Numerical methods for matrix functions

SF2524 - Matrix Computations for Large-scale Systems

Lecture 13

Reading material

- Lecture notes online "Numerical methods for matrix functions"
- (Further reading: Nicholas Higham - Functions of Matrices)
- (Further reading: Golub and Van Loan - Matrix computations)

Agenda Block D Matrix functions

- Lecture 13: Defintions
- Lecture 13: General methods
- Lecture 14: Matrix exponential (underlying `expm(A)` in matlab)
- Lecture 14: Matrix square root, matrix sign function
- Lecture 15: Krylov methods for $f(A)b$
- Lecture 15: Exponential integrators

Functions of matrices

Matrix functions (or functions of matrices) will in this block refer to a certain class of functions

$$f : \mathbb{C}^{n\times n} \to \mathbb{C}^{n\times n}$$

that are consistent extensions of scalar functions.

Simplest examples

- If $f(t) = b_0 + b_1 t + \cdots + b_m t^m$ it is natural to define

$$f(A) = b_0 I + b_1 A + \cdots + b_m A^m.$$

- If $f(t) = \frac{\alpha+t}{\beta+t}$ it is natural to define

$$f(A) = (\alpha I + A)^{-1}(\beta I + A) = (\beta I + A)(\alpha I + A)^{-1}.$$

Not matrix functions: $f(A) = \det(A)$, $\quad f(A) = \|A\|$

Definitions

Definition encountered in earlier courses (maybe)

Consider an analytic function $f : \mathbb{C} \to \mathbb{C}$, with a Taylor expansion with expansion point $\mu = 0$

$$f(z) = f(0) + \frac{f'(0)}{1!} z + \cdots .$$

The matrix function $f(A)$ is defined as

$$f(A) := \sum_{i=0}^{\infty} \frac{f^{(i)}(0)}{i!} A^i = f(0)I + \frac{f'(0)}{1!} A + \cdots .$$

In this course we are more careful. Essentially equivalent definitions:

- Taylor series: Definition 4.1.1
- Jordan based: Definition 4.1.3
- Cauchy integral: Definition 4.1.4

Applications

The most well-known non-trivial matrix function

Consider the linear autonomous ODE

$$y'(t) = Ay(t), \quad y(0) = y_0$$

The **matrix exponential** (`expm(A)` in matlab) is the function that satisfies

$$y(t) = \exp(tA)y_0$$

More generally, the solution to

$$y'(t) = Ay(t) + f(t)$$

satisfies

$$y(t) = \exp(tA)y_0 + \int_0^t \exp(A(t-s))f(s)\,ds$$

For some problems much better than traditional time-stepping methods.

Trigonometric matrix functions and square roots

Suppose $y(t) \in \mathbb{R}^n$ satisfies

$$y''(x) + Au(x) = 0 \;\; y(0) = y_0, \;\; y'(0) = y_0'.$$

The solution is explicitly given by

$$y(t) = \cos(\sqrt{A}t)y_0 + (\sqrt{A})^{-1}\sin(\sqrt{(A)}t)y_0'.$$

Matrix logarithm in Markov chains

The transition probability matrix $P(t)$ is related to the transition intensity matrix Q with

$$P(t) = \exp(Qt)$$

were Q satisfies certain properties. Inverse problem: Given $P(1)$ is there Q such that the properties are satisfied. Method: Compute

$$Q = \log(P(0))$$

and check properties.

Further applications in

- Solving the Riccati equation (in control theory)
- Study of stability of time-delay systems
- Orthogonal procrustes problems
- Geometric mean
- Numerical methods for differential equations
- $\cdots$

See youtube video from Gene Golub summer school:
https://www.youtube.com/watch?v=UXWMYr0LQAk

Definitions of matrix functions

PDF lecture notes section 4.1

Polynomials

If $p(z) = a_0 + a_1 z + \cdots a_p z^p$, then the matrix function extension is

$$p(A) = a_0 I + a_1 A + \cdots a_p A^p$$

Taylor expansion of scalar function $f(z)$ with expansion point μ

$$f(z) = \sum_{i=0}^{\infty} \frac{f^{(i)}(\mu)}{i!}(z-\mu)^i.$$

Definition (Taylor definition)

The Taylor definition with expansion point $\mu \in \mathbb{C}$ of the matrix function associated with $f(z)$ is given by

$$f(A) = \sum_{i=0}^{\infty} \frac{f^{(i)}(\mu)}{i!}(A-\mu I)^i.$$

When is the infinite sum

$$f(A) = \sum_{i=0}^{\infty} \frac{f^{(i)}(\mu)}{i!}(A - \mu I)^i. \tag{1}$$

finite?

Theorem (Convergence of Taylor definition)

Suppose $f(z)$ is analytic in $\bar{D}(\mu, r)$ and suppose $r > \|A - \mu I\|$. Let $f(A)$ be (1) *and*

$$\gamma := \frac{\|A - \mu I\|}{r} < 1.$$

Then, there exists a constant $C > 0$ independent of N such that

$$\|f(A) - \sum_{i=0}^{N} \frac{f^{(i)}(\mu)}{i!}(A - \mu I)^i\| \leq C\gamma^N \to 0 \text{ as } N \to \infty.$$

Consequence: $f(A)$ finite if f entire function

Proof on black board

Simple properties:

- $f(z) = g(z) + h(z) \Rightarrow f(A) = g(A) + h(A)$
- $f(z) = g(z)h(z) \Rightarrow f(A) = g(A)h(A) = h(A)g(A)$
- $f(V^{-1}XV) = V^{-1}f(X)V \qquad (\star)$
- $f(\begin{bmatrix} t_1 & & \\ & \ddots & \\ & & t_n \end{bmatrix}) = \begin{bmatrix} f(t_1) & & \\ & \ddots & \\ & & f(t_n) \end{bmatrix}$
- $f(\begin{bmatrix} t_1 & \times & \times \\ & \ddots & \times \\ & & t_n \end{bmatrix}) = \begin{bmatrix} f(t_1) & \times & \times \\ & \ddots & \times \\ & & f(t_n) \end{bmatrix}$
- $f(\begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix}) \begin{bmatrix} f(A) & 0 \\ 0 & f(B) \end{bmatrix} \qquad (\star\star)$

Note $g(A)g(B) \neq g(B)g(A)$ unless $AB = BA$

Jordan form definition

Use $(\star)$ with Jordan decomposition $A = VJV^{-1}$:

$$f(A) = f(VJV^{-1}) = Vf(J)V^{-1}$$

Use $(\star\star)$:

$$f(J) = f\left(\begin{bmatrix} J_1 & & \\ & \ddots & \\ & & J_q \end{bmatrix}\right) = \begin{bmatrix} f(J_1) & & \\ & \ddots & \\ & & f(J_q) \end{bmatrix}$$

What is the matrix function of a Jordan block?

$$J_i = \begin{bmatrix} \lambda & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ & & & \lambda \end{bmatrix}$$

Example: $f(J)$

Example in Matlab:

$$A = \begin{bmatrix} s & 1 & 0 \\ & s & 1 \\ & & s \end{bmatrix}$$

and $p(z) = z^4$.For this case we have

$$p(J) = \begin{bmatrix} p(\lambda) & p'(\lambda) & \frac{1}{2}p''(\lambda) \\ 0 & p(\lambda) & p'(\lambda) \\ 0 & 0 & p(\lambda) \end{bmatrix}.$$

Can be formalized (proof in PDF lecture notes)...

Definition (Jordan canonical form (JCF) definition)

Suppose $A \in \mathbb{C}^{n\times n}$ and let X and $J_1, \ldots, J_q$ be the JCF. The JCF-definition of the matrix function $f(A)$ is given by

$$f(A) := X \operatorname{diag}(F_1, \ldots, F_q) X^{-1}, \tag{2}$$

where

$$F_i = f(J_i) := \begin{bmatrix} f(\lambda_i) & \frac{f'(\lambda_i)}{1!} & \cdots & \frac{f^{(n_i-1)}(\lambda_i)}{(n_i-1)!} \\ & \ddots & \ddots & \vdots \\ & & \ddots & \frac{f'(\lambda_i)}{1!} \\ & & & f(\lambda_i) \end{bmatrix} \in \mathbb{C}^{n_i\times n_i}. \tag{3}$$

Show specialization when eigenvalues distinct

Cauchy integral definition

From complex analysis: Cauchy integral formula

$$f(x) = \frac{1}{2i\pi} \oint_\Gamma \frac{f(z)}{z - x} \, dz.$$

where Γ encircles x counter-clockwise. Replace x with A:

Definition (Cauchy integral definition)

Suppose f is analytic inside and on a simple, closed, piecewise-smooth curve Γ, which encloses the eigenvalues of A once counter-clockwise. The Cauchy integral definition of matrix functions is given by

$$f(A) := \frac{1}{2i\pi} \oint_\Gamma f(z)(zI - A)^{-1} \, dz.$$

* example in lecture notes *

Equivalence of definitions

We have learned about

- Definition 1: Taylor definition
- Definition 2: Jordan form definition
- Definition 3: Cauchy integral definition

Slightly different different definition domains.

Theorem (Equivalence of the matrix function definitions)

Suppose f is an entire function and suppose $A \in \mathbb{C}^{n\times n}$. Then, the matrix function definitions are equivalent.

Which definition valid for

$$f(x) = \sqrt{x} \text{ with } A = \begin{bmatrix} 0 & 1 \\ 0 & 4 \end{bmatrix}?$$

and

$$f(x) = \sqrt{x} \text{ with } A = \begin{bmatrix} 3 & 1 \\ 0 & 4 \end{bmatrix}?$$

General methods

PDF lecture notes section 4.2

General methods for matrix functions:

- Today: Truncated Taylor series (4.2.1)
- Today: Eigenvalue-eigenvector approach (4.2.2)
- Today: Schur-Parlett method (4.2.3)
- Lecture 15: Krylov methods for $f(A)b$ (4.4)

Truncated Taylor series (naive approach)

First approach based on truncting Taylor series:

$$f(A) \approx F_N = \sum_{i=0}^{N} \frac{f^{(i)}(\mu)}{i!}(A - \mu I)^i$$

Properties

- Can be very slow if Taylor series converges slowly
- We need $N - 1$ matrix-matrix multiplications. Complexity

$$\mathcal{O}(Nn^3)$$

- We need access to the derivatives

The truncated Taylor series is mostly for theoretical purposes.

Eigenvalue-eigenvector approach

If we have distinct eigenvalues or symmetric matrix:

$$f(A) = V \begin{bmatrix} f(\lambda_1) & & \\ & \ddots & \\ & & f(\lambda_n) \end{bmatrix} V^{-1}$$

where $V = [v_1, \ldots, v_n]$ are the eigenvectors.

Main properties

- Requires computation of eigenvalues and eigenvectors: Complexity essentially $\mathcal{O}(n^3)$
- Requires only the function value in the eigenvalues
- Can be numerically unstable
- If A is symmetric $V^{-1} = V^T$.

Conclusion: Can be used for numerical computations if reliability is not important.

Schur-Parlett method

We know how to compute a Schur factorization

$$A = QTQ^*$$

where Q orthogonal and T upper triangular

$$f(A) = f(QTQ^*) = Qf(T)Q^*.$$

Schur-Parlett method:

- Compute a Schur factorization Q, T
- Compute $f(T)$ where T triangular
- Compute $f(A) = Qf(T)Q^*$.

What is $f(T)$ for a triangular matrix?

$f(T)$ where T triangular

Note: $f(T)$ commutes with T:

$$f(T)T = Tf(T).$$

* On black board: two-by-two example. Generalization derivation *

Theorem (Computation of one element of $f(T)$)

Suppose $T \in \mathbb{C}^{n \times n}$ is an upper triangular matrix with distinct eigenvalues. Then, for any i and any $j > i$,

$$f_{ij} = \frac{s}{t_{jj} - t_{ii}}$$

where

$$s = t_{ij}(f_{jj} - f_{ii}) + \sum_{k=i+1}^{j-1} t_{ik} f_{kj} - f_{ik} t_{kj}.$$

$$F: \quad \begin{array}{cccccccccc}
 & & & & & & & j & \\
 & & & & & & & \downarrow & \\
 & + & + & + & + & + & \square & \square & \square \\
 & 0 & + & + & + & + & + & \square & \square \\
i \rightarrow & 0 & 0 & + & + & + & + & f_{ij} & \square \\
 & 0 & 0 & 0 & + & + & + & + & \square \\
 & 0 & 0 & 0 & 0 & + & + & + & + \\
 & 0 & 0 & 0 & 0 & 0 & + & + & + \\
 & 0 & 0 & 0 & 0 & 0 & 0 & + & + \\
 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & +
\end{array}
\qquad
T: \quad \begin{array}{cccccccccc}
 & & & & & & & j & \\
 & & & & & & & \downarrow & \\
 & + & + & + & + & + & + & + & + \\
 & 0 & + & + & + & + & + & + & + \\
i \rightarrow & 0 & 0 & + & + & + & + & + & + \\
 & 0 & 0 & 0 & + & + & + & + & + \\
 & 0 & 0 & 0 & 0 & + & + & + & + \\
 & 0 & 0 & 0 & 0 & 0 & + & + & + \\
 & 0 & 0 & 0 & 0 & 0 & 0 & + & + \\
 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & +
\end{array}$$

Repeat sub-column by sub-column.

* On blackboard *

* Matlab simulation *

Input: A triangular matrix $T \in \mathbb{C}^{n \times n}$ with distinct eigenvalues
Output: The matrix function $F = f(T)$

for $i = 1, \ldots, n$ **do**
  $f_{ii} = f(t_{i,i})$
end
for $p = 1, \ldots, n-1$ **do**
  for $i = 1, \ldots, n-p$ **do**
    $j = i + p$
    $s = t_{ij}(f_{jj} - f_{ii})$
    for k=i+1,...,j-1 **do**
      $s = s + t_{ik} f_{kj} - f_{ik} t_{kj}$
    end
    $f_{ij} = s/(t_{jj} - t_{ii})$
  end
end

Algorithm 1: Simplified Schur-Parlett method

Main properties Schur-Parlett (simplified)

- Requires the computation of a Schur-decomposition ($\mathcal{O}(n^3)$) which is often the dominating computational cost.
- The only usage of f: $f(\lambda_i)$, $i = 1, \ldots, n$
- Only works when eigenvalues distinct
- Numerical cancellation can occur when eigenvalues close: Can repaired with the full version of Schur-Parlett by using $f^{(i)}(z)$.