Nonlinear Eigenvalue Problems: Interpolatory Algorithms and Transient Dynamics

Mark Embree · Virginia Tech

with

Serkan Gugercin
Jonathan Baker, Michael Brennan, Alexander Grimm
Virginia Tech

SIAM Conference on Applied Linear Algebra

Hong Kong · May 2018

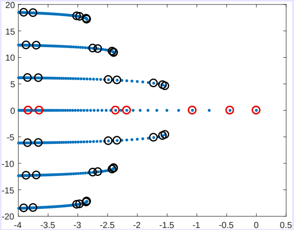
Thanks to: International Linear Algebra Society (ILAS)
US National Science Foundation DMS-1720257

a talk in two parts . . .

rational interpolation for nlevps

Rational / Loewner techniques for nonlinear eigenvalue problems, motivated by algorithms from model reduction.

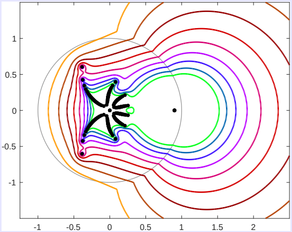
- Structure Preserving Rational Interpolation
- Data-Driven Rational Interpolation Matrix Pencils
- Minimal Realization via Rational Contour Integrals



transients for delay equations

Scalar delay equations: a case-study for how one can apply pseudospectra techniques to analyze the transient behavior of a dynamical system.

- *Finite dimensional nonlinear* problem $\Rightarrow$ *infinite dimensional linear* problem
- Pseudospectral theory applies to the linear problem, *but the choice of norm is important*



nonlinear eigenvalue problems: the final frontier?

problem		typical # eigenvalues
standard eigenvalue problem	$(\mathbf{A} - \lambda \mathbf{I})\mathbf{v} = \mathbf{0}$	n
generalized eigenvalue problem	$(\mathbf{A} - \lambda \mathbf{E})\mathbf{v} = \mathbf{0}$	n
quadratic eigenvalue problem	$(\mathbf{K} + \lambda \mathbf{D} + \lambda^2 \mathbf{M})\mathbf{v} = \mathbf{0}$	$2n$
polynomial eigenvalue problem	$(\sum_{k=0}^{d} \lambda^k \mathbf{A}_k)\mathbf{v} = \mathbf{0}$	dn
nonlinear eigenvalue problem	$(\sum_{k=0}^{d} f_k(\lambda) \mathbf{A}_k)\mathbf{v} = \mathbf{0}$	∞

nonlinear eigenvalue problems: the final frontier?

problem		*typical* # eigenvalues
standard eigenvalue problem	$(\mathbf{A} - \lambda \mathbf{I})\mathbf{v} = \mathbf{0}$	n
generalized eigenvalue problem	$(\mathbf{A} - \lambda \mathbf{E})\mathbf{v} = \mathbf{0}$	n
quadratic eigenvalue problem	$(\mathbf{K} + \lambda \mathbf{D} + \lambda^2 \mathbf{M})\mathbf{v} = \mathbf{0}$	$2n$
polynomial eigenvalue problem	$(\sum_{k=0}^{d} \lambda^k \mathbf{A}_k)\mathbf{v} = \mathbf{0}$	dn
nonlinear eigenvalue problem	$(\sum_{k=0}^{d} f_k(\lambda) \mathbf{A}_k)\mathbf{v} = \mathbf{0}$	∞
nonlinear eigenvector problem	$F(\lambda, \mathbf{v}) = \mathbf{0}$	∞

a basic nonlinear eigenvalue problem

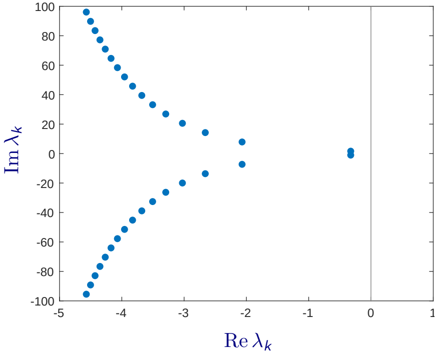
Consider the simple *scalar* delay differential equation

$$\mathbf{x}'(t) = -\mathbf{x}(t-1).$$

Substituting the ansatz $x(t) = \mathrm{e}^{\lambda t}$ yields the *nonlinear eigenvalue problem*

$$T(\lambda) = 1 + \lambda \mathrm{e}^{\lambda} = 0.$$

32 (of infinitely many) eigenvalues of T for this *scalar* ($n = 1$) equation:



eigenvalues determined by the Lambert-W function

See, e.g., [Michiels & Niculescu 2007]

nonlinear eigenvalue problems: many resources

Nonlinear eigenvalue problems have classical roots, but now form a fast-moving field with many excellent resources and new algorithms.

- Helpful surveys:
  Mehrmann & Voss, *GAMM*, [2004]
  Voss, *Handbook of Linear Algebra*, [2014]
  Güttel & Tisseur, *Acta Numerica* survey [2017]

- Software:
  NLEVP test collection [Betcke, Higham, Mehrmann, Schröder, Tisseur 2013]
  SLEPC contains NLEVP algorithm implementations [Roman et al.]

- Many algorithms based on Newton's method, rational approximation, linearization, contour integration, projection, etc.
  Incomplete list of contributors: Asakura, Bai, Betcke, Beyn, Effenberger, Güttel, Ikegami, Jarlebring, Kimura, Kressner, Leitart, Meerbergen, Michiels, Niculescu, Pérez, Sakurai, Tadano, Van Beeumen, Vandereycken, Voss, Yokota, ...

- Infinite dimensional nonlinear spectral problems are even more subtle:
  [Appell, De Pascale, Vignoli 2004] give *seven distinct definitions* of the spectrum.

Rational Interpolation

Algorithms

for

Nonlinear Eigenvalue Problems

rational interpolation of functions and systems

Rational interpolation problem.

Given points $\{z_j\}_{j=1}^{2r} \subset \mathbb{C}$ and data $\{f_j \equiv f(z_j)\}_{j=1}^{2r}$, find a rational function $R(z) = p(z)/q(z)$ of type $(r-1, r-1)$ such that

$$R(z_j) = f_j.$$

rational interpolation of functions and systems

Rational interpolation problem.
Given points $\{z_j\}_{j=1}^{2r} \subset \mathbb{C}$ and data $\{f_j \equiv f(z_j)\}_{j=1}^{2r}$, find a rational function $R(z) = p(z)/q(z)$ of type $(r-1, r-1)$ such that

$$R(z_j) = f_j.$$

Given *Lagrange basis functions* $\ell_j(z) = \prod_{\substack{k=1\\k\neq j}}^{r}(z - z_k)$ and *nodal polynomial* $\ell(z) = \prod_{k=1}^{r}(z - z_k)$,

$$R(z) = \frac{p(z)}{q(z)} = \frac{\sum_{j=1}^{r} \beta_j \ell_j(z)}{\sum_{j=1}^{r} w_j \ell_j(z)} = \frac{\sum_{j=1}^{r} \beta_j \frac{\ell_j(z)}{\ell(z)}}{\sum_{j=1}^{r} w_j \frac{\ell_j(z)}{\ell(z)}} = \frac{\sum_{j=1}^{r} \frac{\beta_j}{z - z_j}}{\sum_{j=1}^{r} \frac{w_j}{z - z_j}}$$

barycentric form

rational interpolation: barycentric perspective

Lagrange basis: $\ell_j(z) = \prod_{\substack{k=1 \\ k \neq j}}^{r} (z - z_k)$

$$R(z) = \frac{p(z)}{q(z)} = \frac{\displaystyle\sum_{j=1}^{r} \beta_j \ell_j(z)}{\displaystyle\sum_{j=1}^{r} w_j \ell_j(z)} = \frac{\displaystyle\sum_{j=1}^{r} \frac{\beta_j}{z - z_j}}{\displaystyle\sum_{j=1}^{r} \frac{w_j}{z - z_j}}$$

rational interpolation: barycentric perspective

$$\text{Lagrange basis: } \ell_j(z) = \prod_{\substack{k=1 \\ k \neq j}}^{r} (z - z_k)$$

$$R(z) \quad = \quad \frac{p(z)}{q(z)} \quad = \quad \frac{\displaystyle\sum_{j=1}^{r} \beta_j \ell_j(z)}{\displaystyle\sum_{j=1}^{r} w_j \ell_j(z)} \quad = \quad \frac{\displaystyle\sum_{j=1}^{r} \frac{\beta_j}{z - z_j}}{\displaystyle\sum_{j=1}^{r} \frac{w_j}{z - z_j}}$$

- Fix $\{\beta_j = f_j w_j\}_{j=1}^{r}$ to interpolate at $z_1, \ldots, z_r$: $R(z_j) = f_j$.

rational interpolation: barycentric perspective

$$\text{Lagrange basis: } \ell_j(z) = \prod_{\substack{k=1\\k\neq j}}^{r} (z - z_k)$$

$$R(z) \quad = \quad \frac{p(z)}{q(z)} \quad = \quad \frac{\displaystyle\sum_{j=1}^{r} \beta_j \ell_j(z)}{\displaystyle\sum_{j=1}^{r} w_j \ell_j(z)} \quad = \quad \frac{\displaystyle\sum_{j=1}^{r} \frac{\beta_j}{z - z_j}}{\displaystyle\sum_{j=1}^{r} \frac{w_j}{z - z_j}}$$

- Fix $\{\beta_j = f_j w_j\}_{j=1}^{r}$ to interpolate at $z_1, \ldots, z_r$: $R(z_j) = f_j$.
- Determine $w_1, \ldots, w_r$ to interpolate at $z_{r+1}, \ldots, z_{2r}$:

$$R(z_k) \quad = \quad \frac{\displaystyle\sum_{j=1}^{r} \frac{f_j\, w_j}{z_k - z_j}}{\displaystyle\sum_{j=1}^{r} \frac{w_j}{z_k - z_j}} \quad = \quad f_k \qquad \Longrightarrow \qquad \sum_{j=1}^{r} \frac{f_j\, w_j}{z_k - z_j} = \sum_{j=1}^{r} \frac{f_k\, w_j}{z_k - z_j}$$

rational interpolation: barycentric perspective

- Fix $\{\beta_j = f_j w_j\}_{j=1}^r$ to interpolate at $z_1, \ldots, z_r$: $r(z_j) = f_j$.
- Determine $w_1, \ldots, w_r$ to interpolate at $z_{r+1}, \ldots, z_{2r}$:

$$R(z_k) = f_k \quad \Longrightarrow \quad \sum_{j=1}^{r} \frac{f_j w_j}{z_k - z_j} = \sum_{j=1}^{r} \frac{f_k w_j}{z_k - z_j} \quad \Longrightarrow \quad \sum_{j=1}^{r} \frac{f_j - f_k}{z_j - z_k} w_j = 0.$$

rational interpolation: barycentric perspective

- Fix $\{\beta_j = f_j w_j\}_{j=1}^r$ to interpolate at $z_1, \ldots, z_r$: $r(z_j) = f_j$.
- Determine $w_1, \ldots, w_r$ to interpolate at $z_{r+1}, \ldots, z_{2r}$:

$$R(z_k) = f_k \quad \Longrightarrow \quad \sum_{j=1}^{r} \frac{f_j w_j}{z_k - z_j} = \sum_{j=1}^{r} \frac{f_k w_j}{z_k - z_j} \quad \Longrightarrow \quad \sum_{j=1}^{r} \frac{f_j - f_k}{z_j - z_k} w_j = 0.$$

$$\underbrace{\begin{bmatrix} \frac{f_1 - f_{r+1}}{z_1 - z_{r+1}} & \frac{f_2 - f_{r+1}}{z_2 - z_{r+1}} & \cdots & \frac{f_r - f_{r+1}}{z_r - z_{r+1}} \\ \frac{f_1 - f_{r+2}}{z_1 - z_{r+2}} & \frac{f_2 - f_{r+2}}{z_2 - z_{r+2}} & \cdots & \frac{f_r - f_{r+2}}{z_r - z_{r+2}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{f_1 - f_{2r}}{z_1 - z_{2r}} & \frac{f_2 - f_{2r}}{z_2 - z_{2r}} & \cdots & \frac{f_r - f_{2r}}{z_r - z_{2r}} \end{bmatrix}}_{\text{Loewner matrix, } \mathbb{L}} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_r \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

rational interpolation: barycentric perspective

- Fix $\{\beta_j = f_j w_j\}_{j=1}^r$ to interpolate at $z_1, \ldots, z_r$: $r(z_j) = f_j$.
- Determine $w_1, \ldots, w_r$ to interpolate at $z_{r+1}, \ldots, z_{2r}$:

$$R(z_k) = f_k \quad \Longrightarrow \quad \sum_{j=1}^{r} \frac{f_j w_j}{z_k - z_j} = \sum_{j=1}^{r} \frac{f_k w_j}{z_k - z_j} \quad \Longrightarrow \quad \sum_{j=1}^{r} \frac{f_j - f_k}{z_j - z_k} w_j = 0.$$

$$\underbrace{\begin{bmatrix} \frac{f_1 - f_{r+1}}{z_1 - z_{r+1}} & \frac{f_2 - f_{r+1}}{z_2 - z_{r+1}} & \cdots & \frac{f_r - f_{r+1}}{z_r - z_{r+1}} \\ \frac{f_1 - f_{r+2}}{z_1 - z_{r+2}} & \frac{f_2 - f_{r+2}}{z_2 - z_{r+2}} & \cdots & \frac{f_r - f_{r+2}}{z_r - z_{r+2}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{f_1 - f_{2r}}{z_1 - z_{2r}} & \frac{f_2 - f_{2r}}{z_2 - z_{2r}} & \cdots & \frac{f_r - f_{2r}}{z_r - z_{2r}} \end{bmatrix}}_{\text{Loewner matrix, } \mathbb{L}} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_r \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

- Barycentric rational interpolation algorithm [Antoulas & Anderson [1986]
- AAA (Adaptive Antoulas–Anderson) Method [Nakatsukasa, Sète, Trefethen, 2016]

rational interpolation: state space perspective

The rational interpolant $R(z)$ to f at $z_1, \ldots, z_{2r}$ can also be formulated in *state-space form* using Loewner matrix techniques.

$$R(z) = \mathbf{c}\,(\mathbb{L}_s - z\mathbb{L})^{-1}\mathbf{b},$$

where $\mathbf{c} = [f_{r+1}, \ldots, f_{2r}]$, $\mathbf{b} = [f_1, \ldots, f_r]^T$ and

$$\underbrace{\begin{bmatrix} \frac{z_1 f_1 - z_{r+1} f_{r+1}}{z_1 - z_{r+1}} & \cdots & \frac{z_r f_r - z_{r+1} f_{r+1}}{z_r - z_{r+1}} \\ \vdots & \ddots & \vdots \\ \frac{z_1 f_1 - z_{2r} f_{2r}}{z_1 - z_{2r}} & \cdots & \frac{z_r f_r - z_{2r} f_{2r}}{z_r - z_{2r}} \end{bmatrix}}_{\text{shifted Loewner matrix, } \mathbb{L}_s}, \quad \underbrace{\begin{bmatrix} \frac{f_1 - f_{r+1}}{z_1 - z_{r+1}} & \cdots & \frac{f_r - f_{r+1}}{z_r - z_{r+1}} \\ \vdots & \ddots & \vdots \\ \frac{f_1 - f_{2r}}{z_1 - z_{2r}} & \cdots & \frac{f_r - f_{2r}}{z_r - z_{2r}} \end{bmatrix}}_{\text{Loewner matrix, } \mathbb{L}}.$$

- State space formulation proposed by Mayo & Antoulas [2007]
- Natural approach for handling *tangential interpolation for vector data*
- For details, applications, and extensions, see [Antoulas, Lefteriu, Ionita 2017]

approach one: structure preserving rational interpolation

Scenario: $\mathbf{T}(\lambda) \in \mathbb{C}^{n\times n}$ has *large dimension n*.

Goal: Reduce dimension of $\mathbf{T}(\lambda)$ *but maintain the nonlinear structure*.
Smaller problem will be more amenable to dense nonlinear eigensolvers.

Method: Rational tangential interpolation of $\mathbf{T}(\lambda)^{-1}$ at r points, directions.

Iteratively Corrected Rational Interpolation method

approach one: structure preserving rational interpolation

Scenario: $\mathbf{T}(\lambda) \in \mathbb{C}^{n\times n}$ has *large dimension n*.

Goal: Reduce dimension of $\mathbf{T}(\lambda)$ *but maintain the nonlinear structure*.
Smaller problem will be more amenable to dense nonlinear eigensolvers.

Method: Rational tangential interpolation of $\mathbf{T}(\lambda)^{-1}$ at r points, directions.

Iteratively Corrected Rational Interpolation method

- Pick r *interpolation points* $\{z_j\}_{j=1}^{r}$ and *interpolation directions* $\{\mathbf{w}_j\}_{j=1}^{r}$.

approach one: structure preserving rational interpolation

Scenario: $\mathbf{T}(\lambda) \in \mathbb{C}^{n\times n}$ has *large dimension n*.

Goal: Reduce dimension of $\mathbf{T}(\lambda)$ *but maintain the nonlinear structure*.
Smaller problem will be more amenable to dense nonlinear eigensolvers.

Method: Rational tangential interpolation of $\mathbf{T}(\lambda)^{-1}$ at r points, directions.

Iteratively Corrected Rational Interpolation method

- Pick r *interpolation points* $\{z_j\}_{j=1}^r$ and *interpolation directions* $\{\mathbf{w}_j\}_{j=1}^r$.
- Construct a basis for projection (cf. *shift-invert Arnoldi*):

$$\mathbf{U} = \mathrm{orth}([\mathbf{T}(z_1)^{-1}\mathbf{w}_1 \;\; \mathbf{T}(z_2)^{-1}\mathbf{w}_2 \;\; \cdots \;\; \mathbf{T}(z_r)^{-1}\mathbf{w}_r] \in \mathbb{C}^{n\times r}.$$

approach one: structure preserving rational interpolation

Scenario: $\mathbf{T}(\lambda) \in \mathbb{C}^{n \times n}$ has *large dimension n*.

Goal: Reduce dimension of $\mathbf{T}(\lambda)$ *but maintain the nonlinear structure*.
Smaller problem will be more amenable to dense nonlinear eigensolvers.

Method: Rational tangential interpolation of $\mathbf{T}(\lambda)^{-1}$ at r points, directions.

Iteratively Corrected Rational Interpolation method

- Pick r *interpolation points* $\{z_j\}_{j=1}^r$ and *interpolation directions* $\{\mathbf{w}_j\}_{j=1}^r$.

- Construct a basis for projection (cf. *shift-invert Arnoldi*):

$$\mathbf{U} = \mathrm{orth}([\mathbf{T}(z_1)^{-1}\mathbf{w}_1 \;\; \mathbf{T}(z_2)^{-1}\mathbf{w}_2 \;\; \cdots \;\; \mathbf{T}(z_r)^{-1}\mathbf{w}_r] \in \mathbb{C}^{n \times r}.$$

- Form the reduced-dimension nonlinear system:

$$\mathbf{T}_r(\lambda) := \mathbf{U}^*\mathbf{T}(\lambda)\mathbf{U} \in \mathbb{C}^{r \times r}.$$

approach one: structure preserving rational interpolation

Scenario: $\mathbf{T}(\lambda) \in \mathbb{C}^{n\times n}$ has *large dimension n*.

Goal: Reduce dimension of $\mathbf{T}(\lambda)$ *but maintain the nonlinear structure*.
Smaller problem will be more amenable to dense nonlinear eigensolvers.

Method: Rational tangential interpolation of $\mathbf{T}(\lambda)^{-1}$ at r points, directions.

Iteratively Corrected Rational Interpolation method

- Pick r *interpolation points* $\{z_j\}_{j=1}^r$ and *interpolation directions* $\{\mathbf{w}_j\}_{j=1}^r$.
- Construct a basis for projection (cf. *shift-invert Arnoldi*):

$$\mathbf{U} = \mathrm{orth}([\mathbf{T}(z_1)^{-1}\mathbf{w}_1 \;\; \mathbf{T}(z_2)^{-1}\mathbf{w}_2 \;\; \cdots \;\; \mathbf{T}(z_r)^{-1}\mathbf{w}_r] \in \mathbb{C}^{n\times r}.$$

- Form the reduced-dimension nonlinear system:

$$\mathbf{T}_r(\lambda) := \mathbf{U}^*\mathbf{T}(\lambda)\mathbf{U} \in \mathbb{C}^{r\times r}.$$

- Compute the spectrum of $\mathbf{T}_r(\lambda)$ and use its eigenvalues and eigenvectors to update $\{z_j\}_{j=1}^r$ and $\{\mathbf{w}_j\}_{j=1}^r$, and repeat.

approach one: structure preserving rational interpolation

The choice of projection subspace $\mathrm{Ran}(\mathbf{U})$ delivers the key *interpolation property.*

Interpolation Theorem.
Provided $z_j \notin \sigma(\mathbf{T}) \cup \sigma(\mathbf{T}_r)$ for all $j = 1, \ldots, r$,

$$\mathbf{T}(z_j)^{-1}\mathbf{w}_j = \mathbf{U}\mathbf{T}_r(z_j)^{-1}\mathbf{U}^*\mathbf{w}_j.$$

Inspiration: model reduction for nonlinear systems w/coprime factorizations [Beattie & Gugercin 2009]; iteration like dominant pole algorithm [Martins, Lima, Pinto 1996]; [Roomes & Martins 2006], IRKA [Gugercin, Antoulas, Beattie 2008].

approach one: structure preserving rational interpolation

The choice of projection subspace Ran($\mathbf{U}$) delivers the key *interpolation property.*

Interpolation Theorem.
Provided $z_j \notin \sigma(\mathbf{T}) \cup \sigma(\mathbf{T}_r)$ for all $j = 1, \ldots, r$,

$$\mathbf{T}(z_j)^{-1}\mathbf{w}_j = \mathbf{U}\mathbf{T}_r(z_j)^{-1}\mathbf{U}^*\mathbf{w}_j.$$

Inspiration: model reduction for nonlinear systems w/coprime factorizations [Beattie & Gugercin 2009]; iteration like dominant pole algorithm [Martins, Lima, Pinto 1996]; [Roomes & Martins 2006], IRKA [Gugercin, Antoulas, Beattie 2008].

Illustration. As for all orthogonal projection methods:

$$\begin{aligned}
\mathbf{T}(\lambda) &= f_0(\lambda)\,\mathbf{A}_0 + f_1(\lambda)\,\mathbf{A}_1 + f_2(\lambda)\,\mathbf{A}_2 \\
\mathbf{T}_r(\lambda) &= f_0(\lambda)\,\mathbf{U}^*\mathbf{A}_0\mathbf{U} + f_1(\lambda)\,\mathbf{U}^*\mathbf{A}_1\mathbf{U} + f_2(\lambda)\,\mathbf{U}^*\mathbf{A}_2\mathbf{U}
\end{aligned}$$

approach one: structure preserving rational interpolation

The choice of projection subspace $\mathrm{Ran}(\mathbf{U})$ delivers the key *interpolation property.*

Interpolation Theorem.
Provided $z_j \notin \sigma(\mathbf{T}) \cup \sigma(\mathbf{T}_r)$ for all $j = 1, \ldots, r$,

$$\mathbf{T}(z_j)^{-1}\mathbf{w}_j = \mathbf{U}\mathbf{T}_r(z_j)^{-1}\mathbf{U}^*\mathbf{w}_j.$$

Inspiration: model reduction for nonlinear systems w/coprime factorizations [Beattie & Gugercin 2009]; iteration like dominant pole algorithm [Martins, Lima, Pinto 1996]; [Roomes & Martins 2006], IRKA [Gugercin, Antoulas, Beattie 2008].

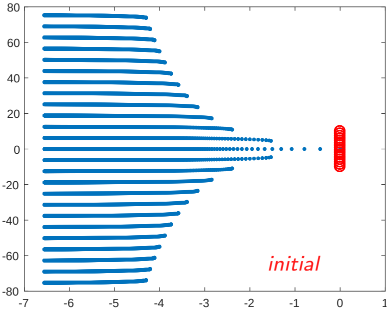
Illustration. As for all orthogonal projection methods:

$$\begin{aligned}
\mathbf{T}(\lambda) &= f_0(\lambda)\quad \mathbf{A}_0 \quad + f_1(\lambda)\quad \mathbf{A}_1 \quad + f_2(\lambda)\quad \mathbf{A}_2 \\
\mathbf{T}_r(\lambda) &= f_0(\lambda)\,\mathbf{U}^*\mathbf{A}_0\mathbf{U} + f_1(\lambda)\,\mathbf{U}^*\mathbf{A}_1\mathbf{U} + f_2(\lambda)\,\mathbf{U}^*\mathbf{A}_2\mathbf{U}
\end{aligned}$$

- The nonlinear functions f_j remain intact: *the structure is preserved*.
- The coefficients $\mathbf{A}_j \in \mathbb{C}^{n\times n}$ are compressed to $\mathbf{U}^*\mathbf{A}_j\mathbf{U} \in \mathbb{C}^{r\times r}$.
- Contrast: [Lietaert, Pérez, Vandereycken, Meerbergen 2018+] apply AAA approximation to $f_j(\lambda)$, leave coefficient matrices intact.

approach one: structure preserving rational interpolation

Example 1. $\mathbf{T}(\lambda) = \lambda\mathbf{I} - \mathbf{A} - e^{-\lambda}\mathbf{I}$,
where $\mathbf{A}$ is symmetric with $n = 1000$; eigenvalues of $\mathbf{A} = \{-1, -2, \ldots, -n\}$.



- Eigenvalues of full $\mathbf{T}(\lambda)$

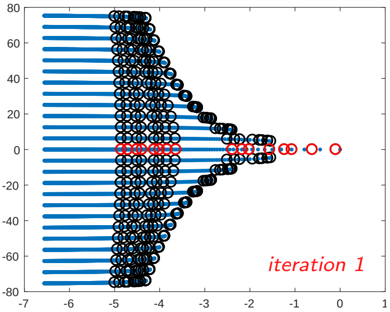
○ Interpolation points $\{z_j\}$

$r = 16$ used at each cycle (new points = real eigenvalues of $\mathbf{T}_r(\lambda)$)

initial $\{z_j\}$ uniformly distributed on $[-10\mathrm{i}, 10\mathrm{i}]$, $\{\mathbf{w}_j\}$ selected randomly

approach one: structure preserving rational interpolation

Example 1. $\mathbf{T}(\lambda) = \lambda\mathbf{I} - \mathbf{A} - e^{-\lambda}\mathbf{I}$,
where $\mathbf{A}$ is symmetric with $n = 1000$; eigenvalues of $\mathbf{A} = \{-1, -2, \ldots, -n\}$.



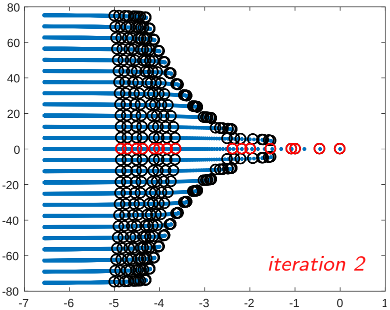
- $\cdot$ Eigenvalues of full $\mathbf{T}(\lambda)$
- $\circ$ Interpolation points $\{z_j\}$
- $\circ$ Eigenvalues of reduced $\mathbf{T}_r(\lambda)$

$r = 16$ used at each cycle (new points = real eigenvalues of $\mathbf{T}_r(\lambda)$)

initial $\{z_j\}$ uniformly distributed on $[-10\mathrm{i}, 10\mathrm{i}]$, $\{\mathbf{w}_j\}$ selected randomly

approach one: structure preserving rational interpolation

Example 1. $\mathbf{T}(\lambda) = \lambda\mathbf{I} - \mathbf{A} - e^{-\lambda}\mathbf{I}$,
where $\mathbf{A}$ is symmetric with $n = 1000$; eigenvalues of $\mathbf{A} = \{-1, -2, \ldots, -n\}$.



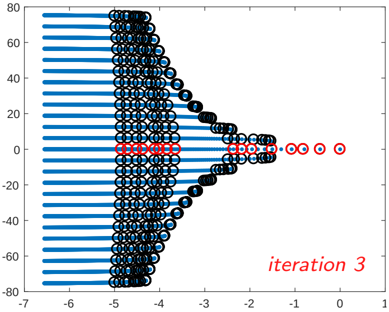
- Eigenvalues of full $\mathbf{T}(\lambda)$
- Interpolation points $\{z_j\}$
- Eigenvalues of reduced $\mathbf{T}_r(\lambda)$

$r = 16$ used at each cycle (new points = real eigenvalues of $\mathbf{T}_r(\lambda)$)

initial $\{z_j\}$ uniformly distributed on $[-10\mathrm{i}, 10\mathrm{i}]$, $\{\mathbf{w}_j\}$ selected randomly

approach one: structure preserving rational interpolation

Example 1. $\mathbf{T}(\lambda) = \lambda\mathbf{I} - \mathbf{A} - e^{-\lambda}\mathbf{I}$,
where $\mathbf{A}$ is symmetric with $n = 1000$; eigenvalues of $\mathbf{A} = \{-1, -2, \ldots, -n\}$.



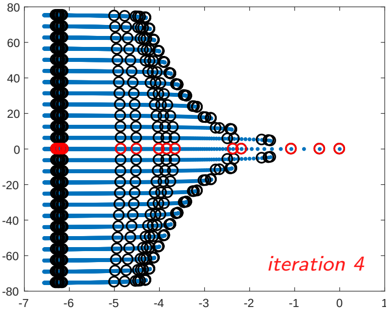
- Eigenvalues of full $\mathbf{T}(\lambda)$
- Interpolation points $\{z_j\}$
- Eigenvalues of reduced $\mathbf{T}_r(\lambda)$

$r = 16$ used at each cycle (new points = real eigenvalues of $\mathbf{T}_r(\lambda)$)

initial $\{z_j\}$ uniformly distributed on $[-10\mathrm{i}, 10\mathrm{i}]$, $\{\mathbf{w}_j\}$ selected randomly

approach one: structure preserving rational interpolation

Example 1. $\mathbf{T}(\lambda) = \lambda\mathbf{I} - \mathbf{A} - e^{-\lambda}\mathbf{I}$,
where $\mathbf{A}$ is symmetric with $n = 1000$; eigenvalues of $\mathbf{A} = \{-1, -2, \ldots, -n\}$.



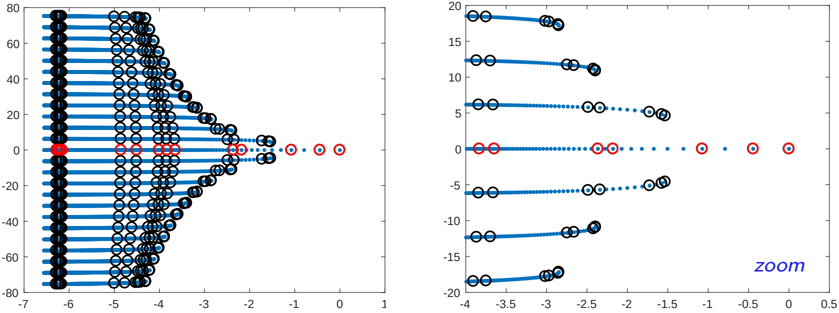
- Eigenvalues of full $\mathbf{T}(\lambda)$
- Interpolation points $\{z_j\}$
- Eigenvalues of reduced $\mathbf{T}_r(\lambda)$

$r = 16$ used at each cycle (new points = real eigenvalues of $\mathbf{T}_r(\lambda)$)

initial $\{z_j\}$ uniformly distributed on $[-10\mathrm{i}, 10\mathrm{i}]$, $\{\mathbf{w}_j\}$ selected randomly

approach one: structure preserving rational interpolation

Example 1. $\mathbf{T}(\lambda) = \lambda\mathbf{I} - \mathbf{A} - e^{-\lambda}\mathbf{I}$,
where $\mathbf{A}$ is symmetric with $n = 1000$; eigenvalues of $\mathbf{A} = \{-1, -2, \ldots, -n\}$.



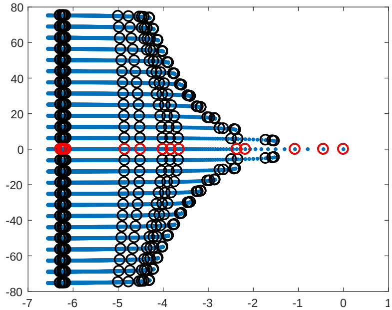
- Eigenvalues of full $\mathbf{T}(\lambda)$
- Eigenvalues of reduced $\mathbf{T}_r(\lambda)$ at the final cycle
- Final interpolation points $\{z_j\}$

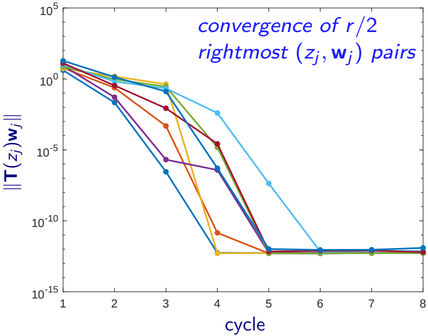
$r = 16$ used at each cycle (new points = real eigenvalues of $\mathbf{T}_r(\lambda)$)

initial $\{z_j\}$ uniformly distributed on $[-10i, 10i]$, $\{\mathbf{w}_j\}$ selected randomly

approach one: structure preserving rational interpolation

Example 1. $\mathbf{T}(\lambda) = \lambda\mathbf{I} - \mathbf{A} - e^{-\lambda}\mathbf{I}$,
where $\mathbf{A}$ is symmetric with $n = 1000$; eigenvalues of $\mathbf{A} = \{-1, -2, \ldots, -n\}$.





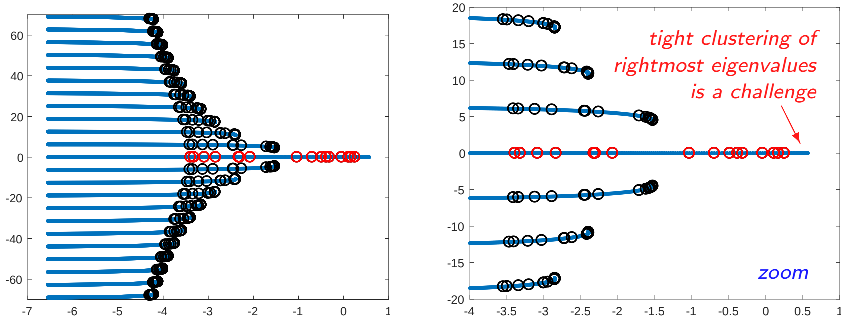
- • Eigenvalues of full $\mathbf{T}(\lambda)$
- ○ Eigenvalues of reduced $\mathbf{T}_r(\lambda)$ at the final cycle
- ○ Final interpolation points $\{z_j\}$

$r = 16$ used at each cycle (new points = real eigenvalues of $\mathbf{T}_r(\lambda)$)

initial $\{z_j\}$ uniformly distributed on $[-10\mathrm{i}, 10\mathrm{i}]$, $\{\mathbf{w}_j\}$ selected randomly

approach one: structure preserving rational interpolation

Example 2. $\mathbf{T}(\lambda) = \lambda\mathbf{I} - \mathbf{A} - e^{-\lambda}\mathbf{I}$,
$\mathbf{A}$ is symmetric with $n = 1000$; eigenvalues of $\mathbf{A} = \{-1/n, -4/n, \ldots, -n\}$.



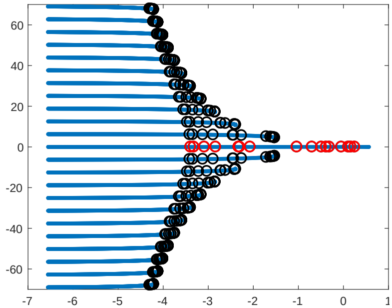
- Eigenvalues of full $\mathbf{T}(\lambda)$
- ○ Eigenvalues of reduced $\mathbf{T}_r(\lambda)$ at the final cycle
- ○ Final interpolation points $\{z_j\}$

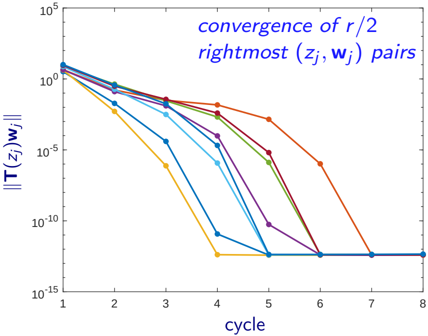
$r = 16$ used at each cycle (new points = real eigenvalues of $\mathbf{T}_r(\lambda)$)

initial $\{z_j\}$ uniformly distributed on $[-10i, 10i]$, $\{\mathbf{w}_j\}$ selected randomly

approach one: structure preserving rational interpolation

Example 2. $\mathbf{T}(\lambda) = \lambda\mathbf{I} - \mathbf{A} - \mathrm{e}^{-\lambda}\mathbf{I}$,
$\mathbf{A}$ is symmetric with $n = 1000$; eigenvalues of $\mathbf{A} = \{-1/n, -4/n, \ldots, -n\}$.





- • Eigenvalues of full $\mathbf{T}(\lambda)$
- ○ Eigenvalues of reduced $\mathbf{T}_r(\lambda)$ at the final cycle
- ○ Final interpolation points $\{z_j\}$

$r = 16$ used at each cycle (new points = real eigenvalues of $\mathbf{T}_r(\lambda)$)

initial $\{z_j\}$ uniformly distributed on $[-10\mathrm{i}, 10\mathrm{i}]$, $\{\mathbf{w}_j\}$ selected randomly

approach two: data-driven rational interpolation

Scenario: $\mathbf{T}(\lambda) \in \mathbb{C}^{n\times n}$ has *large dimension n*.

Goal: Obtain a small *linear matrix pencil* that *interpolates* the nonlinear eigenvalue problem. Smaller problem requires no further linearization.

Method: Data-driven rational interpolation of $\mathbf{T}(\lambda)^{-1}$.

Data-Driven Rational Interpolation Matrix Pencil method

approach two: data-driven rational interpolation

Scenario: $\mathbf{T}(\lambda) \in \mathbb{C}^{n\times n}$ has *large dimension n*.

Goal: Obtain a small *linear matrix pencil* that *interpolates* the nonlinear eigenvalue problem. Smaller problem requires no further linearization.

Method: Data-driven rational interpolation of $\mathbf{T}(\lambda)^{-1}$.

Data-Driven Rational Interpolation Matrix Pencil method

- Specify interpolation data:

left points, directions: $z_1, \ldots, z_r \in \mathbb{C}$, $\mathbf{w}_1, \ldots, \mathbf{w}_r \in \mathbb{C}^n$
right points, directions: $z_{r+1}, \ldots, z_{2r} \in \mathbb{C}$, $\mathbf{w}_{r+1}, \ldots, \mathbf{w}_{2r} \in \mathbb{C}^n$

approach two: data-driven rational interpolation

Scenario: $\mathbf{T}(\lambda) \in \mathbb{C}^{n\times n}$ has *large dimension n*.

Goal: Obtain a small *linear matrix pencil* that *interpolates* the nonlinear eigenvalue problem. Smaller problem requires no further linearization.

Method: Data-driven rational interpolation of $\mathbf{T}(\lambda)^{-1}$.

Data-Driven Rational Interpolation Matrix Pencil method

- Specify interpolation data:

  left points, directions: $z_1, \ldots, z_r \in \mathbb{C}$, $\mathbf{w}_1, \ldots, \mathbf{w}_r \in \mathbb{C}^n$
  right points, directions: $z_{r+1}, \ldots, z_{2r} \in \mathbb{C}$, $\mathbf{w}_{r+1}, \ldots, \mathbf{w}_{2r} \in \mathbb{C}^n$

- Construct $\mathbf{T}_r(\lambda)^{-1} := \mathbf{C}_r(\mathbf{A}_r - \lambda\mathbf{E}_r)^{-1}\mathbf{B}_r$ to *tangentially interpolate* $\mathbf{T}(\lambda)^{-1}$.

Tangential Interpolation Theorem. Provided $z_j \notin \sigma(\mathbf{T}) \cup \sigma(\mathbf{T}_r)$,

$$\mathbf{w}_j^T\mathbf{T}(z_j)^{-1} = \mathbf{w}_j^T\mathbf{T}_r(z_j)^{-1}, \qquad j = 1, \ldots, r;$$

$$\mathbf{T}(z_j)^{-1}\mathbf{w}_j = \mathbf{T}_r(z_j)^{-1}\mathbf{w}_j, \qquad j = r+1, \ldots, 2r.$$

approach two: data-driven rational interpolation

Given	*left points, directions:*	$z_1, \ldots, z_r \in \mathbb{C}$,	$\mathbf{w}_1, \ldots, \mathbf{w}_r \in \mathbb{C}^n$
	right points, directions:	$z_{r+1}, \ldots, z_{2r} \in \mathbb{C}$,	$\mathbf{w}_{r+1}, \ldots, \mathbf{w}_{2r} \in \mathbb{C}^n$
Define	*left interpolation data:*	$\mathbf{f}_1 = \mathbf{T}(z_1)^{-T}\mathbf{w}_1$, $\ldots$,	$\mathbf{f}_r = \mathbf{T}(z_r)^{-T}\mathbf{w}_r$
	right interpolation data:	$\mathbf{f}_{r+1} = \mathbf{T}(z_{r+1})^{-1}\mathbf{w}_{r+1}$, $\ldots$,	$\mathbf{f}_{2r} = \mathbf{T}(z_{2r})^{-1}\mathbf{w}_{2r}$

approach two: data-driven rational interpolation

Given	*left points, directions:*	$z_1, \ldots, z_r \in \mathbb{C}$,	$\mathbf{w}_1, \ldots, \mathbf{w}_r \in \mathbb{C}^n$
	right points, directions:	$z_{r+1}, \ldots, z_{2r} \in \mathbb{C}$,	$\mathbf{w}_{r+1}, \ldots, \mathbf{w}_{2r} \in \mathbb{C}^n$
Define	*left interpolation data:*	$\mathbf{f}_1 = \mathbf{T}(z_1)^{-T}\mathbf{w}_1$,	$\ldots,\quad \mathbf{f}_r = \mathbf{T}(z_r)^{-T}\mathbf{w}_r$
	right interpolation data:	$\mathbf{f}_{r+1} = \mathbf{T}(z_{r+1})^{-1}\mathbf{w}_{r+1}$,	$\ldots,\quad \mathbf{f}_{2r} = \mathbf{T}(z_{2r})^{-1}\mathbf{w}_{2r}$

Order-r (linear) model: $\mathbf{T}_r(z)^{-1} = \mathbf{C}_r(\mathbf{A}_r - z\mathbf{E}_r)^{-1}\mathbf{B}_r$

approach two: data-driven rational interpolation

Given	*left points, directions:*	$z_1, \ldots, z_r \in \mathbb{C}$,	$\mathbf{w}_1, \ldots, \mathbf{w}_r \in \mathbb{C}^n$
	right points, directions:	$z_{r+1}, \ldots, z_{2r} \in \mathbb{C}$,	$\mathbf{w}_{r+1}, \ldots, \mathbf{w}_{2r} \in \mathbb{C}^n$
Define	*left interpolation data:*	$\mathbf{f}_1 = \mathbf{T}(z_1)^{-T}\mathbf{w}_1, \quad \ldots, \quad \mathbf{f}_r = \mathbf{T}(z_r)^{-T}\mathbf{w}_r$	
	right interpolation data:	$\mathbf{f}_{r+1} = \mathbf{T}(z_{r+1})^{-1}\mathbf{w}_{r+1}, \quad \ldots, \quad \mathbf{f}_{2r} = \mathbf{T}(z_{2r})^{-1}\mathbf{w}_{2r}$	

Order-r (linear) model: $\mathbf{T}_r(z)^{-1} = \mathbf{C}_r(\mathbf{A}_r - z\mathbf{E}_r)^{-1}\mathbf{B}_r$

$$\mathbf{C}_r = [\, \mathbf{f}_{r+1}, \ldots, \mathbf{f}_{2r} \,]$$

$$\mathbf{A}_r = \begin{bmatrix} \dfrac{z_1\mathbf{f}_1^T\mathbf{w}_{r+1} - z_{r+1}\mathbf{w}_1^T\mathbf{f}_{r+1}}{z_1 - z_{r+1}} & \cdots & \dfrac{z_r\mathbf{f}_r^T\mathbf{w}_{r+1} - z_{r+1}\mathbf{w}_r^T\mathbf{f}_{r+1}}{z_r - z_{r+1}} \\ \vdots & \ddots & \vdots \\ \dfrac{z_1\mathbf{f}_1^T\mathbf{w}_{2r} - z_{2r}\mathbf{w}_1^T\mathbf{f}_{2r}}{z_1 - z_{2r}} & \cdots & \dfrac{z_r\mathbf{f}_r^T\mathbf{w}_{2r} - z_{2r}\mathbf{w}_r^T\mathbf{f}_{2r}}{z_r - z_{2r}} \end{bmatrix}$$

shifted Loewner

$$\mathbf{E}_r = \begin{bmatrix} \dfrac{\mathbf{f}_1^T\mathbf{w}_{r+1} - \mathbf{w}_1^T\mathbf{f}_{r+1}}{z_1 - z_{r+1}} & \cdots & \dfrac{\mathbf{f}_r^T\mathbf{w}_{r+1} - \mathbf{w}_r^T\mathbf{f}_{r+1}}{z_r - z_{r+1}} \\ \vdots & \ddots & \vdots \\ \dfrac{\mathbf{f}_1^T\mathbf{w}_{2r} - \mathbf{w}_1^T\mathbf{f}_{2r}}{z_1 - z_{2r}} & \cdots & \dfrac{\mathbf{f}_r^T\mathbf{w}_{2r} - \mathbf{w}_r^T\mathbf{f}_{2r}}{z_r - z_{2r}} \end{bmatrix}$$

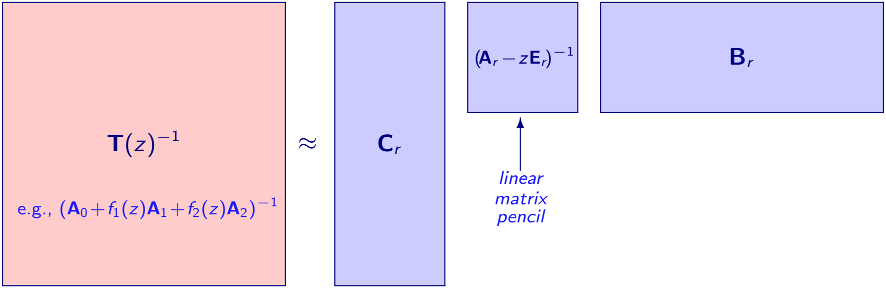
Loewner

$$\mathbf{B}_r = [\, \mathbf{f}_1, \ldots, \mathbf{f}_r \,]^T$$

approach two: data-driven rational interpolation

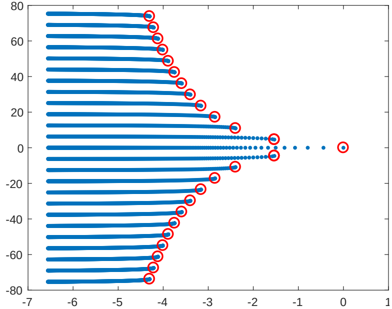
Given	*left points, directions:*	$z_1, \ldots, z_r \in \mathbb{C}$,	$\mathbf{w}_1, \ldots, \mathbf{w}_r \in \mathbb{C}^n$
	right points, directions:	$z_{r+1}, \ldots, z_{2r} \in \mathbb{C}$,	$\mathbf{w}_{r+1}, \ldots, \mathbf{w}_{2r} \in \mathbb{C}^n$
Define	*left interpolation data:*	$\mathbf{f}_1 = \mathbf{T}(z_1)^{-T}\mathbf{w}_1$, $\ldots$,	$\mathbf{f}_r = \mathbf{T}(z_r)^{-T}\mathbf{w}_r$
	right interpolation data:	$\mathbf{f}_{r+1} = \mathbf{T}(z_{r+1})^{-1}\mathbf{w}_{r+1}$, $\ldots$,	$\mathbf{f}_{2r} = \mathbf{T}(z_{2r})^{-1}\mathbf{w}_{2r}$

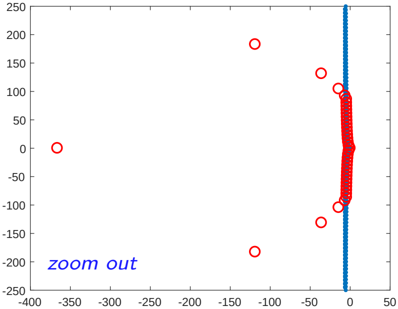
Rank-r (linear) model: $\mathbf{T}_r(z)^{-1} = \mathbf{C}_r(\mathbf{A}_r - z\mathbf{E}_r)^{-1}\mathbf{B}_r$



approach two: data-driven rational interpolation

Example. $\mathbf{T}(\lambda) = \lambda\mathbf{I} - \mathbf{A} - \mathrm{e}^{-\lambda}\mathbf{I}$,
where $\mathbf{A}$ is symmetric with $n = 1000$; eigenvalues of $\mathbf{A} = \{-1, -2, \ldots, -n\}$.





- Eigenvalues of full $\mathbf{T}(\lambda)$
- Eigenvalues of reduced matrix pencil $\mathbf{A}_r - z\mathbf{E}_r$

  $r = 40$ interpolation points used, uniform in interval $[-80\mathrm{i}, 80\mathrm{i}]$

  Hermite interpolation variant that only uses r distinct interpolation points.

  interpolation directions from smallest singular values of $\mathbf{T}(z_j)$.

approach three: Loewner realization via contour integration

Scenario: Seek all eigenvalues of $\mathbf{T}(\lambda) \in \mathbb{C}^{n\times n}$ *in a prescribed region* Ω *of* $\mathbb{C}$.

Goal: Use Keldysh's Theorem to isolate interesting part of $\mathbf{T}(\lambda)$ in Ω.

Method: Contour integration of $\mathbf{T}(\lambda)$ against *rational test functions*.
Loewner matrix will reveal number of eigenvalues in Ω.

Theorem [Keldysh 1951]. Suppose $\mathbf{T}(z)$ has m eigenvalues $\lambda_1, \ldots, \lambda_m$ (counting multiplicity) in the region $\Omega \subset \mathbb{C}$, all semi-simple. Then

$$\mathbf{T}(z)^{-1} = \mathbf{V}(z\mathbf{I} - \mathbf{\Lambda})^{-1}\mathbf{U}^* + \mathbf{R}(z),$$

- $\mathbf{V} = [\mathbf{v}_1 \ \cdots \ \mathbf{v}_m]$, $\mathbf{U} = [\mathbf{u}_1 \ \cdots \ \mathbf{u}_m]$, $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_m)$, $\mathbf{u}_j^* \mathbf{T}'(\lambda_j)\mathbf{v}_j = 1$;
- $\mathbf{R}(z)$ is analytic in Ω.

approach three: Loewner realization via contour integration

Scenario: Seek all eigenvalues of $\mathbf{T}(\lambda) \in \mathbb{C}^{n\times n}$ *in a prescribed region* Ω *of* $\mathbb{C}$.

Goal: Use Keldysh's Theorem to isolate interesting part of $\mathbf{T}(\lambda)$ in Ω.

Method: Contour integration of $\mathbf{T}(\lambda)$ against *rational test functions*.
Loewner matrix will reveal number of eigenvalues in Ω.

Theorem [Keldysh 1951]. Suppose $\mathbf{T}(z)$ has m eigenvalues $\lambda_1, \ldots, \lambda_m$ (counting multiplicity) in the region $\Omega \subset \mathbb{C}$, all semi-simple. Then

$$\mathbf{T}(z)^{-1} = \mathbf{V}(z\mathbf{I} - \mathbf{\Lambda})^{-1}\mathbf{U}^* + \mathbf{R}(z),$$

- $\mathbf{V} = [\mathbf{v}_1 \ \cdots \ \mathbf{v}_m]$, $\mathbf{U} = [\mathbf{u}_1 \ \cdots \ \mathbf{u}_m]$, $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_m)$, $\mathbf{u}_j^* \mathbf{T}'(\lambda_j)\mathbf{v}_j = 1$;
- $\mathbf{R}(z)$ is analytic in Ω.

$$\mathbf{T}(z)^{-1} = \mathbf{H}(z) + \mathbf{R}(z)$$

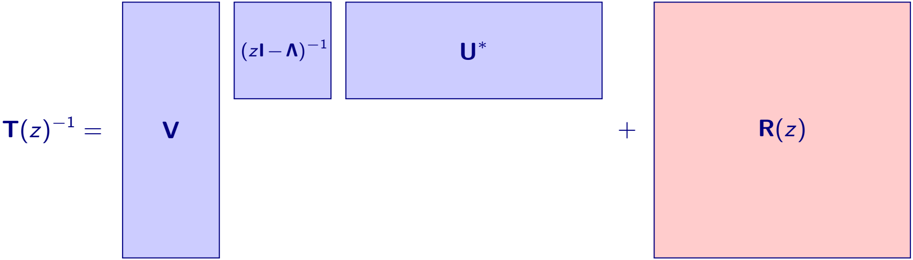
where $\mathbf{H}(z) := \mathbf{V}(z\mathbf{I} - \mathbf{\Lambda})^{-1}\mathbf{U}^*$ is a *transfer function for a linear system*.

approach three: Loewner approximation via contour integration

Theorem [Keldysh 1951]. Suppose $\mathbf{T}(z)$ has m eigenvalues $\lambda_1, \ldots, \lambda_m$ (counting multiplicity) in the region $\Omega \subset \mathbb{C}$, all semi-simple. Then

$$\mathbf{T}(z)^{-1} = \mathbf{V}(z\mathbf{I} - \mathbf{\Lambda})^{-1}\mathbf{U}^* + \mathbf{R}(z),$$

- $\mathbf{V} = [\mathbf{v}_1 \ \cdots \ \mathbf{v}_m]$, $\mathbf{U} = [\mathbf{U}_1 \ \cdots \ \mathbf{U}_m]$, $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_m)$, $\mathbf{U}_j^* \mathbf{T}'(\lambda_j)\mathbf{v}_j = 1$;
- $\mathbf{R}(z)$ is analytic in Ω.



$\mathbf{H}(z) := \mathbf{V}(z\mathbf{I} - \mathbf{\Lambda})^{-1}\mathbf{U}^*$
$n \times n$ linear system, order m:
m poles in Ω

nonlinear system,
but nice in Ω

approach three: Loewner realization via contour integration

$$\mathbf{T}(z)^{-1} = \mathbf{H}(z) + \mathbf{R}(z)$$

where $\mathbf{H}(z) : \mathbf{V}(z\mathbf{I} - \boldsymbol{\Lambda})^{-1}\mathbf{U}^*$ is a *transfer function for a linear system*.

A family of algorithms use the fact that, by the Cauchy integral formula,

$$\frac{1}{2\pi \mathrm{i}} \int_{\partial\Omega} f(z)\mathbf{T}(z)^{-1}\,\mathrm{d}z = \mathbf{V}f(\boldsymbol{\Lambda})\mathbf{U}^*;$$

see [Asakura, Sakurai, Tadano, Ikegami, Kimura 2009], [Beyn 2012], [Yokota & Sakurai 2013], etc., building upon contour integral eigensolvers for matrix pencils [Sakurai & Sugiura 2003], [Polizzi 2009], etc.

These algorithms use $f(z) = z^k$ for $k = 0, 1, \ldots$ to produce Hankel matrix pencils.

approach three: Loewner realization via contour integration

$$\mathbf{T}(z)^{-1} = \mathbf{H}(z) + \mathbf{R}(z)$$

where $\mathbf{H}(z) : \mathbf{V}(z\mathbf{I} - \mathbf{\Lambda})^{-1}\mathbf{U}^*$ is a *transfer function for a linear system*.

A family of algorithms use the fact that, by the Cauchy integral formula,

$$\frac{1}{2\pi \mathrm{i}} \int_{\partial\Omega} f(z)\mathbf{T}(z)^{-1}\,\mathrm{d}z = \mathbf{V}f(\mathbf{\Lambda})\mathbf{U}^*;$$

see [Asakura, Sakurai, Tadano, Ikegami, Kimura 2009], [Beyn 2012], [Yokota & Sakurai 2013], etc., building upon contour integral eigensolvers for matrix pencils [Sakurai & Sugiura 2003], [Polizzi 2009], etc.

These algorithms use $f(z) = z^k$ for $k = 0, 1, \ldots$ to produce Hankel matrix pencils.

Key observation: If we use $f(z) = 1/(z_j - z)$ for z_j exterior to Ω, we obtain

$$\frac{1}{2\pi \mathrm{i}} \int_{\partial\Omega} \frac{1}{z_j - z}\mathbf{T}(z)^{-1}\,\mathrm{d}z = \mathbf{V}(z_j\mathbf{I} - \mathbf{\Lambda})^{-1}\mathbf{U}^* = \mathbf{H}(z_j).$$

Contour integrals yield measurements of the linear system with the desired eigenvalues.

approach three: Loewner realization via contour integration

Minimal Realization via Rational Contour Integrals for m eigenvalues

- Let $r \geq m$, and select interpolation points and directions:

left points, directions:	$z_1, \ldots, z_r \in \mathbb{C} \setminus \Omega,$	$\mathbf{w}_1, \ldots, \mathbf{w}_r \in \mathbb{C}^n$
right points, directions:	$z_{r+1}, \ldots, z_{2r} \in \mathbb{C} \setminus \Omega,$	$\mathbf{w}_{r+1}, \ldots, \mathbf{w}_{2r} \in \mathbb{C}^n$

approach three: Loewner realization via contour integration

Minimal Realization via Rational Contour Integrals for m eigenvalues

- Let $r \geq m$, and select interpolation points and directions:

  left points, directions: $z_1, \ldots, z_r \in \mathbb{C} \setminus \Omega$, $\quad \mathbf{w}_1, \ldots, \mathbf{w}_r \in \mathbb{C}^n$

  right points, directions: $z_{r+1}, \ldots, z_{2r} \in \mathbb{C} \setminus \Omega$, $\quad \mathbf{w}_{r+1}, \ldots, \mathbf{w}_{2r} \in \mathbb{C}^n$

- Use contour integrals to compute the left and right interpolation data:

  left interpolation data: $\mathbf{f}_1 = \mathbf{H}(z_1)^T \mathbf{w}_1, \quad \ldots, \quad \mathbf{f}_r = \mathbf{H}(z_r)^T \mathbf{w}_r$

  right interpolation data: $\mathbf{f}_{r+1} = \mathbf{H}(z_{r+1}) \mathbf{w}_{r+1}, \quad \ldots, \quad \mathbf{f}_{2r} = \mathbf{H}(z_{2r}) \mathbf{w}_{2r}$

$$\mathbf{H}(z_j)\mathbf{w}_j = \frac{1}{2\pi \mathrm{i}} \int_{\partial\Omega} \frac{1}{z_j - z} \mathbf{T}(z)^{-1} \mathbf{w}_j \, \mathrm{d}z.$$

approach three: Loewner realization via contour integration

Minimal Realization via Rational Contour Integrals for m eigenvalues

- Let $r \geq m$, and select interpolation points and directions:

  left points, directions: $z_1, \ldots, z_r \in \mathbb{C} \setminus \Omega$, $\mathbf{w}_1, \ldots, \mathbf{w}_r \in \mathbb{C}^n$

  right points, directions: $z_{r+1}, \ldots, z_{2r} \in \mathbb{C} \setminus \Omega$, $\mathbf{w}_{r+1}, \ldots, \mathbf{w}_{2r} \in \mathbb{C}^n$

- Use contour integrals to compute the left and right interpolation data:

  left interpolation data: $\mathbf{f}_1 = \mathbf{H}(z_1)^T \mathbf{w}_1, \quad \ldots, \quad \mathbf{f}_r = \mathbf{H}(z_r)^T \mathbf{w}_r$

  right interpolation data: $\mathbf{f}_{r+1} = \mathbf{H}(z_{r+1}) \mathbf{w}_{r+1}, \quad \ldots, \quad \mathbf{f}_{2r} = \mathbf{H}(z_{2r}) \mathbf{w}_{2r}$

$$\mathbf{H}(z_j)\mathbf{w}_j = \frac{1}{2\pi \mathrm{i}} \int_{\partial\Omega} \frac{1}{z_j - z} \mathbf{T}(z)^{-1} \mathbf{w}_j \, dz.$$

- Construct *Loewner* and *shifted Loewner* matrices from this data, just as in the Data-Driven Rational Interpolation method:

$$\mathbf{C}_r = [\, \mathbf{f}_{r+1}, \ldots, \mathbf{f}_{2r} \,] \qquad \mathbf{B}_r = [\, \mathbf{f}_1, \ldots, \mathbf{f}_r \,]^T$$

$$\mathbf{A}_r = \text{shifted Loewner matrix} \qquad \mathbf{E}_r = \text{Loewner matrix}$$

approach three: Loewner realization via contour integration

Minimal Realization via Rational Contour Integrals for m eigenvalues

- Let $r \geq m$, and select interpolation points and directions:

  left points, directions: $z_1, \ldots, z_r \in \mathbb{C} \setminus \Omega$, $\quad \mathbf{w}_1, \ldots, \mathbf{w}_r \in \mathbb{C}^n$

  right points, directions: $z_{r+1}, \ldots, z_{2r} \in \mathbb{C} \setminus \Omega$, $\quad \mathbf{w}_{r+1}, \ldots, \mathbf{w}_{2r} \in \mathbb{C}^n$

- Use contour integrals to compute the left and right interpolation data:

  left interpolation data: $\mathbf{f}_1 = \mathbf{H}(z_1)^T\mathbf{w}_1, \quad \ldots, \quad \mathbf{f}_r = \mathbf{H}(z_r)^T\mathbf{w}_r$

  right interpolation data: $\mathbf{f}_{r+1} = \mathbf{H}(z_{r+1})\mathbf{w}_{r+1}, \quad \ldots, \quad \mathbf{f}_{2r} = \mathbf{H}(z_{2r})\mathbf{w}_{2r}$

$$\mathbf{H}(z_j)\mathbf{w}_j = \frac{1}{2\pi \mathrm{i}} \int_{\partial\Omega} \frac{1}{z_j - z} \mathbf{T}(z)^{-1}\mathbf{w}_j \, \mathrm{d}z.$$

- Construct *Loewner* and *shifted Loewner* matrices from this data, just as in the Data-Driven Rational Interpolation method:

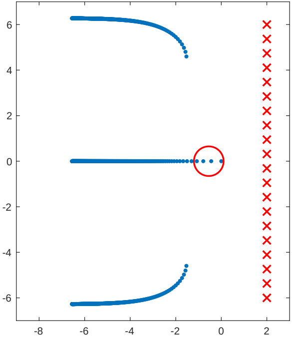
$$\mathbf{C}_r = [\,\mathbf{f}_{r+1}, \ldots, \mathbf{f}_{2r}\,] \qquad \mathbf{B}_r = [\,\mathbf{f}_1, \ldots, \mathbf{f}_r\,]^T$$

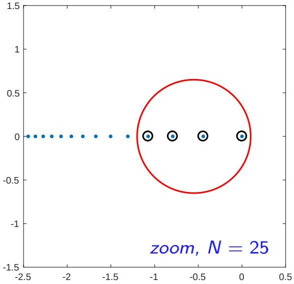
$$\mathbf{A}_r = \text{shifted Loewner matrix} \qquad \mathbf{E}_r = \text{Loewner matrix}$$

- If $r = m$, then $\mathbf{V}(z\mathbf{I} - \boldsymbol{\Lambda})^{-1}\mathbf{U}^* = \mathbf{C}_r(\mathbf{A}_r - z\mathbf{E}_r)^{-1}\mathbf{B}_r$: compute eigenvalues!
  If $r > m$, use SVD truncation / minimum realization techniques to reduce dimension; cf. [Mayo & Antoulas 2007].

approach three: Loewner realization via contour integration

Example. $\mathbf{T}(\lambda) = \lambda\mathbf{I} - \mathbf{A} - e^{-\lambda}\mathbf{I}$,
where $\mathbf{A}$ is symmetric with $n = 1000$; eigenvalues of $\mathbf{A} = \{-1, -2, \ldots, -n\}$.

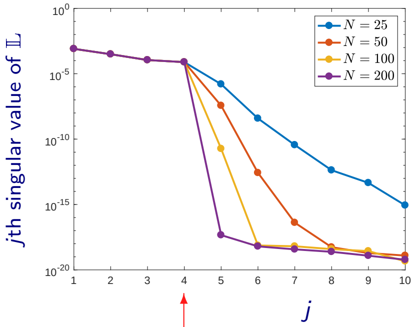




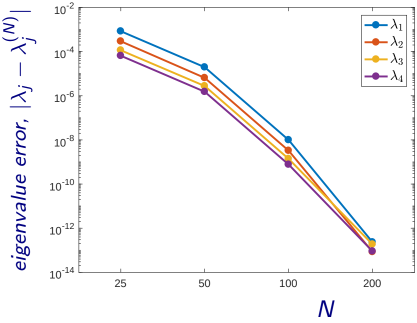
- $\cdot$ Eigenvalues of full $\mathbf{T}(\lambda)$
- $\times$ 20 interpolation points in $2 + [-6i, 6i]$
- $\circ$ Eigenvalues of minimal $(m = 4)$ matrix pencil
- — Contour of integration (circle)

  Trapezoid rule uses $N = 25$, 50, 100, and 200 interpolation points

Example. $\mathbf{T}(\lambda) = \lambda\mathbf{I} - \mathbf{A} - e^{-\lambda}\mathbf{I}$,
where $\mathbf{A}$ is symmetric with $n = 1000$; eigenvalues of $\mathbf{A} = \{-1, -2, \ldots, -n\}$.



4 eigenvalues in Ω
$\Rightarrow$ rank$(\mathbb{L}) = 4$



Cf. [Beyn 2012], [Güttel & Tisseur 2017] for $f(z) = z^k$.
For rank detection for Loewner matrices, see [Hokanson 2018+].

Transient Dynamics

for

Dynamical Systems

with Delays

a case study of pseudospectral analysis

introduction to transient dynamics

We often care about *eigenvalues*
because we seek insight about *dynamics*.

introduction to transient dynamics

We often care about *eigenvalues*
because we seek insight about *dynamics*.

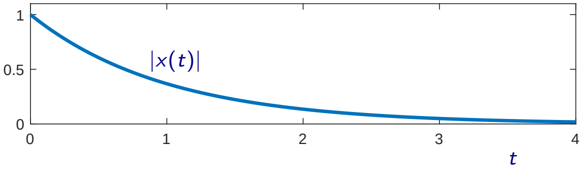
Start with the simple *scalar system*

$$x'(t) = \alpha x(t),$$

with solution

$$x(t) = \mathrm{e}^{t\alpha} x(0).$$

$\boxed{\text{If } \operatorname{Re}\alpha < 0, \text{ then } |x(t)| \to 0 \text{ monotonically as } t \to \infty.}$



introduction to transient dynamics

We often care about *eigenvalues*
because we seek insight about *dynamics*.

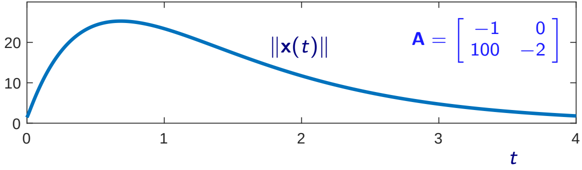
Now consider the n-dimensional system

$$\mathbf{x}'(t) = \mathbf{A}\mathbf{x}(t)$$

with solution

$$\mathbf{x}(t) = \mathrm{e}^{t\mathbf{A}}\mathbf{x}(0).$$

If $\operatorname{Re}\lambda < 0$ for all $\lambda \in \sigma(\mathbf{A})$, then $\|\mathbf{x}(t)\| \to 0$ asymptotically as $t \to \infty$,



introduction to transient dynamics

We often care about *eigenvalues*
because we seek insight about *dynamics*.

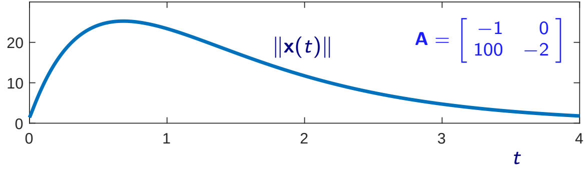
Now consider the n-dimensional system

$$\mathbf{x}'(t) = \mathbf{A}\mathbf{x}(t)$$

with solution

$$\mathbf{x}(t) = \mathrm{e}^{t\mathbf{A}}\mathbf{x}(0).$$

If $\operatorname{Re}\lambda < 0$ for all $\lambda \in \sigma(\mathbf{A})$, then $\|\mathbf{x}(t)\| \to 0$ asymptotically as $t \to \infty$, but it is possible that $\|\mathbf{x}(t_*)\| \gg \|\mathbf{x}(0)\|$ for some $t_* \in (0, \infty)$.



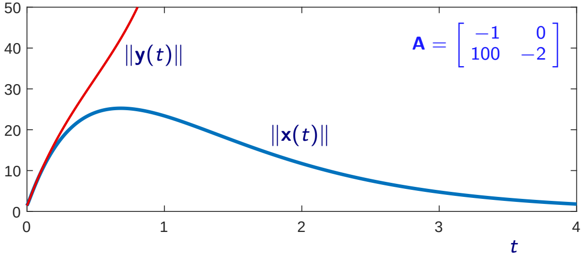
why transients matter

- Often the *linear* dynamical system $\mathbf{x}'(t) = \mathbf{A}\mathbf{x}(t)$ arises from *linear stability analysis for a fixed point of a nonlinear system*

$$\mathbf{y}'(t) = \mathbf{F}(\mathbf{y}(t), t).$$

For example,

$$\mathbf{y}'(t) = \mathbf{A}\mathbf{y}(t) + \tfrac{1}{20}\mathbf{y}(t)^2.$$



- In this example, *linear transient growth feeds the nonlinearity*. Such behavior can provide a mechanism for *transition to turbulence* in fluid flows; see, e.g., [Butler & Farrell 1992], [Trefethen et al. 1993].

detecting the potential for transient growth

One can draw insight about transient growth from the numerical range (field of values) and ε-pseudospectra of $\mathbf{A}$:

$$\begin{aligned}\sigma_\varepsilon(\mathbf{A}) &= \{z \in \mathbb{C} : \|(z\mathbf{I} - \mathbf{A})^{-1}\| > 1/\varepsilon\} \\ &= \{z \in \mathbb{C} : z \in \sigma(\mathbf{A} + \mathbf{E}) \text{ for some } \mathbf{E} \in \mathbb{C}^{n\times n} \text{ with } \|\mathbf{E}\| < \varepsilon\}\end{aligned}$$

For upper and lower bounds on $\|\mathbf{x}(t)\|$, see [Trefethen & E. 2005], e.g.,

$$\sup_{t\geq 0} \|e^{t\mathbf{A}}\| \geq \sup_{z\in\sigma_\varepsilon(\mathbf{A})} \frac{\operatorname{Re} z}{\varepsilon}.$$

If $\sigma_\varepsilon(\mathbf{A})$ extends more than ε across the imaginary axis, $\|e^{t\mathbf{A}}\|$ grows transiently.

detecting the potential for transient growth

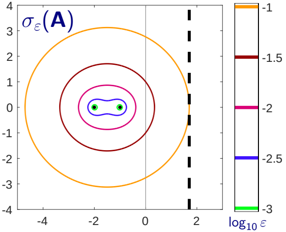
One can draw insight about transient growth from the numerical range (field of values) and ε-pseudospectra of $\mathbf{A}$:

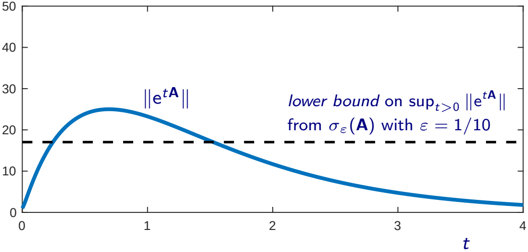
$$\begin{aligned}\sigma_\varepsilon(\mathbf{A}) &= \{z \in \mathbb{C} : \|(z\mathbf{I}-\mathbf{A})^{-1}\| > 1/\varepsilon\} \\ &= \{z \in \mathbb{C} : z \in \sigma(\mathbf{A}+\mathbf{E}) \text{ for some } \mathbf{E} \in \mathbb{C}^{n\times n} \text{ with } \|\mathbf{E}\| < \varepsilon\}\end{aligned}$$

For upper and lower bounds on $\|\mathbf{x}(t)\|$, see [Trefethen & E. 2005], e.g.,

$$\sup_{t\ge 0} \|\mathrm{e}^{t\mathbf{A}}\| \ge \sup_{z\in\sigma_\varepsilon(\mathbf{A})} \frac{\operatorname{Re} z}{\varepsilon}.$$

If $\sigma_\varepsilon(\mathbf{A})$ extends more than ε across the imaginary axis, $\|\mathrm{e}^{t\mathbf{A}}\|$ grows transiently.





Pseudospectra can guarantee that some $\mathbf{x}(0)$ induce transient growth.

two ways to look at pseudospectra

Two *equivalent* definitions give two distinct perspectives.

perturbed eigenvalues

$$\sigma_\varepsilon(\mathbf{A}) = \{z \in \mathbb{C} : z \in \sigma(\mathbf{A}+\mathbf{E}) \text{ for some } \mathbf{E} \in \mathbb{C}^{n\times n} \text{ with } \|\mathbf{E}\| < \varepsilon\}$$

norms of resolvents

$$\sigma_\varepsilon(\mathbf{A}) = \{z \in \mathbb{C} : \|(z\mathbf{I}-\mathbf{A})^{-1}\| > 1/\varepsilon\}$$

two ways to look at pseudospectra

Two *equivalent* definitions give two distinct perspectives.

perturbed eigenvalues

$$\sigma_\varepsilon(\mathbf{A}) = \{z \in \mathbb{C} : z \in \sigma(\mathbf{A}+\mathbf{E}) \text{ for some } \mathbf{E} \in \mathbb{C}^{n\times n} \text{ with } \|\mathbf{E}\| < \varepsilon\}$$

- $\sigma_\varepsilon(\mathbf{A})$ contains the eigenvalues of all matrices with distance ε of **A**.
- Ideal for assessing *asymptotic stability of uncertain systems:*

  Is some matrix *near* **A** unstable?
- Why consider all $\mathbf{E} \in \mathbb{C}^{n\times n}$? *Structured pseudospectra* further restrict **E** (real, Toeplitz, etc.).
  [Hinrichsen & Pritchard], [Karow], [Rump]

norms of resolvents

$$\sigma_\varepsilon(\mathbf{A}) = \{z \in \mathbb{C} : \|(z\mathbf{I}-\mathbf{A})^{-1}\| > 1/\varepsilon\}$$

two ways to look at pseudospectra

Two *equivalent* definitions give two distinct perspectives.

perturbed eigenvalues

$$\sigma_\varepsilon(\mathbf{A}) = \{z \in \mathbb{C} : z \in \sigma(\mathbf{A}+\mathbf{E}) \text{ for some } \mathbf{E} \in \mathbb{C}^{n\times n} \text{ with } \|\mathbf{E}\| < \varepsilon\}$$

- $\sigma_\varepsilon(\mathbf{A})$ contains the eigenvalues of all matrices with distance ε of $\mathbf{A}$.
- Ideal for assessing *asymptotic stability of uncertain systems:*

  Is some matrix *near* $\mathbf{A}$ unstable?
- Why consider all $\mathbf{E} \in \mathbb{C}^{n\times n}$? *Structured pseudospectra* further restrict $\mathbf{E}$ (real, Toeplitz, etc.).

  [Hinrichsen & Pritchard], [Karow], [Rump]

norms of resolvents

$$\sigma_\varepsilon(\mathbf{A}) = \{z \in \mathbb{C} : \|(z\mathbf{I}-\mathbf{A})^{-1}\| > 1/\varepsilon\}$$

- $\sigma_\varepsilon(\mathbf{A})$ is bounded by $1/\varepsilon$ level sets of the resolvent norm.
- Ideal for assessing *transient behavior of stable systems:*

  $\|e^{t\mathbf{A}}\| > 1$ or $\|\mathbf{A}^k\| > 1$?
- Rooted in semigroup theory: based on the solution operator for the dynamical system; structure of $\mathbf{A}$ plays no role.

two ways to look at pseudospectra

Two *equivalent* definitions give two distinct perspectives.

perturbed eigenvalues

$$\sigma_\varepsilon(\mathbf{A}) = \{z \in \mathbb{C} : z \in \sigma(\mathbf{A} + \mathbf{E}) \text{ for some } \mathbf{E} \in \mathbb{C}^{n\times n} \text{ with } \|\mathbf{E}\| < \varepsilon\}$$

- $\sigma_\varepsilon(\mathbf{A})$ contains the eigenvalues of all matrices with distance ε of $\mathbf{A}$.
- Ideal for assessing *asymptotic stability of uncertain systems:*

  Is some matrix *near* $\mathbf{A}$ unstable?
- Why consider all $\mathbf{E} \in \mathbb{C}^{n\times n}$? *Structured pseudospectra* further restrict $\mathbf{E}$ (real, Toeplitz, etc.).

  [Hinrichsen & Pritchard], [Karow], [Rump]

norms of resolvents

$$\sigma_\varepsilon(\mathbf{A}) = \{z \in \mathbb{C} : \|(z\mathbf{I} - \mathbf{A})^{-1}\| > 1/\varepsilon\}$$

- $\sigma_\varepsilon(\mathbf{A})$ is bounded by $1/\varepsilon$ level sets of the resolvent norm.
- Ideal for assessing *transient behavior of stable systems:*

  $\|e^{t\mathbf{A}}\| > 1$ or $\|\mathbf{A}^k\| > 1$?
- Rooted in semigroup theory: based on the solution operator for the dynamical system; structure of $\mathbf{A}$ plays no role.

These perspective match for $\mathbf{x}'(t) = \mathbf{A}\mathbf{x}(t)$, but not for more complicated systems.

scalar delay equations and the nonlinear eigenvalue problem

We shall apply these ideas to explore the potential for

transient growth in solutions to stable delay differential equations.

Solutions of scalar systems $x'(t) = \alpha x(t)$ behave monotonically: $|x(t)| = e^{t \operatorname{Re} \alpha} |x(0)|$. What about scalar delay equations?

$$\boxed{x'(t) = \alpha x(t) + \beta x(t-1)}$$

Using the techniques seen earlier, we associate this system with the NLEVP

$$\boxed{(\lambda - \alpha) e^{\lambda} = \beta,}$$

with infinitely many eigenvalues given by branches of the Lambert-W function:

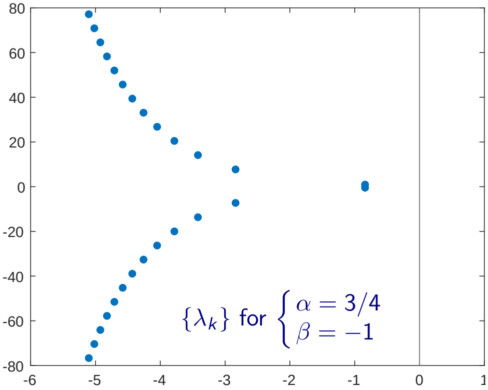
$$\boxed{\lambda_k = \alpha + W_k(\beta e^{-\alpha}).}$$

scalar delay equations and the nonlinear eigenvalue problem

We shall apply these ideas to explore the potential for

transient growth in solutions to stable delay differential equations.

Solutions of scalar systems $x'(t) = \alpha x(t)$ behave monotonically: $|x(t)| = e^{t \operatorname{Re} \alpha}|x(0)|$. What about scalar delay equations?



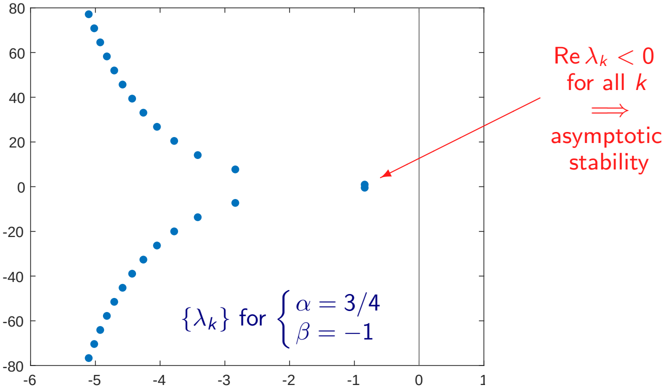
$$x'(t) = \alpha x(t) + \beta x(t-1) \qquad \Longrightarrow \qquad \lambda_k = \alpha + W_k(\beta e^{-\alpha}).$$

scalar delay equations and the nonlinear eigenvalue problem

We shall apply these ideas to explore the potential for

transient growth in solutions to stable delay differential equations.

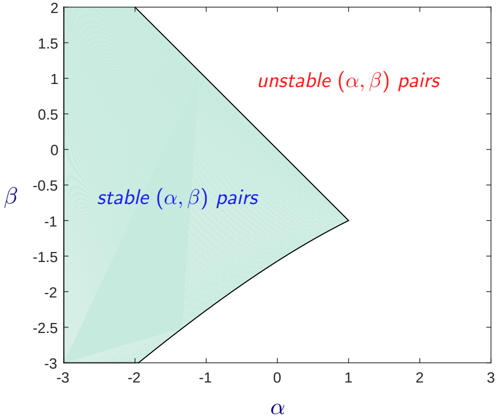
Solutions of scalar systems $x'(t) = \alpha x(t)$ behave monotonically: $|x(t)| = e^{t \operatorname{Re} \alpha}|x(0)|$. What about scalar delay equations?



$$x'(t) = \alpha x(t) + \beta x(t-1) \qquad \Longrightarrow \qquad \lambda_k = \alpha + W_k(\beta e^{-\alpha}).$$

stability chart for two-parameter delay equation

Conventional eigenvalue-based stability analysis reveals the (α, β) combinations that yield *asymptotically stable* solutions.



Such *stability charts* are standard tools for studying stability of parameter-dependent delay systems.

pseudospectra for nonlinear eigenvalue problems

Green & Wagenknecht [2006] and Michiels, Green, Wagenknecht, & Niculescu [2006] define *pseudospectra for nonlinear eigenvalue problems*, and apply them to delay differential equations.

See [Cullum, Ruehli 2001], [Wagenknecht, Michiels, Green 2008], [Bindel, Hood 2013].

Consider the nonlinear eigenvalue problem $\mathbf{T}(\lambda)\mathbf{v} = \mathbf{0}$ with

$$\boxed{\mathbf{T}(\lambda) = \sum_{j=1}^{m} f_j(\lambda)\,\mathbf{A}_j.}$$

For $p, q \in [1, \infty]$ and weights $w_1, \ldots, w_m \in (0, \infty]$, define the norm

$$\|(\mathbf{E}_1, \ldots, \mathbf{E}_m)\|_{p,q} = \left\| \begin{bmatrix} w_1\|\mathbf{E}_1\|_q \\ \vdots \\ w_m\|\mathbf{E}_m\|_q \end{bmatrix} \right\|_p .$$

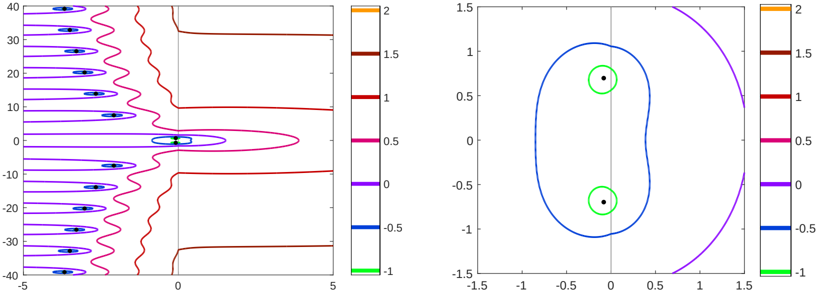
Given this way of measuring a perturbation to $\mathbf{T}(\lambda)$, [MGWN 2006] define

$$\boxed{\begin{aligned}\sigma_\varepsilon(\mathbf{T}) = \Big\{ z \in \mathbb{C} : z \in \sigma\Big(\sum_{j=1}^{m} f_j(\lambda)\,(\mathbf{A}_j + \mathbf{E}_j)\Big) \text{ for some} \\ \mathbf{E}_1, \ldots, \mathbf{E}_m \in \mathbb{C}^{n\times n} \text{ with } \|(\mathbf{E}_1, \ldots, \mathbf{E}_m)\|_{p,q} < \varepsilon \Big\}.\end{aligned}}$$

pseudospectra for the scalar delay equation

$$x'(t) = \alpha x(t) + \beta x(t-1)$$

$$\boxed{T(\lambda) = \lambda - \alpha - \beta e^{-\lambda}.}$$

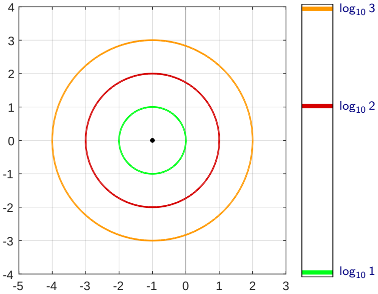


MGWN ε-pseudospectra for $\alpha = \frac{3}{4}$ and $\beta = -1$,
with perturbation norm given by $q \in [1, \infty]$ and $p = \infty$, and $w_1 = w_2 = 1$.

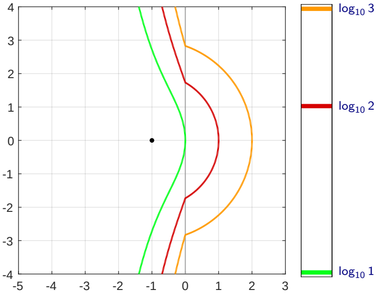
pseudospectra for the scalar delay equation

$$x'(t) = -x(t) + 0\,x(t-1)$$

$$\boxed{T(\lambda) = \lambda + 1}$$



$$\boxed{T(\lambda) = \lambda + 1 - 0\,\mathrm{e}^{-\lambda}}$$



MGWN ε-pseudospectra with $p = \infty$: *structure affects pseudospectra*.

the solution operator

To better understand transient behavior, just integrate the differential equation:

$$x'(t) = \alpha x(t) + \beta x(t-1)$$

history: $x(t-1) = u(t)$ for $t \in [0,1]$.

the solution operator

To better understand transient behavior, just integrate the differential equation:

$$\boxed{\begin{array}{c} x'(t) = \alpha x(t) + \beta x(t-1) \\ \text{history: } x(t-1) = u(t) \text{ for } t \in [0,1]. \end{array}}$$

Integrate

$$x'(t) = \alpha x(t) + \beta u(t)$$

to get, for $t \in [0,1]$,

$$\begin{aligned} x(t) &= \mathrm{e}^{t\alpha} x(0) + \beta \int_0^t \mathrm{e}^{(t-s)\alpha} u(s)\, \mathrm{d}s \\ &= \mathrm{e}^{t\alpha} u(1) + \beta \int_0^t \mathrm{e}^{(t-s)\alpha} u(s)\, \mathrm{d}s. \end{aligned}$$

the solution operator

To better understand transient behavior, just integrate the differential equation:

$$\boxed{\begin{aligned} &x'(t) = \alpha x(t) + \beta x(t-1) \\ &\text{history: } x(t-1) = u(t) \ \text{ for } t \in [0,1]. \end{aligned}}$$

Integrate

$$x'(t) = \alpha x(t) + \beta u(t)$$

to get, for $t \in [0, 1]$,

$$\begin{aligned} x(t) &= e^{t\alpha} x(0) + \beta \int_0^t e^{(t-s)\alpha} u(s)\, ds \\ &= e^{t\alpha} u(1) + \beta \int_0^t e^{(t-s)\alpha} u(s)\, ds. \end{aligned}$$

This operation maps *the history u* to *the solution x* for $t \in [0, 1]$:

$$u \in C([0, 1]) \quad \mapsto \quad x \in C([0, 1]).$$

the solution operator

Define the *solution operator* $\mathbf{K} : C[0,1] \to C[0,1]$ via

$$x(t) = (\mathbf{K}u)(t) = e^{t\alpha} u(1) + \beta \int_0^t e^{(t-s)\alpha} u(s)\, ds, \quad t \in [0,1].$$

the solution operator

Define the *solution operator* $\mathbf{K} : C[0,1] \to C[0,1]$ via

$$x(t) = (\mathbf{K}u)(t) = \mathrm{e}^{t\alpha}u(1) + \beta\int_0^t \mathrm{e}^{(t-s)\alpha}u(s)\,\mathrm{d}s, \qquad t \in [0,1].$$

		$x_m = x(t)\vert_{t\in[m-1,m]}$
define:	$x^{(0)} := u$	$t \in [-1,0]$
to advance t by 1 unit, apply $\mathbf{K}$:	$x^{(1)} := \mathbf{K}x^{(0)}$	$t \in [0,1]$
to advance t by 2 units, apply $\mathbf{K}^2$:	$x^{(2)} := \mathbf{K}x^{(1)} = \mathbf{K}^2x^{(0)}$	$t \in [1,2]$
	$\vdots$	
to advance t by m units, apply $\mathbf{K}^m$:	$x^{(m)} := \mathbf{K}x^{(m-1)} = \mathbf{K}^m x^{(0)}$	$t \in [m-1,m]$

the solution operator

Define the *solution operator* $\mathbf{K} : C[0,1] \to C[0,1]$ via

$$x(t) = (\mathbf{K}u)(t) = \mathrm{e}^{t\alpha} u(1) + \beta \int_0^t \mathrm{e}^{(t-s)\alpha} u(s)\, \mathrm{d}s, \qquad t \in [0,1].$$

		$x_m = x(t)\vert_{t\in[m-1,m]}$
define:	$x^{(0)} := u$	$t \in [-1, 0]$
to advance t by 1 unit, apply $\mathbf{K}$:	$x^{(1)} := \mathbf{K}x^{(0)}$	$t \in [0, 1]$
to advance t by 2 units, apply $\mathbf{K}^2$:	$x^{(2)} := \mathbf{K}x^{(1)} = \mathbf{K}^2 x^{(0)}$	$t \in [1, 2]$
	$\vdots$	
to advance t by m units, apply $\mathbf{K}^m$:	$x^{(m)} := \mathbf{K}x^{(m-1)} = \mathbf{K}^m x^{(0)}$	$t \in [m-1, m]$

View the delay system as a discrete-time dynamical system over 1-unit time intervals:

$$x^{(m)} = \mathbf{K}^m x^{(0)}.$$

discretizing the solution operator

We discretize the solution operator using a Chebyshev pseudospectral method based on [Trefethen 2000]; see [Bueler 2007], [Jarlebring 2008].

$$x(t_j) \approx x_j := \mathrm{e}^{t_j \alpha} u_0 + \sum_{k=0}^{N} \beta w_{j,k} u_k, \qquad w_{j,k} := \int_0^{t_j} \mathrm{e}^{(t_j - s)\alpha} \ell_k(s)\, \mathrm{d}s$$

$$\begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_N \end{bmatrix} = \underbrace{\begin{bmatrix} \mathrm{e}^{t_0\alpha} & 0 & \cdots & 0 \\ \mathrm{e}^{t_1\alpha} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{e}^{t_N\alpha} & 0 & \cdots & 0 \end{bmatrix}}_{\mathbf{E}_N(\alpha)} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix} + \beta \underbrace{\begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,N} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,N} \\ \vdots & \vdots & \ddots & \vdots \\ w_{N,0} & w_{N,1} & \cdots & w_{N,N} \end{bmatrix}}_{\mathbf{W}_N(\alpha)} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix}$$

discretizing the solution operator

We discretize the solution operator using a Chebyshev pseudospectral method based on [Trefethen 2000]; see [Bueler 2007], [Jarlebring 2008].

$$x(t_j) \approx x_j := e^{t_j \alpha} u_0 + \sum_{k=0}^{N} \beta w_{j,k} u_k, \qquad w_{j,k} := \int_0^{t_j} e^{(t_j - s)\alpha} \ell_k(s)\, ds$$

$$\begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_N \end{bmatrix} = \underbrace{\begin{bmatrix} e^{t_0\alpha} & 0 & \cdots & 0 \\ e^{t_1\alpha} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ e^{t_N\alpha} & 0 & \cdots & 0 \end{bmatrix}}_{\mathbf{E}_N(\alpha)} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix} + \beta \underbrace{\begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,N} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,N} \\ \vdots & \vdots & \ddots & \vdots \\ w_{N,0} & w_{N,1} & \cdots & w_{N,N} \end{bmatrix}}_{\mathbf{W}_N(\alpha)} \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_N \end{bmatrix}$$

$$\mathbf{K}_N := \mathbf{E}_N(\alpha) + \beta\, \mathbf{W}_N(\alpha)$$

$$\mathbf{x}^{(1)} := \mathbf{K}_N\, \mathbf{u}$$

$$\mathbf{x}^{(m)} := \mathbf{K}_N^m\, \mathbf{u}$$

approaches to transient analysis of delay equations

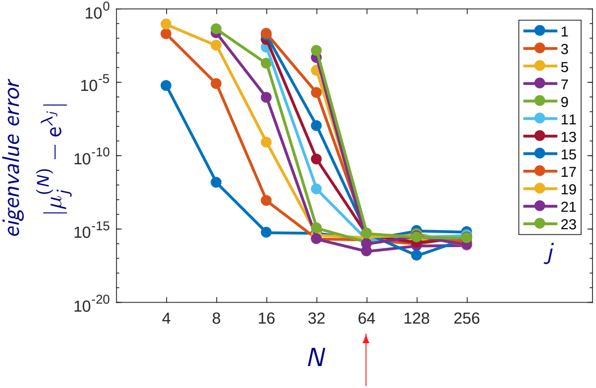
- Jacob Stroh [2006], in a master's thesis advised by Ed Bueler, computes L^2-pseudospectra of Chebyshev discretizations of the compact solution operator and considers nonnormality as a function of a time-varying coefficient in the delay term: *our approach follows closely*.
- Green & Wagenknecht [2006], in their paper about perturbation-based pseudospectra for delay equations, describe computing the pseudospectra of the generator for the solution semigroup as a way of gauging transient behavior; for relevant semigroup theory, see, e.g., [Engel & Nagel 2000].
- Hood & Bindel [2016+] apply Laplace transform/pseudospectral techniques to the solution operator for delay differential equations for upper/lower bounds on transient behavior. See also the Lyapunov approach to analyzing transient behavior in the 2005 Ph.D. thesis of Elmar Plischke.
- Solution operator approach converts a *finite dimensional nonlinear problem* into an *infinite dimensional linear problem*, akin to the *infinite Arnoldi algorithm* [Jarlebring, Meerbergen, Michiels 2010, 2012, 2014].

convergence of the eigenvalues of the solution operator

To study convergence, consider $\alpha = 0$, $\beta = -1$: $x'(t) = -x(t-1)$.

$\mu_j^{(N)}$: *the* j*th largest magnitude eigenvalue of* $\mathbf{K}_N$

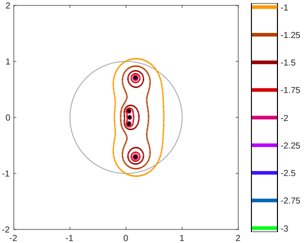
e^{λ_j}: λ_j *is the* j*th rightmost eigenvalue of the NLEVP*



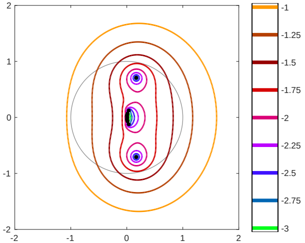
We generally use $N = 64$ *for our computations throughout what follows.*

nonconvergence of the L^2 pseudospectra of the solution operator

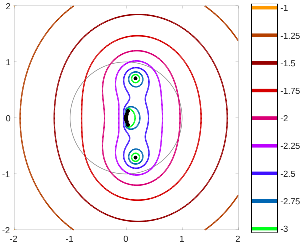
Eigenvalues converge, but the $L^2[0,1]$ pseudospectra of $\mathbf{K}_N$ do not:
the departure from normality increases with N !



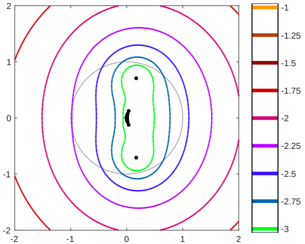
$N = 4$



$N = 16$



$N = 64$



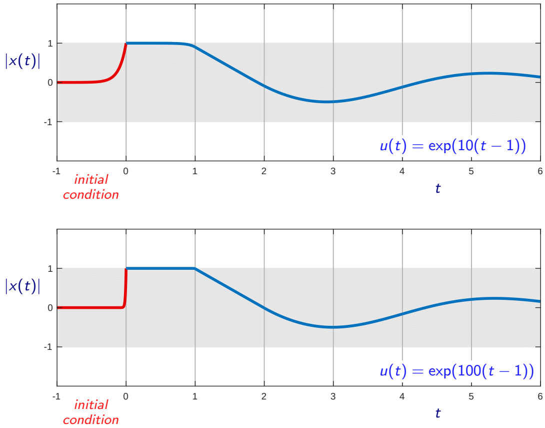
$N = 256$

the problem with the L^2 norm

Problem: *The $L^2(0,1)$ norm does not measure transient growth of $|x(t)|$.*

One can easily find $u(x)$ such that $\|u\|_{L^2[0,1]} \ll 1$ but $\|x\|_{L^2[0,1]} \geq 1$.

Let $\alpha = 0$, $\beta = -1$: $\quad x'(t) = -x(t-1) \quad \Longrightarrow \quad x(t) = u(1) - \int_0^t u(s)\,\mathrm{d}s.$

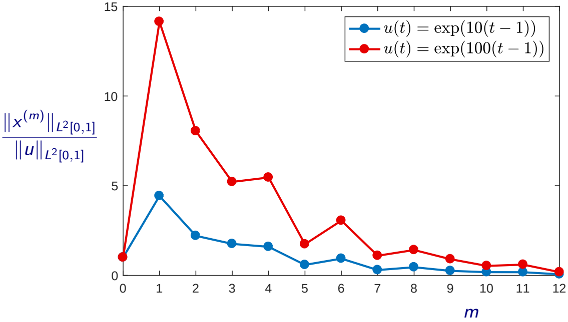


the problem with the L^2 norm

Problem: *The $L^2(0,1)$ norm does not measure transient growth of $|x(t)|$.*

One can easily find $u(x)$ such that $\|u\|_{L^2[0,1]} \ll 1$ but $\|x\|_{L^2[0,1]} \geq 1$.

Let $\alpha = 0$, $\beta = -1$: $\quad x'(t) = -x(t-1) \quad \Longrightarrow \quad x(t) = u(1) - \displaystyle\int_0^t u(s)\,ds.$



pseudospectra and transient growth of matrix powers

Since we care about the largest value $|x(t)|$ can take, we should really study

$$\|x^{(m)}\|_{L^\infty},$$

and thus the ε-pseudospectrum $\sigma_\varepsilon(\mathbf{K}_N)$ *defined using the ∞-norm*:

$$\begin{aligned}\sigma_\varepsilon(\mathbf{K}_N) &:= \{z \in \mathbb{C} : \|(z\mathbf{I} - \mathbf{K}_N)^{-1}\|_\infty > 1/\varepsilon\} \\ &:= \{z \in \mathbb{C} : z \in \sigma(\mathbf{K}_N + \mathbf{E}) \text{ for some } \mathbf{E} \in \mathbb{C}^{n\times n} \text{ with } \|\mathbf{E}\|_\infty < \varepsilon\}.\end{aligned}$$

pseudospectra and transient growth of matrix powers

Since we care about the largest value $|x(t)|$ can take, we should really study

$$\|x^{(m)}\|_{L^\infty},$$

and thus the ε-pseudospectrum $\sigma_\varepsilon(\mathbf{K}_N)$ *defined using the ∞-norm*:

$$\begin{aligned}\sigma_\varepsilon(\mathbf{K}_N) &:= \{z \in \mathbb{C} : \|(z\mathbf{I} - \mathbf{K}_N)^{-1}\|_\infty > 1/\varepsilon\} \\ &:= \{z \in \mathbb{C} : z \in \sigma(\mathbf{K}_N + \mathbf{E}) \text{ for some } \mathbf{E} \in \mathbb{C}^{n\times n} \text{ with } \|\mathbf{E}\|_\infty < \varepsilon\}.\end{aligned}$$

Even in Banach spaces, pseudospectra give lower bounds on transient growth; see, e.g., [Trefethen & E., 2005].

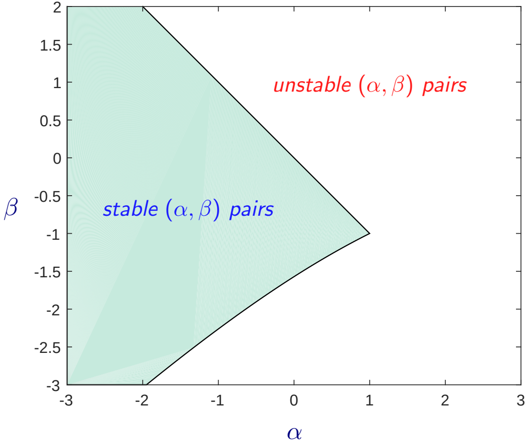
$$\sup_{m\geq 0} \|\mathbf{K}^m\| \geq \sup_{z\in\sigma_\varepsilon(\mathbf{K})} \frac{|z|-1}{\varepsilon}$$

If $\sigma_\varepsilon(\mathbf{K})$ extends more than ε outside the unit disk, $\|\mathbf{K}^m\|$ grows transiently.

Limitations: [Greenbaum & Trefethen 1994], [Ransford et al. 2007, 2009, 2011]

stability versus solution operator norm

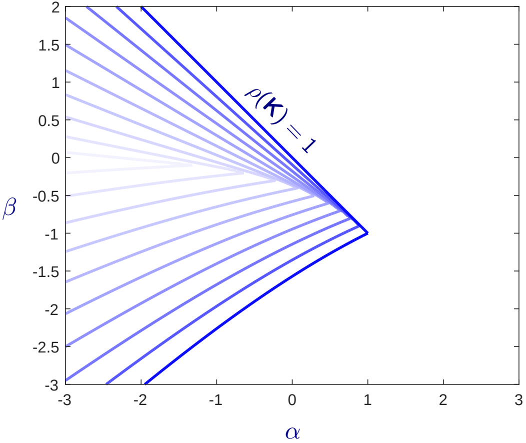
$$x'(t) = \alpha x(t) + \beta x(t-1)$$



Stable choices of the (α, β) parameters

stability versus solution operator norm

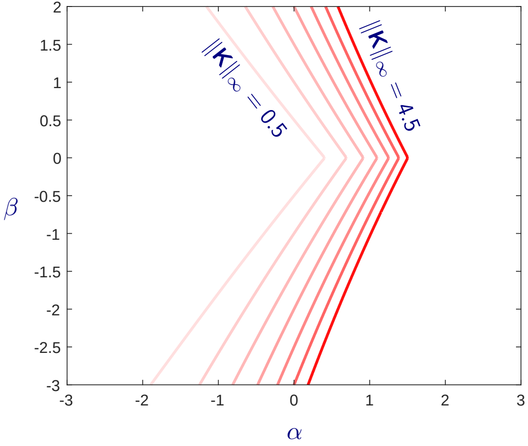
$$x'(t) = \alpha x(t) + \beta x(t-1)$$



Level sets: $\rho(\mathbf{K}) = 0.1, 0.2, \ldots, 1.0$

stability versus solution operator norm

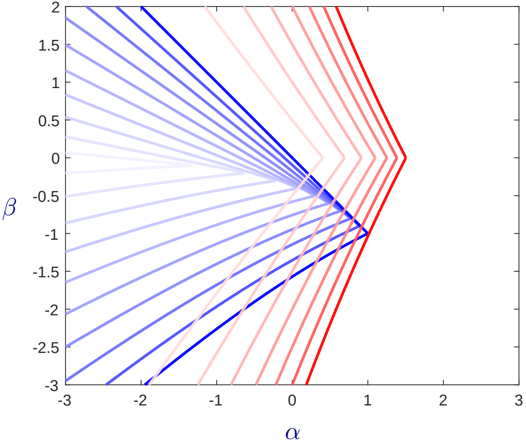
$$x'(t) = \alpha x(t) + \beta x(t-1)$$



Level sets: $\|\mathbf{K}\| = 0.5, 1.0, \ldots, 4.5$

stability versus solution operator norm

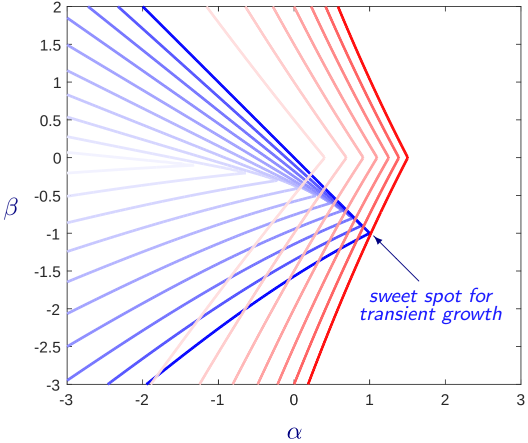
$$x'(t) = \alpha x(t) + \beta x(t-1)$$



Superimposed level sets for $\rho(\mathbf{K})$ and $\|\mathbf{K}\|$

stability versus solution operator norm

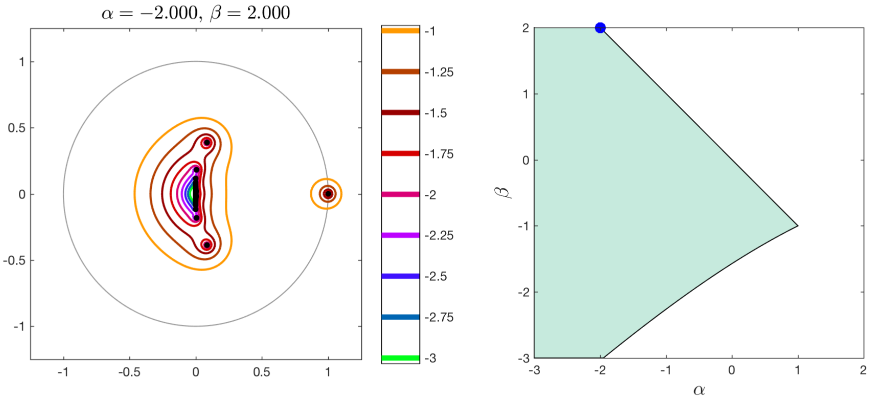
$$x'(t) = \alpha x(t) + \beta x(t-1)$$



Superimposed level sets for $\rho(\mathbf{K})$ and $\|\mathbf{K}\|$

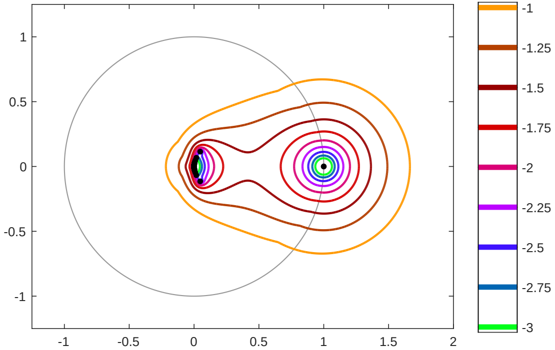
pseudospectra on the stability frontier

Animation of pseudospectra can be viewed using Adobe Acrobat.



solution matrix pseudospectra (∞-norm)

$$x'(t) = \alpha x(t) + \beta x(t-1)$$

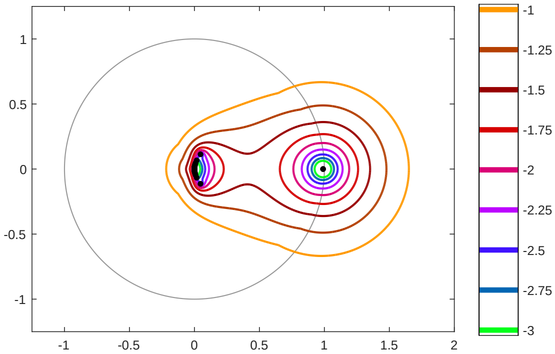


$\alpha = \ \ 1$ $\qquad$ $\rho(\mathbf{K}) = 1$

$\beta = -1$ $\qquad$ $\|\mathbf{K}\|_\infty = 4.43632$

solution matrix pseudospectra (∞-norm)

$$x'(t) = \alpha x(t) + \beta x(t-1)$$

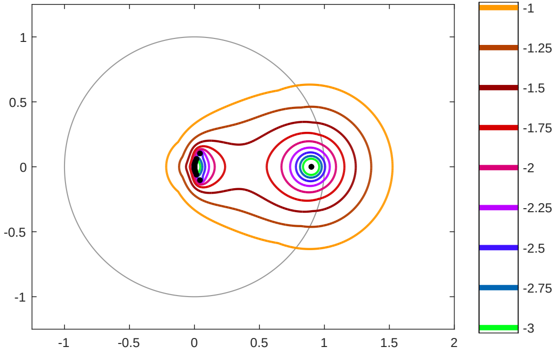


$\alpha = \quad 0.98995$ $\qquad \rho(\mathbf{K}) = 0.99000$

$\beta = -0.99000$ $\qquad \|\mathbf{K}\|_\infty = 4.38204$

solution matrix pseudospectra (∞-norm)

$$x'(t) = \alpha x(t) + \beta x(t-1)$$

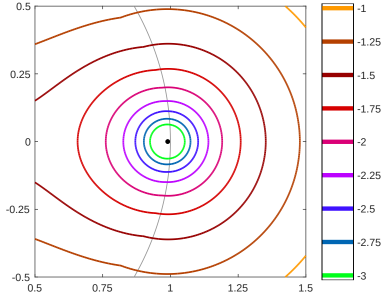


$\alpha = \quad 0.98995$ $\qquad \rho(\mathbf{K}) = 0.90000$

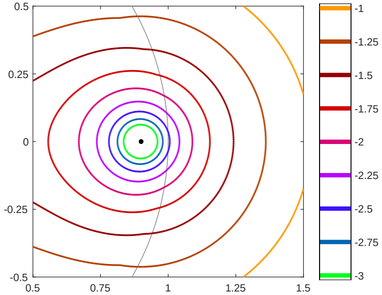
$\beta = -0.90000$ $\qquad \|\mathbf{K}\|_\infty = 3.90135$

solution matrix pseudospectra (∞-norm)

$$x'(t) = \alpha x(t) + \beta x(t-1)$$



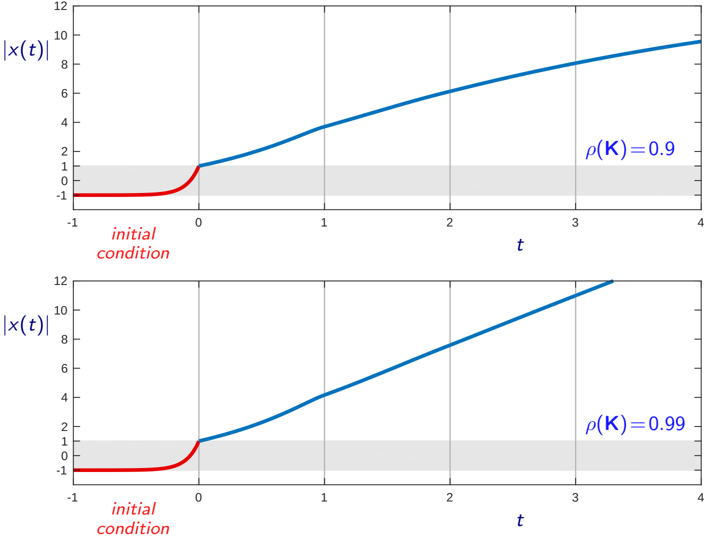
$\alpha = \quad 0.98995$ $\qquad \rho(\mathbf{K}) = 0.99000$
$\beta = -0.99000$ $\qquad \|\mathbf{K}\|_\infty = 4.38204$



$\alpha = \quad 0.98995$ $\qquad \rho(\mathbf{K}) = 0.90000$
$\beta = -0.90000$ $\qquad \|\mathbf{K}\|_\infty = 3.90135$

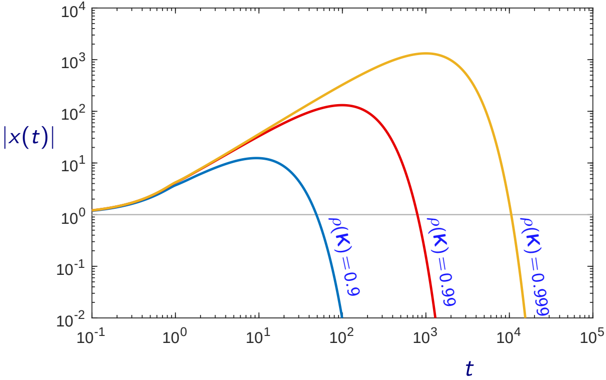
solution operator: transient growth

$$x'(t) = \alpha x(t) + \beta x(t-1)$$



solution operator: transient growth

$$x'(t) = \alpha x(t) + \beta x(t-1)$$



As $\alpha \uparrow 1$ and $\beta \downarrow -1$, solutions exhibit arbitrary transient growth, but slowly.

can scalar equations exhibit stronger transients?

Is faster transient growth possible in a scalar equation if we allow *multiple synchronized delays*?

$$x'(t) = c_0\, x(t) + c_1\, x(t-1) + c_2\, x(t-2) + \cdots + c_d\, x(t-d).$$

can scalar equations exhibit stronger transients?

Is faster transient growth possible in a scalar equation if we allow *multiple synchronized delays*?

$$x'(t) = c_0\, x(t) + c_1\, x(t-1) + c_2\, x(t-2) + \cdots + c_d\, x(t-d).$$

Key: *Look for solutions of the form* $x(t) = t^d e^{\lambda t}$.

can scalar equations exhibit stronger transients?

Is faster transient growth possible in a scalar equation if we allow *multiple synchronized delays*?

$$x'(t) = c_0\, x(t) + c_1\, x(t-1) + c_2\, x(t-2) + \cdots + c_d\, x(t-d).$$

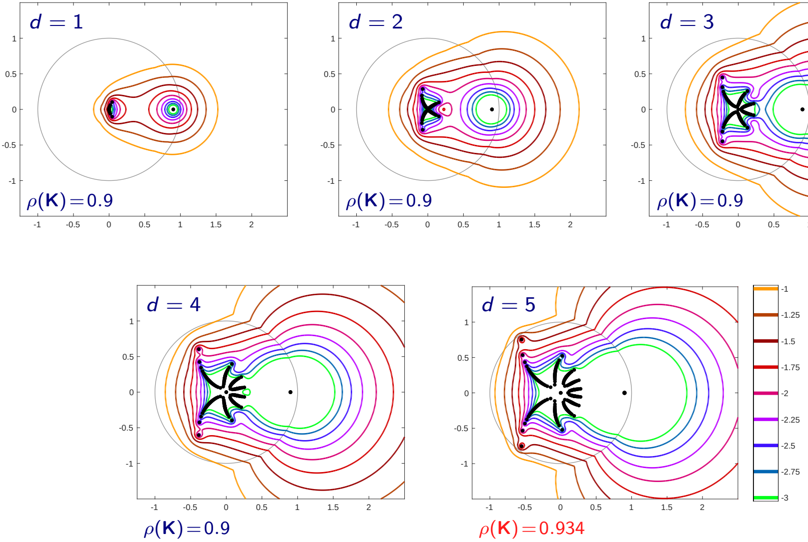
Key: *Look for solutions of the form $x(t) = t^d e^{\lambda t}$.*

One can show that $x(t) = t^d e^{\lambda t}$ is a solution if and only if $c_0, c_1, \ldots, c_d$ solve the Vandermonde linear system

$$\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 0 & 1 & 2 & \cdots & d \\ 0 & 1 & 4 & \cdots & d^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & 2^d & \cdots & d^d \end{bmatrix} \begin{bmatrix} c_0 \\ e^{-\lambda} c_1 \\ e^{-2\lambda} c_2 \\ \vdots \\ e^{-d\lambda} c_d \end{bmatrix} = \begin{bmatrix} \lambda \\ -1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

commensurate delays can give much larger pseudospectra

$$x'(t) = c_0 x(t) + c_1 x(t-1) + \cdots + c_d x(t-d)$$

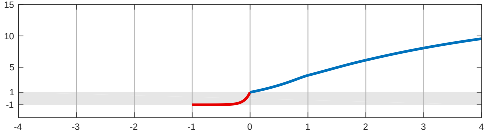


commensurate delays can induce strong transients

$$x'(t) = c_0\,x(t) + c_1\,x(t-1) + c_2\,x(t-2) + \cdots + c_d\,x(t-d)$$

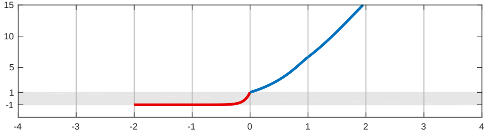
Initial data:
$x(t) = -1 + 2e^{10t}$
for $t \le 0$

$d = 1$



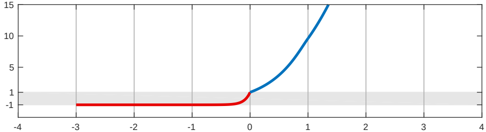
$$c_0 = 0.8946 \qquad c_1 = -0.9000$$

$d = 2$



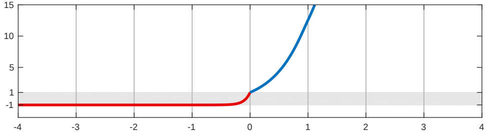
$$c_0 = 1.3946 \qquad c_1 = -1.8000 \qquad c_2 = 0.4050$$

$d = 3$



$$c_0 = 1.7280 \qquad c_1 = -2.7000 \qquad c_2 = 1.2150 \qquad c_3 = -0.2430$$

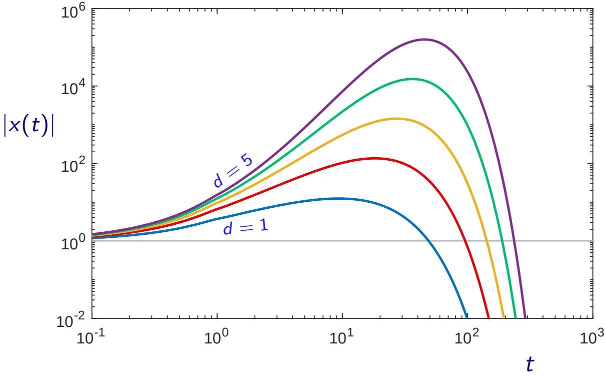
$d = 4$



$$c_0 = 1.9780 \qquad c_1 = -3.9600 \qquad c_2 = 2.4300 \qquad c_3 = -0.9720 \qquad c_4 = 0.1640$$

commensurate delays can induce strong transients

$$x'(t) = c_0 x(t) + c_1 x(t-1) + \cdots + c_d x(t-d)$$



With commensurate delays, solutions to scalar equations can exhibit significant transient growth very quickly in time.

summary

rational interpolation for nlevps

Rational / Loewner techniques motivated by algorithms from model reduction

- *Structure Preserving Rational Interpolation*: iteratively improve projection subspaces via interpolation points and directions.
- *Data-Driven Rational Interpolation Matrix Pencils*: reduce nonlinear problem to linear matrix pencil with tangential interpolation property.
- *Minimal Realization via Rational Contour Integrals*: isolates a transfer function for a *linear system*, recover via Loewner minimal realization techniques.

transients for delay equations

Solutions to *scalar* delay equations can exhibit strong transient growth.

- *Finite dimensional nonlinear* problem $\Rightarrow$ *infinite dimensional linear* problem
- Pseudospectral theory applies to the linear problem, *but the choice of norm is important*.
- Chebyshev collocation keeps the discretization matrix size small.
- Adding commensurate delays enables a *faster rate* of initial transient growth.