



Restarted Krylov Subspace Approximations of Matrix Functions

Stefan Güttel

joint work with

Martin Afanasjew, Michael Eiermann & Oliver G. Ernst

Zürich, July 2007



Problem

Given: $A \in \mathbb{C}^{n \times n}$, $\mathbf{b} \in \mathbb{C}^n$, $\|\mathbf{b}\|_2 = 1$, f nice.

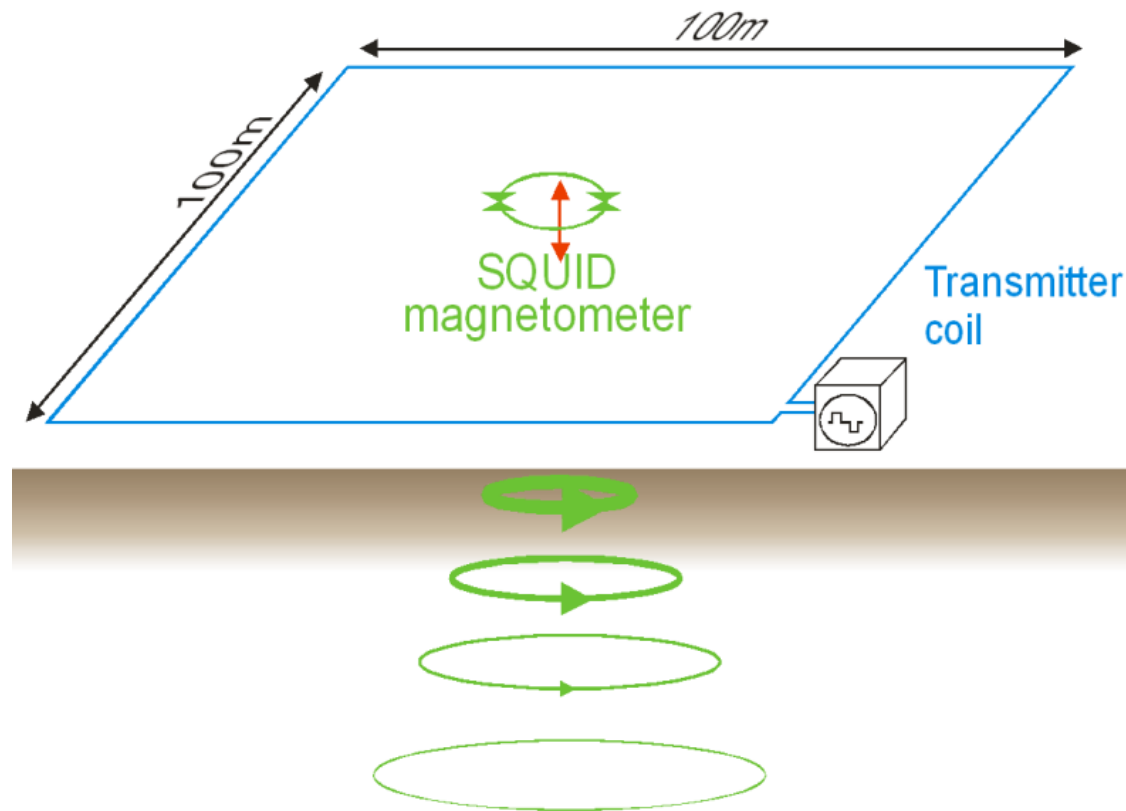
Sought: $f(A)\mathbf{b}$.

Outline

- Motivation
- Arnoldi approximation
- Restarting
- Convergence

Our Original Motivation

Numerical simulation of **transient electromagnetic** (TEM) geophysical exploration (collaboration with Institute of Geophysics, TU Freiberg).



- Time-evolution of electrical field $\mathbf{E}(\mathbf{x}, t)$ can be modeled via

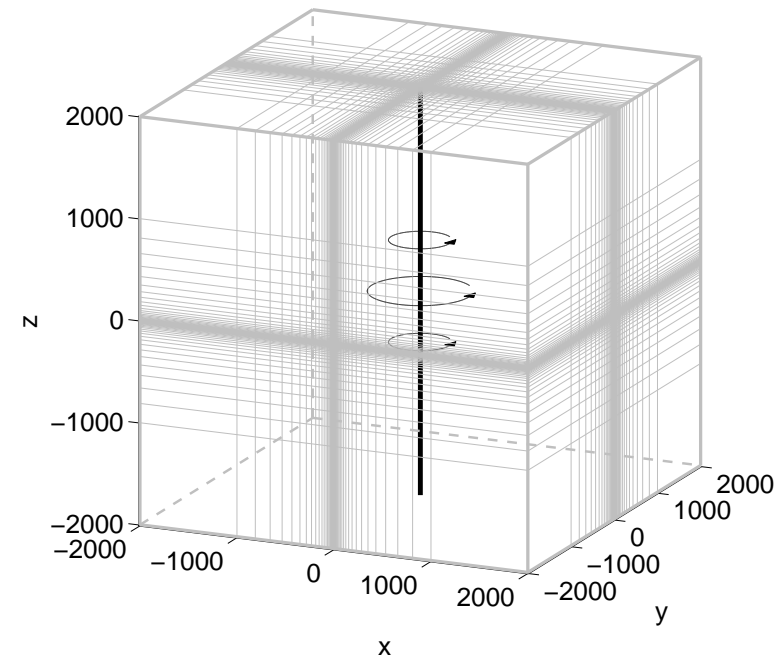
$$\sigma \mathbf{E}_t + \nabla \times (\mu^{-1} \nabla \times \mathbf{E}) = \mathbf{0} \quad \text{on } \Omega \subset \mathbb{R}^3,$$

$$\mathbf{E}(\mathbf{x}, t = 0) = \mathbf{E}^{(0)}(\mathbf{x}) \quad + \text{ boundary conditions.}$$

- Discretization in space leads to homogeneous linear IVP with time-independent coefficients.
- Solution has the form

$$\mathbf{u}(t) = \exp(-tA) \mathbf{u}_0,$$

where A discretizes $\sigma^{-1} \nabla \times (\mu^{-1} \nabla \times \cdot)$ and is large and sparse.



- Very popular in geophysics: **Spectral Lanczos Decomposition Method** [Druskin & Knizhnerman, 1988].

Arnoldi Approximation

Arnoldi process applied to A with initial vector \mathbf{b} yields
Arnoldi (Lanczos) decomposition

$$AV_m = V_m H_m + \eta_{m+1,m} \mathbf{v}_{m+1} \mathbf{e}_m^T,$$

where

$$V_m = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_m] \in \mathbb{C}^{n \times m}, \quad \{\mathbf{v}_1, \dots, \mathbf{v}_m\} \text{ ON basis of } \mathcal{K}_m(A, \mathbf{b}),$$
$$\{\mathbf{v}_1, \dots, \mathbf{v}_{m+1}\} \text{ ON basis of } \mathcal{K}_{m+1}(A, \mathbf{b}),$$
$$\mathbf{v}_1 = \mathbf{b},$$

$$H_m = [\eta_{i,j}] \in \mathbb{C}^{m \times m} \quad \text{unreduced upper Hessenberg matrix,}$$

$$\mathbf{e}_m = [0, \dots, 0, 1]^T \in \mathbb{R}^m.$$

Arnoldi approximation of $f(A)\mathbf{b}$:

$$f(A)\mathbf{b} \approx \mathbf{f}_m := V_m f(H_m) \mathbf{e}_1$$

[Druskin & Knizhnerman, 1989],
[Gallopoulos & Saad, 1992],
[Saad, 1992],
[Hochbruck & Lubich, 1995],
[Philippe & Sidje, 1995, 1998],
[Frommer & Simoncini, 2006].

Interpretation: Let $p_{m-1} \in \mathcal{P}_{m-1}$ be a (the) polynomial which interpolates f at the eigenvalues of H_m . Then

$$\begin{aligned} f(H_m) &= p_{m-1}(H_m) && \text{by definition of } f \\ V_m p_{m-1}(H_m) \mathbf{e}_1 &= p_{m-1}(A) \mathbf{b} && \text{since } p_{m-1} \in \mathcal{P}_{m-1}. \end{aligned}$$

The Arnoldi approximation can therefore be seen as an interpolation process where the nodes are the Ritz values of A .



Advantage: avoids A (only $v \rightarrow Av$), avoids explicit interpolation, requires only evaluation of $f(H_m)$ for (small) matrix H_m .

Drawback: high computational costs for Arnoldi decomposition (in nonsymm. case), requires basis V_m (extensive storage)

in Hermitian case: 2-pass Lanczos

- first pass of Lanczos to determine H_m ,
- compute $f(H_m)e_1$,
- second pass of Lanczos to determine the linear combination of v_j with coefficients from $f(H_m)e_1$.

(doubles the number of matrix-vector products)



Restarting the Arnoldi Approximation

Consider two cycles of Arnoldi

$$\begin{aligned} AV_1 &= V_1 H_1 + \eta_2 \mathbf{v}_{m+1} \mathbf{e}_m^T, & V_1(:, 1) &= \mathbf{b} \\ AV_2 &= V_2 H_2 + \eta_3 \mathbf{v}_{2m+1} \mathbf{e}_m^T, & V_2(:, 1) &= \mathbf{v}_{m+1}. \end{aligned}$$

Since columns of $\widehat{V}_2 := [V_1, V_2]$ form (nonorthogonal) basis of $\mathcal{K}_{2m}(A, \mathbf{b})$, we can combine to **Arnoldi-like decomposition**

$$A\widehat{V}_2 = \widehat{V}_2 \widehat{H}_2 + \eta_3 \mathbf{v}_{2m+1} \mathbf{e}_{2m}^T,$$

$$\widehat{H}_2 := \begin{bmatrix} H_1 & O \\ \eta_2 \mathbf{e}_1 \mathbf{e}_m^T & H_2 \end{bmatrix}.$$

Arnoldi-like approximation: $\widehat{\mathbf{f}}_2 := \widehat{V}_2 f(\widehat{H}_2) \mathbf{e}_1$.

Interpretation: $\widehat{\mathbf{f}}_2 = p_{2m-1}(A)\mathbf{b}$, where p_{2m-1} interpolates f at $\Lambda(\widehat{H}_2) = \Lambda(H_1) \cup \Lambda(H_2)$.

After k cycles: $A\widehat{V}_k = \widehat{V}_k\widehat{H}_k + \eta_{k+1}\mathbf{v}_{km+1}\mathbf{e}_{km}^T$, where $\widehat{V}_k := [V_1, V_2, \dots, V_k]$,

$$\widehat{H}_k := \begin{bmatrix} H_1 & & & & \\ E_2 & H_2 & & & \\ & \ddots & \ddots & & \\ & & E_k & H_k & \end{bmatrix}, \quad E_j := \eta_j \mathbf{e}_1 \mathbf{e}_m^T \in \mathbb{R}^{m \times m}.$$

With

$$\widehat{F}_k := f(\widehat{H}_k) = \begin{bmatrix} F_{1,1} & & & & \\ F_{2,1} & F_{2,2} & & & \\ \vdots & \vdots & \ddots & & \\ F_{k,1} & F_{k,2} & \dots & F_{k,k} & \end{bmatrix}, \quad \text{where } F_{j,j} = f(H_j),$$

$$\widehat{\mathbf{f}}_k := \widehat{V}_k f(\widehat{H}_k) \mathbf{e}_1 = \underbrace{V_1 F_{1,1} \mathbf{e}_1}_{\text{first cycle}} + \underbrace{V_2 F_{2,1} \mathbf{e}_1}_{\text{second cycle}} + \dots + \underbrace{V_k F_{k,1} \mathbf{e}_1}_{k\text{-th cycle}}.$$



Algorithm 1 [Eiermann & Ernst, 2006]

Given A , \mathbf{b} , $\|\mathbf{b}\| = 1$, f

$\mathbf{v}_1 := \mathbf{b}$, $\hat{\mathbf{f}}_0 := \mathbf{0}$

For $k = 1, 2, \dots$

Compute Arnoldi decomposition $AV_k = V_k H_k + \eta_{k+1} \mathbf{v}_{km+1} \mathbf{e}_m^T$
of $\mathcal{K}_m(A, \mathbf{v}_{(k-1)m+1})$

If $k = 1$

$$\hat{H}_k := H_1$$

$$\text{Else } \hat{H}_k := \begin{bmatrix} \hat{H}_{k-1} & \mathbf{O} \\ \eta_k \mathbf{e}_1 \mathbf{e}_{(k-1)m}^T & H_k \end{bmatrix}$$

Update the approximation $\hat{\mathbf{f}}_k := \hat{\mathbf{f}}_{k-1} + V_k [f(\hat{H}_k) \mathbf{e}_1]_{(k-1)m+1:km}$.

Drawback: $f(\hat{H}_k)$ is computed (although only the last m entries of its first column are needed).

Approximating f by rational function r in partial fraction form used by Gallopoulos & Saad [1992], Lopez & Simoncini [2006] for unrestarted case

$$f(\widehat{H}_k) \mathbf{e}_1 \approx r(\widehat{H}_k) \mathbf{e}_1 = \left[\alpha_0 I + \sum_{\ell=1}^N \alpha_\ell (\omega_\ell I - \widehat{H}_k)^{-1} \right] \mathbf{e}_1.$$

$$r(\widehat{H}_k) \mathbf{e}_1 = \alpha_0 \mathbf{e}_1 + \sum_{\ell=1}^N \widehat{\mathbf{r}}_\ell, \quad \text{where}$$

$$(\omega_\ell I - \widehat{H}_k) \widehat{\mathbf{r}}_\ell = \begin{bmatrix} \omega_\ell I - H_1 & & & & & \\ -E_2 & \omega_\ell I - H_2 & & & & \\ & & \ddots & & & \\ & & & \ddots & & \\ & & & & -E_k & \omega_\ell I - H_k \end{bmatrix} \begin{bmatrix} \mathbf{r}_{\ell,1} \\ \mathbf{r}_{\ell,2} \\ \vdots \\ \mathbf{r}_{\ell,k} \end{bmatrix} = \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}$$

or, recursively,

$$\begin{aligned} (\omega_\ell I - H_1) \mathbf{r}_{\ell,1} &= \mathbf{e}_1, \\ (\omega_\ell I - H_j) \mathbf{r}_{\ell,j} &= E_j \mathbf{r}_{\ell,j-1}, \quad j = 2, \dots, k. \end{aligned}$$

Algorithm 2 [Afanasjew, Eiermann, Ernst & G., 2007]

Given A , \mathbf{b} , $\|\mathbf{b}\| = 1$, $(\alpha_\ell, \omega_\ell)_{\ell=1}^N$

$\mathbf{v}_1 := \mathbf{b}$, $\hat{\mathbf{f}}_0 := \alpha_0 \mathbf{b}$

For $k = 1, 2, \dots$

 Compute Arnoldi decomposition $AV_k = V_k H_k + \eta_{k+1} \mathbf{v}_{km+1} \mathbf{e}_m^T$
 of $\mathcal{K}_m(A, \mathbf{v}_{(k-1)m+1})$

 If $k = 1$

 For $\ell = 1, 2, \dots, N$

 Solve $(\omega_\ell I - H_k) \mathbf{r}_{\ell,1} = \mathbf{e}_1$

 Else

 For $\ell = 1, 2, \dots, N$

 Solve $(\omega_\ell I - H_k) \mathbf{r}_{\ell,k} = E_k \mathbf{r}_{\ell,k-1}$

$\mathbf{h}_k := \sum_{\ell=1}^N \alpha_\ell \mathbf{r}_{\ell,k}$

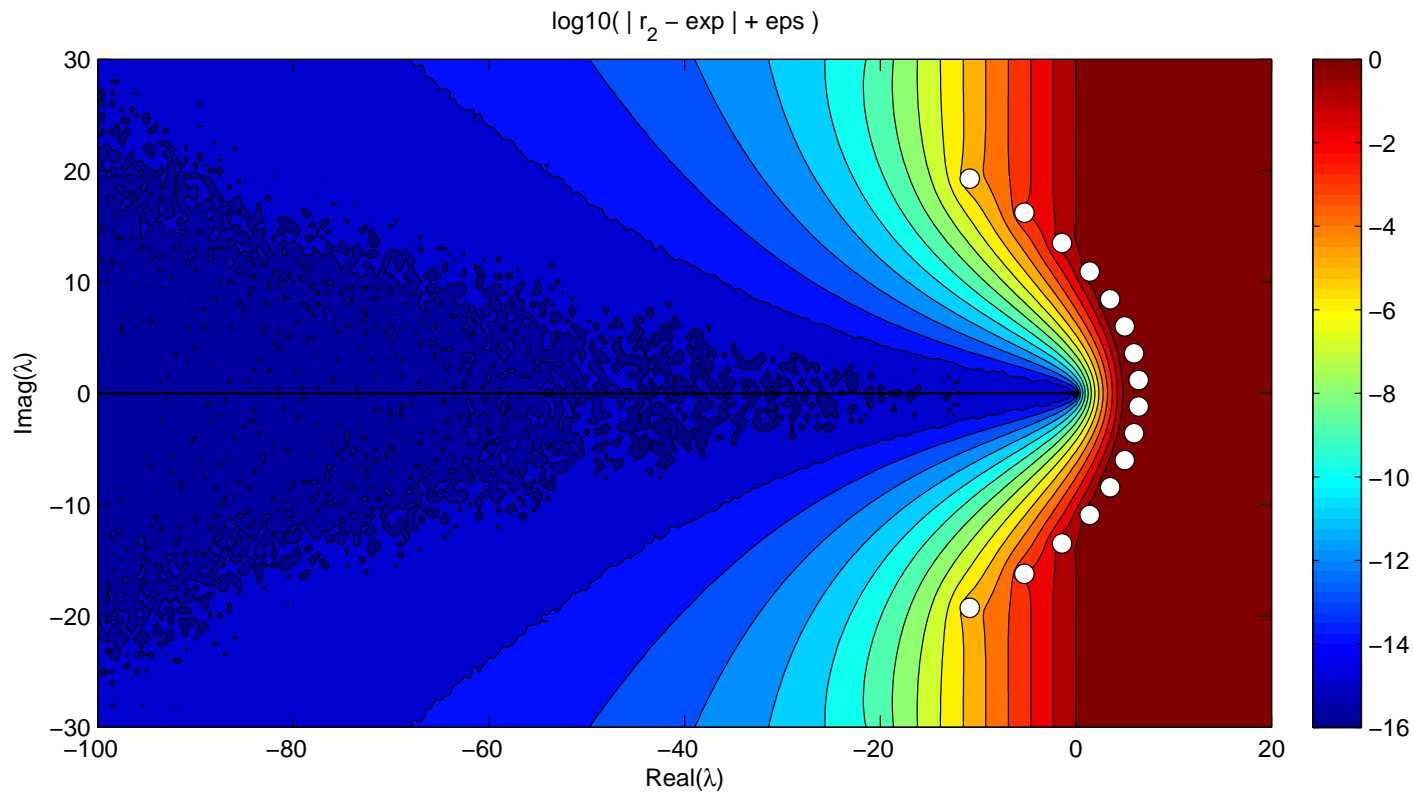
$\hat{\mathbf{f}}_k := \hat{\mathbf{f}}_{k-1} + V_k \mathbf{h}_k$.

Note: Constant work and constant storage space per restart cycle.

Solving Maxwell's Equations

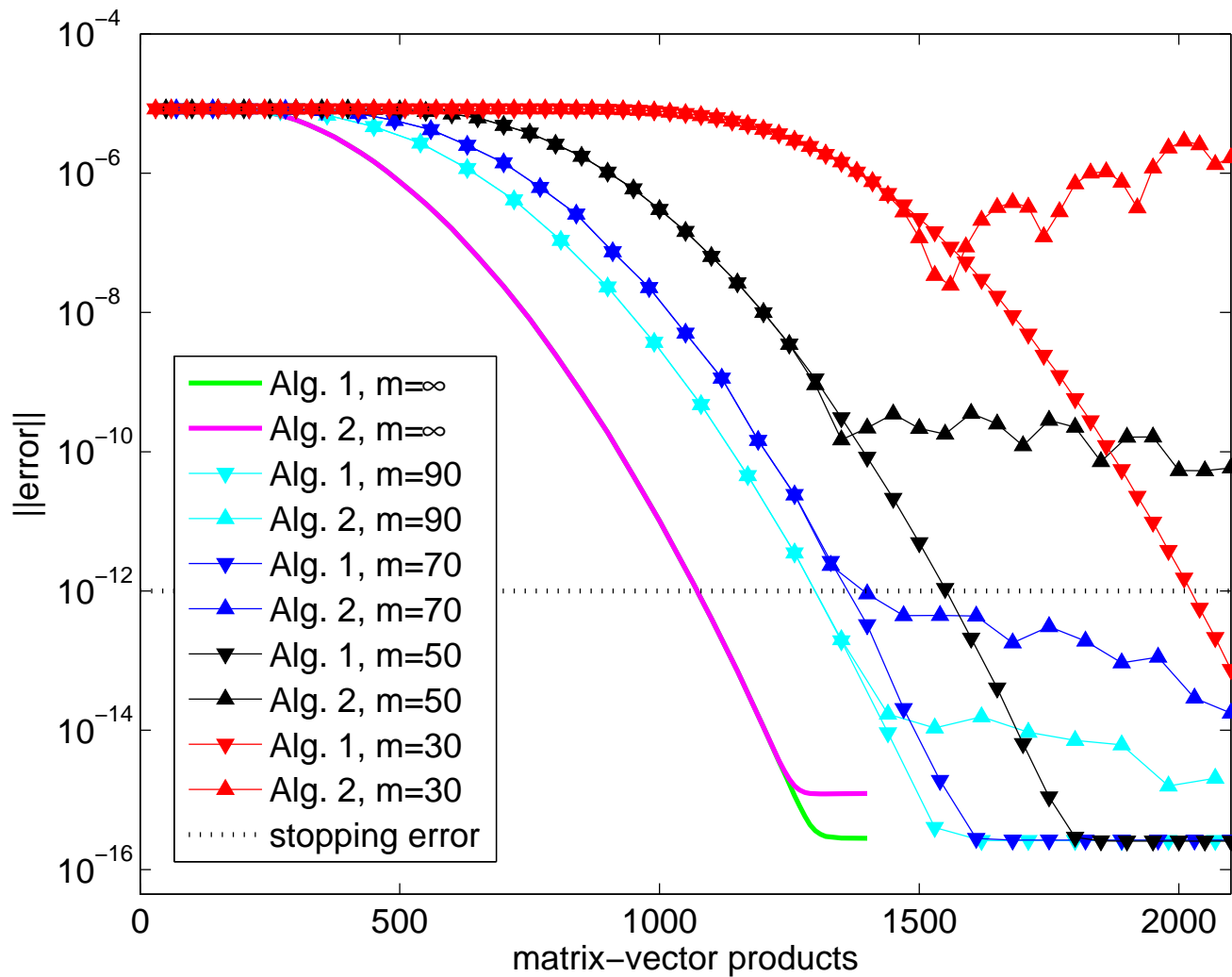
$A \in \mathbb{R}^{n \times n}$ symmetric, $n = 565,326$, $\Lambda(-tA) \in [-10^5, 0]$,

$f = \exp$, $r_2 =$ uniform rational best approximation on $(-\infty, 0]$ of type [16/16]
[Cody, Meinardus & Varga, 1969].



	Algorithm 1			Algorithm 2		
m	time [s]	mvp	acc.	time [s]	mvp	acc.
$\infty (I)$	118	1072	9.93e-13	86	1072	9.93e-13
$\infty (II)$	176	2144	9.93e-13	144	2144	9.93e-13
90	273	1350	1.92e-13	118	1350	2.01e-13
70	339	1400	3.28e-13	112	1400	9.13e-13
50	613	1600	2.10e-13	very slow convergence		
30	2014	2040	5.64e-13	divergence		





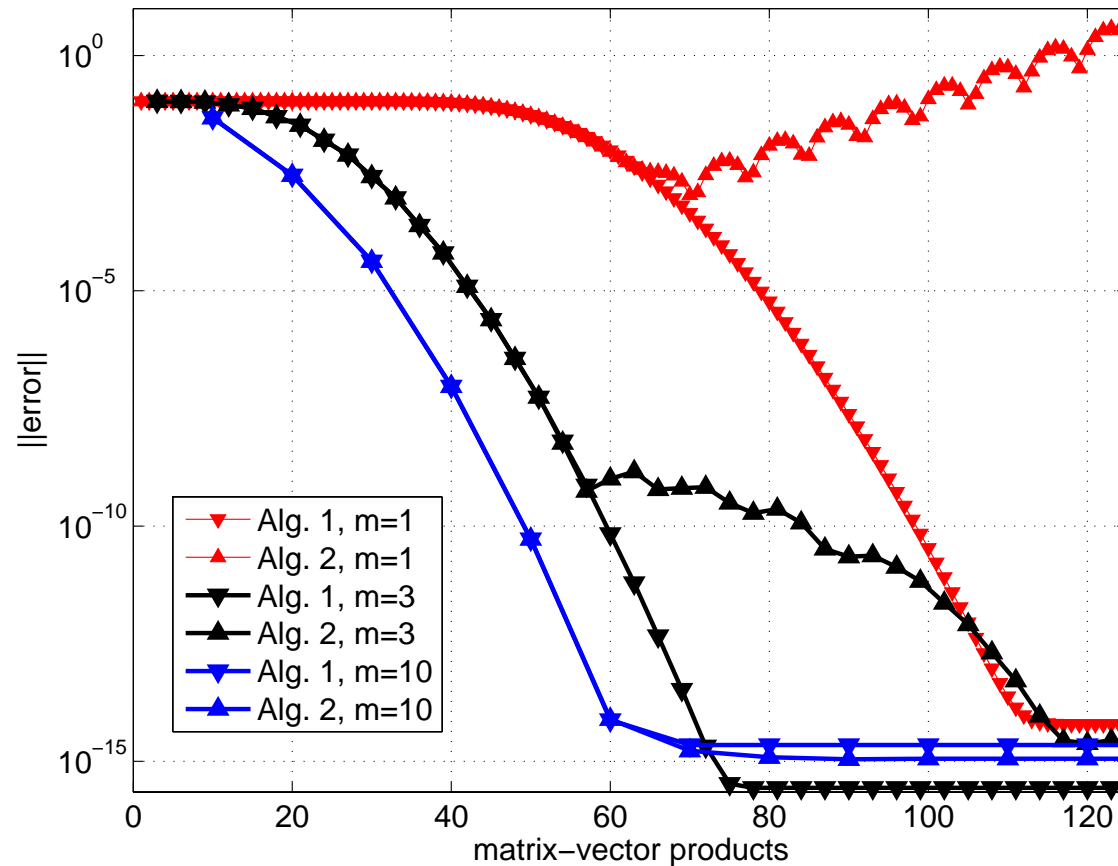
- Observations:**
- Convergence history of Algorithm 2 shows two phases.
 - Final accuracy of Algorithm 2 is lower.



Convergence Analysis

Model problem: $f = \exp$, $r_2 = \text{CMV approximation}$,

$$A = \text{diag}(-100, -99, \dots, 0) \in \mathbb{R}^{101 \times 101}, \quad \mathbf{b} = [1, 1, \dots, 1]^T / \sqrt{101} \in \mathbb{R}^{101}.$$



Algorithm 1: Approximates $\exp(\hat{H}_k)$ by $r_1(\hat{H}_k) = r(\hat{H}_k/2^s)^{2^s}$, where r is a Padé-fraction (expm, [Higham, 2005]), r_1 depends on the argument and is **not a rational function**: if the argument is a scalar λ then $r_1(\lambda)$ is an accurate approximation to $\exp(\lambda)$ regardless of where in the complex plane λ is located; in particular, r_1 has no finite poles.

Algorithm 2: Approximates $\exp(\hat{H}_k)$ by $r_2(\hat{H}_k)$, where r_2 is **a fixed rational function**, namely the CMV approximation of type [16/16]. r_2 approximates \exp well only on $(-\infty, 0]$ and has finite poles.

Main distinction: The restarted Arnoldi method **converges superlinearly** to an entire function (Algorithm 1) and **converges/diverges linearly** since r_2 has finite singularities (Algorithm 2).

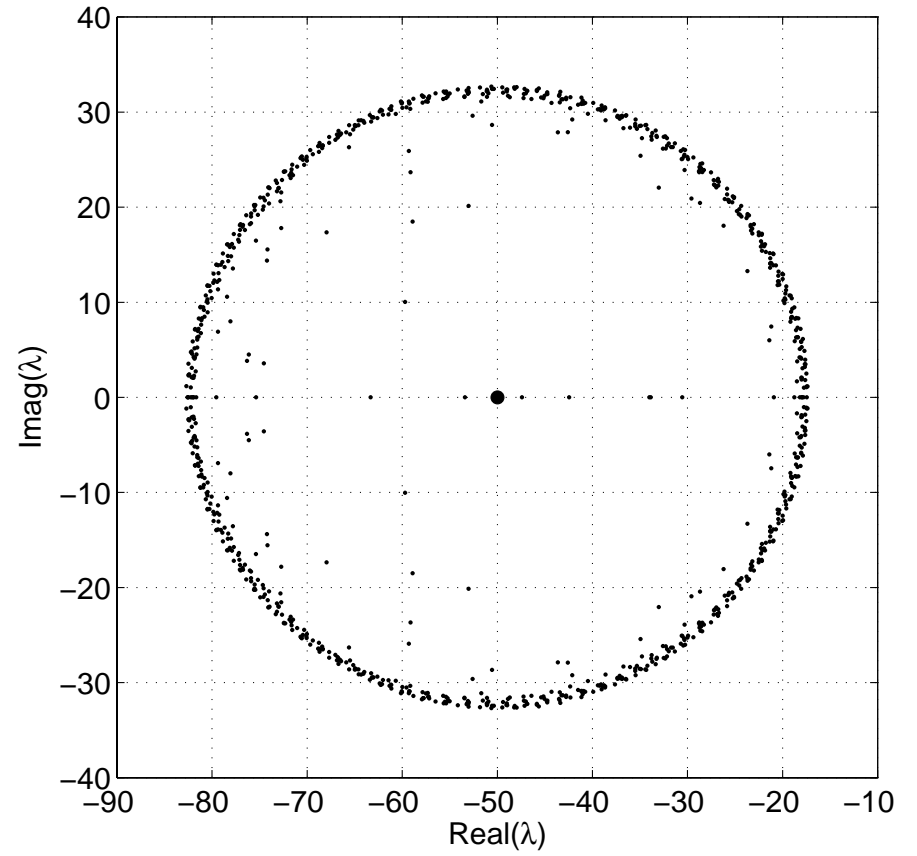
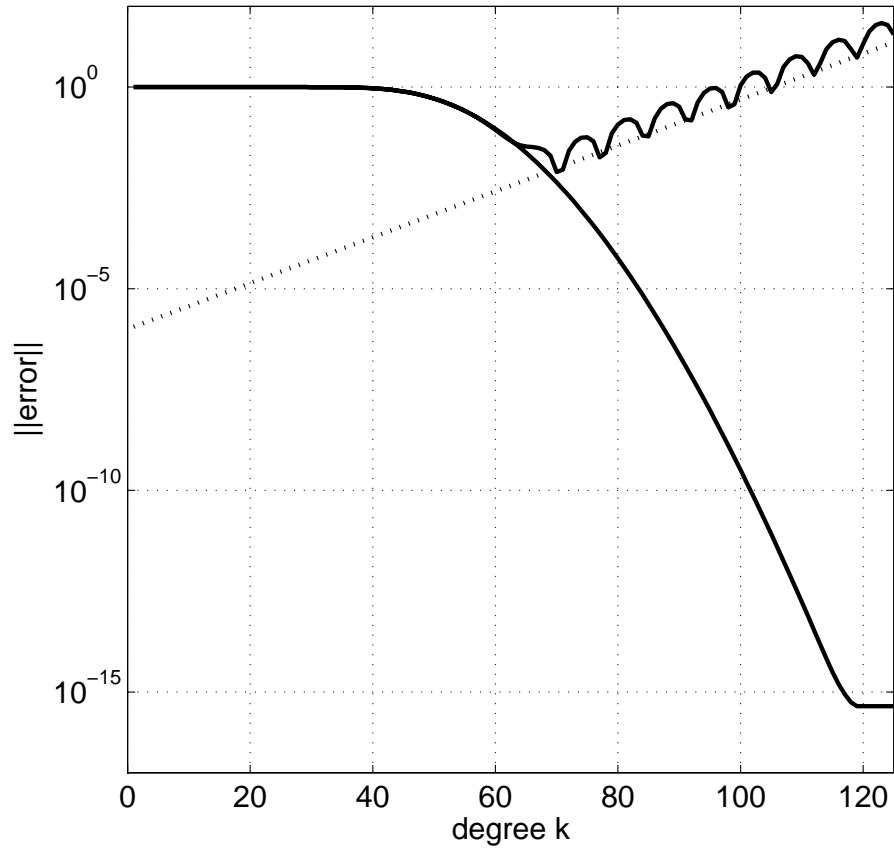
Restart length $k = 1$:

- $\hat{\mathbf{f}}_k = p_{k-1}(A)\mathbf{b}$, where p_{k-1} interpolates $f = \exp$ or $f = r_2$ at the eigenvalues of \hat{H}_k ,
- \hat{H}_k is bidiagonal, all diagonal entries are equal to $\vartheta = -50$,
- the interpolating polynomials are therefore truncated Taylor series,
- the Taylor series for \exp (Algorithm 1) converges everywhere, whereas the Taylor series for r_2 (Algorithm 2) converges only for

$$|\lambda + 50| < \min_{\omega=\text{pole of } r_2} |\omega + 50| \approx 44 < 50 = \max_{\lambda \in \Lambda(A)} |\lambda + 50|,$$

- Algorithm 2 therefore ultimately diverges like

$$\left[\max_{\lambda \in \Lambda(A)} |\lambda + 50|/44 \right]^k = (50/44)^k \approx 1.14^k.$$



Restart length $k > 1$:

- \widehat{H}_k is a block-bidiagonal matrix, where the diagonal blocks are symmetric tridiagonal,
- can show [\[Afanasjew, Eiermann, Ernst & G., 2007\]](#)

$$\lim_{j \rightarrow \infty} H_{2j-1} = \widetilde{H}_1 \quad \text{and} \quad \lim_{j \rightarrow \infty} H_{2j} = \widetilde{H}_2,$$

- the nodal sequence (the eigenvalues of \widehat{H}_k) $\vartheta_1, \vartheta_2, \vartheta_3, \dots$ therefore has the property

$$\lim_{j \rightarrow \infty} \vartheta_{2mj+\nu} = \widetilde{\vartheta}_\nu \quad \text{for } \nu = 1, 2, \dots, 2m,$$

- by theorems of [Walsh \[1969\]](#) the convergence of the interpolation process is asymptotically determined by the lemniscates

$$L_\tau := \{\lambda \in \mathbb{C} : |w_{2m}(\lambda)| = \tau^{2m}\}, \quad \tau > 0, \quad w_{2m}(\lambda) = \prod_{\nu=1}^{2m} (\lambda - \widetilde{\vartheta}_\nu),$$

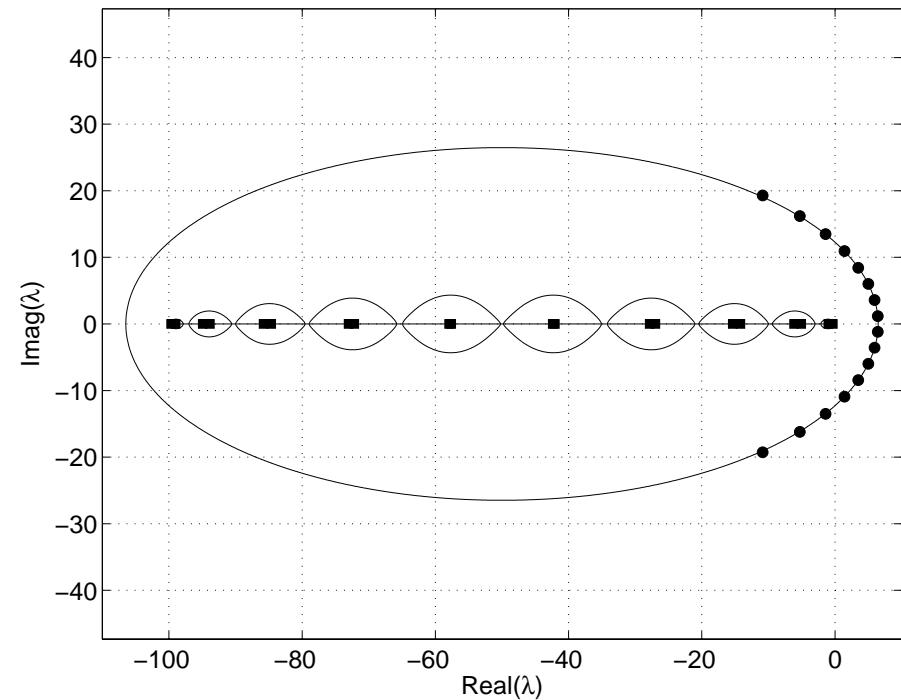
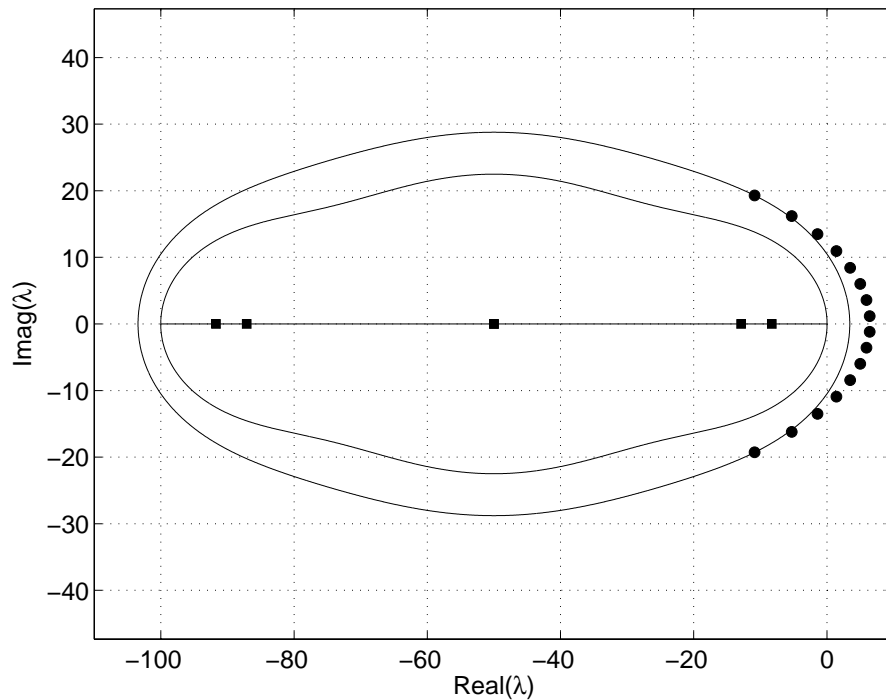
- more precisely,

$$\limsup_{k \rightarrow \infty} \|\widehat{\mathbf{f}}_k - \exp(A)\mathbf{b}\|^{1/km} = \frac{\tau_A}{\tau_s},$$

where

$$\tau_A = \min\{\tau : \Lambda(A) \text{ is contained in the closed interior of } L_\tau\},$$

$$\tau_s = \max\{\tau : r_2 \text{ is analytic in the interior of } L_\tau\}.$$



To summarize: Algorithm 2 converges as follows

- initially, behavior like polynomial interpolation to \exp (no error reduction in the startup phase and then superlinear convergence) as long as
$$\frac{d^{km} r_2}{d\lambda^{km}} \approx \frac{d^{km} \exp}{d\lambda^{km}},$$
- there is a point from where on the poles of r_2 become visible, fairly independent of the restart length,
- after this point of transition, behavior like polynomial interpolation to r_2 (slower linear convergence or even linear divergence).

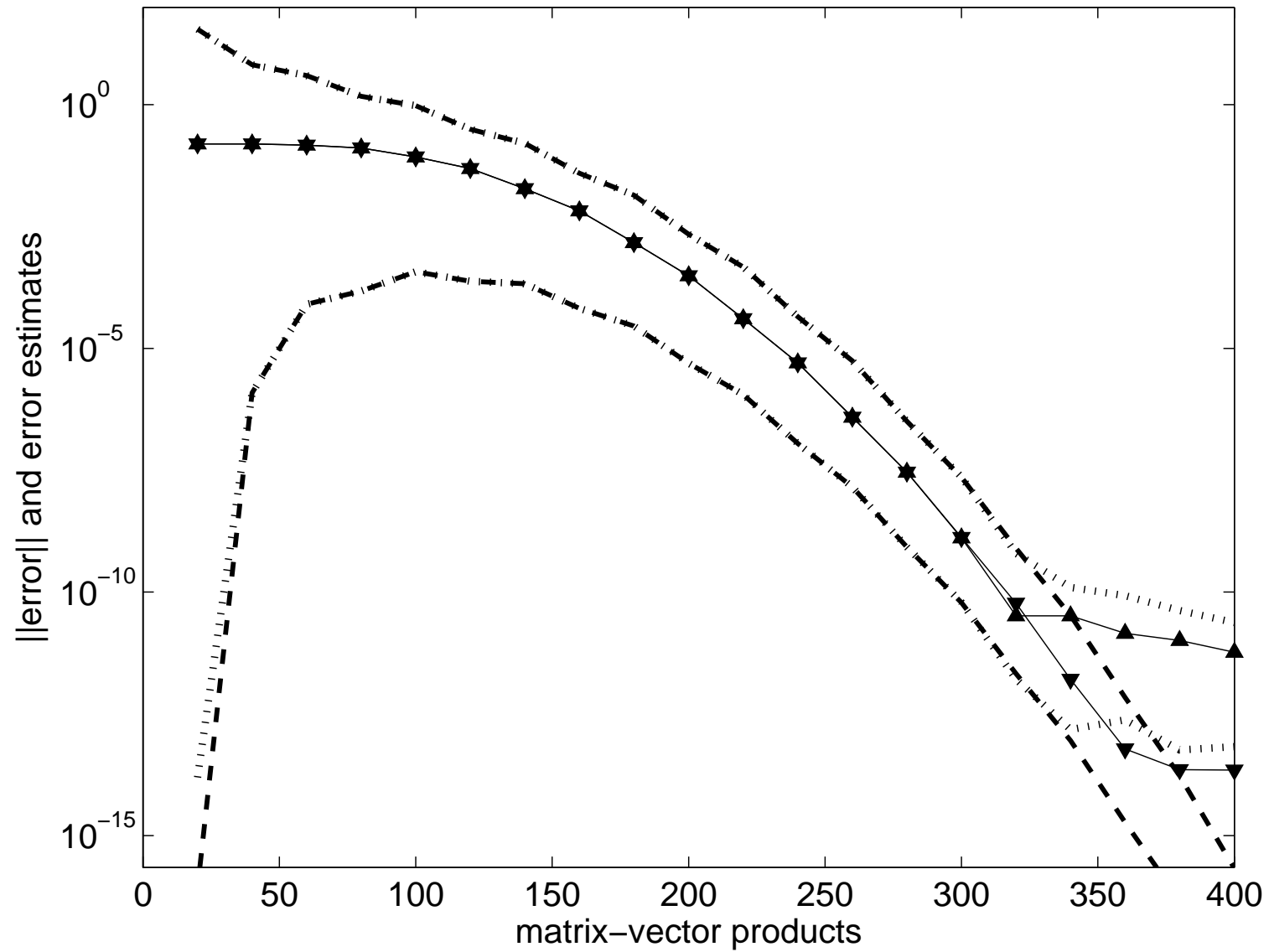
The aim is to choose the restart length large enough such that at the point of transition the desired accuracy is reached.

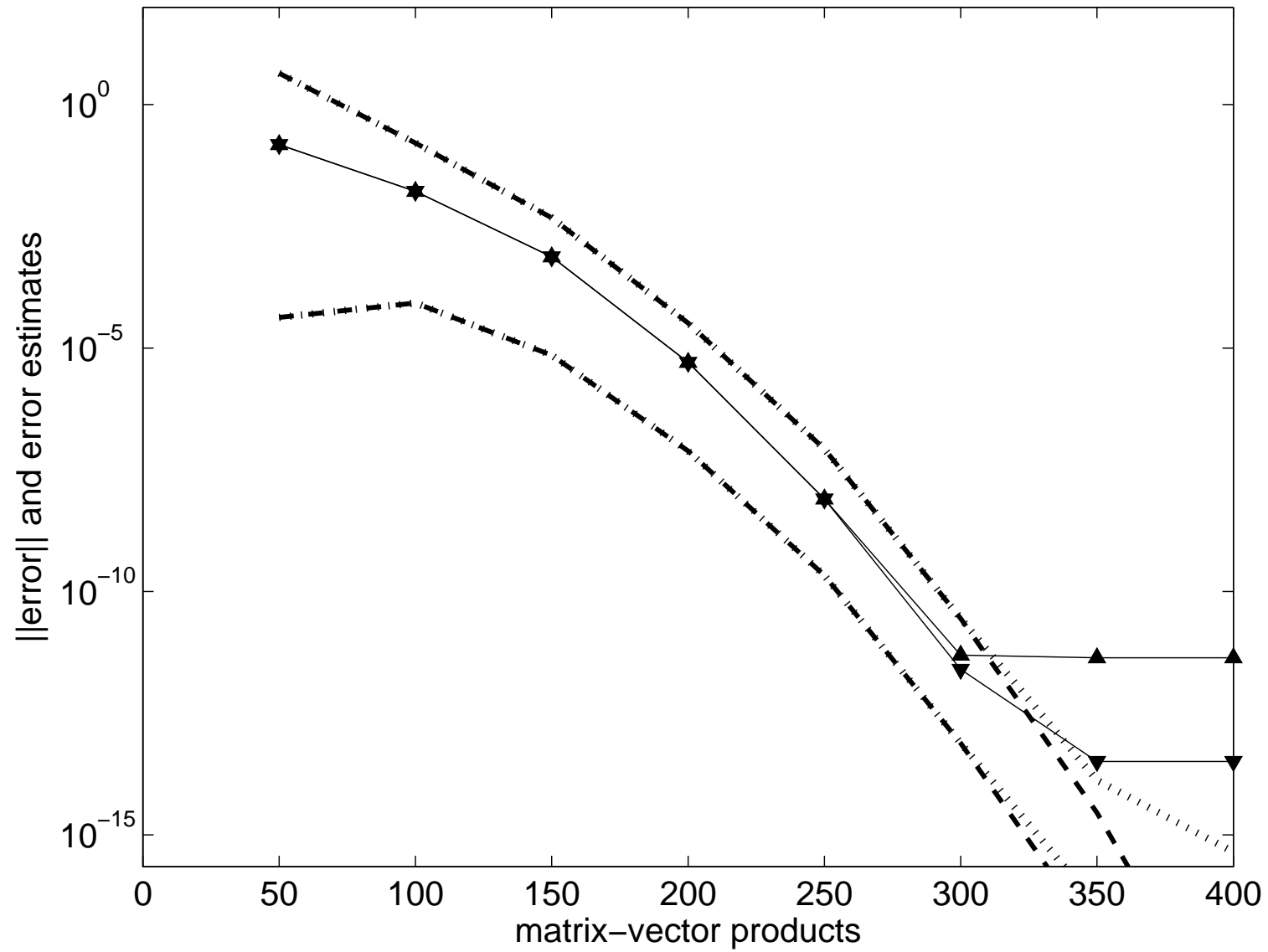


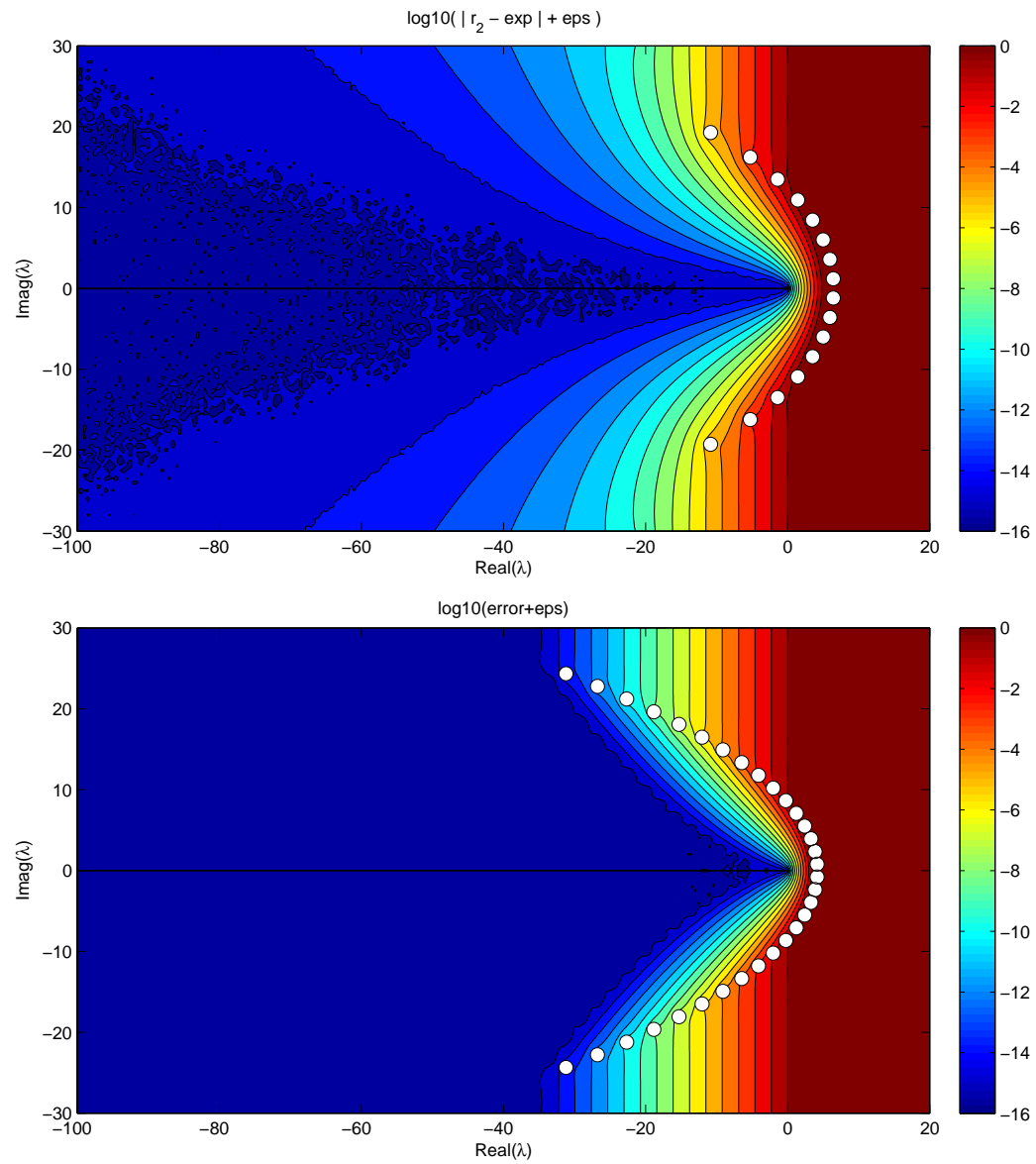
Summary

- Restarted Arnoldi methods result in acceptable storage costs even for very large matrices.
- The new implementation (Algorithm 2) is faster as it solves only a fixed number of linear equations of the size of the restart length m instead of, as is the case with Algorithm 1, evaluating a function of a matrix of increasing size km in the k -th cycle.
- The method converges if the restart length is sufficiently large. The asymptotic convergence behavior is (nearly) understood.
- Stopping criteria are available (Gauss-Radau and Gauss-Lobatto quadrature [Golub & Meurant, 1994] based on generalization of the ‘extended schemes’ of [Saad, 1992] or [Phillipe & Sidje, 1993]) \Rightarrow see [our paper, 2007]).
- **To do:** Spectral transformation, augmenting, preconditioning, rational approximations suited for other classes of functions and/or matrices (as, e.g., in [Trefethen, Weideman & Schmelzer, 2006]).

Thank you for your attention!







An Advection-Diffusion Problem

$$\partial_t u - \frac{1}{Pe} \Delta u + \mathbf{a} \cdot \nabla u = 0 \quad \text{in } \Omega = (-1, 1) \times (0, 1),$$

$$u = 1 - \tanh(Pe) \quad \text{on } \Gamma_0,$$

$$u = 1 + \tanh((2x + 1)Pe), \quad \text{on } \Gamma_{\text{in}},$$

$$\frac{\partial u}{\partial n} = 0 \quad \text{on } \Gamma_{\text{out}}.$$

$$\mathbf{a}(x, y) = \left[2y(1 - x^2), -2x(1 - y^2) \right], \quad (x, y) \in \Omega$$