The Leja method: backward error analysis and implementation

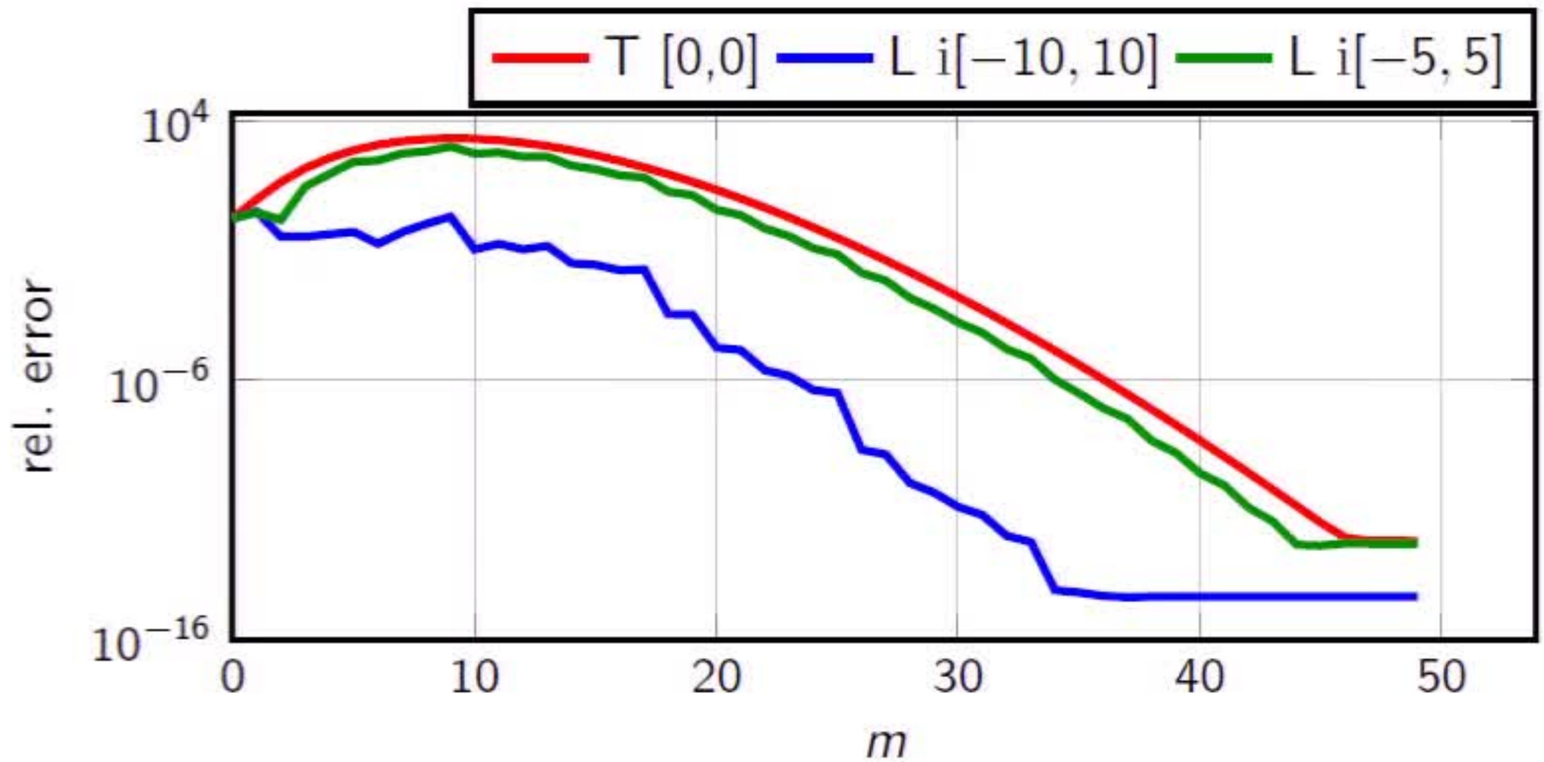
Peter Kandolf

Recipient of a DOC Fellowship of the Austrian Academy of Science at the Department of Mathematics, University of Innsbruck, Austria

joint work with M. Caliari, A. Ostermann, S. Rainer

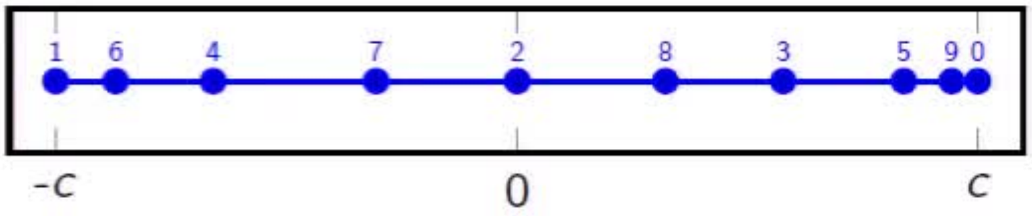
SIAM Conference on Applied Linear Algebra 2015
October 26, 2015

Schrödinger equation in 3D



Leja interpolation

Interpolation points



$$\xi_j \in \arg\max_{\xi \in [-c,c]} \prod_{i=0}^{j-1} |\xi - \xi_i|$$

Interpolation polynomial $L_{m,c}(x) \approx \mathrm{e}^x$ used here for

$$L_{m,c}(A)v \approx \mathrm{e}^A v,$$

costs dominated by matrix-vector products.

Theory 2

A power-series expansion of $\Delta A = h_{m+1,c}(A)$ has the bound

$$\|h_{m+1,c}(A)\| = \|\sum_{k=1}^{\infty} a_{k,c} A^k\| \leq \sum_{k=1}^{\infty} |a_{k,c}| \|A\|^k =: \tilde{h}_{m+1,c}(\|A\|).$$

We can compute

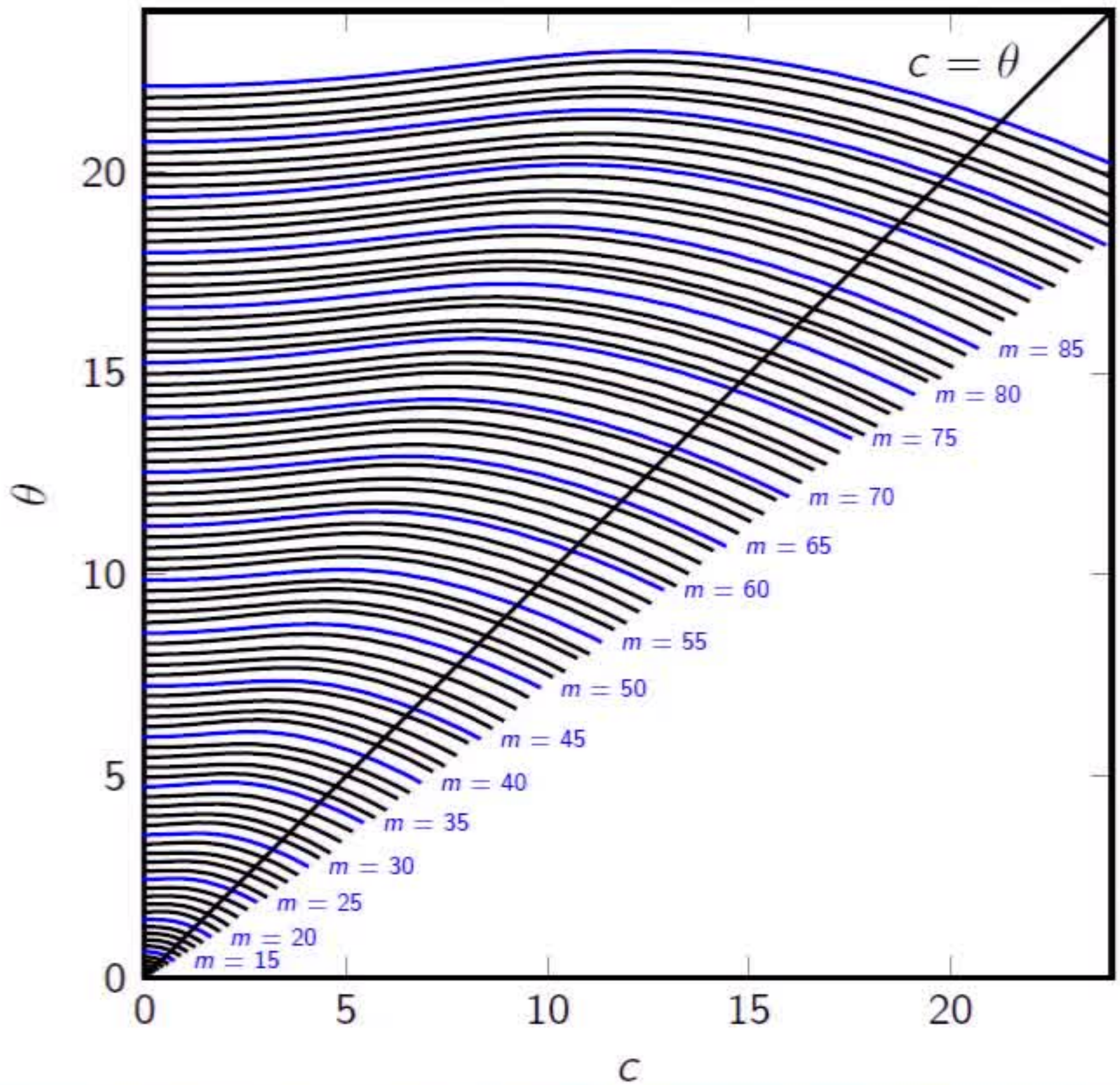
$$\theta_{m,c} := \text{unique positive root of } \frac{\tilde{h}_{m+1,c}(\theta)}{\theta} = \mathrm{tol}.$$

in high accuracy.

---

compare Al-Mohy and Higham (2011)

Theory 2



c = θ
m = 85
m = 80
m = 75
m = 70
m = 65
m = 60
m = 55
m = 50
m = 45
m = 40
m = 35
m = 30
m = 25
m = 20
m = 15
θ
c
0
5
10
15
20

Theory 3

Computation

Select m_* and s such that

$$L_{m_*}(s^{-1}A)^s = \mathrm{e}^{A+\Delta A} \quad \text{with } \|\Delta A\| \leq \mathrm{tol}\|A\|$$

is computed with the least matrix-vector products.

Total cost for interpolation polynomial of degree m

$$C_m(A) = ms = m\left\lceil \|A\|/\theta_m \right\rceil .$$

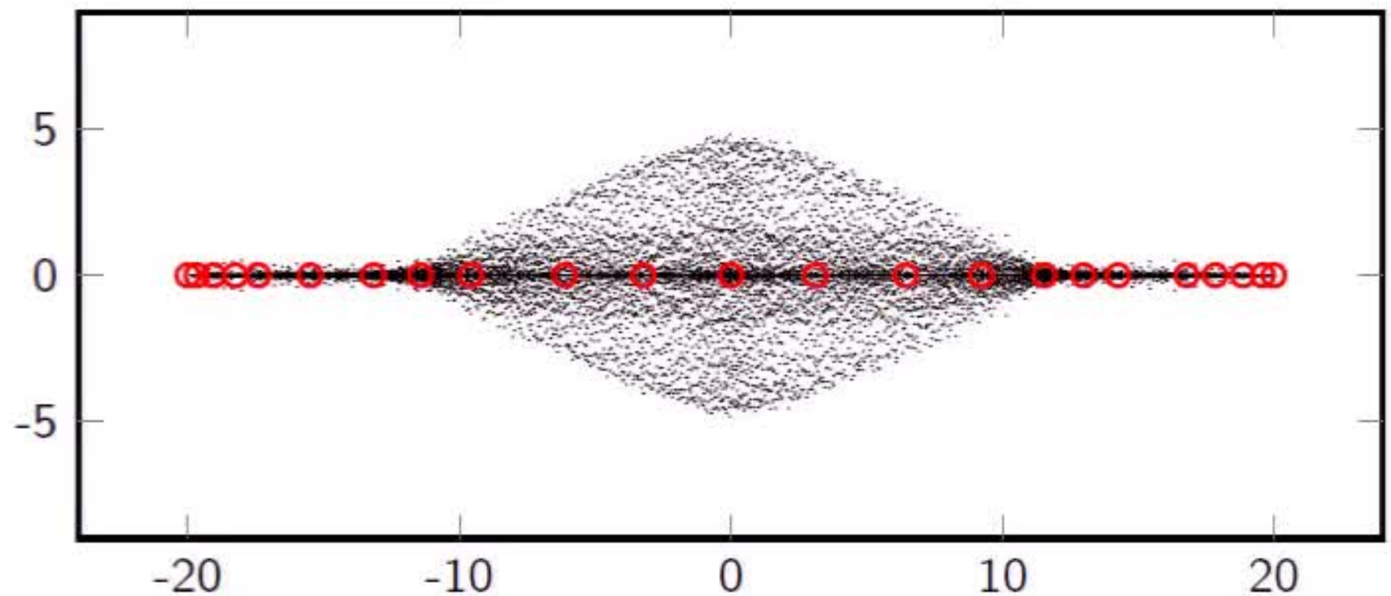
Optimal choice:

$$m_* = \underset{2\leq m \leq m_{max}}{\arg\min} \left\{ m\left\lceil \|A\|/\theta_m \right\rceil \right\} \quad s = \left\lceil \|A\|/\theta_{m_*} \right\rceil .$$

compare Al-Mohy and Higham (2011)

Algorithm - Input A, v, tol

$$\frac{1}{20}(A - \mu I)$$



Compute

$$\left[\mathrm{e}^{\mu/s} L_{98}(s^{-1}(A - \mu I))\right]^s$$

by Newton interpolation with Leja points in $[-\theta_{98}, \theta_{98}]$.