The Scientific Journal of King Faisal University

Basic and Applied Sciences

A Sufficient Condition for the Global Convergence of Conjugate Gradient Methods for Solving Unconstrained Optimisation Problems

Osman O.O.Yousif [1], Awad Abdelrahman [1], Mogtaba Mohammed [2], Mohammed A. Saleh [3]

[1] Department of Mathematics, Faculty of Mathematical and Computer Sciences, University of Gezira, Wad Madani, Sudan
[2] Department of Mathematics, College of Science, Majmaah University, Zulfi, Saudi Arabia
[3] Department of Computer, College of Science and Arts in Ar Rass, Qassim University, Ar Rass, Saudi Arabia

LINK	RECEIVED	ACCEPTED	PUBLISHED ONLINE	ASSIGNED TO AN ISSUE
https://doi.org/10.37575/b/sci/220013	15/03/2022	18/04/2022	18/04/2022	01/06/2022
NO. OF WORDS	**NO. OF PAGES**	**YEAR**	**VOLUME**	**ISSUE**
5805	7	2022	23	1

ABSTRACT

Due to their remarkable convergence properties and performance in practice, conjugate gradient (CG) methods are widely used for solving unconstrained optimisation problems, especially those of large scale. From the 1950s until now, many studies have been carried out to propose new ones to improve existing CG methods. In this paper, we present a condition that guarantees the global convergence of CG methods when they are applied under the exact line search. At the same time, based on this condition, we did a minor modification on the CG methods of Polak-Rebiere-Polyak (PRP) and of Hestenes-Stiefel (HS) to propose new modified methods. Furthermore, to support the theoretical proof of the global convergence of the modified methods in practical computation, a numerical experiment based on comparing the proposed methods with other well-known CG methods was done. It has been found that the new modified methods have the fewest number of iterations and require the shortest time for solving the problems. In addition, they have the highest percentage of the test problems that solved successfully. Hence, we conclude that they can be used successfully for solving unconstrained optimisation problems.

KEYWORDS

Unconstrained optimisation problems; conjugate gradient methods; exact line search; global convergence

CITATION

Yousif, O.O.O., Abdelrahman, A., Mohammed, M., Saleh, M.A. (2022). A sufficient condition for the global convergence of conjugate gradient methods for solving unconstrained optimisation problems. *The Scientific Journal of King Faisal University: Basic and Applied Sciences*, **23**(1), 106–12. DOI: 10.37575/b/sci/220013

1. Introduction

The conjugate gradients (CG) methods are one of the most widely used methods for solving unconstrained optimisation problems, especially those of large scale. The general formula of an unconstrained optimisation problem is

$$\min_{x\in R^n} f(x) \tag{1.1}$$

where $f: R^n \to R$ is a continuously differentiable function. Starting from an initial point $x_0 \in R^n$, a nonlinear conjugate gradient method generates a sequence of approximation points $\{x_k\}$ using the iterative formula

$$x_{k+1} = x_k + \alpha_k d_k, \quad k = 0, 1, 2, \ldots, \tag{1.2}$$

where $\alpha_k > 0$ is a step length that is obtained by means of a one-dimensional search direction method called line search, and d_k is the search direction which is computed as follows:

$$d_k = \begin{cases} -g_k, & \text{if } k = 0, \\ -g_k + \beta_k d_{k-1}, & \text{if } k \geq 1, \end{cases} \tag{1.3}$$

where β_k is known as the conjugate gradient coefficient and $g_k = \nabla f(x_k)$ is the gradient of the function f at x_k.

If the line search is exact, the step length α_k is obtained in the direction d_k by the rule

$$f(x_k + \alpha_k d_k) = \min_{\alpha\geq 0} f(x_k + \alpha d_k) \tag{1.4}$$

which the orthogonality condition

$$g_k^T d_{k-1} = 0, \tag{1.5}$$

is satisfied.

In addition, we note

$$g_k^T d_k = -\|g_k\|^2, \tag{1.6}$$

that is, by combining (1.3) and (1.5) together.

There are other rules for finding $\alpha_k > 0$ that guarantee the global convergence of a CG method. These rules are called the inexact line search methods. The most popular one is expressed by Wolfe conditions (Wolfe, 1969; Wolfe, 1971).

Different choices for the coefficient β_k lead to different CG methods, such as the method of Fletcher-Reeves (1964), Dai-Yuan (2000) and the Conjugate Descent (Fletcher, 1987), where coefficients are respectively given by

$$\beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}, \tag{1.7}$$

$$\beta_k^{DY} = \frac{\|g_k\|^2}{d_{k-1}^T(g_k - g_{k-1})}, \tag{1.8}$$

$$\beta_k^{CD} = -\frac{\|g_k\|^2}{d_{k-1}^T g_{k-1}}, \tag{1.9}$$

where $\|.\|$ stands for the Euclidean norm of vectors. Clearly, (1.7), (1.8) and (1.9) are identical when the line search used is exact line search. However, the methods of Polak-Rebiere (1969) and Polyak (1990), Hestenes-Stiefel (1952) and Liu-Storey (1992), whose coefficients are respectively given by

$$\beta_k^{PRP} = \frac{g_k^T(g_k - g_{k-1})}{\|g_{k-1}\|^2},$$

$$\beta_k^{HS} = \frac{g_k^T(g_k - g_{k-1})}{d_{k-1}^T(g_k - g_{k-1})},$$

$$\beta_k^{LS} = -\frac{g_k^T(g_k - g_{k-1})}{d_{k-1}^T g_{k-1}},$$

are also identical when exact line search used. Many studies have been carried out to analyse the global convergence of conjugate gradient methods under both exact and inexact line searches. The global convergence of the FR(1.7) is known as Fletcher and Reeves

Corresponding Author: Osman O.O.Yousif 0024911933535, osman_om@hotmail.com

(1964) method was established using both exact (Zoutendijk, 1970) and inexact (Al-Baali, 1985) line search on general functions. The HS (known as Hestenes-Stiefel (1952)) and PRP (known as Polak – Ribiere-Polyak (1969)) methods share the common numerator address, the jamming of the FR, that is, when jamming occurs $g_{k+1} \approx g_k, \beta_k^{HS}$ and $\beta_k^{PRP} \approx 0$, so that $d_{k+1} = -g_{k+1}$. In other words, the HS and PRP methods perform a restart when they encounter a bad direction. This explains why HS and PRP perform much better than the FR in practice. Nevertheless, as a consequence, by the example of Powell *et al.* (1984), the HS and PRP methods may not converge, even if the line search is exact. Therefore, Powell *et al.* (1986) suggested that β_k^{PRP} should be modified by

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

which is equivalent to

$$\beta_k^{PRP+} = \begin{cases} \beta_k^{PRP} & \text{if } \beta_k^{PRP} \geq 0 \\ 0, & \text{otherwise.} \end{cases} \tag{1.10}$$

Inspired by Powell's work, Gilbert and Nocedal (1992) proved the convergence of the PRP+ method when the line search is strong and the step length α_k satisfies the sufficient descent property, and showed that PRP+ performs better than PRP. Moreover, Gilbert and Nocedal (1992) extended this by defining

$$\beta_k^{HS+} = \begin{cases} \beta_k^{HS} & \text{if } \beta_k^{HS} \geq 0 \\ 0, & \text{otherwise,} \end{cases} \tag{1.11}$$

and proved that the HS+ method is also convergent. The convergence properties of the HS and PRP methods have been studied by many researchers, e.g. Gonglin *et al.* (2017), Wei *et al.* (2006) and Wei *et al.* (2006).

In the last years, in order to establish the global convergence and to obtain superior numerical performance in practice, much effort has been devoted to develop new conjugate gradient methods and to modify well-known methods, such as the modifications by Abdelrahman *et al.* (2021) on the method in Rivaie *et al.* (2015) and the modification of Abubakar *et al.* (2022) on the Liu-Storey (LS) method.

Rivaie *et al.* (2012) proposed a new coefficient denoted by β_k^{RMIL}, that is

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

and showed that the RMIL method can be used in practical computation and is globally convergent when it is applied under exact line search. However, Dai (2016) pointed out a mistake in the steps of the global convergence proof. To guarantee the convergence via exact line search, he suggested the modified RMIL+ method, in which the coefficient is given by

$$\beta_k^{RMIL+} = \begin{cases} \frac{g_k^T(g_k - g_{k-1})}{\|d_{k-1}\|^2} & \text{if } 0 \leq g_k^T g_{k-1} \leq \|g_k\|^2 \\ 0, & \text{otherwise.} \end{cases} \tag{1.12}$$

The global convergence of the RMIL method is essentially dependent on the inequality

$$0 \leq \beta_k^{RMIL} \leq \frac{\|g_k\|^2}{\|d_{k-1}\|^2}, \text{ for } k \geq 1, \tag{1.13}$$

as shown in Rivaie *et al.* (2012) and Dai (2016).

In addition, in 2020, based on (1.13), Yousif has proven the global convergence of RMIL+ via strong Wolfe line search.

Therefore, if we generalise inequality (1.13) for any CG coefficient β_k^* with the following extension

$$0 \leq \beta_k^* < C\frac{\|g_k\|^2}{\|d_{k-1}\|^2}, \text{ for } k \geq 1 \text{ and a real number } C \geq 1, \tag{1.14}$$

we expect to get better results. Furthermore, based on the condition in (1.14) and for better convergence properties, we can obtain modified CG methods by doing a little modification on any CG coefficient in order to satisfy it (1.14).

In this paper, we will prove that the global convergence to any CG method satisfies the condition in (1.14) when it is applied under the exact line search in Section 2. Based on this condition, we propose new modified coefficients for both the PRP and the HS methods that are in Section 3. In Section 4, in order to show the efficiency of the modified versions of PRP and HS in practical computation, we compare them with the PRP, HS, PRP+, FR, and RMIL methods. In Section 5, we give a conclusion.

2. A Condition for the Coefficient β_k

In this section, motivated by the denominator of β_k^{RMIL} and the steps of the proof in Yousif (2020) of the global convergence of RMIL+, we will show that every CG method whose coefficient β_k^* satisfies the condition in (1.14) is globally convergent when it is applied under the exact line search for solving unconstrained optimisation problems.

Next, we will prove the global convergence of the CG method, whose coefficient is given by (1.14). Before that, we note if the sequences $\{g_k\}$ and $\{d_k\}$ are generated by any CG method via the exact line search:

$$\begin{aligned} \|g_k + d_k\|^2 &= (g_k + d_k)^t (g_k + d_k), \\ &= \|g_k\|^2 + \|d_k\|^2 + 2g_k^t d_k \\ &= \|g_k\|^2 + \|d_k\|^2 - 2\|g_k\|^2 \qquad \text{(By using (1.6))} \\ &= \|d_k\|^2 - \|g_k\|^2. \end{aligned} \tag{2.1}$$

Therefore,

$$\|d_k\|^2 \geq \|g_k\|^2,$$

which means

$$\frac{\|g_k\|^2}{\|d_k\|^2} \leq 1. \tag{2.2}$$

In addition, we can obtain (2.2) by noting that

$$\frac{\|g_k\|^4}{\|d_k\|^2} = \frac{(g_k^T d_k)^2}{\|d_k\|^2} = \frac{\|g_k\|^2\|d_k\|^2 \cos^2\theta_k}{\|d_k\|^2}.$$

Since $0 \leq \cos^2\theta_k \leq 1$, we have

$$\frac{\|g_k\|^4}{\|d_k\|^2} \leq \|g_k\|^2,$$

which leads again to (2.2).

To prove the global convergence, we assume that the objective function $f(x)$ satisfies the following assumption.

Assumption 2.1

i. The level set $\Omega = \{x \in R^n : f(x) \leq f(x_0)\}$ is bounded, where x_0 is the starting point.
ii. In some neighborhood N of Ω, the objective function is continuously differentiable, and its gradient is Lipschitz continuous; namely, there exists a constant $l > 0$ such that $\|g(x) - g(y)\| \leq l\,\|x - y\|$, $\forall\, x, y \in N$.

From (ii) in Assumption 2.1, we have

$$\|g_{k+1}\| - \|g_k\| \leq \|g_{k+1} - g_k\| \leq l\,\|x_{k+1} - x_k\| = l\alpha_k\|d_k\|.$$

Using the iterative formula (1.2) and the inequality (2.3), we come to

$$\|g_{k+1}\| \le \|d_k\|(1 + l\,\alpha_{max}) \quad (2.3)$$

where α_{max} is a user-supplied bound on the maximum step length allowed in the practical computation.

Under Assumption 2.1, we have the following lemma, which was proved by Zoutendijk (1970).

Lemma 2.1

We suppose that Assumption 2.1 holds. Consider any CG method of the form (1.2) - (1.3), where d_k is a descent search direction and α_k is a step length obtained by means of a one-dimensional search direction. Then the following condition known as the Zoutendijk condition holds

$$\sum_{k=0}^{\infty} \|g_k\|^2 \cos^2\theta_k < \infty, \quad (2.4)$$

where θ_k is the angle between d_k and the steepest descent direction $-g_k$.

The Zoutendijk condition (2.4) implies that

$$\lim_{k=0} \|g_k\|^2 \cos^2\theta_k = 0.$$

This means if the angle θ_k is bounded away from $\frac{\pi}{2}$, there exists a positive constant δ such that

$$\cos\theta_k \ge \delta > 0, \text{for all } k,$$

or

$$\tan\theta_k < \infty, \quad \text{for all.}$$

It follows immediately

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

which guarantees the global convergence.

In addition, from (2.5) and (1.6), we have

$$\sum_{k=0}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} < \infty. \quad (2.5)$$

Now, substituting (1.14) in (1.3), gives

$$d_k = -g_k + \beta_k^* d_{k-1}, \quad k \ge 1.$$

Squaring both sides of the above equation and then using (1.5), we come to

$$\|d_k\|^2 = \|g_k\|^2 + (\beta_k^*)^2 \|d_{k-1}\|^2, \quad k \ge 1.$$

From the definition of β_k^* in (1.14), we get

$$\|d_k\|^2 < \|g_k\|^2 + C\beta_k^* \|g_k\|^2, k \ge 1.$$

Therefore, using (2.1), we come to

$$\beta_k^* > \frac{1}{C}\left(\frac{\|d_k\|^2 - \|g_k\|^2}{\|g_k\|^2}\right) = \frac{1}{C}\left(\frac{\|g_k + d_k\|^2}{\|g_k\|^2}\right). \quad (2.6)$$

Also, if θ_k is the angle between d_k and the steepest descent direction $-g_k$, then

$$\cos\theta_k = \frac{-g_k^T d_k}{\|g_k\|\|d_k\|}.$$

From (1.6), we get

$$\cos\theta_k = \frac{\|g_k\|}{\|d_k\|}. \quad (2.7)$$

Since the cosine is positive in the interval $\left[0, \frac{\pi}{2}\right]$ and negative in $\left[\frac{\pi}{2}, \pi\right]$, the equation (2.8) implies that $\theta_k \in \left[0, \frac{\pi}{2}\right]$.

From (2.7), we find

$$\|d_k\| = \|g_k\| \sec\theta_k. \quad (2.8)$$

Both (2.1) and (2.8) together lead to

$$\sec^2\theta_k - 1 = \frac{\|d_k\|^2}{\|g_k\|^2} - 1 = \frac{\|d_k\|^2 - \|g_k\|^2}{\|g_k\|^2} = \frac{\|g_k + d_k\|^2}{\|g_k\|^2}. \quad (2.9)$$

From the formula (2.6) of β_k^* and (2.9) above, we get

$$\beta_k^* > \frac{1}{C}\left(\frac{\|g_k + d_k\|^2}{\|g_k\|^2}\right) = \frac{1}{C}(\sec^2\theta_k - 1) = \frac{1}{C}\tan^2\theta_k. \quad (2.10)$$

The following theorem establishes the global convergence.

Theorem 2.1

We suppose that Assumption 2.1 holds. Then the CG method, its coefficient given by (1.14), is globally convergent under the exact line search, that is

$$\lim_{k\to\infty} \inf \|g_k\| = 0. \quad (2.11)$$

Proof

The proof is by contradiction. It assumes that (2.11) does not hold; then there exists a constant $\varepsilon > 0$ and an integer k_1 such that

$$\|g_k\| \ge \varepsilon, \text{for all } k \ge k_1, \quad (2.12)$$

which leads to

$$\frac{1}{\|g_k\|^2} \le \frac{1}{\varepsilon^2}, \quad \text{for all } k \ge k_1. \quad (2.13)$$

From (1.3), by squaring both sides of $d_k + g_k = \beta_k^* d_{k-1}$, we get

$$\|d_k\|^2 = -\|g_k\|^2 - 2g_k^T d_k + (\beta_k^*)^2 \|d_{k-1}\|^2. \quad (2.14)$$

Using (1.6) and substituting (1.14), we obtain

$$\|d_k\|^2 < \|g_k\|^2 + C^2 \frac{\|g_k\|^4}{\|d_{k-1}\|^2}. \quad (2.15)$$

Dividing both sides of (2.15) by $\|g_k\|^4$, we get

$$\frac{\|d_k\|^2}{\|g_k\|^4} < \frac{1}{\|g_k\|^2} + \frac{C^2}{\|d_{k-1}\|^2}.$$

From (2.2), since $\frac{1}{\|d_k\|^2} \le \frac{1}{\|g_k\|^2}$, we have

$$\frac{\|d_k\|^2}{\|g_k\|^4} < \frac{1}{\|g_k\|^2} + \frac{C^2}{\|g_{k-1}\|^2}. \quad (2.16)$$

Combining (2.13) and (2.16), we come to

$$\frac{\|d_k\|^2}{\|g_k\|^4} < \frac{(1+C^2)}{\varepsilon^2}, \quad \text{for all } k \ge k_1 + 1.$$

This means

$$\frac{\|g_k\|^4}{\|d_k\|^2} > \frac{\varepsilon^2}{(1+C^2)},$$

then

$$\sum_{k=k_1+1}^{n} \frac{\|g_k\|^4}{\|d_k\|^2} > (n - k_1)\frac{\varepsilon^2}{(1+C^2)}.$$

Since $\sum_{k=0}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} \ge \sum_{k=k_1+1}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2}$

and

$$\sum_{k=k_1+1}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} = \lim_{n\to\infty} \sum_{k=k_1+1}^{n} \frac{\|g_k\|^4}{\|d_k\|^2} > \frac{\varepsilon^2}{(1+C^2)} \lim_{n\to\infty}(n - k_1) = \infty,$$

we come to

$$\sum_{k=0}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} > \infty.$$

This contradicts (2.5). Therefore, the proof is completed.

3. Modified Versions for the PRP and HS Methods

In this section, motivated by the proof of the global convergence in Section 2, the outperformance of the PRP and the HS methods in practical computations for the minimisers of functions, and by the formulas in (1.10) and (1.11), we propose modified versions of the PRP and HS methods, that is, to restrict the coefficients β_k^{PRP} and β_k^{HS} in order to satisfy the coefficient β_k^* in (2.1) as follows:

$$\beta_k^{PRP*} = \begin{cases} \beta_k^{PRP}, & \text{if } 0 \le \beta_k^{PRP} < \mu \frac{\|g_k\|^2}{\|d_{k-1}\|^2} \\ 0, & \text{otherwise,} \end{cases} \tag{3.1}$$

and

$$\beta_k^{HS*} = \begin{cases} \beta_k^{HS}, & \text{if } 0 \le \beta_k^{HS} < \mu \frac{\|g_k\|^2}{\|d_{k-1}\|^2} \\ 0, & \text{otherwise,} \end{cases} \tag{3.2}$$

where $\mu \ge 1$. Clearly the new proposed coefficients β_k^{PRP*} and β_k^{HS*} satisfy the condition (2.1), that is,

$$0 \le \beta_k^{PRP*} < C\frac{\|g_k\|^2}{\|d_{k-1}\|^2}, \text{for } k \ge 1 \text{ and a real number } C \ge 1,$$

and

$$0 \le \beta_k^{HS*} < C\frac{\|g_k\|^2}{\|d_{k-1}\|^2}, \ \text{ for } k \ge 1 \text{ and a real number } C \ge 1,$$

with $\mu = C$, so that both coefficients β_k^{PRP*} and β_k^{HS*} can be considered modified versions of β_k^{PRP} and β_k^{HS}. This means, from Theorem 2.1, the proposed CG methods PRP* and HS* are globally convergent when they are applied under the exact line search.

We also note, like the PRP and HS methods, the PRP* and HS* methods perform a restart when they encounter a bad direction, i.e. when g_k approaches g_{k-1}, then both β_k^{PRP*} and β_k^{HS*} approach zero, so that d_k approaches $-g_k$. Hence, we expect that they perform better than the FR method in practice. Also, like PRP, HS, FR, PRP+, and RMIL, both PRP* and HS* are globally convergent under the exact line search as proved in Theorem 2.1, but it remains to show their performance in practical computations. This will be done in the next section.

4. Numerical Results and Discussion

In this section, to show the efficiency of the PRP* and HS* methods in practical computation when they are applied under the exact line search, we compare them with the FR, PRP, HS, PRP+, and RMIL methods. The comparison is based on solving 41 well-known unconstrained optimisation problems; most of them are from Andrei (2008). The test problems were implemented under low, medium, and high dimensions, namely 2, 3, 4, 10, 50, 100, 500, 1000, and 10,000. To show the robustness, two different initial points for each dimension were chosen. The comparison is based on the number of iterations and the time (in seconds) of run (CPU) of each problem. To do the comparison, a MATLAB coded program was run with a stopping criterion set to $\|g_k\| < 10^{-6}$. In Tables 1, 2, 3, and 4, we report 'Fail' if a method failed to solve a problem. In Tables 1, 2, 3, and 4, a method is considered to be failed, and we report 'Fail' if the number of iterations exceeds 15.

Table 1: A comparison between FR, HS, PRP, and PRP+ for low dimensions

No.	Test Problem	Dim.	FR NOI/ CPU	HS NOI/ CPU	PRP NOI/ CPU	PRP+ NOI/ CPU
1	EXTENDED WHITE & HOLST	2	74/ 0.28 283/ 0.88	16/ 0.08 30/ 0.15	16/ 0.08 30/ 0.15	17/ 0.09 46/ 0.22
2	NONSCOMP	**2**	50/ 0.24 915/ 3.72	10/ 0.07 12/ 0.08	10/ 0.07 14/ 0.09	9/ 0.06 14/ 0.09
3	THREE-HUMP	2	19/ 0.11 Fail	30/ 0.15 19/ 0.11	13/ 0.08 16/0.10	22/ 0.12 20/ 0.11
4	SIX-HUMP	2	11/ 0.07 8/ 0.06	4/ 0.04 8/ 0.06	4/ 0.04 8/ 0.06	4/ 0.04 9/ 0.08
5	CUBE	2	37/ 0.22 611/ 3.05	15/ 0.10 30/ 0.18	11/ 0.08 30/ 0.18	9/ 0.07 47/ 0.27
6	LEON	2	74/ 0.36 263/ 1.51	16/ 0.09 30/ 0.15	16/ 0.09 30/ 0.16	17/ 0.10 46/ 0.23
7	DIXON & PRICE	3	15/ 0.11 29/ 0.16	13/ 0.10 43/ 0.23	13/ 0.10 49/ 0.26	10/ 0.09 49/ 0.26
8	QUARTIC	4	160/ 0.88 271/1 .46	454/ 2.46 460/ 2.48	456/ 2.47 365/ 1.97	456/ 2.46 365/ 1.97
9	COLVILLE	4	Fail 34/ 0.20	139/ 0.71 82/ 0.40	139/ 0.71 82/ 0.40	139/ 0.71 92/ 0.46
10	EXTENDED MARATOS	4	2/ 0.02 548/2.31	3/ 0.03 27/ 0.14	3/ 0.03 21/ 0.12	3/ 0.03 30/ 0.17
11	EXTENDED POWELL	4	Fail Fail	1581/8.51 943/5.11	1581/8.51 1207/6.53	1672/9.03 1698/ 9.13
12	EXTENDED WOOD	4	Fail Fail	226/ 1.21 199/1.05	180/ 0.98 259/1.36	463/ 2.40 209/1.12
13	FREUDENSTEIN & ROTH	4	15/ 0.10 27/ 0.13	7/ 0.05 7/ 0.05	7/ 0.05 10/ 0.08	8/ 0.06 Fail
14	GENERALIZED TRIDIAGONAL 2	4	5/ 0.05 Fail	4/ 0.04 9/ 0.08	4/ 0.04 9/ 0.08	4/ 0.04 10/ 0.09
15	GENERALIZED TRIDIAGONAL 1	10	36/ 0.22 43/ 0.26	27/ 0.17 27/ 0.17	27/ 0.17 27/ 0.17	27/ 0.17 27/ 0.17
16	EXTENDED PENALTY	10	13/ 0.10 12/ 0.09	29/ 0.17 6/ 0.07	29/ 0.17 6/ 0.07	26/ 0.14 7/ 0.08
17	ARWHEAD	10	7/ 0.08 9/ 0.09	5/ 0.07 8/ 0.09	5/ 0.07 8/ 0.09	6/ 0.08 9/ 0.09
18	LIARWHD	10	Fail Fail	20/ 0.15 20/ 0.15	20/ 0.15 20/ 0.15	21/ 0.16 21/ 0.15
19	POWER	10	20/ 0.14 24/ 0.14	22/ 0.15 25/ 0.15	21/ 0.13 25/ 0.15	21/ 0.13 25/ 0.15

Table 2: A comparison between HS*, PRP*, and RMIL for low dimensions

No.	Test Problem	Dim.	HS* $(\mu = 10)$ NOI/ CPU	PRP* $(\mu = 10)$ NOI/ CPU	PRP* $(\mu = 5)$ NOI/ CPU	RMIL NOI/ CPU
1	EXTENDED WHITE & HOLST	2	17/ 0.09 28/ 0.12	17/ 0.09 32/ 0.16	26/ 0.11 25/ 0.11	23/ 0.11 24/ 0.10
2	NONSCOMP	**2**	9/ 0.06 15/ 0.10	9/ 0.06 15/ 0.10	11/ 0.08 15/ 0.10	15/ 0.10 16/ 0.12
3	THREE-HUMP	2	19/ 0.11 21/ 0.11	22/ 0.12 22/ 0.12	22/ 0.12 22/ 0.12	14/ 0.09 Fail
4	SIX-HUMP	2	5/ 0.05 8/ 0.06	5/ 0.05 8/ 0.06	5/ 0.05 8/ 0.06	5/ 0.05 8/ 0.06
5	CUBE	2	10/ 0.07 28/ 0.17	10/ 0.07 30/ 0.18	10/ 0.07 25/ 0.16	32/ 0.20 24/ 0.16
6	LEON	2	17/ 0.10 28/ 0.14	17/ 0.10 30/ 0.16	17/ 0.10 25/ 0.14	23/ 0.12 24/ 0.15
7	DIXON & PRICE	3	10/ 0.09 43/ 0.23	10/ 0.09 49/ 0.26	10/ 0.09 49/ 0.26	35/ 0.22 56/ 0.34
8	QUARTIC	4	454/ 2.46 460/ 2.50	456/ 2.47 365/ 1.99	456/ 2.47 365/ 1.97	740/ 3.95 804/ 4.31
9	COLVILLE	4	139/ 0.71 92/ 0.46	139/ 0.71 92/0.46	139/ 0.71 92/ 0.46	375/ 1.88 290/ 1.93
10	EXTENDED MARATOS	4	2/ 0.02 26/ 0.14	2/ 0.02 26/ 0.14	2/ 0.02 29/ 0.16	3/ 0.03 18/ 0.11
11	EXTENDED POWELL	4	1670/9.00 1694/9.20	1672/9.03 1698/9.13	528/2.89 1698/9.13	Fail Fail
12	EXTENDED WOOD	4	175/ 0.95 372/1.92	463/ 2.40 209/ 1.12	182/ 0.98 85/ 0.46	981/5.06 1021/5.30
13	FREUDENSTEIN & ROTH	4	7/ 0.05 Fail	7/ 0.05 Fail	8/ 0.06 9/ 0.07	9/ 0.07 Fail
14	GENERALIZED TRIDIAGONAL 2	4	4/ 0.04 11/ 0.10	4/ 0.04 11/ 0.10	4/ 0.04 8/ 0.06	4/ 0.04 7/ 0.06
15	GENERALIZED TRIDIAGONAL 1	10	27/ 0.17 27/ 0.17	27/ 0.17 27/ 0.17	27/ 0.17 27/ 0.17	25/ 0.16 27/ 0.18
16	EXTENDED PENALTY	10	26/ 0.14 6/ 0.07	26/ 0.14 6/ 0.07	16/ 0.11 6/ 0.07	20/ 0.13 20/ 0.13
17	ARWHEAD	10	5/ 0.07 9/ 0.09	5/ 0.07 9/ 0.09	5/ 0.07 9/ 0.09	6/ 0.08 10/ 0.10
18	LIARWHD	10	50/ 0.29 56/ 0.32	17/ 0.13 21/ 0.15	17/ 0.13 21/ 0.5	19/ 0.12 19/ 0.14
19	POWER	10	22/ 0.15 25/ 0.15	21/ 0.13 25/ 0.15	21/ 0.13 25/ 0.15	123/ 0.60 139/ 0.70

Table 3: A comparison between FR, HS, PRP, and PRP+ for medium and high dimensions

No.	Test Problem	Dim.	FR NOI/ CPU	HS NOI/ CPU	PRP NOI/ CPU	PRP+ NOI/ CPU
1	FLETCHER	50	Fail 35/ 0.17	283/ 1.12 33/ 0.15	283/ 1.12 33/ 0.15	283/ 1.12 33/ 0.15
2	DIXON3DQ	50	27/ 0.20 31/ 0.24	27/ 0.20 30/ 0.23	28/ 0.22 33/ 0.25	28/ 0.22 33/ 0.25
3	QP1	50	55/ 0.27 306/ 1.28	7/ 0.07 Fail	7/ 0.07 Fail	7/ 0.07 Fail
4	QF2	50	116/ 0.60 613/ 2.86	70/ 0.36 55/ 0.29	70/ 0.36 55/ 0.29	70/ 0.36 55/ 0.29
5	QF1	50	38/ 0.21 41/ 0.23	39/ 0.22 41/ 0.23	38/ 0.21 41/ 0.23	38/ 0.21 41/ 0.23
6	HAGER	100	24/ 0.19 21/ 0.21	25/ 0.20 25/ 0.20	25/ 0.20 25/ 0.20	25/ 0.20 25/ 0.20
7	GENERALIZED ROSENBROCK	100	Fail 11018/68.62	842/ 5.56 327/2.16	842/ 5.56 336/2.23	841/ 5.52 336/2.23
8	SUM SQUARE	100	58/ 0.37 61/ 0.38	58/ 0.37 61/ 0.38	58/ 0.37 61/ 0.38	58/ 0.37 61/ 0.38

9	GENERALIZED QUARTIC [16]	100	6/ 0.05 11/ 0.10	6/ 0.05 9/ 0.08	6/ 0.05 9/ 0.08	5/ 0.04 6/ 0.05
10	RAYDAN 1 [16]	100	68/ 0.44 Fail	66/ 0.43 238/ 1.41	67/ 0.44 257/1.53	67/ 0.44 197/ 1.18
12	QP2	500	Fail Fail	45/ 0.86 46/ 0.82	35/ 0.67 46/ 0.82	42/ 0.80 44/ 0.80
12	QUARTC	500	6/ 0.09 12/ 0.14	6/ 0.09 10/ 0.12	6/ 0.09 10/ 0.12	5/ 0.07 7/ 0.10
13	EXTENDED TRIDIAGONAL 1	500 1000	733/ 17.06 452/11.00 843/ 35.98 517/ 21.88	15/ 0.37 13/ 0.32 15/ 0.66 13/ 0.56	15/ 0.37 13/ 0.33 15/ 0.66 13/ 0.56	19/ 0.47 13/ 0.33 19/ 0.81 13/ 0.57
14	EXTENDED DENSCHNB	500 1000	10/ 0.13 13/ 0.15 10/ 0.20 13/ 0.22	5/ 0.08 9/ 0.11 5/ 0.10 9/ 0.21	5/ 0.08 9/ 0.11 5/ 0.10 9/ 0.21	5/ 0.08 9/ 0.11 5/ 0.10 9/ 0.21
15	EXTENDED ROSENBROCK	1000 10000	211/ 2.34 56/ 0.64 227/ 18.23 62/ 5.10	18/ 0.22 20/ 0.25 19/ 1.58 21/ 1.73	18/ 0.22 20/ 0.25 19/ 1.58 20/ 1.66	22/ 0.27 19/ 0.24 22/ 1.84 20/ 1.68
16	EXTENDED HIMMELBLAU	1000 10000	243/ 3.22 15/ 0.23 253/ 25.37 16/ 1.65	9/ 0.15 7/ 0.13 9/ 0.95 7/ 0.73	9/ 0.5 7/ 0.13 9/ 0.95 7/ 0.73	9/ 0.15 7/ 0.13 9/ 0.95 7/ 0.73
17	STRAIT	1000 10000	35/ 0.59 Fail 35/ 3.94 Fail	17/ 0.28 43/ 0.72 17/ 1.94 43/ 4.84	17/ 0.28 43/ 0.72 17/ 1.94 43/ 4.85	17/ 0.28 44/ 0.74 17/ 1.94 44/ 5.95
18	SHALLOW	1000 10000	18/ 0.24 175/ 2.24 19/ 1.71 190/ 16.93	6/ 0.08 9/ 0.14 7/ 0.64 10/ 0.89	6/ 0.08 9/ 0.13 7/ 0.64 10/ 0.89	7/ 0.12 9/ 0.15 7/ 0.65 9/ 0.85
19	EXTENDED BEALE	1000 10000	75/ 3.50 Fai Fail Fail	10/ 0.50 10/ 0.49 10/ 4.35 10/ 4.18	10/ 0.50 10/ 0.49 10/ 4.35 10/ 4.23	10/ 0.50 9/ 0.44 10/ 4.35 9/ 3.84

Table 4: A comparison between HS*, PRP*, and RMIL for medium and high dimensions

No.	Test Problem	Dim.	HS* ($\mu = 10$) NOI/ CPU	PRP* ($\mu = 10$) NOI/ CPU	PRP* ($\mu = 5$) NOI/ CPU	RMIL NOI/ CPU
1	FLETCHER	50	283/ 1.12 33/ 0.15	283/ 1.12 33/ 0.15	283/ 1.12 33/ 0.15	951/3.62 35/ 0.17
2	DIXON3DQ	50	27/ 0.20 30/ 0.23	28/ 0.22 33/ 0.25	28/ 0.22 33/ 0.25	889/ 5.39 992/ 5.89
3	QP1	50	9/ 0.09 10/ 0.10	7/ 0.09 8/ 0.07	8/ 0.08 10/ 0.10	Fail 26/ 0.17
4	QF2	50	70/ 0.36 55/ 0.29	70/ 0.36 55/ 0.29	70/ 0.36 55/ 0.29	78/ 0.42 69/ 0.38
5	QF1	50	39/ 0.22 41/ 0.23	38/ 0.21 41/ 0.23	38/ 0.21 41/ 0.23	69/ 0.37 78/ 0.40
6	HAGER	100	25/ 0.20 25/ 0.20	25/ 0.20 25/ 0.20	25/ 0.20 25/ 0.20	25/ 0.20 26/ 0.21
7	GENERALIZED ROSENBROCK	100	840/5.48 327/2.16	841/ 5.52 336/2.23	841/ 5.52 394/2.60	5179/34.06 7594/49.72
8	SUM SQUARE	100	58/ 0.37 61/ 0.38	58/ 0.37 61/ 0.38	58/ 0.37 61/ 0.38	128/ 0.79 146/ 0.90
9	GENERALIZED QUARTIC [16]	100	5/ 0.04 6/ 0.05	5/ 0.04 6/ 0.05	5/ 0.04 6/ 0.05	6/ 0.05 9/ 0.08
10	RAYDAN 1 [16]	100	66/ 0.43 204/ 1.22	67/ 0.44 197/1.18	67/ 0.44 197/ 1.18	99/ 0.91 690/3.58
11	QP2	500	42/ 0.80 42/ 0.77	45/ 0.86 41/ 0.77	43/ 0.81 37/ 0.68	58/ 0.10 61/ 1.14
12	QUARTC	500	5/ 0.07 7/ 0.10	5/ 0.07 7/ 0.10	5/ 0.07 6/ 0.09	6/ 0.09 10/ 0.12
13	EXTENDED TRIDIAGONAL 1	500 1000	19/ 0.47 19/ 0.47 19/ 0.81 20/ 0.87	19/ 0.47 22/ 0.53 19/ 0.81 22/ 0.94	40/ 0.95 64/ 1.51 52/ 2.22 92/ 3.94	169/ 3.92 186/ 4.29 200/ 8.42 211/ 8.90
14	EXTENDED DENSCHNB	500 1000	5/ 0.08 10/ 0.12 5/ 0.10 10/ 0.17	5/ 0.08 10/ 0.12 5/ 0.10 10/ 0.17	5/ 0.08 10/ 0.12 5/ 0.10 10/ 0.18	6/ 0.09 10/ 0.12 6/ 0.12 10/ 0.17
15	EXTENDED ROSENBROCK	1000 10000	22/ 0.27 19/ 0.24 23/ 1.91 20/ 1.66	22/ 0.27 19/ 0.24 23/ 1.91 20/ 1.66	23/ 0.29 9/ 0.24 23/ 1.91 19/ 1.58	28/ 0.34 22/ 0.26 28/ 2.33 24/ 1.98
16	EXTENDED HIMMELBLAU	1000 10000	9/ 0.15 7/ 0.13 9/ 0.95 8/ 0.84	9/ 0.15 7/ 0.13 9/ 0.95 8/ 0.84	10/ 0.16 7/ 0.13 10/ 1.05 8/ 0.84	7/ 0.13 10/ 0.17 8/ 0.84 10/ .04
17	STRAIT	1000 10000	17/ 0.28 45/ 0.76 17/ 1.94 45/ 5.05	17/ 0.28 44/ 0.73 17/ 1.94 44/ 4.95	17/ 0.28 44/ 0.73 17/ 1.93 44/ 5.00	38/ 0.66 66/ 1.09 38/ 4.48 66/ 7.50
18	SHALLOW	1000 10000	7/ 0.12 10/ 0.16 8/ 0.74 10/ 0.89	7/ 0.12 10/ 0.16 8/ 0.74 10/ 0.89	7/ 0.12 10/ 0.16 8/ 0.74 10/ 0.89	26/ 0.37 11/ 0.17 29/ 2.58 12/ 1.08
19	EXTENDED BEALE	1000 10000	11/ 0.53 12/ 0.57 11/ 4.76 12/ 5.02	11/ 0.53 12/ 0.57 11/ 4.76 12/ 5.02	12/ 0.57 12/ 0.57 12/ 5.03 12/ 5.02	52/ 2.41 24/ 1.12 54/ 23.12 26/ 11.03

As can be seen from Tables 1, 2, 3, and 4, PRP* with $\mu = 5$ solves all test problems, so it reached 100%, whereas FR, HS, PRP, PRP+, PRP* (with $\mu = 10$), HS* (with $\mu = 10$) and RMIL reached about 90%, 99%, 99%, 98%, 99%, 99%, and 95%, respectively. Therefore, based on the ability of solving test problems, there is a little improvement in PRP* with $\mu = 5$. Furthermore, based on the number of iterations and the CPU time, we can show the performance of the CG methods in Tables 1, 2, 3, and 4 by using the performance profile introduced by Dolan and More (2002). According to Dolan and More, benchmark results or performance profiles are formed by running a method or a solver denoted by S on the test problem denoted by P and recording the information in focus, such as the number of iterations and CPU time. Assuming that n_s solvers and n_p problems occur, for each problem p where $p \in P$ and solver S where $s \in S$, they termed

$t_{p,s}$ = Computing time (the number of iterations or CPU time) or others required solving problem p by solver s.

Using a baseline for comparison, they compared the performance on problem p by solver S with the best performance by any solver on this problem, using the performance ratio:

$$r_{p,s} = \frac{t_{p,s}}{min\{t_{p,s}: s \in S\}}$$

Let us suppose that a parameter $r_M \geq r_{p,s}$ for all p, s is chosen, and $r_{p,s} = r_M$ if solver S does not solve problem p. The performance of solver S on any given problem might be of interest, but due to this, they would prefer to obtain an overall assessment of the performance of the solver, then it was termed as:

$$t_{p,s} = \frac{1}{n_p} size\{p \in P: r_p \leq t\}$$

Thus, $p_s(t)$ was the probability for solver $s \in S$ that a performance ratio $r_{p,s}$ was within a factor $t \in R$ of the best possible ratio, and then function p_s was the cumulative distribution function for the performance ratio. The performance profile $p_s: R \rightarrow [0,1]$ for a solver was non-decreasing, piecewise, and continuous from the right. The value of $p_s(1)$ is the probability that a solver will win over the rest of the solvers. In general, the solver with the highest values of $p_s(t)$ or at the top right of the figure represents the best solver.

Figure 1: The performance based on the NOI

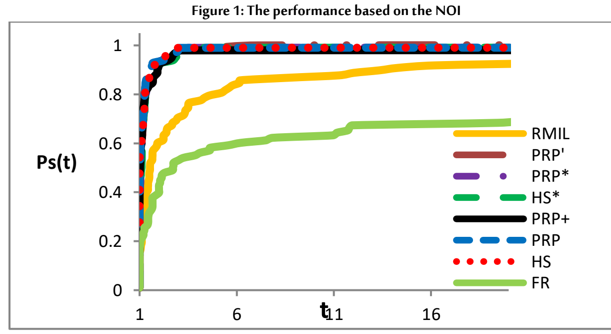
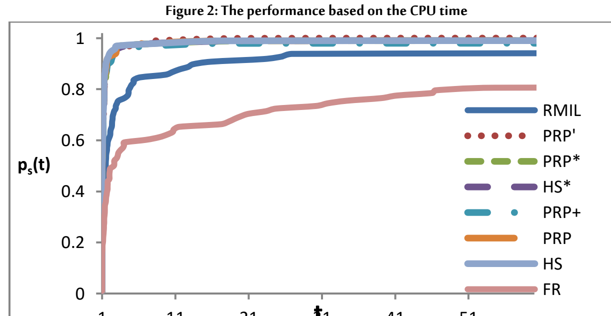


Figure 2: The performance based on the CPU time



In Figures 1 and 2, PRP' represents PRP* with $\mu = 5$. An observation on Figures 1 and 2 shows that HS, PRP, PRP+, HS* with $\mu = 10$, PRP* with $\mu = 10$, and PRP* with $\mu = 5$ are almost identical. Furthermore, their curves lie above the FR and RMIL curves. Therefore, the new HS*, PRP*, and PRP* with $\mu = 5$ perform much better than both the FR and RMIL methods. Moreover, since FR, CD(1.9) is known as

Conjugate Descent (Fletcher, 1987), and DY(1.8) is known as Dai-Yuan (2000) are identical with exact line search, then HS*, PRP*, and PRP* with $\mu = 5$ are also much better than the CD and DY methods.

5. Conclusion

In this paper, we presented a sufficient condition that guarantees the global convergence of the CG methods via the exact line search. Based on the new condition, we proposed new modified coefficients for both the PRP and the HS methods, that is, by restricting their values in order to satisfy the proposed condition. Moreover, to show the efficiency of the modified coefficients of PRP and HS in practical computation, we have compared them with the FR, HS, PRP, PRP+ and RMIL methods. The result of the comparison is that the new ones perform almost as HS, PRP, and PRP+, much better than both FR and RMIL, and a lot better than the CD and DY methods because of the similarities of the FR, CD, and DY methods when the line search is exact. Furthermore, HS* and PRP* are flexible, that is, a certain choice for the value of μ may lead to the solution of an optimisation problem as in Table 1 the PRP* with $\mu = 5$ solved all problems but HS, PRP, PRP+, HS* with $\mu = 10$, and PRP* with $\mu = 10$ did not. Therefore, we conclude that the new modified methods can be used successfully for solving optimisation problems, and they are better than FR, CD, DY, RMIL in all and better than HS, PRP, and PRP+ in some cases.

Biographies

Osman Omer Osman Yousif

Department of Mathematics, Faculty of Mathematical and Computer Sciences, University of Gezira, Wad Madani, Sudan, osman_om@hotmail.com, 00249119733535

Dr. Yousif (Sudanese) is an assistant professor in University of Gezira, Sudan since 2017 after the approve of his Ph.D. degree from University of Malaysia Terengganu, Malaysia. Also, he is now the head of Mathematics department. His research interests lie on the fields of computational mathematics and optimization methods, so he supervised to many graduation researches in this areas. Besides, he participated in four conferences as a presenter in Malaysia and has published four papers in ISI and Scopus Journals.

Awad Abdelrahman Abdalla Mohammed

Department of Mathematics, Faculty of Mathematical and Computer Sciences, University of Gezira, Wad Madani, Sudan, awad.abdalla26@yahoo.com, 00249962255448

Dr. Mohammed (Sudanese) is an assistant Professor in University of Gezira. He has received his Ph.D. degree in mathematical sciences from University of Malaysia Terengganu, Malaysia in 2017. Also, he is now a head of the documentation committee and a member of courses development committee. He interested in optimization and computational mathematics. He supervised to undergraduate and M.Sc. students. He has participated in conferences in Malaysia, Sudan and served as a co-chair of ICCCEEE18 conference which held in Khartoum in 2018. ORCID: 0000-0001-5467-5124.

Mogtaba Mohammed

Department of Mathematics, College of Science, Majmaah University, Zulfi, Saudi Arabia, mogtaba.m@mu.edu.sa, 00966543244128.

Dr. Mohammed (Sudanese) is an associate professor, department of mathematics, Almajmaah University, Saudi Arabia. He has received a Ph.D. and a postdoctorate degrees in mathematical sciences from the University of Pretoria, South Africa. He interests in the areas of mathematical modelling and computational mathematics. He has graduated several M.Sc. students and B.Sc. students. In addition, he has many papers published in conferences proceedings, ISI and Scopus journals. He invited to give talks in several international conferences.

Mohammed A. Saleh

Department of Computer , College of Science and Arts in Ar Rass, Qassim University, Ar Rass, Saudi Arabia, m.saleh@qu.edu.sa, 0096653340144

Dr. Saleh (Sudanese) received the B.Sc. (Honor) in Mathematical and Computer Science, Faculty of Mathematical and Computer Science, University of Gezira, Sudan. He obtained the M.Sc. (First Class) in Information Security, Faculty of Computer Science and Information System, University of Technology (UTM), Malaysia, and the Ph.D. in Information Security (Computer Science), Faculty of Computing, University of Technology (UTM), Malaysia. From 2010 to 2014, he was a Security Engineer with the Sudanese Nation Information Center (NIC). As well, he was Lecturer at Gezira University. Currently, he is an Assistant Professor at Qassim University, KSA. His research interests include Malware Analysis and Artificial Intelligence in Cyber security. He is the author of a couple of journal articles.

References

Abdelrahman, A., Yousif, O.O.O., Mogtaba, M. and Elbahir, M.K. (2021). Global convergence of nonlinear conjugate gradient coefficients with inexact line search. *Scientific Journal of King Faisal University: Basic and Applied Sciences*, **22**(2), 86–91. DOI: 10.37575/b/sci/210058

Abubakar, A.B., Malik, M., Kumam, P., Mohammad, H., Sun, M., Ibrahim, A.H. and Kiri, A.I. (2022). A Liu-Storey-type conjugate gradient method for unconstrained minimization problem with application in motion control. *Journal of King Saud University–Science*, **34**(4), 101923.

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), 121–4.

Andrei, N. (2008). An unconstrained optimization test functions collection. *Adv. Model. Optim*, **10**(1), 147–61.

Dolan, E.D. and Moré, J.J. (2002). Benchmarking optimization software with performance profiles. *Mathematical programming*, **91**(2), 201–13.

Dai, Y.H. and Yuan, Y. (1999). A nonlinear conjugate gradient method with a strong global convergence property. *SIAM Journal on optimization*, **10**(1), 177–82.

Dai, Z. (2016). Comments on a new class of nonlinear conjugate gradient coefficients with global convergence properties. *Applied Mathematics and Computation*, **276**(n/a), 297–300.

Fletcher, R. (1980). *Practical Methods of Optimization: Vol. 1 Unconstrained Optimization*. USA: John Wiley & Sons.

Fletcher, R., and Reeves, C.M. (1964). Function minimization by conjugate gradients. *The Computer Journal*, **7**(2), 149–54.

Gilbert, J.C. and Nocedal, J. (1992). Global convergence properties of conjugate gradient methods for optimization. *SIAM Journal on Optimization*, **2**(1), 21–42.

Hestenes, M.R. and Stiefel, E. (1952). Methods of conjugate gradients for solving. *Journal of research of the National Bureau of Standards*, **49**(6), 409.

Liu, Y. and Storey, C. (1991). Efficient generalized conjugate gradient algorithms, part 1: theory. *Journal of Optimization Theory and Applications*, **69**(1), 129–37.

Rivaie, M., Mamat, M. and Abashar, A. (2015). A new class of nonlinear conjugate gradient coefficients with exact and inexact line searches. *Applied Mathematics and Computation*, **268**(n/a), 1152–63.

Rivaie, M., Mamat, M., June, L.W. and Mohd, I. (2012). A new class of nonlinear conjugate gradient coefficients with global convergence properties. *Applied Mathematics and Computation*, **218**(22), 11323–32.

Polak, E. and Ribiere, G. (1969). Note on the convergence of conjugate direction methods. *Mathematical Modeling and Numerical Analysis*, **3**(1), 35–43.

Polyak, B.T. (1969). The conjugate gradient method in extremal problems. *USSR Computational Mathematics and Mathematical Physics*, **9**(4), 94–112.

Powell, M.J. (1984). Nonconvex minimization calculations and the conjugate gradient method. In *Lecture Notes in Mathematics,* **1066**(n/a), 122–41.

Wei, Z., Li, G. and Qi, L. (2006). New nonlinear conjugate gradient formulas for large-scale unconstrained optimization problems. *Applied Mathematics and Computation*, **179**(2), 407–30.

Wolfe, P. (1969). Convergence conditions for ascent methods. *SIAM Review*, **11**(2), 226–35.

Wolfe, P. (1971). Convergence conditions for ascent methods. II: Some corrections. *SIAM review*, **13**(2), 185–8.

Wei, Z., Yao, S. and Liu, L. (2006). The convergence properties of some new conjugate gradient methods. *Applied Mathematics and Computation*, **183**(2), 1341–50.

Yousif, O.O.O. (2020). The convergence properties of RMIL+ conjugate gradient method under the strong Wolfe line search. *Applied Mathematics and Computation*, **367**(n/a), 124777.

Yuan, G., Wei, Z. and Lu, X. (2017). Global convergence of BFGS and PRP methods under a modified weak Wolfe–Powell line search. *Applied Mathematical Modelling*, **47**(n/a), 811–25.

Zoutendijk, G. (1970). Nonlinear programming, computational methods. In: J. Abadie (ed.) *Integer and Nonlinear Programming*. Amsterdam: North Holllad.