On the Interplay between Acceleration and Identification for the Proximal Gradient algorithm

Gilles Bareilles, Franck Iutzeler

LJK, Univ. Grenoble Alpes

Journées SMAI MODE 2020
7-9 septembre 2020

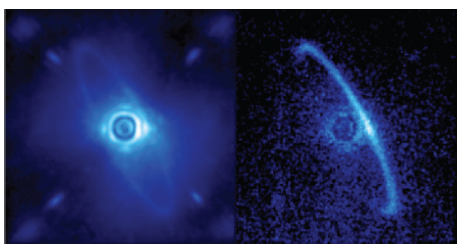
Linear inverse problems...

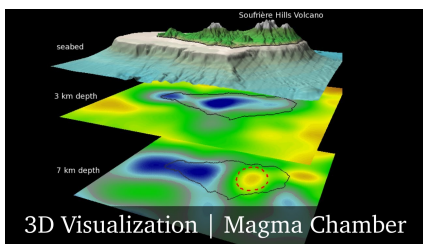
x_0 is the signal of interest, only accessible through measures y, via a (linear) forward model A and noise ξ:

$$y = Ax_0 + \xi$$

Examples

- Image processing diffraction of objective, low-res./damaged sensor
- Medical imaging computerized tomography, MRI, EEG
- Seismic imaging wave propagation
- Machine learning / Statistics regression





Linear inverse problems...

$$y = Ax_0 + \xi$$

... are often ill posed.
$\Rightarrow$ introduce ***prior knowledge*** on the ***structure*** of x_0.

How? For some $\lambda > 0$, some convex differentiable loss ℓ, we minimize the ***empirical risk*** penalized by some ***regularizer*** encoding prior information:

$$x^\star \in \arg\min_x \underbrace{\sum_{i=1}^n \ell(A_i x, y_i)}_{\text{smooth empirical risk}} + \lambda \underbrace{r(x)}_{\text{nonsmooth regularizer}} \qquad \text{(ERM)}$$

E.g.: if x_0 known to have many zero entries, $r : x \mapsto \|x\|_1 = \sum_{i=1}^n |x_i|$ is a common choice.

Prior knowledge ?

Prior knowledge	penalization function
sparsity	$r = \|\cdot\|_1$
group sparsity	$r = \|\cdot\|_{1-2}$
anti-sparsity	$r = \|\cdot\|_\infty$
low-rank	$r = \|\cdot\|_*$
unit norm	$r = \max(\|\cdot\|_p - 1, 0)$
$\vdots$	$\vdots$

and combinations...

◇ Vaiter, S., Peyré, G., Fadili, J.: Low Complexity Regularization of Linear Inverse Problems, chap. Sampling Theory, a Renaissance, pp. 103–153. Springer-Birkhäuser (2015)

In large scale cases, where problems are over determined, regularization improves statistical properties (generalization, ...). E.g. LASSO regression

Composite problems

$$\text{Find } x^\star \in \arg\min_{x\in\mathbb{R}^n} \quad \underbrace{f(x)}_{\text{smooth}} + \underbrace{g(x)}_{\text{non smooth}}$$

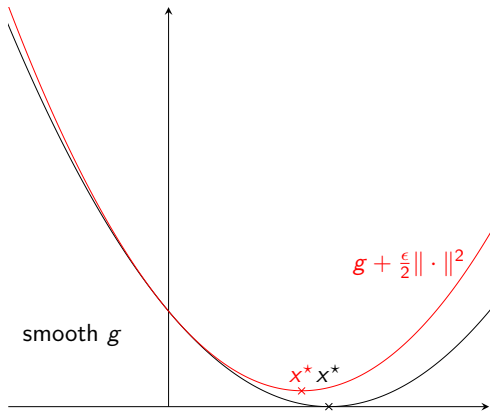
We assume f & g are convex, there exists a unique minimizer.

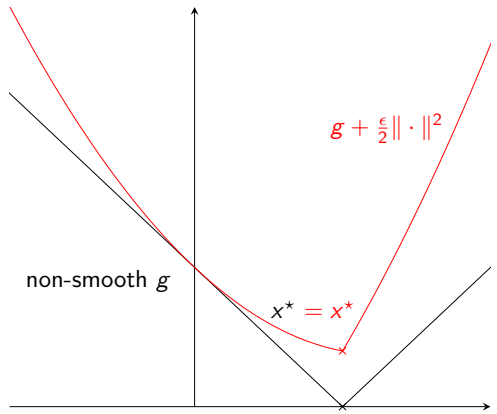
Questions

- 1. How to formalize structure? Why is it interesting?
- 2. Can optimization algorithms detect structure?
- 3. How well do algorithms detect structure in practice? Can they be improved?

What is structure? Robustness to perturbations

Let's perturb a smooth and a non-smooth function with a smooth one:





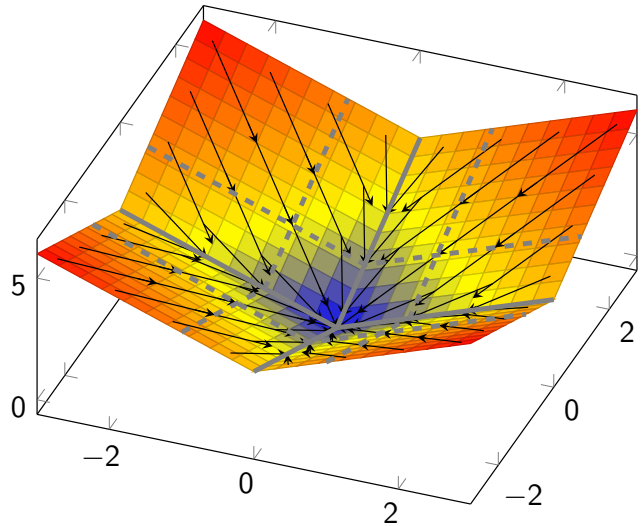
Optimality condition at point $x^\star$ of:

g	$0 = \nabla g(x^\star)$	$0 \in \partial g(x^\star)$
$g + \frac{\epsilon}{2}\|\cdot\|^2$	$\underbrace{0 = \nabla g(x^\star) + \epsilon x^\star}$	$\underbrace{0 \in \partial g(x^\star) + \epsilon x^\star}$
	$\epsilon \neq 0 \Rightarrow x^\star \neq x^\star$	$\partial g(x^\star) \neq \{\cdot\} \Rightarrow x^\star = x^\star$ is possible

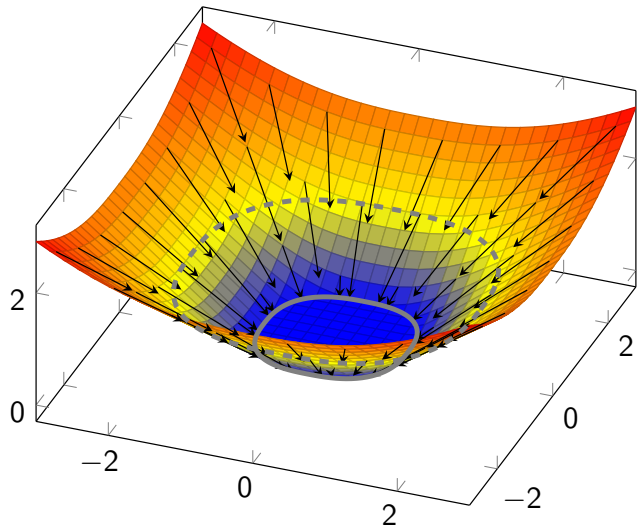
A *small enough* perturbation to this *nonsmooth* function leaves its minimizer unchanged.

g non diff. at $x^\star$ $\iff$ robustness to perturbations $\iff$ structure

Examples of structure



$g = \|\cdot\|_1$



$g = \max(\|\cdot\|_{2.6} - 1, 0)$

Structured points are exactly points of non-differentiability of g.

Here:

- for $g = \|\cdot\|_1$, structured points are the cartesian axes;
- for $g = \max(\|\cdot\|_{2.6} - 1, 0)$, structured subspace is $\{x : \|x\|_{2.6} = 1\}$.

- 1. How to formalize structure? Why is it interesting?
  - The non-smoothness of g imposes some structure on minimizers;
  - This structure formalizes as belonging or not to subspaces, where g is non-differentiable;
  - Structure matters statistically feature selection/reduction and numerically smaller problem

- 2. Can optimization algorithms detect structure?

Given a *composite problem* and an *optimization algorithm*, under which condition can we guarantee that its iterates reach the correct subspace in finite time?

- 3. How well do algorithms detect structure in practice? Can they be improved?

Step aside: what algorithms for composite minimization?

$$\text{Find } x^\star \in \operatorname*{arg\,min}_{x \in \mathbb{R}^n} f(x) + g(x)$$

The non-smooth g is handled via its ***proximity operator***:

$$\mathbf{prox}_{\gamma g}(u) := \operatorname*{arg\,min}_{w \in \mathbb{R}^n} \left\{ g(w) + \frac{1}{2\gamma} \|w - u\|^2 \right\}.$$

Closed form / easily computable for many regularizers: $\|\cdot\|_1$, $\|\cdot\|_*$, TV, ...

We thus look at algorithms that write as

$$\left| \begin{array}{l} u_k = \cdots \\ x_k = \mathbf{prox}_{\gamma g}(u_k) \end{array} \right. \qquad \text{(Prox-based alg.)}$$

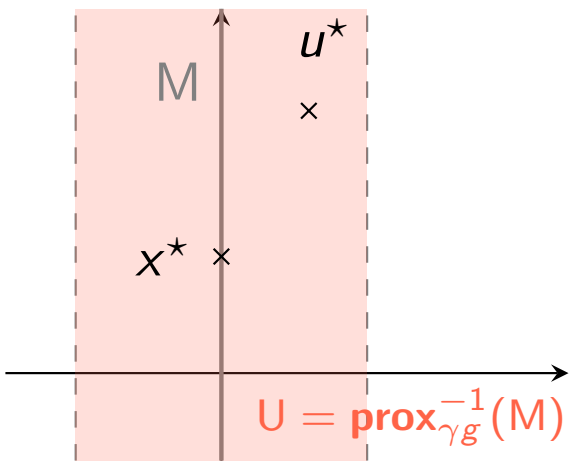
which includes proximal gradient aka ISTA, accelerated proximal gradient aka FISTA among others.

Identification: an example

Consider sequences (u_k), $(x_k = \mathbf{prox}_{\gamma g}(u_k))$ and a subspace M such that:

$$u_k \to u^\star \qquad x_k \to x^\star \triangleq \mathbf{prox}_{\gamma g}(u^\star) \qquad x^\star \in \mathrm{M}$$

Let's look at the set $\mathrm{U} = \mathbf{prox}_{\gamma g}^{-1}(\mathrm{M}) = (I + \gamma \partial g)(\mathrm{M})$:



$$g(x) = \|x\|_1 = \sum_{i=1}^n |x_i|$$

$$[\mathbf{prox}_{\gamma|\cdot|}(u)]_i = \begin{cases} u_i - \gamma & \text{if } u_i > \gamma \\ 0 & \text{if } -\gamma \le u_i \le \gamma \\ u_i + \gamma & \text{if } u_i < -\gamma \end{cases}$$

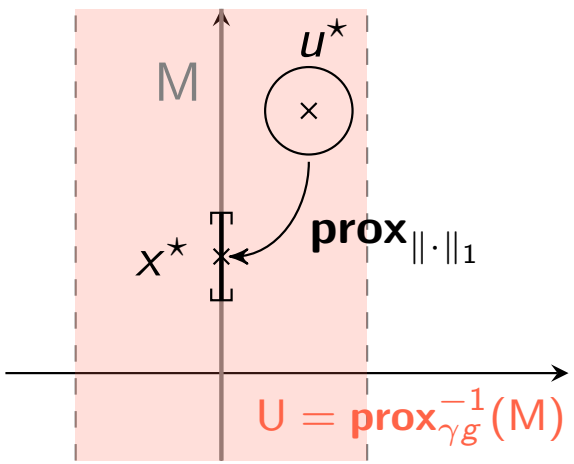
M is the y-axis.

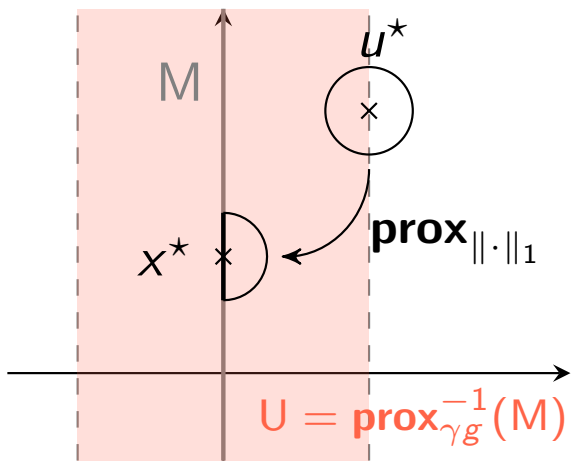
Identification: an example

Consider sequences (u_k), $(x_k = \mathbf{prox}_{\gamma g}(u_k))$ and a subspace M such that:

$$u_k \to u^\star \qquad x_k \to x^\star \triangleq \mathbf{prox}_{\gamma g}(u^\star) \qquad x^\star \in \mathrm{M}$$

Let's look at the set $\mathrm{U} = \mathbf{prox}_{\gamma g}^{-1}(\mathrm{M}) = (I + \gamma \partial g)(\mathrm{M})$:



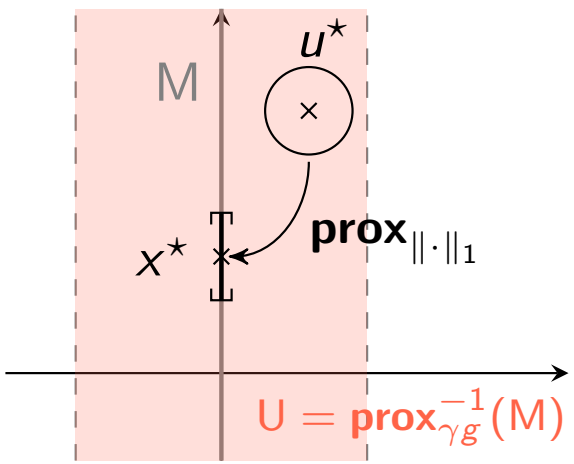


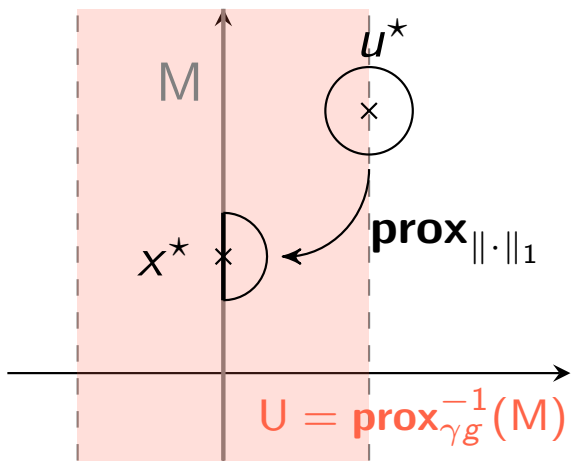
Identification: an example

Consider sequences (u_k), $(x_k = \mathbf{prox}_{\gamma g}(u_k))$ and a subspace M such that:

$$u_k \to u^\star \qquad x_k \to x^\star \triangleq \mathbf{prox}_{\gamma g}(u^\star) \qquad x^\star \in \mathrm{M}$$

Let's look at the set $\mathrm{U} = \mathbf{prox}_{\gamma g}^{-1}(\mathrm{M}) = (I + \gamma \partial g)(\mathrm{M})$:





Identification is ***ensured***: in finite time

$$u_k \in \mathcal{B}(u^\star, \epsilon) \Rightarrow x_k \in \mathrm{M}$$

Identification ***may not happen***, depending on the iterates trajectory.

Contribution: a sufficient condition for identification

Lemma (Identification)

Consider sequences (u_k), $(x_k = \mathbf{prox}_{\gamma g}(u_k))$ *and a manifold* M *such that:*

$$u_k \to u^\star \qquad x_k \to x^\star \triangleq \mathbf{prox}_{\gamma g}(u^\star) \qquad x^\star \in \mathrm{M}$$

If

$$\exists \varepsilon > 0 \text{ such that for all } u \in \mathcal{B}(u^\star, \varepsilon),\ \mathbf{prox}_{\gamma g}(u) \in \mathrm{M}, \qquad \text{(QC)}$$

then, ***after some finite time,*** $x_k \in \mathrm{M}$.

Thus,

- if the ***problem*** satisfies (QC), ***any*** converging prox-based algorithm will identify the minimizer structure;
- otherwise, no guarantee of recovering the minimizer structure.

Note: known in the partial smoothness theory.

$$\text{Partial smoothness} + 0 \in \mathrm{ri}\ (\partial g + \nabla f)(x^\star) \Rightarrow \text{(QC)}$$

- 1. How to formalize structure? Why is it interesting?
  - The non-smoothness of g imposes some structure on minimizers;
  - This structure formalizes as belonging or not to subspaces, where g is non-differentiable;
  - Structure matters statistically feature selection/reduction and numerically smaller problem
- 2. Can optimization algorithms detect structure?
  - if the ***problem*** satisfies (QC), any converging prox-based algorithm will identify the minimizer structure; otherwise, no guarantee of recovering the minimizer structure.

Theory is able to capture whether finite time identification is guaranteed or not for a given problem.

- 3. How well do algorithms detect structure ***in practice***? Can they be improved?

Prox-based algorithms: PG and APG

Proximal Gradient

$$\left|\begin{array}{l} u_{k+1} = x_k - \gamma \nabla f(x_k) \\ x_{k+1} = \mathbf{prox}_{\gamma g}(u_{k+1}) \end{array}\right.$$

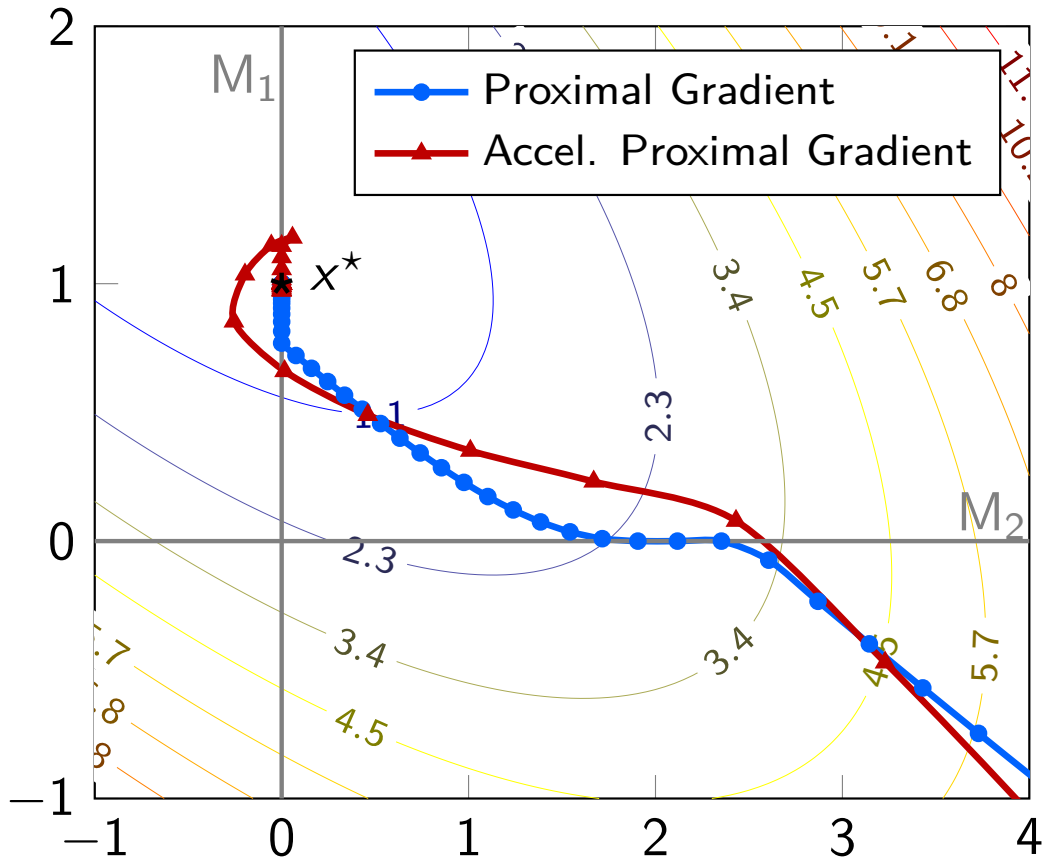
Accelerated proximal gradient

$$\left|\begin{array}{l} u_{k+1} = y_k - \gamma \nabla f(y_k) \\ x_{k+1} = \mathbf{prox}_{\gamma g}(u_{k+1}) \\ y_{k+1} = x_{k+1} + \underbrace{\alpha_{k+1}(x_{k+1} - x_k)}_{\text{inertia / extrapolation}} \end{array}\right.$$

	PG	APG
$F(x_k) - F^\star$	$\mathcal{O}(1/k)$	$\mathcal{O}(1/k^2)$
$\|x_k - x_{k-1}\|^2$	$\mathcal{O}(1/k)$	$\mathcal{O}(1/k^2)$
iterates convergence	yes	yes
monotone functional CV	yes	no
monotone iterates CV	yes	no

◇ Nesterov: A method for solving the convex programming problem with convergence rate $O(1/k^2)$. Dokladi A.N. Sssr (1983)
◇ Beck, Teboulle: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM journal on Imaging Sciences (2009)
◇ Chambolle, Dossal: On the convergence of the iterates of "FISTA". Journal of Optimization theory and Applications (2015)

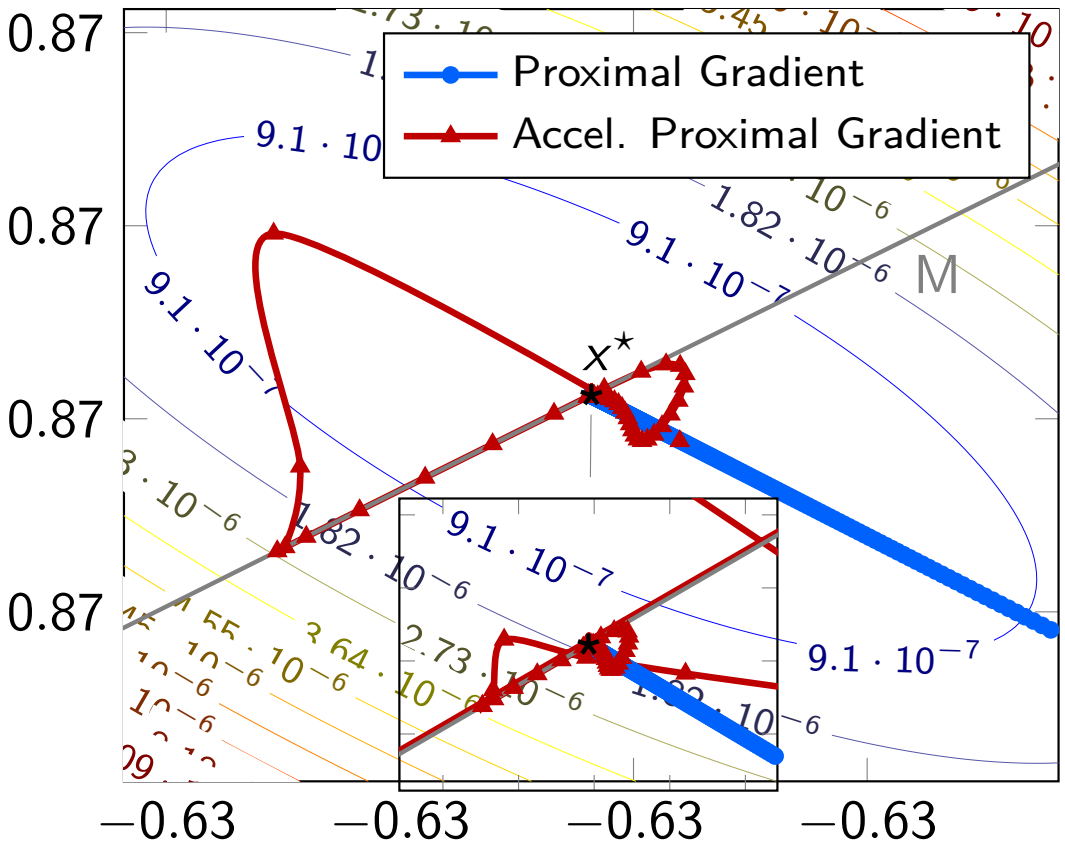
$$\min_{x \in \mathbb{R}^2} \|Ax - b\|_2^2 + \lambda r(x), \qquad \text{with } A \in \mathbb{R}^{2\times 2}$$



$r(x) = \|x\|_1$
$\mathrm{M}_1 = y$-axis;
$\mathrm{M}_2 = x$-axis
(two linear manifolds)

This problem is qualified.

$$\min_{x \in \mathbb{R}^2} \|Ax - b\|_2^2 + \lambda r(x), \qquad \text{with } A \in \mathbb{R}^{2\times 2}$$



$$r(x) = \max(0, \|x\|_{2.6} - 1)$$
$$\mathrm{M} = \mathcal{S}_{\|\cdot\|_{2.6}}(0, 1)$$

(one curved manifold)

This problem is ***not*** qualified.

Observations

Effect of acceleration on identification:

(i) ***Overshooting of to-be identified manifold*** ✗

(ii) ***Misfit of linear extrapolation with curved subspace*** ✗

(iii) ***Exploratory behavior of acceleration*** ✓

Could we get the best of both algorithms?

Contribution: a variant of APG

Each iteration, $\mathsf{T}_k \in \{0, 1\}$ decides whether to accelerate or not.

$$\left| \begin{array}{l} x_{k+1} = \mathbf{prox}_{\gamma g}(y_k - \gamma \nabla f(y_k)) \triangleq \mathcal{T}_\gamma(y_k) \\ y_{k+1} = \begin{cases} x_{k+1} + \alpha_{k+1}(x_{k+1} - x_k) & \text{if } \mathsf{T}_k = 1 \\ x_{k+1} & \text{if } \mathsf{T}_k = 0 \end{cases} \end{array} \right.$$

We want:

- $\mathsf{T}_k = 1$ asymptotically, to get the accelerated $\mathcal{O}(1/k^2)$ rate;
- $\mathsf{T}_k = 0$ only when acceleration is harmful.

For analysis reasons, we allow acceleration only when

$$\|\mathcal{T}_\gamma(y_{k-1}) - y_{k-1}\|^2 \leq \zeta \text{ and } F(\mathcal{T}_\gamma(y_{k-1})) \leq F(x_0)$$

Proposed tests

$$\left| \begin{array}{l} x_{k+1} = \mathbf{prox}_{\gamma g}(y_k - \gamma \nabla f(y_k)) \triangleq \mathcal{T}_\gamma(y_k) \\ y_{k+1} = \begin{cases} x_{k+1} + \alpha_{k+1}(x_{k+1} - x_k) & \text{if } \mathsf{T}_k = 1 \\ x_{k+1} & \text{if } \mathsf{T}_k = 0 \end{cases} \end{array} \right.$$

(i) "Overshooting of to-be identified manifold" No acceleration *i.e.* $\mathsf{T}_k^1 = 0$ when

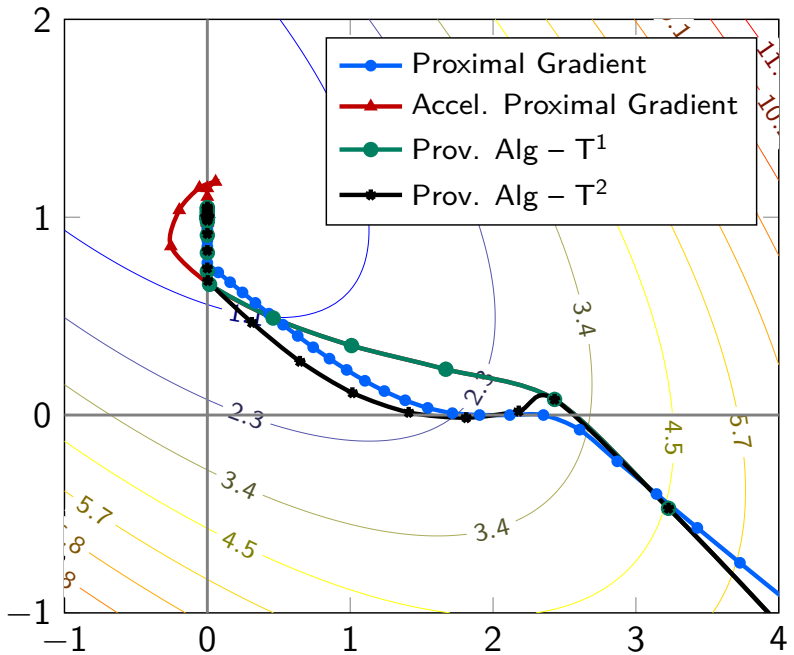
$$\begin{cases} x_k \notin \mathsf{M} \\ x_{k+1} \in \mathsf{M} \end{cases} \quad \text{for some } \mathsf{M} \in \mathcal{C}$$

(ii) "Misfit of linear extrapolation with curved subspace" No acceleration *i.e.* $\mathsf{T}_k^2 = 0$ when

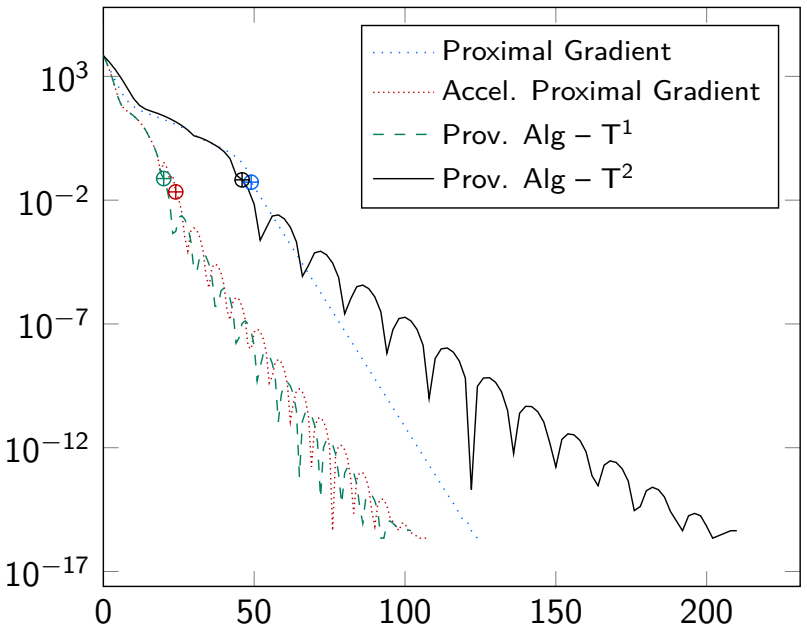
$$\begin{cases} \mathcal{T}_\gamma(x_{k+1}) \in \mathsf{M} \\ \mathcal{T}_\gamma(x_{k+1} + \alpha_{k+1}(x_{k+1} - x_k)) \notin \mathsf{M} \end{cases} \quad \text{for some } \mathsf{M} \in \mathcal{C}$$

Theorem (informal): We maintain the accelerated rate $\mathcal{O}(1/k^2)$ for qualified problems, proximal gradient rate $\mathcal{O}(1/k)$ otherwise.

Numerical experiments

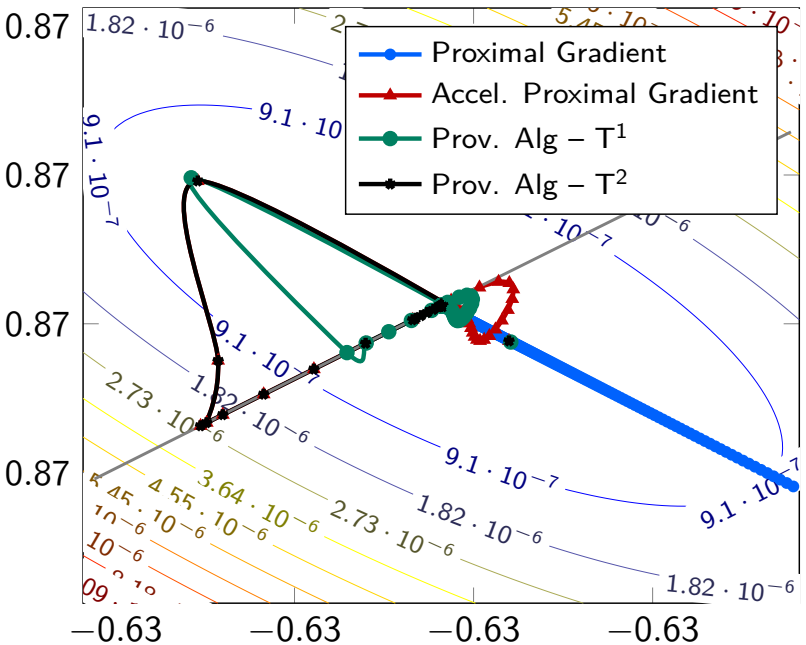


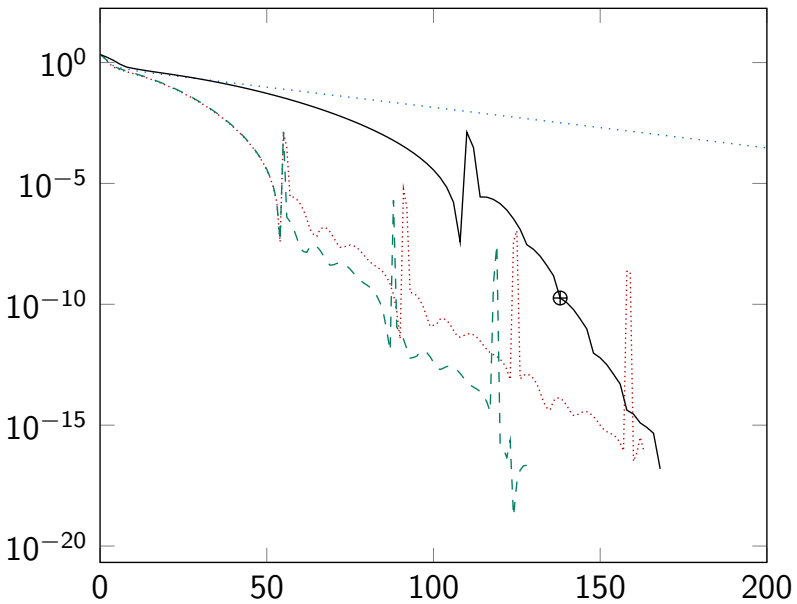
⊕ marks identification time.



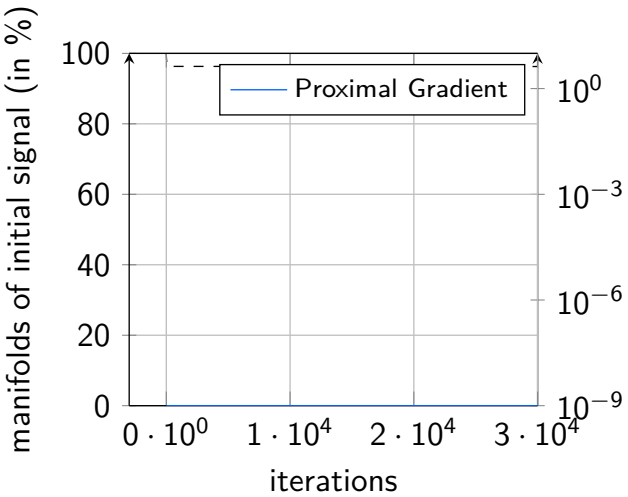
suboptimality vs. number of proximal gradient steps

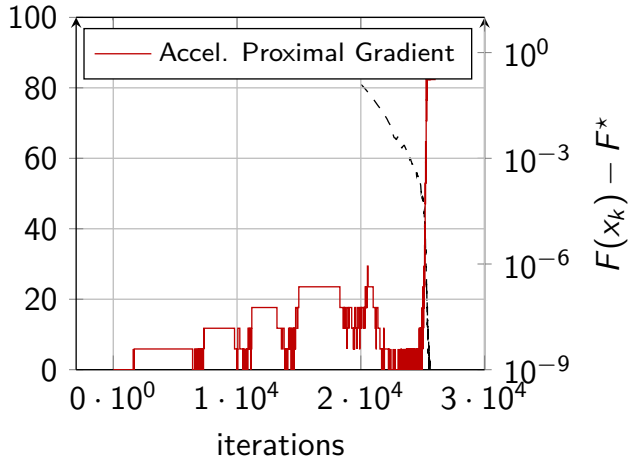
Numerical experiments

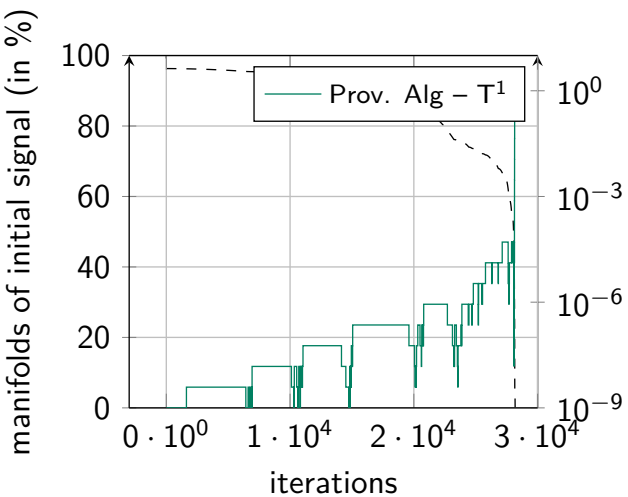


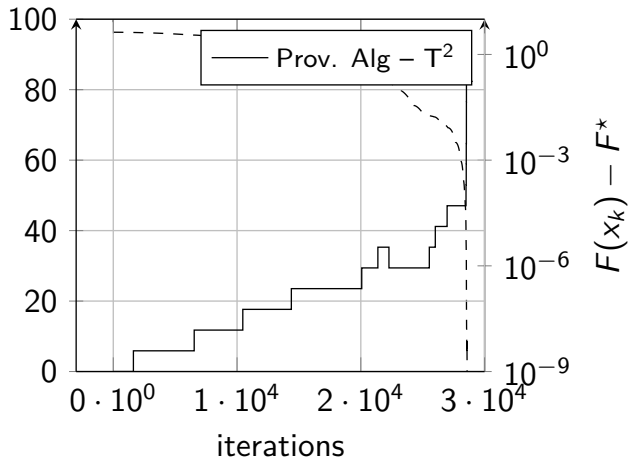


suboptimality vs. number of proximal gradient steps









$$\min_{x \in \mathbb{R}^n} \|Ax - b\|_2^2 + \lambda r(x) \quad \text{and } b = Ax_0 + e \in \mathbb{R}^{16^2},\ x_0 \in \mathbb{R}^{20^2}.$$

$$r = \|\cdot\|_* \quad - \quad x_0 \text{ is rank 3.}$$

Plots: 0% $\leftrightarrow$ rank 20; 100% $\leftrightarrow$ rank 3;

Take-home messages

- Proximal methods can identify the structure of composite problems;
- This structure bears value statistically feature selection and numerically dimensionality reduction;
- Acceleration may harm identification of proximal gradient... we proposed a more stable proximal algorithm with accelerated rate.

Perspectives

- Leverage the progressive identification to improve optimization algorithms;
- Identification with two non-smooth functions?

B. & Iutzeler: *On the Interplay between Acceleration and Identification for the Proximal Gradient algorithm*, Computational Optimization and Applications, 2020. `https://arxiv.org/abs/1909.08944`

Thank you!

Gilles Bareilles – `gbareilles.fr`