On the Newton method for the matrix pth root

Bruno Iannazzo

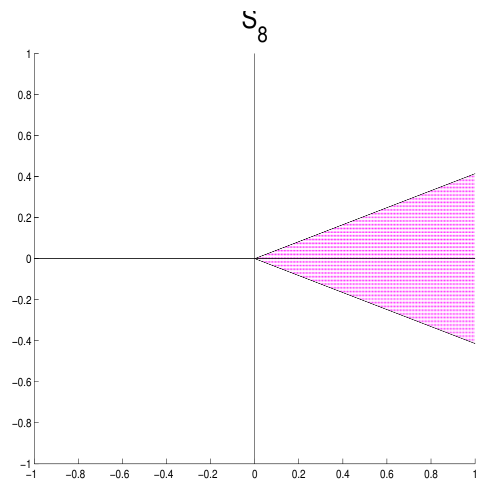
Dipartimento di Matematica

Università di Pisa

Problem

Given a matrix A, with no nonpositive real eigenvalues, $p > 1$ integer, find a solution of the equation $X^p - A = 0$ with eigenvalues in the sector

$$\mathcal{S}_p = \{z \in \mathbb{C} : -\pi/p < \arg(z) < \pi/p\}.$$



Existence and uniqueness are guaranteed. We call this solution $X = A^{1/p}$ principal pth root of A.

Available methods

- Schur decomposition method [Björk & Hammarling '83, Smith '03]
  extremely good accuracy, high cost $O(n^3 p^2)$.
  For $p = 2^q$ taking q times square roots requires
  $O(n^3 \log p)$ operations!
  $\rightarrow$ A desirable cost is $O(n^3 \log p)$.
- Newton's method
  advantage: low cost, $O(n^3 \log p)$ per step, local
  quadratical convergence
  drawbacks: instability and lack of global convergence.

Task

Removing these drawbacks

Newton's Method

Applying Newton's method to the equation

$$X^p - A = 0,$$

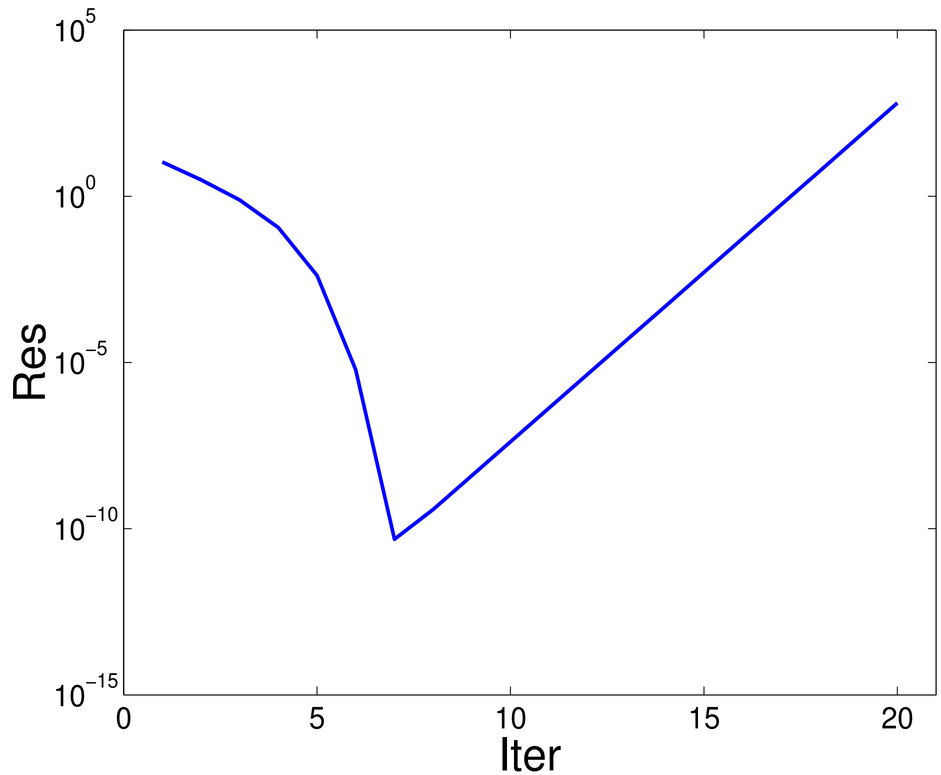
with an initial value X_0 which commutes with A, one obtains

$$X_{k+1} = \frac{(p-1)X_k + AX_k^{1-p}}{p}$$

which generalizes the scalar iteration.

Instability of the Newton method

Let us consider a well-conditioned problem.
Compute the 4th root of a 3×3 matrix having eigenvalues $\sigma(A) = \{0.1, 1, 10\}$



Nature of the instability

The Newton iteration suffers from instability near the solution

Def. An iteration $X_{k+1} = f(X_k)$ is stable in a neighborhood of the solution $X = f(X)$ if the error matrices $E_k = X_k - X$ satisfy

$$E_{k+1} = L(E_k) + O(\|E_k\|^2)$$

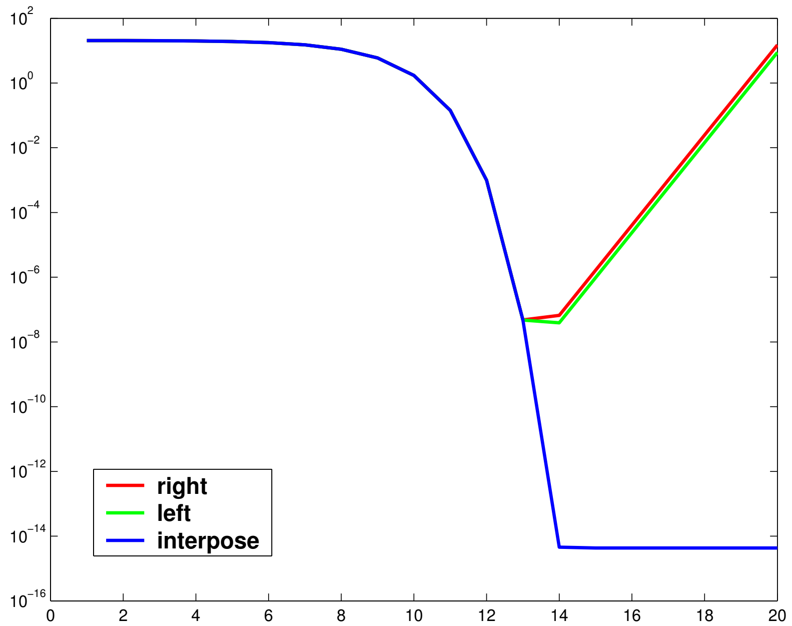
where L is linear and has bounded powers.

Reason: lack of commutativity.

Illustration of instability

Numerically different iterations for the inverse of a matrix

- $X_{k+1} = 2X_k - X_k^2 A, \quad X_{k+1} = 2X_k - AX_k^2$ (unstable)
- $X_{k+1} = 2X_k - X_k A X_k$ (stable)



Interposition makes iterations stable!

It reduces effects of numerical noncommutativity.

Question

How to get rid of monolateral multiplication by A in the Newton iteration?

$$X_{k+1} = \frac{(p-1)X_k + AX_k^p}{p}$$

Solution: implicitly found on the known stable square root algorithms.

Stable algorithms for the square root

$$X_{k+1} = \frac{X_k + AX_k^{-1}}{2}$$

$$\swarrow \qquad\qquad\qquad \searrow$$

$$Y_k = A^{-1}X_k \qquad\qquad H_k = \frac{AX_k^{-1} - X_k}{2}$$

$$\downarrow \qquad\qquad\qquad \downarrow$$

$$X_{k+1} = \frac{X_k + Y_k^{-1}}{2} \qquad\qquad X_{k+1} = X_k + H_k$$

$$Y_{k+1} = \frac{Y_k + X_k^{-1}}{2} \qquad\qquad H_{k+1} = -\frac{H_k X_{k+1}^{-1} H_k}{2}$$

Matrix sign function [Denman-Beavers]

Graeffe's iteration [B. Meini]

Generalization to $p > 2$

$$X_{k+1} = \frac{X_k + AX_k^{1-p}}{p}$$

$$\downarrow$$

$$Y_k = A^{-1} X_k^{p-1}$$

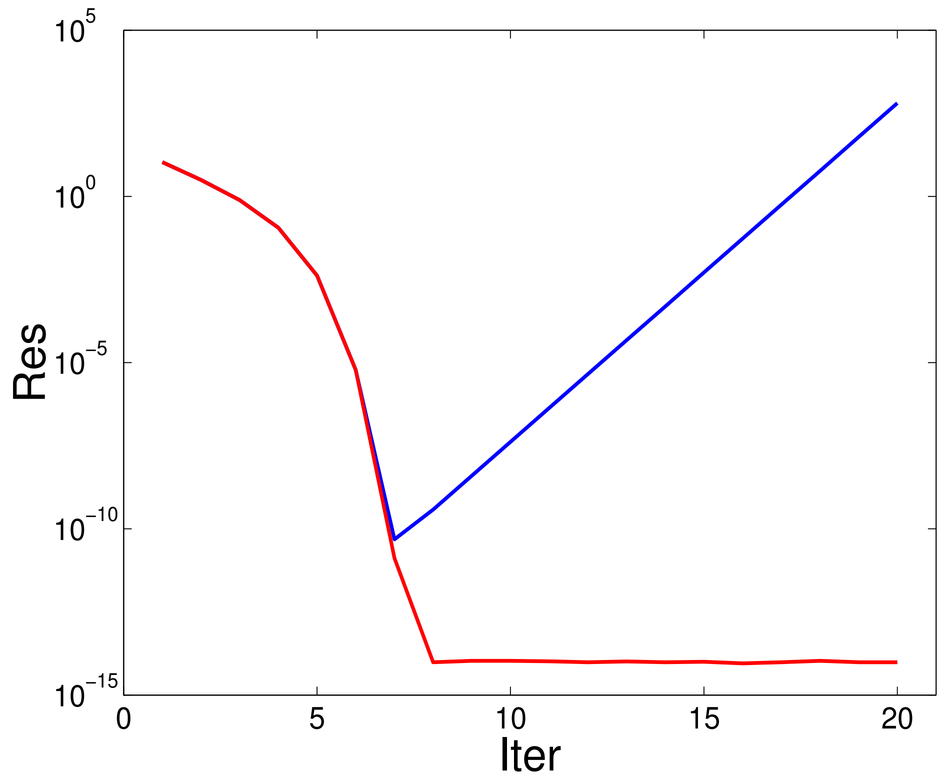
$$\downarrow$$

$$\begin{cases} X_{k+1} = \dfrac{(p-1)X_k + Y_k^{-1}}{p} \\ Y_{k+1} = \left(\dfrac{(p-1)Y_k + X_k^{-1}}{p} Y_k^{-1} \right)^{p-2} \dfrac{(p-1)Y_k + X_k^{-1}}{p} \end{cases}$$

The iteration can be implemented with $O(\log p)$ matrix ops.

Stability

Theorem. The iteration is stable in a neighborhood of the solution.
Example. The iteration provides a stable algorithm for computing the matrix pth root



Convergence

"The problem is to determine the region of the plane, such that P [initial point] being taken at pleasure anywhere within one region we arrive ultimately at the point A [a solution]"

Arthur Cayley, 1879

"J'espére appliquer cette théorie au cas d'une equation cubique, mais les calculs sont beaucoup plus difficiles"

Arthur Cayley, 1890

"Donc, en general, la division du plan en régions, qui conduisent chacune à une racine déterminée de $f(z) = 0$, sera un problème impraticable.
Voilà la raison de l'échec de la tentative de Cayley"

Gaston Julia, 1918

Choice of the initial value

The initial value must

- Commute with A
- Converge to the principal pth root

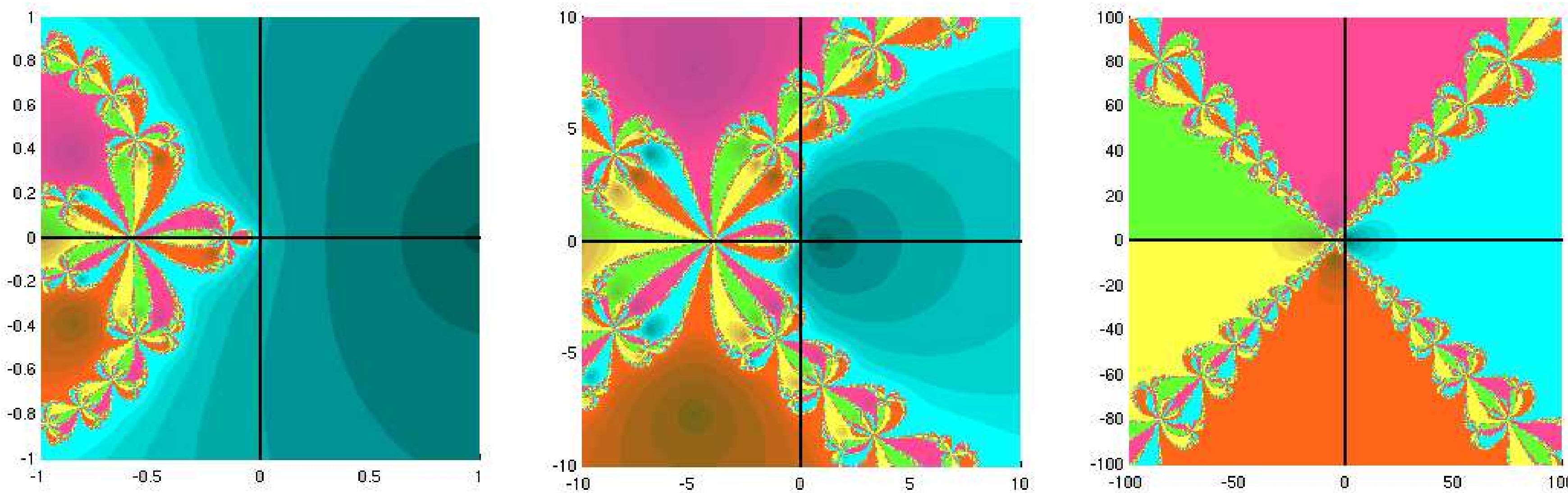
A nice choice is $X_0 = I$
The problem of the convergence can be reduced to the scalar iteration

$$\begin{cases} x_{k+1} = \dfrac{(p-1)x_k + \lambda x_k^{(1-p)}}{p} \\ x_0 = 1 \end{cases}$$

with λ eigenvalue of A.

Question

- Which is the set $\mathcal{B}_p$ of λ for which the scalar iteration converges to the principal pth root $\lambda^{1/p}$?
- A set with fractal boundary.



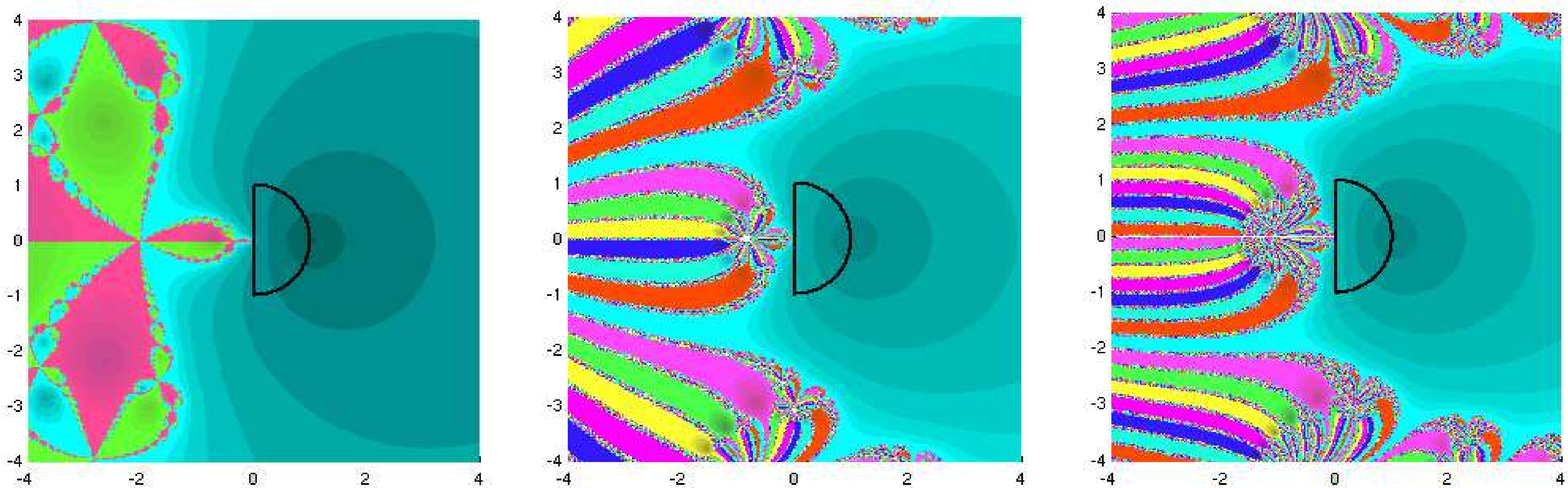
in blue color the set of complex numbers λ for which the sequence converges to the principal root $\lambda^{1/p}$
in red color the ones that generate sequences converging to secondary roots $\omega\lambda^{1/p}$ and so on...

Main theorem

The set

$$\mathcal{D} = \{z \in \mathbb{C}, |z| < 1, \mathrm{Re} z > 0\}$$

is such that $\mathcal{D} \subset \mathcal{B}_p$ for any p.
Therefore convergence occurs for any matrix having eigenvalues in $\mathcal{D}$.



The algorithm

- Compute B, the principal square root of A
- Normalize: $C = B/\|B\|$ so that
  the matrix C has eigenvalues in the set D of convergence
- By means of the iteration proposed:
  - If p is odd compute the $(p/2)$th root of C and set $X = C^{2/p} \cdot \|B\|^{2/p}$
  - If p is even compute the pth root of C and set $X = \left(C^{1/p} \cdot \|B\|^{1/p}\right)^2$

Convergence is guaranteed for any matrix having a principal pth root!

Further results

- High order rational iterations (König, Halley, Schröder). The behavior and techniques are similar. Some of them have very nice convergence regions (Halley's method).
- Scaling to reduce number of steps.
  It is possible to provide a scaling to reduce the number of steps to a fixed value. Proving this is work in progress.

Conclusions

- We have presented stable iterations for the pth root
- Their cost is $O(n^3 \log p)$
- Convergence ensured whenever a solution exists

Further developments

- Proving convergence for the Halley's method and designing a rigorous scaling procedure