A Fast Iterative Shrinkage Algorithm for Convex Regularized Linear Inverse Problems

Marc Teboulle

School of Mathematical Sciences

Tel-Aviv University, Ramat-Aviv, Israel

Joint Work with Amir Beck, Technion, Haifa

International Conference on Nonlinear Programming and Applications

NPA 2008 – April 6–9, 2008, Beijing, China

Outline

- Linear Inverse Problems with Nonsmooth Regularization
  - Formulation and Application Areas
- Current Class of Iterative Methods (ISTA):
  - Iterative Shrinkage-Threshold Algorithms
- FISTA: A Fast Iterative Shrinkage-Threshold Algorithm
  - A global rate of convergence/complexity estimate
- Numerical Examples for Image Deblurring Problems
- Conclusions

Linear Inverse Problem

Problem: Estimate the unknown signal $\mathbf{x}$ from a noisy observation

$$\mathbf{A}\mathbf{x} = \mathbf{b} + \mathbf{w}.$$

- $\mathbf{x} \in \mathbb{R}^n$ - input signal– (Unknown True Image)
- $\mathbf{b} \in \mathbb{R}^m$ - observable output – (Blurred Image)
- $\mathbf{w} \in \mathbb{R}^m$ - unknown noise vector.
- $\mathbf{A} \in \mathbb{R}^{m \times n}$ model – (Blurring matrix (2-dim convolution)).

An Example: The problem of estimating $\mathbf{x}$ from the observed blurred and noisy image is an *Image Deblurring Problem*.

Regularization Approaches

Classical Least Squares (LS) estimator

$$\text{(LS)}: \quad \hat{\mathbf{x}}_{\text{LS}} = \operatorname*{argmin}_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|^2.$$

$\mathbf{A}$ ill-conditioned – meaningless solution

Regularization Approaches

Classical Least Squares (LS) estimator

$$\text{(LS)}: \quad \hat{\mathbf{x}}_{\mathrm{LS}} = \operatorname*{argmin}_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2.$$

$\mathbf{A}$ ill-conditioned – meaningless solution

Tikhonov regularization – quadratic penalty

$$\text{(T)}: \quad \hat{\mathbf{x}}_{\mathrm{TIK}} = \operatorname*{argmin}_{\mathbf{x}} \{\|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 + \lambda\|\mathbf{L}\mathbf{x}\|^2\}, \ \lambda > 0.$$

Regularization Approaches

Classical Least Squares (LS) estimator

$$\text{(LS)}: \quad \hat{\mathbf{x}}_{\text{LS}} = \operatorname*{argmin}_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2.$$

A ill-conditioned – meaningless solution

Tikhonov regularization – quadratic penalty

$$\text{(T)}: \quad \hat{\mathbf{x}}_{\text{TIK}} = \operatorname*{argmin}_{\mathbf{x}} \{\|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 + \lambda\|\mathbf{L}\mathbf{x}\|^2\}, \ \lambda > 0.$$

l_1-norm regularization

$$(L_1) \qquad \min_{\mathbf{x}} \{F(\mathbf{x}) \equiv \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 + \lambda\|\mathbf{x}\|_1\}$$

Less sensitive to outliers (as opposed to l_2 regularization). Has attracted a revived interest and considerable amount of attention in Signal Processing Research.

The l_1-Regularization Model: Old and New Applications

- LASSO in Statistics (Tibshirani (96))
- Basis pursuit denoising (Chen et al. (98))

The l_1-Regularization Model: Old and New Applications

- LASSO in Statistics (Tibshirani (96))
- Basis pursuit denoising (Chen et al. (98))
- Wavelet based image/signal restoration (Donoho (95), Chambolle (04)...)
- Sparse Approximation of signals (Elad (06), Daubechies et al. (07),...)
- Compressed sensing: few measurements are enough to produce good reconstruction (Candes-Tao (06), Donoho(06)...)

♠ The term $\|\mathbf{x}\|_1$ promotes sparsity in the optimal solution.

The l_1-Regularization Model: Old and New Applications

- LASSO in Statistics (Tibshirani (96))
- Basis pursuit denoising (Chen et al. (98))
- Wavelet based image/signal restoration (Donoho (95), Chambolle (04)...)
- Sparse Approximation of signals (Elad (06), Daubechies et al. (07),...)
- Compressed sensing: few measurements are enough to produce good reconstruction (Candes-Tao (06), Donoho(06)...)

♠ The term $\|\mathbf{x}\|_1$ promotes sparsity in the optimal solution.

♢ In image deblurring/wavelet based restoration: most images have a *sparse representation in wavelet domain*.
♠ State of the art regularization for Image Restoration involves **nonsmooth** regularizers.

General Formulation with Nonsmooth Regularizers

A nonsmooth convex minimization model which covers quite a lot of interesting and disparate applications.

$$(P) \qquad \min\{F(\mathbf{x}) \equiv f(\mathbf{x}) + g(\mathbf{x}) : \mathbf{x} \in \mathbb{R}^n\}.$$

- $f : \mathbb{R}^n \to \mathbb{R}$ is a smooth convex function of the type $C^{1,1}$, i.e., continuously differentiable with Lipschitz continuous gradient $L(f)$:

  $$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L(f)\|\mathbf{x} - \mathbf{y}\| \quad \text{for every } \mathbf{x}, \mathbf{y} \in \mathbb{R}^n,$$

  where $\|\cdot\|$ denotes the standard Euclidean norm and $L(f) > 0$ is the Lipschitz constant of ∇f.
- $g : \mathbb{R}^n \to \mathbb{R}$ is a convex function which is ***nonsmooth***.
- Problem (P) is solvable, i.e., $X_* := \operatorname{argmin} f \neq \emptyset$, and for $\mathbf{x}^* \in X_*$ we set $F_* := F(\mathbf{x}^*)$.

♣ Challenges: How do we solve (P)?

- The problem in **nonsmooth**.

♣ Challenges: How do we solve (P)?

- The problem in **nonsmooth**.
- In most applications, can be **very large scale**, e.g., in image deblurring, the dimension varies from $d = 65,536$ to $1,048,576$.

♣ Challenges: How do we solve (P)?

- The problem in **nonsmooth**.
- In most applications, can be **very large scale**, e.g., in image deblurring, the dimension varies from $d = 65,536$ to $1,048,576$.
- Involves **dense matrix data**, precluding the use and potential advantages of well-known methods (storage/factorization impractical), even for the L_1 problem (which can be reformulated as a QP or SOCP).

♣ Challenges: How do we solve (P)?

- The problem in **nonsmooth**.
- In most applications, can be **very large scale**, e.g., in image deblurring, the dimension varies from $d = 65,536$ to $1,048,576$.
- Involves **dense matrix data**, precluding the use and potential advantages of well-known methods (storage/factorization impractical), even for the L_1 problem (which can be reformulated as a QP or SOCP).
- This motivates the search for simple and efficient algorithms where the dominant computational effort is a relatively cheap **matrix-vector** multiplications involving $\mathbf{A}$ and $\mathbf{A}^T$.

♣ Challenges: How do we solve (P)?

- The problem in **nonsmooth**.
- In most applications, can be **very large scale**, e.g., in image deblurring, the dimension varies from $d = 65,536$ to $1,048,576$.
- Involves **dense matrix data**, precluding the use and potential advantages of well-known methods (storage/factorization impractical), even for the L_1 problem (which can be reformulated as a QP or SOCP).
- This motivates the search for simple and efficient algorithms where the dominant computational effort is a relatively cheap **matrix-vector** multiplications involving $\mathbf{A}$ and $\mathbf{A}^T$.
- Simple algorithms exist...But...

A Current Very Popular Algorithm

Class of *Iterative Shrinkage-Threshold Algorithms* (ISTA) for L_1:

$$\mathbf{x}_{k+1} = \mathcal{T}_{\lambda t}\left(\mathbf{x}_k - t\mathbf{A}^T(\mathbf{A}\mathbf{x}_k - \mathbf{b})\right), \ t > 0 \text{ a step size}$$

and $\mathcal{T}_\alpha : \mathbb{R}^n \to \mathbb{R}^n$ is the shrinkage operator defined by

$$\mathcal{T}_\alpha(\mathbf{x})_i = (|x_i| - \alpha)_+ \operatorname{sgn}(x_i).$$

Each iteration involves matrix-vector multiplication involving $\mathbf{A}$ and $\mathbf{A}^T$ followed by a shrinkage/soft-threshold step.

A Current Very Popular Algorithm

Class of *Iterative Shrinkage-Threshold Algorithms* (ISTA) for L_1:

$$\mathbf{x}_{k+1} = \mathcal{T}_{\lambda t}\left(\mathbf{x}_k - t\mathbf{A}^T(\mathbf{A}\mathbf{x}_k - \mathbf{b})\right),\ t > 0 \text{ a step size}$$

and $\mathcal{T}_\alpha : \mathbb{R}^n \to \mathbb{R}^n$ is the shrinkage operator defined by

$$\mathcal{T}_\alpha(\mathbf{x})_i = (|x_i| - \alpha)_+ \mathrm{sgn}\,(x_i).$$

Each iteration involves matrix-vector multiplication involving $\mathbf{A}$ and $\mathbf{A}^T$ followed by a shrinkage/soft-threshold step.

In SP literature: appeared under various names: Iterative denoising, Shrinkage-Thresholded, Landweber, EM wavelet based etc....: Chambolle (98); Figueiredo-Nowak (03, 05); Daubechies et al. (04),...

In Optimization: it is a well known algorithm....

A Basic Approximation Model:Following the well-known gradient scheme

- For any $L > 0$, and a given $\mathbf{z}$:

$$Q_L(\mathbf{x},\mathbf{z}) := f(\mathbf{z}) + \langle \mathbf{x}-\mathbf{z}, \nabla f(\mathbf{z})\rangle + \frac{L}{2}\|\mathbf{x}-\mathbf{z}\|^2 + \mathbf{g}(\mathbf{x}) \swarrow \text{ left untouched}$$

$\min_{\mathbf{x}} F(\mathbf{x}) \hookrightarrow \ \min_{\mathbf{x}} Q_L(\mathbf{x},\mathbf{z})$ which admits a unique minimizer

$$p_L(\mathbf{z}) := \operatorname*{argmin}_{\mathbf{x}} Q_L(\mathbf{x},\mathbf{z}) = \operatorname*{argmin}_{\mathbf{x}}\{g(\mathbf{x}) + \frac{L}{2}\|\mathbf{x} - (\mathbf{z} - \frac{1}{L}\nabla f(\mathbf{z}))\|^2\}.$$

A Basic Approximation Model:Following the well-known gradient scheme

- For any $L > 0$, and a given $\mathbf{z}$:

$$Q_L(\mathbf{x},\mathbf{z}) := f(\mathbf{z})+\langle \mathbf{x}-\mathbf{z}, \nabla f(\mathbf{z})\rangle+\frac{L}{2}\|\mathbf{x}-\mathbf{z}\|^2+\mathbf{g(x)} \swarrow \text{ left untouched}$$

$\min_{\mathbf{x}} F(\mathbf{x}) \hookrightarrow \ \min_{\mathbf{x}} Q_L(\mathbf{x},\mathbf{z})$ which admits a unique minimizer

$$p_L(\mathbf{z}) := \underset{\mathbf{x}}{\operatorname{argmin}}\, Q_L(\mathbf{x},\mathbf{z}) = \underset{\mathbf{x}}{\operatorname{argmin}}\{g(\mathbf{x})+\frac{L}{2}\|\mathbf{x}-(\mathbf{z}-\frac{1}{L}\nabla f(\mathbf{z}))\|^2\}.$$

- **Algorithm:** $\mathbf{x}_0 \in \mathbb{R}^n, \ \mathbf{x}_{k+1} = p_L(\mathbf{x}_k)$.

A Basic Approximation Model:Following the well-known gradient scheme

- For any $L > 0$, and a given $\mathbf{z}$:

$$Q_L(\mathbf{x},\mathbf{z}) := f(\mathbf{z}) + \langle \mathbf{x}-\mathbf{z}, \nabla f(\mathbf{z})\rangle + \frac{L}{2}\|\mathbf{x}-\mathbf{z}\|^2 + g(\mathbf{x}) \swarrow \text{ left untouched}$$

$$\min_{\mathbf{x}} F(\mathbf{x}) \hookrightarrow \ \min_{\mathbf{x}} Q_L(\mathbf{x},\mathbf{z}) \text{ which admits a unique minimizer}$$

$$p_L(\mathbf{z}) := \operatorname*{argmin}_{\mathbf{x}} Q_L(\mathbf{x},\mathbf{z}) = \operatorname*{argmin}_{\mathbf{x}} \{ g(\mathbf{x}) + \frac{L}{2}\|\mathbf{x} - (\mathbf{z} - \frac{1}{L}\nabla f(\mathbf{z}))\|^2 \}.$$

- **Algorithm:** $\mathbf{x}_0 \in \mathbb{R}^n, \ \mathbf{x}_{k+1} = p_L(\mathbf{x}_k)$.
- **Special Case-ISTA** $g(\mathbf{x}) := \lambda\|\mathbf{x}\|_1, \ f(\mathbf{x}) := \|\mathbf{A}\mathbf{x} - b\|^2, \ L := t^{-1}$

A Basic Approximation Model:Following the well-known gradient scheme

- For any $L > 0$, and a given $\mathbf{z}$:

$$Q_L(\mathbf{x},\mathbf{z}) := f(\mathbf{z})+\langle \mathbf{x}-\mathbf{z}, \nabla f(\mathbf{z})\rangle + \frac{L}{2}\|\mathbf{x}-\mathbf{z}\|^2 + g(\mathbf{x}) \swarrow \text{ left untouched}$$

$\min_{\mathbf{x}} F(\mathbf{x}) \hookrightarrow \ \min_{\mathbf{x}} Q_L(\mathbf{x},\mathbf{z})$ which admits a unique minimizer

$$p_L(\mathbf{z}) := \operatorname*{argmin}_{\mathbf{x}} Q_L(\mathbf{x},\mathbf{z}) = \operatorname*{argmin}_{\mathbf{x}} \{g(\mathbf{x}) + \frac{L}{2}\|\mathbf{x} - (\mathbf{z} - \frac{1}{L}\nabla f(\mathbf{z}))\|^2\}.$$

- **Algorithm:** $\mathbf{x}_0 \in \mathbb{R}^n, \ \mathbf{x}_{k+1} = p_L(\mathbf{x}_k).$
- **Special Case-ISTA**$g(\mathbf{x}) := \lambda\|\mathbf{x}\|_1, \ f(\mathbf{x}) := \|\mathbf{A}\mathbf{x} - b\|^2, \ L := t^{-1}$
- Can be viewed as the Proximal-FB Splitting Method (Passty (79)):

$$0 \in \nabla f(\mathbf{x}) + \partial g(\mathbf{x}) \iff \mathbf{x} = (I + s\partial g)^{-1}(I - s\nabla f)(\mathbf{x}), \ (s > 0)$$

Advantage and Drawback of ISTA

- **Advantage:** Simplicity. Useful when $p_L(\cdot)$ can be computed analytically, e.g. when $g(\cdot)$ is separable, reduces to a one dimensional minimization problem, ($g(\mathbf{x}) := \|\mathbf{x}\|_p,\ p \geq 1$).
- **Drawback:** ISTA appears to be a (very) slow method.

Advantage and Drawback of ISTA

- **Advantage:** Simplicity. Useful when $p_L(\cdot)$ can be computed analytically, e.g. when $g(\cdot)$ is separable, reduces to a one dimensional minimization problem, ($g(\mathbf{x}) := \|\mathbf{x}\|_p, \; p \geq 1$).
- **Drawback:** ISTA appears to be a (very) slow method.

$\diamondsuit$ Convergence analysis of methods like ISTA has been well studied in past/ recent literature under various contexts and frameworks, (Facchinei-Pang, Vol II, Chap. 12, 2003).
$\diamondsuit$ The focus is on pointwise convergence of $\{x_k\}$ and *asymptotic* rate of convergence.

Advantage and Drawback of ISTA

- **Advantage:** Simplicity. Useful when $p_L(\cdot)$ can be computed analytically, e.g. when $g(\cdot)$ is separable, reduces to a one dimensional minimization problem, ($g(\mathbf{x}) := \|\mathbf{x}\|_p, \ p \geq 1$).
- **Drawback:** ISTA appears to be a (very) slow method.

$\diamondsuit$ Convergence analysis of methods like ISTA has been well studied in past/ recent literature under various contexts and frameworks, (Facchinei-Pang, Vol II, Chap. 12, 2003).
$\diamondsuit$ The focus is on pointwise convergence of $\{x_k\}$ and *asymptotic* rate of convergence.
Here, we focus on the ***nonasymptotic*** global rate of convergence and efficiency measured through functions values.

A by-product of our analysis theoretically confirms the slow convergence rate:

$$F(\mathbf{x}_k) - F(\mathbf{x}^*) \simeq O(1/k),$$

namely ISTA, shares a **sublinear** global rate of convergence.

Can We Do Better to Solve the NSO $\min_{\mathbf{x}}\{f(\mathbf{x})+g(\mathbf{x})\}$?

- Can we devise a faster method than ISTA such that:
  ♠ The computational effort of the new method will keep the simplicity of ISTA
  ♠ Its global rate of convergence will be significantly better, **theoretically and practically.**

Can We Do Better to Solve the NSO $\min_{\mathbf{x}}\{f(\mathbf{x})+g(\mathbf{x})\}$?

- Can we devise a faster method than ISTA such that:
  ♠ The computational effort of the new method will keep the simplicity of ISTA
  ♠ Its global rate of convergence will be significantly better, **theoretically and practically.**
- **Answer: Yes**, through an equally simple scheme

$$\clubsuit\ \mathbf{x}_{k+1} = \operatorname*{argmin}_{\mathbf{x}} Q_L(\mathbf{x}, \mathbf{y}_k), \ \hookleftarrow \ \mathbf{y}_k \text{ instead of } \mathbf{x}_k$$

The new point $\mathbf{y}_k$ will be smartly chosen and **easy** to compute.

Can We Do Better to Solve the NSO $\min_{\mathbf{x}}\{f(\mathbf{x})+g(\mathbf{x})\}$?

- Can we devise a faster method than ISTA such that:
  ♠ The computational effort of the new method will keep the simplicity of ISTA
  ♠ Its global rate of convergence will be significantly better, **theoretically and practically.**
- **Answer: Yes**, through an equally simple scheme

$$\clubsuit\ \mathbf{x}_{k+1} = \operatorname*{argmin}_{\mathbf{x}} Q_L(\mathbf{x}, \mathbf{y}_k), \quad \hookleftarrow \quad \mathbf{y}_k \text{ instead of } \mathbf{x}_k$$

  The new point $\mathbf{y}_k$ will be smartly chosen and **easy** to compute.
- **Idea:** From an algorithm (Nesterov 1983), designed for minimizing a **smooth** convex function, and proven to be an *"optimal"* first order method (Yudin-Nemirovsky (80).)

Can We Do Better to Solve the NSO $\min_{\mathbf{x}}\{f(\mathbf{x})+g(\mathbf{x})\}$?

- Can we devise a faster method than ISTA such that:
  ♠ The computational effort of the new method will keep the simplicity of ISTA
  ♠ Its global rate of convergence will be significantly better, **theoretically and practically.**
- **Answer: Yes**, through an equally simple scheme

$$\clubsuit\ \mathbf{x}_{k+1} = \operatorname*{argmin}_{\mathbf{x}} Q_L(\mathbf{x}, \mathbf{y}_k), \ \hookleftarrow\ \mathbf{y}_k \text{ instead of } \mathbf{x}_k$$

  The new point $\mathbf{y}_k$ will be smartly chosen and **easy** to compute.
- **Idea:** From an algorithm (Nesterov 1983), designed for minimizing a **smooth** convex function, and proven to be an *"optimal"* first order method (Yudin-Nemirovsky (80).)
- But, here our problem (P) is **nonsmooth !..** Yet, we derive a faster algorithm than ISTA for the general NSO problem (P), proven optimal. We call it **FISTA...**

FISTA: A Fast Iterative Shrinkage/Threshold Algorithm

An equally simple algorithm as ISTA. Here $L(f)$ is known.

FISTA with constant stepsize
Input: $L = L(f)$ - A Lipschitz constant of ∇f.
Step 0. Take $\mathbf{y}_1 = \mathbf{x}_0 \in \mathbb{R}^n,\ t_1 = 1$.
Step k. ($k \geq 1$) Compute

$$
\begin{aligned}
\mathbf{x}_k &= p_L(\mathbf{y}_k), \quad \hookleftarrow \text{ main computation as ISTA} \\
\bullet \quad t_{k+1} &= \frac{1+\sqrt{1+4t_k^2}}{2}, \\
\bullet\bullet \quad \mathbf{y}_{k+1} &= \mathbf{x}_k + \left(\frac{t_k - 1}{t_{k+1}}\right)(\mathbf{x}_k - \mathbf{x}_{k-1}).
\end{aligned}
$$

The requested additional computation for FISTA in ($\bullet$) and ($\bullet\bullet$) is clearly marginal.

Knowledge of $L(f)$ is not Necessary: FISTA With Backtracking

FISTA with backtracking

Step 0. Take $L_0 > 0$, some $\eta > 1$ and $\mathbf{x}_0 \in \mathbb{R}^n$. Set $\mathbf{y}_1 = \mathbf{x}_0,\ t_1 = 1$.

Step k. ($k \geq 1$) Find the smallest nonnegative integers i_k such that with $i = i_k$, $\bar{L} = \eta^{i_k} L_{k-1}$:

$$F(p_{\bar{L}}(\mathbf{y}_k)) \leq Q_{\bar{L}}(p_{\bar{L}}(\mathbf{y}_k), \mathbf{y}_k).$$

Set $L_k = \eta^{i_k} L_{k-1}$ and compute

$$\begin{aligned} \mathbf{x}_k &= p_{L_k}(\mathbf{y}_k), \\ t_{k+1} &= \frac{1 + \sqrt{1 + 4t_k^2}}{2}, \\ \mathbf{y}_{k+1} &= \mathbf{x}_k + \left(\frac{t_k - 1}{t_{k+1}}\right)(\mathbf{x}_k - \mathbf{x}_{k-1}). \end{aligned}$$

Note: FISTA can be easily extended to constrained convex NSO.

Analysis: The 3 Pillars

Lemma 1 (Well-Known) Let $f \in C^{1,1}_{L(f)}(\mathbb{R}^n)$. Then, for any $L \geq L(f)$,

$$f(\mathbf{x}) \leq f(\mathbf{y}) + \langle \mathbf{x} - \mathbf{y}, \nabla f(\mathbf{y}) \rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{y}\|^2, \text{ for every } \mathbf{x}, \mathbf{y} \in \mathbb{R}^n.$$

Analysis: The 3 Pillars

Lemma 1 (Well-Known) Let $f \in C^{1,1}_{L(f)}(\mathbb{R}^n)$. Then, for any $L \geq L(f)$,

$$f(\mathbf{x}) \leq f(\mathbf{y}) + \langle \mathbf{x} - \mathbf{y}, \nabla f(\mathbf{y}) \rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{y}\|^2, \text{ for every } \mathbf{x}, \mathbf{y} \in \mathbb{R}^n.$$

Lemma 2 (A Key Inequality) Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ and $L > 0$ such that $F(p_L(\mathbf{y})) \leq Q(p_L(\mathbf{y}), \mathbf{y})$. Then

$$F(\mathbf{x}) - F(p_L(\mathbf{y})) \geq \frac{L}{2}\|p_L(\mathbf{y}) - \mathbf{y}\|^2 + L\langle \mathbf{y} - \mathbf{x}, p_L(\mathbf{y}) - \mathbf{y} \rangle.$$

Analysis: The 3 Pillars

Lemma 1 (Well-Known) Let $f \in C^{1,1}_{L(f)}(\mathbb{R}^n)$. Then, for any $L \geq L(f)$,

$$f(\mathbf{x}) \leq f(\mathbf{y}) + \langle \mathbf{x} - \mathbf{y}, \nabla f(\mathbf{y}) \rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{y}\|^2, \text{ for every } \mathbf{x}, \mathbf{y} \in \mathbb{R}^n.$$

Lemma 2 (A Key Inequality) Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ and $L > 0$ such that $F(p_L(\mathbf{y})) \leq Q(p_L(\mathbf{y}), \mathbf{y})$. Then

$$F(\mathbf{x}) - F(p_L(\mathbf{y})) \geq \frac{L}{2}\|p_L(\mathbf{y}) - \mathbf{y}\|^2 + L\langle \mathbf{y} - \mathbf{x}, p_L(\mathbf{y}) - \mathbf{y} \rangle.$$

Lemma 3 (A Recursive Relation for Function Values) The sequences $\{\mathbf{x}_k, \mathbf{y}_k\}$ generated via FISTA satisfy for every $k \geq 1$

$$L_k^{-1} t_k^2 v_k - L_{k+1}^{-1} t_{k+1}^2 v_{k+1} \geq (\|\mathbf{u}_{k+1}\|^2 - \|\mathbf{u}_k\|^2)/2,$$

where $v_k := F(\mathbf{x}_k) - F(\mathbf{x}^*)$, $\quad \mathbf{u}_k := t_k \mathbf{x}_k - (t_k - 1)\mathbf{x}_{k-1} - \mathbf{x}^*$.

Theorem – Global Rate of Convergence for FISTA

Let $\{\mathbf{x}_k\}, \{\mathbf{y}_k\}$ be generated by FISTA. Then for any $k \geq 1$

$$F(\mathbf{x}_k) - F(\mathbf{x}^*) \leq \frac{2\alpha L(f)\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{(k+1)^2},$$

where $\alpha = 1$ for the constant stepsize setting and $\alpha = \eta$ for the backtracking stepsize setting.

Theorem – Global Rate of Convergence for FISTA

Let $\{\mathbf{x}_k\}, \{\mathbf{y}_k\}$ be generated by FISTA. Then for any $k \geq 1$

$$F(\mathbf{x}_k) - F(\mathbf{x}^*) \leq \frac{2\alpha L(f)\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{(k+1)^2},$$

where $\alpha = 1$ for the constant stepsize setting and $\alpha = \eta$ for the backtracking stepsize setting.

The number of iterations of FISTA required to obtain an ε-optimal solution, that is an $\tilde{\mathbf{x}}$ such that:

$$F(\tilde{\mathbf{x}}) - F_* \leq \varepsilon,$$

is at most $\sim O(1/\sqrt{\varepsilon})$. This clearly improves ISTA by **a square root factor.**

Theorem – Global Rate of Convergence for FISTA

Let $\{\mathbf{x}_k\}, \{\mathbf{y}_k\}$ be generated by FISTA. Then for any $k \geq 1$

$$F(\mathbf{x}_k) - F(\mathbf{x}^*) \leq \frac{2\alpha L(f)\|\mathbf{x}_0 - \mathbf{x}^*\|^2}{(k+1)^2},$$

where $\alpha = 1$ for the constant stepsize setting and $\alpha = \eta$ for the backtracking stepsize setting.

The number of iterations of FISTA required to obtain an ε-optimal solution, that is an $\tilde{\mathbf{x}}$ such that:

$$F(\tilde{\mathbf{x}}) - F_* \leq \varepsilon,$$

is at most $\sim O(1/\sqrt{\varepsilon})$. This clearly improves ISTA by **a square root factor.**

Do we practically achieve this theoretical rate?

Numerical Examples: Image Deblurring

$$\min_{\mathbf{x}}\{\|\mathbf{A}\mathbf{x}-\mathbf{b}\|^2+\lambda\|\mathbf{x}\|_1\}$$

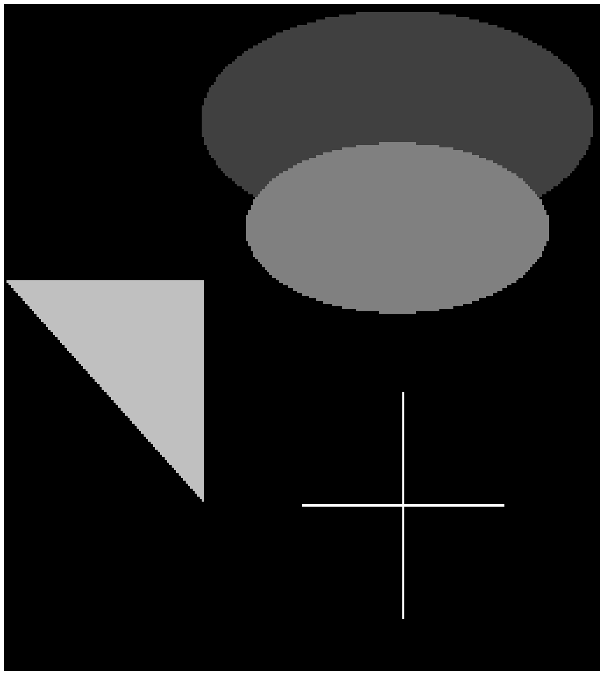
Compare ISTA versus FISTA on

- A Simple Test Image from Regularization Tool (Hansen, (97))
- The Cameraman Test Image
- More Simulations

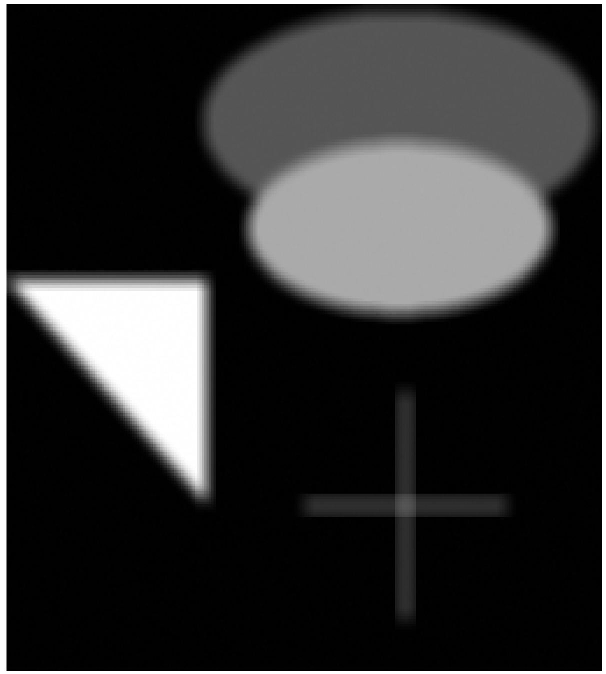
• Problems are in dimension d like
$d = 256 \times 256 = 65,536,$ or/and $512 \times 512 = 262,144$.
• The $d \times d$ matrix $\mathbf{A}$ is ***dense***.
• All problems solved with fixed λ and Gaussian noise.

Deblurring of A Simple Test Image

original

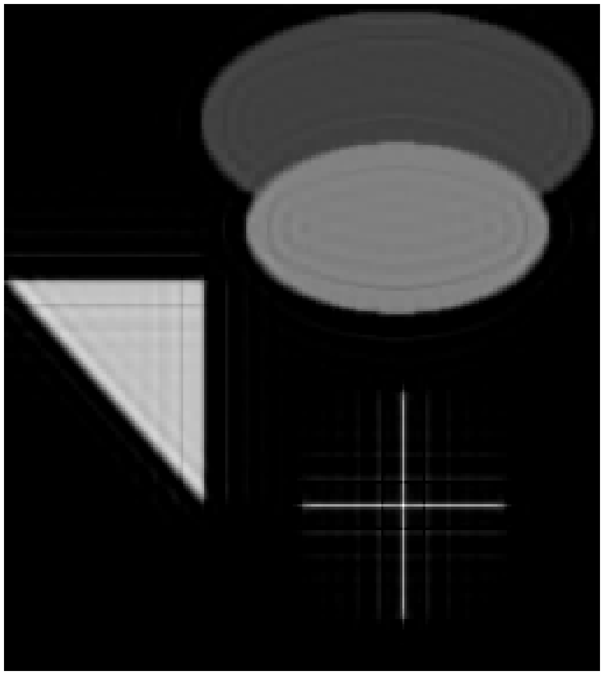


blurred and noisy

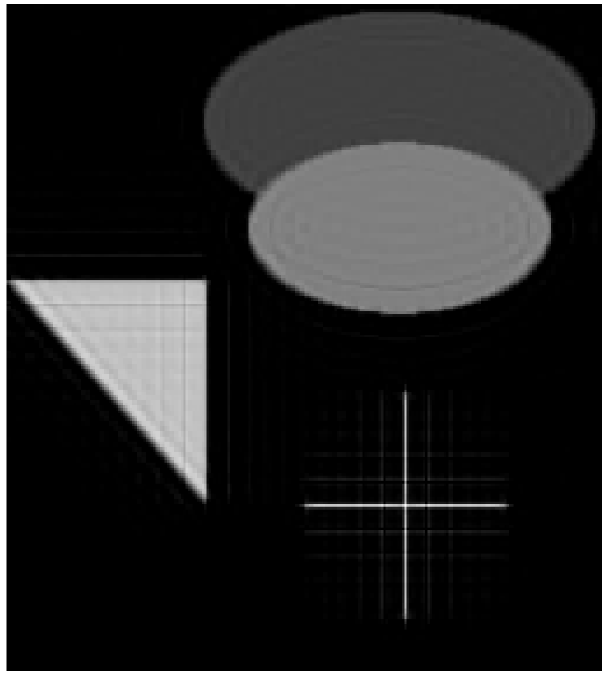


Output of 200 Iterations of ISTA versus 50 of FISTA

ISTA:$F_{200} = 0.42$



FISTA: $F_{50} = 0.23$



After tens of thousands of iterations, ISTA get stuck at $F = 0.32$!

Deblurring of the Cameraman

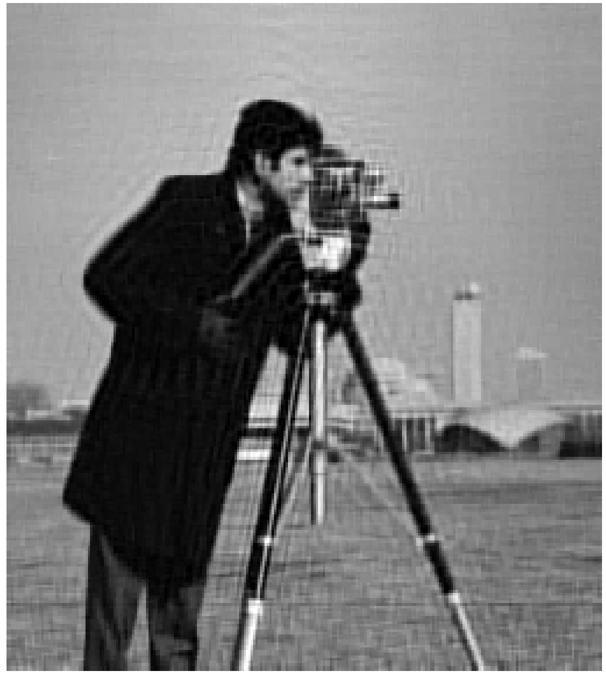
original

blurred and noisy

1000 Iterations of ISTA versus 100 of FISTA

ISTA: **1000 Iterations**

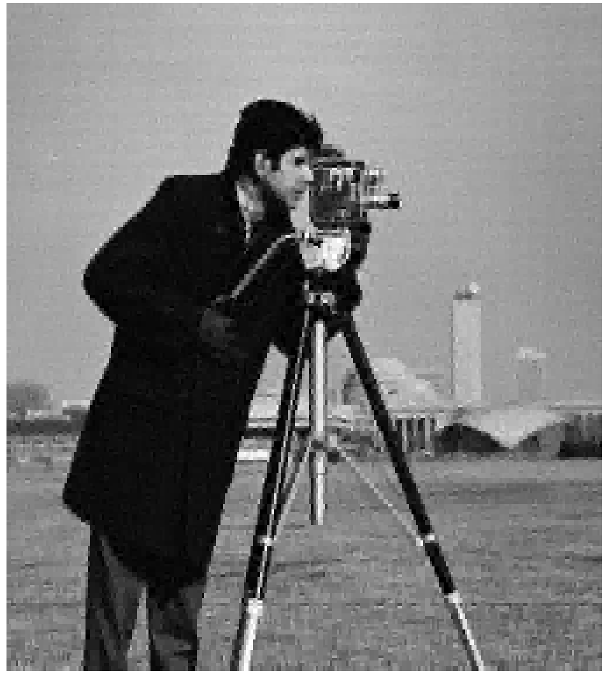
FISTA: **100 Iterations**



Original Versus Deblurring via FISTA

Original

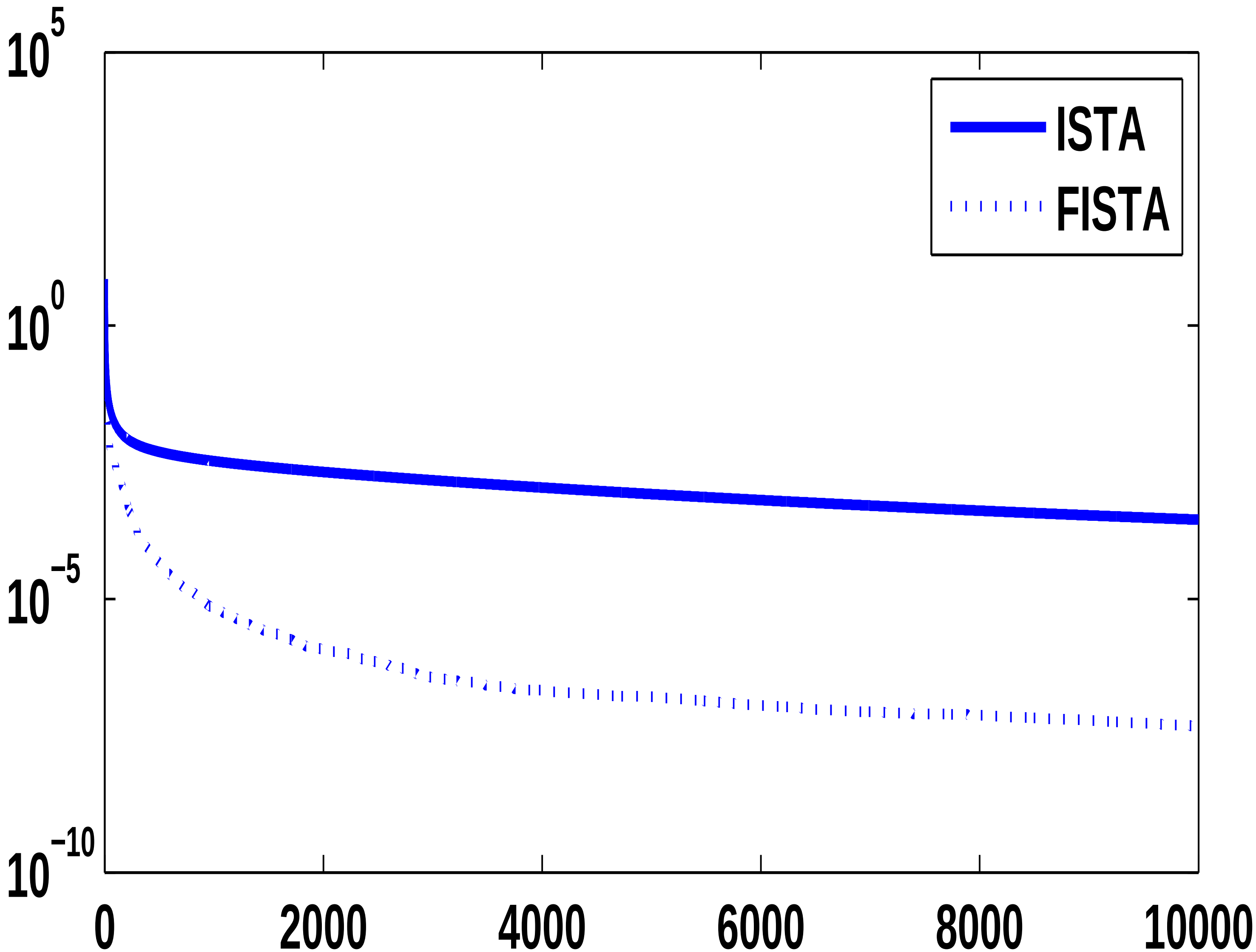
FISTA:1000 Iterations



More Simulations

- Previous simulations indicate that practically FISTA seems to be able to reach accuracies that are beyond the capabilities of ISTA.
- We further tested this hypothesis on an example with known optimal solution.
- This simulation shows that the results of FISTA are better by several order of magnitudes. After 10000 iterations our method reaches accuracy of approximately 10^{-7} while ISTA reaches an accuracy of 10^{-3}.
- Moreover, the value obtained by ISTA at iteration 10000 was already obtained by FISTA at iteration 254.
- The next figure describing function values of both methods for 10000 iterations speaks for itself!

Function Values errors $F(\mathbf{x}_k) - F(\mathbf{x}^*)$



Conclusions

- FISTA is a very simple and promising iterative scheme. Covers a broad class of problems arising in several recent diverse/key applications.
- Appears **even faster than the proven predicted theoretical rate!**
- Work in progress: potential for analyzing and designing faster algorithms in other areas, and with other types of nonsmooth regularizers.

Conclusions

- FISTA is a very simple and promising iterative scheme. Covers a broad class of problems arising in several recent diverse/key applications.
- Appears **even faster than the proven predicted theoretical rate!**
- Work in progress: potential for analyzing and designing faster algorithms in other areas, and with other types of nonsmooth regularizers.

Thank you for listening!