Improving "Fast Iterative Shrinkage-Thresholding Algorithm": Faster, Smarter and Greedier

Jingwei Liang[*] Tao Luo[†] Carola-Bibiane Schönlieb[‡]

Abstract. The "fast iterative shrinkage-thresholding algorithm", a.k.a. FISTA, is one of the most well-known first-order optimization scheme in the literature, as it achieves the worst-case $O(1/k^2)$ optimal convergence rate in terms of objective function value. However, despite such an optimal theoretical convergence rate, in practice the (local) oscillatory behavior of FISTA often damps its efficiency. Over the past years, various efforts are made in the literature to improve the practical performance of FISTA, such as monotone FISTA, restarting FISTA and backtracking strategies. In this paper, we propose a simple yet effective modification to the original FISTA scheme which has two advantages: it allows us to 1) prove the convergence of generated sequence; 2) design a so-called "lazy-start" strategy which can be up to an order faster than the original scheme. Moreover, we propose novel adaptive and greedy strategies which probe the limit of the algorithm. The advantages of the proposed schemes are tested through problems arising from inverse problem, machine learning and signal/image processing.

Key words. FISTA, inertial Forward–Backward, lazy-start strategy, adaptive and greedy acceleration

AMS subject classifications. 65K05, 65K10, 90C25, 90C31.

1 Introduction

The acceleration of first-order optimization methods is an active research topic of non-smooth optimization. Over the past decades, various acceleration techniques are proposed in the literature. Among them, one most widely used is called "inertial technique" which dates back to [26] where Polyak proposed the so called "heavy-ball method" which dramatically speeds up the practical performance of gradient descent. In a similar spirit, in [22] Nesterov proposed another accelerated scheme which improves the $O(1/k)$ objective function convergence rate of gradient descent to $O(1/k^2)$. The extension of [22] to the non-smooth case was due to [6] where Beck and Teboulle proposed the FISTA scheme which is the main focus of this paper.

In this paper, we are interested in the following structured non-smooth optimization problem, which is the sum of two convex functionals,

$$\min_{x\in\mathcal{H}} \Phi(x) \overset{\text{def}}{=} F(x) + R(x), \tag{$\mathscr{P}$}$$

where $\mathcal{H}$ is a real Hilbert space. The following assumptions are assumed throughout the paper

- **(H.1)** $R : \mathcal{H} \to]-\infty, +\infty]$ is proper, convex and lower semi-continuous (lsc);
- **(H.2)** $F : \mathcal{H} \to]-\infty, +\infty[$ is convex and differentiable, with gradient ∇F being L-Lipschitz continuous for some $L > 0$;
- **(H.3)** The set of minimizers is non-empty, *i.e.* $\mathrm{Argmin}(\Phi) \neq \emptyset$.

Problem ($\mathscr{P}$) covers many problems arising from inverse problems, signal/image processing, statistics and machine learning, to name few. We refer to Section 7 the numerical experiment section for concrete examples.

[*] School of Mathematical Sciences, Queen Mary University of London, London UK. E-mail: jl993@cam.ac.uk.

[†] School of Mathematical Sciences, Shanghai Jiao Tong University, Shanghai China. E-mail: luotao41@sjtu.edu.cn.

[‡] DAMTP, University of Cambridge, Cambridge UK. E-mail: cbs31@cam.ac.uk.

1.1 Forward–Backward-type splitting schemes

In the literature, one widely used algorithm for solving ($\mathscr{P}$) is Forward–Backward splitting (FBS) method [17], which is also known as *proximal gradient descent*.

Forward–Backward splitting With initial point $x_0 \in \mathscr{H}$ chosen arbitrarily, the standard FBS iteration without relaxation reads as

$$x_{k+1} \overset{\text{def}}{=} \text{prox}_{\gamma_k R}\big(x_k - \gamma_k \nabla F(x_k)\big), \ \gamma_k \in]0, 2/L], \tag{1.1}$$

where γ_k is the step-size, and $\text{prox}_{\gamma R}$ is called the *proximity operator* of R defined by

$$\text{prox}_{\gamma R}(\cdot) \overset{\text{def}}{=} \text{argmin}_{x \in \mathscr{H}} \gamma R(x) + \tfrac{1}{2}\|x - \cdot\|^2. \tag{1.2}$$

Similar to gradient descent, FBS is a descent method, that is the objective function value $\Phi(x_k)$ is non-increasing under properly chosen step-size γ_k. The convergence properties of FBS are well established in the literature, in terms of both sequence and objective function value:

- The convergence of the generated sequence $\{x_k\}_{k \in \mathbb{N}}$ and the objective function value $\Phi(x_k)$ are guaranteed as long as γ_k is chosen such that $0 < \underline{\gamma} \leq \gamma_k \leq \bar{\gamma} < \frac{2}{L}$ [12].
- Convergence rate: we have $\Phi(x_k) - \min_{x \in \mathscr{H}} \Phi(x) = o(1/k)$ for the objective function value [19] and $\|x_k - x_{k-1}\| = o(1/\sqrt{k})$ for the sequence $\{x_k\}_{k \in \mathbb{N}}$ [15]. Moreover, linear convergence rate can be obtained under for instance strong convexity.

Over the years, numerous variants of FBS have been proposed under different purposes, below we particularly focus on its inertial accelerated variants.

Inertial Forward–Backward The first inertial Forward–Backward was proposed by Moudafi and Oliny in [20], under the setting of finding zeros of monotone inclusion problems. Specifying the algorithm to the case of solving ($\mathscr{P}$), we obtain the following iteration:

$$\begin{aligned} y_k &= x_k + a_k(x_k - x_{k-1}), \\ x_{k+1} &= \text{prox}_{\gamma_k R}\big(y_k - \gamma_k \nabla F(x_k)\big), \gamma_k \in]0, 2/L[, \end{aligned} \tag{1.3}$$

where a_k is the *inertial parameter* which controls the momentum $x_k - x_{k-1}$. The above scheme recovers the heavy-ball method when $R = 0$ [27], and becomes the scheme of [18] if we replace $\nabla F(x_k)$ with $\nabla F(y_k)$. We refer to [16] for a more general discussion of inertial Forward–Backward splitting schemes.

The convergence of (1.3) can be guaranteed under proper choices of γ_k and a_k. Under the same step-size choice, (1.3) could be significantly faster than FBS in practice. However, except for special cases (*e.g.* quadratic problem as in [27]), in general there is no convergence rate established for (1.3).

The original FISTA By the form of iteration, FISTA is a particular example of the class of inertial FBS algorithms. What differentiates FISTA from others is the restriction on step-size γ_k and special rule for updating a_k. Moreover, FISTA schemes have convergence rate guarantee on the objective function value, which is the consequence of a_k updating rule. The original FISTA scheme of [6] is described below in Algorithm 1.

Algorithm 1: The original FISTA scheme (FISTA-BT)

```
Initial: $t_0 = 1$, $\gamma = 1/L$ and $x_0 \in \mathscr{H}, x_{-1} = x_0$, $k = 1$.
repeat
    $t_k = \frac{1+\sqrt{1+4t_{k-1}^2}}{2}, \ a_k = \frac{t_{k-1}-1}{t_k},$
    $y_k = x_k + a_k(x_k - x_{k-1}),$                                  (1.4)
    $x_{k+1} = \text{prox}_{\gamma R}\big(y_k - \gamma \nabla F(y_k)\big).$
    $k = k + 1$;
until convergence;
```

As described, FISTA first computes t_k and then updates a_k with t_k and t_{k-1}. Due to the choices of parameters, FISTA achieves $O(1/k^2)$ convergence rate for $\Phi(x_k) - \min_{x\in\mathscr{H}} \Phi(x)$ which is optimal [21]. For the rest of the paper, to distinguish the original FISTA from the one in [10] and the proposed modified FISTA scheme, we shall use "FISTA-BT" to refer Algorithm 1.

A sequence-convergent FISTA Although achieving optimal convergence rate for objective function value, the convergence of the sequence $\{x_k\}_{k\in\mathbb{N}}$ generated by Algorithm 1 was initially an open problem. This question was answered in [10], where Chambolle and Dossal proved the convergence of $\{x_k\}_{k\in\mathbb{N}}$ by considering the following rule to update t_k: let $d > 2$ and

$$t_k = \frac{k+d}{d}, \ a_k = \frac{t_{k-1}-1}{t_k} = \frac{k-1}{k+d}. \tag{1.5}$$

Such a rule maintains the $O(1/k^2)$ objective convergence rate, and also allows the authors to prove the convergence of $\{x_k\}_{k\in\mathbb{N}}$. Later on in [3], (1.5) was studied under the continuous time dynamical system setting, and the convergence rate of objective function is proved to be $o(1/k^2)$ [2]. For the rest of the paper, we shall use "FISTA-CD" to refer to (1.5).

1.2 Problems

Although theoretically FISTA-BT achieves the optimal $O(1/k^2)$ convergence rate, in practice it could be even slower than the non-accelerated Forward–Backward splitting scheme, which is mainly caused by the oscillatory behavior of the scheme [16]. In the literature, several modifications of FISTA-BT are proposed to deal with such oscillation, such as the monotone FISTA [5] and restarting FISTA [24]. Other work includes FISTA-CD [10] for the convergence of iterates, and a backtracking strategy for adaptive Lipschitz constant estimation [8]. Despite these works, there are still important questions to answer:

- Although [10] proves the convergence of the iterates $\{x_k\}_{k\in\mathbb{N}}$ under t_k updating rule (1.5), the convergence of $\{x_k\}_{k\in\mathbb{N}}$ for the original FISTA-BT remains unclear.
- The practical performance of FISTA-CD is almost identical to FISTA-BT if d of (1.5) is chosen close to 2. However, when relatively large values of d are chosen, significant practical acceleration can be obtained. For instance, it is reported in [16] that for $d = 50$ the resulted performance can be several times faster than $d = 2$. However, there is no proper theoretical justifications on how to choose the value of d in practice.
- When the problem ($\mathscr{P}$) is strongly convex, there exists an optimal choice for a_k [23]. However, in practice, very often the problem is only locally strongly convex with unknown strong convexity, and estimating the strong convexity could be time consuming. This leads to the question of whether there is a low-complexity approach to estimate strong convexity, or do we really need a tight estimation of it?
- Restarting FISTA successfully suppresses the oscillatory behavior of FISTA schemes, hence achieving much faster practical performance. Can we further improve this scheme?

1.3 Contributions

The above questions are the main motivations of this paper, and our contributions are summarized below.

A sequence-convergent FISTA scheme By studying the t_k updating rule (1.4) of FISTA-BT and its difference with (1.5), we propose a modified FISTA scheme which applies the following rule,

$$t_k = \frac{p+\sqrt{q+rt_{k-1}^2}}{2}, \ a_k = \frac{t_{k-1}-1}{t_k}, \tag{1.6}$$

where $p, q \in]0,1]$ and $r \in]0,4]$, see also Algorithm 2. Such a modification has two advantages when $r = 4$,

- It maintains the $O(1/k^2)$ (actually $o(1/k^2)$) convergence rate of the original FISTA-BT (Theorem 3.3);
- It allows us to prove the convergence of the iterates $\{x_k\}_{k\in\mathbb{N}}$ (Theorem 3.5);

It also allows us to show that the original FISTA-BT is also optimal in terms of the constant which appears in the $O(1/k^2)$ rate, see (3.7) in Theorem 3.3.

Lazy-start strategy For the proposed scheme and FISTA-CD, owing to the free parameters in computing t_k, we propose in Section 4 a so-called "lazy-start" strategy for practical acceleration. The idea of such strategy is to slow down the speed of a_k approaching 1, which can lead to a faster practical performance. For certain problems, such a strategy can be an order faster than the original schemes, see Section 7 for illustration. For least squares problems, we show that theoretically there exists optimal choices for a_k update which only depends on the stopping criteria.

Adaptive and greedy acceleration Although the lazy-start strategy can significantly speed up the performance of FISTA, it still suffers the oscillatory behavior since the inertial parameter a_k eventually converges to 1. By combining with the restarting technique of [24], in Section 5 we propose two different acceleration strategies: restarting adaptation to (local) strong convexity and greedy scheme.

The oscillatory behavior of FISTA schemes is often related to strong convexity. When the problem is strongly convex, there exists an optimal choice $a^\star < 1$ for a_k [23], Moreover, under such $a^\star$ the iteration will no longer oscillate. Many problems in practice are only locally strongly convex however, estimating strong convexity in general is time consuming. Therefore in Section 5, we propose an adaptive scheme (Algorithm 4) which combines the restarting technique [24] and parameter update rule (1.6). Such an adaptive scheme avoids the direct estimation of strong convexity and achieve state-of-the-art performance.

We also investigate the mechanism of oscillation and the restarting technique, and propose a greedy scheme (see Algorithm 5) which uses aggressive inertial parameter (*e.g.* $a_k \geq 1$) and step-size (*e.g.* $\gamma \geq 1/L$), hence probing the limit of the restarting technique. Doing so, the greedy scheme can achieve a faster practical performance than the restarting FISTA of [24].

Nesterov's accelerated schemes Given the close relation between FISTA and the Nesterov's accelerated schemes [23], we also extend the above results, particularly the modified FISTA to Nesterov's schemes. Such an extension can also significantly improve the performance when compared to the original schemes.

1.4 Paper organization

The rest of the paper is organized as follows. Some notation and preliminary results are collected in Section 2. The proposed sequence-convergent FISTA scheme is presented in Section 3. The lazy-start strategy and the adaptive/greedy acceleration schemes are presented in Section 4 and Section 5 respectively. In Section 6, we extend the results to Nesterov's accelerated schemes. Numerical experiments are presented in Section 7.

2 Preliminaries

Throughout the paper, $\mathcal{H}$ is a real Hilbert space equipped with scalar product $\langle \cdot, \cdot \rangle$ and norm $\|\cdot\|$. Id denotes the identity operator on $\mathcal{H}$. $\mathbb{N}$ is the set of non-negative integers and $k \in \mathbb{N}$ is the index, $x^\star \in \mathrm{Argmin}(\Phi)$ denotes a global minimizer of ($\mathscr{P}$).

The sub-differential of a proper convex and lower semi-continuous function $R : \mathcal{H} \to]-\infty, +\infty]$ is a set-valued mapping defined by

$$\partial R : \mathcal{H} \rightrightarrows \mathcal{H}, x \mapsto \big\{ g \in \mathcal{H} \,|\, R(x') \geq R(x) + \langle g, x' - x \rangle, \ \forall x' \in \mathcal{H} \big\}. \tag{2.1}$$

Definition 2.1 (Monotone operator). A set-valued mapping $A : \mathcal{H} \rightrightarrows \mathcal{H}$ is said to be monotone if,

$$\langle x_1 - x_2, v_1 - v_2 \rangle \geq 0, \quad \forall v_1 \in A(x_1) \text{ and } v_2 \in A(x_2). \tag{2.2}$$

It is maximal monotone if the graph of A can not be contained in the graph of any other monotone operators.

It is well-known that for proper, convex and lower semi-continuous function $R : \mathcal{H} \to]-\infty, +\infty]$, its sub-differential is maximal monotone [28], and that $\mathrm{prox}_R = (\mathrm{Id} + \partial R)^{-1}$.

Definition 2.2 (Cocoercive operator). Let $\beta \in]0, +\infty[$ and $B : \mathcal{H} \to \mathcal{H}$, then B is β-cocoercive if

$$\langle B(x_1) - B(x_2), x_1 - x_2 \rangle \geq \beta \|B(x_1) - B(x_2)\|^2, \ \forall x_1, x_2 \in \mathcal{H}. \tag{2.3}$$

The L-Lipschitz continuous gradient ∇F of a convex continuously differentiable function F is $\frac{1}{L}$-cocoercive [4].

Lemma 2.3 (Descent lemma [7]). *Suppose that $F : \mathcal{H} \to \mathbb{R}$ is convex, continuously differentiable and ∇F is L-Lipschitz continuous. Then, given any $x, y \in \mathcal{H}$,*

$$F(x) \le F(y) + \langle \nabla F(y), x - y \rangle + \tfrac{L}{2}\|x - y\|^2.$$

Given any $x, y \in \mathcal{H}$, define the energy function $E_\gamma(x,y)$ by

$$E_\gamma(x,y) \overset{\text{def}}{=} R(x) + F(y) + \langle x - y, \nabla F(y) \rangle + \tfrac{1}{2\gamma}\|x - y\|^2.$$

It is obvious that $E_\gamma(x,y)$ is strongly convex with respect to x, hence denote the unique minimizer as

$$\begin{aligned} e_\gamma(y) \overset{\text{def}}{=} \operatorname{argmin}\big\{ E_\gamma(x,y) : x \in \mathbb{R}^n \big\} &= \operatorname{argmin}_x \big\{ \gamma R(x) + \tfrac{1}{2}\|x - (y - \gamma \nabla F(y))\|^2 \big\} \\ &= \operatorname{prox}_{\gamma R}\big(y - \gamma \nabla F(y)\big). \end{aligned} \tag{2.4}$$

The optimality condition of $e_\gamma(y)$ is described below.

Lemma 2.4 (Optimality condition of $e_\gamma(y)$). *Given $y \in \mathcal{H}$, let $y^+ = e_\gamma(y)$, then*

$$0 \in \gamma \partial R(y^+) + \big(y^+ - (y - \gamma \nabla F(y))\big) = \gamma \partial R(y^+) + (y^+ - y) + \gamma \nabla F(y).$$

We have the following basic lemmas from [6].

Lemma 2.5 ([6, Lemma 2.3]). *Let $y \in \mathcal{H}$ and $\gamma \in]0, 2/L[$ such that*

$$\Phi(e_\gamma(y)) \le E_\gamma(e_\gamma(y), y),$$

then for any $x \in \mathcal{H}$, we have $\Phi(x) - \Phi(e_\gamma(y)) \ge \frac{1}{2\gamma}\|e_\gamma(y) - y\|^2 + \frac{1}{\gamma}\langle y - x, e_\gamma(y) - y \rangle$.

Lemma 2.6 ([10, Lemma 3.1]). *Given $y \in \mathcal{H}$ and $\gamma \in]0, 1/L]$, let $y^+ = e_\gamma(y)$, then for any $x \in \mathcal{H}$, we have*

$$\Phi(y^+) + \tfrac{1}{2\gamma}\|y^+ - x\|^2 \le \Phi(y) + \tfrac{1}{2\gamma}\|y - x\|^2.$$

3 A sequence-convergent FISTA scheme

As we mentioned in the introduction, the main problems of the current FISTA schemes are caused by the behavior of a_k, that a_k converges to 1 too fast. As a result, we need some proper way to control this speed. For FISTA-CD, this can be achieved by choosing a relatively large value of d, while for FISTA-BT there is no option so far. In this section, we shall first discuss how to introduce control parameters to FISTA-BT which leads to a modified FISTA scheme, and then present convergence analysis.

3.1 A modified FISTA

Recall the t_k update rule of the original FISTA-BT [6],

$$t_k = \tfrac{1 + \sqrt{1 + 4t_{k-1}^2}}{2}, \ a_k = \tfrac{t_{k-1} - 1}{t_k}.$$

In the following, we replace the constants $1, 1$ and 4 in the update of t_k with three parameters p, q and r and study how they affect the behavior of t_k and consequently a_k.

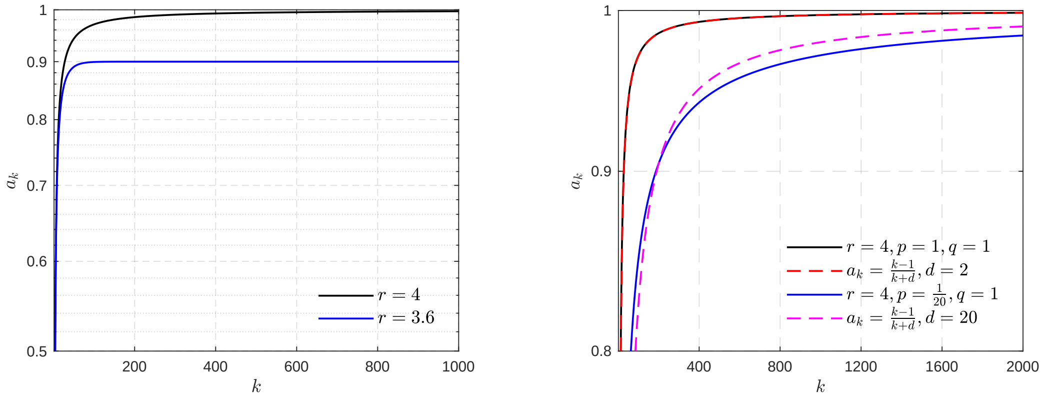
Observation I Consider first replacing 4 with a non-negative r, we get

$$t_k = \tfrac{1 + \sqrt{1 + r t_{k-1}^2}}{2}, \ a_k = \tfrac{t_{k-1} - 1}{t_k}. \tag{3.1}$$

With simple calculation, we obtain:

$$\begin{aligned} r \in]0, 4[&: t_k \to \frac{4}{4 - r} < +\infty, \ a_k \to \frac{r}{4} < 1, \\ r = 4 &: t_k \approx \frac{k+1}{2} \to +\infty, \ a_k \to 1, \\ r \in]4, +\infty[&: t_k \propto \Big(\frac{\sqrt{r}}{2}\Big)^k \to +\infty, \ a_k \to \frac{2}{\sqrt{r}} < 1, \end{aligned} \tag{3.2}$$

which implies that r controls the limiting value of t_k, hence that of a_k. In Figure 1 (a), we show graphically the behavior of a_k under two choices of r: $r = 4$ and $r = 3.6$.



(a) Value of a_k under different r

(b) Value of a_k under different p, q

Figure 1: Different effects of p, q and r. (a) r controls the limiting value of a_k; (b) p, q control the speed of a_k approaching its limit.

Observation II Now further replace the two 1's in (3.1) with $p, q > 0$, and restrict $r \in]0, 4]$:

$$t_k = \tfrac{p+\sqrt{q+rt_{k-1}^2}}{2}, \; a_k = \tfrac{t_{k-1}-1}{t_k}. \tag{3.3}$$

Depending on the choices of p, q and r, this time we have

$$\begin{aligned} r \in]0, 4[: t_k &\to \frac{2p+\Delta}{4-r} < +\infty, \; a_k \to \frac{2p+\Delta-(4-r)}{2p+\Delta} < 1, \\ r = 4 : t_k &\approx \frac{k+1}{2} p \to +\infty, \; a_k \to 1, \end{aligned} \tag{3.4}$$

where $\Delta \overset{\text{def}}{=} \sqrt{rp^2 + (4-r)q}$.

Equation (3.4) is quite similar to (3.2), in the sense that a_k converges to 1 for $r = 4$ and to some value smaller than 1 when $r < 4$. Moreover, for $r = 4$, the growth of t_k is controlled by p, indicating that we can control the speed of a_k approaching 1 via p, which is illustrated graphically in Figure 1 (b). Under $r = 4$, two different choices of p, q are considered, $(p, q) = (1, 1)$ and $(p, q) = (\frac{1}{20}, 1)$. Clearly, a_k approaches 1 much slower for the second choice of p, q. In comparison, we also add a case for (1.5) of FISTA-CD, for which a larger value of d leads to a slower speed of a_k approaching 1.

Remark 3.1. Let $r < 4$, and denote $t_\infty \overset{\text{def}}{=} \frac{2p+\Delta}{4-r}, a_\infty = \frac{2p+\Delta-(4-r)}{2p+\Delta}$ the limiting value of t_k, a_k, respectively. Depending on the initial value of t_0, we have $\begin{cases} t_0 < t_\infty : t_k \nearrow t_\infty, \; a_k \nearrow a_\infty; \\ t_0 = t_\infty : t_k \equiv t_\infty, \; a_k \equiv a_\infty; \\ t_0 > t_\infty : t_k \searrow t_\infty, \; a_k \searrow a_\infty. \end{cases}$

Algorithm 2: A modified FISTA scheme

```
Initial: $p, q > 0$ and $r \in ]0, 4]$, $t_0 = 1$, $\gamma \le 1/L$ and $x_0 \in \mathbb{R}^n, x_{-1} = x_0$.
repeat
    $t_k = \frac{p+\sqrt{q+rt_{k-1}^2}}{2}, \; a_k = \frac{t_{k-1}-1}{t_k},$
    $y_k = x_k + a_k(x_k - x_{k-1}),$                                   (3.5)
    $x_{k+1} = \mathrm{prox}_{\gamma R}\big(y_k - \gamma \nabla F(y_k)\big).$
until convergence;
```

A modified FISTA scheme Based on the above two observations of t_k, we propose a modified FISTA scheme, which we call "FISTA-Mod" for short and describe below in Algorithm 2.

Remark 3.2. When r is strictly smaller than 4, Algorithm 2 is simply a variant of the inertial Forward–Backward, and we refer to [16] for more details on its convergence properties.

3.2 Convergence properties of FISTA-Mod

The parameters p, q and r in FISTA-Mod allow us to control the behavior of t_k and a_k, hence providing possibilities to prove the convergence of the iterates $\{x_k\}_{k\in\mathbb{N}}$. Below we provide two convergence results for Algorithm 2: $o(1/k^2)$ convergence rate for $\Phi(x_k) - \min_{x\in\mathscr{H}}\Phi(x)$ and convergence of $\{x_k\}_{k\in\mathbb{N}}$ together with $o(1/k)$ rate for $\|x_k - x_{k-1}\|$. The proofs of these results are inspired by the work of [10, 2], and for the sake of self-consistency we present the details of the proofs.

3.2.1 Main result

We present below first the main result, and then provide the corresponding proofs. Let $x^\star \in \mathrm{Argmin}(\Phi)$ be a global minimizer of the problem.

Theorem 3.3 (Convergence of objective). *For the FISTA-Mod scheme (3.5), let $r = 4$ and choose $p \in]0,1], q > 0$ such that*

$$q \le (2-p)^2, \tag{3.6}$$

then it holds

$$\Phi(x_k) - \Phi(x^\star) \le \frac{2L}{p^2(k+1)^2}\|x_0 - x^\star\|^2. \tag{3.7}$$

Moreover, if $p \in]0,1[$ and $q \in [p^2, (2-p)^2]$, then $\Phi(x_k) - \Phi(x^\star) = o(1/k^2)$.

Remark 3.4. The $O(1/k^2)$ convergence rate (3.7) recovers the result of FISTA-BT [6] for $p = 1$. Since p appears in the denominator, this suggests that FISTA-BT has the *smallest* constant in the $O(1/k^2)$ rate.

Theorem 3.5 (Convergence of sequence). *For the FISTA-Mod scheme (3.5), let $r = 4, p \in]0,1[$ and $q \in [p^2, (2-p)^2]$, then the sequence $\{x_k\}_{k\in\mathbb{N}}$ generated by FISTA-Mod converges weakly to a global minimizer $x^\star$ of Φ. Moreover, $\|x_k - x_{k-1}\| = o(1/k)$.*

3.2.2 Proofs of Theorem 3.3

Before presenting the proof of Theorem 3.3, we recall the key points for establishing $O(1/k^2)$ convergence for FISTA-BT [6] and $o(1/k^2)$ convergence rate [10, 2]. In particular:

- t_k grows to $+\infty$ at a proper speed, *e.g.* $t_k \approx \frac{k+1}{2}$ as pointed out in [6];
- The sequence $\{t_k\}_{k\in\mathbb{N}}$ satisfies $t_k^2 - t_k \le t_{k-1}^2$. For example, for $t_k = \frac{1+\sqrt{1+4t_{k-1}^2}}{2}$, one has $t_k^2 - t_k = t_{k-1}^2$.

To further improve the $O(1/k^2)$ convergence rate to $o(1/k^2)$, the key is that the difference $t_{k-1}^2 - (t_k^2 - t_k)$ should also grow to $+\infty$ [10, 2]. For instance, for the FISTA-CD update rule (1.5), one has

$$t_{k-1}^2 - (t_k^2 - t_k) = \frac{1}{d^2}\big((d-2)k + d^2 - 3d + 3\big),$$

which goes to $+\infty$ as long as $d > 2$ [10, Eq. (13)]. It is worth noting that $t_{k-1}^2 - (t_k^2 - t_k) \to +\infty$ is also the key for proving the convergence of the iterates $\{x_k\}_{k\in\mathbb{N}}$.

We start with the following supporting lemmas. Recall in (3.4) that $t_k \approx \frac{k+1}{2}p$, we show in the lemma below that $\frac{k+1}{2}p$ is actually a lower bound of t_k.

Lemma 3.6 (Lower bound of t_k). *For the t_k update rule (3.3), set $r = 4$ and $p \in]0,1], q > 0$. Let $t_0 = 1$, then for all $k \in \mathbb{N}$, it holds that*

$$t_k \ge \frac{(k+1)p}{2}. \tag{3.8}$$

Remark 3.7. When $p = 1$, we have $t_k \geq \frac{k+1}{2}$ which recovers [6, Lemma 4.3].

Proof. Since $p \in]0,1]$, it is obvious that $t_0 = 1 \geq \frac{p}{2}$ and $t_1 = \frac{p+\sqrt{q+4}}{2} \geq \frac{p+2}{2} \geq p$. Now suppose (3.8) holds for a given $k \in \mathbb{N}$, *i.e.* $t_k \geq \frac{(k+1)p}{2}$. Then for $k+1$, we have $t_{k+1} - \frac{p}{2} = \frac{p+\sqrt{q+4t_k^2}}{2} - \frac{p}{2} > \frac{p+2t_k}{2} - \frac{p}{2} = t_k$ which concludes the proof. □

Lemma 3.8 (Lower bound of $t_{k-1}^2 - (t_k^2 - t_k)$). *For the t_k update rule (3.3), let $r = 4$ and $p \in [0,1], p^2 - q \leq 0$. Then there holds*

$$\frac{p(1-p)(k+1)}{2} \leq t_{k-1}^2 - (t_k^2 - t_k). \tag{3.9}$$

Remark 3.9. The inequality (3.9) implies that, if we choose $p < 1$, then $t_{k-1}^2 - (t_k^2 - t_k) \to +\infty$.

Proof. For (3.3), when $r = 4$, we have $t_k = \frac{p+\sqrt{q+4t_{k-1}^2}}{2} \Leftrightarrow t_k^2 - pt_k + \frac{1}{4}(p^2 - q) = t_{k-1}^2$. Since $p^2 \leq q$, then

$$\begin{aligned}
t_k^2 - pt_k + \tfrac{1}{4}(p^2 - q) = t_{k-1}^2 &\implies t_k^2 - pt_k \leq t_{k-1}^2 \\
&\iff t_k^2 - t_k + (1-p)t_k \leq t_{k-1}^2 \\
&\implies (1-p)t_k \leq t_{k-1}^2 - (t_k^2 - t_k) \\
\text{(Lemma 3.6)} &\implies \frac{p(1-p)(k+1)}{2} \leq (1-p)t_k \leq t_{k-1}^2 - (t_k^2 - t_k),
\end{aligned} \tag{3.10}$$

which concludes the proof. □

Remark 3.10. The first line of (3.10) implies that $t_k^2 - t_{k-1}^2 \leq pt_k$. Recently it is shown in [1] that $p < 1$ is the key for proving the convergence of the iterates $\{x_k\}_{k \in \mathbb{N}}$, see [1, Theorem 2.1].

The proof below is a combination of the result of [6, 10].

Proofs of Theorem 3.3. For (3.3), when $r = 4$, t_k is monotonically increasing as $t_k - t_{k-1} \geq \frac{p}{2} > 0$. Moreover,

$$\begin{aligned}
t_k^2 - pt_k + \tfrac{1}{4}(p^2 - q) = t_{k-1}^2 &\iff t_k^2 - t_k + (1-p)t_k + \tfrac{1}{4}(p^2 - q) = t_{k-1}^2 \\
&\implies t_k^2 - t_k + (1-p)t_0 + \tfrac{1}{4}(p^2 - q) \leq t_{k-1}^2 \\
(t_0 = 1) &\iff t_k^2 - t_k + \tfrac{1}{4}((2-p)^2 - q) \leq t_{k-1}^2 \\
\text{(owing to (3.6))} &\implies t_k^2 - t_k \leq t_{k-1}^2.
\end{aligned}$$

Define $v_k = \Phi(x_k) - \Phi(x^\star)$. Applying Lemma 2.5 at the points $(x = x_k, y = y_k)$ and at $(x = x^\star, y = y_k)$ leads to

$$\begin{aligned}
\tfrac{2}{L}(v_k - v_{k+1}) &\geq \|x_{k+1} - y_k\|^2 + 2\langle x_{k+1} - y_k,\, y_k - x_k\rangle \\
-\tfrac{2}{L}v_{k+1} &\geq \|x_{k+1} - y_k\|^2 + 2\langle x_{k+1} - y_k,\, y_k - x^\star\rangle,
\end{aligned}$$

where $x_{k+1} = e_\gamma(y_k)$ is used. Multiplying $t_k - 1$ to the first inequality and then adding to the second one yield,

$$\tfrac{2}{L}\big((t_k - 1)v_k - t_k v_{k+1}\big) \geq t_k\|x_{k+1} - y_k\|^2 + 2\langle x_{k+1} - y_k,\, t_k y_k - (t_k - 1)x_k - x^\star\rangle.$$

Multiply t_k to both sides of the above inequality and use the result $t_k^2 - t_k \leq t_{k-1}^2$, we get

$$\tfrac{2}{L}\big(t_{k-1}^2 v_k - t_k^2 v_{k+1}\big) \geq t_k^2\|x_{k+1} - y_k\|^2 + 2t_k\langle x_{k+1} - y_k,\, t_k y_k - (t_k - 1)x_k - x^\star\rangle.$$

Apply the Pythagoras relation $2\langle b - a, a - c\rangle = \|b - c\|^2 - \|a - b\|^2 - \|a - c\|^2$ to the last inner product of the above inequality we get

$$\begin{aligned}
\tfrac{2}{L}\big(t_{k-1}^2 v_k - t_k^2 v_{k+1}\big) &\geq \|t_k x_{k+1} - (t_k - 1)x_k - x^\star\|^2 - \|t_k y_k - (t_k - 1)x_k - x^\star\|^2 \\
&= \|t_k x_{k+1} - (t_k - 1)x_k - x^\star\|^2 - \|t_{k-1}x_k - (t_{k-1} - 1)x_{k-1} - x^\star\|^2.
\end{aligned} \tag{3.11}$$

If $a_k - a_{k+1} \geq b_{k+1} - b_k$ and $a_1 + b_1 < c$, then $a_k < c$ for all $k \geq 1$ [6, Lemma 4.2]. Hence, (3.11) yields,

$$\tfrac{2}{L} t_k^2 v_k \leq \|x_0 - x^\star\|.$$

Apply Lemma 3.6, we get

$$\Phi(x_k) - \Phi(x^\star) \leq \frac{2L}{p^2(k+1)^2}\|x_0 - x^\star\|^2,$$

which concludes the proof for the first claim (3.7).

Let $u_k = x_k + t_k(x_{k+1} - x_k)$. Applying Lemma 2.6 with $y = y_k, y^+ = x_{k+1}$ and $x = (1 - \frac{1}{t_k})x_k + \frac{1}{t_k}x^\star$ yields

$$\Phi(x_{k+1}) + \tfrac{1}{2\gamma}\|\tfrac{1}{t_k}u_k - \tfrac{1}{t_k}x^\star\|^2 \leq \Phi\big((1 - \tfrac{1}{t_k})x_k + \tfrac{1}{t_k}x^\star\big) + \tfrac{1}{2\gamma}\|\tfrac{1}{t_k}u_{k-1} - \tfrac{1}{t_k}x^\star\|^2.$$

Applying the convexity of Φ, we further get

$$\big(\Phi(x_{k+1}) - \Phi(x^\star)\big) - (1 - \tfrac{1}{t_k})\big(\Phi(x_k) - \Phi(x^\star)\big) \leq \frac{1}{2\gamma t_k^2}\big(\|u_{k-1} - x^\star\|^2 - \|u_k - x^\star\|^2\big).$$

Multiply t_k^2 to both sides of the above inequality,

$$t_k^2\big(\Phi(x_{k+1}) - \Phi(x^\star)\big) - (t_k^2 - t_k)\big(\Phi(x_k) - \Phi(x^\star)\big) \leq \tfrac{1}{2\gamma}\big(\|u_{k-1} - x^\star\|^2 - \|u_k - x^\star\|^2\big).$$

From Lemma 3.8, we have $\frac{p(1-p)(k+1)}{2} - t_{k-1}^2 \leq -(t_k^2 - t_k)$, hence

$$t_k^2\big(\Phi(x_{k+1}) - \Phi(x^\star)\big) - t_{k-1}^2\big(\Phi(x_k) - \Phi(x^\star)\big) + \frac{p(1-p)(k+1)}{2}\big(\Phi(x_k) - \Phi(x^\star)\big) \leq \tfrac{1}{2\gamma}\big(\|u_{k-1} - x^\star\|^2 - \|u_k - x^\star\|^2\big).$$

Summing the inequality from $k = 1$ to K, we get

$$t_K^2\big(\Phi(x_{K+1}) - \Phi(x^\star)\big) + \frac{p(1-p)}{2}\textstyle\sum_{j=1}^{K} j\big(\Phi(x_j) - \Phi(x^\star)\big) \leq \tfrac{1}{2\gamma}\big(\|u_0 - x^\star\|^2 - \|u_K - x^\star\|^2\big),$$

which means that $\sum_{j=1}^{+\infty} j\big(\Phi(x_j) - \Phi(x^\star)\big) < +\infty$, that is $\Phi(x_k) - \Phi(x^\star) = o(1/k^2)$. □

3.2.3 Proofs of Theorem 3.5

The proof of Theorem 3.5 is inspired by [10], where the authors showed that the key to prove the convergence of $\{x_k\}_{k\in\mathbb{N}}$ is the following summability

$$\textstyle\sum_{k\in\mathbb{N}} k\|x_k - x_{k-1}\|^2 < +\infty.$$

As previously mentioned, the major difference between FISTA-BT (1.4) and FISTA-CD (1.5) is that $t_{k-1}^2 - (t_k^2 - t_k) \to +\infty$ holds for FISTA-CD. For the proposed FISTA-Mod scheme, as $\frac{p(1-p)k}{2} \leq t_{k-1}^2 - (t_k^2 - t_k)$ also goes to $+\infty$ as long as p is strictly smaller than 1, this allows us to adapt the proof of [10] to FISTA-Mod, hence proving the convergence of $\{x_k\}_{k\in\mathbb{N}}$.

We need two supporting lemmas before presenting the proof of Theorem 3.5. Given $\ell \in \mathbb{N}_+$, define the truncated sum $S_\ell \overset{\text{def}}{=} \frac{q}{4p}\sum_{i=0}^{\ell}\frac{1}{1+i}$ and a new sequence $\bar{t}_k$ by

$$\bar{t}_k \overset{\text{def}}{=} 1 + S_\ell + \big(\tfrac{p}{2} + \tfrac{q}{4p(\ell+1)}\big)k.$$

We have the following lemma showing that $\bar{t}_k$ serves an upper bound of t_k.

Lemma 3.11 (Upper bound of t_k). *For the t_k update rule (3.3), let $r = 4$ and $p, q \in [0,1]$. For all $k \in \mathbb{N}$, it holds that $t_k \leq \bar{t}_k$.*

The purpose of bounding t_k from above by a linear function of k is such that we can eventually bound a_k from above, which is needed by the following lemma.

Proof. Given t_k, t_{k+1}, we have

$$t_{k+1} - t_k = \frac{p + \sqrt{q + 4t_k^2}}{2} - t_k = \frac{p}{2} + \frac{\sqrt{q + 4t_k^2} - 2t_k}{2} \leq \frac{p}{2} + \frac{\sqrt{(2t_k + q/(4t_k))^2} - 2t_k}{2} = \frac{p}{2} + \frac{q}{8t_k}.$$

Clearly, $t_0 \leq \bar{t}_0$. Suppose $t_k \leq \bar{t}_k$ for $\ell \leq k$ and recall that $t_k \geq \frac{k+1}{2}p$, then we have

$$\begin{aligned} t_{k+1} \leq t_k + \tfrac{p}{2} + \tfrac{q}{8t_k} \leq \bar{t}_k + \tfrac{p}{2} + \tfrac{q}{8t_k} &= 1 + S_\ell + \big(\tfrac{p}{2} + \tfrac{q}{4p(\ell+1)}\big)k + \tfrac{p}{2} + \tfrac{q}{8t_k} \\ &\leq 1 + S_\ell + \big(\tfrac{p}{2} + \tfrac{q}{4p(\ell+1)}\big)k + \tfrac{p}{2} + \tfrac{q}{4(k+1)p} \\ &\leq 1 + S_\ell + \big(\tfrac{p}{2} + \tfrac{q}{4p(\ell+1)}\big)k + \tfrac{p}{2} + \tfrac{q}{4(\ell+1)p} = \bar{t}_{k+1}, \end{aligned}$$

and we conclude the proof. □

Denote $\lceil x \rceil$ the smallest integer that is larger than x, and define the following two constants

$$b \overset{\text{def}}{=} \lceil \tfrac{p+2}{p+q/(2p(\ell+1))} \rceil \quad \text{and} \quad c \overset{\text{def}}{=} \tfrac{p+2+2S_\ell}{p+q/(2p(\ell+1))}.$$

Lemma 3.12. *For all $j \geq 1$, define $\beta_{j,k} \overset{\text{def}}{=} \prod_{i=j}^k a_i$ for all j,k, and $\beta_{j,k} = 1$ for all $k < j$. Let $\ell \geq \lceil \frac{q}{p(2-p)} \rceil$, then for all j, it holds that $\sum_{k=j}^\infty \beta_{j,k} \leq j + c + 2b$.*

Proof. We first show that a_k is bounded from above. From the definition of a_k we have

$$\begin{aligned} a_k = \frac{t_{k-1}-1}{t_k} = \frac{2t_{k-1}-2}{p+\sqrt{q+4t_{k-1}^2}} \leq \frac{p+2t_{k-1}-2-p}{p+2t_{k-1}} &= 1 - \frac{2+p}{p+2t_{k-1}} \\ \text{(Lemma 3.11)} &\leq 1 - \frac{2+p}{p+2+2S_\ell+(p+\frac{q}{2p(\ell+1)})k} = 1 - \frac{b}{k+c}. \end{aligned} \tag{3.12}$$

From (3.12) we have that

$$\beta_{j,k} = \prod_{i=j}^k a_i \leq \prod_{i=j}^k \frac{i+c-b}{i+c}.$$

For $k = j, ..., j+2b-1$, we have $\beta_{j,k} < 1$. Then for $k - j \geq 2b$,

$$\begin{aligned} \beta_{j,k} \leq \prod_{i=j}^k \frac{i+c-b}{i+c} &= \frac{j+c-b}{j+c}\,\frac{j+1+c-b}{j+1+c} \cdots \frac{j+c}{j+b+c}\,\frac{j+1+c}{j+b+1+c} \cdots \frac{k+c-b}{k+c} \\ &= \frac{(j+c-b)\cdots(j+c-1)}{(k+c-b+1)\cdots(k+c)} \leq \frac{(j+c-1)^b}{(k+c-b+1)^b}. \end{aligned}$$

Therefore,

$$\begin{aligned} \sum_{k=j}^\infty \beta_{j,k} \leq 2b + \sum_{k=j+2b}^\infty \beta_{j,k} &\leq 2b + (j+c-1)^b \sum_{k=j+2b}^\infty \frac{1}{(k+c-b+1)^b} \\ &\leq 2b + (j+c-1)^b \int_{x=j+2b}^\infty \frac{1}{(x+c-b+1)^b}\,\mathrm{d}x \\ &\leq 2b + (j+c-1)^b \frac{1}{b-1}\frac{1}{(j+b+c+1)^{b-1}} \\ &\leq 2b + \frac{1}{b-1}(j+c-1) \leq j+c+2b. \end{aligned}$$

The last inequality uses the fact that $b \geq 2$ for $\ell \geq \lceil \frac{q}{p(2-p)} \rceil$. □

Proofs of Theorem 3.5. Applying Lemma 2.6 with $y = y_k$ and $x = x_k$, we get

$$\Phi(x_{k+1}) + \tfrac{\|x_k - x_{k+1}\|^2}{2\gamma} \leq \Phi(x_k) + a_k^2 \tfrac{\|x_{k-1} - x_k\|^2}{2\gamma},$$

which means, let $\Delta_k \overset{\text{def}}{=} \frac{1}{2}\|x_k - x_{k-1}\|^2$, that $\Delta_{k+1} - a_k^2 \Delta_k \leq \gamma(v_k - v_{k+1})$. Denote the upper bound of a_k in (3.12) as $\bar{a}_k \overset{\text{def}}{=} 1 - \frac{b}{k+c}$, $\forall k \geq 2$, and let $\bar{a}_1 = 0$ since $a_1 = 0$. It is then straightforward that

$$\Delta_{k+1} - \bar{a}_k^2 \Delta_k \leq \Delta_{k+1} - a_k^2 \Delta_k \leq \gamma(v_k - v_{k+1}).$$

Multiplying the above inequality with $(k+c)^2$ and summing from $k = 1$ to K lead to

$$\sum_{k=1}^K (k+c)^2 (\Delta_{k+1} - \bar{a}_k^2 \Delta_k) \leq \gamma \sum_{k=1}^K (k+c)^2 (v_k - v_{k+1}).$$

Since $\bar{a}_1 = 0$, we derive from above that

$$\begin{aligned}
\textstyle\sum_{k=1}^{K}(k+c)^2(\Delta_{k+1} - \bar{a}_k^2\Delta_k) &= (K+c)^2\Delta_{K+1} + \textstyle\sum_{k=2}^{K}\big((k+c-1)^2 - (k+c)^2\bar{a}_k^2\big)\Delta_k \\
&= (K+c)^2\Delta_{K+1} + \textstyle\sum_{k=2}^{K}\big((k+c-1)^2 - (k+c-b)^2\big)\Delta_k \\
&\leq (K+c)^2\Delta_{K+1} + \textstyle\sum_{k=2}^{K}2(b-1)(k+c)\Delta_k \\
&\leq \gamma\big((c+1)^2 w_1 - (c+K)^2 w_{K+1}\big) + \gamma\textstyle\sum_{k=2}^{K}\big((k+c)^2 - (k+c-1)^2\big)v_k \\
&\leq \gamma\big((c+1)^2 w_1 - (c+K)^2 w_{K+1}\big) + 2\gamma\textstyle\sum_{k=2}^{K}(k+c)v_k.
\end{aligned}$$

From the proof of Theorem 3.3, we have that $\sum_{k\in\mathbb{N}} kv_k < +\infty$, which in turn implies that $\{k\Delta_k\}_{k\in\mathbb{N}}$ is *summable* and that sequence $\{k^2\Delta_k\}_{k\in\mathbb{N}}$ is bounded, which also indicates $\|x_k - x_{k-1}\| = o(1/k)$.

Now define $\psi_k \overset{\text{def}}{=} \frac{1}{2}\|x_k - x^\star\|^2$ and $\phi_k \overset{\text{def}}{=} \frac{1}{2}\|y_k - x_{k+1}\|^2$. By applying the definition of y_k, we have

$$\begin{aligned}
\psi_k - \psi_{k+1} &= \tfrac{1}{2}\langle x_k - x^\star + x_{k+1} - x^\star, x_k - x_{k+1}\rangle \\
&= \Delta_{k+1} + \langle y_{a,k} - x_{k+1}, x_{k+1} - x^\star\rangle - a_k\langle x_k - x_{k-1}, x_{k+1} - x^\star\rangle \\
&\geq \Delta_{k+1} + \gamma\langle \nabla F(y_k) - \nabla F(x^\star), x_{k+1} - x^\star\rangle - a_k\langle x_k - x_{k-1}, x_{k+1} - x^\star\rangle.
\end{aligned} \tag{3.13}$$

As ∇F is $\frac{1}{L}$-cocoercive (Definition 2.2), applying Young's inequality yields

$$\begin{aligned}
\langle \nabla F(y_k) - \nabla F(x^\star), x_{k+1} - x^\star\rangle &\geq \tfrac{1}{L}\|\nabla F(y_k) - \nabla F(x^\star)\|^2 + \langle \nabla F(y_k) - \nabla F(x^\star), x_{k+1} - y_k\rangle \\
&\geq \tfrac{1}{L}\|\nabla F(y_k) - \nabla F(x^\star)\|^2 - \tfrac{1}{L}\|\nabla F(y_k) - \nabla F(x^\star)\|^2 - \tfrac{L}{2}\phi_k = -\tfrac{L}{2}\phi_k.
\end{aligned} \tag{3.14}$$

Back to (3.13), we get

$$\psi_k - \psi_{k+1} \geq \Delta_{k+1} - \frac{\gamma L}{2}\phi_k - a_k\langle x_k - x_{k-1}, x_{k+1} - x^\star\rangle. \tag{3.15}$$

For $\langle x_k - x_{k-1}, x_{k+1} - x^\star\rangle$, we have

$$\begin{aligned}
\langle x_k - x_{k-1}, x_{k+1} - x^\star\rangle &= \langle x_k - x_{k-1}, x_{k+1} - x_k\rangle + \langle x_k - x_{k-1}, x_k - x^\star\rangle \\
&= \langle x_k - x_{k-1}, x_{k+1} - x_k\rangle + (\Delta_k + \psi_k - \psi_{k-1}),
\end{aligned} \tag{3.16}$$

where we applied the usual Pythagoras relation to $\langle x_k - x_{k-1}, x_k - x^\star\rangle$. Putting (3.16) back into (3.15) and rearranging terms yield

$$\begin{aligned}
\psi_{k+1} - \psi_k - a_k(\psi_k - \psi_{k-1}) &\leq -\Delta_{k+1} + \frac{\gamma L}{2}\phi_k + a_k\langle x_k - x_{k-1}, x_{k+1} - x_k\rangle + a_k\Delta_k \\
&= -\Delta_{k+1} + \frac{\gamma L}{2}\phi_k + \langle y_k - x_k, x_{k+1} - x_k\rangle + a_k\Delta_k \\
&= -\Delta_{k+1} + \frac{\gamma L}{2}\phi_k + \big(a_k^2\Delta_k + \Delta_{k+1} - \tfrac{1}{2}\|y_k - x_{k+1}\|^2\big) + a_k\Delta_k \\
&= \frac{\gamma L - 1}{2}\phi_k + (a_k + a_k^2)\Delta_k,
\end{aligned} \tag{3.17}$$

where the Pythagoras relation is applied again to $\langle y_k - x_k, x_{k+1} - x_k\rangle$. Since $\gamma \in]0, 1/L]$ and $a_k \leq 1$, we get from above that

$$\psi_{k+1} - \psi_k - a_k(\psi_k - \psi_{k-1}) \leq 2a_k\Delta_k.$$

Define $\xi_k = \max\{0, \psi_k - \psi_{k-1}\}$, then

$$\xi_{k+1} \leq a_k(\xi_k + 2\Delta_k) \leq 2\textstyle\sum_{j=2}^{k}\big(\prod_{l=j}^{k} a_l\big)\Delta_j = 2\sum_{j=2}^{k}\beta_{j,k}\Delta_j,$$

Applying Lemma 3.12 and the summability of $\{k\Delta_k\}_{k\in\mathbb{N}}$ leads to

$$\textstyle\sum_{k=2}^{+\infty}\xi_k \leq 2\sum_{k=1}^{+\infty}\sum_{j=2}^{k}\beta_{j,k}\Delta_j = 2\sum_{j=2}^{k}\Delta_j\sum_{k=1}^{+\infty}\beta_{j,k} \leq 2\sum_{j=2}^{k}(j+c+2b)\Delta_j < +\infty.$$

Then we have

$$\Phi_{k+1} - \textstyle\sum_{j=1}^{k+1}[\theta_j]_+ \leq \Phi_{k+1} - \theta_{k+1} - \sum_{j=1}^{k}[\theta_j]_+ = \Phi_k - \sum_{j=1}^{k}[\theta_j]_+,$$

which means $\{\Phi_k - \sum_{j=1}^{k}[\theta_j]_+\}_{k\in\mathbb{N}}$ is monotone non-increasing, hence convergent. It is immediate that $\{\Phi_k\}_{k\in\mathbb{N}}$ is also convergent, meaning that $\lim_{k\to+\infty}\|x_k - x^\star\|$ exists for any $x^\star$ such that $0 \in A(x^\star)+B(x^\star)$.

Let $\bar{x}$ be a weak cluster point of $\{x_k\}_{k\in\mathbb{N}}$, and let us fix a subsequence, say $x_{k_j} \rightharpoonup \bar{x}$. Applying Lemma 2.4 with $y = y_{k_j}$, we get

$$g_{k_j} \overset{\text{def}}{=} \tfrac{y_{k_j}-x_{k_j+1}}{\gamma} - \nabla F(y_{k_j}) \in \partial R(x_{k_j+1}).$$

Since ∇F is cocoercive and $y_{k_j} = x_{k_j} + a_{k_j}(x_{k_j} - x_{k_j-1}) \rightharpoonup \bar{x}$, we have $\nabla F(y_{k_j}) \to \nabla F(\bar{x})$. In turn, $u_{k_j} \to -\nabla F(\bar{x})$ since $\gamma > 0$. Since $(x_{k_j+1}, u_{k_j}) \in \mathrm{gph}(\partial R)$, and the graph of the maximal monotone operator ∂R is sequentially weakly-strongly closed in $\mathcal{H}\times\mathcal{H}$, we get that $-\nabla F(\bar{x}) \in \partial R(\bar{x})$, *i.e.* $\bar{x}$ is a solution of ($\mathscr{P}$). Opial's Theorem [25] then concludes the proof. □

4 Lazy-start strategy

From the last section, the benefits of free parameters p,q,r in FISTA-Mod are $o(1/k^2)$ convergence rate in objective function value and convergence of sequence. In this section, we further show that the degree of freedom provided by these parameters allows us to design a so-called "lazy-start strategy" which can make FISTA-Mod/FISTA-CD much faster in practice.

Proposition 4.1 (Lazy-start FISTA). *For FISTA-Mod and FISTA-CD, consider the following choices of p,q and d respectively:*

FISTA-Mod $p \in [\frac{1}{80}, \frac{1}{10}], q \in [0,1]$ *and* $r = 4$*;*
FISTA-CD $d \in [10, 80]$.

Remark 4.2. The intervals for p and d are obtained from practical observations and not inclusive. Take FISTA-CD for example, there can be problems where $d < 10$ or $d > 80$ provides even faster performances.

The main reason of calling the above strategy "lazy-start" is that it slows down the speed of a_k converging to 1; Recall Figure 1 (b). To discuss the advantage of lazy-start, we consider the simple least square problem:

$$\min_{x\in\mathbb{R}^{201}} \{F(x) \overset{\text{def}}{=} \tfrac{1}{2}\|Ax\|^2\}, \tag{4.1}$$

where $A \in \mathbb{R}^{201\times 201}$ is of the form

$$A = \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}_{201\times 201}.$$

In this example, F is strongly convex and admits a unique minimizer $x^\star = 0$.

In what follows, we first discuss the advantage of lazy-start in the discrete setting, and then in the continuous dynamical system setting.

4.1 Advantage of lazy-start

Specialising FISTA-CD to solve (4.1), we get

$$\begin{aligned} y_k &= x_k + \tfrac{k-1}{k+d}(x_k - x_{k-1}) \\ x_{k+1} &= y_k - \tfrac{1}{L}A^T A y_k = (\mathrm{Id} - \tfrac{1}{L}A^T A)y_k. \end{aligned} \tag{4.2}$$

To show the benefits of lazy-start, two different values of d are considered:

- FISTA-CD with $d = 2$;
- Lazy-start FISTA-CD with $d = 20$.

The convergence of $\|x_k - x^\star\|$ for the two choices of d are plotted in Figure 2, where the red line represents $d = 2$ and the black line for $d = 20$. The starting points x_0 for both cases are the same and chosen such that $\|x_0 - x^\star\| = 1$. It can be observed that the lazy-start one is significantly faster than the normal choice after iteration step $k = 2 \times 10^5$.

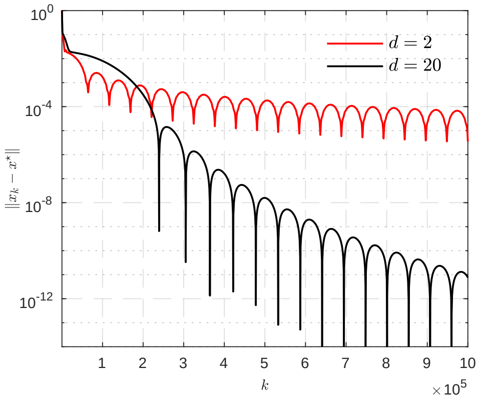


Figure 2: Convergence comparison of $\|x_k - x^\star\|$ of FISTA-CD for $d = 2$ and $d = 20$.

To explain such a difference, we need the following steps:

(1) Fixed-point characterization of (4.2): the iteration can be written as a linear system owing to the quadratic form of the problem; See (4.3).
(2) Spectral property of the linear system: the spectral property of the linear system is controlled only by d.
(3) Advantage of lazy-start: comparison of spectral properties under different choices of d.

It is worth noting that, the convergence seen in Figure 2 appears not only for (4.1), but rather is observed in many problems; see Section 7 for more examples.

Fixed-point formulation of (4.2) Denote $G = \mathrm{Id} - \frac{1}{L}A^T A$, we have from (4.2) that,

$$x_{k+1} - x^\star = G(y_k - x^\star) = (1 + a_k)G(x_k - x^\star) - a_k G(x_{k-1} - x^\star).$$

Define

$$z_k \overset{\text{def}}{=} \begin{pmatrix} x_k - x^\star \\ x_{k-1} - x^\star \end{pmatrix} \quad \text{and} \quad M_{d,k} \overset{\text{def}}{=} \begin{bmatrix} (1+a_k)G & -a_k G \\ \mathrm{Id} & 0 \end{bmatrix}. \tag{4.3}$$

Then it is immediate that

$$z_{k+1} = M_{d,k} z_k, \tag{4.4}$$

which is the fixed-point characterization of (4.2). Denote $\widetilde{M}_{d,k} \overset{\text{def}}{=} \prod_{i=1}^{k-1} M_{d,k-i}$, then recursively apply the above relation, we get

$$z_k = \widetilde{M}_{d,k} z_1.$$

Spectral property of $\widetilde{M}_{d,k}$ From above it is immediate that

$$\|z_k\| = \|\widetilde{M}_{d,k}\| \|z_1\|.$$

To set up the comparison between $d = 2$ and $d = 20$, we need to compute spectral property of $\|\widetilde{M}_k\|$:

- Let $\rho_{d,i}$ be the *leading eigenvalue* of $M_{d,i}$ for $i = 1, ..., k-1$, then there exists $\mathscr{C} > 0$ such that
$$\|\widetilde{M}_{d,k}\| \leq \mathscr{E}_{d,k} \overset{\text{def}}{=} \mathscr{C} \prod_{i=1}^{k-1} |\rho_{d,k-i}| \tag{4.5}$$
holds for all $k \geq 1$. We call $\mathscr{E}_{d,k}$ the envelope of $\|\widetilde{M}_{d,k}\|$. Unfortunately, unlike the case of $M_{d,k}$, this time we can only discuss through numerical illustration.
- Let α be the smallest eigenvalue of $A^T A$ and $\eta = 1 - \alpha/L$ the leading eigenvalue of G. Owing to the

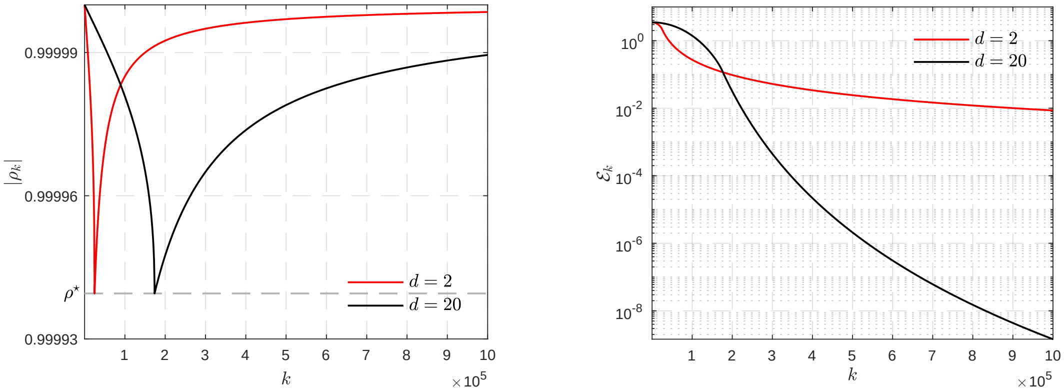
result of [16], for each $M_{d,k}$, the magnitude of its leading eigenvalue $\rho_{d,k}$ reads:

$$|\rho_{d,k}| = \begin{cases} \frac{(1+a_k)\eta + \sqrt{(1+a_k)^2\eta^2 - 4a_k\eta}}{2} < 1 : a_k \leq a^\star, \\ \sqrt{a_k\eta} < 1 : a_k \geq a^\star, \end{cases} \tag{4.6}$$

where $a^\star = \frac{1-\sqrt{\alpha/L}}{1+\sqrt{\alpha/L}}$. Moreover, $|\rho_k|$ attains the minimal value $\rho^\star = 1 - \sqrt{\alpha/L}$ when $a_k = a^\star$ [16]. For more details about the dependence of $\rho_{d,k}$ on η and a_k, we refer to [16, 14]. Below we inspect the value of $|\rho_{d,k}|$ under $d = 2$ and $d = 20$. The modulus of $|\rho_{d,k}|$ for $d = 2, 20$ are shown in Figure 3 (a), where the red line is $|\rho_{2,k}|$ and the black line stands for $|\rho_{20,k}|$:

- In both cases, the values of $|\rho_{2,k}|, |\rho_{20,k}|$ decrease first, until reaching $\rho^\star = 1 - \sqrt{\alpha/L}$, and then start to increase until they reach $\sqrt{\eta}$;
- Choosing $d = 20$ slows the speed at which a_k is increasing (see Figure 1), therefore also slows the speed at which $|\rho_{20,k}|$ approaches $\rho^\star$. Such a difference in approach to $\rho^\star$ is key for the lazy-start strategy being faster.

Denote K_{eq} the point $|\rho_{20,k}|$ equals to $\rho^\star$, then we have $K_{\text{eq}} = \lceil \frac{1+20a^\star}{1-a^\star} \rceil$.



(a) Value of $|\rho_{d,k}|$

(b) Value of $\mathcal{E}_{d,k}$

Figure 3: The value of $|\rho_{d,k}|$ and $\mathcal{E}_{d,k}$ under $d = 2, 20$.

The advantage of lazy-start Now we compare $\mathcal{E}_{2,k}, \mathcal{E}_{20,k}$, whose values are plotted in Figure 3 (b), where the red and black lines are corresponding to $\mathcal{E}_{2,k}$ and $\mathcal{E}_{20,k}$ respectively. Observe that, $\mathcal{E}_{2,k}$ and $\mathcal{E}_{20,k}$ intersect for certain k which turns out very close to K_{eq}. For $k \geq K_{\text{eq}}$, the difference between $\mathcal{E}_{2,k}$ and $\mathcal{E}_{20,k}$ becomes increasingly large.

From (4.6) and the definition of a_k, we have that for $k \geq K_{\text{eq}}$,

$$|\rho_{2,k}| = \sqrt{\tfrac{k-1}{k+2}\eta} \geq |\rho_{20,k}| = \sqrt{\tfrac{k-1}{k+20}\eta}.$$

Define the accumulation of $\frac{|\rho_{2,i}|}{|\rho_{20,i}|}$ by $\mathscr{R}_k \overset{\text{def}}{=} \prod_{i=K_{\text{eq}}}^{k} \frac{|\rho_{2,i}|}{|\rho_{20,i}|} = \prod_{i=K_{\text{eq}}}^{k} \sqrt{\frac{i+20}{i+2}}$ and let $k \geq K_{\text{eq}} + 36$, we get

$$\begin{aligned} \mathscr{R}_k &= \textstyle\prod_{i=K_{\text{eq}}}^{k} \frac{|\rho_{d_1,i}|}{|\rho_{d_2,i}|} = \prod_{i=K_{\text{eq}}}^{k} \sqrt{\frac{i+20}{i+2}} \\ &= \textstyle\prod_{i=K_{\text{eq}}}^{k} \Big(\frac{K_{\text{eq}}+20}{K_{\text{eq}}+2} \frac{K_{\text{eq}}+1+20}{K_{\text{eq}}+1+2} \cdots \frac{K_{\text{eq}}+17+20}{K_{\text{eq}}+17+2} \frac{K_{\text{eq}}+18+20}{K_{\text{eq}}+18+2} \cdots \frac{k-2+20}{k-2+2} \frac{k-1+20}{k-1+2} \frac{k+20}{k+2} \Big)^{1/2} \\ &= \textstyle\prod_{j=0}^{17} \Big(\frac{k+3+j}{K_{\text{eq}}+2+j} \Big)^{1/2} \approx \Big(\frac{k+20}{K_{\text{eq}}+19} \Big)^9 = \Big(\frac{2}{\sqrt{C}+1} \Big)^9 \Big(\frac{k+20}{21} \Big)^9, \end{aligned} \tag{4.7}$$

where $C \overset{\text{def}}{=} L/\alpha$ is the condition number of (4.1). To verify the accuracy of the above approximation, for the considered problem (4.1), we have $L = 16$ and $\alpha = 5.85 \times 10^{-8}$. Consequently, $C = \frac{L}{\alpha} = 2.735 \times 10^8$. Let

$k = 10^6$ and substitute them into (4.7), we have $\mathscr{R}_k \approx 5.98 \times 10^6$, while for $\mathscr{E}_{d,k}$ we have

$$\frac{\mathscr{E}_{2,k=10^6}}{\mathscr{E}_{20,k=10^6}} = 5.96 \times 10^6,$$

which means (4.7) is a good approximation of the envelope ratio $\mathscr{E}_{2,k}/\mathscr{E}_{20,k}$.

The above discussion is mainly about the envelope $\mathscr{E}_{d,k}$. In terms of what really happens on $\|x_k - x^\star\|$ for $d = 2$ and $d = 20$: from Figure 2, we have that for $k = 10^6$, $\|x_k - x^\star\|$ of $d = 2$ is about 2×10^6 larger than that of $d = 20$. Compared with 5.98×10^6, we can conclude that (4.7) is able to accurately estimate the order of acceleration obtained by a lazy-start strategy.

4.2 Quantifying the advantage of lazy-start

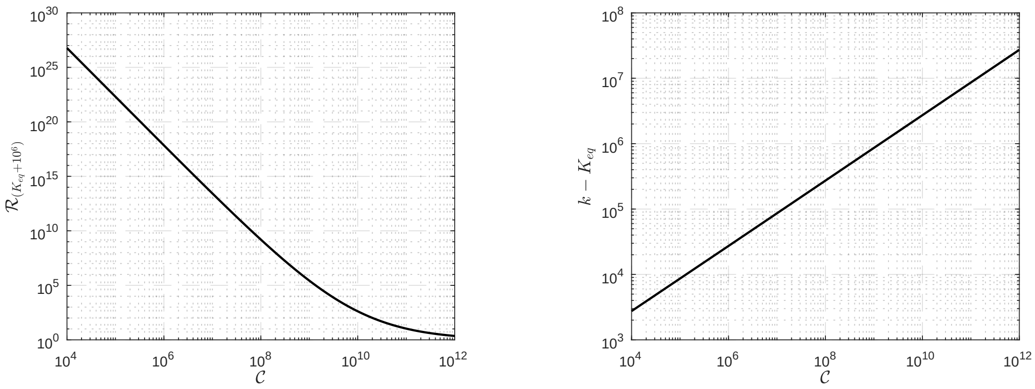
The approximation (4.7) indicates that $\mathscr{R}_k$ is a function of C and k, in the following we discuss the dependence of $\mathscr{R}_k$ on C and k from two perspectives.

Fixed k First consider $C \in [10^4, 10^{12}]$ and let $k = K_{\text{eq}} + 10^6$, note that K_{eq} is changing over C. This setting is to check how much better $d = 20$ is than $d = 2$ in terms of $\|x_k - x^\star\|$ if we run the iteration (4.2) 10^6 more steps after K_{eq}. The obtained value of $\mathscr{R}_k$ is shown in Figure 4 (a). As we can see, when C is small, *e.g.* $C = 10^4$, the advantage can be as large as 10^{27} times and decrease to almost 1 for $C = 10^{12}$. However, it should be noted that for this large C, $K_{\text{eq}} + 10^6$ steps of iteration could be not enough for producing a satisfactory output.

Fixed $\mathscr{R}_k$ The second part is to check for fixed $\mathscr{R}_k = \mathscr{R}$, *e.g.* $\mathscr{R} = 10^5$, how many more steps are needed after K_{eq}. From (4.7), simple calculation yields

$$k - K_{\text{eq}} = \frac{21(\sqrt{C}+1)}{2} \sqrt[9]{\mathscr{R}} - 20.$$

Let again $C \in [10^4, 10^{12}]$, the value of $k - K_{\text{eq}}$ is shown in Figure 4 (b). We can observe that when $C = 10^4$, only around 2,000 steps are needed, while about 2×10^7 steps are needed for $C = 10^{12}$.



(a) Value of $\mathscr{R}_k$ when fix $k = K_{\text{eq}} + 10^6$

(b) Value of $k - K_{\text{eq}}$ when fix $\mathscr{R}_k = 10^5$

Figure 4: The dependence of $\mathscr{R}_k$ on the iteration number k and the condition number C.

Remark 4.3. When C and k are fixed, $\mathscr{R}_k$ increases with d. This means if we consider only $\mathscr{R}_k$, then the larger value of d the better. However, one should not do so in practice, as larger d will make the value of K_{eq} much larger. As a result, proper choice of d is a trade-off between K_{eq} and $\mathscr{R}_k$, which is the content of the next part.

4.3 Continuous dynamical system perspective

The above discussion implies the existence optimal choices of d. From continuous dynamical system perspective, we show that an optimal d does indeed exists. What is interesting is that the optimal d does not depend on condition number of the problem, but the accuracy of solution. The analysis is inspired by the result of [30].

4.3.1 Optimal choice damping coefficient

To prove the claim, we start from continuous dynamical system (4.8) first, showing that larger values of ω below leads to faster convergence, and then back to the discrete setting for the proposed claim.

For problem (4.1), the associated continuous dynamical system reads:

$$\ddot{x} + \frac{\omega}{t}\dot{x} + A^T A x = 0, \tag{4.8}$$

where ω is the damping coefficient. Since $A^T A$ is symmetric, it can be diagonalised with invertible matrix P and diagonal matrix $\Lambda = \mathrm{diag}(\lambda_1, \cdots, \lambda_n)$: $A^T A = P\Lambda P^{-1}$. Let $y = P^{-1}x$, then we get

$$\ddot{y} + \frac{\omega}{t}\dot{y} + \Lambda y = 0.$$

Since Λ is diagonal, it is sufficient to consider each entry of y that

$$\ddot{y}_i + \frac{\omega}{t}\dot{y}_i + \lambda_i y_i = 0, \quad i = 1, \cdots, n,$$

where n is the dimension of the problem. Let $\omega_i = \omega\lambda_i^{-1/2}$, $\nu_i = \frac{\omega_i - 1}{2}$ and $z_i(t) = t^{\nu_i} y_i(\lambda_i^{-1/2} t)$ for $i = 1, \cdots, n$. This change of variables results in Bessel's differential equations [30]:

$$t^2 \ddot{z}_i + t\dot{z}_i + (t^2 - \nu_i^2) z_i = 0, \quad i = 1, \cdots, n,$$

whose solution is

$$z_i = c_{i,1} J_{\nu_i} + c_{i,2} Y_{\nu_i}, \quad i = 1, \cdots, n,$$

where J_{ν_i} and Y_{ν_i} are the first and second kind of Bessel functions. Therefore, we get for y_i that

$$\begin{aligned} y_i(\lambda_i^{-1/2} t) &= t^{-\nu_i} z_i(t) = t^{-\nu_i}\big(c_{i,1} J_{\nu_i}(t) + c_{i,2} Y_{\nu_i}(t)\big), \\ y_i(t) &= (\lambda_i^{1/2} t)^{-\nu_i}\big(c_{i,1} J_{\nu_i}(\lambda_i^{1/2} t) + c_{i,2} Y_{\nu_i}(\lambda_i^{1/2} t)\big). \end{aligned}$$

For J_{ν_i} and Y_{ν_i}, recall the following asymptotic forms of Bessel functions for positive and large argument t:

$$J_\nu(t) = \sqrt{\frac{2}{\pi t}}\Big(\cos\big(t - \frac{\nu\pi}{2} - \frac{\pi}{4}\big) + O(t^{-1})\Big) \quad \text{and} \quad Y_\nu(t) = \sqrt{\frac{2}{\pi t}}\Big(\sin\big(t - \frac{\nu\pi}{2} - \frac{\pi}{4}\big) + O(t^{-1})\Big).$$

As a result,

$$J_{\nu_i}(\lambda_i^{1/2} t) = \sqrt{\frac{2}{\pi\lambda_i^{1/2} t}}\Big(\cos\big(\lambda_i^{1/2} t - \frac{(\omega\lambda_i^{-1/2} - 1)\pi}{4} - \frac{\pi}{4}\big) + O(t^{-1})\Big) = \sqrt{\frac{2}{\pi\lambda_i^{1/2} t}}\Big(\cos\big(\lambda_i^{1/2} t - \frac{\omega\lambda_i^{-1/2}\pi}{4}\big) + O(t^{-1})\Big),$$

$$Y_{\nu_i}(\lambda_i^{1/2} t) = \sqrt{\frac{2}{\pi\lambda_i^{1/2} t}}\Big(\sin\big(\lambda_i^{1/2} t - \frac{\omega\lambda_i^{-1/2}\pi}{4}\big) + O(t^{-1})\Big).$$

Eventually, we get for y_i that

$$y_i(t) = \sqrt{c_{i,1}^2 + c_{i,2}^2}\sqrt{\frac{2}{\pi}}\lambda_i^{-\frac{\omega_i}{4}} t^{-\frac{\omega_i}{2}} \sin\big(\lambda_i^{1/2} t - \frac{\omega\lambda_i^{-1/2}\pi}{4} + \theta_i\big) + O(t^{-1-\frac{\omega_i}{2}}), \tag{4.9}$$

where $\theta_i = \arctan\frac{c_{i,1}}{c_{i,2}}$ depends on $c_{i,1}$ and $c_{i,2}$ which are determined by the initial condition.

From the above asymptotics, we conclude that, in the continuum case (*i.e.* ODEs), the convergence is faster for larger ω. However, in the discrete case, we have to also consider the numerical error. We consider the following FISTA-CD scheme

$$\begin{aligned} y_k &= x_k + \tfrac{k-1}{k+d}(x_k - x_{k-1}), \\ x_{k+1} &= y_k - \gamma\nabla F(y_k), \end{aligned}$$

where $d = \omega - 1$. Note that $x_k \approx x(k\tau)$ with step-size $\tau = \sqrt{\gamma}$. The algorithm is then rewritten as

$$\frac{x_{k+1} - x_k}{\tau} = \frac{k-1}{k+d}\frac{x_k - x{-}_{k-1}}{\tau} - \tau\nabla F(y_k).$$

By Taylor expansion in τ, we have

$$\begin{aligned}\dot{x}(t)+\tfrac{1}{2}\ddot{x}(t)\tau+o(\tau) &= \tfrac{t-\tau}{t+d\tau}\big(\dot{x}(t)-\tfrac{1}{2}\ddot{x}(t)\tau+o(\tau)\big)-\tau\nabla F(x(t))+o(\tau)\\ &= (1-\tfrac{\omega\tau}{t})\big(\dot{x}(t)-\tfrac{1}{2}\ddot{x}(t)\tau+o(\tau)\big)-\tau\nabla F(x(t))+o(\tau).\end{aligned}$$

Note that in the last step we have applied expansion

$$\frac{t-\tau}{t+d\tau}=1-\frac{\omega\tau}{t+(\omega-1)\tau}=1-\frac{\omega\tau}{t}+\frac{\omega(\omega-1)\tau^2}{t^2}+\cdots. \tag{4.10}$$

This makes sense only for $\frac{(\omega-1)\tau}{t}<1$. More precisely, the numerical error at time T is $\varepsilon_{\text{num}}=\frac{\omega\tau}{T}$.

By approximation (4.9), the truncation error (tolerance) is $\varepsilon=|x(T)-x(+\infty)|=|x(T)|=\lambda_1^{-\frac{\bar{\omega}}{4}}T^{-\frac{\bar{\omega}}{2}}$ where $\bar{\omega}=\max_{1\leq i\leq n}\{\omega_i\}$. Thus $T^{-1}=\varepsilon^{\frac{2}{\bar{\omega}}}\lambda_1^{\frac{1}{2}}$ and $\varepsilon_{\text{num}}=\tau\lambda_1^{\frac{1}{2}}\omega\varepsilon^{\frac{2}{\bar{\omega}}}$. We need to minimize

$$\log\varepsilon_{\text{num}}=\log(\tau\lambda_1^{1/2})+\log\omega+\tfrac{2}{\bar{\omega}}\log\varepsilon=\log(\tau\lambda_1^{1/2})+\log\omega+\frac{2\lambda_1^{1/2}}{\bar{\omega}}\log\varepsilon,\quad \bar{\omega}\geq 3,$$

which leads to $0=\frac{1}{\bar{\omega}}+\frac{2\lambda_1^{1/2}}{\bar{\omega}^2}\log\varepsilon$. As a result, the optimal choice of $\bar{\omega}$ is $\omega=-2\lambda_1^{1/2}\log\varepsilon$, hence $-2\lambda_1^{1/2}\log\varepsilon-1$ for d.

4.3.2 Optimal lazy-start parameters

Now we turn to the discrete case and discuss the optimal d, through the envelope $\mathscr{E}_{d,k}$.

Optimal d for $\|x_k-x^\star\|$ We continue using problem (4.1), with condition number $C=2.735\times10^8$. Consider several different values of d which are $d\in[5,15,25,35,45]$. The values of corresponding $\mathscr{E}_{d,k}$ are plotted in Figure 5 (a). For each $k\in[1,10^6]$, the minimum of $\mathscr{E}_{d,k}$ is computed and plotted as a red dotted line.

From Figure 5 (a), it can be observed that for each $d\in[5,15,25,35,45]$, their corresponding $\mathscr{E}_{d,k}$ is the smallest for a certain range of k. For instance, for $d=5$, $\mathscr{E}_{5,k}$ is the smallest for k between 1 and about 1.75×10^5. This verifies the result from continuous dynamical system that

- There exists an optimal choice of d;
- The optimal d depends on the accuracy of x_k.

To illustrate, we consider the following test: under a given tolerance tol $\in\{-2,...,-10\}$, for each $d\in[2,100]$ compute the minimal number of iterations, *i.e.* k, needed such that

$$\log(\mathscr{E}_{d,k})\leq \text{tol}.$$

The obtained results are shown in Figure 5 (b), from where we can observe that for each tol $\in\{-2,...,-10\}$, the corresponding k is a smooth curve that admits a minimal value $k^\star_{\text{tol}}$ for optimal $d^\star_{\text{tol}}$. The red line segment connects all the points of $(d^\star_{\text{tol}},k^\star_{\text{tol}})$ which almost is a straight line. It indicates that one should choose small d for high accuracy and increase the value for lower accuracy.

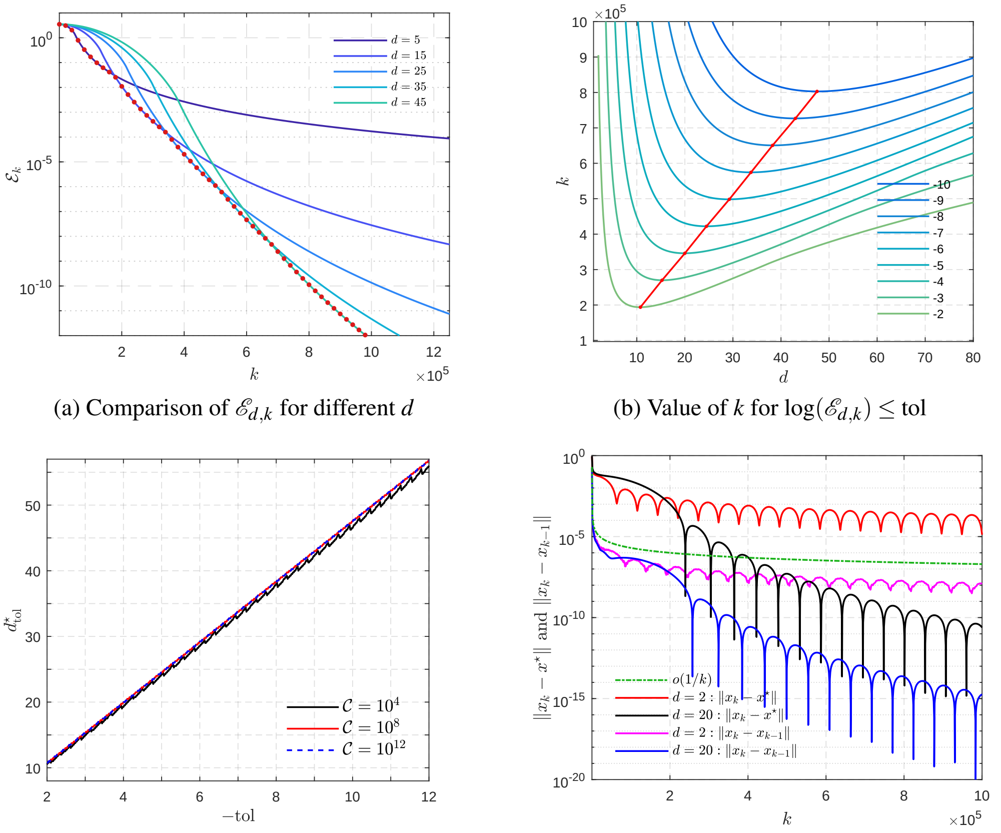
The red line in Figure 5 (b) accounts only for condition number $C=2.735\times10^8$. In Figure 5 (c), we consider three different condition numbers $C\in\{10^4,10^8,10^{12}\}$ and plot their corresponding optimal choices of d under different tol. Surprisingly, the obtained optimal choices for each C are almost same, especially for $C=10^8,10^{12}$. From these three lines, we fit the following linear function

$$d^\star_{\text{tol}}=10.75+4.6(-\text{tol}-2),$$

which can be used to compute the optimal d for a given stopping criterion on $\|x_k-x^\star\|$.

Optimal d for $\|x_k-x_{k-1}\|$ To this point, we have presented detailed analysis on the advantage of lazy-start strategy. However, the analysis is conducted via the envelope $\mathscr{E}_{d,k}$ of $\|x_k-x^\star\|$ which requires the solution $x^\star$. While in practice, only $\|x_k-x_{k-1}\|$ is available, which makes the above discussion on optimal d not practically useful. Therefore, we discuss briefly below on how to adapt the above result to $\|x_k-x_{k-1}\|$.

In Figure 5 (d) we plot both $\|x_k-x^\star\|$ and $\|x_k-x_{k-1}\|$ for the considered problem (4.1) with $d=2$ and $d=20$. The red and magenta lines are for $d=2$ while the black and blue lines are for $d=20$. It can be



(a) Comparison of $\mathscr{E}_{d,k}$ for different d

(b) Value of k for $\log(\mathscr{E}_{d,k}) \leq \text{tol}$

(c) Optimal d over tol under different C

(d) Difference between $\|x_k - x^\star\|$ and $\|x_k - x_{k-1}\|$

Figure 5: Optimal choices of d under different stopping tolerance.

observed that $\|x_k - x_{k-1}\|$ is several orders smaller than $\|x_k - x^\star\|$, which is caused by the significant decay at the beginning of $\|x_k - x_{k-1}\|$, which is due to the fact that at beginning the convergence of $\|x_k - x_{k-1}\|$ is governed by the $o(1/k)$ rate established in Theorem 3.5; see the green dot-dash line.

If we discard the beginning part of $\|x_k - x_{k-1}\|$, then the remainder can be seen as scaled $\|x_k - x^\star\|$, *i.e.* $\|x_k - x_{k-1}\| \approx \|x_k - x^\star\|/10^s$ for some $s > 0$. Therefore, if some prior about this shift could be available, then the optimal choice of d would be

$$d^\star_{\text{tol}} = 10.75 + 4.6(-\text{tol} - 2 - s).$$

For a given problem, in practice the value of s can be estimated through the following strategy:

- Run the FISTA iteration for sufficient number of iterations (*e.g.* 3×10^5 steps in Figure 5 (d)) and obtain a rough solution $\tilde{x}$ and also record the residual sequence $\|x_k - x_{k-1}\|$.
- Rerun the iteration again (*e.g.* for 10^5 steps) and output the value of $\|x_k - \tilde{x}\|$. Comparing $\|x_k - x_{k-1}\|$ and $\|x_k - \tilde{x}\|$ one can then obtain an estimation of s.

In practice, one can also simply choose $d \in [10, 80]$ which can provide consistent faster performance.

Remark 4.4.

- The discussion has been conducted through FISTA-CD, to extend the result to the case of FISTA-Mod, we may simply take $p = \frac{1}{d}$ and let $q \in]0, 1]$. As we have seen from Figure 1, the correspondence between FISTA-CD and FISTA-Mod is roughly $p = \frac{1}{d}$.
- The discussion of this section considers only the least square problem (4.1) which is very simple. However, this does not mean that lazy-start strategy will fail for more complicated problems such as ($\mathscr{P}$), see Section 7 for evidence of this.

5 Adaptive acceleration

We have discussed the advantages of the proposed FISTA-Mod scheme, particularly the lazy-start strategy. However, despite the advantage brought by lazy-start, FISTA-Mod and FISTA-CD still suffer the same drawback of FISTA-BT: the oscillation of $\Phi(x_k) - \Phi(x^\star)$ and $\|x_k - x^\star\|$ as shown in Figure 2. Therefore, in this section we discuss adaptive approaches to avoid oscillation. Note that here we only discuss adaptation to inertia, and refer to [8] for backtracking strategies for Lipschitz constant L.

The presented acceleration schemes cover two different cases: strong convexity is explicitly available, strong convexity is unknown (or 0). For the first case, the optimal parameter choices are available. While for the latter, we need to adaptively estimate the (local) strong convexity.

5.1 Strong convexity is available

For this case, we assume that F of ($\mathscr{P}$) is α-strongly convex and R is only convex, and derive the optimal setting of p, q and r for FISTA-Mod. Recall that under step-size γ, the optimal inertial parameter is $a^\star = \frac{1-\sqrt{\gamma\alpha}}{1+\sqrt{\gamma\alpha}}$. From (3.4) the limiting value of a_k, we have that for given $p, q \in]0,1]$, r should be chosen such that

$$\frac{2p + \sqrt{rp^2 + (4-r)q} - (4-r)}{2p + \sqrt{rp^2 + (4-r)q}} = \frac{1-\sqrt{\gamma\alpha}}{1+\sqrt{\gamma\alpha}}.$$

Solve the above equation we get the optimal choice of r which reads

$$\begin{aligned} r = f(\alpha,\gamma;p,q) &\overset{\text{def}}{=} 4(1-p) + 4pa^\star + (p^2-q)(1-a^\star)^2 \\ &= 4(1-p) + \frac{4p(1-\sqrt{\gamma\alpha})}{1+\sqrt{\gamma\alpha}} + \frac{4\gamma\alpha(p^2-q)}{(1+\sqrt{\gamma\alpha})^2} \le 4. \end{aligned} \tag{5.1}$$

Note that we have $f(\alpha,\gamma;p,q) = 4$ for $\alpha = 0$, and $f(\alpha,\gamma;p,q) < 4$ for $\alpha > 0$.

Based on the above result, we propose below a generalization of FISTA-Mod which is able to adapt to the strong convexity of the problem to solve.

```
Algorithm 3: Strongly convex FISTA-Mod (α-FISTA)
  Initial: let $p,q > 0$ and $\gamma \le 1/L$. For $\alpha \ge 0$, choose $r$ as $r = f(\alpha,\gamma;p,q)$. Let $t_0 \ge 1$,
           and $x_0 \in \mathbb{R}^n, x_{-1} = x_0$.
  repeat
      $t_k = \frac{p+\sqrt{q+rt_{k-1}^2}}{2}, \quad a_k = \frac{t_{k-1}-1}{t_k},$
      $y_k = x_k + a_k(x_k - x_{k-1}),$                                   (5.2)
      $x_{k+1} = \mathrm{prox}_{\gamma R}\big(y_k - \gamma\nabla F(y_k)\big).$
  until convergence;
```

Remark 5.1. Since $f(\alpha,\gamma;p,q) = 4$ when $\alpha = 0$, the above algorithm mains the $o(1/k^2)$ convergence rate for non-strongly convex case, and in general we have the following convergence property for α-FISTA,

$$\Phi(x_k) - \Phi(x^\star) \le \mathscr{C} \min\big\{\tfrac{2L}{p^2(k+1)^2}, (1-\sqrt{\gamma\alpha})^k\big\},$$

where $\mathscr{C} > 0$ is a constant.

Relation with [8] Recently, combing FISTA scheme with strong convexity was studied in [8] where the authors also propose a generalization of FISTA scheme for strongly convex problems. They consider the case that R is α_R-strongly convex and F is α_F-strongly convex, and the whole problem is then $(\alpha = \alpha_R + \alpha_F)$-strongly convex. In [8, Algorithm 1], the following update rule of t_k is considered

$$t_k = \frac{1-qt_{k-1}^2 + \sqrt{(1-qt_{k-1}^2)^2 + 4t_{k-1}^2}}{2} \quad \text{and} \quad a_k = \frac{t_{k-1}-1}{t_k}\frac{1+\gamma\alpha_R - t_k\gamma\alpha}{1-\gamma\alpha_F}, \tag{5.3}$$

where $q = \frac{\gamma\alpha}{1+\gamma\alpha_R}$. As we shall see later in Section 6, the above update rule is equivalent to Nesterov's optimal scheme [23]; see also [11] for discussions.

When $\alpha > 0$, then [8, Algorithm 1] achieves $O((1-\sqrt{q})^k)$ linear convergence rate. When $\alpha_R = 0, \alpha_F > 0$, we have $1-\sqrt{q} = 1-\sqrt{\gamma\alpha}$ which means [8, Algorithm 1] and α-FISTA achieves the same optimal rate. However, if both $\alpha_R > 0$ and $\alpha_F \geq 0$, then $1-\sqrt{\frac{\gamma\alpha}{1+\gamma\alpha_R}} > 1-\sqrt{\gamma\alpha}$, which means (5.3) achieves a sub-optimal convergence rate. As a matter of fact, if we transfer the strong convexity of R to F, that is

$$R \overset{\text{def}}{=} R - \frac{\alpha_R}{2}\|x\|^2 \quad \text{and} \quad F \overset{\text{def}}{=} F + \frac{\alpha_R}{2}\|x\|^2.$$

Then R is convex and F is α-strongly convex, and the optimal rate would be $1-\sqrt{\gamma\alpha}$. Moreover, Moreover, redefining R does not affect the complexity of computing $\text{prox}_{\gamma R}$, as it is simply quadratic perturbation of proximity operator [12, Lemma 2.6].

5.2 Strong convexity is not available

The goal of α-FISTA is to avoid the oscillatory behavior of the FISTA schemes. In the literature, an efficient way to deal with oscillation is the restarting technique developed in [24]. The basic idea of restarting is that, once the objective function value of $\Phi(x_k)$ is about to increase, the algorithm resets t_k and y_k. Doing so, the algorithm achieves an almost monotonic convergence in terms of $\Phi(x_k) - \Phi(x^\star)$, and can be significantly faster than the original scheme; see [24] or Section 7 for detailed comparisons.

The strong convexity adaptive α-FISTA (Algorithm 3) considers only the situation where the strong convexity is explicitly available, which is very often not the case in practice. Moreover, the oscillatory behavior is independent of the strong convexity. As a consequence, an adaptive scheme is needed such that the following scenarios can be covered

- Φ is *globally* strongly convex with unknown modulus α;
- Φ is *locally* strongly convex with unknown modulus α.
- Φ is neither *globally* nor *locally* strongly convex;

Estimating the strong convexity in general is time consuming. Therefore, an efficient estimation approach is also needed. To address these problems, we propose a restarting adaptive scheme (Algorithm 4), which combines the restarting technique of [24] and α-FISTA.

Algorithm 4: Restarting and Adaptive α-FISTA (**Rada-FISTA**)

```
Initial: $p, q \in ]0,1], r = 4$ and $\xi < 1, t_0 = 1, \gamma = 1/L$ and $x_0 \in \mathcal{H}, x_{-1} = x_0$.
repeat
    • Run FISTA-Mod:
        $t_k = \frac{p+\sqrt{q+rt_{k-1}^2}}{2}, \; a_k = \frac{t_{k-1}-1}{t_k},$
        $y_k = x_k + a_k(x_k - x_{k-1}),$
        $x_{k+1} = \text{prox}_{\gamma R}\big(y_k - \gamma\nabla F(y_k)\big).$
    • Restarting: if $(y_k - x_{k+1})^T(x_{k+1} - x_k) \geq 0$,
        ○ Option I: $r = \xi r$ and $y_k = x_k$;
        ○ Option II: $r = \xi r, t_k = 1$ and $y_k = x_k$.
until convergence;
```

For the rest of the paper, we shall refer to Algorithm 4 as "Rada-FISTA". Below, we provide some discussions:

- Compared to α-FISTA, the main difference of Rada-FISTA is the restarting step which is originally proposed in [24]. Such a strategy can successfully avoid the oscillatory behavior of $\Phi(x_k) - \Phi(x^\star)$.
- We provide two different options for the restarting step. In both options, we reset y_k as in [24]. Meanwhile, we also rescale the value of r by a factor ξ which is strictly smaller than 1. The purpose of rescaling is to approximate the optimal choice of r in (5.1).

- The difference between the two options is that t_k is not reset to 1 in “Option I”. Doing so, “Option I” will restart for more times than “Option II”, however it will achieve faster practical performance; see Section 7 the numerical experiments. It is worth noting that, for the restarting FISTA of [24], removing resetting t_k could also lead to an acceleration.

5.3 Greedy FISTA

We conclude this section by discussing how to further improve the performance of the restarting technique, achieving an even faster performance than Rada-FISTA and restarting FISTA [24].

The oscillation of FISTA schemes is caused by the fact that $a_k \to 1$. For the restarting scheme [24], resetting t_k to 1 forces a_k to increase from 0 again, become close enough to 1 and cause the next oscillation, then the scheme restarts. With such a loop, if we can shorten the gap between two restarts, then maybe extra acceleration could be obtained. It turns out that using constant a_k (close or equal to 1) can achieve this goal. Therefore, we propose the following greedy restarting scheme.

```
Algorithm 5: Greedy FISTA
Initial: let $\gamma \in [\frac{1}{L}, \frac{2}{L}[$ and $\xi < 1, S > 1$, choose $x_0 \in \mathbb{R}^n, x_{-1} = x_0$.
repeat
    • Run the iteration:
        $$\begin{aligned} y_k &= x_k + (x_k - x_{k-1}), \\ x_{k+1} &= \mathrm{prox}_{\gamma R}\big(y_k - \gamma \nabla F(y_k)\big). \end{aligned} \quad (5.4)$$
    • Restarting: if $(y_k - x_{k+1})^T(x_{k+1} - x_k) \geq 0$, then $y_k = x_k$;
    • Safeguard: if $\|x_{k+1} - x_k\| \geq S\|x_1 - x_0\|$, then $\gamma = \max\{\xi\gamma, \frac{1}{L}\}$;
until convergence;
```

We abuse the notation by calling the above algorithm “Greedy FISTA”, which uses constant inertial parameter $a_k \equiv 1$ for the momentum term:

- A larger step-size (than $1/L$) is chosen for γ, which can further shorten the oscillation period;
- As such a large step-size may lead to divergence, we add a “safeguard” step to ensure the convergence. This step shrinkages the value of γ when certain condition (*e.g.* $\|x_{k+1} - x_k\| \geq S\|x_1 - x_0\|$) is satisfied. Eventually we will have $\gamma = 1/L$ if the safeguard is activated a sufficient number of times.

In practice, we find that $\gamma \in [1/L, 1.3/L]$ provides faster performance than Rada-FISTA and restarting FISTA of [24]; See Section 7 for more detailed comparisons.

```
Algorithm 6: Accelerated proximal gradient (APG)
Initial: $\tau \in [0,1], \theta_0 = 1, \gamma = 1/L$ and $x_0 \in \mathscr{H}, x_{-1} = x_0$.
repeat
    Estimate the local strong convexity $\alpha_k$;
        $$\theta_k \text{ solves } \theta_k^2 = (1-\theta_k)\theta_{k-1}^2 + \tau\theta_k, \quad a_k = \frac{\theta_{k-1}(1-\theta_{k-1})}{\theta_{k-1}^2 + \theta_k},$$
        $$\begin{aligned} y_k &= x_k + a_k(x_k - x_{k-1}), \\ x_{k+1} &= \mathrm{prox}_{\gamma R}\big(y_k - \gamma \nabla F(y_k)\big). \end{aligned}$$
until convergence;
```

6 Nesterov’s accelerated scheme

In this section, we turn to Nesterov’s accelerated gradient method [23] and extend the above results to this scheme. In the book [23], Nesterov introduces several different acceleration schemes, in the following we mainly focus on the “Constant Step Scheme, III”. Applying this scheme to solve ($\mathscr{P}$), we obtain the accelerated proximal gradient method (APG) described in Algorithm 6.

When the problem ($\mathscr{P}$) is α-strongly convex, then by setting $\tau = \sqrt{\alpha/L}$ and $\theta_0 = \tau$, we have

$$\theta_k \equiv \tau \quad \text{and} \quad a_k \equiv \frac{1-\sqrt{\gamma\alpha}}{1+\sqrt{\gamma\alpha}},$$

and the iterate achieves the optimal linear convergence speed, *i.e.* $1-\sqrt{\gamma\alpha}$, as we have already discussed in the previous sections. In the rest of this section, we first build connections between the parameter updates of APG with α-FISTA, and then extend the lazy-start strategy to APG.

6.1 Connection with α-FISTA

Consider the following equation of θ parametrised by $0 \le \tau \le \sigma \le 1$, which recovers the θ_k update of APG for $\sigma = 1$,

$$\theta^2 + (\sigma\theta_{k-1}^2 - \tau)\theta - \theta_{k-1}^2 = 0. \tag{6.1}$$

The definition of a_k implies $\theta_k \in [0,1]$ for all $k \ge 1$. Therefore, the θ_k we seek from above (6.1) reads

$$\theta_k = \tfrac{-(\sigma\theta_{k-1}^2-\tau)+\sqrt{(\sigma\theta_{k-1}^2-\tau)^2+4\theta_{k-1}^2}}{2}. \tag{6.2}$$

It is then easy to verify that θ_k is convergent and $\lim_{k\to+\infty}\theta_k = \sqrt{\frac{\tau}{\sigma}}$. Back to (6.2), we have

$$\theta_k = \tfrac{2\theta_{k-1}^2}{(\sigma\theta_{k-1}^2-\tau)+\sqrt{(\sigma\theta_{k-1}^2-\tau)^2+4\theta_{k-1}^2}} = \tfrac{2}{(\sigma-\tau/\theta_{k-1}^2)+\sqrt{(\sigma-\tau/\theta_{k-1}^2)^2+4}}.$$

Letting $t_k = 1/\theta_k$ and substituting back to the above equation lead to

$$t_k = \tfrac{(\sigma-\tau t_{k-1}^2)+\sqrt{(\sigma-\tau t_{k-1}^2)^2+4t_{k-1}^2}}{2}. \tag{6.3}$$

Note that the update rule (5.3) of [8] is a special case of above equation with $\sigma = 1$ and $\tau = \frac{\gamma\alpha}{1+\gamma\alpha_R}$. Moreover,

$$t_k \to \begin{cases} +\infty : \tau = 0, \\ \sqrt{\frac{\sigma}{\tau}} : \tau \in]0,1]. \end{cases}$$

Depending on the choices of σ, τ, we have

- When $(\sigma,\tau) = (1,0)$, APG is equivalent to the original FISTA-BT scheme;
- When $(\sigma,\tau) = (1,\gamma\alpha)$, APG is equivalent to [8, Algorithm 1] for adapting to strong convexity.

Building upon the above connection, we can extend the previous result of FISTA-Mod to the case of APG.

6.2 A modified APG

Extending the FISTA-Mod and α-FISTA to the case of APG, we propose the following modified APG scheme which we name as “APG-Mod”.

Algorithm 7: A modified APG scheme (**APG-Mod**)

```
Initial: Let $\sigma \in [0,1], \gamma = 1/L$ and $\tau = \gamma\alpha\sigma, \theta_0 \in [0,1]$. Set $x_0 \in \mathscr{H}, x_{-1} = x_0$.
repeat
    $\theta_k$ solves $\theta_k^2 = (1-\sigma\theta_k)\theta_{k-1}^2 + \tau\theta_k$,
    $a_k = \frac{\theta_{k-1}(1-\theta_{k-1})}{\theta_{k-1}^2+\theta_k}$,                (6.4)
    $y_k = x_k + a_k(x_k - x_{k-1})$,
    $x_{k+1} = \mathrm{prox}_{\gamma R}\big(y_k - \gamma\nabla F(y_k)\big)$.
until convergence;
```

Non-strongly convex case For the case Φ is only convex, we have $\tau = 0$, then θ_k is the root of the equation

$$\theta^2 + \sigma\theta_{k-1}^2\theta - \theta_{k-1}^2 = 0.$$

Owing to Section 6.1, we have that APG-Mod is equivalent to FISTA-Mod with $p = \sigma$ and $q = \sigma^2$. Therefore, we have the following convergence result for APG-Mod which is an extension of Theorems 3.3 and 3.5.

Corollary 6.1. *For APG-Mod scheme Algorithm 7, let $\tau = 0$ and $\sigma \in]0,1]$, then*

- *For the objective function value,*

$$\Phi(x_k) - \Phi(x^\star) \le \frac{2L}{\sigma^2(k+1)^2}\|x_0 - x^\star\|^2.$$

  If moreover $\sigma < 1$, we have $\Phi(x_k) - \Phi(x^\star) = o(1/k^2)$.
- *Let $\sigma < 1$, then there exists an $x^\star \in \mathrm{Argmin}(\Phi)$ to which the sequence $\{x_k\}_{k\in\mathbb{N}}$ converges weakly and $\|x_k - x_{k-1}\| = o(1/k)$.*

Remark 6.2. Given the correspondence between σ of APG-Mod and p of FISTA-Mod, owing to Proposition 4.1, we obtain the lazy-start APG-Mod by choosing $\sigma \in [\frac{1}{80}, \frac{1}{10}]$.

Strongly convex case When the problem ($\mathscr{P}$) is strongly convex with modulus $\alpha > 0$, as $\tau = \gamma\alpha\sigma$, then according to Section 6.1, we have

$$\theta_k \to \sqrt{\frac{\tau}{\sigma}} = \sqrt{\gamma\alpha} \quad \text{and} \quad a_k \to \frac{1-\sqrt{\gamma\alpha}}{1+\sqrt{\gamma\alpha}},$$

which means that APG-Mod achieves the optimal convergence rate $1 - \sqrt{\gamma\alpha}$.

Remark 6.3. We can also extend the Rada-FISTA to APG, we shall forgo the details here as it is rather trivial.

7 Numerical experiments

Now we present numerical experiments on problems arising from inverse problems, signal/image processing, machine learning and computer vision to demonstrate the performance of the proposed schemes. Throughout this section, the following schemes and corresponding settings are considered:

- The original FISTA-BT scheme [6];
- The proposed FISTA-Mod (Algorithm 2) with $p = 1/20$ and $q = 1/2$, *i.e.* the lazy-start strategy;
- The restarting FISTA of [24];
- The Rada-FISTA scheme (Algorithm 4);
- The greedy FISTA (Algorithm 5) with $\gamma = 1.3/L, S = 1$ and $\xi = 0.96$.

The α-FISTA (Algorithm 3) is not considered here, except in Section 7.1, since most of the problems considered are only locally strongly convex along certain direction [16]. The corresponding MATLAB source code for reproducing the experiments is available at: https://github.com/jliang993/Faster-FISTA.

All the schemes are running with same initial point, which is $x_0 = \mathbf{1} \times 10^4$ for the least square problem and $x_0 = \mathbf{0}$ for all other problems. In terms of comparison criterion, we mainly focus on $\|x_k - x^\star\|$ where $x^\star \in \mathrm{Argmin}(\Phi)$ is a global minimizer of the optimization problem.

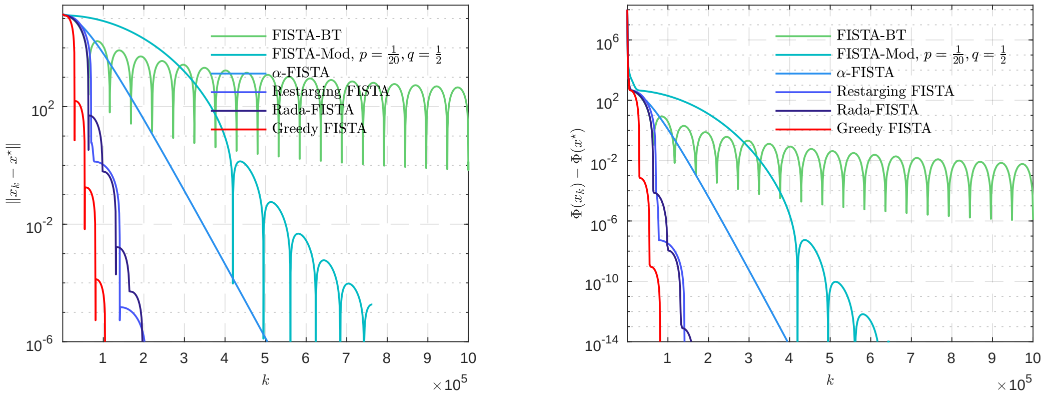
7.1 Least square (4.1) continue

First we continue with the least square estimation (4.1) discussed in Section 4, and present a comparison of different schemes in terms of both $\|x_k - x^\star\|$ and $\Phi(x_k) - \Phi(x^\star)$. Since this problem is strongly convex, the optimal scheme (*i.e.* α-FISTA) is also considered for comparison.

The obtained results are shown in Figure 6, with $\|x_k - x^\star\|$ on the left and $\Phi(x_k) - \Phi(x^\star)$ on the right. From these comparisons, we obtain the following observations:

- FISTA-BT is faster than FISTA-Mod for $k \le 3 \times 10^5$, and becomes increasing slow afterwards. This agrees with our discussion in Figure 5 that each parameter choice (of p and q, and d for FISTA-CD) is the fastest for a certain accuracy;

- α-FISTA is the only scheme whose performance is monotonic in terms of both $\|x_k - x^\star\|$ and $\Phi(x_k) - \Phi(x^\star)$. It is also faster than both FISTA-BT and FISTA-Mod;
- The three restarting adaptive schemes are the fastest among tested schemes, with Greedy FISTA being faster than the other two.



(a) $\|x_k - x^\star\|$ (b) $\Phi(x_k) - \Phi(x^\star)$

Figure 6: Comparison of different FISTA schemes for least square problem (4.1).

7.2 Linear inverse problem and regression problems

From now on, we turn to dealing with problems that are only locally strongly convex around the solution of the problem. We refer to [16] for a detailed characterization of such local neighborhoods.

Linear inverse problem Consider the following regularised least square problem

$$\min_{x \in \mathbb{R}^n} \mu R(x) + \tfrac{1}{2}\|\mathscr{K}x - f\|^2, \tag{7.1}$$

where $\mu > 0$ is trade-off parameter, R is the regularization function. The forward model of (7.1) reads

$$f = \mathscr{K}x_{\mathrm{ob}} + w, \tag{7.2}$$

where $x_{\mathrm{ob}} \in \mathbb{R}^n$ is the original object that obeys certain prior (*e.g.* sparsity and piece-wise constant), $f \in \mathbb{R}^m$ is the observed data, $\mathscr{K} : \mathbb{R}^n \to \mathbb{R}^m$ is some linear operator, and $w \in \mathbb{R}^m$ stands for noise. In the experiments, we consider R being ℓ_∞-norm and total variation [29]. Here $\mathscr{K}$ is generated from the standard Gaussian ensemble and the following setting is considered:

$$\begin{aligned} \boldsymbol{\ell_\infty}\textbf{-norm} \quad & (m,n) = (1020, 1024),\ x_{\mathrm{ob}} \text{ has 32 saturated entries;} \\ \textbf{Total variation} \quad & (m,n) = (256, 1024),\ \nabla x_{\mathrm{ob}} \text{ is 32-sparse.} \end{aligned}$$

Sparse logistic regression A sparse logistic regression problem for binary classification is also considered. Let $(h_i, l_i) \in \mathbb{R}^n \times \{\pm 1\}, i = 1, \cdots, m$ be the training set, where $h_i \in \mathbb{R}^n$ is the feature vector of each data sample, and l_i is the binary label. The formulation of sparse logistic regression reads

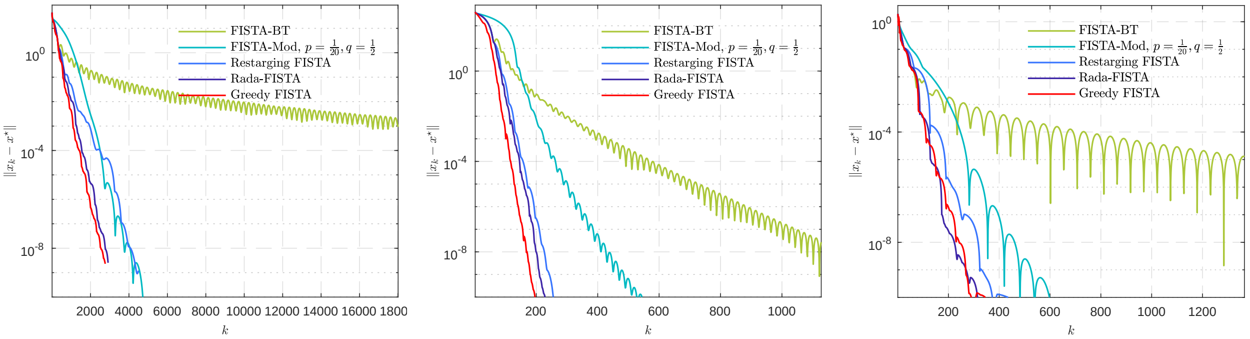
$$\min_{x \in \mathbb{R}^n} \mu\|x\|_1 + \tfrac{1}{m}\textstyle\sum_{i=1}^m \log\left(1 + e^{-l_i h_i^T x}\right). \tag{7.3}$$

The `australian` data set from LIBSVM[1] is considered.

The observations are shown in Figure 7. Although these problems are only locally strongly convex around the solution, the observations are quite close to those of least square problem discussed above:

- The lazy-start FISTA-Mod is slower than FISTA-BT at the beginning, and eventually becomes much faster, as predicted. For the ℓ_∞-norm, it is more than 10 times faster if we need the precision to be $\|x_k - x^\star\| \leq 10^{-10}$;
- The restarting adaptive schemes are the fastest ones, and the Greedy FISTA is the fastest of all.

[1] https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/



(a) ℓ_∞-norm (b) Total variation (c) Sparse logistic regression

Figure 7: Comparison of different FISTA schemes for linear inverse problems and sparse logistic regression.

7.3 Principal component pursuit

Lastly, we consider the principal component pursuit (PCP) problem [9], and apply it to decompose a video sequence into background and foreground.

Assume that a real matrix $f \in \mathbb{R}^{m\times n}$ can be written as

$$f = x_{\mathrm{l,ob}} + x_{\mathrm{s,ob}} + w,$$

where $x_{\mathrm{l,ob}}$ is low–rank, $x_{\mathrm{s,ob}}$ is sparse and w is the noise. The PCP proposed in [9] attempts to recover/approximate $(x_{\mathrm{l,ob}}, x_{\mathrm{s,ob}})$ by solving the following convex optimization problem

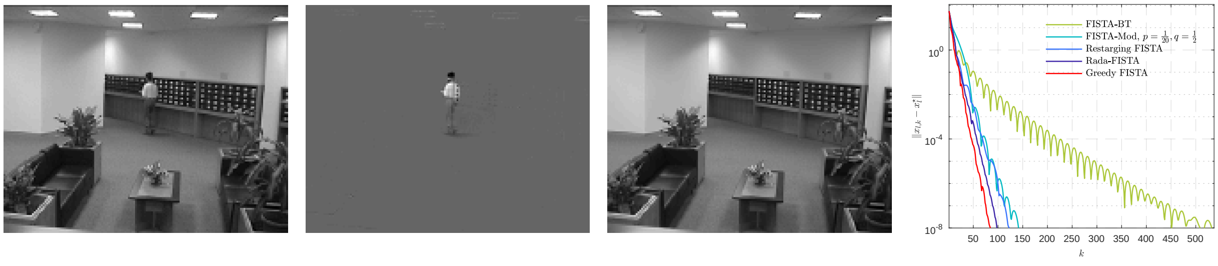
$$\min_{x_{\mathrm{l}}, x_{\mathrm{s}} \in \mathbb{R}^{m\times n}} \frac{1}{2}\|f - x_{\mathrm{l}} - x_{\mathrm{s}}\|_F^2 + \mu\|x_{\mathrm{s}}\|_1 + \nu\|x_{\mathrm{l}}\|_*, \tag{7.4}$$

where $\|\cdot\|_F$ is the Frobenius norm. Observe that for fixed x_{l}, the minimizer of (7.4) is $x_{\mathrm{s}}^\star = \mathrm{prox}_{\mu\|\cdot\|_1}(f - x_{\mathrm{l}})$. Thus, (7.4) is equivalent to

$$\min_{x_{\mathrm{l}} \in \mathbb{R}^{m\times n}} {}^{1}\big(\mu\|\cdot\|_1\big)(f - x_{\mathrm{l}}) + \nu\|x_{\mathrm{l}}\|_*, \tag{7.5}$$

where ${}^{1}\big(\mu\|\cdot\|_1\big)(f - x_{\mathrm{l}}) = \min_z \frac{1}{2}\|f - x_{\mathrm{l}} - z\|_F^2 + \mu\|z\|_1$ is the Moreau Envelope of $\mu\|\cdot\|_1$ of index 1, and hence has 1-Lipschitz continuous gradient.

We use the video sequence from [13] and the obtained result is demonstrated in Figure 8. Again, we obtain consistent observations with the above examples. Moreover, the performance of lazy-start FISTA-Mod is very close to the restarting adaptive schemes.



(a) Original frame (b) Sparse component (c) Low-rank component (d) Performance comparison

Figure 8: Comparison of different FISTA schemes for principal component pursuit problem. (a) original frame; (b) foreground; (c) background; (d) performance comparison.

In all these experiments we find that the proposed variants can perform better than the original versions but restarting are consistently faster. Greedy FISTA was the best in every example shown.

8 Conclusions

We proposed a simple modification to the original FISTA-BT scheme, which allows us to prove the convergence of the sequence generated by the modified scheme. We also proposed a lazy-start strategy which can greatly improve the practical performance of FISTA schemes. Several adaptive schemes were also developed, which can adaptively adjust to the (local) properties of the problem to solve. The performances of the proposed schemes were verified on various problems arising from inverse problems, data science and computer vision.

Acknowledgement

We would like to thank Dr. Robert Tovey for helpful discussions and comments of the paper. JL acknowledges support from the Leverhulme Trust and Newton Trust. CBS acknowledges support from the Leverhulme Trust project on Breaking the Non-Convexity Barrier, and on Unveiling the Invisible, the Philip Leverhulme Prize, the EPSRC grant No. EP/S026045/1, EPSRC grant No. EP/M00483X/1, and EPSRC Centre No. EP/N014588/1, the European Union Horizon 2020 research and innovation programmes under the Marie Skłodowska-Curie grant agreement No. 691070 CHiPS and the Marie Skłodowska-Curie grant agreement No 777826, the Cantab Capital Institute for the Mathematics of Information, and the Alan Turing Institute.

References

[1] H. Attouch, A. Cabot, Z. Chbani, and H. Riahi. Inertial forward–backward algorithms with perturbations: Application to tikhonov regularization. *Journal of Optimization Theory and Applications*, 179(1):1–36, 2018.

[2] H. Attouch and J. Peypouquet. The rate of convergence of Nesterov's accelerated forward-backward method is actually faster than $o(1/k^2)$. *SIAM Journal on Optimization*, 26(3):1824–1834, 2016.

[3] H. Attouch, J. Peypouquet, and P. Redont. On the fast convergence of an inertial gradient-like dynamics with vanishing viscosity. Technical Report arXiv:1507.04782, 2015.

[4] J. B. Baillon and G. Haddad. Quelques propriétés des opérateurs angle-bornés etn-cycliquement monotones. *Israel Journal of Mathematics*, 26(2):137–150, 1977.

[5] A. Beck and M. Teboulle. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE Transactions on Image Processing*, 18(11):2419–2434, 2009.

[6] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[7] D. P. Bertsekas. *Nonlinear programming*. Athena scientific Belmont, 1999.

[8] L. Calatroni and A. Chambolle. Backtracking strategies for accelerated descent methods with smooth composite objectives. *arXiv preprint arXiv:1709.09004*, 2017.

[9] E. J. Candès, X. Li, Y. Ma, and J. Wright. Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11, 2011.

[10] A. Chambolle and C. Dossal. On the convergence of the iterates of the "fast iterative shrinkage/thresholding algorithm". *Journal of Optimization Theory and Applications*, 166(3):968–982, 2015.

[11] A. Chambolle and T. Pock. An introduction to continuous optimization for imaging. *Acta Numerica*, 25:161–319, 2016.

[12] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling & Simulation*, 4(4):1168–1200, 2005.

[13] L. Li, W. Huang, I. Y. Gu, and Q. Tian. Statistical modeling of complex backgrounds for foreground object detection. *IEEE Transactions on Image Processing*, 13(11):1459–1472, 2004.

[14] J. Liang. *Convergence rates of first-order operator splitting methods*. PhD thesis, Normandie Université; GREYC CNRS UMR 6072, 2016.

[15] J. Liang, J. Fadili, and G. Peyré. Convergence rates with inexact non-expansive operators. *Mathematical Programming*, 159(1-2):403–434, 2016.

[16] J. Liang, J. Fadili, and G. Peyré. Activity identification and local linear convergence of Forward–Backward-type methods. *SIAM Journal on Optimization*, 27(1):408–437, 2017.

[17] P. L. Lions and B. Mercier. Splitting algorithms for the sum of two nonlinear operators. *SIAM Journal on Numerical Analysis*, 16(6):964–979, 1979.

[18] D. A. Lorenz and T. Pock. An inertial forward-backward algorithm for monotone inclusions. *Journal of Mathematical Imaging and Vision*, 51(2):311–325, 2015.

[19] C. Molinari, J. Liang, and J. Fadili. Convergence rates of forward–douglas–rachford splitting method. *arXiv preprint arXiv:1801.01088*, 2018.

[20] A. Moudafi and M. Oliny. Convergence of a splitting inertial proximal method for monotone operators. *Journal of Computational and Applied Mathematics*, 155(2):447–454, 2003.

[21] A. S. Nemirovsky and D. B. Yudin. Problem complexity and method efficiency in optimization. 1983.

[22] Y. Nesterov. A method for solving the convex programming problem with convergence rate $O(1/k^2)$. *Dokl. Akad. Nauk SSSR*, 269(3):543–547, 1983.

[23] Y. Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer, 2004.

[24] B. O'Donoghue and E. Candes. Adaptive restart for accelerated gradient schemes. *Foundations of computational mathematics*, 15(3):715–732, 2015.

[25] Z. Opial. Weak convergence of the sequence of successive approximations for nonexpansive mappings. *Bulletin of the American Mathematical Society*, 73(4):591–597, 1967.

[26] B. T. Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

[27] B. T. Polyak. *Introduction to optimization*. Optimization Software, 1987.

[28] R. T. Rockafellar. *Convex analysis*, volume 28. Princeton university press, 1997.

[29] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1):259–268, 1992.

[30] W. Su, S. Boyd, and E. Candes. A differential equation for modeling nesterov's accelerated gradient method: Theory and insights. In *Advances in Neural Information Processing Systems*, pages 2510–2518, 2014.