RESEARCH ARTICLE

Journal of Global Humanities and Social Sciences
2024, Vol. 5(10) 364–371
DOI: 10.61360/BoniGHSS242016851001

A Hybrid Conjugate Gradient Algorithm for Nonlinear System of Equations through Conjugacy Condition

Aliyu Yusuf[1,2,3,*], Abdullahi Adamu Kiri[3,4], Lukman Lawal[3,5] and Aliyu Ibrahim Kiri[3]

[1]Department of Science, Bayero University Kano, Nigeria

[2]Department of Mathematical Sciences, Abubakar Tafawa Balewa University Bauchi, Nigeria

[3]Department of Mathematical Sciences, Bayero University, Nigeria

[4]Jigawa State Institute of Technology, Nigeria

[5]Department of Statistics, Nuhu Bamalli Polytechnic, Nigeria

Abstract: For the purpose of solving a large-scale system of nonlinear equations, a hybrid conjugate gradient algorithm is introduced in this paper, based on the convex combination of β_k^{FR} and β_k^{PRP} parameters. It is made possible by incorporating the conjugacy condition together with the proposed conjugate gradient search direction. Furthermore, a significant property of the method is that through a non-monotone type line search it gives a descent search direction. Under appropriate conditions, the algorithm establishes its global convergence. Finally, results from numerical tests on a set of benchmark test problems indicate that the method is more effective and robust compared to some existing methods.

AMS codes: 65K05, 90C52, 90C26

Keywords: conjugate gradient parameters, convex combination, conjugacy condition, global convergence, numerical experiments

1. Introduction

The generic form of a system of nonlinear equations is:

$$F(x) = 0; \quad x \in R^n; \tag{1}$$

where the mapping $F : R^n \to R^n$ is nonlinear continuously differentiable function. Science and engineering are two disciplines that the nonlinear system of equations plays a significant role. As a result, scholars in this area are now interested and quite a number of methods have been devised including Newton's method and quasi-Newton's method to solve (1), see Halilu and Waziri (2017); Waziri et al. (2010); Fukushima and Li (1999); Dauda et al. (2019), for more details. However, the two methods are costly (not friendly) for solving large-scale nonlinear systems, since the Jacobian matrix needs to be stored and computed at every iteration, or its approximation (Waziri & Sabiu, 2015). The conjugate gradient (CG) method, which is most frequently used to solve large-scale unconstrained optimization problems, is the well-known methods for finding approximate solutions to large-scale nonlinear systems, because it has strong global convergence properties, low memory requirement and simple to implement (Dai & Yuan, 1999, Waziri, Yusuf & Abubakar, 2020).

Mostly, the nonlinear CG method is implemented via the following form:

$$\min f(x); \quad x \in R^n; \tag{2}$$

the function $f : R^n \to R$ is continuously differentiable. Using the following iterative formula, it produces an iterative sequence $\{x_k\}$ starting from a given initial point $x_1 \in R^n$.

$$x_{k+1} = x_k + \alpha_k d_k; \quad k = 1; 2; 3; \ldots \tag{3}$$

where x_k is the k^{th} approximation to the solution of (2), using a suitable line search technique, the step-length $\alpha_k > 0$ is computed and conjugate gradient search direction d_k is calculated by

$$d_k = \begin{cases} -F_k, & if \quad k = 0, \\ -F_k + \beta_k d_{k-1}, & if \quad k \geq 1, \end{cases} \tag{4}$$

where a scalar β_k is known as CG update parameter, F_k is the gradient of f_k at x_k, that is $F_k = \nabla f(x_k)$. Moreover, the update parameter β_k is the most important component of any CG method. As such, different CG methods have been proposed corresponding to different choices of β_k

*Corresponding author: Aliyu Yusuf, Department of Science, Bayero University Kano, Nigeria. Email: ayusuf.sce@buk.edu.ng

[see Hager and Zhang (2006); Dai and Yuan (1999); Polak and Ribiere (1969); Liu and Storey (1991)]. Among the update parameters we have is the Fletcher-Reeves (FR) introduced in 1964, see Fletcher and Reeves (1964) for details, given by:

$$\beta_k^{FR} = \frac{\| F(x_{k+1}) \|^2}{\| F(x_k) \|^2}; \tag{5}$$

Likewise, the Polak-Ribiere and Polyak (PRP) is another type of CG update parameter established in 1969 (Polak & Ribiere, 1969), defined as:

$$\beta_k^{PRP} = \frac{F^T(x_{k+1})y_k}{\| F(x_k) \|^2}. \tag{6}$$

The performance of hybrid CG methods has been shown to be better than classical CG methods when solving nonlinear equations. For instance, the papers by Babaie-Kafaki et al. (2011), Andrei (2008, 2009, 2007), Djordjevic (2016) and, recently, Ioannis et al. (2018) presented different types of hybrid CG methods via convex combination approach. Furthermore, hybrid CG methods are severally used for solving (2), but not much have been proposed to solve equation (1).

This article is focused on a hybrid conjugate gradient algorithm (HCGA) via conjugacy condition for large-scale nonlinear systems of equations. The article is structured as follows: The derivation of the method is presented in Section 2. The algorithm has been shown to be globally converged in Section 3. Section 4 reports the numerical experiment on some set of benchmark test problems. Finally, the conclusion is given in Section 5.

Motivated by the idea of convex combination's approach presented by Ioannis et al. (2018), in our research, we propose a HCGA to solve (1) via conjugacy condition based on the convex combination's technique

Notation: Throughout the research, we have utilized $\|.\|$ to represent the Euclidean norm of vectors, $y_k = F_{k+1} - F_k$, $s_k = x_{k+1} - x_k$, $f_k = f(x_k)$, $\nabla f(x_k) = F(x_k)$ and $F_k = F(x_k)$. We however assume that the function (1) is Lipschitz continuous, f in (2) is defined by:

$$f(x) = \frac{1}{2}\| F(x) \|^2. \tag{7}$$

2. Derivation of the Method

The suggested algorithm is deduced in this part, and it produces a sequence of iterates, $x_1, x_2, x_3, \ldots$, by using the recurrence relation (3), $\alpha_k > 0$ is obtained via a non-monotone type line search (Fukushima & Li, 1999), and from (4), our proposed descent search direction d_k is given by:

$$d_{k+1}^{HCGA} = -\left(1 + \frac{\beta_k^{HCGA} F_{k+1}^T d_k}{\| F_{k+1} \|^2}\right) F_{k+1} + \beta_k^{HCGA} d_k; \quad \forall k \geq 1; \tag{8}$$

where

$$\beta_k^{HCGA} = \lambda_k \beta_k^{FR} + (1 - \lambda_k)\beta_k^{PRP}; \quad \lambda_k \in [0;1]. \tag{9}$$

By considering (5) and (6), (9) can be expressed as:

$$\beta_k^{HCGA} = \lambda_k \frac{\| F_{k+1} \|^2}{\| F_k \|^2} + (1 - \lambda_k)\frac{F_{k+1}^T y_k}{\| F_k \|^2}. \tag{10}$$

From (8) and (10), we have

$$d_{k+1}^{HCGA} = -\left(1 + \left(\lambda_k \frac{\| F_{k+1} \|^2}{\| F_k \|^2} + (1-\lambda_k)\frac{F_{k+1}^T y_k}{\| F_k \|^2}\right)\frac{F_{k+1}^T d_k}{\| F_{k+1} \|^2}\right) F_{k+1} + \lambda_k \frac{\| F_{k+1} \|^2}{\| F_k \|^2} d_k + (1-\lambda_k)\frac{F_{k+1}^T y_k}{\| F_k \|^2} d_k. \tag{11}$$

We use the conjugacy condition to get the hybrid parameter λ_k, and the conjugacy condition for nonlinear conjugate gradient methods is provided by:

$$y_k^T d_{k+1} = 0. \tag{12}$$

Multiply (11) by y_k^T, using (12) after simplification, we have:

$$\lambda_k^* = \frac{\| F_{k+1} \|^2 \left((y_k^T F_{k+1})^2 (F_{k+1}^T s_k) + \| F_k \|^2 (y_k^T F_{k+1}) - (y_k^T F_{k+1})(y_k^T s_k) \right)}{\| F_{k+1} \|^2 \left(\| F_{k+1} \|^2 (y_k^T s_k) + (F_{k+1}^T s_k)(y_k^T F_{k+1}) - (y_k^T F_{k+1}) \right) + (y_k^T F_{k+1})^2 (F_{k+1}^T s_k)}. \tag{13}$$

However, from (8), the condition of a descent search direction holds as follows:

$$F_{k+1}^T d_{k+1}^{HCGA} \leq -\| F_{k+1} \|^2. \tag{14}$$

To compute α_k, we apply the method presented in Fukushima and Li (1999). Suppose that $\omega_1 > 0$, $\omega_2 > 0$ and $r \in (0,1)$. Let also $\{\eta_k\}$ be a given positive sequence such that

$$\sum_{k0} \eta_k < \eta < \infty. \tag{15}$$

Hence, α_k is computed as follows:

$$f_{k+1} - f_k \leq -\omega_1 \| \alpha_k F_k \|^2 - \omega_2 \| \alpha_k d_k \|^2 + \eta_k f_k; \text{ where}; \omega_1; \omega_2 > 0. \tag{16}$$

Let i_k be the lowest positive integer i such that (16) hold for $\alpha = r^i$ and suppose that $\alpha_k = r^{i_k}$.

Algorithm of HCGA Method

Step 1: Given $x_0 \in R^n$, $\alpha_0 > 0$, $\varepsilon = 10^{-4}$, $d_0 = -F_0$, set $k = 0$.
Step 2: Compute F_k.
Step 3: If $\| F_k \| \leq \varepsilon$, then stop, else go to Step4.
Step 4: Compute α_k using (16).
Step 5: Set $x_{k+1} = x_k + \alpha_k d_k$.
Step 6: Compute F_{k+1}.
Step 7: Compute d_{k+1}^{HCGA} using (8), (5), (6) and (13).

Furthermore, the values of λ_k^* obtained from (13) are restricted in the interval [0,1]; if λ_k^* is greater than one (1), then we set λ_k^* to be equal to one (1); if λ_k^* is less than zero (0), then we set λ_k^* to be zero (0), and we have a proper convex combination of FR and PRP parameters, if λ_k^* is between 0 and 1.

Step 8: Set $k = k + 1$ and repeat from step 3.

Table 1
The test problems with their references are listed in the table below

S/N	Problem and reference
1	Problem Number 1 in Jamilu etal. (2017)
2	Problem Number 12 in Jamilu etal. (2017)
3	Problem Number 15 in Jamilu etal. (2017)
4	Problem Number 7 in Halilu and Waziri (2017)
5	Problem Number 6 in Halilu and Waziri (2017)
6	Problem Number 10 in Halilu and Waziri (2017)
7	Problem Number 16 in Jamilu etal. (2017)
8	Problem Number 1 in Waziri and Sabiu (2015)
9	Problem Number 1in Dauda et al. (2016)
10	Problem Number 7 in Waziri and Sabiu (2015)
11	Problem Number 8 in Halilu and Waziri (2017)
12	Problem Number 4 in Waziri and Sabiu (2015)
13	Problem Number 9 in Halilu and Waziri (2017)
14	Problem Number 6 in Waziri and Sabiu (2015)
15	Problem Number 1 in Halilu and Waziri (2017)
16	Problem Number 8 in Dauda et al. (2016)
17	Problem Number 7 in Dauda et al. (2016)
18	Problem Number 5 in Dauda et al. (2016)
19	Problem Number 2 in Halilu and Waziri (2017)
20	Problem Number 2 in Dauda et al. (2016)

3. Convergence Results

The suggested algorithm has proved to be globally converged in this section. Under the following assumptions, the convergence result of the HCGA algorithm is shown.

Assumption 3.1. The set,

$$\Omega = \{x \in R^n \mid \|F(x)\| \leq \|F(x_0)\|\}; \tag{17}$$

is bounded, meaning that there exists a non-negative constant B, such that

$$\|x\| \leq B; \quad \forall x \in \Omega. \tag{18}$$

Assumption 3.2.
(1) There exists $x^* \in R^n$, such that $F(x^*) = 0$.
(2) F is a mapping of differentiable continuous functions.

Assumption 3.3. The continuous function F is Lipschitz. Meaning that $\forall x,y \in \Omega$,

$$\|F(x) - F(y)\| \leq L\|x - y\|; \quad L > 0. \tag{19}$$

Furthermore, it implies that, by Assumptions 3.1 and 3.3, there is a non-negative constant M such that

$$\|F(x)\| \leq M; \quad \forall x \in \Omega. \tag{20}$$

Lemma 1: If a sequence $\{x_k\}$ is generated by the algorithm HCGA, then a direction d_k for F_k at x_k is a descent. That is,

$$F_{k+1}^T d_{k+1} < 0; \quad \forall k \geq 1. \tag{21}$$

Proof For $k = 1$, we have $F_1^T d_1^{HCGA} \| = - \| F_1 \| < 0$. For $k > 1$,

$$F_{k+1}^T d_{k+1}^{HCGA} =$$
$$-\| F_{k+1} \|^2 - \| F_{k+1} \|^2 \beta_k^{HCGA} F_{k+1}^T d_k + \| F_{k+1} \|^2 \beta_k^{HCGA} F_{k+1}^T d_k. \tag{22}$$

Therefore, from (22), we get

$$F_{k+1}^T d_{k+1}^{HCGA} = - \| F_{k+1} \|^2 . \tag{23}$$

Which shows that

$$F_{k+1}^T d_{k+1}^{HCGA} < 0. \tag{24}$$

Lemma 2: If Assumptions 3.1 and 3.3 are met, let the sequence $\{x_k\}$ be generated by the algorithm HCGA. If $m > 0$ such that

$$\|F_k\|^2 \geq m; \tag{25}$$

then,

$$|\beta_k^{HCGA}| \leq \frac{M}{m}(M + 4LB) := \mu. \tag{26}$$

Proof From (10), we have

$$\beta_k^{HCGA} = \lambda_k \beta_k^{FR} + (1 - \lambda_k)\beta_k^{PRP}; \text{ where } \lambda_k \in [0;1]; \forall k. \tag{27}$$

Using (5) and (6), (27) becomes

$$\beta_k^{HCGA} = \lambda_k \frac{\| F_{k+1} \|^2}{\| F_k \|^2} + \frac{F_{k+1}^T y_k}{\| F_k \|^2} - \lambda_k \frac{F_{k+1}^T y_k}{\| F_k \|^2}. \tag{28}$$

When the absolute value is taking from both side of (28), we have:

$$|\beta_k^{HCGA}| \leq |\lambda_k| \frac{\| F_{k+1} \|^2}{\| F_k \|^2} + \frac{|F_{k+1}^T y_k|}{\| F_k \|^2} + |\lambda_k| \frac{|F_{k+1}^T y_k|}{\| F_k \|^2}. \tag{29}$$

Applying Cauchy Schwartz inequality to (29), we have

$$|\beta_k^{HCGA}| \leq |\lambda_k| \frac{\| F_{k+1} \|^2}{\| F_k \|^2} + \frac{\| F_{k+1} \| \| y_k \|}{\| F_k \|^2} + |\lambda_k| \frac{\| F_{k+1} \| \| y_k \|}{\| F_k \|^2}. \tag{30}$$

From Assumption 3.3 and (25), it follows that

$$|\beta_k^{HCGA}| \leq |\lambda_k| \frac{M^2}{m} + \frac{LM \| s_k \|}{m} + |\lambda_k| \frac{LM \| s_k \|}{m}. \tag{31}$$

Rearranging (31), we have

$$|\beta_k^{HCGA}| \leq \frac{M}{m}(|\lambda_k| M + L \| s_k \| (1 + |\lambda_k|)). \tag{32}$$

Thus, by the boundedness of λ_k and Assumption 3.1, we have

$$|\beta_k^{HCGA}| \leq \frac{M}{m}(M + 4LB) := \mu. \tag{33}$$

Lemma 3: If Assumptions (3.1) and (3.3) are satisfied, let the sequence $\{d_k\}$ be produced by the algorithm HCGA. Then,

$$\| d_{k+1}^{HCGA} \| \le M\left(\frac{1}{1-2\mu}\right) + (2\mu)^k \| F_1 \|;\ \forall k. \quad (34)$$

Proof From (8), we have

$$\| d_{k+1}^{HCGA} \| = \left\| -\left(1 + \beta_k^{HCGA} \frac{F_{k+1}^T d_k^{HCGA}}{\| F_{k+1} \|^2}\right) F_{k+1} + \beta_k^{HCGA} d_k^{HCGA} \right\|. \quad (35)$$

Applying triangle inequality on (35) together with (20) and (33), we get

$$\begin{aligned} \| d_{k+1}^{HCGA} \| &\le \left\| \left(1 + \beta_k^{HCGA} \frac{F_{k+1}^T d_k^{HCGA}}{\| F_{k+1} \|^2}\right) F_{k+1} \right\| + |\beta_k^{HCGA}| \, \| d_k^{HCGA} \| \\ &\le \| F_{k+1} \| + \frac{|\beta_k^{HCGA}| \, \| F_{k+1} \|^2 \| d_k^{HCGA} \|}{\| F_{k+1} \|^2} \\ &+ |\beta_k^{HCGA}| \, \| d_k^{HCGA} \| = \| F_{k+1} \| + 2|\beta_k^{HCGA}| \, \| d_k^{HCGA} \| \\ &\le M + 2\mu \| d_k^{HCGA} \| \end{aligned} \quad (36)$$

Now, for $k = 1$, $\| d_1^{HCGA} \| = \| F_1 \|$, which implies

$$\| d_2^{HCGA} \| \le M + 2\mu \| F_1 \|.$$

For $k = 2$, we have $\| d_3^{HCGA} \| \le M + 2\mu \| d_2^{HCGA} \|$. This implies

$$\| d_3^{HCGA} \| \le M + 2\mu(M + 2\mu \| F_1 \|) = M(1 + 2\mu) + (2\mu)^2 \| F_1 \|.$$

For $k = 3$, we have $\| d_4^{HCGA} \| \le M + 2\mu \| d_3^{HCGA} \|$. This implies

$$\begin{aligned} \| d_4^{HCGA} \| &\le M + 2\mu(M(1 + 2\mu) + (2\mu)^2 \| F_1 \|) \\ &= M(1 + 2\mu + 4\mu^2) + (2\mu)^3 \| F_1 \|. \end{aligned}$$

`For $k = 4$, we have $\| d_5^{HCGA} \| \le M + 2\mu \| d_4^{HCGA} \|$. This implies

$$\begin{aligned} \| d_5^{HCGA} \| &\le M + 2\mu(M(1 + 2\mu + 4\mu^2) + (2\mu)^3 \| F_1 \|) \\ &= M(1 + 2\mu + 4\mu^2 + 8\mu^3) + (2\mu)^4 \| F_1 \|. \end{aligned}$$

Therefore,

$$\begin{aligned} \| d_{k+1}^{HCGA} \| &\le M(1 + 2\mu + (2\mu)^2 + (2\mu)^3 + \ldots + (2\mu)^{k-1}) \\ &+ (2\mu)^k \| F_1 \|. \end{aligned}$$

Since $\mu > 0$, we can chose $\mu \in (0; \frac{1}{2})$ such that $2\mu \in (0,1)$. This makes the series $1 + 2\mu + (2\mu)^2 + (2\mu)^3 + \ldots + (2\mu)^{k-1}$ a geometric series. Hence,

$$\| d_{k+1}^{HCGA} \| \le M\left(\frac{1}{1-2\mu}\right) + (2\mu)^k \| F_1 \|. \quad (37)$$

Lemma 4: If Assumption (3.3) is met and the sequence $\{x_k\}$ is produced by the algorithm HCGA. Then, we have:

$$\lim_{k\to\infty} \| \alpha_k d_k \|^2 = 0; \quad (38)$$

and

$$\lim_{k\to\infty} \| \alpha_k F(x_k) \|^2 = 0. \quad (39)$$

Proof From (7) and (16), for all $k > 0$, we have:

$$\begin{aligned} \omega_2 \| \alpha_k d_k \|^2 &\le \omega_1 \| \alpha_k F(x_k) \|^2 + \omega_2 \| \alpha_k d_k \|^2; \\ &\le \| F(x_k) \|^2 - \| F(x_{k+1}) \|^2 + \eta_k \| F(x_k) \|^2. \end{aligned} \quad (40)$$

By summing the relation (40) and using (20), we have

$$\begin{aligned} \omega_2 \sum_{i\geq 0} \| \alpha_i d_i \|^2 &\le \sum_{i\geq 0} (\| F(x_i) \|^2 - \| F(x_{i+1}) \|^2) + \sum_{i\geq 0} \eta_i \| F(x_i) \|^2; \\ &= \| F(x_0) \|^2 - \| F(x_{k+1}) \|^2 + \sum_{i\geq 0} \eta_i \| F(x_i) \|^2; \\ &\le \| F(x_0) \|^2 + \| F(x_0) \|^2 \sum_{i\geq 0} \eta_i; \\ &\le M^2 + M^2 \sum_{i\geq 0} \eta_i. \end{aligned} \quad (41)$$

Since $\{\eta_k\}$ satisfies (15), then by Assumption (3.3) the series $\sum_{i\geq 0} \| \alpha_i d_i \|^2$ is convergent, which implies (38). By same arguments as the above but with $\omega_1 \| \alpha_k F(x_k) \|^2$ on the left-hand side, we obtain (39).

Theorem : If Assumptions (3.1), (3.2) and (3.3) are met, let the sequence $\{x_k\}$ be produced by the algorithm HCGA.

Then,

$$\liminf_{k\to\infty} \| F_k \| = 0. \quad (42)$$

Proof Case 1. If

$$\liminf_{k\to\infty} \| d_k \| = 0. \quad (43)$$

Then, by definition of the direction, we have

$$\liminf_{k\to\infty} \| F_k \| = 0. \quad (44)$$

Case 2. If

$$\liminf_{k\to\infty} \| d_k \| > 0. \quad (45)$$

Then, we have

$$\liminf_{k\to\infty} \| F_k \| > 0. \quad (46)$$

By (39), we obtain

$$\lim_{k\to\infty} \alpha_k = 0. \quad (47)$$

Using (7) and (16), we get

$$\| F_{k+1} \|^2 - \| F_k \|^2 \le \omega_1 \| \alpha_k F_k \|^2 - \omega_2 \| \alpha_k d_k \|^2 + \eta_k \| F_k \|^2. \quad (48)$$

Suppose by contradiction that (48) does not hold, this means that there exists a non-negative integer $i - 1$ such that

Table 2
Numerical experiments of HCGA, NHCG and ICGB algorithms for problems 1–10

Problems	Dim	HCGA			NHCG			ICGB		
		NI	Time (s)	‖F(x)‖	NI	Time (s)	‖F(x)‖	NI	Time (s)	‖F(x)‖
1	1000	2	0.106715	2.01E-05	3	0.100021	2.15E-09	3	0.110477	2.16E-09
	10,000	2	0.603175	3.00E-07	4	0.619942	4.45E-10	3	0.630133	6.82E-09
	100000	2	2.999795	3.31E-05	4	3.943696	3.17E-05	3	3.092768	2.16E-08
2	1000	5	0.163135	2.30E-05	1928	2.932383	9.99E-05	7	0.171407	2.40E-05
	10,000	6	0.557942	7.60E-05	–	–	–	7	0.771150	7.60E-05
	100000	9	3.274933	4.82E-05	475	141.7933	9.99E-05	9	3.483769	4.82E-05
3	1000	2	0.118804	1.48E-05	2	0.467551	1.48E-05	3	0.157118	4.48E-06
	10,000	2	0.524767	2.23E-05	2	0.555511	2.23E-05	4	0.642536	1.65E-08
	100000	2	2.706808	9.34E-05	3	2.967292	1.04E-08	4	2.833617	5.31E-09
4	1000	2	0.076110	6.51E-05	2	0.097069	7.10E-05	3	0.116186	2.84E-07
	10,000	2	0.527870	1.80E-05	4	0.709516	9.66E-07	3	0.564919	3.30E-06
	100000	2	2.761127	4.40E-05	5	3.822294	3.91E-06	3	2.865196	7.81E-05
5	1000	2	0.086872	2.24E-08	3	0.453638	2.24E-08	3	0.105180	2.24E-08
	10,000	2	0.393192	7.10E-08	3	0.630836	7.17E-08	3	0.496078	7.10E-08
	100000	2	2.423738	2.24E-07	5	3.198797	8.35E-14	3	2.797096	2.24E-07
6	1000	2	0.060065	3.75E-06	2	0.132127	4.51E-07	4	0.114342	4.51E-07
	10,000	2	0.055383	1.67E-06	2	0.637730	1.43E-10	5	0.139075	1.43E-10
	100000	3	2.363474	6.81E-06	3	3.120356	6.81E-06	6	2.733970	6.81E-06
7	1000	7	0.124505	2.84E-05	10	0.300784	7.59E-05	8	0.168446	2.84E-05
	10,000	7	0.760381	8.97E-05	20	9.223515	5.06E-05	8	0.772320	8.97E-05
	100000	8	2.366610	7.51E-05	15	56.07195	3.61E-05	9	2.964504	7.51E-05
8	1000	6	0.133922	3.61E-05	142	0.609758	9.53E-10	7	0.164786	5.79E-05
	10,000	7	0.636311	2.28E-05	121	1.771815	1.11E-09	8	0.735432	3.66E-05
	100000	7	3.013609	7.21E-05	101	11.14406	6.90E-07	9	3.638422	2.32E-05
9	1000	10	0.181855	6.77E-05	17	0.216867	1.00E-04	11	0.569179	6.77E-05
	10,000	11	0.774165	8.07E-05	15	1.526369	8.52E-05	12	0.808575	8.07E-05
	100000	12	3.447299	9.61E-05	14	10.02003	7.29E-05	13	3.774406	9.61E-05
10	1000	3	0.031149	6.92E-06	7	0.178253	1.54E-05	3	0.067513	6.39E-08
	10,000	3	0.422564	3.32E-05	–	–	–	3	0.596332	6.55E-07
	100000	5	3.136436	8.76E-05	–	–	–	4	3.117042	8.03E-06

$$\|F_{k+1}\|^2 - \|F_k\|^2 > \omega_1 \|i^{-1}F_k\|^2 - \omega_2 \|i^{-1}d_k\|^2 + \eta_k \|F_k\|^2. \quad (49)$$

Since $\{\|F_k\|\}$ and $\{\|d_k\|\}$ are bounded, then, allowing $i \to \infty$, we have

$$\|F_{k+1}\|^2 - \|F_k\|^2 > \eta_k \|F_k\|^2. \quad (50)$$

By rearranging (50), we obtain

$$\|F_{k+1}\|^2 > \left(1 + \eta_k\right) \|F_k\|^2. \quad (51)$$

By taking the summation on both side of (51), we get

$$\sum_{j=0}^{k} \|F_{j+1}\|^2 > \sum_{j=0}^{k} \left(1 + \eta_j\right) \|F_j\|^2. \quad (52)$$

From (52), we deduce that

$$\|F_1\|^2 + \|F_2\|^2 + \ldots + \|F_{k+1}\|^2 > \|F_0\|^2 + \|F_1\|^2 + \ldots + \|F_k\|^2 + \eta\left(\|F_0\|^2 + \|F_1\|^2 + \ldots + \|F_k\|^2\right) \quad (53)$$

However, (53) can reduce to

$$\|F_{k+1}\|^2 > \|F_0\|^2 + \eta \sum_{j=0}^{k} \|F_j\|^2 > \|F_0\|^2. \quad (54)$$

Which implies that

$$\|F_{k+1}\|^2 > \|F_0\|^2. \quad (55)$$

So,

$$\|F_{k+1}\| > \|F_0\|; \text{for some } k. \quad (56)$$

This contradicts Assumption 3.1. Thus, we finally conclude that

$$\liminf_{k\to\infty} \|F_k\| = 0. \quad (57)$$

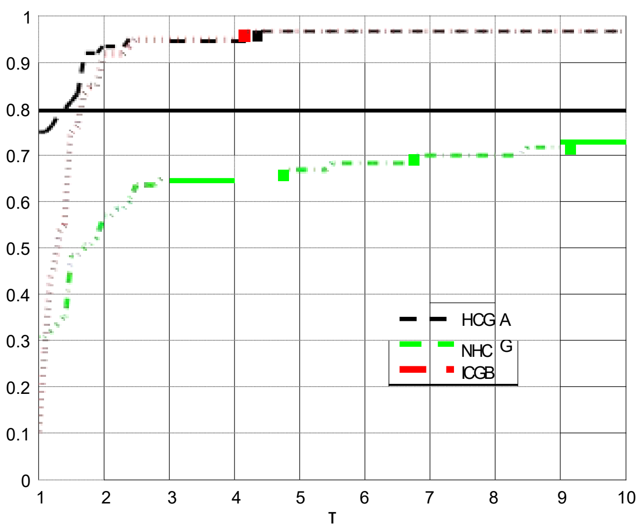
4. Numerical Experiment

The performance of our algorithm is compared with a new hybrid Dai-Yuan and Hestenes-Stiefel conjugate gradient (NHCG) method (Jamilu et al., 2017) and that of improved conjugate

Table 3
Numerical experiments of HCGA, NHCG and ICGB algorithms for problems 11–20

		HCGA			NHCG			ICGB		
Problems	Dim	NI	Time (s)	$\|F(x)\|$	NI	Time (s)	$\|F(x)\|$	NI	Time (s)	$\|F(x)\|$
11	1000	27	0.121965	7.52E-05	16	0.637253	7.34E-05	28	0.263939	8.82E-05
	10,000	30	0.987005	7.40E-05	13	1.489052	7.57E-05	31	0.996675	8.68E-05
	100000	33	5.054224	7.28E-05	20	43.61631	8.17E-05	34	4.684142	8.54E-05
12	1000	32	0.302352	6.50E-05	16	1.109407	7.54E-08	24	0.239484	8.06E-05
	10,000	44	1.164529	7.39E-05	10	4.130877	1.66E-05	42	0.998665	7.02E-05
	100000	58	6.881111	8.39E-05	48	29.09834	7.63E-09	52	5.803645	6.15E-05
13	1000	9	0.038496	5.20E-05	13	0.473268	8.53E-05	10	0.138206	9.31E-05
	10,000	9	0.601466	5.20E-05	13	0.652902	8.53E-05	10	0.695646	9.31E-05
	100000	10	3.104969	5.20E-05	13	2.950284	8.53E-05	11	3.556096	9.31E-05
14	1000	91	0.543911	7.97E-05	56	0.512586	8.14E-06	91	0.655509	7.97E-05
	10,000	94	1.870703	8.07E-05	57	2.273382	4.68E-05	96	1.882174	8.06E-05
	100000	87	11.084340	8.29E-05	56	56.22549	4.98E-05	88	11.803190	9.30E-05
15	1000	18	0.816613	8.16E-05	–	–	–	20	1.010536	6.03E-05
	10,000	17	52.85926	9.45E-05	–	–	–	19	58.208154	7.93E-05
	100000	–	–	–	–	–	–	–	–	–
16	1000	4	0.132100	7.48E-05	6	7.967907	4.12E-09	5	0.261741	9.49E-07
	10,000	5	0.667781	5.78E-10	11	1.029034	3.38E-06	5	0.670711	3.07E-06
	100000	5	2.264731	1.83E-09	–	–	–	5	2.874141	9.72E-06
17	1000	31	0.231619	6.81E-05	26	0.547314	8.07E-05	32	0.265430	6.81E-05
	10,000	33	0.569699	9.77E-05	23	6.107963	7.22E-05	34	0.963672	9.77E-05
	100000	36	4.286214	9.44E-05	22	88.32929	8.36E-05	37	4.707265	9.44E-05
18	1000	24	0.236314	6.20E-05	–	–	–	26	0.592225	6.20E-05
	10,000	26	0.807640	7.06E-05	–	–	–	28	0.980387	7.06E-05
	100000	28	4.793412	8.04E-05	–	–	–	30	4.793710	8.04E-05
19	1000	20	1.255967	9.46E-05	109	5.323423	9.54E-05	26	0.592225	6.20E-05
	10,000	23	74.09088	7.35E-05	109	371.8765	9.49E-05	28	85.561769	8.54E-05
	100000	–	–	–	–	–	–	–	–	–
20	1000	10	0.097154	6.36E-05	68	1.950169	1.17E-04	11	0.201417	6.36E-05
	10,000	11	0.735711	4.02E-05	101	4.387700	1.48E-10	12	0.722308	4.02E-05
	100000	12	3.069437	2.54E-05	101	51.42762	1.46E-11	13	3.321553	2.54E-05

Figure 1
Performance profile of HCGA, NHCG and ICGB algorithms with regard to the number of iterations for the problems 1–20



gradient method for nonlinear system of equations (ICGB) (Waziri et al., 2020) to solve (1).

We set the following parameters for the experiments in our algorithm (HCGA):

$r = 0.8$, $\eta_k = \frac{1}{(k+1)^2}$, $\omega_1 = \omega_2 = 10^{-4}$ and $\delta_k = 0.9$.

The parameters for new hybrid Dai-Yuan and Hestenes-Stiefel conjugate gradient parameters (NHCG) (Jamilu etal., 2017) are as follows:

$r = 0.2$, $\omega_1 = \omega_2 = 10^{-4}$, $\eta_k = \frac{1}{(k+1)^2}$; and $\delta = 0.9$.

Similarly, the parameters for ICGB established in (Waziriet al., 2020) are as follows: $r = 0.2$, $\eta_k = \frac{1}{(k+1)^2}$, $\omega_1 = \omega_2 = 10^{-4}$ and $\delta_k = 0.9$.

All the algorithms were run on a computer with a 2.13 GHz CPU and RAM of 2 GB after being executed in MATLAB 7.71 GB (R2014a).

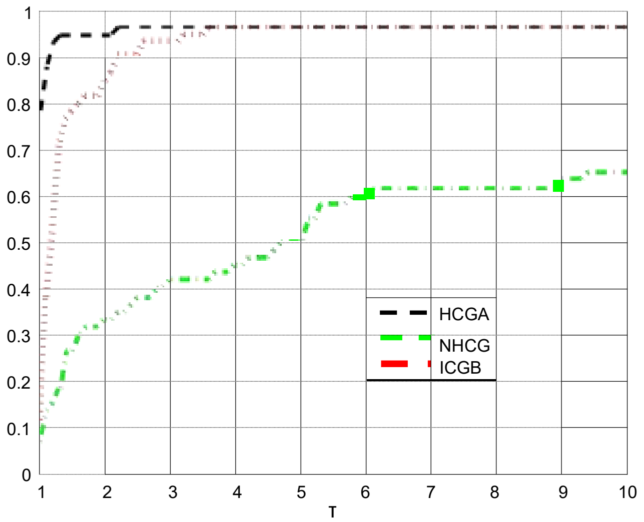
If the total number of iterations reaches 5000 without getting the solution or $\|F_k\| \leq 10^{-4}$, then the iteration would be terminated. Twenty (20) test problems (See Table 1) were used to test the algorithmswith various dimensions (n values) and different initial guess.

Tables 2 and 3 contain the experimental results for the three methods with "NI" and "Time," respectively, representing the total number of iterations and CPU time in seconds, while the norm of the function F is $\|F(x)\|$. We can easily see from the tables that the three algorithms were used to solve (1), but the efficiency, robustness and effectiveness of our algorithm over NHCG and ICGB are clearly shown, because the proposed algorithm requires less CPU time and number of iterations than NHCG and ICGB respectively.

Figures 1 and 2 illustrate how well our method performs in terms of CPU time and number of iterations using the performance profiles of Dolan and Moré (Dolan & Moré, 2002).

Figure 2

Performance profile of HCGA, NHCG and ICGB algorithms with regard to the CPU time (in second) for the problems 1–20



5. Conclusion

In this article, we presented a HCGA for systems of nonlinear equations and compared its effectiveness against NHCG method proposed in Jamilu et al. (2017) and that of ICGB implemented in Waziri et al. (2020) for solving equations (1), by performing some numerical experiments. A non-monotone type line search (Fukushima & Li, 1999) is used to prove the convergence of our suggested algorithm, and the numerical experiments demonstrate that our algorithm is promising.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

References

Al-Baali,M. (1985). Descent property and global convergence of the Fletccher-Reeves method with inexact line search. *IMA Journal of Numerical Analysis*, 5, 121–124.

Andrei, N. (2007). Scaled memory-less BFGS preconditioned conjugate gradient algorithm for unconstrained optimization. *Optimization Method and Software*, 22, 261–571.

Andrei, N. (2008). Another hybrid conjugate gradient algorithm for unconstrained optimization. *Numerical Algorithm*, 47,143–156.

Andrei, N. (2009). Hybrid conjugate gradient algorithm for unconstrained optimization. *Journal of Optimization Theory Application*, 141, 249–264.

Babaie-Kafaki, S., Masoud, F., & Nezam, M. (2011). Two effective hybrid conjugate gradient algorithms based on modified BFGS updates. *Numerical Algorithm*, 58, 315–331.

Dai, Y. H., & Yuan, Y. X. (1999). A nonlinear conjugate gradient with a strong global convergence properties. *SIAM Journal of Optimization*, 10, 177–182.

Dai, Y. H., & Yuan, X. Y. (1999). A nonlinear conjugate Gradient Method with Strong Global Convergence Property. *SIAM Journal of Optimization*, 10, 177–182.

Dai, Y. H., & Yuan, Y. X. (1999). A nonlinear conjugate gradient with a strong global convergence properties. *SIAM Journal on Optimization*, 10, 177–182.

Dauda, M. K., Mamat, M., Mohamed, M. A., & Waziri, M. Y. (2019). Improved Quasi-Newton method via SR1 update for solving symmetric systems of nonlinear equations. *Malaysian Journal of Mathematical Sciences, 15*(1), 117–130.

Dauda, M. K., Mustafa, M., Fatima, S. M., Abubakar, S. M., & Waziri, M. Y. (2019). Derivative free conjugate gradient method via Broyden's update for solving symmetric systems of nonlinear equations. *Journal of Physics: Conference Series*, 1, 1366–1344.

Dauda,M. K., Mustafa,M., Waziri,M. Y., Fadhila,A., & Fatima, S. M. (2016). Inexact CG-method via SR1 update for solving systems of nonlinear equations. *Far East Journal of Mathematical Science (FJMS), 100*(11), 1787–1804.

Djordjevic, S. S. (2016). New hybrid conjugate gradient method as a convex combination of FRandPRP methods. *Filomat, 30*(11), 083–3100.

Dolan,E., & Moré,J. (2002). Benchmark optimization software with performance profiles. *Journal of Mathematical Program*, 91, 201–213.

Fang, X., & Ni,Q. (2017). A new derivative-free conjugate gradient method for large-scale nonlinear systems of equations. *Bulletin of the Australian Mathematical Society*.95, 500–511.

Fletcher, R., & Reeves, C. M. (1964). Function minimization by conjugate gradients. *The Computer Journal*, 7, 149–154.

Fletcher, R., & Reeves, C. M. (1964). Fundamental minimization by conjugate gradients. *The Computer Journal*, 7, 149–154.

Fukushima, M., & Li, D. (1999). A global and super-linear convergent Gauss-Newton base BFGS method for symmetric nonlinear equation. *SIAM Journal on Numerical Analysis*, 37, 152–172.

Hager, W. W., & Zhang, H. (2006). A survey of nonlinear conjugate gradient methods. *Pacific Journal of Optimization*, 2, 35–58.

Halilu, A. S., & Waziri, M. Y. (2017). A transformed double steplength method for solving large-scale system of nonlinear equations. *Journal of Numerical Mathematics and Stochastics, 9*(1), 20–32.

Hesetenes, M. R., & Stiefel, E. (1952). Methods for conjugate gradients for solving linear systems. Journal of Research of the National Bureau Standards, 49, 409–436.

Ioannis,E. L., Vassilis,T., & Panagiotis,P. (2018). Adescent hybrid conjugate gradient method based on memoryless BFGS update. *Numerical Algorithms*, 67, 31–48.

Jamilu, S., Waziri,M. Y., & Abba,I. (2017). A new hybrid Dai-Yuan and Hestenes-Stiefel conjugate gradient parameters for solving system of nonlinear equations. *MAYFEB Journal of Mathematics*, 1, 44–55.

Liu, D. C., & Nocedal, J. (1989). On the limited memory BFGS method for large scale optimization methods. *Journal of Mathematical Program*, 45, 503–528.

Liu, Y., & Storey, C. (1991). Efficient generalized conjugate gradient algorithms, part 1. *Journal of Optimization Theory Application*, 69, 129–137.

Oren,S. S., & Luenberger,D.G.(1974). Self-Scaling variable matrix (SSVM) algorithms, partI, criteria and sufficient conditions for scaling a class of algorithms. *Computational Management Science*, 20, 845–862.

Perry, A. (1978). A modified conjugate gradient algorithm. *Operation Research Technology Notes, 26*(6), 1073–1078.

Polak,E., & Ribiere, G. (1969). Note sur la convergence de methods de direction conjuguees. *Revue Francais d'Informatique et de Recherche Operationnelle*, 16, 35–43.

Powell,M.J. D.(1984). Non-convex Minimization calculations and the conjugate gradient method. *Numerical Analysis*, 1066, 122–141.

Waziri, M. Y., Leong, W. J., & Hassan, M. A. (2011). Jacobian free-diagonal Newton's method for nonlinear systems with singular Jacobian. *Malaysian Journal of Mathematical Sciences, 5*(2), 241–255.

Waziri, M. Y., Leong, W. J., Hassan, M. A., & Monsi, M. (2010). Jacobian computation-free Newton method for systems of Non-Linear equations. *Journal of Numerical Mathematics and Stochastic*, 2, 54–63.

Waziri, M. Y., Leong, W. J., Hassan, M. A., & Monsi, M. (2010). A new Newton's method with diagonal Jacobian approximation for system of nonlinear equations. *Journal of Mathematics and Statistics*, 6(3), 246–252.

Waziri, M. Y., & Sabiu, J. (2015). A derivative-free conjugate gradient method and its global convergence for symmetric nonlinear equations. *Journal of Mathematics and Mathematical Sciences*, 8, 2015.

Waziri, M. Y., Yusuf, A., & Abubakar, A. B. (2020). Improved conjugate gradient method for nonlinear system of equations. *Computational and Applied Mathematics*, 39, 1–17.

Zang, L., Zhou, W., & Li, D. (2006). Global convergence of modified Fletcher-Reeves conjugate gradient method with Armijo-type line search. *Journal of Numerical Mathematics*, 104, 561–572.

How to Cite: Yusuf, A., Kiri, A. A., Lawal, L., & Kiri, A. I. (2024). A Hybrid Conjugate Gradient Algorithm for Nonlinear System of Equations through Conjugacy Condition. *Journal of Global Humanities and Social Sciences, 5*(10), 364–371. https://doi.org/10.61360/BoniGHSS242016851001