Iranian Journal of Operations Research

Vol. 16, No. 1, 2025, pp. 1-17

A modified hybrid three-term conjugate gradient method and its applications in image restoration

Meysam Ranjbar[1], Ali Ashrafi [2,*]

This paper proposes a modified hybrid three-term conjugate gradient (CG) method for solving unconstrained optimization problems. The search direction is constructed by combining the Hestenes-Stiefel (HS) and Liu–Storey (LS) CG parameters in a hybrid three-term formula. We show that the proposed method satisfies the sufficient descent condition independently of line search strategies. A convergence analysis is provided under standard assumptions for general objective functions. Numerical experiments on CUTEr test problems and image-denoising tasks indicate that the proposed method outperforms existing approaches regarding efficiency, accuracy, and robustness, particularly in the presence of high levels of salt-and-pepper noise.

Keywords: *unconstrained optimization, three-term conjugate gradient, sufficient descent property, global convergence, image restoration.*

1. Introduction

The unconstrained optimization problem is as follows:

$$\min_{x\in\mathbb{R}^n} f(x),$$

in which $f:\mathbb{R}^n \to \mathbb{R}$ is a smooth function. In unconstrained optimization, particularly when minimizing the differentiable function f(x), i.e. $\min_{x\in\mathbb{R}^n} f(x)$ CG methods have gained widespread adoption. This surge in popularity can be attributed to several factors, notably the simplicity of the iterations, minimal memory requirements, and rapid convergence rates. Among these techniques, the three-term CG method stands out, and it is determined as follows:

$$d_k = \begin{cases} -g_k, & if \quad k=0, \\ -g_k + \beta_k d_{k-1} + \gamma_k h_k, & if \quad k \geq 1. \end{cases} \tag{1}$$

In this framework, the parameters β_k and γ_k are of significant importance. β_k represents the CG parameter, while γ_k is an arbitrary parameter, with various forms of both having been presented. Moreover, several choices for h_k have been proposed to ensure convergence alongside a sufficient descent condition, i.e.

$$g_k^T d_k \leq -c||g_k||^2, \; k = 0,1,\ldots, \tag{2}$$

[*] Corresponding Author.

[1] Department of Mathematics, Faculty of Mathematics, Statistics and Computer Science, Semnan University, P.O. Box:19111-35131, Semnan, Iran, Email: ranjbar.meysam.73@semnan.ac.ir.

[2] Department of Mathematics, Faculty of Mathematics, Statistics and Computer Science, Semnan University, P.O. Box: 19111-35131, Semnan, Iran, Email: a_ashrafi@semnan.ac.ir.

Where $c > 0$ is constant, g_k represents the gradient of the objective function, and $\|.\|$ denotes the Euclidean norm [8, 17, 21, 22]. Currently, the three-term CG method has garnered significant attention from researchers across multiple scientific disciplines. For instance, Liu et al. [14] developed two sufficient descent CG three-term methods that achieve global convergence when applied with Wolfe line search conditions. These methods were utilized to address the $l_{۱} - \alpha l_2$ regularization problem in sparse signal decoding within the context of compressed sensing. Kim et al. [12] introduce a variable three-term CG technique that uses an approximation of the Hessian matrix to improve search direction, utilizing a variable step size to enhance convergence stability. To assess the efficacy of their approach, they train various artificial neural networks (ANN) on standard datasets for image classification and generation. Additionally, they carry out an analogous experiment involving a grasp generation and selection convolutional neural network (CNN) designed for intelligent robotic grasping. Following evaluations in an environment that is simulated, they also test the grasp generation and selection CNN (GGS-CNN) with a physical gripping robot. Ibrahim and Khudhur [11] created a CG algorithm featuring three limits based on the Dai-Liao conjugate condition. This new algorithm offers global convergence and adequate sufficient descent conditions (2) under certain assumptions and was applied to eliminate noise from images. Mousavi et al. [16] proposed two effective three-term CG methods aimed at eliminating impulse noise. The approach begins with the steepest descent direction, followed by the inclusion of three components: the steepest descent direction, the previous direction, and the gradient difference between the preceding and current points. The second and third components are adjusted using two distinct step sizes, referred to as CG parameters. This adjustment ensures that all components contribute without any of them overwhelmingly dominating the others, except in the vicinity of the optimizer, where the first term takes precedence. The authors applied these methods to remove noise from medical images to demonstrate the effectiveness of their proposed techniques. Recent research by Abubakr et al. [13] has advanced by developing three-term formulas for conjugate gradient (CG) methods through a reformulation of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) direction formula. They went on to propose a novel three-term CG formula that meets the necessary descent condition for global convergence without relying on a line search.

Motivated by the ideas presented in [13, 15, 23], we introduce a hybrid three-term CG method for solving unconstrained optimization problems. The direction combines the three-term HS and LS directions, and to achieve better numerical results than several other methods, we include an additional term. Moreover, the direction closely resembles that of the memoryless Broyden-Fletcher-Goldfarb-Shanno (BFGS) quasi-Newton method and possesses descent properties.

The main contributions of this paper are:

(1) Based on the LBFGS method, we propose a new hybrid three-term CG method for solving unconstrained optimization problems.

(2) The search direction of the proposed method satisfies the sufficient descent property without the need for any line search.

(3) The global convergence of the proposed method is proved using the weak Wolfe line search.

(4) The computational performance of the new method is evaluated on several standard test problems.

(5) Numerical experiments are conducted to test the proposed method on unconstrained problems, including applications in image restoration.

The structure of this paper is outlined below. The subsequent section details the derivation of the three-term method and discusses its convergence. In Section 3, we showcase various numerical experimental results. In Section 4, we present Challenges in Existing CG Methods and Our Contributions. Finally, a conclusion is given in Section 5.

2. Results from theory and algorithms

The hybrid CG approach proposed by Abubakar et al [13] is presented in the following manner:

$$\beta_k = \frac{g_k^T y_{k-1}}{\varsigma} - \frac{\| y_{k-1} \|^2 \, g_k^T d_{k-1}}{\varsigma^2},$$

$$\varsigma = \max\{\mu \| d_{k-1} \| \| y_{k-1} \|, d_{k-1}^T y_{k-1}, -d_{k-1}^T g_{k-1}\}, \qquad \mu > 0,$$

$$\gamma_k = t_k \frac{g_k^T d_{k-1}}{\varsigma_k},$$

$$t_k = \min\left\{\bar{t}, \max\left\{0, \frac{y_{k-1}^T (y_{k-1} - s_{k-1})}{\| y_{k-1} \|^2}\right\}\right\}.$$

This approach, independent of the line search, meets the sufficient descent property and incorporates a trust region. It achieves global convergence for a broad class of functions when using either Wolfe-type or Armijo-type line search. Drawing inspiration from the approach of Abubakr et al [1], we present a three-term CG method with a general structure of (1), defined by the following parameters:

$$\beta_k = \frac{g_k^T y_{k-1}}{\zeta} - \frac{\| y_{k-1} \|^2 \, g_k^T d_{k-1}}{\zeta^2}, \tag{3}$$

$$\zeta = \max\{\mu \| d_{k-1} \| \| y_{k-1} \|, d_{k-1}^T y_{k-1}, -d_{k-1}^T g_{k-1}\} + \lambda \| g_{k-1} \|^2, \qquad \mu, \lambda > 0, \tag{4}$$

$$\gamma_k = t_k \frac{g_k^T d_{k-1}}{\zeta_k}, \tag{5}$$

$$h_k = y_{k-1}. \tag{6}$$

To determine the parameter t_k, we require solving the univariate minimal problem as follows:

$$\min_{t \in \mathbb{R}} \| (y_{k-1} - s_{k-1}) - t y_{k-1} \|^2 .$$

Assuming $A_k = y_{k-١} - s_{k-١}$ and $B_k = A_k - t y_{k-١}$, we find that

$$\begin{aligned} B_k B_k^T &= (A_k - t y_{k-1})(A_k - t y_{k-1})^T, \\ &= t^2 y_{k-1} y_{k-1}^T - t[A_k y_{k-1}^T + y_{k-1} A_k^T] + A_k A_k^T, \end{aligned}$$

and

$$\begin{aligned} tr(B_k B_k^T) &= t^2 \| y_{k-1} \|^2 - t[tr(A_k y_{k-1}^T) + tr(y_{k-1} A_k^T)] + \| A_k \|^2 \\ &= t^2 \| y_{k-1} \|^2 - 2t y_{k-1}^T A_k + \| A_k \|^2 . \end{aligned}$$

By differentiating the expression above concerning t and setting it equal to zero, we acquire

$$t = \frac{y_{k-1}^T(y_{k-1} - s_{k-1})}{\parallel y_{k-1} \parallel^2}, \tag{7}$$

Afterward, we select t_k as

$$t_k = \min\left\{\bar{t}, \max\left\{0, \frac{y_{k-1}^T(y_{k-1} - s_{k-1})}{\parallel y_{k-1} \parallel^2}\right\}\right\} \tag{8}$$

indicating that $0 \leq t_k \leq \bar{t} < 1$. In (8), $\bar{t}$ is a positive constant.

The new algorithm can be shown as follows, based on the analysis above.

Algorithm 1: *MTTHSLS algorithm*

```
Step 0: Given a starting point $x_0 \in \mathbb{R}^n$, parameters $\epsilon > 0$, $\mu > 0$, $\lambda > 0$, $0 < \delta < \sigma < 1$. Calculate $f_0 = f(x_0)$ and $g_0 = \nabla f(x_0)$. Set $d_0 = -g_0$, $k = 0$.
Step 1: If $||g_k||_\infty < \epsilon$, then stop. Otherwise, proceed to step 2.
Step 2: Compute the step-length $\alpha_k$ by the Wolfe line search conditions.
Step 3: Let $x_{k+1} = x_k + \alpha_k d_k$, calculate $g_{k+1}$, $f(g_{k+1})$.
Step 4: Determine $t_k$, $\gamma_k$, and $\beta_k$ by (8), (5), and (3) respectively. Compute the search direction by (1)
Step 5: Set $k = k + 1$, then proceed to step 1.
```

The MTTHSLS method calculates the search direction $d_{k+١}$, and we will now demonstrate its sufficient descent property.

Lemma 2.1 *The search direction (1) defined in equation (3) fulfills condition (2) with* $c = 1 + \frac{(1+t_k)^2}{4}$.

Proof. We have,

$$\begin{aligned}
g_k^T d_k &= - \parallel g_k \parallel^2 + \frac{g_k^T y_{k-1}}{\zeta_k} g_k^T d_{k-1} - \frac{\parallel y_{k-1} \parallel^2}{\zeta_k^2}(g_k^T d_{k-1})^2 + t_k \frac{g_k^T y_{k-1}}{\zeta_k} g_k^T d_{k-1} \\
&= - \parallel g_k \parallel^2 + (1 + t_k)\frac{g_k^T y_{k-1}}{\zeta_k} g_k^T d_{k-1} - \frac{\parallel y_{k-1} \parallel^2}{\zeta_k^2}(g_k^T d_{k-1})^2 \\
&= - \parallel g_k \parallel^2 + 2\left(\frac{1 + t_k}{2} g_k^T\right)\frac{y_{k-1}}{\zeta_k} g_k^T d_{k-1} - \frac{\parallel y_{k-1} \parallel^2}{\zeta_k^2}(g_k^T d_{k-1})^2 \\
&\leq - \parallel g_k \parallel^2 + \frac{(1 + t_k)^2}{4} \parallel g_k \parallel^2 + \frac{\parallel y_{k-1} \parallel^2}{\zeta_k^2}(g_k^T d_{k-1})^2 - \frac{\parallel y_{k-1} \parallel^2}{\zeta_k^2}(g_k^T d_{k-1})^2 \\
&= -\parallel g_k \parallel^2 + \frac{(1 + t_k)^2}{4} \parallel g_k \parallel^2 = -\left(1 + \frac{(1 + t_k)^2}{4}\right) \parallel g_k \parallel^2 \\
&= -c \parallel g_k \parallel^2 .
\end{aligned}$$

Assumptions 2.1 *The level set* $H = \{x: f(x) \leq f(x_0)\}$ *is bounded. Then, the function* f *is bounded, and its gradient is Lipschitz continuous on* H. *This indicates that a constant* $L > 0$ *exists such that for all* x *and* $\bar{x} \in H$, $\parallel g(x) - g(\bar{x}) \parallel \leq L \parallel x - \bar{x} \parallel$.

Based on Assumptions (2.1), we can conclude that for every $x \in H$, there are constants $b_١ > 0$ such that $\parallel g(x) \parallel \leq b_1$. Furthermore, the sequence $\{x_k\}$ is contained within H because the values $\{f(x_k)\}$

are non-increasing. Thus, we will assume that Assumptions (2.1) are satisfied. We will now proceed to demonstrate the convergence result.

Theorem 2.1 . *Assume that the step size α_k is acquired through the Wolfe line search conditions[17]. If*

$$\sum_{k=0}^{\infty} \frac{1}{\| d_k \|^2} = +\infty, \tag{9}$$

then

$$\liminf_{k\to\infty} \| g_k \| = 0. \tag{10}$$

***Proof*.** From (3) and (5), we have

$$|\beta_k| = \left| \frac{g_k^T y_{k-1}}{\zeta} - \frac{\| y_{k-1} \|^2 \, g_k^T d_{k-1}}{\zeta^2} \right|$$

$$\leq \frac{\| g_k \| \| y_{k-1} \|}{\mu \| d_{k-1} \| \| y_{k-1} \| + \lambda \| g_{k-1} \|^2} + \frac{\| y_{k-1} \|^2 \| g_k \| \| d_{k-1} \|}{(\mu \| d_{k-1} \| \| y_{k-1} \| + \lambda \| g_{k-1} \|^2)^2}$$

$$\leq \frac{\| g_k \| \| y_{k-1} \|}{\mu \| d_{k-1} \| \| y_{k-1} \|} + \frac{\| y_{k-1} \|^2 \| g_k \| \| d_{k-1} \|}{(\mu \| d_{k-1} \| \| y_{k-1} \|)^2}$$

$$= \left(\frac{1}{\mu} + \frac{1}{\mu^2}\right) \frac{\| g_k \|}{\| d_{k-1} \|}. \tag{11}$$

Also

$$|\gamma_k| = \left| t_k \frac{g_k^T d_{k-1}}{\zeta_k} \right| \leq \bar{t} \frac{\| g_k \| \| d_{k-1} \|}{\zeta_k} \leq \bar{t} \frac{\| g_k \| \| d_{k-1} \|}{\mu \| d_{k-1} \| \| y_{k-1} \| + \lambda \| g_{k-1} \|^2}$$

$$\leq \bar{t} \frac{\| g_k \| \| d_{k-1} \|}{\mu \| d_{k-1} \| \| y_{k-1} \|} = \frac{\bar{t}}{\mu} \frac{\| g_k \|}{\| y_{k-1} \|}. \tag{12}$$

Now, from (1), (11), and (12), we have

$$\| d_k \| = \| -g_k + \beta_k d_{k-1} + \gamma_k y_{k-1} \| \leq \| g_k \| + \| \beta_k \| \| d_{k-1} \| + \| \gamma_k \| \| y_{k-1} \|$$

$$\leq \| g_k \| + \left(\frac{1}{\mu} + \frac{1}{\mu^2}\right) \frac{\| g_k \|}{\| d_{k-1} \|} \| d_{k-1} \| + \frac{\bar{t}}{\mu} \frac{\| g_k \|}{\| y_{k-1} \|} \| y_{k-1} \|$$

$$= \left(1 + \frac{1+\bar{t}}{\mu} + \frac{1}{\mu^2}\right) \| g_k \| \leq \left(1 + \frac{1+\bar{t}}{\mu} + \frac{1}{\mu^2}\right) b_1.$$

Then, we get $\| d_k \| \leq M$ where $M = (1 + \frac{1+\bar{t}}{\mu} + \frac{1}{\mu^2}) b_1$. This signifies the formation of a relationship (10) due to

$$\| d_k \| \leq M \rightarrow \frac{1}{M} \leq \frac{1}{\| d_k \|},$$

$$\rightarrow \frac{1}{M^2} \leq \frac{1}{\| d_k \|^2},$$

$$\rightarrow \frac{1}{\| d_k \|^2} \frac{\| g_k \|^4}{M^2} \leq \frac{\| g_k \|^4}{\| d_k \|^2},$$

$$\rightarrow \frac{1}{M^2} \sum_{k=1}^{\infty} \| g_k \|^4 \leq \sum_{k=1}^{\infty} \frac{\| g_k \|^4}{\| d_k \|^2},$$

$$\rightarrow \frac{1}{M^2} \sum_{k=1}^{\infty} \| g_k \|^4 < \infty,$$

$$\rightarrow \lim_{k \rightarrow \infty} \| g_k \|^4 = 0,$$

$$\rightarrow \lim_{k \rightarrow \infty} \| g_k \| = 0.$$

3. Numerical experiments

This section includes numerical experiments that assess the effectiveness of MTTHSLS, HTTHSLS[13], HTTWYL[23], and HTHP [15]. Details about the test problems, which include 92 functions from [7], can be found in Tables 1–2. All implementations were carried out using MATLAB 7.14.0.739 (R2012a) on a desktop equipped with an AMD Ryzen 7 PRO 5850U with Radeon Graphics 1.90 GHz, 16 GB of RAM, and the CentOS 6.2 Linux operating system.

In our implementations, we chose $\delta = 0.0001$ and $\sigma = 0.99$ for the strong Wolfe line search conditions [17]. Furthermore, we set the method parameters to the values of $\bar{t} = 0.2$, $\lambda = 0.8$, and $\mu = 0.02$. Furthermore, the algorithms were terminated when either $k > 10000$ or $||g_k|| < 10^{-6}(1 + |f_k|)$ met certain termination criteria. Additionally, the algorithm's efficiency was evaluated using the performance profile introduced by Dolan and Moré [5], focusing on the total number of function and gradient evaluations (TNFGE), as defined in [9], along with CPU time (CPUT), by the nomenclature from [3]. The findings are shown in Figures 1 and 2; it is clear that MTTHSLS surpasses the three other methods.

Table 1. Numerical results for all methods.

Function	n	MTTHSLS		HTTHSLS		HTTWYL		HTHP	
		TNF	TIME	TNF	TIME	TNF	TIME	TNF	TIME
ARGLINA	2	12	1.24E-01	12	7.65E-02	12	7.68E-02	12	9.58E-02
BDEXP	6	16	6.76E-02	16	5.71E-02	16	5.45E-02	16	5.30E-02
BDQRTIC	7	40129	1.27E+01	17561	4.06E+00	25805	6.14E+00	18651	4.45E+00
BIGGSB١	8	40157	6.94E+00	40400	4.79E+00	40332	5.28E+00	38743	4.63E+00
BQPGABIM	150	219	2.04E-02	272	3.19E-02	427	3.82E-02	424	3.10E-02
BQPGASIM	151	219	2.81E-02	272	2.76E-02	427	3.20E-02	424	3.21E-02
BROWNAL	11	18127	2.64E+00	7946	1.07E+00	8942	1.20E+00	5123	7.37E-01
BROYDN٧D	12	40076	3.07E+01	40132	2.89E+01	40141	2.93E+01	40392	2.90E+01
BRYBND	13	1161	5.52E-01	3239	1.35E+00	3066	1.57E+00	3647	1.50E+00
CHAINWOO	14	40085	1.05E+01	40588	8.82E+00	41658	9.31E+00	42051	9.07E+00
CHENHARK	155	14577	2.59E+00	11636	2.05E+00	4152	7.95E-01	528	1.35E-01
CHNROSNB	57	31544	1.37E+00	39810	1.62E+00	42221	1.79E+00	31030	1.27E+00
CLPLATEB	157	40093	1.35E+01	40293	1.17E+01	40408	1.21E+01	40199	1.16E+01
COSINE	15	57	1.32E-01	53	1.21E-01	60	1.31E-01	55	1.20E-01
CRAGGLVY	59	803	5.05E-01	953	5.71E-01	854	5.33E-01	1306	7.59E-01
CURLY١٠	61	5055	2.27E+00	6720	2.93E+00	5291	2.35E+00	40189	1.27E+01
CURLY٢٠	62	16693	9.67E+00	27368	1.28E+01	18875	9.19E+00	5472	2.75E+00
CURLY٣٠	63	40090	2.63E+01	40668	2.49E+01	40436	2.55E+01	35652	2.18E+01
DECONVU	64	40117	1.90E+00	40892	1.91E+00	40780	1.95E+00	15649	7.13E-01
DIXMAANA	16	36	5.03E-02	36	4.58E-02	36	4.52E-02	36	5.07E-02
DIXMAANB	17	37	5.10E-02	32	5.13E-02	32	5.15E-02	32	4.21E-02
DIXMAANC	18	37	5.02E-02	37	4.51E-02	32	5.09E-02	37	5.16E-02
DIXMAAND	19	47	5.41E-02	42	5.75E-02	42	4.75E-02	46	4.56E-02
DIXMAANE	20	12402	2.15E+00	16866	2.78E+00	34644	5.74E+00	10953	1.36E+00
DIXMAANF	21	9420	1.65E+00	16395	2.73E+00	21890	3.48E+00	13072	1.64E+00
DIXMAANG	22	12431	2.18E+00	14780	2.43E+00	30800	4.54E+00	29220	3.68E+00
DIXMAANH	23	7796	1.45E+00	20027	3.40E+00	23933	3.60E+00	5314	6.74E-01
DIXMAANI	24	40074	6.80E+00	42794	5.36E+00	43932	5.65E+00	41215	5.15E+00
DIXMAANJ	25	11773	2.06E+00	11251	1.90E+00	28908	4.48E+00	6647	8.45E-01
DIXMAANK	26	8506	1.51E+00	7765	1.32E+00	7473	1.32E+00	31574	4.65E+00
DIXMAANL	27	3339	6.28E-01	7586	1.30E+00	40614	6.54E+00	13287	1.69E+00
DIXON٣DQ	71	40100	9.94E+00	40701	8.15E+00	40312	8.74E+00	41774	8.31E+00
DMN15102	163	40102	7.07E+01	40828	7.11E+01	41474	7.23E+01	41054	7.15E+01
DMN15103	164	40064	9.03E+01	40084	8.98E+01	40374	9.06E+01	42663	9.50E+01
DMN37142	165	40135	7.07E+01	40188	7.00E+01	40390	7.13E+01	42018	7.31E+01
DMN37143	166	40080	9.02E+01	44103	9.79E+01	40366	9.04E+01	43588	9.96E+01
DQDRTIC	28	641	2.56E-01	3224	9.83E-01	3427	1.10E+00	3789	1.14E+00
DQRTIC	29	4	3.86E-02	4	3.74E-02	4	3.67E-02	4	3.95E-02
DRCAV1LQ	167	4	7.54E-02	4	8.05E-02	4	8.46E-02	4	8.69E-02
DRCAV2LQ	168	4	7.72E-02	4	8.05E-02	4	7.13E-02	4	7.65E-02
DRCAV3LQ	169	4	1.03E-01	4	8.05E-02	4	7.08E-02	4	7.18E-02
EDENSCH	73	522	1.37E-01	474	1.20E-01	82	5.60E-02	310	9.96E-02
EG2	30	23	2.51E-02	23	1.69E-02	23	2.34E-02	23	2.61E-02
EIGENALS	74	40067	5.59E+01	40395	5.43E+01	40540	5.49E+01	40485	5.43E+01
EIGENBLS	75	40057	5.57E+01	40527	5.46E+01	40396	5.47E+01	41800	5.62E+01
EIGENCLS	76	40329	5.99E+01	41052	5.92E+01	42405	6.13E+01	44153	6.33E+01

Table 2. Numerical results for all methods.

Function	n	MTTHSLS		HTTHSLS		HTTWYL		HTHP	
		TNF	TIME	TNF	TIME	TNF	TIME	TNF	TIME
ENGVAL۱	31	71	9.60E-02	67	8.57E-02	63	9.35E-02	76	9.66E-02
ERRINROS	78	40064	1.71E+00	40325	1.69E+00	40295	1.75E+00	41730	1.63E+00
EXTROSNB	32	40090	3.02E+00	40594	3.02E+00	40997	3.15E+00	40865	2.25E+00
FLETCBV۲	33	4	8.21E-02	4	7.09E-02	4	6.97E-02	4	7.05E-02
FLETCBV۳	171	160	1.69E-01	160	1.63E-01	160	1.68E-01	160	1.58E-01
FLETCHBV	172	156	1.62E-01	160	1.55E-01	156	1.55E-01	156	1.52E-01
FLETCHCR	80	31141	2.80E+00	41372	3.65E+00	40990	2.88E+00	14995	9.60E-01
FMINSRF۲	81	9853	2.94E+00	12765	3.65E+00	18069	4.14E+00	11693	2.53E+00
FMINSURF	82	38713	1.06E+01	41636	9.45E+00	40270	9.56E+00	40342	9.12E+00
FREUROTH	83	3938	1.47E+00	5849	2.11E+00	5402	1.98E+00	3624	1.22E+00
GENHUMPS	84	4	4.41E-02	4	4.06E-02	4	4.76E-02	4	4.88E-02
GENHUMPS	175	4	5.73E-02	4	5.87E-02	4	4.57E-02	4	5.57E-02
GENROSE	34	40043	2.63E+00	40325	2.63E+00	40114	2.69E+00	40462	1.96E+00
LIARWHD	35	40233	1.01E+01	46752	8.14E+00	50130	9.30E+00	49122	9.08E+00
MANCINO	104	458	1.06E+00	458	1.05E+00	453	1.05E+00	453	1.04E+00
MANCINO	104	458	1.06E+00	458	1.05E+00	453	1.05E+00	453	1.05E+00
MOREBV	107	367	1.51E-01	1878	4.71E-01	26548	6.31E+00	36207	6.63E+00
MSQRTALS	108	40078	1.43E+01	40391	1.27E+01	40114	1.28E+01	40294	1.26E+01
MSQRTBLS	109	40158	1.43E+01	41043	1.29E+01	40371	1.28E+01	41285	1.29E+01
NCB20	110	393	5.15E-01	1053	1.13E+00	1220	1.29E+00	2362	2.37E+00
NCB20B	111	221	3.57E-01	873	9.60E-01	867	9.71E-01	913	1.00E+00
NONCVXU2	112	4	4.56E-02	4	4.89E-02	4	5.35E-02	4	4.27E-02
NONDIA	36	40017	8.79E+00	20858	2.95E+00	103746	1.61E+01	25306	3.63E+00
NONDQUAR	113	40092	6.96E+00	40289	5.49E+00	40609	5.52E+00	40915	5.15E+00
PENALTY1	37	3197	2.59E-01	2468	1.92E-01	1100	1.11E-01	2561	2.07E-01
PENALTY2	38	4	9.92E-03	4	9.21E-03	4	1.56E-02	4	1.29E-02
POWELLSG	125	40097	6.64E+00	45446	5.30E+00	50498	6.07E+00	41411	4.95E+00
POWER	126	40103	8.46E+00	43750	6.90E+00	40322	6.12E+00	40618	6.62E+00
QUARTC	40	4	3.98E-02	4	3.80E-02	4	3.73E-02	4	3.58E-02
SCHMVETT	41	92	1.31E-01	77	1.16E-01	73	1.14E-01	81	1.19E-01
SENSORS	129	109	2.60E-01	97	2.10E-01	97	2.08E-01	113	2.34E-01
SINQUAD	131	221	2.75E-01	85	2.10E-01	208	2.68E-01	169	2.57E-01
SPARSINE	134	40030	2.11E+01	41393	1.96E+01	42081	2.01E+01	40274	1.91E+01
SPARSQUR	42	605	4.87E-01	597	4.66E-01	602	4.84E-01	597	4.66E-01
SPMSRTLS	43	6773	2.42E+00	9215	3.16E+00	10202	3.42E+00	8237	2.13E+00
SROSENBR	44	15410	2.85E+00	13790	2.05E+00	11476	1.75E+00	10335	1.40E+00
TESTQUAD	135	40039	5.42E+00	40357	4.26E+00	53869	5.08E+00	41714	3.99E+00
TOINTGOR	136	970	6.05E-02	1563	8.74E-02	1420	7.97E-02	1905	1.04E-01
TOINTGSS	45	169	1.60E-01	356	2.72E-01	147	1.42E-01	212	1.78E-01
TOINTPSP	138	1027	6.09E-02	2448	1.14E-01	2477	1.22E-01	2589	1.19E-01
TOINTQOR	137	146	1.69E-02	143	2.43E-02	144	1.84E-02	138	2.21E-02
TQUARTIC	139	40036	8.56E+00	41574	6.60E+00	40005	6.66E+00	40485	6.35E+00
TRIDIA	46	40053	6.53E+00	42796	5.32E+00	40261	5.25E+00	41756	4.88E+00
VARDIM	47	8	3.94E-03	8	7.77E-03	8	8.56E-03	8	9.52E-03
VAREIGVL	48	175	2.02E-02	141	2.53E-02	133	2.14E-02	142	2.40E-02
WOODS	49	117	7.43E-02	244	9.57E-02	41860	8.21E+00	24181	3.44E+00

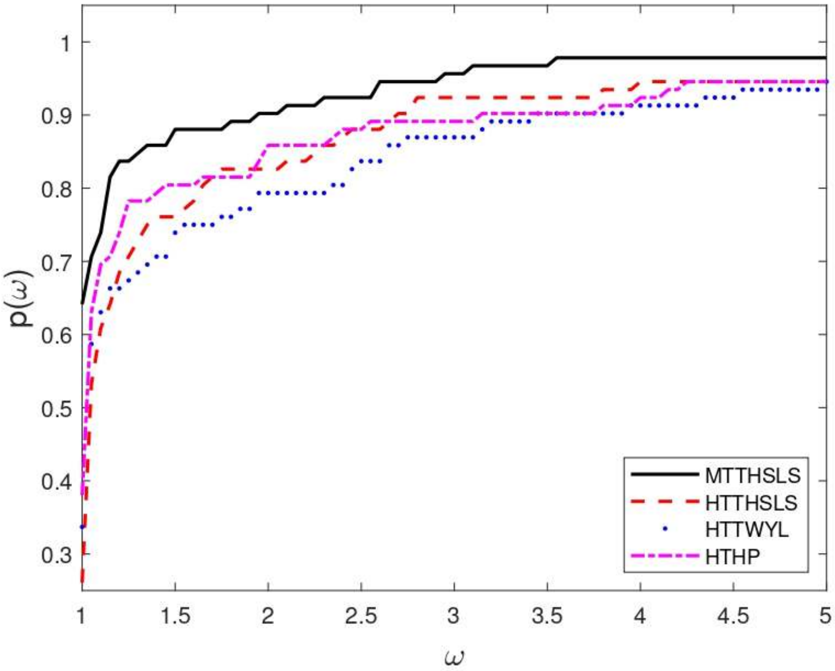


Figure 1. Comparison based on TNFGE with the strong Wolfe line search.

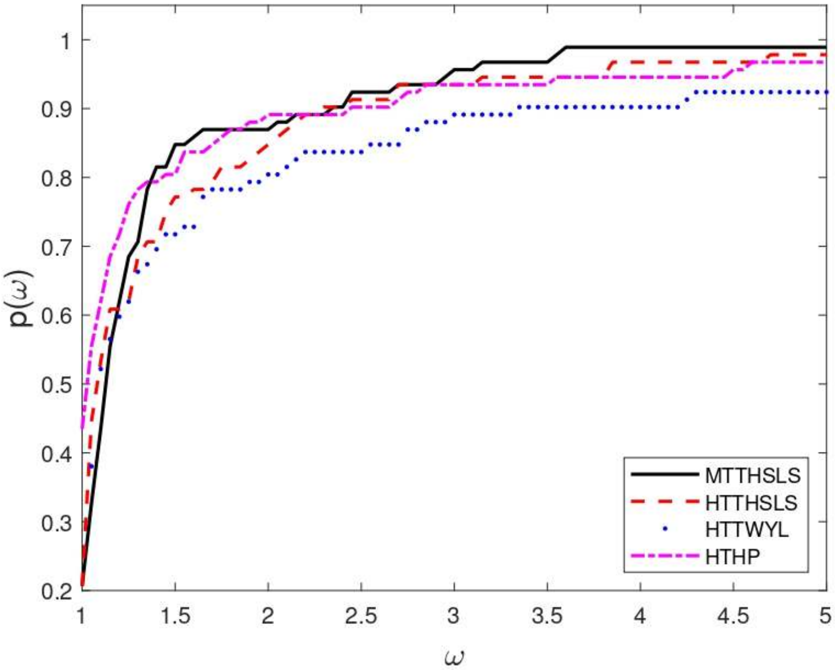


Figure 2. Comparison based on CPUT (B) with the strong Wolfe line search.

We demonstrate applying the suggested techniques to an image reconstruction problem in the sections that follow, offering a practical case study. Images frequently suffer from impulse noise caused by faulty sensors or transmission channels. This type of noise is one of the most prevalent models, where only a subset of the pixels is affected, resulting in a complete loss of information for those specific pixels. To reliably recover the original image, various image-related applications typically require effective noise suppression techniques. These issues are often viewed as complex optimization challenges due to their nonsmooth structures. Recently, researchers have focused on developing nonlinear CG algorithms to tackle such nonsmooth optimization problems; see, for instance, [4, 10, 18, 19, 20]. In this section, we address the problem of smooth image reconstruction presented in [10]:

$$\min\psi(u),$$

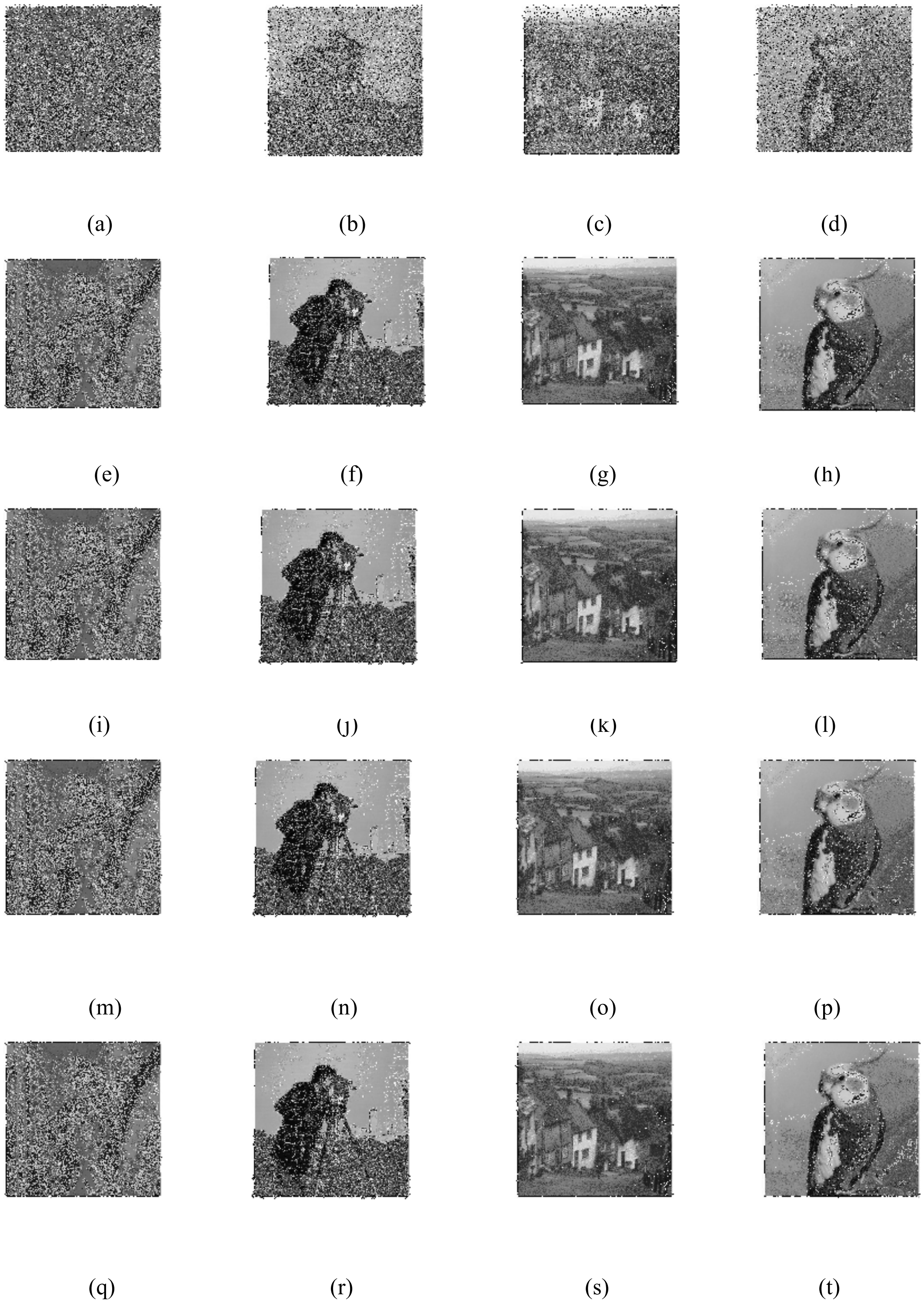
where

$$\psi(u) = \sum_{(i,j)\in\mathcal{N}} \left\{ \sum_{(m,n)\in V_{ij}\setminus\mathcal{N}} \varphi_\alpha(u_{ij} - \zeta_{mn}) + \frac{1}{2} \sum_{(m,n)\in V_{ij}\cap\mathcal{N}} \varphi_\alpha(u_{ij} - u_{mn}) \right\},$$

in which

$$\mathcal{N} = \{(i,j) \in A \mid \bar{\zeta}_{ij} \neq \zeta_{ij}, \zeta_{ij} = s_{\min} \ \ or \ \ s_{\max}\}.$$

The index set for the noise candidate is denoted as follows: let X represent the true image composed of $M \times N$ pixels, where X_{ij} indicates the gray level at the pixel location (i,j) for all $(i,j) \in A = \{1, 2, \dots, M\} \times \{1, 2, \dots, N\}$. The neighborhood around the pixel (i,j) is given by $V_{ij} = \{(i, j-1), (i, j+1), (i-1, j), (i+1, j)\}$. The observed image, ζ, reflects the true image X but it is affected by salt-and-pepper noise. The image $\bar{\zeta}$ is derived by applying an adaptive median filter to the noisy image ζ. Additionally, $s_{\min}$ and $s_{\max}$ represent the minimum and maximum values of a noisy pixel, respectively. We utilize the edge-preserving functional φ_α, defined as $\varphi_\alpha(t) = \sqrt{t^2 + \alpha}$, based on the recommendation in [10]. In our experiments, we set $\alpha = 1$ due to the satisfactory numerical results observed across various choices of α in the set $\{0.5k\}_{k=1}^{10}$.



(a) (b) (c) (d)

(e) (f) (g) (h)

(i) (j) (k) (l)

(m) (n) (o) (p)

(q) (r) (s) (t)

Figure 3: The noisy images corrupted by 25% salt–and–pepper noise: (a)–(d), the restored images via HTTHSLS: (e)–(h), HTTWY: (i)–(l), HTHP : (m)–(p), and MTTHSLS: (q)–(t)

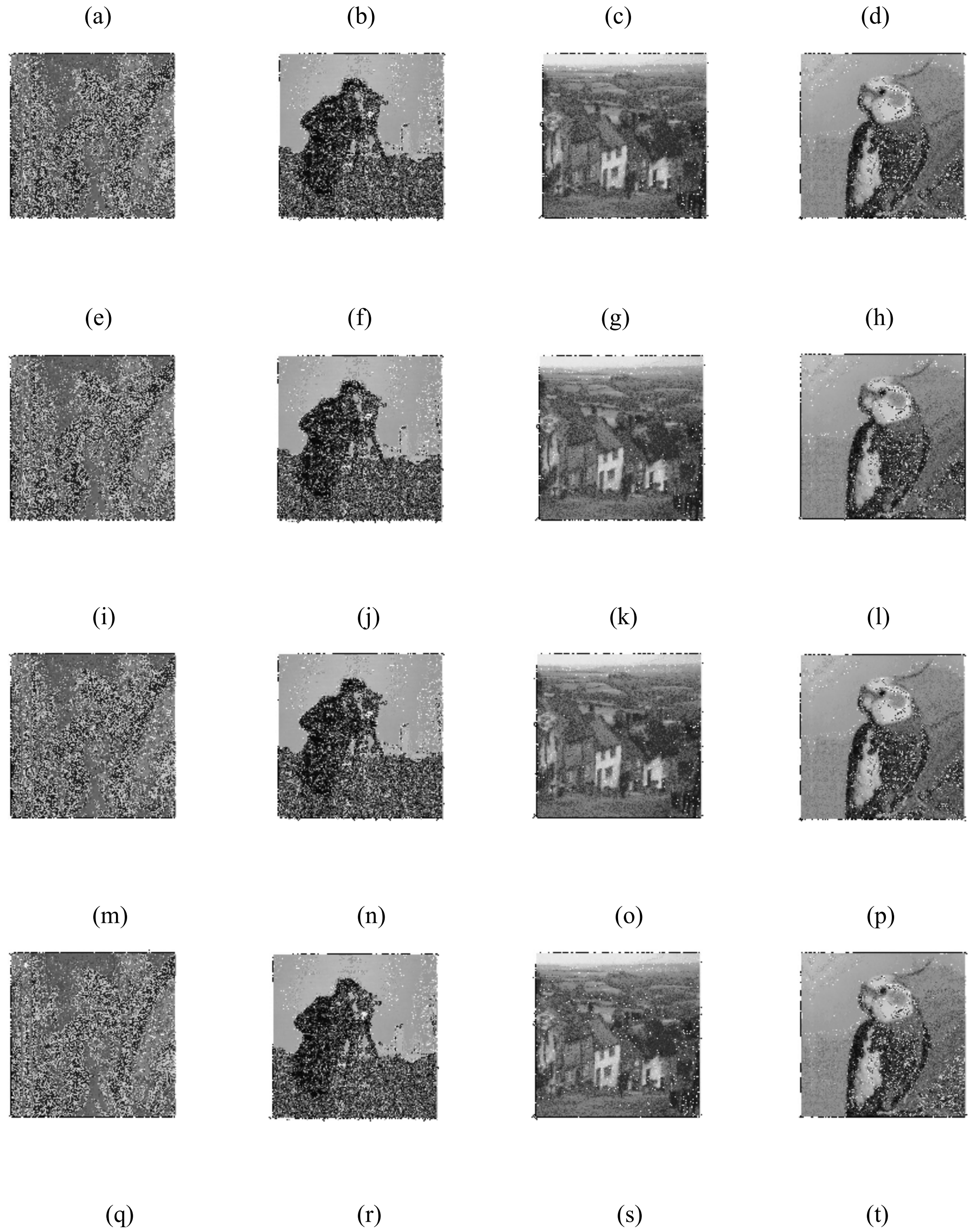
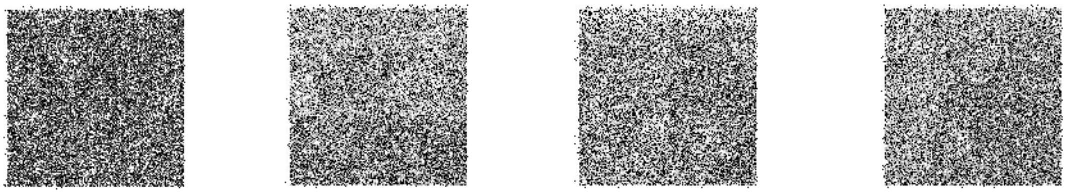


Figure 4: The noisy images corrupted by 50% salt–and–pepper noise: (a)–(d), the restored images via HTTHSLS: (e)–(h), HTTWY: (i)–(l), HTHP : (m)–(p), and MTTHSLS: (q)–(t)



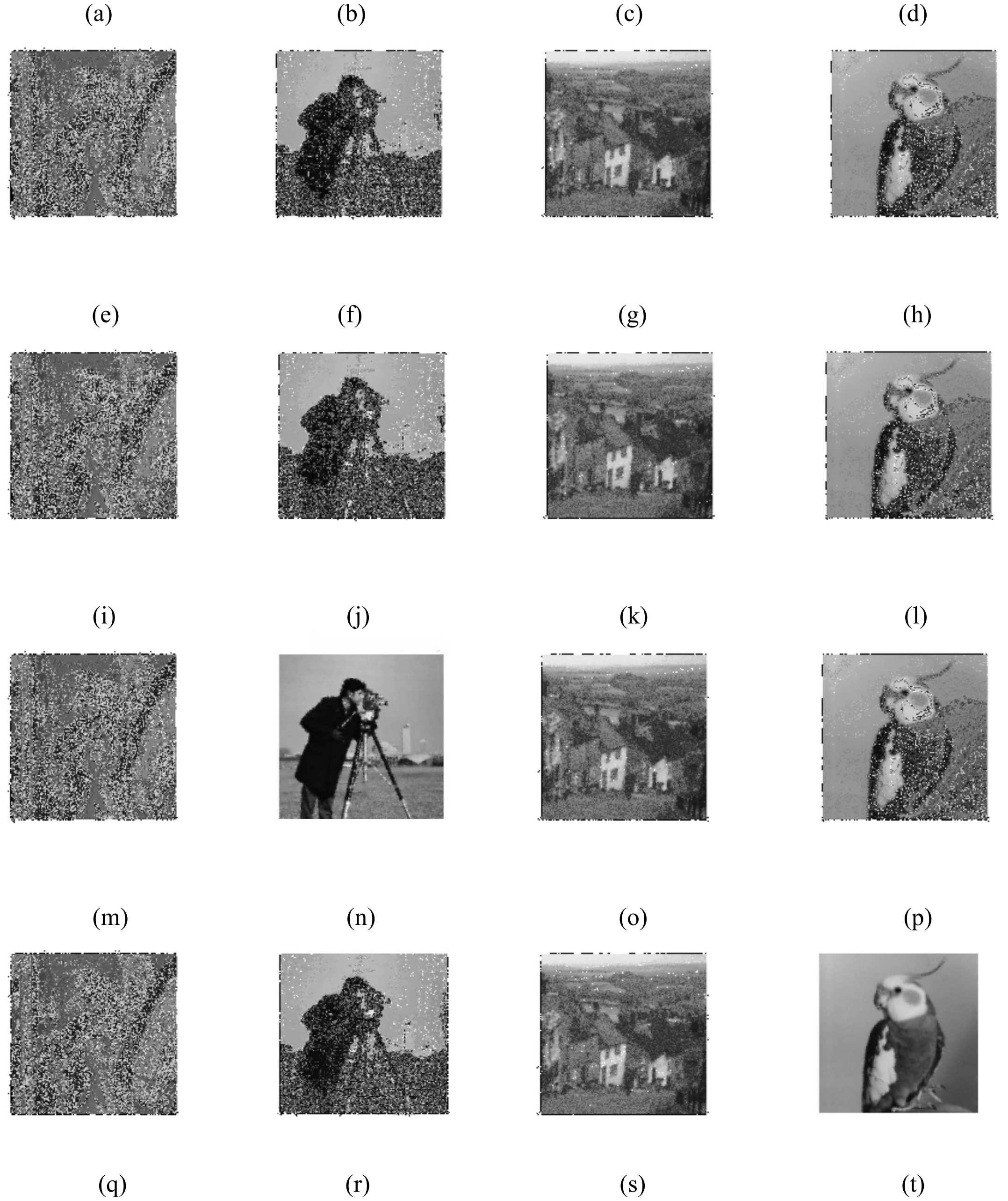


Figure 5: The noisy images corrupted by 75% salt–and–pepper noise: (a)–(d), the restored images via HTTHSLS: (e)–(h), HTTWYL: (i)–(l), HTHP : (m)–(p), and MTTHSLS: (q)–(t)

In this analysis, the images of Lena, Cameraman, and Goldhill, each with dimensions of 256×256 pixels and represented in grayscale, have been used as test cases. Additionally, the Wolfe line search condition and the methods' stopping criteria have been upheld by those established in the first part of our numerical tests. As illustrated in Figures 3–5, all three methods appear to effectively reconstruct the images. To quantitatively evaluate the results, we analyze the computation time (CPUT in seconds), the relative error (RelErr) [6] expressed as a percentage, and the peak signal-to-noise ratio (PSNR) [10] measured in decibels (dB) of the restored image. These metrics are defined as follows:

$$RelErr = 100\frac{||X^{*} - X||}{||X||}, \qquad PSNR = 10\ \log_{10}\frac{255^{2}}{\frac{1}{M \times N}||X^{*} - X||_{F}^{2}},$$

Where X^{*} represents the recovered image. The results can be found in Tables 3–5. As seen in Tables 4 and 5, MTTHSLS is better than the other (image reconstruction) methods from the viewpoints of RelErr and PSNR.

Table 3. Image restoration outputs based on CPUT

Noise	Method	Lena	Cameraman	Goldhill	Bird
25%	MTTHSLS	8.1460	8.1256	8.0166	5.5352
	HTTHSLS	6.5480	6.5470	5.0066	5.1289
	HTTWYL	6.5337	6.4952	5.0044	5.0952
	HTHP	6.5017	6.5451	5.0392	5.1363
50%	MTTHSLS	16.5842	16.4290	13.3039	10.4418
	HTTHSLS	13.1580	13.0136	9.9261	10.0028
	HTTWYL	13.1383	13.1547	9.9237	10.0369
	HTHP	12.9794	13.0835	9.9152	10.0384
75%	MTTHSLS	24.2626	28.4363	19.4133	15.0536
	HTTHSLS	23.9058	23.9368	19.5078	14.9095
	HTTWYL	24.0137	24.4501	19.5736	14.9533
	HTHP	23.9225	23.9974	19.6339	14.9803

Table 4. Image restoration outputs based on RelErr

Noise	Method	Lena	Cameraman	Goldhill	Bird
25%	MTTHSLS	0.7950	0.9598	0.7690	0.3114
	HTTHSLS	0.7996	0.9637	0.7666	0.3177
	HTTWYL	0.7995	0.9637	0.7666	0.3177
	HTHP	0.7995	0.9636	0.7666	0.3177
50%	MTTHSLS	1.4214	1.7816	1.4153	0.5281
	HTTHSLS	1.4240	1.7726	1.4053	0.5507
	HTTWYL	1.4239	1.7722	1.4043	0.5508
	HTHP	1.4240	1.7716	1.4032	0.5507
75%	MTTHSLS	2.2273	2.7163	2.2201	0.8430
	HTTHSLS	2.3435	2.7330	2.3133	0.9136
	HTTWYL	2.3426	2.7316	2.3096	0.9138
	HTHP	2.3440	2.7345	2.3122	0.9136

Table 5. Image restoration outputs based on PSNR

Noise	Method	Lena	Cameraman	Goldhill	Bird
25%	MTTHSLS	34.7438	31.6461	33.1316	42.2355
	HTTHSLS	34.7114	31.6200	33.1398	42.1594
	HTTWYL	34.7126	31.6201	33.1398	42.1594
	HTHP	34.7129	31.6217	33.1417	42.1593
50%	MTTHSLS	30.2496	27.6687	29.4931	37.5014
	HTTHSLS	30.2551	27.6842	29.4380	37.2263
	HTTWYL	30.2582	27.6864	29.4422	37.2258
	HTHP	30.2577	27.6879	29.4488	37.2265
75%	MTTHSLS	26.9772	29.4016	26.9604	34.4020
	HTTHSLS	26.7281	29.4380	26.6662	33.8104
	HTTWYL	26.7293	24.4501	26.6780	33.8093
	HTHP	26.7255	24.4426	26.6714	33.8110

4. Challenges in Existing CG Methods and Our Contributions

The findings of this study offer several key insights for practitioners and decision-makers involved in computational optimization and image restoration applications:

MTTHSLS: A Reliable Solution for Noisy Image Restoration:

The proposed MTTHSLS algorithm demonstrates superior performance in high-noise environments, especially in restoring images with up to 75% salt-and-pepper corruption. This makes it a reliable choice for applications where data degradation occurs often, such as surveillance, medical imaging, or remote sensing systems. Managers overseeing these systems can consider MTTHSLS a dependable preprocessing tool to enhance downstream analytics or decision-making quality.

Reduced computational cost:

Numerical results on the CUTEr benchmark set show that MTTHSLS outperforms existing three-term CG methods in terms of convergence speed and total function evaluations. For operations teams managing large-scale optimization tasks—such as resource allocation, machine learning training, or logistics modeling—this results in lower computational cost, energy consumption, and turnaround time.

Flexibility in implementation:

Since the proposed method maintains the sufficient descent property independently of line search techniques, it allows greater flexibility in tuning or integrating with existing solvers. Managers or engineers can deploy this method with minimal structural changes, reducing adoption barriers and enabling seamless integration into existing software ecosystems.

Scalability and reliability:

The algorithm's strong theoretical foundations, combined with consistent empirical performance across a wide variety of test functions, suggest its applicability to many problems beyond image restoration. This positions MTTHSLS as a versatile optimization tool capable of supporting robust decision-making in various fields, such as finance, engineering design, and data science.

In summary, the proposed MTTHSLS method offers a blend of theoretical rigor and practical effectiveness. Its adoption can improve both the efficiency and reliability of systems that depend on large-scale optimization or sensitive image data reconstruction.

5. Conclusions

Motivated by the research conducted by Abubakr et al. on the creation of three-term formulas for CG methods through a reformulation of the BFGS direction formula, we introduce a new three-term CG method that guarantees sufficient descent conditions without relying on line search. The global convergence characteristics of this method were examined under Wolfe line search conditions. To assess the method's performance in comparison to similar techniques, we carried out numerical experiments using the CUTER benchmark problems, as well as for salt and pepper noise reduction. The outcomes of these experiments demonstrate that our method delivers notable efficiency.

Acknowledgments (All sources of funding for the study must be disclosed)

This research was supported by the Research Council of Semnan University. The authors thank the anonymous reviewer for his/her valuable comments and suggestions that helped to improve the quality of this work. They are grateful to Professor Michael Navon for providing the line search code.

Declarations

Conflict of interest: The authors declare that they have no Conflict of interest.

References

[1] Abubakar, A. B., Kumam, P., Malik, M., Chaipunya, P., & Ibrahim, A. H. (2021). A hybrid FR-DY conjugate gradient algorithm for unconstrained optimization with application in portfolio selection. *AIMS Mathematics*, 6(6), 6506–6527.

[2] Al-Baali, M., & Grandinetti, L. (2012). On the behavior of damped quasi-Newton methods for unconstrained optimization. *Iranian Journal of Operations Research*, 3(1), 1–10.

[3] Aminifard, Z., & Babaie-Kafaki, S. (2019). An optimal parameter choice for the Dai–Liao family of conjugate gradient methods by avoiding a direction of the maximum magnification by the search direction matrix. *4OR*, 17(3), 317–330.

[4] Cao, J., & Wu, J. (2020). A conjugate gradient algorithm and its applications in image restoration. *Applied Numerical Mathematics*, 152, 243–252.

[5] Dolan, E. D., & Moré, J. J. (2002). Benchmarking optimization software with performance profiles. *Mathematical Programming*, 91(2), 201–213.

[6] Esmaeili, H., Shabani, S., & Kimiaei, M. (2019). A new generalized shrinkage conjugate gradient method for sparse recovery. *Calcolo*, 56(1), 1–38.

[7] Gould, N. I., Orban, D., & Toint, P. L. (2021). CUTEr and SifDec: A constrained and unconstrained testing environment, revisited. *ACM Transactions on Mathematical Software*, 29(4), 373–394.

[8] Hager, W. W., & Zhang, H. (2006). A survey of nonlinear conjugate gradient methods. *Pacific Journal of Optimization*, 2(1), 35–58.

[9] Hager, W. W., & Zhang, H. (2006). Algorithm 851: CG_DESCENT, a conjugate gradient method with guaranteed descent. *ACM Transactions on Mathematical Software*, 32(1), 113–137.

[10] Huang, G., Yu, J., & Zhou, Y. (2010). A descent spectral conjugate gradient method for impulse noise removal. *Applied Mathematics Letters*, 23(5), 555–560.

[11] Ibrahim, Y. I., & Khudhur, H. M. (2022). Modified three-term conjugate gradient algorithm and its applications in image restoration. *Indonesian Journal of Electrical Engineering and Computer Science*, 28(3), 1510–1517.

[12] Kim, H., Wang, C., Byun, H., Hu, W., Kim, S., Jiao, Q., & Lee, T. H. (2023). Variable three-term conjugate gradient method for training artificial neural networks. *Neural Networks*, 159, 125–136.

[13] Kumam, P., Abubakar, A. B., Malik, M., Ibrahim, A. H., Pakkaranang, N., & Panyanak, B. (2023). A hybrid HS-LS conjugate gradient algorithm for unconstrained optimization with applications in motion control and image recovery. *Journal of Computational and Applied Mathematics,* 433, 115304.

[14] Liu, Y., Zhu, Z., & Zhang, B. (2022). Two sufficient descent three-term conjugate gradient methods for unconstrained optimization problems with applications in compressive sensing. *Journal of Applied Mathematics and Computing*, 68(3), 1787–1816.

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

[16] Mousavi, A., Esmaeilpour, M., & Sheikhahmadi, A. (2024). Two efficient three-term conjugate gradient methods for impulse noise removal from medical images. *Multimedia Tools and Applications*, 83(28), 43685–43703.

[17] Sun, W., & Yuan, Y. X. (2006). Optimization theory and methods: Nonlinear programming. Springer.

[18] Yuan, G., Li, T., & Hu, W. (2019). A conjugate gradient algorithm and its application in large-scale optimization problems and image restoration. Journal of Inequalities and Applications, 2019(1), 247.

[19] Yuan, G., Li, T., & Hu, W. (2020). A conjugate gradient algorithm for large-scale nonlinear equations and image restoration problems. Applied Numerical Mathematics, 147, 129–141.

[20] Yuan, G., Meng, Z., & Li, Y. (2016). A modified Hestenes and Stiefel conjugate gradient algorithm for large-scale nonsmooth minimizations and nonlinear equations. *Journal of Optimization Theory and Applications*, 168(1), 129–152.

[21] Zhang, L., Zhou, W., & Li, D. H. (2006). A descent modified Polak–Ribière–Polyak conjugate gradient method and its global convergence. *IMA Journal of Numerical Analysis*, 26(3), 629–640.

[22] Zhang, L., Zhou, W., & Li, D. H. (2007). Some descent three-term conjugate gradient methods and their global convergence. *Optimization Methods and Software*, 22(4), 697–711.

[23] Zhang, X., & Yang, Y. (2024). A new hybrid conjugate gradient method close to the memoryless BFGS quasi-Newton method and its application in image restoration and machine learning. *AIMS Mathematics*, 9(12), 27535–27556.