Original Research Paper

A New Spectral Conjugate Gradient Method for Nonlinear Unconstrained Optimization

[1]Ibtisam A. Masmali, [2*]Zabidin Salleh and [3]Ahmad Alhawarat

[1]*Department of Mathematics, Faculty of Science, Jazan University, Jazan, Saudi Arabia*
[2]*Department of Mathematics, Faculty of Ocean Engineering Technology and Informatics, University Malaysia Terengganu, 21030 Kuala Nerus, Terengganu, Malaysia*
[3]*Department of Mathematics, Faculty of Ocean Engineering Technology and Informatics, University Malaysia Terengganu, 21030 Kuala Nerus, Terengganu, Malaysia*

Article history
Received: 13-03-2021
Revised: 17-06-2021
Accepted: 25-06-2021

Corresponding Author:
Zabidin Salleh
Department of Mathematics, Faculty of Ocean Engineering Technology and Informatics, Universiti Malaysia Terengganu, 21030 Kuala Nerus, Terengganu, Malaysia
Email: zabidin@umt.edu.my

Abstract: The conjugate gradient method is widely used to solve large scale unconstrained optimization problems. However, the rate of convergence conjugate gradient method is linear unless it restarted. In this study, we present a new spectral conjugate gradient modification formula with restart property obtains the global convergence and descent properties. In addition, we proposed a new restart condition for Fletcher-Reeves conjugate gradient formula. The numerical results demonstrated that the modified Fletcher-Reeves parameter and the new CG formula with their restart conditions are more efficient and robustness than other conventional methods.

Keywords: Conjugate Gradient, Global Convergence, Descent Condition

Introduction

We consider the following problem:

$$\min f(x), x \in R^n, \tag{1}$$

where, f: $R^n \to R$ is continuous and differentiable function and its gradient $g(x) = \Delta f(x)$ is available. Iterative methods are usually used to solve (1), as follows:

$$x_{k+1} = x_k + \alpha_k d_k, k = 1, 2, ..., \tag{2}$$

starting from initial point $x_1 \in R^n$, where α_k is obtained by some line search. The search direction d_k is defined by:

$$d_k = \begin{cases} -g_k, k = 1, \\ -g_k + \beta_k d_{k-1}, \ k \ge 2, \end{cases} \tag{3}$$

where, $g_k = g(x_k)$ and β_k is known as the conjugate gradient parameter.

The exact line search can be used to find the steplength α_k. Suppose that $\phi(\alpha) = f(x_k+\alpha d_k)$ which is problem that departs from x_k to find a step length in the direction d_k such that $\phi(\alpha) < \phi(0)$. If the step length is defined such that the search direction minimized i.e., this line search is called exact line search where this line search is expensive. Therefore, using the inexact line search with less computation load is better. The inexact line search in particular Strong Wolfe-Powell (SWP) line search inherits the advantages of exact line search and computationally inexpensive. Thus, to reduce the computation cost of exact line search and also to reduce evaluations of the objective function and gradient function, usually the inexact line search is employed. SWP line search is more preferable than other line searches. The SWP line search is defined by:

$$f(x_k + \alpha_k d_k) = \min f(x_k + \alpha d_k), \ \alpha \ge 0, \tag{4}$$

$$f(x_k + \alpha_k d_k) \le f(x_k) + \delta \alpha_k g_k^T d_k \tag{5}$$

and:

$$\left| g(x_k + \alpha_k d_k)^T d_k \right| \le \sigma \left| g_k^T d_k \right| \tag{6}$$

where, $0 < \delta < \sigma < 1$. The Weak Wolfe-Powell (WWP) (Wolfe, 1969; 1971) line search given by (5) and:

$$g(x_k + \alpha_k d_k)^T d_k \ge \sigma \, g_k^T d_k. \tag{7}$$

The convergence of CG method will not be linear if we restart CG method (Powell, 1977). Beale (1972) recommended the use of the two-term CG method instead ($d_k = -g_k$, $\forall k \geq 1$) as the restart search direction. Powell (1984) recommended restarting d_k using Beale's method if:

$$\left|g_k^T g_{k-1}\right| > 0.2 \,\|g_k\|^2, \tag{8}$$

Dai and Yuan (1998) present the following restart criterion:

$$\left|g_k^T g_{k-1}\right| > \tau \,\|g_k\|^2, \tau \in (0,\ 1). \tag{9}$$

The famous formulas for β_k are the Hestenes-Stiefel (HS) (Hestenes and Stiefel, 1952) , Fletcher-Reeves (FR) (Fletcher and Reeves, 1964) and Polak-Ribière-Polyak (PRP) (Polak and Ribiere, 1969) formulas, which are defined as follows:

$$\beta_k^{HS} = \frac{g_k^T y_k}{d_{k-1}^T y_k}, \beta_k^{FR} = \frac{g_k^T g_k}{\|g_{k-1}\|^2}, \beta_k^{PRP} = \frac{g_k^T y_k}{\|g_{k-1}\|^2},$$

where, $y_k = g_k - g_{k-1}$.

Polak and Ribière (1969) proved CG method with the PRP formula and by using exact line search is convergent. Powell (1984) show that the PRP fail to satisfy the convergence by using an example even the exact line is used. Powell recommended to use the non-negative of PRP formula to satisfy the convergence analysis. Gilbert and Nocedal (1992) suggest to use PRP as follows:

$$\beta_k^{PRP+} = \max\left\{0,\ \beta_k^{PRP}\right\}$$

Zoutendijk (1970) obtain the global convergence of FR formula with CG method and the exact line search. Al-Baali (1985) proved FR method with SWP line search when $\sigma < 1/2$ and SWP line search is employed, Guanghui *et al.* (1995) extended the proof to the case for $\sigma \leq 1/2$.

Alhawarat *et al.* (2017) presented the following formula:

$$\beta_k^{AZPRP} = \begin{cases} \dfrac{\|g_k\|^2 - \mu_k\left|g_k^T g_{k-1}\right|}{\|g_{k-1}\|^2}, \text{if} \quad \|g_k\|^2 > \mu_k\left|g_k^T g_{k-1}\right|, \\ 0, \text{ otherwise,} \end{cases} \tag{10}$$

where, μ_k is defined as follows:

$$\mu_k = \frac{\|x_k - x_{k-1}\|}{\|y_k\|}$$

Kaelo *et al.* (2020) proposed the following CG formula:

$$\beta_k^{PKT} = \begin{cases} \dfrac{\|g_k\|^2 - g_k^T g_{k-1}}{\max\left\{d_{k-1}^T y_{k-1}, -g_{k-1}^T d_{k-1}\right\}}, \text{ If } 0 < g_k^T g_{k-1} < \|g_k\|^2 \\ \dfrac{\|g_k\|^2}{\max\left\{d_{k-1}^T y_{k-1}, -g_{k-1}^T d_{k-1}\right\}}, \text{ otherwise.} \end{cases}$$

As we know that in the case of the function is quadratic i.e., $f(x) = g^T x + (1/2)x^T Hx$ and the step size obtained by exact line search (3), the CG method satisfy the conjugacy condition i.e., $d_i^T H d_j^T = 0, \forall i \neq j$. By using the mean value theorem and exact line search with Eq. (2) we can obtain β_k^{HS}. From quasi-Newton method, BFGS method and the limited memory (LBFGS) method and using (2), Dai and Liao (2001) present the following conjugacy condition:

$$d_k^T y_{k-1} = -t g_k^T s_{k-1},$$

where, $S_{k-1} = x_k - x_{k-1}$ and $t \geq 0$. In the case of $t = 0$ Eq. (8) becomes the classical conjugacy condition. By using (2) and (8), (Kaelo *et al.*, 2020) proposed the following CG formula:

$$\beta_k^{DL} = \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}} - t\frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}.$$

However, β_k^{DL} face the same problem as β_k^{PRP} and β_k^{HS} i.e., β_k^{DL} is not non-negative in general. Thus (Dai and Liao, 2001) replaced Eq. (9) by:

$$\beta_k^{DL+} = \max\left\{\beta_k^{HS}, 0\right\} - t\frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}.$$

Hager and Zhang (2005; 2013) presented a modified CG parameter that satisfies the descent property for any inexact line search with $g_k^T d_k \leq -(7/8)\|g_k\|^2$. This formula is given as follows:

$$\beta_k^{HZ} = \max\left\{\beta_k^N,\ \eta_k\right\}$$

where, $\beta_k^N = \dfrac{1}{d_k^T y_k}\left(y_k - 2d_k\dfrac{\|y_k\|^2}{d_k^T y_k}\right)^T g_k$,

$\eta_k = -\dfrac{1}{\|d_k\| \min\{\eta,\ \|g_k\|\}}$ and $\eta > 0$ is a constant.

Notes that if $t = 2\dfrac{\|y_k\|^2}{s_k^T y_k}$ then $\beta_k^N = \beta_k^{DY}$.

The positive scalar denoted by θ_k. Hence, d_k given as:

$$d_k = -\theta_k g_k + \beta_k d_{k-1}.$$

when, θ_k = 1, the search direction is a classical CG method. If β_k = 0, then there are two possibilities of θ_k. If $\theta_k = \Delta^2 f(x_k)^{-1}$ or an approximation of it, then the search direction is Newton or Quasi-Newton, respectively.

The New Formula and the Algorithm

Here, we construct the following new modification to improve the efficiency and robustness of DY CG formula and robustness of PRP CG method.

For $k = 1$, $dk = -g_k$.

For $k \geq 2$:

$$d_k^{ATAZ} = \begin{cases} -\theta_k g_k + \beta_k^{DY} d_{k-1}, if\ g_k^T d_{k-1} \geq 0, \\ -g_k + \beta_k^{PRP+} d_{k-1}, else. \end{cases} \quad (11)$$

where, ||.||means the Euclidean norm and $\theta_k = 1 + \frac{g_k^T d_{k-1}}{g_{k-1}^T d_{k-1}}$ and:

$$\beta_k^{DY} = \frac{\|g_k\|^2}{d_{k-1}^T (g_k - g_{k-1})}.$$

Algorithm 1

Step 1 Provide a starting point x_1. Set the initial search direction $d_1 = -g_1$. Let k: = 1.
Step 2 If a stopping criteria is satisfied, then stop.
Step 3 Compute d_k based on (2) with (11).**Error! Reference source not found.**
Step 4 Compute α_k using (4) and (6).
Step 5 Update x_{k+1} based on (1).
Step 6 Set k: = k+1 and go to Step 2.

In Algorithm1, note that after the step $k = k+1$, the iterates $x_k = x_{k+1}$ takes place after every iteration. The other iterations are updated in a similar manner as x_k.

In following section, we present the global convergence property of the new formula (11). In case of $d_k^{ATRZ} = 0$, then the search direction becomes the steepest descent (negative gradient) which mean the stationary point is obtained.

Convergence of CG Algorithm with the Search Direction d_k^{ATAZ}

Assumption 1

A. The level set $\Omega = \{x | f(x) \leq f(x_1)\}$ is bounded, that is, a positive constant M exists such that

$$\|x\| \leq M, \forall x \in \Omega.$$

B. In some neighbourhood n of Ω, f is continuously differentiable and its gradient is Lipschitz continuous; that is, for all $x, y \in N$, there exists a constant such that:

$$\|g(x) - g(y)\| \leq L\|x - y\|.$$

This assumption implies that there exists a positive constant B such that:

$$\|g(u)\| \leq B, \forall u \in N.$$

The descent condition:

$$g_k^T d_k \leq -\|g_k\|^2, \forall k \geq 1, \quad (12)$$

Al-Baali (1985) modified (12) to the following form and used it to prove the FR method:

$$g_k^T d_k \leq -c\|g_k\|^2, \forall k \geq 1, \quad (13)$$

where, $c \in (0,1)$. Equation (13) is the sufficient descent condition. Note that the general form of the sufficient descent condition is (14) with $c > 0$.

Descent and Convergence Properties for d_k^{ATAZ} with the SWP Line Search

In fact, we have two types of global convergence; weak global convergence and strong global convergence both of them imply the stationary point for optimization problem. However, the convergence and the descent properties will not give any sense in terms of the efficiency for CG methods; for example, FR formula has global convergence properties with poor efficiency. Thus, to improve the efficiency when the method cycle does not reach a solution the CG algorithm should be restarted. In the following section, we will present a new CG method with restart property by using the steepest descent method.

The following lemma is called Zoutendijk condition (Al-Baali, 1985).

Lemma 3.1

Suppose assumption 1 is holds. Suppose method in the form (1), (2) and α_k satisfies the WWP line search (5) and (6), where the search direction satisfied. Then

$$\sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} < \infty. \quad (14)$$

Also we can extended Eq. (14) to the following form:

$$\sum_{k=0}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} < \infty. \quad (15)$$

Kaelo *et al*. (2020) present the following theorem for global convergence properties:

Theorem 3.1

Let assumption 1 holds. Suppose any CG method in the form (1) and (2), where d_k is a descent direction and α_k is obtained by the SWP line search. If:

$$\sum_{k\geq 1}^{\infty}\frac{1}{\|d_k\|^2}=\infty,$$

then:

$$\lim_{k\to\infty}\inf\|g_k\| = 0.$$

Theorem 3.2

Let the sequence $\{g_k\}$and $\{d_k\}$are generated by the methods (2), (3) and (11), then (13) holds.

Proof. By using proof by induction. From (3) for k = 1, $g_1^T d_1 = -\|g_1\|^2$. Suppose that it is true until k - 1, i.e., $g_{i-1}^T d_{i-1} < 0$, for i = 1,2,…, k - 1 then we have the following two cases:

Case 1 $g_k^T d_{k-1} \geq 0$:

$$d_k = -\frac{g_k^T d_{k-1}}{g_{k-1}^T d_{k-1}} g_k - g_k + \frac{\|g_k\|^2}{d_{k-1}(g_k - g_{k-1})} d_{k-1},$$

Multiply both sides by g_k^T:

$$g_k^T d_k = -\frac{g_k^T d_{k-1}}{d_{k-1}(g_k - g_{k-1})}\|g_k\|^2 - \|g_k\|^2 + \frac{\|g_k\|^2}{d_{k-1}(g_k - g_{k-1})} g_k^T d_{k-1},$$

Since $g_k^T d_{k-1} \geq 0$:

$$g_k^T d_k = -\|g_k\|^2$$

Case 2 $g_k^T d_{k-1} < 0$:

$$d_k = -g_k + \beta_k^{PRP+} d_{k-1},$$

Multiply both sides by g_k^T:

$$g_k^T d_k = -\|g_k\|^2 + \beta_k^{PRP+} g_k^T d_{k-1},$$

By using $g_k^T d_{k-1} < 0$ and $\beta_k^{PRP+} \geq 0$, we obtain $g_k^T d_k < 0$.

Theorem 3.3

Let assumption 1 holds. Assume $\{g_k\}$and $\{d_k\}$are obtained by algorithm 1 in which α_k is obtained by the WWP line search and (13) holds. Then the $\lim_{k\to\infty}\inf\|g_k\| = 0$.

$$d_k = -g_k + \frac{g_k^T d_{k-1}}{g_{k-1}^T d_{k-1}} g_k + \frac{\|g_k\|^2}{d_{k-1}(g_k - g_{k-1})} d_{k-1},$$

$$\|d_k\| = \|\theta_k\|\|g\|_k + \frac{\|g_k\|^2}{\|d_{k-1}\|\|(g_k - g_{k-1})\|}\|d_{k-1}\|$$

$$\|d_k\| = \|\theta_k\|\|g\|_k + \frac{\|g_k\|^2}{\|(g_k - g_{k-1})\|}$$

since $\|\theta_k\| \leq 1-\sigma$

$$\|d_k\| = \|g\|_k - \sigma\|g\|_k + \frac{\|g_k\|^2}{\|(g_k - g_{k-1})\|}$$

By using assumption 1:

$$\|d_k\| = 1 - \sigma\gamma + \frac{\gamma^2}{\lambda}\, where\, \lambda > 0.$$

Let: $M = 1 - \sigma\gamma + \frac{\gamma^2}{\lambda}$ $\|d_k\| \leq M$.

By using Theorem 3.1 We obtain the $\lim_{k\to\infty}\inf\ \|g_k\| = 0$.

The New Restart Criteria for FR Family

Fletcher-Reeves formula is simple CG method and has a global convergence property with SWP line search and it satisfies the descent property. However, FR formula is not efficient as β_k^{PRP} which the later has a problem in convergence properties for some optimization functions. Powell studied β_k^{FR} formula and show that this method cycle does not reach a solution when $x_{k+1} \approx x_k$ which implies that $\|g_k\|/\|g_{k-1}\| \approx 1$. To solve this problem we suggest restarting β_k^{FR} as follows:

$$\beta_k^{FR*} = \begin{cases} 0, & 0.9 \leq \frac{\|g_k\|}{\|g_{k-1}\|} \leq 1.1, \\ \beta_k^{FR}, & otherwise. \end{cases} \quad (16)$$

It is clear that when β_k^{FR*} will restart when the $\beta_k^{FR} \approx 1$.

Numerical Results and Discussion

To study the efficiency of the new search direction, we selected several test problems in Table 1 from Cuter (Bongartz *et al*., 1995) and Andrei (2008). The test functions consist of unimodal and multimodal functions. We also selected examples according to the similarities in significant physical properties and shapes. For example, the Rosenbrock function has a long, narrow shape; the Himmelblau function, the six-hump function and the three-hump function have many local minima; the Booth function is plate shaped; and the Sum Squares function is bowl shaped. As the CG method is useful for small-and large-scale optimisation problems, we also select the dimensions of the functions, which varied from 2 to 10000. All of the functions are nonlinear. In Table 1, “Gen” denotes generalised, “Ext” denotes extended, “Dim” denotes dimension/s.

We employed the MATLAB programming environment (ver. 7.9). The results are shown in Fig. 1 and 2, in which a performance measure introduced by Dolan and More (2002) was employed.

The comparison include PRP+, FR, d_k^{ATAZ} and FR* methods. $\|g_k\| \leq 10^{-6}$ is used as the stopping criteria for all algorithms. To obtain the step length we used strong Wolfe-Powell line search with $\delta = 0.01$ and $\sigma = 0.1$.

Since we are interested to find the stationary point/s for optimization problems, we selected more than one initial point to test every function in Table 1 the dimension of functions between 2 and 5000. Different initial points almost will obtain different stationary points, which imply that more than one solution for multimodal functions. Hence, we obtain the best solution. In addition, we select small and large dimensions for every function. The ranges of dimensions are chosen between 2 and 10000. Thus, we conclude that using different dimension and different initial points to obtain the results will be more convince than using original initials and one dimension. However, the starting point needs more study.

Figure 1 and 2 show that the curve of the new formula $\left(\beta_k^{ATAZ}\right)$ is uppermost of all curves. In addition, it is clear that FR* formula is better than original FR formula which demonstrates the discussion that presented by Powell and the program is terminated by the user when the number of iterations exceeds 1000. The PRP+ formula is efficient since its curve started uppermost other curves. However, it is not satisfied the descent property with SWP line search. Thus, the program is terminated automatically.

In addition we present the following two functions the first one called Extended Beale function which given by the following formula:

$$f(x) = \sum_{i=1}^{n/2} \left(1.5 - x_{2i-1}\left(1 - x_{2i}\right)\right)^2 + \left(2.25 - x_{2i-1}\left(1 - x_{2i}^2\right)\right)^2 + \left(2.625 - x_{2i-1}\left(1 - x_{2i}^3\right)\right)^2$$

Number of variables (n): 500, 1000, 5000, 10000 with initial points: (-1,-1, ..., -1), (.5, .5, …, .5), (1, 1,...,1), (2, 2, ..., 2).

This function has only one global minimum surrounded by a flat plateau. At the four corners lie four ascending steep walls that become smaller at the tip. These steep walls become higher as the value of the two variables increases. The minimum is $x^* = (3,0.5)$ and the function value is $f(x^*) = 0$ for two variable functions (note Fig. 3 for a three-dimensional graph).

Table 1: A list of test functions

Function	Initial points
Ext. White & Holst function,	(-1.2,1,-1.2,1…), (5,5,…,5), (10,10,…,10), (15,15,…15)
Ext. Rosenbrock function,	(-1.2,1,-1.2,1…), (5,5,…,5), (10,10,…,10),(15,15,…15)
Six hump function	(1,1), (5,5), (10,10),(15,15)
Ex. Beale function,	(-1,-1,…,-1), (.5,.5,…,.5), (1,1…,1),(2,2,…,5), (5,5,…,5)
Three hump function	(1,1), (5,5), (10,10),(15,15)
Ext. Himmelblau function	(1,1,…,1), (5,5,…,5), (10,10…,10),(15,15,…,15)
Diagonal 2 function	(0.2,0.2,…,0.2),(0.25,0.25,…,0.25), (0.5,…,0.5), (1,1,…,1)
NONSCOMP function	(1,1,…,1), (-1,-1,…,-1), (-2,-2…,-2),(-5,-5,…,-5)
Ext. DENSCHNB function	(1,1,…,1), (5,5,…,5), (10,10…,10),(15,15,…,15)
Shallow function	(-2,-2,…,-2), (2,2,…,2), (5,5…,5), (10,10,…,10)
Booth function	(1,1), (5,5), (10,10),(15,15)
DIXMAANA function, [26]	(2,2,…,.2), (5,5,…,5), (10,10…,10), (15,15,…,15)
DIXMAANB function	(-2,-2,…,-2), (-1,-1,…,-1), (0,0…,0), (1,1,…,1)
NONDIA function	(-2,-2,…,-2), (-1,-1,…,-1), (0,0…,0), (1,1,…,1)
Ext. Tridiagonal 1 function	(1,1,…,1), (5,5,…,5), (10,10…,10),(15,15,…,15)
DQDRTIC function	(-1,-1,…,-1), (1,1,…,1), (2,2…,2), (3,3,…,3)
Diagonal 4 function	(1,1,…,1), (5,5,…,5), (10,10…,10),(15,15,…,15)
Raydan 2 function	(1,1,…,1), (5,5,…,5), (10,10…,10),(15,15,…,15)
Ext. DENSCHNB function	(1,1,…,1), (5,5,…,5), (10,10…,10),(15,15,…,15)
A Quadratic function QF2	(1,1,…,1), (5,5,…,5), (10,10…,10),(15,15,…,15)
Zettl function	(1,1,…,1), (5,5,…,5), (10,10…,10),(15,15,…,15)
Extended Cliff	(1,1,…,1), (5,5,…,5), (10,10…,10),(15,15,…,15)
Ext. Powell function	(1,1,…,1), (5,5,…,5), (10,10…,10),(15,15,…,15)
Generalized Quartic GQ1 function	(1,1,…,1), (5,5,…,5), (10,10…,10),(15,15,…,15)
Ext. Block Diagonal BD1 function	(1,1,…,1), (5,5,…,5), (10,10…,10),(15,15,…,15)

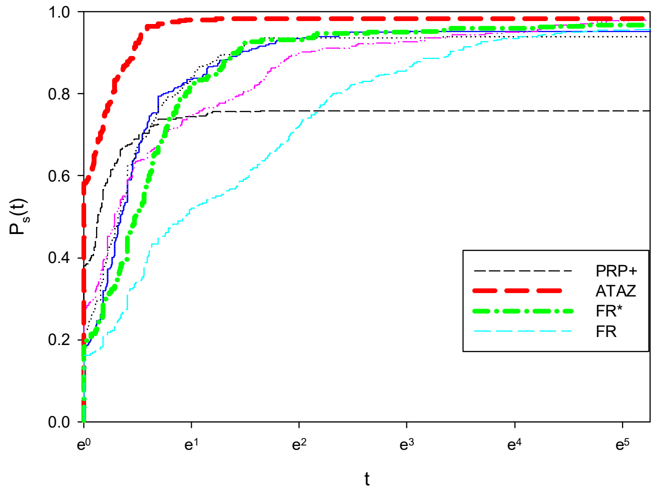

Fig. 1: Performance profile based on the number of iteration

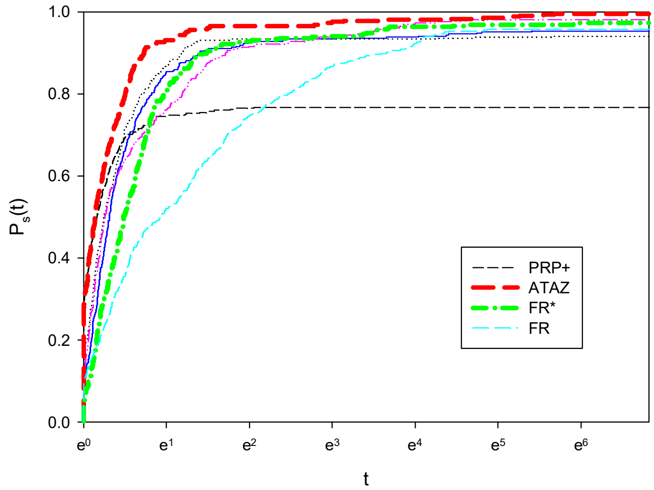

Fig. 2: Performance profile based on the CPU time

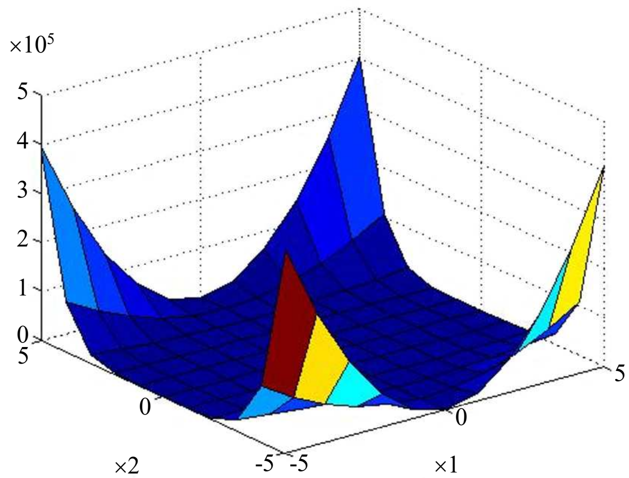

Fig. 3: Extended Beale function in 3D

And the second function called

Perturbed quadratic function:

$$f(x)=\left(\sum_{i=1}^{n}x_i\right)^2+\sum_{i=1}^{n}\frac{i}{100}x_i^2$$

Initial points: (0.5, 0.5, …, 0.5)

This function has a smooth curve, look like dish shape and has a minimum value at x^* = (0,0) and the function value is $f(x^*) = 0$. where lies at the bottom for two variable functions (Fig. 4 for a three-dimensional graph).

Moreover we present another strong comparison between ATAZ and CG-Descent is given with benchmark functions in Table 2. The numerical results in Fig. 5, 6 and 7 show that the new modification ATAZ is better than CG-Descent in term of number of iterations, number of function evaluations and CPU time. The test functions can be downloaded from (Bongartz *et al*., 1995).

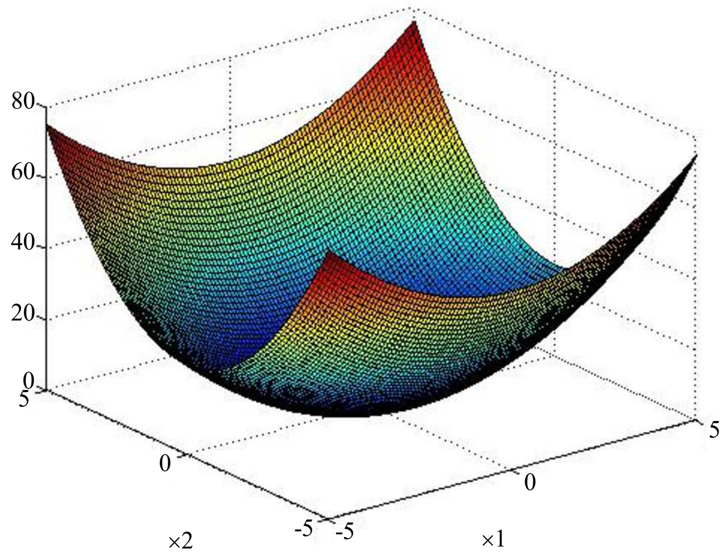

Fig. 4: Perturbed Quadratic function in 3D

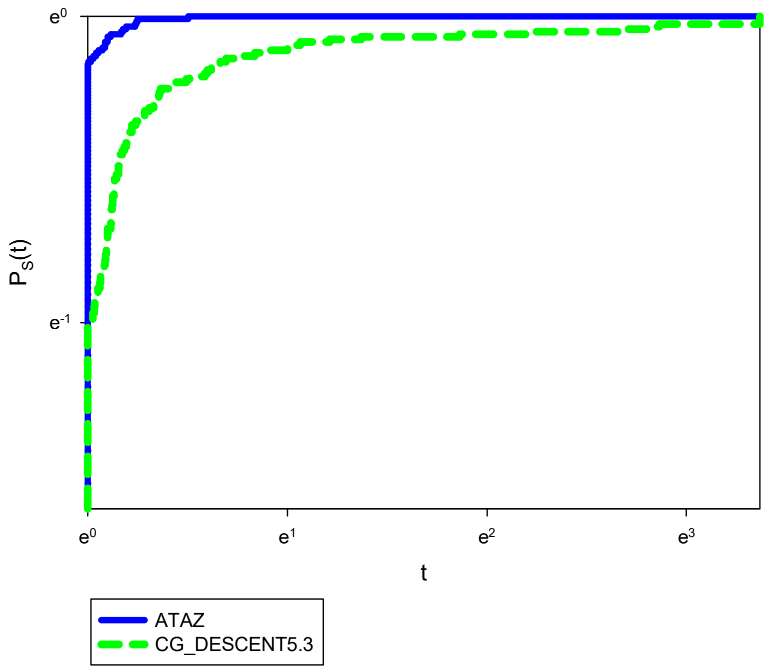

Fig. 5: Performance profile based on the number of iteration

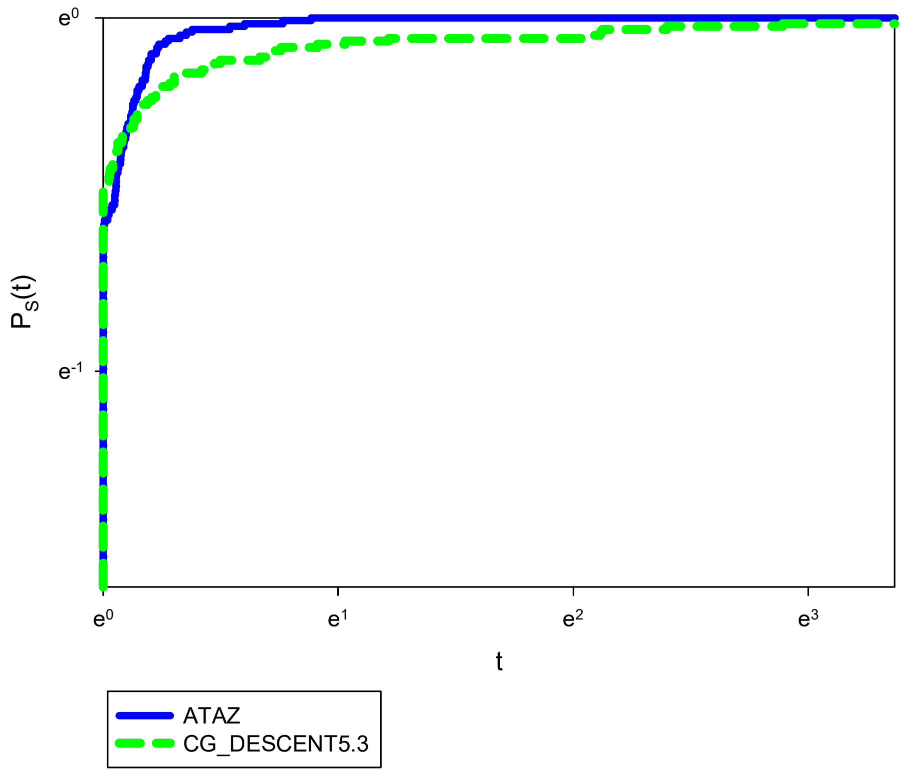

Fig. 6: Performance profile based on the function evaluation

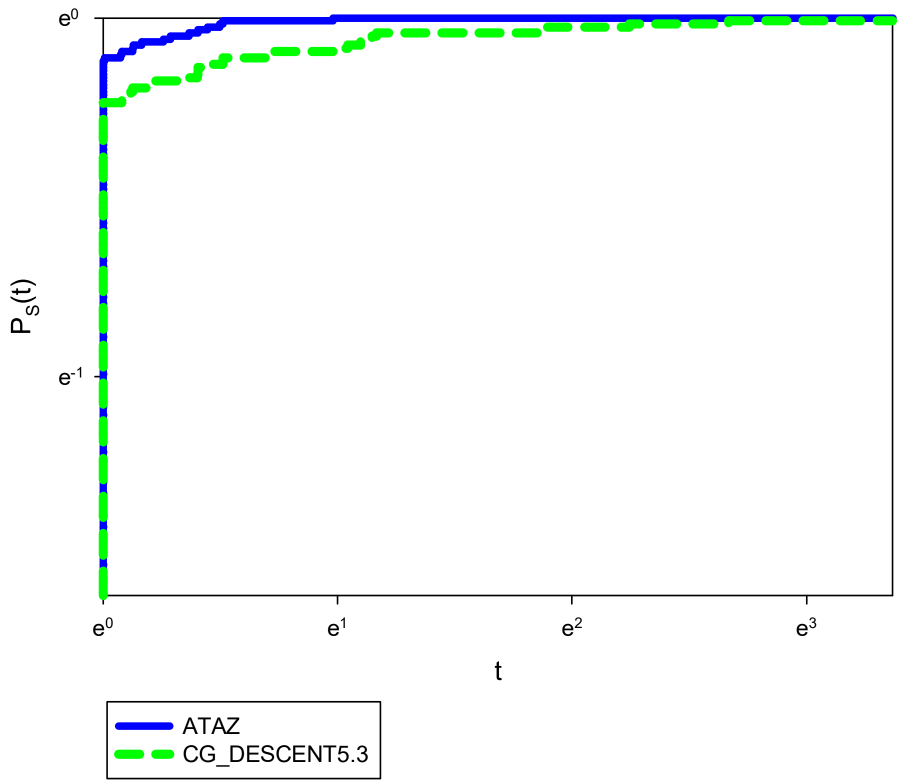

Fig. 7: Performance profile based on CPU time

Table 2: A list of problem functions

		ATAZ			CG-Descent		
function	Dimension	No. iteration	No. function evaluation	CPU time	No. iteration	No. function evaluation	CPU time
AKIVA	2	8	20	0.02	10	21	0.02
ALLINITU	4	9	25	0.02	12	29	0.02
ARGLINA	200	1	3	0.02	1	3	0.02
ARWHEAD	200	6	17	0.02	7	15	0.02
BARD	3	12	32	0.02	16	33	0.02
BEALE	2	11	33	0.02	15	31	0.02
BIGGS6	6	24	64	0.02	27	57	0.02
BOX3	3	10	23	0.02	11	24	0.02
BRKMCC	2	5	11	0.02	5	11	0.02
BROWNAL	200	9	22	0.02	9	25	0.02
BROWNBS	2	10	24	0.02	13	26	0.02
BROWNDEN	4	16	38	0.02	16	31	0.02
CHAINWOO	4000	352	682	0.8	318	619	0.866
CHNROSNB	50	269	549	0.02	287	564	0.02
CLIFF	2	10	46	0.02	18	70	0.02
CUBE	2	17	48	0.02	32	77	0.02
CURLY10	10000	52849	72728	197	47808	67294	173.7
CURLY20	10000	79446	102981	437	66587	89245	383.94
CURLY30	10000	81281	104558	644	79030	102516	639.63
DECONVU	63	390	806	2.00E-02	400	801	2.00E-02
DENSCHNA	2	6	16	0.02	9	19	0.02
DENSCHNB	2	6	18	0.02	7	15	0.02
DENSCHNC	2	11	36	0.02	12	26	0.02
DENSCHND	3	14	46	0.02	47	98	0.02
DENSCHNE	3	12	43	0.02	18	49	0.02
DENSCHNF	2	9	31	0.02	8	17	0.02
DIXMAANA	3000	6	16	0.02	7	15	0.02
DIXMAANB	3000	6	15	0.02	6	13	0.02
DIXMAANC	3000	6	14	0.02	6	13	0.02
DIXMAAND	3000	6	15	0.02	7	15	0.02
DIXMAANE	3000	13	31	0.23	222	239	0.33
DIXMAANF	3000	17	39	0.02	161	323	0.13
DIXMAANH	3000	44	103	0.08	173	347	0.22
DIXMAANJ	3000	415	494	0.52	327	655	0.36
DIXON3DQ	10000	10000	10007	19.12	10000	10007	19.12
DJTL	2	75	1163	0.02	82	917	0.02
DQDRTIC	5000	5	11	0.02	5	11	0.02
DQRTIC	5000	15	32	0.01	17	37	0.03
EDENSCH	2000	26	59	0.08	26	52	0.03
EG2	1000	3	13	0.02	5	11	0.02
EIGENALS	2550	9111	16048	164	10083	18020	178.36
ENGVAL1	5000	24	47	0.08	27	50	0.06
ENGVAL2	3	26	73	0.02	26	61	0.02
EXPFIT	2	9	29	0.02	13	29	0.02
EXTROSNB	1000	23	71	0.8	3808	7759	1.25
FLETCBV2	5000	1	1	0.02	1	1	0.02
FLETCHCR	1000	252	497	0.03	152	290	0.05
FMINSRF2	5625	23	83	1.70E-01	346	693	1.09E+00
FMINSURF	5625	27	86	0.16	473	947	1.51
GENHUMPS	5000	5740	15736	26	6475	12964	20.11
GENROSE	500	1376	2820	0.28	1078	2167	0.17
GROWTHLS	3	109	431	0.02	156	456	0.02
GULF	3	33	95	0.02	37	84	0.02
HAIRY	2	17	82	0.02	36	99	0.02
HATFLDD	3	17	49	0.02	20	43	0.02
HATFLDE	3	13	37	0.02	30	72	0.02
HATFLDFL	3	21	68	0.02	39	92	0.02
HEART6LS	6	375	1137	0.02	684	1576	0.02
HEART8LS	8	253	657	0.02	249	524	0.02
HELIX	3	23	60	0.02	23	49	0.02
HIELOW	3	13	30	0.03	14	30	0.02
HILBERTA	2	2	5	0.02	2	5	0.02
HILBERTB	10	4	9	0.02	4	9	0.02

DOI: 10.3844/jcssp.2021.598.609

Table 2: Continue

HIMMELBB	2	4	18	0.02	10	28	0.02
HIMMELBF	4	23	59	0.02	26	60	0.02
HIMMELBG	2	7	22	0.02	8	20	0.02
HIMMELBH	2	5	13	0.02	7	16	0.02
HUMPS	2	45	223	0.02	52	186	0.02
JENSMP	2	12	47	0.02	15	33	0.02
KOWOSB	4	16	46	0.02	17	39	0.02
LIARWHD	5000	19	50	0.05	21	45	0.03
LOGHAIRY	2	26	196	0.02	27	81	0.02
MANCINO	100	10	21	0.08	11	23	0.08
MARATOSB	2	589	2885	0.02	1145	3657	0.02
MEXHAT	2	14	59	0.02	20	56	0.02
MOREBV	5000	25	102	0.13	161	168	0.41
NCB20B	500	2044	2288	31	2035	4694	46.36
NCB20	5010	303	747	4.17	879	1511	11.83
NONCVXU2	5000	43	81	0.17	6610	12833	15.89
NONDIA	5000	7	25	0.02	7	25	0.03
NONDQUAR	5000	66	216	0.17	1942	3888	2.45
OSBORNEA	5	82	230	0.02	94	213	0.02
OSBORNEB	11	57	134	0.02	62	127	0.02
OSCIPATH	10	295029	781729	2.24	310990	670953	2.08
PALMER1C	8	12	27	0.02	11	26	0.02
PALMER1D	7	10	24	0.02	11	25	0.02
PALMER2C	8	11	21	0.02	11	21	0.02
PALMER3C	8	11	21	0.02	11	20	0.02
PALMER4C	8	11	21	0.02	11	20	0.02
PALMER5C	6	6	13	0.02	6	13	0.02
PALMER6C	8	11	24	0.02	11	24	0.02
PALMER7C	8	11	20	0.02	11	20	0.02
PALMER8C	8	11	19	0.02	11	18	0.02
PENALTY1	1000	14	51	0.02	28	69	0.02
PENALTY2	200	202	238	0.03	191	221	0.05
PENALTY3	200	36	102	0.58	99	285	1.78
POWELLSG	5000	28	70	0.01	26	53	0.02
POWER	10000	360	736	0.63	372	754	0.76
QUARTC	5000	15	32	0.02	17	37	0.03
ROSENBR	2	28	84	0.02	34	77	0.02
S308	2	7	21	0.02	8	19	0.02
SCHMVETT	5000	41	72	0.19	43	73	0.23
SENSORS	100	27	73	0.39	21	50	0.25
SINEVAL	2	46	181	0.02	64	144	0.02
SINQUAD	5000	14	45	0.08	14	40	0.09
SISSER	2	5	19	0.02	6	18	0.02
SNAIL	2	61	251	0.02	100	230	0.02
SPARSINE	5000	22306	22552	86	18358	18647	73
SROSENBR	5000	9	23	0.02	11	23	0.02
STRATEC	10	170	419	6.22	462	1043	19.98
TESTQUAD	5000	1580	1587	1.34E+00	1577	1584	1.52E+00
TOINTGOR	50	118	215	0.02	135	233	0.02
TOINTGSS	5000	4	9	0.02	4	9	0.02
TOINTPSP	50	120	262	0.02	143	279	0.02
TOINTQOR	50	29	36	0.02	29	36	0.02
TQUARTIC	5000	11	41	0.03	14	40	0.03
TRIDIA	5000	783	790	0.91	782	789	0.84
VARDIM	200	9	20	0.02	11	25	0.02
VAREIGVL	50	24	54	0.02	23	47	0.02
VIBRBEAM	8	98	255	0.02	138	323	0.02
WATSON	12	58	219	0.02	49	102	0.02
WOODS	4000	24	61	0.03	22	51	0.06
YFITU	3	68	208	0.02	84	197	0.02
ZANGWIL2	2	1	3	0.02	1	3	0.02

Conclusion

In this study, we proposed a new modification of conjugate gradient method $\left(d_k^{ATAZ}\right)$ with restart condition is presented. In addition, new restart criterion proposed for FR CG method. Our numerical results had shown that the new coefficients are better than other conventional CG methods. Furthermore the efficiency of the FR method improved substantially when the new restart criterion is used.

Acknowledgements

The authors are grateful to the editor and the anonymous reviewers for their valuable comments and suggestions, which have substantially improved this paper. This research was partially supported by the Center of Research and Innovation Management, Universiti Malaysia Terengganu.

Author's Contributions

Ahmad Alhawarat: Methodology and sofrware.
Zabidin Salleh: Proof theory and review the paper with the software.
Ibtisam A. Masmali: Support software and review the main CG formula with its proof.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

Al-Baali, M. (1985). Descent property and global convergence of the Fletcher-Reeves method with inexact line search. IMA Journal of Numerical Analysis, 5(1), 121-124. https://doi.org/10.1093/imanum/5.1.121

Alhawarat, A., Salleh, Z., Mamat, M., & Rivaie, M. (2017). An efficient modified Polak–Ribière–Polyak conjugate gradient method with global convergence properties. Optimization Methods and Software, 32(6), 1299-1312. https://doi.org/10.1080/10556788.2016.1266354

Andrei, N. (2008). An unconstrained optimization test functions collection. Advance Modelling Optima, 10(1), 147-161.

Beale, E. M. L. (1972). A deviation of conjugate gradients. Numerical Methods for Nonlinear Optimization, 39-43.

Bongartz, I., Conn, A. R., Gould, N., & Toint, P. L., (1995). CUTE: Constrained and unconstrained testing environment. ACM Transactions on Mathematical Software (TOMS), 21(1), pp, 123-160. https://doi.org/10.1145/200979.201043

Dai, Y. H., & Liao, L. Z. (2001). New conjugacy conditions and related nonlinear conjugate gradient methods. Applied Mathematics and Optimization, 43(1), 87-101. https://doi.org/10.1007/s002450010019

Dai, Y., & Yuan, Y. (1998). Convergence properties of Beale-Powell restart algorithm. Science in China Series A: Mathematics, 41(11), 1142-1150. https://doi.org/10.1007/BF02871976

Dolan, E. D., & Moré, J. J. (2002). Benchmarking optimization software with performance profiles. Mathematical Programming, 91(2), 201-213. https://doi.org/10.1007/s101070100263

Fletcher, R., & Reeves, C. M. (1964). Function minimization by conjugate gradients. The Computer Journal, 7(2), 149-154. https://doi.org/10.1093/comjnl/7.2.149

Gilbert, J. C., & Nocedal, J. (1992). Global convergence properties of conjugate gradient methods for optimization. SIAM Journal on optimization, 2(1), 21-42. https://doi.org/10.1137/0802003

Guanghui, L., Jiye, H., & Hongxia, Y. (1995). Global convergence of the Fletcher-Reeves algorithm with inexact linesearch. Applied Mathematics-A Journal of Chinese Universities, 10(1), 75-82. https://doi.org/10.1007/BF02663897

Hager, W. W., & Zhang, H. (2005). A new conjugate gradient method with guaranteed descent and an efficient line search. SIAM Journal on optimization, 16(1), 170-192. https://doi.org/10.1137/030601880

Hager, W. W., & Zhang, H. (2013). The limited memory conjugate gradient method. SIAM Journal on Optimization, 23(4), 2150-2168. https://doi.org/10.1137/120898097

Hestenes, M. R., & Stiefel, E. (1952). Methods of conjugate gradients for solving linear systems (Vol. 49, No. 1). Washington, DC: NBS. https://doi.org/10.6028/jres.049.044

Kaelo, P., Mtagulwa, P., & Thuto, M. V. (2020). A globally convergent hybrid conjugate gradient method with strong Wolfe conditions for unconstrained optimization. Mathematical Sciences, 14(1), 1-9. https://doi.org/10.1007/s40096-019-00310-y

Polak, E., & Ribiere, G. (1969). Note on the convergence of conjugate direction methods. ESAIM: Mathematical Modeling and Numerical Analysis-Mathematical Modeling and Numerical Analysis, 3 (R1), 35-43. https://doi.org/10.1051/m2an/196903R100351

Powell, M. J. (1984). Nonconvex minimization calculations and the conjugate gradient method. In Numerical analysis (pp. 122-141). Springer, Berlin, Heidelberg. https://doi.org/10.1007/BFb0099521

Powell, M. J. D. (1977). Restart procedures for the conjugate gradient method. Mathematical Programming, 12(1), 241-254. https://doi.org/10.1007/BF01593790

Wolfe, P. (1969). Convergence conditions for ascent methods. SIAM Review, 11(2), 226-235. https://doi.org/10.1137/1011036

Wolfe, P. (1971). Convergence conditions for ascent methods. II: Some corrections. SIAM Review, 13(2), 185-188. https://doi.org/10.1137/1013035

Zoutendijk, G. (1970). Nonlinear programming, computational methods. Integer and Nonlinear Programming, 37-86. https://ci.nii.ac.jp/naid/10030666308/