Forward-backward methods for convex and nonconvex optimization in imaging

Silvia Bonettini

Dipartimento di Scienze Fisiche,
Informatiche e Matematiche
Università di Modena e Reggio Emilia

Optimization Algorithms and Software
for Inverse problemS
www.oasis.unimore.it

CALCOLO SCIENTIFICO E MODELLI MATEMATICI: alla ricerca delle cose nascoste attraverso le cose manifeste

Roma, 6-9 Aprile 2022

Contents: theoretical convergence analysis and acceleration strategies for Forward-Backward methods.

Collaborators:

UNIVERSITA' di MODENA e REGGIO EMLIA	UNIVERSITA' di FERRARA	UNIVERSITA' di FIRENZE	UNIVERSITE' LIBRE de BRUXELLES
Luca Zanni	Valeria Ruggiero	Simone Rebegoldi	Ignace Loris
Marco Prato	Gaetano Zanighirati		
Federica Porta			

Main references:

- S. B., I. Loris, F. Porta, M. Prato 2016, Variable metric inexact line–search based methods for nonsmooth optimization, *SIAM J. Optim.*, **26**(2), 891-921
- S. B., I. Loris, F. Porta, M. Prato, S. Rebegoldi 2017, On the convergence of a line–search base proximal-gradient method for nonconvex optimization, *Inverse Probl.*, **33**(5), 055005
- S. B., M. Prato, S. Rebegoldi 2020, Convergence of inexact forward–backward algorithms using the forward–backward envelope, *SIAM J. Optim.*, **30**(4), 3069-3097
- S. B., M. Prato, S. Rebegoldi 2021, New convergence results for the inexact variable metric forward–backward method, *Applied Mathematics and Computation*, **392**, 125719
- S. B., F. Porta, V. Ruggiero, L. Zanni, 2021, Variable metric techniques for forward–backward methods in imaging, Journal of Computational and Applied Mathematics, **385**, 113192

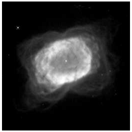
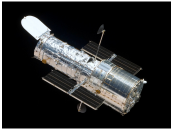
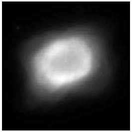
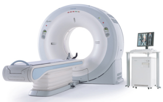
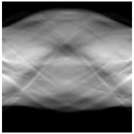
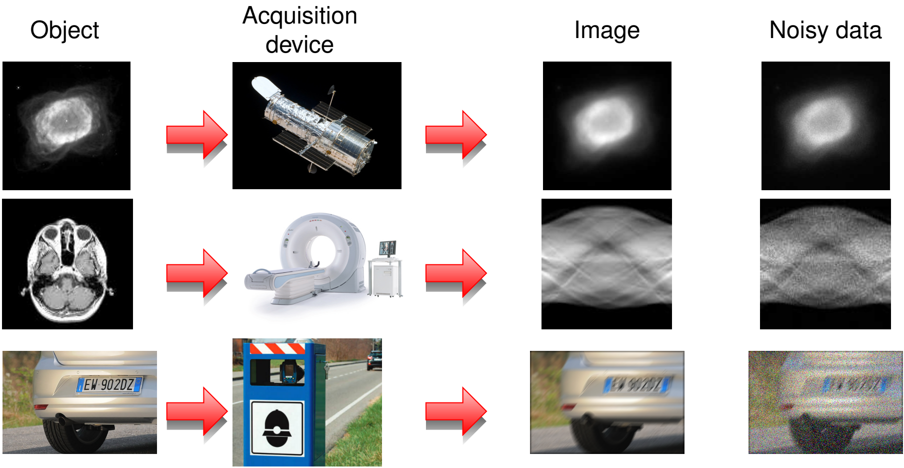
Object	Acquisition device	Image
		
		
		

Image acquisition process: examples



Direct discrete model

g	$=$	Hx^{true}	$+$	ν
data		linear model		noise

$$\left(g, \nu \in \mathbb{R}^m, H \in \mathbb{R}^{m\times n}, x^{true} \in \mathbb{R}^n\right)$$

⬇

Inverse Problem

Try to recover x^{true} by knowing g and H.

⬇

Variational formulation

$$x^{true} \simeq x^* \in \operatorname*{argmin}_{x\in\mathbb{R}^n} f(x), \quad f(x) = d(Hx, g) + \mathcal{R}(x)$$

- $d(Hx, g)$ expresses the data discrepancy
- $\mathcal{R}(x)$ is a regularization term, enforcing some desired property on x^*

$$x^{true} \simeq x^* \in \underset{x \in \mathbb{R}^n}{\operatorname{argmin}}\, d(Hx, g) + \mathcal{R}(x)$$

$$x^{true} \simeq x^* \in \underset{x \in \mathbb{R}^n}{\operatorname{argmin}}\, d(Hx, g) + \mathcal{R}(x)$$

Data discrepancy functions $d(t, g)$ - likelihood functions

Least squares (Gaussian noise)	Convex, quadratic	$\frac{1}{2}\|t-g\|^2$
Kullback-Leibler (Poisson noise)	Convex, nonlinear	$\sum_{i=1}^{n} \log\left(\frac{g_i}{t_i}\right) + t_i - g_i$
Impulse noise	Convex, nonsmooth	$\|t-g\|_1$
Cauchy noise	Nonconvex, nonlinear	$\sum_{i=1}^{n} \log(\rho^2 + (t_i - g_i)^2)$

$$x^{true} \simeq x^* \in \underset{x \in \mathbb{R}^n}{\operatorname{argmin}}\, d(Hx, g) + \mathcal{R}(x)$$

Data discrepancy functions $d(t, g)$ - likelihood functions

Least squares (Gaussian noise)	Convex, quadratic	$\frac{1}{2}\|t-g\|^2$
Kullback-Leibler (Poisson noise)	Convex, nonlinear	$\sum_{i=1}^{n} \log\left(\frac{g_i}{t_i}\right) + t_i - g_i$
Impulse noise	Convex, nonsmooth	$\|t-g\|_1$
Cauchy noise	Nonconvex, nonlinear	$\sum_{i=1}^{n} \log(\rho^2 + (t_i - g_i)^2)$

Regularization functionals $\mathcal{R}(x)$ - Gibbs prior

nonnegativity	Convex, nonsmooth	$\iota_{\geq 0}(x) = \begin{cases} 0 & \text{if } x \geq 0 \\ +\infty & \text{otherwise} \end{cases}$
edge preserving	Convex, nonsmooth	$TV(x) = \beta \sum_{i=0}^{n} \|\nabla_i x\|_2$ (Total Variation)
sparsity	Convex, nonsmooth	$\beta\|Wx\|_1$
smoothness	Convex, smooth	$\beta\|Lx\|_2^2$ (Tichonov)
MRF	Nonconvex, smooth	$\sum_{j=1}^{m} \beta_j \sum_{i=1}^{n} \log(1 + (K_j x)_i^2)$

Reference problem and main assumptions

$$\underset{x\in\mathbb{R}^n}{\operatorname{argmin}}\, f(x) \equiv d(Hx, g) + \mathcal{R}(x)$$

Assumption: all nonconvex terms are smooth, all nonsmooth terms are convex.

$$\underset{x\in\mathbb{R}^n}{\operatorname{argmin}}\, f(x) \equiv d(Hx, g) + \mathcal{R}(x)$$

Assumption: all nonconvex terms are smooth, all nonsmooth terms are convex.

$$\min_{x\in\mathbb{R}^n} f(x) \equiv f_0(x) + f_1(x)$$

f_0 is smooth

f_1 is closed and convex

$$\underset{x\in\mathbb{R}^n}{\operatorname{argmin}} f(x) \equiv d(Hx, g) + \mathcal{R}(x)$$

Assumption: all nonconvex terms are smooth, all nonsmooth terms are convex.

$$\min_{x\in\mathbb{R}^n} f(x) \equiv f_0(x) + f_1(x)$$

f_0 is smooth

↓

we have the **gradient**

$$\nabla f_0(x)$$

f_1 is closed and convex

Reference problem and main assumptions

$$\underset{x \in \mathbb{R}^n}{\operatorname{argmin}}\, f(x) \equiv d(Hx, g) + \mathcal{R}(x)$$

Assumption: all nonconvex terms are smooth, all nonsmooth terms are convex.

$$\min_{x \in \mathbb{R}^n} f(x) \equiv f_0(x) + f_1(x)$$

f_0 is smooth

↓

we have the **gradient**

$$\nabla f_0(x)$$

f_1 is closed and convex

↓

we have the **proximity** (or resolvent) operator:

$$\operatorname{prox}_{f_1}(z) = \underset{x \in \mathbb{R}^n}{\operatorname{argmin}}\, f_1(x) + \frac{1}{2}\|x - z\|^2$$

$$\underset{x\in\mathbb{R}^n}{\operatorname{argmin}}\, f(x) \equiv d(Hx, g) + \mathcal{R}(x)$$

Assumption: all nonconvex terms are smooth, all nonsmooth terms are convex.

$$\min_{x\in\mathbb{R}^n} f(x) \equiv f_0(x) + f_1(x)$$

f_0 is smooth ⇓ we have the **gradient**

$$\nabla f_0(x)$$

f_1 is closed and convex ⇓ we have the **proximity** (or resolvent) operator:

$$\operatorname{prox}_{f_1}(z) = \underset{x\in\mathbb{R}^n}{\operatorname{argmin}}\, f_1(x) + \frac{1}{2}\|x - z\|^2$$

Remark: The proximity operator of the indicator function of a closed convex set $\Omega \subset \mathbb{R}^n$ consists in the orthogonal projection operator onto Ω

$$\iota_\Omega(x) = \begin{cases} 0 & \text{if } x \in \Omega \\ +\infty & \text{otherwise} \end{cases} \Rightarrow \operatorname{prox}_{\iota_\Omega}(z) = \Pi_\Omega(z)$$

Forward-backward iteration

$$
\begin{aligned}
z^{(k)} &= x^{(k)} - \alpha_k \nabla f_0(x^{(k)}) \leftarrow \text{Forward step steepest descent point} \\
y^{(k)} &= \mathrm{prox}_{\alpha_k f_1}(z^{(k)}) \leftarrow \text{Backward step proximal gradient point} \\
d^{(k)} &= y^{(k)} - x^{(k)} \\
x^{(k+1)} &= x^{(k)} + \lambda_k d^{(k)}
\end{aligned}
$$

- two steplength parameters $\alpha_k, \lambda_k \in \mathbb{R}_{>0}$

Forward-backward iteration

$$\begin{aligned} z^{(k)} &= x^{(k)} - \alpha_k \nabla f_0(x^{(k)}) \leftarrow \text{Forward step steepest descent point} \\ y^{(k)} &= \mathrm{prox}_{\alpha_k f_1}(z^{(k)}) \leftarrow \text{Backward step proximal gradient point} \\ d^{(k)} &= y^{(k)} - x^{(k)} \\ x^{(k+1)} &= x^{(k)} + \lambda_k d^{(k)} \end{aligned}$$

- two steplength parameters $\alpha_k, \lambda_k \in \mathbb{R}_{>0}$

Classical FB settings:

- Convexity assumptions;
- Proximity operator available in closed form;
- Lipschitz continuity of ∇f_0 (for steplength computation).

A general forward-backward scheme

Forward-backward iteration

$$\begin{aligned} z^{(k)} &= x^{(k)} - \alpha_k \nabla f_0(x^{(k)}) \leftarrow \text{Forward step steepest descent point} \\ y^{(k)} &= \mathrm{prox}_{\alpha_k f_1}(z^{(k)}) \leftarrow \text{Backward step proximal gradient point} \\ d^{(k)} &= y^{(k)} - x^{(k)} \\ x^{(k+1)} &= x^{(k)} + \lambda_k d^{(k)} \end{aligned}$$

- two steplength parameters $\alpha_k, \lambda_k \in \mathbb{R}_{>0}$

Classical FB settings:

- Convexity assumptions;
- Proximity operator available in closed form;
- Lipschitz continuity of ∇f_0 (for steplength computation).

Challenges in FB methods:

- Nonconvexity;
- Proximity operator not available in closed form;
- Lack of Lipschitz continuity of ∇f_0;
- Implementation complying with theoretical prescriptions;
- Acceleration strategies.

$$\begin{aligned} y^{(k)} &= \mathrm{prox}_{\alpha_k f_1}(x^{(k)} - \alpha_k \nabla f_0(x^{(k)})) \\ d^{(k)} &= y^{(k)} - x^{(k)} \\ x^{(k+1)} &= x^{(k)} + \lambda_k d^{(k)} \end{aligned}$$

Assume that $\alpha_k > 0$ is given.

- The vector $d^{(k)}$ is a **descent direction** for $f(x)$ at the point $x^{(k)}$, i.e.

$$f'(x^{(k)}; d^{(k)}) < 0 \Rightarrow f(x^{(k)} + \lambda d^{(k)}) < f(x^{(k)}) + \lambda f'(x^{(k)}; d^{(k)}) < f(x^{(k)}),$$

for all sufficiently small $\lambda > 0$.

$$\begin{aligned} y^{(k)} &= \mathrm{prox}_{\alpha_k f_1}(x^{(k)} - \alpha_k \nabla f_0(x^{(k)})) \\ d^{(k)} &= y^{(k)} - x^{(k)} \\ x^{(k+1)} &= x^{(k)} + \lambda_k d^{(k)} \end{aligned}$$

Assume that $\alpha_k > 0$ is given.

- The vector $d^{(k)}$ is a **descent direction** for $f(x)$ at the point $x^{(k)}$, i.e.

  $$f'(x^{(k)}; d^{(k)}) < 0 \Rightarrow f(x^{(k)} + \lambda d^{(k)}) < f(x^{(k)}) + \lambda f'(x^{(k)}; d^{(k)}) < f(x^{(k)}),$$

  for all sufficiently small $\lambda > 0$.
- The steplength λ_k can be computed with a backtracking **line–search** loop along $d^{(k)}$, starting from 1, with successive reductions until

  $$f(x^{(k)} + \lambda_k d^{(k)}) \leq f(x^{(k)}) + \lambda_k \Delta_k$$

  where Δ_k is a given negative quantity representing the **sufficient decrease**

Define the following function:

$$h^{(k)}(y) = \nabla f_0(x^{(k)})^T(y - x^{(k)}) + \frac{1}{2\alpha_k}\|y - x^{(k)}\|^2 + f_1(y) - f_1(x^{(k)})$$

It holds that

$$y^{(k)} = \operatorname*{argmin}_{y \in \mathbb{R}^n} h^{(k)}(y)$$

Define the following function:

$$h^{(k)}(y) = \nabla f_0(x^{(k)})^T(y - x^{(k)}) + \frac{1}{2\alpha_k}\|y - x^{(k)}\|^2 + f_1(y) - f_1(x^{(k)})$$

It holds that

$$y^{(k)} = \operatorname*{argmin}_{y \in \mathbb{R}^n} h^{(k)}(y)$$

and

$$f'(x^{(k)}; d^{(k)}) \leq h^{(k)}(y^{(k)}) < 0$$

Define the following function:

$$h^{(k)}(y) = \nabla f_0(x^{(k)})^T(y - x^{(k)}) + \frac{1}{2\alpha_k}\|y - x^{(k)}\|^2 + f_1(y) - f_1(x^{(k)})$$

It holds that

$$y^{(k)} = \operatorname*{argmin}_{y \in \mathbb{R}^n} h^{(k)}(y)$$

and

$$f'(x^{(k)}; d^{(k)}) \le h^{(k)}(y^{(k)}) < 0$$

Generalized Armijo rule [Tseng-Yun, 2009, Porta-Loris, 2015, B. *et al.*, 2016]

$$f(x^{(k)} + \lambda_k d^{(k)}) \le f(x^{(k)}) + \beta\lambda_k h^{(k)}(y^{(k)})$$

where $\beta \in (0, 1)$.

NB: For $f_1 \equiv 0$, dropping the quadratic term gives the standard Armijo rule for smooth optimization.

Define the following function:

$$h^{(k)}(y) = \nabla f_0(x^{(k)})^T(y - x^{(k)}) + \frac{1}{2\alpha_k}\|y - x^{(k)}\|^2 + f_1(y) - f_1(x^{(k)})$$

It holds that

$$y^{(k)} = \underset{y\in\mathbb{R}^n}{\operatorname{argmin}}\, h^{(k)}(y)$$

and

$$f'(x^{(k)}; d^{(k)}) \le h^{(k)}(y^{(k)}) < 0$$

Generalized Armijo rule [Tseng-Yun, 2009, Porta-Loris, 2015, B. *et al.*, 2016]

$$f(x^{(k)} + \lambda_k d^{(k)}) \le f(x^{(k)}) + \beta\lambda_k h^{(k)}(y^{(k)})$$

where $\beta \in (0, 1)$.

NB: For $f_1 \equiv 0$, dropping the quadratic term gives the standard Armijo rule for smooth optimization.

Pros:

- No Lipschitz assumption
- Adaptive selection of λ_k (no user provided parameter)
- Only one proximity operator evaluation per iteration.

$$\tilde{y}^{(k)} \simeq y^{(k)} = \operatorname*{argmin}_{y \in \mathbb{R}^n} h^{(k)}(y) = \operatorname{prox}_{\alpha_k f_1}(x^{(k)} - \alpha_k \nabla f_0(x^{(k)}))$$

Common strategies:

$$\tilde{y}^{(k)} \simeq y^{(k)} = \underset{y \in \mathbb{R}^n}{\operatorname{argmin}}\, h^{(k)}(y) = \operatorname{prox}_{\alpha_k f_1}(x^{(k)} - \alpha_k \nabla f_0(x^{(k)}))$$

Common strategies:

Empirical approach

Apply an iterative optimization method to $\min_{y \in \mathbb{R}^n} h^{(k)}(y)$

Pros:

- Easy to implement.

Cons:

- Theoretical convergence not guaranteed.

$$\tilde{y}^{(k)} \simeq y^{(k)} = \operatorname*{argmin}_{y \in \mathbb{R}^n} h^{(k)}(y) = \mathrm{prox}_{\alpha_k f_1}(x^{(k)} - \alpha_k \nabla f_0(x^{(k)}))$$

Common strategies:

Empirical approach

Apply an iterative optimization method to $\min_{y \in \mathbb{R}^n} h^{(k)}(y)$

Pros:

- Easy to implement.

Cons:

- Theoretical convergence not guaranteed.

Theoretical conditions

Relative error condition:

$$\exists v^{(k)} \in \partial f(\tilde{y}^{(k)}) \text{ s.t. } \|v^{(k)}\| \leq b \|\tilde{y}^{(k)} - x^{(k)}\|$$

[Bolte *et* al. 2014, Ochs 2019]

Pros:

- Theoretical convergence guaranteed.

Cons:

- Not implementable.

$$\tilde{y}^{(k)} \simeq y^{(k)} = \underset{y\in\mathbb{R}^n}{\operatorname{argmin}}\, h^{(k)}(y) = \operatorname{prox}_{\alpha_k f_1}(x^{(k)} - \alpha_k \nabla f_0(x^{(k)}))$$

Borrowing the ideas in [Salzo,Villa 2012], [Villa *et al.* 2013]

replace $0 \in \partial h^{(k)}(y^{(k)})$ with $0 \in \partial_{\epsilon_k} h^{(k)}(\tilde{y}^{(k)})$

$$\partial_{\epsilon_k} h^{(k)}(\tilde{y}^{(k)}) = \{w \in \mathbb{R}^n : h^{(k)}(z) \geq h^{(k)}(\tilde{y}^{(k)}) + w^T(z - \tilde{y}^{(k)}) - \epsilon_k, \ \ \forall z \in \mathbb{R}^n\}$$

$$\tilde{y}^{(k)} \simeq y^{(k)} = \underset{y \in \mathbb{R}^n}{\operatorname{argmin}}\, h^{(k)}(y) = \operatorname{prox}_{\alpha_k f_1}(x^{(k)} - \alpha_k \nabla f_0(x^{(k)}))$$

Borrowing the ideas in [Salzo,Villa 2012], [Villa *et al.* 2013]

> replace $0 \in \partial h^{(k)}(y^{(k)})$ with $0 \in \partial_{\epsilon_k} h^{(k)}(\tilde{y}^{(k)})$

$$\partial_{\epsilon_k} h^{(k)}(\tilde{y}^{(k)}) = \{w \in \mathbb{R}^n : h^{(k)}(z) \geq h^{(k)}(\tilde{y}^{(k)}) + w^T(z - \tilde{y}^{(k)}) - \epsilon_k, \ \ \forall z \in \mathbb{R}^n\}$$

- It satisfies $\|\tilde{y}^{(k)} - y^{(k)}\|^2 \leq \epsilon_k$.
- If, in addition, $h^{(k)}(\tilde{y}^{(k)}) < 0$, the vector $d^{(k)} = \tilde{y}^{(k)} - x^{(k)}$ is still a descent direction for f at $x^{(k)}$.
- It can be realized in practice.

It guarantees both theoretical convergence and practical implementation.

Assume that $f_1(x) = \phi(Ax)$, $A \in \mathbb{R}^{m \times n}$ (easy generalization to $f_1(x) = \sum_{i=1}^{p} \phi_i(A_i x)$).

$$\min_{x \in \mathbb{R}^n} h^{(k)}(x) = \max_{v \in \mathbb{R}^m} \Psi^{(k)}(v) \equiv -\frac{1}{2\alpha_k} \|\alpha_k A^T v - z^{(k)}\|^2 - \phi^*(v) + C_k$$

where ϕ^* is the Fenchel convex conjugate of ϕ.

Assume that $f_1(x) = \phi(Ax)$, $A \in \mathbb{R}^{m\times n}$ (easy generalization to $f_1(x) = \sum_{i=1}^p \phi_i(A_i x)$).

$$\min_{x\in\mathbb{R}^n} h^{(k)}(x) = \max_{v\in\mathbb{R}^m} \Psi^{(k)}(v) \equiv -\frac{1}{2\alpha_k}\|\alpha_k A^T v - z^{(k)}\|^2 - \phi^*(v) + C_k$$

where ϕ^* is the Fenchel convex conjugate of ϕ.

Assume that $f_1(x) = \phi(Ax)$, $A \in \mathbb{R}^{m\times n}$ (easy generalization to $f_1(x) = \sum_{i=1}^{p} \phi_i(A_i x)$).

$$\min_{x\in\mathbb{R}^n} h^{(k)}(x) = \max_{v\in\mathbb{R}^m} \Psi^{(k)}(v) \equiv -\frac{1}{2\alpha_k}\|\alpha_k A^T v - z^{(k)}\|^2 - \phi^*(v) + C_k$$

where ϕ^* is the Fenchel convex conjugate of ϕ. Compute $\tilde{y}^{(k)}$ as follows:

- apply an iterative maximization method to the dual problem, generating the dual sequence $\{v^{(k,\ell)}\}_{\ell\in\mathbb{N}}$ converging to a dual solution
- stop the inner iterations when

$$h^{(k)}(z^{(k)} - \alpha_k A^T v^{(k,\bar{\ell})}) - \Psi^{(k)}(v^{(k,\bar{\ell})}) \leq \epsilon_k$$

- define

$$\tilde{y}^{(k)} = z^{(k)} - \alpha_k A^T v^{(k,\bar{\ell})} \Rightarrow 0 \in \partial_{\epsilon_k} h^{(k)}(\tilde{y}^{(k)})$$

Add a new parameter, a s.p.d. scaling matrix D_k which determines a different metric at each iterate:

$$\text{replace } \|x\| \text{ with } \|x\|_{D_k} = x^T D_k x$$

Variable Metric Inexact Line–Search Algorithm (VMILA)

$$\begin{aligned}
z^{(k)} &= x^{(k)} - \alpha_k D_k \nabla f_0(x^{(k)}) \leftarrow \text{Scaled Forward step} \\
\tilde{y}^{(k)} &\approx \operatorname{prox}_{\alpha_k f_1}^{D_k^{-1}}(z^{(k)}) \leftarrow \text{Scaled Inexact Backward step (loop)} \\
d^{(k)} &= \tilde{y}^{(k)} - x^{(k)} \\
x^{(k+1)} &= x^{(k)} + \lambda_k d^{(k)} \leftarrow \text{Armijo-like line–search (loop)}
\end{aligned}$$

- Inexact proximal gradient point: $\tilde{y}^{(k)}$ s.t. $0 \in \partial_{\epsilon_k} h^{(k)}(\tilde{y}^{(k)})$ and $h^{(k)}(\tilde{y}^{(k)}) < 0$
- Generalized Armijo line–search: compute λ_k by backtracking along $d^{(k)}$ s.t.

$$f(x^{(k)} + \lambda_k d^{(k)}) \leq f(x^{(k)}) + \beta \lambda_k h^{(k)}(\tilde{y}^{(k)})$$

VMILA

λ_k with line–search + ϵ_k-inexact computation of the proximal gradient point

Assumptions:

$D_k \overset{k\to\infty}{\longrightarrow} I$ like C/k^p, $p > 1$

$\alpha_k \in [\alpha_{min}, \alpha_{max}]$

$$\epsilon_k = \begin{cases} \frac{C}{k^q} & \text{with } q > 1 \quad \text{prefixed sequence choice} \\ \text{or} & \\ \eta h^{(k)}(\tilde{y}^{(k)}) & \text{with } \eta \in (0,1] \quad \text{adaptive choice} \end{cases}$$

⬇

- Convergence to a minimizer (without Lipschitz assumptions on $\nabla f_0(x)$)
- Convergence rate $f(x^{(k)}) - f^* = \mathcal{O}(1/k)$ (proof with Lipschitz assumptions on $\nabla f_0(x)$)

Definition: Kurdyka–Łojasiewicz functions

Let $\mathcal{F} : \mathbb{R}^n \longrightarrow \mathbb{R} \cup \{+\infty\}$ be a proper, lower semicontinuous function. The function $\mathcal{F}$ is said to have the KL property at $\bar{z} \in \mathrm{dom}(\partial \mathcal{F})$ if there exist $\upsilon \in (0, +\infty]$, a neighborhood U of $\bar{z}$, a continuous concave function $\phi : [0, \upsilon) \longrightarrow [0, +\infty)$ with $\phi(0) = 0$, $\phi \in C^1(0, \upsilon)$, $\phi'(s) > 0$ for all $s \in (0, \upsilon)$, such that the following inequality is satisfied

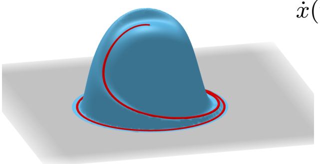
$$\phi'(\mathcal{F}(z) - \mathcal{F}(\bar{z}))\|\partial \mathcal{F}(z)\|_- \geq 1$$

for all $z \in U \cap \{z \in \mathbb{R}^n : \mathcal{F}(\bar{z}) < \mathcal{F}(z) < \mathcal{F}(\bar{z}) + \upsilon\}$.
If $\mathcal{F}$ satisfies the KL property at each point of $\mathrm{dom}(\partial \mathcal{F})$, then $\mathcal{F}$ is called a KL function.

NB: Excludes "pathological" cases for descent methods

$$f(x_1, x_2) = \begin{cases} e^{\frac{1}{r^2-1}} \left(1 - \frac{4r^4}{4r^4 + (1-r^2)^4}\right) \sin\left(\theta - \frac{1}{1-r^2}\right) & \text{if } r < 1 \\ 0 & \text{otherwise} \end{cases}$$

$\dot{x}(t) = -\nabla f(x)$ has not finite length



VMILA

λ_k with line–search + ϵ_k- inexact computation of the proximal gradient point

Assumptions:

D_k have bounded eigenvalues

$\alpha_k \in [\alpha_{min}, \alpha_{max}]$

$\epsilon_k = -\eta h^{(k)}(\tilde{y}^{(k)})$, with $\eta \in (0, 1]$ (adaptive choice)

$f(\cdot) + \|\cdot\|^2$ is a KL function

∇f_0 is Lipschitz

⬇

- If x^* is a limit point of $\{x^{(k)}\}_{k\in\mathbb{N}}$, then it is stationary and the whole sequence converges to it.

Remark:

Theoretical convergence is obtained almost independently on the choice of α_k and D_k.

The idea is to exploit these two almost free parameters to improve practical performances.

Remark:

Theoretical convergence is obtained almost independently on the choice of α_k and D_k.

The idea is to exploit these two almost free parameters to improve practical performances.

- D_k is chosen complying with theoretical prescriptions
  - as a diagonal matrix by mimiking a Majorization-Minimization strategy [Yang, Oja, 2011], [Chouzenoux, Pesquet, 2016]
  - as a LBFGS approximation of the inverse Hessian [Byrd *e*t al., 2016], [Becker *e*t al., 2019]

Remark:

Theoretical convergence is obtained almost independently on the choice of α_k and D_k.

The idea is to exploit these two almost free parameters to improve practical performances.

- D_k is chosen complying with theoretical prescriptions
  - as a diagonal matrix by mimiking a Majorization-Minimization strategy [Yang, Oja, 2011], [Chouzenoux, Pesquet, 2016]
  - as a LBFGS approximation of the inverse Hessian [Byrd *e*t al., 2016], [Becker *e*t al., 2019]
- α_k is computed adapting the well performing strategies for smooth optimization (Barzilai-Borwein, Ritz values [Fletcher 2012])

Remark:

Theoretical convergence is obtained almost independently on the choice of α_k and D_k.

The idea is to exploit these two almost free parameters to improve practical performances.

- D_k is chosen complying with theoretical prescriptions
  - as a diagonal matrix by mimiking a Majorization-Minimization strategy [Yang, Oja, 2011], [Chouzenoux, Pesquet, 2016]
  - as a LBFGS approximation of the inverse Hessian [Byrd *e*t al., 2016], [Becker *e*t al., 2019]
- α_k is computed adapting the well performing strategies for smooth optimization (Barzilai-Borwein, Ritz values [Fletcher 2012])

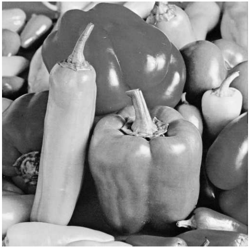
☹ No theoretical results (same rate and lower complexity bound than non-scaled methods).

☺ Good numerical results.

- VMILA has been tested on a variety of convex and nonconvex image restoration problems.
- The numerical comparison shows that its performances are comparable with the ones of state-of-the-art methods such as: Chambolle-Pock (CP) method, preconditioned CP, ADMM, PidSplit+, iPiano, VMFB, FISTA...

Example of application: edge preserving image deblurring in presence of impulse noise.

$$f(x) = \underbrace{\|Hx - g\|_1 + \iota_{\geq 0}(x)}_{f_1(x)} + \underbrace{\rho \sum_{i=1}^{n} \log(1 + \xi \|\nabla_i x\|^2)}_{f_0(x)}$$



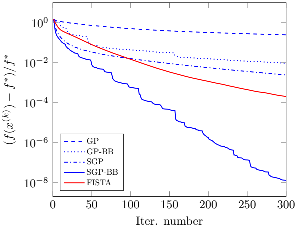
x^{true}

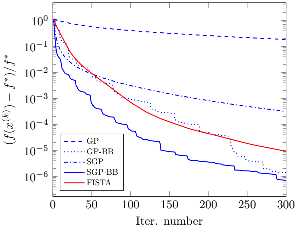
g

x^*
48 outer, 26 av. inner

$$\min_{x\in\mathbb{R}^n} f_0(x) + \iota_{\mathbb{R}^n_{\geq 0}}(x) \iff \min_{x\geq 0} f_0(x)$$

VMILA –> Scaled Gradient Projection (SGP) method
Nonnegative image deconvolution in presence of Poisson noise with smooth TV regularization.





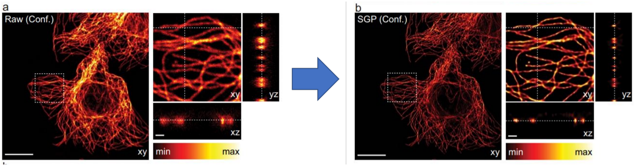
x^*

$x^{(300)}$ SGP

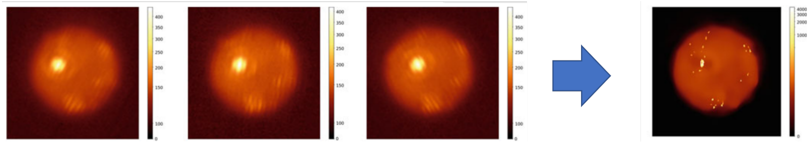
x^*

$x^{(300)}$ SGP

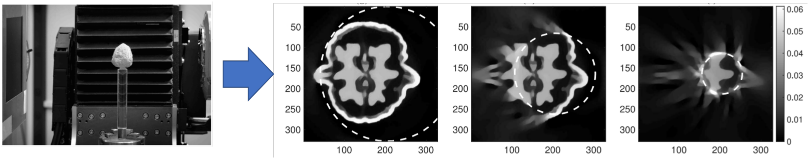
- confocal and STED microscopy (on GPUs) [Zanella *et al.* 2013], [Porta *et al.* 2015]



- astronomical interferometric imaging [Prato *et al.* 2019]



- region of interest computed tomography (ROI-CT) [Bubba *et al.* 2018]



- Algorithm design
  - consider line–search and inexactness in combination with inertial/heavy ball/ FISTA–like acceleration strategies.
  - nonconvex, nonsmooth terms
- Model design
  - Combining machine learning and variational models for image restoration

Classical variational model for image restoration

$$x^{true} \simeq x^* \in \underset{x\in\mathbb{R}^n}{\operatorname{argmin}}\, E(x,\beta)$$

where E is a chosen energy functional containing data discrepancy and regularization, which depends on a set of parameters $\beta \in \mathbb{R}^p$.

Supervised learning - bilevel optimization

Given a dataset of images $\mathcal{D} = \{(x_s^{true}, g_s)\}_{s=1}^N$ where g_s is a noisy version of x_s^{true}, compute the parameters β such that

$$\begin{array}{ll} \min & \sum_{s=1}^N \|x_s^*(\beta) - x_s^{true}\|^2 \\ \beta \in \mathbb{R}^p, & \text{s.t. } x_s^*(\beta) = \operatorname{argmin}_{x\in\mathbb{R}^n} E(x,\beta) \end{array}$$

Unrolling techniques

Replace $\operatorname{argmin}_{x\in\mathbb{R}^n} E(x,\beta)$ with the image obtained after m steps of an optimization algorithm applied to the variational problem $\min_x E(x,\beta)$, possibly learning algorithms and model parameters simultaneously.

Classical variational model for image restoration

$$x^{true} \simeq x^* \in \underset{x\in\mathbb{R}^n}{\operatorname{argmin}}\, E(x,\beta)$$

where E is a chosen energy functional containing data discrepancy and regularization, which depends on a set of parameters $\beta \in \mathbb{R}^p$.

Supervised learning - bilevel optimization

Given a dataset of images $\mathcal{D} = \{(x_s^{true}, g_s)\}_{s=1}^N$ where g_s is a noisy version of x_s^{true}, compute the parameters β such that

$$\begin{array}{ll} \min & \sum_{s=1}^N \|x_s^*(\beta) - x_s^{true}\|^2 \\ \beta \in \mathbb{R}^p, & \text{s.t. } x_s^*(\beta) = \operatorname{argmin}_{x\in\mathbb{R}^n} E(x,\beta) \end{array}$$

Unrolling techniques

Replace $\operatorname{argmin}_{x\in\mathbb{R}^n} E(x,\beta)$ with the image obtained after m steps of an optimization algorithm applied to the variational problem $\min_x E(x,\beta)$, possibly learning algorithms and model parameters simultaneously.

Classical variational model for image restoration

$$x^{true} \simeq x^* \in \underset{x\in\mathbb{R}^n}{\operatorname{argmin}}\, E(x,\beta)$$

where E is a chosen energy functional containing data discrepancy and regularization, which depends on a set of parameters $\beta \in \mathbb{R}^p$.

Supervised learning - bilevel optimization

Given a dataset of images $\mathcal{D} = \{(x_s^{true}, g_s)\}_{s=1}^N$ where g_s is a noisy version of x_s^{true}, compute the parameters β such that

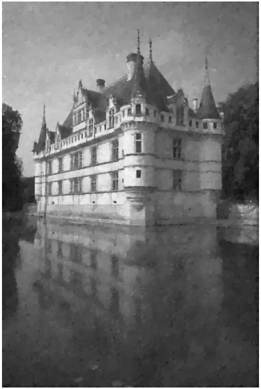
$$\begin{array}{ll} \min & \sum_{s=1}^N \|x_s^*(\beta) - x_s^{true}\|^2 \\ \beta\in\mathbb{R}^p, & \text{s.t. } x_s^*(\beta) = \operatorname{argmin}_{x\in\mathbb{R}^n} E(x,\beta) \end{array}$$

Unrolling techniques

Replace $\operatorname{argmin}_{x\in\mathbb{R}^n} E(x,\beta)$ with the image obtained after m steps of an optimization algorithm applied to the variational problem $\min_x E(x,\beta)$, possibly learning algorithms and model parameters simultaneously.

$$E(x,\beta) = \frac{1}{2}\|x-g\|^2 + \rho \sum_{i=1}^{n} \|\nabla_i x\| \quad \beta \leftrightarrow \rho$$

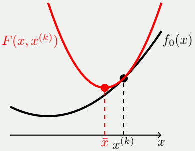
$$E(x,\beta) = \frac{1}{2}\|x-g\|^2 + \sum_{j=1}^{q} \rho_j \sum_{i=1}^{n} \log(1 + ([\kappa_j * x]_i)^2) \quad \beta \leftrightarrow \rho_j, \kappa_j, j = 1, ..., q$$



TV restoration
PSNR 27.85

learned prior restoration
PSNR 29.89

Majorization-Minimization idea



If $F(x, x^{(k)})$ is an auxiliary function for f_0 if

$$F(x^{(k)}, x^{(k)}) = f_0(x^{(k)}) \text{ and } F(x, x^{(k)}) \geq f_0(x) \quad \forall x \in \mathbb{R}^n$$

then,

$$\bar{x} = \operatorname*{argmin}_{x \in \mathbb{R}^n} F(x, x^{(k)}) \Rightarrow f_0(\bar{x}) \leq f_0(x^{(k)})$$

For several relevant f_0, an auxiliary function can be build as

- Quadratic auxiliary function [Chouzenoux, Pesquet, 2016]:

$$F(x, x^{(k)}) = f_0(x^{(k)}) + (x - x^{(k)})^T \nabla f_0(x^{(k)}) + \frac{1}{2}(x - x^{(k)})^T {D_k}^{-1}(x - x^{(k)})$$

- Non quadratic auxiliary function [Yang, Oja, 2011].

In both cases there exists a diagonal matrix D_k build on the component of $\nabla f_0(x^{(k)})$, such that

$$x^{(k)} - D_k \nabla f_0(x^{(k)}) = \operatorname*{argmin}_{x \in \mathbb{R}^n} F(x, x^{(k)})$$

The convergence condition $D_k \to I$ can be fulfilled by squeezing the elements of the diagonal matrix D_k to 1 as k increases.

- Choose D_k using the 0-memory LBFGS idea [Ochs *et al.*, 2019]

$$D_k = \tau_k (I - \rho_k s^{(k-1)} w^{(k-1)^T})(I - \rho_k s^{(k-1)} w^{(k-1)^T})^T + \rho_k s^{(k-1)} s^{(k-1)^T}$$

where

$$s^{(k-1)} = x^{(k)} - x^{(k-1)}, \quad w^{(k-1)} = \nabla f_0(x^{(k)}) - \nabla f_0(x^{(k-1)})$$

$$\rho_k = \frac{1}{s^{(k-1)^T} w^{(k-1)}}, \quad \tau_k = \frac{s^{(k-1)^T} w^{(k-1)}}{\|w^{(k-1)}\|^2}$$

- Non diagonal matrix
  - The scaled direction $D_k \nabla f_0(x^{(k)})$ can be implemented via only scalar products
  - Similar formula for $D_k{}^{-1}$
  - The bound on the eigenvalues can be checked on the coefficients τ_k, ρ_k

Given D_k, we would choose α_k such that

$$\frac{1}{\alpha_k} D_k^{-1} \simeq \nabla^2 f_0(x^{(k)})$$

simulating the Taylor's equality

$$\nabla f_0(x+d) = \nabla f_0(x) + \int_0^1 \nabla^2 f_0(x+td)d\, dt$$

$$\underbrace{\nabla f_0(x^{(k)}) - \nabla f_0(x^{(k-1)})}_{w^{(k-1)}} \simeq \frac{1}{\alpha_k} D_k^{-1} (\underbrace{x^{(k)} - x^{(k-1)}}_{s^{(k-1)}})$$

$$\alpha_k{}^{BB1} = \underset{\alpha}{\operatorname{argmin}} \left\| \frac{1}{\alpha} D_k^{-1} s^{(k-1)} - w^{(k-1)} \right\| = \frac{\| D_k^{-1} s^{(k-1)} \|^2}{s^{(k-1)^T} D_k^{-1} w^{(k-1)}}$$

$$\alpha_k{}^{BB2} = \underset{\alpha}{\operatorname{argmin}} \left\| s^{(k-1)} - \alpha D_k w^{(k-1)} \right\| = \frac{s^{(k-1)^T} D_k w^{(k-1)}}{\| D_k^{-1} w^{(k-1)} \|^2}$$

- Good results when the two values are alternated following an adaptive switching rule and projected onto a given interval $[\alpha_{\min}, \alpha_{\max}]$, with $0 < \alpha_{\min} < \alpha_{\max}$.
- Recent developments in steplength selection rules: Ritz values [Fletcher 2012]