

A new hybrid conjugate gradient algorithm for optimization models and its application to regression analysis

Ibrahim Mohammed Sulaiman¹, Norsuhaily Abu Bakar², Mustafa Mamat³, Basim A. Hassan⁴,
Maulana Malik⁵, Alomari Mohammad Ahmed⁶

^{1,3,5,6}Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Besut Campus, Malaysia

²Faculty of Applied Social Sciences, University Sultan Zainal Abidin, Terengganu, Malaysia

³Institute of Engineering Mathematics, Universiti Malaysia Perlis Arau, Malaysia

⁴Department of Mathematics, College of Computers Sciences and Mathematics, University of Mosul, Iraq

Article Info

Article history:

Received Dec 7, 2020

Revised Jun 4, 2021

Accepted Jun 11, 2021

Keywords:

Conjugate gradient method

Convergence analysis

Line search procedures

Regression analysis

ABSTRACT

The hybrid conjugate gradient (CG) method is among the efficient variants of CG method for solving optimization problems. This is due to their low memory requirements and nice convergence properties. In this paper, we present an efficient hybrid CG method for solving unconstrained optimization models and show that the method satisfies the sufficient descent condition. The global convergence prove of the proposed method would be established under inexact line search. Application of the proposed method to the famous statistical regression model describing the global outbreak of the novel COVID-19 is presented. The study parameterized the model using the weekly increase/decrease of recorded cases from December 30, 2019 to March 30, 2020. Preliminary numerical results on some unconstrained optimization problems show that the proposed method is efficient and promising. Furthermore, the proposed method produced a good regression equation for COVID-19 confirmed cases globally.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Norsuhaily Abu Bakar

Faculty of Applied Social Sciences

Universiti Sultan Zainal Abidin

Gong Badak Campus, 21300, Terengganu, Malaysia

Email: norsuhaily@unisza.edu.my

1. INTRODUCTION

Consider the following nonlinear optimization model

$$\min f(x), \quad x \in R^n \quad (1)$$

where $f: R^n \rightarrow R$ is a smooth function whose first derivative is represented by $g(x) = \nabla f(x)$. The conjugate gradient method is known to play a significant role in solving the above nonlinear optimization models today. Though, many optimization algorithms that are robust with rapid convergence rate than the CG method are available, numerous researchers still refer the CG algorithm due low memory requirement and its simplicity for the large-scale optimization problem [1], [2]. Starting with an initial guess $x_0 \in R^n$, the nonlinear CG algorithm generates the points $\{x_k\} \subset R^n$ using the following recursive formula:

$$x_{k+1} = x_k + \alpha_k d_k, \quad k = 0, 1, 2, \dots \quad (2)$$

α_k is the step size, computed using any line search method along the search direction d_k . For the first iteration, d_k is always the negative gradient i.e., for $k = 0$, $d_k = -g_k$. However, for $k \geq 0$,

$$d_k = -g_k + \beta_k d_{k-1} \tag{3}$$

where the scalar β_k is known as the CG parameter.

For nonlinear conjugate gradient method, the step size α_k is usually determined using the exact and inexact line search. Thus, this study employed the Wolfe inexact line search with formula as follows. The standard Wolfe line search required α_k to satisfy.

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \delta \alpha_k g_k^T d_k \tag{4}$$

$$g_{k+1}^T d_k \geq \sigma g_k^T d_k \tag{5}$$

where $0 < \delta \leq \sigma < 1$. Generally, the line search would be more accurate when the value of σ smaller, though, the computational time would increase. To ease analysis, a stronger condition was proposed using (4) and

$$|g_{k+1}^T d_k| \leq -\sigma g_k^T d_k \tag{6}$$

where the conditions defined by (4) and (6) are called the strong Wolfe line search (SWP) [2]. Generally, d_k is required to satisfy

$$d_k^T g_k < 0 \tag{7}$$

guarantying d_k is a descent direction of $f(x)$ at x_k [3]. To maintain the convergence property, d_k is often require to satisfy the descent condition

$$g_k^T d_k \leq -c \|g_k\|^2 \tag{8}$$

where the constant $c > 0$.

Generally, the CG algorithms differ by the choice of the coefficient β_k . Some of the classical algorithms for β_k are Hestenes-Stiefel (HS) [4], Fletcher-Reeves (FR) [5], Polak-Ribière (PRP) [6], [7], Dai-Yuan (DY) [8], respectively, with their formulas given as:

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

Al-Baali [9] established the convergence of the FR method using SWP method with $\sigma < 1/2$. From computational point of view, the PRP formula possess better numerical results compare to the FR method and [6] proved that PRP method would converge globally when $f(x)$ is strongly convex and the line search condition used is exact. However, the convergence of the PRP method is yet to be established under SWP condition [10]. In fact, Powell [11], [12] gave some counter examples to show that even with the exact minimization condition, there exist some functions, for which PRP method fails to converge. For more reference on convergence properties of recent CG algorithms, please refer to [13], [14]. Recently, Rivaie *et al.* [15] proposed a new variant of the PRP method known as RMIL which represent the researcher's name: Rivaie-Mustafa-Ismail-Leong. This method retained the numerator of the PRP method while defining a new formula for the denominator as:

$$\beta_k^{RMIL} = \frac{g_k^T (g_k - g_{k-1})}{d_{k-1}^T d_{k-1}} \tag{9}$$

and showed that

$$\beta_k^{RMIL} \leq \frac{\|g_k\|^2}{\|d_{k-1}\|^2} \tag{10}$$

This method possesses the restart condition with efficient numerical results. The authors established the global convergence proof of this method under exact line search. The efficiency and robustness of this method motivated us to design a hybrid CG algorithm that would inherit the good convergence result of β_k^{RMIL} but with improved numerical effect of the β_k^{RMIL} algorithm. Furthermore, the proposed algorithm would be extended to solve parameterized regression model to illustrate it applicability to real-life problems.

One of the effective modifications of the CG algorithm is the hybrid CG method. This algorithm usually involves the combination of some of the known CG coefficient to produce a new coefficient with better numerical result while retaining the nice convergence properties of the existing method. Based on the method of Al-Baali [9] defined as:

$$\beta_k \in \{0, \beta_k^{FR}\}$$

Touati and Storey [16] propose the first hybrid CG algorithm as:

$$\beta_k^{TS} = \begin{cases} \beta_k^{PRP} & \text{if } 0 \leq \beta_k^{PRP} \leq \beta_k^{FR} \\ \beta_k^{FR} & \text{else where} \end{cases} \quad (11)$$

This method not only inherited the good convergence result of the FR algorithm, but also produce an efficient numerical performance similar to the PRP method. Based on the structure of (11), Jiangtao *et al.* [14] proposed some efficient modification of FR and DY hybrid CG method and established their convergence results under different line searches. Related hybrid methods were proposed by Dai and Yuan [17] as follows:

$$\beta_k^{hDYZ} = \max\{0, \min\{\beta_k^{HS}, \beta_k^{DY}\}\} \quad (12)$$

$$\beta_k^{hDY} = \max\{-c\beta_k^{DY}, \min\{\beta_k^{HS}, \beta_k^{DY}\}\}$$

where $c = \frac{1-\sigma}{1+\sigma}$. Just like the β_k^{HS} , these hybrid CG methods can perform the restart when a bad direction is generated. Under the Lipschitz continuity of the gradient, the convergence prove of these methods was established with the weak Wolfe line search (4) and (5). For more reading on hybrid CG algorithms and application of the CG method can be referred to [18], [19].

This paper aims to propose a new hybrid CG algorithm for solution of unconstrained minimization functions and further applied the proposed method to solved parameterized regression models. The rest part of the manuscript is organized as follows: Section 2 present our proposed hybrid CG method with its corresponding algorithm, and further established the descent condition and convergence under inexact line search. The results of the numerical experiments based on some standard optimization test problems in Section 3 followed by application to the parameterized regression models is presented in Section. Finally, Section 4 conclude the study.

2. NEW METHOD AND ALGORITHM

Motivated by the good convergence properties and restart condition of RMIL coefficient [15], and the structures of [14], [16], we defined our hybrid coefficient as:

$$\beta_k^{hSM*} = \begin{cases} \beta_k^{RMIL} & \text{if } 0 \leq \beta_k^{RMIL} \leq \beta_k^{hSM} \\ \beta_k^{hSM*} & \text{else where} \end{cases} \quad (13)$$

where

$$\beta_k^{hSM} = \frac{g_k^T(g_k - g_{k-1})}{d_{k-1}^T d_{k-1}} + \frac{2g_k^T g_{k-1}}{d_{k-1}^T d_{k-1}} \quad (14)$$

This method is a combination of a defined β_k^{hSM} algorithm with the classical RMIL algorithm which has better convergence properties. Now, we defined the algorithm for the proposed coefficient below and the assumptions needed for the convergence proof.

Algorithm for β_k^{hSM}

Step 1. Given $x_0 \in \mathbb{R}^n$, $d_0 = -g_0$, set $k := 0$.
 Step 2. Calculate α_k by (4) and (6).
 Step 3. Update the current iterate based on (2).
 Step 4. Compute β_k by (10), (13) and (14)
 Step 5. Update d_k by (3).
 Step 6. If $\|g_k\| \leq \varepsilon$, where $\varepsilon = 10^{-6}$ terminate. Otherwise, go to step 2 with $k := k + 1$.

The following assumptions which are widely used on the objective function, would be used to analyse the global convergence of the proposed hybrid CG method.

Assumption I. $f(x)$ is bounded from below on the level set $\Omega = \{x \in R^n / f(x) \leq f(x_0)\}$.

Assumption II. In some neighborhood N of Ω , f is smooth and $g(x)$ is Lipchitz continuous in N , such that, there exist $L > 0$ (constant) satisfying;

$$\|g(x) - g(y)\| \leq L\|x - y\| \quad \forall x, y \in N. \tag{15}$$

An interesting property of the proposed β_k^{hSM*} algorithm is given by:

Lemma 2.1. The New β_k^{hSM*} satisfies

$$0 \leq \beta_k^{hSM*} \leq \frac{\|g_k\|^2}{\|d_{k-1}\|^2}. \tag{16}$$

Proof. from (10), (13), and (14), it follows that:

$$\min\{\beta_k^{RMIL}, \beta_k^{hSM}\} = \left\{ \frac{g_k^T(g_k - g_{k-1})}{d_{k-1}^T d_{k-1}}, \frac{g_k^T(g_k + g_{k-1})}{d_{k-1}^T d_{k-1}} \right\}$$

Applying (11), we have

$$\min\{\beta_k^{RMIL}, \beta_k^{hSM}\} \leq \left\{ \frac{g_k^T g_{k-1}}{d_{k-1}^T d_{k-1}}, \frac{g_k^T (g_k + g_{k-1})}{d_{k-1}^T d_{k-1}} \right\} \leq \frac{\|g_k\|^2}{\|d_{k-1}\|^2}. \tag{17}$$

Therefore, from (17) and (13), it is obvious that (17) holds true and thus, completes the proof.
Global Convergence Properties of β_k^{hSM*}

Under assumptions (I) and (II), we have the following lemma known as Zoutendijk condition which is very useful in the convergence analysis of any CG method.

Lemma 3.1. Suppose that Assumption (I) and (II) holds true. For any CG method of the form (2), (3) where α_k satisfies the strong Wolfe conditions (4) and (6). Then,

$$\sum_{k=0}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < \infty. \tag{18}$$

The proof of the Lemma follows from Zoutendijk [20].

The following lemma would be used to show that the proposed method possesses the sufficient descent properties under SWP conditions.

Lemma 3.2. Consider the Algorithm 2.1, where α_k satisfies the strong Wolfe conditions (4) and (6) with $0 < \sigma < 1/2$. Then:

$$-\sum_{j=0}^k \sigma^j \leq \frac{g_k^T d_k}{\|g_k\|^2} \leq -2 + \sum_{j=0}^k \sigma^j, \forall k \geq 1 \tag{19}$$

Proof. The proof of this lemma is by induction. For $k = 0$, we have $d_0 = -g_0$. But, $0 < \sigma < 1/2$, then:

$$-1 - \sigma \leq \frac{g_0^T d_0}{\|g_0\|^2} \leq -1 + \sigma.$$

which implies (19) is true for $k = 0$, and so, we can write $g_0^T d_0 = -\|g_0\|^2$. Therefore, based on (8), we deduce that the sufficient descent condition holds. Next, we need to show that this also holds true for $k > 0$. Now, suppose that (19) is true for $k = n, n \in N$, then, we obtain:

$$-\sum_{j=0}^n \sigma^j \leq \frac{g_n^T d_n}{\|g_n\|^2} \leq -2 + \sum_{j=0}^n \sigma^j \tag{20}$$

We will prove that (19) is true for $k = n + 1$. From (3), we get:

$$d_n = -g_n + \beta_n^{hSM*} d_{n-1}$$

Multiplying both side by g_n^T produces

$$g_n^T d_n = -g_n^T g_n + \beta_n^{hSM*} g_n^T d_{n-1} \quad (21)$$

From the strong Wolfe condition and (16), we obtain:

$$-\sigma \beta_n^{hSM*} |g_{n-1}^T d_{n-1}| \leq \beta_n^{hSM*} g_n^T d_{n-1} \leq \sigma \beta_n^{hSM*} |g_{n-1}^T d_{n-1}| \quad (22)$$

From (21) and (22), and applying the Cauchy Schwartz inequality, we have

$$-\|g_n\|^2 - \sigma \beta_n^{hSM*} \|g_{n-1}^T\| \|d_{n-1}\| \leq g_n^T d_n \leq -\|g_n\|^2 + \sigma \beta_n^{hSM*} \|g_{n-1}^T\| \|d_{n-1}\| \quad (23)$$

Dividing through by $\|g_n\|^2$ and applying (16) we get

$$-1 - \sigma \frac{\|g_{n-1}^T\|}{\|d_{n-1}\|} \leq \frac{g_n^T d_n}{\|g_n\|^2} \leq -1 + \sigma \frac{\|g_{n-1}^T\|}{\|d_{n-1}\|} \quad (24)$$

Then it follows from (20) that

$$-\sum_{j=0}^n \sigma^j \leq \frac{g_n^T d_n}{\|g_n\|^2} \leq -2 + \sum_{j=0}^n \sigma^j. \quad (25)$$

This completes the proof.

Theorem 3.1 Suppose that Assumptions I and II hold. Consider the sequences $\{x_k\}$ and $\{d_k\}$ generated using Algorithm 1, where $\alpha_k > 0$ is satisfies the strong Wolfe line search (4) and (6), then

$$\liminf_{k \rightarrow \infty} \|g_k\| = 0 \quad (26)$$

Proof. We prove this theorem by contradiction. Assume the theorem is not true, then, there exist a constant $\psi > 0$ such that

$$\|g_k\| \geq \psi, \forall k. \quad (27)$$

From (4), we have

$$d_k + g_k = \beta_k d_{k-1}, \quad (28)$$

Squaring both side of (28) will give

$$\|d_k\|^2 = \|g_k\|^2 - 2\beta_k g_k^T d_{k-1} + (\beta_k)^2 \|d_{k-1}\|^2. \quad (29)$$

From Lemma 3.2, we have

$$|g_k^T d_k| \leq \|g_k\|^2 \frac{1}{1-\sigma}. \quad (30)$$

Then, from (29) and (30), that

$$\|d_k\|^2 \leq \|g_k\|^2 + 2\beta_k |g_k^T d_{k-1}| + (\beta_k)^2 \|d_{k-1}\|^2, \quad (31)$$

and applying (16), will reduce to

$$\begin{aligned} \|d_k\|^2 &\leq \|g_k\|^2 + 2\sigma \frac{\|g_k\|^2}{\|d_{k-1}\|^2} |g_{k-1}^T d_{k-1}| + \|d_{k-1}\|^2 \left(\frac{\|g_k\|^2}{\|d_{k-1}\|^2} \right)^2 \\ &\leq \|g_k\|^2 + \|g_k\|^2 \left(\frac{2\sigma}{1-\sigma} \right) + \|d_{k-1}\|^2 \frac{\|g_k\|^4}{\|d_{k-1}\|^4} \leq \left(\frac{1+\sigma}{1-\sigma} \right) \|g_k\|^2 + \|d_{k-1}\|^2 \frac{\|g_k\|^4}{\|d_{k-1}\|^4} \end{aligned}$$

Dividing through by $\|g_k\|^4$ and applying (27), we obtain

$$\frac{\|d_k\|^2}{\|g_k\|^4} \leq \left(\frac{1+\sigma}{1-\sigma}\right) \frac{1}{\|g_k\|^2} + \frac{1}{\|d_{k-1}\|^2} \leq \frac{(1+\sigma)}{(1-\sigma)} \frac{1}{\psi^2} + \frac{1}{\|d_{k-1}\|^2}. \tag{32}$$

But $\frac{1}{\|d_0\|^2} = \frac{1}{\|g_0\|^2}$, by using (32) continuously, we have

$$\frac{\|d_k\|^2}{\|g_k\|^4} \leq \frac{(1+\sigma)}{(1-\sigma)} \frac{k}{\psi^2} + \frac{1}{\|g_0\|^2}$$

From (27), it follows that

$$\frac{\|d_k\|^2}{\|g_k\|^4} \leq \frac{(1+\sigma)}{(1-\sigma)} \frac{k}{\psi^2} + \frac{1}{\psi^2}.$$

$$\frac{\|g_k\|^4}{\|d_k\|^2} \geq \frac{\psi^2(1-\sigma)}{k(1+\sigma)+(1-\sigma)}$$

and thus, become

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

However, this contradicts (18) in Lemma 3.1, so, (26) holds true. This completes the proof.

3. RESULTS AND DISCUSSION

In this section, some preliminary numerical results are presented based on Algorithm 1. All unconstrained optimization benchmark problems considered for the computation are from Andrei [21]. The results obtained by the proposed β_{κ}^{hSM*} method is compared with that of RMIL [15], PRP [6], [7] and TS [16] conjugate gradient methods. The problems are implemented using dimensions (DIM) ranging from $2 \leq n \leq 10,000$ to illustrate the efficiency and robustness of the proposed method. The iterations are terminated when the stopping condition $\|g_k\| \leq \epsilon$, $\epsilon = 10^{-6}$ is satisfied.

- a) $hSM *$ is a modification of RMIL method and converge globally in addition to satisfying the descent condition under strong Wolfe line search in section 3.
- b) The RMIL is an efficient variant of PRP algorithm and satisfies descent condition under exact minimization condition [15].
- c) The PRP method is one of the competent and famous CG methods that satisfy the descent condition and its global convergence proof have been discussed under various line searches [6], [7].
- d) The TS method is a combination of the classical methods of PRP and FR satisfying the descent condition and its global convergence proof was established under the strong Wolfe line searches [16].

All algorithms are coded on MATLAB (R2015a). All programs are run on a CoreI5 PC operating system. All algorithms are compared based on CPU time (in seconds) and number of iteration (NOI). In the case whereby the line search cannot find a positive step size, then, the iteration is terminated, and the point of termination is denoted as fail point. Table 1 present set of unconstrained test functions with various initial guesses, and dimensions used for this study.

We shall analyze the performance result of $hSM *$ method and compare the performance with that of the existing methods of RMIL [15], PRP [6], [7], and TS [16]. Figures 1 and 2, presents the performance results of the proposed $hSM *$ iteration number and CPU time respectively. This was done with the aid of Dolan and More [22] performance profile software.

Applying the performance profile Dolan and More, we make comparison by evaluating the performance of $hSM *$, RMIL, PRP, and TS method under SWP. Assume n_s solvers and n_p problems exist, for every p and s , the authors defined

$\tau_{p,s}$ = Computation time (NO.IT. or CPU time) necessary to solve problem p by solver s .

For all the methods, the profile presents the segment P of every defined function such that the algorithm is in the neighborhood of a factor of τ of the fastest time. The highest curvature shows the best algorithm, i.e. the algorithm with the best performance [20]. This implies, the algorithm with the solution of almost all the test functions in the shortest time. From Figure 1, we can observe that the proposed $hSM *$ algorithm does better than the RMIL and TS methods and competed with PRP method both in terms of NOI (Figure 1) and CPU time (Figure 2). These shows that $hSM *$ is competent. Thus, can be used as alternative for optimization models.

Table 1. List of test function

No	Function	Dim	Initial point	No	Function	Dim	Initial Point
1	Treccani	2	(10,10),(20,20),(25,25), (30,30)	11	Quadratic QF1	2,4,10,100,1000	(2,2),(4,4),(6,6),(8,8)
2	Zettl	2	(5,5),(15,10),(20,20),(30,30)	12	Ex - Rosenbrock	100,1000	(11,11),(13,13),(21,21),(23,23)
3	Six-hump camel	2	(-8,-8),(8,8),(-10,-10),(10,10)	13	Shalow	100, 1000	(10,10),(25,25),(30,30),(50,50)
4	Booth	2	(5,5),(10,10),(15,15),(50,50)	14	Ex Tridiagonal 1	100, 1000,10000	(12,12),(17,17),(20,20),(30,30)
5	Hager	2,4,6,10,100	(3,3),(7,7),(9,9),(11,11)	15	Ex White & Holst	100,1000,10000	(5,5),(10,10),(20,20),(50,50)
6	Ex-powell	2,4,10,100	(4,4),(7,7),(8,8),(11,11)	16	Ex Denschnb	100, 1000, 10000	(11,11),(13,13),(30,30),(50,50)
7	Ext penalty	2,4,10,100	(50,50),(60,60),(70,70),(80,80)	17	Extended Beale	100, 1000, 10000	(11,11),(13,13),(23,23),(27,27)
8	Gen Tridiagonal 1	2,4,10,100	(15,15),(30,30),(35,35),(40,40)	18	Ex - Himmelblau	1000	(15,15),(25,25),(50,50),(75,75)
9	Perturbed quadrt	2.4,10,100	(10,10),(15,15),(20,20),(30,30)	19	Ex - Himmelblau	10000	(15,15),(25,25),(50,50),(75,75)
10	Sum squares	2,4,10,100	(10,10),(20,20),(30,30),(50,50)	20	Ex-penalty QP2	1000	(11,11),(13,13),(17,17),(19,19)

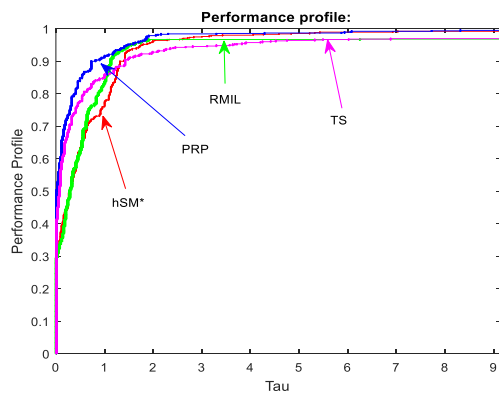


Figure 1. Performance profile based on number of iterations

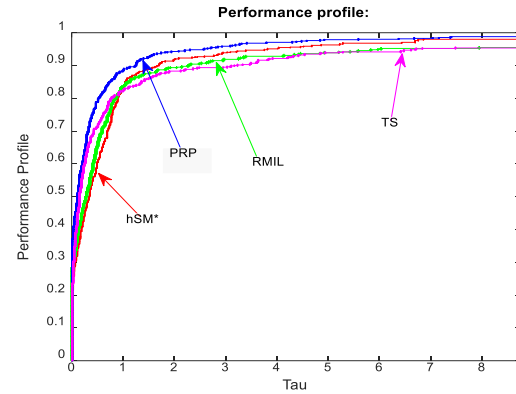


Figure 2. Performance profile based on CPU time

3.1. Application

In this subsection, we consider the global confirmed cases of Covid-19 from week one (30 Dec, 2019) to week 14 (30 Mar, 2020), and calculate the percentage weekly change of the confirmed cases [23]. The ultimate goal is to parameterize the weekly change (%) into an unconstrained optimization model which would be solved using the proposed conjugate gradient method.

Consider the regression analysis function of form:

$$y = h(x_1, x_2, \dots, x_p + \varepsilon) \tag{33}$$

where $x_i, i = 1, 2, \dots, p, p > 0$ is the predictor, y is the response variable, and ε is the error. Functions of the form (32) are often traced to the field of economics, accounting, management, finance, and many more. The regression analysis is among the most efficient statistical modeling tool for estimating the relationships between a dependent variable and one or more independent variables. The linear regression function is derived by computing for y such that:

$$y = a_0 + a_1x_1 + a_2x_2 + \dots + a_px_p + \varepsilon. \tag{34}$$

where the regression parameters are of the form a_0, a_1, \dots, a_p . These regression parameters a_0, a_1, \dots, a_p are estimated such that the error value ε is minimized. An instance of the application of linear regression method is when the relationship between y and x are approximated with a straight line. However, cases of that form rarely occur, and hence, the nonlinear regression approach is frequently employed. This study also considered the nonlinear regression method.

To derive the approximate function, we consider the weekly change (%) of confirmed cases of Covid-19 from week one (30 Dec, 2019) to week 14 (30 Mar, 2020). Table 2 present the description of the process which are considered from data obtained from World Health Organization [23]. We have data for 14 weeks (Dec 30, 2019 – Mar 30, 2020), with x - variable denoting the weeks while y -variable denotes the corresponding weekly change. However, the data for 13 weeks (Dec 6, 2019 to Mar 23, 2020) would be considered for fitting the data, while the data for week 14 (Mar 30, 2020) would be reserve for error analysis.

Table 2. Statistics of weekly change (%) of confirmed Covid-19 cases from December 6 – March 30

Weeks (x)	Confirmed Cases	Weekly Increase	Weekly Change (%) (y)
1	1	-	-
2	44	43	4.300
3	92	48	109.09
4	1,924	1,832	1.990
5	12,561	10,637	552.86
6	23,013	10,452	83.21
7	31,713	8,700	37.80
8	9,630	-22,082	-69.63
9	8,441	-1,189	-12.35
10	19,698	11,257	133.36
11	51,110	31,412	152.47
12	143,523	92,413	180.81
13	350,271	206,748	144.05
14	502,656	152,385	43.50

Next, we illustrate the weekly change of covid-19 cases in Figure 3. This represents the global cases of the pandemic for the period of December 30, 2018 to Mar 30, 2020.

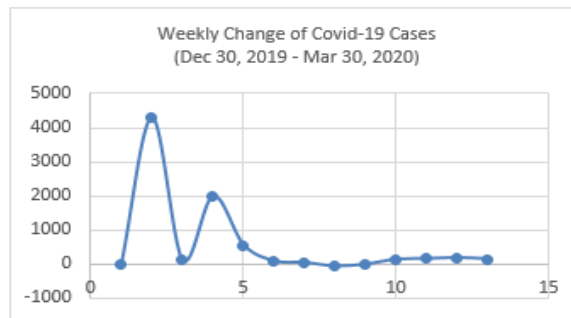


Figure 3. Weekly change of global confirmed cases of Covid-19

The approximate function for the nonlinear least square method follows from the above data

$$f(x) = -0.01444 + 31.74218x - 1.53286x^2. \tag{35}$$

Using the function (34), we need to approximate the value of y based on value of x from week 1-13. Suppose we denote the number of weeks by x_j while y_j is the corresponding weekly change of the confirmed cases, then, the above least square method (34) is transformed into the following unconstrained minimization model:

$$\min_{x \in R^n} f(x) = \sum_{j=1}^n ((u_0 + u_1x_j + u_2x_j^2) - y_j)^2. \tag{36}$$

It is obvious that data x_j and the value of y_j enjoy some parabolic relations with the regression parameters u_0, u_1 and u_2 and the regression function (36).

$$\min_{x \in R^2} \sum_{j=1}^n E_j^2 = \sum_{j=1}^n ((u_0 + u_1x + u_2x^2) - y_j)^2 \tag{37}$$

Next, we apply the data set from Table 2 to transform (36) into our desired nonlinear quadratic unconstrained minimization model as:

$$13u_1^2 + 180u_1u_2 + 1638u_1u_3 - 15203u_1 + 819u_2^2 + 16562u_2u_3 - 53601u_2 + 89271u_3^2 - 290801u_3 + 2287548 \quad (38)$$

For generating the unconstrained optimization model, we considered data from week 1 (December 30, 2019) to week 13 (March 23, 2020), while the data for week 14 (March 30, 2020) is reserved for reserved for the purpose of relative error analysis of the predicted data. By coding the parameterized model as an unconstrained optimization test function on MATLAB, and applying the proposed *hSM* * method under SWP condition, we obtain the performance results based on iteration numbers and CPU time presented in Table 3.

Table 3. Test results for the parameterized Covid-19 model using *hSM* *

Initial points	No if iteration	CPU time
(0.5,0.5,0.5)	23	0.394641448148960
(1,1,1)	14	0.112436480913270
7,7,7)	21	0.072457122307238
(13,13,13)	34	0.141150926030078

The main challenge confronted when solving the parameterized model computing the values of u_0, u_1, u_2 using matrix inverse. This challenge can be overcome using the proposed *hSM* * with four different initial points. We terminate the computation if:

- The defined stopping criteria is satisfied. This is based on value defined for each function.
- The method is unable to solve the model.

3.2. Trend line method

Trendlines are bounding lines presenting the price movement of securities. It is often seen as visual representation of resistance and support in a time frame. Application of trend line can be traced to area of social science [24], [25]. The trendline is formed by drawing a diagonal line amid a minimum of three, four, or even more price pivot points thereby showing the speed and direction of price, at the same time describing the patterns during periods of price contraction. This section tends to estimate the global weekly change of confirmed cases of COVID-19 for a period of 14 weeks (Dec 30, 2019–30 Mar, 2020 using the and least squares and *hSM* * methods. Based on the actual data obtained from Table 2, we employ Microsoft Excel software to plot the trend line as presented in Figure 4. From the plot, it is obvious that the trend line equation obtained is in form of nonlinear quadratic equation. The regression analysis primary purpose is for estimating a_0, a_1, \dots, a_p whereby the error ε is minimized. From the discussion, we can conclude that the *hSM* * method can be used as alternative to the Trendline and Least square methods.

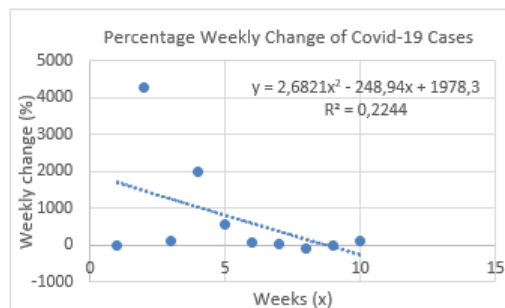


Figure 4. Nonlinear quadratic trend line for weekly change of global confirmed cases of Covid-19

4. CONCLUSION

This paper proposed a variant of RMIL CG method for the solution of unconstrained optimization models. The proposed algorithm named *hSM* * enjoys the descent condition and converge globally under strong Wolfe line search. Preliminary numerical experiment on some benchmark unconstrained optimization functions have been presented to demonstrates the efficacy of our algorithm. The obtained results illustrate that *hSM* * method is effective with better convergence rate compare to the methods of RMIL, TS and competed with the PRP method. This implied that the proposed algorithm can be used as alternative for solution of unconstrained optimization models and in practical applications.

ACKNOWLEDGEMENTS

The authors are grateful to Universiti Sultan Zainal Abidin, Malaysia, for funding this research.

REFERENCES

- [1] A. U. Omesa, *et al.*, "New hybrid conjugate gradient method for solving fuzzy nonlinear equations," *J. of Advanced Res. in Dynamical and Control Systems*, vol. 12, no. 2, 585-590, 2020, doi: 10.5373/JARDCS/V12I2/S20201081.
- [2] I. M. Sulaiman and M. Mamat, "A New Conjugate Gradient Method with Descent Properties and its Application to Regression analysis". *Journal of Numerical Analysis, Industrial and Applied Mathematics*, vol.12, no. 1-2, pp. 25-39, 2020.
- [3] Y. Yuan, *Numerical methods for nonlinear programming*. Shanghai: Shanghai Scientific & Technical Publishers, 1993.
- [4] M. R. Hestenes and E. L. Stiefel, "Methods of conjugate gradients for solving linear systems," *J. Research Nat. Bur. Standards*, vol. 49, no. 6, pp. 409-436, 1952, doi:10.6028/JRES.049.044.
- [5] R. Fletcher and C. Reeves, "Function minimization by conjugate gradients," *The Computer Journal*, vol. 7, no. 2, pp. 149-154, 1964, doi: 10.1093/comjnl/7.2.149.
- [6] E. Polak and G. Ribiere, "Note on the convergence of conjugate directions (in French)," *Rev. Francaise Informat Recherche Opertionelle, 3e ann´ee*, vol.3, no. R1, pp. 35-43, 1969.
- [7] B. T. Polyak, "The conjugate gradient method in extremal problems," *USSR Comp. Math. Math. Phys.*, vol. 9, no. 4, pp. 94-112, 1969, doi: 10.1016/0041-5553(69)90035-4.
- [8] Y. H. Dai and Y. Yuan Y., "A nonlinear conjugate gradient with strong global convergence properties," *SIAM Journal on Optimization*, vol. 10, pp. 177-182, 1999, doi: 10.1137/S1052623497318992.
- [9] M. Al-Baali, "Descent property and global convergence of the Fletcher-Reeves method with inexact line search", *IMA J. Numerical Analysis.*, vol. 5, no. 1, pp. 121-124, 1985, doi: 10.1093/imanum/5.1.121.
- [10] M. Dawahdeh, I. M. Sulaiman, M. Rivaie, and M. Mamat, "A new spectral conjugate gradient method with strong wolfe-powell line search", *International Journal of Emerging Trends in Engineering Research*, vol. 8, no. 2, pp. 391-397, 2020, doi: 10.30534/ijeter/2020/25822020.
- [11] M. J. D. Powell, "Nonconvex minimization calculations and the conjugate gradient method". *Lecture Notes in Mathematics*, vol. 1066 (Berlin: Springer-Verlag), 122-141, 1984, doi: 10.1007/BFb0099521.
- [12] M. J. D. Powell, "Restart procedures for the conjugate gradient method". *Mathematical Programming*, vol. 12, pp. 241-254, 1985, doi: 10.1007/BF01593790.
- [13] I. M. Sulaiman, M. Mamat., A. E. Owoyemi, P. L. Ghazali, M. Rivaie, and M. Malik, "The convergence properties of some descent conjugate gradient algorithms for optimization models". *J. Math. Computer Sci.* vol. 22, no. 3, 204-215, 2020, doi: 10.22436/jmcs.022.03.02.
- [14] M. Jiangtao, G. Nengzhu, and W. Zengxin, "Hybrid conjugate gradient methods for unconstrained optimization," *Optimization method and Software*, vol. 22, no. 2, pp. 297-307, 2007, doi: 10.1080/10556780500518653.
- [15] M. Rivaie, M. Mamat, L.W. June, and I. Mohd, "A new class of nonlinear conjugate gradient coefficients with global convergence properties," *Applied Mathematics and Computation*, vol. 218, no. 22, pp. 11323-11332, 2012, doi: 10.1016/j.amc.2012.05.030
- [16] D. Touati-Ahmed and C. Storey, "Efficient hybrid conjugate gradient techniques," *Journal of optimization theory and applications*, vol. 64, pp. 379-397, 1990, doi: 10.1007/BF00939455.
- [17] Y. H. Dai and Y. Yuan, "An efficient hybrid conjugate gradient method for unconstrained optimization," *Annals of Operations Research*, vol. 103, pp. 33-47, 2001, doi: 10.1023/A:1012930416777
- [18] A. B. Auwal, P. Kumam, M. Malik, P. Chaipunya, and I. H. Abdulkarim, "A hybrid FR-DY conjugate gradient algorithm for unconstrained optimization with application in portfolio selection," *AIMS Mathematics*, vol. 6, no. 6, pp. 6506-6527, 2021, doi: 10.3934/math.2021383.
- [19] N. J. Fanar and M. Al-Naemi, "A new hybrid conjugate gradient algorithm for unconstrained optimization with inexact line search," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 20, no. 2, pp. 939-947, doi: 10.11591/ijeecs.v20.i2.pp939-947.
- [20] G. Zoutendijk, "Nonlinear programming, computational methods, in Integer and Nonlinear Programming," *J. Abadie, ed.*, North-Holland, Amsterdam, pp. 37-86, 1970.
- [21] N. Andrei, "An unconstrained optimization test functions collection," *Advanced Modeling and Optimization*", vol. 10, no. 1, pp. 147-161, 2008.
- [22] E. D. Dolan and J. J. Moré, "Benchmarking optimization software with performance profiles," *Mathematical Programming*, vol. 91, pp. 201-213, 2002, doi: 10.1007/s101070100263
- [23] World health organization, (*Report on coronavirus (Covid-19)*). [Online]. Available: <https://covid19.who.int/>. Accessed Nov. 2020.
- [24] N. Abu Bakar and M.S. Al-Smadi, "Enhancing the Affordability of Cognitive Behavioral Therapy for Children". *International Journal of Psychosocial Rehabilitation*, vol. 24, no. 6, pp. 2218-2231, 2020, doi:10.37200/IJPR/V24I6/PR260212
- [25] N. Abu Bakar and I. Z. Bajuri, "Understanding Special education Teachers' Perspectives on Professionalism in Educating Autistic Students," *International Journal of Advanced Science and Technology*, vol 29, no. 1, pp. 34-41, 2020.