Randomized TSVD Algorithms

Joel A. Tropp

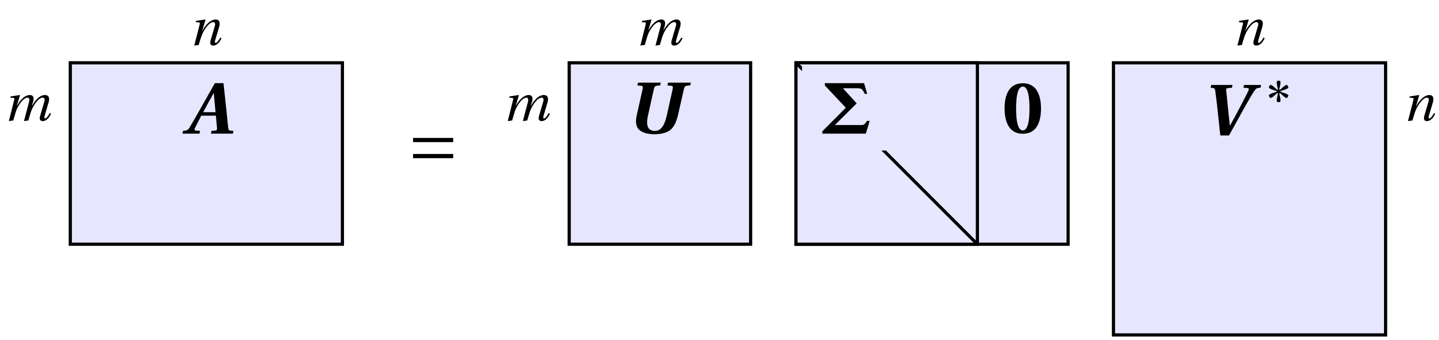
Computing + Mathematical Sciences
California Institute of Technology
jtropp@cms.caltech.edu

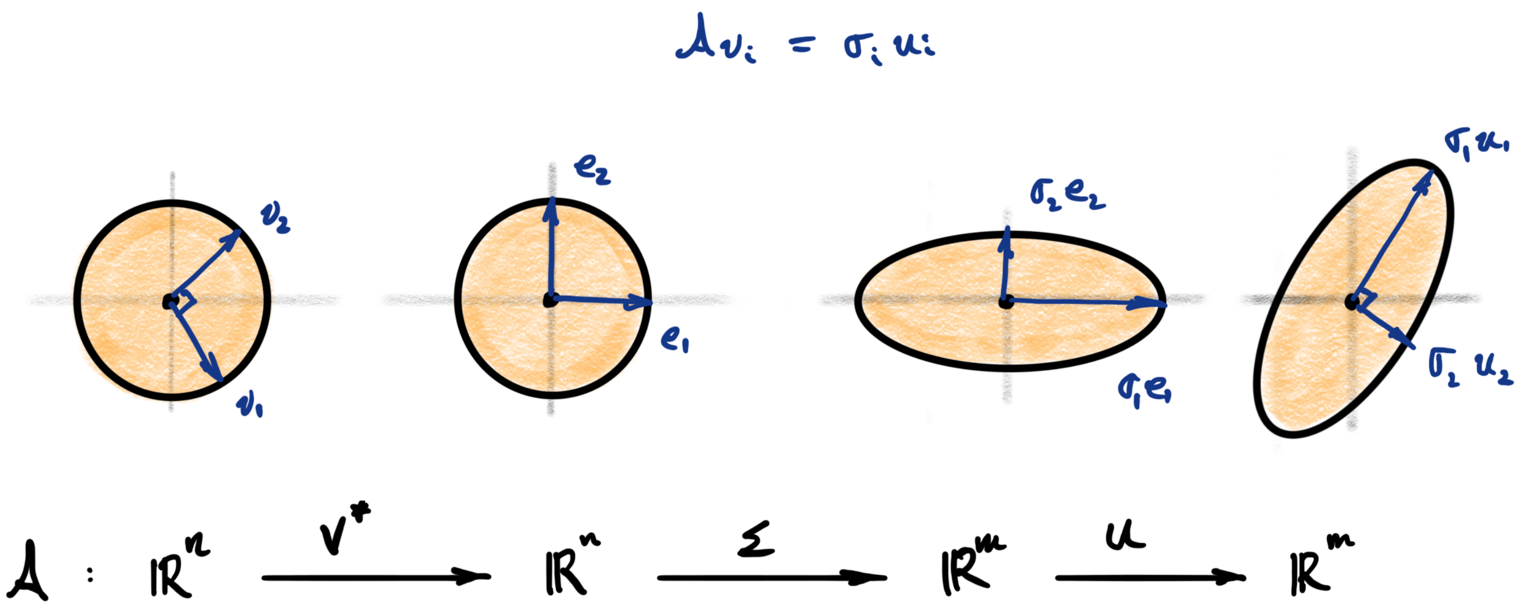
Coauthors:
Nathan Halko; Gunnar Martinsson (UT-Austin);
Rob Webber (Caltech); Yuji Nakatsukasa (Oxford);
Alp Yurtsever (Umeå); Madeleine Udell (Cornell); Volkan Cevher (EPFL);
Katie Bouman, Aviad Levis(Caltech); Charles Gammie, David Lee (UIUC)

Research supported by ONR, AFOSR, NSF, DARPA, Sloan, and Moore.

The Famous Truncated SVD

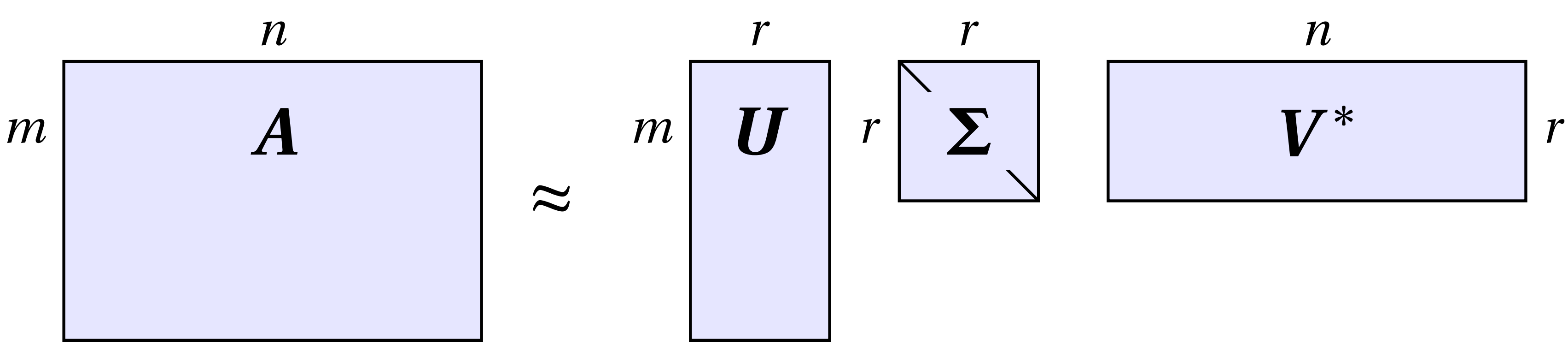
The Singular Value Decomposition





❧ $\boldsymbol{U}, \boldsymbol{V}$ are orthogonal and $\boldsymbol{\Sigma} = \operatorname{diag}(\sigma_1, \sigma_2, \sigma_3, \dots)$ is positive, decreasing

Truncated Singular Value Decomposition (TSVD)



- $\boldsymbol{U}, \boldsymbol{V}$ have orthonormal columns and $\boldsymbol{\Sigma}$ is positive, diagonal, decreasing
- **Eckart–Young:** r-truncated SVD = best rank-r Frobenius-norm approximation
- Approximately $r(m+n)$ degrees of freedom

Applications:

- Least-squares computations (linear regression)
- Principal component analysis (orthogonal regression; total least squares)
- Approximation, summarization, data reduction, visualization, ...

Randomized
Matrix Computations

What's Wrong with Classical TSVD Algorithms?

- Nothing... when the matrices are small

Climate Change

- Medium- to large-scale data (Gigabytes+)
- New architectures (multi-core, distributed, data centers, ...)
- New data presentations (off-core, dynamic, streaming)

The Role of Randomness

- Randomness is becoming a core tool for matrix computations
- Can solve problems that are impossible without randomness
- Can organize computations so they are cheaper (multiplication rich)
- Careful implementation and analysis remain essential!

- **Today:** Practical randomized algorithms for TSVD computations

History of Randomized TSVD Algorithms

Classical Numerical Linear Algebra

- Random initialization for iterative methods (conventional wisdom)
- Guarantees for maximum eigenvalue (Dixon 1983; Kucziński & Woźniakowski 1992; ...)

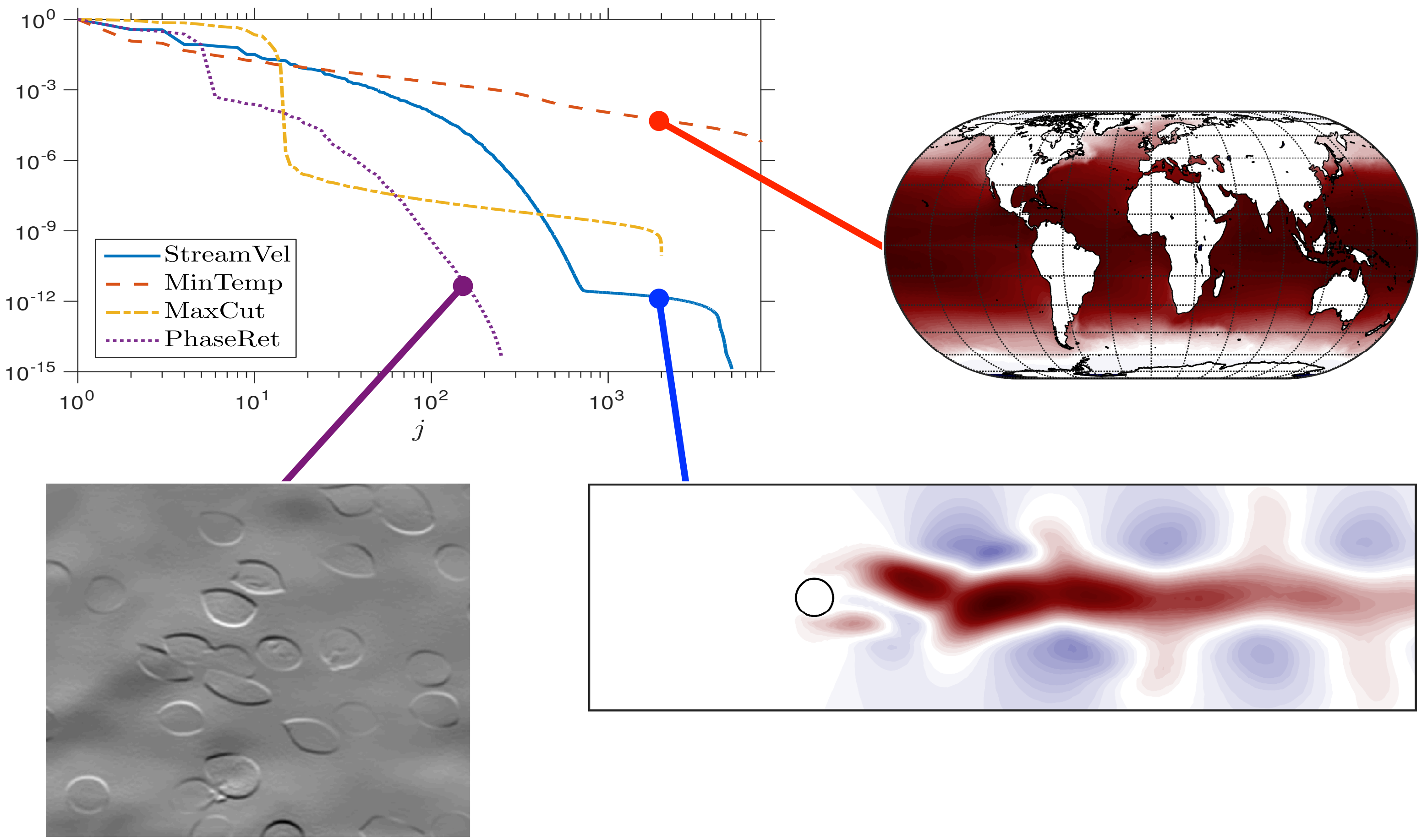
Modern Numerical Linear Algebra

- Randomized low-rank matrix approximation (Martinsson, Rokhlin, Tygert 2004)
- One-pass matrix approximation (Woolfe, Liberty, Rokhlin, Tygert 2007)
- **Randomized SVD framework, algorithms, and analysis** (Halko, Martinsson, Tropp 2008–2011)
- **Randomized block Krylov methods** (Rokhlin & Tygert 2008; Halko et al. 2011; Tropp et al. 2017–2021)
- **Practical streaming SVD algorithms** (Tropp, Yurtsever, Udell, Cevher 2016–2019)
- Fast randomized algorithms for linear systems and eigenvalue problems (Nakatsukasa & Tropp 2021)
- Randomized block Krylov framework, algorithms, and analysis (Tropp & Webber, forthcoming)

CS Theory

- Gaussian dimension reduction (Johnson & Lindenstrauss 1984; Indyk & Motwani 1998)
- Randomized algorithms for LSI (Frieze, Kannan, Papadimitriou, Vempala 1998)
- Randomized linear algebra foundations (Drineas, Kannan, Mahoney 2004)
- Sketch-and-solve framework (Sarlós 2006)
- Streaming linear algebra foundations (Clarkson & Woodruff 2009)
- Qualitative analysis of randomized block Krylov methods (Musco & Musco 2015)

Spectral Decay in Scientific Data



Theme: Tradeoff between spectral decay and computational effort for TSVD

Randomized SVD Framework

[HMT11] Randomized SVD Framework

- Let $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ be an input matrix
- Let $\boldsymbol{B} \in \mathbb{R}^{n \times d}$ be a (random) test matrix

1. **Rangefinder:** Find orthobasis $\boldsymbol{Q} \in \mathbb{R}^{m \times d}$ for range of input matrix:

$$\boldsymbol{Q} = \texttt{orth}(\boldsymbol{AB})$$

2. **Approximation:** Variational formulation of matrix approximation:

$$\text{minimize}_{\boldsymbol{M} \in \mathbb{R}^{d \times n}} \quad \|\boldsymbol{A} - \boldsymbol{QM}\|_{\text{F}}^2 \qquad \text{Solution: } \boldsymbol{M}_\star = \boldsymbol{Q}^* \boldsymbol{A}$$

3. **SVD:** Write rank-d approximation $\widehat{\boldsymbol{A}}$ as an SVD:

$$\widehat{\boldsymbol{A}} = \boldsymbol{QM}_\star = \boldsymbol{Q}(\boldsymbol{U\Sigma V}^*) = (\boldsymbol{QU})\boldsymbol{\Sigma V}^*$$

Sources: Halko et al. 2008–2011, Nakatsukasa & Tropp 2021, Tropp & Webber 2021.

Randomized SVD Prototype

Randomized SVD Framework

Input: Input matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ and (random) test matrix $\boldsymbol{B} \in \mathbb{R}^{n \times d}$

Output: Approximate rank-d SVD in factored form: $\widehat{\boldsymbol{A}} = \boldsymbol{U\Sigma V}^*$

1. Form range sketch: $\boldsymbol{Y} = \boldsymbol{AB}$
2. Orthogonalize columns: $\boldsymbol{Q} = \texttt{orth}(\boldsymbol{Y})$
3. Compress input matrix: $\boldsymbol{M} = \boldsymbol{Q}^* \boldsymbol{A}$
4. Compute small dense SVD: $\boldsymbol{M} = \widehat{\boldsymbol{U}}\boldsymbol{\Sigma V}^*$
5. Consolidate left unitary factor: $\boldsymbol{U} = \boldsymbol{Q}\widehat{\boldsymbol{U}}$

Cost: Two $m \times n \times d$ matmuls with $\boldsymbol{A}$ and $\boldsymbol{A}^*$ plus $(m+n)d^2$ arithmetic

Benefits: Most arithmetic in matmuls (fast!) and very robust

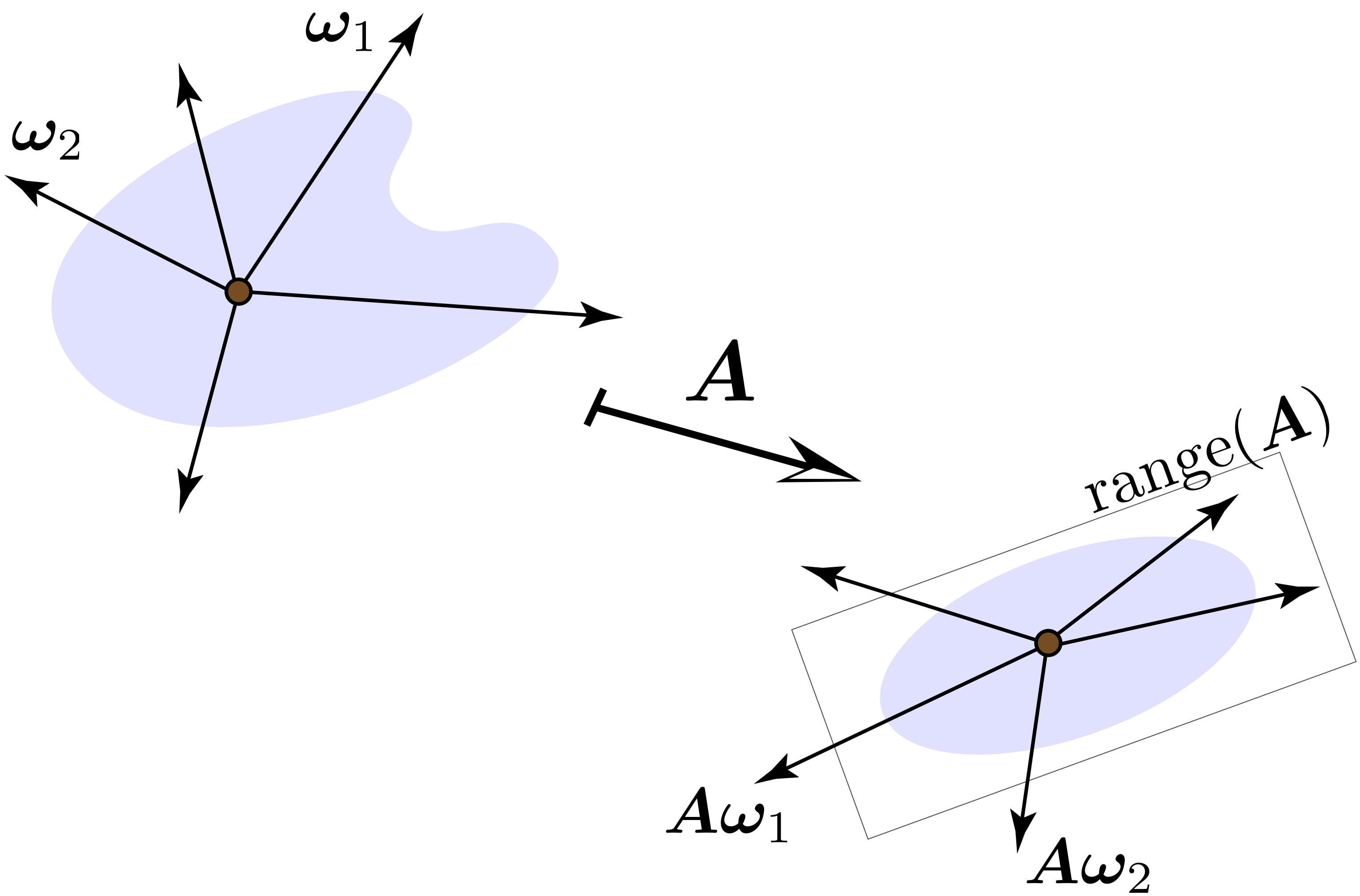
Questions:

- What test matrix?
- Error bounds?

Theme: Tradeoff between spectral decay and computational effort

The HMT11 Randomized TSVD

Crazy Idea: Totally Random Test Matrix



[HMT11] Randomized SVD

Theorem (HMT 2011). **Assume**

- *Input matrix* $\boldsymbol{A} \in \mathbb{R}^{m \times n}$
- *Test matrix* $\boldsymbol{B} \in \mathbb{R}^{m \times d}$ *is standard normal*
- *Form range sketch:* $\boldsymbol{Q} = \texttt{orth}(\boldsymbol{AB})$
- *Compute rank-d approximation:* $\widehat{\boldsymbol{A}} = \boldsymbol{QQ}^* \boldsymbol{A}$

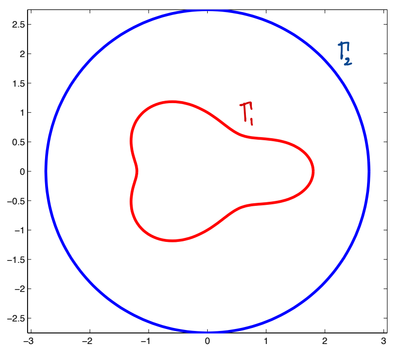
For $d \geq 1 + k/\varepsilon$, *this randomized SVD algorithm guarantees*

$$\mathbb{E} \left\| \boldsymbol{A} - \widehat{\boldsymbol{A}} \right\|_{\mathrm{F}}^2 \quad \leq \quad (1+\varepsilon) \cdot \left\| \boldsymbol{A} - [\![\boldsymbol{A}]\!]_k \right\|_{\mathrm{F}}^2$$

- **Key fact:** Error is small when spectrum of $\boldsymbol{A}$ decays quickly
- Probability of a much larger error is negligible

Source: Halko et al. 2011, §10.2.

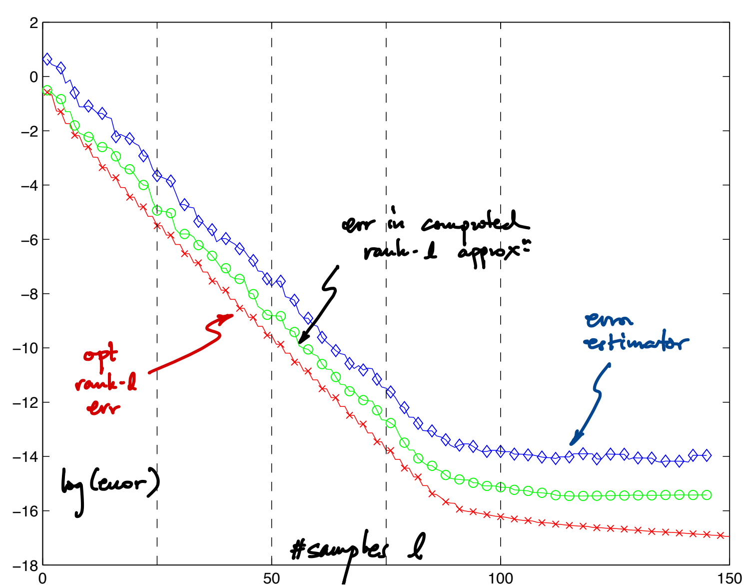
Example: Discretized Integral Operator

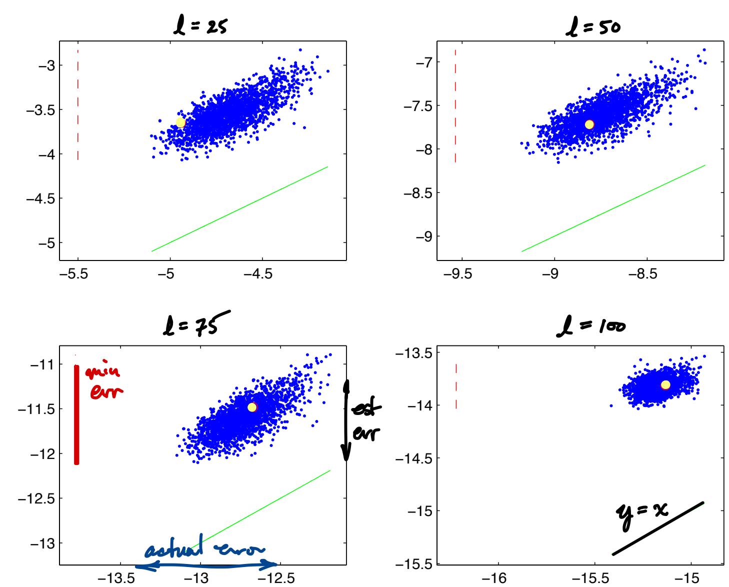


Matrix $\boldsymbol{A}$ is a 200×200 discretization of

$$[S\sigma](\boldsymbol{x}) = \text{const} \cdot \int_{\Gamma_1} \log \|\boldsymbol{x} - \boldsymbol{y}\| \, \sigma(\boldsymbol{y}) \, \mathrm{d}A(\boldsymbol{y})$$

for $\boldsymbol{x} \in \Gamma_2$





Source: Halko et al. 2011, Figures 7.1–7.3.

Randomized Subspace Iteration

Randomized Subspace Iteration (RSI)

- **Issue:** What if the spectrum decays more slowly?
- **RSI idea:** Adapt test matrix to input matrix by powering:
  - Draw standard normal matrix $\boldsymbol{\Omega} \in \mathbb{R}^{n \times d}$
  - (Carefully) form test matrix $\boldsymbol{B} = (\boldsymbol{A}\boldsymbol{A}^*)^q \boldsymbol{A}\boldsymbol{\Omega}$
- **Equivalent:** Apply [HMT11] to $(\boldsymbol{A}\boldsymbol{A}^*)^q \boldsymbol{A}$. Enhances spectral decay!
- **RSI cost:** $2q+2$ matmuls plus $(m+n)qd^2$ arithmetic
- **Spectral-norm error:** For $d = k+2$ and $q \lesssim (\log n)/\varepsilon$,

$$\mathbb{E}\left\| \boldsymbol{A} - \widehat{\boldsymbol{A}} \right\| \;\leq\; (1+\varepsilon) \cdot \left\| \boldsymbol{A} - [\![\boldsymbol{A}]\!]_k \right\|$$

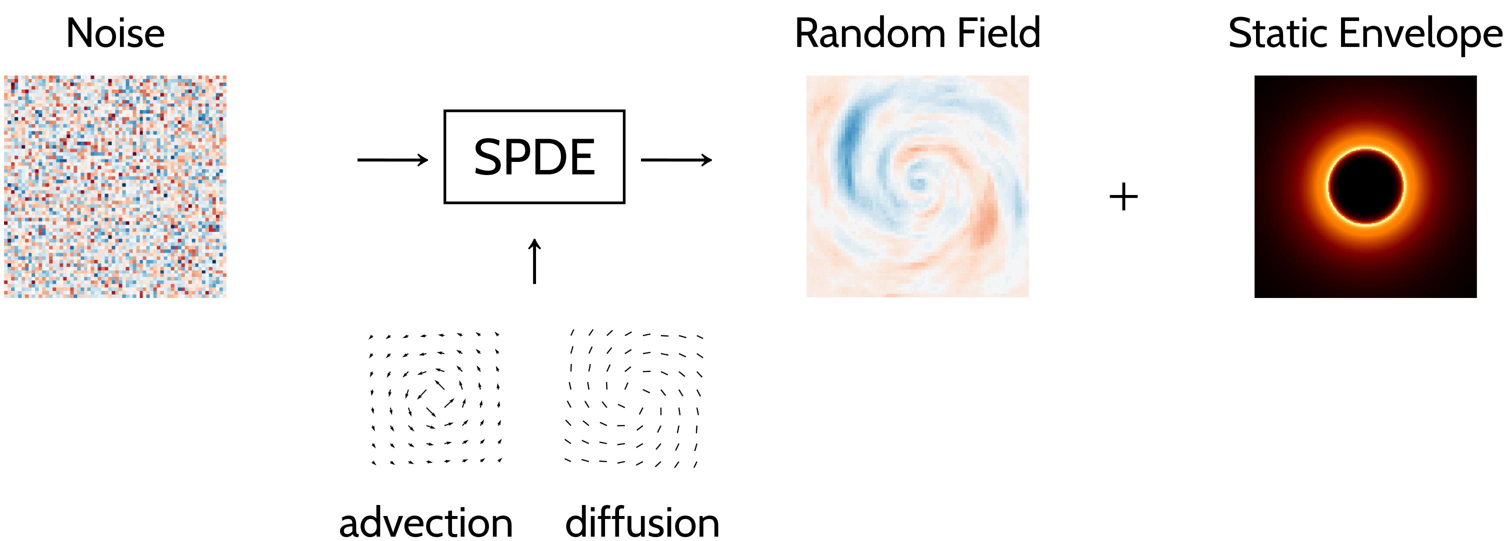
- **In practice:** $q = 2$ or $q = 3$ often suffices. Not asymptotic!

Source: Halko et al. 2011, Algorithms 4.4 and 5.1, Corollary 10.10.

Example: Event Horizon Telescope

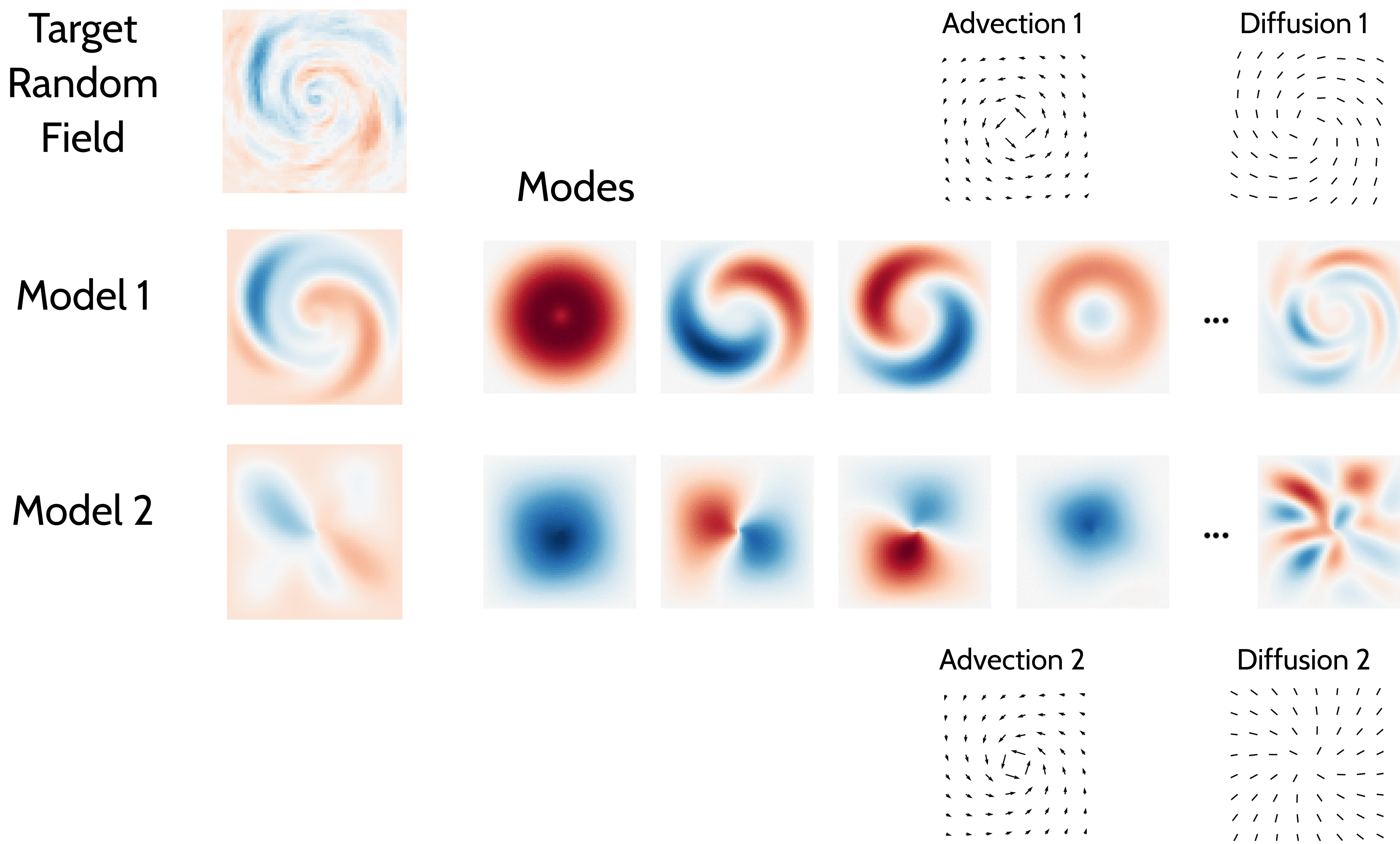
M87

Sgr A* (???)



Source: Bouman, Gammie, Lee, Levis, Tropp 2021.

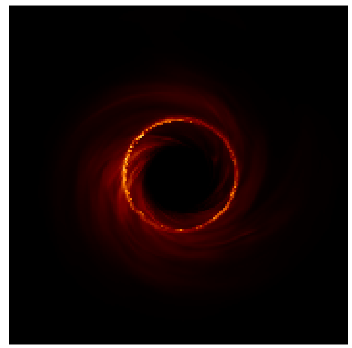
Modal Decomposition of Dynamics



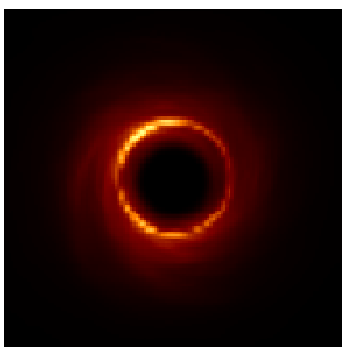
Source: Bouman, Gammie, Lee, Levis, Tropp 2021.

Recovery of Black Hole Dynamics

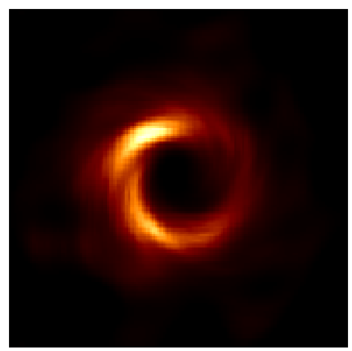
Simulated Black Hole



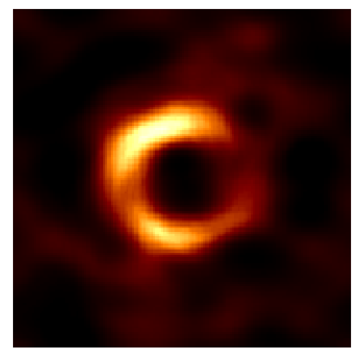
Noisy Full Data



EHT 2025



EHT 2017



Source: Bouman, Gammie, Lee, Levis, Tropp 2021.

Randomized
Block Krylov

Randomized Block Krylov Methods

- **Issue:** What if the spectrum does not really decay?
- **BK idea:** Adapt test matrix to input matrix by including all powers:
  - Draw standard normal matrix $\mathbf{\Omega} \in \mathbb{R}^{n \times d}$
  - (Carefully) form test matrix $\boldsymbol{B} = [\boldsymbol{A\Omega} \quad (\boldsymbol{AA}^*)\boldsymbol{A\Omega} \quad \dots \quad (\boldsymbol{AA}^*)^q \boldsymbol{A\Omega}]$
- **BK cost:** $2q+2$ matmuls plus $(m+n)(qd)^2$ arithmetic
- **Spectral-norm error:** For $d = k+2$ and $q \lesssim (\log n)/\sqrt{\varepsilon}$,

$$\mathbb{E} \left\| \boldsymbol{A} - \widehat{\boldsymbol{A}} \right\| \leq (1+\varepsilon) \cdot \left\| \boldsymbol{A} - [\![\boldsymbol{A}]\!]_k \right\|$$

- **In practice:** $q = 2$ or $q = 3$ often suffices. Not asymptotic!
- **Loss:** Extra storage + orthogonalization

Sources: Rokhlin & Tygert 2008; Halko, Martinsson, Shkolnisky, Tygert 2011; Musco & Musco 2015; Tropp 2018–2021; Martinsson & Tropp 2020; Tropp & Webber 2021.

Example: Geophysical Imaging

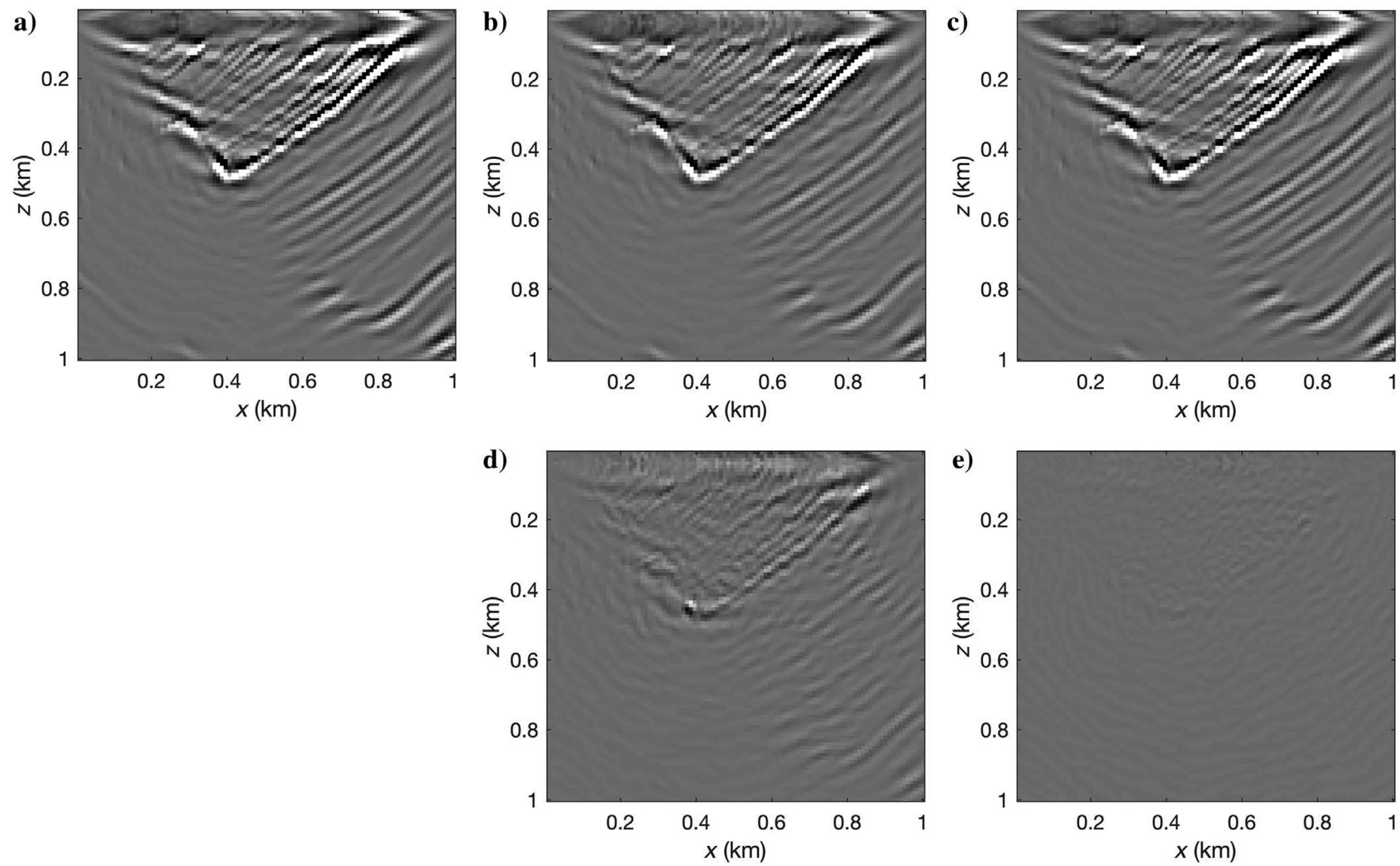


Figure 6. Comparison between RTMs obtained by carrying out (a) conventional SVD, (b) the rSVD, and (c) BKI for $q = 1$. Difference plots with respect to the conventional SVD are included in (d and e). These plots show that the image obtained from a factorization based on BKI is significantly more accurate. Only the first eight singular values are used, i.e., $n_p = 8 \ll n_s$ with $n_s = 100$.

Source: Yang, Graff, Kumar, Herrmann 2021, Figure 6.

Sketchy TSVD Algorithms

Sketchy SVD Framework

- Let $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ be an input matrix
- Let $\boldsymbol{B} \in \mathbb{R}^{n \times d}$ be a (random) test matrix

1. **Rangefinder:** Find a basis $\boldsymbol{Q} \in \mathbb{R}^{m \times d}$ for the range of the input matrix:

$$\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{B} \qquad \text{and} \qquad \boldsymbol{Q} = \texttt{orth}(\boldsymbol{Y})$$

2. **Sketching:** For random $\boldsymbol{S} \in \mathbb{R}^{s \times m}$, compressed matrix approximation:

$$\text{minimize}_{\boldsymbol{M} \in \mathbb{R}^{d \times n}} \quad \|\boldsymbol{S}(\boldsymbol{A} - \boldsymbol{Q}\boldsymbol{M})\|_{\mathrm{F}}^2$$

Solution: $\boldsymbol{M}_\star = (\boldsymbol{S}\boldsymbol{Q})^\dagger (\boldsymbol{S}\boldsymbol{A})$

3. **SVD:** Write rank-d approximation $\widehat{\boldsymbol{A}}$ as an SVD:

$$\widehat{\boldsymbol{A}} = \boldsymbol{Q}\boldsymbol{M}_\star = \boldsymbol{Q}(\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^*) = (\boldsymbol{Q}\boldsymbol{U})\boldsymbol{\Sigma}\boldsymbol{V}^*$$

Sources: Woolfe et al. 2008; Halko et al. 2008–2011; Clarkson & Woodruff 2009; Woodruff 2014; Tropp et al. 2017–2020; Nakatsukasa 2020; Nakatsukasa & Tropp 2021; Tropp & Webber 2021.

Example: One-Pass SVD

Basic Sketchy SVD

Input: Input matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$
Output: Approximate rank-d SVD in factored form: $\widehat{\boldsymbol{A}} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^*$

1. Form range sketch: $\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{B}$ for random $\boldsymbol{B} \in \mathbb{R}^{n \times d}$
2. Form co-range sketch: $\boldsymbol{Z} = \boldsymbol{S}\boldsymbol{A}$ for random $\boldsymbol{S} \in \mathbb{R}^{s \times m}$
3. Orthogonalize columns of range sketch: $\boldsymbol{Q} = \texttt{orth}(\boldsymbol{Y})$
4. Compute matrix approximation: $\boldsymbol{M} = (\boldsymbol{S}\boldsymbol{Q})^\dagger \boldsymbol{Z}$
5. Compute small dense SVD: $\boldsymbol{M} = \widehat{\boldsymbol{U}}\boldsymbol{\Sigma}\boldsymbol{V}^*$
6. Consolidate left unitary factor: $\boldsymbol{U} = \boldsymbol{Q}\widehat{\boldsymbol{U}}$

Cost: Two matmuls plus $(m+n)d^2$ arithmetic

Benefits: Takes linear measurements of matrix, only requires one pass

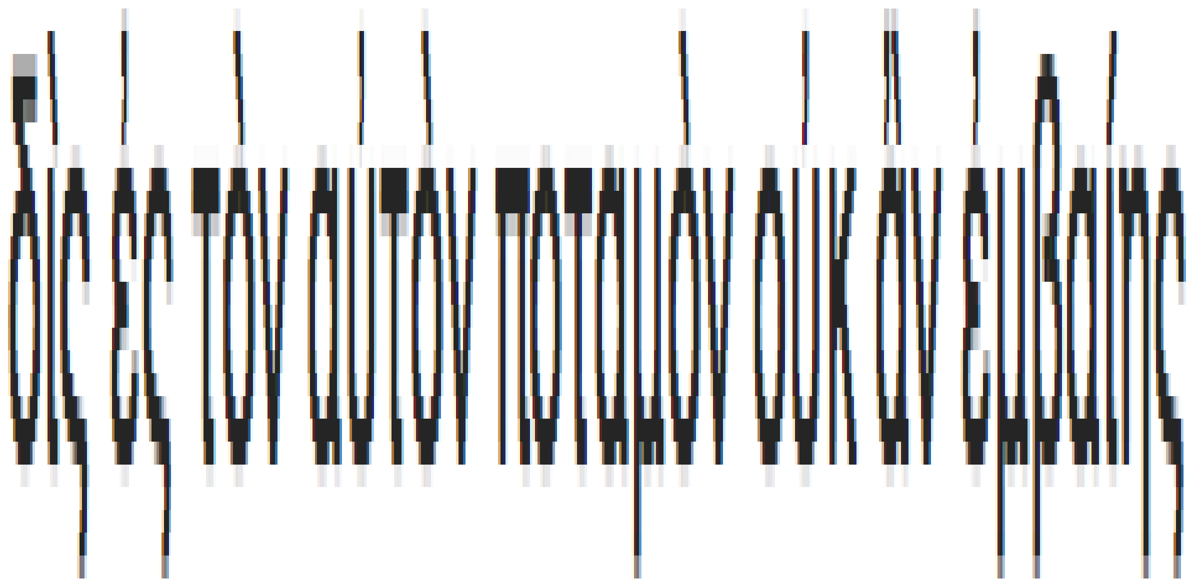
Frobenius-norm error: For $d = 1 + 2k$ and $s = 2 + 4k$,

$$\mathbb{E} \left\| \boldsymbol{A} - \widehat{\boldsymbol{A}} \right\|_{\mathrm{F}} \leq 4 \cdot \left\| \boldsymbol{A} - [\![\boldsymbol{A}]\!]_k \right\|_{\mathrm{F}}$$

Loss: Requires significant spectral decay

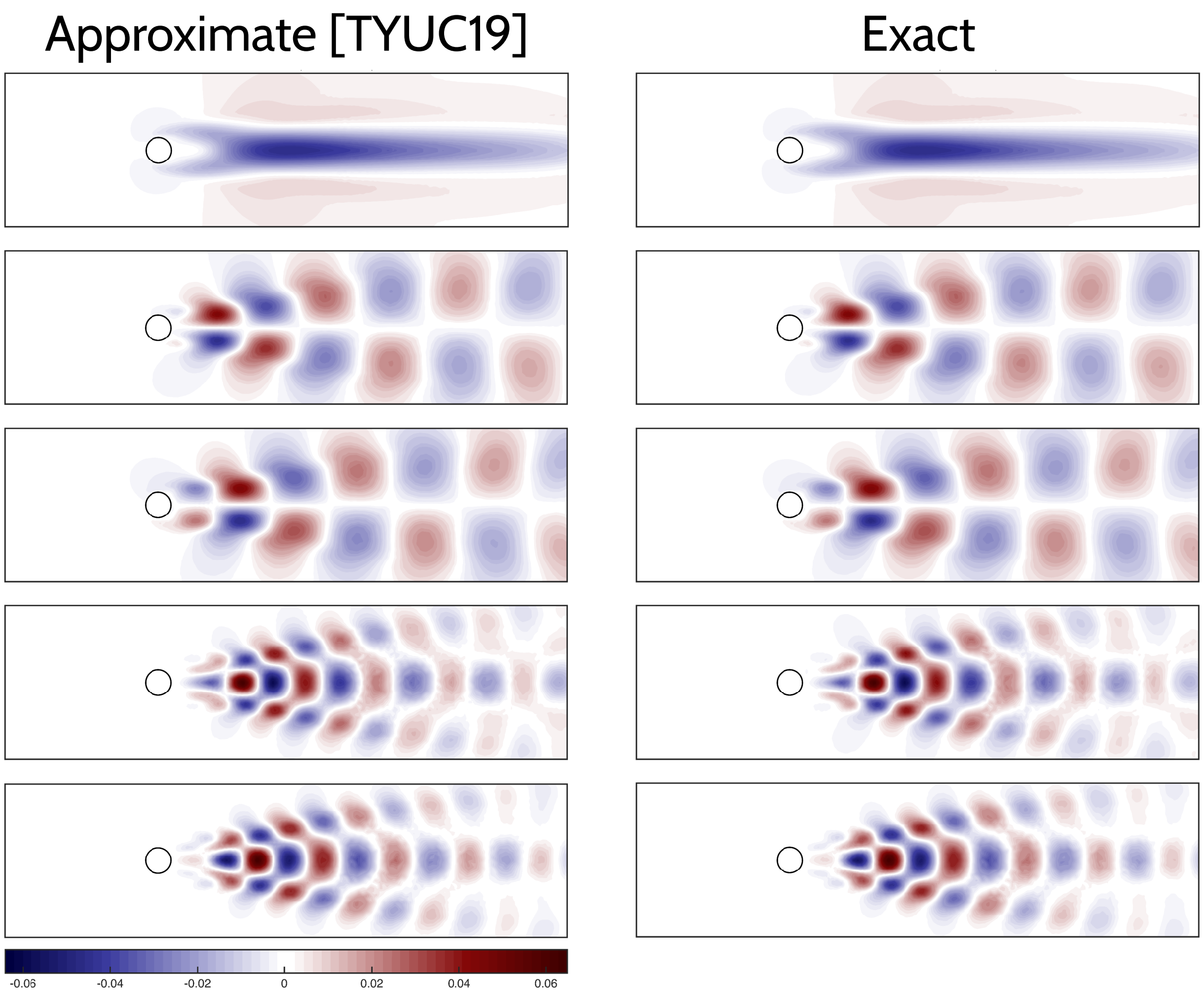
Sources: Woolfe et al. 2008; Halko et al. 2008–2011; Clarkson & Woodruff 2009; Woodruff 2014; Tropp et al. 2017–2019; Nakatsukasa 2020.

Reconstruction of von Kármán Street



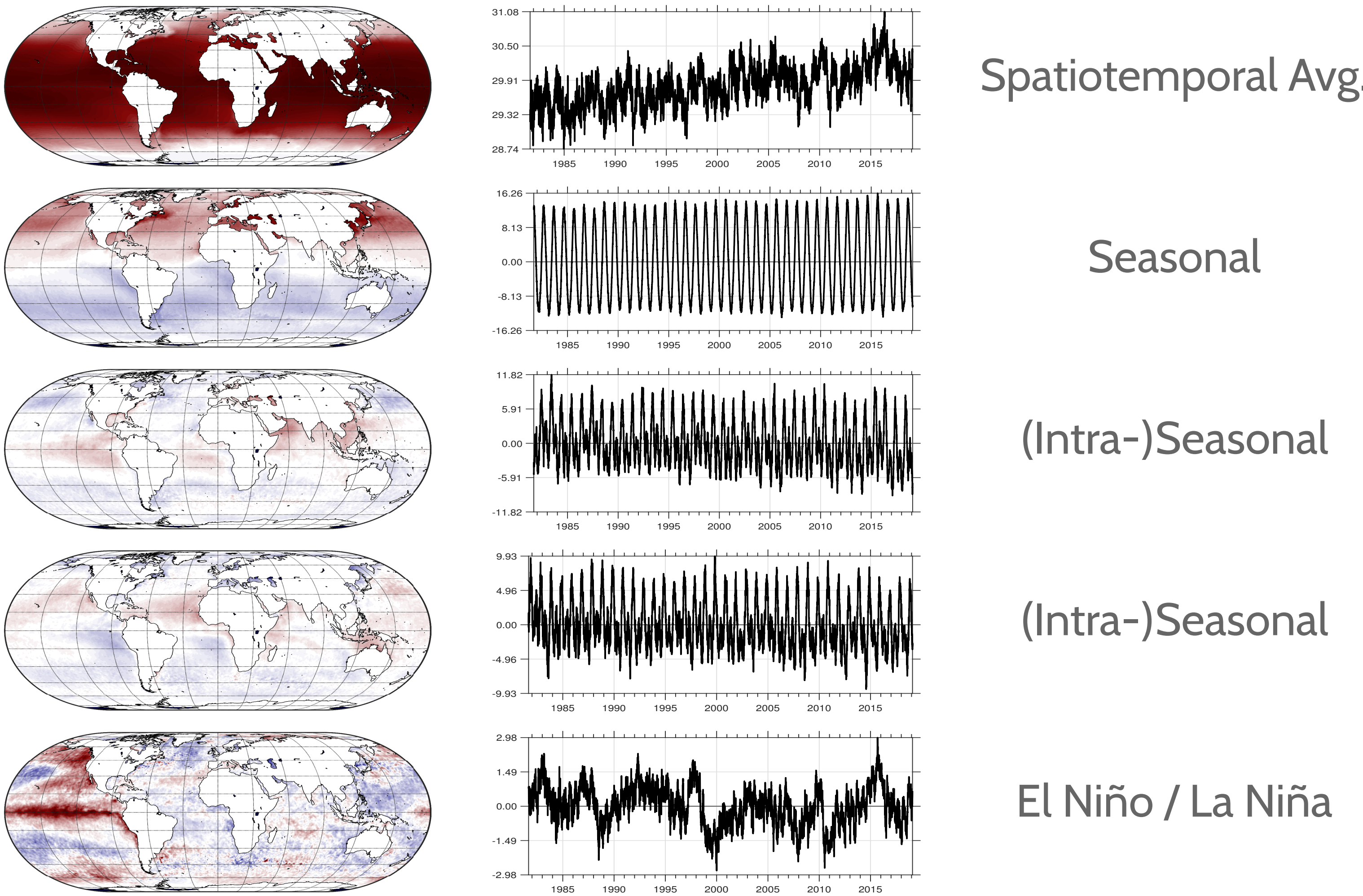
Comments: Data: $m = 10,738$; $n = 5,001$; 430 MB. Algorithm: [TYUC19], sparse maps; rank $r = 5$; storage $T = 48(m+n)$. Compression: $71\times$.

Left Singular Vectors of von Kármán Street



Comments: Data: $m = 10,738$; $n = 5,001$; 430 MB. Algorithm: [TYUC19], sparse maps; rank $r = 5$; storage $T = 48(m + n)$. Compression: $71\times$.

Singular Vectors of Sea Surface Temperature Data



Comments: Data: $m = 691,150$; $n = 13,670$; 75 GB. Algorithm: [TYUC19], sparse maps; $k = 48$; $s = 839$. Compression ratio: $222\times$.

To learn more...

E-mail: jtropp@cms.caltech.edu

Web: http://users.cms.caltech.edu/~jtropp

Some papers:

- Halko, Martinsson, & Tropp, "Finding structure with randomness: Probabilistic algorithms for computing approximate matrix decompositions," *SIREV*, 2011. arXiv 0909.4061.
- Halko, Martinsson, Shkolnisky, & Tygert, "An algorithm for the principal component analysis of large data sets," *SISC*, 2011.
- Tropp, Yurtsever, Udell, & Cevher, "Practical sketching algorithms for low-rank matrix approximation," *SIMAX*, 2017. arXiv 1609.00048.
- Tropp, Yurtsever, Udell, & Cevher, "Streaming low-rank matrix approximation with an application to scientific simulation," *SISC*, 2019. arXiv 1902.08651.
- Martinsson & Tropp, "Randomized numerical linear algebra: Foundations and algorithms," *Acta Numerica*, 2020. arXiv 2002.01387.
- Bouman, Gammie, Lee, Levis, & Tropp, "Inference of black hole fluid-dynamics from sparse interferometric measurements." ICCV 2021.
- Yang, Graff, Kumar, & Herrmann, "Low-rank representation of omnidirectional subsurface extend image volumes," *Geophysics*, 2021.
- Nakatsukasa & Tropp, "Fast & accurate randomized algorithms for linear systems and eigenvalue problems," arXiv 2111.00113.
- Tropp & Webber, "Randomized algorithms for low-rank matrix approximation: Design, analysis, and applications," in preparation.