

# Continuous (convex) optimisation

M2 - PSL / Dauphine / S.U.

Antonin Chambolle, CNRS, CEREMADE

Université Paris Dauphine PSL

Oct.-Dec. 2023

Lecture 4: Splitting algorithms, Acceleration, FISTA

# Contents

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems

Splitting methods

Descent  
algorithms

Forward-Backward

Acceleration

## 1 Algorithms for monotone operators

- Abstract problems
- Splitting methods

## 2 Descent algorithms

- Forward-Backward
- Acceleration

# Abstract methods for Monotone operators

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

General problem:

$$0 \in Ax \quad \text{or} \quad 0 \in Ax + Bx$$

where  $A, B$  are maximal monotone operators (which may or may not be subgradients).

# Explicit methods

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

Generalization of gradient descent:

$$x^{k+1} = x^k - \tau p^k, p^k \in Ax^k.$$

Issue: Even if  $A$  is single-valued and Lipschitz continuous, then this might not work.

Example:  $A = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$ . Then,

$$x^k = \begin{pmatrix} 1 & -\tau \\ \tau & 1 \end{pmatrix}^k x^0.$$

The eigenvalues of this matrix are  $1 + \pm\tau i$  with modulus  $\sqrt{1 + \tau^2}$  and the iteration always diverges (unless  $x^0 = 0$ ).

# Explicit methods

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

So one needs a stronger condition on  $A$ . We recall that the gradient descent works for convex functions with Lipschitz gradient, and the proof relies on the co-coercivity.

## Theorem

Let  $A$  maximal monotone be  $\mu$ -co-coercive (in particular, single-valued):

$$\langle Ax - Ay, x - y \rangle \geq \mu |Ax - Ay|^2.$$

Assume there exists a solution to  $Ax = 0$ . Then the iteration  $x^{k+1} = x^k - \tau Ax^k$  converges to  $x^*$  with  $Ax^* = 0$  if  $0 < \tau < 2\mu$ .

**Remark:** this is the same as  $\mu A$  firmly non-expansive.

Then, the proof relies on proving that  $I - \tau A$  is an averaged operator.

# Explicit methods

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

*Proof:*

$$\begin{aligned} |(I - \tau A)x - (I - \tau A)y|^2 &= |x - y|^2 - 2\tau \langle x - y, Ax - Ay \rangle + \tau^2 |Ax - Ay|^2 \\ &\leq |x - y|^2 - \tau(2\mu - \tau) |Ax - Ay|^2. \end{aligned}$$

This shows that if  $0 \leq \tau \leq 2\mu$ ,  $I - \tau A$  is 1-Lipschitz (nonexpansive). Hence for  $\tau < 2\mu$ ,

$$I - \tau A = \left(1 - \frac{\tau}{2\mu}\right)I + \frac{\tau}{2\mu}(I - (2\mu)A)$$

is averaged. By The K-M Theorem, the iterates weakly converge, as  $k \rightarrow \infty$ , to a fixed point of  $(I - \tau A)$  (if it exists). If  $\tau = 0$  this is not interesting, if  $0 < \tau < 2\mu$ , then it is a zero of  $A$ , which exists by assumption.

# Explicit method without co-coercivity?

Extragradient method (Korpelevich, 1976)

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

In case  $B$  is just  $L$ -Lipschitz continuous, the following method was proposed in 1976 by G. M. Korpelevich:

$$\begin{cases} y^k = x^k - \tau Bx^k \\ x^{k+1} = x^k - \tau By^k \end{cases}$$

## Theorem

*If  $\tau L < 1$ , then the algorithm generates sequences  $x^k$  and  $y^k$  which (weakly) converge to a solution of  $Bx \ni 0$ , if there exists one. In addition,  $|x^k - y^k| \rightarrow 0$ .*

*Remark:* the original paper has an additional projection step (for a convex constraint), the proof is almost identical.

# Extragradient method

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

*Proof:* For this algorithm we cannot use out of the box a previous theorem. We compute, for  $x^*$  with  $Bx^* \ni 0$ ,

$$|x^{k+1} - x^*|^2 = |x^k - x^*|^2 + 2 \langle x^k - x^*, x^{k+1} - x^k \rangle + |x^{k+1} - x^k|^2 = |x^k - x^*|^2 - 2\tau \langle x^k - x^*, By^k \rangle + |x^{k+1} - x^k|^2.$$

We use then that  $\langle x^k - x^*, By^k \rangle = \langle x^k - y^k + y^k - x^*, By^k - Bx^* \rangle \geq \langle x^k - y^k, By^k \rangle$  and deduce:

$$|x^{k+1} - x^*|^2 \leq |x^k - x^*|^2 - 2\tau \langle x^k - y^k, By^k \rangle + |x^{k+1} - x^k|^2 = |x^k - x^*|^2 + 2 \langle x^k - y^k, x^{k+1} - x^k \rangle + |x^{k+1} - x^k|^2.$$

It follows:

$$\begin{aligned} |x^{k+1} - x^*|^2 &\leq |x^k - x^*|^2 + |x^{k+1} - y^k|^2 - |x^k - y^k|^2 \\ &= |x^k - x^*|^2 + |\tau By^k - \tau Bx^k|^2 - |x^k - y^k|^2 \leq |x^k - x^*|^2 - (1 - \tau^2 L^2) |y^k - x^k|^2. \end{aligned}$$

We deduce, when  $\tau L < 1$ , that  $|x^k - x^*|$  is decreasing (Fejér-monotonicity of the sequence), that  $|x^k - y^k| \rightarrow 0$  (and therefore also  $|x^{k+1} - y^k|$  and  $|x^{k+1} - x^k|$ ) and can continue as in the proof of KM's theorem.



# Extragradient method

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems

Splitting methods

Descent  
algorithms

Forward-Backward

Acceleration

One also needs to check that a fixed point is a solution! A fixed point satisfies:

$y = x - \tau Bx$ ,  $x = x - \tau By$ . Hence one has  $y - x = \tau(By - Bx)$  so that  
 $|y - x| \leq \tau L |y - x|$ . If  $\tau L < 1$  then  $y - x = 0$  and  $Bx = 0$ . □

# Proximal point algorithm

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

Now we consider the “implicit descent”:

$$x^{k+1} \in x^k - \tau Ax^{k+1}$$

This is precisely which is solved by

$$x^{k+1} = (I + \tau A)^{-1} x^k = J_{\tau A} x^k$$

which is well-posed for  $A$  is maximal monotone.

This iteration is known as the *proximal point algorithm*. It obviously converges to a fixed point as the operator is  $(1/2)$ -averaged (if the fixed point, that is a point with  $Ax = 0$ , exists).

# Proximal point algorithm

## Overrelaxation

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

The reflexion  $R_{\tau A} = 2(I + \tau A)^{-1} - I$  is 1-Lipschitz and one can generalize as follows:

$$x^{k+1} = (1 - \theta_k)x^k + \theta_k R_{\tau A} x^k = x^k + 2\theta_k \left( (I + \tau A)^{-1} x^k - x^k \right) = x^k - 2\theta_k \tau A_{\tau} x^k,$$

for  $0 < \underline{\theta} \leq \theta_k \leq \bar{\theta} < 1$ .

We still get convergence.

# Proximal point algorithm

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

## Theorem (PPA Algorithm)

Let  $x^0 \in \mathcal{X}$ ,  $\tau_k \geq \underline{\tau} > 0$ ,  $0 \leq \underline{\lambda} \leq \lambda_k \leq \bar{\lambda} \leq 2$ , and let

$$x^{k+1} = x^k + \lambda_k((I + \tau_k A)^{-1}x^k - x^k). \quad (1)$$

If there exists  $x$  with  $Ax \ni 0$ , then  $x^k$  weakly converges to a zero of  $A$ .

We could also consider (summable) errors. (See Bauschke-Combettes for variants, Eckstein-Bertsekas for a proof with errors.)

*Proof.* The proof follows the lines of the proof of the KM Theorem.

We observe that obviously,  $|x^{k+1} - x|^2 \leq |x^k - x|^2$  for each  $k \geq 0$  and for each  $x$  with  $Ax \ni 0$ . But we can be more precise. One has:

# Proximal point algorithm

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

$$\begin{aligned} |x^{k+1} - x|^2 &= |x^k - x|^2 + \lambda_k^2 |J_{\tau_k A} x^k - x^k|^2 + 2\lambda_k \langle x^k - x, J_{\tau_k A} x^k - x^k \rangle \\ &= |x^k - x|^2 + \lambda_k^2 |J_{\tau_k A} x^k - x^k|^2 \\ &\quad + \lambda_k (|J_{\tau_k A} x^k - x|^2 - |x^k - x|^2 - |J_{\tau_k A} x^k - x^k|^2). \end{aligned}$$

As  $J_{\tau_k A}$  is firmly non-expansive:

$$|J_{\tau_k A} x^k - x|^2 + |(I - J_{\tau_k A})x^k - (I - J_{\tau_k A})x|^2 \leq |x^k - x|^2$$

where in addition  $(I - J_{\tau_k A})x = 0$  so that  $|(I - J_{\tau_k A})x^k - (I - J_{\tau_k A})x|^2 = |x^k - J_{\tau_k A} x^k|^2$ . Hence:

$$\begin{aligned} |x^{k+1} - x|^2 &\leq |x^k - x|^2 + \lambda_k^2 |J_{\tau_k A} x^k - x^k|^2 - 2\lambda_k |J_{\tau_k A} x^k - x^k|^2 \\ &= |x^k - x|^2 - \lambda_k(2 - \lambda_k) |J_{\tau_k A} x^k - x^k|^2. \end{aligned}$$

# Proximal Point Algorithm

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

Letting  $c = \underline{\lambda}(2 - \bar{\lambda}) > 0$ , we deduce that  $(x^k)_k$  is Fejér-monotone with respect to  $\{x : Ax \ni 0\}$  and that

$$c \sum_{k=0}^n |J_{\tau_k} A x^k - x^k|^2 + |x^{n+1} - x|^2 \leq |x^0 - x|^2$$

for all  $n \geq 0$ , in particular  $|J_{\tau_k} A x^k - x^k| \rightarrow 0$  (as well as, by the scheme,  $x^{k+1} - x^k$ ).

We would like to deduce convergence as in the proof of KM's Theorem. Yet, with varying  $\tau_k$ , it is not obvious that a limit point  $\bar{x}$  of a subsequence  $x^{k_l}$  is a fixed point (of what?).

But one proves that  $Ax \ni 0$  using the maximal-monotonicity of  $A$ . If  $x' \ni \mathcal{X}$ ,  $y' \in Ax'$ , denoting  $e_k := J_{\tau_k} A x^k - x^k \rightarrow 0$  we have:

$$A(x^k + e^k) \ni -\frac{e^k}{\tau_k},$$

so that

$$\left\langle y' + \frac{e^k}{\tau_k}, x' - x^k - e^k \right\rangle \geq 0.$$

In the limit along the subsequence  $x^{k_l}$ , we find  $\langle y', x' - \bar{x} \rangle \geq 0$ , so that  $A\bar{x} \ni 0$ . The rest of the proof relies on Opial's lemma and is as in the proof of the KM Theorem.

# Splitting methods

## Forward-Backward splitting

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

We can now mix the implicit and explicit algorithms: Let  $A, B$  be maximal-monotone, with  $B$   $\mu$ -co-coercive. We define the *forward-backward splitting* algorithm as:

$$x^{k+1} = (I + \tau A)^{-1}(I - \tau B)x^k$$

If  $0 < \tau < 2\mu$ , the algorithm is the composition of two averaged operator  $\rightarrow$  converges weakly to a fixed point if it exists:

$$(I + \tau A)^{-1}(I - \tau B)x = x \Leftrightarrow x - \tau Bx \in x + \tau Ax \Leftrightarrow Ax + Bx \ni 0.$$

(As  $B$  is continuous, this is equivalent to  $(A + B)x \ni 0$ . Hence, if  $A + B$  has a zero, this algorithm converges to a zero of  $A + B$ .)

# Douglas-Rachford splitting

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems

Splitting methods

Descent  
algorithms

Forward-Backward

Acceleration

Introduced under the following form in a paper of Lions and Mercier (79):

$$x^{k+1} = J_{\tau A}(2J_{\tau B} - I)x^k + (I - J_{\tau B})x^k$$

## Theorem

Let  $x^0 \in \mathcal{X}$ . Then  $x^k \rightarrow x$  such that  $w = J_{\tau B}x$  is a solution of  $Aw + Bw \ni 0$  (if it exists).



# Douglas-Rachford splitting

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems

Splitting methods

Descent  
algorithms

Forward-Backward

Acceleration

Introduced under the following form in a paper of Lions and Mercier (79):

$$x^{k+1} = J_{\tau A}(2J_{\tau B} - I)x^k + (I - J_{\tau B})x^k$$

## Theorem

Let  $x^0 \in \mathcal{X}$ . Then  $x^k \rightarrow x$  such that  $w = J_{\tau B}x$  is a solution of  $Aw + Bw \ni 0$  (if it exists).

To prove this, we express the iterations in terms of the reflexion operators:

$$J_{\tau A} = \frac{1}{2}I + \frac{1}{2}R_{\tau A}, \quad J_{\tau B} = \frac{1}{2}I + \frac{1}{2}R_{\tau B}.$$

# Douglas-Rachford splitting

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

One has then

$$\begin{aligned} J_{TA}(2J_{TB} - I)x + (I - J_{TB})x &= \left( \frac{I + R_{TA}}{2}(R_{TB}) + \frac{I - R_{TB}}{2} \right) (x) \\ &= \frac{I + R_{TA} \circ R_{TB}}{2} (x) \end{aligned}$$

It follows that the iterates are of an averaged operator (with  $1/2$ ).

A fixed points satisfies:

$$\begin{aligned} x = J_{TA}(2J_{TB} - I)x + (I - J_{TB})x &\Leftrightarrow w := J_{TB}x = J_{TA}(2w - x) \\ &\Leftrightarrow w + \tau Aw \ni 2w - x \Leftrightarrow \tau Aw \ni w - x \end{aligned}$$

Now since  $w + \tau Bw \ni x$ , this is  $\tau Aw + \tau Bw \ni 0$ , which shows the theorem.  $\square$

# Douglas-Rachford splitting and Peaceman-Rachford splitting

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems

Splitting methods

Descent  
algorithms

Forward-Backward

Acceleration

*Remark:* In addition: one can consider an “over-relaxed” iteration with operator:

$$(1 - \theta)I + \theta R_{TA} \circ R_{TB} = I + 2\theta(J_{TA}(2J_{TB} - I) - J_{TB}).$$

for  $0 < \theta < 1$ . The case  $\theta = 1$  is called the “Peaceman-Rachford” splitting and converges under some conditions on  $A, B$ .

# Descent algorithms: Forward-backward descent

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

In case  $A = \partial g$ ,  $B = \nabla f$ ,  $g, f$  convex, lsc,  $f$  with  $L$ -Lipschitz gradient, the forward-backward splitting solves  $\partial g(x) + \nabla f(x) \ni 0$ : then  $x$  is a minimizer of the composite minimization problem:

$$\min_x F(x) := f(x) + g(x).$$

We consider the operator:

$$\bar{x} \mapsto \hat{x} = T_\tau \bar{x} := \text{prox}_{\tau g}(\bar{x} - \tau \nabla f(\bar{x})) = (I + \tau \partial g)^{-1}(\bar{x} - \tau \nabla f(\bar{x})).$$

It corresponds to one explicit descent step for  $f$  followed by an implicit descent step for  $g$ .

[Also “composite” gradient descent, where  $(T_\tau(x) - x)/\tau$  is the “composite” gradient of  $f + g$ , cf Nesterov, 2005]

# Forward-Backward descent with fixed step (“ISTA”)

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

We choose  $x^0 \in \mathcal{X}$  and let  $x^{k+1} = T_\tau x^k$  for fixed  $k$ . Then we have seen that if  $\tau < 2/L$ , the method converges to a fixed point of  $T_\tau$  which is a minimizer of  $F$ . In this case we can additionally show, at least for  $\tau \leq 1/L$ :

$$F(x^k) - F(x^*) \leq \frac{1}{2\tau k} |x^* - x^0|^2$$

while in case  $f$  is  $\mu_f$  convex and/or  $g$  is  $\mu_g$  convex ( $\mu_f, \mu_g \geq 0$ ,  $\mu_f + \mu_g > 0$ ) one shows:

$$F(x^k) - F(x^*) + \frac{1+\tau\mu_g}{2\tau} |x^k - x^*|^2 \leq \omega^k \frac{1+\tau\mu_g}{2\tau} |x^0 - x^*|^2.$$

where  $\omega = (1 - \tau\mu_f)/(1 + \tau\mu_g) < 1$ .

# Proof: descent inequality

Let  $\hat{x} = T_\tau \bar{x}$ : then for all  $x \in \mathcal{X}$ ,

$$F(x) + (1 - \tau\mu_f) \frac{|x - \bar{x}|^2}{2\tau} \geq \frac{1 - \tau L}{\tau} \frac{|\hat{x} - \bar{x}|^2}{2} + F(\hat{x}) + (1 + \tau\mu_g) \frac{|x - \hat{x}|^2}{2\tau}.$$

In particular, if  $\tau L \leq 1$ ,

$$F(x) + (1 - \tau\mu_f) \frac{|x - \bar{x}|^2}{2\tau} \geq F(\hat{x}) + (1 + \tau\mu_g) \frac{|x - \hat{x}|^2}{2\tau}.$$

The proof relies on the fact that  $\hat{x}$  is obtained as a minimizer of

$$\min_x f(\bar{x}) + \langle \nabla f(\bar{x}), x - \bar{x} \rangle + g(x) + \frac{1}{2\tau} |x - \bar{x}|^2$$

which is  $(\mu_g + \frac{1}{\tau})$ -convex.

# Descent inequality

Proof: One has

$$\begin{aligned} f(\bar{x}) + \langle \nabla f(\bar{x}), x - \bar{x} \rangle + g(x) + \frac{1}{2\tau} |x - \bar{x}|^2 \\ \geq f(\bar{x}) + \langle \nabla f(\bar{x}), \hat{x} - \bar{x} \rangle + g(\hat{x}) + \frac{1}{2\tau} |\hat{x} - \bar{x}|^2 + (\mu_g + \frac{1}{\tau}) \frac{1}{2} |x - \hat{x}|^2 \end{aligned}$$

Now, on the one hand we have:

$$F(x) = f(x) + g(x) \geq f(\bar{x}) + \langle \nabla f(\bar{x}), x - \bar{x} \rangle + \frac{\mu_f}{2} |x - \hat{x}|^2 + g(x)$$

and on the other hand because  $\nabla f$  is  $L$ -Lipschitz we have

$$f(\bar{x}) + \langle \nabla f(\bar{x}), \hat{x} - \bar{x} \rangle + g(\hat{x}) \geq f(\hat{x}) - \frac{L}{2} |\hat{x} - \bar{x}|^2 + g(\hat{x}) = F(\hat{x}) - \frac{L}{2} |\hat{x} - \bar{x}|^2.$$

Combining these three inequalities we get the descent inequality:

$$F(x) + (1 - \tau\mu_f) \frac{|x - \bar{x}|^2}{2\tau} \geq F(\hat{x}) + (1 + \tau\mu_g) \frac{|x - \hat{x}|^2}{2\tau}.$$

# Rates of convergence for the FB splitting

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

We consider the case  $\mu_f + \mu_g = 0$ . The descent rule with  $x = x^*$  shows that:

$$F(x^{k+1}) + \frac{1}{2\tau}|x^{k+1} - x^*|^2 \leq F(x^*) + \frac{1}{2\tau}|x^k - x^*|^2$$

while for  $x = x^k$  we get:

$$F(x^{k+1}) + \frac{1}{2\tau}|x^{k+1} - x^k|^2 \leq F(x^k)$$

We deduce that for  $N \geq 1$ ,

$$N(F(x^N) - F(x^*)) \leq \sum_{k=0}^{N-1} F(x^{k+1}) - F(x^*) + \frac{1}{2\tau}|x^N - x^*|^2 \leq \frac{1}{2\tau}|x^0 - x^*|^2.$$



# FISTA: acceleration for the FB splitting

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

Due in this form to Beck and Teboulle (2009), see also Nesterov (1983, 2004 “Introductory lectures...”)

**Algorithm:** FISTA with fixed steps:

Choose  $x^0 = x^{-1} \in \mathcal{X}$  and  $t_0 \geq 0$

**for all**  $k \geq 0$  **do**

$$y^k = x^k + \beta_k(x^k - x^{k-1})$$

$$x^{k+1} = T_\tau y^k = \text{prox}_{\tau g}(y^k - \tau \nabla f(y^k))$$

where

$$t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2} \geq \frac{k+1}{2},$$

$$\beta_k = \frac{t_k - 1}{t_{k+1}},$$

**end for**

# FISTA: strongly convex case

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

In case  $\mu = \mu_f + \mu_g > 0$  is known, then the previous method is not optimal. One should choose:

$$t_{k+1} = \frac{1 - qt_k^2 + \sqrt{(1 - qt_k^2)^2 + 4t_k^2}}{2},$$
$$\beta_k = \frac{t_k - 1}{t_{k+1}} \frac{1 + \tau\mu_g - t_{k+1}\tau\mu}{1 - \tau\mu_f},$$

where  $q = \tau\mu / (1 + \tau\mu_g) < 1$ , or alternatively the fixed overrelaxation parameter:

$$\beta = \frac{\sqrt{1 + \tau\mu_g} - \sqrt{\tau\mu}}{\sqrt{1 + \tau\mu_g} + \sqrt{\tau\mu}}.$$

## Theorem

If  $\sqrt{q}t_0 \leq 1$ ,  $t_0 \geq 0$ , then the sequence  $(x^k)$  produced the algorithm satisfies

$$F(x^k) - F(x^*) \leq \min \left\{ \frac{(1 - \sqrt{q})^k}{t_0^2}, \frac{4}{(k+1)^2} \right\} \left( t_0^2 (F(x^0) - F(x^*)) + \frac{1 + \tau\mu g}{2\tau} |x^0 - x^*|^2 \right)$$

if  $t_0 \geq 1$ , and

$$F(x^k) - F(x^*) \leq \min \left\{ (1 + \sqrt{q})(1 - \sqrt{q})^k, \frac{4}{(k+1)^2} \right\} \left( t_0^2 (F(x^0) - F(x^*)) + \frac{1 + \tau\mu g}{2\tau} |x^0 - x^*|^2 \right)$$

if  $t_0 \in [0, 1]$ , where  $x^*$  is a minimiser of  $F$ .

Common choices are  $t_0 = 0$ ,  $t_0 = 1$ . The rate is “optimal”.

# FISTA: proof

Again we prove first  $\mu_f + \mu_g = 0$ .

In that case, the algorithm has the form  $x^{k+1} = T_\tau y^k$  for some  $y^k$  which we will specify later. One has for all  $x$ :

$$F(x^{k+1}) + \frac{|x - x^{k+1}|^2}{2\tau} \leq F(x) + \frac{|x - y^k|^2}{2\tau}$$

The idea is to choose  $x$  as a convex combination of a minimizer  $x^*$  [or any point] and the old point  $x^k$ , and use the convexity to deduce a “better” decrease. Here we choose (as it will make the computation much quicker)  $x = ((t-1)x^k + x^*)/t$ ,  $t \geq 1$ , and we find:

$$\begin{aligned} F(x^{k+1}) - F(x^*) + \frac{|(t-1)x^k + x^* - tx^{k+1}|^2}{2t^2\tau} &\leq F\left(\frac{(t-1)x^k + x^*}{t}\right) - F(x^*) + \frac{|(t-1)x^k + x^* - ty^k|^2}{2t^2\tau} \\ &\leq \frac{t-1}{t}(F(x^k) - F(x^*)) + \frac{|(t-1)x^k + x^* - ty^k|^2}{2t^2\tau}. \end{aligned}$$

Hence multiplying by  $t^2$  and adding an index  $k+1$  to  $t$ :

$$\begin{aligned} t_{k+1}^2(F(x^{k+1}) - F(x^*)) + \frac{|(t_{k+1}-1)x^k + x^* - t_{k+1}x^{k+1}|^2}{2\tau} \\ \leq t_{k+1}(t_{k+1}-1)(F(x^k) - F(x^*)) + \frac{|(t_{k+1}-1)x^k + x^* - t_{k+1}y^k|^2}{2\tau}. \end{aligned}$$

# FISTA: proof

We see here that the factor in front of  $F(x^k)$  is strictly less than in front of  $F(x^{k+1})$ .

$$\begin{aligned} t_{k+1}^2(F(x^{k+1}) - F(x^*)) + \frac{|(t_{k+1} - 1)x^k + x^* - t_{k+1}x^{k+1}|^2}{2\tau} \\ \leq t_{k+1}(t_{k+1} - 1)(F(x^k) - F(x^*)) + \frac{|(t_{k+1} - 1)x^k + x^* - t_{k+1}y^k|^2}{2\tau}. \end{aligned}$$

This iteration can be iterated if the sequences  $t_k$  and  $y_k$  satisfy:

$$\begin{aligned} t_{k+1}(t_{k+1} - 1) &= t_k^2 && (\leq \text{ if } x^* \text{ is a minimizer}) \\ (t_{k+1} - 1)x^k + x^* - t_{k+1}y^k &= (t_k - 1)x^{k-1} + x^* - t_kx^k. \end{aligned}$$

Then, indeed, we have

$$\begin{aligned} t_{k+1}^2(F(x^{k+1}) - F(x^*)) + \frac{|(t_{k+1} - 1)x^k + x^* - t_{k+1}x^{k+1}|^2}{2\tau} \\ \leq t_k^2(F(x^k) - F(x^*)) + \frac{|(t_k - 1)x^{k-1} + x^* - t_kx^k|^2}{2\tau} \end{aligned}$$

and summing we obtain

$$t_N^2(F(x^N) - F(x^*)) \leq t_0^2(F(x^0) - F(x^*)) + \frac{|(t_0 - 1)x^{-1} + x^* - t_0x^0|^2}{2\tau}$$

with by convention  $y^0 = x^0 = x^{-1}$ , and  $t_0$  does not need to be  $\geq 1$  (only  $t_1$ ).

# FISTA: proof

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

To ensure:

$$t_{k+1}(t_{k+1} - 1) = t_k^2 \quad (\leq \text{ if } x^* \text{ is a minimizer})$$

one can solve  $t_{k+1}^2 - t_{k+1} - t_k^2 = 0$  and take

$$t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2}$$

(observe that if  $t_0 \geq 0$ ,  $t_1 \geq 1$ ), or one can also show that  $t_k = (k + a - 1)/a$ ,  $a \geq 2$ , satisfies  $t_{k+1} \geq 1$  and  $t_{k+1}^2 - t_{k+1} \leq t_k^2$  for any  $k \geq 0$ .

# FISTA: proof

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

To ensure:

$$t_{k+1}(t_{k+1} - 1) = t_k^2 \quad (\leq \text{ if } x^* \text{ is a minimizer})$$

one can solve  $t_{k+1}^2 - t_{k+1} - t_k^2 = 0$  and take

$$t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2}$$

(observe that if  $t_0 \geq 0$ ,  $t_1 \geq 1$ ), or one can also show that  $t_k = (k + a - 1)/a$ ,  $a \geq 2$ , satisfies  $t_{k+1} \geq 1$  and  $t_{k+1}^2 - t_{k+1} \leq t_k^2$  for any  $k \geq 0$ .

To ensure:  $(t_{k+1} - 1)x^k + x^* - t_{k+1}y^k = (t_k - 1)x^{k-1} + x^* - t_k x^k$  one has to take, simply,

$$y^k = x^k + \frac{t_k - 1}{t_{k+1}}(x^k - x^{k-1}).$$

# FISTA: analysis

Continuous  
(convex)  
optimisation

A. Chambolle

Algorithms for  
monotone  
operators

Abstract problems  
Splitting methods

Descent  
algorithms

Forward-Backward  
Acceleration

Observe that

$$\frac{1 + \sqrt{1 + 4t_k^2}}{2} \geq \frac{1}{2} + t_k$$

hence if  $t_1 = 1$ ,  $t_k \geq (k + 1)/2$ . Then, the final bound shows, for  $t_0 = 0$  and  $\tau = 1/L$ :

$$F(x^N) - F(x^*) \leq \frac{2L}{(k + 1)^2} |x^0 - x^*|^2$$

which is “optimal”.