LECTURE 8

$$T_m(x) = \sum_{i=0}^{m} \frac{x^i}{i!}$$

$$\tilde{h}_{m+1}(x) = \sum_{k=m+1}^{\infty} |c_k| \, x^k$$

$$\vartheta_m : \quad \tilde{h}_{m+1}(\vartheta_m) = tol \cdot \vartheta_m$$

Choose s such that $\frac{\|A\|_1}{2^s} \leq \vartheta_m \Rightarrow \frac{\|\Delta A\|_1}{\|A\|_1} \leq tol$

$$e^A \approx \left(T_m\left(\frac{A}{2^s}\right)\right)^{2^s}$$

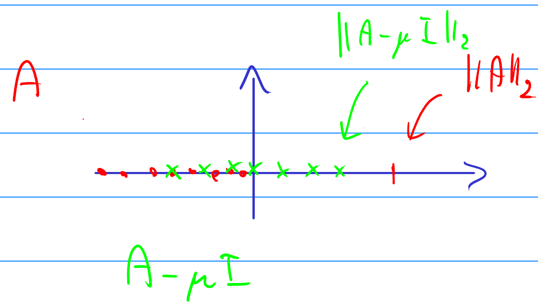
Algorithm not iterative, direct

\+ EARLY TERMINATION

PRECONDITIONING $\quad A \to A - \mu I \; : \; \|A - \mu I\| \leq \|A\|$

$$\mu = \frac{\mathrm{trace}(A)}{N} \qquad \mathrm{trace}(A) = \sum \text{ of eigenvalues}$$

EXAMPLE $\quad A \approx \mathcal{J}_{xx}$



$$e^A = e^{\mu} e^{A-\mu I}$$

SIMILAR REASONING FOR OTHER APPROXIMATIONS $\quad R_{m,m}$ (Padé)

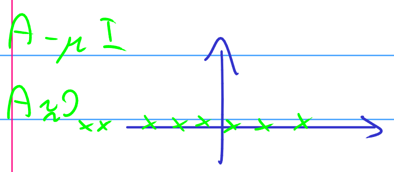
Padé $R_{m,m}$, scaling and squaring, Backward Error is done MATLAB for A not hermitian

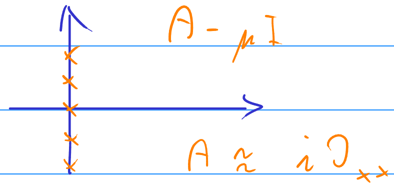
edit expm.m

OTHER APPROXIMATIONS: interpolations (Leja, Chebyshev)

$L_{m,c} : [-c, c] \to \mathbb{R}$ which interpolates

e^x in $[-c,c]$ $\quad (c \in \mathbb{R}, \quad c \in i\mathbb{R})$





$\vartheta_{m,c} : \tilde{h}(\vartheta_{m,c}) = \text{tol} \cdot \vartheta_{m,c}$ For given c, m

$$\left.\begin{array}{l} \vartheta_m := \max\limits_{\substack{0 \leq c \leq c^* \\ \text{TAYLOR}}} \{\vartheta_{m,c}\} \\ c_m := \arg\max\limits_{0 \leq c \leq c^*} \{\vartheta_{m,c}\} \end{array}\right\} L_{m,c_m}(x)$$

EARLY TERMINATION $\Leftrightarrow$ from degree 0 up to m (at most)

NEWTON INTERPOLATION

WHAT ABOUT $\varphi_1(x)$

$h_{m,1}(x) := \log(e^{-x} T_m(x))$ for $T_m(x) \approx e^x$

$$T_m(x) = e^{x + h_{m+1}(x)}$$

People developed QUASI - Backward Error Analysis

$$(x + \cancel{\Delta}) T_{m,1}(x) = e^{x+\Delta} - I$$

$\varphi_1(x)$

For $\varphi_1(x)$ it does not exist a simple ?
shift strategy : $\varphi_1(A-\mu I) \Rightarrow \varphi_1(A)$

SCALING AND SQUARING FOR $\varphi_1(x)$

1) $\varphi_1(x) = \frac{1}{2}\left(e^{x/2}+1\right)\varphi_1\left(\frac{x}{2}\right)$

2) $\varphi_1(x) = \left(\frac{1}{2}\boxed{\frac{x}{2}}\varphi_1\left(\frac{x}{2}\right)+1\right)\varphi_1\left(\frac{x}{2}\right)$

$x = A \approx \partial_{xx}$ $\quad A = \frac{1}{h^2}\begin{bmatrix} -2 & 1 & 0 & \\ 1 & -2 & 1 & \\ & & \ddots & \\ & & & \ddots \end{bmatrix}$

A has large norm.

DON'T USE 2)

$\varphi_1(A), e^A$ have small norm

COMPUTING THE ACTION $e^A v$

$\underbrace{V^{-1} e^{\Lambda} V}\, v$ DIAGONALIZATION: no gain by V

MATRIX FREE

TAYLOR : $e^A v \approx v + Av + \frac{A(Av)}{2} + \frac{A(A(Av))}{6} + \dots$

MATRIX FREE

INTERPOLATION: $e^{Av} \approx d_0 v + d_1 \underline{(A-\xi_0 I)v} + d_2 (A-\xi_1 I)\underline{(A-\xi_0 I)v}$
$+ \dots$

$\{d_0, d_1, \dots\}$ divided differences
$\{\xi_0, \xi_1, \dots\}$ interpolation nodes

$(A-\xi_0 I)v = Av - \xi_0 v$

MATRIX FREE

PADÈ : $\left(I-\frac{A}{2}\right)^{-1}\left[\left(I+\frac{A}{2}\right)v\right]$

MATRIX FREE $\quad J_F(x)v \approx \frac{F(x+\varepsilon v)-F(x)}{\varepsilon}$ $\quad y'(t) = F(y(t))$

$$e^{A} v = \underbrace{e^{\frac{A}{s}} e^{\frac{A}{s}} \cdots e^{\frac{A}{s}}}_{s \text{ times}} v$$

$$\underbrace{e^{\frac{A}{s}} v}_{v_1} = v + \frac{A}{s} v + \frac{1}{2}\frac{A}{s}\left(\frac{A}{s} v\right) + \dots + \frac{1}{m!}\frac{A}{s}\left(\cdots v\right)$$

$$\underbrace{e^{\frac{A}{s}} v_1}_{v_2} = \dots$$

$$e^{\frac{A}{s}} v_2 = \dots$$

NOT MATRIX FREE

$$\frac{\tilde{h}_{m+1}\left(\|s^{-1}A\|\right)}{\|s^{-1}A\|} \leq \text{tol} \implies (s, m)$$

$\|A\|$ without explicit A is HARD

BEA is independent of v

EARLY TERMINATION depends on v

expmv.m MATLAB $\geq$ 2023b

WHAT ABOUT $\varphi_1(A)v$

$$\begin{cases} y'(t) = Ay(t) + v \\ y(0) = 0 \end{cases} \qquad y(1) = \varphi_1(A)v$$

sub stepping

$$y(s^{-1}) = \underline{s^{-1}\varphi_1(s^{-1}A)v}$$

$$\begin{cases} y'(t) = Ay(t) + v \\ y(s^{-1}) = s^{-1}\varphi_1(s^{-1}A)v \end{cases}$$

$$y(2s^{-1}) = e^{s^{-1}A} s^{-1}\varphi_1(s^{-1}A)v + s^{-1}\varphi_1(s^{-1}A)v =$$

$$\underline{s^{-1}\varphi_1(s^{-1}A)v} + s^{-1}\varphi_1(s^{-1}A)\left(A\underline{s^{-1}\varphi_1(s^{-1}A)v} + v\right)$$

ANOTHER POSSIBILITY

AUGMENTED MATRIX

$$\exp\left(\begin{bmatrix} A & v \\ 0 & 0 \end{bmatrix}\right) = \begin{bmatrix} I & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} A & v \\ 0 & 0 \end{bmatrix} + \frac{1}{2}\begin{bmatrix} A^2 & Av \\ 0 & 0 \end{bmatrix} + \frac{1}{6}\begin{bmatrix} A^3 & A^2 v \\ 0 & 0 \end{bmatrix}$$

$$\underset{\mathbb{C}^{N+1\times N+1}}{\cap} \qquad + \frac{1}{24}\begin{bmatrix} A^4 & A^3 v \\ 0 & 0 \end{bmatrix} + \dots$$

$$\begin{bmatrix} e^{A} & \varphi_1(A)v \\ 0 & 1 \end{bmatrix}$$

$$\exp\Big(\underbrace{\begin{bmatrix} A & v \\ 0 & 0 \end{bmatrix}}_{=\hat{A}}\Big)\begin{bmatrix} u_0 \\ 1 \end{bmatrix} = \begin{bmatrix} e^{A}u_0 + \varphi_1(A)v \\ * \end{bmatrix}$$

We can use BEA, shifting, ..., for $\exp(\hat{A})$

We have something similar with Euler-Rosenbrock for non-autonomous systems

identity of size $p-1$

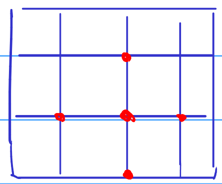
$$\hat{A} = \begin{bmatrix} A & u_p & u_{p-1} & \dots & u_1 \\ 0 & & J & & \end{bmatrix} \qquad J = \begin{bmatrix} 0 & I_{p-1} \\ 0 & 0 \end{bmatrix}$$

$$\exp(\tau\hat{A})\begin{bmatrix} u_0 \\ e_p \end{bmatrix} = \begin{bmatrix} e^{\tau A}u_0 + \tau\varphi_1(\tau A)u_1 + \overset{\tau}{\tau^2}\varphi_2(\tau A)\overset{\tau\cdot u_2}{u_2} + \dots + \tau^p\varphi_p(\tau A)u_p \\ * \end{bmatrix}$$

$$e_p = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \Big\} p-1$$

ODE:	IMPLICIT METHODS	EXPONENTIAL METHODS
LINEAR ALGEBRA:	$(I-\tau A)^{-1}v$	$\exp(\tau A)v$

$A \approx \partial_{xx} + \partial_{yy}$ with FINITE DIFFERENCES



$A \in \mathbb{C}^{N \times N}$ is sperse if $nnz(A) = \mathcal{O}(N)$

$\underset{\mathbb{R}^{m_1 \times m_1}}{\underset{\cap}{A_1}} \approx \partial_{xx}$ $\qquad$ $\underset{\mathbb{R}^{m_2 \times m_2}}{\underset{\cap}{A_2}} \approx \partial_{yy}$

$$I_{m_2} \otimes A_1 + A_2 \otimes I_{m_1} = A \approx \partial_{xx} + \partial_{yy}$$