

# Householder Symposium XXII

## Cornell University, Ithaca

### 8 - 13 June 2025



# Leveraging Numerical Linear Algebra for Robust Learning of Optimal $\mathcal{H}_2$ models from time-domain data

*Michael S. Ackermann, Serkan Gugercin*

## Abstract

We investigate the optimal  $\mathcal{H}_2$  approximation of a discrete-time, single-input single-output system

$$\begin{aligned} \mathbf{x}[k+1] &= \mathbf{A}\mathbf{x}[k] + \mathbf{b}u[k] && \text{with transfer function} && H(z) = \mathbf{c}^\top(z\mathbf{I} - \mathbf{A})^{-1}\mathbf{b}, \\ y[k] &= \mathbf{c}^\top \mathbf{x}[k] \end{aligned} \quad (1)$$

where  $\mathbf{x}[k] \in \mathbb{R}^n$ ,  $u[k] \in \mathbb{R}$ , and  $y[k] \in \mathbb{R}$  are, respectively, the states, input, and output at time  $k$ ;  $\mathbf{A} \in \mathbb{R}^{n \times n}$ ,  $\mathbf{b} \in \mathbb{R}^n$ , and  $\mathbf{c} \in \mathbb{R}^n$ . Even though we explicitly write the state-space matrices in (1), in this work, we will *never* assume access to the system matrices, system state, or evaluations of the transfer function, but only to *time-domain input-output data*

$$\mathbb{U} = [u[0] \dots u[T]]^\top \in \mathbb{R}^{T+1} \quad \text{and} \quad \mathbb{Y} = [y[0] \dots y[T]]^\top \in \mathbb{R}^{T+1}. \quad (2)$$

Given the input/output data (2), we seek to construct a data-driven reduced-order model (DDROM)

$$\begin{aligned} \mathbf{x}_r[k+1] &= \mathbf{A}_r \mathbf{x}_r[k] + \mathbf{b}_r u[k] && \text{with transfer function} && H_r(z) = \mathbf{c}_r^\top(z\mathbf{I}_r - \mathbf{A}_r)^{-1}\mathbf{b}_r, \\ y_r[k] &= \mathbf{c}_r^\top \mathbf{x}_r[k] \end{aligned} \quad (3)$$

where  $\mathbf{x}_r[k] \in \mathbb{R}^r$  is the reduced state,  $y_r[k]$  is the reduced output, and  $\mathbf{A}_r \in \mathbb{R}^{r \times r}$ ,  $\mathbf{b}_r \in \mathbb{R}^r$ , and  $\mathbf{c}_r \in \mathbb{R}^r$  with  $r \ll n$ . Specifically, we would like the DDROM (3) to minimize the  $\mathcal{H}_2$  distance

$$\|H - H_r\|_{\mathcal{H}_2}^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} |H(e^{i\omega}) - H_r(e^{i\omega})|^2 d\omega. \quad (4)$$

The optimal  $\mathcal{H}_2$  reduced order modeling problem is of interest because the  $\mathcal{H}_2$  error (4) provides a bound on the output error for finite energy inputs [4], more specifically,

$$\|y - y_r\|_{\mathcal{L}_\infty} \leq \|H - H_r\|_{\mathcal{H}_2} \|u\|_{\mathcal{L}_2}. \quad (5)$$

The Realization independent Iterative Rational Krylov Algorithm (TF-IRKA) [5] constructs  $\mathcal{H}_2$  optimal DDROMs using only samples of the transfer function  $H(\sigma)$  and  $H'(\sigma)$  without explicit access to the underlying dynamics. However, TF-IRKA requires repeated evaluations of  $H(z)$  and  $H'(z)$  at a priori unknown points outside the unit disc, i.e.,  $|\sigma| > 1$ . In some settings, one cannot actively re-sample  $H(z)$ , but is only provided input-output time-domain data as in (2).

In a recent work by Burohman et al. [8], a new method to calculate transfer function evaluations from time-domain data was presented. This method takes the form of a linear system relating the transfer function value  $H(\sigma)$  to the time domain data  $(\mathbb{U}, \mathbb{Y})$ :

$$\begin{bmatrix} \mathbb{H}_n(\mathbb{U}) & \mathbf{0} \\ \mathbb{H}_n(\mathbb{Y}) & -\gamma_n(\sigma) \end{bmatrix} \begin{bmatrix} \xi \\ H(\sigma) \end{bmatrix} = \begin{bmatrix} \gamma_n(\sigma) \\ \mathbf{0} \end{bmatrix}, \quad (6)$$

where

$$\mathbb{H}_n(\mathbb{U}) = \begin{bmatrix} u[0] & \dots & u[T-n] \\ \vdots & \ddots & \vdots \\ u[n] & \dots & u[T] \end{bmatrix} \in \mathbb{R}^{(n+1) \times (T-n+1)} \quad \text{and} \quad \gamma_n(\sigma) = \begin{bmatrix} 1 \\ \sigma \\ \vdots \\ \sigma^n \end{bmatrix} \in \mathbb{C}^{n+1}.$$

A similar linear system also relates  $H'(\sigma)$  to the time-domain data  $(\mathbb{U}, \mathbb{Y})$ . While in exact arithmetic (6) enables recovery of  $H(\sigma)$  from time domain data (2), the numerics of the problem are much more subtle. In particular, the stacked Hankel matrices are expected to be extremely ill-conditioned [3, 6, 7], and the presence of  $\sigma^n$  in  $\gamma_n(\sigma)$  could lead to overflow for large  $n$  and  $|\sigma| > 1$ . It is these numerical linear algebra considerations that we cover in this talk.

Consider the classical method to solve (6) via the singular value decomposition of the coefficient matrix in (6)

$$\widehat{\mathbf{U}}\widehat{\Sigma}\widehat{\mathbf{V}}^\top = \begin{bmatrix} \mathbb{H}_n(\mathbb{U}) & \mathbf{0} \\ \mathbb{H}_n(\mathbb{Y}) & -\gamma_n(\sigma) \end{bmatrix}.$$

We may then solve (6) by computing

$$\begin{bmatrix} \xi \\ H(\sigma) \end{bmatrix} = \widehat{\mathbf{V}}\widehat{\Sigma}^{-1}\widehat{\mathbf{U}}^\top \begin{bmatrix} \gamma_n(\sigma) \\ \mathbf{0} \end{bmatrix}. \quad (7)$$

If the solution is computed as in (7), we expect the ill-conditioning present in the coefficient matrix (and reflected in the singular values  $\widehat{\Sigma}$ ) to negatively affect the solution accuracy, especially if the data in  $(\mathbb{U}, \mathbb{Y})$  are noisy.

Our first contribution [1] makes use of the fact that we do not need to solve for the whole vector in the linear system (6); indeed the information in  $\xi$  is not used at all; we only require the last entry of the solution vector to recover  $H(\sigma)$ . This allows us to replace all but the last column in the coefficient matrix of (6) by an orthonormal basis for their range and still recover the same last component of the solution vector without needing to invert any singular values.

**Theorem 1.** *Assume access to the data (2) and define*

$$\mathbf{U} = \text{orth} \left( \begin{bmatrix} \mathbb{H}_n(\mathbb{U}) \\ \mathbb{H}_n(\mathbb{Y}) \end{bmatrix} \right). \quad (8)$$

*Then, the solution to the new linear system*

$$\begin{bmatrix} \mathbf{U} & \mathbf{0} \\ & -\gamma_n(\sigma) \end{bmatrix} \begin{bmatrix} \hat{\xi} \\ H(\sigma) \end{bmatrix} = \begin{bmatrix} \gamma_n(\sigma) \\ \mathbf{0} \end{bmatrix} \quad (9)$$

*has the same last component as the original linear system (6).*

Therefore, the highly ill-conditioned stacked Hankel matrices may be replaced by an orthonormal basis for their range without changing the last component of the solution vector. Note that this is different than the solution formula (7) where the whole vector is constructed. While theoretically equivalent, Theorem 1 does not require inverting (small/any) singular values as solving (6) via (7) requires. The effect of Theorem 1 is quite dramatic, in some examples reducing the condition number of the coefficient matrix from  $10^{16}$  to  $10^1$ . Another advantage of Theorem 1 is that when one must recover  $H(\sigma_i)$  for many different  $\sigma_i$  (as is required for  $\mathcal{H}_2$  optimality), the orthonormal basis  $\mathbf{U}$  may be precomputed once and recycled for many transfer function evaluations, reducing the online runtime from  $\mathcal{O}(n^3)$  to  $\mathcal{O}(n^2)$ .

While (9) offers great conditioning improvements over (6) when  $|\sigma| = 1$ , the presence of  $\sigma^n$  in  $\gamma_n(\sigma)$  causes the coefficient matrix in (9) to be badly scaled when  $|\sigma| > 1$ . As we seek to construct  $\mathcal{H}_2$  optimal reduced models, recovering  $H(\sigma)$  where  $|\sigma| > 1$  is required. Exploration of this issue leads to the problem of finding eigenpairs of a rank-one update to an orthogonal projection

$$\mathbf{Q}\mathbf{Q}^H + \mathbf{z}\mathbf{z}^H \quad (10)$$

where  $\mathbf{Q} \in \mathbb{C}^{m \times n}$  is subunitary and  $\mathbf{z} \in \mathbb{C}^m$  is arbitrary. In our work [2], we give explicit formulas for the eigenvectors and eigenvalues of (10).

**Theorem 2.** Let  $\mathbf{Q} \in \mathbb{C}^{m \times n}$  with  $m > n$  be subunitary and  $\mathbf{z} \in \mathbb{C}^m$ . Let  $\mathbf{u} = \mathbf{Q}\mathbf{Q}^H\mathbf{z}$  and  $\mathbf{v} = (\mathbf{I} - \mathbf{Q}\mathbf{Q}^H)\mathbf{z}$ . Assume  $\|\mathbf{v}\| \neq 0$  and  $\|\mathbf{u}\| \neq 0$ . Then the extreme nonzero eigenvalues of  $\mathbf{Q}\mathbf{Q}^H + \mathbf{z}\mathbf{z}^H$  are

$$\lambda = \frac{1}{2} \left( 1 + \|\mathbf{z}\|^2 \pm \sqrt{1 + \|\mathbf{z}\|^4 + 2\|\mathbf{z}\|^2 - 4\|\mathbf{v}\|^2} \right) \quad (11)$$

with associated eigenvectors

$$\frac{1}{2\|\mathbf{u}\|^2} \left( 1 - \|\mathbf{v}\|^2 + \|\mathbf{u}\|^2 \pm \sqrt{(\|\mathbf{v}\|^2 - \|\mathbf{u}\|^2 - 1)^2 + 4\|\mathbf{u}\|^2\|\mathbf{v}\|^2} \right) \mathbf{u} + \mathbf{v}. \quad (12)$$

We remark that the expression for the smallest nonzero eigenvalue of  $\mathbf{Q}\mathbf{Q}^* + \mathbf{z}\mathbf{z}^*$  appears similar to the lower bound for the smallest eigenvalue of a perturbed Hermitian matrix found in [9]. While the expressions are similar, we provide an *exact* expression for the updated extreme nonzero eigenvalues and associated eigenvectors under the additional assumption that the unperturbed matrix is an orthogonal projection.

Clearly, Theorem 2 also gives the condition number of the matrix  $[\mathbf{Q} \ \mathbf{z}]$ , which gives us a formula for the condition number of the coefficient matrix in (9). Expressing this condition number formula in terms of only  $\|\mathbf{z}\|$  allows us to prove that normalizing  $\mathbf{z}$  in (9) is the optimal scaling for (9). This optimal scaling is extremely effective at further reducing the condition number of (9) and opens the door for further analysis.

Results of this analysis include a connection between the underlying dynamical system (1) that produced the data (2) and the condition number of the coefficient matrix in (9). Specifically, we show that conditioning is worse when the relative derivative  $|H'(\sigma)/H(\sigma)|$  is large, a link that is not unexpected from the definition of relative condition number for functions. This leads to a method for preventing overflow in initial computation of  $\gamma_n(\sigma)$ .

We will expand upon these contributions in the talk, and in addition will showcase the efficacy of our final algorithm on benchmark examples. Comparisons with the TF-IRKA algorithm [5] show that obtaining the data  $H(\sigma)$  from time domain data  $(\mathbb{U}, \mathbb{Y})$  does not degrade the approximation quality. Also included in these examples will be a demonstration that we can construct  $\mathcal{H}_2$  optimal DDROMs from time-domain data obtained from a black-box PDE solver, an exciting indication that we may not require data explicitly obtained from a discrete-time LTI system as in (1).

## References

- [1] M. S. Ackermann and S. Gugercin. Frequency-based reduced models from purely time-domain data via data informativity. *arXiv:2311.05012*, Jan. 2024.
- [2] M. S. Ackermann and S. Gugercin. Time-domain iterative rational Krylov method. *arXiv:2407.12670*, July 2024.
- [3] S. Al-Homidan, M. M. Alshahrani, C. G. Petra, and F. A. Potra. Minimal condition number for positive definite Hankel matrices using semidefinite programming. *Linear Algebra and its Applications*, 433(6):1101–1109, Nov. 2010.

- [4] A. Antoulas, C. Beattie, and S. Gürgencin. *Interpolatory Methods for Model Reduction*. Computational Science and Engineering. SIAM, 2020.
- [5] C. Beattie and S. Gugercin. Realization-independent  $H_2$ -approximation. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pages 4953–4958, Maui, HI, USA, Dec. 2012. IEEE.
- [6] B. Beckermann. The condition number of real Vandermonde, Krylov and positive definite Hankel matrices. *Numerische Mathematik*, 85(4):553–577, June 2000.
- [7] B. Beckermann and A. Townsend. Bounds on the singular values of matrices with displacement structure. *SIAM Review*, 61(2):319–344, Jan. 2019.
- [8] A. M. Burohman, B. Besselink, J. M. A. Scherpen, and M. K. Camlibel. From data to reduced-order models via moment matching. *arXiv:2011.00150*, Oct. 2020.
- [9] I. C. F. Ipsen and B. Nadler. Refined Perturbation Bounds for Eigenvalues of Hermitian and Non-Hermitian Matrices. *SIAM Journal on Matrix Analysis and Applications*, 31(1):40–53, Jan. 2009.

# Rational Interpolation, the Loewner Framework and the Kolmogorov Superposition Theorem.

*Athanasios C. Antoulas, Ion Victor Gosea, Charles Poussot-Vassal*

## Abstract

Problem #119, in [1], asks the question: *Are there actually functions of three variables?*

Stated differently: is it possible to use compositions of functions of two or fewer variables to express any function of three variables? This question is related to Hilbert's 13th problem: are there any genuine continuous multivariate functions. As a matter of fact, Hilbert conjectured the existence of a three-variable continuous function which cannot be expressed in terms of composition and addition of two-variable continuous functions. For a recent overview of this problem, see [2].

For continuous function, the Kolmogorov Superposition Theorem (KST) answers this question negatively. It shows namely that continued functions of several variables can be expressed as composition and superposition of functions of one variable. Thus, there are no *true* functions of three variables.

The present contribution presents connections between the Loewner framework for rational interpolation of multivariate functions and KST restricted to rational functions. The result is the formulation of KST for the special case of rational functions. As a byproduct *taming of the curse of dimensionality*, both in computational complexity, storage, and last but not least, numerical accuracy, is achieved.

**Short summary of the Loewner framework.** The Loewner framework is an interpolatory approach designed for approximating linear and nonlinear systems. Reference [3] extends this framework to linear parametric systems with an arbitrary number of parameters, in other words to multivariate functions of  $n$  variables. One main innovation established is the construction of data-based system realizations for any number of parameters. Equally importantly, [3] shows how to alleviate the computational burden, storage and numerical accuracy, by avoiding the explicit construction of higher dimensional Loewner matrices of size  $N \times N$ . Instead, the proposed methodology achieves decoupling of variables, leading to (i) a complexity reduction from  $\mathcal{O}(N^3)$  to below than  $\mathcal{O}(N^{1.5})$  when  $N > 5$  and (ii) to memory storage bounded by the largest variable dimension rather than the product of all variable dimensions, thus taming the curse of dimensionality and making the solution scalable to large data sets.

After defining a new multivariate realization, we introduce the higher dimensional multivariate Loewner matrices and show that they can be computed by solving a coupled set of Sylvester equations. The *null space* of these Loewner matrices then allows the computation of the multivariate *barycentric weights* of the associated rational function. One of the main results of [3] is to show how the null space of  $N$ -dimensional Loewner matrices can be computed using a sequence of 1-dimensional Loewner matrices. This leads to a drastic computational burden reduction. This also leads to the formulation of KST for rational functions. Finally, two algorithms are proposed (one direct and one iterative) to construct, directly from data, multivariate (or parametric) realizations ensuring (approximate) interpolation. For details on the above material see [3].

The purpose of this contribution is to make contact of the above results with the Kolmogorov Superposition Theorem. For clarity of exposition we will illustrate the main features of our approach by means of a three-variable example.

**Example.** Consider the three-variable function  $\mathbf{H}(s, t, x) = \frac{s^2+xs+1}{t+x+st+2}$ . Since the degrees in each variable are  $(2, 1, 1)$ , we will need the integers  $\nu_1 = 3$ ,  $\nu_2 = 2$ , and  $\nu_3 = 2$ , This implies that  $N = \nu_1\nu_2\nu_3 = 12$ . The right and left interpolation points are

$$\begin{aligned} s_1 &= 1, \quad s_2 = 2, \quad s_3 = 3, & t_1 &= 4, \quad t_2 = 5, & x_1 &= 6, \quad x_2 = 7, \quad \text{and} \\ s_4 &= 3/2, \quad s_5 = 5/2, \quad s_6 = 7/2, & t_3 &= 9/5, \quad t_4 = 11/5, & x_3 &= 13/3, \quad x_4 = 5, \quad \text{respectively.} \end{aligned}$$

Following the theory in [3], the right triples of interpolation points are  $\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3] \otimes \mathbb{I}_{1,2} \otimes \mathbb{I}_{1,2}$ ,  $\mathbf{T} = \mathbb{I}_{1,3} \otimes [\mathbf{t}_1, \mathbf{t}_2] \otimes \mathbb{I}_{1,2}$ ,  $\mathbf{X} = \mathbb{I}_{1,3} \otimes \mathbb{I}_{1,2} \otimes [\mathbf{x}_1, \mathbf{x}_2] \in \mathbf{C}^{1 \times N}$ . Thus the resulting 3D-Loewner matrix has dimension  $N \times N$  and the barycentric weights are

$$\mathbf{Bary} = \left[ \frac{16}{29} \quad -\frac{17}{29} \quad -\frac{18}{29} \quad \frac{19}{29} \quad -\frac{40}{29} \quad \frac{42}{29} \quad \frac{46}{29} \quad -\frac{48}{29} \quad \frac{24}{29} \quad -\frac{25}{29} \quad -\frac{28}{29} \quad 1 \right]^T.$$

Again, the theory in [3], allows the demposition of this vector in a (pointwise) product of barycentric weights with respect to each variable, separately. Thus *decoupling* the problem is achieved, one of the important aspects of KST; in [3] we obtain:

$$\mathbf{Bary} = \mathbf{Bary}_x \odot \mathbf{Bary}_t \odot \mathbf{Bary}_s,$$

where  $\odot$  denotes the pointwise product. This is a special case of formula (5.5) in [3].

This is the *key result* which allows the connection with KST and taming the curse of dimensionality. We have thus shown that the 3D multivariate function can be computed in terms of three 1D functions (one in each variable). These functions are denoted below by  $\Phi(x)$ ,  $\Psi(t)$  and  $\Omega(s)$ . Furthermore  $\mathbf{Lag}_x$ ,  $\mathbf{Lag}_t$  and  $\mathbf{Lag}_s$  are the *Lagrange bases components* in each variable. Finally  $\mathbf{W}$  are the right interpolation values for the triples in  $\mathbf{S} \times \mathbf{T} \times \mathbf{X}$ . The ensuing numerical values are as follows:

$$\underbrace{\begin{bmatrix} -\frac{16}{17} \\ 1 \\ -\frac{18}{19} \\ 1 \\ -\frac{20}{21} \\ 1 \\ -\frac{23}{24} \\ 1 \\ -\frac{24}{25} \\ 1 \\ -\frac{28}{29} \\ 1 \end{bmatrix}}_{\mathbf{Bary}_x}, \underbrace{\begin{bmatrix} -\frac{17}{19} \\ -\frac{17}{19} \\ 1 \\ -\frac{7}{8} \\ -\frac{7}{8} \\ 1 \\ -\frac{48}{29} \\ -\frac{48}{29} \\ -\frac{25}{29} \\ -\frac{25}{29} \\ 1 \\ 1 \end{bmatrix}}_{\mathbf{Bary}_t}, \underbrace{\begin{bmatrix} \frac{19}{29} \\ \frac{19}{29} \\ \frac{19}{29} \\ \frac{19}{29} \\ \frac{1}{29} \\ \frac{1}{29} \\ -\frac{48}{29} \\ -\frac{48}{29} \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}}_{\mathbf{Bary}_s}, \underbrace{\begin{bmatrix} \frac{1}{x-6} \\ \frac{1}{x-7} \\ \frac{1}{x-6} \\ \frac{1}{x-7} \end{bmatrix}}_{\mathbf{Lag}_x}, \underbrace{\begin{bmatrix} \frac{1}{t-4} \\ \frac{1}{t-4} \\ \frac{1}{t-5} \\ \frac{1}{t-5} \\ \frac{1}{t-4} \\ \frac{1}{t-4} \\ \frac{1}{t-5} \\ \frac{1}{t-5} \\ \frac{1}{t-4} \\ \frac{1}{t-4} \\ \frac{1}{t-5} \\ \frac{1}{t-5} \end{bmatrix}}_{\mathbf{Lag}_t}, \underbrace{\begin{bmatrix} \frac{1}{s-1} \\ \frac{1}{s-1} \\ \frac{1}{s-1} \\ \frac{1}{s-1} \\ \frac{1}{s-2} \\ \frac{1}{s-2} \\ \frac{1}{s-2} \\ \frac{1}{s-2} \\ \frac{1}{s-3} \\ \frac{1}{s-3} \\ \frac{1}{s-3} \\ \frac{1}{s-3} \end{bmatrix}}_{\mathbf{Lag}_s}, \underbrace{\begin{bmatrix} \frac{1}{2} \\ \frac{9}{17} \\ \frac{4}{9} \\ \frac{9}{19} \\ \frac{17}{20} \\ \frac{19}{21} \\ \frac{17}{23} \\ \frac{19}{24} \\ \frac{7}{6} \\ \frac{31}{25} \\ 1 \\ \frac{31}{29} \end{bmatrix}}_{\mathbf{W}}$$

def  $\Rightarrow \begin{cases} \Phi(x) = \mathbf{Bary}_x \odot \mathbf{Lag}_x, \\ \Psi(t) = \mathbf{Bary}_t \odot \mathbf{Lag}_t, \\ \Omega(s) = \mathbf{Bary}_s \odot \mathbf{Lag}_s. \end{cases}$

With the above notation we can express  $\mathbf{H}$  as the quotient of two rational functions:

$$\begin{aligned} \hat{\mathbf{n}}(s, t, x) &= \sum_{\text{rows}} [\mathbf{W} \odot \Phi(\mathbf{x}) \odot \Psi(\mathbf{t}) \odot \Omega(\mathbf{s})] \\ \hat{\mathbf{d}}(s, t, x) &= \sum_{\text{rows}} [\Phi(\mathbf{x}) \odot \Psi(\mathbf{t}) \odot \Omega(\mathbf{s})] \end{aligned} \quad \left\{ \Rightarrow \frac{\hat{\mathbf{n}}(s, t, x)}{\hat{\mathbf{d}}(s, t, x)} = \mathbf{H}(s, t, x). \right.$$

Consequently, KST for rational functions, as *composition and superposition* of one-variable functions, takes the form:

$$\begin{aligned} \hat{\mathbf{n}}(s, t, x) &= \sum_{\text{rows}} \exp [\log \mathbf{W} + \log \Phi(\mathbf{x}) + \log \Psi(\mathbf{t}) + \log \Omega(\mathbf{s})] \\ \hat{\mathbf{d}}(s, t, x) &= \sum_{\text{rows}} \exp [\log \Phi(\mathbf{x}) + \log \Psi(\mathbf{t}) + \log \Omega(\mathbf{s})]. \end{aligned} \quad \left\{ \right. \quad (*)$$

**Some details of the above computation.** To compute (i)  $\text{Bary}_x$ , computation of the nullspace of six 1D Loewner matrices of size  $2 \times 2$  is needed, (ii)  $\text{Bary}_t$ , computation of three 1D Loewner matrices of size  $2 \times 2$  is needed, and (iii)  $\text{Bary}_s$ , computation of one 1D Loewner matrix of size  $3 \times 3$  is needed. The resulting total computation using 1D Loewner matrices is  $\nu_1^3\nu_2\nu_3 + \nu_2^3\nu_3 + \nu_3^3 = 99$  flops as opposed to  $(\nu_1\nu_2\nu_3)^3 = 1728$  flops, when working with 3D Loewner matrices. For details on the computational complexity, storage and numerical accuracy, we refer to [3]. Note also that the  $nD$  Loewner matrix is of dimension  $12 \times 12$  while in the 1D case, a maximum of  $3 \times 3$  matrix is needed.

**Comparison of KST and (\*).** A number of researchers have contributed in sharpening Kolmogorov's original result, so currently it is often referred to as the *Kolmogorov, Arnol'd, Kahane, Lorenz and Sprecher Theorem* (see [2], theorem 2.1). For simplicity we will follow [2] and state this result for  $n = 3$ , so that we can compare it with (\*).

**Theorem.** Given a continuous function  $f : [0, 1]^3 \rightarrow \mathbb{R}$  of three variables, there exist real numbers  $\lambda_i$ ,  $i = 1, 2$ , and single-variable continuous functions  $\phi_k : [0, 1] \rightarrow \mathbb{R}$ ,  $k = 1, \dots, 7$ , and a single-variable function  $g : \mathbb{R} \rightarrow \mathbb{R}$ , such that

$$f(x_1, x_2) = \sum_{k=1}^7 g(\phi_k(x_1) + \lambda_1\phi_k(x_2) + \lambda_2\phi_k(x_3)), \quad \forall (x_1, x_2, x_3) \in [0, 1]^3.$$

In the above result,  $\lambda_i$  and  $\phi_k$  do not depend on  $f$ . Thus for  $n = 3$ , eight functions are needed together with two real scalars  $\lambda_i$ .

#### Similarities and differences between KST and (\*).

1. While KST refers to continuous functions defined on  $[0, 1]^n$ , (\*) is concerned with rational functions defined on  $\mathbb{C}^n$ .
2. In its present form (\*) is valid in a particular basis, namely the *Lagrange* basis. Multiplication of functions in (\*), is defined with respect to this basis.
3. The composition and superposition property holds for the numerator and denominator. Notice that in KST no explicit denominators are considered. This is important in our case because (\*) preserves interpolation conditions.
4. The parameters needed are  $n = 3$  Lagrange bases (one in each variable) and the barycentric coefficients of numerator and denominator.
5. Both KST and (\*) accomplish the goal of replacing the computation of multivariate functions, by means of a series of computations involving single-variable functions, KST for general continuous functions and (\*) for rational functions. Notice also that (\*) provides a different formulation of the problem than KST.

## References

- [1] G. Pólya and G. Szegö, *Problems and Theorems in Analysis*, Volume 1, Springer Verlag, 1972.
- [2] Sidney A. Morris, *Hilbert 13: Are there any genuine continuous multivariate functions?*, Bulletin (New Series) of the AMS, **58**: 107-118 (2021).
- [3] Athanasios C. Antoulas, Ion Victor Gosea, Charles Poussot-Vassal,  
*The Loewner framework for parametric systems: Taming the curse of dimensionality*,  
<https://arxiv.org/abs/2405.00495>. Submitted to SIREV, May 2024.

# Collect, Commit, Expand: an Efficient Strategy for Column Subset Selection on Extremely Wide Matrices

*Robin Armstrong, Anil Damle*

## Abstract

The column subset selection problem (CSSP) appears in a remarkably wide range of applications. For example, point selection problems that arise in model order reduction [5], computational chemistry [7], spectral clustering [8], low-rank approximation [6, 13], and Gaussian process regression [15] can all be treated as instances of CSSP. Given a matrix  $A \in \mathbb{R}^{m \times n}$  and a target rank  $k \leq \min\{m, n\}$ , CSSP seeks to find a set of  $k$  columns from  $A$  that are “highly linearly independent.” A more formal statement, using the framework of rank-revealing QR factorizations [4, 11, 12], is that algorithms for CSSP produce an index set  $S \in [n]^k$  satisfying

$$\sigma_{\min}(A(:, S)) \geq \frac{\max_{J \in [n]^k} \sigma_{\min}(A(:, J))}{q(n, k)} \quad (1)$$

for some low-degree bivariate polynomial  $q$ . The Golub-Businger algorithm [3], which uses alternating column pivots and Householder reflections to compute a column-pivoted QR (CPQR) factorization  $A\Pi = QR$ , is widely used for this problem. After running this algorithm, choosing  $A(:, S) = A\Pi(:, 1 : k)$  results in an  $S$  which does not provably satisfy (1), but which nearly always brings  $\sigma_{\min}(A(:, S))$  reasonably close to its maximum.

We seek to address a computational bottleneck in the Golub-Businger algorithm that results from sequential application of Householder reflections with level-2 BLAS. Most existing solutions to this problem involve reducing the number of rows manipulated with BLAS-2. For example, the CPQR factorization routine in LAPACK reflects only as many rows as are needed to determine a small block of pivot columns, deferring the full Householder reflection to a later application with BLAS-3 [16]. There also exists a large class of randomized algorithms that apply standard CPQR routines to sketched matrices with far fewer rows [6, 10, 14, 17]. We, however, are interested in problems where the difficulty arises not from the number of rows, but from the number of columns. For example, spectral clustering generates instances of CSSP where each row represents a cluster and each column represents a data point [8], meaning  $m$  may be several orders of magnitude smaller than  $n$ . In these applications, reducing the number of rows being manipulated with BLAS-2 does not address the main bottleneck.

We will demonstrate a new CPQR-based column selection algorithm that effectively mitigates the BLAS-2 bottleneck for matrices with far more columns than rows. Our algorithm divides columns into a “tracked” set, where residual column norms are recorded, and an “untracked” set, where only overall norms are recorded. Pivot columns are selected in blocks, and each block is selected using a three-step strategy:

1. A “collect” step assembles a small number of candidate columns from the tracked set, and forms a conventional CPQR factorization of the candidates.
2. A “commit” step uses the CPQR factors to identify a set of provably “safe” pivots from among the candidates, and updates *only* the tracked columns according to the new pivots.
3. An “expand” step moves a small number of columns from “untracked” to “tracked,” setting up for a new round of candidates to be chosen in the next block.

We call this algorithm CCEQR (“Collect-Commit-Expand QR”).

$n$	GEQP3 Runtime (s)	CCEQR Runtime (s)
$10^2$	$1.9 \times 10^{-5}$	$8.1 \times 10^{-5}$
$10^3$	$2.7 \times 10^{-4}$	$3.7 \times 10^{-4}$
$10^4$	$1.8 \times 10^{-3}$	$7.5 \times 10^{-4}$
$10^5$	$1.8 \times 10^{-2}$	$3.7 \times 10^{-3}$
$10^6$	$4.0 \times 10^{-1}$	$4.5 \times 10^{-2}$

Figure 1: Average runtimes of GEQP3 and CCEQR (over 20 trials) on matrices of size  $20 \times n$ , for increasing  $n$ . Test matrices were generated from a spectral clustering problem, and correspond to Laplacian embeddings of  $n$  data points drawn from a 20-component Gaussian mixture model.

CCEQR is fully deterministic, and unlike CPQR-based column selection algorithms which distribute the column load across several parallel processors [1, 2, 9], it provably selects the same basis columns as the Golub-Businger algorithm (assuming no ties between residual column norms). Using test problems from domains such as computational chemistry, model order reduction, and spectral clustering, we will demonstrate that CCEQR can run several times faster than the standard LAPACK routine (GEQP3) for matrices with an unbalanced column norm distribution. For example, Figure 1 shows that CCEQR can run as much as 10 times faster than GEQP3 for certain spectral clustering problems. We will also show that CCEQR and GEQP3 have essentially the same runtime for large unstructured problems, such as Gaussian random test matrices.

## References

- [1] Christian H. Bischof. A parallel QR factorization algorithm with controlled local pivoting. *SIAM Journal on Scientific and Statistical Computing*, 12(1):36–57, 1991.
- [2] Christian H. Bischof and Per Christian Hansen. Structure-preserving and rank-revealing QR-factorizations. *SIAM Journal on Scientific and Statistical Computing*, 12(6):1332–1350, 1991.
- [3] Peter Businger and Gene H. Golub. Linear least squares solutions by Householder transformations. *Numerische Mathematik*, 7:269 – 276, 1965.
- [4] Shivkumar Chandrasekaran and Ilse C. F. Ipsen. On rank-revealing factorisations. *SIAM Journal on Matrix Analysis and Applications*, 15(2):592–622, 1994.
- [5] Saifon Chaturantabut and Danny C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, 2010.
- [6] H. Cheng, Z. Gimbutas, P. G. Martinsson, and V. Rokhlin. On the compression of low rank matrices. *SIAM Journal on Scientific Computing*, 26(4):1389–1404, 2005.
- [7] Anil Damle, Lin Lin, and Lexing Ying. Compressed representation of Kohn-Sham orbitals via selected columns of the density matrix. *J Chem Theory Comput*, 14:1463–1469, 2015.
- [8] Anil Damle, Victor Minden, and Lexing Ying. Simple, direct and efficient multi-way spectral clustering. *Information and Inference: A Journal of the IMA*, 8(1):181–203, 06 2018.
- [9] James W. Demmel, Laura Grigori, Ming Gu, and Hua Xiang. Communication avoiding rank revealing QR factorization with column pivoting. *SIAM Journal on Matrix Analysis and Applications*, 36(1):55–89, 2015.

- [10] Jed A. Duersch and Ming Gu. Randomized QR with column pivoting. *SIAM Journal on Scientific Computing*, 39(4):C263–C291, January 2017.
- [11] Ming Gu and Stanley C. Eisenstat. Efficient algorithms for computing a strong rank-revealing QR factorization. *SIAM Journal on Scientific Computing*, 17(4):848–869, 1996.
- [12] Y.P. Hong and C.-T. Pan. Rank-revealing QR factorizations and the singular value decomposition. *Mathematics of Computation*, 58(197):213 – 232, 1992.
- [13] Michael W. Mahoney and Petros Drineas. CUR matrix decompositions for improved data analysis. *Proceedings of the National Academy of Sciences*, 106(3):697–702, 2009.
- [14] Per-Gunnar Martinsson, Gregorio Quintana Ortí, Nathan Heavner, and Robert van de Geijn. Householder QR factorization with randomization for column pivoting (HQRRP). *SIAM Journal on Scientific Computing*, 39(2):C96–C115, 2017.
- [15] Victor Minden, Anil Damle, Kenneth L. Ho, and Lexing Ying. Fast spatial Gaussian process maximum likelihood estimation via skeletonization factorizations. *Multiscale Modeling & Simulation*, 15(4):1584–1611, 2017.
- [16] Gregorio Quintana-Ortí, Xiaobai Sun, and Christian H. Bischof. A BLAS-3 version of the QR factorization with column pivoting. *SIAM Journal on Scientific Computing*, 19(5):1486–1494, 1998.
- [17] Franco Woolfe, Edo Liberty, Vladimir Rokhlin, and Mark Tygert. A fast randomized algorithm for the approximation of matrices. *Applied and Computational Harmonic Analysis*, 25(3):335–366, 2008.

# Numerical Linear Algebra on Quantum Computers Made Simple

*Haim Avron, Lior Horesh, Liron Mor-Yosef, Shashanka Ubaru*

## Abstract

The field of quantum computing offers a unique opportunity to revolutionize numerical linear algebra and scientific computing. This is due to the capability of quantum computers to efficiently model complex structures, and their ability to represent and act on high dimensional vectors and matrices using exponentially fewer qubits. These advantages arise due to the principles of superposition and entanglement inherent to qubits.

However, the current landscape of quantum computing research emphasizes intricate, tailor-made circuit designs, created in an adhoc manner for specific mathematical challenges. While such state-of-the-art quantum algorithms offer a powerful approach by translating various computations into circuits, their development not straightforward. Development often requires significant “circuit engineering” to achieve the desired mathematical outcomes. In this talk, I will discuss our recent progress on developing a unified and systemic approach to utilizing quantum computing for numerical linear algebra. Our research centers around a novel Quantum Linear Algebra (qLA) framework offering fundamental matrix algebra building blocks, akin to BLAS – but for Quantum Computers.

The motivation is to make progress toward harnessing the power of quantum computing to perform linear algebra operations efficiently, while enabling seamless development of numerical linear algebra algorithms utilizing quantum computers. Even though quantum computing offers, in principle, the potential for exponential improvements in runtime and storage complexity, for linear algebra operations even modest gains are valuable and seemingly much more realistic. Indeed, even seemingly minor efficiency improvements, such as reducing from  $O(n^3)$  to  $O(n^2)$  the complexity of core operations like matrix-matrix product, can have significant real-world impact. This makes quantum numerical linear algebra a highly promising area for exploration. Developing a high-level framework for quantum linear algebra algorithms is key to unlocking qNLA’s potential in scientific computing and machine learning. Such a framework would empower developers by hiding low-level circuit complexities, ultimately accelerating progress in this exciting field.

The qLA framework is designed to provide a high-level interface for writing and executing linear algebra subroutines by translating mathematical formulas directly into quantum circuits. While qLA-based algorithms are classical to classical, they will produce circuits specifically intended for efficient execution on quantum computers. The framework enables a wide range input models, a broad range of matrix algebra operations, and facilitate seamless circuit design. Beyond its core functionalities, the framework is developed so that it can be leveraged to design quantum algorithms for scientific computing and machine learning problems, particularly those problems involving computationally intensive linear algebra operations. The talk will discuss both the algorithms and functionalities within the qLA framework, and how they can be leveraged to design novel quantum algorithms.

An initial version of the qLA framework, called quantum Matrix States Linear Algebra (qMSLA), is described in our accepted manuscript [1]. qMSLA focuses on a single input model: *state preparation circuits* (the term will be explained in the talk), and provides a minimal set of linear algebra operations that can be used to design quantum algorithm for estimating *multivariate traces*, i.e. the trace of products of matrices. In the talk, I will discuss our recent work on expanding this framework with additional input models (block encoding and density preparation circuits), a comprehensive list of foundational matrix algebra operations, and the relation between input models.

## References

- [1] Liron Mor Yosef, Shashanka Ubaru, Lior Horesh, Haim Avron Multivariate Trace Estimation Using Quantum State Space Linear Algebra *SIAM Journal on Matrix Analysis and Applications*, to appear.

# Some Modified Matrix Eigenvalue Problems

Zhaojun Bai

## Abstract

This is the title of well-known numerical linear algebra survey article by Gene Golub published in 1973 [1]. The article covers a range of matrix eigenvalue problems which require some manipulations before the standard algorithms may be used. I am using the same title to consider a new set of modified matrix eigenvalue problems. This includes constrained and bi-level optimizations arising from algorithms for fairness in machine learning, such as spectral clustering with group fairness [2] and fair principal component analysis [3]. We also consider eigenvalue optimization via 2D eigenvalue problem with applications to the calculation of the distance to instability among others [4], and stationary values of a quadratic form subject to non-homogeneous linear constraints for applications such as image segmentation with constraints [5]. I will discuss how to explore the underlying structures of these problems to turn them into our familiar eigenvalue problems and algorithms. This talk is based on joint work with Ian Davidson, Aaron Davis, Ren-Cang Li, Ding Lu, Tianyi Lu, Junhui Shen, Yangfeng Su, Ji Wang, and Yunshen Zhou.

## References

- [1] G. H. Golub, Some modified matrix eigenvalue problems, SIAM Review, 15(2), pp.318–334, 1973.
- [2] J. Wang, D. Lu, I. Davidson and Z. Bai, *Scalable spectral clustering with group fairness constraints*, Proceedings of The 26th International Conference on Artificial Intelligence and Statistics (AISTAT), PMLR 206:6613-6629, 2023.
- [3] J. Shen, A. Davis, D. Lu and Z. Bai *Fair and efficient: hidden convexity of fair PCA and fast solution via eigenvalue optimization*, submitted, 2024.
- [4] T. Lu, Y. Su and Z. Bai, *Variational characterization and Rayleigh quotient iteration of 2D eigenvalue problem with Applications*, SIAM J. Matrix. Anal. Appl. Vol.45, No.3, pp.1455-1468, 2024
- [5] Y. Zhou, Z. Bai and R.-C. Li, *Linear constrained Rayleigh quotient optimization: theory and algorithms*, CSIAM Trans. Appl. Math., 2(2), pp.195-262, 2021

# Accelerating Randomized Tensor Decompositions using Structured Random Matrices

*Grey Ballard*

## Abstract

Tensor decompositions are generalizations of low-rank matrix approximations to higher dimensional data. They have become popular for their utility across applications—including blind source separation, dimensionality reduction, compression, anomaly detection—where the original data is represented as a multidimensional array. In this talk, I will focus on randomized methods for computing Tucker and Tensor Train (TT) decompositions. The kernel computation is computing a sketch of an unfolding/matricization of the tensor, and we impose structure on the random matrix in order to exploit structure in the tensor unfolding and reduce computational cost. I will discuss theoretical results on the accuracy of these approaches, their accuracy in practice, and the performance improvement they achieve over deterministic methods.

The TT format is useful in particular for tensors of many modes, including representing approximations to the solution of certain types of parametrized partial differential equations. The fundamental operation used to maintain feasible memory and computational time is called rounding, which truncates the internal ranks of a tensor already in TT format. I will present multiple randomized algorithms for this task that are generalizations of randomized low-rank matrix approximation algorithms and provide significant reduction in computation compared to deterministic TT rounding algorithms. We achieve computational efficiency by using random matrix sketches that mirror the TT format of the input tensor. Randomization is particularly effective in the case of rounding a sum of TT tensors, which is the bottleneck computation in the adaptation of GMRES to vectors in TT format. In this talk, I will present the randomized algorithms, compare their empirical accuracy and computational time with deterministic alternatives (including results from [1]), and discuss recent progress on probabilistic error analysis of the algorithms.

I will present two Tucker decomposition algorithms that scale to large data (and many processors), significantly reduce both computation and communication cost compared to previous deterministic and randomized approaches, and obtain nearly the same approximation errors. The key idea in our algorithms is to perform randomized sketches with Kronecker-structured random matrices, which reduces computation compared to unstructured random matrices and can be implemented using a fundamental tensor computational kernel. I will state probabilistic error analysis of our algorithms and present a new parallel algorithm for the structured randomized sketch. Our experimental results demonstrate that our combination of randomization and parallelization achieves accurate Tucker decompositions much faster than alternative approaches. We observe up to a  $16\times$  speedup over the fastest deterministic parallel implementation on 3D simulation data [2].

## References

- [1] Hussam Al Daas, Grey Ballard, Paul Cazeaux, Eric Hallman, Agnieszka Miedlar, Mirjeta Pasha, Tim W. Reid, and Arvind K. Saibaba. Randomized algorithms for rounding in the tensor-train format. *SIAM Journal on Scientific Computing*, 45(1):A74–A95, 2023.
- [2] Rachel Minster, Zitong Li, and Grey Ballard. Parallel randomized Tucker decomposition algorithms. *SIAM Journal on Scientific Computing*, 46(2):A1186–A1213, 2024.

# The Akhiezer iteration for matrix functions and Sylvester equations

*Cade Ballew, Thomas Trogdon, and Heather Wilber*

## Abstract

We consider the computation of matrix functions  $f(\mathbf{A})$  when the eigenvalues of  $\mathbf{A}$  are known to lie on or near a collection of disjoint intervals  $\Sigma \subset \mathbb{R}$ . The Akhiezer iteration is an inverse-free iterative method for this task that arises via an orthogonal polynomial expansion of  $f$  on  $\Sigma$ . When  $\Sigma$  consists of two or more intervals, extensions of the Chebyshev polynomials, often called the Akhiezer polynomials, are employed. This method is an extension of the classical Chebyshev iteration and an effective implementation of the ideas of Saad [7].

The Akhiezer iteration relies on orthogonal polynomial recurrence coefficients and Cauchy integrals. Importantly, orthonormal polynomials  $\{p_j\}_{j=0}^{\infty}$  with respect to a weight function  $w$  satisfy a symmetric three-term recurrence

$$\begin{aligned} xp_0(x) &= a_0 p_0(x) + b_0 p_1(x), \\ xp_j(x) &= b_{j-1} p_{j-1}(x) + a_j p_j(x) + b_j p_{j+1}(x), \quad j \geq 1, \end{aligned} \tag{1}$$

for some recurrence coefficients  $\{a_j\}_{j=0}^{\infty}, \{b_j\}_{j=0}^{\infty}$  where  $b_j > 0$  for all  $j$ . The Cauchy integrals of these polynomials are defined by

$$\mathcal{C}_{\Sigma}[p_j w](z) = \frac{1}{2\pi i} \int_{\Sigma} \frac{p_j(s)w(s)}{s - z} ds.$$

As a particular example, consider  $\Sigma = [a_1, b_1] \cup [a_2, b_2]$ ,  $b_1 < a_2$ . The orthonormal polynomials with respect to the weight function

$$w(x) = \frac{1}{\pi} \mathbb{1}_{\Sigma}(x) \frac{\sqrt{x - b_1}}{\sqrt{b_2 - x} \sqrt{x - a_1} \sqrt{x - a_2}},$$

were constructed by Akhiezer in [1]. The construction gives an explicit formula for these polynomials in terms of Jacobi elliptic and theta functions. From this formula and derivation, formulae for their recurrence coefficients and Cauchy integrals can be derived [2]. When explicit formulae are not known, e.g., when  $\Sigma$  consists of more than two intervals,  $N$  pairs of recurrence coefficients and Cauchy integrals can be computed in  $O(N)$  operations via the numerical method of [3].

Given a function  $f$  that is analytic in a region containing  $\Sigma$ , let  $p_0, p_1, \dots$  denote the orthonormal polynomials with respect to  $w$ . Then, for  $x \in \Sigma$ , a  $p_j$ -series expansion for  $f$  is given by

$$f(x) = \sum_{j=0}^{\infty} \alpha_j p_j(x), \quad \alpha_j = \int_{\Sigma} f(x)p_j(x)w(x)dx.$$

For a matrix  $\mathbf{A}$  with eigenvalues on or near  $\Sigma$ , this extends to an iterative method for computing  $f(\mathbf{A})$  by truncating the series:

$$f(\mathbf{A}) = \sum_{j=0}^{\infty} \alpha_j p_j(\mathbf{A}) \approx \sum_{j=0}^k \alpha_j p_j(\mathbf{A}) =: \mathbf{F}_{k+1}. \tag{2}$$

The coefficients  $\{\alpha_j\}_{j=0}^{\infty}$  and polynomials  $\{p_j(\mathbf{A})\}_{j=0}^{\infty}$  can be generated via Cauchy integrals and recurrence coefficients, respectively. Applying (1), the polynomials are generated as follows:

$$\begin{aligned} p_0(\mathbf{A}) &= \mathbf{I}, \\ p_1(\mathbf{A}) &= \frac{1}{b_0}(\mathbf{A}p_0(\mathbf{A}) - a_0p_0(\mathbf{A})), \\ p_j(\mathbf{A}) &= \frac{1}{b_{j-1}}(\mathbf{A}p_{j-1}(\mathbf{A}) - a_{j-1}p_{j-1}(\mathbf{A}) - b_{j-2}p_{j-2}(\mathbf{A})), \quad j \geq 2. \end{aligned}$$

Let  $\Gamma$  be a counterclockwise oriented curve that encloses the spectrum of  $\mathbf{A}$  such that  $f$  is analytic in a region containing  $\Gamma$ . Then,

$$\alpha_j = \int_{\Sigma} f(x)p_j(x)w(x)dx = \int_{\Sigma} \left( \frac{1}{2\pi i} \int_{\Gamma} \frac{f(z)}{z-x} dz \right) p_j(x)w(x)dx.$$

Applying a quadrature rule with nodes  $\{z_\ell\}_{\ell=1}^m$  and weights  $\{w_\ell\}_{\ell=1}^m$  to the inner integral, the coefficients can be approximated via Cauchy integrals as

$$\alpha_j \approx \int_{\Sigma} \frac{1}{2\pi i} \sum_{\ell=1}^m \frac{f(z_\ell)}{z_\ell - x} p_\ell(x)w(x)dx = - \sum_{\ell=1}^m f(z_\ell) \mathcal{C}_{\Sigma}[p_j w](z_\ell).$$

Assuming that one has access to such an approximation, the truncated series (2) can be implemented as an iteration as in Algorithm 1. The resulting method has a computable and provable geometric rate of convergence that is independent of the dimension of  $\mathbf{A}$  and governed by the classical exterior Green's function with pole at infinity from potential theory. We remark that once the coefficients  $\alpha_j$  are known, this algorithm is the same for all matrix functions.

---

**Algorithm 1:** Akhiezer iteration for matrix function approximation

---

**Input:**  $f$ ,  $\mathbf{A}$ , and functions to compute recurrence coefficients  $a_k, b_k$  and  $p_k$ -series coefficients  $\alpha_k$ .

Set  $\mathbf{F}_0 = 0$ .

**for**  $k=0,1,\dots$  **do**

- | **if**  $k=0$  **then**
- | | Set  $\mathbf{P}_0 = \mathbf{I}$ .
- | **else if**  $k=1$  **then**
- | | Set  $\mathbf{P}_1 = \frac{1}{b_0}(\mathbf{A}\mathbf{P}_0 - a_0\mathbf{P}_0)$ .
- | **else**
- | | Set  $\mathbf{P}_k = \frac{1}{b_{k-1}}(\mathbf{A}\mathbf{P}_{k-1} - a_{k-1}\mathbf{P}_{k-1} - b_{k-2}\mathbf{P}_{k-2})$ .
- | **end**
- Set  $\mathbf{F}_{k+1} = \mathbf{F}_k + \alpha_k \mathbf{P}_k$ .
- if** converged **then**
- | **return**  $\mathbf{F}_{k+1}$ .
- end**

**end**

---

A particular application pertains to the solution of Sylvester equations of the form

$$\mathbf{X}\mathbf{A} - \mathbf{B}\mathbf{X} = \mathbf{C}, \tag{3}$$

$n$	Runtime (seconds)		
	Akhiezer	Factored ADI	Bartels–Stewart
100	0.0639	0.0116	0.0060
500	0.2263	0.3939	0.2836
1000	0.4947	1.9799	1.8147
1500	0.8297	4.8730	6.5464
2000	1.3224	9.6079	21.3945

Table 1: Runtime for solving (3) to full precision where  $\mathbf{A} \in \mathbb{R}^{n \times n}$  has spectrum contained in  $[2, 3]$ ,  $\mathbf{B} \in \mathbb{R}^{n \times n}$  has spectrum contained in  $[-1.8, -0.5]$ , and  $\mathbf{C}$  is rank 2.

where the spectra of  $\mathbf{A} \in \mathbb{C}^{n \times n}$  and  $\mathbf{B} \in \mathbb{C}^{m \times m}$  lie in known intervals. If these intervals are disjoint, the unique solution  $\mathbf{X}$  to (3) is the lower left block of the matrix

$$\text{sign} \begin{pmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{C} & \mathbf{B} \end{pmatrix}, \quad (4)$$

where  $\text{sign}$  evaluates to 1 on the spectrum of  $\mathbf{A}$  and  $-1$  on the spectrum of  $\mathbf{B}$  [6].

Algorithm 1 can be directly applied to compute (4); however, its naive use requires the computation of potentially dense matrix-matrix products and blocks that are irrelevant to the approximate solution. In the case where  $\mathbf{C} = \mathbf{U}\mathbf{V}$  is low-rank, this can be circumvented by deriving an equivalent iteration for only the relevant block entry, writing updates in block form and compressing at each iteration.

Such an implementation is effectively  $O(n^2)$  for  $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{n \times n}$ , as it requires only matrix-vector products and the compression of low-rank objects. In contrast, when the coefficient matrices are dense, rational methods and direct solvers will typically be  $O(n^3)$ . We compare timings of such an implementation with the Bartels–Stewart algorithm [4] and factored Alternating-Directional-Implicit (ADI) iterations [5] in Table 1. The lower computational complexity is reflected in these timings, as the Akhiezer iteration has a shorter runtime than competing methods when  $n \geq 500$ .

## References

- [1] N. I. Akhiezer. *Elements of the Theory of Elliptic Functions*, volume 79 of *Translations of Mathematical Monographs*. American Mathematical Society, 1970.
- [2] C. Ballew and T. Trogdon. The Akhiezer iteration. *arXiv preprint 2312.02384*, 2023.
- [3] C. Ballew and T. Trogdon. A Riemann–Hilbert approach to computing the inverse spectral map for measures supported on disjoint intervals. *Studies in Applied Mathematics*, 152(1):31–72, 2024.
- [4] R. H. Bartels and G. W. Stewart. Algorithm 432 [C2]: Solution of the matrix equation  $\mathbf{AX} + \mathbf{XB} = \mathbf{C}$  [F4]. *Commun. ACM*, 15(9):820–826, September 1972.
- [5] P. Benner, R.-C. Li, and N. Truhar. On the ADI method for Sylvester equations. *Journal of Computational and Applied Mathematics*, 233(4):1035–1045, 2009.
- [6] N. J. Higham. *Functions of matrices: theory and computation*. SIAM, 2008.

- [7] Y. Saad. Iterative Solution of Indefinite Symmetric Linear Systems by Methods Using Orthogonal Polynomials over Two Disjoint Intervals. *SIAM Journal on Numerical Analysis*, 20(4):784–811, 1983.

# On the Computation of the Maximum Conic Singular Values

*Giovanni Barbarino, Nicolas Gillis, David Sossa*

## Abstract

Let  $\mathcal{C}_d$  denote the set of nonzero closed convex cones in  $\mathbb{R}^d$ . Let  $A \in \mathbb{R}^{m \times n}$  and  $(P, Q) \in \mathcal{C}_m \times \mathcal{C}_n$ . The nonconvex optimization problem

$$\min_{\substack{u \in P, \|u\| = 1, \\ v \in Q, \|v\| = 1}} \langle u, Av \rangle, \quad (1)$$

has been studied in depth in [1], mainly from a theoretical point of view. Any critical (stationary) point  $(u, v)$  of (1) satisfies the KKT optimality conditions

$$\begin{cases} P \ni u \perp (Av - \sigma u) \in P^*, \\ Q \ni v \perp (A^\top u - \sigma v) \in Q^*, \\ \|u\| = 1, \|v\| = 1, \end{cases} \quad (2)$$

for some real Lagrange multiplier  $\sigma$ , where  $P^*$  and  $Q^*$  are the dual cones of  $P$  and  $Q$ , respectively. Observe that when  $P = \mathbb{R}^m$  and  $Q = \mathbb{R}^n$ , (2) provides us the (classical) singular values of  $A$ .

The model (1) covers many interesting optimization problems. Some of them are: maximal angle between two cones [2], obtained when  $m = n$  and  $A = I_n$ , expressed by

$$\min_{\substack{u \in P, \|u\| = 1, \\ v \in Q, \|v\| = 1}} \langle u, v \rangle;$$

cone-constrained principal component analysis or Pareto singular values [3, 5], in which the two cones are the positive orthants of the respective spaces as in  $P = \mathbb{R}_+^m$  and  $Q = \mathbb{R}_+^n$ , formalized as

$$\min_{\substack{u \geq 0, \|u\| = 1, \\ v \geq 0, \|v\| = 1}} \langle u, Av \rangle;$$

nonnegative rank-one factorization matrix [4], equivalent to the Pareto singular value problem, and written as

$$\min_{u \geq 0, v \geq 0} \|M - uv^\top\|_F.$$

The above problems can be proven to be in a descending order of complexity. Since the last formulation in particular can be used to solve the Maximal Edge Biclique Problem, this leads to the conclusion that all the above models are NP-hard to solve.

We will discuss the linear algebra techniques used to reduce each problem to the following one, with a focus on sufficient conditions needed for each problem to be instead solved in polynomial time.

An exact (and thus necessarily exponential time) brute force active set algorithm is presented. Its proof of correctness is based on the observation in [1] that when we restrict the problem on the relative interior of the faces of the cones  $P$  and  $Q$ , then the relations (2) reduces to a generalized eigenvalue problem with additional constraints. This can be solved with classical techniques, with some careful handling in case of eigenvalues with relative eigenspace of dimension more than one.

We will describe the algorithm with a focus on how to cut computational cost through the study of the stationary points of the problem in order to distinguish minima from saddle points.

We compare the active set algorithm with an exact non-convex quadratic programming solver, that relies on the McCormick relaxation to solve the problem, and thus performs well in case of sparse problems.

Moreover, we show two additional iterative algorithms to solve the general problem, an alternating method with extrapolation and a fractional programming method. These are methods that are only guaranteed to converge to stationary points, and cannot certify the minimality of the solution they find.

We discuss and illustrate the use of these algorithms on several examples, as in the computation of maximal angles between the Schur cone and other cones or the computation of maximal edge bicliques.

We show how they can lead to rigorous proofs or new conjectures in special cases, such as the maximal angle between the cone of positive semidefinite matrices and the cone of nonnegative symmetric matrices.

## References

- [1] Seeger, Alberto and Sossa, David. Singular value analysis of linear maps under conic constraints. *Set-Valued and Variational Analysis*, 31 (2023).

- [2] Orlitzky, Michael. When a maximal angle among cones is nonobtuse. *Computational and Applied Mathematics*, 83 (2020).
- [3] Montanari, Andrea and Richard, Emile. Non-negative principal component analysis: Message passing algorithms and sharp asymptotics. *IEEE Trans. Informat. Th.*, 62: 1458-1484 (2016).
- [4] Gillis, Nicolas and Glineur, Fran oise. A continuous characterization of the maximum-edge biclique problem. *Journal of Global Optimization*, 58: 439-464 (2014).
- [5] Seeger, Alberto and Sossa, David. Singular value problems under non-negativity constraints. *Positivity*, 27 (2023).

# Deflation for the Half-Arrow Singular Value Decomposition

*Jesse L. Barlow, Stanley Eisenstat, Nevena Jakovčević Stor, and Ivan Slapničar*

## Abstract

A half-arrow matrix  $F$  has the form

$$F = \begin{pmatrix} \Psi & \mathbf{g} \\ \mathbf{0}^T & \rho \end{pmatrix}, \quad \mathbf{g} \in \mathbf{R}^n, \quad \rho \in \mathbf{R}, \quad (1)$$

$$\Psi = \text{diag}(\psi_1, \dots, \psi_n), \quad \psi_1 \geq \psi_2 \geq \dots \geq \psi_n \geq 0. \quad (2)$$

We consider the problem of determining which of the diagonals of  $\Psi$  are close to singular values of  $F$  and how these values can be deflated efficiently. Such deflation techniques were explored in the “conquer” stage of the divide-and-conquer bidiagonal SVD algorithms given by Jessup and Sorensen [9] and Gu and Eisenstat [7].

A version of the algorithm in [9] is coded in the LAPACK [1] subroutine **dlasd2.f** [10] as a part of the bidiagonal SVD subroutine **dbdsbc.f** [1, p.208].

The SVD version of the Cauchy interlace theorem [6, Corollary 8.6.3] states that the singular values  $\sigma_1, \dots, \sigma_{n+1}$  of  $F$  satisfy

$$\sigma_j \geq \psi_j \geq \sigma_{j+1}, \quad j = 1, \dots, n. \quad (3)$$

Interpreting a result in [13, p.95], Jessup and Sorensen [9] point to three cases where  $\psi_j$  is a singular value of  $F$ :

- **Case I:**  $g_j = \mathbf{e}_j^T \mathbf{g} = 0$ , then  $(\psi_j, \mathbf{e}_j, \mathbf{e}_j)$  is a singular triplet of  $F$ ;
- **Case II:**  $\psi_j = 0$ , so we let  $G_{n+1,j}$  be a Givens rotation affecting rows  $j$  and  $n+1$  whose non-trivial part is defined by

$$\begin{pmatrix} c & -s \\ s & c \end{pmatrix} \begin{pmatrix} g_j \\ \rho \end{pmatrix} = \begin{pmatrix} 0 \\ \hat{\rho} \end{pmatrix}, \quad c^2 + s^2 = 1, \quad \hat{\rho} = \pm \sqrt{g_j^2 + \rho^2},$$

and we have that

$$\tilde{F} = G_{n+1,j} F$$

has the singular triplet  $(0, \mathbf{e}_j, \mathbf{e}_j)$ ;

- **Case III:**  $\psi_i = \psi_j$  for some  $i \neq j$ , so we let  $G_{ij}$  be a Givens rotation affecting rows  $i$  and  $j$  where the non-trivial part of  $G_{ij}$  is defined by

$$\begin{pmatrix} c & s \\ -s & c \end{pmatrix} \begin{pmatrix} g_i \\ g_j \end{pmatrix} = \begin{pmatrix} \hat{g}_j \\ 0 \end{pmatrix}, \quad c^2 + s^2 = 1, \quad \hat{g}_j = \pm \sqrt{g_j^2 + g_{j+1}^2}$$

and we have that

$$\tilde{F} = G_{ij} F G_{ij}^T$$

has the singular triplet  $(\psi_j, \mathbf{e}_j, \mathbf{e}_j)$ .

In all three cases, the computation of the SVD of  $F$  is reduced to that of computing the SVD of a lower dimensional half-arrow matrix. If none of these deflations is possible for any  $j$ , then from [13, p.95], we have the strict interlacing property

$$\sigma_j > \psi_j > \sigma_{j+1}, \quad j = 1, \dots, n. \quad (4)$$

The deflation strategies in [9, 7] are based upon the idea that one of these three cases applies to a matrix near  $F$ . We model these strategies as follows: we compute a value  $\gamma_F$  such that  $\|F\|_2 \leq \gamma_F \leq \sqrt{2}\|F\|_2$ , and let  $\tau$  be a small value, usually  $\mathcal{O}(\varepsilon_M)$  where  $\varepsilon_M$  is the machine unit. In some applications,  $\tau$  may be an acceptable level of error.

Corresponding to the three cases for when  $\psi_j$  is a singular value of  $F$ , we can deflate  $g_j$  in the following three cases:

1. If

$$|g_j| \leq \tau\gamma_F \quad (5)$$

we simply set  $g_j$  to zero;

2. If

$$\frac{\psi_j |g_j|}{\sqrt{g_j^2 + \rho^2}} \leq \tau\gamma_F,$$

then we apply the Givens rotation  $G_{n+1,j}$  to rows  $n+1$  and  $j$  setting  $g_j$  to zero producing

$$\tilde{F} + \delta F_{n+1,j} = G_{n+1,j} F, \quad \|\delta F_{n+1,j}\|_2 \leq \sqrt{2}\tau\gamma_F \quad (6)$$

where  $\tilde{F}$  is a half-arrow matrix with  $\Psi$  unchanged;

3. If

$$|\delta_{ij}| \leq \tau\gamma_F, \quad \delta_{ij} = \frac{g_j g_i}{g_i^2 + g_j^2} (\psi_i - \psi_j) \quad (7)$$

and  $|g_j| \leq |g_i|$ , then we apply the Givens rotation  $G_{ij}$  to rows  $i$  and  $j$  setting  $g_j$  to zero producing

$$\tilde{F} + \delta F_{ij} = G_{ij} F G_{ij}^T, \quad \|\delta F_{ij}\|_2 \leq \sqrt{2}\tau\gamma_F \quad (8)$$

where  $\tilde{F}$  is again a half-arrow matrix with  $\Psi$  unchanged. If (7) holds and  $|g_j| > |g_i|$ , we set  $g_i$  to zero in an analogous manner.

The deflations (5) and (8) are discussed in [9, 7] and the deflation (6) is discussed in [9].

We enhance the approach in [9, 7] and in the LAPACK routine **dlassd2.f** [10] by producing a better deflation algorithm that is still  $\mathcal{O}(n)$  operations. We also show that if for a particular value of  $j$ ,  $g_j$  cannot be deflated by (5) or by (6) or by (8) for any  $i \neq j$ , then

$$\sigma_j - \sigma_{j+1} > \tau\gamma_F / \sqrt{2n+1}. \quad (9)$$

However, the only algorithm we give with that guarantee for all  $j$  has a worst case complexity proportional to  $n^2$ . If we weaken these conditions, so that there is no index  $i$  such that  $|i - j| < q$ , and we have (8), then

$$\sigma_j - \sigma_{j+1} > \sqrt{2}\tau^2 q \gamma_F + \mathcal{O}(\tau^4 q^2 \gamma_F). \quad (10)$$

The algorithm we recommend is a heuristic proposed here with worst case complexity proportional to  $n$ , the same asymptotic complexity as the LAPACK procedure, but with better deflation guarantees. It achieves (10) with  $q = 2$  for all  $j$ . Bounds similar to (9) and (10) are not possible for the singular values of deflated structure matrices, for instance, there are no such bounds for bidiagonal matrices.

In light of work by Demmel and Gragg [4] that formulated an algorithm to compute the nonzero singular values of  $F$  to near relative accuracy, we formulate and analyze versions of the deflations in (5) and (8) that preserve relative accuracy in the singular values.

By choosing  $\gamma_F$  to be within a constant factor of  $\|F\|_2$ , these deflations produce no more error in the singular values than would be expected of a normwise backward stable algorithm for finding the SVD of  $F$ . However, for algorithms to compute the SVD of  $F$ , deflation gives us dimension reduction and speeds up the algorithms in [9] and [7]. The LAPACK routine **dlassd2.f** uses only the first and third types of deflation.

Two other applications for this kind of deflation have been investigated. The first is in SVD-based regularization approaches given in [8, §4.3] and [11]. The second is in the implementation of a Krylov-Schur implementation [2, 12] of the Golub-Kahan-Lanczos SVD algorithm [5].

This is a continuation of work in [3].

## References

- [1] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK User's Guide: Third Edition*. SIAM Publications, Philadelphia, PA, 1999.
- [2] J. Baglama and L. Reichel. Augmented implicitly restarted Lanczos bidiagonalization methods. *SIAM J. Sci. Computing*, 27(1):19–42, 2005.
- [3] J. Barlow, S.C. Eisenstat, N. Jakovčević Stor, and I. Slapničar. Deflation for the symmetric arrowhead and diagonal-plus-rank-one eigenvalue problems. *SIAM J. Matrix Anal. Appl.*, 43(2):681–709, 2022.
- [4] J.W. Demmel and W.B. Gragg. On computing accurate singular values and eigenvalues of matrices with acyclic graphs. *Linear Algebra and Its Applications*, 185:203–217, 1993.
- [5] G.H. Golub and W.M. Kahan. Calculating the singular values and pseudoinverse of a matrix. *SIAM J. Num. Anal. Ser. B*, 2:205–224, 1965.
- [6] G.H. Golub and C.F. Van Loan. *Matrix Computations*. The Johns Hopkins Press, Baltimore, MD, Fourth edition, 2013.
- [7] M. Gu and S.C. Eisenstat. A divide-and-conquer algorithm for the bidiagonal svd. *SIAM Journal on Matrix Analysis and Applications*, 16(1):79–92, 1995.
- [8] P.C. Hansen. *Discrete Inverse Problems: Insight and Algorithms*. SIAM Publications, Philadelphia, PA, 2010.
- [9] E. Jessup and D. Sorensen. A parallel algorithm for computing the Singular Value Decomposition of a matrix. *SIAM J. Matrix Anal. Appl.*, 15(2):530–548, 1994.

- [10] LAPACK Project. Lapack subroutine dlasd2.f. URL Page.  
[https://netlib.org/lapack/explore-html/d1/d83/dlasd2\\_8f\\_source.html](https://netlib.org/lapack/explore-html/d1/d83/dlasd2_8f_source.html).
- [11] B.W. Rust. Truncating the singular value decomposition for ill-posed problems. Technical NISTIR, Mathematical and Computational Sciences Division, NIST, Gaithersburg, MD, 1998.
- [12] M. Stoll. A Krylov-Schur approach to the truncated SVD. *Linear Algebra and Its Applications*, 436(8):2795–2806, 2012.
- [13] J.H. Wilkinson. *The Algebraic Eigenvalue Problem*. Oxford University Press, London, 1965.

# Learning Globally Stable Dynamics — a Matrix-theoretic Perspective

*Peter Benner, Pawan K. Goyal, Siddarth Mamidisetti, Igor Pontes Duff, Süleyman Yıldız*

Abstract

## 1 Motivation

The identification of dynamical systems from data has been considered for several decades, basically ever since *Cybernetics* emerged as a discipline from the wider research area of systems theory [Wiener (1948)]. In systems and control theory, system identification is an important and widely used technique in computer-aided control system design, available in any relevant software package, see, e.g., [Ljung (1999), Sima and Benner (2007), Benner et al. (2010)]. Nevertheless, most available identification tools are for linear systems, and do not necessarily consider constraints like *stability* or other physical properties. On the other hand, due to the advance of machine and deep learning methods, learning dynamical systems from data, in particular nonlinear systems, has recently become a field of renewed and massively growing interest. This is due, on the one hand, to the vast availability of measurement data in areas like the Earth system (weather and climate), biology, sociology, traffic etc., where traditionally no first principle mathematical models are available, but models are needed for prediction, while, on the other hand, the massive advancement in computational power nowadays allows to study large data sets using methods from artificial intelligence.

Nevertheless, the trajectories of time-dependent problems are often constrained by underlying physical principles that are usually not respected by classical black-box methods in machine and deep learning. Thus, physics-informed and physics-enhanced or "scientific" machine learning have emerged as new subdisciplines in the computational sciences and engineering and applied mathematics. Here, we will study in particular the stability of learned dynamical systems. In other words, we answer the question how to *guarantee* that the learned model of a dynamical system has desired stability properties, where we assume that some prior knowledge about the expected stability class is available. We show that stability constraints can be hard-coded into the learning model to guarantee, e.g., (asymptotic/Lyapunov/exponential) stability of linear systems, global asymptotic stability of nonlinear systems, and global stability of Hamiltonian systems. The used ideas stem from partially simple and obvious results in the theory of matrices and tensors. We will show that explicit parameterizations of the learned operators (matrices and tensors) defining the dynamical systems lead to constrained least-squares problems that can be solved using optimization routines that are now widely available in all software stacks for machine learning.

Exemplarily, we will demonstrate this approach for linear, uncontrolled, systems. In the talk, we will focus on the more challenging problems for nonlinear systems.

## 2 Learning Stable Linear Systems

Consider uncontrolled linear systems of the form

$$\dot{x}(t) = Ax(t), \quad x(0) = x_0, \tag{1}$$

where  $A \in \mathbb{R}^{n \times n}$ ,  $x(t) \in \mathbb{R}^n$  is the state of the system at time  $t \geq 0$ ,  $\dot{x}$  denotes the derivative of  $x$  with respect to time, and  $x_0 \in \mathbb{R}^n$  is the initial value.

If time-series data for  $x(t)$  and  $\dot{x}(t)$  are available, i.e., we have measured or computed vectors

$$x_0 = x(0), \quad x_1 = x(t_1), \quad \dots, \quad x_\ell = x(t_\ell)$$

and

$$x'_0 = \dot{x}(0), \quad x'_1 = \dot{x}(t_1), \quad \dots, \quad x'_\ell = \dot{x}(t_\ell)$$

at a sequence of time points  $t_0 = 0 < t_1 < t_2 < \dots < t_\ell$ , then a very simple way to infer the matrix  $A$  from the available data is to set up a linear least-squares problem using the data matrices

$$\begin{aligned} X &= [x_0, \dots, x_\ell] \in \mathbb{R}^{n \times \ell+1}, \\ X' &= [x'_0, \dots, x'_\ell] \in \mathbb{R}^{n \times \ell+1}. \end{aligned}$$

For (1), a possible formulation of an *operator inference problem* (*OpInf*) is then

$$A_* := \arg \min_A \|X' - AX\|_F^2 + \mathcal{R}(A) \tag{2}$$

with a potential regularization term  $\mathcal{R}(A)$ , e.g., for classical Tikhonov regularization (aka “ridge regression” in the machine learning community)  $\mathcal{R}(A) = \beta \|A\|_F^2$ , where  $\beta > 0$  would be a regularization parameter. Here, one might also use a sparsity-promoting norm, e.g.,  $\mathcal{R}(A) = \beta \|A\|_q$ ,  $q = 0, 1$ . Note that for  $\beta = 0$ , or in general,  $\mathcal{R}(A) \equiv 0$ , (1) represents a standard linear least-squares problem, as the Frobenius norm turns into the standard Euclidian vector norm once  $X' - AX$  is vectorized. From this, it is also obvious that this becomes a classical overdetermined system once  $\ell \geq n$ . Then minimization problem (1) can be solved using classical solution methods for linear least-squares problems like pivoted QR factorization or singular value decomposition (SVD).

The OpInf framework described so far has seen numerous applications and extensions in recent years. One of its major drawbacks still is that the inferred models can not be guaranteed to be stable (regarding local or global, asymptotic or Lyapunov stability) even if it is known that the data-generating model has a certain stability property. Some stability promoting weak enforcement strategies have been suggested, see, e.g., [Kaptanoglu et al. (2021)], but so far no certified stable models could be produced using the OpInf problem (2). Our goal is to introduce a parametrization of stability directly in the least-squares problem and in this way to obtain guaranteed stable models, following [Goyal et al. (2023b)]. Note that a very similar idea is also used in model reduction methods for port-Hamiltonian systems [Schwerdtner and Voigt (2023)].

The inference of asymptotically stable linear models with guarantee is based on the following somewhat surprising result, that was stated in [Gillis and Sharma (2017)], whereby related partial results can also be found in earlier literature. First, note that asymptotic stability of linear systems of the form (1) is fully characterized by the spectrum of  $A$ ,  $\Lambda(A)$ , and in particular by the property that  $\Lambda(A) \subset \mathbb{C}^-$ , i.e., that the spectrum of  $A$  is fully contained in the open left half of the complex plane. The proof of the theorem is entirely based on this characterization.

**Theorem 2.1** ([Gillis and Sharma (2017)]). *A matrix  $A \in \mathbb{R}^{n \times n}$  is asymptotically stable (Hurwitz, Lyapunov stable) if and only if it can be represented as*

$$A = (J - R)Q,$$

where  $J = -J^T$  and  $R = R^T$ ,  $Q = Q^T$  are both positive definite.

Now, it is straightforward to replace the linear least-squares problem (2) by the following inference problem:

$$(J_*, R_*, Q_*):=\arg\min_{\substack{J=-J^T \\ R=R^T\succeq 0 \\ Q=Q^T\succeq 0}} \|X' - (J-R)QX\|_F^2 + \mathcal{R}(J, Q, R). \quad (3)$$

Unfortunately, this problem is relatively hard to solve due to the positive definiteness constraints. Therefore, in [Goyal et al. (2023b)], we suggest to use the following parametrization:

$$J = S - S^T, \quad R = L^T L, \quad Q = K^T K,$$

where  $S \in \mathbb{R}^{n \times n}$  is a general square matrix and  $L, K \in \mathbb{R}^{n \times n}$  are upper-triangular matrices (or Cholesky factors). Then the OpInf problem for linear systems becomes

$$(S_*, L_*, K_*):=\arg\min_{\substack{L, K \text{ upper} \\ \text{triangular}}} \left( \|X' - (S - S^T - L^T L)K^T K X\|_F^2 + \mathcal{R}(L, K, S) \right). \quad (4)$$

If this problem can be solved, the obtained matrix

$$A_* = (S_* - S_*^T - L_*^T L_*) K_*^T K_*$$

is guaranteed to be asymptotically stable due to Theorem 2.1 and solves (2) under the asymptotic stability constraint. The price paid for inferring a model with stability certificate is that even for a zero regularizer, the inference problem (4) is nonlinear in the entries of  $L, K$ . Fortunately, with the advance of machine learning methods, such problems can be solved using (stochastic) gradient descent methods implemented in tools like PyTorch.<sup>1</sup>

In our talk, we will describe how this result and the related stability-guaranteed OpInf problem can be extended to controlled systems  $\dot{x} = Ax + Bu$ , where  $u$  is a control input [Pontes Duff et al. (2024)] and to parameter dependent systems, where  $A = A(\mu)$  depends on a parameter vector  $\mu \in \mathbb{R}^q$ .

### 3 Outlook: Nonlinear Systems

While for linear systems, local and global stability as well as asymptotic, Lyapunov, and exponential stability concepts are equivalent, these have to be distinguished for nonlinear systems. Obviously, using the parametrization of  $A$  from (4) to an operator inference problem for a nonlinear system with linear part  $Ax(t)$ , the inferred system will be locally Lyapunov stable with Lyapunov function

$$V(x) := \frac{1}{2}x^T Q x.$$

In [Goyal et al. (2023a)], we additionally study the identification of globally Lyapunov stable quadratic systems as well as systems with bounded domain of attraction. This is again based on explicit parameterizations of matrices and tensors. These results will be presented in the talk as well as ideas on how to guarantee global stability of Hamiltonian systems, where we employ techniques from deep learning and elementary properties of symplectic matrices [Goyal et al. (2023c)]. The theoretical findings will be illustrated by several numerical examples.

---

<sup>1</sup><https://pytorch.org/>

## References

- [Benner et al. (2010)] Benner, P., Kressner, D., Sima, V., and Varga, A. (2010). Die SLICOT-Toolboxen für MATLAB. *at-Automatisierungstechnik*, 58(1), 15–25.
- [Gillis and Sharma (2017)] Gillis, N. and Sharma, P. (2017). On computing the distance to stability for matrices using linear dissipative Hamiltonian systems. *Automatica*, 85, 113–121.
- [Goyal et al. (2023a)] Goyal, P., Pontes Duff, I., and Benner, P. (2023a). Guaranteed stable quadratic models and their applications in SINDy and operator inference. e-print arXiv:2308.13819.
- [Goyal et al. (2023b)] Goyal, P., Pontes Duff, I., and Benner, P. (2023b). Stability-guaranteed learning of linear models. e-print arXiv:2301.10060.
- [Goyal et al. (2023c)] Goyal, P., Benner, P., and Yıldız, S. (2023c). Deep learning for structure-preserving universal stable Koopman-inspired embeddings for nonlinear canonical Hamiltonian dynamics. e-print arXiv:2308.13835.
- [Kaptanoglu et al. (2021)] Kaptanoglu, A., Callaham, J., Aravkin, A., Hansen, C., and Brunton, S. (2021). Promoting global stability in data-driven models of quadratic nonlinear dynamics. *Physical Review Fluids*, 6(9), 094401.
- [Ljung (1999)] Ljung, L. (1999). *System Identification – Theory for the User*. Prentice-Hall, Upper Saddle River, NJ, 2nd edition.
- [Peherstorfer and Willcox (2016)] Peherstorfer, B. and Willcox, K. (2016). Data-driven operator inference for nonintrusive projection-based model reduction. *Computer Methods in Applied Mechanics and Engineering*, 306, 196–215.
- [Pontes Duff et al. (2024)] Pontes Duff, I., Goyal, P., and Benner, P. (2024). Stability-certified learning of control systems with quadratic nonlinearities. *IFAC-PapersOnLine*, 58(17), 151–156..
- [Schwerdtner and Voigt (2023)] Schwerdtner, P. and Voigt, M. (2023). SOBMOR: Structured optimization-based model order reduction *SIAM Journal on Scientific Computing* 45(2), A502–A529.
- [Sima and Benner (2007)] Sima, V. and Benner, P. (2007). Fast system identification and model reduction solvers. In B.R. Andrievsky and A.L. Fradkov (eds.), *Preprints 9th IFAC Workshop “Adaptation and Learning in Control and Signal Processing” (ALCOSP 2007)*, August, 29-31, 2007, Saint Petersburg, Russia.
- [Wiener (1948)] Wiener, N. (1948). *Cybernetics: Or Control and Communication in the Animal and the Machine*. MIT Press, Cambridge, Massachusetts.

## Abstract

We study algorithms for approximating the spectral density of a symmetric matrix  $\mathbf{A} \in \mathbb{R}^{n \times n}$  that is accessed through matrix-vector products. By combining an existing Chebyshev polynomial moment matching method with a deflation step that approximately projects off the largest magnitude eigendirections of  $\mathbf{A}$  before estimating the spectral density, we give an  $\epsilon\sigma_\ell(\mathbf{A})$  error approximation in the Wasserstein-1 metric using  $O(\ell \log n + 1/\epsilon)$  matrix-vector products, where  $\sigma_\ell(\mathbf{A})$  is the  $\ell^{\text{th}}$  largest singular value of  $\mathbf{A}$ . When  $\mathbf{A}$  exhibits fast singular value decay, this can be much stronger than prior work, which gives error  $\epsilon\sigma_1(\mathbf{A})$  using  $O(1/\epsilon)$  matrix-vector products. We also show that our bound is nearly tight:  $\Omega(\ell + 1/\epsilon)$  matrix-vector products are required to achieve error  $\epsilon\sigma_\ell(\mathbf{A})$ .

We further show that the popular Stochastic Lanczos Quadrature (SLQ) method matches the above bound, even though SLQ itself is parameter-free and performs no explicit deflation. This explains the strong practical performance of SLQ, and motivates a simple variant that achieves an even tighter error bound. Our error bound for SLQ leverages an analysis that views it as an implicit polynomial moment matching method, along with recent results on low-rank approximation with single-vector Krylov methods. We use these results to show that SLQ can perform implicit deflation as part of moment matching.

## 1 Introduction

Given a symmetric matrix  $\mathbf{A} \in \mathbb{R}^{n \times n}$  with eigenvalues  $\lambda_1(\mathbf{A}), \dots, \lambda_n(\mathbf{A})$ , the *spectral density* of  $\mathbf{A}$  is defined as:

$$s_{\mathbf{A}}(x) = \frac{1}{n} \sum_{i=1}^n \delta(x - \lambda_i(\mathbf{A})),$$

where  $\delta(\cdot)$  is the Dirac delta function. The spectral density  $s_{\mathbf{A}}$  can be computed directly by performing a full eigendecomposition of  $\mathbf{A}$  in  $O(n^\omega)$  time, for  $\omega \approx 2.37$  being the exponent of fast matrix multiplication. However, when  $\mathbf{A}$  is very large or where  $\mathbf{A}$  can only be accessed through a small number of queries, we often want to approximate  $s_{\mathbf{A}}$  by some  $\tilde{s}_{\mathbf{A}}$  such that  $\tilde{s}_{\mathbf{A}}$  and  $s_{\mathbf{A}}$  are close in some metric. Spectral density estimation is applied throughout the sciences [Ski89, SR94, STBB17, SRS20], network science [FDBV01, EG17, DBB19], machine learning and deep learning in particular [RL18, PSG18, MM19, GKX19], numerical linear algebra [DNPS16, LXES19], and beyond. In this work, we focus on the Wasserstein-1 (i.e., earth mover's) distance,  $W_1(s_{\mathbf{A}}, \tilde{s}_{\mathbf{A}})$ , which has been studied in a number of recent works giving formal approximation guarantees for SDE [CTU21, BKM22, CTU22]. Moreover,  $\mathbf{A}$  will be accessed only through matrix vector queries of the form  $\mathbf{A}\mathbf{v}$  for any query vector  $\mathbf{v}$ . Most state-of-the-art matrix-vector query algorithms for SDE are based on Krylov subspace methods that fall into two general classes.

**Moment Matching.** The first class of methods approximates  $s_{\mathbf{A}}$  by approximating its polynomial moments. I.e.,  $\mathbb{E}_{s_{\mathbf{A}}}[p(x)] = \frac{1}{n} \sum_{i=1}^n p(\lambda_i(\mathbf{A})) = \frac{1}{n} \text{tr}(p(\mathbf{A}))$ , where  $p$  is a low-degree polynomial. We can employ stochastic trace estimation methods like Hutchinson's method [Gir87, Hut90] to approximate this trace using just a small number of matrix-vector products with  $p(\mathbf{A})$  and in turn

---

<sup>1</sup>Based on paper to appear at SODA 2025. Preprint available at [BJM<sup>+</sup>24]

$\mathbf{A}$ , since if  $p$  has degree  $k$ , a single matrix-vector product with  $p(\mathbf{A})$  can be performed using  $k$  matrix vector products with  $\mathbf{A}$ . After approximating the moments for a set of low-degree polynomials (e.g., the first  $k$  monomials, or the first  $k$  Chebyshev polynomials), we can let  $\tilde{s}_{\mathbf{A}}$  be a distribution that matches these moments as closely as possible, and thus should closely match  $s_{\mathbf{A}}$ . Moment matching methods include the popular Kernel Polynomial Method (KPM) [SR94, Wan94, WWA06] and its variants [CPB10, LSY16, BKM22, Che23]. Braverman et al. [BKM22] analyze a Chebyshev Moment Matching method, which can be thought of as a simple variant of KPM, showing that the method can compute  $\tilde{s}_{\mathbf{A}}$  satisfying  $W_1(s_{\mathbf{A}}, \tilde{s}_{\mathbf{A}}) \leq \epsilon \cdot \|\mathbf{A}\|_2$  with probability  $\geq 1 - \delta$  using just  $O(b/\epsilon)$  matrix vector products, where  $b = \max(1, \frac{1}{n\epsilon^2} \log^2 \frac{1}{\epsilon\delta} \log^2 \frac{1}{\epsilon})$ . Note that  $b = 1$  in the common case when  $\epsilon = \tilde{\Omega}(1/\sqrt{n})$ . Here  $\|\mathbf{A}\|_2$  denotes the spectral norm of  $\mathbf{A}$  – i.e., its largest eigenvalue magnitude. They prove a similar guarantee for KPM itself, but with a worse dependence on  $\epsilon$ .

**Lanczos-Based Methods.** This class of methods computes a small number of approximate eigenvalues of  $\mathbf{A}$  using the Lanczos method, and lets  $\tilde{s}_{\mathbf{A}}$  be a distribution supported on these eigenvalues, with appropriately chosen probability mass placed at each. The canonical method of this form is Stochastic Lanczos Quadrature (SLQ) [CTU21, GM09]. Many other variants have also been studied. Some place probability mass not just at the approximate eigenvalues, but on Gaussian or other simple distributions centered at these eigenvalues [LG82, BRP92, LSY16, HHK72]. Chen et al. [CTU21, CTU22] prove that the Lanczos-based SLQ method gives essentially the same approximation bound as [BKM22]: error  $\epsilon \cdot \|\mathbf{A}\|_2$  using  $O(1/\epsilon)$  matrix-vector products when  $\epsilon = \tilde{\Omega}(1/\sqrt{n})$ <sup>2</sup>.

## 2 Our Results

Our main contribution is to show that both moment matching and Lanczos based methods for SDE can achieve improved bounds on  $W_1(s_{\mathbf{A}}, \tilde{s}_{\mathbf{A}})$  that depend on  $\sigma_{l+1}(\mathbf{A})$ , the  $(l+1)^{st}$  largest singular value of  $\mathbf{A}$  for some parameter  $l$ , instead of  $\|\mathbf{A}\|_2$ . For matrices that exhibit spectral decay and thus have  $\sigma_{l+1}(\mathbf{A}) \ll \sigma_1(\mathbf{A}) = \|\mathbf{A}\|_2$ , our bounds can be much stronger than the bound  $W_1(s_{\mathbf{A}}, \tilde{s}_{\mathbf{A}}) \leq \epsilon \cdot \|\mathbf{A}\|_2$  achieved in prior work, which roughly corresponds to estimating each eigenvalue to average error  $\epsilon \cdot \|\mathbf{A}\|_2$ . We also provide a lower bound showing that our bounds are near optimal upto some logarithmic factors.

### 2.1 Improved SDE via Moment Matching with Explicit Deflation

Our first contribution is a modification of the moment matching method of [BKM22] that first ‘deflates’ off any eigenvalue of  $\mathbf{A}$  with magnitude significantly larger than  $\sigma_{l+1}(\mathbf{A})$ , before estimating the spectral density. Eigenvalue deflation is widely applied throughout numerical linear algebra to problems like linear system solving [BEPW98, FV01, GOSS16, FTU23], trace estimation [GSO17, Lin17, MMMW21], norm estimation [MNS<sup>+</sup>18], and beyond [CS97]. Specifically, the method uses a block Krylov subspace method to first compute highly accurate approximations to the  $p$  largest magnitude eigenvalues of  $\mathbf{A}$ , for some  $p \leq l$ , along with an orthonormal matrix  $\mathbf{Z} \in \mathbb{R}^{n \times p}$  with columns approximating the corresponding eigenvectors. It uses moment matching to estimate the spectral density of  $\mathbf{A}$  projected away from these approximate eigendirections  $(\mathbf{I} - \mathbf{ZZ}^T)\mathbf{A}(\mathbf{I} - \mathbf{ZZ}^T)$ , achieving error  $\epsilon\sigma_{l+1}(\mathbf{A})$  since this matrix has spectral norm bounded by  $O(\sigma_{p+1}(\mathbf{A})) = O(\sigma_{l+1}(\mathbf{A}))$  if  $\mathbf{Z}$  is sufficiently accurate. It then modifies this approximate density to account for the probability

---

<sup>2</sup>The notations  $\tilde{O}$  and  $\tilde{\Omega}$  means some logarithmic factors are present.

mass at the top  $p$  eigenvalues. While block Krylov methods are well understood for related tasks like eigenvalue and eigenvector computation [Par98, Tro18], low-rank approximation [HMT11, MM15], singular value approximation [MM15, MNS<sup>+</sup>18, BN23], linear system solving [LSY98, Saa03], our work requires a careful analysis of eigenvalue/eigenvector approximation with these methods that may be of independent interest. Overall, the above approach gives the following result:

**Theorem 1** (SDE with Explicit Deflation). *For any  $\epsilon \in (0, 1)$ ,  $l \in [n]$ , and any constants  $c_1, c_2 > 0$ , Algorithm 1 of [BJM<sup>+</sup>24] performs  $O(l \log n + \frac{b}{\epsilon})$  matrix-vector products with  $\mathbf{A}$  where  $b = \max(1, \frac{1}{n\epsilon^2} \log^2 \frac{n}{\epsilon} \log^2 \frac{1}{\epsilon})$  and computes a probability density function  $\tilde{s}_{\mathbf{A}}$  such that, with probability at least  $1 - \frac{1}{n^{c_1}}$ ,*

$$W_1(s_{\mathbf{A}}, \tilde{s}_{\mathbf{A}}) \leq \epsilon \cdot \sigma_{l+1}(\mathbf{A}) + \frac{\|\mathbf{A}\|_2}{n^{c_2}}.$$

The additive error  $\frac{\|\mathbf{A}\|_2}{n^{c_2}}$  can be thought of as negligible – comparable e.g., to round-off error when directly computing  $s_{\mathbf{A}}$  using a full eigendecomposition in finite precision arithmetic [BGVKS22].

We further show that our algorithm is optimal amongst all matrix-vector query algorithms, up to logarithmic factors and the negligible additive error term. Our proof leverages an existing lower bound for distinguishing Wishart matrices of different ranks, previously used to give matrix-vector query lower bounds for the closely related problem of eigenvalue estimation [SW23]. Formally:

**Theorem 2** (SDE Lower Bound). *Any (possibly randomized) algorithm that given symmetric  $\mathbf{A} \in \mathbb{R}^{n \times n}$  outputs  $\tilde{s}_{\mathbf{A}}$  such that, with probability at least  $1/2$ ,  $W_1(s_{\mathbf{A}}, \tilde{s}_{\mathbf{A}}) \leq \epsilon \sigma_{l+1}(\mathbf{A})$  for  $\epsilon \in (0, 1)$  and  $l \in [n]$  must make  $\Omega(l + \frac{1}{\epsilon})$  (possibly adaptively chosen) matrix-vector queries to  $\mathbf{A}$ .*

## 2.2 Implicit Deflation Bounds for Stochastic Lanczos Quadrature

Our second contribution is to show that the popular Stochastic Lanczos Quadrature (SLQ) method for SDE [LSY16, CTU21] nearly matches the improved error bound of Theorem 1 for any choice of  $l$ , even though SLQ is ‘parameter-free’ and performs no explicit deflation step. This result helps to justify the strong practical performance of SLQ and the observed ‘spectrum adaptive’ nature of this method as compared to standard moment matching-based methods like KPM [CTU21].

A key idea used to achieve this bound is to view SLQ as an implicit moment matching method as in [CTU21, CTU22], and to analyze it similarly to KPM and other explicit moment matching methods. We combine this analysis approach with recent work on low-rank approximation with single-vector (i.e., non-block) Krylov methods [MMM24] to show that SLQ can perform ‘implicit deflation’ as part of moment matching to achieve the improved error bound. Formally, we have:

**Theorem 3** (SDE with SLQ). *Let  $l \in [n]$ , and  $\epsilon, \delta \in (0, 1)$ . Let  $g_{\min} = \min_{i \in [l]} \frac{\sigma_i(\mathbf{A}) - \sigma_{i+1}(\mathbf{A})}{\sigma_i(\mathbf{A})}$  and  $\kappa = \frac{\|\mathbf{A}\|_2}{\sigma_{l+1}(\mathbf{A})}$ . SLQ run for  $m = O(l \log \frac{1}{g_{\min}} + \frac{1}{\epsilon} \log \frac{n \cdot \kappa}{\delta})$  iterations performs  $m$  matrix vector products with  $\mathbf{A}$  and outputs a probability density function  $\tilde{s}_{\mathbf{A}}$  such that, with probability at least  $1 - \delta$ ,*

$$W_1(s_{\mathbf{A}}, \tilde{s}_{\mathbf{A}}) \leq \tilde{O} \left( \epsilon \cdot \sigma_{l+1}(\mathbf{A}) + \frac{\sigma_{l+1}(\mathbf{A})}{\sqrt{n}} + \frac{l}{n} \|\mathbf{A}\|_2 \right).$$

Theorem 3 essentially matches our result for moment matching with explicit deflation (Theorem 1) up to some small caveats, discussed below. First, the number of matrix vector products has a logarithmic dependence on the minimum gap  $g_{\min}$  amongst the top  $l$  singular values as well as the condition number  $\kappa = \frac{\|\mathbf{A}\|_2}{\sigma_{l+1}(\mathbf{A})}$ . The dependence on the minimum gap is inherent, as non-block

Krylov methods like SLQ require a dependence on  $g_{\min}$  in order to perform deflation/low-rank approximation [MMM24]. We note that, in practice,  $g_{\min}$  is generally not too small. Also, by adding a random perturbation to  $\mathbf{A}$  with spectral norm bounded by  $\frac{\|\mathbf{A}\|_2}{\text{poly}(n)}$ , one can ensure that both  $g_{\min} \geq \frac{1}{\text{poly}(n)}$  and  $\kappa \leq \text{poly}(n)$  with high probability, and thus replace the  $O(l \log \frac{1}{g_{\min}})$  term with an  $O(l \log n)$  and the  $O(\frac{\log(n\kappa)}{\epsilon})$  term with  $O(\frac{\log n}{\epsilon})$ , matching Theorem 1. See e.g., [MMM24].

Second, Theorem 3 has an additional error term of size  $\tilde{O}(\sigma_{l+1}(\mathbf{A})/\sqrt{n})$ . This term is lower order whenever  $\epsilon = \tilde{\Omega}(1/\sqrt{n})$ . Further, we believe that this term, along with the dependence on  $g_{\min}$  can be removed by using a variant on SLQ that is popular in practice, where the densities output by multiple independent runs of the method are averaged together to produce  $\tilde{s}(\mathbf{A})$ .

Finally, Theorem 3 has an additional error term of size  $\tilde{O}(\|\mathbf{A}\|_2 \cdot l/n)$ . In the natural case when we run for  $m \ll n$  iterations and thus  $l \ll n$ , this term will be small. However, it cannot be avoided: even for a matrix with rank  $\leq l$  with well-separated eigenvalues, while the Lanczos method will converge to near-exact approximations to these eigenvalues (with error bounded by  $\frac{\|\mathbf{A}\|_2}{n^c}$ ), the probability distribution output by SLQ will not place mass exactly  $1/n$  at these approximate eigenvalues and thus will not achieve SDE error  $O(\frac{\|\mathbf{A}\|_2}{n^c})$ .

This limitation motivates us to introduce a simple variant of SLQ, which we call *variance reduced SLQ*, which places mass exactly  $1/n$  at any eigenvalue computed by Lanczos that has converged to sufficiently small error. This variant gives the following stronger error bound:

**Theorem 4** (SDE with Variance Reduced SLQ). *Let  $l \in [n]$ , and  $\epsilon, \delta \in (0, 1)$ . Let  $g_{\min} = \min_{i \in [l]} \frac{\sigma_i(\mathbf{A}) - \sigma_{i+1}(\mathbf{A})}{\sigma_i(\mathbf{A})}$  and  $\kappa = \frac{\|\mathbf{A}\|_2}{\sigma_{l+1}(\mathbf{A})}$ . Algorithm 5 of [BJM<sup>+</sup>24] run for  $m = O(l \log \frac{1}{g_{\min}} + \frac{1}{\epsilon} \log \frac{n \cdot \kappa}{\delta})$  iterations performs  $m$  matrix vector products with  $\mathbf{A}$  and outputs a probability density function  $\tilde{s}_{\mathbf{A}}$  such that, with probability at least  $1 - \delta$ , for some fixed constant  $c > 0$ ,*

$$W_1(s_{\mathbf{A}}, \tilde{s}_{\mathbf{A}}) \leq \tilde{O} \left( \epsilon \cdot \sigma_{l+1}(\mathbf{A}) + \frac{\sigma_{l+1}(\mathbf{A})}{\sqrt{n}} + \frac{l}{n} \sigma_{l+1}(\mathbf{A}) \right) + \frac{\|\mathbf{A}\|_2}{n^c}.$$

### 3 Future Work

There are a number of directions inspired by our work which can be pursued in the future.

**Lanczos based Matrix Function Approximation.** Variants of SLQ and Lanczos have been used to obtain algorithms for estimating general functions of the trace of  $\mathbf{A}$ ,  $\text{tr}(f(\mathbf{A}))$  [UCS17, CTU22, CH23]. The Lanczos method itself can approximate different matrix functions like rational functions very accurately [ACG<sup>+</sup>24]. Our deflation based analysis, particularly that of the variance reduced SLQ, could be used to give improved spectrum adaptive bounds for all these methods.

**Numerical stability.** The Lanczos algorithm is known to suffer from numerical stability issues when implemented in finite precision arithmetic [Che24]. A more careful analysis of how the algorithms perform under finite precision arithmetic will be interesting. However, we note that our experiments (Section 6 of [BJM<sup>+</sup>24]) show that our algorithms work pretty well in practice.

### References

- [ACG<sup>+</sup>24] Noah Amsel, Tyler Chen, Anne Greenbaum, Cameron Musco, and Chris Musco. Nearly optimal approximation of matrix functions by the lanczos method. In

- [BEPW98] Kevin Burrage, Jocelyne Erhel, Bert Pohl, and Alan Williams. A deflation technique for linear systems of equations. *SIAM Journal on Scientific Computing*, 1998.
- [BGVKS22] Jess Banks, Jorge Garza-Vargas, Archit Kulkarni, and Nikhil Srivastava. Pseudospectral shattering, the sign function, and diagonalization in nearly matrix multiplication time. *Foundations of Computational Mathematics*, 2022.
- [BJM<sup>+</sup>24] Rajarshi Bhattacharjee, Rajesh Jayaram, Cameron Musco, Christopher Musco, and Archan Ray. Improved spectral density estimation via explicit and implicit deflation. <https://arxiv.org/abs/2410.21690>, 2024.
- [BKM22] Vladimir Braverman, Aditya Krishnan, and Christopher Musco. Sublinear time spectral density estimation. In *Proceedings of the 54th Annual ACM Symposium on Theory of Computing (STOC)*, 2022.
- [BN23] Ainesh Bakshi and Shyam Narayanan. Krylov methods are (nearly) optimal for low-rank approximation. In *Proceedings of the 64th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2023.
- [BRP92] C Benoit, E Royer, and G Poussigue. The spectral moments method. *Journal of Physics: Condensed Matter*, 1992.
- [CH23] Tyler Chen and Eric Hallman. Krylov-aware stochastic trace estimation. *SIAM Journal on Matrix Analysis and Applications*, 44(3):1218–1244, 2023.
- [Che23] Tyler Chen. A spectrum adaptive kernel polynomial method. *The Journal of Chemical Physics*, 2023.
- [Che24] Tyler Chen. The lanczos algorithm for matrix functions: a handbook for scientists, 2024.
- [CPB10] L Covaci, FM Peeters, and M Berciu. Efficient numerical approach to inhomogeneous superconductivity: The Chebyshev-Bogoliubov-de Gennes method. *Physical review letters*, 2010.
- [CS97] Andrew Chapman and Yousef Saad. Deflated and augmented Krylov subspace techniques. *Numerical linear algebra with applications*, 1997.
- [CTU21] Tyler Chen, Thomas Trogdon, and Shashanka Ubaru. Analysis of stochastic Lanczos quadrature for spectrum approximation. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, 2021.
- [CTU22] Tyler Chen, Thomas Trogdon, and Shashanka Ubaru. Randomized matrix-free quadrature for spectrum and spectral sum approximation. [arXiv:2204.01941](https://arxiv.org/abs/2204.01941), 2022.
- [DBB19] Kun Dong, Austin R Benson, and David Bindel. Network density of states. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2019.
- [DNPS16] Edoardo Di Napoli, Eric Polizzi, and Yousef Saad. Efficient estimation of eigenvalue counts in an interval. *Numerical Linear Algebra with Applications*, 2016.

- [EG17] Nicole Eikmeier and David F Gleich. Revisiting power-law distributions in spectra of real world networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2017.
- [FDBV01] Illés J Farkas, Imre Derényi, Albert-László Barabási, and Tamas Vicsek. Spectra of “real-world” graphs: Beyond the semicircle law. *Physical Review E*, 2001.
- [FTU23] Zachary Frangella, Joel A Tropp, and Madeleine Udell. Randomized Nyström preconditioning. *SIAM Journal on Matrix Analysis and Applications*, 2023.
- [FV01] Jason Frank and Cornelis Vuik. On the construction of deflation-based preconditioners. *SIAM Journal on Scientific Computing*, 2001.
- [Gir87] Didier Girard. Un algorithme simple et rapide pour la validation croisée généralisée sur des problèmes de grande taille. Technical report, 1987.
- [GKX19] Behrooz Ghorbani, Shankar Krishnan, and Ying Xiao. An investigation into neural net optimization via Hessian eigenvalue density. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 2019.
- [GOSS16] Alon Gonen, Francesco Orabona, and Shai Shalev-Shwartz. Solving ridge regression using sketched preconditioned SVRG. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, 2016.
- [GSO17] Arjun Singh Gambhir, Andreas Stathopoulos, and Kostas Orginos. Deflation as a method of variance reduction for estimating the trace of a matrix inverse. *SIAM Journal on Scientific Computing*, 2017.
- [HHK72] R Haydock, V Heine, and M J Kelly. Electronic structure based on the local atomic environment for tight-binding bands. *Journal of Physics C: Solid State Physics*, 1972.
- [HMT11] Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 2011.
- [Hut90] Michael F. Hutchinson. A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines. *Communications in Statistics-Simulation and Computation*, 1990.
- [LG82] P Lambin and J-P. Gaspard. Continued-fraction technique for tight-binding systems. A generalized-moments method. *Physical Review B*, 1982.
- [Lin17] Lin Lin. Randomized estimation of spectral densities of large matrices made accurate. *Numerische Mathematik*, 2017.
- [LSY98] Richard B Lehoucq, Danny C Sorensen, and Chao Yang. *ARPACK users’ guide: solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods*. SIAM, 1998.
- [LSY16] Lin Lin, Yousef Saad, and Chao Yang. Approximating spectral densities of large matrices. *SIAM Review*, 2016.

- [LXES19] Ruipeng Li, Yuanzhe Xi, Lucas Erlandson, and Yousef Saad. The eigenvalues slicing library (EVSL): Algorithms, implementation, and software. *SIAM Journal on Scientific Computing*, 2019.
- [MM15] Cameron Musco and Christopher Musco. Randomized block Krylov methods for stronger and faster approximate singular value decomposition. In *Advances in Neural Information Processing Systems 28 (NIPS)*, 2015.
- [MM19] Michael Mahoney and Charles Martin. Traditional and heavy tailed self regularization in neural network models. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 2019.
- [MMM24] Raphael Meyer, Cameron Musco, and Christopher Musco. On the unreasonable effectiveness of single vector Krylov methods for low-rank approximation. In *Proceedings of the 35th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2024.
- [MMMW21] Raphael A Meyer, Cameron Musco, Christopher Musco, and David P Woodruff. Hutch++: Optimal stochastic trace estimation. In *Symposium on Simplicity in Algorithms (SOSA)*, 2021.
- [MNS<sup>+</sup>18] Cameron Musco, Praneeth Netrapalli, Aaron Sidford, Shashanka Ubaru, and David P. Woodruff. Spectrum approximation beyond fast matrix multiplication: Algorithms and hardness. In *Proceedings of the 9th Conference on Innovations in Theoretical Computer Science (ITCS)*, 2018.
- [Par98] Beresford N Parlett. *The symmetric eigenvalue problem*. SIAM, 1998.
- [PSG18] Jeffrey Pennington, Samuel Schoenholz, and Surya Ganguli. The emergence of spectral universality in deep networks. In *Proceedings of the 21st International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2018.
- [RL18] Aditya Ramesh and Yann LeCun. Backpropagation for implicit spectral densities. [arXiv:1806.00499](https://arxiv.org/abs/1806.00499), 2018.
- [Saa03] Yousef Saad. *Iterative methods for sparse linear systems*. SIAM, 2003.
- [Ski89] John Skilling. The eigenvalues of mega-dimensional matrices. *Maximum Entropy and Bayesian Methods*, 1989.
- [SR94] RN Silver and H Röder. Densities of states of mega-dimensional Hamiltonian matrices. *International Journal of Modern Physics C*, 1994.
- [SRS20] Jürgen Schnack, Johannes Richter, and Robin Steinigeweg. Accuracy of the finite-temperature Lanczos method compared to simple typicality-based estimates. *Physical Review Research*, 2020.
- [STBB17] Björn Sbierski, Maximilian Trescher, Emil J Bergholtz, and Piet W Brouwer. Disordered double Weyl node: Comparison of transport and density of states calculations. *Physical Review B*, 2017.
- [SW23] William Swartworth and David P Woodruff. Optimal eigenvalue approximation via sketching. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing (STOC)*, 2023.

- [Tro18] Joel A. Tropp. Analysis of randomized block Krylov methods. Technical report, California Institute of Technology , Pasadena, CA, 2018.
- [UCS17] Shashanka Ubaru, Jie Chen, and Yousef Saad. Fast estimation of  $\text{tr}(f(a))$  via stochastic Lanczos quadrature. *SIAM Journal on Matrix Analysis and Applications*, 2017.
- [Wan94] Lin-Wang Wang. Calculating the density of states and optical-absorption spectra of large quantum systems by the plane-wave moments method. *Physical Review B*, 1994.
- [WWAF06] Alexander Weiße, Gerhard Wellein, Andreas Alvermann, and Holger Fehske. The kernel polynomial method. *Reviews of Modern Physics*, 2006.
- [GM09] Gene H Golub and Gérard Meurant. *Matrices, moments and quadrature with applications*. Princeton University Press, 2009.

# Birkhoff Averages, Invariant Sets, and Adaptive Filtering

David S. Bindel, Maximillian E. Ruth

## Abstract

In the design of magnetic confinement fusion devices (and many other applications), one is interested in classifying the trajectories of symplectic maps. That is, we consider discrete dynamical systems

$$\mathbf{x}_{t+1} = \mathbf{F}(\mathbf{x}_t)$$

where the map  $\mathbf{F} : X \rightarrow X$  is symplectic. We are interested in classifying such trajectories as quasiperiodic orbits (invariant circles, islands) or as chaotic, and finding simple parameterizations of any quasiperiodic structures. In this talk, we describe a simple approach to these tasks by building a linear time-invariant system representation of the dynamics from a given starting point with a palindromic characteristic polynomial. This allows us to find a Fourier parameterization of invariant circles and islands from a single trajectory, as well as classifying trajectories as regular or chaotic. We connect our approach to ideas from extrapolation methods, adaptive filter design, and Birkhoff averages, and show examples of Birkhoff RRE on the standard map and magnetic field line dynamics.

## References

- [1] Maximillian Ruth and David Bindel. Finding Birkhoff averages via adaptive filtering. *Chaos*, 34(12):123109, December 2024.

# Parallelization of all-at-once preconditioned solvers for time-dependent PDEs

*Matthias Bolten, Ryo Yoda*

## Abstract

Modern high performance computers provide tremendous compute power by utilizing large amounts of cores. As a consequence, traditional spatial parallelization schemes lead to a saturation of the speedup more often. This motivated the use of parallelization in the time direction, as well. Various approaches exist and often two- and multilevel approaches like parareal or space-time multigrid are chosen that introduce a coarse level—or multiple coarse levels—to propagate information faster in time, usually in a serial manner. An alternative that is very natural from a numerical linear algebra viewpoint is to consider all-at-once systems. Consider a linear time dependent PDE:

$$\begin{aligned} \frac{\partial u(x, t)}{\partial t} &= \mathcal{L}u(x, t) + f(x, t), & (x, t) \in \Omega \times (0, T], \\ u &= g, & (x, t) \in \partial\Omega \times (0, T], \\ u(x, 0) &= u_0(x), & x \in \Omega. \end{aligned}$$

Discretization using finite elements and denoting the mass matrix by  $M$  and the stiffness matrix by  $K$  in each time step with step width  $\tau = \frac{T}{n}$  yields

$$M \left( \frac{\mathbf{u}^{(k+1)} - \mathbf{u}^{(k)}}{\tau} \right) = K\mathbf{u}^{(k+1)} + \mathbf{f}^{(k+1)},$$

writing the resulting  $n$  linear systems into one finally gives

$$\begin{bmatrix} M - \tau K & & & \\ -M & M - \tau K & & \\ & \ddots & \ddots & \\ & & -M & M - \tau K \end{bmatrix} \begin{bmatrix} \mathbf{u}^{(1)} \\ \mathbf{u}^{(2)} \\ \vdots \\ \mathbf{u}^{(n)} \end{bmatrix} = \begin{bmatrix} \mathbf{f}^{(1)} + M\mathbf{u}_0 \\ \mathbf{f}^{(2)} \\ \vdots \\ \mathbf{f}^{(n)} \end{bmatrix}.$$

This large system can be solved iteratively using, e.g., GMRES. Further, when  $M$  and  $K$  are symmetric or can be made symmetric simultaneously and when they are not changing like in the example provided, the matrix can be symmetrized by applying a reordering technique allowing to use methods like MINRES. If  $M$  and  $K$  do not change over time preconditioners for block Toeplitz or block Hankel matrices can be used in both cases. The GMRES case has been studied, e.g., in [3], the MINRES case was considered, e.g., in [2, 4, 5, 6]. The preconditioners studied are either block circulant or block  $\epsilon$ -circulant and thus multiplication and inversion can be carried out in almost optimal, i.e.,  $\mathcal{O}(n \log n)$ , complexity by using the FFT. Additionally, the blocks have to be solved efficiently which is usually carried out using the method used for the sequential time-stepping. While the study of the preconditioners is extensive, the actual parallel implementation is studied very little. One option to implement these kinds of methods is the usage of a parallel FFT as studied in [1]. Yet, an efficient parallelization of the FFT is relatively difficult given that it requires a lot of communication in comparison to very few arithmetic operations. This is one reason why multi-dimensional FFTs usually transpose the data such that the individual 1D-FFTs can be carried out sequentially. For the preconditioners considered in this case, in general only 1D-FFTs are needed.

Given the results obtained using efficient multi-dimensional FFTs on parallel computers as an alternative to a direct parallelization of the FFT we propose to transpose the data such that sequential 1D-FFTs can be used and to transpose it back before solving the individual blocks. The method resulting from a parallel implementation of the method proposed in [3] in this way provides an excellent scaling behavior, yielding an additional speedup after saturation of pure time-stepping with parallel solves of the spatial problem alone.

The applications of the preconditioners require the solution of usually complex block system that have the dimension of the spatial problem. Our implementation currently uses smoothed aggregation multigrid from Trilinos to solve these systems. We have used this approach on machines based on CPUs as well as on clusters with GPUs. In all cases we can achieve good scaling results, providing efficient parallelization in time by using preconditioning.

We will provide an overview over the different preconditioners that can be implemented in the proposed way, present the parallelization approach in detail, discuss the solution of the blocks and demonstrate the achieved performance.

## References

- [1] G. CAKLOVIC, R. SPECK, AND M. FRANK, *A parallel-in-time collocation method using diagonalization: theory and implementation for linear problems*, 2023, arXiv:2103.12571 [math.NA].
- [2] S. HON, *Optimal block circulant preconditioners for block Toeplitz systems with application to evolutionary PDEs*, J. Comput. Appl. Math., 407 (2022), p. 15. Id/No 113965.
- [3] X.-L. LIN AND M. NG, *An all-At-once preconditioner for evolutionary partial differential equations*, SIAM J. Sci. Comput., 43 (2021), pp. a2766–a2784.
- [4] E. McDONALD, S. HON, J. PESTANA, AND A. WATHEN, *Preconditioning for nonsymmetry and time-dependence*, in Domain decomposition methods in science and engineering XXIII. Proceedings of the 23rd international conference, Jeju Island, South Korea, July 6–10, 2015, Cham: Springer, 2017, pp. 81–91.
- [5] E. McDONALD, J. PESTANA, AND A. WATHEN, *Preconditioning and iterative solution of all-at-once systems for evolutionary partial differential equations*, SIAM J. Sci. Comput., 40 (2018), pp. a1012–a1033.
- [6] J. PESTANA AND A. J. WATHEN, *A preconditioned MINRES method for nonsymmetric Toeplitz matrices*, SIAM J. Matrix Anal. Appl., 36 (2015), pp. 273–288.

# Parallel Incomplete Factorization Preconditioners

*Erik G Boman<sup>1</sup>, Marc A. Tunnel<sup>2</sup>*

## Abstract

Incomplete factorizations are popular preconditioners and are well known to be effective for a wide range of problems. Additionally, these preconditioners can be used as a “black box” and do not rely on any *a priori* knowledge of the problem. However, traditional algorithms for computing these incomplete factorizations are based on Gaussian elimination and do not parallelize well. Recently, a more parallel incomplete factorization algorithm was proposed by Chow and Patel [4], where the factors are computed iteratively. Here we propose a new iterative approach that is based on alternating triangular solves of  $L$  and  $U$ . We develop two versions: ATS-ILU for a static sparsity pattern, and ATS-ILUT for a dynamic pattern (using thresholding). We show that this new method is similar to the fine-grained iterative ILU method by Chow but has the added advantage that it allows greater reuse of memory and is fully deterministic in parallel, meaning the results do not depend on scheduling. We evaluate the new method on several test matrices from the SuiteSparse collection and show that it is competitive with current ILU methods. When short setup time is important, it is typically better than other methods.

## 1 Introduction

Preconditioning is well known to be essential for improving the speed of convergence of Krylov methods such as Conjugate Gradient (CG) and Generalized Minimal Residual (GMRES) [8]. Incomplete Lower-Upper (ILU) factorizations are a popular class of preconditioners as they can be used as a “black box” on a wide range of problems. There are two main types of ILU factorizations, level-based ILU(k) [3, 6, 7] and threshold-based ILUT [9]. However, these methods are inherently sequential and do not parallelize well.

There has been interest in the parallelization of these more classical interpretations of ILU, largely through graph partitioning schemes. These graph partition-based methods, such as [5], offer a promising approach to parallelizing classical ILU methods. By decomposing the graph corresponding to the matrix and determining variables that can be eliminated in parallel, these methods aim to distribute the computational load more evenly across processors. While these strategies have shown effectiveness for certain types of problems [3], their implementation can be highly complex. Additionally, their performance can be problem-dependent, requiring consideration of the underlying graph structure when choosing a parallelization strategy.

More recently, there have been strides into methods of computing ILU factors iteratively, potentially giving up some of the robustness of the classical methods for better parallel properties [4]. Iterative ILU methods, such as those introduced by Chow [4], offer significant advantages in terms of scalability on modern parallel architectures. For the remainder of this paper, we refer to the method introduced by Chow as ParILU and its thresholded counterpart as ParILUT [1, 4]. These methods approximate the ILU factors through a series of iterative updates, which can be more easily distributed across multiple processors or offloaded to accelerators.

---

<sup>1</sup>Sandia National Labs, [egboman@sandia.gov](mailto:egboman@sandia.gov)

<sup>2</sup>Purdue University, [mtunnell@purdue.edu](mailto:mtunnell@purdue.edu)

By breaking down each iterative update into smaller approximate subproblems and solving them independently, different parts of the factorization can be computed in parallel without the need for complex graph-partitioning algorithms. This approach allows for the use of iterative ILU methods on a wide range of problems, including those with complex or irregular graph structures that may preclude high levels of parallelism in the graph-partitioned classical ILU methods.

Furthermore, iterative ILU methods are adaptable to various hardware accelerators such as graphics processing units (GPUs) [2], which are increasingly important for high-performance computing. By leveraging the parallel processing capability of these accelerators, iterative ILU methods can significantly reduce the real-world time required to compute the ILU factors for large-scale problems, thereby speeding up the overall solution process.

In this paper, we propose a new class of iteratively-computed ILU preconditioners, which we call Alternating triangular Solves ILU (ATS-ILU). This method builds upon the strengths of existing iterative ILU approaches while leveraging improved memory reuse and determinism in parallel. We provide an analysis of the method and evaluate its performance compared to the state of the art on a variety of test matrices. We show that our method is competitive with current ILU methods and has the potential to be a powerful tool for solving large-scale problems on modern parallel architectures.

## 2 Alternating Triangular Solves Method

In this section, we introduce our new method for computing ILU factors, ATS-ILU. This method is based on the idea of alternating iterative updates to the  $L$  and  $U$  factors of the matrix  $A$ . The basic idea is the same as before, where we iteratively update the factors  $L$  and  $U$  until convergence, but where the updates are performed in an alternating manner. This general process is a common method for solving bilinear systems and is outlined in 1.

---

**Algorithm 1** Alternating ILU

---

```

 $U^{(0)} \leftarrow \text{triu}(A)$ 
 $k \leftarrow 0$ 
while not converged do
    Solve  $L^{(k)}U^{(k)} \approx A$  for  $L^{(k)}$ 
    Solve  $L^{(k)}U^{(k+1)} \approx A$  for  $U^{(k+1)}$ 
    Check convergence
     $k \leftarrow k + 1$ 
end while

```

---

One way to perform this procedure would be to perform a triangular solve with the entirety of  $U^{(k)}$  and let  $A$  be the right-hand side vector to solve for  $L^{(k+1)}$ , and similar to solve for  $U^{(k+1)}$ . This entire process can largely be performed in parallel as each row of  $L$  and column of  $U$  can be solved independently. Despite the potential for high levels of parallelism, it is still extremely computationally expensive and likely suffers from significant levels of fill-in during intermediate steps. The computational cost could be reduced by using an approximation.

Additionally, the algorithm as stated above does not guarantee that  $L$  and  $U$  remain lower and upper triangular, respectively. One method to address this issue would be to solve for  $L$  only in the lower triangular part of  $A$  and for  $U$  only in the upper triangular part of  $A$ . This would ensure that the factors remain lower and upper triangular, respectively, but would still leave the problem

of significant levels of fill-in. Instead, we suggest a more practical approach where we impose a sparsity pattern on  $L$  and  $U$ , namely  $\mathcal{L}$  and  $\mathcal{U}$ , respectively. This sparsity pattern can be chosen to be the same as the sparsity pattern of  $A$ , which is the choice we make in this paper.

In order to get around the issue of fill-in, we propose a method where we approximately solve for  $L$  and  $U$  along their given sparsity patterns, which we discuss next.

## 2.1 Alternating Triangular Solves ILU Algorithm

The ATS-ILU algorithm is based on the idea of approximately solving for  $L$  and  $U$  in an alternating fashion along only their given sparsity patterns. Again, the rows of  $L$  and the columns of  $U$  can be solved independently, allowing for a high level of parallelism. The algorithm is shown in 2. We present the algorithm for a general pattern  $\mathcal{S}$  but in practice, this will correspond to the pattern of  $A^k$  for some small power  $k$ .

---

### Algorithm 2 ATS-ILU

---

```

1: Input: Sparse matrix  $A$ , sparsity pattern  $\mathcal{S}$ , starting factors  $L$  and  $U$ 
2: while not converged do
3:   for  $i \in \{1 2 \dots n\}$  do
4:      $\text{idx} \leftarrow \{j \in \mathbb{N} \mid (i, j) \in \mathcal{S}, j \leq i\}$ 
5:      $\ell_{i,\text{idx}} \leftarrow a_{i,\text{idx}} (U_{\text{idx},\text{idx}})^{-1}$ 
6:   end for
7:   for  $j \in \{1 2 \dots n\}$  do
8:      $\text{idx} \leftarrow \{i \in \mathbb{N} \mid (i, j) \in \mathcal{S}, i \geq j\}$ 
9:      $u_{\text{idx},j} \leftarrow (L_{\text{idx},\text{idx}})^{-1} a_{\text{idx},j}$ 
10:  end for
11: end while

```

---

In this algorithm, we solve for each row of  $L$  and each column of  $U$  independently. Recall that the notation  $\mathbf{a}_{i,\text{idx}}$  refers to the  $i^{\text{th}}$  row of  $A$  restricted to the indices in  $\text{idx}$ , and similarly for  $U_{\text{idx},\text{idx}}$  and  $L_{\text{idx},\text{idx}}$ . These submatrices can be viewed as the (dense) non-contiguous submatrices of  $L$  and  $U$  that correspond to the sparsity pattern  $\mathcal{S}$  along the given row or column.

Our method can be extended to do thresholding to maintain a certain fill level. We call this extension ATS-ILUT, and defer the details to the full paper.

## 3 Results

We implemented the ATS-ILUT algorithm in C++ with Kokkos for parallel performance portability. We show some preliminary results in Table 3.

## 4 Conclusions

We have developed a new parallel iterative ILU algorithm ATS-ILU and a thresholded version ATS-ILUT. Experiments show it performs similarly to the ParILU(T) method, but it often provides a better quality preconditioner after just one or two steps (updates) of the setup. This is an advantage

Table 1: Comparison of ATS-ILU Variants with PAR-ILUT across Different Matrices, Fill Levels, and Iterations. The best at each iteration is bolded.

Matrices:		abnormal_sandia					af_shell3					G3_circuit					parabolic_fem					
Method	Fill	Iterations																				
		1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	
ATS-ILUT	1.0	54	<b>44</b>	<b>42</b>	<b>42</b>	<b>42</b>	905	723	594	<b>605</b>	<b>568</b>	1169	<b>1140</b>	<b>1148</b>	<b>1133</b>	<b>1131</b>	1183	<b>1089</b>	<b>1090</b>	<b>1083</b>	<b>1071</b>	
	(a)	2.0	51	33	<b>25</b>	<b>24</b>	23	872	564	395	334	299	860	639	467	426	396	765	561	464	492	444
		3.0	51	31	22	<b>18</b>	<b>17</b>	872	559	378	308	258	860	638	448	373	318	765	551	422	434	381
ATS-ILUT	1.0	50	45	45	46	45	797	657	638	625	631	1153	1186	1189	1187	1183	1261	1192	1224	1220	1201	
	(b)	2.0	<b>42</b>	<b>29</b>	27	27	27	651	<b>402</b>	319	289	274	690	<b>520</b>	408	424	414	<b>729</b>	719	505	547	446
		3.0	<b>42</b>	<b>24</b>	20	20	20	651	<b>397</b>	290	<b>226</b>	<b>204</b>	690	<b>520</b>	<b>357</b>	326	305	<b>729</b>	715	681	525	378
ParILUT	1.0	54	45	45	45	45	822	<b>597</b>	<b>581</b>	616	592	1188	1170	1180	1215	1217	1232	1168	1190	1201	1197	
		2.0	49	32	26	25	25	752	415	<b>311</b>	<b>279</b>	<b>268</b>	758	531	<b>390</b>	<b>365</b>	<b>360</b>	864	<b>482</b>	<b>379</b>	<b>353</b>	<b>354</b>
		3.0	49	30	21	<b>18</b>	<b>17</b>	752	415	293	234	204	758	531	379	<b>295</b>	<b>269</b>	864	<b>479</b>	<b>320</b>	<b>219</b>	<b>191</b>

if setup time is important, e.g., when solving a sequence of linear systems. Also, it is naturally deterministic (though an asynchronous version is also possible).

## Acknowledgments

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy’s National Nuclear Security Administration under contract DE-NA-0003525.

## References

- [1] Hartwig Anzt, Edmond Chow, and Jack Dongarra. Parilut—a new parallel threshold ILU factorization. *SIAM Journal on Scientific Computing*, 40(4):C503–C519, 2018.
- [2] Hartwig Anzt, Tobias Ribizel, Goran Flegar, Edmond Chow, and Jack Dongarra. Parilut - a parallel threshold ILU for GPUs. In *2019 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. IEEE, May 2019.
- [3] Michele Benzi. Preconditioning techniques for large linear systems: A survey. *Journal of Computational Physics*, 182(2):418–477, November 2002.
- [4] Edmond Chow and Aftab Patel. ”A fine-grained parallel ILU factorization”. *SIAM Journal on Scientific Computing*, 37(2):C169–C197, 2015.
- [5] David Hysom and Alex Pothen. Efficient parallel computation of ILU(k) preconditioners. In *Proceedings of the 1999 ACM/IEEE conference on Supercomputing*, SC ’99. ACM, January 1999.
- [6] Na Li, Yousef Saad, and Edmond Chow. Crout versions of ilu for general sparse matrices. *SIAM Journal on Scientific Computing*, 25(2):716–728, January 2003.
- [7] Y Saad. *Iterative methods for sparse linear systems*. SIAM, Philadelphia, MS, 2 edition, 2003.
- [8] Youcef Saad and Martin H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7(3):856–869, July 1986.
- [9] Yousef Saad. ILUT: A dual threshold incomplete lu factorization. *Numerical Linear Algebra with Applications*, 1(4):387–402, July 1994.

# Operator Learning without the Adjoint

*Nicolas Boullé, Diana Halikias, Samuel Otto, Alex Townsend*

## Abstract

There is a mystery at the heart of operator learning: how can one recover a non-self-adjoint operator from data without probing its adjoint? Current practical approaches suggest that one can accurately recover an operator while only using data generated by the forward action of the operator without access to the adjoint [5]. However, naively, it seems essential to sample the action of the adjoint for learning solution operator of time-dependent partial differential equations (PDEs) [3]. This motivates a fundamental question in numerical linear algebra: can one approximate a non-symmetric low-rank matrix without sketching its adjoint?

In this talk, we will explore the limits of adjoint-free low-rank matrix recovery and propose an approach that could help analyze the behavior of structured matrix recovery algorithms. Then, we will show that one can approximate a family of non-self-adjoint infinite-dimensional compact operators via projection onto a Fourier basis without querying the adjoint. We will apply the result to recover Green's functions of elliptic partial differential operators and derive an adjoint-free sample complexity bound. While existing infinite-dimensional numerical linear algebra theory justifies low sample complexity in operator learning [2, 4], ours is the first adjoint-free analysis that attempts to close the gap between theory and practice [1].

**Limits of adjoint-free low-rank matrix recovery.** We start in the fundamental setting of recovering a low-rank matrix by querying the map  $x \mapsto Fx$  but without access to  $x \mapsto F^*x$ . We show that querying  $x \mapsto F^*x$  is essential for recovering  $F$  and prove rigorous guarantees on the quality of the reconstruction in terms of how close  $F$  is to a symmetric matrix. Thus, we conclude that without prior knowledge of the properties of the adjoint, one must have access to its action.

We assume that  $F$  is  $\delta$ -near-symmetric (*i.e.*, its left and right singular subspaces are  $\delta$ -close), but we only have access to partial information regarding the symmetry of  $F$ , namely that  $F$  is  $\epsilon$ -near-symmetric for some  $\epsilon \geq \delta$ , and sketching constraint  $FX$ . To quantify the resulting uncertainty about  $F$ , we define the set of possible matrices one could recover given this prior knowledge as

$$\Omega_{F,X}^\epsilon = \{A \in M_n(\mathbb{C}): \text{rank}(A) = k, AX = FX, \exists Q \in O(k), \|U_A^*V_A - Q\|_2 \leq \epsilon\}, \quad (1)$$

where  $A = U_A S_A V_A^*$  is the singular value decomposition of  $A$ ,  $O(k)$  is the group of  $k \times k$  orthogonal matrices, and  $\|\cdot\|_2$  denotes the spectral norm. Hence, given some tolerance  $\epsilon$ ,  $\Omega_{F,X}^\epsilon$  is the set of  $\epsilon$ -near-symmetric matrices that can be returned by any low-rank recovery algorithm when approximating  $F$ , such as the randomized SVD [6, 7] or the Nyström method [8].

The size of  $\Omega_{F,X}^\epsilon$  is measured by its diameter in the spectral norm and determines the maximum accuracy of any reasonable reconstruction. If the diameter is large, one cannot estimate  $F$  accurately, as one cannot distinguish between any candidate matrix in  $\Omega_{F,X}^\epsilon$ . This is because any matrix in  $\Omega_{F,X}^\epsilon$  satisfies the sketching constraint and is near-symmetric. On the other hand, a small diameter guarantees the fidelity of the reconstruction. We provide sharp upper and lower bounds on the size of  $\Omega_{F,X}^\epsilon$ , *i.e.*, determine how far apart any two matrices in  $\Omega_{F,X}^\epsilon$  can be from each other, with respect to  $\epsilon$ , which measures our prior knowledge of  $F$ 's symmetry. The upper and lower bounds on the diameter of  $\Omega_{F,X}^\epsilon$  reveal that the uncertainty about  $F$  given queries of its action is directly related to the uncertainty about the symmetry of its left and right singular subspaces. For example,

our ability to recover a symmetric rank- $k$  matrix using  $k \leq s < n$  queries is fundamentally limited by our prior knowledge about the proximity of  $\text{Range}(F)$  and  $\text{Range}(F^*)$  because there are many asymmetric matrices with the same rank that satisfy the same sketching constraints. This result is a fundamental limitation of adjoint-free low-rank matrix recovery in numerical linear algebra and has implications for operator learning.

**An adjoint-free operator learning approach.** To provide an operator learning approach that does not need access to the adjoint, we exploit regularity results from PDE theory to estimate the range of the adjoint of the solution operator. This allows us to prove the first guarantees on the accuracy of adjoint-free approximations. Our key insight is to leverage the favorable properties of a prior self-adjoint operator, such as the Laplace–Beltrami operator, to use as an operator preconditioner in the approximation problem. In particular, we query the action of the solution operator on the eigenfunctions of the prior self-adjoint operator, yielding an approximation with an error that decays at a rate determined by the eigenvalues of the prior. This is remarkable because common operator learning techniques always seem to plateau; yet, we construct a simple algorithm that provably converges.

**The effect of non-normality on sample complexity.** We derive a sample complexity bound for our algorithm when applied to second-order uniformly-elliptic PDEs that are perturbed away from self-adjointness by lower-order terms. We show that for small perturbations, our bound on the approximation error grows linearly with the size of the perturbation, and we conjecture that this linear growth continues for large perturbations as well. This aspect of the error growth is also present in common operator learning techniques, as our numerical experiments illustrate. With respect to our operator learning algorithm, this means that the number of samples required to achieve a fixed error tolerance grows algebraically with the perturbation size.

## References

- [1] N. BOULLÉ, D. HALIKIAS, S. E. OTTO, AND A. TOWNSEND, *Operator learning without the adjoint*, arXiv preprint arXiv:2401.17739, (2024).
- [2] N. BOULLÉ, D. HALIKIAS, AND A. TOWNSEND, *Elliptic PDE learning is provably data-efficient*, Proc. Natl. Acad. Sci. USA, 120 (2023), p. e2303904120.
- [3] N. BOULLÉ, S. KIM, T. SHI, AND A. TOWNSEND, *Learning Green’s functions associated with time-dependent partial differential equations*, J. Mach. Learn. Res., 23 (2022), pp. 1–34.
- [4] N. BOULLÉ AND A. TOWNSEND, *Learning elliptic partial differential equations with randomized linear algebra*, Found. Comput. Math., 23 (2023), pp. 709–739.
- [5] N. BOULLÉ AND A. TOWNSEND, *A mathematical guide to operator learning*, in Handbook of Numerical Analysis, vol. 25, Elsevier, 2024, ch. 3, pp. 83–125.
- [6] N. HALKO, P.-G. MARTINSSON, AND J. A. TROPP, *Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions*, SIAM Rev., 53 (2011), pp. 217–288.
- [7] P.-G. MARTINSSON AND J. A. TROPP, *Randomized numerical linear algebra: Foundations and algorithms*, Acta Numer., 29 (2020), pp. 403–572.
- [8] E. J. NYSTRÖM, *Über die praktische Auflösung von Integralgleichungen mit Anwendungen auf Randwertaufgaben*, Acta Math., 54 (1930), pp. 185–204.

# Streaming the Bidiagonal Factorization

*Johannes J. Brust and Michael A. Saunders*

## Abstract

Frequently in online or data-driven applications, new information becomes available in the form of a *stream* (Syamantak et al. [KS24]). Because processing the data for analysis or inference often involves matrix factorizations like the SVD or an eigendecomposition, we develop new updating methods. As repeatedly refactoring a large matrix is expensive, we propose low-rank updates to a previous factorization. Thus we consider the model

$$\bar{A} = A + CW^T,$$

where the previous data  $A$  is  $m \times n$ , and the updates  $C$  and  $W$  are  $m \times t$  and  $n \times t$ . Simply computing  $\bar{A}$  costs  $mnt$  multiplications, which we therefore regard as an optimal complexity. For some well known factorizations, efficient updating methods are known. For instance, the methods of Gill et al. [GGMS74] update the Cholesky or  $LDL^T$  factorization in  $\mathcal{O}(mnt)$  flops, while Golub and Van Loan [GV13, Sec 12.5] describe a method for updating the QR factorization. Also, Bunch et al. [BNS78] describe an algorithm for updating the eigendecomposition, while Brand [Bra06] and Moonen et al. [MVDV92] develop methods for the SVD. The complexity of the latter methods also scales as  $\mathcal{O}(mnt)$ , but updating an eigendecomposition or SVD typically involves iterative nonlinear equation solves. In SVD computation, the first step is to reduce the matrix to (upper) bidiagonal form before computing the singular values of the bidiagonal (e.g., implemented in LAPACK’s `bdsqr` and `gebrd` [ABB+99]). Since the bidiagonalization and SVD are closely related, it is not surprising that attempts have been made to replace the SVD with the bidiagonalization for low-rank matrix approximations (Simon and Zha [SZ00]). Even though the SVD guarantees the best low-rank approximation, the bidiagonalization can be computed with a predetermined number of orthogonal transformations, making it computationally attractive.

The most stable method for computing a bidiagonal factorization uses sequences of orthogonal Householder reflectors. Because this requires and overwrites the matrix elements in memory, it is best suited for dense systems. Its complexity scales as  $\mathcal{O}(mn^2)$  flops and  $mn$  memory and it is therefore limited to small or medium problems. A second approach accesses the data only via matrix-vector products within a short two-vector recursion. This Golub-Kahan bidiagonalization (GKB) produces a partial bidiagonalization after  $k$  iterations. For a general  $A$  the GKB complexity is  $\mathcal{O}(kmn)$ . When the data is sparse or otherwise structured, GKB can exploit the structure with potentially much fewer flops (but without further modifications suffers from rapid loss of orthogonality). Our algorithms reuse an existing bidiagonal factorization  $A = QBP^T$  to compute the next factorization

$$\bar{Q}\bar{B}\bar{P}^T = QBP^T + CW^T$$

at reduced cost. To exploit previous information fully, we develop sparsity-exploiting bidiagonalization algorithms. One method is `gk-bidiag`, which we compare to LAPACK’s bidiagonalization functions (Table 1). We also propose methods such as a compact representation of products of Householder matrices combined with the GKB iteration.

Problem	<i>m</i>	<i>n</i>	gk-bidiag		LAPACK	
			error	secs	error	secs
GL7d12	8899	1019	0.96	<b>0.031</b>	0.97	26
ch6-6-b2	2400	450	1.6	<b>0.0051</b>	0.94	1.2
ch7-6-b2	4200	630	1.1	<b>0.012</b>	0.95	3.9
ch7-7-b2	7350	882	1.3	<b>0.023</b>	0.97	16
cis-n4c6-b2	1330	210	2.7	<b>0.0017</b>	0.91	0.3
mk11-b2	6930	990	1.2	<b>0.02</b>	0.97	16
n4c6-b2	1330	210	2.9	<b>0.0017</b>	0.91	0.3
rel6	2340	157	0.7	<b>0.0028</b>	0.71	0.56
relat6	2340	157	0.74	<b>0.0029</b>	0.74	0.71

Table 1: Updating a rank  $r = 50$  truncated bidiagonal factorization  $\bar{Q}_{1:r}\bar{B}_{1:r}\bar{P}_{1:r}^T$  when a rank-one update is added to a previous factorization. LAPACK subroutines and the sparsity-preserving solver gk-bidiag are applied to 9 SuiteSparse matrices [DH11]. The error is  $\|\bar{A} - \bar{Q}_{1:r}\bar{B}_{1:r}\bar{P}_{1:r}^T\|_F/\|A\|_F$ .

## References

- [ABB+99] Edward Anderson, et al. *LAPACK users' guide*. SIAM, 1999
- [BNS78] James R Bunch, Christopher P Nielsen, and Danny C Sorensen. Rank-one modification of the symmetric eigenproblem. *Numerische Mathematik*, 31(1):31–48, 1978.
- [Bra06] Matthew Brand. Fast low-rank modifications of the thin singular value decomposition. *Linear Algebra and its Applications*, 415(1):20–30, 2006.
- [DH11] Timothy A Davis and Yifan Hu. The University of Florida sparse matrix collection. *ACM Trans. Math. Softw. (TOMS)*, 38(1):1–25, 2011.
- [GGMS74] Philip E Gill, Gene H Golub, Walter Murray, and Michael A Saunders. Methods for modifying matrix factorizations. *Mathematics of Computation*, 28(126):505–535, 1974.
- [GV13] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. The Johns Hopkins University Press, Baltimore, fourth edition, 2013.
- [KS24] Syamantak Kumar and Purnamrita Sarkar. Streaming PCA for Markovian data. *Advances in Neural Information Processing Systems (NeurIPS)*, 36, 2024.
- [MVDV92] Marc Moonen, Paul Van Dooren, and Joos Vandewalle. A singular value decomposition updating algorithm for subspace tracking. *SIAM Journal on Matrix Analysis and Applications*, 13(4):1015–1038, 1992.
- [SZ00] Horst D Simon and Hongyuan Zha. Low-rank matrix approximation using the Lanczos bidiagonalization process with applications. *SIAM Journal on Scientific Computing*, 21(6):2257–2274, 2000.

# Streaming low-rank approximation of tree tensor networks

*Alberto Bucci, Gianfranco Verzella*

## Abstract

Low-rank tensor approximation has emerged as a powerful tool in scientific computing, enabling the efficient handling of large-scale linear and multilinear algebra problems that would otherwise be computationally infeasible with classical methods. By exploiting the inherent low-dimensional structure within high-dimensional data, these techniques reduce storage costs and computational complexity, making it possible to approximate solutions to problems in fields as diverse as quantum physics, machine learning, and computational biology.

Recent advances in randomized techniques for low-rank matrix approximations, including methods like randomized SVD [1] and the generalized Nyström method [2, 3], have paved the way for substantial progress in tensor approximation as well. A range of specialized randomized methods have emerged for tensor decompositions. For instance, the randomized higher-order SVD and its sequential truncated version [4] provide efficient tools for approximating tensors in Tucker format. Likewise, randomized adaptations of TT-SVD [5] extend matrix-based techniques to the tensor train format, enabling the approximation of high-dimensional data while mitigating the curse of dimensionality.

The multilinear Nyström method [6], its sequential counterpart [7], and the streaming tensor train approximation [8] further advance this field, allowing for the streaming low-rank approximation of a given tensor  $\mathcal{A}$  in the Tucker or Tensor-Train format respectively.

Both methods build on the generalized Nyström approach, accessing the tensor  $\mathcal{A}$  exclusively via two-sided random sketches of the original data, making them single-pass and facilitating parallel implementation.

Tucker and tensor train decompositions are specific instances of the more general tree tensor network (TTN) decomposition, where the underlying tree structure takes the form of either a star or chain configuration.

By combining the multilinear Nyström method [6] with the streaming tensor train approximation [8], we introduce the tree tensor network Nyström algorithm [9] (TTNN): a novel approach for the streaming low-rank approximation of tensors in the tree tensor network format. We also introduce a sequential variant of the algorithm that operates on increasingly compressed versions of the tensor, while remarkably preserving streamability. We also provide a detailed analysis of the accuracy of both methods.

However, in practical applications, tensors are often provided in a low-rank TTN format, as working with the full tensor would be computationally prohibitive. In these cases, the challenge lies in achieving further compression or rounding of these representations.

We demonstrate that our algorithm can be readily adapted to this specific setting by leveraging structured embeddings.

Our results indicate that TTNN is capable of achieving nearly optimal approximation error when the sizes of the sketches are appropriately selected. A series of numerical experiments further illustrate the performance of TTNN in comparison to existing deterministic and randomized methods.

## References

- [1] Halko, N., Martinsson, P.G., Tropp, J.A. (2011). Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2), pp.217-288.
- [2] Clarkson K.L., Woodruff, D.P. (2009). Numerical linear algebra in the streaming model. *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pp.205-214.
- [3] Nakatsukasa Y. (2020). Fast and stable randomized low-rank matrix approximation. *arXiv preprint*, arXiv:2009.11392.
- [4] Ahmadi-Asl, S., Abukhovich, S., Asante-Mensah, M.G., Cichocki, A., Phan, A.H., Tanaka, T. and Oseledets, I. (2021). Randomized algorithms for computation of Tucker decomposition and higher order SVD (HOSVD). *IEEE Access*, 9, pp.28684-28706.
- [5] Al Daas, H., Ballard, G., Cazeaux, P., Hallman, E., Miedlar, A., Pasha, M., Reid, T.W. and Saibaba, A.K. (2023). Randomized algorithms for rounding in the tensor-train format. *SIAM Journal on Scientific Computing*, 45(1), pp.A74-A95.
- [6] Bucci A., Robol. L. (2024). A multilinear Nyström algorithm for low-rank approximation of tensors in Tucker format. *SIAM Journal on Matrix Analysis and Applications*, 45(4), pp.1929-1953.
- [7] Bucci A., Hashemi. B. (2024). A sequential multilinear Nyström algorithm for streaming low-rank approximation of tensors in Tucker format. *Applied Mathematics Letters*, 159, p.109271.
- [8] Kressner D., Vandereycken B., Voorhaar R. (2023). Streaming tensor train approximation. *SIAM Journal on Scientific Computing*, 45(5), pp.A2610-A2631.
- [9] Bucci A., Verzella G. (2025). Streaming low-rank approximation of tree tensor networks. *In preparation*.

# Krylov Subspace Recycling With Randomized Sketching For Matrix Functions

*Liam Burke, Stefan Güttel*

## Abstract

I will discuss the importance of randomization in the development of *Krylov subspace recycling* algorithms for the efficient evaluation of a sequence of matrix function applications on a set of vectors [5]. Recycling methods are a special class of augmented Krylov subspace methods where the augmentation subspace for each problem is constructed or *recycled* from the Krylov subspace used to solve a previous problem in the sequence [4]. If selected appropriately, the presence of the recycled subspace can aid in accelerating the convergence of the iterative solver, thereby reducing the overall computational cost and runtime required to solve the full sequence of problems.

I will present the work in [1], where the *recycled Full Orthogonalization Method* (rFOM) for functions of matrices was shown to reduce the computational overhead and runtime required to evaluate a sequence of matrix function applications, when compared to the standard FOM approximation. I will discuss the development of rFOM, and show how it is not possible to develop a restarted implementation, resulting in excessive storage and orthogonalization costs as the number of iterations grows large.

As an alternative to restarts, I will introduce *sketched-recycled FOM* (*srFOM*), which incorporates randomized sketching [2, 3] into rFOM in order to avoid excessive orthogonalization costs when working with non-Hermitian matrices. I will show results of numerical experiments which demonstrate the kind of performance gains we can achieve through sketching.

## References

- [1] L. Burke and S. Güttel, *Krylov subspace recycling with randomized sketching for matrix functions*, Technical Report arXiv:2308.02290, arXiv, (2023) (To appear in *SIAM J. Matrix Anal. Appl.*, (2024)).
- [2] Y. Nakatsukasa and J. A. Tropp, *Fast & Accurate Randomized Algorithms for Linear Systems and Eigenvalue Problems*, *SIAM J. Matrix Anal. Appl.* **45** (2024), no. 2, 1183–1214.
- [3] S. Güttel and M. Schweitzer, *Randomized sketching for Krylov approximations of large-scale matrix functions*, *SIAM J. Matrix Anal. Appl.* **44** (2023), no. 3, 1073–1095.
- [4] M. L. Parks and E. de Sturler and G. Mackey and D. D. Johnson and S. Maiti, *Recycling Krylov subspaces for sequences of linear systems*, *SIAM J. Sci. Comput.* **28** (2006), no. 5, 1651–1674.
- [5] L. Burke and A. Frommer and G. Ramirez-Hidalgo and K. M. Soodhalter, *Krylov subspace recycling For matrix functions*, Technical Report arXiv:2209.14163, (2022).

# Robust Spectral Clustering with Rank Statistics

*Joshua Cape, Xianshi Yu, Jonquil Zhongling Liao*

## Abstract

This talk investigates the performance of a robust spectral clustering method for latent structure recovery in noisy data matrices. We consider eigenvector-based clustering applied to a matrix of nonparametric rank statistics that is derived entrywise from the raw, original data matrix. This approach is robust in the sense that, unlike traditional spectral clustering procedures, it can provably recover population-level latent block structure even when the observed data matrix includes heavy-tailed entries and has a heterogeneous variance profile. Here, the raw input data may be viewed as a weighted adjacency matrix whose entries constitute links that connect nodes in an underlying graph or network.

Our main theoretical contributions are threefold and hold under flexible data generating conditions. First, we establish that robust spectral clustering with rank statistics can consistently recover latent block structure, viewed as communities of nodes in a graph, in the sense that unobserved community memberships for all but a vanishing fraction of nodes are correctly recovered with high probability when the data matrix is large. Second, we refine the former result and further establish that, under certain conditions, the community membership of any individual, specified node of interest can be asymptotically exactly recovered with probability tending to one in the large-data limit. Third, we establish asymptotic normality results associated with the truncated eigenstructure of matrices whose entries are rank statistics, made possible by synthesizing contemporary entrywise matrix perturbation analysis with the classical nonparametric theory of so-called simple linear rank statistics. Collectively, these results demonstrate the statistical utility of rank-based data transformations when paired with spectral techniques for dimensionality reduction. Numerical examples illustrate and support our theoretical findings. Additionally, for a dataset consisting of human connectomes, our approach yields parsimonious dimensionality reduction and improved recovery of ground-truth neuroanatomical cluster structure. We conclude with a discussion of extensions, practical considerations, and future work.

Reference: <https://arxiv.org/abs/2408.10136>, to appear in *Journal of Machine Learning Research*.

Author's note: As a statistician working on entrywise eigenvector perturbation analysis and with a background in applied mathematics, I am eager to engage with the numerical linear algebra community towards advancing research on topics of shared interest.

# The Stability of Split-Preconditioned FGMRES in Four Precisions

*Erin Carson, Ieva Daužickaitė*

## Abstract

We consider the problem of solving a linear system of equations  $Ax = b$ , where  $A \in \mathbb{R}^{n \times n}$  is nonsymmetric and  $x, b \in \mathbb{R}^n$ . When  $A$  is large and sparse, the iterative generalized minimal residual method (GMRES) or its flexible variant (FGMRES) are often used. In these and other Krylov subspace methods, preconditioning is an essential ingredient. Given a preconditioner  $P = M_L M_R$ , the original problem is transformed to

$$M_L^{-1} A M_R^{-1} \tilde{x} = M_L^{-1} b, \quad \text{where } M_R^{-1} \tilde{x} = x.$$

The emergence of mixed precision hardware has motivated work in developing mixed precision algorithms for matrix computations; see, e.g., the recent survey [4]. Modern GPUs offer double, single, half, and even quarter precision, along with specialized tensor core instructions; see, e.g., [5]. The use of lower precision can offer significant performance improvements, although this comes at a numerical cost. With fewer bits, we have a greater unit roundoff and a smaller range of representable numbers. The goal is thus to selectively use different precisions within algorithms such that performance is potentially improved without adversely affecting the desired numerical properties.

In this talk, based on the published work [3], we consider the split-preconditioned FGMRES method in a mixed precision framework, in which four potentially different precisions can be used for computations with the coefficient matrix  $A$  (unit roundoff  $u_A$ ), left-preconditioner  $M_L$  (unit roundoff  $u_L$ ), right-preconditioner  $M_R$  (unit roundoff  $u_R$ ), and all other computations (unit roundoff  $u$ ).

Our analysis is applicable to general preconditioners with minimal assumptions. Briefly, following the strategy of [6], we assume that the application of  $M_L^{-1}$  and  $M_R^{-1}$  can be computed such that

$$\begin{aligned} fl(M_L^{-1} w_j) &= M_L^{-1} w_j + \Delta M_{L,j} w_j, & |\Delta M_{L,j}| &\leq c(n) u_L E_{L,j}, \\ fl(M_R^{-1} w_j) &= M_R^{-1} w_j + \Delta M_{R,j} w_j, & |\Delta M_{R,j}| &\leq c(n) u_R E_{R,j}, \end{aligned}$$

where  $fl(\cdot)$  denotes the quantity computed in floating point arithmetic,  $E_{L,j}$  and  $E_{R,j}$  have positive entries,  $w_j \in \mathbb{R}^n$ , and  $c(n)$  is a constant that depends on  $n$  only. Note that a particular strength of FGMRES is that it allows the right preconditioner to change throughout the iterations; for simplicity, we consider the case here where the preconditioners are static, although our results could be extended to allow dynamic preconditioning.

We define  $\tilde{A} \equiv M_L^{-1} A$  and  $\tilde{b} \equiv M_L^{-1} b$  and assume that matrix-vector products with  $\tilde{A}$  can be computed so that

$$fl(\tilde{A} z_j) = (M_L^{-1} + \Delta M_{L,j})(A + \Delta A_j) z_j.$$

Denoting

$$u_A \psi_{A,j} = \frac{\|M_L^{-1} \Delta A_j z_j\|}{\|\tilde{A}\| \|z_j\|} \quad \text{and} \quad u_L \psi_{L,j} = \frac{\|\Delta M_{L,j} A z_j\|}{\|\tilde{A}\| \|z_j\|},$$

where  $\|\cdot\|$  denotes the 2-norm, and ignoring the second order terms, we can write

$$fl(\tilde{A} z_j) \approx \tilde{A} z_j + f_j, \quad \text{where } \|f_j\| \leq (u_A \psi_{A,j} + u_L \psi_{L,j}) \|\tilde{A}\| \|z_j\|.$$

We first present general bounds on the backward and forward errors in split-preconditioned FGMRES, which is based on the previous works [1] and [2]. Our analysis provides guidance on how the precisions should be set when the target backward error is of order  $u$ . To summarize, the precision for applying  $M_L$  must be chosen in relation to  $u$ ,  $u_A$ , and the required backward and forward errors, because  $u_L$  heavily influences the achievable backward error. We can be more flexible when choosing  $u_R$  as it does not influence the backward error directly. Our analysis holds under a sufficient but not necessary assumption on  $u_R$  in relation to  $M_R$ . As long as  $M_R$  is not singular in precision  $u_R$  (note that scaling strategies may be used to ensure this), setting  $u_R$  to a low precision is sufficient. Very low precisions  $u_L$  and  $u_R$  may delay the convergence, but setting  $u_L \leq u$  or  $u_R \leq u$  does not improve the convergence in general. Note that these conclusions apply to the full left- and right-preconditioning cases as well.

We observe that the forward error is determined by the backward error and the condition number of the left-preconditioned coefficient matrix. This motivates concentrating effort on constructing an appropriate left-preconditioner when aiming for a small forward error: the preconditioner should reduce the condition number sufficiently and needs to be applied in a suitably chosen precision.

We further provide insights on which preconditioning strategy (left, right, or split) may be preferred under certain objectives related to the desired the backward and forward errors. To summarize, if a small backward error is the main concern and  $A$  is ill-conditioned, and we have a ‘good’ preconditioner, so that  $\kappa(\tilde{A})$  is small and we can afford setting  $u_A$  and  $u_L$  to precisions that are high enough to neutralize the  $\psi_A$  and  $\psi_L$  terms, then left-preconditioning should be used. If however, we cannot afford setting  $u_A$  and  $u_L$  to high precisions but can construct a split-preconditioner such that  $\kappa(M_L)$  is small, then split-preconditioning (note that in this case  $\psi_A$  and  $\psi_L$  may be smaller too) or full right-preconditioning may be preferential. If our main concern is applying the preconditioner in lower than the working precision (which may be relevant, for example, when  $A$  is very sparse and the preconditioner uses some dense factors), the bounds suggest that full left-preconditioning should not be used as  $u_A\psi_A$  and  $u_L\psi_L$  may be large. Full right-preconditioning may be most suitable in this case.

We present a suite of numerical experiments which support our theoretical results. Essentially, the experiments confirm that the precision in which the left preconditioner is applied has a significant effect on the forward and backward errors, but very little effect on the number of FGMRES iterations required for convergence. Conversely, the precision in which the right preconditioner is applied has almost no effect on the resulting forward and backward errors, but can affect the FGMRES convergence.

## References

- [1] Mario Arioli and Iain S Duff. Using FGMRES to obtain backward stability in mixed precision. *Electronic Transactions on Numerical Analysis*, 33:31–44, 2009.
- [2] Mario Arioli, Iain S Duff, Serge Gratton, and Stéphane Pralet. A note on GMRES preconditioned by a perturbed  $LDL^T$  decomposition with static pivoting. *SIAM Journal on Scientific Computing*, 29(5):2024–2044, 2007.
- [3] Erin Carson and Ieva Daužickaitė. The stability of split-preconditioned FGMRES in four precisions. *Electronic Transactions on Numerical Analysis*, 60:40–58, 2024.

- [4] Nicholas J. Higham and Theo Mary. Mixed precision algorithms in numerical linear algebra. *Acta Numerica*, 31:347–414, 2022.
- [5] NVIDIA H100 Tensor Core GPU. NVIDIA, <https://www.nvidia.com/en-us/data-center/h100/>, 2023.
- [6] Bastien Vieublé. *Mixed precision iterative refinement for the solution of large sparse linear systems*. PhD thesis, INP Toulouse, University of Toulouse, France, 2022.

# A low-memory Lanczos method with rational Krylov compression for matrix functions

Angelo A. Casulli, Igor Simunec

## Abstract

A fundamental problem in numerical linear algebra is the approximation of the action of a matrix function  $f(A)$  on a vector  $\mathbf{b}$ , where  $A \in \mathbb{C}^{n \times n}$  is a matrix that is typically large and sparse,  $\mathbf{b} \in \mathbb{C}^n$  is a vector and  $f$  is a function defined on the spectrum of  $A$ . In this work, we focus on the case of a Hermitian matrix  $A$ . We recall that when  $A$  is Hermitian, given an eigendecomposition  $A = UDU^H$ , the matrix function  $f(A)$  is defined as  $f(A) = Uf(D)U^H$ , where  $f(D)$  is a diagonal matrix obtained by applying  $f$  to each diagonal entry of  $D$ . We refer to [12] for an extensive discussion of matrix functions.

Popular methods for the approximation of  $f(A)\mathbf{b}$  are polynomial [16, 13, 8, 7, 11] and rational Krylov methods [6, 15, 9, 1, 3]. The former only access  $A$  via matrix-vector products, while the latter require the solution of shifted linear systems with  $A$ . When the linear systems can be solved efficiently, rational Krylov methods can be more effective than polynomial Krylov methods since they usually require much fewer iterations to converge. However, there are several situations in which rational Krylov methods are not applicable, either because the matrix  $A$  is only available implicitly via a function that computes matrix-vector products, or because  $A$  is very large and the solution of linear systems is prohibitively expensive.

When  $A$  is Hermitian, the core component of a polynomial Krylov method is the Lanczos algorithm [14], which constructs an orthonormal basis  $\mathbf{Q}_M = [\mathbf{q}_1 \dots \mathbf{q}_M]$  of the polynomial Krylov subspace  $\mathcal{K}_M(A, \mathbf{b}) = \text{span}\{\mathbf{b}, A\mathbf{b}, \dots, A^{M-1}\mathbf{b}\}$  by exploiting a short-term recurrence. The product  $f(A)\mathbf{b}$  can then be approximated by the Lanczos approximation

$$\mathbf{f}_M := \mathbf{Q}_M f(\mathbf{T}_M) \mathbf{e}_1 \|\mathbf{b}\|_2, \quad \mathbf{T}_M := \mathbf{Q}_M^H A \mathbf{Q}_M, \quad (1)$$

where  $\mathbf{e}_1$  denotes the first unit vector. The Lanczos algorithm uses a short-term recurrence in the orthogonalization step, so each new basis vector is orthogonalized only against the last two basis vectors, and only three vectors need to be kept in memory to compute the basis  $\mathbf{Q}_M$ . Although the basis  $\mathbf{Q}_M$  and the projected matrix  $\mathbf{T}_M$  can be computed by using the short-term recurrence that only requires storage of the last three basis vectors, forming the approximate solution  $\mathbf{f}_M$  still requires the full basis  $\mathbf{Q}_M$ . When the matrix  $A$  is very large, there may be a limit on the maximum number of basis vectors that can be stored, so with a straightforward implementation of the Lanczos method there is a limit on the number of iterations of Lanczos that can be performed and hence on the attainable accuracy. In the literature, several strategies have been proposed to deal with low memory issues. See the recent surveys [10, 11] for a comparison of several low-memory methods.

In this presentation we propose a new low-memory algorithm for the approximation of  $f(A)\mathbf{b}$ . Our method combines an outer Lanczos iteration with an inner rational Krylov subspace, which is used to compress the outer Krylov basis whenever it reaches a certain size.

The fundamental insight underlying this method is that, leveraging the results presented in [2], the vector  $\mathbf{f}_M$  defined in (1) (for simplicity, assuming  $\|\mathbf{b}\|_2 = 1$ ) can be approximated by

$$\mathbf{f}_M \approx \mathbf{Q}_M \begin{bmatrix} f(T_1)\mathbf{e}_1 - U_1 f(U_1^H T_1 U_1) U_1^H \mathbf{e}_1 \\ 0 \end{bmatrix} + \mathbf{Q}_M \begin{bmatrix} U_1 & I \end{bmatrix} f \left( \begin{bmatrix} U_1^H & \\ & I \end{bmatrix} \mathbf{T}_M \begin{bmatrix} U_1 & \\ & I \end{bmatrix} \right) \begin{bmatrix} U_1^H \mathbf{e}_1 \\ 0 \end{bmatrix},$$

where  $T_1$  is an  $m \times m$  leading principal submatrix of  $\mathbf{T}_M$ , and  $U_1$  is an orthonormal basis of a rational Krylov subspace generated using the small matrix  $T_1$ . One can observe that the first summand of this expression can be computed after  $m$  steps of the Lanczos algorithm. Moreover, once the first term has been computed, it is no longer necessary to keep all the first  $m$  columns of the matrix  $\mathbf{Q}_M$  in memory, since computing the second term only requires the few vectors obtained by multiplying the first  $m$  columns of  $\mathbf{Q}_M$  on the right by the matrix  $U_1$ . Finally, the second term can be computed by recursively applying the same procedure.

Similarly to [4], the inner rational Krylov subspace does not involve the matrix  $A$ , but only small matrices. This is fundamental, since constructing a basis of the inner subspace does not require the solution of linear systems with  $A$ , and hence it is cheap compared to the cost of the outer Lanczos iteration. Theoretical results show that the approximate solutions computed by our algorithm coincide with the ones constructed by the outer Krylov subspace method when  $f$  is a rational function, and for a general function they differ by a quantity that depends on the best rational approximant of  $f$  with the poles used in the inner rational Krylov subspace.

If the outer Krylov basis is compressed every  $m$  iterations and the inner rational Krylov subspace has  $k$  poles, our approach requires the storage of approximately  $m + k$  vectors. Additionally, due to the basis compression, our approximation involves the computation of matrix functions of size at most  $(m + k) \times (m + k)$ , so the cost does not grow with the number of iterations. This represents an important advantage with respect to the Lanczos method, since when the number of iterations is very large the evaluation of  $f$  on the projected matrix can become quite expensive.

Numerical experiments show that the proposed algorithm is competitive with other low-memory methods based on polynomial Krylov subspaces.

The content of this presentation draws on the findings presented in [5].

## References

- [1] L. Aceto, D. Bertaccini, F. Durastante, and P. Novati. Rational Krylov methods for functions of matrices with applications to fractional partial differential equations. *J. Comput. Phys.*, 396:470–482, 2019.
- [2] Bernhard Beckermann, Alice Cortinovis, Daniel Kressner, and Marcel Schweitzer. Low-rank updates of matrix functions II: rational Krylov methods. *SIAM J. Numer. Anal.*, 59(3):1325–1347, 2021.
- [3] Michele Benzi and Igor Simunec. Rational Krylov methods for fractional diffusion problems on graphs. *BIT*, 62(2):357–385, 2022.
- [4] Angelo A. Casulli, Daniel Kressner, and Leonardo Robol. Computing functions of symmetric hierarchically semiseparable matrices, 2024.
- [5] Angelo A. Casulli and Igor Simunec. A low-memory Lanczos method with rational Krylov compression for matrix functions. *arXiv preprint arXiv:2403.04390*, 2024.
- [6] Vladimir Druskin and Leonid Knizhnerman. Extended Krylov subspaces: approximation of the matrix square root and related functions. *SIAM J. Matrix Anal. Appl.*, 19(3):755–771, 1998.

- [7] Andreas Frommer, Stefan Güttel, and Marcel Schweitzer. Efficient and stable Arnoldi restarts for matrix functions based on quadrature. *SIAM J. Matrix Anal. Appl.*, 35(2):661–683, 2014.
- [8] Andreas Frommer and Valeria Simoncini. Matrix functions. In *Model Order Reduction: Theory, Research Aspects and Applications*, volume 13 of *Math. Ind.*, pages 275–303. Springer, Berlin, 2008.
- [9] Stefan Güttel. Rational Krylov approximation of matrix functions: numerical methods and optimal pole selection. *GAMM-Mitt.*, 36(1):8–31, 2013.
- [10] Stefan Güttel, Daniel Kressner, and Kathryn Lund. Limited-memory polynomial methods for large-scale matrix functions. *GAMM-Mitt.*, 43(3):e202000019, 19, 2020.
- [11] Stefan Güttel and Marcel Schweitzer. A comparison of limited-memory Krylov methods for Stieltjes functions of Hermitian matrices. *SIAM J. Matrix Anal. Appl.*, 42(1):83–107, 2021.
- [12] Nicholas J. Higham. *Functions of Matrices: Theory and Computation*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2008.
- [13] Marlis Hochbruck and Christian Lubich. On Krylov subspace approximations to the matrix exponential operator. *SIAM J. Numer. Anal.*, 34(5):1911–1925, 1997.
- [14] Cornelius Lanczos. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *J. Research Nat. Bur. Standards*, 45:255–282, 1950.
- [15] I. Moret and P. Novati. RD-rational approximations of the matrix exponential. *BIT*, 44(3):595–615, 2004.
- [16] Y. Saad. Analysis of some Krylov subspace approximations to the matrix exponential operator. *SIAM J. Numer. Anal.*, 29(1):209–228, 1992.

# Convergence Behavior of GMRES on Tridiagonal Toeplitz Systems

*Fei Chen, Kirk M. Soodhalter*

## Abstract

Discretizing PDEs leads to linear systems with large, sparse coefficient matrices. When linear, constant-coefficient PDEs with Dirichlet boundary conditions are discretized on uniform meshes, one can obtain Toeplitz, multilevel Toeplitz and/or block Toeplitz systems [1, 2]. Toeplitz matrices have constant diagonals, and multilevel and block Toeplitz matrices have related structures, that can be exploited to speed up GMRES, and aid convergence analysis. Such systems are widely solved by Krylov subspace methods.

Let

$$Ax = b, \quad (1)$$

where the matrix  $A$  is Toeplitz,  $b$  is a known right-hand side, and  $x$  is the unknown solution. GMRES starts with an initial guess,  $x_0$ , and select  $x_k, k = 1, 2, \dots$ , such that  $x_k - x_0 \in \mathcal{K}_k(A, r_0) := \text{span} \{r_0, Ar_0, \dots, A^{(k-1)}r_0\}$ , where  $r_0 = A(x - x_0)$ .

Let  $Y$  be the reverse identity matrix, then  $YA$  is symmetric Hankel. One can solve (1) through

$$YAx = Yb, \quad (2)$$

by applying MINRES [3], which is mathematically equivalent to GMRES for a symmetric system. Through our experiments, we find that MINRES on (2) requires about twice as many iterations as GMRES on (1) to converge, especially when preconditioned.

For a symmetric system such as (2), the convergence behavior of MINRES can usually be characterized by the eigenvalues and the RHS. However, when  $A$  is nonsymmetric, GMRES convergence behavior is much more complicated to describe; in extreme cases the spectrum bears no relation to the convergence rate. In [4], the authors prove that any nonincreasing convergence curve is possible for GMRES by constructing a linear system of prescribed nonzero eigenvalues with a given convergence curve.

In this work, we explore the convergence behavior of GMRES when applied to real tridiagonal Toeplitz systems, where the matrix

$$A = \begin{bmatrix} \alpha & \gamma & 0 & \cdots & 0 \\ \beta & \alpha & \gamma & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \beta & \alpha & \gamma \\ 0 & 0 & 0 & \beta & \alpha \end{bmatrix}, \quad (3)$$

$\alpha, \beta$ , and  $\gamma \in \mathbb{R}$ .

We show that different GMRES convergence behavior is possible for different Toeplitz systems that share the same spectra, regardless of their right hand sides. We also explore the connection between

the GMRES convergence behavior and the singular values of the tridiagonal Toeplitz matrices.

In spite of the difficulties, there has been plenty of work in the literature inspecting the convergence behavior of GMRES. For instance, in [5], the authors point out that, for a general nonsingular matrix  $A$ , the convergence behavior of GMRES is related to the distribution of eigenvalues of  $A$ , and provide an upper bound. However, each eigenvalue is either treated as a member of some cluster, or an outlier to any cluster. For the cases where eigenvalues are not spreading out far away from each other, for example, those of a tridiagonal Toeplitz matrix, or when the clusters are far away from each other, one fails to find a meaningful upper bound since it becomes too loose. In [6], Meurant shows through APS parametrization of  $A$  that GMRES could have different convergence behaviors for two different matrices with the same spectrum. Nevertheless, a reconstructed matrix  $A$  in this case does not preserve the Toeplitz structure in general. As for tridiagonal matrix systems, Liesen and Strakoš analyze the convergence behavior of GMRES when  $|\alpha| \approx |\beta| \gg |\gamma|$  [7]. For a more general case where  $|\beta| \neq |\gamma|$ , Li and Zhang provide upper bounds and asymptotic speeds of the 2-norm of the  $k^{\text{th}}$  residual via Chebyshev polynomial of the first kind[8] and the second kind[9]. In our work, we formulate equations based on APS parametrization of  $A$  with the constraint that each diagonal is constant to explore what convergence regimes are possible for a tridiagonal Toeplitz system.

## References

- [1] C. Garoni and S. Serra-Capizzano, *Generalized Locally Toeplitz Sequences: Theory and Applications*, vol. I Springer Cham, 2017.
- [2] C. Garoni and S. Serra-Capizzano, *Generalized Locally Toeplitz Sequences: Theory and Applications*, vol. II Springer Cham, 2018.
- [3] J. Pestana and A. Wathen, *A preconditioned MINRES method for nonsymmetric Toeplitz matrices*, SIAM J. Matrix Anal. Appl., vol 36(2015), no. 1, 273-288.
- [4] A. Greenbaum, V. Pták, Z. Strakoš *Any non increasing convergence curve is possible for GMRES*, SIAM J. Matrix Anal. Appl., vol 17(1996), no. 3, 465-469.
- [5] S. L. Campbell, I. C. F. Ipsen, C. T. Kelley and C. D. Meyer, *GMRES and the minimal polynomial*, BIT Numerical Mathematics, vol 36(1996), no. 4, 664-675.
- [6] G. Meurant, *GMRES and the Arioli, Pták, and Strakoš parametrization*, BIT Numerical Mathematics, vol 52(2012), 687-702.
- [7] J. Liesen and Z. Strakoš, *Convergence of GMRES for tridiagonal Toeplitz matrices*, SIAM J. Matrix Anal. Appl., vol 26(2004), no. 1, 233-251.
- [8] R. C. Li and W. Zhang, *The rate of convergence of GMRES on a tridiagonal Toeplitz linear system*, Numerische Mathematik, vol 112(2009), 267-293.
- [9] R. C. Li and W. Zhang, *The rate of convergence of GMRES on a tridiagonal Toeplitz linear system. II*, Linear Algebra and its Applications, vol 421(2009), 2425-2436.

# Preconditioning without a preconditioner: faster ridge-regression and Gaussian sampling with randomized block Krylov methods

Tyler Chen, Caroline Huber, Ethan Lin, Hajar Zaid

Abstract

One of the most important tasks in numerical linear algebra is solving the linear system

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \quad (1)$$

where  $\mathbf{A} \in \mathbb{R}^{d \times d}$  is symmetric positive definite with eigenvalues  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d > 0$ . Krylov subspace methods (KSMs) such as the conjugate gradient method are among the most powerful methods for this problem and are guaranteed to converge extremely rapidly if the system is well-conditioned; i.e. if  $\lambda_1 \approx \lambda_d$ . For ill-conditioned systems, *preconditioning* can greatly accelerate the convergence of KSMs. When  $\mathbf{A}$  has a rapidly decaying spectrum, a technique called Nyström preconditioning has proven effective [1].

Consider the Nyström approximation

$$\mathbf{A}\langle\mathbf{K}_s\rangle := (\mathbf{A}\mathbf{K}_s)(\mathbf{K}_s^\top \mathbf{A}\mathbf{K}_s)^\dagger(\mathbf{K}_s^\top \mathbf{A}), \quad (2)$$

where  $\Omega \in \mathbb{R}^{d \times (r+2)}$  is a matrix of independent standard normal random variables and  $\mathbf{K}_s := [\Omega \ \mathbf{A}\Omega \ \dots \ \mathbf{A}^{s-1}\Omega] \in \mathbb{R}^{d \times s(r+2)}$ . It can be guaranteed that if  $s = O(\log(d))$ , then with high probability,  $\mathbf{A}\langle\mathbf{K}_s\rangle$  approximates  $\mathbf{A}$  with spectral-norm error comparable to the best-possible rank- $r$  approximation to  $\mathbf{A}$ ; i.e.  $\|\mathbf{A} - \mathbf{A}\langle\mathbf{K}_s\rangle\| = O(\lambda_{r+1})$  [3]. Define a preconditioner

$$\mathbf{P} := \frac{1}{\lambda_{r+1}} \mathbf{U}\mathbf{D}\mathbf{U}^\top + (\mathbf{I} - \mathbf{U}\mathbf{U}^\top), \quad (3)$$

where  $\mathbf{U}\mathbf{D}\mathbf{U}^\top$  is the eigendecomposition of  $\mathbf{A}\langle\mathbf{K}_s\rangle$ . Following the approach of [1], we show that if  $\theta \in [\lambda_d, \lambda_{r+1}]$  and  $s = O(\log(d))$ , then with high probability, then

$$\kappa(\mathbf{P}^{-1/2}\mathbf{A}\mathbf{P}^{-1/2}) = O(\lambda_{r+1}/\lambda_d). \quad (4)$$

As a result, preconditioned-CG with the preconditioner (3) converges at a rate depending on  $\sqrt{\lambda_{r+1}/\lambda_d}$  [2]. If  $\mathbf{A}$  has just  $r$  large eigenvalues, the convergence of preconditioned-CG will be extremely rapid.

One downside to Nyström preconditioning is the need to choose hyperparameters such as  $\theta$  and  $s$ . Our observation is that, after  $t$  iterations, block-CG with a starting block  $[\mathbf{b} \ \Omega]$  has error at most that of Nyström preconditioned CG after  $t - s - 1$  iterations. Thus, *block-CG enjoys the effects of (Nyström) preconditioning, without the need for constructing a preconditioner or choose parameters*.<sup>1</sup> This allows us to prove the following convergence guarantee.<sup>2</sup>

**Theorem 1.** *Fix a value  $r \geq 0$  and let  $\mathbf{b}_2, \dots, \mathbf{b}_{r+2}$  be independent standard Gaussian vectors. Then after  $t$  iterations the block-CG iterate  $\mathbf{x}_t^{\text{b-CG}}$  corresponding to a starting block  $[\mathbf{b} \ \mathbf{b}_2 \ \dots \ \mathbf{b}_{r+2}]$  satisfies, with probability at least 99/100,*

$$\frac{\|\mathbf{A}^{-1}\mathbf{b} - \mathbf{x}_t^{\text{b-CG}}\|_\mathbf{A}}{\|\mathbf{A}^{-1}\mathbf{b}\|_\mathbf{A}} \leq 2 \exp\left(-\frac{t - (3 + \log(d)/2)}{3\sqrt{\lambda_{r+1}/\lambda_d}}\right).$$

<sup>1</sup>We are assuming iterations, not matrix-vector products, are the dominant cost.

<sup>2</sup>This bound is reminiscent of the “killing off the top eigenvalues” bounds for CG. However, instead of a burn-in period of  $r$  iterations, we require a burn-in period of  $O(\log(d))$  iterations (independent of  $r$ ).

More generally, for any  $\mu \geq 0$ , block-CG (and Nyström preconditioned CG) can be used to solve the regularized linear system

$$(\mathbf{A} + \mu\mathbf{I})\mathbf{x} = \mathbf{b}. \quad (5)$$

Systems of the form (5) arise in a variety of settings, but we are particularly motivated by two critical tasks in machine learning and data science: solving ridge-regression problems and sampling Gaussian vectors. By adapting our bound [Theorem 1](#) for block-CG, we obtain state-of-the-art convergence guarantees for existing Lanczos-based methods used to solve these tasks.

## References

- [1] Z. Frangella, J. Tropp, and M. Udell (2023). Randomized Nyström Preconditioning. *SIAM Journal on Matrix Analysis and Applications*, 44(2), 718–752.
- [2] A. Greenbaum (1997). *Iterative Methods for Solving Linear Systems*. Society for Industrial and Applied Mathematics.
- [3] J. Tropp and R. Webber (2023). Randomized algorithms for low-rank matrix approximation: Design, analysis, and applications.

# Surrogate-based Autotuning for Randomized Numerical Linear Algebra

Younghyun Cho, James W. Demmel, Michał Dereziński, Haoyun Li, Hengrui Luo,  
Michael W. Mahoney, Riley J. Murray

## Abstract

The field of Randomized Numerical Linear Algebra (RandNLA) has made significant developments and shown high quality empirical performance in some scenarios (e.g., overdetermined least-squares solvers). However, the practical performance of a RandNLA method usually hinges on the careful selection of multiple algorithm-specific tuning parameters. In addition, such a parameter selection would affect both the runtime of the algorithm and the accuracy of the result, which makes the parameter selection even harder. This motivates us to develop an automated process that helps find the (near-)optimal parameters for practical performance, with a focus on the applications relevant to RandNLA practitioners.

This extended abstract, which is based on our ongoing work [1], presents a surrogate-based autotuning approach for tuning RandNLA algorithms. We present a tuning pipeline that is built based on Bayesian optimization (BO) with Gaussian Process (GP) regression, which is an empirical approach where we aim to find the optimal parameter selection for a given tuning budget. At a high level, our pipeline follows the typical BO procedure, where we evaluate several parameter configurations, (iteratively) build a surrogate performance model based on the obtained evaluation results, and then find the next sample to evaluate until we reach the given tuning budget, along with an objective function to minimize the runtime of the algorithm while providing a satisfactory accuracy. Furthermore, we also apply a transfer learning approach to further reduce the tuning cost, especially when there are previously collected evaluation data from other similar but different tasks (e.g., the same algorithm but solving with different input data matrices). This makes the tuning approach more cost efficient and practical for RandNLA practitioners. The tuning pipeline uses GPTune [11] as the BO framework. GPTune is an open-source autotuner that was originally designed for tuning large-scale high-performance computing codes but is also general and can support tuning other domains of codes.

In particular, we show the efficacy of our tuning pipeline, in the context of sketch-and-precondition (SAP) based randomized least squares methods in solving large-scale overdetermined problems, minimizing  $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2$ , where  $\mathbf{A}$  is with the size of  $m$  by  $n$  with  $m \gg n$ , as SAP-based randomized least squares solvers that have been one of the successful applications in RandNLA. The SAP least squares approach can be summarized into following five steps: (1) Construct a sketching matrix  $\mathbf{S}$  (with size of  $d$  by  $m$ ; multiple schemes exist such as Sparse Johnson–Lindenstrauss Transform (SJLT) [5] and LessUniform [6, 7] to form a sketching matrix) to approximate the input data matrix  $\mathbf{A}$ , (2) Compute  $\hat{\mathbf{A}} = \mathbf{S}\mathbf{A}$ , (3) Generate a preconditioner matrix  $\mathbf{M}$  from  $\hat{\mathbf{A}}$  (e.g., using QR or SVD), (4) Use an iterative method for the preconditioned least squares for minimizing  $\|\mathbf{A}\mathbf{M}\mathbf{z} - \mathbf{b}\|_2^2$  (e.g., using preconditioned LSQR or preconditioned gradient descent (PGD)), and finally (5) Compute the result vector,  $\mathbf{M}\mathbf{z}$ .

We observe that the SAP-based least squares solver has multiple types of parameters to be tuned. The possible tuning parameters include some categorical variables to choose what the sparse sketching operator and the iterative solver for the preconditioned least squares to be used, as well as continuous/integer parameters to configure the size of the sketching matrix ( $d$  of  $\mathbf{S}$ ) as well as the sparsity of the sketching matrix (i.e., number of nonzero elements per row or column of  $\mathbf{S}$ ). In our experiments, we search this categorical space, using several implementations that are motivated by

the well-known works such as Blendenpik [2], LSRN [3], and NewtonSketch [4]. Then, we search a certain range of continuous/integer parameters to configure the sketching matrix, in terms of the size of the sketching matrix and the sparsity of the sketching matrix. In addition, the iterative solvers such as LSQR [8] and PGD finish their iterations based on the termination criteria with a desired level of accuracy (which we call “safety factor”). We regard that as a tuning parameter, and our tuning pipeline computes a relative residual error by comparing the results of the SAP least squares solver and the result obtained from a traditional direct solver. The relative error is used as the key indicator to quantify the quality of the SAP least squares solver for a given parameter configuration as well as the running time of the algorithm. For the SAP least squares solvers, we used a Python version prototype RandNLA package, PARLA [9], that provides the implementations for the SAP least squares solvers with the interface to control the abovementioned parameters.

We use multiple synthetic matrices and several real-world input matrices to test the efficacy of our tuning pipeline [1]. Our experimental results show promising results that GP-based BO approach is effective in tuning the parameters for RandNLA algorithms, in comparison with other primitives such as random search or grid search. Moreover, we also show that transfer learning can further improve the tuning efficiency by leveraging the data obtained from other input data matrices. For transfer learning, within the Bayesian optimization process, our tuner chooses the categorical variable, i.e., the SAP algorithm and the sketching operator, using the Upper Confidence Bound (UCB) bandit function, and then we apply a GP-based multitask learning technique [12], called Linear Coregionalization Model (LCM), in order to learn from historical samples within the same chosen category from the source matrices. That improves the tuning quality and cost, compared to non transfer learning-based tuning. Overall, the success of the empirical tuning approach suggests possible practical use cases. For example, users can use our autotuning pipeline in order select the parameters for running a RandNLA algorithm. If the user has a larger dataset size, the user can down-sample their input data and perform autotuning (with or without transfer learning), and then use the chosen parameter configuration to run the algorithm on a larger dataset.

For future work, our tuning pipeline can be extended or tested for other RandNLA problems. While our experiments have primarily focused on the problem of overdetermined least squares, the basic lessons from our work are applicable in other contexts, such as low-rank approximation, and also for tuning large-scale high-performance computing applications. In addition, our tuning pipeline can further be improved to be even more robust and effective in tuning RandNLA workloads that are hard to achieve valid parameter configurations for a given residual accuracy requirement. From a theoretical perspective, the integration of surrogate-based optimization techniques with RandNLA algorithms opens up new avenues for research at the intersection of machine learning and numerical linear algebra. We can also explore how these autotuning techniques could be incorporated directly into adaptive algorithms, allowing numerical methods to automatically adjust their behavior based on the properties of the input data. In conclusion, the development of these surrogate-based autotuning techniques represents a significant step forward in bridging the gap between theoretical advances in RandNLA and their practical performance engineering.

## References

- [1] Y. CHO, J. W. DEMMEL, M. DEREZIŃSKI, H. LI, H. LUO, M. W. MAHONEY, R. J. MURRAY, *Surrogate-based Autotuning for Randomized Sketching Algorithms in Regression Problems*, in arXiv:2308.15720 (2023).

- [2] H. AVRON, P. MAYMOUNKOV, AND S. TOLEDO, *Blendenpik: Supercharging LAPACK's Least-Squares Solver*, SIAM Journal on Scientific Computing, 32 (2010).
- [3] X. MENG, M. A. SAUNDERS, AND M. W. MAHONEY, *LSRN: A parallel iterative solver for strongly over- or underdetermined systems*, SIAM Journal on Scientific Computing, 36 (2014).
- [4] M. PILANCI AND M. J. WAINWRIGHT, *Newton Sketch: A near linear-time optimization algorithm with linear-quadratic convergence*, SIAM Journal on Optimization, 27 (2017).
- [5] A. DASGUPTA, R. KUMAR, AND T. SARLOS, *A sparse Johnson-Lindenstrauss transform*, in Proceedings of the Forty-Second ACM Symposium on Theory of Computing (STOC), STOC '10, 2010, Association for Computing Machinery, p. 341–350.
- [6] M. DEREZIŃSKI, Z. LIAO, E. DOBRIBAN, AND M. MAHONEY, *Sparse sketches with small inversion bias*, in Conference on Learning Theory (COLT), PMLR, 2021, pp. 1467–1510.
- [7] M. DEREZIŃSKI, J. LACOTTE, M. PILANCI, AND M. W. MAHONEY, *Newton-LESS: Sparification without trade-offs for the sketched Newton update*, Advances in Neural Information Processing Systems, 34 (2021).
- [8] C. C. PAIGE AND M. A. SAUNDERS, *LSQR: An algorithm for sparse linear equations and sparse least squares*, ACM Trans. Math. Softw., 8 (1982), pp. 43–71.
- [9] BALLISTIC, *Python Algorithms for Randomized Linear Algebra (PARLA)*, 2022, <https://github.com/BallisticLA/parla/tree/main>.
- [10] R. MURRAY, J. DEMMEL, M. W. MAHONEY, N. B. ERICHSON, M. MELNICHENKO, O. A. MALIK, L. GRIGORI, P. LUSCZEK, M. DEREZIŃSKI, M. E. LOPES, T. LIANG, H. LUO, AND J. DONGARRA, *Randomized Numerical Linear Algebra : A perspective on the field with an eye to software*, arXiv:2302.11474v2 (2023).
- [11] Y. CHO, J. W. DEMMEL, G. DINH, X. S. LI, Y. LIU, H. LUO, O. MARQUES, AND W. M. SID-LAKHDAR, *GPTune user guide*. <https://gptune.lbl.gov/documentation/gptune-user-guide>, 2022.
- [12] Y. LIU, W. M. SID-LAKHDAR, O. MARQUES, X. ZHU, C. MENG, J. W. DEMMEL, AND X. S. LI, *GPTune: Multitask learning for autotuning exascale applications*, in Proceedings of the 26th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, PPoPP '21, 2021, Association for Computing Machinery, pp. 234–246.

# Online Machine Learning for Solving a Sequence of Linear Systems

*Mikhail Khodak, Edmond Chow, Maria-Florina Balcan, Ameet Talwalkar*

## Abstract

Machine learning is often presented as an alternative to well-established and effective numerical methods. In this work, we present an example where machine learning is used to augment existing numerical methods.

Consider solving a sequence of linear systems

$$A_t x = f_t, \quad t = 1, \dots, T$$

with SOR( $\omega$ ), or some other preconditioner–solver combination in general, where we need to choose a parameter for the preconditioner or solver for each system. We are to solve each system before the next system is presented to us. Our goal is to choose the SOR parameter  $\omega$  for each system to minimize the total number of iterations. To accomplish this, we can make use of the information about the number of iterations used to solve previous systems.

There must be some assumptions for us to do anything interesting. We could assume, for example, that the sequence of matrices  $\{A_t\}$  changes slowly. This type of assumption could be useful if we are further allowed to use a method to obtain a good estimate of  $\omega$ , when needed. Then, we could use this value of  $\omega$  for solving several linear systems until the number of iterations required for a system becomes so large that it becomes profitable to estimate a new value of  $\omega$ . This kind of strategy has appeared in various guises in the literature and is perhaps the best competitor strategy to what we will present here.

In this work, we consider using multi-armed bandits from online machine learning to select the value of  $\omega$  for solving each system. Such algorithms are very effective for the following class of practical problems. Suppose every time a user visits your web page, you have the choice of showing an advertisement in one of four locations: top, bottom, left, and right. You wish to choose the location each time to maximize the total number of times users visiting your web page will click on the advertisement. The underlying assumption is that there is an unknown probability that a user will click on the advertisement in each of the four cases. Your problem is to discover the case that has the highest such probability (exploration), while also trying to maximize the number of clicks (exploitation) by not wasting time on low probability cases, and possibly not knowing the number of users your web page will ultimately have. Formally, the multi-armed bandit problem is the following:

### Multi-armed bandit problem

```
for  $t = 1, \dots, T$  do
    Choose and perform action  $a_t$  from  $\{1, \dots, d\}$ 
    Receive reward (or loss)  $y_t$ 
        Different actions lead to different rewards.
        Do not see rewards for actions not taken.
end for
```

The actions are choosing among the four locations where we can place the advertisement. For our sequence of linear systems, the actions are choosing a (discretized) value of  $\omega$ . The goal is to choose the actions such that the *cumulative regret* is minimized. The cumulative regret is the difference between the expected reward for the single best action and the expected reward for our choice of

actions, summed over  $t$  rounds. In particular, the goal is to obtain strategies that give cumulative regret that is sublinear in  $t$ .

We address two types of assumptions about our sequence of linear systems: (1) the optimal  $\omega$  follows a fixed distribution and (2) the optimal  $\omega$  follows a distribution that changes. Case (1) can be handled with stochastic bandits such as UCB1, an upper confidence bound algorithm. Case (2) can be handled with adversarial bandits such as Exp3, the exponential-weight algorithm for exploration and exploitation. We further look at sequences of matrices of the form  $A_t = A + c_t I$  where the scalar shift  $c_t$  is known before  $\omega$  is chosen. This case can be handled by contextual bandits.

The simplest contextual bandit algorithm will discretize the contexts (shifts) into intervals and use an adversarial bandit separately on each interval. However, we want an approach that exploits the smoothness of the optimal mapping from the context (shift  $c$ ) to the action ( $\omega$ ). For this, we reduce the online contextual bandit problem to a problem of online regression to finding a weight vector  $w$  given observations that arrive in sequence:

**Online regression protocol** for  $y = f(x; w)$

```

Initialize regression weights  $w$ 
for  $t = 1, \dots, T$  do
    Observe  $x_t$ 
    Predict  $\hat{y}_t = f(x_t; w)$ 
    Observe  $y_t$  and suffer loss  $(\hat{y}_t - y_t)^2$ 
    Update  $w$ 
end for

```

In online regression, the goal is to choose the weights to minimize the cumulative loss. For our contextual bandit, we assume we have a good method for solving this problem (the oracle). In particular, in our contextual bandit, we use online regression to fit the loss vs. (context, action), i.e.,  $y$  = number of iterations vs.  $x = (c, \omega)$ .

An example of such an approach is the SquareCB algorithm (Foster and Rakhlin, 2020):

**SquareCB algorithm**

```

Input: learning rate  $\eta > 0$ , exploration parameter  $\mu > 0$ 
for  $t = 1, \dots, T$  do
    Observe context  $c_t$ 
    Compute  $\hat{y}_{t,a} = f(c_t, a; w)$  for all possible  $a$ 
     $b_t = \arg \min_a \hat{y}_{t,a}$ 
     $p_{t,a} = \frac{1}{\mu + \eta(\hat{y}_{t,a} - \hat{y}_{t,b_t})}, \quad \forall a \neq b_t$ 
     $p_{t,b_t} = 1 - \sum_{a \neq b_t} p_{t,a}$ 
    Sample  $a_t \sim p_t$  and perform action  $a_t$ 
    Observe actual loss  $y_t$ 
    Update the online regression oracle with example  $((c_t, a_t), y_t)$ 
end for

```

Above, the action  $a$  can be associated with possible values of  $\omega$  for our setting of solving a sequence of linear systems.

We develop a contextual bandit called ChebCB, a contextual bandit using Chebyshev regression. For each action (possible  $\omega$ ) separately, we fit the loss vs. context  $c$  using regularized polynomial regression. In particular, we use polynomials in a Chebyshev basis with coefficients for each

Chebyshev polynomial constrained to be small.

We do not show the results here, but tests on a 2-D heat equation with time-dependent coefficients and time-dependent forcing show that the ChebCB contextual bandit method asymptotically achieves the performance of the instance-optimal policy, which selects the best  $\omega$  for each instance.

In summary, this work shows the potential of using well-understood learning algorithms to augment and speed up linear system solvers, without sacrificing the ability to obtain high accuracy. Additional information can be found in the reference below.

- [1] M. Khodak, E. Chow, M.-F. Balcan, and A. Talwalkar, Learning to Relax: Setting Solver Parameters Across a Sequence of Linear System Instances, Proceedings of the 12th International Conference on Learning Representations (ICLR), 2024. Spotlight. <https://arxiv.org/abs/2310.02246>

# Efficient sample average approximation techniques for hyperparameter estimation in Bayesian inverse problems

*Julianne Chung, Malena Sabaté Landman, Scot M. Miller, Arvind K. Saibaba*

## Abstract

Inverse problems arise in many important applications, where the aim is to estimate some unknown inverse parameters from given observations. For large-scale problems where the number of unknowns can be large (e.g., due to the desire to reconstruct high-resolution images or dynamic image reconstructions) or for problems where observational datasets are huge, estimating the inverse parameters can be a computationally challenging task. Although there have been significant advancements in solving inverse problems, many of these approaches rely on a pre-determined, carefully-tuned set of hyperparameters (e.g., that define the noise and prior models) that must be estimated from the data. The need to estimate these hyperparameters further exacerbates the problem, often requiring repeated solves for many combinations of hyperparameters. In this work, we propose a sample average approximation (SAA) method that couples a Monte Carlo estimator with a preconditioned Lanczos method for the efficient estimation of hyperparameters in Bayesian inverse problems.

We are interested in linear inverse problems that involve recovering the parameters  $\mathbf{s} \in \mathbb{R}^n$  from measurements  $\mathbf{d} \in \mathbb{R}^m$ , which have been corrupted by additive Gaussian measurement noise,  $\boldsymbol{\eta} \in \mathbb{R}^m$ , and takes the form

$$\mathbf{d} = \mathbf{A}\mathbf{s} + \boldsymbol{\eta}, \quad \boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}(\boldsymbol{\theta}))$$

where  $\mathbf{A} \in \mathbb{R}^{m \times n}$  represents the forward map and  $\boldsymbol{\theta} \in \mathbb{R}_+^K$ , represents the (nonnegative) hyperparameters. In the hierarchical Bayes approach, we treat  $\boldsymbol{\theta}$  as a random variable, which we endow with prior density  $\pi_{\text{hyp}}(\boldsymbol{\theta})$ . We assume that the noise covariance matrix  $\mathbf{R} : \mathbb{R}_+^K \rightarrow \mathbb{R}^{m \times m}$ , where  $\mathbf{R}(\cdot)$  is symmetric and positive definite (SPD), and has an inverse and square root that is computationally easy to obtain for any input (e.g., a diagonal matrix or a scalar times the identity). We assume that the prior distribution for the parameters  $\mathbf{s}$  is also Gaussian of the form  $\mathcal{N}(\boldsymbol{\mu}(\boldsymbol{\theta}), \mathbf{Q}(\boldsymbol{\theta}))$ , where  $\boldsymbol{\mu} : \mathbb{R}_+^K \rightarrow \mathbb{R}^n$  and  $\mathbf{Q} : \mathbb{R}_+^K \rightarrow \mathbb{R}^{n \times n}$ , where  $\mathbf{Q}(\cdot)$  is assumed to be SPD.

With the above assumptions, we obtain the marginal posterior density,

$$\pi(\boldsymbol{\theta} | \mathbf{d}) \propto \pi_{\text{hyp}}(\boldsymbol{\theta}) \det(\boldsymbol{\Psi}(\boldsymbol{\theta}))^{-1/2} \exp\left(-\frac{1}{2} \|\mathbf{A}\boldsymbol{\mu}(\boldsymbol{\theta}) - \mathbf{d}\|_{\boldsymbol{\Psi}^{-1}(\boldsymbol{\theta})}^2\right), \quad (1)$$

where  $\boldsymbol{\Psi}(\boldsymbol{\theta}) = \mathbf{A}\mathbf{Q}(\boldsymbol{\theta})\mathbf{A}^\top + \mathbf{R}(\boldsymbol{\theta})$ . One goal would be to draw samples (e.g., using Markov Chain Monte Carlo) from (1), and using the samples to quantify the uncertainty in the hyperparameters. However, this may be prohibitive for large-scale problems because evaluating the density function (or its logarithm) requires evaluating the determinant of and multiple solves with the matrix  $\boldsymbol{\Psi}$  that depends on  $\boldsymbol{\theta}$ , which can be expensive. To compound matters, hundreds of samples are required to get accurate statistics, which can involve several hundred thousand density function evaluations.

Instead, we follow an empirical Bayes approach and focus on computing the maximum a posteriori (MAP) estimate, that is, the point estimate that maximizes the marginal posterior distribution or, equivalently, minimizes the negative log of the marginal posterior. That is, the problem of hyperparameter estimation becomes solving an optimization problem:

$$\min_{\boldsymbol{\theta} \in \mathbb{R}_+^K} \mathcal{F}(\boldsymbol{\theta}) \equiv -\log \pi_{\text{hyp}}(\boldsymbol{\theta}) + \frac{1}{2} \log \det(\boldsymbol{\Psi}(\boldsymbol{\theta})) + \frac{1}{2} \|\mathbf{A}\boldsymbol{\mu}(\boldsymbol{\theta}) - \mathbf{d}\|_{\boldsymbol{\Psi}(\boldsymbol{\theta})^{-1}}^2. \quad (2)$$

Notice that solving (2) is a computationally intensive task since it involves computing log determinants. To address this challenge, we consider an SAA method for computing the MAP estimate of the marginalized posterior distribution that combines a stochastic average approximation of the objective function and the preconditioned Lanczos method to compute efficient approximations of the function and gradient evaluations. The novel contributions of this work are as follows.

1. The method to estimate the objective function combines a Monte Carlo estimator for the log-determinant of the matrix with a preconditioned Lanczos approach to apply the matrix logarithm. We analyze the impact of the number of Monte Carlo samples and Lanczos iterations on the accuracy of the log-determinant estimator.
2. We use a novel preconditioner to accelerate the Lanczos iterations. The preconditioner is based on a parametric low-rank approximation of the prior covariance matrix, that is easy to update for new values of the hyperparameters. In particular, no access to the forward/adjoint solver is needed to update the preconditioner, and only a modest amount of precomputation is needed as a setup cost (independent of the optimization).
3. We also use a trace estimator to approximate the gradient that has two features: first, it works with a symmetric form of the argument inside the trace, and second, it is able to reuse Lanczos iterates from the objective function computations. Therefore, the gradient can be computed essentially for free (i.e., requiring no additional forward/adjoint applications).

**Related works.** The methods we describe here have some similarity to existing literature and share certain techniques in common. The problem of optimizing for hyperparameters is closely related to parameter estimation in Gaussian processes on maximum likelihood (we may think of it as setting the forward operator as the identity matrix). The literature on this topic is vast, but we mention a few key references that are relevant to our approach. In [3], the authors propose a matrix-free approach to estimate the hyperparameters and also use an SAA for optimization. In [2], the authors propose a reformulation of the problem that avoids computing the inversion of the (prior) covariance matrix. Approaches based on hierarchical matrices are considered in [8, 10, 1]. Preconditioned Lanczos methods for estimating the log-determinant and its gradient are considered in [6, 7]. However, the main difference is that the Gaussian process methods do not involve forward operators. This raises two issues: first, we have to account for the problem structure which is different from Gaussian processes, and second, we have to account for the computational cost of the forward operator (and its adjoint), which may be comparable or greater than the cost of the covariance matrices.

On the inverse problem side, there have been relatively few works on computing the hyperparameters by optimization. Several works (e.g., [4]) instead use sampling methods (e.g., Markov Chain Monte Carlo), but these methods are extremely expensive since they require several thousand evaluations of the likelihood to achieve accurate uncertainty estimates. In [9], we developed efficient methods for hyperparameter estimation based on low-rank approximations using the generalized Golub-Kahan iterative method. A brief review of other techniques is also given in the same paper.

## References

- [1] S. Ambikasaran, A. K. Saibaba, E. F. Darve, and P. K. Kitanidis. Fast algorithms for Bayesian inversion. In *Computational Challenges in the Geosciences*, pages 101–142. Springer, 2013.

- [2] M. Anitescu, J. Chen, and M. L. Stein. An inversion-free estimating equations approach for Gaussian process models. *Journal of Computational and Graphical Statistics*, 26(1):98–107, 2017.
- [3] M. Anitescu, J. Chen, and L. Wang. A matrix-free approach for solving the parametric Gaussian process maximum likelihood problem. *SIAM Journal on Scientific Computing*, 34(1):A240–A262, 2012.
- [4] J. M. Bardsley. Computational uncertainty quantification for inverse problems, volume 19 of *Computational Science & Engineering*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2018.
- [5] E. Chow and Y. Saad. Preconditioned Krylov subspace methods for sampling multivariate Gaussian distributions. *SIAM Journal on Scientific Computing*, 36(2):A588–A608, 2014.
- [6] K. Dong, D. Eriksson, H. Nickisch, D. Bindel, and A. G. Wilson. Scalable log determinants for Gaussian process kernel learning. *Advances in Neural Information Processing Systems*, 30, 2017.
- [7] J. Gardner, G. Pleiss, K. Q. Weinberger, D. Bindel, and A. G. Wilson. Gpytorch: Blackbox matrix-matrix Gaussian process inference with GPU acceleration. *Advances in neural information processing systems*, 31, 2018.
- [8] C. J. Geoga, M. Anitescu, and M. L. Stein. Scalable Gaussian process computations using hierarchical matrices. *Journal of Computational and Graphical Statistics*, 29(2):227–237, 2020.
- [9] K. A. Hall-Hooper, A. K. Saibaba, J. Chung, and S. M. Miller. Efficient iterative methods for hyperparameter estimation in large-scale linear inverse problems. arXiv preprint arXiv:2311.15827, 2023.
- [10] V. Minden, A. Damle, K. L. Ho, and L. Ying. Fast spatial Gaussian process maximum likelihood estimation via skeletonization factorizations. *Multiscale Modeling & Simulation*, 15(4):1584–1611, 2017.

# Bridging Linear Algebra and Autoencoders

Matthias Chung

## Abstract

In recent years *autoencoders* – mappings  $A_\theta : \mathcal{X} \rightarrow \mathcal{X}$  parameterized by  $\theta \in \mathbb{R}^\ell$  – have emerged as a cornerstone of machine learning and data science, playing a pivotal role in numerous applications. Their ability to learn efficient low-dimensional representations of data has led to significant advancements in fields such as image and natural language processing, anomaly detection, and generative modeling.

While universal approximation theorems provide a general theoretical foundation of autoencoder, various analytical aspects such as interpretability, robustness, network design, and hyperparameter selection remain relatively unexplored. *Numerical linear algebra* has played a fundamental and crucial role in the development of modern science and technology and its impact on autoencoders remains under-utilized.

The connection between linear autoencoder and singular value decomposition/principal component analysis has been laid out in various works. Recognizing the connection between linear autoencoders and singular value decomposition has sparked novel research utilizing autoencoders in fields such as matrix factorizations, model reduction, denoising, spectral clustering, and low-rank approximations to name a few.

In this work, we aim to investigate and initiate discussions on how tools from the numerical linear algebra community may provide fundamental and novel results for autoencoders, scientific machine learning, and beyond. We will discuss fundamental connections between matrix factorizations, classical inverse problems, and autoencoders in the field of signal compression and inverse problems. In the following, we provide details on the formulation of linear autoencoders through the Bayes risk formulation and the linear algebra involved in its analysis.

*Linear autoencoder.* Autoencoders are neural networks that learn to encode input data  $x$  into a compressed representation (latent representation) and then decode it back to reconstruct the original data  $x \in \mathbb{R}^n$ . Let us consider a linear autoencoder  $A \in \mathbb{R}^{n \times n}$ , where each element in  $A$  represents a trainable parameter. Assuming we have an  $\ell$ -dimensional *latent space* we may compute a generic optimal autoencoder by minimizing the *Bayes risk*, i.e.,

$$\min_{\text{rank}(A) \leq \ell} f(A) = \mathbb{E} \|(A - I)x\|_2^2, \quad (1)$$

given a distribution of the random variable  $x$  and where  $\mathbb{E}$  denotes the expectation and  $I$  the identity mapping. Assuming the random variable  $x$  has symmetric positive definite second moment  $\mathbb{E} xx^\top = \Gamma$  with Cholesky decomposition  $\Gamma = BB^\top$ , then

$$\mathbb{E} \|(A - I)x\|_2^2 = \text{tr}((A - I)\Gamma(A^\top - I)) = \|AB - B\|_F^2 \quad (2)$$

and (1) is equivalent to

$$\min_{\text{rank}(A) \leq \ell} \|AB - B\|_F^2. \quad (3)$$

For  $\ell = n$  the identity mapping  $A = I$  is an optimal solution. For rank constraint problems  $\ell < n$  an optimal low-rank solution can be found using the following generalization of the Eckart–Young–Mirsky theorem.

**Theorem 1.** Let matrix  $B \in \mathbb{R}^{n \times n}$  have full row rank with SVD given by  $B = U\Sigma V^\top$ . Then

$$\widehat{A} = U_\ell U_\ell^\top$$

is a solution to the minimization problem

$$\min_{\text{rank}(A) \leq \ell} \|AB - B\|_F^2,$$

having a minimal Frobenius norm  $\|\widehat{A}\|_F = \sqrt{\ell}$  and  $\|\widehat{A}B - B\|_F^2 = \sum_{k=\ell+1}^n \sigma_k(B)$ . This solution is unique if and only if either  $\ell = n$  or  $1 \leq \ell < n$  and  $\sigma_\ell(B) > \sigma_{\ell+1}(B)$ .

Following this result, the natural choice for the autoencoder  $\widehat{A}$  to be decomposed into an encoder and a decoder is  $\widehat{A} = \widehat{D}\widehat{E}$ , with encoder and decoder being  $\widehat{E} = U_\ell^\top$  and  $\widehat{D} = U_\ell$ , respectively. Note that this decomposition is not unique, e.g., let  $K$  be any  $n \times n$  invertible matrix then  $\widehat{E} = U_\ell^\top K$  and  $\widehat{D} = K^{-1}U_\ell$ , are valid choices.

*Sparse autoencoder.* While for small latent spaces  $\ell \ll n$  one obtains a low-rank approximation and a compressed approximation on the original signal  $x$ . However, compression can also be obtained utilizing a compressed sensing framework. Let us consider the problem of finding an optimal linear autoencoder  $A$  with the decomposition  $A = DE$  into encoder  $E \in \mathbb{R}^{\ell \times n}$  and  $D \in \mathbb{R}^{n \times \ell}$  where  $\ell > n$  by minimizing  $L^1$ -regularized optimization problem

$$\min_{D \in \mathbb{R}^{n \times \ell}, E \in \mathbb{R}^{\ell \times n}} \mathbb{E} \|(DE - I)x\|^2 + \lambda \|Ex\|_1 \quad (4)$$

with  $\lambda > 0$ . Autoencoders with  $\ell > n$  are referred to as overcomplete autoencoders. Such sparsity-promoting overcomplete autoencoders were first been introduced in the 2010s with pioneering work from various research groups but are not commonly utilized. The generalized lasso approach (4) may generate sparse vectors  $Ex$  while maintaining the same expected squared error as an undercomplete linear autoencoder where  $\ell < n$ .

*Numerical results.* We present our analytical findings and confirm them through numerical examples. We approach linear inverse problems using linear autoencoder approximations with theoretical guarantees. Here, we illustrate this with medical tomography, deblurring, and a classic heat equation. Furthermore, we analyze small angle scattering (SAS) data – a technique from material science to obtain information about the size, shape, and arrangement of material – via the proposed sparse autoencoder. We are able to obtain superior compression rates compared to state-of-the-art approaches.

# Fast Randomized Column Subset Selection Using Strong Rank-revealing QR

*Alice Cortinovis and Lexing Ying*

## Abstract

Many large-scale matrices arising in applications have a low numerical rank, and while the truncated singular value decomposition gives a way to construct the *best* low-rank approximation with respect to all unitarily invariant norms, this is often too expensive to compute. For this reason, different types of low-rank approximation strategies have been analyzed in the literature, for example, approximations constructed from some rows and columns of the matrix. In practice, the strategy for choosing rows and columns depends on the properties and the size of the matrix. Several deterministic and randomized strategies for selecting rows and columns for CUR approximation have been developed; see, e.g., [1] for an overview.

This talk is concerned with the analysis of a randomized algorithm that selects suitable rows and columns. The algorithm is based on an initial uniformly random selection of rows and columns, followed by a refinement of this choice using a strong rank-revealing QR factorization. We show bounds on the error of the corresponding low-rank approximation (more precisely, the CUR approximation error) when the matrix is a perturbation of a low-rank matrix that can be factorized into the product of matrices with suitable incoherence and/or sparsity assumptions. The talk is based on the paper [2].

## The column subset selection problem

Let  $A \in \mathbb{R}^{n \times n}$  be the matrix we want to approximate (the discussion easily generalizes to rectangular matrices). Let us denote by  $I, J \in \{1, \dots, n\}^\ell$  ordered index sets that correspond to rows and columns of  $A$ , respectively, for some  $\ell \ll n$ , and let us denote by  $A(I, :) \in \mathbb{R}^{\ell \times n}$  and  $A(:, J) \in \mathbb{R}^{n \times \ell}$  the submatrices of  $A$  corresponding to the rows indexed by  $I$  and the columns indexed by  $J$ , respectively. An approximation of  $A$  using these rows and columns has the form

$$A \approx A(:, J)MA(I, :),$$

for some matrix  $M \in \mathbb{R}^{\ell \times \ell}$ . The choice of  $M$  that minimizes the low-rank approximation error  $\|A - A(:, J)MA(I, :)\|_F$  in the Frobenius norm is the orthogonal projection  $M = A(:, J)^\dagger AA(I, :)^{\dagger}$ , where  $\dagger$  denotes the Moore-Penrose pseudoinverse of a matrix. The resulting approximation is usually called a “CUR approximation”.

The quality of the low-rank approximation, that is, the norm of the error matrix  $A - A(:, J)MA(I, :)$ , depends on the choice of rows and columns, and can be bounded, in the spectral norm, by

$$\|A - A(:, J)MA(I, :)\|_2 \leq \|A - A(:, J)A(:, J)^\dagger A\|_2 + \|A - AA(I, :)^{\dagger} A(I, :)\|_2, \quad (1)$$

where the two terms on the right-hand-side are the column and row subset selection error, respectively. For the remaining part of the talk, we focus on the problem of choosing columns, because the rows can be selected in the same way and the error of the corresponding CUR approximation is bounded as in (1).

## The proposed strategy

The simplest method to select columns is to choose some columns uniformly at random, which gives good low-rank approximations in many cases of interest. In [3], it was shown that if  $A$  is a rank- $k$  matrix that admits a low-rank decomposition with *incoherent* factors, uniform sampling of rows and columns allows to recover the matrix. Given a matrix  $X \in \mathbb{R}^{n \times k}$  with orthonormal columns, the coherence of  $X$  is defined as

$$\mu := n \max_{\substack{1 \leq i \leq n \\ 1 \leq j \leq k}} |x_{ij}|^2,$$

and we say that  $X$  is  $\mu$ -coherent. We say that a matrix is incoherent when  $\mu$  is small. The concept of incoherence informally means that the information about the matrix is “evenly spread out” across all rows and columns.

The favorable property of uniform sampling can be extended to matrices that have low *numerical* rank [4]. When the matrix  $A$  does not satisfy these incoherence assumptions, heuristic approaches were considered, e.g., in [5, 6], where the idea is to refine the choice of the uniform sampled columns using a rank-revealing decomposition. The algorithm that we consider is the following.

---

### Algorithm 1 Proposed algorithm for column subset selection

---

**Require:** Matrix  $A$ , number of indices  $\ell_0, \ell_a, \ell_b$

**Ensure:** Column index set  $J$  of cardinality  $\ell_a + \ell_b$

- 1: Select  $\ell_0$  rows of  $A$  uniformly at random (index set  $I_0$ )
  - 2: Select  $\ell_a$  columns of  $A(I_0, :)$  by sRRQR (index set  $J_a$ )
  - 3: Select another  $\ell_b$  columns of  $A$  uniformly at random (index set  $J_b$ )
  - 4: Return the column index set  $J = (J_a, J_b)$
- 

Here, sRRQR denotes the strong rank-revealing QR factorization [7]. Informally, this is a partial pivoted QR factorization that ensures that the first  $\ell_a$  columns of  $A(I_0, :)$  are a good approximation of the range of the columns of  $A(I_0, :)$ . A rank- $k$  sRRQR factorization for an  $m \times n$  matrix can be computed in time  $\mathcal{O}(mnk \log n)$ , therefore the algorithm runs in time  $\mathcal{O}(n\ell^2 \log n)$ , where  $\ell = \max\{\ell_0, \ell_a, \ell_b\}$ ; in particular, the cost is sublinear with respect to the size of the matrix.

## When is there hope for Algorithm 1 to work?

Let us look at a few illustrative examples to see when Algorithm 1 is likely to return a good column set for low-rank approximation purposes. For example, if  $A$  is a matrix of all ones (and thus has rank 1), uniformly sampling just one single column gives a vector that spans the range of  $A$ . The singular vectors of  $A$  are as incoherent as they could possibly be. Now consider, instead, a matrix  $B$  which is made of zeros except for one entry: in this case, neither uniform sampling nor Algorithm 1 will be able to correctly locate the only important column with high probability. The singular vectors of  $B$  have coherence  $n$ , the highest possible value.

There is some interesting middle ground in which uniform sampling alone is not good enough, but the combination with sRRQR gives us a good column subset. For example, consider the case of a rank-2 matrix  $C \in \mathbb{R}^{n \times n}$  that has entries  $c_{1j} = c_{j1} = 1$  for  $1 \leq j \leq n$  and zeros elsewhere. The row set  $I_0$ , chosen uniformly at random, will likely not include the first row. However, when looking at the matrix  $C(I_0, :)$ , the sRRQR algorithm will select a set  $J_a$  containing the first column, plus some other  $\ell_a - 1$  columns sampled uniformly at random. Now, the set  $J$  will contain the first column

and at least another column; therefore, it is enough to span the range of  $C$ . We can decompose

$$C = \begin{bmatrix} \frac{1}{\sqrt{n}} & 1 \\ \frac{1}{\sqrt{n}} & 0 \\ \frac{1}{\sqrt{n}} & 0 \\ \frac{1}{\sqrt{n}} & 0 \\ \vdots & \vdots \\ \frac{1}{\sqrt{n}} & 0 \end{bmatrix} \begin{bmatrix} \sqrt{n} & 0 \\ 0 & \sqrt{n-1} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & \frac{1}{\sqrt{n-1}} & \frac{1}{\sqrt{n-1}} & \cdots & \frac{1}{\sqrt{n-1}} \end{bmatrix} = XZY^T.$$

Note that, for each  $j = 1, 2$ , one between the  $j$ -th column of  $X$  and the  $j$ -th column of  $Y$  is sparse and the other one is incoherent. This example suggests that when a matrix has a rank- $k$  decomposition  $XZY^T$  (possibly, up to an additive error  $E$ ), there is hope for Algorithm 1 to work when, for each  $i = 1, \dots, k$ , one between the  $i$ -th columns of  $X$  and of  $Y$  is sparse, and the other is incoherent.

### Analysis of column quality

Our analysis considers the case in which  $A$  has rank exactly  $k$  and the case in which  $A$  is a small perturbation of the exact case. For simplicity, we state our results in the perturbed case, with slightly simplified assumptions, and we omit explicit constants; the precise results are in our paper [2].

**Assumptions.** We assume that  $A$  admits an approximate rank- $k$  factorization  $A = XZY^T + E$ , for some  $X \in \mathbb{R}^{n \times k}$  and  $Y \in \mathbb{R}^{n \times k}$ , where  $X$  and  $Y$  have orthonormal columns,  $Z \in \mathbb{R}^{k \times k}$  is diagonal, and the corresponding pairs of vectors of  $X$  and  $Y$  are either both incoherent ( $\mu$ -coherent with a small value of  $\mu$ ) or one is sparse and the other one is incoherent. Moreover, we assume that  $\|E\|_2 \leq \varepsilon$ .

**Main theorem.** If the assumptions hold and we take  $\ell_0, \ell_a, \ell_b$  to be a small multiple of  $\mu k$ , then the column index  $J$  returned by Algorithm 1 satisfies

$$\|A - A(:, J)A(:, J)^\dagger A\|_2 \leq \mathcal{O}\left(\varepsilon n \sqrt{\frac{k}{\ell}} \cdot \frac{\sigma_1(XZY^T)}{\sigma_k(XZY^T)}\right)$$

with high probability.

**Sketch of proof ingredients.** One important ingredient in the proof of our main result is the fact that selecting uniformly random rows from a matrix with orthonormal columns gives, with high probability, a well conditioned matrix [8]. The second ingredient is the sRRQR, which allows us to determine what are the most “important” columns in a given matrix (since this is used on a rectangular matrix which is much smaller than  $A$ , this is fast to do).

Intuitively, the columns corresponding to the index set  $J_a$  generated by lines 1 and 2 of Algorithm 1 are a good approximation to the part of  $A$  that corresponds to the pairs of vectors of  $X$  and  $Y$  that are of type (incoherent,incoherent) or (incoherent,sparse). The additional selection of  $\ell_b$  uniformly random columns in line 3 ensures that, with high probability, also the information from the pairs of vectors of  $X$  and  $Y$  of type (sparse,incoherent) is taken care of.

## Take-away messages and open questions

The analysis of Algorithm 1 shows that this combination of randomness and sRRQR is able to combine the speed of randomized algorithms with the reliability of sRRQR, for the matrices that admit a decomposition with the assumptions above. While it is difficult, in general, to check whether a matrix  $A$  admits a decomposition satisfying these assumptions, the objective of this talk is to shed some light on the excellent practical performance of simple sublinear-time algorithms for column and row subset selection. It is easier to think of  $XZY^T$  as the singular value decomposition of  $A$  or its best rank- $k$  approximation, but actually, we do not require  $X$  and  $Y$  to have orthonormal columns, as long as they are well-conditioned. This flexibility allows us to apply our bounds to a larger class of matrices.

Our results do not cover all the matrices for which *there is hope*. For example, a scenario that is not covered by the current theory and is left for future work consists of matrices that have some pairs of vectors of  $X$  and  $Y$  for which one of them is incoherent and the other one does not have any specific assumption (that is, it may be coherent but not sparse).

It is possible to formulate an iterative version of Algorithm 1, such as the one considered in [6], in which one, after line 3, again performs an sRRQR factorization, adds some uniformly sampled rows, and then repeats this procedure a couple of times alternating between the selection of rows and columns. While the practical benefits of this “iterative refinement” for many matrices have been well documented, a theoretical analysis is still lacking and is an interesting direction for future research.

## References

- [1] P.-G. Martinsson and J. Tropp, Randomized numerical linear algebra: Foundations and algorithms, *Acta Numerica*, 29 (2020), pp. 403–572.
- [2] A. Cortinovis and L. Ying, A sublinear-time randomized algorithm for column and row subset selection based on strong rank-revealing QR factorizations, *SIAM J. Matrix Anal. Appl. (to appear)*, 2024.
- [3] A. Talwalkar and A. Rostamizadeh, Matrix coherence and the Nyström method, in *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*, 2010, pp. 572–579.
- [4] J. Chiu and L. Demanet, Sublinear randomized algorithms for skeleton decompositions, *SIAM J. Matrix Anal. Appl.*, 34 (2013), pp. 1361–1383.
- [5] Y. Li, H. Yang, E. R. Martin, K. L. Ho, and L. Ying, Butterfly factorization, *Multiscale Model. Simul.*, 13 (2015), pp. 714–732.
- [6] J. Xia, Making the Nyström method highly accurate for low-rank approximations, *SIAM J. Sci. Comput.*, 46 (2024), pp. A1076–A1101.
- [7] M. Gu and S. C. Eisenstat, Efficient algorithms for computing a strong rank-revealing QR factorization, *SIAM J. Sci. Comput.*, 17 (1996), pp. 848–869.
- [8] J. A. Tropp, Improved analysis of the subsampled randomized Hadamard transform, *Adv. Adapt. Data Anal.*, 3 (2011), pp. 115–126.

# Rank-revealing QR factorizations: applications, algorithms, and theory

Anil Damle

## Abstract

Rank-revealing factorizations, e.g., [2, 7], have a long history in numerical linear algebra. We continue this story in multiple directions by discussing recent highlights of their development and use. This starts with a discussion about how pivoted QR factorizations play a central role in techniques for compressing modern, large-scale deep learning models [3, 5]. Motivated by that work we briefly highlight recent advances in computational methods for computing interpolative decompositions that leverage tools from randomized numerical linear algebra [1] and discuss associated theoretical developments that more clearly capture the behavior of low-rank matrix approximations derived from pivoted factorizations.

Modern deep learning models are often vastly overparametrized for their desired task; it is difficult to determine an optimal model size based on a description of the problem and/or training data. However, this has consequence as it leads to large models that are expensive to store and run inference on. We show that given a small amount of (potentially unlabeled) data we can compress a given model into one of smaller size that retains the same structure as the original model—it is just smaller. To illustrate this process we can consider a one-hidden layer neural network

$$f(x) = \sigma(x^T W)\alpha,$$

where  $x \in \mathbb{R}^d$  represents a data point,  $W \in \mathbb{R}^{d \times n}$  is the weight matrix,  $\alpha \in \mathbb{R}^n$  is a linear last layer, and  $\sigma$  is a non-linear function applied entrywise. Our task is to compute  $\widehat{W}^{d \times m}$  and  $\widehat{\alpha}^m$  with  $m < n$  such that  $f(x) \approx \sigma(x^T \widehat{W})\widehat{\alpha}$  to the desired accuracy and for all sensible  $x$ .

Given some small amount of data points, which we encode as the columns of  $X_C$ , we accomplish this goal by computing an interpolative decomposition [4] of  $Z = \sigma(X_C^T W)\alpha$  as

$$Z \approx Z(:, \mathcal{C})T,$$

where  $\mathcal{C}$  represents a subset of the columns of  $Z$ . Because the non-linear function is applied entrywise it commutes with subset selection and we have that

$$f(x) \approx \sigma(x^T W(:, \mathcal{C}))(T\alpha).$$

Letting  $\widehat{W} = W(:, \mathcal{C})$  and  $\widehat{\alpha} = T\alpha$  accomplishes our goal. This idea can be extended to multiple layers and more complicated layer types.

In the preceding use case, the matrices that we have to compute interpolative decompositions of can be quite large. However, the final quality of the process is not typically dependent on the exact subset of columns chosen—we just need a sufficiently good subset. This motivates the use of randomized algorithms to rapidly compute a suitable  $\mathcal{C}$ . Numerous algorithms exist for this task, and we provide a novel randomized version of the Golub-Klema-Stewart subset selection algorithm [6] that performs admirably in practice. In particular, we observe that its performance (and that of alternatives) depends on properties of singular vectors and we derive theoretical bounds that highlight this fact [1].

## References

- [1] R. ARMSTRONG, A. BUZALI, AND A. DAMLE, *Structure-aware analyses and algorithms for interpolative decompositions*, arXiv preprint arXiv:2310.09452, (2023).
- [2] S. CHANDRASEKARAN AND I. C. IPSEN, *On rank-revealing factorisations*, SIAM Journal on Matrix Analysis and Applications, 15 (1994), pp. 592–622.
- [3] J. CHEE, M. RENZ, A. DAMLE, AND C. D. SA, *Model preserving compression for neural networks*, in Advances in Neural Information Processing Systems, A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, eds., 2022.
- [4] H. CHENG, Z. GIMBUTAS, P.-G. MARTINSSON, AND V. ROKHLIN, *On the compression of low rank matrices*, SIAM Journal on Scientific Computing, 26 (2005), pp. 1389–1404.
- [5] M. FLYNN, A. WANG, D. E. ALVAREZ, C. DE SA, AND A. DAMLE, *STAT: Shrinking transformers after training*, arXiv preprint arXiv:2406.00061, (2024).
- [6] G. GOLUB, V. KLEMA, AND G.W. STEWART, *Rank degeneracy and least squares problems*, Stanford University department of Computer Science, (1976).
- [7] M. GU AND S. C. EISENSTAT, *Efficient algorithms for computing a strong rank-revealing QR factorization*, SIAM Journal on Scientific Computing, 17 (1996), pp. 848–869.

# On Minimizing Arithmetic and Communication Complexity of Jacobi’s Eigenvalue Method: Review and Beyond

*Yifu Wang, James Demmel, Hengrui Luo, Ryan Schneider*

## Abstract

Jacobi’s method iteratively computes the eigenvalues and eigenvectors of a symmetric matrix. Remarkably simple to implement, Jacobi’s method is a compelling candidate for use on large-scale applications. On the other hand, matrix multiplication is fundamental in numerical linear algebra, often regarded as a building block for other matrix computations.

With these in mind, we establish theoretical bounds on the asymptotic complexity of Jacobi’s method in both arithmetic and communication, aiming for efficiency comparable to matrix multiplication.

We not only analyze the complexity of sequential and parallel Jacobi using classical  $O(n^3)$  matrix multiplication, but also introduce recursive Jacobi’s methods that leverage Strassen-like  $O(n^{\omega_0})$  matrix multiplication to achieve optimal arithmetic and communication lower bounds. We also offer rigorous proofs of convergence for the recursive algorithms. The main contributions are as follows:

1. Starting from a dense real symmetric matrix  $\mathbf{A} \in \mathbf{M}_n(\mathbb{R})$  (without loss of generality, we only consider the real case), the **Classical Jacobi’s method** sequentially rotates all off-diagonal entries of  $\mathbf{A}$  in some given ordering. We denote one *sweep* as rotating through all off-diagonal entries of  $\mathbf{A}$  once. Since Classical Jacobi almost always converges, we assume that the algorithm converges in  $O(1)$  sweeps and the corresponding total arithmetic cost is  $O(1) \cdot \Theta(n^3) = \Theta(n^3)$ .

For estimating the lower bound on the communication cost, assume for now that we could only change the ordering of rotations. We denote the size of fast memory by  $M$ . Then when  $M^{1/2} < n < M$ , we can attain a lower bound of  $\Omega(n^4/M)$  reads and writes to slow memory, asymptotically exceeding the  $O(n^3/\sqrt{M})$  cost of classical matrix multiplication. To attain the cost of matrix multiplication requires more changes to the algorithm.

2. Allowing ourselves more freedom than just choosing the ordering of to-be-rotated entries, we next consider the **Block Jacobi’s method**, in which we rotate  $2b$ -by- $2b$  blocks instead of one off-diagonal entry each time. We still assume  $O(1)$  sweeps for the algorithm to converge and choose  $b$  to be able to fit three  $2b$ -by- $2b$  sub-matrices into the fast memory, i.e.  $b = \Theta(\sqrt{M})$ . In this case, the algorithm attains the communication lower bound  $\Omega(n^3/\sqrt{M})$  with  $O(n^3)$  matrix multiplication.
3. The highlight of this paper is the **Recursive Jacobi’s method** we introduce, along with a series of its variations. To the best of our knowledge, this is the first work which can asymptotically attain the arithmetic and communication costs of Strassen-like matrix multiplication, including a convergence proof.

We first propose a “vanilla” recursive algorithm, in which we apply a divide-and-conquer strategy, where the algorithm recursively partitions the input  $n$ -by- $n$  matrix into smaller  $2b$ -by- $2b$  blocks, until the size of the to-be-rotated sub-matrices reach a certain threshold, where  $b = n^f$  and  $0 < f < 1$  is the block parameter. We show that under the assumption that

the outermost sweep is executed  $O(1)$  times, the arithmetic complexity is  $F(n) = O(\frac{\log \log n}{-\log f} \cdot n^{3(1-f)+\omega_0 f})$ , which asymptotically approaches  $O(n^{\omega_0})$  as  $f$  approaches 1.

Convergence analysis for Jacobi's methods has been widely discussed, taking into account various pivoting strategies (such as rotation orderings and the choice between block and cyclic) as well as processing architectures (sequential or parallel). We refer readers to [7, 8, 10, 14] for further details. A key ingredient in [7] towards convergence of Classical Jacobi is to restrict the rotation angles of off-diagonal entries in a proper open subset of  $(-\frac{\pi}{2}, \frac{\pi}{2})$ . An analog for block Jacobi is uniformly bounded cosine transformations [6]. By reordering the columns of the orthogonal rotation matrix  $\mathbf{Q}$  via applying QR decomposition with column pivoting (QRCP for short) to the first-half leading rows of  $\mathbf{Q}$ , [6] successfully proved convergence for the block cyclic Jacobi. We leverage this idea and introduce our first variant of recursive Jacobi with convergence guarantee, the **Recursive Jacobi with QRCP**. With a slight trade-off between optimal arithmetic complexity lower bound and convergence guarantee, the recursive Jacobi with QRCP, within  $O(1)$  sweeps, can achieve

$$F(n) = \begin{cases} O(\frac{\log \log n}{-\log f} \cdot n^{3(1-f)+\omega_0 f}), & 0 < f \leq \frac{1}{4-\omega_0} \\ O(\frac{\log \log n}{-\log f} \cdot n^{2+f}), & \frac{1}{4-\omega_0} < f < 1 \end{cases}$$

due to the  $O(n^3)$  expensive complexity of QRCP.

The key of ensuring convergence in [6, 7] is to bound the cosines of rotation angles away from zero, which could also be done by applying LU decomposition with partial pivoting (LUPP for short). Unlike QRCP, LU decomposition can be implemented recursively with complexity of  $O(n^{\omega_0})$  [2, Section 4.2], and adding partial pivoting to the algorithm doesn't increase the arithmetic complexity. By applying LUPP to the transpose of the first-half leading columns of  $\mathbf{Q}$ , we introduce the **Recursive Jacobi with LUPP** which enjoys both optimal arithmetic complexity and convergence.

In the sequential case, for  $2 < \omega_0 \leq 3$ , the recursive Jacobi is shown to analogously get close to attaining the expected communication lower bound  $\Omega(n^{\omega_0}/M^{\omega_0/2-1})$  [13]. In practical terms, recursive Jacobi should be considered as a "galactic algorithm" since the size  $n$  where the algorithm shows benefits grows rapidly as  $f$  approaches 1.

4. In addition to the sequential cases, we also studied **parallel block Jacobi** with  $O(n^3)$  matrix multiplication, in which the algorithm simultaneously rotates off-diagonal blocks in different columns and rows [1, 9, 12]. We store the  $n$ -by- $n$  matrix  $\mathbf{A}$  on a  $\sqrt{P} \times \sqrt{P}$  grid of  $P$  processors, with block sizes  $b = n/\sqrt{P}$ , which we assume to be an integer for simplicity. Under this scenario, the arithmetic complexity is  $O(n^3/P)$ , which demonstrates the optimal linear speedup, and the communication complexity is  $O(n^2/\sqrt{P})$  words and  $O(\sqrt{P} \log P)$  messages, which attains the communication lower bound (except for the  $\log P$  factor) for classical matrix multiplication using the minimum amount of memory.

One remark is that the above studies and estimates readily extend to the SVD due to its strong connection with Jacobi's method [4, 5]. Furthermore, by not restricting ourselves to Jacobi-like methods, our recursive algorithm technique can also benefit non-Jacobi methods, for example combined with QDWH (QR-based dynamically weighted Halley algorithm) [11].

Additionally, since all our recursive algorithms follow a divide-and-conquer paradigm utilizing  $O(n^{\omega_0})$  matrix multiplication, it follows from the analysis in [2, 3] that all the proposed algorithms are backward stable.

In conclusion:

1. We have demonstrated an asymptotic approach to make the Jacobi's eigenvalue method and SVD nearly as fast as matrix multiplication, in terms of both arithmetic and communication complexity, across several scenarios. For  $O(n^3)$  matrix multiplication, we analyzed both sequential and parallel Jacobi's methods.

*A remaining open question is whether the (better) lower bound and communication complexity for matrix multiplication using more than the minimum memory is attainable for Jacobi.*

2. For  $O(n^{\omega_0})$  matrix multiplication, we introduced a series of recursive Jacobi's methods, focusing on minimizing arithmetic cost while also ensuring the convergence of the proposed algorithms.

*Another remaining open question is whether these asymptotically faster recursive Jacobi's methods can be parallelized and attain both the arithmetic and communication complexity lower bounds of matrix multiplication.*

## References

- [1] M. Berry and A. Sameh. An overview of parallel algorithms for the singular value and symmetric eigenvalue problems. *J. Comp. Appl. Math.*, 27:191–213, 1989.
- [2] James Demmel, Ioana Dumitriu, and Olga Holtz. Fast linear algebra is stable. *Numerische Mathematik*, 108(1):59–91, 2007.
- [3] James Demmel, Ioana Dumitriu, Olga Holtz, and Robert D. Kleinberg. Fast matrix multiplication is stable. *Numerische Mathematik*, 106:199–224, 2006.
- [4] K. Drmač and K. Veselić. New fast and accurate Jacobi SVD algorithm, I. *SIAM J. Mat. Anal. Appl.*, 29(4):1322–1342, 2008.
- [5] K. Drmač and K. Veselić. New fast and accurate Jacobi SVD algorithm, II. *SIAM J. Mat. Anal. Appl.*, 29(4):1343–1362, 2008.
- [6] Zlatko Drmač. A Global Convergence Proof for Cyclic Jacobi Methods with Block Rotations. *SIAM Journal on Matrix Analysis and Applications*, 31(3):1329–1350, 2010.
- [7] G. E. Forsythe and P. Henrici. The Cyclic Jacobi Method for Computing the Principal Values of a Complex Matrix. *Transactions of the American Mathematical Society*, 94(1):1–23, 1960.
- [8] V. Hari. Convergence to diagonal form of block Jacobi-type methods. *Numer. Math.*, 129:449–481, 2015.
- [9] Franklin T. Luk and Haesun Park. A Proof of Convergence for Two Parallel Jacobi SVD Algorithms. *IEEE Trans. Computers*, 38:806–811, 1989.
- [10] W. Mascarenhas. Convergence of the Jacobi method for arbitrary orderings. *SIAM J. Mat. Anal. Appl.*, 16(4):1197–1209, Oct 1995.
- [11] Yuji Nakatsukasa and Nicholas J. Higham. Stable and Efficient Spectral Divide and Conquer Algorithms for the Symmetric Eigenvalue Decomposition and the SVD. *SIAM Journal on Scientific Computing*, 35(3):A1325–A1349, 2013.

- [12] A. Sameh. On Jacobi and Jacobi-like algorithms for parallel computers. *Math. Comp.*, 25(115):579–590, July 1971.
- [13] Jacob Scott. *An I/O-Complexity Lower Bound for All Recursive Matrix Multiplication Algorithms by Path-Routing*. PhD thesis, UC Berkeley Mathematics PhD thesis, 2015.
- [14] G. Shroff and R. Schreiber. On the convergence of the cyclic Jacobi method for parallel block orderings. *SIAM J. Mat. Anal. Appl.*, 10(3):326–346, 1989 1989.

# Randomized Algorithms for Solving Linear Systems with Low-rank Structure

*Michał Dereziński, Daniel LeJeune, Christopher Musco,  
Deanna Needell, Elizaveta Rebrova, and Jiaming Yang*

## Abstract

We consider the task of solving a large system of linear equations  $Ax = b$ , where for simplicity, we will assume that  $A$  is real, square, and full-rank. Iterative algorithms, such as LSQR, Conjugate Gradient and other Krylov subspace methods, are a powerful tool for solving such linear systems. Yet, the convergence properties of these methods are highly dependent on the singular value structure of the matrix  $A$ , and characterizing these properties effectively requires going beyond the usual notion of condition number. In this talk, we will consider this problem in the context of linear systems whose singular values exhibit a low-rank structure, in the sense that  $A$  can be decomposed into a low-rank ill-conditioned matrix (the “signal”) and a full-rank well-conditioned matrix (the “noise”). Such linear systems are motivated by a range of problem settings, including in statistics, machine learning, and optimization, where the “signal” is often low-rank due to inherent structure of the data, while the “noise” may be coming from measurement error, data transformations, or an explicit regularizer imposed by the user. We will show how randomized sketching techniques, including our recent works on randomized preconditioning [DMY25] and stochastic solvers [DR24, DY24, DLNR24], can be used to exploit this low-rank structure in order to accelerate linear system solving in ways that go beyond what is possible with Krylov subspace methods.

**Linear systems with low-rank structure.** Consider the following linear system task:

$$\text{Solve } Ax = b, \quad \text{given } A \in \mathbb{R}^{n \times n} \quad \text{and } b \in \mathbb{R}^n,$$

where  $A$  is a full-rank matrix with singular values  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n$ . For a given low-rank parameter  $k \in \{1, \dots, n\}$ , we will allow the top- $k$  part of the singular values to be very ill-conditioned, but assume that the tail is moderately well-conditioned, as measured by  $\kappa_k = \sigma_{k+1}/\sigma_n$ . For example, if the matrix  $A$  is explicitly regularized, e.g.,  $A = B + \lambda I_n$  as in ridge regression [AM15] or cubic-regularized Newton’s method [NP06], then  $k$  may correspond to the number of singular values above the  $\lambda$  threshold. Similar regularization effect occurs when  $A$  is distorted by isotropic noise,  $A = B + \delta N$ , e.g., where  $N$  is Subgaussian [Joh01], or it is the error from stochastic rounding [DBM<sup>+</sup>24]. Also,  $A$  may exhibit a power law singular value distribution ( $\sigma_i \propto i^{-\beta}$ ), e.g., due to a data transformation with the Matérn kernel function [RW06]. Here, different values of  $k$  capture different signal-to-noise trade-offs. Our goal is to describe the convergence and computational cost of iterative algorithms for solving  $Ax = b$  in terms of the parameters  $n$ ,  $k$ , and  $\kappa_k$ . One can also consider the sparsity of  $A$ , but for simplicity, we will focus on the dense setting.

**Effectiveness and limitations of Krylov subspace methods.** A careful analysis of Krylov subspace methods such as LSQR and CG for solving linear systems with low-rank structure [AL86] shows that they need  $k$  iterations to capture the top- $k$  singular vectors, and then  $O(\kappa_k \log(1/\epsilon))$  iterations to converge at a rate that depends only on  $\kappa_k$  (with  $\kappa_k$  replaced by  $\sqrt{\kappa_k}$  when  $A$  is positive definite). Thus, for a dense  $A$ , before reaching a fast convergence rate of  $O(n^2 \kappa_k \log 1/\epsilon)$  operations, Krylov methods require an initial  $O(n^2 k)$  cost (corresponding to roughly  $k$  matrix-vector products) to capture the low-rank structure of  $A$ , which is expensive for large  $k$ . This  $n^2 k$  bottleneck can be established as a lower bound not just for Krylov methods but for any algorithms that access  $A$  only through matrix-vector products [DLNR24].

Given the above problem formulation and discussion, the central question of this talk is:

*Can the  $n^2k$  bottleneck in solving linear systems with low-rank structure be overcome,  
when given direct access to  $A$  and allowing randomization?*

**Randomized preconditioning via sparse sketching.** Randomized sketching offers a powerful set of tools for accelerating linear solvers. While these approaches have traditionally focused on very tall least squares problems [AMT10], linear systems with low-rank structure offer another setting where sketching can be beneficial. Such an algorithm starts by applying a random matrix  $S \in \mathbb{R}^{s \times n}$  (e.g., Gaussian) to the matrix  $A$ , producing a smaller sketch  $\tilde{A} = SA \in \mathbb{R}^{s \times n}$ , where  $s \ll n$  is the sketch size. This sketch can now be used to construct an approximate low-rank decomposition of  $A$ , e.g., by orthonormalizing the columns of  $\tilde{A}^\top$  to obtain an  $n \times s$  matrix  $Q$  and projecting  $A$  onto the subspace defined by those columns,  $\hat{A} = AQQ^\top \approx A$  [HMT11]. The intuition here is that  $\hat{A}$  approximates  $A$  reasonably well in the top- $k$  singular directions as long as the sketch size  $s$  is sufficiently larger than  $k$ , and this approximation can be further boosted via subspace iteration.

If implemented naively, sketching does not appear to overcome the  $O(n^2k)$  computational barrier exhibited by Krylov methods, due to three bottlenecks: (1) applying the sketching matrix  $S$ , (2) projecting via the orthogonal matrix  $Q$ , and (3) performing subspace iteration. Each of these require at least  $k$  matrix-vector products to produce a decent preconditioner for a linear system with rank  $k$  structure. However, given direct access to  $A$ , the sketching cost (bottleneck 1) can be reduced by using fast sketching methods, e.g., by making  $S$  extremely sparse, which is known to retain similar guarantees as a Gaussian matrix. Moreover, recent works have shown that a careful construction of the preconditioner can avoid the full projection step (bottleneck 2): in the positive definite case, by relying on Nyström approximations [FTU23], and in the general case, by using an inner solver to construct the preconditioner implicitly [DMY25]. In the latter work, we showed that this approach can be used to solve a linear system in  $\tilde{O}(n^2\kappa_k \sqrt{n/k} \log 1/\epsilon + k^3)$  operations (up to minor logarithmic factors), where the term  $\sqrt{n/k}$  comes as a trade-off from omitting subspace iteration (bottleneck 3). When  $k$  is sufficiently large and  $\kappa_k$  small enough, this overcomes the  $n^2k$  barrier.

**Stochastic solvers via Sketch-and-Project.** Another class of methods that use randomized sketching and/or sub-sampling to go beyond matrix-vector products are stochastic iterative solvers such as randomized Kaczmarz and coordinate descent, among others. Viewed in the context of sketching, many of these methods can be unified under the framework of Sketch-and-Project [GR15]. Here, we consider a solver that updates an iterate  $x_t$  by repeatedly sketching the system  $Ax = b$  and projecting  $x_t$  onto the solutions of the sketched system:

$$x_{t+1} = \arg \min_{x \in \mathbb{R}^n} \|x_t - x\| \quad \text{subject to} \quad SAx = Sb.$$

While stochastic solvers have traditionally been considered effective primarily in specialized settings where we may not be able to perform full matrix-vector products with  $A$  (e.g., due to memory or bandwidth constraints), we have shown in recent works that these methods can also be particularly effective for linear systems with low-rank structure. Here, the intuition is that the sketched system  $SAx = Sb$  retains the information about the top- $k$  singular directions of  $A$ , which gives the Sketch-and-Project solver a convergence rate akin to being preconditioned with a rank  $k$  approximation [DR24]. We have adapted this approach to a simple Randomized Block Kaczmarz method [DY24], as well as a variant with Nesterov's acceleration [DLNR24], showing that these algorithms can solve a linear system in  $\tilde{O}((n^2 + nk^2)\kappa_k \log 1/\epsilon)$  operations. This recovers the fast Krylov convergence of  $\tilde{O}(n^2\kappa_k \log 1/\epsilon)$  operations for up to  $k = O(\sqrt{n})$ , while entirely avoiding the  $n^2k$  bottleneck.

## References

- [AL86] Owe Axelsson and Gunhild Lindskog. On the rate of convergence of the preconditioned conjugate gradient method. *Numerische Mathematik*, 48:499–523, 1986.
- [AM15] Ahmed El Alaoui and Michael W. Mahoney. Fast randomized kernel ridge regression with statistical guarantees. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, pages 775–783, 2015.
- [AMT10] Haim Avron, Petar Maymounkov, and Sivan Toledo. Blendenpik: Supercharging lapack’s least-squares solver. *SIAM Journal on Scientific Computing*, 32(3):1217–1236, 2010.
- [DBM<sup>+</sup>24] Gregory Dexter, Christos Boutsikas, Linkai Ma, Ilse CF Ipsen, and Petros Drineas. Stochastic rounding implicitly regularizes tall-and-thin matrices. *arXiv preprint arXiv:2403.12278*, 2024.
- [DLNR24] Michał Dereziński, Daniel LeJeune, Deanna Needell, and Elizaveta Rebrova. Fine-grained analysis and faster algorithms for iteratively solving linear systems. *arXiv preprint arXiv:2405.05818*, 2024.
- [DMY25] Michał Dereziński, Christopher Musco, and Jiaming Yang. Faster linear systems and matrix norm approximation via multi-level sketched preconditioning. *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2025.
- [DR24] Michał Dereziński and Elizaveta Rebrova. Sharp analysis of sketch-and-project methods via a connection to randomized singular value decomposition. *SIAM Journal on Mathematics of Data Science*, 6(1):127–153, 2024.
- [DY24] Michał Dereziński and Jiaming Yang. Solving dense linear systems faster than via preconditioning. In *56th Annual ACM Symposium on Theory of Computing*, 2024.
- [FTU23] Zachary Frangella, Joel A Tropp, and Madeleine Udell. Randomized Nyström preconditioning. *SIAM Journal on Matrix Analysis and Applications*, 44(2):718–752, 2023.
- [GR15] Robert M Gower and Peter Richtárik. Randomized iterative methods for linear systems. *SIAM Journal on Matrix Analysis and Applications*, 36(4):1660–1690, 2015.
- [HMT11] Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011.
- [Joh01] Iain M Johnstone. On the distribution of the largest eigenvalue in principal components analysis. *The Annals of Statistics*, 29(2):295–327, 2001.
- [NP06] Yurii Nesterov and Boris T Polyak. Cubic regularization of newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.
- [RW06] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

# Sketched GCRODR and its Convergence Analysis

Eric de Sturler and Fei Xue

## Abstract

We develop a sketched version of the GCRODR algorithm for the solution of a sequence of linear systems. The recycling approach in GCRODR with an approximate invariant subspace allows us to derive upperbounds on the convergence of GCRODR based on the field of values of the projected system (see below). We extend this convergence result to upperbounds on the convergence of a sketched GCRODR (S-GCRODR). The bounds for S-GCRODR deteriorate from those for GCRODR as a function of the subspace embedding distortion  $\epsilon$ , and we provide expressions for this relation.

Sketching offers the opportunity to substantially reduce the high orthogonalization cost in long-recurrence solvers like GMRES. Several approaches have been explored. Balabanov and Grigori [1] replace the inner products in the orthogonalization by sketched inner products, replacing an orthogonal projection by a oblique projection (but typically close to orthogonal), which maintains the stability of the Arnoldi process. While they demonstrate good performance improvements on HPC architectures, the approach does not reduce the computational complexity,  $O(nm^2)$  for  $m$  iterations with  $A \in \mathbb{C}^{n \times n}$ . On the other hand, Nakatsukasa and Tropp [5] generate the Krylov space with truncated Arnoldi and use sketching for the LS solution of the resulting, potentially very ill-conditioned, system. This approach has the significant advantage that it drastically reduces the computational complexity to  $O(nm \log m)$  for  $m$  iterations. However, the severe ill-conditioning of the basis vectors typically leads to some deterioration of the convergence. We propose an efficient and convergence-wise effective combination of the two approaches.

We consider the solution of a sequence of linear systems,  $A^{(j)}x^{(j)} = b^{(j)}$ , where  $A \in \mathbb{C}^{n \times n}$ , and where the matrices change slowly. We aim for robustness and reduced iterations as well as a significant reduction in the average cost per iteration. Recycling Krylov subspaces from previous linear solves can drastically reduce the total number of iterations, which suggests that the approximate orthogonalization by sketching of new Krylov vectors against the recycle space is important. This introduces only a linear cost in the number of iterations. In addition, we substantially reduce the computational complexity by using only selective orthogonalization with a fixed number of orthogonalizations when we extend the (augmented) Krylov search space and solve the least squares problem in a sketched fashion following the approach proposed in [5].

**GCRODR and S-GCRODR** Consider a recycle space of dimension  $k$ , defined by (range)  $R(U)$ , where  $U \in \mathbb{C}^{n \times k}$  such that (for convenience)  $C = AU$  has orthonormal columns,  $C^*C = I$ . We define the (orthogonal) projection  $\Phi = CC^*$ . We also define  $C_\perp$  such that the matrix  $[C \ C_\perp] \in \mathbb{C}^{n \times n}$  is unitary. We assume here, for simplicity, that  $U$  has been selected such that  $R(U)$  is a low accuracy approximation (see below) to an invariant subspace with eigenvalues near the origin. As shown in [7], using the recycle space  $R(U)$ , we can update the initial solution,  $\tilde{x}_0$ , and residual,  $\tilde{r}_0$ , as  $x_0 = \tilde{x}_0 + UC^*\tilde{r}_0$ , and  $r_0 = (I - \Phi)\tilde{r}_0$ , and subsequently solve the *projected system*  $(I - \Phi)Az = (I - \Phi)\tilde{r}_0$  with GMRES. For this (consistent) system, the right hand side  $(I - \Phi)\tilde{r}_0 \in R(C)^\perp = R(C_\perp)$  and  $(I - \Phi)Az : R(C_\perp) \rightarrow R(C_\perp)$ . So, we can analyze the convergence for GMRES for the linear operator  $(I - \Phi)A$  over the space  $R(C_\perp)$ .

After defining a sketching matrix  $S \in \mathbb{C}^{s \times n}$ , which provides an  $\ell_2$  embedding of a suitable vector space  $\mathcal{V}$ , which contains the right hand side or residual, the  $R(C)$ , and a suitable Krylov space, we

let  $SC = YR_Y$  and  $S^*Y = QR_Q$  be reduced QR decompositions. In S-GCRODR the orthogonal projection  $I - \Phi$  in GCRODR is replaced by the (oblique) projection  $I - \widehat{\Phi}$ , with range  $R(\widehat{\Phi}) = R(C)$  and null space  $N(\widehat{\Phi}) = R(Q)^\perp = R(Q_\perp)$ , where  $[Q Q_\perp]$  is a unitary matrix. This implies that  $(I - \widehat{\Phi}) = Q_\perp(C_\perp^* Q_\perp)^{-1} C_\perp^*$ . After computing the updates to the initial guess and residual, S-GCRODR solves the *projected system*  $(I - \widehat{\Phi})Az = (I - \widehat{\Phi})\tilde{r}_0$  using GMRES.

**Convergence** We give bounds on the convergence for GCRODR while recycling an approximate invariant subspace and compare these with convergence bounds for the sketched version, S-GCRODR, recycling the same invariant subspace. We show that the convergence bounds for S-GCRODR can deteriorate due the oblique projection; however, the deterioration can be bounded in terms of the embedding subspace distortion  $\epsilon$ .

We can analyze the convergence of GCRODR by considering convergence bounds for GMRES for the linear operator  $(I - \Phi)A$  restricted to the space  $R(C_\perp)$ , which can be derived using the field of values (FOV) [4, 3] of  $C_\perp^* AC_\perp$ . Now let  $A$  have the block Schur decomposition (with unitary  $[V V_\perp] \in \mathbb{C}^{n \times n}$ )

$$A = [V V_\perp] \begin{bmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{bmatrix} [V V_\perp]^*, \quad (1)$$

where the eigenvalues of  $T_{11}$  are near the origin (possibly surrounding the origin) and  $\|T_{11}\|_2$  is small, and the eigenvalues of  $T_{22}$  are further away in the right half plane, and let  $\|(I - \Phi)V\|_2 = \delta < 1$ . In the derivation of FOV bounds, we use the following notation. We use calligraphic script to denote sets:  $\mathcal{F}(T_{11})$  denotes the field of values of  $T_{11}$ ,  $\mathcal{F}(T_{22})$  denotes the field of values of  $T_{22}$  and  $\mathcal{D}$  denotes the unit disk. Set addition is defined in the usual way, and for a scalar  $\tau$  and set  $\mathcal{S}$ , the set  $\tau\mathcal{S}$  is defined as  $\tau\mathcal{S} = \{\tau x \mid x \in \mathcal{S}\}$  and  $[\tau_1, \tau_2]\mathcal{S} = \{\tau x \mid x \in \mathcal{S} \text{ and } \tau \in [\tau_1, \tau_2]\}$ . We can then bound the FOV of the linear operator  $(I - \Phi)A$  restricted to the space  $R(C_\perp)$ ,  $\mathcal{F}(C_\perp^* AC_\perp)$  as

$$\mathcal{F}(C_\perp^* AC_\perp) \subset [1 - \delta^2, 1]\mathcal{F}(T_{22}) + [0, \delta^2]\mathcal{F}(T_{11}) + \delta(1 - \delta^2)^{1/2}\|T_{12}\|_2\mathcal{D}. \quad (2)$$

This equation shows that even for  $\delta$  not very small, say  $\delta = 10^{-2}$  (which can be achieved with modest effort [6]),  $\mathcal{F}(C_\perp^* AC_\perp)$  is only slightly larger than  $\mathcal{F}(T_{22})$ , unless  $\|T_{12}\|_2$  is (relatively) large. We can now bound the convergence of GCRODR for  $A$  with the recycle space  $R(U)$  using the FOV convergence bounds for GMRES with the FOV bounds from (2).

We can bound the convergence of S-GCRODR in a similar fashion as for GCRODR using bounds on the FOV of  $(I - \widehat{\Phi})A$  restricted to the space  $R(Q_\perp)$ , that is the set  $\{z^*(I - \widehat{\Phi})Az : z = Q_\perp\zeta, \zeta \in \mathbb{C}^{n-k}, \|\zeta\|_2 = 1\}$ , which is also given by  $\mathcal{F}(Q_\perp^* Q_\perp (C_\perp^* Q_\perp)^{-1} C_\perp^* AQ_\perp) = \mathcal{F}((C_\perp^* Q_\perp)^{-1} C_\perp^* AQ_\perp)$ .

To understand the relation between the FOV bounds for GCRODR and S-GCRODR, we consider the singular values of  $C_\perp^* Q_\perp$ . We assume  $k \ll n$ . Let  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_k$  be the singular values of  $C^* Q$ . Then we can derive the singular values of  $C_\perp^* Q_\perp$  from the CS-decomposition of  $[C C_\perp]^*[Q Q_\perp]$ :  $\sigma(C_\perp^* Q_\perp) \in \{1, \lambda_1, \dots, \lambda_k\}$ . Furthermore, we can prove, based on the  $\epsilon$ -embedding, that  $\lambda_k \geq \sqrt{(1 - \epsilon)/(1 + \epsilon)}$ , and therefore, for  $\epsilon \rightarrow 0$ ,  $R(Q_\perp) \rightarrow R(C_\perp)$ . This in turn implies that  $(C_\perp^* Q_\perp)^{-1} C_\perp^* AQ_\perp \rightarrow C_\perp^* AC_\perp$ , and hence the FOVs that govern the convergence bounds for GCRODR and S-GCRODR get closer and closer as  $\epsilon$  becomes small.

We can describe the dependence of  $\mathcal{F}((C_\perp^* Q_\perp)^{-1} C_\perp^* AQ_\perp)$  on  $\epsilon$  in substantial detail by deriving detailed expressions of the type (for unit vectors  $\zeta$ )

$$\zeta^* Q_\perp^* Q_\perp (C_\perp^* Q_\perp)^{-1} C_\perp^* AQ_\perp \zeta = \begin{pmatrix} \eta_1(\epsilon) \\ \eta_2(\epsilon) \end{pmatrix}^* \begin{pmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{pmatrix} \begin{pmatrix} \eta_1(\epsilon) \\ \eta_2(\epsilon) \end{pmatrix}, \quad (3)$$

and analyze how close bounds for (3) are to (2) as a function of  $\epsilon$ . These bounds clarify how the convergence of S-GCRODR may deteriorate as a function of the distortion parameter  $\epsilon$  as a consequence of how far the oblique projection  $I - \hat{\Phi}$  deviates from the orthogonal projection  $I - \Phi$ .

**A Numerical Experiment** We present one set of numerical results to compare several sketched variants of GMRES and GCRODR. The results are derived from solving the following nonlinear Helmholtz equation on the 2D domain  $\Omega = (0, 1) \times (0, 1)$ ,

$$\begin{cases} \Delta u + \kappa^2(1 + \epsilon|u|^2)u = 0, \\ u_x + i\kappa u = 2i\kappa, \quad \text{at } x = 0, y \in (0, 1) \\ u_x - i\kappa u = -i\kappa, \quad \text{at } x = 1, y \in (0, 1) \\ \text{periodic boundary condition at } y = 0, 1, x \in (0, 1), \end{cases} \quad (4)$$

using Anderson acceleration (AA). We take  $\epsilon = 0.40$  and  $\kappa = 12$ . We discretize  $\Omega$  using a uniform mesh with  $n + 1$  equispaced nodes in the  $x$  and in the  $y$  directions, respectively. We also use the standard 2nd order finite difference to approximate the Laplacian operator, and use ghost nodes at the left ( $x = 0$ ) and the right ( $x = 1$ ) boundaries. We let  $n = 512$  so that the number of elements in the  $u$  vectors is  $n(n+1) = 262656$ . To set up the corresponding nonlinear system, define  $I_x = I_{n+1}$ ,  $I_y = I_n$ ,  $D_{2x} = \text{tridiag}(\mathbf{1}, -\mathbf{2}, \mathbf{1}) \in \mathbb{R}^{(n+1) \times (n+1)}$ , except that  $D_{2x}(1, 1) = D_{2x}(n+1, n+1) = 2(-1 + i\kappa h)$ , and  $D_{2x}(1, 2) = D_{2x}(n+1, n) = 2$ ,  $D_{2y} = \text{tridiag}(\mathbf{1}, -\mathbf{2}, \mathbf{1}) \in \mathbb{R}^{n \times n}$ , except that  $D_{2y}(1, n) = D_{2y}(n, 1) = 1$ ,  $F(u) = \frac{1}{h^2} (I_y \otimes D_{2x} + D_{2y} \otimes I_x) + \kappa^2 \text{diag}(1 + \epsilon|u|^2) - f_{bdy}$ , where  $f_{bdy} = \mathbf{1}_n \otimes [\frac{2(2)i\kappa}{h}; \mathbf{0}_{n-1}; \frac{2(-1)i\kappa}{h}]$ , so that the nonlinear system is  $F(u)u = 0$ . To define the Picard iteration, we let the squared term of  $u$  be the current iterate  $u^{(k)}$  and solve for the next iterate  $u^{(k+1)}$ . That is, at each step we solve the linear system

$$\left( \frac{1}{h^2} (I_y \otimes D_{2x} + D_{2y} \otimes I_x) + \kappa^2 \text{diag}(1 + \epsilon|u^{(k)}|^2) \right) u^{(k+1)} = f_{bdy} \quad (5)$$

for  $u^{(k+1)}$ . The initial vector  $u^{(0)}$  is the vectorization of  $u(x, y) = e^{i(2\pi y + \kappa x)}$  on the mesh. To set up AA, we let the damping parameter be 1, the optimization involve all previous iterates  $u^{(k)}$ , and the iteration is terminated when  $\|u^{(k+1)} - u^{(k)}\|_\infty \leq 10^{-6}$ .

At each step of AA, the linear system (5) is solved by the following methods, and a new ILUTP preconditioner is constructed using approximate minimum degree ordering and drop tolerance 0.002. We compare GMRES(120), the sketched version S-GMRES(120) as proposed in [5], (standard) GCRODR(120,20) and the sketched version S-GCRODR(120,20) discussed above, and two versions of the method GMRES-SDR(120,20) proposed in the recent paper [2], where the authors combine a sketched version of GMRES with deflated restarting. This approach differs from S-GCRODR in that the authors apply the deflated restarting by augmenting the search space with the deflation vectors, using truncated/selective orthogonalization when generating new Krylov search directions, and then using sketching to solve the least squares problem over both deflation and new Krylov vectors. In this approach, the new Krylov space that extends the solution search space is not generated (approximately) orthogonal to (the image under  $A$  of) the recycle space. This may lead to less effective search spaces and hence a reduced convergence rate. On the other hand, for the same total search space dimension in a cycle, it leads to a further reduction in complexity compared with the method we propose.

In Table 1, for each linear solver, we give the total and average runtime and number of preconditioned matrix-vector products for solving the sequence of linear systems (5) arising from Anderson

Table 1: Total and average numbers of preconditioned matrix-vector products and runtimes for several methods solving the sequence of linear systems (5) arising from Anderson acceleration for a nonlinear Helmholtz equation.

	GMRES(120)	GCRO-DR(120,20)	S-GMRES(120)	GCRO-DR(120,20)	(m) GMRES-SDR(120,20)	(s) GMRES-SDR(120,20)
matvecs	9014 (361)	2794 (121)	10319 (382)	2819 (123)	10144 (423)	6037 (232)
time (secs)	1374.3 (55.0)	343.8 (14.9)	691.3 (25.6)	183.2 (8.0)	622.5 (25.9)	393.4 (15.1)

acceleration for the nonlinear system (4). For all linear solvers, we let the maximum dimension of the subspace be  $m = 120$  and let the recycle space dimension be  $k = 20$ . Due to the irregular convergence behavior of Anderson acceleration with the linear systems (5) solved *approximately*, it takes Anderson acceleration a slightly different number of steps to satisfy the stopping criterion  $\|u^{(k+1)} - u^{(k)}\|_\infty \leq 10^{-6}$  for each solver. AA based on GCRO-DR and S-GCRO-DR takes 23 (the fewest) steps, whereas AA based on S-GMRES takes 27 (the most) steps to converge. In the column ‘(m) GMRES-SDR(120,20)’ we report results when GMRES-SDR recycles search spaces from one linear system to the next, whereas under the column ‘(s) GMRES-SDR(120,20)’ we report results with GMRES-SDR starting each linear system without a recycle space (which seems to work better). Finally, we note that, while for this system S-GCRODR is the clear winner, by a large margin, in terms of the runtime, for other test problems GMRES-SDR was competitive and sometimes faster.

## References

- [1] O. BALABANOV AND L. GRIGORI, *Randomized Gram–Schmidt process with application to GMRES*, SIAM J. Sci. Comput., 44 (2022), pp. A1450–A1474.
- [2] L. BURKE, S. GÜTTEL, AND K. SOODHALTER, *GMRES with randomized sketching and deflated restarting*, arXiv:2311.14206, arXiv, 2023.
- [3] M. EMBREE, *Extending Elman’s Bound for GMRES*, arXiv:2312.15022v1, arXiv, 2023.
- [4] A. GREENBAUM, *Iterative methods for solving linear systems*, SIAM 1997.
- [5] Y. NAKATSUKASA AND J. A. TROPP, *Fast and accurate randomized algorithms for linear systems and eigenvalue problems*, SIAM J. Matrix Anal. Appl., 45 (2024), pp. 1183–1214.
- [6] M. L. PARKS, E. DE STURLER, G. MACKEY, D. D. JOHNSON, AND S. MAITI, *Recycling Krylov subspaces for sequences of linear systems*, SIAM J. Sci. Comput. 28 (2006), pp. 1651–1674.
- [7] K.M. SOODHALTER, E. DE STURLER, AND M. E. KILMER, *A survey of subspace recycling iterative methods*, GAMM-Mitteilungen 43 (2020), p. e202000016.

# CASPR: Combining Axis Preconditioners using Kronecker Sums/Products for Training Large Neural Networks

*Inderjit S. Dhillon, Sai S. Duvvuri*

## Abstract

Deep Neural Networks (DNNs) have transformed fields like computer vision, natural language processing, and scientific research by enabling systems to learn complex patterns, make high-level predictions, and analyze large data sets. DNNs have driven advancements in material sciences, chemistry, and physics, significantly aiding scientific discovery. However, they are difficult to optimize due to their large parameter spaces and can require extensive computational resources, and thus effectively training DNNs is a contemporary challenge.

Most DNNs, including Large Language Models, are trained using adaptive regularization methods such as Adam, which can be regarded as diagonally preconditioned stochastic gradient descent. This diagonal preconditioner comes from a diagonal approximation of the gradient outer product matrix. However, a recent open competition called “AlgoPerf: Training Algorithms benchmark competition” [1] revealed an intriguing discovery: a non-diagonal preconditioning method called Shampoo [2], which uses a Kronecker product approximation of the outer-product matrix, was found to be the best method on a varied suite of benchmark problems.

In this talk, I will introduce adaptive methods and show how Kroencker products can be used to get a computationally efficient preconditioner. I will then talk about a general technique called Combining AxeS PReconditioners (CASPR) [3], which optimizes matrix-shaped DNN parameters by finding different preconditioners for each mode/axis of the parameter and combining them using a Kronecker-sum based approximation. The Kronecker-sum based combination allows us to show that CASPR is ordered between the Kronecker product based combination, Shampoo, and full-matrix “Adagrad” preconditioners in Loewner order, and as a result it is nearer to full-matrix Adagrad than Shampoo. Experimental results demonstrate that CASPR approximates the gradient second-moment matrix more accurately, and shows improvement in training and generalization performance compared to the existing practical adaptive regularization methods in a variety of tasks including graph neural network on OGBG-molpcba, Transformer on a universal dependencies dataset and auto-regressive large language modeling on the C4 dataset.

## References

- [1] <https://mlcommons.org/2024/08/mlc-algoperf-benchmark-competition>, 2024.
- [2] V. Gupta, T. Koren and Y. Singer. Shampoo: Preconditioned Stochastic Tensor Optimization. *Proceedings of The 35th International Conference on Machine Learning (ICML)*, 2018.
- [3] S. S. Duvvuri, F. Devvrit, R. Anil, C. Hsieh and I. S Dhillon. Combining Axes Preconditioners through Kronecker Approximation for Deep Learning. *Proceedings of The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

# General Methods for Sparsity Structure Description and Cost Estimation

*Grace Dinh, James Demmel, Zhiru Zhang*

## Abstract

Sparse tensor operations (especially matrix multiplications and tensor contractions in general) can be used to represent many problems in diverse fields such as genomics, machine learning, network analysis, and electronic design automation. Casting a domain-specific problem as a sparse linear algebra operation allows domain experts to leverage existing optimized software libraries and hardware. However the cost (in terms of flops, data movement/accesses, or memory footprint) of a sparse operation can vary significantly depending on the sparsity structure of its input, making the development of *general* high-performance tools for sparse linear algebra challenging. Estimating and bounding these costs is important for many applications: coming up with a performance objective for optimizing a software or hardware implementation, developing a notion of “peak performance” to compare a benchmark against, determining how much space to allocate for scratch or for the output of an operation, load balancing, and many more.

Cost estimation is straightforward for dense linear algebra, as the exact set of arithmetic instructions is always the same and known ahead of time. However, in the sparse case, this is not possible unless the *exact* sparsity structure (i.e. the locations of every nonzero) of the inputs is known. As a result, previous cost modeling approaches, e.g. [12], tend to either require that users provide a specific input matrix (precluding their use to develop and evaluate *general* tools) or provide results restricted to specific sparsity structures (e.g. uniformly distributed sparsity, block sparsity, band matrices). For input matrices that do not neatly fit into one of these predetermined categories, however, significant case-by-case work is required on the part of users to develop statistical models that both describe their matrices and provide good cost estimates and bounds.

This abstract sketches out an approach to *generalize* and *automate* the construction of such sparsity models, and to build cost *estimates* and *bounds* that take them into account, building on techniques from database and information theory. In Section 1, we describe a way to describe sparsity structure using *matrix statistics*. We then describe how to use these statistics to bound and estimate costs in Section 2, and to optimize storage formats for sparse matrices in Section 3.

## 1 Characterizing Sparsity Structure

Our goal in this section is to describe a framework for *matrix statistics* - quantities describing the sparsity structure of a matrix that are (a) well-defined for any sparse matrix, regardless of structure, and (b) can be effectively used to predict the performance of a tensor operation. The most well known matrix statistic is the number of nonzeros (nnz) of a matrix. However, nnz alone is clearly insufficient for the cost estimation problem. Consider, for instance, the square matrix  $A$  whose first column is nonzero and whose other columns are all zero. Despite having the same input nnzs,  $A^T A$  and  $AA^T$  differ drastically in output memory footprint (and therefore data movement). As a result, accurate performance modeling requires additional statistics to describing a matrix’s sparsity structure in more detail.

One way to do so is to count the number of nonzeros in each row and column, which we refer to as the *row* and *column counts*, as in [6]. These statistics require significantly more space to store than the nnzs (as they are vectors with length equal to the number of columns and rows, respectively, of

a matrix), but provide significantly more information: taking the dot product the column count of  $A$  and row count of  $B$  gives the exact number of flops required to compute  $A \times B$ . Furthermore, row and column counts can be *summarized* using by taking  $L^p$  norms for a few small  $p$ . These norms provide a compact, *easily generalizable* way to represent how “skewed” the sparsity structure of a matrix is (e.g. how heavy-tailed the distribution of connections is in a social network graph) which can also be used to derive bounds on the cost of a matrix multiplication, as we will briefly discuss in Section 2 (see [1] for a discussion of these bounds from a database point of view).

However, row and column counts alone are insufficient to describe many forms of commonly seen sparsity patterns, e.g. band and block-sparse matrices. To represent these patterns, we will extend the notion of *indices* to *functions* of the rows and column index. For a concrete example, consider a tridiagonal matrix  $A$  indexed by  $(i, j)$ . All of the locations of its nonzeros take on only three distinct values of  $i - j$ ; as a result, “number of distinct values of  $i - j$ ” is a useful statistic that allows us to encapsulate tridiagonal matrices (and band matrices in general).

To formalize and generalize, let us view a sparse matrix  $A$  indexed by  $(i, j)$  as a *set* consisting of the location of its nonzeros:  $\{(i, j) : A_{ij} \neq 0\}$ . Let  $e_1, \dots : \mathbb{Z}^2 \rightarrow \mathbb{Z}$  be some functions (such as  $i - j$  in the above example), which we will refer to as *index maps*. Define the following two operations:

**Definition 1.** The *projection* operation  $\pi_{e_k}$  projects its input onto dimension  $e_k$  - that is,  $\pi_{e_k}(A)$  has a nonzero at location  $l$  if there exists some nonzero value in  $A$  at location  $i, j$  such that  $e_k(i, j) = l$ .

**Definition 2.** The *selection* operation  $\sigma_{e_k=\eta}$  returns the subset of nonzero locations  $(i, j)$  in  $A$  such that  $e_k(i, j) = \eta$ . When no value for  $\eta$  is given, the selection operator  $\sigma_{e_k}$  will be used to represent the list  $(\sigma_{e_k=\eta} : \eta \in \text{Im}(e_k))$ .

Let  $\circ$  represent function composition. If the output of  $g$  is a vector, let  $f \vec{o} g$  denote the *vector* obtained by applying  $f$  to every element of the output of  $g$ . Then many natural matrix statistics can be represented by choosing appropriate index maps  $e_k$ :

- Row counts: first select each row ( $i$ ), then count the number of nonzeros in each ( $|\cdot|$ ):  $|\cdot| \vec{o} \sigma_i$ . Column counts are identical, with  $j$  replacing  $i$ .
- Band width of a band matrix: first project onto  $i - j$ , then count:  $|\cdot| \circ \pi_{i-j}$
- Number of nonzero blocks in a block-sparse matrix with block size  $b$ : project onto blocks  $(\lfloor i/b \rfloor, \lfloor j/b \rfloor)$ , then count:  $|\cdot| \circ \pi_{\lfloor i/b \rfloor, \lfloor j/b \rfloor}$
- Fine-grained structured sparsity (maximum number of nonzeros in each block): for each block (i.e. selection operator on  $\lfloor i/b \rfloor, \lfloor j/b \rfloor$ ), count the number of nonzeros, then take the max:  $\max \circ |\cdot| \vec{o} \sigma_{\lfloor x_1/b \rfloor, \lfloor x_2/b \rfloor}$

Furthermore, appropriately chosen index maps can be used to characterize matrices with sparsity structures that do not align with “standard” patterns. For example, the Tuma1<sup>1</sup> matrix could be decomposed into several components, each of which would have a very small value for  $|\cdot| \circ \pi_{\alpha i-j}$  (for some constant  $\alpha$ ). Preliminary experiments show that computer vision methods such as Hough transforms [7] as well as modern machine learning methods such as symbolic regression [9] can be used to extract descriptive index maps from many real-world matrices that can be used to derive useful bounds; we leave further experimentation to future work.

---

<sup>1</sup>[https://sparse.tamu.edu/GHS\\_indef/tuma1](https://sparse.tamu.edu/GHS_indef/tuma1)

## 2 Bounds from Matrix Statistics

This section describes approaches to deriving cost bounds from matrix statistics derived in Section 1. While we focus on matrix multiplication here, our approach can generalize to most “nested loop” style programs acting on sparse data; we leave such generalization to future work. As in the previous section, we will view a sparse matrix as a set whose elements are its nonzero indices. Then a sparse matrix multiplication  $A \times B$ , where  $A$  is indexed by  $(i, j)$  and  $B$  by  $(j, k)$ , can be viewed as the set of nontrivial arithmetic instructions - that is,  $\{(i, j, k) : A_{ij} \neq 0, B_{jk} \neq 0\}$ , which we denote  $T$ . Note that this matrix multiplication tensor can be viewed as the *database join*  $A(i, j) \wedge B(j, k)$ . Several cost functions immediately fall from this representation:

- The *number of flops* required to compute  $A \times B$  is simply the cardinality of the matrix multiplication tensor  $|A(i, j) \wedge B(j, k)|$ .
- The *size of the output* is the size of the *projection* of the matrix multiply tensor onto the  $i, k$  face  $|\pi_{i,k}(A(i, j) \wedge B(j, k))|$ .
- The *arithmetic intensity* of  $A \times B$  on a system with fast memory  $M$  can upper bounded by computing the maximum number of elements for any subset of of  $T$  subject to the constraint that the projections of that subset onto the  $(i, j)$  and  $(j, k)$  dimensions are bounded by  $M$ . In previous work focusing on dense linear algebra [8, 5], this immediately provides a data movement lower bound of  $(M \times \#\text{total flops}) / (\max T\text{-subset size})$ ; however, the number of flops may not be exactly known in the sparse setting, so we will focus on upper bounding the arithmetic intensity instead.

One approach we can take to bounding these quantities is to transform the indices of the nested loops in such a way that the resulting loop nest, when treated as a dense operation, produces useful bounds. For instance, suppose we wish to multiply two band matrices  $A$  and  $B$ , which have band width  $w_1$  and  $w_2$  respectively:

$$\begin{aligned} &\text{for } i, j, k \in [0, N]^3 \\ &C(i, k) += A(i, j) \times B(j, k) \end{aligned}$$

As the two matrices are banded, we know that  $|\cdot| \circ \pi_{i-j} = w_1$  and  $|\cdot| \circ \pi_{k-j} = w_2$ . As a result, if we let  $e_1 = i - j$  and  $e_2 = k - j$ , we can rewrite this nested of loops as:

$$\begin{aligned} &\text{for } e_1 \in [0, w_1], e_2 \in [0, w_2], j \in [0, N] \\ &C(e_1 + j, e_2 + j) += A(e_1 + j, j) \times B(j, e_2 + j) \end{aligned}$$

which provides an upper bound for flops of  $w_1 w_2 N$ . Furthermore, using Brascamp-Lieb inequalities [5, 10, 4] provides an arithmetic intensity upper bound (on a system with cache size  $M$ ) of  $\sqrt{M}/2$ .

Unfortunately, this method is not easily generalized: we were able to transform indices  $i$  and  $k$  into new indices that could easily be bounded using the given matrix statistics because  $A$  and  $B$  shared band structure; this would not be possible if they were not. To address this problem, we adapt information-theoretic techniques previously used for database cardinality estimation [3, 2]. Specifically, given *any* probability distribution over set of arithmetic instructions  $T$  in the sparse matrix multiplication, let  $h$  denote the Shannon entropies of its marginal distributions  $h$  (e.g. use  $h(ij)$  to denote the entropy of the marginal distribution over  $i, j$ ). Clearly,  $h(ijk)$  is upper bounded by  $\lg |T|$ , the number of flops of the matrix multiplication. Furthermore, notice that for

an instruction  $(i, j, k)$  to be in  $T$ ,  $A_{ij}$  and  $B_{jk}$  must both be nonzero; as a result, the entropies  $h(ij)$  and  $h(jk)$  are upper bounded by  $\lg \text{nnz}(A)$  and  $\lg \text{nnz}(B)$  respectively. These inequalities can be combined with those inherent to entropy (nonnegativity, submodularity, and subadditivity) to produce bounds on cost.

For example, it can be shown that for *any* distribution on  $i, j, k$ :

$$3h(ijk) \leq h(i,j) + h(j,k) + h(i,k) + h(j|i) + h(k|j) + h(i|k)$$

Letting the distribution be the uniform distribution over  $T$  sets the left side of the above inequality to  $\lg(\#\text{flops}^3)$ , while  $h(i,j)$ ,  $h(j,i)$ , and  $h(i,k)$  are upper bounded by  $\lg \text{nnz}A$ ,  $\lg \text{nnz}B$ , and  $\lg \text{nnz}C$  respectively. Furthermore,  $h(j|i)$  is upper bounded by the log of the maximum number of nonzero elements in any row of  $A$  (similarly for the remaining terms), giving an inequality that ties together computation cost, output size, and memory footprint. In this framework, all of the cost functions above can be described: number of flops and output size are immediately derivable from entropic inequalities, and arithmetic intensity can be found by adding constraints the entropies  $h(ij)$  and  $h(jk)$  are upper bounded by  $\lg M$ . In order to adapt matrix statistics using arbitrary index maps (e.g.  $e_1 = i - j$ ), we can add additional constraints: specifically that  $h(e_1|ij) = h(i|je_1) = h(j|ie_1) = 0$ . This allows for the *automated* construction of new lower bounds for, say, the cost of multiplying of a band matrix by a block-sparse one, based on statistics such as the number of dense blocks sharing an index with a given band.

### 3 Matrix Format Optimizations

We also wish to find efficient ways to *store* sparse matrices. Consider, for example, a band matrix with a small band width. Standard sparse matrix formats, such as CSR, would require significantly more storage for metadata (row pointers and column indices) than a similar format indexed by  $i - j$  and  $j$  [11]. Furthermore, the *order* in which the indices are stored can significantly affect size and performance too - just as  $(i, j)$  (CSR) and  $(j, i)$  (CSC) are significantly different formats, so would  $(i - j, j)$  and  $(j, i - j)$ .

The choice of data structures and layouts directly impacts computing performance. For instance, to efficiently use the Gustavson algorithm, the operand tensors should ideally be stored in row-major formats. We will describe how entropic bounds (specifically, the chain bound) can suggest optimal *orderings* and *data structures* for sparse matrix storage formats.

However, performance is often heavily affected by the underlying hardware architecture. For parallel processing systems like GPUs, maintaining workload balance often outweighs achieving a high compression ratio in terms of format selection. As a result, formats with zero padding, such as ELLPACK, are commonly preferred over those that store only non-zero elements. Blocking formats, while introducing additional memory access and metadata overhead on architectures with a unified memory model, are well-suited for many-core architectures with banked memory. Work is ongoing to extend our cost models to account for hardware-specific performance factors.

## References

- [1] Mahmoud Abo Khamis, Vasileios Nakos, Dan Olteanu, and Dan Suciu. Join Size Bounds using  $l_p$ -Norms on Degree Sequences. *Proc. ACM Manag. Data*, 2(2):96:1–96:24, May 2024.
- [2] Mahmoud Abo Khamis, Hung Q. Ngo, and Dan Suciu. Computing Join Queries with Functional Dependencies. In *Proceedings of the 35th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, PODS ’16, pages 327–342, New York, NY, USA, June 2016. Association for Computing Machinery.
- [3] Mahmoud Abo Khamis, Hung Q. Ngo, and Dan Suciu. What Do Shannon-type Inequalities, Submodular Width, and Disjunctive Datalog Have to Do with One Another? In *Proceedings of the 36th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, PODS ’17, pages 429–444, New York, NY, USA, May 2017. Association for Computing Machinery.
- [4] Anthony Chen, James Demmel, Grace Dinh, Mason Haberle, and Olga Holtz. Communication bounds for convolutional neural networks. In *Proceedings of the Platform for Advanced Scientific Computing Conference*, PASC ’22, New York, NY, USA, 2022. Association for Computing Machinery.
- [5] Michael Christ, James Demmel, Nicholas Knight, Thomas Scanlon, and Katherine A. Yelick. Communication Lower Bounds and Optimal Algorithms for Programs that Reference Arrays - Part 1:. Technical report, Defense Technical Information Center, Fort Belvoir, VA, May 2013.
- [6] Kyle Deeds, Willow Ahrens, Magda Balazinska, and Dan Suciu. Galley: Modern Query Optimization for Sparse Tensor Programs, August 2024.
- [7] Paul VC Hough. Method and means for recognizing complex patterns, December 1962.
- [8] Dror Irony, Sivan Toledo, and Alexander Tiskin. Communication lower bounds for distributed-memory matrix multiplication. *Journal of Parallel and Distributed Computing*, 64(9):1017–1026, 2004.
- [9] William La Cava, Bogdan Burlacu, Marco Virgolin, Michael Kommenda, Patryk Orzechowski, Fabrício Olivetti de França, Ying Jin, and Jason H. Moore. Contemporary Symbolic Regression Methods and their Relative Performance. *Advances in Neural Information Processing Systems*, 2021(DB1):1–16, December 2021.
- [10] Auguste Olivry, Julien Langou, Louis-Noël Pouchet, P. Sadayappan, and Fabrice Rastello. Automated derivation of parametric data movement lower bounds for affine programs. In *Proceedings of the 41st ACM SIGPLAN Conference on Programming Language Design and Implementation*, PLDI ’20, pages 808–822, New York, NY, USA, 2020. Association for Computing Machinery.
- [11] Yousef Saad. *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, second edition, January 2003.
- [12] Yannan Nellie Wu, Po-An Tsai, Angshuman Parashar, Vivienne Sze, and Joel S. Emer. Sparseloop: An Analytical Approach To Sparse Tensor Accelerator Modeling. In *2022 55th IEEE/ACM International Symposium on Microarchitecture (MICRO)*, pages 1377–1395, Chicago, IL, USA, October 2022. IEEE.

# Toward Fast and Provable Data Selection under Low Intrinsic Dimension

*Yijun Dong, Per-Gunnar Martinsson, Qi Lei, Hoang Phan, Xiang Pan, Chao Chen, Katherine Pearce*

## Abstract

As the data volume and model size explode with the unprecedented successes of modern machine learning algorithms, high dimensionality is turning to the major computational bottleneck that impedes the development and democratization of large models. Since redundancies in high dimensions are ubiquitous in most real-world learning problems, the notion of *low intrinsic dimension* is introduced to characterize the minimal size of any low-dimensional manifolds that can encapsulate the essential information in the learning problem. Leveraging such low intrinsic dimensions is crucial for designing fast and sample-efficient learning algorithms for large-scale problems.

Fine-tuning that adapts powerful pre-trained models to specific downstream tasks is arguably one of the most common examples of efficient learning through low intrinsic dimensions. Intuitively, with the general knowledge encoded in the pre-trained model with high-dimensional parameters, fine-tuning within a low-dimensional parameter subspace is usually sufficient for adapting the model to new tasks. Leveraging such low intrinsic dimensions allows learning with much fewer samples (than the high parameter dimension, i.e., in the overparametrized setting) and computational resources.

In practice, natural data generally come with heterogeneous qualities and considerable redundancies, which brings about a critical question:

*How to select the most informative data for sample-efficient learning under low intrinsic dimension?*

Answers to this question are highly objective-dependent. This talk aims to provide an overview of some recent progress in two common objectives for data selection:

- (i) row (or column) subset selection for low-rank interpolative decomposition, and
- (ii) data selection for statistical learning models in kernel regime (e.g., fine-tuning).

By diving into a few randomized algorithms for interpolative (or CUR) decompositions and data selection based on random pivoting and sketching, we will unveil the power of randomization in fast and robust data selection, from both the empirical and theoretical perspectives.

## Data Selection for Low-rank Interpolative Decompositions

The interpolative decomposition (ID) aims to construct a low-rank approximation formed by a basis consisting of row (or column) skeletons in the original matrix and a corresponding interpolation matrix. We explore fast and accurate ID algorithms from five essential perspectives for empirical performance:

- (i) *skeleton complexity* that measures the minimum possible ID rank for a given low-rank approximation error,
- (ii) *asymptotic complexity* in floating point operations (FLOPs),
- (iii) *parallelizability* of the computational bottleneck, i.e., whether the steps with dominant cost can be cast into matrix-matrix, instead of matrix-vector, multiplications,
- (iv) *error-revealing property* that enables automatic rank detection for given error tolerances with-

out prior knowledge of target ranks,

- (v) *ID-revealing property* that ensures efficient construction of the optimal interpolation matrix after selecting the skeletons.

While many algorithms have been developed to optimize parts of the aforementioned perspectives, practical ID algorithms proficient in all perspectives remain absent. To fill in the gap, we introduce *robust blockwise random pivoting (RBRP)* that is parallelizable, error-revealing, and exactly ID-revealing, with comparable skeleton and asymptotic complexities to the best existing ID algorithms in practice. Through extensive numerical experiments on various synthetic and natural datasets, we demonstrate the appealing empirical performance of RBRP from the five perspectives above, as well as its robustness to adversarial inputs.

In a nutshell, random pivoting for interpolative decomposition involves adaptively sampling rows (or columns) according to their squared  $\ell_2$ -norm and updating the data matrix by projecting the remaining rows (or columns) onto the orthogonal complement of the current basis. Such an adaptive sampling scheme ensures that the selected rows (or columns) are informative and diverse, leading to a small skeleton complexity for given low-rank approximation errors. However, the sequential nature of random pivoting compromises its parallelizability and empirical efficiency. Alternatively, the sequential random pivoting can be naïvely extended to a faster blockwise version that samples a block of  $b > 1$  points according to the current squared  $\ell_2$ -norm in each step and updates the data matrix blockwisely. However, such plain blockwise random pivoting tends to suffer from unnecessarily large skeleton complexity under adversarial inputs due to the lack of local adaptiveness within each block. As a remedy, RBRP leverages *robust blockwise filtering*—applying CPQR to every small sampled block locally and discarding the potentially redundant points through a truncation on the relative residual of the CPQR. By choosing a reasonable block size, such robust blockwise filtering effectively resolves the inefficiency in skeleton complexity encountered by the plain blockwise random pivoting, with negligible additional cost.

## Data Selection for Statistical Learning Models in Kernel Regime

Fine-tuning can be viewed as learning with a good pre-trained initialization that lies in some neighborhood of an optimal solution, whose dynamics fall into the kernel regime. Therefore, fine-tuning a regression task (under Tikhonov regularization with a suitable hyperparameter) can be well approximated by

- (i) a linear regression problem in the low-dimensional (overdetermined) setting, or
- (ii) a ridge regression problem in the high-dimensional (overparametrized) setting<sup>1</sup>.

For overdetermined linear regression in low dimension, data selection falls in the classical frames of coresets selection for linear regression and optimal experimental design where the generalization gap can be reduced by selecting data that minimize the associated variance. However, for overparametrized problems, variance minimization alone is insufficient to characterize the generalization. In particular, when the parameter dimension  $r$  is higher than the coreset size  $n$ , the selected data necessarily miss a parameter subspace of dimension at least  $r - n$ , leading to errors in addition to variance.

Nevertheless, the prevailing empirical and theoretical evidence on the ubiquitous intrinsic low-dimensional structures in high-dimensional problems motivates a natural question:

---

<sup>1</sup>We refer to “low-dimension” as the setting where the number of parameters  $r$  is smaller than the selected downstream sample size  $n$ , while “high-dimension” refers to the opposite,  $r > n$ .

*Can the low intrinsic dimension be leveraged in data selection for high-dimensional fine-tuning?*

We provide a positive answer to this question through a variance-bias tradeoff perspective. Intuitively, we consider a low-dimensional subspace  $\mathcal{S}$  in the fine-tuning parameter space where the model learns the necessary knowledge for the downstream task. The generalization gap can be controlled by simultaneously reducing *the bias (redundant information)* by “exploring” the parameter space to find a suitable  $\mathcal{S}$  and *the variance* by “exploiting” the useful knowledge in  $\mathcal{S}$ .

Given the high-dimensional nature of the parameter space, a direct search for such suitable subspace  $\mathcal{S}$  is computationally infeasible in general. This leads to a follow-up question:

*How to explore the intrinsic low-dimensional structure efficiently for data selection?*

We propose *Sketchy Moment Matching (SkMM)*, a two-stage solution for this question:

- (i) **Gradient sketching for bias reduction:** First, we construct a low-dimensional parameter subspace  $\mathcal{S}$  by sketching the model gradients. Sketching is a well-established dimensionality reduction tool known for affordable and accurate low-rank approximations. In deep learning, sketching recently extends its empirical applications to scalable estimations of influence functions for data selection. We make a first step toward the theoretical guarantee of gradient sketching for data selection: *gradient sketching efficiently finds a low-dimensional subspace  $\mathcal{S}$  with small bias such that selecting  $n$  samples by reducing variance over  $\mathcal{S}$  is sufficient to preserve the fast-rate generalization  $O(\dim(\mathcal{S})/n)$ , linear in the low intrinsic dimension  $\dim(\mathcal{S})$  while independent of the high parameter dimension  $r$ .*
- (ii) **Moment matching for variance reduction:** Second, we select data that reduce variance in the low-dimensional subspace  $\mathcal{S}$  via moment matching. The variance of data selection is characterized by matching between the sketched gradient moments of the original and selected datasets,  $\tilde{\Sigma}, \tilde{\Sigma}_S$ , respectively. The objective  $\text{tr}(\tilde{\Sigma}\tilde{\Sigma}_S^\dagger)$  takes the form of V-optimality in optimal experimental design, whose exact minimization is computationally intractable. Existing polynomial-time heuristics for V-optimality are generally based on the continuous relaxation of the V-optimality objective followed by a fast rounding process. However, solving such a continuous relaxation can be challenging in practice, as it involves inverting a potentially ill-conditioned matrix  $\tilde{\Sigma}_S$  in each iteration. Under a common heuristic assumption that  $\tilde{\Sigma}, \tilde{\Sigma}_S$  commute, we introduce a continuous relaxation with a quadratic objective and linear constraints that is numerically stable (free of inversions) and can be efficiently optimized via projected gradient descent.

With synthetic mixtures of Gaussian data, we first demonstrate how SkMM balances variance and bias in data selection for overparametrized ridge regression and leads to sample-efficient learning. Then, with extensive experiments on fine-tuning CLIP or ImageNet pre-trained vision models for both regression and classification tasks, we show the appealing sample and computational efficiency of SkMM, along with its surprising robustness to data heterogeneity.

## References

- Robust Blockwise Random Pivoting: Fast and Accurate Adaptive Interpolative Decomposition. Yijun Dong, Chao Chen, Per-Gunnar Martinsson, Katherine Pearce. *arXiv*: 2309.16002, 2023.
- Sketchy Moment Matching: Toward Fast and Provable Data Selection for Finetuning. Yijun Dong\*, Hoang Phan\*, Xiang Pan\*, Qi Lei. *Conference on Neural Information Processing Systems (NeurIPS)*, 2024. (to appear)

# Structured rational matrices: properties and strongly minimal linearizations

*Froilán Dopico, Vanni Noferini, María C. Quintana and Paul Van Dooren*

## Abstract

Rational matrices, that is, matrices whose entries are univariate rational functions appear in control problems and also in the numerical solution of non-linear eigenvalue problems as approximations of other matrices whose entries are more general univariate functions. Very often the rational matrices arising in applications have particular structures that should be preserved/used in the numerical computation of their poles, zeros and minimal indices.

In this talk, we consider three classes of rational matrices  $R(z)$  that are Hermitian upon evaluation on (a) the real axis, (b) the imaginary axis, or (c) the unit circle. Our goal is to show how to construct linear polynomial system matrices, i.e., pencils, for those  $R(z)$  that preserve the corresponding structures and are strongly minimal, a property that guarantee that such polynomial system matrices allow for a complete recovery of the poles, zeros, and minimal indices of  $R(z)$ . Thus, structured generalized eigenvalue algorithms applied to these pencils will allow us to compute all these quantities in a structure preserving manner.

Our goal is fully achieved for the Hermitian structures on the real and on the imaginary axes, but for the Hermitian structure on the unit circle some obstacles arise, which require to modify the original problem at some extent and to construct a structured linear polynomial system matrix for the rational function  $(1+z)R(z)$  instead of for  $R(z)$ . In order to do this, we need to prove a number of previously unknown properties of rational matrices which are Hermitian on the unit circle.

The results presented in this talk are based on the references [1] and [2].

## References

- [1] F. Dopico, M.C. Quintana, P. Van Dooren, Strongly minimal self-conjugate linearizations for polynomial and rational matrices, *SIAM J. Matrix Anal. Appl.* **43**, (2022), 1354–1381.
- [2] F. Dopico, V. Noferini, M.C. Quintana, P. Van Dooren, Para-Hermitian rational matrices, to appear in *SIAM J. Matrix Anal. Appl.* (arXiv:2407.13563).

This work has been partially supported by the *Agencia Estatal de Investigación of Spain* MCIN/AEI/10.13039/501100011033/ through grant PID2023-147366NB-I00.

# Numerical linear algebra for data driven analysis of nonlinear dynamics: Koopman-Schur Decomposition

*Zlatko Drmač, Igor Mezić*

## Abstract

The Extended Dynamic Mode Decomposition (DMD/EDMD) has become a tool of trade in computational data driven analysis of complex nonlinear dynamical systems, e.g. fluid flows, where it can be used to reveal coherent structures by decomposing the flow field into component fluid structures, called DMD modes, that describe the evolution of the flow. The theoretical underpinning of the EDMD is the Koopman composition operator that can be used for spectral analysis of nonlinear dynamical system [6]. The numerical realization and software implementation pose several challenges to numerical linear algebra, and this contribution discusses few selected ones.

To set the stage, consider the initial value problem

$$\dot{\mathbf{x}}(t) = F(\mathbf{x}(t)) \equiv \begin{pmatrix} F_1(\mathbf{x}(t)) \\ \vdots \\ F_N(\mathbf{x}(t)) \end{pmatrix}, \quad \mathbf{x}(t_0) = \mathbf{x}_0, \quad (1)$$

with state space  $\mathcal{X} \subset \mathbb{R}^N$  and vector-valued nonlinear function  $F : \mathcal{X} \rightarrow \mathbb{R}^N$ . The corresponding flow map  $\phi^t$  is defined as  $\phi^t(\mathbf{x}(t_0)) = \mathbf{x}(t_0 + t) = \mathbf{x}(t_0) + \int_{t_0}^{t_0+t} F(\mathbf{x}(\tau))d\tau$ .

Instead of the states, consider observables (functions of the states)  $f : \mathcal{X} \rightarrow \mathbb{C}$ ,  $f \in \mathcal{F}$ ; e.g.  $\mathcal{F} = L^p(\mathcal{X}, \mu)$  ( $1 \leq p \leq \infty$ ). Koopman operator semigroup  $(\mathcal{U}_{\phi^t})_{t \geq 0}$  is defined by  $\mathcal{U}_{\phi^t}f = f \circ \phi^t$ ,  $f \in \mathcal{F}$ . The Koopman (composition) operator  $\mathcal{U}_{\phi^t}$  is linear operator that can be considered an infinite dimensional linearization of (1) that takes the action into the space  $\mathcal{F}$  of observables.

In the case of discrete dynamical system  $\mathbf{x}_{i+1} = T(\mathbf{x}_i)$ , the Koopman operator  $\mathcal{U} \equiv \mathcal{U}_T$  is defined on a space of observables  $\mathcal{F}$  by  $\mathcal{U}f = f \circ T$ ,  $f \in \mathcal{F}$ . In practical computation, one always works with discrete systems. If we run a numerical simulation of the ODE's (1) in a time interval  $[t_0, t_*]$ , the numerical solution is obtained on a discrete equidistant grid with fixed time lag  $\Delta t$ :  $t_0, t_1 = t_0 + \Delta t, \dots, t_{i-1} = t_{i-2} + \Delta t, t_i = t_{i-1} + \Delta t, \dots$ . In this case, a black-box software toolbox acts as a discrete dynamical system  $\mathbf{z}_i = T(\mathbf{z}_{i-1})$  that produces the discrete sequence of  $\mathbf{z}_i \approx \mathbf{x}(t_i)$ . For  $t_i = t_0 + i\Delta t$  we have  $f(\mathbf{x}(t_0 + i\Delta t)) = (f \circ \phi^{i\Delta t})(\mathbf{x}(t_0)) = (\mathcal{U}_{\phi^{i\Delta t}}f)(\mathbf{x}(t_0)) = (\mathcal{U}_{\phi^{\Delta t}}^i f)(\mathbf{x}(t_0))$ , where  $\mathcal{U}_{\phi^{\Delta t}}^i = \mathcal{U}_{\phi^{\Delta t}} \circ \dots \circ \mathcal{U}_{\phi^{\Delta t}}$ .

On the other hand, using  $\mathcal{U}f = f \circ T$ ,  $\mathbf{z}_i \approx \mathbf{x}(t_i)$ ,  $T^2 = T \circ T$ ,  $T^i = T \circ T^{i-1}$ ,  $f(\mathbf{z}_i) = f(T(\mathbf{z}_{i-1})) = \dots = f(T^i(\mathbf{z}_0)) = (\mathcal{U}^i f)(\mathbf{z}_0)$ . Hence, in a software simulation of (1) with the initial condition  $\mathbf{z}_0 = \mathbf{x}(t_0)$ , we have an approximation  $(\mathcal{U}^i f)(\mathbf{z}_0) \approx (\mathcal{U}_{\phi^{\Delta t}}^i f)(\mathbf{z}_0)$ ,  $f \in \mathcal{F}$ ,  $\mathbf{z}_0 \in \mathcal{X}$ ,  $i = 0, 1, 2, \dots, m$ . Such a sequence of values from the trajectory of the system may be also obtained by experimental measurements (e.g. high speed camera recording, wind tunnel measurements, particle image velocimetry/thermometry), without using/knowing the governing equations. In general, the observables are vector valued,  $\mathbf{f} = (f_1, \dots, f_n)^T$ , and  $\mathcal{U}\mathbf{f} = (\mathcal{U}f_1, \dots, \mathcal{U}f_n)^T$  is defined component-wise. Often  $n \gg m$ . The acquired numerical values of the observables (data snapshots) are assembled column-wise into the matrix (column index corresponds to discrete time step)

$$F = (\mathbf{f}(\mathbf{x}_0) \dots \mathbf{f}(\mathbf{x}_{m-1}) \mathbf{f}(\mathbf{x}_m)) = \begin{pmatrix} f_1(\mathbf{x}_0) & \dots & f_1(\mathbf{x}_{m-1}) & f_1(\mathbf{x}_m) \\ f_2(\mathbf{x}_0) & \dots & f_2(\mathbf{x}_{m-1}) & f_2(\mathbf{x}_m) \\ \vdots & \vdots & \vdots & \vdots \\ f_n(\mathbf{x}_0) & \dots & f_n(\mathbf{x}_{m-1}) & f_n(\mathbf{x}_m) \end{pmatrix}, \quad \mathbf{f}(\mathbf{x}_{i+1}) = \mathbf{f}(T(\mathbf{x}_i)) = (\mathcal{U}\mathbf{f})(\mathbf{x}_i) = \mathbf{f}(T(T(\mathbf{x}_{i-1}))).$$

The columns of the data snapshot matrix  $F$  can be interpreted as the Krylov sequence  $\mathbf{f}, \mathcal{U}\mathbf{f}, \dots, \mathcal{U}^m\mathbf{f}$ , evaluated at  $\mathbf{x}_0$ , which motivates looking for approximate eigenvalues and eigenvectors of  $\mathcal{U}$ .

Why is spectral data of  $\mathcal{U}$  interesting? If  $(\mathcal{U}\phi_i)(s) \approx \lambda_i\phi_i(s)$ ,  $i = 1, \dots, m$ , and if for some carefully and judiciously selected  $\lambda_{i_1}, \dots, \lambda_{i_\ell}$  and vector coefficients  $\mathbf{z}_{i_j}$  (that must be computed)

$$\mathbf{f}(s) \approx \sum_{j=1}^{\ell} \mathbf{z}_{i_j} \phi_{i_j}(s), \text{ then } (\mathcal{U}^k \mathbf{f})(s) = \begin{pmatrix} (\mathcal{U}^k f_1)(s) \\ \vdots \\ (\mathcal{U}^k f_n)(s) \end{pmatrix} \approx \sum_{j=1}^{\ell} \mathbf{z}_{i_j} \phi_{i_j}(s) \lambda_{i_j}^k, \quad k = 0, 1, \dots \quad (2)$$

This decomposition (called the Koopman Mode Decomposition, KMD) reveals the latent structure of the dynamics (in particular when  $\ell$  is relatively small) and allows for forecasting future values, because applying the powers of  $\mathcal{U}$  in (2) means pushing the dynamics forward in time.

In this decomposition, the vector coefficients  $\mathbf{z}_i$ 's are approximate eigenvectors of a matrix  $A$  such that  $A\mathbf{f}(\mathbf{x}_i) = \mathbf{f}(T(\mathbf{x}_i))$  for all  $i$ . The matrix  $A$  is the DMD matrix – it may not be uniquely determined by the data and only certain Ritz pairs can be computed by a data driven version of the classical Rayleigh-Ritz extraction procedure, as in the DMD algorithm [8] and its enhancement [5] that provides computable residuals and uses them to select physically meaningful eigenvalues and modes, and to guide sparse representation of the snapshot in the KMD.

But, there is a problem. One of the computational/numerical challenges in the Koopman/DMD framework is the case of non-normal operators, when the computed (Ritz) eigenvectors of the DMD matrix become severely ill-conditioned. High non-normality of the eigenmodes is not just a mathematically manufactured and for the sake of academic exercise contrived misfortune. It does occur in important applications. For instance, Schmid [7] discusses examples of non-normal operators in fluid flows, and the impact of non-normality on treatment of stability of such flows. A well-known and intensively studied example is the formulation of the viscous stability problem for parallel shear flows, in which linearization of the Navier-Stokes equation leads to the Orr-Sommerfeld linear partial differential equation whose solutions exhibit highly non-normal behavior.

The severity of the problem can be easily demonstrated by running a numerical simulation and visualizing the pseudospectrum and computing the angles between the eigenvectors. This issue is mostly ignored in the DMD literature, but the practitioners have experienced the problem in applications, and it is listed in [10] as one of the challenges in applications of the DMD.

To alleviate the problem of ill-conditioned eigenvectors in the existing implementations of the Dynamic Mode Decomposition (DMD) and the Extended Dynamic Mode Decomposition (EDMD, [9]), in [4] we introduce a Koopman-Schur decomposition – Schur decomposition of a data driven compression of the Koopman operator onto a subspace  $\mathcal{F}_N$  defined by a given dictionary  $\psi_1, \dots, \psi_N$ .

The first step in this approach is as follows. As in the EDMD, compute the data driven compression

$$P_{\mathcal{F}_N} \mathcal{U}|_{\mathcal{F}_N} ((\psi_1(x) \ \dots \ \psi_N(x)) \begin{pmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_N \end{pmatrix}) = (\psi_1(x) \ \dots \ \psi_N(x)) (U_N \begin{pmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_N \end{pmatrix}),$$

where  $P_{\mathcal{F}_N}$  stands for the algebraic least squares projection using the available data, and  $U_N$  is the matrix of the compression. With the notation  $\Psi(x) = (\psi_1(x), \dots, \psi_N(x))^T \in \mathbb{C}^N$ , the action of  $\mathcal{U}$  on  $\mathcal{F}_N$  can be compactly written as  $\mathcal{U}(\Psi(\mathbf{x})^T \mathbf{z}) = \Psi(\mathbf{x})^T U_N \mathbf{z} + R(\mathbf{x})^T \mathbf{z}$ ,  $\mathbf{x} \in \mathbb{C}^n$ ,  $\mathbf{z} \in \mathbb{C}^N$ , where  $R(x) = (\rho_1(x), \dots, \rho_N(x))^T \in \mathbb{C}^N$  is the residual that has been minimized over the available data. In this setting, a well defined object is, instead of  $U_N$ , another compression –  $U_N$  is compressed

onto a  $N \times r$  POD basis  $V$  (computed using the SVD of a data snapshot matrix, with  $r < N$ ) and replaced with the  $r \times r$  Rayleigh quotient  $\widehat{U}$ . In the proposed approach [4], we compute a Schur decomposition of  $\widehat{U}$ ,  $\widehat{U} = QTQ^*$ , and  $U_N Z = ZT$  becomes a partial Schur form with  $Z = VQ$ . On the operator level, this becomes

$$\begin{aligned}\mathcal{U}((\psi_1(x) \dots \psi_N(x)) Z) &= (\psi_1(x) \dots \psi_N(x)) U_N Z + R(x)^T Z \\ &= (\psi_1(x) \dots \psi_N(x)) ZT + R(x)^T Z \simeq (\psi_1(x) \dots \psi_N(x)) ZT.\end{aligned}\quad (3)$$

If we define a new sequence  $(\zeta_1(x) \dots \zeta_r(x)) = (\psi_1(x) \dots \psi_N(x)) Z$ , then (3) reads

$$\mathcal{U}(\zeta_1(x) \dots \zeta_r(x)) = (\zeta_1(x) \dots \zeta_r(x)) T + R(x)^T Z \simeq (\zeta_1(x) \dots \zeta_r(x)) T. \quad (4)$$

Since  $T$  is upper triangular, (4) contains a nested sequence of partial triangulations

$$\mathcal{U}(\zeta_1 \ \zeta_2 \ \dots \ \zeta_i) \simeq (\zeta_1 \ \zeta_2 \ \dots \ \zeta_i) T(1:i, 1:i), \quad i = 1, \dots, r. \quad (5)$$

This Schur form can be reordered – with a suitable unitary matrix  $\Theta$ ,  $\tilde{T} = \Theta^* T \Theta$  is again upper triangular with diagonal entries corresponding to the eigenvalues in any given order. The new partial Schur form of  $U_N$  becomes  $U_N(VQ\Theta) = (VQ\Theta)\tilde{T}$ ,  $\tilde{T} = \Theta^* T \Theta$ , and we replace (4) with

$$\mathcal{U}(\zeta_1(x) \ \zeta_2(x) \ \dots \ \zeta_r(x)) \Theta \simeq (\zeta_1(x) \ \zeta_2(x) \ \dots \ \zeta_r(x)) \Theta(\Theta^* T \Theta), \quad (6)$$

i.e. the new functions are generated using  $\tilde{Z} = Z\Theta = VQ\Theta$  ( $\tilde{Z}^* \tilde{Z} = I_r$ ),

$$(\tilde{\zeta}_1(x) \ \dots \ \tilde{\zeta}_r(x)) = (\zeta_1(x) \ \dots \ \zeta_r(x)) \Theta = (\psi_1(x) \ \dots \ \psi_N(x)) \tilde{Z}. \quad (7)$$

In this framework, the spectral analysis of the snapshots, including the KMD (2), is entirely based on unitary transformations. The analysis in terms of the eigenvectors as modes of a Koopman operator compression is replaced with a modal decomposition in terms of a flag of invariant subspaces that correspond to selected eigenvalues – the partial ordered Schur decomposition provides convenient orthonormal bases for subspaces determined by any given selection  $\lambda_{i_1}, \dots, \lambda_{i_\ell}$  of the eigenvalues.

From this point, we proceed in two direction. First, we analyze the convergence (as the size of the dictionary and the number of data snapshots become infinite) to obtain results analogous to the EDMD. Then, to have the same functionality as the existing EDMD (snapshot reconstruction using selected eigenvalues, dynamically changed data window in online applications, forecasting, formulation with the kernel trick etc.), many technical (algorithms and software related) details have to be worked out. For instance, in the case of real data, a real ordered partial Schur form is used and the computation is based on real orthogonal transformations, even when the spectrum is complex. Other details include e.g. streaming data with dynamically changing data windows. Numerical experiments show superior performances in the numerically difficult non-normal cases.

The second topic that will be discussed is numerical implementation of the Hermitian case of the physic informed DMD [1], [3] – if it is a priori known that the underlying operator is Hermitian, how to ensure that the numerical implementation of the DMD guarantees real spectrum and orthonormal eigenvectors? What are the other issues when it comes to software implementation [2] e.g. in the framework of the LAPACK library?

These themes are excellent case study examples that demonstrate the importance of numerical linear algebra and of the power of numerical analysis of an algorithm – it precisely predicts in what way it may fail and indicates what has to be done to provably fix the problem.

## References

- [1] P. J. Baddoo, B. Herrmann, B. J. McKeon, J. N. Kutz, and S. L. Brunton. Physics-informed dynamic mode decomposition (piDMD). *arXiv e-prints*, page arXiv:2112.04307, December 2021.
- [2] Zlatko Drmač. A LAPACK implementation of the Dynamic Mode Decomposition. *ACM Trans. Math. Softw.*, jan 2024.
- [3] Zlatko Drmač. Hermitian Dynamic Mode Decomposition – numerical analysis and software solution. *ACM Trans. Math. Softw.*, jan 2024.
- [4] Zlatko Drmač and Igor Mezić. A data driven Koopman–Schur decomposition for computational analysis of nonlinear dynamics. *arXiv e-prints*, page arXiv:2312.15837, December 2023.
- [5] Zlatko Drmač, Igor Mezić, and Ryan Mohr. Data driven modal decompositions: Analysis and enhancements. *SIAM Journal on Scientific Computing*, 40(4):A2253–A2285, 2018.
- [6] Clarence W. Rowley, Igor Mezić, Shervin Bagheri, PhilippP Schlatter, and Dan S. Henningson. Spectral analysis of nonlinear flows. *Journal of Fluid Mechanics*, 641:115–127, 2009.
- [7] Peter J Schmid. Nonmodal stability theory. *Annual review of fluid mechanics*, 39(1):129–162, 2007.
- [8] Peter J. Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656:5–28, 2010.
- [9] M.O. Williams, I.G. Kevrekidis, and C.W. Rowley. A data–driven approximation of the Koopman operator: extending dynamic mode decomposition. *Journal of Nonlinear Science*, 25(6):1307–1346, 2015.
- [10] Ziyou Wu, Steven L. Brunton, and Shai Revzen. Challenges in dynamic mode decomposition. *Journal of The Royal Society Interface*, 18(185):20210686, 2021.

# Nonlinear inverse scattering data transforms via causal transmutation matrices

*Vladimir Druskin, Shari Moskow, Mikhail Zaslavskiy*

## Abstract

Many important problems in remote sensing, where measurements are not available in the domain of interest (radar imaging, seismic exploration, medical array ultrasound, etc.) lead to inverse scattering problems which can be strongly nonlinear in case of large perturbations of the unknown PDE coefficients. The model based nonlinear optimization which is the method of choice for the solution of such problems can be unreliable and prohibitively expensive. Data driven nonlinear transforms emerged as an attractive alternative, however it was recently shown that the most common ReLU neural networks are intractable for reliable solution of non-parametric inverse problems. Data driven ROMs recently emerged as a feasible option for such problems.

The key of this approach is data-driven computation of the state solution in the domain of interest not available for direct measurements for a black-box model via a nonlinear transform. It implicitly was used for different imaging applications with data-driven ROMs, e.g., see, [2, 3, 1]. Here we give its new explicit formulation allowing simple analysis and clear connection with preceding work.

We assume the following wave model problem for a domain  $\Omega \subset R^d$

$$u_{tt} - \Delta u + q(x)u = 0 \text{ in } \Omega \times [0, \infty) \quad (1)$$

with initial conditions

$$u(t=0) = g \text{ in } \Omega \quad (2)$$

$$u_t(t=0) = 0 \text{ in } \Omega \quad (3)$$

where  $g(x)$  is an initial condition representing a localized source near the boundary, and we assume homogeneous Neumann boundary conditions on the spatial boundary  $\partial\Omega$ . We assume  $q(x) \geq 0$  is our unknown potential, not necessarily small, but with compact support. The exact forward solution to (1-3) is

$$u(x, t) = \cos(\sqrt{A}t)g(x), \quad (4)$$

where

$$A = -\Delta + q(x),$$

with the square root and cosine are defined via the spectral theorem. This solution is assumed to be unknown, except near the receivers.

Assume we measure the SISO transfer function at the receiver collocated with the at the  $2n - 1$  evenly spaced time steps  $t = k\tau$  for  $k = 0, \dots, 2n - 2$ , modeled by

$$\begin{aligned} F(k\tau) &= \int_{\Omega} g(x)u(x, k\tau)dx \\ &= \int_{\Omega} g(x) \cos(\sqrt{A}k\tau)g(x)dx. \end{aligned} \quad (5)$$

The inverse problem we consider is as follows: Given

$$\{F(k\tau)\} \text{ for } k = 0, \dots, 2n - 2,$$

reconstruct  $q$ . This generally nonlinear problem becomes a simple linear problem if  $u_k = u(x, k\tau)$  is known in the entire domain, that may not be accessible to the direct measurement.

*So our objective here is to compute, from the measured data  $F(k\tau)$  only, approximations of the internal snapshots  $u_k = u(x, k\tau)$  for  $k = 0, \dots, n - 1$  assuming that  $q(x)$  is unknown.*

We introduce Gramian matrix  $M$  with elements

$$M_{kl} = \int_{\Omega} u_k u_l dx \quad (6)$$

for  $k, l = 0, \dots, n - 1$ , can be written as

$$M_{kl} = \int_{\Omega} g(x) \cos(\sqrt{A}k\tau) \cos(\sqrt{A}l\tau) g(x) dx. \quad (7)$$

thanks to the formula (4). Then, from (5), (7), and the cosine angle sum formula, one has

$$M_{kl} = \frac{1}{2} (F((k-l)\tau) + F((k+l)\tau)), \quad (8)$$

so *Gramian  $M$  can obtained directly from the data [2]*. Formula (7) is the Chebyshev moment problem yielding  $M$  given by 8 as the sum of Toeplitz and Hankel matrices.

Now, let

$$U = [u_1(x), \dots, u_n(x)]$$

be a row vector of the true snapshots, so that we can write

$$M = \int_{\Omega} U^T U \in R^{n \times n}. \quad (9)$$

Consider also the background field  $u^0(x, t)$ , the solution to (1-3) with  $q(x) = 0$ , which we assume that we know. Let

$$U_0 = [u_1^0(x), \dots, u_n^0(x)]$$

be a row vector of the background snapshots  $u_k^0 = u^0(x, k\tau)$ , and let

$$M_0 = \int_{\Omega} U_0^T U_0 \in R^{n \times n}$$

be the background mass matrix. We want to obtain approximation

$$U \approx \mathbf{U} = U_0 T$$

satisfying (9) via projection, i.e., condition

$$\int_{\Omega} \mathbf{U}^T \mathbf{U} = M \quad (10)$$

that yields

$$M = T^T M_0 T. \quad (11)$$

Equation (11) was inspired by the celebrated Marchenko-Gelfand-Levitan-Krein (MGLK) Volterra equation, e.g., see [5] and references wherein, with  $T$  being so-called transmutation matrix. Its discrete analogy first appeared in study of connection between the discrete MGLK and Lanczos algorithms [6]. A critical observation is that the waves in background ( $q = 0$ ) and true (unknown  $q$ ) media travel with the same speed, so thanks to the causality principle,  $T$  is upper triangular matrix. This restriction leads to its uniqueness, that can be shown by direct calculations.

**Proposition 1** *The row vector of data generated internal fields  $\mathbf{U} = U_0 T$  satisfying (10) with upper triangular transmutation matrix  $T$  is unique and can be computed as*

$$T = (L_0^\top)^{-1} L^\top. \quad (12)$$

where upper triangular matrices  $L$  and  $L_0$  are defined via Cholesky factorizations

$$M = LL^\top \quad M_0 = L_0 L_0^\top$$

The Cholesky factorization of  $M$  constitutes the nonlinear part of the data transform. The internal solution generated via SISO data was successfully used for radar imaging applications, however it had significant artifacts due to lack of aperture [4].

For seismic exploration and medical array ultrasound the SISO data (5) can be replaced by the square MIMO transfer function

$$\begin{aligned} F(k\tau) &= \int_{\Omega} G(x) u(x, k\tau) dx \\ &= \int_{\Omega} g(x) \cos(\sqrt{A}k\tau) G(x) dx \in R^{m \times m}, \end{aligned} \quad (13)$$

where  $G(x) = [g_1(x), \dots, g_m(x)]$  is the row vector of  $m$  transmitters collocated with receivers. The Proposition 1 can be replaced by its block analogy for the data given by (13), which was implicitly used in e.g., [3].

Another important application, the synthetic aperture radars (SAR) used for imaging from airborne platforms can only access  $\text{diag } F$ , and for computing block-transmutation matrix they require data-completion.

Finally we outline the list of computational linear algebra bottlenecks in the proposal framework:

- Fast Cholesky factorization and spectral decomposition of sum of block Hankel and Toeplitz matrices
- Lifting partial data matrices to full square MIMO data, e.g., from diagonal as in the SAR framework
- Efficient truncation or correction of spurious non-Hamiltonian modes (with negative eigenvalues, appeared due to measurement errors or inaccuracies of the above mentioned data-completion or lifting) of data-driven Gramians
- Estimation of non-strictly triangular data-driven transmutation matrices
- Extension to problems with dissipation

## References

- [1] Liliana Borcea, Josslin Garnier Alexander V. Mamonov, and Jörn Zimmerling, *When data driven reduced order modeling meets full waveform inversion* SIAM Review , 66 (3) (216): 501-532.

- [2] Druskin, V.r, A. Mamonov, A. Thaler, and M. Zaslavsky. "Direct, nonlinear inversion algorithm for hyperbolic problems via projection-based model reduction." SIAM Journal on Imaging Sciences 9, no. 2 (2016): 684-747.
- [3] Druskin, V., Mamonov, A. V., Zaslavsky, M. (2018). A nonlinear method for imaging with acoustic waves via reduced order model backprojection. SIAM Journal on Imaging Sciences, 11(1), 164-196.
- [4] Druskin, V., Moskow, S. and Zaslavsky, M., 2024. Reduced Order Modeling Inversion of Monostatic Data in a Multi-scattering Environment. SIAM Journal on Imaging Sciences, 17(1), pp.334-350.
- [5] T. Habashy. A generalized Gelfand–Levitan–Marchenko integral equation. Inverse Problems, 7, pp.703–711, 1991.
- [6] F. Natterer. *A discrete Gelfand–Levitan theory*. Electronic, from the author's web page.

# Julia, Portable Numerical Linear Algebra, and Beyond

*Alan Edelman*

## Abstract

Nearly 20 years ago, Demmel, Dongarra et. al. wrote in the Linear Algebra Working Notes (LAWN) 181 what appears to be a nearly impossible combinatorial explosion of challenges:

- (1) for all linear algebra problems  
(linear systems, eigenproblems, ...)
- (2) for all matrix types  
(general, symmetric, banded, ...)
- (3) for all data types  
(real, complex, single, double, higher precision)
- (4) for all machine architectures  
and communication topologies
- (5) for all programming interfaces
- (6) provide the best algorithm(s) available in terms of  
performance and accuracy ("algorithms" is plural because sometimes  
no single one is always best)

Twenty years later the concept of data types has extended to many more important possibilities (e.g., quaternion, mixed precision), GPUs have grown in importance and in number, and how linear algebra is integrated into larger applications has grown to become more complex than the traditional library model. Nonetheless, the dream of solving this problem remains, and we believe that the abstractions provided by Julia may be key. In this talk we will report some of the solutions provided by the Julia Lab at MIT and beyond.

# Matrix-less spectral approximation for large structured matrices

*Giovanni Barbarino, Melker Claesson, Sven-Erik Ekström,  
Carlo Garoni, David Meadon, and Hendrik Speleers*

## Abstract

Sequences of structured matrices of increasing size arise in many scientific applications and especially in the numerical discretization of linear differential problems; for example by using Finite Differences (FD), Finite Elements (FE), Finite Volumes (FV), Discontinuous Galerkin (DG), Iso-greometric Analysis (IgA). The eigenvalues  $\lambda_j(A_n)$  of matrices  $A_n$ , belonging to such a sequence  $\{A_n\}_n$ , can often be approximated by a regular expansion:

$$\lambda_j(A_n) = \sum_{k=0}^{\alpha} c_k(\theta_{j,n}) h^k + E_{j,n,\alpha}, \quad j = 1, \dots, n \quad \theta_{j,n} = \frac{j\pi}{n+1}, \quad (1)$$

where  $c_k : [-\pi, \pi] \rightarrow \mathbb{C}$  ( $c_0$  is called the *spectral symbol* and  $c_k, k > 0$  are called *higher order symbols*) and the errors  $E_{j,n,\alpha} = \mathcal{O}(h^{\alpha+1})$ .

Hence, if we know these functions  $c_k(\theta)$ , or approximate them since they are often not known analytically, we can accurately (and very fast) approximate some (or all) of the eigenvalues of any matrix  $A_n$  simply by evaluating (1).

It was previously shown (under appropriate assumptions, [4, 5]) [1, 7, 8, 9, 10] that for Hermitian sequences  $\{A_n\}_n$ , where  $c_0$  is known, that we can approximate  $c_k(\theta_{j,n_0}), k = 1, \dots, \alpha$  at specified grid points  $\theta_{j,n_0}$  using so-called *matrix-less* methods. The name is derived from the fact that the spectrum for any matrix  $A_n$  in the sequence  $\{A_n\}_n$  can be approximated by (1) *without ever constructing the matrix*; only the spectrum of a few small matrices have to be computed. That is, we have equality in (1), up to machine precision, for some chosen  $n = n_0$  and  $\alpha$ . These approximations  $c_k(\theta_{j,n_0})$  can then be used for interpolation-extrapolation to any grid  $\theta_{j,n}$  (for any  $n$ ) to approximate  $\lambda_j(A_n)$ .

In the current presentation, mainly inspired by [3], but also [6, 11], we extend the previous algorithms with two important features:

1. The function  $c_0$  is not needed as an input and is approximated; this is necessary for most non-Hermitian matrix sequences, but also for discretizations of problems with variable coefficients.
2. The algorithm can handle discretizations of variable coefficient problems, e.g.,  $(a(x)u'(x))'$ .

We here briefly present these two new features.

## 1. No knowledge of $c_0$ necessary.

We begin by presenting two simple but representative pure Toeplitz matrix sequences; one Hermitian  $\{T_n(f_1)\}_n$  and one non-Hermitian  $\{T_n(f_2)\}_n$ .

$$f_1(\theta) = 6 - 8 \cos(\theta) + 2 \cos(2\theta)$$

$$T_n(f_1) = \begin{bmatrix} 6 & -4 & 1 \\ -4 & 6 & -4 \\ 1 & -4 & 6 & -4 & 1 \\ \ddots & \ddots & \ddots & \ddots & \ddots \\ & 1 & -4 & 6 & -4 & 1 \\ & & 1 & -4 & 6 & -4 \\ & & & 1 & -4 & 6 \end{bmatrix}$$

$$f_2(\theta) = -e^{i\theta} + 3 - 3e^{-i\theta} + e^{-2i\theta}$$

$$T_n(f_2) = \begin{bmatrix} 3 & -3 & 1 \\ -1 & 3 & -3 \\ -1 & -1 & 3 & -3 & 1 \\ \ddots & \ddots & \ddots & \ddots & \ddots \\ & -1 & 3 & -3 & 1 \\ & & -1 & 3 & -3 \\ & & & -1 & 3 \end{bmatrix}$$

For matrices in the Hermitian sequence  $\{T_n(f_1)\}_n$ , we have that the eigenvalues can be approximated by  $\lambda_j(T_n(f_1)) \approx c_0(\theta_{j,n}) = f_1(\theta_{j,n})$  where  $\theta_{j,n} = j\pi/(n+1)$ ; the spectral symbol  $c_0$  is known and equal to  $f_1$ .

For matrices in the non-Hermitian sequence  $\{T_n(f_2)\}_n$ , we have that  $\lambda_j(T_n(f_1)) \not\approx f_1(\theta_{j,n})$ , we only know that the eigenvalues lie in the convex hull of the complex valued function  $f_2$ ; the spectral symbol  $c_0$  is not equal to  $f_2$ . For most non-Hermitian matrix sequences the  $c_0$  is not known analytically, and the new matrix-less method presented in [3, 11] does not require it to be known. However, the matrix-less method is more efficient and accurate if it is provided.

**Remark 1** In the specific case of a non-Hermitian sequence  $\{T_n(f_2)\}_n$  presented above we do know that the spectrum is real and there are many viable  $c_0$ , e.g.  $c_0(\theta) = \sin^3(\theta)/(\sin(\theta/3)\sin^2(2\theta/3))$ ; see [13] for details.

For clarity we show a Julia implementation below on how to compute a matrix  $C = [c_k(\theta_{j,n_0})]_{k,j=1}^{\alpha+1,n_0}$ ; the inputs are  $n_0$  ( $\approx 100$ ),  $\alpha$  ( $\approx 3$ ) and `eigfun` (a function that returns an ordered set of eigenvalues  $\lambda_j(A_n)$  for a matrix  $A_n$  in  $\{A_n\}_n$ ).

```
function compute_C(n_0, α, eigfun)
    hs = zeros(α+1)
    E = zeros(α+1, n_0)
    for kk = 1:α+1
        nk = 2^(kk-1)*(n_0+1)-1
        jk = 2^(kk-1)*(1:n_0)
        hs[kk] = 1/(nk+1)
        E[kk,:] = eigfun(nk)[jk]
    end
    V=[hs[ii]^(jj-1) for ii=1:α+1, jj=1:α+1]
    return C=V\E
end
```

As is seen above, the algorithm relies on the computed spectrum for  $\alpha + 1$  small matrices (of sizes  $n_k = 2^{k-1}(n_0 + 1) - 1$ , for  $k = 1, \dots, \alpha + 1$ ) to compute the elements of  $C$ . Subsequently  $c_k(\theta_{j,n})$  is approximated, using interpolation-extrapolation, for arbitrary  $n$ , and used in (1) to approximate  $\lambda_j(A_n)$ .

**Remark 2** If the spectral symbol is non-monotone (e.g., the stiffness matrix for IgA or  $f(\theta) = 6 - 8\cos(\theta) + 4\cos(2\theta)$ ), the matrix-less method does typically not work in the non-monotone region, since we usually do not know how to order the eigenvalues correctly.

## 2. Variable coefficients.

The spectral symbol  $f$  of the 2nd order FD discretization of  $(a(x)u'(x))'$  is two-dimensional, namely  $f(x, \theta) = a(x)(2 - 2\cos(\theta))$ , where  $f : [0, 1] \times [-\pi, \pi] \rightarrow \mathbb{C}$ ; e.g., see [12].

In [3] we show that we can use the rearranged symbol [2] to compute an expansion (1) for discretizations of variable coefficient problems; i.e., we map the function  $f : [0, 1] \times [-\pi, \pi] \rightarrow \mathbb{R}$  to a rearranged symbol  $g : [0, 1] \rightarrow \mathbb{R}$ . In the new matrix-less method we have  $c_0 = g$ .

**Remark 3** We emphasize that the class of problems and matrices where this approach can be applied is extensive, e.g.,

- multi-dimensional problems (size of matrices are then  $d_{\mathbf{n}}(\mathbf{n})$  and not  $n$ );

- *block matrices (e.g., FE/FV/DG);*
- *boundary conditions ( $c_0$  is the same,  $c_k, k > 0$  changes);*
- *h-dependence, space-time, sums/inverses/products;*
- *eigenvectors, singular values, generalized eigenvalue problems;*

*and the approach could also be used to construct preconditioners and other solver techniques.*

Apart from the two main points mentioned above, we will also discuss the current framework in detail, possible extension and current developments, and possible applications.

## References

- [1] Fayyaz Ahmad, Eman Salem Al-Aidarous, Dina Abdullah Alrehaili, Sven-Erik Ekström, Isabella Furci, and Stefano Serra-Capizzano, *Are the eigenvalues of preconditioned banded symmetric Toeplitz matrices known in almost closed form?*, Numerical Algorithms, 78(3), pp. 867-893 (2018)
- [2] Giovanni Barbarino, Davide Bianchi, and Carlo Garoni, *Constructive approach to the monotone rearrangement of functions*, Expositiones Mathematicae, 40(1), pp. 155–175 (2022)
- [3] Giovanni Barbarino, Melker Claesson, Sven-Erik Ekström, Carlo Garoni, David Meadon, and Hendrik Speleers, *Matrix-less spectral approximation for large structured matrices*, BIT Numerical Mathematics, (in press, 2024) DOI:10.1007/s10543-024-01041-w
- [4] Mauricio Barrera, Albrecht Böttcher, Sergei M. Grudsky, and Egor A. Maximenko, *Eigenvalues of even very nice Toeplitz matrices can be unexpectedly erratic*, In: Böttcher, A., Potts, D., Stollmann, P., Wenzel, D. (eds) The Diversity and Beauty of Applied Operator Theory. Operator Theory: Advances and Applications, vol 268. Birkhäuser, Cham. 2018
- [5] Manuel Bogoya, Sven-Erik Ekström, and Stefano Serra-Capizzano, *Fast Toeplitz eigenvalue computations, joining interpolation-extrapolation matrix-less algorithms and simple-loop theory*, Numerical Algorithms, 91, pp. 1653-1676 (2022)
- [6] Manuel Bogoya, Sven-Erik Ekström, Stefano Serra-Capizzano, and Paris Vassalos, *Matrix-less methods for the spectral approximation of large non-Hermitian Toeplitz matrices: A concise theoretical analysis and a numerical study*, Numerical Linear Algebra with Applications, 34(4), pp. e2545 (2024)
- [7] Sven-Erik Ekström, Isabella Furci, Carlo Garoni, Carla Manni, Stefano Serra-Capizzano, and Hendrik Speleers, *Are the eigenvalues of the B-spline isogeometric analysis approximation of  $-\Delta u = \lambda u$  known in almost closed form?*, Numerical Linear Algebra with Applications, 25(5), pp. e2198 (2018)
- [8] Sven-Erik Ekström, Isabella Furci, and Stefano Serra-Capizzano, *Exact formulae and matrix-less eigensolvers for block banded symmetric Toeplitz matrices*, BIT Numerical Mathematics, 58(4), pp. 937-968 (2018)

- [9] Sven-Erik Ekström and Carlo Garoni, *A matrix-less and parallel interpolation-extrapolation algorithm for computing the eigenvalues of preconditioned banded symmetric Toeplitz matrices*, Numerical Algorithms, 80(3), pp. 819-848 (2019)
- [10] Sven-Erik Ekström, Carlo Garoni, and Stefano Serra-Capizzano, *Are the Eigenvalues of Banded Symmetric Toeplitz Matrices Known in Almost Closed Form?*, Experimental Mathematics, 27(4), pp. 478-487 (2018)
- [11] Sven-Erik Ekström and Paris Vassalos, *A Matrix-Less Method to Approximate the Spectrum and the Spectral Function of Toeplitz Matrices with Real Eigenvalues*, Numerical Algorithms, 89, pp. 701-720 (2022)
- [12] Carlo Garoni and Stefano Serra-Capizzano, *Generalized Locally Toeplitz Sequences: Theory and Applications: Volume I, Volume 1*, Springer, 2017.
- [13] Boris Shapiro, František Štampach, *Non-selfadjoint Toeplitz matrices whose principal submatrices have real spectrum*, Constructive Approximation, 49(2), pp. 191–226 (2019)

# Spectral Computations for Quasicrystal Models

*Mark Embree, Matthew J. Colbrook, David Damanik, Jake Fillman,  
Anton Gorodetski, May Mei, Charles Puelz*

## Abstract

Mathematical models of aperiodic materials – quasicrystals – lead to fascinating problems in spectral theory that can test the limits of conventional approaches to eigenvalue computation. Quasicrystals are exotic objects that were discovered in the 1980s by Dan Schechtman, who was recognized with the 2011 Nobel Prize in Chemistry.

The periodic structure of conventional crystals gives rise to Schrödinger operators whose spectra consist of the union of real intervals, which are neatly characterized by Floquet–Bloch theory. At the other extreme, disordered materials lead to random Schrödinger operators that typically have eigenvectors whose entries exponentially decay from some central site (“Anderson localization”). Sitting between these extremes, quasicrystal models are deterministic but not periodic, and the associated self-adjoint linear operators often exhibit intriguing spectral structure. For example, the spectrum can be a zero-measure Cantor set (a closed set that contains its limit points but no intervals). How can one approach such problems using tools from numerical linear algebra?

In this talk we will survey several problems that arise in the computational study of quasicrystals, highlighting the motivating questions, describing algorithmic approaches, and showing numerical results, based on [2, 3, 4, 7]. We focus on three general problems.

- *Approximating the spectrum of the Fibonacci Hamiltonian.* The most carefully studied quasicrystal model is the Fibonacci Hamiltonian  $H : \ell^2(\mathbf{Z}) \rightarrow \ell^2(\mathbf{Z})$ , defined for each site  $k \in \mathbf{Z}$  by the difference equation

$$(Hx)_k = x_{k-1} + V_k x_k + x_{k+1}, \quad V_k = \begin{cases} 0, & k\alpha \bmod 1 \in [0, 1-\alpha); \\ \lambda, & k\alpha \bmod 1 \in [1-\alpha, 1); \end{cases}$$

for the irrational  $\alpha = (\sqrt{5}-1)/2 = 0.6180\dots$  (the reciprocal of the golden ratio); see [4, 5] for summaries of key results. In 1987, Sütő [8] proved that the spectrum is a zero-measure Cantor set for all  $\lambda > 0$ , which one can approximate by replacing  $\alpha$  with rational approximations given by the ratio of successive Fibonacci numbers. With such approximations the potential  $\{V_k\}$  becomes periodic, and the resulting spectrum follows from Floquet–Bloch theory. Specifically, if  $\{V_k\}$  has period  $p$ , then the spectrum of  $H$  is the union of  $p$  real intervals whose end points are eigenvalues of two  $p \times p$  symmetric tridiagonal matrices plus corner entries:

$$J_{\pm}^{(p)} = \begin{bmatrix} V_1 & 1 & & \pm 1 \\ 1 & V_2 & 1 & \\ & 1 & V_3 & \ddots \\ & & \ddots & \ddots & 1 \\ \pm 1 & & 1 & V_p \end{bmatrix}.$$

To study the Cantor spectrum of the Fibonacci Hamiltonian demands approximations with very large period  $p$ , giving intervals so narrow that the computed eigenvalues of  $J_+^{(p)}$  and  $J_-^{(p)}$  can violate their theoretical ordering properties. For this reason we propose these examples

as physically-motivated test matrices for symmetric eigensolvers. (Indeed, these models can exhibit similar behavior to Wilkinson’s famous  $W_{21}^+$  example [9, p. 309].)

We will describe the numerical linear algebra challenges associated with the approximation of the Cantor spectrum of the Fibonacci model and several other aperiodic models derived from substitution rules (period-doubling and Thue–Morse) [7].

- *Quantities derived from Cantor spectra.*

The computation of the spectrum is often the first step in a more elaborate process. For example, one can gain physical insight from the fractal (box-counting and Hausdorff) dimension of the spectrum of the Schrödinger operator. How can one use estimates to the spectrum to approximate these quantities? Simple two- and three-dimensional quasicrystal models follow from coupling one-dimensional models on a square or cubic lattice. The resulting spectra are now *sums of Cantor sets*, which could potentially be intervals, Cantor sets, or more exotic sets called *Canvorvals*. We will discuss computational approaches and obstacles for such problems [2, 4, 7], using the perspective of the Solvability Complexity Index [1].

- *Locally supported eigenvectors of the graph Laplacian for Penrose tilings.*

A different class of two-dimension quasicrystal models derive their structure from aperiodic tilings of the plane, such as the Penrose or Ammann–Beenker constructions. From a finite section of such tiling we construct a graph, and then study spectral properties of the associated graph Laplacian. Generalizing work from the physics literature [6], we show that a variety of Penrose models exhibit eigenvectors that are nonzero only on a small (repeating) pattern of tiles, and thus arise with high algebraic multiplicity. We illustrate these configurations, and discuss some associated numerical challenges (finding sparse bases for the invariant subspaces, predicting eigenvalue multiplicity as the tiling grows, identifying gaps in the spectrum) [2, 3].

## References

- [1] M.J. COLBROOK, *On the computation of geometric features of spectra of linear operators on Hilbert spaces*, Found. Comput. Math. 24 (2024) 723–804.
- [2] M.J. COLBROOK, M. EMBREE, J. FILLMAN, *Optimal algorithms for quantifying spectral size with applications to quasicrystals*, preprint: arXiv:2407.20353.
- [3] D. DAMANIK, M. EMBREE, J. FILLMAN, M. MEI, *Discontinuities of the integrated density of states for Laplacians associated with Penrose and Ammann–Beenker tilings*, Exp. Math., to appear (preprint: arXiv:2209.01443).
- [4] D. DAMANIK, M. EMBREE, A. GORODETSKI, *Spectral properties of Schrödinger operators arising in the study of quasicrystals*, In *Mathematics of Aperiodic Order*, p. 307–370; J. Kellendonk, D. Lenz, J. Savinien, eds., Springer, 2015.
- [5] D. DAMANIK, A. GORODETSKI, W. YESSEN, *The Fibonacci Hamiltonian*, Invent. Math. 206 (2016) 629–692.
- [6] T. FUJIWARA, M. ARAI, T. TOKIHIRO, M. KOHMOTO, *Localized states and self-similar states of electrons on a two-dimensional Penrose lattice*, Phys. Rev. B 37 (1988) 2797–2804.
- [7] C. PUELZ, M. EMBREE, AND J. FILLMAN, *Spectral approximation for quasiperiodic Jacobi operators*, Integral Equations Operator Theory 82 (2015) 533–554.
- [8] A. SÜTŐ, *The spectrum of a quasiperiodic Schrödinger operator*, Comm. Math. Phys. 111 (1987) 409–415.
- [9] J.H. WILKINSON, *The Algebraic Eigenvalue Problem*, Oxford University Press, Oxford, 1965.

# Randomly Pivoted Cholesky: Near-Optimal Positive Semidefinite Low-Rank Approximation from a Small Number of Entry Evaluations

Ethan N. Epperly, Yifan Chen, Joel A. Tropp, Robert J. Webber

## Abstract

This talk describes randomly pivoted Cholesky (RPCHOLESKY), a randomized algorithm for computing a low-rank approximation to a Hermitian positive semidefinite (psd) matrix. To compute a rank- $k$  approximation to an  $N \times N$  matrix, RPCHOLESKY performs a  $k$ -step partial Cholesky decomposition with a pivot entry randomly chosen with probabilities proportional to diagonal entries of the current residual matrix (i.e., Schur complement). The algorithm requires  $\mathcal{O}(k^2N)$  operations and reads only  $(k + 1)N$  entries of the input matrix.

The RPCHOLESKY method has an interesting history. The existence of the method was briefly noted in a 2017 paper of Musco and Woodruff [9], and it is algebraically related to an earlier “randomly pivoted QR” algorithm of Deshpande, Rademacher, Vempala, and Wang (2006, [3]). Our paper [2], originally released in 2022, reintroduced the algorithm, described its connection to Cholesky decomposition, evaluated the method numerically, and provided new theoretical results.

Surprisingly, this simple algorithm is guaranteed to produce a near-optimal low-rank approximation. The output of RPCHOLESKY, and any other partial Cholesky decomposition, is low-rank approximation of the form

$$\widehat{\mathbf{A}} = \mathbf{A}(:, \mathbf{S}) \mathbf{A}(\mathbf{S}, \mathbf{S})^\dagger \mathbf{A}(\mathbf{S}, :),$$

where  $\mathbf{S}$  denotes the set of pivots selected by the algorithm and  $^\dagger$  denotes the Moore–Penrose pseudoinverse. This type of low-rank approximation is known as a *(column) Nyström approximation* and is used widely to accelerate kernel machine learning methods. It is known [7] that  $k \geq r/\varepsilon$  columns  $\mathbf{S}$  are needed to produce a Nyström approximation  $\widehat{\mathbf{A}}$  within a  $1 + \varepsilon$  multiplicative factor of the best rank- $r$  approximation  $\llbracket \mathbf{A} \rrbracket_r$ , i.e.,

$$\|\mathbf{A} - \widehat{\mathbf{A}}\|_* \leq (1 + \varepsilon) \|\mathbf{A} - \llbracket \mathbf{A} \rrbracket_r\|_*.$$

Here,  $\|\cdot\|_*$  denotes the trace norm. In [2], we showed that RPCHOLESKY achieves the guarantee:

$$\mathbb{E} [\|\mathbf{A} - \widehat{\mathbf{A}}\|_*] \leq (1 + \varepsilon) \|\mathbf{A} - \llbracket \mathbf{A} \rrbracket_r\|_* \quad \text{when } k \geq \frac{r}{\varepsilon} + r \log \left( \frac{1}{\varepsilon \eta} \right).$$

Here,  $\widehat{\mathbf{A}}$  is the approximation produced by  $k$  steps of RPCHOLESKY and  $\eta = \|\mathbf{A} - \llbracket \mathbf{A} \rrbracket_r\|_*/\|\mathbf{A}\|_*$  denotes the relative error of the best rank- $r$  approximation. In expectation, RPCHOLESKY achieves the optimal scaling  $k = r/\varepsilon$  up to an additive term that is logarithmic in the relative error  $\eta$ .

RPCHOLESKY has proven effective at accelerating kernel machine learning methods. Given a data set  $\mathbf{x}_1, \dots, \mathbf{x}_N$ , kernel methods perform machine learning tasks such as regression and clustering by manipulating a psd kernel matrix  $\mathbf{A} = (\kappa(\mathbf{x}_i, \mathbf{x}_j))_{1 \leq i, j \leq N}$ , where  $\kappa$  is a given positive definite kernel function. When implemented directly, kernel methods require  $\mathcal{O}(N^3)$  operations and  $\mathcal{O}(N^2)$  storage. By replacing  $\mathbf{A}$  with a low-rank approximation  $\widehat{\mathbf{A}}$  (say, of rank  $k = \mathcal{O}(1)$ ), the storage and runtime costs of these methods are reduced to  $\mathcal{O}(N)$ . This talk will present numerical experiments from [2], which show that an RPCHOLESKY-accelerated clustering method can be  $9\times$  to  $14\times$  more accurate than accelerated clustering methods using other low-rank approximation techniques. Subsequent papers have applied RPCHOLESKY to accelerate learning of committer functions in

biochemistry [1], as a preconditioner for conjugate gradient [4], for quadrature in reproducing kernel Hilbert spaces [5], and compression of data sets [8].

While the standard version of RPCHOLESKY is already fast, it is slower than it could be because it processes the columns of the input matrix one-by-one. A blocked version of the method is faster, but can produce approximations of lower accuracy. This talk will conclude by discussing the recently introduced *accelerated RPCHOLESKY method* [6], which simulates the performance of original RPCHOLESKY using a combination of rejection sampling and block-wise computations. The accelerated RPCHOLESKY method can be up to  $40\times$  faster than the original method while producing the same random output (in exact arithmetic).

## References

- [1] David Aristoff, Mats Johnson, Gideon Simpson, and Robert J. Webber. The fast committor machine: Interpretable prediction with kernels. *The Journal of Chemical Physics*, 161(8):084113, 2024.
- [2] Yifan Chen, Ethan N. Epperly, Joel A. Tropp, and Robert J. Webber. Randomly pivoted Cholesky: Practical approximation of a kernel matrix with few entry evaluations. *Communications on Pure and Applied Mathematics*, accepted, 2024.
- [3] Amit Deshpande, Luis Rademacher, Santosh Vempala, and Grant Wang. Matrix approximation and projective clustering via volume sampling. In *Proceedings of the 2006 Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1117–1126, 2006.
- [4] Mateo Díaz, Ethan N. Epperly, Zachary Frangella, Joel A. Tropp, and Robert J. Webber. Robust, randomized preconditioning for kernel ridge regression. *arXiv preprint arXiv:2304.12465*, 2024.
- [5] Ethan N. Epperly and Elvira Moreno. Kernel quadrature with randomly pivoted Cholesky. In *Advances in Neural Information Processing Systems 36*, 2023.
- [6] Ethan N. Epperly, Joel A. Tropp, and Robert J. Webber. Embrace rejection: Kernel matrix approximation by accelerated randomly pivoted Cholesky. *arXiv preprint arXiv:2410.03969*, 2024.
- [7] Venkatesan Guruswami and Ali Kemal Sinop. Optimal column-based low-rank matrix reconstruction. In *Proceedings of the 2012 Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1207–1214. 2012.
- [8] Lingxiao Li, Raaz Dwivedi, and Lester Mackey. Debiased distribution compression. *arXiv preprint arXiv:2404.12290*, 2024.
- [9] C. Musco and D. P. Woodruff. Sublinear Time Low-Rank Approximation of Positive Semidefinite Matrices. In *2017 IEEE Annual Symposium on Foundations of Computer Science*, pages 672–683, 2017.

# Variable Projection Methods for Regularized Separable Nonlinear Inverse Problems

*Malena I. Español and Gabriela Jeronimo*

## Abstract

We consider discrete ill-posed inverse problems of the form

$$\mathbf{A}(\mathbf{y})\mathbf{x} \approx \mathbf{b} = \mathbf{b}_{\text{true}} + \epsilon \quad \text{with } \mathbf{A}(\mathbf{y}_{\text{true}})\mathbf{x}_{\text{true}} = \mathbf{b}_{\text{true}}, \quad (1)$$

where the vector  $\mathbf{b}_{\text{true}} \in \mathbb{R}^m$  denotes an unknown error-free vector associated with available data and  $\epsilon \in \mathbb{R}^m$  is an unknown vector that represents the noise/errors in  $\mathbf{b}$ . The matrix  $\mathbf{A}(\mathbf{y}) \in \mathbb{R}^{m \times n}$  with  $m \geq n$  models a forward operator and is typically severely ill-conditioned. We assume that  $\mathbf{A}$  is unknown but can be parametrized by a vector  $\mathbf{y} \in \mathbb{R}^r$  with  $r \ll n$  in such a way that the map  $\mathbf{y} \mapsto \mathbf{A}(\mathbf{y})$  is differentiable. We aim to compute good approximations of  $\mathbf{x}_{\text{true}}$  and  $\mathbf{y}_{\text{true}}$ , given a data vector  $\mathbf{b}$  and a matrix function that maps the unknown vector  $\mathbf{y}$  to an  $m \times n$  matrix  $\mathbf{A}$ . To accomplish this task, we could solve

$$\min_{\mathbf{x}, \mathbf{y}} \frac{1}{2} \|\mathbf{A}(\mathbf{y})\mathbf{x} - \mathbf{b}\|_2^2 + \frac{\lambda^2}{2} \|\mathbf{L}\mathbf{x}\|_2^2, \quad (2)$$

where  $\lambda > 0$  is a *regularization parameter* and  $\mathbf{L} \in \mathbb{R}^{q \times n}$  is a *regularization operator*. We assume that  $\mathbf{L}$  satisfies that  $\mathcal{N}(\mathbf{A}(\mathbf{y})) \cap \mathcal{N}(\mathbf{L}) = \{0\}$  for all feasible values of  $\mathbf{y}$ , so that the minimization problem (2) has a unique solution for  $\mathbf{y}$  fixed. We call problems of the form (2) regularized *separable nonlinear inverse problems* since the observations depend nonlinearly on the vector of unknown parameters  $\mathbf{y}$  and linearly on the solution vector  $\mathbf{x}$ .

The Variable Projection (VarPro) method was originally developed in the 1970s by Golub and Pereyra [3] to solve (2) for  $\lambda = 0$  and has been widely recognized for its efficiency in solving separable nonlinear least squares problems. VarPro eliminates the linear variables  $\mathbf{x}$  through projection and reduces the original problem to a smaller nonlinear least squares problem in the parameters  $\mathbf{y}$ . This reduced nonlinear least squares problem can be solved using the Gauss-Newton Method.

In [1], Español and Pasha extended VarPro to solve inverse problems with general-form Tikhonov regularization for general matrices  $\mathbf{L}$ . They named this method GenVarPro. For special cases where computing the generalized singular value decomposition (GSVD) of the pair  $\{\mathbf{A}(\mathbf{y}), \mathbf{L}\}$  for a fixed value of  $\mathbf{y}$  is feasible or a joint spectral decomposition exists, they provided efficient ways to compute the Jacobian matrix and the solution of the linear subproblems. For large-scale problems, where matrix decompositions are not an option, they proposed computing a reduced Jacobian and applying projection-based iterative methods and generalized Krylov subspace methods to solve the linear subproblems. Following on this theme, Español and Jeronimo introduced in [2], the Inexact-GenVarPro that considers a new approximate Jacobian where iterative methods such as LSQR and LSMR are used to solve the linear subproblems. Furthermore, specific stopping criteria were proposed to ensure Inexact-GenVarPro's local convergence.

In this talk, we will show how to extend GenVarPro and Inexact-GenVarPro to solve

$$\min_{\mathbf{x}, \mathbf{y}} \frac{1}{2} \|\mathbf{A}(\mathbf{y})\mathbf{x} - \mathbf{b}\|_2^2 + \frac{\lambda^2}{2} \|\mathbf{L}\mathbf{x}\|_2^2 + \mu \mathcal{R}(\mathbf{y}), \quad (3)$$

where  $\mu > 0$  is another regularization parameter and  $\mathcal{R}(\mathbf{y})$  plays the role of regularization on the parameter vector  $\mathbf{y}$ . Similar variational formulations have appeared in recent papers in the context

of training neural networks [4] and computerized tomographic reconstruction [5]. We will motivate the need to incorporate this regularization term on  $\mathbf{y}$  in the context of a semi-blind image deblurring problem by showing some examples where, without it, the solution of the reduced problem does not exist (i.e., no minimizer exists) or is trivial (e.g.,  $\mathbf{y} = \mathbf{0}$  and  $\mathbf{A}(\mathbf{y})$  becomes the identity matrix). We will show in particular, how to extend GenVarPro and Inexact-GenVarPro to the case when  $\mathcal{R}(\mathbf{y}) = \|\mathbf{y} - \mathbf{y}_0\|_2^2$  and  $\mathcal{R}(\mathbf{y}) = -\sum_j \log(y_j)$  in the context of a large-scale semi-blind image deblurring problem. Furthermore, we will present theoretical results with sufficient conditions on the matrices involved to ensure local convergence. Numerical experiments will also be presented to illustrate their efficiency and confirm the theoretical results.

## References

- [1] M. I. Español and M. Pasha. Variable projection methods for separable nonlinear inverse problems with general-form Tikhonov regularization. *Inverse Problems*, 39(8):084002, 2023.
- [2] M. I. Español and G. Jeronimo. Convergence analysis of a variable projection method for regularized separable nonlinear inverse problems. *arXiv preprint arXiv:2402.08568*, 2024.
- [3] G. H. Golub and V. Pereyra. The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate. *SIAM Journal on Numerical Analysis*, 10(2):413–432, 1973.
- [4] E. Newman, L. Ruthotto, J. Hart, and B. van Bloemen Waanders. Train like a (Var) Pro: Efficient training of neural networks with variable projection. *SIAM Journal on Mathematics of Data Science*, 3(4):1041–1066, 2021.
- [5] F. Uribe, J. M. Bardsley, Y. Dong, P. C. Hansen, and N. A. B. Riis. A hybrid Gibbs sampler for edge-preserving tomographic reconstruction with uncertain view angles. *SIAM/ASA Journal on Uncertainty Quantification*, 10(3):1293–1320, 2022.

# Bayesian Optimal Experiment Design via Column Subset Selection

*Srinivas Eswar, Amit N. Subrahmanya, Vishwas Rao, Arvind K. Saibaba*

## Abstract

Inverse problems involve the process of calculating parameters of a mathematical model from observational data [3]. Often these problems are ill-posed and a Bayesian approach is used to produce a posterior distribution for the unobservable parameters. A key question is “how best to acquire data” in such a setting. We consider the case of Bayesian linear inverse problems where there are  $m$  candidate sensor locations, and we need to pick the  $k$  “best” ones.

Consider the measurement equation

$$\mathbf{d} = \mathbf{F}\mathbf{m} + \boldsymbol{\epsilon}, \quad (1)$$

where  $\mathbf{d} \in \mathbb{R}^m$  is the data,  $\mathbf{F} \in \mathbb{R}^{m \times n}$  is the mathematical model, and  $\mathbf{m} \in \mathbb{R}^n$  is the parameter to be reconstructed. The observations are assumed to be perturbed with additive uncorrelated Gaussian noise, i.e.  $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Gamma}_{\text{noise}})$ . We assume that  $m < n$ , which makes the problem underdetermined. If we assume our prior to also be Gaussian,  $\mathbf{m} \sim \mathcal{N}(\boldsymbol{\mu}_{\text{pr}}, \boldsymbol{\Gamma}_{\text{pr}})$ , the posterior will also be a Gaussian with covariance  $\boldsymbol{\Gamma}_{\text{post}} = (\mathbf{F}^\top \boldsymbol{\Gamma}_{\text{noise}}^{-1} \mathbf{F} + \boldsymbol{\Gamma}_{\text{pr}}^{-1})^{-1}$  and mean  $\boldsymbol{\mu}_{\text{post}} = \boldsymbol{\Gamma}_{\text{post}}(\mathbf{F}^\top \boldsymbol{\Gamma}_{\text{noise}}^{-1} \mathbf{d} + \boldsymbol{\Gamma}_{\text{pr}}^{-1} \boldsymbol{\mu}_{\text{pr}})$ .

The rows of  $\mathbf{F}$  correspond to the  $m$  different candidate sensor locations and we would like to select only  $k$  locations to collect data. To determine the optimal sensor placements, we solve the following combinatorial optimization problem

$$\min_{W \subset \{1, \dots, m\}} \phi(W), \quad \text{subject to } |W| \leq k. \quad (2)$$

Here  $\phi(W)$  is a set-valued function which determines the quality of the sensor placement. In this work we focus on the A-optimality criterion, which minimizes average posterior variance, and D-optimality, which measures the information gain from the prior to the posterior. These criteria amounts to measuring the trace and log-determinant of the posterior covariance matrices respectively. For the current problem, these criteria take the form

$$\phi_A(W) = \text{trace} \left( \boldsymbol{\Gamma}_{\text{pr}}^{1/2} \left( \mathbf{I} + \mathbf{C}\mathbf{C}^\top \right)^{-1} \boldsymbol{\Gamma}_{\text{pr}}^{1/2} \right) \quad \text{and} \quad \phi_D(W) = -\log \det \left( \mathbf{I} + \mathbf{C}\mathbf{C}^\top \right), \quad (3)$$

where  $\mathbf{C} = \mathbf{A}(:, W)$  are the columns of an appropriately formed matrix indexed by  $W$ . Here  $\mathbf{A} := \boldsymbol{\Gamma}_{\text{pr}}^{1/2} \mathbf{F}^\top \boldsymbol{\Gamma}_{\text{noise}}^{-1/2} \in \mathbb{R}^{n \times m}$  is the prior-preconditioned forward operator and selecting  $k$  columns is akin to selecting sensors. Note that we use  $\phi(W)$  and  $\phi(\mathbf{C})$  interchangeably.

Assuming the following partitioned SVD of  $\mathbf{A}$  with  $1 \leq k \leq m$ ,

$$\mathbf{A} = [\mathbf{U}_k \quad \mathbf{U}_\perp] \begin{bmatrix} \boldsymbol{\Sigma}_k & \\ & \boldsymbol{\Sigma}_\perp \end{bmatrix} [\mathbf{V}_k \quad \mathbf{V}_\perp]^\top.$$

Now our structural bounds are for column selection of the form  $\mathbf{A}\boldsymbol{\Pi} = [\mathbf{A}\boldsymbol{\Pi}_1 \quad \mathbf{A}\boldsymbol{\Pi}_2] = [\mathbf{C} \quad \mathbf{T}]$  with an identical permutation of the truncated right singular vectors  $\mathbf{V}_k^\top \boldsymbol{\Pi} = [\mathbf{V}_{11} \quad \mathbf{V}_{12}]$ .

**Theorem 1** [1] *Let  $\mathbf{A} \in \mathbb{R}^{n \times m}$  with  $k \leq \text{rank}(\mathbf{A})$ . Then for any permutation  $\boldsymbol{\Pi}$  such that  $\text{rank}(\mathbf{V}_{11}) = k$  and  $\mathbf{A}\boldsymbol{\Pi} = [\mathbf{C} \quad \mathbf{T}]$  we have,*

$$\frac{\sigma_i(\mathbf{A})}{\|\mathbf{V}_{11}^{-1}\|_2} \leq \sigma_i(\mathbf{C}) \leq \sigma_i(\mathbf{A}), \quad 1 \leq i \leq k.$$

The bounds on individual singular values of  $\mathbf{C}$  are key to obtaining bounds and algorithms for the different OED objectives. Let  $\mathbf{C}_D^{\text{opt}}$  denote the optimal selection for the D-optimality criteria (respectively  $\mathbf{C}_A^{\text{opt}}$  for A-optimality). Then utilizing Theorem 1, we can see that

$$\begin{aligned} \phi_D(\mathbf{A}) &\leq \phi_D(\boldsymbol{\Sigma}_k) \leq \phi_D(\mathbf{C}_D^{\text{opt}}) \leq \phi_D(\mathbf{C}) \leq \phi_D\left(\boldsymbol{\Sigma}_k / \|\mathbf{V}_{11}^{-1}\|_2\right) \text{ and} \\ \frac{t(\boldsymbol{\Sigma}_k) + (n - k)}{\|\mathbf{\Gamma}_{\text{pr}}^{-1}\|_2} &\leq \phi_A(\mathbf{C}_A^{\text{opt}}) \leq \phi_A(\mathbf{C}) \leq \|\mathbf{\Gamma}_{\text{pr}}\|_2 \left( t\left(\boldsymbol{\Sigma}_k / \|\mathbf{V}_{11}^{-1}\|_2\right) + (n - k) \right), \end{aligned} \quad (4)$$

where  $t(\mathbf{X}) = \sum_{i=1}^{\text{rank}(\mathbf{X})} \frac{1}{1+\sigma_j^2(\mathbf{X})}$ . Not surprisingly, the performance of the selected columns depend on the top- $k$  singular values of  $\mathbf{A}$ . If the discarded singular values,  $\boldsymbol{\Sigma}_{\perp}$ , are not negligible, we cannot expect  $\mathbf{C}^{\text{opt}}$  to be close to  $\mathbf{A}$  in either criterion. Note that the error bounds for the D-optimality case is much cleaner than A-optimality due to the absence of the prior term which factors out as a constant because of the logdet objective. Another point of concern is the presence of the terms with  $n$  for A-optimality, which in principle can be extremely large. This term arises due to the ill-posed nature of the inverse problem and corresponds to the singular values of 1 in  $\mathbf{I}_n + \mathbf{CC}^T$ . These values multiply out for D-optimality but are harder to remove in the A-optimality case prompting the development of relative bounds.

Equation (4) clearly identifies the factor  $\|\mathbf{V}_{11}^{-1}\|_2$  to optimize for in an OED algorithm. Also since  $\mathbf{V}_{11}$  is an invertible submatrix of  $\mathbf{V}_k$ , we have  $\|\mathbf{V}_{11}^{-1}\|_2 \geq 1$ . We wish to make this value as close to 1 as possible by finding a set of  $k$  well-conditioned columns of  $\mathbf{V}_k^T$ . This is exactly the Golub-Klema-Stewart approach for subset selection [4], which we further accelerate using randomized approaches. Inspired by rank-revealing factorizations [2] and exchange algorithms for OED [5], we also investigate column-swapping based methods on model inverse problems.

The explicit connection to column subset selection gives us many avenues for future work. Is it possible to extend our techniques to the correlated noise or to nonlinear problems? Can we reduce the gap to  $\phi(\boldsymbol{\Sigma}_k)$  by combining sensor information in a sensible manner? What if our optimization criteria is some user specified goal?

## References

- [1] Eswar, S., Rao, V. & Saibaba, A. Bayesian D-Optimal Experimental Designs via Column Subset Selection. *ArXiv Preprint ArXiv:2402.16000*. (2024)
- [2] Gu, M. & Eisenstat, S. Efficient algorithms for computing a strong rank-revealing QR factorization. *SIAM Journal On Scientific Computing*. **17**, 848-869 (1996)
- [3] Hansen, P. Rank-deficient and discrete ill-posed problems: numerical aspects of linear inversion. (SIAM,1998)
- [4] Golub, G., Klema, V. & Stewart, G. Rank degeneracy and least squares problems. (Stanford University,1976)
- [5] Fedorov, V. Theory of optimal experiments. (Academic Press,1972)

# High-Accuracy Floating-Point Matrix Multiplication on Low-Precision Floating-Point and Fixed-Point Hardware

Ahmad Abdelfattah, Jack Dongarra, Massimiliano Fasi, Mantas Mikaitis, Fran oise Tisseur

## Abstract

We have officially entered the exascale era. At the forefront is the Frontier supercomputer, topping the June 2024 Top500 list<sup>1</sup> as the first machine capable of performing over  $10^{18}$  operations per second in binary64 (*double precision*) arithmetic. Modern supercomputers achieve their remarkable speeds by leveraging machine-learning hardware accelerators, which deliver extraordinary throughput by trading off some degree of accuracy. While these accelerators currently support binary64 arithmetic, the field is shifting, and soon many will be optimized exclusively for lower precision.

Today, fully utilizing the potential of these accelerators requires relying on low-precision formats: TensorFloat-32, bfloat16, binary16 (*half precision*), E4M3, E5M2, and even compact integer data types, such as INT8. These reduced-precision formats can have a throughput up to two orders of magnitude higher than binary64, but they lack the precision needed for traditional scientific simulations, which require higher accuracy to yield meaningful results.

To integrate GPUs effectively into scientific computing, we must reimagine high-precision computations by strategically applying lower precision when feasible. Here, we explore techniques to reformulate a high-precision matrix multiplication as a series of low-precision operations, and we outline two strategies for assigning different precision levels across computations. Matrix multiplication is a fundamental kernel in scientific computing, and efficient implementations underpin the performance of many algorithms in numerical linear algebra. The techniques we discuss will enable numerical codes to make better use of current accelerators, where the performance gap between low and high precision is widening, and of future ones, where high precision will be missing altogether.

**General scheme for mixed-precision matrix multiplication** Let  $\mathcal{F}_{\text{low}}$  and  $\mathcal{F}_{\text{high}}$  be a low-precision and a high-precision floating-point format, respectively, and let  $u_{\text{low}}$  and  $u_{\text{high}}$  be their unit roundoffs. We consider the computation of  $C = AB \in \mathcal{F}_{\text{high}}^{m \times n}$ , where  $A \in \mathcal{F}_{\text{high}}^{m \times p}$  and  $B \in \mathcal{F}_{\text{high}}^{p \times n}$ . Rows of  $A$  and column of  $B$  with only zeros do not affect the result, thus we assume that each row of  $A$  and column of  $B$  contains at least one nonzero element. The high-precision matrices  $A$  and  $B$  can be written as the unevaluated sum of low-precision matrices

$$A = A^{(1)} + A^{(2)} + \cdots + A^{(s_A)} + \Delta A, \quad B = B^{(1)} + B^{(2)} + \cdots + B^{(s_B)} + \Delta B, \quad (1)$$

where the entries of  $A^{(1)}, A^{(2)}, \dots, A^{(s_A)}, B^{(1)}, B^{(2)}, \dots, B^{(s_B)}$  belong to  $\mathcal{F}_{\text{low}}$ , while  $\Delta A$  and  $\Delta B$  are truncation errors. With the decomposition (1), we can approximate the product as

$$\tilde{C} \approx \sum_{k=1}^{s_A} \sum_{\ell=1}^{s_B} A^{(k)} B^{(\ell)}. \quad (2)$$

In terms of runtime, (2) will achieve good performance if the low-precision matrix products of the form  $A^{(k)} B^{(\ell)}$  are executed on hardware that can efficiently multiply matrices stored in  $\mathcal{F}_{\text{low}}$  and accumulate the result in  $\mathcal{F}_{\text{high}}$ . Two terms contribute to the total error in the approximation  $\tilde{C}$ :

- the *truncation error*  $\Delta AB + A\Delta B$ , which depends on the splitting strategy in (1); and

---

<sup>1</sup><https://www.top500.org/lists/top500/list/2024/06/>

- a rounding error, caused by the matrix products and sums in (2)

**Matrix multiplication using multi-word arithmetic** A natural way to obtain the decomposition (1) is to split  $A$  and  $B$  as sum of low-precision floating-point matrices [4]. This can be accomplished by applying the splitting algorithm:

$$A^{(k)} = \text{fl}_{\text{low}} \left( A - \sum_{t=1}^{k-1} A^{(t)} \right), \quad B^{(\ell)} = \text{fl}_{\text{low}} \left( B - \sum_{t=1}^{\ell-1} B^{(t)} \right), \quad (3)$$

where  $\text{fl}_{\text{low}}(X)$  rounds the entries of the input matrix  $X$  to precision  $\mathcal{F}_{\text{low}}$ . In this case, we can set  $s_A = s_B = s$ , as the final accuracy will be limited by the smaller between  $s_A$  and  $s_B$ .

If the splitting (1) is obtained using (3), and the approximation  $\tilde{C}$  is computed using (2), then [1]

$$|\tilde{C} - C| \leq (2u_{\text{low}}^s + u_{\text{low}}^{2s})|A||B| + (n + s^2 - 1)u_{\text{high}} \sum_{k=1}^s \sum_{\ell=1}^s |A^{(\ell)}||B^{(\ell)}|. \quad (4)$$

For practical choices of  $u_{\text{low}}$  and  $u_{\text{high}}$ , a small value of  $s$ , 2 or 3 say, is sufficient to make the two terms in (4) of similar size. Furthermore, not all  $s^2$  products in (2) need be computed, since the magnitude of the elements of  $A^{(k)}$  and  $B^{(\ell)}$  decreases rapidly as  $k$  and  $\ell$  increase. Ignoring all products of the form  $A^{(k)}B^{(\ell)}$ , for  $k + \ell > s + 1$ , yields a faster algorithm and an error bounded by

$$|\tilde{C} - C| \leq ((s+1)u_{\text{low}}^s + (n + s^2 - 1)u_{\text{high}})|A||B| + \mathcal{O}(u_{\text{high}}u_{\text{low}} + u_{\text{low}}^{s+1}), \quad (5)$$

which is just slightly weaker than (4). We evaluated this scheme using double-binary16 ( $s = 2$  and  $u_{\text{low}} = 2^{-11}$ ) arithmetic to compute binary32 matrix products ( $u_{\text{high}} = 2^{-24}$ ). We run our implementations of the algorithm described above on NVIDIA GPUs equipped with *tensor cores*—mixed-precision units that compute the product of binary16 matrices using binary32 arithmetic. We identified some cases where, surprisingly, double-binary16 fails to achieve binary32 accuracy: this is the case, for example, if the entries of the matrix are drawn from the interval  $(0, 1]$ . This phenomenon does not contradict the bounds (4) and (5), and with the help of probabilistic rounding error analysis we showed that a possible cause is the fact that the tensor cores use a custom rounding mode that is less accurate than round-to-nearest [2]. To support this conclusion, we used the CPFloat library [3] to simulate a variant of the tensor cores that uses round-to-nearest throughout, and we showed that switching between rounding modes has indeed the expected effect on accuracy.

**The Ozaki scheme for matrix multiplication** An alternative technique, which goes back to Rump, Ogita, and Oishi [8], uses a fixed-point representation to recast the matrix product as a sequence of error-free transformations. In the case of matrix multiplication [7], this technique is known as the *Ozaki scheme*. The decomposition (1) is computed using the element-wise algorithm

$$\begin{aligned} a_{ij}^{(k)} &= \text{fl} \left( \text{fl} \left( \alpha_i + \left( a_{ij} - \sum_{t=1}^{k-1} a_{ij}^{(t)} \right) \right) - \alpha_i \right), & \alpha_i &= 2^{\max_{1 \leq j \leq p} \lceil \log_2 |a_{ij}| \rceil + f(a_{ij})}, \\ b_{ij}^{(\ell)} &= \text{fl} \left( \text{fl} \left( \beta_j + \left( b_{ij} - \sum_{t=1}^{\ell-1} b_{ij}^{(t)} \right) \right) - \beta_j \right), & \beta_j &= 2^{\max_{1 \leq i \leq p} \lceil \log_2 |b_{ij}| \rceil + f(b_{ij})}, \end{aligned} \quad (6)$$

where  $f(x)$  returns 1 if  $x$  is a power of two, and 0 otherwise. If the routine computing  $A^{(k)}B^{(\ell)}$  in (2) takes matrices with elements in  $\mathcal{F}_{\text{low}}$  as input but uses precision  $u_{\text{high}}$  internally, then the intermediate precision used by the fl operator in (6) can have at most

$$q = \left\lceil (\log_2 u_{\text{high}}^{-1} - \log_2 p)/2 \right\rceil$$

bits, where  $p$  is the common dimension of  $A$  and  $B$ . This choice of  $q$  ensures that all multiplications of the form  $A^{(k)}B^{(\ell)}$  will be exact.

Implicitly, the algorithm (6) performs two actions. First, it scales all entries in the  $i$ th row of  $A$  by  $\alpha_i^{-1}$ , where  $\alpha_i$  is the smallest power of two that is strictly larger, in magnitude, than all elements in the  $i$ th row of  $A$ ; this ensures that  $\alpha_i^{-1}a_{ij}$  has magnitude in the interval  $[0, 1)$ . Next, each  $\alpha_i^{-1}a_{ij}$  is interpreted as a fixed-point number, and its representation is divided up into  $q$ -bit segments, each assigned to a different low-precision slice  $A^{(k)}$ . The matrix  $B$  is sliced analogously, with the proviso that the algorithm operates by columns rather than by rows. This gives the representation

$$A = \Delta A + \text{diag}(\alpha) \sum_{k=1}^{s_A} 2^{-kq} A^{(k)}, \quad B = \Delta B + \sum_{\ell=1}^{s_B} 2^{-\ell q} B^{(\ell)} \text{diag}(\beta), \quad (7)$$

where  $A^{(1)}, A^{(2)}, \dots, A^{(s_A)}$  and  $B^{(1)}, B^{(2)}, \dots, B^{(s_B)}$  are slices of a fixed-point representation of the elements in  $A$  and  $B$ . Since  $\alpha_i$  and  $\beta_j$  depend on the magnitude of the largest entry in row  $i$  and column  $j$ , respectively, the leading matrices may have zeros in positions corresponding to small elements in  $A$  and  $B$ .

If  $s_A$  and  $s_B$  are large enough to guarantee that  $\Delta A = 0$  and  $\Delta B = 0$  in (7), then algorithm (2) will produce an extremely accurate approximation  $\tilde{C}$ , where the only rounding errors are due to the  $s_A s_B$  floating-point sums. Mukunoki et al. [5] have specialized this algorithm and have implemented it to obtain binary64 accuracy by using binary16 arithmetic on the NVIDIA tensor cores.

The latest NVIDIA GPUs can perform matrix multiplication even more efficiently using integer arithmetic. The tensor cores of NVIDIA H100 cards, for example, can compute the product of matrices stored in INT8 format (an 8-bit signed format) using 32-bit signed integer arithmetic. Exploiting the fixed-point nature of the Ozaki scheme, Ootomo, Ozaki, and Yokota [6] have therefore developed a method that computes the product of two binary64 matrices using only INT8 matrix multiplications. This initial idea was further refined by Uchino, Ozaki, and Imamura [9], who developed a more accurate and efficient variant of this scheme and gave a first error analysis. For  $s_A = s_B = s$ , they show that

$$|\tilde{C} - C| \leq 4(s+1)k2^{-qs}\alpha\beta^T + (s-1)u_{\text{high}}|A||B|,$$

where  $u_{\text{high}}$  is the unit roundoff of the floating-point arithmetic used to accumulate the partial matrix products in (2). This result suggests that the algorithm can be inaccurate if  $s$  is too small, or if the entries of the matrix are large in absolute value, as this will cause the entries of the vectors  $\alpha$  and  $\beta$  to be large.

We propose an alternative error analysis that can be used to inform the choice of the parameters  $s_A$  and  $s_B$ , which we argue need not be equal in the Ozaki scheme. First, we note that the terms in (7) satisfy

$$|\delta a_{ij}| < \alpha_i u_A, \quad u_A := 2^{-s_A q}, \quad |\delta b_{ij}| < \beta_j u_B, \quad u_B := 2^{-s_B q}. \quad (8)$$

In error analysis, it is often more informative to bound the relative error. Such bounds arise naturally when using floating-point arithmetic, because floating-point numbers have constant precision. In fixed-point arithmetic, precision is tapered, so bounds like those in (8) are more familiar, but it is still possible to bound  $|\delta a_{ij}|$  and  $|\delta b_{ij}|$  in terms of  $|a_{ij}|$  and  $|b_{ij}|$ , respectively, since

$$\begin{aligned} |\delta a_{ij}| &\leq \kappa_A u_A |a_{ij}|, & \kappa_A := 2 \max_{1 \leq i \leq m} \frac{\max\{|a_{ij}| : 1 \leq j \leq p\}}{\min_j \{|a_{ij}| : 1 \leq j \leq p \text{ and } a_{ij} \neq 0\}}, \\ |\delta b_{ij}| &\leq \kappa_B u_B |b_{ij}|, & \kappa_B := 2 \max_{1 \leq j \leq n} \frac{\max\{|b_{ij}| : 1 \leq i \leq p\}}{\min_i \{|b_{ij}| : 1 \leq i \leq p \text{ and } b_{ij} \neq 0\}}. \end{aligned}$$

Our analysis yields the alternative error bound

$$|\tilde{C} - C| \leq \kappa_A u_A + \kappa_B u_B + \kappa_A \kappa_B u_A u_B + \gamma_{s_A s_B - 1} (1 + \kappa_A u_A + \kappa_B u_B + \kappa_A \kappa_B u_A u_B)) |A| |B|.$$

In other words, the overall error can be substantial if either  $\kappa_A$  or  $\kappa_B$  are large. One can counteract the prominence of these two terms by increasing  $s_A$  and  $s_B$ , but doing so will negatively impact the performance of the algorithm, which needs to perform  $\mathcal{O}(s_A s_B)$  integer matrix multiplications.

The integer-based Ozaki scheme can be much faster than traditional high-precision alternatives, but our analysis suggests that it can also be significantly less accurate, depending on the dynamic range of the entries of  $A$  and  $B$ . The value of the parameters  $s_A$  and  $s_B$  required to meet a specific accuracy target can be determined by examining  $\kappa_A$  and  $\kappa_B$ , which are inexpensive to compute. For a given choice of  $s_A$  and  $s_B$ , we can estimate the runtime of the scheme, and we can opt for a traditional high-precision routine when the latter is expected to be faster.

## References

- [1] M. Fasi, N. J. Higham, F. Lopez, T. Mary, and M. Mikaitis. [Matrix multiplication in multiword arithmetic: Error analysis and application to GPU tensor cores](#). *SIAM J. Sci. Comput.*, 45(1):C1–C19, 2023.
- [2] M. Fasi, N. J. Higham, M. Mikaitis, and S. Pranesh. [Numerical behavior of NVIDIA tensor cores](#). *PeerJ Comput. Sci.*, 7:e330(1–19), 2021.
- [3] M. Fasi and M. Mikaitis. [CPFloat: A C library for simulating low-precision arithmetic](#). *ACM Trans. Math. Software*, 49(2):1–32, 2023.
- [4] S. Markidis, S. W. D. Chien, E. Laure, I. B. Peng, and J. S. Vetter. [NVIDIA tensor core programmability, performance & precision](#). In *Proceedings of the 2018 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, May 2018.
- [5] D. Mukunoki, K. Ozaki, T. Ogita, and T. Imamura. [DGEMM using tensor cores, and its accurate and reproducible versions](#). In *High Performance Computing*, P. Sadayappan, B. L. Chamberlain, G. Juckeland, and H. Ltaief, editors, Springer-Verlag, 2020, page 230–248.
- [6] H. Ootomo, K. Ozaki, and R. Yokota. [DGEMM on integer matrix multiplication unit](#). *Int. J. High Performance Computing Applications*, 38(4):297–313, 2024.
- [7] K. Ozaki, T. Ogita, S. Oishi, and S. M. Rump. [Error-free transformations of matrix multiplication by using fast routines of matrix multiplication and its applications](#). *Numer. Algorithms*, 59(1):95–118, 2012.
- [8] S. M. Rump, T. Ogita, and S. Oishi. [Accurate floating-point summation part I: Faithful rounding](#). *SIAM J. Sci. Comput.*, 31(1):189–224, 2008.
- [9] Y. Uchino, K. Ozaki, and T. Imamura. [Performance enhancement of the Ozaki scheme on integer matrix multiplication unit](#). Technical report, September 2024. arXiv:2409.13313 [cs.DC].

# Multigrid Methods for Solving Indefinite Problems in Port-Hamiltonian Systems

*Paola Ferrari, Matthias Bolten*

## Abstract

In this study, we develop and analyze multigrid methods for efficiently solving large-scale indefinite structured linear systems that arise in port-Hamiltonian systems. Port-Hamiltonian systems are open dynamical systems characterized by a Hamiltonian function representing the stored energy, and they interact with their environment through ports, which facilitate the exchange of energy. Mathematically, these systems are described by differential equations coupled with algebraic constraints, forming a framework that inherently preserves the energy-conserving properties of physical systems. Resistive effects are included by terminating some of the ports on energy-dissipating elements, such as resistors, which introduce dissipation into the system. Finding the numerical solution of such problems results in resolving linear systems with specific structures, often indefinite and involving skew-symmetric and symmetric blocks [6, 8].

The combination of energy-conserving and dissipative properties leads to saddle-point structures. When discretized, they result in large, indefinite systems of equations that pose significant computational challenges. The matrices involved are often block-based, with components that are skew-symmetric or symmetric, and with proper discretizations, they are of (multilevel block) Toeplitz type. The skew-symmetry can lead to Hermitian but indefinite matrices when multiplied by the imaginary unit. The coupling introduces additional challenges due to the skew-symmetric nature of the interactions, necessitating specialized numerical techniques for efficient and stable solutions.

Multigrid methods have long been recognized as optimal solvers for a wide range of elliptic partial differential equations (PDEs) due to their ability to provide convergence rates independent of the problem size. However, adapting these methods to indefinite and structured problems, such as those encountered in port-Hamiltonian systems, requires careful consideration of the matrix properties and the selection of appropriate smoothers and grid transfer operators. In this study, we address such challenges by focusing on two key aspects: solving general saddle-point systems that arise naturally in port-Hamiltonian models and solving skew-symmetric Toeplitz systems. Our tailored multigrid methods are designed to handle the indefiniteness and structured properties of the resulting linear systems, providing efficient and stable solutions to the considered complex problems.

To tackle these computational challenges, we propose applying an approach inspired by Notay's method [7] for solving saddle-point systems. This method involves transforming the original system through pre- and post-multiplication with sparse block triangular matrices, effectively preconditioning the system to be more suitable for resolution by multigrid techniques. After this transformation, the diagonal blocks become symmetric and positive definite, resembling discrete Laplace operators that are well-suited for multigrid solvers. In our previous works [2, 3], we successfully applied this approach to multilevel block Toeplitz matrices arising from finite element approximations of systems of PDEs such as the Stokes problem, demonstrating that it leads to efficient multigrid methods with convergence rates independent of the matrix size. We extend this approach to port-Hamiltonian systems.

To establish a rigorous convergence analysis of our proposed multigrid methods, it is crucial to examine the detailed matrix structures of the discretized systems. These matrices are low-rank

corrections of multilevel block Toeplitz matrices, whose properties can be characterized by their (matrix-valued) generating functions, also called symbols. By deriving the generating functions, we gain insight into the spectral properties of the transformed matrices, which are essential for analyzing the effectiveness of multigrid methods. Specifically, for saddle-point problems with hidden block Toeplitz structure, the analysis of the generating functions enables us to determine optimal preconditioning strategies and to select appropriate smoothers and grid transfer operators within the multigrid hierarchy.

For structured problems involving multilevel block Toeplitz matrices, such as those arising in port-Hamiltonian systems, it is essential to preserve the matrix structure across different grid levels. This focus guides the design of our grid transfer operators. By retaining the Toeplitz-like structure, we can apply symbol-based convergence analysis uniformly across all grid levels. This approach enables us to rigorously analyze and predict the convergence behavior of the multigrid method, providing both theoretical guarantees and practical performance benefits.

Additionally, we present a preliminary analysis of multigrid methods for port-Hamiltonian-derived saddle-point problems with hierarchical and recursive configurations. These configurations commonly arise in complex port-Hamiltonian models where multiple nested levels of interactions or coupling mechanisms are present. By extending our multigrid framework to accommodate hierarchical and recursive structures, we highlight the potential for scalability and effectiveness in a broader class of indefinite and structured linear systems.

To validate our theoretical findings, we present numerical experiments focusing on field-circuit coupling problems [4] and also on non-convex shape optimization problems [1]. In the field-circuit coupling experiments, we tackle large, indefinite linear systems arising from the interaction of Maxwell's equations with circuit equations, simplified under certain modeling assumptions to quasi-stationary models or dimensionally reduced forms like the telegraph equation. Concerning shape optimization, we address non-convex problems involving mechanical components, such as optimizing the shape of ceramic parts under reliability, volume, and construction space constraints. These numerical experiments confirm that the convergence rates of our multigrid methods are indeed independent of the problem size, validating the effectiveness of our approach in practical, complex applications.

Furthermore, we focus on the resolution of skew-symmetric structured linear systems. Multiplying a skew-symmetric Toeplitz matrix by the imaginary unit transforms it into a Hermitian matrix; however, the resulting matrix is not positive definite, complicating the application of standard multigrid methods.

In particular, we introduce a novel approach by interpreting a scalar skew-symmetric Toeplitz matrix as a block Toeplitz matrix with  $2 \times 2$  blocks. This block formulation allows us to associate the transformed Hermitian matrix with a matrix-valued generating function, which we demonstrate is diagonalizable and possesses one positive and one negative eigenvalue function. According to the relationship between the eigenvalues of Hermitian Toeplitz matrices and their generating functions—as established by Szegő’s theorem—the zeros of these eigenvalue functions lead to near-zero eigenvalues in the matrix. The near-zero eigenvalues can significantly hinder the convergence of multigrid methods due to slow error reduction in the associated spectral components [5].

To address this challenge, we develop grid transfer operators that consider the two eigenvalue functions separately, effectively tailoring the multigrid hierarchy to the distinct spectral properties of the positive and negative eigenvalues. By analyzing the eigenvalue functions and their zeros, we design interpolation and restriction operators that enhance the coarse-grid correction process.

Additionally, we employ block-Jacobi methods as smoothers and establish a preliminary theoretical framework to estimate optimal smoothing parameters, thereby improving the overall efficiency of the multigrid method.

Our approach ensures robust convergence of multigrid methods for skew-symmetric Toeplitz systems, even in the presence of near-zero eigenvalues. Numerical experiments confirm the effectiveness of our method, showing optimal convergence rates also in the multilevel setting.

Overall, our work advances the development of efficient multigrid methods for large-scale, indefinite, and structured linear systems arising in port-Hamiltonian systems. By addressing the challenges associated with saddle-point structures and skew-symmetric Toeplitz matrices, we provide a robust computational framework with convergence rates independent of the problem size. Our theoretical analyses, supported by numerical experiments, demonstrate the potential of these methods to significantly improve computational efficiency in modeling complex physical systems.

## References

- [1] M. Bolten, O. T. Doganay, H. Gottschalk, and K. Klamroth. Non-convex shape optimization by dissipative hamiltonian flows. *Engineering Optimization*, pages 1–20, 2024.
- [2] M. Bolten, M. Donatelli, P. Ferrari, and I. Furci. Symbol based convergence analysis in block multigrid methods with applications for Stokes problems. *Appl. Numer. Math.*, 193:109–130, 2023.
- [3] M. Bolten, M. Donatelli, P. Ferrari, and I. Furci. Symbol based convergence analysis in multigrid methods for saddle point problems. *Linear Algebra Appl.*, 671:67–108, 2023.
- [4] M. Clemens, F. Kasolis, and M. Günther. Port-hamiltonian system framework for conservatively coupled discrete electromagnetics and multi-physics problem formulations. In *11th Conference on the Computation of Electromagnetic Fields (CEM 2023), Cannes, France*, 2023.
- [5] G. Fiorentino and S. Serra. Multigrid methods for Toeplitz matrices. *Calcolo*, 28:283–305, 1991.
- [6] C. Güdücü, J. Liesen, V. Mehrmann, and D. B. Szyld. On non-hermitian positive (semi)definite linear algebraic systems arising from dissipative hamiltonian daes. *SIAM Journal on Scientific Computing*, 44(4):A2871–A2894, 2022.
- [7] Y. Notay. A new algebraic multigrid approach for Stokes problems. *Numer. Math.*, 132(1):51–84, 2016.
- [8] A. van der Schaft. Port-hamiltonian systems: an introductory survey. In *Proceedings of the International Congress of Mathematicians Vol. III*, number suppl 2, pages 1339–1365. European Mathematical Society Publishing House (EMS Ph), 2006.

# Interpolated Compressed Inverse Preconditioning: Fast and Accurate Simulation of Close-to-Touching Discs in Stokes Flow

*Daniel Fortunato, Mariana Martínez Aguilar, Dhairyा Malhotra*

## Abstract

We consider the flow of dense suspensions of rigid bodies in a Stokesian fluid. Such flows are difficult to compute numerically due to the presence of close-to-touching interactions, which may require a large number of unknowns to resolve sharply peaked surface forces, a large number of GMRES iterations to solve the discretized PDE, and an extremely small time step. A common way of dealing with these difficulties is to introduce a repulsion force between particles to prevent them from getting too close. However, this additional repulsion force is non-physical and may fundamentally alter the results of a simulation.

For suspensions of identical discs in 2D, we present a fast and accurate boundary integral method that mitigates these challenges without introducing artificial forces. Through precomputation, compression and interpolation of the close-to-touching part of the interaction operator, our method—termed *interpolated compressed inverse preconditioning*—efficiently handles close-to-touching interactions down to distances of  $10^{-10}$  with only a coarse discretization of the boundary. Additionally, we present a preconditioner that significantly reduces the number of GMRES iterations required to solve the Stokes mobility problem at each time step by effectively reusing the Krylov subspace from previous time steps. Coupled with high-order, adaptive time-stepping using spectral deferred correction, we are able to take larger time steps, mitigating the temporal stiffness resulting from close-to-touching interactions.

For a graphical description of this work, see: <https://danfortunato.com/talks/ICIP-poster.pdf>.

## 1 Stokes mobility problem

We consider  $N_\Omega$  rigid discs  $\Omega = \{\Omega_1, \dots, \Omega_{N_\Omega}\}$  embedded in a Stokesian fluid. The fluid velocity in the exterior of  $\Omega$  is governed by the Stokes equations,

$$-\Delta \mathbf{u} + \nabla p = 0 \quad \text{in } \mathbb{R}^2 \setminus \Omega, \quad (1)$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \mathbb{R}^2 \setminus \Omega, \quad (2)$$

where  $\mathbf{u}$  is the fluid velocity and  $p$  is the fluid pressure. Equations (1) and (2) denote the momentum balance and incompressibility constraints, respectively. In addition, we also assume that the fluid velocity at infinity decays to zero,

$$\mathbf{u}(\mathbf{x}) \rightarrow 0 \quad \text{as } |\mathbf{x}| \rightarrow \infty.$$

Each disc has a net force  $\mathbf{F}_k$  and a net torque  $T_k$  acting about a point  $\mathbf{x}_k^c$ . The discs undergo rigid body motion with the velocity  $\mathbf{V}$  given by,

$$\mathbf{V}(x) = \mathbf{v}_k + \omega_k (\mathbf{x} - \mathbf{x}_k^c)^\perp \quad \text{for all } \mathbf{x} \in \Omega_k,$$

where  $\mathbf{v}_k$  is the translational velocity and  $\omega_k$  is the angular velocity of  $\Omega_k$  about the point  $\mathbf{x}_k^c$ . A slip velocity boundary condition  $\mathbf{u}_s$  between the rigid bodies and the fluid is prescribed. Therefore, the fluid velocity on the boundary  $\partial\Omega$  is given by,

$$\mathbf{u} = \mathbf{V} + \mathbf{u}_s \quad \text{on } \partial\Omega.$$

In the mobility problem, we are given  $\mathbf{u}_s$ ,  $\mathbf{F}_k$ , and  $T_k$  about  $\mathbf{x}_k^c$  for each  $\Omega_k$ . The rigid body motion  $\mathbf{V}$  (i.e.,  $\mathbf{v}_k$  and  $\omega_k$  for each  $\Omega_k$ ) is not known and must be determined.

Using the Stokes single- and double-layer potentials to represent the fluid velocity  $\mathbf{u}$  in terms of an unknown surface density  $\boldsymbol{\sigma}$ , a boundary integral equation (BIE) for the Stokes mobility problem can be formulated as given in [1]:

$$\mathcal{K}\boldsymbol{\sigma} = g \quad \text{on } \partial\Omega, \quad (3)$$

where  $\mathcal{K}$  is a second-kind boundary integral operator and  $g$  encodes the given slip velocity, net force, and net torque.

## 2 Close-to-touching interactions

Consider the model problem of two discs separated by a distance  $d$ , with each disc discretized into a set of high-order Gauss–Legendre panels. The two disc problem serves as an effective pairwise preconditioner in a simulation with many close-touching discs. When the distance  $d$  between two discs gets small, the solution  $\sigma$  to the BIE in (3) becomes highly peaked. This requires an extremely fine discretization of the boundary in the close-to-touching region. We label the close-to-touching region as  $\Gamma_2$  and the remaining boundary as  $\Gamma_1 = \partial\Omega \setminus \Gamma_2$ . Then, (3) can be discretized as a block linear system,

$$\begin{pmatrix} K_{11} & K_{12} \\ K_{21} & K_{22} \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \sigma_2 \end{pmatrix} = \begin{pmatrix} g_1 \\ g_2 \end{pmatrix}, \quad (4)$$

where  $g_1$  and  $\sigma_1$  are the boundary conditions and unknowns on  $\Gamma_1$ ,  $g_2$  and  $\sigma_2$  are the boundary conditions and unknowns on  $\Gamma_2$ , and  $K_{ij}$  represents a sub-block of the discretized operator  $\mathcal{K}$  that computes interactions from sources on  $\Gamma_j$  to targets on  $\Gamma_i$ . Right preconditioning (4) with the block diagonal preconditioner  $\begin{pmatrix} I & 0 \\ 0 & K_{22}^{-1} \end{pmatrix}$  yields the system

$$\begin{pmatrix} K_{11} & K_{12}K_{22}^{-1} \\ K_{21} & I \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \bar{\sigma}_2 \end{pmatrix} = \begin{pmatrix} g_1 \\ g_2 \end{pmatrix}, \quad (5)$$

where  $\bar{\sigma}_2 = K_{22}\sigma_2$  is a new unknown on  $\Gamma_2$ . While (5) may require fewer GMRES iterations to solve than (4), it still requires an excessively fine discretization in the close-to-touching region  $\Gamma_2$ . Additionally, computing  $K_{22}^{-1}$  on the fly can be expensive, especially for problems with moving boundaries. However, one may show that  $\bar{\sigma}_2$  can be discretized on a coarse mesh and that the off-diagonal block  $K_{12}K_{22}^{-1}$  is low rank.

### 2.1 Compressing close-to-touching interactions

Since  $\Gamma_1$  and  $\Gamma_2$  are disjoint, the discretized boundary integral operators  $K_{12}$  and  $K_{21}$  are low rank, with the numerical rank independent of the distance  $d$ . Hence, the column space of  $K_{21}$  is comprised of smooth functions that can be discretized using piecewise polynomials on a coarse mesh. From (5), we have  $\bar{\sigma}_2 = g_2 - K_{21}\sigma_1$ ; therefore,  $\bar{\sigma}_2$  is smooth whenever  $g_2$  is smooth and it can be discretized on a coarse mesh. Since  $K_{12}$  is low rank, so is  $K_{12}K_{22}^{-1}$  (with numerical rank independent of  $d$ ). There are several ways of constructing a compressed representation for  $K_{12}K_{22}^{-1}$ . To retain the boundary integral structure and allow for acceleration by the fast multipole method (FMM), we use a representation of the form  $K_{12}R$  where  $R$  is a low-rank operator such that

$K_{12}R \approx K_{12}K_{22}^{-1}$ , up to a given numerical tolerance. We now describe the numerical construction of  $R$ .

Consider two different panelizations of  $\Gamma_2$ : a fine mesh where the panels on each disc are refined dyadically towards the closest point between the discs, and a coarse mesh with a small number of uniformly sized panels on each disc. We denote quantities on the coarse mesh with a superscript “ $c$ ”; all other quantities are assumed to live on the fine mesh. Define the prolongation operator  $P$  that interpolates data from the coarse mesh to the fine mesh, and diagonal matrices  $W_f$  and  $W_c$  containing the weights for smooth integration on the fine and coarse meshes, respectively. Then,  $W_c^{-1}P^T W_f$  computes an  $L^2$  projection from the fine mesh to the coarse mesh.

Assuming that the boundary data  $g_2$  is smooth and therefore representable on the coarse mesh, we have

$$g_2 = P g_2^c, \quad (6)$$

$$\bar{\sigma}_2 = P \bar{\sigma}_2^c. \quad (7)$$

Since  $K_{12}$  and  $K_{21}$  are low rank, they can be approximated accurately by their coarse discretizations,

$$K_{12} = K_{12}^c W_c^{-1} P^T W_f, \quad (8)$$

$$K_{21} = P K_{21}^c, \quad (9)$$

Substituting (6)–(9) in (5), we obtain

$$\begin{pmatrix} K_{11} & K_{12}^c R \\ K_{21}^c & I \end{pmatrix} \begin{pmatrix} \sigma_1 \\ \bar{\sigma}_2^c \end{pmatrix} = \begin{pmatrix} g_1 \\ g_2^c \end{pmatrix} \quad (10)$$

where  $R = W_c^{-1}P^T W_f K_{22}^{-1}P$ . This definition of  $R$  is used in the RCIP (recursively compressed inverse preconditioning) method [2]. For an order- $p$  fine mesh with  $\mathcal{O}(\log d)$  levels of refinement, direct construction of  $R$  takes  $\mathcal{O}(p^3(\log d)^3)$  operations; the RCIP method provides a faster algorithm to construct  $R$  without directly computing  $K_{22}^{-1}$ , taking  $\mathcal{O}(p^3 \log d)$  operations. Our main result—termed ICIP (interpolated compressed inverse preconditioning)—instead constructs  $R$  through pre-computation and interpolation, requiring only  $\mathcal{O}(p^2)$  work.

## 2.2 Interpolated compressed inverse preconditioning (ICIP)

Constructing  $R = R(d)$  each time for a different value of  $d$  can be expensive since it requires computing  $K_{22}^{-1}$ . Instead, we construct a polynomial interpolant for  $R(d)$  over a range  $d \in [d_{\min}, d_{\max}]$  (where  $0 < d_{\min} < d_{\max}$ ). Then for any value of  $d$  in the interval, we construct  $R(d)$  through entrywise interpolation. We use Chebyshev polynomials in  $\log d$  as our interpolation basis, i.e.,

$$[R(d)]_{ij} \approx \sum_{k=0}^q [R_k]_{ij} T_k(\log d)$$

where  $[R_k]_{ij}$  is the  $k$ th Chebyshev coefficient for the  $ij$ th entry of  $R(d)$ . For accurate interpolation of  $R(d)$  over a large dynamic range of  $10^{-10} < d < 10^{-1}$ , only a moderate interpolation order of  $q = 32$  is required. After an offline precomputation to generate  $\{R_k\}_{k=0}^q$ , constructing  $R(d)$  at each time step costs  $\mathcal{O}(p^2 q)$  operations.

### 3 Accelerating timestepping with subspace recycling

While the two-disc preconditioner is effective at lowering the number of GMRES iterations induced by close-to-touching interactions, a significant number of GMRES iterations may still be required at each time step for problems with many discs. To ameliorate this effect, we propose a preconditioner which effectively reuses the Krylov subspace from previous time steps.

After the  $k$ th iteration of GMRES, the Krylov matrix is given by  $X = [b \ Ab \ \dots \ A^{k-1}b]$ . Let  $QR = AX$  be the QR decomposition of  $AX$ . Then the matrix  $P$  given by

$$P = I - QQ^T + XR^{-1}Q^T$$

has the following properties:

$$PAx = x \text{ for all } x \in \text{span}(X),$$

$$Py = y \text{ for all } y \perp \text{span}(X).$$

Hence,  $P$  effectively reuses the given Krylov subspace  $X$  when used as a preconditioner in a Krylov method. In a high-order time-stepping scheme based on spectral deferred corrections, this preconditioner can drastically reduce the number GMRES iterations by recycling the Krylov subspace between time steps.

## References

- [1] C. Pozrikidis, *Boundary Integral and Singularity Methods for Linearized Viscous Flow*, Cambridge University Press, 1992.
- [2] J. Helsing, *Solving integral equations on piecewise smooth boundaries using the RCIP method: a tutorial*, 2022.

# Analysis on Aggregation and Block Smoothers in Multigrid Methods for Block Structured Linear Systems

*Matthias Bolten, Marco Donatelli, Paola Ferrari, Isabella Furci*

## Abstract

In this talk we present a detailed analysis of block smoothers and propose new aggregation-based strategies in multigrid methods for structured linear systems [3].

Multigrid methods are well-known for their efficiency in solving large linear systems, especially when the coefficient matrices exhibit a large (multilevel block) structure [2, 7, 8]. These methods are widely used in various scientific and engineering applications, including the discretization of partial differential equations (PDEs), image processing, and approximation problems.

The key to developing effective multigrid methods lies in the careful selection of the grid transfer operator  $P$  and the iterative methods that serve as pre/post smoothers within the multigrid iteration. In previous works [2, 8], a comprehensive convergence analysis for two-grid and V-cycle methods applied to scalar Toeplitz and circulant systems was developed. Specifically, the convergence requirements for  $P$  and smoothers were formulated elegantly using conditions on the (scalar-valued) function  $f$  associated with the structured coefficient matrix. Extending this *symbol-based* analysis to block systems introduces challenges such as the non-commutativity of matrix-valued functions and the need for suitable grid transfer operators to manage coarse-grid corrections.

Scalar smoothers can be defined by carefully selecting the relaxation parameter based on the matrix-valued function  $\mathbf{f}$ , mimicking the approach used for scalar-valued functions. However, we show that, as the block dimension  $d$  increases, block smoothers become more appropriate because they align more naturally with the system's block structure.

One of the main goals of this talk is to provide an in-depth analysis of block smoothers, which are more effective in handling block-structured systems. Specifically, we introduce a relaxed block Jacobi method and derive general conditions for the smoothing parameter  $\omega$  that ensure convergence. This method proves more efficient than scalar smoothers, both in terms of solving time and set-up time. From our comparison of scalar and block Jacobi smoothers, we demonstrate that the block Jacobi method consistently outperforms its scalar counterpart in terms of convergence rate and computational efficiency, particularly for systems with large block dimensions. Moreover, we show that the general conditions on smoothing parameters can be calculated with negligible computational cost in some practical applications.

Regarding the choice of grid transfer operators, a rigorous convergence analysis for the two-grid method (TGM), further extended to the V-cycle, was derived in [6]. The latter demonstrates that certain classical grid transfer operators, such as the geometric projection and standard bisection operators, meet the necessary conditions for convergence in many practical cases. The grid transfer operator in this context is written as

$$P = \mathcal{A}_n(\mathbf{p})(K \otimes I_d),$$

where  $K$  is an  $n \times k$  matrix that reduces the dimension of the problem by selecting specific rows, and  $\mathcal{A}_n(\mathbf{p})$  is a structured matrix analogous to the original problem. A key aspect of such grid transfer operators is that they preserve the block structure at coarser levels, and the proof of the approximation convergence property is based on validating further commutativity conditions on the matrix-valued symbol  $\mathbf{p}$ .

However, many known grid transfer operators were not covered by the previous theory. To address this, new conditions were obtained in [4], expressing the projector's approximation property in terms of the eigenvector associated with the ill-conditioned subspace, thereby broadening the class of valid operators. Specifically, convergence results in the block-structured setting were derived by exploiting block-symbol analysis. If  $\mathbf{f}(\theta)$  is a trigonometric polynomial with exactly one zero eigenvalue at  $\theta_0$  and is positive definite in  $[0, 2\pi) \setminus \{\theta_0\}$ , it can be diagonalized as

$$\mathbf{f}(\theta) = Q(\theta)D(\theta)Q(\theta)^H,$$

where  $Q(\theta)$  is the matrix of eigenvectors and  $D(\theta)$  is the diagonal matrix of eigenvalues. We denote by  $q_{\bar{j}}(\theta)$  the normalized eigenvector associated with the zero eigenvalue  $\lambda_{\bar{j}}(\mathbf{f}(\theta_0)) = 0$ . Under certain assumptions, sufficient conditions for the linear convergence of the TGM involve choosing  $\mathbf{p}$  such that the function  $\mathbf{p}(\theta)^H \mathbf{p}(\theta) + \mathbf{p}(\theta + \pi)^H \mathbf{p}(\theta + \pi)$  is positive definite for all  $\theta \in [0, 2\pi)$  and specific limit conditions as  $\theta$  approaches  $\theta_0$  are satisfied. These requirements can be simplified in specific applications, and the V-cycle convergence conditions are based on these results. An important aspect of the strategy outlined above is that the grid transfer operator maintains the block structure at coarser levels. While this is theoretically convenient for proving V-cycle convergence, it introduces computational challenges, as the block structure remains, and the matrix-valued function can be difficult to analyze.

The second aim of this talk is to present a new symbol-based multigrid method with an aggregation-based approach, which reduces the system to a scalar form at coarser levels. In particular, from the decomposition of the matrix-valued trigonometric polynomial  $\mathbf{f}$ , we propose a grid transfer operator of the form

$$P = I_n \otimes q_{\bar{j}}(\theta_0).$$

This approach offers significant computational advantages, especially for large-scale systems [1]. At the coarse level, the coefficient matrix simplifies, resulting in a scalar-valued function

$$\tilde{f}(\theta) = q_{\bar{j}}^H(\theta_0) \mathbf{f}(\theta) q_{\bar{j}}(\theta_0),$$

which maintains some properties of the original problem. Indeed,  $\tilde{f}$  vanishes at  $\theta_0$  (and only at  $\theta_0$ ), and with a zero of order smaller than or equal to that of the original  $\lambda_{\bar{j}}(\mathbf{f})$ , ensuring that the conditioning of the problem does not worsen.

We derive the convergence and optimality of the TGM by combining the approximation property verified by  $P$  and the findings on the block smoothers. We also extend the TGM analysis to the V-cycle, where the properties of the scalar-valued symbol at the coarse level are crucial in determining the convergence rate. In particular, we show that V-cycle convergence can be achieved by combining the TGM results with an analysis of the scalar coarse-level symbol. This allows us to formulate clear conditions for convergence and optimality, even in complex block settings.

From a computational perspective, aggregation-based methods provide substantial savings while maintaining convergence, particularly when combined with over-relaxation strategies. As proposed by Braess [5], over-relaxation of the coarse-grid correction significantly enhances multigrid performance, and we show the effectiveness of this strategy when combined with our symbol-based grid transfer operators. This corresponds to performing an over-relaxation with a parameter  $\alpha > 1$  when computing the interpolation of the error,  $\alpha P y$ . Consequently, we present a strategy to select the optimal pair  $(\omega, \alpha)$  to minimize the spectral radius of the TGM iteration matrix, thereby improving results.

To validate our theoretical findings, we conduct numerical experiments using the proposed approach both as a standalone method and as a preconditioner for Krylov iterative methods. The tested large-scale block circulant, block Toeplitz-like, and Generalized Locally Toeplitz (GLT) linear systems stem from discretizations using  $\mathbb{Q}_d$  Lagrangian FEM approximation of second-order differential problems and B-spline discretization with non-maximal regularity. In addition to confirming our theoretical results, these examples demonstrate how the conditions outlined for block smoother convergence simplify in practical scenarios.

## References

- [1] C. An, Y. Su. An aggregation-based Two-Grid method for multilevel block Toeplitz linear systems. *J. Sci. Comput.*, 98(3): Paper No. 54, 2024.
- [2] A. Aricò, M. Donatelli. A V-cycle multigrid for multilevel matrix algebras: proof of optimality. *Numer. Math.*, 105 (4): 511–547, 2007.
- [3] M. Bolten, M. Donatelli, P. Ferrari, I. Furci. Analysis on aggregation and block smoothers in multigrid methods for block Toeplitz linear systems. (*under revision*), arXiv:2403.02139v1
- [4] M. Bolten, M. Donatelli, P. Ferrari, I. Furci. A symbol based analysis for multigrid methods for Block-Circulant and Block-Toeplitz Systems. *SIAM J. Matrix Anal. Appl.*, 43(1): 405–438, 2022.
- [5] D. Braess. Towards algebraic multigrid for elliptic problems of second order. *Computing*, 55(4): 379–393, 1995.
- [6] M. Donatelli, P. Ferrari, I. Furci, S. Serra–Capizzano, D. Sesana. Multigrid methods for Block-Toeplitz linear systems: Convergence Analysis and Applications. *Numer. Linear Algebra Appl.*, 28(4): e2356, 2021.
- [7] H. Elman, A. Ramage. Fourier Analysis of multigrid for a model two-dimensional convection-diffusion equation. *BIT Numer Math*, 46, 283–306, 2006
- [8] G. Fiorentino, S. Serra, A. Ramage. Multigrid methods for symmetric positive definite block Toeplitz matrices with nonnegative generating functions. *SIAM J. Sci. Comput.*, 17(5), 1068–1081, 1996.

# Proving Rapid Global Convergence for the Shifted QR Algorithm

*Jess Banks, Jorge Garza-Vargas, Nikhil Srivastava*

## Abstract

The design of efficient and reliable algorithms for computing the eigenvalues and eigenvectors of a matrix is of unquestionable importance in both science and engineering. However, despite significant advancements in various practical aspects, fundamental theoretical questions about the eigenvalue problem remain poorly understood. In this talk I will discuss work [BGVSA, BGVSB, BGVSC] that provides nearly optimal rigorous guarantees, on all inputs, for the shifted QR algorithm. Similar results were established by Wilkinson in [Wil68] and Dekker and Traub in [DT71] for Hermitian inputs; however, despite sustained interest and several attempts, the non-Hermitian case remained elusive for the last five decades.

**The QR iteration.** The QR algorithm, which originated in the works of Francis [Fra61, Fra62] and Kublanovskaya [Kub62] (see [GU09] for some history), has been listed as one of the top ten most influential algorithms of the 20th century [DS00] and is the preferred method for computing the full eigendecomposition of an arbitrary input matrix.

In its simpler form, the QR algorithm starts by putting the input matrix  $A \in \mathbb{C}^{n \times n}$  into Hessenberg form, that is, it computes a unitary matrix  $U$  such that  $H = U^*AU$  is an upper Hessenberg matrix.<sup>1</sup> Then, it computes a sequence of Hessenberg matrices  $H_0 = H, H_1, H_2 \dots$  via the iteration:

$$\begin{aligned} [Q_t, R_t] &= \text{qr}(H_t), \\ H_{t+1} &= Q_t^* H_t Q_t. \end{aligned} \tag{1}$$

Where  $Q_t R_t = H_t$  is the QR decomposition of  $H_t$ , and from  $H_{t+1} = Q_t^* H_t Q_t$  we see that

$$A = U_t H_t U_t^* \quad \text{for} \quad U_t = U Q_0 \cdots Q_t.$$

This iteration has the fascinating property (see [Wat82]) that for generic<sup>2</sup> inputs  $A$ , as  $t$  goes to infinity, the  $H_t$  converge to an upper triangular matrix, say,  $T$ . In such situation, we can set  $V = \lim_{t \rightarrow \infty} U_t$ , so that

$$A = V T V^*,$$

therefore obtaining the Schur decomposition of  $A$ .<sup>3</sup> The appeal of this method resides on the simplicity of the iteration described in (1). The drawback is that the convergence  $H_t \rightarrow T$  happens at a prohibitively slow rate for most inputs, ultimately turning it into an impractical algorithm.

**The shifted QR algorithm.** In practice the QR iteration is endowed with “shifts” that seek to accelerate convergence. Concretely, at each time  $t$ , a polynomial  $p_t(z)$  is computed as a function of  $H_t$  (see Wilkinson’s shift below for an example) and the iteration now is given by:

$$\begin{aligned} [Q_t, R_t] &= \text{qr}(p_t(H_t)), \\ H_{t+1} &= Q_t^* H_t Q_t. \end{aligned} \tag{2}$$

---

<sup>1</sup>This can be done by applying a sequence of  $n - 1$  suitably chosen Householder transformations. This procedure is numerically stable and can be executed in  $O(n^3)$  arithmetic operations, see [Wat08] for details.

<sup>2</sup>That is, all but a set of Lebesgue measure zero.

<sup>3</sup>Recall that one can read the eigenvalues of  $A$  from the diagonal entries of  $T$ , and if desired, easily compute the eigenvectors of  $A$  from the columns of  $V$ .

Intuitively, one should think of the roots of  $p_t(z)$  as “guesses” for the eigenvalues of  $H_t$  (which by unitary equivalence are the same as the eigenvalues of  $A$ ) and, the better the guesses the more progress towards convergence one will make while going from  $H_t$  to  $H_{t+1}$ . Moreover, the closer  $H_t$  is to an upper triangular matrix, the more its eigenvalues have been “revealed”, which allows one to make better guesses, all together yielding a virtuous cycle that is in part responsible for the undefeated performance of the shifted QR algorithm. This intuition can be made rigorous by understanding the connection between the shifted QR algorithm and shifted inverse iteration [Wat82, Wat08], where the aforementioned “virtuous cycle” can be established via a *local* analysis of convergence, e.g. see [Par74] or [Par98, §4.7].

The chosen algorithm for computing the  $p_t(z)$  as a function of the  $H_t$  is referred to as the *shifting strategy*, and the main purpose of any shifting strategy is to guarantee rapid *global* convergence, that is, rapid convergence to an upper triangular matrix regardless of the starting condition  $H_0$ . Although local convergence is intuitive (as explained above) and typically easy to establish, devising a shifting strategy that ensures rapid global convergence remained an important open problem throughout the years [Par74, Mol78, Dem97, Sma97, HDG<sup>+</sup>15].

**Exploiting the Hessenberg structure.** Working with Hessenberg matrices has several computational advantages that ultimately permit obtaining the full eigendecomposition of the input in nearly  $n^3$  operations, which is the initial cost of putting the input matrix into Hessenberg form.

*Easy deflation.* In practice, one can only hope to compute an approximate Schur form (resp. approximate eigendecomposition) for the input matrix. In turn, when seeking to solve the an approximate version of the eigenvalue problem, one can exploit the Hessenberg structure to accelerate the algorithm as follows. We will say that an upper Hessenberg matrix  $H$  is  $\delta$ -decoupled if one of its subdiagonals satisfies  $|H(i, i - 1)| \leq \delta \|H\|$ . So, in the iteration (2), once one of the matrices  $H_t$  is  $\delta$ -decoupled for some  $\delta$  small enough, one can zero out the small subdiagonal:

$$\left( \begin{array}{ccccc} * & * & * & * & * \\ * & * & * & * & * \\ 0 & \text{small} & * & * & * \\ 0 & 0 & * & * & * \\ 0 & 0 & 0 & * & * \end{array} \right) \longrightarrow \left( \begin{array}{cc|cc|cc} * & * & * & * & * & * \\ * & * & * & * & * & * \\ \hline 0 & 0 & * & * & * & * \\ 0 & 0 & * & * & * & * \\ 0 & 0 & 0 & * & * & * \end{array} \right).$$

This procedure, called *deflation*, incurs a small error in the computation but has the advantage that the resulting matrix is now block upper triangular, and therefore the spectrum of the big matrix is the union of the spectra of each of the smaller block diagonal parts, which happen to again have a Hessenberg structure. Moreover, the eigenvectors of the big matrix can be related in a similar way to the eigenvectors of the smaller diagonal blocks. With this, the eigenvalue problem has been reduced to two subproblems of smaller dimension, on which one can again call the QR algorithm.

*Implicit shifts.* Another advantage of the Hessenberg structure is that in the iteration (2) one can compute  $H_{t+1}$  from  $H_t$  without having to explicitly compute  $p_t(H_t)$ . Concretely, if  $p_t(z)$  is of degree  $k$  and  $H_t$  is of dimension  $n$ , one can compute  $H_{t+1}$  from  $H_t$  in  $O(kn^2)$  operations using a procedure commonly known as *chasing the bulge* (see [Tis96] or [Wat08]). Moreover, when the input matrix is Hermitian, the iterates  $H_t$  are tridiagonal, and in this case  $H_{t+1}$  can be computed from  $H_t$  in  $O(kn)$  operations.

*Meaningful corners.* If  $H$  is a normal upper Hessenberg matrix then the lower-right corners of  $H$  can be related to the orthogonal polynomials associated to a natural probability measure supported on the spectrum of  $H$ , and, there is a natural potential theory interpretation of the subdiagonals

of such corners. In the general non-normal case these interpretations are no longer valid, but still provide great intuition for the dynamics of the shifted QR algorithm. In part, this is the reason why many of the shifting strategies use small lower-right corners of the  $H_t$  to compute  $p_t(z)$ .

**Previous theoretical guarantees.** When the input  $A \in \mathbb{C}^{n \times n}$  is Hermitian, and therefore all the iterates  $H_t$  are too, Wilkinson introduced a shifting strategy that guarantees rapid global convergence. At time  $t$ , Wilkinson's shift computes the two eigenvalues of the lower-right  $2 \times 2$  matrix of  $H_t$  and takes the one (call it  $w_t$ ) that is closest to  $H_t(n, n)$  to then set  $p_t(z) = z - w_t$ . In [Wil68] Wilkinson proved that for any initial Hermitian  $H_0$ , if one runs the iteration (2) using his shifting strategy, it holds that  $\lim_{t \rightarrow \infty} H_t(n, n-1) = 0$ , which in particular implies that for any  $\delta > 0$ , the matrix  $H_t$  is  $\delta$ -decoupled once  $t$  is large enough. This was then revisited by Dekker and Traub [DT71] who obtained a rate of convergence for Wilkinson's shift by showing that

$$|H_{t+1}(n, n-1)^2 H_{t+1}(n-1, n-2)| \leq \frac{|H_t(n, n-1)^2 H_t(n-1, n-2)|}{\sqrt{2}}, \quad \text{for all } t \geq 0. \quad (3)$$

In particular, this implies that for any  $\delta > 0$ ,  $\delta$ -decoupling occurs in  $O(\log(1/\delta))$  iterations. Combining this with the deflation technique and the implicit shifts described above, one gets that any Hermitian matrix can be fully diagonalized to accuracy  $\delta$  in  $O(n^3 + \log(1/\delta)n^2)$  operations.

The case in which the input matrix  $A \in \mathbb{C}^{n \times n}$  is unitary was later solved by Eberlein and Huang [EH75] and Wang and Gragg [WG02]. When  $H_0$  is unitary the Wilkinson shift is no longer guaranteed to eventually produce decoupling. In fact, if the Wilkinson shift is used it can occur that  $H_0 = H_1 = H_2 = \dots$ , and similarly many other natural shifting strategies have certain unitary matrices as fixed points (see [Par66]). The insight of Eberlein and Huang [EH75] was that these commonly used shifting strategies could be combined with an *exceptional shift* that avoids stagnation. In essence, their idea was to choose a *main shift* (e.g. one could choose the Wilkinson shift), and then exploit the knowledge that the input matrix is unitary to identify fixed points for the main shift, to then escape them by invoking the exceptional shift whenever necessary. Later, Gragg and Wang [WG02] revisited this idea and showed that, on unitary inputs, a mixed strategy that combines the Wilkinson shift and an exceptional shift satisfies a more complicated version of (3). Their analysis implies that this mixed strategy achieves  $\delta$ -decoupling in  $O(\log(1/\delta))$  iterations, ultimately implying that any unitary input can be diagonalized to accuracy  $\delta$  in  $O(\log(1/\delta)n^3)$  operations.

Beyond Hermitian and unitary matrices not much was known and proving rapid global convergence was open even in the normal case. We refer the reader to [BGVSa, §1.2] for a comprehensive literature review.

**The main result.** In the series [BGVSa, BGVSb, BGVSc] we introduced a shifting strategy that provably achieves global rapid convergence (in the space of all matrices). Hereon, if all the  $p_t(z)$  in a shifting strategy are of degree  $k$  we will say that the shifting strategy is of degree  $k$ .

The condition number of the eigenvector matrix turned to be a fundamental quantity in our analysis. To be precise, if  $A \in \mathbb{C}^{n \times n}$  is diagonalizable, define

$$\kappa_V(A) = \inf_{V: A = VDV^{-1}} \|V\| \|V^{-1}\|,$$

where  $\|\cdot\|$  denotes the operator norm and the infimum runs over all diagonalizations of  $A$ . Note that when  $A$  is normal one has  $\kappa_V(A) = 1$  and when  $A$  is non-diagonalizable the convention is

that  $\kappa_V(A) = \infty$ , so  $\kappa_V(\cdot)$  can be viewed as a measure of non-normality. Fundamentally, [BGVSA] proves the following.<sup>4</sup>

**Theorem 1.** *For every positive integer  $k$ , there exists a shifting strategy of degree  $k$  that is ensured to achieve  $\delta$ -decoupling in  $\log(1/\delta)$  iterations provided that the starting matrix  $H_0$  satisfies*

$$\log(1 + \kappa_V(H_0)) \cdot \log(1 + \log(1 + \kappa_V(H_0))) \leq ck, \quad (4)$$

where  $c > 0$  is some absolute constant.

In some sense, our analysis articulates that the complexity of shifted QR is tied to  $\kappa_V$  of the input. In particular, the above theorem implies that rapid global convergence on normal matrices is possible using a shifting strategy of degree  $O(1)$ , just as in the case of Hermitian and unitary matrices. In contrast, when the input is non-diagonalizable the strategy needed is “infinitely complex” and the theorem becomes vacuous. That said, the latter situation can be addressed using an idea from smoothed analysis [ST04] which in the context of the eigenvalue problem can be traced back to Davies [Dav08]. In short, to obtain guarantees for arbitrary inputs, instead of running the algorithm on the original input matrix  $A \in \mathbb{C}^{n \times n}$  we run it on  $A + \gamma G_n$ , where  $\gamma G_n$  is a tiny random perturbation of  $A$ . One can then invoke results from random matrix theory (e.g. from [ABB<sup>+</sup>18, BKMS21, BGVKS24, JSS21]), which for example imply that if  $G_n$  is a normalized  $n \times n$  Ginibre matrix<sup>5</sup>,  $\|A\| \leq 1$ , and  $\gamma > 0$ , with high probability

$$\kappa_V(A + \gamma G_n) \leq \frac{n^4}{\gamma}. \quad (5)$$

Certainly, this *preprocessing* random perturbation incurs an error in the computation (just as the deflation step does), but if the scale of  $\gamma$  is chosen appropriately, it will not preclude one from being able to obtain an accurate approximate version of the eigenvalue problem. Then, putting (4) and (5) together, in [BGVSb] we were able to show that a randomized version of the QR algorithm can diagonalize any input matrix  $A \in \mathbb{C}^{n \times n}$  with accuracy  $\delta$  in  $O(n^3 \log(n/\delta)^2 \log \log(n/\delta)^2)$  operations.

**Our shifting strategy.** As in [DT71] and other works that served as inspiration (e.g. [Bat94]), we used the lower subdiagonal entries of the iterates  $H_t$  to keep track of progress towards convergence. Specifically, to analyze the shifting strategy of degree  $k$  mentioned in Theorem 1, we used the potential function  $\psi_k$  which on a Hessenberg matrix  $H$  is defined as

$$\psi_k(H) = |H(n, n-1)H(n-1, n-2)\cdots H(n-k+1, n-k)|^{\frac{1}{k}}.$$

Then, as in [EH75, WG02], we used a mixed strategy consisting of a main shift and an exceptional shift. If at time  $t$  an iteration with the main shift *did not* satisfy that  $\psi_k(H_{t+1}) \leq .8\psi_k(H_t)$  (i.e. if progress is not being made), then our shifting strategy recomputes  $H_{t+1}$ , this time using the exceptional shift, and in [BGVSA] we show that provided that  $\kappa_V$  of the input matrix satisfies the bound (5) the exceptional shift *does* succeed in guaranteeing  $\psi_k(H_{t+1}) \leq .8\psi_k(H_t)$ . Our mixed strategy then guarantees a geometric decrease of the quantity  $\psi_k(H_t)$ , which in turn implies that  $\delta$ -decoupling will occur after  $O(\log(1/\delta))$  iterations.

**A final caveat.** Our theoretical algorithm is not a prescription for practitioners and does not seek to replace the current very efficient LAPACK routines, which have been fine-tuned over the

---

<sup>4</sup>This theorem was not stated verbatim and strictly speaking only  $k$ 's that are powers of 2 were treated in the paper, however, the ideas in [BGVSA] yield, with very little extra work, the theorem stated here.

<sup>5</sup>That is, and  $n \times n$  matrix with i.i.d. complex Gaussian entries of variance  $\frac{1}{n}$ .

decades and for which several patches have been added to avoid convergence failures. We do warn the reader however, that such routines are by now quite sophisticated and do not come with theoretical guarantees. This does make one wonder if there is an algorithm that is as efficient as the existing implementations, but that is conceptually simple and for which one can give rigorous guarantees.

## References

- [ABB<sup>+</sup>18] Diego Armentano, Carlos Beltrán, Peter Bürgisser, Felipe Cucker, and Michael Shub. A stable, polynomial-time algorithm for the eigenpair problem. *Journal of the European Mathematical Society*, 20(6):1375–1437, 2018.
- [Bat94] Steve Batterson. Convergence of the Francis shifted QR algorithm on normal matrices. *Linear algebra and its applications*, 207:181–195, 1994.
- [BGVKS24] Jess Banks, Jorge Garza-Vargas, Archit Kulkarni, and Nikhil Srivastava. Overlaps, eigenvalue gaps, and pseudospectrum under real ginibre and absolutely continuous perturbations. In *Annales de l’Institut Henri Poincaré (B) Probabilités et statistiques*, volume 60, pages 2736–2766. Institut Henri Poincaré, 2024.
- [BGVSa] Jess Banks, Jorge Garza-Vargas, and Nikhil Srivastava. Global convergence of Hessenberg shifted QR I: Dynamics. *To appear in Foundations of Computational Mathematics*.
- [BGVSb] Jess Banks, Jorge Garza-Vargas, and Nikhil Srivastava. Global convergence of Hessenberg shifted QR II: Numerical stability. *To appear in SIAM Journal on Matrix Analysis and Applications*.
- [BGVSc] Jess Banks, Jorge Garza-Vargas, and Nikhil Srivastava. Global convergence of Hessenberg shifted QR III: Approximate Ritz values via shifted inverse iteration. *To appear in SIAM Journal on Matrix Analysis and Applications*.
- [BKMS21] Jess Banks, Archit Kulkarni, Satyaki Mukherjee, and Nikhil Srivastava. Gaussian regularization of the pseudospectrum and Davies’ conjecture. *Communications on Pure and Applied Mathematics*, 74(10):2114–2131, 2021.
- [Dav08] E. Brian Davies. Approximate diagonalization. *SIAM Journal on Matrix Analysis and Applications*, 29(4):1051–1064, 2008.
- [Dem97] James W Demmel. *Applied numerical linear algebra*. SIAM, 1997.
- [DS00] Jack Dongarra and Francis Sullivan. Guest editors introduction to the top 10 algorithms. *Computing in Science & Engineering*, 2(01):22–23, 2000.
- [DT71] Theodorus J Dekker and Joseph F Traub. The shifted QR algorithm for Hermitian matrices. *Linear Algebra Appl.*, 4:137–154, 1971.
- [EH75] Patricia J Eberlein and C. P. Huang. Global convergence of the QR algorithm for unitary matrices with some results for normal matrices. *SIAM Journal on Numerical Analysis*, 12(1):97–104, 1975.

- [Fra61] John GF Francis. The QR transformation a unitary analogue to the LR transformation—part 1. *The Computer Journal*, 4(3):265–271, 1961.
- [Fra62] John GF Francis. The QR transformation—part 2. *The Computer Journal*, 4(4):332–345, 1962.
- [GU09] Gene Golub and Frank Uhlig. The QR algorithm: 50 years later its genesis by John Francis and Vera Kublanovskaya and subsequent developments. *IMA Journal of Numerical Analysis*, 29(3):467–485, 2009.
- [HDG<sup>+</sup>15] Nicholas J Higham, Mark R Dennis, Paul Glendinning, Paul A Martin, Fadil Santosa, and Jared Tanner. *The Princeton companion to applied mathematics*. Princeton University Press Princeton, NJ, USA:, 2015.
- [JSS21] Vishesh Jain, Ashwin Sah, and Mehtaab Sawhney. On the real Davies’ conjecture. *The Annals of Probability*, 49(6):3011–3031, 2021.
- [Kub62] Vera N Kublanovskaya. On some algorithms for the solution of the complete eigenvalue problem. *USSR Computational Mathematics and Mathematical Physics*, 1(3):637–657, 1962.
- [Mol78] Cleve B Moler. Three research problems in numerical linear algebra. *Numerical Analysis*, 22:1–18, 1978.
- [Par66] Beresford Parlett. Singular and invariant matrices under the QR transformation. *Mathematics of Computation*, 20(96):611–615, 1966.
- [Par74] Beresford N Parlett. The Rayleigh quotient iteration and some generalizations for nonnormal matrices. *Mathematics of Computation*, 28(127):679–693, 1974.
- [Par98] Beresford N Parlett. *The symmetric eigenvalue problem*. SIAM, 1998.
- [Sma97] Steve Smale. Complexity theory and numerical analysis. *Acta numerica*, 6:523–551, 1997.
- [ST04] Daniel A Spielman and Shang-Hua Teng. Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time. *Journal of the ACM (JACM)*, 51(3):385–463, 2004.
- [Tis96] Francoise Tisseur. Backward stability of the QR algorithm. Technical report, 239, UMR 5585, Lyon Saint-Etienne, 1996.
- [Wat82] David S Watkins. Understanding the QR algorithm. *SIAM review*, 24(4):427–440, 1982.
- [Wat08] David S Watkins. The QR algorithm revisited. *SIAM review*, 50(1):133–145, 2008.
- [WG02] Tai-Lin Wang and William Gragg. Convergence of the shifted QR algorithm for unitary Hessenberg matrices. *Mathematics of computation*, 71(240):1473–1496, 2002.
- [Wil68] James Hardy Wilkinson. Global convergence of tridiagonal QR algorithm with origin shifts. *Linear Algebra and its Applications*, 1(3):409–420, 1968.

# Flexible Golub-Kahan Factorization for Linear Inverse Problems

Silvia Gazzola

## Abstract

Discrete linear inverse problems arising in many applications in Science and Engineering are formulated as the solution of large-scale linear systems of equations of the form

$$\mathbf{A}\mathbf{x}_{\text{true}} + \mathbf{n} = \mathbf{b}, \quad (1)$$

where the discretized forward operator  $\mathbf{A} \in \mathbb{R}^{m \times n}$  is large-scale with ill-determined rank, and  $\mathbf{n} \in \mathbb{R}^m$  are some unknown perturbations (noise) affecting the available data  $\mathbf{b} \in \mathbb{R}^m$ . In this setting, in order to recover a meaningful approximation of  $\mathbf{x}_{\text{true}} \in \mathbb{R}^n$ , one should regularize (1).

In this talk we consider variational regularization methods that compute an approximation  $\mathbf{x}_{\text{reg}}$  of  $\mathbf{x}_{\text{true}}$  as

$$\mathbf{x}_{\text{reg}} = \arg \min_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{R}(\mathbf{A}\mathbf{x} - \mathbf{b})\|_p^p + \lambda \|\mathbf{L}\mathbf{x}\|_q^q, \quad \text{where } \lambda \geq 0, p, q > 0, \mathbf{R} \in \mathbb{R}^{m \times m} \mathbf{L} \in \mathbb{R}^{l \times n}. \quad (2)$$

In the above formulation, when  $p = q = 2$ , many standard numerical linear algebra tools can be employed to approximate  $\mathbf{x}_{\text{reg}}$ : these include the SVD of  $\mathbf{A}$  (when  $\mathbf{A}$  has some exploitable structure and  $\mathbf{L}$  is the identity), early termination of Krylov solvers for (1) (when  $\lambda = 0$ ), and hybrid projection methods. We refer to [2] for a recent survey of these strategies. However, by properly setting  $p, q \neq 2$ , better approximations of  $\mathbf{x}_{\text{true}}$  can be obtained in many scenarios, including: when the noise  $\mathbf{n}$  is not Gaussian, nor white, and/or when wanting to enforce sparsity onto  $\mathbf{L}\mathbf{x}_{\text{reg}}$  (e.g., in the compressive sensing framework, when  $\mathbf{A}$  is heavily underdetermined). Although many classes of well-established optimization methods are usually employed to handle the non-smooth and possibly non-convex instances of (2), in the last decades a number of new solvers based on ‘non-standard’ (such as flexible [1, 4] or generalized [5]) Krylov methods have been successfully considered for this purpose; see also [3, 7]. Even though the common starting point of such ‘non-standard’ Krylov solvers is the reformulation of a smoothed version of (2) as an iteratively reweighted least squares problem, flexible Krylov methods for  $p = 2$  are typically more efficient and stable than generalized Krylov methods, while the latter can handle also the  $p \neq 2$  case and many options for  $\mathbf{L}$ .

This talk introduces new solvers for (2), based on a new flexible Golub-Kahan factorization of the kind

$$\widehat{\mathbf{A}}\mathbf{Z}_k = \mathbf{U}_{k+1}\bar{\mathbf{M}}_k, \quad \widehat{\mathbf{A}}^\top \mathbf{Y}_{k+1} = \mathbf{V}_{k+1}\mathbf{T}_{k+1},$$

where:  $\mathbf{U}_{k+1} \in \mathbb{R}^{m \times (k+1)}$  and  $\mathbf{V}_k \in \mathbb{R}^{n \times k}$  have orthonormal columns  $\mathbf{u}_i$  ( $i = 1, \dots, k+1$ ) and  $\mathbf{v}_i$  ( $i = 1, \dots, k$ ), respectively;  $\mathbf{Z}_k = [\mathbf{L}_1^\dagger \mathbf{v}_1, \dots, \mathbf{L}_k^\dagger \mathbf{v}_k]$ ,  $\mathbf{Y}_{k+1} = [\mathbf{R}_1^\dagger \mathbf{u}_1, \dots, \mathbf{R}_{k+1}^\dagger \mathbf{u}_{k+1}]$ ;  $\bar{\mathbf{M}}_k \in \mathbb{R}^{(k+1) \times k}$  is upper Hessenberg and  $\mathbf{T}_{k+1} \in \mathbb{R}^{(k+1) \times (k+1)}$  is upper triangular;  $k \ll \min\{m, n\}$ . The  $i$ th approximate solution of  $\mathbf{x}_{\text{reg}}$  in (2) is defined as

$$\mathbf{x}_i = \mathbf{Z}_i \arg \min_{\mathbf{s} \in \mathbb{R}^i} \|f(\mathbf{T}_{i+1}, \bar{\mathbf{M}}_i)\mathbf{s} - \mathbf{c}_i\|_2^2 + \lambda_i \|\mathbf{S}_i \mathbf{s}\|_2^2,$$

where the regularization parameter  $\lambda_i$  is adaptively set,  $\mathbf{S}_i \in \mathbb{R}^{i \times i}$  is a regularization matrix for the projected variable  $\mathbf{s}$ ,  $\mathbf{c}_i$  is a projected right-hand side, and  $f$  compactly denotes products and/or sums of (possibly slight modifications and transposes of) both matrices  $\mathbf{T}_{i+1}$  and  $\bar{\mathbf{M}}_i$ ; different choices of  $f$  and  $\mathbf{S}_i$  define different solvers. Note that  $\mathbf{R}_i^\dagger$  and  $\mathbf{L}_i^\dagger$  act as variable ‘preconditioners’ for the constraint and solution subspaces, respectively; their role is to enforce iteration-dependent information useful for a successful regularization. Different choices of  $\widehat{\mathbf{A}}$ ,  $\mathbf{R}_i^\dagger$  and  $\mathbf{L}_i^\dagger$  allow to handle different instances of (2). Namely:

- (a)  $\widehat{\mathbf{A}} = [\mathbf{A}^\top, \mathbf{L}^\top]^\top$ ,  $\mathbf{R}_i^\dagger = \text{diag}(\mathbf{I}, \lambda_i \mathbf{I})$  and  $\mathbf{L}_i^\dagger = \mathbf{I}$  solves Tikhonov problems in general form in the 2-norm, with adaptive regularization parameter choice strategy; this provides an alternative to the generalized Krylov method in [6].
- (b)  $\widehat{\mathbf{A}} = \mathbf{A}$ ,  $\mathbf{R}_i^\dagger = \mathbf{I}$  and  $\mathbf{L}_i^\dagger = \text{diag}(g_q^{-1}(\mathbf{x}_{i-1}))$  (where  $g_q$  is a function that depends on the  $q$ -norm and is applied entry-wise) solves the so-called  $\ell_2 - \ell_q$  regularized problem, with adaptive regularization parameter choice strategy; this coincides with the basic version of the method in [1] (and can be reformulated to cover all the options in [1]).
- (c)  $\widehat{\mathbf{A}} = [\mathbf{A}^\top, \mathbf{L}^\top]^\top$ ,  $\mathbf{R}_i^\dagger = \text{diag}(g_p(\mathbf{R}(\mathbf{A}\mathbf{x}_{i-1} - \mathbf{b})), \lambda_i g_q(\mathbf{L}\mathbf{x}_{i-1}))$  (where, similarly to  $g_q$ ,  $g_p$  is a function that depends on the  $p$ -norm and is applied entry-wise) and  $\mathbf{L}_i^\dagger = \mathbf{I}$  solves the so-called  $\ell_p - \ell_q$  regularized problem, with adaptive regularization parameter choice strategy; this extends the methods in [1] and provides an alternative to the generalized Krylov method in [5]. As a particular case, setting  $\lambda_i = 0$ ,  $i = 1, 2, \dots$  solves a  $p$ -norm residual minimization problem.

The new solvers are theoretically analyzed by providing optimality properties and by studying the effect of variations in  $\mathbf{R}_i^\dagger$  and  $\mathbf{L}_i^\dagger$  on their convergence. The new solvers can efficiently be applied to both underdetermined and overdetermined problems, and successfully extend the current flexible Krylov solvers to handle different matrices  $\mathbf{R}$  (typically the inverse square root of the noise covariance matrix), as well as regularization matrices  $\mathbf{L}$  whose  $\mathbf{A}$ -weighted generalized pseudo-inverse cannot be cheaply computed.

Numerical experiments on inverse problems in imaging, such as deblurring and computed tomography, show that the new solvers are competitive with other state-of-the-art nonsmooth and non-convex optimization methods, as well as generalized Krylov methods.

## References

- [1] J. Chung, S. Gazzola. Flexible Krylov Methods for  $\ell_p$  Regularization. *SIAM J. Sci. Comput.*, 41(5), pp. S149–S171, 2019.
- [2] J. Chung, S. Gazzola. Computational Methods for Large-Scale Inverse Problems: A Survey on Hybrid Projection Methods. *SIAM Review*, 66(2), pp. 205–284, 2024.
- [3] J. Cornelis, W. Vanroose. Projected Newton method for noise constrained  $\ell_p$  regularization. *Inverse Problems*, 36 125004, 2020.
- [4] S. Gazzola, J. G. Nagy, and M. Sabatè Landman. Iteratively reweighted FGMRES and FLSQR for sparse reconstruction. *SIAM J. Sci. Comput.*, 43, pp. S47–S69, 2021.
- [5] G. Huang, A. Lanza, S. Morigi, L. Reichel, F. Sgallari. Majorization–minimization generalized Krylov subspace methods for  $\ell_p - \ell_q$  optimization applied to image restoration. *BIT Numerical Mathematics*, 57(2), pp. 351–378, 2017.
- [6] J. Lampe, L. Reichel, H. Voss. Large-scale Tikhonov regularization via reduction by orthogonal projection. *Linear Algebra Appl.*, 436(8), pp. 2845–2865, 2012.
- [7] M. Sabate Landman, J. Chung. Flexible Krylov Methods for Group Sparsity Regularization. *Physica Scripta*, 2024.

# Numerical Approximation of the Distance to Singularity for Matrix-valued Functions

Miryam Gnazzo, Nicola Guglielmi

## Abstract

We consider matrix-valued functions in the form

$$\mathcal{F}(\lambda) = \sum_{i=1}^d f_i(\lambda) A_i,$$

where  $A_i \in \mathbb{C}^{n \times n}$  and  $f_i : \mathbb{C} \mapsto \mathbb{C}$  entire functions for  $i = 1, \dots, d$ . Given a regular matrix-valued function, that is a function whose determinant  $\det(\mathcal{F}(\lambda))$  is not identically zero, we discuss the problem of computing the singular matrix-valued function closest to it in the Frobenius norm. This problem is known in literature as the computation of the *distance to singularity* for  $\mathcal{F}(\lambda)$ . More precisely, we are interested in approximating the nearest matrix-valued function  $\mathcal{F}(\lambda) + \Delta\mathcal{F}(\lambda)$  such that

$$\det(\mathcal{F}(\lambda) + \Delta\mathcal{F}(\lambda)) \equiv 0, \quad (1)$$

where  $\Delta\mathcal{F}(\lambda) = \sum_{i=1}^d f_i(\lambda) \Delta A_i$ , with  $\Delta A_i \in \mathbb{C}^{n \times n}$ , for  $i = 1, \dots, d$ . The problem of the numerical approximation of the distance to singularity for  $\mathcal{F}(\lambda)$  is well-known to be challenging, even for linear cases, where it reduces to the computation of the distance to singularity for matrix pencils [2]. Recently, the problem has gained increasing attention and numerical approaches have been developed both for matrix pencils, as in [4], and in the case of polynomial nonlinearities, as in [3]. Nevertheless, none of the currently available techniques has been applied to the approximation of the distance to singularity for general nonlinearities.

The solution of this problem for general nonlinearities becomes important in the context of differential algebraic equations and delay differential algebraic equations. Indeed, in this framework, the characteristic equation associated with the differential equation has the form

$$\det(A_1 - \lambda A_2 + e^{-\tau_1 \lambda} A_3 + \dots + e^{-\tau_k \lambda} A_{k+2}) = 0,$$

and the eigenstructure of the matrix-valued function is a crucial tool in the solvability of the delay differential algebraic equation with discrete constant delays  $\tau_j$ , for  $j = 1, \dots, k$ .

As an illustrative example, indeed, we underline that in many practical cases the function  $\mathcal{D}(\lambda) = A_1 - \lambda A_2 + e^{-\tau \lambda} A_3$ , in presence of a small delay  $\tau$ , may be numerically singular, even in situations where the pencil  $A_1 - \lambda A_2$  is regular, leading to a severe ill-posedness of the problem. In this setting, an a-priori computation of the distance to singularity associated with  $\mathcal{D}(\lambda)$  would act like a measure for the lack of robustness of the differential equation.

A major difficulty is due to the presence of nonlinearities in the matrix-valued function, which represents a delicate point of the problem, since a general matrix-valued function may have an infinite number of eigenvalues. Observe that this feature of the problem does not arise when dealing with matrix pencils and matrix polynomials, and, to our knowledge, this characteristic may prevent the extension of the currently available methods to nonlinearities different from the polynomial one.

In this talk, we propose a method for the numerical approximation of the distance to singularity for nonlinear matrix-valued functions [5]. We show that the problem can be rephrased as a nearness

problem and the property of singularity of the matrix-valued function is translated into a discrete numerical constraint for a suitable minimization problem. Nevertheless, this resulting problem turns out to be highly non-convex. In order to solve it, we propose an iterative procedure made by two nested optimization subproblems, of whose the inner one introduces a constraint gradient system of matrix differential equations and the outer one consists in the optimization of the norm  $\| [\Delta A_1 \dots \Delta A_d] \|_F$  via a Newton-like method.

We dedicate special attention to the numerical treatment of the continuous constraint (1), since a careful translation of this condition into its discrete version is an essential step for the applicability of our numerical approach. To this purpose, we employ results from approximation theory for analytic functions [1].

In many practical applications, such as in the ones arising from engineering and mechanical modeling, matrix-valued functions  $\mathcal{F}(\lambda)$  are often endowed with additional structures. Indeed, the coefficients  $A_i$  frequently encode data coming from the underlying application: for instance, they may represent the stiffness or damping matrix in a PDE setting. In this framework, it is important to employ an approach with the desired feature of addressing different structures, in which case the search of the closest singular function is restricted to the class of functions preserving the structure of the matrices.

Nevertheless, the possibility of including additional structural constraints into a nearness problem is not an easy task and it leads to a more challenging version of the problem. Indeed, techniques that are able to compute the unstructured distance to singularity often can not be directly extended to their structured counterparts.

One of the advantages of the nested approach we propose consists in the fact that it can be naturally extended to its structured version, with minor changes, and, therefore, it is able to tackle nearness problems with the additional constraint of structured perturbations. In the talk, we practically demonstrate this feature of our technique, by providing a number of case studies. For example, the method allows us to limit the perturbations to just a few matrices and also to include individual structures, such as the preservation of the sparsity pattern of one or more matrices  $A_i$ , and collective-like properties, like a palindromic structure of the function  $\mathcal{F}(\lambda)$ .

## References

- [1] A.P. Austin, P. Kravanja and L.N. Trefethen. *Numerical Algorithms Based on Analytic Function Values at Roots of Unity*. SIAM Journal on Numerical Analysis. 52, 1795–1821, 2014.
- [2] R. Byers, C. He and V. Mehrmann. *Where is the nearest non-regular pencil?* Linear Algebra Appl. 285, 81–105, 1998.
- [3] B. Das and S. Bora. *Nearest rank deficient matrix polynomials*. Linear Algebra Appl. 674, 304–350, 2023.
- [4] F. Dopico, V. Noferini and L. Nyman. *A Riemannian Optimization Method to Compute the Nearest Singular Pencil*. SIAM J. on Matrix Anal. Appl. 45, 2007–2038, 2024.
- [5] M. Gnazzo and N. Guglielmi. *On the numerical approximation of the distance to singularity for matrix-valued functions*. Preprint, 2023.

# $\mathcal{H}_2$ optimal model reduction of linear systems with quadratic outputs: from rational function interpolation to data-driven modeling

*Sean Reiter , Ion Victor Gosea, Igor Pontes Duff, Serkan Gugercin*

## Abstract

$\mathcal{H}_2$  optimal reduction of linear dynamical systems represents a long-lasting, worthwhile problem in system-theoretical model order reduction. In this short note, we propose extensions of first-order necessary conditions, both based on system Gramians and transfer functions, to the  $\mathcal{H}_2$  problem for the class of linear systems with quadratic outputs.

## 1 Introduction

Model-order reduction (MOR) refers to the procedure by which one approximates a large-scale dynamical system, modeled by systems of ordinary differential equations, with a comparatively lower-order surrogate model which can be used as a cheap-to-evaluate surrogate in downstream computational tasks, such as optimization or control. In order to be an effective surrogate, the computed reduced-order model (ROM) should recover the dominant input-to-output response characteristics of the original complex system, as well as preserve qualitative features like internal structures. We refer to [1,2] for more details on system-theoretical MOR, since this category is of interest to us.

The primary consideration of this work is the development of methods for the MOR of linear dynamical systems which contain quadratic output functions, or linear quadratic-output (LQO) systems. Here, we develop extensions of classical MOR approaches applicable solely to systems with linear dynamics and linear outputs. Our contributions are threefold: First, we consider the  $\mathcal{H}_2$  optimal model reduction problem, and derive first-order necessary conditions for  $\mathcal{H}_2$  optimality based on rational transfer function interpolation. These provide a natural extension of the well-known interpolation-based  $\mathcal{H}_2$  optimality framework of Meier and Leunberger [7,8] for linear model reduction. Based on the developed theoretical optimality framework, we propose an extension of the well-known iterative rational Krylov algorithm (IRKA) [7] for linear  $\mathcal{H}_2$  optimal model reduction. Finally, we show how to compute  $\mathcal{H}_2$  optimal reduced models using only evaluations of the linear- and quadratic-output transfer functions.

## 2 Transfer functions, norms and MOR of LQO systems

In this work, we consider large-scale dynamical systems with linear dynamics and outputs which are (up to) quadratic functions of the state vector. In state-space, such systems are formulated as

$$\Sigma : \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{M}(\mathbf{x}(t) \otimes \mathbf{x}(t)), \end{cases} \quad (1)$$

where  $\mathbf{x}(t) \in \mathbb{R}^n$  is the system's internal state,  $\mathbf{u}(t) \in \mathbb{R}^m$  are the control inputs, and  $\mathbf{y}(t) \in \mathbb{R}^p$  are the observed outputs. The system matrices satisfy  $\mathbf{A} \in \mathbb{R}^{n \times n}$ ,  $\mathbf{B} \in \mathbb{R}^{n \times m}$ ,  $\mathbf{C} \in \mathbb{R}^{p \times n}$  and  $\mathbf{M} \in \mathbb{R}^{p \times n^2}$ . We assume that the system in (1) is asymptotically stable, i.e., the eigenvalues of  $\mathbf{A}$

to be in the left-half plane. Systems that consider quadratic observables as quantities of interest arise in a variety of applications, and particularly whenever one is interested in observing quantities computed as the product of time or frequency-domain components of the state [4, 11].

The frequency-domain response of system (1) is fully specified by 2 rational transfer functions [4, 6]:

$$\mathbf{H}_1(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} \quad \text{and} \quad \mathbf{H}_2(s, z) = \mathbf{M}((s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B} \otimes (z\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}). \quad (2)$$

The first function  $\mathbf{H}_1(s)$  is the typical transfer function of a linear-output system, and describes the transfer from input  $\mathbf{u}(t)$  to output  $\mathbf{y}_1(t) := \mathbf{C}\mathbf{x}(t)$ , which is linear in  $\mathbf{x}(t)$ . The second bivariate function  $\mathbf{H}_2(s, z)$  the transfer from input  $\mathbf{u}(t)$  to output  $\mathbf{y}_2(t) := \mathbf{M}(\mathbf{x}(t) \otimes \mathbf{x}(t))$ , which is quadratic in  $\mathbf{x}(t)$ . The  $\mathcal{H}_2$  norm for systems of the form (1) can be defined via these transfer functions as [4, 5]

$$\|\Sigma\|_{\mathcal{H}_2}^2 := \frac{1}{2\pi} \int_{-\infty}^{\infty} \|\mathbf{H}_1(i\omega)\|_F^2 d\omega + \frac{1}{(2\pi)^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \|\mathbf{H}_2(i\omega_1, i\omega_2)\|_F^2 d\omega_1 d\omega_2. \quad (3)$$

We note that when  $\mathbf{M} = \mathbf{0}$ , it implies that  $\mathbf{H}(s, z) = 0$ , and the formula above simplifies to the first integral term only, which is the standard formula as used in [7].

In practical applications, the state dimension  $n$  can be rather large, e.g., in the order of the millions, and any repeated action involving the full-order model (FOM) (1) becomes prohibitively expensive. Model reduction seeks to remedy this problem with the construction of cheap-to-evaluate surrogate models having the same form as (1), but described by a comparatively much smaller number of differential equations. Mathematically, this amounts to computing a system

$$\widehat{\Sigma} : \begin{cases} \dot{\widehat{\mathbf{x}}}(t) = \widehat{\mathbf{A}}\widehat{\mathbf{x}}(t) + \widehat{\mathbf{B}}\mathbf{u}(t), \\ \widehat{\mathbf{y}}(t) = \widehat{\mathbf{C}}\widehat{\mathbf{x}}(t) + \widehat{\mathbf{M}}(\widehat{\mathbf{x}}(t) \otimes \widehat{\mathbf{x}}(t)), \end{cases} \quad (4)$$

with a significantly reduced dimension  $1 \leq r \ll n$ .  $\widehat{\mathbf{x}}(t) \in \mathbb{R}^r$  contains the reduced-order state variables, and  $\widehat{\mathbf{y}}(t) \in \mathbb{R}^p$  are the approximateds output. The reduced-order matrix operators satisfy  $\widehat{\mathbf{A}} \in \mathbb{R}^{r \times r}$ ,  $\widehat{\mathbf{B}} \in \mathbb{R}^{r \times m}$ ,  $\widehat{\mathbf{C}} \in \mathbb{R}^{p \times r}$ , and  $\widehat{\mathbf{M}} \in \mathbb{R}^{p \times r^2}$ . In order to be an effective surrogate, the reduced model (4) should replicate the input-to-output response characteristics of the large-scale system (1). In order words, for a given tolerance  $\tau > 0$ , the output deviation should satisfy  $\|\mathbf{y} - \widehat{\mathbf{y}}\| \leq \tau \|\mathbf{u}\|$  in an appropriate norm for a range of admissible inputs  $\mathbf{u}$ .

Suppose that one is interested in controlling the  $\mathcal{L}_\infty^p$ , or “worst case” deviation in the output  $\|\mathbf{y} - \widehat{\mathbf{y}}\|_{\mathcal{L}_\infty^p} := \sup_{t \geq 0} \|\mathbf{y}(t) - \widehat{\mathbf{y}}(t)\|_\infty$ . Significantly, one can show following error bound [4]:

$$\|\mathbf{y} - \widehat{\mathbf{y}}\|_{\mathcal{L}_\infty^p} \leq \|\Sigma - \widehat{\Sigma}\|_{\mathcal{H}_2} (\|\mathbf{u}\|_{\mathcal{L}_2^m}^2 + \|\mathbf{u} \otimes \mathbf{u}\|_{\mathcal{L}_2^{m^2}}^2)^{1/2}. \quad (5)$$

In other words, the  $\mathcal{H}_2$  model error bounds the  $\mathcal{L}_\infty^p$  output error. Based on the bound (5), we consider the  $\mathcal{H}_2$  optimal model reduction problem for the LQO system class (1). Given the system in (1), find a ROM such that

$$\min_{\dim(\widehat{\Sigma})=r} \mathcal{J}(\widehat{\Sigma}), \quad \mathcal{J}(\widehat{\Sigma}) := \|\Sigma - \widehat{\Sigma}\|_{\mathcal{H}_2}^2. \quad (6)$$

### 3 One main result

We follow here the results in [9]. To simplify the approximation problem, we assume the approximate system in (4) has simple poles. Then, the reduced-order linear- and quadratic-output transfer

functions can be expressed in pole-residue form:

$$\widehat{\mathbf{H}}_1(s) = \sum_{i=1}^r \frac{\mathbf{c}_i \mathbf{b}_i^\top}{s - \lambda_i} \quad \text{and} \quad \widehat{\mathbf{H}}_2(s, z) = \sum_{j=1}^r \sum_{k=1}^r \frac{\mathbf{m}_{j,k} (\mathbf{b}_j \otimes \mathbf{b}_k)^\top}{(s - \lambda_j)(z - \lambda_k)}, \quad (7)$$

where  $\mathbf{b}_k \in \mathbb{C}^m$ , and  $\mathbf{c}_k \in \mathbb{C}^p$ ,  $\mathbf{m}_{j,k} \in \mathbb{C}^p$ , for all  $j, k = 1, \dots, r$ . We define  $\mathbf{c}_i \mathbf{b}_i^\top \in \mathbb{C}^{p \times m}$  and  $\mathbf{m}_{j,k} (\mathbf{b}_j \otimes \mathbf{b}_k)^\top \in \mathbb{C}^{p \times m^2}$  to be the *residues* of  $\widehat{\mathbf{H}}_1(s)$  and  $\widehat{\mathbf{H}}_2(s, z)$  corresponding to  $\lambda_i$  and  $(\lambda_j, \lambda_k)$ , respectively. We are able to show that

$$\|\Sigma - \widehat{\Sigma}\|_{\mathcal{H}_2}^2 = \|\Sigma\|_{\mathcal{H}_2}^2 - 2 \left( \sum_{i=1}^r \mathbf{c}_i^\top \mathbf{H}_1(-\lambda_i) \mathbf{b}_i + \sum_{j=1}^r \sum_{k=1}^r \mathbf{m}_{j,k}^\top \mathbf{H}_2(-\lambda_j, -\lambda_k) (\mathbf{b}_j \otimes \mathbf{b}_k) \right) + \|\widehat{\Sigma}\|_{\mathcal{H}_2}^2. \quad (8)$$

This makes the  $\mathcal{H}_2$  optimal model reduction problem tractable by minimally parameterizing the ROM in (4) in terms of the transfer function poles and residues.

**Theorem 1** Suppose that  $\widehat{\Sigma}$  has simple poles  $\lambda_1, \dots, \lambda_r \in \mathbb{C}_-$ , and is a local minimizer of the squared  $\mathcal{H}_2$  error  $\|\Sigma - \widehat{\Sigma}\|_{\mathcal{H}_2}^2$ . Then, for all  $i, j, k = 1, \dots, r$ , it holds that

$$\begin{aligned} \mathbf{0} &= \left( \mathbf{H}_1(-\lambda_i) - \widehat{\mathbf{H}}_1(-\lambda_i) \right) \mathbf{b}_i, \\ \mathbf{0} &= \left( \mathbf{H}_2(-\lambda_j, -\lambda_k) - \widehat{\mathbf{H}}_2(-\lambda_j, -\lambda_k) \right) (\mathbf{b}_j \otimes \mathbf{b}_k), \\ \mathbf{0} &= \mathbf{c}_k^\top \left( \mathbf{H}_1(-\lambda_k) - \widehat{\mathbf{H}}_1(-\lambda_k) \right) + \sum_{\ell=1}^r \mathbf{m}_{k,\ell}^\top \left( \mathbf{H}_2(-\lambda_k, -\lambda_\ell) - \widehat{\mathbf{H}}_2(-\lambda_k, -\lambda_\ell) \right) (\mathbf{I}_m \otimes \mathbf{b}_\ell) \\ &\quad + \sum_{\ell=1}^r \mathbf{m}_{\ell,k}^\top \left( \mathbf{H}_2(-\lambda_\ell, -\lambda_k) - \widehat{\mathbf{H}}_2(-\lambda_\ell, -\lambda_k) \right) (\mathbf{b}_\ell \otimes \mathbf{I}_m), \\ \mathbf{0} &= \mathbf{c}_k^\top \left( \frac{d}{ds} \mathbf{H}_1(-\lambda_k) - \frac{d}{ds} \widehat{\mathbf{H}}_1(-\lambda_k) \right) \mathbf{b}_k + \sum_{\ell=1}^r \mathbf{m}_{k,\ell}^\top \left( \frac{\partial}{\partial s_1} \mathbf{H}_2(-\lambda_k, -\lambda_\ell) - \frac{\partial}{\partial s_1} \widehat{\mathbf{H}}_2(-\lambda_k, -\lambda_\ell) \right) (\mathbf{b}_k \otimes \mathbf{b}_\ell) \\ &\quad + \sum_{\ell=1}^r \mathbf{m}_{\ell,k}^\top \left( \frac{\partial}{\partial s_2} \mathbf{H}_2(-\lambda_\ell, -\lambda_k) - \frac{\partial}{\partial s_2} \widehat{\mathbf{H}}_2(-\lambda_\ell, -\lambda_k) \right) (\mathbf{b}_\ell \otimes \mathbf{b}_k). \end{aligned}$$

In other words, tangential interpolation is a necessary condition for  $\mathcal{H}_2$  optimality. We also note that when  $\mathbf{M} = \mathbf{0}$ , it implies that the formulae above simplify accordingly to the standard interpolation-based FONCs for classical linear systems, as in [7, 8].

## 4 Summary of all proposed results

Based on the bound in (5), we have considered the  $\mathcal{H}_2$  optimal model reduction problem for the class of systems in (1). We went about this in two different ways, corresponding to two types of FONCs, namely for the first one mentioned earlier introduced in [8]. Then, for the second, e.g., the Gramian-based FONCs as introduced in [13], we analyzed it in [10].

Our contributions to the interpolation-based formulation are threefold:

- A. First, we derive interpolation-based first-order necessary conditions for  $\mathcal{H}_2$  optimal model reduction. These amount to tangential interpolation of a weighted sum of the transfer functions in (2), and generalize the analogous optimality conditions for linear  $\mathcal{H}_2$  model reduction. We show how to enforce these conditions in the construction of the ROM using projection.

- B. Secondly, we show that these conditions are equivalent to the Gramian-based  $\mathcal{H}_2$  optimality conditions for LQO systems as in (1).
- C. Thirdly, we propose an extension of TF-IRKA in [3] to systems of the form (1). The algorithm enforces the necessary  $\mathcal{H}_2$  optimality conditions at every step and produces locally  $\mathcal{H}_2$  optimal approximants upon convergence. Additionally, at every step, the matrices of the ROM are computed solely in terms of data, i.e., samples of the two transfer functions in (2).

Due to space limitations, we are not able to go into a detailed analysis of the results concerning Gramian-based FONCs, as presented in [10]. In short, we derive gradients of the squared  $\mathcal{H}_2$  system error with respect to the system matrices of the LQO-ROM as parameters. The stationary points of these gradients directly yield Gramian-based FONCs for  $\mathcal{H}_2$  optimality. These results generalize the analogous Gramian-based FONCs for linear  $\mathcal{H}_2$  optimal model [12, 13] to the LQO setting. We also show that a  $\mathcal{H}_2$  optimal LQO-ROM is necessarily defined by Petrov-Galerkin projection. The relevant projection matrices are obtained as solutions to a pair of Sylvester equations.

## References

- [1] A. C. Antoulas. *Approximation of Large-Scale Dynamical Systems*, volume 6 of *Adv. Des. Control*. SIAM Publications, Philadelphia, PA, 2005.
- [2] A. C. Antoulas, C. A. Beattie, and S. Gugercin. *Interpolatory Methods for Model Reduction*. Computational Science & Engineering. SIAM, Philadelphia, PA, 2020.
- [3] C. A. Beattie and S. Gugercin. Realization-independent  $\mathcal{H}_2$ -approximation. In *51st IEEE Conference on Decision and Control (CDC)*, pages 4953–4958, 2012.
- [4] P. Benner, P. Goyal, and I. Pontes Duff. Gramians, energy functionals and balanced truncation for linear dynamical systems with quadratic outputs. *IEEE Trans. Autom. Control*, 67(2):886–893, 2022.
- [5] I. V. Gosea and A. C. Antoulas. A two-sided iterative framework for model reduction of linear systems with quadratic output. In *58th IEEE Conference on Decision and Control (CDC), December 11–13, Nice, France*, pages 7812–7817, 2019.
- [6] I. V. Gosea and S. Gugercin. Data-driven modeling of linear dynamical systems with quadratic output in the AAA framework. *J. Sci. Comput.*, 91(1):1–28, 2022.
- [7] S. Gugercin, A. C. Antoulas, and C. Beattie.  $\mathcal{H}_2$  model reduction for large-scale linear dynamical systems. *SIAM J. Matrix Anal. Appl.*, 30(2):609–638, 2008.
- [8] L. Meier and D. Luenberger. Approximation of linear constant systems. *IEEE Trans. Autom. Control*, 12(5):585–588, 1967.
- [9] S. Reiter, I. V. Gosea, I. Pontes Duff, and S. Gugercin.  $\mathcal{H}_2$ -optimal model reduction of linear quadratic-output systems by multivariate rational interpolation. e-print 2505.03057, arXiv, 2025. math.NA, eess.SY, math.DS, math.OC.
- [10] S. Reiter, I. Pontes Duff, I. V. Gosea, and S. Gugercin.  $\mathcal{H}_2$  optimal model reduction of linear systems with multiple quadratic outputs. e-print 2405.05951, arXiv, 2024. math.NA, eess.SY, math.DS, math.OC.
- [11] S. Reiter and S. W. R. Werner. Interpolatory model reduction of dynamical systems with root mean squared error. *IFAC-PapersOnLine*, 59(1):385–390, 2025. 11th Vienna International Conference on Mathematical Modelling MATHMOD 2025.
- [12] P. Van Dooren, K. A. Gallivan, and P-A Absil.  $H_2$ -optimal model reduction of MIMO systems. *Appl. Math. Lett.*, 21(12):1267–1273, 2008.
- [13] D. A. Wilson. Optimum solution of model-reduction problem. In *Proceedings of the Institution of Electrical Engineers*, volume 117, pages 1161–1165. IET, 1970.

# Towards Efficient Algorithms for Approximately Solving (Overdetermined) Systems of Polynomial Equations

*N. Govindarajan, R. Widdershoven, L. De Lathauwer*

## Abstract

We revisit the age-old problem of solving a system of multivariate polynomial equations. This problem can be viewed as a simultaneous generalization of solving linear systems and finding roots of univariate polynomials. It is well-known that the aforementioned special cases have widespread applications in science and engineering. It should come as no surprise that the same is true about the more general version of this problem. This is particularly the case in the noisy overdetermined setting, which extends the familiar engineering notion of solving a system of linear equations in a “least-squares” fashion to the polynomial case.

Classically, solving a system of polynomial equations belongs to the field of computational algebraic geometry. The literature advocates two major approaches to the problem. The first approach involves homotopy continuation, where the roots of the desired system are found by continuous deformation of a starting system for which the roots are already known. The second approach, which is more in line with our work, reduces the problem to an eigenvalue problem. The algebraic approach has its origins in resultant theory, tracing back to original contributions by B’ezout, Sylvester, Cayley, and Macaulay. The main idea behind this line of attack is to unveil the structure of the quotient algebra of the ideal generated by the polynomial system. Solutions of the system are subsequently extracted from the eigen-structure of the generated multiplication tables [1].

Up to this point, the literature has primarily focused on the noiseless square case. Herein, it is assumed that the coefficients of the polynomials are known with full precision and the number of equations equals the number of unknowns. On the contrary, a critical component of engineering applications is the estimation of system parameters from an overcomplete set of equations corrupted by noise. In the case of linear systems, numerical linear algebra already provides effective methods to deal with such problems. Analogous methods to treat the more general polynomial case are, however, far fewer, and relatively underdeveloped.

Our work hopes to fill this gap by taking a fresh perspective on the problem. The cornerstone of our proposed framework is the (tensor-based) Macaulay method [4]. This method rests on the idea that a basis for the quotient algebra can be formed by computing a numerical null space of a resultant map. The classical Macaulay matrix, which generalizes the Sylvester resultant matrix of two univariate polynomials to the multivariate case, is a canonical example of such a map. The fact that the null space is obtained through numerical means is an essential ingredient in enabling the solvability of polynomial systems in an approximate sense [5].

This core key feature of the methodology is best explained through an example. Suppose that one wants to solve the overdetermined linear system

$$\left[ \begin{array}{c|cc} -3 & -1 & -2 \\ -2 & -1 & 1 \\ 1 & 7 & 1 \end{array} \right] \left[ \begin{array}{c} 1 \\ x \\ y \end{array} \right] = \left[ \begin{array}{c} 0 \\ 0 \\ 0 \end{array} \right]$$

in a total least-squares sense. As it is well-known, the solution is trivially found by retrieving the smallest right-singular vector of the matrix on the left-hand side of the expression. Since the corresponding singular value is strictly positive, the right-singular vector (after normalization)

will only yield an approximate solution of the system. Interestingly, a similar strategy may be employed to find approximate solutions for polynomial systems. For example, the overdetermined polynomial system

$$\begin{array}{c} p_1(x,y) \\ p_2(x,y) \\ p_3(x,y) \end{array} \left[ \begin{array}{c|cc|ccc} -3 & -1 & -2 & 4 & 6 & 7 \\ -2 & -1 & 1 & 3 & -7 & 5 \\ 1 & 7 & 1 & -8 & 3 & 1 \end{array} \right] \begin{bmatrix} \frac{1}{x} \\ \frac{x}{x^2} \\ \frac{y}{x^2} \\ \frac{xy}{y^2} \\ \frac{y^2}{y^3} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (1)$$

has an exact solution at  $(x, y) = (1, 0)$ , which is also the only solution of the system. This single solution disappears under small perturbations of the coefficients, but one can still view  $(x, y) = (1, 0)$  as an approximate solution of the noise-perturbed system. After all, the Vandermonde vector evaluated at this point, i.e.,  $[1 \ 1 \ 0 \ 1 \ 0 \ 0]^\top$ , is still an approximate null vector for the noise perturbed matrix. It can therefore be subsequently used to derive approximate solutions.

Unlike the linear system, there are several complications that one has to treat in the polynomial case. Firstly, the existence of other artificial null vectors in (1) make the retrieval of the Vandermonde solution vector impossible. This first complication is resolved by adding additional equations to the system:

$$\begin{array}{c} p_1(x,y) \\ p_2(x,y) \\ p_3(x,y) \\ xp_1(x,y) \\ xp_2(x,y) \\ xp_3(x,y) \\ yp_1(x,y) \\ yp_2(x,y) \\ yp_3(x,y) \end{array} \left[ \begin{array}{c|cc|ccc|cccc} -3 & -1 & -2 & 4 & 6 & 7 & 0 & 0 & 0 & 0 \\ -2 & -1 & 1 & 3 & -7 & 5 & 0 & 0 & 0 & 0 \\ 1 & 7 & 1 & -8 & 3 & 1 & 0 & 0 & 0 & 0 \\ 0 & -3 & 0 & -1 & -2 & 0 & 4 & 6 & 7 & 0 \\ 0 & -2 & 0 & -1 & 1 & 0 & 3 & -7 & 5 & 0 \\ 0 & 1 & 0 & 7 & 1 & 0 & -8 & 3 & 1 & 0 \\ 0 & 0 & -3 & 0 & -1 & -2 & 0 & 4 & 6 & 7 \\ 0 & 0 & -2 & 0 & -1 & 1 & 0 & 3 & -7 & 5 \\ 0 & 0 & 1 & 0 & 7 & 1 & 0 & -8 & 3 & 1 \end{array} \right] \begin{bmatrix} \frac{1}{x} \\ \frac{x}{x^2} \\ \frac{y}{x^2} \\ \frac{xy}{y^2} \\ \frac{y^2}{y^3} \\ \frac{x^3}{x^3} \\ \frac{x^2y}{y^2} \\ \frac{xy^2}{y^3} \\ \frac{y^3}{y^3} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} .$$

The above matrix presents an example of a Macaulay matrix at a certain degree. This Macaulay matrix contains the Vandermonde Vector

$$[1 \ | \ 1 \ 0 \ | \ 1 \ 0 \ 0 \ | \ 1 \ 0 \ 0 \ 0]^\top$$

as the only null vector up to scaling ambiguity. Secondly, a polynomial system can, in general, admit multiple solutions. Consequently, the numerically obtained (approximate) null basis will not be immediately in Vandermonde form. This second complication is resolved by performing an additional unmixing operation to retrieve the Vandermonde basis. This Vandermonde basis reconstruction can be viewed as a multi-dimensional harmonic retrieval problem [3], and can be effectively solved by computing a canonical polyadic decomposition of a tensor formed from the obtained null space basis [4].

In general, solutions for a polynomial system can be determined in two computational steps: (i) compute the null space of the Macaulay matrix, and (ii) compute a polyadic decomposition of a tensor to retrieve the solutions. In the case of overdetermined systems that only possess a handful of solutions, the first step is a major computational bottleneck. Macaulay matrices grow very rapidly in size for even moderately-sized polynomial systems. This makes the null space computation

prohibitively expensive. To get an impression of the complexity, a polynomial in  $N$  variables of degree  $D$  contains  $\binom{N+D}{N}$  monomial terms. The Macaulay matrix at degree  $D_{\text{mac}} \geq D$  of  $S$  such polynomials is the matrix that is constructed from the polynomial coefficients of the system  $\{p_s\}_{s=1}^S$  such that its rows span the set of polynomials

$$\left\{ p := \sum_{s=1}^S h_s \cdot p_s : \deg(p) \leq D_{\text{mac}} \right\}. \quad (2)$$

Both the numbers of rows and columns of this matrix grow combinatorially in size with respect to  $D_{\text{mac}}$ ,  $D$ , and  $N$ .

Fortunately, the Macaulay matrix is highly structured. In the monomial basis, the Macaulay matrix possesses multilevel Toeplitz structures. It also has recursive properties, and in certain cases, the matrix can be highly sparse. Furthermore, the Chebyshev variant of the Macaulay matrix is multilevel Toeplitz-plus-Hankel. Subsequently, much of our research has been dedicated to developing efficient null-space computation algorithms that exploit the structures in the matrix. Our efforts in this area have led to some interesting work; some of which is still ongoing research:

1. In [2] we considered exploiting the low-displacement rank structure of the Macaulay matrix to compute the null space through a rank-revealing LU factorization using the Gohberg-Kailath-Olshevsky (GKO) algorithm. Although this approach reduced the memory and time complexity of the null-space computation significantly, the savings do not scale well for polynomial systems in many variables.
2. In recent (almost completed) work [6], we introduced an alternative approach that exploits the shift structure of the Macaulay matrix in a different way. This method scales more gracefully with the number of variables. The method relies on computing the null spaces of nested subblocks of increasing size by using the fact that the same subblocks are repeated throughout the Macaulay matrix and that the null space of two stacked (block-)rows equals the intersection of their individual null spaces.
3. Currently, we are also investigating an approach to compute the null space with a Krylov-based iterative technique. This option seems to be particularly attractive for overdetermined systems that only have a few (approximate) solutions. This is because, in this case, the Macaulay matrix will have a relatively small (numerical) null space. In the monomial basis, the Macaulay matrix has fast matrix-vector product with the help of fast Fourier transforms. Interestingly, one can also obtain a fast matrix-vector product for the Macaulay matrix in the Chebyshev basis using fast cosine transforms. A big challenge is to make the Krylov-based method converge in a reasonable number of iterations. Finding good preconditioners may be essential.

At the Householder Symposium, our goal is to share this research with our colleagues.

## References

- [1] David A Cox. Stickelberger and the eigenvalue theorem. In *Commutative Algebra*, pages 283–298. Springer, 2021.

- [2] Nithin Govindarajan, Raphaël Widdershoven, Shivkumar Chandrasekaran, and Lieven De Lathauwer. A fast algorithm for computing macaulay null spaces of bivariate polynomial systems. *SIAM J. Matrix Anal. Appl.*, 45(1):368–396, 2024.
- [3] M. Sørensen and L. De Lathauwer. Multidimensional harmonic retrieval via coupled canonical polyadic decomposition — Part I: Model and identifiability. *IEEE Transactions on Signal Processing*, 65(2):517–527, 1 2017.
- [4] Jeroen Vanderstukken and Lieven De Lathauwer. Systems of polynomial equations, higher-order tensor decompositions and multidimensional harmonic retrieval: A unifying framework. Part I: The canonical polyadic decomposition. *SIAM J. Matrix Anal. Appl.*, 42(2):883–912, 2021.
- [5] Raphaël Widdershoven, Nithin Govindarajan, and Lieven De Lathauwer. Overdetermined systems of polynomial equations: tensor-based solution and application. In *2023 31st European Signal Processing Conference (EUSIPCO)*, pages 650–654. IEEE, 2023.
- [6] Raphaël Widdershoven, Nithin Govindarajan, and Lieven De Lathauwer. Fast macaulay null space through the intersection of shifted null spaces. *Technical Report 25-56, ESAT-STADIUS, KU Leuven (Leuven, Belgium)*, 2025.

# When is the Resolvent Like a Rank One Matrix?

*Anne Greenbaum, Abbas Salemi, Faranges Kyanfar*

## Abstract

For a square matrix  $A$ , the *resolvent* at a point  $z \in \mathbf{C}$  is defined as  $(A - zI)^{-1}$ . It was observed in [2] that for certain matrices  $A$  with ill-conditioned eigenvalues the resolvent is close to the rank one matrix  $\sigma_1(z)u_1(z)v_1(z)^H$ , for a wide range of  $z$  values, where  $\sigma_1(z)$  is the largest singular value of  $(A - zI)^{-1}$  and  $u_1(z)$  and  $v_1(z)$  are the corresponding left and right singular vectors. Moreover, for a slightly smaller range of  $z$  values,  $u_1(z)$  and  $v_1(z)$  are almost orthogonal to each other. Here we provide a partial explanation for this phenomenon.

The distance in 2-norm from  $(A - zI)^{-1}$  to the nearest rank one matrix,  $\sigma_1(z)u_1(z)v_1(z)^H$ , is  $\sigma_2(z)$ , the second largest singular value of  $(A - zI)^{-1}$ , and one might define the relative distance as  $\sigma_2(z)/\|(A - zI)^{-1}\|_2 = \sigma_2(z)/\sigma_1(z)$ . Given  $\epsilon > 0$ , we are interested in

$$\{z \in \mathbf{C} : \sigma_2(z)/\sigma_1(z) < \epsilon\}. \quad (1)$$

Recall that the  $\epsilon$ -*pseudospectrum* of  $A$  can be defined as [3]:

$$\{z \in \mathbf{C} : 1/\sigma_1(z) < \epsilon\}. \quad (2)$$

If it turns out that  $\sigma_2(z) \sim 1$  throughout the  $\epsilon$ -pseudospectrum, then these two sets may look very similar. Indeed, the plots in [2] look much like pseudospectra.

To study this phenomenon, we will work with the matrix  $A - zI$ , whose singular values are the inverses of those of  $(A - zI)^{-1}$  and whose right and left singular vectors are the left and right singular vectors of  $(A - zI)^{-1}$ . If  $s_n(z)$  and  $s_{n-1}(z)$  denote the smallest and second smallest singular values of  $A - zI$ , then we are interested in the ratio  $s_n(z)/s_{n-1}(z)$ .

The following theorem and corollary are proved in a paper currently in progress [1]:

**Theorem.** Let  $\lambda$  be a simple eigenvalue of  $A$  and let  $A_0 := A - \lambda I = USV^H$  be a singular value decomposition of  $A_0$ , where  $U := [u_1, \dots, u_n]$ ,  $V := [v_1, \dots, v_n]$ ,  $S := \text{diag}(s_1, \dots, s_{n-1}, 0)$ ,  $s_1 \geq \dots \geq s_{n-1} > 0$ . Let  $A_0^\dagger$  denote the Moore-Penrose pseudoinverse of  $A_0$ :

$$A_0^\dagger := V_{n-1} S_{n-1}^{-1} U_{n-1}^H, \quad (3)$$

where  $U_{n-1} := [u_1, \dots, u_{n-1}]$ ,  $V_{n-1} := [v_1, \dots, v_{n-1}]$ , and  $S_{n-1} := \text{diag}(s_1, \dots, s_{n-1})$ . For each  $k = 1, 2, \dots$ , the smallest singular value of  $A_0 - zI$  is less than or equal to

$$\sum_{j=1}^k |z|^j |u_n^H (A_0^\dagger)^{j-1} v_n| + |z|^{k+1} / s_{n-1}^k. \quad (4)$$

Taking  $k = 1$  in the theorem, we obtain the bound

$$s_n(A_0 - zI) \leq |u_n^H v_n| |z| + |z|^2 / s_{n-1}.$$

If  $\lambda$  is *ill-conditioned*, it means that the inner product of the left and right unit eigenvectors of  $A$  corresponding to eigenvalue  $\lambda$  is tiny, but these eigenvectors are the same as the left and right

singular vectors  $u_n$  and  $v_n$  corresponding to the zero singular value of  $A_0$ . In this case, if also  $s_{n-1} \sim 1$ , then  $s_n(A_0 - zI)$  grows more like  $|z|^2$  than like  $|z|$  for  $|u_n^H v_n| \ll |z| \ll 1$ . If  $u_n$  is also nearly orthogonal to  $A_0^\dagger v_n$ , then taking  $k = 2$  in the theorem suggests that the growth rate of  $s_n(A_0 - zI)$  with  $|z|$  may be more like  $|z|^3$ , and the more powers  $j$  for which  $|u_n^H (A_0^\dagger)^j v_n|$  is small, the higher the power of  $|z|$  describing the growth of  $s_n(A_0 - zI)$ , for  $|z| \ll 1$ . If the absolute value of  $z$  times each eigenvalue of  $A_0^\dagger$  is less than one, then the first sum in (4) will converge to a finite value as  $k \rightarrow \infty$ , and for  $|z| < s_{n-1}$ , the second term in (4) will go to 0 as  $k \rightarrow \infty$ . In this case, the smallest bound may be obtained by taking  $k = \infty$ .

Although we are not yet sure how to interpret the conditions that  $|u_n^H (A_0^\dagger)^j v_n|$  be small, these conditions seem to be satisfied by many test problems with ill-conditioned eigenvalues, such as those available through the 'gallery' command in MATLAB and many in [3].

**Corollary.** With the notation and assumptions of the previous theorem, let  $\epsilon \in (0, 1)$  be given. The region where the ratio of the second largest to the largest singular value of the resolvent  $(A - zI)^{-1}$  is less than  $\epsilon$  contains the set of points  $z \in \mathbf{C}$  such that  $|z - \lambda| < s_{n-1}$  and

$$\min_{k=1,2,\dots} \left[ \sum_{j=1}^k |z - \lambda|^j |u_n^H (A_0^\dagger)^{j-1} v_n| + |z - \lambda|^{k+1} / s_{n-1}^k \right] / (s_{n-1} - |z - \lambda|) < \epsilon. \quad (5)$$

The  $\epsilon$ -pseudospectrum of  $A$  contains the set of points  $z \in \mathbf{C}$  such that

$$\min_{k=1,2,\dots} \left[ \sum_{j=1}^k |z - \lambda|^j |u_n^H (A_0^\dagger)^{j-1} v_n| + |z - \lambda|^{k+1} / s_{n-1}^k \right] < \epsilon. \quad (6)$$

This corollary defines disks about each eigenvalue that are known to lie within the regions defined in (1) and (2). In our numerical tests, they are not far from the largest disks about the eigenvalues that are contained in these regions.

We will report on these results, as well as some results obtained by differentiating the singular values and vectors of  $A - zI$ .

## References

- [1] A. GREENBAUM, F. KYANFAR, AND A. SALEMI, *When is the Resolvent Like a Rank One Matrix?*, to appear.
- [2] A. GREENBAUM AND N. WELLEN, *Comparison of K-Spectral Set Bounds on Norms of Functions of a Matrix or Operator*, Lin. Alg. Appl. 694, pp. 52-77, 2024.
- [3] L. N. TREFETHEN AND M. EMBREE, *Spectra and Pseudospectra: The Behavior of Nonnormal Matrices and Operators*, Princeton University Press, 2005.

# Randomization techniques for solving eigenvalue problems

*Laura Grigori, Jean-Guillaume de Damas*

## Abstract

In this talk we will discuss randomization techniques for computing a few eigenpairs of a large, sparse, non-symmetric matrix  $A$ . We consider Krylov subspace methods and the Rayleigh-Ritz process that relies on projection onto the Krylov subspace  $\mathcal{K}_k(A, v_1) = \text{span}\{v_1, Av_1, A^2v_1, \dots, A^{k-1}v_1\}$  formed after  $k$  iterations for a given starting vector  $v_1$ .

Randomization for Krylov subspace methods was introduced in the recent years, in particular for solving linear systems of equations [1, 7, 8]. Randomized Arnoldi introduced in [1] relies on a randomized orthogonalization process that produces a well conditioned basis of the Krylov subspace and thus can be efficiently used in a randomized version of GMRES. It is shown in [1] that randomized GMRES is quasi-optimal since it relies on solving a sketched least squares problem to minimize the residual and obtain a new solution. A different approach consists in using sketching independently of the construction of the Krylov basis, as in sketched GMRES [7], or sketch and select [5]. Restarting in the context of linear systems is discussed in [2, 6].

We first discuss the usage of randomization for orthogonalizing a set of vectors that are of very large dimension. This operation that occurs in many computations is very often the bottleneck in terms of communication. Indeed, when the vectors to be orthogonalized are distributed over many processors and are obtained one by one as in Krylov subspace methods, the orthogonalization of each vector with respect to the previous ones requires synchronizing all processors, and this hinders drastically the scalability of a parallel algorithm. Two different methods are in general used. The first one is classical Gram-Schmidt (CGS), which requires one synchronization to orthogonalize a new vector against the previous ones and thus can be considered to be efficient, but suffers from numerical stability issues since it depends quadratically on the condition number of the vectors. The second one is modified Gram-Schmidt (MGS) which has better numerical stability with linear dependency on the condition number, but is inefficient since it relies on vector-vector operations and requires  $j$  synchronizations for orthogonalizing a new vector against the previous  $j$  orthogonalized vectors. Householder QR is a highly accurate process for orthogonalizing a set of vectors, but is in general used for orthogonalizing a set of vectors that are given all at once.

Several different algorithms have been introduced in the literature, as randomized Gram-Schmidt [1] and randomized Householder QR [4]. The main idea is to use randomization to sketch the vectors that need to be orthogonalized, obtain a smaller problem for which a highly accurate orthogonalization algorithm can be used as Householder QR, and then use the orthogonalized sketch vectors to obtain a well conditioned basis for the vectors of very large dimension. This approach is suitable for the usage of mixed precision, since after the random projection, a smaller problem is obtained that can be solved in higher precision. This leads to obtaining a basis for the Krylov subspace whose sketch is orthogonal, but not the basis itself. We will discuss in particular a reorthogonalization process that allows to improve the stability of randomized Gram-Schmidt.

We then discuss the usage of such an orthogonalization process that produces a well conditioned basis within an eigenvalue solver, that leads to a randomized Rayleigh-Ritz process. We introduce the randomized Implicitly Restarted Arnoldi (randomized IRA) method, that relies on a sketched orthonormal basis and a restarting scheme that allows to seek a specific subset of eigenpairs of a non-symmetric matrix  $A$ . We provide a theoretical analysis that shows that some of the results

defining the convergence behavior of IRA hold for randomized IRA, up to a factor of  $1 + O(\epsilon)$  and with high probability. More details can be found in [3].

Finally, we will discuss one of the challenges that arises when using randomization for symmetric matrices, that is related to the fact that randomization destroys symmetry. Thus the Hessenberg matrix associated with the Arnoldi process is not symmetric as in the case of deterministic methods. We discuss implications and possible solutions to this problem.

## References

- [1] Oleg Balabanov and Laura Grigori. Randomized gram–schmidt process with application to gmres. *SIAM Journal on Scientific Computing*, 44(3):A1450–A1474, 2022.
- [2] Liam Burke, Stefan Güttel, and Kirk M. Soodhalter. GMRES with randomized sketching and deflated restarting, 2023.
- [3] Jean-Guillaume de Damas and Laura Grigori. Randomized implicitly restarted arnoldi method for the non-symmetric eigenvalue problem, 2024.
- [4] Laura Grigori and Edouard Timsit. Randomized householder qr, 2024.
- [5] Stefan Güttel and Igor Simunec. A sketch-and-select Arnoldi process, 2023.
- [6] Yongseok Jang, Laura Grigori, Emeric Martin, and Cédric Content. Randomized flexible GMRES with deflated restarting. *Numerical Algorithms*, March 2024.
- [7] Yuji Nakatsukasa and Joel A. Tropp. Fast and accurate randomized algorithms for linear systems and eigenvalue problems. *SIAM Journal on Matrix Analysis and Applications*, 45(2):1183–1214, 2024.
- [8] Edouard Timsit, Laura Grigori, and Oleg Balabanov. Randomized orthogonal projection methods for Krylov subspace solvers, 2023.

# Subspace accelerated contour integration methods for eigenvalue problems

Luka Grubišić

## Abstract

In this talk, we will present a class of adaptive approximation methods for computing the partial solution of eigenvalue problems. We will concentrate on algorithms which are matrix-free in the sense that they treat a matrix  $A$ , or its shifted inverse  $(z - A)^{-1}$ , as a mapping  $A : x \mapsto Ax$ , and  $(z - A)^{-1} : x \mapsto (z - A)^{-1}x$ , respectively. We present a Beyn-type eigensolver (see [1]) accelerated by the use of adaptive reduced-order model of the matrix resolvent. As prototype examples, we will consider both linear as well as nonlinear (in the spectral parameter) eigenvalue problems. In particular, we will study examples from thermoacoustics applications [14].

In the interest of clarity, let us first concentrate on the standard linear eigenvalue problem for a diagonalisable matrix  $A$ . When the resolvent is given as a mapping  $(z - A)^{-1} : x \mapsto (z - A)^{-1}x$ , one has to incorporate the inexactness (due to the approximation truncation) of the evaluation of this mapping into an analysis. This is a known and structurally challenging problem in the theory of Krylov-type solvers [10, 16]. An alternative approach is to transform the problem of approximating the eigenvalue cluster enclosed by the finite contour  $\Gamma$  into an eigenvector problem for the spectral projector  $P_\Gamma$

$$P_\Gamma = \frac{1}{2\pi i} \int_{\Gamma} (z - A)^{-1} dz \approx \Pi_\Gamma := \sum_{i=1}^N \omega_i (z_i - A)^{-1}.$$

One can then apply the standard subspace iteration to extract eigenvector information using the approximation

$$x_j \mapsto P_\Gamma x_j = \frac{1}{2\pi i} \int_{\Gamma} (z - A)^{-1} x_j dz \approx \sum_{i=1}^N \omega_i (z_i - A)^{-1} x_j, \quad j = 1, \dots, d$$

For this talk we choose not to discuss the implications of embarrassing parallelism (in terms of sampling the resolvent with respect to the spectral parameter  $z_i$  and vectors  $x_j$ ) on the evaluation of the action of  $P_\Gamma$ .

We will loosely call this approach interpolatory and nonintrusive. Namely, to produce a reliable eigenvalue/vector approximation method, one only needs a solver for the shifted system  $(z, x) \mapsto (z - A)^{-1}x$  as a black box, but with an error estimate and error control. The projection  $P_\Gamma$  is a dense, but low-rank matrix. The dimension of its range equals the joint algebraic multiplicity of the eigenvalues enclosed by the contour  $\Gamma$ , denoted by  $\#\Gamma$ . The problem of computing an orthonormal basis of the eigensubspace associated with the enclosed cluster of eigenvalues can now be reduced to the calculation of the SVD of a large implicitly defined matrix  $P_\Gamma$  of low rank. This orthonormal basis can then be used to construct a small auxiliary spectral problem from which eigenvector/eigenvalue information can be directly and robustly extracted (not the topic of this talk). Randomized SVD has distinguished itself as a method of choice for analyzing approximate low rank matrices. It has been studied in many settings, including its infinite-dimensional incarnation [13, 3, 2], which is suitable for the study of numerical methods applied to discretizations of partial differential operators in physics and engineering. Note that in our notation the randomized SVD algorithm for  $\Pi_\Gamma \approx P_\Gamma$  starts with the random draw of the interpolation directions  $x_j$ ,  $j = 1, \dots, d$  for  $d \geq \#\Gamma + 2$ . Here we assume that  $x_j$  have been drawn appropriately, [2].

The nonintrusive nature of contour integration methods is the reason for inclusion in SLEPc or even as **Extended Eigensolver Routines** in the Intel MKL library. This is the easiest way to incorporate any monolithic solver for the shifted system into an eigenvalue/eigenvector approximation routine. Large-scale matrices in NLA are typically discretizations of partial differential operators, and the use of contour integration approach allows more flexibility to seamlessly incorporate various discretisations of the shifted system (called in the engineering jargon the Helmholtz solvers). These include rectangular approximations of the resolvent such as those from [8] used in `chebop` object or the Discontinuous Petrov Gelerkin approach which also leads to rectangular approximations of the resolvent [11, 9, 7].

Based on the (infinite-dimensional) randomized SVD for Hilbert–Schmidt operators, an extension of Beyn’s contour integration method for operators in Hilbert spaces has been described in [5]. The key ingredient, encapsulated in the phrase *solve than discretize*, is adaptive error control for the Helmholtz solver. Pushing discretization by adaptivity to the later stage, the randomized part of the algorithm gives us means to explore the Hilbert space more broadly and generate an accelerating subspace with better candidates for eigenvector approximations.

The use of advances in the rational function approximation problem in the context of the solution of the spectral problem has been thoroughly analyzed, in a slightly different context, in [6]. To coarsely assess the performance of this method, consider a finite difference discretization of  $A = -\Delta - V$ ,  $V > 0$ , with Gaussian potential  $V$ ,  $\|V\|_\infty < \infty$ . Using the MATLAB toolbox `SpecSolve`<sup>1</sup> on a computer with 10 cores, it took 104 seconds to approximate the spectral density in the interval  $[-\|V\|_\infty, 0]$  with tolerance  $\varepsilon = 0.05$ . In comparison, MATLAB `eigs` on the same machine applied to a  $10^4 \times 10^4$  discretization computed all eigenvalues in the same interval within 0.5 seconds. Apart from the further use of obvious embarrassing parallelism in the sampling of the resolvent, a speedup can be achieved by exploiting the product structure in the construction of random vectors [4] (not this talk) or by speeding up the evaluation of the resolvent using subspace acceleration [14] (this talk).

As prototypes, we will consider a large class of (nonlinear) eigenvalue problems which are defined by the generalized resolvent

$$R(z) = (A_0 + f_1(z)A_1 + \cdots + f_s(z)A_s)^{-1}$$

with self-adjoint coefficients  $A_i$ ,  $i = 0, \dots, s$  and scalar functions  $f_i$ ,  $i = 1, \dots, s$ . We will present an analysis and improvements of the method described in [14] which uses subspace acceleration together with reduced-order interpolatory modeling of the nonlinear resolvent  $R$ . Our method will be cast within the context of scientific computing with particular emphasis on problems in thermoacoustics. We will discuss the comparison of the performance of the contour integration method with the performance of the method based on the direct rational interpolation of the resolvent and the application of the rational Arnoldi to its linearization [15, 12]. Finally, we will present a general analysis of the randomized SVD algorithm for operators of the form

$$\mathbf{r}(A_0 + V) + W .$$

Here  $\mathbf{r}$  is a rational function approximation of an indicator function,  $A_0$  is self-adjoint and positive definite, potential  $V$  is relatively compact with respect to  $A_0$ , and we use functions of  $A_0$  to construct a Gaussian kernel for random sampling. Finally,  $W$  (not necessarily self-adjoint) is a small bounded operator presenting the errors caused by adaptive discretization.

---

<sup>1</sup><https://github.com/SpecSolve/SpecSolve>

## References

- [1] Wolf-Jürgen Beyn (2012). An integral method for solving nonlinear eigenvalue problems. *Linear Algebra and its Applications*, 436(10), p.3839–3863.
- [2] N. Boullé and A. Townsend (2022), A generalization of the randomized singular value decomposition, in *International Conference on Learning Representations*.
- [3] N. Boullé and A. Townsend (2023), ‘Learning elliptic partial differential equations with randomized linear algebra’, *Found. Comput. Math.* **23**(2), 709–739.
- [4] Zvonimir Bujanović and Luka Grubišić and Daniel Kressner and Hei Yin Lam (2024), Subspace embedding with random Khatri-Rao products and its application to eigensolvers, *arXiv:2405.11962*.
- [5] M. Colbrook and A. Townsend (2023), Avoiding discretization issues for nonlinear eigenvalue problems, *arXiv:2305.01691*.
- [6] M. Colbrook, A. Horning and A. Townsend (2021), ‘Computing spectral measures of self-adjoint operators’, *SIAM Review* **63**(3), 489–524.
- [7] L. Demkowicz and J. Gopalakrishnan (2017), *Discontinuous Petrov–Galerkin (DPG) Method*, John Wiley and Sons, Ltd, pp. 1–15.
- [8] T. A. Driscoll and N. Hale (2016), ‘Rectangular spectral collocation’, *IMA Journal of Numerical Analysis* **36**(1), 108–132.
- [9] X. Feng and H. Wu (2009), ‘Discontinuous galerkin methods for the helmholtz equation with large wave number’, *SIAM Journal on Numerical Analysis* **47**(4), 2872–2896.
- [10] Freitag, M. and Spence, A. (2010). Shift-Invert Arnoldi’s Method with Preconditioned Iterative Solves. *SIAM Journal on Matrix Analysis and Applications*, 31(3), 942-969.
- [11] J. Gopalakrishnan, L. Grubišić, J. Ovall and B. Parker (2019), ‘Analysis of FEAST spectral approximations using the DPG discretization’, *Comput. Methods Appl. Math.* **19**(2), 251–266.
- [12] S. Güttel, R. V. Beeumen, K. Meerbergen and W. Michiels (2013), NLEIGS: a class of robust fully rational Krylov methods for nonlinear eigenvalue problems, TW Reports, TW633, Department of Computer Science, KU Leuven.
- [13] P.-G. Martinsson and J. A. Tropp (2020), ‘Randomized numerical linear algebra: foundations and algorithms’, *Acta Numer.* **29**, 403–572.
- [14] G. A. Mensah, A. Orchini, P. E. Buschmann and L. Grubišić (2022), ‘A subspace-accelerated method for solving nonlinear thermoacoustic eigenvalue problems’, *Journal of Sound and Vibration* **520**, 116553.
- [15] M. Merk, P. E. Buschmann, J. P. Moeck and W. Polifke (2022), ‘The Nonlinear Thermoacoustic Eigenvalue Problem and Its Rational Approximations: Assessment of Solution Strategies’, *Journal of Engineering for Gas Turbines and Power* **145**(2), 021028.
- [16] V. Simoncini and D. B. Szyld (2003), ‘Theory of inexact krylov subspace methods and applications to scientific computing’, *SIAM Journal on Scientific Computing* **25**(2), 454–477.

# Separable Low-rank Barycentric Forms in the p-AAA Algorithm

*Linus Balicki, Serkan Gugercin*

## Abstract

Rational approximation is a powerful tool for capturing the behavior of functions which have singularities on or near a domain of interest. This circumstance makes rational functions ubiquitous in fields such as signal processing, model reduction, and partial differential equations. In recent years the adaptive Antoulas-Anderson (AAA) algorithm [2] has established itself as a successful method for computing rational approximations from a set of sampled data. Our recent work [3] introduced the p-AAA algorithm, extending the original AAA framework to multivariate functions. In order to allow for a clear presentation, we first discuss the two variable case, where the goal is to approximate a function  $\mathbf{f} : \mathbb{C}^2 \rightarrow \mathbb{C}$ . In this case, p-AAA is given a set of samples

$$\mathbf{F} = \{\mathbf{f}(x_1, x_2) \mid x_1 \in \mathbf{X}_1, x_2 \in \mathbf{X}_2\} \subset \mathbb{C}$$

with the corresponding sampling points

$$\mathbf{X}_1 = \{X_{11}, \dots, X_{1N_1}\} \subset \mathbb{C} \quad \text{and} \quad \mathbf{X}_2 = \{X_{21}, \dots, X_{2N_2}\} \subset \mathbb{C},$$

where  $X_{ij}$  denotes the  $j$ -th sampling point of the  $i$ -th variable. Then the goal is to approximate this data via a rational function represented as a multivariate barycentric form, i.e.,

$$\mathbf{r}(x_1, x_2) = \frac{\mathbf{n}(x_1, x_2)}{\mathbf{d}(x_1, x_2)} = \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \frac{\alpha_{i_1 i_2} \mathbf{f}(\xi_{1i_1}, \xi_{2i_2})}{(x_1 - \xi_{1i_1})(x_2 - \xi_{2i_2})} \left/ \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \frac{\alpha_{i_1 i_2}}{(x_1 - \xi_{1i_1})(x_2 - \xi_{2i_2})} \right. \quad (1)$$

where  $\alpha_{i_1 i_2} \neq 0$ . We note that  $\mathbf{r}$  is interpolatory in the sense that

$$\mathbf{r}(x_1, x_2) = \mathbf{f}(x_1, x_2) \quad \text{for } x_1 \in \boldsymbol{\xi}_1 \quad \text{and} \quad x_2 \in \boldsymbol{\xi}_2,$$

where

$$\boldsymbol{\xi}_1 = \{\xi_{11}, \dots, \xi_{1n_1}\} \subset \mathbf{X}_1 \quad \text{and} \quad \boldsymbol{\xi}_2 = \{\xi_{21}, \dots, \xi_{2n_2}\} \subset \mathbf{X}_2$$

are the respective sets of interpolation nodes. Similar to before,  $\xi_{ij}$  denotes the  $j$ -th interpolation node of the  $i$ -th variable. The p-AAA algorithm follows an iterative procedure to choose the interpolation nodes as well as the matrix of barycentric coefficients

$$\alpha \in \mathbb{C}^{n_1 \times n_2}, \quad \text{i.e., } \alpha(i_1, i_2) = \alpha_{i_1 i_2}.$$

Each iteration consists of first performing a greedy selection where we determine

$$(x_1^*, x_2^*) = \arg \max_{(x_1, x_2) \in \mathbf{X}_1 \times \mathbf{X}_2} |\mathbf{r}(x_1, x_2) - \mathbf{f}(x_1, x_2)| \quad (2)$$

and update the interpolation sets via

$$\boldsymbol{\xi}_1 \leftarrow \boldsymbol{\xi}_1 \cup \{x_1^*\} \quad \text{and} \quad \boldsymbol{\xi} \leftarrow \boldsymbol{\xi} \cup \{x_2^*\}.$$

As a second step, the barycentric coefficients are computed by solving a linear least-squares (LS) problem of the form

$$\min_{\|\alpha\|_F=1} \|\mathbb{L}_2 \text{vec}(\alpha)\|_2^2, \quad (3)$$

where  $\mathbb{L}_2 \in \mathbb{C}^{N_1 N_2 \times n_1 n_2}$  is the 2D Loewner matrix. The LS problem above arises as the minimization of the approximation error

$$\begin{aligned} \sum_{i_1=1}^{N_1} \sum_{i_2=1}^{N_2} |\mathbf{f}(X_{1i_1}, X_{2i_2}) - \mathbf{r}(X_{1i_1}, X_{2i_2})|^2 = \\ \sum_{i_1=1}^{N_1} \sum_{i_2=1}^{N_2} \left| \frac{1}{\mathbf{d}(X_{1i_1}, X_{2i_2})} (\mathbf{d}(X_{1i_1}, X_{2i_2}) \mathbf{f}(X_{1i_1}, X_{2i_2}) - \mathbf{n}(X_{1i_1}, X_{2i_2})) \right|^2, \end{aligned}$$

which is linearized by dropping the  $1/\mathbf{d}(X_{1i_1}, X_{2i_2})$  terms. In other words, the linearized LS problem minimizes the expression

$$\|\mathbb{L}_2 \text{vec}(\alpha)\|_2^2 = \sum_{i_1=1}^{N_1} \sum_{i_2=1}^{N_2} |(\mathbf{d}(X_{1i_1}, X_{2i_2}) \mathbf{f}(X_{1i_1}, X_{2i_2}) - \mathbf{n}(X_{1i_1}, X_{2i_2}))|^2.$$

This procedure is repeated until the approximation error indicated in (2) drops below a desired error tolerance. Solving the LS problem in (3) is the dominant cost of p-AAA and is done via a singular value decomposition of  $\mathbb{L}_2$ . More precisely, (3) has a closed form solution which is given in terms of the right singular vector of  $\mathbb{L}_2$  which corresponds to the smallest singular value.

While we only outlined the two variable case so far, the algorithm can easily be formulated as an approximation procedure for functions  $\mathbf{f} : \mathbb{C}^d \rightarrow \mathbb{C}$  that depend on  $d > 2$  variables. The key adjustments that need to be taken into account are that the multivariate approximant  $\mathbf{r}(x_1, x_2, \dots, x_d)$  will depend on a tensor

$$\alpha \in \mathbb{C}^{n_1 \times \dots \times n_d} \quad (4)$$

of barycentric coefficients (rather than a matrix) and the p-AAA LS problem will be based on the higher-order Loewner matrix  $\mathbb{L}_d \in \mathbb{C}^{N_1 \cdots N_d \times n_1 \cdots n_d}$ . In this case solving this dense LS problem via SVD requires  $\mathcal{O}(N_1 \cdots N_d n_1^2 \cdots n_d^2)$  operations and thus computing  $\alpha$  or even forming  $\mathbb{L}_d$  may become an infeasible task. We note that this growth in complexity is a common issue in multivariate approximation algorithms and is typically referred to as the “curse of dimensionality”. While there exist approaches to overcome these obstacles in multivariate function approximation (e.g., sparse grids, radial basis schemes), we focus here on a method that leverages a separable representation of the denominator  $\mathbf{d}$  of the rational approximant  $\mathbf{r}$ . As we will point out in the following, such a representation is directly connected to low-rank representations of higher-order tensors and allows for partially overcoming the curse of dimensionality associated with the p-AAA LS problem.

In order to introduce our proposed approach, we consider a canonical (CP) [1] decomposition of the tensor  $\alpha$  in (4) which we write in its vectorized form as

$$\text{vec}(\alpha) = \sum_{\ell=1}^r \beta_{1\ell} \otimes \dots \otimes \beta_{d\ell} \in \mathbb{C}^{n_1 \cdots n_d},$$

where  $\beta_{i\ell} \in \mathbb{C}^{n_i}$  for  $\ell = 1, \dots, r$ . The matrices  $\beta_1 = [\beta_{11}, \dots, \beta_{1r}] \in \mathbb{C}^{n_1 \times r}$ , ...,  $\beta_d = [\beta_{d1}, \dots, \beta_{dr}] \in \mathbb{C}^{n_d \times r}$  are called CP factors and the smallest  $r$  for which such a decomposition exists defines the tensor rank of  $\alpha$ . The CP decomposition is particularly useful if it is able to represent (or approximate)  $\alpha$  with a small number of terms  $r \ll n_1, \dots, n_d$ . In this case the storage requirement for the CP factors is merely  $\mathcal{O}(r(n_1 + \dots + n_d))$  rather than  $\mathcal{O}(n_1 \cdots n_d)$  for the full tensor. We

propose to take advantage of this reduction in the degrees of freedom in the representation of  $\alpha$  within the p-AAA algorithm.

To make this idea more clear, we revisit the two variable case. There the CP decomposition is analogous to a rank- $r$  outer product representation given by

$$\alpha = \beta_1 \beta_2^\top, \quad (5)$$

Plugging this representation for  $\alpha$  into the denominator in (1) gives the separable representation

$$\mathbf{d}(x_1, x_2) = \sum_{\ell=1}^r \left( \sum_{i_1=1}^{n_1} \frac{(\beta_{1\ell})_{i_1}}{x_1 - \xi_{1i_1}} \right) \left( \sum_{i_2=1}^{n_2} \frac{(\beta_{2\ell})_{i_2}}{x_2 - \xi_{2i_2}} \right),$$

where  $(\beta_{1\ell})_{i_1} \in \mathbb{C}$  is the  $i_1$ -th entry of the vector  $\beta_{1\ell} \in \mathbb{C}^{n_1}$  and  $(\beta_{2\ell})_{i_2} \in \mathbb{C}$  is the  $i_2$ -th entry of the vector  $\beta_{2\ell} \in \mathbb{C}^{n_2}$ . Our main idea for incorporating such separable representations and the associated low-rank decomposition for the barycentric coefficients into the p-AAA algorithm is to add the decomposition introduced in (5) as a constraint to the LS problem in (3). In this case we obtain the LS problem

$$\min_{\beta_1, \beta_2} \left\| \mathbb{L}_2 \sum_{\ell=1}^r \beta_{1\ell} \otimes \beta_{2\ell} \right\|_2^2 \quad \text{s.t.} \quad \left\| \sum_{\ell=1}^r \beta_{1\ell} \otimes \beta_{2\ell} \right\|_2 = 1. \quad (6)$$

We introduce the matrices

$$K_{\beta_1} := [\beta_{11} \otimes I_{n_2}, \dots, \beta_{1r} \otimes I_{n_2}] \in \mathbb{C}^{n_1 n_2 \times n_2 r} \quad \text{and} \quad K_{\beta_2} = [I_{n_1} \otimes \beta_{21}, \dots, I_{n_1} \otimes \beta_{2r}] \in \mathbb{C}^{n_1 n_2 \times n_1 r},$$

as well as the contracted Loewner matrices

$$\mathbb{L}_{\beta_1} := \mathbb{L}_2 K_{\beta_1} \in \mathbb{C}^{N_1 N_2 \times n_2 r} \quad \text{and} \quad \mathbb{L}_{\beta_2} := \mathbb{L}_2 K_{\beta_2} \in \mathbb{C}^{N_1 N_2 \times n_1 r},$$

which allow for writing the constrained LS problem in two distinct ways

$$\min_{\beta_1, \beta_2} \|\mathbb{L}_{\beta_2} \text{vec}(\beta_1)\|_2^2 \quad \text{s.t.} \quad \|K_{\beta_2} \text{vec}(\beta_1)\|_2 = 1, \quad (7)$$

$$\min_{\beta_1, \beta_2} \|\mathbb{L}_{\beta_1} \text{vec}(\beta_2)\|_2^2 \quad \text{s.t.} \quad \|K_{\beta_1} \text{vec}(\beta_2)\|_2 = 1. \quad (8)$$

We note that if one of the factors  $\beta_1$  or  $\beta_2$  is fixed, the other one can be obtained by solving an equality constrained LS problem based on the formulation above. In this case, these linear LS problems have a closed form solution in terms of the generalized SVD [4] of the matrix tuple  $(\mathbb{L}_{\beta_2}, K_{\beta_2})$  or  $(\mathbb{L}_{\beta_1}, K_{\beta_1})$ , respectively. Hence, the constrained LS problem in (6) has a separable structure which can be tackled via an alternating least-squares (ALS) procedure. In this procedure, we start with an initial guess for  $\beta_1$  and  $\beta_2$ , then repeatedly solve the problem in (7) while keeping  $\beta_2$  fixed, and the problem in (8) while keeping  $\beta_1$  fixed. This ALS approach requires  $\mathcal{O}(N_1 N_2 r^2(n_1^2 + n_2^2))$  operations which corresponds to the cost of computing the generalized SVDs. While this change in complexity may only have a small impact in the two variable case, it can be critical when moving to  $d > 2$  variables where ALS requires  $\mathcal{O}(N_1 \cdots N_d r^2(n_1^2 + \cdots + n_d^2))$  operations. Additionally, we note that the contracted Loewner matrices can be assembled efficiently by exploiting the Kronecker structure present in  $\mathbb{L}_d$ . These facts make it appealing to combine p-AAA with a separable representation for the denominator  $\mathbf{d}$ .

We conclude this abstract by considering a simple example where our proposed low-rank version of the p-AAA algorithm yields a high-fidelity rational approximant, while the standard p-AAA algorithm runs out of memory on our machine during the construction of the Loewner matrix  $\mathbb{L}_d$ . Specifically, we consider approximating the function

$$\mathbf{f}(x_1, x_2, x_3, x_4, x_5) = \frac{x_1 + x_2 + x_3 + x_4 + x_5}{10 + \sin(x_1) + \sin(x_2) + \sin(x_3) + \sin(x_4) + \sin(x_5)}$$

on the domain  $[-3, 3]^5$ . For each variable we choose the same sampling points corresponding to 20 linearly spaced values in the interval  $[-3, 3]$ . We run the proposed low-rank version of p-AAA and enforce a rank  $r = 1$  constraint on the coefficient tensor  $\alpha$ . After 6 iterations the relative maximum approximation error over the sampled data is approximately  $8.514 \times 10^{-3}$  and p-AAA chose 6 interpolation nodes for each variable. The standard p-AAA algorithm does not make it past the third iteration on our machine, due to running out of memory. Note that in this example the memory requirement for  $\mathbb{L}_5$  is around 24.4 GB in double precision once 4 interpolation nodes are chosen for each variable. In order to evaluate the quality of the computed approximation for unsampled data, we validate  $\mathbf{r}$  on a set of samples obtained by sampling 50 linearly spaced points in  $[-3, 3]$  for each variable. The maximum relative error on this validation set is approximately  $8.704 \times 10^{-3}$ , which closely matches the maximum error on the training data set.

## References

- [1] T. G. Kolda and B. W. Bader. Tensor Decompositions and Applications. *SIAM Review*, 51(3):455–500, Aug. 2009. Publisher: Society for Industrial and Applied Mathematics.
- [2] Y. Nakatsukasa, O. Sète, and L. N. Trefethen. The AAA Algorithm for Rational Approximation. *SIAM Journal on Scientific Computing*, 40(3):A1494–A1522, 2018.
- [3] A. C. Rodriguez, L. Balicki, and S. Gugercin. The p-AAA Algorithm for Data-Driven Modeling of Parametric Dynamical Systems. *SIAM Journal on Scientific Computing*, 45(3):A1332–A1358, 2023.
- [4] C. F. Van Loan. Generalizing the Singular Value Decomposition. *SIAM Journal on Numerical Analysis*, 13(1):76–83, Mar. 1976. Publisher: Society for Industrial and Applied Mathematics.

# Robust Hierarchical Matrix Approximation from Sketches

*Diana Halikias, Tyler Chen, Feyza D. Keles, Cameron Musco, Christopher Musco, David Persson*

### Abstract

Sketching is a tool for dimensionality reduction that lies at the heart of many fast and highly accurate “matrix-free” algorithms for fundamental tasks such as solving linear systems and eigenvalue problems, low-rank approximation, and trace estimation. Broadly, to solve a problem in the sketching model, one only queries a matrix of interest  $A \in \mathbb{R}^{n \times n}$  with relatively few matrix-vector products  $x \mapsto Ax$  and  $y \mapsto A^\top y$ , as opposed to accessing and working with  $A$ ’s individual entries. The sketching model is increasingly prevalent in numerical linear algebra for three reasons. First,  $A$  may be unknown and accessible only via sketching. Second, even if  $A$  is known, it may be too large to operate on or fit in memory. Finally, many matrices that arise in applications exhibit structure that enables fast matrix-vector products.

Hierarchical matrices are one such matrix class that frequently arises in practice. These matrices exhibit low-rank structure away from the diagonal, which represents the smoothness of long-range interactions between points in a discretized domain. Shorter-range interactions are treated recursively, as they are subdivided into finer domains over which the matrix is approximately low-rank again. This structure has been exploited in a variety of applications, including fast direct solvers for differential and integral equations, discretizations of boundary integral operators, preconditioners, and even infinite-dimensional operator learning. Below, we define  $H \in \mathbb{R}^{n \times n}$ , a hierarchical matrix with 3 levels of partitioning. Each off-diagonal block is given by a rank- $k$  factorization.

$$H = \begin{array}{|c|c|c|c|} \hline H_{11} & W_3 Z_3^\top & & \\ \hline U_3 V_3^\top & H_{22} & W_1 Z_1^\top & \\ \hline & U_1 V_1^\top & H_{33} & W_4 Z_4^\top \\ & & U_4 V_4^\top & H_{44} \\ \hline & U_0 V_0^\top & H_{55} & W_5 Z_5^\top \\ & & U_5 V_5^\top & H_{66} \\ \hline & U_2 V_2^\top & H_{77} & W_6 Z_6^\top \\ & & U_6 V_6^\top & H_{88} \\ \hline \end{array}$$

Peeling algorithms recover a hierarchical matrix like  $H$  using  $\mathcal{O}(k \log_2(n))$  sketches with  $H$  and  $H^\top$ . In general, they selectively apply the randomized SVD to recover all of the low-rank blocks at a given level, starting with the top level. That is, using  $\mathcal{O}(k)$  cleverly constructed input vectors which consist of alternating blocks of zeros and random Gaussian entries, one can restrict the outputs to sketch each of the individual low-rank blocks. Then, one can recover  $W_0 Z_0^\top$  and  $U_0 V_0^\top$  to high accuracy using a low-rank approximation algorithm. The learned blocks are stored in an approximation matrix  $\tilde{H}^{(1)} \in \mathbb{R}^{n \times n}$ :

$$\tilde{H}^{(1)} = \begin{bmatrix} 0 & \tilde{W}_0 \tilde{Z}_0^\top \\ \tilde{U}_0 \tilde{V}_0^\top & 0 \end{bmatrix}$$

These learned blocks are then “peeled” away, as the same process is applied to the matrix  $H - \tilde{H}^{(1)}$  to recover  $W_1 Z_1^\top, U_1 V_1^\top, W_2 Z_2^\top$ , and  $U_2 V_2^\top$  simultaneously with  $\mathcal{O}(k)$  sketches. This is because  $H - \tilde{H}^{(1)}$  zeros out the first level’s off-diagonal blocks, and subsequent matrix-vector products can sketch the action of the low-rank blocks at the next level. Once learned, these blocks are stored along with the first level’s blocks in  $\tilde{H}^{(2)}$ . The algorithm continues recursively, peeling away the learned blocks repeatedly and moving to finer blocks toward the diagonal. There are  $\log_2(n)$  levels, and each is recovered using  $\mathcal{O}(k)$  sketches, yielding an overall complexity of  $\mathcal{O}(k \log_2(n))$  queries.

Peeling algorithms are extremely useful and observed to be stable in practice. However, the pre-determined order of the algorithm, as well as its recursive subtraction, raise questions about its theoretical stability, particularly when the underlying matrix does not have hierarchical structure. For example, if the largest off-diagonal blocks are not exactly rank- $k$ , but rather numerically rank- $k$  as is often the case in applications, error may be propagated from the first level to all subsequent levels and deteriorate the overall approximation quality. In this talk, we describe the first provably stable and near-optimal variant of the peeling algorithm. That is, for a general matrix  $B$ , we use  $\mathcal{O}(k \log_2^4(n)/\varepsilon^3)$  sketches to obtain an approximation  $\tilde{B}$  satisfying  $\|B - \tilde{B}\|_F \leq (1 + \varepsilon)\|B - \hat{B}\|_F$ , where  $\hat{B}$  is the best hierarchical approximation to  $B$ . We complement this upper bound by proving that any matrix-vector query algorithm must use at least  $\Omega(k \log_2(n) + k/\varepsilon)$  queries to obtain a  $(1 + \varepsilon)$ -approximation.

We discuss the variety of techniques used to derive these results. To control the propagation of error between levels of hierarchical approximation, we introduce a new perturbation bound for low-rank approximation, which is of independent interest in numerical linear algebra. We show that the widely used Generalized Nyström method enjoys inherent stability when implemented with noisy matrix-vector products. This brings to light a surprising fact; the same result cannot be obtained if the more standard randomized SVD method is used for low-rank approximation within peeling.

For even stronger control of error buildup across recursive levels, we also introduce a new “randomly perforated” Gaussian sketching distribution. The key idea is to increase the sparsity of the query vectors, so that a higher fraction of nonzero blocks are set to zero. Thus, when recovering each block at a given level, we incur error due to a smaller number of inexactly recovered blocks from the previous levels. We note that this may not decrease the magnitude of error if the error is all concentrated on a few blocks. Thus, we choose the nonzero blocks of our sketches randomly, ensuring that the expected error when recovering each block at each level is small.

We also describe lower bounds on the query complexity of hierarchical matrix recovery and approximation. These results build on a growing body of work on lower bounds for adaptive matrix-vector product algorithms. We reduce the problem to fixed-pattern sparse matrix approximation, which arises when we restrict to recovering the diagonal block matrices of a hierarchical matrix, a strictly easier problem than hierarchical matrix recovery. Formally, we prove that, if we had an algorithm for finding a near-optimal hierarchical approximation with  $\mathcal{O}(k/\varepsilon)$  sketches, then the result could be post-processed to obtain a near-optimal block-diagonal approximation, which we know to be impossible. This is then combined with a query complexity lower bound for exact recovery to obtain the lower bound of  $\Omega(k \log_2(n) + k/\varepsilon)$ .

Finally, I will emphasize how our work in hierarchical matrix approximation fits into the new paradigm of stability analysis for randomized sketching algorithms, which are increasingly common in modern linear algebra techniques. Moving forward, we may also consider analyzing the stability of recovery algorithms for the subfamily of hierarchical semi-separable matrices, which also frequently arise in practice. Studying peeling also provides insight into an analysis of other similar recursive algorithms, such as butterfly and skeletonization factorizations.

# Convergence Analysis for Nonlinear GMRES

*Yunhui He*

Department of Mathematics, University of Houston, Houston, USA

## Abstract

We are interested in solving the following nonlinear system of equations

$$g(x) = 0, \quad (1)$$

where  $x \in \mathbb{R}^n$  and  $g : \mathbb{R}^n \rightarrow \mathbb{R}^n$ .

Consider the following fixed-point iteration

$$x_{k+1} = q(x_k) = x_k - g(x_k). \quad (2)$$

In practice, the fixed-point iteration converges slowly or even diverges. We seek methods to accelerate it.

Define the  $k$ -th residual of the fixed-point iteration as

$$r(x_k) = x_k - q(x_k) = g(x_k). \quad (3)$$

We revisit the nonlinear generalized minimal residual method (NGMRES) following [2, 1]. NGMRES has been used to accelerate the convergence of a fixed-point iteration, given by Algorithm 1. In practice, we consider the windowed NGMRES, i.e., fixing  $m$ , denoted as NGMRES( $m$ ), which is different than restart GMRES.

---

**Algorithm 1** NGMRES with window size  $m$ , denoted as (NGMRES( $m$ ))

---

- 1: Given  $x_0$  and  $m \geq 0$
- 2: For  $k = 1, 2, \dots$  until convergence Do:

- set  $m_k = \min\{k, m\}$
- compute

$$x_{k+1} = q(x_k) + \sum_{i=0}^{m_k} \beta_i^{(k)} (q(x_k) - x_{k-i}), \quad (4)$$

where  $\beta_i^{(k)}$  is obtained by solving the following least-squares problem

$$\min_{\beta_i^{(k)}} \left\| g(q(x_k)) + \sum_{i=0}^{m_k} \beta_i^{(k)} (g(q(x_k)) - g(x_{k-i})) \right\|_2^2. \quad (5)$$

---

EndDo

---

To the best of our knowledge, no convergence analysis exists for NGMRES( $m$ ) when applied to nonlinear problems. In this work, under some standard assumptions used for iterative methods in nonlinear problems, we prove that for general  $m > 0$ , the residuals of NGMRES( $m$ ) converge r-linearly. For  $m = 0$ , we prove that the residuals of NGMRES(0) converge q-linearly.

Finally, we present numerical results to demonstrate the performance of the NGMRES( $m$ ) method, and we make a comparison with the well-known Anderson acceleration [3]

## References

- [1] Sterck, Hans De, *Steepest descent preconditioning for nonlinear GMRES optimization*, Numerical Linear Algebra with Applications, 20(3): 453–471, 2013, Wiley Online Library
- [2] Sterck, Hans De *A nonlinear GMRES optimization algorithm for canonical tensor decomposition*, SIAM Journal on Scientific Computing, 34(3): A1351–A1379, 2012, SIAM
- [3] D. G. Anderson, *Iterative procedures for nonlinear integral equations*, J. Assoc. Comput. Mach., 12 (1965), pp. 547–560.

# Efficient Iterative Methods for the Solution of Sparse Tree-Coupled Saddle-Point Systems

*Bernhard Heinzelreiter, John W. Pearson, Christoph Hansknecht, Andreas Potschka*

## Abstract

The efficient solution of huge-scale sparse systems of linear or linearized equations poses a pivotal question in many applications of optimization and, with that, in numerical linear algebra. In particular, the design of bespoke iterative solvers is often invaluable, in order to mitigate the large storage requirements that may be required by off-the-shelf methods for systems of very high dimensions. In order to make a numerical solver feasible and effective, information about the linear system and the structure of the optimization problem from which it is obtained must frequently be taken into account when designing the solver.

A broad class of optimization problems with numerous applications involves sparsely-connected optimization problems. These consist of a series of constrained subproblems linked through a relatively small subset of variables. A general form of such problems can be written as

$$\begin{aligned} \min_{\zeta_1, \dots, \zeta_N} \quad & \sum_{i=1}^N \phi_i(\zeta_i) \\ \text{s.t.} \quad & C_{i,j}^+ \zeta_i + C_{i,j}^- \zeta_j = 0 \text{ for all } (i, j) \in \mathbb{A}, \\ & c_i(\zeta_i) = 0 \text{ for all } i \in \mathbb{V}, \end{aligned} \tag{1}$$

where  $\mathbb{V} = \{1, \dots, N\}$  with  $N \in \mathbb{N}$  is an index set for the subproblems, and  $\mathbb{A} \subseteq \mathbb{V} \times \mathbb{V}$  represent the connecting graph. The functions  $\phi_i$  denote the optimization functionals,  $c_i$  the constraint for each subproblem, and the vectors  $\zeta_i \in \mathbb{R}^{n_i}$  are to be determined. These problems play a crucial role in engineering applications, including stochastic programming, robust nonlinear model predictive control, and optimal control of networks (e.g., gas pipelines). They are also relevant in domain decomposition methods for partial differential equations (PDEs) and parallel-in-time approaches. Upon discretization and linearization, problem (1) reduces to a large-scale sparse linear system of saddle-point structure of which the efficient solution is desirable.

In this talk, we derive a suite of direct and (in particular) iterative solvers for saddle-point systems with a tree-coupled structure [2], corresponding to a special case of a linearization of (1). Specifically, we extend well-studied structure-exploiting approaches for saddle-point systems [1] by incorporating the graph-based coupling structure, where interactions between the individual and otherwise isolated subsystems are expressed via generic coupling constraints. This allows us to make use of the special, sparse structure of the resulting Schur complement. We develop a range of structured preconditioners which may be embedded within suitable Krylov subspace methods, including block preconditioners, recursive preconditioners, and multi-level approaches. The majority of these methods are vastly parallelizable, allowing them to be applied in a real-time fashion. We prove a range of results relating to the convergence, complexity, and spectral properties of our algorithms. The performance of the preconditioners is showcased by applying them to an array of model problems. This includes model predictive control, multiple shooting for optimal control, and domain decomposition for PDEs. The numerical experiments validate our theoretical results and show improved performance over direct methods. Additionally, the experiments show that our methods are capable of coping with very large regimes of  $N$ .

We, furthermore, outline future work on the analysis of coupled systems with an even more general graph-based structure, including cyclic dependencies and the derivation of methods to automatically detect this exploitable structure within given linear systems. There is also the potential to combine this with parallel-in-time methodologies derived by the author for fluid flow control problems [3]. Moreover, we discuss the potential of this method to be embedded within the sequential homotopy method [4], which leads to global convergence for a number of nonlinear optimization problems.

## References

- [1] Jacek Gondzio and Andreas Grothey. Parallel interior-point solver for structured quadratic programs: Application to financial planning problems. *Annals of Operations Research*, 152:319–339, 2007.
- [2] Christoph Hansknecht, Bernhard Heinzelreiter, John W. Pearson, and Andreas Potschka. A framework for the solution of tree-coupled saddle-point systems. arXiv:2410.23385. 2024.
- [3] Bernhard Heinzelreiter and John W. Pearson. Diagonalization-based parallel-in-time preconditioners for instationary fluid flow control problems. arXiv:2405.18964. 2024.
- [4] Andreas Potschka and Hans Georg Bock. A sequential homotopy method for mathematical programming problems. *Mathematical Programming, Series A*, 187(1–2):459–486, 2021.

# Randomized Householder-Cholesky QR Factorization with Multisketching

*Andrew J. Higgins, Daniel B. Szyld, Erik G. Boman, Ichitaro Yamazaki*

## Abstract

Computing the QR factorization of tall-and-skinny matrices is a critical component of many scientific and engineering applications, including the solution of least squares problems, block orthogonalization kernels for solving linear systems and eigenvalue problems within block or  $s$ -step Krylov methods, dimensionality reduction methods for data analysis like Principal Component Analysis, and many others. Two of the most popular high performance QR algorithms for tall-and-skinny matrices are the CholeskyQR2 and shifted CholeskyQR3 algorithms [3, 4], thanks to their communication-avoiding properties along with their exploitation of vendor provided highly-optimized dense linear algebra subroutines, allowing them to achieve high performance on rapidly evolving modern computer architectures. However, CholeskyQR2 may fail to accurately factorize a matrix  $V$  when its condition number  $\kappa(V) \gtrapprox \mathbf{u}^{-1/2}$ , where  $\mathbf{u}$  is unit roundoff [12]. Shifted CholeskyQR3 is numerically stable as long as  $\kappa(V) \lesssim \mathbf{u}^{-1}$ , but it requires over 50% more computational and communication cost than CholeskyQR2 [3]. Although TSQR [2] is a more stable communication-avoiding algorithm than the aforementioned Cholesky-based methods, it relies on a non-standard reduction operator, which can make it substantially slower than CholeskyQR2 in practice [4], and is significantly harder to implement efficiently on high performance GPUs. Hence, Cholesky-based QR methods remain popular on modern architectures.

Random sketching has become a popular dimension reduction technique in the fields of numerical linear algebra and data analysis. The central premise of random sketching is to embed a set  $\mathcal{V} \subset \mathbb{R}^n$  into a lower-dimensional space via some random projection  $S : \mathbb{R}^n \rightarrow \mathbb{R}^s$ , with  $s \ll n$ . In numerical linear algebra applications, the random sketch matrix  $S \in \mathbb{R}^{s \times n}$  is often selected to be an  $(\varepsilon, d, m)$  oblivious subspace embedding, i.e., for any  $m$ -dimensional subspace  $\mathcal{V} \subset \mathbb{R}^n$  and  $x \in \mathcal{V}$ , there is some  $\varepsilon \in [0, 1)$  such that

$$\sqrt{1 - \varepsilon} \|x\|_2 \leq \|Sx\|_2 \leq \sqrt{1 + \varepsilon} \|x\|_2,$$

with probability at least  $1 - d$  [8, 9]. Such  $(\varepsilon, d, m)$  oblivious subspace embeddings  $S$  are attractive in numerical linear algebra, because if one chooses the subspace  $\mathcal{V} \subset \mathbb{R}^n$  to be the column space of a matrix  $V \in \mathbb{R}^{n \times m}$ , the embeddings can be shown to approximately preserve singular values,

$$(1 + \varepsilon)^{-1/2} \sigma_{min}(SV) \leq \sigma_{min}(V) \leq \sigma_{max}(V) \leq (1 - \varepsilon)^{-1/2} \sigma_{max}(SV),$$

and therefore approximately preserve condition numbers,

$$\kappa(V) \leq \sqrt{\frac{1 + \varepsilon}{1 - \varepsilon}} \kappa(SV),$$

with probability at least  $1 - d$ . In the context of QR factorizations, one can factorize the small sketched matrix  $QR = SV$ , and use the triangular factor  $R$  as a preconditioner for the large unsketched matrix  $V$ , which is effective because

$$\kappa(VR^{-1}) \leq \sqrt{\frac{1 + \varepsilon}{1 - \varepsilon}} \kappa(SVR^{-1}) = \sqrt{\frac{1 + \varepsilon}{1 - \varepsilon}} = O(1),$$

for  $\varepsilon$  sufficiently below 1. This approach is known as the *sketch-and-precondition* framework [7].

In this talk, we present the results from our recent work [5], which analyzes a randomized tall-skinny QR algorithm called randomized Householder-Cholesky QR (`rand_cholQR`). The algorithm uses the sketch-and-precondition framework with Householder QR as a preprocessing step before following up with a pass of CholeskyQR to fully orthogonalize the preconditioned matrix with little computational and communication cost. In order to reduce the cost of the computations even further, we propose to use “multisketching,” i.e., the use of two consecutive random sketch matrices, within the sketch-and-precondition framework. Our approach is general in the sense that our analysis applies to any two oblivious subspace embedding sketching matrices, but is specifically motivated by the use of a large sparse sketch followed by a smaller dense sketch, such as a Gaussian or Radamacher sketch [1], as this particular strategy significantly reduces the complexity of applying the sketch operator. Our analysis applies in particular to Count-Gauss (one application of CountSketch followed by a Gaussian sketch), as described in [6, 10, 11].

We prove that with high probability, the orthogonality error of `rand_cholQR` is on the order of unit roundoff for any numerically full-rank matrix  $V$  (i.e.,  $\kappa(V) \lesssim \mathbf{u}^{-1}$ ) and hence it is as stable as shifted CholeskyQR3 and it is significantly more numerically stable than CholeskyQR2. Our numerical experiments illustrate the theoretical results and suggest that `rand_cholQR` often succeeds for numerically rank-deficient problems as well, unlike either CholeskyQR2 or shifted CholeskyQR3. In addition, the `rand_cholQR` algorithm may be implemented using the same basic linear algebra kernels as CholeskyQR2. Therefore, it is simple to implement and has the same communication-avoiding properties. We perform a computational study on a state-of-the-art GPU to demonstrate that `rand_cholQR` can perform up to 4% faster than CholeskyQR2 and 56.6% faster than shifted CholeskyQR3, while significantly improving the robustness of CholeskyQR2.

In summary, our primary contribution consists of a new error analysis of a multisketched randomized QR algorithm, proving it can be safely used for matrices of larger condition number than CholeskyQR2 can handle. Numerical experiments confirm and illustrate the theory. Our secondary contribution is a computational study on a state-of-the-art GPU that tangibly demonstrates that the multisketched algorithm has superior performance over the single sketch algorithms and similar performance to the high performance but less stable CholeskyQR2 algorithm.

## References

- [1] Dimitris Achlioptas. Database-friendly random projections: Johnson- Lindenstrauss with binary coins. *Journal of Computer and System Sciences*, 66:671–687, 2003. Special Issue on PODS 2001.
- [2] James W. Demmel, Laura Grigori, Mark Hoemmen, and Julien Langou. Communication-optimal parallel and sequential QR and LU factorizations. *SIAM Journal on Scientific Computing*, 34:A206–A239, 2012.
- [3] Takeshi Fukaya, Ramaseshan Kannan, Yuji Nakatsukasa, Yusaku Yamamoto, and Yuka Yanagisawa. Shifted Cholesky QR for computing the QR factorization of ill-conditioned matrices. *SIAM Journal on Scientific Computing*, 42:477–503, 2020.
- [4] Takeshi Fukaya, Yuji Nakatsukasa, Yuka Yanagisawa, and Yusaku Yamamoto. CholeskyQR2: A simple and communication-avoiding algorithm for computing a tall-skinny QR factorization on a large-scale parallel system. In *Proceedings of ScalA: 5th Workshop on Latest Advances in Scalable Algorithms for Large-Scale Systems*, pages 31–38, Los Alamitos, CA, 2014. IEEE Computer Society.

- [5] Andrew J. Higgins, Daniel B. Szyld, Erik G. Boman, and Ichitaro Yamazaki. Analysis of Randomized Householder-Cholesky QR Factorization with Multisketching, 2024. arXiv:2309.05868.
- [6] Michael Kapralov, Vamsi Potluru, and David Woodruff. How to fake multiply by a Gaussian matrix. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48, pages 2101–2110. Proceedings of Machine Learning Research, 2016.
- [7] Per-Gunnar Martinsson and Joel A. Tropp. Randomized numerical linear algebra: foundations and algorithms. *Acta Numerica*, 29:403–572, 2020.
- [8] Yuji Nakatsukasa and Joel A. Tropp. Fast & accurate randomized algorithms for linear systems and eigenvalue problems, 2021. arXiv:2111.00113.
- [9] Tamás Sarlós. Improved approximation algorithms for large matrices via random projections. In *Proceedings of 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS’06)*, pages 143–152, Los Alamitos, CA, 2006. IEEE Computer Society.
- [10] Aleksandros Sobczyk and Efstratios Gallopoulos. Estimating leverage scores via rank revealing methods and randomization. *SIAM Journal on Matrix Analysis and Applications*, 42:199–1228, 2021.
- [11] Aleksandros Sobczyk and Efstratios Gallopoulos. pylspack: Parallel algorithms and data structures for sketching, column subset selection, regression, and leverage scores. *ACM Transactions on Mathematical Software*, 48:1–27, 2022.
- [12] Yusaku Yamamoto, Yuji Nakatsukasa, Yuka Yanagisawa, and Takeshi Fukaya. Roundoff error analysis of the Cholesky QR2 algorithm. *Electronic Transactions on Numerical Analysis*, 44:306–326, 2015.

# Optimal preconditioners for nonsymmetric multilevel Toeplitz systems with application to solving non-local evolutionary PDEs

*Yuan-Yuan Huang, Sean Y. Hon, Lot-Kei Chou, and Siu-Long Lei.*

## Abstract

Preconditioning for multilevel Toeplitz systems has long been a focal point of research in numerical linear algebra. In this talk, we present a new preconditioning method for nonsymmetric multilevel Toeplitz systems, including those from evolutionary PDEs. These systems have recently garnered considerable attention in the literature. For these equations, we propose a symmetric positive definite multilevel Tau preconditioner that is efficient and optimal, ensuring mesh-independent convergence with the preconditioned generalized minimal residual method. Numerical examples highlight our method's effectiveness, particularly for non-local, time-dependent PDEs solved in parallel.

# Contour Integral Methods for Exponentials of Matrices and Operators with Explicit High-Order Error Bounds

*Andrew Horning\* and Adam R. Gerlach*

\**Department of Mathematical Sciences, Rensselaer Polytechnic Institute*

## Abstract

Exponential integrators based on contour integral transforms lead to powerful numerical solvers for a variety of ODEs, PDEs, and other time-evolution equations. They are easy to parallelize and lead to global-in-time approximations that can be efficiently evaluated anywhere within a finite time horizon. However, there are theoretical challenges that restrict their use-cases to classes of smooth evolution equations called analytic semigroups. In this talk, we show how to use carefully regularized contour integral representations to construct high-order quadrature schemes for the much larger class of strongly continuous semigroups. Our algorithms are accompanied by explicit high-order error bounds and near-optimal parameter selection. We illustrate the method's attractive features through several PDE examples associated with singular behavior, causality, and non-normality. Our approach is firmly rooted in contour integral techniques for matrix functions. Along the way, we highlight how simple ideas from semigroup theory can augment traditional techniques in numerical linear algebra to tackle common tensions that arise while computing functions of operators.

**Contour integral methods.** Contour integral methods approximate the action of the matrix exponential with a quadrature rule. Given  $\gamma$ , a Jordan curve winding once clockwise around the spectrum of a square matrix  $A$ , quadrature nodes  $z_1, \dots, z_N$ , and weights  $w_1, \dots, w_N$ , then

$$\exp(At)x = \frac{1}{2\pi i} \int_{\gamma} e^{zt}(zI - A)^{-1}x dz \approx \frac{1}{2\pi i} \sum_{k=1}^N w_k e^{z_k t} (z_k I - A)^{-1}x. \quad (1)$$

The main cost of the scheme is solving a shifted linear equation at each quadrature node. These can be done in parallel and may be accelerated with iterative methods. After solving these linear systems, the approximation can be evaluated at any time  $t > 0$  with only vector-vector operations.

When  $A$  is obtained by discretizing a differential operator, its spectrum may fill a large region of the complex plane. As the discretization is refined, this region typically grows and the quadrature approximation in (1) may rapidly deteriorate along with the performance of iterative methods for the linear systems. Famously, this effect can be mitigated through *Talbot quadrature* if the spectrum of  $A$  lies in a modest sector along the negative real axis [1]. These techniques have been used to develop highly parallelizable schemes for a broad class of parabolic equations [2]. Recently, Colbrook was awarded the Leslie Fox Prize for extending these techniques to analytic semigroups, which are associated with smooth evolution problems like parabolic and damped wave equations [3].

**Regularization.** Fundamentally, Talbot contour integral schemes exploit *regularity in the operator exponential* for analytic semigroups. Unfortunately, this is not possible for less regular evolution equations associated with challenging causal, singular, or non-normal behavior. Instead, we show how to regularize contour integral schemes to exploit *regularity in the vector  $x$* . This is common in time-stepping methods but global-in-time methods require new tools from semigroup theory.

To exploit regularity in  $x$  in our scheme and analysis, we develop the algorithm at the infinite-dimensional level and then carefully assess the impact of discretization. Consider a linear operator

$\mathcal{A} : D(\mathcal{A}) \rightarrow \mathcal{X}$  on a separable Banach space  $\mathcal{X}$  and replace  $x$  by  $u \in \mathcal{X}$  and  $A$  by  $\mathcal{A}$ . If  $\mathcal{A}$  generates a strongly continuous semigroup on  $\mathcal{X}$ , then its spectrum is contained in a left half-plane  $\text{Im}(z) < \omega$  and its resolvent decays uniformly in the complimentary right-half plane. We use a regularized analogue of the contour integral in (1) with form, for some  $\delta > \omega$  and integer  $m \geq 2$ ,

$$\exp(\mathcal{A}t)u = (2\delta\mathcal{I} - \mathcal{A})^m \frac{1}{2\pi i} \int_{\delta-i\infty}^{\delta+i\infty} \frac{e^{zt}}{(2\delta-z)^m} (z\mathcal{I} - \mathcal{A})^{-1}x dz, \quad u \in D(\mathcal{A}^2). \quad (2)$$

To approximate (2), we apply a truncated  $N$ -point trapezoidal rule with node spacing  $h > 0$ ,

$$S_N^{(m)}(t)u = (2\delta\mathcal{I} - \mathcal{A})^m \left[ \frac{h}{2\pi} \sum_{k=-N}^N \frac{e^{(\delta+ihk)t}}{(\delta-ihk)^m} ((\delta+ihk)\mathcal{I} - \mathcal{A})^{-1} \right] u. \quad (3)$$

For a practical computational scheme, the shifted linear equations can be solved with a suitable discretization of smooth functions  $u \in D(\mathcal{A}^2) \subset \mathcal{X}$ . The main requirement for convergence is that  $Ax \rightarrow \mathcal{A}u$  and  $(z_k\mathcal{I} - A)^{-1}x \rightarrow (z_k\mathcal{I} - \mathcal{A})^{-1}u$ ,  $k = -N, \dots, N$ , as the discretization is refined.

**Explicit Error Bounds.** At the operator level, the regularized quadrature approximation in (2)-(3) has several key advantages over (1). The integral converges absolutely and uniformly for  $u \in D(\mathcal{A}^2)$ , while the amplification power of  $(2\delta\mathcal{I} - \mathcal{A})^m$  is controlled by the regularity of the vector  $u$ . Moreover, the integrand decays at a controlled rate along the contour. These features allow us to derive simple explicit error bounds for smooth vectors  $u \in D(\mathcal{A}^m)$ . In fact, we can derive closed-form expressions for quadrature parameters that achieve [4]

$$\sup_{0 \leq t \leq T} \|\exp(\mathcal{A}t)u - S_N^{(m)}(t)u\| \leq C \left( \frac{\delta}{hN} \right)^{m-1} \|(2\delta\mathcal{I} - \mathcal{A})^m\|, \quad \text{when } u \in D(\mathcal{A}^m). \quad (4)$$

Here,  $C$  is an explicit constant depending on  $A$ , the contour location  $\delta$ , and the time horizon  $T$ .

In practice, one must discretize the infinite-dimensional objects for a numerical algorithm. We show how semigroup theory provides simple, computable bounds for the action of the discretized resolvent that hold for a very broad class of discretization techniques including Galerkin-based schemes and modern spectral methods. These bounds are inspired by the residual bounds for linear systems commonly used in numerical linear algebra.

**Applications and Outlook.** High-order contour integral schemes for strongly continuous semigroups open the door to highly parallelizable methods for traditionally challenging PDEs and related time-evolution processes. They also provide a compact modal-like representation of semigroups that may be useful in operator learning, system identification, and other data-driven modeling techniques. We will illustrate some applications of our scheme to challenging simulation scenarios related to uncertainty quantification, where verified simulation of Koopman semigroups is a key ingredient. We will also discuss work in progress on applications of contour integral representations in data-driven settings. Finally, we will outline the potential of semigroup theory to resolve related tensions encountered while computing matrix functions of discretized operators.

- [1] A. TALBOT, *The accurate numerical inversion of Laplace transforms*, IMA J. Appl. Math., (1979).
- [2] L.N. TREFETHEN, J.A.C. WEIDEMAN, AND T. SCHMELZER, *Talbot quadratures and rational approximations*, BIT Numer. Math., (2006).
- [3] M.J. COLBROOK, *Computing semigroups with error control*, SIAM J. Numer. Anal., (2022).
- [4] A. HORNING AND A.R. GERLACH., *A family of high-order accurate contour integral methods for strongly continuous semigroups*, Arxiv preprint arXiv:2408.07691, (2024).

# Least squares solvers based on randomized normal equations

Ilse C.F. Ipsen

## Abstract

For the better part of my life I have taught that least squares problems are to be solved with a QR decomposition or SVD, cautioning that formation of the normal equations is to be avoided if possible.

Now I am re-thinking this advice, in light of developments underlying the Blendenpik least squares solver [1, 2], and our version of the randomized preconditioned Cholesky-QR algorithm [3].

**Proposed Algorithm.** Given a real  $m \times n$  matrix  $\mathbf{A}$  with  $\text{rank}(\mathbf{A}) = n$ , we investigate the solution of the least squares problems  $\min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|_2$  by solving the normal equation of a randomized preconditioned problem. In the spirit of the original Householder meetings, this is work in progress.

Specifically, we right-precondition  $\mathbf{A}$  with a randomized preconditioner  $\mathbf{R}_s$ , to obtain  $\mathbf{A}_1 \equiv \mathbf{AR}_s^{-1}$ . Instead of taking the Blendenpik route and solving  $\min_{\mathbf{y}} \|\mathbf{A}_1\mathbf{y} - \mathbf{b}\|_2$  via the iterative method LSQR, we solve the normal equations. That is, the Gram matrix  $\mathbf{G} \equiv \mathbf{A}_1^T \mathbf{A}_1$  is formed explicitly, followed by solution of the normal equations  $\mathbf{G}\mathbf{y} = \mathbf{A}_1^T \mathbf{b}$ . This can be done with a Cholesky factorization of  $\mathbf{G}$  or of the bordered matrix  $[\mathbf{A}_1 \ \mathbf{b}]$  [4, §2.2]. At last, one recovers the solution of the original problem via the triangular solve  $\mathbf{R}_s \mathbf{x} = \mathbf{y}$ .

To compute the randomized preconditioner  $\mathbf{R}_s$ , first improve the coherence of  $\mathbf{A}$  with a random orthogonal matrix  $\mathbf{FA}$ , where  $\mathbf{F}$  is the product of a fast transform (FFT, Walsh-Hadamard, DCT, Hartley) and a random diagonal matrix with independent Rademacher variables on the diagonal. Then sample  $c$  rows, uniformly and independently with replacement from  $\mathbf{FA}$  to obtain the sampled matrix  $\mathbf{A}_s = \mathbf{SFA}$ . At last compute the thin QR decomposition  $\mathbf{A}_s = \mathbf{Q}_s \mathbf{R}_s$ .

**Advantages.** Unlike Blendenpik [1] which solves a  $m \times n$  least squares problem with an iterative method, we solve a small  $n \times n$  problem with a direct method. Direct methods, and Cholesky decompositions in particular, tend to perform well on cache-based and parallel architectures, where data movement and synchronization can dominate arithmetic. This is in contrast to the normal equations like approach with iterative methods in [5, 6], which also requires an initial guess.

The simplicity of our approach, in contrast to the involved multi-stage [5, Algorithm 4], will lead to a rigorous and informative perturbation analysis for the accuracy of the computed solution. The potential backward stability issues due to the formation of the Gram matrix can be handled in the same way as for the randomized Cholesky-QR algorithm in [3].

The preconditioner  $\mathbf{R}_s$  needs to be applied only once and is applied explicitly, thereby improving the backward stability issues discussed in [7]. Even for matrices  $\mathbf{A}$  with worst case coherence and a condition number  $\kappa(\mathbf{A}) \approx 10^{15}$ , a sampling amount of  $c = 3n$  suffices to produce preconditioned matrices  $\mathbf{A}_1$  that are very well conditioned, with condition numbers  $\kappa(\mathbf{A}_1) \approx 10$ .

Preliminary numerical results suggest that for matrices with condition number  $\kappa(\mathbf{A}) \leq 10^9$ , the preconditioner  $\mathbf{R}_s$  can be computed faster, in single precision, without loss of accuracy in the preconditioned problem. We will show that solving the normal equations via a Cholesky decomposition represents an efficient least squares solver on NVIDIA RTX 2080 GPUs.

- [1] H. Avron, P. Maymounkov, and S. Toledo. Blendenpik: Supercharging Lapack’s Least-Squares Solver, *SIAM J. Sci. Comput.*, vol. 32, no. 3, pp 1217–1236 (2010)
- [2] I.C.F. Ipsen and T. Wentworth. The Effect of Coherence on Sampling from Matrices with Orthonormal Columns, and Preconditioned Least Squares Problems, *SIAM J. Matrix Anal. Appl.*, vol. 35, no. 4, pp 149–1520 (2014)
- [3] J.E. Garrison, and I.C.F. Ipsen. A Randomized preconditioned Cholesky-QR Algorithm, *SIAM J. Matrix Anal. Appl.*, under review (2024)
- [4] Å. Björck. *Numerical Methods for Least Squares Problems*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA (1996)
- [5] E.N. Epperly, M. Meier and Y. Nakatsukasa. Fast Randomized Least-Squares Solvers can be just as Accurate and Stable as Classical Direct Solvers, arXiv:2406.03468 (2024)
- [6] R. Xu and Y. Lu. Randomized Iterative Solver as Iterative Refinement: A Simple Fix Towards Backward Stability, arXiv:2410.11115 (2024)
- [7] M. Meier, Y. Nakatsukasa, A. Townsend, and M. Webb. Are sketch-and-precondition least squares solvers numerically stable? *SIAM J. Matrix Anal. Appl.*, vol. 45, no. 2, pp 905–929 (2024)

# Randomized orthogonalization in GMRES with deflation and augmentation

Yongseok Jang, Laura Grigori

## Abstract

In the context of large-scale linear algebra, random sketching has emerged as a powerful technique to reduce computational costs and memory requirements. As data dimensions grow, traditional methods often become impractical. Random sketching provides a way to approximate large matrices and datasets by projecting them onto lower-dimensional subspaces using randomized linear maps, which capture essential properties of the original data –such as norms of vectors– but at a fraction of the size.

In this talk, we present the application of random sketching to Gram-Schmidt process for orthonormalizing a set of vectors. One [1] can provide a set of vectors, which are not  $l_2$  orthogonal but their low dimensional images through random sketching are orthonormal. Using this method, called *randomized Gram-Schmidt* (RGS) algorithm, for QR factorization of a matrix  $W \in \mathbb{R}^{n \times m}$  (with  $m \leq n$ ), we obtain  $W = QR$ , where  $Q$  is not orthonormal, but  $\Theta Q$  becomes orthonormal with a random sketching matrix  $\Theta \in \mathbb{R}^{t \times n}$  for  $t \ll n$ . Furthermore, inspired by the reorthogonalization techniques in *classical Gram-Schmidt* (CGS) and *modified Gram-Schmidt* (MGS) (namely CGS2 and MGS2, respectively), we develop the RGS2 algorithm [2]. The RGS2 algorithm combines RGS with CGS/MGS to result improved numerical stability and  $l_2$  orthonormal  $Q$ .

By employing fast computation techniques for sketching, such as fast Walsh Hadamard transformation, our RGS algorithms bring significant computational cost reductions. RGS has half the complexity of CGS/MGS, and RGS2 reduces computational costs by 25% compared to CGS2/MGS2. Furthermore, with the probabilistic rounding model, we analyze rounding errors and show that RGS exhibits better numerical stability than CGS and comparable stability to MGS. Additionally, under a numerical non-singularity condition, the loss of orthogonality in RGS2 is independent of the condition number of  $W$ . Thus, the randomized orthogonalization process offers both reduced computational cost and enhanced numerical stability.

When solving linear systems with GMRES, the quality of Krylov basis vectors is crucial; poor quality can deteriorate GMRES convergence. To accelerate the convergence of GMRES, a deflation strategy is combined together. However, in GMRES with deflated restarting (GMRES-DR), where the previous Krylov subspace is reused, loss of orthogonality may accumulate over iterations, potentially leading to stagnation or divergence. Hence, the orthogonalizing process is particularly important in GMRES-DR. Here, RGS-based Arnoldi iterations can ensure numerical stability in generating Krylov basis vectors rather than other GS algorithms. Consequently, the randomized GMRES and the randomized GMRES-DR exhibit better numerical performance [3].

In this presentation, we introduce the randomized variants of FGMRES-DR, FGCRO-DR, SVD based deflation, and augmentation, (e.g., please refer to [4, 5, 6, 7] and the references therein for the GMRES methods with deflation and augmentation). To validate the stability and convergence improvements, we present numerical examples that solve ill-conditioned systems arising from compressible turbulent CFD simulations.

## References

- [1] Balabanov, Oleg, and Laura Grigori. “Randomized Gram–Schmidt process with application to GMRES.” *SIAM Journal on Scientific Computing* 44, no. 3 (2022): A1450-A1474.
- [2] Jang, Yongseok, and Laura Grigori. “Randomized orthogonalization process with reorthogonalization.” HAL open science (2024).
- [3] Jang, Yongseok, Laura Grigori, Emeric Martin, and Cédric Content. “Randomized flexible GMRES with deflated restarting.” *Numerical Algorithms* (2024): 1-35.
- [4] Giraud, Luc, Serge Gratton, Xavier Pinel, and Xavier Vasseur. “Flexible GMRES with deflated restarting.” *SIAM Journal on Scientific Computing* 32, no. 4 (2010): 1858-1878.
- [5] Parks, Michael L., Eric De Sturler, Greg Mackey, Duane D. Johnson, and Spandan Maiti. “Recycling Krylov subspaces for sequences of linear systems.” *SIAM Journal on Scientific Computing* 28, no. 5 (2006): 1651-1674.
- [6] Daas, Hussam Al, Laura Grigori, Pascal Hénon, and Philippe Ricoux. “Recycling Krylov subspaces and truncating deflation subspaces for solving sequence of linear systems.” *ACM Transactions on Mathematical Software (TOMS)* 47, no. 2 (2021): 1-30.
- [7] Soodhalter, Kirk M., Eric de Sturler, and Misha E. Kilmer. “A survey of subspace recycling iterative methods.” *GAMM-Mitteilungen* 43, no. 4 (2020): e202000016.

# On the Convergence of the Singular Value Expansion of 2D functions

Sungwoo Jeong, Alex Townsend

## Abstract

In this work we study the convergence of the singular value expansion (SVE) of 2D functions (kernels). Consider a square-integrable kernel  $K : [a, b] \times [c, d] \rightarrow \mathbb{R}$ , where  $[a, b], [c, d] \subset \mathbb{R}$ . We define (i) Right singular functions denoted by  $u_1, u_2, \dots$ , which are orthonormal with respect to  $L^2([a, b])$  and (ii) Left singular functions denoted by  $v_1, v_2, \dots$ , which are orthonormal with respect to  $L^2([c, d])$ . These singular functions are defined to satisfy the relationships

$$\sigma_n u_n(x) = \int_c^d K(x, y) v_n(y) dy, \quad \sigma_n v_n(y) = \int_a^b K(x, y) u_n(x) dx. \quad (1)$$

The values  $\sigma_1 \geq \sigma_2 \geq \dots > 0$  are called the (positive) singular values of  $K$ . The SVE of  $K$  is then defined as

$$K(x, y) = \sum_{n=1}^{\infty} \sigma_n u_n(x) v_n(y). \quad (2)$$

Recall that the singular vectors of a matrix  $A$  is defined with relationships  $Av_n = \sigma_n u_n$ ,  $u_n^* A = \sigma_n v_n^*$  and the singular value decomposition (SVD) can be defined as  $A = \sum_n \sigma_n u_n v_n^*$ . Thus, the SVE can be thought of as a continuous analogue of the SVD [1].

Before the SVD of a matrix, several pioneers of modern functional analysis in the early 20th century figured out the existence and properties of the SVE for a general square-integrable kernel. Within these developments, Mercer [2], in 1909, showed that any continuous, symmetric positive definite kernel  $K : [a, b] \times [a, b] \rightarrow \mathbb{R}$  has a uniformly and absolutely converging SVE,

$$K(x, y) = \sum_{n=1}^{\infty} \lambda_n u_n(x) u_n(y), \quad (x, y) \in [a, b] \times [a, b], \quad (3)$$

which is also equivalent to its eigenfunction expansion. This is often called Mercer's theorem. For general kernels without positive definiteness or symmetry, the convergence property (pointwise, uniform, and absolute) of the SVE is an open problem.

In this work, we first prove that the conclusion of Mercer's theorem does not hold for general symmetric and asymmetric kernels, whenever the positive-definiteness condition is dropped. We provide novel examples which lead to the following result.

**Theorem 1.** *For any  $[a, b] \subset \mathbb{R}$  there are continuous symmetric indefinite kernels on  $[a, b] \times [a, b]$  such that the SVE, equation (2), (i) does not converge pointwise, (ii) converges pointwise but not uniformly, or (iii) converges pointwise but not absolutely.*

We hope this theorem will clarify some confusion in the literature regarding the convergence of the SVE whenever a symmetric kernel is not positive definite. In practice, a symmetric indefinite kernel often possesses a pointwise converging SVE but we prove that such convergence is not always guaranteed. Our work provides a rigorous underpinning for kernel methods using indefinite and asymmetric kernels.

We then prove our second main result, which is the convergence result of the SVE when a kernel is equipped with a mild regularity condition. We say a kernel  $K : [a, b] \times [c, d] \rightarrow \mathbb{R}$  is of *uniform bounded variation* if

$$\int_a^b \frac{\partial}{\partial x} K(x, y) dx < V, \quad \int_c^d \frac{\partial}{\partial y} K(x, y) dy < V, \quad (4)$$

holds for any fixed  $x, y$  and a uniform constant  $V > 0$ . We remark that this is a larger class of general continuous kernels which includes, for instance, Lipschitz continuous kernels. For a continuous kernel of uniform bounded variation, we prove the following result using the singular value decay and a generalization of the Rademacher-Menchov theorem. (In fact, we prove that the same conclusion holds for any continuous kernel that has a singular value decay  $\sigma_n = \mathcal{O}(n^{-\alpha})$  with  $\alpha > \frac{1}{2}$ .)

**Theorem 2.** *For any  $[a, b], [c, d] \subset \mathbb{R}$ , let  $K : [a, b] \times [c, d] \rightarrow \mathbb{R}$  be a continuous kernel of uniform bounded variation (see equation (4)). Then, the SVE of  $K$ , equation (2), converges pointwise almost everywhere, unconditionally almost everywhere, and almost uniformly.*

To prove the second theorem, we also provide a new bound on the decay of singular values, which is stated in the following proposition. We use a recent result [3] on the decay of the error of truncated Legendre series approximation to prove the decay bound.

**Proposition 1.** *For any  $[a, b], [c, d] \subset \mathbb{R}$ , a continuous kernel  $K : [a, b] \times [c, d] \rightarrow \mathbb{R}$  of uniform bounded variation has  $\sigma_n = \mathcal{O}(n^{-1})$  as  $n \rightarrow \infty$ .*

Furthermore, we provide an efficient numerical algorithm for computing the SVE of a given function. The algorithm is divided into two steps. In the first step, we compute a pseudo-skeleton approximation using Gaussian elimination with complete pivoting (GECP), which is an iterative procedure to approximate the kernel  $K(x, y)$  as a sum of rank-1 functions [4]. After we have computed a rank  $\leq R$  pseudo-skeleton approximation,  $K_R(x, y)$ , in the first step, we improve it by performing a low-rank SVD. The SVD decomposes  $K_R(x, y)$  into a sum of outer products of orthonormal functions with singular values and gives us an accurate truncated singular value expansion of  $K$ .

## References

- [1] Alex Townsend and Lloyd N Trefethen. Continuous analogues of matrix factorizations. *Proc. Roy. Soc. A*, 471(2173):20140585, 2015.
- [2] James Mercer. XVI. Functions of positive and negative type, and their connection the theory of integral equations. *Philosophical Transactions of the Royal Society of London. Series A*, 209(441-458):415–446, 1909.
- [3] Haiyong Wang. New error bounds for Legendre approximations of differentiable functions. *Journal of Fourier Analysis and Applications*, 29(4):42, 2023.
- [4] Alex Townsend and Lloyd N Trefethen, An extension of Chebfun to two dimensions. *SIAM Journal on Scientific Computing*, 35(6):C495–C518, 2013.

# On a Multi-Stage Tensor Reduction Strategy for Arbitrary Order- $p$ Tensorial Data under the Tensor T-Product Algebra

Harshit Kapadia, Lihong Feng, Peter Benner

## Abstract

We present a novel multi-stage tensor reduction (MSTR) framework for tensorial data arising from experimental measurements or high-fidelity simulations of physical systems. The order  $p$  of the tensor under consideration can be arbitrarily large. At the heart of the framework are a series of strategic tensor factorizations and compressions, ultimately leading to a *final* order-preserving reduced representation of the original tensor. We also augment the MSTR framework by performing efficient kernel-based interpolation/regression over certain reduced tensor representations, amounting to a new non-intrusive model reduction approach capable of handling dynamical, parametric steady, and parametric dynamical systems. Furthermore, to efficiently build the parametric reduced-order model in the offline stage, we develop a tensor empirical interpolation method (t-EIM). We formalize our ideas using the tensor t-product algebra [7, 3, 6] and provide a rigorous upper bound for the error of the tensor approximation from the MSTR strategy.

The idea to factor any order-3 tensor in two orthogonal tensors of order-3 and an  $f$ -diagonal tensor of order-3 first appeared in [7]. The notion of orthogonal tensors and such a factorization strategy, analogous to matrix factorization—rendered by the singular value decomposition (SVD)—is possible due to the tensor multiplication, referred to as the t-product [7]. An extension for order- $p$  tensors of the t-product and t-SVD is proposed in [8], which is used in our work. Moreover, by following the approach taken in [9] for order-3 tensors, we develop a randomized variant of t-SVD for order- $p$  tensors, which is utilized to accelerate the tensor factorizations in our MSTR framework.

To aid our discussion, let us define the t-SVD for an order- $p$  tensor  $\mathcal{T}$ :

$$\mathcal{T} = \mathcal{L} * \mathcal{M} * \mathcal{R}^\top, \quad (1)$$

where  $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times n_3 \times \cdots \times n_p}$ ,  $\mathcal{L} \in \mathbb{R}^{n_1 \times n_1 \times n_3 \times \cdots \times n_p}$ ,  $\mathcal{R} \in \mathbb{R}^{n_2 \times n_2 \times n_3 \times \cdots \times n_p}$ , and  $\mathcal{M} \in \mathbb{R}^{n_1 \times n_2 \times n_3 \times \cdots \times n_p}$ . Here,  $\mathcal{L}$  and  $\mathcal{R}$  are orthogonal, and  $\mathcal{M}$  has entries  $\mathcal{M}_{i_1 i_2 i_3 \dots i_p}$  such that  $\mathcal{M}_{i_1 i_2 i_3 \dots i_p} = 0$  unless  $i_1 = i_2$ . When  $p = 3$ , authors in [7] refer to  $\mathcal{M}$  as an  $f$ -diagonal tensor. The symbol  $*$  in (1) refers to the t-product, and  $\mathcal{R}^\top$  is the t-transpose of  $\mathcal{R}$ .

We are concerned with data arising from high-fidelity simulations or physical measurements. In the most general setting, the solution tensor  $\mathcal{S}$  can have the following form:

$$\mathcal{S} \in \mathbb{R}^{N_x \times N_y \times N_z \times m_1 \times m_2 \times \cdots \times m_{N_\mu} \times N_t}, \quad (2)$$

where  $N_x$ ,  $N_y$ , and  $N_z$  refer to the size of each spatial dimension;  $N_\mu$  refers to the parameter space dimensions, with  $m_1, m_2, \dots, m_{N_\mu}$  corresponding to the size of each dimension of the parameter space;  $N_t$  refers to the total number of time steps or the frequency of measurements. We seek to efficiently reduce this high-dimensional tensorial data, obtaining its reduced tensor representation, which describes the original tensor with reasonable accuracy.

The MSTR strategy begins by identifying the *target variable* of interest, along which we do not seek to perform a tensor compression. For physical systems, it is typical to either collect measurements or high-fidelity solution values across the spatial domain, corresponding to various time instances and/or parameter configurations. As a result, the *target variable* could be either time or parameter.

We seek to compress the solution tensor along all remaining dimensions. For instance, consider an order-5 tensor  $\mathcal{S} \in \mathbb{R}^{N_x \times N_y \times N_z \times m \times N_t}$ , where  $m = \prod_{j=\{1,2,\dots,N_\mu\}} m_j$ . If the *target variable* is the parameter, then the reduced representation we aim for lies in  $\mathbb{R}^{r_1 \times r_2 \times r_3 \times m \times r_5}$ , whereas if the *target variable* is time, then the reduced representation we aim for lies in  $\mathbb{R}^{r_1 \times r_2 \times r_3 \times r_4 \times N_t}$ . Here,  $r_1 \ll N_x$ ,  $r_2 \ll N_y$ ,  $r_3 \ll N_z$ ,  $r_4 \ll m$ , and  $r_5 \ll N_t$ .

Based on the t-SVD in (1), we can seek a compression of any tensor  $\mathcal{T} \in \mathbb{R}^{n_1 \times n_2 \times n_3 \times \dots \times n_p}$  by truncating  $\mathcal{L} \in \mathbb{R}^{n_1 \times n_1 \times n_3 \times \dots \times n_p}$  along the second dimension, obtaining  $\tilde{\mathcal{L}} \in \mathbb{R}^{n_1 \times r \times n_3 \times \dots \times n_p}$ , where  $r \ll n_1$ , projecting  $\mathcal{T}$  on  $\tilde{\mathcal{L}}$ , and producing  $\mathcal{A} \in \mathbb{R}^{r \times n_2 \times n_3 \times \dots \times n_p}$ . Note that  $\mathcal{A}$  provides a reduced representation of  $\mathcal{T}$ , where information along the first dimension is compressed.

The central idea pertaining to the MSTR strategy is to recursively perform a tensor factorization for obtaining a truncated orthogonal tensor, onto which the parent unfactored tensor can be projected, leading to compression along one tensor dimension at every stage. The tensor factorizations are performed sequentially over subsequent *intermediate* reduced representations of the original tensor  $\mathcal{S}$ , ultimately leading to the *final* reduced tensor representation where all dimensions except the one corresponding to the *target variable* are compressed. While employing t-SVD to undertake the multi-stage tensor factorizations, it is imperative to appropriately permute the dimensions of the *intermediate* reduced tensor representations, allowing us to *attack* all relevant dimensions, leading to a compression of information along them. Moreover, for  $\mathcal{S}$  and all subsequent *intermediate* reduced tensor representations, it is important to maintain a specific tensor orientation. We will provide further details about these intricacies in our talk.

We demonstrate an application of the MSTR strategy in the context of reduced-order modeling by using it to extract the *final* reduced tensor representation  $\mathcal{A}_{n_s}$  for any given  $\mathcal{S}$ , along with the truncated orthogonal tensors  $\{\tilde{\mathcal{L}}_i\}_{i=1}^{n_s}$  from  $n_s$  tensor reduction stages. The primary motivation is to utilize the order-preserving compressed version of  $\mathcal{S}$ , enabling efficient operations within our reduced-order model, which can then deliver reliable predictions during the online phase at previously unseen parameter and/or time locations. After carrying out the MSTR procedure, we interpolate/regress between specific slices of  $\mathcal{A}_{n_s}$ , generating a map  $\mathcal{M}_{\text{tv}}$  capable of accurately rendering the *final* reduced tensor representation corresponding to new locations of the *target variable*, i.e., either parameter or time. We denote this approximation as  $\hat{\mathcal{A}}_{n_s}$ , which is obtained in the online phase. Note that the subscript **tv** in  $\mathcal{M}_{\text{tv}}$  refers to the *target variable*. Using the truncated orthogonal tensors  $\{\tilde{\mathcal{L}}_i\}_{i=1}^{n_s}$  and  $\hat{\mathcal{A}}_{n_s}$ , we obtain an approximation of the solution tensor at new locations of the *target variable*.  $\mathcal{M}_{\text{tv}}$  is built using a kernel-based shallow neural network (KSNN) with trainable kernel activation functions, where the parameters—kernel widths and center locations—are automatically determined via an alternating dual-staged iterative training procedure from our prior work [4].

In the *final* reduced tensor representation, the variable staying uncompressed is viewed as the *target variable*, whereas the variable compressed in the final stage of the MSTR procedure is referred to as the *secondary target variable*. To build a reduced-order model capable of providing predictions at new locations of both the parameter and time, it is necessary to ensure that they correspond to either the *target variable* or the *secondary target variable*. Upon ascertaining this, another interpolation/regression map  $\mathcal{M}_{\text{basis}}$  is created, which can provide an approximation of the truncated orthogonal tensor appearing in the second-last stage of the MSTR procedure, i.e.,  $\tilde{\mathcal{L}}_{n_s-1}$ , at new locations of the *secondary target variable*. The approximation from  $\mathcal{M}_{\text{basis}}$  is denoted as  $\hat{\tilde{\mathcal{L}}}_{n_s-1}$ .

For this variant of our reduced-order model, the online phase involves querying  $\mathcal{M}_{\text{tv}}$  to obtain  $\hat{\mathcal{A}}_{n_s}$ . This is then used, along with  $\tilde{\mathcal{L}}_{n_s}$ , to construct  $\hat{\mathcal{A}}_{n_s-1}$ . Later, yet another map  $\mathcal{M}_{\text{stv}}$  is constructed

by interpolation/regression between specific slices of  $\hat{\mathcal{A}}_{n_s-1}$ , capable of providing an approximation of the *intermediate* reduced tensor representation from the second-last stage, corresponding to new locations of the *secondary target variable*. We denote this approximation by  $\hat{\mathcal{A}}_{n_s-1}$ . Next,  $\hat{\mathcal{L}}_{n_s-1}$  is obtained by querying  $\mathcal{M}_{\text{basis}}$ , and in conjugation with  $\hat{\mathcal{A}}_{n_s-1}$ , an approximation of the *intermediate* reduced tensor representation from the third-last stage is constructed. By using this approximation in conjugation with the truncated orthogonal tensors  $\{\hat{\mathcal{L}}_i\}_{i=1}^{n_s-2}$ , we obtain the approximation of the high-dimensional solution tensor at new locations of the *target variable* and *secondary target variable*, i.e., parameter and time. We create  $\mathcal{M}_{\text{basis}}$  and  $\mathcal{M}_{\text{stv}}$  using KSNNs, which is especially useful for efficiently constructing  $\mathcal{M}_{\text{stv}}$  during the online phase. Moreover, an accurate construction of  $\mathcal{M}_{\text{basis}}$  is non-trivial, requiring interpolation over a Grassmann manifold. We investigate several approaches to accomplish this.

To train the MSTR-based reduced-order model, we need the solution tensor  $\mathcal{S} \in \mathbb{R}^{N_x \times N_y \times N_z \times m \times N_t}$ , which requires data from either physical measurements or high-fidelity simulations across  $N_t$  time instances and  $m$  parameter configurations. This can be challenging if obtaining spatio-temporal solution fields for many parameters is infeasible or computationally expensive. To address this, we develop a method to progressively expand the solution tensor  $\mathcal{S}$  along the parameter dimension from a small initial value  $m_0$  to a moderate final value  $m_{final}$ . This incremental-learning procedure involves iterative applications of the MSTR strategy to create a surrogate  $\hat{\mathcal{S}} \in \mathbb{R}^{N_x \times N_y \times N_z \times m_{fine} \times N_t}$ , where the growth of  $\mathcal{S}$  is guided by iteratively applying our t-EIM [5] over cheaply computable  $\hat{\mathcal{S}}$  to extract critical parameter locations from a fine candidate set with cardinality  $m_{fine}$ . Moreover, employing randomized t-SVD for all factorizations in the MSTR procedure further enhances efficiency during the offline phase. Related to our t-EIM are the recently proposed tensor discrete empirical interpolation methods [1, 2] that use t-SVD to get the interpolation basis. In [1], a greedy procedure is used to pick the interpolation indices, while [2] uses the pivoted t-QR decomposition [3]. In contrast, t-EIM employs a greedy procedure to select both the interpolation indices and the basis.

We have observed excellent performance of the proposed framework over numerous high-dimensional tensor-valued datasets, comprising climate measurements as well as various parametric spatio-temporal flow phenomena exhibiting rich dynamics, including convection-dominated behavior. In the talk, we primarily intend to highlight the theoretical and algorithmic contributions of our work. Furthermore, we will illustrate the robustness of the MSTR strategy and the reduced-order model based on it over an appropriately selected numerical example.

**Numerical results:** Table 1 provides the configurations of selected order-3 and order-4 tensor datasets, detailing the training set dimensions, representing about half of the parameter samples and time steps; the rest form the test sets. Table 3 lists the average relative errors for tensor approximations using the MSTR procedure and the MSTR-based reduced-order model. To demonstrate the MSTR procedure's advantage, Table 3 also includes average relative errors when the tensor is matricized to  $S \in \mathbb{R}^{N_x N_y \times m N_t}$ , thereby obtaining the truncated left singular matrix in  $\mathbb{R}^{N_x N_y \times r}$  via

Tensor datasets	Spatial dimension(s)	# parameter samples	# time steps
Wave equation	$N = 10201$	$m = 19$	$N_t = 401$
Burgers' equation	$N_x = 161, N_y = 161$	$m = 34$	$N_t = 101$
Navier-Stokes equations	$N = 37514$	$m = 18$	$N_t = 201$

Table 1: Details about the dimensions of selected order-3 and order-4 tensor datasets. All the examples have a 2D spatial domain with  $N$  denoting the total number of unstructured grid nodes.

Tensor datasets	SVD		MSTR		% drop
	Matricized	Compressed	Original	Compressed	
Wave equation	$\mathbb{R}^{10201 \times 7619}$	$\mathbb{R}^{10 \times 7619}$	$\mathbb{R}^{10201 \times 19 \times 401}$	$\mathbb{R}^{10 \times 19 \times 10}$	97.51%
Burgers' equation	$\mathbb{R}^{25921 \times 3434}$	$\mathbb{R}^{60 \times 3434}$	$\mathbb{R}^{161 \times 34 \times 161 \times 101}$	$\mathbb{R}^{10 \times 34 \times 10 \times 10}$	83.49%
Navier-Stokes equations	$\mathbb{R}^{37514 \times 3618}$	$\mathbb{R}^{10 \times 3618}$	$\mathbb{R}^{37514 \times 18 \times 201}$	$\mathbb{R}^{10 \times 18 \times 10}$	95.02%

Table 2: Details about the achieved level of compression for SVD and MSTR. The last column highlights % reduction in the entries of the *final* reduced tensor representation from MSTR in comparison with the compression achieved via SVD. The corresponding errors are reported in Table 3.

Tensor datasets	SVD	SVD-based ROM	MSTR	MSTR-based ROM
Wave equation	$3.91 \times 10^{-3}$	$4.08 \times 10^{-3}$	$6.98 \times 10^{-4}$	$1.21 \times 10^{-3}$
Burgers' equation	$1.14 \times 10^{-2}$	$8.61 \times 10^{-2}$	$5.45 \times 10^{-3}$	$6.14 \times 10^{-3}$
Navier-Stokes equations	$2.25 \times 10^{-2}$	$2.26 \times 10^{-2}$	$4.29 \times 10^{-4}$	$6.38 \times 10^{-4}$

Table 3: An illustrative comparison between average relative errors of the SVD, MSTR, and their respective reduced-order model (ROM) approximations over the test sets for selected datasets.

its SVD, projecting the matricized tensor on it, and producing the compressed representation in  $\mathbb{R}^{r \times mN_t}$ , where  $r \ll mN_t$ . The comparison between the results from SVD and MSTR is for equivalent compressions of the spatial dimensions. Table 2 details the compression levels, showing that the representation from MSTR possesses fewer total entries than the representation from SVD.

## References

- [1] S. Ahmadi-Asl, A.-H. Phan, C. F. Caiafa, and A. Cichocki. Robust low tubal rank tensor recovery using discrete empirical interpolation method with optimized slice/feature selection. *Advances in Computational Mathematics*, 50(2):23, 2024. doi:10.1007/s10444-024-10117-8.
- [2] S. Chellappa, L. Feng, and P. Benner. Discrete empirical interpolation in the tensor t-product framework. e-prints 2410.14519v1, arXiv, 2024. doi:10.48550/arXiv.2410.14519.
- [3] N. Hao, M. E. Kilmer, K. Braman, and R. C. Hoover. Facial recognition using tensor-tensor decompositions. *SIAM Journal on Imaging Sciences*, 435(1):437–463, 2013. doi:10.1137/110842570.
- [4] H. Kapadia, L. Feng, and P. Benner. Active-learning-driven surrogate modeling for efficient simulation of parametric nonlinear systems. *Computer Methods in Applied Mechanics and Engineering*, 419:116657, 2024. doi:10.1016/j.cma.2023.116657.
- [5] H. Kapadia, L. Feng, and P. Benner. Data-driven optimal sensor placement for tensorial data reconstruction via the tensor t-product framework. Technical report, 2024. In preparation.
- [6] M. E. Kilmer, K. Braman, N. Hao, and R. C. Hoover. Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging. *SIAM Journal on Matrix Analysis and Applications*, 34(1):148–172, 2013. doi:10.1137/110837711.
- [7] M. E. Kilmer and C. D. Martin. Factorization strategies for third-order tensors. *Linear Algebra and its Application*, 435(3):641–658, 2011. Special issue: Dedication to Pete Stewart on the occasion of his 70th birthday. doi:10.1016/j.laa.2010.09.020.
- [8] C. D. Martin, R. Shafer, and B. LaRue. An order- $p$  tensor factorization with applications in imaging. *SIAM Journal on Scientific Computing*, 35(1):A474–A490, 2013. doi:10.1137/110841229.
- [9] J. Zhang, A. K. Saibaba, M. E. Kilmer, and S. Aeron. A randomized tensor singular value decomposition based on the t-product. *Numerical Linear Algebra with Applications*, 25(5):e2179, 2018. doi:10.1002/nla.2179.

# QCLAB: A MATLAB Toolbox for Quantum Numerical Linear Algebra

Sophia Keip, Daan Camps, Roel Van Beeumen

## Abstract

Quantum numerical linear algebra is about solving numerical linear algebra problems on quantum computers - a field that has seen exciting and significant progress in the past few years. Rapid advancements in quantum hardware continue to drive this momentum forward and highlight the fast-paced progress of the field. To facilitate quantum algorithm research, especially as quantum hardware is still maturing, access to robust computational tools is crucial. We present QCLAB (<https://github.com/QuantumComputingLab/qclab>) [1], an object-oriented MATLAB toolbox for creating, representing and simulating quantum circuits. What sets QCLAB apart is its emphasis on numerical linear algebra, prioritizing numerical stability, efficiency, and performance. This dedication to robust numerical techniques underlies its role as the foundational framework for a range of derived software packages and quantum compilers.

In this talk, featuring a MATLAB tutorial on QCLAB, we will not only showcasing the key features of QCLAB, but also providing an overview of the state of the art in quantum numerical linear algebra research. To offer concrete insights, the presentation will focus on three landmark quantum algorithms: the Quantum Fourier Transform (QFT) [2], Quantum Phase Estimation (QPE) [4], and the Quantum Singular Value Estimation (QSVE) [3]. By demonstrating QCLAB along the way and introducing fundamental concepts in quantum computing, the audience will be encouraged to engage actively with this promising research area.

The QFT is a quantum version of the discrete Fourier transform forming the foundation for numerous quantum algorithms in quantum numerical linear algebra. Building on the QFT, QPE is a principal quantum algorithm used to determine the eigenvalue (or phase) corresponding to an eigenvector of a unitary operator. In simple terms, it estimates the phase  $\theta$  in the equation:

$$U|\psi\rangle = e^{2\pi i \theta}|\psi\rangle,$$

where  $U$  is a unitary operator and  $|\psi\rangle$  is an eigenvector of  $U$ . QPE is essential for applications like factoring large numbers (as in Shor's algorithm) and finding eigenvalues in quantum simulations. Finally, the QSVE extends these principles to non-unitary matrices, allowing singular values to be estimated directly through quantum methods. Both, QPE and QSVE promise polynomial complexity in  $n$  when applied to matrices of size  $2^n$ .

To better understand how these quantum algorithms function, we will begin with the underlying principles of quantum computation, highlighting its accessibility for researchers with a linear algebra background [5, 6]. A quantum computation involves the following three key components:

- **Quantum State:** The representation of information, a unit vector in a complex vector space.
- **State Evolution:** The transformation of the quantum state via unitary operators.
- **Measurement:** The process of extracting information from the quantum state.

A common way of representing those three components is a so-called *quantum circuit*. A Quantum circuit is an intuitive visual way to track the evolution and measurement of a quantum state. QCLAB, based on this circuit model, offers a user-friendly interface for constructing, simulating, and visualizing quantum circuits, in line with most modern quantum hardware platforms.

**Quantum states and qubits:** A quantum circuit acts on a register of *qubits*, which hold quantum information. Qubits are the quantum counterpart to classical bits. While a classical bit is either 0 or 1, a qubit can exist in a linear combination, called *superposition*, of two *basis states*  $|0\rangle$  and  $|1\rangle$ . To represent a state  $|\phi\rangle$  of a single qubit as vector in  $\mathbb{C}^2$ , we choose the standard basis  $|0\rangle = [1, 0]^T$ ,  $|1\rangle = [0, 1]^T$ . This leads to

$$|\phi\rangle = \alpha|0\rangle + \beta|1\rangle = \alpha \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \beta \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha \\ \beta \end{bmatrix},$$

where  $\alpha$  and  $\beta$  are complex numbers, called *amplitudes*, and  $|\alpha|^2 + |\beta|^2 = 1$ . For multiple qubits, the state space grows exponentially. Consequently, the state of a  $n$ -qubit register corresponds to a vector in  $\mathbb{C}^{2^n}$  and is described by a linear combination of  $2^n$  basis states. These basis states are formed as tensor products of the 1-qubit basis states, i.e.,  $|b_1 b_2 \cdots b_n\rangle = |b_1\rangle \otimes |b_2\rangle \otimes \cdots \otimes |b_n\rangle$  with  $b_i \in \{0, 1\}$ . Note that  $b_1 b_2 \cdots b_n$  are bit strings that can be interpreted as the binary representation of the integers from 0 to  $2^n - 1$ . The  $n$ -qubit state can thus be represented as  $\sum_{b=0}^{2^n-1} \alpha_b |b\rangle$ , with normalization  $\sum_{b=0}^{2^n-1} |\alpha_b|^2 = 1$ . For instance, a 2-qubit register has the four basis states  $|00\rangle$ ,  $|01\rangle$ ,  $|10\rangle$ ,  $|11\rangle$  with  $|00\rangle = |0\rangle \otimes |0\rangle = [1, 0, 0, 0]^T$  and analogous expressions for the rest.

To set up a 2-qubit quantum circuit in QCLAB, we use the following code:

```
>> circuit = qclab.QCircuit(2);
          q0 —
          q1 —
```

where the circuit diagram on the right represents the empty qubits  $q_0$  and  $q_1$ .

**State evolution:** Once we have prepared an  $n$ -qubit register in a certain state  $|\psi\rangle \in \mathbb{C}^{2^n}$ , we can evolve it over time. To ensure that the quantum state remains normalized, this evolution is achieved by applying unitary transformations  $|\psi\rangle \rightarrow |\psi'\rangle$ , known as quantum gates, which are represented by unitary matrices  $U \in \mathbb{C}^{2^n \times 2^n}$  and are depicted in circuit formulas as blocks:

$$|\psi'\rangle = U|\psi\rangle, \quad |\psi\rangle \xrightarrow{\boxed{U}} |\psi'\rangle$$

In theory, any unitary  $U$  can serve as quantum gate. However, in practice, the implementation of these gates is constrained by the underlying hardware, which determines which gates can actually be realized. An important 1-qubit gate is the Hadamard gate  $H$ , which transforms the basis states into equal superpositions, i.e. a linear combination with equal amplitudes.

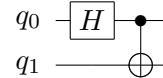
$$H|0\rangle = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle), \quad H|1\rangle = \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle).$$

A common 2-qubit gate is the controlled NOT gate, abbreviated to CNOT. This gate flips the state of the target qubit if and only the control qubit is in state  $|1\rangle$ . The unitary matrix representations of these two gates are

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad \text{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

QCLAB natively implements a wide variety of commonly used quantum gates as well as the option to implement your own gate based on its matrix representation. Going back to our example circuit, we can add a Hadamard gate to the first qubit (qubit 0) and a CNOT gate with control qubit 0 and target qubit 1 using the `push_back` function. On the right you see that our quantum circuit grows from left to right, which reflects the order the gates are applied to the state.

```
>> circuit.push_back(qclab.qgates.Hadamard(0));
>> circuit.push_back(qclab.qgates.CNOT(0,1));
```



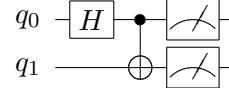
**Measurement:** In contrast to classical bits, it is not possible to observe the quantum state directly. Instead, we can only gain information through measurements, which inherently affect the state itself since they collapse the state to one of the basis states. In a 1-qubit state  $\alpha|0\rangle + \beta|1\rangle$ , the probability of measuring 0 is  $|\alpha|^2$ , while the probability of measuring 1 is  $|\beta|^2$ . After measurement, the state collapses to either  $|0\rangle$  or  $|1\rangle$ , based on the observed outcome. For a general state  $|\psi\rangle = \sum_b \alpha_b |b\rangle$ , the probability of measuring a basis state  $|b\rangle$  is

$$P(b) = |\alpha_b|^2.$$

Since  $|\psi\rangle$  is normalized, this defines a valid probability distribution over the possible measurement outcomes. For instance, for a 2-qubit register in the state  $\alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle$ , the probability of measuring 0 for both qubits is  $|\alpha_{00}|^2$ , and the system collapses to  $|00\rangle$  after the measurement.

Let us add measurements to both qubits in our QCLAB example circuit:

```
>> circuit.push_back(qclab.Measurement(0));
>> circuit.push_back(qclab.Measurement(1));
```



**Simulating quantum circuits:** After constructing a circuit, the next step is to execute it and observe the results. Here, we set up a 2-qubit circuit consisting of a Hadamard gate on the first qubit, a CNOT gate with control qubit 0 and target qubit 1, and two measurements. Let us see what the circuit does on the initial state  $|00\rangle = |0\rangle \otimes |0\rangle$ . Applying the Hadamard gate to the first qubit yields

$$H|0\rangle \otimes |0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle.$$

Next, the CNOT gate flips the bit of the second qubit whenever the first qubit is 1, so

$$\text{CNOT} \left( \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle \right) = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle.$$

Finally, by measuring both qubits, we measure  $|00\rangle$  and  $|11\rangle$  both with probability  $|\frac{1}{\sqrt{2}}|^2 = 0.5$ . To simulate the circuit in QCLAB, we use the `simulate` function with the initial state as input:

```
>> simulation = circuit.simulate('00');
```

For all measurements we can get the possible measurement results together with the corresponding probabilities and collapsed states by

<code>&gt;&gt; simulation.results</code>	<code>&gt;&gt; simulation.probabilities</code>	<code>&gt;&gt; simulation.states</code>
<code>ans = 2x1 cell</code>	<code>ans = 2x1 cell</code>	<code>ans = 2x1 cell</code>
<code>'00'</code>	<code>0.5000</code>	<code>[1;0;0;0]</code>
<code>'11'</code>	<code>0.5000</code>	<code>[0;0;0;1]</code>

QCLAB provides a full state simulation, meaning it accurately represents the entire quantum state vector, allowing precise tracking of amplitudes and phase information for each qubit throughout the computation. The straightforward simulation of quantum circuits in QCLAB makes it a valuable

tool for rapid prototyping of quantum algorithms. This capability allows researchers to efficiently experiment with and refine their algorithms prior to moving on to more advanced stages.

**Additional features:** Other than the computational tasks, QCLAB enables the visualization of quantum circuits directly in the MATLAB command window and supports saving a circuit diagram to LaTeX source files, as demonstrated in the diagrams presented within this abstract. Both can be done using the following commands:

```
>> circuit.draw;
>> circuit.toTex;
```

This functionality makes it particularly useful for research documentation and presentations. QCLAB also provides input/output compatibility with openQASM, a low-level programming language used to describe quantum circuits, which is compatible with quantum hardware. This allows users to test their quantum circuits on real quantum computers and is achieved by the command:

```
>> circuit.toQASM;
```

Alongside the numerical linear algebra applications we present in this talk, QCLAB offers a variety of other examples that help users getting familiar with both quantum computing concepts and the toolbox itself. Additionally, extensive documentation is available to make the learning process as smooth as possible.

QCLAB also has a C++ counterpart, QCLAB++, [7, 8], which is designed for more computationally demanding tasks by leveraging GPU capabilities. QCLAB++ retains the same user-friendly syntax as QCLAB, allowing researchers to easily transition from prototyping in MATLAB to scaling up simulations with C++ on GPUs.

This talk is designed for both researchers in numerical linear algebra seeking an easy entry point into quantum computing, and experienced quantum computing practitioners looking for a tool to facilitate rapid prototyping of quantum algorithms.

## References

- [1] D. Camps and R. Van Beeumen. “QCLAB v0.1.2,” Aug. 2021. doi:10.5281/zenodo.5160555. [github.com/QuantumComputingLab/qclab](https://github.com/QuantumComputingLab/qclab)
- [2] D. Coppersmith. “An approximate Fourier transform useful in quantum factoring.” IBM Research Report RC 19642, (1994).
- [3] A. Gilyén, Y. Su, G. H. Low and N. Wiebe. “Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics.” In Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing (pp. 193-204), (2019).
- [4] A. W. Harrow, A. Hassidim and S. Lloyd. “Quantum algorithm for linear systems of equations.” Physical review letters, 103(15), (2009).
- [5] G. Nannicini. “An introduction to quantum computing, without the physics.” SIAM Review 62.4: 936-981, (2020).
- [6] M. A. Nielsen and I. L. Chuang. “Quantum computation and quantum information,” Vol. 2. Cambridge: Cambridge University Press, (2001).

- [7] R. Van Beeumen and D. Camps. “QCLAB++ v0.1.2,” Aug. 2021. doi:10.5281/zenodo.5160682. [github.com/QuantumComputingLab/qclabpp](https://github.com/QuantumComputingLab/qclabpp)
- [8] R. Van Beeumen, D. Camps, and N. Mehta. “QCLAB++: Simulating Quantum Circuits on GPUs,” arXiv:2303.00123 (2023).

# A Memory-efficient MM-GKS Variant for Large-scale Dynamic or Streaming Inverse Problems

Misha E. Kilmer, Mirjeta Pasha, Eric de Sturler

## Abstract

Reconstructing high-quality images with sharp edges requires edge-preserving regularization operators. Using a general  $\ell_q$ -norm on the gradient of the image is a common approach. For implementation purposes, the  $\ell_q$ -norm regularization term is typically replaced with a sequence of  $\ell_2$ -norm weighted gradient terms with the weights determined from the current solution estimate. While (hybrid) Krylov subspace methods can be employed on this sequence, it would require generating a new Krylov subspace for every update of the regularization operator. The majorization-minimization generalized Krylov subspace method (MM-GKS) addresses this disadvantage by combining the updating of the regularization operator with generalized Krylov subspaces (GKS). After projecting the problem onto a lower dimensional subspace - one that expands each iteration - the regularization parameter is selected for the projected problem. Basis expansion continues until a sufficiently accurate solution is found. Unfortunately, for large-scale problems that require many iterations to converge, storage and the cost of repeated orthogonalization present overwhelming memory requirements and computational costs.

We present a variant of MM-GKS that provably converges to the minimum of the smoothed functional even if the search space dimension remains very small. This substantially improves theoretical results for MM-GKS where the convergence proof relies on (eventually) spanning the full problem space. Using this result, we develop a new method that solves the minimization/imaging problem by alternatingly compressing and expanding the search space while maintaining strict monotonic convergence. Our method can solve large-scale problems efficiently both in terms of computational complexity and memory requirements. In the compression phase, we select a subspace of small dimension that is considered the most “important” for convergence by one of four compression strategies. We further generalize our proposed method to handle *streaming problems* where the data is either not all available simultaneously or the size of the problem demands it be treated as such. We demonstrate the utility of our approach on several image reconstruction and restoration problems.

# Randomized solvers for joint eigenvalue problems

Daniel Kressner, Haoze He

## Abstract

Here is a Matlab one-liner for computing the eigenvectors of a *normal* matrix  $A$ :

```
H = A+A'; S = A-A'; [U,~] = eig(randn*H+randn*1i*S);
```

This talk will explain why this one-liner works and what accuracy we can expect.

More generally, this talk is concerned with randomized methods for solving joint eigenvalue problems associated with a matrix family  $A_1, \dots, A_d$ . This problem class includes the diagonalization of (nearly) commuting symmetric or nonsymmetric matrices, as well as simultaneous diagonalization by congruence. As in the Matlab one-liner above (which exploits that the Hermitian and skew-Hermitian parts of a normal matrix commute), a common idea of these randomized methods is to first reduce the matrix family to one or two matrices by random linear combinations and then apply a standard eigensolver. These methods are remarkably simple and robust, and provide a decent level of accuracy with very high probability. If needed, accuracy can be improved further with optimization techniques or other refinement strategies.

Besides algorithms, we will also discuss the theory and applications of randomized solvers for joint eigenvalue problems. It turns out that classical eigenvalue perturbation theory, with a small dose of probabilistic analysis, can explain much of the success of these solvers. Applications include signal processing tasks (e.g., for image separation and EEG analysis) and tensor decomposition (e.g., for learning latent variable models). A particularly successful and important application are joint eigenvalue methods for multivariate root-finding problems, which we have explored in joint work with Bor Plestenjak.

## References

- [1] H. He and D. Kressner. Randomized joint diagonalization of symmetric matrices. *SIAM J. Matrix Anal. Appl.*, 45(1):661–684, 2024.
- [2] H. He and D. Kressner. A randomized algorithm for simultaneously diagonalizing symmetric matrices by congruence. arXiv:2402.16557, 2024.
- [3] H. He and D. Kressner. A simple, randomized algorithm for diagonalizing normal matrices. arXiv:2405.18399, 2024.
- [4] H. He, D. Kressner, and B. Plestenjak. Randomized methods for computing joint eigenvalues, with applications to multiparameter eigenvalue problems and root finding arXiv:2409.00500, 2024.

# Most matrix manifold optimization problems are NP-hard

Zehua Lai, Lek-Heng Lim, Ke Ye

## Abstract

Some of the most common Riemannian manifolds in geometry, including many that are important in engineering applications, may be represented as matrix manifolds, i.e., submanifolds or quotient manifolds of  $\mathbb{R}^{m \times n}$  endowed with various Riemannian metrics.

We consider four of the best known ones: The Cartan manifold of ellipsoids in  $\mathbb{R}^n$  may be modeled as the set of positive definite matrices

$$E(\mathbb{R}^n) \cong \{A \in \mathbb{R}^{n \times n} : A^\top = A, x^\top A x > 0 \text{ for all } x \neq 0\}. \quad (1)$$

The compact Stiefel manifold of orthogonal  $k$ -frames in  $\mathbb{R}^n$  may be modeled as the set of  $n \times k$  orthonormal matrices

$$V_k(\mathbb{R}^n) \cong \{Y \in \mathbb{R}^{n \times k} : Y^\top Y = I\} \quad (2)$$

The noncompact Stiefel manifold of  $k$ -frames in  $\mathbb{R}^n$  may be modeled as the set of  $n \times k$  full-rank matrices

$$St_k(\mathbb{R}^n) \cong \{S \in \mathbb{R}^{n \times k} : \text{rank}(S) = k\} \quad (3)$$

The Grassmannian of  $k$ -planes in  $\mathbb{R}^n$  may be modeled either as projection matrices, involution matrices, or, more generally, quadratic matrices with appropriate trace values:

$$\begin{aligned} Gr_k(\mathbb{R}^n) &\cong \{P \in \mathbb{R}^{n \times n} : P^2 = P = P^\top, \text{tr}(P) = k\} \\ &\cong \{Q \in \mathbb{R}^{n \times n} : Q^\top Q = I, Q^\top = Q, \text{tr}(Q) = 2k - n\} \\ &\cong \{W \in \mathbb{R}^{n \times n} : W^\top = W, (W - aI)(W - bI) = 0, \text{tr}(W) = ka + (n - k)b\}. \end{aligned} \quad (4)$$

Further examples may be obtained from these four basic cases as product, quotient, or submanifolds.

We will show that unconstrained quadratic optimization over any of these models of Grassmannian is NP-hard. Our results cover all scenarios: (i) when  $k$  and  $n$  are both allowed to grow; (ii) when  $k$  is arbitrary but fixed; (iii) when  $k$  is fixed at its lowest possible value of 1.

For example, the clique decision problem, i.e., deciding if a clique of size  $k$  exists in a graph  $G = (V, E)$ , may be formulated as the maximization problem

$$\max_{P \in Gr(k,n)} \left[ \sum_{(i,j) \in E} e_i^\top P e_i e_j^\top P e_j + \sum_{i \in V} e_i^\top P e_i e_i^\top P e_i \right], \quad (5)$$

where  $Gr(k,n)$  is the projection model of Grassmannian in (4) and  $e_1, \dots, e_n \in \mathbb{R}^n$  the standard basis. This establishes NP-hardness of (i) for the projection model but we will extend it to other models and also to (ii) and (iii).

We will establish similar NP-hardness results for unconstrained quadratic optimization over the Cartan manifold in (1), as well as unconstrained cubic optimization over the compact and noncompact Stiefel manifolds in (2) and (3).

In all cases, we will rule out the existence of FPTAS and show that these hardness results hold regardless of the choice of Riemannian metrics one puts on these manifolds. If time permits, we

will also discuss the NP-hardness of optimizing over various representations of these manifolds as quotient matrix manifolds, including

$$\begin{aligned}\mathrm{Gr}_k(\mathbb{R}^n) &\cong \mathrm{O}(n)/(\mathrm{O}(k) \times \mathrm{O}(n-k)), & \mathrm{V}_k(\mathbb{R}^n) &\cong \mathrm{O}(n)/\mathrm{O}(n-k), \\ \mathrm{St}_k(\mathbb{R}^n) &\cong \mathrm{GL}(n)/\mathrm{P}_1(k,n), & \mathrm{E}(\mathbb{R}^n) &\cong \mathrm{GL}(n)/\mathrm{O}(n).\end{aligned}$$

## References

- [1] Z. Lai, L.-H. Lim, and K. Ye, “Grassmannian optimization is NP-hard,” [arXiv:2406.19377](https://arxiv.org/abs/2406.19377), 2024.

# Randomized low-rank Runge-Kutta methods

*Hei Yin Lam, Gianluca Ceruti, Daniel Kressner*

## Abstract

In this work, we aim at approximating the solution  $A(t)$  to large-scale matrix differential equations of the form

$$\dot{A}(t) = F(A(t)), \quad A(0) = A_0 \in \mathbb{R}^{m \times n}. \quad (1)$$

For large  $m$  and  $n$ , the solution of (1) becomes expensive; in fact, it may not even be possible to store the entire matrix  $A(t)$  explicitly. To circumvent this limitation, model order reduction techniques based on exploiting (approximate) low-rank structure of  $A(t)$  can be employed. In particular, dynamical low-rank approximation [3] approximates  $A(t)$  by evolving matrices  $Y(t)$  on the manifold  $\mathcal{M}_r$  of rank- $r$  matrices, reducing memory usage when  $r \ll m, n$ . By the Dirac-Frenkel variational principle, the matrix  $Y(t)$  is obtained by solving the differential equation

$$\dot{Y}(t) = P_r(Y(t))F(Y(t)), \quad Y(0) = Y_0 \in \mathcal{M}_r, \quad (2)$$

where  $P_r(Y(t))$  denotes the orthogonal projection onto  $T_{Y(t)}\mathcal{M}_r$ , the tangent space of  $\mathcal{M}_r$  at  $Y(t)$ . However, the stiffness of this equation leads to a severe step size restriction for standard explicit time integration methods. To address this issue, special integrators for this equation have been proposed [4, 2, 5]. Under the assumption

$$\|F(Y) - P_r(Y)F(Y)\|_F \leq \tilde{\epsilon}, \quad \text{for all } Y \in \mathcal{M}_r \cap \{\text{suitable neighbourhood of } A(t)\} \quad (3)$$

all these methods exhibit at least first-order convergence up to  $\mathcal{O}(\tilde{\epsilon})$ .

Assumption (3), which states that  $F(Y)$  is nearly contained in the tangent space, is arguably a strong assumption. According to [2] and the examples shown, it is possible that  $A(t)$  can be well approximated by a rank- $r$  matrix even if (3) is not satisfied with small  $\tilde{\epsilon}$ . When this assumption fails for small  $\tilde{\epsilon}$ , using tangent space projections in numerical methods risks introducing significant errors.

In this work, we develop low-rank time integration methods for (1) that do not rely on (3) but only require  $A(t)$  to admit accurate low-rank approximations. Our approach is based on the notion of projected integrators, which first perform a standard time integration step and then project back to the manifold. For the manifold  $\mathcal{M}_r$ , the efficiency of projected integrators is impaired by the occurrence of high-rank matrices, e.g., during the intermediate stages of a Runge-Kutta method. Previous work [2] mitigated this with repeated tangent space projections. Here, we propose a novel alternative using randomized low-rank approximation, employing random sketches instead of tangent projections to control rank growth efficiently.

To the best of our knowledge, this is the first work to propose and analyze randomized low-rank approximation methods for time integration. The randomized low-rank Runge-Kutta (RK) methods proposed in this work combine explicit RK methods with randomized low-rank approximation. Assuming that the dynamics generated by  $F$  preserve rank- $r$  matrices approximately, we derive a probabilistic result that establishes a convergence order (up to the level of rank- $r$  approximation error) based on the so-called stage order of the underlying RK method, which matches the order established in [2] for projected RK methods. However, unlike the results in [2], our numerical experiments indicate that randomized low-rank RK methods actually achieve the usual convergence order of the RK method, which can be significantly higher. For the randomized low-rank RK 4, we also establish order 4 theoretically when allowing for modest intermediate rank increases in the stages. This compares favorably to order 2 implied by the techniques in [2].

## References

- [1] Hei Yin Lam, Gianluca Ceruti, and Daniel Kressner. Randomized low-rank Runge-Kutta methods. *arXiv preprint arXiv:2409.06384*, 2024.
- [2] Emil Kieri and Bart Vandereycken. Projection methods for dynamical low-rank approximation of high-dimensional problems. *Comput. Methods Appl. Math.*, 19(1):73–92, 2019.
- [3] Othmar Koch and Christian Lubich. Dynamical low-rank approximation. *SIAM J. Matrix Anal. Appl.*, 29(2):434–454, 2007.
- [4] Christian Lubich and Ivan V. Oseledets. A projector-splitting integrator for dynamical low-rank approximation. *BIT*, 54(1):171–188, 2014.
- [5] Gianluca Ceruti, Lukas Einkemmer, Jonas Kusch, and Christian Lubich. A robust second-order low-rank BUG integrator based on the midpoint rule. *BIT*, 64(3):Paper No. 30, 2024.

# Inner-product free Krylov subspace methods for inverse problems

*Malena Sabaté Landman, Ariana Brown, Julianne Chung, James G. Nagy*

## Abstract

We consider linear discrete inverse problems, which involve the reconstruction of hidden objects from possibly noisy indirect measurements. Rapid advances in technology and computation have resulted in enormous and growing data-sets, creating an urgent need for more efficient algorithms that can handle the increasing dimension of these problems while maintaining both reliability and, crucially, speed. New (and not so new) promising directions include reducing the working floating point arithmetic and increase parallelization, both of which can suffer from problems related to the use of inner-products in the algorithms. In this work, I present a **general class of Krylov methods** that are inherently **inner-product free**, while maintaining regularizing properties, making them a powerful and efficient alternative to traditional Krylov solvers in the context of **large-scale linear inverse problems**. Important applications appear, for example, in medical imaging (computed tomography), non-destructive testing of engineering designs, and geophysics (seismic exploration).

We write a linear system with additive noise as

$$Ax + e = b, \quad A \in \mathbf{R}^{m \times n}, \quad (1)$$

where we typically consider  $e$  to be a realization of a white Gaussian random variable, and we focus on problems that are ill-posed in the sense that the system matrix  $A$  has decaying singular values which cluster at zero, but where  $A$  has an ill-determined numerical rank. Therefore, the reconstructions of  $x$  are typically very sensitive to perturbations in the measurements (a.k.a. noise) and need regularization.

Krylov subspace methods are a class of very popular projection methods to solve (1) which show very fast convergence and have inherent regularization properties when equipped with early stopping of the iterations [1]. In other words, the approximate solutions approach the true solution in the first iterations but, if the algorithms are not stopped, they continue to converge towards to unwanted solution of the least-squares problems associated with the noisy right hand side, which suffers badly from noise amplification: this is also referred to as semi-convergence. These are typically based on the stable construction of bases for Krylov subspaces, or search spaces for the solution  $x$ , defined as:

$$\mathcal{K}_k(C, d) = \mathcal{R}(V_k), \quad V_k = [d, Cd, \dots, C^{k-1}d], \quad (2)$$

which are related to the original linear system and where  $\mathcal{R}(\cdot)$  represents the range of a matrix. Moreover, these are iterative methods, so that a new direction is added to the space  $\mathcal{K}_k(C, d)$  at each iteration  $k$ . Usually, this is done using either the Arnoldi or Golub-Kahan bidiagonalization (GKB) algorithms, which, applied to (1), differ on the choice of the matrix-vector pair  $\{C, d\}$ . In particular, Arnoldi considers  $\{A, b\}$ ; and GKB constructs basis for two Krylov subspaces; one taking  $\{A^T A, A^T b\}$  and one taking  $\{A A^T, b\}$ . Both methods rely on the implicit construction of a QR factorization of the matrix  $V_k$  in (2), and they require the orthogonalization and normalization of the new basis vectors against the previous vectors in the basis, see, e.g. [2, Chapter 4]. However, there are scenarios where the inner-products required in this process can hinder the usability of the solvers. For example, in low precision arithmetic, standard Krylov solvers might break-down too early due to the norm of the new vector after orthogonalization being numerically zero in the given working precision, or over/under-flows can occur during norm computations. Moreover,

inner-products can be a limiting factor for high performance computing, since they require global communication [3]. On the other hand, the most used inner-product free solvers, e.g. Chebyshev semi-iterations, can show very slow convergence.

In this work, we present a general family of solvers which leverage the fast converge of Krylov methods while being inherently inner-product free, and which are based off implicit LU factorizations of the matrix  $V_k$  in (2). These solvers construct bases that span the same Krylov subspaces as those associated to their traditional counterparts, but contain only linearly independent vectors that are not orthogonal by construction. The choice of the matrix-vector pair  $\{C, d\}$  in (2) gives rise to different frameworks, which hold a strong parallelism to the Arnoldi and GKB algorithms: on the one hand, we use the standard Hessenberg method for  $\{A, b\}$ , see [4, 5], and on the other hand we develop a new modified version of the Hessenberg method for  $\{A^T A, A^T b\}$  and for  $\{A A^T, b\}$ , see [7]. Note that, in practice, partial pivoting is needed to avoid unwanted break-down of the algorithms. Moreover, we describe the (quasi minimal residual) solvers associated with each approach. In the first place, we revisit the changing minimal residual Hessenberg method (CMRH), see e.g. [4, 5], and then we develop a completely new method, which we call the least squares LU (LSLU) [7]. In particular, we show that CMRH and LSLU can be used to tackle large-scale linear inverse problems efficiently, since they both present very fast convergence and regularization properties. Note that previous work on the CMRH method did not consider its application to ill-posed problems, and it was therefore not known if it was a regularizing method. Now we know that both CMRH and LSLU have empirical spectral filtering properties, i.e., early iterations reconstruct smooth component of the solution, and later iterations reconstruct high frequency components, so that early stopping of the iterations gives rise to a regularized solution. Finally, we extend both methods to include Tikhonov regularization, in the fashion of hybrid regularization [8], so that a projected Tikhonov regularization problem is solved at each iteration. This is a very powerful framework, since the solution of such problems does not show semi-convergence (and therefore is less sensitive to the stopping criteria), and we can choose the regularization parameters throughout the iterations using standard parameter choice criteria.

For more details on this work, where theoretical results and extensive numerical experiments suggest that inner-product free variants exhibit comparable performance to established methods, see [6, 7].

## References

- [1] P. C. Hansen. *Discrete Inverse Problems: Insight and Algorithms*, SIAM (2010).
- [2] G. Meurant and J. Duintjer Tebbens, *Krylov Methods for Nonsymmetric Linear Systems: From Theory to Computations*, Springer Cham (2020).
- [3] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, SIAM (1994).
- [4] H. Sadok, *CMRH: A new method for solving nonsymmetric linear systems based on the Hessenberg reduction algorithm*, Numerical Algorithms, 20 (1999), pp.485-501.
- [5] H. Sadok and D. B. Szyld, *A new look at CMRH and its relation to GMRES.*, BIT Numer. Math., 52 (2012), pp. 485–501.

- [6] A. N. Brown, J. G. Nagy and M. Sabaté Landman, *H-CMRH: A Novel Inner Product Free Hybrid Krylov Method for Large-Scale Inverse Problems*, Accepted for publication in SIMAX, (2024), arXiv:2401.06918.
- [7] A. N. Brown, J. Chung, J. G. Nagy and M. Sabaté Landman, *Inner Product Free Krylov Methods for Large-Scale Inverse Problems*, arXiv preprint, (2024), arXiv: arXiv:2409.05239.
- [8] J. Chung and S. Gazzola, *Computational methods for large-scale inverse problems: a survey on hybrid projection methods*, SIAM Review, 66 (2024), pp. 205–284.

# New results on the I/O complexity of some Numerical Linear Algebra kernels

Julien Langou

## Abstract

When designing an algorithm, one cares about arithmetic/computational complexity, but data movement (I/O) complexity is playing an increasingly important role that highly impacts performance and energy consumption. The objective of I/O complexity analysis is to compute, for a given program, its minimal I/O requirement among all valid schedules. We consider a sequential execution model with two memories, an infinite one, and a small one of size  $S$  on which a computation unit retrieves and produces data. The I/O is the number of reads and writes between the two memories. From this model, we review various Numerical Linear Algebra kernels that are increasingly complicated from matrix-matrix multiplication, to LU factorization, then to symmetric rank-k update, to Cholesky factorization, then to Modified Gram-Schmidt to Householder QR factorization. We will show practical examples of these results too.

In particular, we will focus on two recent results.

First, we present the “*hourglass pattern*” which is useful in analysing algorithm such as, for example, Modified Gram-Schmidt or Householder QR factorization. We identify a common hourglass pattern in the dependency graphs of several common linear algebra kernels. Using the properties of this pattern, we mathematically prove tighter lower bounds on their I/O complexity, which improves the previous state-of-the-art bound by a parametric ratio. This proof was integrated inside the IOLB automatic lower bound derivation tool. These results were presented in [1]. In addition to lower bound results, we will show a tiling (valid for Modified Gram-Schmidt or Householder QR factorization) which enables to reach the lower bound. We present numerical experiments on modern platforms that shows the effectiveness of the new tiling.

Second, in [6], we focus on the problem of to apply a chain of sequences of Givens rotations to a matrix  $A$ . Applying a chain of Givens rotations efficiently is an important building tool in numerical linear algebra. Some examples are the implicit QR algorithm [2] and the Jacobi method for the singular value decomposition [3]. To achieve high performance, many factorizations limit their initial calculations to a smaller submatrix of the original matrix. Updating the rest of the matrix (which often involves the bulk of the floating-point operations) can then be done efficiently with an optimized routine. In practice, we observe that a vanilla algorithm performs poorly and is memory-bound. However this algorithm has a three-loop structure reminiscent of a Level 3 BLAS subroutine, and one would want to reorganize the operations to get a compute-bound algorithm. Kågström et al. [4] and later Van Zee et al. [5] demonstrated two ways to increase efficiency: wavefront pattern and fused rotations. We present a new algorithm that is innovative in three main ways. Firstly, we introduce a kernel that is optimized for register reuse in a novel way. Secondly, we introduce a blocking and packing scheme that improves the cache efficiency of the algorithm. Finally, we thoroughly analyze the memory operations of the algorithm which leads to important theoretical insights and makes it easier to select good parameters. Numerical experiments show that our algorithm outperforms the state-of-the-art and achieves a flop rate close to the theoretical peak on modern hardware. In addition to a practical new algorithm, we use our I/O lower bound theory to prove that our tiling is optimal in terms of I/O. A technical report explaining these new findings will be released soon.

1. Lionel Eyraud-Dubois, Guillaume Iooss, Julien Langou, and Fabrice Rastello. Tightening I/O lower bounds through the hourglass dependency pattern. In *the 36th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA '24)*, Nantes, France, June 17–21, 2024. DOI: 10.1145/3626183.3659986.
2. John GF Francis. The QR transformation a unitary analogue to the LR transformation—Part 1. *The Computer Journal*, 4(3):265–271, 1961. DOI: 10.1093/comjnl/4.3.265.
3. Carl Gustav Jacob Jacobi. Über ein leichtes verfahren die in der theorie der säcularstörungen vorkommenden gleichungen numerisch aufzulösen. *Journal für die reine und angewandte Mathematik*, 30:51-94, 1846. DOI: 10.1017/CBO9781139568012.016
4. Bo Kågström, Daniel Kressner, Enrique S. Quintana-Ortí, and Gregorio Quintana-Ortí. Blocked algorithms for the reduction to Hessenberg-triangular form revisited. *BIT Numerical Mathematics*, 48:563–584, 2008. DOI: 10.1007/s10543-008-0180-1
5. Field G. Van Zee, Robert A. van de Geijn and Gregorio Quintana-Ortí. Restructuring the tridiagonal and bidiagonal QR algorithms for performance. *ACM Transactions on Mathematical Software (TOMS)*, 40(3):1–34, 2014. DOI: 10.1145/2535371.
6. Thijs Steel and Julien Langou. Communication efficient application of chains of sequences of planar rotations to a matrix. Technical Report to be released late 2024 or early 2025.

# Optimal accuracy for linear sets of equations with the graph Laplacian

*Rich Lehoucq, Jon Berry, Danny Dunlavy, Natalie Wellen & Michael Weylandt*

## Abstract

An approximate solution  $\hat{x}$  for a linear set of equations  $Lx = b$  has roughly  $d$  digits of accuracy when  $\|\hat{x} - x\|/\|x\| \approx 10^{-d}$ . The classical two-sided inequality

$$\frac{1}{\kappa(L)} \frac{\|b - L\hat{x}\|}{\|b\|} \leq \frac{\|x - \hat{x}\|}{\|x\|} \leq \kappa(L) \frac{\|b - L\hat{x}\|}{\|b\|} \quad x, b \neq 0 \quad (1)$$

implies that the relative error is norm-equivalent to the relative residual error  $\|b - L\hat{x}\|/\|b\|$  with constants given by the condition number  $\kappa(L)$  and its reciprocal. For us, the matrix  $L$  is a graph Laplacian and the vector  $x$  represents a network centrality measure indicating the importance of the vertices, e.g., the PageRank [1] vector or the vector of mean hitting-times. Unfortunately, the condition number  $\kappa(L)$  increases with graph size or with the PageRank teleportation parameter rendering (1) useless in practice. We establish improved variants of the two-sided inequality and explore their profound computational implications.

We focus our analysis on the relationship between the relative error and the relative residual. This relationship is key to assessing the quality of  $\hat{x}$  because it relates the observable quantity  $\|L\hat{x} - b\|$  with the unobservable quantity  $\|\hat{x} - x\|$ . We show that the strength of this relationship is determined by the angle between  $b$  and the vector of all ones on an undirected graph. This relationship is also dependent on the so-called *Markov chain discount*, a classical concept that we find is equivalent to the PageRank teleportation parameter for undirected graphs. This provides an elegant probabilistic basis for the degree-normalized PageRank variant and the simple characterization of PageRank on an undirected graph sought by Gleich [2, p.356].

Our contributions are twofold: i) we establish a more informative variant of (1) using a *data-dependent condition number* and ii) we reframe graph centrality measures in the language of discrete potential theory and show how certain potentials can achieve asymptotically optimal accuracy. We discuss the application of these results to PageRank and conclude with numerical simulations highlighting the impact of our improved bounds. We refer the reader to the associated report [Optimal accuracy for linear sets of equations with the graph Laplacian](#) available at <https://arxiv.org/>.

## References

- [1] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Comput. Netw. ISDN Syst.*, 30:107–117, 1998.
- [2] David F. Gleich. PageRank beyond the web. *SIAM Review*, 57(3):321–363, 2015.

# Spectral Density Estimation of Kernel Matrices with Applications

Mikhail Lepilov

## Abstract

Kernel matrices formed from large sets of observations arise frequently in data science, for example during classification tasks. It is desirable to know the eigenvalue decay properties of these matrices without explicitly forming them, such as when determining if a low-rank approximation is feasible. In this talk, I will introduce a new spectral density framework based on quantile bounds. This framework gives meaningful bounds for all the eigenvalues of a kernel matrix while avoiding the cost of constructing the full matrix. The kernel matrices under consideration come from a kernel with quick decay away from the diagonal applied to uniformly-distributed sets of points in Euclidean space of any dimension. I will prove certain results enabling this framework whenever the kernel function satisfies a certain decay condition, and I will give empirical evidence for its accuracy. In the process, I will also prove a very general interlacing-type theorem for finite sets of numbers. Additionally, I will give an application of this framework to the study of the intrinsic dimension of data. In doing so, I introduce a new “local” notion of intrinsic dimension, which has the power to test a certain interpretation of the so-called “manifold hypothesis” for a given dataset.

# Fast Solvers for the Runge–Kutta Integration of the Instationary Incompressible Navier–Stokes Equations

*Santolo Leveque, Yunhui He, and Maxim Olshanskii*

## Abstract

Time-dependent PDEs arise very often in many scientific areas, such as mechanics, biology, economics, or chemistry, just to name a few. The lack of a closed form solution for general time-dependent PDEs requires one to employ numerical methods in order to find an approximation of it. These methods consider suitable discretizations of the quantity involved; in particular, they employ suitable time-stepping schemes as discretization of the time derivative. The majority of the solvers for time-dependent PDEs is based on classical linear multistep methods. Within this framework, the approximation of the solution at time  $t_n$  is evaluated as a linear combination of the  $s$  previous steps. The wide use of multistep methods is due to their simplicity; in fact, the structure of the discretized problem allows one to employ solvers for the corresponding stationary PDE as a solver for the time-dependent one. Despite this favourable quality, multistep methods have a drawback: they are (in general) not A-stable, a property that allows one to choose an arbitrary time-step for the integration. In fact, as stated by the second Dahlquist barrier, see for example [4, Theorem 6.6], an A-stable linear multistep method cannot have order of convergence greater than two. By contrast, one can devise an implicit Runge–Kutta method so that not only it is A-stable, but also such that the method possesses more desirable stability properties (e.g., L- or B-stability, see for instance [2, 3, 4]). However, the better stability properties of Runge–Kutta methods come to a price: the discretization results in a non-linear block system, to be solved for the so called *stages* of the method at each time step. For this reason, in recent years researchers have devoted their effort in devising efficient and robust linear solvers for the solution of block systems arising from the Runge–Kutta discretization of a time-dependent PDE, see for example [1, 5, 6, 7].

Consider the integration of an ODE of the form  $v'(t) = f(v(t), t)$  between 0 and a final time  $t_f > 0$ , given the initial condition  $v(0) = v_0$ . By employing a constant time-step  $\tau$ , an  $s$ -stage Runge–Kutta time-stepping scheme applied to  $v'(t) = f(v(t), t)$  reads as follows:

$$v_{n+1} = v_n + \tau \sum_{i=1}^s b_i k_{i,n}, \quad n = 0, \dots, n_t - 1,$$

where the stages  $k_{i,n}$  are given by

$$k_{i,n} = f \left( v_n + \tau \sum_{j=1}^s a_{i,j} k_{j,n}, t_n + c_i \tau \right), \quad i = 1, \dots, s, \quad (1)$$

with  $t_n = n\tau$ . The Runge–Kutta method is uniquely defined by the coefficients  $a_{i,j}$ , the weights  $b_i$ , and the nodes  $c_i$ , for  $i, j = 1, \dots, s$ . For this reason, an  $s$ -stage Runge–Kutta method is represented by the following Butcher tableau:

$\mathbf{c}_{\text{RK}}$	$A_{\text{RK}}$
$\mathbf{b}_{\text{RK}}^\top$	

where  $A_{\text{RK}} = \{a_{i,j}\}_{i,j=1}^s$ ,  $\mathbf{b}_{\text{RK}} = [b_1, \dots, b_s]^\top$ , and  $\mathbf{c}_{\text{RK}} = [c_1, \dots, c_s]^\top$ .

In this talk, we consider the following instationary incompressible Navier–Stokes equations

$$\left\{ \begin{array}{ll} \frac{\partial \vec{v}}{\partial t} - \nu \nabla^2 \vec{v} + \vec{v} \cdot \nabla \vec{v} + \nabla p = \vec{f}(\mathbf{x}, t) & \text{in } \Omega \times (0, t_f), \\ -\nabla \cdot \vec{v} = 0 & \text{in } \Omega \times (0, t_f), \\ \vec{v}(\mathbf{x}, t) = \vec{g}(\mathbf{x}, t) & \text{on } \partial\Omega \times (0, t_f), \\ \vec{v}(\mathbf{x}, 0) = \vec{v}_0(\mathbf{x}) & \text{in } \Omega. \end{array} \right.$$

The functions  $\vec{f}$  and  $\vec{g}$  as well as the initial condition  $\vec{v}_0(\mathbf{x})$  are known. Further,  $\nu$  is the viscosity of the fluid. We integrate the problem with a Runge–Kutta scheme in time. The time discretization results in a non-linear system to be solved for the stages of the method at each time step. Specifically, introducing the variables

$$\begin{aligned} \vec{w}_{n,i}^v &= \vec{v}_n + \tau \sum_{j=1}^s a_{i,j} \vec{k}_{j,n}^v, & i &= 1, \dots, s, \\ w_{n,i}^p &= p_n + \tau \sum_{j=1}^s a_{i,j} k_{j,n}^p, & i &= 1, \dots, s, \end{aligned}$$

at each time step the non-linear system (1) that characterizes the stages of the Runge–Kutta method reads as follows:

$$\left\{ \begin{array}{ll} \vec{w}_{n,i}^v - \nu \nabla^2 \vec{w}_{n,i}^v + \vec{w}_{n,i}^v \cdot \nabla \vec{w}_{n,i}^v + \nabla w_{n,i}^p = \vec{f}(\mathbf{x}, t) & i = 1, \dots, s, \\ -\nabla \cdot \vec{w}_{n,i}^v = 0 & i = 1, \dots, s. \end{array} \right. \quad (2)$$

Then, the solutions at time  $t_n + \tau$  are given by

$$\begin{aligned} \vec{v}_{n+1} &= \vec{v}_n + \tau \sum_{i=1}^s b_i \vec{k}_{i,n}^v, \\ p_{n+1} &= p_n + \tau \sum_{i=1}^s b_i k_{i,n}^p. \end{aligned}$$

In order to find a numerical solution, we consider a Newton linearization of the non-linear problem in (2), which is then discretized with suitable finite elements. The resulting linear system presents a saddle-point block structure, and can be very large and sparse in real-life applications. For this reason, in order to find a solution one requires the use of preconditioned iterative methods. We adopt an augmented Lagrangian-based preconditioner, and employ saddle-point theory for deriving approximations of the  $(1, 1)$ -block and the Schur complement. Numerical experiments show the effectiveness and robustness of our approach, for a range of problem parameters.

## References

- [1] O. AXELSSON, I. DRAVINS, AND M. NEYTCHEVA, *Stage-parallel preconditioners for implicit Runge–Kutta methods of arbitrarily high order. Linear problems*, Numer. Linear Algebra Appl., 31 (2024), p. e2532.
- [2] J. C. BUTCHER, *Numerical methods for ordinary differential equations*, John Wiley & Sons, Ltd, 3rd ed., 2016.
- [3] E. HAIRER AND G. WANNER, *Solving ordinary differential equations II: stiff and differential-algebraic problems*, Springer Berlin, Heidelberg, 2nd ed., 1996.
- [4] J. D. LAMBERT, *Numerical method for ordinary differential systems: the initial value problem*, John Wiley & Sons, Ltd, New York, 1991.

- [5] K.-A. MARDAL, T. K. NILSSEN, AND G. A. STAFF, *Order-optimal preconditioners for implicit Runge–Kutta schemes applied to parabolic PDEs*, SIAM J. Sci. Comput., 29 (2007), pp. 361–375.
- [6] M. M. RANA, V. E. HOWLE, K. LONG, A. MEEK, AND W. MILESTONE, *A new block preconditioner for implicit Runge–Kutta methods for parabolic PDE problems*, SIAM J. Sci. Comput., 43 (2021), pp. S475–S495.
- [7] J. VAN LENT AND S. VANDEWALLE, *Multigrid methods for implicit Runge–Kutta and boundary value method discretizations of parabolic PDEs*, SIAM J. Sci. Comput., 27 (2005), pp. 67–92.

# The NPDo Approach For Optimization On The Stiefel Manifold with Applications

*Ren-Cang Li*

## Abstract

NPDo stands for *nonlinear polar decomposition with orthogonal factor dependency*. The NPDo approach is a unified framework recently proposed in [3] for solving certain optimization on the Stiefel manifold. Previously, the approach was implicitly employed in [5]. In this talk, we will explain the theory behind the approach, why it works, the known types of problems for which it is guaranteed to work, and discuss some of its applications in today's data science, including subspace learning and partially joint block diagonalization of several Hermitian matrices.

## References

- [1] Z. Bai, R.-C. Li, and D. Lu. Sharp estimation of convergence rate for self-consistent field iteration to solve eigenvector-dependent nonlinear eigenvalue problems. *SIAM J. Matrix Anal. Appl.*, 43(1):301–327, 2022.
- [2] Y. Cai, L.-H. Zhang, Z. Bai, and R.-C. Li. On an eigenvector-dependent nonlinear eigenvalue problem. *SIAM J. Matrix Anal. Appl.*, 39(3):1360–1382, 2018.
- [3] R.-C. Li. A theory of the NEPv approach for optimization on the Stiefel manifold. [arXiv:2305.00091](https://arxiv.org/abs/2305.00091) (88 pages), to appear in *Found. Comput. Math.*, 2024.
- [4] D. Lu and R.-C. Li. Locally unitarily invariantizable NEPv and convergence analysis of SCF. *Math. Comp.*, 93(349):2291–2329, 2024.
- [5] L. Wang, L.-H. Zhang, and R.-C. Li. Maximizing sum of coupled traces with applications. *Numer. Math.*, 152:587–629, 2022.
- [6] L. Wang, L.-H. Zhang, and R.-C. Li. Trace ratio optimization with an application to multi-view learning. *Math. Program.*, 201:97–131, 2023.
- [7] L.-H. Zhang, L. Wang, Z. Bai, and R.-C. Li. A self-consistent-field iteration for orthogonal canonical correlation analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(2):890–904, 2022.

# Adaptive Sketching Based Construction of $\mathcal{H}^2$ Matrices on GPUs

*Sherry Li, Wajih Boukaram, Yang Liu, Pieter Ghysels*

## Abstract

We present a novel linear-complexity bottom-up sketching-based algorithm for constructing a  $\mathcal{H}^2$  matrix and its high performance GPU implementation. The construction algorithm requires both a black-box sketching operator and an entry evaluation function. The novelty of our GPU approach centers around the design and implementation of the above two operations in batched mode on GPU with accommodation for variable-size data structures in a batch. The batch algorithms minimize the number of kernel launches and maximize the GPU throughput. When applied to covariance matrices, volume IE matrices and  $\mathcal{H}^2$  update operations, our proposed GPU implementation achieves up to  $13\times$  speedup over our CPU implementation, and up to  $1000\times$  speedup over an existing GPU implementation of the top-down sketching-based algorithm from the H2Opus library. This is the first GPU implementation of the class of bottom-up sketching-based  $\mathcal{H}^2$  construction algorithms.

## Reference

W. Boukaram, Y. Liu, P. Ghysels, X.S. Li, “Adaptive Sketching Based Construction of  $\mathcal{H}^2$  Matrices on GPUs”, Proc. of IPDPS Workshop Parallel and Distributed Scientific and Engineering Computing, Milano, Italy, June 3-7, 2025.

Stochastic algebraic Riccati equations are almost as easy as deterministic ones

Xin Liang, Zhen-Chen Guo

Abstract

Algebraic Riccati equations (AREs) arise in various models related to control theory, especially in linear-quadratic optimal control design. The deterministic/classical ones are considered for the deterministic linear time-invariant systems, including discrete-time algebraic Riccati equations (DAREs)

$$X = A^T X A + Q - (A^T X B + L)(R + B^T X B)^{-1}(B^T X A + L^T),$$

and continuous-time algebraic Riccati equations (CAREs)

$$A^T X + X A + Q - (X B + L) R^{-1} (B^T X + L^T) = 0.$$

During many years, people have developed rich theoretical results and numerical methods for the DAREs and CAREs. See [22, 21, 18, 3, 16, 2] to obtain an overview for both theories and algorithms.

In comparison, the stochastic/rational ones are considered for the stochastic linear time-invariant systems, including stochastic discrete-time algebraic Riccati equations (SDAREs)

$$\begin{aligned} X &= A_0^T X A_0 + \sum_{i=1}^{r-1} A_i^T X A_i + Q \\ &\quad - (A_0^T X B_0 + \sum_{i=1}^{r-1} A_i^T X B_i + L)(B_0^T X B_0 + \sum_{i=1}^{r-1} B_i^T X B_i + R)^{-1}(B_0^T X A_0 + \sum_{i=1}^{r-1} B_i^T X A_i + L^T), \end{aligned}$$

and stochastic continuous-time algebraic Riccati equations (SCAREs)

$$\begin{aligned} &A_0^T X + X A_0 + \sum_{i=1}^{r-1} A_i^T X A_i + Q \\ &\quad - (X B_0 + \sum_{i=1}^{r-1} A_i^T X B_i + L)(\sum_{i=1}^{r-1} B_i^T X B_i + R)^{-1}(B_0^T X + \sum_{i=1}^{r-1} B_i^T X A_i + L^T) = 0. \end{aligned}$$

Here  $r - 1$  is the number of stochastic processes involved in the stochastic systems dealt with, and it is easy to check that for the case  $r = 1$  SDAREs and SCAREs degenerate to DAREs and CAREs respectively. Due to the complicated forms, we may recognize it would be much more difficult to analyze their properties and obtain their solutions. There are still literature, e.g., [6, 7, 8], discussing the stochastic linear systems and the induced stochastic AREs.

As we can see, the stochastic AREs are still algebraic, and it is quite natural to ask whether algebraic methods could be developed to solve them. However, limited by lack of clear algebraic structures, to the best of the authors' knowledge, nearly all of the existing algorithms are based on the differentiability or continuity of the equations, such as Newton's method [6, 5], modified Newton's method [12, 19, 4], Lyapunov/Stein iterations [9, 20, 24], comparison theorem based method [10, 11], LMI's (linear matrix inequality) method [23, 17], and homotopy method [25].

The key to the problem is the algebraic structures behind the equations. In this talk, we will build up a simple and clear algebraic interpretation of SDAREs and SCAREs with the help of the so-called left semi-tensor product. In the analysis we find out the Toeplitz structure and the symplectic structure appearing in the equations, and illustrate the fact that the fixed point iteration and the doubling iteration are also valid for them. The algebraic structures found here will shed light on the theoretical analysis and numerical algorithms design, and strongly imply that stochastic AREs are almost as easy as deterministic ones.

As an example, we show how to propose a RADI-type method for large-scale stochastic continuous-time algebraic Riccati equations with sparse and low-rank matrices ( $A_i$  are large-scale and sparse, and  $B_i$  and  $Q - LR^{-1}L^T$  are low-rank), based on revealed algebraic structure and motivated by the relation illustrated in [13] between the algebraic structure and the efficient RADI method [1] for CAREs. Unlike many existing methods for large-scale problems such as Newton-type methods and homotopy method, it calculates the residual at a low cost and does not require a stabilizing initial approximation, which can often be challenging to find. Numerical experiments are provided to demonstrate its efficiency.

This talk is based on [13, 14, 15].

## References

- [1] P. Benner, Z. Bujanović, P. Kürschner, and J. Saak, “RADI: a low-rank ADI-type algorithm for large-scale algebraic Riccati equations,” *Numer. Math.*, vol. 138, pp. 301–330, 2018.
- [2] ———, “A numerical comparison of different solvers for large-scale, continuous-time algebraic Riccati equations and LQR problems,” *SIAM J. Sci. Comput.*, vol. 42, no. 2, pp. A957–A996, 2020.
- [3] D. A. Bini, B. Iannazzo, and B. Meini, *Numerical Solution of Algebraic Riccati Equations*, ser. Fundamentals of Algorithms. Philadelphia: SIAM Publications, 2012, vol. 9.
- [4] E. K.-w. Chu, T. Li, W.-W. Lin, and C.-Y. Weng, “A modified newton’s method for rational riccati equations arising in stochastic control,” in *2011 International Conference on Communications, Computing and Control Applications (CCCC)*, 2011, pp. 1–6.
- [5] T. Damm and D. Hinrichsen, “Newton’s method for a rational matrix equation occurring in stochastic control,” *Linear Algebra Appl.*, vol. 332-334, pp. 81–109, 2001.
- [6] T. Damm, *Rational Matrix Equations in Stochastic Control*. Berlin/Heidelberg, Germany: Springer-Verlag, 2004.
- [7] V. Dragan, T. Morozan, and A.-M. Stoica, *Mathematical Methods in Robust Control of Discrete-Time Linear Stochastic Systems*. New York, NY, USA: Springer-Verlag, 2010.
- [8] ———, *Mathematical Methods in Robust Control of Linear Stochastic Systems*, 2nd ed. New York, NY, USA: Springer-Verlag, 2013.
- [9] H.-Y. Fan, P. C.-Y. Weng, and E. K. wah Chu, “Smith method for generalized Lyapunov/Stein and rational Riccati equations in stochastic control,” *Numer. Alg.*, vol. 71, pp. 245–272, 2016.
- [10] G. Freiling and A. Hochhaus, “Properties of the solutions of ration matrix difference equations,” *Computers Math. Appl.*, vol. 45, pp. 1137–1154, 2003.

- [11] ——, “On a class of rational matrix differential equations arising in stochastic control,” *Linear Algebra Appl.*, vol. 379, pp. 43–68, 2004.
- [12] C.-H. Guo, “Iterative solution of a matrix Riccati equation arising in stochastic control,” *Oper. Theory: Adv. Appl.*, vol. 130, pp. 209–221, 2001.
- [13] Z.-C. Guo and X. Liang, “The intrinsic Toeplitz structure and its applications in algebraic Riccati equations,” *Numer. Alg.*, vol. 93, pp. 227–267, 2023.
- [14] ——, “Stochastic algebraic Riccati equations are almost as easy as deterministic ones theoretically,” *SIAM J. Matrix Anal. Appl.*, vol. 44, no. 4, pp. 1749–1770, 2023.
- [15] ——, “An RADI-type method for stochastic continuous-time algebraic Riccati equations,” 2024, 21 pages.
- [16] T.-M. Huang, R.-C. Li, and W.-W. Lin, *Structure-Preserving Doubling Algorithms for Nonlinear Matrix Equations*, ser. Fundamentals of Algorithms. Philadelphia: SIAM, 2018, vol. 14.
- [17] H. Iiduka and I. Yamada, “Computational method for solving a stochastic linear-quadratic control problem given an unsolvable stochastic algebraic Riccati equation,” *SIAM J. Control Optim.*, vol. 50, no. 4, pp. 2173–2192, 2012.
- [18] V. Ionescu, C. Oară, and M. Weiss, *Generalized Riccati Theory and Robust Control: A Popov Function Approach*. Chichester, UK: John Wiley & Sons, 1999.
- [19] I. G. Ivanov, “Iterations for solving a rational Riccati equations arising in stochastic control,” *Computers Math. Appl.*, vol. 53, pp. 977–988, 2007.
- [20] ——, “Properties of Stein (Lyapunov) iterations for solving a general Riccati equation,” *Nonlinear Anal.*, vol. 67, pp. 1155–1166, 2007.
- [21] P. Lancaster and L. Rodman, *Algebraic Riccati Equations*. New York: The Clarendon Press, Oxford Science Publications, 1995.
- [22] V. L. Mehrmann, “The autonomous linear quadratic control problems,” in *Lecture Notes in Control and Information Sciences*. Berlin: Springer-Verlag, 1991, vol. 163.
- [23] M. A. Rami and X. Y. Zhou, “Linear matrix inequalities, Riccati equations, and indefinite stochastic linear quadratic controls,” *IEEE Trans. Automat. Control*, vol. 45, no. 6, pp. 1131–1143, 2000.
- [24] N. Takahashi, M. Kono, T. Suzuki, and O. Sato, “A numerical solution of the stochastic discrete algebraic Riccati equation,” *J. Archaeological Sci.*, vol. 13, pp. 451–454, 2009.
- [25] L. Zhang, H.-Y. Fan, E. K. Wah Chu, and Y. Wei, “Homotopy for rational Riccati equations arising in stochastic optimal control,” *SIAM J. Sci. Comput.*, vol. 37, no. 1, pp. B103–B125, 2015.

# Mixed precision HODLR matrices

*Xiaobo Liu<sup>1</sup>, Erin Carson<sup>2</sup>, Xinye Chen<sup>2</sup>*

## Abstract

**Introduction and overview.** Hierarchical matrices, often abbreviated as  $\mathcal{H}$ -matrices [1], comprise a class of dense rank-structured matrices with a hierarchical low-rank structure, which is used to approximate a dense or sparse matrix by dividing it into multiple submatrices in a hierarchical way, where a number of submatrices are selected to be approximated by low-rank factors according to an admissibility condition.

Computations of hierarchical matrices have attracted significant attention in the science and engineering community as exploiting data-sparse structures can significantly reduce the computational complexity of many important kernels such as matrix–vector products, matrix factorizations, etc. One particularly popular option within this class is the *Hierarchical Off-Diagonal Low-Rank* (HODLR) format, whose definition is associated with the binary cluster tree  $\mathcal{T}_\ell$  of depth  $\ell \in \mathbb{N}^+$  [4].

**Definition 1** (( $\mathcal{T}_\ell, p$ )-HODLR matrix).  *$H \in \mathbb{R}^{n \times n}$  is ( $\mathcal{T}_\ell, p$ )-HODLR matrix if every off-diagonal block  $H(I_i^k, I_j^k)$  associated with siblings  $I_i^k$  and  $I_j^k$  in  $\mathcal{T}_\ell$ ,  $k = 1, \dots, \ell$ , has rank at most  $p$ .*

In the proposed talk, we consider constructing HODLR matrices in a mixed precision manner and offer insights into the resulting behavior of finite precision computations. Our analysis confirms what is largely intuitive: the lower the quality of the low-rank approximation, the lower the precision which can be used without detriment. We provide theoretical bounds which determine which precisions can safely be used in order to balance the overall error.

**Practical definition of HODLR matrix.** In order to quantify the error incurred in the low-rank factorization of the off-diagonal blocks, we introduce the practical definition of  $(\mathcal{T}_\ell, p, \varepsilon)$ -HODLR matrix as in Definition 2. The approximation error in the diagonal blocks of all levels of the  $(\mathcal{T}_\ell, p, \varepsilon)$ -HODLR matrix  $\tilde{H}$  is immediately obtainable following Definition 2 in the Frobenius norm, and, as a special case, one can show  $\|\tilde{H} - H\|_F \leq \varepsilon \|H\|_F$ .

**Definition 2** (( $\mathcal{T}_\ell, p, \varepsilon$ )-HODLR matrix). *Let  $H \in \mathbb{R}^{n \times n}$  be a  $(\mathcal{T}_\ell, p)$ -HODLR matrix. Then  $\tilde{H} \in \mathbb{R}^{n \times n}$  is defined to be a  $(\mathcal{T}_\ell, p, \varepsilon)$ -HODLR matrix to  $H$ , if every off-diagonal block  $\tilde{H}(I_i^k, I_j^k)$  associated with siblings  $I_i^k$  and  $I_j^k$  in  $\mathcal{T}_\ell$ ,  $k = 1, \dots, \ell$ , satisfies  $\|\tilde{H}(I_i^k, I_j^k) - H(I_i^k, I_j^k)\| \leq \varepsilon \|H(I_i^k, I_j^k)\|$ , where  $0 \leq \varepsilon < 1$ .*

**Mixed-precision representation.** First, we develop a mixed precision algorithm for constructing HODLR matrices. Let us assume that the off-diagonal blocks from the  $k$ th level of  $\tilde{H}$ ,  $1 \leq k \leq \ell$ , are compressed in the form

$$\tilde{H}_{ij}^{(k)} = \tilde{U}_i^{(k)} (\tilde{V}_j^{(k)})^T, \quad |i - j| = 1, \quad (1)$$

where  $\tilde{U}_i^{(k)} \in \mathbb{R}^{n/2^k \times p}$  has orthonormal columns to precision  $u$  and  $\tilde{V}_j^{(k)} \in \mathbb{R}^{n/2^k \times p}$ . Our idea is to compress the low-rank blocks  $\tilde{H}_{ij}^{(k)}$  and represent the low-rank factors  $\tilde{U}_i^{(k)}$  and  $\tilde{V}_j^{(k)}$  in precisions potentially lower than the working precision; given a set of available precisions, the same precision, say,  $u_k$ , is used for the storage of all low-rank factors at level  $k$ . To keep the global error in the

---

<sup>1</sup>Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany.

<sup>2</sup>Department of Numerical Mathematics, Charles University, Prague, Czech Republic.

mixed-precision representation at the same level as an unified working-precision representation, we choose

$$u_k \leq \varepsilon / (2^{k/2} \xi_k),$$

where  $\varepsilon > u$  (since the factorizations are calculated in the working precision  $u$ ) can be thought of as the accuracy threshold in the low-rank factorizations (1) and

$$\xi_k := \max_{|i-j|=1} \|\tilde{H}_{ij}^{(k)}\|_F / \|\tilde{H}\|_F, \quad 1 \leq k \leq \ell,$$

which essentially characterizes the relative importance of the off-diagonal blocks in level- $k$  to the whole matrix in terms of magnitude. This means that, as the tree depth increases, the unit roundoff  $u_k$  must be smaller to offset the error between the HODLR matrix and the original matrix and that, since  $0 < \xi_k < 1$  holds for  $k = 1 : \ell$ , generally no higher-than-working precisions are needed among  $u_k$  for a HODLR matrix with mild depth  $\ell$ , say,  $\ell \leq 10$  (so  $2^{k/2} \leq 32$ ). We then propose an adaptive scheme for precision selection, which dynamically determines what degree of precision is required for the computations in each level of the cluster tree. We show that the error in the resulting mixed-precision representation  $\hat{H}$  satisfies

$$\|H - \hat{H}\|_F \lesssim (2\sqrt{2\ell} + 1)\varepsilon\|H\|_F.$$

**Matrix–vector products.** Next, we give error bounds on the working precision  $u$  so that the backward error in computing the matrix–vector product in finite precision does not exceed the error resulting from inexact representation of the matrix. The key idea is that, if the HODLR matrix  $H$  is approximated by the mixed-precision representation  $\hat{H}$ , to calculate the matrix–vector product  $b \leftarrow \hat{H}x$  we should try to balance the errors occurring in the approximation of  $\hat{H}$  and in the finite-precision computation, as shown from the following result.

**Lemma 1.** *Let  $\hat{A}_p$  an approximation of  $A$  such that  $\|A - \hat{A}_p\|_F \lesssim \eta$  for some  $\eta > 0$ . Then the error due to finite precision computation of  $\hat{y} = \text{fl}(\hat{A}_p x)$  will be no larger than the error due to the computed inexact representation when the working precision has unit roundoff  $u \leq \eta / (n\|\hat{A}_p\|_F)$ .*

Applying Lemma 1 to the computation of the matrix–vector product associated with  $\hat{H}_{ij}^{(k)}$  and ignoring the errors in the summation of the vector elements (which are usually negligible compared with the error in the block matrix–vector products), we can obtain the following result.

**Theorem 1.** *Let  $\tilde{H}$  be a  $(\mathcal{T}_\ell, p, \varepsilon)$ -HODLR matrix associated with the HODLR matrix  $H$ , and let  $\hat{H}$  denote our mixed-precision representation. If  $b = \hat{H}x$  is computed in a working  $u \leq \varepsilon/n$ , then the computed  $\hat{b}$  satisfies*

$$\hat{b} = \text{fl}(\hat{H}x) = (H + \Delta H)x, \quad \|\Delta H\|_F \leq 10 \cdot 2^{\ell/2} \varepsilon \|H\|_F.$$

**LU factorization.** Finally, we derive error bounds on the LU factorization of the mixed-precision HODLR matrix  $\hat{H}$ . The factorization is done by a recursive algorithm which computes for all but the bottom level the block LU factorization

$$\begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix} = \begin{bmatrix} L_{11} & \\ L_{21} & L_{22} \end{bmatrix} \begin{bmatrix} U_{11} & U_{12} \\ & U_{22} \end{bmatrix},$$

where  $L_{11}$  and  $L_{22}$  are lower triangular and  $U_{11}$  and  $U_{22}$  are upper triangular, and it invokes dense routines on the bottom level. Based on the results from [3, sect. 3.5] and [3, Thm. 8.5], we first look at the backward error in the LU factorization of the HODLR matrices at level  $k = \ell - 1$  and then use induction to quantify the backward error in the LU decomposition of diagonal blocks in the other levels, up to the level  $k = 0$  ( $H_{11}^{(0)} := H$ ). We arrive at the following result.

**Theorem 2.** Let  $\widehat{H}$  be the mixed-precision  $\ell$ -level HODLR representation. If the LU decomposition of  $\widehat{H}$  is computed in a working precision  $u \lesssim \varepsilon/n$ , then the LU factorization of the HODLR matrix  $\widehat{H}$  satisfies

$$\widehat{L}\widehat{U} = H + \Delta H, \quad \|\Delta H\|_F \lesssim 2^{\ell+1}\varepsilon\|H\|_F + 11 \cdot 2^\ell\varepsilon\|\widehat{L}\|_F\|\widehat{U}\|_F.$$

Noted that our finite precision analysis remains valid in the case where the HODLR matrices are stored in one precision and therefore also provides new results for this case. We will also present the numerical simulations we performed across various datasets to verify our theoretical results.

The talk is based on [2]. We have also developed a MATLAB toolbox called `mhodlr` for matrix computations with HODLR representation and mixed-precision simulations, which supports other important operations within the class of HODLR matrix such as (mixed-precision) matrix multiplication and Cholesky factorization. The documentation webpage of `mhodlr` MATLAB toolbox is at <https://mhodlr.readthedocs.io/en/latest/index.html>.

## References

- [1] Steffen Börm, Lars Grasedyck, and Wolfgang Hackbusch. Introduction to hierarchical matrices with applications. *Eng. Anal. Bound. Elem.*, 27(5):405–422, 2003.
- [2] Erin Carson, Xinye Chen, and Xiaobo Liu. Mixed precision HODLR matrices. ArXiv:2407.21637 [math.NA], July 2024.
- [3] Nicholas J. Higham. *Accuracy and Stability of Numerical Algorithms*. SIAM, Philadelphia, PA, USA, second edition, 2002.
- [4] Stefano Massei, Leonardo Robol, and Daniel Kressner. hm-toolbox: MATLAB software for HODLR and HSS matrices. *SIAM J. Sci. Comput.*, 42(2):C43–C68, 2020.

# Convergence Analysis of SCF Iteration for Eigenvector-Dependent Nonlinear Eigenvalue Problems

Ding Lu

Abstract

Eigenvector-dependent Nonlinear Eigenvalue Problems (NEPv) are fundamental in computational science and engineering, presenting intriguing challenges for both analysis and computation. In an NEPv, the goal is to find an orthonormal matrix  $V \in \mathbb{C}^{n \times k}$ , i.e.,  $V^H V = I_k$ , and a square matrix  $\Lambda \in \mathbb{C}^{k \times k}$  that satisfy the nonlinear equation:

$$H(V)V = V\Lambda,$$

where  $H(V) \in \mathbb{C}^{n \times n}$  is a Hermitian matrix that continuously depends on the eigenvectors  $V$ . It is typically assumed that the matrix-valued function  $H(V)$  is right unitarily invariant, meaning  $H(VQ) = H(V)$  for any unitary matrix  $Q \in \mathbb{C}^{k \times k}$ , and that the eigenvalues of  $\Lambda \equiv V^H H(V)V$  correspond to the  $k$  smallest (or largest) eigenvalues of  $H(V)$ .

NEPv arise in various fields, from traditional applications such as electronic structure calculations in computational physics and chemistry, to more recent uses in machine learning in data science, and signal processing of brain-computer interface in neuroscience and biomedical engineering.

The self-consistent field (SCF) iteration, originally introduced in molecular quantum mechanics in the 1950s, is the most general and widely used method for solving NEPv and serves as a foundation for other approaches. Starting from an orthonormal matrix  $V_0 \in \mathbb{C}^{n \times k}$ , SCF iteratively computes

$$H(V_i)V_{i+1} = V_{i+1}\Lambda_{i+1}, \quad \text{for } i = 0, 1, 2, \dots,$$

where  $V_{i+1} \in \mathbb{C}^{n \times k}$  is orthonormal and  $\Lambda_{i+1}$  is a diagonal matrix containing the  $k$  smallest eigenvalues of  $H(V_i)$ . This basic form is known as the plain SCF iteration. Despite its simplicity, plain SCF can suffer from slow convergence or even non-convergence in practice. Understanding when and how plain SCF converges has been a longstanding research challenge, as these insights are crucial for developing techniques to stabilize and accelerate the convergence of SCF iteration.

In this presentation, we cover some of our recent advances in the convergence analysis of plain SCF and its variants. The first part focuses on the local convergence analysis of plain SCF. Using tangent-angle matrix as an intermediate measure for approximation error, we establish new formulas for two fundamental quantities that characterize the local convergence behavior of the plain SCF: the local contraction factor and the local asymptotic average contraction factor. Our new convergence rate estimates yield sharper bounds on the convergence speed compared to previously established results. These findings also provide a new justification for the guaranteed local convergence of a popular SCF variant—the level-shifted SCF. Details are found in [1]. We also mention [3], where we extended the analysis to an SCF-type iteration for unitarily-invariantizable NEPv.

The second part presents a geometric interpretation of SCF to improve our understanding of its global convergence behavior. We begin by focusing on a class of NEPv which we refer to as monotone NEPv (mNEPv). Using a variational characterization of mNEPv, we can visualize plain SCF as a steepest feasible direction method for the associated optimization problem. This interpretation reveals the global and monotonic convergence of plain SCF for mNEPv; Further details can be found in [2]. Finally, we will show how to extend this geometric framework to the level-shifted SCF for general NEPv, thereby establishing its guaranteed global convergence.

This presentation is based on joint work with Zhaojun Bai and Ren-Cang Li.

## References

- [1] Zhaojun Bai, Ren-Cang Li, and Ding Lu. Sharp estimation of convergence rate for self-consistent field iteration to solve eigenvector-dependent nonlinear eigenvalue problems. *SIAM Journal on Matrix Analysis and Applications*, 43(1):301–327, 2022.
- [2] Zhaojun Bai and Ding Lu. Variational characterization of monotone nonlinear eigenvector problems and geometry of self-consistent field iteration. *SIAM Journal on Matrix Analysis and Applications*, 45(1):84–111, 2024.
- [3] Ding Lu and Ren-Cang Li. Locally unitarily invariantizable NEPv and convergence analysis of SCF. *Mathematics of Computation*, (93):2291–2329, 2024.

# A MATLAB Toolbox for Toeplitz-Like Matrix Computations

Robert Luce

## Abstract

A *Toeplitz matrix*  $T \in \mathbb{C}^{n,n}$  is defined by  $2n - 1$  parameters  $t_{-n+1}, \dots, t_{n-1} \in \mathbb{C}$  by

$$T = [t_{|i-j|}]_{i,j} = \begin{bmatrix} t_0 & t_1 & \dots & t_{n-1} \\ t_{-1} & t_0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & t_1 \\ t_{-n+1} & \dots & t_{-1} & t_0 \end{bmatrix}.$$

Such matrices arise in many applications from signal processing to finance, and the design and analysis of algorithms for *computations with Toeplitz matrices* that take advantage of the matrix structure is an ever-continuing quest. In this work we present a MATLAB toolbox for convenient and efficient computations with Toeplitz matrices and "Toeplitz-like" matrices, which we will define in the following, based on *displacement structure*. This more general class of structured matrices enables fast algorithms not only for Toeplitz matrices themselves, but all matrices that satisfy a certain low-rank property, which includes products, polynomials and rational functions of Toeplitz matrices.

We will now discuss the crucial low-rank property that enables fast algorithms in more detail. In the following we need notation for the two unit circulant matrices

$$Z_{\pm 1} := [e_2, e_3, \dots, e_n, \pm e_1] = \begin{bmatrix} & & & \pm 1 \\ 1 & & & \\ & \ddots & & \\ & & & 1 \end{bmatrix},$$

and for a vector  $x \in \mathbb{C}^n$  we denote  $Z_{\pm 1}(x) := \sum_{k=1}^n x_i Z_{\pm 1}^{k-1}$ .

For a matrix  $A \in \mathbb{C}^{n,n}$  the *displacement* of  $A$  is defined as

$$\nabla(A) := \nabla_{Z_1, Z_{-1}}(A) := Z_1 A - A Z_{-1} \in \mathbb{C}^{n,n}.$$

The *displacement rank* of  $A$  is the rank of  $\nabla(A)$ , and when we have a decomposition

$$\nabla(A) = G B^*, \quad G, B \in \mathbb{C}^{n,d},$$

we call the pair  $(G, B)$  a *generator* of  $A$ . It is easily seen that the displacement rank of a Toeplitz matrix cannot exceed 2, and whenever  $\text{rank}(\nabla(A)) \ll n$  we will say that  $A$  is *Toeplitz-like*. The overall mechanics of displacement structure are much more general than what we need for our purpose here; we refer to the classic volume of Kailath et. al. [4] for a broader presentation.

The property of  $\nabla(A)$  having low rank has several important algorithmic consequences for computations involving Toeplitz-like matrices, which we take advantage of in our toolbox. For example, from a generator  $(G, B)$  of  $A$ , having columns  $g_1, \dots, g_d$  and  $b_1, \dots, b_d$ , respectively, one obtains the representation (e.g., [6])

$$A = \sum_{k=1}^d Z_1(g_k) Z_{-1}(J \bar{b}_k), \quad (J \text{ is the anti-identity}),$$

enabling fast multiplication with  $A$  via the FFT without ever forming  $A$  explicitly.

Another important property is that Schur complements of displacement structured matrices inherit the displacement rank [4]. A compact and constructive way to state this property is as follows.

**Theorem.** *Let  $M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \in \mathbb{C}^{2n \times 2n}$  with each block being an  $n \times n$  matrix. If  $M$  satisfies the displacement equation*

$$(Z_1 \oplus Z_1)M - M(Z_{-1} \oplus Z_{-1}) = \begin{bmatrix} G_1 \\ G_2 \end{bmatrix} \begin{bmatrix} B_1^* & B_2^* \end{bmatrix} =: G_M B_M^*,$$

where  $G_M, B_M \in \mathbb{C}^{2n \times d}$  are conformally partitioned with  $M$ , then the Schur complement  $S := M_{22} - M_{21}M_{11}^{-1}M_{12}$  of  $M_{11}$  in  $M$  satisfies the displacement equation  $\nabla(S) = G_S B_S^*$  with

$$G_S = G_2 - M_{21}M_{11}^{-1}G_1, \quad B_S = B_2 - M_{12}^*M_{11}^{-*}B_1.$$

In particular  $S$  has displacement rank at most  $d$ .

The preceding theorem actually applies to other displacement operators, and forms the basis of the famous GKO algorithm [3], which allows solving linear systems with  $A$  via an implicit LU factorization in  $\mathcal{O}(dn^2)$  (after transformation to a Cauchy-like matrix). A more immediate consequence though is that one can derive generator formulas for the result of algebraic operations with Toeplitz-like matrices directly from their generators. The case of a product of two Toeplitz-like matrices is an instructive example.

**Example.** *Let  $A_1, A_2 \in \mathbb{C}^{n \times n}$  two Toeplitz-like matrices of displacement ranks  $d_1, d_2$  and with generators  $(G_1, B_1)$  and  $(G_2, B_2)$ , respectively. Then a generator for the product  $A_1 A_2$  can be obtained by using the preceding theorem on the embedding*

$$M = \begin{bmatrix} -I_n & A_2 \\ A_1 & 0 \end{bmatrix}$$

which is seen to have displacement rank at most  $d_1 + d_2 + 1$ , and a possible generator for  $M$  is

$$G = \begin{bmatrix} e_1 & G_2 & 0 \\ 0 & 0 & G_1 \end{bmatrix}, \quad B = \begin{bmatrix} -2e_n & 0 & B_1 \\ 0 & B_2 & 0 \end{bmatrix}.$$

Hence the preceding theorem asserts that  $S = A_1 A_2$  has displacement rank at most  $d_1 + d_2 + 1$  and a generator for  $A_1 A_2$  is

$$G_S = [A_1 e_1 \quad A_1 G_2 \quad G_1], \quad B_S = [-2A_2^* e_n \quad B_2 \quad A_2^* B_1].$$

The preceding example is typical in the sense that the generator formulas provide a recipe for implementing matrix operations solely on the basis of the generators of the operands and resultant. Further important examples are integer powers, polynomials and rational functions.

Our toolbox TLCOMP implements algorithms for arithmetic and other computations with Toeplitz-like matrices, typically based either on the FFT or by delegation to unstructured, dense computations on their generators. Toeplitz and Toeplitz-like matrices are never stored as full matrices, but instead a generator representation is maintained throughout. Table 1 lists a few examples of the supported operations and their computational complexity.

operation	$\mathcal{O}$ -complexity	dominant operation
$A_1 + A_2$	$n(d_1 + d_2)^2$	generator (re-)compression
$A_1 b$	$d_1 n \log n$	FFT
$A_1 A_2$	$d_1 d_2 n \log n$	FFT
$\text{full}(A_1)$	$d_1 n^2$	None
$\text{mpower}(T, s)$	$s n \log n$	FFT
$\text{polyvalm}(p, T)$	$s n \log n$	FFT
$\text{polyvalm}(p, A_1)$	$d_1 s n \log n$	FFT
$T \backslash b$	$n^2$	GKO
$A_1 \backslash b$	$d_1 n^2$	GKO

Table 1: Selected operations in TLCOMP. Here  $T$  is a Toeplitz matrix,  $A_1$  and  $A_2$  are Toeplitz-like matrices of displacement rank  $d_1$  and  $d_2$ , respectively,  $b \in \mathbb{C}^n$  and  $p$  is a polynomial of degree  $s$ .

In order to maintain the generator representation throughout, an underlying generator  $(G, B)$ , say, comprising  $d$  columns, will be *compressed* to the numerical rank of the displacement, or sharp rank bounds (if available). In our toolbox this is achieved by thin QR factorizations of both  $G$  and  $B$ , followed by an SVD of a smaller  $d$ -by- $d$  matrix to determine the rank. The overall complexity of this recompression procedure is only in  $\mathcal{O}(d^2n)$  and is typically dominated by other computational costs.

Our workhorse for solving linear systems of equations with Toeplitz-like matrices is the GKO algorithm [3] as implemented in the excellent MATLAB toolbox “drsolve” by Aricò and Rodriguez [1]. It may be interesting to add an option for using super-fast solvers in applicable cases (e.g., [5]), but in our experience the GKO approach is highly competitive in practice up to very large matrix dimensions despite having a worse complexity.

In order to give an idea on how TLCOMP can be used, we will show a few simple command prompts that involve our toolbox. Toeplitz matrices are represented by a `ToepMat` class. When possible, arithmetic with Toeplitz matrices yield Toeplitz matrices again:

```
% Generate data for two random Toeplitz matrices
[c1, r1] = random_toeplitz(1000, 1000);
[c2, r2] = random_toeplitz(1000, 1000);

% We provide a class |ToepMat|
TM1 = ToepMat(c1, r1);
TM2 = ToepMat(c2, r2);

% Addition, scalar multiplication yield a ToepMat object
disp(TM1 + TM2)
disp(TM1 - TM2)
disp(2i*pi * TM1)

1000x1000 ToepMat
1000x1000 ToepMat
1000x1000 ToepMat
```

If the result of an operation cannot be represented as a Toeplitz matrix, it will be type-promoted to a Toeplitz-like matrix, represented by the `TLMat` class:

```

disp(TM1 * TM2)
disp(TM1 \ TM2)

1000x1000 TLMat, displacement rank 4
1000x1000 TLMat, displacement rank 3

```

Evaluate Taylor polynomial of degree six for the exponential function:

```

p = 1./factorial(6:-1:0);
E = polyvalm(p, TM1); % No "full" arithmetic here!
disp(E); % Result is a TLMat

1000x1000 TLMat, displacement rank 12

EE = polyvalm(p, full(TM1)); % Compare with result from "full" computation
disp(norm(E - EE, 'fro') / norm(EE, 'fro'));

6.8215e-15

```

A preliminary version of this toolbox with some fewer features has been used to facilitate the numerical experiments in [2]. This preliminary version is already available on GitHub at

<https://github.com/rduce/tlcomp>

and we hope that it will aid our community and beyond to embrace structured matrix computations in research and applications.

## References

- [1] A. ARICÒ AND G. RODRIGUEZ, *A fast solver for linear systems with displacement structure*, Numer. Algorithms, 55 (2010), pp. 529–556.
- [2] B. BECKERMANN, J. BISCH, AND R. LUCE, *On the rational approximation of Markov functions, with applications to the computation of Markov functions of Toeplitz matrices*, Numer. Algor., 91 (2022), pp. 109–144.
- [3] I. GOHBERG, T. KAILATH, AND V. OLSHEVSKY, *Fast Gaussian elimination with partial pivoting for matrices with displacement structure*, Math. Comp., 64 (1995), pp. 1557–1576.
- [4] T. KAILATH AND A. H. SAYED, eds., *Fast reliable algorithms for matrices with structure*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1999.
- [5] S. MASSEI, L. ROBOL, AND D. KRESSNER, *hm-toolbox: MATLAB Software for HODLR and HSS Matrices*, SIAM Journal on Sci. Comp., 42(2) (2020), pp. C43–C68.
- [6] V. Y. PAN, *Structured matrices and polynomials*, Birkhäuser Boston, Inc., Boston, MA; Springer-Verlag, New York, 2001.

# Building Scalable Tensor Regression Models: Linear Solvers and Beyond

Hengrui Luo, Anna Ma, Akira Horiguchi, Li Ma

## Abstract

The regression problem  $y = f(\mathbf{X}) + \epsilon$ , where  $\mathbf{X} \in \mathbb{R}^{n \times d_1 \times d_2}$ ,  $y \in \mathbb{R}^1$  is a tensor input variable, presents unique challenges at the intersection of numerical linear algebra and statistical modeling. This extended abstract synthesizes recent advancements in solving both linear and non-linear variants of this problem, focusing on scalable methods that leverage tensor structures.

In the linear case, we consider the tensor regression model  $f(\mathbf{X}) = \mathbf{B} \circ \mathbf{X}$ , where  $\circ$  denotes a tensor product. In [1], we focus on a generalization of matrix multiplication to tensors. This formulation leads to large-scale tensor linear systems that are computationally challenging to solve using traditional methods. Our primary contribution in this domain is the development of frontal slice approaches for iteratively solving these systems.

Our innovation in [1] lies in adapting classical iterative methods from numerical linear algebra to the tensor domain. By focusing on frontal slices of the coefficient tensor  $\mathbf{B}$ , we develop a family of iterative algorithms that significantly reduce the computational burden compared to direct solvers. Our frontal slice methods come in cyclic, block, and randomized variants, each offering different trade-offs between convergence speed and computational efficiency. These approaches draw inspiration from classical numerical linear algebra techniques such as Gauss-Seidel, block iterative methods, and randomized algorithms.

Another aspect of our work is the rigorous convergence analysis provided for these methods. We establish conditions for convergence and derive bounds on the convergence rate in terms of tensor properties. This analysis not only validates the proposed methods but also offers insights into algorithm behavior, guiding practitioners in method selection and parameter tuning.

The computational advantages of our approach are particularly striking for large-scale problems. We achieve a computational complexity of  $O(d_2 \cdot \max(d_1, n)^3)$  for  $n$  samples and tensor dimensions  $d_1$  and  $d_2$ , comparing favorably to the  $O(d_2^2 \cdot \max(d_1, n)^3)$  complexity of naive iterative solver via gradient descent, when  $d_2$  is large.

Extending beyond linear models,  $f(\mathbf{X})$  are assumed to be nonlinear. We address the challenge of non-linear tensor regression through the development of *tensor-input tree* (TT) models in [2]. Our model innovation generalize decision trees to handle tensor inputs, offering a flexible, non-parametric approach to modeling complex relationships in tensor data. The TT framework can be viewed as approximating the non-linear function  $f(\mathbf{X})$  with a piecewise linear tensor function, where each piece corresponds to a leaf in the decision tree.

The development of TT models required innovative solutions to several challenges at the interface of numerical linear algebra and statistical learning. Our key contribution in [2] is the design of splitting criteria that effectively capture the multi-dimensional structure of tensor inputs. We propose criteria based on both variance reduction and low-rank approximation errors, leveraging tensor algebraic concepts to inform the tree-building process. To address the computational challenges of finding optimal splits in high-dimensional tensor spaces, we introduce two randomized numerical linear algebra techniques: leverage score sampling and branch-and-bound optimization. The complexity for generating the tree structure (via exhaustive search) is  $O(n^2 \cdot d_1^2 \cdot d_2^2 \cdot \log k)$ , where  $k$  is the number of nodes in the tree, in contrast to  $O(n^3)$  in a tensor Gaussian process [3]. This approach aligns with theoretical results on tensor decompositions and proves highly effective in practice, especially for datasets exhibiting intrinsic low-rank structure.

The theoretical analysis provided for both our linear solvers and nonlinear TT models offers valuable insights into their behavior and limitations. For the linear case, we provide detailed convergence analysis, including results for both consistent and inconsistent systems. In the context of TT models, we establish consistency guarantees for coefficient estimates and derive oracle bounds for prediction errors.

Empirical evaluations on diverse datasets demonstrate the effectiveness of our approaches. In the linear case, we show the efficiency of frontal slice methods in applications like image deblurring. For non-linear regression, we compare TT models against state-of-the-art approaches like tensor Gaussian Processes, highlighting scenarios where TT offers superior performance or significant computational advantages.

From a numerical linear algebra perspective, our work opens several intriguing avenues for future research. The frontal slice methods suggest possibilities for developing tensor analogues of classical iterative methods. The efficient splitting criteria used in TT models could inspire new preconditioning techniques for tensor computations. Furthermore, the integration of randomized algorithms in both our linear solvers and regression models points to a broader trend of leveraging stochastic methods to tackle high-dimensional problems.

In conclusion, our research demonstrates the power of combining insights from numerical linear algebra with advanced statistical modeling techniques to address the challenges of tensor regression. By developing scalable methods for both linear and non-linear tensor regression (including tensor-on-tensor and tensor-on-scalar), we contribute to a comprehensive toolkit for tensor-based data analysis, laying the groundwork for future advancements in high-dimensional data analysis and scientific computing.

#### *References*

- [1] Hengrui Luo, and Anna Ma. “Frontal Slice Approaches for Tensor Linear Systems.” arXiv preprint arXiv:2408.13547 (2024).
- [2] Hengrui Luo, Akira Horiguchi, and Li Ma. “Efficient Decision Trees for Tensor Regressions.” arXiv preprint arXiv:2408.01926 (2024).
- [3] Rose Yu, Guangyu Li, and Yan Liu. “Tensor regression meets Gaussian processes.” International Conference on Artificial Intelligence and Statistics. PMLR, 2018.

# Randomized Kaczmarz on doubly noisy systems and its applications

*Anna Ma, El Houcine Bergou, Soumia Boucherouite, Aritra Dutta and Xin Li*

## Abstract

Large-scale linear systems,  $Ax = b$ , frequently arise in practice and demand effective iterative solvers. Often, these systems are noisy due to operational errors or faulty data-collection processes. In the past decade, the randomized Kaczmarz (RK) algorithm has been studied extensively as an efficient iterative solver for such systems. However, the convergence of RK in the noisy system regime is limited and typically only considers measurement noise in the right-hand side vector,  $b$ . Unfortunately, in practice, that is not always the case; the coefficient matrix  $A$  can also be noisy. In this talk, we present the analysis of the convergence of RK for *doubly-noisy* linear systems, i.e., when the coefficient matrix,  $A$ , has additive or multiplicative noise, and  $b$  is also noisy. In our analyses, we provide convergence bounds depending on the quantity  $\tilde{R} = \|\tilde{A}^\dagger\|^2 \|\tilde{A}\|_F^2$ , where  $\tilde{A}$  represents a noisy version of  $A$ . This work opens the doors to applications, two of which we highlight in this talk. The first is additive preconditioning in which additive noise to the matrix is *intentionally* added to improve the initial convergence of RK. The second considers the extremely large-scale setting in which even entire rows of the matrix  $A$  cannot be loaded or used in a single iteration. In such a case, we use noise to model the sparsification of the rows of  $A$  to decrease computational memory or communication demand. The work presented is joint work with El Houcine Bergou, Soumia Boucherouite, Aritra Dutta and Xin Li [1].

## References

- [1] El Houcine Bergou, Soumia Boucherouite, Aritra Dutta, Xin Li, and Anna Ma. A note on the randomized kaczmarz algorithm for solving doubly noisy linear systems. *SIAM Journal on Matrix Analysis and Applications*, 45(2):992–1006, 2024.

# Efficient tensor network contraction algorithms

*Linjian Ma, Edgar Solomonik*

## Abstract

Tensors are multidimensional arrays that generalize the vector and matrix concepts. Formally-speaking, an  $N$ -way or  $N$ th-order tensor is an element of the tensor product of  $N$  vector spaces. A scalar, vector, and matrix correspond to tensors of order zero, one, and two, respectively. One of the key challenges in working with high-order tensors is called the “curse of dimensionality”, where tensors with large dimensionality can have an extremely large number of components, making it difficult to analyze and extract meaningful information from them. *Tensor networks* are powerful techniques for addressing this challenge. A tensor network [14] employs a collection of small tensors, where some or all of their dimensions are contracted according to some pattern, to implicitly represent a high-dimensional tensor. Tensor networks have been originally used in computational quantum physics [23, 22, 24, 21, 20, 19], where low-rank tensor networks can be used efficiently and accurately to represent quantum states and operators based on the area law. Recently, tensor networks are also widely used in simulating quantum computers [11, 25, 18, 17] and neural networks [13].

Tensor network contraction explicitly evaluates the single tensor represented by a given tensor network. When each tensor in the network is dense, tensor network contraction is typically achieved through a sequence of pairwise tensor contractions. This sequence, known as the *contraction path*, is determined by a topological sort of the underlying *contraction tree*. The contraction tree is a rooted binary tree that depicts the complete contraction of the tensor network. In this tree, the leaves correspond to the tensors in the network, and each internal vertex represents the tensor contraction of its two children.

Tensor network contraction has found diverse applications in different fields of research. For instance, in quantum computing, each quantum algorithm can be viewed as a tensor network contraction, making this method a useful tool for simulating quantum computers [11, 25, 18, 17]. In statistical physics, tensor network contraction has been used to evaluate the classical partition function of physical models defined on specific graphs [8]. Tensor network contraction has also been used for counting satisfying assignments of constraint satisfaction problems (#CSPs) [7]. In this approach, an arbitrary #CSP formula is transformed into a tensor network, where its full contraction yields the number of satisfying assignments of that formula.

Contracting tensor networks with arbitrary structure is #P-hard in the general case [3, 16, 1], even when the network represents a scalar. The reason for this is that during the contraction of general tensor networks, intermediate tensors with high orders or large dimension sizes can emerge, leading to a substantial computational cost for precise contraction. Nonetheless, in some applications such as many-body physics, it has been observed that tensor networks built on top of specific models can often be approximately contracted with satisfactory accuracy, without incurring exponential costs [15]. A common approach is to represent or approximate large intermediate tensors as (low-rank) tensor networks, which reduces the memory usage and computational overhead for downstream contractions. Common tensor networks used for approximation include the matrix product states (MPS) and the tree tensor networks (TTN) [20].

Efficient approximate contraction algorithms based on MPSs have been proposed for tensor network contractions defined on regular structures such as the Projected Entangled Pair States (PEPS)

[21, 22, 10, 9], which has a 2D lattice structure. However, these methods are not easily extendable to other general tensor network structures.

Recent works have proposed approximation algorithms for contracting tensor networks with more general graph structures. For example, [6] approximates each intermediate tensor produced during the contraction path as a binary tree tensor network, while [17] approximates each intermediate tensor as an MPS. In [2], each intermediate tensor is also approximated as an MPS, but the system is designed for the specific unbalanced contraction paths and only targets the approximate contraction of tensor networks defined on planar graphs. Another approach proposed in [5] is to perform low-rank approximation on the remaining tensor network after contractions, rather than on the intermediate tensors. The experimental results demonstrate that this framework is more efficient and accurate than [17].

We introduce two approximate tensor network contraction algorithms. First of all, we present a swap-based algorithm named Contracting Arbitrary Tensor Network with Global Ordering (CATN-GO) that can efficiently approximate the contraction of arbitrary tensor networks. Our algorithm builds on the approach outlined in [17], which approximates each intermediate tensor generated during the contraction as an MPS with a bounded rank. When contracting two tensors, the algorithm merges two MPSs, with swaps of adjacent dimensions in the MPS being the bottleneck for complexity.

For a tensor network defined on  $G = (V, E)$ , we prove that the minimum number of swaps required during contraction is lower bounded by the least number of edge crossings in any vertex linear ordering of the tensor network graph, denoted by  $\min_{\sigma} \text{cr}(G, \sigma)$ . A vertex linear ordering  $\sigma : V \rightarrow \{1, \dots, |V|\}$  assigns each vertex a unique number, and two edges with adjacent vertex orders  $(i, j), (k, l)$  cross if  $i < k < j < l$ . Hence, we reduce the problem of finding the minimum number of swaps to the problem of finding a vertex linear ordering that minimizes the number of edge crossings. In addition, for a fixed vertex ordering  $\sigma^V$ , the number of swaps used in CATN-GO equals the lower bound,  $\text{cr}(G, \sigma^V)$ , implying optimality for this metric. Furthermore, CATN-GO includes a dynamic programming algorithm to select the contraction tree under a given vertex ordering. This algorithm aims to minimize the overall computational cost, under the assumption that all MPSs have a uniform rank. The uniform rank assumption makes the problem equivalent to minimizing the total length of the MPSs generated during the contractions and has a time complexity of  $O(|V|^3|E|)$ . Experimental results demonstrate that when contracting tensor networks defined on 3D lattices using the Ising model, our algorithm is more efficient than the algorithm proposed in [17] in terms of speed, and achieves a 5.9X speed-up while maintaining the same accuracy.

We propose another approximate tensor network contraction method named Partitioned Contract. Like similar methods proposed in [6, 17, 2], our algorithm approximates each intermediate tensor as a binary tree tensor network. Compared to previous works, the proposed algorithm has the flexibility to incorporate a larger portion of the environment when performing low-rank approximations. Here, the environment refers to the remaining set of tensors in the network, and low-rank approximations with larger environments can generally provide higher accuracy. In addition, our proposed algorithm includes a cost-efficient density matrix algorithm [12, 4] for approximating a tensor network with a general graph structure into a tree structure. The computational cost of the density matrix algorithm is asymptotically upper-bounded by that of the standard algorithm that uses canonicalization (the process of orthogonalizing all tensors except one in the tensor network). Experimental results indicate that the proposed algorithm outperforms both algorithms proposed in [17] and [2] when considering tensor networks defined on lattices using the Ising model. Specifically, our approach achieves a 9.2X speed-up while maintaining the same level of accuracy.

## References

- [1] J. D. Biamonte, J. Morton, and J. Turner. Tensor network contractions for # SAT. *Journal of Statistical Physics*, 160(5):1389–1404, 2015.
- [2] C. T. Chubb. General tensor network decoding of 2D Pauli codes. *arXiv preprint arXiv:2101.04125*, 2021.
- [3] C. Damm, M. Holzer, and P. McKenzie. The complexity of tensor calculus. *computational complexity*, 11(1-2):54–89, 2002.
- [4] M. Fishman, S. White, and E. Stoudenmire. The ITensor software library for tensor network calculations. *SciPost Physics Codebases*, page 004, 2022.
- [5] J. Gray and G. K. Chan. Hyper-optimized compressed contraction of tensor networks with arbitrary geometry. *arXiv preprint arXiv:2206.07044*, 2022.
- [6] A. Jermyn. Automatic contraction of unstructured tensor networks. *SciPost Physics*, 8(1):005, 2020.
- [7] S. Kourtis, C. Chamon, E. Mucciolo, and A. Ruckenstein. Fast counting with tensor networks. *SciPost Physics*, 7(5):060, 2019.
- [8] M. Levin and C. P. Nave. Tensor renormalization group approach to two-dimensional classical lattice models. *Physical review letters*, 99(12):120601, 2007.
- [9] M. Lubasch, J. I. Cirac, and M.-C. Banuls. Algorithms for finite projected entangled pair states. *Physical Review B*, 90(6):064425, 2014.
- [10] M. Lubasch, J. I. Cirac, and M.-C. Banuls. Unifying projected entangled pair state contractions. *New Journal of Physics*, 16(3):033014, 2014.
- [11] L. Ma and C. Yang. Low rank approximation in simulations of quantum algorithms. *Journal of Computational Science*, page 101561, 2022.
- [12] Tensornetwork.org contributors. Density matrix algorithm - tensornetwork.org. 2021.
- [13] A. Novikov, D. Podoprikhin, A. Osokin, and D. P. Vetrov. Tensorizing neural networks. In *Advances in neural information processing systems*, pages 442–450, 2015.
- [14] R. Orús. A practical introduction to tensor networks: Matrix product states and projected entangled pair states. *Annals of Physics*, 349:117–158, 2014.
- [15] R. Orús. Tensor networks for complex quantum systems. *Nature Reviews Physics*, 1(9):538–550, 2019.
- [16] B. O’Gorman. Parameterization of tensor network contraction. In *14th Conference on the Theory of Quantum Computation, Communication and Cryptography*, 2019.
- [17] F. Pan, P. Zhou, S. Li, and P. Zhang. Contracting arbitrary tensor networks: general approximate algorithm and applications in graphical models and quantum circuit simulations. *Physical Review Letters*, 125(6):060503, 2020.

- [18] Y. Pang, T. Hao, A. Dugad, Y. Zhou, and E. Solomonik. Efficient 2D tensor network simulation of quantum systems. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–14. IEEE, 2020.
- [19] U. Schollwöck. The density-matrix renormalization group. *Reviews of modern physics*, 77(1):259, 2005.
- [20] Y.-Y. Shi, L.-M. Duan, and G. Vidal. Classical simulation of quantum many-body systems with a tree tensor network. *Physical review A*, 74(2):022320, 2006.
- [21] F. Verstraete and J. I. Cirac. Renormalization algorithms for quantum-many body systems in two and higher dimensions. *arXiv preprint cond-mat/0407066*, 2004.
- [22] F. Verstraete, V. Murg, and J. I. Cirac. Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems. *Advances in physics*, 57(2):143–224, 2008.
- [23] G. Vidal. Efficient classical simulation of slightly entangled quantum computations. *Physical review letters*, 91(14):147902, 2003.
- [24] S. R. White. Density matrix formulation for quantum renormalization groups. *Physical review letters*, 69(19):2863, 1992.
- [25] Y. Zhou, E. M. Stoudenmire, and X. Waintal. What limits the simulation of quantum computers? *Physical Review X*, 10(4):041038, 2020.

# Backward stability of $s$ -step GMRES

*Yuxin Ma, Erin Carson*

## Abstract

Communication, i.e., data movement, is a critical bottleneck for the performance of classical Krylov subspace method solvers on modern computer architectures. Variants of these methods which avoid communication have been introduced, which, while equivalent in exact arithmetic, can be unstable in finite precision. In this work, we address the backward stability of  $s$ -step GMRES, also known as communication-avoiding GMRES. We present a framework for simplifying the analysis of  $s$ -step GMRES, which includes standard GMRES ( $s = 1$ ) as a special case, by isolating the effects of rounding errors in the QR factorization and the solution of the least squares problem. Using this framework, we analyze  $s$ -step GMRES with popular block orthogonalization methods: block modified Gram–Schmidt and reorthogonalized block classical Gram–Schmidt algorithms.

An example illustrates the resulting instability of  $s$ -step GMRES when paired with the classical  $s$ -step Arnoldi process and shows the limitations of popular strategies for resolving this instability. To address this issue, we propose a modified Arnoldi process that allows for much larger block size  $s$  while maintaining satisfactory accuracy, as confirmed by our numerical experiments.

PS: I am a postdoc researcher in Charles University. My interests are about numerical analysis and high performance computing, i.e., mixed precision algorithms, communication-avoiding algorithms, and finite precision analysis.

# Sign Characteristic in the Inverse Problem for Hermitian Matrix Polynomials

D. Steven Mackey, F. Tisseur

## Abstract

For any matrix polynomial  $P(\lambda) = \sum_{j=0}^d \lambda^j A_j$ ,  $A_j \in \mathbb{F}^{m \times n}$ , there is a multiset<sup>1</sup>  $\mathcal{S}$  comprising the key structural data of  $P$  relevant for applications. This structural data consists of the elementary divisors (finite and infinite) of  $P$ , together with its left and right minimal indices. The basic inverse problem for matrix polynomials, then, consists of two parts:

Given a multiset  $\mathcal{S}$  of structural data and a choice of degree  $d$ ,

- (a) does there exist any matrix polynomial  $P$  of degree  $d$  (of any size  $m \times n$ ) whose structural data multiset is exactly  $\mathcal{S}$ ?
- (b) if such a matrix polynomial  $P$  exists, can one be explicitly constructed, especially in such a way that the structural data in  $\mathcal{S}$  is transparently visible in that  $P$  (e.g., in the spirit of the Kronecker canonical form), or at least can be exactly recovered from  $P$  without any numerical computation, only combinatorial manipulations?

Analogous questions can be immediately posed for matrix polynomials restricted to various classes of structured matrix polynomials important in applications, e.g., palindromic, alternating, or Hermitian matrix polynomials.

Much progress has been made on these questions in the last decade, although much remains to be understood. Some of the earliest results on these questions were for general quadratic [3] and quadratic palindromic matrix polynomials [4], where Kronecker-like quasi-canonical forms were found. In the 2015 paper [5], De Terán, Dopico, and Van Dooren completely solved the existence question (in part (a)) for general matrix polynomials of arbitrary degree over any infinite field  $\mathbb{F}$ , giving simple necessary and sufficient conditions for realizability of a structural data multiset. However, the construction used to prove sufficiency in [5] produces a matrix polynomial realization in which the given data  $\mathcal{S}$  is usually not transparently visible anymore. Recent work in [8] has substantially remedied this deficiency, giving a complete solution for part (b) of the inverse problem for general matrix polynomials, but only over algebraically closed fields.

This talk, however, focuses more specifically on the inverse problem for Hermitian matrix polynomials. This class of matrix polynomials has an additional element in its structural data multiset, the *sign characteristic*, that is relevant for both theory and applications, playing an important role in the bifurcations of dynamical systems, as well as the behavior of eigenvalues under structured perturbations. First recognized and investigated in the seminal paper [7] as an important invariant of Hermitian polynomials under unimodular congruence, the sign characteristic consists of a plus or minus sign attached to each elementary divisor associated with a real or infinite eigenvalue. In [7], the authors considered sign characteristic only for regular Hermitian polynomials with no eigenvalues at  $\infty$ . But more recently, the concepts and results in [7] concerning sign characteristic have been extended in [12] to include all Hermitian polynomials, both regular and singular, with or without eigenvalues at  $\infty$ .

---

<sup>1</sup>A *multiset* is a set with repetitions allowed, i.e., a “set with multiplicities”. [9, p.454]

Now for general Hermitian matrix polynomials, there are several well-known (pairing) constraints on elementary divisors and minimal indices; e.g., for any Hermitian polynomial  $H(\lambda)$ , the multiset of left minimal indices of  $H$  is identical to the multiset of the right minimal indices of  $H$ . But are there any constraints on how *signs* can be attached to real and infinite elementary divisors? For Hermitian pencils it is known that there are no constraints on signs; any real or infinite elementary divisor may have any sign, in any combination with the signs of all of the pencil's other elementary divisors. This follows immediately from the canonical form for Hermitian pencils found in [10].

For higher degree Hermitian polynomials, though, there is one known constraint on signs. This is the so-called *signature constraint*, first proved in [7] for regular Hermitian polynomials without any infinite elementary divisors, and extended to all Hermitian polynomials in [12]. For even degree Hermitian polynomials, the signature constraint says that the sum of the signs attached to all of the odd degree real and infinite elementary divisors is zero. The constraint is somewhat more complicated to state when the Hermitian polynomial  $H(\lambda)$  has odd degree — in this case the sum of the signs for odd degree elementary divisors of real eigenvalues together with those of the even degree elementary divisors of an infinite eigenvalue must equal the signature of the leading coefficient of  $H$  (hence the name of the condition).

Is the signature constraint the only condition on the sign characteristic? In the quadratic case, this has recently been shown to be true [11]. But for all degrees higher than 2, this is not so. Starting with degree 3, there are additional conditions on the sign characteristic that are independent of the signature constraint. Unfortunately, the techniques used in [11] to solve the quadratic Hermitian inverse problem do not easily extend to higher degrees, even to degree 3; there is a combinatorial explosion of cases that make this approach intractable. Thus it is reasonable to restrict the problem to something more manageable, yet still of some significance. Motivated by the recent investigations into the generic behavior in various structured classes of matrix polynomials [1, 2], it is sensible to restrict the question to Hermitian matrix polynomials in which *all eigenvalues (including  $\infty$ ) are simple*. It is in this class that one can expect to find the structural data multisets that are generic for all Hermitian polynomials. Going forward, then, we make the simple eigenvalue assumption.

When a Hermitian polynomial has all simple eigenvalues, the signs attached to these eigenvalues can be naturally ordered to form a *sign sequence*. For an  $n \times n$  Hermitian polynomial of degree  $d$  with the maximum number ( $dn$ ) of simple real eigenvalues, there are  $2^{dn}$  conceivable sign sequences, most of which cannot be realized by any degree  $d$  Hermitian polynomial. Perhaps surprisingly, it is the *order* of the signs in a sign sequence that turns out to be crucial for Hermitian realizability, but in ways that are not immediately apparent. For example, consider a  $3 \times 3$  Hermitian matrix polynomial of degree 4, with 12 simple real eigenvalues. It is easy to see that the sign sequence  $+++++----$  satisfies the signature constraint for degree 4, but it can be shown that this sign sequence cannot be realized by any  $3 \times 3$  Hermitian polynomial of degree 4.

Is it possible to determine exactly which sign sequences are realizable and which are not? And what role, if any, the degree might play in the story? This talk answers these questions, introducing several new constraints on signs beyond the signature constraint, as well as an underlying group of symmetries acting on the collection of all sign sequences, that sheds additional light on this issue. The result of this analysis is a *characterization* of the sign sequences that are realizable by a Hermitian matrix polynomial with all simple eigenvalues, as well as a solution of the inverse problem for this class of Hermitian polynomials.

This characterization also allows us to get a quantitative sense for just how few of the possible sign sequences are actually realizable. Consider again the scenario above, i.e.,  $n \times n$  Hermitian polynomials of degree  $d$  with the maximum number ( $dn$ ) of simple real eigenvalues, in particular

the special case of  $n = 2$  with any even degree  $d \geq 4$ . In this case the fraction of the  $2^{2d}$  possible sign sequences that satisfy the signature constraint is asymptotically  $O(\frac{1}{\sqrt{\pi d}})$  as  $d \rightarrow \infty$ . By contrast, however, the characterization implies that the fraction of possible sign sequences that are actually Hermitian realizable is asymptotically  $O(\frac{1}{2^d})$  as  $d \rightarrow \infty$ . A reasonable conjecture is that this asymptotic behavior will persist for other values of  $n$  and  $d$ . But this does at least show that the number of realizable sign sequences can not only be exponentially smaller than the number of all possible sign sequences, it can also even be exponentially smaller than the number of sign sequences that satisfy the signature constraint. Hence we see that the signature constraint is by itself very far from capturing the property of Hermitian realizability of sign sequences.

Finally, it is worth noting that the key tool for proving the necessity of these new constraints on the sign characteristic is an old theorem of Rellich on analytic decompositions of analytic Hermitian matrix-valued functions (see [12, Thm.2.1]). On the other hand, sufficiency of these new constraints is proved using the new technique of *product realizations* of matrix polynomials [6].

## References

- [1] F. DE TERÁN, A. DMYTRYSHYN AND F. M. DOPICO, Generic symmetric matrix polynomials with bounded rank and fixed odd grade. *SIAM J. Matrix Anal. Appl.*, 41 (2020), p.1033–1058.
- [2] F. DE TERÁN, A. DMYTRYSHYN AND F. M. DOPICO, Generic eigenstructures of Hermitian pencils. *SIAM J. Matrix Anal. Appl.*, 45 (2024), p.260–283.
- [3] F. DE TERÁN, F.M. DOPICO, AND D.S. MACKEY. A quasi-canonical form for quadratic matrix polynomials. In preparation.
- [4] F. DE TERÁN, F.M. DOPICO, D.S. MACKEY, AND V. PEROVIĆ. Quadratic realizability of palindromic matrix polynomials. *Linear Algebra Appl.*, 567 (2019), p.202–262.
- [5] F. DE TERÁN, F.M. DOPICO, AND P. VAN DOOREN. Matrix polynomials with completely prescribed eigenstructure. *SIAM J. Matrix Anal. Appl.*, 36 (2015), p.302–328.
- [6] F.M. DOPICO, D.S. MACKEY, AND P. VAN DOOREN. Product realizations of structural data for matrix polynomials. In preparation.
- [7] I. GOHBERG, P. LANCASTER, AND L. RODMAN. Spectral analysis of self-adjoint matrix polynomials. *Annals of Math.*, 112, no.1 (1980), p.33–71.
- [8] R. HOLLISTER. *Inverse Problems for Polynomial and Rational Matrices*. PhD Thesis, Western Michigan University, 2020.
- [9] D.E. KNUTH. *The Art of Computer Programming. Vol. 2: Seminumerical Algorithms*. Addison-Wesley, Reading, Massachusetts, 1981, 2nd Ed.
- [10] P. LANCASTER AND L. RODMAN. Canonical forms for Hermitian matrix pairs under strict equivalence and congruence. *SIAM Rev*, 47, no.3 (2005), p.407–443.
- [11] D.S. MACKEY AND F. TISSEUR, The Hermitian quadratic realizability problem. In preparation.
- [12] V. MEHRMANN, V. NOFERINI, F. TISSEUR, AND H. XU, On the sign characteristics of Hermitian matrix polynomials. *Linear Algebra Appl.*, 511 (2016), pp. 328–364.

# Solving Generalized Lyapunov Equations with guarantees: application to the Reduction of Linear Switched Systems.

*Mattia Manucci, Benjamin Unger*

Abstract

We deal with the efficient and certified approximation of the *generalized Lyapunov equation* (GLEs)

$$\mathbf{A}\mathbf{X} + \mathbf{X}\mathbf{A}^\top + \sum_{j=1}^M \left( \mathbf{N}_j \mathbf{X} \mathbf{N}_j^\top \right) + \mathbf{B}\mathbf{B}^\top = \mathbf{0}, \quad (1)$$

where  $\mathbf{A}, \mathbf{N}_j \in \mathbb{R}^{n \times n}$ ,  $\mathbf{A}$  is Hurwitz, i.e., its spectrum is contained in the open left-half complex plane, and  $\mathbf{B} \in \mathbb{R}^{n \times m}$  with  $m$  typically much smaller than  $n$ . GLEs with these features naturally arise in the context of *model order reduction* (MOR) of bilinear control systems [2, 5] and linear parameter-varying systems as well as in the context of stochastic differential equations for stability analysis [4]. For switched linear systems of the form

$$\Sigma_q \quad \begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}_{q(t)}\mathbf{x}(t) + \mathbf{B}_{q(t)}\mathbf{u}(t), & \mathbf{x}(t_0) = \mathbf{0}, \\ \mathbf{y}(t) = \mathbf{C}_{q(t)}\mathbf{x}(t), \end{cases} \quad (2)$$

the authors of [6] introduced a balancing-based MOR method that requires the solution of certain GLEs. In (2),  $q: \mathbb{R} \rightarrow \mathcal{J} := \{1, \dots, M\}$  is the external switching signal, which we assume to be an element of the set of allowed switching signals

$$\mathcal{S} := \{q: \mathbb{R} \rightarrow \mathcal{J} \mid q \text{ is right continuous with locally finite number of jumps}\}. \quad (3)$$

The symbols  $\mathbf{x}(t) \in \mathbb{R}^n$ ,  $\mathbf{u}(t) \in \mathbb{R}^m$ , and  $\mathbf{y}(t) \in \mathbb{R}^p$  denote the *state*, the controlled *input*, and the measured *output*, respectively. The system matrices  $\mathbf{A}_j \in \mathbb{R}^{n \times n}$ ,  $\mathbf{B}_j \in \mathbb{R}^{n \times m}$ , and  $\mathbf{C}_j \in \mathbb{R}^{p \times n}$  correspond to the *ordinary differential equation* (ODE) active in mode  $j \in \mathcal{J}$ . Typically one refers to (2) as the *full-order model* (FOM). Sample applications of switched systems include robot manipulators, traffic management, automatic gear shifting, and power systems; see for instance [3] and the references therein.

If (2) has to be evaluated repeatedly, for instance, in a simulation context for different inputs or switching signals, or if matrix equalities or inequalities in the context of synthesis have to be solved, then a large dimension  $n$  of the state renders this a computationally expensive task. In such scenarios, one can rely on MOR and replace (2) by the *reduced-order model* (ROM)

$$\tilde{\Sigma}_q \quad \begin{cases} \dot{\tilde{\mathbf{x}}}(t) = \tilde{\mathbf{A}}_{q(t)}\tilde{\mathbf{x}}(t) + \tilde{\mathbf{B}}_{q(t)}\mathbf{u}(t), & \tilde{\mathbf{x}}(t_0) = \mathbf{0}, \\ \tilde{\mathbf{y}}(t) = \tilde{\mathbf{C}}_{q(t)}\tilde{\mathbf{x}}(t), \end{cases} \quad (4)$$

with  $\tilde{\mathbf{A}}_j \in \mathbb{R}^{r \times r}$ ,  $\tilde{\mathbf{B}}_j \in \mathbb{R}^{r \times m}$ , and  $\tilde{\mathbf{C}}_j \in \mathbb{R}^{p \times r}$ , and  $r \ll n$ . In many cases, see for instance [1], the reduced system matrices are obtained via Petrov–Galerkin projection, i.e., one constructs matrices  $\mathbf{V}, \mathbf{W} \in \mathbb{R}^{n \times r}$  and then defines

$$\tilde{\mathbf{A}}_j := \mathbf{W}^\top \mathbf{A}_j \mathbf{V}, \quad \tilde{\mathbf{B}}_j := \mathbf{W}^\top \mathbf{B}_j, \quad \tilde{\mathbf{C}}_j := \mathbf{C}_j \mathbf{V}. \quad (5)$$

The goal of MOR is thus to derive in a computationally efficient and robust way the matrices  $\mathbf{W}, \mathbf{V}$  such that the error  $\mathbf{y} - \tilde{\mathbf{y}}$  is small in some given norm. One way to do so, originally presented in [6],

is to solve opportune defined GLEs to obtain the projection matrices  $\mathbf{W}, \mathbf{V}$  and thus the reduced system (5). Therefore, solving efficiently large-scale generalized Lyapunov equation becomes crucial for MOR. More in detail the MOR algorithm from [6] proceeds in two steps. First, we have to define the matrices  $\mathbf{A} := \mathbf{A}_1$  and  $\mathbf{N}_j := \mathbf{A}_j - \mathbf{A}_1$  for  $j = 1, \dots, M$  and solve the GLEs

$$\mathbf{A}\mathcal{P} + \mathcal{P}\mathbf{A}^\top + \sum_{j=1}^M \left( \mathbf{N}_j \mathcal{P} \mathbf{N}_j^\top + \mathbf{B}_j \mathbf{B}_j^\top \right) = \mathbf{0}, \quad (6a)$$

$$\mathbf{A}^\top \mathcal{Q} + \mathcal{Q}\mathbf{A} + \sum_{j=1}^M \left( \mathbf{N}_j^\top \mathcal{Q} \mathbf{N}_j + \mathbf{C}_j^\top \mathbf{C}_j \right) = \mathbf{0}. \quad (6b)$$

Note that the matrix equations in (6) are of the form (1) by defining  $\mathbf{B} := [\mathbf{B}_1, \dots, \mathbf{B}_M]$  for (6a) and  $\mathbf{C} := [\mathbf{C}_1^\top, \dots, \mathbf{C}_M^\top]$ , taking the transport on the other matrices for (6b). The symmetric and positive semi-definite solutions  $\mathcal{P}, \mathcal{Q} \in \mathbb{R}^{n \times n}$  are referred to as the Gramians of (2). Second, let  $\mathcal{P} = \mathbf{S}\mathbf{S}^\top$  and  $\mathcal{Q} = \mathbf{R}\mathbf{R}^\top$  and compute the *singular value decomposition* (SVD) of the product of the Gramians factors

$$\mathbf{S}^\top \mathbf{R} = [\mathbf{U}_1, \mathbf{U}_2] \begin{bmatrix} \Sigma_1 & \mathbf{0} \\ \mathbf{0} & \Sigma_2 \end{bmatrix} [\mathbf{V}_1, \mathbf{V}_2]^\top, \quad (7)$$

and the projection matrices  $\mathbf{V}$  and  $\mathbf{W}$  are obtained via

$$\mathbf{V} = \mathbf{S}\mathbf{U}_1 \Sigma_1^{-1/2} \quad \text{and} \quad \mathbf{W} = \mathbf{R}\mathbf{V}_1 \Sigma_1^{-1/2}. \quad (8)$$

This procedure is denoted as square-root balanced truncation (see [1, Sec. 7.3]). The use of the solutions of (6) as system Gramians is justified by [6, Thm. 3], where the authors show that the image of  $\mathcal{P}$  and  $\mathcal{Q}$  encode the reachability set and observability set of the switched system (2).

**Main contributions:** To deal with the large-scale setting, we apply the stationary algorithm from [7] in combination with a subspace projection framework [8] to solve GLEs. We emphasize that this is a common strategy in the literature when dealing with GLEs. Our first contribution is the derivation of efficiently computable error estimates such that for any prescribed user tolerance `tol` an approximation  $\tilde{\mathbf{X}}$  of (1) with guaranteed bound  $\|\mathbf{X} - \tilde{\mathbf{X}}\|_2 \leq \text{tol}$  can be computed. Second, we show how the numerical error introduced in approximating (1) may deteriorate the quality and the stability of the ROM (4). This motivates us to propose a novel strategy that, by relying on the error certification provided by our algorithm, ensures stability and error certification of the MOR system. Finally, the results are validated through a synthetic example and a switched system arising from a parametric *partial differential equation* (PDE).

## References

- [1] A. C. ANTOULAS. *Approximation of large-scale dynamical systems*. Adv. Des. Control. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2005.
- [2] P. BENNER AND T. DAMM. Lyapunov equations, energy functionals, and model order reduction of bilinear and stochastic systems. *SIAM J. Cont. Optim.*, 49(2):686–711, 2011.
- [3] D. CHENG. Stabilization of planar switched systems. *Systems Control Lett.*, 51:79–88, 2004.
- [4] T. DAMM AND D. HINRICHSEN. Newton’s method for a rational matrix equation occurring in stochastic control. *Linear Algebra and its Applications*, 332-334:81–109, 2001.

- [5] W. S. GRAY AND J. MESKO. Energy functions and algebraic Gramians for bilinear systems. *IFAC Proceedings Volumes*, 31(17):101–106, 1998.
- [6] I. PONTES DUFF, S. GRUNDEL, AND P. BENNER. New Gramians for switched linear systems: Reachability, observability, and model reduction. *IEEE Trans. Automat. Control*, 65(6):2526–2535, 2020.
- [7] S. D. SHANK, V. SIMONCINI, AND D. B. SZYLD. Efficient low-rank solution of generalized Lyapunov equations. *Numer. Math.*, 134(2):327–342, 2016.
- [8] V. SIMONCINI. Computational methods for linear matrix equations. *SIAM Rev.*, 58(3):377–441, 2016.

# Inverse Eigenvalue Difference Problems Arising in Quantum Sensing

*Boaz Ilan, Roummel Marcia, Michael Scheibner, and Kyle Wright*

## Abstract

This research is motivated by the emerging field of quantum sensing, which facilitates high-resolution sensing of gravitation, acoustic waves, and electromagnetic fields [1, 3, 4]. The discrete energy levels of coupled quantum dots (QDs) can be represented as eigenvalues of a quantum Hamiltonian matrix, whose entries are defined as polynomials of an applied electric field [2, 5]. Our aim is to recover the coefficients of these polynomials, which correspond to intrinsic physical constants, such as spin-coupling strength. The eigenvalue differences can be obtained from experimental measurements for varying electric field values. Standard inverse eigenvalue problems (IEP) seek to recover a constant matrix from the eigenvalues. In contrast, here the matrix elements are functions of a “tunable” parameter (the applied electric field) and only the *differences* between the eigenvalues are known. We formulate this as an inverse eigenvalue difference problem (IEDP).

**Problem formulation.** The steady-state energy levels of coupled QDs correspond to eigenvalues of quantum Hamiltonians. Specifically, the ground state of this system can be described by a  $3 \times 3$  real symmetric matrix [2] of the form

$$G(F) = \begin{bmatrix} g_1(F) & y_0 & y_1 \\ y_0 & g_2(F) & y_2 \\ y_1 & y_2 & g_3(F) \end{bmatrix}, \quad (1)$$

where the diagonal elements depend quadratically on the applied electric field,  $F \in \mathbb{R}$ , as

$$g_i(F) = \alpha_i + \beta_i F + \gamma_i F^2, \quad i = 1, 2, 3, \quad (2)$$

where the coefficients  $\{\alpha_i, \beta_i, \gamma_i\}$  are real. We note that the assumption that the off-diagonal elements are independent of  $F$  is a good approximation for weak electric fields and for weak tunnel coupling between the QDs. We shall further assume that  $y_1 = y_0$  and  $y_2 = 0$ , which corresponds to symmetries between QDs.

Since  $G(F)$  is symmetric, its eigenvalues are real. We denote its eigenvalues by  $\{\xi_1(F), \xi_2(F), \xi_3(F)\}$ . The physical measurements can be used to determine the *differences* between the eigenvalues of  $G$ . Thus, the measured data is provided over a set of  $n$  values of  $F \in \mathbb{R}$ , denoted by  $\{F_k\}_{k=1}^n$ . The eigenvalue differences are denoted by

$$D_{2,1}(F) \equiv \xi_2(F) - \xi_1(F), \quad (3)$$

$$D_{3,1}(F) \equiv \xi_3(F) - \xi_1(F), \quad (4)$$

$$D_{3,2}(F) \equiv \xi_3(F) - \xi_2(F). \quad (5)$$

Note that  $D_{3,2}(F) = D_{3,1}(F) - D_{2,1}(F)$ . Hence, we consider the measured dataset to be

$$M = \{F_k, D_{2,1}(F_k), D_{3,1}(F_k)\}_{k=1}^n. \quad (6)$$

Our objective is to recover the coefficients that define  $G(F)$ . We can prove that, without loss of generality, one can set  $g_1(F) = 0$ . This has the effect of eliminating an arbitrary shift in the

diagonal elements and in the eigenvalues of  $G(F)$  that would satisfy the dataset  $M$ . We shall do so henceforth and redefine  $G$  as

$$G(F) = \begin{bmatrix} 0 & y_0 & y_0 \\ y_0 & g_2(F) & 0 \\ y_0 & 0 & g_3(F) \end{bmatrix}. \quad (7)$$

We denote the vector of coefficients as  $\mathbf{p} = [y_0, \alpha_2, \beta_2, \gamma_2, \alpha_3, \beta_3, \gamma_3] \in \mathbb{R}^7$ . In this work, we seek to solve the following:

### Inverse Eigenvalue Difference Problem (IEDP)

Given the eigenvalue difference dataset  $M$  in (6), find coefficients  $\mathbf{p}$ , such that  $G(F)$  in (7) generates  $M$ .

In particular, our work addresses two inter-related questions:

1. What optimization approach is efficient for solving this IEDP, especially in the presence of noisy data?
2. What domain knowledge can we utilize to improve the efficacy of the proposed approach?

## References

- [1] Stefano Pirandola, B Roy Bardhan, Tobias Gehring, Christian Weedbrook, and Seth Lloyd. Advances in photonic quantum sensing. *Nature Photonics*, 12(12):724–733, 2018.
- [2] Michael Scheibner, MF Doty, Ilya V Ponomarev, Allan S Bracker, Eric A Stinaff, VL Koronev, TL Reinecke, and Daniel Gammon. Spin fine structure of optically excited quantum dot molecules. *Physical Review B*, 75(24):245318, 2007.
- [3] Ben Stray, Andrew Lamb, Aisha Kaushik, Jamie Vovrosh, Anthony Rodgers, Jonathan Winch, Farzad Hayati, Daniel Boddice, Artur Stabrawa, Alexander Niggebaum, et al. Quantum sensing for gravity cartography. *Nature*, 602(7898):590–594, 2022.
- [4] SP Walborn, AH Pimentel, L Davidovich, and RL de Matos Filho. Quantum-enhanced sensing from hyperentanglement. *Physical Review A*, 97(1):010301, 2018.
- [5] Kyle Wright, Roummel Marcia, Michael Scheibner, and Boaz Ilan. Parameterized inverse eigenvalue problem for quantum sensing. In *2023 IEEE 9th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pages 351–355. IEEE, 2023.

# On the quasiseparability of the solution of continuous-time Riccati equations with quasiseparable coefficients

*Stefano Massei, Luca Saluzzi*

## Abstract

Solving large-scale continuous-time algebraic Riccati equations (CARE) of the form

$$A^\top X + XA - XFX + Q = 0, \quad (1)$$

is a significant challenge in various control theory applications. When the numerical range  $\mathcal{W}(A)$  of  $A$  is in the left part of the complex plane, and  $F, Q$  are low-rank matrices the stabilizing solution  $X$  is numerically low-rank. The latter property can be shown by rewriting (1) as a Sylvester equation with low-rank right-hand side, and by applying singular values inequalities for the solution of this type of equations [1]. In this scenario, one can apply Krylov subspace projection methods or ADI-type methods to efficiently get approximate solutions.

This work is concerned with the non standard large-scale case where the coefficients  $A, F, Q$  are full rank quasiseparable matrices, i.e., all their offdiagonal blocks are low-rank. Within this setting, we provide decay bounds for the singular values of the offdiagonal blocks of  $X$ , justifying the approximability of  $X$  by a quasiseparable matrix. To derive these bounds we relate a generic offdiagonal block with the solution of a Sylvester equation with low-rank right-hand side; note that, this can not be trivially obtained by moving the term  $Q - XFX$  to the right-hand side of (1). Our results establish a link between the rate of decay of the offdiagonal singular values and certain rational approximation problems, known as Zolotarev problems, involving the set  $\mathcal{W}(L^{-1}AL)$ , where  $F = LL^\top$  is a Cholesky factorization of  $F$ . More explicitly, for a generic offdiagonal submatrix  $M$  of  $X$ , we get inequalities of the form

$$\frac{\sigma_{ht+r}(M)}{\|X\|_2} \leq \kappa \cdot Z_h(W(L^{-1}AL), -W(L^{-1}AL)), \quad h = 1, 2, 3, \dots,$$

where the shift parameters  $t, r \geq 0$  depend only on the quasiseparable ranks of the coefficients of the CARE,  $\kappa$  is a constant including the condition number of  $F$ , and  $Z_h$  indicates the optimal value of the aforementioned Zolotarev problem. When  $W(L^{-1}AL) \subseteq \mathbb{C}^-$ , the quantity  $Z_h(W(L^{-1}AL), -W(L^{-1}AL))$  decays rapidly as  $h$  increases.

Quite interestingly, the rank of the offdiagonal submatrices of  $X$  are linked to the *tensor train ranks* (TT ranks) of the tensor train representation of the value function

$$V(\mathbf{y}) := \mathbf{y}^T X \mathbf{y}$$

associated with the solution of the CARE [3]. As a byproduct of our analysis, we improve and enlarge the scope of existing upper bounds for the numerical TT ranks of  $V(\mathbf{y})$  [2, Theorem 3.1].

From the algorithmic view point, we propose two fast Riccati solvers: the first one applies to CAREs with quasiseparable coefficients, and the second one specific to the banded case. The former method exploits the representation of  $A, F$ , and  $Q$ , in the *hierarchically semiseparable format* (HSS) [7], and is based on a divide-and-conquer scheme, similarly to other recent solvers for matrix equations with hierarchically low-rank coefficients [4, 5]. The method for the banded coefficients case aims at providing a sparse approximation of  $X$ , by means of an inexact Newton-Kleinman iteration

(NK) combined with a thresholding mechanism that keeps under control the level of sparsity of the iterates. Under reasonable assumptions, both algorithms have at most a linear-logarithmic complexity; when the NK iterate remains sufficiently banded, the second procedure provides a significant speed up thanks to the use of sparse arithmetic.

The content of this talk is based on [6].

## References

- [1] B. Beckermann, and A. Townsend. “Bounds on the singular values of matrices with displacement structure.” SIAM Review 61.2 (2019): 319-344.
- [2] S. Dolgov, and B. Khoromskij. “Two-level QTT-Tucker format for optimized tensor calculus.” SIAM Journal on Matrix Analysis and Applications 34.2 (2013): 593-623.
- [3] V. Kazeev, O. Reichmann, and Ch. Schwab. “Low-rank tensor structure of linear diffusion operators in the TT and QTT formats.” Linear Algebra and its Applications 438.11 (2013): 4204-4221.
- [4] D. Kressner, S. Massei, and L. Robol. “Low-rank updates and a divide-and-conquer method for linear matrix equations”. SIAM Journal on Scientific Computing, 41.2 (2019): A848-A876.
- [5] D. Kressner, P. Kürschner, S. Massei. “Low-rank updates and divide-and-conquer methods for quadratic matrix equations”. Numerical Algorithms, 84.2 (2020): 717-741.
- [6] S. Massei, and L. Saluzzi. “On the data-sparsity of the solution of Riccati equations with applications to feedback control”. arXiv preprint arXiv:2408.16569 (2024).
- [7] J. Xia, S. Chandrasekaran, M. Gu, and X. Li. “Fast algorithms for hierarchically semiseparable matrices.” Numerical Linear Algebra with Applications 17.6 (2010): 953-976.

# Shift-and-invert Arnoldi for singular eigenvalue problems

Karl Meerbergen, Zhijun Wang

## Abstract

The solution of the regular  $n \times n$  generalized eigenvalue problem  $Ax = \lambda Bx$  is pretty well understood. This is not so for singular pencils, i.e., pencils for which  $A - \lambda B$  is singular for any  $\lambda \in \mathbb{C} \cup \{\infty\}$ . We say that  $\lambda$  is an eigenvalue iff the rank of  $A - \lambda B$  is below the normal rank, i.e., the maximum rank of  $A - \sigma B$  for  $\sigma \in \mathbb{C} \cup \{\infty\}$ .

The staircase method separates the regular and singular parts of the pencil using the Kronecker canonical form. The QZ method can also perform such a separation but may suffer from numerical instabilities. Recently, a rank perturbation was proposed [1][2] which is related to a bordered regular pencil. In this talk, we use a border of the form

$$\begin{bmatrix} A - \lambda B & W \\ V^T & 0 \end{bmatrix}$$

where  $W$  and  $V$  are chosen so that both

$$\begin{bmatrix} A - \lambda B & W \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} A - \lambda B \\ V^T \end{bmatrix}$$

have rank  $n$  for all  $\lambda$ . This bordered eigenvalue problem has a spurious infinite eigenvalue, and may have spurious finite eigenvalue too, but for every eigenvector a simple test can be performed to check whether the eigenvalue is true or spurious.

The aim is to solve the bordered eigenvalue problem using the shift-and-invert Arnoldi method. We employ a special inner product and implicit restarting to reduce the impact of the infinite eigenvalue. To determine a suitable  $V$  and  $W$  we propose a rank revealing LU factorization that should enable computations for large sparse problems. We illustrate the algorithm and the theory for problems arising from multiparameter eigenvalue problems, rectangular pencils and singular quadratic eigenvalue problems.

## References

- [1] M. E. Hochstenbach, C. Mehl, and B. Plestenjak. Solving singular generalized eigenvalue problems by a rank-completing perturbation. *SIAM Journal on Matrix Analysis and Applications*, 40(3):1022–1046, January 2019.
- [2] M. E. Hochstenbach, C. Mehl, and B. Plestenjak. Solving singular generalized eigenvalue problems. part ii: Projection and augmentation. *SIAM Journal on Matrix Analysis and Applications*, 44(4):1589–1618, October 2023.

# Regularization and stabilization of port-Hamiltonian descriptor systems via output feedback

Volker Mehrmann, Delin Chu

## Abstract

The structure preserving stabilization of (possibly non-regular) linear port-Hamiltonian descriptor (pHDAE) systems by output feedback is discussed. While for general descriptor systems the characterization when there exist output feedbacks that lead to an asymptotically stable closed loop system is a very hard and partially open problem, for systems in pHDAE representation this problem can be completely solved. Necessary and sufficient conditions are presented that guarantee that there exist a proportional output feedback such that the resulting closed-loop port-Hamiltonian descriptor system is (robustly) asymptotically stable. For this it is also necessary that the output feedback also makes the problem regular and of index at most one. A complete characterization when this is possible is presented as well.

## References

- [1] D. Chu and V. Mehrmann *Stabilization of linear Port-Hamiltonian Descriptor Systems via Output Feedback*, <http://arxiv.org/abs/2403.18967>

# Exploiting mathematical structures in spectral imaging to accelerate experiments and improve iterative reconstructions

*Maike Meier, Hussam Al Daas, Boris Shustein, Lorenzo Lazzarino, and Paul Quinn*

## Abstract

Spectral imaging covers a range of techniques that aim to reconstruct the chemical state and structural properties of materials. Samples are simultaneously imaged in space, using methods such as microscopy, ptychography, or tomography, as well as resolved in the spectral dimension, giving information about chemical composition. Long acquisition times, sample degradation, and low signal-to-noise ratios plague spectral imaging; what is needed are ways to get more out of the data with fewer experiments.

Although a spectral imaging dataset is necessarily a three-dimensional tensor, its underlying dimension is small. In real space, the sample consists of a small number of spectrally distinguishable components present in each pixel in varying thicknesses. We develop computational routines to exploit this low-dimensionality; both in the experimental design phase, as well as in the reconstruction phase. By leveraging established NLA techniques such as random leverage-score sampling [2], CUR decompositions [5], iterative non-negative factorisation [3], and regularisation, we accelerate spectral imaging experiments through subsampling and improve reconstructions.

For example, consider spectro-microscopy, which combines spectroscopy and microscopy. The underlying structure of a (flattened) dataset is

$$D = \text{Pois}(\mu t), \quad \mu \in \mathbb{R}_{\geq 0}^{n_E \times S}, \quad t \in \mathbb{R}_{\geq 0}^{S \times n_X n_Y},$$

where  $\mu$  are absorption spectra in  $n_E$  energies of  $S$  different components and  $t$  are thickness maps of the corresponding components in the  $n_X n_Y$  pixels. We firstly develop a data-driven scheme to determine what entries of the dataset should be measured and what measurements can be skipped, building on [7]. The scheme is based on leverage score sampling important wavelengths and important pixels, determined on-the-fly. The result is a non-negative dataset with 80-95 % missing entries. We show that if the measured entries are structured in columns and rows of the flattened dataset, reconstructions are significantly more accurate than random missing entries.

Furthermore, we provide a novel algorithm to reconstruct a non-negative factorisation from a dataset with missing entries, which is especially efficient for a CUR decomposition of the data. This algorithm also allows regularisation, and leads to more physically-relevant reconstructions than existing methods [4].

We present similar schemes for spectro-ptychography [1], in which the low-rank structure is not readily present in the measured datasets. This leads to a combination of non-negative factorisation schemes combined with nonlinear conjugate gradient algorithms [6].

The work demonstrates that numerical linear algebra results from the last two decades have a wealth of applications beyond straightforward large-scale matrix manipulations. In particular, the ideas of modern NLA can be applied before any data has been obtained, to aid in experiment design and accelerating acquisition times.

## References

- [1] Batey, D.J., Claus, D. and Rodenburg, J.M., 2014. *Information multiplexing in ptychography*. Ultramicroscopy, 138, pp.13-21.
- [2] Drineas, P., Mahoney, M.W. and Muthukrishnan, S., 2006. *Sampling algorithms for  $l_2$  regression and applications*. Proc. Annu. ACM-SIAM Symp. on Discrete Algorithms, pp.1127-1136.
- [3] Gillis, N., 2020. *Nonnegative matrix factorization*. SIAM.
- [4] Lericot, M., Jacobsen, C., Schäfer, T. and Vogt, S., 2004. *Cluster analysis of soft X-ray spectromicroscopy data*. Ultramicroscopy, 100(1-2), pp.35-57.
- [5] Mahoney, M.W. and Drineas, P., 2009. *CUR matrix decompositions for improved data analysis*. PNAS, 106(3), pp.697-702.
- [6] Thibault, P. and Guizar-Sicairos, M., 2012. *Maximum-likelihood refinement for coherent diffractive imaging*. New J. Phys., 14(6), p.063004.
- [7] Quinn, P.D., Sabaté Landman, M., Davis, T., Freitag, M., Gazzola, S. and Dolgov, S., 2024. *Optimal Sparse Energy Sampling for X-ray Spectro-Microscopy: Reducing the X-ray Dose and Experiment Time Using Model Order Reduction*. Chem. Biomed. Imaging, 2(4), pp.283-292.

# On the Convergence of the CROP-Anderson Acceleration Method

Agnieszka Międlar, Ning Wan

## Abstract

Consider the following problem: Given a function  $g : \mathbb{C}^n \rightarrow \mathbb{C}^n$  find  $x \in \mathbb{C}^n$  such that

$$x = g(x), \quad \text{or alternatively} \quad f(x) = 0, \quad \text{with} \quad f(x) := g(x) - x. \quad (1)$$

Obviously, a simplest method of choice to solve this problem is the fixed-point iteration

$$x^{(k+1)} = g(x^{(k)}), \quad \text{for all } k \in \mathbb{N}. \quad (2)$$

Unfortunately, its convergence is often extremely slow. The problem of slow (or no) convergence of a sequence of iterates has been extensively studied by researchers since the early 20th century. Aitken's delta-squared process was introduced in 1926 [1] for nonlinear sequences, and since then, people have been investigating various extrapolation and convergence acceleration methods with Shanks transformation [2] providing one of the most important and fundamental ideas. In the following, we will consider two mixing acceleration methods: the Anderson Acceleration [3, 4] (also referred to as Pulay mixing [5, 6] in computational chemistry) and the Conjugate Residual algorithm with **O**Ptimal trial vectors (CROP) [7, 8]. Anderson Acceleration method has a long history in mathematics literature, which goes back to Anderson's 1965 seminal paper [3]. Over the years, the method has been successfully applied to many challenging problems [9, 10, 11]. An independent line of research on accelerating convergence of nonlinear solvers established by physicists and chemists has led to developments of techniques such as Pulay mixing [5, 6], also known as the Direct Inversion of the Iterative Subspace (DIIS) algorithm, which is instrumental in accelerating the Self-Consistent Field (SCF) iteration method in electronic structure calculations [12]. It is well-known that Anderson Acceleration method has connections with the Generalized Minimal Residual Method (GMRES) algorithm [13, Section 6.5] and can be categorized as a multisecant method [14, 15, 16, 17]. The first convergence theory for Anderson Acceleration, under the assumption of a contraction mapping, appears in [18]. The convergence of Anderson(1), a topic of particular interest to many researchers, is discussed separately in [19, 20]. The acceleration properties of Anderson Acceleration are theoretically justified in [21, 22]. For detailed and more comprehensive presentation of history, theoretical and practical results on the acceleration methods and their applications we refer readers to [23, 24] and references therein.

Given Anderson iterates  $x_A^{(k)}, k = 0, 1, \dots$  and corresponding residual (error) vectors, e.g.,  $f_A^{(k)} := g(x_A^{(k)}) - x_A^{(k)}$ , consider weighted averages of the prior iterates, i.e.,

$$\bar{x}_A^{(k)} := \sum_{i=0}^{m_A^{(k)}} \alpha_{A,i}^{(k)} x_A^{(k-m_A^{(k)}+i)} \quad \text{and} \quad \bar{f}_A^{(k)} := \sum_{i=0}^{m_A^{(k)}} \alpha_{A,i}^{(k)} f_A^{(k-m_A^{(k)}+i)}, \quad (3)$$

with weights  $\alpha_{A,0}^{(k)}, \dots, \alpha_{A,m_A^{(k)}}^{(k)} \in \mathbb{R}$  satisfying  $\sum_{i=0}^{m_A^{(k)}} \alpha_{A,i}^{(k)} = 1$ , a fixed depth (history or window size)

parameter  $m$  and a truncation parameter  $m_A^{(k)} := \min\{m, k\}$ . Anderson Acceleration achieves a faster convergence than a simple fixed-point iteration by using the past information to generate

new iterates as linear combinations of previous  $m_A^{(k)}$  iterates [5, 6, 14], i.e.,

$$\begin{aligned} x_A^{(k+1)} &= \bar{x}_A^{(k)} + \beta^{(k)} \bar{f}_A^{(k)} \\ &= (1 - \beta^{(k)}) \sum_{i=0}^{m_A^{(k)}} \alpha_{A,i}^{(k)} x_A^{(k-m_A^{(k)}+i)} + \beta^{(k)} \sum_{i=0}^{m_A^{(k)}} \alpha_{A,i}^{(k)} g(x_A^{(k-m_A^{(k)}+i)}), \end{aligned} \quad (4)$$

with given relaxation (or damping) parameters  $\beta^{(k)} \in \mathbb{R}^+$  and mixing coefficients  $\alpha_{A,i}^{(k)} \in \mathbb{R}$ ,  $i = 0, \dots, m_A^{(k)}$  selected to minimize the linearized residual (error) of a new iterate within an affine space  $\text{Aff}\{f_A^{(k-m_A^{(k)})}, \dots, f_A^{(k)}\}$ , i.e., obtained as a solution of the least-squares problem

$$\min_{\alpha_0, \dots, \alpha_{m_A^{(k)}}} \left\| \sum_{i=0}^{m_A^{(k)}} \alpha_i f_A^{(k-m_A^{(k)}+i)} \right\|_2^2 \quad \text{s. t.} \quad \sum_{i=0}^{m_A^{(k)}} \alpha_i = 1. \quad (5)$$

Note that in the case of  $\beta^{(k)} = 1$  a general formulation (4) introduced in the original work of Anderson [3, 4] reduces to the Pulay mixing [5, 6], i.e.,

$$x_A^{(k+1)} = \sum_{i=0}^{m_A^{(k)}} \alpha_{A,i}^{(k)} g(x_A^{(k-m_A^{(k)}+i)}). \quad (6)$$

The CROP method, introduced in [7], is a generalization of the Conjugate Residual (CR) method [13, Section 6.8], which is a well-known iterative algorithm for solving linear systems. Analogously, we consider iterates  $x_C^{(k)}$ , a sequence of recorded search directions  $\Delta x_C^{(i)} := x_C^{(i+1)} - x_C^{(i)}$ ,  $i = k - m_C^{(k)}, \dots, k - 1$ , and the residual (error) vectors  $f_C^{(k)}$  generated by the CROP algorithm. Then the new search direction  $\Delta x_C^{(k)} = x_C^{(k+1)} - x_C^{(k)}$  is chosen in the space spanned by the prior  $m_C^{(k)}$  search directions  $\Delta x_C^{(i)}$ ,  $i = k - m_C^{(k)}, \dots, k - 1$  and the most recent residual (error) vector  $f_C^{(k)}$ , i.e.,

$$x_C^{(k+1)} = x_C^{(k)} + \sum_{i=k-m_C^{(k)}}^{k-1} \eta_i \Delta x_C^{(i)} + \eta_k f_C^{(k)}, \quad \text{with some } \eta_{k-m_C^{(k)}}, \dots, \eta_k \in \mathbb{R}.$$

Let us assume we have carried  $k$  steps of the CROP algorithm, i.e., we have the subspace of optimal vectors  $\text{span}\{x_C^{(1)}, \dots, x_C^{(k)}\}$  at hand. From the residual vector  $f_C^{(k)}$ , we can introduce a preliminary improvement of the current iterate  $x_C^{(k)}$ , i.e.,

$$\tilde{x}_C^{(k+1)} := x_C^{(k)} + f_C^{(k)}. \quad (7)$$

Now, since (7) is equivalent to  $f_C^{(k)} = \tilde{x}_C^{(k+1)} - x_C^{(k)}$ , we can find the optimal vector  $x_C^{(k+1)}$  within the affine subspace  $\text{span}\{x_C^{(1)}, \dots, x_C^{(k)}, \tilde{x}_C^{(k+1)}\}$ , i.e.,

$$x_C^{(k+1)} = \sum_{i=0}^{m_C^{(k+1)}-1} \alpha_{C,i}^{(k+1)} x_C^{(k+1-m_C^{(k+1)}+i)} + \alpha_{C,m_C^{(k+1)}}^{(k+1)} \tilde{x}_C^{(k+1)}, \quad \text{with} \quad \sum_{i=0}^{m_C^{(k+1)}} \alpha_{C,i}^{(k+1)} = 1. \quad (8)$$

The estimated residual (error)  $f_C^{(k+1)}$  corresponding to the iterate  $x_C^{(k+1)}$  is constructed as the linear combination of the estimated residuals of each component in (8) with the same coefficients, i.e.,

$$f_C^{(k+1)} = \sum_{i=0}^{m_C^{(k+1)}-1} \alpha_{C,i}^{(k+1)} f_C^{(k+1-m_C^{(k+1)}+i)} + \alpha_{C,m_C^{(k+1)}}^{(k+1)} \tilde{f}_C^{(k+1)}. \quad (9)$$

Note that in general, unlike for the Anderson Acceleration method,  $f_C^{(k+1)} \neq f(x_C^{(k+1)})$ . Minimizing the norm of the residual (error) defined in (9) results in a constrained least-squares problem

$$\min_{\alpha_0, \dots, \alpha_{m_C^{(k+1)}}} \left\| \sum_{i=0}^{m_C^{(k+1)}-1} \alpha_i f_C^{(k+1-m_C^{(k+1)}+i)} + \alpha_{m_C^{(k+1)}} \tilde{f}_C^{(k+1)} \right\|_2^2, \quad \text{such that} \quad \sum_{i=0}^{m_C^{(k+1)}} \alpha_{C,i}^{(k+1)} = 1. \quad (10)$$

Anderson Acceleration method is a well-established method that allows to speed up or encourage convergence of fixed-point iterations and it has been successfully used in a variety of applications. In recent years, the Conjugate Residual with OPTimal trial vectors (CROP) algorithm was introduced and shown to have a better performance than the classical Anderson Acceleration with less storage needed. In this work we aim to delve into the intricate connections between the classical Anderson Acceleration method and the CROP algorithm. Our objectives include a comprehensive study of their convergence properties, explaining the underlying relationships, and substantiating our findings through some numerical examples. Through this exploration, we contribute valuable insights that can enhance the understanding and application of acceleration methods in practical computations, as well as the developments of new and more efficient acceleration schemes. In particular, we will discuss the connection between the CROP algorithm and some other well-known methods, analyze its equivalence with Anderson Acceleration method and investigate convergence for linear and nonlinear problems. We will present a unified Anderson-type framework and show the equivalence between Anderson Acceleration method and the CROP algorithm. Moreover, we will compare the CROP algorithm with some Krylov subspace methods for linear problems and with multisection methods in the general case. We will illustrate the connection between the CROP algorithm and Anderson Acceleration method and explain the CROP-Anderson variant. Furthermore, we will show situations in which CROP and CROP-Anderson algorithms work better than Anderson Acceleration method. We will discuss the convergence results for CROP and CROP-Anderson algorithms for linear and nonlinear problems, and extend CROP and CROP-Anderson algorithms to rCROP and rCROP-Anderson, respectively, by incorporating real residuals to make them work better for nonlinear problems.

## References

- [1] A. C. Aitken. On Bernoulli's Numerical Solution of Algebraic Equations. *Proc. R. Soc. Edinb.*, 46:289–305, 1926.
- [2] D. Shanks. Non-linear transformations of divergent and slowly convergent sequences. *J. Math. Phys.*, 34(1-4):1–42, 1955.
- [3] D. G. Anderson. Iterative procedures for nonlinear integral equations. *J. ACM*, 12(4):547–560, 1965.
- [4] D. G. M. Anderson. Comments on “Anderson acceleration, mixing and extrapolation”. *Numer. Algorithms*, 80(1):135–234, 2019.
- [5] P. Pulay. Convergence acceleration of iterative sequences. The case of SCF iteration. *Chem. Phys. Lett.*, 73(2):393–398, 1980.
- [6] P. Pulay. Improved SCF convergence acceleration. *J. Comput. Chem.*, 3(4):556–560, 1982.
- [7] M. Ziolkowski, V. Weijo, P. Jørgensen, and J. Olsen. An efficient algorithm for solving nonlinear equations with a minimal number of trial vectors: Applications to atomic-orbital based coupled-cluster theory. *J. Chem. Phys.*, 128(20):204105, 2008.

- [8] P. Ettenhuber and P. Jørgensen. Discarding information from previous iterations in an optimal way to solve the coupled cluster amplitude equations. *J. Chem. Theory. Comput.*, 11(4):1518–1524, 2015.
- [9] E. Cancès and C. Le Bris. On the convergence of SCF algorithms for the Hartree-Fock equations. *M2AN Math. Model. Numer. Anal.*, 34(4):749–774, 2000.
- [10] L. Lin, J. Lu, and L. Ying. Numerical methods for Kohn-Sham density functional theory. *Acta Numer.*, 28:405–539, 2019.
- [11] G. Kemlin E. Cancès and A. Levitt. Convergence analysis of direct minimization and self-consistent iterations. *SIAM J. Matrix Anal. Appl.*, 42(1):243–274, 2021.
- [12] P. Ni. *Anderson acceleration of fixed-point iteration with applications to electronic structure computations*. PhD thesis, Worcester Polytechnic Institute, 2009.
- [13] Y. Saad. *Iterative methods for sparse linear systems*. Society for Industrial and Applied Mathematics, Philadelphia, PA, second edition, 2003.
- [14] H. F. Walker and P. Ni. Anderson acceleration for fixed-point iterations. *SIAM J. Numer. Anal.*, 49(4):1715–1735, 2011.
- [15] T. Rohwedder and R. Schneider. An analysis for the DIIS acceleration method used in quantum chemistry calculations. *J. Math. Chem.*, 49(9):1889–1914, 2011.
- [16] V. Eyert. A comparative study on methods for convergence acceleration of iterative vector sequences. *J. Comput. Phys.*, 124(2):271–285, 1996.
- [17] H. Fang and Y. Saad. Two classes of multisecant methods for nonlinear acceleration. *Numer. Linear Algebra Appl.*, 16(3):197–221, 2009.
- [18] A. Toth and C. T. Kelley. Convergence analysis for Anderson acceleration. *SIAM J. Numer. Anal.*, 53(2):805–819, 2015.
- [19] L. G. Rebholz and M. Xiao. The effect of Anderson Acceleration on superlinear and sublinear convergence. *SIAM J. Sci. Comput.*, 96(34), 2023.
- [20] H. De Sterck and Y. He. Linear asymptotic convergence of Anderson Acceleration: Fixed-point analysis. *SIAM J. Matrix Anal. Appl.*, 43(4):1755–1783, 2022.
- [21] M. Chupin, M.-S. Dupuy, G. Legendre, and É. Séré. Convergence analysis of adaptive DIIS algorithms with application to electronic ground state calculations. *arXiv:2002.12850*, 2020. <https://arxiv.org/abs/2002.12850>.
- [22] C. Evans, S. Pollock, L. G. Rebholz, and M. Xiao. A proof that Anderson acceleration improves the convergence rate in linearly converging fixed-point methods (but not in those converging quadratically). *SIAM J. Numer. Anal.*, 58(1):788–810, 2020.
- [23] C. Brezinski. Convergence acceleration during the 20th century. In *Numerical analysis 2000, Vol. II: Interpolation and extrapolation*, volume 122 of *J. Comput. Appl. Math.*, pages 1–21. 2000.
- [24] C. Brezinski, S. Cipolla, M. Redivo-Zaglia, and Y. Saad. Shanks and Anderson-type acceleration techniques for systems of nonlinear equations. *IMA J. Numer. Anal.*, 42(4):3058–3093, 2022.

# Interpolation-Based Algorithms to Compute the $\mathcal{H}_\infty$ Norm of a Parametric System

*Peter Benner, Tim Mitchell*

## Abstract

Consider the following continuous-time linear time-invariant *parametric* system:

$$E(\mathbf{p})\dot{x} = A(\mathbf{p})x + B(\mathbf{p})u \quad (1a)$$

$$y = C(\mathbf{p})x + D(\mathbf{p})u, \quad (1b)$$

where matrices  $E(\mathbf{p}), A(\mathbf{p}) \in \mathbb{C}^{n \times n}$ ,  $B(\mathbf{p}) \in \mathbb{C}^{n \times m}$ ,  $C(\mathbf{p}) \in \mathbb{C}^{p \times n}$ , and  $D(\mathbf{p}) \in \mathbb{C}^{p \times m}$  describe the dynamics and vary continuously with respect to the real-valued scalar parameter  $\mathbf{p} \in \mathcal{P} \subset \mathbb{R}$ , while the vectors  $x \in \mathbb{C}^n$ ,  $u \in \mathbb{C}^m$ , and  $y \in \mathbb{C}^p$  respectively describe the state, input, and output. The  $\mathcal{H}_\infty$  norm of (1) is an important quantity in many domains. In engineering applications, it measures how robust the system remains in the presence of noise, while in model order reduction, it is used to measure how well a reduced-order model mimics the dynamical behavior of a large-scale system. For a fixed value of  $\mathbf{p} \in \mathcal{P}$ , globally convergent methods for computing the  $\mathcal{H}_\infty$  norm go back to [BB90, BS90], but here we are interested in efficiently computing the *worst (highest) value of  $\mathcal{H}_\infty$  norm* of (1) that occurs over the parameter domain  $\mathcal{P}$ , which we denote  $h_*$ , or said another way, the parameter(s)  $\mathbf{p}_* \in \mathcal{P}$  where  $h_*$  is attained and (1) is the *least* robust to noise.

We begin with some preliminaries. We assume that the matrix pencil  $\lambda E(\mathbf{p}) - A(\mathbf{p})$  is regular and rank 1 for all values of  $\mathbf{p} \in \mathcal{P}$ , all the matrices are differentiable with respect to  $\mathbf{p}$  (except for possibly on a subset of  $\mathcal{P}$  of measure zero), and that the parameter domain consists of a finite number of intervals. The associated transfer function for (1) is

$$G(s; \mathbf{p}) = C(\mathbf{p})(sE(\mathbf{p}) - A(\mathbf{p}))^{-1}B(\mathbf{p}) + D(\mathbf{p}), \quad (2)$$

where  $s \in \mathbb{C}$ , and for a *fixed value* of  $\mathbf{p}$ , its  $\mathcal{H}_\infty$  norm is defined as

$$\|G(\cdot; \mathbf{p})\|_\infty = \max_{s \in \mathbb{C}_+} \|C(\mathbf{p})(sE(\mathbf{p}) - A(\mathbf{p}))^{-1}B(\mathbf{p}) + D(\mathbf{p})\|_2 =: \max_{s \in \mathbb{C}_+} g(s; \mathbf{p}), \quad (3)$$

where  $\mathbb{C}_+$  is the closed right half of the complex plane. If the system is known to be asymptotically stable, then the  $\mathcal{H}_\infty$  norm coincides with the  $\mathcal{L}_\infty$  norm, i.e., the maximization of the norm of the transfer function can be limited to the imaginary axis instead of all  $\mathbb{C}_+$ . For fixed  $\mathbf{p}$ , let  $\lambda$  be such that  $\det(\lambda E(\mathbf{p}) - A(\mathbf{p})) = 0$  and let  $x$  and  $y$  respectively be its right and left eigenvectors. Then eigenvalue  $\lambda$  is *controllable* if  $B(\mathbf{p})^*y \neq 0$  and it is *observable* if  $C(\mathbf{p})x \neq 0$ . Then  $\|G(\cdot; \mathbf{p})\|_\infty < \infty$  provided that all the eigenvalues of  $\lambda E(\mathbf{p}) - A(\mathbf{p})$  that are both controllable and observable are finite and in the open left half plane. In sum, our quantities of interest are given by

$$h_* = \max_{\mathbf{p} \in \mathcal{P}} \|G(\cdot; \mathbf{p})\|_\infty \quad \text{and} \quad \mathbf{p}_* = \arg \max_{\mathbf{p} \in \mathcal{P}} \|G(\cdot; \mathbf{p})\|_\infty. \quad (4)$$

One direct way to estimate  $h_*$  would be to simply evaluate (3) using the standard level-set method [BB90, BS90] over a grid on the parameter domain  $\mathcal{P}$ , but doing so provides no guarantee that  $h_*$  will be estimated to even moderate accuracy. A more refined yet still rather direct approach would be to globally approximate the value of  $\|G(\cdot; \mathbf{p})\|_\infty$  as it varies with  $\mathbf{p}$  over  $\mathcal{P}$ ,

say, by using Chebfun [DHT14], and then simply extract  $h_*$  and  $\mathbf{p}_*$  from the resulting interpolant<sup>1</sup>, but this is likely to be unnecessarily expensive. For each evaluation of  $\|G(\cdot; \mathbf{p})\|_\infty$ , of which many will be needed, the standard level-set method generally needs several iterations of computing the  $\gamma$ -level set points of  $g(i\omega)$ , which involves computing all imaginary eigenvalues of the matrix pencil  $M_\gamma(\mathbf{p}) - \lambda N_\gamma(\mathbf{p})$ , where

$$M_\gamma(\mathbf{p}) := \begin{bmatrix} A(\mathbf{p}) - B(\mathbf{p})R(\mathbf{p})^{-1}D(\mathbf{p})^*C(\mathbf{p}) & -\gamma B(\mathbf{p})R(\mathbf{p})^{-1}B(\mathbf{p})^* \\ \gamma C(\mathbf{p})S(\mathbf{p})^{-1}C(\mathbf{p}) & -(A(\mathbf{p}) - B(\mathbf{p})R(\mathbf{p})^{-1}D(\mathbf{p})^*C(\mathbf{p}))^* \end{bmatrix}, \quad (5a)$$

$$N_\gamma(\mathbf{p}) := \begin{bmatrix} E(\mathbf{p}) & 0 \\ 0 & E(\mathbf{p})^* \end{bmatrix}, \quad (5b)$$

$$R(\mathbf{p}) := D(\mathbf{p})^*D(\mathbf{p}) - \gamma^2 I, \quad (5c)$$

$$S(\mathbf{p}) := D(\mathbf{p})D(\mathbf{p})^* - \gamma^2 I. \quad (5d)$$

The cost can be significantly reduced by instead using the  $\mathcal{H}_\infty$ -norm method of [BM18], as it typically reduces the number of eigenvalue computations of  $M_\gamma(\mathbf{p}) - \lambda N_\gamma(\mathbf{p})$  to just one or two values of  $\gamma$ . However, if the system (1) is unstable for some values of  $\mathbf{p} \in \mathcal{P}$ , then  $h_* = \infty$ , but many eigenvalue computations of  $M_\gamma(\mathbf{p}) - \lambda N_\gamma(\mathbf{p})$  may be incurred to ascertain that fact in the process interpolating  $\|G(\cdot; \mathbf{p})\|_\infty$  over  $\mathcal{P}$ . This suggests that in order to be efficient, a new algorithm that first separately addresses the question of stability and then only proceeds with further computation when  $h_* < \infty$  is needed.

For any  $\mathbf{p} \in \mathcal{P}$ , define

$$\Lambda(\mathbf{p}) := \{\lambda \in \mathbb{C} : \det(\lambda E(\mathbf{p}) - A(\mathbf{p})) = 0, \lambda \text{ is both controllable and observable}\}, \quad (6a)$$

$$\alpha(\mathbf{p}) := \max\{\operatorname{Re} \lambda : \lambda \in \Lambda(\mathbf{p})\}, \quad (6b)$$

where we take  $\operatorname{Re} \lambda = +\infty$  for any non-finite  $\lambda \in \Lambda(\mathbf{p})$ . Then the system (1) is asymptotically stable if

$$\alpha_* := \max_{\mathbf{p} \in \mathcal{P}} \alpha(\mathbf{p}) < 0, \quad (7)$$

and so we can determine if  $h_* < \infty$  by approximating function  $\alpha$  over  $\mathcal{P}$  using Chebfun, as has been done in [HMMS22] to check stability when constructing stable  $\mathcal{H}_2 \otimes \mathcal{L}_2$  reduced order models for parametric systems via optimization. Although  $\alpha$  may be discontinuous, either because  $\mathcal{P}$  consists of more than one interval or an eigenvalue becomes or ceases to be controllable or observable as  $\mathbf{p}$  varies, Chebfun can reliably approximate functions with jumps [PPT10].

Although we propose using global approximation of  $\alpha$  to ascertain  $h_* < \infty$ , we do not suggest globally approximating  $\|G(\cdot; \mathbf{p})\|_\infty$  to compute  $h_*$  when it is finite. Instead, we propose an optimization-with-restarts method that directly computes local maximizers of the two-real variable optimization problem

$$h_* = \max_{\omega \in \mathbb{R}, \mathbf{p} \in \mathcal{P}} g(i\omega; \mathbf{p}), \quad (8)$$

and then uses an *interpolation-based globality certificate* to either certify that the local maximizer is in fact a global maximizer where  $h_*$  is attained or provides new starting points on the  $\gamma$ -level set of  $g$  to restart the local optimization phase, where  $\gamma = g(i\hat{\omega}, \hat{\mathbf{p}})$  for a computed local maximizer  $(\hat{\omega}, \hat{\mathbf{p}})$ . Interpolation-based globality certificates were first conceived in [Mit21] to develop faster and more reliable algorithms for computing Kreiss constants and the distance to uncontrollability and have since been extended to computing the quantity sep-lambda [Mit23].

---

<sup>1</sup>Chebfun can do this extraction phase exceptionally fast as it produces a piecewise Chebyshev polynomial.

Even though  $g$  may have points where it is a nonsmooth, the subset of such points has measure zero, so obtaining local maximizers of  $g$  can be done with relative ease and efficiency using gradient-based methods such as BFGS [LO13] or gradient sampling [BCL<sup>+</sup>20], particularly since there are only two optimization variables and often  $m, p \ll n$ . Then, with a candidate local maximizer  $(\hat{\omega}, \hat{p})$  of  $g$  in hand and  $\gamma = g(i\hat{\omega}, \hat{p})$ , we check whether it is a global maximizer by approximating the one-variable function

$$c_\gamma(p) := \min\{(\operatorname{Re} \lambda)^2 : \det(M_\gamma(p) - \lambda N_\gamma(p)) = 0, \operatorname{Re} \lambda \geq 0\}, \quad (9)$$

which is continuous on each interval in  $\mathcal{P}$  and where the squaring acts to smooth out non-Lipschitz behavior when a double imaginary eigenvalues bifurcates into a pair of eigenvalues with imaginary axis symmetry. Function  $c_\gamma$  is analogous to the eigenvalue-based functions that are globally approximated in the interpolation-based globality certificates used in [Mit21, Mit23], and in our setting here, has the following key properties:

- (i)  $c_\gamma(p) \geq 0$  for all  $p \in \mathcal{P}$ .
- (ii) If  $\gamma > h_*$ , then  $c_\gamma(p) > 0$  for all  $p \in \mathcal{P}$ .
- (iii) If  $\gamma < h_*$ , then  $c_\gamma(p) = 0$  holds on subset of  $\mathcal{P}$  with *positive measure*.

By approximating  $c_\gamma$  globally on  $\mathcal{P}$ , we can determine whether or not  $\gamma$  is above or below  $h_*$ . When it is below, we need only find zeros of  $c_\gamma$ , which are relatively easy to find by Property (iii). Meanwhile, if  $\gamma > h_*$ , which will be true if  $(\hat{\omega}, \hat{p})$  is a global maximizer and we perturb the value  $\gamma = g(i\hat{\omega}, \hat{p})$  slightly upward by a tolerance, then globally approximating  $c_\gamma$  determines that it is strictly positive on  $\mathcal{P}$ , thus certifying that  $(\hat{\omega}, \hat{p})$  is indeed a global maximizer and  $h_*$  has been attained. A practical benefit of approximating  $c_\gamma$  is cost; evaluating  $c_\gamma(p)$  always only requires a single eigenvalue computation with  $M_\gamma(p) - \lambda N_\gamma(p)$  and negligible amount of constant-time additional work, while evaluating  $\|G(\cdot; p)\|_\infty$  may require more than one eigenvalue computation and also does other non-constant-time work on top of that.

In general, only a handful of restarts are needed by our method and the overall work is almost entirely dominated by approximating the function  $c_\gamma$  for the final value of  $\gamma \approx h_*$ , properties which we have also observed in our prior work with interpolation-based globality certificates [Mit21, Mit23]. In total, the algorithm requires  $\mathcal{O}(kn^3)$  work, where  $k$  is the total number of evaluations of  $c_\gamma$  over all values of  $\gamma$ . Although  $k$  may be large, it often is not strongly correlated with the number of system states  $n$ , and it corresponds to a task that is embarrassingly parallel and so its effect can be significantly diminished on multi-core machines. Consequently, our method tends to act like a cubically scaling method that has a large constant term. We have also extended this approach to compute the worst-case  $\mathcal{H}_\infty$  norm of parametric discrete-time systems. In contrast, while it might be possible to extend 2D level-set tests [Gu00, GMO<sup>+</sup>06, GO06, Mit20] to finding a global maximizer of  $g$  or its discrete-time analogue, at least in some cases, based on our past experience with that technique, we believe the resulting methods would likely be both much slower and less reliable due to rounding error.

## References

- [BB90] S. Boyd and V. Balakrishnan. A regularity result for the singular values of a transfer matrix and a quadratically convergent algorithm for computing its  $L_\infty$ -norm. *Systems Control Lett.*, 15(1):1–7, 1990.

- [BCL<sup>+</sup>20] J. V. Burke, F. E. Curtis, A. S. Lewis, M. L. Overton, and L. E. A. Simões. Gradient sampling methods for nonsmooth optimization. In A. M. Bagirov, M. Gaudioso, N. Karmitsa, M. M. Mäkelä, and S. Taheri, editors, *Numerical Nonsmooth Optimization: State of the Art Algorithms*, pages 201–225, Cham, 2020. Springer International Publishing.
- [BM18] P. Benner and T. Mitchell. Faster and more accurate computation of the  $\mathcal{H}_\infty$  norm via optimization. *SIAM J. Sci. Comput.*, 40(5):A3609–A3635, October 2018.
- [BS90] N. A. Bruinsma and M. Steinbuch. A fast algorithm to compute the  $H_\infty$ -norm of a transfer function matrix. *Systems Control Lett.*, 14(4):287–293, 1990.
- [DHT14] T. A Driscoll, N. Hale, and L. N. Trefethen. *Chebfun Guide*. Pafnuty Publications, Oxford, UK, 2014.
- [GMO<sup>+</sup>06] M. Gu, E. Mengi, M. L. Overton, J. Xia, and J. Zhu. Fast methods for estimating the distance to uncontrollability. *SIAM J. Matrix Anal. Appl.*, 28(2):477–502, 2006.
- [GO06] M. Gu and M. L. Overton. An algorithm to compute  $\text{Sep}_\lambda$ . *SIAM J. Matrix Anal. Appl.*, 28(2):348–359, 2006.
- [Gu00] M. Gu. New methods for estimating the distance to uncontrollability. *SIAM J. Matrix Anal. Appl.*, 21(3):989–1003, 2000.
- [HMMS22] M. Hund, T. Mitchell, P. Mlinarić, and J. Saak. Optimization-based parametric model order reduction via  $\mathcal{H}_2 \otimes \mathcal{L}_2$  first-order necessary conditions. *SIAM J. Sci. Comput.*, 44(3):A1554–A1578, 2022.
- [LO13] A. S. Lewis and M. L. Overton. Nonsmooth optimization via quasi-Newton methods. *Math. Program.*, 141(1–2, Ser. A):135–163, 2013.
- [Mit20] T. Mitchell. Computing the Kreiss constant of a matrix. *SIAM J. Matrix Anal. Appl.*, 41(4):1944–1975, 2020.
- [Mit21] T. Mitchell. Fast interpolation-based globality certificates for computing Kreiss constants and the distance to uncontrollability. *SIAM J. Matrix Anal. Appl.*, 42(2):578–607, 2021.
- [Mit23] T. Mitchell. Fast computation of  $\text{sep}_\lambda$  via interpolation-based globality certificates. *Electron. Trans. Numer. Anal.*, 58:402–431, 2023.
- [PPT10] R. Pachón, R. B. Platte, and L. N. Trefethen. Piecewise-smooth chebfuns. *IMA J. Numer. Anal.*, 30(4):898–916, July 2010.

# A Million-Dollar Matrix

*Cleve Moler*

## Abstract

The Redheffer matrix has been included in the MATLAB Gallery for many years, but, until now, I didn't know much about it. The Riemann Hypothesis is subject of one of the Clay Mathematics Institute Millenium Prizes that are worth one million dollars each.

If we could find an  $n$ -by- $n$  Redheffer matrix

```
R = gallery('redheff',n)
```

with a determinant that satisfies

```
det(R) > sqrt(n)
```

it would be worth a million dollars.

For over 100 years, mathematicians believed that such an  $n$  might exist. This is the story of Redheffer matrices, the Mertens conjecture, five different ways to compute  $n$ , and the proof that a prize-winning matrix does not exist.

<https://blogs.mathworks.com/cleve/2024/10/22/mobius-mertens-and-redheffer>

# MINARES: An Iterative Solver for Symmetric Linear Systems

*Alexis Montoison, Dominique Orban, Michael Saunders*

Abstract

## 1 MinAres

Suppose  $A \in \mathbb{R}^{n \times n}$  is a large symmetric matrix for which matrix-vector products  $Av$  can be computed efficiently for any vector  $v \in \mathbb{R}^n$ . We present a Krylov subspace method called MINARES for computing a solution to the following problems:

$$\text{Symmetric linear systems:} \quad Ax = b, \quad (1)$$

$$\text{Symmetric least-squares problems:} \quad \min \|Ax - b\|, \quad (2)$$

$$\text{Symmetric nullspace problems:} \quad Ar = 0, \quad (3)$$

$$\text{Symmetric eigenvalue problems:} \quad Ar = \lambda r, \quad (4)$$

$$\text{Singular value problems for rectangular } B: \begin{bmatrix} & B \\ B^T & \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \sigma \begin{bmatrix} u \\ v \end{bmatrix}. \quad (5)$$

If  $A$  is nonsingular, problems (1)–(2) have a unique solution  $x^*$ . When  $A$  is singular, if  $b$  is not in the range of  $A$  then (1) has no solution; otherwise, (1)–(2) have an infinite number of solutions, and we seek the unique  $x^*$  that solves the problem

$$\min \frac{1}{2} \|x\|^2 \quad \text{s.t.} \quad A^2x = Ab. \quad (6)$$

Let  $x_k$  be an approximation to  $x^*$  with residual  $r_k = b - Ax_k$ . If  $A$  were unsymmetric or rectangular, applicable solvers for (1)–(2) would be LSQR [10] and LSMR [3], which reduce  $\|r_k\|$  and  $\|A^T r_k\|$  respectively within the  $k$ th Krylov subspace  $\mathcal{K}_k(A^T A, A^T b)$  generated by the Golub-Kahan bidiagonalization on  $(A, b)$  [4].

For (1)–(5), our algorithm MINARES solves (6) by reducing  $\|Ar_k\|$  within the  $k$ th Krylov subspace  $\mathcal{K}_k(A, b)$  generated by the symmetric Lanczos process on  $(A, b)$  [6]. Thus when  $A$  is symmetric, MINARES minimizes the same quantity  $\|Ar_k\|$  as LSMR, but in different (more effective) subspaces, and it requires only one matrix-vector product  $Av$  per iteration, whereas LSMR would need two.

Qualitatively, certain residual norms decrease smoothly for these iterative methods, but other norms are more erratic as they approach zero. It is ideal if stopping criteria involve the smooth quantities. For LSQR and LSMR on general (possibly rectangular) systems,  $\|r_k\|$  decreases smoothly for both methods. We observe that while LSQR is always ahead by construction, it is never by very much. Thus on consistent systems  $Ax = b$ , LSQR may terminate slightly sooner than LSMR. On inconsistent systems  $Ax \approx b$ , the comparison is more striking.  $\|A^T r_k\|$  decreases erratically for LSQR but smoothly for LSMR, and there is usually a significance difference between the two. Thus LSMR may terminate significantly sooner [3].

Similarly for MINRES [9] and MINARES,  $\|r_k\|$  decreases smoothly for both methods, and on consistent symmetric systems  $Ax = b$ , MINRES may have a small advantage. On inconsistent symmetric systems  $Ax \approx b$ ,  $\|Ar_k\|$  decreases erratically for MINRES and its variant MINRES-QLP [2] but smoothly for MINARES, and there is usually a significant difference between them. Thus MINARES may terminate sooner.

MINARES completes the family of Krylov methods based on the symmetric Lanczos process. As it minimizes  $\|Ar_k\|$  (which always converges to zero), MINARES can be applied safely to any symmetric system.

On consistent symmetric systems, MINARES is a relevant alternative to MINRES and MINRES-QLP because it converges in a similar number of iterations if the stopping condition is based on  $\|r_k\|$ , and much sooner if the stopping condition is based on  $\|Ar_k\|$ . On singular inconsistent symmetric systems, MINARES outperforms MINRES-QLP and LSMR, and should be the preferred method. Furthermore, a lifting step [7] can be applied to move from the final iterate to the minimum-length solution (pseudoinverse) at negligible cost.

## 2 CAR

We introduce CAR, a new conjugate direction method similar to CG and CR (the conjugate gradient and conjugate residual methods of Hestenes and Stiefel [5, 11] for solving symmetric positive definite (SPD) systems  $Ax = b$ ). Each of these methods generates a sequence of approximate solutions  $x_k$  in the Krylov subspaces  $\mathcal{K}_k(A, b)$  by minimizing a quadratic function  $f(x)$ :

$$f_{\text{CG}}(x) = \frac{1}{2}x^T Ax - b^T x, \quad f_{\text{CR}}(x) = \frac{1}{2}\|Ax - b\|^2, \quad f_{\text{CAR}}(x) = \frac{1}{2}\|A^2x - Ab\|^2.$$

CAR is to MINARES as CR is to MINRES. For SPD  $A$ , CAR is mathematically equivalent to MINARES, and both methods exhibit monotonic decrease in  $\|Ar_k\|$ ,  $\|r_k\|$ ,  $\|x_k - x^*\|$ , and  $\|x_k - x^*\|_A$ . The name CAR reflects its property of generating successive  $A$ -residuals that are conjugate with respect to  $A$ . Designed to minimize  $\|Ar_k\|$  in  $\mathcal{K}_k(A, b)$ , CAR complements the family of conjugate direction methods CG and CR for SPD systems.

---

**Algorithm 1** CG

---

**Require:**  $A, b, \epsilon > 0$

$$\begin{aligned} k &= 0, x_0 = 0 \\ r_0 &= b, p_0 = r_0 \\ q_0 &= Ap_0 \\ \rho_0 &= r_0^T r_0 \\ \text{while } \|r_k\| > \epsilon \text{ do} \\ \quad \alpha_k &= \rho_k / p_k^T q_k \\ \quad x_{k+1} &= x_k + \alpha_k p_k \\ \quad r_{k+1} &= r_k - \alpha_k q_k \\ \quad \rho_{k+1} &= r_{k+1}^T r_{k+1} \\ \quad \beta_k &= \rho_{k+1} / \rho_k \\ \quad p_{k+1} &= r_{k+1} + \beta_k p_k \\ \quad q_{k+1} &= Ap_{k+1} \\ \quad k &\leftarrow k + 1 \\ \text{end while} \end{aligned}$$


---



---

**Algorithm 2** CR

---

**Require:**  $A, b, \epsilon > 0$

$$\begin{aligned} k &= 0, x_0 = 0 \\ r_0 &= b, p_0 = r_0 \\ s_0 &= Ar_0, q_0 = s_0 \\ \rho_0 &= r_0^T s_0 \\ \text{while } \|r_k\| > \epsilon \text{ do} \\ \quad \alpha_k &= \rho_k / \|q_k\|^2 \\ \quad x_{k+1} &= x_k + \alpha_k p_k \\ \quad r_{k+1} &= r_k - \alpha_k q_k \\ \quad s_{k+1} &= Ar_{k+1} \\ \quad \rho_{k+1} &= r_{k+1}^T s_{k+1} \\ \quad \beta_k &= \rho_{k+1} / \rho_k \\ \quad p_{k+1} &= r_{k+1} + \beta_k p_k \\ \quad q_{k+1} &= s_{k+1} + \beta_k q_k \\ \quad k &\leftarrow k + 1 \\ \text{end while} \end{aligned}$$


---



---

**Algorithm 3** CAR

---

**Require:**  $A, b, \epsilon > 0$

$$\begin{aligned} k &= 0, x_0 = 0 \\ r_0 &= b, p_0 = r_0 \\ s_0 &= Ar_0, q_0 = s_0 \\ t_0 &= As_0, u_0 = t_0 \\ \rho_0 &= s_0^T t_0 \\ \text{while } \|r_k\| > \epsilon \text{ do} \\ \quad \alpha_k &= \rho_k / \|u_k\|^2 \\ \quad x_{k+1} &= x_k + \alpha_k p_k \\ \quad r_{k+1} &= r_k - \alpha_k q_k \\ \quad s_{k+1} &= s_k - \alpha_k u_k \\ \quad t_{k+1} &= As_{k+1} \\ \quad \rho_{k+1} &= s_{k+1}^T t_{k+1} \\ \quad \beta_k &= \rho_{k+1} / \rho_k \\ \quad p_{k+1} &= r_{k+1} + \beta_k p_k \\ \quad q_{k+1} &= s_{k+1} + \beta_k q_k \\ \quad u_{k+1} &= t_{k+1} + \beta_k u_k \\ \quad k &\leftarrow k + 1 \\ \text{end while} \end{aligned}$$


---

### 3 Krylov.jl

The algorithms MINRES and CAR have been implemented in Julia [1] as part of the package `Krylov.jl` [8], which provides a suite of Krylov and block-Krylov methods. Leveraging Julia’s flexibility and multiple dispatch capabilities, our implementations are compatible with all floating-point systems supported by the language, including complex numbers. These methods are optimized for both CPU and GPU architectures, ensuring high performance across a wide range of computational platforms. Additionally, our implementations support preconditioners, enhancing convergence and robustness across various problem classes.

## References

- [1] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah. Julia: A fresh approach to numerical computing. *SIAM Rev.*, 59(1):65–98, 2017.
- [2] S.-C. Choi, C. C. Paige, and M. A. Saunders. MINRES-QLP: A Krylov subspace method for indefinite or singular symmetric systems. *SIAM J. Sci. Comput.*, 33(4):1810–1836, 2011.
- [3] D. C.-L. Fong and M. A. Saunders. LSMR: An iterative algorithm for sparse least-squares problems. *SIAM J. Sci. Comput.*, 33(5):2950–2971, 2011.
- [4] G. H. Golub and W. Kahan. Calculating the singular values and pseudo-inverse of a matrix. *SIAM J. Numer. Anal.*, 2(2):205–224, 1965.
- [5] M. R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *J. Res. Natl. Bur. Stand.*, 49(6):409–436, 1952.
- [6] C. Lanczos. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *J. Res. Natl. Bur. Stand.*, 45:225–280, 1950.
- [7] Y. Liu, A. Milzarek, and F. Roosta. Obtaining pseudo-inverse solutions with MINRES. *arXiv preprint arXiv:2309.17096*, 2023.
- [8] A. Montoison and D. Orban. Krylov.jl: A Julia basket of hand-picked Krylov methods. *Journal of Open Source Software*, 8(89):5187, 2023.
- [9] C. C. Paige and M. A. Saunders. Solution of sparse indefinite systems of linear equations. *SIAM J. Numer. Anal.*, 12(4):617–629, 1975.
- [10] C. C. Paige and M. A. Saunders. LSQR: An algorithm for sparse linear equations and sparse least squares. *ACM Trans. Math. Software*, 8(1):43–71, 1982.
- [11] E. Stiefel. Relaxationsmethoden bester strategie zur lösung linearer gleichungssysteme. *Commentarii Mathematici Helvetici*, 29(1):157–179, 1955.

# QUASITUBAL TENSOR Framework with APPLICATIONS TO MULTIWAY FUNCTIONAL DATA

*Uriya Mor, Haim Avron*

Abstract

Multiway arrays, commonly referred to as higher-order tensors, are a natural data structure for representing multi-dimensional data and modeling processes consisting of composite interactions between factors. The tubal tensor framework [6, 1, 5] views a tensor as a ‘matrix of tubes’, where tubes are elements of a vector space supplemented with a binary, bilinear tubal multiplication, thus endowing the set of tubes with scalar-like properties that enable matrix mimetic tensor-tensor multiplication. From this perspective, tensors represent t-(tube-) linear mappings between Hilbert C\*-modules over the algebra of tubes [3, 2], for example, a 3rd order tensor  $\mathbf{X} \in \mathbb{R}^{m \times p \times n}$  represents a t-linear mapping from  $\mathbb{R}^{p \times 1 \times n}$  to  $\mathbb{R}^{m \times 1 \times n}$ , and the t-product of  $\mathbf{X}$  with a tensor  $\mathbf{Y} \in \mathbb{R}^{p \times q \times n}$  is a tensor  $\mathbf{X} * \mathbf{Y} \in \mathbb{R}^{m \times q \times n}$  that represents the composition of the two mappings. The matrix mimetic nature of the t-product enables an almost direct translation of many matrix computations to the tensor setting in a way that preserve, to some extent, the theoretical properties of the original operations, e.g., perhaps most notable, the t-SVD which is a straightforward extension of the matrix SVD, and enjoys an Eckart-Young like optimality result for rank truncations of a tensor [2, 4, 5]. The extensive, still-growing set of matrix algorithms and tools, and the ease of their extension to tensors via the tubal framework, make it a powerful tool for dealing with multi-dimensional problems.

In many applications, tensor data is obtained by a finite set of observations of a multi-dimensional process evolving over a domain such as time or space. These processes are often modeled as elements within an infinite-dimensional Hilbert space. However, when tubes reside in an infinite-dimensional Hilbert space, the associated tubal algebra lacks certain properties present in the finite-dimensional case, such as a multiplicative identity and von Neumann regularity. This limitation hinders any direct extension of the tubal tensor framework to infinite-dimensional spaces, and, in particular, the tubal SVD is no longer viable.

In this work, we introduce the quasitubal tensor framework, an extension of the tubal tensor framework to tubal algebras defined on infinite-dimensional separable Hilbert spaces. Notably, we establish the existence of a quasitubal SVD and prove Eckart-Young optimality results for low-rank truncations of quasitubal SVD. With a strong theoretical basis, the quasitubal framework offers attractive approach for tackling multi-way problems in infinite-dimensional spaces.

**Background.** An order- $N$  tensor  $\mathbf{X}$  over a field  $\mathbb{F}$  (either  $\mathbb{C}$  or  $\mathbb{R}$ ) is an object in  $\mathbb{F}^{d_1 \times \dots \times d_N}$ . The line of research on tubal tensor algebra [5, 1, 6, 4] views tensors ‘matrices of tubes’. For example, a 3rd order tensor  $\mathbf{X} \in \mathbb{F}^{m \times p \times n}$  is considered as an  $m \times p$  matrix over  $\mathbb{F}^n$  whose  $j, k$  (tubal) entry is  $\mathbf{x}_{jk} \in \mathbb{F}^n$ . The *t-product* [6, 1, 5] of two tubes  $\mathbf{x}, \mathbf{y} \in \mathbb{F}^n$  is defined as  $\mathbf{x} * \mathbf{y} = \text{ifft}(\widehat{\mathbf{x}} \odot \widehat{\mathbf{y}})$ , where  $\widehat{\mathbf{x}} = \text{fft}(\mathbf{x})$  is the Fourier transform of  $\mathbf{x}$ , and  $\odot$  is the Hadamard product.

The *mode-3 multiplication* of  $\mathbf{X}$  by a matrix  $\mathbf{A} \in \mathbb{F}^{r \times n}$  is the tensor  $\mathbf{X} \times_3 \mathbf{A} \in \mathbb{F}^{m \times p \times r}$  whose  $j, k$  tube fiber is given by  $\mathbf{A}\mathbf{x}_{jk} \in \mathbb{F}^r$ . In particular, let  $\mathbf{F}$  be the  $n \times n$  DFT matrix and define  $\widehat{\mathbf{X}} = \mathbf{X} \times_3 \mathbf{F}$ . The *tensor-tensor t-product* of  $\mathbf{X} \in \mathbb{F}^{m \times p \times n}, \mathbf{Y} \in \mathbb{F}^{p \times q \times n}$  is defined by  $\mathbf{X} * \mathbf{Y} = (\widehat{\mathbf{X}} \triangle \widehat{\mathbf{Y}}) \times_3 \mathbf{F}^{-1}$  with  $\mathbf{Z} = \mathbf{X} \triangle \mathbf{Y}$  a tensor such that  $\mathbf{Z}_{:, :, j} = \mathbf{X}_{:, :, j} \mathbf{Y}_{:, :, j}$ . Note that the t-product of two tubal-tensors is in-fact the multiplication of matrices over the tubal ring. More general version of the t-product is obtained by replacing  $\mathbf{F}$  with any invertible matrix  $\mathbf{M}$  [2, 4] (so  $\widehat{\mathbf{X}} = \mathbf{X} \times_3 \mathbf{M}$ ), resulting in the  $*_{\mathbf{M}}$ -product,  $\mathbf{X} *_{\mathbf{M}} \mathbf{Y} = (\widehat{\mathbf{X}} \triangle \widehat{\mathbf{Y}}) \times_3 \mathbf{M}^{-1}$ .

The  $p \times p$  identity tensor  $\mathbf{J}_p \in \mathbb{F}^{p \times p \times n}$  is such that  $\mathbf{X} \star_{\mathbf{M}} \mathbf{J}_p = \mathbf{X}, \mathbf{J}_p \star_{\mathbf{M}} \mathbf{Y} = \mathbf{Y}$ . The *Hermitian adjoint* of  $\mathbf{X} \in \mathbb{F}^{m \times p \times n}$  is the tensor  $\mathbf{X}^H \in \mathbb{F}^{p \times m \times n}$  with  $\widehat{\mathbf{X}}_{j,k,h}^H = \widehat{x}_{k,j,h}$ . A slice  $\vec{\mathbf{A}} \in \mathbb{F}^{p \times 1 \times n}$  is  $\star_{\mathbf{M}}$  unit normalized if  $\vec{\mathbf{A}}^H \star_{\mathbf{M}} \vec{\mathbf{A}} = \mathbf{1}$ , and we say that  $\vec{\mathbf{A}}, \vec{\mathbf{B}} \in \mathbb{F}^{p \times 1 \times n}$  are  $\star_{\mathbf{M}}$ -orthogonal if  $\vec{\mathbf{A}}^H \star_{\mathbf{M}} \vec{\mathbf{B}} = \mathbf{0}$ . A tensor  $\mathbf{U}$  is said to be  $\star_{\mathbf{M}}$ -unitary if  $\mathbf{U}^H * \mathbf{U} = \mathbf{U} \star_{\mathbf{M}} \mathbf{U}^H = \mathbf{J}$ . The *t-SVDM* of  $\mathbf{X} \in \mathbb{F}^{m \times p \times n}$  is a decomposition  $\mathbf{X} = \mathbf{U} \star_{\mathbf{M}} \mathbf{S} \star_{\mathbf{M}} \mathbf{V}^H$  where  $\mathbf{U} \in \mathbb{F}^{m \times m \times n}, \mathbf{V} \in \mathbb{F}^{p \times p \times n}$  are  $\star_{\mathbf{M}}$ -unitary, and  $\mathbf{S} \in \mathbb{F}^{m \times p \times n}$  is f-diagonal, i.e.,  $\mathbf{S}_{:,,:,k}$  are diagonal for all  $k$ . The *t-rank* of  $\mathbf{X}$  under  $\star_{\mathbf{M}}$  [5, 2] is the number of non-zero diagonal tubes in  $\mathbf{S}$ , and the *multi-rank* of  $\mathbf{X}$  under  $\star_{\mathbf{M}}$  [3, 4] is a vector  $\boldsymbol{\rho}$  of integers  $\rho_k = \text{rank}(\widehat{\mathbf{X}}_{:,,:,k})$ . Given  $r \leq \min(m, p)$ , the *t-rank r truncation* of  $\mathbf{X}$  under  $\star_{\mathbf{M}}$  is the tensor  $\mathbf{X}_r = \mathbf{U}_{:,1:r,:} \star_{\mathbf{M}} \mathbf{S}_{1:r,1:r,:} \star_{\mathbf{M}} \mathbf{V}_{:,1:r,:}^H = \sum_{j=1}^r \vec{\mathbf{U}}_j \star_{\mathbf{M}} \mathbf{S}_{j,j,:} \star_{\mathbf{M}} \vec{\mathbf{V}}_j^H$  with  $\vec{\mathbf{U}}_j = \mathbf{U}_{:,j,:}$  being the  $j$ th ‘column’ slice of  $\mathbf{U}$ . For  $\boldsymbol{\rho} = (\rho_1, \dots, \rho_n)$  with  $\rho_k \leq \min(m, p)$ , the *multi-rank  $\boldsymbol{\rho}$  truncation* of  $\mathbf{X}$  under  $\star_{\mathbf{M}}$  is the tensor  $\mathbf{X}_{\boldsymbol{\rho}}$  such that  $[\widehat{\mathbf{X}}_{\boldsymbol{\rho}}]_{:,,:,k} = \widehat{\mathbf{U}}_{:,1:\rho_k,:} \triangle \widehat{\mathbf{S}}_{1:\rho_k,1:\rho_k,:} \triangle \widehat{\mathbf{V}}_{:,1:\rho_k,:}^H$ . The central result of the tubal framework is that the above truncations are optimal in the sense of Frobenius norm error, provided that  $\mathbf{M}$  is a nonzero multiple of a unitary matrix. Formally, let  $\mathbf{M}$  be a nonzero multiple of a unitary matrix and  $\mathbf{X} \in \mathbb{F}^{m \times p \times n}$ . If  $\mathbf{Y} \in \mathbb{F}^{m \times p \times n}$  is of t-rank  $r$  (respectively, multirank  $\boldsymbol{\rho}$ ) under  $\star_{\mathbf{M}}$  then  $\|\mathbf{X} - \mathbf{Y}\|_F \geq \|\mathbf{X} - \mathbf{X}_r\|_F$  [5, 2] (respectively,  $\|\mathbf{X} - \mathbf{Y}\|_F \geq \|\mathbf{X} - \mathbf{X}_{\boldsymbol{\rho}}\|_F$  [4]).

In the above 3rd order example, each entry  $\mathbf{x}_{jk} \in \mathbb{F}^n$  of  $\mathbf{X}$  represents a *function*  $\mathbf{x}_{jk}: \Omega \rightarrow \mathbb{F}$ , where  $\Omega = [n] = \{1, \dots, n\}$  and  $x_{j,k,t} = \mathbf{x}_{jk}(t)$ . A common assumption in practice, is that the domain  $\Omega$  of  $\mathbf{x}_{jk}$  is actually a compact subset of  $\mathbb{R}$  and the values  $x_{j,k,h}$  are point evaluations of  $\mathbf{x}_{jk}$  on a grid  $t_1 \leq t_2 \leq \dots \leq t_n \in \Omega$  such that  $x_{j,k,h} = \mathbf{x}_{jk}(t_h)$ . Furthermore, it is possible to consider the functions  $\mathbf{x}_{jk}$  as elements of a Hilbert space  $(\mathcal{H}, \langle \cdot, \cdot \rangle_{\mathcal{H}})$  in which vector addition and scalar multiplication are defined pointwise. In this case, we have a ‘matrix of functions’ in  $\mathcal{H}$  and we write  $\mathbf{X} \in \mathcal{H}^{m \times p}$ .

**Matrices over Hilbert Spaces.** Suppose that  $\mathcal{H}$  is a separable Hilbert space over  $\mathbb{F}$ , and let  $\{\phi_j\}_{j \in \mathbb{Z}}$  be an orthonormal basis in  $\mathcal{H}$ . Then, the mapping  $\mathbf{x} \mapsto \Phi \mathbf{x} = \sum_j \langle \mathbf{x}, \phi_j \rangle_{\mathcal{H}} \mathbf{e}_j$  where  $\mathbf{e}_j$  is the  $j$ th standard basis vector in the space  $\ell_2$  of square summable sequences with the usual dot product, is an isometry. Note that if  $\mathbf{a}, \mathbf{b} \in \ell_2$  then the elementwise multiplication  $\mathbf{a} \odot \mathbf{b}$  is also in  $\ell_2$ . A natural extension of the  $\star_{\mathbf{M}}$  product to  $\mathcal{H}$  is given by  $\mathbf{x} \star_{\Phi} \mathbf{y} = \Phi^*(\Phi \mathbf{x} \odot \Phi \mathbf{y})$  where  $\Phi^*$  is the adjoint (and inverse) of  $\Phi$ . Let  $\mathbf{X} \in \mathcal{H}^{m \times p}$  and define the mode-3 operation of  $\Phi$  on  $\mathbf{X}$  as the tensor  $\widehat{\mathbf{X}} = \mathbf{X} \times_3 \Phi \in \ell_2^{m \times p}$  with  $\widehat{\mathbf{x}}_{jk} = \Phi \mathbf{x}_{jk}$ . Correspondingly, the *the tensor-tensor  $\star_{\Phi}$ -product* of  $\mathbf{X} \in \mathcal{H}^{m \times p}, \mathbf{Y} \in \mathcal{H}^{p \times q}$  is  $\mathbf{X} \star_{\Phi} \mathbf{Y} = (\widehat{\mathbf{X}} \triangle \widehat{\mathbf{Y}}) \times_3 \Phi^*$ .

**The Challenge of Defining Tubal SVD in Infinite Dimensional Hilbert Space .** Let  $\mathbf{x} \in \mathcal{H}$ , then the operation  $T_{\mathbf{x}}$  defined by  $T_{\mathbf{x}} \mathbf{y} = \mathbf{x} \star_{\Phi} \mathbf{y}$  is a bounded linear operator on  $\mathcal{H}$ . Furthermore,  $T_{\mathbf{x}}$  is Hilbert-Schmidt operator since  $\sum_j \|T_{\mathbf{x}} \phi_j\|_{\mathcal{H}}^2 = \sum_j \|\widehat{\mathbf{x}} \odot \mathbf{e}_j\|_{\ell_2}^2 = \sum_j |\widehat{x}_j|^2 = \sum_j |\langle \mathbf{x}, \phi_j \rangle_{\mathcal{H}}|^2 = \|\mathbf{x}\|_{\mathcal{H}}^2$ . Thus, the a multiplicative identity in  $\mathcal{H}$  is impossible since it would imply that the identity operator is a Hilbert-Schmidt operator, in contradiction to the infinite-dimensionality of  $\mathcal{H}$ . Direct consequences of this are that 1) there are no unit normalized slices in  $\mathcal{H}^p$  2) there are no  $\star_{\Phi}$ -unitary tensors in  $\mathcal{H}^{m \times m}$ . Most importantly, no decomposition of the form  $\mathbf{X} = \mathbf{U} \star_{\Phi} \mathbf{S} \star_{\Phi} \mathbf{V}^H$  can be defined in  $\mathcal{H}^{m \times p}$  such that  $\mathbf{U} \in \mathcal{H}^{m \times m}, \mathbf{V} \in \mathcal{H}^{p \times p}$  are isometries.

**Quasitubal Framework.** Consider the set  $\mathcal{H}^p := \bigoplus_{j=1}^p \mathcal{H}$  of slices  $\vec{\mathbf{X}} = (\mathbf{x}_1, \dots, \mathbf{x}_p)$  with elementwise addition and  $\star_{\Phi}$ -product by  $\mathcal{H}$  elements, e.g.,  $\vec{\mathbf{X}} \star_{\Phi} \mathbf{a} = \mathbf{a} \star_{\Phi} \vec{\mathbf{X}} = (\mathbf{a} \star_{\Phi} \mathbf{x}_1, \dots, \mathbf{a} \star_{\Phi} \mathbf{x}_p)$  and  $\vec{\mathbf{X}} + \vec{\mathbf{Y}} = (\mathbf{x}_1 + \mathbf{y}_1, \dots, \mathbf{x}_p + \mathbf{y}_p)$  for  $\vec{\mathbf{X}}, \vec{\mathbf{Y}} \in \mathcal{H}^p$  and  $\mathbf{a} \in \mathcal{H}$ . An operator  $T: \mathcal{H}^p \rightarrow \mathcal{H}^m$  is said to be **t-linear** (or  $\mathcal{H}$ -linear) if  $T(\mathbf{a} \star_{\Phi} \vec{\mathbf{X}}) = \mathbf{a} \star_{\Phi} T \vec{\mathbf{X}}$  for all  $\mathbf{a} \in \mathcal{H}, \vec{\mathbf{X}} \in \mathcal{H}^p$ . Let  $L(\mathcal{H}^p, \mathcal{H}^m)$  be the set of t-linear operators from  $\mathcal{H}^p$  to  $\mathcal{H}^m$ . Note that such operators are necessarily bounded and linear over  $\mathbb{F}$ , hence  $L(\mathcal{H}^p, \mathcal{H}^m) \subset B(\mathcal{H}^p, \mathcal{H}^m)$ .

Our theory is based on the following fundamental observations

**Lemma 0.1.** *An operator  $T$  is in  $L(\mathcal{H})$  if and only if there exists a bounded sequence  $\widehat{\boldsymbol{\tau}} \in \ell_\infty$  such that  $T\mathbf{a} = \Phi^*(\widehat{\boldsymbol{\tau}} \odot \widehat{\mathbf{a}})$  for all  $\mathbf{a} \in \mathcal{H}$ . If in addition,  $T$  is Hilbert-Schmidt then  $\widehat{\boldsymbol{\tau}} \in \ell_2$  and there exists  $\boldsymbol{\tau} \in \mathcal{H}$  such that  $T\mathbf{a} = \boldsymbol{\tau} \star_{\Phi} \mathbf{a}$  for all  $\mathbf{a} \in \mathcal{H}$ .*

As a consequence,  $T \in L(\mathcal{H}^p, \mathcal{H}^m)$  if and only if there exists  $\widehat{\boldsymbol{\mathfrak{T}}} \in \ell_\infty^{m \times p}$  such that  $T\vec{\mathbf{X}} = \Phi^*(\widehat{\boldsymbol{\mathfrak{T}}} \triangle \widehat{\vec{\mathbf{X}}})$  for all  $\vec{\mathbf{X}} \in \mathcal{H}^p$ . If in addition,  $T$  is Hilbert-Schmidt then  $\widehat{\boldsymbol{\mathfrak{T}}} \in \ell_2^{m \times p}$  and there exists  $\boldsymbol{\mathfrak{T}} \in \mathcal{H}^{m \times p}$  such that  $T\vec{\mathbf{X}} = \boldsymbol{\mathfrak{T}} \star_{\Phi} \vec{\mathbf{X}}$  for all  $\vec{\mathbf{X}} \in \mathcal{H}^p$ .

We call  $L(\mathcal{H})$  elements **quasitubes** due to their tubal representation in  $\ell_\infty$ . Respectively, operators in  $L(\mathcal{H}^p, \mathcal{H}^m)$  are called **quasitubal tensors** as they retain a tubal tensor structure in the coordinates of the transform domain. We use the same notation for  $L(\mathcal{H})$  and  $L(\mathcal{H}^p, \mathcal{H}^m)$  operators as for elements in  $\mathcal{H}, \mathcal{H}^{m \times p}$ , therefore, the  $\star_{\Phi}$  product of quasitubal tensors reads as composition of t-linear operators. While it is not possible to identify the space  $L(\mathcal{H}^p, \mathcal{H}^m)$  with  $\mathcal{H}^{m \times p}$  (as in the finite-dimensional case), the notation is still compatible, valid and useful.

**Lemma 0.2.** *The set  $L(\mathcal{H})$  with the usual operator addition, scaling, composition, adjoint and norm, is the smallest commutative, unital  $C^*$ -algebra in which  $\mathcal{H}$  is embedded as a  $*$ -ideal. And it follows that  $L(\mathcal{H}, \mathcal{H}^p) \cong L(\mathcal{H})^p$  together with the  $L(\mathcal{H})$ -valued inner-product  $\langle\langle \vec{\mathbf{X}}, \vec{\mathbf{Y}} \rangle\rangle = \sum_{j=1}^p \mathbf{x}_j^* \star_{\Phi} \mathbf{y}_j$  is a Hilbert  $C^*$ -module over  $L(\mathcal{H})$ , in which  $\mathcal{H}^p$  is embedded as a  $*$ -invariant submodule.*

Given  $\vec{\mathbf{X}} \in L(\mathcal{H})^p$  we have  $|\vec{\mathbf{X}}|_{L(\mathcal{H})^p}^2 = \langle\langle \vec{\mathbf{X}}, \vec{\mathbf{X}} \rangle\rangle$  which is a non-negative element in a  $C^*$ -algebra, hence has a unique square root  $|\vec{\mathbf{X}}|_{L(\mathcal{H})^p}$ , and the real valued norm  $\|\vec{\mathbf{X}}\|_{L(\mathcal{H})^p} = \| |\vec{\mathbf{X}}|_{L(\mathcal{H})^p} \|$ . The induced “operator norm” of an  $m \times p$  quasitubal tensors is then  $\|\mathbf{X}\| = \sup_{\|\vec{\mathbf{Y}}\|_{L(\mathcal{H})^p}=1} \|\mathbf{X} \star_{\Phi} \vec{\mathbf{Y}}\|_{L(\mathcal{H})^m}$ . Another consequence of the Hilbert  $C^*$ -module structure over a unital  $C^*$ -algebra, is the ability to define  $\star_{\Phi}$ -orthogonality and  $\star_{\Phi}$ -unitarity for quasitubal tensors similarly to the finite-dimensional case. With the above, the ground is set for construction of a quasitubal SVD:

**Theorem 0.3.** *Let  $\mathbf{X}$  be an  $m \times p$  quasitubal tensor, then there exists a decomposition  $\mathbf{X} = \mathbf{U} \star_{\Phi} \mathbf{S} \star_{\Phi} \mathbf{V}^*$  with  $\mathbf{U} \in L(\mathcal{H}^m)$ ,  $\mathbf{V} \in L(\mathcal{H}^p)$  being  $\star_{\Phi}$ -unitary, and  $\mathbf{S} \in L(\mathcal{H}^m, \mathcal{H}^p)$  an f-diagonal tensor with diagonal entries  $s_1 \geq_{L(\mathcal{H})} s_2 \geq_{L(\mathcal{H})} \dots \geq_{L(\mathcal{H})} s_{\min(m,p)} \geq_{L(\mathcal{H})} \mathbf{0}$ .*

The t-rank and multirank of a quasitensor  $\mathbf{X}$  under  $\star_{\Phi}$ , as well as t-rank and multi-rank truncations, are defined similarly to the finite-dimensional case. And we have the main result:

**Theorem 0.4.** *Given an  $m \times p$  quasitubal tensor  $\mathbf{X}$ , if  $\mathbf{Y} \in L(\mathcal{H}^p, \mathcal{H}^m)$  is of t-rank  $r$  (respectively, multirank  $\boldsymbol{\rho}$ ) under  $\star_{\Phi}$  then  $\|\mathbf{X} - \mathbf{Y}\| \geq \|\mathbf{X} - \mathbf{X}_r\|$  (respectively,  $\|\mathbf{X} - \mathbf{Y}\| \geq \|\mathbf{X} - \mathbf{X}_{\boldsymbol{\rho}}\|$ ).*

Objects in  $\mathcal{H}^{m \times p}$  have the elementwise  $\mathcal{H}$  norm:  $\|\mathbf{X}\|_{\mathcal{H}}^2 = \sum_{j,k} \|\mathbf{x}_{jk}\|_{\mathcal{H}}^2$ , which is an equivalent to the Frobenius norm in the finite-dimensional case. Consider  $\mathbf{X} = \mathbf{U} \star_{\Phi} \mathbf{S} \star_{\Phi} \mathbf{V}^* \in \mathcal{H}^{m \times p}$ , then  $\mathbf{S} \in \mathcal{H}^{m \times p}$  and  $\|\mathbf{X}\|_{\mathcal{H}} = \|\mathbf{S}\|_{\mathcal{H}}$ . Importantly

**Theorem 0.5.** *Given  $\mathbf{X} \in \mathcal{H}^{m \times p}$ , if  $\mathbf{Y} \in L(\mathcal{H}^p, \mathcal{H}^m)$  is of t-rank  $r$  (respectively, multirank  $\boldsymbol{\rho}$ ) under  $\star_{\Phi}$  then  $\|\mathbf{X} - \mathbf{Y}\|_{\mathcal{H}} \geq \|\mathbf{X} - \mathbf{X}_r\|_{\mathcal{H}}$  (respectively,  $\|\mathbf{X} - \mathbf{Y}\|_{\mathcal{H}} \geq \|\mathbf{X} - \mathbf{X}_{\boldsymbol{\rho}}\|_{\mathcal{H}}$ ). In particular  $\mathbf{X}_r, \mathbf{X}_{\boldsymbol{\rho}} \in \mathcal{H}^{m \times p}$ .*

**Possible Applications.** Due to the strong theoretical foundation of the quasitubal SVD, a promising line of research is the development of multivariate functional PCA, in a similar spirit to our previous work on the finite-dimensional settings [7]. Furthermore, the matrix mimetic nature of the platform, combined with the optimality results for low-rank truncations suggest that direct extensions of randomized algorithms for low-rank matrix approximations to the quasitubal setting are

possible and should offer theoretical guarantees. This opens the door to computational speedups in modeling and simulations of multi-input multi-output dynamical systems where the quality of the approximation is about as crucial as the computational cost. We provide numerical examples for the application of the quasitubal framework to multivariate functional data analysis and signal processing, and demonstrate the potential of the framework for developing efficient tensor-based algorithms for such settings.

## References

- [1] K. Braman. Third-Order Tensors as Linear Operators on a Space of Matrices. *Linear Algebra and its Applications*, 433(7):1241–1253, 12 2010.
- [2] E. Kernfeld, M. Kilmer, and S. Aeron. Tensor–tensor products with invertible linear transforms. *Linear Algebra and its Applications*, 485:545–570, Nov. 2015.
- [3] M. E. Kilmer, K. Braman, N. Hao, and R. C. Hoover. Third-Order Tensors as Operators on Matrices: A Theoretical and Computational Framework with Applications in Imaging. *SIAM Journal on Matrix Analysis and Applications*, 34(1):148–172, 2 2013.
- [4] M. E. Kilmer, L. Horesh, H. Avron, and E. Newman. Tensor-tensor algebra for optimal representation and compression of multiway data. *Proceedings of the National Academy of Sciences*, 118(28):e2015851118, July 2021.
- [5] M. E. Kilmer and C. D. Martin. Factorization strategies for third-order tensors. *Linear Algebra and its Applications*, 435(3):641–658, 8 2011.
- [6] M. E. Kilmer and C. D. M. Martin. Decomposing a tensor - are there analogues to the svd, lu, qr, and other matrix decompositions for tensors?, 11 2004.
- [7] U. Mor, Y. Cohen, R. Valdés-Mas, D. Kviatcovsky, E. Elinav, and H. Avron. Dimensionality reduction of longitudinal ’omics data using modern tensor factorizations. *PLOS Computational Biology*, 18(7):e1010212, July 2022.

# Krylov Methods and Polynomials

Ron Morgan

## Abstract

It is well-known that all Krylov methods have polynomials underlying them. Here these polynomials are used in new ways to develop iterative methods. There are several applications including systems of linear equations with multiple right-hand sides.

## 1 Polynomial Preconditioning

Polynomial preconditioning goes way back, see for instance [6, 11, 10, 2, 12], but has never become standard. In [5, 8], polynomial preconditioning is made a more practical method with these improvements:

1. The polynomial is easy to determine. It is generated by GMRES instead of from eigenvalue estimates as some approaches have done.
2. The implementation is efficient due to using roots of the GMRES residual polynomial.
3. The method is more stable than some other approaches. This is from the implementation with roots instead of coefficients. Also, additional stability control comes from adding extra copies of outstanding roots.

With polynomial preconditioning,  $Ax = b$  becomes

$$Ap(A)y = b, \quad x = p(A)y. \quad (1)$$

Polynomial preconditioning works because it transforms the spectrum of  $A$  into a better spectrum. Another reason why it is effective is that there is more power per iteration and much less orthogonalization. Also, there is greater opportunity for parallelism. However, polynomial preconditioning may not needed for fairly easy systems.

*Example* Polynomial preconditioning is used for a matrix from a biharmonic differential equation. A degree 200 polynomial is applied, then a degree 50 polynomial is used along with an incomplete factorization preconditioner. Table 1 first shows that polynomial preconditioning can be very effective for the difficult original system of equations. The time is reduced from hours to under a minute. The problem is easier when standard incomplete factorization preconditioning is applied. Then the effect of the polynomial preconditioning is not as dramatic, but it is still helpful. The time goes from over 3 minutes to below 9 seconds. See [8] for more results.

Table 1: Polynomial preconditioning applied to a biharmonic matrix of size  $n = 40,000$  from the differential equation  $-u_{xxxx} - u_{yyyy} + u_{xxx} = f$  on the unit square. IF stands for incomplete factorization with no fill-in after shifting the matrix by  $0.5 * I$ .

Method	GMRES(50)	PP(200)-GMRES(50)	IF-GMRES(50)	PP(50)-IF-GMRES(50)
Time	14.6 hours	55 seconds	3.5 minutes	8.5 seconds

## 2 Polynomial Approximation of $A^{-1}$

It is surprising that even for a large matrix, it is often possible to find a polynomial  $p$  so that  $p(A)$  is a good approximation to  $A^{-1}$  [4]. We use the  $p$  polynomial from Equation (1) that is generated by GMRES. This is implemented using the roots of the GMRES residual polynomial.

It can be proved for the symmetric case that the accuracy of the approximating polynomial follows the residual norm curve. Stability control with added roots is even more important here than for polynomial preconditioning, because a high degree polynomial is needed.

*Example:* For a convection-diffusion matrix, Table 2 shows that the relative accuracy of the polynomial approximation to the inverse follows the GMRES residual norm. See [4] for more.

Table 2: A polynomial approximation is found to the inverse of a matrix of size  $n = 2500$  from the convection-diffusion equation  $-u_{xx} - u_{yy} + 2u_x = f$  on the unit square. The starting vector is a random vector normed to one. Relative accuracy of the polynomial is compared to the GMRES residual norm.

Degree	GMRES residual norm	$\frac{\ A^{-1} - p(A)\ }{\ A^{-1}\ }$
50	$8 * 10^{-3}$	$2 * 10^{-1}$
100	$4 * 10^{-5}$	$6 * 10^{-4}$
150	$1 * 10^{-8}$	$3 * 10^{-7}$
200	$8 * 10^{-12}$	$3 * 10^{-10}$

Applications of polynomial approximation to  $A^{-1}$  include:

1. Systems with multiple right-hand sides [4]. The polynomial approximation that is generated with one right-hand side can be applied to other right-hand sides to solve the systems.
2. Multilevel Monte Carlo for the trace of the inverse in quantum chromodynamics [7]. Polynomials form the basis for this approach, but deflation of eigenvalues is also crucial.
3. Functions of matrices (in progress). A polynomial approximation to the function of a matrix can be found by interpolating the function at the harmonic Ritz values (the roots of the GMRES polynomial).

## 3 Twin CG for multiple right-hand side systems. Also Twin CR, BiCGStab, BiCG, GMRES and QMR.

The first application in the previous section used the same polynomial for multiple right-hand sides. Here we again have the same polynomial, but applied in a simpler way. We use the same iteration coefficients for all right-hand sides. When this approach is used for the conjugate gradient method (CG), we call this “Twin CG for multiple right-hand sides.”

As an example, consider two systems,  $Ax^{(1)} = b^{(1)}$  and  $Ax^{(2)} = b^{(2)}$ . One step in the CG iteration for the first system is  $x^{(1)} = x^{(1)} + \alpha^{(1)} * p^{(1)}$ . Our approach is to also do  $x^{(2)} = x^{(2)} + \alpha^{(1)} * p^{(2)}$  with the same  $\alpha^{(1)}$  as for the first system. Similarly, the other CG steps have the same constants for both right-hand sides.

The residual polynomial helps explain why this is effective. For the second right-hand side, the residual polynomial is the same as for the first. So if  $r^{(1)} = \pi^{(1)}(A)b^{(1)}$ , then also  $r^{(2)} = \pi^{(1)}(A)b^{(2)}$ .

The polynomial  $\pi^{(1)}$  needs to be small at the eigenvalues of  $A$  in order for the first system to converge, and this makes the second system converge also. The first right-hand side,  $b^{(1)}$ , does need to not be deficient in some eigencomponents. For the deficient case, creating a first system with a random right-hand side is recommended.

With this twin approach for CG, the multiple right-hand systems generally converge together. However, there can be momentary instability due to steep slope of the polynomial at an outstanding eigenvalue. This tends to be quickly automatically corrected.

Remarkable things about Twin CG:

1. The code can be very simple. All  $x$  vectors for all right-hand sides can be grouped together into one matrix, and similarly for  $r$  and  $p$  vectors. Then most operations can be done with matrices instead of vectors.
2. This method is extremely parallelizable. All dot products are eliminated except for the first right-hand side (one may need to monitor residual norms for other right-hand sides occasionally after the first system converges). The matrix-vector products can be done together for all right-hand sides. The DAXPY's can also be performed together, so they become matrix operations.
3. Stability control happens automatically due to roundoff error. *So roundoff error is essential to the success of the method.* This is because the Lanczos algorithm that is the basis for CG develops extra copies of outstanding eigenvalues [9]. This mimics the addition of roots given in [5, 8] for stability control with polynomial preconditioning. With this Twin CG approach, it is surprising that the extra copy appears almost as soon as stability control is needed. However, for cases with several outstanding eigenvalues, extra copies will not all appear simultaneously, so we have a way of augmenting with some manual stability control.
4. Seed methods [3, 1] for deflating eigenvalues can be added. Seeding is done during solution of the first system, then the Twin CG method is applied to the second and subsequent systems. In addition, we are working on a new seed approach for the case where roundoff error interferes with the accuracy of the seeding [1].

Finally, a few comments about nonsymmetric systems:

1. Similar to CG for symmetric systems, a Twin BiCG method can be given for the nonsymmetric case. BiCG has automatic stability control similar to CG.
2. Twin BiCG uses only half of the matrix-vector products for the second and subsequent right-hand sides as for the first.
3. BiCGStab does not have the automatic stability control.
4. The twin approach for multiple right-hand sides can even be used with restarted GMRES. Dot products are eliminated except for the first right-hand side, but the rest of the orthogonalization cost is needed for each right-hand side system.

## References

- [1] A. M. ABDEL-REHIM, R. B. MORGAN, AND W. WILCOX, *Improved seed methods for symmetric positive definite linear equations with multiple right-hand sides*, Numer. Linear Algebra Appl., 21 (2014), pp. 453–471.
- [2] S. F. ASHBY, *Polynomial preconditioning for conjugate gradient methods*. PhD Thesis, University of Illinois at Urbana-Champaign, 1987.

- [3] T. F. CHAN AND W. WAN, *Analysis of projection methods for solving linear systems with multiple right-hand sides*, SIAM J. Sci. Comput., 18 (1997), pp. 1698–1721.
- [4] M. EMBREE, J. A. HENNINGSEN, J. JACKSON, AND R. B. MORGAN, *Polynomial approximation to the inverse of a large matrix*, tech. rep., Baylor University, 2024. Available at [https://sites.baylor.edu/ronald\\_morgan/reports/](https://sites.baylor.edu/ronald_morgan/reports/).
- [5] M. EMBREE, J. A. LOE, AND R. B. MORGAN, *Polynomial preconditioned Arnoldi with stability control*, SIAM J. Sci. Comput., 43 (2021), pp. A1–A25.
- [6] C. LANCZOS, *Solution of systems of linear equations by minimized iterations*, J. Res. Nat. Bur. Standards, 49 (1952), pp. 33–53.
- [7] P. LASHOMB, R. B. MORGAN, T. WHYTE, AND W. WILCOX, *Multi-polynomial Monte Carlo for QCD trace estimation*, Comput. Phys. Commun., 300 (2024), pp. 109163–1 – 109163–10.
- [8] J. A. LOE AND R. B. MORGAN, *Toward efficient polynomial preconditioning for GMRES*, Numer. Linear Algebra Appl., 29 (2021), pp. 1–21.
- [9] C. C. PAIGE, *The computation of eigenvectors and eigenvalues of very large sparse matrices*. PhD Thesis, University of London, 1971.
- [10] Y. SAAD, *Practical use of some Krylov subspace methods for solving indefinite and unsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 5 (1984), pp. 203–228.
- [11] E. L. STIEFEL, *Kernel polynomials in linear algebra and their numerical applications*, Nat. Bur. Standards, Appl. Math. Ser., 49 (1958), pp. 1–22.
- [12] H. K. THORNQUIST, *Fixed-Polynomial Approximate Spectral Transformations for Preconditioning the Eigenvalue Problem*, PhD thesis, Rice University, 2006. Technical report TR06-05, Department of Computational and Applied Mathematics, Rice University.

# Block cross-interactive residual smoothing for Lanczos-type solvers for linear systems with multiple right-hand sides

*Kensuke Aihara, Akira Imaoka, Keiichi Morikuni*

## Abstract

Block Lanczos-type solvers, such as the block BiCGSTAB method [3], for large sparse linear systems

$$AX = B, \quad A \in \mathbb{R}^{n \times n}, \quad B \in \mathbb{R}^{n \times s}, \quad s \ll n$$

often exhibit large oscillations in the residual norms. In finite precision arithmetic, the large oscillations lead to a large residual gap (the difference  $G_{R_k}$  between the recursively updated residual  $R_k$  and the explicitly computed residual  $B - AX_k$ ) and a loss of attainable accuracy of the approximations, as observed in

$$\|G_{R_k}\| - \|R_k\| \leq \|B - AX_k\| \leq \|G_{R_k}\| + \|R_k\|, \quad G_{R_k} := (B - AX_k) - R_k.$$

This problem is addressed by using cross-interactive residual smoothing (CIRS). Just as the standard Lanczos-type solvers for a single linear system have been extended to their global and block versions for solving systems with multiple right-hand sides, similar extensions of CIRS are naturally considered. While we have developed the global CIRS scheme (Gl-CIRS) in our previous study [1], we propose a block version (Bl-CIRS) in this study. Then, we demonstrate the effectiveness of Bl-CIRS from various perspectives, such as theoretical insights into the convergence behaviors of the residual and approximation norms, numerical experiments on model problems, and a detailed rounding error analysis for the residual gap. In particular, we show for the case of Bl-CIRS that orthonormalizing the columns of direction matrices plays an important role in reducing the residual gap. The presented analysis also complements our previous study above that includes an evaluation for the residual gap of the block Lanczos-type solvers.

**Advances in residual smoothing** Block Lanczos-type solvers typically update the  $k$ th approximation  $X_k$  and residual  $R_k$  by using the recursion formulas

$$X_{k+1} = X_k + P_k \alpha_k^\square, \quad R_{k+1} = R_k - (AP_k)\alpha_k^\square, \quad k = 0, 1, \dots,$$

respectively, where  $P_k \in \mathbb{R}^{n \times s}$  is a direction matrix and  $\alpha_k^\square \in \mathbb{R}^{s \times s}$  is determined under a certain condition. Residual smoothing was introduced by Schönauer [7] to Lanczos-type solvers for a single linear system to get a non-increasing sequence of residual norms [8]. A block version of the simple residual smoothing (Bl-SRS) was presented by Jbilou [5]. Let  $X_k$  and  $R_k$  be the  $k$ th primary approximation and residual, respectively. Then, new sequences of approximations  $Y_k$  and the corresponding smoothed residuals  $S_k$  ( $:= B - AY_k$ ) are generated by the recursion formulas

$$Y_{k+1} = Y_k + (X_{k+1} - Y_k)\eta_{k+1}^\square, \quad S_{k+1} = S_k + (R_{k+1} - S_k)\eta_{k+1}^\square, \quad k = 0, 1, \dots,$$

respectively, where  $Y_0 = X_0$ ,  $S_0 = R_0$ , and  $\eta_k^\square \in \mathbb{R}^{s \times s}$  is a parameter matrix. With a local minimization of the smoothed residual norm in choosing  $\eta_k^\square$ , a monotonically decreasing sequence of  $\|S_k\|$  is obtained.

Studies on the relationship between residual smoothing and the residual gap have an interesting history. For a single right-hand side case, Gutknecht and Rozložník [4] clarified that the conventional

Table 1: Difference in the recursion formulas for updating smoothed residuals.

Type	SRS scheme	CIRS scheme
Standard $Ax = b$	[7, 8] $s_k = s_{k-1} + \eta_k(r_k - s_{k-1}),$ $r_k, s_{k-1} \in \mathbb{R}^n, \quad \eta_k \in \mathbb{R}$	[2, 6] $s_k = s_{k-1} - \eta_k A v_k,$ $v_k, s_{k-1} \in \mathbb{R}^n, \quad \eta_k \in \mathbb{R}$
Global $AX = B$	[9] $S_k = S_{k-1} + \eta_k(R_k - S_{k-1}),$ $R_k, S_{k-1} \in \mathbb{R}^{n \times s}, \quad \eta_k \in \mathbb{R}$	[1] $S_k = S_{k-1} - \eta_k A V_k,$ $V_k, S_{k-1} \in \mathbb{R}^{n \times s}, \quad \eta_k \in \mathbb{R}$
Block $AX = B$	[5] $S_k = S_{k-1} + (R_k - S_{k-1})\eta_k^\square,$ $R_k, S_{k-1} \in \mathbb{R}^{n \times s}, \quad \eta_k^\square \in \mathbb{R}^{s \times s}$	Present study $S_k = S_{k-1} - (A\tilde{Q}_k)\tilde{\eta}_k^\square,$ $\tilde{Q}_k, S_{k-1} \in \mathbb{R}^{n \times s}, \quad \tilde{\eta}_k^\square \in \mathbb{R}^{s \times s}$

smoothing schemes (including the Zhou–Walker implementation [10]) do not help to improve the attainable accuracy. To be more specific, rounding errors accumulated in the primary sequences propagate to the smoothed sequences, and the smoothed true residual norms stagnate at the same order of magnitude as the primary ones. In order to remedy this phenomenon, Komeyama et al. [2, 6] modified the Zhou–Walker implementation so that the primary and smoothed sequences influence one another. This modification is referred to as cross-interactive residual smoothing (CIRS) and is indeed effective in reducing the residual gap and increasing attainable accuracy. As SRS has been extended to global and block versions [9, 5], CIRS is also extended. In this perspective, our previous study [1] presented a global version of CIRS (Gl-CIRS) for the global- and block-type solvers, and therefore, we propose a block version of CIRS (Bl-CIRS) in this study. Table 1 summarizes the recursion formulas for the aforementioned residual smoothing schemes. This table shows that this study fills a gap in the literature of the CIRS schemes.

**Block cross-interactive residual smoothing** This study proposes updating approximations and the corresponding residuals by the recursion formulas

$$\begin{array}{lll} \text{smoothed} & Y_{k+1} = Y_k + V_{k+1}\eta_{k+1}^\square, & S_{k+1} = S_k - (AV_{k+1})\eta_{k+1}^\square, \\ \text{primary} & X_{k+1} = Y_{k+1} + V_{k+1}(I_s - \eta_{k+1}^\square), & R_{k+1} = S_{k+1} - (AV_{k+1})(I_s - \eta_{k+1}^\square), \end{array}$$

respectively, for  $k = 0, 1, \dots$  with  $Y_0 = X_0$  and  $S_0 = R_0$  so that the primary and smoothed sequences influence one another, where  $V_{k+1} = V_k(I_s - \eta_k^\square) + \tilde{P}_k$  is an auxiliary matrix for  $\eta_0^\square = O \in \mathbb{R}^{s \times s}$  and  $V_0 = O \in \mathbb{R}^{n \times s}$ . Here,  $\tilde{P}_k \in \mathbb{R}^{n \times s}$  is a direction matrix in the recursion formula  $X_{k+1} = X_k + \tilde{P}_k$ . Again with a local minimization of the smoothed residual norm in choosing  $\eta_k^\square$ , a monotonically decreasing sequence of  $\|S_k\|$  is obtained. Note that the essential difference of Bl-CIRS from Gl-CIRS [1, Algorithm 3.1] is that the smoothing parameter  $\eta_k^\square$  of Bl-CIRS is an  $s$ -by- $s$  matrix instead of a scalar.

For numerical stability, Bl-CIRS needs to orthonormalize the columns of the auxiliary matrix  $V_k$  for each iteration. Let  $V_k = \tilde{Q}_k \tilde{\xi}_k$  be the QR decomposition of  $V_k$ , where  $\tilde{Q}_k \in \mathbb{R}^{n \times s}$  and  $\tilde{\xi}_k \in \mathbb{R}^{s \times s}$  are the Q- and R-factors, respectively. With the auxiliary matrix  $\tilde{\eta}_k^\square := \tilde{\xi}_k \eta_k^\square$ , the above formulas are equivalently rewritten as

$$\begin{array}{lll} \text{smoothed} & Y_{k+1} = Y_k + \tilde{Q}_{k+1}\tilde{\eta}_{k+1}^\square, & S_{k+1} = S_k - (A\tilde{Q}_{k+1})\tilde{\eta}_{k+1}^\square, \\ \text{primary} & X_{k+1} = Y_{k+1} + \tilde{Q}_{k+1}(\tilde{\xi}_{k+1} - \tilde{\eta}_{k+1}^\square), & R_{k+1} = S_{k+1} - (A\tilde{Q}_{k+1})(\tilde{\xi}_{k+1} - \tilde{\eta}_{k+1}^\square) \end{array}$$

together with  $V_{k+1} = \tilde{Q}_k(\tilde{\xi}_k - \tilde{\eta}_k^\square) + \tilde{P}_k$  for  $k = 0, 1, \dots$ .

Our main results via a rounding error analysis shows that Bl-CIRS with the orthonormalization strategy suppresses the residual gap.

**Theorem 1.** *In finite precision arithmetic, let  $X_k \in \mathbb{F}^{n \times s}$  and  $R_k \in \mathbb{F}^{n \times s}$  be the  $k$ th approximation and residual generated by the recursion formulas*

$$X_k = X_{k-1} + \hat{Q}_{k-1}\alpha_{k-1}^\square, \quad R_k = R_{k-1} - (A\hat{Q}_{k-1})\alpha_{k-1}^\square, \quad k = 0, 1, \dots, \quad (1)$$

respectively, where  $\mathbb{F} \subset \mathbb{R}$  is a set of floating point numbers and  $\hat{Q}_{k-1}$  is a  $Q$ -factor of the direction matrix  $P_{k-1}$  obtained from the QR decomposition with Givens rotations or Householder transformations. Then, with  $X_0 = O$ , the residual gap  $G_{R_k}$  satisfies the bound

$$\|G_{R_k}\| < (8\sqrt{s}\gamma_{m+3s} + \gamma_1) k \|A\| \max_{0 < i \leq k} \|X_i\| + k\gamma_1 \max_{0 < i \leq k} \|R_i\|,$$

where  $\gamma_k := k\mathbf{u}/(1 - k\mathbf{u})$  with a unit roundoff  $\mathbf{u}$  and  $m$  is the maximum number of nonzero entries per row of  $A$ .

In the case of Bl-CIRS, replacing  $X_k$  and  $R_k$  by  $Y_k$  and  $S_k$ , respectively, in (1), this theorem holds for the residual gap  $G_{S_k} = (B - AY_k) - S_k$ . Therefore, even when using an inexact orthonormalization for the columns of iteration matrices, Bl-CIRS in which the residual and approximation norms converge smoothly is indeed useful to reduce the residual gap. This theoretical result is consistent with our numerical results. Numerical experiments demonstrate that Bl-CIRS is effective for suppressing the residual gap and improving the attainable accuracy of the approximations.

## References

- [1] K. AIHARA, A. IMAKURA, AND K. MORIKIUNI, *Cross-interactive residual smoothing for global and block Lanczos-type solvers for linear systems with multiple right-hand sides*, SIAM J. Matrix Anal. Appl., 43 (2022), pp. 1308–1330.
- [2] K. AIHARA, R. KOMEYAMA, AND E. ISHIWATA, *Variants of residual smoothing with a small residual gap*, BIT, 59 (2019), pp. 565–584.
- [3] A. EL GUENNOUNI, K. JBILOU, AND H. SADOK, *A block version of BiCGSTAB for linear systems with multiple right-hand sides*, Electron. Trans. Numer. Anal., 16 (2003), pp. 129–142.
- [4] M. H. GUTKNECHT AND M. ROZLOŽNÍK, *Residual smoothing techniques: Do they improve the limiting accuracy of iterative solvers?*, BIT, 41 (2001), pp. 86–114.
- [5] K. JBILOU, *Smoothing iterative block methods for linear systems with multiple right-hand sides*, J. Comput. Appl. Math., 107 (1999), pp. 97–109.
- [6] R. KOMEYAMA, K. AIHARA, AND E. ISHIWATA, *Reconsideration of residual smoothing technique for improving the accuracy of approximate solutions of CGS-type iterative methods*, Trans. Japan Soc. Ind. Appl. Math., 28 (2018), pp. 18–38.
- [7] W. SCHÖNAUER, *Scientific Computing on Vector Computers*, Elsevier, Amsterdam, 1987.
- [8] R. WEISS, *Parameter-free iterative linear solvers*, Akademie Verlag, Berlin, 1996.

- [9] J. ZHANG AND F. DAI, *Global CGS algorithm for linear systems with multiple right-hand sides*, Numer. Math. J. Chinese Univ., 30 (2008), pp. 390–399.
- [10] L. ZHOU AND H. F. WALKER, *Residual smoothing techniques for iterative methods*, SIAM J. Sci. Comput., 15 (1994), pp. 297–312.

# Inverse Problems, Kronecker Products and Mixed Precision Computations

*James Nagy*

## Abstract

The gaming industry, machine learning (ML), and artificial intelligence (AI) are areas that require substantial computational resources and/or require very fast computations, but do not always require high accuracy in certain computational problems. This has motivated GPU vendors, such as NVIDIA, Google and AMD to manufacture hardware that can perform computations using low precision 16-bit floating-point formats [4]. Two examples are `bfloat16` and `FP16`. In comparison, IEEE single precision uses a 32-bit floating-point format, and double precision (e.g., the default in MATLAB) uses a 64-bit floating-point format. The use of 16-bit format can result in a  $4\times$  speedup compared to double precision, and certain hardware accelerators (called Tensor Cores) can further accelerate performance for operations such as matrix-vector multiplications [4].

The potential for much faster computations has fueled a growing interest in the last decade to use powerful GPU servers for scientific applications, and in particular to use mixed precision algorithms for problems that require high accuracy; that is, when possible, use low precision for speed, but mix in some high precision computations to improve accuracy. Recent previous work for solving general, well-conditioned linear systems, including iterative refinement [1, 2, 7], Cholesky factorization and least squares problems [1, 5], QR factorization [8], and GMRES [6].

Relatively little work has been done to exploit mixed precision computations for inverse problems, where the aim is to compute approximations of  $x$  from measured data,  $b$ , where

$$b = Ax + e. \quad (1)$$

$A$  is assumed to be a large, severely ill-conditioned matrix, and  $e$  represents unknown noise and other data measurement errors. In some applications  $A$  is known to high accuracy, while in other applications it may be that only an approximation of  $A$  is given, or that  $A \equiv A(y)$  is given in parametric form. Even in the case when  $A$  is known to high accuracy, due to the ill-posedness of the problem, and the presence of noise in the measured data, computing accurate approximations of  $x$  is a nontrivial task; special considerations, such as regularization approaches, need to be considered for these problems [3]. In this presentation we show how Kronecker product structure can be exploited and used in mixed precision algorithms for inverse problems.

## References

- [1] E. Carson, N. J. Higham, and S. Pranesh. Three-precision GMRES-based iterative refinement for least squares problems. *SIAM Journal on Scientific Computing*, 42(6):A4063–A4083, 2020.
- [2] A. Haidar, H. Bayraktar, S. Tomov, J. Dongarra, and N. J. Higham. Mixed-precision iterative refinement using tensor cores on GPUs to accelerate solution of linear systems. *Proceedings of the Royal Society A*, 476(2243):20200110, 2020.
- [3] P. C. Hansen. *Discrete Inverse Problems: Insight and Algorithms*. SIAM, Philadelphia, PA, 2010.
- [4] N. Higham and T. Mary. Mixed precision algorithms in numerical linear algebra. *Acta Numerica*, 31:347–414, 2022.

- [5] N. J. Higham and S. Pranesh. Exploiting lower precision arithmetic in solving symmetric positive definite linear systems and least squares problems. *SIAM Journal on Scientific Computing*, 43(1):A258–A277, 2021.
- [6] J. A. Loe, C. A. Glusa, I. Yamazaki, E. G. Boman, and S. Rajamanickam. A study of mixed precision strategies for gmres on gpus. *arXiv preprint arXiv:2105.07544*, 2021.
- [7] E. Oktay and E. Carson. Multistage mixed precision iterative refinement. *Numerical Linear Algebra with Applications*, 29(4):e2434, 2022.
- [8] L. M. Yang, A. Fox, and G. Sanders. Rounding error analysis of mixed precision block Householder QR algorithms. *SIAM J. Sci. Comput.*, 43:A1723–A1753, 2021.

# A fast algorithm for low-rank approximation with error control

*Yuji Nakatsukasa*

## Abstract

Computing a low-rank approximation to a large  $m \times n$  matrix  $A$  is a ubiquitous task in Numerical Linear Algebra (NLA), and possibly the single topic that contributed the most to making randomized NLA algorithms popular, trusted, and widely used. Typically [1, 5], the first step is to compute a random sketch of the form  $AS$  (or  $\hat{S}A$ , or both [12]), where the size of the sketch is at least the target rank, which is often unknown. Extensive theory is now available [5, 8, 11] that gives strong guarantees for the quality of the resulting approximation that hold with extremely high probability.

In this work we develop an algorithm for low-rank approximation that (i) requires only an  $O(1)$  sketch size, (ii) comes with high-probability error control to achieve a user-defined error tolerance, without requiring the knowledge of the rank, (iii) avoids computing orthogonal projections, and (iv) is based on the CUR decomposition [6] and its stable implementation [10], so inherits properties of  $A$  such as sparsity and nonnegativity, if present. These are achieved by bringing together techniques in randomized NLA algorithms, including CUR, subset selection methods [2, 9] based on a sketch-and-pivot strategy [3, 4], and error estimation via trace estimation [7].

The algorithm finds a near-optimal (up to a modest polynomial in  $r$ ) rank- $r$  approximation in  $O(N + (m + n)r^2)$  operations, where  $N$  is the cost of a matrix-vector multiplication with  $A$ . Advantages over the MATLAB routine `svdsketch` [13] include faster runtime and the ability to set the error tolerance to be smaller than  $\sqrt{u}$ , where  $u$  is the unit roundoff.

This talk is based on joint projects with the following collaborators: Per-Gunnar Martinsson and Nathaniel Pritchard; Anjali Narendran; and Taejun Park.

## References

- [1] K. L. Clarkson and D. P. Woodruff. Low-rank approximation and regression in input sparsity time. *Journal of the ACM*, 63(6):54, 2017.
- [2] A. Cortinovis and D. Kressner. Low-rank approximation in the frobenius norm by column and row subset selection. *SIAM J. Matrix Anal. Appl.*, 41(4):1651–1673, 2020.
- [3] Y. Dong and P.-G. Martinsson. Simpler is better: a comparative study of randomized pivoting algorithms for CUR and interpolative decompositions. *Adv. in Comput. Math.*, 49(4):66, 2023.
- [4] J. A. Duersch and M. Gu. Randomized projection for rank-revealing matrix factorizations and low-rank approximations. *SIAM Rev.*, 62(3):661–682, 2020.
- [5] N. Halko, P.-G. Martinsson, and J. A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev.*, 53(2):217–288, 2011.
- [6] M. W. Mahoney and P. Drineas. CUR matrix decompositions for improved data analysis. *Proc. Natl. Acad. Sci.*, 106(3):697–702, 2009.
- [7] P.-G. Martinsson and J. A. Tropp. Randomized numerical linear algebra: Foundations and algorithms. *Acta Numer.*, pages 403–572, 2020.
- [8] Y. Nakatsukasa. Fast and stable randomized low-rank matrix approximation. *arXiv preprint arXiv:2009.11392*, 2020.

- [9] A. Osinsky. Close to optimal column approximations with a single SVD. *arXiv preprint arXiv:2308.09068*, 2023.
- [10] T. Park and Y. Nakatsukasa. Accuracy and stability of CUR decompositions with oversampling. *arXiv preprint arXiv:2405.06375*, 2024.
- [11] J. A. Tropp, A. Yurtsever, M. Udell, and V. Cevher. Practical sketching algorithms for low-rank matrix approximation. *SIAM J. Matrix Anal. Appl.*, 38(4):1454–1485, 2017.
- [12] F. Woolfe, E. Liberty, V. Rokhlin, and M. Tygert. A fast randomized algorithm for the approximation of matrices. *Appl. Comput. Harmon. Anal.*, 25(3):335–366, 2008.
- [13] W. Yu, Y. Gu, and Y. Li. Efficient randomized algorithms for the fixed-precision low-rank matrix approximation. *SIAM J. Matrix Anal. Appl.*, 39(3):1339–1359, 2018.

# Optimal Matrix-Mimetic Tensor Algebras

*Elizabeth Newman, Katherine Keegan*

## Abstract

With the explosion of big data, the need for explainable data analysis tools, efficient representations, and structure-exploiting operations has exploded as well. Many data and operators are naturally multiway, and as a result, multilinear or tensor methods have revolutionized the interpretability of feature extraction, the compressibility of large-scale data, and the computational efficiency of multiway operations. Despite numerous successes, many tensor frameworks suffer from a so-called “curse of multidimensionality;” that is, that fundamental linear algebra properties break down in higher dimensions, particularly the notion of optimality. Recent advances in matrix-mimetic tensor frameworks have made it possible to preserve linear algebraic properties for multilinear analysis and, as a result, obtain optimal representations of multiway data.

Matrix mimeticity arises from interpreting tensors as operators that can be multiplied, factorized, and analyzed analogously to matrices. Underlying the tensor operation is an algebraic framework parameterized by an invertible linear transformation. Specifically, consider a third-order tensor  $\mathbf{A} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ ; i.e., a multiway arrays with rows, columns, and depth indices. We can view  $\mathbf{A}$  as an  $n_1 \times n_2$  matrix where each entry is a  $1 \times 1 \times n_3$  tube. We multiply tubes  $\mathbf{a}, \mathbf{b} \in \mathbb{R}^{1 \times 1 \times n_3}$  using the  $\star_{\mathbf{M}}$ -product [5] (the prefix is pronounced “star-M”) via

$$\mathbf{a} \star_{\mathbf{M}} \mathbf{b} = \text{vec}^{-1}(\mathbf{R}_{\mathbf{M}}[\mathbf{a}] \text{vec}(\mathbf{b})) \quad \text{where} \quad \mathbf{R}_{\mathbf{M}}[\mathbf{a}] = \mathbf{M}^{-1} \text{diag}(\mathbf{M} \text{vec}(\mathbf{a})) \mathbf{M}, \quad (1)$$

$\text{vec} : \mathbb{R}^{1 \times 1 \times n_3} \rightarrow \mathbb{R}_3^n$  is a bijective map that vectorizes tubes and  $\text{diag} : \mathbb{R}^{n_3} \rightarrow \mathbb{R}^{n_3 \times n_3}$  forms a diagonal matrix from the entries of a vector. We say the action  $\mathbf{a}$  on  $\mathbf{b}$  under the  $\star_{\mathbf{M}}$ -product is equivalent to left multiplication by the structured matrix  $\mathbf{R}_{\mathbf{M}}[\mathbf{a}]$ . A given invertible matrix  $\mathbf{M}$  thereby induces a matrix subalgebra that equips the vector space of tubes with a bilinear operation given by  $\mathbf{R}_{\mathbf{M}}[\cdot]$ ; the term *tensor algebra* refers to this operation.

We define tensor-tensor products analogously to matrix-matrix products by replacing scalar with tubal multiplication given by (1). Using MATLAB indexing notation, the tubal entrywise definition of the tensor-tensor product of  $\mathbf{A} \in \mathbb{R}^{n_1 \times m \times n_3}$  and  $\mathbf{B} \in \mathbb{R}^{m \times n_2 \times n_3}$  is

$$(\mathbf{A} \star_{\mathbf{M}} \mathbf{B})_{i_1, i_2, :} = \sum_{k=1}^m \mathbf{A}_{i_1, k, :} \star_{\mathbf{M}} \mathbf{B}_{k, i_2, :} \quad (2)$$

for  $i_1 = 1, \dots, n_1$  and  $i_2 = 1, \dots, n_2$ . Under the algebraically-consistent  $\star_{\mathbf{M}}$ -product, we obtain matrix-mimetic generalizations of  $\star_{\mathbf{M}}$ -rank, -orthogonality, -transposition, more [6]. Notably, we can define a tensor singular value decomposition that satisfies an Eckart-Young-like theorem, resulting in optimal, low-rank approximations of multiway data [7].

The choice of linear mapping  $\mathbf{M}$  and associated tensor algebra is crucial to approximation quality. Traditionally,  $\mathbf{M}$  is chosen heuristically to leverage expected correlations in the data. However, in many cases, these correlations are unknown and common heuristic mappings lead to suboptimal performance. This presentation, based on the work in [8], introduces  $\star_{\mathbf{M}}$ -optimization, an algorithm to learn optimal linear transformations and corresponding optimal tensor representations (e.g., low- $\star_{\mathbf{M}}$ -rank) simultaneously. The new framework explicitly captures the coupling between the transformation and representation by solving the bilevel optimization problem

$$\min_{\mathbf{M} \in \mathcal{O}_{n_3}} \Phi(\mathbf{M}, \mathcal{X}(\mathbf{M})) \quad \text{s.t.} \quad \mathcal{X}(\mathbf{M}) \in \arg \min_{\mathcal{X} \in \mathcal{X}} \Phi(\mathbf{M}, \mathcal{X}). \quad (3)$$

Here,  $\mathcal{X}$  is the desired representation belonging to feasible set  $\mathcal{X}$ , and  $\mathcal{X}(\mathbf{M})$  is an optimal representation for a given transformation,  $\mathbf{M}$ . Our goal is to learn an invertible  $\mathbf{M}$ , which we guarantee by optimizing over the orthogonal group of  $n_3 \times n_3$  matrices,  $\mathcal{O}_{n_3}$ . The objective function  $\Phi : \mathcal{O}_{n_3} \times \mathcal{X} \rightarrow \mathbb{R}$  measures the quality of the representation. We solve (3) for  $\mathbf{M}$  using Riemannian optimization over the orthogonal group [2, 1, 3].

A key innovation of  $\star_{\mathbf{M}}$ -optimization is the use of variable projection to form  $\mathcal{X}(\mathbf{M})$ , which eliminates the variable  $\mathcal{X}$  via partial optimization [4]. We heavily leverage the optimality of  $\star_{\mathbf{M}}$ -representations to guarantee the existence of an optimal  $\mathcal{X}(\mathbf{M})$ ; other comparable tensor approaches typically only guarantee quasi-optimality.

In the talk, we will highlight the generality of the  $\star_{\mathbf{M}}$ -optimization framework by considering two prototype problems for fitting tensor data and for finding compressed representations. We will present new theoretical results regarding the uniqueness and invariances of the  $\star_{\mathbf{M}}$ -operator and convergence guarantees of  $\star_{\mathbf{M}}$ -optimization. We will demonstrate the efficacy of learning the transformation and provide interpretable insight into  $\star_{\mathbf{M}}$ -optimization behavior through several numerical examples, including image compression and reduced order modeling.

## References

- [1] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*, Princeton University Press, Princeton, NJ, 2008.
- [2] N. Boumal, *An introduction to optimization on smooth manifolds*, Cambridge University Press, 2023, <https://doi.org/10.1017/9781009166164>.
- [3] A. Edelman, T. A. Arias, and S. T. Smith, *The geometry of algorithms with orthogonality constraints*, SIAM Journal on Matrix Analysis and Applications, 20 (1998), pp. 303–353, <https://doi.org/10.1137/S0895479895290954>.
- [4] G. H. Golub and V. Pereyra, *The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate*, SIAM Journal on Numerical Analysis, 10 (1973), pp. 413–432, <https://doi.org/10.1137/0710036>.
- [5] E. Kurnfeld, M. Kilmer, and S. Aeron, *Tensor-tensor products with invertible linear transforms*, Linear Algebra and its Applications, 485 (2015), pp. 545–570, <https://doi.org/10.1016/j.laa.2015.07.021>.
- [6] Misha E. Kilmer, Karen Braman, Ning Hao, and Randy C. Hoover. *Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging*. SIAM Journal on Matrix Analysis and Applications, 34(1):148–172, 2013, <https://doi.org/10.1137/110837711>.
- [7] M. E. Kilmer, L. Horesh, H. Avron, and E. Newman, *Tensor-tensor algebra for optimal representation and compression of multiway data*, Proceedings of the National Academy of Sciences of the United States of America, 118 (2021), <https://doi.org/10.1073/pnas.2015851118>.
- [8] Elizabeth Newman and Katherine Keegan. *Optimal matrix-mimetic tensor algebras via variable projection*, 2024, <https://arxiv.org/abs/2406.06942>.

# Recent Results on Improving Performance of Sparse Cholesky Factorization by Reordering Columns within Supernodes

*M. Ozan Karsavuran, Esmond G. Ng, Barry W. Peyton*

## Abstract

Let  $A$  be an  $n$  by  $n$  sparse symmetric positive definite matrix, and let  $A = LL^T$  be the Cholesky factorization of  $A$ , where  $L$  is a lower triangular matrix. It is well known that  $L$  suffers *fill* during such a factorization; that is,  $L$  will have nonzero entries in locations occupied by zeros in  $A$ . As a practical matter, it is important to limit the number of such fill entries in  $L$ . Consequently, software for solving a sparse symmetric positive definite linear system  $Ax = b$  via sparse Cholesky factorization requires the following four steps.

First, compute a fill-reducing ordering of  $A$  using either the *nested dissection* [5, 11] or the *minimum degree* [1, 6, 12, 15] ordering heuristic (the **ordering** step). Second, compute the needed information concerning and data structures for the sparse Cholesky factor matrix (the **symbolic factorization** step). Third, compute the sparse Cholesky factor within the data structures computed during the symbolic factorization step (the **numerical factorization** step). Fourth, solve the linear system by performing in succession a sparse forward solve and a sparse backward solve using the sparse Cholesky factor and its transpose, respectively (the **solve** step).

The authors of this work, along with J. L. Peyton, presented a thorough look [10] at some serial algorithms for the third step in the solution process (the numerical factorization step). Our goal was to improve the performance of serial sparse Cholesky factorization algorithms on multicore processors when only the multithreaded BLAS are used to parallelize the computation. Essentially, our first paper [10] explored what can be done for serial sparse Cholesky factorization using the techniques and methodology used in LAPACK.

Our primary contribution in [10] is the factorization method that we called *right-looking blocked* (RLB). Like all of the other factorization methods studied in [10], RLB relies on *supernodes* to obtain efficiency, where *supernodes* are, roughly speaking, sets of consecutive columns in the factor matrix sharing the same zero-nonzero structure. RLB, however, is unique among the factorization methods studied in [10] in that it requires *no floating-point working storage or assembly operations*; that is, the computation is performed in place within the data structures computed for the factor matrix during the symbolic factorization step. RLB is also unique among the factorization methods studied in [10] in that it is *entirely* dependent for efficiency on the existence of few and large dense blocks joining together pairs of supernodes in the factor matrix. Furthermore, the number of and size of these dense blocks are *crucially* dependent on how the columns of the factor matrix are ordered *within* supernodes. As a result, RLB is perfectly suited for studying the *quality* of algorithms for reordering columns within supernodes. It is precisely a study of this sort that will occupy our attention in this work. It should be noted that reordering the columns (and the corresponding rows) within each supernode does not change the number of nonzeros in the Cholesky factor.

Pichon, Faverge, Ramet, and Roman [14] were the first to take seriously the problem of reordering columns within supernodes, in that they were the first to treat it in a highly technical manner. They ingeniously formulated the underlying optimization problem as a *traveling salesman problem*, for which there exist powerful and effective heuristics. We will refer to their approach as TSP. The problem with their approach was not ordering quality; it was the cost, in time, of computing the needed TSP distances [8, 14]. In 2021, Jacquelin, Ng, and Peyton [9] devised a much faster way to compute the needed distances, which greatly reduces the runtimes for the TSP method.

In 2017, Jacquelin, Ng, and Peyton [8] proposed a simpler heuristic for reordering columns within supernodes based on *partition refinement* [13]. In their paper, they report faster runtimes for their method than TSP, while obtaining similar ordering quality. We will refer to their method as PR.

In this work, we perform a careful comparison of TSP and PR; we compare them, primarily, by measuring the impact of TSP and PR on RLB factorization times using Intel’s MKL multithreaded BLAS on 48 cores of our test machine. This approach is justifiable since, as alluded to above, the performance of RLB depends on the quality of the TSP or PR reorderings.

The comparisons are conducted using a set of large matrices from the SuiteSparse collection [3]. In our experiments, certain small supernodes are merged together to create a coarser supernode partition. This idea was first introduced by Ashcraft and Grimes [2], and was demonstrated to reduce the factorization time at the expense of a relatively small increase in the size of the data structures. Merging supernodes has become a standard practice in software for sparse symmetric factorization, such as MA57 [4] and MA87 [7].

In this presentation, we will describe two techniques for improving the quality of the TSP reorderings; we will show that the best results for TSP are obtained when the two techniques are combined. We will also introduce a new way to reorganize the PR reordering algorithm to make it much more time and storage efficient. In addition, we will introduce a single technique for modestly improving the quality of the PR reorderings. We will further show that the enhanced PR and enhanced TSP produce orderings of virtually equal quality. However, the former requires significantly less storage to implement and runs much faster than the latter.

## References

- [1] Patrick R. Amestoy, Timothy A. Davis, and Iain S. Duff. An approximate minimum degree ordering algorithm. *SIAM J. Matrix Anal. Appl.*, 17(4):886–905, 1996.
- [2] Cleve C. Ashcraft and Roger G. Grimes. The influence of relaxed supernode partitions on the multifrontal method. *ACM Trans. Math. Softw.*, 15(4):291–309, 1989.
- [3] Timothy A. Davis and Yifan Hu. The University of Florida sparse matrix collection. *ACM Trans. Math. Softw.*, 38(1):1–28, 2011.
- [4] Iain S. Duff. MA57—a code for the solution of sparse symmetric definite and indefinite systems. *ACM Trans. Math. Softw.*, 30:118–144, 2004.
- [5] Alan George. Nested dissection of a regular finite element mesh. *SIAM J. Numer. Anal.*, 10(2):345–363, 1973.
- [6] Alan George and Joseph W. H. Liu. The evolution of the minimum degree ordering algorithm. *SIAM Rev.*, 31(1):1–19, 1989.
- [7] Jonathan D. Hogg, John K. Reid, and Jennifer A. Scott. Design of a multicore sparse Cholesky factorization using DAGs. *SIAM J. Sci. Comput.*, 32(6):3627–3649, 2010.
- [8] Mathias Jacquelin, Esmond G. Ng, and Barry W. Peyton. Fast and effective reordering of columns within supernodes using partition refinement. In *2018 Proceedings of the Eighth SIAM Workshop on Combinatorial Scientific Computing*, pages 76–86, 2018.

- [9] Mathias Jacquelin, Esmond G. Ng, and Barry W. Peyton. Fast implementation of the Traveling-Salesman-Problem method for reordering columns within supernodes. *SIAM J. Matrix Anal. Appl.*, 42(3):1337–1364, 2021.
- [10] M. Ozan Karsavuran, Esmond G. Ng, Barry W. Peyton, and Jonathan L. Peyton. Some new techniques to use in serial sparse Cholesky factorization algorithms. Submitted to TOMS, 2024.
- [11] George Karypis and Vipin Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. Sci. Comput.*, 20(1):359–392, 1999.
- [12] Joseph W. H. Liu. Modification of the minimum-degree algorithm by multiple elimination. *ACM Trans. Math. Softw.*, 11(2):141–153, 1985.
- [13] Robert Paige and Robert E. Tarjan. Three partition refinement algorithms. *SIAM J. Comput.*, 16(6):973–989, 1987.
- [14] Gregoire Pichon, Mathieu Faverge, Pierre Ramet, and Jean Roman. Reordering strategy for blocking optimization in sparse linear solvers. *SIAM J. Matrix Anal. Appl.*, 38(1):226–248, 2017.
- [15] W. F. Tinney and J. W. Walker. Direct solution of sparse network equations by optimally ordered triangular factorization. *Proc. IEEE*, 55:1801–1809, 1967.

# Riemannian optimization for matrix nearness problems

*Vanni Noferini, Froilán Dopico, Miryam Gnazzo, Lauri Nyman, Federico Poloni*

## Abstract

Matrix nearness problems are central to numerical linear algebra and matrix theory. As highlighted in [6], these problems have wide-ranging applications. The proposed talk, outlined below, is based on several recent papers of mine, written with different coauthors [3, 4, 7].

## Overview of matrix nearness problems

Let us start with a general description, inspired in part by Nick Higham's PhD thesis [5].

A matrix nearness problem involves finding a matrix  $B$  that is closest to a given matrix  $A$ , such that  $B$  satisfies a specific property  $\mathfrak{P}$  which  $A$  does not. Formally, if  $\mathcal{Q}$  represents the set of matrices that possess property  $\mathfrak{P}$ , the goal is to solve the following optimization problem:

$$\min \|A - X\| \quad \text{subject to} \quad X \in \mathcal{Q}. \quad (1)$$

The distance to be minimized in (1) is typically the Euclidean one, i.e., the distance induced by the Frobenius norm  $\|X\|_F^2 = \text{tr}(X^*X)$ , though other options can also be considered. Matrix nearness problems can be generalized to matrix pencils or matrix polynomials. For example, given a matrix polynomial  $A(z) = \sum_{i=0}^d A_i z^i$ , the objective becomes finding a polynomial  $B(z) = \sum_{i=0}^d B_i z^i$  that minimizes the squared distance  $\sum_{i=0}^d \|A_i - B_i\|^2$  while ensuring that  $B(z)$  has the desired property  $\mathfrak{P}$ , which  $A(z)$  lacks.

Some matrix nearness problems have well-understood, efficient solutions. For example, by the Eckart–Young–Mirsky theorem, the nearest singular matrix to a full-rank matrix  $A \in GL(n, \mathbb{C})$  can be found via singular value decomposition (SVD). The solution is  $B = \sum_{i=0}^{n-1} \sigma_i u_i v_i^*$ , with the distance given by the smallest singular value  $\sigma_n$ .

However, many matrix nearness problems are more difficult, and have been either shown or conjectured to be NP-hard. The feasible set  $\mathcal{Q}$  is often non-convex, making it challenging to find anything beyond a local minimum. Moreover, optimization algorithms that attempt the task are frequently quite slow and inefficient, making it almost impossible in practice to find even a local minimizer when the input size grows beyond very small matrices.

## A new approach

In this talk, I will propose a new method to solve matrix nearness problems. There are three key features of the new approach:

1. It can handle a broad class of matrix nearness problems, including many described in the literature;
2. In extensive numerical experiments, the new method consistently outperforms existing algorithms, especially in challenging cases;

3. The problem is reformulated as an optimization task over Riemannian manifolds, offering a novel approach to previously intractable problems.

Our method is based on a key insight: many matrix nearness problems become more tractable when supplemented with additional information about the minimizer. This is akin to the concept of an “oracle” in theoretical computer science: an abstract machine that can solve specific problem instances in one step.

Let us denote the supplementary information about the optimizer by  $\theta$ . The problem can then be stiffened by restricting the feasible set to matrices that share this information. Specifically, if we decompose the set  $\mathcal{Q}$  as  $\mathcal{Q} = \bigcup_{\theta} \mathcal{Q}_{\theta}$ , where  $\mathcal{Q}_{\theta}$  represents matrices with property  $\mathfrak{P}$  and the additional attribute  $\theta$ , we can solve the restricted problem:

$$f(\theta) = \min \|A - X\| \quad \text{subject to} \quad X \in \mathcal{Q}_{\theta}.$$

The original problem can then be equivalently reformulated by optimizing over  $\theta$ :

$$\min_{\theta} f(\theta).$$

This often reduces the problem to an optimization task over a Riemannian manifold. For example, if  $\theta$  is an eigenvector, the optimization is over the unit sphere. If  $\theta$  represents a set of  $d$  independent eigenvectors, the optimization is over the Grassmann manifold of  $d$ -dimensional subspaces.

Of course, the idea to use Riemannian optimization to solve nearness problems is not new. Many works have applied manifold optimization to specific matrix nearness problems, but typically in a much more direct manner than the approach that I will present in this talk. To give but a couple of the many examples, Vandereycken [8] addressed low-rank matrix completion via optimization on fixed-rank manifolds, and Borsdorf [1] used augmented Lagrangian techniques for a chemistry-related problem over the Stiefel manifold. Oracle-based strategies have also previously appeared, such as the method by Byers [2] for finding the distance to Hurwitz instability.

However, our recent work has further advanced these ideas, and its distinctive feature is to use at the same time both oracles and Riemannian optimization, as well as other classical optimization tools such as regularization. It is, to my knowledge, the first time that this strategy has been attempted in a systematic way to tackle a wide range of matrix nearness problems.

## A two-level optimization framework

More in detail, our method introduces a two-level optimization framework:

- **Inner Problem: Minimize the distance over  $\mathcal{Q}_{\theta}$ .** The inner subproblem is chosen so that it has a closed-form solution, or at least so that its solution can be computed cheaply.
- **Outer Problem: Minimize  $f(\theta)$  over the manifold of all possible  $\theta$ .** Riemannian optimization is then used to efficiently solve the outer problem.

We refer to this strategy as the *Riemann-Oracle* approach.

In the talk, I plan to describe how and why the Riemann-Oracle method can be applied to several matrix nearness problems of practical importance, including applications in numerical linear algebra, control theory, and computer algebra. Our experiments show that this approach consistently outperforms many specialized algorithms in both speed and accuracy.

I will explore in the talk both the theoretical details and the practical implementation aspects of this method. Furthermore, I will provide numerical experiments demonstrating concrete success stories. Examples include:

1. Finding the nearest matrix whose eigenvalues are all in a given set  $\Omega$  (nearest stable matrix) [7]. Here the information  $\theta$  is a unitary matrix that brings the solution into Schur form;
2. Finding the nearest singular pencil to a given one [3], including variants of the problem such as the nearest pencil with a prescribed minimal index. Here the information  $\theta$  is a pair of unitary matrices that bring the solution into generalized Schur form;
3. Finding the nearest singular matrix with an additional linear constraint (e.g. sparsity pattern, Toeplitz structure, etc.) [4]. Here the information  $\theta$  is an eigenvector;
4. Finding the nearest matrix of prescribed rank  $r$  and an additional linear constraint [4]. Here the information  $\theta$  is a set of  $n - r$  eigenvectors;
5. Finding the nearest unstable matrix, i.e., requiring at least one eigenvalue to lie outside the set  $\Omega$  [4]. Here the information  $\theta$  is an eigenvector;
6. Finding the nearest matrix polynomial of prescribed rank [4]. Here the information  $\theta$  is a null polynomial vector;
7. Finding the approximate GCD of two scalar polynomials [4]. Here the information  $\theta$  is a vector containing the coefficients of two polynomials related to the problem.

## References

- [1] R. Borsdorf. An algorithm for finding an optimal projection of a symmetric matrix onto a diagonal matrix. *SIAM J. Matrix Anal. Appl.*, 35(1):198–224, 2014.
- [2] R. Byers. A bisection method for measuring the distance of a stable matrix to the unstable matrices. *SIAM J. Sci. Stat. Comput.*, 9(5):875–881, 1988.
- [3] F. Dopico, V. Noferini, and L. Nyman. A Riemannian optimization method to compute the nearest singular pencil. *SIAM J. Matrix Anal. Appl.*, To appear, 2024.
- [4] M. Gnazzo, V. Noferini, L. Nyman, and F. Poloni. Riemann-Oracle: A general-purpose Riemannian optimizer to solve nearness problems in matrix theory. Preprint, 2024.
- [5] N. J. Higham. *Nearness Problems in Numerical Linear Algebra*. PhD thesis, University of Manchester, 1985.
- [6] N. J. Higham. Matrix nearness problems and applications. In *M. J. C. Gover and S. Barnett, editors, Applications of Matrix Theory*, pages 1–27. Oxford University Press, 1989.

- [7] V. Noferini and F. Poloni. Nearest  $\Omega$ -stable matrix via Riemannian optimization. *Numer. Math.*, 148:817–851, 2021.
- [8] B. Vandereycken. Low-rank matrix completion by Riemannian optimization. *SIAM Journal on Optimization*, 23(2):1214–1236, 2013.

# Recent Advances in Mixed-Precision (Hybrid) Iterative Methods

Eda Oktay, Erin Carson

## Abstract

Mixed-precision hardware has recently become commercially available, and more than 25% of the supercomputers in the TOP500 list now have mixed-precision capabilities. Using lower precision in algorithms can be beneficial in terms of reducing both computation and communication costs. According to the recently developed mixed-precision benchmark, HPL-MxP, multiple supercomputers today already exceed exascale ( $10^{18}$  floating-point operations per second) performance through the use of mixed-precision computations. Many current efforts are focused on developing mixed-precision numerical linear algebra algorithms, which will lead to speedups in real applications. These new algorithms are increasingly being implemented in libraries, such as the MAGMA library.

Motivated by this, the aim of this talk is to discuss recent advances in developing and analyzing mixed-precision variants of iterative methods. Iterative methods for solving linear systems and least squares problems are useful in practice when the coefficient matrix is large and sparse or not explicitly stored and/or when accuracy less than machine precision is sufficient. An iterative method starts with an initial guess and then iteratively improves the solution to the desired accuracy. One can use stationary methods, Krylov subspace methods, or some hybrid approach, depending on the problem. We focus on *hybrid methods*, where we use a Krylov subspace method as an inner solver of a variant of Newton's approach (stationary method), such as RQI and iterative refinement.

Iterative methods can be used to improve the accuracy of solutions to least squares (LS) problems  $\min_x \|b - Ax\|_2$ , where  $A \in \mathbb{R}^{m \times n}$ . Using the QR factorization  $A = [Q_1 \ Q_2][R \ 0]^T$ , the solution to the LS problem is given by  $x = U^{-1}Q_1^T b$  and the residual by  $r = \|b - Ax\|_2 = \|Q_2^T b\|_2$ . The LS problem can also be solved via the normal equations,  $A^T Ax = A^T b$ , which are equivalent to the augmented system [1]

$$\underbrace{\begin{bmatrix} I^{m \times m} & A \\ A^T & 0 \end{bmatrix}}_{\tilde{A}} \underbrace{\begin{bmatrix} r \\ x \end{bmatrix}}_{\tilde{x}} = \underbrace{\begin{bmatrix} b \\ 0 \end{bmatrix}}_{\tilde{b}} \quad \text{or} \quad \tilde{A}\tilde{x} = \tilde{b}.$$

If  $m > n$ , then the system is called overdetermined, and if  $m < n$ , it is underdetermined. Weighted LS (WLS) is used when there are discrepant rows in  $A$ . In this case, weights can be assigned to these rows to minimize discrepancy. In classical least squares, there is an assumption that perturbations are confined to the vector  $b$ . This is not necessarily realistic in practice. If  $A$  and  $b$  may both be perturbed ( $\hat{A}, \hat{b}$ , respectively) so that  $\hat{b}$  is in column space of  $\hat{A}$ , this problem is called total LS (TLS).

Krylov subspace methods work by selecting approximate solutions from a Krylov subspace. The search space is formed via nested Krylov subspaces, and the solution is obtained from a sequence of projections onto the search space. Although these solvers can be fast and/or stable, for large problems, they may not be memory efficient and slow down performance. To speed up and exploit parallelism, techniques such as mixed-precision can be used.

Error analysis is important for determining how rounding errors propagate in computations and identifying potential sources of amplification. For a function  $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ , the backward error in the approximation  $y$  to  $f(x)$  is the smallest  $\Delta x$  such that  $y = f(x + \Delta x)$ , i.e., [10]

$$\eta(y) = \min\{\epsilon : y = f(x + \Delta x), \|\Delta x\| \leq \epsilon\|x\|\}.$$

Backward error analysis [9] aims to derive a bound on the backward error. If the backward error is small, then we say the algorithm is backward stable. The forward error measures the difference between the computed and the exact solution. As defined in [10], the relative forward error of  $y \approx f(x)$  can be bounded in terms of the relative backward error  $\eta(y)$  by

$$\frac{\|y - f(x)\|}{\|f\|} \leq \text{cond}(f, x)\eta(y) + O(\eta(y))^2,$$

where

$$\text{cond}(f, x) = \lim_{\epsilon \rightarrow 0} \sup_{\|\Delta x\| \leq \epsilon \|x\|} \frac{\|f(x + \Delta x) - f(x)\|}{\epsilon \|f(x)\|}$$

is the condition number, which measures the sensitivity of the solution to small perturbations in the input data.

### Mixed-precision Rayleigh quotient iteration for total least squares problems

We first focus on the use of Rayleigh quotient iteration (RQI) to solve the TLS problem, which is the approach advocated in [2] for large-scale problems, and introduce a mixed-precision variant of the RQI-PCGTLs algorithm (RQI-PCGTLs-MP) [8]. This approach solves the eigenvalue problem

$$\begin{bmatrix} A^T A & A^T b \\ b^T A & b^T b \end{bmatrix} \begin{bmatrix} x \\ -1 \end{bmatrix} = \lambda \begin{bmatrix} x \\ -1 \end{bmatrix}$$

to find  $x = x_{TLS}$ , where  $\lambda = \sigma_{n+1}^2$ , and  $\sigma_{n+1}^2$  is the smallest singular value of  $[A, b]$ . Our approach potentially decreases the computational cost of RQI-PCGTLs by using up to three different precisions in the algorithm. Moreover, to enable the use of lower precision for more ill-conditioned systems, we use the R-factor from the Householder QR factorization of  $A$  instead of the Cholesky factorization of  $A^T A$  within RQI-PCGTLs-MP. We discuss the convergence and accuracy of our algorithm and derive two theoretical constraints on the precision that can be used for the construction of the preconditioner within the inner solver. To evaluate to what extent the computational cost can be reduced by using the mixed-precision variant with Householder QR factorization, we construct a performance model. Our numerical experiments and performance model show that one can get up to  $4\times$  speedup while keeping the working precision accuracy when fp16 is used in computing QR factors.

### GMRES-based iterative refinement and its variants

Another variant of Newton's method is the iterative refinement (IR) algorithm. As RQI, IR algorithms require a linear solver in each outer iteration. The standard IR (we refer to as SIR) algorithm in [9] first computes the initial approximation using Gaussian elimination with partial pivoting and uses approximate LU factors of  $A$  to solve for the correction term which then refines the current solution. To increase the range of problems that can be solved with IR, a Krylov subspace method, such as preconditioned GMRES, can be used to solve the linear systems as in RQI-PCGTLs; this three-precision approach is called GMRES-IR [3]. GMRES-IR uses precisions with unit round-off  $u_f$  for LU factorization,  $u_r$  for residual computation, and  $u$  for storing data and solution. For stability analysis of methods such as IR variants, we can derive forward and error bounds under a constraint on the conditioning of the coefficient matrix,  $\kappa(A)$ . For a non-singular square matrix, the condition number is defined as  $\kappa_p(A) = \|A\|_p \|A^{-1}\|_p$  with the associated norm

$p$ . As long as  $\kappa_\infty(A) \leq u^{-1/2}u_f^{-1}$  and  $u_r = u^2$ , GMRES-IR provides accurate solutions with the forward and (normwise) backward errors

$$\frac{\|\hat{x} - x\|_\infty}{\|x\|_\infty} \approx \mathcal{O}(u) \quad \text{and} \quad \frac{\|b - A\hat{x}\|_\infty}{\|A\|_\infty\|x\|_\infty + \|b\|_\infty} \approx \mathcal{O}(u),$$

respectively, while SIR is guaranteed to have this forward error only if  $\kappa_\infty(A) \leq u_f^{-1}$  and  $u_r = u^2$ .

GMRES-IR can be much more expensive than SIR, depending on the number of iterations performed. One way to speed up the convergence of the GMRES solver is using recycling. In an effort to reduce the overall computational cost of the GMRES solves within GMRES-IR, we introduce a recycled GMRES-based iterative refinement algorithm called RGMRES-IR [6]. The algorithm starts with computing the LU factors of  $A$  and computing the initial approximate solution in the same manner as GMRES-IR. Instead of preconditioned GMRES, however, the algorithm uses preconditioned GCRO-DR to solve the correction equation. In the RGMRES-IR algorithm, as in GMRES-IR, we use three precisions. Numerical experiments show that RGMRES-IR decreases the total GMRES iterations performed, especially when the matrix is badly conditioned. Even when GMRES-IR cannot converge, we observe that our variant can still converge.

Overdetermined standard least squares problems can be solved by using mixed-precision within the iterative refinement approach. It has been shown that mixed-precision GMRES-IR can also be used, in an approach termed GMRES-LSIR [4]. GMRES-LSIR solves the augmented system using GMRES preconditioned by a preconditioner  $M$  computed using the QR factors of  $A$ :

$$M = \begin{bmatrix} \alpha I & Q_1 U \\ U^T Q_1^T & 0 \end{bmatrix},$$

where  $A = Q_1 U$  is the thin QR factorization of  $A$ . As long as  $\kappa_\infty(A) \leq u^{-1/2}u_f^{-1}$ , and assuming  $u_r = u^2$ , GMRES-LSIR provides  $\mathcal{O}(u)$  backward and forward error. Furthermore, using the left preconditioner  $M$ , the conditioning of the preconditioned augmented matrix can be bounded by

$$\kappa_\infty(M^{-1}\tilde{A}) \lesssim (1 + 2m\sqrt{n}\tilde{\gamma}_{mn}^f\kappa_\infty(A))^2, \quad \text{where } \tilde{\gamma}_{mn}^f = \frac{cmn}{1 - mn u_f},$$

and  $c$  is a small constant. In practice, we often encounter types of least squares problems beyond standard least squares, including the WLS problem  $\min_x \|D^{1/2}(b - Ax)\|_2$ , where  $D^{1/2}$  is a diagonal matrix of weights, which is possibly ill-conditioned. WLS problems can be solved via the normal equations or the corresponding augmented system,

$$A^T D A x = A^T D b \quad \text{and} \quad \begin{bmatrix} \alpha D^{-1} & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} \alpha^{-1} y \\ x \end{bmatrix} = \begin{bmatrix} b \\ 0 \end{bmatrix},$$

respectively, where  $y = D(b - Ax)$ ,  $\alpha$  is the scaling factor for stability. We present the FGMRES-WLSIR algorithm, a variant of GMRES-LSIR for solving WLS problems using flexible GMRES (FGMRES), and discuss and analyze two different preconditioners [5]; a left preconditioner and a block diagonal split preconditioner,

$$M_l = \begin{bmatrix} \alpha D^{-1} & Q \hat{R} \\ \hat{R}^T Q^T & 0 \end{bmatrix}, \quad \text{and} \quad M_b = \begin{bmatrix} \alpha D^{-1} & 0 \\ 0 & \hat{C} \end{bmatrix},$$

respectively, where  $\hat{C} \approx \alpha^{-1} A^T D A$  is a symmetric positive definite approximation to the Schur complement.

## Multistage mixed-precision iterative refinement

In some cases, SIR can fail depending on the conditioning of the matrix and the precisions used. However, using GMRES-IR can be more expensive since one GMRES-IR iteration is more expensive than one SIR iteration. To benefit from both approaches and their variants, we propose a multistage IR approach (MSIR) to reduce the computation cost while improving applicability [7]. Our approach automatically switches between solvers and precisions if slow convergence (of the refinement scheme itself or of the inner GMRES solves) is detected using stopping criteria. With MSIR we attempt to use “stronger” solvers before resorting to increasing the precision of the factorization, and when executing a GMRES-based refinement algorithm, we modify the stopping criteria to also restrict the number of GMRES iterations per refinement step. Our experiments show that since the algorithmic variants often outperform what is dictated by the theoretical condition number constraints there can be an advantage to first trying other solvers before resorting to increasing the precision and refactorizing.

## References

- [1] Åke Björck (1967) *Iterative refinement of linear least squares solutions i.* BIT Numerical Mathematics, 7(4):257–278.
- [2] Åke Björck, Pinar Heggernes, and Pontus Matstoms (2000) *Methods for large scale total least squares problems.* SIAM Journal on Matrix Analysis and Applications, 22(2):413–429.
- [3] Erin Carson and Nicholas J. Higham (2017) *A new analysis of iterative refinement and its application to accurate solution of ill-conditioned sparse linear systems.* SIAM Journal on Scientific Computing, 39(6):A2834–A2856.
- [4] Erin Carson, Nicholas J. Higham, and Srikara Pranesh (2020) *Three-precision GMRES-based iterative refinement for least squares problems.* SIAM Journal on Scientific Computing, 42(6):A4063–A4083.
- [5] Erin Carson and Eda Oktay (2024) *Mixed precision FGMRES-based iterative refinement for weighted least squares.* arXiv preprint arXiv:2401.03755.
- [6] Eda Oktay and Erin Carson (2022) *Mixed precision GMRES-based iterative refinement with recycling.* In Jan Chleboun and Pavel Kůš and Jan Papež and Miroslav Rozložník and Karel Segeth and Jakub Šístek, editors, Programs and Algorithms of Numerical Mathematics, Proceedings of Seminar, pp.149–162. Institute of Mathematics CAS.
- [7] Eda Oktay and Erin Carson (2022) *Multistage mixed precision iterative refinement.* Numerical Linear Algebra with Applications, 29(4):e2434.
- [8] Eda Oktay and Erin Carson (2024) *Mixed precision Rayleigh quotient iteration for total least squares problems.* Numerical Algorithms, 96: 777–798.
- [9] James Hardy Wilkinson (1963) *Rounding errors in algebraic processes.* Prentice-Hall.
- [10] Nicholas J. Higham (2002) *Accuracy and stability of numerical algorithms.* SIAM.

# Mixed Precision Iterative Refinement for Linear Inverse Problems

*Lucas Onisk and James G. Nagy*

## Abstract

We are interested in linear discrete inverse problems which involve the reconstruction of objects or signals from noisy observed data. The available linear system is given by

$$Ax + e = b, \quad (1)$$

where  $A \in \mathbb{R}^{m \times n}$ ,  $m \geq n$  is a matrix whose singular values decay without significant gap and cluster at the origin (i.e., the matrix is ill-conditioned). Discrete inverse problems can arise through the discretization of Fredholm integral equations of the first kind; see [5, 3], but can also arise in massive data streaming problems such as the training of the random feature model in machine learning [8] or limited angle imaging problems including, for example, those from medical imaging [2]. To derive a meaningful solution from the available problem, regularization is needed.

In Tikhonov regularization, the least-squares problem associated with (1) is replaced by the penalized least-squares problem

$$\min_{x \in \mathbb{R}^n} \left\{ \|Ax - b\|_2^2 + \alpha^2 \|Lx\|_2^2 \right\} \quad (2)$$

where  $\alpha > 0$  is a regularization parameter that balances the sensitivity of the solution vector to the error in  $b$ , as well as the closeness to the desired solution of the unavailable error-free problem. When the regularization matrix  $L \in \mathbb{R}^{s \times n}$  is chosen so that the null spaces of  $A$  and  $L$  trivially intersect then the solution of (2) may be written in closed form.

Iterative refinement (IR) has long been utilized as an iterative strategy to improve the accuracy of numerical solutions to linear systems of equations. Recent works by Higham and collaborators have considered the use of IR in conjunction with mixed precision computing in light of recent advancements in hardware capabilities; see [6, 1]. Our interests of studying IR applied to the Tikhonov problem were motivated by the work [7] which considered the solution of symmetric positive definite linear systems and least-squares problems in mixed precision which showed regularization to be a key requirement when computing low precision factorizations.

The  $k^{th}$  iterate of IR applied to the Tikhonov problem in standard form,  $(A^T A + \alpha^2 I)x^{(k)} = A^T b$ , where  $L = I$  may be written recursively as

$$x^{(k)} = x^{(k-1)} + (A^T A + \alpha^2 I)^{-1} A^T r^{(k-1)} - \alpha^2 (A^T A + \alpha^2 I)^{-1} x^{(k-1)} \quad (3)$$

where  $r^{(k-1)} = b - Ax^{(k-1)}$  denotes the  $(k-1)^{th}$  residual. Riley in [9] and Golub in [4] note that the IR procedure in (3) is equivalent to iterated Tikhonov regularization in exact arithmetic whose  $k^{th}$  iterate is given by

$$x^{(k)} = x^{(k-1)} + (A^T A + \alpha^2 I)^{-1} A^T r^{(k-1)}$$

which may be interpreted as a preconditioned Landweber method, or, from a mathematical optimization point of view - a preconditioned gradient descent method with fixed step size.

To better understand the application of mixed precision IR applied to the Tikhonov problem we derive a methodology to formulate the iterates as filtered solutions by writing them as a recursive

relationship between the iterates of preconditioned Landweber with a Tikhonov-type preconditioner and previous iterates. A filtered solution is of the form

$$x_{filt} = \sum_j \phi_j \frac{u_j^T b}{\sigma_j} v_j$$

where vectors  $u_j$  and  $v_j$  correspond to left and right singular vectors of  $A$ , respectively. The  $\sigma_j$  correspond to the singular values of  $A$ . An intelligent selection of the filter factors  $\phi_j$  can remove deleterious components of the approximate solution to the least-squares problem stemming from (1). By considering a filtered solution, we are able to study the effect that each level of precision utilized in IR has on (i) the quality of the approximate solution and (ii) the number of iterations the algorithm requires to terminate according to some termination criterion. We demonstrate in our numerical results that mixed precision IR on the Tikhonov problem gives comparable or superior accuracy against results computed in double precision as well as another benchmark which supports its use in modern applications that natively support mixed precision floating-point arithmetic.

## References

- [1] E. Carson, N. J. Higham, and S. Pranesh, *Three-precision GMRES-based iterative refinement for least squares problems*, SIAM Journal on Scientific Computing, 42 (2020), pp. A4063–A4083.
- [2] J. Chung and L. Nguyen, *Motion estimation and correction in photoacoustic tomographic reconstruction*, SIAM Journal of Imaging Sciences, 10 (2017), pp. 216–242.
- [3] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization of Inverse Problems*, Kluwer, Dordrecht, 1996.
- [4] G. Golub, *Numerical methods for solving linear least squares problems*, Numerische Mathematik, 7 (1965), pp. 206–216.
- [5] P. C. Hansen, *Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion*, SIAM, Philadelphia, 1998.
- [6] N. Higham and T. Mary, *Mixed precision algorithms in numerical linear algebra*, Acta Numerica, 31 (2022), pp. 347–414.
- [7] N. J. Higham and S. Pranesh, *Exploiting lower precision arithmetic in solving symmetric positive definite linear systems and least squares problems*, SIAM Journal on Scientific Computing, 43 (2021), pp. A258–A277.
- [8] A. Rahim and B. Recht, *Random features for large-scale kernel machines*, Advances in Neural Information Processing Systems, 20 (2007), p. 1177–1184.
- [9] J. D. Riley, *Solving systems of linear equations with a positive definite, symmetric, but possibly ill-conditioned matrix*, Mathematical Tables and other Aids to Computation, (1955), pp. 96–101.

Absorbing boundary conditions form Padé approximants (sometimes):  
continued fractions are the key

*Michal Ostrata, Lukáš Jakabčin, Martin J. Gander*

## Abstract

The solution process of problems on unbounded domains usually require a domain truncation and therefore artificial boundary conditions, leading to techniques such as *perfectly matched layers* (PML) or *absorbing boundary conditions* (ABC), see [1, 2] for references. To be concrete, taking  $\Omega \subset \mathbb{R}^2$  as an infinite strip (sometimes called a *waveguide*), then the original problem (or its discretization)

$$\begin{aligned}\mathcal{L}u &= f && \text{in } \Omega, \\ \mathcal{B}u &= g && \text{on } \partial\Omega,\end{aligned}\quad (\text{or } L^\infty \mathbf{u} = \mathbf{f}^\infty)$$

is truncated to

$$\begin{aligned}\mathcal{L}v &= f && \text{in } \Omega^{\text{trunc}}, \\ \mathcal{B}^{\text{trunc}}v &= g && \text{on } \partial\Omega^{\text{trunc}},\end{aligned}\quad (\text{or } Lv = \mathbf{f})$$

where  $\hat{\Omega}$  is the region in which we want to approximately compute  $u$ ,  $\Omega^{\text{ABC}}$  is the bounded region with which we replace the (originally unbounded)  $\Omega \setminus \hat{\Omega}$  and  $\Omega^{\text{trunc}} = \hat{\Omega} \cup \Omega^{\text{ABC}}$  is bounded. We have  $\mathcal{B}^{\text{trunc}} = \mathcal{B}$  wherever  $\partial\Omega^{\text{trunc}}$  coincide with  $\partial\Omega$  and usually introduce a simple boundary condition along the remainder of  $\partial\Omega^{\text{trunc}}$ , e.g., Dirichlet. Naturally, this is also reflected at the discrete level where the infinite matrix  $L^\infty$  is replaced by a finite matrix  $L$ , which is identical with  $L^\infty$  for the unknowns of the interior of  $\Omega^{\text{trunc}}$  and those where  $\partial\Omega^{\text{trunc}}$  coincide with  $\partial\Omega$ . Domain truncation is also important in domain decomposition methods where a given computational domain is decomposed into many smaller subdomains, and then subdomain solutions are computed independently in parallel. The solutions on the smaller subdomains can naturally be interpreted as solutions on truncated domains, and thus it is of interest to use ABC or PML techniques at the interfaces between the subdomains. The classical Schwarz method uses Dirichlet transmission conditions between subdomains and an overlap to achieve convergence [2]. The overlap coupled with the Dirichlet boundary condition can be thus interpreted as a specific ABC once the unknowns of the overlap are folded onto the interface – an idea that inspired number of iterative solvers, see [1] and the references therein.

An interesting question of a *discrete optimized ABC/PML* for problems with finite difference grids has been discussed in [3] for  $\mathcal{L}$  being the Laplacian and then extended to the Helmholtz equation in [4] – in both of these, the authors answer the question:

*Having  $\Omega^{\text{ABC}}$  fixed, what is the best mesh for finite difference discretization (possibly staggered) so that  $v|_{\hat{\Omega}} \approx u|_{\hat{\Omega}}$ ?*

Here, we are interested in the complementary question:

*If the discretization method is fixed, what is the effect of prolonging the truncation domain  $\Omega^{\text{ABC}}$  on  $v|_{\hat{\Omega}} \approx u|_{\hat{\Omega}}$ ?*

We also start with  $\mathcal{L}$  being the Laplacian and, after discretization, start with the known correspondence of the discrete ABC and the Schur complement, see [1, Remark 14 and below]. We use its eigendecomposition, which is closely linked with its Fourier analysis (sometimes also called the frequency domain analysis), and show that in the spectral domain the ABC is naturally represented as the  $i$ -th convergent of a particular *continued fraction*, namely

$$\text{ABC}(z) \sim 2 + z - \frac{1}{2 + z - \frac{1}{\ddots 2 + z - \frac{1}{2 + z}}},$$

where the fraction has “ $i$  levels” and  $z$  corresponds to the Fourier frequency. After relating  $i$  to the prolongation of  $\Omega^{\text{ABC}}$ , as posed in our question, we also show that the *infinite* continued fraction (i.e., without stopping after  $i$  levels) gives a natural representation of the *optimal* ABC for the infinite problem  $L\mathbf{u} = \mathbf{f}$ , hence obtaining the first part of the answer:

*Prolonging  $\Omega^{\text{ABC}}$  corresponds, in the spectral domain, to approximation of the optimal ABC in the sense of truncation of its continued fraction expansion.*

Thanks to the deep results connecting continued fractions and approximation theory, namely Padé approximation (see [6]), we expand on this by concluding

*In the spectral domain, the ABC approximates the optimal one in the sense of Padé approximation about the right endpoint of the spectrum of  $L$ . Prolonging (shrinking)  $\Omega^{\text{ABC}}$  corresponds to increasing (decreasing) the order of the Padé approximant.*

This suggest that for  $i$  not too large the approximation quality is rather poor around the left endpoint of the spectrum of  $L$ , showing us some room for improvement. One such improvement corresponds to considering different boundary conditions where we can, i.e., along what we above called “the remainder of  $\partial\Omega^{\text{trunc}}$ ”, e.g., Robin boundary condition. Using the free parameters well, e.g., the Robin parameter, we can decrease the approximation error. Another, different, to improve on the above ABC is to change the Padé expansion point, hence introducing a new ABC/PML technique. Notice that in such case, we still obtain a different PML to these in [3, 4] as we do not change the discretization. Both of these improvements can be optimized so as to decrease the approximation error  $v|_{\tilde{\Omega}} \approx u|_{\tilde{\Omega}}$ . We demonstrate all of our results also numerically and then comment on possible generalizations. Some of these results have been published in [5].

## References

- [1] M. J. Gander, H. Zhang. A class of iterative solvers for the Helmholtz equation: factorizations, sweeping preconditioners, source transfer, single layer potentials, polarized traces, and optimized Schwarz methods. *SIAM Review*, volume (61): 3–76, 2019
- [2] M. M. Rana, V. E. Howle, K. Long, A. Meek, W. Milestone. A New Block Preconditioner for Implicit Runge-Kutta Methods for Parabolic PDE Problems. *SIAM Journal on Scientific Computing*, volume (43): S475–S495, 2021

- [3] D. Ingerman, V. Druskin, L. Knizhnerman. Optimal finite difference grids and rational approximations of the square root : I. Elliptic problems. *Communications on Pure and Applied Mathematics*, volume (53): 1039–1066, 2000
- [4] V. Druskin, S. Güttel, L. Knizhnerman. Near-optimal perfectly matched layers for indefinite Helmholtz problems. *SIAM Review*, volume (58): 90–116, 2016
- [6] L. Lorentzen, H. Waadeland. Continued Fractions with Applications. *Elsevier, North Holland*, in series *Studies in Computational Mathematics 3*, 1992
- [5] M. J. Gander, L. Jakabčin, M. Otrata. Domain truncation, absorbing boundary conditions, Schur Complements, and Padé approximation. *Electronic Transactions on Numerical Analysis*, volume (59): 319–341, 2024

# Error estimate and stopping criteria for least-squares problems solved by CG-like algorithms CGLS and LSQR

*Jan Papež, Petr Tichý*

## Abstract

In [2], we presented an adaptive estimate for the energy norm of the error in the conjugate gradient (CG) method. Using the notation from [2, Alg. 1],  $A$ -norm of the error between the exact solution of  $Ax = b$  and the CG approximation  $x_\ell$  given in the  $\ell$ th step is estimated as

$$\|x - x_\ell\|_A^2 \approx \Delta_{\ell:k}^{\text{CG}} := \sum_{j=\ell}^k \alpha_j \|r_j\|^2, \quad (1)$$

where  $\|v\|_A^2 \equiv v^T A v$  denotes the squared  $A$ -norm. Integrating the estimate into the existing CG code is straightforward and simple; see [4, Alg. 1]. At the current  $k$ th CG iteration, we get an estimate with the delay  $d = k - \ell$  for previous approximation  $x_\ell$ . The delay  $d$  is set adaptively by [4, Alg. 2]. From the construction,  $\Delta_{\ell:k}^{\text{CG}}$  yields a lower bound

$$\|x - x_\ell\|_A^2 \geq \Delta_{\ell:k}^{\text{CG}}.$$

In [4] and in the prospective talk at Householder Symposium XXII we consider algorithms for solving least-squares problems with a general, possibly rectangular matrix

$$\min_{x \in \mathbb{R}^m} \|b - Ax\|, \quad b \in \mathbb{R}^n, A \in \mathbb{R}^{n \times m}, n \geq m,$$

that are mathematically based on applying CG to a system with a positive (semi-)definite matrix  $A^T A$ . We discuss CGLS based on Hestenes–Stiefel-like implementation as well as LSQR based on Golub–Kahan bidiagonalization, and both unpreconditioned and preconditioned variants. We show that the adaptive estimate used in CG can be extended for these algorithms to estimate the monotonically decreasing quantity

$$\|x - x_\ell\|_{A^T A}^2 = \|r_\ell\|^2 - \|r\|^2, \quad (2)$$

where  $x = A^\dagger b$  is the minimal norm solution,  $x_\ell$  is the approximation in the  $\ell$ th step of CGLS or LSQR,  $r_\ell = b - Ax_\ell$ , and  $r = b - b_{|\mathcal{R}(A)}$  with  $b_{|\mathcal{R}(A)}$  being the orthogonal projection of  $b$  onto the range of  $A$ .

For example, the estimate

$$\|x - x_\ell\|_{A^T A}^2 \approx \Delta_{\ell:k}^{\text{LSQR}} := \sum_{j=k}^k \phi_{j+1}^2,$$

for estimating the quantity of interest (2) in LSQR algorithm is given, analogously to  $\Delta_{\ell:k}^{\text{CG}}$ , as a sum of scalar terms, which are available in the algorithm; here we use the notation from [4, Alg. 4]. Moreover,  $\Delta_{\ell:k}^{\text{LSQR}}$  provides a lower bound on  $\|x - x_\ell\|_{A^T A}^2$ .

We emphasize the applicability of the estimates (bounds) for the computations in finite-precision arithmetic. Their derivation is only based on local orthogonality, which is typically well preserved

during computations; see [5]. We demonstrate that the estimates remain computationally inexpensive to evaluate and are numerically reliable in finite-precision arithmetic under mild assumptions. These qualities make the estimates highly suitable for stopping the iterations.

One can consider the stopping criterion requiring that

$$\frac{\|r_\ell\|^2 - \|r\|^2}{\|r\|^2} \leq \varepsilon \quad (3)$$

for a prescribed tolerance  $\varepsilon$ . It is clear from (2), that after  $\|r_\ell\| \approx \|r\|$ , further iterations bring no significant decrease of the residual norm  $\|r_\ell\|$ . Using (2), the criterion (3) is equivalent to

$$\|x - x_\ell\|_{A^T A}^2 \leq \frac{\varepsilon}{1 - \varepsilon} \|r_\ell\|^2,$$

where the estimate for  $\|x - x_\ell\|_{A^T A}^2$  can be used.

Another stopping criterion based on a backward error can also be considered when applying our estimates. This criterion aims to identify the iteration at which the computed approximation can be interpreted as the least-squares solution to a perturbed system

$$\min_x \|(b + f) - (A + E)x\|,$$

with

$$\min_{f, E, \zeta} \{\zeta \text{ such that } \|E\| \leq \zeta \|A\|, \|f\| \leq \zeta \|b\|\} \leq \varepsilon.$$

This backward error for stopping LSQR iterations has been studied, e.g., in [3, 1], and can be approximated using the asymptotically tight bound

$$\frac{\|x - x_\ell\|_{A^T A}^2}{\|A\| \|x_\ell\| + \|b\|};$$

see [1].

Finally, we present a range of numerical experiments to confirm the robustness and very satisfactory behaviour of the estimates for CGLS, LSQR, and also their preconditioned variants. We hope that these estimate will prove to be useful in practical computations. They allow us to approximate, with the prescribed relative accuracy, the quantity of interest at a negligible cost.

## References

- [1] X.-W. Chang, C. C. Paige, and D. Titley-Peloquin. Stopping criteria for the iterative solution of linear least squares problems. *SIAM J. Matrix Anal. Appl.*, 31(2):831–852, 2009.
- [2] G. Meurant, J. Papež, and P. Tichý. Accurate error estimation in CG. *Numer. Algorithms*, 88(3):1337–1359, 2021.
- [3] C. C. Paige and M. A. Saunders. LSQR: an algorithm for sparse linear equations and sparse least squares. *ACM Trans. Math. Software*, 8(1):43–71, 1982.
- [4] J. Papež and P. Tichý. Estimating error norms in CG-like algorithms for least-squares and least-norm problems. *Numer. Algorithms*, 97(1):1–28, 2024.
- [5] Z. Strakoš and P. Tichý. On error estimation in the conjugate gradient method and why it works in finite precision computations. *Electron. Trans. Numer. Anal.*, 13:56–80, 2002.

# AdaCUR: Efficient Low-rank Approximation of Parameter-dependent matrices $A(t)$ via CUR decomposition

Taejun Park, Yuji Nakatsukasa

## Abstract

Let  $A(t) \in \mathbb{R}^{m \times n}$  be a parameter-dependent matrix and suppose we want to compute its low-rank approximation at a finite number of parameter values  $t_1, t_2, \dots, t_q$ . This problem arises in several applications including the compression of a series of images, dynamical systems, and Gaussian process regression, where low-rank approximations are needed for the sequence  $A(t_1), A(t_2), \dots, A(t_q)$ . While existing methods such as dynamical low-rank approximation [6] and random embedding techniques [7] offer solutions, they typically incur a complexity of  $\mathcal{O}(r_i T_{A(t_i)})$  for each parameter  $t_i$ , with  $r_i$  as the target rank and  $T_{A(t_i)}$  as the cost of a matrix-vector product with  $A(t)$ . We propose an alternative approach using the CUR decomposition, which can accelerate low-rank approximation to an average complexity of  $\mathcal{O}(T_{A(t_i)})$  while addressing key challenges, such as rank-adaptivity and error control, often missing in other methods.

The CUR decomposition [3, 5, 8, 11] approximates a matrix  $A$  using subsets of its rows and columns:

$$A \approx CU^\dagger R,$$

where  $C$  and  $R$  are subsets of  $A$ 's columns and rows, and  $U$  is their intersection. This decomposition, in contrast to methods like the truncated SVD, preserves properties such as sparsity and aids in data interpretation by identifying significant columns and rows. For  $A(t)$ , recomputing row and column indices for each  $t_i$  is inefficient, as indices derived for one parameter value may still provide useful information for nearby parameters. Building on this insight, we introduce an algorithm, AdaCUR [12], which maximizes the reuse of row and column indices across parameter values.

AdaCUR computes low-rank approximations of parameter-dependent matrices via CUR decomposition:

$$A(t) \approx C(t)U(t)^\dagger R(t),$$

where  $C(t)$  and  $R(t)$  are subsets of the columns and rows of  $A(t)$ , and  $U(t)$  is their intersection. Starting from an initial CUR decomposition, AdaCUR reuses row and column indices until the error exceeds a specified threshold, at which point the indices are recomputed. To achieve this efficiently and reliably, we rely on a variety of tools from randomized numerical linear algebra [9]. Specifically, we use pivoting on a random sketch [1, 2] to obtain a reliable set of row and column indices, randomized rank estimation [10] to adapt to rank changes across parameter values, and randomized norm estimation [4] to approximate the relative error, ensuring effective error control. The resulting algorithm is efficient, rank-adaptive, and incorporates error control.

Additionally, we present FastAdaCUR, a variation that prioritizes speed over precision. FastAdaCUR achieves linear complexity in  $m$  and  $n$  after an initial index computation phase. Although highly efficient and rank-adaptive, it lacks rigorous error control, as it emphasizes speed over accuracy by only examining a subset of rows and columns of the matrix.

## References

- [1] Y. Dong and P.-G. Martinsson, *Simpler is better: a comparative study of randomized pivoting algorithms for CUR and interpolative decompositions*, Adv. Comput. Math., 49 (2023).

- [2] J. A. Duersch and M. Gu, *Randomized projection for rank-revealing matrix factorizations and low-rank approximations*, SIAM Rev., 62 (2020), pp. 661–682.
- [3] S. Goreinov, E. Tyrtyshnikov, and N. Zamarashkin, *A theory of pseudoskeleton approximations*, Linear Algebra Appl., 261 (1997), pp. 1–21.
- [4] S. Gratton and D. Titley-Peloquin, *Improved bounds for small-sample estimation*, SIAM J. Matrix Anal. Appl., 39 (2018), pp. 922–931.
- [5] K. Hamm and L. Huang, *Perspectives on CUR decompositions*, Appl. Comput. Harmon. Anal., 48 (2020), pp. 1088–1099.
- [6] O. Koch and C. Lubich, *Dynamical low-rank approximation*, SIAM J. Matrix Anal. Appl., 29 (2007), pp. 434–454.
- [7] D. Kressner and H. Y. Lam, *Randomized low-rank approximation of parameter-dependent matrices*, Numer. Lin. Alg. Appl., (2024), p. e2576.
- [8] M. W. Mahoney and P. Drineas, *CUR matrix decompositions for improved data analysis*, Proc. Natl. Acad. Sci., 106 (2009), pp. 697–702.
- [9] P.-G. Martinsson and J. A. Tropp, *Randomized numerical linear algebra: Foundations and algorithms*, Acta Numer., 29 (2020), p. 403–572.
- [10] M. Meier and Y. Nakatsukasa, *Fast randomized numerical rank estimation for numerically low-rank matrices*, Linear Algebra Appl., 686 (2024), pp. 1–32.
- [11] T. Park and Y. Nakatsukasa, *Accuracy and stability of CUR decompositions with oversampling*, arXiv:2405.06375, (2024).
- [12] T. Park and Y. Nakatsukasa, *Low-rank approximation of parameter-dependent matrices via CUR decomposition*, arXiv:2408.05595, (2024).

# Efficient Dynamic Image Reconstruction with Motion Estimation

*Mirjeta Pasha, Toluwani Okuanola, Misha Kilmer, and Melina Freitag*

## Abstract

Large-scale dynamic inverse problems are typically ill-posed and suffer from complexity of the model constraints and large dimensionality of the parameters. A common approach to overcome ill-posedness is through regularization that aims to add constraints on the desired parameters in both space and temporal dimensions. In this work, we propose an efficient method that incorporates a model for the temporal dimension by estimating the motion of the objects alongside solving the regularized problem. In particular, we consider the optical flow model as part of the regularization that simultaneously estimates the motion and provides an approximation for the desired image sequence. To overcome high computational cost when processing massive scale problems, we combine our approach with a generalized Krylov subspace method that efficiently solves the problem on relatively small subspaces. Further, we explore subspace restarting and recycling to overcome limited memory constraints and preconditioning to accelerate convergence. The effectiveness of the prescribed approaches is illustrated through numerical experiments arising in dynamic computerized tomography and image deblurring applications.

# Fast Iterative Solvers for Optimization of Nonlocal PDEs

John W. Pearson

## Abstract

In this talk, we consider fast and effective numerical methods for optimization problems where partial differential equations (PDEs) act as constraints, so-called *PDE-constrained optimization*. Such problems have numerous applications across science and engineering, for instance in fluid flow control problems, chemical and biological processes, mathematical finance, and medical imaging, to name a few. To give an example of a problem structure, consider the following formulation:

$$\min_{y,u} \quad \frac{1}{2} \|y - \hat{y}\|_{Q_1(\Omega)}^2 + \frac{\beta}{2} \|u\|_{Q_2(\Omega)}^2 \quad \text{s.t. } \mathcal{D}y = u \text{ in } \Omega,$$

where  $y$  and  $u$  denote one or more *state variables* (PDE variables) and optimal *control variables* respectively,  $\hat{y}$  is a *desired state*,  $\beta > 0$  is a *regularization parameter*, and  $\mathcal{D}$  represents a differential operator equipped with boundary conditions. The problem is posed on a (generally space–time) domain  $\Omega$ , with  $Q_1$  and  $Q_2$  two (given) norms. It is possible to impose additional algebraic constraints on the states and/or controls.

The vast majority of work on the optimal control of PDEs has involved *local PDEs*, where the behaviour of the PDE at a point in  $\Omega$  can be described by problem features within a small neighbourhood of that point. In this talk we consider the emerging field of *nonlocal PDE-constrained optimization*, including problems with fractional derivatives, integro-differential equations, or (integral) kernel functions. On the numerical linear algebra level, this leads to dense linear systems, as opposed to the sparse matrices obtained from many discretizations of local PDEs. However, by exploiting the structures of the relevant matrices, we are nonetheless able to construct viable and robust schemes for problems of dimensions that would otherwise be out of reach.

Specifically, we derive preconditioned iterative methods to tackle huge-scale linear(ized) systems that result from nonlocal PDE-constrained optimization, and carefully utilize structures that arise in such systems to enhance the efficiency of solvers. For example,

- We build a spectral-in-time Newton–Krylov method for solving multiscale particle dynamics problems, closely related to mean-field games and mean-field control, to high accuracy [1] (see also [2]). At each Newton iteration for the nonlinear problem, we may apply column operations to the Jacobian matrix to ensure invertibility of the leading block, then construct structured, Kronecker product-based preconditioners for the resulting Schur complement. This leads to rapid GMRES convergence for large and dense systems.
- We devise a new technology for nonlocal image denoising problems [3], applying an unnormalized extended Gaussian ANOVA kernel within a bilevel optimization routine. Matrix–vector multiplications may be applied via a (matrix-free) Nonequispaced Fast Fourier Transform, and the Conjugate Gradient method accelerated using a novel change of basis approach coupled with a diagonal preconditioning strategy. As a result, rapid and effective denoising for very large problems may be achieved with modest storage requirements on a computer.
- We tackle optimization problems constrained by certain space–time fractional differential equations with additional state and/or control constraints [4]. This is accomplished by exploiting the multilevel Toeplitz structure of many of the matrix sub-blocks, deriving multilevel

circulant preconditioners based on this, and designing a recursive linear algebra which leads to very low storage requirements and operation costs.

This talk will outline some of the progress made in the above areas. In each case, as opposed to exploiting the property of sparsity as one would do for local problems, we utilize structure which the problem provides us with (Kronecker-product approximation, Gaussian kernel which allows a fast discrete transform, or multilevel Toeplitz) in a bespoke way. We will also provide an outlook of the subject area, discussing new applications of nonlocal PDEs and optimization problems, and outlining how the above methods could be adapted to resolve these challenges.

## References

- [1] M. Aduamoah, B. D. Goddard, J. W. Pearson, and J. C. Roden, Pseudospectral methods and iterative solvers for optimization problems from multiscale particle dynamics. *BIT Numerical Mathematics*, 62 (4): pp. 1703–1743, 2022.
- [2] S. Güttel and J. W. Pearson, A spectral-in-time Newton–Krylov method for nonlinear PDE-constrained optimization. *IMA Journal of Numerical Analysis*, 42 (2): pp. 1478–1499, 2022.
- [3] A. Miniguano-Trujillo, J. W. Pearson, and B. D. Goddard, Efficient nonlocal linear image denoising: Bilevel optimization with Nonequispaced Fast Fourier Transform and matrix-free preconditioning. *arXiv preprint arXiv:2407.06834*, 2024.
- [4] S. Pougkakiotis, J. W. Pearson, S. Leveque, and J. Gondzio, Fast solution methods for convex quadratic optimization of fractional differential equations. *SIAM Journal on Matrix Analysis and Applications*, 41 (3): pp. 1443–1476, 2020.

# Global and local growth behavior of GEPP and GECP

John Peca-Medlin

## Abstract

Gaussian elimination (GE) remains the most used approach to solve dense linear systems in modern applications. For instance, GE with partial pivoting (GEPP) is the default solver in MATLAB when using the backslash operator with a general input matrix. Additionally, GE is a staple of introductory linear algebra courses (although, based on anecdotal evidence, it may not be a favorite among all students).

GE iteratively uses elimination updates below the diagonal on an initial input matrix  $A \in \mathbb{R}^{n \times n}$  to transform  $A$  into an upper triangular linear system, which eventually builds the matrix factorization  $A = LU$ , where  $L$  is a unit lower triangular matrix and  $U$  is upper triangular. General nonsingular  $A$  will not have a  $LU$  factorization if any leading minors are singular, i.e., if  $\det(A_{ij})_{i,j=1}^k = 0$  for some  $k \leq n - 1$ . Moreover, numerical stability of GE when using floating-point arithmetic is sensitive to any elimination steps that involve division by numbers close to 0. Hence, combining GE with certain pivoting schemes is preferable even if not necessary. A pivoting strategy results instead in the factorization  $PAQ = LU$  where  $P$  and  $Q$  are permutation matrices (see [10, 13] for studies of random permutations generated using GEPP). I will focus on the two particular pivoting strategies GEPP and GE with complete pivoting (GECP), as well as the standard GE with no alternative pivoting strategy (GENP). I am interested in studying how GEPP and GECP behave on the same linear systems as well as studying large growth (see below) on particular subclasses of matrices, including orthogonal matrices. Moreover, as a means to better address the question of why large growth is rarely encountered, I further study matrices with a large difference in growth between using GEPP and GECP, and I explore how the smaller growth strategy dominates behavior in a small neighborhood of the initial matrix. This is a summary overview for my recently published results in [11].

Understanding the behavior of GE under floating-point arithmetic has been an ongoing focus of numerical analysis for over 60 years. Early work by Wilkinson in [17], which led to the start of modern error analysis, considered studying the relative errors of computed solutions  $\hat{\mathbf{x}}$  to  $A\mathbf{x} = \mathbf{b}$  through the bound

$$\frac{\|\mathbf{x} - \hat{\mathbf{x}}\|_\infty}{\|\mathbf{x}\|_\infty} \leq 4n^2 \epsilon_{\text{machine}} \kappa_\infty(A) \rho(A), \quad \text{where } \rho(A) = \frac{\max_{i,j,k} |A_{ij}^{(k)}|}{\max_{i,j} |A_{ij}|}$$

is the *growth factor*,  $\kappa_\infty(A) = \|A\|_\infty \|A^{-1}\|_\infty$  is the  $\infty$ -condition number, and  $A^{(k)}$  is the intermediate form of  $A$  after  $k - 1$  GE steps, with  $A^{(1)} = A$  and  $A^{(n)} = U$ . The growth factor returns the relative largest entry ever encountered running through the entire GE process. Using GECP,  $\rho(A)$  takes the simpler form  $\rho(A) = \max_k |U_{kk}| / |U_{11}|$  while in general the largest entry is not necessarily captured by  $U$ . In well-conditioned systems, error analysis of GE can then be reduced to analysis of the growth factor.

Understanding worst-case bounds on  $\rho$  (i.e., the *growth problem*) using different pivoting strategies has been a continued area of research since Wilkinson's initial analysis in the early 1960s. Wilkinson resolved the worst-case behavior of GEPP, showing the bound  $\rho^{\text{GEPP}}(A) \leq 2^{n-1}$  for  $A \in \mathbb{R}^{n \times n}$  is sharp [17, 18]. For GECP, however, Wilkinson in the same work only provided an upper bound of

$$\rho^{\text{GECP}}(A) \leq (n \cdot 2 \cdot 3^{1/2} \cdots n^{1/(n-1)})^{1/2} \leq 2 \cdot n^{0.25 \ln n + 0.5},$$

which is believed to be very pessimistic. For a long time a conjecture (of apocryphal origins; cf. [5]) ventured the bound  $\rho^{\text{GECP}} \leq n$  (for real matrices). This conjecture survived for almost three decades, until Gould [6] found a “counterexample” for  $n = 13$  by producing a matrix using floating-point arithmetic with GECP growth of 13.0205 in 1991. Edelman [4] confirmed a true counterexample in exact arithmetic one year later. No substantial progress on the GECP growth problem was made for another 30 years later until 2023. Edelman and Urschel [5] establish a linear lower bound on  $\rho^{\text{GECP}}$  at least  $1.0045n$  for  $n > 10$ , by upgrading Edelman’s computational technique to upgrade Gould’s floating-point “counterexample” into an exact arithmetic counterexample (by updating one entry by  $10^{-7}$ ) into a theorem establishing floating-point and exact arithmetic growth factors cannot be too far from one another (cf. Theorem 4.2). Moreover, later that same year, Edelman and Urschel along with now Bisain [3] provide the first substantial improvement to Wilkinson’s upper bound for  $\rho^{\text{GECP}}$ , where (using a modified Hadamard inequality) they provide a bound of  $n^{c \log n + 0.91}$  using  $c \approx 0.20781$ , which beats Wilkinson’s original exponential  $\log n$  coefficient of 0.25. The huge gulf between these lower and upper bounds leaves much on the table for future improvements.

More recent analysis of matrix algorithm efficiency and accuracy has shifted away from worst-case analysis to more modern approaches, such as smoothed analysis (cf. [15] and references therein), which studies behavior under random perturbations. A full smoothed analysis using Gaussian (additive) perturbations has been successfully implemented in the case of GENP growth factors by Sankar, Spielman, and Teng [14], but has remained out of reach for both the GEPP and GECP growth problem. The closest such result for GEPP was the recent average-case analysis work of Huang and Tikhomirov [8], which established high probability polynomial growth bounds using input matrices with independent and identically distributed (iid) standard normal entries, but their proof strategy cannot be upgraded to a smoothed analysis approach (i.e., they only establish bounds on Gaussian perturbations of the zero matrix). No such (non-empirical) result for GECP has come close to smoothed analysis nor a full average-case analysis (other than with very particular structured random matrices in [9]). So worst-case analysis remains relevant for these pivoting strategies.

Our focus will be on studying the growth behavior of using both GEPP and GECP on the same linear system, and on how each strategy can inform growth behavior about the other. In particular, I am interested in local growth behavior around a system with large differences in growth behavior between both strategies. For instance, I am interested in the question of whether small perturbations on an initial system with a large discrepancy in growth behavior between both strategies concentrates toward the smaller growth for both strategies? For example, if  $A$  has large GEPP growth and small GECP growth, and if  $G$  is an iid Gaussian matrix, I am interested in how often  $\rho^{\text{GEPP}}(A + \varepsilon G) \approx \rho^{\text{GECP}}(A)$  for sufficiently small  $\varepsilon$ . To move toward addressing this question, I will establish bounds on how far apart differences in growth between both strategies can be when used on the same input matrix.

I will further focus on large growth for particularly structured matrices, including orthogonal matrices. This will include a refinement to the largest possible GEPP growth on orthogonal matrices, while also establishing a rich set of matrices for further study with large growth difference behavior. Growth for structured systems has proven fruitful in recent studies. For instance, growth using GENP, GEPP and GE with rook pivoting (GERP) for matrices formed using the Kronecker products of rotation matrices (i.e.,  $\bigotimes^n \text{SO}(2)$ ) is now completely understood [12], with even a full picture understood using GECP on a further subclass of these Kronecker product matrices [9]. Although GE should not be a first choice for solving orthogonal linear systems (viz.,  $Q\mathbf{x} = \mathbf{b}$  has

the solution  $\mathbf{x} = Q^T \mathbf{b}$  when  $Q$  is orthogonal), there are situations when applying GE to orthogonal matrices makes sense. For example, Barlow needed to understand the effect of GEPP on orthogonal matrices to carry out error analysis of bidiagonal reduction [1]; this led to Barlow and Zha's analysis in [2] using GEPP on orthogonal matrices, which they showed maximized a different  $L^2$ -growth factor. Additionally, while original studies of random growth factors tended to focus on ensembles with iid entries (cf. [16]), many authors noted and explored the potential that orthogonal matrices can produce large growth [7, 16]. Hence, orthogonal matrices remain a rich source of study for potential large growth factors.

## References

- [1] J. L. Barlow. More accurate bidiagonal reduction for computing the singular value decomposition. *SIAM Journal of Matrix Analysis and Applications*, 23:761–798, 2002.
- [2] J. L. Barlow and H. Zha. Growth in Gaussian elimination, orthogonal matrices, and the 2-norm. *SIAM Journal of Matrix Analysis and Applications*, 19:807–815, 1998.
- [3] A. Bisain, A. Edelman, and J. Urschel. A new upper bound for the growth factor in Gaussian elimination with complete pivoting. *arXiv preprint arXiv:2312.00994*, 2023.
- [4] A. Edelman. The complete pivoting conjecture for Gaussian elimination is false. *The Mathematica Journal*, 2:58–61, 1992.
- [5] A. Edelman and J. Urschel. Some new results on the maximum growth factor in Gaussian elimination. *SIAM Journal of Matrix Analysis and Applications*, 45(2):967–991, 2024.
- [6] N. Gould. On growth in Gaussian elimination with complete pivoting. *SIAM Journal of Matrix Analysis and Applications*, 12:354–361, 1991.
- [7] N. J. Higham and D. Higham. Large growth factors in Gaussian elimination with pivoting. *SIAM Journal of Matrix Analysis and Applications*, 10:155–164, 1989.
- [8] H. Huang and K. Tikhomirov. Average-case analysis of the Gaussian elimination with partial pivoting. *Probability Theory and Related Fields*, 189:501–567, 2024.
- [9] J. Peca-Medlin. Complete pivoting growth of butterfly matrices and butterfly Hadamard matrices. *arXiv preprint arXiv:2410.06477*, 2024.
- [10] J. Peca-Medlin. Distribution of the number of pivots needed using Gaussian elimination with partial pivoting on random matrices. *Annals of Applied Probability*, 24(2):2294–2325, 2024.
- [11] J. Peca-Medlin. Growth factors of orthogonal matrices and local behavior of Gaussian elimination with partial and complete pivoting. *SIAM Journal on Matrix Analysis and Applications*, 45(3):1599–1620, 2024.
- [12] J. Peca-Medlin and T. Trogdon. Growth factors of random butterfly matrices and the stability of avoiding pivoting. *SIAM Journal of Matrix Analysis and Applications*, 44(3):945–970, 2023.
- [13] J. Peca-Medlin and C. Zhong. On the longest increasing subsequence and number of cycles of butterfly permutations. *arXiv preprint arXiv:2410.20952*, 2024.

- [14] A. Sankar, D. Spielman, and S.-H. Teng. Smoothed analysis of the condition numbers and growth factors of matrices. *SIAM Journal of Matrix Analysis and Applications*, 11:335–360, 1990.
- [15] D. Spielman and S.-H. Teng. *Smoothed analysis: an attempt to explain the behavior of algorithms in practice*, Volume 52 (10). ACM, New York, 2009.
- [16] L. Trefethen and R. Schreiber. Average case stability of Gaussian elimination. *SIAM Journal of Matrix Analysis and Applications*, 11:335–360, 1990.
- [17] J. Wilkinson. Error analysis of direct methods of matrix inversion. *J. Assoc. Comput. Mach.*, 8:281–330, 1961.
- [18] J. Wilkinson. *The Algebraic Eigenvalue Problem*. Oxford University Press, London, UK, 1965.

# Using a Blocked Adaptive Randomized Range Finder to Reduce Memory Requirements in Deep Learning Based on the Householder QR Decomposition

Carolin Penke

## Abstract

Deep neural networks, such as GPT-like transformer architectures, are increasingly prevalent and consume significant portions of global computing infrastructure, predominantly using GPUs. These models demand vast datasets and are constrained by available compute capabilities during both the pre-training stage on supercomputers and the fine-tuning stage on smaller workstations. Enhancing training efficiency is therefore highly impactful. This work introduces techniques to leverage low-rank structures for reducing memory requirements and outlines a method to efficiently acquire the necessary subspaces by using a randomized range finder. We propose a GPU-accelerated algorithm, based on the Householder QR decomposition that is also applicable beyond deep learning contexts.

In the following, we briefly give background about the *randomized rangefinder*[3, 5, 8] and about low-rank methods in deep learning [4, 9]. Here, the randomized rangefinder can be used as a tool to efficiently compute necessary subspace bases. We present a GPU-accelerated variant, based on a Householder QR decomposition instead of the common Gram-Schmidt-based approach.

Given a matrix  $A \in \mathbb{R}^{m \times n}$  and a rank  $r \in \mathbb{N}$ , the most simple version of the randomized range finder [3] finds an orthogonal subspace basis  $Q \in \mathbb{R}^{m \times r}$  such that  $\text{range}(Q) \approx \text{range}(A)$  as

1.  $\Omega \leftarrow \text{randn}(n, r)$  (fill  $\Omega$  with random values)
2.  $Y \leftarrow A\Omega$  (matrix multiply)
3.  $Q \leftarrow \text{orth}(Y)$  (e.g. QR decomposition of  $Y$ ).

The notion  $\text{range}(Q) \approx \text{range}(A)$  holds in a probabilistic sense. With  $B := Q^T A$ , the method yields a decomposition

$$A = QB. \quad (1)$$

The minimal rank  $r$  to reach a certain error tolerance  $\epsilon > 0$ , such that

$$\|A - QQ^T A\| \leq \epsilon, \quad (2)$$

can not be known in advance. Instead, an *adaptive* randomized rangefinder can be employed to iteratively construct a subspace basis until the desired accuracy is reached.

In the training of a deep neural network, each layer is represented by matrices, including weights, gradients, and optimizer states. The weights are updated using gradients computed by back propagation, typically along with optimizer states, e.g. in the popular Adam optimizer. These states encode moving averages of the gradient's first and second moments, incorporating past iteration data to guide updates more effectively. However, storing optimizer states requires considerable memory. Frameworks like LoRA [4] and GaLore [9] reduce memory demands by exploiting the low-rank structure of gradients.

The popular LoRA framework utilizes a low-rank network architecture to efficiently accumulate weight updates derived from gradients and optimizer states. The GaLore framework follows another approach and dynamically computes a dominant subspace basis of low rank for the gradient

matrix during training. The optimizer states are represented within this subspace to reduce storage requirements. Rather than relying on a computationally intensive singular value decomposition (SVD), the use of a randomized range finder offers a more practical and efficient alternative.

In GaLore, as in LoRA, the rank  $r$  is treated as a hyperparameter, typically chosen based on intuition or experience (e.g.,  $r = 128$ ). An alternative approach is to use the approximation quality  $\epsilon$  as a more interpretable hyperparameter, alongside an adaptive variant of the randomized range finder.

With this adaptive method, the dimensionality of subspaces across consecutive training steps can vary. This enunciates the problem, that adding optimizer states, represented in different subspaces, is not very meaningful and can lead to deteriorated performance, even when the rank is fixed. A linear transformation can be applied to ensure subspace consistency. A low-rank optimizer state  $M_t \in \mathbb{R}^{m \times r_t}$ , should at step  $t + 1$  be substituted by  $Q_{t+1} Q_t M_t$ .

With the goal of exploiting the memory hierarchy in modern hardware, a blocked variant of the Adaptive Randomized Range Finder is presented in [5]. In each iteration, a random matrix  $\Omega \in \mathbb{R}^{n \times b}$  is generated to sample the columns of  $A$  via a matrix multiplication  $A\Omega$ . The result is orthogonalized with respect to previously generated basis vectors using a Gram-Schmidt procedure.

Here, a non-probabilistic stopping criterion is devised by keeping track of the residual  $A - QB$ . The array  $A$  is updated to reflect this and approaches zero. A downside of this criterion is the higher memory requirements as three arrays ( $Q$ ,  $B$ ,  $A$ ) need to be maintained. This is relevant in the context of gradient approximation in deep learning, because, here, available GPU memory is a significant bottleneck.

Another stopping criterion is devised in [8], which does not necessitate maintaining the residual matrix, but is derived from the newly computed panels of  $B$  instead. Furthermore, the authors devise an algorithm that avoids passing over  $A$  during the loop and move the generation of random matrices outside the loop. In [2], the randomized range finder is applied as a crucial step in compressing matrices to Hierarchically Semi-Separable structure. A new probabilistic stopping allows for a relative error bound.

The other works [3, 5, 8, 2] present algorithms based on the adaptive, blocked, Gram-Schmidt-orthogonalization of  $A\Omega$ , where  $\Omega = [\Omega_0 \dots \Omega_k]$  contains the random panels constructed in the context of the iteration. In this work, we want to explore the alternative approach of computing the Householder- $QR$  decomposition of  $A\Omega$  adaptively, i.e. generating sampled panels  $A\Omega$  on the fly, and only computing as many Householder vectors as necessary to approximate the subspace to a given tolerance.

Our motivation to use Householder over Gram-Schmidt is foremost a practical one. The subspace computations introduce a significant computational load into the training process, that otherwise utilizes GPUs very efficiently. The gradients already reside inside GPU memory, so it makes sense to use a GPU-accelerated algorithm. GPU-accelerated implementations of the blocked Householder QR decomposition (LAPACK routine `*geqrt3`) are available [1, 7] and can be adapted to perform the algorithm outlined in the following.

We divide  $A$  into blocks, store Householder vectors in  $V$ , and successively compute block rows of  $B$ , which can be stored in the memory location of  $A$ . Each storage-efficient factorization [6] of a block yields an upper triangular matrix block, all of which are stored in  $T$ .

As a notation for referring to blocks, block rows and block columns we use

$$A = \begin{bmatrix} A_{0,0} & \cdots & A_{0,k} \\ \vdots & \ddots & \vdots \\ A_{j,0} & \cdots & A_{j,k} \end{bmatrix}, \quad V = \begin{bmatrix} V_{0,0} & \cdots & V_{0,k} \\ \vdots & \ddots & \vdots \\ V_{j,0} & \cdots & V_{j,k} \end{bmatrix}, \quad B = \begin{bmatrix} - & B_0 & - \\ - & \vdots & - \\ - & B_k & - \end{bmatrix}, \quad T = [T_0 \quad \cdots \quad T_k].$$

The orthogonal subspace basis in (1) is represented as  $Q = \prod_{i=0}^k (I - V_i T_i V_i^T)$ . We use colon notation to refer to a submatrix of  $M$  as  $M_{i:l,p:q}$ , or to part of a block-column as  $M_{i:j,p}$ .

Algorithm 1 successively creates the block columns of  $V$  and block rows of  $B$ .  $A$  can be overwritten by  $B$  due to the error criterion from [8], which only relies on the Frobenius norm of the current panel of  $B$ . For simpler notation we assume the matrix dimensions to be divisible by  $b$ .

---

**Algorithm 1** Householder Block Adaptive Randomized Range Finder

---

**Require:** A matrix  $A \in \mathbb{R}^{m \times n}$ , a tolerance  $\epsilon$ , and a block size  $b$ .

```

1:  $E \leftarrow \|A\|_F$ 
2:  $B \leftarrow A$ 
3:  $i \leftarrow 0$ 
4: while  $E > \epsilon$  do
5:   Fill  $\Omega \in \mathbb{R}^{n \times b}$  with values from a standard Gaussian distribution.
6:    $(V_{i:j,i}, T_i) \leftarrow \text{qr}(B_{i:j,0:k} \Omega)$  ▷ Storage-efficient QR decomposition, geqrt
7:    $B_{i:k} \leftarrow (I - V_i T_i V_i^T) B_{i:k}$ 
8:    $E \leftarrow E - \|B_i\|_F$ 
9:    $i \leftarrow i + 1$ 
10: end while
11:  $V \leftarrow V_{:,0:i-1}$ 
12:  $B \leftarrow B_{0:i-1,:}$ 
13:  $r \leftarrow (i - 1) \cdot b$ 

```

**Ensure:** Rank  $r$ , Householder vectors  $V \in \mathbb{R}^{m \times r}$ ,  $B \in \mathbb{R}^{r \times n}$ ,  $T_0, \dots, T_{i-1} \in \mathbb{R}^{b \times b}$  such that  $\|A - QB\|_{\text{Fro}} \leq \epsilon$ , where  $Q = \prod_{l=0}^{i-1} (I - V_l T_l V_l^T)$ .

---

When the sampled panel  $B_{i:j,0:k} \Omega$  is updated independently of the matrix update in the previous iteration (line 7 in Algorithm 1), this update together with the panel factorization on the CPU, can be overlapped with the matrix update of  $B$ . This introduces extra computations but shortens the critical path, when the block size is chosen in a way to completely hide the panel update and factorization.

Apart from allowing a GPU-accelerated implementation, the presented Householder QR approach has the advantage of improved stability over the Gram-Schmidt approach, not needing a reorthogonalization step. As we are dealing with rapidly decaying singular matrices in the gradient matrix, stability becomes a relevant practical consideration.

Methods from randomized numerical linear algebra have promise to become a viable tool in the context of resource-efficient low-rank deep learning.

## References

- [1] E. ELMROTH AND F. G. GUSTAVSON, *Applying recursion to serial and parallel QR factorization leads to better performance*, 44, pp. 605–624.

- [2] C. GORMAN, G. CHÁVEZ, P. GHYSELS, T. MARY, F.-H. ROUET, AND X. S. LI, *Robust and Accurate Stopping Criteria for Adaptive Randomized Sampling in Matrix-Free Hierarchically Semiseparable Construction*, 41, pp. S61–S85.
- [3] N. HALKO, P. G. MARTINSSON, AND J. A. TROPP, *Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions*, SIAM Review, 53 (2011), pp. 217–288.
- [4] E. J. HU, Y. SHEN, P. WALLIS, Z. ALLEN-ZHU, Y. LI, S. WANG, L. WANG, AND W. CHEN, *LoRA: Low-rank adaptation of large language models*, in International Conference on Learning Representations, 2022.
- [5] P.-G. MARTINSSON AND S. VORONIN, *A randomized blocked algorithm for efficiently computing rank-revealing factorizations of matrices*, 38, pp. S485–S507.
- [6] R. S. SCHREIBER AND C. VAN LOAN, *A storage efficient  $wy$  representation for products of Householder transformations*.
- [7] S. TOMOV, J. DONGARRA, AND M. BABOULIN, *Towards dense linear algebra for hybrid GPU accelerated manycore systems*, Parallel Computing, 36 (2010), pp. 232–240.
- [8] W. YU, Y. GU, AND Y. LI, *Efficient randomized algorithms for the fixed-precision low-rank matrix approximation*, SIAM Journal on Matrix Analysis and Applications, 39 (2018), pp. 1339–1359.
- [9] J. ZHAO, Z. ZHANG, B. CHEN, Z. WANG, A. ANANDKUMAR, AND Y. TIAN, *Galore: Memory-efficient LLM training by gradient low-rank projection*, in 5th Workshop on practical ML for limited/low resource settings, 2024.

# A hybrid method for computing a few singular triplets of very large sparse matrices

*James Baglama, Jeniffer Picucci, Vasilije Perović*

## Abstract

Ability to efficiently compute a (partial) Singular Value Decomposition (SVD) of a matrix is essential in a wide range of problems, including data mining, genomics, machine learning, PCA, and signal processing. In such applications matrices tend to be very large and sparse and one is typically interested in computing only a few of the largest singular triplets. Over the last several decades this problem has led to a considerable amount of research and software development, see e.g., [4, 6, 7, 9, 10, 11, 15] and the references therein.

In this talk, for a large and sparse  $A \in \mathbb{R}^{\ell \times n}$ , we present a new hybrid restarted Lanczos bidiagonalization method for the computation of a small number,  $k$ , of its extreme singular triplets, i.e., we compute  $\{\sigma_j, u_j, v_j\}_{j=1}^k$  such that

$$Av_j = \sigma_j u_j, \quad A^T u_j = \sigma_j v_j, \quad j = 1, 2, \dots, k.$$

At the core of our proposed algorithm [3], as in many of the above listed ones, is the Golub-Kahan-Lanczos (GKL) bidiagonalization procedure which at step  $m$  results in the  $m$ -GKL factorization

$$AP_m = Q_m B_m, \tag{1}$$

$$A^T Q_m = P_m B_m^T + f e_m^T = [P_m \ p_{m+1}] \begin{bmatrix} B_m^T \\ \beta_m e_m^T \end{bmatrix}, \tag{2}$$

where  $P_m = [p_1, \dots, p_m] \in \mathbb{R}^{n \times m}$  and  $Q_m = [q_1, \dots, q_m] \in \mathbb{R}^{\ell \times m}$  have orthonormal columns, the residual vector  $f \in \mathbb{R}^n$  satisfies  $P_m^T f = 0$ ,  $\beta_m = \|f\|$ , and  $p_{m+1} = f/\beta_m$ , and  $B_m$  is an  $m \times m$  upper bidiagonal matrix.

Approximations of the singular triplets of  $A$  can then be obtained from the singular triplets of  $B_m$ , and in the case when the norm of the residual vector  $f$  is small, the singular values of  $B_m$  are close to the singular values of  $A$ . But for modest values of  $m$  these approximations are typically poor (assuming limited memory makes increasing  $m$  not an option) and thus leaving one with an option to modify, explicitly or implicitly, the starting vector  $p_1$  and *restart the GKL process*. In [4] the authors exploited the mathematical equivalence for symmetric eigenvalue computations of the implicitly restarted Arnoldi (Lanczos) method of Sorensen [13] and the thick-restarting scheme of Wu and Simon [14], and applied it to a restarted GKL procedure. The resulting thick-restarted GKL routine, `irlba`, turns out to be a simple and computationally fast method for computing a few of the extreme singular triplets that is less sensitive to propagated round-off errors.

However, the `irlba` routine [4] often struggles when the dimension,  $m$ , of the Krylov subspaces is memory limited and kept relatively small in relationship to the number of desired singular triplets  $k$ . Very recently, in the context of symmetric eigenvalue computation, we were able to overcome this memory restriction by creating a hybrid restarted Lanczos method that combines thick-restarting with Ritz vectors with a new technique, *iteratively refined Ritz vectors* [1]. We recall that in [8] Jia proposed to use *refined Ritz* vectors in place of Ritz vectors as eigenvector approximations of a square matrix  $M$  [8]. More specifically, for a given approximate eigenvalue  $\mu_j$  of  $M$ , Jia's method looks to minimize  $\|Mz_j - \mu_j z_j\|$  for a unit vector  $z_j$  from a given subspace  $\mathcal{W}$ . Moreover, in [8] it was

shown that on the subspace  $\mathcal{W}$  an approximate eigenpair using the refined Ritz vector produced a “smaller” residual norm than an eigenpair approximation with the Ritz pair. More recently, in [1] we were able to extend this idea to *iterative refined Ritz* values/vectors for the symmetric eigenvalue problem that produces an even smaller residual norm than refined Ritz – this resulted in a better converging method that outperformed using Ritz or refined Ritz vectors. But this comes at the price as working with iteratively refined Ritz vectors is more challenging, in comparison to just Ritz vectors, due to the fact that the scheme of thick-restarting of Wu and Simon is not available [1]. However, not all is lost and in [1] we introduce an alternate scheme in which, based on the relationships first proposed by Sorensen [13] and later outlined in detail by Morgan [12], the iteratively refined Ritz vectors are linearly combined and then used to restart the process. We choose the constants in a way that the linear combination of the iteratively refined Ritz vectors resembles a restart, in a somewhat asymptotic sense, of thick-restarting, see [1] for details.

In our most recent work [3], we applied the earlier results from [1] by making a natural connection between the symmetric eigenvalue problem and the SVD of  $A \in \mathbb{R}^{\ell \times n}$  and considered both the normal matrix  $A^T A \in \mathbb{R}^{n \times n}$  and the augmented matrix  $C = \begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix} \in \mathbb{R}^{(\ell+n) \times (\ell+n)}$ . Multiplying (1) from the left by  $A^T$  produces the Lanczos tridiagonal decomposition of  $A^T A$ , namely

$$A^T A P_m = P_m B_m^T B_m + \alpha_m f_m e_m^T = \begin{bmatrix} P_m & p_{m+1} \end{bmatrix} \begin{bmatrix} B_m^T B_m \\ \alpha_m \beta_m e_m^T \end{bmatrix}. \quad (3)$$

Similarly, in the case of matrix  $C$ , after performing  $2m$  steps of the standard Lanczos algorithm with the starting vector  $[0; p_1] \in \mathbb{R}^{\ell+n}$  we have a  $2m \times 2m$  tridiagonal projection matrix, which when followed by an odd-even permutation gives the following Lanczos factorization [10]

$$\begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix} \begin{bmatrix} Q_m & 0 \\ 0 & P_m \end{bmatrix} = \begin{bmatrix} Q_m & 0 & 0 \\ 0 & P_m & p_{m+1} \end{bmatrix} \begin{bmatrix} 0 & B_m \\ B_m^T & 0 \\ \beta_m e_m^T & 0 \end{bmatrix}. \quad (4)$$

With the two Lanczos factorization relationships (3) and (4), the theoretical results and properties related to the hybrid iterative refined Ritz (eigenvalue) scheme in [1] are carried over resulting into two hybrid routines capable of computing few extreme singular triplets  $A$  based on either  $A^T A$  or  $C = \begin{bmatrix} 0 & A \\ A^T & 0 \end{bmatrix}$ . While this extension seems straightforward its implementation on (4), as well as the new development of iterative refined Ritz working on the normal system, is nontrivial.

Through numerical examples, we have observed in [1, 3] that when memory was limited and only iterative refined Ritz vectors were used to restart the method there was potential for either slow or no convergence. In order to overcome this challenge, we developed a hybrid method that, based on extensive numerical tests and existing heuristics, switches between thick-restarted with Ritz vectors and under certain criteria it restarts with a linear combination of iterative refined Ritz vectors. We note that a careful balance is needed here, since on the one side the iterative refined Ritz vectors can give a better approximation but with possible stagnation, while on the other side thick-restarted is a more efficient restarting scheme, but with not as good of approximations.

In the rest of the talk we discuss multiple criteria we used to determine when to make a switch and provide some justification along with several useful heuristics. We also describe a simple yet powerful variant of our proposed hybrid algorithm ( $\approx 100$  lines of MATLAB code) where the Krylov basis size is  $m = 2$  and which requires a nontrivial purging of converged vectors. This simplified code, with potential for extensions [2], does not require calls to any LAPACK routines, thus making

it portable and at the same time very competitive on a number of large matrices. For instance, in [3] we computed the largest singular triplet of a 214 million  $\times$  214 million matrix (1.2GB) from the `kmerV1r` dataset on a laptop in just 31 minutes, whereas the `irlba` method [4] took 75 minutes, and MATLAB’s internal `svds` function required 4 hours. We conclude with several additional examples that illustrate the proposed hybrid scheme is competitive with other publicly available code when there are limited memory requirements.

## References

- [1] Baglama, J., Bella, T., Picucci, J.: Hybrid iterative refined method for computing a few extreme eigenpairs of a symmetric matrix. *SIAM J. Sci. Comput.* **43**(5), S200–S224 (2021)
- [2] Baglama, J., Perović, V.: Explicit deflation in Golub-Kahan-Lanczos bidiagonalization methods. *ETNA*, **58** (2023)
- [3] Baglama, J., Perović, V., Picucci, J.: Hybrid iterative refined restarted Lanczos bidiagonalization methods. *Numer. Algorithms*, **92**(2) (2023)
- [4] Baglama, J., Reichel, L.: Augmented implicitly restarted Lanczos bidiagonalization methods. *SIAM J. Sci. Comput.* **27**(1), 19–42 (2005)
- [5] Baglama, J., Reichel, L.: An implicitly restarted block Lanczos bidiagonalization method using Leja shifts. *BIT Numer. Math.* **53**(2), 285–310 (2013)
- [6] Goldenberg, S., Stathopoulos, A., Romero, E.: A Golub–Kahan–Davidson method for accurately computing a few singular triplets of large sparse matrices. *SIAM J. Sci. Comput.* **41**(4), A2172–A2192 (2019)
- [7] Hochstenbach, M.E.: Harmonic and refined extraction methods for the singular value problem, with applications in least squares problems. *BIT Numer. Math.* **44**(4), 721–754 (2004)
- [8] Jia, Z.: Refined iterative algorithms based on Arnoldi’s process for large unsymmetric eigenproblems. *Linear Algebra Appl.* **259**, 1–23 (1997)
- [9] Jia, Z., Niu, D.: An implicitly restarted refined bidiagonalization Lanczos method for computing a partial SVD. *SIAM J. Matrix Anal. Appl.* **25**(1), 246–265 (2003)
- [10] Kokiopoulou, E., Bekas, C., Gallopoulos, E.: Computing smallest singular triplets with implicitly restarted Lanczos bidiagonalization. *Applied Numer. Math.* **49**, 39–61 (2004).
- [11] Lehoucq, R.B., Sorensen, D.C., Yang, C.: ARPACK users’ guide: Solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods. SIAM (1998)
- [12] Morgan, R.B.: On restarting the Arnoldi method for large nonsymmetric eigenvalue problems. *Math. Comp.* **65**(215), 1213–1230 (1996)
- [13] Sorensen, D.C.: Implicit application of polynomial filters in a  $k$ -step Arnoldi method. *SIAM J. Matrix Anal. Appl.* **13**(1), 357–385 (1992)
- [14] Wu, K., Simon, H.: Thick-restart Lanczos method for large symmetric eigenvalue problems. *SIAM J. Matrix Anal. Appl.* **22**(2), 602–616 (2000)
- [15] Wu, L., Romero, E., Stathopoulos, A.: Primme\_svds: A high-performance preconditioned svd solver for accurate large-scale computations. *SIAM J. Sci. Comput.* **39**(5), S248–S271 (2017)

# Randomized Nyström approximation of non-negative self-adjoint operators

*David Persson, Nicolas Boullé, Daniel Kressner*

## Abstract

A ubiquitous task in numerical linear algebra is to compute a low-rank approximation to a matrix  $\mathbf{A}$ . Randomized techniques [8, 9, 10, 12] are becoming increasingly popular for computing cheap, yet accurate, low-rank approximations to matrices. Most notably, the *randomized singular value decomposition* (SVD) [9] has evolved into one of the primary choices, due to its simplicity, performance, and reliability. In its most basic form, the randomized SVD performs the approximation  $\mathbf{Q}\mathbf{Q}^*\mathbf{A} \approx \mathbf{A}$ , where  $\mathbf{Q}$  is an orthonormal basis for the range of  $\mathbf{A}\Omega$ , with  $\Omega$  being a tall and skinny random sketch matrix. In many applications of low-rank approximation, such as  $k$ -means clustering [13], PCA [14], and Gaussian process regression [7], it is known that  $\mathbf{A}$  is symmetric positive semi-definite. In this case, one usually prefers the so-called *randomized Nyström approximation* [8]

$$\hat{\mathbf{A}} := \mathbf{A}\Omega(\Omega^*\mathbf{A}\Omega)^\dagger\Omega^*\mathbf{A} \approx \mathbf{A}, \quad (1)$$

where  $\Omega$  is, again, a random sketch matrix. This approximation has received significant attention in the literature [8, 11, 12] and, like the randomized SVD, it enjoys strong theoretical guarantees. With the same number of matrix-vector products, the randomized Nyström approximation is typically significantly more accurate than the randomized SVD when the matrix has rapidly decaying singular values. Additionally, the Nyström method requires only a single pass over the matrix, compared to two passes for the randomized SVD, enabling all matrix-vector products to be performed in parallel.

Recently, Boullé and Townsend [4, 5] generalized the randomized SVD from matrices to Hilbert-Schmidt operators. Subsequent works [3, 6] employed this infinite-dimensional generalization of the randomized SVD to learn Green's functions associated with an elliptic or parabolic partial differential equations (PDE) from a few solutions of the PDE. This approach uses hierarchical low-rank techniques and exploits the fact that Green's functions are smooth away from the diagonal and therefore admit accurate off-diagonal low-rank approximations [1, 2]. Other applications, like Gaussian process regression and Support Vector Machines, involve integral operators that feature positive and *globally* smooth kernels. In turn, the operator is not only self-adjoint and positive but it also allows for directly applying low-rank approximation, without the need to resort to hierarchical techniques. Given existing results on matrices, it would be sensible to use an infinite-dimensional extension of the randomized Nyström approximation in such situations.

In this work, we present and analyze an infinite-dimensional extension of the randomized Nyström approximation for computing low-rank approximations to self-adjoint, positive, trace class operators. A significant advantage of the proposed framework is that once a low-rank approximation of the operator is computed, one can use this approximation to compute a low-rank approximation to *any* discretization of the operator.

## References

- [1] Mario Bebendorf. *Hierarchical matrices*. Springer, 2008.
- [2] Mario Bebendorf and Wolfgang Hackbusch. Existence of  $\mathcal{H}$ -matrix approximants to the inverse FE-matrix of elliptic operators with  $L^\infty$ -coefficients. *Numer. Math.*, 95:1–28, 2003.

- [3] Nicolas Boullé, Seick Kim, Tianyi Shi, and Alex Townsend. Learning Green’s functions associated with time-dependent partial differential equations. *J. Mach. Learn. Res.*, 23(1):9797–9830, 2022.
- [4] Nicolas Boullé and Alex Townsend. A generalization of the randomized singular value decomposition. In *International Conference on Learning Representations*, 2022.
- [5] Nicolas Boullé and Alex Townsend. Learning elliptic partial differential equations with randomized linear algebra. *Found. Comput. Math.*, 23(2):709–739, 2023.
- [6] Nicolas Boullé, Diana Halikias, and Alex Townsend. Elliptic PDE learning is provably data-efficient. *Proc. Natl. Acad. Sci. U.S.A.*, 120(39):e2303904120, 2023.
- [7] Jacob Gardner, Geoff Pleiss, Kilian Q Weinberger, David Bindel, and Andrew G Wilson. GPyTorch: Blackbox matrix-matrix Gaussian process inference with GPU acceleration. In *Advances in Neural Information Processing Systems*, volume 31, 2018.
- [8] Alex Gittens and Michael W. Mahoney. Revisiting the Nyström method for improved large-scale machine learning. *J. Mach. Learn. Res.*, 17:Paper No. 117, 65, 2016.
- [9] N. Halko, P. G. Martinsson, and J. A. Tropp. Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions. *SIAM Rev.*, 53(2):217–288, 2011.
- [10] Yuji Nakatsukasa. Fast and stable randomized low-rank matrix approximation. *arXiv preprint arXiv:2009.11392*, 2020.
- [11] David Persson and Daniel Kressner. Randomized low-rank approximation of monotone matrix functions. *SIAM J. Matrix Anal. Appl.*, 44(2):894–918, 2023.
- [12] Joel A Tropp, Alp Yurtsever, Madeleine Udell, and Volkan Cevher. Fixed-rank approximation of a positive-semidefinite matrix from streaming data. In *Advances in Neural Information Processing Systems*, volume 30, 2017.
- [13] Shusen Wang, Alex Gittens, and Michael W Mahoney. Scalable kernel K-means clustering with Nyström approximation: relative-error bounds. *J. Mach. Learn. Res.*, 20(1):431–479, 2019.
- [14] Kai Zhang, Ivor W. Tsang, and James T. Kwok. Improved Nyström low-rank approximation and error analysis. In *Proc. International Conference on Machine Learning*, pages 1232–1239, 2008.

# A Randomized Numerical Method for Joint Eigenvalues of Commuting Matrices

*Haoze He, Daniel Kressner, Bor Plestenjak*

## Abstract

Let  $\mathcal{A} = \{A_1, \dots, A_d\}$  be a *commuting family* of  $n \times n$  complex matrices, i.e.,  $A_j A_k = A_k A_j$  for all  $1 \leq j, k \leq d$ . Then there exists a unitary matrix  $U$  such that all matrices  $U^* A_1 U, \dots, U^* A_d U$  are upper triangular and the  $n$   $d$ -tuples containing the diagonal elements of  $U^* A_1 U, \dots, U^* A_d U$  are called the *joint eigenvalues* of  $\mathcal{A}$ . For every joint eigenvalue  $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_d)$  of  $\mathcal{A}$  there exists a nonzero *common eigenvector*  $x$ , such that  $A_i x = \lambda_i x$  for  $i = 1, \dots, d$ .

The task of numerical computation of joint eigenvalues for a commuting family arises, e.g., in solvers for multiparameter eigenvalue problems and systems of multivariate polynomials. We propose and analyze a simple approach, summarized in Algorithm 1, that computes eigenvalues as one-sided or two-sided Rayleigh quotients from eigenvectors of a random linear combination

$$A(\mu) = \mu_1 A_1 + \mu_2 A_2 + \cdots + \mu_d A_d, \quad (1)$$

where  $\mu = [\mu_1 \ \cdots \ \mu_d]^T$  is a random vector from the uniform distribution on the unit sphere in  $\mathbb{C}^d$ .

We show that Algorithm 1, in particular the use of two-sided Rayleigh quotients, accurately computes well-conditioned semisimple joint eigenvalues with high probability. It still works satisfactorily in the presence of defective eigenvalues. Experiments show that the method can be efficiently used in solvers for multiparameter eigenvalue problems and roots of systems of multivariate polynomials.

---

### Algorithm 1 Randomized Joint Eigenvalue Approximation

---

**Input:** A nearly commuting family  $\mathcal{A} = \{A_1, \dots, A_d\}$ ,  $\text{opt} \in \{\text{RQ1}, \text{RQ2}\}$ .

**Output:** Approximations of joint eigenvalues of  $\mathcal{A}$ .

- 1: Draw  $\mu \in \mathbb{C}^d$  from the uniform distribution on the unit sphere.
  - 2: Compute  $A(\mu) = \mu_1 A_1 + \cdots + \mu_d A_d$ .
  - 3: Compute invertible matrices  $X, Y$  such that the columns of  $X$  have norm 1,  $Y^* X = I$ , and  $Y^* A(\mu) X$  is diagonal.
  - 4: **if**  $\text{opt} = \text{RQ1}$  **then**
  - 5:   **return**  $\boldsymbol{\lambda}_{\text{RQ1}}^{(i)} = (x_i^* A_1 x_i, \dots, x_i^* A_d x_i), \quad i = 1, \dots, n$ .
  - 6: **else if**  $\text{opt} = \text{RQ2}$  **then**
  - 7:   **return**  $\boldsymbol{\lambda}_{\text{RQ2}}^{(i)} = (y_i^* A_1 x_i, \dots, y_i^* A_d x_i), \quad i = 1, \dots, n$ .
  - 8: **end if**
- 

The idea of using a random linear combination like (1) is not new. For example, in [1, 4] the unitary matrix  $U$  from the Schur decomposition  $A(\mu) = U^* R U$  is used to transform all matrices from  $\mathcal{A}$  to *block* upper triangular form. Using the Schur decomposition, however, requires clustering to group multiple eigenvalues together, and this is a numerically subtle task. On the other hand, Algorithm 1 does not require clustering and in practice often leads to equally good or even better results for, e.g., multiparameter eigenvalue problems [5] and multivariate root finding problems.

For a significantly simpler situation of commuting *Hermitian* matrices, where a unitary matrix exists that jointly diagonalizes all matrices, randomized methods based on (1) have recently been analyzed in [2], establishing favorable robustness and stability properties.

An important source of joint eigenvalue problems are eigenvector-based methods for solving systems of multivariate polynomial equations. If we are looking for roots of a set of polynomials

$$p_i(x_1, \dots, x_d) = 0, \quad i = 1, \dots, m, \quad (2)$$

such that the solution consists of finitely many points, then a common feature of these methods is that they construct so called *multiplication matrices*  $M_{x_1}, \dots, M_{x_d}$  that commute and their joint eigenvalues are the roots  $(x_1, \dots, x_d)$  of (2). Many techniques that use symbolic and/or numerical computation, including Gröbner basis, various resultants, and Macaulay matrices, are used to construct the multiplication matrices, see, e.g., [6].

Another source are *multiparameter eigenvalue problems*. A  $d$ -parameter version has the form

$$A_{i0}x_i = \lambda_1 A_{i1}x_i + \dots + \lambda_d A_{id}x_i, \quad i = 1, \dots, d, \quad (3)$$

where  $A_{ij}$  is an  $n_i \times n_i$  complex matrix and  $x_i \neq 0$  for  $i = 1, \dots, d$ . When (3) is satisfied, a  $d$ -tuple  $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_d) \in \mathbb{C}^d$  is called an *eigenvalue* and  $x_1 \otimes \dots \otimes x_d$  is a corresponding eigenvector. Generically, a multiparameter eigenvalue problem (3) has  $N = n_1 \cdots n_d$  eigenvalues. The problem (3) is related to a system of  $d$  generalized eigenvalue problems

$$\Delta_i z = \lambda_i \Delta_0 z, \quad i = 1, \dots, d,$$

with  $z = x_1 \otimes \dots \otimes x_d$  and the  $N \times N$  matrices (that are called *operator determinants*)

$$\Delta_0 = \begin{vmatrix} A_{11} & \cdots & A_{1d} \\ \vdots & & \vdots \\ A_{d1} & \cdots & A_{dd} \end{vmatrix}_{\otimes}, \quad \Delta_i = \begin{vmatrix} A_{11} & \cdots & A_{1,i-1} & A_{10} & A_{1,i+1} & \cdots & A_{1d} \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ A_{d1} & \cdots & A_{d,i-1} & A_{d0} & A_{d,i+1} & \cdots & A_{dd} \end{vmatrix}_{\otimes}, \quad i = 1, \dots, d.$$

If  $\Delta_0$  is invertible, then the matrices  $\Gamma_i := \Delta_0^{-1} \Delta_i$  for  $i = 1, \dots, d$  commute. If  $N$  is not too large, then a standard approach to solve (3) is to explicitly compute the matrices  $\Gamma_1, \dots, \Gamma_d$  and then solve the joint eigenvalue problem.

## References

- [1] R.M. CORLESS, P.M. GIANNI, B.M. TRAGER, *A reordered Schur factorization method for zero dimensional polynomial systems with multiple roots*, in Proceedings of the 1997 International Symposium on Symbolic and Algebraic Computation (Kihei, HI) 133–140, ACM, New York, 1997.
- [2] H. HE, D. KRESSNER, *Randomized joint diagonalization of symmetric matrices*, SIAM J. Matrix Anal. Appl. 45 (2024) 661–684.
- [3] H. HE, D. KRESSNER, B. PLESTENJAK, *Randomized methods for computing joint eigenvalues, with applications to multiparameter eigenvalue problems and root finding*, arXiv:2409.00500 (2024), to appear in Numer. Algorithms.
- [4] D. MANOCHA, J. DEMMEL, *Algorithms for intersecting parametric and algebraic curves I: simple intersections*, ACM Trans. Graph. 13 (1994) 73–100.
- [5] B. PLESTENJAK, *MultiParEig. Toolbox for multiparameter and singular eigenvalue problems*, [www.mathworks.com/matlabcentral/fileexchange/47844-multipareig](http://www.mathworks.com/matlabcentral/fileexchange/47844-multipareig), MATLAB Central File Exchange.
- [6] H. J. STETTER, *Numerical Polynomial Algebra*, SIAM, Philadelphia, PA, 2004.

# Matrix equations from the $\star$ -algebra with quantum chemistry applications

Stefano Pozza, Christian Bonhomme, Lorenzo Lazzarino, Niel Van Buggenhout

## Abstract

Consider the matrix-valued function  $\tilde{A}(t) \in \mathbb{C}^{N \times N}$  analytic over the bounded interval  $\mathcal{I} = [a, b]$ , the vector  $v \in \mathbb{C}^N$ , and let  $\tilde{u}(t) \in \mathbb{C}^{N \times N}$  be the solution of the non-autonomous ordinary differential equation

$$\frac{\partial}{\partial t} \tilde{u}(t) = \tilde{A}(t) \tilde{u}(t), \quad \tilde{u}(a) = v, \quad t \in \mathcal{I} = [a, b]. \quad (1)$$

When  $\tilde{A}(t)$  commutes with itself at different times, i.e.,  $\tilde{A}(t_1)\tilde{A}(t_2) = \tilde{A}(t_2)\tilde{A}(t_1)$ , the solution is given through the matrix exponential as  $\tilde{u}(t) = \exp(\int_a^t \tilde{A}(\tau) d\tau)v$ ,  $t \in [a, b]$ . However, in the general case, there is no explicit formula for  $\tilde{U}(t)$  in terms of usual matrix functions. In quantum chemistry, spin dynamics are often modeled by Eq. (1) where the time-dependent matrix takes the form

$$\tilde{A}(t) = A_1 f_1(t) + \dots + A_k f_k(t),$$

with  $A_1, \dots, A_k$  (constant) large and sparse matrices,  $f_1(t), \dots, f_k(t)$  scalar analytic functions, and  $k$  a small integer; see, e.g., [11]

In [16, 17], we introduced a new spectral approach for this kind of ODEs, that gives the solution in terms of the coefficients of the expansion  $\tilde{u}(t) = \sum_{j=0}^{\infty} u_j p_j(t)$ ,  $t \in [-1, 1]$ , with  $p_0(t), p_1(t), p_2(t), \dots$  the orthonormal Legendre polynomials, and  $u_j \in \mathbb{C}^N$ . For a large enough integer  $m$ , it is possible to approximate the coefficients  $u_0, \dots, u_m$  of the truncated expansion  $\tilde{u}(t) \approx \sum_{j=0}^{m-1} u_j p_j(t)$  by solving the matrix equation

$$X - F_1 X A_1^T - \dots - F_k X A_k^T = \phi v^T, \quad (2)$$

for a certain vector  $\phi$ , where the  $m \times m$  matrices  $F_1, \dots, F_k$  represents the functions  $f_1(t), \dots, f_k(t)$  in the so-called  $\star$ -algebra [18]. We named the described strategy  $\star$ -approach. The matrix equation (2) enjoys many properties: (i) the matrices  $F_1, \dots, F_k$  are banded; (ii) in the applications of interest, the matrices  $A_1, \dots, A_k$  have a Kronecker structure; (iii) the equation has a rank 1 right-hand side. We can solve the matrix equation by iterative methods exploiting properties (i) and (ii) to reduce the cost of matrix-vector multiplication. Moreover, the rank 1 right-hand side suggests that the solution  $X$  might be numerically low-rank. Indeed, this is the case in all the applications we treated (e.g., [16]). Hence, property (iii) allows for the use of low-rank approximation.

This novel numerical approach has proved highly competitive in the solution of ODEs related to a specific model, the *generalized Rosen-Zener model*. In [2], with Christian Bonhomme (Sorbonne University) and Niel Van Buggenhout (Universidad Carlos III), we introduced a new algorithm, named  $\star$ -method, that exploits properties (i)–(iii). Its computational cost scales linearly with the model size (Fig 2, [2]) and is also highly competitive for increasing interval sizes (Fig 4, [2]). These first results might open the way to more general efficient methods for spin simulations. However, the spectral properties and structure of other, more complex, quantum problems can make the solution of Eq. (2) challenging. For instance, we are working on the solution of an ODE system that considers dipolar interactions in a Nuclear Magnetic Resonance application [12]. In this case, the strategies used in [2] are not efficient enough. This is why we are currently testing randomized approaches (joint work with Lorenzo Lazzarino, University of Oxford) and tensor methods, with promising results.

## A closer look into the $\star$ -approach

The difficulties emerging in the numerical solution of Eq. (1) are linked with the lack of a general analytic expression of  $\tilde{u}(t)$ . Analytic approaches are typically based on Floquet formalism [10, 19], Magnus series [1, 13], or hybrids of these with ad-hoc approximate/numerical methods [3, 14, 20]. These analytic approaches rarely provide exact solutions in a finite number of steps, might suffer from convergence issues [1], and be intractable [4]. There is a perception in the physics community that no exact solutions are achievable [7]. This also influences the development of numerical solvers since the most advanced numerical methods are typically built on analytical approaches [1, 9, 10]. As noted by M. Grifoni and P. Hänggi in [8]: “Solving the time-dependent Schrödinger equation necessitates the development of novel analytic and computational schemes [...] in a nonperturbative manner” – a remark still relevant today. The  $\star$ -approach follows this suggestion as it is based on a new nonperturbative expression for  $\tilde{u}(t)$ , obtained through the so-called  $\star$ -product.

Let us define the set  $\mathcal{A}(\mathcal{I})$  of the bivariate distributions for which there exists a finite  $k$  so that

$$f(t, s) = \tilde{f}_{-1}(t, s)\Theta(t - s) + \tilde{f}_0(t, s)\delta(t - s) + \cdots + \tilde{f}_k(t, s)\delta^{(k)}(t - s),$$

where  $\Theta(t - s)$  is the Heaviside function ( $\Theta(t - s) = 1$  for  $t \geq s$ , and 0 otherwise), and  $\delta(t - s), \delta'(t - s), \delta^{(2)}(t - s), \dots$  are the Dirac delta and its derivatives. The  $\star$ -product of  $f, g \in \mathcal{A}(\mathcal{I})$  is the non-commutative product defined as

$$(f \star g)(t, s) := \int_{\mathcal{I}} f(t, \tau)g(\tau, s) d\tau \in \mathcal{A}(\mathcal{I});$$

see [18]. The  $\star$ -product straightforwardly extends to a scalar-matrix  $\star$ -product and to a matrix-matrix (matrix-vector)  $\star$ -product for matrices with compatible sizes composed of elements from  $\mathcal{A}(\mathcal{I})$ . We denote with  $\mathcal{A}^{N \times M}(\mathcal{I})$  the space of the  $N \times M$  matrices with elements from  $\mathcal{A}(\mathcal{I})$ . Note that  $I_\star = I\delta(t - s)$  is the identity matrix in  $\mathcal{A}^{N \times N}(\mathcal{I})$ . As shown in [5], the solution  $\tilde{u}(t)$  of the Eq. (1) can then be expressed as

$$\tilde{u}(t) = u(t, a), \quad u(t, s) = \Theta(t - s) \star x(t, s), \quad (3)$$

$$\left( I_\star - \tilde{A}(t)\Theta(t - s) \right) \star x(t, s) = \tilde{v}\delta(t - s), \quad t \in [a, b], \quad (4)$$

with  $x(t, s) \in \mathcal{A}^N(\mathcal{I})$ . Therefore, solving a system of non-autonomous linear ODEs is equivalent to solving a linear system in the  $\star$ -algebra. Note that, for  $m = \infty$ , the matrix equation (2) is the matrix algebra counterpart of the  $\star$ -linear system (4); see [15, 16, 17]. As a consequence, numerical methods for the solution of Eq. (2) can be interpreted as algorithms in the  $\star$ -algebra. Vice versa, it is possible to devise new techniques (such as preconditioners) and numerical methods (e.g., the  $\star$ -Lanczos algorithm [6]) in the  $\star$ -algebra and then map them in the usual algebra of matrices [15] where they can be implemented.

In conclusion, the  $\star$ -approach has proved extremely fast in certain quantum applications [2]. Its success is based, on the one hand, on advanced linear algebra techniques and, on the other, on applying  $\star$ -algebra results in numerical linear algebra.

## References

- [1] S. Blanes, F. Casas, J. A. Oteo, and J. Ros. The Magnus expansion and some of its applications. *Phys. Rep.*, 470(5):151 – 238, 2009.

- [2] C. Bonhomme, S. Pozza, and N. Van Buggenhout. A new fast numerical method for the generalized Rosen-Zener model. arXiv:2311.04144 [math.NA], 2023.
- [3] A. Brinkmann. Introduction to average Hamiltonian theory. I. Basics. *Concepts in Magnetic Resonance Part A: Bridging Education and Research*, 45A:e21414, 2016.
- [4] M. Dalgaard and F. Motzoi. Fast, high precision dynamics in quantum optimal control theory. *J. Phys. B*, 55(8):085501, 2022.
- [5] P.-L. Giscard, K. Lui, S. J. Thwaite, and D. Jaksch. An exact formulation of the time-ordered exponential using path-sums. *J. Math. Phys.*, 56(5):053503, 2015.
- [6] P.-L. Giscard and S. Pozza. A Lanczos-like method for non-autonomous linear ordinary differential equations. *Boll. Unione Mat. Ital.*, 16:81–102, 2023.
- [7] C. Graf, A. Rund, C. S. Aigner, and R. Stollberger. Accuracy and performance analysis for Bloch and Bloch-McConnell simulation methods. *J. Magn. Reson.*, 329:107011, 2021.
- [8] M. Grifoni and P. Hänggi. Driven quantum tunneling. *Phys. Rep.*, 304(5):229–354, 1998.
- [9] M. Hochbruck and C. Lubich. On Magnus integrators for time-dependent Schrödinger equations. *SIAM J. Numer. Anal.*, 41(3):945–963, 2003.
- [10] K. L. Ivanov, K. R. Mote, M. Ernst, A. Equbal, and P. K. Madhu. Floquet theory in magnetic resonance: Formalism and applications. *Prog. Nucl. Magn. Reson. Spectrosc.*, 126-127:17–58, 2021.
- [11] I. Kuprov. *Spin: From Basic Symmetries to Quantum Optimal Control*. Springer International Publishing, 2023.
- [12] L. Lazzarino. *Numerical approximation of the time-ordered exponential for spin dynamic simulation*. PhD thesis, Univerzita Karlova, Prague, 2023.
- [13] W. Magnus. On the exponential solution of differential equations for a linear operator. *Comm. Pure Appl. Math.*, 7(4):649–673, 1954.
- [14] E. Mananga. Theoretical perspectives of spin dynamics in solid-state Nuclear Magnetic Resonance and physics. *J. Mod. Phys.*, 9:1645–1659, 2018.
- [15] S. Pozza. A new closed-form expression for the solution of ODEs in a ring of distributions and its connection with the matrix algebra. *Linear Multilinear Algebra*, advance online publication, DOI:10.1080/03081087.2024.2303058, 2024.
- [16] S. Pozza and N. Van Buggenhout. A new matrix equation expression for the solution of non-autonomous linear systems of ODEs. *PAMM*, 22:e202200117, 2023.
- [17] S. Pozza and N. Van Buggenhout. A new Legendre polynomial-based approach for non-autonomous linear ODEs. *Electron. Trans. Numer. Anal.*, 60:292–326, 2024.
- [18] M. Ryckebusch. A Fréchet-Lie group on distributions. arXiv:2307.09037 [math.FA], 2023.
- [19] J. H. Shirley. Solution of the Schrödinger equation with a Hamiltonian periodic in time. *Phys. Rev.*, 138:B979–B987, 1965.
- [20] M. Vogl, P. Laurell, A. D. Barr, and G. A. Fiete. Flow equation approach to periodically driven quantum systems. *Phys. Rev. X*, 9:021037, 2019.

# Optimizing Rayleigh quotient with symmetric constraints and its application to eigenvalue backward errors of polynomial and rational eigenvalue problems

*Anshul Prajapati, Punit Sharma*

## Abstract

Let  $H \in \mathbb{C}^{n,n}$  be Hermitian and  $S_0, S_1, \dots, S_k \in \mathbb{C}^{n,n}$  be symmetric matrices. We consider the problem of maximizing the Rayleigh quotient of  $H$  with respect to certain constraints involving symmetric matrices  $S_0, S_1, \dots, S_k$ . More precisely, we compute

$$m_{hs_0s_1\dots s_k}(H, S_0, S_1, \dots, S_k) := \sup \left\{ \frac{v^* H v}{v^* v} : v \in \mathbb{C}^n \setminus \{0\}, v^T S_i v = 0 \right. \\ \left. \text{for } i = 0, \dots, k \right\}, \quad (\mathcal{P})$$

where  $T$  and  $*$  denote respectively the transpose and the conjugate transpose of a matrix or a vector.

Such problems occur in stability analysis of uncertain systems and in the eigenvalue perturbation theory of matrices and matrix polynomials [3, 4]. A particular case of problem  $(\mathcal{P})$  with only one symmetric constraint (i.e., when  $k = 0$ ) is used to characterize the  $\mu$ -value of the matrix under skew-symmetric perturbations [4]. An explicit computable formula was obtained for  $m_{hs_0}(H, S_0)$  in [4, Theorem 6.3] and given by

$$m_{hs_0}(H, S_0) = \inf_{t \in [0, \infty)} \lambda_2 \left( \begin{bmatrix} H & t\bar{S}_0 \\ tS_0 & \bar{H} \end{bmatrix} \right),$$

where  $\lambda_2(A)$  stands for the second largest eigenvalue of a Hermitian matrix  $A$ . However, the solution to the problem  $(\mathcal{P})$  with more than one symmetric constraint was not known.

We derive an explicit computable formula for  $(\mathcal{P})$  in terms of minimizing the second largest eigenvalue of a parameter-depending Hermitian matrix under a simplicity assumption. The results are then applied to derive computable formulas for the structured eigenvalue backward errors of rational matrix functions (RMFs) of the following form

$$G(z) = A_0 + zA_1 + \dots + z^d A_d + \frac{s_1(z)}{q_1(z)} E_1 + \dots + \frac{s_k(z)}{q_k(z)} E_k$$

where the coefficients  $A_p, p = 0, 1, \dots, d$  and  $E_j, j = 1, 2, \dots, k$  are  $n \times n$  matrices, and  $s_j(z), q_j(z)$ , for  $j = 1, 2, \dots, k$  are scalar polynomials.

Eigenvalue backward errors of matrix polynomials, both for unstructured and structure-preserving perturbations, have been studied in the literature; see [9] for unstructured, [1] for Hermitian and related structures, and [2] for palindromic and related structures. However, the literature on RMFs is relatively limited, and the structured eigenvalue backward errors have not been explored before.

To explore this, we first aim to reformulate the problem of computing structured eigenvalue backward errors for RMFs with symmetric, skew-symmetric, T-even, T-odd, and T-palindromic structures into the optimization problem  $(\mathcal{P})$ . We then apply the results obtained for this optimization

problem to derive computable formulas for the structured eigenvalue backward errors of RMFs. As a specific case of RMFs, formulas for the structured eigenvalue backward errors of matrix polynomials with the aforementioned structures can also be derived. Numerical experiments suggest that our results [5] provide a more accurate estimation of the supremum in  $(\mathcal{P})$  compared to the one in [7]. Some of these results are published in Linear Algebra and its Applications [5], while others in BIT Numerical Mathematics [6].

## References

- [1] S. Bora, M. Karow, C. Mehl, P. Sharma. Structured eigenvalue backward errors of matrix pencils and polynomials with Hermitian and related structures. *SIAM J. Matrix Anal. Appl.*, 35: 453–475 (2014).
- [2] S. Bora, M. Karow, C. Mehl, P. Sharma. Structured eigenvalue backward errors of matrix pencils and polynomials with palindromic structures. *SIAM J. Matrix Anal. Appl.*, 36: 393–416 (2015).
- [3] J. Doyle. Analysis of feedback systems with structured uncertainties. *IEE Proc. Part D, Control Theory Appl.*, 129: 242–250 (1982).
- [4] M. Karow.  $\mu$ -values and spectral value sets for linear perturbation classes defined by a scalar product. *SIAM J. Matrix Anal. Appl.*, 32: 845–865 (2011).
- [5] A. Prajapati, P. Sharma. Optimizing the Rayleigh quotient with symmetric constraints and its application to perturbations of structured polynomial eigenvalue problems. *Linear Algebra Appl.*, 645: 256–277 (2022).
- [6] A. Prajapati, P. Sharma. Structured eigenvalue backward errors for rational matrix functions with symmetry structures, *BIT Numer Math*, 64, 10 (2024).
- [7] P. Sharma. Eigenvalue Backward errors of polynomial eigenvalue problems under structure preserving perturbations. *PhD thesis, Department of Mathematics, Indian Institute of Technology Guwahati, India*, (2016).
- [8] F. Tisseur, K. Meerbergen. The quadratic eigenvalue problem. *SIAM Review*, 43: 235–286 (2001).
- [9] F. Tisseur. Backward error and condition of polynomial eigenvalue problems. *Linear Algebra Appl.*, 309: 339–361 (2000).

# The Fundamental Subspaces of Ensemble Kalman Inversion

*Elizabeth Qian, Christopher Beattie*

## Abstract

Ensemble Kalman Inversion (EKI) methods are a family of iterative methods for solving weighted least-squares problems of the form

$$\min_{\mathbf{v} \in \mathbb{R}^d} (\mathbf{y} - \mathbf{H}(\mathbf{v}))^\top \boldsymbol{\Sigma}^{-1} (\mathbf{y} - \mathbf{H}(\mathbf{v})) = \min_{\mathbf{v} \in \mathbb{R}^d} \|\mathbf{y} - \mathbf{H}(\mathbf{v})\|_{\boldsymbol{\Sigma}^{-1}}^2, \quad (1)$$

where  $\boldsymbol{\Sigma} \in \mathbb{R}^{n \times n}$  is symmetric positive definite, and  $\mathbf{H} : \mathbb{R}^d \rightarrow \mathbb{R}^n$ . Such problems arise in many settings, including in *inverse problems* in which  $\mathbf{v} \in \mathbb{R}^d$  represents an unknown parameter or state of a system of interest which must be inferred from observed data  $\mathbf{y} \in \mathbb{R}^n$ . Inverse problems arise in many disciplines across science, engineering, and medicine, including earth, atmospheric, and ocean modeling, medical imaging, robotics and autonomy, and more. In large-scale scientific and engineering applications, solving (1) using standard gradient-based optimization methods can be prohibitively expensive due to the high cost of evaluating derivatives or adjoints of the forward operator  $\mathbf{H}$ . In contrast, EKI methods can be implemented in an *adjoint-/derivative-free* way. This makes EKI an attractive alternative to gradient-based methods for solving (1) in large-scale inverse problems.

We introduce a basic version of EKI from [4] in Algorithm 1, noting that other EKI methods can be viewed as variations on this theme. In Algorithm 1, we use  $\mathsf{E}$  and  $\text{cov}$  to denote the *empirical* (sample) mean and covariance operators, respectively: given  $J \in \mathbb{N}$  samples  $\{\mathbf{a}^{(j)}\}_{j=1}^J$  and  $\{\mathbf{b}^{(j)}\}_{j=1}^J$ , we define  $\mathsf{E}[\mathbf{a}^{(1:J)}] = \frac{1}{J} \sum_{j=1}^J \mathbf{a}^{(j)}$ , and

$$\text{cov}[\mathbf{a}^{(1:J)}, \mathbf{b}^{(1:J)}] = \frac{1}{J-1} \sum_{j=1}^J (\mathbf{a}^{(j)} - \mathsf{E}[\mathbf{a}^{(1:J)}])(\mathbf{b}^{(j)} - \mathsf{E}[\mathbf{b}^{(1:J)}])^\top,$$

and  $\text{cov}[\mathbf{a}^{(1:J)}] = \text{cov}[\mathbf{a}^{(1:J)}, \mathbf{a}^{(1:J)}]$ . Algorithm 1 prescribes the evolution of an ensemble of  $J$  particles,  $\{\mathbf{v}_i^{(1)}, \dots, \mathbf{v}_i^{(J)}\}$ , initialized at  $i = 0$  in some way, e.g., by drawing from a suitable prior distribution, and subsequently updated for  $i = 1, 2, 3$ , etc. We emphasize that Algorithm 1 does not require the evaluation of adjoints or derivatives of  $\mathbf{H}$ . Those familiar with ensemble Kalman *filtering* methods will recognize familiar elements in Algorithm 1. Indeed, one way to obtain Algorithm 1 is to apply the ensemble Kalman filter to a system whose dynamics are given by the identity map in the “forecast” step of the filter. The connection to the ensemble Kalman filter also motivates the perturbation of the observations by random noise in Step 7; these perturbations ensure unbiased estimates of the filtering statistics in the linear Gaussian setting.

There is a very rich literature developing both EKI methods and accompanying theory (see [3] for an extensive survey). Variants of the basic method include the incorporation of a Tikhonov regularization term into the least-squares objective function, the enforcement of constraints in the optimization, or hierarchical, multilevel, and parallel versions of the algorithm. Beyond the successful use of EKI for solving diverse inverse problems in the physical sciences, e.g., in geophysical and biological contexts, EKI has also been used as an optimizer for training machine learning models. In particular, the use of EKI for training neural networks has motivated the development of EKI variants based on ideas used for gradient-based training of neural networks, including dropout,

---

**Algorithm 1** Basic Ensemble Kalman Inversion (EKI)

---

- 0: **Input:** forward operator  $\mathbf{H} : \mathbb{R}^d \rightarrow \mathbb{R}^n$ , initial ensemble  $\{\mathbf{v}_0^{(1)}, \dots, \mathbf{v}_0^{(J)}\} \subset \mathbb{R}^d$ , observations  $\mathbf{y} \in \mathbb{R}^n$ , observation error covariance  $\Sigma \in \mathbb{R}^{n \times n}$
- 1: **for**  $i = 0, 1, 2, \dots$ , **do**
- 2:   Compute observation-space ensemble:  $\mathbf{h}_i^{(j)} = \mathbf{H}(\mathbf{v}_i^{(j)})$ ,  $j = 1, 2, \dots, J$ .
- 3:   Compute empirical covariances:  $\text{cov}[\mathbf{v}_i^{(1:J)}, \mathbf{h}_i^{(1:J)}]$  and  $\text{cov}[\mathbf{h}_i^{(1:J)}]$
- 4:   Compute Kalman gain:  $\mathbf{K}_i = \text{cov}[\mathbf{v}_i^{(1:J)}, \mathbf{h}_i^{(1:J)}] \cdot (\text{cov}[\mathbf{h}_i^{(1:J)}] + \Sigma)^{-1}$
- 5:   Sample  $\boldsymbol{\varepsilon}_i^{(j)}$  i.i.d. from  $\mathcal{N}(0, \Sigma)$  for  $j = 1, 2, \dots, J$ .
- 6:   Perturb observations: set  $\mathbf{y}_i^{(j)} = \mathbf{y} + \boldsymbol{\varepsilon}_i^{(j)}$  for  $j = 1, 2, \dots, J$ .
- 7:   Compute particle update:  $\mathbf{v}_{i+1}^{(j)} = \mathbf{v}_i^{(j)} + \mathbf{K}_i(\mathbf{y}_i^{(j)} - \mathbf{H}\mathbf{v}_i^{(j)})$  for  $j = 1, 2, \dots, J$ .
- 8:   **if** converged **then**
- 9:     **return** current ensemble mean,  $E[\mathbf{v}_{i+1}^{(1:J)}]$

This is **Stochastic EKI**. For **Deterministic EKI**, skip 5-6 and assign  $\mathbf{y}_i^{(j)} = \mathbf{y}$  in 7.

---

data subsampling (also called ‘(mini-)batching’), adaptive step sizes, and convergence acceleration with Nesterov momentum.

Theoretical analyses of EKI convergence behavior have mostly considered linear observation operators  $\mathbf{H} \in \mathbb{R}^{n \times d}$ , for which the standard norm-minimizing solution of (1) is given by

$$\mathbf{v}^* = (\mathbf{H}^\top \Sigma^{-1} \mathbf{H})^\dagger \mathbf{H}^\top \Sigma^{-1} \mathbf{y} \equiv \mathbf{H}^+ \mathbf{y}, \quad (2)$$

where “ $\dagger$ ” denotes the usual Moore-Penrose pseudoinverse and we have introduced the weighted pseudoinverse,  $\mathbf{H}^+ = (\mathbf{H}^\top \Sigma^{-1} \mathbf{H})^\dagger \mathbf{H}^\top \Sigma^{-1}$ . Previous analyses of linear EKI have largely considered mean-field limits (equivalent to an infinitely large ensemble) [3] or continuous-time limits of the EKI iteration, in which the deterministic iteration becomes a system of coupled ordinary differential equations (ODEs) [2, 6] and the stochastic iteration becomes a system of coupled stochastic differential equations (SDEs) [1]. These continuous-time analyses have shown that the EKI ensemble covariance collapses at a rate inversely proportional to time [6, 1, 2], meaning that the residual of the EKI iteration (with respect to the final solution) converges at a  $1/\sqrt{i}$  rate. These analyses have also characterized EKI solutions either by assuming  $\mathbf{H}$  is one-to-one or by assuming the ensemble covariance is full rank. In particular, the works [6] show that if  $\mathbf{H}$  is one-to-one, then EKI converges to the pre-image of the data restricted to the span of the ensemble [6, 1]. On the other hand, the work [2] shows that if the ensemble covariance is full rank (and  $\mathbf{H}$  may be low-rank), then EKI converges to the (non-standard) minimizer of (1) closest to the initial ensemble mean in the norm induced by the initial ensemble covariance. The characterization of EKI solutions in the general case where both  $\mathbf{H}$  and the ensemble covariance may be low-rank, and the relationship between EKI solutions and the standard minimum-norm least-squares solution (2), are open questions addressed in this work.

In this work, we provide a new analysis of EKI for linear observation operators  $\mathbf{H} \in \mathbb{R}^{n \times d}$  which directly considers the discrete iteration for a finite ensemble, relying principally on linear algebra as an analysis tool. Our analysis yields new results relating EKI solutions to the standard minimum-norm least-squares solution (2), together with a new and natural interpretation of EKI convergence behavior in terms of ‘fundamental subspaces of EKI’, analogous to the four fundamental subspaces characterizing Strang’s ‘fundamental theorem of linear algebra’ [7], which we now review.

Strang’s four ‘fundamental subspaces of linear algebra’ arise from dividing observation space  $\mathbb{R}^n$

and state space  $\mathbb{R}^d$  into two subspaces each, one subspace associated with ‘observable’ directions and a complementary subspace associated with ‘unobservable’ directions [7]. That is, in observation space  $\mathbb{R}^n$ , the two fundamental subspaces are:

1.  $\text{Ran}(\mathbf{H})$  (denoting the range of  $\mathbf{H}$ ), and
2.  $\text{Ker}(\mathbf{H}^\top \boldsymbol{\Sigma}^{-1})$  (denoting the null space of  $\mathbf{H}^\top \boldsymbol{\Sigma}^{-1}$ ), the  $\boldsymbol{\Sigma}^{-1}$ -orthogonal complement to  $\text{Ran}(\mathbf{H})$ .

In state space  $\mathbb{R}^d$ , the two fundamental subspaces are:

1.  $\text{Ran}(\mathbf{H}^\top)$ , and
2.  $\text{Ker}(\mathbf{H})$ , the orthogonal complement to  $\text{Ran}(\mathbf{H}^\top)$  with respect to the Euclidean norm.

The standard minimum-norm solution (2) to the linear least-squares problem (1) can be understood in terms of these fundamental subspaces as follows (see [5, Figure 1]): in observation space  $\mathbb{R}^n$ , the closest that  $\mathbf{H}\mathbf{v}$  can come to  $\mathbf{y} \in \mathbb{R}^n$  with respect to the  $\boldsymbol{\Sigma}^{-1}$ -norm is the  $\boldsymbol{\Sigma}^{-1}$ -orthogonal projection of  $\mathbf{y}$  onto the observable space  $\text{Ran}(\mathbf{H})$ , which then has a zero component in the (unobservable) subspace  $\text{Ker}(\mathbf{H}^\top \boldsymbol{\Sigma}^{-1})$ . In state space  $\mathbb{R}^d$ , directions in  $\text{Ker}(\mathbf{H})$  are unobservable because they are mapped by  $\mathbf{H}$  to zero and thus do not influence the minimand of (1). If  $\text{Ker}(\mathbf{H})$  is non-trivial, multiple minimizers of (1) exist. The unique norm-minimizing solution (2) lies in the observable space  $\text{Ran}(\mathbf{H}^\top)$  and has a zero component in the unobservable space  $\text{Ker}(\mathbf{H})$ .

Our analysis reveals that EKI solutions to the weighted least squares problem admit a similar interpretation in terms of fundamental subspaces of EKI. However, the EKI fundamental subspaces arise first from dividing the state and observation spaces into directions that are ‘populated’ by particles, lying in the range of ensemble covariance  $\boldsymbol{\Gamma}_i$  (it is well-known [1, 2, 4, 6] that  $\text{Ran}(\boldsymbol{\Gamma}_i)$  is invariant for all  $i$ ), and ‘unpopulated’ directions lying in a complementary subspace. The populated subspace can then be further divided into two subspaces associated with observable and unobservable directions. There are therefore three subspaces in each of the observation and state spaces. In observation space  $\mathbb{R}^n$ , the three fundamental subspaces of EKI are associated with three complementary oblique projection operators,  $\mathcal{P}, \mathcal{Q}, \mathcal{N} \in \mathbb{R}^{n \times n}$ . These projections are defined via a spectral analysis of the iteration map which governs the evolution of the *data misfit*,  $\mathbf{H}\mathbf{v}_i^{(j)} - \mathbf{y}$ , so that the range of each projector is an invariant subspace under the misfit iteration map. The three fundamental subspaces of observation space  $\mathbb{R}^n$  are then

1.  $\text{Ran}(\mathcal{P}) \equiv \text{Ran}(\mathbf{H}\boldsymbol{\Gamma}_i)$ , associated with observable populated directions,
2.  $\text{Ran}(\mathcal{Q}) \equiv \mathbf{H}\text{Ker}(\boldsymbol{\Gamma}_i \mathbf{H}^\top \boldsymbol{\Sigma}^{-1} \mathbf{H})$ , associated with observable but *unpopulated* directions, and
3.  $\text{Ran}(\mathcal{N}) \equiv \text{Ker}(\mathbf{H}^\top \boldsymbol{\Sigma}^{-1})$ , associated with unobservable directions.

In state space  $\mathbb{R}^d$ , the three fundamental subspaces of EKI are also associated with three complementary oblique projection operators,  $\mathbb{P}, \mathbb{Q}, \mathbb{N} \in \mathbb{R}^{n \times n}$ . These projections are defined via a spectral analysis of the iteration map which governs the evolution of the *least squares residual*,  $\mathbf{v}_i^{(j)} - \mathbf{v}^*$ . The range of each projector is again an invariant subspace under the residual iteration map. The three fundamental subspaces of state space  $\mathbb{R}^d$  are then

1.  $\text{Ran}(\mathbb{P}) \subset \text{Ran}(\boldsymbol{\Gamma}_i)$ , associated with observable populated directions (but generally *not* simply the intersection of  $\text{Ran}(\boldsymbol{\Gamma}_i)$  with  $\text{Ran}(\mathbf{H}^\top)$ ),
2.  $\text{Ran}(\mathbb{Q}) \subset \text{Ran}(\mathbf{H}^\top)$ , associated with observable *unpopulated* directions, and
3.  $\text{Ran}(\mathbb{N})$ , associated with unobservable directions.

The fundamental subspaces of EKI are depicted in [5, Figure 2], and an interactive three-dimensional visualization is available at <https://elizqian.github.io/eki-fundamental-subspaces/>.

We show that EKI misfits [residuals] converge to zero at a  $1/\sqrt{i}$  rate in the fundamental subspace associated with observable and populated directions,  $\text{Ran}(\mathcal{P})$  [ $\text{Ran}(\mathbb{P})$ ], and remain constant in the fundamental subspaces associated with observable unpopulated directions,  $\text{Ran}(\mathcal{Q})$  [ $\text{Ran}(\mathbb{Q})$ ]. The misfits [residuals] also remain constant in the unobservable directions,  $\text{Ran}(\mathcal{N})$  [ $\text{Ran}(\mathbb{N})$ ]. Numerical experiments illustrating these results may be found in [5, Figure 3]. Our results verify for the discrete iteration and finite ensemble case the  $1/\sqrt{i}$  convergence rate previously shown in continuous time or infinite ensemble limits, and provide the first results describing the relationship between EKI solutions and the standard minimum-norm least squares solution (2).

Our analysis sheds light on several directions of interest for future work connecting EKI with classical iterative methods. Because we have shown that the convergence behavior of deterministic EKI can be characterized in terms of an evolving spectral problem that has invariant subspaces that are independent of iteration index, this allows for straightforward EKI acceleration strategies analogous to overrelaxation schemes in classical stationary iterative methods. Other potential directions of interest could exploit the well-known connection between the extended Kalman filter and Gauss-Newton methods to establish further connections between EKI and classical methods.

## References

- [1] Dirk Blömker, Claudia Schillings, Philipp Wacker, and Simon Weissmann. Well posedness and convergence analysis of the ensemble Kalman inversion. *Inverse Problems*, 35(8):085007, 2019.
- [2] Leon Bungert and Philipp Wacker. Complete deterministic dynamics and spectral decomposition of the linear ensemble Kalman inversion. *SIAM/ASA Journal on Uncertainty Quantification*, 11(1):320–357, 2023.
- [3] Edoardo Calvello, Sebastian Reich, and Andrew M Stuart. Ensemble Kalman methods: A mean field perspective. *arXiv preprint arXiv:2209.11371*, 2022.
- [4] Marco A. Iglesias, Kody J. H. Law, and Andrew M. Stuart. Ensemble Kalman methods for inverse problems. *Inverse Problems*, 29:045001, 2013.
- [5] Elizabeth Qian and Christopher Beattie. The fundamental subspaces of ensemble Kalman inversion. *arXiv:2409.08862*, 2024.
- [6] Claudia Schillings and Andrew M. Stuart. Analysis of the ensemble Kalman filter for inverse problems. *SIAM Journal on Numerical Analysis*, 55(3):1264–1290, 2017.
- [7] Gilbert Strang. The fundamental theorem of linear algebra. *The American Mathematical Monthly*, 100(9):848–855, 1993.

# On efficiency and adaptivity of sketch-and-project approach in randomized linear solvers

*Elizaveta Rebrova*

*Based on the joint works with Michal Derezhinski, Deanna Needell, Daniel LeJeune, Jackie Lok.*

## Abstract

The sketch-and-project is a unifying framework for many known randomized iterative methods for solving linear systems, such as randomized Kaczmarz and coordinate descent algorithms, their block variants, as well as the extensions to non-linear optimization problems. Given a linear system  $Ax = b$ , the general scheme iterates to find its solution in the following way: for  $t = 0, 1, \dots$  a random sketching matrix  $S = S(t)$  is sampled from a distribution of (random) matrices and the update rule is given by

$$x_{t+1} = \arg \min_{x \in R^n} \|x_t - x\|_B^2 \quad \text{such that} \quad SAx = Sb. \quad (1)$$

This optimization problem can be solved directly and is equivalent to an iterative step

$$x_{t+1} = x_t - B^{-1}A^\top S^\top (SAB^{-1}A^\top S^\top)^\dagger S(Ax_t - b).$$

The performance of these methods is often measured via the expected convergence rate:

$$E \|x_t - x_*\|_B^2 \leq (1 - \rho)^t \cdot \|x_0 - x_*\|_B^2 \quad \forall x_0, t$$

where  $\rho \geq \lambda_{\min}(E[(S\tilde{A})^\dagger S\tilde{A}])$  and  $\tilde{A} = AB^{-1/2}$ . Some of well-known special cases, such as Randomized Kaczmarz method (when the sketching matrix  $S$  samples individual rows of  $A$  and  $B = I$ ), have been mainly used as simple linear solvers that can demonstrate computational efficiency in the cases of vastly overdetermined linear systems, essentially, when the equations arrive in the streaming way. However, this perspective is far from complete. In the talk, I will discuss several recent results demonstrating other conceptual advantages of the sketch-and-project based linear solvers.

## 1. Fast linear solvers for the systems with low-rank structure.

It is expected that the sketching matrices  $S \in R^{k \times m}$  with larger sketch size  $k$  improve per-iteration convergence of the solver but clearly they become more computationally expensive within every step. In [DR24], we have quantified the advantage of increasing the sketch size and connected it to the *tail condition number* of the system, that is, singular values of  $A$  excluding the top  $k$  of them.

Based on this improved convergence rate analysis, one can build a practical linear solver with several very natural enhancements of the generic computation scheme, that is, (a) particular not too small sketch size and very sparse sketching, or block sampling on the system preconditioned by the Randomized Hadamard Transform (b) inexact solve in place of the pseudoinverse inversion, and (c) adding the momentum. Such solver can compute  $\tilde{x}$  such that  $\|A\tilde{x} - b\| \leq \epsilon\|b\|$  in time:  $\tilde{O}\left(\frac{\sigma_\ell}{\sigma_n} \cdot n^2 \log 1/\epsilon\right)$ , where  $\ell = C\sqrt{n}$  for some  $C > 0$ , where  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n > 0$  are singular values of the matrix  $A$ . This directly improves the standard time complexity for linear solvers, such

as  $\tilde{O}(\kappa \cdot n^2)$ , due to its independence from the full condition number (and the top of the spectrum in particular) on the linear systems with relatively flat tails of the spectrum. Such linear systems appear in a variety of standard applications such as spiked covariance models and kernel machines, or when the linear system is explicitly regularized, such as ridge regression.

Another consequence of these results, obtained in [DLNR24], is that the sketch-and-project approach can be also viewed as implicit preconditioning by the iterative sparse sketching (or sampling) procedure.

## 2. Subspace constrained iterative methods for linear solvers with prior information

The adaptive nature of the sketch-and-project iteration can be also used to guide the iterations when additional information is available about the system. In [LR24], we propose a version of the randomized Kaczmarz algorithm for solving systems of linear equations where the iterates are confined to the solution space of a selected subsystem. We show that the subspace constraint leads to an accelerated convergence rate, especially when the system has approximately low-rank structure that can be estimated before solving the system. Another natural place for a subspace constraint appears if only a part of a linear system changes with time or one is solving a sequence of otherwise connected linear systems. On Gaussian-like random data, we show that the proposed Subspace Constrained Randomized Kaczmarz method results in a form of dimension reduction that effectively increases the aspect ratio of the system.

## 3. Robust linear solvers for the systems with sparse corruptions

Finally, the adaptivity of the considered iterative framework gives another prominent application to solving linear systems contaminated with sparse corruptions. This setting models applications where some measurements are corrupted by arbitrarily large errors, which may occur during the data collection, transmission, and storage process due to malfunctioning sensors or faulty components. For sufficiently tall and regular systems  $A \in \mathbb{R}^{m \times n}$ ,  $m \geq O(n)$ , the iterates of a simple data-aware method based on the Randomized Kaczmarz algorithm converge to  $x_*$  avoiding a significant portion of arbitrary corruptions [HNRS22]. Moreover, the subspace constraining approach discussed above allows to efficiently utilize external knowledge about corruption-free equations and achieve convergence in difficult settings, such as not very overdetermined  $((1 + \alpha)n \times n)$  linear systems [LR24].

## References

- [DLNR24] Michał Dereziński, Daniel LeJeune, Deanna Needell, and Elizaveta Rebrova. Fine-grained analysis and faster algorithms for iteratively solving linear systems. *arXiv preprint arXiv:2405.05818*, 2024.
- [DR24] Michał Dereziński and Elizaveta Rebrova. Sharp analysis of sketch-and-project methods via a connection to randomized singular value decomposition. *SIAM Journal on Mathematics of Data Science*, 6(1):127–153, 2024.

- [HNRs22] Jamie Haddock, Deanna Needell, Elizaveta Rebrova, and William Swartworth. Quantile-based iterative methods for corrupted systems of linear equations. *SIAM Journal on Matrix Analysis and Applications*, 43(2):605–637, 2022.
- [LR24] Jackie Lok and Elizaveta Rebrova. A subspace constrained randomized kaczmarz method for structure or external knowledge exploitation. *Linear Algebra and its Applications*, 2024.

# Analysis of Stochastic Probing Methods for Estimating the Trace of Functions of Sparse Symmetric Matrices

*Andreas Frommer, Michele Rinelli, Marcel Schweitzer*

## Abstract

Estimating the trace of an implicitly given matrix  $B \in \mathbb{R}^{n \times n}$ ,

$$\mathrm{tr}(B) = \sum_{i=1}^n [B]_{ii}, \quad (1)$$

is an important task in many areas of applied mathematics and computer science. In many of these applications, we have  $B = f(A)$ , where  $A \in \mathbb{R}^{n \times n}$  is a large and sparse (or structured) matrix. A common practice is to approximate (1) with an estimator of the form

$$\mathrm{tr}(B) \approx \sum_{k=1}^N \mathbf{v}_k^T B \mathbf{v}_k, \quad (2)$$

for suitably crafted vectors  $\mathbf{v}_1, \dots, \mathbf{v}_N$ . With this approach, approximating (1) relies on matrix-vector products or quadratic forms with  $B$ , which are, e.g., performed by applying a polynomial (or rational) Krylov subspace method or a Chebyshev expansion for approximating  $f(A)\mathbf{v}$  or  $\mathbf{v}^T f(A)\mathbf{v}$ , avoiding the often prohibitive tasks of forming  $B$  or computing the eigenvalues of  $A$ .

Prominent examples are *stochastic estimators*, including Hutchinson's method [5], based on choosing random vectors in (2), and recent variants based on low-rank approximations, Hutch++ [6] and XTrace [2], which work especially well if a fast decay is present in the singular values of  $B$ .

When  $B = f(A)$  with sparse, symmetric  $A$ , a popular other class of methods are based on *probing* [4, 7]. This approach requires the computation of a distance- $d$  coloring of the graph  $\mathcal{G}(A)$  associated with  $A$ , which is a feasible task only under suitable assumptions; see [4]. The *probing estimator* is obtained by using *probing vectors* in (2), i.e., vectors associated with each color whose entries are 0 or 1 depending on the coloring pattern. In [4], the authors show that the error of the probing approximation is bounded by  $n \cdot \eta_d$ , where  $\eta_d$  decays with a rate that depends on how regular  $f$  is over the spectrum of  $A$ . The numerical experiments in [4] prove that  $O(n)$  bounds are the best we can achieve with this method.

We consider a *stochastic probing* approach, given by the combination of probing techniques with stochastic estimators. The nonzero entries of the *stochastic probing vectors* are the same as the deterministic counterparts, but filled with  $\pm 1$  with a uniform distribution (Rademacher entries). This allows to average more than one vector per color,

with an improvement on the convergence related to Hutchinson's estimator. Although this combination is algorithmically quite straightforward and has already been used before by practitioners [1], a detailed analysis was lacking.

In [3], we show for which matrix functions  $f$  and matrices  $A$  the standard deviation of the stochastic probing estimator can be bounded by quantities of the form  $\sqrt{n} \cdot \eta_d$ , where  $\eta_d$  has the same asymptotic behavior as the deterministic case. This significantly improves on the linear scaling with the size of the error in the deterministic case, even if just one stochastic vector is associated to any color. As a by-product of our analysis, we refined classical results on sign patterns in the entries of  $f(A)$ .

Our theoretical findings are illustrated and confirmed by a variety of numerical experiments, where we observed the scaling of the error with the size and compared the performance with other known estimators, indicating when stochastic probing can be the method of choice.

## References

- [1] E. Aune, D. P. Simpson, and J. Eidsvik. Parameter estimation in high dimensional Gaussian distributions. *Stat. Comput.*, 24:247–263, 2014.
- [2] E. N. Epperly, J. A. Tropp, and R. J. Webber. XTrace: making the most of every sample in stochastic trace estimation. *SIAM J. Matrix Anal. Appl.*, 45(1):1–23, 2024.
- [3] A. Frommer, M. Rinelli, and M. Schweitzer. Analysis of stochastic probing methods for estimating the trace of functions of sparse symmetric matrices. *Math. Comp.*, Published online, 2024.
- [4] A. Frommer, C. Schimmel, and M. Schweitzer. Analysis of probing techniques for sparse approximation and trace estimation of decaying matrix functions. *SIAM J. Matrix Anal. Appl.*, 42(3):1290–1318, 2021.
- [5] M. F. Hutchinson. A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines. *Commun. Stat. Simul.*, 19(2):433–450, 1990.
- [6] R. A. Meyer, C. Musco, C. Musco, and D. P. Woodruff. Hutch++: Optimal stochastic trace estimation. In *Symposium on Simplicity in Algorithms (SOSA)*, 142–155. SIAM, 2021.
- [7] J. M. Tang and Y. Saad. A probing method for computing the diagonal of a matrix inverse. *Numer. Linear Algebra Appl.*, 19(3):485–501, 2012.

# Preconditioned Low-Rank Riemannian Optimization for Symmetric Positive Definite Linear Matrix Equations

*Ivan Bioli, Daniel Kressner, Leonardo Robol*

## Abstract

The topic of this work is the solution of multiterm linear matrix equations of the form

$$\mathcal{L}(X) := A_1 X B_1 + \dots + A_\ell X B_\ell = F,$$

under the assumption that the linear operator  $\mathcal{L}$  is positive definite. These equations often appear when discretizing elliptic PDEs, and in control problems. In this setting, it is often the case that  $F$  is either a low-rank matrix, or can be well-approximated in this way, i.e., it has fast decaying singular values. Throughout this work, we assume that  $F$  is a low-rank matrix.

A few special cases are easy to handle, and we can prove that the low-rank structure of  $F$  is inherited by  $X$ . If  $\ell = 1$ ,  $X$  and  $F$  have the same rank; if  $\ell = 2$  we have an elegant theory based on rational approximation that provides upper bounds for the singular values of  $X$  with a weak dependency on the condition number of the operator  $\mathcal{L}$  [2].

When  $\ell > 2$ , most theoretical guarantees are lost: except for a few special cases, none of the arguments used for  $\ell = 2$  can be extended easily. Nevertheless, it can be experimentally verified that the structure is often present in  $X$ ; sometimes, this can be justified via other means (such as the regularity of the solution of the discretized PDE).

Algorithmically, we face similar challenges: the cases  $\ell = 1$  and  $\ell = 2$  are well understood, and efficient low-rank solvers are available (examples include rational and extended Krylov methods, or the Alternative Direction Implicit method, known as ADI). When  $\ell > 2$ , fewer and less appealing options are available, mainly based on Krylov solvers for  $\mathcal{L}$  with low-rank truncation [5]. A major issue with this class of methods is that the intermediate iterates often have much higher ranks than the final solution, and aggressive truncations lead to degraded convergence properties.

We focus on the case  $\ell > 2$ , assuming that  $X$  is the discretization of the solution to an elliptic PDE. Assuming that  $X$  is well-approximated by a matrix of rank  $k$ , the problem can be recast into the optimization problem

$$\text{Find } \min_X \Phi(X), \quad \text{subject to } \text{rank}(X) = k, \quad \Phi(X) := \frac{1}{2} \langle \mathcal{L}(X), X \rangle - \langle F, X \rangle,$$

for some moderate  $k$ , where  $\langle \cdot, \cdot \rangle$  is the standard scalar product inducing the Frobenius norm. The matrices of rank  $k$  have a Riemannian manifold structure, and thus the problem can be conveniently solved using Riemannian optimization [1, 4]. This idea has been successfully exploited in the past to solve positive definite Lyapunov equations, which correspond to  $\ell = 2$  [6]. This strategy restricts our search to matrices of rank  $k$ , and avoids the rank-growth phenomenon often observed in Krylov methods. The convergence theory for Riemannian optimization schemes can be used to prove convergence.

However, a straightforward application of a vanilla Riemannian optimization method presents a major issue: when  $\mathcal{L}$  arises from an elliptic PDE, it is often ill-conditioned, and so is the Hessian of the objective function  $\Phi(X)$ . This leads to poor convergence for first-order optimization schemes.

We discuss how the idea of preconditioning can be extended to this setting, and show that one can design effective modifications to the problem that drastically improve the convergence. The key

ingredient is finding a map  $\mathcal{P}_X$  on the tangent space at  $X$  which is positive definite and approximates the Hessian in a suitable sense. Then, the preconditioning can be described in two similar ways, which are only approximately equivalent:

1. We may take gradients with respect to the inner product induced by  $\mathcal{P}_X$ , instead of the canonical one; this turns the first-order method to be closer to a quasi-Newton scheme.
2. Alternatively, we may use  $\mathcal{P}_X$  to define a non-standard metric and inner product on the Riemannian manifold, and then apply the first-order scheme in this new setup.

We discuss differences, advantages, and disadvantages of these strategies; we show that it is possible to employ both at the same time at once to design preconditioners for a large class of PDE problems discretized with finite element methods.

In the examples that we consider, the multiterm matrix equation is often obtained by non-constant diffusion coefficients in 2D diffusion problems; in this setting it is natural to choose  $\mathcal{P}_X$  as the projection of the operator obtained discretizing the same equation, but with constant diffusion coefficients; this typically is an operator of the same form, but with  $\ell = 2$ . Hence, we consider preconditioners of the form  $\mathcal{P}_X = AXB$ ,  $\mathcal{P}_X = AX + XB$ , and  $\mathcal{P}_X = AXB + CXD$ . Applying these preconditioners in any of the sense described above require to solve an equation on the tangent space, which is low-dimensional. This is relatively easy to do in the first two cases, but it proves challenging in the latter, when  $A, B, C, D \neq I$ . We show how to this effectively by combining the two viewpoints discussed above, and interpreting this preconditioner as a composition of the first two options.

The proposed algorithm is competitive or faster than the state-of-the art in most cases, and in particular when the target rank  $k$  is moderate. When  $k$  is larger, solving the equations on the tangent space can become a bottleneck. For these cases, we propose yet another preconditioner, obtained generalizing the ADI iteration to an iteration on the tangent space of the manifold; our experiments show that it preserves the good convergence properties of the classical ADI scheme for matrix equations, while being much cheaper to compute than the previous approaches. A few steps of this “tangent ADI” iteration will prove to be a very good preconditioner for a large class of problems.

Time permitting, a few extra details essential to produce a robust implementation will be discussed, such as rank-adaptivity, and the use of randomized linear algebra to effectively estimate the residual; the latter helps in detecting possible stagnation due to wrong estimates for the solution rank, and is essential for deciding when to perform rank changes inside the rank-adaptive scheme.

## References

- [1] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2008.
- [2] Bernhard Beckermann and Alex Townsend. On the singular values of matrices with displacement structure. *SIAM Journal on Matrix Analysis and Applications*, 38(4):1227–1248, 2017.
- [3] Ivan Bioli, Daniel Kressner, and Leonardo Robol. Preconditioned Low-Rank Riemannian Optimization for Symmetric Positive Definite Linear Matrix Equations. *arXiv preprint arXiv:2408.16416*, 2024.

- [4] Nicolas Boumal. *An introduction to optimization on smooth manifolds*. Cambridge University Press, 2023.
- [5] Davide Palitta and Patrick Kürschner. On the convergence of Krylov methods with low-rank truncations. *Numerical Algorithms*, 88(3):1383–1417, 2021.
- [6] Bart Vandereycken and Stefan Vandewalle. A Riemannian optimization approach for computing low-rank solutions of Lyapunov equations. *SIAM Journal on Matrix Analysis and Applications*, 31(5):2553–2579, 2010.

Algorithm NCL for constrained optimization:  
Solving the linear systems within interior methods

*Michael Saunders, Ding Ma, Alexis Montoison, Dominique Orban*

Abstract

## 1 Constrained optimization

We consider large smooth constrained optimization problems of the form

NC	$\min_{x \in \Re^n} \phi(x)$ subject to $c(x) = 0, \ell \leq x \leq u,$
----	--

where  $\phi(x)$  is a smooth scalar function and  $c(x) \in \Re^m$  is a vector of smooth linear or nonlinear functions. We assume that first and second derivatives are available. If the constraints include any linear or nonlinear inequalities, we assume that slack variables have already been included as part of  $x$ , and appropriate bounds are included in  $\ell$  and  $u$ . Problem NC is general in this sense.

## 2 LANCELOT

LANCELOT [1, 2, 6] is designed to solve large, smooth constrained optimization problems. For problem NC, LANCELOT solves a sequence of about 10 BCL (Bound-Constrained augmented Lagrangian) subproblems of the form

BC <sub>k</sub>	$\min_{x \in \Re^n} \phi(x) - y_k^T c(x) + \frac{1}{2} \rho_k c(x)^T c(x)$ subject to $\ell \leq x \leq u,$
-----------------	--

where  $y_k$  is an estimate of the dual variables for the nonlinear constraints  $c(x) = 0$ , and  $\rho_k > 0$  is a penalty parameter. After BC<sub>k</sub> is solved (perhaps approximately) to give a subproblem solution  $x_k^*$ , the size of  $\|c(x_k^*)\|$  is used to define BC<sub>k+1</sub>:

- If  $\|c(x_k^*)\|$  is sufficiently small, stop with “Optimal solution found”.
- If  $\|c(x_k^*)\| < \|c(x_{k-1}^*)\|$  sufficiently, update  $y_{k+1} = y_k - \rho_k c(x_k^*)$  and keep  $\rho_{k+1} = \rho_k$ .
- Otherwise, keep  $y_{k+1} = y_k$  and increase the penalty (say  $\rho_{k+1} = 10\rho_k$ ).
- If the penalty is too large (say  $\rho_{k+1} > 10^{10}$ ), stop with “The problem is infeasible”.

## 3 Algorithm NCL

Algorithm NCL [7] mimics LANCELOT with only one change: subproblem BC<sub>k</sub> is replaced by the equivalent larger subproblem

NC <sub>k</sub>	$\min_{x \in \Re^n, r \in \Re^m} \phi(x) + y_k^T r + \frac{1}{2} \rho_k r^T r$ subject to $c(x) + r = 0, \ell \leq x \leq u.$
-----------------	--

Given a subproblem solution  $(x_k^*, r_k^*)$ , the choice between updating  $y_k$  or increasing  $\rho_k$  is based on  $\|r_k^*\|$ . We expect  $\|r_k^*\| \rightarrow 0$ , so that  $x_k^*$  is increasingly close to solving NC.

The active-set solvers CONOPT [3], MINOS [8], and SNOPT [13] are nominally applicable to  $\text{NC}_k$ . Their reduced-gradient algorithms would naturally choose  $r$  as basic variables, and the  $x$  variables would be either superbasic (free to move) or nonbasic (fixed at one of the bounds). However, this is inefficient on large problems unless most bounds are active at the subproblem solution  $x_k^*$ .

In contrast, interior methods welcome the extra variables  $r$  in  $\text{NC}_k$ , as explained in [7]:

- The Jacobian of  $c(x) + r$  always has full row rank. NCL can therefore solve problems whose solution does not satisfy LICQ (the linear independence constraint qualification). It is also applicable to MPEC problems (Mathematical programming problems with equilibrium constraints).
- The sparse-matrix methods used for each iteration of an interior method are affected very little by the increased matrix size.

## 4 The linear system in nonlinear interior methods

For simplicity, we assume that the bounds  $\ell \leq x \leq u$  are simply  $x \geq 0$ . Let  $y$  and  $z$  be dual variables associated with the constraints  $c(x) = 0$  and  $x \geq 0$  respectively, and let  $X = \text{diag}(x)$ ,  $Z = \text{diag}(z)$ . When a nonlinear primal-dual interior method such as IPOPT [4] or KNITRO [5] is applied to  $\text{NC}_k$ , each search direction is obtained from a linear system of the form

$$\begin{pmatrix} -(H + X^{-1}Z) & J^T \\ J & -\rho_k I & I \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta r \\ \Delta y \end{pmatrix} = \begin{pmatrix} r_2 \\ r_3 \\ r_1 \end{pmatrix}. \quad (\text{K3})$$

Although this system large, the additional variables  $\delta r$  do not damage the sparsity of the matrix. IPOPT and KNITRO have performed well on problem  $\text{NC}_k$  as it stands, solving systems (K3).

## 5 Reducing the size of (K3)

For all  $\text{NC}_k$ ,  $\rho_k \geq 1$  (and ultimately  $\rho_k \gg 1$ ), and it is stable to eliminate  $\Delta r$  from (K3) to obtain

$$\begin{pmatrix} -(H + X^{-1}Z) & J^T \\ J & \frac{1}{\rho_k} I \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} = \begin{pmatrix} r_2 \\ r_1 + \frac{r_3}{\rho_k} \end{pmatrix}, \quad \Delta r = \frac{1}{\rho_k}(\Delta y - r_3). \quad (\text{K2})$$

If the original problem is convex,  $H + X^{-1}Z$  is symmetric positive definite (SPD) and it is possible to eliminate  $\Delta y$ :

$$(H + X^{-1}Z + \rho_k J^T J) \Delta x = -r_2 + J^T(r_3 + \rho_k r_1), \quad \Delta y = r_3 + \rho_k(r_1 - J \Delta x). \quad (\text{K1})$$

These reductions would require recoding of IPOPT and KNITRO (which is not likely to happen), but they are practical within the nonlinear interior solver MadNLP [11].

## 6 MadNLP, MadNCL, and GPUs

Algorithm NCL has been implemented as MadNCL [10], using MadNLP [11] as the solver for subproblems  $\text{NC}_k$ . MadNLP has the option of solving (K2) or (K1) rather than (K3).

For convex problems, system (K2) is symmetric quasidefinite (SQD) [14] and it is practical to use sparse indefinite  $\text{LDL}^T$  factorization. MadNLP implements this option using the cuDSS library [12, 9] to utilize GPUs. Alternatively (and again for convex problems), (K1) is SPD and MadNLP can use the cuDSS sparse Cholesky  $\text{LDL}^T$  factorization (unless  $J^T J$  is dense).

Thus, for certain large optimization problems, MadNCL is a solver that employs GPUs and in general is much faster than IPOPT or KNITRO. Numerical results are presented for solving security constrained optimal power flow (SCOPF) problems on GPUs.

## References

- [1] A. R. Conn, N. I. M. Gould, and Ph. L. Toint. *LANCELOT: A Fortran Package for Large-scale Nonlinear Optimization (Release A)*. Lecture Notes in Computational Mathematics 17. Springer Verlag, Berlin, Heidelberg, New York, London, Paris and Tokyo, 1992.
- [2] A. R. Conn, N. I. M. Gould, and Ph. L. Toint. *Trust-Region Methods*. MOS-SIAM Ser. Optim. 1. SIAM, Philadelphia, 2000.
- [3] CONOPT home page. <https://www.gams.com/products/conopt/>.
- [4] IPOPT open source optimization software. <https://projects.coin-or.org/Ipopt>, accessed July 12, 2024.
- [5] KNITRO optimization software. <https://www.artelys.com/solvers/knitro/>, accessed July 12, 2024.
- [6] LANCELOT optimization software. <https://github.com/ralna/LANCELOT>, accessed July 12, 2024.
- [7] D. Ma, K. L. Judd, D. Orban, and M. A. Saunders. Stabilized optimization via an NCL algorithm. In M. Al-Baali, Lucio Grandinetti, and Anton Purnama, editors, *Numerical Analysis and Optimization, NAO-IV, Muscat, Oman, January 2017*, volume 235 of *Springer Proceedings in Mathematics and Statistics*, pages 173–191. Springer International Publishing Switzerland, 2018.
- [8] MINOS sparse nonlinear optimization solver. [https://www.gams.com/latest/docs/S\\_MINOS.html](https://www.gams.com/latest/docs/S_MINOS.html), accessed July 12, 2024.
- [9] A. Montoison, *CUDSS.jl: Julia interface for NVIDIA cuDSS*. <https://github.com/exanauts/CUDSS.jl>.
- [10] A. Montoison, F. Pacaud, M. A. Saunders, S. Shin, and D. Orban. MadNCL: GPU implementation of algorithm NCL. Working paper on Overleaf, 2024.
- [11] A. Montoison, F. Pacaud, S. Shin, et al. MadNLP: a solver for nonlinear programming. <https://github.com/MadNLP/MadNLP.jl>, 2024.

- [12] NVIDIA cuDSS (Preview): A high-performance CUDA Library for Direct Sparse Solvers  
<https://docs.nvidia.com/cuda/cudss/index.html>.
- [13] SNOPT sparse nonlinear optimization solver. <http://ccom.ucsd.edu/~optimizers/solvers/snpt/>, accessed July 12, 2024.
- [14] R. J. Vanderbei. Symmetric quasi-definite matrices. *SIAM J. Optim.*, 5:100–113, 1995.

# Symmetric Pseudospectral Shattering and Fast Divide-and-Conquer for the Definite Generalized Eigenvalue Problem

*James Demmel, Ioana Dumitriu, and Ryan Schneider*

## Abstract

**Overview:** We adapt the asymptotically fastest-known algorithm for diagonalizing arbitrary matrix pencils – as well as the related phenomenon of pseudospectral shattering – to the definite generalized eigenvalue problem. Put simply, we obtain significant efficiency gains by preserving and exploiting structure, in this case symmetry. In doing so, our work provides a general road map for tailoring fast diagonalization to structured problems.

Recent work in randomized numerical linear algebra produced the first sub- $O(n^3)$  algorithms for diagonalizing any matrix  $A$  or matrix pencil  $(A, B)$  [3, 5]. The key insight of this work is the phenomenon of *pseudospectral shattering*, where a random perturbation to a matrix, or pencil, has a regularizing effect on its (pseudo)spectrum. A result of smoothed analysis [11], shattering is characterized by a minimum eigenvalue gap and minimally well-conditioned eigenvectors. Moreover, it implies success for fast divide-and-conquer eigensolvers, which can diagonalize a perturbed matrix/pencil with essentially optimal complexity (that is, complexity equal to matrix multiplication up to log factors). The name “pseudospectral shattering” is derived from the fact that a random grid covering the  $\epsilon$ -pseudospectra of the perturbed problem separates its disjoint components, and the eigenvalues they contain, into separate grid boxes for  $\epsilon$  sufficiently small – i.e., inverse polynomial in  $n$ .

Pseudospectral shattering suggests a simple, high-level approach to eigenvalue problems: apply a random perturbation and run a fast version of divide-and-conquer, where the shattering grid can be used to reliably divide the spectrum at each step. The result is an accurate diagonalization, in the backward-error sense, provided the initial perturbation is small. Prior to [3], which introduced pseudospectral shattering in the context of the standard eigenvalue problem, no way of leveraging divide-and-conquer’s natural parallelization to obtain fast diagonalizations of arbitrary matrices (or pencils) was known. Importantly, [5] established that this approach can be implemented without relying on matrix inversion, thereby promoting stability while also minimizing associated communication costs (following Ballard et al. [2]).

These randomized eigensolvers, which we refer to collectively as pseudospectral divide-and-conquer, are fully general. In particular, both [3] and [5] allow matrices to be arbitrary and apply Ginibre perturbations to obtain a guarantee of pseudospectral shattering. This begs the question: how can we adapt these algorithms to better handle symmetric or sparse inputs, for which dense Gaussian perturbations are structure-destroying? Going further: if we can achieve pseudospectral shattering while maintaining structure – i.e., via structured perturbations – how can we translate that into efficiency gains in divide-and-conquer?

We answer these questions for the definite generalized eigenvalue problem, which corresponds to pencils  $(A, B)$  in which  $A$  and  $B$  are Hermitian and the *Crawford number*  $\gamma(A, B)$  satisfies

$$\gamma(A, B) = \min_{\|x\|_2=1} |x^H(A + iB)x| > 0. \quad (1)$$

Pencils arising in scientific computing and machine learning are often definite [7, 6]. We note two important sub-problems in particular: (1) the Hermitian eigenvalue problem, corresponding

to  $B = I$ , and (2) the generalized symmetric definite eigenvalue problem, in which  $B$  is positive definite.

As inputs to divide-and-conquer, definite pencils exhibit a number of properties that can be leveraged for improved efficiency. Most notably, the eigenvalues of a definite pencil  $(A, B)$  are real (and in fact any  $\epsilon$ -pseudospectrum that considers only Hermitian perturbations to  $A$  and  $B$  will be constrained to the real axis for  $\epsilon$  sufficiently small). Additionally, definite pencils are regular and satisfy stronger eigenvalue/eigenvector perturbation bounds than the generic case (see e.g., [12]). Finally, where an arbitrary pencil  $(A, B)$  is diagonalized by a pair of eigenvector matrices – if it is diagonalizable at all – a definite pencil can always be diagonalized by a single matrix. That is, for any definite pencil  $(A, B)$  there exists invertible  $X$  such that

$$(X^H AX, X^H BX) = (\Lambda_A, \Lambda_B) \quad (2)$$

for  $\Lambda_A$  and  $\Lambda_B$  diagonal.

Motivated by these observations, we devise a version of pseudospectral divide-and-conquer that pursues efficiency by maintaining definiteness through both the initial random perturbation and the subsequent recursive procedure. The main ingredients are the following:

1. We prove shattering for a symmetric pseudospectrum

$$\Lambda_\epsilon^{\text{sym}}(A, B) = \left\{ z : \begin{array}{l} (A + E)u = z(B + F)u \text{ for } u \neq 0 \text{ and} \\ E, F \text{ Hermitian with } \sqrt{\|E\|_2^2 + \|F\|_2^2} \leq \epsilon \end{array} \right\} \quad (3)$$

under random perturbations that are either diagonal or sampled from the Gaussian Unitary Ensemble (GUE). The diagonal case builds on work of Minami [8] and implies a remarkably simple path to structured shattering for (Hermitian) banded matrices. The GUE case, meanwhile, leverages recent results of Aizenman et al. [1]. In both settings, the key insight is a bound on the probability that a perturbed Hermitian matrix has a certain number of eigenvalues in a given interval of the real axis.

2. Next, we demonstrate that (inverse-free) iterative methods for computing spectral projectors of  $(A, B)$  – i.e., projectors onto deflating subspaces corresponding to sets of eigenvalues, which are the key to the recursive splits of divide-and-conquer – can be optimized for fast convergence on problems with real eigenvalues [4]. This is the primary advantage we gain access to by preserving definiteness (and itself generalizes work of Nakatsukasa et al. [10]).

Combining points one and two yields a specialized version of pseudospectral divide-and-conquer that is significantly more efficient on definite inputs. Ongoing work seeks high performance implementations of both standard pseudospectral divide-and-conquer and this specialization. Accordingly, and in parallel with broader efforts to deploy randomized algorithms in numerical linear algebra [9], our work represents an important step towards bringing fast, randomized diagonalization to practice.

## References

- [1] M. Aizenman, R. Peled, J. Schenker, M. Shamis, and S. Sodin. Matrix regularizing effects of Gaussian perturbations. *Communications in Contemporary Mathematics*, 19(03):1750028, 2017.

- [2] G. Ballard, J. Demmel, O. Holtz, and O. Schwartz. Minimizing Communication in Numerical Linear Algebra. *SIAM Journal on Matrix Analysis and Applications*, 32(3):866–901, 2011.
- [3] J. Banks, J. Garza-Vargas, A. Kulkarni, and N. Srivastava. Pseudospectral Shattering, the Sign Function, and Diagonalization in Nearly Matrix Multiplication Time. *Foundations of Computational Mathematics*, 23:1959–2047, 2023.
- [4] J. Demmel, I. Dumitriu, and R. Schneider. Fast and Inverse-Free Algorithms for Deflating Subspaces. arXiv:2310.00193, 2024.
- [5] J. Demmel, I. Dumitriu, and R. Schneider. Generalized Pseudospectral Shattering and Inverse-Free Matrix Pencil Diagonalization. *Foundations of Computational Mathematics*, 2024.
- [6] B. Ford and G. Hall. The generalized eigenvalue problem in quantum chemistry. *Computer Physics Communications*, 8(5):337–348, 1974.
- [7] O. Mangasarian and E. Wild. Multisurface proximal support vector machine classification via generalized eigenvalues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1):69–74, 2006.
- [8] N. Minami. Local fluctuation of the spectrum of a multidimensional Anderson tight binding model. *Communications in Mathematical Physics*, 177(3):709–725, 1996.
- [9] R. Murray, J. Demmel, M. W. Mahoney, N. B. Erichson, M. Melnichenko, O. A. Malik, L. Grigori, P. Luszczek, M. Derezhinski, M. E. Lopes, T. Liang, H. Luo, and J. Dongarra. Randomized Numerical Linear Algebra: A Perspective on the Field with an Eye to Software. Technical Report UCB/EECS-2023-19, EECS Department, University of California, Berkeley, Feb 2023.
- [10] Y. Nakatsukasa, Z. Bai, and F. Gygi. Optimizing Halley’s Iteration for Computing the Matrix Polar Decomposition. *SIAM Journal on Matrix Analysis and Applications*, 31(5):2700–2720, 2010.
- [11] D. A. Spielman and S.-H. Teng. Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time. *Journal of the ACM (JACM)*, 51(3):385–463, 2004.
- [12] G. W. Stewart. Perturbation bounds for the definite generalized eigenvalue problem. *Linear Algebra and its Applications*, 23:69–85, 1979.

# Sparse Pseudospectral Shattering

*Rikhav Shah, Edward Zeng, Nikhil Srivastava*

## Abstract

A central question in numerical analysis is the following: how do the eigenvalues and eigenvectors of a matrix behave under perturbations of its entries? For Hermitian matrices, the eigenvalues are 1–Lipschitz functions of the entries, and the eigenvectors are stable under perturbations if the minimum eigenvalue gap is large. This fact is essential to the rapid convergence and rigorous analysis of algorithms for the Hermitian eigenvalue problem and its cousins.

For non-Hermitian matrices, two related difficulties appear: *non-orthogonality* of the eigenvectors and *spectral instability*, i.e. high sensitivity of the eigenvalues to perturbations of the matrix entries. Non-orthogonality slows down the convergence of iterative algorithms (such as the power method) and spectral instability makes it difficult to rigorously reason about convergence in the presence of roundoff error. The main tool used to surmount these difficulties in recent years is smoothed analysis, i.e., adding a small random perturbation to the input and solving the perturbed problem, incurring a small backward error. Specifically it was shown in [BGVKS23] that adding small i.i.d. complex Gaussian random variables to each entry of a matrix produces a matrix with well-conditioned eigenvectors and a large eigenvalue gap, a phenomenon termed “pseudospectral shattering”. This was then generalized to other random variables in [BVKS20, JSS20], and is currently an essential mechanism in all of the known convergence results about diagonalizing arbitrary dense matrices. Crucially, however, all current work examines the setting where i.i.d. noise is added to every single entry of a given matrix.

This paper asks if it possible to achieve pseudospectral shattering by adding noise to only a subset of entries, selected at random. We provide a positive answer.

In fact, we show only  $O(n \log^2(n))$  entries need to be perturbed to achieve sufficient regularization for many downstream algorithmic tasks. Our results are phrased in terms of the sparsity  $\rho = \rho(n)$  of the added noise. In our model, each entry of a given matrix  $M$  is perturbed by a complex Gaussian  $g$  with probability  $\rho$ , and left unchanged otherwise. As one might expect, our guarantee provides stronger regularization the larger  $\rho$  is. We measure regularization in terms of the *eigenvector condition number*  $\kappa_V(A)$  and *minimum eigenvalue gap*  $\eta(A)$ . In the following definitions of these quantities,  $A = VDV^{-1}$  is any diagonalization of  $A$  and  $\lambda_1(A), \dots, \lambda_n(A)$  are the eigenvalues of  $A$ .

$$\kappa_V(A) = \inf_{A=VDV^{-1}} \|V^{-1}\| \|V\| \quad \text{and} \quad \eta(A) = \min_{i \neq j} |\lambda_i(A) - \lambda_j(A)|.$$

Given a matrix  $M$ , the perturbation described above has the form  $M + N_g$ , where the entries of  $N_g$  are i.i.d. copies of  $\delta \cdot g$  where  $\delta \sim \text{Bernoulli}(\rho)$  and  $g \sim \mathcal{N}(0, 1_{\mathbb{C}})$ . Our main theorem is as follows.

**Theorem 1.** *Set  $K = 2 \log(n) / \log(n\rho)$ . For any  $M \in \mathbb{C}^{n \times n}$ , if  $n\rho = \Omega(\log(n) \log(\|M\| + n))$  then*

$$\Pr(\kappa_V(M + N_g) \geq (\|M\| + n^2\rho)^{10K}) \leq O(n^{-K}), \tag{1}$$

and

$$\Pr(\eta(M + N_g) \leq (\|M\| + n^2\rho)^{-35K}) \leq O(n^{-K}). \quad (2)$$

The proof of this theorem consists of three steps. In the first two steps,  $A$  can be any random matrix. In the third step, we need our particular model of sparse perturbations,  $A = M + N_g$ .

**1. Bootstrapping:** An important object related to spectral stability is the  $\varepsilon$ -pseudospectrum of  $A$ , defined as

$$\Lambda_\varepsilon(A) = \{z \in \mathbb{C} : \sigma_n(z - A) \leq \varepsilon\}.$$

It always contains disks of radius  $\varepsilon$  around each eigenvalue; equality is achieved if and only if  $A$  is a normal matrix, i.e.  $\kappa_V(A) = 1$ . For less well conditioned matrices, the  $\varepsilon$ -pseudospectrum will be larger. A quantifiable version of this statement relates the area of the pseudospectrum to both  $\kappa_V(\cdot)$  and  $\eta(\cdot)$ . The bootstrapping argument of [JSS20] turns this observation into a probabilistic tail bound: a strong upper bound on  $\mathbb{E} \text{vol } \Lambda_\varepsilon(A)$  and a lower tail bound on  $\eta(A)$  establishes an upper tail bound on  $\kappa_V(A)$ . We adapt their argument and improve it by dramatically lessening the control on  $\mathbb{E} \text{vol } \Lambda_\varepsilon(A)$  required for the argument to go through. The ideal control would be of the form

$$\mathbb{E} \text{vol } \Lambda_\varepsilon(A) \leq \text{poly}(n) \cdot \varepsilon^2.$$

The bootstrapping argument of [JSS20] shows that it suffices to have  $\varepsilon^2 \log(1/\varepsilon)$  in place of  $\varepsilon^2$ . We show it suffices to have  $\varepsilon^c + \exp(-n)$  for any constant  $c > 0$  in place of  $\varepsilon^2$ .

**2. Reduction to bottom two singular values:** This step relies on known arguments which were also used in [BKMS21, BGVKS23]. The previous step shows we need control over  $\mathbb{E} \text{vol } \Lambda_\varepsilon(A)$  and  $\eta(A)$ . As may be clear from the definition,  $\mathbb{E} \text{vol } \Lambda_\varepsilon(A)$  is immediately convertible to lower tail estimates for the least singular value  $\sigma_n(z - A)$  for  $z \in \mathbb{C}$ . We also show a lower tail bound on  $\eta(A)$  can be reduced to lower tail bounds on the *bottom two* singular values  $\sigma_n(z - A), \sigma_{n-1}(z - A)$ .

The strength of the tail bound can be characterized in terms of the *power*  $c_m$  of  $\varepsilon$  on the right-hand side of a bound of the form

$$\Pr(\sigma_{n-m}(A) \leq \varepsilon) \leq \text{poly}(n) \varepsilon^{c_m} + \exp(-n). \quad (3)$$

The reduction from  $\eta(A)$  described in Lemma ?? goes through when

$$\frac{1}{c_0} + \frac{1}{c_1} < 1.$$

For sufficient control on  $\mathbb{E} \text{vol } \Lambda_\varepsilon(A)$ , we just need  $c_0 > 0$ . Thus the bottleneck is the reduction from  $\eta(A)$ .

**3. Control on bottom two singular values:** We show the required control over the bottom two singular values holds with room to spare. Specifically, we show a bound of the form (3) holds for  $c_0 = 2$  and  $c_1 = 4$  (in fact, we show it holds for  $c_m = 2m + 2$  for any constant  $m$ ). The argument is based on an  $\varepsilon$ -net construction following the compressible/incompressible or rich/poor

decomposition in [TV07]. That work considers lower tail bounds of  $\sigma_n(M + N_x)$  where  $x$  is a general sub-Gaussian random variable and the sparsity parameter  $\rho(n) = n^{\alpha-1}$ ,  $\alpha > 0$  is a fixed polynomial in  $n$ . They show for every polynomial  $n^A$ , there exists a polynomial  $n^B$  such that

$$\Pr(\sigma_n(A) \leq n^{-A}) \leq n^{-B}.$$

By tracing their argument, one can show there is a linear relationship between  $A$  and  $B$  so that their bound more closely resembles the form (3) for an unspecified tiny constant  $c_0$  and  $\varepsilon = \frac{1}{\text{poly}(n)}$ .

By our improvement to the bootstrapping argument, their result gives enough control over  $\mathbb{E} \text{vol } \Lambda_\varepsilon(A)$ . However, it is not enough for  $\eta(A)$ . We specialize to the complex Gaussian case  $x = g$  (or really the case of  $x$  having bounded density on  $\mathbb{C}$ ) and achieve three advantages over [TV07] in this setting. Firstly, our argument applies to every  $m$  (not just  $m = 0$ ), and we show the optimal power of  $c_0 = 2$  in the  $m = 0$  case. Secondly, we may take  $\varepsilon$  to be arbitrarily small. Lastly, we are able to push the sparsity parameter down to  $(\log n)^2/n$ .

Furthermore, because  $g$  has a continuous density, we avoid the additive combinatorics required by [TV07], resulting in much simpler proofs.

**Algorithmic application.** As alluded to already, establishing control over the eigenvector condition number of a matrix is essential for rigorous analysis of non-Hermitian eigenvalue problems. The work of [BKMS21] does this by adding a dense perturbation  $N$ . The drawback is that the cost of computing matrix-vector products goes from  $O(\text{nnz}(M))$  to  $O(\text{nnz}(M) + \text{nnz}(N)) = O(n^2)$  where  $\text{nnz}(A)$  is the number of nonzero entries in the matrix  $A$ . The algorithmic content of this paper is that it suffices to take  $N$  to be a sparse perturbation, with  $\mathbb{E} \text{nnz}(N) = n^2\rho$  for  $\rho = \Omega(\log(n)^2/n)$ .

As a simple example of an application of Theorem 1, we show it implies an algorithm for computing the spectral radius  $spr(M)$  of any matrix up to mixed forwards-backwards error  $\varepsilon$  using just

$$O\left(\frac{\log(n)}{\log(np)} \cdot \frac{\log(n/\varepsilon)}{\varepsilon} \cdot (\text{nnz}(M) + n^2\rho)\right)$$

floating point operations.

A full preprint can be found at

<https://math.berkeley.edu/~rdshah/files/sparsepseudospectralshattering.pdf>

## References

- [BGVKS23] Jess Banks, Jorge Garza-Vargas, Archit Kulkarni, and Nikhil Srivastava. Pseudospectral shattering, the sign function, and diagonalization in nearly matrix multiplication time. *Foundations of computational mathematics*, 23(6):1959–2047, 2023.
- [BVKS20] Jess Banks, Jorge Garza Vargas, Archit Kulkarni, and Nikhil Srivastava. Overlaps, eigenvalue gaps, and pseudospectrum under real ginibre and absolutely continuous perturbations, 2020.

- [JSS20] Vishesh Jain, Ashwin Sah, and Mehtaab Sawhney. On the Real Davies’ Conjecture, 2020.
- [BKMS21] Jess Banks, Archit Kulkarni, Satyaki Mukherjee, and Nikhil Srivastava. Gaussian regularization of the pseudospectrum and davies’ conjecture. *Communications on Pure and Applied Mathematics*, 74(10):2114–2131, 2021.
- [TV07] Terence Tao and Van Vu. Random matrices: The circular law. *Communications in Contemporary Mathematics*, pages 261–307, 2007.
- [BR17] Anirban Basak and Mark Rudelson. Invertibility of sparse non-hermitian matrices. *Advances in Mathematics*, 320:426–483, 2017.

# Randomized small-block Lanczos for null space computations

*Daniel Kressner and Nian Shao*

## Abstract

Computing the null space of a large matrix  $A$ , having no particular structure beyond being sparse, is a challenging computational problem, especially if the nullity  $N$ —the dimension of the null space—is large.

When  $N = 1$ , standard choices include the Lanczos method for computing the smallest eigenvalue and eigenvector of  $A^\top A$ . When  $N > 1$ , the Lanczos method (with a single starting vector) becomes unreliable and prone to miss components of the null space. In fact, in exact arithmetic only a one-dimensional subspace of the null space can be extracted via Lanczos. Block Lanczos methods address this issue by utilizing a block of  $d > 1$  starting vectors instead of a single starting vector. Common wisdom and existing convergence analyses all suggest to choose  $d$  not smaller than the size of the eigenvalue cluster of interest. Applied to null space computation this would require  $d \geq N$  and in exact arithmetic this condition is indeed necessary. The presence of round-off error blurs the situation. As an example, consider the  $420 \times 420$  diagonal matrix

$$A = \text{diag}(0, \dots, 0, 1, 2, \dots, 399), \quad (1)$$

that is, the first 21 diagonal entries are zero. When applying block Lanczos to  $A$  in order to compute its null space and choosing a block of  $d$  (Gaussian) random starting vectors, one would expect to obtain no more than  $d$  (approximate) zero eigenvalues and corresponding eigenvectors. However, when executing block Lanczos in double precision arithmetic without breakdown and declaring eigenvalues less than  $10^{-4}$  as zero, we obtain the results reported in Table 1.

Table 1: Null space dimension obtained when applying block Lanczos with  $d$  random starting vectors to the matrix from (1).

Block size $d$	1	2	3	4	5	6	7	10	12	14	15	20	21
Dimension	11	11	16	12	15	15	21	20	21	21	21	21	21

While the null space dimension estimated by block Lanczos for small values of  $d$  is erratic and smaller than 21, it is always larger than  $d$ . In fact, already for  $d = 12$ , the correct null space dimension is detected. Experiments with other matrices lead to similar findings, suggesting that round-off error breaks some of the curse incurred by eigenvalue clusters, but it does not fully address the convergence issues of single-vector or small-block Lanczos methods either.

The situation described above is reminiscent of recent work [MMM2024] on the single-vector Lanczos method for low-rank approximation. Taking the randomness of the starting vector into account they establish a convergence result that still requires the (large) singular values to be distinct but the dependence of the complexity on the gaps between singular values is very mild, in fact logarithmic. Although single-vector or small-block Krylov subspace methods do not satisfy gap-independent convergence bounds, small random perturbations of  $A$  can easily break the adverse role of repeated singular values. As we have seen, perturbations due to round-off are not sufficient to achieve this effect, but (slightly larger) random perturbations will do, thanks to eigenvalue repulsion results.

In this work [KS2024], we propose a randomized small-block Lanczos method for null space calculation, sketched as Algorithm 1. Compared to the task of low-rank approximation, there are

two major differences. First, in low-rank approximation, the largest few singular values are usually unknown and, in most applications, they do not form tight large clusters. In the context of null spaces, the desired singular values form one large cluster of zeros. Second, while convergence to the relevant invariant subspace is not necessary for low-rank approximation, such convergence is imperative to obtain a good null space approximation.

---

**Algorithm 1:** Randomized small-block Lanczos for null space computations

---

**Input:** Matrix  $A \in \mathbb{R}^{m \times n}$  with  $m \geq n$  and  $N$  zero singular values. Block size  $d$ . Perturbation parameter  $\epsilon > 0$ . Number of iterations  $\ell$ .

**Output:** Orthonormal basis  $V$  approximating null space of  $A$ .

Set  $B = A^\top A + \epsilon D$ , with a diagonal matrix  $D$  having diagonal entries uniformly i.i.d. in  $[0, 1]$ ;

Draw a Gaussian random matrix  $\Omega \in \mathbb{R}^{n \times d}$ ;

Perform block Lanczos to compute the decomposition  $BZ_\ell = Z_\ell T_\ell + Q_{\ell+1} E_{\ell+1}$ , where  $Z_\ell$  is an orthonormal basis of Krylov subspace  $\text{span}\{\Omega, B\Omega, \dots, B^{\ell-1}\Omega\}$  and  $T_\ell$  is block tridiagonal;

Compute an orthonormal basis  $V_Z$  for the invariant space of the  $N$  smallest eigenvalues of  $T_\ell$ .

**return**  $V = Z_\ell V_Z$ ;

---

In the theoretical part, we examine the impact of the random perturbation  $\epsilon D$ . In particular, we present a new eigenvalue repulsion result for the perturbed zero eigenvalues of  $A^\top A$ . At the same time, the perturbation incurs a limit on the attainable accuracy of the null space approximation, and we quantify this effect by a perturbation analysis. For the single-vector case ( $d = 1$ ), we achieve more refined results by analyzing the convergence of individual Ritz vectors and establish sharp bounds on the accuracy of the null space approximation.

In the numerical part, we incorporate several practical improvements, including preconditioning, restarting, and partial reorthogonalization, and provide various numerical results for applications such as the computation of connected graph components and cohomology. These numerical experiments not only validate our theoretical findings regarding the randomized single-vector Lanczos method but also highlight the efficiency of the small-block Lanczos method. Notably, it highlights when our method is preferable compared to other null space solvers, particularly when the nullity is substantial and memory constraints are a limiting factor.

## References

- [KS2024] Daniel Kressner and Nian Shao. “A randomized small-block Lanczos method for large-scale null space computations.” arXiv preprint: 2407.04634.
- [MMM2024] Meyer, Raphael, Cameron Musco, and Christopher Musco. “On the unreasonable effectiveness of single vector krylov methods for low-rank approximation.” Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA).

# DATA-PARALLEL adaptive TENSOR-TRAIN CROSS approximation

Tianyi Shi, Daniel Hayes, Jing-Mei Qiu

## Abstract

The tensor-train (TT) format is a low rank tensor representation frequently used for high order tensors. Traditionally, the TT format is computed directly with all the elements in the tensor. In this talk, we propose a TT decomposition algorithm that partitions the tensor into subtensors and performs decomposition individually before merging back together. This factorization routine is ideal for distributed memory parallelism. In addition, instead of computing the TT format with singular value decomposition based techniques, our proposed method, parallel adaptive TT cross, is a data-centric iterative method based on data skeletonization and has a low computational cost. In particular, our method is based on two innovative iterative formulations for data extraction and TT format construction, and we provide theoretical guarantees, communication analysis, and scaling results. For example, strong scaling results on synthetic datasets and discretized solutions of 2D and 3D Maxwellian equations suggest that this algorithm scales well with the number of computing cores, with respect to both storage and timing. This talk is based on the preprint <https://arxiv.org/abs/2407.11290>

BIO: Tianyi Shi is a postdoctoral fellow in the Scalable Solvers Group in Applied Mathematics and Computational Research Division at Lawrence Berkeley National Laboratory. He obtained his PhD from the Center for Applied Mathematics at Cornell University. His research interests include numerical linear algebra with a focus on sparse and data-sparse matrices and tensors, and high-performance computing.

# Estimation of Spectral Gaps for Sparse Symmetric Matrices

*Michele Benzi, Michele Rinelli, Igor Simunec*

## Abstract

Identifying the gap between two consecutive eigenvalues of a real symmetric matrix  $A \in \mathbb{R}^{n \times n}$  is an important task that is often encountered in applications, such as in electronic structure computations. For instance, in Kohn-Sham Density Functional Theory [5] one has to determine the spectral projector  $P_\mu$  associated with the *Fermi level* (or *chemical potential*)  $\mu \in \mathbb{R}$ , which corresponds to the occupied states of a system described by a discrete Hamiltonian  $A$ . In the case of insulators at zero electronic temperature, there is a gap separating the first  $k$  eigenvalues of  $A$  from the rest of the spectrum, and the Fermi level  $\mu$  lies inside the gap between the  $k$ -th and  $(k+1)$ -th eigenvalue of  $A$ , where  $k$  is the number of electrons in the system. In this setting, the spectral projector  $P_\mu = h_\mu(A)$  is often approximated using polynomials or rational functions that approximate the step function  $h_\mu(\lambda)$ , which takes the value 1 for  $\lambda < \mu$  and 0 for  $\lambda > \mu$ . In order to use this approach, one first needs to compute a value of  $\mu$  that lies in the gap between  $\lambda_k$  and  $\lambda_{k+1}$ , where we denote the eigenvalues of  $A$  as  $\lambda_1, \dots, \lambda_n$  in nondecreasing order.

Instead of looking for the gap between  $\lambda_k$  and  $\lambda_{k+1}$  for a specific  $k$ , in this talk we tackle this problem from a different perspective, and aim to find all gaps in the spectrum of  $A$  that are larger than a certain threshold. Since in practical applications the gap between  $\lambda_k$  and  $\lambda_{k+1}$  is often relatively large, we expect that this approach will provide useful results even when one needs to find a single, specific gap.

Let us denote by  $n_e(\mu)$  the number of eigenvalues of  $A$  that are strictly smaller than  $\mu$ . Assuming that  $\mu$  does not coincide with an eigenvalue of  $A$ , we have  $n_e(\mu) = \text{rank}(P_\mu) = \text{tr}(P_\mu)$ . Therefore,  $n_e(\mu)$  can be estimated by estimating  $\text{tr}(P_\mu)$  with Hutchinson's stochastic trace estimator [4] combined with the Lanczos algorithm [7, Algorithm 6.15]. Given  $s$  random Gaussian vectors  $\{\mathbf{x}_i\}_{i=1}^s$ , Hutchinson's stochastic trace estimator approximates  $\text{tr}(P_\mu)$  with

$$\text{tr}_s^H(P_\mu) := \frac{1}{s} \sum_{i=1}^s \mathbf{x}_i^T P_\mu \mathbf{x}_i.$$

Since  $P_\mu = h_\mu(A)$ , each quadratic form can be approximated using the Lanczos algorithm. Let  $V_m^{(i)} \in \mathbb{R}^{n \times m}$  be an orthonormal basis of the Krylov subspace

$$\mathcal{K}_m(A, \mathbf{x}_i) = \text{span}\{\mathbf{x}_i, A\mathbf{x}_i, \dots, A^{m-1}\mathbf{x}_i\},$$

constructed with the Lanczos algorithm, and let the tridiagonal matrix  $T_m^{(i)} := V_m^{(i)T} A V_m^{(i)}$  be the projection of  $A$  onto  $\mathcal{K}_m(A, \mathbf{x}_i)$ . We can approximate  $\mathbf{x}_i^T P_\mu \mathbf{x}_i$  with

$$\psi_m^{(i)}(\mu) := \|\mathbf{x}_i\|_2^2 \mathbf{e}_1^T h_\mu(T_m^{(i)}) \mathbf{e}_1, \quad j = 1, \dots, N_f,$$

so we obtain the trace approximations

$$\text{tr}(P_\mu) \approx \frac{1}{s} \sum_{i=1}^s \psi_m^{(i)}(\mu).$$

If we use the same vectors  $\{\mathbf{x}_i\}_{i=1}^s$  for different  $\mu$ , these trace approximations can be computed simultaneously for many different  $\mu$  by using the same Krylov subspaces  $\mathcal{K}_m(A, \mathbf{x}_i)$ , with a cost that

is only slightly higher than for a single value of  $\mu$ . This approach has already been used in literature on related problems, such as the estimation of the number of eigenvalues of  $A$  in an interval or the estimation of spectral densities [3, 6]. In this talk we will focus on thoroughly analyzing this method for the detection of spectral gaps, with the goal of determining how to choose the parameters  $s$  and  $m$  in order to minimize the computational cost and ensure that all gaps with relative width above a given threshold  $\theta \in (0, 1)$  are found (up to a failure probability  $\delta$ ).

The error of Hutchinson's estimator can be bounded, for instance, with [2, Theorem 1], which states that

$$\mathbb{P}(|\text{tr}(P_\mu) - \text{tr}_s^H(P_\mu)| \geq \varepsilon) \leq \delta \quad \text{if} \quad s \geq \frac{4}{\varepsilon^2} (\|P_\mu\|_F^2 + \varepsilon \|P_\mu\|_2) \log(2/\delta).$$

However, we have  $\|P_\mu\|_F^2 = n_e(\mu)$ , and  $n_e(\mu) = O(n)$  when  $\mu$  is near the middle of the spectrum, so to achieve any fixed absolute accuracy  $\varepsilon$  we would have to take  $s = O(n)$ , which becomes unfeasible as  $n$  grows. This means that with this approach it is prohibitively expensive to try and find a value of  $\mu$  in the gap between  $\lambda_k$  and  $\lambda_{k+1}$  by requiring that  $\text{tr}_s^H(P_\mu) \approx n_e(\mu) = k$ . Instead, we use a different point of view.

If we consider  $\text{tr}(P_\mu)$  as a function of  $\mu$ , it is a nondecreasing and piecewise constant function, with a jump of height 1 whenever  $\mu$  coincides with an eigenvalue of  $A$ . A similar property holds for  $\text{tr}_s^H(P_\mu)$ , with the difference that the jumps have random heights, each with expected value 1. In particular,  $\text{tr}_s^H(P_\mu)$  is constant for all  $\mu \in [a, b]$  if the interval  $[a, b]$  contains no eigenvalues of  $A$ . This observation can be exploited to find gaps in the spectrum of  $A$ , by looking for intervals in which the Lanczos approximation  $\frac{1}{s} \sum_{i=1}^s \psi_m^{(i)}(\mu)$  is almost constant in  $\mu$  and has a small error. For instance, if for a constant  $C$  and a small  $\varepsilon > 0$  we can show that

$$\text{tr}_s^H(P_\mu) \in [C - \varepsilon, C + \varepsilon] \quad \text{for all } \mu \in [a, b],$$

then we can conclude that either  $\text{tr}_s^H(P_\mu)$  is constant in the interval  $[a, b]$  and hence  $[a, b]$  is a gap in the spectrum of  $A$ , or all jumps in  $\text{tr}_s^H(P_\mu)$  associated with eigenvalues in  $[a, b]$  have heights smaller than  $2\varepsilon$ . If  $\varepsilon$  is small enough, the latter event has a small chance of occurring.

In order to make the argument outlined above rigorous, we obtain a bound on the probability of having small jumps in  $\text{tr}_s^H(P_\mu)$ , as well as a posteriori error bounds and estimates for the Lanczos approximation of the quadratic forms  $\mathbf{x}_i^T P_\mu \mathbf{x}_i$ . By combining these bounds, we will show that for a given budget of matrix-vector products with  $A$ , the best way to allocate it is to set  $s = 1$ , i.e., use a single random vector for Hutchinson's estimator and concentrate all the computational effort on the Lanczos algorithm. We also obtain an a priori bound on the accuracy of the Lanczos approximation that depends on the relative gap width  $\theta$ , which will allow us to predict how many Lanczos iterations are needed to ensure that all gaps larger than a given width are detected.

The theoretical analysis is complemented by a detailed computational discussion, leading to an algorithm that is able to detect gaps efficiently and reliably. The effectiveness of the proposed method will be showcased with several numerical examples. Further details can be found in the preprint [1].

## References

- [1] M. Benzi, M. Rinelli, and I. Simunec. Estimation of spectral gaps for sparse symmetric matrices, arXiv:2410.15349, 2024.

- [2] A. Cortinovis and D. Kressner. On randomized trace estimates for indefinite matrices with an application to determinants. *Found. Comput. Math.*, 22(3):875–903, 2022.
- [3] E. Di Napoli, E. Polizzi, and Y. Saad. Efficient estimation of eigenvalue counts in an interval. *Numer. Linear Algebra Appl.*, 23(4):674–692, 2016.
- [4] M. F. Hutchinson. A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines. *Comm. Statist. Simulation Comput.*, 18(3):1059–1076, 1989.
- [5] L. Lin, J. Lu, and L. Ying. Numerical methods for Kohn-Sham density functional theory. *Acta Numer.*, 28:405–539, 2019.
- [6] L. Lin, Y. Saad, and C. Yang. Approximating spectral densities of large matrices. *SIAM Rev.*, 58(1):34–65, 2016.
- [7] Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, Philadelphia, PA, second edition, 2003.

# Alternating Mahalanobis Distance Minimization for CP Tensor Decomposition

Navjot Singh, Edgar Solomonik

## Abstract

Tensors generalize matrices by representing data in more than two dimensions. Tensor decompositions are mathematical constructs used to efficiently represent, approximate, and manipulate tensors. Tensor decompositions have applications in various fields such as in image analysis [2], in signal processing [6], in quantum chemistry [5], in chemometrics [4] and many more. Finding the most accurate low rank tensor decomposition of a tensor is an NP-hard problem [1] in most cases. Consequently, numerical optimization algorithms are used to compute a low rank approximation efficiently. In this talk, we present a novel alternating optimization algorithm for CP tensor decomposition (CPD) [7].

**CP Decomposition.** The CPD problem, for an order 3 tensor  $\mathcal{T}$  is formulated as following,

$$\min_{\mathbf{A}, \mathbf{B}, \mathbf{C}} \frac{1}{2} \left\| \mathcal{T} - [\mathbf{A}, \mathbf{B}, \mathbf{C}] \right\|_F^2,$$

$$([\mathbf{A}, \mathbf{B}, \mathbf{C}])_{ijk} = \sum_r a_{ir} b_{jr} c_{kr}.$$

The most used algorithm to solve the above problem is alternating least squares (ALS). ALS solves for one factor matrix at a time which results in least squares equation. Solving for factor matrix  $\mathbf{A}$  results in the following least squares equation,

$$\mathbf{A}(\mathbf{C} \odot \mathbf{B})^T \cong \mathbf{T}_{(1)},$$

where  $\odot$  denotes the Khatri-Rao product. ALS solves these equations via normal equations where the solution is given as

$$\mathbf{A} = \mathbf{T}_{(1)} (\mathbf{C} \odot \mathbf{B})^{\dagger T} = \mathbf{T}_{(1)} (\mathbf{C} \odot \mathbf{B}) (\mathbf{B}^T \mathbf{B} * \mathbf{C}^T \mathbf{C})^\dagger,$$

where  $*$  denotes the Hadamard product, and  $\dagger$  denotes the Moore-Penrose inverse. The normal equations result in a solution that is optimal in Frobenius norm and matrix 2-norm. We propose an update that solves the least squares equations for factor  $\mathbf{A}$  as

$$\mathbf{A} = \mathbf{T}_{(1)} (\mathbf{C}^{\dagger T} \odot \mathbf{B}^{\dagger T}).$$

We prove that the above update leads to an alternating minimization algorithm which has a **local superlinear convergence rate** for exact CP rank problems, when rank is smaller than the dimensions. The above update is an optimal solution of the least squares equations in Mahalanobis norm. The algorithm corresponding to these alternating updates is called alternating Mahalanobis distance minimization (AMDM) [7]. The update for factor  $\mathbf{A}$  is derived by minimizing the following objective function

$$\min_{\mathbf{A}} \frac{1}{2} \text{vec}(\mathcal{T} - [\mathbf{A}, \mathbf{B}, \mathbf{C}])^T M \text{vec}(\mathcal{T} - [\mathbf{A}, \mathbf{B}, \mathbf{C}]),$$

where  $\mathbf{M}$  is a Kronecker structured positive definite matrix.  $\mathbf{M}$  is defined as

$$\begin{aligned}\mathbf{M} &= (\mathbf{M}^{(A)})^{-1} \otimes (\mathbf{M}^{(B)})^{-1} \otimes (\mathbf{M}^{(C)})^{-1}, \\ \mathbf{M}^{(B)} &= \mathbf{B}\mathbf{B}^T + (\mathbf{I} - \mathbf{B}\mathbf{B}^\dagger),\end{aligned}$$

and similarly defined for  $\mathbf{M}^{(A)}$  and  $\mathbf{M}^{(C)}$ . Mahalanobis norm [3] is a generalization of Frobenius norm by using covariance or ground metric matrices. For exact rank problems, the minima for any Mahalanobis norm corresponds to the minima of Frobenius norm. However, for approximation of a tensor with low CP rank, the stationary point of the AMDM algorithm may not be optimal in Frobenius norm metric that is the most used metric for assessing quality of the decomposition.

We empirically show that changing the metric  $\mathbf{M}$  from  $\mathbf{I}$  (which corresponds to the ALS update) to the proposed AMDM metric leads to a well-conditioned decomposition for approximation problems. A well-conditioned decomposition is useful for separation of components, clustering using CPD factors, and stability of application of the operator when an operator is approximated via CPD. We also show that by interpolating between AMDM and ALS updates, we obtain a hybrid algorithm that leads to better fitness as compared to ALS while maintaining a the quality of decomposition.

## References

- [1] C. J. Hillar and L.-H. Lim. Most tensor problems are NP-hard. *J. ACM*, 60(6):45:1–45:39, Nov. 2013.
- [2] M. Jouni, M. D. Mura, and P. Comon. Hyperspectral image classification based on mathematical morphology and tensor decomposition. *Mathematical Morphology-Theory and Applications*, 4(1):1–30, 2020.
- [3] P. Mahalanobis. On the generalised distance in statistics (reprint, 2018). *Sankhyā A*, 80(1):1–7, 1936.
- [4] K. R. Murphy, C. A. Stedmon, D. Graeber, and R. Bro. Fluorescence spectroscopy and multi-way techniques. PARAFAC. *Analytical Methods*, 5(23):6557–6566, 2013.
- [5] R. M. Parrish, E. G. Hohenstein, T. J. Martínez, and C. D. Sherrill. Tensor hypercontraction. ii. least-squares renormalization. *The Journal of chemical physics*, 137(22), 2012.
- [6] N. D. Sidiropoulos, L. De Lathauwer, X. Fu, K. Huang, E. E. Papalexakis, and C. Faloutsos. Tensor decomposition for signal processing and machine learning. *IEEE Transactions on Signal Processing*, 65(13):3551–3582, 2017.
- [7] N. Singh and E. Solomonik. Alternating mahalanobis distance minimization for accurate and well-conditioned cp decomposition. *SIAM Journal on Scientific Computing*, 45(6):A2781–A2812, 2023.

# Algorithms for Hermitian eigenproblems and their complexity

Aleksandros Sobczyk

## Abstract

Hermitian eigenproblems, and, more broadly, Hermitian-definite pencils, arise naturally in many real world applications in Machine Learning, Scientific Computing, and Engineering. Given a Hermitian matrix  $\mathbf{A}$  and a Hermitian positive-definite matrix  $\mathbf{B}$ , the goal is to compute (a subset of) the eigenvalues  $\lambda$  and/or the eigenvectors  $\mathbf{v}$ , which satisfy

$$\mathbf{Av} = \lambda \mathbf{Bv}.$$

In data science and machine learning, for example, they arise in Spectral Clustering [31, 39], Language Models [22], Image Processing [36, 1], Principal Components Analysis [11, 24, 37], and many others [12, 29, 14, 9, 28, 2]. A ubiquitous application in Scientific Computing is the computation of the density matrices and the electron densities in Density Functional Theory (DFT) [27].

Algorithms for eigenproblems have been studied since at least the nineteenth century, some early references being attributed to Jacobi [23, 16]. In this work we revisit algorithms from the computational complexity point of view, targeting (*i*) *eigenvalue problems*, which involve the approximation of eigenvalues, singular values, spectral gaps, and condition numbers, (*ii*) *eigenspace problems*, such as the approximation of eigenvectors, spectral projectors, and invariant subspaces, and (*iii*) *diagonalization problems*, which refer to the computation of all the eigenvalues and eigenvectors of a matrix, for example, a full spectral factorization, or the SVD.

This work is a summary of the results in [41, 40] for some variants of the aforementioned problems, which were part of the corresponding authors' doctoral dissertation. All of the algorithms and complexity bounds are in the following three *models of computation*:

**Exact real arithmetic (Real RAM)**, where a processor can execute the following operations:  $\{+, -, \times, /, \sqrt{\cdot}, >\}$ , on real numbers, in constant time, without any rounding errors;

**Rational arithmetic**, where numbers are represented as rationals consisting of an integral numerator and denominator of finite (but variable bit length, and each arithmetic operation does not introduce errors but takes time proportional to the number of bits);

**Floating point**, where any real number  $\alpha \in \mathbb{R}$  is rounded to a floating point number  $\mathbf{fl}(\alpha) = s \times 2^{e-t} \times m$ , and each arithmetic operation  $\odot \in \{+, -, \times, /, \sqrt{\cdot}\}$  introduces also some errors that satisfy  $\mathbf{fl}(\alpha \odot \beta) = (1 + \theta)(\alpha \odot \beta)$ ,  $\mathbf{fl}(\sqrt{a}) = (1 + \theta)\sqrt{a}$ , where  $|\theta| \leq \mathbf{u}$ . Assuming a total of  $b$  bits for each number, every floating point operation costs  $\mathcal{F}(b)$  bit operations, where typically it is assumed that  $\mathcal{F}(b) \in \tilde{O}(b)$  [19].

**Algorithms in Real RAM.** In the Real RAM model, we analyze the following problems:

1. Symmetric arrowhead/tridiagonal diagonalization,
2. Hermitian diagonalization,
3. Singular Value Decomposition.

We first provide an end-to-end complexity analysis of the divide-and-conquer algorithm of Gu and Eisenstat [18] for the diagonalization of tridiagonal and arrowhead matrices, when accelerated with the Fast Multipole Method [35]. By carefully analyzing all of the steps of the algorithm, we show that it provides provable approximation guarantees with the claimed nearly- $O(n^2)$  arithmetic complexity, which significantly improves classic (dense) eigensolvers such as the QR algorithm.

The tridiagonal diagonalization algorithm can be efficiently combined with the (rather overlooked) tridiagonal reduction algorithm of Schönhage [38], who proved that a Hermitian matrix can be reduced to tridiagonal form with unitary similarity transformations in  $O(n^\omega)$  arithmetic operations. Here  $\omega \lesssim 2.371$  is the current best known upper bound for the matrix multiplication exponent. This way, we can diagonalize a Hermitian matrix in nearly matrix multiplication time, improving the  $O(n^3)$  arithmetic complexity of classic algorithms [30, 17], as well as the more recent  $O(n^\omega \log^2(\frac{n}{\epsilon}))$  complexity of the randomized algorithm of [3]. Similar bounds are obtained for the deterministic complexity of the SVD. Many theoretical works assume the exact computation of an SVD as “black-box” subroutine; see e.g. [34, 15, 13, 7, 9, 8], to name a few. However, its complexity and approximation guarantees are often unspecified. Our main results and comparisons with existing algorithms are outlined in Table 1.

Table 1: Complexities for diagonalization problems in the Real RAM model. The randomized algorithms succeed with high probability (at least  $1 - 1/\text{poly}(n)$ ).  $O(n^{\omega(a,b,c)})$  denotes the complexity of multiplying two matrices with sizes  $n^a \times n^b$  and  $n^b \times n^c$ , respectively.

	Arithmetic Complexity	Deterministic
Arrowhead/Tridiagonal diagonalization		
[10, 32, 18]	$O(n^3) + \tilde{O}(n^2)$	✓
Our results	$O(n^2 \text{polylog}(\frac{n}{\epsilon}))$	✓
Hermitian diagonalization		
[3, 25]	$O(n^\omega \log^2(\frac{n}{\epsilon}))$	✗
[4]	$\tilde{O}(n^{\omega+1})$	✗
[30]	$O(n^3)$	✓
Our results	$O(n^\omega \log(n) + n^2 \text{polylog}(\frac{n}{\epsilon}))$	✓
SVD		
Fast MM + [30]	$O(n^{\omega(1,k,1)}) + \tilde{O}(n^3)$	✓
[3, 25]	$O(n^{\omega(1,k,1)} + n^\omega \log^2(\frac{n}{\epsilon}))$	✗
Our results	$O(n^{\omega(1,k,1)} + n^\omega \log(n) + n^2 \text{polylog}(\frac{n\kappa(\mathbf{A})}{\epsilon}))$	✓

**Algorithms in finite precision.** Similar deterministic complexity upper bounds are obtained for several problems in finite precision. We will present a stability analysis of the tridiagonal reduction algorithm of Schönhage, and its combination with the tridiagonal eigenvalue solver of [5, 6] (in rational arithmetic), to compute all the eigenvalues of a Hermitian matrix in nearly  $O(n^\omega)$  bit operations, deterministically.

Our main results, which are summarized in Table 2, improve several existing algorithms in different aspects. Pan and Chen [33] showed how to achieve additive errors in the eigenvalues in  $O(n^\omega)$  arithmetic operations, but this increases to  $O(n^{\omega+1})$  boolean operations in rational arithmetic. The algorithm of [30] achieves  $\tilde{O}(n^3)$  bit operations for all the eigenvalues. In the randomized setting, [3] also provides forward errors for the approximate eigenvalues in the Hermitian case, by exploiting a perturbation bound by Kahan [26, 41] in  $O(n^\omega \text{polylog}(n/\epsilon))$  boolean operations, but the logarithm power is fairly large. We also provide the analysis for several other useful subroutines related to

eigenvalue computations, including singular values, condition numbers, Hermitian-definite pencil eigenvalues, spectral gaps, and spectral projectors.

Table 2: Boolean complexity comparison in finite precision. Here  $\epsilon \in (0, \frac{1}{2})$ . FP stands for Floating Point and RA for Rational Arithmetic.

	Boolean Complexity	Success probability	Model
Tridiagonal Reduction			
[21, 20]	$O(n^3 \mathcal{F}(\log(\frac{n}{\epsilon})))$	Deterministic	FP
Our results	$O(n^\omega \log(n) \mathcal{F}(\log(\frac{n}{\epsilon})))$	Deterministic	FP
Hermitian Eigenvalues			
[30]	$O(n^3 \mathcal{F}(\log(\frac{n}{\epsilon})))$	Deterministic	FP
[3]+[26]	$O(n^\omega \log^2(\frac{n}{\epsilon}) \mathcal{F}(\log^4(\frac{n}{\epsilon}) \log(n)))$	$1 - O(1/n)$	FP
Our results	$O(n^\omega \mathcal{F}(\log(\frac{n}{\epsilon})) + n^2 \text{polylog}(\frac{n}{\epsilon}))$	Deterministic	FP+RA

## References

- [1] Gökhan H Bakir, Jason Weston, and Bernhard Schölkopf. Learning to find pre-images. *Advances in neural information processing systems*, 16:449–456, 2004.
- [2] Pierre Baldi and Kurt Hornik. Neural networks and principal component analysis: Learning from examples without local minima. *Neural networks*, 2(1):53–58, 1989.
- [3] Jess Banks, Jorge Garza-Vargas, Archit Kulkarni, and Nikhil Srivastava. Pseudospectral Shattering, the Sign Function, and Diagonalization in Nearly Matrix Multiplication Time. *Foundations of Computational Mathematics*, pages 1–89, 2022.
- [4] Michael Ben-Or and Lior Eldar. A Quasi-Random Approach to Matrix Spectral Analysis. In *Proc. 9th Innovations in Theoretical Computer Science Conference*, pages 6:1–6:22. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2018.
- [5] Dario Bini and Victor Pan. Parallel complexity of tridiagonal symmetric eigenvalue problem. In *Proc. of the Second Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 384–393, 1991.
- [6] Dario Bini and Victor Y Pan. Computing matrix eigenvalues and polynomial zeros where the output is real. *SIAM Journal on Computing*, 27(4):1099–1115, 1998.
- [7] Christos Boutsidis and David P Woodruff. Optimal cur matrix decompositions. In *Proc. of the forty-sixth Symposium on Theory of Computing*, pages 353–362, 2014.
- [8] Kenneth L Clarkson and David P Woodruff. Low-rank approximation and regression in input sparsity time. *Journal of the ACM (JACM)*, 63(6):1–45, 2017.
- [9] Michael B Cohen, Sam Elder, Cameron Musco, Christopher Musco, and Madalina Persu. Dimensionality reduction for k-means clustering and low rank approximation. In *Proc. 47th Symposium on Theory of Computing*, pages 163–172, 2015.
- [10] Jan JM Cuppen. A divide and conquer method for the symmetric tridiagonal eigenproblem. *Numerische Mathematik*, 36:177–195, 1980.
- [11] Konstantinos I Diamantaras and Sun Yuan Kung. *Principal component neural networks: theory and applications*. John Wiley & Sons, Inc., 1996.
- [12] Petros Drineas, Iordanis Kerenidis, and Prabhakar Raghavan. Competitive recommendation systems. In *Proc. Thiry-Fourth Annual ACM Symposium on Theory of Computing*, pages 82–90, 2002.
- [13] Petros Drineas, Michael W Mahoney, and Shan Muthukrishnan. Relative-error cur matrix decompositions. *SIAM Journal on Matrix Analysis and Applications*, 30(2):844–881, 2008.
- [14] Dan Feldman, Melanie Schmidt, and Christian Sohler. Turning big data into tiny data: Constant-size coresets for k-means, pca, and projective clustering. *SIAM Journal on Computing*, 49(3):601–657, 2020.
- [15] Alan Frieze, Ravi Kannan, and Santosh Vempala. Fast monte-carlo algorithms for finding low-rank approximations. *Journal of the ACM (JACM)*, 51(6):1025–1041, 2004.

- [16] Gene H Golub and Henk A Van der Vorst. Eigenvalue computation in the 20th century. *Journal of Computational and Applied Mathematics*, 123(1-2):35–65, 2000.
- [17] Gene H Golub and Charles F Van Loan. *Matrix Computations*. Johns Hopkins University Press, 2013.
- [18] Ming Gu and Stanley C Eisenstat. A divide-and-conquer algorithm for the symmetric tridiagonal eigenproblem. *SIAM Journal on Matrix Analysis and Applications*, 16(1):172–191, 1995.
- [19] David Harvey and Joris Van Der Hoeven. Integer multiplication in time  $O(n \log n)$ . *Annals of Mathematics*, 193(2):563–617, 2021.
- [20] Nicholas J Higham. *Accuracy and stability of numerical algorithms*. SIAM, 2002.
- [21] Alston S Householder. Unitary triangulation of a nonsymmetric matrix. *Journal of the ACM*, 5(4):339–342, 1958.
- [22] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.
- [23] C.G.J. Jacobi. Über ein leichtes verfahren die in der theorie der säcularstörungen vorkommenden gleichungen numerisch aufzulösen. *Journal für die reine und angewandte Mathematik*, 30:51–94, 1846.
- [24] Ian T Jolliffe. *Principal component analysis for special types of data*. Springer, 2002.
- [25] Praneeth Kacham and David P Woodruff. Faster algorithms for schatten-p low rank approximation. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2024)*. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2024.
- [26] W Kahan. Spectra of nearly hermitian matrices. *Proc. of the American Mathematical Society*, 48(1):11–17, 1975.
- [27] Walter Kohn and Lu Jeu Sham. Self-consistent equations including exchange and correlation effects. *Physical Review*, 140(4A):A1133, 1965.
- [28] Arnaz Malhi and Robert X Gao. Pca-based feature selection scheme for machine defect classification. *IEEE transactions on instrumentation and measurement*, 53(6):1517–1525, 2004.
- [29] Olvi L Mangasarian and Edward W Wild. Multisurface proximal support vector machine classification via generalized eigenvalues. *IEEE transactions on pattern analysis and machine intelligence*, 28(1):69–74, 2005.
- [30] Yuji Nakatsukasa and Nicholas J Higham. Stable and efficient spectral divide and conquer algorithms for the symmetric eigenvalue decomposition and the svd. *SIAM Journal on Scientific Computing*, 35(3):A1325–A1349, 2013.
- [31] Andrew Ng, Michael Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. *Advances in Neural Information Processing Systems*, 14, 2001.
- [32] Dianne P O’Leary and Gilbert W Stewart. Computing the eigenvalues and eigenvectors of symmetric arrowhead matrices. *Journal of Computational Physics*, 90(2):497–505, 1990.
- [33] Victor Y Pan and Zhao Q Chen. The complexity of the matrix eigenproblem. In *Proc. 31st Annual ACM Symposium on Theory of Computing*, pages 507–516, 1999.
- [34] Christos H Papadimitriou, Hisao Tamaki, Prabhakar Raghavan, and Santosh Vempala. Latent semantic indexing: A probabilistic analysis. In *Proc. of the seventeenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*, pages 159–168, 1998.
- [35] Vladimir Rokhlin. Rapid solution of integral equations of classical potential theory. *Journal of computational physics*, 60(2):187–207, 1985.
- [36] Awwal Mohammed Rufai, Gholamreza Anbarjafari, and Hasan Demirel. Lossy image compression using singular value decomposition and wavelet difference reduction. *Digital signal processing*, 24:117–123, 2014.
- [37] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation*, 10(5):1299–1319, 1998.
- [38] Arnold Schönhage and Volker Strassen. Fast multiplication of large numbers. *Computing*, 7:281–292, 1971.
- [39] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905, 2000.
- [40] Aleksandros Sobczyk. Deterministic complexity analysis of Hermitian eigenproblems. *arXiv preprint arXiv:2410.21550*, Under review, 2024.
- [41] Aleksandros Sobczyk, Marko Mladenović, and Mathieu Luisier. Invariant subspaces and PCA in nearly matrix multiplication time, 2024.

# Filtration of Lanczos vectors in hybrid CG Tikhonov iteration

*Kirk M. Soodhalter, Daniel Gerth*

## Abstract

We consider iterative methods for solving a linear ill-posed problem of the form

$$Ax \approx y = y^\delta - \delta \cdot n$$

wherein  $A : \mathcal{X} \rightarrow \mathcal{Y}$  is a compact linear operator, and  $y^\delta$  is a version of the right-hand side obtained by noisy measurements, with  $\|n\| = 1$  and  $0 < \delta \ll 1$ . We assume that we only have access to  $y^\delta$ . It is well known that naïve solution using the pseudoinverse operator  $A^\dagger y^\delta$  may lead to amplification of the measurement noise, unbounded in the infinite-dimensional case and bounded but large in the finite-dimensional case.

Conjugate gradients applied to the normal equations (CG)  $A^* A x^\delta = A^* y^\delta$  with an appropriate stopping rule and CG applied to the system solving for a Tikhonov-regularized solution (CGT)  $(A^* A + cI_{\mathcal{X}})x^{(\delta,c)} = A^* y^\delta$  ( $c > 0$  is the Tikhonov parameter) are closely related methods. It has been long observed that they build iterates from the same family of Krylov subspaces, due to the scalar shift invariance property of Krylov subspaces [4]; i.e.,  $\mathcal{K}_m(A^* A, A^* y^\delta) = \mathcal{K}_m(A^* A + cI_{\mathcal{X}}, A^* y^\delta)$ . With this in mind, one can express both CG-based iterates with respect to the same Lanczos basis. In particular, one can use this to understand how the representation of the CGT iterates change as a function of  $c$  with  $c \rightarrow 0$  yielding a CG iterate. Let  $x_m^\delta = \sum_{i=1}^m z_i^{(m)} v_i$  be the CG iterate where  $\{v_i\}_{i=1}^m$  is the Lanczos basis for  $\mathcal{K}_m(A^* A, A^* y^\delta)$ . Via linear algebraic manipulations, one can show that the CGT iterate can be expressed as

$$x_m^{(\delta,c)} = \sum_{i=1}^m \gamma_i^{(m)}(c) z_i^{(m)} v_i,$$

where  $\{\gamma_i^{(m)}(c)\}_{i=1}^m$  are functions of the Tikhonov parameter. These coefficient multiplier functions can be shown to have decay properties as  $c \rightarrow \infty$  with the speed of decay increasing with  $i$ , asymptotically. This has the effect of filtering out the contribution of the later terms of the CG iterate. Thus, we call these functions *Lanczos filters*, as they express the effect of CGT regularization in terms of the CG expressed in the Lanczos basis rather than in terms of the singular vector basis, as is the case of classical definition of filter function in regularization theory [3].

Much of this work is explored in the context of infinite dimensional ill-posed problems to present the analysis as generally as possible. For this, we build upon the work in [1] (which works with the equivalent Golub-Kahan/LSQR formulation of these methods) to prove some additional convergence results, to help us understand the behavior of the CG and CGT iterates.

If we restrict our focus on the behavior of these methods when applied to finite-dimensional discrete ill-posed problems, we can understand these filters as dampening the influence of Lanczos vectors that are more highly polluted with noise. The mechanics by which noise comes to pollute the Lanczos vectors has been illuminated by means of Gauss-Radau quadrature in [5, 6], and the review [2] and references therein discuss how this has been previously used for Tikhonov parameter selection.

We demonstrate with numerical experiments that good parameter choices correspond to appropriate damping of the Lanczos vectors corresponding to larger amplifications of the measurement noise.

Building on this idea, one can consider approaches other than Tikhonov for damping amplified noise. We conclude by noting that analysis of other hybrid regularization schemes via damping of (Krylov) subspace basis vectors from the iteration itself may be a useful avenue for understanding the behavior of these methods for different choice of parameter, etc.

## References

- [1] A. ALQAHTANI, R. RAMLAU, AND L. REICHEL, *Error estimates for Golub–Kahan bidiagonalization with tikhonov regularization for ill-posed operator equations*, 39, p. 025002.
- [2] J. CHUNG AND S. GAZZOLA, *Computational methods for large-scale inverse problems: A survey on hybrid projection methods*, 66, pp. 205–284.
- [3] H. W. ENGL, M. HANKE, AND A. NEUBAUER, *Regularization of inverse problems*, vol. 375 of Mathematics and its Applications, Kluwer Academic Publishers Group, Dordrecht, 1996.
- [4] A. FROMMER AND P. MAASS, *Fast CG-based methods for Tikhonov–Phillips regularization*, 20, pp. 1831–1850.
- [5] I. HNĚTYNKOVÁ, M. KUBÍNOVÁ, AND M. PLEŠINGER, *Noise representation in residuals of LSQR, LSMR, and CRAIG regularization*, 533, pp. 357–379.
- [6] I. HNĚTYNKOVÁ, M. PLEŠINGER, AND Z. STRAKOŠ, *The regularizing effect of the Golub-Kahan iterative bidiagonalization and revealing the noise level in the data*, 49, pp. 669–696.

# GMRES with Preconditioning, Weighted norm and Deflation

Nicole Spillane, Daniel B. Szyld

## Abstract

We consider the general problem of solving a linear system of the form

$$\mathbf{Ax} = \mathbf{b}; \quad \mathbf{A} \in \mathbb{C}^{n \times n}; \quad \mathbf{b} \in \mathbb{C}^n.$$

The matrices  $\mathbf{A}$  that we consider are non-singular, sparse and of high order  $n$ . For solving these matrices, GMRES [3, Chapter 6] is a natural choice. We address two fundamental and connected questions: How can the convergence of GMRES be predicted? How can the convergence of GMRES be accelerated? Our aim is to combine three ways of accelerating GMRES convergence:

- Weighting by a Hermitian positive definite (hpd) matrix  $\mathbf{W}$ : all inner products and norms in the GMRES algorithm are replaced by the ones induced by  $\mathbf{W}$  (see [1]),
- Preconditioning by a non-singular matrix  $\mathbf{H}$ : GMRES is applied to the preconditioned problem  $\mathbf{AHu} = \mathbf{b}$  with  $\mathbf{x} = \mathbf{Hu}$  (see [3, Section 9.3]),
- Deflation by a projection operator  $\mathbf{\Pi} := \mathbf{I} - \mathbf{AZ}(\mathbf{Y}^* \mathbf{AZ})^{-1} \mathbf{Y}^*$  (with  $\mathbf{Y}, \mathbf{Z} \in \mathbb{C}^{n \times m}$ ): GMRES is applied to the projected problem  $\mathbf{\Pi A H u} = \mathbf{\Pi b}$  (see [7, 4]). A suitable initialization is also performed that accounts for the part of the solution that has been projected away.

We refer to  $\mathbf{W}$ ,  $\mathbf{H}$  and  $\mathbf{\Pi}$  as accelerators for GMRES. With words, the strategy is that the preconditioner  $\mathbf{H}$  should be a *good* approximation of  $\mathbf{A}^{-1}$ , the deflation operator should handle the space where  $\mathbf{H}$  does not *well* approximate  $\mathbf{A}^{-1}$ , and the weighted inner product should facilitate the analysis. In practice, identifying efficient accelerators requires a GMRES convergence bound where the influence of  $\mathbf{H}$ ,  $\mathbf{W}$  and  $\mathbf{\Pi}$  is explicit. We prove in [6, Theorem 4.1] that the convergence rate is bounded by

$$\frac{\|\mathbf{r}_{i+1}\|_{\mathbf{W}}^2}{\|\mathbf{r}_i\|_{\mathbf{W}}^2} \leq 1 - \inf_{\mathbf{y} \in \text{range}(\mathbf{\Pi}) \setminus \{\mathbf{0}\}} \frac{|\langle \mathbf{\Pi A H y}, \mathbf{y} \rangle_{\mathbf{W}}|^2}{\|\mathbf{\Pi A H y}\|_{\mathbf{W}}^2 \|\mathbf{y}\|_{\mathbf{W}}^2}.$$

**Further Assumptions** Major simplifications occur in the case where  $\mathbf{A}$  is positive definite (*i.e.*, its Hermitian part is hpd), the preconditioner  $\mathbf{H}$  is hpd, and the weight equals the preconditioner  $\mathbf{W} = \mathbf{H}$ . In this case (and with a technical assumption on the deflation operator), it holds that

$$\frac{\|\mathbf{r}_{i+1}\|_{\mathbf{H}}^2}{\|\mathbf{r}_i\|_{\mathbf{H}}^2} \leq 1 - \inf_{\mathbf{y} \in \text{range}(\mathbf{A H \Pi}) \setminus \{\mathbf{0}\}} \frac{|\langle \mathbf{A}^{-1} \mathbf{y}, \mathbf{y} \rangle|}{\langle \mathbf{y}, \mathbf{M}^{-1} \mathbf{y} \rangle} \times \frac{\lambda_{\min}(\mathbf{H M})}{\lambda_{\max}(\mathbf{H M})},$$

where  $\mathbf{M} = 1/2(\mathbf{A} + \mathbf{A}^*)$  and  $\mathbf{N} = 1/2(\mathbf{A} - \mathbf{A}^*)$  are the Hermitian and skew-Hermitian parts of  $\mathbf{A}$ , and the spectrum of  $\mathbf{H M}$  is in the interval  $[\lambda_{\min}(\mathbf{H M}), \lambda_{\max}(\mathbf{H M})]$ .

**Convergence without deflation** Setting  $\mathbf{\Pi} = \mathbf{I}$  (no deflation) and with an identity from [2] it is proved in [5, Theorem 4.3] that

$$\frac{\|\mathbf{r}_{i+1}\|_{\mathbf{H}}^2}{\|\mathbf{r}_i\|_{\mathbf{H}}^2} \leq 1 - \frac{1}{1 + \rho(\mathbf{M}^{-1} \mathbf{N})^2} \times \frac{\lambda_{\min}(\mathbf{H M})}{\lambda_{\max}(\mathbf{H M})}$$

where  $\rho(\cdot)$  denotes the spectral radius of a matrix. The residuals are bounded with respect to two quantities. The first is the condition number of  $\mathbf{HM}$ , a measure of whether  $\mathbf{H}$  is a good preconditioner for the hpd matrix  $\mathbf{M}$ . The second is the spectral radius of  $\mathbf{M}^{-1}\mathbf{N}$ , a measure of how non-Hermitian the problem is. The takeaway is that fast convergence is guaranteed if the problem is mildly non-Hermitian and  $\mathbf{H}$  is a good preconditioner for  $\mathbf{M}$ . The bound also has important consequences for parallel computing and the analysis of domain decomposition methods.

**A new deflation space [6, Theorem 6.3]** When the problem is significantly non-Hermitian (in terms of  $\rho(\mathbf{M}^{-1}\mathbf{N})$ ), we propose to combine Hermitian preconditioning with spectral deflation. Under the same assumptions as above, we choose the matrices  $\mathbf{Z}$  and  $\mathbf{Y}$  in the characterization of the projection operator  $\mathbf{\Pi}$  as follows. First, we denote by  $(\lambda_j, \mathbf{z}^{(j)}) \in i\mathbb{R} \times \mathbb{C}^n$  (for  $j = 1, \dots, n$ ) the eigenpairs of the generalized eigenvalue problem  $\mathbf{N}\mathbf{z}^{(j)} = \lambda_j \mathbf{M}\mathbf{z}^{(j)}$ . Then, with a chosen threshold  $\tau > 0$  we select for the deflation operator, the highest frequency eigenvectors, by setting

$$\text{span}(\mathbf{Z}) := \text{span}\{\mathbf{z}^{(j)}; |\lambda_j| > \tau\} \text{ and } \mathbf{Y} = \mathbf{HAZ}.$$

Then the convergence of weighted, preconditioned and deflated GMRES is bounded by

$$\frac{\|\mathbf{r}_{i+1}\|_{\mathbf{H}}^2}{\|\mathbf{r}_i\|_{\mathbf{H}}^2} \leq 1 - \frac{1}{(1 + \tau^2)} \times \frac{\lambda_{\min}(\mathbf{HM})}{\lambda_{\max}(\mathbf{HM})}.$$

Numerical illustrations show that preconditioning the Hermitian part in a way that is scalable leads to overall scalability and that spectral deflation accelerates convergence when the problems become more strongly non-Hermitian.

## References

- [1] A. Essai. Weighted FOM and GMRES for solving nonsymmetric linear systems. *Numer. Algorithms*, 18(3-4):277–292, 1998.
- [2] C. R. Johnson. Inequalities for a complex matrix whose real part is positive definite. *Trans. Am. Math. Soc.*, 212:149–154, 1975.
- [3] Y. Saad. *Iterative methods for sparse linear systems*. Philadelphia, PA: SIAM Society for Industrial and Applied Mathematics, 2nd ed. edition, 2003.
- [4] K. M. Soodhalter, E. de Sturler, and M. E. Kilmer. A survey of subspace recycling iterative methods. *GAMM-Mitt.*, 43(4):29, 2020. Id/No e202000016.
- [5] N. Spillane. Hermitian preconditioning for a class of non-Hermitian linear systems. *SIAM J. Sci. Comput.*, 46(3):a1903–a1922, 2024.
- [6] N. Spillane and D. B. Szyld. New convergence analysis of GMRES with weighted norms, preconditioning, and deflation, leading to a new deflation space. *SIAM J. Matrix Anal. Appl.*, 45(4):1721–1745, 2024.
- [7] J. M. Tang, R. Nabben, C. Vuik, and Y. A. Erlangga. Comparison of two-level preconditioners derived from deflation, domain decomposition and multigrid methods. *J. Sci. Comput.*, 39:340–370, 2009.

# Evaluating and improving streaming methods for large scale SVD problems

*Andreas Stathopoulos, Jeremy Myers, Toon Tran*

## Abstract

Large scale eigenvalue and singular value problems are typically solved using iterative methods [10, 12]. For extreme scale matrix sizes randomized projection methods can be much faster while still delivering sufficient accuracy, measured as the distance from the optimal low rank matrix. The accuracy can be improved by following the projection with subspace iteration [7, 8]. However, achieving high accuracy for individual singular vectors is generally more challenging.

In this work we address the problem where the matrix is so large that cannot be stored in its entirety and its re-computation is either too expensive or not possible; hence even iterative methods are infeasible. This situation is becoming increasingly common in the era of “bigger” data and is often referred to as streaming, i.e., when the data arrives in some order, is processed, and then forgotten. In terms of matrices, we assume that a matrix is streamed in  $m$  linear updates (see [14]),  $A = \sum_{i=1}^m H_i$ , but we focus our attention to streaming by rows, i.e.,  $H_i$  is a set of rows of  $A$ .

Randomized methods are naturally suited for streaming. When a new  $H_i$  arrives, its randomized projection is recorded, and the method continues until the entire matrix has been streamed, at which point an approximate SVD can be computed from the projection. A series of improvements on this basic idea [18, 15, 16, 13] have resulted in an efficient randomized method called SketchySVD [14].

A different class of deterministic streaming SVD methods has been proposed but has not received as much attention despite its potential for more accurate approximations. We use Incremental SVD (iSVD) as a prototype such method. Inductively at step  $i + 1$ , iSVD appends the new window  $H_{i+1}$  to an existing rank- $k$  approximation  $B^{(i)}$ , computes the SVD of  $[B^{(i)}; H_{i+1}]$ , and then updates  $B^{(i+1)}$  based on the rank- $k$  truncation of the SVD. Earlier works [9, 2, 4, 3, 5, 20] compute only the left or right singular vectors, while the iSVD of Baker et al. [1] generalized these approaches to track both left and right singular vectors albeit at a higher computational cost.

A notable difference is that iSVD provides a running low-rank approximation at every window, while SketchySVD can wait till the end of streaming to compute it. Broadly speaking, randomized sketching methods have low time and space complexity whereas deterministic sketching methods have higher accuracy. However, the trade-offs have not been carefully studied in the literature. Empirical results with randomized sketching methods do not compare with streaming or use datasets that are typically small enough to be processed by batch methods [14]. In this work we explore these missing comparisons and we introduce some new ideas for improving iSVD. Our contributions can be summarized as follows:

- Traditional iSVD methods update the low rank approximation one row or a small number of rows at a time. Because iSVD accuracy improves with larger window size (number of rows in  $H_i$ ), we instead make the window size as large as memory can hold. Because only a low rank approximation is needed, iterative methods can be used to solve the partial SVD of the large rectangular window.
- To evaluate the benefits of these streaming methods we need to address enormous problem sizes. For this reason, we provide a high-performance C++ implementation of both SketchySVD and iSVD called Skema (available at <https://github.com/jeremy-myers/>)

`skema`). To perform dense and sparse matrix-vector multiplications (matvecs), Skema leverages the Kokkos Performance Portability Ecosystem [6] for hierarchical parallelism on heterogeneous architectures, including x86 and accelerators. To compute a few eigenpairs or singular triplets, Skema uses the PRIMME [11, 19] library. Dense matrix operations inside PRIMME utilize multithreaded BLAS on CPUs and MAGMA on accelerators.

- We provide a complexity analysis of the “large window” streaming method and compare it with a similar analysis of SketchySVD. We show that iSVD is more expensive than the same iterative method applied to the entire matrix for the same number of iterations. The overhead is proportional to the number of windows, especially for very sparse matrices. Therefore, choosing larger window size not only improves accuracy but also reduces the time overhead.
- We also perform extensive numerical results on problems with enormous dimensions that are much larger than those in the literature. Sources of problems include: stock price prediction (kernel learning), social networks/analysis graphs, and scientific simulation data. We observe that iSVD approximations are at least as accurate as SketchySVD ones and often several orders of magnitude more accurate. Comparing runtimes, SketchySVD is typically faster, but not as much as the complexity analysis suggests. This is because dense Gaussian embeddings involve too many operations while sparsenmaps implementations present a challenging memory access pattern.
- Using iterative methods as the SVD solver at each window allows the use of initial guesses which are readily available from the previous window, i.e.,  $B^{(i)}$  transformed appropriately to correspond to the  $[B^{(i)}H_i]$  matrix. Since the low rank approximations  $B^{(i)}$  change relatively slowly between windows, initial guesses provide a substantial reduction in the number of iterations, around 30-50%.
- iSVD allows for further optimizations when solving for the largest eigenvalues of a symmetric positive definite (SPD) matrix. Such problems are common in large graph Laplacians, covariance matrices, and kernel methods in machine learning. Since the SVD and the eigenvalue problem are equivalent for SPD matrices, any right singular vector  $v$  of the rectangular window at any iSVD step is an approximation to an eigenvector of  $A$ . Therefore, for any row  $m$  of  $A$  we can compute the  $m$ -th value of the eigenvalue residual as  $A(m,:)v - \lambda v(m)$ . This motivates the following convergence criterion for each iSVD window. Notice that while the iterative method converges to the SVD of the rectangular window, the corresponding eigenvalue residuals for  $A$  stop making progress after some iterations. If we can estimate the  $A$  residuals we can check for this and stop early. We use reservoir sampling [17] to create and store a subset of rows of  $A$ ,  $A_S$ , for which we can estimate the residual values and extrapolate the residual norm to the entire matrix. Reservoir sampling is a method to maintain a uniform sample of elements that have been streamed up to now. Preliminary results on this idea have been promising for further reducing the number of iterations.
- In some cases, the streaming order of rows can be chosen by the user. An example is when a row of a covariance or a kernel matrix is computed on demand from the data (data requires  $O(n)$  storage vs  $O(n^2)$  storage for the entire matrix). Therefore, the question arises of what is the effect of streaming order in the final accuracy of the low rank space, and whether this is achieved early or late during streaming. Based on the convergence analysis of [1] we show that, on average, each window provides a similar additive improvement on accuracy. This means that iSVD does not see the best accuracy until the last window. We have observed that if rows are streamed in the order of decreasing row norms of  $A$ , the final accuracy is

achieved very early in streaming. Thus, we explore the idea of using the row norms of the low rank approximation  $B^{(i)}$  as leverage scores to stream first the remaining rows with the largest norms. This heuristic also achieves a similar behavior where most of the accuracy is achieved earlier. If combined with our residual norm estimation using reservoir sampling, this heuristic may suggest stopping the streaming before all windows have been streamed. This approach is still under investigation.

## References

- [1] Chris G. Baker, Kyle A. Gallivan, and Paul Van Dooren. Low-rank incremental methods for computing dominant singular subspaces. *Linear Algebra and its Applications*, 436(8):2866–2888, 2012.
- [2] Matthew Brand. Incremental Singular Value Decomposition of Uncertain Data with Missing Values. In Anders Heyden, Gunnar Sparr, Mads Nielsen, and Peter Johansen, editors, *Computer Vision — ECCV 2002*, volume 2350 of *Lecture Notes in Computer Science*, pages 707–720, Berlin, 2002. Springer.
- [3] Matthew Brand. Fast low-rank modifications of the thin singular value decomposition. *Linear Algebra and its Applications*, 415(1):20–30, 2006.
- [4] Y. Chahlaoui, K. Gallivan, and P. Van Dooren. Recursive calculation of dominant singular subspaces. *SIAM Journal on Matrix Analysis and Applications*, 25(2):445–463, 2003.
- [5] Yongsheng Cheng, Jiang Zhu, and Xiaokang Lin. An enhanced incremental SVD algorithm for change point detection in dynamic networks. *IEEE access : practical innovations, open solutions*, 6:75442–75451, 2018.
- [6] H. Carter Edwards, Christian R. Trott, and Daniel Sunderland. Kokkos: Enabling manycore performance portability through polymorphic memory access patterns. *Journal of Parallel and Distributed Computing*, 74(12):3202–3216, December 2014.
- [7] M. Gu. Subspace Iteration Randomization and Singular Value Problems. *SIAM Journal on Scientific Computing*, 37(3):A1139–A1173, 2015.
- [8] N. Halko, P. G. Martinsson, and J. A. Tropp. Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions. *Siam Review*, 53(2):217–288, May 2011.
- [9] A. Levey and M. Lindenbaum. Sequential Karhunen-Loeve basis extraction and its application to images. *IEEE Transactions on Image Processing*, 9(8):1371–1374, 2000.
- [10] Y. Saad. *Numerical methods for large eigenvalue problems*. Manchester University Press, 1992.
- [11] Andreas Stathopoulos and James R. McCombs. PRIMME: PReconditioned Iterative Multi-Method Eigensolver – Methods and software description. *ACM Transactions on Mathematical Software*, 37(2):21:1–21:30, 2010.
- [12] G. W. Stewart. *Matrix Algorithms Vol.II: Eigensystems*. SIAM, Philadelphia, PA, 2001.

- [13] Joel A. Tropp, Alp Yurtsever, Madeleine Udell, and Volkan Cevher. Fixed-Rank Approximation of a Positive-Semidefinite Matrix from Streaming Data. *CoRR*, abs/1706.05736, 2017.
- [14] Joel A. Tropp, Alp Yurtsever, Madeleine Udell, and Volkan Cevher. Streaming Low-Rank Matrix Approximation with an Application to Scientific Simulation. *SIAM Journal on Scientific Computing*, 41(4):A2430–A2463, 2019.
- [15] Jalaj Upadhyay. Fast and space-optimal low-rank factorization in the streaming model with application in differential privacy, April 2016.
- [16] Jalaj Upadhyay. The price of privacy for low-rank factorization. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS’18, pages 4180–4191, Red Hook, NY, USA, 2018. Curran Associates Inc.
- [17] Jeffrey S. Vitter. Random sampling with a reservoir. *ACM Transactions on Mathematical Software*, 11(1):37–57, March 1985.
- [18] Franco Woolfe, Edo Liberty, Vladimir Rokhlin, and Mark Tygert. A fast randomized algorithm for the approximation of matrices. *Applied and Computational Harmonic Analysis*, 25(3):335–366, November 2008.
- [19] Lingfei Wu, Eloy Romero, and Andreas Stathopoulos. PRIMME\_SVDS: A High-Performance Preconditioned SVD Solver for Accurate Large-Scale Computations. *Siam Journal On Scientific Computing*, 39(5):S248–S271, 2017.
- [20] Yangwen Zhang. An answer to an open question in the incremental SVD, 2022.

# Rational Krylov methods for exponential Runge-Kutta integrators on networks

*Martin Stoll, Kai Bergermann*

## Abstract

We consider the solution of large stiff systems of ordinary differential equations with explicit exponential Runge–Kutta integrators [2] for systems of the form

$$\frac{\partial \mathbf{u}(t)}{\partial t} = F(t, \mathbf{u}(t)) = -\mathbf{A}\mathbf{u}(t) + g(t, \mathbf{u}(t)), \quad \mathbf{u}(0) = \mathbf{u}_0, \quad (1)$$

where we view them as being semi-discretized semi-linear parabolic partial differential equations on continuous domains or on inherently discrete graph domains. This results in  $\mathbf{A} = \Delta$  (the continuour case) or  $\mathbf{A} = \mathbf{L}$  the graph case.

We suggest the use computing linear combinations of  $\varphi$ -functions in exponential integrators [1] to solve the above problem. State-of-the-art computational methods use polynomial Krylov subspaces of adaptive size for this task. These methods often suffer from that the required number of Krylov subspace iterations to obtain a desired tolerance increase drastically with the spectral radius of the system matrix  $\mathbf{A}$ .

We present an approach that leverages rational Krylov subspace methods for this task (cf. [5]). The main workhorse are Runge-Kutta schemes. For these, one can now employ the scheme with  $s$  internal stages  $t + c_1 h_i, \dots, t + c_s h_i$  with  $c_j \in [0, 1]$  for  $1 \leq j \leq s$  into the time interval  $[t, t + h_i]$  leading to schemes of the form

$$\mathbf{u}_{i+1} = \chi(-h_i \mathbf{A}) \mathbf{u}_i + h_i \sum_{j=1}^s b_j (-h_i \mathbf{A}) \mathbf{G}_{ij}. \quad (2)$$

$$\mathbf{U}_{ij} = \chi_j(-h_i \mathbf{A}) \mathbf{u}_i + h_i \sum_{k=1}^s a_{jk} (-h_i \mathbf{A}) \mathbf{G}_{ik}, \quad (3)$$

$$\mathbf{G}_{ik} = g(t_i + c_k h_i, \mathbf{U}_{ik}), \quad (4)$$

where  $\chi$ ,  $\chi_j$ ,  $a_{jk}$ , and  $b_j$  are  $\varphi$ -functions (cf. [4]).

Our exponential integrator approach relies on the use of rational Krylov approximations to the matrix functions via

$$f(\tilde{\mathbf{A}})\tilde{\mathbf{c}} \approx \|\tilde{\mathbf{c}}\|_2 \mathbf{V}_m f(\mathbf{H}_m \mathbf{K}_m^{-1}) \mathbf{e}_1, \quad (5)$$

where  $\mathbf{H}_m$  and  $\mathbf{K}_m$  are small matrices coming from the rational Arnoldi method. We then give a novel a-posteriori error estimate of the rational Krylov approximation to the action of the matrix exponential on vectors for single time points. This bound allows for an adaptive approach similar to existing polynomial Krylov techniques. We then briefly discuss the selection of poles and the need for solving linear systems efficiently. The key to the convincing performance is to construct preconditioners that lead to approximately constant iteration numbers.

Numerical experiments show that our method outperforms the state of the art for sufficiently large spectral radii of the discrete linear differential operators [3]. We focus on well-known nonlinear partial differential equation models allowing the fast simulations of examples including Turing patterns. Additionally, we show that this approach allows the fast simulation of nonlinear network dynamical systems.

## References

- [1] A. H. AL-MOHY AND N. J. HIGHAM, *Computing the action of the matrix exponential, with an application to exponential integrators*, SIAM J. Sci. Comput., 33 (2011), pp. 488–511.
- [2] R. ALEXANDER, *Diagonally implicit Runge–Kutta methods for stiff ODE’s*, SIAM J. Numer. Anal., 14 (1977), pp. 1006–1021.
- [3] S. M. ALLEN AND J. W. CAHN, *A microscopic theory for antiphase boundary motion and its application to antiphase domain coarsening*, Acta Metallurgica, 27 (1979), pp. 1085–1095.
- [4] K. BERGERMANN AND M. STOLL, *Adaptive rational Krylov methods for exponential Runge–Kutta integrators*, SIAM J. Matrix Anal. Appl., 45 (2023), pp. 744–770.
- [5] S. MASSEI AND L. ROBOL, *Rational Krylov for Stieltjes matrix functions: Convergence and pole selection*, BIT, 61 (2021), pp. 237–273.

# Sparsify Latent Factor Matrix by Householder Transformations

Xiaobai Sun

## Abstract

In 1958 A. S. Householder (1904-1993) introduced the reflection transformation in his highly influential paper, *Unitary Triangularization of a Nonsymmetric Matrix*, published in the Journal of the ACM. He presented the reflection as a special case of nonsingular transformation matrices in the form of a rank-1 deviation from the identity matrix. In that same year, H. F. Kaiser (1927-1992) published the seminal paper *the Varimax Criterion for Analytic Rotations in Factor Analysis* in Psychometrika. Both papers have seen increasing citations in recent years, as will be demonstrated. This work introduces the use of Householder transformations for effective and efficient rotations and sparsification of latent factors. It has several advantages over the state-of-the-art factor rotation methods. This appears to be the first connection between these two lines of research.<sup>1</sup>

Analytic rotations are central to multiple factor analysis. Factor analysis is a statistical method to uncover one or more than one latent variables, a.k.a. factors, that explain or interpret the correlations among observable and observed variables. Latent variable analysis, originated from the pioneering work of C. Spearman in 1904 in psychology, is indispensable to modern exploratory analysis of data from various study fields, especially in social sciences and biomedical sciences. In multi-factor analysis, the relationship between the observable and latent variables is represented by a factor (loading) matrix. The concept of model simplification by factor rotations was conceived and developed between 1932 and 1938 by L. L Thurstone (1887-1955). Factor rotations are preceded by an initial factor extraction, which can be obtained manually based on expert knowledge or automatically via principal component analysis, maximal likelihood estimate, or other approaches. The factor axes are then re-oriented by orthogonal or oblique rotation transformations, to simplify (i.e. sparsify) the factor matrix pattern. The purpose is to identify salient relationships between the observed variables and the latent factors and to explain or interpret the correlation in observed phenomena. The term *simplification* here refers to reducing the complexity of observed variables in terms of the underlying factors. Thurstone's five simplification rules have been notably refined over time. Kaiser's varimax criterion and solution methods have a broad and lasting impact.

The factor rotation problem can be generally described as a factor matrix transformation governed by a constrained nonlinear optimization problem. An objective function specifies a simplification (or sparsification) criterion based on desired properties of the rotated factor matrix. All existing criteria, including Kaiser's criterion, are nonlinear functions with respect to the elements of the rotated factor matrix. Constraints include equations to ensure the orthogonality in orthogonal rotations or to preserve the variance per factor and avoid factor collapse in oblique rotations. Additional constraints may be imposed in confirmatory factor analysis to align with reference or target factor patterns.

Consider a particular case. Let  $B_{p \times m}$  be the initial factor (loading) matrix with  $p$  variables and  $m$  factors, where  $1 < m < p$ . Let  $L(U) = BU$  be the loading matrix rotated from  $B$  by an orthogonal transformation  $U$ . Kaiser's criterion can then be described as follows,

$$U_* = \arg \max_{U^T U = I} \phi(L(U)) = \frac{e^T (L^T L)^{-1} e}{p} - \sum_{j=1:m} \left( \frac{L(:, j)^T L(:, j)}{p} \right)^2, \quad L(U) = BU, \quad (1)$$

---

<sup>1</sup>This abstract is based on a manuscript not yet submitted anywhere to be considered for publication.

where  $e$  denotes the constant-1 vector<sup>2</sup>. The rotated factor matrix is  $L(U_*)$ .

Methods for computational solution of a factor rotation problem, such as (1), are inherently iterative due to the non-linearity of the objective criterion function. Kaiser's solution method involves iterative sweeps of plane rotations. Every sweep comprises  $m(m - 1)/2$  plane rotations across all pairwise factor axes. The single parameter for each plane rotation can be determined by a single equation derived from (1). Without being confined to the plane rotation sweeps, some methods take the alternative approach, which determines an  $m \times m$  orthogonal matrix  $U$ , with  $m(m - 1)/2$  equations for the orthogonality. More specifically, one may use the Lagrange approach, with up to  $m(m - 1)/2$  multipliers, or deploy the gradient ascending method followed by a projection into the feasible solution space.

The specialized use of Householder transformations for factor matrix sparsification can effectively mitigate or eliminate certain issues present in existing factor rotation methods. For orthogonal factor rotations, a Householder reflection is used in place of a plane-rotation sweep as in Kaiser's method. In comparison to the alternative approach, this new approach implicitly decomposes a general orthogonal matrix  $U$  into orthogonal factors of a compact form. At each step, there are  $(m - 1)$  parameters to be determined, as opposed to just 1 at one extreme with Kaiser's method or  $m(m - 1)/2$  at the other extreme with the alternative approach. For any  $m > 1$ , there is only one Lagrange multiplier. The new factor rotation approach is simple in derivation as well as in implementation. It effectively eliminates the sequencing or scheduling problem within each sweep of plane rotations and resolves the projection issue encountered in gradient-based iteration methods. Additionally, numerical experiments, which will be presented, demonstrate that the new approach is more efficient. For oblique rotations, the use of an oblique Householder transformation is introduced, with similar benefits. Not restricted to Kaiser's criterion, the new method for sparsifying the factor matrix is compatible with and applicable to all factor rotation criteria commonly used in practice.

As a curious application, the simplification criteria and methods are also utilized to sparsify the base vectors for a multi-dimensional Householder reflection.

---

<sup>2</sup>The criterion will be explained in the presentation

# Asynchronous methods meet randomized: Provable convergence rate

Daniel B.Szyld

## Abstract

Asynchronous methods refer to parallel iterative procedures where each process performs its task without waiting for other processes to be completed, i.e., with whatever information it has locally available and with no synchronizations with other processes. For the numerical solution of a general partial differential equation on a domain, Schwarz iterative methods use a decomposition of the domain into two or more (usually overlapping) subdomains. In essence one is introducing new artificial boundary conditions. Thus each process corresponds to a local solve with boundary conditions from the values in the neighboring subdomains.

Using this method as a solver, avoids the pitfall of synchronization required by the inner products in Krylov subspace methods. A scalable method results with either optimized Schwarz or when a coarse grid is added. Numerical results are presented on large three-dimensional problems illustrating the efficiency of asynchronous parallel implementations.

Most theorems show convergence of the asynchronous methods, but not a rate of convergence. In this talk, using the concepts of randomized linear algebra, we present provable convergence rate for the methods for a class of nonsymmetric linear systems. A key element in the new results is the choice of norm for which we can prove convergence of the residual in the expected value sense. Joint work with Andreas Frommer.

# Preconditioning Weak-Constraint 4D-Var: A Parallelisable Implementation in Firedrake

*Jemima M. Tabeart, David Ham, Josh Hope-Collins*

## Abstract

Data assimilation refers to a class of methods which seek to find the most likely state of a dynamical system by combining information from a (numerical) model of the system of interest with measurements of the system [2]. The most mature application of data assimilation is to numerical weather prediction, where large dimensional problems ( $10^9$  dimensional states and  $10^7$  measurements) need to be solved in a very short amount of time. Algorithms also need to be highly parallelisable in order to exploit high performance computing resources available at meteorological centres.

In variational data assimilation methods, a non-linear least squares problem is solved via a Gauss-Newton approach [5]. One computationally expensive component of this implementation consists of approximately solving large linear systems. Preconditioners can help to speed up the convergence of iterative methods, but it can be challenging to design effective and efficient preconditioning methods. This is particularly true for the weak-constraint 4D-Var problem, which accounts for the fact that the numerical model itself is imperfect. Relaxing the assumption of a perfect model increases the size of the state space, but introduces the possibility of using parallel-in-time [3] methods, compared to the strong constraint method where model evaluations must be performed in serial.

For a fixed time window  $[t_0, t_N]$ ,  $\mathbf{x}_i^t \in \mathbb{R}^s$  denotes the true state of a dynamical system of interest at time  $t_i$ , with observations  $\mathbf{y}_i \in \mathbb{R}^{p_i}$  made at times  $t_i$ . Prior information obtained from a numerical model,  $\mathbf{x}_b \in \mathbb{R}^s$ , is then combined with the observation information to find  $\mathbf{x}_i \in \mathbb{R}^s$ , the most likely state of the system at time  $t_i$ . The prior, or background state, is valid at initial time  $t_0$  and can be written as an approximation to the true state via  $\mathbf{x}_b = \mathbf{x}_0^t + \epsilon^b$ . We assume that the background errors  $\epsilon^b \sim \mathcal{N}(0, B)$ . We define a, possibly non-linear, observation operator  $\mathcal{H}_i : \mathbb{R}^s \rightarrow \mathbb{R}^{p_i}$  which maps from state variable space to observation space at time  $t_i$ . Observations are written as  $\mathbf{y}_i = \mathcal{H}_i[\mathbf{x}_i^t] + \epsilon_i \in \mathbb{R}^{p_i}$ , for  $i = 0, 1, \dots, N$ , where the observation error  $\epsilon_i \sim \mathcal{N}(0, R_i)$  for  $R_i \in \mathbb{R}^{p_i \times p_i}$ .

This weak constraint 4D-Var problem then leads to a non-linear objective function of the form:

$$\begin{aligned} J(\mathbf{x}) &= \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^T B^{-1}(\mathbf{x} - \mathbf{x}_0) + \frac{1}{2} \sum_{i=0}^N (\mathbf{y}_i - \mathcal{H}_i[\mathbf{x}_i])^T R_i^{-1}(\mathbf{y}_i - \mathcal{H}_i[\mathbf{x}_i]) \\ &\quad + \frac{1}{2} \sum_{i=1}^N (\mathbf{x}_i - \mathcal{M}_i(\mathbf{x}_{i-1}))^\top Q_i^{-1}(\mathbf{x}_i - \mathcal{M}_i(\mathbf{x}_{i-1})), \end{aligned}$$

Within each outer loop, the inner loop minimises a quadratic objective function to find  $\delta\mathbf{x}^{(l)} \in \mathbb{R}^{s(N+1)}$ , where  $\delta\mathbf{x}^{(l)} = \mathbf{x}^{(l+1)} - \mathbf{x}^{(l)}$ . Writing  $\delta\mathbf{x} = (\delta\mathbf{x}_0^\top, \delta\mathbf{x}_1^\top, \dots, \delta\mathbf{x}_N^\top)^\top$ , the full non-linear observation operator  $\mathcal{H}_i$  (similarly the model operator  $\mathcal{M}_i$ ) is linearised about the current best guess  $\mathbf{x}_i^{(l)}$  to obtain the linearised operator  $H_i^{(l)}$  (respectively  $M_i^{(l)}$ ). The updated initial guess  $\delta\mathbf{x}_0^{(l)}$  is propagated forward between observation times by  $M_i^{(l)}$  to obtain  $\delta\mathbf{x}_{i+1}^{(l)} = M_i^{(l)}\delta\mathbf{x}_i^{(l)}$ .

The aim of the inner loop is to solve the symmetric positive definite system given by

$$\mathbf{S}\delta\mathbf{x} = \mathbf{L}^T \mathbf{D}^{-1} \mathbf{b} + \mathbf{H}^T \mathbf{R}^{-1} \mathbf{d}, \quad \mathbf{S} = \mathbf{L}^T \mathbf{D}^{-1} \mathbf{L} + \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \quad (1)$$

where

$$\mathbf{D} = \begin{pmatrix} B & & \\ & Q_1 & \\ & \ddots & \\ & & Q_N \end{pmatrix}, \quad \mathbf{L} = \begin{pmatrix} I & & & \\ -M_1 & I & & \\ & \ddots & \ddots & \\ & & -M_N & I \end{pmatrix}, \quad \mathbf{R} = \begin{pmatrix} R_0 & & & \\ & R_1 & & \\ & & \ddots & \\ & & & R_N \end{pmatrix}, \quad \mathbf{H} = \begin{pmatrix} H_0 & & & \\ & H_1 & & \\ & & \ddots & \\ & & & H_N \end{pmatrix},$$

with  $M_i, H_i$  being the linearisations of  $\mathcal{M}_i$  and  $\mathcal{H}_i$  about the current solution.

However, the primal formulation, (1), has limited potential for acceleration via preconditioning approaches. In particular, it is difficult to exploit the inherent parallelism in the forward problem when designing preconditioners. Recent work has focused on a reformulation of the linearised objective function as a saddle point system [6] which takes the form

$$\begin{pmatrix} \mathbf{D} & \mathbf{0} & \mathbf{L} \\ \mathbf{0} & \mathbf{R} & \mathbf{H} \\ \mathbf{L}^\top & \mathbf{H}^\top & \mathbf{0} \end{pmatrix} \begin{pmatrix} \delta\boldsymbol{\eta} \\ \delta\nu \\ \delta\mathbf{x} \end{pmatrix} = \begin{pmatrix} \mathbf{b} \\ \mathbf{d} \\ \mathbf{0} \end{pmatrix}. \quad (2)$$

A number of approaches to precondition the model term,  $\mathbf{L}$ , have been proposed (see e.g. [6, 7, 12, 11]), many of which impose parallel-in-time structure on  $\widehat{\mathbf{L}}^{-1} \approx \mathbf{L}^{-1}$ . However, testing and comparing these new preconditioners in realistic frameworks is not straightforward. Variational data assimilation requires access to linearised model and adjoint operators. The time cost of implementing these for a new problem means that researchers often test their new approaches on a limited number of toy problems, such as the Lorenz 96 problem [9] or the shallow water equations. If available (and accessible to the researchers), the next step is a full scale implementation within an operational code, meaning that even in the best case there is a gap in test problems. Data assimilation methods are also increasingly being applied to other dynamical systems, and the properties of the usual toy models that make them appropriate for weather applications may no longer be desirable or relevant for other applications.

In this project we integrate both weak- and strong-constraint 4D-Var algorithms within the Firedrake [8] and PETSc [1] frameworks, for both the primal (1) and saddle point (2) problems. Firedrake is an automated system for the solution of partial differential equations using the finite element method. In particular, this means that tangent linear and adjoint operators are available automatically, significantly reducing the implementational burden of applying variational data assimilation methods to new models. For the user, the cost of setting up the 4D-Var system is not substantially higher than the cost of a single run of the forward model  $\mathcal{M}_i$  and application of the observation operators  $\mathcal{H}_i$ . This implementation is also done in parallel.

In this talk I will present the integration of variational data assimilation problems within Firedrake, and demonstrate how this ‘plug-and-play’ approach allows users to focus on the numerical linear algebra aspects of their problem rather than the implementation of test problems. In particular I will present a theoretical and practical comparison of existing and new preconditioners for the model term,  $\mathbf{L}$ , including the block Toeplitz approaches of [6, 11], block diagonal approximations as studied in [12], and block  $(\alpha)$ -circulant preconditioners as used in the all-at-once setting (see e.g [10, 4]).

## References

- [1] S. Balay, S. Abhyankar, M. F. Adams, S. Benson, J. Brown, P. Brune, K. Buschelman, E.

Constantinescu, L. Dalcin, A. Dener, V. Eijkhout, J. Faibusowitsch, W. D. Gropp, V. Hapla, T. Isaac, P. Jolivet, D. Karpeev, D. Kaushik, M. G. Knepley, F. Kong, S. Kruger, D. A. May, L. Curfman McInnes, R. Tran Mills, L. Mitchell, T. Munson, J. E. Roman, K. Rupp, P. Sanan, J. Sarich, B. F. Smith, H. Suh, S. Zampini, H. Zhang, H. Zhang and J. Zhang, *PETSc/TAO Users Manual*, Argonne National Laboratory, ANL-21/39 - Revision 3.22, = 10.2172/2205494, 2024.

- [2] A. Carrassai, M. Bocquet, Marc, L. Bertino, and G. Evensen. *Data assimilation in the geosciences: An overview of methods, issues, and perspectives*. Wiley Interdisciplinary Reviews: Climate Change, 9 (5): e535, 2018.
- [3] M. J. Gander, *50 Years of Time Parallel Time Integration* Multiple Shooting and Time Domain Decomposition Methods: MuS-TDD, Heidelberg, May 6-8, 2013. Cham: Springer International Publishing, 69-113, 2015.
- [4] M. J. Gander, and S. L. Wu. *Convergence analysis of a periodic-like waveform relaxation method for initial-value problems via the diagonalization technique*. Numerische Mathematik, 143, 489-527, 2019.
- [5] S. Gratton, A. S. Lawless, and N. K. Nichols. *Approximate Gauss-Newton Methods for Nonlinear Least Squares Problems*. SIAM Journal on Optimization, 18(1):106–132, 2007.
- [6] M. Fisher and S. Gürol. *Parallelization in the time dimension of four-dimensional variational data assimilation*. Quarterly Journal of the Royal Meteorological Society, 143(703):1136–1147, 2017.
- [7] M. A. Freitag and D. L. H. Green. *A low-rank approach to the solution of weak constraint variational data assimilation problems*. Journal of Computational Physics, 357:263–281, 2018.
- [8] D. A. Ham, P. H. J. Kelly, L. Mitchell, C. J. Cotter, R. C. Kirby, K. Sagiyyama, N. Bouziani, S. Vorderwuelbecke, T. J. Gregory, J. Betteridge, D. R. Shapero, R. W. Nixon-Hill, C. J. Ward, P. E. Farrell, P. D. Brubeck, I. Marsden, T. H. Gibson, M. Homolya, T. Sun, A. T. T. McRae, F. Luporini, A. Gregory, M. Lange, S. W. Funke, F. Rathgeber, G.-T. Bercea and G. R. Markall. *Firedrake User Manual*. First edition, doi:10.25561/104839, 2023.
- [9] E. N. Lorenz, *Predictability: a problem partly solved*, in Seminar on Predictability, 4–8 September 1995, ECMWF, Reading, 1995.
- [10] E. McDonald, J. Pestana, and A. J. Wathen. *Preconditioning and Iterative Solution of All-at-Once Systems for Evolutionary Partial Differential Equations*. SIAM J. Sci. Comput., 40, 2018.
- [11] D. Palitta and J. M. Taboart. *Stein-based preconditioners for weak-constraint 4D-Var*. Journal of Computational Physics, 482:112068, 2023.
- [12] J. M. Taboart and J. W. Pearson. *Saddle point preconditioners for weak-constraint 4D-Var*. ETNA - Electronic Transactions on Numerical Analysis, 60:197–220, 2024.

# Computing Accurate Eigenvalues of Symmetric Matrices With a Mixed Precision Jacobi Algorithm

*Nicholas J. Higham, Françoise Tisseur, Marcus Webb, Zhengbo Zhou*

## Abstract

Modern hardware increasingly supports not only single and double precisions, but also half and quadruple precisions. These precisions provide new opportunities to considerably accelerate linear algebra computations while maintaining numerical stability and accuracy. Efforts on developing mixed precision algorithms in the numerical linear algebra and high performance computing communities have mainly focussed on linear systems and least squares problems. Eigenvalue problems are considerably more challenging to solve and have a larger solution space that cannot be computed in a finite number of steps [5].

There are two classes of algorithms for symmetric eigenproblems: (i) those that work directly on the matrix, such as the Jacobi algorithm and the QR-based Dynamically Weighted Halley (QDWH-eig) algorithm and (ii) those that reduce the matrix to tridiagonal form in a finite number of steps and then employ an iterative scheme to compute all or just part of the eigenvalues and/or the eigenvectors, such as bisection and inverse iteration (BI), the QR algorithm, and the divide-and-conquer algorithm (DC). All these algorithms have pros and cons. DC and the method of multiple relatively robust representations (MR), which is a sophisticated variant of inverse iteration, are generally much faster than QR and BI on large matrices, with MR performing the fewest floating point operations but at a lower MFLOPS rate than DC. The latter and QR are the most accurate algorithms with observed accuracy  $O(\sqrt{n}u)$ , where  $u$  is the working precision,  $n$  the size of the matrix, and accuracy is measured in terms of scaled residual norms and loss of orthogonality for the eigenvectors [1]. None of these eigensolvers exploits the low precisions available in modern hardware.

A key question is how can we exploit access to multiple precisions arithmetic to accelerate symmetric eigensolvers while maintaining numerical stability and accuracy?

In terms of arithmetic cost, solving a symmetric eigenvalue problem is about 27 times more expensive than solving a symmetric positive definite linear system. Unlike for linear systems for which the  $O(n^3)$  part of the computation can be performed at low precision and the  $n$ -dimensional solution refined at working precision in  $O(n^2)$  operations, it can be shown that for the eigenvalue problem, some of the  $O(n^3)$  operations need to be performed in the working precision if one hopes to maintain numerical stability and achieve accuracy. So to gain any speedup, these should be BLAS 3 operations, i.e., highly optimized matrix-matrix multiplies. Modern architectures execute matrix multiplies of large size  $n$  at least 18 faster than symmetric eigensolvers on the same size matrices. Low precision arithmetic can be used to preprocess or to precondition the eigenproblem to allow for a faster solution.

In this talk we concentrate on symmetric positive definite matrices  $A \in \mathbb{R}^{n \times n}$  and consider a mixed precision preconditioned Jacobi algorithm that uses three precisions  $u_h < u < u_\ell$ . The preconditioner  $\tilde{Q}$  is an approximate eigenvector matrix that is efficiently computed using a combination of low and working precisions. Zhang and Bai [7] and Zhou [8] suggested to compute an eigenvector matrix at low precision and then orthogonalize it to working precision so that

$$\|\tilde{Q}^T \tilde{Q} - I\|_2 \leq p_1 u < 1, \quad (1)$$

where  $p_1$  is a low degree polynomial in  $n$ . It is essential that the preconditioner  $\tilde{Q}$  satisfies (1) to ensure that the eigenvectors returned by the mixed precision preconditioned Jacobi algorithm are orthogonal to working precision  $u$ . We discuss several alternative efficient ways to construct such preconditioner and prove it reduces the off-diagonal entries of  $A$  to a level determined by the chosen low precision  $u_\ell$  so that the initial slow convergence phase of the Jacobi algorithm can be skipped.

Demmel and Veselić [2] showed that the eigenvalues computed by the Jacobi algorithm with stopping criterion  $|a_{ij}| \leq \sqrt{a_{ii}a_{jj}}$  for all  $i, j$  satisfy

$$\frac{|\lambda_i(A) - \tilde{\lambda}_i(A)|}{|\lambda_i(A)|} \leq p(n) u \kappa_2^S(A), \quad (2)$$

where  $\lambda_i(A)$  and  $\tilde{\lambda}_i(A)$  denote the  $i$ th largest exact and computed eigenvalue of  $A$ ,  $p(n)$  is a low degree polynomial and  $u$  is the working precision. Here  $\kappa_2^S(A)$  is the *scaled condition number* of  $A$  defined by

$$\kappa_2^S(A) = \kappa_2(DAD), \quad D = \text{diag}(a_{ii}^{-1/2}),$$

where  $\kappa_2(B) = \lambda_1(B)/\lambda_n(B)$ . For the QR and DC algorithms, the relative error is bounded by  $n^{1/2}p(n)u\kappa_2(A)$  so when  $\kappa_2(DAD) \ll \kappa_2(A)$ , the Jacobi algorithm can produce much more accurate approximations to the smaller eigenvalues than QR or DC algorithms.

Malyshev [6] and Drygalla [3, 4] suggest that preconditioning the matrix at a precision  $u_h$  higher than the working precision  $u$  improves the accuracy of the spectral decomposition computed by the preconditioned Jacobi algorithm. However, Malyshev only discuss the backward error and Drygalla only claims the high accuracy property without proving it. Let us denote by  $\tilde{A}$  and  $\tilde{A}_{\text{comp}}$  the product  $\tilde{Q}^T A \tilde{Q}$  computed in exact and floating point arithmetic, respectively. We prove under mild assumptions that the relative errors in the computed eigenvalues are proportional to  $u\kappa_2^S(\tilde{A}_{\text{comp}})$  and  $u\kappa_2^S(\tilde{A})$  instead of  $u\kappa_2^S(A)$  which appears in (2). Moreover, we prove that if  $\tilde{A}$  is  $\theta$ -scaled diagonally dominant, i.e.,  $\theta = \|\tilde{D}\tilde{A}\tilde{D}\|_2 < 1$  then the scaled condition numbers  $\kappa_2^S(\tilde{A})$  and  $\kappa_2^S(\tilde{A}_{\text{comp}})$  are of order 1. Hence, all the eigenvalues are computed to high relative accuracy. We remark that any preconditioner  $\tilde{Q}$  such that  $\text{off}(\tilde{A})/\min_i(\tilde{a}_{ii}) < 1$ , where  $\text{off}(\tilde{A}) = (\sum_{i \neq j} \tilde{a}_{ij}^2)^{1/2}$ , yields an  $\tilde{A}$  that is scaled diagonally dominant. For a preconditioned matrix  $\tilde{A}$  that is not scaled diagonally dominant, we use a result by Demmel and Veselić [2, Prop. 6.2] to argue that if  $\text{off}(\tilde{A})$  is sufficiently small so that we can treat the diagonals of  $\tilde{A}$  as its approximate eigenvalues, the scaled condition numbers  $\kappa_2^S(\tilde{A}_{\text{comp}})$  and  $\kappa_2^S(\tilde{A})$  are significantly smaller than  $\kappa_2^S(A)$ .

Finally, we present numerical results to support our theoretical analysis.

## References

- [1] J. W. Demmel, O. A. Marques, B. N. Parlett, and C. Vömel. Performance and accuracy of LAPACK's symmetric tridiagonal eigensolvers. *SIAM J. Sci. Comput.*, 30(3):1508–1526, 2008.
- [2] J. W. Demmel and K. Veselić. Jacobi's method is more accurate than QR. *SIAM J. Matrix Anal. Appl.*, 13(4):1204–1245, 1992.
- [3] V. Drygalla. Extra precise preconditioning for non-Hermitian eigenvalue problems. *Proc. Appl. Math. Mech.*, 6(1):713–714, Dec. 2006.
- [4] V. Drygalla. Exploiting mixed precision for computing eigenvalues of symmetric matrices and singular values. *Proc. Appl. Math. Mech.*, 8(1):10809–10810, Dec. 2008.

- [5] N. J. Higham and T. Mary. Mixed precision algorithms in numerical linear algebra. *Acta Numerica*, 31:347–414, May 2022.
- [6] A. N. Malyshev. On iterative refinement for the spectral decomposition of symmetric matrices. Research Report 1651, INRIA/IRISA, Unité de Recherche, Rennes, France, 1992.
- [7] Z. Zhang and Z.-J. Bai. A mixed precision Jacobi method for the symmetric eigenvalue problem. Technical Report arXiv:2211.03339v1, 2022.
- [8] Z. Zhou. A mixed-precision eigensolver based on the Jacobi algorithm. M.Sc. Thesis, The University of Manchester, Manchester, UK, Sept. 2022.

# Quantum Computing in MATLAB

*Matt Bowring, Steve Grikschat, Paul Kerr-Delworth, Patrick Quillen, Christine Tobler*

## Abstract

We present the new MATLAB Support Package for Quantum Computing, which provides utilities to build, simulate, and visualize quantum circuits. Additionally, it is possible to connect to hardware providers and run circuits on their quantum computers.

The capabilities of this software package include

- Constructing a circuit from a set of quantum gates, which are applied to specific qubits. In addition to a set of standard simple gates, more complex gates are available: `mcxGate` [1]; `initGate`, `unitaryGate`, `ucrxGate`, `ucryGate`, and `ucrzGate` [2, 3].
- Verifying the quantum algorithm by simulating it on the local computer or sending it to a remote simulator through cloud services.
- Executing the circuit by connecting to quantum computing hardware through cloud services (specifically, IBM Qiskit Runtime Services and Amazon Web Services). This involves sending hardware-specific quantum assembly (OpenQASM) code to these services.
- Creating quadratic unconstrained binary optimization (QUBO) problems and solving them on the local computer using Tabu search [4].

This package enables the prototyping of quantum algorithms that have applications in optimization, scenario simulation, machine learning, as well as chemistry and material simulations.

## References

- [1] A. Barenco et al. Elementary Gates for Quantum Computation. *Physical Review A*, 52(5):3457–3467, 1995.
- [2] V. V. Shende, S. S. Bullock, and I. L. Markov. Synthesis of Quantum-Logic Circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 25(6):1000–1010, 2006.
- [3] D. Camps and R. Van Beeumen. FABLE: Fast Approximate Quantum Circuits for Block Encodings. *2022 IEEE International Conference on Quantum Computing and Engineering (QCE)*, 104–113. 2022.
- [4] G. Palubeckis. Iterated Tabu Search for the Unconstrained Binary Quadratic Optimization Problem. *Informatica* 17(2), 279–296, 2006.

# The Quest for a Numerically Stable Multivariate Polynomial Rootfinder

*Alex Townsend, Emil Graf*

## Abstract

**Introduction.** The search for a robust, global, and numerically stable algorithm for solving multivariate polynomial systems has persisted for decades [2, 3, 5, 6, 7, 8, 9]. The goal is to compute all the solutions to zero-dimensional polynomial systems of the form:

$$\begin{pmatrix} p_1(x_1, \dots, x_d) \\ \vdots \\ p_d(x_1, \dots, x_d) \end{pmatrix} = \mathbf{0}, \quad (1)$$

where  $d \geq 2$  and  $p_1, \dots, p_d$  are polynomials in  $x_1, \dots, x_d$  with complex coefficients. All the promising solvers are based on an elegant approach of converting the multivariate rootfinding problem in (1) into one or more eigenproblems. At first this approach appears to be a practitioner's dream as a difficult rootfinding problem can be solved by the robust QR or QZ algorithm. For this reason, these methods have received considerable research attention from the scientific computing community. However, we are currently stuck waiting for new ideas to emerge from algebraic geometry or hoping for novel structured eigensolvers from numerical linear algebra.

A popular class of techniques known as hidden variable resultant methods are notoriously difficult—and maybe impossible—to make numerically robust [6]. Naive implementations are plagued with unwanted spurious solutions, inaccurate roots, and miss zeros. In this talk, we will discuss the ongoing quest for a numerically stable rootfinder using Sylvester resultants, Gröbner bases, Möller–Stetter matrices, and Macaulay resultants. Our focus will be on understanding the sources of the instability in these approaches, in the hope that they can be circumvented.

**Motivation for Eigenvalue-Based Approaches.** Given a well-conditioned rootfinding problem, we would like to derive a stable algorithm to solve it. Roughly speaking, an algorithm is stable if it computes an accurate solution to well-conditioned problems. The search for a stable algorithm for multivariate polynomial rootfinding is motivated by the existence of stable algorithms for many related problems. All the univariate problems, such as eigenproblems, univariate polynomial rootfinding, and matrix polynomial eigenproblems, have stable algorithms. Likewise, there are stable algorithms to solve linear systems of the form  $A\mathbf{x} = \mathbf{b}$ , which are multivariate.

For univariate rootfinding, instead of solving a rootfinding problem directly, one can first construct an eigenproblem whose eigenvalues match the desired roots. The companion matrix of a polynomial  $p(x)$  is an example of this, as its characteristic polynomial is  $p$ , so its eigenvalues are the roots of  $p$ . One can solve the companion eigenproblem using an eigensolver, which is one of the most reliable algorithms in numerical linear algebra. For roots in  $[-1, 1]$ , a provably stable algorithm for univariate polynomial rootfinding is based on the colleague matrix [4]. For multivariable rootfinding, we attempt the same conversion, i.e., we try to convert (1) into one or more generalized eigenvalue problems (GEPs). For polynomial systems in (1) in  $d$  variables, one usually constructs  $d$  GEPs, the eigenvalues of which give the coordinates of each root. The Macaulay resultant method is an exception as it constructs a single GEP and extracts the roots from the eigenvectors, not eigenvalues. Analogously to the univariate case, one hopes that the eigenproblems are as well-conditioned as the original rootfinding problem. Unfortunately, this is not always the case.

Gröbner bases, Möller–Stetter matrices, rational univariate reductions, multiparameter eigenvalue problems, and Macaulay resultants all convert (1) into one or more GEPs, either directly or by

way of a univariate rootfinding problem. For each method, we show that either a constructed eigenproblem or an intermediate univariate rootfinding problem can be more ill-conditioned than the original rootfinding problem by a factor that is exponentially large in  $d$ .

**A devastating example.** The analysis of the instability of Sylvester and Cayley resultant method appears in [5, 6] and studied the following “devastating” polynomial rootfinding problem.

**Example 1** Let  $Q$  be a  $d \times d$  orthogonal matrix,  $\sigma > 0$ , and consider (1) with

$$p_i(x_1, \dots, x_d) = x_i^2 + \sigma \sum_{j=1}^d q_{ij} x_j, \quad 1 \leq i \leq d,$$

where  $q_{ij}$  is the  $(i, j)$  entry of  $Q$ . The system has a root at  $(0, \dots, 0)$ .

By a conditioning analysis, one should expect to find the root at  $(0, \dots, 0)$  to within  $\approx \sigma u$ , where  $u$  is the unit roundoff on a computer. However, it has been shown that Sylvester resultants can only achieve  $\approx \sigma^{-2}u$  when  $d = 2$  and Cayley can only achieve  $\approx \sigma^{-d}u$  [6]. The eigenproblems constructed by these methods can be far more sensitive to perturbations than the original rootfinding problem, which is a hallmark of an unstable algorithm. Similar examples show that Gröbner bases, Möller–Stetter matrices, rational univariate reductions, multiparameter eigenvalue problems, and Macaulay resultants can also generate highly ill-conditioned eigenproblems.

**Theoretical results supported by numerical experiments.** One might be hopeful that these worst-case examples are rarely and never realized in practice. Unfortunately, this is not the case in practice. We regularly observe the ill-conditioning in numerical experiments, causing solvers to generate spurious solutions, miss roots, or compute them inaccurately. There is a small glimmer of hope in some new approaches that construct structured eigenproblems and solve them with eigensolvers that respect that structure [1].

We hope that our work inspires new approaches that circumvent the limitations of current techniques and provide robust solutions to the fundamental multivariate polynomial rootfinding problem.

## References

- [1] H. He, D. Kressner, and B. Plestenjak. Randomized methods for computing joint eigenvalues, with applications to multiparameter eigenvalue problems and root finding. *arXiv preprint arXiv:2409.00500*, 2024.
- [2] G. F. Jónsson and S. A. Vavasis. Accurate solution of polynomial equations using Macaulay resultant matrices. *Math. Comp.*, 74(249):221–262, 2005.
- [3] B. Mourrain, S. Telen, and M. Van Barel. Truncated normal forms for solving polynomial systems: generalized and efficient algorithms. *J. Symbolic Comput.*, 102:63–85, 2021.
- [4] Y. Nakatsukasa and V. Noferini. On the stability of computing polynomial roots via confederate linearizations. *Math. Comp.*, 85(301):2391–2425, 2016.
- [5] Y. Nakatsukasa, V. Noferini, and A. Townsend. Computing the common zeros of two bivariate functions via Bézout resultants. *Numer. Math.*, 129(1):181–209, 2015.
- [6] V. Noferini and A. Townsend. Numerical instability of resultant methods for multidimensional rootfinding. *SIAM J. Numer. Anal.*, 54(2):719–743, 2016.
- [7] B. Plestenjak and M. E. Hochstenbach. Roots of bivariate polynomial systems via determinantal representations. *SIAM J. Sci. Comput.*, 38(2):A765–A788, 2016.
- [8] F. Rouillier. Solving zero-dimensional systems through the rational univariate representation. *Appl. Algebra Engrg. Comm. Comput.*, 9(5):433–461, 1999.
- [9] S. Telen, B. Mourrain, and M. Van Barel. Solving polynomial systems via truncated normal forms. *SIAM J. Matrix Anal. Appl.*, 39(3):1421–1447, 2018.

# From Zolotarev Problems in Linear Algebra to a New Approach to Quadrature

*Lloyd N. Trefethen*

## Abstract

Beckermann, Townsend, Wilber, and Rubin have recently drawn attention to the importance of the classical Zolotarev rational approximation problems, in their generalized forms as analyzed among others by Gonchar, Starke, Istrate, Thiran, Druskin, Knizhnerman, and Simoncini, to large-scale linear algebra problems including ADI iteration, Lyapunov and Sylvester equations, and low-rank approximation [2, 3, 5, 12, 13, 18]. The paper by Beckermann and Townsend on this topic was selected for *SIGEST* as an exceptional contribution of recent years in the *SIAM Journal on Matrix Analysis and its Applications*.

Specifically, at issue here are what are traditionally called the third and fourth Zolotarev problems, for which the following names are perhaps more memorable. The *Zolotarev Ratio Problem* is the problem of finding a rational function of prescribed degree  $n$  with a smallest possible ratio of its maximal size on one set  $E$  in the complex plane to its minimal size on another set  $F$ . The *Zolotarev Sign Problem* is the problem of finding a rational function with  $r \approx -1$  on  $E$  and  $r \approx 1$  on  $F$ . In linear algebra applications,  $E$  and  $F$  are related to the spectra of large matrices.

Istrate and Thiran showed that these two problems are mathematically equivalent [7], but no reliable method has been available for solving either of them numerically. In the first half of this talk I will present such a method, developed jointly with Heather Wilber [17]. In a fraction of a second on a laptop for typical examples, we can now compute numerical solutions to both Zolotarev problems to several digits of accuracy. Fourteen examples are displayed graphically in [17], which can be found online at [https://people.maths.ox.ac.uk/trefethen/trefethen\\_wilber.pdf](https://people.maths.ox.ac.uk/trefethen/trefethen_wilber.pdf). These computations are based on the AAA and AAA-Lawson algorithms for best rational approximation [9, 10], but until 2024, these algorithms were not successful for Zolotarev problems. Wilber and I found we had to modify them for this study, and we acknowledge a crucial contribution here from Yuji Nakatsukasa.

All this falls within a familiar frame of numerical linear algebra problems of current interest. However, investigating these new algorithms has led to something new that may eventually be of deeper importance in numerical analysis, and this will be the subject of the second half of my talk. It has long been known that there are connections between rational approximation and quadrature, and these connections have been exploited in various ways, for example in the FEAST eigensolver [1, 5, 11]. Building on the new Zolotarev algorithms, we have found a general framework of this kind that appears to encompass many problems. (It may also ultimately lead to a fulfillment of a dream a few of us have had of achieving a truly numerical realization of the theory of hyperfunctions.)

I very much regret that Householder abstracts do not permit the inclusion of graphics, for a few pictures show strikingly the new connections that are emerging. (This work is being developed in unpublished memos of mine, which are not yet in a form for public circulation.) Here are four instances where rational approximation gives new insight into old methods and an easy route to methods for quadrature:

1. *Trapezoidal rule on the unit circle.* It has been known since Lyness and Delves in 1967 that the trapezoidal rule applied in roots of unity gives exponential accuracy for functions analytic on the unit circle. Unexpectedly, this morphs into a Zolotarev sign problem if we consider the function

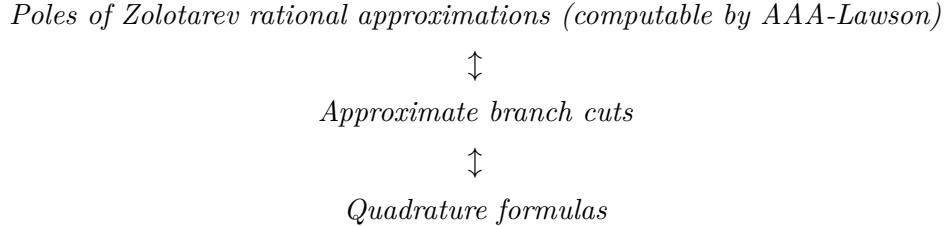
$\phi(z)$  defined by  $\phi(z) = 0$  for  $|z| > 1$  and  $\phi(z) = -1$  for  $|z| < 1$ . Rational approximations to  $\phi$  put a string of poles along the unit circle, and via residue calculus, these can be converted to trapezoidal-style quadrature rules. (The nodes are the poles, and the weights are the residues.) Exponential convergence of the rational approximations becomes equivalent to exponential convergence of the quadrature formulas.

2. *Inverse Laplace transforms and Hankel contours.* Another established topic is Bromwich or Fourier-Mellin or inverse-Laplace-transform quadrature involving an exponential kernel over Hankel contours in the complex plane. Here, following [16], we consider rational approximations to  $e^x$  for  $x \in (-\infty, 0]$ . The poles of near-best approximations line up along Hankel contours curving around  $(-\infty, 0]$ , leading to near-optimal quadrature formulas.

3. *Computing functions of matrices like  $A^\alpha$  and  $\log(A)$ .* The “three Nicks paper” [6] applied conformal maps to derive exponentially accurate quadrature formulas for computing functions of matrices, which have had impact for example in electronic structure calculations [8]. The methods required tricky derivation via elliptic functions, but it turns out that on-the-fly Zolotarev approximation can achieve the same results. This time, the approximation problem consists of finding a rational function  $r$  with  $r(x) \approx 1$  for  $x \in [m, M]$  (an interval containing the spectrum of  $A$ ) and  $r(x) \approx 0$  for  $x \in (-\infty, 0]$ . In the rational approximation framework, it is an easy matter to derive variant formulas for cases where, say,  $A$  is a nonsymmetric matrix with spectrum in a complex domain rather than an interval.

4. *Gauss-Legendre and other quadrature formulas on  $[-1, 1]$ .* Ordinary Gauss-Legendre quadrature and other formulas for integration over  $[-1, 1]$  can be given the same kind of derivation, which is related to how Gauss originally derived his method in 1814 and was later exploited by Takahasi and Mori [14, 15]. Now the rational approximation problem is to find a function  $r(z)$  that closely approximates  $\log((z+1)/(z-1))$  on a Bernstein ellipse enclosing  $[-1, 1]$ .

In summary, these are the connections that are emerging:



## References

- [1] A. P. AUSTIN AND L. N. TREFETHEN, *Computing eigenvalues of real symmetric matrices with rational filters in real arithmetic*, SIAM J. Sci. Comput., 37 (2015), A1365–A1387.
- [2] B. BECKERMANN AND A. TOWNSEND, *Bounds on the singular values of matrices with displacement structure*, SIAM Rev., 61 (2019), 319–344.
- [3] V. DRUSKIN, L. KNIZHNERMAN, AND V. SIMONCINI, *Analysis of the rational Krylov subspace and ADI methods for solving the Lyapunov equation*, SIAM J. Numer. Anal., 49 (2011), 1875–1898.
- [4] A. A. GONCHAR, *Zolotarev problems connected with rational functions*, Math. USSR-Sb., 7 (1969), 623–635.

- [5] S. GÜTTEL, E. POLIZZI, P. T. P. TANG, AND G. VIAUD, *Zolotarev quadrature rules and load balancing for the FEAST eigensolver*, SIAM J. Sci. Comput., 37 (2015), A2100–A2122.
- [6] N. HALE, N. J. HIGHAM, AND L. N. TREFETHEN, *Computing  $A^\alpha$ ,  $\log(A)$ , and related matrix functions by contour integrals*, SIAM J. Numer. Anal., 46 (2008), 2505–2523.
- [7] M.-P. ISTACE AND J.-P. THIRAN, *On the third and fourth Zolotarev problems in the complex plane*, SIAM J. Numer. Anal., 32 (1995), 249–259.
- [8] L. LIN, J. LU, L. YING, R. CAR, AND W. E, *Fast algorithm for extracting the diagonal of the inverse matrix with application to the electronic structure analysis of metallic systems*, Commun. Math. Sci., 130 (2009), 204511.
- [9] Y. NAKATSUKASA, O. SÈTE, AND L. N. TREFETHEN, *The AAA algorithm for rational approximation*, SIAM J. Sci. Comput., 40 (2018), A1494–A1522.
- [10] Y. NAKATSUKASA AND L. N. TREFETHEN, *An algorithm for real and complex rational minimax approximation*, SIAM J. Sci. Comput., 42 (2020), A3157–A3179.
- [11] E. POLIZZI, *Density-matrix-based algorithm for solving eigenvalue problems*, Phys. Rev. B, 79 (2009), 115112.
- [12] D. RUBIN, A. TOWNSEND, AND H. WILBER, *Bounding Zolotarev numbers using Faber rational functions*, Constr. Approx., 56 (2022), 207–232.
- [13] G. STARKE, *Optimal alternating direction implicit parameters for nonsymmetric systems of linear equations*, SIAM J. Numer. Anal., 28 (1991), 1431–1445.
- [14] H. TAKAHASI AND M. MORI, *Estimation of errors in the numerical quadrature of analytic functions*, Appl. Anal., 1 (1971), 201–229.
- [15] L. N. TREFETHEN, *Is Gauss quadrature better than Clenshaw-Curtis?*, SIAM Rev., 50 (2008), 67–87.
- [16] L. N. TREFETHEN, J. A. C. WEIDEMAN, AND T. SCHMELZER, *Talbot quadratures and rational approximations*, BIT Numer. Math., 46 (2006), 653–670.
- [17] L. N. TREFETHEN AND H. D. WILBER, *Computation of Zolotarev rational functions*, SIAM J. Sci. Comp., submitted. ([https://people.maths.ox.ac.uk/trefethen/trefethen\\_wilber.pdf](https://people.maths.ox.ac.uk/trefethen/trefethen_wilber.pdf))
- [18] H. D. WILBER, *Computing Numerically with Rational Functions*, PhD thesis, Dept. of Mathematics, Cornell U., 2021.

# On the Unitary Block-Diagonalisation of General Matrices and Applications

*Frank Uhlig*

Abstract

## Abstract

We study new matrix based computations for a recent cluster of extraordinary results in six distinct branches of mathematics that are inter-connected in multiple multi-dimensional ways. The first quoted paper deals with fractional ordinary differential equations and the proportional secting method for accelerating terminal value problems therein by a factor of around 8. Then we study how a century old and previously unsolved quantum physics problem can be solved by using the hermitean Johnson  $\mathcal{F}(t)$  function of field of values computations. In quantum physics terms, this solution makes an assessment of our Chemical Element Tables finally possible after 100 + years of not knowing. Then we study accurate and fast computations of field of values boundary curves, even for decomposable matrices. This was impossible before and has been abandoned for several years now. To solve the unitary block decomposition problem for general square matrices we use a discretized predictive Zhang Neural Network method for the resulting Johnson block  $\mathcal{F}_j(t)$  field of values functions. Overall the computational methods in this cluster are all conditionally stable and none is just backward stable. They all give us highly accurate results such as adapted ZNN methods for matrix flow  $A(t)$  problems that find nonsingular static matrix  $A$  symmetrizers with small condition numbers for the first time. A detailed survey of Zhang Neural Networks details their seven step set-up process for the first time, giving ten matrix flow example derivations of this new process. B O X

**Keywords:** math history, numerical analysis, fractional ODEs, shooting methods, proportional secting, Linear algebra, unitary block-decomposition, invariant subspace theory, quantum physics, time-varying matrix problem, conditionally stable algorithm, Johnson matrix flow, neural network, zeroing neural network, discretized ZNN algorithm, field of values, matrix flows, time-varying numerical algorithm, ZNN set-up, predictive numerical method, matrix symmetrizer

**AMS :** 65F10, 65F15, 65F30, 65J10, 65L10, 34A08, 35A08, 15A18, 15A21, 15A23, 15A60, 15B57, 81Q10

## An Introduction – of sorts

For general complex or real 1-parameter matrix flows  $A(t)_{n,n}$  or for static matrices  $A$  this paper considers ways to decompose matrix flows  $A(t)$  or single matrices  $A_{n,n}$  globally via one constant hermitean matrix similarity  $C_{n,n}$  as

$$A(t) = C^{-1} \cdot \text{diag}(A_1(t), \dots, A_\ell(t)) \cdot C$$

or

$$A = C^{-1} \cdot \text{diag}(A_1, \dots, A_\ell) \cdot C.$$

Here each diagonal block  $A_k(t)$  or  $A_k$  is square and their number  $\ell$  exceeds 1 – if this is possible.

The theory behind our proposed algorithm is elementary and uses the concept of invariant subspaces for the MATLAB `eig` computed ‘eigenvectors’ of another associated flow matrix  $B(t_a)$  to find the coarsest simultaneous block structure for all flow matrices  $B(t_b)$  and consequently block-diagonalizes the given matrix flow  $A(t)$  or the given static matrix  $A$  itself.

The method works in  $O(n^2)$  time for all matrices  $A_{n,n}$  and all matrix flows  $A(t)$ , be they real or complex, normal, with Jordan structures or repeated eigenvalues, and differentiable, continuous, or discontinuous.

We aim to discover **unitarily diagonal-block decomposable matrices and flows** from sensor given data. For unitarily block-diagonalisable  $A$  or  $A(t)$ , the complexity of their numerical treatment decreases swiftly for  $O(n^3)$  matrix processes when working on each of their diagonal blocks separately.

### *The proof : The Unitary Block-Decompositions and Fast and Accurate Field of Values Boundary Computations of all Square Matrices*

The unitary block decomposition of matrix flows or single square matrices has never been resolved by algebraic means in 100 + years. Our new way of testing and establishing unitary decomposability of a matrix flow or a static matrix hinges on a numerical algorithm that decides the lay of the near zero entries and of the sizable ones in the hermitean Johnson matrix flow

$$F(t) = \cos(t)H + \sin(t)K$$

for  $0 \leq t \leq 2\pi$ .

[ We assume today that all matrices and matrix flows are real in this talk. ]

The algorithm starts from two linear independent Johnson flow matrices  $F(t_1)$  and  $F(t_2)$ . We form the 0-1 logic matrices  $Flogic(t_1)$  and  $Flogic(t_2)$  in Matlab’s `spy` function and assemble the normalized eigenvectors of  $F(t_1)$  in  $V'(t_1)$  so that adjacent rows in  $Flogic(t_1)$  have equal 0 and 1 patterned blocks.

We reorder the rows of  $Flogic(t_2)$  so that they have the same 0-1 pattern as  $Flogic(t_1)$  by using a combinatorial algorithm that establishes a joint proper unitary block decomposition of both logic 0-1 spy pattern matrices.

Going down from the top block leads to a joint unitary block diagonal structure for all  $F(t)$  with  $t \in [0, 2\pi]$ . Since  $F(0) = H$  and  $F(\pi/2) = K$ , the matrix  $A$  or the flow  $A(t)$  have the same unitary block structure as the hermitean matrices  $F(t_1)$  and  $F(t_2)$  do, and our algorithm is complete.

This algorithm depends on the magnitude relations between the non-zero entries and the near zero entries in  $F(t_i)$ . To distinguish between computed entries as being 'zero' or definitely 'nonzero' is an uncharted question. We have set the 'zero' threshold heuristically to  $\|A\|_{fro} \cdot 10^{-13} \approx \|A\|_{fro} \cdot \text{eps}$  when working in double precision in Matlab.

---

To prove the main **Unitary Diagonability Theorem**, we assume that the hermitean Johnson flow  $F(t)$  contains two linearly independent matrices  $F(t_a) \neq F(t_b)$  where  $F(t) = \cos(t)H + \sin(t)K$  for the hermitean and skew parts  $H = (A + A^*)/2 = H^*$  and  $K = (A - A^*)/(2i) = K^*$  of  $A$ .

We know of no way to prove the unitary block-diagonalization theorem algebraically; an algebraic proof of our result was never attempted. An algorithmic proof was nearly impossible before the advent of computers and the understanding of matrix flows such as the Johnson flow  $\mathcal{F}(t)$  and of the predictive qualities and accuracy of Zhang Neural Networks.

The ability to construct the field of values  $F(A)$  boundary curve of some unitarily decomposable matrices quickly and accurately via shooting methods led to recent further studies of matrix eigencurve behavior by many who tried to locate their crossings or hyperbolic avoidances that had first appeared in original Quantum Physics studies in Copenhagen, Berlin and Göttingen in the 1920s. Albert Einstein shunned this fundamental Quantum Theory problem and rather worked on relativity. Bohr's group wanted to understand eigencurve crossings and their relation to the unitary matrix block-decomposability of matrices, but neither succeeded.

Eigencurve crossing studies for the unsolved unitary block-diagonalization problem began anew with Charlie Johnson's work on the field of values in the 1970s and advanced in the 1990s in Luca Dieci et al's, Loisel and Maxwell's, and my eigencurve papers. By 2020 several small sub-results of the unitary block-decomposition problem had been solved, but no complete classification had been found and the century old problem was still open in 2023.

Our final classification of unitary block-decompositions for both, general matrix flows and static matrices, was inspired by Yunong Zhang's parameter-varying Neural Networks that date back to his doctoral thesis of 2001.

These recent developments are inter-woven and all have contributed to each other's solution and to the author's understanding of this cluster's new fun-

damental computational results. Each of the new results has advanced its area tremendously where no math methods had been found for decades, by researchers that relied on standard mathematical or computational approaches.

The mathematical process and its results were never predictable – until everything fell into place in 2023/24, was mathematically sound, proved, and their papers quickly accepted.

The **conditionally stable** Matrix Computational algorithms of this research cluster deliver **highly accurate results at high speed** and they do not suffer from the inaccuracies of our the traditional 'backward stable' methods

*This is the end of my math talk with selections from the new extraordinary mathematical research cluster.*

***What is our most urgent task today, in Linear Algebra and Matrix Theory, teaching and research wise ?***

***What needs to be done immediately ?***

We must modernize our early College teachings in Linear Algebra and enliven this area of Mathematics that helps us all so extraordinarily in our daily cell-phone and internet based lives and in our AI and engineering advances.

How do we, how do you mostly teach beginners' linear algebra classes today?

Which subjects do we, you, and I teach?

Using modern Matrix Theory based ideas **or** from a classical and algebraic standpoint ?

Top-down taught or taught interactively ?

**That is the question**

**Math Education : the Necessity of Modernizing the First Linear Algebra Course via coherent Matrix Theory Based Lesson Plans**

This is the subject of a *serious Math Education session*, some time late at night, in Ithaka.                            All are welcome

# Estimating the Numerical Range with a Krylov Subspace

*John Urschel, Cecilia Chen*

## Abstract

Moment-based methods are ubiquitous in applied mathematics, and numerical linear algebra is no exception. Krylov subspace methods are an incredibly popular family of algorithms that approximate the solution to some problem involving a matrix  $A$  using the Krylov subspace

$$\mathcal{K}_m(A, b) = \text{span}\{b, Ab, A^2b, \dots, A^{m-1}b\}.$$

This includes methods for approximating linear systems (conjugate gradient method, GMRES, MINRES, etc.), extremal eigenvalue problems (Arnoldi iteration, Lanczos method, etc.), and matrix functions. In many applications,  $m$  is much smaller than  $n$ , and a key benefit of this framework is that only matrix-vector products are involved, allowing for fast computation.

In this talk, we focus on the quality of estimate on the numerical range

$$W(A) = \left\{ \frac{x^*Ax}{x^*x} \mid x \in \mathbb{C}^{n \times n} \right\}$$

of a matrix  $A$  provided by the numerical range of the orthogonal projection of  $A$  onto the Krylov subspace  $\mathcal{K}_m(A, b)$  for some vector  $b$ , denoted by  $H_m$ . Estimates on  $W(A)$  are important not only in the computation of extremal eigenvalues, but for error estimates for other methods. For instance, standard error bounds for the residual in the GMRES algorithm for  $Ax = b$  after  $m$  steps depend on the quantity

$$\min_{p \in \mathcal{P}_m \text{ s.t. } p(0)=1} \max_{\lambda \in \Lambda(A)} |p(\lambda)|, \quad (1)$$

where  $\mathcal{P}_m$  is the set of complex polynomials of degree at most  $m$  and  $\Lambda(A)$  is the spectrum of  $A$ . If  $W(H_m)$  provides a good estimate of  $W(A)$ , or even  $\text{conv}\{\Lambda(A)\}$ , then computing  $W(H_m)$  and estimating the separation from zero can produce guarantees for the convergence rate.

Typically, estimates for approximating an extreme eigenvalue  $\lambda$  with a Krylov subspace  $\mathcal{K}_{m+1}(A, b)$  depend on the quality of the initial guess  $b$  in relation to the eigenspace of  $\lambda$ , the eigenvector condition number, and the quantity

$$\min_{p \in \mathcal{P}_m \text{ s.t. } p(\lambda)=1} \max_{\mu \in \Lambda(A) \setminus \lambda} |p(\mu)|. \quad (2)$$

The latter quantity is small when there is separation between  $\lambda$  and  $\Lambda(A) \setminus \lambda$  in the complex plane, but can become arbitrarily close to one in many cases, producing unnecessarily pessimistic bounds. These issues can persist even when  $A$  is Hermitian. For instance, consider the tridiagonal matrix  $A$  resulting from the discretization of the Laplacian operator on an interval with Dirichlet boundary conditions (i.e.,  $A_{ii} = 2$ ,  $A_{i,i+1} = A_{i+1,i} = -1$ ). This matrix has no small eigenvalue gaps, and so standard gap-dependent error estimates significantly over estimate the error in approximation for extreme eigenvalues.

In practice, Krylov subspace methods perform well at estimating extreme eigenvalues, even when eigenvalue gaps are small. In their 2015 paper, Musco and Musco argue that bounds on distances to any particular eigenvector are not necessarily needed in situations where the goal is only to estimate the eigenvalues. While a good estimate of an eigenvector naturally implies a good estimate of its

corresponding eigenvalue, the converse is simply not true. For instance, while having two extreme eigenvalues  $\lambda$  and  $\mu$  very close together significantly hinders eigenvector approximation, it can actually improve eigenvalue estimation, as a good approximation to any vector in the direct sum of the eigenspaces of  $\lambda$  and  $\mu$  (assuming the matrix is well-conditioned) implies a good estimate to both eigenvalues, since they are both close together.

Kuczynski and Wozniakowski were arguably the first to fully recognize this phenomenon in a quantitative way, producing a probabilistic upper bound of the form

$$\frac{\lambda_{\max}(A) - \lambda_{\max}^{(m)}}{\lambda_{\max}(A) - \lambda_{\min}(A)} \lesssim \frac{\ln^2 n}{m^2} \quad (3)$$

for a real symmetric matrix  $A \in \mathbb{R}^{n \times n}$ , where  $\lambda_{\max}^{(m)}$  is the largest eigenvalue of  $H_m$ , the orthogonal projection of  $A$  onto  $\mathcal{K}_m(A, b)$ , for an initial guess  $b$  sampled uniformly from the hypersphere.

In this talk, we consider the extension of this type of gap-independent result to general matrices. In particular, we consider the approximation that the numerical range of  $H_m$  provides to the numerical range of  $A$  and to the convex hull of the eigenvalues of  $A$ . This is quantified by the Hausdorff distance  $d_H(W(H_m), W(A))$  between  $W(H_m)$  and  $W(A)$ , and the one-sided Hausdorff distance  $\sup_{\mu \in \text{conv}(\Lambda(A))} d(\mu, W(H_m))$  between  $\text{conv}(\Lambda(A))$  and  $W(H_m)$ . We will consider three distinct cases: normal matrices, normal matrices with their spectrum on a circle (e.g., unitary matrices), and non-normal matrices. Time permitting, we may also discuss bounds for the behavior of the quantities (1) and (2), depending on the structure of the eigenvalues of  $A$ .

# Accelerating Operator Sinkhorn Iteration with Overrelaxation

André Uschmajew, Tasuku Soma

## Abstract

In *operator scaling*, we are given matrices  $A_1, \dots, A_k \in \mathbb{R}^{m \times n}$  and the goal is to find a joint transformation  $\bar{A}_i = LA_iR^\top$  by square invertible matrices  $L$  and  $R$  such that

$$\sum_{i=1}^k \bar{A}_i \bar{A}_i^\top = L \left( \sum_{i=1}^k A_i R^\top R A_i^\top \right) L^\top = \frac{1}{m} I_m$$

and

$$\sum_{i=1}^k \bar{A}_i^\top \bar{A}_i = R \left( \sum_{i=1}^k A_i^\top L^\top L A_i \right) R^\top = \frac{1}{n} I_n,$$

where  $I_m$  and  $I_n$  denote identity matrices. In several respects, this problem naturally generalizes the famous matrix scaling problem, where one is given a nonnegative matrix and looks for a scaling of the columns and rows such that the matrix becomes doubly stochastic. While introduced by Gurvits in a quantum complexity context [1], operator scaling actually has found various applications, including non-commutative polynomial identity testing, computational invariant theory, functional analysis, scatter estimation, and signal processing.

The *operator Sinkhorn iteration* (OSI) is an iterative algorithm for finding a solution to the scaling problem. It is based on the observation that each single condition can be easily satisfied when ignoring the other. For instance, when fixing  $R$ , it is easy to find  $L$  in the first equation using a Cholesky decomposition of the term in brackets, and similar vice versa. Updating  $L$  and  $R$  in an alternating way yields OSI. It admits a natural interpretation as an alternating minimization method on cones of symmetric positive definite matrices. Indeed, substituting  $X = L^\top L$  and  $Y = R^\top R$ , the operator scaling problem can be rewritten as a coupled fixed point equation

$$X = \frac{1}{m} \Phi(Y)^{-1}, \quad Y = \frac{1}{n} \Phi^*(X)^{-1}$$

where  $\Phi(Y) = \sum_{i=1}^k A_i Y A_i^\top$   $\Phi$  is a completely positive map, and  $\Phi^*$  is its adjoint. As it turns out, these equations are the first-order optimality conditions for the function

$$f(X, Y) = \text{tr}(X\Phi(Y)) - \frac{1}{m} \log \det(X) - \frac{1}{n} \log \det(Y),$$

which is known to be geodesically convex with respect to the so called affine invariant metric on symmetric positive definite matrices. As a result, OSI can be interpreted as an alternating minimization method for finding the global minimum of  $f$ . Its global convergence (under some additional conditions) is well known and can be established using nonlinear Perron–Frobenius theory, which implies that the underlying fixed point iteration is a contraction in the Hilbert metric on positive definite matrices. However, in certain applications the convergence rate is observed to be slow. The goal of this work is to accelerate OSI using overrelaxation.

Conceptually, an OSI including relaxation could be an iteration of the form

$$X_{t+1} = \omega \frac{1}{m} \Phi(Y_t)^{-1} + (1 - \omega) X_t, \quad Y_{t+1} = \omega \frac{1}{n} \Phi^*(X_{t+1})^{-1} + (1 - \omega) Y_t,$$

that is, combining old and new iterates with a relaxation parameter  $\omega > 0$ . While this particular version can work in certain instances, it is not well defined in general when  $\omega > 1$  (which is the case of interest), since it is not guaranteed that positive definite matrices are produced. Among other variants, we therefore propose a more natural way of combining iterates along geodesics with respect to the Hilbert metric. For two such matrices  $X$  and  $\tilde{X}$ , the geodesic connecting them is known to be  $X \#_{\omega} \tilde{X} = X^{1/2} (X^{-1/2} \tilde{X} X^{-1/2})^{\omega} X^{1/2}$  with  $\omega \geq 0$ . Importantly, here  $\omega$  can be larger than one. By replacing the above affine linear combinations with this operation, we obtain a geodesic version of overrelaxation. This version in fact is a natural generalization of the overrelaxation methods proposed in [4] and [2] for the matrix Sinkhorn iteration based on log-coordinates.

The mathematical contributions of this work include a rigorous local convergence analysis for the proposed overrelaxation methods based on linearization, which allows to determine the asymptotically optimal relaxation parameter  $\omega$  (which is larger than one) using a variant of Young's linear SOR theorem. This analysis follows an established pattern as in [4, 2], but executing it for OSI requires several nontrivial steps that provide additional insight into the problem and the geometry of positive definite matrices. For the version based on geodesic relaxation we can establish its global convergence at least in a limited range of  $\omega$  which contains values larger than one. As for practical results, we demonstrate with an instance of frame scaling that OSI with overrelaxation can significantly accelerate the standard variant in certain applications.

The talk is based on the preprint [3].

## References

- [1] L. Gurvits. Classical complexity and quantum entanglement. *J. Comput. System Sci.*, 69(3):448–484, 2004.
- [2] T. Lehmann, M.-K. von Renesse, A. Sambale, and A. Uschmajew. A note on overrelaxation in the Sinkhorn algorithm. *Optim. Lett.*, 16(8):2209–2220, 2022.
- [3] T. Soma, A. Uschmajew. Accelerating operator Sinkhorn iteration with overrelaxation. *arXiv:2410.14104*, 2024.
- [4] A. Thibault, L. Chizat, C. Dossal, and N. Papadakis. Overrelaxed Sinkhorn-Knopp algorithm for regularized optimal transport. *Algorithms (Basel)*, 14(5):Paper No. 143, 16, 2021.

# Quantum Krylov Methods for Eigenvalue Calculations

*Roel Van Beeumen, Daan Camps, Katherine Klymko, Yizhi Shen, Niel Van Buggenhout*

## Abstract

We consider the problem of computing a few of the smallest eigenvalues of the Hermitian eigenvalue problem

$$Hx = \lambda x, \quad (1)$$

where  $H$  is a Hermitian matrix of exponential dimension  $N = 2^n$ . This problem is of critical importance in fields such as condensed matter physics, quantum chemistry, and materials science, where solving such eigenvalue problems yields ground and excited state energies of quantum many-body Hamiltonians.

Classical numerical methods based on Krylov subspaces rank among the most successful techniques in numerical linear algebra. In this talk, we introduce various *quantum Krylov methods* for solving the Hermitian eigenvalue problem (1). These quantum subspace techniques present promising tools within the growing field of hybrid quantum-classical algorithms. Hybrid strategies deploy a quantum computer for the tasks where it excels, for example evolving a wavefunction with unitary operators, while offloading other parts of the computation to a classical computer. The hybrid quantum-classical paradigm allows to bridge the gap between the current noisy intermediate-scale quantum (NISQ) devices with all their limitations and the era of large-scale fault-tolerant quantum computers. We focus on quantum eigenvalue algorithms that are particularly well-suited for near-term applications on NISQ hardware.

The first and oldest class of hybrid quantum-classical methods for solving (1) are *variational quantum algorithms* [1]. In these algorithms, we employ a parameterized *ansatz* eigenvector  $x(\theta)$  and optimize the Rayleigh quotient

$$R(H, x(\theta)) = x(\theta)^* H x(\theta), \quad (2)$$

which can be measured on a quantum computer as the expectation value of the observable corresponding to  $H$  when the system is in state  $x(\theta)$ . A classical optimizer iteratively updates the parameters  $\theta$  based on the expectation values measured on the quantum computer. However, variational quantum algorithms face challenges, notably the phenomenon of barren plateaus, where the gradient of the optimization landscape vanishes exponentially, making it difficult for classical optimizers to converge [7].

Recently, *quantum subspace methods* have emerged as a robust alternative class of hybrid eigenvalue algorithms [2, 3, 5]. These methods operate within subspaces, rather than iteratively optimizing a single vector, and often achieve faster convergence than variational quantum algorithms. We focus on quantum Krylov subspaces constructed via real-time evolution

$$v_0 \rightarrow e^{-iHt} v_0, \quad (3)$$

a unitary operation native to quantum hardware and well-suited for NISQ implementations, via, for example, Trotterization using Lie product formulas [4]. The corresponding Krylov subspace is constructed as follows

$$\mathcal{K}_m(U, v_0) = [v_0, Uv_0, U^2v_0, \dots, U^{m-1}v_0], \quad (4)$$

where  $U^k = e^{-iHk\Delta t}$  is the unitary obtained from evolving  $H$  over time  $t = k\Delta t$ .

Unlike classical Krylov subspace methods, we do not construct the subspace explicitly on the quantum computer. Instead, we obtain the projected eigenvalue problem through quantum measurements and solve this small projected problem classically. To compute the matrix elements of the projected Hamiltonian  $\hat{H}$ , we start by preparing the initial quantum state  $v_0$  and evolving it over time  $k\Delta t$ , resulting in  $v_k = e^{-iHk\Delta t}v_0$ . We then measure the matrix elements

$$\hat{H}_{j,k} = v_j^* H v_k. \quad (5)$$

Because orthogonalizing vectors on a quantum computer is challenging and definitely impossible in the NISQ era, quantum subspace methods utilize non-orthogonal bases. Consequently, we also need to measure the inner products

$$\hat{S}_{j,k} = v_j^* v_k, \quad (6)$$

leading to the projected *generalized* eigenvalue problem

$$\hat{H}x = \lambda \hat{S}x. \quad (7)$$

However, solving for the Ritz values of this generalized eigenvalue problem (7) is often challenging due to the ill-conditioning of the problem. To address this issue, we present a dynamic mode decomposition approach, a method designed to circumvent this ill-conditioning and capable of robustly obtaining eigenvalue estimations from noise quantum observables [5]. In this talk, we present a numerical linear algebra perspective on quantum subspace algorithms, discuss strategies to avoid ill-conditioned eigenvalue problems, and provide both theoretical and numerical evidence of convergence. Additionally, we introduce strategies to enhance robustness, particularly in noisy quantum environments.

A third class of hybrid quantum-classical eigenvalue methods leverages rational functions, which offer rapid convergence in scientific computing but remain underexplored in quantum algorithms. We present efficient implementations of rational transformations on quantum hardware [6]. By using integral representations of the resolvent, we can efficiently perform quantum rational transformations through Hamiltonian simulations and the linear-combination-of-unitaries (LCU) method. We introduce two complementary LCU-based strategies—discrete-time and continuous-time—offering flexible approaches for quantum rational transformations. We also illustrate these novel methods through numerical simulations on spin systems, achieving precise calculations of low-lying eigenvalues.

## References

- [1] M. Cerezo, A. Arrasmith, R. Babbush, S.C. Benjamin, S. Endo, K. Fujii, J.R. McClean, et al., *Variational quantum algorithms*, *Nature Reviews Physics*, 3 (2021), pp. 625–644.
- [2] E.N. Epperly, L. Lin, and Y. Nakatsukasa, *A theory of quantum subspace diagonalization*, *SIAM Journal on Matrix Analysis and Applications*, 43 (2022), pp. 1263–1290.
- [3] W. Kirby, M. Motta, and A. Mezzacapo, *Exact and efficient Lanczos method on a quantum computer*, *Quantum*, 7 (2023), p. 1018.
- [4] K. Klymko, C. Mejuto-Zaera, S.J. Cotton, F. Wudarski, M. Urbanek, D. Hait, M. Head-Gordon, et al., *Real-time evolution for ultracompact Hamiltonian eigenstates on quantum hardware*, *PRX Quantum*, 3 (2022), p. 20323.
- [5] Y. Shen, D. Camps, A. Szasz, S. Darbha, K. Klymko, D.B. Williams-Young, N.M. Tubman, and R. Van Beeumen, *Estimating eigenenergies from quantum dynamics: A unified noise-resilient measurement-driven approach*, 2023, [arXiv.2306.01858](https://arxiv.org/abs/2306.01858).
- [6] Y. Shen, N. Van Buggenhout, D. Camps, K. Klymko, and R. Van Beeumen, *Quantum rational transformation using linear combinations of Hamiltonian simulations*, 2024, [arXiv.2408.07742](https://arxiv.org/abs/2408.07742).
- [7] S. Wang, E. Fontana, M. Cerezo, K. Sharma, A. Sone, L. Cincio, and P.J. Coles, *Noise-induced barren plateaus in variational quantum algorithms*, *Nature Communications*, 12 (2021), p. 6961.

# Numerically generating Sobolev orthogonal polynomials

Niel Van Buggenhout, Francisco Marcellán, Nicola Mastronardi

## Abstract

Based on structured matrix techniques, we propose new numerical algorithms to generate a sequence  $\{S_0(x), S_1(x), \dots, S_{K-1}(x)\}$  of Sobolev orthogonal polynomials (SOPs). In this sequence, the polynomial  $S_k(x)$  of degree  $k$  satisfies orthogonality conditions with respect to a Sobolev inner product  $(\cdot, \cdot)$ , namely

$$(S_k, S_\ell) \begin{cases} = 0 & \text{if } k \neq \ell, \\ \neq 0 & \text{if } k = \ell. \end{cases}$$

A Sobolev inner product is a linear functional where the functions themselves and their derivatives, up to order  $M$ , appear with (possibly different) weight functions  $w_m(x)$ , i.e.,

$$(p, q) = \sum_{m=0}^M \int_{\Omega} p^{(m)}(x) q^{(m)}(x) w_m(x) dx.$$

Sobolev orthogonal polynomials were first studied in approximation theory, when solving interpolation problems where values of the function and its derivatives are known [1]. The study of their analytical properties, e.g., the behavior of zeros, is an active area of research [4]. Moreover, since the weak form of differential equations gives rise to a Sobolev inner product, SOPs can be used to develop spectral methods [2]. The generation of sequences of SOPs is central to these applications.

### Main problem:

Given a Sobolev inner product  $(\cdot, \cdot)$ , generate the sequence  $\{S_0(x), S_1(x), \dots, S_{K-1}(x)\}$  such that

- $S_k(x)$  is a polynomial of exact degree  $k$ ,
- $(S_k, S_\ell) = 0$ , for  $k \neq \ell$ , and  $(S_k, S_\ell) \neq 0$ , for  $k = \ell$ .

Our proposed algorithms can be used for general Sobolev inner products, in this presentation we focus on a particular, interesting family of SOPs. Gegenbauer-Sobolev polynomials are orthogonal with respect to the continuous Sobolev inner product

$$(p, q)_\mu = \int_{-1}^1 p(x) q(x) (1 - x^2)^\mu dx + \lambda \int_{-1}^1 p'(x) q'(x) (1 - x^2)^\mu dx.$$

A finite sequence of SOPs  $\{S_0(x), S_1(x), \dots, S_{K-1}(x)\}$  is also orthogonal to a discrete Sobolev inner product. For Gegenbauer-Sobolev polynomials this discrete inner product can be obtained by applying the Gauss-Jacobi quadrature rule with weights  $\{\alpha_n\}_{n=1}^K$  and nodes  $\{x_n\}_{n=1}^K$  to  $(\cdot, \cdot)_\mu$ :

$$(S_k, S_\ell)_\mu \approx \langle S_k, S_\ell \rangle_\mu = \sum_{n=1}^K \alpha_n S_k(x_n) S_\ell(x_n) + \lambda \sum_{n=1}^K \alpha_n S'_k(x_n) S'_\ell(x_n).$$

For  $\langle \cdot, \cdot \rangle_\mu$ , a Hessenberg matrix  $H_K$  of size  $K \times K$  represents the recurrence relation of the SOPs,

$$x \begin{bmatrix} S_0(x) & S_1(x) & S_2(x) & \dots & S_{K-1}(x) \end{bmatrix} = \begin{bmatrix} S_0(x) & S_1(x) & S_2(x) & \dots & S_{K-1}(x) \end{bmatrix} H_K.$$

We show that the matrix  $H_K$  can be computed by solving the following inverse eigenvalue problem [5], where  $H_K$  appears as the  $K \times K$  leading principal submatrix of the solution  $\tilde{H}$ .

**Inverse eigenvalue problem:**

Given a discrete Sobolev inner product  $\langle \cdot, \cdot \rangle$ , construct the  $2K \times 2K$  matrices  $\tilde{H}, \tilde{Q}$  such that

- $\tilde{H}$  is a Hessenberg matrix and  $\tilde{Q}$  is unitary,  $\tilde{Q}^* \tilde{Q} = I$ .
- The first entries of the unitary matrix  $\tilde{Q}$  are determined by the weights  $\{\alpha_n\}_{n=1}^K$  in the discrete inner product, i.e.,  $\tilde{Q}e_1 = w/\|w\|_2$ , where the vector  $w$  contains the weights  $\alpha_n$ .
- The matrix  $\tilde{H}$  satisfies the decomposition  $\tilde{H} = \tilde{Q}^* J \tilde{Q}$ , for a Jordan-like matrix  $J$  determined by the nodes  $\{x_n\}_{n=1}^K$  and their multiplicity, as they appear in the discrete inner product.

For the Gegenbauer-Sobolev polynomials, the vector of weights  $w$  and Jordan-like matrix  $J$  are

$$w = \begin{bmatrix} 0 \\ \sqrt{\alpha_1} \\ \vdots \\ 0 \\ \sqrt{\alpha_K} \end{bmatrix} \quad \text{and} \quad J = \begin{bmatrix} x_1 & \sqrt{\lambda} & & \\ & x_1 & & \\ & & \ddots & \\ & & & x_K & \sqrt{\lambda} \\ & & & & x_K \end{bmatrix}.$$

Thanks to this reformulation as a Hessenberg inverse eigenvalue problem, we can use structured matrix techniques to compute  $H_K$ . Our proposed numerical algorithm is based on plane rotations and constructs  $H_K$  by employing unitary similarity transformations to the Jordan-like matrix  $J$ . It does not require the storage of the whole matrix  $Q_k$ , only its first column and is, therefore, more efficient than the state-of-the-art algorithms [3]. Under mild assumptions on the Sobolev inner product, our numerical algorithm can be applied to any sequence of SOPs.

For certain families of SOPs, specialized algorithms can be developed that exploit their properties. For Gegenbauer-Sobolev polynomials we exploit the fact that the Gegenbauer measure forms a symmetrical coherent pair with itself and use properties of (classical) Gegenbauer polynomials to obtain a factorization of  $H_K$  as the product of three structured matrices. The entries of all three matrices are given analytically by closed form expressions, which could reduce the accumulation of rounding error in  $H_K$ . We discuss numerical experiments in which we compare the general purpose, specialized, and state-of-the-art algorithms [3] for Gegenbauer-Sobolev polynomials.

## References

- [1] P. ALTHAMMER, *Eine Erweiterung des Orthogonalitätsbegriffes bei Polynomen und deren Anwendung auf die beste Approximation*, J. Reine Angew. Math., 211 (1962), pp. 192–204.
- [2] L. FERNÁNDEZ, F. MARCELLÁN, T. E. PÉREZ, AND M. A. PIÑAR, *Sobolev orthogonal polynomials and spectral methods in boundary value problems*, Appl. Numer. Math., 200 (2024), pp. 254–272.
- [3] W. GAUTSCHI AND M. ZHANG, *Computing orthogonal polynomials in Sobolev spaces*, Numer. Math., 71 (1995), pp. 159–183.
- [4] F. MARCELLÁN AND Y. XU, *On Sobolev orthogonal polynomials*, Expo. Math., 33 (2015), pp. 308–352.
- [5] N. VAN BUGGENHOUT, *On generating Sobolev orthogonal polynomials*, Numer. Math., 155 (2023), pp. 415–443.

# Spectral problems through the lens of optimization: new ideas and improved algorithms?

Bart Vandereycken

## Abstract

Thanks to influential works like [8, 1], many classical problems in numerical linear algebra (NLA) can be formulated as optimization problems on smooth and differentiable manifolds. The link with optimization on manifolds allows us to approach these problems from the world of numerical optimization. The archetypical example is the symmetric eigenvalue problem (EVP): the dominant  $k$ -dimensional eigenspaces of  $A$  correspond to extrema of the partial trace function

$$f(X) = -\text{Trace}(X^T AX), \quad (1)$$

where  $X \in \mathbb{R}^{n \times k}$  is an orthonormal matrix (that is,  $X^T X = I_k$ ). Due to the partial trace being invariant by orthogonal transformation on the right (that is,  $X \rightsquigarrow XQ$  with orthogonal  $Q$ ), this problem is naturally stated on  $\text{Gr}(n, k)$ , the Grassmann manifold of  $k$ -dimensional subspaces in  $\mathbb{R}^n$ . Minimizing  $f$  by the Riemannian steepest descent method is, in specific cases, equivalent to the power method.

It is well known that the steepest descent method converges exponentially fast, in distance to the optimizer and in function value, when the objective function is locally strongly convex. Applied to spectral problems in NLA, a nonzero spectral gap is required to ensure uniqueness and the initial estimate has to be sufficiently close to the optimal subspace. Unfortunately, the latter condition is usually very stringent. For a symmetric matrix  $A$  with eigenvalues  $\lambda_1 \geq \dots \geq \lambda_n$ , for example, we have shown in [5] that (1) is geodesically convex in

$$N = \left\{ \text{span}(X) \in \text{Gr}(n, k) : \sin^2(\theta_k) \leq \frac{\lambda_k - \lambda_{k-1}}{\lambda_1 + \lambda_k} \right\}.$$

Here,  $\theta_k$  is the  $k$ th principal angle between  $\text{span}(X)$  and the dominant eigenspace  $\text{span}(V)$ . While this is an improvement over more direct estimates that require  $\theta_k = O(\delta)$ , the condition  $\theta_k = O(\sqrt{\delta})$  is still small.

Fortunately, classical (geodesic) convexity is not needed to have gradient descent converge exponentially fast. In the Euclidean case, an old result by [11] proves that the Polyak–Łojasiewicz (PL) condition,

$$\exists \mu > 0 \quad \text{s.t.} \quad \|\nabla f(x)\|^2 \geq 2\mu(f(x) - f^*), \quad \forall x \in \mathbb{R}^n, \quad (2)$$

is sufficient to guarantee fast (exponential) convergence in function value. The PL condition with constant  $\mu$  is weaker than  $\mu$  strong convexity

More recently, an even weaker notion of strong convexity that relates to convergence with respect to distance to the optimum, has been studied [7, 10, 5]. The property is called weak-quasi-strong-convexity (WQSC) and is defined in the Euclidean case as follows:

$$\exists a > 0, \mu > 0 \quad \text{s.t.} \quad f(x) - f^* \leq \frac{1}{a} \langle \nabla f(x), x - x_p \rangle - \frac{\mu}{2} \|x - x_p\|^2, \quad \forall x \in \mathbb{R}^n,$$

with  $x_p$  the projection of  $x$  onto the solution set of minimizers of  $f$ .

We have shown in [5, 2] that the manifold version of the WQSC property applies to the following spectral problems:

- Symmetric EVP of  $A$ : the objective function  $f$  in (1) is WQSC with parameters  $a(\text{span}(X)) = \theta_k / \tan \theta_k$  and  $\mu = 8\delta/\pi^2$ .
- Symmetric generalized EVP of  $(A, B)$  with  $B \succ 0$ : the objective function

$$f(\text{span}(X)) = -\text{Trace}((X^T BX)^{-1} X^T AX)$$

is WQSC with parameters  $a(\text{span}(X)) = \sigma_{\min}(V^T BX(X^T BX)^{-1/2})$  and  $\mu = 8\delta/\pi^2$ .

Once WQSC is shown to hold, it can be used to analyse accelerated versions of gradient descent [7, 6]. For the symmetric EVP, the Riemannian conjugate gradient method from [4] also leads to practical improvements when comparing to other accelerated gradient methods, like the LOBPCG method of [9].

Would it be possible to relax these generalized convexity properties even more? In other words, suppose gradient descent converges exponentially fast when started in any point in a set around the optimum, then which property does  $f$  satisfy? As shown in [3], the objective needs to be WQSC when measuring convergence in distance to the optimum. Recently, we have also shown that only the PL condition is required for convergence in function value. Hence, PL and WQSC are in some sense necessary and sufficient for a fast gradient method.

An added bonus of the optimization viewpoint is that gapless problems can be treated and analysed fairly easily. The convergence of gradient descent is no longer exponential but only algebraic.

This talk will present a general overview of these properties and highlight algorithmic and analytical applications from NLA. The contents are based on joint work with Pierre-Antoine Absil, Foivos Alimisis, and Yousef Saad.

## References

- [1] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2008.
- [2] Pierre-Antoine Absil, Foivos Alimisis, and Bart Vandereycken. Riemannian inexact gradient descent for high-dimensional canonical correlation analysis. *In preparation*, 2024.
- [3] Foivos Alimisis. Characterization of optimization problems that are solvable iteratively with linear convergence. *MTNS*, 2024a.
- [4] Foivos Alimisis, Yousef Saad, and Bart Vandereycken. Gradient-type subspace iteration methods for the symmetric eigenvalue problem. *arXiv preprint arXiv:2306.10379*, 2023.
- [5] Foivos Alimisis and Bart Vandereycken. Geodesic convexity of the symmetric eigenvalue problem and convergence of steepest descent. *Journal of Optimization Theory and Applications*, pages 1–40, 2024.
- [6] Foivos Alimisis, Simon Vary, and Bart Vandereycken. A nesterov-style accelerated gradient descent algorithm for the symmetric eigenvalue problem. *arXiv preprint arXiv:2406.18433*, 2024.
- [7] Jingjing Bu and Mehran Mesbahi. A note on Nesterov’s accelerated method in nonconvex optimization: a weak estimate sequence approach. *arXiv preprint arXiv:2006.08548*, 2020.

- [8] Alan Edelman, Tomás A Arias, and Steven T Smith. The geometry of algorithms with orthogonality constraints. *SIAM journal on Matrix Analysis and Applications*, 20(2):303–353, 1998.
- [9] Andrew V Knyazev. Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method. *SIAM journal on scientific computing*, 23(2):517–541, 2001.
- [10] Ion Necoara, Yurii Nesterov, and Francois Glineur. Linear convergence of first order methods for non-strongly convex optimization. *Mathematical Programming*, 175:69–107, 2019.
- [11] Boris Teodorovich Polyak. Gradient methods for minimizing functionals. *Zhurnal vychislitel'noi matematiki i matematicheskoi fiziki*, 3(4):643–653, 1963.

# Subspace methods with asymptotic Krylov convergence for bounded variable problems.

Wim Vanroose, Bas Symoens

Abstract

**Introduction.** Krylov subspace methods are highly effective in solving problems with sparse matrices at scale. Many large-scale problems in mechanics and fluid dynamics can be addressed using preconditioned Krylov solvers, and numerous specialized algorithms based on subspace methods have been developed that scale to the largest supercomputers. However, many scientific and engineering problems impose bounds on variables. Examples include nonnegative matrix factorization, contact problems in mechanics, and planning problems with resource and capacity constraints. Data science problems often have  $\|\cdot\|_\infty$ - or  $\|\cdot\|_1$ -norms. In these cases, active bounds manifest as boundary conditions, but it is often unknown in advance which bounds will be active.

Each bound on a variable is expressed as an inequality. In the optimality conditions, each inequality leads to a nonlinear complementarity condition that couples the variable and the corresponding Lagrange multiplier. The traditional approach is to linearize the system of optimality conditions and solve the resulting saddle point system. Each linearization is solved within a subspace with a iterative method for the saddle point problems [1]. However, this subspace is not reused in the next linearization; a new subspace is built in the subsequent outer iteration. This is inefficient.

We propose using a single subspace that is kept over all the iterations. Let us summarize how subspace methods reduce the problem and how we can generalize this.

A Krylov subspace for a square matrix  $A \in \mathbb{R}^{n \times n}$  and a vector  $v \in \mathbb{R}^n$  is defined as  $\mathcal{K}_k(A, v) := \text{span}\{v, Av, A^2v, \dots, A^{k-1}v\}$ . Equivalently, this subspace can be written  $\text{span}\{r_0, r_1, r_2, \dots, r_{k-1}\}$ , where  $r_i$  are the residuals, which are mutually orthogonal.

The *conjugate gradients* methods (CG) minimizes the error in the  $A$ -norm over the Krylov subspace for a symmetric and positive definite matrix  $A$ . Specifically, it solves  $\min_{x \in x_0 + \mathcal{K}_k(A, r_0)} \|x - x^*\|_A^2$ . Expanding the solution as  $x_k = x_0 + V_k y_k$  and writing down the optimality conditions leads to **a small linear system:**  $(V_k^T A V_k) y_k = \|r_0\|_2 e_1$ . Since  $A$  is symmetric, the matrix  $V_k^T A V_k$  is a tridiagonal. If we have found the solution for a basis  $V_k$ , the solution for the next iteration can be warm-started from the previous solution.

In the *generalized minimal residual* (GMRES), we minimize the 2-norm of the residual over the Krylov subspace. It is  $\min_{x \in x_0 + \mathcal{K}(A, r_0)} \|b - Ax\|_2$ , for a general square matrix  $A$ . The optimality conditions correspond to **a small least-squares problem:**  $\min_{y \in \mathbb{R}^k} \| \|r_0\|_2 e_1 - (V_{k+1}^T A V_k) y_k \|_2$ . Here, the matrix  $V_{k+1}^T A V_k$  has a Hessenberg structure. Again, if we have found the solution for iteration  $k$ , the next solution is easily found using warm-starting.

In this talk, we generalize this approach to a bounded variable least squares problem:

$$\min \|Ax - b\|^2 \quad \text{subject to} \quad \ell \leq x \leq u, \quad (1)$$

where  $A \in \mathbb{R}^{m \times n}$ ,  $b \in \mathbb{R}^m$  and  $\ell, u \in \mathbb{R}^n$ , lower and upper bounds.

We will restrict the solution of (1) to a subspace, leading to a small projected problem. **Instead of a small linear system as in CG or a small least-squares problem as in GMRES, this will now result in a small quadratic programming (QP) problem.** This system will solve

for the optimal coefficients of the solution and for the Lagrange multipliers. With these, we will calculate a residual that will be added to the basis.

**Optimality conditions.** Let us start with the optimality conditions for problem (1). These are

$$\begin{aligned} A^T(Ax - b) - \lambda + \mu &= 0, \\ \lambda_i(x_i - \ell_i) &= 0, \quad i \in \{1, \dots, m\} \\ \mu_i(u_i - x_i) &= 0, \quad i \in \{1, \dots, m\} \\ \ell_i \leq x_i \leq u_i, \quad i &\in \{1, \dots, m\} \\ \lambda \geq 0 \quad \mu \geq 0. \end{aligned} \tag{2}$$

We now expand the solution in a orthogonal basis  $V_k$  as  $x_k = V_k y_k$ . The problem (1) then becomes

$$\min \|AV_k y_k - b\|_2^2 \quad \text{subject to} \quad l \leq V_k y \leq u. \tag{3}$$

The corresponding optimality conditions are now:

$$\begin{aligned} V_k^T (A^T(AV_k y_k - b) - \lambda_k + \mu_k) &= 0, \\ \lambda_i([V_k y_k]_i - \ell_i) &= 0, \quad i \in \{1, \dots, m\} \\ \mu_i(u_i - [V_k y_k]_i) &= 0, \quad i \in \{1, \dots, m\} \\ \ell_i \leq [V_k y_k]_i \leq u_i, \quad i &\in \{1, \dots, m\} \\ \lambda \geq 0 \quad \mu \geq 0. \end{aligned} \tag{4}$$

These are very similar to (2) but now the first equation is a projection of the residual onto the basis  $V_k$ . The number of complementarity conditions is the same as in the original problem.

**Residual Quadratic Programming Active Set Subspace (ResQPASS).** We are now in a position to define the ResQPASS iteration. It solves a series of small projected optimisation problems that can be warm-started with the previous solution.

**Definition 1.** *The residual quadratic programming active-set subspace (ResQPASS) [2] iteration for  $A \in \mathbb{R}^{m \times n}$ ,  $b \in \mathbb{R}^m$  and  $\ell, u \in \mathbb{R}^n$ , lower and upper bounds such that  $\ell \leq 0 \leq u$  with associated Lagrange multipliers  $\lambda_k, \mu_k \in \mathbb{R}^n$ , generates a series of approximations  $\{x_k\}_{k \in \mathbb{N}}$  that solve*

$$x_k = \underset{x \in \text{span}\{r_0, \dots, r_{k-1}\}}{\operatorname{argmin}} \|Ax - b\|_2^2 \quad \text{subject to} \quad \ell \leq x \leq u, \tag{5}$$

where

$$r_k := A^T(Ax_k - b) - \lambda_k + \mu_k. \tag{6}$$

The feasible initial guess is  $x_0 = 0$ , with  $\lambda_0 = \mu_0 = 0$  and  $r_0 := -A^T b$ .

The definition of the residual, (6), includes the current guess for the Lagrange multiplier  $\lambda_k$  and  $\mu_k$ . A non-zero Lagrange multiplier indicates where the solution in the subspace touches the bounds.

Note that the residual, as defined in Eq. (6), also appears in the first equation of the optimality conditions, (2) and (4).

The restriction  $\ell \leq 0 \leq u$  does not limit the applicability, we can use an initial guess  $x_0$  to shift to problem such that this is satisfied.

---

**Algorithm 1** Residual quadratic programming active-set subspace (ResQPASS)

---

**Require:**  $A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m, \ell, u \in \mathbb{R}^n, tol > 0$

- 1:  $r_0 = A^T b$
- 2:  $V_1 = r_0 / \|r_0\|$
- 3:  $y_1 = 0$
- 4:  $\mathcal{W}_1 = \emptyset$
- 5: **for**  $k = 1, 2, \dots, m$  **do**
- 6:    $y_k^*, \mathcal{W}_k^*, \lambda_k, \mu_k \leftarrow$  Solve Eq. (4) using QPAS, with initial guess  $y_k$  and initial working set  $\mathcal{W}_k$
- 7:    $r_k = A^T (AV_k y_k^* - b) - \lambda_k + \mu_k$
- 8:   **if**  $\|r_k\|_2 \leq tol$  **then**
- 9:      $x = V_k y_k$ , break;
- 10:   **end if**
- 11:    $V_{k+1} \leftarrow [V_k \ r_k / \|r_k\|]$
- 12:    $y_{k+1} \leftarrow [(y_k^*)^T \ 0]^T$
- 13:    $\mathcal{W}_{k+1} \leftarrow \mathcal{W}_k^*$
- 14: **end for**

---

This definition is translated in the algorithm described in Algorithm 1, which has a very similar structure as Krylov subspace methods.

If we solve the system of the projected optimality conditions (4) only to feasibility (i.e., not all Lagrange multipliers are positive), we obtain an orthogonal series of residuals  $r_k$ . Indeed, the first equation of (4) shows the current residual projected on the previous residuals. Thus, achieving feasibility means that the current residual will be orthogonal to all previous residuals.

Another observation is that when the bounds,  $\ell$  and  $u$ , do not restrict the problem (i.e. none of the bounds are active), the corresponding Lagrange multipliers  $\mu$  and  $\lambda$  will be zero, due to the complementarity conditions. In this case, the residual simplifies to the classical form  $A^T(AV_k y_k - b)$ , as in LSQR or CG. The method ResQPASS will then corresponds to a classical Krylov method.

However, when there are active bounds, the residuals will differ — but not significantly. When only a few bounds are active, the vectors of Lagrange multipliers are sparse, meaning that only a few of the elements  $\lambda_i$  and  $\mu_j$  are non-zero.

In figure 1, we studied model problems where the number of active constraints in the solution can be adjusted. What we observe is that, in the initial iterations, progress is slow because the bounds prevent a full step. However, once the limiting bounds are discovered, regular Krylov convergence sets in due to the orthogonality of the residuals.

This observation is explained by a convergence analysis [2], which reveals that after a certain number of iterations, the residuals can be expressed as polynomials of the normal matrix on some subspace,  $p(A^T A)V_0$ . At this point, regular Krylov convergence occurs. This connection to Krylov subspaces suggests superlinear convergence for problems with a small number of active constraints.

**Numerical implementation.** In the numerical implementation, we solve a series of QP problems that grow in size. Similar to CG and GMRES, solving the next problem becomes easier if the previous problem has already been solved. We can warm-start with the previous solution as the initial guess, along with the working set from the previous problem. Additionally, the factorisation of the saddle point system can be reused, as the active set changes one element at a time, and the

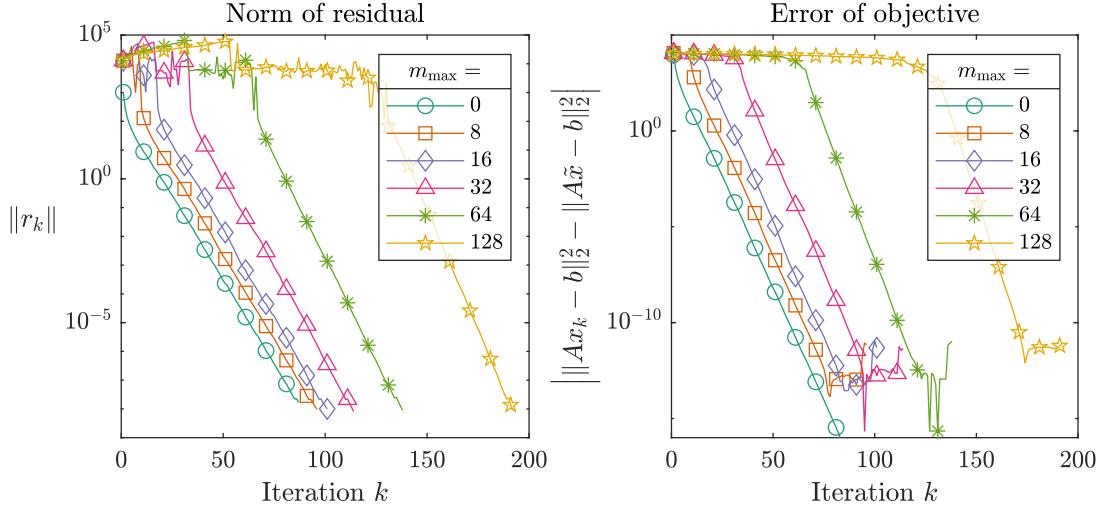


Figure 1: This figure illustrates the convergence behavior for different number of active constraints. The residual and objective behave similar to the unbounded ( $i_{\max} = 0$ , Krylov convergence) case, with a delay that is roughly equal to  $i_{\max}$ , the number of active constraints in the problem.  $\tilde{x}$  is an ‘exact’ solution found by applying MATLAB’s quadprog with a tolerance of  $10^{-15}$ .

matrices only change by a rank-1 update.

We use a Cholesky factorization of the projected Hessian  $V_k^T A^T A V_k$  or a orthogonalisation  $A V_k = U_k B_k$ , which gives asymptotically the bidiagonalisation. These factorisations ar efficiently updated as the subspace expands. Similarly, in the inner QP iterations, we use a QR factorization of the Cholesky factors applied to the active constraints, which further improves the efficiency.

By limiting the inner iterations we can choose to solve only for feasibility. In the early iterations, it is beneficial to prioritize subspace expansion over achieving full optimality within each subspace. This control over the number of inner iterations balances solution accuracy and speed. We also incorporate additional recurrence relations to avoid redundant computations, similar to techniques used in the CG method.

This results in an algorithm that performs very well in problems with a limited number of active constraints such as contact problems, offering significantly faster convergence compared to traditional methods like interior-point methods. However, the performance degrades as the number of active constraints becomes too large.

It is important to note that ResQPASS is a matrix-free method, as it primarily relies on matrix-vector products, making it suitable for problems where explicit matrix storage is impractical.

## References

- [1] Michele Benzi, Gene H Golub, and Jörg Liesen. Numerical solution of saddle point problems. *Acta numerica*, 14:1–137, 2005.
- [2] Bas Symoens and Wim Vanroose. ResQPASS: an algorithm for bounded variable linear least squares with asymptotic Krylov convergence. *arXiv preprint arXiv:2302.13616*, 2023.

# When does the randomized SVD actually compute an SVD?

Randomized subspace approximation beyond singular value gaps

*Christopher Wang, Alex Townsend*

## Abstract

The randomized SVD (rSVD) is excellent at constructing a low-rank approximation to a matrix with rapidly decaying singular values, and its theoretical behavior as such was thoroughly explained in [6]. However, the singular values and singular subspaces of a good low-rank approximation may not accurately approximate the true singular values and singular subspaces, even with oversampling. The following example illustrates the problem.

Let  $A$  be a  $1000 \times 1000$  matrix, where five of its eigenvalues are 1 and the remaining 995 eigenvalues are 0.05. We aim to estimate the 5-dimensional dominant eigenspace corresponding to the 1's using the rSVD with standard Gaussian test vectors and oversampling by 10. A simple numerical experiment returns the following results, over 1000 iterations of the rSVD:

1. Average relative low-rank error in the Frobenius norm $\frac{\ A - \tilde{A}\ _F}{\ A - A_5\ _F}$ :	1.176969
2. Average relative low-rank error in the spectral norm $\frac{\ A - \tilde{A}\ _2}{\ A - A_5\ _2}$ :	12.152447
3. Average maximum relative error for eigenvalues $\max_{j=1,\dots,5} \frac{ \sigma_j - \tilde{\sigma}_j }{ \sigma_j }$ :	0.209691
4. Average principal angle error for dominant eigenspace $\ \Theta(\mathcal{U}_5, \tilde{\mathcal{U}}_5)\ _2$ :	0.654832

Here,  $\tilde{A}$  is the rank-5 truncated approximant generated by the rSVD,  $A_5$  is the best rank-5 approximant to  $A$  (that is, the diagonal matrix with five 1's in the top left corner),  $\sigma_j, \tilde{\sigma}_j$  are the eigenvalues of  $A, \tilde{A}$  respectively, and  $\mathcal{U}_5, \tilde{\mathcal{U}}_5$  are the 5-dimensional dominant eigenspaces of  $A, \tilde{A}$  respectively. Observe that  $\tilde{A}$  is only a small factor away from being the optimal rank-5 approximant to  $A$  in the Frobenius norm, but it is much further from optimality in the spectral norm. Additionally, the rSVD has a hard time distinguishing between the eigenvalue 1 and the eigenvalue 0.1, and the rSVD's estimate for the dominant eigenspace is off by nearly 40 degrees, for the largest principal angle between the subspaces, on average. In short, the rSVD fails to be an SVD, even though it gives a good low-rank approximation in the Frobenius norm.

Therefore, we ask: for what matrices  $A$  does the rSVD actually compute an accurate SVD? We approach the question by considering how well the rSVD approximates the  $k$ th dominant singular subspaces (indeed, if the rSVD manages to accurately capture the dominant singular subspaces, then it must be accurate in the singular values as well [7]). This problem has received considerable attention over the past decade, especially through the lens of perturbation theory. The Davis-Kahan theorem, and its generalization by Wedin to nonsymmetric matrices, provides bounds, dependent on the size of the singular value gap between  $\sigma_k$  and  $\sigma_{k+1}$ , on the angular change of eigenvectors under a deterministic perturbation of a matrix. Building on these two early results, gap-dependent bounds on the accuracy of singular subspace approximations by projection-based methods, such as the rSVD, were derived for both deterministic and random settings by [5, 13, 14, 15]. Gap-independent bounds have also appeared in the works of [2, 9, 12]. While such bounds tell us when we expect to have a hard time determining the singular subspaces of a given matrix, they may fail to tell us whether a matrix is in fact conducive to singular subspace approximation. In

fact, we observe in practice matrices with small or nonexistent singular value gaps whose singular subspaces are nevertheless well-approximated by the rSVD with high probability.

Our contribution is an exact, relatively straightforward formula for the cumulative distribution function of the largest principal angle between the true and the approximate dominant singular subspace, when using the rSVD with Gaussian test vectors. Our formula encapsulates the advantages of previous works in that (a) it is computable, interpretable, and a priori in the sense that it is space-agnostic, meaning it does not depend on prior knowledge of the singular subspaces; (b) it applies for any power of subspace iteration, with any amount of oversampling (including none at all); and (c) it can be used to derive existing bounds for the largest principal angle. Since our result is exact, it is certainly gap-independent. More importantly, it helps explain why a large singular value gap improves subspace estimates, but it also explains when and why the rSVD succeeds at singular subspace approximation even when the singular value gap is small. We show that the gap-dependent bounds of [15] assume the worst-case scenario given a gap, and we show that that worst-case scenario is when the singular values of  $A$  are  $\sigma_1 = \dots = \sigma_k > \sigma_{k+1} = \dots = \sigma_n$ —the dominant singular values are as small as possible, while the tail is as large as possible.

To be precise, let  $N \geq n$  and fix the target rank  $k \geq 1$  and oversampling  $p \geq 0$  such that  $k + p < \frac{n}{2}$ . Let  $M \in \mathbb{R}^{N \times n}$  and let  $\Sigma_1, \Sigma_2$  be  $k \times k$  and  $(n - k) \times (n - k)$  diagonal matrices of the dominant and tail singular values of  $M$ , respectively. If  $\theta_1$  denotes the largest principal angle between the  $k$ th dominant left singular subspace of  $M$  and the  $(k + p)$ -dimensional column space of  $M\Omega$ , where  $\Omega$  is an  $n \times (k + p)$  standard Gaussian matrix, then the cumulative distribution function of  $\theta_1$  is given, for  $0 \leq \theta \leq \frac{\pi}{2}$ , by

$$\mathbb{P}(\theta_1 < \theta) = \mathbb{E} \left[ \det(S(\theta, \Sigma, Q, H_1, Q_1))^{\frac{n-k-p}{2}} {}_2F_1\left(\frac{-p+1}{2}, \frac{n-k-p}{2}; \frac{-p+1}{2}; I_k - S(\theta, \Sigma, Q, H_1, Q_1)\right) \right]$$

where

1.  $S(\theta, \Sigma, Q, H_1, Q_1)$  is the  $k \times k$  matrix

$$\sin^2(\theta)\Sigma_1 (\sin^2(\theta)\Sigma_1^2 + \cos^2(\theta)QQ'_1(H'_1\Sigma_2^{-2}H_1)^{-1}Q_1Q')^{-1}\Sigma_1;$$

2.  $Q$ ,  $H_1$ , and  $Q_1$  are random matrices which are respectively the  $Q$  factors in the QR decomposition of a  $k \times k$  standard Gaussian matrix, an  $(n - k) \times (k + p)$  matrix whose columns are independently sampled from the multivariate Gaussian  $\mathcal{N}(0, \Sigma_2^2)$ , and a  $(k + p) \times k$  matrix whose columns are independently sampled from  $\mathcal{N}(0, H_1^\top \Sigma_2^{-2} H_1)$ ; and
3.  ${}_2F_1(a, b; c; X)$  is the Gaussian hypergeometric function of matrix argument.

All of these quantities are computable given the singular values of  $M$ ; the expectation can be computed by Monte–Carlo simulation, while the hypergeometric function can be evaluated extremely quickly via [8]. Numerical experiments demonstrate excellent agreement between our formula and empirical observations.

The primary tools used to derive our formula for the cumulative distribution function come from the statistical side of random matrix theory [4, 11]. Our result and proof generalizes those of [1, 3], the latter of which plays a major role in the theoretical guarantees of [6, 15]. We expect that our formula and techniques can be used to explain, in more detail, empirically observed phenomena in randomized methods for computing SVDs, including randomized Krylov iteration. We also expect that our formula can be used to analyze the possibility of using the rSVD as a test for low-rank structure, which has come up in [10, 16], or as a rank revealer. Finally, we hope to use our result to distinguish the classes of matrices for which randomized subspace methods succeed or fail at different tasks, including singular value and singular subspace approximation.

## References

- [1] ABSIL, P.-A., EDELMAN, A., AND KOEV, P. On the largest principal angle between random subspaces. *Linear Algebra its Appl.* 414 (2006), 288–294.
- [2] ALLEN-ZHU, Z., AND LI, Y. First efficient convergence for streaming  $k$ -PCA: A global, gap-free, and near-optimal rate. In *IEEE 58th Annual Symposium on Foundations of Computer Science* (2017), pp. 487–492.
- [3] CHEN, Z., AND DONGARRA, J. Condition numbers of Gaussian random matrices. *SIAM J. Matrix Anal. Appl.* 26, 3 (2005), 1389–1404.
- [4] CHIKUSE, Y. *Statistics on Special Manifolds*. Springer, 2003.
- [5] DONG, Y., MARTINSSON, P.-G., AND NAKATSUKASA, Y. Efficient bounds and estimates for canonical angles in randomized subspace approximations. *SIAM J. Matrix Anal. Appl.* 45, 4 (2024), 1978–2006.
- [6] HALKO, N., MARTINSSON, P.-G., AND TROPP, J. A. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev.* 53, 2 (2011), 217–288.
- [7] KNYAZEV, A. V. Sharp a priori error estimates of the Rayleigh–Ritz method without assumptions of fixed sign or compactness. *Math. Notes* 38 (1985), 998–1002.
- [8] KOEV, P., AND EDELMAN, A. The efficient evaluation of the hypergeometric function of a matrix argument. *Math. Comput.* 75, 254 (2006), 833–846.
- [9] MASSEY, P. Admissible subspaces and the subspace iteration method. *BIT Numer. Math.* 64, 1 (2024).
- [10] MEIER, M., AND NAKATSUKASA, Y. Fast randomized numerical rank estimation for numerically low-rank matrices. *Linear Algebra its Appl.* 686 (2024), 1–32.
- [11] MUIRHEAD, R. J. *Aspects of Multivariate Statistical Theory*. John Wiley & Sons, 1982.
- [12] MUSCO, C., AND MUSCO, C. Randomized block Krylov methods for stronger and faster approximate singular value decomposition. In *Advances in Neural Information Processing Systems* (2015), vol. 28.
- [13] NAKATSUKASA, Y. Accuracy of singular vectors obtained by projection-based SVD methods. *BIT Numer. Math.* 57 (2017), 1137–1152.
- [14] NAKATSUKASA, Y. Sharp error bounds for Ritz vectors and approximate singular vectors. *Math. Comput.* 89, 324 (2020), 1843–1866.
- [15] SAIBABA, A. K. Randomized subspace iteration: Analysis of canonical angles and unitarily invariant norms. *SIAM J. Matrix Anal. Appl.* 40, 1 (2019), 23–48.
- [16] WANG, C., AND TOWNSEND, A. Operator learning for hyperbolic partial differential equations. Preprint arXiv:2312.17489 (2023).

# Bulge Chasing is Pole Swapping

*David S. Watkins*

## Abstract

For at least fifty years, the dominant work-horse algorithms for solving small to medium-sized eigenvalue problems have been variants of Francis's implicitly-shifted QR algorithm, including the Moler-Stewart QZ algorithm and refinements. These are bulge-chasing algorithms. They create bulges at one end of the (Hessenberg) matrix or pencil and chase them to the other end. A few years ago a new class of algorithms, pole-swapping algorithms, was introduced by Camps, Meerbergen, Vandebril, and others. It turns out that pole swapping is a generalization of bulge chasing. It might happen that new pole-swapping codes will supplant the current QR and QZ codes in the major software packages. Whether this turns out to be true or not, the pole-swapping viewpoint is extremely valuable for a detailed understanding of what makes this class of algorithms, both bulge-chasing and pole-swapping, work. The purpose of this talk is to describe pole-swapping algorithms briefly and explain what makes them tick. Every expert in the field of eigensystem computations should be in possession of this information.

# Structured Representations of Rational Functions for Learning Mechanical Dynamical Systems: A Barycentric Approach

*Steffen W. R. Werner, Michael S. Ackermann, Ion Victor Gosea, Serkan Gugercin*

## Abstract

In recent years, the importance of learning dynamical systems from data has emerged as a pivotal area of research, bridging the realms of mathematics, engineering, and data science. Dynamical systems, which describe how states evolve over time based on underlying mathematical relations, are fundamental to understanding a wide range of time-dependent phenomena—from physics and biology to economics and social sciences. For the use of these systems in practical applications like predictive simulations and control, high modeling accuracy as well as interpretability and explainability are essential. While high accuracy of models can usually be achieved by the incorporation of data from simulations or real-world measurements, the interpretability and explainability are typically not given in most blackbox and unstructured modeling approaches. In this work, we propose a new framework of data-driven modeling algorithms based on a novel representation of rational functions leading that allows us in the case of mechanical applications the modeling of accurate dynamical systems from given data while providing a structured system representation, which gives physical meaning to the terms describing the dynamical system.

The dynamical systems that we are interested in are given via second-order ordinary differential equations of the form

$$M\ddot{x}(t) + D\dot{x}(t) + Kx(t) = bu(t), \quad y(t) = c^T x(t), \quad (1)$$

with  $M, D, K \in \mathbb{R}^{n \times n}$  and  $b, c^T \in \mathbb{R}^n$ . Thereby, the function  $u: \mathbb{R} \rightarrow \mathbb{R}$  models the external inputs that allow us to interfere with the internal system behavior given by the states  $x: \mathbb{R} \rightarrow \mathbb{R}^n$ . Typically, one cannot observe the complete state behavior but has access to a low-dimensional output  $y: \mathbb{R} \rightarrow \mathbb{R}$  modeling quantities of interest of the system. The unique format of (1) usually appears in applications with mechanical structures, acoustic phenomena or electro-mechanical components. Consequently, the matrices in (1) can be associated with a certain physical meaning:  $M$  is describing the distribution of mass in the system,  $D$  yields the dissipation or preservation of energy, and  $K$  explains the forces between the different components of the system. An equivalent description of (1) is given in the complex frequency domain by taking the Laplace transformation of (1) leading to the system's transfer function

$$H(s) = c^T(s^2M + sD + K)^{-1}b, \quad (2)$$

with  $s \in \mathbb{C}$ . The function  $H: \mathbb{C} \rightarrow \mathbb{C}$  in (2) is at its core a complex rational function with a structured representation. In the case of the aforementioned applications, data is typically given in form of transfer function measurements

$$H(\mu_1) = h_1, \quad H(\mu_2) = h_2, \quad \dots, \quad H(\mu_N) = h_N. \quad (3)$$

With all these components, the structured data-driven modeling problem that we consider in this work reads as follows: Find a transfer function  $\hat{H}$  that has the same structure as (2) and that approximates the given data (3) like

$$\hat{H}(\mu_1) \approx h_1, \quad \hat{H}(\mu_2) \approx h_2, \quad \dots, \quad \hat{H}(\mu_N) \approx h_N. \quad (4)$$

To solve the structured data-driven modeling problem, we have extended key tools from numerical linear algebra that have been used for the unstructured modeling problem before. In the unstructured case, linear dynamical systems are given in the form

$$E\dot{x}(t) = Ax(t) + bu(t), \quad y(t) = c^T x(t), \quad (5)$$

with  $E, A \in \mathbb{R}^{n \times n}$  and  $b, c^T \in \mathbb{R}^n$ , and the corresponding transfer function

$$G(s) = c^T(sE - A)^{-1}b. \quad (6)$$

Many efficient and effective methods for the modeling of transfer functions  $\hat{G}$  of the form (6) from data (3), utilize a reformulation of (6) into its barycentric form

$$G(s) = \frac{\sum_{k=1}^n \frac{h_k \omega_k}{(s-\lambda_k)}}{1 + \sum_{k=1}^n \frac{\omega_k}{(s-\lambda_k)}}, \quad (7)$$

where  $\lambda_k \in \mathbb{C}$  are the support points,  $h_k \in \mathbb{C}$  function values and  $\omega_k \in \mathbb{C}$  the barycentric weights. This representation (7) eases the problem of fitting data significantly as it allows interpolation by construction and provides desired numerical properties in least squares problems which become linear systems with Loewner matrices. Consequently, popular data-driven modeling approaches are based on (7). Enforcing interpolation in all given data leads to the Loewner framework [1], matching the data in a least-squares sense results in the vector fitting method [3], and mixing interpolation conditions for parts of the data with a least square fit for the rest yields the AAA algorithm [4]. Due to the classical barycentric form (7) corresponding to unstructured systems (5), the models obtained via these approaches typically cannot be rewritten into the second-order form (1) even when the data was coming from a mechanical application.

With the barycentric form (7) being the key component in the data-driven modeling approaches above, we developed a new structured variant of the barycentric form corresponding to the second-order transfer function (2). The structured transfer function (2) can be written in the form

$$H(s) = \frac{\sum_{k=1}^n \frac{h_k \omega_k}{(s-\lambda_k)(s-\sigma_k)}}{1 + \sum_{k=1}^n \frac{\omega_k}{(s-\lambda_k)(s-\sigma_k)}}, \quad (8)$$

where  $\lambda_k \in \mathbb{C}$  are support points,  $h_k \in \mathbb{C}$  are function values and  $\omega_k \in \mathbb{C}$  are barycentric weights as in the classical variant (7); see [2]. In contrast to (7), the new structured form has an additional set of parameters  $\sigma_k \in \mathbb{C}$  that we denote as quasi-support points. The structured barycentric form (8) shares important properties with the classical variant (7), in particular the interpolation of the data  $(\lambda_k, h_k)_{k=1}^n$  by construction, such that it can be similarly used as the backbone of data-driven modeling algorithms. Additionally, second-order systems of the form (1) can easily be recovered from (8) via

$$M = I_n, \quad D = -\Lambda - \Sigma, \quad K = b\mathbf{1}_n^T + \Lambda\Sigma, \quad b = [w_1 \ \dots \ w_n]^T, \quad c = [h_1 \ \dots \ h_n]^T,$$

where  $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$  and  $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_n)$  are diagonal matrices containing the support and quasi-support points, and  $I_n$  and  $\mathbf{1}_n$  denote the  $n$ -dimensional identity matrix and the vector of all ones of length  $n$ , respectively; see [2] for more details.

Based on the structured barycentric form (8), we can now develop new approaches that solve the structured data fitting problem (4). Previously, we introduced a new structured version of the Loewner framework based on (8) in [2], in which the use of (8) leads to linear systems of Loewner-like matrices to be solved to match additional interpolation conditions. In this work, we will provide an extension of the AAA algorithm for the structured second-order case. To this end, we consider a similar step-by-step construction of a lower dimensional model in barycentric form, interpolating in the most important data points and approximating the rest of the data (3) effectively in a least-squares sense for which we need to solve nonlinear least-squares problems with Loewner-like matrices. We will provide a variety of numerical examples including the vibrational response of a plate and the sound behavior of an acoustic cavity to show that the proposed approach is capable of efficiently constructing low-dimensional high-fidelity models from given data that are interpretable and explainable as second-order systems (1).

## References

- [1] A. C. Antoulas and B. D. O. Anderson. On the scalar rational interpolation problem. *IMA J. Math. Control Inf.*, 3(2–3):61–8, 1986. <https://doi.org/10.1093/imamci/3.2-3.61>
- [2] I. V. Gosea, S. Gugercin, and S. W. R. Werner. Structured barycentric forms for interpolation-based data-driven reduced modeling of second-order systems. *Adv. Comput. Math.*, 50(2):26, 2024. <https://doi.org/10.1007/s10444-024-10118-7>
- [3] B. Gustavsen and A. Semlyen. Rational approximation of frequency domain responses by vector fitting. *IEEE Trans. Power Del.*, 14(3):1052–1061, 1999. <https://doi.org/10.1109/61.772353>
- [4] Y. Nakatsukasa, O. Sète, and L. N. Trefethen. The AAA algorithm for rational approximation. *SIAM J. Sci. Comput.*, 40(3):A1494–A1522, 2018. <https://doi.org/10.1137/16M1106122>

# A Time-Frequency Method for Acoustic Scattering in Unfriendly Domains

*Heather Wilber, Abi Gopal, Gunnar Martinsson, Wietse Vaes*

## Abstract

The acoustic scattering problem asks for the recovery of a scattered wavefield that is produced when an incoming incident wavefield strikes a scattering object. If sound-soft boundary conditions are imposed and the incident wavefield  $u_{inc}$  is pulse-like and away from the scatterer at  $t = 0$ , then the scattered field  $u$  satisfies the following:

$$\frac{\partial^2 u}{\partial t^2}(x, t) - c^2 \Delta u(x, t) = 0, \quad (x, t) \in \Omega \times [0, T], \quad (1)$$

$$u(x, 0) = \frac{\partial u}{\partial t}(x, 0) = 0, \quad x \in \Omega, \quad (2)$$

$$u(x, t) = -u_{inc}(x, t), \quad (x, t) \in \partial\Omega \times [0, T]. \quad (3)$$

Here,  $c$  is the wave speed associated with the domain,  $\Omega$  is an exterior domain, and  $[0, T]$  is some relevant time period over which the scattered waves are observable. This problem is challenging to solve for a number of reasons. Domain discretization schemes must manage the fact that the exterior domain is unbounded as  $|x| \rightarrow \infty$  and impose artificial absorbing boundary layers that appropriately handle outgoing waves. Traditional direct-in-time methods must combat pervasive issues related to accumulating dispersive error and potentially prohibitive restrictions on time step sizes. When the domain includes corners, cusps, or trapping regions where the scattered waves can get stuck and decay very slowly, these issues become severely exacerbated.

Under the condition that the incident wavefield is enveloped by a Gaussian or is otherwise approximately bandlimited, some of the complications of direct-in-time methods can be avoided by using *hybrid time-frequency solvers* [1, 8]. This class of methods considers an equivalent problem in the frequency domain. If  $\hat{U}(x, \omega)$  is the Fourier transform of  $u(x, t)$  with respect to time, then it follows that  $\hat{U}$  satisfies the Helmholtz equation with a continuously parametrized wavenumber:

$$\Delta \hat{U}(x, \omega) + \frac{\omega^2}{c^2} \hat{U}(x, \omega) = 0, \quad x \in \Omega, \quad (4)$$

$$\hat{U}(x, \omega) = -\hat{U}_{inc}(x, \omega), \quad x \in \partial\Omega, \quad (5)$$

The Sommerfeld radiation condition is additionally imposed at all  $\omega$ . Now  $u(x, t)$  can be expressed in terms of an inverse Fourier transform. If  $\hat{U}(x, \omega)$  is sufficiently small for all  $x$  whenever  $\omega$  is outside the band  $[W_1, W_2]$ , then

$$u(x, t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \hat{U}(x, \omega) e^{-i\omega t} d\omega \approx \frac{1}{2\pi} \int_{W_1}^{W_2} \hat{U}(x, \omega) e^{-i\omega t} d\omega. \quad (6)$$

The primary task in these methods is the evaluation of the integral in (6) via quadrature, which in turn requires the solving of the Helmholtz equation at quadrature points  $\{\omega_1, \omega_2, \dots, \omega_N\}$ , and then the evaluation of the solutions over all domain points of interest. There are major advantages to this formulation, including the ability to compute solutions that are virtually free of dispersive error, and the ability to evaluate  $u(x, t)$  at arbitrary points in time (time-skipping). Moreover, solutions to the Helmholtz equation can be expresesd via boundary integral equations that nicely handle the exterior domain by reducing it to a contour integral around the boundary of the scatterer [7].

However, there is no free lunch! These methods become prohibitively expensive or intractable when the domain involves corners, trapping regions, or both. We introduce developments that overcome these challenges so that hybrid time-frequency methods are effective in domains with these “unfriendly” features. Example domains where our methods work well include multiply-connected regions (e.g., whisper galleries or panel sets), as well as keyhole regions with severe trapping, multiple corners, and long channels.

Our work involves two major developments that make hybrid time-frequency methods effective in unfriendly domains:

*For trapping regions:* A damping+correction fast quadrature scheme for the inverse Fourier transform that combines contour integration with fast transforms (the FFT and the fast sinc transform) to make evaluation of the integral in (6) tractable even when there is trapping.

*For domains with corners:* State-of-the-art hierarchical linear algebra routines based on recursive skeletonization and the balanced use of “universal skeletons” for solving and evaluating the broadband Helmholtz equation. Excellent efficiency makes it possible to handle discretizations with many points (e.g., induced by refinement into corners).

We briefly discuss these two developments, starting with a method to make evaluation feasible in domains with trapping.

There are two reasons that the integral in (6) can be challenging to discretize efficiently. First, if  $t$  is large, the integral becomes highly oscillatory and naive discretizations (e.g., Gauss-Legendre quadrature) require many points to capture the oscillations. Coupled to this problem is the fact that when  $u(x, t)$  is in a trapping region and thus decays slowly over time,  $\hat{U}(x, \omega)$  has poles in the complex plane very near to the interval  $[W_1, W_2]$ . These poles are associated with near-resonant modes of the Helmholtz operator. Since  $\hat{U}(x, \omega)$  has only a very small region of analyticity that encloses  $[W_1, W_2]$ , it is inherently difficult to approximate with polynomials.

To improve upon the analyticity properties, we note that the poles of  $\hat{U}(x, \omega)$  always lie below the real line. This suggests the consideration of the perturbed function  $\hat{U}(x, \omega + i\delta)$ , where  $\delta > 0$ . One can view this perturbation as the introduction of mild damping into the solution. Since  $\hat{U}(x, z)$  is analytic with respect to  $z$  in the upper half plane, we can apply Cauchy’s integral theorem on a rectangular contour in the upper half plane and represent the undamped solution as

$$\int_{W_1}^{W_2} \hat{U}(x, \omega) e^{-i\omega t} d\omega = - \underbrace{\int_{W_1}^{W_2} \hat{U}(x, \omega + \delta i) e^{-i(\omega + \delta i)t} d\omega}_{I_\delta} - I_{cL} - I_{cR}, \quad (7)$$

where the correction terms  $I_{cL}$  and  $I_{cR}$  are integrals along the vertical sides of the rectangle. There is an inherent upper limit on  $\delta$ . We show that if it is taken too large, the correction terms cannot be stably evaluated. However, the room afforded by  $\delta$  is generous enough that when paired with a fast evaluation scheme for the integral  $I_\delta$ , we are able to solve the acoustic scattering problem even with severe trapping over long time horizons.

The integral  $I_\delta$  is still problematic in that it can be highly oscillatory when  $t$  is large. To manage this, we follow an idea originally presented in [1] that is also related to sampling schemes for bandlimited functions [6]. We approximate the periodization of  $\hat{U}(x, \omega + \delta i)$  with a trigonometric polynomial. Critically, this is possible because of the improved analyticity afforded by damping. Substituting this approximation into the integral  $I_\delta$  results in a simplification of the integral so

that for each fixed  $x$ ,

$$I_\delta \approx \frac{W_2 - W_1}{2\pi(2m+1)} e^{-it((W_2-W_1)/2+W_1+i\delta)} \sum_{j=-m}^m (-1)^j c_j \text{sinc}\left(\frac{(W_2-W_1)t}{2\pi} - j\right), \quad (8)$$

where  $\text{sinc}(x) = \sin(\pi x)/(\pi x)$ . The set of coefficients  $\{c_j\}_{j=-m}^m$  is  $x$ -dependent and can be found using the FFT on equally spaced samples of  $\hat{U}(x, \omega + \delta i)$  over the band  $[W_1, W_2]$ . Rather than naively evaluating the sum, we apply the fast sinc transform [5] (via the nonuniform FFT of type-III [2]) to evaluate the weighted sum of sincs at locations  $\{t_1, t_2, \dots, t_N\}$  in only  $\mathcal{O}(m \log m + N)$  operations. Note that the complexity of this method no longer depends linearly on  $t$ , as would be true with a naive discretization. Instead, it depends on  $m$ , which has an implicit but much weaker dependence on  $t$  since the size of  $t$  for which  $u(x, t)$  is relevant is correlated to the distance of  $[W_1, W_2] + \delta i$  from the poles associated with near-resonant modes.

Even with damping, the discretization of  $I_\delta$  still requires  $2m+1$  solves of the Helmholtz equation, and then evaluations of each of these solutions at every point  $x$  of interest in the domain. Corners induced by singularities in the solutions require special care. They can be handled with highly specialized quadrature that requires precise knowledge about the geometry [3, 9], or they can be handled with refinement strategies in the Nyström discretization of the integral. The latter is practical, but makes the solve and evaluation steps much more expensive. To handle the expense, we employ so-called *universal skeletons* in a fast solver based on recursive skeletonization. Evaluations are then handled with the fast multipole method. Universal skeletons were first introduced in [4]. They supply a way to reuse low rank approximation factors in hierarchical discretizations of boundary integral equations as the wavenumber parameter changes. We discuss how they can be adapted and applied in balanced and effective ways in the context of the acoustic scattering problem.

## References

- [1] Thomas G Anderson, Oscar P Bruno, and Mark Lyon. High-order, dispersionless “fast-hybrid” wave equation solver. part i:  $\mathcal{O}(1)$  sampling cost via incident-field windowing and recentering. *SIAM J. Sci. Comput.*, 42(2):A1348–A1379, 2020.
- [2] Alex Barnett and J. Magland. Non-uniform fast Fourier transform library of types 1, 2, 3 in dimensions 1, 2, 3, 2018.
- [3] Oscar P Bruno, Jeffrey S Ovall, and Catalin Turc. A high-order integral algorithm for highly singular PDE solutions in Lipschitz domains. *Computing*, 84:149–181, 2009.
- [4] Abinand Gopal and Per-Gunnar Martinsson. Broadband recursive skeletonization. In *Spectral and High Order Methods for Partial Differential Equations ICOSAHOM 2020+ 1: Selected Papers from the ICOSAHOM Conference, Vienna, Austria, July 12-16, 2021*, pages 31–66. Springer, 2022.
- [5] Leslie Greengard, June-Yub Lee, and Souheil Inati. The fast sinc transform and image reconstruction from nonuniform samples in k-space. *Communications in Applied Mathematics and Computational Science*, 1(1):121–131, 2007.
- [6] Melanie Kircheis, Daniel Potts, and Manfred Tasche. On numerical realizations of Shannon’s sampling theorem. *Sampling Theory, Sig. Proc., and Data Anal.*, 22(1):1–33, 2024.

- [7] Per-Gunnar Martinsson. *Fast direct solvers for elliptic PDEs*. SIAM, 2019.
- [8] Eleonora Mecocci, Luciano Misici, Maria C Recchioni, and Francesco Zirilli. A new formalism for time-dependent wave scattering from a bounded obstacle. *J. Acoustical Soc. of America*, 107(4):1825–1840, 2000.
- [9] Kirill Serkh. On the solution of elliptic partial differential equations on regions with corners II: detailed analysis. *Appl. and Comp. Harmonic Anal.*, 46(2):250–287, 2019.

# Data-driven Numerical Methods for Kernel Matrices

*Yuanzhe Xi, Difeng Cai, Tianshi Xu, Hua Huang, Edmond Chow*

## Abstract

Kernel matrices play a pivotal role in various machine learning and scientific applications, with their structure critically influenced by both the parameters of the kernel function and the data distribution [2]. This talk will begin with a geometric analysis of the Schur complement of the kernel matrix, examining the effects of kernel bandwidth and data distribution on its structure. Building on these geometric insights, we design the Adaptive Factorized Nyström (AFN) preconditioner [1] for solving linear systems associated with the regularized kernel matrix. The AFN preconditioner enhances the Nyström approximation by constructing a sparse approximate inverse for the Schur complement, significantly improving robustness and efficiency across a wide range of parameters. Finally, we will introduce HiGP [4], a high-performance Python package designed for Gaussian Process Regression (GPR) and Classification (GPC). HiGP integrates AFN and some preconditioned iterative methods [3] to boost the efficiency and scalability of model training and inference across various datasets.

## References

- [1] S. ZHAO, T. XU, H. HUANG, E. CHOW, AND Y. XI, *An Adaptive Factorized Nyström Preconditioner for Regularized Kernel Matrices*, SIAM J. Sci. Comput., 46(4), (2024), A2351-A2376.
- [2] D. CAI, H. HUANG, E. CHOW, AND Y. XI, *Data-Driven Construction of Hierarchical Matrices With Nested Bases*, SIAM Journal on Scientific Computing, 46(2), (2024), S24-S50.
- [3] D. CAI, E. CHOW, AND Y. XI, *Posterior Covariance Structures in Gaussian Processes*, arXiv preprint arXiv:2408.07379.
- [4] H. HUANG, T. XU, Y. XI, AND E. CHOW, *HiGP: High-Performance Python Package for Gaussian Process Regression*, Retrieved from <https://github.com/huanghua1994/HiGP>

# Matrix Analysis and Fast Solvers for Neural Network Computations

Jianlin Xia

## Abstract

Neural networks provide a powerful tool for machine learning and other data science techniques. They can also serve as new ways for mathematical and numerical tasks such as function approximations and PDE solutions. Although there have been significant developments in neural network methods, the analysis of relevant matrices and the design of relevant fast and stable matrix computation techniques are typically overlooked.

In fact, neural network methods provide highly interesting new opportunities to perform matrix analysis and design matrix algorithms that can benefit modern data analysis and machine learning. Examples of scenarios where large and challenging matrices arise include the following.

- In neural network least-squares approximations of functions, large mass matrices and Hessian matrices may be constructed from activation functions such as ReLU functions as basis functions.
- Sparse structured matrices have been often used in the design of effective neural networks and efficient training algorithms (and a simple example is the use of sparse Toeplitz matrices as weight matrices in some neural networks).
- In optimization and training algorithms such as ADAM and BFGS, the underlying matrices are often closely related to certain preconditioners.

In this talk, we aim to *bridge the gap between some neural network methods and fast and reliable matrix computations*. We present rigorous analysis for some of these matrices and show two aspects.

- Why some of these matrices pose significant challenges (say, in the conditioning, spectrum distribution, and frequency modes) for matrix computations.
- Why it is feasible to design new fast and reliable solvers for these problems based on certain underlying structures.

In particular, consider the approximation of a function  $u : \Omega(\subset \mathbb{R}^d) \rightarrow \mathbb{R}$  by

$$v = \sum_{i=1}^n c_i \sigma(\mathbf{w}_i^T \mathbf{x} + b_i), \quad \mathbf{x} \in \Omega,$$

where  $\mathbf{w}_i$ 's are weight vectors,  $b_i$ 's are biases,  $c_i$ 's are scalar coefficients, and  $\sigma(t) = \max\{t, 0\}$  is the ReLU function. Let  $W = (\mathbf{w}_1, \dots, \mathbf{w}_n)$ ,  $\mathbf{b} = (b_1, \dots, b_n)^T$ , and  $\mathbf{c} = (c_1, \dots, c_n)$ . The least-squares approximation of  $u$  by  $v$  solves the following optimization problem:

$$\min_{W, \mathbf{b}, \mathbf{c}} \mathcal{J} \quad \text{with} \quad \mathcal{J} := \langle v - u, v - u \rangle = \frac{1}{2} \int_{\Omega} (v - u)^2 d\mathbf{x}.$$

By viewing  $\mathbf{c}$  as a set of linear parameters and  $\{W, \mathbf{b}\}$  as nonlinear parameters, we can look at the gradient of  $\mathcal{J}$  with respect to one of the two sets of parameters with the other set fixed [1]. Setting

the gradients to be 0 leads to a linear system for the linear parameters and a nonlinear system for the nonlinear parameters. The former system has a mass matrix  $\tilde{A}$  as the coefficient matrix. The solution of the latter system with Newton or Gauss-Newton methods lead to linear systems involving a Hessian or Gauss-Newton matrix  $\tilde{H}$ . In a highly simplified setting,  $\tilde{A}$  and  $\tilde{H}$  may be related to the following matrix forms, respectively:

$$A = (A_{ij})_{n \times n}, \quad A_{ij} = \langle \sigma(\mathbf{w}_i^T \mathbf{x} + b_i), \sigma(\mathbf{w}_j^T \mathbf{x} + b_j) \rangle,$$

$$H = (H_{ij})_{n \times n}, \quad H_{ij} = \langle h(\mathbf{w}_i^T \mathbf{x} + b_i), h(\mathbf{w}_j^T \mathbf{x} + b_j) \rangle,$$

where  $h(t) = \sigma'(t) = \begin{cases} 1, & t > 0, \\ 0, & t < 0. \end{cases}$

Some interesting matrix analysis may be performed for  $A$  and  $H$ . For example, we can show the following aspects.

- $A$  and  $H$  are positive definite (with modest assumptions) but are highly ill conditioned due to the fast decay of the eigenvalues. For instance, even in the 1D case with uniform breakpoints that define the ReLU basis functions,  $A$  has 2-norm condition number proportional to  $\frac{1}{n^4}$ .
- The behaviors of the low and high frequency modes further make them challenging for iterative solvers.
- Some preconditioning strategies may be designed based on basis function modifications, but the effectiveness is limited.
- On the other hand, the matrices have nice inherent structures. In particular, relevant off-diagonal blocks of the matrices are low rank (the 1D case) or numerically low rank (with appropriate conditions). These structures make it feasible to design fast and stable direct solvers for the relevant linear systems.

These problems thus give a nice opportunity to apply structured matrix methods. This also shows how advanced matrix analysis may benefit modern neural network methods. The talk includes joint with Z. Cai, T. Ding, M. Liu, and X. Liu.

## References

- [1] Z. CAI, T. DING, M. LIU, X. LIU, AND J. XIA, *A structure-guided Gauss-Newton method for shallow ReLU neural network*, arXiv:2404.05064, submitted to SIAM J. Sci. Comput., (2024).

# A block conjugate gradient method with polynomial filters for symmetric eigenvalue problems: practice and global quasi-optimality

*Fei Xue, Tianqi Zhang*

## Abstract

We propose a new block variant of the preconditioned conjugate gradient (PCG) method for computing the lowest eigenvalues of standard symmetric ( $Av = \lambda v$ ) and product eigenvalue problems ( $KMv = \lambda^2 v$ ) [1] that arise, for example, from the Bethe-Salpeter equation. The algorithm combines the well-known strengths of locally optimal PCG [4] and Chebyshev polynomial filter methods [2, 3] to exhibit robust and rapid convergence for computing potentially many lowest eigenvalues. The convergence of the new method is not very sensitive to the quality of the preconditioner or the parameters of the polynomial filter, which is usually critical for achieving good performance of PCG methods and polynomial-based algorithms. Numerical experiments show its improved robustness and runtime compared to several other algorithms, such as the *M*-LOBPCG [6], Chebyshev-Davidson [2, 3] and LOBP4DCG [5].

On the theoretical side, we show that the ideal version of this algorithm (and similar others) exhibits a *global quasi-optimality* if the starting vector is not far from the eigenvector associated with the lowest eigenvalue: the Rayleigh quotient of the iterates computed by this locally optimal algorithm is close to the one computed by the corresponding globally optimal algorithm in early iterations, until the latter eventually outperforms. Such a behavior is similar to the global optimality of the conjugate gradient method for solving a symmetric positive definite system of linear equations [8, Section 5.1]. This theory provides insight into the competitiveness of the family of locally optimal methods with search directions, such as LOBPCG and thick-restarted Lanczos +  $k$  methods [7].

## References

- [1] T. Zhang and F. Xue, *A Chebyshev locally optimal block preconditioned conjugate gradient method for product and standard symmetric eigenvalue problems*, to appear on SIAM. J. Matrix Anal. Appl., 2024.
- [2] Y. Zhou and Y. Saad, *A Chebyshev-Davidson algorithm for large symmetric eigenproblems*, SIAM. J. Matrix Anal. Appl., Vol. 29 (2007), pp. 954-971.
- [3] Y. Zhou, *A block Chebyshev-Davidson method with inner-outer restart for large eigenvalue problems*, J. Comput. Phys., Vol. 229 (2010), pp. 9188-9200.
- [4] A. V. Knyazev, *Toward the optimal preconditioned eigensolver: locally optimal block preconditioned conjugate gradient method*, SIAM. J. Sci. Comput., Vol. 23 (2001), pp. 517-541.
- [5] Z. Bai and R.-C. Li, *Minimization principles for the linear response eigenvalue problem II: Computation*, Vol. 34 (2013), pp. 392-416.
- [6] E. Vecharynski, J. Brabec, M. Shao, N. Govind, and C. Yang, *Efficient block preconditioned eigensolvers for linear response time-dependent density functional theory*, Comput. Phys. Commun., Vol. 221 (2017), pp. 42-52.

- [7] L. Wu, F. Xue, and A. Stathopoulos, *TRPL+k: Thick-restart preconditioned Lanczos+k method for large symmetric eigenvalue problems*, SIAM. J. Sci. Comput., Vol. 41 (2019), pp. A1013-A1040.
- [8] J. Nocedal and S. J. Wright, *Numerical Optimization (2nd edition)*, Springer, New York, 2006.

# Randomized Algorithms for the Simultaneous Compression and LU Factorization of Hierarchical Matrices

Anna Yesypenko, Per-Gunnar Martinsson

## Abstract

Dense matrices related to solving elliptic PDEs often have internal structure that allows for the linear system  $\mathbf{Ax} = \mathbf{b}$  to be solved in linear or near-linear time. The hierarchical matrix ( $\mathcal{H}^2$ -matrix) formalism [1] partitions  $\mathbf{A}$  into blocks, with each block small enough to be stored densely or of numerically low rank  $k$ , where  $k$  controls the accuracy of the approximation. This format enables matrix-vector products to be computed to a desired accuracy in linear or near-linear time, allowing for the rapid solution of dense linear systems using iterative methods when  $\mathbf{A}$  is well-conditioned. However, in many situations, the matrix  $\mathbf{A}$  is ill-conditioned, and iterative methods do not offer satisfactory performance.

The objective of this work is to compute an  $\mathcal{H}^2$  approximation to the inverse  $\mathbf{A}^{-1}$ , enabling the fast solution of ill-conditioned dense linear systems. Previously, the use of invertible factorizations of  $\mathcal{H}^2$  matrices has been limited due to two challenges:

1. **Compression.** Although a fast matrix vector product for  $\mathbf{A}$  is often available, individual entries may be challenging to access, which leads to challenges in compressing  $\mathbf{A}$ .
2. **Invertible Factorization.** While invertible factorization algorithms are efficient for specialized formats like HBS/HSS matrices [2, 4, 7], inversion algorithms for general  $\mathcal{H}^2$ -matrices typically involve nested recursions and recompressions, making efficient implementation challenging.

This work [8] describes a novel algorithm for *simultaneously* compressing and inverting an  $\mathcal{H}^2$ -matrix and extends the applicability of  $\mathcal{H}^2$  inversion algorithms [5, 6] to a generic class of dense matrices for which there is a means of applying the matrix and its adjoint to vectors. The method leverages the randomized SVD (RSVD) and novel sketching techniques [3] to efficiently recover numerically low-rank sub-blocks of the matrix  $\mathbf{A}$  within a hierarchical framework by reusing random sketches of the matrix. The precise problem formulation is:

Suppose that  $\mathbf{A} \in \mathbb{R}^{N \times N}$  is an  $\mathcal{H}^2$ -matrix, and that you are given a fast method for applying  $\mathbf{A}$  and its adjoint  $\mathbf{A}^*$  to vectors. We are also given geometric information corresponding to the matrix  $\mathbf{A}$ , e.g. entry  $\mathbf{A}_{ij}$  corresponds to interactions between points  $\mathbf{x}_i$  and  $\mathbf{x}_j$  in 2 or 3-dimensional space. Using test matrices  $\Omega$  and  $\Psi$  and the set of matrices  $\{\mathbf{Y}, \mathbf{Z}, \Omega, \Psi\}$ , where

$$\mathbf{Y}_{N \times s} = \mathbf{A}_{N \times N} \mathbf{\Omega}_{N \times s} \quad \text{and} \quad \mathbf{Z}_{N \times s} = \mathbf{A}_{N \times N}^* \mathbf{\Psi}_{N \times s},$$

the framework constructs an invertible  $\mathcal{H}^2$  factorization  $\mathbf{K}$  such that  $\mathbf{K}^{-1} \approx \mathbf{A}^{-1}$ .

The number of samples  $s$  needed to construct an  $\mathcal{H}^2$  approximation of the inverse  $\mathbf{A}^{-1}$  is independent of  $N$  and depends only on the chosen rank parameter  $k$  as well as properties of the geometry. The numerical results demonstrate the effectiveness of the algorithm across a range of problems, including the discretization of partial differential equations (PDEs) and boundary integral equations (BIEs) in both 2D and 3D. The results demonstrate the algorithm's performance in terms of speed, memory efficiency, and precision, as well as its robustness in handling indefiniteness and ill-conditioning.

## References

- [1] Steffen Börm, Lars Grasedyck, and Wolfgang Hackbusch. Introduction to hierarchical matrices with applications. *Engineering analysis with boundary elements*, 27(5):405–422, 2003.
- [2] Kenneth L Ho and Leslie Greengard. A fast direct solver for structured linear systems by recursive skeletonization. *SIAM Journal on Scientific Computing*, 34(5):A2507–A2532, 2012.
- [3] James Levitt and Per-Gunnar Martinsson. Linear-complexity black-box randomized compression of rank-structured matrices. *SIAM Journal on Scientific Computing*, 46(3):A1747–A1763, 2024.
- [4] Per-Gunnar Martinsson and Vladimir Rokhlin. A fast direct solver for boundary integral equations in two dimensions. *Journal of Computational Physics*, 205(1):1–23, 2005.
- [5] Victor Minden, Kenneth L Ho, Anil Damle, and Lexing Ying. A recursive skeletonization factorization based on strong admissibility. *Multiscale Modeling & Simulation*, 15(2):768–796, 2017.
- [6] Daria Sushnikova, Leslie Greengard, Michael O’Neil, and Manas Rachh. FMM-LU: A fast direct solver for multiscale boundary integral equations in three dimensions. *Multiscale Modeling & Simulation*, 21(4):1570–1601, 2023.
- [7] Jianlin Xia, Shivkumar Chandrasekaran, Ming Gu, and Xiaoye S Li. Fast algorithms for hierarchically semiseparable matrices. *Numerical Linear Algebra with Applications*, 17(6):953–976, 2010.
- [8] Anna Yesypenko and Per-Gunnar Martinsson. Randomized Strong Recursive Skeletonization: Simultaneous compression and factorization of H-matrices in the Black-Box Setting. *arXiv preprint arXiv:2311.01451*, 2023.

# Efficient Classical-Quantum Algorithms for Matrix Encoding

*Liron Mor Yosef, Haim Avron*

## Abstract

We introduce an efficient classical-quantum algorithm for encoding arbitrary dense Hermitian matrices as Block Encoding circuits ( $\mathbf{U}_\mathbf{A} \in \mathbf{BE}_{\alpha,\theta}(\mathbf{A})$ ). Our work is motivated by Block Encoding’s fundamental role as the leading paradigm for quantum linear algebra, providing a unified framework for leveraging quantum computing to accelerate numerical linear algebra operations. Our algorithms, accepts four distinct input representations: (1) classical matrix description  $\mathbf{A} \in \mathbb{C}^{n \times n}$ , (2) an  $4 \times 4 \times \dots \times 4$  ( $\log n$  times) Pauli coefficients tensor  $\mathbf{A}_P$ , (3) matrix state preparation circuit  $\mathcal{U}_\mathbf{A}$ , or (4) matrix state preparation circuit for the Pauli tensor  $\mathcal{U}_{\mathbf{A}_P}$ . This flexibility optimizes performance across different data availability scenarios, with the classical matrix input achieving  $O(n^2 \log n)$  run-time complexity in the worst case. Moreover, the third input model demonstrates a significant breakthrough: the first known method to construct a Block Encoding circuit directly from a matrix state preparation circuit without requiring additional classical information (such as row norms) or additional quantum hardware (such as QRAM). This establishes a new bidirectional equivalence between block encoding and matrix state preparation input models, providing a unified framework for matrix encoding in quantum algorithms.

## 1 Introduction

### 1.1 Motivation and problem statement

Quantum computers hold hope for significant speedups in scientific computing and machine learning due to their ability to handle matrix operations efficiently [3]. However, unlocking this potential hinges the algorithm’s ability to efficiently access classical data within the quantum system. The mechanism in which classical input is fed into a quantum algorithm is known as the algorithm’s *input model*.

Leveraging breakthroughs in quantum linear algebra, researchers have proposed many quantum algorithms for scientific computing and machine learning. However, the feasibility of their input model assumptions remains critical to their effectiveness. As shown by Chakraborty et al. [4], these assumptions often significantly impact the performance and efficiency of such algorithms. Prime examples of quantum linear algebra algorithms include the HHL algorithm [8], and others [5, 7, 14, 2, 11].

Given the essential role of the input model in defining how classical data interacts with the quantum system, researchers have explored various approaches. Two noteworthy examples include the sparse-data access model [1, 5] and various quantum data structure based models [9, 10].

In this work, we study the use of Pauli decomposition in developing efficient algorithms to encode arbitrary dense or sparse Hermitian matrices into Block Encoding circuits, either provided as classical data or as quantum circuits, into Block Encoding circuits.

## 1.2 Brief overview of Block Encoding and State preparation

Chakraborty et al. [4] showed that a variety of the aforementioned widely used input models can be reduced to an input model in which matrices are inputed using *block encodings* and vectors are inputed as *state preparation circuits*:

**Definition 1** (State preparation Circuit). We say that a  $\log_2 n$ -qubit circuit  $\mathcal{U}$  is a *state preparation circuit* for a vector  $\mathbf{x} \in \mathbb{C}^n$  if applying  $\mathcal{U}$  to the state  $|0\rangle_{\log_2 n}$  results in the state  $|\mathbf{x}\rangle_{\log_2 n}$ .

**Definition 2** (Block encoding of a matrix). For  $\alpha \geq 0$  and  $\theta \in [0, 2\pi)$ , a circuit  $\mathcal{U}$  is a  $(\alpha, \theta)$ -*Block Encoding* of  $\mathbf{A} \in \mathbb{C}^{m \times n}$ , denoted as  $\mathcal{U} \in \mathbf{BE}_{\alpha, \theta}(\mathbf{A})$ , if

$$\alpha e^{i\theta} \mathbf{M}(\mathcal{U}) = \begin{bmatrix} \mathbf{A} & * \\ * & * \end{bmatrix}$$

where  $*$  denotes arbitrary entries, and  $\mathbf{M}(\mathcal{U})$  denote the unique unitary matrix of the circuit  $\mathcal{U}$ . We refer to  $\alpha$  as the *scale* and  $\theta$  as the *phase*.

We refer to the input model in which matrices are accessed using block encodings and vectors are accessed as state preparation circuits as the *block encoding input model*. There are powerful algorithms that operate under the block encoding model. In particular, in the block encoding model we can perform Quantum Singular Value Transformation [7], a powerful technique that leads to efficient algorithms for solving linear equations, amplitude amplification, quantum simulation, and more [12].

Another relevant input model is the *state preparation input model*. In this model, matrices accessed via *matrix state preparation circuit* and vectors are accessed via state preparation circuits. Mor-Yosef et al. [15] recently introduced an algorithm for multivariate trace estimation and spectral sums estimation under this model.

**Definition 3** (Matrix state preparation circuit). We say that a  $(\log_2 n + \log_2 m)$ -qubit circuit  $\mathcal{U}$  is a *matrix state preparation circuit* for a matrix  $\mathbf{A} \in \mathbb{C}^{m \times n}$  if applying  $\mathcal{U}$  to the state  $|0\rangle_{\log_2 mn}$  results in the state  $||\mathbf{A}\rangle := |\text{vec}(\mathbf{A})\rangle$ . Equivalently, the first column of  $\mathbf{M}(\mathcal{U})$  is  $\text{vec}(\mathbf{A})$ .

For convenience, where appropriate, we add the matrix as sub-index when denoting state preparation circuits, e.g.  $\mathcal{U}_\mathbf{A}$ . In such cases, with an abuse of notation, the number of gates in  $\mathcal{U}_\mathbf{A}$  is denoted by  $g_\mathbf{A}$ , and the depth by  $d_\mathbf{A}$ .

## 1.3 Statement of main results

While existing block encoding methods typically exploit specific matrix properties like structure, sparsity, or rank, we introduce an efficient general-purpose technique for arbitrary matrices (dense and sparse) through Pauli decomposition. These techniques will utilize the principles of Pauli decomposition, and the use of a quantum multiplexer.

The Pauli decomposition represents matrices as a sum of tensor products of Pauli matrices. Formally, let  $\Sigma = \{\mathbf{I} = 0, \mathbf{X} = 1, \mathbf{Y} = 2, \mathbf{Z} = 3\}$  represent a set of indices corresponding to the four  $2 \times 2$  Pauli matrices:  $\sigma_I, \sigma_X, \sigma_Y, \sigma_Z$ . Assume that  $n = 2^q$ . Given a *word* (i.e., sequence)  $\mathcal{W} = (w_1, w_2, \dots, w_q) \in \Sigma^q$ , we define the corresponding  $q$ -wise Pauli matrix as  $\sigma_{\mathcal{W}} := \sigma_{w_1} \otimes \sigma_{w_2} \otimes \dots \otimes \sigma_{w_q}$ . Then, the Pauli decomposition of matrix  $\mathbf{A}$  can be expressed mathematically as:

$$\mathbf{A} = \sum_{\mathcal{W} \in \Sigma^q} \alpha_{\mathcal{W}} \sigma_{\mathcal{W}}$$

where  $\alpha_{\mathcal{W}} \in \mathbb{R}$  are real coefficients.

The quantum multiplexer [13] acts like a switch within a quantum circuit. It uses control qubits to selectively apply different unitary operations to a target qubit. Given a set of quantum circuits  $\mathcal{U}_0, \dots, \mathcal{U}_{k-1}$  the log  $k$ -qubit multiplexer is defined as:

$$\mathcal{M}\mathcal{X}_{\log k} := \sum_{i=0}^{k-1} |i\rangle\langle i| \otimes \mathcal{U}_i.$$

Importantly,  $\mathcal{M}\mathcal{X}_{\log k}$  acts as a multiplexer. In other words:

$$\mathcal{M}\mathcal{X}_{\log k}(\underbrace{|i\rangle}_{\text{control input}} \underbrace{|\psi\rangle}_{\text{}}) = \underbrace{|i\rangle}_{\text{control output}} \underbrace{\mathcal{U}_i |\psi\rangle}_{\text{}}.$$

The matrix that represents the multiplexer is a block diagonal matrix of the corresponding operators:

$$\mathbf{M}(\mathcal{M}\mathcal{X}_{\log k}) = \mathbf{diag}(\mathbf{M}(\mathcal{U}_0), \dots, \mathbf{M}(\mathcal{U}_{k-1}))$$

Once we have obtained the Pauli coefficients of the matrix, we can utilize a multiplexer to construct a block-diagonal matrix composed of the corresponding  $q$ -wise Pauli matrices. By employing a state preparation circuit for the coefficients, we can efficiently implement linear combinations of coefficients multiplied by matrices, effectively creating a block encoding of the matrix  $\mathbf{A}$ .

To efficiently determine the Pauli coefficients classically, we require some theoretical groundwork.

## 1.4 Contribution

This work makes three key contributions: (a) the first bidirectional equivalence between block encoding and matrix state preparation input models, (b) a novel classical Pauli decomposition algorithm with  $O(n^2 \log n)$  run-time complexity, and (c) an efficient quantum circuit implementation for multiplexed Pauli tensor products with  $O(n^2)$  gate complexity. This general-purpose approach improves upon existing techniques for arbitrary matrix encoding, achieving particular efficiency when the Pauli decomposition is sparse.

## 2 Block encoding equivalence

Given a matrix state preparation circuit for  $\mathbf{A}$  and a state preparation circuit for a vector  $\mathbf{w}$  whose entries are the row norms of  $\mathbf{A}$ , it is possible to construct a block encoding of  $\mathbf{A}$  [6, Section I.D]. We are unaware of any efficient algorithm that given *only* a matrix state preparation circuit for  $\mathbf{A}$  constructs a block encoding of  $\mathbf{A}$ . In this section we will show a way to create block encoding from state preparation circuits and vice versa.

### 2.1 Block encoding → matrix state preparation circuit

Mor-Yosef et al. [15] show that given a circuit  $\mathcal{U}$  we can construct a matrix state preparation of  $\mathbf{M}(\mathcal{U})$ . Thus, given a block encoding of  $\mathbf{A}$  we can immediately construct a matrix state preparation circuit for a matrix that contains  $\mathbf{A}$ .

The following provides a proof for creating a state preparation circuit, including auxiliary (garbage) quantum states, from Block Encoding.

**Proposition 4.** Suppose that  $\mathcal{U} \in \mathbf{BE}_{\alpha,\theta}(\mathbf{A})$ . Applying the 'qml.matrix' results  $\mathcal{U}_{\mathbf{M}(\mathcal{U})}$  s.t  $\mathcal{U}_{\mathbf{M}(\mathcal{U})} \in \mathbf{MS}_{\alpha,\theta}(\mathbf{A})$

*Proof.* We have that,

$$\begin{aligned}\mathcal{U}_{\mathbf{M}(\mathcal{U})} &= [\mathbf{vec}(\mathbf{M}(\mathcal{U})) \ *] \\ &= \alpha^{-1} e^{-i\theta} \left[ \mathbf{vec} \left( \begin{bmatrix} \mathbf{A} & * \\ * & * \end{bmatrix} \right) \ * \right] \\ &= \alpha^{-1} e^{-i\theta} \left[ \begin{array}{cc} \mathbf{vec}(\mathbf{A}) & * \\ \psi & * \end{array} \right]\end{aligned}$$

□

## 2.2 Matrix state preparation circuit → block encoding

We introduce a method to create block encoding solely from a matrix state preparation circuit. Preliminary examples are provided to illustrate this approach, with the full methodology detailed in the paper. At a high level, we construct  $\mathcal{U}_{\mathbf{A}_p}$  from  $\mathcal{U}_{\mathbf{A}}$ , and then apply the technique from the previous section to block encode  $\mathbf{A}$ .

In high level, we construct  $\mathcal{U}_{\mathbf{A}_p}$  from  $\mathcal{U}_{\mathbf{A}}$ , then use the technique from the previous section to block encode  $\mathbf{A}$ . We will now demonstrate how to compute  $\mathcal{U}_{\mathbf{A}_p}$  in the following section.

### 2.2.1 Construct $\mathcal{U}_{\mathbf{A}_p}$ from $\mathcal{U}_{\mathbf{A}}$ ( Warm-up: $q = 1$ and Real $\mathbf{A}$ )

As a warm-up, let us first consider the case of  $q = 1$ , i.e. the  $\mathbf{A} \in \mathbb{R}^{n \times n}$  a 2 by 2 real and Hermitian matrices. To keep notation simple, we use the 1 base index, i.e.

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \quad a_{12} = a_{21}$$

Note that we can write  $\mathbf{A}$  as a linear combination of the following matrices,

$$\mathbf{E}_{11} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \mathbf{E}_{12} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \mathbf{E}_{21} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}, \mathbf{E}_{22} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

Indeed we have that

$$\mathbf{A} = \sum_{i,j} a_{ij} \mathbf{E}_{ij} = a_{11} \mathbf{E}_{11} + a_{12} \mathbf{E}_{12} + a_{21} \mathbf{E}_{21} + a_{22} \mathbf{E}_{22}$$

From linearity of  $(\cdot)_p$  we have that

$$\mathbf{A}_p = \left( \sum_{i,j} a_{ij} \mathbf{E}_{ij} \right)_p = \mathbf{A} = \sum_{i,j} a_{ij} (\mathbf{E}_{ij})_p$$

Note that we can create state preparation circuits  $\{\mathcal{U}_{(\mathbf{E}_{ij})_p}\}_{i,j}$ , using the standard state preparation operation ([13]). Now we can create the  $\mathcal{U}_{\mathbf{A}_p}$  as follow:

This observation can easily be used to implement, via qMSLA operations, an algorithm that takes  $\mathcal{U}_{\mathbf{A}}$  and outputs  $\mathcal{U}_{\mathbf{A}_p}$ .

## References

- [1] Andris Ambainis. Variable time amplitude amplification and quantum algorithms for linear algebra problems. In *STACS'12 (29th Symposium on Theoretical Aspects of Computer Science)*, volume 14, pages 636–647. LIPIcs, 2012.
- [2] Dong An and Lin Lin. Quantum linear system solver based on time-optimal adiabatic quantum computing and quantum approximate optimization algorithm. *ACM Transactions on Quantum Computing*, 3(2):1–28, 2022.
- [3] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. *Nature*, 549(7671):195–202, 2017.
- [4] Shantanav Chakraborty, András Gilyén, and Stacey Jeffery. The Power of Block-Encoded Matrix Powers: Improved Regression Techniques via Faster Hamiltonian Simulation. In Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi, editors, *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*, volume 132 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 33:1–33:14, Dagstuhl, Germany, 2019. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.
- [5] Andrew M Childs, Robin Kothari, and Rolando D Somma. Quantum algorithm for systems of linear equations with exponentially improved dependence on precision. *SIAM Journal on Computing*, 46(6):1920–1950, 2017.
- [6] B David Clader, Alexander M Dalzell, Nikitas Stamatopoulos, Grant Salton, Mario Berta, and William J Zeng. Quantum resources required to block-encode a matrix of classical data. *IEEE Transactions on Quantum Engineering*, 3:1–23, 2022.
- [7] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 193–204, 2019.
- [8] Aram W Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Physical Review Letters*, 103(15):150502, 2009.
- [9] Iordanis Kerenidis and Anupam Prakash. Quantum Recommendation Systems. In Christos H. Papadimitriou, editor, *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, volume 67 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 49:1–49:21, Dagstuhl, Germany, 2017. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.
- [10] Iordanis Kerenidis and Anupam Prakash. Quantum gradient descent for linear systems and least squares. *Physical Review A*, 101(2):022316, 2020.
- [11] Lin Lin and Yu Tong. Optimal polynomial based quantum eigenstate filtering with application to solving quantum linear systems. *Quantum*, 4:361, 2020.
- [12] John M Martyn, Zane M Rossi, Andrew K Tan, and Isaac L Chuang. Grand unification of quantum algorithms. *PRX Quantum*, 2(4):040203, 2021.
- [13] Vivek V Shende, Stephen S Bullock, and Igor L Markov. Synthesis of quantum logic circuits. In *Proceedings of the 2005 Asia and South Pacific Design Automation Conference*, pages 272–275, 2005.

- [14] Yigit Subasi, Rolando D Somma, and Davide Orsucci. Quantum algorithms for systems of linear equations inspired by adiabatic quantum computing. *Physical Review Letters*, 122(6):060504, 2019.
- [15] Liron Mor Yosef, Shashanka Ubaru, Lior Horesh, and Haim Avron. Multivariate trace estimation using quantum state space linear algebra. *arXiv preprint arXiv:2405.01098*, 2024.

# Adaptive data-driven reduced-order models of port-Hamiltonian dynamical systems for nonlinear inverse scattering applications

*Mikhail Zaslavskiy, Vladimir Druskin, Shari Moskow*

Abstract

## 1 Problem formulation

The inverse scattering problem formulated for the Schrödinger operators arises in various fields, including quantum mechanics, radars, viscoelasticity, Biot problems, remote sensing, geophysical, and medical imaging. The goal of imaging is to find medium properties in the domain using near-field measured data. The model based nonlinear optimization which is the method of choice for the solution of the inverse problems can be unreliable and particularly expensive for such problems. Data driven nonlinear transforms can be an opening, however it was recently shown that the ReLU networks are intractable for reliable solution of the inverse problems in continuum using conventional digital computers. In the present work, following the success of data-driven reduced-order models (ROMs) developed in recent years, we propose a robust direct method for solving inverse scattering problems for the Schrödinger equation. Our approach is based on a Lippmann-Schwinger algorithm with a crucial component composed of adaptive data-driven ROMs in the frequency domain and efficient learning the internal solutions. Below we discuss the details of the algorithms as well as some bottlenecks.

We consider first-order formulation of frequency-domain wave problem in lossy medium

$$\nabla \cdot v + \frac{1}{2} \nabla(\ln(\sigma)) \cdot v + ru + i\omega u = f \quad (1)$$

$$\nabla u - \frac{1}{2} \nabla(\ln(\sigma)) u + i\omega v = 0 \quad (2)$$

in a bounded domain  $\Omega$  for  $m$  sources  $f \in \mathbb{R}^{\infty \times m}$ . Here  $u, v \in \mathbb{C}^{\infty \times m}$  with columns being solutions for the corresponding sources. We assume that the measured data is given by  $f^T u$  where columns of  $f$  and  $u$  are multiplied with respect to  $L_2$  inner product  $(g; h)_{L_2} = \int_{\Omega} g h dV$ . After spatial discretization we obtain MIMO port-Hamiltonian LTI dynamical system [2] in the frequency-domain

$$(A + P)w + i\omega w = F \quad (3)$$

with skew-symmetric matrix  $A = -A^T \in \mathbb{R}^{N \times N}$ , symmetric matrix  $P = P^T \in \mathbb{R}^{N \times N}$ ,  $F \in \mathbb{R}^{N \times m}$ . The measured data is given by  $D(\omega) = F^T w$ . We note that the obtained system (3) is symmetric with respect to indefinite (pseudo-)inner product  $(x; y)_J = x^H J x$  where  $H$  denotes Hermitian conjugate and

$$J = \begin{pmatrix} I & 0 \\ 0 & -I \end{pmatrix} \quad (4)$$

In the inverse scattering problem in lossy medium the goal is to recover unknown damping term  $r$  and reflectivity  $\ln(\sigma)$  under known data  $D(\omega)$ . In the LTI system (3) the discrete counterparts of both unknowns compose matrix  $P$ .

Lippmann-Schwinger approach has been proven to be a powerful tool for solving inverse scattering problem [3]. It can be formulated in terms of integral equation involving solution  $(u; v)$  for unknown

medium as well as solution  $(u^0; v^0)$  for background with some a priori known parameters (say,  $r = \ln(\sigma) = 0$ ):

$$\nabla \cdot v^0 + i\omega u^0 = f \quad (5)$$

$$\nabla u^0 + i\omega v^0 = 0. \quad (6)$$

After spatial discretization we obtain

$$Aw^0 + i\omega w^0 = F \quad (7)$$

with background data given by  $D^0(\omega) = F^T w^0$ . The discrete counterpart of Lippmann-Schwinger equation can be formulated then as

$$(w^0)^H(\omega) J P w(\omega) = D^0(\omega) - D(\omega). \quad (8)$$

This is a system of nonlinear equations with respect to unknown parameter matrix  $P$  because  $w(\omega)$  itself depends on  $P$ . Traditional way to linearize this problem is so-called Born approximation, i.e.  $w(\omega) \approx w^0(\omega)$ , however it is known to work for small  $P$  only. Below we will show how to exploit the measured data  $D(\omega)$  to construct a better approximant of  $w(\omega)$  for further linearization of Lippmann-Schwinger equation (8).

## 2 Adaptive data-driven ROMs

Consider Galerkin projection of (3) onto the rational Krylov subspace spanned on columns of  $V = \{w_1 = w(\omega_1), \dots, w_n = w(\omega_n)\}$  with respect to  $(\cdot, \cdot)_J$  inner product. Here  $w(\omega_i)$ ,  $i = 1, \dots, n$  are solutions of (3) for  $\omega = \omega_1, \dots, \omega_n$ , respectively. The projected system (3) has a form

$$\mathcal{S}\mathcal{W} + i\omega\mathcal{M}\mathcal{W} = \mathcal{B} \quad (9)$$

where  $w \approx V\mathcal{W}$ ,  $\mathcal{S}$  is Hermitian indefinite stiffness matrix with block elements

$$\mathcal{S}_{pq} = w_q^H J(A + P)w_p \in \mathbb{C}^{m \times m}, \quad q, p = 1, \dots, n \quad (10)$$

,  $\mathcal{M}$  is Hermitian indefinite mass matrix with block elements

$$\mathcal{M}_{pq} = w_q^H J w_p \in \mathbb{C}^{m \times m}, \quad q, p = 1, \dots, n \quad (11)$$

and blocks of  $\mathcal{B}$  are given by

$$\mathcal{B}_q = w_q^H F = \bar{D}_q = \bar{D}(\omega_q) \in \mathbb{C}^{m \times m} \quad q = 1, \dots, n. \quad (12)$$

The measured data in Galerkin formulation is given by

$$\mathcal{F} = \mathcal{B}^H \mathcal{W} \quad (13)$$

We note that although internal solutions  $w_p$ ,  $p = 1, \dots, n$  are not accessible because  $P$  in (3) is unknown, blocks of mass and stiffness matrices can still be obtained directly from the data via Loewner framework:

$$w_q^H J w_p = \frac{D_p - \bar{D}_q}{i\omega_q + i\omega_p} \quad (14)$$

and

$$w_q^H J(A + P) w_p = \frac{\omega_q \bar{D}_q + D_p \omega_p}{\omega_q + \omega_p} \quad (15)$$

(with derivatives of the data  $D$  and  $\omega D$  for mass and stiffness matrices when  $\omega_p = -\omega_q$ , respectively).

To improve the efficiency of the constructed ROM we employ greedy algorithms for adaptive choice of interpolation frequencies  $\omega_1, \dots, \omega_n$  which is similar to AAA algorithm [4]:

- Algorithm 1**
1. For the given frequency range of interest  $\omega \in [\omega_{min}, \omega_{max}]$  set  $\omega_1 = \sqrt{\omega_{min}\omega_{max}}$ ,  $n = 1$
  2. Compute matrix pencil  $(\mathcal{S}; \mathcal{M})$  via Loewner approach (14) and (15) for interpolation points  $\omega_1, \dots, \omega_n$
  3. Compute ROM data  $\mathcal{F}$  for  $\omega \in [\omega_{min}, \omega_{max}]$  via (9) and (13)
  4. Evaluate error  $\mathcal{F} - D$  and set  $\omega_{n+1} = \text{argmax}(\|\mathcal{F} - D\|)$ . If there are several frequencies for which the maximum is attained, it suffices to select any one of the corresponding frequencies.
  5. Set  $n = n + 1$ .
  6. Repeat steps 2–5 until convergence to the desired accuracy.

We note that the obtained ROM is not structure-preserving, i.e. it may not inherit passivity and even stability of the original full-scale system (3).

### 3 Lippmann-Schwinger-Lanczos approach

In this section we show how to exploit the constructed data-driven ROMs to construct an approximant of internal solution  $w$  in (8). Similar to the unknown medium, we can construct background matrix pencil  $(\mathcal{S}^0; \mathcal{M}^0)$  for the selected set of frequencies  $\omega_1, \dots, \omega_n$ . Note that it can be performed in model-driven way because all the internal solutions for known background  $P = 0$  are accessible. Background counterpart of ROM (9) has a form

$$\mathcal{S}^0 \mathcal{W}^0 + i\omega \mathcal{M}^0 \mathcal{W}^0 = \mathcal{B}^0 \quad (16)$$

where  $w^0 \approx V^0 \mathcal{W}^0$  and  $V^0 = \{w_1^0 = w^0(\omega_1), \dots, w_n^0 = w^0(\omega_n)\}$ . Let's perform Lanczos orthogonalization for matrix  $(\mathcal{M})^{-1} \mathcal{S}$  with respect to indefinite inner product  $(\cdot, \cdot)_\mathcal{M}$  and starting vector  $(\mathcal{M})^{-1} \mathcal{B} / \|(\mathcal{M})^{-1} \mathcal{B}\|_\mathcal{M}$  and do the same for background part  $(\mathcal{M}^0)^{-1} \mathcal{S}^0$  with respect to  $(\cdot, \cdot)_{\mathcal{M}^0}$  and starting vector  $(\mathcal{M}^0)^{-1} \mathcal{B}^0 / \|(\mathcal{M}^0)^{-1} \mathcal{B}^0\|$ :

$$(\mathcal{M})^{-1} \mathcal{S} Q = QT, \quad Q^H \mathcal{M} Q = I \quad (17)$$

$$(\mathcal{M}^0)^{-1} \mathcal{S}^0 Q^0 = Q^0 T^0, \quad (Q^0)^H \mathcal{M}^0 Q^0 = I. \quad (18)$$

As has been noted in [1], although  $V$  is totally different from  $V_0$ , we have  $VQ \approx V_0 Q_0$ . It has been explained in that paper for lossless case via drawing an analogy with causal time-domain solutions, however similar reasoning is applicable for lossy scenario and ROM we developed. Therefore, we can construct an approximant of internal solution as

$$w \approx V(\mathcal{S} + i\omega \mathcal{M})^{-1} \mathcal{B} = VQ(T + i\omega I)^{-1} E_1 |(\mathcal{M})^{-1} \mathcal{B}|_\mathcal{M} \approx V^0 Q^0 (T + i\omega I)^{-1} E_1 |(\mathcal{M})^{-1} \mathcal{B}|_\mathcal{M} = \mathbf{w} \quad (19)$$

Once plugged in into the Lippmann–Schwinger equation (8), we obtain a linear equation with respect to  $P$ :

$$(w^0)^H(\omega)JP\mathbf{w}(\omega) = D^0(\omega) - D(\omega). \quad (20)$$

We call this algorithm Lippmann–Schwinger–Lanczos to emphasize the crucial component of constructing internal solution that is based Lanczos orthogonalization. There are multiple parts of our approach that need to be addressed to improve its performance:

- Efficient handling of overfitting that results in rank-deficient matrix  $\mathcal{M}$  and may breakdown Lanczos algorithm
- Fast and robust solution of (20) that is typically underdetermined
- Data completion approach to handle missing data in square MIMO transfer function  $D(\omega)$
- Construction of passive ROMs
- Convergence estimates

## References

- [1] Justin Baker and Elena Cherkaev and Vladimir Druskin and Shari Moskow and Mikhail Zaslavsky, *Regularized Reduced Order Lippman-Schwinger-Lanczos Method for Inverse Scattering Problems in the Frequency Domain*, submitted to Journal of Computational Physics, arXiv preprint arXiv:2311.16367, 2023
- [2] Christopher A. Beattie, Volker Mehrmann, Paul Van Dooren, *Robust port-Hamiltonian representations of passive systems*, Automatica, 100, 2019: 182–186
- [3] D. Colton and R. Kress, *Inverse Acoustic and Electromagnetic Scattering Theory*, Springer, Berlin, 2nd ed., 1998.
- [4] Nakatsukasa, Yuji and Sète, Olivier and Trefethen, Lloyd N, *The AAA Algorithm for Rational Approximation*, SIAM Journal on Scientific Computing, 40 (3), 2018: A1494-A1522

# Iterative Methods for Sylvester-like Variational Inequality Problems

Ning Zheng

## Abstract

We consider the solution of Sylvester-like variational inequality problem (SVIP), or linear matrix equation complementarity problem (LMECP),

$$X \geq \mathbf{0}, \quad W = AX + XB - F \geq \mathbf{0} \quad \text{and} \quad \langle X, W \rangle = \mathbf{0} \quad (1)$$

where  $A \in \mathbf{R}^{m \times m}$ ,  $B \in \mathbf{R}^{n \times n}$  and  $F \in \mathbf{R}^{m \times n}$  are large, sparse and structured discretization matrices from partial differential operators, and  $X \in \mathbf{R}^{m \times n}$  is an unknown matrix. Here,  $\langle X, W \rangle = \text{Tr}(X^\top W)$  denotes the Frobenius inner product of two matrices, where  $X^\top$  denotes the transpose of the matrix  $X$ . If  $B = A^\top$  and  $F$  is symmetric, we refer (1) as the Lyapunov-like linear matrix equation complementarity problem. LMECP (1) generally arises from the finite discretization of free boundary problems

$$v(x) \geq g(x), \quad w(x) = \mathcal{L}v(x) - f(x) \geq 0 \quad \text{and} \quad (v(x) - g(x))w(x) = 0, \quad (2)$$

where  $\mathcal{L}$  is a given partial differential operator, and  $x \in D \subseteq \mathbf{R}^n$  where  $D$  is a given domain with boundary  $\partial D$ . The boundary condition of (2) is  $v(x) = g(x)$ ,  $x \in \partial D$ . Well known examples of free boundary problems which can be written in the form (2) include American option pricing, porous flow through dams, journal bearing lubrication, and elastic-plastic torsion, etc.

The vectorization of the LMECP (1) gives a mathematically equivalent linear complementarity problem (LCP),

$$\mathbf{x} \geq \mathbf{0}, \quad \mathcal{A}\mathbf{x} - \mathbf{f} \geq \mathbf{0} \quad \text{and} \quad \mathbf{x}^\top(\mathcal{A}\mathbf{x} - \mathbf{f}) = 0, \quad (3)$$

where  $\mathcal{A} = I_n \otimes A + B^\top \otimes I_m \in \mathbf{R}^{mn \times mn}$ ,  $\mathbf{f} = \text{vec}(F) \in \mathbf{R}^{mn \times 1}$  and  $\mathbf{x} = \text{vec}(X) \in \mathbf{R}^{mn \times 1}$ . Here, the symbol  $\otimes$  denotes the Kronecker product, and  $\text{vec}(\cdot)$  denotes the vectorization operator that converts a matrix into a vector by stacking the columns of the matrix on top of one another. There are few numerical methods specifically designed for solving LMECP (1). Numerical methods [6, 4, 2, 3, 7] for LCP (3) are generally not efficient for directly solving LMECP (1) due to the large storage and complexity.

When the matrix  $\mathcal{A}$  arises from the finite difference discretization of elliptic and parabolic partial differential equations, it has structure that contains discretization components from different spatial derivatives. Hence, the idea of alternating direction implicit (ADI) method is to split the finite difference operator into separate operators, where each operator corresponds to the discretization of one-dimensional spatial derivative term, so that the solution of discretized system can be transformed to the alternative solutions of discretized sub-systems, which have a simpler structure that requires fewer storage and computational costs. Let  $\mathcal{A} = \mathcal{H} + \mathcal{V}$  be the matrix splitting of  $\mathcal{A}$ , where  $\mathcal{H} = I_n \otimes A$  and  $\mathcal{V} = B^\top \otimes I_m$  are respectively generated from discrete central difference approximations to the particular one-dimensional equation. The Peaceman-Rachford ADI for LCP (3) proposed by Lin and Cryer [5], and the matricization of ADI is described as follows.

---

**Algorithm 1** Peaceman-Rachford ADI for LMECP (1)

---

1: Input an initial guess  $X^{(0)}$  and positive parameters  $r_k$

2: **for**  $k = 0, 1, 2, \dots$  until convergence **do**

3:   Compute  $X^{(k+\frac{1}{2})}$  by solving the LCP subproblem

$$\begin{cases} W^{(k+\frac{1}{2})} = (A + r_k I_m)X^{(k+\frac{1}{2})} + X^{(k)}(B - r_k I_n) - F \geq \mathbf{0}, \\ X^{(k+\frac{1}{2})} \geq \mathbf{0}, \quad \langle X^{(k+\frac{1}{2})}, W^{(k+\frac{1}{2})} \rangle = 0 \end{cases} \quad (4)$$

4:   Compute  $X^{(k+1)}$  by solving the LCP subproblem

$$\begin{cases} W^{(k+1)} = X^{(k+1)}(B + r_k I_n) + (A - r_k I_m)X^{(k+\frac{1}{2})} - F \geq \mathbf{0}, \\ X^{(k+1)} \geq \mathbf{0}, \quad \langle W^{(k+1)}, X^{(k+1)} \rangle = 0. \end{cases} \quad (5)$$

5: **end for**

---

First, we propose a projection method for solving LMECP (1) by transforming the matrix equation into LCP (3) with a vector form by means of the Kronecker product. We can reformulate LCP (3) to an equivalent fixed-point equation

$$\mathbf{x} = \mathbf{Proj}(\mathbf{x} - \alpha[\mathcal{A}\mathbf{x} - \mathbf{f}]),$$

where  $\alpha > 0$  is a scalar and  $\mathbf{Proj}(\cdot) = \max(\cdot, 0)$  denotes the orthogonal projection of vector or matrix onto nonnegative cone. The matricization form gives

$$X = \mathbf{Proj}(X - \alpha(AX + XB - F)).$$

The gradient projection method for LMECP (1) is listed in Algorithm 2.

---

**Algorithm 2** Projection method for LMECP (1)

---

1: Input an initial guess  $X^{(0)}$  and positive parameter  $\alpha$

2: **for**  $k = 0, 1, 2, \dots$  until convergence **do**

3:   Compute the residual  $R^{(k)} = F - AX^{(k)} - X^{(k)}B$

4:   Compute

$$X^{(k+1)} = \mathbf{Proj}(X^{(k)} + \alpha R^{(k)}). \quad (6)$$

5: **end for**

---

Next, we discuss the convergence of Peaceman-Rachford ADI algorithm Algorithm 1 for the non-Hermitian case. Unlike the symmetric case [1, 5], convergence properties for nonsymmetric situations cannot be established relying on the descent function of the quadratic form. Rather, as with most iterative methods for solving systems of equations, the recursive relation between two successive iterations will be utilized here. We first equivalently reformulate the LCP (1) as an implicit fixed-point equation by variable transformation, and thus the ADI Algorithm 1 can be correspondingly reformulated. Then the recursive error relations are constructed based on the fixed-point equations. In addition, we consider the case when  $\mathcal{H}$  and  $\mathcal{V}$  are  $H_+$ -matrices. We study the convergence analysis of ADI algorithm for LMECP when  $\mathcal{H}\mathcal{V} = \mathcal{V}\mathcal{H}$  does not necessarily hold.

Denote

$$\begin{aligned} L_\alpha(\mathcal{H}) &= |(\alpha I + \mathcal{H} + r_k I)^{-1}(\alpha I - \mathcal{H} - r_k I)|, \\ L_\beta(\mathcal{V}) &= |(\beta I + \mathcal{V} + r_k I)^{-1}(\beta I - \mathcal{V} - r_k I)|, \\ K_\alpha(\mathcal{H}, \mathcal{V}) &= 2|(\alpha I + \mathcal{H} + r_k I)^{-1}(\mathcal{V} - r_k I)| \\ K_\beta(\mathcal{V}, \mathcal{H}) &= 2|(\beta I + \mathcal{V} + r_k I)^{-1}(\mathcal{H} - r_k I)| \end{aligned}$$

We have the following convergence theorem.

**Theorem 1** *Peaceman-Rachford ADI algorithm converges to the unique solution for any initial vector if  $\rho(L_\alpha(\mathcal{H})) < 1$ ,  $\rho(L_\beta(\mathcal{V})) < 1$  and  $\rho(G) < 1$ , where  $\rho(\cdot)$  denotes for the spectral radius of the matrix and*

$$G = [I - L_\beta(\mathcal{V})]^{-1} K_\beta(\mathcal{V}, \mathcal{H}) [I - L_\alpha(\mathcal{H})]^{-1} K_\alpha(\mathcal{H}, \mathcal{V}).$$

Consider the case when both  $\mathcal{H}$  and  $\mathcal{V}$  are  $H_+$ -matrices. Let  $\mathcal{H} = D_{\mathcal{H}} + B_{\mathcal{H}}$  and  $\mathcal{V} = D_{\mathcal{V}} + B_{\mathcal{V}}$ , where  $D_{\mathcal{H}}$  and  $B_{\mathcal{H}}$  are the diagonal and off-diagonal parts of  $\mathcal{H}$ , respectively, and  $D_{\mathcal{V}}$  and  $B_{\mathcal{V}}$  are the diagonal and off-diagonal parts of  $\mathcal{V}$ , respectively.

**Theorem 2** *Peaceman-Rachford ADI algorithm converges to the unique solution for any initial vector, provided that  $\mathcal{H}$  and  $\mathcal{V}$  are  $H_+$ -matrices and*

$$\begin{aligned} (\alpha - r_k)I - D_{\mathcal{H}} &\geq \mathbf{0} \quad \text{and} \quad D_{\mathcal{V}} - r_k I \leq \mathbf{0}, \\ (\beta - r_k)I - D_{\mathcal{V}} &\geq \mathbf{0} \quad \text{and} \quad D_{\mathcal{H}} - r_k I \leq \mathbf{0}, \end{aligned}$$

In the following analysis, we assume that  $\mathcal{H}$  and  $\mathcal{V}$  are commute, that is to say,  $\mathcal{H}\mathcal{V} = \mathcal{V}\mathcal{H}$ . Remark that for  $\mathcal{H}$  and  $\mathcal{V}$  arising from the finite difference discretization of a separable second-order elliptic operator in a rectangular region, it can be shown that  $\mathcal{H}\mathcal{V} = \mathcal{V}\mathcal{H}$  holds.

Instead of taking the absolute value, we give another general convergence result based on the matrix norm as follows. Denote

$$\begin{aligned} \delta_\alpha(\mathcal{H}) &= \|(\alpha I + \mathcal{H} + r_k I)^{-1}(\alpha I - \mathcal{H} - r_k I)\|, \\ \delta_\beta(\mathcal{V}) &= \|(\beta I + \mathcal{V} + r_k I)^{-1}(\beta I - \mathcal{V} - r_k I)\|, \\ \tau_\alpha(\mathcal{H}) &= 2\|(\alpha I + \mathcal{H} + r_k I)^{-1}(\mathcal{V} - r_k I)\|, \\ \tau_\beta(\mathcal{V}) &= 2\|(\beta I + \mathcal{V} + r_k I)^{-1}(\mathcal{H} - r_k I)\|, \end{aligned}$$

where  $\|\cdot\|$  denotes for matrix norm.

**Theorem 3** *Peaceman-Rachford ADI algorithm converges to the unique solution for any initial vector if*

$$\delta_\alpha(\mathcal{H}) < 1, \quad \delta_\beta(\mathcal{V}) < 1 \quad \text{and} \quad \frac{\tau_\alpha(\mathcal{H})\tau_\beta(\mathcal{V})}{[1 - \delta_\alpha(\mathcal{H})][1 - \delta_\beta(\mathcal{V})]} < 1. \quad (7)$$

Consider the case when both  $\mathcal{H}$  and  $\mathcal{V}$  are Hermitian positive definite matrices, and thus  $\mathcal{A} = \mathcal{H} + \mathcal{V}$  is Hermitian positive definite.

**Theorem 4** Suppose that  $\mathcal{H}$  and  $\mathcal{V}$  are Hermitian positive definite matrices, and  $\mathcal{H}$  and  $\mathcal{V}$  are commute. If

$$r_k \geq \frac{1}{2} \max(\lambda_1 + \lambda_n, \sigma_1 + \sigma_n),$$

then Peaceman-Rachford ADI algorithm for LCP converges to the unique solution for any initial vector.

Finally, we present numerical experiments to show the convergence of proposed methods. We consider the free boundary problem arises from fractional Black-Scholes American option pricing. Assume that the asset prices  $S_1$  and  $S_2$  satisfy independent Lévy stochastic processes

$$\mathcal{L}u = -\frac{\partial u}{\partial t} + a_1 \frac{\partial u}{\partial x} + a_2 \frac{\partial u}{\partial y} - b_1 \left[ {}_{-\infty} D_x^\alpha u \right] - b_2 \left[ {}_{-\infty} D_y^\beta u \right] + ru$$

where  $x = \ln S_1$  and  $y = \ln S_2$ . Here,  ${}_{-\infty} D_x^\alpha u$  and  ${}_{-\infty} D_y^\beta u$  represents Caputo derivatives of  $u$  on  $x$  and  $y$ , and  $\alpha, \beta \in (1, 2)$ .

$$\begin{aligned} a_1 &= -r - \frac{1}{2} \sigma_1^\alpha \sec\left(\frac{\alpha\pi}{2}\right), & b_1 &= -\frac{1}{2} \sigma_1^\alpha \sec\left(\frac{\alpha\pi}{2}\right) \\ a_2 &= -r - \frac{1}{2} \sigma_2^\beta \sec\left(\frac{\beta\pi}{2}\right), & b_2 &= -\frac{1}{2} \sigma_2^\beta \sec\left(\frac{\beta\pi}{2}\right) \end{aligned}$$

$\sigma_1, \sigma_2$  are volatilities of asset prices. By finite difference discretization of the model, we apply the projection method and ADI algorithm to solve the resulting LMECP, and the numerical results further confirm our convergence analysis.

## References

- [1] B. H. AHN, *Solution of nonsymmetric linear complementarity problems by iterative methods*, Journal of Optimization Theory and Applications, 33 (1981), pp. 175–185.
- [2] Z.-Z. BAI, *Modulus-based matrix splitting iteration methods for linear complementarity problems*, Numerical Linear Algebra with Applications, 6 (2010), pp. 917–933.
- [3] A. HADJIDIMOS AND M. TZOUMAS, *Nonstationary extrapolated modulus algorithms for the solution of the linear complementarity problem*, Linear Algebra and Its Applications, 429 (2009), pp. 197–210.
- [4] N. W. KAPPLE AND L. T. WATSON, *Iterative algorithms for the linear complementarity problem*, International Journal of Computer Mathematics, 19 (1986), pp. 273–297.
- [5] Y. LIN AND C. W. CRYER, *An alternating direction implicit algorithm for the solution of linear complementarity problems arising from free boundary problems*, Applied Mathematics and Optimization, 13 (1985), pp. 1–17.
- [6] K. G. MURTY AND F.-T. YU, *Linear complementarity, linear and nonlinear programming*, volume 3, Citeseer, 1988.
- [7] N. ZHENG, K. HAYAMI AND J.-F. YIN, *Modulus-type inner outer iteration methods for nonnegative constrained least squares problems*, SIAM Journal on Matrix Analysis and Applications, 37 (2016), pp. 1250–1278.

# Monotonicity, Bounds, and Averaging of Block-Gauss and Gauss-Radau Quadrature for Computing $B^T \phi(A) B$

*Jörn Zimmerling, Vladimir Druskin, Valeria Simoncini*

## Abstract

We explore quadratures for  $\mathcal{F}(s) = B^T \phi(A, s) B$  where  $A$  is a symmetric, nonnegative-definite matrix in  $\mathbb{R}^{n \times n}$ ,  $B$  is a tall matrix in  $\mathbb{R}^{n \times p}$ , and  $\phi(\cdot, s)$  is a matrix function with parameter  $s$  [1, 2]. These formulations commonly arise in the computation of multiple-input, multiple-output transfer functions for diffusion PDEs.

We derive bounds and averaging schemes for quadrature rules for  $B^T \phi(A, s) B$  computed via the block-Lanczos algorithm, which are particularly efficient for discretizations of PDE operators with continuous spectra. Additionally, we demonstrate that these bounds and averaging schemes are applicable to parametric model reduction of dynamical systems via Galerkin projections.

## 1 Block-Lanczos Approximations to $B^T \phi(A, s) B$

We propose an approximation scheme for  $\mathcal{F}(s) = B^T \phi(A, s) B$  leveraging the block-Lanczos algorithm [3] and its representation via Stieltjes matrix continued fractions.

The block-Lanczos recursion for the block-Lanczos vectors  $Q_i \in \mathbb{R}^{n \times p}$  reads

$$AQ_i = Q_{i+1}\beta_{i+1} + Q_i\alpha_i + Q_{i-1}(\beta_i)^T, \quad (1)$$

with block coefficients  $\alpha_i, \beta_i \in \mathbb{R}^{p \times p}$ . Using Stieltjes matrix continued fractions, we show that this block-Lanczos algorithm defines a block-Gauss quadrature approximation  $\mathcal{F}_m(s)$  and converges monotonically for  $\phi(A, s) = (A+sI)^{-1}$  with  $s \in \mathbb{R}^+$  and  $I$  the identity matrix. We further show that a monotonically convergent block Gauss-Radau quadrature  $\tilde{\mathcal{F}}_m(s)$  can be readily defined through this Stieltjes continued fraction representation. In the literature, Gauss-Radau quadratures for symmetric matrices are often defined via rank-one updates of the Lanczos matrix in the non-block case [4, 5] or rank  $p$  updates in the block case [6].

Here we define Gauss-Radau quadrature through Stieltjes matrix continued fractions which can be written via the recursion

$$\mathcal{C}_j(s) = \frac{1}{s\hat{\gamma}_j + \frac{1}{\gamma_j + \mathcal{C}_{j+1}(s)}},$$

where  $\hat{\gamma}_j, \gamma_j \in \mathbb{R}^{p \times p}$  are symmetric positive definite matrices directly related to the block-Lanczos coefficients  $\alpha_j$  and  $\beta_j$ . We show that the Gauss quadrature approximation to  $B^T(A+sI)^{-1}B$  after  $m$  iterations of block-Lanczos corresponds to  $\mathcal{C}_1(s)$ , defined through the above recursion, terminated with  $\mathcal{C}_{m+1} = 0$ , whereas the Gauss-Radau quadrature corresponds to truncation with  $\mathcal{C}_{m+1} = \infty$ .

Through Stieltjes matrix continued fractions, we demonstrate that Gauss quadrature provides a lower bound, while Gauss-Radau quadrature provides an upper bound to the matrix function. Combined with the monotonicity result, this yields an ordering of the Gauss and Gauss-Radau quadrature approximations to  $\mathcal{F}$ . Given  $m$ , the quadrature order, we obtain

$$0 < \mathcal{F}_{m-1}(s) < \mathcal{F}_m(s) < \mathcal{F}(s) < \tilde{\mathcal{F}}_m(s) < \tilde{\mathcal{F}}_{m-1}(s) \quad \forall s \in \mathbb{R}_+,$$

where, for two symmetric matrices  $G_1, G_2$ , the notation  $G_1 < G_2$  indicates that  $G_2 - G_1$  is positive definite.

This ordering further enables derivation of easily computable error bounds of the form

$$\|\mathcal{F} - \mathcal{F}_m\| < \|\tilde{\mathcal{F}}_m - \mathcal{F}_m\|.$$

In this contribution, we discuss averaging schemes of Gauss and Gauss-Radau quadrature motivated by potential theory for Padé approximations. We show numerical examples for various  $\phi(A, s)$ , where  $A$  is a graph Laplacian or discretization of an operator with a continuous spectrum (e.g., PDE operators in unbounded domains). These examples demonstrate that the derived error bound is tight in important applications, and that the averaging schemes reduce the approximation error by an order of magnitude for discretizations of operators with continuous spectra, as shown in [7]. In the next section we discuss the applicability of such Gauss-Radau bounds to parametric model reduction, a subsequent result not covered in [7].

## 2 Parametric Model Reduction via Galerkin Projection

The first iteration of the block-Lanczos procedure can be interpreted as a Galerkin projection of a symmetric positive definite matrix on a general basis. This insight allows us to directly apply the Gauss-Radau bound and averaging schemes to, for instance, projection-based parametric model reduction [8].

Consider the parametric dynamical system

$$(A(\rho) + sI)X(s, \rho) = B, \quad \text{with transfer function } \mathcal{F}(s) = B^T X(s, \rho),$$

where  $A(\rho) \in \mathbb{R}^{n \times n}$  is symmetric positive definite for all parameters  $\rho$  of interest,  $B \in \mathbb{R}^{n \times p}$ , and the Laplace frequency  $s \in \mathbb{R}^+$ . This formulation arises in inverse problems where  $A(\rho)$  represents a discretized PDE operator with PDE coefficients parametrized by  $\rho$ .

Assuming for simplicity, that the Galerkin projection basis  $U \in \mathbb{R}^{n \times k}$  contains  $B$  as  $U = [B, U_0]$  and is orthogonalized  $U^T U = I_k$  (e.g. a rational Krylov subspace with a shift at  $\infty$ ). Then the Galerkin approximation of  $\mathcal{F}$  is

$$\begin{aligned} \mathcal{F}_{\text{Gal}}(s) &= (U^T B)^T (U^T A(\rho) U + sI_k)^{-1} (U^T B) \\ &= \begin{bmatrix} I_p \\ 0 \end{bmatrix}^T (H^{\text{ROM}}(\rho) + sI_k)^{-1} \begin{bmatrix} I_p \\ 0 \end{bmatrix}. \end{aligned}$$

We can interpret  $U$  as the first Lanczos vector  $Q_1$  in equation (1) for  $i = 1$  with  $Q_0 = 0$  and  $H^{\text{ROM}}(\rho)$  as the  $\boldsymbol{\alpha}_1$  block-Lanczos coefficents. Then we can further define the  $\boldsymbol{\beta}_2$  coefficent according to the block-Lanczos recursion

$$(\boldsymbol{\beta}_2)^T \boldsymbol{\beta}_2 = U^T A^2 U - (H^{\text{ROM}}(\rho))^2.$$

Then the Gauss-Radau quadrature approximation to the  $k \times k$  transfer function  $\mathcal{F}_U = U^T (A(\rho) + sI)U$  as defined in [7] reads

$$\tilde{\mathcal{F}}_U(s) = \begin{bmatrix} I_k \\ 0 \end{bmatrix}^T \left( \begin{bmatrix} H^{\text{ROM}}(\rho) & \boldsymbol{\beta}_2^T \\ \boldsymbol{\beta}_2 & \boldsymbol{\beta}_2 (H^{\text{ROM}}(\rho))^{-1} \boldsymbol{\beta}_2^T \end{bmatrix} + sI_{2k} \right)^{-1} \begin{bmatrix} I_k \\ 0 \end{bmatrix}.$$

Since the Gauss Radau approximation  $\tilde{\mathcal{F}}_U$  is an upper bound to  $\mathcal{F}_U$  their difference  $\tilde{\mathcal{F}}_U - \mathcal{F}_U$  is s.p.d. and any subspace projection of a s.p.d. matrix is s.p.d. Hence the leading  $p \times p$  block of  $\tilde{\mathcal{F}}_U$  provides an easy-to-compute error bound for the leading block of  $\mathcal{F}_U$  which coincides with  $\mathcal{F}_{\text{Gal}}$  and holds for any  $U$  and  $\rho$ . In this contribution, we will show that such a bound provides a meaningful tool for applications in inverse problems and greedy selection of interpolation points in parametric model reduction.

## References

- [1] Gene H. Golub and Gérard Meurant. *Matrices, Moments and Quadrature with Applications*. Princeton University Press, 2010.
- [2] C. Fenu, D. Martin, L. Reichel, and G. Rodriguez. Block Gauss and anti-Gauss quadrature with application to networks. *SIAM Journal on Matrix Analysis and Applications*, 34(4):1655–1684, 2013.
- [3] G.H. Golub and R. Underwood. The block Lanczos method for computing eigenvalues. In John R. Rice, editor, *Mathematical Software*, pages 361–377. Academic Press, 1977.
- [4] G. López Lagomasino, L. Reichel, and L. Wunderlich. Matrices, moments, and rational quadrature. *Linear Algebra and its Applications*, 429(10):2540–2554, 2008. Special Issue in honor of Richard S. Varga.
- [5] Andreas Frommer, Kathryn Lund, Marcel Schweitzer, and Daniel B. Szyld. The Radau–Lanczos method for matrix functions. *SIAM Journal on Matrix Analysis and Applications*, 38(3):710–732, 2017.
- [6] Kathryn Lund. *A New Block Krylov Subspace Framework with Applications to Functions of Matrices Acting on Multiple Vectors*. Phd thesis, Department of Mathematics, Temple University and Fakultät Mathematik und Naturwissenschaften der Bergischen Universität Wuppertal, Philadelphia, Pennsylvania, USA, May 2018.
- [7] Jörn Zimmerling, Vladimir Druskin and Valeria Simoncini. Monotonicity, bounds and extrapolation of Block-Gauss and Gauss-Radau quadrature for computing  $B^T \phi(A)B$ , 2024; arXiv:2407.21505.
- [8] Benner, P., Gugercin, S. & Willcox, K. A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems. *SIAM Review*. 57, 483–531 (2015),