Multilevel Proximal Methods for Image Restoration

Guillaume Lauga [1], Elisa Riccietti [1], Nelly Pustelnik [2], Paulo Gonçalves [1]

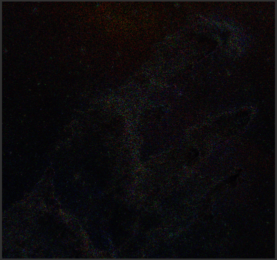
[1] Univ Lyon, Inria, Ens de Lyon, Univ Lyon 1, CNRS, LIP, Lyon, France.

[2] Ens de Lyon, CNRS, Laboratoire de Physique, F-69342, Lyon, France.

SIAM CSE March 1st 2023

Our goal: Recovering $\hat{x}$ as close as possible as to the original image $\bar{x}$ from a degraded observation z

$\bar{x}$



z

$\widehat{x}$

Context: Restoration of large-scale images ($N > 10^6$ variables)

Problem formulation:

Classic degradation model: $z = \mathrm{A}\bar{x} + \epsilon$

- $\mathrm{A} \in \mathbb{R}^{M\times N}$ a linear degradation
- $\epsilon \in \mathbb{R}^{M}$ some gaussian noise

Ill-posed problem, it leads to the following minimization problem:

$$\widehat{x} \in \underset{x\in\mathbb{R}^N}{\mathrm{Argmin}} \frac{1}{2}\|\mathrm{A}x - z\|_2^2 + \lambda\|\mathrm{D}x\|_1$$

- N image size
- $\mathrm{D} \in \mathbb{R}^{K\times N}$ linear transform on x from which we will seek sparsity
- $\lambda > 0$ regularization parameter

Problem formulation:

Classic degradation model: $z = \mathrm{A}\bar{x} + \epsilon$

- $\mathrm{A} \in \mathbb{R}^{M \times N}$ a linear degradation
- $\epsilon \in \mathbb{R}^{M}$ some gaussian noise

Ill-posed problem, it leads to the following minimization problem:

$$\min_{x \in \mathbb{R}^N} F(x) := \underbrace{f(\mathrm{A}x)}_{\text{data fidelity}} + \underbrace{g(\mathrm{D}x)}_{\text{regularization}}$$

f and g proper, lower semi continuous and convex. f is assumed differentiable.

Motivations

In a multilevel setting we want to:

- tackle high-dimensional optimization problems
- tackle non-smooth optimization
- include inexact proximal steps to handle SOTA regularization: Total Variation (TV) and Non-Local Total Variation (NLTV) based semi-norm

In this context we provide IML FISTA a *convergent multilevel* inexact and inertial proximal gradient algorithm that works when:

- g is non-smooth
- the proximity operator of g is explicit (cf. `http://proximity-operator.net/index.html`)
- the proximity operator of $g(\mathrm{D}\cdot)$ is not known under closed form

Framework

Inertial convergent proximal algorithm that can handle inexact steps:

$$x_{k+1} \approx_{\epsilon_k} \mathrm{prox}_{\tau g \circ \mathrm{D}} \left(y_k - \tau \nabla f(\mathrm{A} y_k)\right)$$
$$y_{k+1} = x_{k+1} + \alpha_k (x_{k+1} - x_k)$$

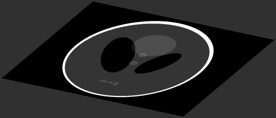
where α_k is chosen as in [Aujol and Dossal, 2015].

Our idea (inspired by [Parpas, 2016-2017]): update y_k through a multilevel step.

How to construct such multilevel update ?
How to guarantee convergence ?

Classical scheme for two levels

Goal: Exploit a hierarchy of approximations of the objective function.
Two levels case: fine (h) and coarse (H)

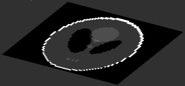


$$\underset{x_h \in \mathbb{R}^{N_h}}{\text{Argmin}}\ F_h(x_h)$$

$$\downarrow$$

$$y_{h,k}$$

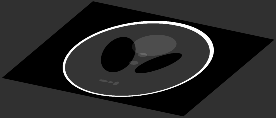
$$\xrightarrow{I_h^H}$$



$$\underset{x_H \in \mathbb{R}^{N_H}}{\text{Argmin}}\ F_H$$

$$x_{H,k,0} \longrightarrow x_{H,k,m}$$

$$\xrightarrow{I_H^h}$$



$$\underset{x_h \in \mathbb{R}^{N_h}}{\text{Argmin}}\ F_h(x_h)$$

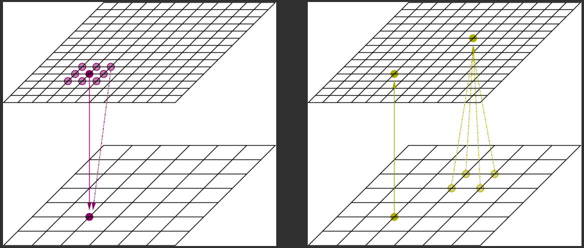
$$\uparrow$$

$$y_{h,k} + I_H^h(x_{H,k,m} - x_{H,k,0})$$

Ingredients

Information transfer operators

- $I_h^H \in \mathbb{R}^{N_H \times N_h}$: transfer from fine to coarse scales ($N_H < N_h$)
- $I_H^h \in \mathbb{R}^{N_h \times N_H}$: transfer from coarse to fine scales
- Coherence between operators: $I_H^h = \nu(I_h^H)^T$
- Example: multiresolution analysis (orthogonal wavelets)



I_h^H I_H^h

Ingredients

Objective function at coarse level:

$$f_H(\mathrm{A}_H \cdot) + g_H(\mathrm{D}_H \cdot)$$

Image restoration context

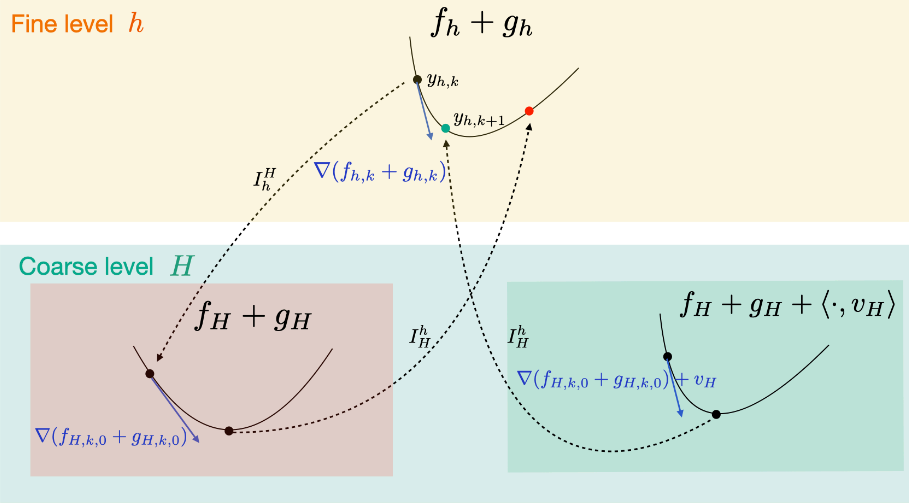
$$\forall x \in \mathbb{R}^{N_h} \quad f_h(x) = \frac{1}{2}\|x - z\|_2^2 \quad g_h(x) = \lambda\|x\|_1$$

$$\forall x \in \mathbb{R}^{N_H} \quad f_H(x) = \frac{1}{2}\|x - I_h^H z\|_2^2 \quad g_H(x) = \lambda\|x\|_1$$

where:

- A_H is a reduced order version of A_h (e.g. Galerkin approximation, decimation)
- D_H is a reduced order version of D_h (e.g. TV at fine and coarse levels)

First order coherence when g_h, g_H are smooth



Smoothing of F_h and F_H with the Moreau envelope

Moreau envelope of g_H: ${}^{\gamma}g_H = \inf_{y\in\mathcal{H}} g_H(y) + \frac{1}{2\gamma}\|\cdot - y\|^2$

Properties of the Moreau envelope:

- $\nabla^{\gamma} g_H = \gamma^{-1}(\mathrm{Id} - \mathrm{prox}_{\gamma g_H})$
- $\nabla^{\gamma} g_H$ γ^{-1} - Lipschitz

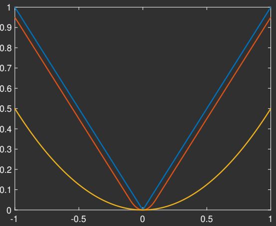


Figure: Moreau envelope of l_1-norm for $\gamma = 0.1$ and $\gamma = 1$

H

First order coherence for g non-smooth

Coarse model F_H for non-smooth functions

$$F_H = f_H + ({}^{\gamma_H} g_H \circ \mathrm{D}_H) + \langle v_H, \cdot \rangle$$

where ${}^{\gamma} g$ is the Moreau envelope of g
and the coherence term involves

$$\begin{aligned} v_H = & I_h^H \left(\nabla f_h(y_h) + \nabla({}^{\gamma_h} g_h \circ \mathrm{D}_h)(y_h) \right) \\ & - \left(\nabla f_H(I_h^H y_h) + \nabla({}^{\gamma_H} g_H \circ \mathrm{D}_H)(I_h^H y_h) \right) \end{aligned}$$

Thanks to Moreau envelope properties, we have:

$$\nabla \left({}^{\gamma_h} g_h \circ \mathrm{D}_h \right)(\cdot) = \gamma_h^{-1} \mathrm{D}_h^* \left(\mathrm{D}_h \cdot - \mathrm{prox}_{\gamma_h g_h}(\mathrm{D}_h \cdot) \right)$$

Decrease at fine level

With this first order coherence, any algorithm that decreases F_H implies

$$F_h(y_h + \bar{\tau} I_H^h(x_{H,m} - x_{H,0})) \leq F_h(y_h) + \eta\gamma_h$$

where :

- $\eta, \gamma_h > 0$ depend on the Moreau approximation
- $\bar{\tau}$ is some step size.

$\rightarrow$ Bound on one multilevel but not enough for global convergence guarantees

Our algorithm

for k **do**

  if *Descent condition and* $r < p$ **then**

  Coarse model (r = r+1):

  $x_{H,k,0} = I_h^H y_{h,k}$ Projection
  $x_{H,k,m} = \Phi_{H,k,m-1} \circ .. \circ \Phi_{H,k,0}(x_{H,k,0})$ Coarse minimization
  Set $\bar{\tau}_{h,k} > 0$, Update by coarse step
  $\bar{y}_{h,k} = y_{h,k} + \bar{\tau}_{h,k} I_H^h \left(x_{H,k,m} - x_{H,k,0}\right)$

  else

  $\bar{y}_{h,k} = y_{h,k}$

  end

  Fine level :

  $x_{h,k+1} \approx_{\epsilon_k} \text{FB}(\bar{y}_{h,k})$ Forward-backward step
  $t_{h,k+1} = \left(\frac{k+a}{a}\right)^d$, $\alpha_{h,k} = \frac{t_{h,k}-1}{t_{h,k+1}}$
  $y_{h,k+1} = x_{h,k+1} + \alpha_{h,k}(x_{h,k+1} - x_{h,k})$ Inertial step

end

Convergence of the algorithm

With the proximity operator approximated at each iteration, solving:

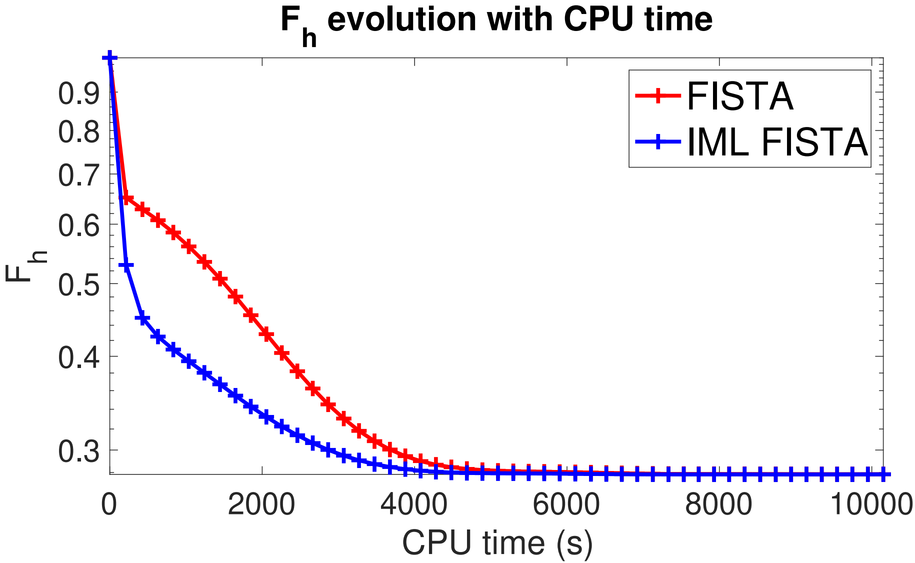
$$\mathrm{prox}_{\gamma g_h \circ \mathrm{D}_h}(y) \approx_\epsilon \widehat{u} \in \underset{u \in \mathbb{R}^K}{\arg\min} \frac{1}{2}\|\mathrm{D}_h^* u - y\|^2 + \gamma g_h^*(u)$$

Convergence theorem

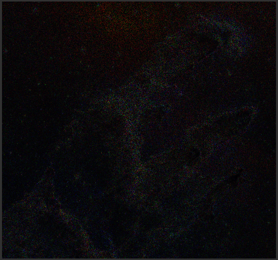
With at most p coarse corrections and summable errors to obtain the proximity operators:

- $(k^{2d}(F_h(x_{h,k}) - F_h(x^*)))_{k\in\mathbb{N}}$ belongs to $\ell_\infty(\mathbb{N})$
- $(x_{h,k})_{k\in\mathbb{N}}$ converges to a minimizer of F_h

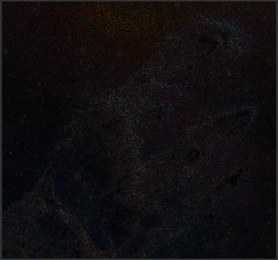
Evolution of F_h for a $N_h = 2048 \times 2048 \times 3$ image



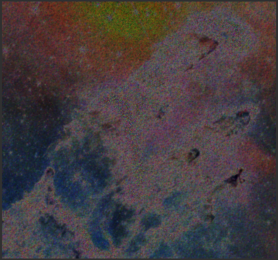
Reconstruction after 2 iterations with NLTV



z



x_2 FISTA

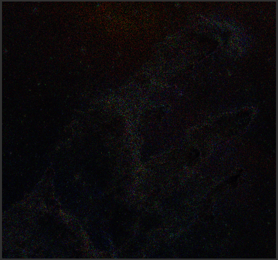


x_2 IML FISTA

$\widehat{x}$ estimated after 2 iterations of IML FISTA. Parameters: $\tau = 1$, $\lambda = 3e-2$, $\gamma = 1.1$, $l = 5$, $p = 2$, $m = 5$, inpainting: 90% along all channels, and $\sigma(\epsilon) = 5e-2$.

Reconstruction after 50 iterations with NLTV

$\bar{x}$



z

$\widehat{x}$

$\widehat{x}$ estimated after 50 iterations of IML FISTA. Parameters: $\tau = 1$, $\lambda = 3e - 2$, $\gamma = 1.1$, $l = 5$, $p = 2$, $m = 5$, inpainting: 90% along all channels, and $\sigma(\epsilon) = 5e - 2$.

Summary

We have developed a *convergent multilevel algorithm* with:

- same convergence rates as FISTA
- efficient construction of coarse models
- good experimental performances on large scale images
- embed-able state-of-the-art regularizations for inverse problems

Future work:

- Application to challenging large scale problems
- Better guarantees for the multilevel steps

References

- **Multilevel FISTA for Image Restoration, ICASSP 2023**
- **Méthodes proximales multi-niveaux pour la restauration d'images, GRETSI 2022**

Slides (and preprint soon) available at
`https://laugaguillaume.github.io`