k^Tr_k}\right) \eta_{k+1}y_k = 0 \quad (15) \]

Similarly, by using CG methods for quadratic functions with exact line searches, formula (16) is obtained:

\[ S_{k+1}^\Theta y_k = -\eta_{k+1}^2y_k + \beta_k^\Theta S_k^\Theta y_k = 0 \quad (16) \]

from (15) and (16), the (17 a and b) is derived as follows:

\[ -\left(\frac{r_k^Tr_k}{f(x_k) - f(x_{k+1}) - \lambda_k\eta_kS_k/2}\right) \eta_{k+1}y_k = -\eta_{k+1}^2y_k + \beta_k^\Theta S_k^\Theta y_k \]
\[ -\left(\frac{1}{2y_k^T\eta_k + 3(f_k - f_{k+1}) + \frac{3}{2}\eta_{k+1}^2 + \eta_k^2r_k^Tr_k}\right) \eta_{k+1}y_k = -\eta_{k+1}^2y_k + \beta_k^\Theta S_k^\Theta y_k \quad (17 a) \]

from above equation, we get:

\[ \beta_k^\Theta S_k^\Theta y_k = -\left(\frac{r_k^Tr_k}{f(x_k) - f(x_{k+1}) - \lambda_k\eta_kS_k/2}\right) \eta_{k+1}y_k + \eta_{k+1}^2y_k \]
\[ \beta_k^\Theta g_k y_k = -\left(\frac{1}{2y_k^T\eta_k + 3(f_k - f_{k+1}) + \frac{3}{2}\eta_{k+1}^2 + \eta_k^2r_k^Tr_k}\right) \eta_{k+1}y_k + \eta_{k+1}^2y_k \quad (17 b) \]

then, the following equations are obtained:

\[ \beta_k^{\Theta Q} = \left(1 - \frac{r_k^Tr_k}{f(x_k) - f(x_{k+1}) - \lambda_k\eta_kS_k/2}\right) \eta_{k+1}y_k, \quad \beta_k^{\Theta C} = \left(1 - \frac{r_k^Tr_k}{2y_k^T\eta_k + 3(f_k - f_{k+1}) + \frac{3}{2}\eta_{k+1}^2 + \eta_k^2r_k^Tr_k}\right) \eta_{k+1}y_k \quad (18) \]

putting (18) in (5), we obtained:

\[ S_{k+1} = -\eta_{k+1} + \left(1 - \frac{r_k^Tr_k}{f(x_k) - f(x_{k+1}) - \lambda_k\eta_kS_k/2}\right) g_k y_k S_k \]
\[ S_{k+1} = -g_k + \left(1 - \frac{1}{2y_k^T\eta_k + 3(f_k - f_{k+1}) + \frac{3}{2}\eta_{k+1}^2 + \eta_k^2r_k^Tr_k}\right) \eta_{k+1}y_k S_k \quad (19) \]

for simplicity, equation (19) is called by \( \beta_k \) method. Also, \( \beta_k \) can be written in this way and denoted by \( \beta_k^{\Theta Q} \) and \( \beta_k^{\Theta C} \):

\[ \beta_k^{\Theta Q} = \left(\frac{1}{\eta_k y_k} \left(y_k - r_k^Tr_k\right)\right)^T \eta_{k+1}, \quad \beta_k^{\Theta C} = \left(\frac{1}{\eta_k y_k} \left(y_k - r_k^Tr_k\right)\right)^T \eta_{k+1} \]

where,

\[ T_1 = \left(\frac{\left(r_k^Tr_k\right)^2}{\eta_k y_k} \left[r_k^Tr_k \left(\frac{r_k^Tr_k}{f(x_k) - f(x_{k+1}) - \lambda_k\eta_kS_k/2}\right)\right] \right), \quad T_2 = \left(\frac{\left(r_k^Tr_k\right)^2}{\eta_k y_k} \left[r_k^Tr_k \left(\frac{1}{2y_k^T\eta_k + 3(f_k - f_{k+1}) + \frac{3}{2}\eta_{k+1}^2 + \eta_k^2r_k^Tr_k}\right)\right] \right) \]
On the basis of above discussion, this section describes the algorithm frame of this study without fixed line search in this way. New algorithms (BTQ and BTC algorithms):

Step 1: Give $x_1 \in R^n$, $\varepsilon > 0$. Set $\eta_1 = -\eta_1$, $k = 1$. If $\|\eta_1\| \leq 10^{-6}$, then stop.

Step 2: Compute $\lambda_k$ satisfying the conditions (3-4).

Step 3: Let $x_{k+1} = x_k + \lambda_k S_k$ and $\eta_{k+1} = \eta(x_{k+1})$. If $\|\eta_{k+1}\| \leq 10^{-6}$, then stop.

Step 4: Compute $\beta_k$ by the formula (12) then generate $S_{k+1}$ by equation (13)

Step 5: Set $k = k + 1$ and continue with stage 2.

3. CONVERGENT ANALYSIS

The following section proves the property of global convergence of new method. Theorem 3.1 demonstrates that the direction of search in algorithms is continuously sufficient descent based on no line search. The property of sufficient descent is one of the important properties of the all conjugate gradient methods.

3.1. Theorem

Let $r_k, y_k, \eta_{k+1} \in R^n$, $\beta_k \in R$ and $\beta_k^{BTC} = \frac{1}{r_k^T y_k} \left( y_k - \tau \frac{\|y_k\|^2}{r_k^T y_k} r_k \right)^T \eta_{k+1}$, where $\tau \in (1/4, \infty)$. If $r_k^T y_k \neq 0$, then $S_{k+1} \eta_{k+1} \leq -[1 - 1/4\tau]\|\eta_{k+1}\|^2$.

Proof: Since $S_0 = -\eta_0$, we have $\eta_0^T S_0 = -\|\eta_0\|^2$, satisfying (6). Through multiplying (19) by $\eta_{k+1}$ (20) is obtained:

$$S_{k+1} \eta_{k+1} = -\|\eta_{k+1}\|^2 + \left( \frac{\eta_{k+1}^T y_k}{r_k^T y_k} - \tau \frac{\|y_k\|^2}{(r_k^T y_k)^2} \eta_{k+1}^T y_k \right) r_k^T \eta_{k+1}$$

yielding

$$S_{k+1} \eta_{k+1} = \left( \frac{\eta_{k+1}^T y_k}{r_k^T y_k} (r_k^T y_k) - \|y_k\|^2 \eta_{k+1}^T y_k \right) r_k^T \eta_{k+1}$$

The inequality $\omega^T v \leq \frac{1}{2} (\|\omega\|^2 + \|v\|^2)$ is applied with $\omega = \frac{1}{\delta} (r_k^T y_k) \eta_{k+1}$ and $v = \delta (\eta_{k+1}^T y_k) y_k$, where $\delta \in (\frac{1}{\sqrt{2}}, \sqrt{2})$, to the first term of the above equality, the (23) is obtained:

$$(\eta_{k+1}^T y_k) (r_k^T \eta_{k+1}) (r_k^T y_k) \leq \frac{1}{2} \left( \frac{1}{\delta^2} (r_k^T y_k)^2 \|\eta_{k+1}\|^2 + \delta^2 (r_k^T \eta_{k+1})^2 \|y_k\|^2 \right)$$

this yields,

$$S_{k+1} \eta_{k+1} \leq \frac{1}{2\delta^2 - 1} \left( \frac{1}{\delta^2} (r_k^T y_k)^2 \|\eta_{k+1}\|^2 \right) \|y_k\|^2 \left( \frac{1}{\delta^2} (r_k^T \eta_{k+1})^2 \|\eta_{k+1}\|^2 \right) \|y_k\|^2$$

From (18), the (24) is derived as follows:

$$S_{k+1} \eta_{k+1} \leq \frac{1}{2\delta^2 - 1} \left( 1 - \frac{1}{\delta^2} \right) \|\eta_{k+1}\|^2$$

Therefore, the (25) is obtained:

$$S_{k+1} \eta_{k+1} \leq \frac{1}{4\delta} \|\eta_{k+1}\|^2$$

Consequently, it is necessary to have Assumption 3.2 for analyzing the global convergence of algorithms.

3.2. Assumption

i. The level set $L = \{x \in R^n | f(x) \leq f(x_0)\}$ is constrained.

ii. In a number of areas, $U$ and $L$, $f(x)$ are continuously differentiable and their gradient is Lipschitz continuous, i.e., a constant $L > 0$ exists, like that:
\begin{equation}
\|\eta(z) - \eta(o)\| \leq L\|z - o\|, \forall z, o \in U
\end{equation}

Under the above assumptions on \( f \), a constant \( \Gamma > 0 \) exists, like that:

\begin{equation}
\|\eta_{k+1}\| \geq \Gamma
\end{equation}

for all \( x \in L \). More details can be found in [13] verified that the next general result is applied to any CG method with strong Wolfe line search:

3.3. Lemma

Supposing that assumptions (i) and (ii) are held, then consider any method of conjugate gradient (2) and (5) where \( S_{k+1} \) is a descent direction and \( \lambda_k \) is achieved by the strong Wolfe line search (3) and (4). If:

\begin{equation}
\sum_{k=0}^{1} \frac{1}{\|S_{k+1}\|^2} = \infty,
\end{equation}

then,

\begin{equation}
limit_{k \to \infty} \inf \|\eta_{k+1}\| = 0
\end{equation}

3.4. Theorem

Supposing that assumptions are held, then consider methods (2) and (5), where is a descent direction with and given by (18), and \( \lambda_k \) is found by the Wolfe line search. If the objective function is uniformly, then

\begin{equation}
limit_{n \to \infty} \inf \|\eta_{k+1}\| = 0.
\end{equation}

This relation shows that:

\begin{equation}
\sum_{k=1}^{1} \frac{1}{\|S_{k+1}\|^2} \geq \left( \frac{1}{2+\sigma} \right) \sum_{k=1}^{1} \|S_{k+1}\|^2 = \infty
\end{equation}

based on Lemma 1, \( \lim \inf \|\eta_{k+1}\| = 0 \) is derived, which equals \( \lim \inf \|\eta_{k+1}\| = 0 \) for uniformly convex function.

4. NUMERICAL RESULTS

This section explains some numerical experiments conducted for testing BTQ and BTC algorithms. Some test problems studied by Andrei [11] were used in this study (see Table 1) to analyze the efficiency of the new formula formed in this study in comparison to the method of FR. Comparison is based on iterations number (NI) and function evaluations number (NF) the CG algorithms by steepest descent directions. In all CG, the step length \( \lambda_k \) is yielded by Wolfe line search with \( \alpha = 0.001 \) and \( \sigma = 0.9 \), and the termination condition is \( \|\eta_{k+1}\| \leq 10^{-6} \). Some noted papers can be see [14]-[25].

Tables 1 present list of some numerical results of this study. Based on the current numerical results, the proposed methods, BTQ and BTC, have minimum numbers of iterations, restarts and function evaluations in all implemented test problems in this study, except for problems 7 and 10, where the FR algorithm has less numbers of iterations, restarts and function evaluations against the new proposed BTQ and BTC algorithms. Generally, the percentage performance of the new proposed algorithms BTQ and BTC can be computed as compared to the standard FR algorithm for the general Tools NI, NR and NF shown in Table 2.
Table 1. Comparison of FR and new algorithms (BTQ and BTC) with n=100 and n=1000, test function

<table>
<thead>
<tr>
<th>P. No</th>
<th>n</th>
<th>FR algorithm</th>
<th>BTQ algorithm</th>
<th>BTC algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>NI</td>
<td>NR</td>
<td>NF</td>
</tr>
<tr>
<td>1</td>
<td>100</td>
<td>47</td>
<td>93</td>
<td>38</td>
</tr>
<tr>
<td>2</td>
<td>100</td>
<td>78</td>
<td>131</td>
<td>37</td>
</tr>
<tr>
<td>3</td>
<td>100</td>
<td>43</td>
<td>88</td>
<td>43</td>
</tr>
<tr>
<td>4</td>
<td>100</td>
<td>32</td>
<td>52</td>
<td>15</td>
</tr>
<tr>
<td>5</td>
<td>100</td>
<td>25</td>
<td>43</td>
<td>23</td>
</tr>
<tr>
<td>6</td>
<td>100</td>
<td>46</td>
<td>741</td>
<td>30</td>
</tr>
<tr>
<td>7</td>
<td>100</td>
<td>15</td>
<td>31</td>
<td>11</td>
</tr>
<tr>
<td>8</td>
<td>100</td>
<td>32</td>
<td>65</td>
<td>21</td>
</tr>
<tr>
<td>9</td>
<td>100</td>
<td>74</td>
<td>123</td>
<td>92</td>
</tr>
<tr>
<td>10</td>
<td>100</td>
<td>71</td>
<td>110</td>
<td>40</td>
</tr>
<tr>
<td>11</td>
<td>100</td>
<td>32</td>
<td>65</td>
<td>21</td>
</tr>
<tr>
<td>12</td>
<td>100</td>
<td>49</td>
<td>80</td>
<td>10</td>
</tr>
<tr>
<td>13</td>
<td>100</td>
<td>12</td>
<td>25</td>
<td>11</td>
</tr>
<tr>
<td>14</td>
<td>100</td>
<td>122</td>
<td>156</td>
<td>14</td>
</tr>
<tr>
<td>15</td>
<td>100</td>
<td>112</td>
<td>147</td>
<td>43</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>2157</td>
<td>7090</td>
<td>1343</td>
</tr>
</tbody>
</table>

Table 2. Relative efficiency of the new algorithms

<table>
<thead>
<tr>
<th></th>
<th>FR algorithm</th>
<th>BTQ algorithm</th>
<th>BTC algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NI</td>
<td>100%</td>
<td>62.26%</td>
</tr>
<tr>
<td></td>
<td>NF</td>
<td>100%</td>
<td>37.60%</td>
</tr>
</tbody>
</table>

5. CONCLUSIONS

Practically, when the complexity and size of the test problem increase, greater improvements could be realized by the new algorithms because the new proposed algorithm is more stable and always preserves the descent search directions. Our reported results showed that the proposed methods are efficient for solving large-scale unconstrained optimization. Generally, the percentage performance of the new proposed algorithms BTQ and BTC can be computed as compared to the standard FR algorithm for the general tools NI, NR and NF.

ACKNOWLEDGMENT

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

REFERENCE

Two versions of descent conjugate gradient methods for large-scale … (Hawraz N. Jabbar)