Mathematics

http://www.aimspress.com/journal/Math

AIMS Mathematics, 8(12): 28791–28807.
DOI:10.3934/math.20231475
Received: 31 August 2023
Revised: 10 October 2023
Accepted: 15 October 2023
Published: 23 October 2023

Research article

A new conjugate gradient method with a restart direction and its application in image restoration

Yixin Li, Chunguang Li*, **Wei Yang and Wensheng Zhang**

School of Mathematics and Information Science, North Minzu University, Yinchuan, 750021, China

* **Correspondence:** Email: cglizd@hotmail.com.

Abstract: We established a new conjugate gradient method with an efficient restart direction for solving large scale unconstrained optimization problems. The modified method was proposed under the Polak-Ribière-Polyak conjugate gradient method. Under the strong Wolfe line search, the search direction of the new method was sufficiently descent and its global convergence property could be proved. Compared with other methods having good numerical performances, numerical experiments and image restorations showed that the modified method was more effective.

Keywords: unconstrained optimization; conjugate gradient method; strong Wolfe line search; global convergence; numerical experiment; image restoration
Mathematics Subject Classification: 65K10, 68U10, 90C06

1. Introduction

The conjugate gradient method is an efficient method for solving the large scale nonlinear unconstrained optimization problem

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

where $f : R^n \to R$ is a continuously differentiable function and the gradient of f at x is denoted by $g(x)$ for $x \in R^n$. The iteration formulas of the conjugate gradient methods are obtained by

$$x_{k+1} = x_k + \alpha_k d_k, k \geq 0, \tag{1.2}$$

$$d_k = \begin{cases} -g_k, & k = 0, \\ -g_k + \beta_k d_{k-1}, & k \geq 1. \end{cases} \tag{1.3}$$

Starting from an initial guess, $x_0 \in R^n$, $\alpha_k > 0$ is a step-length that was obtained by some line searches, and d_k is the descent direction. β_k is the conjugate parameter, and different β_k lead to different conjugate methods (CGMs). The classical and famous CGMs include the Hestenes-Stiefel (HS) method [1], Polak-Ribière-Polyak (PRP) method [2, 3], Fletcher-Reeves (FR) method [4], Liu-Storey (LS) method [5] and Dai-Yuan (DY) method [6]. Their conjugate parameters β_k are specified as follows:

$$\beta_k^{HS} = \frac{g_k^T (g_k - g_{k-1})}{d_{k-1}^T (g_k - g_{k-1})},$$

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

$$\beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2},$$

$$\beta_k^{LS} = \frac{g_k^T (g_k - g_{k-1})}{-g_{k-1}^T d_{k-1}},$$

$$\beta_k^{DY} = \frac{\|g_k\|^2}{d_{k-1}^T (g_k - g_{k-1})}.$$

HS, PRP, FR, LS and DY are equivalent when the objective function is a strictly convex quadratic function and are under the exact line search. However, they have large differences whenever they use inexact line search and the objective function is a general non-convex function. Usually, PRP, HS and LS have good numerical performances because they have an approximate restart feature when jamming occurs, but the above three methods may not satisfy the convergence property. On the contrary, the numerators of β_k^{FR} and β_k^{DY} are positive, so FR and DY have good convergence properties under mild conditions, while, their numerical performances are not so good.

In the past decades, many outstanding experts have proposed some efficient CGMs that combine the advantages of PRP and FR methods. Undoubtedly, the proposed methods have good numerical performances and convergence properties.

In 2006, Wei, Yao and Liu [7] gave a modified PRP method which we call the WYL method. The β_k in this method is written as:

$$\beta_k^{WYL} = \frac{g_k^T \left(g_k - \frac{\|g_k\|}{\|g_{k-1}\|} g_{k-1}\right)}{\|g_{k-1}\|^2},$$

and the WYL method inherits the advantages of the PRP method, which has good numerical and theoretical properties.

A modified WYL method is given by Dai et al. [8] and it is called the DPRP method, which is designed by

$$\beta_k^{DPRP} = \frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|} \left|g_k^T g_{k-1}\right|}{\|g_{k-1}\|^2 + \mu \left|g_k^T d_{k-1}\right|},$$

where $\mu > 1$. The DRPR method satisfies the sufficient descent property independent of any line search, and it is globally convergent under the standard Wolfe line search [9, 10].

Zhu et al. [11] relied on the restart conditions to come up with two conjugate gradient methods (DDY1,DDY2). The conjugate parameters of the DDY1 and DDY2 methods are written as β_k^1 and β_k^2, respectively,

$$\beta_k^1 = \begin{cases} \dfrac{g_k^T\left(g_k - \frac{\mu_1\left(g_k^T d_{k-1}\right)^2}{\|g_k\|\|g_{k-1}\|\|d_{k-1}\|^2} g_{k-1}\right)}{d_{k-1}^T(g_k - g_{k-1})}, & g_k^T g_{k-1} \geq 0, \\ \dfrac{g_k^T\left(g_k + \frac{\mu_1\left(g_k^T d_{k-1}\right)^2}{\|g_k\|\|g_{k-1}\|\|d_{k-1}\|^2} g_{k-1}\right)}{d_{k-1}^T(g_k - g_{k-1})}, & g_k^T g_{k-1} < 0, \end{cases}$$

$$\beta_k^2 = \begin{cases} \dfrac{g_k^T\left(g_k - \frac{g_k^T d_{k-1}}{\|d_{k-1}\|^2} d_{k-1}\right)}{d_{k-1}^T(g_k - g_{k-1}) + \mu_2 g_k^T d_{k-1}}, & g_k^T d_{k-1} \geq 0, \\ 0, & g_k^T d_{k-1} < 0, \end{cases}$$

the DDY1 and DDY2 methods are sufficiently descent under the standard Wolfe line search when the parameters satisfy $\mu_1 \in [0, 1]$, $\mu_2 > 0$.

In 2023, Liu at el. [12] based on [11, 13] gave two modified HS methods (EHS-RD1, EHS-RD2), the descent direction of the EHS-RD1 is given by:

$$d_k = \begin{cases} -g_k, & (k = 0), \\ -g_k + \beta_k^{EHS1} d_{k-1}, & (k \geq 1) \text{ if } g_k^\top (g_k - g_{k-1}) \geq 0, \\ -g_k + \xi_1 \frac{g_k^\top g_{k-1}}{\|g_{k-1}\|^2} g_{k-1}, & (k \geq 1) \text{ if } g_k^\top (g_k - g_{k-1}) < 0, \end{cases}$$

where $\beta_k^{EHS1} = \frac{\|g_k\|^2 - \mu_1 \frac{\|g_k\|}{\|g_k - g_{k-1}\|} g_k^\top (g_k - g_{k-1})}{d_{k-1}^\top (g_k - g_{k-1})}$ and $\mu_1 \in [0, 1]$, $\xi_1 \in [0, 1)$.

The descent direction of the EHS-RD2 is given by:

$$d_k = \begin{cases} -g_k, & (k = 0), \\ -g_k + \beta_k^{EHS2} d_{k-1}, & (k \geq 1) \text{ if } g_k^\top d_{k-1} \geq 0, \\ -g_k + \xi_2 \frac{g_k^\top d_{k-1}}{\|d_{k-1}\|^2} d_{k-1}, & (k \geq 1) \text{ if } g_k^\top d_{k-1} < 0, \end{cases}$$

where $\beta_k^{EHS2} = \frac{\|g_k\|^2 - \frac{\left(g_k^\top (g_k - g_{k-1})\right)^2}{\|g_k - g_{k-1}\|^2}}{d_{k-1}^\top y_{k-1} + \mu_2 g_k^\top d_{k-1}}$ and $\mu_2 > 0$, $\xi_2 \in [0, 1)$.

The conjugate gradient methods EHS-RD1 and EHS-RD2 all have good numerical performances under the standard Wolfe line search.

Inspired by the numerator of β_k^{WYL} in [7] and the denominator of β_k^{DPRP} in [8], we propose a new parameter β_k^{LLYZ} as

$$\beta_k^{LLYZ} = \frac{\|g_k\|^2 - \mu_1 \frac{\|g_k\|}{\|g_{k-1}\|} g_k^T g_{k-1}}{\|g_{k-1}\|^2 + \mu_2 \left|g_k^\top d_{k-1}\right|}, \tag{1.4}$$

where $\mu_1 \in [0, 1]$, $\mu_2 > 0$.

Inspired by the restart conditions of β_k^1 and β_k^2 in [11] and the restart directions of EHS-RD1, EHS-RD2 in [12], we conduct a descent direction d_k with a restart direction, which can be written as

$$d_k^{LLYZ} = \begin{cases} -g_k, & (k = 0) \\ -g_k + \beta_k^{LLYZ} d_{k-1}, & (k \geq 1) \text{ if } g_k^{\mathrm{T}} g_{k-1} \geq 0, \\ -g_k + \xi \frac{g_k^{\mathrm{T}} g_{k-1}}{\|g_{k-1}\|^2} g_{k-1}, & (k \geq 1) \text{ if } g_k^{\mathrm{T}} g_{k-1} < 0, \end{cases} \tag{1.5}$$

where $\xi \in (0, 1)$.

We use the strong Wolfe line search where the formal express is as follows

$$\begin{cases} f(x_k + \alpha_k d_k) \le f(x_k) + \delta \alpha_k g_k^T d_k, \\ \left| g(x_k + \alpha_k d_k)^T d_k \right| \le \sigma \left| g_k^T d_k \right|, \end{cases} \tag{1.6}$$

where $0 < \delta < \sigma < 1$.

For convenience's sake, we call the iteration method which is decided by β_k^{LLYZ} and d_k^{LLYZ} the LLYZ algorithm.

In section two, we mainly introduce the LLYZ algorithm and prove its sufficient descent and global convergence property. In section three, we conduct some numerical experiments to demonstrate the good numerical performance of the LLYZ algorithm. In section four, the effectiveness of the LLYZ algorithm is verified by solving image restoration problems. In section five, a conclusion for this work is made.

2. LLYZ algorithm and global convergence

Now, based on the search direction (1.5) and the strong Wolfe line search, we formally present the LLYZ algorithm as follows.

Algorithm : LLYZ

Step 0. (Initialization) Given an initial point $x_0 \in R^n$, $\xi \in (0, 1)$ and set $k := 0$, $\varepsilon > 0$, $d_0 = -g_0$.
Step 1. if $\|g_0\| \le \varepsilon$, then stop.
Step 2. Compute a step-length α_k by the strong Wolfe line search (1.6).
Step 3. Generate the x_{k+1} by (1.2).
Step 4. Compute the β_k by the formula (1.4), and generate d_{k+1} by the formula (1.5).
Step 5. Set $k := k + 1$ and go to Step 1.

Next, we analyze the descent property and global convergence of the LLYZ algorithm.

Lemma 2.1. The parameter β_k^{LLYZ} satisfies the following formula from (1.4) and (1.5):

$$0 \le \beta_k^{LLYZ} \le \beta_k^{FR}. \tag{2.1}$$

Proof.

$$\beta_k^{LLYZ} = \frac{\|g_k\|^2 - \mu_1 \frac{\|g_k\|}{\|g_{k-1}\|} g_k^T g_{k-1}}{\|g_{k-1}\|^2 + \mu_2 \left| g_k^T d_{k-1} \right|} \ge \frac{(1-\mu_1)\|g_k\|^2}{\|g_{k-1}\|^2 + \mu_2 \left| g_k^T d_{k-1} \right|} \ge 0, \tag{2.2}$$

$$
\begin{aligned}
\beta_k^{FR}-\beta_k^{LLYZ} &= \frac{\|g_k\|^2}{\|g_{k-1}\|^2}-\frac{\|g_k\|^2-\mu_1\frac{\|g_k\|}{\|g_{k-1}\|}g_k^Tg_{k-1}}{\|g_{k-1}\|^2+\mu_2\mid g_k^Td_{k-1}|}\\
&\geqslant \frac{\|g_k\|^2}{\|g_{k-1}\|^2}-\frac{\|g_k\|^2-\mu_1\frac{\|g_k\|}{\|g_{k-1}\|}g_k^Tg_{k-1}}{\|g_{k-1}\|^2}\\
&= \frac{\mu_1\frac{\|g_k\|}{\|g_{k-1}\|}g_k^Tg_{k-1}}{\|g_{k-1}\|^2}\\
&\geqslant 0,
\end{aligned}
\tag{2.3}
$$

combine formulas (2.2) and (2.3), then we have

$$
0\leq\beta_k^{LLYZ}\leq\beta_k^{FR}. \tag{2.4}
$$

□

Lemma 2.2. [14] Consider the CGM yield by (1.2) and (1.3), where the step-length α_k satisfies the strong Wolfe line search (1.6) and parameter $|\beta_k|\leq\beta_k^{\mathrm{FR}}$. If $0<\sigma<\frac{1}{2}$, then the CGMs is sufficiently descent with

$$
-\frac{1}{1-\sigma}\leq\frac{g_k^Td_k}{\|g_k\|^2}\leq-\frac{1-2\sigma}{1-\sigma},\forall k\geq 0. \tag{2.5}
$$

The next Lemma proves that the LLYZ algorithm is sufficiently descent, and Lemma 2.2 plays a very important role in proving the descent property of LLYZ.

Lemma 2.3. If the direction d_k is yielded by (1.4)-(1.5) and the step-length α_k satisfies the strong Wolfe line search (1.6), then the following equality holds for any k:

$$
\frac{g_k^Td_k}{\|g_k\|^2}<0. \tag{2.6}
$$

That is to say, the LLYZ algorithm is sufficiently descent.

Proof.
When $k=0$, then $d_0=-g_0$ and it's easy to conclude

$$
\frac{g_0^Td_0}{\|g_0\|^2}=-1, \tag{2.7}
$$

thus, relation (2.6) holds.

Next, we assume that $g_{k-1}^Td_{k-1}<0$ holds for all $k\geqslant 1$, and from the second inequality of the strong Wolfe line search (1.6) we deduce

$$
\left|g_k^Td_{k-1}\right|\leq\sigma\left|g_{k-1}^Td_{k-1}\right|=-\sigma g_{k-1}^Td_{k-1}, \tag{2.8}
$$

then we prove (2.6) is true for $k\geq 1$ by the following two situations:

Case (i) When $k \geq 1$ and $g_k^T g_{k-1} < 0$, then $d_k = -g_k + \xi \frac{g_k^T g_{k-1}}{\|g_{k-1}\|^2} g_{k-1}$ and we can deduce

$$\begin{aligned} \frac{g_k^T d_k}{\|g_k\|^2} &= \frac{-\|g_k\|^2 + \xi \frac{(g_k^T g_{k-1})^2}{\|g_{k-1}\|^2}}{\|g_k\|^2} \\ &\geq -\frac{\|g_k\|^2}{\|g_k\|^2} \\ &= -1, \end{aligned} \tag{2.9}$$

$$\begin{aligned} \frac{g_k^T d_k}{\|g_k\|^2} &= \frac{-\|g_k\|^2 + \xi \frac{(g_k^T g_{k-1})^2}{\|g_{k-1}\|^2}}{\|g_k\|^2} \\ &\leq \frac{-\|g_k\|^2 + \xi \frac{\|g_k\|^2 \|g_{k-1}\|^2}{\|g_{k-1}\|^2}}{\|g_k\|^2} \\ &= -(1-\xi), \end{aligned} \tag{2.10}$$

therefore,

$$-1 \leq \frac{g_k^T d_k}{\|g_k\|^2} \leq -(1-\xi), \tag{2.11}$$

and relation (2.6) holds.

Case (ii) When $k \geq 1$ and $g_k^T g_{k-1} \geq 0$, then $d_k = -g_k + \beta_k^{LLYZ} d_{k-1}$

$$\begin{aligned} \frac{g_k^T d_k}{\|g_k\|^2} &= \frac{g_k^T \left(-g_k + \beta_k^{LLYZ} d_{k-1}\right)}{\|g_k\|^2} \\ &= -1 + \beta_k^{LLYZ} \frac{g_k^T d_{k-1}}{\|g_k\|^2}, \end{aligned}$$

together with $\beta_k^{LLYZ} \geq 0$ in (2.4), we have

$$-1 - \beta_k^{LLYZ} \frac{|g_k^T d_{k-1}|}{\|g_k\|^2} \leq \frac{g_k^T d_k}{\|g_k\|^2} \leq -1 + \beta_k^{LLYZ} \frac{|g_k^T d_{k-1}|}{\|g_k\|^2}.$$

The above equation together with $\beta_k^{LLYZ} \leq \beta_k^{FR}$ in (2.4), we have

$$\begin{aligned} -1 - \beta_k^{LLYZ} \frac{|g_k^T d_{k-1}|}{\|g_k\|^2} &\geq -1 - \beta_k^{FR} \frac{|g_k^T d_{k-1}|}{\|g_k\|^2} \\ &= -1 - \frac{|g_k^T d_{k-1}|}{\|g_{k-1}\|^2}, \end{aligned}$$

$$\begin{aligned} -1 + \beta_k^{LLYZ} \frac{|g_k^T d_{k-1}|}{\|g_k\|^2} &\leq -1 + \beta_k^{FR} \frac{|g_k^T d_{k-1}|}{\|g_k\|^2} \\ &= -1 + \frac{|g_k^T d_{k-1}|}{\|g_{k-1}\|^2}, \end{aligned}$$

namely,

$$-1-\frac{|g_k^T d_{k-1}|}{||g_{k-1}||^2} \le \frac{g_k^T d_k}{||g_k||^2} \le -1+\frac{|g_k^T d_{k-1}|}{||g_{k-1}||^2}, \tag{2.12}$$

together with (2.8), we get that

$$-1+\sigma\frac{g_{k-1}^T d_{k-1}}{||g_{k-1}||^2} \le \frac{g_k^T d_k}{||g_k||^2} \le -1-\sigma\frac{g_{k-1}^T d_{k-1}}{||g_{k-1}||^2}. \tag{2.13}$$

Case (a) If d_{k-1} is generated by $d_{k-1} = -g_{k-1} + \beta_{k-1}^{LLYZ} d_{k-2}$, together with (2.5) and on the left side of (2.13), we have

$$\begin{aligned} -1+\sigma\frac{g_{k-1}^T d_{k-1}}{||g_{k-1}||^2} &\ge -1+\sigma(-\frac{1}{1-\sigma}) \\ &= -\frac{1}{1-\sigma}, \end{aligned} \tag{2.14}$$

and on the right side of (2.13) we have

$$\begin{aligned} -1-\sigma\frac{g_{k-1}^T d_{k-1}}{||g_{k-1}||^2} &\le -1-\sigma(-\frac{1}{1-\sigma}) \\ &= -\frac{1-2\sigma}{1-\sigma} \\ &< 0, \end{aligned} \tag{2.15}$$

then we obtain

$$-\frac{1}{1-\sigma} \le \frac{g_k^T d_k}{||g_k||^2} < 0. \tag{2.16}$$

Case (b) If d_{k-1} is generated by $d_{k-1} = -g_{k-1} + \xi \frac{g_{k-1}^T g_{k-2}}{||g_{k-2}||^2} g_{k-2}$, together with (2.7), (2.9) and (2.13), we have

$$\begin{aligned} \frac{g_k^T d_k}{||g_k||^2} &\le -1-\sigma\frac{g_{k-1}^T d_{k-1}}{||g_{k-1}||^2} \\ &\le \sigma - 1 \\ &< 0, \end{aligned} \tag{2.17}$$

then relation (2.6) holds. □

Next, in order to prove LLYZ's global convergence, we need to use the following assumptions as well as the Zoutendijk condition [15]:

Assumption :
(H1) The objective function $f(x)$ is bounded on the level set $\Lambda = \{x \in R^n \mid f(x) \le f(x_0)\}$.
(H2) In the neighborhood U of Λ, the objective function $f(x)$ is continuously differentiable and the gradient g_k is Lipschitz continuous; that is, there exists a constant $L > 0$ such that

$$||g(x) - g(y)|| \le L||x - y||, \forall x, y \in U.$$

Lemma 2.4. [15] Suppose that Assumptions (H1) and (H2) hold and the iteration consists of (1.2) and (1.3). If the direction d_k is descent and the step-length α_k conforms to the Wolfe line search, then

$$\sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} < +\infty, \tag{2.18}$$

and if the sufficient descent condition $g_k^T d_k \leq -c \|g_k\|^2$ $(c > 0)$ is satisfied, then (2.18) can be written as

$$\sum_{k=0}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} < \infty. \tag{2.19}$$

Now, we prove the global convergence of the LLYZ algorithm.

Theorem 2.1. If Assumptions (H1) and (H2) hold and the iteration sequence $\{x_k\}$ is generated by the LLYZ algorithm, then the LLYZ algorithm is globally convergent, namely,

$$\lim_{k\to\infty} inf \|g_k\| = 0. \tag{2.20}$$

Proof.
We give a proof by contradiction for its convergence, that is, there exists a constant $\gamma > 0$, such that $\|g_k\|^2 \geq \gamma$, $\forall k \geq 0$.

Case (i) If the search direction derives from $d_k = -g_k + \xi \frac{g_k^T g_{k-1}}{\|g_{k-1}\|^2} g_{k-1}$ and $\xi \in (0, 1)$, we have

$$\begin{aligned} \|d_k\|^2 &= \|g_k\|^2 - 2\xi \frac{\left(g_k^T g_{k-1}\right)^2}{\|g_{k-1}\|^2} + \xi^2 \frac{\left(g_k^T g_{k-1}\right)^2}{\|g_{k-1}\|^4} \|g_{k-1}\|^2 \\ &= \|g_k\|^2 - \xi (2-\xi) \frac{\left(g_k^T g_{k-1}\right)^2}{\|g_{k-1}\|^2} \\ &< \|g_k\|^2 . \end{aligned} \tag{2.21}$$

Case (ii) If the search direction derives from $d_k = -g_k + \beta_k^{LLYZ} d_{k-1}$, we have

$$\|d_k\|^2 = \|g_k\|^2 - 2\beta_k^{LLYZ} g_k^T d_{k-1} + \left(\beta_k^{LLYZ}\right)^2 \|d_{k-1}\|^2 . \tag{2.22}$$

By Lemma 2.1, we can deduce:

$$\begin{aligned} \|d_k\|^2 &\leq \|g_k\|^2 + 2\beta_k^{LLYZ} \left|g_k^T d_{k-1}\right| + \left(\beta_k^{LLYZ}\right)^2 \|d_{k-1}\|^2 \\ &\leq \|g_k\|^2 + 2\beta_k^{FR} \left|g_k^T d_{k-1}\right| + \left(\beta_k^{FR}\right)^2 \|d_{k-1}\|^2 , \end{aligned} \tag{2.23}$$

and together with the second inequality of the strong Wolfe line search (1.6) and Lemma 2.2, we have

$$
\begin{aligned}
\frac{\|d_k\|^2}{\|g_k\|^4} &\leqslant \frac{1}{\|g_k\|^2} + 2\frac{\left|g_k^T d_{k-1}\right|}{\|g_k\|^2\|g_{k-1}\|^2} + \frac{\|d_{k-1}\|^2}{\|g_{k-1}\|^4} \\
&\leqslant \frac{1}{\|g_k\|^2} + 2\sigma\frac{\left|g_{k-1}^T d_{k-1}\right|}{\|g_k\|^2\|g_{k-1}\|^2} + \frac{\|d_{k-1}\|^2}{\|g_{k-1}\|^4} \\
&\leqslant \frac{1+\sigma}{1-\sigma}\frac{1}{\|g_k\|^2} + \frac{\|d_{k-1}\|^2}{\|g_{k-1}\|^4} \\
&= \Gamma\frac{1}{\|g_k\|^2} + \frac{\|d_{k-1}\|^2}{\|g_{k-1}\|^4} \\
&\leq \sum_{i=1}^{k}\frac{\Gamma}{\|g_i\|^2} + \frac{1}{\|g_0\|^2} \\
&\leq \frac{\Gamma k}{\gamma} + \frac{1}{\gamma} \\
&= \frac{\Gamma k + 1}{\gamma},
\end{aligned}
\tag{2.24}
$$

where $\Gamma = \frac{1+\sigma}{1-\sigma}$, then we can obtain

$$
\frac{\|g_k\|^4}{\|d_k\|^2} \geq \frac{\gamma}{\Gamma k + 1},
\tag{2.25}
$$

where (2.25) contradicts with (2.19). Thus, the LLYZ algorithm is globally convergent. □

3. Numerical experiments

In this section, we test the LLYZ method with the N. Andrei (AN) [16], DDY1 [11], EHS-RD2 [12] and WYL [7] methods. The above four methods proposed by previous researchers all have good numerical performances. We use 92 unconstrained optimization problems to test the above five conjugate gradient methods. The test problems come from [17, 18, 19], and the dimensions of them are from two to 82,000. The testing environment is Matlab2018b, Win10 operating system and Dell desktop computer (Intel (R) Core (TM) i7-10700 CPU@2.90GHz) 16.0GB memory.

All the algorithms use the standard Wolfe line search, and the algorithm termination condition is $\|g_k\| \leq 10^{-6}$ or $Itr > 2000$. Denote "NaN" when $Itr > 2000$ occurs. We set up $\delta = 0.01$, $\sigma = 0.1$ as the parameters of the standard Wolfe line search and $\mu_1 = 0.78$, $\mu_2 = 1.50$, $\xi = 0.05$ for the LLYZ method, while the parameters in AN, DDY1 and EHS-RD2 are the same as their original settings.

We use Dolan and Moré (D-M) performance profiles [20] to visually show the numerical comparison results. Figures 1 and 2 respectively represent the LLYZ performance best compared to other methods in terms of Central Processing Unit (CPU) computation time and iterations. The specific numerical experimental results are shown in Tables 1, 2 and 3.

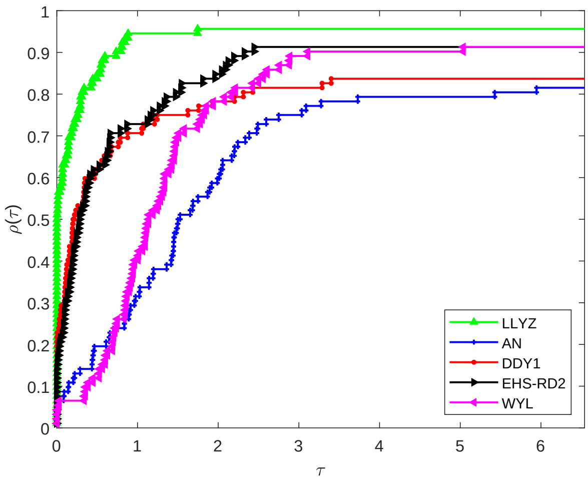


Figure 1. CPU time performance profile.

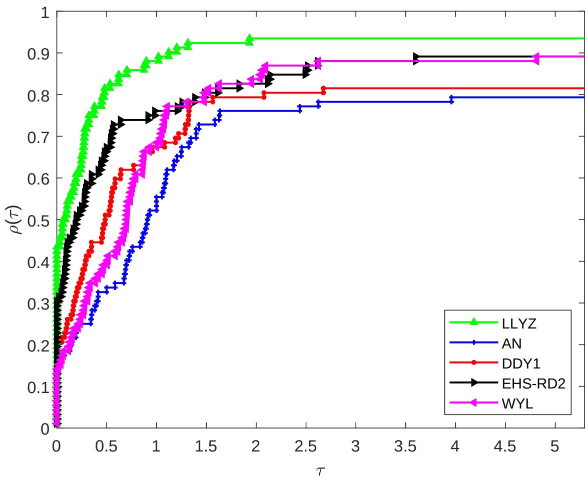


Figure 2. Iterations performance profile.

Table 1. Numerical results of the five method.

	LLYZ	AN	DDY1	EHS-RD2	WYL
Name/n	Tcpu/NG/Itr	Tcpu/NG/Itr	Tcpu/NG/Itr	Tcpu/NG/Itr	Tcpu/NG/Itr
badscp/2	0.00/1.1e+02/28	0.01/1.1e+02/27	0.01/1.1e+02/29	0.00/1.1e+02/28	0.01/2.2e+02/37
cosine/200	0.00/7.2e+01/19	0.01/9.3e+01/29	0.00/8.0e+01/25	0.00/7.0e+01/16	0.00/1.5e+02/35
cosine/800	0.01/6.9e+01/17	0.01/1.2e+02/45	0.01/7.6e+01/20	0.01/6.4e+01/16	0.01/1.2e+02/20
cosine/2000	0.01/8.0e+01/18	0.01/8.7e+01/25	0.01/9.7e+01/31	0.01/8.0e+01/19	0.02/1.8e+02/37
dixmaana/90	0.00/6.8e+01/14	0.03/1.7e+02/28	0.01/7.6e+01/17	0.01/7.2e+01/19	0.01/1.6e+02/27
dixmaana/105	0.00/5.5e+01/11	0.01/1.8e+02/34	0.00/7.1e+01/14	0.01/8.1e+01/16	0.01/2.0e+02/34
dixmaana/120	0.01/7.4e+01/16	0.01/1.3e+02/25	0.00/7.2e+01/14	0.01/7.0e+01/15	0.01/1.3e+02/31
dixmaanb/11400	0.27/9.0e+01/23	0.47/1.5e+02/29	0.21/7.0e+01/13	0.26/8.7e+01/21	0.56/1.9e+02/28
dixmaanb/12000	0.24/7.7e+01/15	0.57/1.8e+02/35	0.26/8.2e+01/16	0.26/8.2e+01/16	0.53/1.7e+02/31
dixmaanc/15600	0.38/8.6e+01/18	0.65/1.5e+02/27	0.38/8.9e+01/22	0.39/9.3e+01/20	0.75/1.7e+02/32
dixmaanc/15000	0.35/8.4e+01/17	0.67/1.6e+02/28	0.35/8.4e+01/18	0.36/8.7e+01/20	0.61/1.5e+02/26
dixmaand/300	0.01/7.1e+01/17	0.04/1.4e+02/28	0.01/7.8e+01/21	0.01/7.4e+01/14	0.04/2.0e+02/33
dixmaand/900	0.02/8.0e+01/13	0.06/1.9e+02/32	0.03/8.3e+01/20	0.02/7.5e+01/14	0.04/1.2e+02/24
dixmaanf/300	0.05/3.7e+02/216	0.16/8.6e+02/193	0.04/3.1e+02/250	0.10/7.8e+02/465	0.05/4.5e+02/129
dixmaanf/1200	0.30/7.8e+02/436	0.58/1.6e+03/372	0.13/3.4e+02/272	0.48/1.3e+03/713	0.41/1.1e+03/281
dixmaanh/3	0.00/6.6e+01/17	0.02/1.8e+02/32	0.00/6.4e+01/19	0.00/7.4e+01/24	0.00/1.3e+02/29
dixmaanh/9	0.01/9.5e+01/44	0.01/2.0e+02/42	0.01/1.1e+02/60	0.01/1.0e+02/48	0.01/1.5e+02/33

Table 2. Numerical results of the five method (continued).

	LLYZ	AN	DDY1	EHS-RD2	WYL
Name/n	Tcpu/NG/Itr	Tcpu/NG/Itr	Tcpu/NG/Itr	Tcpu/NG/Itr	Tcpu/NG/Itr
dixmaanh/12	0.01/1.1e+02/48	0.02/2.1e+02/44	0.01/1.2e+02/61	0.01/9.2e+01/38	0.02/2.9e+02/67
dixmaani/3600	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
dixmaanj/90	0.10/1.8e+03/1034	0.24/3.7e+03/896	0.09/1.5e+03/1388	0.16/2.9e+03/1644	0.09/1.7e+03/420
dixmaanj/108	0.09/1.5e+03/919	0.21/3.4e+03/773	NaN/NaN/NaN	NaN/NaN/NaN	0.17/2.9e+03/722
dixmaanl/1080	0.13/3.9e+02/216	NaN/NaN/NaN	NaN/NaN/NaN	0.16/4.7e+02/291	0.55/1.6e+03/419
dixmaanl/1170	0.12/3.3e+02/195	NaN/NaN/NaN	NaN/NaN/NaN	0.10/2.8e+02/160	NaN/NaN/NaN
dixon3dq/3000	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
dqrtic/240	0.01/1.0e+02/27	0.04/4.8e+02/71	0.01/1.0e+02/27	0.01/1.0e+02/24	0.01/1.4e+02/30
dqrtic/270	0.01/9.6e+01/22	0.03/4.0e+02/65	0.01/1.0e+02/26	0.01/1.0e+02/27	0.01/1.8e+02/34
dqrtic/300	0.01/9.4e+01/19	0.02/3.2e+02/50	0.01/1.1e+02/30	0.01/1.1e+02/28	0.01/1.3e+02/23
edensch/3600	0.06/1.8e+02/49	0.09/2.1e+02/60	0.14/2.1e+02/51	0.08/2.7e+02/62	0.87/1.3e+03/154
edensch/3800	0.06/1.6e+02/60	0.09/2.4e+02/57	0.12/1.7e+02/51	0.11/1.9e+02/46	0.24/3.4e+02/60
edensch/3500	0.01/1.1e+02/49	0.02/2.1e+02/56	0.05/4.5e+02/101	0.01/1.4e+02/49	0.06/5.4e+02/70
eg2/500	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
fletchcr/2000	0.01/1.9e+02/91	0.01/1.6e+02/77	0.03/9.8e+02/133	0.01/2.1e+02/116	0.07/2.5e+03/272
fletchcr/2100	0.01/2.1e+02/112	0.01/2.5e+02/148	0.05/1.6e+03/203	0.01/2.5e+02/149	0.08/2.6e+03/273
fletchcr/2500	0.01/2.5e+02/129	0.01/2.4e+02/143	0.02/7.0e+02/111	0.01/2.6e+02/139	0.02/6.8e+02/90
freuroth/8	0.01/7.2e+02/353	NaN/NaN/NaN	0.01/7.3e+02/205	NaN/NaN/NaN	0.06/3.2e+03/558
freuroth/4	0.01/5.7e+02/293	NaN/NaN/NaN	0.01/5.2e+02/142	NaN/NaN/NaN	0.04/2.2e+03/436
genrose/3	0.01/4.1e+02/221	NaN/NaN/NaN	0.01/3.4e+02/162	0.04/1.8e+03/1083	0.02/1.1e+03/221
genrose/12	0.03/1.3e+03/904	NaN/NaN/NaN	NaN/NaN/NaN	0.07/3.3e+03/1909	0.03/1.8e+03/464
himmelbg/2	0.00/5.0e+00/2	0.01/5.0e+00/2	0.00/5.0e+00/2	0.00/5.0e+00/2	0.00/5.0e+00/2
himmelbg/2000	0.00/1.0e+01/2	0.00/1.0e+01/2	0.00/1.0e+01/2	0.00/1.0e+01/2	0.00/1.0e+01/2
penalty1/800	0.11/1.8e+02/29	0.14/2.2e+02/31	0.15/2.6e+02/43	0.11/2.0e+02/34	1.54/2.8e+03/391
penalty1/1000	0.18/2.1e+02/31	0.21/2.5e+02/36	0.19/2.2e+02/33	0.19/2.3e+02/33	6.14/7.3e+03/1015
penalty1/1500	0.13/7.0e+01/12	0.17/9.5e+01/16	0.13/7.0e+01/12	0.13/7.0e+01/12	0.13/7.0e+01/12
penalty1/2000	0.30/9.4e+01/24	0.36/1.1e+02/22	0.31/9.5e+01/22	0.35/1.1e+02/24	0.82/2.5e+02/41
quartc/1200	0.03/1.2e+02/29	0.09/3.3e+02/54	0.03/1.4e+02/42	0.03/1.3e+02/37	0.05/2.2e+02/50
quartc/1600	0.05/1.4e+02/44	0.09/2.6e+02/51	0.05/1.4e+02/45	0.05/1.3e+02/39	0.08/2.3e+02/51
quartc/1800	0.05/1.3e+02/32	0.21/5.7e+02/90	0.06/1.5e+02/52	0.05/1.2e+02/28	0.08/2.3e+02/49
quartc/2000	0.05/1.3e+02/37	0.15/3.7e+02/53	0.05/1.4e+02/38	0.05/1.3e+02/31	0.08/2.0e+02/48
woods/100	0.02/8.1e+02/460	NaN/NaN/NaN	NaN/NaN/NaN	0.05/2.6e+03/1504	0.05/2.8e+03/647
woods/200	0.02/8.2e+02/443	NaN/NaN/NaN	0.01/6.1e+02/336	0.03/1.6e+03/889	0.02/8.5e+02/195
woods/300	0.02/8.9e+02/526	NaN/NaN/NaN	0.03/1.3e+03/976	0.04/1.8e+03/1044	0.03/1.5e+03/342
bdexp/20	0.00/7.0e+00/2	0.01/2.3e+01/5	0.00/7.0e+00/2	0.00/7.0e+00/2	0.00/7.0e+00/2
bdexp/2000	0.00/8.0e+00/2	0.00/1.9e+01/3	0.00/8.0e+00/2	0.00/8.0e+00/2	0.00/8.0e+00/2
exdenschnb/600	0.00/7.4e+01/15	0.01/2.0e+02/39	0.00/1.0e+02/38	0.00/7.6e+01/16	0.00/1.4e+02/27
exdenschnb/900	0.00/7.9e+01/19	0.01/2.3e+02/37	0.00/8.2e+01/22	0.00/7.7e+01/16	0.00/1.6e+02/27
exdenschnb/1000	0.00/7.2e+01/16	0.00/1.8e+02/32	0.00/8.3e+01/23	0.00/7.3e+01/17	0.00/1.2e+02/24
genquartic/2000	0.00/7.6e+01/20	0.02/1.7e+02/33	0.00/9.1e+01/28	0.00/8.9e+01/27	0.01/1.8e+02/38
genquartic/1600	0.00/7.4e+01/16	0.01/1.4e+02/26	0.00/1.0e+02/40	0.00/7.8e+01/17	0.01/2.5e+02/52
genquartic/1900	0.00/8.5e+01/23	0.01/2.1e+02/38	0.00/9.5e+01/40	0.00/9.0e+01/25	0.01/2.4e+02/42
sine/1000	0.00/7.5e+01/18	0.03/2.6e+02/148	0.01/9.5e+01/34	0.01/8.3e+01/22	0.02/2.6e+02/51
sine/1200	0.01/8.8e+01/23	0.04/3.6e+02/219	0.01/1.3e+02/61	0.01/9.4e+01/27	0.01/2.1e+02/45
sine/1400	NaN/NaN/NaN	0.06/7.2e+02/485	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
fletcbv3/2	0.00/1.0e+00/0	0.00/1.0e+00/0	0.00/1.0e+00/0	0.00/1.0e+00/0	0.00/1.0e+00/0
nonscomp/78000	0.12/1.7e+02/76	0.64/7.4e+02/481	0.42/4.9e+02/402	0.12/1.5e+02/68	0.29/4.3e+02/90
nonscomp/80000	0.13/1.6e+02/77	0.14/1.8e+02/92	NaN/NaN/NaN	0.26/3.1e+02/165	0.55/7.2e+02/204
nonscomp/82000	0.22/2.7e+02/126	0.15/1.7e+02/84	NaN/NaN/NaN	0.13/1.5e+02/66	0.29/3.7e+02/87
raydan2/1200	0.00/5.6e+01/10	0.01/7.1e+01/11	0.00/5.6e+01/10	0.00/5.6e+01/10	0.00/8.7e+01/13
raydan2/1600	0.00/7.4e+01/17	0.00/1.1e+02/15	0.00/7.4e+01/17	0.00/7.2e+01/15	0.00/8.8e+01/19
raydan2/1800	0.00/7.6e+01/17	0.00/1.0e+02/15	0.00/7.7e+01/18	0.00/8.0e+01/22	0.01/1.4e+02/25
raydan2/2000	0.00/6.9e+01/13	0.00/9.1e+01/14	0.00/7.1e+01/14	0.00/7.1e+01/14	0.00/7.5e+01/10
diagonal1/20	0.00/1.5e+02/82	0.01/1.4e+02/69	0.00/2.0e+02/90	0.00/1.3e+02/70	0.01/2.6e+02/60
diagonal1/60	0.01/2.8e+02/140	0.01/3.2e+02/186	0.01/4.7e+02/134	0.00/1.9e+02/98	0.01/5.6e+02/110
diagonal1/1000	0.41/1.1e+04/1351	0.43/1.1e+04/1760	0.55/1.5e+04/1612	0.44/1.2e+04/1683	NaN/NaN/NaN
diagonal1/1200	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	0.55/1.3e+04/1712	NaN/NaN/NaN
diagonal2/60	0.00/2.0e+02/110	0.02/7.1e+02/124	0.00/1.3e+02/70	0.00/1.9e+02/115	0.01/3.2e+02/79
diagonal2/80	0.00/2.0e+02/119	0.02/8.6e+02/143	0.00/1.3e+02/72	0.01/2.4e+02/148	0.01/3.5e+02/80
diagonal2/100	0.00/1.8e+02/90	0.02/1.2e+03/193	0.00/1.6e+02/86	0.01/2.9e+02/167	0.01/3.8e+02/100
diagonal3/800	0.29/5.9e+03/826	0.25/4.4e+03/1898	0.66/1.4e+04/1502	0.12/2.4e+03/1112	0.43/9.0e+03/1033
diagonal3/1000	0.60/1.1e+04/1524	NaN/NaN/NaN	NaN/NaN/NaN	0.42/7.3e+03/1340	NaN/NaN/NaN
bv/4000	0.04/1.0e+00/0	0.04/1.0e+00/0	0.04/1.0e+00/0	0.04/1.0e+00/0	0.04/1.0e+00/0
ie/300	0.99/5.7e+01/14	2.52/1.5e+02/22	0.99/5.7e+01/14	0.92/5.3e+01/13	2.28/1.3e+02/22
ie/330	1.05/5.0e+01/11	2.69/1.3e+02/21	1.13/5.4e+01/13	1.19/5.7e+01/14	1.86/8.9e+01/20

Table 3. Numerical results of the five method (continued).

	LLYZ	AN	DDY1	EHS-RD2	WYL
Name/n	Tcpu/NG/Itr	Tcpu/NG/Itr	Tcpu/NG/Itr	Tcpu/NG/Itr	Tcpu/NG/Itr
ie/360	1.19/4.8e+01/11	4.88/2.0e+02/29	1.49/6.0e+01/15	1.79/7.2e+01/15	5.83/2.4e+02/42
lin/100	0.02/6.3e+01/13	0.06/1.4e+02/22	0.02/6.3e+01/13	0.02/6.3e+01/13	0.02/1.0e+02/15
lin/200	0.04/7.8e+01/13	0.05/9.3e+01/16	0.04/7.8e+01/13	0.04/7.8e+01/13	0.04/8.4e+01/13
pen1/2000	7.70/6.7e+02/246	NaN/NaN/NaN	5.47/4.7e+02/87	29.55/2.4e+03/1153	5.49/4.5e+02/77
pen1/2800	17.40/7.5e+02/236	22.00/9.9e+02/135	NaN/NaN/NaN	36.27/1.5e+03/707	123.09/5.3e+03/760
pen1/3000	17.28/6.7e+02/156	NaN/NaN/NaN	30.17/1.1e+03/168	65.61/2.5e+03/1379	60.03/2.1e+03/320
trid/100	0.01/1.3e+02/69	0.02/1.3e+02/68	0.01/1.9e+02/134	0.01/1.2e+02/57	0.03/5.3e+02/125
trid/160	0.01/1.3e+02/71	0.01/1.3e+02/71	0.01/1.3e+02/70	0.02/2.0e+02/139	0.01/1.2e+02/57
trid/180	0.01/1.2e+02/65	0.01/1.2e+02/61	0.01/1.2e+02/64	0.02/2.0e+02/145	0.01/1.1e+02/52
trid/200	0.01/1.2e+02/68	0.01/1.2e+02/68	0.01/1.2e+02/62	0.02/2.0e+02/138	0.01/1.1e+02/48

From Figures 1 and 2, and Tables 1, 2 and 3, we have that the numerical performance of the LLYZ is better than the AN, DDY1, EHS-RD2 and WYL methods, and it can successfully solve about 96% of the test problems. The LLYZ also uses less CPU time and fewer iterations, which indicates that it is more effective than the other four methods.

4. Image restoration problems

In this section, we use the LLYZ, Hager-Zhang (HZ) [21], Dai-Kou (DK) [22] and DPRP [8] methods to restore images that added salt and pepper noise, then we compare the CPU calculation time and peak signal-to-noise ratio (PSNR) of restored images for four algorithms.

Raymond et al. proposed the two-phase scheme to remove pulse noise in [23]. In the first stage, we use the median filter to find the salt and pepper noise points in the images. Mark X represents the original image with pixels $M*N$, let $A = \{1, 2, 3, \cdots, M\}*\{1, 2, 3, \cdots, N\}$ be the index set of the image X and $\mathcal{N} \subset A$ represents the set of noise point indicators detected in the first stage. In the second stage, we use conjugate gradient method to solve the following smooth unconstrained optimization problem, then removing noise and repairing images:

$$\min_{\mathbf{u}} F_\alpha(\mathbf{u}) := \sum_{(i,j)\in\mathcal{N}} \left(2 \sum_{(m,n)\in V_{i,j}\setminus\mathcal{N}} \varphi_\alpha\left(u_{i,j} - y_{m,n}\right) + \sum_{(m,n)\in V_{i,j}\cap\mathcal{N}} \varphi_\alpha\left(u_{i,j} - u_{m,n}\right)\right). \quad [24] \tag{4.1}$$

Where $V_{i,j} = \{(i, j-1), (i, j+1), (i-1, j), (i+1, j)\}$ is the set of the neighbors for the pixel at pixel location $(i, j) \in A$, $y_{m,n}$ is the observed pixel value of the image at pixel location (m, n), $\varphi_\alpha(t) = \sqrt{t^2 + \alpha}$ $(\alpha > 0)$ is an edge-preserving function and $u = \left[u_{i,j}\right]_{(i,j)\in\mathcal{N}}$ is a column vector with length c, where c is the elements number of set $\mathcal{N}$.

We use the PSNR calculation method from [25]

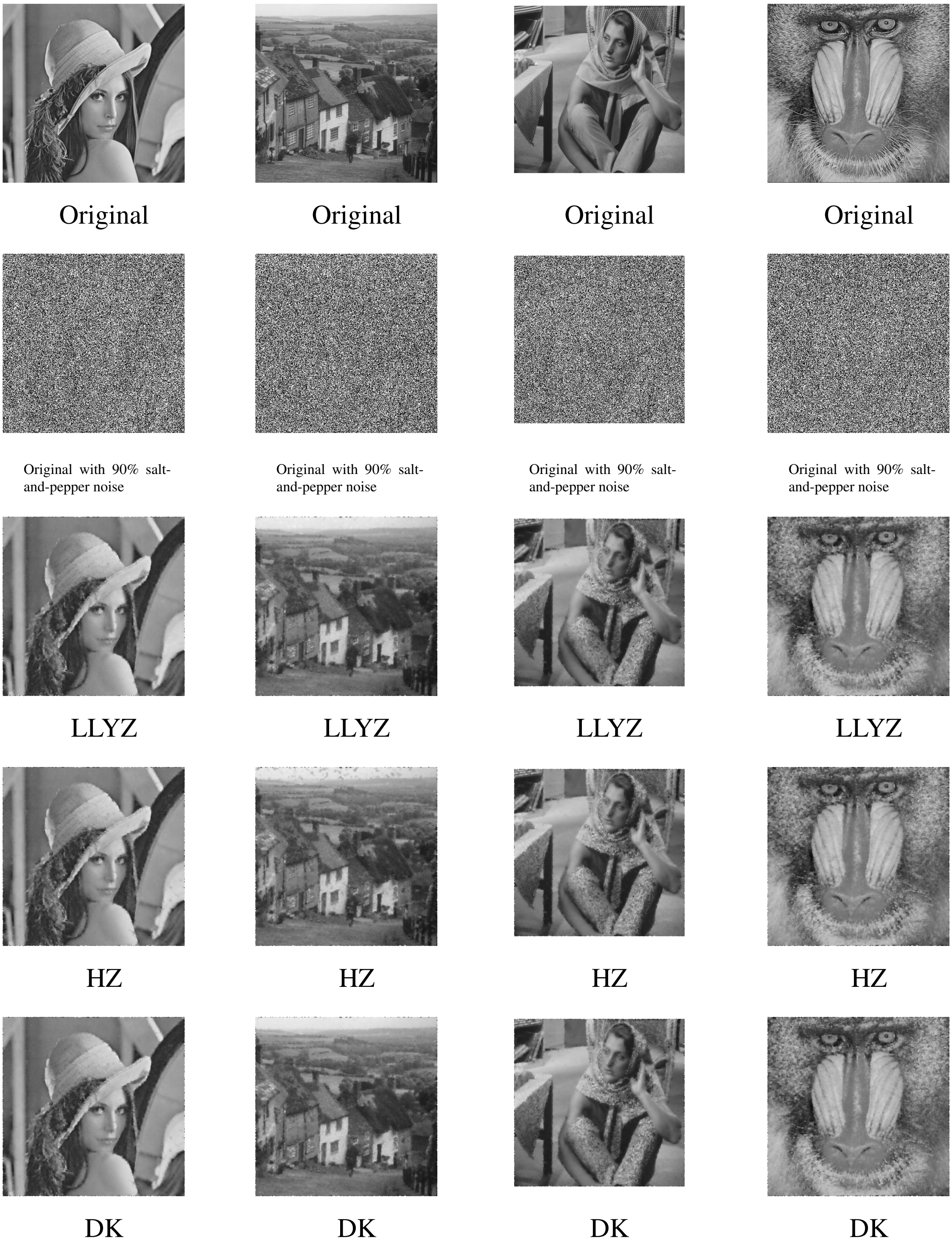
$$\mathrm{PSNR} = 10 \log_{10} \frac{255^2}{\frac{1}{MN}\sum_{i,j}\left(x^r_{i,j} - x^*_{i,j}\right)^2}, \tag{4.2}$$

where $x^r_{i,j}$ and $x^*_{i,j}$ denote the pixel values of the restored image and the original image.

We use four original pictures to test the image restoration capability of the above four methods. The four pictures' pixels are all $512 * 512$ and the pictures have been added with $30\%, 50\%, 70\%$ and 90%

salt and pepper noise, then we use the above four different conjugate gradient methods to restore them. The termination condition of the algorithm is $Itr > 300$ or $\frac{|F_\alpha(\mathbf{u}_k)-F_\alpha(\mathbf{u}_{k-1})|}{|F_\alpha(\mathbf{u}_k)|} \leq 10^{-4}$.

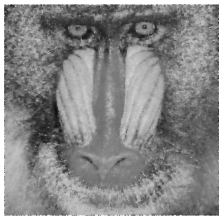
To save space, Figure 3 only shows the original pictures, which added 90% salt and pepper noise, and their image restoration results. Table 4 shows us the detailed data of the CPU calculation time and PSNR of the restored images.



DPRP

DPRP

DPRP



DPRP

Figure 3. First row: The original images, second row: The noisy images with 90% salt-and pepper noise, third row to last row: Restored images by LLYZ method, HZ method, DK method, DPRP method.

Table 4. Numerical results of image restoration testing.

Image	Noise ratio	LLYZ	HZ	DK	DPRP
		Tcpu/PSNR	Tcpu/PSNR	Tcpu/PSNR	Tcpu/PSNR
Lena.bmp	30%	3.64/36.82	4.33/36.94	3.77/36.96	3.85/36.99
Lena.bmp	50%	6.45/34.32	6.01/34.18	6.45/34.43	6.39/34.44
Lena.bmp	70%	7.82/30.98	10.14/31.10	7.70/31.07	8.17/31.09
Lena.bmp	90%	15.76/26.27	14.17/26.03	10.95/26.05	10.17/25.99
Goldhill.bmp	30%	3.92/35.01	3.37/34.96	3.61/35.03	3.26/35.01
Goldhill.bmp	50%	5.12/32.46	6.03/32.65	5.94/32.66	6.13/32.73
Goldhill.bmp	70%	8.19/29.61	7.94/29.52	8.28/29.80	8.16/29.81
Goldhill.bmp	90%	17.24/25.60	13.38/25.05	12.96/25.59	11.76/25.54
Barbara.bmp	30%	3.72/28.67	3.96/28.65	3.73/28.66	3.24/28.66
Barbara.bmp	50%	5.70/26.71	6.22/26.71	6.00/26.69	6.39/26.70
Barbara.bmp	70%	7.71/24.60	8.77/24.64	7.68/24.62	7.67/24.61
Barbara.bmp	90%	15.49/22.52	15.86/22.52	12.41/22.52	10.95/22.50
Baboon.bmp	30%	3.72/26.52	3.90/26.49	3.77/26.50	3.67/26.51
Baboon.bmp	50%	6.43/24.56	6.30/24.56	6.61/24.55	6.87/24.54
Baboon.bmp	70%	9.33/22.48	7.69/22.47	8.25/22.46	8.21/22.45
Baboon.bmp	90%	15.61/20.14	14.94/20.15	12.54/20.15	13.25/20.13

From the above pictures, we can see that the above four modified conjugate methods can all restore noisy images very well. The CPU time and PSNR of image restoration is clearly displayed in Table 4. From Table 4, We find that the PSNR values of the images restored by LLYZ is higher than HZ, DK and DPRP in most cases, which means images restored by LLYZ are clearer than other methods. The above results indicate that the LLYZ method is effective.

5. Conclusions

In this paper we have proposed a new conjugate gradient method with a restart direction. First, based on the β_k^{PRP} and previous research experiences, we appropriately increased the denominator and decreased the numerator of β_k, and proposed a new conjugate parameter β_k^{LLYZ}. Meanwhile, by calculating the size of $g_k^{\mathrm{T}} g_{k-1}$, we determined different descent directions of the LLYZ method to ensure its sufficient descent. Second, we proved the sufficient descent property and global convergence of the LLYZ method under the strong Wolfe line search [9, 10], then we compared the LLYZ method with the AN [16], DDY1 [11], EHS-RD2 [12] and WYL [7] methods in terms of numerical performances by numerical experiments. From Figures 1 and 2 and Tables 1, 2 and 3, we found that the LLYZ method performed better than other methods and it could successfully solve about 96% of the test problems. Lastly, we used the LLYZ, HZ [21], DK [22] and DPRP [8] methods to restore images that added salt-and-pepper noise. From Figure 3 and Table 4, we have that the LLYZ method performed well in image restoration problems by solving the smooth unconstrained optimization problem (4.1). To sum up, the proposed method LLYZ is an effective algorithm.

In the future, we will focus on the innovation of restart directions of the conjugate method, which can make the improved methods converge in less CPU time and solve most problems successfully.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This research is funded by the National Natural Science Foundation of China (No.11761005), and all researchers would like to thank the School of Mathematics and Information Science at the North Minzu University.

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

1. M. R. Hestenes, E. L. Steifel, Method of conjugate gradients for solving linear systems, *J. Res. Nat. Standard.,* **49** (1952), 409–436. https://dx.doi.org/10.6028/jres.049.044

2. E. Polak, G. Ribière, Note sur la convergence de méthodes de directions conjuguées, *Rev. Fr. Informat Rech. Operationelle 3e Annee.,* **16** (1969), 35–43. Available from: https://www.esaim-m2an.org/articles/m2an/pdf/1969/01/m2an196903R100351.pdf

3. B. T. Polyak, The conjugate gradient method in extremal problems, *UssR Comput. Math. Math. Phys.,* **9** (1969), 94–112. https://dx.doi.org/10.1016/0041-5553(69)90035-4

4. R. Fletcher, C. Reeves, Function minimization by conjugate gradients, *Comput. J.,* **7** (1964), 149–154. https://doi.org/10.1093/comjnl/7.2.149

5. Y. Liu, C. Storey, Efficient generalized conjugate gradient algorithms, part1: theory., *J. Optim. Theory. Appl.,* **69** (1991), 129–137. https://doi.org/10.1007/BF00940464

6. Y. H. Dai, Y. Yuan, A nonlinear conjugate gradient method with a strong global convergence property, *SIAM J. Optim.,* **10** (1999), 177–182. https://doi.org/10.1137/s1052623497318992
7. Z. X. Wei, S. W. Yao, L. Y. Liu, The convergence properties of some new conjugate gradient methods, *Appl. Math. Comput.,* **183** (2006), 1341–1350. https://doi.org/10.1016/j.amc.2006.05.150
8. Z. F. Dai, F. H. Wen, Another improved Wei-Yao-Liu nonlinear conjugate gradient method with sufficient descent property, *Appl. Math. Comput.,* **218** (2012), 7421–7430. https://doi.org/10.1016/j.amc.2011.12.091
9. P. Wolfe, Convergence conditions for ascent methods, *Siam Rev.,* **11** (1969), 226–235. https://doi.org/10.1137/1011036
10. P. Wolfe, Convergence conditions for ascent methods II: Some Corrections, *Siam Rev.,* **13** (1971), 185–188. https://doi.org/10.1137/1013035
11. Z. Zhu, D. Zhang, S. Wang, Two modified DY conjugate gradient methods for unconstrained optimization problems, *Appl. Math. Comput.,* **373** (2020), 125004. https://doi.org/10.1016/j.amc.2019.125004
12. L. Pengjie, W. Yanqiang, S. Feng, Z. Yan, S. Hu, Two extended HS-type conjugate gradient methods with restart directions, *Acta Math. Sci.,* **43** (2023), 570–580. Available from: http://121.43.60.238/sxwlxbA/EN/Y2023/V43/I2/570
13. Y. H. Dai, C. X. Kou, A modified self-Scaling memoryless Broyden-Fletcher-Goldfarb-Shanno method for unconstrained optimization, *J. Optim. Theory. Appl.,* **165** (2015), 209–224. https://doi.org/10.1007/s10957-014-0528-4
14. J. C. Gilbert, J. Nocedal, Global convergence properties of conjugate gradient methods for optimization, *SIAM J. Optim.,* **2** (1992), 21–42. https://doi.org/10.1137/0802003
15. G. Zoutendijk, Nonlinear programming computational methods, in: J. Abadie (Ed.), *Integer nonlin. Program.,* (1970), 37–86.
16. N. Andrei, Hybrid conjugate gradient algorithm for unconstrained optimization, *J. Optim. Theory Appl.,* **141** (2009), 249–264. https://doi.org/10.1007/s10957-008-9505-0
17. N. Gould, D. Orban, P. L. Toint, CUTEr and SifDec: A constrained and unconstrained testing environment, *ACM Trans. Math. Softw.,* **29** (2003), 373–394. https://doi.org/10.1145/962437.962439
18. J. J. More, B. S. Garbow, K. E. Hillstrom, Testing unconstrained optimization software, *ACM T. Math. Softw.,* **7** (1981), 17–41. https://doi.org/10.1145/355934.355936
19. N. Andrei, An unconstrained optimization test functions collection, *Adv. Model. Optim.,* **10** (2008), 147–161.
20. E. D. Dolan, J. J. Moré, Benchmarking optimization software with performance profiles, *Math. Program., Ser. A.,* **91** (2002), 201–213. https://doi.org/10.48550/arXiv.cs/0102001
21. W. W. Hager, H. C. Zhang, A new conjugate gradient method with guaranteed descent and an efficient line search, *Siam J. Optim.,* **16** (2005), 170–192. https://doi.org/10.1137/030601880

22. Y. H. Dai, C. X. Kou, A nonlinear conjugate gradient algorithm with an optimal property and an improved Wolfe line search, *Siam J. Optim.,* **23** (2013), 296–320. https://doi.org/10.1137/100813026

23. R. H. Chan, C. W. Ho, M. Nikolova, Salt-and-pepper noise removal by median-type noise detectorsand detail-preserving regularization, *IEEE Trans. Image Process.,* **14** (2005), 1479–1485. https://doi.org/10.1109/TIP.2005.852196

24. X. Z. Jiang, W. Liao, J. H. Yin, J. Jian, A new family of hybrid three-term conjugate gradient methods with applications in image restoration, *Numer. Algorithms,* **91** (2022), 161–191. https://doi.org/10.1007/s11075-022-01258-2

25. A. C. Bovik, *Handbook of image and video processing*, Academic press, 2000. https://dl.acm.org/doi/book/10.5555/556230

AIMS Press