Recent Developments (and Some History) in Iterative Thresholding Algorithms

Mário A. T. Figueiredo

Instituto de Telecomunicações
and
Instituto Superior Técnico,
Technical University of Lisbon

PORTUGAL

mario.figueiredo@lx.it.pt

www.lx.it.pt/~mtf

Joint work with: M. Afonso, J. Bioucas-Dias, R. Nowak, S. Wright.

Signal Restoration/Representation/Reconstruction

Many signal/image reconstruction/approximation criteria have the form

$$\min_{\mathbf{x}\in\mathbb{R}^n} \phi(\mathbf{x}) := f(\mathbf{x}) + \tau c(\mathbf{x})$$

$f : \mathbb{R}^n \to \mathbb{R}$ is smooth and convex (the data fidelity term); often

$$f(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2$$

e.g., modelling linear observations with additve Gaussian noise.

$c : \mathbb{R}^n \to \bar{\mathbb{R}}$ is a regularization/penalty function (e.g., ℓ_1 or TV);

typically convex (sometimes not), often non-differentiable.

Examples: TV- and wavelet-based ($\mathbf{A} = \mathbf{HW}$) restoration/reconstruction, sparse representations, sparse (linear or logistic) regression, compressive sensing ($\mathbf{A} = \mathbf{HW}$)

Outline

1. The optimization problem (previous slide)
2. IST Algorithms: 4 derivations
3. Convergence results
4. Enhanced (fast) versions: TwIST and SpaRSA
5. Warm starting and continuation
6. Other (even faster) IST-type algorithms: variable splitting
7. Dealing with non-Gaussian data (e.g., Poisson)

Joint work with: M. Afonso, J. Bioucas-Dias, R. Nowak, S. Wright.

Denoising/shrinkage operators

$$\min_{\mathbf{x}\in\mathbb{R}^n} \phi(\mathbf{x}) := \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2^2 + \tau c(\mathbf{x})$$

If $\mathbf{A}=\mathbf{I}$ (a denoising problem), c is proper and convex, then ϕ is strictly convex, thus there is a unique minimizer.

The so-called shrinkage/thresholding/denoising function

$$\Psi_\lambda(\mathbf{u}) = \arg\min_{\mathbf{z}} \frac{1}{2}\|\mathbf{z}-\mathbf{u}\|_2^2 + \lambda c(\mathbf{z})$$

is well defined (*Moreau proximal mapping*) [Moreau 1962], [Combettes 2001]

Examples:

$$c(\mathbf{z}) = \|\mathbf{z}\| \;\Rightarrow\; \Psi_\lambda(\mathbf{z}) = (\mathbf{I} - P_{\lambda S_{c^*}})\mathbf{z}$$

$$c(\mathbf{z}) = \|\mathbf{z}\|_1 \;\Rightarrow\; \Psi_\lambda(\mathbf{z}) = \mathrm{soft}(\mathbf{z},\lambda)$$

$$c(\mathbf{z}) = \|\mathbf{z}\|_\infty \;\Rightarrow\; \Psi_\lambda(\mathbf{z}) = (\mathbf{I} - P_{B_\lambda^{\ell_1}})\mathbf{z}$$

(not convex, not norm) $$c(\mathbf{z}) = \|\mathbf{z}\|_0 \;\Rightarrow\; \Psi_\lambda(\mathbf{z}) = \mathrm{hard}(\mathbf{z},\lambda)$$

Iterative Shrinkage/Thresholding (IST)

Problem: $$\min_{\mathbf{x}\in\mathbb{R}^n} \phi(\mathbf{x}) := \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2^2 + \tau c(\mathbf{x})$$

IST algorithm: $$\mathbf{x}^{k+1} = \Psi_{\tau/\alpha}\left(\mathbf{x}^k - \frac{1}{\alpha}\mathbf{A}^T(\mathbf{A}\mathbf{x}^k - \mathbf{y})\right)$$

Gradient $\nabla \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2^2\Big|_{\mathbf{x}^k}$

Products by $\mathbf{A}$ and $\mathbf{A}^T$ should be inexpensive (via FFT, DWT); e.g. $\mathbf{A} = \mathbf{H}\mathbf{W}$

convex set

If $c(\mathbf{x}) = \iota_C(\mathbf{x}) = \begin{cases} 0 & \Leftarrow \quad \mathbf{x}\in C \\ +\infty & \Leftarrow \quad \mathbf{x}\notin C \end{cases}$ then $\Psi_\lambda(\mathbf{z}) = P_C(\mathbf{z})$

IST becomes simply the projected gradient algorithm
(recent example in [Daubechies, Fornasier, Loris, 2007], with C an ℓ_1 ball).

IST as Expectation-Maximization [F. and Nowak, 2001, 2003]

Underlying observation model: $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}, \quad \mathbf{n} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$

Equivalent model: $\mathbf{y} = \mathbf{A}(\mathbf{x} + \mathbf{n}_1) + \mathbf{n}_2, \quad \mathbf{n}_1 \sim \mathcal{N}(\mathbf{0}, \mathbf{I}/\alpha)$

$$\mathbf{n}_2 \sim \mathcal{N}(\mathbf{0}, \mathbf{I} - \mathbf{A}\mathbf{A}^T/\alpha)$$

Hidden data: $\mathbf{z} = \mathbf{x} + \mathbf{n}_1, \quad p(\mathbf{y}|\mathbf{z}) = \mathcal{N}(\mathbf{y}|\mathbf{A}\mathbf{z}, \mathbf{I} - \mathbf{A}\mathbf{A}^T/\alpha)$

$$p(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mathbf{z}|\mathbf{x}, \mathbf{I}/\alpha)$$

E-step: $\mathbf{z}^k = \mathbb{E}[\mathbf{z}|\mathbf{y}, \mathbf{x}^k] = \mathbf{x}^k + \mathbf{A}^T(\mathbf{y} - \mathbf{A}\mathbf{x}^k)/\alpha$ (Wiener)

M-step: $\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} \frac{\alpha}{2}\|\mathbf{z}^k - \mathbf{x}\|_2^2 + \tau c(\mathbf{x}) = \Psi_{\tau/\alpha}(\mathbf{z}^k)$

$\lambda_{\max}(\mathbf{A}^T\mathbf{A}) \leq \alpha \Rightarrow$ Monotonicity ($\mathbf{I} - \mathbf{A}\mathbf{A}^T/\alpha$ is a valid covariance)

IST as Majorization-Minimization [De Mol and Defrise, 2002]

Majorization function: $\arg\min_{\mathbf{x}} Q(\mathbf{x},\mathbf{y}) - \phi(\mathbf{x}) = \mathbf{y} \qquad (a)$

MM algorithm: $\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} Q(\mathbf{x},\mathbf{x}^k) \qquad (b)$

Monotonicity: $Q(\mathbf{x}^{k+1},\mathbf{x}^k) - \phi(\mathbf{x}^{k+1}) \overset{(a)}{\geq} Q(\mathbf{x}^k,\mathbf{x}^k) - \phi(\mathbf{x}^k)$

$$Q(\mathbf{x}^{k+1},\mathbf{x}^k) \overset{(b)}{\leq} Q(\mathbf{x}^k,\mathbf{x}^k)$$

$$(a) \wedge (b) \Rightarrow \phi(\mathbf{x}^{k+1}) \leq \phi(\mathbf{x}^k)$$

If $\lambda_{\max}(\mathbf{A}^T\mathbf{A}) \leq \alpha$, we can set $Q(\mathbf{x},\mathbf{x}^k) = \frac{\alpha}{2}\|\mathbf{x}-\mathbf{z}^k\|_2^2 + \tau c(\mathbf{x})$

Thus, $\mathbf{x}^{k+1} = \Psi_{\tau/\alpha}(\mathbf{z}^k)$

$$\mathbf{z}^k = \mathbf{x}^k + \frac{1}{\alpha}\mathbf{A}^T(\mathbf{y}-\mathbf{A}\mathbf{x}^k)$$

IST as Forward-Backward Splitting [Combettes and Wajs, 2003]

$$\Psi_\tau(\mathbf{u}) = \mathbf{a} \quad \Leftrightarrow \quad \mathbf{a} = \arg\min_{\mathbf{z}} \frac{1}{2}\|\mathbf{z}-\mathbf{u}\|_2^2 + \tau c(\mathbf{z})$$

$$\Leftrightarrow \quad \mathbf{0} \in \tau\, \partial c(\mathbf{a}) + (\mathbf{a}-\mathbf{u})$$

$$\Leftrightarrow \quad \mathbf{u} \in (\mathbf{I} + \tau\, \partial c)\mathbf{a}$$

$$\Leftrightarrow \quad \mathbf{a} = (\mathbf{I} + \tau\, \partial c)^{-1}\mathbf{u} = \Psi_\tau(\mathbf{u})$$

(the minimizer is unique)

Back to the problem $\widehat{\mathbf{x}} \in \arg\min_{\mathbf{x}} f(\mathbf{x}) + \tau c(\mathbf{x})$

f differentiable

c convex

$$\Leftrightarrow \quad \mathbf{0} \in \nabla f(\widehat{\mathbf{x}}) + \tau\, \partial c(\widehat{\mathbf{x}}) + (\widehat{\mathbf{x}} - \widehat{\mathbf{x}})\alpha$$

$$\Leftrightarrow \quad (\alpha\mathbf{I} - \nabla f)\widehat{\mathbf{x}} \in (\alpha\mathbf{I} + \tau\, \partial c)\widehat{\mathbf{x}}$$

$$\Leftrightarrow \quad \widehat{\mathbf{x}} \in (\alpha\mathbf{I} + \tau\, \partial c)^{-1}(\alpha\mathbf{I} - \nabla f)\widehat{\mathbf{x}}$$

$$\Leftrightarrow \quad \widehat{\mathbf{x}} = \Psi_{\tau/\alpha}(\widehat{\mathbf{x}} - \nabla f(\widehat{\mathbf{x}})/\alpha)$$

(fixed point equation)

Fixed point scheme: $\mathbf{x}^{k+1} = \Psi_{\tau/\alpha}(\widehat{\mathbf{x}}^k - \frac{1}{\alpha}\nabla f(\mathbf{x}^k))$

IST as Separable Approximation [Wright, Nowak, and F., 2008]

Recall the problem: $\min_{\mathbf{x}\in\mathbb{R}^n} \phi(\mathbf{x}) := f(\mathbf{x}) + \tau c(\mathbf{x})$

Iteration:

Separable approximation to $f(\mathbf{z})$

$$(\mathbf{u}^{k+1}, \mathbf{v}^{k+1}) = \arg\min_{\mathbf{u},\mathbf{v}} f(\mathbf{u}) + g(\mathbf{v}) + \frac{\alpha}{2}\|\mathbf{G}\,\mathbf{u} - \mathbf{v} - \mathbf{d}^k\|_2^2$$

Can be re-written as $\mathbf{x}^{k+1} \in \arg\min_{\mathbf{z}} \frac{\alpha_k}{2}\|\mathbf{z} - \mathbf{z}^k\|_2^2 + \tau\, c(\mathbf{z})$

$$\mathbf{z}^k = \mathbf{x}^k - \frac{1}{\alpha_k}\nabla f(\mathbf{x}^k)$$

Thus, with c convex, $\mathbf{x}^{k+1} = \Psi_{\tau/\alpha_k}(\mathbf{z}^k)$

Bibliographical Notes

IST as expectation-maximization: [F. and Nowak, 2001, 2003]

IST as majorization-minimization: [De Mol, Defrise, 2002], [Daubechies, Defrise, De Mol, 2004]
[F., Nowak, Bioucas-Dias, 2005, 2007]

Forward-backward schemes in math: [Bruck, 1977], [Passty, 1979], [Lions and Mercier, 1979]

Forward-backward schemes in signal recovery: [Combettes and Wajs, 2003, 2004]

Separable approximation: [Wright, Nowak, and F., 2008]

Other authors independently proposed IST-like schemes for signal/image recovery:
[Bect, Blanc-Féraud, Aubert, and Chambolle, 2004],
[Elad, Matalon, and Zibulevsky, 2006],
[Hale, Yin, Zhang, 2007],
[Starck, Nguyen, Murtagh, 2003],
[Starck, Candès, Donoho, 2003],

Convergence Results (I)

Problem: $\min_{\mathbf{x}\in\mathbb{R}^n} \phi(\mathbf{x}) := \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2^2 + \tau c(\mathbf{x})$

IST algorithm: $\mathbf{x}^{k+1} = \Psi_{\tau/\alpha}\left(\mathbf{x}^k - \frac{1}{\alpha}\mathbf{A}^T(\mathbf{A}\mathbf{x}^k - \mathbf{y})\right)$

[Daubechies, Defrise, De Mol, 2004]: (applies in a Hilbert space setting)

Let $c(\mathbf{x}) = \|\mathbf{x}\|_p^p, \quad p \in [1,\, 2]$, and $\alpha > \|\mathbf{A}\|_2^2$.

Then, IST converges to a minimizer of ϕ

Convergence Results (II)

Problem: $$\min_{\mathbf{x}\in\mathbb{R}^n} \phi(\mathbf{x}) := f(\mathbf{x}) + \tau c(\mathbf{x})$$

IST algorithm: $$\mathbf{x}^{k+1} = \Psi_{\tau/\alpha_k}\left(\mathbf{x}^k - \frac{1}{\alpha_k}\nabla f(\mathbf{x}^k)\right)$$

[Combettes and Wajs, 2005]:

Let c be convex and proper (never $-\infty$, not $+\infty$ everywhere);

Let f have γ-Lipschitz continuous gradient, and $\frac{\gamma}{2} < \alpha_k < +\infty$

Then, IST converges to a minimizer of ϕ

For $f(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2$, the condition is $\frac{\|\mathbf{A}\|_2^2}{2} < \alpha_k < +\infty$

thus weaker than in [DDD, 2004].

Convergence Results (III)

Problem: $\min_{\mathbf{x}\in\mathbb{R}^n} \phi(\mathbf{x}) := f(\mathbf{x}) + \tau c(\mathbf{x})$

IST algorithm: $\mathbf{x}^{k+1} = \Psi_{\tau/\alpha}\left(\mathbf{x}^k - \frac{1}{\alpha}\nabla f(\mathbf{x}^k)\right)$

[Hale, Yin, Zhang, 2007]:

Let $c(\mathbf{x}) = \|\mathbf{x}\|_1$ and $\alpha > \frac{\gamma}{2}$

Then, IST converges to some $\mathbf{x}^* \in G = \arg\min_{\mathbf{x}} \phi(\mathbf{x})$

Moreover, for all but a finite number of iterations:

$$x_i^k = x_i^* = 0, \quad \forall i \in L$$

$$\text{sign}\left[\left(\mathbf{x}^k - \nabla f(\mathbf{x}^k)/\alpha\right)_i\right] = \text{sign}\left[\left(\mathbf{x}^* - \nabla f(\mathbf{x}^*)/\alpha\right)_i\right], \quad \forall i \in E$$

where $S_c = \{\mathbf{x} : \|\mathbf{x}\|_1 \leq 1\}$

Accelerating IST: Two-Step IST (TwIST)

IST becomes slow when $\mathbf{A}$ is very ill-conditioned and τ is small

Inspired by two-step method for linear systems [Frankel, 1950], [Axelsson, 1996],

TwIST algorithm [Bioucas-Dias and F., 2007]:

$$\mathbf{x}^{k+1} = (\delta - \beta)\mathbf{x}^k + (1-\delta)\mathbf{x}^{k-1} + \beta\,\Psi_\tau\left(\mathbf{x}^k + \mathbf{A}^T(\mathbf{y} - \mathbf{A}\mathbf{x}^k)\right)$$

Simplified analysis with $0 < m \leq \lambda_{\min}(\mathbf{A}^T\mathbf{A}) \leq \lambda_{\max}(\mathbf{A}^T\mathbf{A}) = 1$

The minimizer $\widehat{\mathbf{x}}$ is unique and TwIST converges to $\widehat{\mathbf{x}}$, $\lim_{t\to\infty} \|\mathbf{x}^t - \widehat{\mathbf{x}}\| = 0.$

There is an optimal choice for δ and β for which

$$\|\mathbf{x}^{t+1} - \widehat{\mathbf{x}}\| \leq \frac{1-\sqrt{m}}{1+\sqrt{m}}\,\|\mathbf{x}^t - \widehat{\mathbf{x}}\|$$

Accelerating IST: TwIST (II)

A one-step method is recovered for $\delta = 1$

$$\mathbf{x}^{t+1} = (1-\beta)\mathbf{x}^t + \beta\,\Psi_\tau\left(\mathbf{x}^t + \mathbf{K}^T(\mathbf{y} - \mathbf{K}\mathbf{x}^t)\right)$$

which is an over-relaxed version of the original IST.

For the optimal choice of β : $\|\mathbf{x}^{t+1} - \widehat{\mathbf{x}}\| \leq \dfrac{1-m}{1+m}\,\|\mathbf{x}^t - \widehat{\mathbf{x}}\|$

$-1/\log_{10}\dfrac{1-m}{1+m}$ ~ number of iterations to decrease error by factor of 10.

Example:

$$m = 10^{-3} \quad \rightarrow \quad -1/\log\frac{1-m}{1+m} \sim 1150 \qquad -1/\log\frac{1-\sqrt{m}}{1+\sqrt{m}} \sim 35$$

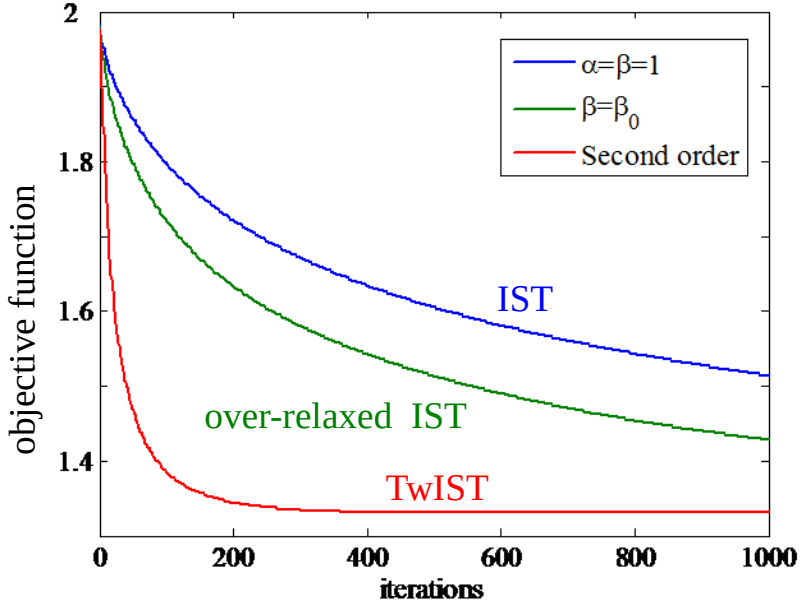
Another two-step method was recently proposed in [Beck and Teboulle, 2008]

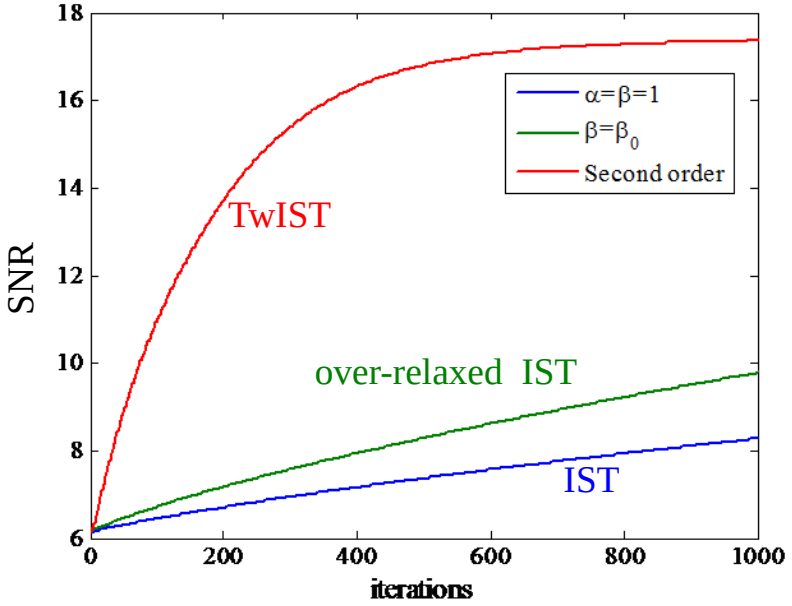
Experiments with TwIST (Total-Variation Regularization)

original

blurred, 9x9, 40db noise

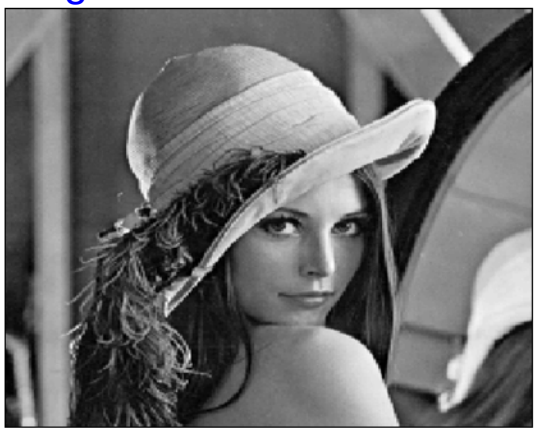
restored



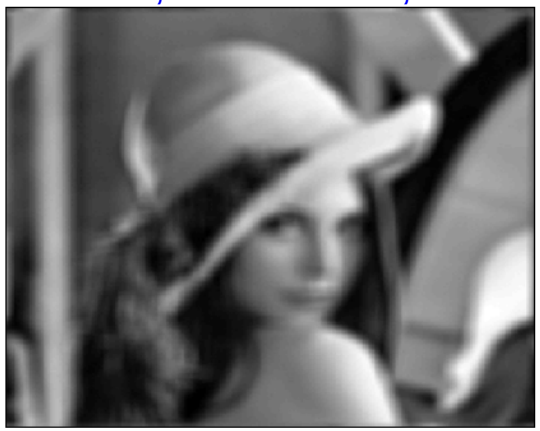


Experiments with TwIST (Total-Variation Regularization)

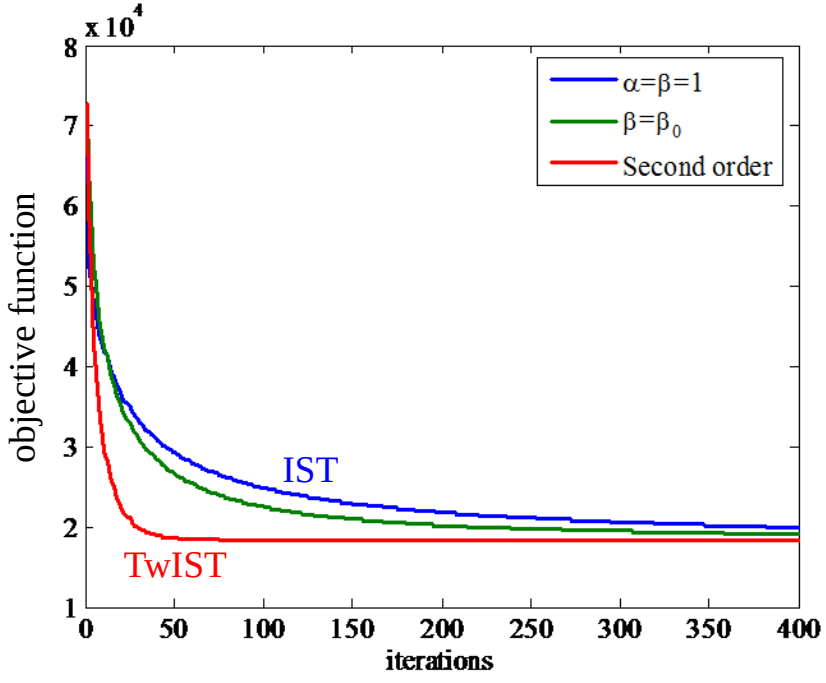
original

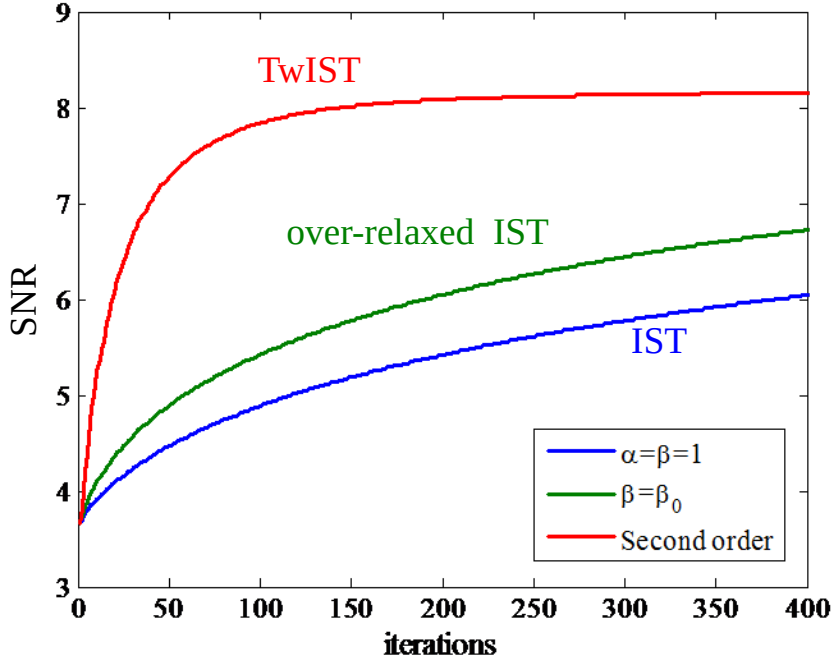


Blurred, 9x9 uniform, 40db



restored

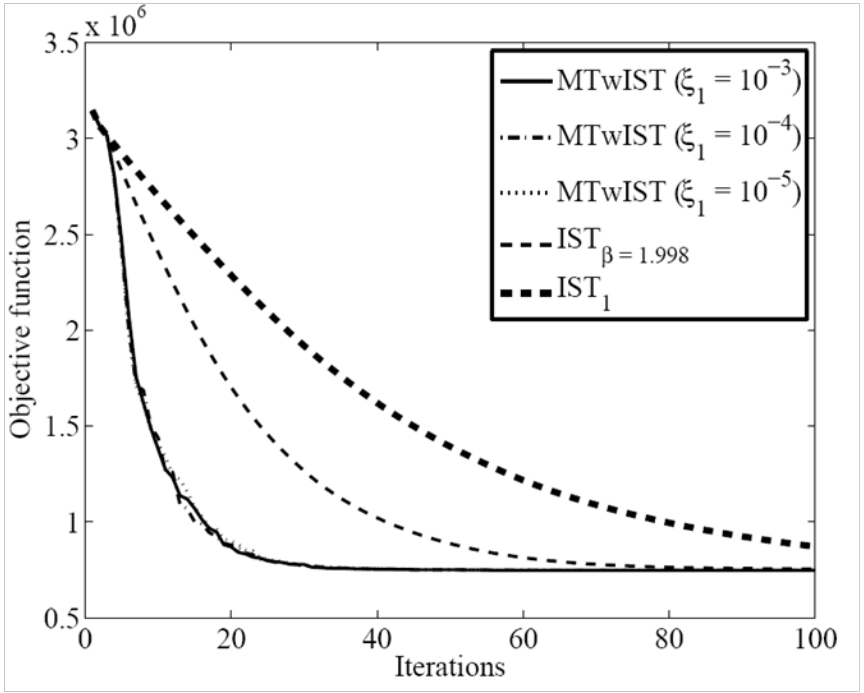




Experiments with TwIST (Total-Variation Regularization)

40% missing samples

Monotone TwIST (MTwIST): enforce monotonicity.



Accelerating IST: The SpaRSA Algorithmic Framework

Initialization: choose $\eta > 1$, $\alpha_{\min} \ll \alpha_{\max}$, and $\mathbf{x}^0$; set $k \leftarrow 0$

repeat:

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

repeat:

$$\mathbf{x}^{k+1} \leftarrow \Psi_{\tau/\alpha_k}\left(\mathbf{x}^k - \frac{1}{\alpha_k}\nabla f(\mathbf{x}^k)\right)$$

$$\alpha_k \leftarrow \eta\,\alpha_k$$

until $Acc(\mathbf{x}^{k+1}) == 1$ (* acceptance criterion *)

$$k \leftarrow k + 1$$

until stopping criterion is satisfied.

[Wright, Nowak, F., 2008]

Variants of SpaRSA are distinguished by the choice of α_k, Ψ_λ, and Acc

Examples: $Acc = 1$, $\alpha_k = \alpha$ yields standard IST.

$Acc(\mathbf{x}^{k+1}, \mathbf{x}^k) = 1_{\phi(\mathbf{x}^{k+1}) < \phi(\mathbf{x}^k)}$ yields monotone SpaRSA

Choosing α_k for Speed

The Barzilai-Borwein approach: seek α_k to mimic a Newton step,

a less conservative choice than in IST:

$$\alpha_k \, \mathbf{I} \simeq \nabla^2 f(\mathbf{x})$$

With a least-squares criterion over the last step,

$$\alpha_k = \arg\min_{\alpha} \left\| \alpha(\mathbf{x}^k - \mathbf{x}^{k-1}) - (\nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^{k-1})) \right\|_2^2$$

If $f(\mathbf{x}) = \frac{1}{2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2$, then $\alpha_k = \dfrac{\|\mathbf{A}(\mathbf{x}^k - \mathbf{x}^{k-1})\|_2^2}{\|\mathbf{x}^k - \mathbf{x}^{k-1}\|_2^2}$

Alternative rule (SpaRSA-monotone): $\alpha_k = \beta \, \alpha_{k-1}$, with $\beta < 1$

Benchmark Compressed Sensing Experiment

$$f(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2 \qquad c(\mathbf{x}) = \|\mathbf{x}\|_1$$

$\mathbf{A}$ 2^{10} x 2^{12} random (Gaussian), $\mathbf{x}$ 160 randomly located non-zeros

$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}$, where $\mathbf{e} \sim \mathcal{N}(0, 10^{-4})$

Algorithm	CPU time (secs.)	MSE
SpaRSA	0.33	2.89e-3
SpaRSA-monotone	0.34	2.91e-3
GPSR-BB-monotone	0.42	2.92e-3
GPSR-Basic	0.67	2.93e-3
FPC	1.55	2.95e-3
`l1_ls`	9.80	2.96e-3
AC	2.83	2.91e-3
TwIST	0.63	2.91e-3

[F., Nowak, Wright, 2007]

[Hale, Yin, Zhang, 2007]

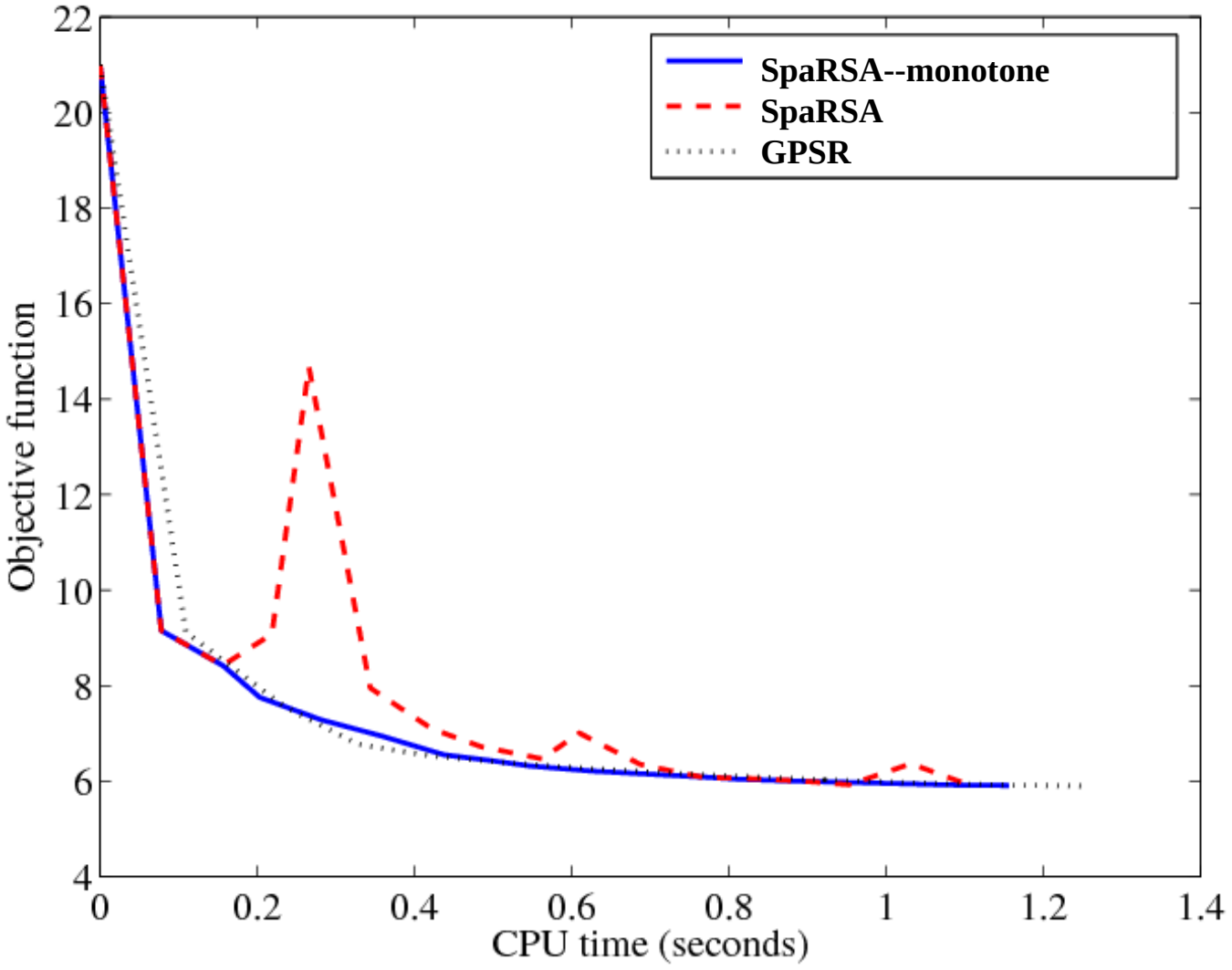
[Kim, Koh, Lustig, Boyd, Gorinvesky, 2007]

[Nesterov, 2007]

[Bioucas-Dias, F., 2007]

GPSR and *l1_ls* are "hardwired" for $c(\mathbf{x}) = \|\mathbf{x}\|_1$

Non-monotonicity



Empirical Complexity Exponents

Experiment proposed in [Kim, Koh, Lustig, Boyd, Gorinvesky, 2007]

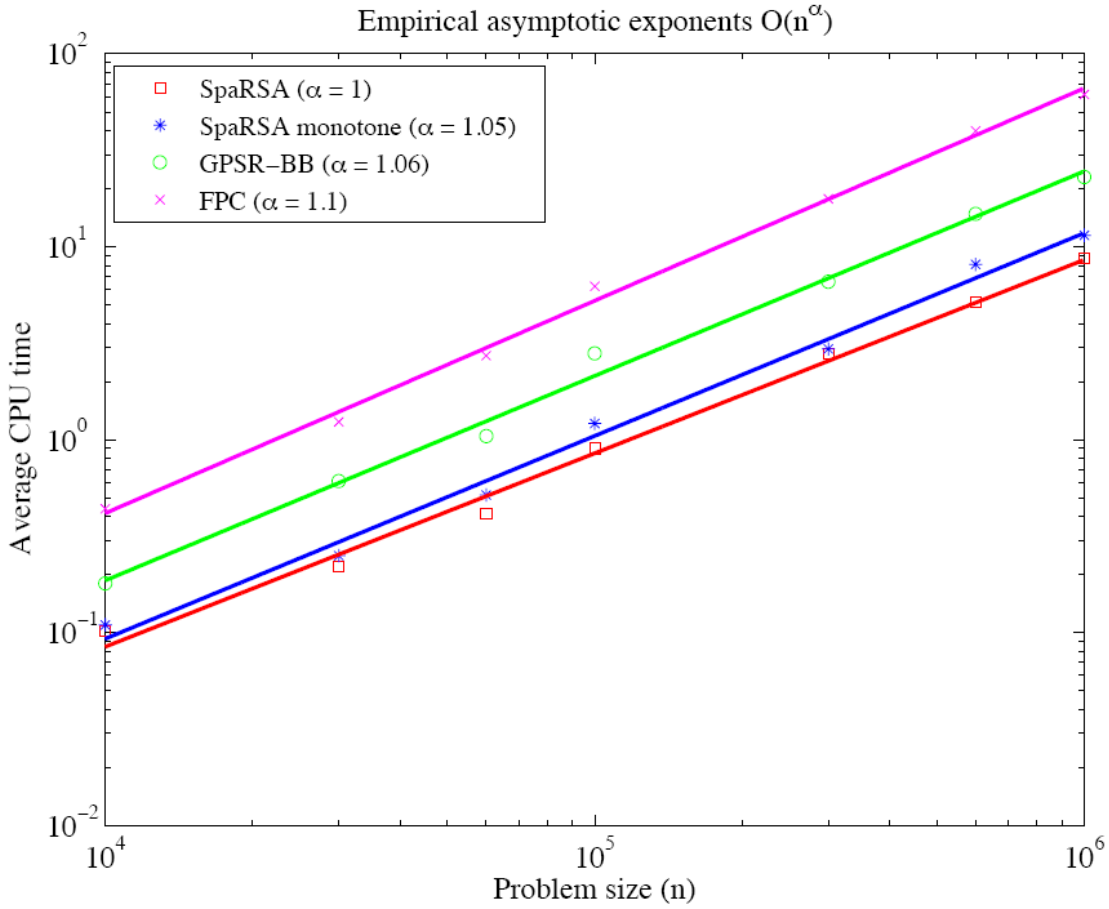
$\mathbf{A} \longleftarrow 0.1\,n \times n$ sparse random (Gaussian) with $3n$ non-zeros

$\mathbf{x} \longleftarrow n/4$ non-zero elements

$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}$, where $\mathbf{e} \sim \mathcal{N}(0, 10^{-4})$

Assumption:

$$CPU = O(n^{\alpha})$$



Group-separable regularization

Group-separable regularizers: $c(\mathbf{x}) = \sum_{j=1}^{m} c_j(\mathbf{x}_{[j]})$

$\{\mathbf{x}_{[1]}, \ldots, \mathbf{x}_{[m]}\}$ are disjoint sub-vectors of $\mathbf{x}$

If c is group-separable, minimization is also group-separable:

$$\mathbf{x}_{[j]}^{k+1} \leftarrow \arg\min_{\mathbf{z}} \frac{1}{2}\|\mathbf{z} - \mathbf{u}_{[j]}^{k}\|_2^2 + \lambda_k\, c_j(\mathbf{z})$$

General solution for a proper, 1-homogenous, convex regularizer (e.g., a norm)

$$\arg\min_{\mathbf{z}} \frac{1}{2}\|\mathbf{z} - \mathbf{u}\|_2^2 + \lambda\, c(\mathbf{z}) = (\mathbf{I} - \mathbf{P}_{\lambda S_c})\,\mathbf{u}$$

$$S_c = \{\mathbf{x} \in \mathbb{R}^n : \; c^*(\mathbf{x}) \leq 1\}$$

Examples:

$c_j(\mathbf{z}) = \|\mathbf{z}\|_2$ ⇒ $\mathbf{x}_{[j]}^{k+1} \leftarrow \mathbf{u}_{[j]}^{k} \dfrac{\max\{\|\mathbf{u}_{[j]}^{k}\|_2 - \lambda_k, 0\}}{\max\{\|\mathbf{u}_{[j]}^{k}\|_2 - \lambda_k, 0\} + \lambda_k}$

(group-ℓ_2)

$c_j(\mathbf{z}) = \|\mathbf{z}\|_\infty$ ⇒ $S_c = \{\mathbf{x} : \|\mathbf{x}\|_1 \leq 1\}$ *O(n log n), O(n)*

(group-ℓ_∞)

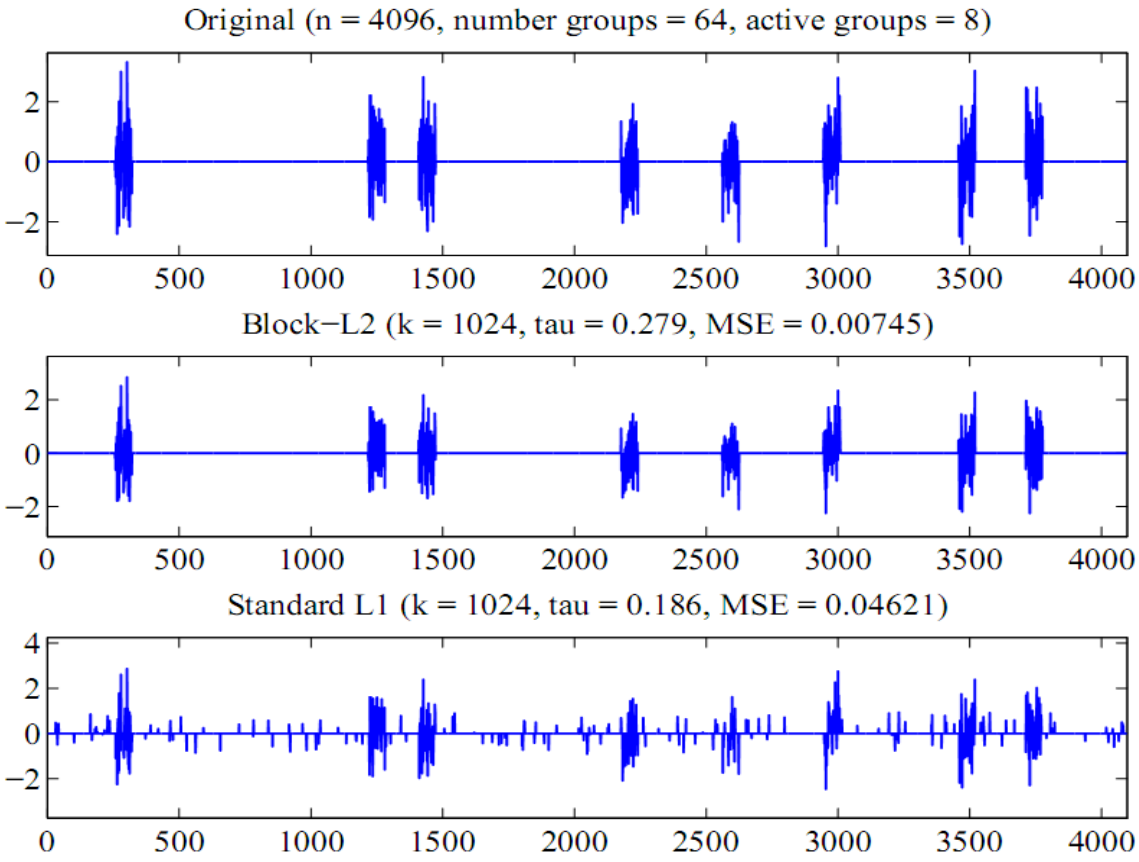
Experiments with Group-ℓ_2

$\mathbf{A}$ ⟵ 10^{10} x 10^{12} random (Gaussian)

$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}$, where $\mathbf{e} \sim \mathcal{N}(0, 10^{-4})$

$\mathbf{x} = \left(\mathbf{x}_{[1]}, \ldots, \mathbf{x}_{[m]}\right)$ ⟵ 64 groups of 64 (consecutive) entries

8 of these filled with Gaussian samples.



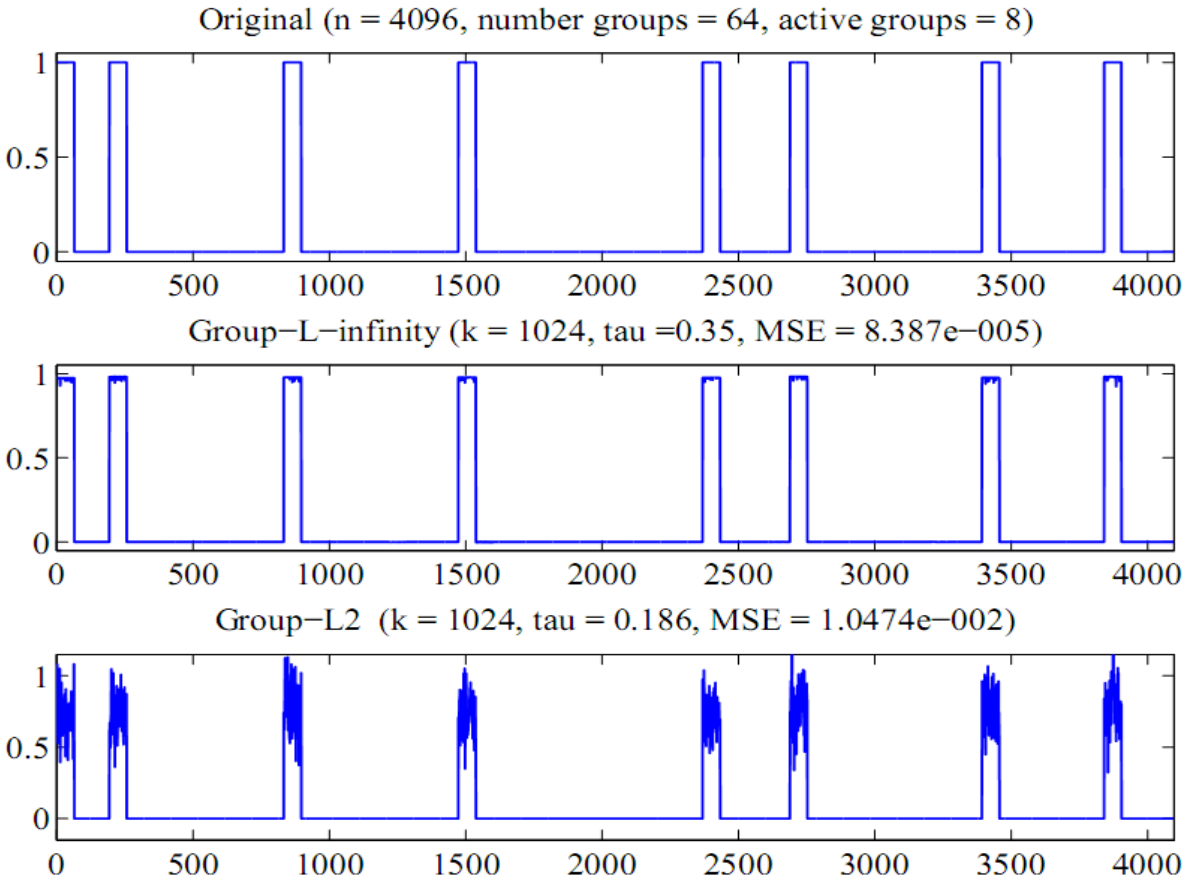
Experiments with Group-ℓ_∞

$\mathbf{A}$ ⟵ 10^{10} x 10^{12} random (Gaussian)

$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}$, where $\mathbf{e} \sim \mathcal{N}(0, 10^{-4})$

$\mathbf{x} = \left(\mathbf{x}_{[1]}, \ldots, \mathbf{x}_{[m]}\right)$ ⟵ 64 groups of 64 (consecutive) entries

8 of these filled with ones.



Convergence of SpaRSA

Problem: $\min_{\mathbf{x}\in\mathbb{R}^n} \phi(\mathbf{x}) := f(\mathbf{x}) + \tau c(\mathbf{x})$

Critical point $\bar{\mathbf{x}}$ if $\mathbf{0} \in \partial\phi(\bar{\mathbf{x}}) = \nabla f(\bar{\mathbf{x}}) + \tau \partial c(\bar{\mathbf{x}})$

Criticality is necessary for optimality.
If both c and ϕ are convex, it is also sufficient.

Safeguarded SpaRSA (S-SParRSA) [Wright, Nowak, F., 2008]

$$Acc(\mathbf{x}^{k+1}) = 1 \;\Leftrightarrow\; \phi(\mathbf{x}^{k+1}) \leq \max_{t=k-M,\ldots,k} \phi(\mathbf{x}^t) - \frac{\sigma\,\alpha_t}{2}\|\mathbf{x}^{t+1} - \mathbf{x}^t\|_2^2$$

where $\sigma \in]0,\, 1[$, usually $\sigma \ll 1$, e.g., $\sigma = 10^{-5}$

Let ϕ be Lipschitz continuously differentiable, c convex and finite-valued, and ϕ bounded below. Then, all accumulation points of S-SpaRSA are critical points of ϕ

Warm Starting and Continuation

SpaRSA (as GPSR, IST, etc) is slow for small τ

SpaRSA (as IST) is "warm-startable", i.e., it benefits from a good initialization

Continuation scheme: start with large τ

slowly decrease ("cool") τ while tracking the solution.

Question: how to set the "cooling schedule" for τ ?

Question: what is large/small τ ?

Focusing on the case $c(\mathbf{x}) = \|\mathbf{x}\|_1$

Easy to show that if $\tau > \|\mathbf{A}^T\mathbf{y}\|_\infty$, then $\widehat{\mathbf{x}} = 0$

Thus, $\|\mathbf{A}^T\mathbf{y}\|_\infty$ sets the scale for large/small τ :

large: $\tau \lesssim \|\mathbf{A}^T\mathbf{y}\|_\infty$

small: $\tau \ll \|\mathbf{A}^T\mathbf{y}\|_\infty$

Adaptive Continuation Scheme

Initialization: choose $\zeta < 1$, $\mathbf{x}^0$, $k \leftarrow 0$, and $\mathbf{y}^t \leftarrow \mathbf{y}$

repeat:

$$\tau_t \leftarrow \max\{\tau, \zeta \|\mathbf{A}^T \mathbf{y}^t\|_\infty\}$$

$$\mathbf{x}^{t+1} \leftarrow \mathrm{SpaRSA}(\mathbf{y}, \mathbf{A}, \tau_t, \mathbf{x}^t)$$

$$\mathbf{y}^{t+1} \leftarrow \mathbf{y} - \mathbf{A}\mathbf{x}^{t+1}$$

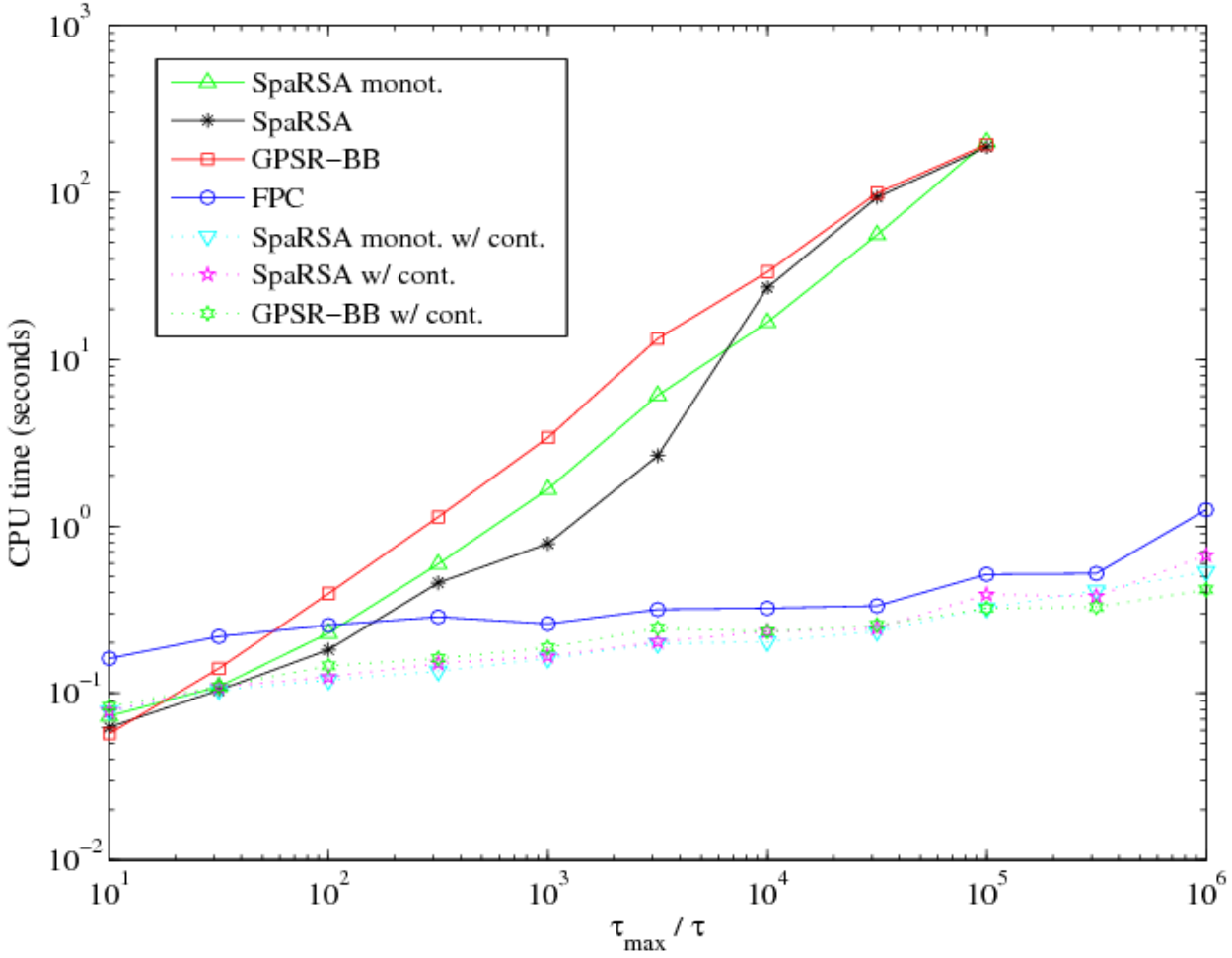
$$t \leftarrow t + 1$$

until $\tau_t = \tau$

Typical values of ζ: 0.2 ~ 0.5

Note: SpaRSA is always applied with $\mathbf{y}$, not $\mathbf{y}^t$: this isn't a pursuit method

Continuation Experiment



$$\tau_{\max} = \|\mathbf{A}^T\mathbf{y}\|_\infty$$

For $\tau \geq \tau_{\max}$, the solution is the zero vector

Another Approach: Variable Splitting

Rewrite the original problem

$$\min_{\mathbf{x}\in\mathbb{R}^n} \phi(\mathbf{x}) := \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2^2 + \tau c(\mathbf{x})$$

as a (equivalent) constrained problem

$$\begin{aligned} \min_{\mathbf{x},\mathbf{z}} \quad & \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2^2 + \tau c(\mathbf{z}) \\ \text{s.t.} \quad & \mathbf{x} = \mathbf{z} \end{aligned}$$

and address it using an augmented Lagragian approach.

Related to recent split-Bregman methods [Goldstein and Osher, 2008], but here used with a different goal.

Detour: Augmented Larangian

Consider a constrained optimization problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{C}\mathbf{x} = \mathbf{b} \end{aligned}$$

The Lagrangian for this problem is $L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T(\mathbf{C}\mathbf{x} - \mathbf{b})$

Lagrange multipliers

The augmented Lagrangian (AL) is

Penalty parameter

$$L_\alpha(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T(\mathbf{C}\mathbf{x} - \mathbf{b}) + \frac{\alpha}{2}\|\mathbf{C}\mathbf{x} - \mathbf{b}\|_2^2$$

Detour: Augmented Larangian (II)

The problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & E(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{C}\mathbf{x} = \mathbf{b} \end{aligned}$$

The augmented Lagrangian (AL)

$$L_\alpha(\mathbf{x}, \boldsymbol{\lambda}) = E(\mathbf{x}) + \boldsymbol{\lambda}^T(\mathbf{C}\mathbf{x} - \mathbf{b}) + \frac{\alpha}{2}\|\mathbf{C}\mathbf{x} - \mathbf{b}\|_2^2$$

The AL algorithm
(method of multipliers)
[Hestenes, 1969]

$$\begin{aligned} \mathbf{x}^{k+1} &= \arg\min_{\mathbf{x}} L_\alpha(\mathbf{x}, \boldsymbol{\lambda}^k) \\ \boldsymbol{\lambda}^{k+1} &= \boldsymbol{\lambda}^k + \alpha(\mathbf{C}\,\mathbf{x}^{k+1} - \mathbf{b}) \end{aligned}$$

Can be written as

$$\begin{aligned} \mathbf{x}^{k+1} &= \arg\min_{\mathbf{x}} E(\mathbf{x}) + \frac{\alpha}{2}\|\mathbf{C}\,\mathbf{x} - \mathbf{d}^k\|_2^2 \\ \mathbf{d}^{k+1} &= \mathbf{d}^k - (\mathbf{C}\,\mathbf{x}^{k+1} - \mathbf{b}) \end{aligned}$$

Detour: AL for Variable Splitting

Consider the problem $\min_{\mathbf{u}} f(\mathbf{u}) + g(\mathbf{G}\,\mathbf{u})$

Equivalent constrained formulation

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{u}) + g(\mathbf{v}) \\ \text{s.t.} \quad & \mathbf{G}\mathbf{u} = \mathbf{v} \end{aligned}$$

Can be written as

$$\begin{aligned} \min_{\mathbf{x}} \quad & E(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{C}\mathbf{x} = \mathbf{b} \end{aligned}$$

$$\mathbf{x} = \begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix} \qquad \mathbf{b} = \mathbf{0}$$

$$\mathbf{C} = [\mathbf{G} \;\; -\mathbf{I}]$$

AL algorithm:

$$(\mathbf{u}^{k+1}, \mathbf{v}^{k+1}) = \arg\min_{\mathbf{u},\mathbf{v}} f(\mathbf{u}) + g(\mathbf{v}) + \frac{\alpha}{2}\|\mathbf{G}\,\mathbf{u} - \mathbf{v} - \mathbf{d}^k\|_2^2$$

$$\mathbf{d}^{k+1} = \mathbf{d}^k - (\mathbf{G}\,\mathbf{u}^{k+1} - \mathbf{v}^{k+1})$$

Detour: AL for Variable Splitting

It can be hard to solve

$$(\mathbf{u}^{k+1}, \mathbf{v}^{k+1}) = \arg\min_{\mathbf{u},\mathbf{v}} f(\mathbf{u}) + g(\mathbf{v}) + \frac{\alpha}{2}\|\mathbf{G}\,\mathbf{u} - \mathbf{v} - \mathbf{d}^k\|_2^2$$

Alternative: take a single block Gauss-Seidel step:

$$\begin{aligned}
\mathbf{u}^{k+1} &= \arg\min_{\mathbf{u}} f(\mathbf{u}) + \frac{\alpha}{2}\|\mathbf{G}\,\mathbf{u} - \mathbf{v}^k - \mathbf{d}^k\|_2^2 \\
\mathbf{v}^{k+1} &= \arg\min_{\mathbf{v}} g(\mathbf{v}) + \frac{\alpha}{2}\|\mathbf{G}\,\mathbf{u}^{k+1} - \mathbf{v} - \mathbf{d}^k\|_2^2 \\
\mathbf{d}^{k+1} &= \mathbf{d}^k - (\mathbf{G}\,\mathbf{u}^{k+1} - \mathbf{v}^{k+1})
\end{aligned}$$

Alternating directions method of multipliers (ADMM)

[Glowinsky, Marrocco, 1975], [Gabay, Mercier, 1976], [Bertsekas, Tsitsiklis, 1989].

Convergence shown in [Eckstein and Bertsekas, 1992]

(Rachford-Douglas splitting in the dual problem)

Application of ADMM

Back to our problem: $\min_{\mathbf{x},\mathbf{z}} \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{y}\|_2^2 + \tau c(\mathbf{z}), \quad \text{s.t.} \;\; \mathbf{x}=\mathbf{z}$

The resulting ADMM algorithm:

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{A}\,\mathbf{x}-\mathbf{y}\|_2^2 + \frac{\alpha}{2}\|\mathbf{x}-\mathbf{z}^k-\mathbf{d}^k\|_2^2$$

$$\mathbf{z}^{k+1} = \arg\min_{\mathbf{z}} \tau\, c(\mathbf{z}) + \frac{\alpha}{2}\|\mathbf{x}^{k+1}-\mathbf{z}-\mathbf{d}^k\|_2^2$$

$$\mathbf{d}^{k+1} = \mathbf{d}^k - \mathbf{x}^{k+1} + \mathbf{z}^{k+1}$$

The first optimization is quadratic, thus the solution is a linear system.

The second one corresponds to a shrinkage (denoising) operation.

$$\mathbf{x}^{k+1} = \left(\mathbf{A}^T\mathbf{A} + \alpha\mathbf{I}\right)^{-1}\left(\mathbf{A}^T\mathbf{y} + \alpha(\mathbf{z}^k + \mathbf{d}^k)\right)$$

$$\mathbf{z}^{k+1} = \mathbf{\Psi}_{\tau/\alpha}(\mathbf{x}^{k+1} - \mathbf{d}^k)$$

$$\mathbf{d}^{k+1} = \mathbf{d}^k - \mathbf{x}^{k+1} + \mathbf{z}^{k+1}$$

Implementation: The SALSA Algorithm

The "split augmented Lagrangian shrinkage algorithm" (SALSA)

[F., Bioucas-Dias, Afonso, 2009]

$$\mathbf{x}^{k+1} = \left(\mathbf{A}^T\mathbf{A} + \alpha\mathbf{I}\right)^{-1}\left(\mathbf{A}^T\mathbf{y} + \alpha(\mathbf{z}^k + \mathbf{d}^k)\right)$$

$$\mathbf{z}^{k+1} = \mathbf{\Psi}_{\tau/\alpha}(\mathbf{x}^{k+1} - \mathbf{d}^k)$$

$$\mathbf{d}^{k+1} = \mathbf{d}^k - \mathbf{x}^{k+1} + \mathbf{z}^{k+1}$$

SALSA is only interesting if $\mathbf{A}^T\mathbf{A} + \alpha\mathbf{I}$ can be inverted inexpensively!

Examples: TV-based deconvolution, frame-based deconvolution,
restoration of missing samples, MRI reconstruction.

SALSA Application: Image Deconvolution

In wavelet (frame) based deconvolution $\mathbf{A} = \mathbf{BW}$

convolution (e.g. blur)

basis/Parseval frame $\mathbf{W}\mathbf{W}^T = \mathbf{I}$

$$\left(\mathbf{A}^T\mathbf{A} + \alpha\mathbf{I}\right)^{-1} = \frac{1}{\alpha}\left(\mathbf{I} - \mathbf{A}^T(\mathbf{A}\mathbf{A}^T + \alpha\mathbf{I})^{-1}\mathbf{A}\right)$$

$$= \frac{1}{\alpha}\left(\mathbf{I} - \mathbf{W}^T\mathbf{B}^T(\mathbf{B}\mathbf{B}^T + \alpha\mathbf{I})^{-1}\mathbf{B}\mathbf{W}\right)$$

Can be computed in the Fourier domain using the FFT.

In TV-based deconvolution, $\left(\mathbf{A}^T\mathbf{A} + \alpha\mathbf{I}\right)^{-1}$ can itself be computed via the FFT

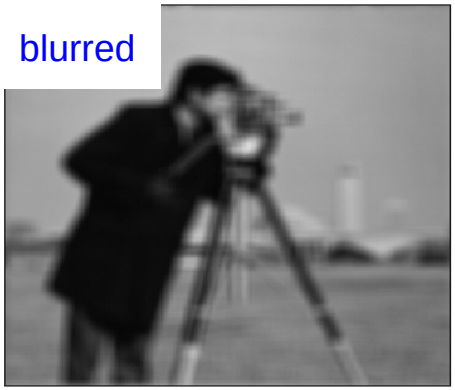
In any if these cases, the cost per iteration is $O(n \log n)$

SALSA Application: Image Deconvolution

9x9 uniform blur, 40dB BSNR

undecimated Haar wavelets,

ℓ_1 regularization.



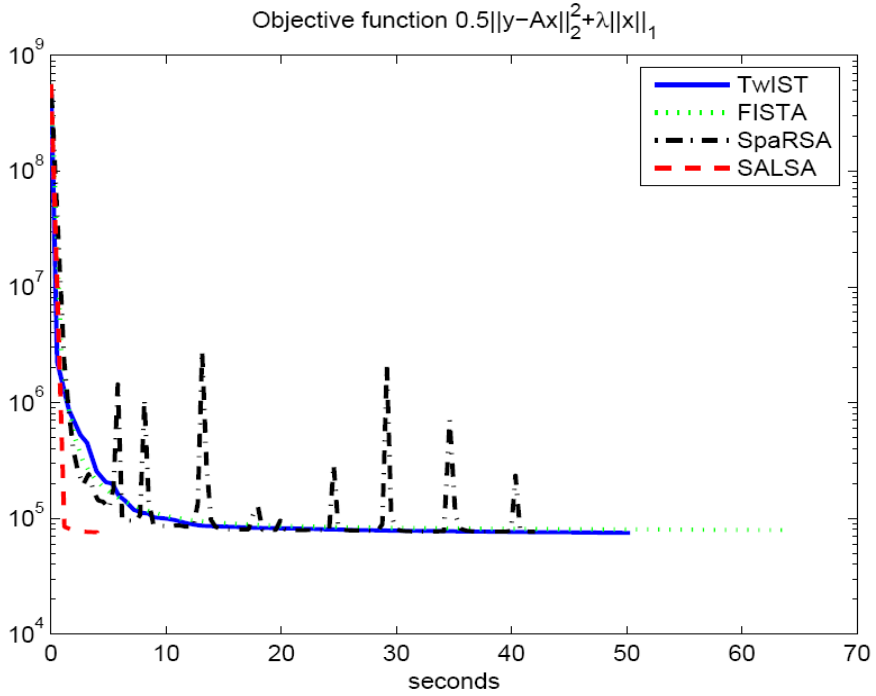


Table 2. CPU times (in seconds) for the various algorithms.

Experiment	TwIST	SpARSA	FISTA	SALSA
1	50.2969	42.0469	64.2344	4.000
2A	30.7656	40.6094	61.7031	4.03125
2B	14.4063	6.92188	15.0781	1.9375
3A	23.5313	17.0156	33.7969	2.60938
3B	8.1875	6.17188	18.0781	1.89063

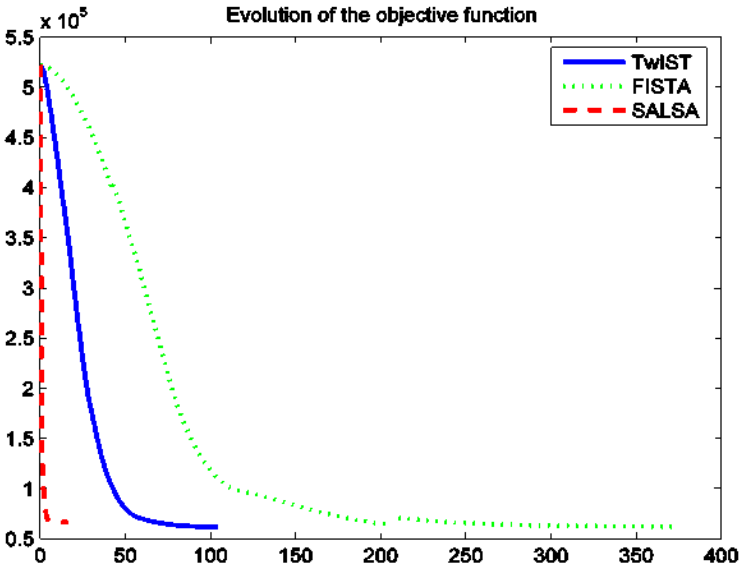
SALSA Application: Missing Samples Restoration (TV)

Even simpler:

$\mathbf{A}$ is an $m \times n$ subsampling matrix $(m < n)$

$$\mathbf{A}\mathbf{A}^T = \mathbf{I}_m$$

$$\left(\mathbf{A}^T\mathbf{A} + \alpha\mathbf{I}\right)^{-1} = \frac{1}{\alpha}\left(\mathbf{I} - \frac{\mathbf{A}^T\mathbf{A}}{\alpha+1}\right)$$



Deconvolution of Poissonian Images

Using the Poisson log-likelihood:

$$\min_{\mathbf{x}} \sum_{i=1}^{n} (\mathbf{A}\,\mathbf{x})_i - y_i \log(\mathbf{A}\,\mathbf{x})_i + \tau c(\mathbf{x})$$

$$\text{s.t. } \mathbf{A}\,\mathbf{x} \geq 0$$

e.g., total-variation

IST/SpaRSA convergence not guaranteed because derivative of log is not Lipschitz

Variable splitting

$$\min_{\mathbf{x}} \sum_{i=1}^{n} z_i - y_i \log z_i + \tau c(\mathbf{u})$$

$$\text{s.t.} \quad \mathbf{A}\,\mathbf{x} = \mathbf{z}$$

$$\mathbf{x} = \mathbf{u}$$

$$\mathbf{z} \geq 0$$

← can be ignored; justified later.

Deconvolution of Poissonian Images

Constrained problem

$$\min_{\mathbf{x}} \sum_{i=1}^{n} z_i - y_i \log z_i + \tau c(\mathbf{u}) \qquad \text{s.t.} \quad \mathbf{A}\,\mathbf{x} = \mathbf{z}, \quad \mathbf{x} = \mathbf{u}$$

Can be written as

$$\begin{aligned} \min_{\mathbf{x}} \quad & E(\mathbf{v}) \\ \text{s.t.} \quad & \mathbf{C}\mathbf{v} = \mathbf{b} \end{aligned} \qquad \mathbf{v} = \begin{bmatrix} \mathbf{x} \\ \mathbf{z} \\ \mathbf{u} \end{bmatrix} \qquad \mathbf{C} = \begin{bmatrix} \mathbf{A} & -\mathbf{I} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} & -\mathbf{I} \end{bmatrix}, \quad \mathbf{b} = \mathbf{0}$$

Thus the AL can be formed and the alternating directions method of multipliers (ADMM) can be used.

Deconvolution of Poissonian Images

PIDAL (Poisson image deconvolution by AL) algorithm [F. and Bioucas-Dias, 2009]

$$\mathbf{x}' = \mathbf{z}_k + \mathbf{d}_k^{(1)}$$

$$\mathbf{x}'' = \mathbf{u}_k + \mathbf{d}_k^{(2)}$$

$$\mathbf{x}_{k+1} := \arg\min_{\mathbf{x}} \|\mathbf{K}\,\mathbf{x} - \mathbf{x}'\|_2^2 + \|\mathbf{x} - \mathbf{x}''\|_2^2$$

← Quadratic problem, linear solution (as in SALSA)

$$\mathbf{z}' = \mathbf{K}\mathbf{x}_{k+1} - \mathbf{d}_k^{(1)}$$

$$\mathbf{z}_{k+1} := \arg\min_{\mathbf{z}} \sum_{i=1}^{n} z_i - y_i \log z_i + \frac{\mu}{2}\|\mathbf{z} - \mathbf{z}'\|_2^2$$

← Separable

$$\mathbf{u}' = \mathbf{x}_{k+1} - \mathbf{d}_k^{(2)}$$

$$\mathbf{u}_{k+1} := \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{u} - \mathbf{u}'\|^2 + (\tau/\mu)\,\phi(\mathbf{u}).$$

← Denoising/shringkage (as in SALSA)

$$\mathbf{d}_{k+1}^{(1)} := \mathbf{d}_k^{(1)} - (\mathbf{K}\,\mathbf{x}_{k+1} - \mathbf{z}_{k+1})$$

$$\mathbf{d}_{k+1}^{(2)} := \mathbf{d}_k^{(2)} - (\mathbf{x}_{k+1} - \mathbf{u}_{k+1})$$

Deconvolution of Poissonian Images

Solving $\mathbf{z}_{k+1} := \arg\min_{\mathbf{z}} \sum_{i=1}^{n} z_i - y_i \log z_i + \frac{\mu}{2}\|\mathbf{z} - \mathbf{z}'\|_2^2$

Separable, strictly convex, with closed form solution:

$$z_{i,k+1} = \frac{\mu z_i' - 1 + \sqrt{(\mu z_i' - 1)^2 + 4\mu y_i}}{2\mu}$$

automatically non-negative: we don't need to use the constraint.

Comparison with recent state-of-the-art method [Dupé, Fadili, Stark, 2009]

Cameraman image, 7x7 uniform blur, SNR controlled by scaling image.

Resulting mean absolute errors

max intensity	5	30	100	255
PIDAL	0.37	1.34	3.99	8.65
Algorithm from [9]	0.44	1.44	4.69	10.40

PIDAL is

10~100 times faster.

Summary

- Reviewed several ways to derive the IST class of algorithms

- Reviewed several convergence results for IST algorithms

- Described recent accelerated versions: TwIST, SpaRSA

- Presented the AL-based approach to derive IST-like algorithms

- Presented new restoration algorithm (SALSA)

- Presented new deconvolution algorithm for Poisson data (PIDAL)