Flexible Krylov Methods for ℓ_p Regularization

Silvia Gazzola

Joint work with **Julianne Chung** and **Malena Sabaté Landman**

Department of Mathematical Sciences

UNIVERSITY OF BATH

Modern Challenges in Imaging
Tufts University, August 6, 2019

What is this talk about?

What is this talk about?

Regularization of linear inverse problems

$$\mathbf{A}\mathbf{x}_{\mathrm{true}} + \boldsymbol{\epsilon} = \mathbf{b}\,,$$

where

$\mathbf{b} \in \mathbb{R}^M$	observations or measurements
$\mathbf{x}_{\mathrm{true}} \in \mathbb{R}^N$	desired parameters
$\mathbf{A} \in \mathbb{R}^{M\times N}$	ill-conditioned matrix models forward process
$\boldsymbol{\epsilon} \in \mathbb{R}^M$	additive Gaussian noise

What is this talk about?

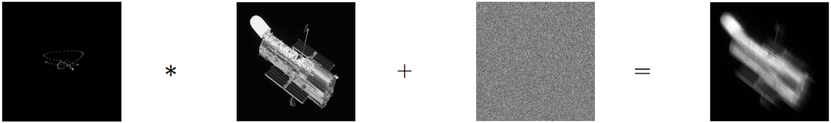
Regularization of linear inverse problems

$$\mathbf{A}\mathbf{x}_{\mathrm{true}} + \boldsymbol{\epsilon} = \mathbf{b}\,,$$

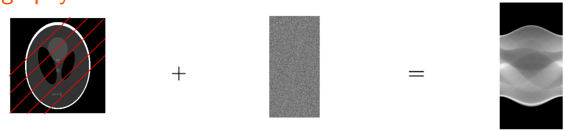
where

$\mathbf{b} \in \mathbb{R}^M$	observations or measurements
$\mathbf{x}_{\mathrm{true}} \in \mathbb{R}^N$	desired parameters
$\mathbf{A} \in \mathbb{R}^{M \times N}$	ill-conditioned matrix models forward process
$\boldsymbol{\epsilon} \in \mathbb{R}^M$	additive Gaussian noise

image deblurring and denoising



computed tomography



Outline

Applying variational regularization...

$$\mathbf{x}^{\mathrm{reg}} = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\mathcal{R}(\mathbf{x}), \quad \lambda > 0$$

Applying variational regularization...

$$\mathbf{x}^{\text{reg}} = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\mathcal{R}(\mathbf{x}), \quad \lambda > 0$$

- $\mathcal{R}(\mathbf{x}) = \|\mathbf{L}\mathbf{x}\|_2^2$

Applying variational regularization...

$$\mathbf{x}^{\text{reg}} = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\mathcal{R}(\mathbf{x}), \quad \lambda > 0$$

- $\mathcal{R}(\mathbf{x}) = \|\mathbf{L}\mathbf{x}\|_2^2$

Krylov methods are popular in this setting

Applying variational regularization...

$$\mathbf{x}^{\mathrm{reg}} = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\mathcal{R}(\mathbf{x}), \quad \lambda > 0$$

- $\mathcal{R}(\mathbf{x}) = \|\mathbf{L}\mathbf{x}\|_2^2$

Krylov methods are popular in this setting

$$\begin{aligned} \mathbf{x}_k \in \mathcal{K}_k(\mathbf{C},\mathbf{d}) &= \mathrm{span}\{\mathbf{d}, \mathbf{C}\mathbf{d}, \dots, \mathbf{C}^{k-1}\mathbf{d}\} \\ \mathbf{r}_k &= \mathbf{b}-\mathbf{A}\mathbf{x}_k \perp \mathcal{K}_k(\mathbf{C}',\mathbf{d}') \end{aligned}$$

Applying variational regularization...

$$\mathbf{x}^{\mathrm{reg}} = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\mathcal{R}(\mathbf{x}), \quad \lambda > 0$$

- $\mathcal{R}(\mathbf{x}) = \|\mathbf{L}\mathbf{x}\|_2^2$

Krylov methods are popular in this setting

$$\begin{aligned} \mathbf{x}_k \in \mathcal{K}_k(\mathbf{C},\mathbf{d}) &= \mathrm{span}\{\mathbf{d}, \mathbf{C}\mathbf{d}, \dots, \mathbf{C}^{k-1}\mathbf{d}\} \\ \mathbf{r}_k &= \mathbf{b} - \mathbf{A}\mathbf{x}_k \perp \mathcal{K}_k(\mathbf{C}', \mathbf{d}') \end{aligned}$$

$$\mathbf{A}\mathbf{Z}_k = \mathbf{W}_{k+1}\mathbf{G}_k$$

$$\mathbf{x}_k = \mathbf{Z}_k\mathbf{y}_k, \quad \mathbf{y}_k = \arg\min_{\mathbf{y}\in\mathbb{R}^k} \|\mathbf{g}_k - \mathbf{G}_k\mathbf{y}\|_2^2$$

Applying variational regularization...

$$\mathbf{x}^{\text{reg}} = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\mathcal{R}(\mathbf{x}), \quad \lambda > 0$$

- $\mathcal{R}(\mathbf{x}) = \|\mathbf{L}\mathbf{x}\|_2^2$

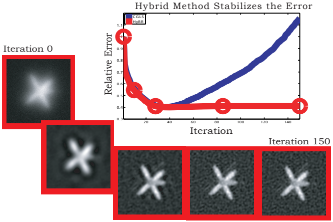
Krylov methods are popular in this setting

$$\begin{aligned} \mathbf{x}_k \in \mathcal{K}_k(\mathbf{C},\mathbf{d}) &= \text{span}\{\mathbf{d}, \mathbf{C}\mathbf{d}, \dots, \mathbf{C}^{k-1}\mathbf{d}\} \\ \mathbf{r}_k &= \mathbf{b} - \mathbf{A}\mathbf{x}_k \perp \mathcal{K}_k(\mathbf{C}', \mathbf{d}') \end{aligned}$$

$$\mathbf{A}\mathbf{Z}_k = \mathbf{W}_{k+1}\mathbf{G}_k$$

$$\mathbf{x}_k = \mathbf{Z}_k\mathbf{y}_k, \quad \mathbf{y}_k = \arg\min_{\mathbf{y}\in\mathbb{R}^k} \|\mathbf{g}_k - \mathbf{G}_k\mathbf{y}\|_2^2$$

Fast semi-convergence



Applying variational regularization...

$$\mathbf{x}^{\text{reg}} = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\mathcal{R}(\mathbf{x}), \quad \lambda > 0$$

- $\mathcal{R}(\mathbf{x}) = \|\mathbf{L}\mathbf{x}\|_2^2$

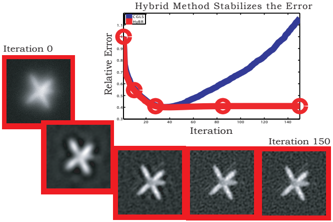
Krylov methods are popular in this setting

$$\begin{aligned} \mathbf{x}_k \in \mathcal{K}_k(\mathbf{C},\mathbf{d}) &= \text{span}\{\mathbf{d}, \mathbf{C}\mathbf{d}, \dots, \mathbf{C}^{k-1}\mathbf{d}\} \\ \mathbf{r}_k &= \mathbf{b} - \mathbf{A}\mathbf{x}_k \perp \mathcal{K}_k(\mathbf{C}', \mathbf{d}') \end{aligned}$$

$$\mathbf{A}\mathbf{Z}_k = \mathbf{W}_{k+1}\mathbf{G}_k$$

$$\mathbf{x}_k = \mathbf{Z}_k\mathbf{y}_k, \quad \mathbf{y}_k = \arg\min_{\mathbf{y}\in\mathbb{R}^k} \|\mathbf{g}_k - \mathbf{G}_k\mathbf{y}\|_2^2 + \lambda_k\|\mathbf{L}_k\mathbf{y}\|_2^2$$

Fast semi-convergence



Applying variational regularization...

$$\mathbf{x}^{\text{reg}} = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\mathcal{R}(\mathbf{x}), \quad \lambda > 0$$

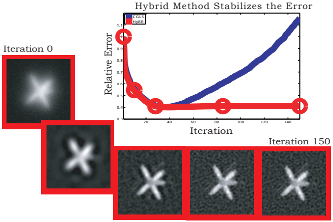
- $\mathcal{R}(\mathbf{x}) = \|\mathbf{L}\mathbf{x}\|_2^2$

Krylov methods are popular in this setting

$$\begin{aligned} \mathbf{x}_k \in \mathcal{K}_k(\mathbf{C},\mathbf{d}) &= \text{span}\{\mathbf{d}, \mathbf{C}\mathbf{d}, \dots, \mathbf{C}^{k-1}\mathbf{d}\} \\ \mathbf{r}_k &= \mathbf{b} - \mathbf{A}\mathbf{x}_k \perp \mathcal{K}_k(\mathbf{C}', \mathbf{d}') \end{aligned}$$

$$\mathbf{A}\mathbf{Z}_k = \mathbf{W}_{k+1}\mathbf{G}_k$$

$$\mathbf{x}_k = \mathbf{Z}_k\mathbf{y}_k, \quad \mathbf{y}_k = \arg\min_{\mathbf{y}\in\mathbb{R}^k} \|\mathbf{g}_k - \mathbf{G}_k\mathbf{y}\|_2^2 + \lambda_k\|\mathbf{L}_k\mathbf{y}\|_2^2$$



Fast semi-convergence

Hanke (1995); Frommer and Maas (1999);
O'Leary and Simmons (1981); Calvetti, Morigi, Reichel, Sgallari (2000);
Kilmer, Hansen, Espanol (2007); Chung and Palmer (2015); G., Novati, Russo (2015)
...

Applying variational regularization...

$$\mathbf{x}^{\mathrm{reg}} = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\mathcal{R}(\mathbf{x}), \quad \lambda > 0$$

Applying variational regularization...

$$\mathbf{x}^{\mathrm{reg}} = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\mathcal{R}(\mathbf{x}), \quad \lambda > 0$$

- ℓ^p regularization
  (we will consider $\mathcal{R}(\mathbf{x}) = \|\mathbf{x}\|_p^p$, $\mathcal{R}(\mathbf{x}) = \|\mathbf{\Psi}\mathbf{x}\|_p^p$, $\mathcal{R}(\mathbf{x}) = \mathrm{TV}_p(\mathbf{x})$; $p \geq 1$, $p > 0$)

Applying variational regularization...

$$\mathbf{x}^{\text{reg}} = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\mathcal{R}(\mathbf{x}), \quad \lambda > 0$$

- ℓ^p regularization
  (we will consider $\mathcal{R}(\mathbf{x}) = \|\mathbf{x}\|_p^p$, $\mathcal{R}(\mathbf{x}) = \|\mathbf{\Psi x}\|_p^p$, $\mathcal{R}(\mathbf{x}) = \text{TV}_p(\mathbf{x})$; $p \geq 1$, $p > 0$)
  - Sub-gradient strategies
    Shevade and Keerthi (2003), Perkins (2003), Andrew and Gao (2007), ...
  - Constrained optimization
    Chen et al (1999), Bertsekas (2004), Gafni and Bertsekas (1984), ...
  - Iterative shinkage-thresholding algorithms (ISTA)
    Bioucas-Dias and Figueiredo(2007), Giryes, Elad and Eldar (2011), Beck and Teboulle (2009), Goldstein and Osher (2009), Osher et al (2005) ....
  - Iteratively re-weighted norm
    Rodriguez and Wohlberg (2008), Renaut et al (2017), ...
  - Generalized Krylov methods for $\ell_p - \ell_q$
    Lanza et al (2015), Huang, Lanza, Morigi, Reichel, Sgallari (2017), Buccini and Reichel (2019),...
  - Flexible Arnoldi methods (for square problems)
    Gazzola and Nagy (2014)
  - ...

Applying variational regularization...

$$\mathbf{x}^{\mathrm{reg}} = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\mathcal{R}(\mathbf{x}), \quad \lambda > 0$$

- ℓ^p regularization
  (we will consider $\mathcal{R}(\mathbf{x}) = \|\mathbf{x}\|_p^p$, $\mathcal{R}(\mathbf{x}) = \|\mathbf{\Psi}\mathbf{x}\|_p^p$, $\mathcal{R}(\mathbf{x}) = \mathrm{TV}_p(\mathbf{x})$; $p \geq 1$, $p > 0$)
  - Sub-gradient strategies
    Shevade and Keerthi (2003), Perkins (2003), Andrew and Gao (2007), ...
  - Constrained optimization
    Chen et al (1999), Bertsekas (2004), Gafni and Bertsekas (1984), ...
  - Iterative shinkage-thresholding algorithms (ISTA)
    Bioucas-Dias and Figueiredo(2007), Giryes, Elad and Eldar (2011), Beck and Teboulle (2009), Goldstein and Osher (2009), Osher et al (2005) ....
  - **Iteratively re-weighted norm**
    Rodriguez and Wohlberg (2008), Renaut et al (2017), ...
  - Generalized Krylov methods for $\ell_p - \ell_q$
    Lanza et al (2015), Huang, Lanza, Morigi, Reichel, Sgallari (2017), Buccini and Reichel (2019),...
  - **Flexible Arnoldi methods (for square problems)**
    Gazzola and Nagy (2014)
  - ...

A basic Iteratively Re-weighted Norm (IRN) strategy ...

Let $\mathbf{\Psi} = \mathbf{I}$, $p = 1$.

A basic Iteratively Re-weighted Norm (IRN) strategy ...

Let $\boldsymbol{\Psi} = \mathbf{I}$, $p = 1$. Turn ℓ_1-problems into a sequence of ℓ_2-problems:

A basic Iteratively Re-weighted Norm (IRN) strategy ...

Let $\boldsymbol{\Psi} = \mathbf{I}$, $p = 1$. Turn ℓ_1-problems into a sequence of ℓ_2-problems:

$$\|\mathbf{x}\|_1 \approx \|\mathbf{L}(\mathbf{x})\mathbf{x}\|_2^2$$

A basic Iteratively Re-weighted Norm (IRN) strategy ...

[Rodriguez and Wohlberg (2008)]

Let $\boldsymbol{\Psi} = \mathbf{I}, p = 1$. Turn ℓ_1-problems into a sequence of ℓ_2-problems:

$$\|\mathbf{x}\|_1 \approx \|\mathbf{L}(\mathbf{x})\mathbf{x}\|_2^2$$

IRN algorithm

Input: $\mathbf{A}$, $\mathbf{b}$, $\mathbf{x}_0 (= 0)$, $\mathbf{L}_0 = \mathbf{L}(\mathbf{x}_0) (= \mathbf{I})$

- For $k = 1, \dots,$ till a stopping criterion is satisfied

$$\mathbf{x}_k = \arg\min_{\mathbf{x} \in \mathbb{R}^N} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda \|\mathbf{L}_{k-1}\mathbf{x}\|_2^2$$

- Update $\mathbf{L}_k = \mathrm{diag}\left(1/\sqrt{f_\tau(|\mathbf{x}_k|)}\right)$, $f_\tau(|[\mathbf{x}_k]_i|) = \begin{cases} |[\mathbf{x}_k]_i| & \text{if } |[\mathbf{x}_k]_i| \geq \tau_1 \\ \tau_2 & \text{if } |[\mathbf{x}_k]_i| < \tau_1 \end{cases}$

A basic Iteratively Re-weighted Norm (IRN) strategy ...

[Rodriguez and Wohlberg (2008)]

Let $\mathbf{\Psi} = \mathbf{I}$, $p = 1$. Turn ℓ_1-problems into a sequence of ℓ_2-problems:

$$\|\mathbf{x}\|_1 \approx \|\mathbf{L}(\mathbf{x})\mathbf{x}\|_2^2$$

IRN algorithm

Input: $\mathbf{A}$, $\mathbf{b}$, $\mathbf{x}_0 (= 0)$, $\mathbf{L}_0 = \mathbf{L}(\mathbf{x}_0) (= \mathbf{I})$

- For $k = 1, \dots,$ till a stopping criterion is satisfied
  - Till a stopping criterion is satisfied: run an (iterative) solver for

$$\mathbf{x}_k = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda\|\mathbf{L}_{k-1}\mathbf{x}\|_2^2$$

- Update $\mathbf{L}_k = \mathrm{diag}\left(1/\sqrt{f_\tau(|\mathbf{x}_k|)}\right)$, $f_\tau(|[\mathbf{x}_k]_i|) = \begin{cases} |[\mathbf{x}_k]_i| & \text{if } |[\mathbf{x}_k]_i| \geq \tau_1 \\ \tau_2 & \text{if } |[\mathbf{x}_k]_i| < \tau_1 \end{cases}$

A basic Iteratively Re-weighted Norm (IRN) strategy ...

[Rodriguez and Wohlberg (2008)]

Let $\mathbf{\Psi} = \mathbf{I}, p = 1$. Turn ℓ_1-problems into a sequence of ℓ_2-problems:

$$\|\mathbf{x}\|_1 \approx \|\mathbf{L}(\mathbf{x})\mathbf{x}\|_2^2$$

IRN algorithm

Input: $\mathbf{A}$, $\mathbf{b}$, $\mathbf{x}_0(=0)$, $\mathbf{L}_0 = \mathbf{L}(\mathbf{x}_0)(=\mathbf{I})$

- For $k = 1, \dots,$ till a stopping criterion is satisfied
  - Till a stopping criterion is satisfied: run an (iterative) solver for

$$\mathbf{x}_k = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda\|\mathbf{L}_{k-1}\mathbf{x}\|_2^2$$

- Update $\mathbf{L}_k = \mathrm{diag}\left(1/\sqrt{f_\tau(|\mathbf{x}_k|)}\right)$, $f_\tau(|[\mathbf{x}_k]_i|) = \begin{cases} |[\mathbf{x}_k]_i| & \text{if } |[\mathbf{x}_k]_i| \geq \tau_1 \\ \tau_2 & \text{if } |[\mathbf{x}_k]_i| < \tau_1 \end{cases}$

Let $\mathbf{L}_k = \mathbf{L}(\mathbf{x}_k)$, then $\mathbf{x}_{k+1} = \mathbf{L}_k^{-1}\mathbf{y}_{k+1}$ where

$$\mathbf{y}_{k+1} = \arg\min_{\mathbf{y}} \left\|\mathbf{A}\mathbf{L}_k^{-1}\mathbf{y} - \mathbf{b}\right\|_2^2 + \lambda\,\|\mathbf{y}\|_2^2$$

A basic Iteratively Re-weighted Norm (IRN) strategy ...

[Rodriguez and Wohlberg (2008)]

Let $\boldsymbol{\Psi} = \mathbf{I}, p = 1$. Turn ℓ_1-problems into a sequence of ℓ_2-problems:

$$\|\mathbf{x}\|_1 \approx \|\mathbf{L}(\mathbf{x})\mathbf{x}\|_2^2$$

IRN algorithm

Input: $\mathbf{A}$, $\mathbf{b}$, $\mathbf{x}_0 (= 0)$, $\mathbf{L}_0 = \mathbf{L}(\mathbf{x}_0) (= \mathbf{I})$

- For $k = 1, \dots,$ till a stopping criterion is satisfied
  - Till a stopping criterion is satisfied: run an (iterative) solver for

$$\mathbf{x}_k = \arg\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda\|\mathbf{L}_{k-1}\mathbf{x}\|_2^2$$

- Update $\mathbf{L}_k = \mathrm{diag}\left(1/\sqrt{f_\tau(|\mathbf{x}_k|)}\right)$, $f_\tau(|[\mathbf{x}_k]_i|) = \begin{cases} |[\mathbf{x}_k]_i| & \text{if } |[\mathbf{x}_k]_i| \geq \tau_1 \\ \tau_2 & \text{if } |[\mathbf{x}_k]_i| < \tau_1 \end{cases}$

Let $\mathbf{L}_k = \mathbf{L}(\mathbf{x}_k)$, then $\mathbf{x}_{k+1} = \mathbf{L}_k^{-1}\mathbf{y}_{k+1}$ where

$$\mathbf{y}_{k+1} = \arg\min_{\mathbf{y}} \left\|\mathbf{A}\mathbf{L}_k^{-1}\mathbf{y} - \mathbf{b}\right\|_2^2 + \lambda\,\|\mathbf{y}\|_2^2$$

... revisited within Flexible Krylov methods ...

[G. and Nagy (2014)]

1. For $\mathbf{A} \in \mathbb{R}^{N \times N}$, use *flexible* Arnoldi to generate basis vectors: [Saad (1993, 2003)]

$$\mathbf{Z}_k = \begin{bmatrix} \mathbf{L}_1^{-1}\mathbf{v}_1 & \cdots & \mathbf{L}_k^{-1}\mathbf{v}_k \end{bmatrix} \in \mathbb{R}^{N \times k}$$

where

$$\mathbf{A}\mathbf{Z}_k = \mathbf{V}_{k+1}\mathbf{H}_k$$

- $\mathbf{V}_{k+1} = \begin{bmatrix} \mathbf{v}_1 & \ldots & \mathbf{v}_{k+1} \end{bmatrix} \in \mathbb{R}^{N \times (k+1)}$ has orthonormal columns (ONC)
- $\mathbf{H}_k \in \mathbb{R}^{(k+1) \times k}$ is upper Hessenberg

... revisited within Flexible Krylov methods ...

[G. and Nagy (2014)]

1. For $\mathbf{A} \in \mathbb{R}^{N \times N}$, use *flexible* Arnoldi to generate basis vectors:
   [Saad (1993, 2003)]

$$\mathbf{Z}_k = \begin{bmatrix} \mathbf{L}_1^{-1}\mathbf{v}_1 & \cdots & \mathbf{L}_k^{-1}\mathbf{v}_k \end{bmatrix} \in \mathbb{R}^{N \times k}$$

where

$$\mathbf{A}\mathbf{Z}_k = \mathbf{V}_{k+1}\mathbf{H}_k$$

- $\mathbf{V}_{k+1} = \begin{bmatrix} \mathbf{v}_1 & \dots & \mathbf{v}_{k+1} \end{bmatrix} \in \mathbb{R}^{N \times (k+1)}$ has orthonormal columns (ONC)
- $\mathbf{H}_k \in \mathbb{R}^{(k+1) \times k}$ is upper Hessenberg

2. Compute solution $\quad \mathbf{x}_k = \mathbf{x}_0 + \mathbf{Z}_k \mathbf{y}_k \quad$ where

$$\mathbf{y}_k = \arg\min_{\mathbf{y}} \frac{1}{2} \left\| \mathbf{H}_k \mathbf{y} - \|\mathbf{r}_0\|_2 \, \mathbf{e}_1 \right\|_2^2 + \lambda \|\mathbf{y}\|_2^2$$

Flexible Golub-Kahan (FGK) Process

[Chung and G. (2018)]

A new flexible factorization

Flexible Golub-Kahan (FGK) Process

[Chung and G. (2018)]

Given $\mathbf{A} \in \mathbb{R}^{M\times N}, \mathbf{b} \in \mathbb{R}^{M}$, initialize $\mathbf{u}_1 = \mathbf{b}/\beta_1$ where $\beta_1 = \|\mathbf{b}\|$.

After k iterations with changing preconditioners $\mathbf{L}_k$, we have

Related to inexact Krylov methods [Simoncini and Szyld (2007)]

- $\mathbf{Z}_k = \begin{bmatrix} \mathbf{L}_1^{-1}\mathbf{v}_1 & \cdots & \mathbf{L}_k^{-1}\mathbf{v}_k \end{bmatrix} \in \mathbb{R}^{N\times k}$
- $\mathbf{M}_k \in \mathbb{R}^{(k+1)\times k}$ upper Hessenberg
- $\mathbf{T}_k \in \mathbb{R}^{k\times k}$ upper triangular
- $\mathbf{U}_{k+1} = \begin{bmatrix} \mathbf{u}_1 & \ldots & \mathbf{u}_{k+1} \end{bmatrix} \in \mathbb{R}^{M\times(k+1)}$ ONC
- $\mathbf{V}_k = \begin{bmatrix} \mathbf{v}_1 & \ldots & \mathbf{v}_k \end{bmatrix} \in \mathbb{R}^{N\times k}$ ONC

such that

$$\mathbf{A}\mathbf{Z}_k = \mathbf{U}_{k+1}\mathbf{M}_k \quad \text{and} \quad \mathbf{A}^\top\mathbf{U}_{k+1} = \mathbf{V}_{k+1}\mathbf{T}_{k+1}$$

Flexible Golub-Kahan (FGK) Process

[Chung and G. (2018)]

Given $\mathbf{A} \in \mathbb{R}^{M\times N}, \mathbf{b} \in \mathbb{R}^{M}$, initialize $\mathbf{u}_1 = \mathbf{b}/\beta_1$ where $\beta_1 = \|\mathbf{b}\|$.

After k iterations with changing preconditioners $\mathbf{L}_k$, we have

Related to inexact Krylov methods [Simoncini and Szyld (2007)]

- $\mathbf{Z}_k = \begin{bmatrix} \mathbf{L}_1^{-1}\mathbf{v}_1 & \cdots & \mathbf{L}_k^{-1}\mathbf{v}_k \end{bmatrix} \in \mathbb{R}^{N\times k}$
- $\mathbf{M}_k \in \mathbb{R}^{(k+1)\times k}$ upper Hessenberg
- $\mathbf{T}_k \in \mathbb{R}^{k\times k}$ upper triangular
- $\mathbf{U}_{k+1} = \begin{bmatrix} \mathbf{u}_1 & \ldots & \mathbf{u}_{k+1} \end{bmatrix} \in \mathbb{R}^{M\times (k+1)}$ ONC
- $\mathbf{V}_k = \begin{bmatrix} \mathbf{v}_1 & \ldots & \mathbf{v}_k \end{bmatrix} \in \mathbb{R}^{N\times k}$ ONC

such that

$$\mathbf{A}\mathbf{Z}_k = \mathbf{U}_{k+1}\mathbf{M}_k \quad \text{and} \quad \mathbf{A}^\top \mathbf{U}_{k+1} = \mathbf{V}_{k+1}\mathbf{T}_{k+1}$$

Remarks:

- If $\mathbf{L}_k = \mathbf{L}$, get right-preconditioned GK bidiagonalization
- Additional orthogonalizations and storage

Flexible LSQR and flexible LSMR

New flexible solvers

Flexible LSQR and flexible LSMR

1. Use *flexible* GK to generate basis vectors:

$$\mathbf{Z}_k = \begin{bmatrix} \mathbf{L}_1^{-1}\mathbf{v}_1 & \cdots & \mathbf{L}_k^{-1}\mathbf{v}_k \end{bmatrix} \in \mathbb{R}^{n\times k}$$

$$\mathbf{A}\mathbf{Z}_k = \mathbf{U}_{k+1}\mathbf{M}_k \quad \text{and} \quad \mathbf{A}^\top\mathbf{U}_{k+1} = \mathbf{V}_{k+1}\mathbf{T}_{k+1}$$

Flexible LSQR and flexible LSMR

1. Use *flexible* GK to generate basis vectors:

$$\mathbf{Z}_k = \begin{bmatrix} \mathbf{L}_1^{-1}\mathbf{v}_1 & \cdots & \mathbf{L}_k^{-1}\mathbf{v}_k \end{bmatrix} \in \mathbb{R}^{n\times k}$$

$$\mathbf{A}\mathbf{Z}_k = \mathbf{U}_{k+1}\mathbf{M}_k \quad \text{and} \quad \mathbf{A}^\top\mathbf{U}_{k+1} = \mathbf{V}_{k+1}\mathbf{T}_{k+1}$$

2. Compute solution $\mathbf{x}_k = \mathbf{Z}_k\mathbf{y}_k$ where
   - Flexible LSQR (FLSQR)

$$\mathbf{y}_k = \arg\min_{\mathbf{y}\in\mathbb{R}^k} \|\mathbf{M}_k\mathbf{y} - \beta_1\mathbf{e}_1\|_2^2$$

   - Flexible LSMR (FLSMR)

$$\mathbf{y}_k = \arg\min_{\mathbf{y}\in\mathbb{R}^k} \|\mathbf{T}_{k+1}\mathbf{M}_k\mathbf{y} - \beta_1 m_{11}\mathbf{e}_1\|_2^2$$

Flexible LSQR and flexible LSMR

1. Use *flexible* GK to generate basis vectors:

$$\mathbf{Z}_k = \begin{bmatrix} \mathbf{L}_1^{-1}\mathbf{v}_1 & \cdots & \mathbf{L}_k^{-1}\mathbf{v}_k \end{bmatrix} \in \mathbb{R}^{n\times k}$$

$$\mathbf{A}\mathbf{Z}_k = \mathbf{U}_{k+1}\mathbf{M}_k \quad \text{and} \quad \mathbf{A}^\top \mathbf{U}_{k+1} = \mathbf{V}_{k+1}\mathbf{T}_{k+1}$$

2. Compute solution $\mathbf{x}_k = \mathbf{Z}_k\mathbf{y}_k$ where
   - Flexible LSQR (FLSQR)

$$\mathbf{y}_k = \arg\min_{\mathbf{y}\in\mathbb{R}^k} \|\mathbf{M}_k\mathbf{y} - \beta_1\mathbf{e}_1\|_2^2$$

     Optimality property:

     $\mathbf{x}_k$ minimizes $\|\mathbf{A}\mathbf{x}_k - \mathbf{b}\|_2$ over $\mathbf{x}_0 + \mathrm{span}\{\mathbf{Z}_k\}$.

   - Flexible LSMR (FLSMR)

$$\mathbf{y}_k = \arg\min_{\mathbf{y}\in\mathbb{R}^k} \|\mathbf{T}_{k+1}\mathbf{M}_k\mathbf{y} - \beta_1 m_{11}\mathbf{e}_1\|_2^2$$

Flexible LSQR and flexible LSMR

1. Use *flexible* GK to generate basis vectors:

$$\mathbf{Z}_k = \begin{bmatrix} \mathbf{L}_1^{-1}\mathbf{v}_1 & \cdots & \mathbf{L}_k^{-1}\mathbf{v}_k \end{bmatrix} \in \mathbb{R}^{n\times k}$$

$$\mathbf{A}\mathbf{Z}_k = \mathbf{U}_{k+1}\mathbf{M}_k \quad \text{and} \quad \mathbf{A}^\top \mathbf{U}_{k+1} = \mathbf{V}_{k+1}\mathbf{T}_{k+1}$$

2. Compute solution $\mathbf{x}_k = \mathbf{Z}_k\mathbf{y}_k$ where
   - Flexible LSQR (FLSQR)

$$\mathbf{y}_k = \arg\min_{\mathbf{y}\in\mathbb{R}^k} \|\mathbf{M}_k\mathbf{y} - \beta_1\mathbf{e}_1\|_2^2$$

Optimality property:

$\mathbf{x}_k$ minimizes $\|\mathbf{A}\mathbf{x}_k - \mathbf{b}\|_2$ over $\mathbf{x}_0 + \mathrm{span}\{\mathbf{Z}_k\}$.

   - Flexible LSMR (FLSMR)

$$\mathbf{y}_k = \arg\min_{\mathbf{y}\in\mathbb{R}^k} \|\mathbf{T}_{k+1}\mathbf{M}_k\mathbf{y} - \beta_1 m_{11}\mathbf{e}_1\|_2^2$$

Optimality property:
$\mathbf{x}_k$ minimizes $\left\|\mathbf{A}^\top(\mathbf{A}\mathbf{x}_k - \mathbf{b})\right\|_2$ over $\mathbf{x}_0 + \mathrm{span}\{\mathbf{Z}_k\}$.
Equivalency result:
FLSMR is equivalent to FGMRES applied to the normal equations.

Flexible GK (FGK) *hybrid* methods

New flexible solvers
used in a hybrid framework

Flexible GK (FGK) *hybrid* methods

1. Use *flexible* GK to generate basis vectors:

$$\mathbf{Z}_k = \left[\mathbf{L}_1^{-1}\mathbf{v}_1 \quad \cdots \quad \mathbf{L}_k^{-1}\mathbf{v}_k\right] \in \mathbb{R}^{N\times k}$$

$$\mathbf{A}\mathbf{Z}_k = \mathbf{U}_{k+1}\mathbf{M}_k \quad \text{and} \quad \mathbf{A}^\top\mathbf{U}_{k+1} = \mathbf{V}_{k+1}\mathbf{T}_{k+1}$$

Flexible GK (FGK) *hybrid* methods

1. Use *flexible* GK to generate basis vectors:

$$\mathbf{Z}_k = \begin{bmatrix} \mathbf{L}_1^{-1}\mathbf{v}_1 & \cdots & \mathbf{L}_k^{-1}\mathbf{v}_k \end{bmatrix} \in \mathbb{R}^{N\times k}$$

$$\mathbf{A}\mathbf{Z}_k = \mathbf{U}_{k+1}\mathbf{M}_k \quad \text{and} \quad \mathbf{A}^\top\mathbf{U}_{k+1} = \mathbf{V}_{k+1}\mathbf{T}_{k+1}$$

2. Compute solution $\mathbf{x}_k = \mathbf{Z}_k\mathbf{y}_k$

Flexible GK (FGK) *hybrid* methods

1. Use *flexible* GK to generate basis vectors:

$$\mathbf{Z}_k = \begin{bmatrix} \mathbf{L}_1^{-1}\mathbf{v}_1 & \cdots & \mathbf{L}_k^{-1}\mathbf{v}_k \end{bmatrix} \in \mathbb{R}^{N\times k}$$

$$\mathbf{A}\mathbf{Z}_k = \mathbf{U}_{k+1}\mathbf{M}_k \quad \text{and} \quad \mathbf{A}^\top \mathbf{U}_{k+1} = \mathbf{V}_{k+1}\mathbf{T}_{k+1}$$

2. Compute solution $\mathbf{x}_k = \mathbf{Z}_k\mathbf{y}_k$, where
   - Flexible GK Tikhonov - R (FLSQR-R)

$$\mathbf{y}_k = \arg\min_{\mathbf{y}\in\mathbb{R}^k} \left\|\mathbf{M}_k\mathbf{y} - \beta_1\mathbf{e}_1\right\|_2^2 + \lambda_k \left\|\mathbf{R}_k\mathbf{y}\right\|_2^2 , \quad \mathbf{Z}_k = \mathbf{Q}_k\mathbf{R}_k$$

Flexible GK (FGK) *hybrid* methods

1. Use *flexible* GK to generate basis vectors:

$$\mathbf{Z}_k = \begin{bmatrix} \mathbf{L}_1^{-1}\mathbf{v}_1 & \cdots & \mathbf{L}_k^{-1}\mathbf{v}_k \end{bmatrix} \in \mathbb{R}^{N \times k}$$

$$\mathbf{A}\mathbf{Z}_k = \mathbf{U}_{k+1}\mathbf{M}_k \quad \text{and} \quad \mathbf{A}^\top \mathbf{U}_{k+1} = \mathbf{V}_{k+1}\mathbf{T}_{k+1}$$

2. Compute solution $\mathbf{x}_k = \mathbf{Z}_k\mathbf{y}_k$, where
   - Flexible GK Tikhonov - R (FLSQR-R)

$$\mathbf{y}_k = \arg\min_{\mathbf{y}\in\mathbb{R}^k} \|\mathbf{M}_k\mathbf{y} - \beta_1\mathbf{e}_1\|_2^2 + \lambda_k \|\mathbf{R}_k\mathbf{y}\|_2^2 \,, \quad \mathbf{Z}_k = \mathbf{Q}_k\mathbf{R}_k$$

   - Flexible GK Tikhonov - I (FLSQR-I)

$$\mathbf{y}_k = \arg\min_{\mathbf{y}\in\mathbb{R}^k} \|\mathbf{M}_k\mathbf{y} - \beta_1\mathbf{e}_1\|_2^2 + \lambda_k \|\mathbf{y}\|_2^2$$

FLSQR-R: Approximate singular values of $\mathbf{A}$

$$\mathbf{R}_k^{-\top}\mathbf{M}_k^{\top}\mathbf{M}_k\mathbf{R}_k^{-1} = \mathbf{R}_k^{-\top}\mathbf{M}_k^{\top}\mathbf{U}_{k+1}^{\top}\mathbf{U}_{k+1}\mathbf{M}_k\mathbf{R}_k^{-1} = \mathbf{Q}_k^{\top}\mathbf{A}^{\top}\mathbf{A}\mathbf{Q}_k$$

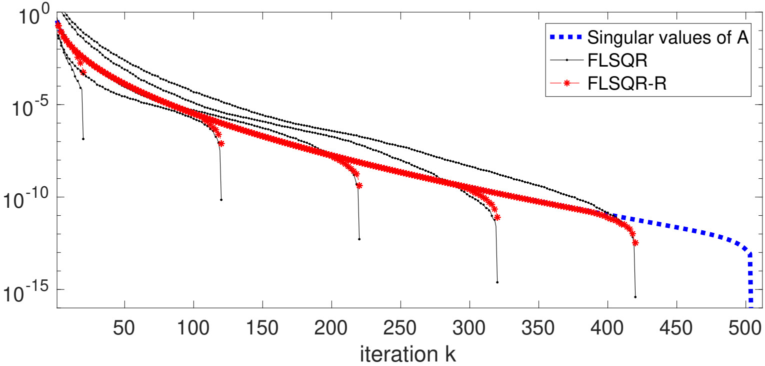


Figure: This plot compares the singular values of $\mathbf{A}$ to the singular values of $\mathbf{M}_k$ from FLSQR and of $\mathbf{M}_k\mathbf{R}_k^{-1}$ from FLSQR-R, for iterations k between 20 and 420 in increments of 100.

Solving the transformed problem

Let $\mathbf{\Psi} \neq \mathbf{I}$ (invertible), $p = 1$ (e.g., $\mathbf{\Psi}$: image domain $\rightarrow$ wavelet domain)

Solving the transformed problem

Let $\mathbf{\Psi} \neq \mathbf{I}$ (invertible), $p = 1$ (e.g., $\mathbf{\Psi}$: image domain $\rightarrow$ wavelet domain)

Equivalent problems (for $\widetilde{\mathbf{\Psi}}$ orthogonal):

[Belge, Kilmer, Miller (2000)]

$$\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda \|\mathbf{\Psi}\mathbf{x}\|_1 \Leftrightarrow \min_{\mathbf{x}\in\mathbb{R}^N} \|\underbrace{\widetilde{\mathbf{\Psi}}\mathbf{A}\mathbf{\Psi}^{-1}}_{\mathbf{H}}\underbrace{\mathbf{\Psi}\mathbf{x}}_{\mathbf{s}} - \underbrace{\widetilde{\mathbf{\Psi}}\mathbf{b}}_{\mathbf{d}}\|_2^2 + \lambda\|\underbrace{\mathbf{\Psi}\mathbf{x}}_{\mathbf{s}}\|_1$$

Solution subspace for flexible Arnoldi:

$$\mathbf{s}_k \in \mathrm{span}\{\mathbf{L}_1^{-1}\widehat{\mathbf{v}}_1, \mathbf{L}_2^{-1}\widehat{\mathbf{v}}_2, \dots, \mathbf{L}_k^{-1}\widehat{\mathbf{v}}_k\}, \quad \text{where} \quad \begin{array}{rcl} \widehat{\mathbf{v}}_1 & = & \mathbf{d}/\|\mathbf{d}\|_2 \\ \widehat{\mathbf{v}}_2 & = & \mathrm{ONC}(\mathbf{H}\mathbf{L}_1^{-1}\widehat{\mathbf{v}}_1) \\ \widehat{\mathbf{v}}_3 & = & \mathrm{ONC}(\mathbf{H}\mathbf{L}_2^{-1}\widehat{\mathbf{v}}_2) \\ \dots & & \end{array}$$

Solving the transformed problem

Let $\boldsymbol{\Psi} \neq \mathbf{I}$ (invertible), $p = 1$ (e.g., $\boldsymbol{\Psi}$: image domain $\rightarrow$ wavelet domain)
Equivalent problems (for $\widetilde{\boldsymbol{\Psi}}$ orthogonal):
[Belge, Kilmer, Miller (2000)]

$$\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\|\boldsymbol{\Psi}\mathbf{x}\|_1 \Leftrightarrow \min_{\mathbf{x}\in\mathbb{R}^N} \|\underbrace{\widetilde{\boldsymbol{\Psi}}\mathbf{A}\boldsymbol{\Psi}^{-1}}_{\mathbf{H}}\underbrace{\boldsymbol{\Psi}\mathbf{x}}_{\mathbf{s}} - \underbrace{\widetilde{\boldsymbol{\Psi}}\mathbf{b}}_{\mathbf{d}}\|_2^2 + \lambda\|\underbrace{\boldsymbol{\Psi}\mathbf{x}}_{\mathbf{s}}\|_1$$

Solution subspace for flexible Arnoldi:

$$\mathbf{s}_k \in \operatorname{span}\{\mathbf{L}_1^{-1}\widehat{\mathbf{v}}_1, \mathbf{L}_2^{-1}\widehat{\mathbf{v}}_2, \dots, \mathbf{L}_k^{-1}\widehat{\mathbf{v}}_k\}, \quad \text{where} \quad \begin{aligned} \widehat{\mathbf{v}}_1 &= \mathbf{d}/\|\mathbf{d}\|_2 \\ \widehat{\mathbf{v}}_2 &= \mathrm{ONC}(\mathbf{H}\mathbf{L}_1^{-1}\widehat{\mathbf{v}}_1) \\ \widehat{\mathbf{v}}_3 &= \mathrm{ONC}(\mathbf{H}\mathbf{L}_2^{-1}\widehat{\mathbf{v}}_2) \\ &\dots \end{aligned}$$

$$\mathbf{x}_k = \boldsymbol{\Psi}^{-1}\mathbf{s}_k$$

Solving the transformed problem

Let $\mathbf{\Psi} \neq \mathbf{I}$ (invertible), $p = 1$ (e.g., $\mathbf{\Psi}$: image domain $\rightarrow$ wavelet domain)
Equivalent problems (for $\widetilde{\mathbf{\Psi}}$ orthogonal):
[Belge, Kilmer, Miller (2000)]

$$\min_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{Ax} - \mathbf{b}\|_2^2 + \lambda \|\mathbf{\Psi x}\|_1 \Leftrightarrow \min_{\mathbf{x}\in\mathbb{R}^N} \|\underbrace{\widetilde{\mathbf{\Psi}}\mathbf{A}\mathbf{\Psi}^{-1}}_{\mathbf{H}}\underbrace{\mathbf{\Psi x}}_{\mathbf{s}} - \underbrace{\widetilde{\mathbf{\Psi}}\mathbf{b}}_{\mathbf{d}}\|_2^2 + \lambda\|\underbrace{\mathbf{\Psi x}}_{\mathbf{s}}\|_1$$

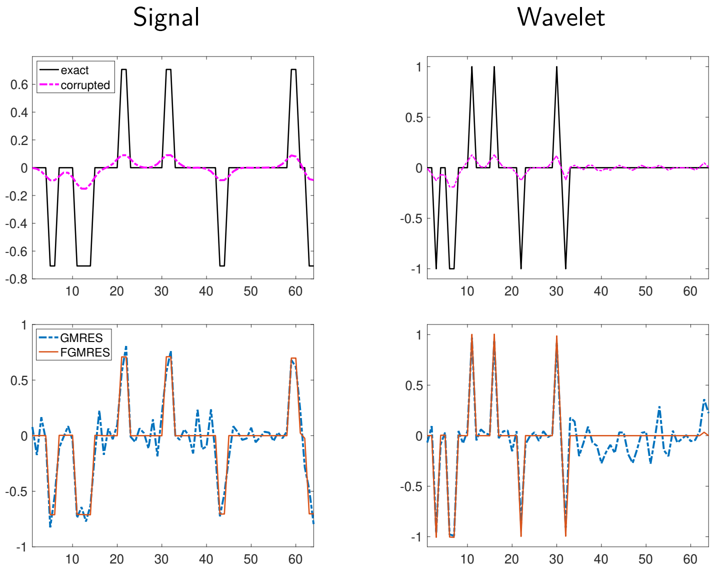
Solution subspace for flexible Arnoldi:

$$\mathbf{s}_k \in \operatorname{span}\{\mathbf{L}_1^{-1}\widehat{\mathbf{v}}_1, \mathbf{L}_2^{-1}\widehat{\mathbf{v}}_2, \dots, \mathbf{L}_k^{-1}\widehat{\mathbf{v}}_k\}, \quad \text{where} \quad \begin{array}{rcl} \widehat{\mathbf{v}}_1 & = & \mathbf{d}/\|\mathbf{d}\|_2 \\ \widehat{\mathbf{v}}_2 & = & \mathrm{ONC}(\mathbf{HL}_1^{-1}\widehat{\mathbf{v}}_1) \\ \widehat{\mathbf{v}}_3 & = & \mathrm{ONC}(\mathbf{HL}_2^{-1}\widehat{\mathbf{v}}_2) \\ \dots & & \end{array}$$

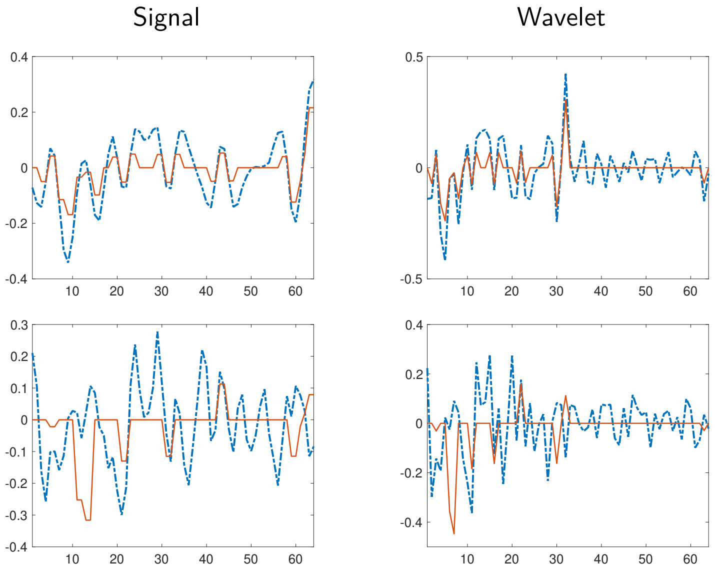
$$\mathbf{x}_k = \mathbf{\Psi}^{-1}\mathbf{s}_k$$

Analogously for flexible Golub-Kahan (possibly without $\widetilde{\mathbf{\Psi}}$).

An illustration: sparsity in a wavelet domain



An illustration: 2^{nd} and 4^{th} basis vectors



TV penalization

Let $\mathcal{R}(\mathbf{x}) = \mathrm{TV}(\mathbf{x})$.

- 1d case:
  $\mathrm{TV}(\mathbf{x}) = \|\mathbf{D}_{1\mathrm{d}}\mathbf{x}\|_1 \simeq \|\mathbf{W}_{1\mathrm{d}}\mathbf{D}\mathbf{x}\|_2^2$, where

$$\mathbf{D}_{1\mathrm{d}} = \begin{bmatrix} 1 & -1 & & \\ & \ddots & \ddots & \\ & & 1 & -1 \end{bmatrix} \in \mathbb{R}^{(N-1)\times N}, \quad \mathbf{W} = \mathrm{diag}\left(|\mathbf{D}_{1\mathrm{d}}\mathbf{x}|^{-1/2}\right)$$

- 2d case:
  [Wohlberg and Rodriguez. An iteratively reweighted norm algorithm for TV. *IEEE*, 2007]
  $\mathrm{TV}(\mathbf{x}) = \|\left((\mathbf{D}^{\mathrm{h}}\mathbf{x})^2 + (\mathbf{D}^{\mathrm{v}}\mathbf{x})^2\right)^{1/2}\|_1 \simeq \|\mathbf{W}\mathbf{D}_{2\mathrm{d}}\mathbf{x}\|_2^2$, where

$$\mathbf{D}_{2\mathrm{d}} = \begin{bmatrix} \mathbf{D}^{\mathrm{h}} \\ \mathbf{D}^{\mathrm{v}} \end{bmatrix} = \begin{bmatrix} \mathbf{D}_{1\mathrm{d}} \otimes \mathbf{I} \\ \mathbf{I} \otimes \mathbf{D}_{1\mathrm{d}} \end{bmatrix}, \hat{\mathbf{W}} = \mathrm{diag}\left(\left((\mathbf{D}^{\mathrm{h}}\mathbf{x})^2 + (\mathbf{D}^{\mathrm{v}}\mathbf{x})^2\right)^{-1/4}\right), W = \begin{bmatrix} \hat{\mathbf{W}} & 0 \\ 0 & \hat{\mathbf{W}} \end{bmatrix}$$

Smoothing Norm, $\mathbf{A} \in \mathbb{R}^{N\times N}$

Standard form transformation:

$$\bar{\mathbf{y}}_L = \arg\min_{\bar{\mathbf{y}}} \|\bar{\mathbf{A}}\bar{\mathbf{y}} - \bar{\mathbf{b}}\|_2^2 + \lambda\|\bar{\mathbf{y}}\|_2^2 \,, \quad \text{where} \quad \begin{array}{rcl} \bar{\mathbf{A}} & = & \mathbf{A}\mathbf{L}_{\mathbf{A}}^{\dagger} = \mathbf{A}[\mathbf{I} - (\mathbf{A}(\mathbf{I} - \mathbf{L}^{\dagger}\mathbf{L}))^{\dagger}\mathbf{A}] \\ \bar{\mathbf{b}} & = & \mathbf{b} - \mathbf{A}\mathbf{x}_0 \\ \mathbf{x}_L & = & \mathbf{L}_A^{\dagger}\bar{\mathbf{y}}_L + x_0 = \bar{\mathbf{x}}_L + \mathbf{x}_0 \end{array} .$$

Smoothing Norm, $\mathbf{A} \in \mathbb{R}^{N\times N}$

Standard form transformation:

$$\bar{\mathbf{y}}_L = \arg\min_{\bar{\mathbf{y}}} \|\bar{\mathbf{A}}\bar{\mathbf{y}} - \bar{\mathbf{b}}\|_2^2 + \lambda\|\bar{\mathbf{y}}\|_2^2\,, \quad \text{where} \quad \begin{array}{rcl} \bar{\mathbf{A}} & = & \mathbf{A}\mathbf{L}_{\mathbf{A}}^{\dagger} = \mathbf{A}[\mathbf{I} - (\mathbf{A}(\mathbf{I} - \mathbf{L}^{\dagger}\mathbf{L}))^{\dagger}\mathbf{A}] \\ \bar{\mathbf{b}} & = & \mathbf{b} - \mathbf{A}\mathbf{x}_0 \\ \mathbf{x}_L & = & \mathbf{L}_A^{\dagger}\bar{\mathbf{y}}_L + x_0 = \bar{\mathbf{x}}_L + \mathbf{x}_0 \end{array} .$$

[Hansen and Jensen. Smoothing-Norm Preconditioning for Reg. Min.-Res. *SIMAX*, 2007]

Write:

$$\mathbf{x}_L = \bar{\mathbf{x}}_L + \mathbf{x}_0 = \mathbf{L}_A^{\dagger}\bar{\mathbf{y}}_L + \mathbf{x}_0 = \mathbf{L}_A^{\dagger}\bar{\mathbf{y}}_L + \mathbf{K}\mathbf{t}_0\,, \quad \text{where} \quad \mathcal{R}(\mathbf{K}) = \mathcal{N}(\mathbf{L})\,, \quad \mathbf{L}_A^{\dagger} \text{ rectangular}\,.$$

Equivalently:

$$\mathbf{A}\left[\mathbf{L}_A^{\dagger},\ \mathbf{K}\right]\begin{bmatrix} \bar{\mathbf{y}}_L \\ \mathbf{t}_0 \end{bmatrix} = \mathbf{b}\,,$$

and, further:

$$\begin{bmatrix} (\mathbf{L}_A^{\dagger})^T\mathbf{A}\mathbf{L}_A^{\dagger} & (\mathbf{L}_A^{\dagger})^T\mathbf{A}\mathbf{K} \\ \mathbf{K}^T\mathbf{A}\mathbf{L}_A^{\dagger} & \mathbf{K}^T\mathbf{A}\mathbf{K} \end{bmatrix}\begin{bmatrix} \bar{\mathbf{y}}_L \\ \mathbf{t}_0 \end{bmatrix} = \begin{bmatrix} (\mathbf{L}_A^{\dagger})^T\mathbf{b} \\ \mathbf{K}^T\mathbf{b} \end{bmatrix}.$$

Smoothing Norm, $\mathbf{A} \in \mathbb{R}^{N\times N}$

Standard form transformation:

$$\bar{\mathbf{y}}_L = \arg\min_{\bar{\mathbf{y}}} \|\bar{\mathbf{A}}\bar{\mathbf{y}} - \bar{\mathbf{b}}\|_2^2 + \lambda\|\bar{\mathbf{y}}\|_2^2\,, \quad \text{where} \quad \begin{array}{rcl} \bar{\mathbf{A}} & = & \mathbf{A}\mathbf{L}_{\mathbf{A}}^{\dagger} = \mathbf{A}[\mathbf{I} - (\mathbf{A}(\mathbf{I} - \mathbf{L}^{\dagger}\mathbf{L}))^{\dagger}\mathbf{A}] \\ \bar{\mathbf{b}} & = & \mathbf{b} - \mathbf{A}\mathbf{x}_0 \\ \mathbf{x}_L & = & \mathbf{L}_A^{\dagger}\bar{\mathbf{y}}_L + x_0 = \bar{\mathbf{x}}_L + \mathbf{x}_0 \end{array}\,.$$

[Hansen and Jensen. Smoothing-Norm Preconditioning for Reg. Min.-Res. *SIMAX*, 2007]

Write:

$$\mathbf{x}_L = \bar{\mathbf{x}}_L + \mathbf{x}_0 = \mathbf{L}_A^{\dagger}\bar{\mathbf{y}}_L + \mathbf{x}_0 = \mathbf{L}_A^{\dagger}\bar{\mathbf{y}}_L + \mathbf{K}\mathbf{t}_0\,, \quad \text{where} \quad \mathcal{R}(\mathbf{K}) = \mathcal{N}(\mathbf{L})\,, \quad \mathbf{L}_A^{\dagger} \text{ rectangular}\,.$$

Equivalently:

$$\mathbf{A}\left[\mathbf{L}_A^{\dagger},\ \mathbf{K}\right]\begin{bmatrix} \bar{\mathbf{y}}_L \\ \mathbf{t}_0 \end{bmatrix} = \mathbf{b}\,,$$

and, further:

$$\begin{bmatrix} (\mathbf{L}_A^{\dagger})^T\mathbf{A}\mathbf{L}_A^{\dagger} & (\mathbf{L}_A^{\dagger})^T\mathbf{A}\mathbf{K} \\ \mathbf{K}^T\mathbf{A}\mathbf{L}_A^{\dagger} & \mathbf{K}^T\mathbf{A}\mathbf{K} \end{bmatrix}\begin{bmatrix} \bar{\mathbf{y}}_L \\ \mathbf{t}_0 \end{bmatrix} = \begin{bmatrix} (\mathbf{L}_A^{\dagger})^T\mathbf{b} \\ \mathbf{K}^T\mathbf{b} \end{bmatrix}.$$

Schur complement system:

$$(\mathbf{L}_A^{\dagger})^T\mathbf{P}\mathbf{A}\mathbf{L}_A^{\dagger}\bar{\mathbf{y}} = (\mathbf{L}_A^{\dagger})^T\mathbf{P}\mathbf{b}, \quad \text{where} \quad \mathbf{P} = \mathbf{I} - \mathbf{A}\mathbf{K}(\mathbf{K}^T\mathbf{A}\mathbf{K})^{-1}\mathbf{K}^T \in \mathbb{R}^{N\times N}.$$

TV regularization, $\mathbf{A} \in \mathbb{R}^{N \times N}$

[G. and Sabaté Landman (2019)]

Similar idea, with reweighting...

$$(\mathbf{D}^{\dagger})^T \mathbf{P}\mathbf{A}(\mathbf{W}\mathbf{D})_A^{\dagger}\bar{\mathbf{y}} = (\mathbf{D}^{\dagger})^T \mathbf{P}\mathbf{b}$$

Building a better approximation subspace for the solution!

TV regularization, $\mathbf{A} \in \mathbb{R}^{N\times N}$

[G. and Sabaté Landman (2019)]

Similar idea, with reweighting...

$$(\mathbf{D}^{\dagger})^T \mathbf{P}\mathbf{A}(\mathbf{W}\mathbf{D})_A^{\dagger}\bar{\mathbf{y}} = (\mathbf{D}^{\dagger})^T \mathbf{P}\mathbf{b}$$

Building a better approximation subspace for the solution!

- $\mathbf{L} = \mathbf{W}\mathbf{D}$ (with $\mathbf{W} = \mathbf{W}(\mathbf{x}_k)$):
  flexible GMRES (instead of restarted GMRES);

TV regularization, $\mathbf{A} \in \mathbb{R}^{N \times N}$

[G. and Sabaté Landman (2019)]

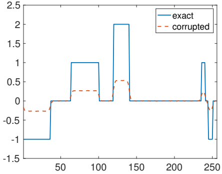
Similar idea, with reweighting...

$$(\mathbf{D}^{\dagger})^T \mathbf{P} \mathbf{A} (\mathbf{W}\mathbf{D})^{\dagger}_{A} \bar{\mathbf{y}} = (\mathbf{D}^{\dagger})^T \mathbf{P} \mathbf{b}$$

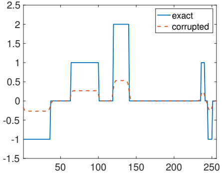
Building a better approximation subspace for the solution!

- $\mathbf{L} = \mathbf{W}\mathbf{D}$ (with $\mathbf{W} = \mathbf{W}(\mathbf{x}_k)$):
  flexible GMRES (instead of restarted GMRES);
- large-scale computations:
  - approximating $\mathbf{L}^{\dagger}$
    (exploiting structure, and running preconditioned LSQR or LSMR)
  - thresholding the weights

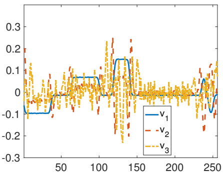
A simple 1D example...

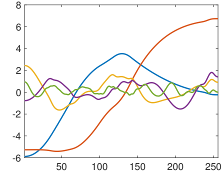


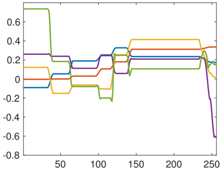
A simple 1D example...

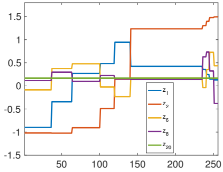


basis vectors

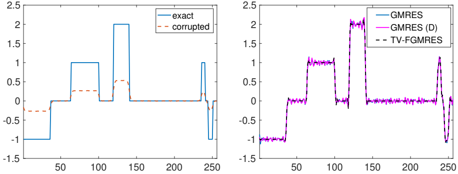








A simple 1D example...



basis vectors

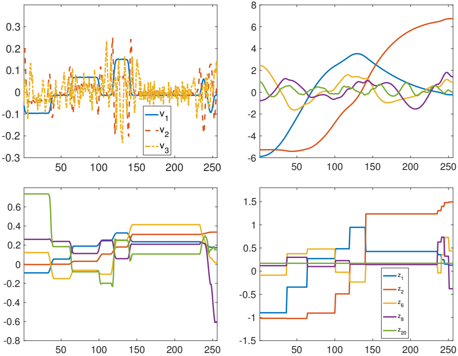


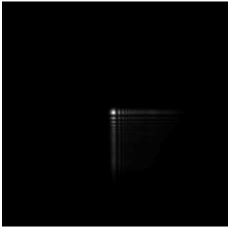
Image deblurring example with $\mathbf{\Psi} = \mathbf{I}$

[G., Hansen, Nagy. *IR Tools* (2018)]
https://github.com/silviagazzola/IRtools

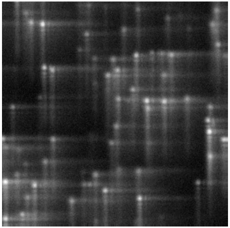
http://www2.compute.dtu.dk/ pcha/IRtools/

true

PSF

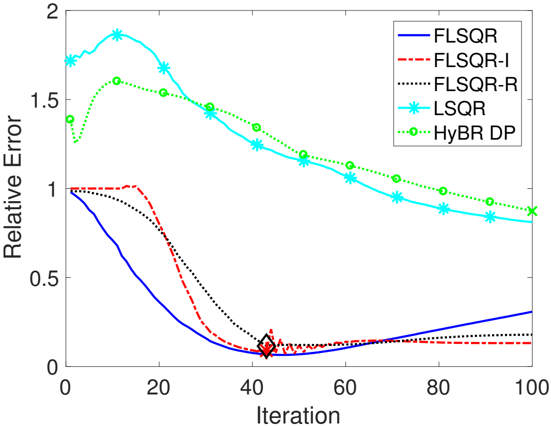


observed



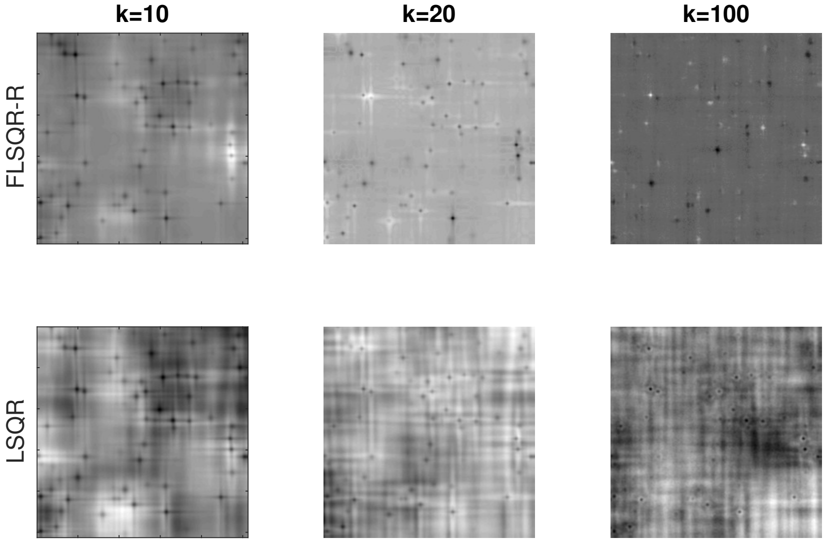
- Image is 256×256 pixels
- Noise level is 5×10^{-2}
- Reflexive boundary conditions

Reconstruction errors

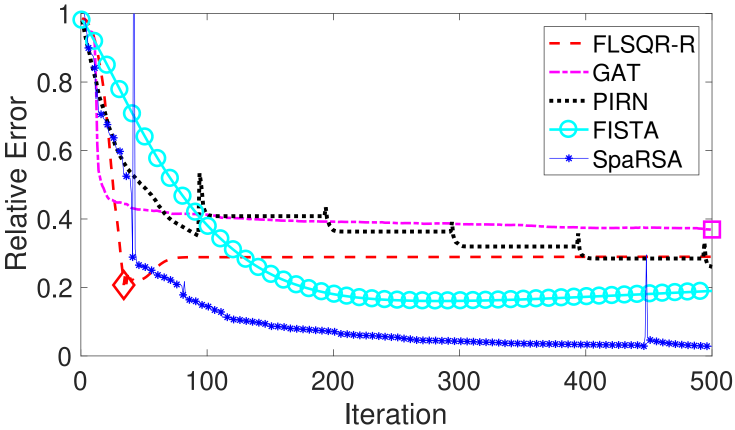


- Reconstruction errors computed as $\frac{\|\mathbf{x}_k - \mathbf{x}_{\text{true}}\|_2}{\|\mathbf{x}_{\text{true}}\|_2}$
- λ for FLSQR-I and FLSQR-R use discrepancy principle

Basis images



Comparison to other methods



- GAT = Generalized Arnoldi-Tikhonov
- PIRN† = Preconditioned iteratively re-weighted norm
- FISTA† = Fast iterative-shrinkage-thresholding algorithm
- SpaRSA† = Sparse Reconstruction by Separable Approximation

(† uses λ from FLSQR-R)

Tomography example with $\mathbf{\Phi} \neq \mathbf{I}$

Tomography example with $\mathbf{\Phi} \neq \mathbf{I}$

[G., Hansen, Nagy. *IR Tools* (2018)]

```
n = 256; optn = PRtomo('defaults');
optn=PRset(optn,'angles',0:2:179,'p',round(sqrt(2)*n),'d',sqrt(2)*n);
[A, b, x, ProbInfo] = PRtomo(n, optn);
```

- phantom is 256×256 pixels
- $\mathbf{A}$ has size 32580×65536 (approx. 50% undersampling)

Tomography example with $\mathbf{\Phi} \neq \mathbf{I}$

[G., Hansen, Nagy. *IR Tools* (2018)]

```
n = 256; optn = PRtomo('defaults');
optn=PRset(optn,'angles',0:2:179,'p',round(sqrt(2)*n),'d',sqrt(2)*n);
[A, b, x, ProbInfo] = PRtomo(n, optn);
figure, PRshowx(x, ProbInfo)
```

- phantom is 256×256 pixels
- $\mathbf{A}$ has size 32580×65536 (approx. 50% undersampling)

Tomography example with $\mathbf{\Phi} \neq \mathbf{I}$

[G., Hansen, Nagy. *IR Tools* (2018)]

```
n = 256; optn = PRtomo('defaults');
optn=PRset(optn,'angles',0:2:179,'p',round(sqrt(2)*n),'d',sqrt(2)*n);
[A, b, x, ProbInfo] = PRtomo(n, optn);
figure, PRshowx(x, ProbInfo)
bn = PRnoise(b, 1e-2);
```

- phantom is 256×256 pixels
- $\mathbf{A}$ has size 32580×65536 (approx. 50% undersampling)

Tomography example with $\mathbf{\Phi} \neq \mathbf{I}$

[G., Hansen, Nagy. *IR Tools* (2018)]

```
n = 256; optn = PRtomo('defaults');
optn=PRset(optn,'angles',0:2:179,'p',round(sqrt(2)*n),'d',sqrt(2)*n);
[A, b, x, ProbInfo] = PRtomo(n, optn);
figure, PRshowx(x, ProbInfo)
bn = PRnoise(b, 1e-2);
```

- phantom is 256×256 pixels
- $\mathbf{A}$ has size 32580×65536 (approx. 50% undersampling)
- $\mathbf{\Psi}$ is a 4-level 2D Haar wavelet transform

Reconstructed phantoms

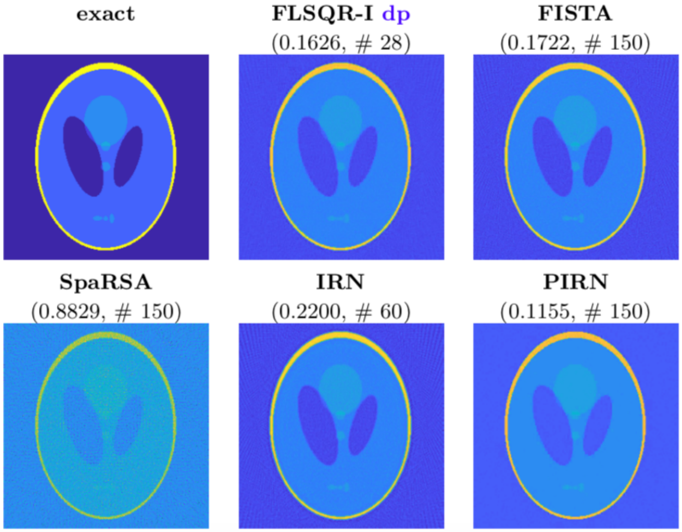


Image deblurring example

[G., Hansen, Nagy. *IR Tools* (2018)]

Cameraman example: 256×256 pixels.

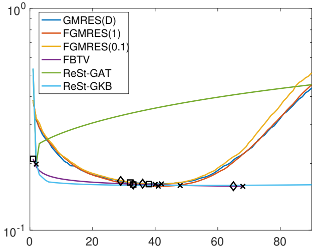
blurred & noisy

SN-GMRES

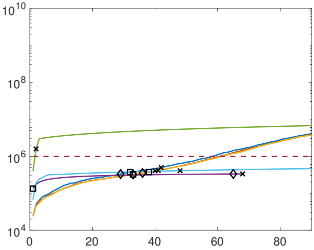
TV-FGMRES

fast gradient-based method

Image deblurring example



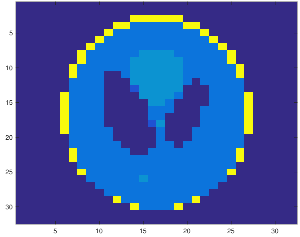
Relative Error History

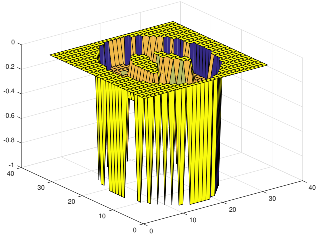


Total Variation History

Tomography example with flexible TV regularization

small PRtomo example: 32×32 pixels; $\mathbf{A} \in \mathbb{R}^{2025\times 1024}$
... ongoing work



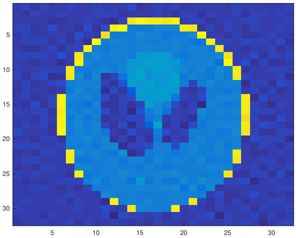


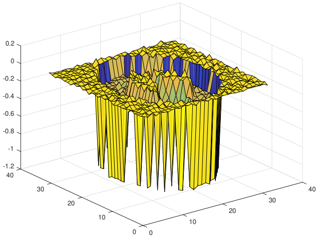
Exact phantom
noisy (Gaussian white noise, $\|\mathbf{e}\|/\|\mathbf{b}^{\mathrm{true}}\| = 10^{-2}$) image

Tomography example with flexible TV regularization

small `PRtomo` example: 32×32 pixels; $\mathbf{A} \in \mathbb{R}^{2025 \times 1024}$

... ongoing work

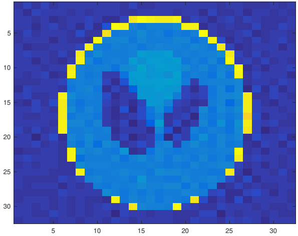


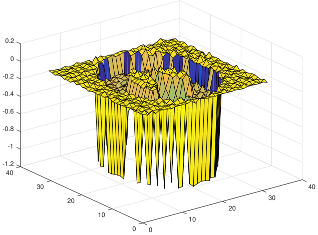


LSQR

Tomography example with flexible TV regularization

small PRtomo example: 32×32 pixels; $\mathbf{A} \in \mathbb{R}^{2025 \times 1024}$
... ongoing work

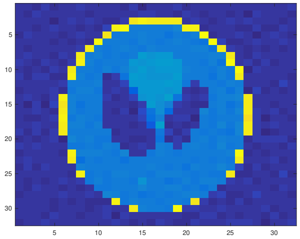


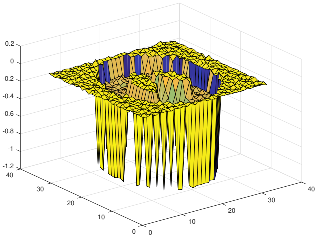


LSQR (D)

Tomography example with flexible TV regularization

small PRtomo example: 32×32 pixels; $\mathbf{A} \in \mathbb{R}^{2025 \times 1024}$
... ongoing work



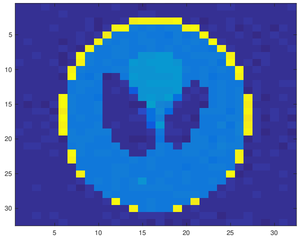


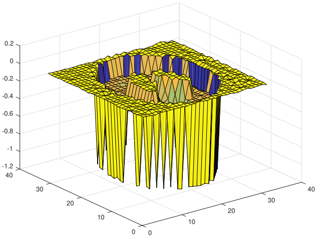
TV-LSQR

Tomography example with flexible TV regularization

small PRtomo example: 32×32 pixels; $\mathbf{A} \in \mathbb{R}^{2025 \times 1024}$

... ongoing work





TV-LSQR "0 norm"

Summary of benefits ...

- **Flexible Krylov methods**
  - ✔ Avoid inner-outer schemes (current solution immediately incorporated in basis)
  - ✔ Both square (flexible Arnoldi) non-square problems (flexible Golub-Kahan)
  - ✔ Optimality and equivalency results
- **Hybrid method**
  - ✔ Stabilize reconstruction errors
  - ✔ Automatic choice of λ and stopping criteria
- **Transformed problem**
  - ✔ Enforce sparsity in a transform
  - ✔ Connections to multi-parameter regularization

... and (hopefully) (much) more work to do ...

... and (hopefully) (much) more work to do ...

- deeper theoretical analysis (convergence, recovery guarantees);

... and (hopefully) (much) more work to do ...

- deeper theoretical analysis (convergence, recovery guarantees);
- extension to yet other regularization terms or constraints;

... and (hopefully) (much) more work to do ...

- deeper theoretical analysis (convergence, recovery guarantees);
- extension to yet other regularization terms or constraints;
- parameter choice for nonlinear problems and solvers;

... and (hopefully) (much) more work to do ...

- deeper theoretical analysis (convergence, recovery guarantees);
- extension to yet other regularization terms or constraints;
- parameter choice for nonlinear problems and solvers;

References

- J. Chung and S. G. *Flexible Krylov methods for ℓ^p regularization*. SISC, 2019.
- S. G. and M. Sabaté Landman. *Flexible GMRES for Total Variation regularization*. BIT, 2019.
- S. G., P. C. Hansen, and J. Nagy. *IR Tools: A MATLAB Package of Iterative Regularization Methods and Large-Scale Test Problems*. Numer. Algorithms, 2018. https://github.com/silviagazzola/IRtools