Available online at www.sciencedirect.com
ScienceDirect
Mathematics and Computers in Simulation 187 (2021) 520–539
MATHEMATICS AND COMPUTERS IN SIMULATION
www.elsevier.com/locate/matcom

Original articles

Signal recovery with convex constrained nonlinear monotone equations through conjugate gradient hybrid approach

Abubakar Sani Halilu[a,b,d], Arunava Majumder[a,*], Mohammed Yusuf Waziri[c,d], Kabiru Ahmed[c,d]

[a] Department of Mathematics, School of Chemical Engineering and Physical Sciences, Lovely Professional University, Phagwara, 144411, India
[b] Department of Mathematics and Computer Science Sule Lamido University, Kafin Hausa, Nigeria
[c] Department of Mathematical Sciences, Bayero University, Kano, Nigeria
[d] Numerical Optimization Research Group, Bayero University, Kano, Nigeria

Received 15 July 2020; received in revised form 1 February 2021; accepted 15 March 2021
Available online 23 March 2021

Abstract

In recent years there is a vast application of conjugate gradient methods to restore the disturbed signals in compressive sensing. This research aims at developing a scheme, which is more effective for restoring disturbed signals than the popular PCG method (Liu & Li, 2015). To realize the desired goal, a new conjugate gradient approach combined with the projection scheme of Solodov and Svaiter [Kluwer Academic Publishers, pp. 355-369(1998)] for solving monotone nonlinear equations with convex constraints is presented. The main idea employed in this algorithm is to approximate the Jacobian matrix via acceleration parameter in order to propose an effective conjugate gradient parameter. In addition, the step length is calculated using inexact line search technique. The proposed approach is proved to converge globally under some mild conditions . The numerical experiment, depicts the efficacy our method. Apart from generating search directions that are vital for global convergence, a significant contribution of the new method lies in its applications to solve the ℓ_1-norm regularization problem in signal recovery. Experiments with the scheme and the effective PCG solver, existing in the previous literature, shows that the new method provides much better results.

Keywords: Acceleration parameter; Conjugate gradient parameter; Jacobian matrix; Derivative free; Global convergence

1. Introduction

The constrained nonlinear equation to be considered in this paper is represented by

$$F(a) = 0, \quad a \in \Omega, \tag{1}$$

where $F : \mathbb{R}^n \to \mathbb{R}^n$ is continuous and monotone, i.e.;

$$(a - b)^T (F(a) - F(b)) \geq 0, \quad \forall a, b \in \mathbb{R}^n. \tag{2}$$

* Corresponding author at: Department of Mathematics, School of Chemical Engineering and Physical Sciences, Lovely Professional University, Phagwara, 144411, India.

E-mail addresses: abubakars.halilu@slu.edu.ng, ashalilu.mth@gmail.com (A.S. Halilu), am.arunavamajumder@gmail.com, arunava.23440@lpu.co.in (A. Majumder), mywaziri.mth@buk.edu.ng (M.Y. Waziri), kabiruhungu16@gmail.com (K. Ahmed).

https://doi.org/10.1016/j.matcom.2021.03.020

In addition, the set $\Omega \subseteq \mathbb{R}^n$ is convex, closed and nonempty. Throughout this paper, the space $\mathbb{R}^n$ denote the n-dimensional real space, $\|\cdot\|$ is the Euclidean norm and $F_k = F(a_k)$.

The monotone mappings were initially researched by Zarantonello [62], Minty [36], Kačurovskii [26], in Hilbert spaces. The studies of such mappings are practically applied in many scientific fields, like in the systems of economic equilibrium [47] and chemical equilibrium [35]. It has practical application in ℓ_1-norm regularization problem in signal recovery [58] as well. Some iterative methods for solving these problems include Newton and quasi-Newton methods [8,54,59,61], the Gauss–Newton and Levenberg–Marquardt methods [25,28,29,34,52], the matrix-free methods [1,16,53,55], and double direction and step length methods [17–23].

Several methods are used to solve (1) but the Newton and quasi-Newton methods are the widely used methods. These approaches are easy to implement with fast convergence from a reasonably good starting point. Nonetheless, at each iteration they will compute either the Jacobian or its approximation, so they are not ideal for solving large-scale problems. For this reason, conjugate gradient (CG) approaches have been proposed [1,7,13,45,58]. The methods are very promising to solve the large-scale problems because of their low storage requirements. Their pertaining iterative procedure is defined as

$$a_{k+1} = a_k + s_k, \quad s_k = \alpha_k d_k, \quad k = 0, 1, \ldots, \tag{3}$$

where, $s_k = a_{k+1} - a_k$ and α_k is a step length. Line search is fundamentally required to establish

$$\|F_{k+1}\| \le \|F_k\|.$$

The conjugate gradient search direction d_k is computed by

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

where, β_k is a scalar that describe the nature of the CG methods. The rationale behind any CG method is to produce an effective conjugate gradient parameter β_k. Below are some of the well known CG parameters: $\beta_k^{FR} = \frac{\|F_k\|^2}{\|F_{k-1}\|^2}$ [13], $\beta_k^{PRP} = \frac{F_k^T y_{k-1}}{\|F_{k-1}\|^2}$ [45], $\beta_k^{HS} = \frac{F_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}$ [24], $\beta_k^{DY} = \frac{\|F_k\|^2}{d_{k-1}^T y_{k-1}}$ [7]. Note that $y_{k-1} = F_k - F_{k-1}$.

The Properties of convergence of the above-mentioned approaches have been widely researched in literature [12, 39]. Although the HS and PRP methods have adequate numerical performance, their convergence properties are not efficient. On the other hand, FR and DY methods have certain good convergence features, but their numerical results are not effective [12]. A great deal of effort has been made in recent years to develop and build new and efficient formulas for the CG parameters to facilitate the application of these methods to different fields, that perform good numerical results with global convergence properties. For these reasons, the hybrid methods have been introduced, which are based on the combination of two different methods in order to present an effective method with better numerical performance and convergence properties [3,27,44]. Petrovic et al. in [41], hybridized the accelerated gradient descent (SM) method [42] with the Picard–Mann hybrid iterative scheme [27], where the Picard–Mann hybrid iterative process is defined by:

$$\begin{cases} a_1 = a \in \Omega, \\ b_k = (1 - \eta_k) a_k + \eta_k T a_k, \\ a_{k+1} = T b_k, \quad k \in \mathbb{N}, \end{cases} \tag{5}$$

where $T : \Omega \longrightarrow \Omega$ is a mapping defined on nonempty convex subset Ω of a normed space $\mathbf{E}$, a_k and b_k are sequences determined by the iteration (5) and $\{\eta_k\}$ is the sequence of positive numbers in $(0, 1)$. In this paper, η_k is denoted as correction parameter.

To improve the convergence properties and the numerical performance of the accelerated double direction method in [43], Petrovic et al. [44] presented a hybrid modification of accelerated double direction method and the scheme generated a sequence of iterates $\{a_k\}$ such that

$$a_{k+1} - a_k = \beta {\alpha_k}^2 d_k - \beta \alpha_k \gamma_k^{-1} g_k,$$

where the correction parameter is defined as $\beta = (\eta_k + 1) \in (1, 2), \forall k$.

From the above literatures, we can conclude that applying hybridization process is a good way to improve the convergence properties and numerical experiments of some existing methods.

In general, d_k is needed to satisfy the descent condition

$$F_k^T d_k < 0.$$

The descent direction is very important for any CG method to converge globally. Nevertheless, some CG directions are not descent. In order to produce descent directions, some three-term CG methods that generated descent directions are presented in [38,63]. The convergence analysis of these methods is demonstrated under certain appropriate conditions. Narushima, once more, presented a smoothing CG algorithm in [37] with descent direction, that combined the smoothing technique with the PRP method in [31] to solve nonsmooth equations for unconstrained optimization problem.

In recent decades, unconstrained optimization problems have been extended to solve the nonlinear system of equations. This is because of their global convergence properties and low memory requirements. Cheng in [4] proposed PRP-type algorithm for monotone equations, that combined the PRP method for unconstrained optimization problem with the hyperplane approach in [46], that was assumed to be Lipschitz continuous and monotone. However, in [60] Yu proposed a spectral gradient-type method for solving large-scale nonlinear system of equations which extended the classical PRP scheme for unconstrained minimization problems. The approach possessed some nice properties where each iteration is well defined irrespective of whether the search direction is descent or not that led the algorithm to converge globally. Furthermore, Ming et al. [30] developed the idea for solving a family of CG methods for unconstrained optimization [6]. The scheme comprised the family of Hager–Zhang method as well as the Dai–Kou method. In addition, the numerical results indicated the effectiveness of the method in [30] by comparing it with the existing method in the literature. However, Waziri et al. [51] incorporated the idea in [6] and presented a method via modified secant equation, where the presented method converged globally using the nonmonotone line search proposed in [29]. Recently, Waziri et al. [52] proposed a Hager–Zhang family of CG methods for monotone nonlinear equations, inspired by the unconstrained optimization problems in [15] and the Solodov and Svaiter hyperplane technique [46]. This is an interested idea which clearly proved that the proposed search direction is decent using Eigenvalue analysis.

Motivated by the projection method [46], many methods are developed by researchers to solve constrained monotone nonlinear equations. For instance, Wang et al. [50], incorporated the approach [46] and solved the monotone nonlinear equations with convex constraint, where, the linear system of equations is approximately solved after the initialization in order to obtain a point of trial and then use the projection method to achieve the next iteration. The method in [50] is proved to be globally convergent with linear rate of convergence. To increase the rate of convergence of the approach in [50], its modification was developed by Wang and Wang [49] with superlinear rate of convergence.

Therefore, motivated by the work in [27] and the projection technique in [46], the purpose of this article is to develop a method with convex constraints that is globally convergent. This is made possible by combining the direction proposed by Halilu and Waziri in [20] with the spectral gradient direction and employ the hybrid iterative process in [27] in order to obtain an effective conjugate gradient and an appropriate correction parameters respectively. And to further demonstrate and highlight contribution of the new method, it is applied to restore disturbed signals in compressive sensing.

From Table 1, PCG is the method in [32] and CHCG stands for the scheme in this article.

The paper is organized as follows. In the next section, we will present the proposed method's algorithm. The convergence analysis of the proposed algorithm is shown in Section 3. Section 4 lists some numerical experiments and the applications of the proposed approach to signal recovery. The article ends in Section 5.

2. Preliminaries and algorithm

First, we introduce the projection operator and make some assumptions. Let the set $\Omega \subset \mathbb{R}^n$ be convex, closed and nonempty. For any $a \in \mathbb{R}^n$, its projection onto Ω is given by

$$P_\Omega(a) = \arg\min \|a - b\| : b \in \Omega. \tag{6}$$

The mapping $P_\Omega : \mathbb{R}^n \to \Omega$ is known as a projection operator which has nonexpansive property namely, for any $a, b \in \mathbb{R}^n$ it holds that

$$\|P_\Omega(a) - P_\Omega(b)\| \le \|a - b\|, \tag{7}$$

Table 1
Author's contribution table.

Author's name	PCG method	CHCG Method	Derivative-free	Matrix-free	Convex constraints	Global convergence	Application
Li and Fukushima [29]			✓	✓		✓	
Dai and Liao [6]			✓	✓		✓	
Kanzow et al. [25]			✓				
Zhou and Li [63]			✓			✓	
Yuan and Lu [61]			✓			✓	
Yu [60]			✓	✓		✓	
Xiao et al. [57]			✓		✓	✓	✓
Xiao and Zhu [58]			✓	✓	✓	✓	✓
Liu and Li [32]	✓		✓	✓	✓	✓	✓
Waziri and Sabiu [56]			✓	✓		✓	
Halilu and Waziri [19]			✓	✓		✓	
Abdullahi et al. [1]			✓	✓		✓	
Halilu and Waziri [19]			✓	✓		✓	
Waziri et al. [53]			✓	✓		✓	
Abubakar et al. [2]	✓		✓	✓	✓	✓	✓
Halilu et al. [16]			✓	✓		✓	
This article	✓	✓	✓	✓	✓	✓	✓

as a result, we have

$$\|P_\Omega(a) - b\| \leq \|a - b\|, \quad \forall b \in \Omega. \tag{8}$$

On the other hand, let us consider the derivative-free double direction method in [20]. The method developed a derivative-free scheme for solving (1) via

$$F'_k \approx \gamma_k I, \tag{9}$$

where I is an identity matrix and $\gamma_k > 0$ is an acceleration parameter. The method in [20] produced a sequence of iterates $\{a_k\}$ such that $a_{k+1} = a_k + (\alpha_k + \alpha_k^2 \gamma_k) d_k$ and the direction d_k is given as

$$d_k = -\gamma_k^{-1} F_k, \tag{10}$$

and γ_k is obtained as

$$\gamma_k = \frac{y_{k-1}^T y_{k-1}}{y_{k-1}^T s_{k-1}}, \tag{11}$$

where, $y_{k-1} = F_k - F_{k-1}$ and $s_k = (\alpha_k + \alpha_k^2 \gamma_k) d_k$.

Despite the fact that the method [20] has strong convergence properties, its numerical performance is weak when α_k approaches to zero. For this reason, we are motivated to propose a hybrid CG method with strong convergence properties and effective numerical results.

To define a hybrid form of the approach in [20], the mapping T is assumed to be defined by an improved double direction method as $Tb_k = b_k - (\alpha_k + \alpha_k^2 \gamma_k)\gamma_k^{-1} F_k$. By this assumption and (5) we have

$$\begin{cases} a_1 = x \in \Omega, \\ b_k = (1-\eta_k)a_k + \eta_k T a_k = (1-\eta_k)a_k + \eta_k(a_k - (\alpha_k + \alpha_k^2\gamma_k)\gamma_k^{-1}F_k) \\ = a_k - (\eta_k + 1)(\alpha_k + \alpha_k^2\gamma_k)\gamma_k^{-1}F_k, \\ a_{k+1} = T y_k = b_k - (\alpha_k + \alpha_k^2\gamma_k)\gamma_k^{-1}F_k, \quad k \in \mathbb{N}. \end{cases} \tag{12}$$

From the second and third equations in (12) we obtain the iterative scheme,

$$a_{k+1} = a_k - t_k(\alpha_k + \alpha_k^2\gamma_k)\gamma_k^{-1}F_k, \tag{13}$$

where, $t_k = (\eta_k + 1) \in (1, 2)$ is a correction parameter. we can easily show that, the search direction in (13) is defined as:

$$d_k = -t_k \gamma_k^{-1} F_k. \tag{14}$$

Now, we are going to derive our proposed direction. This is made possible by combining (4) and (14) as follows:

$$t_k \gamma_k^{-1} F_k = (F_k - \beta_k d_{k-1}), \tag{15}$$

by substituting (11) into (15) we have

$$t_k (y_{k-1}^T s_{k-1}) F_k = y_{k-1}^T y_{k-1} (F_k - \beta_k d_{k-1}). \tag{16}$$

Observe that after multiplying (16) by y_{k-1}^T it can be written as

$$\beta_k (y_{k-1}^T y_{k-1})(y_{k-1}^T d_{k-1}) = (y_{k-1}^T y_{k-1})(y_{k-1}^T F_k) - t_k (y_{k-1}^T F_k)(y_{k-1}^T s_{k-1}). \tag{17}$$

Applying some algebraic calculation, we can write the proposed CG parameter and search direction as

$$\beta_k = \frac{(\|y_{k-1}\|^2 - t_k y_{k-1}^T s_{k-1}) y_{k-1}^T F_k}{\|y_{k-1}\|^2 y_{k-1}^T d_{k-1}}, \tag{18}$$

$$d_k = \begin{cases} -F_k, & \text{if} \quad k = 0, \\ -\theta_k F_k + \dfrac{(\|y_{k-1}\|^2 - t_k y_{k-1}^T s_{k-1}) y_{k-1}^T F_k}{\|y_{k-1}\|^2 y_{k-1}^T d_{k-1}} d_{k-1}, & \text{if} \quad k \geq 1. \end{cases} \tag{19}$$

where $\theta_k = \frac{s_{k-1}^T s_{k-1}}{y_{k-1}^T s_{k-1}}$. (see [56]).

Next, the algorithm of the proposed method is specified as follows:

Algorithm 1: Conjugate gradient hybrid method(CHCG)

Input: Given $x_0 \in R^n$, $d_0 = -F_0$, $\gamma_0 = 0.01$, $t = t_k \in (1, 2)$, $\rho \in (0, 1)$, $\sigma > 0$, $\epsilon = 10^{-10}$, $\xi > 0$, set $k = 0$.
Step 1: Compute F_k. If $\|F_k\| \leq \epsilon$ then stop, else goto the next step.
Step 2: Let $\mu_k = (\alpha_k + \alpha_k^2 \gamma_k)$ and set $r_k = a_k + \mu_k d_k$, where, $\alpha_k = \xi \rho^{m_k}$ with m_k being the smallest nonnegative integer m satisfying

$$-F(a_k + \mu_k d_k)^T d_k \geq \sigma \mu_k \|d_k\|^2. \tag{20}$$

Step 3: If $r_k \in \Omega$ and $\|F(r_k)\| \leq \epsilon$, then stop, otherwise go to the next Step.
Step 4: Compute the next iterate by

$$a_{k+1} = P_\Omega [a_k - \lambda_k F(r_k)], \tag{21}$$

where $\lambda_k = \dfrac{(a_k - r_k)^T F(r_k)}{\|F(r_k)\|^2}$.
Step 5: Compute d_{k+1} using (19).
Step 6: Determine $\gamma_{k+1} = \dfrac{y_k^T y_k}{y_k^T s_k}$, where, $s_k = r_k - a_k$ and $y_k = (F(r_k) - F(a_k))$.
Step 7: Set k=k+1, and go to STEP 2.

---

3. Convergence analysis

In this section, the analysis of the global convergence of CHCG algorithm is presented. We consider the following assumptions:

Assumption 3.1.

(A1) The mapping F has nonexpansive property; namely, for any $a, b \in \mathbb{R}^n$ it holds that

$$\|F(a) - F(b)\| \leq \|a - b\|. \tag{22}$$

(A2) The mapping F is uniformly monotone, namely, there exists a positive constant c such that

$$(a - b)^T (F(a) - F(b)) \geq c\|a - b\|^2, \quad \forall a, b \in \mathbb{R}^n. \tag{23}$$

(A3) For any $a \in S^a$ there exists a constant $\eta > 0$ such that

$$\eta \text{dist}(a, S^a) \leq \|F(a)\|^2, \quad \forall a \in N(a^*, S^a), \tag{24}$$

where, $\text{dist}(a, S^a)$ denotes the distance from a to the solution set S^a, $N(a^*, \Omega) := \{a \in \mathbb{R}^n | \|a - a^*\| \leq \eta\}$. This is the local bound condition.

Remark 3.2. We give the following remarks

(a) From (23), we have

$$y_{k-1}^T s_{k-1} = s_{k-1}^T (F(r_{k-1}) - F(a_{k-1})) \geq c\|s_{k-1}\|^2 > 0. \tag{25}$$

From (22) and (25) we have

$$y_{k-1}^T s_{k-1} = (F(r_{k-1}) - F(a_{k-1}))^T (r_{k-1} - a_{k-1}) \leq \|s_{k-1}\|^2. \tag{26}$$

Also, from (25) and (26) it is simple to check that, $\|s_{k-1}\|^2 \geq y_{k-1}^T s_{k-1} \geq c\|s_{k-1}\|^2$, this implies that $\frac{\|s_{k-1}\|^2}{y_{k-1}^T s_{k-1}} \leq \frac{1}{c}$ and $\frac{\|s_{k-1}\|^2}{y_{k-1}^T s_{k-1}} \geq 1$. This means θ_k in (19) is well defined. Therefore, we have

$$1 \leq \theta_k \leq \frac{1}{c}. \tag{27}$$

(b) From assumption (A1) we have

$$\|y_{k-1}\| = \|F(r_{k-1}) - F(a_{k-1})\| \leq \|s_{k-1}\| = \mu_{k-1}\|d_{k-1}\|. \tag{28}$$

(c) From assumption (A2) we have

$$y_{k-1}^T d_{k-1} = (F(r_{k-1}) - F(a_{k-1}))^T \frac{s_{k-1}}{\mu_{k-1}} \geq \frac{c\|s_{k-1}\|^2}{\mu_{k-1}} = c\mu_{k-1}\|d_{k-1}\|^2 > 0. \tag{29}$$

Since $y_{k-1}^T d_{k-1} > 0$, the proposed β_k in (18) is well defined.

(d) From assumptions (A1) and (A3) we have

$$\frac{\|y_{k-1}\|^2}{y_{k-1}^T s_{k-1}} \leq \frac{\|s_{k-1}\|^2}{c\|s_{k-1}\|^2} = \frac{1}{c}. \tag{30}$$

Therefore, from (11) and (30) we have

$$\gamma_k = \frac{\|y_{k-1}\|^2}{y_{k-1}^T s_{k-1}} \leq \frac{1}{c}. \tag{31}$$

Lemma 3.3. *Suppose assumptions (A1)–(A3) hold and $\{a_k\}$ be generated by CHCG algorithm, then the search direction d_k fulfills the descent property i.e.,*

$$F_k^T d_k < 0. \tag{32}$$

Proof. Here two cases are to be considered:
Case 1: For $k = 0$, by (19) we have $F_0^T d_0 = -\|F_0\|^2 < 0$.
Case 2: From (31) we have $\|y_{k-1}\|^2 \leq \frac{1}{c} y_{k-1}^T s_{k-1}$.
Now, from (19) and since $t = t_k \in (1, 2)$, we have

$$\begin{aligned}
F_k^T d_k &= -\theta_k \|F_k\|^2 + \frac{(\|y_{k-1}\|^2 - t y_{k-1}^T s_{k-1})(y_{k-1}^T F_k)(F_k^T d_{k-1})}{\|y_{k-1}\|^2 (y_{k-1}^T d_{k-1})}, \\
&\leq -\theta_k \|F_k\|^2 + \frac{(\frac{1}{c} y_{k-1}^T s_{k-1} - t y_{k-1}^T s_{k-1})(y_{k-1}^T F_k)(F_k^T d_{k-1})}{\|y_{k-1}\|^2 (y_{k-1}^T d_{k-1})}, \\
&\leq -\theta_k \|F_k\|^2 + \frac{(\frac{1}{c} y_{k-1}^T s_{k-1} - \frac{1}{c} y_{k-1}^T s_{k-1})(y_{k-1}^T F_k)(F_k^T d_{k-1})}{\|y_{k-1}\|^2 (y_{k-1}^T d_{k-1})},
\end{aligned}$$

$$
\begin{aligned}
&\leq -\theta_k \|F_k\|^2, \\
&\leq -\|F_k\|^2. \qquad (33)
\end{aligned}
$$

The last inequality follows from (27). □

Using the next lemma, we demonstrate that the line search used in CHCG algorithm is well-defined.

Lemma 3.4. *Suppose assumptions (A1)–(A3) hold. Then there exists a step length α_k satisfying (20) for all $k \geq 0$.*

Proof. Going by contradiction, we assume that there exists a constant $k_0 \geq 0$, such that given any nonnegative integer m, we have

$$
-F(x_{k_0} + (\xi\rho^m + (\xi\rho^m)^2\gamma_{k_0})d_{k_0})^T d_{k_0} < \sigma(\xi\rho^m + (\xi\rho^m)^2\gamma_{k_0})\|d_{k_0}\|^2. \qquad (34)
$$

Since $\rho \in (0, 1)$, by using assumption (A2), (20) and letting $m \to \infty$, we get

$$
-F(a_{k_0})^T d_{k_0} \leq 0. \qquad (35)
$$

Also from (33), it follows that

$$
-F(a_{k_0})^T d_{k_0} \geq \|F(a_{k_0})\|^2 > 0, \qquad (36)
$$

which clearly contradicts (35). Hence, the line search is well-defined. □

Lemma 3.5. *Suppose assumptions (A1)–(A3) hold and let sequences $\{a_k\}$ and $\{r_k\}$ be generated by CHCG algorithm, then $\{a_k\}$ and $\{r_k\}$ are bounded. Moreover, we have*

$$
\lim_{k\to\infty} \|a_k - r_k\| = 0, \qquad (37)
$$

and

$$
\lim_{k\to\infty} \|a_{k+1} - a_k\| = 0. \qquad (38)
$$

Proof. First, we show the boundedness of the sequences $\{a_k\}$ and $\{r_k\}$. Let $\bar{a} \in S^a$ be any solution of (1). Then by monotonicity of F we can write

$$
(a_k - \bar{a})^T F(r_k) \geq (a_k - r_k)^T F(r_k). \qquad (39)
$$

Using the line search condition (20) and definition of r_k, we have

$$
(a_k - r_k)^T F(r_k) \geq \sigma\mu_k^2 \|d_k\|^2 > 0. \qquad (40)
$$

Also, using (8) and (21) we have

$$
\begin{aligned}
\|a_{k+1} - \bar{a}\|^2 &= \|P_\Omega(a_k - \lambda_k F(r_k)) - \bar{a}\|^2, \\
&\leq \|a_k - \lambda_k F(r_k) - \bar{a}\|^2, \\
&= \|a_k - \bar{a}\|^2 - 2\lambda_k (a_k - \bar{a})^T F(r_k) + \lambda_k^2 \|F(r_k)\|^2, \\
&\leq \|a_k - \bar{a}\|^2 - 2\lambda_k (a_k - r_k)^T F(r_k) + \lambda_k^2 \|F(r_k)\|^2, \qquad (41) \\
&= \|a_k - \bar{a}\|^2 - \frac{((a_k - r_k)^T F(r_k))^2}{\|F(r_k)\|^2}, \\
&\leq \|a_k - \bar{a}\|^2 - \sigma^2 \|a_k - r_k\|^4,
\end{aligned}
$$

for which we obtain

$$
\|a_{k+1} - \bar{a}\| \leq \|a_k - \bar{a}\|. \qquad (42)
$$

This recursively implies that $\|a_k - \bar{a}\| \leq \|a_0 - \bar{a}\|$ $\forall k$.

So, the sequence $\{\|a_k - \bar{a}\|\}$ is clearly a decreasing, which implies that $\{a_k\}$ is bounded. Furthermore, utilizing assumptions (A1), (A3) and (42) we have

$$\|F(a_k)\| = \|F(a_k) - F(\bar{a})\| \le \|a_k - \bar{a}\| \le \|a_0 - \bar{a}\|. \tag{43}$$

Let $m_1 = \|a_0 - \bar{a}\|$, then we have

$$\|F(a_k)\| \le m_1. \tag{44}$$

Moreover, from the definition of r_k, monotonicity of F, (40) and the Cauchy–Schwartz inequality, we have

$$\sigma\|a_k - r_k\| = \frac{\sigma\|\mu_k d_k\|^2}{\|a_k - r_k\|} \le \frac{(a_k - r_k)^T F(r_k)}{\|a_k - r_k\|} \le \frac{(a_k - r_k)^T F(a_k)}{\|a_k - r_k\|} \le \|F(a_k)\|. \tag{45}$$

So, using the boundedness of the sequences $\{x_k\}$, (44) and (45), we see that $\{r_k\}$ is bounded. Therefore, by the boundedness of the $\{r_k\}$, the sequence $\{\|r_k - \bar{a}\|\}$ is also bounded, i.e., there exists a constant $\kappa > 0$ for any $\bar{a} \in \Omega$, such that

$$\|r_k - \bar{a}\| \le \kappa. \tag{46}$$

From (22) and (46), we have

$$\|F(r_k)\| = \|F(r_k) - F(\bar{a})\| \le \|r_k - \bar{a}\| \le \kappa. \tag{47}$$

Also, from (57), (45), (47), we obtain

$$\frac{\sigma^2}{(\kappa)^2}\sum_{k=0}^{\infty}\|a_k - r_k\|^4 \le \sum_{k=0}^{\infty}\frac{((a_k - r_k)^T F(r_k))^2}{\|F(r_k)\|^2} \le \sum_{k=0}^{\infty}(\|a_k - \bar{a}\|^2 - \|a_{k+1} - \bar{a}\|^2) < \infty. \tag{48}$$

This implies (37).

On the other hand, from (8), the definition of λ_k we have

$$\begin{aligned} \|a_{k+1} - a_k\| &= \|P_\Omega(a_k - \lambda_k F(r_k)) - a_k\|, \\ &\le \|a_k - \lambda_k F(r_k) - a_k\|, \\ &= \|\lambda_k F(r_k)\|, \\ &\le \|a_k - r_k\|. \end{aligned} \tag{49}$$

This implies (38).

Notwithstanding, from (37) and the definition of r_k we have

$$\lim_{k\to\infty}\mu_k\|d_k\| = 0. \quad \square \tag{50}$$

Lemma 3.6. *Suppose assumptions (A1)–(A3) hold and $\{a_k\}$ be generated by CHCG algorithm. Then there exist some positive constants M such that for all $k > 0$,*

$$\|d_k\| \le M. \tag{51}$$

Proof. Here two cases are to be considered:
Case 1: For $k = 0$, by (19) and (44) we have $\|d_0\| = -\|F_0\| < m_1$.
Case 2: For $k \ge 1$, also from (19), (44) and remarks (a)–(d) we have

$$\begin{aligned} \|d_k\| &= \left\| -\theta_k F_k + \frac{(\|y_{k-1}\|^2 - t y_{k-1}^T s_{k-1}) y_{k-1}^T F_k}{\|y_{k-1}\|^2 (y_{k-1}^T d_{k-1})} d_{k-1} \right\|, \\ &\le \theta_k\|F_k\| + \left| \frac{y_{k-1}^T F_k}{y_{k-1}^T d_{k-1}} - \frac{t(y_{k-1}^T s_{k-1})(y_{k-1}^T F_k)}{\|y_{k-1}\|^2 (y_{k-1}^T d_{k-1})} \right| \|d_{k-1}\|, \\ &\le \theta_k\|F_k\| + \left[\frac{|y_{k-1}^T F_k|}{c\mu_{k-1}\|d_{k-1}\|^2} + \frac{t|y_{k-1}^T s_{k-1}||y_{k-1}^T F_k|}{\|y_{k-1}\|^2 (c\mu_{k-1}\|d_{k-1}\|^2)} \right] \|d_{k-1}\|, \end{aligned}$$

$$\begin{aligned}
&\le \theta_k\|F_k\| + \left[\frac{\|y_{k-1}\|\|F_k\|}{c\mu_{k-1}\|d_{k-1}\|^2} + \frac{t\|y_{k-1}\|^2\|s_{k-1}\|\|F_k\|}{\|y_{k-1}\|^2(c\mu_{k-1}\|d_{k-1}\|^2)}\right]\|d_{k-1}\|,\\
&\le \theta_k\|F_k\| + \left[\frac{(\mu_{k-1}\|d_{k-1}\|)\|F_k\|}{c\mu_{k-1}\|d_{k-1}\|^2} + \frac{t(\mu_{k-1}\|d_{k-1}\|)\|F_k\|}{c\mu_{k-1}\|d_{k-1}\|^2}\right]\|d_{k-1}\|,\\
&\le \left[\theta_k + \frac{(1+t)}{c}\right]\|F_k\|,\\
&\le \left[\frac{(2+t)}{c}\right]m_1. \qquad (52)
\end{aligned}$$

The last inequality follows from (27). Taking $M = \left[\frac{(2+t)}{c}\right]m_1$, we have (51). □

Theorem 3.7. *Suppose assumptions (A1)–(A3) hold and $\{a_k\}$ be generated by CHCG algorithm. Then*

$$\liminf_{k\to\infty}\|F_k\| = 0. \qquad (53)$$

Proof. Going by contradiction, suppose that (53) is not true, then there exists $n_0 > 0$ such that

$$\|F_k\| \ge \delta_0 \quad \text{holds}, \quad \forall k > 0. \qquad (54)$$

By (33) and Cauchy–Schwartz inequality, we have

$$\|F_k\|^2 \le -F_k^T d_k \le \|F_k\|\|d_k\|, \quad \forall k \in \mathbf{N}. \qquad (55)$$

So,

$$\|d_k\| \ge \delta_0 > 0. \qquad (56)$$

If $\mu_k \ne \xi$, then by the definition of μ_k, $\rho^{-1}\mu_k$ does not satisfy (20) i.e.,

$$-F(a_k + \rho^{-1}\mu_k d_k)^T d_k < \sigma\rho^{-1}\mu_k\|d_k\|^2,$$

by (22) and (55), we have

$$\begin{aligned}
\|F_k\|^2 &= -F_k^T d_k,\\
&= (F(a_k + \rho^{-1}\mu_k d_k) - F_k)^T d_k - F(a_k + \rho^{-1}\mu_k d_k)^T d_k,\\
&\le \rho^{-1}\mu_k\|d_k\|^2 + \sigma\rho^{-1}\mu_k\|d_k\|^2.
\end{aligned}$$

The above inequality implies

$$\mu_k \ge \frac{\rho}{(1+\sigma)}\frac{\|F_k\|^2}{\|d_k\|^2}.$$

From (51) and (54) we have,

$$\mu_k\|d_k\| \ge \frac{\rho}{(1+\sigma)}\frac{\|F_k\|^2}{\|d_k\|} \ge \frac{\rho}{(1+\sigma)}\frac{\delta_0^2}{M}.$$

This contradicts (50). Therefore (53) holds.

The proof is completed. □

The following theorem analyzes the linearly convergent rate of the CHCG method.

Theorem 3.8. *Suppose assumptions (A1)–(A3) hold and $\{a_k\}$ be generated by CHCG algorithm. Then the sequence $dist\{a_k, S^a\}$ Q-linearly converges to 0.*

Table 2
Starting points used to test problems.

Initial Points (IP)	Values
$a1$	$(0.5, 0.5, \ldots, 0.5)^T$
$a2$	$(0.2, 0.2, \ldots, 0.2)^T$
$a3$	$(1, 1, \ldots, 1)^T$
$a4$	$\left(\frac{2}{5}, \frac{2}{5}, \ldots, \frac{2}{5}\right)^T$
$a5$	$\left(0, \frac{1}{2}, \frac{2}{3}, \ldots, 1\right)^T$
$a6$	$\left(\frac{1}{4}, \frac{-1}{4}, \ldots, \frac{(-1)^n}{4}\right)^T$
$a7$	$(4, 4, \ldots, 4)^T$

Proof. Let $u_k = \arg\min\{\|a_k - u\| | u \in S^a\}$. This means u_k is the nearest solution from a_k i.e., $\|a_k - u_k\| = \text{dist}(a_k, S^a)$. Therefore, from the last inequality of (57), for $u_k \in S^a$ we have

$$\begin{aligned} \text{dist}(a_{k+1}, S^a)^2 &\le \|a_{k+1} - u_k\|^2 \\ &\le \text{dist}(a_k, S^a)^2 - \sigma^2\|\alpha_k d_k\|^4 \\ &\le \text{dist}(a_k, S^a)^2 - \sigma^2\alpha_k^4\|F_k\|^4 \\ &\le \text{dist}(a_k, S^a)^2 - \sigma^2\eta^2\alpha_k^4\text{dis}(a_k, S^a)^2 \\ &= (1 - \sigma^2\eta^2\alpha_k^4)\text{dis}(a_k, S^a)^2, \end{aligned} \tag{57}$$

where the third inequality follows from (55) and the fourth inequality follows from (24). By taking $\eta = \frac{1}{\sigma}$, $(1 - \sigma^2\eta^2\alpha_k^4) \in (0, 1)$ holds. This implies that the sequence $\{\text{dist}(a_k, S^a)\}$ Q-linearly converges to 0. □

4. Numerical experiments

Some numerical results are provided in the first part of this section, to show the effectiveness of our method by comparing it with the following existing methods in the literature.

- A projection method for convex constrained monotone nonlinear equations with applications (**PCG**) [32].
- An efficient three-term conjugate gradient method for nonlinear monotone equations with convex constraints (**ETTCG**) [14].

In the second part, CHCG method is applied to solve ℓ_1-norm regularization problem in compressive sensing. The computer codes used are written in Matlab 9.4.0 (R2018a) and run on a personal computer equipped with a 1.80 GHz CPU processor and 8 GB RAM.

4.1. Numerical results

The same line search in (20) used when implementing the three algorithms in this experiments, and the following parameters are set $\xi = 0.5$, $\sigma = 10^{-4}$ and $\rho = 0.9$, we however set $t = 1.2$ in our algorithm. The iteration is set to stop for all the methods if $\|F_k\| \le 10^{-10}$ or when the iterations exceed 1000. We use the symbol '–' to represent failure due to (i) Memory requirement (ii) Number of iterations exceed 1000. We have tried the three methods on the previous four test problems with different initial guess and dimension (n values). The test Problems 1–3 are from [2] while the remaining one is from [5]. To show the extensive numerical experiments of CHCG, PCG and ETTCG methods, the experimentation was carried out with the dimensions 1000, 5000, 10,000, 50,000 and 100,000. We use the following initial points for the test problems (see Table 2).

It should be noted here, that the parameters employed for the experiments conducted are carefully selected to yield the most desired results. In addition, the initial starting points used were selected in an evenly spaced as well as mixed valued fashion to avoid a biased computation.

The following test problems were used in the experiments:

Problem 1.

$$F_1(a) = e^{a_1} - 1,$$
$$F_i(a) = e^{a_i} + a_{i-1} - 1, \quad i = 2, \ldots, n-1,$$

where $\Omega = \mathbb{R}^n_+$.

Problem 2.

$$F_i(a) = 2a_i - \sin|a_i|, \quad i = 1, \ldots, n,$$

where $\Omega = \mathbb{R}^n_+$.

Problem 3.

$$F_i(a) = e^{a_i} - 1, \quad i = 1, 2, \ldots, n,$$

where $\Omega = \mathbb{R}^n_+$.

Problem 4.

$$F_i(a) = \min(\min(|a_i|, a_i^2), \max(|a_i|, a_i^3)), \quad i = 1, 2 \ldots, n,$$

where $\Omega = \mathbb{R}^n_-$.

The reported results of the three (3) methods are shown in Tables 3–6. From the Tables, "IT" represents the total number of iterations, "TIME(s)" represents the CPU time (in seconds), "IP" represents the initial points and "FV" represents the functions evaluations, and $\|F_k\|$ represents the norm of function at the termination point. From Tables 3–6 one can easily see that, the three methods are trying to solve (1), but the improved efficacy of the proposed method is clear. For example, CHCG solves all the problems but ETTCG fails, this is quite clear with Problem 1. In fact, the CHCG approach significantly outperforms the PCG and ETTCG methods for nearly all the problems assessed, since it has the least number of iterations, functions evaluations and CPU time, which are far below the number of iterations, functions evaluations and the CPU time for the PCG and ETTCG methods. This is apparently due to the hybridization of conjugate gradient iterative scheme, correction parameter, as well as Jacobian approximation via acceleration parameter at each iteration.

The results reported in Tables 3–6 are summarized in Table 7 to illustrate which approach is the winner in terms of number of iterations, functions evaluations and CPU time. The summary shows that the CHCG method is the most efficient as it solves 65.00% (91 out of 140) of the problems with least number of iterations than PCG and ETTCG methods which solve 0.00% (0 out of 140) and 3.57% (5 out of 140) respectively. However, the summarized result shows that the two or three methods solve 44 out of 140 problems with the same number of iteration, which translates to 31.43% and is reported as undecided. The table also shows that the CHCG method solves 72.86% (102 out of 140) of the problems with less CPU time compared to PCG and ETTCG methods which solve 20.00% (28 out of 140) and 7.14% (10 out of 140) respectively. Moreover, for the function values, the proposed method solves 48.75% (68 out of 140), PCG method solves 17.86% (25 out of 140), and ETTCG method solves 0.00% (0 out of 140), and 33.57% (47 out of 140) is reported as undecided. Finally, from three Figures and the results in Tables 3–6 it is obvious that our approach is very successful in solving large-scale nonlinear problems.

The results and summary presented in Tables 3–6 and 7 are best described through a pictorial representation. To achieve this, we generate Figs. 1–3 using the performance profiles of Dolan and Moré [9], which shows the performance of each of the three methods. For each problem $\rho \in \mathcal{P}$ and solver $s \in \mathcal{S}$, the performance profile is obtained in terms of the performance measure $t_{\rho,s} > 0$. For any pair (ρ, s) of problem ρ and solver s, the performance ratio is given as

$$r_{\rho,s} = \frac{t_{\rho,s}}{\min\{t_{\rho,s} | s \in \mathcal{S}\}}. \tag{58}$$

Table 3
Numerical results of CHCG, PCG and ETTCG methods for problem 1.

Dimension	IP	CHCG			$\|F_k\|$	PCG			$\|F_k\|$	ETTCG			$\|F_k\|$
		IT	FV	TIME (s)		IT	FV	TIME (s)		IT	FV	TIME (s)	
1000	a1	2	20	0.163684	0	9	13	0.118308	7.16E−12	11	26	0.051611	2.3E−11
	a2	2	19	0.017372	0	8	11	0.022756	6.17E−11	–	–	–	–
	a3	2	21	0.037511	0	9	15	0.01738	5.17E−11	–	–	–	–
	a4	2	20	0.010232	0	9	13	0.016886	7.36E−12	–	–	–	–
	a5	2	21	0.024716	0	21	38	0.039699	1.38E−11	13	29	0.029597	5.43E−11
	a6	1	7	0.010254	0	35	37	0.036661	7.32E−11	32	34	0.034886	7.22E−11
	a7	4	49	0.01853	0	8	33	0.015206	5.74E−12	2	25	0.012648	0
5000	a1	2	20	0.02189	0	9	13	0.029987	1.46E−11	8	19	0.036214	0
	a2	2	19	0.017597	0	9	12	0.025697	5.97E−12	7	16	0.033642	0
	a3	2	21	0.01811	0	9	14	0.03354	1.22E−11	–	–	–	–
	a4	2	20	0.016753	0	9	13	0.026176	1.58E−11	9	21	0.043088	0
	a5	2	21	0.017572	0	22	29	0.061706	1.5E−11	13	29	0.053321	1.2E−11
	a6	1	7	0.008953	0	35	37	0.091173	7.09E−11	32	34	0.093882	7.16E−11
	a7	4	49	0.04442	0	7	32	0.034183	4.68E−11	2	25	0.034272	0
10 000	a1	2	20	0.027082	0	9	13	0.051895	2.04E−11	–	–	–	–
	a2	2	19	0.029402	0	9	12	0.043989	8.13E−12	7	16	0.05215	0
	a3	2	21	0.028343	0	9	14	0.0453	2.29E−11	–	–	–	–
	a4	2	20	0.027543	0	9	13	0.047154	2.23E−11	8	19	0.061535	0
	a5	2	21	0.030014	0	21	32	0.10264	5.12E−11	12	29	0.091707	1.1E−11
	a6	1	7	0.013536	0	35	37	0.14576	7.06E−11	32	34	0.15223	7.15E−11
	a7	4	49	0.071851	0	7	32	0.058053	3.31E−11	2	25	0.051753	0
50 000	a1	2	20	0.108286	0	9	13	0.16896	4.52E−11	8	19	0.232204	0
	a2	2	19	0.094496	0	9	12	0.158822	1.76E−11	7	16	0.196321	0
	a3	2	21	0.103246	0	9	14	0.163749	2.54E−11	–	–	–	–
	a4	2	20	0.100786	0	9	13	0.16446	4.97E−11	9	21	0.260351	0
	a5	2	21	0.101266	0	26	31	0.429475	8.05E−12	12	27	0.321023	9.38E−11
	a6	1	7	0.042233	0	35	37	0.537817	7.03E−11	32	34	0.56907	7.14E−11
	a7	4	49	0.274494	0	7	32	0.209848	1.48E−11	2	25	0.208295	0
100 000	a1	2	20	0.189274	0	9	13	0.322127	6.39E−11	–	–	–	–
	a2	2	19	0.195335	0	9	12	0.32179	2.47E−11	–	–	–	–
	a3	2	21	0.186434	0	9	14	0.338365	2.72E−11	8	20	0.444494	0
	a4	2	20	0.183533	0	9	13	0.31985	7.01E−11	8	19	0.495861	0
	a5	2	21	0.203235	0	27	32	0.861223	6.6E−12	12	27	0.627386	9.67E−11
	a6	1	7	0.077777	0	35	37	1.052519	7.03E−11	32	34	1.118661	7.14E−11
	a7	4	49	0.509883	0	7	32	0.424138	1.05E−11	2	25	0.389371	0

The best solver for a particular problem reaches the lower bound $r_{\rho,s} = 1$. If a solver s fails to meet the convergence test for problem ρ, then $r_{\rho,s}$ is set to infinity. The performance profile of a solver s is defined as

$$p(\tau) = \frac{1}{n_\rho} \text{size}\{\rho \in \mathcal{P} | r_{\rho,s} \leq \tau\}, \tag{59}$$

where n_p is the number of problems. Therefore, $p(\tau)$ is the probability for solver $s \in \mathcal{S}$ that a performance ratio $r_{\rho,s}$ is within a factor $\tau \in \mathbb{R}$ of the best possible ratio. However, the function P is the cumulative distribution function for the performance ratio. For this, we plot the fraction $p(\tau)$ of the problems for which each method is within τ of the smallest number of iterations, CPU time and function evaluations respectively.

Observe that, in Figs. 1–3, the curves corresponding to the CHCG method remain above the other curves representing the PCG and ETTCG methods. This shows that the method proposed outperforms the methods compared in terms of fewer iterations, functions evaluation and CPU time (in seconds) and is therefore the most efficient method. It is vital to state here that the advantages the CHCG method had over the PCG and ETTCG methods includes

(i) the ability to converge much faster to solutions of the problems, which is shown by the final norm value attained for each problem,

Table 4
Numerical results of CHCG, PCG and ETTCG methods for problem 2.

Dimension	IP	CHCG			$\|F_k\|$	PCG			$\|F_k\|$	ETTCG			$\|F_k\|$
		IT	FV	TIME (s)		IT	FV	TIME (s)		IT	FV	TIME (s)	
1000	a1	2	15	0.029374	0	40	42	0.036352	7.66E−11	37	39	0.039826	5.45E−11
	a2	2	15	0.00737	0	39	41	0.048604	6.03E−11	35	37	0.033532	9.12E−11
	a3	2	15	0.007552	0	41	43	0.036147	6.96E−11	37	39	0.036506	9.31E−11
	a4	2	15	0.006319	0	40	42	0.033371	6.25E−11	36	38	0.035451	8.88E−11
	a5	2	15	0.008652	0	41	43	0.036249	6.93E−11	37	39	0.035895	9.27E−11
	a6	1	10	0.003742	0	1	6	0.004953	0	1	6	0.004605	0
	a7	2	20	0.006947	0	38	41	0.037409	7.74E−11	35	38	0.036112	6.3E−11
5000	a1	2	15	0.017194	0	41	43	0.09602	9.1E−11	38	40	0.107244	6.09E−11
	a2	2	15	0.017678	0	40	42	0.094852	7.17E−11	37	39	0.097091	5.1E−11
	a3	2	15	0.016511	0	42	44	0.120174	8.27E−11	39	41	0.102489	5.2E−11
	a4	2	15	0.014199	0	41	43	0.100866	7.42E−11	37	39	0.104148	9.93E−11
	a5	2	15	0.018053	0	42	44	0.119664	8.26E−11	39	41	0.105474	5.2E−11
	a6	1	10	0.008182	0	1	6	0.012946	0	1	6	0.008185	0
	a7	2	20	0.029637	0	39	42	0.088982	9.2E−11	36	39	0.112181	7.04E−11
10 000	a1	2	15	0.027072	0	42	44	0.174262	6.83E−11	38	40	0.201885	8.61E−11
	a2	2	15	0.026655	0	41	43	0.170038	5.38E−11	37	39	0.196989	7.21E−11
	a3	2	15	0.027178	0	43	45	0.17995	6.21E−11	39	41	0.208645	7.36E−11
	a4	2	15	0.025912	0	42	44	0.173002	5.57E−11	38	40	0.189378	7.02E−11
	a5	2	15	0.026861	0	43	45	0.203715	6.21E−11	39	41	0.200187	7.36E−11
	a6	1	10	0.01539	0	1	6	0.010418	0	1	6	0.013979	0
	a7	2	20	0.057997	0	40	43	0.170794	6.91E−11	36	39	0.180222	9.96E−11
50 000	a1	2	15	0.099602	0	43	45	0.711364	8.12E−11	39	41	0.746359	9.63E−11
	a2	2	15	0.096692	0	42	44	0.66636	6.39E−11	38	40	0.721019	8.06E−11
	a3	2	15	0.098632	0	44	46	0.75902	7.38E−11	40	42	0.756109	8.23E−11
	a4	2	15	0.098672	0	43	45	0.691368	6.62E−11	39	41	0.730112	7.85E−11
	a5	2	15	0.103324	0	44	46	0.696946	7.38E−11	40	42	0.755147	8.23E−11
	a6	1	10	0.054669	0	1	6	0.038919	0	1	6	0.049837	0
	a7	2	20	0.240239	0	41	44	0.649783	8.21E−11	38	41	0.73583	5.57E−11
100 000	a1	2	15	0.184932	0	44	46	1.317218	6.1E−11	40	42	1.420589	6.81E−11
	a2	2	15	0.177485	0	42	44	1.285065	9.04E−11	39	41	1.390621	5.7E−11
	a3	2	15	0.194041	0	45	47	1.362545	5.54E−11	41	43	1.441766	5.82E−11
	a4	2	15	0.189785	0	43	45	1.303302	9.36E−11	40	42	1.425622	5.55E−11
	a5	2	15	0.189841	0	45	47	1.374635	5.54E−11	41	43	1.438479	5.82E−11
	a6	1	10	0.09437	0	1	6	0.07481	0	1	6	0.097027	0
	a7	2	20	0.476423	0	42	45	1.332267	6.17E−11	38	41	1.351728	7.87E−11

(ii) better performance with uniform as well as mixed valued initial starting points.

The limitations of the CHCG method lie in the fact that if the correction parameter t_k is assigned outside the interval $(1, 2)$, then it takes much time to converge and in some instances diverges.

4.2. Applications in compressive sensing

This part is dedicated to application of the proposed algorithm (CHCG) to signal recovery problems in compressive sensing. This is the process for the efficient acquisition and reconstruction of a signal. It compresses the signal acquired at the time of the sensing. Its application rises in many applications including the statistics and signal processing [11,48]. The most prominent approach requires optimizing problems in sparse recovery is represented by a convex unconstrained optimization problem:

$$\min_{a} \frac{1}{2}\|w - Qa\|_2^2 + \tau\|a\|_1, \tag{60}$$

Table 5
Numerical results of CHCG, PCG and ETTCG methods for problem 3.

Dimension	IP	CHCG			$\|F_k\|$	PCG			$\|F_k\|$	ETTCG			$\|F_k\|$
		IT	FV	TIME (s)		IT	FV	TIME (s)		IT	FV	TIME (s)	
1000	a1	2	16	0.021986	0	39	41	0.036379	9.02E−11	36	38	0.035475	6.77E−11
	a2	2	15	0.006916	0	38	40	0.036641	9.39E−11	35	37	0.030722	7.51E−11
	a3	2	18	0.005059	0	39	41	0.033073	7.45E−11	36	38	0.036125	5.62E−11
	a4	2	16	0.006708	0	39	41	0.033011	8.08E−11	36	38	0.035223	6.07E−11
	a5	2	18	0.006346	0	39	41	0.033247	7.83E−11	36	38	0.035404	5.92E−11
	a6	1	2	0.002984	0	1	2	0.002689	0	1	2	0.002352	0
	a7	1	13	0.005176	0	1	13	0.00729	0	1	13	0.005587	0
5000	a1	2	16	0.014442	0	41	43	0.092703	5.69E−11	37	39	0.083218	7.56E−11
	a2	2	15	0.013644	0	40	42	0.090625	5.93E−11	36	38	0.085523	8.4E−11
	a3	2	18	0.013496	0	40	42	0.079467	8.85E−11	37	39	0.088383	6.29E−11
	a4	2	16	0.019168	0	40	42	0.084485	9.6E−11	37	39	0.091678	6.78E−11
	a5	2	18	0.014326	0	40	42	0.085871	8.96E−11	37	39	0.10509	6.37E−11
	a6	1	2	0.004804	0	1	2	0.005276	0	1	2	0.004213	0
	a7	1	13	0.010387	0	1	13	0.009144	0	1	13	0.014674	0
10 000	a1	2	16	0.021618	0	41	43	0.144399	8.05E−11	38	40	0.157972	5.35E−11
	a2	2	15	0.020327	0	40	42	0.137939	8.38E−11	37	39	0.148613	5.94E−11
	a3	2	18	0.019475	0	41	43	0.143064	6.65E−11	37	39	0.14805	8.89E−11
	a4	2	16	0.023093	0	41	43	0.147929	7.21E−11	37	39	0.148313	9.59E−11
	a5	2	18	0.021758	0	41	43	0.141932	6.69E−11	37	39	0.147792	8.95E−11
	a6	1	2	0.008368	0	1	2	0.006086	0	1	2	0.006806	0
	a7	1	13	0.015422	0	1	13	0.018181	0	1	13	0.024177	0
50 000	a1	2	16	0.065526	0	42	44	0.600517	9.56E−11	39	41	0.590405	5.98E−11
	a2	2	15	0.075864	0	41	43	0.58302	9.95E−11	38	40	0.571569	6.64E−11
	a3	2	18	0.072682	0	42	44	0.537692	7.89E−11	38	40	0.557761	9.94E−11
	a4	2	16	0.066479	0	42	44	0.531169	8.56E−11	39	41	0.581891	5.36E−11
	a5	2	18	0.07032	0	42	44	0.543927	7.91E−11	38	40	0.558955	9.95E−11
	a6	1	2	0.016532	0	1	2	0.019537	0	1	2	0.019875	0
	a7	1	13	0.060173	0	1	13	0.058134	0	1	13	0.095164	0
100 000	a1	2	16	0.120074	0	43	45	1.049815	7.18E−11	39	41	1.166702	8.45E−11
	a2	2	15	0.1372	0	42	44	1.032128	7.48E−11	38	40	1.185053	9.39E−11
	a3	2	18	0.134959	0	43	45	1.058502	5.93E−11	39	41	1.116671	7.03E−11
	a4	2	16	0.124429	0	43	45	1.057841	6.44E−11	39	41	1.114056	7.58E−11
	a5	2	18	0.133714	0	43	45	1.04834	5.94E−11	39	41	1.109764	7.03E−11
	a6	1	2	0.038047	0	1	2	0.038581	0	1	2	0.040341	0
	a7	1	13	0.10898	0	1	13	0.106661	0	1	13	0.179577	0

where $a \in \mathbb{R}^n$, $w \in \mathbb{R}^k$, $Q \in \mathbb{R}^{k\times n}(k \ll n)$ denotes a linear operator, the parameter $\tau \geq 0$, and $\|a\|_1 = \sum_{i=1}^{n} |a_i|$. In the literature of compressive sensing, (60) is often called Basis Pursuit Denoising Problem (BPDN) or ℓ_1-regularized least square problem. Some iterative methods for solving (60), can be found in [10,11,33]. But, gradient-based methods are the most prominent [11]. In the scheme, problem (60) is expressed as follows. Any vector $a \in \mathbb{R}^n$ is parted into the positive and negative portions as:

$$a = p - q, \quad p \geq 0, \quad q \geq 0, \quad p, q \in \mathbb{R}^n. \tag{61}$$

Let $p_i = (a_i)_+$, and $q_i = (-a_i)_+$ for $i = 1, 2, \ldots, n$, where $(.)_+$ is the positive operator, which is defined as $(a)_+ = \max\{0, a\}$. Applying the definition of the ℓ_1-norm, we have $\|a\|_1 = e_n^T p + e_n^T q$, with $e_n = (1, 1, 1, \ldots, 1)^T \in \mathbb{R}^n$. So, problem (60) can be reformulated as the following

$$\min_{p,q} \frac{1}{2}\|w - Q(p-q)\|_2^2 + \tau e_n^T p + \tau e_n^T q, \quad p, q \geq 0. \tag{62}$$

Table 6
Numerical results of CHCG, PCG and ETTCG methods for problem 4.

Dimension	IP	CHCG			$\|F_k\|$	PCG			$\|F_k\|$	ETTCG			$\|F_k\|$
		IT	FV	TIME (s)		IT	FV	TIME (s)		IT	FV	TIME (s)	
1000	a1	1	2	0.019786	0	1	2	0.005798	0	1	2	0.004697	0
	a2	1	2	0.00311	0	1	2	0.00499	0	1	2	0.00424	0
	a3	1	3	0.003708	0	1	2	0.003718	0	1	2	0.003332	0
	a4	1	2	0.0039	0	1	2	0.003028	0	1	2	0.004362	0
	a5	6	51	0.075968	0	1	2	0.006383	0	1	2	0.004437	0
	a6	2	15	0.014309	0	39	41	0.07612	7.61E−11	36	38	0.075619	5.75E−11
	a7	1	3	0.003759	0	1	2	0.003951	0	1	2	0.003512	0
5000	a1	1	2	0.009895	0	1	2	0.009894	0	1	2	0.00961	0
	a2	1	2	0.00818	0	1	2	0.010049	0	1	2	0.010358	0
	a3	1	3	0.008176	0	1	2	0.009523	0	1	2	0.008871	0
	a4	1	2	0.008361	0	1	2	0.009188	0	1	2	0.009654	0
	a5	6	52	0.10194	0	1	2	0.009636	0	1	2	0.008926	0
	a6	2	15	0.033915	0	40	42	0.280228	9.04E−11	37	39	0.280049	6.43E−11
	a7	1	3	0.008834	0	1	2	0.00833	0	1	2	0.008089	0
10 000	a1	1	2	0.015593	0	1	2	0.013689	0	1	2	0.015973	0
	a2	1	2	0.014588	0	1	2	0.015749	0	1	2	0.017866	0
	a3	1	3	0.014447	0	1	2	0.010881	0	1	2	0.014791	0
	a4	1	2	0.013478	0	1	2	0.016782	0	1	2	0.017438	0
	a5	6	52	0.185686	0	1	2	0.014733	0	1	2	0.015988	0
	a6	2	15	0.057634	0	41	43	0.480515	6.79E−11	37	39	0.514008	9.09E−11
	a7	1	3	0.012115	0	1	2	0.012953	0	1	2	0.014184	0
50 000	a1	1	2	0.05701	0	1	2	0.053925	0	1	2	0.068732	0
	a2	1	2	0.054249	0	1	2	0.055785	0	1	2	0.066109	0
	a3	1	3	0.050533	0	1	2	0.043045	0	1	2	0.05295	0
	a4	1	2	0.056469	0	1	2	0.057981	0	1	2	0.064508	0
	a5	5	51	0.739303	0	1	2	0.056216	0	1	2	0.062733	0
	a6	2	15	0.245875	0	42	44	2.118536	8.06E−11	39	41	2.375988	5.08E−11
	a7	1	3	0.047474	0	1	2	0.047494	0	1	2	0.052935	0
100 000	a1	1	2	0.113308	0	1	2	0.113189	0	1	2	0.129614	0
	a2	1	2	0.106407	0	1	2	0.111957	0	1	2	0.132337	0
	a3	1	3	0.094876	0	1	2	0.081419	0	1	2	0.102163	0
	a4	1	2	0.103429	0	1	2	0.118873	0	1	2	0.133462	0
	a5	5	51	1.457223	0	1	2	0.118314	0	1	2	0.12823	0
	a6	2	15	0.474047	0	43	45	4.310932	6.06E−11	39	41	4.837805	7.19E−11
	a7	1	3	0.098135	0	1	2	0.086115	0	1	2	0.109128	0

Table 7
Summary of test results reported in Tables 3–6.

Methods	NI	Percentage	FEV	Percentage	TIME (s)	Percentage
CHCG	91	65.00%	68	48.57%	102	72.86%
PCG	0	0.00%	25	17.86%	28	20.00%
ETTG	5	3.57%	0	0.00%	10	7.14%
Undecided	44	31.43%	47	33.57%	0	0.00%

It was shown in [11] that problem (62) can be formulated in more standard bound-constrained quadratic program as follows

$$\min_{z} \frac{1}{2} z^T H z + c^T z, \quad \text{s.t} \quad z \geq 0, \tag{63}$$

where

$$z = \begin{pmatrix} p \\ q \end{pmatrix}, \; c = \tau e_{2n} + \begin{pmatrix} -h \\ h \end{pmatrix}, \; h = Q^T w, \; H = \begin{pmatrix} Q^T Q & -Q^T Q \\ -Q^T Q & Q^T Q \end{pmatrix}.$$

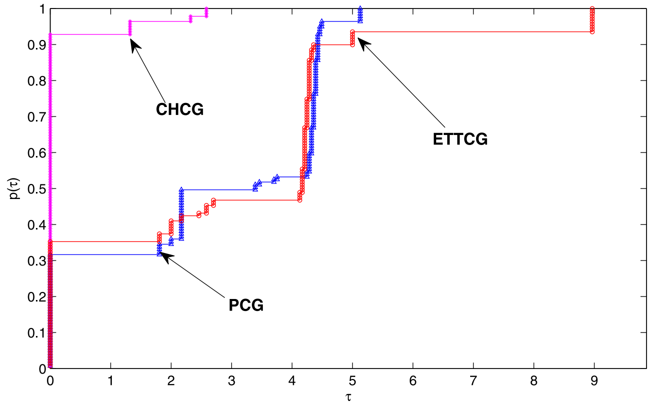

Fig. 1. Performance profile for the number of iterations.

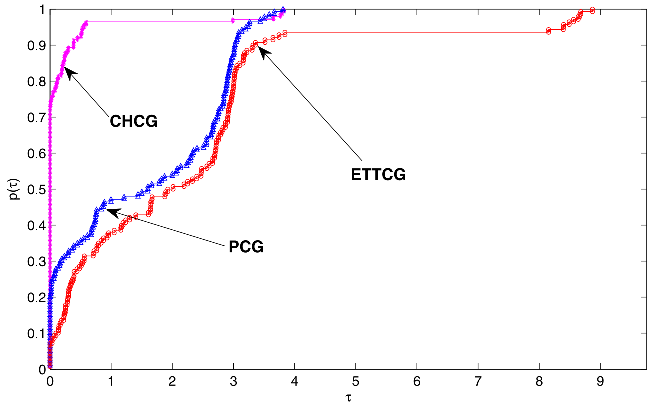

Fig. 2. Performance profile for the CPU time (in second).

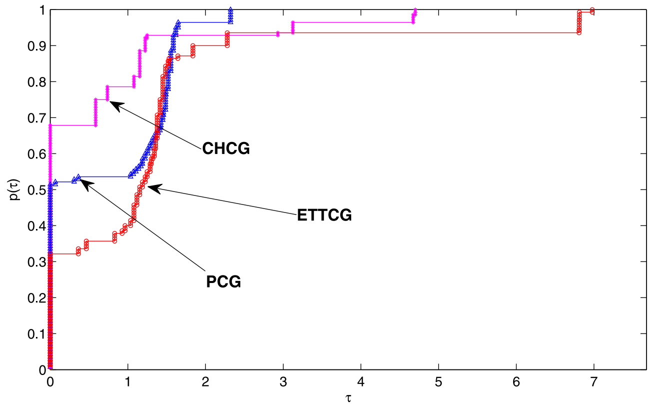

Fig. 3. Performance profile for the functions evaluations.

Clearly, the matrix H is positive semi-definite. So, problem (63) is a convex quadratic programming problem. Xiao et al. [57] translated (63) into a problem of linear variable inequality (LVI), which is equivalent to a linear complementary problem if and only if is a solution of the nonlinear equations defined by

$$F(z) = \min\{z,\ \mathrm{Hz} + c\} = 0, \tag{64}$$

Table 8
Ten numerical results with their average of ℓ_1-norm regularization problem for CHCG and PCG methods.

	CHCG			PCG		
	MSE	IT	TIME (s)	MSE	Iter	TIME (s)
	1.05E−05	118	3.17	3.47E−05	146	4.50
	2.54E−05	119	3.63	5.19E−05	126	3.38
	1.72E−05	90	2.47	2.36E−05	125	3.31
	2.36E−05	102	3.81	2.27E−05	142	3.58
	3.42E−05	117	3.51	4.51E−05	144	3.75
	2.63E−05	122	3.22	5.54E−05	135	3.86
	2.14E−05	130	3.70	2.50E−05	147	3.72
	3.56E−05	139	3.81	5.56E−05	146	4.56
	2.61E−05	124	3.53	3.69E−05	153	4.13
	2.40E−05	114	3.33	3.03E−05	151	5.20
Average	2.44E−05	117.5	3.418	3.81E−05	141.5	3.999

where, F is a vector-valued function and the "min" interpreted as component-wise minimum. Pang [40] and Xiao et al. [57] showed that the mapping F is Lipschitz continuous and monotone. So, problem (60) can be translated into (1). Therefore, our algorithm (CHCG) can be applied to solve it.

To further highlight the performance of the CHCG scheme, some numerical experiments were carried out. We begin by considering a typical compressive sensing problem in which the main aim is to reconstruct a sparse signal of length$-n$ from k observations. The mean of squared error (MSE) to the original signal $\hat{a}$ is used to assess the quality of the restoration i.e.,

$$MSE = \frac{1}{n}\|\hat{a} - \tilde{a}\|^2, \tag{65}$$

where $\tilde{a}$ denotes the restored or recovered signal. In the experiment, the size of the signal is tested with $n = 2^{12}$ and $k = 2^{10}$. However, the original signal contains 2^7 randomly nonzero elements. Also, the measurement w is distributed by noise, $w = Q\hat{a} + \varphi$ where φ is the Gaussian noise distributed as $N(0, 10^{-4})$ and Q is the Gaussian matrix generated in Matlab by the command $randn(k, n)$.

In order to compare the performance of the CHCG scheme in compressive sensing, we compare it with the effective PCG method [32]. For both methods the parameters are set as $\sigma = 10^{-4}$, $\rho = 0.5$, and $\xi = 10$, we however set the parameter r = 0.2 in PCG method. The merit function $f(a) = \frac{1}{2}\|w - Qa\|_2^2 + \tau\|a\|_1$ is also used. However, the same continuation technique on the parameter τ is used to show nature of convergence of each method. In addition, the measuring signal begins the iterative process for the experiment, i.e, $a_0 = Q^T w$ and terminate when the objective function's relative change satisfies

$$\frac{|f(a_k) - f(a_{k-1})|}{|f(a_{k-1})|} < 10^{-5}. \tag{66}$$

To highlight the efficiency of the proposed CHCG method, we test it against PCG solver that is applied to solve monotone equations and has application in compressive sensing. For both methods, we carried out ten (10) experiments with different samples of noise, the average of the experiments is also computed and the results are reported in Table 6.

Furthermore, from Fig. 4, almost exactly the disturbed signal has been restored by the two methods. Nonetheless, to show the performance of these methods visibly, four graphs are plotted to exhibit the convergence behavior of the two methods through their mean square error (MSE) and function evaluations results, as well as the number of iterations and CPU time respectively. As can be seen from Fig. 5, our proposed method exhibits much faster descent rates of MSE and function evaluations than the PCG method. It can also be seen from the figures that the CHCG method requires less number of iterations and CPU time in order to recover the original signal compared to that required by the PCG method for the same process (see Table 8).

5. Conclusion

In this paper, a convex constrained hybrid conjugate gradient method for monotone nonlinear equations with application to signal recovery is presented. This was achieved by combining the direction proposed in [20] with the

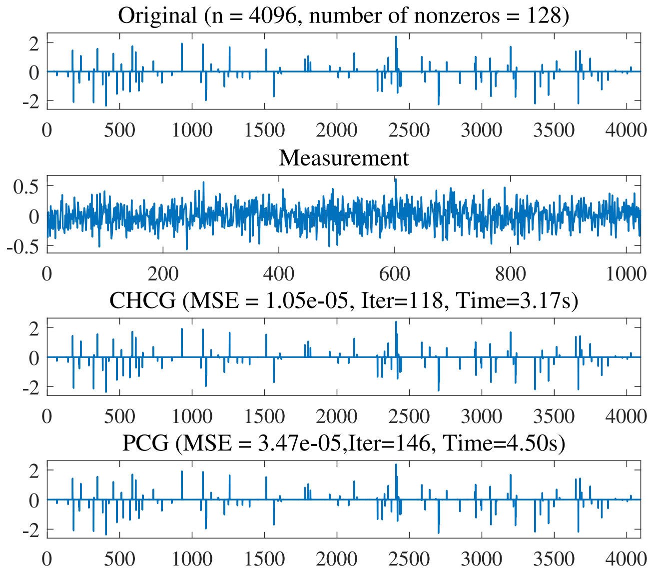

Fig. 4. The original signal, the measurement, and the recovery signals by CHCG and PCG methods.

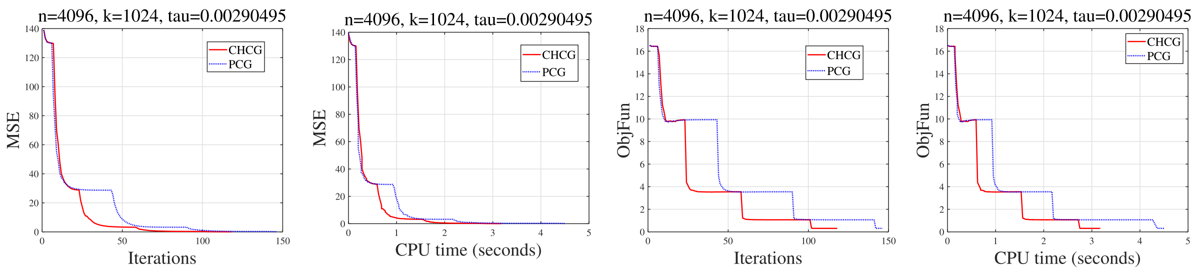

Fig. 5. Comparison of the results of CHCG and PCG methods, where MSE and ObjFun stand for mean of squared error and function values respectively.

classical conjugate gradient direction by applying Picard–Mann hybrid iterative process and derived an effective conjugate gradient parameter. It is a completely matrix-free method that is globally convergent under certain appropriate conditions. Numerical comparisons have been made using large-scale test problems. Furthermore, Tables and Figures showed that the proposed method is practically quite welcome, because it has the least number of iteration and CPU time compared to PCG and ETTCG methods respectively. In addition, the proposed method is successfully applied to deal with the experiments on the ℓ_1-norm regularization problem in compressive sensing, where ten (10) experiments are conducted with different samples of noise, and the experiments average values are also reported in Table 7 which clearly showed a better efficiency for CHCG method. Finally, Fig. 5 indicated that the CHCG method had better restoration of the disturbed signal because it had the lowest mean square error (MSE) value, iteration number, and CPU time than PCG method. In the future research, Eigenvalue analysis will be applied to CHCG scheme in order to obtain the appropriate value of the correction parameter t_k to enhance the convergence speed of CHCG method.

CRediT authorship contribution statement

Abubakar Sani Halilu: Methodology, Software, Data curation, Writing - original draft, Writing - review & editing. **Arunava Majumder:** Conceptualization, Methodology, Validation, Formal analysis, Investigation, Writing - review & editing, Supervision. **Mohammed Yusuf Waziri:** Conceptualization, Methodology, Validation, Data curation, Supervision. **Kabiru Ahmed:** Conceptualization, Methodology, Validation, Writing - review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] H. Abdullahi, A.S. Halilu, M.Y. Waziri, A modified conjugate gradient method via a double direction approach for solving large-scale Symmetric Nonlinear Systems, J. Numer. Math. Stoch. 10 (1) (2018) 32–44.

[2] A.B. Abubakar, P. Kumam, A.M. Awwal, Global convergence via descent modified three-term conjugate gradient projection algorithm with applications to signal recovery, Res. Appl. Math. 4 (2019) 1–20.

[3] N. Andrei, A hybrid conjugate gradient algorithm for unconstrained optimization as a convex combination of Hestenes-Stiefel and Dai-Yuan, Stud. Inform. Control 17 (4) (2008) 55–70.

[4] W. Cheng, A PRP type method for systems of monotone equations, Math. Comput. Modelling 50 (2009) 15–20.

[5] W.L. Cruz, A Spectral algorithm for large-scale systems of nonlinear monotone equations, Numer. Algorithms (2017) http://dx.doi.org/10.1007/s1107s-017-0299-8.

[6] Y.H. Dai, L.Z. Liao, New conjugacy conditions and related nonlinear conjugate gradient methods, Appl. Math. Optim. 3 (1) (2001) 87–101.

[7] Y.H. Dai, Y.X. Yuan, A nonlinear conjugate gradient method with a strong global convergence property, SIAM J. Optim. 10 (1999) 177–182.

[8] J.E. Dennis, R.B. Schnabel, Numerical Methods for Unconstrained Optimization and Nonlinear Equations, Prentice Hall, Englewood Cliffs, NJ., 1983.

[9] E. Dolan, J. Moré, Benchmarking opyimization software with performance profiles, J. Math. Prog. 91 (2002) 201–213.

[10] T. Elaine, Y. Wotao, Z. Yin, A Fixed-Point Continuation Method for ℓ_1-Regularized Minimization with Applications To Compressed Sensing, CAAM TR07-07, Rice Univ., 2007, pp. 43–44.

[11] M. Figueiredo, R. Nowak, S.J. Wright, Gradient Projection for Sparse Reconstruction, Application to Compressed Sensing and Other Inverse Problems, IEEE J-STSP IEEE Press, Piscataway, NJ, 2007, pp. 586–597.

[12] R. Fletcher, Practical Methods of Optimization, John Wiley and Sons, New york, 1987.

[13] R. Fletcher, C.M. Reeves, Function minimization by conjugate gradient methods, J. Comput. 7 (1964) 149–154.

[14] P. Gao, C. He, An efficient three-term conjugate gradient method for nonlinear monotone equations with convex constraints, Calcolo 53 (2018) 17.

[15] W.W. Hager, H. Zhang, A survey of nonlinear conjugate gradient methods, Pac. J. Optim. 2 (1) (2006) 35–58.

[16] A.S. Halilu, M.K. Dauda, M.Y. Waziri, M. Mamat, A derivative-free decent method via acceleration parameter for solving systems of nonlinear equations, Open J. Sci. Technol. 2 (3) (2019) 1–4.

[17] A.S. Halilu, A. Majumder, M.Y. Waziri, H. Abdullahi, Double direction and step length method for solving system of nonlinear equations, Eur. J. Mol. Clin. Med. 7 (7) (2020) 3899–3913.

[18] A.S. Halilu, M.Y. Waziri, En enhanced matrix-free method via double step length approach for solving systems of nonlinear equations, Int. J. Appl. Math. Res. 6 (2017) 147–156.

[19] A.S. Halilu, M.Y. Waziri, A transformed double step length method for solving large-scale systems of nonlinear equations, J. Numer. Math. Stoch. 9 (2017) 20–32.

[20] A.S. Halilu, M.Y. Waziri, An improved derivative-free method via double direction approach for solving systems of nonlinear equations, J. Ramanujan Math. Soc. 33 (2018) 75–89.

[21] A.S. Halilu, M.Y. Waziri, Inexact double step length method for Solving Systems of Nonlinear Equations, Stat. Optim. Inf. Comput. 8 (2020) 165–174.

[22] A.S. Halilu, M.Y. Waziri, Solving system of nonlinear equations using improved double direction method, J. Nigerian Math. Soc. 39 (2) (2020) 287–301.

[23] A.S. Halilu, M.Y. Waziri, I. Yusuf, Efficient matrix-free direction method with line search for solving large-scale system of nonlinear equations, Yugosl. J. Oper. Res. 30 (4) (2020) 399–412.

[24] M.R. Hestenes, E. Stiefel, Method of conjugate gradient for solving linear systems, J. Res. Natl. Bur. Stand. 49 (1952) 409–436.

[25] C. Kanzow, N. Yamashita, M. Fukushima, Levenberg–Marquardt methods for constrained nonlinear equations with strong local convergence properties, J. Comput. Appl. Math. 172 (2004) 375–397.

[26] R.I. Kačurovskii, On monotone operators and convex functionals, Usp. Math. Nauk. 15 (4) (1960) 213–215.

[27] S.H. Khan, A Picard-Mann hybrid iterative process, Fixed Point Theory Appl. (2013) 69.

[28] K. Levenberg, A method for the solution of certain non-linear problems in least squares, Q. Appl. Math. 2 (1944) 164–166.

[29] D. Li, M.M. Fukushima, A global and superlinear convergent Gauss-Newton based BFGS method for symmetric nonlinear equation, SIAM J. Numer. Anal. 37 (1999) 152–172.

[30] M. Li, H. Liu, Z. Liu, A new family of conjugate gradient methods for unconstrained optimization, J. Appl. Math. Comput. 58 (2018) 219–234.

[31] Z. Li, Z. Weijun, L. Dong-Hui, A descent modified Polak-Ribiére-Polyak conjugate gradient method and its global convergence, IMA J. Numer. Anal. 26 (4) (2006) 29–40.

[32] J.K. Liu, S.J. Li, A projection method for convex constrained monotone nonlinear equations with applications, Comput. Math. Appl. 70 (10) (2015) 2442–2453.

[33] A.T. Mario, R. Figueiredo, D. Nowak, An EM algorithm for wavelet-based image restoration, IEEE Trans. Image Process. 12 (8) (2003) 906–916.
[34] D.W. Marquardt, An algorithm for least-squares estimation of nonlinear parameters, SIAM J. Appl. Math. 11 (1963) 431–441.
[35] K. Meintjes, A.P. Morgan, A methodology for solving chemical equilibrium systems, Appl. Math. Comput. 22 (1987) 333–361.
[36] G.J. Minty, Monotone (nonlinear) operators in Hilbert space, Duke Math. J. 29 (3) (1962) 341–346, http://dx.doi.org/10.1215/S0012-7094-62-02933-2.
[37] Y. Narushima, A smoothing conjugate gradient method for solving systems of nonsmooth equations, Appl. Math. Comput. 219 (16) (2013) 46–55.
[38] Y. Narushima, H. Yabe, J. Ford, A three-term conjugate gradient method with sufficient descent property for unconstrained optimization, SIAM J. Optim. 21 (1) (2011) 12–30.
[39] J. Nocedal, S.J. Wright, Numerical Optimization, in: Springer Series in Operations Research, Springer, New York, 1999.
[40] J.S. Pang, Inexact Newton methods for the nonlinear complementarity problem, Math. Program. 1 (1986) 54–71.
[41] M. Petrovic, V. Racocevic, N. Kontrec, S. Panic, D. Ilic, Hybridization of accelerated gradient descent method, Numer. Algorithms 79 (2018) 769–786.
[42] M.J. Petrovic, P.S. Stanimirovic, Accelerated gradient descent methods with line search, Numer. Algorithms 51 (2010) 503–520.
[43] M.J. Petrovic, P.S. Stanimirovic, Accelerated double direction method for solving unconstrained optimization problems, Math. Probl. Eng. (2014).
[44] M.J. Petrovic, P.S. Stanimirovic, Hybrid modification of Accelerated Double Direction Method, Math. Probl. Eng. 2018 (2018) http://dx.doi.org/10.1155/2018/1523267.
[45] B.T. Polyak, The conjugate gradient method in extreme problems, USSR Comput. Math. Math. Phys. 9 (1969) 94–112.
[46] M.V. Solodov, B.F. Svaiter, A globally convergent inexact Newton method for systems of monotone equations, in: Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods, Springer, 1999, pp. 355–369.
[47] M. Sun, M.Y. Tian, Y.J. Wang, Multi-step discrete-time Zhang neural networks with application to time-varying nonlinear optimization, Discrete Dyn. Nat. Soc. (2019) 4745759, 1–14.
[48] R. Tibshirani, Regression shrinkage and selection via the lasso, J. R. Stat. Soc. 58 (1) (1996) 267–288.
[49] C.W. Wang, Y.J. Wang, A super linearly convergent projection method for constrained systems of nonlinear equations, J. Global Optim. 44 (2) (2009) 283–296.
[50] C. Wang, Y. Wang, C. Xu, A projection method for a system of nonlinear monotone equations with convex constraints, Math. Methods Oper. Res. 66 (1) (2007) 33–46.
[51] M.Y. Waziri, K. Ahmad, J. Sabiu, A Dai-Liao conjugate gradient method via modified secant equation for system of nonlinear equations, Arab. J. Math. (2019) http://dx.doi.org/10.1007/s40065-019-0264-6.
[52] M.Y. Waziri, K. Ahmad, J. Sabiu, A family of Hager-Zhang conjugate gradient methods for system of monotone nonlinear equations, Appl. Math. Comput. 361 (2019) 645–660.
[53] M.Y. Waziri, K. Ahmad, J. Sabiu, A.S. Halilu, Enhanced Dai-Liao conjugate gradient methods for systems of monotone nonlinear equations, Appl. Math. Comput. (2020) http://dx.doi.org/10.1007/s40324-020-00228-9.
[54] M.Y. Waziri, W.J. Leong, M.A. Hassan, Jacobian-Free Diagonal Newton Method for solving nonlinear systems with Singular Jacobian, Malays. J. Math. Sci. 5 (2011) 241–255.
[55] M.Y. Waziri, H.U. Muhammad, A.S. Halilu, K. Ahmad, A modified matrix-free methods for solving system of nonlinear equations, OPt. (2020) 1–20, http://dx.doi.org/10.1080/02331934.2020.1778689.
[56] M.Y. Waziri, J. Sabiu, A derivative-free conjugate gradient method and its global convergence for symmetric nonlinear equations, J. Math. Math. Sci. (2015) 1–8.
[57] Y. Xiao, Q. Wang, Q. Hu, Nonsmooth equations based method for l1-norm problems with applications to compressed sensing, Nonlinear Anal. Theory Methods Appl. 74 (11) (2011) 3570–3577.
[58] Y.H. Xiao, H. Zhu, A conjugate gradient method to solve convex constrained monotone equations with applications in compressive sensing, J. Math. Anal. Appl. 405 (2013) 310–319.
[59] Q.R. Yana, X.Z. Penga, D.H. Li, A globally convergent derivative-free method for solving large-scale nonlinear monotone equations, J. Comput. Appl. Math. 234 (2010) 649–657.
[60] G. Yu, A derivative-free method for solving large-scale nonlinear systems of equations, J. Ind. Manag. Optim. 6 (2010) 149–160.
[61] G. Yuan, X. Lu, A new backtracking inexact BFGS method for symmetric nonlinear equations, Comput. Math. Appl. 55 (2008) 116–129.
[62] E.H. Zarantonello, Solving Functional Equations by Contractive Averaging, Tech. Rep. 160, U. S. Army Math. Research Center, Madison, Wisconsin, 1960.
[63] W. Zhou, D. Li, Limited memory BFGS method for nonlinear monotone equations, J. Comput. Math. 25 (1) (2007) 89–96.