Research article

An efficient spectral minimization of the Dai-Yuan method with application to image reconstruction

Nasiru Salihu\textsuperscript{1,2}, Poom Kumam\textsuperscript{1,3,*}, Ibrahim Mohammed Sulaiman\textsuperscript{4,5} and Thidaporn Seangwattana\textsuperscript{6}

\textsuperscript{1} 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
\textsuperscript{2} Department of Mathematics, Faculty of Sciences, Modibbo Adama University, Yola 652105, Nigeria
\textsuperscript{3} 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
\textsuperscript{4} Institute of Strategic Industrial Decision Modelling, School of Quantitative Sciences, Universiti Utara Malaysia, Sintok, Kedah 06010, Malaysia
\textsuperscript{5} Faculty of Education and Arts, Sohar University, Sohar 311, Oman
\textsuperscript{6} Faculty of Science Energy and Environment, King Mongkut’s University of Technology North Bangkok, Rayong Campus (KMUTNB), Rayong 21120, Thailand

\* Correspondence: Email: poom.kum@kmutt.ac.th.

Abstract: In this paper, a spectral Dai and Yuan conjugate gradient (CG) method is proposed based on the generalized conjugacy condition for large-scale unconstrained optimization, in which the spectral parameter is motivated by some interesting theoretical features of quadratic convergence associated with the Newton method. Accordingly, utilizing the strong Wolfe line search to yield the step-length, the search direction of the proposed spectral method is sufficiently descending and converges globally. By applying some standard Euclidean optimization test functions, numerical results reports show the advantage of the method over some modified Dai and Yuan CG schemes in literature. In addition, the method also shows some reliable results, when applied to solve an image reconstruction model.

Keywords: unconstrained optimization; spectral conjugate gradient method; generalized conjugacy condition; Newton direction; global convergence; image reconstruction

Mathematics Subject Classification: 65K05, 90C06, 90C30, 90C47, 90C90
1. Introduction

The unconstrained optimization basically deals with finding the global maximum or minimum of a continuously differentiable function \( f : \mathbb{R}^n \rightarrow \mathbb{R} \). The general form of an unconstrained optimization problem is as follows:

\[
\min f(x), \ x \in \mathbb{R}^n,
\]

where the gradient \( g(x) = \nabla f(x) \). Solving the complexity of this problem (1.1) would require the use of some symbolic aided algebra system and an effective numerical method that would be able to perform the necessary computations, plot the numerical results and manipulate the mathematical expression in an analytical form. One of the widely and most preferred methods considered for solving (1.1) is the nonlinear conjugate gradient (CG) method because of its robustness and ability to deal with large-scale problems [1,2]. The simplicity and efficiency of the CG algorithm has further motivated its application to numerous real-life problems including the feed forward training of neural networks [3], robotic motion control [4], signal recovery [5], regression analysis [1, 6], image deblurring [7, 8] and portfolio selection [9] among others.

Starting with an initial guess of \( x_0 \in \mathbb{R}^n \), the CG method generates a sequence of iterates \( \{x_k\} \) via the following formula:

\[
x_{k+1} = x_k + \alpha_k d_k, \ k = 0, 1, 2, \ldots,
\]

where \( \alpha_k \) represents the step size that is computed along the search direction \( d_k \) [10]. Usually, a line search procedure is required to obtain the step size \( (\alpha_k) \) which could either be an exact or inexact procedure. While the exact line search yields the exact minimizer for a given problem, the inexact scheme computes \( \alpha_k \) such that it satisfies the Wolfe conditions at each iteration. The most commonly used Wolfe conditions includes the standard Wolfe condition that requires \( \alpha_k \) to satisfy

\[
f(x_k + \alpha_k d_k) \leq f(x_k) + \delta \alpha_k g_k^T d_k,
\]

\[
g(x_k + \alpha_k d_k)^T d_k \geq \sigma g_k^T d_k,
\]

and the strong Wolfe condition is computed such that \( \alpha_k \) satisfies (1.3) and

\[
|g(x_k + \alpha_k d_k)^T d_k| \leq -\sigma g_k^T d_k,
\]

where \( 0 < \delta < \frac{1}{2}, \delta < \sigma < 1 \).

Another important component of the CG method is the search direction \( d_k \). This component plays an important role in the performance and convergence analysis of any CG method. Also, the search direction distinguishes the classes of different CG algorithms including the spectral and three-term CG methods. The search direction \( d_k \) for the classical CG method is computed as follows:

\[
d_0 = -g_0, \quad d_{k+1} = -g_{k+1} + \beta_{k} d_k, \quad \forall k \geq 1,
\]

where the coefficient \( \beta_k \) characterizes the different CG formulas. Selection of the CG choice parameter and search direction is always crucial in the study of the unconstrained optimization because these
two components are responsible for the numerical performance and theoretical analysis of any CG method [11].

For any optimization method to fulfill the general optimization criteria and be able to use the line search procedure, it is required to possess the following descent property:

\[ d_k^T g_k < 0 \quad \forall k. \]  

(1.7)

Furthermore, a CG method is said to be sufficiently descending if there exists some constant \( c > 0 \) such that the following condition holds:

\[ d_k^T g_k \leq -c \| g_k \|^2 \quad \forall k. \]  

(1.8)

This condition (1.8) plays an important role in the convergence analysis of any nonlinear CG algorithm, which is always preferred over (1.7). Some of the earliest and known CG coefficients that possess this condition (1.8) include those developed by Fletcher and Reeves (FR) [12], Dai and Yuan (DY) [13] and Fletcher (conjugate descent (CD)) [14], with formulas as follows:

\[
\beta_{FR}^k = \frac{\| g_{k+1} \|^2}{\| g_k \|^2}, \quad \beta_{DY}^k = \frac{\| g_{k+1} \|^2}{d_k^T y_k}, \quad \beta_{CD}^k = -\frac{\| g_{k+1} \|^2}{d_k^T g_k},
\]  

(1.9)

where \( y_k = g_{k+1} - g_k \) and \( \| \cdot \| \) denotes the \( \ell_2 \) norm. This class of CG methods is characterized by simplicity and less memory requirements under different line search procedures. However, their numerical performance in practical computations is often affected by jamming phenomena. For instance, if any of the above CG algorithm generates a tiny step size from \( x_k \) to \( x_{k+1} \) and poor search direction and as a result if restart is not performed along the negative direction, then, it is likely that the subsequent step size and direction will also have poor performance [15].

Due to the challenges reported by the above category of CG algorithms, several studies have shown that the methods possess nice convergence properties (see [11, 16, 17]). To address the issue related to the computational performance, several studies involved constructing new CG formulas by either combining the methods in (1.9) with other efficient formulas or introducing new terms to the set of methods in (1.9) to improve the computational efficiency and their general structure [18].

In this study, we are interested in modifying the classical DY CG formula for solving optimization and image restoration problems. The proposed method introduces a new spectral parameter \( \theta_k \) to scale the search direction \( d_k \) in (1.6) such that it satisfies the Newton search direction and the well-known D–L [19] conjugacy condition in Section 2. Section 3 demonstrates how the proposed method satisfies the descent condition and further proves the global convergence under suitable conditions. The preliminary computational results on a set of optimization functions are analyzed in Section 4 while Section 5 reports results related to real-world application problem. The last section summarizes the whole idea related to the study and presents a general conclusion.

2. Formulation of the spectral algorithm and motivation

The spectral CG methods have been proposed to improve the general structure of CG methods. The methods are based on the works of Raydan [20], Barzilai and Borwein [21] and Birgin and Martínez [22]. For some recent results on spectral CG methods, see [8,23–31]. Similarly, the standard conjugacy condition given by

\[ y_k^T d_{k+1} = 0 \]  

(2.1)
is crucial in the convergence analysis of these CG methods. The CG methods proposed with this structure (2.1) largely depend on exact line criteria for the choice of step size, $\alpha_k$. However, this requirement is computationally expensive for large-scale problems. Therefore, denoting $s_k = x_{k+1} - x_k$, the selection of $\alpha_k$ uses its generalized form called the D–L conjugacy condition introduced in [19],

$$y_k^T d_{k+1} = -t g_{k+1}^T s_k, \quad t > 0,$$

(2.2)
is usually preferable in the design of some nice CG algorithms. The parameter $t$, called the D-L parameter is capable of ensuring better convergence. Therefore, it must be carefully chosen.

To address some weaknesses associated with the DY method, its modification has been investigated recently. For instance, utilizing the strong Wolfe line search, Jiang and Jian [32] proposed the improved DY [13] CG parameter with the following structure:

$$\beta_{k}^{DY} = r_k \frac{\|g_{k+1}\|^2}{d_k^T y_k},$$

(2.3)

where $r_k = \frac{|g_k^T d_k|}{g_k^T g_k}$. Motivated by this idea, Jian et al. [33] introduced a spectral parameter that yields the descent direction with

$$\theta_{k}^{JY JLL} = 1 + \frac{|g_k^T d_k|}{-g_k^T g_k}.$$  

(2.4)

Since $\theta_{k}^{JY JLL} \geq 1$ it is obvious that $d_k$ is a descent direction. Following the idea in [34], they proposed its conjugate parameter as follows:

$$\beta_{k}^{JY JLL} = \frac{\|g_{k+1}\|^2 - \frac{(g_{k+1}^T d_k)^2}{\|g_k\|^2}}{\max\{\|g_k\|^2, d_k^T y_k\}}.$$  

(2.5)

However, the search direction including the method in (2.5) only satisfies the descent condition, (1.7) and it converges globally based on two cases of $\max\{\|g_k\|^2, d_k^T y_k\}$ with the standard Wolfe criteria respectively. According to this formulation, two other modified DY CG methods were proposed in [35] with the following forms:

$$\beta_{k}^{(1)} = \frac{\|g_{k+1}\|^2 - \frac{(g_{k+1}^T g_k)^2}{\|g_k\|^2}}{\|g_k\|^2},$$

(2.6)

$$\beta_{k}^{(2)} = \frac{\|g_{k+1}\|^2 - \frac{\|g_{k+1}\|^2}{\|g_k\|^2} g_{k+1}^T g_k}{\mu d_k^T y_k}.$$  

(2.7)

Inspired by these modifications, Zhu et al. [36] suggested another DY modification namely, the DDY1 scheme which has the form of

$$\beta_{k}^{DDY1} = \begin{cases} \frac{\|g_{k+1}\|^2 - \frac{(g_{k+1}^T g_k)^2}{\|g_k\|^2}}{\|g_k\|^2}, & g_{k+1}^T g_k \geq 0, \\ \frac{d_k^T y_k}{\|g_{k+1}\|^2 + \frac{(g_{k+1}^T g_k)^2}{\|g_k\|^2} g_{k+1}^T g_k}, & g_{k+1}^T g_k < 0. \end{cases}$$  

(2.8)
Nonetheless, the method has similar theoretical characteristics with the scheme described by (2.5). Thus, in order to improve the general structure and establish sufficient descent, as given by (1.8) for the DY method, we scale the search direction $d_k$ in (1.6) with a spectral parameter $\theta_k$ such that it satisfies the well-known D–L conjugacy condition (2.2), as well as the Newton search direction in the following form

$$d_{k+1} = -\theta_k g_{k+1} + \beta_k d_k, \quad \forall k \geq 1. \quad (2.9)$$

Pre-multiplying by $y_k^T$, we obtain

$$y_k^T d_{k+1} = -\theta_k y_k^T g_{k+1} + \beta_k^{DY} y_k^T d_k.$$  

Using the above relation with (2.2) gives

$$-t_s^T g_{k+1} = -\theta_k y_k^T g_{k+1} + \beta_k^{DY} y_k^T d_k.$$  

Re-arranging implies that

$$\theta_k^{SOL} = \frac{t_s^T g_{k+1}}{y_k^T g_{k+1}} + \beta_k^{DY} \frac{y_k^T d_k}{y_k^T g_{k+1}}. \quad (2.10)$$

Similarly, in some neighborhood of the minimizer, if the current point $x_{k+1}$ is close enough to a local minimizer and the objective function behaves like a quadratic function, then the optimal search direction to follow is the Newton direction:

$$d_{k+1} = -\nabla^2 f(x_{k+1})^{-1} g_{k+1}, \quad (2.11)$$

where $\nabla^2 f(x_{k+1})$ is the Hessian matrix of $f$. Thus, the Newton method requires the Hessian matrix, i.e., the second derivative information to update (2.11), which provides a nice convergence rate. Motivated by its quadratic convergence property, we assume that $\nabla^2 f(x_{k+1})$ exists at every iteration and satisfies the conditions of a suitable secant equation; for instance,

$$\nabla^2 f(x_{k+1}) s_k = y_k. \quad (2.12)$$

Now, equating (2.9) with (2.11) gives

$$-\nabla^2 f(x_{k+1})^{-1} g_{k+1} = -\theta_k g_{k+1} + \beta_k^{DY} d_k.$$  

Pre-multiplying by $y_k^T$ and using the secant equation (2.12), we get

$$-\nabla^2 f(x_{k+1})^{-1} g_{k+1} = -\theta_k g_{k+1} + \beta_k^{DY} d_k$$

$$-s_k^T g_{k+1} = -\theta_k s_k^T \nabla^2 f(x_{k+1}) g_{k+1} + \beta_k^{DY} s_k^T \nabla^2 f(x_{k+1}) d_k$$

$$-s_k^T g_{k+1} = -\theta_k y_k^T g_{k+1} + \beta_k^{DY} y_k^T d_k.$$  

After some simple simplification, we obtain

$$\theta_k^{SNM} = \frac{s_k^T g_{k+1}}{y_k^T g_{k+1}} + \beta_k^{DY} \frac{y_k^T d_k}{y_k^T g_{k+1}}. \quad (2.13)$$
Remark 2.1. Observe that, if \( t = 1 \), then the parameter \( \theta_k^{SDL} = \theta_k^{SNM} \), which implies that, the spectral search direction \( d_{k+1} \) does not only satisfy the generalized D-L conjugacy condition it also takes advantage of the nice convergence property of the Newton direction. However, for a sufficient descent condition to hold, we always select \( t > 1 \) in this paper.

To achieve the sufficient descent of the search direction described by (2.9), we suggest the following modification of (2.10) as follows:

\[
\theta_k^{SNM1} = \frac{t_s g_{k+1}^T}{\text{max}\{|y_k g_{k+1}|, d_k^T y_k\}} + \frac{y_k d_k}{\text{max}\{|y_k g_{k+1}|, d_k^T y_k\}}.
\]  

(2.14)

Thus, if \( |y_k g_{k+1}| > d_k^T y_k \), then (2.14) reduces (2.10). Hence, for \( d_k^T y_k > |y_k g_{k+1}| \), we obtain another spectral parameter:

\[
\theta_k^{SNM2} = \frac{t_s g_{k+1}}{d_k^T y_k} + \frac{y_k d_k}{d_k^T y_k}.
\]  

(2.15)

Combining (2.10) and (2.15) implies that the optimal \( \theta_k \) can be computed as follows:

\[
\theta_k^{NSDY} = \begin{cases} 
\theta_k^{SDL}, & \text{if } |y_k g_{k+1}| > d_k^T y_k, \\
\theta_k^{SNM2}, & \text{otherwise}.
\end{cases}
\]  

(2.16)

Now, we describe the new spectral DY (NSDY) algorithm as follows:

**Algorithm 1: NSDY**

1. **Step 1:** Apply \( x_0 \in \mathbb{R}^n \) and the parameters \( 0 < \delta \leq \sigma < 1 \). Set \( d_0 = -g_0, \alpha_0 = 1 \).
2. **Step 2:** Verify the convergence: If \( \|g_k\| \leq \epsilon \), then stop. Otherwise, proceed to the next step.
3. **Step 3:** Select \( \alpha_k > 0 \) such that (1.3) and (1.5) are satisfied.
4. **Step 4:** Compute \( \theta_k^{NSDY} \) using (2.16).
5. **Step 5:** Compute \( \beta_{k}^{DY} \) from (1.9) and \( d_k \) using (2.9).
6. **Step 6:** Update the next iteration from Step 2.

**Theorem 2.2.** Let \( \theta_k \) be given by (2.16) with \( 0 < \alpha_k \leq \sigma < 1 \). Suppose that the NSDY algorithm generates sequences \( \{g_k\} \) and \( \{d_k\} \), where \( \alpha_k \) is determined by strong Wolfe rules (1.3)–(1.5); then, there exists a constant \( \rho > 0 \) for which the condition

\[
d_{k+1}^T g_{k+1} \leq -\rho \|g_{k+1}\|^2, \quad \forall k \geq 0.
\]  

(2.17)

holds.

The criterion (2.17) is called the sufficient descent condition, and it ensure that the direction described by (2.9) is indeed sufficient for the minimizer.

**Proof.** The proof is achieved by mathematical induction. Initially, for \( k = 0 \) it follows easily that \( g_0^T d_0 = -\|g_0\|^2 \leq -\rho \|g_0\|^2 \). Suppose that (2.17) holds for index \( k \), that is, \( d_k^T g_k \leq -\rho \|g_k\|^2 \). We now show for \( k + 1 \). Now, according to the strong Wolfe criterion (1.3), it holds that

\[
|g_{k+1}^T d_k| \leq -\sigma_g g_k^T d_k,
\]
i.e.,
\[ d_k^T y_k = d_k^T g_{k+1} - d_k^T g_k \geq -(1 - \sigma)d_k^T g_k. \] (2.18)

Then
\[ \frac{|g_{k+1}^T d_k|}{|d_k^T y_k|} \leq \frac{\sigma}{1 - \sigma}. \] (2.19)

Pre-multiplying (2.9) by \( g_k^T \), we have
\[ g_k^T d_{k+1} = -\theta_k \| g_{k+1} \| + \beta_k^{DY} g_k^T d_k. \] (2.20)

Using (2.14) with \( s_k = \alpha_k d_k \) gives
\[
\theta_k^{S^N M1} = \frac{\max\{|y_k^T g_{k+1}|, d_k^T y_k\}}{|d_k^T y_k|} + \frac{\beta_k^{DY} y_k^T d_k}{\max\{|y_k^T g_{k+1}|, d_k^T y_k\}} \\
\leq \frac{\max\{|y_k^T g_{k+1}|, d_k^T y_k\}}{|d_k^T y_k|} + \frac{\beta_k^{DY} y_k^T d_k}{\max\{|y_k^T g_{k+1}|, d_k^T y_k\}} \\
= \frac{\max\{|y_k^T g_{k+1}|, d_k^T y_k\}}{|d_k^T y_k|} + \frac{\theta_{k+1} ||g_{k+1}||^2}{|d_k^T y_k|} \\
\leq \frac{\max\{|y_k^T g_{k+1}|, d_k^T y_k\}}{|d_k^T y_k|} + \frac{\theta_{k+1} ||g_{k+1}||^2}{|d_k^T y_k|} + \frac{\sigma \theta_{k+1} ||g_{k+1}||^2}{1 - \sigma} \| g_{k+1} \|^2. \] (2.21)

Obtaining \( \beta_k^{DY} \) from (1.9) and applying it with (2.21) and (2.20), we get
\[
g_k^T d_{k+1} \leq -\left( \frac{\max\{|y_k^T g_{k+1}|, d_k^T y_k\}}{|d_k^T y_k|} + \frac{\beta_k^{DY} y_k^T d_k}{|d_k^T y_k|} \right) ||g_{k+1}||^2 + \beta_k^{DY} g_k^T d_k. \\
= -\left( \frac{\max\{|y_k^T g_{k+1}|, d_k^T y_k\}}{|d_k^T y_k|} + \frac{||g_{k+1}||^2}{|d_k^T y_k|} \right) ||g_{k+1}||^2 + \frac{||g_{k+1}||^2}{|d_k^T y_k|} g_k^T d_k. \\
\leq -\left( \frac{\max\{|y_k^T g_{k+1}|, d_k^T y_k\}}{|d_k^T y_k|} + \frac{||g_{k+1}||^2}{|d_k^T y_k|} \right) ||g_{k+1}||^2 + \frac{||g_{k+1}||^2}{|d_k^T y_k|} ||g_{k+1}||^2 \\
\leq -\left( \frac{\sigma \theta_{k+1}}{1 - \sigma} \frac{||g_{k+1}||^2}{|d_k^T y_k|} \right) ||g_{k+1}||^2 + \frac{\sigma}{1 - \sigma} ||g_{k+1}||^2 \] (2.22)

where the second to the last and last inequalities follow (2.19) respectively. Now, from (2.18), we get
\[
\frac{1}{d_k^T y_k} \leq \frac{1}{-(1 - \sigma)d_k^T g_k}. 
\]

Combining this with (2.17), we get
\[
\frac{1}{d_k^T y_k} \leq \frac{1}{(1 - \sigma)\|g_k\|^2}. \] (2.23)

The result by the last inequality follows from the induction hypothesis. Finally, using (2.22) and (2.23), we conclude that
\[
g_{k+1}^T d_{k+1} \leq -\left( \frac{\sigma \theta_{k+1}}{1 - \sigma} + \frac{||g_{k+1}||^2}{(1 - \sigma)\|g_k\|^2} - \frac{\sigma}{(1 - \sigma)} \right) ||g_{k+1}||^2. 
\]
\[ \leq - \left( \frac{\sigma t \alpha_k}{1 - \sigma} - \frac{\sigma}{1 - \sigma} \right) \|g_{k+1}\|^2. \]

Since \[ \frac{\|g_{k+1}\|^2}{(1 - \sigma)t\|g_k\|^2} > 0, \]
we have
\[ g_{k+1}^T d_{k+1} \leq - \left( \frac{\sigma t \alpha_k}{1 - \sigma} - \frac{\sigma}{1 - \sigma} \right) \|g_{k+1}\|^2. \]

Denoting \[ \rho = \left( \frac{\sigma t \alpha_k}{1 - \sigma} - \frac{\sigma}{1 - \sigma} \right), \]
with \( t \geq 1.2, \alpha_k \in (0, 0.9] \) and \( \sigma \in (0, 0.9] \), the required result is achieved. \( \square \)

3. Convergence analysis

The convergence analysis requires the following presumptions.

**Assumption 3.1.**

1. Given an initial point \( x_0 \), the function \( f(x) \) is bounded below on the level set \( \eta = \{ x \in \mathbb{R}^n : f(x_0) \geq f(x) \} \).
2. Denote \( \Gamma \) as some neighborhood of \( \eta \), and the function \( f \) is smooth with its gradient that is Lipschitz-continuous and satisfying
\[ \|g(x) - g(y)\| \leq L\|x - y\|, \quad \forall x, y \in \Gamma, \quad L > 0. \] (3.1)

Note that these assumptions, imply that
\[ \|g(x)\| \leq \gamma, \quad \forall x \in \eta, \quad \gamma > 0, \] (3.2)
\[ \|x - y\| \leq b, \quad \forall x, y \in \eta, \quad b > 0. \] (3.3)

The following lemma was taken from [17] and is very crucial in our analysis.

**Lemma 3.2.** Suppose that \( \{x_k\} \) and \( \{d_k\} \) are sequences generated by the NSDY algorithm, where the spectral search direction \( d_k \) is descending and \( \alpha_k \) satisfies the strong Wolfe conditions; then,
\[ \sum_{k=0}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < +\infty. \] (3.4)

**Theorem 3.3.** If Assumption (3.1) holds and the sequence of iterations \( \{x_k\} \) is produced by the NSDY algorithm, then
\[ \liminf_{k \to \infty} \|g_k\| = 0. \] (3.5)

**Proof.** If (3.5) does not hold, then there exists some constant \( r > 0 \) so that
\[ \|g_k\| > r. \] (3.6)

**Claim:** The search direction defined by (2.9) is bounded, i.e., there exists a constant \( P > 0 \) such that
\[ \|d_{k+1}\| \leq P, \quad \forall k \geq 0. \] (3.7)
To proceed with the proof of this claim by induction, we need to show that \(|\beta_k^D|\) and \(|\theta_k|\) are bounded below. According to (2.18), it yields

\[
\frac{1}{d_k^T y_k} \leq \frac{1}{(1 - \sigma)d_k^T g_k}.
\]

Combining this with (2.17), we get

\[
\frac{1}{d_k^T y_k} \leq \frac{1}{(1 - \sigma)\rho \|g_k\|^2}.
\] (3.8)

Therefore, obtaining \(\beta_k^D\) from (1.9) and applying it with (3.2), (3.6) and (3.8), we have

\[
|\beta_k^D| \leq \frac{\|g_{k+1}\|^2}{d_k^T y_k} \leq \frac{\gamma^2}{\rho(1 - \sigma)\rho^2} = E.
\] (3.9)

Next, from (2.14), (3.2), (3.8) and (3.9), we obtain

\[
|\theta_k| \leq \left| \frac{ts_k^T g_{k+1}}{\max\{\|y_k^T g_{k+1}\|, \|d_k^T y_k\|\}} + \beta_k^D \frac{y_k^T d_k}{\max\{\|y_k^T g_{k+1}\|, \|d_k^T y_k\|\}} \right|
\leq \frac{ts_k^T g_{k+1}}{d_k^T y_k} + |\beta_k^D|
\leq \frac{ts_k\|s_k\|\|g_{k+1}\|}{|d_k^T y_k|} + E
\leq \frac{iby}{\rho(1 - \sigma)\rho^2} + E = M.
\] (3.10)

Now, for \(k = 0\), we get that \(d_1 = -\theta_1 g_1 + \beta_1 d_0\) from (2.9), which implies that \(d_1 = -\theta_1 g_1 - \beta_1 g_0\), since \(d_0 = -g_0\); this yields

\[
\|d_1\| \leq |\theta_1|\|g_1\| + |\beta_1|\|g_0\|
\leq \gamma + E\gamma = \gamma^*,
\]

that is, the claim (3.7) holds for \(k = 0\). Next we assume that the claim (3.7) is true for \(k\), that is, \(\|d_k\| \leq P\). To show that it is true for \(k + 1\), consider the search direction described by (2.9).

\[
d_{k+1} = -\beta_k g_{k+1} + \beta_1 d_k.
\]

Thus, using (3.2), (3.9) and (3.10), we obtain

\[
\|d_{k+1}\| \leq |\theta_k|\|g_{k+1}\| + |\beta_k|\|d_k\|
\leq M\gamma + EP;
\]

therefore, the claim also holds for \(k + 1\). Now since (3.7) holds for all values of \(k\), then we have

\[
\frac{1}{\|d_k\|} \geq \frac{1}{P}, \quad P > 0.
\] (3.11)
From the above inequality, it follows that
\[ \sum_{k=0}^{\infty} \frac{1}{\|d_k\|^2} = +\infty. \] (3.12)

Accordingly, from (2.17), (3.4) and (3.6), it follows that
\[ \rho^2 r^4 \sum_{k=0}^{\infty} \frac{1}{\|d_k\|^2} \leq \sum_{k=0}^{\infty} \rho^2 \|g_k\|^4 \leq \sum_{k=0}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < +\infty \] (3.13)

Therefore, this implies that
\[ \sum_{k=0}^{\infty} \frac{1}{\|d_k\|^2} < +\infty. \] (3.14)

It is obvious that, (3.12) and (3.14) cannot hold concurrently. Thus, (3.5) must hold. \qed

4. Numerical results

In this part, we present a numerical comparison of the NSDY method versus the classical DY method and three other modifications of the DY method. The evaluations are based on a number of test functions considered from [37,38]. For the computation, we consider the dimension ranging from 2 to 100,000 as presented in Tables 1 and 2. To analyze the results further, the following specifications are considered for the algorithms:

- JYJLL algorithm of Jian et al. [33] that uses (2.4) and (2.5) for the search direction described by (2.9) and \( \delta = 0.01 \) and \( \sigma = 0.1 \) in (1.3) and (1.4) respectively.
- IDY algorithm of Jiang and Jian [32] that uses (2.3) for the search direction described by (1.6) and \( \delta = 0.01 \) and \( \sigma = 0.1 \) in (1.3) and (1.4) respectively.
- DDY1 algorithm of Zhu et al. [36] that uses (2.8) for the search direction described by (1.6) with \( \mu_1 = 0.4 \) and \( \delta = 0.01 \) and \( \sigma = 0.1 \) in (1.3) and (1.4) respectively.
- DY algorithm of Dai and Yuan [13] with \( \beta_k^{DY} = \frac{\|g_k\|^2}{d_k^T y_k} \) for the search direction described by (1.6) and \( \delta = 0.01 \) and \( \sigma = 0.1 \) in (1.3) and (1.4) respectively.

The MATLAB R2022a codes for the numerical experiment were run on a Dell Core i7 laptop with 16 GB RAM and a 2.90 GHz CPU. We set \( \delta = 10^{-3} \) and \( \sigma = 0.9 \) in (1.3) and (1.5) for the NSDY method and \( \|g_k\| \leq 10^{-6} \) as the stopping condition for all schemes. We also use the symbol \(* * *\) to indicate a situation in which the stopping condition is not met. The results from the computational experiments based on number of iterations (NI), number of function evaluations (FE), and CPU time (ET) is presented at the following link https://acrobat.adobe.com/link/review?uri=urn:aaid:scds:US:ed485b92-05e2-40f3-a1f2-6e159858c515.

We will employ the performance profiling technique to interpret and discuss the performance of the methods examined. Let \( P \) be the collection of \( n_p \) test problems and \( S \) be the set of \( n_s \) solvers used in the comparison. The performance profiles thus constitute the performance metric for a problem \( p \in P \) and a solver \( s \in S \), which denote either NI, FE or ET relative to the other solvers \( s \in S \) on a set of
problems \( p \in P \). Then, Dolan and Moré [39] described the measure of performance ratio to compare and evaluate the performance:

\[
    r_{p,s} = \frac{f_{p,s}}{\min\{f_{p,s} : s \in S \text{ and } p \in P\}}.
\]

Thus, the best method is indicated by the performance profile plot at the top curve. The experiments can therefore be interpreted graphically by using Figures 1–3 based on numerical performance.

The interpretation of Figure 1 for the value \( \tau \) chosen within the \( 0 < \tau < 0.5 \) interval shows that the NSDY method solved the test problems and won in 98%/38% to be the best, followed by the JYJLL method with 95%/45%, whereas the DY, IDY and DDY1 methods solved and won in 96%/25%, 90%/37% and 55%/20% of the given problems respectively. Accordingly, if we increase the \( \tau \) to an interval \( \tau \geq 1 \), the NSDY method is the best with 68% accuracy, whereas the JYJLL, IDY, DY and DDY1 methods have 66%, 60%, 55% and 40% accuracy, respectively. Similarly, the comparison among the five schemes based on interpretation of Figure 2 shows that the NSDY, DY, JYJLL, IDY and DDY1 methods solved the problems and won in 98%/12%, 96%/28%, 94%/54%, 90%/30% and 57%/14% respectively for the value \( \tau \) chosen within \( 0 < \tau < 0.5 \). However, increasing the \( \tau \) value to the interval \( \tau \geq 6 \) reveals that the NSDY method wins when solving 97% of the test problems compared to JYJLL, IDY, DY and DDY1 methods with 93%, 88%, 92% and 52% of problems solved to the best respectively. Finally, Figure 3 also shows that for the value \( \tau \) chosen within \( 0 < \tau < 0.5 \), NSDY, DY, JYJLL, IDY and DDY1 methods solved and won in 98%/33%, 96%/08%, 94%/38%, 90%/22% and 58%/05% problems, respectively. Alternatively, taking the value of \( \tau \) in the interval \( \tau \geq 6 \) reveals that the NSDY method wins when solving 97% of the test problems compared to the JYJLL, IDY, DY and DDY1 methods with 93%, 88%, 92%, and 52% of problems solved to the best, respectively. Therefore, interpretations of Figures 1–3 indicate that the NSDY method is more preferable than other CG methods.

![Figure 1. NI performance profile for the methods.](image-url)
Figure 2. FE performance profile for the methods.

Figure 3. ET performance profile for the methods.
<table>
<thead>
<tr>
<th>NO</th>
<th>Problem</th>
<th>Dim</th>
<th>NO</th>
<th>Problem</th>
<th>Dim</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>EXTENDED PENALTY</td>
<td>100</td>
<td>43</td>
<td>DIAGONAL 1</td>
<td>100</td>
</tr>
<tr>
<td>2</td>
<td>EXTENDED PENALTY</td>
<td>10000</td>
<td>44</td>
<td>HAGER</td>
<td>500</td>
</tr>
<tr>
<td>3</td>
<td>EXTENDED PENALTY</td>
<td>20000</td>
<td>45</td>
<td>HAGER</td>
<td>300</td>
</tr>
<tr>
<td>4</td>
<td>EXTENDED MARATOS</td>
<td>2</td>
<td>46</td>
<td>HAGER</td>
<td>5000</td>
</tr>
<tr>
<td>5</td>
<td>DIAGONAL 5</td>
<td>10</td>
<td>47</td>
<td>HAGER</td>
<td>1000</td>
</tr>
<tr>
<td>6</td>
<td>DIAGONAL 5</td>
<td>50000</td>
<td>48</td>
<td>ZIRILLI</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>DIAGONAL 5</td>
<td>100000</td>
<td>49</td>
<td>RAYDAN 1</td>
<td>100</td>
</tr>
<tr>
<td>8</td>
<td>TRECANNI</td>
<td>2</td>
<td>50</td>
<td>RAYDAN 1</td>
<td>10</td>
</tr>
<tr>
<td>9</td>
<td>TRECANNI</td>
<td>2</td>
<td>51</td>
<td>RAYDAN 1</td>
<td>10</td>
</tr>
<tr>
<td>10</td>
<td>QUADRATIC PENALTY 1</td>
<td>4</td>
<td>52</td>
<td>RAYDAN 1</td>
<td>50</td>
</tr>
<tr>
<td>11</td>
<td>QUADRATIC PENALTY 1</td>
<td>100</td>
<td>53</td>
<td>RAYDAN 2</td>
<td>10000</td>
</tr>
<tr>
<td>12</td>
<td>QUADRATIC PENALTY 1</td>
<td>1000</td>
<td>54</td>
<td>RAYDAN 2</td>
<td>50000</td>
</tr>
<tr>
<td>13</td>
<td>QUADRATIC PENALTY 2</td>
<td>500</td>
<td>55</td>
<td>RAYDAN 2</td>
<td>100000</td>
</tr>
<tr>
<td>14</td>
<td>QUADRATIC PENALTY 2</td>
<td>100</td>
<td>56</td>
<td>FLETCHCER</td>
<td>10</td>
</tr>
<tr>
<td>15</td>
<td>QUADRATIC PENALTY 2</td>
<td>10</td>
<td>57</td>
<td>FLETCHCER</td>
<td>1000</td>
</tr>
<tr>
<td>16</td>
<td>QUADRATIC FUNCTION 1</td>
<td>100</td>
<td>58</td>
<td>FLETCHCER</td>
<td>50000</td>
</tr>
<tr>
<td>17</td>
<td>QUADRATIC FUNCTION 1</td>
<td>10</td>
<td>59</td>
<td>DIAGONAL 3</td>
<td>2</td>
</tr>
<tr>
<td>18</td>
<td>QUADRATIC FUNCTION 1</td>
<td>10</td>
<td>60</td>
<td>DIAGONAL 3</td>
<td>10</td>
</tr>
<tr>
<td>19</td>
<td>QUADRATIC FUNCTION 2</td>
<td>50</td>
<td>61</td>
<td>EXTENDED DENSCHN B</td>
<td>100</td>
</tr>
<tr>
<td>20</td>
<td>QUADRATIC FUNCTION 2</td>
<td>1000</td>
<td>62</td>
<td>EXTENDED DENSCHN B</td>
<td>5000</td>
</tr>
<tr>
<td>21</td>
<td>QUADRATIC FUNCTION 2</td>
<td>5000</td>
<td>63</td>
<td>EXTENDED DENSCHN B</td>
<td>10000</td>
</tr>
<tr>
<td>22</td>
<td>POWER</td>
<td>2</td>
<td>64</td>
<td>DIAGONAL 6</td>
<td>10000</td>
</tr>
<tr>
<td>23</td>
<td>POWER</td>
<td>2</td>
<td>65</td>
<td>DIAGONAL 6</td>
<td>5000</td>
</tr>
<tr>
<td>24</td>
<td>ZETTL</td>
<td>2</td>
<td>66</td>
<td>DIAGONAL 6</td>
<td>10000</td>
</tr>
<tr>
<td>25</td>
<td>DIAGONAL 2</td>
<td>1000</td>
<td>67</td>
<td>DIAGONAL 6</td>
<td>50000</td>
</tr>
<tr>
<td>26</td>
<td>DIAGONAL 2</td>
<td>10000</td>
<td>68</td>
<td>DIAGONAL 4</td>
<td>100</td>
</tr>
<tr>
<td>27</td>
<td>DIAGONAL 2</td>
<td>50000</td>
<td>69</td>
<td>DIAGONAL 4</td>
<td>10000</td>
</tr>
<tr>
<td>28</td>
<td>TEST</td>
<td>3</td>
<td>70</td>
<td>DIAGONAL 4</td>
<td>100000</td>
</tr>
<tr>
<td>29</td>
<td>TEST</td>
<td>3</td>
<td>71</td>
<td>DIAGONAL 7</td>
<td>10</td>
</tr>
<tr>
<td>30</td>
<td>SUM OF SQUARES</td>
<td>100</td>
<td>72</td>
<td>DIAGONAL 7</td>
<td>100</td>
</tr>
<tr>
<td>31</td>
<td>SUM OF SQUARES</td>
<td>2000</td>
<td>73</td>
<td>DIAGONAL 7</td>
<td>100</td>
</tr>
<tr>
<td>32</td>
<td>SUM OF SQUARES</td>
<td>5000</td>
<td>74</td>
<td>DIAGONAL 8</td>
<td>100</td>
</tr>
<tr>
<td>33</td>
<td>SHALLOW</td>
<td>1000</td>
<td>75</td>
<td>DIAGONAL 8</td>
<td>500</td>
</tr>
<tr>
<td>34</td>
<td>QUARTIC</td>
<td>100</td>
<td>76</td>
<td>DIAGONAL 9</td>
<td>10</td>
</tr>
<tr>
<td>35</td>
<td>QUARTIC</td>
<td>1000</td>
<td>77</td>
<td>DIAGONAL 9</td>
<td>100</td>
</tr>
<tr>
<td>36</td>
<td>QUARTIC</td>
<td>5000</td>
<td>78</td>
<td>DENSCHN A</td>
<td>3000</td>
</tr>
<tr>
<td>37</td>
<td>QUARTIC</td>
<td>10000</td>
<td>79</td>
<td>DENSCHN A</td>
<td>15000</td>
</tr>
<tr>
<td>38</td>
<td>MATYAS</td>
<td>2</td>
<td>80</td>
<td>DENSCHN C</td>
<td>1000</td>
</tr>
<tr>
<td>39</td>
<td>MATYAS</td>
<td>2</td>
<td>81</td>
<td>DENSCHN C</td>
<td>10000</td>
</tr>
<tr>
<td>40</td>
<td>DIAGONAL 1</td>
<td>10</td>
<td>82</td>
<td>EXTENDED BLOCK DIAGONAL 1</td>
<td>100</td>
</tr>
<tr>
<td>41</td>
<td>DIAGONAL 1</td>
<td>100</td>
<td>83</td>
<td>EXTENDED BLOCK DIAGONAL 1</td>
<td>100</td>
</tr>
<tr>
<td>42</td>
<td>DIAGONAL 1</td>
<td>10</td>
<td>84</td>
<td>EXTENDED BLOCK DIAGONAL 1</td>
<td>1000</td>
</tr>
</tbody>
</table>
### Table 2. List of test problems part 2/2.

<table>
<thead>
<tr>
<th>NO</th>
<th>Problem</th>
<th>Dim</th>
</tr>
</thead>
<tbody>
<tr>
<td>85</td>
<td>HIMMELBLAU</td>
<td>1000</td>
</tr>
<tr>
<td>86</td>
<td>HIMMELBLAU</td>
<td>10000</td>
</tr>
<tr>
<td>87</td>
<td>HIMMELBLAU</td>
<td>50000</td>
</tr>
<tr>
<td>88</td>
<td>HIMMELBLAU</td>
<td>10000</td>
</tr>
<tr>
<td>89</td>
<td>DQDRTIC</td>
<td>1000</td>
</tr>
<tr>
<td>90</td>
<td>DQDRTIC</td>
<td>10000</td>
</tr>
<tr>
<td>91</td>
<td>DQDRTIC</td>
<td>10</td>
</tr>
<tr>
<td>92</td>
<td>DQDRTIC</td>
<td>10</td>
</tr>
<tr>
<td>93</td>
<td>QUARTICM</td>
<td>1000</td>
</tr>
<tr>
<td>94</td>
<td>QUARTICM</td>
<td>10000</td>
</tr>
<tr>
<td>95</td>
<td>LINEAR PERTURBED</td>
<td>5000</td>
</tr>
<tr>
<td>96</td>
<td>LINEAR PERTURBED</td>
<td>10000</td>
</tr>
<tr>
<td>97</td>
<td>LINEAR PERTURBED</td>
<td>20000</td>
</tr>
<tr>
<td>98</td>
<td>TWH</td>
<td>2</td>
</tr>
<tr>
<td>99</td>
<td>ENGV AL1</td>
<td>2</td>
</tr>
<tr>
<td>100</td>
<td>ENGV AL1</td>
<td>2</td>
</tr>
<tr>
<td>101</td>
<td>ENGV AL8</td>
<td>4</td>
</tr>
<tr>
<td>102</td>
<td>ENGV AL8</td>
<td>2</td>
</tr>
<tr>
<td>103</td>
<td>DENSCHN F</td>
<td>1000</td>
</tr>
<tr>
<td>104</td>
<td>DENSCHN F</td>
<td>10000</td>
</tr>
<tr>
<td>105</td>
<td>DENSCHN F</td>
<td>50000</td>
</tr>
<tr>
<td>106</td>
<td>ARWHEAD</td>
<td>10</td>
</tr>
<tr>
<td>107</td>
<td>ARWHEAD</td>
<td>100</td>
</tr>
<tr>
<td>108</td>
<td>ARWHEAD</td>
<td>500</td>
</tr>
<tr>
<td>109</td>
<td>SIX HUMP</td>
<td>2</td>
</tr>
<tr>
<td>110</td>
<td>PRICE4</td>
<td>2</td>
</tr>
<tr>
<td>111</td>
<td>PRICE4</td>
<td>2</td>
</tr>
<tr>
<td>112</td>
<td>ZIRILLI</td>
<td>2</td>
</tr>
<tr>
<td>113</td>
<td>ZIRILLI</td>
<td>2</td>
</tr>
<tr>
<td>114</td>
<td>EXTENDED HIMMELBLAU</td>
<td>500</td>
</tr>
<tr>
<td>115</td>
<td>EXTENDED HIMMELBLAU</td>
<td>1000</td>
</tr>
<tr>
<td>116</td>
<td>EXTENDED HIMMELBLAU</td>
<td>2000</td>
</tr>
<tr>
<td>117</td>
<td>ROTATED ELLIPSE</td>
<td>2</td>
</tr>
<tr>
<td>118</td>
<td>ROTATED ELLIPSE</td>
<td>2</td>
</tr>
<tr>
<td>119</td>
<td>EL-ATTAR-VIDYASAGAR-DUTTA</td>
<td>2</td>
</tr>
<tr>
<td>120</td>
<td>EL-ATTAR-VIDYASAGAR-DUTTA</td>
<td>2</td>
</tr>
<tr>
<td>121</td>
<td>EXTENDED HIEBERT</td>
<td>2</td>
</tr>
<tr>
<td>122</td>
<td>EXTENDED TRIDIAGONAL 1</td>
<td>100</td>
</tr>
<tr>
<td>123</td>
<td>EXTENDED TRIDIAGONAL 1</td>
<td>500</td>
</tr>
<tr>
<td>124</td>
<td>THUMP</td>
<td>2</td>
</tr>
<tr>
<td>125</td>
<td>THUMP</td>
<td>2</td>
</tr>
</tbody>
</table>

5. Application of the NSDY algorithm in restoration of corrupted images

The CG method is among the most efficient minimization algorithms used for faster optimization of both linear and non-linear systems because of its rapid solution for large-scale problems. Recently, the
CG iterative scheme has been widely considered for solving real-world application problems because of its fewer iterations and less computational resources required to optimize a given problem. Some of the most relevant problems solved by the CG method includes the feed forward training of neural networks [3], the problem of motion control of robotic manipulators [8], regression analysis [1,6], and the restoration of corrupted images [8,9].

Image restoration is among the family of inverse problems used to obtain some highly-quality images from those images that have possibly been corrupted during the data acquisition process. Choosing an appropriate algorithm that is capable of restoring these corrupted images is very crucial because knowledge of the degradation system has to be considered in order to achieve better quality images. In this study, we consider restoring some images which include a building (512 × 512) and camera (512 × 512). These images were possibly corrupted by salt-and-pepper impulse noise during the data acquisition or storage process. For the purpose of this study, the following metrics have been employed to measure the quality of the restored images and evaluate the performance of the algorithms. These metrics include the following

- peak signal-to-noise ratio (PSNR),
- relative error (RelErr),
- CPU time.

The PSNR has widely been used to measure the perceptual quality of restored images because, it calculates the ratio between the power of corrupting noises and the maximum possible power of a signal affecting the fidelity of the representation. This metric is very crucial when evaluating the quality of both corrupted and restored images. It is important to note that a method with higher PSNR will produce images with better quality [40]. The PSNR can be obtained as follows

\[
PNSR = 10 \cdot \log_{10} \left( \frac{MAX_I^2}{MSE} \right) = 20 \cdot \log_{10} \left( \frac{MAX_I}{\sqrt{MSE}} \right) = 20 \cdot \log_{10} (MAX_I) - 10 \cdot \log_{10} (MSE) \tag{5.1}
\]

where \( MSE \) is used to measure the average pixel differences of the complete images and \( MAX_I \) represents the maximum pixel value of the images. Also, an algorithm with higher \( MSE \) values will produce greater difference between the original and corrupted images. The \( MSE \) values can be obtained via:

\[
MSE = \frac{1}{N} \sum \sum (E_{i,j} - o_{i,j})^2, \tag{5.2}
\]

where \( N \) denotes image size, \( E \) is the edge image and \( o \) represents the original image.

5.1. Image restoration problem

Consider the following image restoration problem:

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

AIMS Mathematics Volume 8, Issue 12, 30940–30962.
where the edge-preserving potential function $\phi_\alpha(t) = \sqrt{t^2 + \alpha}$ for some constant $\alpha$ whose value is given as 1.

The above problem is transformed into the following optimization problem [41]

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

where $x$ is the original $M \times N$ pixel image. Also, $G$ denotes the following index set of noise candidates $x$:

$$G = \{(i, j) \in Q | \tilde{\xi}_{ij} \neq \xi_{ij}, \xi_{ij} = s_{\text{min}} \text{ or } s_{\text{max}}\}, \quad (5.3)$$

where $i, j \in Q = \{1, 2, \cdots, M\} \times \{1, 2, \cdots, N\}$ its neighborhood is defined as

$$T_{i,j} = \{(i, j-1), (i, j+1), (i-1, j), (i+1, j)\}. \quad (5.4)$$

From (5.3), the maximum and minimum values of the noisy pixel are $s_{\text{max}}$ and $s_{\text{min}}$. Also, $\xi$ is the observed noisy corrupted image whose adaptive median filter is denoted as $\tilde{\xi}$.

Next, we present the computational performance of the proposed algorithm in Tables 3, 4, and 5. The performance was compared with that of other similar algorithms with the same characteristics in terms of the PNSR, RelErr, and CPU time to demonstrate the robustness of the algorithm. All computations were carried out under the conditions of the strong Wolfe line search with the restoration noise degrees set as 30, 50 and 80, respectively.

From the results presented in the tables above, it is obvious that all of the methods are very competitive. Starting with the CPU time of the camera image, it can be observed that at 30% noise degree, the CPU time for the proposed method is lower than that of DDY1 and IDY algorithms but higher than that of DY and JYJLL methods. However, at the 50% noise degree, the proposed algorithm outperformed all of the other methods, achieving the shortest CPU time. Also, at the 80% noise degree, our method was superior to all other algorithms except the IDY algorithm which produced the shortest CPU time. Similarly, by considering the Performance of the new method in terms of restoring the building corrupted image also shows that the method is very competitive. Furthermore, overall performance analysis of the methods (see Tables 4 and 5) in terms of the PSNR and RelErr shows that the proposed method performed better because its produced higher values for PSNR and RelErr in most of the noise degrees considered. Based on the previous discussion, the higher the PNSR and RelErr values, the better the quality of the output images. For the graphical representation of the results, we refer the reader to Figures 4 and 5. These results were obtained for the 30%, 50%, and 80% noise degrees, respectively.

Based on Tables 3, 4 and 5 and Figures 4 and 5, we can conclude that the proposed method is more efficient and robust because it restored most of the corrupted images with high accuracy than other existing methods.
Table 3. Image restoration outputs for NSDY, DY, DDY1, JYJLL, and IDY in terms of CPU time (CPUT).

<table>
<thead>
<tr>
<th>METHOD</th>
<th>NSDY</th>
<th>DY</th>
<th>DDY1</th>
<th>JYJLL</th>
<th>IDY</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMAGE</td>
<td>NOISE</td>
<td>CPUT</td>
<td>CPUT</td>
<td>CPUT</td>
<td>CPUT</td>
</tr>
<tr>
<td>CAMERA</td>
<td>30%</td>
<td>48.2371</td>
<td>46.2840</td>
<td>48.3230</td>
<td>46.6328</td>
</tr>
<tr>
<td></td>
<td>50%</td>
<td><strong>76.4981</strong></td>
<td>112.2052</td>
<td>78.4817</td>
<td>76.7701</td>
</tr>
<tr>
<td></td>
<td>80%</td>
<td>170.6165</td>
<td>261.6915</td>
<td>184.9280</td>
<td>174.2533</td>
</tr>
<tr>
<td>BUILDING</td>
<td>30%</td>
<td>60.4763</td>
<td>61.8992</td>
<td>64.5338</td>
<td>46.4718</td>
</tr>
<tr>
<td></td>
<td>50%</td>
<td>143.6863</td>
<td>147.1558</td>
<td>104.1381</td>
<td>82.7440</td>
</tr>
<tr>
<td></td>
<td>80%</td>
<td>174.7825</td>
<td>187.3404</td>
<td><strong>159.8783</strong></td>
<td>191.3842</td>
</tr>
</tbody>
</table>

Table 4. Image restoration outputs for NSDY, DY, DDY1, JYJLL, and IDY in terms of RelErr.

<table>
<thead>
<tr>
<th>METHOD</th>
<th>NSDY</th>
<th>DY</th>
<th>DDY1</th>
<th>JYJLL</th>
<th>IDY</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMAGE</td>
<td>NOISE</td>
<td>RelErr</td>
<td>RelErr</td>
<td>RelErr</td>
<td>RelErr</td>
</tr>
<tr>
<td>CAMERA</td>
<td>30%</td>
<td>1.2046</td>
<td>1.0732</td>
<td><em>1.2665</em></td>
<td>1.1387</td>
</tr>
<tr>
<td></td>
<td>50%</td>
<td>1.6940</td>
<td><em>1.9179</em></td>
<td>1.6558</td>
<td>1.5858</td>
</tr>
<tr>
<td></td>
<td>80%</td>
<td><em>3.6860</em></td>
<td>3.5995</td>
<td>3.2609</td>
<td>3.0341</td>
</tr>
<tr>
<td>BUILDING</td>
<td>30%</td>
<td>1.4926</td>
<td>1.4439</td>
<td>1.4831</td>
<td>1.3775</td>
</tr>
<tr>
<td></td>
<td>50%</td>
<td>2.5823</td>
<td>2.6177</td>
<td>2.5245</td>
<td>2.5926</td>
</tr>
<tr>
<td></td>
<td>80%</td>
<td>4.9713</td>
<td>5.3007</td>
<td><em>5.4765</em></td>
<td>4.9022</td>
</tr>
</tbody>
</table>

Table 5. Image restoration outputs for NSDY, DY, DDY1, JYJLL, and IDY in terms of PSNR.

<table>
<thead>
<tr>
<th>METHOD</th>
<th>NSDY</th>
<th>DY</th>
<th>DDY1</th>
<th>JYJLL</th>
<th>IDY</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMAGE</td>
<td>NOISE</td>
<td>PSNR</td>
<td>PSNR</td>
<td>PSNR</td>
<td>PSNR</td>
</tr>
<tr>
<td>CAMERA</td>
<td>30%</td>
<td>30.6398</td>
<td><em>30.9961</em></td>
<td>30.5249</td>
<td>30.8717</td>
</tr>
<tr>
<td></td>
<td>50%</td>
<td>27.6515</td>
<td>27.2421</td>
<td>27.5891</td>
<td><em>27.7729</em></td>
</tr>
<tr>
<td></td>
<td>80%</td>
<td>23.7462</td>
<td>23.2416</td>
<td>23.3517</td>
<td>23.6676</td>
</tr>
<tr>
<td>BUILDING</td>
<td>30%</td>
<td>29.8875</td>
<td>29.8283</td>
<td>29.7804</td>
<td><em>29.9679</em></td>
</tr>
<tr>
<td></td>
<td>80%</td>
<td>22.2909</td>
<td>21.9993</td>
<td>22.0771</td>
<td>22.2824</td>
</tr>
</tbody>
</table>
Figure 4. Camera image corrupted by (a) 30%, (b) 50% and (c) 80% salt-and-pepper noise; the restored images using NSDY: (d,e,f), DY: (g,h,i), DDY1: (j,k,l), JYJLL (m,n,o), IDY (p,q,r).
Figure 5. Building image corrupted (a) 30%, (b) 50% and (c) 80% salt-and-pepper noise; the restored images using NSDY: (d,e,f), DY: (g,h,i), DDY1: (j,k,l), JYJLL (m,n,o), IDY (p,q,r).
6. Conclusions

In this study, considering the D-L conjugacy condition as well as the Newton search direction, a spectral DY CG method for large-scale unconstrained optimization and image restoration problems is suggested. The proposed method introduces a new spectral parameter $\theta_k$ to scale the search direction $d_k$ such that it satisfies the sufficient descent property and converges globally with strong Wolfe criteria. The preliminary computational results a set of optimization functions analyzed in comparison to those obtained for some DY modifications in literature indicated that the proposed method is efficient and reliable.

Use of AI tools declaration

The authors declare that they have not used artificial intelligence tools in the creation of this article.

Acknowledgments

The authors acknowledge the financial support provided by the Center of Excellence in Theoretical and Computational Science (TaCS-CoE), KMUTT and Petchra Pra Jom Klao PhD Scholarship of King Mongkut’s University of Technology Thonburi (KMUTT), contract No. 52/2564. Moreover, the authors also acknowledged the financial support provided by the Mid-Career Research Grant (N41A640089).

Conflict of interest

The authors declare no competing interests.

References


