FISTA is an automatic geometrically optimized algorithm for strongly convex functions

Jean-François Aujol, Charles Dossal, Aude Rondepierre

Institut de Mathématiques de Toulouse, INSA de Toulouse & LAAS-CNRS

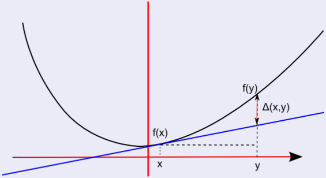
Séminaire Français d'Optimisation - Jeudi 10 mars 2022

The setting: composite optimization

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

where:

- f is a convex differentiable function 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 ($\min_{x \in \mathbb{R}^N} \frac{1}{2}\|Ax - b\|^2 + \|x\|_1$)

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

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

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 $F^* = \min F$.

- Relaxation of strong convexity.
- Equivalent (in the convex setting) to a global version of the Łojasiewicz property with an exponent $\frac{1}{2}$.

Let $\varepsilon > 0$. The minimizers of a composite function F 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: $\text{prox}_{\gamma h}(x) = \arg\min_{y \in \mathbb{R}^n} \gamma h(y) + \frac{1}{2}\|y - x\|^2$.

Definition (ε-solution)

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

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

where: $g(x) := L\left(x - \text{prox}_{\gamma h}(x - \frac{1}{L}\nabla f(x))\right)$ 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]

- x_k is an ε-solution of $\min_{x\in\mathbb{R}^n} F(x)$ if:

$$F(x_k) - F^* \leqslant \frac{1}{2L}\varepsilon^2.$$

2. $\forall x \in \mathbb{R}^n,\ F(x^+) - F^* \leqslant \frac{2}{\mu}\|g(x)\|^2$ [Aujol Dossal Labarrière R. 2021]

A tractable stopping criterion

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

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

Forward-Backward algorithm
Definition

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

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

$$x_0 \in \mathbb{R}^n$$
$$x_{k+1} = prox_{\gamma h}(x_k - \gamma \nabla f(x_k)), \quad \gamma > 0.$$

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

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

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

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

Hence:

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

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

$$x_{k+1} = x_k - \frac{1}{L}\nabla f(x_k)$$

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):

$$x_{k+1} = x_k - \frac{1}{L}\nabla f(x_k)$$

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):

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

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):

$$x_{k+1} = x_k - \frac{1}{L}\nabla f(x_k)$$

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):

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

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) ($h = \|\cdot\|_1$):

$$x_{k+1} = prox_{\frac{1}{L}h}(x_k - \frac{1}{L}\nabla f(x_k))$$

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

Assume that F is convex. Then:

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

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

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

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

$$
\begin{aligned}
y_k &= x_k + \frac{t_k - 1}{t_{k+1}}(x_k - x_{n-1}) \\
x_{k+1} &= prox_{\frac{1}{L}h}\left(y_k - \frac{1}{L}\nabla f(y_k))\right).
\end{aligned}
$$

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

$$t_{k+1}^2 - t_{k+1} = t_k^2.$$

For the class of convex functions, they prove:

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

but they do not prove the convergence of the iterates.

FISTA a fast proximal gradient method

FISTA - Chambolle Dossal 2015, Su Boyd Candès 2016

Let $\alpha \geqslant 3$.

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

- Initially Nesterov (1984) proposes $\alpha = 3$.
- For the class of composite convex functions:

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

and Chambolle Dossal prove the weak convergence of the iterates.

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

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

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

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

Convergence rate for FISTA [Su Boyd Candès 2015], [Attouch Cabot 2017].

Assume additionally that F has a unique minimizer.

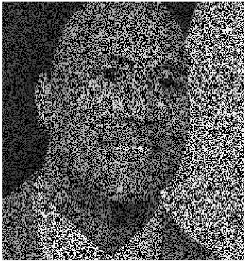
$$\forall \alpha > 0, \ \forall k \in \mathbb{N}, \ F(x_k) - F^* = \mathcal{O}\left(k^{-\frac{2\alpha}{3}}\right)$$

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

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

where M is a random masking operator and T an orthogonal wavelet transform.

target x^o

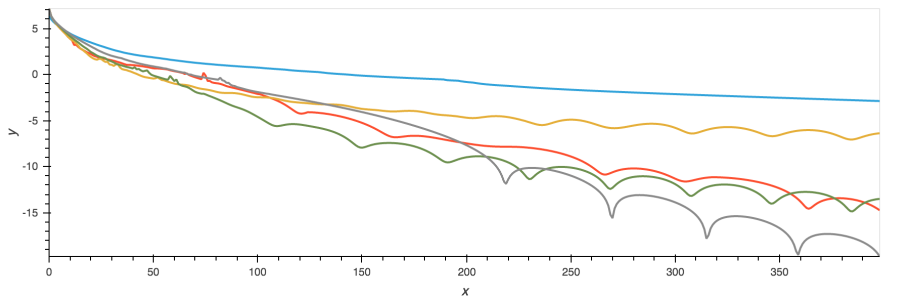


masked image Mx^o

solution x^*

FB vs FISTA in the strongly convex case

Exponential rate vs Polynomial rate (3/3)



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

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

$$y_k = x_k + \frac{1-\sqrt{\kappa}}{1+\sqrt{\kappa}}(x_k - x_{n-1})$$
$$x_{k+1} = y_k - \tfrac{1}{L}\nabla F(y_k)$$

Theorem (Theorem 2.2.3, Nesterov 2013)

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

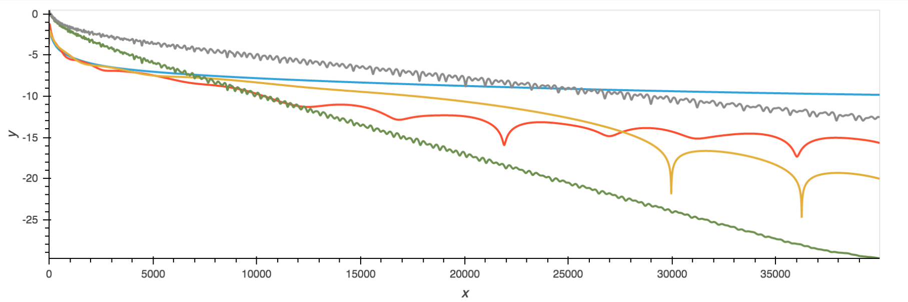
$$\forall n \in \mathbb{N},\ F(x_k) - F(x^*) \leqslant 2(1-\sqrt{\kappa})^n \left(F(x_0) - F(x^*)\right),$$

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

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

The iterations require an estimation of $\kappa = \frac{\mu}{L}$!

FISTA in the strongly convex case
Differentiable case

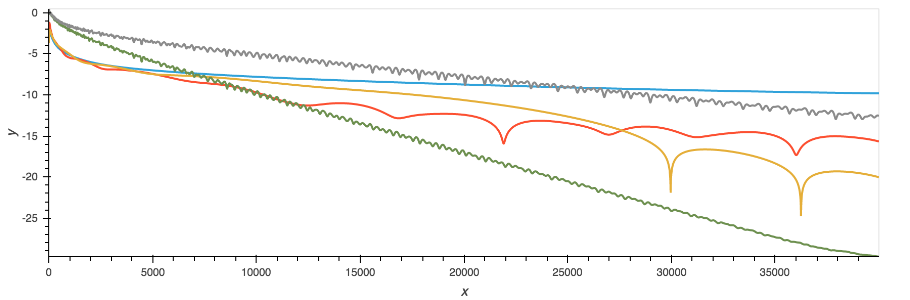


$\log(\|g(x_k)\|)$ along the iterations

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

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

FISTA in the strongly convex case
Differentiable case



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

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

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

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_k and y_k as follows:

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

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

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

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

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

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

General methodology to analyze optimization algorithms

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

$$\text{Gradient descent iteration:} \quad \frac{x_{k+1} - x_k}{h} + \nabla F(x_k) = 0$$

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

- 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

Illustration for the gradient descent method

A Lyapunov analysis of the ODE $\dot{x}(t) + \nabla F(x(t)) = 0$

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

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:

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

$$\mathcal{E}_k = F(x_k) - F^* \qquad \text{with:} \quad x_{k+1} = x_k - h\nabla F(x_k).$$

$$\begin{aligned}
\mathcal{E}_{k+1} - \mathcal{E}_k &= F(x_{k+1}) - F(x_k) \leqslant \langle \nabla F(x_k), x_{k+1} - x_k \rangle + \frac{L}{2}\|x_{k+1} - x_k\|^2 \\
&\leqslant -h\left(1 - \frac{L}{2}h\right)\|\nabla F(x_k)\|^2
\end{aligned}$$

- If $h < \frac{2}{L}$ then the GD is a descent algorithm: $\forall k, F(x_{k+1}) < F(x_k)$.

Gradient descent for strongly convex functions

From the continuous to the discrete

$$\mathcal{E}_k = F(x_k) - F^* \qquad \text{with:} \quad x_{k+1} = x_k - h\nabla F(x_k).$$

$$\begin{aligned} \mathcal{E}_{k+1} - \mathcal{E}_k &= F(x_{k+1}) - F(x_k) \leqslant \langle \nabla F(x_k), x_{k+1} - x_k \rangle + \frac{L}{2}\|x_{k+1} - x_k\|^2 \\ &\leqslant -h\left(1 - \frac{L}{2}h\right)\|\nabla F(x_k)\|^2 \end{aligned}$$

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

  $$\forall k,\ \|\nabla F(x_k)\|^2 \geqslant 2\mu(F(x_k) - F^*) = 2\mu\mathcal{E}_k,$$

  hence:

  $$\mathcal{E}_{k+1} - \mathcal{E}_k \leqslant -2\mu h\left(1 - \frac{L}{2}h\right)\mathcal{E}_k.$$

  For example si $h = \frac{1}{L}$ we get:

  $$\forall k,\ \mathcal{E}_{k+1} - \mathcal{E}_k \leqslant -\frac{\mu}{L}\mathcal{E}_k \quad \Rightarrow \quad \mathcal{E}_k \leqslant (1 - \frac{\mu}{L})^k\mathcal{E}_0$$

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

The scheme defined by

$$x_{k+1} = y_k - h\nabla F(y_k) \text{ with } y_k = x_n + \frac{n}{n+\alpha}(x_n - x_{n-1})$$

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 $X^* := \arg\min(F) \neq \emptyset$.

$$\left| \begin{array}{ccl} y_k & = & x_k + \dfrac{k}{k+\alpha}(x_k - x_{k-1}) \\ x_{k+1} & = & y_k - h\nabla F(y_k) \end{array} \right. , \quad \alpha > 0$$

- If $\alpha \geqslant 3$
  $$F(x_k) - F(x^*) = \mathcal{O}\left(\frac{1}{k^2}\right)$$
  [Attouch, Peypouquet 2016]
- If $\alpha > 3$, then $(x_k)_{k\geqslant 1}$ cv and:
  $$F(x_k) - F(x^*) = o\left(\frac{1}{k^2}\right)$$
  [Chambolle, Dossal 2015]
  [Attouch, Peypouquet 2015]
- If $\alpha \leqslant 3$
  $$F(x_k) - F(x^*) = \mathcal{O}\left(\frac{1}{k^{\frac{2\alpha}{3}}}\right).$$
  [Attouch, Chbani, Riahi 2018]
  [Apidopoulos, Aujol, Dossal 2018]

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

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 $t_1 \leqslant t_0$ 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 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:

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

$\hookrightarrow$ 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:

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

$\hookrightarrow$ Polynomial decay.

- Choose now:

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

Then

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

$\hookrightarrow$ Fast exponential decay !

Theorem

Let $\varepsilon > 0$ *and*

$$\alpha_{1,\varepsilon} := 3\log\left(\frac{5\sqrt{LM_0}}{e\varepsilon}\right) \qquad \textit{where:} \qquad M_0 = F(x_0) - F^*. \tag{2}$$

Let $(x_k)_{n\in\mathbb{R}^N}$ *be a sequence of iterates generated by the FISTA algorithm with parameter* $\alpha_{1,\varepsilon}$*. Then for* $\kappa = \frac{\mu}{L}$ *small enough, an* ε*-solution is reached in at most:*

$$n_{1,\varepsilon}^{FISTA} := \frac{8e^2}{3\sqrt{\kappa}}\alpha_{1,\varepsilon} = \frac{8e^2}{\sqrt{\kappa}}\log\left(\frac{5\sqrt{LM_0}}{e\varepsilon}\right) \tag{3}$$

iterations.

- $\alpha_{1,\varepsilon}$ does not depend on μ or any estimation of μ !
- $n_{1,\varepsilon}^{FISTA}$ depends on the real value of μ.
- Fast exponential decay.

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_k as follows:

$$x_{k+1} = \text{prox}_{\frac{1}{L}h}(x_k - \frac{1}{L}\nabla f(x_k)) \tag{4}$$

until $\|g(x_k)\| = \|x_{k+1} - x_k\| \leqslant \varepsilon$.

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

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

where:

$$\begin{aligned} n_\varepsilon^{FB} &= \frac{1}{|\log(1-\kappa)|}\log\left(\frac{2LM_0}{\varepsilon^2}\right) \sim \frac{1}{\kappa}\log\left(\frac{2LM_0}{\varepsilon^2}\right) \\ n_\varepsilon^{FISTA} &= \frac{4e^2}{\sqrt{\kappa}}\log\left(\frac{5LM_0}{e^2\varepsilon^2}\right) \quad \text{with} \quad \alpha = 3\log\left(\frac{5\sqrt{LM_0}}{e\varepsilon}\right) \end{aligned}$$

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_k and y_k as follows:

$$\begin{cases} y_k = x_k + \dfrac{1-\sqrt{\kappa}}{1+\sqrt{\kappa}}(x_k - x_{n-1}) \\ x_{k+1} = y_k - \frac{1}{L}\nabla F(y_k) \end{cases} \tag{5}$$

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

Let $\varepsilon > 0$. If μ is known, for $\kappa = \frac{\mu}{L}$ small enough, NSC is faster than FISTA. But if μ is not perfectly known and for $\tilde{\mu} \leqslant \mu$

$$n_\varepsilon^{NSC} = \frac{1}{\left|\log(1-\sqrt{\frac{\tilde{\mu}}{L}})\right|}\log\left(\frac{4LM_0}{\varepsilon^2}\right) \geqslant \frac{1}{|\log(1-\sqrt{\kappa})|}\log\left(\frac{4LM_0}{\varepsilon^2}\right) \tag{6}$$

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

Conclusion/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:

$$\alpha = 3\log\left(\frac{5}{e\varepsilon}\sqrt{L(F(x_0)-F^*)}\right).$$

- 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.R. FISTA is an automatic geometrically optimized algorithm for strongly convex functions. 2021. ⟨hal-03491527⟩