On a class of first-order algorithms for convex problems with applications to imaging

Antonin Chambolle

CMAP, Ecole Polytechnique, CNRS, Palaiseau, France[1]

Journée de Bilan, Chaire EADS/MMNS – October 13, 2010

[1] Joint work with Thomas Pock, T.U. Graz, Austria

Outline

Topic: a primal-dual first order algorithm which is simple and versatile

- the class of problems
- general algorithm
- $O(1/N)$ efficiency
- accelerations
- examples

A general form

$$\min_{x \in X} F(Kx) + G(x)$$

where

- F, G are "simple" *convex* functions
- $K : X \to Y$ is a linear, *continuous* operator from a Hilbert space X to a Hibert space Y
- The problem is **nonsmooth** (for instance, $F = |\cdot|_1$)

Remark In most interesting infinite-dimensional cases, K is *not* continuous, however, our complexity estimates will be dimension independent (but the number of operations depends of course on the size of the vectors...)

Example: image reconstruction

In image reconstruction, problems often involve

- very large arrays $\rightarrow$ high order methods are hard to use
- nonsmooth terms (l^1 norms, total variation...)

Example: TV-deconvolution vs Wiener filtering

(i) original (ii) motion blurred (iii) Wiener (iv) TV

"Simple" convex functions

We say that F is **simple** if one knows how to compute the "proximity" operator

$$\text{prox}_{\tau F}(x) \;=\; \arg\min_z \left\{ \tau F(z) + \frac{\|x-z\|^2}{2} \right\}$$

The Euler-Lagrange equation for the unique solution to this problem writes

$$x - z + \tau \partial F(z) \;\ni\; 0$$

where ∂F is the subgradient of F (which is of course not assumed to have any smoothness).
Hence one usually writes

$$\text{prox}_{\tau F}(x) \;=\; (I + \tau \partial F)^{-1}(x)$$

as the "resolvent" operator of the subgradient.

An important remark

Moreau's identity

$$x \;=\; (I+\tau\partial F)^{-1}(x) \;+\; \tau\left(I+\frac{1}{\tau}\partial F^*\right)^{-1}\left(\frac{x}{\tau}\right),$$

shows that F is simple iff F^* is, where F^* is the *Legendre-Fenchel conjugate* of F defined by

$$F^*(y) \;=\; \sup_x \langle y, x\rangle \;-\; F(x)$$

(And, $F^{**} = F$ if F convex, l.s.c...)

Typical problems

- "ROF" denoising

$$min_u \lambda TV(u) + \frac{1}{2}\|u - g\|^2$$

where
$TV(u) = \|\nabla u\|_{2,1} \approx \sum_{i,j} \sqrt{(u_{i+1,j} - u_{i,j})^2 + (u_{i,j+1} - u_{i,j})^2}$
is a discrete approximation of the total variation.
Here, $x = u$, $K = \nabla$, $F = \lambda\|\cdot\|_{2,1}$, $G = \|\cdot - g\|^2/2$ and we are in the presented framework. In particular, F is simple: the operator $\text{prox}_{\tau F}$ is a component-by-component "shrinkage" of size τ:

$$\arg\min_z \tau\|z\|_{1,2} + \frac{1}{2}\|z - z^0\|^2$$

is given by $z_{i,j} = (|z_{i,j}| - \tau)^+ z_{i,j}/|z_{i,j}|$.

- "ROF" deblurring

$$min_u \lambda TV(u) + \frac{1}{2}\|Au - g\|^2$$

can be cast into our framework in two different ways. Either one knows how to solve efficiently $\text{prox}_{\tau\|Au-g\|^2/2}$, that is,

$$\min_z \tau\|Az - g\|^2 + \|z - u\|^2$$

which require to compute $(I + \tau A^*A)^{-1}$ and we can simply let as before $x = u$, $K = \nabla$, $F = \lambda\|\cdot\|_{2,1}$, $G = \|A \cdot -g\|^2/2$.
A variant is to let $K = (\nabla, A) : X \to Y$, $G = 0$, and

$$F(y) \;=\; \lambda\|y^\nabla\|_{2,1} + \frac{1}{2}\|y^A - g\|^2$$

which is also "simple".

Why first order methods?

- Higher-order method such as Newton, for the above presented problems, need information on the second derivative, and smoothness, and require to invert huge matrices $\rightarrow$ need to regularize the problem; fewer iteration but the cost per iteration is huge
- Some first order methods work with no smoothness and there is a hope to "adapt" them automatically to the smoothness of the problem (work in progress...)

The dual problem

We have (cf Fenchel-Rockafellar)

$$\begin{aligned}\inf_x F(Kx) + G(x) &= \inf_x \sup_y \langle Kx, y\rangle - F^*(y) + G(x) \\ &= \sup_y -F^*(y) + \inf_x \langle x, K^*y\rangle + G(x) \\ &= \sup_y -(F^*(y) + G^*(-K^*y))\end{aligned}$$

and existence of a saddle-point $(\hat{x}, \hat{y})$ under quite mild conditions: for all $(x, y) \in X \times Y$:

$$\langle K\hat{x}, y\rangle - F^*(y) + G(\hat{x}) \leq \langle K\hat{x}, \hat{y}\rangle - F^*(\hat{y}) + G(\hat{x}) \leq \langle Kx, \hat{y}\rangle - F^*(\hat{y}) + G(x)$$

Euler-Lagrange Equation

The saddle point must satisfy

$$\begin{cases} K^*\hat{y} + \partial G(\hat{x}) \ \ni \ 0 \\ K\hat{x} - \partial F^*(\hat{y}) \ \ni \ 0 \end{cases}$$

which also may be written

$$\begin{pmatrix} 0 & K^* \\ -K & 0 \end{pmatrix} \begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix} + \begin{pmatrix} \partial G(\hat{x}) \\ \partial F^*(\hat{y}) \end{pmatrix} \ni \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

This yields a "primal-dual" form of the problem, which shows the symmetry between G and F^*, and from which interesting algorithms can be proposed.

Many methods...

There are numerous approaches to solve such problems (in smooth, nonsmooth, simple, high or low-dimensional cases...) and research has been very active since at least the 70's (and even before).
An important concept is the "Proximal point" algorithm (Martinet'70, Rockafellar'76):

$$\textit{Minimize the convex function } H \textit{ by iterating } (I + \tau \partial H)^{-1}$$

- Often conceptual since this might be as hard as solving $0 \in \partial H(x)$)
- However, many other algorithms can be shown to be particular instances of this approach, and hence convergent (e.g., Eckstein-Bertsekas 1992)

Primal-Dual Proximal Point

- The class we investigate here derives from the idea to apply the "proximal point" method to the Primal-Dual problem: the basic idea should be to choose $X^0 = (x^0, y^0) \in X \times Y$, pick $\tau > 0$, and let

$$\begin{cases} x^{n+1} = (I + \tau \partial G)^{-1}(x^n - \tau K^* y^{n+1}) \\ y^{n+1} = (I + \tau \partial F^*)^{-1}(y^n + \tau K x^{n+1}) \end{cases}$$

however in practice this is not implementable. An *explicit* approach (where x^{n+1}, y^{n+1} is replaced with x^n, y^n in the right-hand side) is shown to diverge. The *semi-implicit* scheme (with for instance x^n, y^{n+1}) is the standard "Arrow-Hurwicz" algorithm which is shown converges under strong assumptions (on the time-steps, or the domain of G, F^*) [Zhu-Chan'08,Esser'09,CP'10].

Primal-dual "extragradient"

It has been observed in the 70's (Korpelevich 76, Popov 80) that a slight modification of the primal-dual proximal point yields a convergent scheme. The idea is to take an "extragradient step", by choosing

$$\begin{cases} x^{n+\frac{1}{2}} = (I + \tau\partial G)^{-1}(x^n - \tau K y^n) \\ y^{n+\frac{1}{2}} = (I + \tau\partial F^*)^{-1}(y^n - \tau K x^n) \end{cases}$$

and then

$$\begin{cases} x^{n+1} = (I + \tau\partial G)^{-1}(x^n - \tau K y^{n+\frac{1}{2}}) \\ y^{n+1} = (I + \tau\partial F^*)^{-1}(y^n - \tau K x^{n+\frac{1}{2}}) \end{cases}$$

Estimates (of type $O(1/N)$) are shown by Nemirovsky (2004). [A forgotten Popov proposes a similar approach, 4 years after Korpelevich though]

A drawback of these approaches is that they require to evaluate $\text{prox}_{\tau F}$ and $\text{prox}_{\tau G}$ twice per iteration (which in fact could not be a real issue if the convergence were twice faster than variants not requiring this step, but we have checked experimentally that this is not the case).

A general "approximate" extra-gradient framework

Algorithm 1.

- Initialization: Choose $\tau, \sigma > 0$, $\theta \in [0,1]$, $(x^0, y^0) \in X \times Y$ and set $\bar{x}^0 = x^0$.
- Iterations ($n \geq 0$): Update $x^n, y^n, \bar{x}^n$ as follows:

$$\begin{cases} y^{n+1} &= (I + \sigma \partial F^*)^{-1}(y^n + \sigma K \bar{x}^n) \\ x^{n+1} &= (I + \tau \partial G)^{-1}(x^n - \tau K^* y^{n+1}) \\ \bar{x}^{n+1} &= x^{n+1} + \theta(x^{n+1} - x^n) \end{cases}$$

The (simple) idea, here, is to use a semi-implicit scheme in one variable (here, y), and an approximate "extra-gradient" (or over-relaxation) in the other variable where the "gradient" is simply replaced with the difference between the two last iterates.

Remark 1: in case $K = Id$ this is strictly identical to a "Douglas Rachford" splitting of ∂F and ∂G (PL Lions-Mercier, 1979)

Remark 2: when $\theta = 0$ then this is the basic Arrow-Hurwicz approach (1958). However, convergence is not clear in general (but apparently occurs more often and faster than actually proved, see e.g. Zhu-Chan'08). This will not be discussed in this talk (see CP'10).

"Partial gap"

We measure the convergence using a variant of the primal-dual gap

$$\mathcal{G}(x,y) \;=\; F(Kx) + G(x) + F^*(y) + G^*(-K^*y) \;\geq\; 0$$

which vanishes only when (x,y) is a saddle-point. For $B_1 \subset X$ and $B_2 \subset Y$ we let

$$\begin{aligned}\mathcal{G}_{B_1\times B_2}(x,y) \;=\; &\max_{y'\in B_2} \left\langle y', Kx\right\rangle - F^*(y') + G(x) \\ &- \min_{x'\in B_1} \left\langle y, Kx'\right\rangle - F^*(y) + G(x')\,,\end{aligned}$$

($\mathcal{G} = \mathcal{G}_{X\times Y}$). Then, if $B_1 \times B_2$ contains a saddle-point in its interior, this is non-negative and is a (not very good) measure of optimality for points in the interior of $B_1 \times B_2$.

Convergence analysis in case $\theta = 1$

Theorem Let $L = \|K\|$ and assume there is a saddle-point $(\hat{x}, \hat{y})$. Choose $\theta = 1$, $\tau\sigma L^2 < 1$, and let $(x_n, \bar{x}_n, y_n)$ be defined as in Algorithm 1. Then:

(a) For any n,

$$\frac{\|y^n - \hat{y}\|^2}{2\sigma} + \frac{\|x^n - \hat{x}\|^2}{2\tau} \leq C\left(\frac{\|y^0 - \hat{y}\|^2}{2\sigma} + \frac{\|x^0 - \hat{x}\|^2}{2\tau}\right)$$

where the constant $C \leq (1 - \tau\sigma L^2)^{-1}$;

(b) If we let $x_N = (\sum_{n=1}^{N} x^n)/N$ and $y_N = (\sum_{n=1}^{N} y^n)/N$, for any bounded $B_1 \times B_2 \subset X \times Y$ the restricted gap has the following bound:

$$\mathcal{G}_{B_1 \times B_2}(x_N, y_N) \;\leq\; \frac{D(B_1, B_2)}{N}\,,$$

where

$$D(B_1, B_2) = \sup_{(x,y) \in B_1 \times B_2} \frac{\|x - x^0\|^2}{2\tau} + \frac{\|y - y^0\|^2}{2\sigma}$$

Moreover, the weak cluster points of (x_N, y_N) are saddle-points.

(c) If the dimension of the spaces X and Y is finite, then there exists a saddle-point (x^*, y^*) such that $x^n \to x^*$ and $y^n \to y^*$.

Remarks: • Nemirovsky has a similar result but with a more complex (of Extragradient type) scheme. He uses the full primal-dual gap but this is only due to his assumption that the domains of G and F^* (then $B_1 = \mathrm{dom} G$, $B_2 = \mathrm{dom} F^*$), which we do not want to make here.

• We can show that we get an estimate of the global gap if F and G^* have full domain.

• This result shows convergence, but the rate, the way it converges, and the measure of optimality are terrible: in practice, the algorithm is too slow for some classes of problems (e.g., linear)

Convergence analysis (sketch)

Idea: the scheme has the general form

$$\begin{cases} y^{n+1} &= (I + \sigma \partial F^*)^{-1}(y^n + \sigma K \bar{x}) \\ x^{n+1} &= (I + \tau \partial G)^{-1}(x^n - \tau K^* \bar{y}) \end{cases},$$

that is:

$$\begin{cases} \partial F^*(y^{n+1}) \ni \dfrac{y^n - y^{n+1}}{\sigma} + K \bar{x} \\ \partial G(x^{n+1}) \ni \dfrac{x^n - x^{n+1}}{\tau} - K^* \bar{y} \end{cases}$$

By definition of the subgradient it means that for any x, y:

$$F^*(y) \geq F^*(y^{n+1}) + \left\langle \frac{y^n - y^{n+1}}{\sigma}, y - y^{n+1} \right\rangle + \left\langle K\bar{x}, y - y^{n+1} \right\rangle$$
$$G(x) \geq G(x^{n+1}) + \left\langle \frac{x^n - x^{n+1}}{\tau}, x - x^{n+1} \right\rangle - \left\langle K(x - x^{n+1}), \bar{y} \right\rangle ,$$

which we sum to obtain

$$\frac{\|y - y^n\|^2}{2\sigma} + \frac{\|x - x^n\|^2}{2\tau} \geq$$
$$\left[\left\langle Kx^{n+1}, y \right\rangle - F^*(y) + G(x^{n+1})\right] - \left[\left\langle Kx, y^{n+1} \right\rangle - F^*(y^{n+1}) + G(x)\right]$$
$$+ \frac{\|y - y^{n+1}\|^2}{2\sigma} + \frac{\|x - x^{n+1}\|^2}{2\tau} + \frac{\|y^n - y^{n+1}\|^2}{2\sigma} + \frac{\|x^n - x^{n+1}\|^2}{2\tau}$$
$$+ \left\langle K(x^{n+1} - \bar{x}), y^{n+1} - y \right\rangle - \left\langle K(x^{n+1} - x), y^{n+1} - \bar{y} \right\rangle .$$

The best would be to cancel the last term by choosing $\bar{y} = y^{n+1}$ and $\bar{x} = x^{n+1}$... but this totally implicit scheme is not implementable (it is a proximal point iteration).
Our choice for $\theta = 1$ corresponds to letting $y^{n+1} - \bar{y} = 0$ (semi-implicit) and $x^{n+1} - \bar{x} = (x^{n+1} - x^n) - (x^n - x^{n-1})$, from which we hope to get cancellations when we will sum the inequality from $n = 0$ to N.

Then, summing and using $\tau\sigma\|K\|^2 < 1$, we can show the thesis of the theorem. $\square$

Accelerations

In case ∇F is Lipschitz, Nesterov (1983/2007) and Beck-Teboulle (2008) have provided algorithms with $O(1/N^2)$ convergence (of the objective $F(Ax) + G(x)$). We can check that our approach yields a similar efficiency, provided it is slightly modified.
We recall that ∇F is $1/\delta-$Lipschitz if and only if F^* is $\delta-$uniformly convex:

$$F^*(y') \geq F^*(y) + \langle p, y' - y \rangle + \frac{\delta}{2}\|y - y'\|^2$$

for any y, y' and $p \in \partial F^*(y)$. In this case, one easily checks that $\hat{y}$ is unique (not $\hat{x}$). We can consider also the the symmetric case where G is uniformly convex with constant γ (or ∇G^* is $1/\gamma - Lipschitz$), in which case it is $\hat{x}$ which is unique.

G uniformly convex

In that case, when $(x, y) = (\hat{x}, \hat{y})$ is a saddle-point, the gap

$$\left[\left\langle Kx^{n+1}, \hat{y}\right\rangle - F^*(\hat{y}) + G(x^{n+1})\right] - \left[\left\langle K\hat{x}, y^{n+1}\right\rangle - F^*(y^{n+1}) + G(\hat{x})\right]$$

is easily shown to bound $(\gamma/2)\|x^{n+1} - \hat{x}\|^2$. In fact, our main estimate can be modified as follows:

$$\frac{\|\hat{y} - y^n\|^2}{2\sigma} + \frac{\|\hat{x} - x^n\|^2}{2\tau} \geq$$
$$+ \frac{\|\hat{y} - y^{n+1}\|^2}{2\sigma} + (1 + 2\gamma\tau)\frac{\|\hat{x} - x^{n+1}\|^2}{2\tau}$$
$$+ \text{etc}\ldots$$

The trick now is to use variable time-steps σ_n, τ_n and a variable factor θ_n, to make in particular the first lines of this equation look like

$$\frac{\|\hat{y} - y^n\|^2}{2\sigma_n} + \frac{\|\hat{x} - x^n\|^2}{2\tau_n} \geq$$
$$+ \frac{\sigma_{n+1}}{\sigma_n} \frac{\|\hat{y} - y^{n+1}\|^2}{2\sigma_{n+1}} + (1 + 2\gamma\tau_n)\frac{\tau_{n+1}}{\tau_n} \frac{\|\hat{x} - x^{n+1}\|^2}{2\tau_{n+1}}$$
$$+ \cdots$$

and it becomes evident that we can obtain something interesting if we can choose the sequences $(\tau_n, \sigma_n)_n$ in such a way that

$$\frac{\sigma_{n+1}}{\sigma_n} = (1 + 2\gamma\tau_n)\frac{\tau_{n+1}}{\tau_n} > 1\,.$$

This motivates a varying step variant of Alg. 1:

Algorithm 2.

- Initialization: Choose $\tau_0, \sigma_0 > 0$ with $\tau_0 \sigma_0 L^2 \leq 1$, $(x^0, y^0) \in X \times Y$, and $\bar{x}^0 = x^0$.
- Iterations ($n \geq 0$): Update $x^n, y^n, \bar{x}^n, \theta_n, \tau_n, \sigma_n$ as follows:

$$\begin{cases} y^{n+1} &= (I + \sigma_n \partial F^*)^{-1}(y^n + \sigma_n K \bar{x}^n) \\ x^{n+1} &= (I + \tau_n \partial G)^{-1}(x^n - \tau_n K^* y^{n+1}) \\ \theta_n = 1/\sqrt{1 + 2\gamma\tau_n}, \ \tau_{n+1} = \theta_n \tau_n, \ \sigma_{n+1} &= \sigma_n / \theta_n \\ \bar{x}^{n+1} &= x^{n+1} + \theta_n (x^{n+1} - x^n) \end{cases}$$

Estimate for Algorithm 2

Theorem Choose $\tau_0 > 0$, $\sigma_0 = 1/(\tau_0 L^2)$, and let $(x^n, y^n)_{n\geq 1}$ be defined by Algorithm 2: then,

$$\|\hat{x} - x^N\|^2 \leq \tau_N^2 \left(\frac{\|\hat{x} - x^0\|^2}{\tau_0^2} + L^2 \|\hat{y} - y^0\|^2 \right).$$

So what? in fact, one can check that $\tau_n \approx 1/(\gamma n)$ for n (not too) large, so that this is essentially a $O(1/N^2)$ convergence result.
It is less good than the estimate of Nesterov/Beck-Teboulle — which here would estimate the dual objective $G^*(-A^* y^N) + F^*(y^N) - (G^*(-A^* \bar{y}) + F^*(\bar{y}))$, which is shown to control $\|x^N - \hat{x}\|^2$. However in practice we found our method simpler and more efficient...

An interesting behaviour

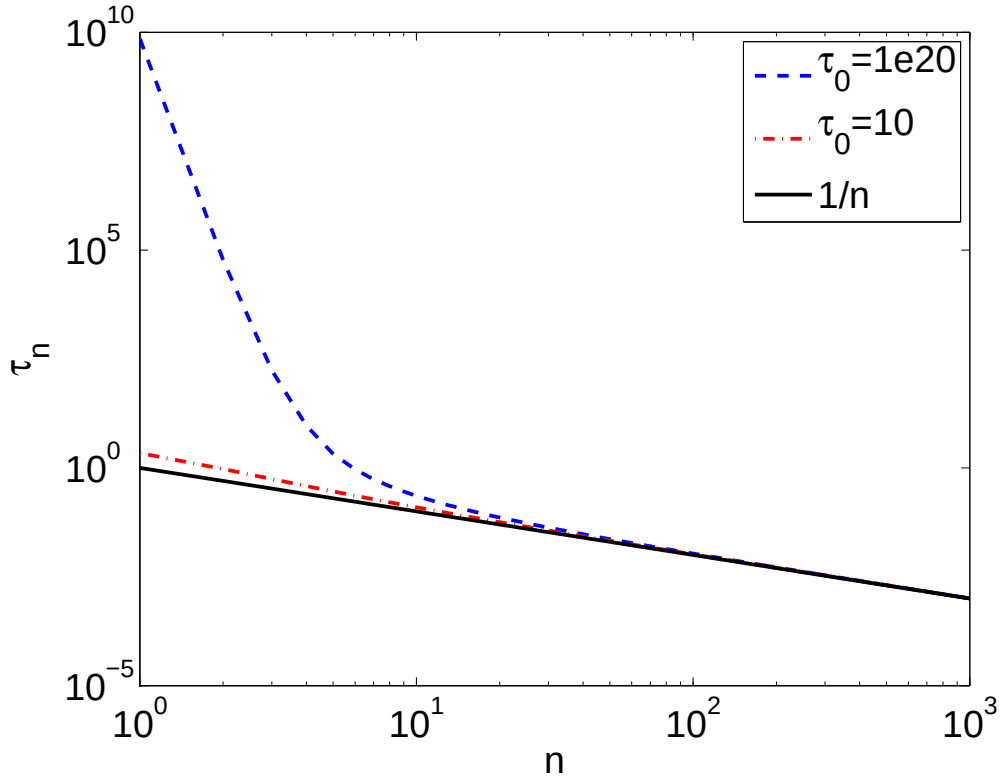

Figure: The figure shows the sequence $\tau'_n = \gamma\tau_n$ for $n \geq 1$, Observe that it behaves very fast like $1/n$, in a way which is quite insensitive to the initial τ'_0

More regularity and more acceleration

It is well known that if both G and F^* are uniformly convex (or both F and G^* are $C^{1,1}$), resp. with parameter γ and δ, then one can find strategies which yield linear convergence to the (unique) saddle-point. It turns out that it is also the case for the proposed approach.

Algorithm 3

- Initialization: Choose $\mu \leq 2\sqrt{\gamma\delta}/L$, $\tau = \mu/(2\gamma)$, $\sigma = \mu/(2\delta)$, and $\theta \in [1/(1+\mu), 1]$. Let $(x^0, y^0) \in X \times Y$, and $\bar{x}^0 = x^0$.
- Iterations ($n \geq 0$): Update $x^n, y^n, \bar{x}^n$ as follows:

$$\begin{cases} y^{n+1} &= (I + \sigma\partial F^*)^{-1}(y^n + \sigma K\bar{x}^n) \\ x^{n+1} &= (I + \tau\partial G)^{-1}(x^n - \tau K^* y^{n+1}) \\ \bar{x}^{n+1} &= x^{n+1} + \theta(x^{n+1} - x^n) \end{cases}$$

Estimate

Theorem Consider the sequence (x^n, y^n) provided by Algorithm 3. Let

$$\omega = \frac{1+\theta}{2+\mu} = \frac{1+\theta}{2\left(1+\frac{\sqrt{\gamma\delta}}{L}\right)} < 1.$$

Then

$$\gamma\|x^N-\hat{x}\|^2+(1-\omega)\delta\|y^N-\hat{y}\|^2 \leq \omega^N\left(\gamma\|x^0-\hat{x}\|^2+\delta\|y^0-\hat{y}\|^2\right)$$

The interesting point here is that in particular the *same* Algorithm 1, with an appropriate choice of τ and σ (and $\theta = 1$, although $1/(1+\mu)$ is theoretically better), converges linearly. An interesting open question is whether there is some intrinsic way to "guess" the parameters γ and δ when they are unknown, that is, to have some meta-algorithm which functions as the "best" of the three algorithms we have presented according to the situation.

Example: ROF denoising

The problem is

$$\min_u \|\nabla u\|_{2,1} + \frac{\lambda}{2}\|u-g\|^2$$

and $F = \|\cdot\|_{2,1}$, $G = \frac{\lambda}{2}\|u-g\|^2$ is $\lambda-$uniformly convex so that Alg. 2 can be used.

	$\lambda = 16$		$\lambda = 8$	
	$\varepsilon = 10^{-4}$	$\varepsilon = 10^{-6}$	$\varepsilon = 10^{-4}$	$\varepsilon = 10^{-6}$
ALG1	214 (3.38s)	19544 (318.35s)	309 (5.20s)	24505 (392.73s)
ALG2	108 (1.95s)	937 (14.55s)	174 (2.76s)	1479 (23.74s)
AHMOD	64 (0.91s)	498 (6.99s)	122 (1.69s)	805 (10.97s)
AHZC	65 (0.98s)	634 (9.19s)	105 (1.65s)	1001 (14.48s)
FISTA	107 (2.11s)	999 (20.36s)	173 (3.84s)	1540 (29.48s)
NEST	106 (3.32s)	1213 (38.23s)	174 (5.54s)	1963 (58.28s)
ADMM	284 (4.91s)	25584 (421.75s)	414 (7.31s)	33917 (547.35s)
PGD	620 (9.14s)	58804 (919.64s)	1621 (23.25s)	–
CFP	1396 (20.65s)	–	3658 (54.52s)	–

Table: (Matlab) Performance evaluation using a 256×256 image. The entries in the table refer to the number of iterations respectively the CPU times in seconds the algorithms needed to drop the RMSE of the solution below the error tolerance ε. The "–" entries indicate that the algorithm failed to drop the error below ε within a maximum number of 100000 iterations.

The 3 leading methods

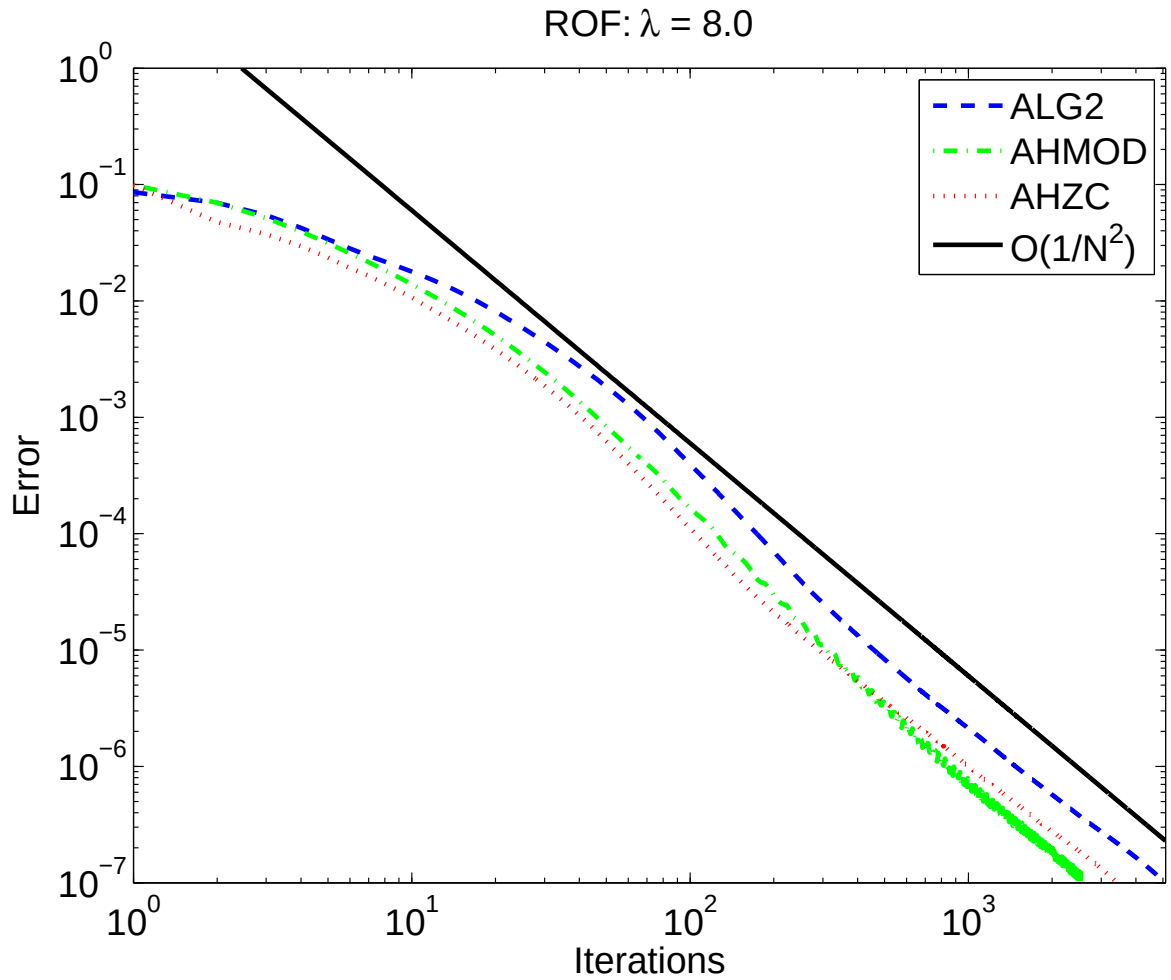

Figure: Convergence of AHZC and ALG2 for the experiment in the last column of the Table

"Huber-ROF"

Now we investigate the minimization of

$$\min_u \|\nabla u\|_{1,2}^{\alpha} + \frac{\lambda}{2}\|u-g\|^2 \ .$$

where

$$F(y) \ = \ \|y\|_{1,2}^{\alpha} \ = \ \sum_{i,j} |\vec{y}_{i,j}|_{\alpha}$$

and

$$|\vec{p}|_{\alpha} \ = \ \begin{cases} \frac{|p|^2}{2\alpha} & \text{if } |p| \leq \alpha \\ |p| - \frac{\alpha}{2} & \text{else.} \end{cases}$$

It turns out that with this new model, F^* is α-uniformly convex. Since $G(u) = (\lambda/2)\|u-g\|^2$ is λ-uniformly convex we can use Algo. 3. Experiments show that we reach machine precision in a quite short number of iterations. We get a similar behaviour with a restarted variant of Nesterov's algorithm, which is shown to have the same convergence rate.

	$\lambda = 5,\ \alpha = 0.05$
	$\varepsilon = 10^{-15}$
ALG3	187 (3.85s)
NEST	248 (5.52s)

Table: Performance evaluation using the image same image as for ROF denoising. The entries in the table refer to the number of iterations respectively the CPU times in seconds the algorithms needed to drop the root mean squared error below the error tolerance ε.

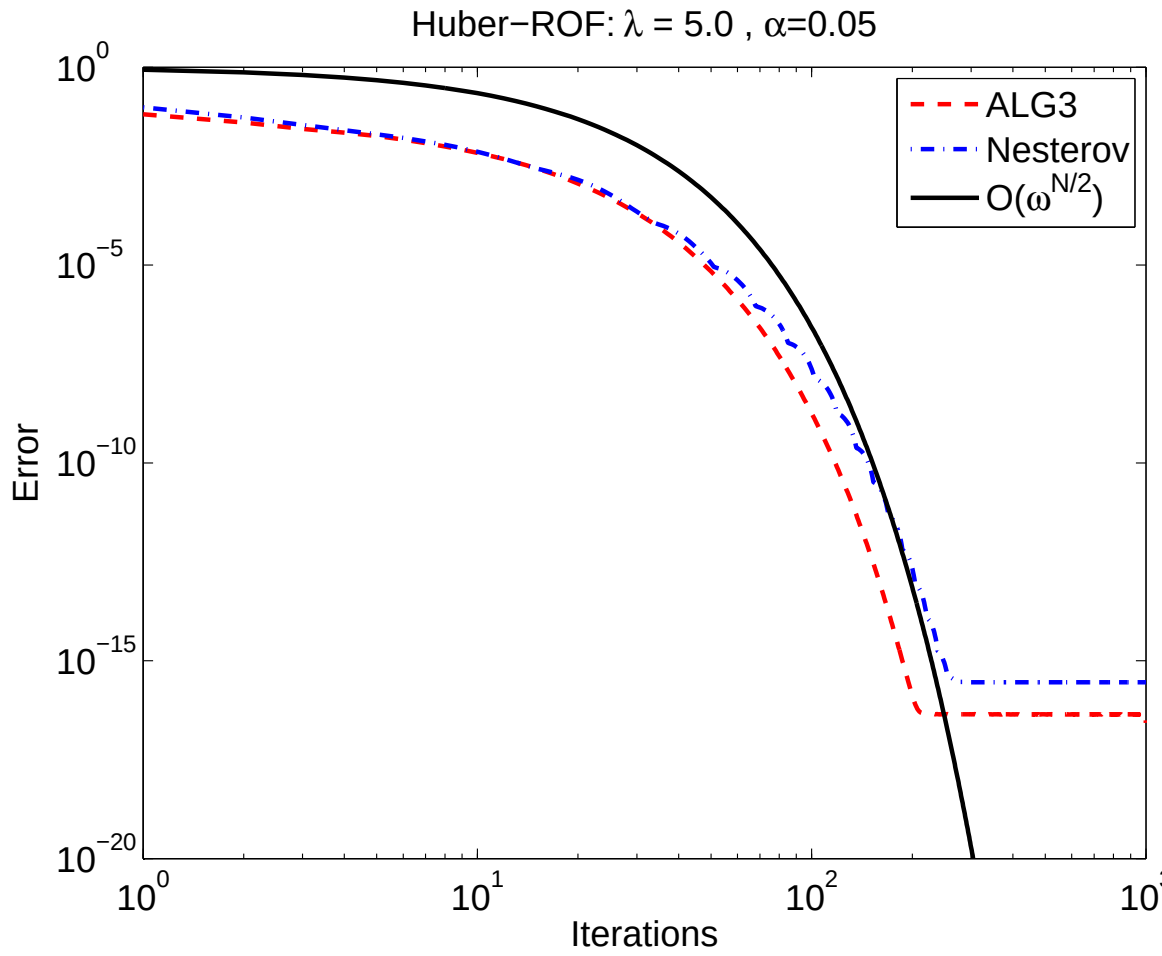

Figure: Linear convergence of ALG3 and NEST for the Huber-ROF model. Note that after approximately 200 iterations, ALG3 reaches machine precision.

Thank you for your attention.