Alternating Direction Optimization for Imaging Inverse Problems

Mário A. T. Figueiredo

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

Joint work with:

Manya Afonso  José Bioucas-Dias  Mariana Almeida
Outline

1. Variational/optimization approaches to inverse problems
2. Formulations and key tools
3. The canonical ADMM and its extension for more than two functions
4. Linear-Gaussian observations: the SALSA algorithm.
5. Poisson observations: the PIDAL algorithm
6. Handling non periodic boundaries
7. Into the non-convex realm: blind deconvolution
Many inference criteria (in signal processing, machine learning) have the form

\[
\hat{x} \in \arg \min_{x \in \mathbb{R}^n} f(x) + \tau c(x)
\]

\( f : \mathbb{R}^n \rightarrow \mathbb{R} \) data fidelity, observation model, negative log-likelihood, loss,…

… usually smooth and convex.

\( c : \mathbb{R}^n \rightarrow \mathbb{R} \) regularization/penalty function, negative log-prior, …

… typically convex, often non-differentiable (to induce sparsity)

Examples: signal/image restoration/reconstruction, sparse representations, compressive sensing/imaging, linear regression, logistic regression, channel sensing, support vector machines, …
Unconstrained Versus Constrained Optimization

Unconstrained optimization formulation

\[ \hat{x} \in \arg \min_x f(x) + \tau c(x) \]  
(Tikhonov regularization)

Constrained optimization formulations

\[ \hat{x} \in \arg \min_x c(x) \]  
(Morozov regularization)

\[ \begin{array}{ll} \hat{x} \in & \arg \min_x f(x) \\ \text{s. t.} & f(x) \leq \varepsilon \end{array} \]  
(Ivanov regularization)

All “equivalent”, under mild conditions; often not equally convenient/easy

[Lorenz, 12]
A Fundamental Dichotomy: Analysis vs Synthesis

[Elad, Milanfar, Rubinstein, 2007], [Selesnick, F, 2010],

\[ \hat{x} \in \arg \min_{x \in \mathbb{R}^n} f(x) + \tau c(x) \]

**Synthesis** regularization:

\[ \hat{x} \in \arg \min_{x} \mathcal{L}(Ax) + \tau c(x) \]

\[ A = BW, \text{ where } B \text{ is the observation operator} \]

\[ W \text{ is a synthesis operator; e.g., a Parseval frame } WW^* = I \]

\[ \mathcal{L} \text{ depends on the noise model; e.g., } \mathcal{L}(z) = \frac{1}{2}\|z - y\|_2^2 \]

**Typical (sparseness-inducing) regularizer:**

\[ c(x) = \|x\|_1 \]

proper, lower semi-continuous (lsc), convex (not strictly), coercive.
A Fundamental Dichotomy: Analysis vs Synthesis (II)

[Elad, Milanfar, Rubinstein, 2007], [Selesnick, F, 2010],

\[ \hat{x} \in \arg \min_x \mathcal{L}(Ax) + \tau c(x) \]

**Analysis regularization**

\( x \) is the signal/image itself, \( A \) is the observation operator

typical frame-based analysis regularizer:

\[ c(x) = \| P x \|_1 \]

analysis operator (e.g., of a Parseval frame, \( P^*P = I \))

proper, lsc, convex (not strictly), and coercive.

**Total variation (TV) is also “analysis”; proper, lsc, convex (not strictly),**

... but not coercive.
Typical Convex Data Terms

Let: \( f(x) = \mathcal{L}(Ax) \) where \( \mathcal{L}(z) \equiv \sum_{i=1}^{m} \xi(z_i, y_i) \)

where \( \xi \) is one (e.g.) of these functions (log-likelihoods):

Gaussian observations: \( \xi_G(z, y) = \frac{1}{2}(z - y)^2 \) \( \rightarrow \mathcal{L}_G \)

Poissonian observations: \( \xi_P(z, y) = z + \nu_{\mathbb{R}^+}(z) - y \log(z) \) \( \rightarrow \mathcal{L}_P \)

Multiplicative noise: \( \xi_M(z, y) = L(z + e^{y-z}) \) \( \rightarrow \mathcal{L}_M \)

...all proper, lower semi-continuous (lsc), coercive, convex.

\( \mathcal{L}_G \) and \( \mathcal{L}_M \) are strictly convex. \( \mathcal{L}_P \) is strictly convex if \( y_i > 0, \forall i \)
A Key Tool: The Moreau Proximity Operator

The Moreau proximity operator [Moreau 62], [Combettes, Pesquet, Wajs, 01, 03, 05, 07, 10, 11].

\[
\text{prox}_{\tau c}(u) = \arg \min_{x} \frac{1}{2} \|x - u\|^2_2 + \tau c(x)
\]

Classical cases:

\[
c(z) = \nu_C(z) = \begin{cases} 
0 & \iff z \in C \\
+\infty & \iff z \notin C
\end{cases} \Rightarrow \text{prox}_{\tau c}(u) = \Pi_C(u)
\]

\[
c(z) = \frac{1}{2} \|z\|_2^2 \Rightarrow \text{prox}_{\tau c}(u) = \frac{u}{1 + \tau}
\]

\[
c(z) = \|z\|_1 \Rightarrow \text{prox}_{\tau c}(u) = \text{soft}(u, \tau) = \text{sign}(u) \odot \max(|u| - \tau, 0)
\]

Separability: \[c(z) = \sum_{i} c_i(z_i) \Rightarrow \left(\text{prox}_{\tau c}(u)\right)_i = \text{prox}_{\tau c_i}(u_i)\]
Moreau Proximity Operators

<table>
<thead>
<tr>
<th>$\phi(x)$</th>
<th>$\text{prox}_{\phi}(x)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$i$</td>
<td>$t_{[\omega,\bar{\omega}]}(x)$</td>
</tr>
<tr>
<td>$ii$</td>
<td>$\sigma_{[\omega,\bar{\omega}]}(x) = \begin{cases} \omega x &amp; \text{if } x &lt; 0 \ 0 &amp; \text{if } x = 0 \ \frac{\omega x}{\overline{x}} &amp; \text{otherwise} \end{cases}$</td>
</tr>
<tr>
<td>$iii$</td>
<td>$\psi(x) = \begin{cases} 0 &amp; \text{if } x &lt; 0 \ \psi(0) = 0 &amp; \text{if } x = 0 \ \psi(x) &amp; \text{if } x &gt; 0 \end{cases}$</td>
</tr>
<tr>
<td>$iv$</td>
<td>$\max{</td>
</tr>
<tr>
<td>$v$</td>
<td>$\kappa</td>
</tr>
<tr>
<td>$vi$</td>
<td>$\begin{cases} \kappa x^2 &amp; \text{if }</td>
</tr>
<tr>
<td>$vii$</td>
<td>$\omega</td>
</tr>
<tr>
<td>$viii$</td>
<td>$\omega</td>
</tr>
<tr>
<td>$ix$</td>
<td>$\begin{cases} \omega x &amp; \text{if } x \geq 0 \ +\infty &amp; \text{otherwise} \end{cases}$</td>
</tr>
<tr>
<td>$x$</td>
<td>$\begin{cases} -\omega x^{\frac{1}{q}} &amp; \text{if } x \geq 0 \ +\infty &amp; \text{otherwise} \end{cases}$</td>
</tr>
<tr>
<td>$xi$</td>
<td>$\omega x^{\frac{1}{q}}$</td>
</tr>
<tr>
<td>$xii$</td>
<td>$\begin{cases} x \ln(x) &amp; \text{if } x &gt; 0 \ 0 &amp; \text{if } x = 0 \ +\infty &amp; \text{otherwise} \end{cases}$</td>
</tr>
<tr>
<td>$xiii$</td>
<td>$\begin{cases} -\ln(x - \omega) + \ln(-\omega) &amp; \text{if } x \in [\omega, 0] \ -\ln(-\overline{x}) + \ln(-\omega) &amp; \text{if } x \in [0, \overline{\omega}] \ +\infty &amp; \text{otherwise} \end{cases}$</td>
</tr>
<tr>
<td>$xiv$</td>
<td>$\begin{cases} -\kappa \ln(x) + \tau x^2/2 + \kappa x &amp; \text{if } x \geq 0 \ +\infty &amp; \text{otherwise} \end{cases}$</td>
</tr>
<tr>
<td>$xv$</td>
<td>$\begin{cases} -\kappa \ln(x) + \alpha x &amp; \text{if } x \geq 0 \ +\infty &amp; \text{otherwise} \end{cases}$</td>
</tr>
<tr>
<td>$xvi$</td>
<td>$\begin{cases} -\kappa \ln(x) + \omega x^q &amp; \text{if } x \geq 0 \ +\infty &amp; \text{otherwise} \end{cases}$</td>
</tr>
<tr>
<td>$xiv$</td>
<td>$\begin{cases} -\kappa \ln(x - \omega) - \tau \ln(-\omega) &amp; \text{if } x \in [\omega, \overline{\omega}] \ +\infty &amp; \text{otherwise} \end{cases}$</td>
</tr>
</tbody>
</table>

...many more!

[Combettes, Pesquet, 2010]
Iterative Shrinkage/Thresholding (IST)

\[ \hat{x} \in \arg \min_{x \in \mathbb{R}^n} f(x) + \tau c(x) \]

\[ x_{k+1} = \text{prox}_{\tau c/\alpha} \left( x_k - \frac{1}{\alpha} \nabla f(x_k) \right) \]

Key condition in convergence proofs: \( \nabla f \) is Lipschitz

...not true, e.g., with Poisson or multiplicative noise.

Not directly applicable with analysis formulations (see [Loris, Verhoeven, 11])

IST is usually **slow** (specially if \( \tau \) is small); several accelerated versions:

- **Two-step IST** (TwIST) [Bioucas-Dias, F, 07]
- **Fast IST** (FISTA) [Beck, Teboulle, 09], [Tseng, 08]
- **Continuation** [Hale, Yin, Zhang, 07], [Wright, Nowak, F, 07, 09]
- **SpaRSA** [Wright, Nowak, F, 08, 09]
Alternating Direction Method of Multipliers (ADMM)

Unconstrained (convex) optimization problem: \[ \min_{\mathbf{z} \in \mathbb{R}^d} f_1(\mathbf{z}) + f_2(G \mathbf{z}) \]

**ADMM** [Glowinski, Marrocco, 75], [Gabay, Mercier, 76], [Gabay, 83], [Eckstein, Bertsekas, 92]

\[
\begin{align*}
\mathbf{z}_{k+1} &= \arg \min_{\mathbf{z}} f_1(\mathbf{z}) + \frac{\mu}{2} \lVert G \mathbf{z} - \mathbf{u}_k - \mathbf{d}_k \rVert^2 \\
\mathbf{u}_{k+1} &= \arg \min_{\mathbf{u}} f_2(\mathbf{u}) + \frac{\mu}{2} \lVert G \mathbf{z}_{k+1} - \mathbf{u} - \mathbf{d}_k \rVert^2 \\
\mathbf{d}_{k+1} &= \mathbf{d}_k - (G \mathbf{z}_{k+1} - \mathbf{u}_{k+1})
\end{align*}
\]

**Interpretations:** variable splitting + augmented Lagrangian + NLBGS;
Douglas-Rachford splitting on the dual [Eckstein, Bertsekas, 92]
split-Bregman approach [Goldstein, Osher, 08]

Explosion of applications in signal processing, machine learning, statistics, ...
[Giovanneli, Coulais, 05], Giannakis et al, 08, 09,...], [Tomioka et al, 09], [Boyd et al, 11], [Goldfarb, Ma, 10,...], [Fessler et al, 11, ...], [Mota et al, 10], [Jakovetić et al, 12], [Banerjee et al, 12], [Esser, 09], [Ng et al, 20], [Setzer, Steidl, Teuber, 09], [Yang, Zhang, 11], [Combettes, Pesquet, 10,...], [Chan, Yang, Yuan, 11], ............
The problem

$$\min_{z \in \mathbb{R}^d} f_1(z) + f_2(Gz)$$

$f_1$ and $f_2$ are closed, proper, convex; $G$ has full column rank.

$$(z_k, k = 0, 1, 2, \ldots)$$ is the sequence produced by ADMM, with $\mu > 0$ then, if the problem has a solution, say $\bar{z}$, then

$$\lim_{k \to \infty} z_k = \bar{u}$$

(inexact minimizations allowed, as long as the errors are absolutely summable).
Applying ADMM

Synthesis formulation:
\[
\min_x \mathcal{L}(BWx) + \tau c(x)
\]

Template problem for ADMM
\[
\min_z f_1(z) + f_2(Gz)
\]

Naïve mapping:
\[
G = BW, \quad f_1 = \tau c, \quad f_2 = L
\]

ADMM

\[
z_{k+1} = \arg\min_z \tau c(z) + \frac{\mu}{2}\|BWz - u_k - d_k\|^2
\]

\[
u_{k+1} = \arg\min_u \mathcal{L}(u) + \frac{\mu}{2}\|BWz_{k+1} - u - d_k\|^2
\]

\[
d_{k+1} = d_k - (BWz_{k+1} - u_{k+1})
\]

usually hard!

usually easy

\[
\text{prox}_{\mathcal{L}/\mu}
\]
Applying ADMM

Analysis formulation:

\[ \min_x \mathcal{L}(Bx) + \tau c(Px) \]

Template problem for ADMM

\[ \min_z f_1(z) + f_2(Gz) \]

Naïve mapping: \( G = P, \quad f_1 = \mathcal{L} \circ B, \quad f_2 = \tau c \)

\[ z_{k+1} = \arg \min_z \mathcal{L}(Bz) + \frac{\mu}{2}\|Pz - u_k - d_k\|^2 \]

\[ u_{k+1} = \arg \min_u \tau c(u) + \frac{\mu}{2}\|Pz_{k+1} - u - d_k\|^2 \]

\[ d_{k+1} = d_k - (Pz_{k+1} - u_{k+1}) \]

Easy if: \( \mathcal{L} \) is quadratic and \( B \) and \( P \) diagonalized by common transform (e.g., DFT) (split-Bregman [Goldstein, Osher, 08])

usually easy \( \text{prox}_{\tau c/\mu} \)
Consider a more general problem

\[
\min_{z \in \mathbb{R}^d} \sum_{j=1}^{J} g_j(H^{(j)} z) \quad (P)
\]

where \( g_j : \mathbb{R}^{p_j} \rightarrow \mathbb{R} \) are proper, closed, convex functions and \( H^{(j)} \in \mathbb{R}^{p_j \times d} \) are arbitrary matrices.

There are many ways to write \((P)\) as

\[
\min_{z \in \mathbb{R}^d} f_1(z) + f_2(G z)
\]

We propose:

\[
f_1(z) = 0, \quad G = \begin{bmatrix} H^{(1)} \\ \vdots \\ H^{(J)} \end{bmatrix}, \quad f_2 \left( \begin{bmatrix} u^{(1)} \\ \vdots \\ u^{(J)} \end{bmatrix} \right) = \sum_{j=1}^{J} g_j(u^{(j)})
\]
ADMM for Two or More Functions

\[
\min_{\mathbf{z} \in \mathbb{R}^d} \sum_{j=1}^{J} g_j(\mathbf{H}(j)\mathbf{z}), \quad \min_{\mathbf{z} \in \mathbb{R}^d} f_2(\mathbf{G} \mathbf{z}), \quad \mathbf{G} = \begin{bmatrix} \mathbf{H}(1) \\ \vdots \\ \mathbf{H}(J) \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} \mathbf{u}(1) \\ \vdots \\ \mathbf{u}(J) \end{bmatrix}
\]

\[
\mathbf{z}_{k+1} = \left( \sum_{j=1}^{J} (\mathbf{H}(j))^* \mathbf{H}(j) \right)^{-1} \sum_{j=1}^{J} (\mathbf{H}(j))^* \left( \mathbf{u}_k^{(j)} + \mathbf{d}_k^{(j)} \right)
\]

\[
\mathbf{u}_{k+1}^{(1)} = \arg \min_{\mathbf{u}} g_1(\mathbf{u}) + \frac{\mu}{2} \| \mathbf{u} - \mathbf{H}(1)\mathbf{z}_{k+1} + \mathbf{d}_k^{(1)} \|^2 = \text{prox}_{g_1/\mu}(\mathbf{H}(1)\mathbf{z}_{k+1} - \mathbf{d}_k^{(1)})
\]

\[
\vdots
\]

\[
\mathbf{u}_{k+1}^{(J)} = \arg \min_{\mathbf{u}} g_J(\mathbf{u}) + \frac{\mu}{2} \| \mathbf{u} - \mathbf{H}(J)\mathbf{z}_{k+1} + \mathbf{d}_k^{(J)} \|^2 = \text{prox}_{g_J/\mu}(\mathbf{H}(J)\mathbf{z}_{k+1} - \mathbf{d}_k^{(J)})
\]

\[
\mathbf{d}_{k+1}^{(1)} = \mathbf{d}_k^{(1)} - (\mathbf{H}(1)\mathbf{z}_{k+1} - \mathbf{u}_{k+1}^{(1)})
\]

\[
\vdots
\]

\[
\mathbf{d}_{k+1}^{(J)} = \mathbf{d}_k^{(J)} - (\mathbf{H}(J)\mathbf{z}_{k+1} - \mathbf{u}_{k+1}^{(J)})
\]
ADMM for Two or More Functions

\[
\mathbf{z}_{k+1} = \left( \sum_{j=1}^{J} (\mathbf{H}^{(j)})^* \mathbf{H}^{(j)} \right)^{-1} \sum_{j=1}^{J} (\mathbf{H}^{(j)})^* \left( \mathbf{u}^{(j)}_k + \mathbf{d}^{(j)}_k \right)
\]

\[
\mathbf{u}^{(1)}_{k+1} = \text{prox}_{g_1/\mu} (\mathbf{H}^{(1)} \mathbf{z}_{k+1} - \mathbf{d}^{(1)}_k)
\]

\[
\vdots
\]

\[
\mathbf{u}^{(J)}_{k+1} = \text{prox}_{g_1/\mu} (\mathbf{H}^{(J)} \mathbf{z}_{k+1} - \mathbf{d}^{(J)}_k)
\]

\[
\mathbf{d}^{(1)}_{k+1} = \mathbf{d}^{(1)}_k - (\mathbf{H}^{(1)} \mathbf{z}_{k+1} - \mathbf{u}^{(1)}_{k+1})
\]

\[
\vdots \quad \vdots \quad \vdots
\]

\[
\mathbf{d}^{(J)}_{k+1} = \mathbf{d}^{(J)}_k - (\mathbf{H}^{(J)} \mathbf{z}_{k+1} - \mathbf{u}^{(J)}_{k+1})
\]

Conditions for easy applicability: inexpensive proximity operators, inexpensive matrix inversion, ...a cursing and a blessing!
ADMM for Two or More Functions

Applies to sum of convex terms

\[
\min_{\mathbf{z} \in \mathbb{R}^d} \sum_{j=1}^{J} g_j (\mathbf{H}^{(j)} \mathbf{z})
\]

Computation of proximity operators is parallelizable

Handling of matrices is isolated in a pure quadratic problem

Conditions for easy applicability:
- inexpensive proximity operators
- inexpensive matrix inversion

Matrix inversion may be a curse or a blessing! (more later)

Similar algorithm: *simultaneous directions method of multipliers* (SDMM)

[Setzer, Steidl, Teuber, 2010], [Combettes, Pesquet, 2010]

Other ADMM versions for more than two functions

[Hong, Luo, 2012, 2013], [Ma, 2012]
Linear/Gaussian Observations: Frame-Based Analysis

Problem: \( \hat{x} \in \arg \min_x \frac{1}{2} \| Ax - y \|_2^2 + \tau \| Px \|_1 \)

Template: \( \min_{z \in \mathbb{R}^d} \sum_{j=1}^{J} g_j(H^{(j)}z) \)

Mapping: \( J = 2, \ g_1(z) = \frac{1}{2} \| z - y \|_2^2, \ g_2(z) = \tau \| z \|_1 \)

\( H^{(1)} = A, \quad H^{(2)} = P, \)

Convergence conditions: \( g_1 \) and \( g_2 \) are closed, proper, and convex.

\( G = \begin{bmatrix} A \\ P \end{bmatrix} \) has full column rank.

Resulting algorithm: SALSA

(split augmented Lagrangian shrinkage algorithm) [Afonso, Bioucas-Dias, F, 2009, 2010]
ADMM for the Linear/Gaussian Problem: SALSA

Key steps of SALSA (both for analysis and synthesis):

Moreau proximity operator of

\[ g_1(z) = \frac{1}{2} \| z - y \|_2^2, \]

\[ \text{prox}_{g_1/\mu}(u) = \arg\min_z \frac{1}{2\mu} \| z - y \|_2^2 + \frac{1}{2} \| z - u \|_2^2 = \frac{y + \mu u}{1 + \mu} \]

Moreau proximity operator of

\[ g_2(z) = \tau \| z \|_1, \]

\[ \text{prox}_{g_2/\mu}(u) = \text{soft}\left( u, \frac{\tau}{\mu} \right) \]

Matrix inversion:

\[ z_{k+1} = \left[ A^* A + P^* P \right]^{-1} \left( A^* \left( u_k^{(1)} + d_k^{(1)} \right) + P^* \left( u_k^{(2)} + d_k^{(2)} \right) \right) \]

...next slide!
Handling the Matrix Inversion: Frame-Based Analysis

Frame-based analysis: \[ (A^*A + P^*P)^{-1} = (A^*A + I)^{-1} \]

Periodic deconvolution: \[ A = U^*DU \]

Compressive imaging (MRI): \[ A = MU \]

Inpainting (recovery of lost pixels): \[ A = S \]

Parseval frame

DFT (FFT)

Subsampling matrix: \[ MM^* = I \]

Matrix inversion lemma

Subsampling matrix: \[ S^*S \] is diagonal

Diagonal inversion
SALSA for Frame-Based Synthesis

Problem: \( \hat{x} \in \arg \min_x \frac{1}{2} \| Ax - y \|_2^2 + \tau \| x \|_1 \)

Template: \( \min \sum_{j=1}^{J} g_j(H^{(j)}z) \quad A = BW \)

Mapping: \( J = 2, \quad g_1(z) = \frac{1}{2} \| z - y \|_2^2, \quad g_2(z) = \tau \| z \|_1 \)

\( H^{(1)} = A = BW \quad H^{(2)} = I, \)

Convergence conditions: \( g_1 \) and \( g_2 \) are closed, proper, and convex.

\( G = \begin{bmatrix} B & W \\ I \end{bmatrix} \) has full column rank.
Handling the Matrix Inversion: Frame-Based Synthesis

Frame-based analysis: \[
\left[ \sum_{j=1}^{J} (H^{(j)})^* H^{(j)} \right]^{-1} = \left[ W^* B^* B W + I \right]^{-1}
\]

Periodic deconvolution: \[ B = U^* D U \]

Compressive imaging (MRI): \[ B = M U \]

Inpainting (recovery of lost pixels): \[ B = S \]

\[ O(n \log n) \]
SALSA Experiments

9x9 uniform blur, 40dB BSNR

undecimated Haar frame, \( \ell_1 \) regularization.

TV regularization
SALSA Experiments

Image inpainting
(50% missing)

Conjecture: SALSA is fast because it’s blessed by the matrix inversion

The inverted matrix (e.g., $\mathbf{A}^* \mathbf{A} + \mathbf{I}$) is (almost) the Hessian of the data term;

...second-order (curvature) information (as Newton’s method)
Frame-Based Analysis Deconvolution of Poissonian Images

Problem template: \( \min_{u \in \mathbb{R}^d} \sum_{j=1}^{J} g_j(H^{(j)}u) \) \hspace{1cm} (P1)

Frame-analysis regularization: \( \hat{x} \in \arg \min_x \mathcal{L}_P(Bx) + \lambda \| Px \|_1 + \nu_{\mathbb{R}^n_+}(x) \)

Same form as (P1) with: \( J = 3, \; g_1 = \mathcal{L}_P, \; g_2 = \| \cdot \|_1, \; g_3 = \nu_{\mathbb{R}^n_+} \)

Convergence conditions: \( g_1, g_2, \) and \( g_3 \) are closed, proper, and convex.

\[ G = \begin{bmatrix} B \\ P \\ I \end{bmatrix} \quad \text{has full column rank} \]

Required inversion: \( \left[ B^*B + P^*P + I \right]^{-1} = \left[ B^*B + 2I \right]^{-1} \)

...again, easy in periodic deconvolution, MRI, inpainting, ...
Proximity Operator of the Poisson Log-Likelihood

Proximity operator of the Poisson log-likelihood

$$\text{prox}_{\mathcal{L}/\mu}(u) = \arg \min_z \sum_i \xi(z_i, y_i) + \frac{\mu}{2} \| z - u \|^2$$

$$\xi(z, y) = z + \nu_{\mathbb{R}_+}(z) - y \log(z+)$$

Separable problem with closed-form (non-negative) solution

[Combettes, Pesquet, 09, 11]:

$$\text{prox}_{\xi(\cdot, y)}(u) = \frac{1}{2} \left( u - \frac{1}{\mu} + \sqrt{\left( u - (1/\mu) \right)^2 + 4 \frac{y}{\mu}} \right)$$

Proximity operator of $g_3 = \nu_{\mathbb{R}_+^n}$ is simply

$$\text{prox}_{\nu_{\mathbb{R}_+^n}}(x) = (x)_+$$
Experiments

Comparison with [Dupé, Fadili, Starck, 09] and [Starck, Bijaoui, Murtagh, 95]

PIDAL = Poisson image deconvolution by augmented Lagrangian

<table>
<thead>
<tr>
<th>Image</th>
<th>$M$</th>
<th>PIDAL-TV MAE</th>
<th>iterations</th>
<th>time</th>
<th>PIDAL-FA MAE</th>
<th>iterations</th>
<th>time</th>
<th>[Dupé, Fadili, Starck, 09] MAE</th>
<th>iterations</th>
<th>time</th>
<th>[Starck et al, 95] MAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cameraman</td>
<td>5</td>
<td>0.27</td>
<td>120</td>
<td>22</td>
<td>0.26</td>
<td>70</td>
<td>13</td>
<td>0.35</td>
<td>6</td>
<td>4.5</td>
<td>0.37</td>
</tr>
<tr>
<td>Cameraman</td>
<td>30</td>
<td>1.29</td>
<td>51</td>
<td>9.1</td>
<td>1.22</td>
<td>39</td>
<td>7.4</td>
<td>1.47</td>
<td>98</td>
<td>75</td>
<td>2.06</td>
</tr>
<tr>
<td>Cameraman</td>
<td>100</td>
<td>3.99</td>
<td>33</td>
<td>6.0</td>
<td>3.63</td>
<td>36</td>
<td>6.8</td>
<td>4.31</td>
<td>426</td>
<td>318</td>
<td>5.58</td>
</tr>
<tr>
<td>Cameraman</td>
<td>255</td>
<td>8.99</td>
<td>32</td>
<td>5.8</td>
<td>8.45</td>
<td>37</td>
<td>7.0</td>
<td>10.26</td>
<td>480</td>
<td>358</td>
<td>12.3</td>
</tr>
<tr>
<td>Neuron</td>
<td>5</td>
<td>0.17</td>
<td>117</td>
<td>3.6</td>
<td>0.18</td>
<td>66</td>
<td>2.9</td>
<td>0.19</td>
<td>6</td>
<td>3.9</td>
<td>0.19</td>
</tr>
<tr>
<td>Neuron</td>
<td>30</td>
<td>0.68</td>
<td>54</td>
<td>1.8</td>
<td>0.77</td>
<td>44</td>
<td>2.0</td>
<td>0.82</td>
<td>161</td>
<td>77</td>
<td>0.95</td>
</tr>
<tr>
<td>Neuron</td>
<td>100</td>
<td>1.75</td>
<td>43</td>
<td>1.4</td>
<td>2.04</td>
<td>41</td>
<td>1.8</td>
<td>2.32</td>
<td>427</td>
<td>199</td>
<td>2.88</td>
</tr>
<tr>
<td>Neuron</td>
<td>255</td>
<td>3.52</td>
<td>43</td>
<td>1.4</td>
<td>3.47</td>
<td>42</td>
<td>1.9</td>
<td>5.25</td>
<td>202</td>
<td>97</td>
<td>6.31</td>
</tr>
<tr>
<td>Cell</td>
<td>5</td>
<td>0.12</td>
<td>56</td>
<td>10</td>
<td>0.11</td>
<td>36</td>
<td>7.6</td>
<td>0.12</td>
<td>6</td>
<td>4.5</td>
<td>0.12</td>
</tr>
<tr>
<td>Cell</td>
<td>30</td>
<td>0.57</td>
<td>31</td>
<td>6.5</td>
<td>0.54</td>
<td>39</td>
<td>8.2</td>
<td>0.56</td>
<td>85</td>
<td>64</td>
<td>0.47</td>
</tr>
<tr>
<td>Cell</td>
<td>100</td>
<td>1.71</td>
<td>85</td>
<td>15</td>
<td>1.46</td>
<td>31</td>
<td>6.4</td>
<td>1.72</td>
<td>215</td>
<td>162</td>
<td>1.37</td>
</tr>
<tr>
<td>Cell</td>
<td>255</td>
<td>3.77</td>
<td>89</td>
<td>17</td>
<td>3.33</td>
<td>34</td>
<td>7.0</td>
<td>5.45</td>
<td>410</td>
<td>308</td>
<td>3.10</td>
</tr>
</tbody>
</table>

MAE $\equiv \frac{\|\hat{x} - x\|_1}{n}$
Morozov Formulation

Unconstrained optimization formulation:
\[
\min_x \frac{1}{2} \| Ax - y \|_2^2 + \tau c(x)
\]

Constrained optimization (Morozov) formulation:
\[
\min_x c(x) \quad \text{s.t.} \quad \| Ax - y \|_2^2 \leq \varepsilon
\]

basis pursuit denoising, if \( c(x) = \| x \|_1 \)

[Chen, Donoho, Saunders, 1998]

Both analysis and synthesis can be used:

- frame-based analysis,
  \[
  c(x) = \| Px \|_1
  \]

- frame-based synthesis
  \[
  c(x) = \| x \|_1
  \]

\[
A = BW
\]
Proposed Approach for Constrained Formulation

Constrained problem: \[
\min_x c(x)
\]
\[
\text{s.t. } \|Ax - y\|_2^2 \leq \varepsilon
\]

...can be written as
\[
\min_x c(x) + \nu_B(\varepsilon, y)(Ax)
\]

\[B(\varepsilon, y) = \{x \in \mathbb{R}^n : \|x - y\|_2 \cdot \varepsilon\}\]

...which has the form
\[
\min_{u \in \mathbb{R}^d} \sum_{j=1}^J g_j(H^{(j)}u) \quad (P1)
\]

with \( J = 2 \), \( g_1(z) = c(z) \), \( H^{(1)} = I \)

\( g_2(z) = \nu_{E(\varepsilon, y)}(z) \), \( H^{(2)} = A \)

Resulting algorithm: C-SALSA (constrained-SALSA)

[Afonso, Bioucas-Dias, F, 2010,2011]
Some Aspects of C-SALSA

Moreau proximity operator of $\mathcal{B}(\varepsilon, y)$ is simply a projection on an $\ell_2$ ball:

$$
\text{prox}_{\mathcal{B}(\varepsilon, y)}(u) = \arg \min_z \mathcal{B}(\varepsilon, y) + \frac{1}{2} \| z - u \|^2
$$

$$
= \begin{cases} 
  u & \iff \| u - y \|_2 \leq \varepsilon \\
  y + \frac{\varepsilon (u - y)}{\| u - y \|_2} & \iff \| u - y \|_2 > \varepsilon
\end{cases}
$$

As SALSA, also C-SALSA involves inversion of the form

$$
\left[ W^* B^* B W + I \right]^{-1} \quad \text{or} \quad \left[ B^* B + P^* P \right]^{-1}
$$

…all the same tricks as above.
### Image deconvolution benchmark problems:

<table>
<thead>
<tr>
<th>Experiment</th>
<th>blur kernel</th>
<th>$\sigma^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9 × 9 uniform</td>
<td>0.56^2</td>
</tr>
<tr>
<td>2A</td>
<td>Gaussian</td>
<td>2</td>
</tr>
<tr>
<td>2B</td>
<td>Gaussian</td>
<td>8</td>
</tr>
<tr>
<td>3A</td>
<td>$h_{ij} = 1/(1 + i^2 + j^2)$</td>
<td>2</td>
</tr>
<tr>
<td>3B</td>
<td>$h_{ij} = 1/(1 + i^2 + j^2)$</td>
<td>8</td>
</tr>
</tbody>
</table>

**NESTA:** [Becker, Bobin, Candès, 2011]

**SPGL1:** [van den Berg, Friedlander, 2009]

### Frame-synthesis

<table>
<thead>
<tr>
<th>Expt.</th>
<th>Avg. calls to $B, B^H$ (min/max)</th>
<th>Iterations</th>
<th>CPU time (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SPGL1</td>
<td>NESTA</td>
<td>C-SALSA</td>
</tr>
<tr>
<td>1</td>
<td>1029 (659/1290)</td>
<td>3520 (3501/3541)</td>
<td>398 (388/406)</td>
</tr>
<tr>
<td>2A</td>
<td>511 (279/663)</td>
<td>4897 (4777/4981)</td>
<td>451 (442/460)</td>
</tr>
<tr>
<td>2B</td>
<td>377 (141/532)</td>
<td>3397 (3345/3473)</td>
<td>362 (355/370)</td>
</tr>
<tr>
<td>3A</td>
<td>675 (378/772)</td>
<td>2622 (2589/2661)</td>
<td>172 (166/175)</td>
</tr>
<tr>
<td>3B</td>
<td>404 (300/475)</td>
<td>2446 (2401/2485)</td>
<td>134 (130/136)</td>
</tr>
</tbody>
</table>

### Frame-analysis

<table>
<thead>
<tr>
<th>Expt.</th>
<th>Avg. calls to $B, B^H$ (min/max)</th>
<th>Iterations</th>
<th>CPU time (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NESTA</td>
<td>C-SALSA</td>
<td>NESTA</td>
</tr>
<tr>
<td>1</td>
<td>2881 (2861/2889)</td>
<td>413 (404/419)</td>
<td>720</td>
</tr>
<tr>
<td>2A</td>
<td>2451 (2377/2505)</td>
<td>362 (344/371)</td>
<td>613</td>
</tr>
<tr>
<td>2B</td>
<td>2139 (2065/2197)</td>
<td>290 (278/299)</td>
<td>535</td>
</tr>
<tr>
<td>3A</td>
<td>2203 (2181/2217)</td>
<td>137 (134/143)</td>
<td>551</td>
</tr>
</tbody>
</table>

### Total-variation

<table>
<thead>
<tr>
<th>Expt.</th>
<th>Avg. calls to $B, B^H$ (min/max)</th>
<th>Iterations</th>
<th>CPU time (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NESTA</td>
<td>C-SALSA</td>
<td>NESTA</td>
</tr>
<tr>
<td>1</td>
<td>7783 (7767/7795)</td>
<td>695 (680/710)</td>
<td>1945</td>
</tr>
<tr>
<td>2A</td>
<td>7323 (7291/7351)</td>
<td>559 (536/578)</td>
<td>1830</td>
</tr>
<tr>
<td>2B</td>
<td>6828 (6775/6883)</td>
<td>299 (269/329)</td>
<td>1707</td>
</tr>
<tr>
<td>3A</td>
<td>6594 (6513/6661)</td>
<td>176 (98/209)</td>
<td>1649</td>
</tr>
<tr>
<td>3B</td>
<td>5514 (5417/5585)</td>
<td>108 (104/110)</td>
<td>1379</td>
</tr>
</tbody>
</table>
Non-Periodic Deconvolution

Analysis formulation for deconvolution

$$\hat{x} \in \arg\min_x \frac{1}{2}\|Ax - y\|_2^2 + \tau c(x)$$

ADMM / SALSA handles this “easily” if $A$ is circulant (periodic convolution)

Periodicity is an artificial assumption

...as are other boundary conditions (BC)

Neumann

Dirichlet

$A$ is (block) circulant

$A$ is (block) Toeplitz + Hankel

[Ng, Chan, Tang, 1999]

$A$ is (block) Toeplitz
Why Periodic, Neumann, Dirichlet Boundary Conditions are “wrong”
Non-Periodic Deconvolution

A natural BC: unknown values
[Chan, Yip, Park, 05], [Reeves, 05], [Sorel, 12], [Almeida, F, 12,13], [Matakos, Ramani, Fessler, 12, 13]

\[ \hat{x} \in \arg\min_x \frac{1}{2} \|MBx - y\|_2^2 + \tau c(x) \]

unknown values

- convolution, arbitrary BC
- masking
- mask
- periodic convolution
Non-Periodic Deconvolution (Frame-Analysis)

Problem: \( \hat{x} \in \arg\min_x \frac{1}{2} \|MBx - y\|_2^2 + \tau \|Px\|_1 \)

Template: \( \min_{z \in \mathbb{R}^d} \sum_{j=1}^{J} g_j(H^{(j)}z) \)

Naïve mapping: \( J = 2 \), \( g_1(z) = \frac{1}{2} \|z - y\|_2^2 \), \( g_2(z) = \tau \|z\|_1 \)

\( H^{(1)} = MB \quad H^{(2)} = P \),

Difficulty: need to compute \( \left[ B^*M^*MB + P^*P \right]^{-1} = \left[ B^*M^*MB + I \right]^{-1} \)

...tricks above no longer applicable.
Non-Periodic Deconvolution (Frame-Analysis)

Problem: \[ \hat{x} \in \text{arg min} \frac{1}{2} \|MBx - y\|_2^2 + \tau \|Px\|_1 \]

Template: \[ \min_{z \in \mathbb{R}^d} \sum_{j=1}^{J} g_j(H^{(j)}z) \]

Better mapping: \[ J = 2, \quad g_1(z) = \frac{1}{2} \|Mz - y\|_2^2, \quad g_2(z) = \tau \|z\|_1 \]
\[
\begin{align*}
H^{(1)} &= B & H^{(2)} &= P, \\
\left[ B^*B + P^*P \right]^{-1} &= \left[ B^*B + I \right]^{-1} \\
\text{prox}_{g_2/\mu}(u) &= \arg \min_z \frac{1}{2\mu} \|Mz - y\|_2^2 + \frac{1}{2} \|z - u\|_2^2 \\
&= \left( M^TM + \mu I \right)^{-1} (M^Ty + \mu u)
\end{align*}
\]

easy via FFT (\( B \) is circulant)

CIMI, Toulouse, 2013
Non-Periodic Deconvolution: Example (19x19 uniform blur)

Assuming periodic BC

Edge tapering

FA-BC (ISNR = -2.52dB)

FA-ET (ISNR = 3.06dB)

Proposed

FA-MD (ISRN = 10.63dB)
Non-Periodic Deconvolution: Example (19x19 motion blur)

original (256 × 256)  
observed (238 × 238)  

Assuming periodic BC  
Edge tapering

TV-BC (ISNR = 0.91dB)  
TV-ET (ISNR = 9.38dB)  

TV-MD (ISNR = 12.59dB)
Non-Periodic Deconvolution + Inpainting

\[ \hat{x} \in \arg \min_x \frac{1}{2} \| MBx - y \|_2^2 + \tau c(x) \]

Mask the boundary and missing pixels

periodic convolution

Also applicable to super-resolution (ongoing work)
Non-Periodic Deconvolution via Accelerated IST

The synthesis formulation is easily handled by IST (or FISTA, TwiST, SpaRSA,...) [Matakos, Ramani, Fessler, 12, 13]

\[
\hat{x} \in \arg\min_x \frac{1}{2} \| MBWx - y \|_2^2 + \tau \| x \|_1
\]

periodic convolution

mask

Parseval frame synthesis

Ingredients:

\[
\text{prox}_{\tau \| \cdot \|_1}(u) = \text{soft}(u, \tau)
\]

\[
\nabla \frac{1}{2} \| MBWx - y \|_2^2 = W^*B^*M^* (MBWx - y)
\]

(analysis formulation cannot be addressed by IST, FIST, SpaRSA, TwiST,...)
Blind Image Deconvolution (BID)

\[ y = h \ast x + n \]
Both \( x \) and \( h \) are unknown

Objective function (non-convex):

\[
C_\lambda (x, h) = \frac{1}{2} \| y - M B x \|_2^2 + \lambda \sum_{i=1}^{m} (\| F_i x \|_2)^q + \iota_{S^+}(h)
\]

Support and positivity

\([\text{Almeida and F, 13}]\)

\( \Phi(x) \) is “enhanced” TV; \( q \in (0, 1] \) (typically 0.5);

\( F_i \) is the convolution with four “edge filters” at location \( i \)

\( F_i \in \mathbb{R}^{4 \times m} \)

\( F_i x \in \mathbb{R}^4 \)
Blind Image Deconvolution (BID)

Algorithm 1: Continuation-based BID.

1. Set $\hat{h}$ to the identity filter, $\hat{x} = y$ and $\lambda = \lambda_0$; choose $\alpha < 1$.
2. repeat
3. $\hat{x} \leftarrow \text{arg min}_x C_\lambda(x, \hat{h})$ \hspace{1cm} \text{update image estimate}
4. $\hat{h} \leftarrow \text{arg min}_h C_\lambda(\hat{x}, h)$ \hspace{1cm} \text{update blur estimate}
5. $\lambda \leftarrow \alpha \lambda$
6. until stopping criterion is satisfied

[Almeida et al, 2010, 2013]

Updating the image estimate

$$\hat{x} \leftarrow \text{arg min}_x \frac{1}{2} \|y - MHx\|^2 + \lambda \Phi(x)$$

Standard image deconvolution, with unknown boundaries; ADMM as above.
Blind Image Deconvolution (BID)

Updating the image estimate

$$\hat{x} \leftarrow \arg \min_{x \in \mathbb{R}^m} \frac{1}{2} \|y - MBx\|^2 + \lambda \sum_{i=1}^{m} (\|F_i x\|_2)^q$$

Template: $\min_{z \in \mathbb{R}^d} \sum_{j=1}^{J} g_j(H^{(j)} z)$

Mapping: $J = m + 1$, $g_i(z) = \|z\|^q_2$, $i = 1, \ldots, m$,

$$H^{(i)} = F_i, \quad i = 1, \ldots, m,$$

$$g_{m+1}(z) = \frac{1}{2} \|Mz - y\|^2_2, \quad H^{(m+1)} = B$$

All the matrices are circulant: matrix inversion step in ADMM easy with FFT.

Also possible to compute $\text{prox}_{\tau \cdot \|\cdot\|^q_2}(u) = \arg \min_x \frac{1}{2} \|x - u\|^2_2 + \tau \|x\|^q_2$

for $q \in \{0, \frac{1}{2}, \frac{2}{3}, 1, \frac{4}{3}, \frac{3}{2}, 2\}$
Blind Image Deconvolution (BID)

Algorithm 1: Continuation-based BID.

1. Set $\hat{h}$ to the identity filter, $\hat{x} = y$ and $\lambda = \lambda_0$; choose $\alpha < 1$.
2. repeat
3.   $\hat{x} \leftarrow \arg\min_x C_{\lambda}(x, \hat{h})$ \hspace{1cm} update image estimate
4.   $\hat{h} \leftarrow \arg\min_h C_{\lambda}(\hat{x}, h)$, \hspace{1cm} update blur estimate
5.   $\lambda \leftarrow \alpha \lambda$
6. until stopping criterion is satisfied

Updating the blur estimate: notice that $h \ast x = Hx = Xh$

$$\hat{h} \leftarrow \arg\min_h \frac{1}{2} \| y - MXh \|^2 + \iota_{S^+}(h)$$

Like standard image deconvolution, with a support and positivity constraint.

Prox of support and positivity constraint is trivial: $\text{prox}_{\iota_{S^+}}(h) = \Pi_{S^+}(h)$
Blind Image Deconvolution (BID)

Algorithm 1: Continuation-based BID.

1. Set $\hat{h}$ to the identity filter, $\hat{x} = y$ and $\lambda = \lambda_0$; choose $\alpha < 1$.
2. repeat
3. \[ \hat{x} \leftarrow \arg \min_x C_\lambda(x, \hat{h}) \]
4. \[ \hat{h} \leftarrow \arg \min_h C_\lambda(\hat{x}, h) \]
5. \[ \lambda \leftarrow \alpha \lambda \]
6. until stopping criterion is satisfied

Question: when to stop? What value of $\lambda$ to choose?

For non-blind deconvolution, many approaches for choosing $\lambda$

generalized cross validation, L-curve, SURE and variants thereof

[Thomson, Brown, Kay, Titterington, 92], [Hansen, O'Leary, 93], [Eldar, 09], [Giryes, Elad, Eldar 11],
[Luisier, Blu, Unser 09], [Ramani, Blu, Unser, 10], [Ramani, Liu,Rosen, Nielsen, Fessler, 12]

Bayesian methods (some for BID)

[Babacan, Molina, Katsaggelos, 09], [Fergus et al, 06], [Amizic, Babacan, Molina, Katsaggelos, 10],
[Chantas, Galatsanos, Molina, Katsaggelos, 10], [Oliveira, Bioucas-Dias, F, 09]

No-reference quality measures

[Lee, Lai, Chen, 07], [Zhu, Milanfar, 10]
Blind Image Deconvolution: Stopping Criterion

Proposed rationale: if the blur kernel is well estimated, the residual is white.

Autocorrelation:

\[
\begin{align*}
R_{rr} & \\
\end{align*}
\]

Estimated residual

Whiteness:

\[
\begin{align*}
M^l_R & \\
\end{align*}
\]

\[
\begin{align*}
\text{ISNR} & \\
\end{align*}
\]
Blind Image Deconvolution (BID)

Experiment with real motion blurred photo

Blurred photo

[14], 70 seconds

[16], 100 seconds

Proposed method, 55 seconds

[Krishnan et al, 2011]

[Levin et al, 2011]
Blind Image Deconvolution (BID)

Experiment with real out-of-focus photo

- Observed photo.
- [Almeida et al, 2010]
- proposed
Blind Image Deconvolution (BID): Synthetic Results

Realistic motion blurs:
[Levin, Weiss, Durant, Freeman, 09]

Images: Lena, Cameraman

Average results over 2 images and 8 blurs:

<table>
<thead>
<tr>
<th>ISNR* (dB)</th>
<th>Method</th>
<th>0dB</th>
<th>40dB</th>
<th>30dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>[31]</td>
<td>6.14</td>
<td>5.90</td>
<td>4.91</td>
<td></td>
</tr>
<tr>
<td>[35]</td>
<td>5.51</td>
<td>5.72</td>
<td>4.79</td>
<td></td>
</tr>
<tr>
<td>[50]</td>
<td>4.70</td>
<td>4.70</td>
<td>4.30</td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td>9.00</td>
<td>8.43</td>
<td>6.70</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Time (s)</th>
<th>Method</th>
<th>0dB</th>
<th>40dB</th>
<th>30dB</th>
</tr>
</thead>
<tbody>
<tr>
<td>[31]</td>
<td>80</td>
<td>66</td>
<td>62</td>
<td></td>
</tr>
<tr>
<td>[35]</td>
<td>399</td>
<td>399</td>
<td>399</td>
<td></td>
</tr>
<tr>
<td>[50]</td>
<td>$1.5^2$</td>
<td>$1.5^2$</td>
<td>$1.5^2$</td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td>70</td>
<td>55</td>
<td>45</td>
<td></td>
</tr>
</tbody>
</table>

[Levin et al, 11]  
[Xu, Jia, 10]  
[Krishnan et al, 11]  
[Krishnan et al, 11]  
[Xu, Jia, 10] (GPU)
Blind Image Deconvolution (BID): Handling Saturations

Several digital images have saturated pixels (at 0 or max): this impacts BID!

Easy to handle in our approach: just mask them out

\[
\min(\alpha x \ast h, 255)
\]

ignoring saturations

knowing saturations

out-of-focus (disk) blur
Summary:

- Alternating direction optimization (ADMM) is powerful, versatile, modular.
- Main hurdle: need to solve a linear system (invert a matrix) at each iteration…
- …however, sometimes this turns out to be an advantage.
- State of the art results in several image/signal reconstruction problems.