FISTA: achieving a rate of convergence proportional to k^{-3} for small/medium values of k

Gustavo Silva and Paul Rodriguez
Electrical Department
Pontificia Universidad Católica del Perú
Lima, Peru
Email: [gustavo.silva, prodrig]@pucp.edu.pe

***Abstract*—The fast iterative shrinkage-thresholding algorithm (FISTA) is a widely used procedure for minimizing the sum of two convex functions, such that one has a L-Lipschitz continuous gradient and the other is possible nonsmooth.**

While FISTA's theoretical rate of convergence (RoC) is proportional to $\frac{1}{\alpha_k t_k^2}$, and it is related to (i) its extragradient rule / inertial sequence, which depends on sequence t_k, and (ii) the step-size α_k, which estimates L, its worst-case complexity results in $\mathcal{O}(k^{-2})$ since, originally, (i) by construction $t_k \geq \frac{k+1}{2}$, and (ii) the condition $\alpha_k \geq \alpha_{k+1}$ was imposed. Attempts to improve FISTA's RoC include alternative inertial sequences, and intertwining the selection of the inertial sequence and the step-size.

In this paper, we show that if a bounded and non-decreasing step-size sequence ($\alpha_k \leq \alpha_{k+1}$, decoupled from the inertial sequence) can be generated via some adaptive scheme, then FISTA can achieve a RoC proportional to k^{-3} for the indexes where the step-size exhibits an approximate linear growth, with the default $\mathcal{O}(k^{-2})$ behavior when the step-size's bound is reached. Furthermore, such exceptional step-size sequence can be easily generated, and it indeed boots FISTA's practical performance.

***Index Terms*—FISTA, step-size, convolutional sparse representations.**

I. Introduction

The optimization of

$$\min_{\mathbf{x}\in\mathbb{R}^N} F(\mathbf{x}) := f(\mathbf{x}) + g(\mathbf{x}), \tag{1}$$

where $f,\, g : \mathbb{R}^N \mapsto \mathbb{R}$ are both convex functions, gradient ∇f is L-Lipschitz continuous: $\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2 \leq L(f)\|\mathbf{x} - \mathbf{y}\|_2$, and g's proximal operator,

$$\operatorname*{prox}_g(\mathbf{y}) = \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{x} - \mathbf{y}\| + g(\mathbf{x}), \tag{2}$$

has a computationally simple or affordable solution even if $g(\cdot)$ is nonsmooth, has several applications in inverse problems related to signal/image processing and machine learning.

There exists several numerical algorithms[1] to minimize (1), being FISTA [5] a widely used choice (specially if $g(\mathbf{x}) = \lambda \cdot \|\mathbf{x}\|_1$), due to its simplicity and theoretical $\mathcal{O}(k^{-2})$ rate of convergence. In general, FISTA generates the iterates

$$\mathbf{x}_k = \operatorname*{prox}_g(\mathbf{y}_k - \alpha_k \nabla f(\mathbf{y}_k)) \tag{3}$$

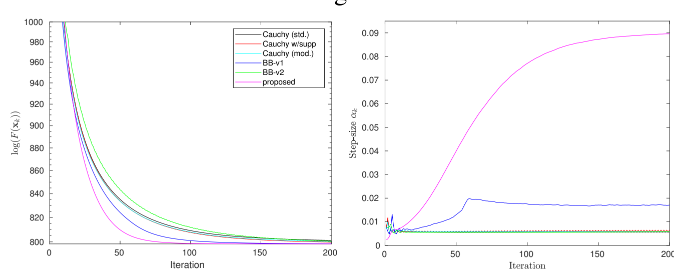
$$\mathbf{y}_k = \mathbf{x}_k + \gamma_k(\mathbf{x}_k - \mathbf{x}_{k-1}) \tag{4}$$

for $k \geq 1$, where $\alpha_k \in [0, \frac{1}{L}]$ is the step-size and γ_k, the inertial sequence, satisfies

$$\gamma_k = \frac{t_k - 1}{t_{k+1}}, \qquad t_{k+1}^2 - t_{k+1} \leq t_k^2 \;\; \forall k \geq 1. \tag{5a,5b}$$

While FISTA's theoretical $\mathcal{O}(k^{-2})$ rate of convergence (RoC) is related to the extragradient rule[2] and proper choice of the inertial sequence (see Section II-B), the selection of the step-size α_k i.e. the estimation of L, the Lipschitz constant, also impacts FISTA's practical performance.

To illustrate the above statement we set $F(\mathbf{x})$ as (23), i.e. the convolutional sparse coding problem (CSC; among many others, see [6], [7]), and solve it with the FISTA algorithm considering six different methods[3] to adaptively select the step-size α_k: (i) "Cauchy (std.)", (ii) "Cauchy w/supp" and (iii) "Cauchy (mod.)" represent three variants of the Cauchy step-size [8] (described in Section II-A3); (iv) "BB-v1" and (v) "BB-v2" represent two variants of the well-known Barzilai-Borwein step-size [9] (see also Section II-A2); and (vi) "proposed" which represents the case when the step-size sequence is bounded and non-decreasing.



(a) (b)

Fig. 1: Evolution of cost function (23) in logarithmic scale (a) and corresponding step-size sequence (b) for different adaptive schemes for selecting FISTA's step-size[3].

For the above mentioned choices of the step-size α_k, in Fig. 1a and 1b we depict the evolution of the cost function (23), in logarithmic scale, and corresponding step-size's values versus iteration respectively. This example highlights the impact of sequence $\{\alpha_k\}$ over FISTA's RoC; moreover, we hypothesize

[1] E.g. Douglas-Rachford splitting [1], forward-backward splitting [2], ADMM [3], etc. Also, (1) is generally referred to as the *composite unconstrained convex programming* problem; see [4, Section 5.2] for several variants, which also include FISTA.

[2] ∇f is evaluated at a linear combination of the past two iterates, see (3).

[3] For this experiment, for all cases, we use the original inertial sequence (8a) proposed in [5] and an independent selection of the step-size α_k; further details are given Section IV-B.

978-9-0827-9703-9/19/$31.00 ©2019 IEEE

that a bounded and non-decreasing sequence with a large limit value implies (i) a faster convergence and (ii) a reduction of the local oscillatory behavior of FISTA (originally observed in [10]; see also [11] for a formal description).

Furthermore, as proven in Section III, if we assume that $\{\alpha_k\}$ exhibits an approximate linear growth $\forall k \in [1, \kappa]$, then FISTA achieves a RoC proportional to k^{-3} for such interval, as summarized in (21). Finally, in Section IV, we show, via computational experiments, that all the above mentioned properties of $\{\alpha_k\}$ can be met by underestimating a local approximation of the Lipschitz constant (see (24a)), which is dependent on the current solution's support.

II. Previous related work

A. Step-sizes for the Gradient method

On what follows, based on [12], we succinctly described some alternatives (for a full list, see [12]) on how to select the step-size for the Gradient method, where problem (1) is simplify by taking $g(\mathbf{x}) = 0$, and the next iterate is defined by $\mathbf{x}_k = \mathbf{x}_k - \alpha_k \mathbf{g}_k$, where $\mathbf{g}_k = \nabla F(\mathbf{x}_k)$.

1) Exact/inexact line search: The exact line search defines $\alpha_k = \arg\min_\alpha F(\mathbf{x}_k - \alpha \mathbf{g}_k)$, whereas for the inexact case α_k can be computed by some line search conditions, such as Goldstein, Wolfe or Armijo conditions (see [13]).

However, usually (see for instance [14]), a simple line search scheme undermines FISTA's performance.

2) Barzilai-Borwein method: [9] proposed to use the information in the previous iteration to estimate α_k.

Considering $\mathbf{z}_k = \mathbf{x}_k - \mathbf{x}_{k-1}$ and $\mathbf{q}_k = \nabla F(\mathbf{x}_k) - \nabla F(\mathbf{x}_{k-1})$, [9] proposed two variants, henceforth labeled BB-v1 (6a) and BB-v2 (6a), where $\langle \cdot, \cdot \rangle$ represents inner product, which can be shown to exhibit R-superlinear convergence for the Gradient method; it has also been evaluated [15, Section 3.B] in the context of compressed sensing.

$$\alpha_k = \frac{\langle \mathbf{z}_k, \mathbf{q}_k \rangle}{\|\mathbf{q}_k\|_2^2}, \qquad \alpha_k = \frac{\|\mathbf{z}_k\|_2^2}{\langle \mathbf{z}_k, \mathbf{q}_k \rangle}. \tag{6a,6b}$$

3) Cauchy step and variants: While it is well-known that the standard Cauchy step (7a) can be inefficient (i.e. produces a slow convergence, as can be observed in the particular example associated with Fig. 1) and that it is always too long [12, Section 3], we emphasize that there are successful variants: (i) in the context of sparse representations [16] proposed to use (7b), where $\mathbf{s}_k = I_{[|\mathbf{x}_k|>0]}$, $I_{[\text{COND}]}$ represents the Indicator function[4] and $\odot$ represents element-wise product, (ii) in the context of convex quadratic optimization, [8] proposed (7c) and proved[5] that it asymptotically converges to (7a). Henceforth, (7a), (7b) and (7c) are labeled "Cauchy (std.)", "Cauchy w/supp" and "Cauchy (mod.)" respectively.

$$\alpha_k = \frac{\|\mathbf{g}_k\|_2^2}{\|\Phi \mathbf{g}_k\|_2^2}, \quad \alpha_k = \frac{\|\mathbf{s}_k \odot \mathbf{g}_k\|_2^2}{\|\Phi(\mathbf{s}_k \odot \mathbf{g}_k)\|_2^2}, \quad \alpha_k^2 = \frac{\|\mathbf{g}_k\|_2^2}{\|\Phi^T \Phi \mathbf{g}_k\|_2^2}.$$
(7a,7b,7c)

[4]Equal to 1 if "COND" is true, 0 otherwise

[5][8] also noticed that BB-v2 or (6b) is the Cauchy step evaluated at the previous iteration k-1.

B. Inertial sequences for FISTA

Simple choices for the inertial sequence $\{\gamma_k\}$, considering $t_1 = 1$, can be generated using (8)[6]: Originally, [5] proposed to use (8a)[7], while more recently, among others, [20], [21], [22] used (8b) for several values of $b \geq 2$ (being $b = 2$ common practice). Furthermore, [23] proposed a generalization of (8b), resulting in (8c), with $b = 2$ and $a \in [50,\, 80]$ as default values.

$$t_k = \frac{1 + \sqrt{1 + 4 * t_{k-1}^2}}{2}, \qquad t_k = \frac{k-1+b}{b},\ b \geq 2,$$
$$t_k = \frac{k-1+a}{b},\ b \geq 2, a \geq b{-}1. \tag{8a,8b,8c}$$

C. Inertial sequence and step-size: Intertwined selection

As mentioned in Section I, FISTA is one particular variant among several accelerated methods (see [4, Section 5.2]) to solve problem (1), all with nearly identical theoretical RoC, proportional to $\frac{1}{\alpha_k t_k^2}$, where α_k and t_k are related to the step-size and inertial sequences.

[24] noticed that a simple backtracking / line search will cause the above mentioned error bound to rise unnecessarily, and proposed to adapt both α_k and t_k accordingly. Furthermore [25, Proposition 1] proved that FISTA's convergence is preserved if

$$\alpha_k t_k^2 \geq \alpha_{k+1} t_{k+1} (t_{k+1} - 1). \tag{9}$$

Several works have exploited (9) or variants. To further improve FISTA's performance [4, Section 5.3] also proposed to increase α_k "when conditions permit"; several numerical examples in [4] provided computational evidence for the effectiveness of such approach. More recently, [26] also intertwined the selection α_k and t_k, via the BB-v1 (6a) step-size along with a line search, as to adaptively choose a step-size as large as possible. By considering the general case of the Forward-Backward splitting method, [11, Theorem 2.3] proved and exploited an alternative relationship for α_k and t_k. Based on a generalization of (9), [27] proposed a new FISTA-like method along with a robust step size search.

III. Achieving a rate of convergence proportional to k^{-3} for small/medium values of k

A. FISTA's rate of convergence: key results

FISTA's convergence analysis is thoroughly detailed in [5]. On what follows we highlight its key results, which will be also used as a starting point for our new convergence analysis (see Section III-B).

We start by reproducing FISTA's approximation model [5, Section 2.3] for $F(\mathbf{x})$ (see (1)), summarized in (10)

$$Q_\alpha(\mathbf{x}, \mathbf{y}) = f(\mathbf{y}) + \langle \mathbf{x} - \mathbf{y}, \nabla f(\mathbf{y}) \rangle + \frac{1}{2\alpha} \|\mathbf{x} - \mathbf{y}\|_2^2 + g(\mathbf{x}). \tag{10}$$

Clearly $Q_\alpha(\cdot)$ admits a unique minimizer given by $p_\alpha(\mathbf{y}) = \arg\min_\mathbf{x} Q_\alpha(\mathbf{x}, \mathbf{y})$, which can be expressed as

$$p_\alpha(\mathbf{y}) \quad = \quad \arg\min_\mathbf{x} \frac{1}{2\alpha} \|\mathbf{x} - (\mathbf{y} - \alpha \nabla f(\mathbf{y}))\|_2^2 + g(\mathbf{x}). \tag{11}$$

[6]Other choices [17], [18] include ad-hoc rules or many more parameters.

[7]Same as Nesterov's acceleration scheme [19].

As mentioned in [5, Section 2.4], one key results is needed to prove FISTA's convergence rate, namely Lemma 2.3, reproduced here[8] as Lemma III.1.

Lemma III.1. *Let* $\mathbf{y} \in \mathbb{R}^N$, $\alpha > 0$ *s.t.* $F(\mathbf{x}) \leq Q_\alpha(\mathbf{x}, \mathbf{y})$, *then*

$$F(\mathbf{x}) - F(p_\alpha(\mathbf{y})) \;\geq\; \frac{1}{2\alpha}(\|\mathbf{x} - p_\alpha(\mathbf{y})\|_2^2 - \|\mathbf{x} - \mathbf{y}\|_2^2). \quad (12)$$

By applying Lemma III.1 at the points $\mathbf{x} = \mathbf{x}_k$, $\mathbf{y} = \mathbf{y}_{k+1}$ and $\mathbf{x} = \mathbf{x}^*, \mathbf{y} = \mathbf{y}_{k+1}$, along with $\alpha = \alpha_{k+1}$ and $\mathbf{x}_{k+1} = p_{\alpha_{k+1}}(\mathbf{y}_{k+1})$, which is a consequence of (3), then by adequately combining the resulting inequalities, (see proof of Lemma 4.1 in [5]), we can get

$$2\alpha_{k+1}\left(\tau_{k+1}v_k - t_{k+1}^2 v_{k+1}\right) \;\geq\; \|\mathbf{u}_{k+1}\|_2^2 - \|\mathbf{u}_k\|_2^2, \quad (13)$$

where t_k is the sequence (see (5a) and (5b)) used to generate the inertial sequence γ_k in (4), $\tau_{k+1} = t_{k+1}(t_{k+1} - 1)$, $v_k = F(\mathbf{x}_k) - F(\mathbf{x}^*)$ and $\mathbf{u}_k = t_k\mathbf{x}_k - (t_k - 1)\mathbf{x}_{k-1} - \mathbf{x}^*$.

From this point onward, in order to get the well-known FISTA's RoC, i.e. $\mathcal{O}(k^{-2})$, [5] used the fact that its chosen inertial sequence satisfies equality in (5b), i.e. $t_k^2 = t_{k+1}(t_{k+1} - 1)$, and that by construction, it always chooses a step-size s.t. $\frac{1}{\varsigma L(f)} \leq \alpha_{k+1} \leq \alpha_k$, with $\varsigma \geq 1$. By combining theses facts into (13), inequality (14) follows,

$$2\alpha_k t_k^2 v_k - 2\alpha_{k+1}t_{k+1}^2 v_{k+1} \;\geq\; \|\mathbf{u}_{k+1}\|_2^2 - \|\mathbf{u}_k\|_2^2, \quad (14)$$

from which it is easy to check (15), since $t_k \geq \frac{k+1}{2}$ is a consequence of using equality in (5b).

$$F(\mathbf{x}_k) - F(\mathbf{x}^*) \leq \frac{2\|\mathbf{x}_0 - \mathbf{x}^*\|_2^2}{\alpha_k(k+1)^2} \leq \frac{2\varsigma L(f)\|\mathbf{x}_0 - \mathbf{x}^*\|_2^2}{(k+1)^2}. \quad (15)$$

B. New convergence analysis

Our convergence analysis is motivated by the example presented in Section I (further explained in Section IV-C), which highlights the practical impact of the step-size sequence $\{\alpha_k\}$ over FISTA's RoC. Furthermore, similar results have also been observed when the selection of α_k is intertwined the inertial sequence (see Section II-C and references therein).

On what follows we start by assuming that the step-size sequence $\{\alpha_k\}$ is bounded and non-decreasing, i.e. $\alpha_{k+1} \geq \alpha_k$. Furthermore, in order to ease our analysis, we also assume that for $k \in [1,\, \kappa]$, the step-size sequence is linear, i.e. $\alpha_k = \alpha_0 + k\mu$, where $\alpha_0 > 0$ and $\mu > 0$.

Our convergence analysis diverges from FISTA's original one from (13) onward: instead of considering $t_k^2 = t_{k+1}(t_{k+1} - 1)$, which results in (8a), we consider

$$t_k^2 = d_k + t_{k+1}(t_{k+1} - 1) \quad (16)$$

which is the case for either (8b) or (8c) for some $d_k > 0$. Furthermore, by assuming that $k + 1 < \kappa$, a simple algebraic manipulation leads to (17).

$$C_k = \frac{2}{k}\sum_{n=1}^{k} \alpha_n, \quad \alpha_{k+1} = C_k - \alpha_0 = C_{k+1} - \alpha_1. \quad (17a, 17b)$$

[8] We note that there are small difference in notation w.r.t. [5]: we use α instead of L, and for Lemma III.1 we use the Pythagoras relation $\|\mathbf{b} - \mathbf{a}\|_2^2 + 2\langle \mathbf{b} - \mathbf{a}, \mathbf{a} - \mathbf{c}\rangle = \|\mathbf{b} - \mathbf{c}\|_2^2 - \|\mathbf{a} - \mathbf{c}\|_2^2$ to summarized (12).

By replacing (16) in (13) and adequately expressing α_{k+1} as a function of C_k or C_{k+1} (see (17)), we get

$$2C_k t_k^2 v_k - 2C_{k+1}t_{k+1}^2 v_{k+1} - \beta_k \;\geq\; \|\mathbf{u}_{k+1}\|_2^2 - \|\mathbf{u}_k\|_2^2, (18)$$

where $\beta_k = 2(\alpha_{k+1}d_k v_k + \alpha_0 t_k^2 v_k - \alpha_1 t_{k+1}^2 v_{k+1})$.

If we now assume that $\forall k \in [1,\, \kappa]$ (i) $\beta_k \geq 0$ or (ii) $\beta_k < 0$ but it is small enough so it does not affect inequality (18), then

$$2C_k t_k^2 v_k - 2C_{k+1}t_{k+1}^2 v_{k+1} \;\geq\; \|\mathbf{u}_{k+1}\|_2^2 - \|\mathbf{u}_k\|_2^2, \quad (19)$$

holds $\forall k \in [1,\, \kappa]$. By using the same arguments as in [5], then

$$F(\mathbf{x}_k) - F(\mathbf{x}^*) \leq \frac{\|\mathbf{x}_0 - \mathbf{x}^*\|_2^2}{2C_k t_k^2} \quad \forall k \in [1,\, \kappa], \quad (20)$$

whereas for $k > \kappa$, the bound given by (15), with α_k replaced by the bound on the step-size sequence, will hold.

Finally, since it is trivial to show that $C_k \geq \mu \cdot (k+1)$ and that for either (8b) or (8c) $t_k \geq \frac{k+1}{b}$ holds, then

$$F(\mathbf{x}_k) - F(\mathbf{x}^*) \leq \frac{b^2\|\mathbf{x}_0 - \mathbf{x}^*\|_2^2}{2\mu(k+1)^3} \quad \forall k \in [1,\, \kappa]. \quad (21)$$

C. Motivation for a non-decreasing step-size sequence

In the context of the ℓ_0 regularized optimization (ℓ_0-RO) problem, i.e. $g(\mathbf{x}) = \lambda\|\mathbf{x}\|_0$ in (1): $\min_{\mathbf{x}\in\mathbb{R}^N} F(\mathbf{x}) = \|A\mathbf{x} - \mathbf{b}\|_2^2 + \lambda \cdot \|\mathbf{x}\|_0$, where where $A \in \mathbb{R}^{M\times N}$ is called a dictionary with N atoms, $\mathbf{x} \in \mathbb{R}^N$, $\mathbf{b} \in \mathbb{R}^M$ and $\|\mathbf{x}\|_0$ is the semi-norm that counts the number of non-zero elements in $\mathbf{x}$, it has recently been proved [28, Lemma 1] that the ℓ_0-RO problem is equivalent to (22),

$$\min_{\mathbf{z}\in\mathbb{R}^L} F_S(\mathbf{z}) = f_s(\mathbf{z}) + g(\mathbf{z}) = \|A_S\mathbf{z} - \mathbf{b}\|_2^2 + \lambda \cdot \|\mathbf{z}\|_0, \quad (22)$$

where $A_S \in \mathbb{R}^{M\times L}$, $L < N$, considers a properly chosen reduced number of atoms.

The equivalence between the original ℓ_0-RO problem and (22) also implies that the actual Lipschitz constant for its quadratic term is $L(f_s)$ rather than $L(f)$. Clearly the support of $\mathbf{x}^*$ is unknown in advanced; however, if an iterative solution for the ℓ_0-RO problem complies with $\mathrm{supp}(\mathbf{x}_{k+1}) \subseteq \mathrm{supp}(\mathbf{x}_k)$, $\forall k \geq 0$ (this is one of the results of [28, Lemma 1]), then the Lipschitz constant varies when the support of the current solution effectively shrinks.

To the best of our knowledge, there is no equivalent result to [28, Lemma 1] when $g(\mathbf{x}) = \lambda\|\mathbf{x}\|_1$. However, in the context of an intertwined selection of the inertial sequence and step-size (see Section II-C), some works (e.g. [26, Fig. 2a], [27, Fig. 1b], etc.) have observed that a step-size sequence, with a non-decreasing behavior for a limited interval is related to a better performance in FISTA.

The bound summarized in (21) indeed implies that the Lipschitz constant should change up to a given iteration to then settle. In our experimental results we provide computational evidence that such behavior can be exploited to attain (21).

D. Implications of (21)

Several implications can be easily deduced from (21): (i) a larger slope (μ) implies a faster convergence; (ii) for very small values of k we expect a slower RoC w.r.t. the case when a constant step-size (or backtracking) is used; (iii) if μ is too large, at some point the assumption about the sign of β_k or it being "small enough" will break, and thus FISTA's performance will be undermined; (iv) (8b) / (8c) have better performance than (8a) since for the formers $t_k^2 > t_{k+1}(t_{k+1}-1)$ and thus, with higher probability, assumptions about β_k are true.

In Section IV, (see also Fig. 1 and 2), we provide computational evidence for the above mentioned implications.

IV. Computational results

A. Experimental setup

Our experiments were carried out on an Intel i7-6820HK (2.70 GHz, 8GB Cache, 64GB RAM) based laptop with a nvidia GTX1070 (8GB memory) GPU card; our publicly available GPU-enabled Matlab code [29] can be used to reproduce our computational results.

Due to space constrains, we focus our experiments on the convolutional sparse coding (CSC) problem (see Section IV-B), where we consider five test images ("Lena", "Barbara", "Kiel" and "Bridge", each 512×512 pixel, and "Man", 1024×1024 pixel). Other problems, such ℓ_0 regularized optimization, Wavelet-based inpainting (noiseless) and and ℓ_0-CSC can also be solved via our companion Matlab code [29].

B. Convolutional Sparse Coding (CSC)

Convolutional sparse representation (CSR) [30], [31] models an entire signal or image as a sum over a set of convolutions of coefficient maps, of the same size as the signal or image, with their corresponding dictionary filters. Given a set of separable or non-separable[9] dictionary filters, the most widely used formulation of the convolutional sparse coding (CSC) problem is Convolutional BPDN (CBPDN) [7], defined as

$$\arg\min_{\{\mathbf{u}_k\}} \; \frac{1}{2}\left\|\sum_{k=1}^{K} H_k * \mathbf{u}_k - \mathbf{b}\right\|_2^2 + \lambda \sum_{k=1}^{K} \|\mathbf{u}_k\|_1, \qquad (23)$$

where $\{H_k\}$ represents a set of K, $L_1 \times L_2$ filters, $\{\mathbf{u}_k\}$ is the corresponding set of coefficient maps (each with $N_1 \times N_2$ samples), $\mathbf{b}$ is the $N_1 \times N_2$ input image, and λ is the regularization parameter.

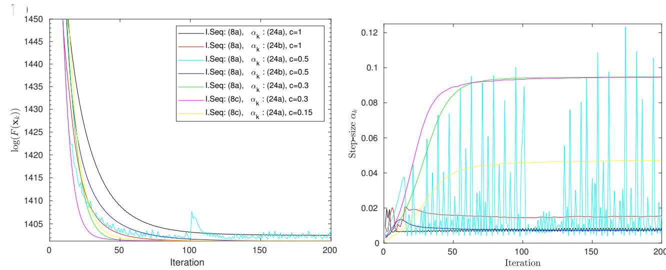
For the experiments presented below, we highlight that the test images were not used in the dictionary learning stage, and that $\mathbf{b}$, in (23), is the original image corrupted with uncorrelated additive Gaussian noise, i.e. $\mathbf{b} = \mathbf{b}^* + \boldsymbol{\eta}$.

C. Non-decreasing and bounded step-size sequence: Generation and assessment

In general, the Cauchy step-size is too long (see Section II-A3). However, we have noticed that (7b) multiplied by a small, manually selected constant[10] $0 < c \leq 1$, i.e (24a)

$$\alpha_k = c\frac{\|\mathbf{s}_k \odot \mathbf{g}_k\|_2^2}{\|\Phi(\mathbf{s}_k \odot \mathbf{g}_k)\|_2^2}, \qquad \alpha_k = c\frac{\langle \mathbf{z}_k, \mathbf{q}_k\rangle}{\|\mathbf{q}_k\|_2^2}, \qquad (24a, 24b)$$

where all other variables defined in Section II-A with $\Phi\mathbf{u} = \sum_{k=1}^{K} H_k * \mathbf{u}_k$, can indeed generate a bounded and non-decreasing step-size sequence as shown in Fig. 1 (labeled "proposed") and in Fig. 2 where (23) is solved for a noisy ($\sigma_{\boldsymbol{\eta}}^2 = 0.01$) "Kiel" and "Barbara" respectively, for different step-size's choices, inertial sequences (I.Seq) and values of c (for Fig. 1, $c = 0.3$; for Fig. 2, values are listed in its legend).



(a) (b)

Fig. 2: Evolution of cost function (23) in logarithmic scale (a) and corresponding step-size sequence (b) for different I.Seq and several alternatives values of c in (24a) and (24b).

Furthermore, as it was claimed in Section III-D, if the slope of step-size sequence, in its linear region, is too large, then the RoC will negatively suffer: this can be observed in Fig. 2 when I.Seq is generated by (8a) and the step-size via (24a) with $c = \{0.5,\, 1.0\}$ (cyan and black lines). When constant c is correctly chosen (magenta, green and yellow lines), the best RoC will be associated to the step-size sequence with the largest slope; we also note that the RoC associated with the I.Seq generated by (8c) is better (compare the magenta and green lines) than that generated by (8a).

Finally we stress that other adaptive choices for α_k, such (24b) with $c < 1$, are counter-productive, as can be observed for the red and blue lines in Fig. 2.

V. Conclusions

When FISTA is used to optimize a problem where its solution is sparse, we have provided experimental evidence showing that it is feasible to adaptively compute a bounded and non-decreasing step-size sequence, i.e. $\alpha_k \leq \alpha_{k+1}$, which is dependent on the current solution's support and is decoupled from FISTA's inertial sequence. Furthermore, if we assume that $\{\alpha_k\}$ exhibits an approximate linear growth $\forall k \in [1,\, \kappa]$, then we can prove that FISTA achieves a rate of convergence proportional to k^{-3} for such interval, effectively boosting FISTA's performance when compare to the de-facto case where $\alpha_k \geq \alpha_{k+1}$ or for other educated choices of α_k.

For the CSC problem, our experimental results shown that, to attain the same cost functional value, the proposed selection of α_k can roughly reduce FISTA's global number of iterations by half when compared to other well-established choices.

[9]In our experiments, we use a separable dictionary filter since they can match [32], [33], [34], [35] the performance of non-separable filters.

[10]This modification was originally proposed in [36] for the standard Cauchy step (7a) and was thoroughly analyzed in [37], along with other random variants, in the context of the gradient descent method.

REFERENCES

[1] J. Eckstein and D. Bertsekas, “On the douglas-rachford splitting method and the proximal point algorithm for maximal monotone operators,” *Mathematical Programming*, vol. 55, no. 1, pp. 293–318, Apr 1992.

[2] P. Combettes and V. Wajs, “Signal recovery by proximal forward-backward splitting,” *Multiscale Modeling & Simulation*, vol. 4, no. 4, pp. 1168–1200, 2005.

[3] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[4] S. Becker, E. Candès, and M. Grant, “Templates for convex cone problems with applications to sparse signal recovery,” *Mathematical Programming Computation*, vol. 3, no. 3, p. 165, Jul 2011.

[5] A. Beck and M. Teboulle, “A fast iterative shrinkage-thresholding algorithm for linear inverse problems,” *SIAM J. Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[6] H. Bristow, A. Eriksson, and S. Lucey, “Fast convolutional sparse coding,” in *IEEE CVPR*, 2013, pp. 391–398.

[7] B. Wohlberg, “Efficient algorithms for convolutional sparse representations,” *IEEE TIP*, vol. 25, no. 1, pp. 301–315, Jan. 2016.

[8] Y. Dai and X. Yang, “A new gradient method with an optimal stepsize property,” *Computational Optimization and Applications*, vol. 33, no. 1, pp. 73–88, Jan 2006.

[9] J. Barzilai and J. Borwein, “Two-point step size gradient methods,” *IMA Journal of Numerical Analysis*, vol. 8, no. 1, pp. 141–148, 1988.

[10] A. Beck and M. Teboulle, “Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems,” *IEEE TIP*, vol. 18, no. 11, pp. 2419–2434, Nov. 2009.

[11] J. Liang, J. Fadili, and G. Peyré, “Activity identification and local linear convergence of forward–backward-type methods,” *SIAM Journal on Optimization*, vol. 27, no. 1, pp. 408–437, 2017.

[12] Y. Yuan, “Step-sizes for the gradient method,” 2008.

[13] J. Nocedal and S. Wright, *Numerical Optimization*, 2nd ed. Springer, 2006.

[14] K. Scheinberg, D. Goldfarb, and X. Bai, “Fast first-order methods for composite convex optimization with backtracking,” *Foundations of Computational Mathematics*, vol. 14, no. 3, pp. 389–417, 2014.

[15] M. Figueiredo, R. Nowak, and S. Wright, “Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 1, no. 4, pp. 586–597, Dec 2007.

[16] T. Blumensath and M. Davies, “Iterative thresholding for sparse approximations,” *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 629–654, Dec 2008.

[17] F. Iutzeler and J. Malick, “On the proximal gradient algorithm with alternated inertia,” *Journal of Optimization Theory and Applications*, vol. 176, no. 3, pp. 688–710, Mar 2018.

[18] J. Liang and C. Schönlieb, “Faster FISTA,” *arXiv e-prints*, p. arXiv:1807.04005, Jul 2018.

[19] Y. Nesterov, “A method of solving a convex programming problem with convergence rate O(1/k^2),” *Soviet Mathematics Doklady*, vol. 27, no. 2, pp. 372–376, 1983.

[20] A. Chambolle and C. Dossal, “On the convergence of the iterates of ”FISTA”,” *J. of Optimization Theory and Applications*, vol. Volume 166, no. Issue 3, p. 25, Aug. 2015.

[21] W. Su, S. Boyd, and E. Candès, “A differential equation for modeling nesterov’s accelerated gradient method: Theory and insights,” *J. of Machine Learning Research*, vol. 17, no. 153, pp. 1–43, 2016.

[22] H. Attouch and J. Peypouquet, “The rate of convergence of nesterov’s accelerated forward-backward method is actually faster than 1/k^2,” *SIAM J. on Optimization*, vol. 26, no. 3, pp. 1824–1834, 2016.

[23] P. Rodriguez, “Improving FISTA’s speed of convergence via a novel inertial sequence,” in *European Signal Processing Conference (EUSIPCO)*, A Coruña, Spain, Sep. 2019.

[24] Y. Nesterov, “Gradient methods for minimizing composite objective function,” Université Catholique de Louvain, CORE Discussion Papers 2007/076, 2007.

[25] P. Tseng, “On accelerated proximal gradient methods for convex-concave optimization,” 2008, *submitted* SIAM Journal on Optimization.

[26] R. Gu and A. Dogandzić, “Projected nesterov’s proximal-gradient algorithm for sparse signal recovery,” *IEEE Transactions on Signal Processing*, vol. 65, no. 13, pp. 3510–3525, July 2017.

[27] M. Florea and S. Vorobyov, “A robust fista-like algorithm,” in *IEEE ICASSP*, March 2017, pp. 4521–4525.

[28] Y. Yang, J. Feng, N. Jojic, J. Yang, and T. S. Huang, “On the Suboptimality of Proximal Gradient Descent for ℓ^0 Sparse Approximation,” *ArXiv e-prints*, no. 1709.01230v1 [math.OC], Sep. 2017.

[29] P. Rodriguez, “Simulations for FISTA,” http://goo.gl/gjaj3p.

[30] J. Yang, K. Yu, and T. Huang, “Supervised translation-invariant sparse coding,” in *IEEE CVPR*, June 2010, pp. 3517–3524.

[31] M. Zeiler, D. Krishnan, G. Taylor, and R. Fergus, “Deconvolutional networks,” in *IEEE CVPR*, June 2010, pp. 2528–2535.

[32] G. Silva, J. Quesada, P. Rodriguez, and B. Wohlberg, “Fast convolutional sparse coding with separable filters,” in *IEEE ICASSP*, March 2017, pp. 6035–6039.

[33] J. Quesada, P. Rodriguez, and B. Wohlberg, “Separable dictionary learning for convolutional sparse coding via split updates,” in *IEEE ICASSP*, April 2018, pp. 4094–4098.

[34] G. Silva, J. Quesada, and P. Rodriguez, “Efficient separable filter estimation using rank-1 convolutional dictionary learning,” in *IEEE MLSP*, Sept. 2018.

[35] J. Quesada, G. Silva, P. Rodriguez, and B. Wohlberg, “Combinatorial separable convolutional dictionaries,” in *Symposium on Image, Signal Processing and Artificial Vision (STSIVA)*, Bucaramanga, Colombia, Apr. 2019.

[36] M. Raydan and B. Svaiter, “Relaxed steepest descent and cauchy-barzilai-borwein method,” *Computational Optimization and Applications*, vol. 21, no. 2, pp. 155–167, Feb 2002.

[37] S. Đorđević, “Two modifications of the method of the multiplicative parameters in descent gradient methods,” *Applied Mathematics and Computation*, vol. 218, no. 17, pp. 8672 – 8683, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0096300312001518