Efficient inverse problem solvers via randomized Krylov methods

Silvia Gazzola[1]
Joint work with: **Alberto Bucci**[2], **Julianne Chung**[3] and **Leonardo Robol**[1]

[1]Dipartimento di Matematica, Università di Pisa; `silvia.gazzola@unipi.it`, `leonardo.robol@unipi.it`

[2]School of Mathematics, The University of Edinburgh; `abucci2@ed.ac.uk`

[3]Department of Mathematics, Emory University; `jmchung@emory.edu`

March 2, 2026
Bayesian Inverse Problems and UQ workshop @ ICERM

Setting the stage: linear inverse problems

Setting the stage: linear inverse problems

[Hansen. *Discrete inverse problems*. SIAM, 2010]

Solving

$$\mathbf{A}\mathbf{x}_{\mathrm{true}} + \mathbf{e} = \mathbf{b}$$

where

$\mathbf{b} \in \mathbb{R}^m$	available observations or measurements
$\mathbf{x}_{\mathrm{true}} \in \mathbb{R}^n$	unknown quantity of interest
$\mathbf{A} \in \mathbb{R}^{m \times n}$	available large-scale ill-conditioned matrix models forward process
$\mathbf{e} \in \mathbb{R}^m$	unknown additive noise (assume white Gaussian)

Setting the stage: linear inverse problems

[Hansen. *Discrete inverse problems*. SIAM, 2010]

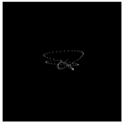
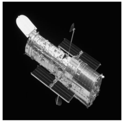
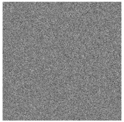
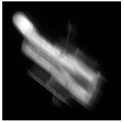
Solving

$$\mathbf{A}\mathbf{x}_{\mathrm{true}} + \mathbf{e} = \mathbf{b}$$

where

$\mathbf{b} \in \mathbb{R}^m$	available observations or measurements
$\mathbf{x}_{\mathrm{true}} \in \mathbb{R}^n$	unknown quantity of interest
$\mathbf{A} \in \mathbb{R}^{m\times n}$	available large-scale ill-conditioned matrix models forward process
$\mathbf{e} \in \mathbb{R}^m$	unknown additive noise (assume white Gaussian)

Example: image deblurring and denoising

 $*$  $+$  $=$ 

Setting the stage: linear inverse problems

[Hansen. *Discrete inverse problems*. SIAM, 2010]

Solving

$$\mathbf{A}\mathbf{x}_{\mathrm{true}} + \mathbf{e} = \mathbf{b}$$

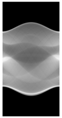
where

$\mathbf{b} \in \mathbb{R}^m$	available observations or measurements
$\mathbf{x}_{\mathrm{true}} \in \mathbb{R}^n$	unknown quantity of interest
$\mathbf{A} \in \mathbb{R}^{m \times n}$	available large-scale ill-conditioned matrix models forward process
$\mathbf{e} \in \mathbb{R}^m$	unknown additive noise (assume white Gaussian)

Example: computed tomography

$+$

$=$



Setting the stage: linear inverse problems

[Hansen. *Discrete inverse problems*. SIAM, 2010]

Solving

$$\mathbf{A}\mathbf{x}_{\mathrm{true}} + \mathbf{e} = \mathbf{b}$$

where

$\mathbf{b} \in \mathbb{R}^m$	available observations or measurements
$\mathbf{x}_{\mathrm{true}} \in \mathbb{R}^n$	unknown quantity of interest
$\mathbf{A} \in \mathbb{R}^{m \times n}$	available large-scale ill-conditioned matrix models forward process
$\mathbf{e} \in \mathbb{R}^m$	unknown additive noise (assume white Gaussian)

Computing $\mathbf{x} = \mathbf{A}^{\dagger}\mathbf{b}$:

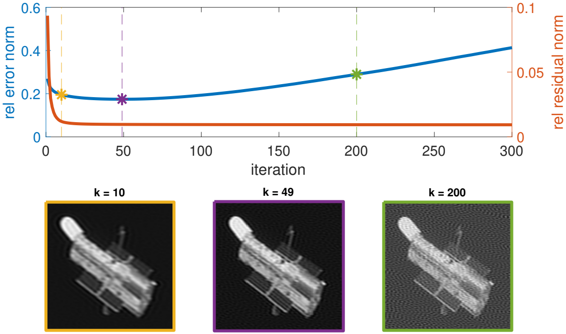
Overview: deterministic regularization methods

Overview: deterministic regularization methods

[Chung and Gazzola. *Computational Methods for Large-Scale Inverse Pbs.* SIAM Review, 2024]

- **Iterative regularization**
  - ✔ matrix-vector multiplications
  - ⚠ need good stopping criteria

$$\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x}-\mathbf{b}\|^2_{\mathbf{R}^{-1}}$$



Overview: deterministic regularization methods

[Chung and Gazzola. *Computational Methods for Large-Scale Inverse Pbs.* SIAM Review, 2024]

- **Iterative regularization**
  - ✔ matrix-vector multiplications
  - ⚠ need good stopping criteria
- **Variational regularization**
  - ✔ include different priors
  - ⚠ need regularization parameter

$$\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2_{\mathbf{R}^{-1}} + \lambda \mathcal{R}(\mathbf{x})$$

Common Regularizers

- $\mathcal{R}(\mathbf{x}) = \|\mathbf{L}\mathbf{x}\|_2^2$
- $\mathcal{R}(\mathbf{x}) = \|\mathbf{x}\|_p^p,\ p > 0$
- $\mathcal{R}(\mathbf{x}) = \mathbf{x}^\top \mathbf{Q}^{-1}\mathbf{x},\ \mathbf{Q}$ SPD
- $\mathcal{R}(\mathbf{x}) = TV(\mathbf{x})$

Choosing $\lambda \geq 0$ (regularization parameter)

- Discrepancy principle (dp)
- Generalized cross validation (gcv)

Overview: deterministic regularization methods

[Chung and Gazzola. *Computational Methods for Large-Scale Inverse Pbs.* SIAM Review, 2024]

- **Iterative regularization**
  - ✔ matrix-vector multiplications
  - ⚠ need good stopping criteria
- **Variational regularization**
  - ✔ include different priors
  - ⚠ need regularization parameter

The best of both worlds!

- **Hybrid projection methods**
  - ✔ matrix-vector multiplications
  - ✔ automatic regularization parameter
  - ✔ automatic stopping iteration

Overview: deterministic regularization methods

[Chung and Gazzola. *Computational Methods for Large-Scale Inverse Pbs.* SIAM Review, 2024]

- **Iterative regularization**
  - ✔ matrix-vector multiplications
  - ⚠ need good stopping criteria
- **Variational regularization**
  - ✔ include different priors
  - ⚠ need regularization parameter

The best of both worlds!

- **Hybrid projection methods**
  - ✔ matrix-vector multiplications
  - ✔ automatic regularization parameter
  - ✔ automatic stopping iteration

Standard Krylov methods:
approximation and constraint subspaces generated by
- Arnoldi
- Golub-Kahan bidiagonalization (with reorthogonalization)

Overview: deterministic regularization methods

[Chung and Gazzola. *Computational Methods for Large-Scale Inverse Pbs.* SIAM Review, 2024]

- **Iterative regularization**
  - ✔ matrix-vector multiplications
  - ⚠ need good stopping criteria
- **Variational regularization**
  - ✔ include different priors
  - ⚠ need regularization parameter

The best of both worlds!

- **Hybrid projection methods**
  - ✔ matrix-vector multiplications
  - ✔ automatic regularization parameter
  - ✔ automatic stopping iteration

Standard Krylov methods

- ✔ can include different priors

Non-Standard Krylov methods:

Efficient iteratively reweighted norms via

- Flexible Krylov methods (e.g., FGMRES, FLSQR)
- Generalized Krylov sunspace methods

Overview: deterministic regularization methods

[Chung and Gazzola. *Computational Methods for Large-Scale Inverse Pbs.* SIAM Review, 2024]

- **Iterative regularization**
  - ✔ matrix-vector multiplications
  - ⚠ need good stopping criteria

- **Variational regularization**
  - ✔ include different priors
  - ⚠ need regularization parameter

The best of both worlds!

- **Hybrid projection methods**
  - ✔ matrix-vector multiplications
  - ✔ automatic regularization parameter
  - ✔ automatic stopping iteration

Standard Krylov methods

✔ can include different priors

Non-Standard Krylov methods

Overview: (hybrid) projection methods for Bayesian inverse problems

[Chung and Gazzola. *Computational Methods for Large-Scale Inverse Pbs.* SIAM Review, 2024]

Overview: (hybrid) projection methods for Bayesian inverse problems

[Chung and Gazzola. *Computational Methods for Large-Scale Inverse Pbs.* SIAM Review, 2024]

$$\mathbf{b} \sim \underbrace{\mathcal{N}(\mathbf{A}\mathbf{x}_{\mathrm{true}}, \sigma^2\mathbf{R})}_{\text{determines likelihood}} \quad \text{and} \quad \mathbf{x} \sim \underbrace{\mathcal{N}(\boldsymbol{\mu}, \alpha^2\mathbf{Q})}_{\text{determines prior}}, \quad \mathbf{Q} \in \mathbb{R}^{n\times n},\ \mathbf{R} \in \mathbb{R}^{m\times m} \quad \text{SPD}$$

Posterior density function from Bayes' theorem

$$\pi_{\mathrm{post}}(\mathbf{x} \mid \mathbf{b}) = \frac{\pi_{\mathrm{like}}(\mathbf{b} \mid \mathbf{x})\pi_{\mathrm{prior}}(\mathbf{x})}{\pi(\mathbf{b})} \propto \exp\left(-\frac{1}{2\sigma^2}\|\mathbf{b} - \mathbf{A}\mathbf{x}\|^2_{\mathbf{R}^{-1}} - \frac{1}{2\alpha^2}\|\mathbf{x} - \boldsymbol{\mu}\|^2_{\mathbf{Q}^{-1}}\right)$$

Overview: (hybrid) projection methods for Bayesian inverse problems

[Chung and Gazzola. *Computational Methods for Large-Scale Inverse Pbs.* SIAM Review, 2024]

$$\mathbf{b} \sim \underbrace{\mathcal{N}(\mathbf{A}\mathbf{x}_{\mathrm{true}}, \sigma^2\mathbf{R})}_{\text{determines likelihood}} \quad \text{and} \quad \mathbf{x} \sim \underbrace{\mathcal{N}(\boldsymbol{\mu}, \alpha^2\mathbf{Q})}_{\text{determines prior}}, \quad \mathbf{Q} \in \mathbb{R}^{n\times n},\ \mathbf{R} \in \mathbb{R}^{m\times m} \quad \text{SPD}$$

Posterior density function from Bayes' theorem

$$\pi_{\mathrm{post}}(\mathbf{x} \mid \mathbf{b}) = \frac{\pi_{\mathrm{like}}(\mathbf{b} \mid \mathbf{x})\pi_{\mathrm{prior}}(\mathbf{x})}{\pi(\mathbf{b})} \propto \exp\left(-\frac{1}{2\sigma^2}\|\mathbf{b} - \mathbf{A}\mathbf{x}\|^2_{\mathbf{R}^{-1}} - \frac{1}{2\alpha^2}\|\mathbf{x} - \boldsymbol{\mu}\|^2_{\mathbf{Q}^{-1}}\right)$$

Krylov and hybrid projection methods for **exploring the posterior**:

Overview: (hybrid) projection methods for Bayesian inverse problems

[Chung and Gazzola. *Computational Methods for Large-Scale Inverse Pbs.* SIAM Review, 2024]

$$\mathbf{b} \sim \underbrace{\mathcal{N}(\mathbf{A}\mathbf{x}_{\text{true}}, \sigma^2\mathbf{R})}_{\text{determines likelihood}} \quad \text{and} \quad \mathbf{x} \sim \underbrace{\mathcal{N}(\boldsymbol{\mu}, \alpha^2\mathbf{Q})}_{\text{determines prior}}, \quad \mathbf{Q} \in \mathbb{R}^{n\times n},\ \mathbf{R} \in \mathbb{R}^{m\times m} \quad \text{SPD}$$

Posterior density function from Bayes' theorem

$$\pi_{\text{post}}(\mathbf{x} \mid \mathbf{b}) = \frac{\pi_{\text{like}}(\mathbf{b} \mid \mathbf{x})\pi_{\text{prior}}(\mathbf{x})}{\pi(\mathbf{b})} \propto \exp\left(-\frac{1}{2\sigma^2}\|\mathbf{b} - \mathbf{A}\mathbf{x}\|^2_{\mathbf{R}^{-1}} - \frac{1}{2\alpha^2}\|\mathbf{x} - \boldsymbol{\mu}\|^2_{\mathbf{Q}^{-1}}\right)$$

Krylov and hybrid projection methods for **exploring the posterior**:

- MAP estimate

$$\mathbf{x}_{\text{MAP}} = \arg\max_{\mathbf{x}} \pi_{\text{post}}(\mathbf{x} \mid \mathbf{b}) = \left(\mathbf{A}^\top\mathbf{R}^{-1}\mathbf{A} + \lambda\mathbf{Q}^{-1}\right)^{-1}\left(\mathbf{A}^\top\mathbf{R}^{-1}\mathbf{b} + \lambda\mathbf{Q}^{-1}\boldsymbol{\mu}\right), \quad \lambda = \frac{\sigma^2}{\alpha^2}$$

  - handling unknown hyperparameters (automatic regularization parameter selection)
    [Arridge, Betcke, Harhanen(2014), Chung and Saibaba (217)]
  - as sub-solvers for hierarchical Bayesian models (within alternating optimization)
    [Calvetti, Pitolli, Somersalo, Vantaggi (2018), Calvetti, Pragliola, Somersalo, Strang (2020)]

Overview: (hybrid) projection methods for Bayesian inverse problems

[Chung and Gazzola. *Computational Methods for Large-Scale Inverse Pbs.* SIAM Review, 2024]

$$\mathbf{b} \sim \underbrace{\mathcal{N}(\mathbf{A}\mathbf{x}_{\text{true}}, \sigma^2 \mathbf{R})}_{\text{determines likelihood}} \quad \text{and} \quad \mathbf{x} \sim \underbrace{\mathcal{N}(\boldsymbol{\mu}, \alpha^2 \mathbf{Q})}_{\text{determines prior}}, \quad \mathbf{Q} \in \mathbb{R}^{n\times n},\ \mathbf{R} \in \mathbb{R}^{m \times m} \quad \text{SPD}$$

Posterior density function from Bayes' theorem

$$\pi_{\text{post}}(\mathbf{x} \mid \mathbf{b}) = \frac{\pi_{\text{like}}(\mathbf{b} \mid \mathbf{x})\pi_{\text{prior}}(\mathbf{x})}{\pi(\mathbf{b})} \propto \exp\left(-\frac{1}{2\sigma^2}\|\mathbf{b} - \mathbf{A}\mathbf{x}\|^2_{\mathbf{R}^{-1}} - \frac{1}{2\alpha^2}\|\mathbf{x} - \boldsymbol{\mu}\|^2_{\mathbf{Q}^{-1}}\right)$$

Krylov and hybrid projection methods for **exploring the posterior**:

- MAP estimate

$$\mathbf{x}_{\text{MAP}} = \arg\max_{\mathbf{x}} \pi_{\text{post}}(\mathbf{x} \mid \mathbf{b}) = \left(\mathbf{A}^\top \mathbf{R}^{-1}\mathbf{A} + \lambda \mathbf{Q}^{-1}\right)^{-1}\left(\mathbf{A}^\top \mathbf{R}^{-1}\mathbf{b} + \lambda \mathbf{Q}^{-1}\boldsymbol{\mu}\right), \quad \lambda = \frac{\sigma^2}{\alpha^2}$$

- Gaussian posterior variance-covariance estimation and sampling (via low-rank approx)
  [Cho, Chung, Miller, Saibaba (2022), Chow and Y. Saad (2014), Schneider and Willsky (2003)]

$$\boldsymbol{\Gamma} \equiv \left(\tfrac{1}{\alpha^2}\mathbf{I} + \tfrac{1}{\sigma^2}\mathbf{A}^\top\mathbf{A}\right)^{-1} \simeq \boldsymbol{\Gamma}_k := \sigma^2(\lambda^{-1}\mathbf{I}_n - \mathbf{Z}_k\boldsymbol{\Delta}_k\mathbf{Z}_k^\top)$$

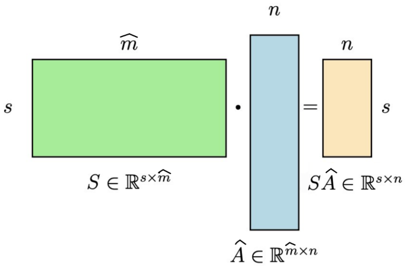
A look at randomization and efficiency considerations

A look at randomization and efficiency considerations

- Randomization through 'sketch and solve'

  [Avron, Clarkson and Woodruff (2017), Iyer and Drineas (2016), Gazagnadou, Ibrahim and Gower (2022)]

$$\min_{\mathbf{x}} \|\mathbf{S}(\underbrace{\begin{bmatrix}\mathbf{A}\\ \lambda\mathbf{I}\end{bmatrix}}_{=:\widehat{\mathbf{A}}}\mathbf{x} - \underbrace{\begin{bmatrix}\mathbf{b}\\ \mathbf{0}\end{bmatrix}}_{=:\widehat{\mathbf{b}}})\|_2^2$$

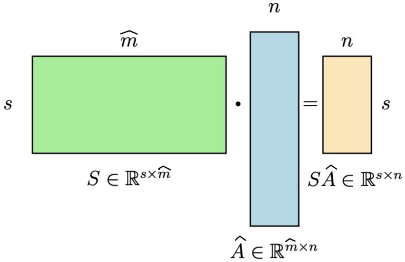


A look at randomization and efficiency considerations

- Randomization through 'sketch and solve'

  [Avron, Clarkson and Woodruff (2017), Iyer and Drineas (2016), Gazagnadou, Ibrahim and Gower (2022)]

$$\min_{\mathbf{x}} \|\mathbf{S}(\underbrace{\begin{bmatrix}\mathbf{A}\\ \lambda\mathbf{I}\end{bmatrix}}_{=:\widehat{\mathbf{A}}}\mathbf{x} - \underbrace{\begin{bmatrix}\mathbf{b}\\ \mathbf{0}\end{bmatrix}}_{=:\widehat{\mathbf{b}}})\|_2^2$$



- Avoiding inner products

  [Brown, Chung, Nagy, and Sabaté Landman. *Inner-product free Krylov methods for inverse problems*, SISC, 2025]

  - expensive if the vectors are very high-dimensional or the number of iterations is high
  - communication bottleneck if vectors are distributed across multiple processors

Overall goals

Overall goals

- Develop a randomized Golub-Kahan (rGK) algorithm, based on sketched inner products, extending randomized Arnoldi

  [Balabanov and Grigori. *Randomized Gram–Schmidt process with application to GMRES*, SISC (2022)]

Overall goals

- Develop a randomized Golub-Kahan (rGK) algorithm, based on sketched inner products, extending randomized Arnoldi

  [Balabanov and Grigori. *Randomized Gram–Schmidt process with application to GMRES*, SISC (2022)]

- Introduce new randomized Krylov subspace methods, built on rGK, extending randomized GMRES (rGMRES)

  randomized LSQR (rLSQR), randomized CGLS (rCGLS), randomized LSMR (rLSMR)

- Derive hybrid rGMRES, rLSQR, rCGLS and rLSMR methods (projection + Tikhonov)

$$\mathbf{x}_{\text{reg}} = \arg\min_{\mathbf{x}\in\mathbb{R}^n} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 + \lambda^2\|\mathbf{x}\|^2$$

with automatic regularization parameter estimation

Randomized inner products

Main idea

Estimate l_2-inner product in $\mathbb{R}^n$ by

$$\langle \cdot, \cdot \rangle \approx \langle \mathbf{\Theta}^{(n)} \cdot, \mathbf{\Theta} \cdot^{(n)} \rangle$$

where $\mathbf{\Theta}^{(n)} \in \mathbb{R}^{\ell_n \times n}$ is a sketching matrix with $\ell_n \ll n$

Randomized inner products

Main idea

Estimate l_2-inner product in $\mathbb{R}^n$ by

$$\langle \cdot, \cdot \rangle \approx \langle \boldsymbol{\Theta}^{(n)} \cdot, \boldsymbol{\Theta} \cdot^{(n)} \rangle$$

where $\boldsymbol{\Theta}^{(n)} \in \mathbb{R}^{\ell_n \times n}$ is a sketching matrix with $\ell_n \ll n$

- $\boldsymbol{\Theta}^{(n)}$ is an ε embedding for $\mathcal{V}$ if

$$\left| \langle \mathbf{x}, \mathbf{y} \rangle - \langle \boldsymbol{\Theta}^{(n)} \mathbf{x}, \boldsymbol{\Theta}^{(n)} \mathbf{y} \rangle \right| \leq \varepsilon \|\mathbf{x}\| \|\mathbf{y}\|, \quad \text{for all } \mathbf{x}, \mathbf{y} \in \mathcal{V}$$

implying

$$(1+\varepsilon)^{-1} \|\boldsymbol{\Theta}^{(n)} \mathbf{x}\|^2 \leq \|\mathbf{x}\|^2 \leq (1-\varepsilon)^{-1} \|\boldsymbol{\Theta}^{(n)} \mathbf{x}\|^2$$

Randomized inner products

Main idea

Estimate l_2-inner product in $\mathbb{R}^n$ by

$$\langle \cdot, \cdot \rangle \approx \langle \boldsymbol{\Theta}^{(n)} \cdot, \boldsymbol{\Theta} \cdot^{(n)} \rangle$$

where $\boldsymbol{\Theta}^{(n)} \in \mathbb{R}^{\ell_n \times n}$ is a sketching matrix with $\ell_n \ll n$

- $\boldsymbol{\Theta}^{(n)}$ is an ε embedding for $\mathcal{V}$ if

  $$\left| \langle \mathbf{x}, \mathbf{y} \rangle - \langle \boldsymbol{\Theta}^{(n)} \mathbf{x}, \boldsymbol{\Theta}^{(n)} \mathbf{y} \rangle \right| \leq \varepsilon \|\mathbf{x}\| \|\mathbf{y}\|, \quad \text{for all } \mathbf{x}, \mathbf{y} \in \mathcal{V}$$

  implying

  $$(1+\varepsilon)^{-1} \|\boldsymbol{\Theta}^{(n)} \mathbf{x}\|^2 \leq \|\mathbf{x}\|^2 \leq (1-\varepsilon)^{-1} \|\boldsymbol{\Theta}^{(n)} \mathbf{x}\|^2$$

- $(\varepsilon, \delta, K+1)$-oblivious embeddings:
  with probability at least $1-\delta$ for any fixed $(K+1)$-dimensional subspace of $\mathbb{R}^n$

Randomized Arnoldi

Exploits randomized Gram-Schmidt to compute a basis $\{\mathbf{q}_1, \dots, \mathbf{q}_{k+1}\}$ of

$$\mathcal{K}_{k+1}(\mathbf{A}, \mathbf{b}) = \mathrm{span}\{\mathbf{b}, \mathbf{A}\mathbf{b}, \dots, \mathbf{A}^k\mathbf{b}\}$$

After $k+1$ iterations, we have

- $\mathbf{Q}_{k+1} = [\mathbf{q}_1, \dots, \mathbf{q}_{k+1}] \in \mathbb{R}^{n\times(k+1)}$ with columns $\mathbf{q}_i$, *orthonormal with respect to the sketched inner product* $\langle \mathbf{\Theta}^{(n)}\cdot, \mathbf{\Theta}^{(n)}\cdot \rangle$
- upper Hessenberg matrix $\mathbf{H}_k \in \mathbb{R}^{(k+1)\times k}$

such that

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

Randomized GMRES (rGMRES) and hybrid rGMRES, $k \leq K$ iterates

rGMRES

Computes

$$\mathbf{x}_k = \mathbf{Q}_k \mathbf{z}_k \quad \text{where} \quad \mathbf{z}_k = \arg\min_{\mathbf{z}} \|\mathbf{H}_k \mathbf{z} - \beta \mathbf{e}_1\| \quad \text{and} \quad \beta = \|\boldsymbol{\Theta}^{(n)} \mathbf{b}\|$$

hybrid rGMRES (with Tikhonov regularization)

Computes

$$\mathbf{x}_k = \mathbf{Q}_k \mathbf{z}_k \quad \text{where} \quad \mathbf{z}_k = \arg\min_{\mathbf{z}} \|\mathbf{H}_k \mathbf{z} - \beta \mathbf{e}_1\|^2 + \lambda^2 \|\mathbf{z}\|^2$$

Equivalently,

$$\min_{\mathbf{x} \in \mathcal{K}_k(\mathbf{A},\mathbf{b})} \|\boldsymbol{\Theta}^{(n)}(\mathbf{A}\mathbf{x} - \mathbf{b})\|^2 + \lambda^2 \|\boldsymbol{\Theta}^{(n)} \mathbf{x}\|^2 ,$$

- ‘regularize-then-project’ $\neq$ ‘project-then-regularize’
- $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$ independent of λ, allowing automatic selection of λ (e.g., dp, gcv, wgcv, ...)

rGMRES: algorithmic ‘sketch’

Require: $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{b} \in \mathbb{R}^{n}$, $\boldsymbol{\Theta}^{(n)} \in \mathbb{R}^{\ell_n \times n}$ $(\ell_n \ll n)$

Initialize $\widetilde{\mathbf{q}}_1 = \mathbf{b}$, sketch $\widetilde{\mathbf{s}}_1 = \boldsymbol{\Theta}^{(n)}\widetilde{\mathbf{q}}_1$

Compute the sketched norm $\beta = \|\widetilde{\mathbf{s}}_1\|$

Scale vectors $\mathbf{s}_1 = \widetilde{\mathbf{s}}_1/\beta$, $\mathbf{q}_1 = \widetilde{\mathbf{q}}_1/\beta$

for $k = 1, 2, \dots K$, or a stopping criterion is satisfied **do**

  Randomized Arnoldi (Randomized Gram-Schmidt)

  Get new vector $\widetilde{\mathbf{q}}_{k+1} = \mathbf{A}\mathbf{q}_k$, sketch $\widetilde{\mathbf{s}}_{k+1} = \boldsymbol{\Theta}^{(n)}\widetilde{\mathbf{q}}_{k+1}$

  Compute $[\mathbf{R}]_{(1:k,k+1)} = \mathbf{S}_k{}^{\top}\widetilde{\mathbf{s}}_{k+1}$

  Compute projection: $\widetilde{\mathbf{q}}_{k+1} = \widetilde{\mathbf{q}}_{k+1} - \mathbf{Q}_k[\mathbf{R}]_{(1:k,k+1)}$

  Compute the sketched projection $\widetilde{\mathbf{s}}_{k+1} = \widetilde{\mathbf{s}}_{k+1} - \mathbf{S}_k[\mathbf{R}]_{(1:k,k+1)}$

  Compute the sketched norm $r_{k+1,k+1} = \|\widetilde{\mathbf{s}}_{k+1}\|$

  Scale vectors: $\mathbf{s}_{k+1} = \widetilde{\mathbf{s}}_{k+1}/r_{k+1,k+1}$, $\mathbf{q}_{k+1} = \widetilde{\mathbf{q}}_{k+1}/r_{k+1,k+1}$

  Solve the projected problem & compute the approximate solution

  $\mathbf{z}_k = \arg\min_{\mathbf{z}} \|\mathbf{H}_k\mathbf{z} - \beta\mathbf{e}_1\|$, where $\mathbf{H}_k = [\mathbf{R}]_{(1:k+1,2:k)}$

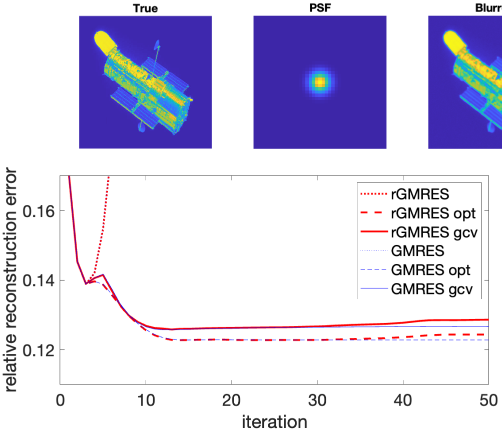
end for

$\mathbf{x}_k = \mathbf{Q}_k\mathbf{z}_k$

Numerical Results: deblurring

- $\mathbf{A} \in \mathbb{R}^{262,144 \times 262,144}$
- $\mathbf{b} \in \mathbb{R}^{262,144}$ with 1% noise
- $\mathbf{\Theta}^{(n)} \in \mathbb{R}^{\ell_n \times n}$
  Subsampled Randomized Hadamard Transform (SRHT), $\ell_n = 13,106$ (5% of n)
- Relative reconstruction error,

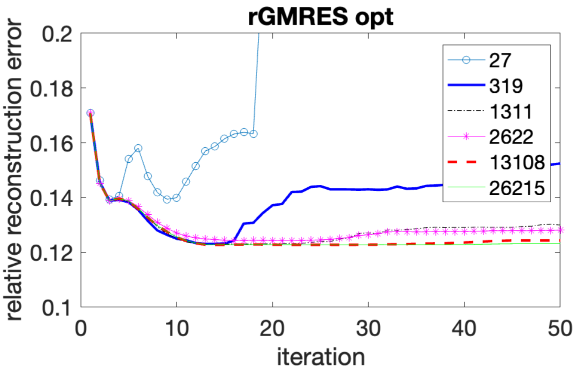
$$\frac{\|\mathbf{x}_k - \mathbf{x}_{\mathrm{true}}\|}{\|\mathbf{x}_{\mathrm{true}}\|}$$



Deblurring: hybrid rGMRES with different sampling size

- $K = 50$ iterations
- Legend values correspond to ℓ_n:
  - 319 is default value
  - $\min(n, \texttt{ceil}(2 * K * \log(n)/\log(K)))$
  - $0.01\%, 0.5\%, 1\%, 5\%$ and 10%

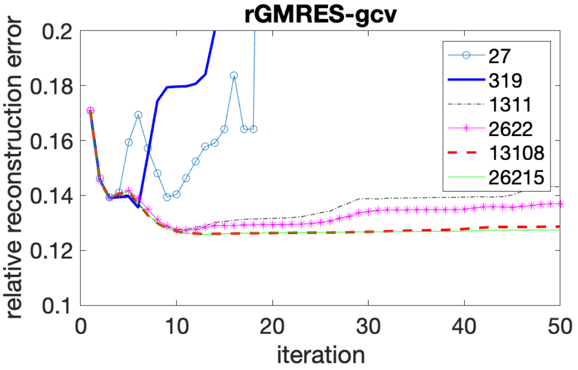
  (percentage of the dimension n)
- Optimal regularization parameter



Deblurring: hybrid rGMRES with different sampling size

- $K = 50$ iterations
- Legend values correspond to ℓ_n:
  - 319 is default value
  - $\min(n, \texttt{ceil}(2 * K * \log(n)/\log(K)))$
  - $0.01\%, 0.5\%, 1\%, 5\%$ and 10%

  (percentage of the dimension n)
- Regularization parameter chosen by gcv



Extension to rectangular matrices, $\mathbf{A} \in \mathbb{R}^{m\times n}$

$$\text{Building } \mathcal{K}_k(\mathbf{A}^\top\mathbf{A}, \mathbf{A}^\top\mathbf{b}) \subset \mathbb{R}^n \text{ and } \mathcal{K}_k(\mathbf{A}\mathbf{A}^\top, \mathbf{b}) \subset \mathbb{R}^m$$

Let

- $\mathbf{\Theta}^{(n)} \in \mathbb{R}^{\ell_n \times n}$, with $\ell_n \ll n$, be an oblivious sketching matrix for vectors in $\mathbb{R}^n$
- $\mathbf{\Theta}^{(m)} \in \mathbb{R}^{\ell_m \times m}$, with $\ell_m \ll m$, be an oblivious sketching matrix for vectors in $\mathbb{R}^m$

Main idea

- Estimate ℓ_2-inner product in $\mathbb{R}^n$ by

$$\langle \cdot, \cdot \rangle \approx \langle \mathbf{\Theta}^{(n)} \cdot, \mathbf{\Theta}^{(n)} \cdot \rangle$$

- Estimate ℓ_2-inner product in $\mathbb{R}^m$ by

$$\langle \cdot, \cdot \rangle \approx \langle \mathbf{\Theta}^{(m)} \cdot, \mathbf{\Theta}^{(m)} \cdot \rangle$$

Randomized Golub-Kahan (rGK)

After $k \leq K$ iterations, we have the partial matrix factorizations,

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

where

- $\mathbf{V}_{k+1} = [\mathbf{v}_1, \ldots, \mathbf{v}_{k+1}] \in \mathbb{R}^{n \times (k+1)}$ has $\mathbf{\Theta}^{(n)}$-orthogonal columns
- $\mathbf{U}_{k+1} = [\mathbf{u}_1, \ldots, \mathbf{u}_{k+1}] \in \mathbb{R}^{m \times (k+1)}$ has $\mathbf{\Theta}^{(m)}$-orthogonal columns
- $\mathbf{M}_k \in \mathbb{R}^{(k+1) \times k}$ is upper Hessenberg
- $\mathbf{T}_{k+1} \in \mathbb{R}^{(k+1) \times (k+1)}$ is upper triangular

Randomized Golub-Kahan (rGK)

After $k \leq K$ iterations, we have the partial matrix factorizations,

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

where

- $\mathbf{V}_{k+1} = [\mathbf{v}_1, \dots, \mathbf{v}_{k+1}] \in \mathbb{R}^{n\times(k+1)}$ has $\mathbf{\Theta}^{(n)}$-orthogonal columns
- $\mathbf{U}_{k+1} = [\mathbf{u}_1, \dots, \mathbf{u}_{k+1}] \in \mathbb{R}^{m\times(k+1)}$ has $\mathbf{\Theta}^{(m)}$-orthogonal columns
- $\mathbf{M}_k \in \mathbb{R}^{(k+1)\times k}$ is upper Hessenberg
- $\mathbf{T}_{k+1} \in \mathbb{R}^{(k+1)\times(k+1)}$ is upper triangular

No further simplifications as

$$\mathbf{V}_{k+1}^\top((\mathbf{\Theta}^{(n)})^\top \mathbf{\Theta}^{(n)})(\mathbf{A}^\top \mathbf{A})\mathbf{V}_k = \mathbf{T}_{k+1}\mathbf{M}_k \quad \text{not symmetric}$$

rGK: algorithmic 'sketch'

Require: $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{\Theta}^{(n)} \in \mathbb{R}^{\ell_n \times n}$ $(\ell_n \ll n)$, $\mathbf{\Theta}^{(m)} \in \mathbb{R}^{\ell_m \times m}$ $(\ell_m \ll m)$

1: Initialize & sketch: $\widetilde{\mathbf{u}}_1 = \mathbf{b}$, $\widetilde{\mathbf{s}}_1 = \mathbf{\Theta}^{(m)}\widetilde{\mathbf{u}}_1$; scale: $\mathbf{s}_1 = \widetilde{\mathbf{s}}_1/\beta$, $\mathbf{u}_1 = \widetilde{\mathbf{u}}_1/\beta$, where $\beta = \|\widetilde{\mathbf{s}}_1\|$

2: Compute & sketch: $\widetilde{\mathbf{v}}_1 = \mathbf{A}^\top \mathbf{u}_1$, $\widetilde{\mathbf{p}}_1 = \mathbf{\Theta}^{(n)}\widetilde{\mathbf{v}}_1$

3: Scale: $\mathbf{p}_1 = \widetilde{\mathbf{p}}_1/t_{1,1}$, $\mathbf{v}_1 = \widetilde{\mathbf{v}}_1/t_{1,1}$, where $t_{1,1} = \|\widetilde{\mathbf{p}}_1\|$

rGK: algorithmic 'sketch'

Require: $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{\Theta}^{(n)} \in \mathbb{R}^{\ell_n \times n}$ $(\ell_n \ll n)$, $\mathbf{\Theta}^{(m)} \in \mathbb{R}^{\ell_m \times m}$ $(\ell_m \ll m)$

1: Initialize & sketch: $\widetilde{\mathbf{u}}_1 = \mathbf{b}$, $\widetilde{\mathbf{s}}_1 = \mathbf{\Theta}^{(m)}\widetilde{\mathbf{u}}_1$; scale: $\mathbf{s}_1 = \widetilde{\mathbf{s}}_1/\beta$, $\mathbf{u}_1 = \widetilde{\mathbf{u}}_1/\beta$, where $\beta = \|\widetilde{\mathbf{s}}_1\|$

2: Compute & sketch: $\widetilde{\mathbf{v}}_1 = \mathbf{A}^\top \mathbf{u}_1$, $\widetilde{\mathbf{p}}_1 = \mathbf{\Theta}^{(n)}\widetilde{\mathbf{v}}_1$

3: Scale: $\mathbf{p}_1 = \widetilde{\mathbf{p}}_1/t_{1,1}$, $\mathbf{v}_1 = \widetilde{\mathbf{v}}_1/t_{1,1}$, where $t_{1,1} = \|\widetilde{\mathbf{p}}_1\|$

4: **for** $k = 1, 2, \dots K$, or a stopping criterion is satisfied **do**

5: **Randomized Gram-Schmidt for constructing U**

6: Get new vector & sketch: $\widetilde{\mathbf{u}}_{k+1} = \mathbf{A}\mathbf{v}_k$, $\widetilde{\mathbf{s}}_{k+1} = \mathbf{\Theta}^{(m)}\widetilde{\mathbf{u}}_{k+1}$

7: Compute $[\mathbf{M}]_{(1:k,k)} = \mathbf{S}_k{}^\top \widetilde{\mathbf{s}}_{k+1}$

8: Compute the projections: $\widetilde{\mathbf{u}}_{k+1} = \widetilde{\mathbf{u}}_{k+1} - \mathbf{U}_k[\mathbf{M}]_{(1:k,k)}$, $\widetilde{\mathbf{s}}_{k+1} = \widetilde{\mathbf{s}}_{k+1} - \mathbf{S}_k[\mathbf{M}]_{(1:k,k)}$

9: Scale vectors: $\mathbf{s}_{k+1} = \widetilde{\mathbf{s}}_{k+1}/m_{k+1,k}$, $\mathbf{u}_{k+1} = \widetilde{\mathbf{u}}_{k+1}/m_{k+1,k}$, where $m_{k+1,k} = \|\widetilde{\mathbf{s}}_{k+1}\|$

rGK: algorithmic 'sketch'

Require: $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{\Theta}^{(n)} \in \mathbb{R}^{\ell_n \times n}$ $(\ell_n \ll n)$, $\mathbf{\Theta}^{(m)} \in \mathbb{R}^{\ell_m \times m}$ $(\ell_m \ll m)$

1: Initialize & sketch: $\widetilde{\mathbf{u}}_1 = \mathbf{b}$, $\widetilde{\mathbf{s}}_1 = \mathbf{\Theta}^{(m)}\widetilde{\mathbf{u}}_1$; scale: $\mathbf{s}_1 = \widetilde{\mathbf{s}}_1/\beta$, $\mathbf{u}_1 = \widetilde{\mathbf{u}}_1/\beta$, where $\beta = \|\widetilde{\mathbf{s}}_1\|$
2: Compute & sketch: $\widetilde{\mathbf{v}}_1 = \mathbf{A}^\top \mathbf{u}_1$, $\widetilde{\mathbf{p}}_1 = \mathbf{\Theta}^{(n)}\widetilde{\mathbf{v}}_1$
3: Scale: $\mathbf{p}_1 = \widetilde{\mathbf{p}}_1/t_{1,1}$, $\mathbf{v}_1 = \widetilde{\mathbf{v}}_1/t_{1,1}$, where $t_{1,1} = \|\widetilde{\mathbf{p}}_1\|$
4: **for** $k = 1, 2, \dots K$, or a stopping criterion is satisfied **do**
5: **Randomized Gram-Schmidt for constructing U**
6: Get new vector & sketch: $\widetilde{\mathbf{u}}_{k+1} = \mathbf{A}\mathbf{v}_k$, $\widetilde{\mathbf{s}}_{k+1} = \mathbf{\Theta}^{(m)}\widetilde{\mathbf{u}}_{k+1}$
7: Compute $[\mathbf{M}]_{(1:k,k)} = \mathbf{S}_k{}^\top \widetilde{\mathbf{s}}_{k+1}$
8: Compute the projections: $\widetilde{\mathbf{u}}_{k+1} = \widetilde{\mathbf{u}}_{k+1} - \mathbf{U}_k[\mathbf{M}]_{(1:k,k)}$, $\widetilde{\mathbf{s}}_{k+1} = \widetilde{\mathbf{s}}_{k+1} - \mathbf{S}_k[\mathbf{M}]_{(1:k,k)}$
9: Scale vectors: $\mathbf{s}_{k+1} = \widetilde{\mathbf{s}}_{k+1}/m_{k+1,k}$, $\mathbf{u}_{k+1} = \widetilde{\mathbf{u}}_{k+1}/m_{k+1,k}$, where $m_{k+1,k} = \|\widetilde{\mathbf{s}}_{k+1}\|$
10: **Randomized Gram-Schmidt for constructing V**
11: Get new vector & sketch: $\widetilde{\mathbf{v}}_{k+1} = \mathbf{A}^\top \mathbf{u}_{k+1}$, $\widetilde{\mathbf{p}}_{k+1} = \mathbf{\Theta}^{(n)}\widetilde{\mathbf{v}}_{k+1}$
12: Compute $[\mathbf{T}]_{(1:k,k+1)} = \mathbf{P}_k{}^\top \widetilde{\mathbf{p}}_{k+1}$
13: Compute the projections: $\widetilde{\mathbf{v}}_{k+1} = \widetilde{\mathbf{v}}_{k+1} - \mathbf{V}_k[\mathbf{T}]_{(1:k,k+1)}$, $\widetilde{\mathbf{p}}_{k+1} = \widetilde{\mathbf{p}}_{k+1} - \mathbf{P}_k[\mathbf{T}]_{(1:k,k+1)}$
14: Scale vectors: $\mathbf{p}_{k+1} = \widetilde{\mathbf{p}}_{k+1}/t_{k+1,k+1}$, $\mathbf{v}_{k+1} = \widetilde{\mathbf{v}}_{k+1}/t_{k+1,k+1}$, where $t_{k+1,k+1} = \|\widetilde{\mathbf{p}}_{k+1}\|$
15: **end for**

rGK-based iterative solvers

$$\mathbf{x}_k = \mathbf{V}_k \mathbf{z}_k \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})$$

- **randomized LSQR (rLSQR)**

$$\mathbf{z}_k = \arg\min_{\mathbf{z}\in\mathbb{R}^k} \|\mathbf{M}_k \mathbf{z} - \beta \mathbf{e}_1\|, \quad \text{with} \quad \beta = \|\boldsymbol{\Theta}^{(m)} \mathbf{b}\|$$

- **randomized CGLS (rCGLS)**

$$\widetilde{\mathbf{T}}_{k+1} \mathbf{M}_k \mathbf{z}_k = \beta t_{1,1} \mathbf{e}_1, \quad \text{with} \quad \widetilde{\mathbf{T}}_{k+1} = [\mathbf{I}_k, \mathbf{0}] \mathbf{T}_{k+1} \in \mathbb{R}^{k\times(k+1)},\ t_{1,1} = [\mathbf{T}_{k+1}]_{1,1}$$

- **randomized LSMR (rLSMR)**

$$\mathbf{z}_k = \arg\min_{\mathbf{z}\in\mathbb{R}^k} \|\mathbf{T}_{k+1} \mathbf{M}_k \mathbf{z} - \beta t_{1,1} \mathbf{e}_1\|$$

rGK-based iterative solvers - optimality properties

$$\mathbf{x}_k = \mathbf{V}_k \mathbf{z}_k \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})$$

- **randomized LSQR (rLSQR)**

$$\mathbf{x}_k = \arg \min_{\mathbf{x} \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})} \|\mathbf{\Theta}^{(m)}(\mathbf{A}\mathbf{x} - \mathbf{b})\|$$

- **randomized CGLS (rCGLS)**

N/A

- **randomized LSMR (rLSMR)**

$$\mathbf{x}_k = \arg \min_{\mathbf{x} \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})} \|\mathbf{\Theta}^{(n)}\mathbf{A}^\top(\mathbf{A}\mathbf{x} - \mathbf{b})\|$$

rGK-based iterative solvers - characterizations

$$\mathbf{x}_k = \mathbf{V}_k \mathbf{z}_k \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})$$

- **randomized LSQR (rLSQR)**

$$\mathbf{x}_k \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b}) \quad \text{and} \quad \mathbf{r}_k \perp_{\boldsymbol{\Theta}^{(m)}} \mathbf{A}\mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})$$

- **randomized CGLS (rCGLS)**

$$\mathbf{x}_k \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b}) \quad \text{and} \quad \mathbf{A}^\top \mathbf{r}_k \perp_{\boldsymbol{\Theta}^{(n)}} \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})$$

- **randomized LSMR (rLSMR)**

$$\mathbf{x}_k \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b}) \quad \text{and} \quad \mathbf{A}^\top \mathbf{r}_k \perp_{\boldsymbol{\Theta}^{(n)}} \mathbf{A}^\top \mathbf{A}\mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})$$

rGK-based iterative solvers - other characterizations

$$\mathbf{x}_k = \mathbf{V}_k \mathbf{z}_k \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})$$

- **randomized LSQR (rLSQR)**

Associated normal equations:

$$\mathbf{A}^\top (\boldsymbol{\Theta}^{(m)})^\top \boldsymbol{\Theta}^{(m)} \mathbf{A}\mathbf{x} = \mathbf{A}^\top (\boldsymbol{\Theta}^{(m)})^\top \boldsymbol{\Theta}^{(m)} \mathbf{b}$$

Conditions:

$$\mathbf{x}_k \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b}) \quad \text{and} \quad \mathbf{A}^\top (\boldsymbol{\Theta}^{(m)})^\top \boldsymbol{\Theta}^{(m)} \mathbf{A}\mathbf{r}_k \perp \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})$$

- **randomized CGLS (rCGLS)**

Associated normal equations:

$$(\boldsymbol{\Theta}^{(n)})^\top \boldsymbol{\Theta}^{(n)} \mathbf{A}^\top \mathbf{A}\mathbf{x} = (\boldsymbol{\Theta}^{(n)})^\top \boldsymbol{\Theta}^{(n)} \mathbf{A}^\top \mathbf{b}$$

Conditions:

$$\mathbf{x}_k \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b}) \quad \text{and} \quad (\boldsymbol{\Theta}^{(n)})^\top \boldsymbol{\Theta}^{(n)} \mathbf{A}^\top \mathbf{A}\mathbf{r}_k \perp \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})$$

Hybrid rGK-based methods

$$\mathbf{x}_k = \mathbf{V}_k \mathbf{z}_k \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})$$

- **hybrid rLSQR**

$$\mathbf{z}_k = \arg\min_{\mathbf{z}\in\mathbb{R}^k} \|\mathbf{M}_k \mathbf{z} - \beta \mathbf{e}_1\|^2 + \lambda^2 \|\mathbf{z}\|^2, \quad \text{with} \quad \beta = \|\boldsymbol{\Theta}^{(m)} \mathbf{b}\|$$

- **hybrid rCGLS**

$$(\widetilde{\mathbf{T}}_{k+1} \mathbf{M}_k + \lambda^2 \mathbf{I}) \mathbf{z}_k = \beta t_{1,1} \mathbf{e}_1, \quad \text{with} \quad \widetilde{\mathbf{T}}_{k+1} = [\mathbf{I}_k, \mathbf{0}] \mathbf{T}_{k+1} \in \mathbb{R}^{k\times(k+1)}$$

- **hybrid rLSMR**

$$\mathbf{z}_k = \arg\min_{\mathbf{z}\in\mathbb{R}^k} \|\mathbf{T}_{k+1} \mathbf{M}_k \mathbf{z} - \beta t_{1,1} \mathbf{e}_1\|^2 + \lambda^2 \|\mathbf{x}\|^2$$

Hybrid rGK-based methods - characterizations

$$\mathbf{x}_k = \mathbf{V}_k \mathbf{z}_k \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})$$

- **hybrid rLSQR**

$$\mathbf{x}_k = \arg \min_{\mathbf{x} \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})} \|\boldsymbol{\Theta}^{(m)}(\mathbf{A}\mathbf{x} - \mathbf{b})\|^2 + \lambda^2 \|\boldsymbol{\Theta}^{(n)} \mathbf{x}\|^2$$

- **hybrid rCGLS**

$$\mathbf{x}_k \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b}) \quad \text{and} \quad \mathbf{A}_\lambda^\top \mathbf{r}_{\lambda,k} \perp_{\boldsymbol{\Theta}^{(n)}} \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b}), \qquad \begin{array}{l} \mathbf{A}_\lambda = [\mathbf{A}^\top, \lambda \mathbf{I}]^\top \\ \mathbf{r}_{\lambda,k} = [(\mathbf{b} - \mathbf{A}\mathbf{x}_k)^\top, \lambda \mathbf{x}_k^\top] \end{array}$$

- **hybrid rLSMR**

$$\mathbf{x}_k = \arg \min_{\mathbf{x} \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})} \|\boldsymbol{\Theta}^{(n)} \mathbf{A}^\top (\mathbf{A}\mathbf{x} - \mathbf{b})\|^2 + \lambda^2 \|\boldsymbol{\Theta}^{(n)} \mathbf{z}\|^2$$

Hybrid rGK-based methods

$$\mathbf{x}_k = \mathbf{V}_k \mathbf{z}_k \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})$$

- **hybrid rLSQR**

$$\mathbf{x}_k = \arg \min_{\mathbf{x} \in \mathcal{K}_k(\mathbf{A}^\top \mathbf{A}, \mathbf{A}^\top \mathbf{b})} \left\| \begin{bmatrix} \mathbf{\Theta}^{(m)} & \\ & \mathbf{\Theta}^{(n)} \end{bmatrix} \left(\begin{bmatrix} \mathbf{A} \\ \lambda \mathbf{I} \end{bmatrix} \mathbf{x} - \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix} \right) \right\|^2$$

- **rLSQR** applied **to** the **Tikhonov**-regularized problem

$$\mathbf{x}_k = \arg \min_{\mathbf{x} \in \mathcal{K}_k((\mathbf{A}^\top \mathbf{A} + \lambda^2 \mathbf{I}), \mathbf{A}^\top \mathbf{b})} \left\| \mathbf{\Theta}^{(m+n)} \left(\begin{bmatrix} \mathbf{A} \\ \lambda \mathbf{I} \end{bmatrix} \mathbf{x} - \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix} \right) \right\|^2$$

Hybrid rGK-based methods

$$\mathbf{x}_k = \mathbf{V}_k \mathbf{z}_k \in \mathcal{K}_k(\mathbf{A}^\top\mathbf{A}, \mathbf{A}^\top\mathbf{b})$$

- **hybrid rLSQR**

$$\mathbf{x}_k = \arg\min_{\mathbf{x}\in\mathcal{K}_k(\mathbf{A}^\top\mathbf{A},\mathbf{A}^\top\mathbf{b})} \left\| \begin{bmatrix} \mathbf{\Theta}^{(m)} & \\ & \mathbf{\Theta}^{(n)} \end{bmatrix} \left(\begin{bmatrix} \mathbf{A} \\ \lambda\mathbf{I} \end{bmatrix} \mathbf{x} - \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix} \right) \right\|^2$$

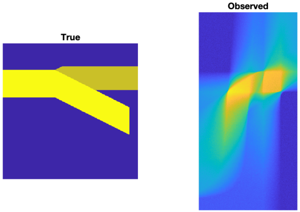
- **rLSQR** applied **to** the **Tikhonov**-regularized problem

$$\mathbf{x}_k = \arg\min_{\mathbf{x}\in\mathcal{K}_k((\mathbf{A}^\top\mathbf{A}+\lambda^2\mathbf{I}),\mathbf{A}^\top\mathbf{b})} \left\| \mathbf{\Theta}^{(m+n)} \left(\begin{bmatrix} \mathbf{A} \\ \lambda\mathbf{I} \end{bmatrix} \mathbf{x} - \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix} \right) \right\|^2$$

  - closer to ‘sketch-and-solve’ approaches
  - no natural way to select λ adaptively

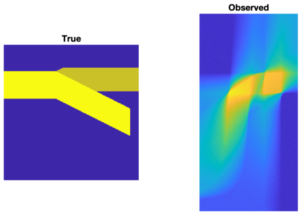
Numerical Results: seismic tomography

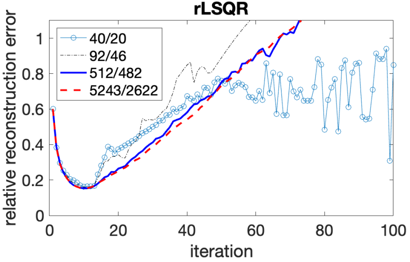
- $\mathbf{A} \in \mathbb{R}^{131{,}072 \times 65{,}536}$
- $\mathbf{b} \in \mathbb{R}^{131{,}072}$ with 4% noise
- $\boldsymbol{\Theta}^{(n)} \in \mathbb{R}^{\ell_n \times n}$, $\boldsymbol{\Theta}^{(m)} \in \mathbb{R}^{\ell_m \times n}$ Subsampled Randomized Hadamard Transform (SRHT)



Numerical Results: seismic tomography

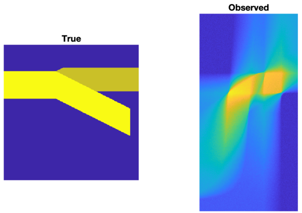
- $\mathbf{A} \in \mathbb{R}^{131,072 \times 65,536}$
- $\mathbf{b} \in \mathbb{R}^{131,072}$ with 4% noise
- $\boldsymbol{\Theta}^{(n)} \in \mathbb{R}^{\ell_n \times n}$, $\boldsymbol{\Theta}^{(m)} \in \mathbb{R}^{\ell_m \times n}$ Subsampled Randomized Hadamard Transform (SRHT)
- Legend values: ℓ_m / ℓ_n

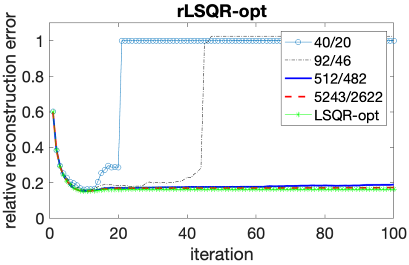




Numerical Results: seismic tomography

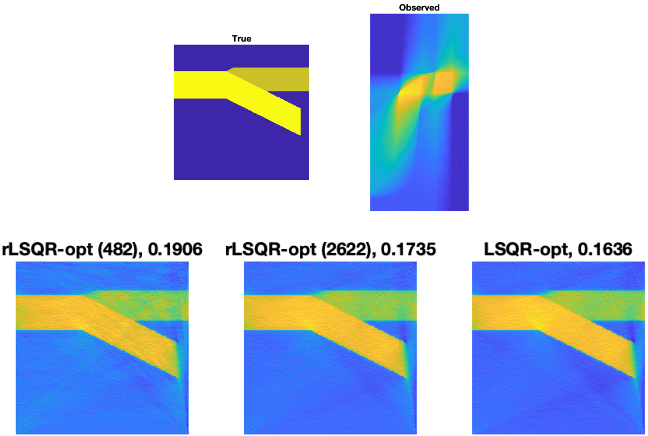
- $\mathbf{A} \in \mathbb{R}^{131{,}072 \times 65{,}536}$
- $\mathbf{b} \in \mathbb{R}^{131{,}072}$ with 4% noise
- $\boldsymbol{\Theta}^{(n)} \in \mathbb{R}^{\ell_n \times n}$, $\boldsymbol{\Theta}^{(m)} \in \mathbb{R}^{\ell_m \times n}$
  Subsampled Randomized Hadamard Transform (SRHT)
- Legend values: ℓ_m/ℓ_n





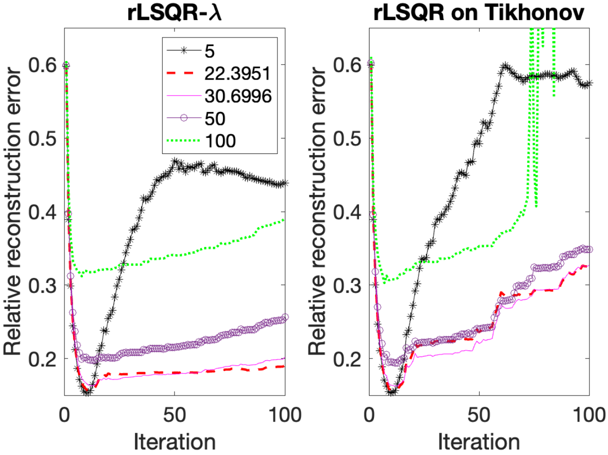
Numerical Results: seismic tomography

- $\mathbf{A} \in \mathbb{R}^{131,072 \times 65,536}$
- $\mathbf{b} \in \mathbb{R}^{131,072}$ with 4% noise
- $\mathbf{\Theta}^{(n)} \in \mathbb{R}^{\ell_n \times n}$, $\mathbf{\Theta}^{(m)} \in \mathbb{R}^{\ell_m \times n}$ Subsampled Randomized Hadamard Transform (SRHT)



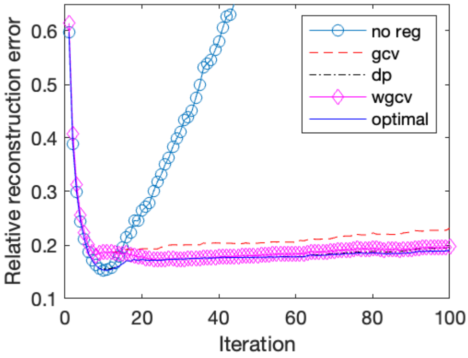
Seismic tomography: hybrid rLSQR vs. rLSQR for Tikhonov

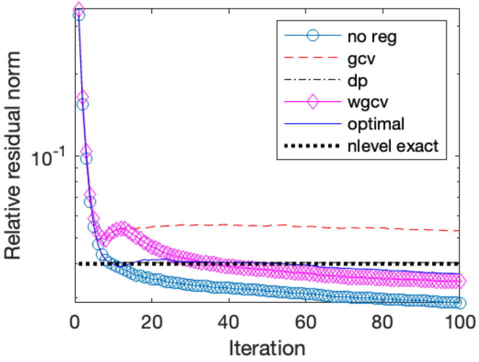
values of (fixed) λ noted in the legend



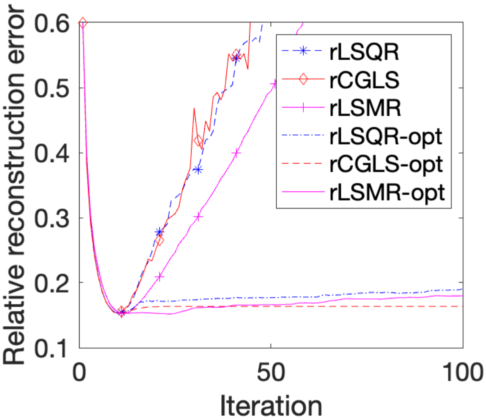
Seismic tomography: hybrid rLSQR with adaptively selected λ

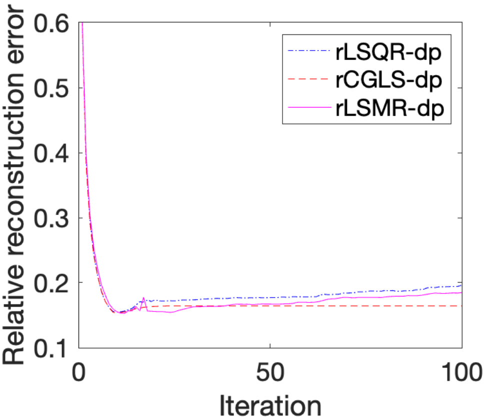
regularization parameter choice strategies noted in the legend





Seismic tomography: different hybrid rGK solvers





Conclusions and outlook

Take-home message

- developed a range of randomized Krylov solvers, based on sketched inner products:
- valid alternative to std Krylov solvers, with proper sketching size
- to sketch efficiently, the residuals should be small;
  if they are not, try to change the problem to make them small.

J. Chung and S. G. *Randomized Krylov methods for inverse problems*. `arXiv:2508.20269`

Conclusions and outlook

Take-home message

- developed a range of randomized Krylov solvers, based on sketched inner products:
- valid alternative to std Krylov solvers, with proper sketching size
- to sketch efficiently, the residuals should be small;
  if they are not, try to change the problem to make them small.

J. Chung and S. G. *Randomized Krylov methods for inverse problems*. `arXiv:2508.20269`

Future investigations

- different SVD approximations for different projection methods
- partial recurrences (decay of off diagonal entries): factorizations in expectation?
- memory-efficient implementations for low-rank regularization

Conclusions and outlook

Take-home message

- developed a range of randomized Krylov solvers, based on sketched inner products:
- valid alternative to std Krylov solvers, with proper sketching size
- to sketch efficiently, the residuals should be small;
  if they are not, try to change the problem to make them small.

J. Chung and S. G. *Randomized Krylov methods for inverse problems*. `arXiv:2508.20269`

Future investigations

- different SVD approximations for different projection methods
- partial recurrences (decay of off diagonal entries): factorizations in expectation?
- memory-efficient implementations for low-rank regularization

Thank you for your attention!

Seismic example (smaller): SVD Approximations