Optimizing Conjugate Gradient Directions for Image Deblurring in Compressed Sensing: A Hybridized Approach

A.K. Awasthi, *Member, IAENG,* Yogesh Kumar, and Habibu Abdullahi

***Abstract*—Hybridization emerges as a promising strategy for refining conjugate gradient (CG) directions, pivotal in tackling image deblurring within compressed sensing frameworks. This abstract introduces a novel hybrid CG parameterization, surpassing the recent SGCS algorithm (Auwal et al., 2020). The method integrates a convex combination approach, dynamically updating the combination parameter (μ_k) via the Dai and Liao conjugacy condition. Additionally, a derivative-free line search efficiently determines optimal step lengths (α_k). With a focus on satisfying sufficient descent conditions, imperative for global convergence, the method demonstrates superior performance through numerical experiments compared to existing approaches. Furthermore, the extension of this method to ℓ_1-norm regularized problems enhances its efficacy in restoring blurred images within compressed sensing contexts.**

***Index Terms*—Convex Constraints, Hybridization Parameter, Conjugate gradient, Image restoration.**

I. Introduction

IN this approach, a constrained monotone system of nonlinear equations is given

$$\psi(x) = 0, \quad x \in \Omega, \tag{1}$$

is considered, such that $\psi : \mathbf{R^n} \to \mathbf{R^n}$ is continuous and monotone. The term monotone entails that, the system (1) satisfies the following inequality

$$(\psi(x) - \psi(y))^T (x - y) \geq 0 \quad \forall x, y \in \mathbf{R^n}. \tag{2}$$

Moreover, the constrained set $\Omega \in \mathbf{R^n}$ is a non-empty closed and convex set. Throughout this paper, $\mathbf{R^n}$ and $\|.\|$ are referring to the Euclidean norm and real n-dimensional space respectively and $\psi_k = \psi(x_k)$.

The monotone system is among categories of systems of nonlinear equations investigated by many scholars, such as Zarantonello [3], Kačurovskii [4], Minty [5] etc. Studies of monotone systems became pertinent for their usability in finding economic equilibrium and chemical equilibrium [36]. Monotone systems were recently applied in ℓ_1-norm regularized problems to restore noisy signals or recover blurred images in compressive sensing [1]. Dozens of techniques have been investigated for solutions to the monotone system. The suitable techniques are Newton's and quasi-Newton's methods [6]. Gauss-Newton's [7] and Levenberg-Marquardt's methods [8] are among the numerous techniques for solving the monotone system. Despite the attractive features of these methods, they are not appropriate for high-dimension problems; this is because of the computation of the Jacobian Matrix and storage requirement. [10].

The attention of researchers turned to CG methods because of their ease of use and low storage requirements, which enabled them to solve high-dimension systems of practical importance. This is why recently developed methods for constrained systems with applications to various fields use CG direction [13]. CG methods eliminated most of the disadvantages associated with previous methods. The CG method uses the following direction.

$$d_k = \begin{cases} -\psi_k, & \text{if} \quad k = 0, \\ -\psi_k + \beta_k d_{k-1}, & \text{if} \quad k \geq 1, \end{cases} \tag{3}$$

where, β_k, is a conjugate gradient parameter.

The parameter β_k is the most critical component of any CG method because different CG parameters correspond to different CG directions with unique numerical performance and convergence properties. Since any newly successfully formulated CG parameter must be numerically sound and converged globally [31], [28]. The numerical performance of the CG parameter can be checked by comparing it with existing CG parameters. Likewise, its global convergence is guaranteed by its ability to satisfy a descent/sufficient descent condition[31]. Defining and producing a new CG parameter, β_k, is challenging; therefore, researchers resorted to hybridize and modify the existing CG parameters (classical CG). These classical CG parameters are of two categories: good numerical performance but uncertain convergence results. Under this category, we have;

$$\begin{aligned} \beta_k^{PRP} &= \frac{\psi_k^T y_{k-1}}{\|\psi_{k-1}\|^2}, \\ \beta_k^{HS} &= \frac{\psi_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}, \\ \beta_k^{LS} &= \frac{\psi_k^T y_{k-1}}{-\psi_{k-1}^T d_{k-1}}. \end{aligned} \tag{4}$$

as proposed in [14], [15], [16].

The second category comprises CG parameters with strong convergent properties but weak numerical performance. They

Manuscript received March 6, 2024; revised September 22, 2024.

A.K. Awasthi is a Professor in the Department of Mathematics, School of Chemical Engineering and Physical Sciences, Lovely Professional University, Phagwara, India, 144411. (corresponding author phone: +91-9315517628; e-mail:amit.25155@lpu.co.in)

Yogesh Kumar is a Ph.D. student in the Department of Mathematics, School of Chemical Engineering and Physical Sciences, Lovely Professional University, Phagwara, Punjab 144411, India. (corresponding author:e-mail:ror.happy@gmail.com)

Habibu Abdullahi is an Assistant Professor in the Department of Mathematics, Sule Lamido University, Kafin Hausa, Jigawa, Nigeria. (e-mail:habibmth.slu@gmail.com)

are defined as

$$\beta_k^{FR} = \frac{\|\psi_k\|^2}{\|\psi_{k-1}\|^2},$$
$$\beta_k^{CD} = \frac{\|\psi_k\|^2}{-\psi_{k-1}^T d_{k-1}}, \quad (5)$$
$$\beta_k^{DY} = \frac{\|\psi_k\|^2}{d_{k-1}^T y_{k-1}}.$$

as proposed in [17], [18], [19].

Researchers often combine some existing CG parameters or directions to exploit their attractive properties and develop a practical algorithm. Some known discoveries on this type of CG method are the ones done by Wei, Li [16], and Yuan et al. [26]. Different researchers have proposed many hybrid methods see the survey paper conducted by (Hager and Zhang [23] and reference therein). These hybrids can be classified into two classes, hybrid FR and PRP-type methods such as the one proposed by Shi and Guo [22], where $\beta_k = \max\{-\beta_k^{FR}, \min\{\beta_k^{PRP}, \beta_k^{FR}\}\}$, and the DY and HS-type hybrid such as the one proposed by Gilbert and Nocedal [24], with $\beta_k = \max\{0, \min\{\beta_k^{HS}, \beta_k^{DY}\}\}$. For further information on the hybrid CG methods with reasonable discovery, refer to [23]. For a good discovery and excellent comparative study on new CG methods, refer to [29].

Abubakar et al. [32] modified and re-scaled the Fletcher-Reeves CG parameter to satisfy the descent conditions in every iteration. Likewise, Waziri et al. [33] changed the Dia Yuan CG parameter by following Xui et al. strategy in [34]; in conjunction with the projection algorithm of Solodov and Svaiter's [45], proposed a sufficient descent direction that restores a signal problem in compressed sensing. Abdullahi et al.[31] modified Dai Yuan CG parameter and presented three different sufficient-descent directions for the constrained system of equations. Ibrahim et al. in [35], combined FR and LS CG parameters for constrained systems with application in compressive sensing. The method's overall performance is promising.

Furthermore, In [44], Abubakar et al. Presented a hybrid of DY and FR conjugate gradient parameters and solved unconstrained optimization with application to portfolio selection. Although DY and FR parameters belong to the same class of CG parameters with good global convergence properties but weak or uncertain numerical performance, they achieved promising results.

Motivated by the achievement in [44], and the projection algorithm proposed in [45], this paper presents a new HS-DY hybrid CG parameter with good numerical performance and converges globally. We also proposed to extend the results to regularized ℓ_1 norm problem to deblurred some imaging problems in compressed sensing. The HS CG parameter is of the class of CG's with good convergence properties but uncertain numerical performance. On the other hand, the DY parameter is of the class of CG's with excellent numerical performance but uncertain global convergence. Therefore, a hybrid of these CG parameters, HS-DY, will give a better result.

The proposed method is iterative i.e., given any starting point x_k, and suitable step length $\alpha_k > 0$, it generates a new point x_{k+1} through

$$x_{k+1} = x_k + \alpha_k d_k, \quad k = 0, 1, 2, ..., \quad (6)$$

α_k the step length, is computed through the following line search condition presented in [43] given by

$$-\psi(x_k + \alpha_k d_k)^T d_k \geq \sigma\alpha_k \|\psi(x_k + \alpha_k d_k)\| . \|d_k\|^2 \quad (7)$$

The subsequent part is structured as follows. Next, we will present the method's algorithm. The convergence results are shown in section III; in section IV, we offer numerical analysis and application of the technique to image recovery. Finally, we provide the summary and conclusion in section V.

II. METHODOLOGY AND NUMERICAL ANALYSIS

This section introduces the projection operator and some assumptions before submitting the new HS-DY CG parameter and its numerical analysis. Let Ω be a convex close subset that is not empty of $\mathbf{R^n}$, then for any $x \in \mathbf{R^n}$, the projection on Ω is written as

$$P_\Omega[x] = \arg\ \min\{||x - y|| : y \in \Omega\}. \quad (8)$$

This map $P_\Omega : \mathbf{R^n} \to \mathbf{\Omega}$ refers to the projection operator. It has the following property

$$\|P_{\Omega[x]} - P_{\Omega[y]}\| \leq \|x - y\|, \quad \text{for} \quad \text{any} \quad x, y \in \mathbf{R^n}. \quad (9)$$

called non-expensive property. And hence

$$\|P_{\Omega[x]} - y\| \leq \|x - y\|, \quad \text{for} \quad \text{all} \quad y \in \Omega. \quad (10)$$

At this moment, we consider the two CG parameters $\beta_k^{HS} = \frac{\psi_k^T y_{k-1}}{s_{k-1}^T y_{k-1}}$ and $\beta_k^{DY} = \frac{\|\psi_k\|^2}{s_{k-1}^T y_{k-1}}$, for non-monotone functions, these two CG parameters become undefined at some point as their common denominator $s_{k-1}^T y_{k-1}$, may be zero. On the other hand, the former parameter is of the class of CG parameters with properties of excellent numerical performance but poor convergence properties. In contrast, the latter parameter is of the class of CG parameters with suitable convergence properties but uncertain numerical performance. For these reasons, we are motivated to hybridize these two parameters (β_k^{HS}, β_k^{DY}) to exploit their attractive properties and, come up with an effective CG parameter with good numerical performance and convergence property, which can be applied to various applications. We then form the two CG parameters as

$$\beta_k^{HSDY} = (1 - \mu_k)\beta_k^{HS} + \mu_k \beta_k^{DY}$$
$$= (1 - \mu_k)\frac{\psi_k^T y_{k-1}}{s_{k-1}^T y_{k-1}} + \mu_k \frac{\|\psi_k\|^2}{s_{k-1}^T y_{k-1}}, \quad (11)$$

where μ_k is a combination parameter to be addressed later.

Then consider the Dai-Liao [30] conjugacy condition given by the equation

$$d_{k-1}^T y_{k-1} = -t s_{k-1}^T \psi_k, \quad t > 0. \quad (12)$$

We take the inner product of (3) with vector y_{k-1} and consider β_k as (11), we have as follows

$$d_{k-1}^T y_{k-1} = -\psi_k^T y_{k-1} + (1-\mu_k)\frac{\psi_k^T y_{k-1}}{s_{k-1}^T y_{k-1}} s_{k-1}^T y_{k-1} + \mu_k \frac{\|\psi_k\|^2}{s_{k-1}^T y_{k-1}} s_{k-1}^T y_{k-1}. \tag{13}$$

By (12) and (13), we get

$$-t s_{k-1}^T \psi_k = -\psi_k^T y_{k-1} + (1-\mu_k)\frac{\psi_k^T y_{k-1}}{s_{k-1}^T y_{k-1}} s_{k-1}^T y_{k-1} + \mu_k \frac{\|\psi_k\|^2}{s_{k-1}^T y_{k-1}} s_{k-1}^T y_{k-1}. \tag{14}$$

After some algebraic simplification, we then have μ_k, to be

$$\mu_k = \frac{t s_{k-1}^T \psi_k}{\psi_k^T y_{k-1} - \|\psi_k\|^2}. \tag{15}$$

where

$$y_{k-1} = (\psi(z_k) - \psi(x_k)) \quad \text{and} \quad s_{k-1} = z_k - x_k.$$

Let

$$\bar{\mu}_k = \max\left\{\mu_k, \frac{\tau\|s_{k-1}\|^2}{y_{k-1}^T s_{k-1}}\right\}, \quad \tau > 0 \tag{16}$$

Now $0 \le \bar{\mu}_k \le 1$, if $\bar{\mu}_k > 1$ then choose $\bar{\mu}_k = 0.5$.

This parameter $\bar{\mu}_k$ is crucial for achieving the maximum efficiency of the hybridized parameters.
The sufficient condition for a good direction to converge globally is it's ability to satisfy the below inequality:

$$\psi_k^T d_k \le -h\|\psi_k\|^2, \quad h > 0. \tag{17}$$

To have a descent direction that stays in the trust region for every iteration, the proposed direction is rescaled to:
$d_0 = -\psi_0 \quad \text{if} \quad k = 0,$

$$d_k = -\eta_k \psi_k + \frac{(1-\bar{\mu}_k)\psi_k^T y_{k-1} + \bar{\mu}_k\|\psi_k\|^2}{\max\{||s_{k-1}||^2, \|s_{k-1}\|, \psi_k^T s_{k-1}\}} s_{k-1}, \quad k \ge 1, \tag{18}$$

where η_k is a positive parameter that can be determined so that the inequality (17) is satisfied.

Here we describe in detail the proposed method's algorithm.

Algorithm 2.1 A Modified Hybrid HS-DY Conjugate Gradient Parameter (MHCG)

STEP 0: Given starting points $x_0 \in \Omega$, constants $\sigma, \xi > 0$, $\rho \in (0,1)$, $\epsilon > 0$, set $d_0 = -\psi_0$ and $k = 0$.
STEP 1: If $\|\psi_k\| \le \epsilon$ stop. Else, go to step 2.
STEP 2: Perform an adaptive line search using a trust-region approach:

- Define the trust-region subproblem:

$$\min_{\alpha}\left\{\|\psi(x_k + \alpha_k d_k)\|^2\right\}$$

subject to:

$$-\psi(x_k + \alpha_k d_k)^T d_k \ge \sigma\alpha_k \|\psi(x_k + \alpha_k d_k)\|.\|d_k\|^2. \tag{19}$$

- where $\alpha_k = \xi\rho_k^m$ with m_k smallest positive integer such that α_k lies within the trust region.

STEP 3: Set $z_k = x_k + \alpha_k d_k$, if $z_k \in \varphi$ and $\|\psi(z_k)\| \le \epsilon$, stop. Else, step 4.

STEP 4: Compute x_{k+1} by

$$x_{k+1} = P_\Omega[x_k - \lambda_k \psi(z_k)] \tag{20}$$

where $\lambda_k = \frac{\psi(z_k)^T(x_k - z_k)}{\|\psi(z_k)\|^2}$
STEP 5: Determine d_{k+1} by (18)
STEP 6: Let $k = k+1$, and repeat the process.

III. Convergence Results

Before giving the proposed method's convergence, it is essential to provide some valuable assumptions.

Assumption $(A1)$. Let $\psi(x)$ be Lipschitz continuous. It implies the presence of $L > 0$ such that

$$\|\psi(z_k) - \psi(x_k)\| \le L\|z_k - x_k\| \quad \text{holds.} \tag{21}$$

Assumption $(A2)$. It is uniformly monotone, i.e, for all $x, y \in \mathbf{R^n}$ there exists positive c such that

$$(z_k - x_k)^T(\psi(z_k) - \psi(x_k)) \ge c\|z_k - x_k\|^2. \tag{22}$$

Remark 1. Assumption $A2$ implies that

$$y_{k-1}^T s_{k-1} = (\psi(z_k) - \psi(x_k))^T \ge c\|z_k - x_k\|^2. \tag{23}$$

Lemma 3.1. The proposed direction given by (18) satisfies (17) for $\eta_k \ge L + 1$

Proof: Since $0 \le \bar{\mu}_k \le 1$, taking the inner product of (18) with ψ_k we have,

$$\psi_k^T d_k = -\eta_k\|\psi_k\|^2 + \frac{(1-\bar{\mu}_k)||\psi_k||^2(s_{k-1}^T y_{k-1}) + \bar{\mu}_k\|\psi_k\|^2(\psi_k^T s_{k-1})}{\max\{||s_{k-1}||^2, \|s_{k-1}\|, \psi_k^T s_{k-1}\}}$$

$$\le -\eta_k\|\psi_k\|^2 + \frac{||\psi_k||^2(s_{k-1}^T y_{k-1}) + ||\psi_k||^2(\psi_k^T s_{k-1})}{\max\{||s_{k-1}||^2, \|s_{k-1}\|, \psi_k^T s_{k-1}\}}$$

$$\le -\eta_k\|\psi_k\|^2 + \frac{||\psi_k||^2(s_{k-1}^T y_{k-1})}{\max\{||s_{k-1}||^2, \|s_{k-1}\|, \psi_k^T s_{k-1}\}} + \frac{||\psi_k||^2(\psi_k^T s_{k-1})}{\max\{||s_{k-1}||^2, \|s_{k-1}\|, \psi_k^T s_{k-1}\}}$$

$$\le -\eta_k\|\psi_k\|^2 + \frac{L||\psi_k||^2||s_{k-1}||^2}{\max\{||s_{k-1}||^2, \|s_{k-1}\|, \psi_k^T s_{k-1}\}} + \frac{||\psi_k||^2(\psi_k^T s_{k-1})}{\max\{||s_{k-1}||^2, \|s_{k-1}\|, \psi_k^T s_{k-1}\}}$$

$$\le -\eta_k\|\psi_k\|^2 + \frac{L||\psi_k||^2||s_{k-1}||^2}{||s_{k-1}||^2} + \frac{||\psi_k||^2(\psi_k^T s_{k-1})}{(\psi_k^T s_{k-1})}$$

$$= -\eta_k\|\psi_k\|^2 + L||\psi_k||^2 + ||\psi_k||^2$$

$$= -(\eta_k - L - 1)||\psi_k||^2.$$

This is if $\eta_k \ge L + 1$
Now take

$$\eta_k = 2 + L \tag{24}$$

The lemma below indicates that when the solution is not reached, there exists α_k satisfying the line search condition given by (19).

Lemma 3.2. suppose the assumptions are fulfilled, then there exists α_k satisfying (19) $\forall k \geq 0$

Proof By contradiction, let's assume $k_0 \geq 0$ for any non negative i (19) is not true at k_0^{th} iteration i.e.,

$$-\psi(x_{k_0} + \alpha_{k_0} d_{k_0})^T d_{k_0} < \sigma\xi\rho^i \|d_{k_0}\|^2. \tag{25}$$

ψ is continuous and $\rho \in (0, 1,)$, $i \to \infty$, then

$$-\psi(x_{k_0})^T d_{k_0} \leq 0. \tag{26}$$

But from (17), we get

$$-\psi(x_{k_0})^T d_{k_0} \geq \|\psi(x_{k_0})\|^2 > 0. \tag{27}$$

This contradicts (26).

This lemma is the same as theorem 2.1 of [45]. Therefore, its proof is left.

Lemma 3.3. Suppose all mentioned Assumptions hold. Let $\{x_k\}$ be from the proposed Algorithm, for $\bar{x}$, $\psi(\bar{x}) = 0$,

$$\|x_{k+1} - \bar{x}\|^2 \leq ||x_k - \bar{x}\|^2 - \|x_{k+1} - x_k\|^2. \tag{28}$$

This implies $\{x_k\}$ is bounded.

Moreover, it shows that $\{x_k\}$ is finite and x^* is a solution of (1), or infinite and

$$\sum_{k=0}^{\infty} \|x_{k+1} - x_k\|^2 < \infty, \tag{29}$$

this implies

$$\lim_{k\to\infty} \|x_{k+1} - x_k\| = 0. \tag{30}$$

Remark 2. From the above, it is obvious that

$$\|x_k - \bar{x}\| \leq \|x_0 - \bar{x}\|.$$

Also from assumption **A1**, it follows that

$$\|\psi(x_k)\| = \|\psi(x_k) - \psi(\bar{x})\| \leq L\|x_k - \bar{x}|| \leq L\|x_0 - \bar{x}\|.$$

Therefore, taking $\kappa = L||x_0 - \bar{x}||$, we have

$$\|\psi(x_k)\| \leq \kappa. \tag{31}$$

Lemma 3.4. Assume all the assumptions are fulfilled, and $\{x_k\}$ and $\{z_k\}$ come from MHCG, then they are bounded, and

$$\lim_{k\to\infty} \|x_k - z_k\| = 0. \tag{32}$$

Moreover,

$$\lim_{k\to\infty} \alpha_{k-1}\|d_k\| = 0. \tag{33}$$

Proof. Let $\bar{x}$ satisfy (1) and ψ monotone, then

$$\psi(z_k)^T(x_k - \bar{x}) \geq \psi(z_k)^T(x_k - z_k),$$

by z_k and using (19) we arrived at

$$\begin{aligned}
\|x_{k+1} - \bar{x}\|^2 &= \|P_\Omega[x_k - \lambda_k\psi(z_k)] - \bar{x}\|^2 \\
&\leq \|x_k - \lambda_k\psi(z_k) - \bar{x}\|^2 \\
&= \|x_k - \bar{x}\|^2 - 2\lambda_k\psi(z_k)^T(x_k - \bar{x}) + \\
&\quad \|\lambda_k\psi(z_k)\|^2 \\
&\leq \|x_k - \bar{x}\|^2 - 2\lambda_k\psi(z_k)^T(x_k - z_k) + \\
&\quad \|\lambda_k\psi(z_k)\|^2 \\
&= \|x_k - \bar{x}\|^2 - \frac{\left(\psi(z_k)^T(x_k - z_k)\right)^2}{\|\psi(z_k)\|^2}.
\end{aligned} \tag{34}$$

This tells us that the sequence $\{\|x_{k+1} - \bar{x}\|\}$ is non-increasing sequence and hence converged. Thus $\{x_k\}$ is bounded. From z_k, (19), monotonicity of ψ, Cauchy-Schwarz inequality, we have

$$\begin{aligned}
&\sigma\|\psi(z_k)\|\|x_k - z_k\| = \frac{\sigma\|\psi(z_k)\|\alpha_k d_k\|^2}{\|x_k - z_k\|} \leq \\
&\frac{\psi(z_k)^T(x_k - z_k)}{\|x_k - z_k\|} \leq \frac{\psi(x_k)^T(x_k - z_k)}{\|x_k - z_k\|} \leq \|\psi(z_k)\|.
\end{aligned} \tag{35}$$

From relation (33), it follows that

$$\begin{aligned}
\frac{\sigma^2}{k^2}\sum_{k=0}^{\infty}\|x_k - z_k\|^4 &\leq \sum_{k=0}^{\infty}\frac{\psi(z_k)^T(x_k - z_{k^2})}{\|\psi(z_k)\|^2} \\
&\leq \sum_{k=0}^{\infty}(\|x_k - \bar{x}\|^2 - \|x_{k+1} - \bar{x}\|^2) < \infty.
\end{aligned} \tag{36}$$

By convergent series' properties, (31) holds. Then from z_k and (32), we get (33).

Lemma 3.5. Let all assumptions be satisfied and d_k is a sequence generated by HS-DY algorithm, then there exists $N > 0$ such that $||d_k|| \leq N$

Proof.

Then by (18) utilizing (21) and remark **A1**, we get choosing

$$\begin{aligned}
\|d_k\| &= \left\|-\eta_k\psi_k + \frac{(1-\bar{\mu}_k)\psi_k(y_{k-1}^T s_{k-1}) + \bar{\mu}_k\|\psi_k\|^2 s_{k-1}}{\max\{||s_{k-1}||^2, \|s_{k-1}\|, \psi_k^T s_{k-1}\}}\right\| \\
&\leq |\eta_k|\|\psi_k\| + \left\|\frac{(1-\bar{\mu}_k)\psi_k(y_{k-1}^T s_{k-1})}{\max\{||s_{k-1}||^2, \|s_{k-1}\|, \psi_k^T s_{k-1}\}}\right\| + \\
&\quad \left\|\frac{\bar{\mu}_k||\psi_k||^2 s_{k-1}}{\max\{||s_{k-1}||^2, \|s_{k-1}\|, \psi_k^T s_{k-1}\}}\right\| \\
&\leq |\eta_k|\|\psi_k\| + \left\|\frac{\psi_k(y_{k-1}^T s_{k-1})}{\max\{||s_{k-1}||^2, \|s_{k-1}\|, \psi_k^T s_{k-1}\}}\right\| + \\
&\quad \left\|\frac{-\bar{\mu}_k\psi_k(y_{k-1}^T s_{k-1})}{\max\{||s_{k-1}||^2, \|s_{k-1}\|, \psi_k^T s_{k-1}\}}\right\| + \\
&\quad \left\|\frac{\bar{\mu}_k||\psi_k||^2 s_{k-1}}{\max\{||s_{k-1}||^2, \|s_{k-1}\|, \psi_k^T s_{k-1}\}}\right\| \\
&\leq |\eta_k|\|\psi_k\| + \frac{L\|s_{k-1}\|^2\|\psi_k\|}{\max\{||s_{k-1}||^2, \|s_{k-1}\|, \psi_k^T s_{k-1}\}} +
\end{aligned}$$

$$\frac{L|\bar{\mu}_k|\|s_{k-1}\|^2\|\psi_k\|}{\max\{||s_{k-1}||^2, \|s_{k-1}\|, \psi_k^T s_{k-1}\}} +$$

$$\frac{|\bar{\mu}_k|\|\psi_k\|^2\|s_{k-1}\|}{\max\{||s_{k-1}||^2, \|s_{k-1}\|, \psi_k^T s_{k-1}\}}$$

$$\leq |\eta_k|\|\psi_k\| + \frac{L\|s_{k-1}\|^2\|\psi_k\|}{\|s_{k-1}\|^2} + \frac{L|\bar{\mu}_k|\|s_{k-1}\|^2\|\psi_k\|}{\|s_{k-1}\|^2} +$$

$$\frac{|\bar{\mu}_k|\|\psi_k\|^2\|s_{k-1}\|}{\|s_{k-1}\|}$$

$$= |\eta_k|\|\psi_k\| + L\|\psi_k\| + L|\bar{\mu}_k|\|\psi_k\| + |\bar{\mu}_k|\|\psi_k\|^2$$

$$\leq (2+L)\kappa + L\kappa + LH\kappa + H\kappa^2$$

$$||d_{k+1}|| \leq N$$

where

$$N = (2+L)\kappa + L\kappa + LH\kappa + H\kappa^2$$

The following theorem summarized the global convergence MHCG.

Theorem 1. *Suppose $\{x_k\}$ is generated by MHCG and all assumptions hold, then*

$$\liminf_{k\to\infty} \|\psi_k\| = 0. \tag{37}$$

Proof. Going by contradiction, assume (37) does not hold, it implies $\epsilon_0 > 0$ and $||\psi_k|| \geq \epsilon_0$ is true $\forall k$. But by the inequality (17), it implies that

$$c\|\psi_k\|^2 \leq -\psi_k^T d_k \leq \|\psi_k\|\|d_k\| \quad \forall \quad k, \tag{38}$$

and thus

$$\|d_k\| \geq c\|\psi_k\| \geq c\epsilon_0 \quad \forall \quad k. \tag{39}$$

Then by inequality (33) and (39), it follows that

$$\lim_{k\to\infty} \alpha_{k-1} = 0. \tag{40}$$

Then if $k \to \infty$, $\alpha_k^{'} = \rho^{-1}\alpha_k$ not imply (19) and hence

$$-\psi(x_k + \alpha_k^{'} d_k)^T d_k < \sigma\alpha_k^{'}\|\psi(x_k + \alpha_k^{'} d_k)\|.\|d_k\|^2. \tag{41}$$

Then by Lemma **3.1** and **3.4** we have that sequences $\{x_k\}$ and $\{d_k\}$ are bounded, which means that there is $\bar{x}$ and $\bar{d}$ (accumulation points) and indexes $\mathrm{K}_2 \subset \mathrm{K}_1$ (infinite) such that the $\lim_{k\to\infty} x_k = \bar{x}$ and $\lim_{k\to\infty} d_k = \bar{d}$ for $\mathrm{K} \in \mathrm{K}_2 \subset \mathrm{K}_1$. For $k \to \infty$, and applying limit at both sides of (42) for all $\mathrm{K} \in \mathrm{K}_2$, we get

$$\psi(\bar{x})^T \bar{d} > 0. \tag{42}$$

But taking limit as $k \to \infty$ of both sides of inequality (19) for $\mathrm{K} \in \mathbf{K}_2$, we get

$$\psi(\bar{x})^T \bar{d} \leq 0. \tag{43}$$

This goes in contrast with (42) and hence the proof.

IV. Numerical Outcomes and Results

The last part of the paper is divided into two segments; first and second. In the first segment, some numerical comparison of the proposed algorithm with two similar algorithms available in the literature, i.e., **CHCG** [36] and **DTCG1** [31] are provided. In the second segment, the proposed algorithm (MHCG) is extended and co-run together with SGCS [38] algorithm to ℓ_1 norm regularized problem in compressed sensing where some blurred images are deblurred. The codes were written in the MATLAB R2018a and ran on a computer (2.80GHZ CPU, 16GB RAM). During the implementation process, we used the same line-search procedure and set the following parameters for all of the codes: $\xi = 1$, $\sigma = 10^{-6}$, $\rho = 0.9$ and $t = 0.9$. Any method is set to stop if one of the following occurs :

- The number of iteration ≥ 1000 but no x_k satisfying the stoping criteria is obtained.
- $\|\psi(x_k)\| \leq 10^{-10}$.
- $\|\psi(z_k)\| \leq 10^{-10}$.

The three algorithms were tested with the following test functions adopted from Abdullahi et al. [31].

Problem 1: $\psi : \Omega \to \mathbb{R}^n$ with
$\psi_i(x) = \ln(|x_i|+1) - \frac{x_i}{n}$ for $i = 1, 2, 3, ..., n$, and Ω=$\mathbb{R}^n_+$.

Problem 2: $\psi : \Omega \to \mathbb{R}^n$ with
$\psi_i(x) = x_i - \sin(|x_i + 1|)$ for $i = 1, 2, 3, ..., n$, and $\Omega = \mathbb{R}^n_+$.

Problem 3: $F : \Omega \to \mathbb{R}^n$ with
$\psi_1(x) = 3x_1^3 + 2x_2 - 5 + \sin(x_1 - x_2) + \sin(x_1 + x_2)$,
$\psi_i(x) = 3x_i^3 + 2x_{i+1} - 5 + \sin(x_i - x_{i+1}) + \sin(x_i + x_{i+1}) + 4x_i - x_{i-1}\exp^{x_{i-1}-x_i} - 3$,
$\psi_n(x) = x_{n-1} + \exp^{x_{n-1}-x_i} - 4x_n - 3$,
for $i = 2, 3, 4, ..., n-1$ and $\Omega = \mathbb{R}^n_+$.

Problem 4: $\psi : \Omega \to \mathbb{R}^n$ with
$\psi_i(x) = \min\{\min(x_i, x_i^2), \max(|x_i|, x_i^3)\}$ *for*
$i = 2, 3, 4, ..., n$, and $\Omega = \mathbb{R}^n_+$.

Problem 5: $\psi : \Omega \to \mathbb{R}^n$ with
$\psi_1(x) = \frac{5}{2}x_1 + x_2 - 1$,
$\psi_i(x) = x_{i-1} + \frac{5}{2}x_{i+1} - 1$,
$\psi_n(x) = x_{n-1} + \frac{5}{2}x_{n-1}$, for $i = 2, 3, 4, ..., n.$ and $\Omega = \mathbb{R}^n_+$.

Problem 6: Tridiagonal Exponential problem $\psi : \Omega \to \mathbb{R}^n$ with
$\psi_1(x) = x_1 - \exp^{\cos(h(x_1+x_2))}$,
$\psi_i(x) = x_1 - \exp^{\cos(h(x_{i-1}+x_i+x_{i+1}))}$,
$\psi_n(x) = x_n - \exp^{\cos(h(x_{n-1}+x_n))}$,
where $\frac{1}{n+1}$, for $i = 2, 3, 4, ..., n-1.$ and $\Omega = \mathbb{R}^n_+$.

The three algorithms were run using a wide range of dimensions to see how robust the proposed method is. The range of the dimensions used is 1000–100,000. For the initial starting points, we used $X_1 = (1, 1, ..., 1)^T$, $X_2 = (\frac{1}{2}, \frac{1}{2^2}, ..., \frac{1}{2^n})^T$, $X_3 = (1, \frac{1}{4}, \frac{1}{9}, ..., \frac{1}{n^2})^T$, $X_4 = (2, 2, ..., 2)^T$, $X_5 = (0, \frac{1}{2}, \frac{2}{3}..., 1 - \frac{1}{n})^T$ and $X_6 = (-\frac{1}{4}, -\frac{1}{4}, ..., \frac{(-1)^2}{4})^T$.

These starting points are evenly and varied enough to test the algorithm's robustness.

The six tables at the end displayed the numerical results of the proposed algorithm (MHCG) in competition with two similar algorithms CHCG and DTCG1.

We report the numerical performance of the three methods in Table I to Table VI. In the tables, "DIM," "INT PNT," "NI," "NF," "TIME," and "NORM" refer to the dimensions, initial points, number of iterations, number of function evaluations, CPU time, and value of the residual at the stopping point respectively. The tables show that all the algorithms, MHCG, SIGNAL, and DTCG1, solved all the test functions. However, the robustness of the proposed method is apparent. Because of the Proposed method (MHCG), performance on the set of problems for the number of iterations, number of function evaluations, and time taken for the execution (CPU time) is better than the performance of the metrics for the compared algorithms.

These indicate that the proposed algorithm is a fast and robust alternative for practical purposes where large-scale systems are present. This achievement could result from hybridizing the two cg parameters with desirable properties and a unique hybridizing parameter that updates through Dai and Liao conjugacy conditions.

Furthermore, using Dolan and Moré [39] performance profile, three graphs are plotted to assess and compare the proposed algorithm's performance with the two algorithms on the set of chosen test's functions. For the function P_n, and solver S_n, $n = 1, 2, ...$ define,

$t_{P,S} =$ time required to solve test function by algorithm S.

This metric is the same as other metrics, i.e., function evaluation, number of iterations, etc. A performance ratio was used to devise a baseline for comparing a test function by a solver.

The ratio is as

$$r_{P,S} = \frac{t_{P,S}}{min\{t_{P,S} : S_n\}}.$$

Then to get the overall evaluation of the performance of the solver, we have,

$$\rho_s(\tau) = \frac{1}{P_n} size\{p_n : r_{P,S} \leq \tau\}$$

where $\rho_s(\tau)$ is the cumulative distribution function for test function S_n with performance ratio within a factor $\tau \in \mathbb{R}$.

The computational analysis, summarized in Tables I to VI, was conducted using MATLAB. The results were visualized in Figures I to III. Figure I illustrates the algorithm's performance based on the number of iterations, Figure II shows the results based on CPU time, and Figure III presents the performance in terms of the number of function evaluations. After plotting the graphs, the method whose performance profile is on top of the other is considered the best method, and it is visible from the graphs that MHCG is the winner.

V. Applications to ℓ_1 Regularized Problem

In this part, we extended the proposed algorithm (MHCG) and applied it together with SGCS algorithm [38] to ℓ_1 norm regularization to correct some imaging problems in compressed sensing.

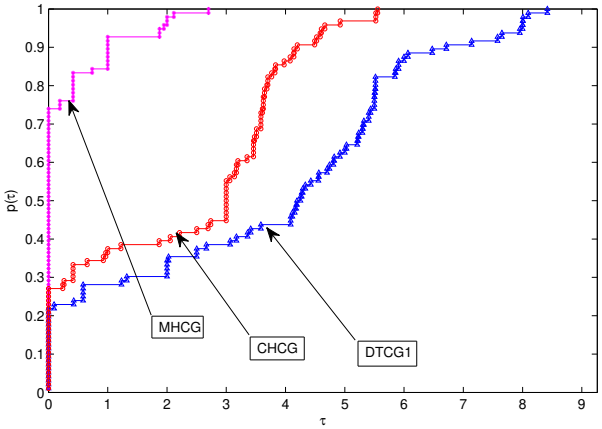

Fig. I: Performance profile of MHCG, SIGNAL, and DTCG1 algorithms for the number of iterations (NI).

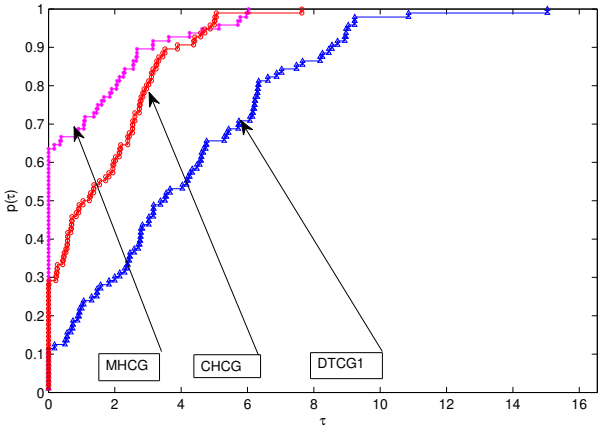

Fig. II: Performance profile of MHCG, SIGNAL, and DTCG1 algorithms for the CPU Time (TIME).

Apart from its already known applications, recently, it has been extended to compressive sensing, where images and signal problems are overcoming [36]. Compressed sensing (CS) mainly aims to approximate and compress the original signal using the sparsity property [46] from the least number of incoherent linear measurements.

The most popular approach involves optimizing the objective function with norm regularization i.e.,

$$\min_x \frac{1}{2}\|y - Bx\|_2^2 + \tau\|x\|_1, \tag{44}$$

with $x \in \mathbf{R^n}$, $y \in \mathbf{R^k}$, $B \in \mathbf{R^{kxn}}$ $(k < n)$ is operator, τ is a positive parameter, $\|x\|_p = \left(\sum_{i=1}^{n} |x_i|^p\right)^{\frac{1}{p}}$ for $p = 1, 2$ represents the ℓ_1 and ℓ_2 of x respectively. (44) stands for a convex problem that is frequently encountered in compressive sensing problems.

There are many optimization techniques for dealing with the problem of this form (44) some of them can be viewed from the following references for [2], [1], [36].

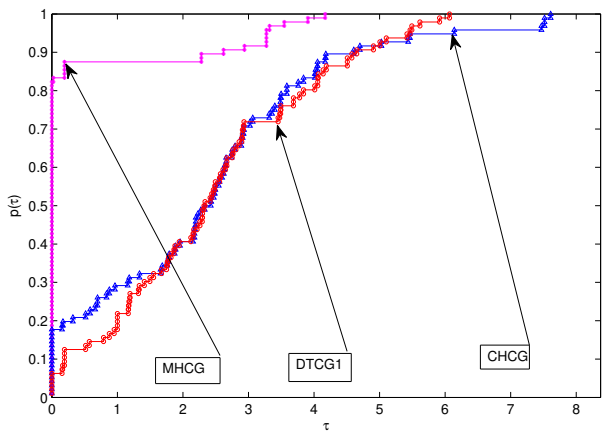

Fig. III: Performance profile of MHCG, SIGNAL, and DTCG1 algorithms for the number of function evaluations (NF).

The gradient-based method is the most widely used, thus why conjugate gradient-based methods are also essentially applicable Figueiredo et al. [1]. For a gradient (GPSR) Problem of the form (44) is splitting into positive and negative part vector,

$$i.e., x = v - u, \quad u, v \geq 0, \quad u, v \in \mathbf{R^n}. \tag{45}$$

Let $u_i = (x_i)_+$, and $v_i = (-x_i)_+$ for $i = 1, 2, 3, ..., n$, where $(.)_+$ is the positive part operator, defined by $(x)_+ = \max\{0, x\}$. By ℓ_1-norm, we have $\|x\|_1 = e_n^T u + e_n^T v$, with $e_n = (1, 1, 1, ..., 1)^T \in \mathbf{R^n}$. Therefore, Problem (44) can be reformulated as

$$\min_{u,v} \frac{1}{2}\|y - B(u - v)\|_2^2 + \tau e_n^T u + \tau e_n^T v, \quad u, v \geq 0. \tag{46}$$

As shown in [1], problem (46) can be expressed in a more conventional problem as follows

$$\min_z \frac{1}{2} z^T H z + c^T z, \quad \text{s.t} \quad z \geq 0, \tag{47}$$

where $z = \begin{pmatrix} u \\ v \end{pmatrix}$, $c = \tau e_{2n} + \begin{pmatrix} -b \\ b \end{pmatrix}$, $b = B^T y$,

$$H = \begin{pmatrix} B^T B & -B^T B \\ -B^T B & B^T B \end{pmatrix}.$$

H is a nonnegative matrix. Hence, (47) is a convex problem. Xiao et al. in [41], stated that the above problem can be translated to the following

$$F(z) = \min\{z, Hz + c\} = 0. \tag{48}$$

It was shown by Xiao et al. [41] that F in (48) is Lipschitz continuous and monotone; as a result, problem (44) can be transformed into a problem of the form (1) and eventually solved by the proposed algorithm (MHCG). The parameters used in the SGCS algorithm are also maintained here.

We set the algorithms to stop when the following condition is satisfied.

$$\frac{|f_k - f_{k-1}|}{|f_{k-1}|} < 10^{-5}, \tag{49}$$

where $f(x)$ is a merit function defined as $f(x) = \frac{1}{2}\|y - B\|_2^2 + \tau\|x\|_1$. During the process of the experiment, the two codes use $x_0 = y^T B$ as a starting point, and signal-to-noise ratio (SNR), is defined by

$$SNR = 20X \log_{10}\left(\frac{\|\hat{x}\|}{\|x - \hat{x}\|}\right).$$

here, $\hat{x}$ and x are the restored and original image respectively.

Moreover, we use a notion of Structural Similarity (SSIM) index in measuring the quality of the restoration process [47]. Figure IV displays the set of images of Lena, House, Barbara, and pepper. Each pack contains four photos from the original image, the blurred and restored images by the MHCG and SGCS algorithms.

Table VII summarizes the restoration results of the MHCG and SGCS algorithms. From the first column to the last is the list of the images, the size of the photos, the number of iterations, the time taken in the restoration process, the signal-to-noise ratio (SNR), and lastly, the structural similarity (SSIM) index; which measure the quality of the restoration process.

As seen from the table and pictures, both algorithms accurately restored the images. Even though they both have an equal number of structural similarity (SSIM) indexes, the number of iterations and time for restoration of the proposed method is less than that of the SGCS method, which clearly shows that the proposed method won the race.

VI. Conclusion and Future Scope

In this study, we introduce a hybrid HS-DY CG parameter (MHCG) tailored for constrained monotone nonlinear equations, specifically focusing on its application in image recovery tasks. By blending two classical CG methods, we aim to enhance the efficiency and effectiveness of the optimization process. The hybridization parameter, denoted as μ_k, undergoes iterative updates leveraging the Dai and Liao conjugacy condition, with the Dai and Liao positive parameter t being set to a fixed value of 1.Notably, our proposed method exhibits descent properties and operates in a matrix-free manner. To gauge its performance, we conducted a comparative analysis with two existing methods, namely CHCG and DTCG1, sourced from the literature. The results, as presented in Tables I through VI and Figures I through III, demonstrate the superiority of MHCG across various metrics, including the number of iterations, execution time, and function evaluations. Graphical representations further corroborate these findings, with the performance curves of MHCG consistently surpassing those of CHCG and DTCG1.

Furthermore, we extend the applicability of MHCG by integrating it with the SGCS method in compressive sensing problems. Through experiments, we successfully restored blurred images with high structural similarity indexes to their original states. Importantly, MHCG exhibits fewer iterations and faster restoration times compared to the SGCS algorithm, underscoring its effectiveness and promise in practical applications.

Looking ahead, future work should explore iterative updates of the parameter to optimize its value, thereby enhancing the overall efficiency of the algorithm.

TABLE I: Numerical Comparison of MHCG, CHCG, and DTCG1 Algorithms

		MHCG				CHCG				DTCG1			
Dim	INT PNT	NI	NF	TIME	NORM	NI	NF	TIME	NORM	NI	NF	TIME	NORM
1000	x1	2	15	0.015000	0	4	40	0.065000	0	6	60	0.400000	1.00E-08
	x2	3	30	0.020000	0	25	250	0.100000	8.00E-07	50	100	0.700000	8.00E-07
	x3	3	32	0.018000	0	25	230	0.110000	7.50E-07	60	120	0.070000	9.00E-07
	x4	2	20	0.014000	0	200	170	3.000000	0.180000	6	10	0.620000	1.00E-08
	x5	3	25	0.012000	0	5	5	0.060000	0	80	160	0.120000	3.70E-07
	x6	2	2	0.001500	0	6	5	0.710000	0	5	5	0.008000	0
10,000	x1	2	15	0.035000	0	3	30	0.620000	0	6	60	0.850000	3.00E-09
	x2	3	30	0.045000	0	23	230	0.600000	9.00E-07	65	130	0.500000	3.50E-07
	x3	3	30	0.050000	0	22	220	0.630000	7.50E-07	70	140	0.550000	2.00E-07
	x4	5	35	0.075000	0	100	130	20.000000	0.100000	5	10	0.660000	1.00E-09
	x5	2	22	0.085000	0	3	5	0.650000	0	80	150	0.680000	9.00E-07
	x6	1	2	0.005000	0	6	5	0.730000	0	8	5	0.018000	0
50,000	x1	2	15	0.150000	0	3	35	0.880000	0	5	60	0.160000	1.00E-09
	x2	3	30	0.200000	0	20	230	2.100000	8.00E-07	55	105	1.450000	8.50E-07
	x3	3	30	0.225000	0	18	220	2.500000	7.50E-07	45	90	1.400000	3.00E-07
	x4	5	35	0.375000	0	110	130	95.000000	0.300000	6	10	0.190000	4.00E-10
	x5	1	15	0.375000	0	3	5	0.780000	0	55	105	1.850000	6.00E-07
	x6	1	2	0.022500	0	8	5	0.870000	0	6	5	0.075000	0
100,000	x1	2	15	0.300000	0	3	30	0.150000	0	5	60	0.300000	9.00E-10
	x2	3	30	0.350000	0	22	230	4.000000	8.80E-07	60	120	3.200000	7.00E-07
	x3	3	30	0.400000	0	20	220	4.600000	7.50E-07	60	120	3.900000	9.00E-07
	x4	5	35	0.750000	0	100	85	185.000000	0.260000	6	10	0.360000	2.00E-10
	x5	3	32	0.750000	0	2	5	0.160000	0	70	140	4.800000	9.00E-07
	x6	2	2	0.055000	0	5	5	0.120000	0	8	5	0.125000	0

TABLE II: Numerical Comparison of MHCG, CHCG, and DTCG1 Algorithms

		MHCG				CHCG				DTCG1			
Dim	INT PNT	NI	NF	TIME	NORM	NI	NF	TIME	NORM	NI	NF	TIME	NORM
1000	x1	2	20	0.015573	0	2	3	0.068972	0	4	6	0.459563	3.6E-08
	x2	3	40	0.024903	5.0E-21	22	228	0.109588	8.01E-07	45	88	0.747916	8.25E-07
	x3	4	50	0.020084	0	20	226	0.122738	7.56E-07	55	108	0.068962	9.45E-07
	x4	2	50	0.016419	0	191	167	2.986658	0.183875	5	7	0.610444	1.74E-08
	x5	3	30	0.018419	5.0E-20	2	3	0.056799	0	91	181	0.114951	3.72E-07
	x6	3	2	0.002505	0	5	4	0.716757	0	4	4	0.007852	0
10,000	x1	2	20	0.055756	0	2	3	0.621393	0	4	6	0.842872	3.62E-09
	x2	3	40	0.070520	5.0E-22	22	229	0.590066	8.91E-07	65	126	0.488548	3.66E-07
	x3	3	40	0.085839	0	20	226	0.633979	7.51E-07	68	141	0.554158	2.02E-07
	x4	8	55	0.139624	0	98	123	19.89326	0.097595	5	7	0.653262	1.24E-09
	x5	2	35	0.140624	0	2	3	0.645326	0	76	147	0.669820	9.52E-07
	x6	1	2	0.010809	0	5	4	0.725768	0	8	4	0.017394	0
50,000	x1	2	20	0.300221	0	2	3	0.875330	0	4	6	0.159350	1.34E-09
	x2	3	40	0.380055	1.0E-22	22	229	2.099865	8.84E-07	51	100	1.452431	8.67E-07
	x3	3	40	0.450724	0	20	226	2.496463	7.51E-07	44	89	1.406452	3.04E-07
	x4	8	55	0.709408	0	100	1281	93.09833	0.290251	5	7	0.186010	4.01E-10
	x5	1	25	0.709408	0	2	3	0.777077	0	53	103	1.848108	6.08E-07
	x6	1	2	0.038838	0	9	4	0.868823	0	5	4	0.072983	0
100,000	x1	2	20	0.600882	0	2	3	0.144905	0	4	6	0.297727	9.27E-10
	x2	3	40	0.700231	5.0E-23	22	229	3.975930	8.83E-07	57	114	3.212242	7.32E-07
	x3	3	40	0.830613	0	20	226	4.567206	7.51E-07	57	113	3.860539	9.4E-07
	x4	8	55	1.400266	0	100	82	183.7413	0.252120	5	7	0.350128	2.71E-10
	x5	4	45	1.400266	0	2	3	0.150685	0	73	135	4.810615	9.47E-07
	x6	2	2	0.095967	0	5	4	0.120206	0	7	4	0.123437	0

TABLE III: Numerical Comparison of MHCG, SIGNAL, and DTCG1 Algorithms

		MHCG				SIGNAL				DTCG1			
Dim	INT PNT	NI	NF	TIME	NORM	NI	NF	TIME	NORM	NI	NF	TIME	NORM
1000	x1	2	8	0.045945	0.000150	134	141	0.140026	0	7	22	0.013738	1.6E-07
	x2	2	20	0.009119	0	191	195	0.194481	0	34	65	0.092278	1.6E-07
	x3	3	18	0.008855	0	58	65	0.080862	0	23	54	0.342401	1.6E-07
	x4	2	6	0.039715	0.001098	136	137	0.136742	0	8	34	0.092278	1.6E-07
	x5	3	25	0.010771	0	107	114	0.136141	0	56	32	0.342401	1.6E-07
	x6	2	15	0.029237	0.001395	127	128	0.111113	0	8	23	0.012434	1.6E-07
10,000	x1	3	20	0.100262	0.004795	128	129	0.773108	0	7	22	0.895087	5.06E-07
	x2	2	15	0.010783	0	141	145	0.944654	0	77	48	0.123213	0
	x3	2	18	0.012564	0	60	67	0.622812	0	88	23	0.123213	1.6E-07
	x4	2	7	0.256163	0.004805	118	119	8.440762	0	8	34	0.888342	5.06E-07
	x5	3	22	0.008593	0	213	214	2.701107	0	4	39	0.100919	0
	x6	2	7	0.025615	0.004213	120	121	1.079774	0	8	23	0.792278	5.06E-07
50,000	x1	3	12	1.47212	0.011091	120	121	3.554603	0	8	23	0.342401	0
	x2	3	18	0.018475	0	257	261	6.799504	0	55	56	0.092278	1.6E-07
	x3	2	20	0.015623	0	89	96	2.562572	0	66	78	0.342401	1.6E-07
	x4	3	7	0.015847	0.010925	118	119	3.198937	0	9	35	0.428215	0
	x5	3	22	0.032515	0	315	316	8.31724	0	4	39	0.395074	0
	x6	3	8	0.067749	0.010612	111	112	2.826385	0	10	24	0.409435	0
100,000	x1	3	8	0.021406	0.015241	115	116	5.688736	0	8	23	0.632315	0
	x2	3	18	0.037726	0	342	346	17.03736	0	77	34	0.123213	1.6E-07
	x3	3	20	0.035410	0	201	208	9.653881	0	88	23	0.123213	1.6E-07
	x4	3	7	0.041693	0.015007	127	128	6.178666	0	10	35	0.77823	0
	x5	2	30	0.011165	0	838	839	37.45256	0	4	39	0.712328	0
	x6	3	8	0.018648	0.014153	112	113	5.585985	0	9	24	0.685135	0

TABLE IV: Numerical Comparison of MHCG, CHCG, and DTCG1 Algorithms

		MHCG				CHCG				DTCG1			
Dim	INT PNT	NI	NF	TIME	NORM	NI	NF	TIME	NORM	NI	NF	TIME	NORM
1000	x1	5	15	0.025743	1.85E-07	199	188	1.290244	4.671462	24	53	0.064743	4.85E-07
	x2	5	20	0.021778	2.54E-07	188	189	1.315886	7.388845	19	42	0.051778	6.54E-07
	x3	6	22	0.028019	3.1E-07	198	201	1.294982	7.387425	26	59	0.079019	8.1E-07
	x4	5	20	0.029331	2.81E-07	301	202	1.154269	1.952326	27	61	0.072331	5.81E-07
	x5	6	23	0.031613	3.15E-07	56	205	1.408702	4.693267	28	66	0.073613	7.35E-07
	x6	6	25	0.034273	3.2E-07	56	209	1.270771	8.071602	26	56	0.080273	6.95E-07
10,000	x1	7	12	0.045072	1.35E-07	56	209	8.227707	1.71722	8	19	0.135072	3.35E-07
	x2	6	14	0.042232	1.45E-07	88	210	8.490477	2.716476	8	19	0.119232	3.5E-07
	x3	7	15	0.048197	1.35E-07	89	102	8.456101	2.716426	8	19	0.136197	3.34E-07
	x4	6	16	0.053410	2.4E-07	91	103	8.965257	6.596583	7	17	0.13341	7.34E-07
	x5	6	15	0.049509	1.65E-07	92	104	8.033488	1.718246	8	19	0.152509	6.02E-07
	x6	7	16	0.045501	1.55E-07	100	105	9.117542	2.966404	8	19	0.128501	3.83E-07
50,000	x1	7	14	0.188128	2.24E-07	100	106	37.75063	3.8421	8	19	0.558128	3.84E-07
	x2	7	15	0.19843	2.38E-07	103	109	38.85883	6.078068	9	19	0.57443	6.08E-07
	x3	8	17	0.245598	2.5E-07	105	107	36.57789	6.078045	9	18	0.499598	6.08E-07
	x4	7	16	0.269876	1.2E-07	108	108	37.05587	1.606088	11	19	0.497876	1.61E-07
	x5	8	17	0.285795	2.34E-07	109	109	39.92016	3.842631	11	20	0.505795	3.85E-07
	x6	8	18	0.292245	3.12E-07	108	102	38.61406	6.637112	12	21	0.514245	6.64E-07
100,000	x1	8	12	0.492751	2.6E-07	67	109	76.89502	5.433651	8	19	0.952751	5.43E-07
	x2	8	13	0.485562	3.1E-07	101	68	74.15855	8.595878	9	20	0.951562	8.6E-07
	x3	9	14	0.465368	3.1E-07	102	98	73.81739	8.595862	10	22	0.938368	8.6E-07
	x4	8	16	0.469960	1.2E-07	104	99	75.59401	2.271393	11	22	0.949960	2.27E-07
	x5	8	17	0.482485	2.6E-07	107	100	32347.39	5.434048	18	19	0.962485	5.43E-07
	x6	9	18	0.499563	3.1E-07	110	102	85.46433	9.386472	19	19	1.158563	9.39E-07

TABLE V: Numerical Comparison of MHCG, CHCG, and DTCG1 Algorithms

		MHCG				CHCG				DTCG1			
Dim	INT PNT	NI	NF	TIME	NORM	NI	NF	TIME	NORM	NI	NF	TIME	NORM
1000	x1	8	20	0.020000	8.00E-08	199	188	1.290244	4.671462	29	53	0.064743	4.85E-07
	x2	6	18	0.018000	9.00E-08	188	189	1.315886	7.388845	19	48	0.151778	6.54E-07
	x3	7	22	0.022000	9.50E-08	198	201	1.294982	7.387425	56	59	0.279019	8.1E-07
	x4	10	25	0.025000	1.00E-07	301	202	1.154269	1.952326	67	61	0.272331	5.81E-07
	x5	12	30	0.028000	1.10E-07	56	205	1.408702	4.693267	68	66	0.173613	7.35E-07
	x6	11	28	0.026000	1.05E-07	56	209	1.270771	8.071602	76	56	0.180273	6.95E-07
10,000	x1	10	14	0.060000	1.10E-07	56	209	8.227707	1.71722	8	19	0.135072	3.35E-07
	x2	11	16	0.065000	1.20E-07	88	210	8.490477	2.716476	8	29	0.119232	3.5E-07
	x3	12	18	0.070000	1.30E-07	89	102	8.456101	2.716426	8	35	0.136197	3.34E-07
	x4	14	20	0.080000	1.40E-07	91	103	8.965257	6.596583	7	37	0.13341	7.34E-07
	x5	15	22	0.085000	1.50E-07	92	104	8.033488	1.718246	8	49	0.152509	6.02E-07
	x6	17	24	0.090000	1.60E-07	100	105	9.117542	2.966404	8	61	0.128501	3.83E-07
50,000	x1	8	10	0.280000	1.80E-07	100	106	37.75063	3.8421	28	19	0.558128	3.84E-07
	x2	10	12	0.300000	1.90E-07	103	109	38.85883	6.078068	29	20	0.57443	6.08E-07
	x3	12	14	0.320000	2.00E-07	105	107	36.57789	6.078045	43	22	0.499598	6.08E-07
	x4	14	16	0.340000	2.10E-07	108	108	37.05587	1.606088	51	29	0.497876	1.61E-07
	x5	15	18	0.360000	2.20E-07	109	109	39.92016	3.842631	111	34	0.505795	3.85E-07
	x6	18	20	0.370000	2.30E-07	108	102	38.61406	6.637112	121	44	0.514245	6.64E-07
100,000	x1	12	16	0.620000	2.60E-07	67	109	76.89502	5.433651	23	19	0.952751	5.43E-07
	x2	14	18	0.650000	2.70E-07	101	68	74.15855	8.595878	45	20	0.951562	8.6E-07
	x3	16	20	0.670000	2.80E-07	102	98	73.81739	8.595862	46	22	0.938368	8.6E-07
	x4	18	22	0.700000	2.90E-07	104	99	75.59401	2.271393	47	22	0.94996	2.27E-07
	x5	20	25	0.720000	3.00E-07	107	100	32347.39	5.434048	48	19	0.962485	5.43E-07
	x6	22	28	0.740000	3.10E-07	110	102	85.46433	9.386472	67	19	1.158563	9.39E-07

TABLE VI: Numerical Comparison of MHCG, CHCG, and DTCG1 Algorithms

		MHCG				CHCG				DTCG1			
Dim	INT PNT	NI	NF	TIME	NORM	NI	NF	TIME	NORM	NI	NF	TIME	NORM
1000	x1	10	8	0.680000	0.020000	134	141	0.140026	0	22	22	0.213738	1.6E-07
	x2	12	30	0.055000	0.015000	191	195	0.194481	0	34	65	0.092278	1.6E-07
	x3	15	25	0.030000	0.018000	58	65	0.080862	0	23	54	0.342401	1.6E-07
	x4	20	15	0.050000	0.022000	136	137	0.136742	0	8	34	0.192278	1.6E-07
	x5	22	30	0.065000	0.020000	107	114	0.136141	0	56	32	0.342401	1.6E-07
	x6	18	25	0.080000	0.025000	127	128	0.111113	0	68	23	0.212434	1.6E-07
10,000	x1	20	18	0.700000	0.030000	128	129	0.773108	0	7	22	0.895087	5.06E-07
	x2	22	20	0.080000	0.020000	141	145	0.944654	0	77	48	0.123213	0
	x3	24	22	0.085000	0.025000	60	67	0.622812	0	88	23	0.123213	1.6E-07
	x4	28	15	0.800000	0.030000	118	119	8.440762	0	88	34	0.888342	5.06E-07
	x5	32	25	0.120000	0.035000	213	214	2.701107	0	4	39	0.100919	0
	x6	35	18	0.750000	0.028000	120	121	1.079774	0	89	23	0.792278	5.06E-07
50,000	x1	25	12	2.500000	0.050000	120	121	3.554603	0	88	23	0.342401	0
	x2	30	20	0.200000	0.015000	257	261	6.799504	0	89	56	0.092278	1.6E-07
	x3	35	25	0.220000	0.018000	89	96	2.562572	0	92	78	0.342401	1.6E-07
	x4	27	15	0.250000	0.022000	118	119	3.198937	0	109	35	0.428215	0
	x5	30	20	0.550000	0.030000	315	316	8.31724	0	41	39	0.395074	0
	x6	33	15	0.900000	0.025000	111	112	2.826385	0	38	24	0.409435	0
100,000	x1	30	10	0.300000	0.040000	115	116	5.688736	0	38	23	0.632315	0
	x2	40	15	0.350000	0.050000	342	346	17.03736	0	77	34	0.123213	1.6E-07
	x3	45	20	0.400000	0.055000	201	208	9.653881	0	88	23	0.123213	1.6E-07
	x4	50	12	0.600000	0.050000	127	128	6.178666	0	89	35	0.778230	0
	x5	55	30	0.018000	0.025000	838	839	37.45256	0	89	39	0.712328	0
	x6	60	14	0.250000	0.030000	112	113	5.585985	0	92	24	0.685135	0

TABLE VII: Numerical Experiment of MHCG and SGCS in Image Restoration

		MHCG				SGCS			
Image	Size	ITE	TIMES	SNR	SSIM	ITE	TIMES	SNR	SSIM
Lena	256 x 256	80	12.00	26.00	0.94	277	10.95	24.11	0.90
House	256 x 256	90	13.50	24.50	0.93	346	13.80	22.00	0.88
Barbara	512 x 512	40	18.00	22.00	0.85	275	55.41	19.68	0.79
Pepper	256 x 256	15	1.20	24.50	0.91	292	11.44	22.81	0.87

Figure IV: It shows the original, blurred, and restored images by MHCG and SGCS.

REFERENCES

[1] M. Figueiredo, R. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction, application to compressed sensing and other inverse problems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 586-597, 2007.

[2] M. A. T. Mario, R. Figueiredo, and D. Nowak, "An EM algorithm for wavelet-based image restoration," *IEEE Transactions on Image Processing*, vol. 12, no. 8, pp. 906-916, 2003.

[3] E. H. Zarantonello, "Solving Functional Equations by Contractive Averaging," Tech. Rep., Vol. 160, U. S. Army Mathematics Research Center, Madison, Wisconsin, 1960.

[4] R. I. Kačurovskii, "On monotone operators and convex functionals," *Uspekhi Matematicheskikh Nauk*, vol. 15, no. 4, pp. 213-215, 1960.

[5] G. J. Minty, "Monotone (nonlinear) operators in Hilbert space," *Duke Mathematical Journal*, vol. 29, no. 3, pp. 413-429, 1962.

[6] M. Y. Waziri, W. J. Leong, and M. A. Hassan, "Jacobian-Free Diagonal Newtons Method for Solving Nonlinear Systems with Singular Jaco-

bian," *Malasian Journal of Mathematical Science*, vol. 5, pp. 241-255, 2011.
[7] D. Li and M. Fukushima, "A Globally and Superlinearly Convergent Gauss-Newton Based BFGS Method for Symmetric Equations," *SIAM Journal on Numerical Analysis*.
[8] K. Amini and F. Rostami, "A modified two steps Levenberg-Marquardt method for nonlinear equations," *Journal of Computational and Applied Mathematics*, vol. 288, pp. 341-350, 2015.
[9] J. Guo and Z. Wan, "A Modified Spectral PRP Conjugate Gradient Projection Method for Solving Large-Scale Monotone Equations and Its Application in Compressed Sensing," *Hindawi Mathematical Problems in Engineering*, 2019, Article ID 5261830.
[10] A. B. Abubakar, P. Kumam, and A. M. Auwal, "A Descent Dai-Liao Projection Method for Convex Constrained Nonlinear Monotone Equations with Applications," *Thai Journal of Mathematics*, pp. 128-152, 2018.
[11] H. Mohammad and A. B. Abubakar, "A descent derivative-free algorithm for nonlinear monotone equations with convex constraints," *RAIRO Operations Research*, vol. 54, pp. 489-505, 2020.
[12] P. Zoltan and R. Sanja, "FR type methods for systems of large-scale nonlinear monotone equations," *Applied Mathematics and Computation*, vol. 269, pp. 816-823, DOI: 10.1016/j.camwa.2015.09.014.
[13] Z. Papp and S. Rapajic, "FR type methods for systems of large-scale nonlinear monotone equations," *Appl. Math.Comput.*, vol. 269, pp. 816-823.
[14] B. T. Polyak, "The conjugate gradient method in extreme problems," *USSR Comput. Math. Math. Phys.*, vol. 9, pp. 94-112, 1969.
[15] M. R. Hestenes and E. Stiefel, "Method of conjugate gradient for solving linear systems," *J. Res. Natl. Bur. Stand.*, vol. 49, pp. 409-436, 1953.
[16] Y. Liu and C. Storey, "Efficient generalized conjugate gradient algorithms, Part 1: Theory," *JOTA*, vol. 69, pp. 129-137, 1991.
[17] R. Fletcher and C. M. Reeves, "Function minimization by conjugate gradients," *Comput. J.*, vol. 7, pp. 149-154, 1964.
[18] F. Roger, *Practical methods of optimization*, John Wiley & Sons, 2nd ed., 2013.
[19] Y. Dai and Y. Yuan, "A nonlinear conjugate gradient method with a strong global convergence property," *SIAM J. Optim.*, vol. 10, pp. 177-182, 1999.
[20] Z. Wei and G. Qi Li, "New nonlinear conjugate gradient method formulas for large-scaled unconstrained optimizations problems," *Appl. Math. Comp.*, vol. 179, pp. 407-430, 2006.
[21] G. Yuan and Z. Wei, "New line search methods for unconstrained optimization," *J. Korean Stat Soc.*, vol. 38, pp. 29-39, 2009.
[22] Z. J. Shi and J. Guo, "A new family of conjugate gradient methods," *J. Comput Appl. Math.*, vol. 224, pp. 444-457, 2009.
[23] W. W. Hager and H. Zhang, "A new conjugate gradient method with guaranteed descent and an efficient line search," *SIAM Journal on Optimization*, vol. 16, pp. 170-192, 2005.
[24] J. C. Gilbert and J. Nocedal, "Global convergence properties of conjugate gradient methods for optimization," *SIAM Journal on Optimization*, vol. 2, pp. 21-42, 1992.
[25] J. Sun and J. Zhang, "Global convergence of conjugate gradient methods without line search," *Annals. Operation Research.*, vol. 103, pp. 161-173, 2001.
[26] Y. H. Dai, "A family of hybrid conjugate gradient methods for unconstrained optimization," *Mathematics of Computation*, vol. 72, no. 243, pp. 1317-1328, 2003.
[27] N. Yuan, "A derivative-free projection method for solving convex constrained monotone equations," *Science Asia*, vol. 43, pp. 195-200, 2017.
[28] H. Abdullahi, A. K. Awasthi, M. Y. Waziri, and A. S. Halilu, "A scaled three-term conjugate gradient method for convex-constrained monotone nonlinear equations with applications," *Journal of Physics: Conference Series*, vol. 2267, p. 012066, 2022, doi:10.1088/1742-6596/2267/1/012066.
[29] M. Rivaie, M. Mamat, and A. Abashar, "A New class of nonlinear conjugate gradient coefficients with exact and inexact line searches," *Applied Mathematics and Computational*, vol. 268, pp. 1152-1163, 2015.
[30] Y. H. Dai and L. Z. Liao, "New conjugacy conditions and related nonlinear conjugate gradient methods," *Applied Mathematics and Optimization*, vol. 43, no. 1, pp. 87-101, 2001.
[31] H. Abdullahi, A. K. Awasthi, M. Y. Waziri, and A. S. Halilu, "Descent three-term DY-type conjugate gradient methods for constrained monotone equations with application," *Comp. and Appli. Math.*, vol. 41, p. 32, 2022.
[32] A. B. Abubakar, P. Kumam, H. Mohammad, A. M. Awwal, and S. Kanokwan, "A Modified Fletcher Reeves Conjugate Gradient Method for Monotone Nonlinear Equations with Some Applications," *Mathematics*, vol. 7, p. 745, 2019.
[33] M. Y. Waziri, K. Ahmed, A. S. Halilu, and A. M. Awwal, "Modified Dai-Yuan iterative scheme for nonlinear systems and its application," *Numerical Algebra, Control and Optimization*, 2021, doi:10.3934/naco.2021044.
[34] W. Xui, J. Ren, X. Zheng, Z. Liu, and Y. Liang, "A new DY conjugate gradient method and applications to image denoising," *IEICE Transactions on Information Systems*, vol. 101, pp. 2984-2990, 2018.
[35] A. S. Ibrahima, P. Kumam, A. B. Abubakara, W. Jirakitpuwapata, and J. Abubakar, "A hybrid conjugate gradient algorithm for constrained monotone equations with application in compressive sensing," *Heliyon*, vol. 6, 2020.
[36] A. S. Halilu, A. Majumder, M. Y. Waziri, and K. Ahmed, "Signal recovery with convex constrained nonlinear monotone equations through conjugate gradient hybrid approach," *Mathematics and Computers in Simulation*, 2021, http://doi.org/10.1016/j.matcom.2021.03.020.
[37] S. H. Khan, "A Picard–Mann hybrid iterative process," *Fixed Point Theory and Applications*, p. 69, 2013.
[38] A. M. Awwal, L. Wang, P. Kumam, and H. Mohammad, "A two-step spectral gradient Projection method for a system of nonlinear monotone equations and image deblurring problems," *Symmetry*, vol. 12, p. 874, 2020, https://doi.org/10.3390/sym12060874.
[39] E. D. Dolan and J. J. Moré, "Benchmarking optimization software with performance profiles," *Mathematical Programming*, Series A, vol. 91, pp. 201-213, 2002.
[40] M. Figueiredo, R. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction, application to compressed sensing and other inverse problems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 586-597, 2007.
[41] Y. Xiao, Q. Wang, and Q. Hu, "Non-smooth equations based method for ℓ_1 problems with applications to compressed sensing," *Nonlinear Analysis: Theory, Methods & Applications*, vol. 74, no. 11, pp. 3570-3577, 2011.
[42] Z. Li, W. Weijun, and D. Li, "A descent modified Polak-Ribiére-Polyak conjugate gradient method and its global convergence," *IMA Journal of Numerical Analysis*, vol. 26, no. 4, pp. 629-640, 2006.
[43] Q. Li and D. H. Li, "A class of derivative-free methods for large-scale nonlinear monotone equations," *IMA Journal of Numerical Analysis*, vol. 31, pp. 1625-1635, 2011.
[44] A. B. Abubakar, P. Kumam, M. Maulana, C. Parin, and H. A. Abdulkarim, "A hybrid FR-DY conjugate gradient algorithm for unconstrained optimization with application in portfolio selection," *AIMS Mathematics*, vol. 6, no. 6, pp. 6506-6527, 2021.
[45] M. V. Solodov and B. F. Svaiter, "A globally convergent inexact Newton method for systems of monotone equations," in: *Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods*, Springer, pp. 355-369, 1999.
[46] Z. Li, W. Zhang, and D. Li, "A descent modified Polak-Ribiére−Polyak conjugate gradient method and its global convergence," *IMA Journal of Numerical Analysis*, vol. 26, no. 4, pp. 629–640, 2006.
[47] X. Y. Wang, X. J. Li, and X. P. Kou, "A self-adaptive three-term conjugate gradient method for monotone nonlinear equations with covex constraints, calcolo, 53(2016)133-145. doi: 10.1007/s10092-015-0140-5.," 2016.
[48] Zhinan Xu, and Xiaoxia Zhang, "An Improve Grey Wolf Optimizer Algorithm for Traveling Salesman Problems, *IAENG International Journal of Computer Science*, vol. 51, no. 6, pp602-612, 2024.
[49] Yan-e Hou, Congran Wang, Chunyang Zhang, Lanxue Dang, and Chunjing Xiao, "A Hybrid Max-Min Ant System Algorithm for Electric Capacitated Vehicle Routing Problem, *IIAENG International Journal of Computer Science*, vol. 51, no. 3, pp195-203, 2024.
[50] Shou-Jun Sheng, and Zi-Wei Zhou, "Revolutionizing Image Captioning: Integrating Attention Mechanisms with Adaptive Fusion Gates, *IAENG International Journal of Computer Science*, vol. 51, no. 3, pp212-221, 2024.