A New Preconditioned Inexact Line-Search Technique for Unconstrained Optimization

Abbas Y. Al-Bayati
College of Computer Sciences and Mathematics
University of Mosul

Ivan S. Latif
College of Scientific Education
University of Salahaddin

Received on: 18/12/2006
Accepted on: 16/04/2007

ABSTRACT

In this paper, we study the global convergence properties of the new class of preconditioned conjugate gradient descent algorithm, when applied to convex objective non-linear unconstrained optimization functions.

We assume that a new inexact line search rule which is similar to the Armijo line search rule is used. It's an estimation formula to choose a large step-size at each iteration and use the same formula to find the direction search. A new preconditioned conjugate gradient direction search is used to replace the conjugate gradient descent direction of ZIR-algorithm. Numerical results on twenty five well-know test functions with various dimensions show that the new inexact line-search and the new preconditioned conjugate gradient search directions are efficient for solving unconstrained nonlinear optimization problem in many situations.

Keywords: Preconditioned CG, Unconstrained Optimization, Self-Scaling VM-update, inexact Line-Search.

1. Introduction

Some important global convergence result for various methods using line-search procedures have been given [1], [4] the above mentioned line search methods are monotone descent for unconstrained optimization [10], [11]. Non monotone line-searches have been investigated also by many authors see [6], [9]. The Barzilai-Borwein method [2], [8] is a non monotone descent method which is an efficient algorithm for solving some special problem, Zirilli [12] extend the Armijo line search rule ant analyze the global convergence of the corresponding method.

In this paper, we extend the Armijo line-search rule so that we can design a new inexact line search technique and we choose the search directions of AL-Bayati Self-Scaling [3] variable metric update which based on two parameter family of rank-two updating formulae. Numerical results show that the new algorithm which enables us to
choose large step-size at each iteration and reduce the number of functions. The new algorithm is efficient for solving unconstrained optimization problems.

We consider the following unconstrained optimization problem of $n$ variables,

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

where $f(x)$ is twice continuously differentiable and its gradient $g$ is exist available. We consider iterations of the form

$$
x_{k+1} = x_k + \alpha_k d_k
$$

where $d_k$ is a search direction and $\alpha_k$ is the step-length obtained by means of one-dimensional search. In conjugate gradient method when the function is quadratic and the line search is exact, another broad class of methods may be defined by the following search direction:

$$
d_k = -H_k^{-1}g_k
$$

where $H_k$ is a non singular symmetric matrix. Important special cases are given by

- $H_k = I$ (Steepest descent direction)
- $H_k = \nabla^2 f(x_k)$ (Newton's direction)

Variable Metric (VM) methods are also of the form (3) and in this case $H_k$ is not only a function of $x_k$, but depends also on $H_{k-1}$ and $x_{k-1}$.

All these methods are implemented so that $d_k$ is a descent direction, i.e.

$$
d_k^T g_k < 0
$$

which guarantees that the function can be decreased by taking a small step along $d_k$ for the Newton type method (3). We can ensure that $d_k$ is a descent direction by defining $H_k$ to be positive definite.

For conjugate gradient method, obtaining descent direction is not easy and requires a careful choice properties of line search methods and it can be studied by measuring the goodness of the search direction and by considering the length of the step. The quality of the angle between the steepest descent direction $-g_k$ and the search direction. We can define:

$$
\cos(-g_k, d_k) = -g_k^T d_k / \|g_k\| \|d_k\| \geq \eta_0
$$

The length of the step is determined by the line search iteration. A strategy that will play a central role in this paper is to set scalars $s_k$, $\beta$, $L$, $\sigma > 0$ with:

$$
s_k = -g_k^T d_k / \|d_k\|^2, \quad \beta \in (0, 1); \quad \sigma \in (0, 1/2).
$$

Let $\alpha_k$ be the largest $\alpha$ in $\{s_k, \beta s_k, \beta^2 s_k, \ldots, \}$ such that

$$
f_k - f(x_k + \alpha d_k) \geq -\sigma g_k^T d_k
$$

The inequality ensures that the function is reduced sufficiently, we will call these relations as Armijo condition.

2. Zirlli Inexact Line-Search Algorithm (Zir):

Inexact line-search rule was implemented the following assumptions [7], [11].
A New Preconditioned Inexact Line-Search Technique for Unconstrained Optimization

(H1) The function \( f(x) \) has a lower bound on the level set
\[
L(x_0) = \{ x \in \mathbb{R}^n \mid f(x) - f(x_0) \} \text{ where } x_0 \text{ is given}
\]

(H2) The gradient \( g(x) \) of \( f(x) \) is Lipschitz continuous in an open convex set \( B \) that contains \( L_0 \) the; i.e., there exists \( L \) such that
\[
\| g(x) - g(y) \| \leq L \| x - y \| \quad \forall x, y \in B \quad \text{...(7)}
\]
The modified Armijo line search rule as [1]:

Set scalars \( \mu, \beta, \kappa, \mu \) and \( \sigma \) with \( s_k = -g_k^T d_k / (L \| d_k \|^2) \), \( \beta \in (0, 1), L_k > 0, \mu \in [0, 1) \) and \( \sigma \in (0, 1/2) \).

Let \( \alpha_k \) be the largest \( \alpha \) in \( \{ s_k, \beta s_k, \beta^2 s_k, \ldots \} \) such that
\[
f(x_k + \alpha d_k) - f_k \leq \sigma \alpha \left[ g_k^T d_k + \frac{1}{2} \alpha L \| d_k \|^2 \right] \quad \text{...(8)}
\]

2.1. Outlines of the Zir Algorithm:

The implementable inexact line search algorithm is stated as follows [12]:

Step1: Given some parameters, \( \sigma \in (0, 1/2), \quad x_0 \in \mathbb{R}^n, \quad \beta \in (0, 1), \mu \in (0, 2), \quad L_0 = 1 \)

let and set \( K = 0, \varepsilon \) is a small parameter.

Step2: If \( \| g_k \| \leq \varepsilon \) then stop. Else go to step3.

Step3: Choose \( d_k \), to satisfy the angle property (5) and set \( d_k = -g_k \).

Step4: Set \( x_{k+1} = x_k + \alpha_k d_k \), where \( \alpha_k \) is defined by the modified Armijo line search rule (8).

Step5: Set \( V_k = x_{k+1} - x_k ; \quad Y_k = g_{k+1} - g_k \) and \( L_{k+1} \) is determined by
\[
L_{k+1} = \frac{\| Y_k \|}{\| V_k \|} \quad \text{...(9)}
\]

Step6: Set \( k = k + 1 \) and go to step 2.

2.2. Some Properties of the Zir Algorithm:

Theorem 2.2.1: Assume that (H1) and (H2) hold, the search direction \( d_k \) satisfies (4) and \( \alpha_k \) is determined by the modified Armijo line-search rule. Zir Algorithm generates an infinite sequence \{\( x_n \)\} with
\[
0 < L_k < m_k L \quad \text{...(10)}
\]
where \( m_k \) is appositive integer and \( m_k \leq M_0 \leq \infty \) with \( M_0 \) being large positive constant then
\[
\sum_{k=1}^{\infty} \left( \frac{g_k^T d_k}{\| d_k \|} \right)^2 < +\infty \quad \text{...(11)}
\]
for the details of the proof see [12].
Corollary 2.2.1: If the condition in theorem 2.2.1 hold then
\[
\lim_{k \to \infty} \left( \frac{g_k^T d_k}{\|d_k\|} \right) = 0
\]  

In fact, Assumption (H2) can be replaced by the following weaker assumption. (H2') the gradient \( g(x) \) of \( f(x) \) is uniformly continuous on an open convex set \( B \) that contains \( L_0 \) see [9].

3. A New Proposed Preconditioned Inexact Line-Search Algorithm (New):

In this section we propose a new algorithm which implements the step-size \( \alpha_k \) with inexact line search rule. This formula is implemented with AL-Bayati self-scaling [3] variable metric update.

3.1. Outlines of the New Algorithm:

The outlines of the new proposed Algorithm are stated as follows:

Step1: Given some parameters \( \sigma \in (0, \frac{1}{2}) \), \( x_0 \in \mathbb{R}^n \), \( \beta \in (0,1) \), \( M = 10^6 \), \( H_0 \) is identity positive definite matrix and \( L_0 = 0.1 \). Let, \( \epsilon \) is a small parameter and set \( K = 0 \).

Step2: If \( \|g_k\| \leq \epsilon \) then stop. Else go to step3.

Step3: Choose \( d_k \) to satisfy the angle property (5) and satisfy the new search direction.
\[
d_k = \begin{cases} 
-H_k g_k, & \text{if } k = 1, \\
-H_k g_k + L_k' d_k, & \text{if } k \geq 1,
\end{cases}
\]  

Step4: Set \( x_{k+1} = x_k + \alpha_k d_k \) where \( \alpha_k \) is defined later by a new modified line search rule (19), (20).

Step5: Set \( V_k = x_{k+1} - x_k \), \( Y_k = g_{k+1} - g_k \) and \( L_{k+1} \) is determined by
\[
L_{k+1} = \min \left\{ L_k, \frac{g_k^T H_k Y_k}{\|V_k\|}, \frac{Y_k^T H_k Y_k}{\|V_k\|} \right\},
\]  

Step6: Update \( H_k \) by \( H_{k+1} \), see [3]
\[
H_{k+1} = \left( H_k - \frac{H_k Y_k Y_k^T H_k}{Y_k H_k Y_k} + W_k W_k^T \right) \mu_k + \frac{V_k V_k^T}{V_k Y_k}
\]  

\[
W_k = \left( Y_k^T H_k Y_k \right)^{1/2} \left[ \frac{V_k}{V_k Y_k} - \frac{H_k Y_k}{Y_k H_k Y_k} \right]
\]  

\[
\mu_k = \frac{Y_k^T H_k Y_k}{V_k Y_k}
\]

Step7: If available storage is exceeded then employ a restart option either with \( k = n \) or \( g_{k+1}^T g_k > g_k^T g_k \) i.e. orthogonality condition is not satisfy see [7].

Steps: Set \( k = k + 1 \) and go to step2.
3.2. Some Theoretical Properties of the New Algorithm:

We analyze the global convergence of the proposed new inexact line-search algorithm. For the proof of convergence we adopt the assumptions (H1), (H2') on the function \( f \) which is commonly used and we suppose that \( \{H_k\} \) is a sequence of positive definite matrices. Assume also that there exist parameters \( v_{\min} > 0 \) and \( v_{\max} > 0 \) such that \( \forall d \in \mathbb{R}^n \)

\[
    v_{\min} d^T d \leq d^T H_k d \leq v_{\max} d^T d \quad \text{(18)}
\]

this condition would be satisfied for instance, if \( H_k = H \) and \( H \) is positive definite as in Al-Bayati VM-update [3]. We analyze the conjugate gradient algorithm that use the following modified line-search formula: Set scalars \( \mu, \beta, L, \sigma \) with

\[
    s_k = -g_k^T d_k / \left( \|d_k\|_{\|H_k\|}^2 \right) \quad \text{...(19)}
\]

where, \( \beta \in (0, 1), L_k > 0 \) is a new parameter, \( \mu \in [0,2] \) and \( \sigma \in (0, v_{\min}/\mu) \). Note that the specification of \( \sigma \) ensures \( \frac{\rho \mu}{v_{\min}} < 1 \).

Let \( \alpha_k \) be the largest \( \alpha \in \{s_k, \beta s_k, \beta^2 s_k, \ldots\} \) such that

\[
    f(x_k + \alpha d_k) - f_k \leq \sigma \alpha \left[ g_k^T d_k + \left( \frac{1}{2} \right) \mu L_k \|d_k\|^2 \right] \quad \text{...(20)}
\]

where \( \|d_k\|_{\|H_k\|} = \sqrt{d_k^T H_k d_k} \)

**Lemma 3.2.1:** Suppose that \( x_k \) is given by the new proposed algorithm defined by \{2\), (13), (14) and (19)\} then

\[
    g_{k+1}^T d_k = \rho_k g_k^T d_k \quad \text{...(21)}
\]

holds for all \( k \), where

\[
    \rho_k = 1 - \frac{\phi_k g_k^T d_k}{L_k^* \|d_k\|^2} \quad \text{...(22)}
\]

and \( L_k^* \) is known as a new scalar defined in (14). Let

\[
    \phi_k = \begin{cases} 
    0 & \text{for } \alpha_k = 0 \\
    \frac{y_k^T V_k}{\|y_k\|^2} & \text{for } \alpha_k \neq 0 
    \end{cases} \quad \text{...(23)}
\]

**Proof:**

The case of \( \alpha_k = 0 \) implies that \( \rho_k = 1 \) and \( g_{k+1} = g_k \) hence (21) is valid, we now prove for the case of \( \alpha_k \neq 0 \) from (2) and new modified inexact line search \( \alpha_k \) we have

\[
    g_{k+1}^T d_k = g_k^T d_k + (g_{k+1} - g_k) d_k \\
    = g_k^T d_k + \alpha_k^{-1} (g_{k+1} - g_k) (x_{k+1} - x_k) \quad \text{from (2) we have } d_k = \alpha_k^{-1} (x_{k+1} - x_k)
\]
\[ = g_k^r d_k + \alpha_k^{-1} \phi_k \|x_{k+1} - x_k\|^2 \] from (23)
\[ = g_k^r d_k + \alpha_k^{-1} \phi_k \|d_k\|^2 \]
\[ = g_k^r d_k - \left( \frac{g_k^r d_k}{L_k \|d_k\|^2} \right) \phi_k \|d_k\|^2 \]
\[ = \left( 1 - \frac{\phi_k \|d_k\|^2}{L_k} \right) g_k^r d_k \] from (23)
\[ = \rho_k g_k^r d_k \]

The proof is complete. #

**Theorem 3.2.1:** If (H1) and (H2') hold, then the new algorithm generates an infinite number of sequences \( \{\alpha_k\} \) and satisfy

\[ 0 < L^*_k < mL < M \] …(24)

where \( m \) is a positive integer, \( M \) is a large positive constant then

\[ \lim_{k \to \infty} \left( - \frac{g_k^r d_k}{\|d_k\|^2} \right) = 0 \]

**Proof:**

Let \( K_1 = \{k \mid \alpha_k = s_k\} \), \( K_2 = \{k \mid \alpha_k < s_k\} \)

**Case (I):**

If \( k \in K_1 \) then

\[ f(x_k + \alpha d_k) - f_k \leq \sigma \left[ g_k^r d_k + \left( \frac{1}{2} \right) \alpha_k L^*_k \|d_k\|^2 \right] \]
\[ = -\sigma \left[ g_k^r d_k / L^*_k \|d_k\|^2 \right] \left( g_k^r d_k - \left( \frac{1}{2} \right) \mu g_k^r d_k \right) \]
\[ = - \left[ \sigma \left( 1 - \left( \frac{1}{2} \right) \mu \right) / L^*_k \right] \left( g_k^r d_k \right)^2 / \|d_k\|^2 \]

Thus

\[ f(x_k + \alpha d_k) - f_k \leq - \left[ \sigma \left( 1 - \left( \frac{1}{2} \right) \mu \right) / L^*_k \right] \left( g_k^r d_k \right)^2 / \|d_k\|^2 , \quad k \in K_1 \] …(27)

Let

\[ \eta_k = - \sigma \left( 1 - \left( \frac{1}{2} \right) \mu \right) / L^*_k , \quad k \in K_1 \]

By (24) we have

\[ \eta_k = - \sigma \left( 1 - \left( \frac{1}{2} \right) \mu \right) / L^*_k \]
\[ \leq -\sigma \left(1 - \left(\frac{1}{2}\right)\mu\right)/mL \]
\[ \leq -\sigma \left(1 - \left(\frac{1}{2}\right)\mu\right)/ML \]
\[ < 0 \]

Let
\[ \eta' \leq -\sigma \left(1 - \left(\frac{1}{2}\right)\mu\right)/ML \]

This and (27) imply that \( \eta_k \leq \eta' \) and
\[ f_{k+1} - f_k \leq \eta \left( g_i^* d_k / \| d_k \|^2 \right), \quad k \in K_1 \]
\[ \ldots (28) \]

Thus if \( k \in K_1 \), from (28) we can prove that
\[ \lim_{k \in K_1, k \to \infty} \left( -\frac{g_i^* d_k}{\| d_k \|^2} \right) = 0. \]

**Case (2):**

If \( k \in K_2 \), then \( \alpha_k < s_k \), this show that \( s_k \), can not satisfy the new suggested line search and thus \( \alpha_k < \beta s_k \) we show that \( \alpha = \alpha_k / \beta \) were \( \alpha_k \) be the largest \( \alpha \) in \( \{ s_k, \beta s_k, \beta^2 s_k, \ldots \} \) can not satisfy (14) and thus
\[ f(x_k + \frac{\alpha_k d_k}{\beta}) - f_k \leq \sigma \frac{\alpha_k \beta}{2} \left[ g_i^* d_k + \frac{\left( \frac{1}{2} \alpha_k \mu L_k^* \| d_k \|^2 \right)}{\beta} \right] \]

using the mean-value theorem on the left hand side of the above inequality, we see that there exists \( \theta_k \in [0,1] \) such that
\[ g \left( x_k + \frac{\theta_k \alpha_k d_k}{\beta} \right) > \sigma \frac{\alpha_k \beta}{2} \left[ g_i^* d_k + \frac{\left( \frac{1}{2} \alpha_k \mu L_k^* \| d_k \|^2 \right)}{\beta} \right] \]

Therefore
\[ g \left( x_k + \frac{\theta_k \alpha_k d_k}{\beta} \right) \cdot d_k > \rho \left[ g_i^* d_k + \frac{\left( \frac{1}{2} \alpha_k \mu L_k^* \| d_k \|^2 \right)}{\beta} \right] \]
\[ \ldots (30) \]

in this case of \( k \in K_2 \), by (19) and (20) we have
\[ f(x_k + \alpha d_k) - f_k \leq \sigma \alpha \left[ g_i^* d_k + \frac{\left( \frac{1}{2} \alpha_k \mu L_k^* \| d_k \|^2 \right)}{\beta} \right] \]
\[
\leq \sigma \alpha \left[ g_k^T d_k + \left( \frac{1}{2} \right) s_k L_k \| d_k \|^2 \right]
\]
\[
\leq \sigma \alpha_k \left[ 1 + \left( \frac{1}{2} \right) \mu \right] g_k^T d_k
\]

By (H1) we have
\[
\lim_{k \to K_2, k \to \infty} \left( \frac{-g_k^T d_k}{\| d_k \|} \right) = 0 \quad \text{...(31)}
\]

If there exist \( \varepsilon > 0 \) and an infinite subset \( K_3 \subseteq K_2 \) such that
\[
-\frac{g_k^T d_k}{\| d_k \|} \geq \varepsilon, \quad \forall k \in K_3 \quad \text{...(32)}
\]

then by (31), (32) we have
\[
\lim_{k \to K_3, k \to \infty} \alpha_k \| d_k \| = 0 \quad \text{...(33)}
\]

by (30) we have
\[
g \left( x_k + \frac{\theta_k \alpha_k d_k}{\beta} \right) \geq \rho \frac{g_k^T d_k}{\| d_k \|}, \quad k \in K_3 \quad \text{...(34)}
\]

where \( \theta_k \in [0,1] \) is defined in the proof. By the Cauchy Schwarz inequality and (34) we have
\[
\left\| g \left( x_k + \frac{\theta_k \alpha_k d_k}{\beta} \right) - g_k \right\| \geq \rho \left\| g_k^T d_k \right\| \|
\]
\[
\geq \left( g \left( x_k + \frac{\theta_k \alpha_k d_k}{\beta} \right) - g_k \right)^T \frac{d_k}{\| d_k \|}
\]
\[
\geq \frac{- (1 - \rho) g_k^T d_k}{\| d_k \|^2}, \quad k \in K_3
\]

by (H2') and (31) we obtain
\[
\lim_{k \to K_1, k \to \infty} \left( \frac{-g_k^T d_k}{\| d_k \|} \right) = 0
\]

which contradicts (32) this show that
\[
\lim_{k \to K_3, k \to \infty} \left( \frac{-g_k^T d_k}{\| d_k \|} \right) = 0 \quad \text{...(35)}
\]

by (29), (35) and noting that \( K_1 \cup K_2 = \{1,2,\ldots\} \) we show that (25) holds. #

Lemma 3.2.2.: suppose that (H1), (H2') holds and \( x_k \) is given by the new proposed algorithm defined by \{(2), (13), (14) and (19)\} then
\[
\sum_{d_k \neq 0} - \frac{g_k^T d_k}{\|d_k\|^2} < \infty \quad \ldots(36)
\]

**Proof:**

By the mean value theorem we have

\[
f(x_{k+1}) - f(x_k) = g^T (x_{k+1} - x_k)
\]

from (19) we have

\[
f(x_{k+1}) - f(x_k) \leq -\sigma \left[1 - \left(\frac{1}{2}\right) \mu L_d\right] \left(\frac{g_k^T d_k}{\|d_k\|^2}\right)^2 \quad \ldots(37)
\]

which implies that \( f(x_{k+1}) \leq f(x_k) \). It follows by assumption (H1), (H2') that \( \lim_{k \to \infty} f(x_k) \) exists thus from (18) and (37) we have

\[
\frac{(g_k^T d_k)^2}{\|d_k\|^2} \leq \frac{v_{\max}}{\sigma \left[1 - \left(\frac{1}{2}\right) \mu L_d\right]} \left[f(x_{k+1}) - f(x_k)\right]
\]

this finishes our proof. 

4. Numerical results:

In this section, we compare the numerical behavior of the new algorithm with the Zir algorithm for different dimensions of test functions. Comparative tests were performed with (25) (specified in the Appendices 1 and 2) to test functions see [5]. All the results are obtained with newly-programmed FORTRAN routines which employ double precautions. We solve each of these test function by the:

1- Zirlli algorithm (Zir).
2- The new algorithm (New).

and for each algorithm we used the following stopping criterion \( \|g_{k+1}\| < 1 \times 10^{-5} \).

All the numerical results are summarized in Table (1), Table (2) and Table (3). They present the numbers of iterations (NOI) versus the numbers of function evaluations (NOF) that are needed to obtain the condition \( \|g_{k+1}\| < 1 \times 10^{-5} \) while Table (3) gives the percentage performance of the new algorithm based on both NOI and NOF against the original Zit algorithm.

The important thing is that the new algorithm solves each particular problem measured by NOI and NOF respectively, while the other algorithm may fail in some cases. Moreover, the new proposed algorithm always performs more stably and efficiently.

Namely there are about (50-52)% on NOI for all dimensions also there are (63-78)% improvements on NOF for all test functions.
Table (1). Comparison between the New and Zri algorithms using different values of $12<N<5000$ for 1st test functions

<table>
<thead>
<tr>
<th>N. OF TEST</th>
<th>TEST FUNCTION</th>
<th>Zir N(OF)</th>
<th>New N(OF)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>EX-beal</td>
<td>804 644</td>
<td>855 684</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>956 764</td>
<td>1004 802</td>
</tr>
<tr>
<td></td>
<td>N=20</td>
<td>1073 855</td>
<td>1086 872</td>
</tr>
<tr>
<td></td>
<td>N=5000</td>
<td>137 115</td>
<td>142 128</td>
</tr>
<tr>
<td>2</td>
<td>GEN-edger</td>
<td>53 22</td>
<td>59 24</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>61 27</td>
<td>60 28</td>
</tr>
<tr>
<td></td>
<td>N=20</td>
<td>14 23</td>
<td>27 28</td>
</tr>
<tr>
<td></td>
<td>N=5000</td>
<td>24 21</td>
<td>27 28</td>
</tr>
<tr>
<td>3</td>
<td>Pull Heslan</td>
<td>85 19</td>
<td>116 22</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>183 25</td>
<td>265 32</td>
</tr>
<tr>
<td></td>
<td>N=20</td>
<td>154 16</td>
<td>154 16</td>
</tr>
<tr>
<td></td>
<td>N=5000</td>
<td>25 21</td>
<td>30 27</td>
</tr>
<tr>
<td>4</td>
<td>GEN-Q2</td>
<td>164 162</td>
<td>160 162</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>159 160</td>
<td>159 160</td>
</tr>
<tr>
<td></td>
<td>N=160</td>
<td>159 156</td>
<td>156 156</td>
</tr>
<tr>
<td>5</td>
<td>Digonal4</td>
<td>91 17</td>
<td>97 19</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>109 20</td>
<td>115 21</td>
</tr>
<tr>
<td></td>
<td>N=20</td>
<td>118 22</td>
<td>119 22</td>
</tr>
<tr>
<td>6</td>
<td>GEN-quadratic</td>
<td>243 169</td>
<td>241 235</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>241 235</td>
<td>241 235</td>
</tr>
<tr>
<td></td>
<td>N=20</td>
<td>241 235</td>
<td>241 235</td>
</tr>
<tr>
<td>7</td>
<td>Digonal6</td>
<td>20 17</td>
<td>24 22</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>27 24</td>
<td>26 24</td>
</tr>
<tr>
<td></td>
<td>N=20</td>
<td>15 16</td>
<td>18 19</td>
</tr>
<tr>
<td>8</td>
<td>GEN-Wolf</td>
<td>267 166</td>
<td>238 289</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>365 324</td>
<td>382 328</td>
</tr>
<tr>
<td></td>
<td>N=20</td>
<td>394 332</td>
<td>394 332</td>
</tr>
<tr>
<td>9</td>
<td>GEN-Shallow</td>
<td>763 422</td>
<td>817 499</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>931 506</td>
<td>985 533</td>
</tr>
<tr>
<td></td>
<td>N=20</td>
<td>982 534</td>
<td>988 556</td>
</tr>
<tr>
<td>10</td>
<td>Quadratich</td>
<td>106 30</td>
<td>94 373</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>3043 373</td>
<td>3140 433</td>
</tr>
<tr>
<td></td>
<td>N=20</td>
<td>3280 482</td>
<td>3328 506</td>
</tr>
</tbody>
</table>

Table (2). Comparison between the New and Zir algorithms using different values of $12<N<5000$ for 2nd test functions

<table>
<thead>
<tr>
<th>N. OF TEST</th>
<th>TEST FUNCTION</th>
<th>Zir N(OF)</th>
<th>New N(OF)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GEN-Heloyal</td>
<td>F F F F F</td>
<td>F F F F F</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>70 53</td>
<td>71 57</td>
</tr>
<tr>
<td></td>
<td>N=36</td>
<td>73 59</td>
<td>74 62</td>
</tr>
<tr>
<td>2</td>
<td>Fred</td>
<td>F F F F F</td>
<td>F F F F F</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>139 125</td>
<td>142 136</td>
</tr>
<tr>
<td></td>
<td>N=36</td>
<td>147 144</td>
<td>149 146</td>
</tr>
<tr>
<td>3</td>
<td>liamhick</td>
<td>F F F F F</td>
<td>F F F F F</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>107 95</td>
<td>96 424</td>
</tr>
<tr>
<td></td>
<td>N=36</td>
<td>240 201</td>
<td>234 200</td>
</tr>
<tr>
<td>4</td>
<td>starcase</td>
<td>F F F F F</td>
<td>F F F F F</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>22 17</td>
<td>52 44</td>
</tr>
<tr>
<td></td>
<td>N=36</td>
<td>466 426</td>
<td>4310 3792</td>
</tr>
<tr>
<td>5</td>
<td>TDP</td>
<td>F F F F F</td>
<td>F F F F F</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>144 130</td>
<td>211 183</td>
</tr>
<tr>
<td></td>
<td>N=36</td>
<td>1179 1087</td>
<td>2467 2163</td>
</tr>
<tr>
<td>6</td>
<td>Biggar</td>
<td>F F F F F</td>
<td>F F F F F</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>16 10</td>
<td>32 23</td>
</tr>
<tr>
<td></td>
<td>N=36</td>
<td>232 214</td>
<td>268 214</td>
</tr>
<tr>
<td>7</td>
<td>Mike</td>
<td>F F F F F</td>
<td>F F F F F</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>184 146</td>
<td>194 180</td>
</tr>
<tr>
<td></td>
<td>N=36</td>
<td>269 214</td>
<td>268 214</td>
</tr>
<tr>
<td>8</td>
<td>GEN-Powell</td>
<td>F F F F F</td>
<td>F F F F F</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>196 131</td>
<td>199 141</td>
</tr>
<tr>
<td></td>
<td>N=36</td>
<td>187 145</td>
<td>195 145</td>
</tr>
<tr>
<td>9</td>
<td>EX-Fredrie &amp;</td>
<td>F F F F F</td>
<td>F F F F F</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>196 177</td>
<td>199 191</td>
</tr>
<tr>
<td></td>
<td>N=36</td>
<td>204 201</td>
<td>206 201</td>
</tr>
<tr>
<td>10</td>
<td>TR1</td>
<td>F F F F F</td>
<td>F F F F F</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>74 61</td>
<td>260 230</td>
</tr>
<tr>
<td></td>
<td>N=36</td>
<td>1311 1001</td>
<td>1770 1196</td>
</tr>
<tr>
<td>11</td>
<td>Almost Purified</td>
<td>F F F F F</td>
<td>F F F F F</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>20 16</td>
<td>33 32</td>
</tr>
<tr>
<td></td>
<td>N=36</td>
<td>247 201</td>
<td>470 354</td>
</tr>
<tr>
<td>12</td>
<td>QDP</td>
<td>F F F F F</td>
<td>F F F F F</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>23 13</td>
<td>43 25</td>
</tr>
<tr>
<td></td>
<td>N=36</td>
<td>112 57</td>
<td>171 88</td>
</tr>
<tr>
<td>13</td>
<td>GEN-Centaur</td>
<td>F F F F F</td>
<td>F F F F F</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>61 69</td>
<td>91 75</td>
</tr>
<tr>
<td></td>
<td>N=36</td>
<td>95 79</td>
<td>96 79</td>
</tr>
<tr>
<td>14</td>
<td>singular</td>
<td>F F F F F</td>
<td>F F F F F</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>146 75</td>
<td>224 124</td>
</tr>
<tr>
<td></td>
<td>N=36</td>
<td>385 206</td>
<td>190 161</td>
</tr>
<tr>
<td>15</td>
<td>OSP</td>
<td>F F F F F</td>
<td>F F F F F</td>
</tr>
<tr>
<td></td>
<td>N=12</td>
<td>833 483</td>
<td>2674 1918</td>
</tr>
<tr>
<td></td>
<td>N=36</td>
<td>2720 1964</td>
<td>2842 1981</td>
</tr>
</tbody>
</table>
Table (3). Percentage performance of the New algorithm against Zri algorithm for 100% in both NOI and NOF

<table>
<thead>
<tr>
<th>N</th>
<th>Costs</th>
<th>NEW</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>NOF</td>
<td>63.13</td>
</tr>
<tr>
<td>36</td>
<td>NOF</td>
<td>68.72</td>
</tr>
<tr>
<td>360</td>
<td>NOF</td>
<td>78.82</td>
</tr>
<tr>
<td>1080</td>
<td>NOF</td>
<td>78.59</td>
</tr>
<tr>
<td>4320</td>
<td>NOF</td>
<td>78.12</td>
</tr>
<tr>
<td>5000</td>
<td>NOF</td>
<td>78.05</td>
</tr>
</tbody>
</table>

5. Conclusions:

In this paper, a new PCG-algorithm with a self-scaling VM-update and a new search direction formula is proposed. A modified formula of an inexact line search is implemented to solve a large-scale unconstrained optimization test functions. Our numerical results supports our claim and also indicate that the new algorithm sufficiently decrease the function values and iterations and it needs an extra line search conditions satisfied near the stationary point of the proposed line search procedure.

Appendix 1:

All the test functions used in Table (1) for this paper arc from general literature. See [5]:

1. Generalized Beale Function:

$$f(x) = \sum_{i=1}^{n/2} \left[ 1.5 - x_{2i} + (1 - x_{2i}^2) \right]^2 + \left[ 2.25 - x_{2i-1} (1 - x_{2i}^2) \right]^2 + \left[ 2.625 - x_{2i-1} (1 - x_{2i}^2) \right]^2,$$

$$x_0 = [-1,-1,\ldots,-1].$$

2. Generalized Edger Function:

$$f(x) = \sum_{i=1}^{n/2} (x_{2i-1} - 2)^4 + (x_{2i-1} - 2)^2 x_{2i}^2 + (x_{2i} + 1)^2,$$

$$x_0 = [1,0,\ldots,1,0].$$

3. Full Hessian Function:

$$f(x) = \left( \sum_{i=1}^{n} x_i \right)^2 + \sum_{i=1}^{n} \left( x_i \exp(x_i) - 2x_i - x_i^2 \right),$$

$$x_0 = [1,1,\ldots,1].$$

4. Generalized quadratic Function GQ2:
5. Diagonal 4 Function:
\[ f(x) = \sum_{i=1}^{\sqrt{n}/2} \left( x_{2i-1}^2 + c x_{2i}^2 \right), \]
\[ x_0 = [1,1,1,...,1], \quad c = 100. \]

6. Generalized quadratic Function GQ1
\[ f(x) = \sum_{i=1}^{n} x_i^2 + (x_{i+1} + x_{i-1})^2, \]
\[ x_0 = [1,1,1,...,1]. \]

7. Diagonal 6 Function:
\[ f(x) = \sum_{i=1}^{n} (\exp(x_i) - (1 + x_i)), \]
\[ x_0 = [1,1,1,...,1]. \]

8. Generalized Wolfe Function:
\[ f(x) = (-x_i (3 - x_i / 2) + 2x_2 - 1)^2 + \sum_{i=1}^{n-1} (x_{i+1} - x_i (3 - x_i / 2 + 2x_{i+1} - 1))^2 + (x_{n-1} - x_n (3 - x_n / 2) - 1)^2, \]
\[ x_0 = [-1,-1,-1]. \]

9. Generalized Shallow function:
\[ f(x) = \sum_{i=1}^{n/3} (x_{2i-1}^2 - x_{2i})^2 + (1 - x_{2i-1})^2, \]
\[ x_0 = [-2,-2,-2,...,-2,-2]. \]

10. Quadratic Function QF2:
\[ f(x) = \frac{1}{2} \sum_{i=1}^{n} (x_i^2 - 1)^2 - x_n, \]
\[ x_0 = [0.5,0.5,...,0.5]. \]

**Appendix 2:**

All the test Functions used in Table (2) for this paper are from general literature. See [5]:

1. General Helical Function:
\[ f(x) = \sum_{i=1}^{n/3} \left( 100x_{3i} - 10^6 H_{ij} \right)^2 + 100(R_i - 1)^2 + x_{3i}^2, \]
where \( R_i = \sqrt{x_{3i-2}^2 + x_{3i-1}^2}, H_i = (2\pi)^{-1} \tan^{-1} \frac{x_{3i-1}}{x_{3i-2}} \) if \( x_{3i-2} > 0 \)
\( H_i = 0.5 + (2\pi)^{-1} \tan^{-1} \frac{x_{3i-1}}{x_{3i-2}} \) if \( x_{3i-2} < 0 \)

\( x_0 = [-1,0,0,...,-1,0],0.. \)

2. Extended Fred Function:
\[
f(x) = \sum_{i=1}^{n/2} (-13 + x_{2i-1} + (5 - x_{2i}) + (x_{2i} - 2)(x_{2i}))^2 + \sum_{j=1}^{n/2} (-29 + x_{2j-1} + (1 - x_{2j}) + (x_{2j} - 14)(x_{2j}))^2,
\]
\( x_0 = [1,2,...,n] \)

3. Liarwhd Function (cut):
\[
f(x) = \sum_{i=1}^{n} 4(-x_1 + x_i^2)^2 + \sum_{j=1}^{n} (x_j - 1)^2,
\]
\( x_0 = [4,4,...,4] . \)

4. Staircase2 Function:
\[
f(x) = \sum_{i=1}^{n} \left[ \left( \sum_{j=1}^{i} x_j \right) - i \right]^2,
\]
\( x_0 = [0,0,...,0] . \)

5. Tridiagonal Perturbed Quadratic Function:
\[
f(x) = x_1^2 + \sum_{i=2}^{n-1} i x_i^2 + (x_{i+1} + x_i + x_{i+1})^2,
\]
\( x_0 = [0.5,0.5,...,0.5] . \)

6. Biggsbl Function (CUTE):
\[
f(x) = (x_1 - 1)^2 + \sum_{i=1}^{n/2} (x_{i+1} - x_i)^2 + (1 - x_n)^2,
\]
\( x_0 = [1,1,...,1] . \)

7. Mill and Cornwell function:
\[
f(x) = \sum_{i=1}^{n/4} \left[ \exp(x_{4i+3} + 10x_{4i-2}) + 5(x_{4i+1} - x_{4i})^2 (x_{4i+2} - 2x_{4i-1})^4 + 10(x_{4i-3} - x_{4i})^4 \right],
\]
\( x_0 = [1,2,2,2,...,1,2,2,2] . \)

8. Generalized Powell function:
\[
f(x) = \sum_{i=1}^{n/2} \left( 3 - \left( \frac{1}{\sqrt{1+(x_i - x_{i+1})^2}} \right) - \sin \left( \frac{\sqrt{x_i+x_{i+1}}}{2} \right) - \exp \left[ \left( \frac{\sqrt{x_i+x_{i+1}}}{2} - 2 \right)^2 \right] \right),
\]
\( x_0 = [0,1,2,...,0,1,2] . \)

9. Extended Freudenstein & Roth Function:
\[ f(x) = \sum_{i=1}^{\frac{n}{2}} \left( -13 + x_{2i-1} + ((5 - x_{2i})x_{2i} - 2)x_{2i} \right)^2 + \left( -29 + x_{2i-1} + ((x_{2i} + 1)x_{2i} - 14)x_{2i} \right)^2, \]
\[ x_0 = [0.5, -2.0, 0.5, -2, ..., 0.5, -2]. \]

10. Extended Tridiagonal-1 Function:
\[ f(X) = \sum_{i=1}^{\frac{n}{2}} (x_{2i-1} + x_{2i} - 3)^2 + (x_{2i-1} - x_{2i} + 1)^4, \]
\[ x_0 = [2, 2, ..., 2]. \]

11. Almost Perturbed Quadratic Function:
\[ f(x) = \sum_{i=1}^{n} i x_i^2 + \frac{1}{100} (x_1 + x_n)^2, \]
\[ x_0 = [0.5, 0.5, ..., 0.5]. \]

12. Quadratic Diagonal Perturbed Function:
\[ f(x) = \left( \sum_{i=1}^{n} x_i \right)^2 + \sum_{i=1}^{n} \frac{1}{100} x_i^2, \]
\[ x_0 = [0.5, 0.5, ..., 0.5]. \]

13. Generalized Cant real Function:
\[ f(x) = \sum_{i=1}^{\frac{n}{4}} \left[ \exp(x_{4i} - 3) - x_{4i-2} \right]^4 + 100(x_{4i-2} - x_{4i-1})^6 + (\arctan(x_{4i-1} - x_{4i}))^4 + x_{4i-3}, \]
\[ x_0 = [1.2, 2.2, 2, ..., 1.2, 2.2]. \]

14. Sinquad Function (CUTE):
\[ f(x) = (x_i - 1)^4 + \sum_{i=1}^{\frac{n}{2}} \left( \sin(x_i - x_n) - x_i^2 + x_i^2 \right)^2 + (x_n^2 - x_i^2)^2, \]
\[ x_0 = [0.1, 0.1, ..., 0.1]. \]

15. Generalized OSP (Oren and Spedicato) Function:
\[ f(x) = \left\lfloor \sum_{i=1}^{n} i x_i^2 \right\rfloor^2, \]
\[ x_0 = [1, ..., 1]. \]
REFERENCES


