Nonlinear Convex Analysis and Optimization
Vol. 3, No. 1, 2024, pp. 25–46
https://ncao.design.blog

A Descent Matrix-free Nonlinear Conjugate Gradient Algorithm for Impulse Noise Removal

Nasiru Salihu [a,b,1], Poom Kumam [a,c,2*], Ibrahim Mohammed Sulaiman [d,e,3], Suraj Salihu [f,4]

[a] Center of Excellence in Theoretical and Computational Science (TaCS-CoE). Fixed Point Research Laboratory,Fixed Point Theory and Applications Research Group, Faculty of Science, King Mongkut's University of Technology Thonburi (KMUTT), Bangkok 10140, Thailand
[b]Department of Mathematics, Faculty of Sciences, Modibbo Adama University, Yola, 652105, Nigeria.
[c] KMUTT-Fixed Point Research Laboratory, Room SCL 802, Science Laboratory Building, Department of Mathematics, Faculty of Science, King Mongkut's University of Technology Thonburi (KMUTT), Bangkok 10140, Thailand.
[d] Institute of Strategic Industrial Decision Modelling, School of Quantitative Sciences, Universiti Utara Malaysia, UUM Sintok, Kedah, Malaysia.
[e] Faculty of Education and Arts, Sohar University, Sohar 311, Oman.
[f] Department of Computer Science, Faculty of Science, Gombe State University, Gombe, Nigeria.
[1] nasirussalihu@gmail.com; [2] poom.kum@kmutt.ac.th; [3] sumlaimanzz@su.edu.om [4] surajsalihu@gmail.com
* Corresponding Author

ABSTRACT

The convergence of the Polak, Ribie're-Polyak (PRP) conjugate gradient (CG) method requires some modifications for improved theoretical properties. In this article, we explore an optimal choice of the Perry conjugacy condition to propose a hybrid CG parameter for solving optimization and inverse problems, particularly in an image reconstruction model. This parameter is selected to satisfy a combination of revised version of the PRP and Dai-Yuan (DY) CG methods. The numerical implementation includes inexact line search, showcasing the scheme's robustness (highest number of solved functions) compared to other known CG algorithms. The efficiency is shown in terms of Real error (RelErr), peak signal noise ratio (PNSR), and CPU time in seconds for impulse noise removal while for unconstrained minimization problems, the study evaluated the efficiency based on number of iterations, function evaluation, and CPU time in seconds. An interesting feature of the proposed method is its ability to converges to the minimizer regardless of the initial guess, relying on certain established assumptions.

Article History
Received 9 Jan 2024
Accepted 25 Jan 2024
Keywords:
Unconstrained optimization;
Conjugate gradient method;
Conjugacy condition;
Newton direction;
Global convergence;
Image reconstruction
MSC
90C29; 90C52; 90C30; 90C26; 49M37

Please cite this article as: Salihu et al., A Descent Matrix-free Nonlinear Conjugate Gradient Algorithm for Impulse Noise Removal, Nonlinear Convex Anal. & Optim., Vol. 3, No. 1, 25–46. https://doi.org/10.58715/ncao.2024.3.2

1. Introduction

In 1964, Fletcher and Reeves suggested the non-linear version of the CG model to minimize an unconstrained problem:

$$\min f(p),\ x \in \mathbb{R}^n, \tag{1.1}$$

where the gradient, $q(p) = \nabla f(p)$, exists for the smooth function $f : \mathbb{R}^n \longrightarrow \mathbb{R}$. Among the iterative schemes existing in the literature, the CG method stands out as particularly intriguing for addressing large-scale optimization models defined by equation (1.1). This is due to its favorable theoretical properties and absence of matrix storage [8]. Moreover, this method has garnered considerable attention in inverse and time-varying optimization problems, finding applications in image restoration [37, 6], signal recovery [17, 45], and robotic motion [43, 34]. The scheme initializes guess for the solution, generating a sequence of points $\{p_k\}$ utilizing the relation:

$$p_{k+1} = p_k + \ell_k h_k, \quad k = 0, 1, 2, \ldots. \tag{1.2}$$

At this stage, h_k signifies the search direction, while ℓ_k represents the step size. The line search process during the k-th iteration determines ℓ_k to fulfill specific criteria [4]. The conventional Wolfe line method involves the following set of inequalities:

$$f(p_k + \ell_k h_k) \leq f(p_k) + \delta \ell_k q_k^T h_k, \tag{1.3}$$

$$q(p_k + \ell_k h_k)^T h_k \geq \sigma q_k^T h_k. \tag{1.4}$$

However, the convergence of certain CG methods necessitates replacing (1.4) with

$$|q(p_k + \ell_k h_k)^T h_k| \leq -\sigma q_k^T h_k. \tag{1.5}$$

This criteria, combined, is referred to as the strong Wolfe condition, where $0 < \delta < \frac{1}{2}$, $\delta < \sigma < 1$, and is usually computed along the search direction h_k using

$$h_0 = -q_0,\ h_{k+1} = -q_{k+1} + \beta_k h_k, \tag{1.6}$$

the scalar parameter β_k, usually refers to as the CG formula, plays a pivotal role in defining the behavior of any CG method. While numerous options for the scalar β_k exist, selecting an appropriate one is crucial for enhancing both theoretical properties and numerical efficiency [16]. Some popular choices for β_k are provided in [15, 10, 14, 18, 32, 31, 25] along with the following formulas:

$$\beta_k^{FR} = \frac{\|q_{k+1}\|^2}{\|q_k\|^2}, \quad \beta_k^{DY} = \frac{\|q_{k+1}\|^2}{h_k^T r_k}, \quad \beta_k^{CD} = -\frac{\|q_{k+1}\|^2}{h_k^T q_k}, \tag{1.7}$$

$$\beta_k^{HS} = \frac{q_{k+1}^T r_k}{h_k^T r_k}, \quad \beta_k^{PRP} = \frac{q_{k+1}^T r_k}{\|q_k\|^2}, \quad \beta_k^{LS} = -\frac{q_{k+1}^T r_k}{h_k^T q_k}, \tag{1.8}$$

where $\|\cdot\|$ denotes the ℓ_2 norm and $r_k = q_{k+1} - q_k$.
The schemes outlined in (1.7) have demonstrated nice convergence properties, yet they face numerical uncertainties due to jamming [24]. To address this issue, the two CG types have been integrated. Particularly, combining β_k^{PRP} and β_k^{DY} CG schemes with the following parameters:

$$\beta_k^{PRP} = \frac{q_{k+1}^T r_k}{\|q_k\|^2}, \tag{1.9}$$

$$\beta_k^{DY} = \frac{\|q_{k+1}\|^2}{h_k^T r_k}. \tag{1.10}$$

Even though the PRP scheme is acknowledged as a highly effective CG parameter, it continues to face convergence challenges [7]. To tackle these issues, various modifications have been proposed, including the work by Wei et al. [42], as an example:

$$\beta_k^{WLY} = \frac{\|q_{k+1}\|^2 - \frac{\|q_{k+1}\|}{\|q_k\|} q_{k+1}^T q_k}{\|q_k\|^2},$$

where other variants of [42] have also been suggested by Zhang[46] and Dai et al. [11] as

$$\beta_k^{IWLY} = \frac{\|q_{k+1}\|^2 - \frac{\|q_{k+1}\|}{\|q_k\|} |q_{k+1}^T q_k|}{\|q_k\|^2},$$

$$\beta_k^{DPRP} = \frac{\|q_{k+1}\|^2 - \frac{\|q_{k+1}\|}{\|q_k\|} |q_{k+1}^T q_k|}{\mu |q_{k+1}^T h_k| + \|q_k\|^2},$$

respectively. Alternatively, in response to certain computational challenges linked to the DY method, Jiang and Jian [23] introduced its adapted form as follows:

$$\beta_k^{IDY} = r_k \frac{\|q_{k+1}\|^2}{h_k^T r_k},$$

where $r_k = \frac{|q_{k+1}^T h_k|}{-q_k^T h_k}$. Building upon the CG techniques discussed in [21, 22], [47] introduced two modified DY CG approaches:

$$\beta_k^{(1)} = \frac{\|q_{k+1}\|^2 - \frac{(q_{k+1}^T q_k)^2}{\|q_k\|^2}}{h_k^T r_k},$$

$$\beta_k^{(2)} = \frac{\|q_{k+1}\|^2 - \frac{\|q_{k+1}\|}{\|q_k\|} q_{k+1}^T q_k}{\mu h_k^T r_k}.$$

Following these adaptations, Zhu et al. [48] proposed the DDY1 scheme:

$$\beta_k^{DDY1} = \begin{cases} \frac{\|q_{k+1}\|^2 - \frac{\mu_1 (q_{k+1}^T d_k)^2}{\|q_{k+1}\| \|q_k\| \|d_k\|^2} q_{k+1}^T q_k}{h_k^T r_k}, & q_{k+1}^T q_k \geq 0, \\ \frac{\|q_{k+1}\|^2 + \frac{\mu_1 (q_{k+1}^T d_k)^2}{\|q_{k+1}\| \|q_k\| \|d_k\|^2} q_{k+1}^T q_k}{h_k^T r_k}, & q_{k+1}^T q_k < 0. \end{cases}$$

These techniques introduced additional complexities to the classical PRP and DY CG methods, potentially altering their original behavior. Moreover, these modifications often overlooked hybrid CG techniques. Consequently, there has been an increasing need for combining two or more CG methods to improve their overall performance [3]. The initial attempt of this combination was proposed by Touati-Ahmed and Storey [41], merging β_k^{PRP} and β_k^{FR} methods. Computational results demonstrate that this technique outperforms each of these strategies when implemented separately. Following this idea, to ensure non-negativity of β_k^{PRP}, Hu and Storey [19] introduced another structure of CG parameter based on distinct combination of

the schemes outlined in (1.7) and (1.8). However, algorithms that can automatically switch between the methods in the two categories are highly preferable for enhancing the nature of CG methods [39, 29]. Recent combinations exemplifying this approach include; LS and CD procedures [38, 20], HS and DY schemes [39, 1], PRP and FR methods [28], along with LS and DY methods [24], and PRP and DY methods [2]. Moreover, these schemes are primarily derived from the traditional conjugacy condition described by

$$h_{k+1}^T r_k = 0. \tag{1.11}$$

This criterion plays a pivotal role in establishing the convergence of various CG methods. However, for improved convergence and better numerical outcomes, more generalized forms of this condition, as proposed by Perry [30] and Dai and Liao [9], are often necessary

$$h_{k+1}^T r_k = -q_{k+1}^T s_k \tag{1.12}$$

$$h_{k+1}^T r_k = -t q_{k+1}^T s_k, \quad t \geq 0. \tag{1.13}$$

Motivated by the methodologies outlined in [20, 39] and considering the practical numerical benefits associated with the PRP method, we introduce a CG method that utilizes a hybrid parameter calculated from:

$$\beta_k = (1-\theta_k)\beta_k^{OPRP} + \theta_k \beta_k^{DY}, \tag{1.14}$$

where $\theta_k \in [0,1]$ is the convex combination parameter and *OPRP* stands for optimal PRP method which is inspired by the nice convergence structure of the β_k^{DY} to modify β_k^{PRP} as

$$\beta_k^{OPRP} = \frac{q_{k+1}^T r_k}{\max\{\|q_k\|^2,\ h_k^T r_k\}}. \tag{1.15}$$

This modification ensures desirable convergence features and effective numerical performance of the hybrid parameter. Thus, next section systematically suggests a new hybrid parameter based on the generalized conjugacy condition. Section three describes the overall convergence characteristic of the proposed. In the fourth section, the stability and competitive nature of the proposed scheme are presented by comparing it with various known CG methods using different benchmark optimization problems. The new hybrid algorithm is further employed to solve an image reconstruction problem. Finally, the last section entails a brief conclusion.

2. Formulation of the Hybrid Method

To obtain the CG parameter as a revised version of the β_k in (1.6) combining (1.10) and (1.15), we define the hybrid scheme as

$$\begin{aligned}\beta_k &= (1-\theta_k)\beta_k^{OPRP} + \theta_k \beta_k^{DY} \\ &= \beta_k^{OPRP} + \theta_k(\beta_k^{DY} - \beta_k^{OPRP}) \\ &= \frac{q_{k+1}^T r_k}{\max\{\|q_k\|^2,\ h_k^T r_k\}} + \theta_k\left(\frac{\|q_{k+1}\|^2}{h_k^T r_k} - \frac{q_{k+1}^T r_k}{\max\{\|q_k\|^2,\ h_k^T r_k\}}\right).\end{aligned} \tag{2.1}$$

Multiplying (1.6) by r_k^T and using (2.1), we get

$$\begin{aligned} h_{k+1}^T r_k &= -q_{k+1}^T r_k + \beta_k h_k^T r_k \\ &= -q_{k+1}^T r_k + \frac{q_{k+1}^T r_k}{\max\{\|q_k\|^2,\ h_k^T r_k\}} h_k^T r_k + \theta_k \left(\frac{\|q_{k+1}\|^2}{h_k^T r_k} - \frac{q_{k+1}^T r_k}{\max\{\|q_k\|^2,\ h_k^T r_k\}} \right) h_k^T r_k. \end{aligned} \tag{2.2}$$

Equating (1.13) with (2.2) gives

$$-ts_k^T q_{k+1} = -q_{k+1}^T r_k + \frac{q_{k+1}^T r_k}{\max\{\|q_k\|^2,\ h_k^T r_k\}} h_k^T r_k + \theta_k \left(\frac{\|q_{k+1}\|^2}{h_k^T r_k} - \frac{q_{k+1}^T r_k}{\max\{\|q_k\|^2,\ h_k^T r_k\}} \right) h_k^T r_k. \tag{2.3}$$

There are two cases for $\max\{\|q_k\|^2,\ h_k^T r_k\}$.

Case I: When $\max\{\|q_k\|^2,\ h_k^T r_k\} = \|q_k\|^2$ this follows from (2.3) that

$$\frac{(q_{k+1}^T r_k)\|q_k\|^2 - (ts_k^T q_{k+1})\|q_k\|^2 - (q_{k+1}^T r_k) h_k^T r_k}{\|q_k\|^2} = \theta_k \left(\frac{\|q_{k+1}\|^2}{h_k^T r_k} - \frac{q_{k+1}^T r_k}{\|q_k\|^2} \right) h_k^T r_k.$$

So that after re-arranging, we obtain

$$\theta_k^1 = \frac{(q_{k+1}^T r_k - ts_k^T q_{k+1})\|q_k\|^2 - (q_{k+1}^T r_k) h_k^T r_k}{\|q_{k+1}\|^2 \|q_k\|^2 - (q_{k+1}^T r_k) h_k^T r_k}. \tag{2.4}$$

Case II: When $\max\{\|q_k\|^2,\ h_k^T r_k\} = h_k^T r_k$, then this follows directly by changing $\|q_k\|^2$ with $h_k^T r_k$ in (2.4) and after some simplifications, we obtain

$$\theta_k^2 = \frac{(q_{k+1}^T r_k - ts_k^T q_{k+1}) - (q_{k+1}^T r_k)}{\|q_{k+1}\|^2 - (q_{k+1}^T r_k)}. \tag{2.5}$$

To make an optimal choice, we select the parameter θ_k in a way that if $\theta_k \leq 0$, we set $\beta_k = \beta_k^{OPRP}$; if $\theta_k \geq 1$, we set $\beta_k = \beta_k^{DY}$. Otherwise, when $0 < \theta_k < 1$, then β_k includes both β_k^{OPRP} and β_k^{DY}. Therefore, these selections of θ_k^1 and θ_k^2 imply that we can compute the optimal hybrid parameter as

$$\theta_k^* = \begin{cases} 0, & \text{if } \ \theta_k \leq 0, \\ \hat{\theta}_k, & \text{if } \ 0 < \theta_k < 1, \\ 1, & \text{if } \ \theta_k \geq 1, \end{cases} \tag{2.6}$$

with

$$\hat{\theta}_k = \begin{cases} \theta_k^1, & \text{if } \ \|q_k\|^2 > h_k^T r_k, \\ \theta_k^2, & \text{otherwise.} \end{cases} \tag{2.7}$$

Now, based on (2.1), we can represent h_{k+1} as a convex combination of β_k^{OPRP} and β_k^{DY}. Therefore, from (1.6), we have

$$\begin{aligned} h_{k+1} &= -(\theta_k q_{k+1} + (1-\theta_k) q_{k+1}) + \beta_k h_k \\ &= \theta_k(-q_{k+1} + \beta_k^{DY} h_k) + (1-\theta_k)(-q_{k+1} + \beta_k^{OPRP} h_k) \\ &= \theta_k h_{k+1}^{DY} + (1-\theta_k) h_{k+1}^{OPRP}. \end{aligned} \tag{2.8}$$

Now, let's describe the new CG method called the PPRPDY method as follows:

Algorithm 1: PPRPDY

Step 1: Let $0 < \delta < \sigma < 1$ and $x_0 \in \mathbb{R}^n$ be given. Then $h_0 = -q_0$,
Step 2: If $\|q_k\| \leq \epsilon$, then stop. Otherwise
Step 3: Compute $\ell_k > 0$ that fulfils (1.3) and (1.5).
Step 4: Compute β_k and θ_k^* as in (2.1) and (2.6), respectively.
Step 5: Check the condition

$$|q_{k+1}^T q_k| > 0.2\|q_{k+1}\|^2, \tag{2.9}$$

is satisfied, so that h_{k+1} is steepest descent or otherwise CG direction.
Step 6: Update the next iterate from Step 2.

Remark 2.1. It's evident that if $t = 0$, the convex combination parameters θ_k satisfy the pure conjugacy condition, that is, $h_{k+1}^T r_k = 0$. However, in the PPRPDY Algorithm, if $t = 1$, Perry's condition (1.12) is satisfied. Therefore, for better numerical outcomes, we consistently opt for $t = 1$ in this paper.

3. Convergence Analysis

In this section, we explore the convergence characteristics of the suggested hybrid method. To proceed, we state the following assumptions.

Assumption 3.1. The level set $\eta = \{p \in \mathbb{R}^n : f(p_0) \geq f(p)\}$ is bounded, i.e. there exists a constant $B > 0$ such that

$$\|p\| \leq B, \forall p \in \eta.$$

Assumption 3.2. Denoting η as some neighborhood of Γ, and function f is smooth with its gradient is Lipschitz continuous satisfying

$$\|q(p) - q(v)\| \leq L\|p - v\|, \quad \forall p, v \in \eta, \quad L > 0. \tag{3.1}$$

Note that these assumptions, imply that

$$\|q(p)\| \leq \gamma, \quad \forall p \in \Gamma, \quad \gamma > 0, \tag{3.2}$$
$$\|p - v\| \leq b, \quad \forall p \in \Gamma, \quad b > 0. \tag{3.3}$$

Next, we establish the sufficient descent of PPRPDY method.

Theorem 3.3. *Assume that assumptions (3.1)-(3.2) hold. Suppose that the PPRPDY Algorithm generates sequences $\{q_k\}$ and $\{h_k\}$ and let strong Wolfe rules (1.3)-(1.5) hold with $\sigma < \frac{1}{2}$. Then*

$$h_{k+1}^T q_{k+1} \leq -c\|q_{k+1}\|^2, \quad \forall k \geq 0. \tag{3.4}$$

Proof. Pre-multiply (2.8) by q_{k+1}^T, we have

$$q_{k+1}^T h_{k+1} = \theta_k q_{k+1}^T h_{k+1}^{DY} + (1 - \theta_k) q_{k+1}^T h_{k+1}^{OPRP}. \tag{3.5}$$

Case I: When $\theta_k \leq 0$, we set $\theta_k = 0$, which means $\beta_k = \beta_k^{OPRP}$.
If the restart criterion of Powell [33], as expressed in condition (2.9), does not hold, then the following inequalities arise:

$$|q_{k+1}^T q_k| \leq 0.2\|q_{k+1}\|^2. \tag{3.6}$$

Since $r_k = q_{k+1} - q_k$, using (3.6), we get

$$|q_{k+1}^T r_k| \leq 1.2\|q_{k+1}\|^2.$$

And

$$h_k^T r_k = h_k^T q_{k+1} - h_k^T q_k \geq -(1-\sigma)h_k^T q_k \geq 0.$$

So that (1.5) gives

$$|q_{k+1}^T h_k| \leq -\sigma h_k^T q_k. \tag{3.7}$$

Now, by combining the aforementioned inequalities with equations (1.15) and (3.5), we obtain

$$\begin{aligned}
q_{k+1}^T h_{k+1} &= -\|q_{k+1}\|^2 + \beta_k^{OPRP} d_k^T q_{k+1} \\
&\leq -\|q_{k+1}\|^2 + \frac{|q_{k+1}^T r_k|}{\max\{\|q_k\|^2, h_k^T r_k\}}|h_k^T q_{k+1}| \\
&\leq -\|q_{k+1}\|^2 - \frac{1.2\sigma\|q_{k+1}\|^2}{\max\{\|q_k\|^2, h_k^T r_k\}} q_k^T h_k \\
&= -\|q_{k+1}\|^2 - \frac{1.2\sigma\|q_{k+1}\|^2}{h_k^T r_k} q_k^T h_k \\
&\leq -\|q_{k+1}\|^2 + \frac{1.2\sigma\|q_{k+1}\|^2}{(1-\sigma)q_k^T h_k} q_k^T h_k \\
&\leq -\|q_{k+1}\|^2 + \frac{1.2\sigma}{1-\sigma}\|q_{k+1}\|^2 \\
&= -\tau_1\|q_{k+1}\|^2,
\end{aligned} \tag{3.8}$$

where $\tau_1 = (1 - \frac{1.2\sigma}{1-\sigma})$.

Case II: Now, when $\theta_k \geq 1$, we set $\beta_k = \beta_k^{DY}$, so that from (1.10) and (3.5), we obtain

$$\begin{aligned}
q_{k+1}^T h_{k+1} &\leq -\|q_{k+1}\|^2 + \frac{\|q_{k+1}\|^2}{d_k^T r_k}|h_k^T q_{k+1}| \\
&\leq -\frac{(1-2\sigma)}{1-\sigma}\|q_{k+1}\|^2 \\
&\leq -\tau_2\|q_{k+1}\|^2,
\end{aligned} \tag{3.9}$$

where $\tau_2 = \frac{(1-2\sigma)}{1-\sigma}$.

Case III: When $\theta_k \in (0,1)$, that is, there exists some constants μ_1, μ_2 such that $0 < \mu_1 \leq \theta_k \leq \mu_2 < 1$, then from (3.5), (3.8) and (3.9), we get

$$q_{k+1}^T h_{k+1} = \mu_1 q_{k+1}^T h_{k+1}^{DY} + (1-\mu_2) q_{k+1}^T h_{k+1}^{OPRP}.$$

In consequence, denoting $c = \mu_1\tau_2 + (1-\mu_2)\tau_1$, we end-up getting

$$q_{k+1}^T h_{k+1} \leq -c\|q_{k+1}\|^2, \tag{3.10}$$

which shows that (3.4) holds for $k+1$. ∎

Lemma 3.4. *Consider the CG methods (1.2)-(1.3), where d_k is sufficiently descent, and ℓ_k satisfies the strong Wolfe condition. Then*

$$\sum_{k=0}^{\infty} \frac{\|q_k\|^4}{\|h_k\|^2} < +\infty. \tag{3.11}$$

Theorem 3.5. *Suppose that the PPRPDY Algorithm generated sequences $\{p_k\}$ and $\{h_k\}$, where its search direction h_k is descending, and ℓ_k satisfies strong Wolfe conditions. Then*

$$\liminf_{k\to\infty} \|q_k\| = 0. \tag{3.12}$$

Proof. Suppose by contradiction that (3.12) is false. Then there exists $\omega > 0$ such that

$$\|q_k\| \geq \omega, \quad \forall k \geq 0. \tag{3.13}$$

Next, we claim that the direction set by equation (1.6) is lower bounded by a constant $F > 0$, i.e.

$$\|h_{k+1}\| \leq F, \quad \forall k \geq 0. \tag{3.14}$$

We prove this claim by induction. Now, from (3.1) and (3.3) , we can write $\|r_k\| = \|q_{k+1} - q_k\| \leq L\|p_{k+1} - p_k\| \leq Lb$.

Therefore, using (1.10), (1.15), (2.1) and (3.2), we have

$$\begin{aligned}
|\beta_k| &= |(1-\theta_k)\beta_k^{OPRP} + \theta_k\beta_k^{DY}| \\
&= \left|(1-\theta_k)\frac{q_{k+1}^T r_k}{\max\{\|q_k\|^2, h_k^T r_k\}} + \theta_k \frac{\|q_{k+1}\|^2}{\|q_k\|^2}\right| \\
&\leq \left|\frac{q_{k+1}^T r_k}{\max\{\|q_k\|^2, h_k^T r_k\}} + \frac{\|q_{k+1}\|^2}{\|q_k\|^2}\right| \\
&\leq \frac{\|q_{k+1}\|\|r_k\|}{\|q_k\|^2} + \frac{\|q_{k+1}\|^2}{\|q_k\|^2} \\
&\leq \frac{Lb\|q_{k+1}\| + \|q_{k+1}\|^2}{\|q_k\|^2} \\
&\leq \frac{Lb\|q_{k+1}\| + \gamma^2}{\|q_k\|^2} \\
&\leq \frac{\gamma Lb + \gamma^2}{\omega^2} = G.
\end{aligned} \tag{3.15}$$

From step 1 of the PPRPDY algorithm, we have, $h_1 = -q_1 + \beta_1 h_0$, implying that $h_1 = -q_1 - \beta_1 q_0$, since $h_0 = -q_0$, we have

$$\begin{aligned}
\|h_1\| &\leq \|q_1\| + |\beta_1|\|q_0\| \\
&\leq \gamma + G\gamma = \gamma^*.
\end{aligned}$$

By assuming that the claim (3.14) hold for k, we can show for $k+1$ from relation (1.6)

$$h_{k+1} = -q_{k+1} + \beta_k h_k.$$

Using (3.2) together with (3.15) gives

$$\begin{aligned}\|h_{k+1}\| &\leq \|q_{k+1}\| + |\beta_k|\|h_k\| \\ &\leq \gamma + GF.\end{aligned}$$

This shows that the claim holds but from (3.4), we have $(h_{k+1}^T q_{k+1})^2 \geq c^2\|q_{k+1}\|^4$, dividing by $\|h_{k+1}\|^2$ and summing gives

$$c^2 \sum_{k=0}^{\infty} \frac{\|q_{k+1}\|^4}{\|h_{k+1}\|^2} \geq \sum_{k=0}^{\infty} \frac{c^2\omega^4}{(\gamma + GF)^2} \geq \frac{c^2\omega^4}{(\gamma + GF)^2} \sum_{k=0}^{\infty} 1 = +\infty. \tag{3.16}$$

Clearly, this presents a contradiction since equations (3.11) and (3.16) cannot hold simultaneously. Hence, the proof is complete. ∎

4. Numerical Results

This section demonstrates the experimentation of the PPRPDY method in comparison to the HYB [20], HLSDY[24] and HLSFR [12] methods using 121 test problems considered in [5] and [27]. To exhibit the method's stability, the implemented test problems range in dimensions from 2 to 100, 000, as outlined in Table 1. The codes are implemented in MATLAB 9.12 (R2022a) on an Intel(R) Core i7-1195G7 PC with 16 GB RAM and a 2.90 GHz CPU. During the code execution, parameters $\delta = 10^{-3}$ and $\sigma = 10^{-6}$ are set in relations (1.3) and (1.5). The termination condition for all schemes is set as $\|q_k\| \leq 10^{-6}$ to halt the code executions. Regarding the numerical results of the PPRPDY, HYB, HLSDY, and HLSFR methods, we interpreted the results in figures based on three metric consisting; number of iterations , number of function evaluations and central processing unit. Moreover, the interpretation of the results is facilitated by a tool introduced by Dolan and Moré[13] known as a performance profile. Let P represent the collection of n_p test problems and S denote the set of n_s solvers used in the comparison. This profile serves as a measurement for each problem $p \in P$ and solver $s \in S$, representing the three metrics required to solve any problem. The measure of the performance ratio used to compare and evaluate the solvers' performance is given by

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

Consequently, the best method is indicated by the top curve on the performance profile plot. The experiments are interpreted graphically through Figures 1–3, demonstrating that the PPRPDY method is efficient and preferable over other methods across the three metrics. Specifically, the PPRPDY method exhibits the best performance by successfully solving an average of 68% of 78% of the problems. This establishes it as the superior solver against HYB, HLSDY, and HLSFR methods, which respectively achieve 11% of 72%, 18% of 68% and 10% of 72%, as observed in Figures 1–2. Similarly, in Figure 3, the PPRPDY prevails in 20% of 60%, compared to HYB, HLSDY, and HLSFR methods, which secure 20% of 60% and 20% of 65% respectively as the best methods. Thus, the curve representing the PPRPDY method consistently remains at the top when compared to other methods, affirming its effectiveness.

Table 1. List of test problems

Problem	Function	Dimension	Initial Point
1	Arwhead	10	(7,7,. . . ,7)
2	Denschn A	100	(6,6,. . . ,6)
3	Denschn A	500	(6,6,. . . ,6)
4	Denschn A	1000	(6,6,. . . ,6)
5	Denschn A	3000	(6,6,. . . ,6)
6	Denschn A	15000	(6,6,. . . ,6)
7	Denschn C	100	(100,100,. . . ,100)
8	Denschn C	500	(100,100,. . . ,100)
9	Denschn C	1000	(100,100,. . . ,100)
10	Denschn F	2	(0.001,0.001,. . . ,0.001)
11	Denschn F	500	(0.001,0.001,. . . ,0.001)
12	Denschn F	5000	(0.001,0.001,. . . ,0.001)
13	Denschn F	10000	(0.001,0.001,. . . ,0.001)
14	Denschn F	50000	(0.001,0.001,. . . ,0.001)
15	Diagonal 5	10	(1.1,1.1,. . . ,1.1)
16	Diagonal 5	5000	(1.1,1.1,. . . ,1.1)
17	Diagonal 5	50000	(1.1,1.1,. . . ,1.1)
18	Diagonal 1	4	(-33,-33,-33,-33)
19	Diagonal 1	10	(-33,-33,...,-33)
20	Diagonal 1	10	(-21,-21,...-12)
21	Diagonal 2	1000	(0.0010,0.0010,. . . ,0.0010)
22	Diagonal 2	10000	(0.0010,0.0010,. . . ,0.0010)
23	Diagonal 2	50000	(0.0010,0.0010,. . . ,0.0010)
24	Diagonal 2	100000	(0.0010,0.0010,. . . ,0.0010)
25	Diagonal 3	2	(11,11)
26	Diagonal 3	4	(13,13,13,13)
27	Diagonal 3	10	(13,13,. . . ,13)
28	Diagonal 3	10	(11,11,. . . ,11)
29	Diagonal 4	100	(1,1,...,1)
30	Diagonal 4	500	(1,1,...,1)
31	Diagonal 4	1000	(1,1,...,1)
32	Diagonal 4	10000	(1,1,...,1)
33	Diagonal 4	100000	(1,1,...,1)
34	Diagonal 6	100	(-1.01,-1.01,. . . ,-1.01)
35	Diagonal 6	500	(-1.01,-1.01,. . . ,-1.01)
36	Diagonal 7	50	(1,1,...,1)
37	Diagonal 7	100	(1,1,...,1)
38	Diagonal 9	10	(-7,-7,. . . ,-7)
39	Diagonal 9	100	(-7,-7,. . . ,-7)
40	Dqdrtic	10	(11,11,. . . ,11)

Problem	Function	Dimension	Initial Point
41	Dqdrtic	100	(2,2,. . . ,2)
42	Dqdrtic	200	(2,2,. . . ,2)
43	Dqdrtic	1000	(2,2,. . . ,2)
44	Dqdrtic	2000	(2,2,. . . ,2)
45	Dqdrtic	10000	(2,2,. . . ,2)
46	El-Attar-Vidyasagar-Dutta	2	(11,11)
47	Engval 1	2	(-1.01,-1.01)
48	Engval 1	2	(-1.03,-1.03)
49	Engval 8	2	(0.003,0.003)
50	Engval 8	4	(0.003,0.003,. . . ,0.003)
51	Ext Denschnb	200	(-0.03,-0.03,...-0.03)
52	Ext Denschnb	500	(-0.03,-0.03,...-0.03)
53	Ext Denschnb	1000	(-0.03,-0.03,...-0.03)
54	Ext Denschnb	2000	(-0.03,-0.03,...-0.03)
55	Extended Denschn B	100	(0.03,0.03,. . . ,0.03)
56	Extended Denschn B	5000	(0.03,0.03,. . . ,0.03)
57	Extended Denschn B	10000	(0.03,0.03,. . . ,0.03)
58	Extended Himmelblau	100	(-8,-8,. . . ,-8)
59	Extended Himmelblau	500	(-8,-8,. . . ,-8)
60	Extended Himmelblau	1000	(-8,-8,. . . ,-8)
61	Extended Himmelblau	2000	(-8,-8,. . . ,-8)
62	Extended Maratos	2	(21,21)
63	Extended Maratos	30	(21,21,...,21)
64	Extended Tridiagonal 1	2	(-11,-11)
65	Extended Tridiagonal 1	10	(-11,-11,. . . ,-11)
66	Extended Tridiagonal 1	100	(-11,-11,. . . ,-11)
67	Extended Tridiagonal 1	500	(-11,-11,. . . ,-11)
68	Fletchcer	1000	(0.99,0.99,. . . ,0.99)
69	Himmelblau	1000	(0.03,0.03,. . . ,0.03)
70	Himmelblau	10000	(0.03,0.03,. . . ,0.03)
71	Himmelblau	10000	(0.2,0.2,. . . ,0.2)
72	Himmelblau	50000	(0.2,0.2,. . . ,0.2)
73	Linear Perturbed	1000	(13,13,. . . ,13)
74	Linear Perturbed	2000	(13,13,. . . ,13)
75	Matyas	2	(11,11)
76	Matyas	2	(0.5,0.5)
77	Power	2	(1,1)
78	Power	2	(15,15)
79	Price4	2	(-0.03,-0.03)
80	Price4	2	(-0.01,-0.01)

Problem	Function	Dimension	Initial Point
81	Quadratic Function 1	10	(1,1,. . . ,1)
82	Quadratic Function 1	200	(1,1,. . . ,1)
83	Quadratic Function 2	100	(1,1,. . . ,1)
84	Quadratic Function 2	500	(1,1,. . . ,1)
85	Quadratic Penalty 1	4	(-1,-1,-1,-1)
86	Quadratic Penalty 1	100	(-1,-1,...,-1)
87	Quadratic Penalty 1	100	(1,1,. . . ,1)
88	Quadratic Penalty 1	1000	(-1,-1,. . . ,-1)
89	Quadratic Penalty 1	2000	(-1,-1,. . . ,-1)
90	Quadratic Penalty 1	5000	(-1,-1,. . . ,-1)
91	Quadratic Penalty 1	10000	(-1,-1,...,-1)
92	Quadratic Penalty 2	100	(10,10,. . . ,10)
93	Quadratic Penalty 2	500	(10,10,. . . ,10)
94	Quartic	500	(0.15,0.15,. . . ,0.15)
95	Quartic	1000	(0.15,0.15,. . . ,0.15)
96	Raydan 1	10	(1,1,. . . ,1)
97	Raydan 1	10	(-0.5,-0.5,. . . ,-0.5)
98	Raydan 1	50	(1,1,. . . ,1)
99	Raydan 1	50	(-0.5,-0.5,. . . ,-0.5)
100	Raydan 1	100	(1,1,. . . ,1)
101	Raydan 2	10000	(0.3,0.3,. . . ,0.3)
102	Raydan 2	100000	(0.3,0.3,. . . ,0.3)
103	Ref	2	(0.03,0.03)
104	Shallow	100	(10,10,. . . ,10)
105	Shallow	200	(10,10,. . . ,10)
106	Shallow	500	(10,10,. . . ,10)
107	Shallow	1000	(10,10,. . . ,10)
108	Shallow	10000	(10,10,. . . ,10)
109	Six Hump	2	(19,19)
110	Six Hump	2	(0.01,0.01)
111	Test	3	(10,10,10)
112	Test	3	(7,7,7)
113	Thump	2	(0.001,0.001)
114	Trecanni	2	(0.5,0.5)
115	Trecanni	2	(1.05,1.05)
116	Twh	2	(-15,-15)
117	Twh	2	(-25,0.25)
118	Zettl	2	(13,-0.013)
119	Zirilli	2	(0.30,0.03)
120	Zirilli	2	(0.31,0.31)
121	Zirilli	2	(0.29,0.29)

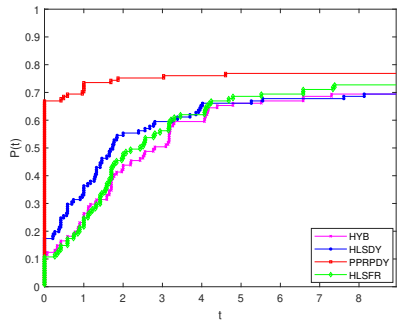

Fig. 1. Performance of the methods based on iterations

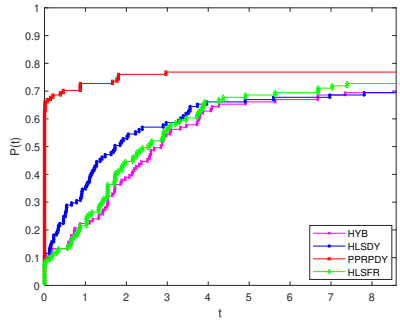

Fig. 2. Performance of the methods based on function evaluation

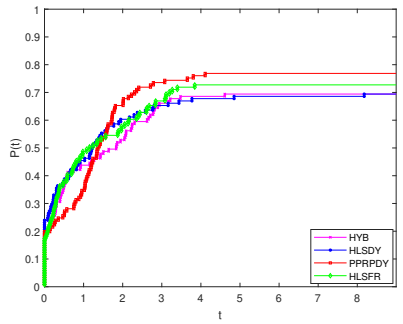

Fig. 3. Performance of the methods based on time

5. Application of PPRPDY in Image Restoration Problem

Image restoration is the process of estimating original and clean images from noisy or corrupted images. This corruption could result from camera mis-focus, motion blur, or impulse noise, and can be corrected by imaging the point sources. Using the point source images helps restore information lost in the blurring process for all images. This problem of restoring corrupted images is currently gaining attention from researchers due to its importance in the fields of security, health, sciences, and engineering [26, 36]. Recent studies on this problem have considered gradient-based algorithms to restore original images previously corrupted by salt-and-pepper impulse noise [45, 44]. Gradient-based methods are characterized by simplicity, low memory requirements, and favorable convergence results [16, 40, 35].

In this section, we investigate the performance of the proposed PPRPDY gradient-based algorithm on the image restoration problem to illustrate its efficiency and applicability in handling a broader range of practical real-life problems. For this study, we restore the following images: Building (512×512) and Forest (512×512), which have been corrupted by salt-and-pepper impulse noise. We measure the quality of the restored images using peak signal-to-noise ratio (PSNR), relative error (RelErr), and CPU time.

5.1. Image Restoration Problem

Consider the following index set of noise candidates for true image x :

$$G = \{(i,j) \in Q | \bar{\xi}_{ij} \neq \xi_{ij},\ \xi_{ij} = s_{\min} \text{ or } s_{\max}\}, \tag{5.1}$$

with $M \times N$ pixels, where $i, j \in Q = \{1, 2, \cdot, M\} \times \{1, 2, \cdot, N\}$ and the neighborhood of (i,j) is defined as

$$T_{i\,j} = \{(i,\ j-1), (i, j+1), (i-1, j), (i+1, j)\}. \tag{5.2}$$

From (5.1),

- ξ denotes the observed noisy image corrupted image,
- $\bar{\xi}$ defines the adaptive median filter to the noisy image ξ,
- $s_{\min}$ represents the minimum of a noisy pixel,

- $s_{\max}$ is the maximum of a noisy pixel.

The image restoration problem is modeled into the following optimization problem [44]:

$$\min \mathcal{H}(u),$$

where

$$\mathcal{H}(u) = \sum_{(i,j)\in G} \left\{ \sum_{(m,n)\in T_{i,j}/G} \phi_\alpha(u_{i,j} - \xi_{m,n}) + \frac{1}{2} \sum_{(m,n)\in T_{i,j}\cap G} \phi_\alpha(u_{i,j} - u_{m,n}) \right\}.$$

The function ϕ_α represent an edge-preserving potential function whose value is obtained as follows: $\phi_\alpha(t) = \sqrt{t^2 + \alpha}$ with the value of $\alpha = 1$.

The results of the computational experiments are presented in Tables 2, 3, and 4 below. To ascertain the efficiency and robustness of the proposed method, the performance results were compared with other classical CG algorithms, including HYB [20], HLSDY[24] and HLSFR [12] using strong Wolfe Powell line search. For each corrupted image, the noise degrees used for the restoration are set as 30, 50, and 80, respectively.

Table 2. Image restoration outputs for PPRPDY, HYB, HLSDY, and HLSFR based on CPUT

METHOD		PPRPDY	HYB	HLSDY	HLSFR
IMAGE	NOISE	CPUT	CPUT	CPUT	CPUT
BUILDING	30%	62.4159	62.6282	62.4700	62.4270
	50%	102.0279	101.9619	104.0741	103.6416
	80%	194.3993	193.8164	232.6671	231.2071
FOREST	30%	63.4620	63.0381	63.2526	63.4626
	50%	110.6644	145.0043	106.2185	112.4114
	80%	161.4432	161.2821	193.7045	193.5787

Table 3. Image restoration outputs for PPRPDY, HYB, HLSDY, and HLSFR based on RelErr

METHOD		PPRPDY	HYB	HLSDY	HLSFR
IMAGE	NOISE	RelErr	RelErr	RelErr	RelErr
BUILDING	30%	1.4104	1.4329	1.4783	1.4414
	50%	2.5548	2.6700	2.5230	2.5761
	80%	4.8313	4.8461	5.0861	4.8323
FOREST	30%	1.0936	1.0979	1.1496	1.1285
	50%	1.6217	1.6243	1.6532	1.6647
	80%	2.5026	2.5598	2.7502	2.7449

Table 4. Image restoration outputs for PPRPDY, HYB, HLSDY, and HLSFR based on PSNR

METHOD		PPRPDY	HYB	HLSDY	HLSFR
IMAGE	NOISE	PSNR	PSNR	PSNR	PSNR
BUILDING	30%	29.6317	29.7734	29.7463	29.9338
	50%	26.2685	26.0247	26.5364	26.5825
	80%	22.4042	22.4246	22.3326	22.5295
FOREST	30%	28.4333	28.5001	28.3767	28.4810
	50%	25.4001	25.4312	25.5076	25.3647
	80%	22.1955	21.9519	21.7812	21.8414

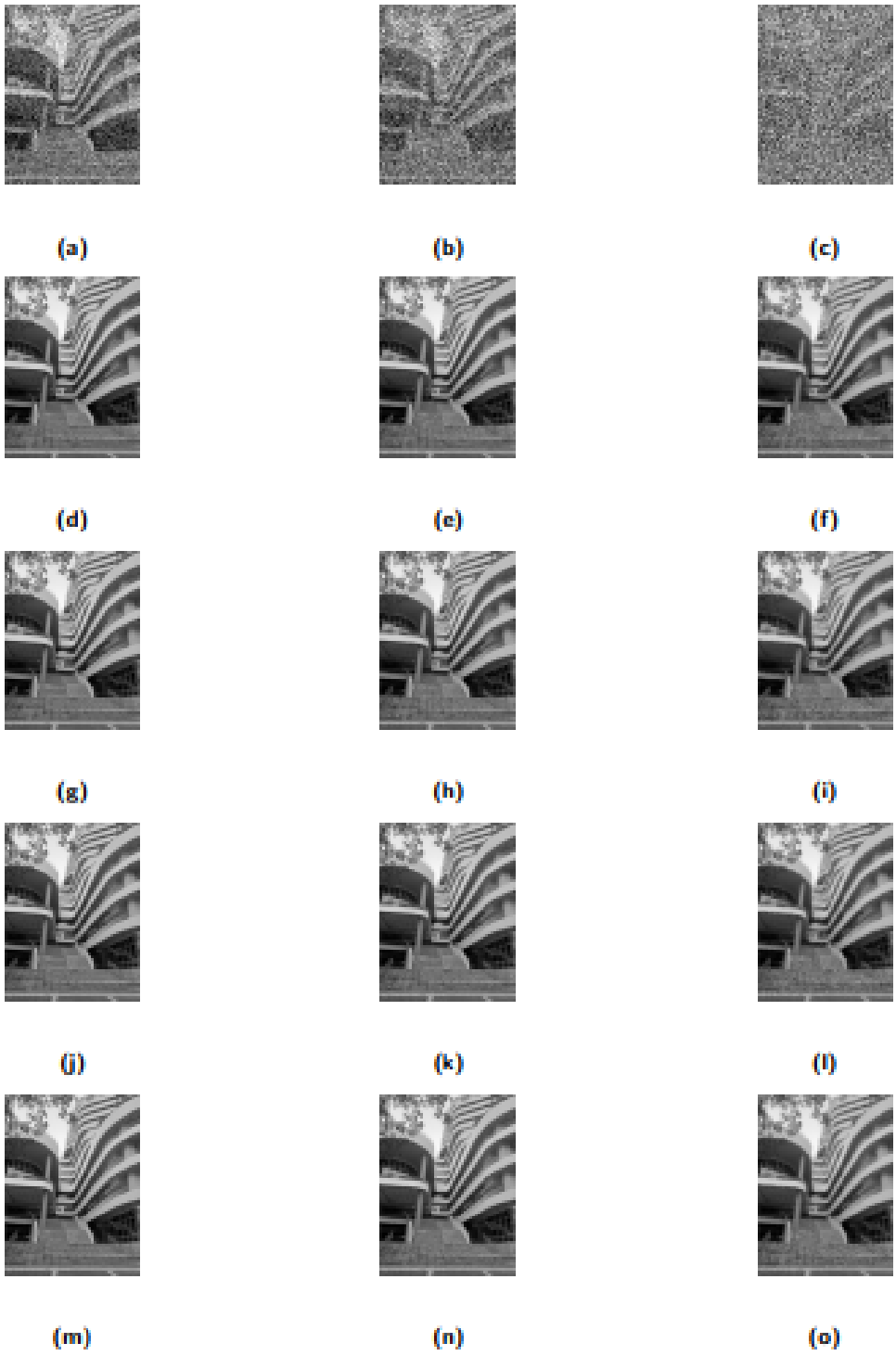

Fig. 4. Building image corrupted by 30, 50, and 80 % salt-and-pepper noise: (a,b,c), the restored images using PPRPDY: (d,e,f), HYB: (g,h,i), HLSDY: (j,k,l), HLSFR (m,n,o)

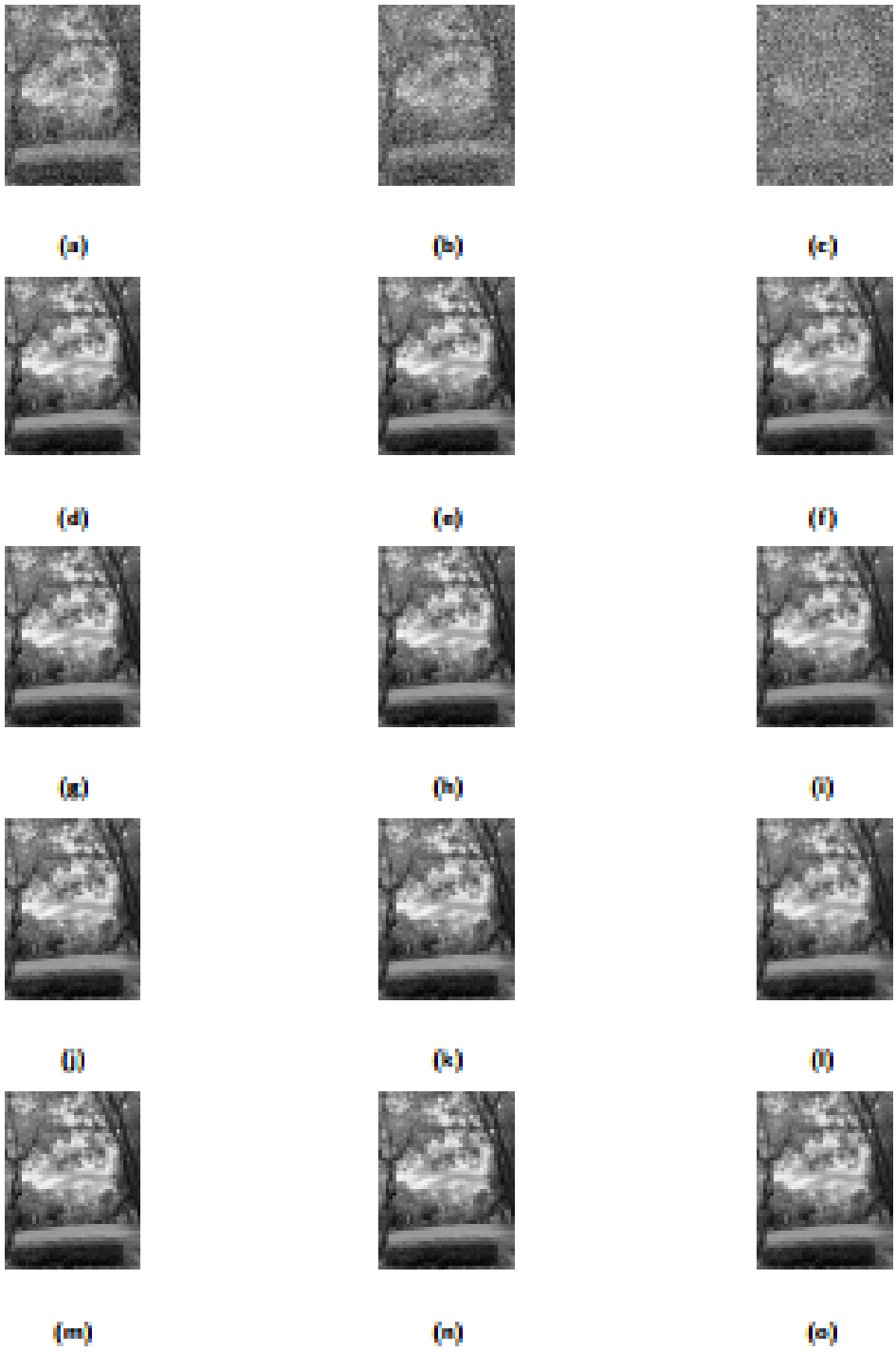

Fig. 5. Forest image corrupted by 30, 50, and 80 % salt-and-pepper noise: (a,b,c), the restored images using PPRPDY: (d,e,f), HYB: (g,h,i), HLSDY: (j,k,l), HLSFR (m,n,o)

Table 2 demonstrates the performance of all the methods on the restored Building and Forest (512×512) grayscale images based on CPU time, Table 3 based on RelErr, and 4 based on PSNR as defined by [26]. Figures 4 and 5 further illustrate the graphical performance of the methods on restoring the images based on 30%, 50%, and 80% noise degrees. Upon analyzing the outcome, it is observed that the proposed PPRPDY restored all the corrupted images with better accuracy compared to the HYB, HLSDY, and HLSFR methods.

6. Conclusion

In this paper, considering Perry's [30] conjugacy condition, we propose a hybrid conjugate gradient method. The method utilizes a revised PRP and DY methods in a convex combination to introduce a new CG parameter. By applying Powell's restart technique [33], we achieve the sufficient descent property and, consequently, a convergence result. Numerical tests on standard nonlinear problems and an image restoration problem indicate the efficiency of the proposed method compared to earlier CG methods. Moreover, robust numerical results are obtained for the proposed scheme when $t = 1$.

Acknowledgments The authors acknowledged the financial support provided by the Petchra Pra Jom Klao PhD Scholarship of King Mongkut's University of Technology Thonburi (KMUTT) with Contract No. 52/2564 and Center of Excellence in Theoretical and Computational Science (TaCS-CoE), KMUTT.

References

[1] N. Andrei, Another hybrid conjugate gradient algorithm for unconstrained optimization, Numer. Algorithms, 47 (2008), 143–156.

[2] N. Andrei, Hybrid conjugate gradient algorithm for unconstrained optimization, J. Optim. Theory Appl., 141 (2009), 249–264.

[3] N. Andrei, An adaptive scaled BFGS method for unconstrained optimization, Numer. Algorithms, 77 (2018), 413–432.

[4] N. Andrei, A Dai-Liao conjugate gradient algorithm with clustering of eigenvalues, Numer. Algorithms, 77 (2018), 1273–1282.

[5] N. Andrei, Nonlinear conjugate gradient methods for unconstrained optimization, Springer, 2020.

[6] S. Babaie-Kafaki, A survey on the Dai-Liao family of nonlinear conjugate gradient methods, RAIRO Oper. Res., 57 (2023), 43–58.

[7] S. Babaie-Kafaki and R. Ghanbari, The Dai-Liao nonlinear conjugate gradient method with optimal parameter choices, European J. Oper. Res., 234 (2014), 625–630.

[8] H. A. Babando, M. Barma, S. Nasiru, and S. Suraj, A dai-liao hybrid prp and dy schemes for unconstrained optimization, International Journal of Development Mathematics (IJDM), 1 (2024), 41–53.

[9] Y.-H. Dai and L.-Z. Liao, New conjugacy conditions and related nonlinear conjugate gradient methods, Appl. Math. Optim., 43 (2001), 87–101.

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

[11] Z. Dai and F. Wen, Another improved Wei-Yao-Liu nonlinear conjugate gradient method with sufficient descent property, Appl. Math. Comput., 218 (2012), 7421–7430.

[12] S. S. Djordjević, New hybrid conjugate gradient method as a convex combination of LS and FR methods, Acta Math. Sci., 39 (2019), 214–228.

[13] E. D. Dolan and J. J. Moré, Benchmarking optimization software with performance profiles, Math. Program., 91 (2002), 201–213.

[14] R. Fletcher, Practical methods of optimization, A Wiley Interscience Publication, 1987.

[15] R. Fletcher and C. M. Reeves, Function minimization by conjugate gradients, Comput. J., 7 (1964), 149–154.

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

[17] A. S. Halilu, A. Majumder, M. Y. Waziri, and K. Ahmed, Signal recovery with convex constrained nonlinear monotone equations through conjugate gradient hybrid approach, Math. Comput. Simulation, 187 (2021), 520–539.

[18] M. R. Hestenes and E. Stiefel, Methods of conjugate gradients for solving linear systems, J. Research Nat. Bur. Standards, 49 (1952), 409–436.

[19] Y. F. Hu and C. Storey, Global convergence result for conjugate gradient methods, J. Optim. Theory Appl., 71 (1991), 399–405.

[20] A. H. Ibrahim, P. Kumam, A. Kamandi, and A. B. Abubakar, An efficient hybrid conjugate gradient method for unconstrained optimization, Optim. Methods Softw., 37 (2022), 1370–1383.

[21] J. Jian, L. Han, and X. Jiang, A hybrid conjugate gradient method with descent property for unconstrained optimization, Appl. Math. Model., 39 (2015):1281–1290.

[22] J. Jian, L. Yang, X. Jiang, P. Liu, and M. Liu, A spectral conjugate gradient method with descent property, Mathematics, 8 (2020), 280.

[23] X. Jiang and J. Jian, Improved Fletcher-Reeves and Dai-Yuan conjugate gradient methods with the strong Wolfe line search, J. Comput. Appl. Math., 348 (2019), 525–534.

[24] J. K. Liu and S. J. Li, New hybrid conjugate gradient method for unconstrained optimization, Appl. Math. Comput., 245 (2014),36–43.

[25] Y. Liu and C. Storey, Efficient generalized conjugate gradient algorithms, part I: Theory, J. Optim. Theory Appl., 69 (1991), 129–137.

[26] M. Malik, I. M. Sulaiman, A. B. Abubakar, G. Ardaneswari, and Sukono, A new family of hybrid three-term conjugate gradient method for unconstrained optimization with application to image restoration and portfolio selection, AIMS Math., 8 (2023), 1–28.

[27] J. Momin and Y. Xin-She, A literature survey of benchmark functions for global optimization problems, Int. J. Math. Mod. Numer. Opt., 4 (2013), 150–194.

[28] P. Mtagulwa and P. Kaelo, A convergent modified HS-DY hybrid conjugate gradient method for unconstrained optimization problems, J. Inf. Optim. Sci., 40 (2019), 97–113.

[29] S. Nasiru, R. O. Mathew, Y. W. Mohammed, S. H. Abubakar, and S. Suraj, A Dai-Liao hybrid conjugate gradient method for unconstrained optimization, Int. J. Indust. Opt., 2 (2021), 69–84.

[30] A. Perry, A modified conjugate gradient algorithm, Oper. Res., 26 (1978), 1073–1078.

[31] E. Polak and G. Ribiére, Note sur la convergence de méthodes de directions conjuguées, USSR Comput. Math. Math. Phys., 9 (1969), 94–112.

[32] B. T. Polyak, A general method for solving extremal problems, Dokl. Akad. Nauk. SSSR, 174 (1967), 33–36.

[33] M. J. D. Powell, Nonconvex minimization calculations and the conjugate gradient method, In Numerical analysis (Dundee, 1983), pages 122–141. Springer, Berlin, 1984.

[34] N. Salihu, H. A. Babando, I. Arzuka, and S. Salihu, A hybrid conjugate gradient method for unconstrained optimization with application, Bangmod International Journal of Mathematical and Computational Science, 9 (2023), 24–44.

[35] N. Salihu, P. Kumam, A. M. Awwal, I. Arzuka, and T. Seangwattana, A Structured Fletcher-Revees Spectral Conjugate Gradient Method for Unconstrained Optimization with Application in Robotic Model, Oper. Res. Forum, 4 (2023), Paper No. 81.

[36] N. Salihu, P. Kumam, A. M. Awwal, I. M. Sulaiman, and T. Seangwattana, The global convergence of spectral RMIL conjugate gradient method for unconstrained optimization with applications to robotic model and image recovery, Plos one, 18 (2023), e0281250.

[37] N. Salihu, P. Kumam, M. Muhammad Yahaya, and T. Seangwattana, A revised liu–storey conjugate gradient parameter for unconstrained optimization problems with applications, Engineering Optimization, (2004), 1–25.

[38] N. Salihu, M. Odekunle, M. Waziri, and A. Halilu. A new hybrid conjugate gradient method based on secant equation for solving large scale unconstrained optimization problems, Iran. J. Opt., 12 (2020), 33–44.

[39] N. Salihu, M. R. Odekunle, A. M. Saleh, and S. Salihu, A Dai-Liao hybrid Hestenes-Stiefel and Fletcher-Revees methods for unconstrained optimization, Int. J. Indust. Opt., 2 (2021), 33–50.

[40] I. M. Sulaiman and M. Mamat, A new conjugate gradient method with descent properties and its application to regression analysis, JNAIAM. J. Numer. Anal. Ind. Appl. Math., 14 (2020), 25–39.

[41] D. Touati-Ahmed and C. Storey, Efficient hybrid conjugate gradient techniques, J. Optim. Theory Appl., 64 (1990), 379–397.

[42] Z. Wei, S. Yao, and L. Liu, The convergence properties of some new conjugate gradient methods, Appl. Math. Comput., 183 (2006), 1341–1350.

[43] M. M. Yahaya, P. Kumam, A. M. Awwal, P. Chaipunya, S. Aji, and S. Salisu, A new generalized quasi-newton algorithm based on structured diagonal Hessian approximation for solving nonlinear least-squares problems with application to 3dof planar robot arm manipulator, IEEE Access, 10 (2022), 10816–10826.

[44] G. Yu, J. Huang, and Y. Zhou, A descent spectral conjugate gradient method for impulse noise removal, Appl. Math. Lett., 23 (2010), 555–560.

[45] G. Yuan, J. Lu, and Z. Wang, The PRP conjugate gradient algorithm with a modified WWP line search and its application in the image restoration problems, Appl. Numer. Math., 152 (2020), 1–11.

[46] L. Zhang, An improved Wei-Yao-Liu nonlinear conjugate gradient method for optimization computation, Appl. Math. Comput., 215 (2009), 2269–2274.

[47] X. Zhou and L. Lu, The global convergence of modified DY conjugate gradient methods under the wolfe line search, J. Chongqing Normal Univ.(Nat. Sci. Ed.), 33 (2016), 6–10.

[48] Z. Zhu, D. Zhang, and S. Wang, Two modified DY conjugate gradient methods for unconstrained optimization problems, Appl. Math. Comput., 373 (2020), 1–10.