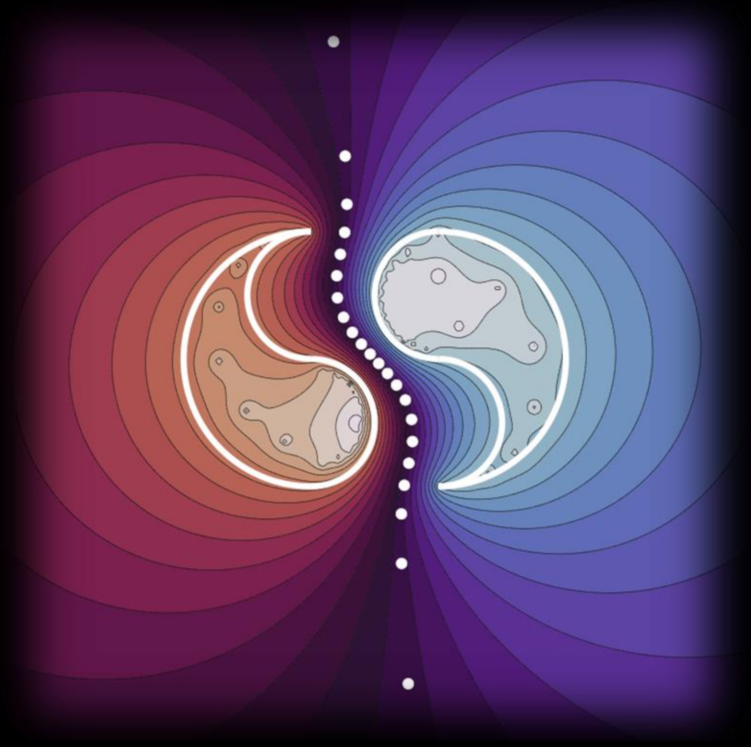
Cauchy matrices explain a lot

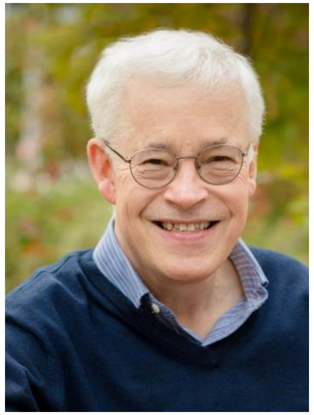


Heather Wilber
18 March 2026
Berkeley Applied Math Seminar

This work is supported via NSF grant DMS-2410045

My collaborators

Raaga Vangala
Univ. of Washington



Nick Trefethen
Harvard University

Marc Aurelle-Gills
Princeton University

Ethan Epperly
UC Berkeley

Alex Barnett
Flatiron Institute

Bernhard Beckermann
Univ. de Lille

Daniel Kressner
EPFL

What is a Cauchy matrix?

Cauchy matrix:

$$\hat{C} = \begin{bmatrix} \frac{1}{x_1 - y_1} & \cdots & \frac{1}{x_1 - y_n} \\ \frac{1}{x_2 - y_1} & \cdots & \frac{1}{x_2 - y_n} \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \frac{1}{x_m - y_1} & \cdots & \frac{1}{x_m - y_n} \end{bmatrix}$$

Left set

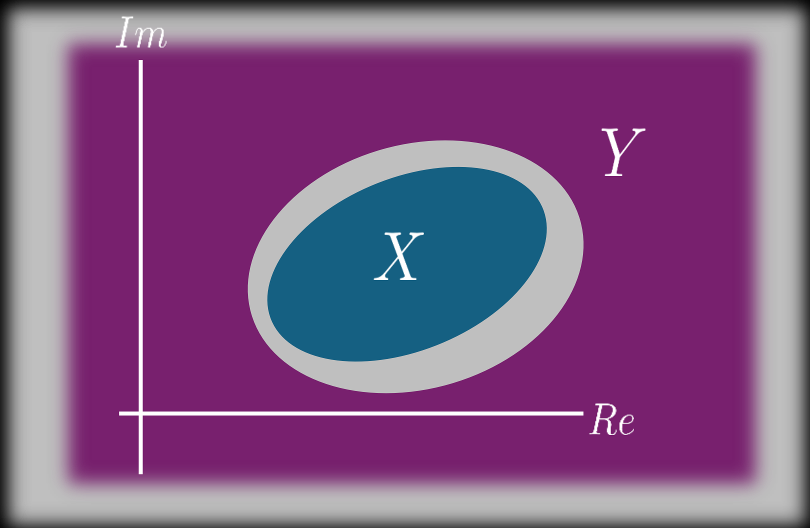
$\{x_1, \cdots, x_m\}$

Right set

$\{y_1, \cdots, y_m\}$

Cauchy function:

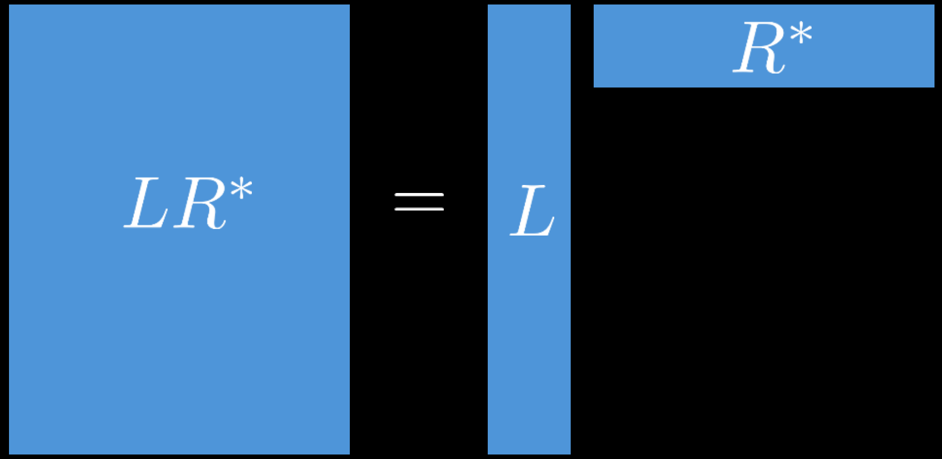
$$\mathcal{C}(x, y) = \frac{1}{x - y}$$



What is a Cauchy-like matrix?

Cauchy-like matrix:

$$C = \begin{bmatrix} \frac{(LR^*)_{11}}{x_1 - y_1} & \cdots & \frac{(LR^*)_{1n}}{x_1 - y_n} \\ \frac{(LR^*)_{21}}{x_2 - y_1} & \cdots & \frac{(LR^*)_{2n}}{x_2 - y_n} \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \frac{(LR^*)_{m1}}{x_m - y_1} & \cdots & \frac{(LR^*)_{mn}}{x_m - y_n} \end{bmatrix}$$



Left set

$\{x_1, \cdots, x_m\}$

Right set

$\{y_1, \cdots, y_m\}$

What is a Cauchy-like matrix?

🔑 A Cauchy-like matrix is the Hadamard product of a Cauchy matrix and a low rank matrix.

$$C = \hat{C} \circ (LR^*)$$

Famous Cauchy-like matrices:

- Loewner matrices
- Pick matrices
- The Hilbert matrix

``A fast transform away" from a Cauchy-like matrix:

- Toeplitz matrices
- Hankel matrices
- Vandermonde matrices

Why are they so commonly encountered?

Why are they so commonly encountered?

Cauchy's integral equation

- Quadrature design
- Resolvent-based methods
- Functions of matrices/operators

Rational approximation

- Function approximation
- Method of fund. sol (PDEs)
- Various iterative methods in NLA
- Digital filter design

Displacement structure

- Low rank approximation
- Fast direct solvers for structured linear systems
- Iterative methods for linear matrix equations

Cauchy matrices in action!

I. Application with Cauchy's integral eqn. + rational approx.:

Proxy point methods via latent Cauchy-like decompositions

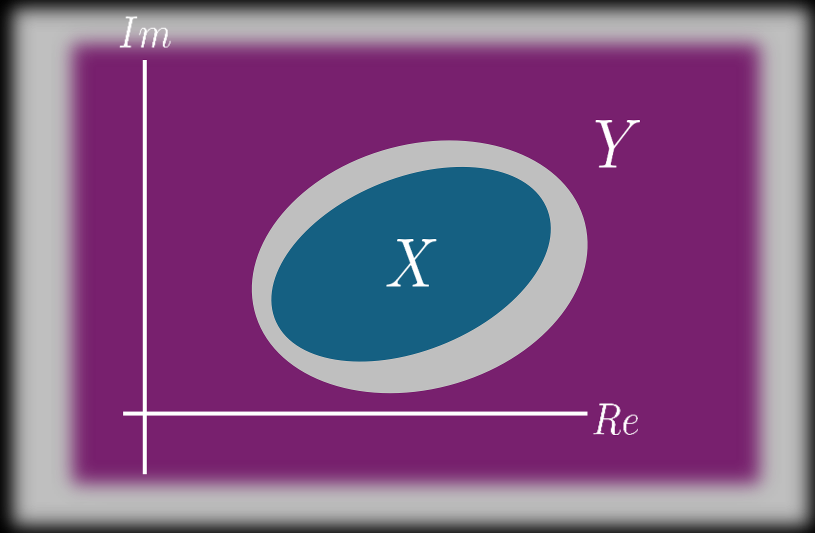
II. Exploiting displacement structure:

Extracting the Cauchy-like ``core" of displacement structure

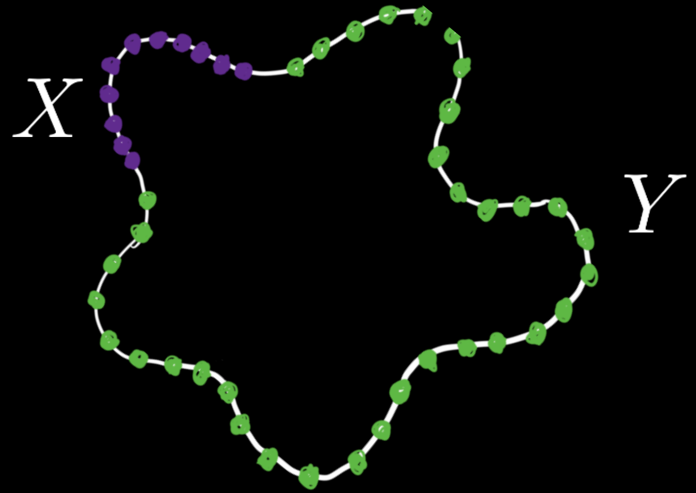
I. Proxy points via latent Cauchy-like decompositions

Interaction matrices

$$A = \begin{bmatrix} K(x_1, y_1) & \cdots & K(x_1, y_n) \\ K(x_2, y_1) & \cdots & K(x_2, y_n) \\ \vdots & & \vdots \\ K(x_m, y_1) & \cdots & K(x_m, y_n) \end{bmatrix}$$

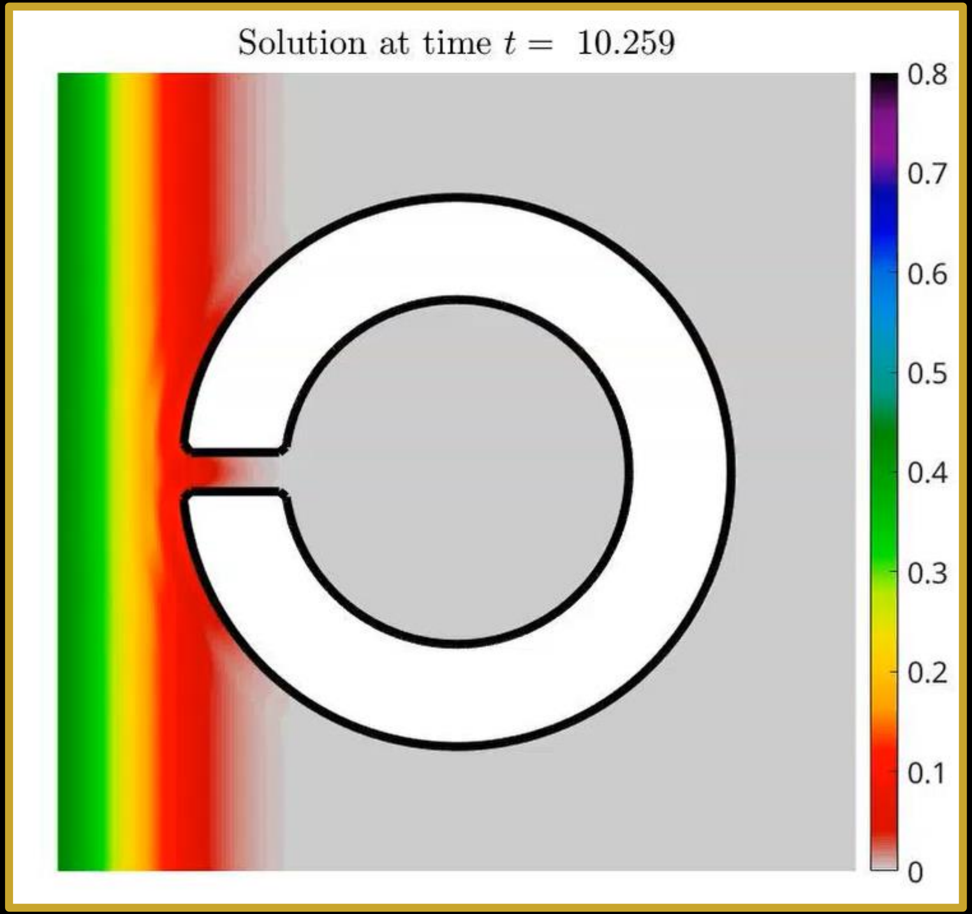


- K encodes *interactions* between points x_j and y_k
- One dimension of A is much *larger* than the other



Motivation: boundary integral equations

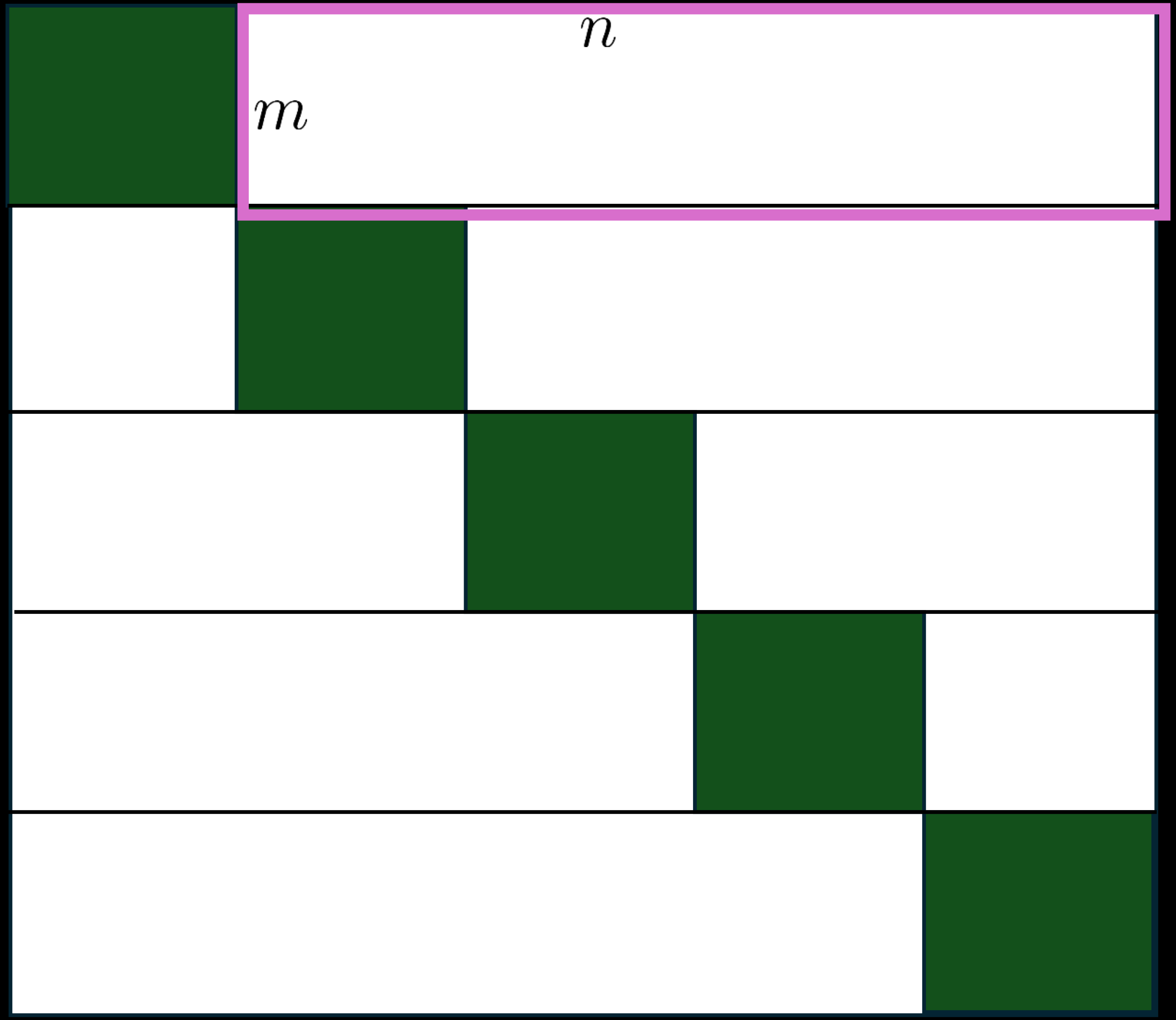
Real part of total field (acoustic waves)

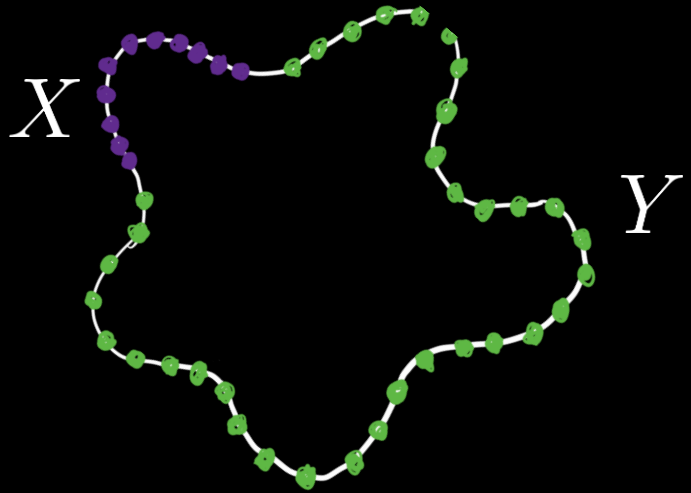


Magnitude of total field (trapped acoustic waves)

(W., Vaes, Gopal, & Martinsson, 2025) , See also: (Martinsson,2019) and refs. therein

Motivation: HSS hierarchical matrices

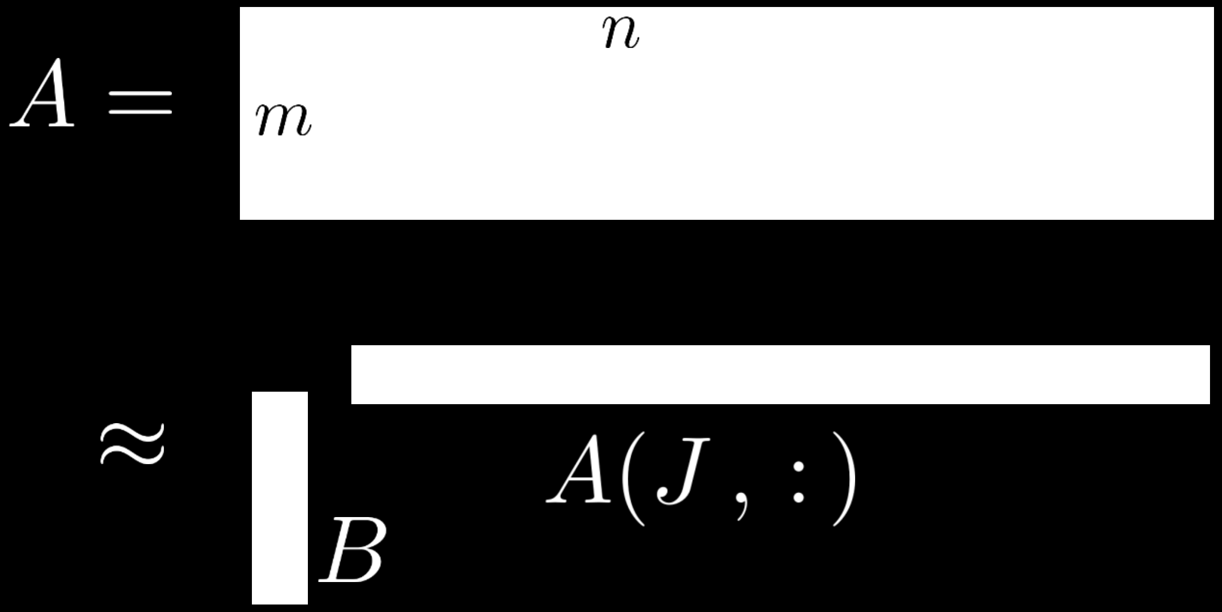




Proxy points: 46 seconds

Standard method (CPQR): 44 min.

Interpolative decompositions



$$J \subset \{1, 2, \cdots, n\}$$

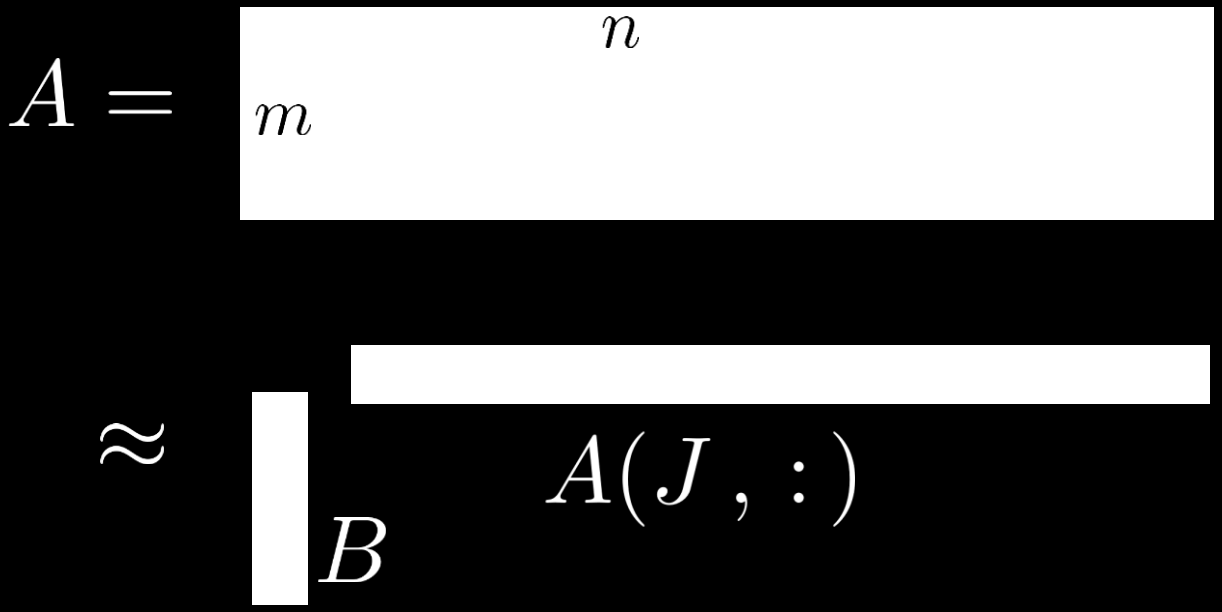
Goal: construct low rank approximations to “highly rectangular” submatrices: m << n.

Use an interpolative decomposition: saves storage + other benefits

The row subset selection problem:

Which k rows of A approximately span the rowspace of A ?

Interpolative decompositions



$$J \subset \{1, 2, \cdots, n\}$$

Goal: construct low rank approximations to "highly rectangular" submatrices: m << n.

Use an interpolative decomposition: saves storage + other benefits

Can we solve with a cost sublinear in n?

The proxy point idea

$$A = \begin{bmatrix} K(x_1,y_1) & K(x_1,y_2) & K(x_1,y_3) & K(x_1,y_4) & \cdots & K(x_1,y_n) \\ \boxed{K(x_2,y_1)} & \boxed{K(x_2,y_2)} & \boxed{K(x_2,y_3)} & \boxed{K(x_2,y_4)} & \boxed{\cdots} & \boxed{K(x_2,y_n)} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ K(x_m,y_1) & K(x_m,y_2) & K(x_m,y_3) & K(x_m,y_4) & \cdots & K(x_m,y_n) \end{bmatrix}$$

$$A_{\text{proxy}} = \begin{bmatrix} K(x_1,z_1) \cdots K(x_1,z_r) \\ \boxed{K(x_2,z_1) \cdots K(x_2,z_r)} \\ \vdots \qquad\qquad \vdots \\ K(x_m,z_1) \cdots K(x_m,z_r) \end{bmatrix}$$

1. Solve the row subset selection problem with A_{proxy}

$$A_{\text{proxy}} \approx B A_{\text{proxy}}(J, :)$$

2. Replace the rows with rows of A:

$$A \approx BA(J, :)$$

The proxy point idea: "near-set points"

$$A = \begin{bmatrix} K(x_1,y_1) & K(x_1,y_2) & K(x_1,y_3) & K(x_1,y_4) & \cdots & K(x_1,y_n) \\ K(x_2,y_1) & K(x_2,y_2) & K(x_2,y_3) & K(x_2,y_4) & & K(x_2,y_n) \\ \vdots & \vdots & & \vdots & & \vdots \\ K(x_m,y_1) & K(x_m,y_2) & K(x_m,y_3) & K(x_m,y_4) & \cdots & K(x_m,y_n) \end{bmatrix}$$

$$A_{\text{proxy}} = \begin{bmatrix} K(x_1,z_1) & \cdots & K(x_1,z_r) \\ K(x_2,z_1) & \cdots & K(x_2,z_r) \\ \vdots & & \vdots \\ K(x_m,z_1) & \cdots & K(x_m,z_r) \end{bmatrix}$$

Example Heuristics:

(i) Cluster: Set $z_1 = \text{mean}(y_1, \ldots, y_p)$

(ii) Subselect columns of A

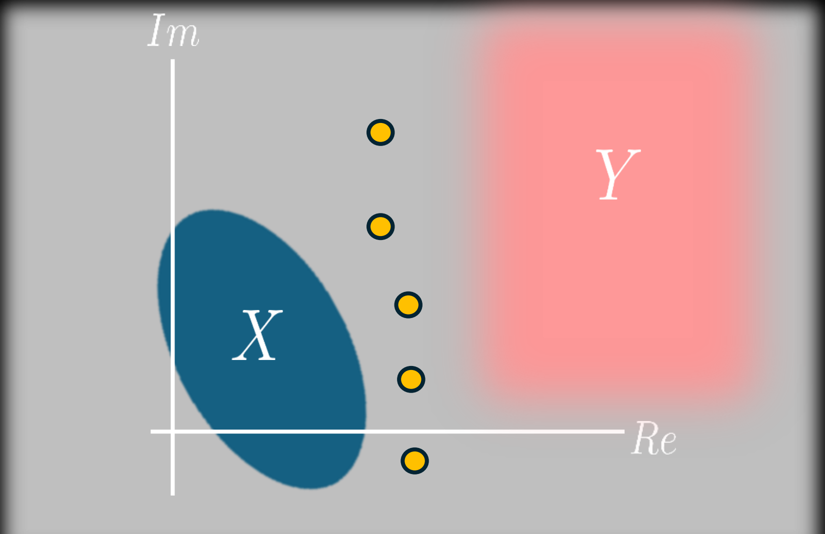
Too expensive for practical use!

The proxy point idea: "between-set points"

$$A = \begin{matrix} K(x_1,y_1) & K(x_1,y_2) & K(x_1,y_3) & K(x_1,y_4) & \cdots & K(x_1,y_n) \\ K(x_2,y_1) & K(x_2,y_2) & K(x_2,y_3) & K(x_2,y_4) & & K(x_2,y_n) \\ \vdots & \vdots & & \vdots & & \vdots \\ K(x_m,y_1) & K(x_m,y_2) & K(x_m,y_3) & K(x_m,y_4) & \cdots & K(x_m,y_n) \end{matrix}$$

Choose { z_1, z_2, ..., z_r} as "transitive source points"

$$A_{\text{proxy}} = \begin{matrix} K(x_1,z_1) \cdots K(x_1,z_r) \\ K(x_2,z_1) \cdots K(x_2,z_r) \\ \vdots \qquad\qquad \vdots \\ K(x_m,z_1) \cdots K(x_m,z_r) \end{matrix}$$

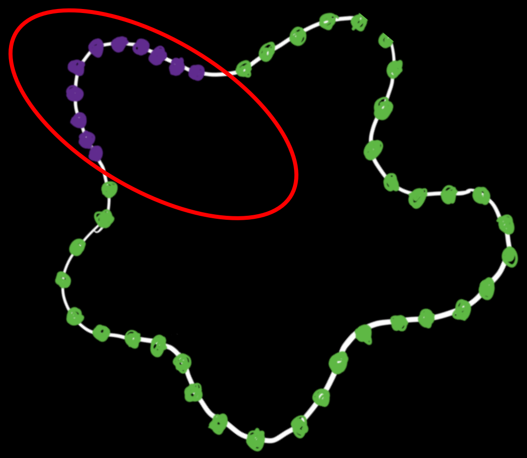


The proxy point idea: "between-set points"

$$A = \begin{bmatrix} K(x_1,y_1) & K(x_1,y_2) & K(x_1,y_3) & K(x_1,y_4) & \cdots & K(x_1,y_n) \\ K(x_2,y_1) & K(x_2,y_2) & K(x_2,y_3) & K(x_2,y_4) & & K(x_2,y_n) \\ \vdots & \vdots & & \vdots & & \vdots \\ K(x_m,y_1) & K(x_m,y_2) & K(x_m,y_3) & K(x_m,y_4) & \cdots & K(x_m,y_n) \end{bmatrix}$$

$$A_{\text{proxy}} = \begin{bmatrix} K(x_1,z_1) \cdots K(x_1,z_r) \\ K(x_2,z_1) \cdots K(x_2,z_r) \\ \vdots \qquad\qquad \vdots \\ K(x_m,z_1) \cdots K(x_m,z_r) \end{bmatrix}$$

Heuristic: the "proxy ring":



A theoretical framework for understanding proxy point methods

Questions:

1. When do we expect that proxy point methods will be effective? What needs to be true about X, Y, K ?

2. Given X, Y, K, how many proxy points are needed to achieve a specified accuracy?

3. How can one choose "good" proxy points?

Answers:

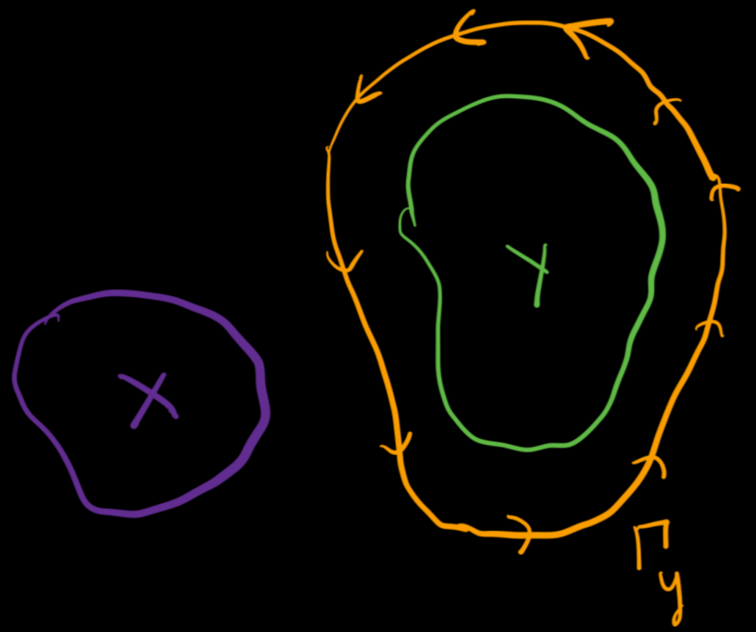
Rational approximation theory!

See also: Ye, Xia & Ying (2020)

Introducing rational functions

Assume $K(\hat{x}, \cdot)$ is analytic in $\mathbb{C}$ except near X.

For $y \in Y$, $\quad K(\hat{x}, y) = \int_{\Gamma_y} \frac{K(\hat{x}, z)}{z - y} \mathrm{d}z.$



See also: Trefethen & Horning (2025), Webb (2025)

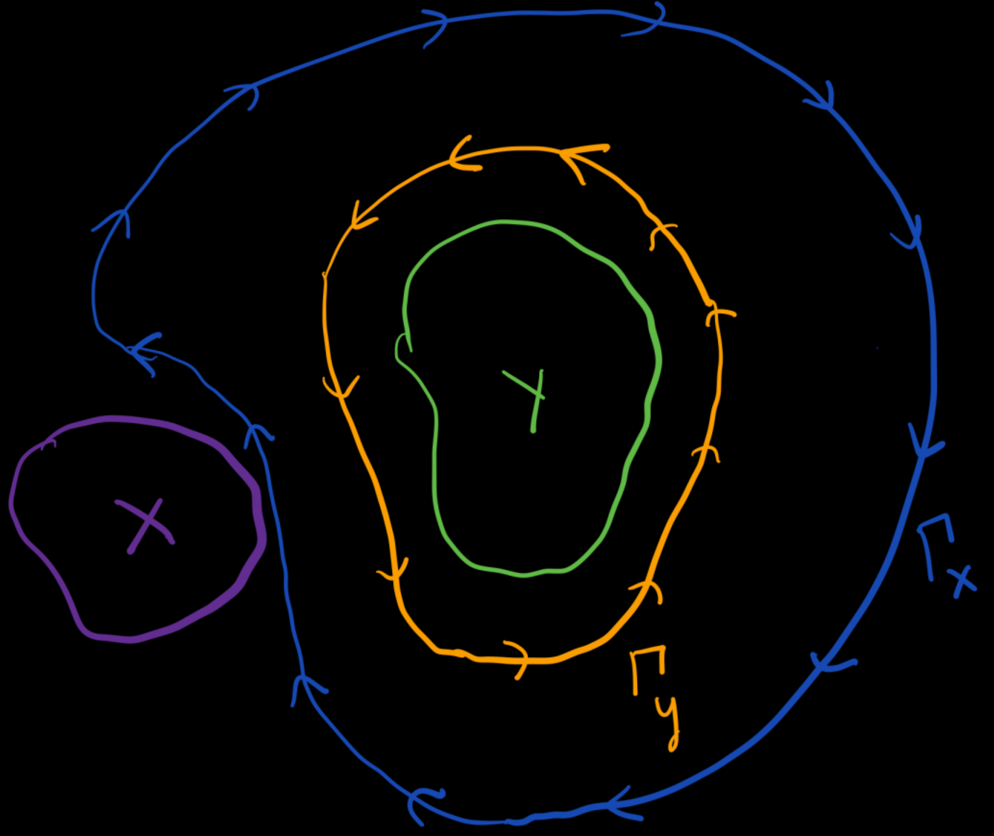
Sneaking in rational functions

Assume $K(\hat{x}, \cdot)$ is analytic in $\mathbb{C}$ except near

For $y \in Y$, $\quad K(\hat{x}, y) = \int_{\Gamma_y} \frac{K(\hat{x}, z)}{z - y} \mathrm{d}z.$

$$K(\hat{x}, y) = \int_{\Gamma_y \cup \Gamma_x} \frac{K(\hat{x}, z)}{z - y} \chi(z) \mathrm{d}z,$$

$$\chi(z) = \begin{cases} 1, & z \in \Gamma_y, \\ 0, & z \in \Gamma_x. \end{cases}$$



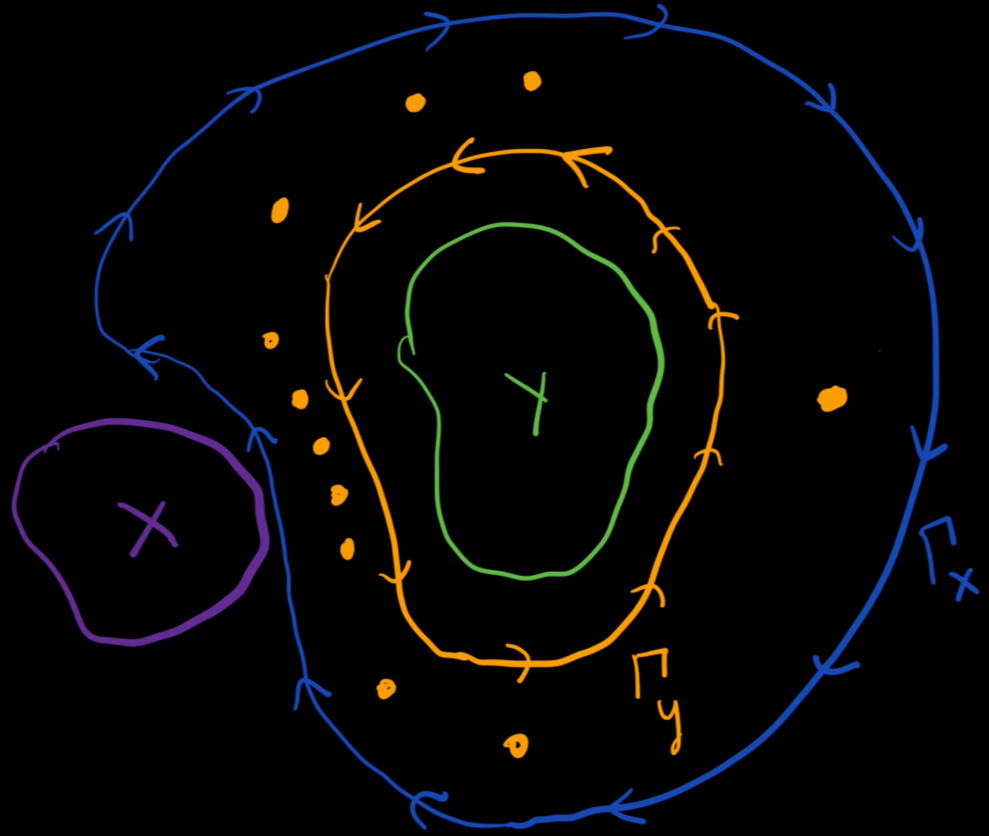
See also: Trefethen & Horning (2025), Webb (2025)

Rational approximation

$$\text{Let} \quad \eta_r(z) = \sum_{\ell=1}^{r} \frac{\gamma_\ell}{z - z_\ell} + C, \quad \eta_r(z) \approx \chi(z)$$

$$\text{Then,} \;\; K(\hat{x}, y) \approx \int_{\Gamma_y \cup \Gamma_x} \frac{K(\hat{x}, z)}{z - y} \eta_r(z) \mathrm{d}z,$$

$$\boxed{K(x, y) \approx \sum_{\ell=1}^{r} \frac{-\gamma_\ell K(x, z_\ell)}{z_\ell - y}}$$



See also: Trefethen & Horning (2025), Webb (2025)

Poles are proxy points!

AKA, the matrix A!

$$K\left(\begin{bmatrix} x_1 \\ \vdots \\ x_m \end{bmatrix}, \begin{bmatrix} y_1 & \cdots & y_n \end{bmatrix}\right) \approx$$

Set as A_{proxy}: Use to solve the row subset selection problem!

$$\boxed{\begin{bmatrix} K(x_1, z_1) & \cdots & K(x_1, z_r) \\ K(x_2, z_1) & \cdots & K(x_2, z_r) \\ \vdots & & \vdots \\ \vdots & & \vdots \\ K(x_m, z_1) & \cdots & K(x_m, z_r) \end{bmatrix}} \begin{bmatrix} -\gamma_1 & & \\ & \ddots & \\ & & -\gamma_r \end{bmatrix} \begin{bmatrix} \frac{1}{z_1 - y_1} & \cdots & \cdots & \frac{1}{z_1 - y_n} \\ \vdots & \cdots & \cdots & \vdots \\ \frac{1}{z_r - y_1} & \cdots & \cdots & \frac{1}{z_r - y_n} \end{bmatrix}$$

Error bounds

Theorem (W. & Vangala):

Given $X \subset G, Y \subset F$, with F being simply connected, and G well-separated from F in $\mathbb{C}$, suppose that for all $\hat{x} \in X$, $K(\hat{x}, \cdot)$ is analytic outside G. Then,

$$\frac{\|K - K_z\|_{\mathcal{L}^\infty(X \cup Y)}}{\|K\|_{\mathcal{L}^\infty(X \cup Y)}} \leq \frac{C}{\delta} \|\chi(z) - \eta_r(z)\|_{\Gamma_y \cup \Gamma_x},$$

where δ is the distance from Y to Γ_y and C depends only on properties of K, G, and F.

Key observation: We know a lot about approximating the characteristic function with rationals!

(This is a "Zolotarev" problem!)

Error bounds

Corollary:

With K, X, Y as before, there exists a set of proxy points $\{z_1, z_2, \cdots, z_r\}$ and a small number r_0 such that for $r > r_0$, an interpolative decomposition $BA(J,:)$ can be constructed with error

$$\|A - BA(J,:)\|_F \le C_1 p(m,n,r) h^{-r/2} + \frac{C_2}{\delta} h^{-r},$$

where $h = \exp(1/\mathrm{cap(G,F)}) > 1$ and $p(r,m,n) = \sqrt{mn}(1 + r + \sqrt{r(m-n)}$.

We need $r = \mathcal{O}\left(\log 1/\epsilon + \log(\sqrt{mn})\right)$ proxy points to achieve $\mathcal{O}(\epsilon)$ error!

Rubin, Townsend & Wilber (2020)

Even better error bounds!

Modified corollary (via M. Webb, 2025):

With K, X, Y as before, there exists a set of proxy points $\{z_1, z_2, \cdots, z_r\}$ and a small number r_0 such that for $r > r_0$, an interpolative decomposition $BA(J,:)$ can be constructed with error

$$\|A - BA(J,:)\|_F \leq C_1 p(m,n,r) h^{-r/\cancel{2}} + \frac{C_2}{\delta} h^{-2r},$$

where $h = \exp(1/\mathrm{cap}(\mathrm{G},\mathrm{F})) > 1$ and $p(r,m,n) = \sqrt{mn}(1 + r + \sqrt{r(m-n)})$.

$$K(\hat{x}, y) \approx \int_{\Gamma_\phi} \frac{K(\hat{x}, z)}{z - y}\left(1 - \frac{\phi_r(y)}{\phi_r(z)}\right) \mathrm{d}z,$$ $\phi_r(z)$ is large on Γ_ϕ (near X), and small on Y.

Webb (2025)

Two families of Zolotarev rationals, two styles of proxy points

Type IV Zolotarev rationals

$$\eta_r(z) = \text{argmin}_{t\in\mathcal{R}_r} \max_{z\in X\cup Y} |t(z) - \chi(z)|$$

Poles of $\eta_r(z)$ = "in-between set" proxy points

Type III Zolotarev rationals

$$\phi_r(z) = \text{argmin}_{t\in\mathcal{R}_r} \frac{\max_{z\in Y} |t(z)|}{\min_{z\in X} |t(z)|}$$

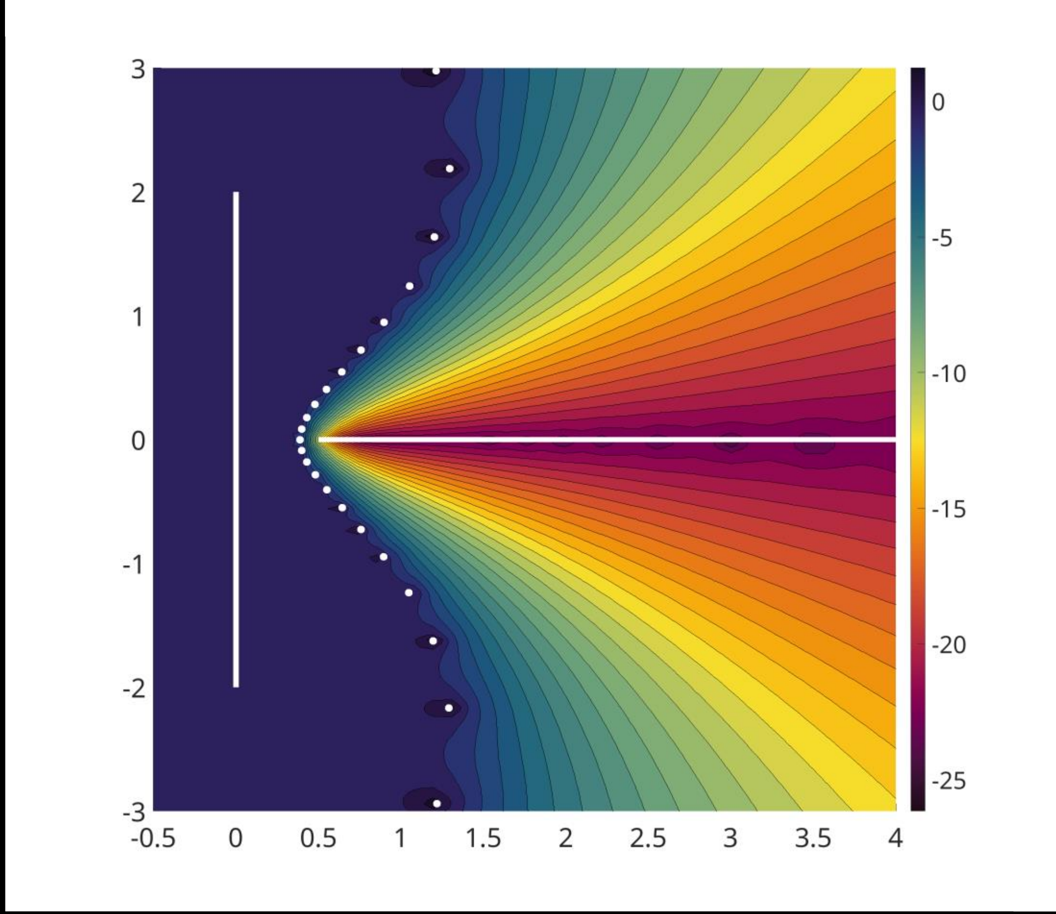
Zeros of $\phi_r(z)$ = "near-set" proxy points

Trefethen & Wilber (2024)

Two families of Zolotarev rationals, two styles of proxy points

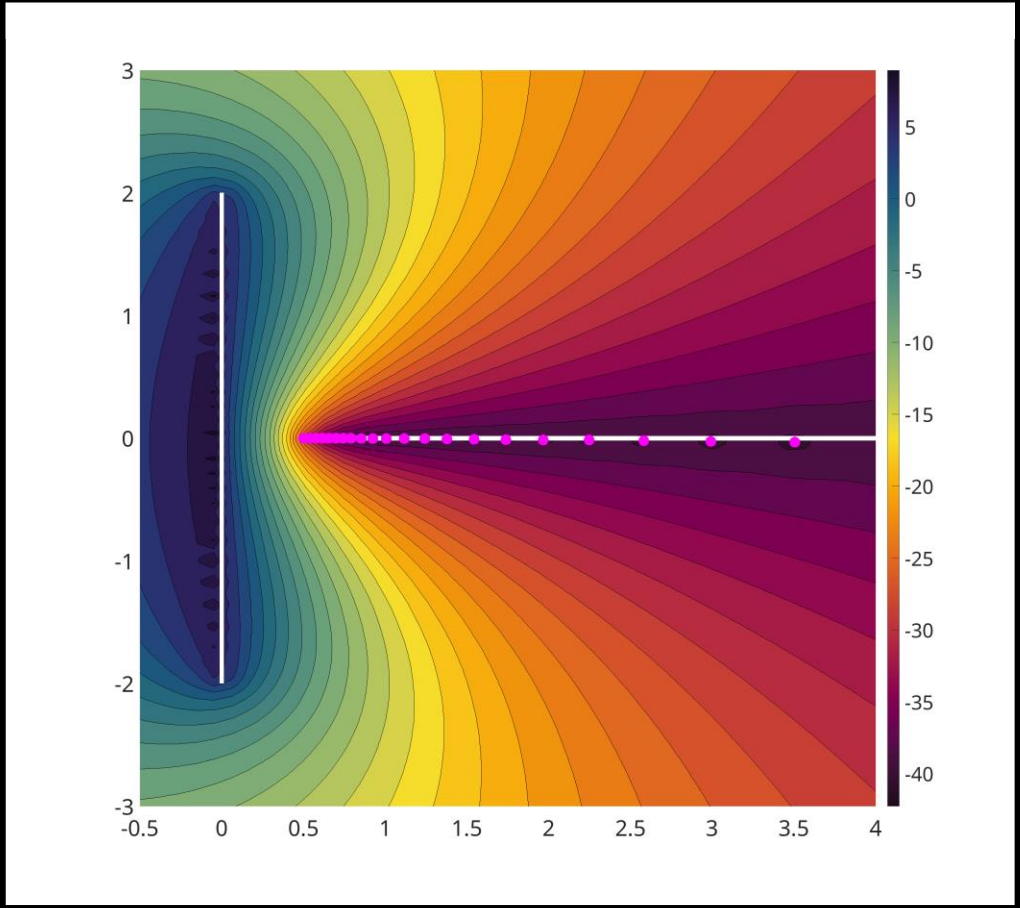
Type IV Zolotarev rationals

$$\log|\eta_r(z)|$$



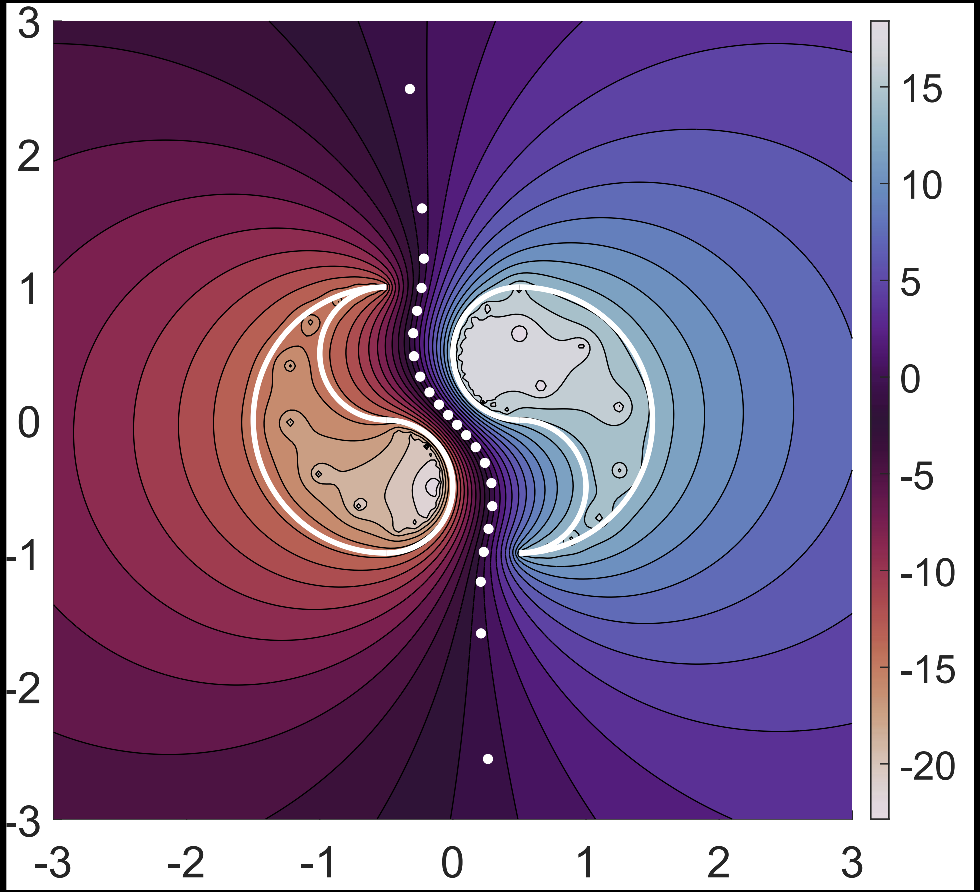
Type III Zolotarev rationals

$$\log|\phi_r(z)|$$



Trefethen & Wilber (2024)

Computationally constructing Zolotarev rationals



$$\log\left(\frac{|\eta_r(x)-1|}{|\eta_r(x)|}\right)$$

1. Construct $\eta_r(z) \approx \chi(z)$ by applying modified AAA on boundaries of X and Y.

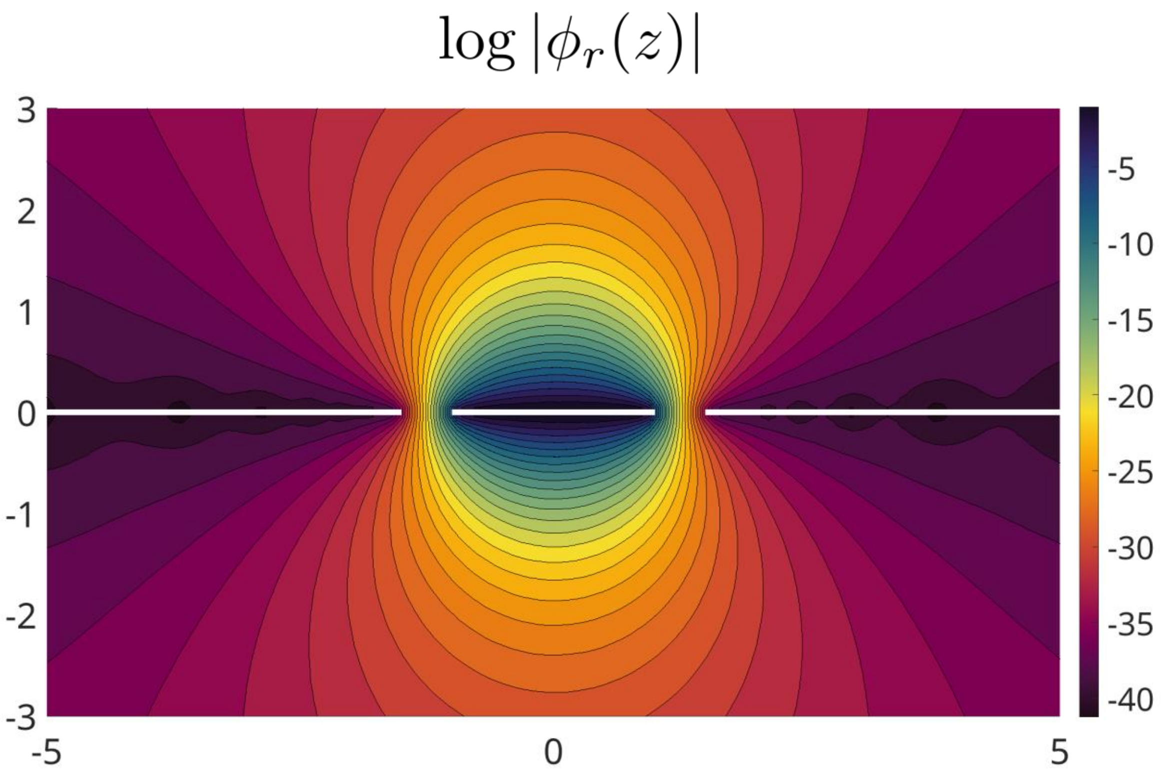
2. Use equivalence theorem to get ϕ_r:

$$\phi_r(z) = \sqrt{\sigma}\frac{(1-\sigma)(\eta_r(z)-1)}{(1+\sigma)\eta_r(z)}$$

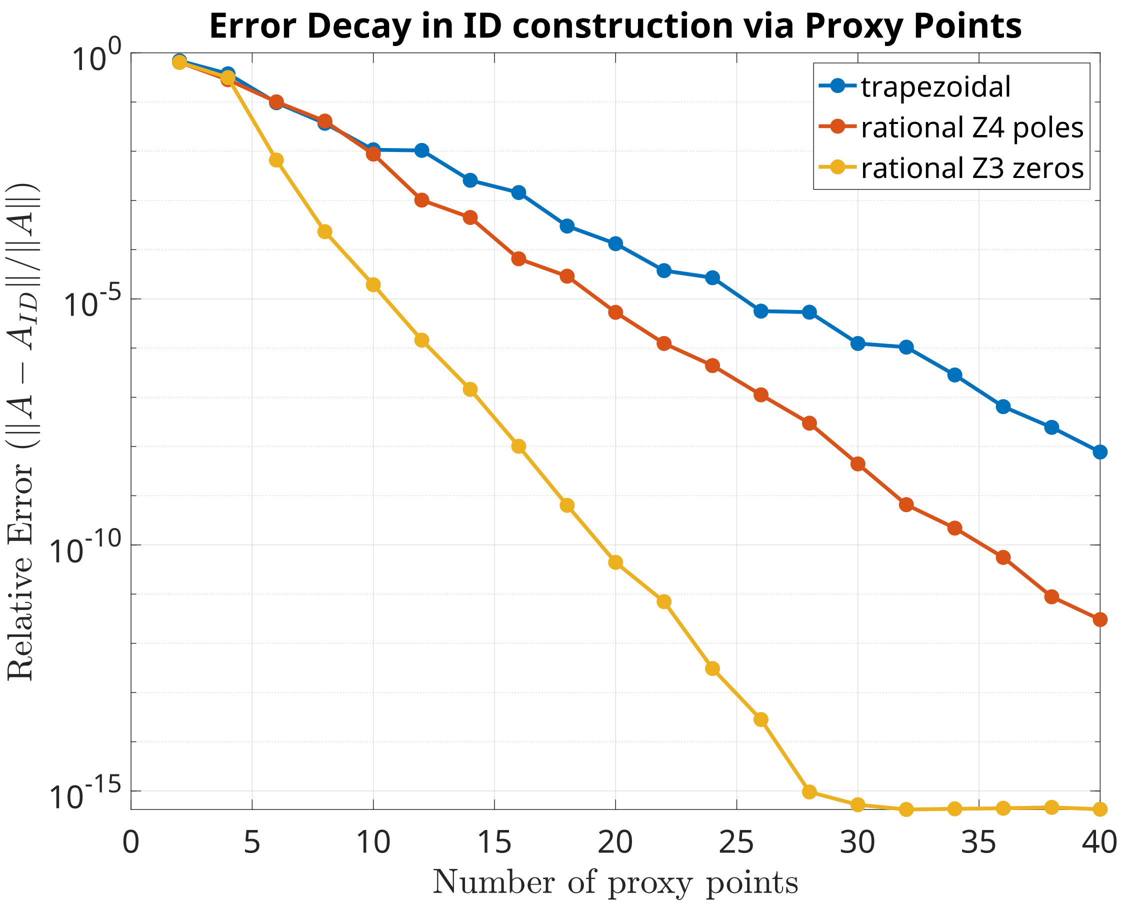
3. Zeros or poles for either one can be computed in $\mathcal{O}(r^3)$.

Trefethen & Wilber (2024)

A practical algorithm for picking proxy points

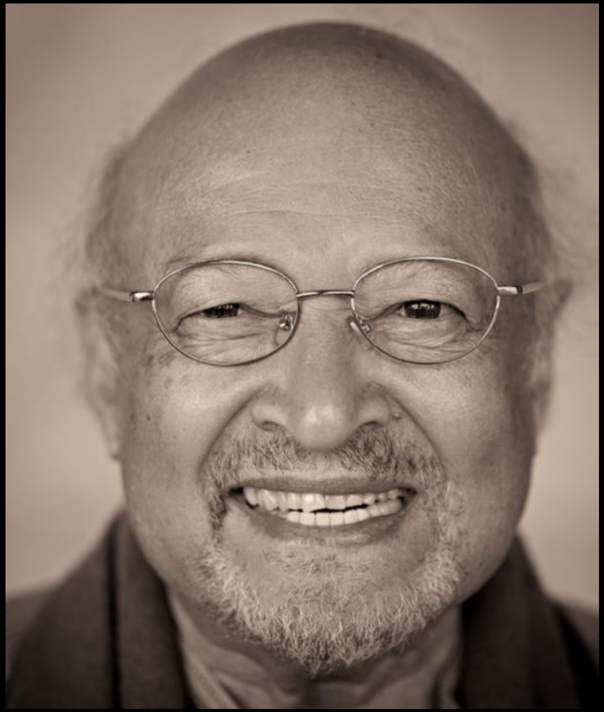


$$A \in \mathbb{C}^{200\times 1000} \qquad K(x,y) = \frac{1}{|x-y|^2}$$

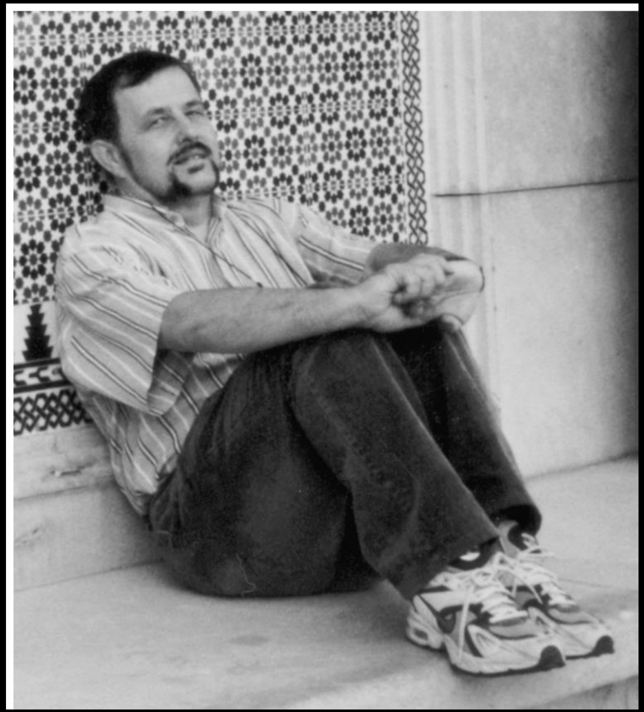


II. Cauchy-like matrices: the core of displacement structure

Matrices with displacement structure: an irresistible game!



Thomas Kailath (Stanford Univ.)



Georg Heinig (1947-2005)

Victor Pan (Lehman College, CUNY)

Levinson (1947), Durbin (1960), Musicus (1988), Gohberg, Kailath & Olshevsky (1995), Kailath & Sayed (1995), Pan (2001), Heinig (1995), Martinsson, Rohklin & Tygert (2005), Chandrasekaran, Gu, Sun, Xia, & Zhu (2008), many more…

Displacement structure

$$A = \mathbb{C}^{m\times m},\ B \in \mathbb{C}^{n\times n},\ X, F \in \mathbb{C}^{m\times n}$$

$$AX - XB = F.$$

"X has (A, B) displacement structure"

Appears with X as an unknown: discretization of PDEs, reduced order modeling, signal processing (see Simoncini SIAM REV, 2016 + ref. therein).

Special (A, B, F) triplets characterize properties of X for structured matrices: e.g., X = Toeplitz, Hankel, Cauchy, Vandermonde, and more.

A and B are selected so that F is very low rank.

See Kailath & Sayed (1995), Heinig (1995), Pan (2001), Beckermann & Townsend (2016)

Example: Vandermonde matrix

$$\begin{pmatrix} \gamma_1 & & & \\ & \gamma_2 & & \\ & & \ddots & \\ & & & \gamma_m \end{pmatrix} \begin{pmatrix} (\gamma_1)^0 & (\gamma_1)^1 & \cdots & (\gamma_1)^{n-1} \\ (\gamma_2)^0 & (\gamma_2)^1 & \cdots & (\gamma_2)^{n-1} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ (\gamma_m)^0 & (\gamma_m)^1 & \cdots & (\gamma_m)^{n-1} \end{pmatrix} - \begin{pmatrix} (\gamma_1)^0 & (\gamma_1)^1 & \cdots & (\gamma_1)^{n-1} \\ (\gamma_2)^0 & (\gamma_2)^1 & \cdots & (\gamma_2)^{n-1} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ (\gamma_m)^0 & (\gamma_m)^1 & \cdots & (\gamma_m)^{n-1} \end{pmatrix} \begin{pmatrix} 0 & \cdots & 0 & 1 \\ 1 & 0 & & 0 \\ & \ddots & \ddots & \\ 0 & & 1 & 0 \end{pmatrix}$$

$$=$$

$$\begin{pmatrix} (\gamma_1)^1 & \cdots & (\gamma_1)^{n-1} & (\gamma_1)^n \\ (\gamma_2)^1 & \cdots & (\gamma_2)^{n-1} & (\gamma_2)^n \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ (\gamma_m)^1 & \cdots & (\gamma_m)^{n-1} & (\gamma_m)^n \end{pmatrix} - \begin{pmatrix} (\gamma_1)^1 & \cdots & (\gamma_1)^{n-1} & (\gamma_1)^0 \\ (\gamma_2)^1 & \cdots & (\gamma_2)^{n-1} & (\gamma_2)^0 \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ (\gamma_m)^1 & \cdots & (\gamma_m)^{n-1} & (\gamma_m)^0 \end{pmatrix}$$

Diagonalization leads to Cauchy-like matrices

A Cauchy-like matrix C satisfies

$$D_x C - C D_y = LR^*, \text{ where}$$

$$D_x = \begin{pmatrix} x_1 & & & \\ & x_2 & & \\ & & \ddots & \\ & & & x_m \end{pmatrix}, \qquad D_y = \begin{pmatrix} y_1 & & & \\ & y_2 & & \\ & & \ddots & \\ & & & y_m \end{pmatrix}.$$

$$\boxed{C = \hat{C} \circ (LR^*)}$$

Diagonalization leads to Cauchy-like matrices

An *expensive* way to solve $AX - XB = F$ is to diagonalize A and B.

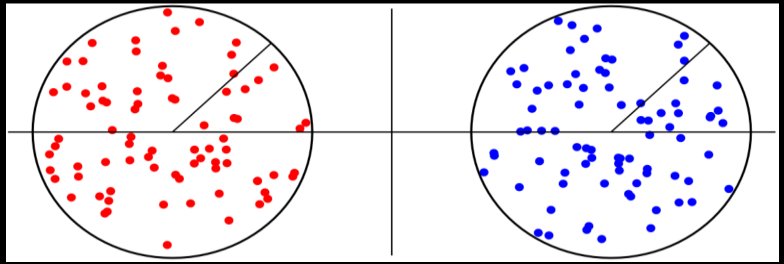
$$V_a \Lambda_a V_a^{-1} X - X V_b \Lambda_b V_b^{-1} = F$$

$$\Longrightarrow$$

$$\Lambda_a (V_a^{-1} X V_b) - (V_a^{-1} X V_b) \Lambda_b = V_a^{-1} F V_b$$

$$\Longrightarrow$$

$$V_a^{-1} X V_b = \boxed{\frac{1}{\Lambda_a - \Lambda_b}} \circ \left(V_a^{-1} F V_b\right)$$



Convergence behavior for iterative methods, low rank properties of X

(Beckermann & Townsend, 2019), (Sabino, 2008), (Penzl, 1999), (Benner, Truhar & Li, 2009), (Li & White, 2002), (Druskin, Knizhnerman & Simoncini, 2011), (Peaceman & Rachford, 1955), (Lu & Wachspress, 1991), (Townsend & W., 2018), (Rubin, Townsend, & W., 2022)

Toeplitz-to-Cauchy via diagonalization...

If $\hat{T} \in \mathbb{C}^{m \times m}$ is circulant...

$$\mathcal{F}\hat{T}\mathcal{F}^* = \begin{bmatrix} \omega_1 & & \\ & \ddots & \\ & & \omega_m \end{bmatrix}, \quad \text{with } |\omega_j| = 1.$$

$$\boxed{\mathcal{O}(m \log m) \text{ direct solver for } Tx = b.}$$

Toeplitz-to-Cauchy

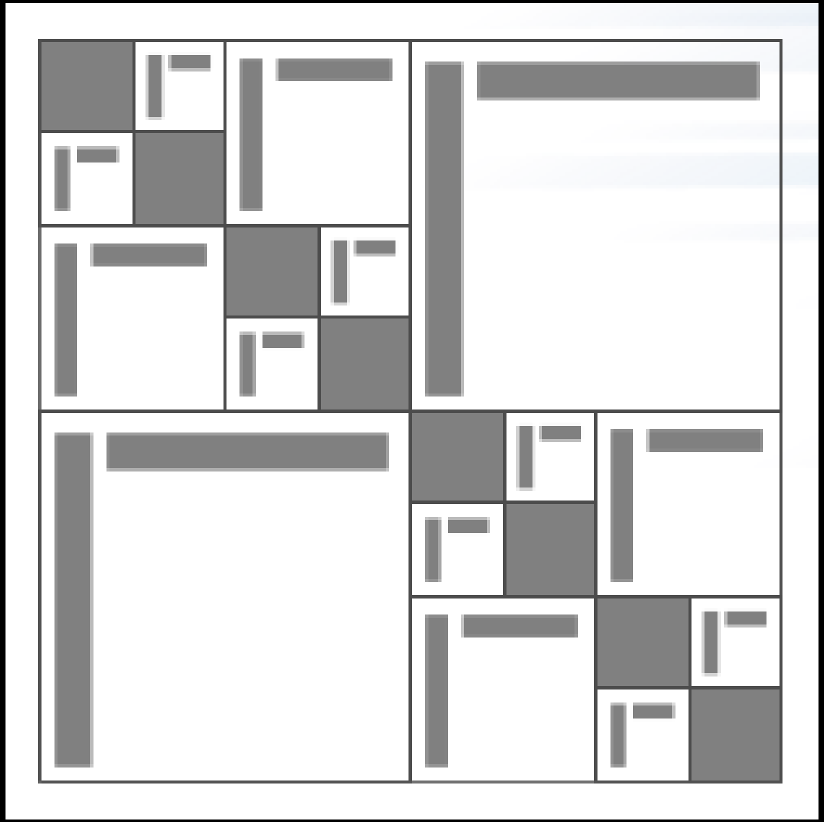
If $T \in \mathbb{C}^{m \times m}$ is Toeplitz, but not circulant...

Use the fact that $Q_1 T - T Q_{-1} = LR^*$:

$$\implies \quad \mathcal{F} T \mathcal{F}^* = C, \quad \text{with}$$

$$C_{jk} = \frac{(\mathcal{F} L R^* \mathcal{F}^*)_{jk}}{\omega_j - \omega_{k+1/2}},$$

$\boxed{\mathcal{O}(m \log^2 m) \text{ direct solver for } Tx = b.}$



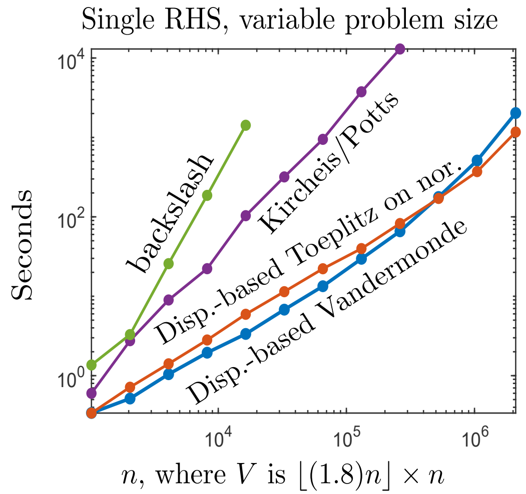
(Beckermann, Kressner, & W., 2025), Martinsson, Rohklin & Tygert (2005),
(, Gu, Sun, Xia, & Zhu ,2008), (Xi, Xia, Cauley, Balakrishnan, 2014)

Additional displacement-based applications

Vandermonde-to-Cauchy

- $\mathcal{O}(m^2 \log^2 m)$ direct solver for $Vx = b$, where V is a nonuniform DFT matrix.
- Effective for clustered/irregularly spaced measurements
- Fast least-squares solver for hierarchical matrices

(See *W., Epperly, & Barnett, 2024*)



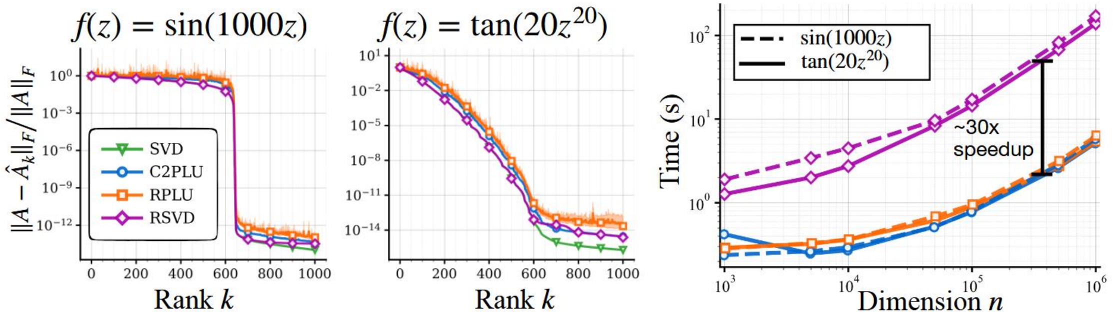
Fast Cauchy-like compression

- Uses partial LU factorizations for Cauchy-like matrices with randomized pivoting strategy via rejection sampling.
- Uses classic results on Cauchy-like matrices:

(i) Schur complement of Cauchy-like matrix is also Cauchy-like

(ii) Barnes-Hut = fast matvecs for Cauchy-like matrices

(See *Aurele-Gilles & W., 2026*)



Thank you!

heatherw3521.github.io