manuscript No.
(will be inserted by the editor)

Some Modified Fast Iteration Shrinkage Thresholding Algorithms with a New Adaptive Non-monotone Stepsize Strategy for Nonsmooth and Convex Minimization Problems

Hongwei Liu[1] · Ting Wang[1] · Zexian Liu[23] ·

Received: date / Accepted: date

Abstract The " fast iterative shrinkage-thresholding algorithm " (FISTA) is one of the most famous first order optimization scheme, and the stepsize, which plays an important role in theoretical analysis and numerical experiment, is always determined by a constant related to the Lipschitz constant or by a backtracking strategy. In this paper, we design a new adaptive non-monotonic stepsize strategy (NMS), which allows the stepsize increases monotonically after finite iterations. It is remarkable that NMS can be successfully implemented without knowing the Lipschitz constant or without backtracking. And the additional cost of NMS is less than the cost of some existing backtracking strategies. For using NMS to the original FISTA (FISTA_NMS) and the modified FISTA (MFISTA_NMS), we show that the convergence results stay the same. Moreover, under the error bound condition, we show that FISTA_NMS achieves the rate of convergence to $o\left(\frac{1}{k^6}\right)$ and MFISTA_NMS enjoys the convergence rate related to the value of parameter of t_k, that is $o\left(\frac{1}{k^{2(a+1)}}\right)$; And the iterates generated by the above two algorithms are strong convergent. In addition, by taking advantage of the restart technique to accelerate the above two methods, we establish the linearly convergences of the function value and iterates under the error bound condition. Similar results can not be obtained if we use the backtracking schemes. We conduct some numerical experiments to examine the effectiveness of the proposed algorithms.

Keywords FISTA, Proximal-based method, Adaptive non-monotone stepsize strategy, Inertial forward-backward algorithms, Convergence rate, Convex optimization

Mathematics Subject Classification (2000) 94A12 · 65K10 · 94A08 · 90C06 · 90C25 ·

Ting Wang (✉)
E-mail: wangting_7640@163.com

Hongwei Liu
E-mail: hwliu@mail.xidian.edu.cn

Zexian Liu
E-mail: liuzexian2008@163.com

[1] School of Mathematics and Statistics, Xidian University, Xi'an, 710126, China
[2] School of Mathematics and Statistics, Guizhou University, Guiyang, 550025, China
[3] State Key Laboratory of Scientific and Engineering Computing, Institute of Computational Mathematics and Scientific/Engineering computing, AMSS, Chinese Academy of Sciences, Beijing, 100190, China.

1 Introduction

We consider the non-smooth optimization problem:
(P) $$\min_{x\in R^n} F(x) = f(x) + g(x).$$

The following assumptions are made throughout the paper:
A) $g : R^n \to]-\infty, +\infty]$ is a proper, convex, " proximal-friendly " [10] and lower semi-continuous function.
B) $f : R^n \to]-\infty, +\infty[$ is a smooth convex function and continuously differentiable with Lipschitz continuous gradient, i.e., there exists a Lipschitz constant L_f such that for every $x, y \in R^n$, $\|\nabla f(x) - \nabla f(y)\| \le L_f \|x - y\|$ and $\|\cdot\|$ denotes the standard Euclidean norm.
C) Problem (P) is solvable, i.e., $X^* := \arg\min F \ne \emptyset$, and for $x^* \in X^*$ we set $F^* := F(x^*)$.

Problem (P) arises in many contemporary applications such as machine learning [24], compressed sensing [12], and image processing [7]. And due to the importance and the popularity of the problem (P), various attempts have been made to solve it efficiently, especially when the problem instances are of large scale. One popular class of methods for solving problem (P) are first-order methods due to their cheap iteration cost and good convergence properties. Among them, the proximal gradient (PG) method [13,16,21] is arguably the most fundamental one, in which the basic iteration is

$$x_{k+1} = \mathrm{prox}_{\lambda_k g}(x_k - \lambda_k \nabla f(x_k)),\ \lambda_k \in \left]0, 1/L_f\right], \tag{1}$$

where $\mathrm{prox}_{\lambda g}(\cdot) = \arg\min\limits_x \left\{g(x) + \frac{1}{2\lambda}\|x - \cdot\|^2\right\}$ denotes the proximal operator of g, and λ_k indicates the stepsize and has an upper bound related to Lipschitz constant. The convergence of PG has been well studied in the literature under various contexts and frameworks (The detailed information can be referred to [6,8,17,19]). However, PG can be slow in practice, see, for example, [23].

Various ways have thus been made to accelerate the proximal gradient algorithm. By performing the extrapolation technique, a prototypical algorithm takes the following form:

$$\begin{aligned} y_{k+1} &= x_k + \gamma_k (x_k - x_{k-1}), \\ x_{k+1} &= p_{\lambda_{k+1} g}(y_{k+1}), \end{aligned} \tag{2}$$

where γ_k is the extrapolation parameter satisfying $0 \le \gamma_k \le 1$, $\lambda_{k+1} \in \left]0, 1/L_f\right]$, and

$$p_{\lambda g}(y) = \arg\min_x \{Q_\lambda(x, y)\} = \mathrm{prox}_{\lambda g}(y - \lambda \nabla f(y)). \tag{3}$$

Here $Q(x, y)$ be the approximation function of $F(x)$ at the given point y, where

$$Q_\lambda(x, y) = g(x) + f(y) + \langle \nabla f(y), x - y\rangle + \frac{1}{2\lambda}\|x - y\|^2, \quad \forall x \in R^n. \tag{4}$$

One representative algorithm that takes the form of (2) and with the extrapolation parameter

$$\gamma_k = \frac{t_k - 1}{t_{k+1}},\ \text{where } t_1 = 1, t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2} \tag{5}$$

is the fast iterative shrinkage-thresholding algorithm (FISTA), which was proposed by Beck and Teboulle [4] and was based on the idea that introduced and developed by Nesterov [20] for minimizing a smooth convex function. The stepsize λ_k can be dynamically updated to estimate the Lipschitz constant L_f by a backtracking stepsize rule. FISTA is a very effective algorithm that keeps the simplicity scheme like PG and improves the convergence rate of objective function value to $O\left(1/k^2\right)$ for solving the problem (P); hence, it has become a standard algorithm [16] and motivated subsequent studies on the extrapolation scheme (2), see, for example, [5,22,23,28]. Though FISTA is surprisingly efficient, the convergence of the whole iterative sequence generated by FISTA is still unclear [11,29]. Chambolle and

Dossal [11] established the convergence of the sequence generated by FISTA with the new parameter $\gamma_k = \frac{k-1}{k+a}$ for a fixed $a > 2$ and the assumption $\lambda_k \in \left]0, \frac{1}{L_f}\right]$, for problem (P). Furthermore, Attouch and Peypouquet [1] proved that the convergence rate of function value generated by the algorithm in [11] is $o\left(\frac{1}{k^2}\right)$ and they considered the convergence of iterates and the rate of convergence of function value for the scheme of (2) with various options of extrapolation parameter γ_k in [2]. Under the error bound conditon, Wen, Chen and Pong [29] showed that there exists a threshold depending on the Lipschitz constant L_f such that if γ_k is below this threshold, then the sequence generated by (2) is $R-$linearly convergent when f in problem (P) is possible nonconvex. As we see, at least one parameter in these algorithm such as [1,2,11,29] is directly related to the Lipschitz constant, which results in that the algorithm implementation as well as the theoretical analysis rely heavily on the Lipschitz constant.

Backtracking for estimating Lipschitz constant works well in practice but the principal drawback is that the stepsize λ_k generated by the backtracking strategy in FISTA is non-increasing, which can substantially limit the performance of FISTA when a small stepsize is encountered early in the algorithm since this causes the stepsize taken at that point, and at all subsequent iterates, to be very small. Scheinberg [26] develop a new backtracking strategy, which allows stepsize to increase. This new backtracking strategy [26] starts with a new initial value at the beginning of each iteration, rather than the stepsize of last iteration like the backtracking in FISTA, and estimates the local Lipschitz constant L_k, which is often smaller than L_f. Hence, $\frac{1}{L_k}$ may be a better estimate for the stepsize than $\frac{1}{L_f}$. With this new backtracking strategy, they proposed a new versions of accelerated FISTA (FISTA_BKTR), which reduces the number of iteration greatly and the calculating cost is less more than the one in backtracking rule of original FISTA, and the convergence result is still $O\left(1/k^2\right)$.

It is natrual that each time the backtracking step operates, the calculating cost of algorithm will increase. Although both of the mentioned backtracking strategies works well, we still pursue to develop a stepsize strategy, which dose not use the backtracking procedure and can bring some numerical improvements and some new theoretical results. In this paper, we exploit a new adaptive non-monotone stepsize technique (NMS) to determine λ_k in (2), where the stepsize increases monotonically after finite interations. We prove that FISTA with NMS keeps $O\left(1/k^2\right)$ convergence rate of the objective function value, which is similar with original FISTA and FISTA_BKTR. By using the new choice of t_k in FISTA [11] and the new adaptive non-monotone technique, we present a modified FISTA with NMS which also achieves $o\left(\frac{1}{k^2}\right)$ convergence rate of the objective function value. Also, the convergence of the iterative sequence is established without dependence on the Lipschitz constant L_f unlike the analysis in [11]. Meanwhile, we prove that both of those two algorithms with NMS enjoy $o\left(\frac{1}{k}\right)$ convergent rate of the norm of subdifferential of the objective function. Furthermore, under the error bound condition, we prove that FISTA and FISTA_CD with NMS can achieve some improved convergence rates for objective function value and iterates convergent strongly; we also take advantage of the restart technique in [23] to accelerate the above FISTA methods with NMS, and establish the linear convergences of the function value and iterative sequence under the error bound condition.

The reminder of the paper is organized as follows. In Section 2, we provide a new adaptive non-monotone stepsize strategy. In Section 3, we propose an algorithm FISTA_NMS by combining FISTA with the new adaptive non-monotone technique, which ensures the similar convergence rate of the objective function value with FISTA, and a greater convergence rate of the norm of subdifferential of function value than FISTA. In Section 4, with a small modification, we present a MFISTA_NMS which has similar theoretical results like [1,11,25]. In Section 5, we use the restart technique in [23] to accelerate the above methods and establish the linear convergences of the function value and iterates under the error bound condition. Numerical results are reported in Section 6. In the last section, conclusions and discussions are presented.

2 Adaptive non-monotone stepsize strategy

In this section, we present a new adaptive non-monotone stepsize strategy.

Denote that the computations of $t_{k+1} := \frac{1+\sqrt{1+4\theta_k t_k^2}}{2}$ and $y_{k+1} := x_k + \frac{t_k - 1}{t_{k+1}}\left(x_k - x_{k-1}\right)$ by $\left(t_{k+1}, y_{k+1}\right) = FistaStep\left(x_k, x_{k-1}, t_k, \theta_k\right)$.

We first state the algorithms of FISTA with backtracking [4] and the detailed algorithm of FISTA_BKTR [26] as follows.

Algorithm 1 FISTA with backtracking

```
Step 0. Set $t_1 = 1$ and $y_1 = x_0, \lambda_0 > 0$; $\eta < 1$.
Step k. (1) Finding the smallest nonnegative integers $i_k$ such that with $\lambda_k = \eta^{i_k}\lambda_{k-1}$
            $$F\left(p_{\lambda_k g}\left(y_k\right)\right) \leq Q_{\lambda_k}\left(p_{\lambda_k g}\left(y_k\right), y_k\right). \tag{6}$$
        (2) compute $\left(t_{k+1}, y_{k+1}\right) = FistaStep\left(x_k, x_{k-1}, t_k, 1\right)$
```

Since that (6) holds if $\lambda_k \leq \frac{1}{L_f}$, we have that $\lambda_k > \frac{\eta}{L_f}$, which means that the lower bound of stepsize is related to L_f, and λ_k in Algorithm 1 can be seen an estimate for the global Lipschitz constant. It is easy to obtain that there are at most $\log_{\frac{1}{\eta}}\left(\lambda_0 L_f\right) + 1$ backtracking steps at each iteration [26]. Each time the backtracking performs, $p_{\lambda_k g}\left(y_k\right)$ and $f\left(p_{\lambda_k g}\left(y_k\right)\right)$ must be recomputed, that is the main cost of FISTA_backtracking.

To obtain larger stepsize than Algorithm 1, The following Algorithm 2 propose a new backtracking step rule, which starts with a new initial value at the beginning of each iteration and can be reduced to Algorithm 1 if we set $\lambda_k^0 = \lambda_{k-1}$.

Algorithm 2 FISTA_BKTR

```
Step 0. Set $t_1 = 1, t_0 = 0, 0 < \beta < 1, \theta_0 = 1$ and $y_1 = x_0 = x^{-1}, \lambda_1^0 > 0$;
Step k. (1) Set $\lambda_k := \lambda_1^0$, and compute $\nabla f\left(y_k\right), p_{\lambda_k}\left(y_k\right)$
        (2) If $F\left(p_{\lambda_k}\left(y_k\right)\right) > Q_{\lambda_k}\left(p_{\lambda_k}\left(y_k\right), y_k\right)$,
                    set $\lambda_k := \beta\lambda_k, \theta_{k-1} := \theta_{k-1}/\beta$
                $\left(t_k, y_k\right) = FistaStep\left(x_{k-1}, x_{k-2}, t_{k-1}, \theta_{k-1}\right)$
            return to (2)
        (3) $x_k := p_{\lambda_k}\left(y_k\right)$
                    choose $\lambda_{k+1}^0 > 0$ and set $\theta_{k-1} := \lambda_k/\lambda_{k+1}^0$
                    $\left(t_{k+1}, y_{k+1}\right) = FistaStep\left(x_k, x_{k-1}, t_k, \theta_k\right)$
```

We see that the updating θ_k equivalent to $\theta_k = \frac{\lambda_k}{\lambda_{k+1}}$ and λ_k in Algorithm 2 is an estimate for the local Lipschitz constant, while the λ_k in Algorithm 1 is an estimate global of Lipschitz constant. Similar to the analysis of Algorithm 1, the lower bound of stepsize is related to the local Lipschitz constant L_k for $\nabla f\left(x\right)$ restricted to the interval $\left[p_{\lambda_k g}\left(y_k\right), y_k\right]$ for any $\lambda_k \leq \frac{1}{L_k}$, which is less than or equals to L_f. If the backtracking step performed, the values of $f\left(y_k\right)$, $\nabla f\left(y_k\right)$, $p_{\lambda_k g}\left(y_k\right)$ and $f\left(p_{\lambda_k g}\left(y_k\right)\right)$ must be recomputed. Here, we can see that computation of $f\left(y_k\right)$ and $\nabla f\left(y_k\right)$ will be additional costs over against Algorithm 1 for the case that ∇f is non-linear; otherwise, those computation can be negligible. Since the option of initial stepsize is related to the number of backtracking steps closely, based on the idea of Nesterov [21], the author choose $\lambda_k^0 = \frac{\lambda_k}{\sigma}\left(\sigma \geq \eta\right)$ to reduce the total number of backtracking steps to $[1+\frac{\ln\sigma}{\ln\eta}]\left(Iter + 1\right) + \frac{1}{\ln\eta}[\ln\frac{\sigma\lambda_0}{\eta/L_f}]_+$, where $Iter$ means the total number of iterations of Algorithm 2. When we set $\sigma = \eta$, the average number of backtracking steps at each iteration is 2.

Although Algorithm 2 greatly reduces the number of cycle of the internal loop, and generates better stepsize, it still may have additional costs per backtracking step, especially, when the function f is non-linear, the computations of $f(y_k)$, $\nabla f(y_k)$, $p_{\lambda_k g}(y_k)$ and $f\left(p_{\lambda_k g}(y_k)\right)$ will occupy the CPU time. Hence, we design a stepsize strategy that directly gives the stepsize at each iteration, which avoids any extra computations due to line search. We present the adaptive non-monotone stepsize strategy as follows.

Algorithm 3 Adaptive non-monotone stepsize strategy

```
Let $\{x_k\}, \{y_k\}$ be generated by the scheme of FISTA, and $\sum_{k=1}^{\infty} E_k$ is a convergent nonnegative series.
Set $0 < \mu_1 < \mu_0 < 1$.
    if $\langle \nabla f(x_k) - \nabla f(y_k), x_k - y_k \rangle > \frac{\mu_0}{\lambda_k} \|x_k - y_k\|^2$ holds, set
        $$\lambda_{k+1} = \mu_1 \frac{\|x_k - y_k\|^2}{\langle \nabla f(x_k) - \nabla f(y_k), x_k - y_k \rangle}, \quad (7)$$
    otherwise,
        $$\lambda_{k+1} = \lambda_k (1+E_k). \quad (8)$$
```

In Algorithm 3, we use the condition

$$\langle \nabla f(x) - \nabla f(y), x - y \rangle \leq \frac{\mu_0}{\lambda} \|x - y\|^2, \text{ where } \mu_0 \in \left]0, 1\right[, \tag{9}$$

to control the increase or decrease of the stepsize λ_k. When the condition (9) does not holds, the stepsize λ_k is determined by (7), which implies that $\lambda_{k+1} < \lambda_k$. Conversely, $\lambda_{k+1} \geq \lambda_k$. The $\sum_{k=1}^{\infty} E_k$ is called control series, which can be corrected adaptively for better control of stepsize growth. For the choice E_k, we will discuss later in this section.

It is remarkable that it is not required to know the Lipschitz constant or use a line search procedure when one uses Algorithm 3 to determine the stepsize λ_k. Now we study some significant properties of the stepsize $\{\lambda_k\}$ generated by Algorithm 3.

Lemma 2.1 *Let $\{\lambda_k\}$ be the sequence generated by Algorithm 3. We have that the sequence $\{\lambda_k\}$ is convergent, and*

$$\lambda_k \geq \lambda_{min} = \min\left\{\lambda_1, \frac{\mu_1}{L_f}\right\}, \quad \forall k \geq 1. \tag{10}$$

Proof. First, we prove that $\forall k \geq 1$, $\lambda_k \geq \min\left\{\lambda_1, \frac{\mu_1}{L_f}\right\}$ holds by induction.

For $k = 1$, the conclusion is obvious. Suppose that for $\forall p > 1$ such that for $k = p$, the conclusion holds. Then, for $k = p + 1$, there are two situations:
(1) λ_{p+1} is generated by (7). We obtain

$$\lambda_{p+1} = \mu_1 \frac{\|x_p - y_p\|^2}{\langle \nabla f(x_p) - \nabla f(y_p), x_p - y_p \rangle} \geq \frac{\mu_1}{L_f}, \tag{11}$$

the inequality is follows from the fact that f is Lipschitz continuous gradient.
(2) λ_{p+1} is generated by (8). We obtain

$$\lambda_{p+1} \geq \lambda_p \geq \min(\lambda_1, \frac{\mu_1}{L_f}). \tag{12}$$

From (11) and (12), we conclude that $\forall k \geq 1$, $\lambda_k \geq \min\left\{\lambda_1, \frac{\mu_1}{L_f}\right\}$ holds for $\forall k \geq 1$.

Denote that

$$\ln \lambda_{i+1} - \ln \lambda_i = (\ln \lambda_{i+1} - \ln \lambda_i)^+ - (\ln \lambda_{i+1} - \ln \lambda_i)^-, \tag{13}$$

where $(\cdot)^+ = \max\{0, \cdot\}$, $(\cdot)^- = -\min\{0, \cdot\}$. Following the fact that

$$\ln \lambda_{i+1} - \ln \lambda_i \leq \ln(1 + E_i) \leq E_i, \forall i \geq 1, \tag{14}$$

we have

$$(\ln \lambda_{i+1} - \ln \lambda_i)^+ \leq E_i, \forall i = 1, 2, \cdots, \tag{15}$$

which implies that $\sum_{i=1}^{\infty} (\ln \lambda_{i+1} - \ln \lambda_i)^+$ is convergent from the fact that $\sum_{i=1}^{\infty} E_i$ is a convergent nonnegative series.

The convergence of $\sum_{i=1}^{\infty} (\ln \lambda_{i+1} - \ln \lambda_i)^-$ also can be proved as follows.

Assume by contradiction that $\sum_{i=1}^{\infty} (\ln \lambda_{i+1} - \ln \lambda_i)^- = +\infty$. Based on the convergence of $\sum_{i=1}^{\infty} (\ln \lambda_{i+1} - \ln \lambda_i)^+$ and the equality

$$\begin{aligned} \ln \lambda_{k+1} - \ln \lambda_1 &= \sum_{i=1}^{k} (\ln \lambda_{i+1} - \ln \lambda_i) \\ &= \sum_{i=1}^{k} (\ln \lambda_{i+1} - \ln \lambda_i)^+ - \sum_{i=1}^{k} (\ln \lambda_{i+1} - \ln \lambda_i)^- \end{aligned} \tag{16}$$

we can easily deduce $\lim\limits_{k\to\infty} \ln \lambda_k = -\infty$, which is a contradiction with $\lambda_k \geq \min\left\{\lambda_1, \frac{\mu_1}{L_f}\right\} > 0$. As a result, $\sum_{i=1}^{\infty} (\ln \lambda_{i+1} - \ln \lambda_i)^-$ is a convergent series. Then, in view of (16), we obtain the sequence $\{\lambda_k\}$ is convergent. □

Lemma 2.2 *For the sequence* $\{\lambda_k\}$ *generated by Algorithm 3, there exists a positive integer* $\hat{k} \geq 1$ *such that condition (9) holds constantly for every* $k > \hat{k}$.

Proof. Suppose the conclusion is not true, i.e. there exists a sequence $\{k_j\}$, where $k_j \to \infty$, such that

$$\begin{aligned} \left\|x_{k_j} - y_{k_j}\right\|^2 &< \frac{\lambda_{k_j}}{\mu_0} \left\langle \nabla f\left(x_{k_j}\right) - \nabla f\left(y_{k_j}\right), x_{k_j} - y_{k_j} \right\rangle \\ &= \frac{\lambda_{k_j}}{\lambda_{k_j+1}} \frac{1}{\mu_0} \lambda_{k_j+1} \left\langle \nabla f\left(x_{k_j}\right) - \nabla f\left(y_{k_j}\right), x_{k_j} - y_{k_j} \right\rangle \\ &= \frac{\lambda_{k_j}}{\lambda_{k_j+1}} \frac{\mu_1}{\mu_0} \left\|x_{k_j} - y_{k_j}\right\|^2. \end{aligned} \tag{17}$$

Combining this with the fact

$$\lim_{j\to\infty} \frac{\lambda_{k_j}}{\lambda_{k_j+1}} \frac{\mu_1}{\mu_0} = \frac{\mu_1}{\mu_0} < 1, \tag{18}$$

which follows from Lemma 2.1, we obtain

$$\left\|x_{k_j} - y_{k_j}\right\|^2 < \left\|x_{k_j} - y_{k_j}\right\|^2, \quad \text{for } j \text{ sufficient large}, \tag{19}$$

which is a contradiction. Therefore, (9) will holds constantly after a finite iterations $\hat{k}$. □

For the rest of this article, we always denote that $k_0 = \hat{k} + 1$ is the first positive integer such that λ_k satisfy the condition (9), which means that condition (9) holds for any $k \geq k_0$. It follows from Lemma 2.2 that the stepsize $\{\lambda_k\}$ generated by Algorithm 3 increase monotonically after $\hat{k}$ step.

According to Lemma 2.1 and Lemma 2.2, we can easily obtain the following conclusion.

Corollary 2.1 *For the sequence $\{\lambda_k\}$ generated by Algorithm 3, denote that $\lim\limits_{k\to\infty} \lambda_k = \lambda^*$. Then, for any k sufficient large, we have $\lambda_k \le \lambda_{k+1} \le \lambda^*$.*

Now, we discuss the choice of E_k. In Algorithm 3, we set $E_k := \frac{w_k}{k^p}\,(p > 1)$, where w_i is a nonnegative bounded constant. Generally, we set the value of p is close to 1. For the choice of w_k, we can adjust the value of w_k based on the angle between the vectors $x_k - x_{k-1}$ and $x_{k-1} - x_{k-2}$. If the value $\frac{\langle x_k - x_{k-1}, x_{k-1} - x_{k-2}\rangle}{\|x_k - x_{k-1}\|\|x_{k-1} - x_{k-2}\|}$ is close to 1, it may be caused by a small stepsize, then, we expect a larger stepsize. Hence, we can set the value of w_k adaptively. In the following, we give the details for setting w_k.
Set $w_k = \eta_1$, if $\langle x_k - x_{k-1}, x_{k-1} - x_{k-2}\rangle \le 0.9\,\|x_k - x_{k-1}\|\,\|x_{k-1} - x_{k-2}\|$;
set $w_k = \eta_3$, if $\langle x_k - x_{k-1}, x_{k-1} - x_{k-2}\rangle \ge 0.98\,\|x_k - x_{k-1}\|\,\|x_{k-1} - x_{k-2}\|$;
set $w_k = \eta_2$, otherwise, where $0 < \eta_1 < \eta_2 < \eta_3$. In the numerical experiment, $\eta_1 = 1, \eta_2 = 2, \eta_3 = 10$.

3 FISTA algorithm with the adaptive non-monotone stepsize

Based on the adaptive non-monotone stepsize strategy, we present a accelerated FISTA algorithm. This algorithm enjoys the $O\left(1/k^2\right)$ convergence rate of the objective function value and $o\left(\frac{1}{k}\right)$ convergence rate of the norm of subdifferential of function value.

We present the FISTA algorithm with the adaptive non-monotone stepsize (FISTA_NMS) as follows.

Algorithm 4 FISTA_NMS

```
Step 0. Take $y_1 = x_0 \in R^n, t_1 = 1, 0 < \mu_1 < \mu_0 < 1$ and $\lambda_1 > 0$
Step k. compute
            $x_k = p_{\lambda_k g}(y_k)$
    Set $\lambda_{k+1}$ via the adaptive non-monotone stepsize strategy (Algorithm 3)
            $t_{k+1} = \frac{1+\sqrt{1+4(\lambda_k/\lambda_{k+1})t_k^2}}{2}$
            $y_{k+1} = x_k + ((t_k - 1)/t_{k+1})(x_k - x_{k-1})$
```

Next, we show the convergence result of Algorithm 4. For ease of description, we denote several sequences firstly.
Notation 3.1 Let $\{x_k\}$ and $\{y_k\}$ be generated by the Algorithm 4 and x^* is a fixed minimizer of F. Then, for the convergence of objective function value holds, the sequence $\{v_k\}$ tends to zero when n goes to infinity

$$v_k := F(x_k) - F(x^*). \tag{20}$$

The sequence $\{\delta_k\}$ means the local variation of the sequence $\{x_k\}$

$$\delta_k := \frac{1}{2}\|x_k - x_{k-1}\|^2, \tag{21}$$

and the sequence $\{\varGamma_k\}$, denoting the distance between $\{y_k\}$ and $\left\{p_{\lambda_k g}(y_k)\right\}$, is

$$\varGamma_k := \frac{1}{2}\|x_k - y_k\|^2, \tag{22}$$

and we define $\varPhi_k$ is the distance between $\{x_k\}$ and a fixed minimizer $\{x^*\}$

$$\varPhi_k := \frac{1}{2}\left\|x_k - x^*\right\|^2. \tag{23}$$

These definitions are frequently used in following proofs, and we declare that these definitions are applicable to any algorithm in this paper.

We are now construct a key result and the theoretical analysis of the algorithms proposed in this paper relies heavily on it.

Lemma 3.1 *For any $y \in R^n, \mu_0 \in \left]0,1\right]$, if y and $p_\lambda(y)$ satisfy the condition (9), then, for any $x \in R^n$,*

$$F(x) - F(p_\lambda(y)) \geq \frac{\bar{\mu}}{\lambda}\|p_\lambda(y) - y\|^2 + \frac{1}{\lambda}\langle y - x, p_\lambda(y) - y\rangle. \tag{24}$$

where $\bar{\mu} = 1 - \frac{\mu_0}{2}$, if f is a quadratic function, $\bar{\mu} = 1 - \mu_0$, if f is a non-quadratic function.

Proof. Since f, g are convex, we have

$$\begin{array}{l} f(x) \geq f(y) + \langle x - y, \nabla f(y)\rangle, \\ g(x) \geq g(p_\lambda(y)) + \langle x - p_\lambda(y), \gamma(y)\rangle, \end{array} \tag{25}$$

where $\gamma(y) = -\nabla f(y) - \frac{1}{\lambda}(p_\lambda(y) - y) \in \partial g(p_\lambda(y))$, and $\partial g(\cdot)$ denotes the subdifferential of $g(\cdot)$.
Then,

$$\begin{array}{l} F(x) - F(p_\lambda(y)) \\ = f(x) + g(x) - f(p_\lambda(y)) - g(p_\lambda(y)) \\ \geq f(y) + \langle x - y, \nabla f(y)\rangle + \left\langle p_\lambda(y) - x, \nabla f(y) + \frac{1}{\lambda}(p_\lambda(y) - y)\right\rangle - f(p_\lambda(y)) \\ = f(y) - f(p_\lambda(y)) + \langle p_\lambda(y) - y, \nabla f(y)\rangle + \frac{1}{\lambda}\langle p_\lambda(y) - x, p_\lambda(y) - y\rangle \\ = f(y) - f(p_\lambda(y)) + \langle p_\lambda(y) - y, \nabla f(y)\rangle + \frac{1}{\lambda}\langle y - x, p_\lambda(y) - y\rangle + \frac{1}{\lambda}\|p_\lambda(y) - y\|^2. \end{array} \tag{26}$$

Denote

$$\Delta = f(y) - f(p_\lambda(y)) + \langle p_\lambda(y) - y, \nabla f(y)\rangle + \frac{1}{\lambda}\langle y - x, p_\lambda(y) - y\rangle + \frac{1}{\lambda}\|p_\lambda(y) - y\|^2. \tag{27}$$

The proof is derived by dividing the function into two cases.
1) In the case that f is a quadratic function, without loss of generality, assume that

$$f(x) = \frac{1}{2}x^T Ax + b^T x, \nabla f(x) = Ax + b. \tag{28}$$

It is easy to obtain that

$$f(x) - f(y) = \frac{1}{2}\langle \nabla f(x) + \nabla f(y), x - y\rangle. \tag{29}$$

Then,

$$\begin{array}{l} \Delta = \frac{1}{2}\langle \nabla f(y) + \nabla f(p_\lambda(y)), y - p_\lambda(y)\rangle + \langle \nabla f(y), p_\lambda(y) - y\rangle \\ \qquad\qquad + \frac{1}{\lambda}\langle y - x, p_\lambda(y) - y\rangle + \frac{1}{\lambda}\|p_\lambda(y) - y\|^2 \\ \quad = \frac{1}{2}\langle \nabla f(y) - \nabla f(p_\lambda(y)), p_\lambda(y) - y\rangle + \frac{1}{\lambda}\langle y - x, p_\lambda(y) - y\rangle + \frac{1}{\lambda}\|p_\lambda(y) - y\|^2 \\ \quad \geq \frac{1}{\lambda}\left(1 - \frac{\mu_0}{2}\right)\|p_\lambda(y) - y\|^2 + \frac{1}{\lambda}\langle y - x, p_\lambda(y) - y\rangle. \end{array} \tag{30}$$

2) In the case that f is a non-quadratic function,

$$\begin{array}{l} \Delta \geq \langle \nabla f(p_\lambda(y)), y - p_\lambda(y)\rangle + \langle \nabla f(y), p_\lambda(y) - y\rangle \\ \qquad\qquad + \frac{1}{\lambda}\langle y - x, p_\lambda(y) - y\rangle + \frac{1}{\lambda}\|p_\lambda(y) - y\|^2 \\ \quad = \langle \nabla f(y) - \nabla f(p_\lambda(y)), p_\lambda(y) - y\rangle + \frac{1}{\lambda}\langle y - x, p_\lambda(y) - y\rangle + \frac{1}{\lambda}\|p_\lambda(y) - y\|^2 \\ \quad \geq \frac{1}{\lambda}(1 - \mu_0)\|p_\lambda(y) - y\|^2 + \frac{1}{\lambda}\langle y - x, p_\lambda(y) - y\rangle. \end{array} \tag{31}$$

The last inequalities of (30) and (31) are from the condition (9).

By combining (26), (27), (30) and (31), we can easliy obtain (24). □

Remark 3.1. when $\bar{\mu} \geq \frac{1}{2}$, the result (24) of Lemma 3.1 will reduces to the lemma 2.3 in [4]

$$F(x) - F(p_\lambda(y)) \geq \frac{1}{2\lambda}\|p_\lambda(y) - y\|^2 + \frac{1}{\lambda}\langle y - x, p_\lambda(y) - y\rangle, \tag{32}$$

which plays a crucial role for the analysis of FISTA.

We always choose the value range of $\mu_0 \in \,]0,1[$ for the quadratic function and $\mu_0 \in \left]0,\frac{1}{2}\right[$ for the non-quadratic function, i.e. $\bar{\mu} > \frac{1}{2}$, which means that Lemma 3.1 is a result stronger than Lemma 2.3 of [4].

Further, it follows from the identity

$$\langle a-b, a-c\rangle = \frac{1}{2}\|a-b\|^2 + \frac{1}{2}\|a-c\|^2 - \frac{1}{2}\|b-c\|^2, \tag{33}$$

and (24) that

$$\begin{aligned} F\left(p_\lambda\left(y\right)\right) + \tfrac{\|p_\lambda(y)-x\|^2}{2\lambda} &\le F\left(x\right) + \tfrac{\|y-x\|^2}{2\lambda} - \left(\tfrac{2\bar{\mu}-1}{2\lambda}\right)\|p_\lambda\left(y\right)-y\|^2 \\ &\le F\left(x\right) + \tfrac{\|x-y\|^2}{2\lambda},\ \forall x \in R^n, \bar{\mu} > \tfrac{1}{2}. \end{aligned} \tag{34}$$

We also prove a trivial fact about the $\{t_k\}$ generated by Algorithm 4.

Lemma 3.2 *Let $\{t_k\}$ be generated by Algorithm 4. Then, we obtain that $1/t_k = O\left(1/k\right)$.*

Proof. Rearranging the expression of t_{k+1}, we have $2\sqrt{\lambda_{k+1}}t_{k+1} = \sqrt{\lambda_{k+1}}+\sqrt{\lambda_{k+1}+4\lambda_k t_k^2}$. Denote that $w_k = \sqrt{\lambda_k}t_k$, then $2w_{k+1} = \sqrt{\lambda_{k+1}}+\sqrt{\lambda_{k+1}+4w_k^2}$, it is easy to get that $\{w_k\}$ increasing monotonically.

The following proves that $\lim\limits_{k\to\infty} w_k = +\infty$. Suppose that $\lim\limits_{k\to\infty} w_k = w < +\infty$. Using Lemma 2.1 and Lemma 2.2, denoted $\lim\limits_{k\to\infty} \lambda_k = \lambda^* > 0$, we have $2w = \sqrt{\lambda^*} + \sqrt{\lambda^* + 4w^2}$, which implies a contradiction that $4w^2 - 4w\sqrt{\lambda^*} = 4w^2$. Therefore, $\lim\limits_{k\to\infty} w_k = +\infty$.

Using the Stolz theorem, we deduce

$$\begin{aligned} \lim_{k\to\infty} \tfrac{t_k}{k} &= \lim_{k\to\infty} \tfrac{w_k}{\sqrt{\lambda_k}k} = \tfrac{1}{\sqrt{\lambda^*}} \lim_{k\to\infty}\left(w_{k+1}-w_k\right) \\ &= \tfrac{1}{\sqrt{\lambda^*}} \lim_{k\to\infty} \tfrac{2\sqrt{\lambda_{k+1}}}{\left(\sqrt{\lambda_{k+1}/w_k^2+4}+2-\sqrt{\lambda_{k+1}/w_k^2}\right)} = \tfrac{1}{2}. \end{aligned} \tag{35}$$

Hence, we have $1/t_k = O\left(1/k\right)$. □

In the following theorem, we show that the convergence rate of the objective function value and some other results which will be use to prove the convergence of $\{x_k\}$ generated by Algorithm 4.

Theorem 3.1 *(Convergence Rate) Let $\{x_k\},\{y_k\}$ be generated by the Algorithm 4. Then, (a)*

$$F\left(x_k\right) - F\left(x^*\right) \le O\left(1\big/k^2\right), \quad \forall x^* \in X^* \text{ and } \forall k \ge 1. \tag{36}$$

(b) The series $\sum\limits_{k=1}^{\infty} k^2\|x_k - y_k\|^2$ is convergent and $\liminf\limits_{k\to\infty} k^{1.5}\left\|x_k - y_k\right\| = 0$.

Proof. Invoking Lemma 2.2 and Lemma 3.1, we obtain that (24) holds for every $k \ge k_0$.

Denote that $u_k = t_k x_k - \left(t_k - 1\right)x_{k-1} - x^*$. We apply the inequality (24) at the points $(x := x_k, y := y_{k+1})$ with $\lambda := \lambda_{k+1}$, and likewise at the points $\left(x := x^*, y := y_{k+1}\right)$, to get

$$\lambda_{k+1}\left(v_k - v_{k+1}\right) \ge \bar{\mu}\|x_{k+1} - y_{k+1}\|^2 + \langle x_{k+1} - y_{k+1}, y_{k+1} - x_k\rangle, \tag{37}$$

$$-\lambda_{k+1}v_{k+1} \ge \bar{\mu}\|x_{k+1} - y_{k+1}\|^2 + \left\langle x_{k+1} - y_{k+1}, y_{k+1} - x^*\right\rangle. \tag{38}$$

Multiplying the first inequality above by $(t_{k+1} - 1)$ and adding it to the second inequality, we have

$$\begin{aligned} &\lambda_{k+1}\left(\left(t_{k+1}-1\right)v_k - t_{k+1}v_{k+1}\right) \\ &\ge \bar{\mu}t_{k+1}\|x_{k+1} - y_{k+1}\|^2 + \langle x_{k+1} - y_{k+1}, t_{k+1}y_{k+1} - \left(t_{k+1}-1\right)x_k - x^*\rangle. \end{aligned} \tag{39}$$

Further, multiplying (39) by t_{k+1}, and from the definition of t_k, we obtain

$$\begin{aligned}
&\lambda_k t_k^2 v_k - \lambda_{k+1} t_{k+1}^2 v_{k+1} \\
&\geq \bar{\mu} \|t_{k+1}\left(x_{k+1} - y_{k+1}\right)\|^2 + \left\langle t_{k+1}\left(x_{k+1} - y_{k+1}\right), t_{k+1} y_{k+1} - \left(t_{k+1} - 1\right) x_k - x^* \right\rangle \\
&= \tfrac{1}{2} \|t_{k+1}\left(x_{k+1} - y_{k+1}\right)\|^2 + \left\langle t_{k+1}\left(x_{k+1} - y_{k+1}\right), t_{k+1} y_{k+1} - \left(t_{k+1} - 1\right) x_k - x^* \right\rangle \\
&\qquad\qquad + \left(\bar{\mu} - \tfrac{1}{2}\right) \|t_{k+1}\left(x_{k+1} - y_{k+1}\right)\|^2 \\
&= \tfrac{1}{2}\left(\|u_{k+1}\|^2 - \|u_k\|^2\right) + \left(\bar{\mu} - \tfrac{1}{2}\right) \|t_{k+1}\left(x_{k+1} - y_{k+1}\right)\|^2.
\end{aligned} \tag{40}$$

Define the quantities $a_k = \lambda_k t_k^2 v_k, b_k = \frac{1}{2}\|u_k\|^2$. The above inequality (40) can be rewritten by

$$a_k - a_{k+1} \geq \left(b_{k+1} - b_k\right) + \left(\bar{\mu} - \frac{1}{2}\right) \|t_{k+1}\left(x_{k+1} - y_{k+1}\right)\|^2 \geq b_{k+1} - b_k. \tag{41}$$

It is not difficult to show that there exists a constant $c > 0$ such that

$$a_k + b_k \leq a_{k_0} + b_{k_0} \leq c, \tag{42}$$

which implies that $\lambda_k t_k^2 v_k \leq c$. Applying (10), we have

$$v_k = F\left(x_k\right) - F\left(x^*\right) \leq \frac{c}{\lambda_k t_k^2} \leq \frac{c}{\lambda_{\min} t_k^2}, \tag{43}$$

then, Lemma 3.2 yields the result that $F\left(x_k\right) - F\left(x^*\right) \leq O\left(1/k^2\right)$.

Rearranging (41) we see that

$$\left(a_k + b_k\right) - \left(a_{k+1} + b_{k+1}\right) \geq \left(\bar{\mu} - \frac{1}{2}\right) \|t_{k+1}\left(x_{k+1} - y_{k+1}\right)\|^2.$$

Summing the inequality from $k = N_s$ to $k = N$, where N_s is a sufficient large positive integer, we obtain that

$$\left(a_{N_s} + b_{N_s}\right) - \left(a_{N+1} + b_{N+1}\right) \geq \left(\bar{\mu} - \frac{1}{2}\right) \sum_{k=N_s}^{N} \|t_{k+1}\left(x_{k+1} - y_{k+1}\right)\|^2, \tag{44}$$

where $\bar{\mu} - \frac{1}{2} > 0$ based on the choice of μ_0. Then, from $a_k + b_k > 0$ and Lemma 3.2 we have $\sum\limits_{k=1}^{\infty} k^2 \|x_k - y_k\|^2$ is convergent. Further, we can easily obtain that $\liminf\limits_{k\to\infty} k^{1.5} \|x_k - y_k\| = 0$. □

Lemma 3.3 *[4] For any $y \in R^n$, one has $z = p_{\lambda g}\left(y\right)$ if and only if there exists $\sigma\left(y\right) \in \partial g\left(z\right)$ the subdifferential of $g\left(\cdot\right)$, such that*

$$\nabla f\left(y\right) + \frac{1}{\lambda}\left(z - y\right) + \sigma\left(y\right) = 0.$$

Remark 3.2. Denote $\psi_k = \nabla f\left(x_k\right) - \frac{1}{\lambda_k}\left(x_k - y_k + \lambda_k \nabla f\left(y_k\right)\right)$. Based on Lemma 3.3, we have $\psi_k \in \partial F\left(x_k\right)$. It follows from Lemma 2.1, Lemma 3.2, the conclusion $\lim\limits_{k\to\infty} k^2 \|x_k - y_k\|^2 = 0$ and the fact that $\|\psi_k\| \leq \left(L_f + 1/\lambda_{\min}\right) \|x_k - y_k\|$ that $\lim\limits_{k\to\infty} k \|\psi_k\| = 0$, which implies that $\|\psi_k\| = o\left(\frac{1}{k}\right)$. However, for the FISTA, we deduce that $t_k^2 \|x_k - y_k\|^2$ is bounded from the proof of Lemma 4.1 in [4], i.e. $\|\psi_k\| = O\left(\frac{1}{k}\right)$. Hence, the sequence $\{\|\psi_k\|\}$ generated by Algorithm 4 converges to zero faster than the one generated by FISTA. In Section 6, the numerical performances verify this.

In following theorem, we will show that the sequence $\{x_k\}$ exists at least one accumulation point, and any accumulation point belongs to X^*.

Theorem 3.2 *For $\forall k \geq 1$, we have the sequence $\{x_k\}$ generated from Algorithm 4 is bounded, and all the accumulation points of $\{x_k\}$ belongs to X^*.*

Proof. From (42), we have that $\{b_k\}$ is bounded for any $k \geq k_0$.

With the definition of b_k and trigonometric inequality, we see that

$$\left\|x_k - x^*\right\| \leq \frac{\sqrt{2b_k}}{t_k} + \left(1 - \frac{1}{t_k}\right) \left\|x_{k-1} - x^*\right\| \leq \frac{\sqrt{2c}}{t_k} + \left(1 - \frac{1}{t_k}\right) \left\|x_{k-1} - x^*\right\|. \tag{45}$$

Let $M_0 = \max\left(2c, \left\|x_{k_0} - x^*\right\|\right)$. Then, we can easily prove that $\|x_k - x^*\| \leq M_0$ by induction, which implies $\{x_k\}$ is bounded. Assume that $\left\{x_{k_j}\right\}$ is a convergent subsequence of $\{x_k\}$ and $\lim\limits_{j\to\infty} x_{k_j} = \bar{x}$.

In view of (36) and F is lower semi-continuous, we see that

$$F\left(\bar{x}\right) \leq \liminf_{j\to\infty} F\left(x_{k_j}\right) = \lim_{j\to\infty} F\left(x_{k_j}\right) = F\left(x^*\right). \tag{46}$$

Combining this with the fact that $F\left(\bar{x}\right) \geq F\left(x^*\right)$, we have $F\left(\bar{x}\right) = F\left(x^*\right)$, which means that $\bar{x} \in X^*$. □

4 Modified FISTA algorithm with the adaptive non-monotone stepsize

As mentioned in Section 1, Chambolle and Dossal [11] exploited a new $\gamma_k = \frac{t_k - 1}{t_{k+1}}$ with $t_k = \frac{k+a-1}{a}$, $a > 2$ for FISTA, and establish the convergence of the iterates generated by FISTA with this new parameter γ_k and a constant stepsize $\lambda_k \equiv \frac{1}{L_f}$ (FISTA_CD). Attouch and Peypouquet [1] proved that the convergence rate of function value of FISTA_CD is actually $o\left(\frac{1}{k^2}\right)$, better than $O\left(\frac{1}{k^2}\right)$ of FISTA. Moreover, FISTA_CD has a better numerical performance than FISTA.

Based on the above analysis, we present the modified FISTA algorithm with the new adaptive non-monotone stepsize (MFISTA_NMS) as follows.

Algorithm 5 MFISTA_NMS

```
Step 0. Take $y_1 = x_0 \in R^n, 0 < \mu_1 < \mu_0 \leq 1, a > 2$, and $\lambda_1 > 0$
Step k. Compute
            $x_k = p_{\lambda_k g}\left(y_k\right)$
            $y_{k+1} = x_k + \left(\frac{k-1}{k+a}\right)\left(x_k - x_{k-1}\right)$        (47)
    Set $\lambda_{k+1}$ via the Algorithm 3.
```

From Theorem 3.1, it's easy to see that if the sequence $\{t_k\}$ satisfies

$$\rho_k = \lambda_k t_k^2 - \lambda_{k+1}\left(t_{k+1}^2 - t_{k+1}\right) \geq 0, \tag{48}$$

where $t_1 = 1$ and $\{\lambda_k\}$ is generated by the Algorithm 3, then the objective function value has the $O\left(1/k^2\right)$ convergence rate. Particularly, for $t_{k+1} = \frac{1+\sqrt{1+4(\lambda_k/\lambda_{k+1})t_k^2}}{2}$ from Algorithm 4, one have that $\rho_k = 0$. The following result is based on the analysis of ρ_k.

Lemma 4.1 *Let $\{x_k, y_k\}$ be the sequences generated via Algorithm 5. Assume that $\sum\limits_{k=1}^{\infty} E_k$ is a convergent nonnegative series and $\{E_k\}$ decreasing monotonically. We obtain the following conclusions.*

(a) The series $\sum\limits_{k=1}^{\infty} k\left(F\left(x_k\right) - F\left(x^*\right)\right)$ is convergent.

(b) The series $\sum\limits_{k=1}^{\infty} k^2\|x_k - y_k\|^2$ is convergent and $\liminf\limits_{k\to\infty} k^{1.5}\left\|x_k - y_k\right\| = 0$.

Proof. From the fact that $\sum\limits_{k=1}^{\infty} E_k$ is a convergent nonnegative series and $\{E_k\}$ decreasing monotonically, we can easily obtain that $\lim\limits_{k\to\infty} kE_k = 0$. Then, the following equality

$$\begin{aligned}
\rho_k &= \tfrac{1}{a^2}\left(\lambda_{k-1}(k+a-2)^2 - \lambda_k\,(k-1)\,(k+a-1)\right)\\
&= \tfrac{1}{a^2}\left(\lambda_{k-1}(k+a-2)^2 - \lambda_k\left((k+a-2)^2 + (2-a)\,(k+a-2) + 1 - a\right)\right)\\
&= \tfrac{1}{a^2}\left((\lambda_{k-1}-\lambda_k)\,(k+a-2)^2 + \lambda_k\,((a-2)\,(k+a-2) + a - 1)\right)\\
&= \tfrac{1}{a^2}\left(-\left|\lambda_{k-1}-\lambda_k\right|(k+a-2)^2 + \lambda_k\,((a-2)\,(k+a-2) + a - 1)\right)\\
&= \tfrac{1}{a^2}\left(-\lambda_{k-1}\cdot E_{k-1}\cdot(k+a-2)^2 + \lambda_k\,((a-2)\,(k+a-2) + a - 1)\right)
\end{aligned} \tag{49}$$

yields that $\lim\limits_{k\to\infty} \frac{\rho_k}{k} = \frac{a-2}{a^2}\lambda^* \geq \omega_3$, where $\omega_3 = \frac{(a-2)}{a^2}\lambda_{\min}$.

Invoking (37)–(39) and combining $\lim\limits_{k\to\infty} \frac{\rho_k}{k} \geq \omega_3$, we have $\frac{\rho_k}{k} \geq \frac{\omega_3}{2}$ for all k sufficiently large, and

$$\begin{aligned}
&\lambda_k t_k^2 v_k - \lambda_{k+1} t_{k+1}^2 v_{k+1}\\
&\geq \bar{\mu}\|t_{k+1}\,(x_{k+1}-y_{k+1})\|^2 + \tfrac{\omega_3}{2} k v_k\\
&\qquad\qquad + \langle t_{k+1}\,(x_{k+1}-y_{k+1})\,, t_{k+1}y_{k+1} - (t_{k+1}-1)\,x_k - x^*\rangle\\
&= \tfrac{1}{2}\|t_{k+1}\,(x_{k+1}-y_{k+1})\|^2 + \left(\bar{\mu}-\tfrac{1}{2}\right)\|t_{k+1}\,(x_{k+1}-y_{k+1})\|^2 + \tfrac{\omega_3}{2} k v_k\\
&\qquad\qquad + \langle t_{k+1}\,(x_{k+1}-y_{k+1})\,, t_{k+1}y_{k+1} - (t_{k+1}-1)\,x_k - x^*\rangle\\
&= \tfrac{1}{2}\left(\|u_{k+1}\|^2 - \|u_k\|^2\right) + \left(\bar{\mu}-\tfrac{1}{2}\right)\|t_{k+1}\,(x_{k+1}-y_{k+1})\|^2 + \tfrac{\omega_3}{2} k v_k,
\end{aligned} \tag{50}$$

where $u_k = t_k x_k - (t_k - 1)\,x_{k-1} - x^*$. We rearrange (50) into

$$\left(\lambda_k t_k^2 v_k + \frac{1}{2}\|u_k\|^2\right) - \left(\lambda_{k+1} t_{k+1}^2 v_{k+1} + \frac{1}{2}\|u_{k+1}\|^2\right) \geq \frac{\omega_3}{2} k v_k,$$

and

$$\left(\lambda_k t_k^2 v_k + \tfrac{1}{2}\|u_k\|^2\right) - \left(\lambda_{k+1} t_{k+1}^2 v_{k+1} + \tfrac{1}{2}\|u_{k+1}\|^2\right) \geq \left(\bar{\mu}-\tfrac{1}{2}\right)\|t_{k+1}\,(x_{k+1}-y_{k+1})\|^2.$$

Summing the above two inequalities from $k = N_s$ to $k = N$, where N_s is sufficient large, yields that $\sum\limits_{k=1}^{\infty} k\,(F\,(x_k) - F\,(x^*))$ and $\sum\limits_{k=1}^{\infty} t_k^2\|x_k - y_k\|^2$ are convergent. With the definition of $t_k = \frac{k+a-1}{a}\,(a>2)$, we can further obtain that $\sum\limits_{k=1}^{\infty} k^2\|x_k - y_k\|^2$. In addition, we can easily show that $\lim\limits_{k\to\infty}\inf k^{1.5}\,\|x_k - y_k\| = 0$. □

Remark 4.1. It follows that $\sum\limits_{k=1}^{\infty} k^2\|\psi_k\|^2$ is convergent from Lemma 4.1 (*b*), which is stronger than the fact follows [1] that $\|\psi_k\| = o\left(\frac{1}{k}\right)$ for FISTA_CD.

Lemma 4.2 *Let $\{x_k\}$ be generated by Algorithm 5. Then, the series $\sum\limits_{k=1}^{\infty} k\delta_k$ is convergent.*

Proof. From (34) with $x := x_k, y := y_{k+1}$, we have that

$$\frac{\delta_{k+1}}{\lambda_{k+1}} - \gamma_k^2\frac{\delta_k}{\lambda_k} \leq v_k - v_{k+1}, \tag{51}$$

where $\gamma_k = (k-1)\,/\,(k+a)\,.$

Multiplying this inequality by $(k+a)^2$ and summing from $k = N_s$ to $k = N$, where N_s sufficient large, leads to

$$\begin{aligned}
&(N+a)^2\tfrac{\delta_{N+1}}{\lambda_{N+1}} - (k_0-1)^2\tfrac{\delta_{k_0}}{\lambda_{k_0}} + \sum_{k=k_0+1}^{N} a\,(2k-2+a)\,\tfrac{\delta_k}{\lambda_k}\\
&\qquad\leq (k_0+a)^2 v_{k_0} - (N+a)^2 v_{N+1} + \sum_{k=k_0+1}^{N}(2k+2a-1)\,v_k.
\end{aligned} \tag{52}$$

From Lemma 4.1 (a) and $a > 2$, the series $\sum\limits_{k=1}^{\infty} k\frac{\delta_k}{\lambda_k}$ is convergence. Further, we obtain that $\sum\limits_{k=1}^{\infty} k\delta_k$ is convergence by using $\lim\limits_{k\to\infty} \lambda_k = \lambda^* > 0$. □

Next, we construct the convergence rate of function value generated by the Algorithm 5.

Theorem 4.1 *For the sequence $\{x_k\}$ generated by Algorithm 5, we have $F(x_k) - F(x^*) = o\left(\frac{1}{k^2}\right)$ and $\|x_k - x_{k-1}\| = o\left(\frac{1}{k}\right)$.*

Proof. Denote $\phi_k = \nu_k + \frac{\delta_k}{\lambda_k}$.

From (51) and $\gamma_k \le 1$, we can easily deduce that $\phi_{k+1} \le \phi_k$ for $k \ge k_0$. Multiplying by $(k+1)^2$ we have

$$(k+1)^2\phi_{k+1} \le (k+1)^2\phi_k = k^2\phi_k + (2k+1)\phi_k. \tag{53}$$

It follows that $\sum\limits_{k=1}^{\infty} k\phi_k$ is convergent from Lemma 4.1 (a) and Lemma 4.2. Then, we can prove that $\left\{k^2\phi_k\right\}$ is convergent, the proof is similar with the proof of the convergence of $\{\lambda_k\}$ in Lemma 2.1. Further, we have $\liminf\limits_{k\to\infty} k^2\phi_k = 0$ by using the convergence of $\sum\limits_{k=1}^{\infty} k\phi_k$. Hence, $\lim\limits_{k\to\infty} k^2\phi_k = 0$, which implies $F(x_k) - F(x^*) = o\left(\frac{1}{k^2}\right)$ and $\|x_k - x_{k-1}\| = o\left(\frac{1}{k}\right)$ follows the Lemma 2.1. □

Now, we give the proof of convergence of the sequence $\{x_k\}$ generated by Algorithm 5. Before that, we give some auxiliary results.

Lemma 4.3 *For $\forall k \ge 1$, we have the sequence $\{x_k\}$ generated from Algorithm 5 is bounded, and all the accumulation points of $\{x_k\}$ belongs to X^*.*

Proof. The proof is similar with the Theorem 3.2.

Lemma 4.4 *For any $x^* \in X^*$, and the sequence $\{x_k\}$ generated by Algorithm 5, we have $\varPhi_k = \frac{1}{2}\|x_k - x^*\|^2$ is convergent.*

Proof. Recalling (50) in the proof of lemma 4.1, we have $a_k + b_k \ge a_{k+1} + b_{k+1}$ for all $k \ge k_0$, where $a_k := \lambda_k t_k^2 v_k$ and $b_k := \frac{1}{2}\|(t_k - 1)(x_k - x_{k-1}) + (x_k - x^*)\|^2$. Combining this and the fact that $a_k + b_k \ge 0$, we can easily deduce that the sequence $\{a_k + b_k\}$ is convergent.

With Lemma 2.1, Lemma 3.2 and $\lim\limits_{k\to\infty} k^2(F(x_k) - F(x^*)) = 0$ from Theorem 4.1, we obtain that $\lim\limits_{k\to\infty} a_k = 0$, which implies that $\{b_k\}$ is convergent. From the definition of $\{b_k\}$, we see that

$$\begin{aligned} b_k &= \tfrac{1}{2}\|(t_k - 1)(x_k - x_{k-1}) + (x_k - x^*)\|^2 \\ &= \tfrac{1}{2}(t_k - 1)^2\|x_k - x_{k-1}\|^2 + \langle (t_k - 1)(x_k - x_{k-1}), (x_k - x^*)\rangle + \tfrac{1}{2}\|x_k - x^*\|^2. \end{aligned} \tag{54}$$

It follows from the Lemma 3.2 and Theorem 4.1 that the first item of (54) converges to zero, i.e.

$$\lim_{k\to\infty} \frac{1}{2}(t_k - 1)^2\|x_k - x_{k-1}\|^2 = 0. \tag{55}$$

In addition, from Lemma 3.2, Theorem 4.1 and the fact from Lemma 4.3 that $\|x_k - x^*\|$ is bounded, we have

$$\lim_{k\to\infty} \left\langle (t_k - 1)(x_k - x_{k-1}), (x_k - x^*)\right\rangle = 0. \tag{56}$$

With (54), (55), (56) and the fact that $\{b_k\}$ is convergent, we can obtain that $\varPhi_k = \frac{1}{2}\|x_k - x^*\|^2$ is convergent. □

Theorem 4.2 *The sequence $\{x_k\}$ generated by Algorithm 5 converges to a minimizer of F.*

Proof. From Lemma 4.3, we have $\lim\limits_{j\to\infty} x_{k_j} = \bar{x} \in X^*$. And with the result of Lemma 4.4, we have $\|x_k - \bar{x}\|^2$ is convergent by using $\bar{x}$ to replace x^*. Then, easily to deduce that the sequence $\left\{\|x_k - \bar{x}\|^2\right\}$ converge to zero, which implies $\{x_k\}$ convergent to a minimizer of F. □

It is worth mentioning that the biggest difference between Algorithm 5 and FISTA_CD is that it is not required to do any assumption related to the L_f, while the condition $\lambda \in \left]0, \frac{1}{L_f}\right]$ in FISTA_CD plays an important role in algorithm implementation and theoretical analysis.

Meanwhile, it is unclear that whether the iterative sequence generated by FISTA_CD with the backtracking stepsize converges. However, MFISTA_NMS keeps the similar theoretical results with FISTA_CD including the convergence rate of objective function value and the convergence of iterative sequence.

In the above analysis, we proved that for the Algorithm 3 and Algorithm 4, function values keep similar convergence rate with FISTA and FISTA_CD, but the convergence of iterates generated by FISTA or FISTA_NMS is still unknown. It is widely known that error bound condition is a key ingredient in proving convergence of iterative methods. Major contributions on developing and using error bound condition to derive rates of convergence rate of iterative descent algorithms have been developed in a series of papers [3,14,15,27, 29].

Assumption 3.1: (Error Bound Condition) For any $\xi \geq F^*$, there exist a $\varepsilon > 0$ and $\bar{\tau} > 0$ such that

$$dist\left(x, X^*\right) \leq \bar{\tau} \left\| p_{\frac{1}{L_f} g}\left(x\right) - x \right\| \tag{57}$$

whenever $\left\| p_{\frac{1}{L_f} g}\left(x\right) - x \right\| < \varepsilon$ and $F\left(x\right) \leq \xi$.

In [18], under the error bound condition, the author uses a *comparison method* to prove the convergence of iterates generated by FISTA and FISTA_CD with constant stepsize, similar results can be derived for the ones with Algorithm 3.

Corollary 4.1 *Suppose that Assumption 3.1 holds. Let $\{x_k\}$ be generated by Algorithm 4 and $x^* \in X^*$. Then,*
1) $F\left(x_k\right) - F\left(x^*\right) = o\left(\frac{1}{k^6}\right)$ and $\|x_k - x_{k-1}\| = O\left(\frac{1}{k^3}\right)$.
2) $\{x_k\}$ sublinearly converges to $\bar{x} \in X^*$ at the $O\left(\frac{1}{k^2}\right)$ rate of convergence.

Corollary 4.2 *Suppose that Assumption 3.1 holds. Let $\{x_k\}$ be generated by Algorithm 5 and $x^* \in X^*$. Then,*
1) $F\left(x_k\right) - F\left(x^*\right) = o\left(\frac{1}{k^{2(a+1)}}\right)$ and $\|x_k - x_{k-1}\| = O\left(\frac{1}{k^{a+1}}\right)$.
2) $\{x_k\}$ sublinearly converges to $\bar{x} \in X^*$ at the $O\left(\frac{1}{k^a}\right)$ rate of convergence.

The proofs of Corollary 4.1 and Corollary 4.2 follow the proof of Theorem 2.6 in [18]. Difference from the constant stepsize setting in [18], both of Algorithm 4 and Algorithm 5 are based on the new adaptive nonmonotone stepsize setting, which is convergent and have a property that increasing monotonically after finite iterations. We point out that the main of proof of Theorem 2.6 in [18] is the following inequality

$$\begin{aligned} F\left(x_{k+1}\right) - F\left(x^*\right) + \frac{1-\mu}{2\lambda_{k+1}}\|x_{k+1} - y_{k+1}\|^2 + \frac{1}{2\lambda_{k+1}}\|x_{k+1} - x_k\|^2 \\ \leq F\left(x_k\right) - F\left(x^*\right) + \frac{\gamma_k^2}{2\lambda_k}\|x_k - x_{k-1}\|^2, \end{aligned}$$

which is obtained by applying (34) at $y := y_{k+1}$, $x := x_k$ and $\lambda := \lambda_{k+1}$, and the fact that $\lambda_{k+1} \geq \lambda_k$ for k is sufficient large. Here, we omit the remaining proof.

It is noted that the stepsize λ_k generated by Algorithm 3 increases monotonically after finite iterations, while the stepsize λ_k generated by backtracking of FISTA BKTR may increases or decreases in the backtracking process. Meanwhile, we can not obtain a similar inequality with (34) based on FISTA_BKTR. Hence, using the same idea of proof in [18], backtracking of FISTA_BKTR can not obtain the results in Corollary 4.1 and Corollary 4.2, and to our knowledge, there don't have similar results in the literature. From this point of view, FISTA with λ_k generated by the new stepsize strategy (Algorithm 3) enjoys better theoretical properties than FISTA with backtracking in Algorithm 2 (FISTA_BETR).

To further illustrate this point, we consider a restart technique, which is crucially important in improving the theoretical results and accelerating the numerical performance of the algorithm, to improve our algorithms in next section.

5 Restart FISTA algorithm with the new non-monotone stepsize strategy

Brendan and Emmanuel [23] introduced two simple heuristic adaptive restart techniques that can improve the convergence rate of accelerated gradient schemes. One restart technique is fixed restarting, that restarting the algorithm every K iterations and taking the last point generated by the algorithm as the starting point. Another is the adaptive restart, which starting the algorithm based on the following schemes: 1) function scheme: $F(x_k) > F(x_{k-1})$; 2)gradient scheme: $(y_k - x_k)^T (x_k - x_{k-1}) > 0$.

Brendan and Emmanuel pointed out that both of the two adaptive restart schemes perform similarly well. But when the iteration point is close to the minimum, the algorithm with the gradient restart technique is more numerically stable. Therefore, we combine the fixed restarting with the gradient restart technique to improve the performance of FISTA_NMS and MFISTA_NMS in this section.

We present algorithms as follows.

Algorithm 6 FISTA_NMS_restart

```
Step 0. Given $K \in R$ and take $y_1 = x_0 \in R^n, t_1 = 1, 0 < \mu_1 < \mu_0 \leq 1$, $\lambda_1 > 0$ and $\tilde{k} = 1$
Step k. Compute
            $x_k = p_{\lambda_k g}(y_k)$
    Set $\tilde{k} = \tilde{k} + 1$ and compute $\lambda_{k+1}$ via Algorithm 3.
            If $(y_k - x_k)^T (x_k - x_{k-1}) > 0$ or $\tilde{k} = K$ holds, set $t_k = 1, \tilde{k} = 1$
            $t_{k+1} = \left(1 + \sqrt{1 + 4(\lambda_k/\lambda_{k+1}) t_k^2}\right)$        (58)
            $y_{k+1} = x_k + ((t_k - 1)/t_{k+1})(x_k - x_{k-1})$.
```

Algorithm 7 MFISTA_NMS_restart

```
Step 0. Given $K \in R$ and take $y_1 = x_0 \in R^n, 0 < \mu_1 < \mu_0 \leq 1, a > 2, \lambda_1 > 0$ and $\tilde{k} = 1$.
Step k. Compute
            $x_k = p_{\lambda_k g}(y_k)$
    Set $\tilde{k} = \tilde{k} + 1$
            If $(y_k - x_k)^T (x_k - x_{k-1}) > 0$ or $\tilde{k} = K$ holds, set $\tilde{k} = 1$
            $y_{k+1} = x_k + \left(\frac{\tilde{k} - 1}{\tilde{k} + a}\right)(x_k - x_{k-1})$        (59)
    Compute $\lambda_{k+1}$ via Algorithm 3.
```

The schemes of FISTA_BKTR and FISTA_CD_BKTR combining the restart strategy separately namely FISTA_BKTR_restart and FISTA_CD_BKTR_restart are similar to the above two algorithms. Here we omit the unnecessary details. In the following, we prove that under the error bound condition, the sequences generated by Algorithm 6 and Algorithm 7 are R-linearly convergent; Moreover, the corresponding sequences of objective values are also R-linearly convergent. Note that whether the FISTA_BKTR with restart strategy enjoy similar convergence results is unknown.

Before proceeding with the convergence results, we give some auxiliary conclusions as follows.

Definition 5.1 *[29] For a sequence $\{x_k\}$, we say that x_k is $Q-$linearly to its limit if there exist $0 < c < 1$ and k_l such that*

$$\left\|x_{k+1} - x^*\right\| \le c \left\|x_k - x^*\right\|, \quad \forall k \ge k_l$$

and we say that x_k converges $R-$linearly to its limit if

$$\lim_{k\to\infty} sup \left\|x_k - x^*\right\|^{\frac{1}{k}} < 1.$$

Lemma 5.1 *[29] Suppose that $\{p_k\}$ and $\{q_k\}$ be two sequences with $0 \le p_k \le q_k$ and $\{q_k\}$ is $Q-$linearly convergent to zero. Then $\{p_k\}$ is $R-$linearly convergent to zero.*

Lemma 5.2 *Let $\{A_k\}$, $\{B_k\}$ and $\{C_k\}$ be three nonnegative sequences. Suppose that there exist $0 < \tau < 1, l > 0$ and $k_l > 0$ such that $A_{k+1} + B_{k+1} + C_{k+1} \le A_k + \tau B_k$ and $A_k \le lC_k$ hold for any $k > k_l$, we have $\{A_{k+1} + \alpha B_{k+1}\}$ is $Q-$linear convergent to zero, where $\alpha = \min\left(\frac{1}{1+\frac{1}{l}}, \tau\right)$. And both of $\{A_k\}$ and $\{B_k\}$ are $R-$linear convergent to zero.*

Proof. We can easy to deduce that for any $k > k_l$,

$$\left(1 + \frac{1}{l}\right) A_{k+1} + B_{k+1} \le A_k + \tau B_k. \tag{60}$$

Denote $\alpha = \min\left(\frac{1}{1+\frac{1}{l}}, \tau\right)$ and $\beta = \max\left(\frac{1}{1+\frac{1}{l}}, \tau\right)$. Using the definition of α and β and (60), we obtain

$$A_{k+1} + \alpha B_{k+1} \le A_{k+1} + \left(\frac{1}{1+\frac{1}{l}}\right) B_{k+1} \le \left(\frac{1}{1+\frac{1}{l}}\right) A_k + \left(\frac{\tau}{1+\frac{1}{l}}\right) B_k \le \beta \left(A_k + \alpha B_k\right), \tag{61}$$

which means that $\{A_k + \alpha B_k\}$ is $Q-$linearly convergent to zero.

Further, we can deduce that $\{A_k\}$ and $\{B_k\}$ are $R-$linearly convergent to zeros using Lemma 5.1. □

Lemma 5.3 *[21] For $L_1 \ge L_2 > 0$, we have*

$$\left\|G^{f,g}_{L_1}(x)\right\| \ge \left\|G^{f,g}_{L_2}(x)\right\| \quad \text{and} \quad \frac{\left\|G^{f,g}_{L_1}(x)\right\|}{L_1} \le \frac{\left\|G^{f,g}_{L_2}(x)\right\|}{L_2}, \tag{62}$$

where $G^{f,g}_{L_f}(x) = L_f\left(p_{\frac{1}{L_f}g}(x) - x\right)$.

Theorem 5.1 *Suppose that Assumption 3.1 holds. Then, both of the sequences $\{x_k\}$ generated by the Algorithm 6 and Algorithm 7 are convergence and $R-$linearly convergent to their limit. Also, $\{F(x_k)\}$ are $R-$linearly convergent to $F(x^*)$.*

Proof. For the t_k generated by Algorithm 6, it follows from

$$t_{k+1}-1=\frac{\sqrt{1+4\left(\frac{\lambda_k}{\lambda_{k+1}}\right)t_k^2}-1}{2}<\frac{\sqrt{1+4t_k^2}-1}{2}<t_k,\ \forall k \text{ sufficient large} \tag{63}$$

that there exists a $\hat{M}$ such that $t_k \le \hat{M}$. Based on Lemma 2.1 and Corollary 2.1, we have

$$0 \le 1-\frac{\lambda_k}{\lambda_{k+1}} \le \frac{1}{\hat{M}} \le \frac{1}{t_k},$$

holds for sufficient large k. Then,

$$\begin{aligned} t_{k+1}-t_k &= \frac{1+\sqrt{1+4\frac{\lambda_k}{\lambda_{k+1}}t_k^2}}{2}-t_k \\ &= \frac{-4\left(1-\frac{\lambda_k}{\lambda_{k+1}}\right)t_k^2+4t_k}{2\left(\sqrt{1+4\frac{\lambda_k}{\lambda_{k+1}}t_k^2}+2t_k-1\right)} \ge \frac{-4t_k+4t_k}{2\left(\sqrt{1+4\frac{\lambda_k}{\lambda_{k+1}}t_k^2}+2t_k-1\right)}=0. \end{aligned}$$

Further, it's easy to show that

$$\gamma_k=\frac{t_k-1}{t_{k+1}} \le \frac{t_k-1}{t_k}=1-\frac{1}{t_k} \le \frac{\hat{M}-1}{\hat{M}}<1.$$

From Algorithm 7, it is obvious that $\gamma_k=\frac{k-1}{k+a} \le \frac{K-1}{K+a}<1$. Thus, either algorithm 6 or algorithm 7, there exists a $\bar{\gamma}$ such that $\gamma_k \le \bar{\gamma}<1$.

Denote N_s is a sufficient large positive integer. Let $\xi = v_{N_s}+\frac{\delta_{N_s}}{\lambda_{N_s}}+F\left(x^*\right)$. From the Assumption 3.1, we can deduce that for this ξ, there exist a $\varepsilon>0$ and $\bar{\tau}>0$ such that $dist\left(x,X^*\right) \le \bar{\tau}\left\|p_{\frac{1}{L_f}g}\left(x\right)-x\right\|$ holds for $\left\|p_{\frac{1}{L_f}g}\left(x\right)-x\right\|<\varepsilon$ and $F\left(x\right) \le \xi$.

From (34), we obtain that

$$v_{k+1}+\frac{\delta_{k+1}}{\lambda_{k+1}}+\frac{2\bar{\mu}-1}{\lambda_{k+1}}\varGamma_{k+1} \le v_k+\bar{\gamma}^2\frac{\delta_k}{\lambda_k}. \tag{64}$$

It's easy to get

$$v_{k+1}+\frac{\delta_{k+1}}{\lambda_{k+1}} \le v_k+\frac{\delta_k}{\lambda_k}, \tag{65}$$

which means that for k sufficient large, $\left\{v_k+\frac{\delta_k}{\lambda_k}\right\}$ is nonincreasing.
This together with the fact that $\left\{v_k+\frac{\delta_k}{\lambda_k}\right\}$ is bound below deduces to $\left\{v_k+\frac{\delta_k}{\lambda_k}\right\}$ is convergent.

Recalling (64), for $k \ge N_s$ we have

$$\left(\frac{2\bar{\mu}-1}{\lambda_{k+1}}\right)\varGamma_{k+1} \le \left(v_k+\frac{\delta_k}{\lambda_k}\right)-\left(v_{k+1}+\frac{\delta_{k+1}}{\lambda_{k+1}}\right).$$

Summing from $k=N_s$ to $k=N$ and letting $N \to \infty$, we obtain that $\sum\limits_{k=1}^{\infty}\varGamma_k$, i.e. $\sum\limits_{k=1}^{\infty}\|x_k-y_k\|^2$ is convergent from Lemma 2.1.

In addition, it follows from (65) that

$$v_k+\frac{\delta_k}{\lambda_k} \le v_{N_s}+\frac{\delta_{N_s}}{\lambda_{N_s}}$$

which implies that for $k \ge N_s$

$$F\left(x_k\right) \le \xi. \tag{66}$$

Based on the nonexpansiveness property of the proximal operator [8], Δf is Lipschitz continuous and $\lambda_k \leq \lambda^*$ for k sufficiently large, we deduce to

$$\begin{aligned} \left\|p_{\lambda_k g}(x_k) - x_k\right\| &= \left\|p_{\lambda_k g}(x_k) - p_{\lambda_k g}(y_k)\right\| \\ &= \left\|\mathrm{prox}_{\lambda_k g}(x_k - \lambda_k \nabla f(x_k)) - \mathrm{prox}_{\lambda_k g}(y_k - \lambda_k \nabla f(y_k))\right\| \\ &\leq \left(1 + \lambda^* \cdot L_f\right) \|x_k - y_k\|. \end{aligned} \tag{67}$$

In addition, we discussion λ_k in two cases: For the case that $\lambda_k \leq \frac{1}{L_f}$, it follows from Lemma 5.3 and (10) that

$$\frac{\left\|G_{L_f g}(x_k)\right\|}{L_f} \leq \frac{\left\|G_{\frac{1}{\lambda_k} g}(x_k)\right\|}{1/\lambda_k} \frac{1}{L_f \cdot \lambda_k} \leq \frac{1}{L_f \cdot \lambda_{\min}} \left\|p_{\lambda_k g}(x_k) - x_k\right\|. \tag{68}$$

For the case that $\lambda_k > \frac{1}{L_f}$, we have

$$\frac{\left\|G_{L_f g}(x_k)\right\|}{L_f} < \frac{\left\|G_{\frac{1}{\lambda_k} g}(x_k)\right\|}{1/\lambda_k} = \left\|p_{\lambda_k g}(x_k) - x_k\right\|. \tag{69}$$

It follows from (68) and (69) that

$$\frac{\left\|G_{L_f g}(x_k)\right\|}{L_f} \leq \tau_1 \left\|p_{\lambda_k g}(x_k) - x_k\right\|, \tag{70}$$

where $\tau_1 = \max(\frac{1}{L_f \cdot \lambda_{\min}}, 1)$. Combining (67), (70) and Theorem 3.1 (b), we have

$$\lim_{k\to\infty} \frac{\left\|G_{L_f g}(x_k)\right\|}{L_f} = \lim_{k\to\infty} \left\|p_{\frac{1}{L_f} g}(x_k) - x_k\right\| = 0. \tag{71}$$

Following (66) and (71), we have $F(x_k) \leq \xi$ and $\left\|p_{\frac{1}{L_f} g}(x_k) - x_k\right\| < \varepsilon$ hold for k sufficient large. Then, combining (57), (70) and (67), there exist $\tau_2 > 0$ for k sufficient large,

$$dist\left(x_k, X^*\right) \leq \tau_2 \|x_k - y_k\|. \tag{72}$$

From (34) with $y := y_{k+1}$, we have

$$\begin{aligned} F(x_{k+1}) &\leq F(x) + \tfrac{\|x-y_{k+1}\|^2}{2\lambda_{k+1}} = F(x) + \tfrac{\|x-x_{k+1}+x_{k+1}-y_{k+1}\|^2}{2\lambda_{k+1}} \\ &\leq F(x) + \tfrac{1}{\lambda_{k+1}} \left(\|x - x_{k+1}\|^2 + \|x_{k+1} - y_{k+1}\|^2\right). \end{aligned} \tag{73}$$

Choose x to be an $x^*_{k+1} \in X^*$ so that $\left\|x^*_{k+1} - x_{k+1}\right\| = dist(x_{k+1}, X^*)$, then,

$$\begin{aligned} F(x_{k+1}) - F(x^*) &= F(x_{k+1}) - F\left(x^*_{k+1}\right) \leq \tfrac{1}{\lambda_{k+1}} \left(\left\|x^*_{k+1} - x_{k+1}\right\|^2 + \|x_{k+1} - y_{k+1}\|^2\right) \\ &= \tfrac{1}{\lambda_{k+1}} \left(dist^2(x_{k+1}, X^*) + \|x_{k+1} - y_{k+1}\|^2\right) \\ &\leq \tau_3 \|x_{k+1} - y_{k+1}\|^2, \end{aligned} \tag{74}$$

where $\tau_3 = \frac{1+(\tau_2)^2}{\lambda_{\min}}$ and the last inequality is from (72) and Lemma 2.1, i.e.,

$$\upsilon_{k+1} \leq \tau_3 \varGamma_{k+1} \tag{75}$$

hold.

It follows from (64) and (75) and Lemma 5.2 that $\left\{\nu_k + \alpha \frac{\delta_k}{\lambda_k}\right\}$ is $Q-$linearly convergent to zero. And $F(x_k)$ is $R-$linear convergent to $F(x^*)$, $\left\{\|x_{k+1} - x_k\|^2\right\}$ is $R-$linear convergent to zero.

With the $R-$linearly convergence of $\left\{\|x_{k+1}-x_k\|^2\right\}$, we obtain that there exist $0 < \bar{c} < 1$, and $M_1 > 0$, such that

$$\|x_k - x_{k-1}\| \leq M_1 \bar{c}^k.$$

Consequently, for any $m_2 > m_1 > 0$, we have

$$\|x_{m_2} - x_{m_1}\| \leq \sum_{k=m_1+1}^{m_2} \|x_k - x_{k-1}\| \leq M_1 \cdot \frac{\bar{c}^{m_1}}{1-\bar{c}}$$

showing that $\{x_k\}$ is a Cauchy sequence and hence convergent. Denoting its limit by $x^*$and passing to the limit as $m_2 \to \infty$ in the above relation, we see further that

$$\left\|x_{m_1} - x^*\right\| \leq M_1 \cdot \frac{\bar{c}^{m_1}}{1-\bar{c}}$$

that means that the sequence $\{x_k\}$ is $R-$linearly convergent to its limit. □

Remark 5.1. Under the error bound condition, Wen, Chen and Pong [29] proved that for FISTA equipped with the restart scheme and the constant stepsize $\frac{1}{L_f}$, the sequences $\{x_k - x^*\}$ and $\{F(x_k) - F(x^*)\}$ are $R-$linear convergent to zero. In Theorem 5.1, we show the similar results hold for FISTA and FISTA_CD with stepsize generated by Algorithm 3 based on the error bound condition and restart scheme. The proposed algorithm implementations are independent of L_f. In the proof of Theorem 5.1, the main contribution of Algorithm 3 is that it generates a stepsize sequence which is convergent and increases monotonically after finite iterations. We see that backtracking strategy in FISTA_BKTR does not have this property, hence, it is not clear whether FISTA_BKTR can obtain the linearly convergence.

6 Numerical Experiments

6.1. We conduct numerical experiments to demonstrate our algorithms' effectiveness by testing the following five algorithms:
— FISTA_backtracking
— FISTA_BKTR
— FISTA_CD_BKTR($a = 4$)
— FISTA_NMS
— MFISTA_NMS($a = 4$)

termination condition

The inequality $\|\psi_k\| \leq \varepsilon$ is often used to be the termination condition for all comparison algorithms, where $\psi_k = \nabla f(x_k) - \frac{1}{\lambda_k}(x_k - y_k + \lambda_k \nabla f(y_k)) \in \partial F(x_k)$. However, we notice that if F is flat, the distance between two iterates will be very far but the value of $\|\psi_k\|$ is close to 0; oterwise, conversely. Hence, we terminate the test algorithms when $\min(\|\psi_k\|, \|x_k - x_{k-1}\|) \leq \varepsilon$.

Test Function

The numerical experiments are conducted on the following two types of test functions: (1) The linear inverse problem; (2) the l_1-regularized logistic regression. It's obvious that the first problem is the case that f is a quadratic function, thus we need to restrict the parameter $\mu_1 < \mu_0 < 1$; for the latter that f is a non-quadratic function, $\mu_1 < \mu_0 < 1/2$. In the numerical experiment, we set $E_k = \frac{w_k}{k^{1.1}}$, $\forall k \geq 1$ be the control series for the new adaptive non-monotone stepsize strategy, parameter w_k same as the setting we introduced in Section 2; $\mu_0 = 0.99, \mu_1 = 0.95$ for the test function (1), $\mu_0 = 0.49, \mu_1 = 0.45$ for the test function (2); $\varepsilon = 1.e-5$. For the backtracking scheme, we set $\eta = 0.5$, $\lambda_k^0 = \frac{\lambda_{k-1}}{\eta}$.

6.1.1. the Linear Inverse problem.

The Linear Inverse problem is described as follows:

$$\min_x F(x) = \frac{1}{2}\|Ax - b\|^2 + \sigma\|x\|_1, \tag{76}$$

where the linear operator A and observation b is generated by the following scheme:

$$\begin{aligned}&\mathrm{A = randn(n, m);}\\&\mathrm{xstar = ones(m, 1);}\\&\mathrm{Set\ s : The\ number\ of\ non-zero\ elements\ of\ xstar}\\&\mathrm{I = randperm(m);\ xstar(I(1 : m - s)) = 0;}\\&\mathrm{b = A * xstar{+}0.1 * randn(n, 1);}\end{aligned}$$

In the numerical experiments, we take $n = 1000, m = 10000$.

Note that in this linear inverse problem, $\nabla f(x) = A^T (Ax - b)$, which is linear, hence, we can directly compute $\nabla f(y_k)$ by linear relationship between $\nabla f(x_{k-1})$ and $\nabla f(x_{k-2})$; since that $Ay_k - b$ can be computed by linear relationship between $Ax_{k-1} - b$ and $Ax_{k-2} - b$, so the computation of $f(y_k)$ is negligible. Through numerical experiments, we find that for FISTA_backtracking and FISTA_BKTR, the condition $F\left(p_{\lambda_k g}(y_k)\right) \le Q_{\lambda_k}\left(p_{\lambda_k g}(y_k), y_k\right)$ is difficult to distinguish if we set ε too small, which means that these two backtracking schemes are not suitable for applications with high precision requirements like Medical imaging. We consider the influence of such factors like sparsity $\left(\frac{s}{m}\right)$ and regularization parameter σ on the algorithm. The selection of regularization parameter is separately $\sigma = 1$ and $\sigma = 0.1$. *Iter* denotes the total number of iterations and *Mult* denotes the number of matrix-vector product for compute $Ax - b$ and *Time* denotes the CPU time.

From Table 1–3, we see that under the setting of different parameters and different sparsity, our algorithms FISTA_NMS and MFISTA_NMS hava significant improvment over FISTA_backtracking, and comparison with FISTA_BKTR and FISTA_CD_BKTR, we see that FISTA_BKTR is a little better than FISTA_NMS for the total number of iterations, but much more than FISTA_NMS for the number of matrix-vector product, the comparison with other two algorithms MFISTA_NMS and FISTA_CD_BKTR show similar results. In order to more intuitively show the effectiveness of our algorithms, we plot how $\|\psi_k\|$ and $F(x_k) - F(x^*)$ changes during time taken by these five algorithms, where F^* be the smallest $F(x_k)$ among all methods.

		Iter	Mult	Time
σ=1,s=80	FISTA_NMS	4586	**9174**	**34.8816**
	FISTA_BKTR	**4030**	12140	47.6178
	FISTA_backtracking	9527	20070	76.3822
	MFISTA_NMS	3093	**6188**	**23.5663**
	FISTA_CD_BKTR	**2822**	9481	37.6727

Table 1: Comparison of algorithms for solving (76) with n=800, m=8000, s=80, $\sigma = 1$

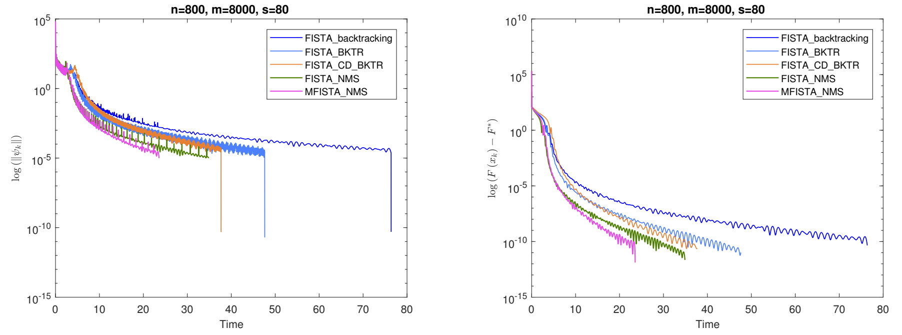

Fig. 1: Performance profile for the convergences of $\|\psi_k\|$ and $F(x_k) - F(x^*)$ with $\sigma = 1$.

		Iter	Mult	Time
σ=0.1,s=80	FISTA_NMS	8756	**17514**	**65.8372**
	FISTA_BKTR	**8293**	24880	99.5572
	FISTA_backtracking	30550	62116	234.7369
	MFISTA_NMS	**7329**	**14660**	**55.2078**
	FISTA_CD_BKTR	7640	22921	88.9125

Table 2: Comparison of algorithms for solving (76) with n=800, m=8000, s=80, $\sigma = 0.1$

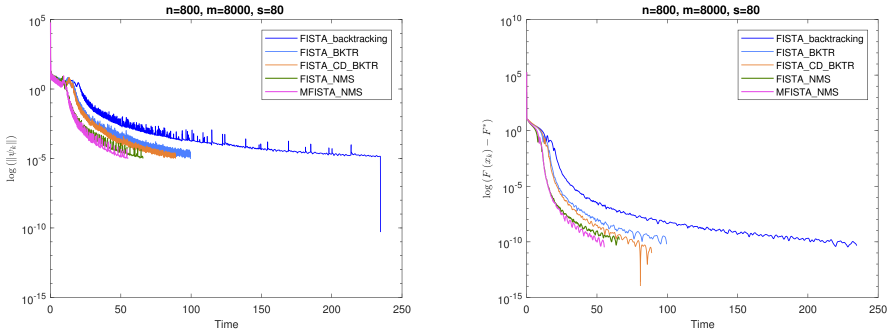

Fig. 2: Performance profile for the convergences of $\|\psi_k\|$ and $F(x_k) - F(x^*)$ with $\sigma = 0.1$.

		Iter	Mult	Time
σ=1,s=200	FISTA_NMS	29838	**59678**	**234.7507**
	FISTA_BKTR	**23619**	70915	279.8414
	FISTA_backtracking	66079	133174	525.9265
	MFISTA_NMS	18962	**37926**	**146.4571**
	FISTA_CD_BKTR	**13654**	41977	169.4404

Table 3: Comparison of algorithms for solving (76) with n=800, m=8000, s=200, $\sigma = 1$

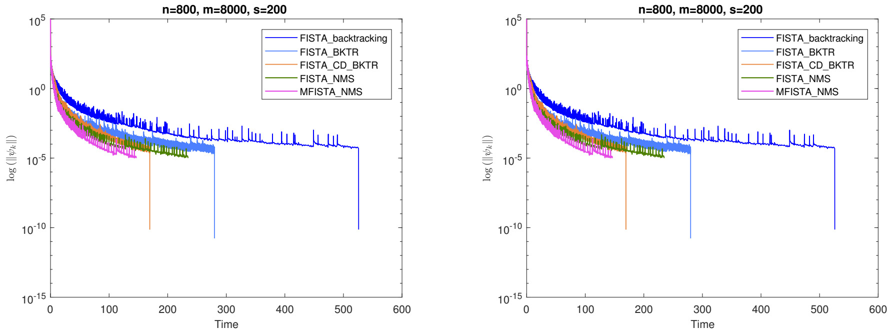

Fig. 3: Performance profile for the convergences of $\|\psi_k\|$ and $F(x_k) - F(x^*)$ with $\sigma = 1$.

From Fig.1–3, we can see that even if regularization parameter selection and sparsity are different, FISTA_NMS has a significant improvement over the FISTA_BKTR and FISTA_backtracking for the given test problem. Moreover, we can see that MFISTA_NMS is more efficient than FISTA_CD_BKTR, which means that our stepsize strategy is also effective for the modified algorithm FISTA_CD. Numerical experiments show that the new adaptive nonmonotone stepsize strategy is very useful for improving algorithm performances and our algorithms are very suitable for practical application problems such as sparse signal processing.

Since that FISTA_BKTR successfully improves the FISTA in practice, in the following computational experiments, we just compare the algorithms: FISTA_NMS, MFISTA_NMS, FISTA_BKTR and FISTA_CD_BKTR.

6.1.2. Sparse Logistic Regression

Consider the question

$$\min_{x} F\left(x\right) := \frac{1}{n}\sum_{i=1}^{n} \log\left(1+\exp\left(-l_i\left\langle h_i, x\right\rangle\right)\right) + \sigma\|x\|_1, \tag{77}$$

where $x \in R^m, h_i \in R^n, l_i \in \{-1, 1\}, i = 1, \cdots, n$, and $\sigma = 1.e-2$. The problem sparse logistic regression is a popular problem in machine learning applications, where $f(x) = \frac{1}{n}\sum\limits_{i=1}^{n} \log\left(1+\exp\left(-l_i\left\langle h_i, x\right\rangle\right)\right)$ is non-linear. Define $K_{ij} = -l_i h_{ij}$, and set $\tilde{f}\left(y\right) = \sum\limits_{i=1}^{m} \log\left(1+\exp\left(y_i\right)\right)$. Then $f\left(x\right) = \tilde{f}\left(Kx\right)$, and $L_f = \frac{4}{n}\left\|K^T K\right\|$. Initial point $\mathrm{x}_0 = \mathrm{zeros}(\mathrm{m}, 1)$. We take three datasets 'heart_test', 'sonar_test' and 'mushroom' from LIBSVM [9]. We report the number of iterations (*Iter*), calculation of function value (*Fval*), calculation of gradient value (*Gval*) and CPU time (*Time*).

	Iter	Fval	Gval	Time
FISTA_NMS	81392	**81392**	**162784**	**6.6532**
FISTA_BKTR	**75583**	199829	175497	10.8337
MFISTA_NMS	25864	**25864**	**51728**	**2.1469**
FISTA_CD_BKTR	**20412**	81645	61234	4.0928

Table 4: Comparison of algorithms for solving "heart_test".

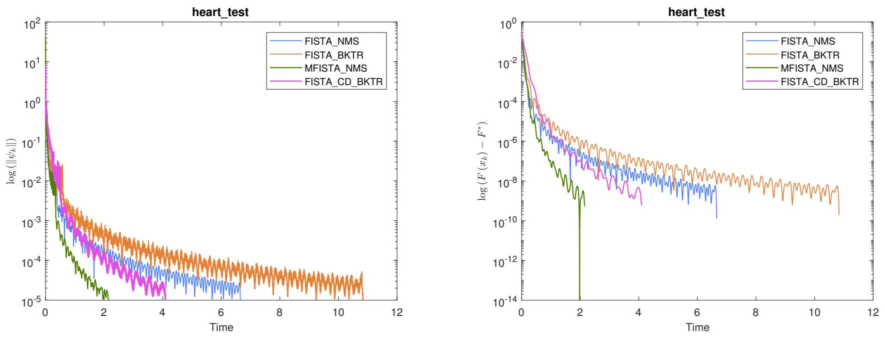

Fig. 4: Performance profile for solving "heart_test".

	Iter	Fval	Gval	Time
FISTA_NMS	1044	**1044**	**2088**	**0.144**
FISTA_BKTR	**916**	2420	2126	0.1806
MFISTA_NMS	719	**719**	**1438**	**0.0975**
FISTA_CD_BKTR	**530**	2114	1587	0.151

Table 5: Comparison of algorithms for solving “sonar_test”.

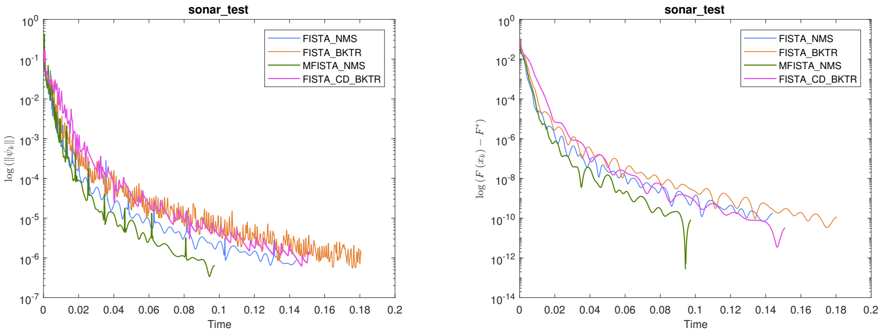

Fig. 5: Performance profile for solving “sonar_test”.

	Iter	Fval	Gval	Time
FISTA_NMS	116	**116**	**232**	**0.0585**
FISTA_BKTR	**107**	249	231	0.0703
MFISTA_NMS	100	**100**	**200**	**0.0379**
FISTA_CD_BKTR	**93**	339	262	0.0656

Table 6: Comparison of algorithms for solving “mushroom”.

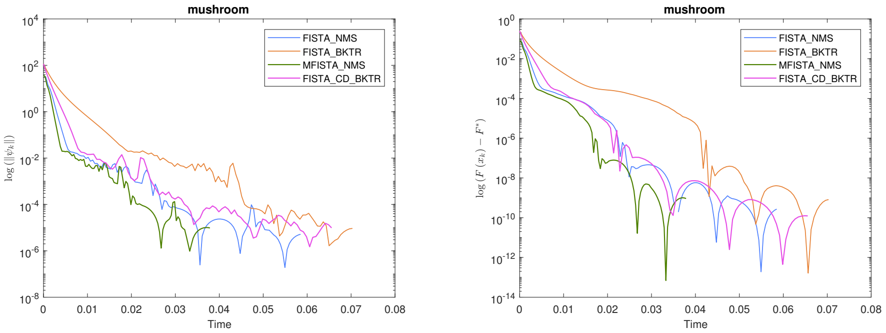

Fig. 6: Performance profile for solving “mushroom”.

The algorithms for solving the sparse logistic regression problem obtain similar results, i.e., though the number of iterations of algorithms with NMS is slightly worse than algorithms with BKTR, we can see that the algorithms with NMS are obviously better from the calculation times of function and gradient value and CPU time. Hence, FISTA_NMS outperforms the FISTA_BKTR, and meanwhile, MFISTA_NMS is more efficient than the FISTA_CD_BKTR. Observe that sometimes FISTA_NMS is faster than FISTA_CD_BKTR for some test problem, like the sparse logistic regression with "sonar_test" and "mushroom" datasets.

6.2. The main goal of our experiments is to test that our algorithms combining with the *Restart* scheme are still effective. The test functions and the related parameter settings are same as Subsection 6.1.

First, we compare the following four algorithms: FISTA_NMS; FISTA_NMS_restart; MFISTA_NMS and MFISTA_NMS_restart. We can see that using the restart strategy, both of our algorithms' performances can be greatly improved, which shows from Table 7 that *Iter*, *Mult* and *Time* for solving the linear inverse problem be greatly reduced.

		Iter	Mult	Time
σ=1,s=80	FISTA_NMS	4292	8586	32.4994
	FISTA_NMS_restart	**693**	**1388**	**5.2372**
	MFISTA_NMS	3331	6664	25.2049
	MFISTA_NMS_restart	**803**	**1608**	**6.0839**
σ=0.1,s=80	FISTA_NMS	8769	17540	64.7331
	FISTA_NMS_restart	**2866**	**5734**	**21.3431**
	MFISTA_NMS	7260	14522	53.6101
	MFISTA_NMS_restart	**2970**	**5942**	**21.9513**
σ=1,s=200	FISTA_NMS	28423	56848	219.7712
	FISTA_NMS_restart	**19410**	**38822**	**155.6988**
	MFISTA_NMS	26345	52692	200.5118
	MFISTA_NMS_restart	**21844**	**43690**	**163.4164**

Table 7: Comparison of algorithms with restart scheme and without restart scheme for solving (76) with n=800, m=8000.

In the following, we compare the following four algorithms: FISTA_BKTR_restart; FISTA_CD_BKTR_restart; FISTA_NMS_restart and MFISTA_NMS_restart. We present numerical results to elaborate that: after incorporating restart strategy into all the comparison algorithms, our algorithms are still superior to the other two comparison algorithms, which shows the stability of our algorithms. From Fig.7–Fig.9, we show the comparison

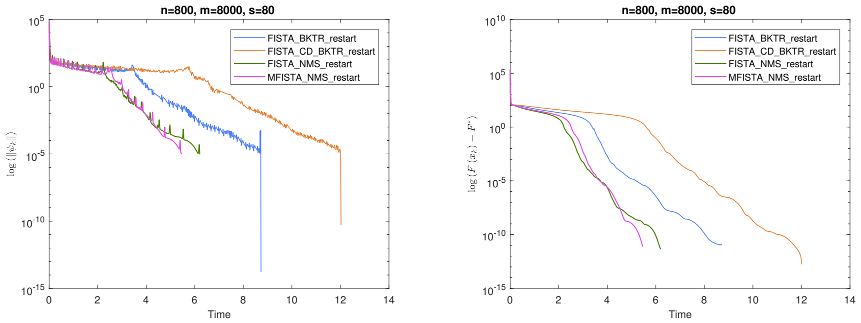

Fig. 7: Performance profile for the convergences of $\|\psi_k\|$ and $F(x_k) - F(x^*)$ with $\sigma = 1$.

results for solving the linear inverse problem with different regularization parameter values and sparsity:

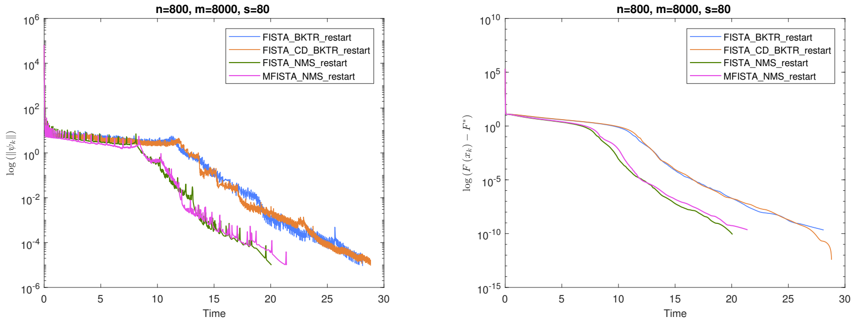

Fig. 8: Performance profile for the convergences of $\|\psi_k\|$ and $F(x_k) - F(x^*)$ with $\sigma = 0.1$.

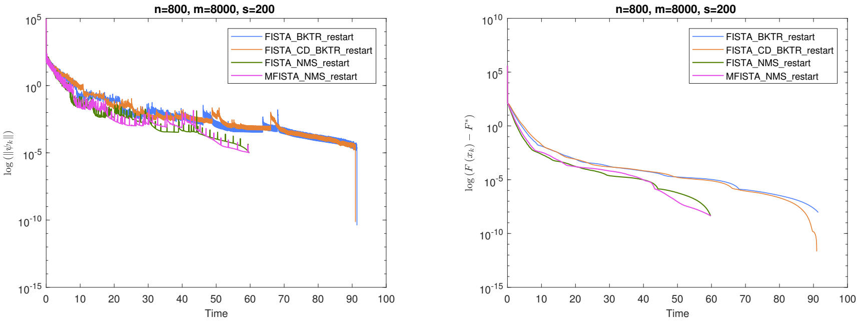

Fig. 9: Performance profile for the convergences of $\|\psi_k\|$ and $F(x_k) - F(x^*)$ with $\sigma = 1$.

From Fig.10–Fig.12, we show the comparison results for solving the sparse logistic regression problem:

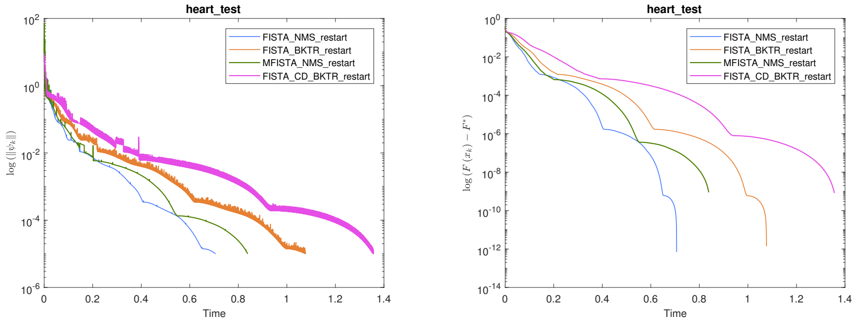

Fig. 10: Performance profile for solving “heart_test.”

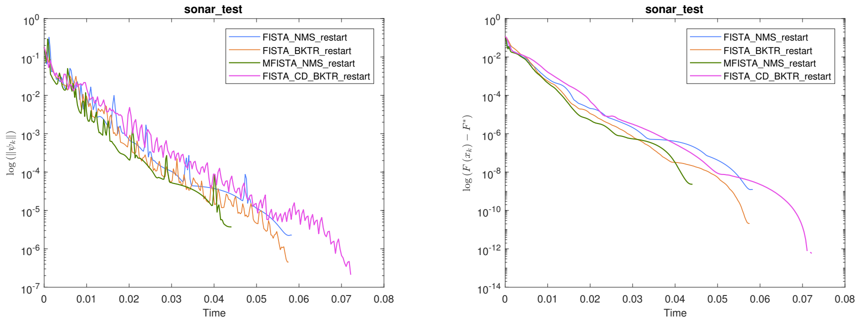

Fig. 11: Performance profile for solving "sonar_test."

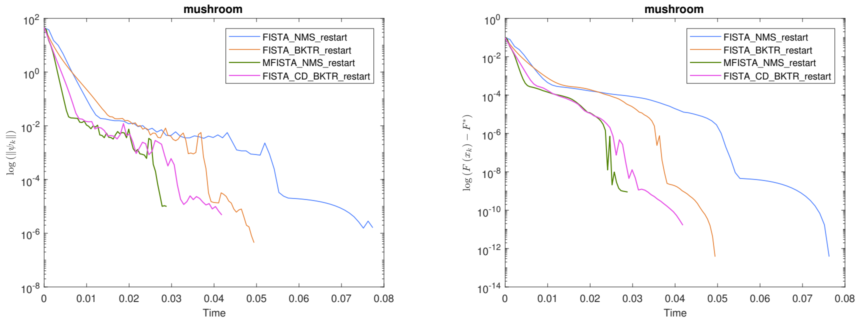

Fig. 12: Performance profile for solving "mushroom."

7 Conclusion

In this paper, we introduce a new adaptive nonmonotone stepsize strategy (NMS), which does not execute line search and is independent of the Lipschitz constant. Based on NMS, we propose FISTA_NMS that has $O\left(\frac{1}{k^2}\right)$ convergence rate of the objective function value, which is similar with FISTA. We construct the convergence of iterates generated by MFISTA_NMS based on the new adaptive nonmonotone stepsize without dependent on the Lipschitz constant. Also, the convergence rate of objective function value shares $o\left(\frac{1}{k^2}\right)$. Further, our algorithms FISTA_NMS and MFISTA_NMS acheive similar convergence rate in the norm of subdifferential of objective function. Under error bound condition, we prove that FISTA_NMS and MFISTA_NMS have improved convergence results, i.e., for FISTA_NMS, convergence rates of function value and iterates can be achieved to $o\left(\frac{1}{k^6}\right)$ and $O\left(\frac{1}{k^2}\right)$; for MFISTA_NMS, that are $o\left(\frac{1}{k^{2(a+1)}}\right)$ and $o\left(\frac{1}{k^a}\right)$. In addition, we improve our algorithms and give the proof of the linear convergence of function value and iterates by combining our algorithms with the restart strategy. Note that FISTA and FISTA_CD with backtracking schemes can not achieve the same results, which means that NMS has theoretical advantages. We demonstrate the performance of our schemes on some numerical examples to show that our stepsize strategy outperforms the backtracking.

Acknowledgements The work was supported by the National Natural Science Foundation of China (No.11901561), the Natural Science Foundation of Guangxi (No.2018GXNSFBA281180) and the Postdoctoral Fund Project of China (Grant No.2019M660833).

References

1. Attouch, H., Peypouquet, J.: The rate of convergence of Nesterov's accelerated forwardbackward method is actually faster than $\frac{1}{k^2}$. SIAM J. Optim. 26, 1824–1834 (2016)
2. Attouch, H., Cabot, A.: Convergence rates of inertial forward-backward algorithms. SIAM J. Optim. 28, 849–874 (2018)
3. Beck, A., Teboulle, M.: A linearly convergent dual-based gradient projection algorithm for quadratically constrained convex minimization. Math. Oper. Res. 31, 398–417 (2006)
4. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM J. Imaging Sci. 2, 183–202 (2009)
5. Becker, S. R., Cands, E.J., Grant, M. C.: Templates for convex cone problems with applications to sparse signal recovery. Math. Prog. Comp. 3, 165–218 (2011)
6. Bello, C., Jose, Y., Nghia, T. T. A.: On the convergence of the forward-backward splitting method with linesearches. Optim. Method Softw. 31, 1209–1238 (2016)
7. Chambolle, A.: An algorithm for total variation minimization and applications. J. Math. Imaging Vis. 20,. 89–97 (2004)
8. Combettes, P. L., Wajs, V. R.: Signal recovery by proximal forward-backward splitting. Multiscale Model. Simul. 4, 1168–1200 (2005).
9. Chang, C. C., Lin, C. J.: LIBSVM: a library for support vector machines. ACM. Trans. Intell. Syst. Technol. 2, 1–27 (2011)
10. Combettes, P.L., Pesquet, J.C.: Proximal splitting methods in signal processing. In: Fixed-Point Algorithms for Inverse Problems in Science and Engineering. Springer, New York (2011)
11. Chambolle, A., Dossal, C.: On the convergence of the iterates of the "fast iterative shrinkage-thresholding algorithm". J. Optim. Theory Appl. 166, 968–982 (2015).
12. Donoho, D. L.: Compressed sensing. IEEE Trans. inf. Theory. 52, 1289–1306 (2006)
13. Lions, P. L., Mercier, B.: Splitting algorithms for the sum of two nonlinear operators. SIAM J. Numer. Anal. 16, 964–979 (1979)
14. Luo, Z. Q., Tseng, P.: Error bound and the convergence analysis of matrix splitting algorithms for the affine variational inequality problem. SIAM J. Optim. 2, 43–54 (1992)
15. Luo, Z. Q.: New error bounds and their applications to convergence analysis of iterative algorithms. Math. Program. 88, 341–355 (2000)
16. Lorenz, D.A., Pock, T.: An inertial forward-backward algorithm for monotone inclusions. J. Math. Imaging Vis. 51, 311–325 (2015)
17. Liang, J., Fadili, J., Peyr, G.: Convergence rates with inexact non-expansive operators. Math. Program. 159, 403–434 (2016)
18. H. W. Liu, T. Wang and Z. X. Liu, Convergence rate of inertial forward-backward algorithms based on the local error bound condition. http://arxiv.org/pdf/2007.07432
19. Molinari, C., Liang, J., Fadili, J.: Convergence rates of forward-douglas-rachford splitting Method. J. Optim. Theory Appl. 182, 606–639 (2019)
20. Nesterov, Y.: A method for solving the convex programming problem with convergence rate $O\left(\frac{1}{k^2}\right)$. Dokl. Akad. Nauk SSSR. 269, 543–547 (1983)
21. Nesterov, Y.: Gradient methods for minimizing composite objective function. Math. Program. 140, 125–161 (2012)
22. Nesterov, Y.: Gradient methods for minimizing composite functions. Math. Program. 140, 125–161 (2013)
23. O'Donoghue, B., Cands, E.: Adaptive restart for accelerated gradient schemes. Found Comput Math. 15, 715–732 (2015)
24. Sra, S., Nowozin, S., Wright, S.J.: Optimization for machine learning. MIT Press, Cambridge, Massachusetts (2012)
25. W. Su, S. Boyd and E. J. Candes, A differential equation for modeling Nesterov's accelerated gradient method: Theory and insights, J. Mach. Learn. Res, 17 (2016), pp. 1–43.
26. K. Scheinberg, D. Goldfarb and X. Bai, Fast First-Order Methods for Composite Convex Optimization with Backtracking, Found. Comput. Math, 14 (2014), pp. 389–417.
27. Tseng, P.: Approximation accuracy, gradient methods, and error bound for structured convex optimization. Math. Program. 125, 263–295 (2010)
28. Tao, S., Boley, D., Zhang, S.: Local linear convergence of ISTA and FISTA on the LASSO problem. SIAM J. Optim. 26, 313–336 (2016)
29. Wen, B., Chen, X. J., Pong, T. K.: Linear convergence of proximal gradient algorithm with extrapolation for a class of nonconvex nonsmooth minimization problems. SIAM J. Optim. 27, 124–145 (2017)