Improving FISTA's Speed of Convergence via a Novel Inertial Sequence

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

***Abstract*—The FISTA (fast iterative shrinkage-thresholding algorithm) is a well-known and fast (theoretical $\mathcal{O}(k^{-2})$ rate of convergence) procedure for solving optimization problems composed by the sum of two convex functions, such that one is smooth (differentiable) and the other is possible nonsmooth.**

FISTA can be understood as a first order method with one important aspect: it uses a suitable extragradient rule, i.e.: the gradient is evaluated at a linear combination of the past two iterates, whose weights, are usually referred to as the inertial sequence. While problem dependent, it has a direct impact on the FISTA's practical computational performance.

In this paper we propose a novel inertial sequence; when compared to well-established alternative choices, in the context of convolutional sparse coding and Wavelet-based inpainting, our proposed inertial sequence can reduce the number of FISTA's global iterations (and thus overall computational time) by $30\% \sim 50\%$ to attain the same level of reduction in the cost functional.

***Index Terms*—FISTA, inertial sequence, proximal gradient method, convolutional sparse coding, Wavelet-based inpainting.**

I. Introduction

FISTA [1] is part of the family of accelerated methods (see [2, Section 5.2]), with theoretical $\mathcal{O}(k^{-2})$ rate of convergence[1], and it targets the minimization of the composite function $F(\mathbf{u}) = f(\mathbf{u}) + g(\mathbf{u})$, where $f,\, g : \mathbb{R}^N \mapsto \mathbb{R}$ are both convex, $f(\cdot)$ is continuously differentiable with L-Lipschitz continuous gradient, and, while $g(\cdot)$ may be nonsmooth, its proximal operator,

$$\underset{g}{\text{prox}}(\mathbf{y}) = \arg\min_{\mathbf{u}} \frac{1}{2}\|\mathbf{u}-\mathbf{y}\| + g(\mathbf{u}), \tag{1}$$

has a computationally simple or affordable solution.

FISTA is a very popular choice among several others[2] to minimize the composite function $F(\mathbf{u})$, especially when $g(\mathbf{u}) = \lambda \cdot \|\mathbf{u}\|_1$, and, in general, it generates the iterates

$$\mathbf{u}^{(k)} = \underset{g}{\text{prox}}(\mathbf{y}^{(k)} - \alpha_k \nabla f(\mathbf{y}^{(k)})) \tag{2}$$

$$\mathbf{y}^{(k+1)} = \mathbf{u}^{(k)} + \gamma_k(\mathbf{u}^{(k)} - \mathbf{u}^{(k-1)}) \tag{3}$$

for $k \geq 1$, where $\alpha_k \leq \frac{1}{L}$ and γ_k, the inertial sequence, satisfies

$$\gamma_k = \frac{t_k - 1}{t_{k+1}}, \qquad t_k^2 - t_k \leq t_{k-1}^2 \;\forall k \geq 2. \tag{4a,4b}$$

Popular choices for the inertial sequence $\{\gamma_k\}$, considering $t_1 = 1$, can be generated using (4)[3]: Originally, [1] proposed to use (5a)[4], while more recently, among others, [10], [11], [3] used (5b) for several values of $b \geq 2$ (being $b = 2$ common practice).

$$t_k = \frac{1+\sqrt{1+4*t_{k-1}^2}}{2}, \quad t_k = \frac{k-1+b}{b},\; b \geq 2. \tag{5a,5b}$$

Moreover, in [10] the inertial sequences generated by (5a) and (5b), which yields $\gamma_k = \frac{k-1}{k-1+(b+1)}$, were assessed for different tasks (Wavelet-based inpainting and deblurring, TV denoising) with $b = \{2,\, 3,\, 4\}$; it concluded that the best sequence is problem dependent. Similarly, [11] assessed[5] (5b), also with $b = \{2,\, 3,\, 4\}$, for basis pursuit denoising [12] (and variants) and the logistic regression and its ℓ_1 variant; for those experiments, (5b) with $b = 4$ gave the best performance. On the other hand, [3][6] focused on a theoretical analysis of inertial sequences generated by (5b), and proved that it gives FISTA a rate of convergence of $o(k^{-2})$ rather than $\mathcal{O}(k^{-2})$.

In this paper we propose a novel inertial sequence, which can be generated by considering $t_1 = 1$, and (6a), leading to (6b) for $k \geq 2$.

$$t_k = \frac{k-1+a}{b},\; b \geq 2, a \geq b-1; \quad \gamma_k = \frac{k-(1+b)+a}{k+a}. \tag{6a,6b}$$

While in Section III we give a theoretical analysis of (6a), where we prove that FISTA along with (6b) indeed converges to the minimum of $F(\cdot)$ with a speed of, at least, $\mathcal{O}(k^{-2})$, along with comments about its relationship to (5b)[7], here we highlight that a is a free parameter in (6a) and that it offers a great deal of flexibility, when compared to either (5a) or (5b), for selecting the actual inertial sequence. Furthermore, our simulations (see Section IV), which focus on the Wavelet-based inpainting and convolutional sparse coding (CSC) [13] problems, also offer additional computational evidence that FISTA along with (6b) delivers better performance than either (5a) or (5b); namely our proposed inertial sequence (i) can reduce the number of FISTA's global iterations (and thus overall computational time) by $30\% \sim 50\%$ to attain the same level of reduction in the cost functional and (ii) the rate of reconstruction quality (w.r.t. global iterations) is better or equal to that of the alternatives.

II. Previous related works

In this Section, we provide a succinct review of three topics which are related to the assessment of theoretical / practical aspects of the proposed inertial sequence.

A. A brief review of FISTA

The Fast Iterative Shrinkage Thresholding algorithm (FISTA) [1] is a proximal gradient method [14, Section 7.1.1] which considers

[1]Recently, [3] has proved that FISTA's rate of convergence is $o(k^{-2})$.

[2]E.g. Douglas-Rachford splitting [4], forward-backward splitting [5], ADMM [6], etc.

[3]Other recent alternatives such as [7], [8], include ad-hoc rules or many more parameters than (6), are not listed due to their additional complexity.

[4]same as Nesterov's acceleration scheme [9].

[5][11] also presented an ordinary differential equation (ODE) interpretation of the FISTA algorithm and evaluated the merit of several re-start strategies for the inertial sequence.

[6][3] also exploited FISTA's ODE interpretation.

[7]It is straightforward to notice that (6a) is indeed a generalization of (5b), i.e. set $a = b \geq 2$ in (6a).

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

$g(\mathbf{u}) = \lambda \cdot \|\mathbf{u}\|_1$ along with a Nesterov's multi-step gradient method [9], i.e. the gradient is evaluated at a particular linear combination of the past two iterates, and can be proved to achieve a theoretical $\mathcal{O}(k^{-2})$ rate of convergence.

FISTA is usually referred to as an accelerated version of ISTA (Iterative Shrinkage Thresholding algorithm) [15], and it is popular in image processing applications that promote sparsity, such as basis pursuit denoising (BPDN) [12], Total Variation [16], Principal Component Pursuit [17], etc.

The computational steps of FISTA are given in (2)-(3), where γ_k is referred to as the inertial sequence and α_k denotes a non negative real number such that $\alpha_k \leq \frac{1}{L}$, where L is the Lipschitz constant of ∇f.

In Section I we have already given remarks about the importance of the inertial sequence γ_k. However, from a practical point of view, the selection of the parameter α_k is also as important, since for large-scale problems, L is not always known or computable. In the original work, [1, Section 3] proposed a backtracking line search, to determine α_k at each iteration; however such selection has a drawback: multiple evaluations of the cost functional, which, for large-scale problem, can be computationally costly. While there are several alternatives (see for instance [18]), in our computational results (Section IV) we will use (7), a variant of the Cauchy step-size

$$\alpha_k = \frac{\|\mathbf{s}^{(k)} \odot \mathbf{p}^{(k)}\|_2^2}{\|\Phi(\mathbf{s}^{(k)} \odot \mathbf{p}^{(k)})\|_2^2}, \tag{7}$$

where $\mathbf{s}^{(k)} = I_{[|\mathbf{u}^{(k)}|>0]}$, $I_{[\text{COND}]}$ represents the Indicator function[8] and $\odot$ represents element-wise product. We note that the computation of (7) is fast; furthermore, it was originally tested in context of sparse representations [19], and more recently, it has also delivered good results in the convolutional sparse coding context [20], [21]

B. Wavelet domain methods

A simple yet effective method for image restoration is to consider that the noise-free version of the observed data has a sparse representation in the Wavelet domain (among many others, see [22, Chapter 11]). In this case, the minimization of the general composite function $F(\mathbf{u})$ has the form of

$$\underset{\{\mathbf{u}\}}{\arg\min} \quad \frac{1}{2}\|A\Phi\mathbf{u} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{u}\|_1, \tag{8}$$

where A is a linear operator and Φ represents the inverse Wavelet transform.

While problem (8) can be easily solved via a large number of algorithms, here we highlight that its FISTA-based solution is straightforward, and will be used to solve the inpainting problem, i.e. A is a mask operator (in our case, a diagonal matrix with values either 1 or 0) and $\mathbf{b}$, the observed image, is equal to the noise-free image multiplied by A. We consider this problem (details in Section IV-B) to match one of the examples considered in [10].

C. Convolutional Sparse Coding (CSC)

Convolutional sparse representation (CSR) [23], [24] 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) [28], defined as

$$\underset{\{\mathbf{u}_k\}}{\arg\min} \quad \frac{1}{2}\left\|\sum_{k=1}^{K} H_k * \mathbf{u}_k - \mathbf{b}\right\|_2^2 + \lambda \sum_{k=1}^{K} \|\mathbf{u}_k\|_1, \tag{9}$$

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.

There exist several algorithms that directly solve (9), which can be loosely differentiated by the domain in which the convolutions are performed. Earlier spatial domain approaches were based on ISTA [29] or FISTA [30], assuming a non-separable filter bank (FB) $\{H_k\}$, while more recently, [20] proposed a FISTA based algorithm, assuming a separable FB. On the other hand, frequency domain approaches are usually based on the ADMM framework [13], [31], [28], although some recent works [32], [21] have made use of the Accelerated Proximal Gradient / FISTA framework.

III. Analysis of the Proposed Inertial Sequence

In order to analytically assess the proposed inertial sequence (6a), we first summarize the key results from [1]. To that end, we will follow the procedure described in [10]; in particular, we first reproduce [10, Th. 1]:

Theorem 1. *For any $\mathbf{u}^{(0)} \in \mathbb{R}^N$, if the (non-negative) sequence $\{t_k\}$, $k \in \mathbb{N}^+$ satisfies (4b), i.e. $t_k^2 - t_k \leq t_{k-1}^2$, $\forall k \geq 2$, and $t_1 = 1$, then sequence $\{\mathbf{u}^{(k)}\}$, generated by (2)-(3), with $\alpha_k = \alpha \leq \frac{1}{L}$, satisfies $\forall k \in \mathbb{N}$*

$$F(\mathbf{u}^{(k)}) - F(\mathbf{u}^*) \leq \frac{1}{2 \cdot \alpha \cdot t_k^2}\|\mathbf{u}^{(0)} - \mathbf{u}^*\|_2^2 \tag{10}$$

for any minimizer $\mathbf{u}^*$ of F.

While it is straightforward to show that sequences (5a), (5b), as well as the proposed sequence (6a), i.e. $t_k = \frac{k-1+a}{b}$, all with $t_1 = 1$, satisfy (4b), we will explicitly show such relationship for our proposed sequence. In fact, we seek $\rho_k = t_k^2 - t_k - t_{k-1}^2 \leq 0$, thus by using (6a), we get

$$\begin{aligned} \rho_k &= \left(\frac{k-1+a}{b}\right)^2 - \frac{k-1+a}{b} - \left(\frac{k-2+a}{b}\right)^2 \\ &= -\frac{b \cdot (k-1+a)}{b^2} + \frac{2 \cdot (k-1+a)}{b^2} - \frac{1}{b^2}; \end{aligned} \tag{11}$$

if we take $b = 2$, then

$$\rho_k = -\frac{1}{4} < 0 \tag{12}$$

$\forall a \geq 1$, since sequence t_k, and the inertial sequence generated by it, given in (6b), i.e. $\gamma_k = \frac{k-3+a}{k+a}$ for $k \geq 2$ (by definition, $\gamma_0 = 0$) must be non-negative. Here we stress that parameter a is basically a "free" parameter: from a theoretical point of view (although there are practical limitations), we can choose any value of $a \geq 1$; as we will show next, by adequately selecting a, we can directly control the bound on $F(\mathbf{u}^{(k)}) - F(\mathbf{u}^*)$ for small/medium values of k, which results (i) in a faster reduction of the cost functional of F and (ii) in an improved rate of reconstruction quality (see Section IV).

For sequences (5a), (5b), as well as the proposed sequence (6a), Theorem 1 ensures that

$$\forall n \in \mathbb{N} \quad F(\mathbf{u}^{(k)}) - F(\mathbf{u}^*) \leq \frac{C}{t_k^2}; \tag{13}$$

in Figure 1 we plot (in log-scale) the evolution of $\frac{1}{t_k^2}$ for (5a), (5b) with $b = \{2, 3, 4\}$, and (6a) with $b = 2$ and $a = \{2, 25\}$, all with

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

[9] [25], [26], [27] showed that natively learned separable filters consistently attain the same reconstruction quality (noise-free and restoration cases) as when using standard non-separable filters of the same characteristics (size and number).

$t_k = 1$. While for large enough values of k, all sequences will be indistinguishable, clearly the proposed sequence, with large values of parameter a, can impose a better bound of $F(\mathbf{u}^{(k)}) - F(\mathbf{u}^*)$ for small/medium values of k.

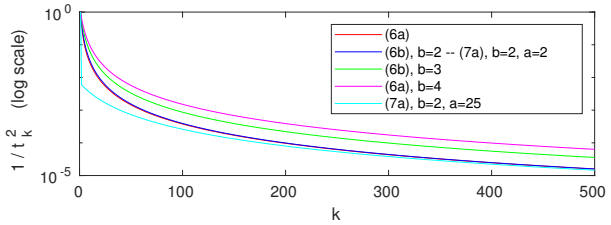

Fig. 1: The evolution of $\frac{1}{t_k^2}$ is plotted for sequences (5a), (5b) with $b = \{2,\ 3,\ 4\}$, and the proposed inertial sequence (6a) with $b = 2$ and $a = \{2,\ 25\}$.

A. Additional comments

1) Relationship between (5b) and (6a): As it was mentioned in the Introduction (Section I), the proposed sequence (6a) can be understood as a generalization of (5b). Indeed, if we take $a = b \geq 2$ in (6a) we recover (5b); this is also experimentally shown in Figure 1 for the case of $a = b = 2$.

However, by decoupling the values a and b, (6a) offers more flexibility than (5b) when selecting the actual inertial sequence, and, crucially, it also offers a fine control over the bound of $F(\mathbf{u}^{(k)}) - F(\mathbf{u}^*)$ for small/medium values of k.

2) Selecting $b > 2$ in (6a): In the previous Section, we showed that (6a) satisfies (4b) for $b = 2$. Clearly, if we take $b > 2$, (6a) also satisfies (4b), however from a computational point of view[10] we have not observed any practical gain by doing so.

3) Selecting a variable parameter in (6a): From our analysis, summarized by (11)-(12), variable a in (6a) is a free parameter, thus it can be replaced by a_k, i.e. a function of k.

In this case, ρ_k (see (11)) considering $b = 2$ will be given by

$$\begin{aligned}\rho_k &= \left(\frac{k-1+a_k}{2}\right)^2 - \frac{k-1+a_k}{2} - \left(\frac{k-2+a_{k-1}}{2}\right)^2 \\ &= \frac{\beta_k \cdot (a_k - a_{k-1})}{4} - \frac{a_k - a_{k-1}}{2} - \frac{1}{4}, \end{aligned} \quad (14)$$

where $\beta_k = 2k - 2 + a_k + a_{k-1}$. Since $\beta_k > 0 \, \forall k \geq 1$, and, at least, it grows as $\frac{k}{2}$, (14) can be forced to be negative by considering $a_k \leq a_{k-1}$. A simple rule to select a_k, such as taking a_1 equal to a large value and then decreasing it to a particular value, e.g. 2, can improve FISTA's practical performance for some cases: In our experimental results (Section IV) we show numerical evidence that such rule does improve FISTA's practical performance for the Wavelet-based inpainting problem (see Section IV-B); however, for the CSC problem (Section IV-C) such rule does not give any practical gain.

IV. Computational assessment of the proposed inertial sequence

A. Experimental setup

The set of experiments detailed below were carried out using Matlab, running 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 [33] can be used to reproduce the computational results presented here, along with some extended simulations.

Our experiments focus on assessing the computational behavior of several inertial sequences (I.Seq) when solving (i) the Wavelet-based inpainting (noiseless) problem and (ii) the CSC (noiseless and noisy cases) problem, along with separable filters, via a FISTA-based approach, which runs for at most 300 iterations[11].

The considered I.Seq are those generated by (5a), (5b) with $b = \{2,\ 3,\ 4\}$, and the proposed I.Seq (6a) with $b = 2$ and a variable (see Section III-A3) or fixed parameter a, heuristically chosen in both cases. For both problems, we consider five test images ("Lena", "Barbara", "Kiel" and "Bridge", each 512×512 pixel, and "Man", 1024×1024 pixel).

B. Wavelet-based inpainting (W_INPT) problem

For the W_INPT problem we consider that A in (8) is a diagonal matrix, whose entries are either 0 or 1, i.e. a mask operator. For the results presented below, we consider that the mask operator remove 50% of the original image pixel at random locations. Furthermore, we select Φ in (8) as the orthonormal Daubechies wavelet transform, in order to match one of the examples considered in [10]

In Figure 2 we depict the cost functional and reconstruction (SNR) evolution when solving the W_INPT problem with $\lambda = 7.5e\text{-}4$ along with the considered I.Seq, for the "Barbara" test image[12].

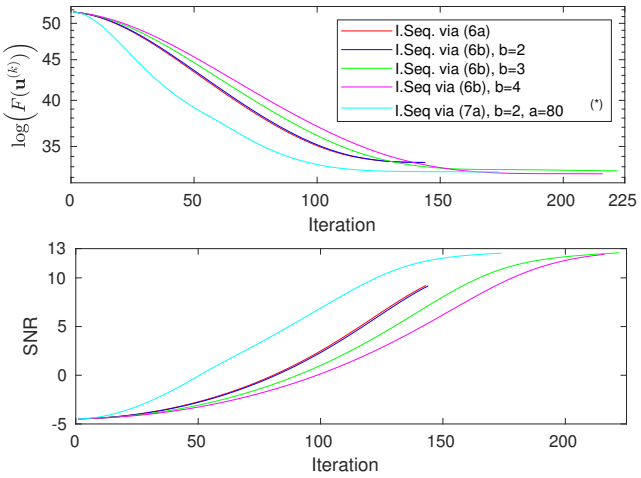

Fig. 2: Comparison between several inertial sequences when solving the W_INPT problem (8) with $\lambda = 7.5e\text{-}4$ for the "Barbara" test image[12]. The top/bottom plots depict the cost functional, in logarithmic scale, and reconstruction (SNR) evolution versus global iterations. $^{(*)}$ The variable parameter selection, described in Section III-A3 is used: $a_k = \max(80 - 1.56k, 2)$.

The cost functional in Figure 2 (top plot, in logarithmic scale) clearly follows the theoretical cost functional bound[13], depicted in Figure 1, up to the iteration where the stopping criterion is met[10].

[10]Due to space constraints, we do not present such experiments in Section IV, however they can be reproduced via our freely available Matlab code [33]

[11]It is well-known that a local oscillatory behavior can be observed in FISTA (originally observed in [34]; see also [35] for a formal description); thus, in our code, we include an exit condition if such behavior is observed.

[12]These results are representative; results for other test images, noise levels, λ values or wavelet / convolutional dictionaries can be generated by our companion Matlab code [33].

[13]The I.Seq generated by (5a) and (5b) with $b = 2$ have a very similar performance, whereas (5b) with $b = \{3,\ 4\}$, while also exhibiting a quadratic reduction of the cost functional, their actual values are larger than any of the former for small/medium values of k.

For the W_INPT problem we have observed that using the variable parameter selection, described in Section III-A3, gives the best trade-off between cost functional reduction and reconstruction quality. In particular we set $a_k = \max(80 - 1.56k, 2)$: i.e. variable a decreases from 80 down to 2 for the first 50 iterations.

C. CSC problem

For this problem (both noiseless and noisy cases), we use a set of 36 separable filters, size 12×12, learned via [25], and solve the CSC problem (9) via [20], which is a FISTA-based approach, explicitly tailored for separable filters. We must note that [25], [26], [27] showed that natively learned separable filters consistently attain the same reconstruction quality (noise-free, denoising and inpainting cases) as when using standard non-separable filters of the same characteristics (size and number). The above mentioned test images were not used in the dictionary learning stage.

1) Noiseless case: In Figure 3 we depict the cost functional and reconstruction (SNR) evolution when solving the CSC problem (9) with $\lambda = 0.01$ along with the considered I.Seq, for the noiseless "Man" test image[12].

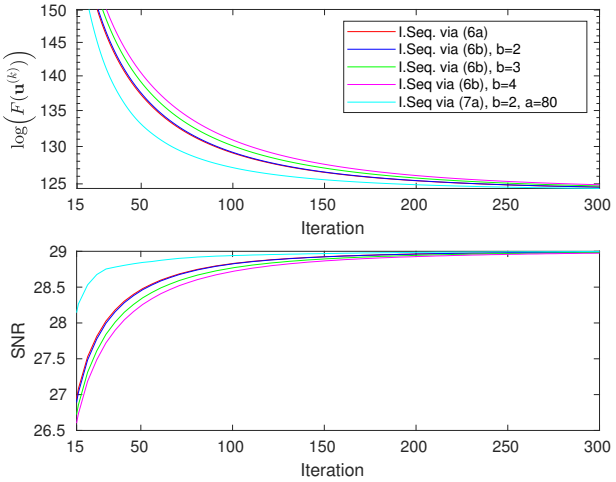

Fig. 3: Comparison between several inertial sequences when solving the CSC problem (9) with $\lambda = 0.01$ for the "Man" test image[12]. The top/bottom plots depict the cost functional, in logarithmic scale, and reconstruction (SNR) evolution versus global iterations.

The cost functional (Figure 3, top plot, in logarithmic scale) is shown from iteration 15 onward (up to 300) in order to visually distinguish the effect of each I.Seq. As for the W_INPT problem (see Section IV-B), all evaluated I.Seq follow the theoretical cost functional bound[13], depicted in Figure 1.

2) Noisy case: In Figure 4 we depict the cost functional and reconstruction (SNR) evolution when solving the CSC problem (9), along with the considered I.Seq, for the noisy (corrupted with uncorrelated additive Gaussian noise, $\sigma_\eta^2 = 0.01$) "Lena" test image[12]. The regularization parameter, $\lambda = 0.245$, was manually selected to produce the best (SNR) quality result for the I.Seq defined by (5a), i.e. FISTA along with the original I.Seq proposed in [1].

The cost functional (Figure 4, top plot, in logarithmic scale) is shown from iteration 10 up to 100 in order to visually distinguish the effect of each I.Seq. (from iteration 100 onward there is no significant change for any case). As for the W_INPT problem (Section IV-B) and the noiseless CSC case (Section IV-C1) all the considered I.Seq follow the theoretical cost functional bound[13], depicted in Figure 1, where we note that the behavior of the proposed I.Seq is superior

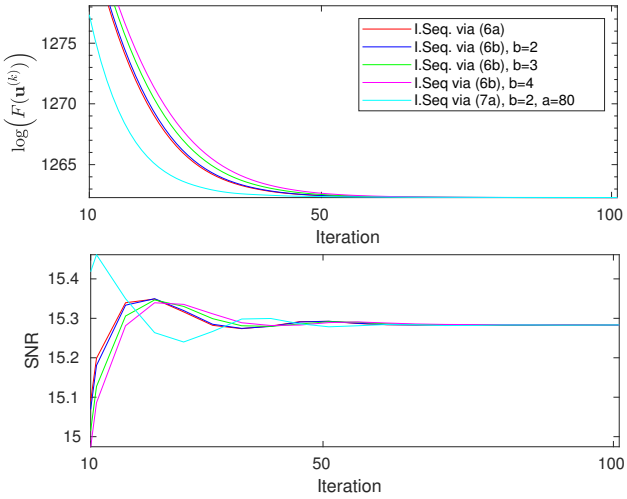

Fig. 4: Comparison between several inertial sequences when solving the CSC problem (9), noisy case (uncorrelated additive Gaussian noise, $\sigma_\eta^2 = 0.01$) with $\lambda = 0.245$ for the "Lena" test image[12]. The top/bottom plots depict the cost functional, in logarithmic scale, and reconstruction (SNR) evolution versus global iterations.

to all the alternatives in terms of speed of reduction of (9)'s cost functional while achieving the same reconstruction quality.

D. Discussion

For the two considered (W_INPT and CSC) problems, the experimental results detailed in Sections IV-B and IV-C heuristically confirm the worthwhile theoretical behavior of the proposed I.Seq: for small/medium values of k, it exhibits a faster speed of reduction of the cost functional, of either (8) or (9), than any of the popular I.Seq choices (see (5a) and (5b)).

For the W_INPT problem, the proposed I.Seq roughly saves between 30% and 50% of global iterations w.r.t. (5a) and variants of (5b) to attain the same level of reduction in the cost functional (see Fig. 2). Furthermore, its associated rate of reconstruction quality (SNR) follows a similar trend, outperforming the alternatives.

Similarly, for the CSC problem, the proposed I.Seq also roughly saves between 30% and 50% of global iterations w.r.t. the alternative I.Seq (see Figures 3 and 4). However, in this case, the associated rate of reconstruction quality varies depending on the noise level: for the noiseless case, our proposed I.Seq also clearly outperforms all the alternatives; however, for the noisy case all the alternatives behave very similar.

V. Conclusion

In this work we have proposed a novel inertial sequence for FISTA, and assessed its theoretical properties along with its computational worthiness in the context of Wavelet-based inpainting (W_INPT) and Convolutional Sparse Coding (CSC).

While well-established inertial sequences, as well as the proposed one, used in FISTA, all deliver a theoretical $\mathcal{O}(k^{-2})$ rate of convergence, their actual speed varies for small/medium values of k (or global iterations). Particularly, in the context of W_INPT and CSC, the proposed inertial sequence exhibits the best performance (cost function reduction point of view, and in some instances, reconstruction quality point of view) for small/medium values of k and can be used to reduce FISTA's global number of iterations by $30\% \sim 50\%$.

REFERENCES

[1] 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.

[2] 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.

[3] 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.

[4] 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.

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

[6] 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.

[7] F. Iutzeler and J. Malick, "On the proximal gradient algorithm with alternated inertia," *J. Optimization Theory and Applications*, vol. 176, no. 3, pp. 688–710, 2018.

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

[9] 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.

[10] 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.

[11] 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.

[12] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, Jan. 2001.

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

[14] N. Parikh and S. Boyd, "Proximal algorithms," *Foundations and Trends in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.

[15] I. Daubechies, M. Defrise, and C. D. Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Comm. on Pure and Applied Math.*, vol. 57, no. 11, pp. 1413–1457, 2004.

[16] 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.

[17] Z. Lin, A. Ganesh, J. Wright, L. Wu, M. Chen, and Y. Ma, "Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix," in *Int'l. Workshop on Comp. Adv. in Multi-Sensor Adapt. Processing*, 2009.

[18] Y. Yuan, "A new stepsize for the steepest descent method," *J. of Computational Mathematics*, vol. 24, pp. 149–156, 03 2006.

[19] 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.

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

[21] G. Silva and P. Rodríguez, "Efficient convolutional dictionary learning using partial update fast iterative shrinkage-thresholding algorithm," *IEEE ICASSP*, pp. 4674–4678, 2018.

[22] M. Unser and P. Tafti, *An introduction to sparse stochastic processes*. Cambridge University Press, 01 2014.

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

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

[25] 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.

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

[27] 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.

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

[29] M. Zeiler, G. Taylor, and R. Fergus, "Adaptive deconvolutional networks for mid and high level feature learning," in *ICCV*, 2011, pp. 2018–2025.

[30] R. Chalasani, J. C. Principe, and N. Ramakrishnan, "A fast proximal method for convolutional sparse coding," in *Int'l Joint Conf. on Neural Net.*, Aug. 2013, pp. 1–5.

[31] F. Heide, W. Heidrich, and G. Wetzstein, "Fast and flexible convolutional sparse coding," in *IEEE CVPR*, Jun. 2015, pp. 5135–5143.

[32] C. Garcia-Cardona and B. Wohlberg, "Convolutional dictionary learning: A comparative review and new algorithms," *IEEE Transactions on Computational Imaging*, vol. 4, no. 3, pp. 366–381, Sep. 2018.

[33] P. Rodriguez, "Simulations for FISTA," http://goo.gl/gjaj3p.

[34] 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.

[35] 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.