FISTA is an automatic geometrically optimized algorithm for strongly convex functions

Jean-François Aujol [1]

Joint work with Charles Dossal [2] and Aude Rondepierre [2]

[1] Institut de Mathématiques de Bordeaux, Université de Bordeaux
[2] Institut de Mathématiques de Toulouse, INSA Toulouse

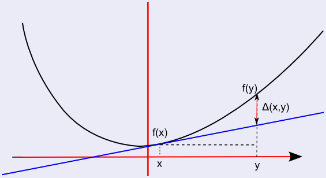
SMAI 2023

Composite optimization

$$\text{Minimize } F(x) = f(x) + h(x), \quad x \in \mathbb{R}^N,$$

where:

- f is a convex differentiable function,
  i.e: $f(y) \geqslant f(x) + \langle \nabla f(x), y - x \rangle$, with a L-Lipschitz gradient:



For all $(x, y) \in \mathbb{R}^N \times \mathbb{R}^N$, we have:

$$f(y) \leqslant \underbrace{f(x) + \langle \nabla f(x), y - x \rangle}_{\text{linear approximation}} + \underbrace{\frac{L}{2}\|y - x\|^2}_{=\Delta(x,y)}$$

- h is a convex lower semicontinuous (lsc) *simple* function.

$\hookrightarrow$ Application to least square problems, LASSO ($\min_{x \in \mathbb{R}^N} \frac{1}{2}\|Ax - b\|_2^2 + \|x\|_1$)

$\hookrightarrow$ Applications in Image and Signal processing, machine learning, deep learning, Artificial Intelligence, ...

The setting: local geometry of convex functions

In this talk we assume that the composite convex function $F = f + h$ satisfies a quadratic growth condition around its set of minimizers:

Quadratic growth condition $\mathcal{G}_\mu^2$

There exists $\mu > 0$ such that:

$$\forall x \in \mathbb{R}^N,\ F(x) - F(x^*) \geqslant \frac{\mu}{2} d(x, X^*)^2$$

where $X^* = \arg\min F$ and $F^* := F(x^*) = \min F$.

Strong convexity property

$$\forall x \in \mathbb{R}^N, y \in \mathbb{R}^N,\ F(y) \geqslant F(x) + \langle \nabla F(x), y - x \rangle + \frac{\mu}{2}\|x - y\|^2$$

The quadratic growth condition is a relaxation of the strong convexity property.

Strongly convex functions or quadratic growth functions

LASSO problem with A invertible

$$F(x) = \frac{1}{2}\|Ax - b\|_2^2 + \|x\|_1$$

Then there exists $\mu > 0$ such that F is μ strongly convex.

LASSO problem with A non injective

$$F(x) = \frac{1}{2}\|Ax - b\|_2^2 + \|x\|_1$$

Then there exists $\mu > 0$ such that F satisfies $\mathcal{G}_\mu^2$, but F is not μ strongly convex.
[Bolte et al 2013]

The setting: local geometry of convex functions

In this talk we assume that the composite convex function $F = f + h$ satisfies a quadratic growth condition around its set of minimizers:

Quadratic growth condition $\mathcal{G}^2_\mu$

There exists $\mu > 0$ such that:

$$\forall x \in \mathbb{R}^N,\ F(x) - F(x^*) \geqslant \frac{\mu}{2} d(x, X^*)^2$$

where $X^* = \arg\min F$ and $F^* = \min F$.

Łojasiewicz property with an exponent $\frac{1}{2}$

$$F(x) - F(x^*) \leqslant \frac{1}{2\mu} \|\nabla F(x)\|^2$$

In the convex setting, both properties are equivalent.

The setting: large scale optimization

In this talk we assume that the composite convex function $F = f + h$ satisfies a quadratic growth condition around its set of minimizers:

Quadratic growth condition $\mathcal{G}_\mu^2$

There exists $\mu > 0$ such that:

$$\forall x \in \mathbb{R}^N,\ F(x) - F(x^*) \geqslant \frac{\mu}{2} d(x, X^*)^2$$

Lipshitz gradient

f is convex with L Lipshitz gradient, i.e;: $\|\nabla f(x) - \nabla f(y)\| \leqslant \|x - y\|$.

Conditionning

We denote by

$$\kappa := \frac{\mu}{L}.$$

We have $0 \leqslant \kappa \leqslant 1$, and in large scale optimization problems, κ is usually very small.

The setting: large scale optimization

In this talk we assume that the composite convex function $F = f + h$ satisfies a μ quadratic growth condition around its set of minimizers in $\mathbb{R}^N$.
f is convex with L Lipshitz gradient.
h is a convex lower semi-continuous function.

$$\kappa := \frac{\mu}{L} = o(1)$$

First order optimization

Since we deal with large scale optimization, we only consider first order optimization methods, i.e. methods that can only use the values of the function to minimize and/or the values of its gradient/subgradient.

Goal

We assume the existence of a minimizer of F on $\mathbb{R}^N$. We are interested in how fast we can compute it. Speed in term of decrease of $F(x_n) - F^*$ with F^* the minimum of F.

Let $\varepsilon > 0$. The minimizers of a composite function $F = f + h$ are characterized by:

$$0 \in \partial F(x) = \nabla f(x) + \partial h(x),$$

or equivalently, for any $\gamma > 0$,

$$x = prox_{\gamma h}\left(x - \gamma \nabla f(x)\right)$$

where: $\mathrm{prox}_{\gamma h}(x) = \arg\min_{y \in \mathbb{R}^N} \gamma h(y) + \frac{1}{2}\|y - x\|^2$.

Definition (ε-solution)

An iterate x_n is said to be an ε-solution of $\min_{x \in \mathbb{R}^N} F(x)$ if:

$$\|g(x_n)\| \leqslant \varepsilon$$

where: $g(x) := L(x - x^+) := L\left(x - \mathrm{prox}_{\frac{1}{L}h}(x - \frac{1}{L}\nabla f(x))\right)$ is the composite gradient mapping.

Two useful properties

1. $\forall x \in \mathbb{R}^N,\ \frac{1}{2L}\|g(x)\|^2 \leqslant F(x) - F^*$ [Nesterov 2007]

$$\text{If } F(x_n) - F^* \leqslant \frac{1}{2L}\varepsilon^2,$$

then x_n is an ε-solution of $\min_{x\in\mathbb{R}^N} F(x)$.

2. $\forall x \in \mathbb{R}^N,\ F(x^+) - F^* \leqslant \frac{2}{\mu}\|g(x)\|^2$ [Aujol-Dossal-Labarrière-Rondepierre 2021]
with $x^+ := \mathrm{prox}_{\frac{1}{L}h}(x - \frac{1}{L}\nabla f(x))$

A tractable stopping criterion

$$\|g(x_n)\| \leqslant \varepsilon$$

Outline

1. **The Forward-Backward and FISTA algorithms**
   - The Forward-Backward algorithm
   - FISTA a fast proximal gradient method
   - FB vs FISTA in the strongly convex case
2. **FISTA is an automatic geometrically optimized algorithm**
   - The dynamical system intuition
   - Convergence rates under some quadratic growth condition
   - Comparisons
3. **Going further: Reducing oscillations**
   - Restart
   - Hessian damping

Forward-Backward algorithm

$$\text{Minimize } F(x) = f(x) + h(x), \quad x \in \mathbb{R}^N.$$

Optimality condition:

$$\{0\} \in \nabla f(x) + \partial h(x)$$

or equivalently, for any $\gamma > 0$,

$$x = prox_{\gamma h}\left(x - \gamma \nabla f(x)\right)$$

where: $\mathrm{prox}_{\gamma h}(x) = \arg\min_{y \in \mathbb{R}^N} \gamma h(y) + \frac{1}{2}\|y - x\|^2$.

Forward-Backward algorithm

$$\begin{aligned} & x_0 \in \mathbb{R}^N \\ & x_{n+1} = prox_{\gamma h}(x_n - \gamma \nabla f(x_n)), \quad 0 < \gamma < \tfrac{2}{L}. \end{aligned}$$

If $\gamma = \frac{1}{L}$, then $x_{n+1} = x_n^+$, and $g(x_n) = L(x_n - x_{n+1})$.
x_n is an ε-solution if $\|g(x_n)\| \leq \epsilon$.

Forward-Backward algorithm to minimize $F = f + h$ with $\gamma = \frac{1}{L}$

$$x_0 \in \mathbb{R}^N$$
$$x_{n+1} = prox_{\frac{1}{L}h}(x_n - \tfrac{1}{L}\nabla f(x_n)) = x_n^+.$$

Instead of minimizing directly $F = f + h$, minimize at each iteration n its quadratic upper bound:

$$x \mapsto f(x_n) + \langle \nabla f(x_n), x - x_n \rangle + \frac{L}{2}\|x - x_n\|^2 + h(x)$$

Hence:

$$\begin{aligned}
x_{n+1} &= \arg\min_{x \in \mathbb{R}^N} \left(f(x_n) + \langle \nabla f(x_n), x - x_k \rangle + \frac{L}{2}\|x - x_n\|^2 + h(x) \right) \\
&= \arg\min_{x \in \mathbb{R}^N} \left(h(x) + \frac{L}{2}\|x - (x_n - \frac{1}{L}\nabla f(x_n))\|^2 + f(x_n) - \frac{1}{2L}\|\nabla f(x_n)\|^2 \right) \\
&= prox_{\frac{1}{L}h}\left(x_n - \frac{1}{L}\nabla f(x_n) \right)
\end{aligned}$$

Forward-Backward algorithm

Basic examples

- Gradient method ($h = 0$, unconstrained optimization):

$$x_{n+1} = x_n - \frac{1}{L}\nabla f(x_n)$$

since: $prox_h(x) = \arg\min_{y \in \mathbb{R}^N} \left(0 + \frac{1}{2}\|y - x\|^2\right) = x$.

- Gradient method ($h = 0$, unconstrained optimization):

$$x_{n+1} = x_n - \frac{1}{L}\nabla f(x_n)$$

since: $prox_h(x) = \arg\min_{y \in \mathbb{R}^N} \left(0 + \frac{1}{2}\|y - x\|^2\right) = x$.

- Gradient projection method ($h = i_C$, constrained convex optimization):

$$x_{n+1} = P_C^{\perp}(x_n - \frac{1}{L}\nabla f(x_n))$$

since: $prox_h(x) = \arg\min_{y \in \mathbb{R}^N} \left(i_C(y) + \frac{1}{2}\|y - x\|^2\right) = P_C^{\perp}(x)$.

- Gradient method ($h = 0$, unconstrained optimization):

$$x_{n+1} = x_n - \frac{1}{L}\nabla f(x_n)$$

since: $prox_h(x) = \arg\min_{y\in\mathbb{R}^N}\left(0 + \frac{1}{2}\|y-x\|^2\right) = x.$

- Gradient projection method ($h = i_C$, constrained convex optimization):

$$x_{n+1} = P_C^{\perp}(x_n - \frac{1}{L}\nabla f(x_n))$$

since: $prox_h(x) = \arg\min_{y\in\mathbb{R}^N}\left(i_C(y) + \frac{1}{2}\|y-x\|^2\right) = P_C^{\perp}(x).$

- Iterative Soft-Thresholding Algorithm (ISTA) ($h = \|\cdot\|_1$):

$$x_{n+1} = prox_{\frac{1}{L}h}\left(x_n - \frac{1}{L}\nabla f(x_n)\right)$$

with: $prox_{\gamma h}(x) = sign(x)\max(0, |x| - \gamma).$

Assume that F is convex. Then:

$$\forall n \geqslant 1,\ F(x_n) - F^* \leqslant \frac{2L\|x_0 - x^*\|^2}{n}.$$

Remember that if $F(x_n) - F^* \leqslant \frac{1}{2L}\varepsilon^2$, then x_n is an ε-solution of $\min_{x \in \mathbb{R}^N} F(x)$.

The number of iterations required by FB to reach an ε-solution in the sense that:

$$\frac{2L\|x_0 - x^*\|^2}{n} \leqslant \frac{1}{2L}\varepsilon^2$$

is at most:

$$\frac{4L^2}{\varepsilon^2}\|x_0 - x^*\|^2 \left(= \mathcal{O}\left(\frac{L^2}{\varepsilon^2}\right)\right).$$

FISTA an accelerated proximal gradient method

FISTA - Beck Teboulle 2009, Nesterov 1984

$$
\begin{aligned}
y_n &= x_n + \frac{t_n - 1}{t_{n+1}}(x_n - x_{n-1}) \\
x_{n+1} &= prox_{\frac{1}{L}h}\left(y_n - \frac{1}{L}\nabla f(y_n)\right).
\end{aligned}
$$

where $t_1 = 1$ and the sequence $(t_n)_{n\in\mathbb{N}}$ is determined as the positive root of:

$$t_{n+1}^2 - t_{n+1} = t_n^2.$$

For the class of convex functions, they prove:

$$F(x_n) - F^* \leqslant \frac{2L\|x_0 - x^*\|^2}{(n+1)^2}$$

[Nesterov 1984] The $\mathcal{O}\left(\frac{1}{n^2}\right)$ rate is optimal for first order methods in the class of convex functions.

FISTA a fast proximal gradient method

FISTA - Chambolle Dossal 2015, Su Boyd Candès 2016

Let $\alpha \geqslant 3$.

$$\begin{aligned} y_n &= x_n + \frac{n}{n+\alpha}(x_n - x_{n-1}) \\ x_{n+1} &= prox_{\frac{1}{L}h}\left(y_n - \frac{1}{L}\nabla f(y_n))\right). \end{aligned}$$

- Initially Nesterov (1984) proposed a choice equivalent to $\alpha = 3$. Convergence of iterates for $\alpha > 3$ [Chambolle-Dossal 2015].
- For the class of composite convex functions:

$$\forall n \geqslant 1,\ F(x_n) - F^* \leqslant \frac{L(\alpha-1)^2\|x_0 - x^*\|^2}{2(n+\alpha-2)^2}$$

  i.e. when $\alpha = 3$: $\forall n \geqslant 1,\ F(x_n) - F^* \leqslant \frac{2L\|x_0 - x^*\|^2}{(n+1)^2}$.

The number of iterations required for FISTA to reach an ε-solution is in $\mathcal{O}\left(\frac{L^2}{\varepsilon}\right)$ which is better than FB.

FB vs FISTA in the strongly convex case
Exponential rate vs Polynomial rate (1/3)

Assume now that F additionally satisfies some quadratic growth condition:

$$\forall x \in \mathbb{R}^N,\ F(x) - F^* \geqslant \frac{\mu}{2} d(x, X^*)^2.$$

Convergence rate for FB [Garrigos, Rosasco, Villa 2017]

$$\forall n \in \mathbb{N},\ F(x_n) - F^* \leqslant (1-\kappa)^n (F(x_0) - F^*).$$

The number of iterations required to reach an ε-solution is:

$$n_\varepsilon^{FB} = \frac{1}{|\log(1-\kappa)|} \log\left(\frac{2L}{\varepsilon^2}(F(x_0) - F^*)\right) \sim \frac{1}{\kappa} \log\left(\frac{2L}{\varepsilon^2} M_0\right).$$

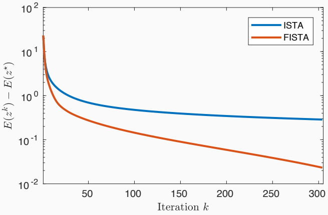
Convergence rate for FISTA [Candès et al 2015], [Attouch Cabot 2017], [ADR 2018].

Assume additionally that F has a unique minimizer.

$$\forall \alpha > 0,\ \forall n \in \mathbb{N},\ F(x_n) - F^* = \mathcal{O}\left(n^{-\frac{2\alpha}{3}}\right)$$

FB vs FISTA in the strongly convex case
Exponential rate vs Polynomial rate (2/3)

(a) Input y: motion blur + noise (σ = 2)



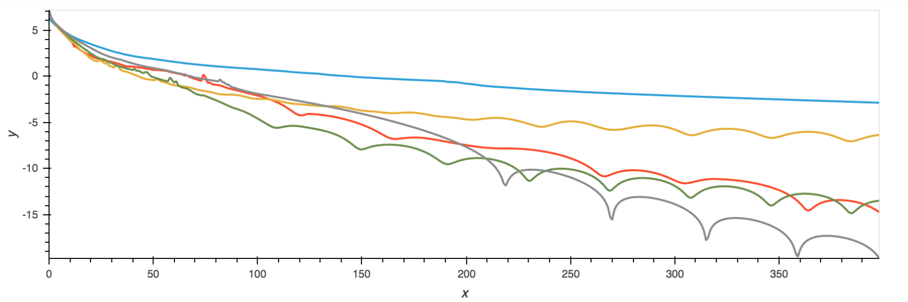
(b) Convergence prof les

(c) Deconvolution ISTA(300)+UDWT

(d) Deconvolution FISTA(300)+UDWT

FB vs FISTA in the strongly convex case

Exponential rate vs Polynomial rate (3/3)



$\log(\|g(x_n)\|)$ along the iterations n

FB, FISTA-restart, FISTA with $\alpha = 3$, FISTA with $\alpha = 12$, FISTA with $\alpha = 30$.

Motivation to provide a non-asymptotic analysis of FISTA and to compare rates in finite time.

Nesterov accelerated algorithm for strongly convex functions

Nesterov accelerated algorithm for strongly convex functions (NSC)

$$y_n = x_n + \frac{1-\sqrt{\kappa}}{1+\sqrt{\kappa}}(x_n - x_{n-1})$$
$$x_{n+1} = prox_{\frac{1}{L}h}\left(y_n - \tfrac{1}{L}\nabla f(y_n)\right)\right).$$

Theorem (Theorem 2.2.3, Nesterov 2013)

Assume that F is μ-strongly convex for some $\mu > 0$. Let $\varepsilon > 0$. Then if $\kappa = \frac{\mu}{L}$,

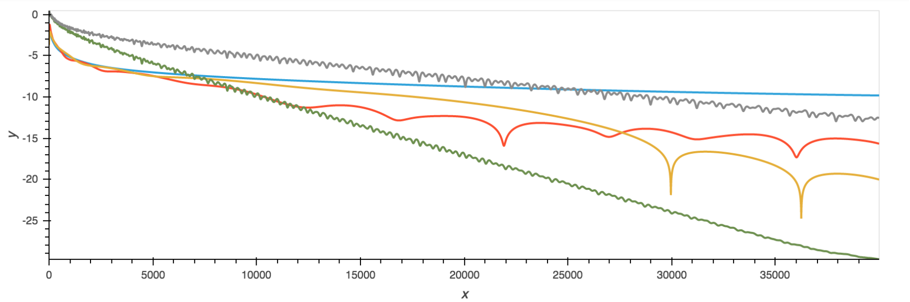
$$\forall n \in \mathbb{N},\ F(x_n) - F(x^*) \leqslant 2(1-\sqrt{\kappa})^n \left(F(x_0) - F(x^*)\right),$$

which means that an ε-solution can be obtained in at most:

$$n_\varepsilon^{NSC} = \frac{1}{|\log(1-\sqrt{\kappa})|} \log\left(\frac{4LM_0}{\varepsilon^2}\right) \sim \frac{1}{\sqrt{\kappa}} \log\left(\frac{4LM_0}{\varepsilon^2}\right). \quad (1)$$

The iterations require an estimation of $\kappa = \frac{\mu}{L}$.
In large scale optimization problems, we usually have $\kappa = o(\sqrt{\kappa})$.

FISTA in the strongly convex case

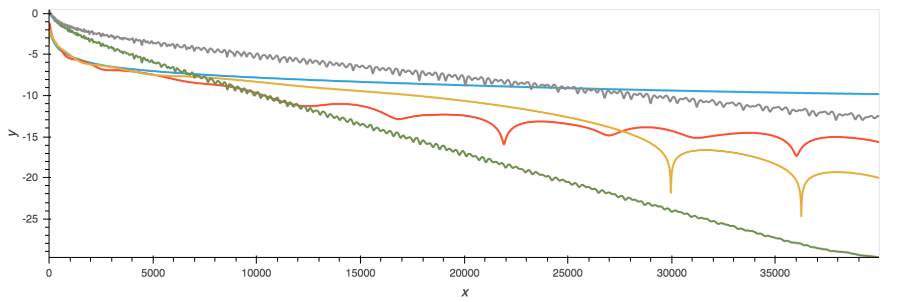


$\log(\|g(x_n)\|)$ along the iterations

FB, FISTA with $\alpha = 8$, FISTA with $\alpha = 30$,

NSC with the true value of μ, NSC with $\widetilde{\mu} = \frac{\mu}{10}$.

FISTA in the strongly convex case



$\log(\|g(x_n)\|)$ along the iterations

FB, FISTA with $\alpha = 8$, FISTA with $\alpha = 30$,

NSC with the true value of μ, NSC with $\widetilde{\mu} = \frac{\mu}{10}$.

FISTA is efficient without knowing μ and its convergence rate does not suffer from any underestimation of μ

Outline

What we want to do now

FISTA: Nesterov accelerated algorithm for convex functions

- *Initialization:* $x_0 \in \mathbb{R}^N$, $x_{-1} = x_0$, $\varepsilon > 0$, $\alpha \geq 3$.
- *Iterations* $(n \geq 0)$*:* update x_n and y_n as follows:

$$\begin{cases} y_n = x_n + \frac{n}{n+\alpha}(x_n - x_{n-1}) \\ x_{n+1} = \mathrm{prox}_{\frac{1}{L}h}(y_n - \frac{1}{L}\nabla f(y_n)) \end{cases}$$

until $\|g(x_n)\| \leq \varepsilon$ i.e. until an ε-solution is reached.

Convergence rate analysis for a given $\varepsilon > 0$.

- How to get bounds in finite time on $F(x_n) - F^*$?
- Interpretation in terms of ε-solution:
  - Since:

$$\forall x \in \mathbb{R}^N,\ \frac{1}{2L}\|g(x)\|^2 \leqslant F(x) - F^*,$$

x_n is an ε solution if $F(x_n) - F^* \leqslant \frac{1}{2L}\varepsilon^2$.

General methodology to analyze optimization algorithms

- Interpreting the optimization algorithm as a discretization of a given ODE:

Gradient descent iteration: $$\frac{x_{n+1} - x_n}{s} + \nabla F(x_n) = 0$$

Associated ODE: $$\dot{x}(t) + \nabla F(x(t)) = 0.$$

- Analysis of ODEs using a Lyapunov approach:

$$\mathcal{E}(t) = F(x(t)) - F^*.$$

$$\mathcal{E}(t) = t(F(x(t)) - F^*) + \frac{1}{2}\|x(t) - x^*\|^2.$$

- Building a sequence of discrete Lyapunov energies adapted to the optimization scheme to get the same decay rates

$$\mathcal{E}(t) = F(x(t)) - F^*.$$

1. $\mathcal{E}$ is a Lyapunov energy (i.e. non increasing along the trajectories $x(t)$):

$$\mathcal{E}'(t) = \langle \nabla F(x(t)), \dot{x}(t) \rangle = -\|\nabla F(x(t))\|^2 \leqslant 0$$

hence:

$$\forall t \geqslant t_0,\ F(x(t)) - F^* \leqslant F(x_0) - F^*$$

$$\mathcal{E}(t) = F(x(t)) - F^*.$$

1. $\mathcal{E}$ is a Lyapunov energy (i.e. non increasing along the trajectories $x(t)$):

$$\mathcal{E}'(t) = \langle \nabla F(x(t)), \dot{x}(t) \rangle = -\|\nabla F(x(t))\|^2 \leqslant 0$$

hence:

$$\forall t \geqslant t_0,\ F(x(t)) - F^* \leqslant F(x_0) - F^*$$

2. Assume now that F is additionally μ-strongly convex. Then:

$$\forall y \in \mathbb{R}^N,\ \|\nabla F(y)\|^2 \geqslant 2\mu(F(y) - F^*),$$

hence:

$$\mathcal{E}'(t) \quad = \quad -\|\nabla F(x(t))\|^2 \leqslant -2\mu(F(x(t)) - F^*) \leqslant -2\mu\mathcal{E}(t)$$

and we deduce:

$$\forall t \geqslant t_0,\ F(x(t)) - F^* \leqslant (F(x_0) - F^*)e^{-2\mu(t-t_0)}.$$

Gradient descent for strongly convex functions

$$\mathcal{E}_n = F(x_n) - F^* \qquad \text{with:} \quad x_{n+1} = x_n - s\nabla F(x_n).$$

$$\begin{aligned}
\mathcal{E}_{n+1} - \mathcal{E}_n &= F(x_{n+1}) - F(x_n) \leqslant \langle \nabla F(x_n), x_{n+1} - x_n \rangle + \frac{L}{2}\|x_{n+1} - x_n\|^2 \\
&\leqslant -s\left(1 - \frac{L}{2}s\right)\|\nabla F(x_n)\|^2
\end{aligned}$$

- If $s < \frac{2}{L}$ then the GD is a descent algorithm: $\forall n,\ F(x_{n+1}) \leq F(x_n)$.

Gradient descent for strongly convex functions

$$\mathcal{E}_n = F(x_n) - F^* \qquad \text{with:} \quad x_{n+1} = x_n - s\nabla F(x_n).$$

$$\begin{aligned}\mathcal{E}_{n+1} - \mathcal{E}_n &= F(x_{n+1}) - F(x_n) \leqslant \langle \nabla F(x_n), x_{n+1} - x_n \rangle + \frac{L}{2}\|x_{n+1} - x_n\|^2 \\ &\leqslant -s\left(1 - \frac{L}{2}s\right)\|\nabla F(x_n)\|^2\end{aligned}$$

- If $s < \frac{2}{L}$ then the GD is a descent algorithm: $\forall n,\ F(x_{n+1}) \leq F(x_n)$.
- Assume that F is additionally μ-strongly convex:

$$\forall n,\ \|\nabla F(x_n)\|^2 \geqslant 2\mu(F(x_n) - F^*) = 2\mu\mathcal{E}_n,$$

$$\text{hence:}\ \ \mathcal{E}_{n+1} - \mathcal{E}_n \leqslant -2\mu s\left(1 - \frac{L}{2}s\right)\mathcal{E}_n.$$

For example if $s = \frac{1}{L}$ we get:

$$\forall n,\ \mathcal{E}_{n+1} - \mathcal{E}_n \leqslant -\frac{\mu}{L}\mathcal{E}_n \quad \Rightarrow \quad \mathcal{E}_n \leqslant \left(1 - \frac{\mu}{L}\right)^n \mathcal{E}_0$$

hence: $F(x_n) - F^* \leqslant (F(x_0) - F^*)\left(1 - \frac{\mu}{L}\right)^n$.

Discretization of an ODE, Su Boyd and Candès (15)

The scheme defined by

$$x_{n+1} = y_n - s\nabla F(y_n) \text{ with } y_n = x_n + \frac{n}{n+\alpha}(x_n - x_{n-1})$$

can be written

$$x_{n+1} - 2x_n + x_{n-1} + \frac{\alpha}{n}(x_{n+1} - x_n) + h\frac{n+\alpha}{n}\nabla F(y_n) = 0.$$

This can be seen as a semi-implicit discretization of a solution of

$$\ddot{x}(t) + \frac{\alpha}{t}\dot{x}(t) + \nabla F(x(t)) = 0, \tag{ODE}$$

with $\dot{x}(t_0) = 0$. Move of a solid in a potential field with a vanishing viscosity $\frac{\alpha}{t}$.

(Discretization step: $\delta = \sqrt{s}$ and $x_n \simeq x(n\sqrt{s})$)

Discretization of an ODE, Su Boyd and Candès (15)

The scheme defined by

$$x_{n+1} = y_n - s\nabla F(y_n) \text{ with } y_n = x_n + \frac{n}{n+\alpha}(x_n - x_{n-1})$$

can be seen as a semi-implicit discretization of a solution of

$$\ddot{x}(t) + \frac{\alpha}{t}\dot{x}(t) + \nabla F(x(t)) = 0, \tag{ODE}$$

with $\dot{x}(t_0) = 0$. Move of a solid in a potential field with a vanishing viscosity $\frac{\alpha}{t}$.

Advantages of the continuous setting

1. A simpler Lyapunov analysis, better insight
2. Optimality of bounds

Let $F : \mathbb{R}^N \to \mathbb{R}$ be a differentiable convex function and $x^* \in \arg\min(F) \neq \emptyset$.

- If $\alpha \geqslant 3$,

$$F(x(t)) - F(x^*) = \mathcal{O}\left(\frac{1}{t^2}\right)$$

[Attouch, Chbani, Peypouquet, Redont 2016]

- If $\alpha > 3$, then $x(t)$ cv to a minimizer of F and:

$$F(x(t)) - F(x^*) = o\left(\frac{1}{t^2}\right)$$

[Su, Boyd, Candes 2016]
[Chambolle, Dossal 2015]
[May 2017]

- If $\alpha < 3$ then no proof of cv of $x(t)$ but:

$$F(x(t)) - F(x^*) = \mathcal{O}\left(\frac{1}{t^{\frac{2\alpha}{3}}}\right)$$

[Attouch, Chbani, Riahi 2019]
[Aujol, Dossal 2017]

Nesterov, Proof of the convergence rate $\mathcal{O}\left(\frac{1}{t^2}\right)$ under convexity

We define:

$$\mathcal{E}(t) = t^2(F(x(t)) - F(x^*)) + \frac{1}{2}\left\|(\alpha-1)(x(t)-x^*) + t\dot{x}(t)\right\|^2.$$

Using (ODE), a straightforward computation shows that:

$$\begin{aligned}
\mathcal{E}'(t) &= -(\alpha-1)t \underbrace{\langle \nabla F(x(t)), x(t)-x^* \rangle}_{\geqslant F(x(t))-F(x^*) \text{ by convexity}} +2t(F(x(t))-F(x^*)) \\
&\leqslant (3-\alpha)t(F(x(t)-F(x^*)).
\end{aligned}$$

1. If $\alpha \geqslant 3$, $\forall t \geqslant t_0$, $t^2(F(x(t)) - F(x^*)) \leqslant \mathcal{E}(t_0)$.
2. If $\alpha > 3$, $\displaystyle\int_{t=t_0}^{+\infty}(\alpha-3)t(F(x(t)-F(x^*))dt \leqslant \mathcal{E}(t_0)$.

If F is convex and if $\alpha \geqslant 3$, the solution of (ODE) satisfies

$$F(x(t)) - F(x^*) = \mathcal{O}\left(\frac{1}{t^2}\right)$$

Let $F : \mathbb{R}^N \to \mathbb{R}$ be a differentiable convex function with $X^* := \arg\min(F) \neq \emptyset$.

$$\left| \begin{array}{lcl} y_n & = & x_n + \frac{n}{n+\alpha}(x_n - x_{n-1}) \\ x_{n+1} & = & y_n - \frac{1}{L}\nabla F(y_n) \end{array} \right. , \quad \alpha > 0$$

- If $\alpha \geqslant 3$

$$F(x_n) - F(x^*) = \mathcal{O}\left(\frac{1}{n^2}\right)$$

[Attouch, Peypouquet 2016]

- If $\alpha > 3$, then $(x_n)_{n\geqslant 1}$ cv and:

$$F(x_n) - F(x^*) = o\left(\frac{1}{n^2}\right)$$

[Chambolle, Dossal 2015]
[Attouch, Peypouquet 2015]

- If $\alpha \leqslant 3$

$$F(x_n) - F(x^*) = \mathcal{O}\left(\frac{1}{n^{\frac{2\alpha}{3}}}\right).$$

[Attouch, Chbani, Riahi 2018]
[Apidopoulos, Aujol, Dossal 2018]

Convergence rate analysis in finite time

Sketch of proof

$$\mathcal{E}(t) = t^2(F(x(t)) - F(x^*)) + \frac{1}{2}\left\|\lambda(x(t)-x^*) + t\dot{x}(t)\right\|^2, \quad \lambda = \frac{2\alpha}{3}.$$

Assume that F satisfies a quadratic growth condition and admits a unique minimizer.

1. Prove some differential inequation:

$$\forall t \geqslant t_0,\ \mathcal{E}'(t) + \frac{\lambda-2}{t}\mathcal{E}(t) \leqslant \varphi(t)\mathcal{E}(t).$$

2. Integrate it between any $t_1 \geqslant t_0$ and t:

$$\forall t \geqslant t_1,\ \mathcal{E}(t) \leqslant \mathcal{E}(t_1)\left(\frac{t_1}{t}\right)^{\lambda-2} e^{\phi(t_1)}.$$

3. Choose t_1 such that the previous inequality is as tight as possible:

$$\forall t \geqslant t_1,\ F(x(t)) - F^* \leqslant C_1 e^{\frac{2}{3}C_2(\alpha-3)}\left(\frac{\alpha}{t\sqrt{\mu}}\right)^{\frac{2\alpha}{3}}.$$

Convergence rate analysis in finite time

Optimize α to get a fast exponential decay

Let ε be a given accuracy. Let us make some rough calculations:

- For any $\alpha > 3$, we have:

$$\left(\frac{\alpha}{t\sqrt{\mu}}\right)^{\frac{2\alpha}{3}} \leqslant \varepsilon^2 \iff t \geqslant \frac{\alpha}{\sqrt{\mu}}\left(\frac{1}{\varepsilon}\right)^{\frac{3}{\alpha}}$$

$\hookrightarrow$ Polynomial decay.

Convergence rate analysis in finite time

Optimize α to get a fast exponential decay

Let ε be a given accuracy. Let us make some rough calculations:

- For any $\alpha > 3$, we have:

$$\left(\frac{\alpha}{t\sqrt{\mu}}\right)^{\frac{2\alpha}{3}} \leqslant \varepsilon^2 \iff t \geqslant \frac{\alpha}{\sqrt{\mu}}\left(\frac{1}{\varepsilon}\right)^{\frac{3}{\alpha}}$$

  $\hookrightarrow$ Polynomial decay.

- Choose now:

$$\alpha = C \log\left(\frac{1}{\varepsilon}\right).$$

  Then

$$\left(\frac{\alpha}{t\sqrt{\mu}}\right)^{\frac{2\alpha}{3}} \leqslant \varepsilon^2 \iff t \geqslant \frac{Ce^{\frac{3}{C}}}{\sqrt{\mu}} \log\left(\frac{1}{\varepsilon}\right)$$

  $\hookrightarrow$ Fast exponential decay !

Convergence rate analysis in finite time [ADR 2021]

FISTA for composite optimization with a quadratic growth condition

Theorem

Let $\varepsilon > 0$ *and*

$$\alpha_\varepsilon := 3\log\left(\frac{3\sqrt{LM_0}}{e\sqrt{2}\varepsilon}\right) \quad \textit{where:} \quad M_0 = F(x_0) - F^*.$$

Let $(x_n)_{n\in\mathbb{R}^N}$ *be a sequence of iterates generated by the FISTA algorithm with parameter* α_ε*. Then for* $\kappa = \frac{\mu}{L}$ *small enough, an* ε*-solution is reached in at most:*

$$n_\varepsilon^{FISTA} := \frac{8e^2}{3\sqrt{\kappa}}\alpha_\varepsilon = \frac{8e^2}{\sqrt{\kappa}}\log\left(\frac{3\sqrt{LM_0}}{e\sqrt{2}\varepsilon}\right)$$

iterations.

- α_ε does not depend on μ or any estimation of μ.
- n_ε^{FISTA} depends on the real value of μ.
- Fast exponential decay (we have turned a polynomial decay $\mathcal{O}\left(\frac{1}{n^{2\alpha/3}}\right)$ into an exponential one).

Comparison with Forward-Backward

Forward-Backward algorithm to minimize $F = f + h$

- *Initialization:* $x_0 \in \mathbb{R}^N$, $\varepsilon > 0$.
- *Iterations $(n \geq 0)$:* update x_n as follows:

$$x_{n+1} = \mathrm{prox}_{\frac{1}{L}h}(x_n - \frac{1}{L}\nabla f(x_n))$$

until $\|g(x_n)\| \leqslant \varepsilon$.

Let $\varepsilon > 0$. For $\kappa = \frac{\mu}{L}$ small enough,

$$n_\varepsilon^{FISTA} \leqslant n_\varepsilon^{FB}$$

where:

$$\begin{aligned} n_\varepsilon^{FB} &= \frac{1}{|\log(1-\kappa)|}\log\left(\frac{2LM_0}{\varepsilon^2}\right) \sim \frac{1}{\kappa}\log\left(\frac{2LM_0}{\varepsilon^2}\right) \\ n_\varepsilon^{FISTA} &= \frac{4e^2}{\sqrt{\kappa}}\log\left(\frac{9LM_0}{2e^2\varepsilon^2}\right) \quad \text{with} \quad \alpha = 3\log\left(\frac{3\sqrt{LM_0}}{e\sqrt{2}\varepsilon}\right) \end{aligned}$$

Comparison with Nesterov for strongly convex functions

Nesterov accelerated algorithm for strongly convex functions

- *Initialization:* $x_0 \in \mathbb{R}^N$, $x_{-1} = x_0$.
- *Iterations* $(n \geq 0)$*:* update x_n and y_n as follows:

$$\begin{cases} y_n = x_n + \dfrac{1-\sqrt{\kappa}}{1+\sqrt{\kappa}}(x_n - x_{n-1}) \\ x_{n+1} = \mathrm{prox}_{\frac{1}{L}h}(x_n - \frac{1}{L}\nabla f(x_n)) \end{cases}$$

until $\|g(x_n)\| \leq \varepsilon$.

Let $\varepsilon > 0$. If μ is known, for $\kappa = \frac{\mu}{L}$ small enough, NSC is faster than FISTA.

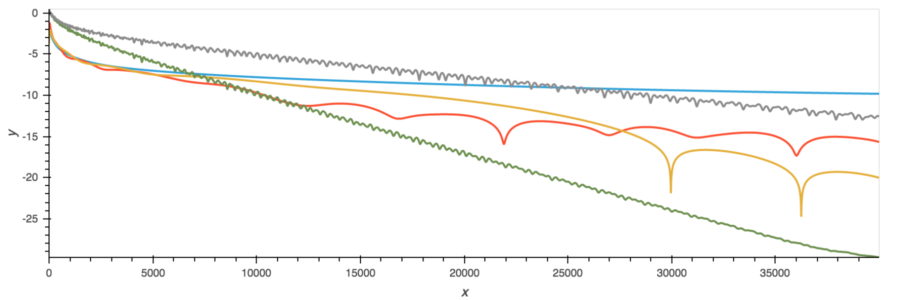
Comparison with Nesterov for strongly convex functions

But if μ is not perfectly known and for $\tilde{\mu} \leqslant \mu$

$$
\begin{aligned}
n_\varepsilon^{NSC} = \frac{1}{\left|\log(1-\sqrt{\frac{\tilde{\mu}}{L}})\right|}\log\left(\frac{4LM_0}{\varepsilon^2}\right) \quad & \geqslant \quad \frac{1}{|\log(1-\sqrt{\kappa})|}\log\left(\frac{4LM_0}{\varepsilon^2}\right) \\
& \sim \quad \frac{1}{\sqrt{\kappa}}\log\left(\frac{4LM_0}{\varepsilon^2}\right)
\end{aligned}
$$

In practice, FISTA may outperform NSC even for smaller underestimations of μ.

Comparisons



$\log(\|g(x_k)\|)$ along the iterations

FB, FISTA with $\alpha = 8$, FISTA with $\alpha = 30$,

NSC with the true value of μ, NSC with $\widetilde{\mu} = \frac{\mu}{10}$.

To sum up

- The version of FISTA proposed by Chambolle Dossal (2015) and Su Boyd Candès (2016) can reach an ε-solution with at most

$$\mathcal{O}\left(\sqrt{\frac{L}{\mu}}\log\left(\frac{1}{\varepsilon}\right)\right) \text{ iterations.}$$

  when the friction coefficient α is chosen as:

$$\alpha = 3\log\left(\frac{3}{e\sqrt{2}\varepsilon}\sqrt{L(F(x_0)-F^*)}\right).$$

- No need to estimate the growth parameter μ and the convergence rate does not suffer from an underestimation of μ.

  J-F Aujol, Ch. Dossal, A. Rondepierre, FISTA is an automatic geometrically optimized algorithm for strongly convex functions, Mathematical Programming 2023.

To sum up

	Geometry of F	References	Convergence rate for $F(x_n) - F^*$	Number of iterations to reach an ε solution
FB	Convex	*N84*, *BT09*	$\frac{2L\|x_0 - x^*\|^2}{n}$	$\frac{4L^2}{\varepsilon^2}\|x_0 - x^*\|^2$
FISTA with $\alpha = 3$	Convex	*N84*, *BT09*	$\frac{2L\|x_0 - x^*\|^2}{(n+1)^2}$	$\frac{2L}{\varepsilon}\|x_0 - x^*\|$
FB	Convex and $\mathcal{G}^2_\mu$	*Garrigos* 17	$(1+\kappa)^{-n}(F(x_0) - F^*)$	$\mathcal{O}\left(\frac{1}{\kappa}\log\left(\frac{1}{\varepsilon}\right)\right)$
NSC	Strongly convex Requires estimate of μ	*Nesterov* 13	$2(1-\sqrt{\kappa})^n(F(x_0) - F^*)$	$\mathcal{O}\left(\frac{1}{\sqrt{\kappa}}\log\left(\frac{1}{\varepsilon}\right)\right)$
FISTA $\alpha \geqslant 3$	Convex and $\mathcal{G}^2_\mu$ Uniqueness of minimizer	*Attouch* 18 *ADR19*	$\mathcal{O}\left(n^{-\frac{2\alpha}{3}}\right)$	Unknown
FISTA $\alpha = 3\log\left(\frac{5\sqrt{LM_0}}{e\,\varepsilon}\right)$	Convex and $\mathcal{G}^2_\mu$ Uniqueness of minimizer	*ADR21*	$\mathcal{O}\left(e^{-Cn\sqrt{\kappa}}\right)$	$\mathcal{O}\left(\frac{1}{\sqrt{\kappa}}\log\left(\frac{1}{\varepsilon}\right)\right)$

Outline

This last part is related to works within the PhD of Hippolyte Labarrière.

About inertia

Recall the definition of FISTA (for $\alpha = 3$)) to minimize $F = f + h$:

$$\forall k > 0, \begin{cases} x_k = prox_{\frac{1}{L}h}\left(y_{k-1} - \frac{1}{L}\nabla f(y_{k-1})\right) \\ y_k = x_k + \frac{k-1}{k+2}(x_k - x_{k-1}) \end{cases}$$

or if $h = 0$ and thus $f = F$,

$$\forall k > 0, \begin{cases} x_k = y_{k-1} - \frac{1}{L}\nabla F(y_{k-1}) \\ y_k = x_k + \frac{k-1}{k+2}(x_k - x_{k-1}) \end{cases}$$

$\rightarrow$ taking in account the previous iterates generates inertia.

Restarting FISTA, why?

- to take advantage of inertia,
- to avoid oscillations.

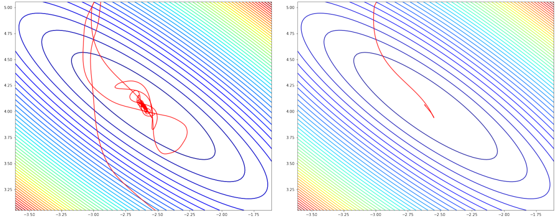


Figure: Trajectory of the iterates of FISTA (left) and FISTA restart (right) for a least-squares problem ($N = 20$).

Restarting FISTA, how?

Algorithm 1 : FISTA restart

Require: $x_0 \in \mathbb{R}^N,\ y_0 = x_0,\ k = 0,\ i = 0.$
  repeat
    $k = k + 1, i = i + 1$
    $x_k = prox_{\frac{1}{L}h}\left(y_{k-1} - \frac{1}{L}\nabla f(y_{k-1})\right)$
    if Restart condition is *True* **then**
      $i = 1$
    end if
    $y_k = x_k + \frac{i-1}{i+2}(x_k - x_{k-1})$
  until Exit condition is *True*

$\rightarrow$ Cutting inertia is equivalent to restarting the algorithm from the last iterate.

Minimize a composite convex function $F = f + h$ that satisfies a μ quadratic growth condition around its set of minimizers in $\mathbb{R}^N$.
f is convex with L Lipshitz gradient.
h is a convex lower semi-continuous function.

$$\kappa := \frac{\mu}{L} = o(1)$$

Objective: get a restart condition that

- does not require to know the growth parameter μ,
- ensures a fast convergence of the method: $F(x_k) - F^* = \mathcal{O}(e^{-K\sqrt{\frac{\mu}{L}}k})$,
- is not computationnaly expensive,
- is easy to implement.

Empiric FISTA restart (O'Donoghue and Candès, 2015, Beck and Teboulle, 2009)

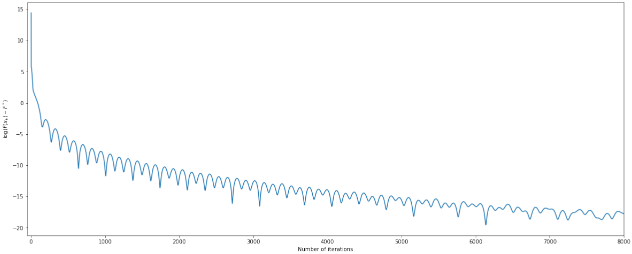
Restart under some exit condition

- on F:

$$F(x_k) > F(x_{k-1}),$$

- on ∇F:

$$\langle \nabla F(x_k), x_k - x_{k-1} \rangle > 0.$$



Fixed FISTA restart (Necoara et al., 2017)

Restart every k^* iterations where k^* is defined according to the growth parameter μ. If $k^* = \left\lfloor 2e\sqrt{\frac{L}{\mu}} \right\rfloor$:

$$F(x_k) - F^* = \mathcal{O}\left(e^{-\frac{1}{e}\sqrt{\frac{\mu}{L}}k}\right).$$

Restart strategies

Fixed FISTA restart (Necoara et al., 2017)

Restart every k^* iterations where k^* is defined according to the growth parameter μ. If $k^* = \left\lfloor 2e\sqrt{\frac{L}{\mu}} \right\rfloor$:

$$F(x_k) - F^* = \mathcal{O}\left(e^{-\frac{1}{e}\sqrt{\frac{\mu}{L}}k}\right).$$

Adaptive FISTA restart (Alamo et al., 2019, Fercoq and Qu, 2019)

Restart according to the geometry of F and previous iterations.

- Adaptive restart by Alamo et al.: $F(x_k) - F^* = \mathcal{O}\left(e^{-\frac{1}{16}\sqrt{\frac{\mu}{L}}k}\right)$.
- Adaptive restart by Fercoq and Qu:

$$F(x_k) - F^* = \mathcal{O}\left(e^{-\frac{\sqrt{2}-1}{2\sqrt{e}\left(2-\sqrt{\frac{\mu}{\mu_0}}\right)}\sqrt{\frac{\mu}{L}}k}\right).$$

Strategy of our scheme:

- to estimate the growth parameter μ at each restart,
- to adapt the number of iterations of the following restart according to this estimation.
- to stop the algorithm when the exit condition $\|\nabla g(r_j)\| \leqslant \varepsilon$ is satisfied.

Algorithm 2 : Automatic FISTA restart

Require: $r_0 \in \mathbb{R}^N, j = 1$

$n_0 = \lfloor 2C \rfloor$

$r_1 = \mathrm{FISTA}(r_0, n_0)$

$n_1 = \lfloor 2C \rfloor$

repeat

$j = j + 1$

$r_j = \mathrm{FISTA}(r_{j-1}, n_{j-1})$

$$\tilde{\mu}_j = \min_{\substack{i \in \mathbb{N}^* \\ i<j}} \frac{4L}{(n_{i-1}+1)^2} \frac{F(r_{i-1}) - F(r_j)}{F(r_i) - F(r_j)}$$

Estimate of the parameter μ.

if $n_{j-1} \leqslant C\sqrt{\frac{L}{\tilde{\mu}_j}}$ **then**

$n_j = 2n_{j-1}$

Update of the number of iterations per restart.

end if

until $\|g(r_j)\| \leqslant \varepsilon$

Why it works

Lemma

If (x_k) is generated with FISTA, we have

$$F(x_k) - F^* \leqslant \frac{4L}{\mu(k+1)^2} \left(F(x_0) - F^*\right).$$

Hence

$$\forall j \in \mathbb{N}^*,\ \mu \leqslant \frac{4L}{(n_{j-1}+1)^2} \frac{F(r_{j-1}) - F^*}{F(r_j) - F^*}.$$

But in fact, we can even show:

$$\forall j \in \mathbb{N}^*,\ \mu \leqslant \frac{4L}{(n_{j-1}+1)^2} \frac{F(r_{j-1}) - F(r_{j+1})}{F(r_j) - F(r_{j+1})}.$$

Hence the definition of $\tilde{\mu}$ in Algorithm 1.

Lemma

The sequence $(n_j)_{j\in\mathbb{N}}$ provided by Algorithm 2 satisfies $n_j \leqslant 2C\sqrt{\frac{L}{\mu}}$.

Theorem (Aujol, Dossal, Labarrière, Rondepierre, 2021)

If F satisfies the assumptions stated before and $C > 4$, then

$$F(r_j^+) - F^* = \mathcal{O}\left(e^{-\frac{\log\left(\frac{C^2}{4}-1\right)}{4C}\sqrt{\frac{\mu}{L}}\sum_{i=0}^{j} n_i} \right).$$

Let $C = 6.38$, then

$$F(r_j^+) - F^* = \mathcal{O}\left(e^{-\frac{1}{12}\sqrt{\frac{\mu}{L}}\sum_{i=0}^{j} n_i} \right).$$

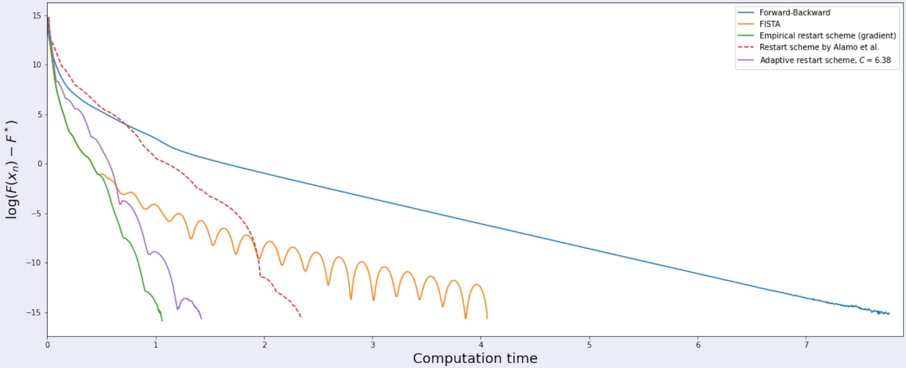
Image inpainting:

$$\min_x F(x) := \frac{1}{2}\|Mx - y\|^2 + \lambda\|Tx\|_1,$$

where M is a mask operator and T is an orthogonal transformation ensuring that Tx^0 is sparse.

Restart strategies

Image inpainting:



Attenuating oscillations introducing Hessian-driven damping

Hessian-driven damping

(DIN-AVD) system (**Attouch, Peypouquet and Redont, 2016**)

$$\ddot{x}(t) + \frac{\alpha}{t}\dot{x}(t) + \beta H_F(x(t))\dot{x}(t) + \nabla F(x(t)) = 0.$$

- Attenuation of the oscillations through the introduction of a geometry-driven damping term.

Attenuating oscillations introducing Hessian-driven damping

Integrability properties

- **Attouch, Peypouquet and Redont, 2016**: if F is convex and C^2, $\alpha \geqslant 3$ and $\beta > 0$:

$$\int_{t_0}^{+\infty} t^2 \|\nabla F(x(t))\|^2 dt < +\infty,$$

- **Aujol, Dossal, Hoàng, Labarrière and Rondepierre, 2022**: if F is convex and C^2, satisfies $\mathcal{G}_\mu^2$ and has a unique minimizer. Then, for $\alpha \geqslant 3$ and $\beta > 0$:

$$\int_{t_0}^{+\infty} t^{\alpha-\varepsilon} \|\nabla F(x(t))\|^2 dt < +\infty, \forall \varepsilon \in (0,1).$$

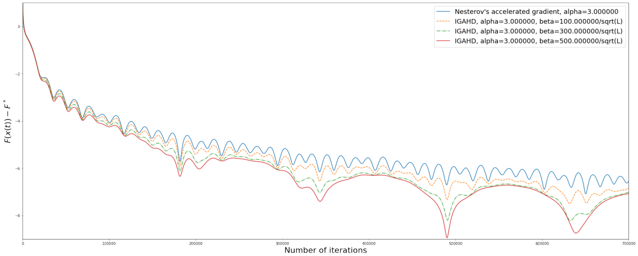
Attenuating oscillations introducing Hessian-driven damping

Derivating a numerical scheme: IGAHD (Attouch, Chbani, Fadili and Riahi, 2020)

$$\ddot{x}(t) + \frac{\alpha}{t}\dot{x}(t) + \beta H_F(x(t))\dot{x}(t) + \left(1 + \frac{\beta}{t}\right)\nabla F(x(t)) = 0.$$

$$\downarrow$$

$$\begin{cases} x_k = y_{k-1} - s\nabla F(y_{k-1}), \\ y_k = x_k + \dfrac{k-1}{k+\alpha-1}(x_k - x_{k-1}) - \beta\sqrt{s}(\nabla F(x_k) - \nabla F(x_{k-1})) - \dfrac{\beta\sqrt{s}}{k}\nabla F(x_{k-1}), \end{cases}$$



Attenuating oscillations introducing Hessian-driven damping

Summary

The Hessian-driven damping term is a **physical way** to attenuate oscillations. As this is a relatively recent subject of research, there are some limitations:

- the behavior of the numerical schemes derivated from (DIN-AVD) is not fully understood (current convergence rates hold if β is **small**),
- the dependency in β is not known,
- there is no proof showing that it is faster than classical inertial schemes.

Conclusion/To sum up

	Geometry of F	References	Convergence rate for $F(x_n) - F^*$	Number of iterations to reach an ε solution
FB	Convex	*N84*, *BT09*	$\frac{2L\|x_0 - x^*\|^2}{n}$	$\frac{4L^2}{\varepsilon^2}\|x_0 - x^*\|^2$
FISTA with $\alpha = 3$	Convex	*N84*, *BT09*	$\frac{2L\|x_0 - x^*\|^2}{(n+1)^2}$	$\frac{2L}{\varepsilon}\|x_0 - x^*\|$
FB	Convex and $\mathcal{G}^2_\mu$	*Garrigos* 17	$(1+\kappa)^{-n}(F(x_0) - F^*)$	$\mathcal{O}\left(\frac{1}{\kappa}\log\left(\frac{1}{\varepsilon}\right)\right)$
NSC	Strongly convex Requires estimatate of μ	*Nesterov* 13	$2(1-\sqrt{\kappa})^n(F(x_0) - F^*)$	$\mathcal{O}\left(\frac{1}{\sqrt{\kappa}}\log\left(\frac{1}{\varepsilon}\right)\right)$
FISTA $\alpha \geqslant 3$	Convex and $\mathcal{G}^2_\mu$ Uniqueness of minimizer	*Attouch* 18 *ADR19*	$\mathcal{O}\left(n^{-\frac{2\alpha}{3}}\right)$	Unknown
FISTA $\alpha = 3\log\left(\frac{5\sqrt{LM_0}}{e\,\varepsilon}\right)$	Convex and $\mathcal{G}^2_\mu$ Uniqueness of minimizer	*ADR21*	$\mathcal{O}\left(e^{-Cn\sqrt{\kappa}}\right)$	$\mathcal{O}\left(\frac{1}{\sqrt{\kappa}}\log\left(\frac{1}{\varepsilon}\right)\right)$
Optimal FISTA restart	Strongly convex Requires estimate of μ	*Necoara* 19	$\mathcal{O}\left(e^{-\frac{1}{e}\sqrt{\kappa}n}\right)$	$\mathcal{O}\left(\frac{1}{\sqrt{\kappa}}\log\left(\frac{1}{\varepsilon}\right)\right)$
FISTA restart	Convex and $\mathcal{G}^2_\mu$	*Aujol etal21*	$\mathcal{O}\left(e^{-\frac{1}{12}\sqrt{\kappa}n}\right)$	$\mathcal{O}\left(\frac{1}{\sqrt{\kappa}}\log\left(\frac{1}{\varepsilon}\right)\right)$

Next step $\Longrightarrow$ remove the convexity assumption on F (new Lyapounov functions, ...).