The Computation of Matrix Functions in Particular, The Matrix Exponential

By

SYED MUHAMMAD GHUFRAN

A thesis submitted to
The University of Birmingham
for the Degree of
Master of Philosophy

School of Mathematics
The University of Birmingham
October, 2009

Acknowledgements

I am extremely grateful to my supervisor, *Professor Roy Mathias*, for sharing his knowledge, excellent guidance, patience and advice throughout this dissertation. Without his support it would not be possible in short span of time.

I wish to thank *Prof C.Parker , and Mrs J Lowe* for their magnificent support, professional advice and experience to overcome my problems.

I am grateful to my colleagues *Ali Muhammad Farah*, *She Li* and *Dr. Jamal ud Din* for their help and support, and my cousins *Faqir Hussain*, *Faizan ur Rasheed* and my uncle *Haji Aziz* and my close friend *Shah Jehan* for his kindness.

I am also very grateful to my *parents*, *brothers*, *sisters*, my *wife* for their support and trust throughout these long hard times. Further I would like to thank my family and especially to my kids and my nephew *Abdul Muqeet* how missed me a lot.

Abstract

Matrix functions in general are an interesting area in matrix analysis and are used in many areas of linear algebra and arise in numerous applications in science and engineering. We consider how to define matrix functions and how to compute matrix functions. To be concrete, we pay particular attention to the matrix exponential.

The matrix exponential is one of the most important functions of a matrix. In this thesis, we discuss some of the more common matrix functions and their general properties, and we specifically explore the matrix exponential. In principle, there are many different methods to calculate the exponential of a matrix. In practice, some of the methods are preferable to others, but none of which is entirely satisfactory from either a theoretical or a computational point of view. Computations of the matrix exponential using Taylor Series, Padé Approximation, Scaling and Squaring, Eigenvectors, and Schur Decomposition methods are provided.

In this project we checked rate of convergence and accuracy of the matrix exponential.

Keywords : Condition number, Functions of Matrices, Matrix Exponential, roundoff error.

Dedicated to

In Loving Memory of My Father and My Mother

Syed Abdul Ghaffar (Late)
Shereen Taj

Contents

1 Introduction **2**
- 1.1 Overview . . . 2
- 1.2 Special Matrix Definitions . . . 3

2 Norms for vectors and matrices **7**
- 2.1 Vector Norms . . . 7
- 2.2 Rounding Error . . . 13
  - 2.2.1 Absolute and Relative Error . . . 13
- 2.3 Floating Point Arithmetic . . . 14
- 2.4 Matrix Norm . . . 16
  - 2.4.1 Induced Matrix Norm . . . 17
  - 2.4.2 Matrix p-Norm . . . 18
  - 2.4.3 Frobenius Matrix Norm . . . 18
  - 2.4.4 The Euclidean norm or 2-Norm . . . 20
  - 2.4.5 Spectral Norm . . . 20
- 2.5 Convergence of a Matrix Power Series . . . 26
- 2.6 Relation between Norms and the Spectral Radius of a Matrix. 26
- 2.7 Sensitivity of Linear Systems . . . 30
  - 2.7.1 Condition Numbers . . . 30
- 2.8 Eigenvalues and Eigenvectors . . . 40
  - 2.8.1 The characteristic polynomial . . . 41
- 2.9 The Multiplicities of an Eigenvalues . . . 45

3 Matrix Functions **47**
- 3.1 Definitions of $f(A)$ . . . 48
  - 3.1.1 Jordan Canonical Form . . . 48
  - 3.1.2 Definition of a Matrix Function via The Jordan Canonical Form . . . 57
- 3.2 Polynomial Matrix Function . . . 58
  - 3.2.1 Matrix Function via Hermite interpolation . . . 65
- 3.3 Matrix function via Cauchy Integral Formula . . . 69

3.4 Functions of diagonal matrices 71
3.5 Power Series Expansions 73
3.6 The Relationship of the Definitions of Matrix Function 74
3.7 A Schur-Parlett Algorithm for Computation Matrix Functions 86
3.7.1 Parlett's Algorithm 87

4 The Matrix Exponential Theory 93
4.1 Matrix exponential 94
4.2 The Matrix Exponential as a Limit of Powers 100
4.3 The Matrix Exponential via Interpolation 101
4.3.1 Lagrange Interpolation Formula 101
4.3.2 Newton's Divided Difference Interpolation 101
4.4 Additional Theory 103

5 The Matrix Exponential Functions: Algorithms 109
5.1 Series Methods 109
5.1.1 Taylor Series 109
5.1.2 Padé Approximation 128
5.1.3 Scaling and Squaring 135
5.2 Matrix Decomposition Methods 142
5.2.1 Eigenvalue-eigenvector method 142
5.2.2 Schur Parlett Method 147
5.3 Cauchy's integral formula 152

6 Conclusion and Future work 157
6.1 Conclusion 157
6.2 Future work 158

Chapter 1

Introduction

1.1 Overview

In this thesis we are concerned with the numerical computation of the matrix exponential functions e^A, where $A \in \mathbb{M}_n$. The interest in numerical computation of matrix exponential functions is because of its occurrence in the solution of Ordinary Differential Equations. There are many different methods for the computation of matrix exponential functions, like, series method, differential equation methods, polynomial methods and matrix decomposition methods, but none of which is entirely satisfactory from either a theoretical or a computational point of view.

This thesis consists of five chapter.

Chapter 1: Introduction

In this chapter we define some useful definitions from linear algebra, numerical linear algebra and matrix analysis.

Chapter 2: Norms for Vectors and Matrices

The main purpose of this chapter is to introduce norms of vectors and matrices and condition numbers of a matrix and to study perturbation in the linear system of equation $Ax = b$. This will allow us to assess the accuracy of a method for the computation of a function of matrix and the matrix exponential.

Chapter 3: Matrix Functions

In this chapter we discuss the method of how to compute functions of matrices via Jordan Canonical Form, interpolating polynomial, Schur-

Parlett algorithm and the relationship between the different definitions of matrix function.

Chapter 4: The Matrix Exponential Theory

This chapter is devoted to matrix exponential functions and some identities for the next final chapter.

Chapter 5: The Matrix Exponential Functions: Algorithms

The main studies are on the different methods for the computation of matrix exponential via series methods and Matrix Decomposition methods, and their accuracy. We present different examples to that each method has particular strengths and weaknesses. We feel this is more useful then choosing a single example and testing each of the methods on it, since conclusion would depend on the particular example chosen rather than the relative merits of each method. The problem of computing e^A accurately and fast methods is relevant to many problems. We do not consider iterative methods, because they are of a very different character, and in general do not yield the exact exponential.

Chapter 6: Conclusion and Future work

Conclusion of the work and future work be presented in this chapter.

1.2 Special Matrix Definitions

A matrix is an m-by-n array of scalars from a field $\mathbb{F}$. If $m = n$, the matrix is said is said to be square. The set of all m-by-n matrices over $\mathbb{F}$ is denoted by $\mathbb{M}_{m,n}(\mathbb{F})$ and $\mathbb{M}_{n,n}(\mathbb{F})$ is abbreviated to $\mathbb{M}_n(\mathbb{F})$. In the most common case in which $\mathbb{F} = \mathbb{C}$, the complex numbers, $\mathbb{M}_n(\mathbb{C})$ is further abbreviated to $\mathbb{M}_n$, and $\mathbb{M}_{m,n}(\mathbb{C})$. Matrices are usually denoted by capital letters. Throughout the thesis we will require the definition of the following types of matrices.

$\mathbb{C}$ = the set of all complex numbers.
$\mathbb{C}^n$ = the set of all complex n-column vectors.
$\mathbb{C}^{m\times n}$= the set of all complex m×n matrices

- A set of vectors $\{a_1, ..., a_n\}$ in $\mathbb{C}^m$ is *linearly independent* if

$$\sum_{i=0}^{n} \alpha_i a_i = 0 \;\Leftrightarrow\; \alpha_1 = \alpha_2 = ... = \alpha_n = 0.$$

Otherwise, a nontrivial combination of $a_1, ..., a_n$ is zero and $\{a_1, ..., a_n\}$ is said to be *linearly dependent.*

- The *range* of A is defined by

$$R(A) = \{y \in \mathbb{C}^m | y = Ax \ \ for\ some\ x \in \mathbb{C}^n\}$$

and the *null space* of A by

$$N(A) = \{x \in \mathbb{R}^n | Ax = 0\}.$$

If $A = \{a_1, ..., a_n\}$ then

$$R(A) = span\{a_1, ..., a_n\}.$$

- If $A = [a_{ij}] \in \mathbb{M}_{m,n}(\mathbf{F})$, then A^T denotes the *transpose* of A in $\mathbb{M}_{n,m}(\mathbf{F})$ whose entries are a_{ji}; that is, rows are exchanged for columns and vice versa. For example,

$$A^T = \begin{pmatrix} 1 & 2 \\ 3 & 5 \end{pmatrix}^T = \begin{pmatrix} 1 & 3 \\ 2 & 5 \end{pmatrix}$$

- The *rank* of a matrix A is defined by

$$rank(A) = dim[R(A)].$$

A very useful result is that "$rank(A^T) = rank(A)$", and thus the rank of a matrix equals the maximal number of independent rows or columns, i.e., "$row\ rank = column\ rank$."

- $A \in \mathbb{R}^{n \times n}$ is *symmetric* if $A = A^T$,i.e., $a_{ij} = a_{ji}$. Symmetric matrices have the following properties:
  - The eigenvalues are real, not complex.
  - Eigenvectors are orthogonal.
  - Always diagonalizable.
- $A \in \mathbb{R}^{n \times n}$ is said to be *skew-symmetric* if $A = -A^T$
- $A \in \mathbb{C}^{n \times n}$ is *Hermitian* if $A^* = A$ where A^* denotes the conjugate transpose of A.
- $A \in \mathbb{M}^n$ is said to be *skew-Hermitian* if $A^* = -A$

- $A \in \mathbb{C}^{n\times n}$ is diagonal if $a_{ij} = 0$ for $i \neq j$ we use the notation

$$A = diag(x_1, x_2, ...x_n)$$

for a diagonal matrix with $a_{ii} = x_i$ for $i = 1 : n$.

- $A \in \mathbb{C}^{n\times n}$ is upper (lower) triangular if $a_{ij} = 0$ for $i > j(i < j)$. We also say that A is strictly upper(lower) triangular if it is upper (lower) triangular with $a_{ii} = 0$ for $i = 1 : n$.

- $A \in \mathbb{C}^{n\times n}$ is upper (lower) bidiagonal if $a_{ij} = 0$ for $i > j(j > i)$ and $i + 1 < j(j + 1 < i)$.

- $A \in \mathbb{C}^{n\times n}$ is tridiagonal if $a_{ij} = 0$ for $i + 1 < j$ and $j + 1 < i$.

- $A \in \mathbb{C}^{n\times n}$ is upper (lower) Hessenberg if $a_{ij} = 0$ for $i > j + 1$(i<j-1)

- A matrix $A \in \mathbb{M}_n$ is *nilpotent* if $A^k = 0$ for some positive integer k.

- $A \in \mathbb{C}^{n\times n}$ is *positive definite* if $x^*Ax > 0$ for all $x \in \mathbb{C}^n$, $x \neq 0$.

- $A \in \mathbb{M}_n(\mathbb{R})$ is orthogonal if $A^TA = I$ where I is the $n\times n$ identity matrix. with ones on the diagonal and zeros elsewhere. We also say that $A \in \mathbb{C}^{n\times n}$ is unitary if $A^*A = I$.

- A matrix $A \in \mathbb{M}_n$ is defective if it has a defective eigenvalue, or, equivalently, if it does not have a complete set of linearly independent eigenvectors. An example of a defective matrix is

$$A = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$$

- *Diagonalizable.* If $A \in \mathbb{C}^{n\times n}$ is nondefective if and only if there exists a nonsingular $X \in \mathbb{C}^{n\times n}$ such that

$$X^{-1}AX = D = diag(\lambda_1, ..., \lambda_n).$$

- A matrix $B \in \mathbb{M}_n$ is said to be *similar* to a matrix $A \in \mathbb{M}_n$ if there exists a nonsigular matrix $S \in \mathbb{M}_n$ such that

$$B = S^{-1}AS$$

The transformation $A \to S^{-1}AS$ is called a *similarity transformation* by the *similarity matrix* S. The relation "B is similar to A" is sometimes abbreviated $B \sim A$.

- *Schur Decomposition.* If $A \in \mathbb{C}^{n\times n}$ then there exists a unitary $U \in \mathbb{C}^{n\times n}$ such that

$$U^* AU = T = D + N$$

where D=diag$(\lambda_1, \lambda_2, ..., \lambda_n)$ and $N \in \mathbb{C}^{n\times n}$ is strictly upper triangular. furthermore U can be chosen so that the eigenvalues λ_i appear in any order along the diagonal.

- *Real Schur Decomposition.* If $A \in \mathbb{R}^{n\times n}$ then there exists an orthogonal $U \in \mathbb{R}^{n\times n}$ such that

$$U^T AU = R$$

where

$$R = \begin{pmatrix} R_{11} & R_{12} & \cdots & R_{1m} \\ 0 & R_{22} & \cdots & R_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & R_{mn} \end{pmatrix}$$

each R_{ii} is either 1×1 or 2×2 matrix having complex conjugate eigenvalues.

- *Kronecker product.* The *Kronecker product* of $A = [a_{ij}] \in \mathbb{M}_{mn}$ and $B = [b_{ij}] \in \mathbb{M}_{pq}$ is denoted by $A \otimes B$ and is defined to be the block matrix

$$A \otimes B \equiv \begin{pmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{pmatrix} \in \mathbb{M}_{mp,nq}$$

Notice that $A \otimes B \neq B \otimes A$ in general.

Chapter 2

Norms for vectors and matrices

In order to study the effects of perturbations or error analysis in vectors and matrices, we need to measure the 'size' of the errors in a vector or a matrix. A norm can tell us which vector or matrix is 'smaller' or 'larger'. There are two common kinds of error analysis in numerical linear algebra: componentwise and normwise. In general, normwise error analysis is easier (but less precise). For this purpose we need to introduce vector and matrix norms, which provide a way to measure the distance between vectors and matrices. They also provide a measure of "closeness" that is used to define convergence.

Any norm can be used to measure the length or magnitude (in a generalized sense) of vectors in $\mathbb{R}^n$ (or $\mathbb{C}^n$). In other words we think of $\|x\|$ as the (generalized) length of x. The (generalized) distance between two vectors x and y is defined to be $\|x - y\|$. Throughout this chapter we shall consider real or complex vector space only. All of the major results hold for both fields, but within each result one must be consistent as to which field is used. Thus, we shall often state results in terms of a field $\mathbb{F}$ (with $\mathbb{F} = \mathbb{R}$ *or* $\mathbb{C}$ at the outset) and then refer to the same field $\mathbb{F}$ in the rest of the argument.

2.1 Vector Norms

Definition 1. A *norm* on $\mathbb{C}^n$ is a *nonnegative* real-valued function, $\|\cdot\| : \mathbb{C}^n \longrightarrow \mathbb{R}$ satisfying

- Positive definite property:
  $\|x\| \geqslant 0 \ \forall\ x \in \mathbb{C}^n$ and $\|x\| = 0 \Leftrightarrow x = 0$,
- Absolute homogeneity:
  $\|\alpha x\| = \mid \alpha \mid \|x\| \ \ \forall, \alpha \in \mathbb{C}, x \in \mathbb{C}^n$,

- triangle inequality:
  $\|x+y\| \leqslant \|x\| + \|y\|, \ \forall, \ x, y \in \mathbb{C}^n,$

The function value $\|x\|$ is called the norm of x.
Note that $\|0\|=0$, since $0=\|0 \cdot 0\|=|0|\|0\|=0$

Remarks 1.

The definition of a norm on $\mathbb{C}^n$ also defines a norm on $\mathbb{R}^n$ with $\mathbb{C}$ replaced by $\mathbb{R}$, but not vice versa.

Example 1. *Let* $x = [x_1.....x_n]^T$

- l_p, *for* $1 \leqslant p < \infty$*:*

$$\|x\|_p = (|x_1|^p + |x_2|^p + |x_3|^p + ... + |x_n|^p)^{\frac{1}{p}}$$

$$\|x\|_p = (\sum_{i=1}^{n} |x_i|^p)^{\frac{1}{p}}), \quad \textit{for all } x \in \mathbb{C}^n.$$

For $p = 1, 2, \infty$ *we have;*

- l_1*=the 1-norm*

$$\|x\|_1 = \sum_{i=1}^{n} |x_i| \qquad \textit{for all } x \in \mathbb{C}^n$$

- l_2*=the 2-norm (Euclidean norm):*

$$\|x\|_2 = \sqrt{\sum_{i=1}^{n} |x_i|^2} \qquad \forall \ x \in \mathbb{C}^n$$

Note. This is equal to $\sqrt{x^T x}$ *whenever* $x \in \mathbb{R}^n$ *and* $\sqrt{x^* x}$ *whenever* $x \in \mathbb{C}^n$*. The 2-norm is special because its value doesn't change under orthogonal(unitary) transformations.*
For unitary Q *we have* $Q^*Q = I$ *and so*

$$\|Qx\|_2^2 = x^*Q^*Qx = x^*x = \|x\|_2^2$$

hence

$$\|Qx\|_2 = \|x\|_2$$

we therefore say that the 2-norm is invariant under unitary transformations.

- l_∞, *(infinity norm or the max norm):*

$$\|x\|_\infty = \max_{1 \leqslant i \leqslant n} |x_i|$$

- *Weighted norm* $\|.\|_{W,p}$ *: Given* $W = diag(w_1, w_2, w_3, ..., w_n)$,

$$\|x\|_{W,p} = \|Wx\|_p = (\sum_{i=1}^{n} |w_i x_i|^p)^{\frac{1}{p}}$$

Another common norm is the A-norm, defined in terms of a positive definite matrix A by

$$\|x\|_A = (x^T Ax)^{\frac{1}{2}}$$

We also have a relationship that applies to products of norms, the Hölder inequality.

$$|x^T y| \leqslant \|x\|_p \|y\|_q, \quad \frac{1}{p} + \frac{1}{q} = 1$$

A well-known corollary arises when $p = q = 2$*, the Cauchy-Schwartz inequality.*

$$|x^T y| \leqslant \|x\|_2 \|y\|_2.$$

A very important property of norms is that they are all continuous functions of the entries of their arguments. It follows that a sequence of vectors $x_0, x_1, x_2, ...$ in a $\mathbb{C}^n$ or $\mathbb{R}^n$ converges to a vector x if and only if

$$\lim_{k \to \infty} \|x_k - x\| = 0$$

for any norm on $\mathbb{C}^n$ or $\mathbb{R}^n$. In this case we write $\lim_{i \to \infty} x_i = x$ *or* $x_i \to x$ *as* $i \to \infty$ *where* $i \in \mathbb{P}(a\, positive\, integer)$. For this reason, norms are very useful to measure the error in an approximation.

We now highlight some additional, and useful, relationships involving norms. First of all, the triangle inequality generalizes directly to sums of more than two vectors:

$$\|x + y + z\| \leqslant \|x + y\| + \|z\| \leqslant \|x\| + \|y\| + \|z\|$$

and in general,

$$\|\sum_{i=1}^{m} x_i\| \leqslant \sum_{i=1}^{m} \|x_i\|$$

What can we say about the norm of the difference of two vectors? While we know that $\|x - y\| \leqslant \|x\| + \|y\|$, we can obtain a more useful relationship as follows: From

$$\|x\| = \|(x - y) + y\| \leqslant \|x - y\| + \|y\|$$

we obtain

$$\|x - y\| \geqslant |(\|x\| - \|y\|)| \tag{2.1}$$

There are also interesting relationships among different norms. First and foremost, all norms on $\mathbb{C}^n$ *or* $(\mathbb{R}^n)$, in some sense, are equivalent.

Definition 2. (**Norm equivalence**): If $\|\cdot\|_a$ and $\|\cdot\|_b$ are norms on $\mathbb{C}^n$ *or* $(\mathbb{R}^n)$ then there exist positive constants such that $0 < c_1 \leqslant c_2 < \infty$ with

$$c_1\|x\|_a \leqslant \|x\|_b \leqslant c_2\|x\|_a \qquad \forall x \in \mathbb{C}^n \tag{2.2}$$

for all $x \in \mathbb{R}^n$.

For example, for any $x \in \mathbb{C}^n$ we have

$$\|x\|_2 \leqslant \|x\|_1 \leqslant \sqrt{n}\|x\|_2 \tag{2.3}$$

$$\|x\|_\infty \leqslant \|x\|_2 \leqslant \sqrt{n}\|x\|_\infty \tag{2.4}$$

$$\|x\|_\infty \leqslant \|x\|_1 \leqslant n\|x\|_\infty \tag{2.5}$$

$$\frac{1}{n}\|x\|_1 \leqslant \|x\|_\infty \leqslant \|x\|_1 \tag{2.6}$$

Remarks 2.

- The same result holds for norms on $\mathbb{R}^n$.
- A norm is absolute if

  $$\||x|\| = \|x\|$$

  where $|x|$ is the vector with components given by the absolute values of the components of x
- A norm is called monotone if

  $$|x| \leqslant |y| \Longrightarrow \|x\| \leqslant \|y\|$$

  where $|x| \leqslant |y|$ means a componentwise inequality i.e $|x_i| \leqslant |y_i| \ \forall \ i$.
- A vector norm $\|\cdot\|$ on $\mathbb{F}^n$ $(\mathbb{C}^n$ *or* $\mathbb{R}^n)$ is monotone iff it is absolute.[8, p. 285, Theorem 5.5.10]

It is a fact that for example a norm on $\mathbb{R}^2$ that is not monotone is given by

$$N(x) = |x_1| + |x_2| + |x_1 - x_2|. \tag{2.7}$$

This is a norm that is not monotone and not absolute either.
Let us show this is a norm.

- For any $x \in \mathbb{R}^2$

  $$N(x) = |x_1| + |x_2| + |x_1 - x_2| \geqslant 0.$$

  if $N(x) = 0$ then $|x_1| = 0$, $|x_2| = 0$, $|x_1 - x_2| = 0$ thus $x = 0$.
  If $x = 0$ then $N(x) = 0$.

- For any $x \in \mathbb{R}^2$, $\alpha \in \mathbb{R}$

  $$N(\alpha x) = |\alpha x_1| + |\alpha x_2| + |\alpha(x_1 - x_2)|$$

  $$N(\alpha x) = |\alpha|(|x_1| + |x_2| + |x_1 - x_2|)$$
  $$N(\alpha x) = |\alpha||N(x)|$$

- For any $x, y \in \mathbb{R}^2$

  $$N(x + y) = |x_1 + y_1| + |x_2 + y_2| + |(x_1 + y_1) - (x_2 + y_2)|$$

  $$N(x + y) \leqslant |x_1| + |y_1| + |x_2| + |y_2| + |x_1 - x_2| + |y_1 - y_2|$$
  $$N(x + y) \leqslant (|x_1| + |x_2| + |x_1 - x_2|) + (|y_2| + |y_2| + |y_1 - y_2|)$$
  $$N(x + y) \leqslant N(x) + N(y)$$

  But, if

  $$x = \begin{pmatrix} 0.9 \\ -0.9 \end{pmatrix} \quad y = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

  and $N(x) = 3.6$ $N(y) = 2$ then $N(x) \nleqslant N(y)$.

Table 1.

Vector Norm in Matlab	
Quantity	*Matlab syntax*
$\|x\|_1$	*norm(x,1)*
$\|x\|_2$	*norm(x,2)*
$\|x\|_\infty$	*norm(x,inf)*
$\|x\|_p$	*norm(x,p)*

Let us consider the following example, that shows how to compute vector norm.

Example 2. *Let*

$$x = \begin{pmatrix} 1 \\ -2 \\ 3 \\ 4 \end{pmatrix}$$

Compute $\|x\|_1$, $\|x\|_2$, $\|x\|_\infty$.
We know that

$$\|x\|_1 = \sum_{i=1}^{n} |x_i| \quad \forall\, x \in \mathbb{R}^n$$

$$\|x\|_1 = |1| + |-2| + 3| + |-4|$$

$$= 1 + 2 + 3 + 4 = 10$$

$$\|x\|_2 = \sqrt{\sum_{i=1}^{n} |x_i|^2}$$

$$\|x\|_2 = \sqrt{(1)^2 + (-2)^2 + (3)^2 + (-4)^2}$$

$$= \sqrt{1 + 4 + 9 + 16} = \sqrt{30} = 5.4772$$

$$\|x\|_\infty = \max_{1 \leqslant i \leqslant n} \{|x_i|\}$$

$$\|x\|_\infty = max\{|1|, |-2|, |3|, |-4|\}$$

$$= max\{1, 2, 3, 4\} = 4$$

So,

$$\|x\|_1 = 10, \quad \|x\|_2 = 5.4772, \quad \|x\|_\infty = 4$$

Example 3. *Let*

$$x = \begin{pmatrix} i \\ 2 \\ 1 - i \\ 0 \\ 1 + i \end{pmatrix}$$

Compute $\|x\|_2$

$$\|x\| = \sqrt{x^* x} = \sqrt{1 + 4 + 2 + 0 + 2} = 3$$

2.2 Rounding Error

When calculations are performed on a computer, each arithmetic operation is generally affected by *roundoff error*. This error arises because the machine hardware can only represent a subset of the real numbers. For example, if we divide 1 by 3 in the decimal system, we obtain the nonterminating fraction 0.33333..... Since we can store only a finite number of these $3's$, we must round or truncate the fraction to some fixed number of digits, say 0.3333. The remaining $3's$ are lost, and forever after we have no way of knowing whether we are working with the fraction 1/3 or some other number like 0.33331415.... We will confine ourselves to sketching how rounding error affects our algorithms. To understand the sketches, we must be familiar with the basic ideas- absolute and relative error, floating-point arithmetic, forward and backward error analysis, and perturbation theory.

2.2.1 Absolute and Relative Error

Definition 3. Let $x \in \mathbb{R}^n$ and $\hat{x} \in \mathbb{R}^n$. Then the absolute error in x as an approximation to $\hat{x}$ is the number

$$\varepsilon = \|x - \hat{x}\|. \tag{2.8}$$

Definition 4. Let $x \neq 0$ and $\hat{x}$ be scalars. Then the relative error in $\hat{x}$ as an approximation to x is the number

$$\varepsilon = \frac{\|x - \hat{x}\|}{\|x\|}. \tag{2.9}$$

As above, if the relative error of $\hat{x}$ is, say 10^{-5}, then we say that $\hat{x}$ is accurate to 5 decimal digits. The following example illustrates these ideas:

Example 4. *Let*

$$x = \begin{pmatrix} 1 \\ 100 \\ 9 \end{pmatrix}$$

$$\hat{x} = \begin{pmatrix} 1.1 \\ 99 \\ 11 \end{pmatrix}$$

$$\|x\|_1 = 110.0000, \quad \|\hat{x} - x\|_1 = 3.1000$$

$$\frac{\|\hat{x}-x\|_1}{\|x\|_1} = 0.0282, \quad \frac{\|\hat{x}-x\|_1}{\|\hat{x}\|_1} = 0.0279$$

$$\|x\|_2 = 100.4092, \quad \|\hat{x}-x\|_2 = 2.2383$$

$$\frac{\|\hat{x}-x\|_2}{\|x\|_2} = 0.0223, \quad \frac{\|\hat{x}-x\|_2}{\|\hat{x}\|_2} = 0.0225$$

$$\|x\|_\infty = 100, \quad \|\hat{x}-x\|_\infty = 2$$

$$\frac{\|\hat{x}-x\|_\infty}{\|x\|_\infty} = 0.0200, \quad \frac{\|\hat{x}-x\|_\infty}{\|\hat{x}\|_\infty} = 0.0202$$

Thus, we would say that $\hat{x}$ approximates x to 2 decimal digits.

Definition 5. The component-wise relative error.

$$\varepsilon_{elem} = \|y\|$$
$$y_i = \frac{\hat{x}_i - x_i}{x_i}$$

From the above example (4)

$$y = \begin{pmatrix} 0.1 \\ \frac{1}{100} \\ \frac{2}{9} \end{pmatrix}$$

$$\|y\|_\infty = \frac{2}{9} \approx 0.2 \approx 2 \times 10^{-1}$$

2.3 Floating Point Arithmetic

The number -3.1416 may be expressed in *scientific notation* as follows:

$$-.31416 \times 10^1.$$

- sign$'-'$
- mantissa=fraction= .31416
- base= 10

- exponent= 1

Computers use a similar representation called *floating point*, but generally the base is 2 (with exceptions, such as 16 for IBM 370 and 10 for some spreadsheets and most calculators). For example, $.10101_2 \times 2^3 = 5.25_{10}$.
A floating point number is called *normalized* if the leading digit of the fraction is nonzero. For example, $.10101_2 \times 2^3$ is normalized, but $.010101_2 \times 2^4$ is not. Floating point numbers are usually normalized. The advantage of the floating-point representation is that it allows very large and very small numbers to be represented accurately. For example the floating point numbers are $.6542 \times 10^{36}$, a large four-digit decimal number, and $-.71236 \times 10^{-42}$, a small five-digit decimal number.

Definition 6. A nonzero real number x can be represented using *floating point* notation. Floating point is based on exponential or scientific notation. In exponential notation, a nonzero real number x is expressed in decimal as

$$x = \pm\alpha \times \beta^e \tag{2.10}$$

where

$$\alpha = .d_1 d_2 ... d_n,$$

and

$$e = is\ an\ integer$$

The number α, n and e are called *the nonnegative real number the mantissa*, *the mantissa length* and the *exponent* respectively. Most computers are designed with binary arithmetic $\beta = 2$ or hexadecimal arithmetic $\beta = 16$, while humans use decimal arithmetic $\beta = 10$.

- **Floating point numbers with base 10**

$$x = \pm(.d_1 d_2 ... d_n)_{10} \times 10^e \tag{2.11}$$

$$x = \pm(d_1 10^{-1} + d_2 10^{-2} + ... + d_n 10^{-n}) \times 10^e$$

For example, with $n = 7$

$$12.3456 = +(.1234560)_{10} \times 10^2$$

$$= +(1\cdot 10^{-1} + 2\cdot 10^{-2} + 3\cdot 10^{-3} + 4\cdot 10^{-4} + 5\cdot 10^{-5} + 6\cdot 10^{-6} + 0\cdot 10^{-7}) \times 10^2$$

- **Floating point numbers with base 2**

$$x = \pm(.d_1 d_2 ... d_n)_2 \times 2^e \tag{2.12}$$

$$x = \pm(d_1 2^{-1} + d_2 2^{-2} + ... + d_n 2^{-n}) \times 2^e$$

For example, with $n = 7$

$$-3.5 = -(0.111)_2 \times 2^2$$

$$= (1 \cdot 2^{-1} + 1 \cdot 2^{-2} + 1 \cdot 2^{-3}) \times 2^2$$

2.4 Matrix Norm

Just like the determinant, the norm of a matrix is a simple unique scalar. It is a measure of the size or magnitude of the matrix. However, a norm is always non-negative and is defined for all matrices-square or rectangular, invertbile or non-invertbile square matrices.

Definition 7. Matrix norm on $\mathbb{M}_n$ is a *non-negative* real-valued function $\|\cdot\| : \mathbb{C}^{m\times n} \longrightarrow \mathbb{R}$, if for all $A, B \in \mathbb{M}_n$ it satisfies the following

- Nonnegativity:
  $\|A\| \geqslant 0$ for all $A \in \mathbb{M}_n$
- Positive:
  $\|A\| = 0$ if and only if $A = 0$
- Homogeneous:
  $\|\alpha A\| =\mid \alpha \mid \|A\| \ \forall \ \alpha \in \mathbb{C}^n, \ A \in \mathbb{M}_n$
- Triangule inequality:
  $\|A + B\| \leqslant \|A\| + \|B\| \qquad \forall \ A, B \in \mathbb{M}^{m\times n}$
- Submultiplicative property:
  $\|AB\| \leqslant \|A\|\|B\|, \quad \forall \ A, B \in \mathbb{M}^{n\times n}$

We note that the five properties of a matrix norm do not imply that it is invariant under transposition, and in general, $\|A^T\| \neq \|A\|$. Some matrix norms are the same for the transpose of a matrix as for the original matrix.

For a square matrix A, the consistency property for a matrix norm yields

$$\|A^k\| \leqslant \|A\|^k \tag{2.13}$$

for any positive integer k.

A matrix norm $\|\cdot\|$ is *unitarily invariant* if $\|A\| = \|UAV^*\|$ for all matrix A and all unitary matrices U and V of appropriate dimension.

Definition 8. A *matrix norm* $\|\cdot\|$ on $\mathbb{C}^{m\times n}$ is said to be consistent with respect to the *vector norms* $\|\cdot\|_a$ on $\mathbb{C}^n$ and $\|\cdot\|_b$ on $\mathbb{C}^m$ if

$$\|Ax\|_b \leqslant \|A\|\|x\|_a,\ \forall x \in \mathbb{C}^n,\ A \in \mathbb{C}^{m\times n}. \tag{2.14}$$

Usually $\|\cdot\|_\alpha = \|\cdot\|_\beta$

Example 5. *Suppose $A \in \mathbb{C}^{n\times n}$ is a diagonal matrix with entries $d_1, d_2, ..., d_n$, i.e.,*

$$A = \begin{pmatrix} d_1 & \cdots & \cdots & \cdots \\ \vdots & d_2 & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \vdots & d_n \end{pmatrix}$$

and define

$$\|A\|_2 = \max_{\|x\|_2=1} \|Ax\|_2 = \max_{\|x\|_2=1} \sqrt{\sum_{i=1}^{m} |d_i x_i|^2}$$

Now

$$\sum_{i=1}^{n} |d_i x_i|^2 \leqslant \max_{1\leqslant i\leqslant n} |d_i|^2 \sum_{i=1}^{n} |x_i|^2 = \max_{1\leqslant i\leqslant n} |d_i|^2$$

when $\|x\|_2 = 1$. Therefore,

$$\|A\|_2 = \max_{1\leqslant i\leqslant n} |d_i| \tag{2.15}$$

but same is true for $\|A\|_1$, $\|A\|_\infty$ or any induced $\|.\|_p$

2.4.1 Induced Matrix Norm

Definition 9. Given a vector norm $\|.\|$ on $\mathbb{C}^n$, we define the induced matrix norm $\|.\|$ on $\mathbb{C}^{n\times n}$ by

$$\|A\| = \max_{x\neq 0} \frac{\|Ax\|}{\|x\|} = \max_{\|x\|=1} \|Ax\|,\ \forall x \in \mathbb{C}^n,\ A \in \mathbb{C}^{n\times n} \tag{2.16}$$

Where $\|x\|$ is any given vector norm. We say that the vector norm $\|x\|$ induces the matrix norm $\|A\|$.

Lemma 1. *[21, p. 91, Theorem 2.2.16] For any $x \neq 0$, matrix A, and any natural norm $\|\cdot\|$, we have*

$$\|Ax\| \leqslant \|A\|\|x\| \tag{2.17}$$

Proof. Consider $x \in \mathbb{C}^n$ and $x = 0$. Clearly the result is true trivially. Otherwise, if $x \neq 0$. Then we have

$$\|A\| = \max_{z \neq 0} \frac{\|Az\|}{\|z\|}$$

Thus for the given x

$$\|A\| \geqslant \frac{\|Ax\|}{\|x\|}$$
$$\|A\|\|x\| \geqslant \|Ax\|$$

or

$$\|Ax\| \leqslant \|A\|\|x\|.$$

□

2.4.2 Matrix p-Norm

The matrix p-norm is the norm subordinate to the Hölder p-norm. Hence, for all $A \in \mathbb{C}^{m \times n}$ define

$$\|A\|_p = \begin{cases} \sum_{i=1}^{n} \sum_{j=1}^{m} |A_(i,j)|^p)^{1/p} & 1 \leqslant p < \infty \\ \max_{\substack{i \in \{1,\ldots,n\} \\ j \in 1,\ldots,m}} |A_(i,j)| & p = \infty \end{cases}$$

2.4.3 Frobenius Matrix Norm

Definition 10. The Frobenius norm of $A \in \mathbb{C}^{m \times n}$ is defined by the equation

$$\|A\|_F = (\sum_{i,j} |a_{ij}|^2)^{\frac{1}{2}}. \tag{2.18}$$

It has the equivalent definition

$$\|A\|_F = (trace(A^*A))^{\frac{1}{2}} = (trace(AA^*))^{\frac{1}{2}} \tag{2.19}$$

- It is not a $p - norm$ since $\|I_n\|_F = \sqrt{n}$.
- It can be viewed as the $2-norm$ (*Euclidean norm*) of a vector in $\mathbb{R}^{m \times n}$.
- Some useful properties:

$$\|AB\|_F \leqslant \|A\|_2 \|B\|_F$$
$$\|AB\|_F \leqslant \|A\|_F \|B\|_2$$
$$\|A^n\|_F \leqslant \|A\|_F^n.$$

Since the norm $\|\cdot\|_F$ is unitarily invariant, we have the important fact that

$$\begin{aligned}\|UAV^*\|_F^2 &= tr(VA^*U^*UAV^*)\\ &= tr(VA^*AV^*)\\ &= tr(V^*(VA^*AV^*)V)\\ &= tr(A^*AV^*V)\\ &= tr(A^*A)\\ &= \|A\|_F^2\end{aligned}$$

Matrix norms examples.

In this section Matrix norms are commonly used in scientific computing. The most familiar examples are the l_p norms for *p=1,2,*∞.

- The *p-norm* defined for $A \in \mathbb{C}^{m\times n}$

$$\|A\|_p = \max_{x\neq 0} \frac{\|Ax\|_p}{\|x\|_p},\ \ A \in \mathbb{C}^{m\times n},\ \ 1 \leqslant p < \infty \tag{2.20}$$

- The l_1 *norm*, defined for $A \in \mathbb{M}_n$ by

$$\|A\|_1 = \max_{1\leqslant j\leqslant n} \sum_{i=1}^{m} |a_{ij}| \tag{2.21}$$

  is a matrix norm because

$$\|AB\|_1 = \sum_{i,j=1}^{n} |\sum_{k=1}^{n} a_{ik}b_{kj}| \leqslant \sum_{i,j,k=1}^{n} |a_{ik}b_{kj}|$$

$$\leqslant \sum_{i,j,k,m=1}^{n} |a_{ik}b_{mj}| = (\sum_{i,k=1}^{n} |a_{ik}|)(\sum_{j,m=1}^{n} |b_{mj}|) = \|A\|_1\|B\|_1$$

- The l_∞ *norm*, defined for $A \in \mathbb{M}_n$ by

$$\|A\|_\infty = \max_{1\leqslant i\leqslant m} \sum_{j=1}^{n} |a_{ij}|. \tag{2.22}$$

2.4.4 The Euclidean norm or 2-Norm

Definition 11. The l_2 *norm* defined for $A \in \mathbb{M}_n$ by

$$\|A\|_2 = \max_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2} = \max_{\|x\|_2 = 1} \|Ax\|_2 \tag{2.23}$$

Example 6. *Let I_n denote the $n \times n$ identity matrix then,*

$$\|I_n\|_2 = \sqrt{n}$$

$$\|I\| = 1$$

Since the l_2 *norm* i.e., $\|\cdot\|_2$, is unitarily invariant, we have the important fact that

$$\begin{aligned} \|UAV^*\|_2^2 &= \max_{\|x\|_2 \neq 0} \frac{x^*VA^*U^*UAV^*x}{x^*x} \\ &= \max_{\|x\|_2 \neq 0} \frac{x^*VA^*AV^*x}{x^*V^*Vx} \\ &= \max_{\|y\|_2 \neq 0} \frac{y^*A^*Ay}{y^*y} \quad where \quad y = V^*x \\ &= \|A\|_2^2 \end{aligned}$$

2.4.5 Spectral Norm

The matrix 2-norm is also known as the *spectral norm.* This name is connected to the fact that the norm is given by the square root of the largest eigenvalue of A^*A.

Definition 12. The *spectral norm* $\|\cdot\|_2$ is defined on $A \in \mathbb{M}_n$ by

$$\|A\|_2 = max\{\sqrt{\lambda} : \lambda \, is \, an \;\; eigenvalue \, of \; A^*A\}$$

Note that if $A^*Ax = \lambda x$ and $x \neq 0$, then $x^*A^*Ax = \|Ax\|_2^2 = \lambda\|x\|_2^2$, so $\lambda \geqslant 0$ and $\sqrt{\lambda}$ is real and nonnegative.

Definition 13. The *spectral radius* $\rho(A)$ of a matrix $A \in \mathbb{M}_n$ is

$$\rho(A) = max\{|\lambda| : \lambda \, is \, an \, eigenvalue \, of \; A\}$$

Theorem 1. [2, p. 22, Lemma 1.7.7] Let $A \in \mathbb{M}_n$

$$\|A\|_2 = \sqrt{\lambda_{max}(A^*A)}$$

where $\lambda_{max}(A^*A)$ is the maximum eigenvalue of A^*A.

Proof.

$$\|A\|_2 = \max_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2}$$

$$= \max_{x \neq 0} \frac{(x^*A^*Ax)^{\frac{1}{2}}}{\|x\|_2}$$

Since A^*A is Hermitian, there exists an eigendecomposition $A^*A = U\Lambda U^*$ with U a unitary matrix, and Λ is a diagonal matrix containing the eigenvalues of A^*A, which must all be non-negative. Let $y = Q^*x$ Then

$$\|A\|_2 = \max_{x \neq 0} \frac{(x^*(Q\Lambda Q^*)x)^{\frac{1}{2}}}{\|x\|_2}$$

$$= \max_{x \neq 0} \frac{((Q^*x)^*\Lambda(Q^*x))^{\frac{1}{2}}}{\|Q^*x\|_2}$$

Since A^*A is positive definite, all the eigenvalues are positive.

$$\|A\|_2 = \max_{y \neq 0} \frac{(y^*\Lambda y)^{\frac{1}{2}}}{\|y\|_2}$$

$$= \max_{y \neq 0} \sqrt{\frac{\Sigma \lambda_i y_i^2}{\Sigma y_i^2}} \leqslant \sqrt{\lambda_{max}(A^*A)\frac{\Sigma y_i^2}{\Sigma y_i^2}} = \sqrt{\lambda_{max}(A^*A)}$$

which can be attained by choosing y to be the jth column vector of the identity matrix if, say, λ_j is the largest eigenvalue. □

When A is non-singular, then

$$\|A^{-1}\|_2 = \frac{1}{\min_{\|x\|_2=1} \|Ax\|_2} = \frac{1}{\sqrt{\lambda_{min}}} \tag{2.24}$$

where λ_{min} is the smallest eigenvalue of (A^*A).

Example 7. *Determine the induced norm* $\|A\|_2$ *and* $\|A^{-1}\|_2$ *for the nonsigular matrix*

$$A = \frac{1}{\sqrt{3}} \begin{pmatrix} 3 & -1 \\ 0 & \sqrt{8} \end{pmatrix}$$

The eigenvalues of (A^*A) *are* $\lambda = 2$ *and* $\lambda = 4$, *so* $\lambda_{min} = 2$ *and* $\lambda_{max} = 4$. *Consequently,*

$$\|A\|_2 = \sqrt{\lambda_{max}} = 2 \quad and \quad \|A^{-1}\|_2 = \sqrt{\lambda_{min}} = \frac{1}{\sqrt{2}}$$

Earlier, we had shown that

$$\|A\|_2 = (\rho(A^*A))^{\frac{1}{2}}$$

if A is symmetric, then

$$\|A\|_2 = \rho(A). \tag{2.25}$$

Thus $\rho(A)$ is a norm on the space of real symmetric matrices, but we will see that it is not a norm on the set of general matrices $\mathbb{M}_n$:

Example 8. *Let*

$$A = \begin{pmatrix} 1 & 0 \\ 2 & 0 \end{pmatrix} \ and \ B = \begin{pmatrix} 0 & 2 \\ 0 & 1 \end{pmatrix}$$

which gives

$$\rho(A) = max\{|1|, |0|\} = 1 \ and \ \rho(B) = max\{|0|, |1|\} = 1$$

However,

$$A + B = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}$$

and

$$\rho(A + B) = max\{|-1|, |3|\} = 3$$

hence

$$\rho(A + B) \nleq \rho(A) + \rho(B).$$

So the spectral radius does not satisfy the properties of a norm i.e triangular inequality.

Properties of 2-Norm

1. $\|A\|_2 = \|A^*\|_2$
2. $\|A^*A\|_2 = \|A\|_2^2$
3. $\|U^*AV\|_2 = \|A\|_2$, where $U^*U = I$ and $V^*V = I$
4. $\|T\| = max\{\|A\|_2, \|B\|_2\}$ where T is,

$$T = \begin{pmatrix} A & 0 \\ 0 & B \end{pmatrix}$$

5. $$\|A\|_2 = \max_{\|x\|_2=1} \max_{\|y\|_2=1} \|y^*Ax\|_2$$
6. $\|AU\|_2 = \|A\|_2$ $\|UA\|_2 = \|A\|_2$ where U is a unitary matrix,
7. $\|AB\|_2 \leqslant \|A\|_2\|B\|_2$
8. $\|A\|_2 \leqslant \|A\|_F$

Note that the equivalences (2.3), (2.4), (2.5), (2.6) between vector norms on $\mathbb{C}^n$ imply the corresponding equivalences between the subordinate matrix norms. Namely, for any matrix $A \in \mathbb{C}^{m\times n}$. Then the following inequalities holds for A,

$$\frac{1}{\sqrt{n}}\|A\|_2 \leqslant \|A\|_1 \leqslant \sqrt{n}\|A\|_2 \tag{2.26}$$

$$\frac{1}{\sqrt{n}}\|A\|_\infty \leqslant \|A\|_2 \leqslant \sqrt{n}\|A\|_\infty \tag{2.27}$$

$$\frac{1}{n}\|A\|_1 \leqslant \|A\|_\infty \leqslant n\|A\|_1 \tag{2.28}$$

$$\|A\|_2^2 \leqslant \|A\|_1\|A\|_\infty \tag{2.29}$$

We will prove these results.

1. $$\frac{1}{\sqrt{n}}\|A\|_2 \leqslant \|A\|_1 \leqslant \sqrt{n}\|A\|_2$$

   Let x be such that $\|x\|_2 = 1$ and $\|Ax\|_2 = \|A\|_2$. Then

$$\|A\|_2 = \|Ax\|_2 \leqslant \|Ax\|_1 \leqslant \|A\|_1\|x\|_1 \leqslant \|A\|_1\sqrt{n}\|x\|_2 = \sqrt{n}\|A\|_1$$

proving the left inequality:

$$\|A\|_2 \leqslant \sqrt{n}\|A\|_1\|$$

$$\frac{1}{\sqrt{n}}\|A\|_2 \leqslant \|A_1$$

For the right inequality, pick y such that $\|y\|_1 = 1$ and $\|Ay\|_1 = \|A\|_1$. Then

$$\|A\|_1 = \|Ay\|_1 \leqslant \sqrt{n}\|Ay\|_2 \leqslant \sqrt{n}\|A\|_2\|y\|_2 \leqslant \sqrt{n}\|A\|_2\|y\|_1 = \sqrt{n}\|A\|_2$$

$$\|A\|_1 \leqslant \sqrt{n}\|A\|_2$$

Hence

$$\frac{1}{\sqrt{n}}\|A\|_2 \leqslant \|A\|_1 \leqslant \sqrt{n}\|A\|_2$$

2.

$$\frac{1}{\sqrt{n}}\|A\|_\infty \leqslant \|A\|_2 \leqslant \sqrt{n}\|A\|_\infty$$

Applying the result in (1) to A^T, we get

$$\|A^T\|_2 \leqslant \sqrt{n}\|A^T\|_1 \leqslant \sqrt{n}\|A\|_\infty$$

and

$$\|A\|_\infty = \|A^T\|_1 \leqslant \sqrt{n}\|A^T\|_2$$

But

$$\|A\|_2 = \|A^T\|_2$$

(if $A = U\Sigma V^T$, then $A^T = V\Sigma U^T$, so A and A^T have the same singular values). So

$$\|A\|_2 \leqslant \sqrt{n}\|A\|_\infty$$

and

$$\|A\|_\infty \leqslant \sqrt{n}\|A\|_2 \Rightarrow \|A\|_2 \geqslant \frac{1}{\sqrt{n}}\|A\|_\infty$$

Hence

$$\frac{1}{\sqrt{n}}\|A\|_\infty \leqslant \|A\|_2 \leqslant \sqrt{n}\|A\|_\infty$$

3. From result (1) and (2), we get

$$\|A\|_1 \leqslant \sqrt{n}\|A\|_2 \leqslant n\|A\|_\infty$$
$$\|A\|_\infty \leqslant \sqrt{n}\|A\|_2 \leqslant n\|A\|_1$$

combining them gives

$$\frac{1}{n}\|A\|_1 \leqslant \|A\|_\infty \leqslant n\|A\|_1$$

as required.

4.

$$\|A\|_2^2 \leqslant \|A\|_1\|A\|_\infty$$

If $z \neq 0$ is such that with $\mu = \|A\|_2$ satisfies

$$A^TAz = \mu^2 z$$

μ^2 is the largest eigenvalue of A^TA and z is the corresponding eigenvector. Take the 1-norm, we have

$$\mu^2\|z\|_1 = \|A^TAz\|_1 \leqslant \|A^T\|_1\|A\|_1\|z\|_1 = \|A\|_\infty\|A\|_1\|z\|_1$$

divide by $\|z\|_1$

$$\mu^2 \leqslant \|A\|_\infty\|A\|_1$$

hence

$$\|A\|_2^2 \leqslant \|A\|_\infty\|A\|_1$$

We can also follow like Since $\|A\| \geqslant \rho(A)$ for any natural matrix norm, we have

$$\begin{aligned}\|A\|_2^2 &= \rho(A^TA)\\ &\leqslant \|A^TA\|_\infty\\ &\leqslant \|A^T\|_\infty\|A\|_\infty\\ &\|A\|_1\|A\|_\infty\end{aligned}$$

hence

$$\|A\|_2 \leqslant \sqrt{\|}A\|_1\|A\|_\infty$$

Table 2.

Matrix Norm in Matlab	
Quantity	*Matlab syntax*
$\|A\|_1$	*norm(A,1)*
$\|A\|_2$	*norm(A,2)*
$\|A\|_\infty$	*norm(A,inf)*
$\|A\|_F$	*norm(A,fro)*

2.5 Convergence of a Matrix Power Series

We say that a sequence of a matrices $A_1, A_2, ...$(of same order) *converges* to the matrix A with respect to the norm $\|\cdot\|$ if the sequence of real numbers $\|A_1 - A\|$, $\|A_2 - A\|, ...$ *converges* to 0. So, $\{A_i\}_{i=1}^{\infty} \in \mathbb{F}^{m\times n}$ *converges* to $A \in \mathbb{F}^{m\times n}$ if

$$\lim_{i\to\infty} \|A_i - A\| = 0 \tag{2.30}$$

where $\|\cdot\|$ is a norm on $A \in \mathbb{F}^{m\times n}$.
In this case, we write

$$\lim_{i\to\infty} A_i = A \, or \, A_i \to A \, as \, i \to \infty, \, where \, i \in \mathbb{P}.$$

Because of the equivalence property of norms, the choice of the norm is irrelevant. For a square matrix A, we have the important fact that

$$A^k \to 0, \Leftrightarrow \|A\| < 1, \tag{2.31}$$

where 0 is the square zero matrix of the same order as A and $\|\cdot\|$ is any matrix norm. This convergence follows from inequality (2.13) because that yields

$$\lim_{k\to\infty} \|A^k\| \leqslant \lim_{k\to\infty} \|A\|^k,$$

and so if $\|A\| < 1$, then

$$\lim_{k\to\infty} \|A^k\| = 0.$$

2.6 Relation between Norms and the Spectral Radius of a Matrix.

We next recall some results that relate the spectral radius of a matrix to matrix norms.

Theorem 2. [8, p. 297, Theorem 5.6.9] For any induced matrix norm $\|.\|$ and for $A \in \mathbb{C}^{n\times n}$, we have

$$\rho(A) \leqslant \|A\| \tag{2.32}$$

Proof. Let λ be the eigenvalue of A corresponding to the spectral radius i.e $\rho(A) = |\lambda|$ and z be the corresponding eigenvector.
consider

$$\begin{aligned} Az &= \lambda z \\ \|Az\| &= \|\lambda z\| \\ &= |\lambda|\|z\| \\ &= \rho(A)\|z\| \end{aligned}$$

$$\rho(A) = \frac{\|Az\|}{\|z\|} \leqslant \max_{x \neq 0} \frac{\|Ax\|}{\|x\|} = \|A\|$$

hence

$$\rho(A) \leqslant \|A\|$$

□

The inequality (2.32) and the l_1 and l_∞ norms yield useful bounds on the eigenvalues and the maximum absolute row and column sums of matrices. The modulus of any eigenvalue is no greater than the largest sum of absolute values of the elements in any row or column.

The inequality (2.32) and equation (2.25) also yield a minimum property of l_2 norm of a symmetric matrix A:

$$\|A\|_2 = \rho(A) \leqslant \|A\|. \tag{2.33}$$

for any matrix norm $\|\cdot\|$.

Theorem 3. [8, p. 299] Let $A \in \mathbb{C}^{n\times n}$ be a complex-valued matrix, $\rho(A)$ its spectral radius and $\|\cdot\|$ a consistent matrix norm; for each $k \in \mathbb{N}$: then

$$\rho(A) \leqslant \|A^k\|^{1/k} \quad \forall k \in \mathbb{N} \tag{2.34}$$

Proof. Let (x, λ) be an eigenvector-eigenvalue pair for a matrix A. As a consequence, since $\|\cdot\|$ is consistent, we have

$$|\lambda^k|\|x\| = \|\lambda^k x\| = \|A^k x\| \leqslant \|A^k\|\|x\|$$

and since $x \neq 0$ for each λ we have

$$|\lambda^k| \leqslant \|A^k\|$$

and therefore

$$\rho(A^k) \leqslant \|A^k\|^{1/k}$$

$$(\rho(A))^k \leqslant \|A^k\|^{1/k}$$
$$\rho(A) \leqslant \|A^k\|^{1/k}$$

□

Theorem 4. [18, p. 31, Theorem 2.8] Let $A \in \mathbb{C}^{n\times n}$ and $\varepsilon > 0$. Then, there exist a consistent matrix norm $\|.\|$ (depending on ε) such that

$$\|A\| \leqslant \rho(A) + \varepsilon \tag{2.35}$$

Proof. By Schur triangularization theorem, $\exists$ a unitary matrix U and an upper triangular matrix T such that $U^*AU = T = \Lambda + N$ in which Λ is diagonal and N is strictly upper triangular.
For $\alpha > 0$, let

$$G_\alpha = diag(1, \alpha, ..., \alpha^{n-1})$$

Then for $i < j$ the (i, j)-element of $G^{-1}NG$ is $\nu_{ij}\alpha^{j-i}$. Since for $i < j$ we have $\alpha^{j-i} \to 0$ as $\alpha \to 0$, there is an α such that $\|G^{-1}NG\|_1 < \epsilon$. It follows that

$$\|G^{-1}U^*AUG\|_1 = \|\Lambda + G^{-1}NG\|_1$$
$$\leqslant \|\Lambda\|_1 + \|G^{-1}NG\|_1 = \rho(A) + \epsilon$$

Define the matrix norm $\|\cdot\|$ on $\mathbb{M}_n$ by

$$\|B\| = \|(UG_\alpha^{-1})^{-1}B(UG_\alpha^{-1})\|_1, \;\; B \in \mathbb{M}_n$$

Let us compute

$$\|A\| = \|(UG_\alpha^{-1})^{-1}A(UG_\alpha^{-1})\|_1$$
$$= \|G_\alpha U^*AUG_\alpha^{-1}\|_1 = \|\Lambda + G_\alpha NG_\alpha^{-1}\|_1$$
$$\leqslant \max_{1\leqslant j\leqslant n} |\lambda_j| + \epsilon = \rho(A) + \epsilon$$

This completes the proof. □

Theorem 5. [8, p. 298, Theorem 5.6.12] Let $A \in \mathbb{C}^{n\times n}$ then

$$\lim_{k\to\infty} A^k = 0 \quad \Leftrightarrow \quad \rho(A) < 1 \tag{2.36}$$

Proof. Assume $\rho(A) < 1$. By theorem (4), there exists $\epsilon > 0$ and a consistent matrix norm $\|.\|$ such that $\|A\| \leqslant \rho(A) + \varepsilon < 1$.
Now

$$\|A^k\| \leqslant \|A\|^k \leqslant (\rho(A) + \varepsilon)^k < 1.$$

Thus $\lim_{k\to\infty} \|A^k\| = 0$. and then $\| \lim_{k\to\infty} A^k\| = 0$,

therefore

$$\lim_{k\to\infty} A^k = 0.$$

Conversely, assume that

$$\lim_{k\to\infty} A^k = 0.$$

Let $Ax = \lambda x, \quad x \neq 0$, then $A^k x = \lambda^k x$.

$$(\lim_{k\to\infty} A^k)x = \lim_{k\to\infty} \lambda^k A \quad \Longrightarrow |\lambda| < 1$$

therefore

$$\rho(A) < 1.$$

□

Theorem 6. [8, p. 299, Corollary 2.6.14] Let $\| \cdot \|$ be a matrix norm on $\mathbb{M}_n$. Then

$$\lim_{k\to\infty} \|A^k\|^{1/k} = \rho(A) \tag{2.37}$$

for all $A \in \mathbb{M}_n$.

Proof. Since, from inequality (2.33) and the fact that $\rho(A)^k = \rho(A^k)$, we have $\rho(A) \leqslant \|A^k\|^{1/k}$ for all $k = 1, 2....$ Now for any $\epsilon > 0$, the matrix $\rho(A/(\rho(A) + \epsilon)) < 1$ and so

$$\lim_{k\to\infty} (A/(\rho(A) + \epsilon)^k = 0$$

from equation (2.18); hence,

$$\lim_{k\to\infty} \frac{\|A^k\|}{(\rho(A) + \epsilon)^k} = 0$$

There is therefore a positive integer M_ϵ such that

$$\frac{\|A^k\|}{(\rho(A) + \epsilon)^k} < 1 \quad for\ all\ k > M_\epsilon,$$

and hence

$$\|A^k\|^{1/k} < (\rho(A) + \epsilon)\ \ for\ k > M_\epsilon.$$

We have therefore, for any $\epsilon > 0$,

$$\rho(A) \leqslant \|A^k\|^{1/k} < \rho(A) + \epsilon\ \ for\ k > M_\epsilon$$

and thus

$$\lim_{k\to\infty} \|A^k\|^{1/k} = \rho(A)$$

□

2.7 Sensitivity of Linear Systems

2.7.1 Condition Numbers

In this section we introduce and discuss the condition number of a matrix. The condition number of $A \in \mathbb{C}^{n\times n}$ is a simple but important in the numerical analysis of a linear system $Ax = b\,(x, b \in \mathbb{C}^n)$. Assume that $detA \neq 0$ so that x is the unique solution of the equation. The condition number provides a measure of ill-posedness of the system, which means how large the change in the solution can be relative to the change in the determinant.

In general terms, a problem is said to be stable, or well conditioned if "small" causes (perturbations in A or b) give rise to "small" effects (perturbation in x).

Perturbation in b

Consider the linear system

$$Ax = b \tag{2.38}$$

where $A \in \mathbb{C}^{n\times n}$ is nonsingular and $b \in \mathbb{C}^n$ is nonzero. The system has a unique solution x. Now suppose that A (non-singular) is fixed and b is perturbed to a new vector $b + \delta b$ and consider the perturbed linear system is

$$A\hat{x} = b + \delta b$$

$$A(x + \delta x) = b + \delta b. \tag{2.39}$$

This system also has a unique solution $\hat{x}$, which is hoped to be not too far from x, and $\delta b \in \mathbb{C}^n$ represents the perturbations in b. Subtract both the system to get,

$$A\delta x = \delta b$$

$$\delta x = A^{-1}\delta b$$

take the norms to get

$$\|\delta x\| = \|A^{-1}\delta b\| \leqslant \|A^{-1}\|\|\delta b\|$$

$$\|\hat{x} - x\| \leqslant \|A^{-1}\|\|\delta b\| \tag{2.40}$$

this gives an upper bound on the absolute error in x.

1. small $\|A^{-1}\|$ means that $\|\delta x\|$ is small when $\|\delta b\|$ is small.
2. large $\|A^{-1}\|$ means that $\|\delta x\|$ can be large, even when $\|\delta b\|$ is small.

To estimate the relative error, note that

$$Ax = b$$

gives

$$\begin{aligned} \|b\| &= \|Ax\| \\ \|b\| &\leqslant \|A\|\|x\| \\ \frac{1}{\|x\|} &\leqslant \|A\|\frac{1}{\|b\|} \end{aligned} \tag{2.41}$$

from equation (2.40) and equation (2.41) we get,

$$\begin{aligned} \frac{\|\hat{x} - x\|}{\|x\|} &\leqslant \|A\|\|A^{-1}\|\frac{\|\delta b\|}{\|b\|} \\ \frac{\|\delta x\|}{\|x\|} &\leqslant \|A\|\|A^{-1}\|\frac{\|\delta b\|}{\|b\|} \end{aligned} \tag{2.42}$$

The quantity $\kappa(A) = \|A\|\|A^{-}1\|$ is called the condition number of A.
This equation says
"relative change in x $\leqslant$Condition number times relative change in b".

1. small $\kappa(A)$ means $\frac{\|\delta x\|}{\|x\|}$ is small when $\frac{\|\delta b\|}{\|b\|}$ is small
2. large $\kappa(A)$ means $\frac{\|\delta x\|}{\|x\|}$ can be large when $\frac{\|\delta b\|}{\|b\|}$ is small

Definition 14. The *p-condition number* $\kappa_p(A)$ of a matrix (with respect to inversion) is defined by

$$\kappa_p(A) = \|A\|_p\|A^{-1}\|_p$$

We set $\kappa_p = \infty$ if A is not invertible.
If

$$Ax = b$$

and

$$A(x + \delta x) = b + \delta b$$

then the relative error satisfies

$$\frac{\|\delta x\|}{\|x\|} \leqslant \kappa_p(A)\frac{\|\delta b\|}{\|b\|}$$

Perturbation in A

So far we have considered only the effect of perturbing b. We must also consider perturbations of A while keeping b fixed, so that we are interested in the exact solution of $(A + \delta A)(x + \delta x) = b$.
Now let us consider the relationship between the solution of

$$Ax = b \tag{2.43}$$

and the perturbed linear system

$$(A + \delta A)\hat{x} = b.$$

Let $\delta x = \hat{x} - x$, so that $\hat{x} = x + \delta x$ then

$$(A + \delta A)(x + \delta x) = b. \tag{2.44}$$

Subtraction and rearrange to get

$$A\delta x = \delta A(x + \delta x). \tag{2.45}$$

Assume that δA is small enough so that δx is small and that $\|\delta x\| = O(\|\delta A\|)$ then

$$\|(\delta A)(\delta x)\| \leqslant \|\delta A\|\|\delta x\| = O(\|\delta A\|^2)$$

so to first order in δA, we have

$$A(\delta x) + (\delta A)x \approx 0$$

$$\delta x \approx -A^{-1}\delta A x.$$

Take norm we get

$$\frac{\|\delta x\|}{\|x\|} \leqslant \|A^{-1}\|\|\delta A\|,$$

to first order. Multiplying and dividing the right hand side by $\|A\|$ we obtain the bound in the following form

$$\frac{\|\delta x\|}{\|x\|} \leqslant \|A^{-1}\|\|A\|\frac{\|\delta A\|}{\|A\|}, \tag{2.46}$$

to first order in $\|\delta A\|$.

The quantity $\kappa(A) = \|A^{-1}\|\|A\|$ appears in both equation (2.42) and equation (2.46) serves as a measure of how perturbation in the data of the problem $Ax = b$ affects the solution. An example of using Norms to solve a linear system of equations.

Example 9. *We will consider an innocent looking set of equations that results in a large condition number.*

Consider the linear system $Wx = b$*, where* W *is the Wilson Matrix[16].*

$$W = \begin{pmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{pmatrix}$$

and the vector

$$x = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

so that

$$Wx = b = \begin{pmatrix} 32 \\ 23 \\ 33 \\ 31 \end{pmatrix}$$

Now suppose we actually solve

$$W\hat{x} = b + \delta b = \begin{pmatrix} 32.1 \\ 22.9 \\ 33.1 \\ 30.9 \end{pmatrix},$$

i.e.,

$$\delta b = \begin{pmatrix} +0.1 \\ -0.1 \\ +0.1 \\ -0.1 \end{pmatrix}.$$

First, we must find the inverse of W*:*

$$W^{-1} = \begin{pmatrix} 25 & -41 & 10 & -6 \\ -41 & 68 & -17 & 10 \\ 10 & -17 & 5 & -3 \\ -6 & 10 & -3 & 2 \end{pmatrix}$$

Then from $W\hat{x} = b + \delta b$, we find

$$\hat{x} = W^{-1}b + W^{-1}\delta b = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 8.2 \\ -13.6 \\ 3.5 \\ -2.1 \end{pmatrix}$$

It is clear that the system is sensitive to changes, a small change to b has had a very large effect on the solution. So the Wilson Matrix is an example of an ill-conditioned matrix and we would expect the condition number to be very large.

To evaluate the condition number, $K(W)$ for the Wilson Matrix we need to select a particular norm. First, we select the 1-norm(maximum column sum of absolute values) and estimate the error

$$\|W\|_1 = 33 \qquad and \qquad \|W^{-1}\|_1 = 136$$

and hence,

$$K_1(W) = \|W\|_1 \times \|W^{-1}\|_1 = 33 \times 136 = 4488$$

which of course is considerably bigger than 1.
Remember that

$$\frac{\|\hat{x} - x\|_1}{\|x\|_1} \leqslant K_1(W)\frac{\|\delta b\|_1}{\|b\|_1}$$

such that the

$$error\ in\ x \leqslant 4488 \times \frac{0.4}{119} \approx 15$$

So far, we have used the 1-norm but If one is interested in the error in individual components of the solution then it is more natural to look at the $\infty - norm$.

Since W is symmetric, $\|W\|_1 = \|W\|_\infty$ and $\|W^{-1}\|_1 = \|W^{-1}\|_\infty$ so,

$$\frac{\|\hat{x} - x\|_\infty}{\|x\|_\infty} \leqslant K_\infty(W)\frac{\|\delta b\|_\infty}{\|b\|_\infty} \leqslant 4488 \times \frac{0.1}{33} \simeq 13.6$$

This is exactly equal to the biggest error we found, so a very realistic estimate. For this example, the bound provides a good estimate of the scale of any change in the solution.

Theorem 7. [**?**, p. 105, Theorem 2.3.18] If A is nonsingular, and $\frac{\|\delta A\|}{\|A\|} < \frac{1}{\kappa(A)}$, $Ax = b$, and $(A + \delta A)(x + \delta x) = b$, then

$$\frac{\|\delta x\|}{\|x\|} \leqslant \frac{\kappa(A)\frac{\|\delta A\|}{\|A\|}}{1 - \kappa(A)\frac{\|\delta A\|}{\|A\|}} \tag{2.47}$$

Proof. The equation

$$(A + \delta A)(x + \delta x) = b$$

can be rewritten as

$$Ax + A\delta x + \delta A(x + \delta x) = b$$

using the fact that $Ax = b$ and rearranging the terms, we find that

$$\delta x = -A^{-1}\delta A(x + \delta x)$$

using the various properties of the vector norm and its induced matrix norm, we have that

$$\|\delta x\| \leqslant \|A^{-1}\|\|\delta A\|(\|x\| + \|\delta x\|) = \kappa(A)\frac{\|\delta A\|}{\|A\|}(\|x\| + \|\delta x\|)$$

now rewrite the inequality as

$$(1 - \kappa(A)\frac{\|\delta A\|}{\|A\|})\|\delta x\| \leqslant \kappa(A)\frac{\|\delta\|}{\|A\|}\|x\|$$

The assumption $\frac{\|\delta A\|}{\|A\|} < \frac{1}{\kappa(A)}$ guarantees that the factor that multiplies $\|\delta x\|$ is positive, so we can divide by it without reversing the inequality. If we also divide through by $\|x\|$, we get

$$\frac{\|\delta x\|}{\|x\|} \leqslant \frac{\kappa(A)\frac{\|\delta A\|}{\|A\|}}{1 - \kappa(A)\frac{\|\delta A\|}{\|A\|}}$$

□

Theorem 8. [8, p. 301, Corollary 5.6.16] Let $A \in \mathbb{M}_n$, and assume that $\rho(A) < 1$. Then there exists a submultiplicative norm $\|\cdot\|$ on $\mathbb{M}_n$ such that if $\|A\| < 1$ then $I - A$ is invertible and

$$(I - A)^{-1} = \sum_{k=0}^{\infty} A^k \tag{2.48}$$

$$\frac{1}{1 + \|A\|} \leqslant \|(I - A)^{-1}\| \leqslant \frac{1}{1 - \|A\|} \tag{2.49}$$

Proof. Since $\rho(A) < 1$, we deduce that $I - A$ is nonsingular. Assume $I - A$ is singular. Take $z \neq 0$. Then

$$\|(I - A)z\| = \|z - Az\| \geqslant \|z\| - \|Az\| \geqslant (1 - \|A\|)\|z\|$$

follows for all z. From $(1 - \|A\|) > 0$ it follows that $\|(I - A)z\| > 0$ for $z \neq 0$; that is, $(I - A)z = 0$ has only the trivial solution $x = 0$, and thus $(I - A)$ is nonsigular. Note that

$$(I - A)\sum_{k=0}^{\infty} A^k = \sum_{k=0}^{\infty} A^k - \sum_{k=1}^{\infty} A^k = I + \sum_{k=1}^{\infty} A^k - \sum_{k=1}^{\infty} A^k = I$$

Hence

$$(I + A)^{-1} = \sum_{k=0}^{\infty} A^k$$

Since $\|A\| < 1$. For any submultiplicative norm. Then

$$(I - A)^{-1} = \sum_{k=0}^{\infty} A^k$$

so

$$\|(I - A)^{-1}\| = \|\sum_{k=0}^{\infty} A^k\| \leqslant \sum_{k=0}^{\infty} \|A^k\| \leqslant \sum_{k=0}^{\infty} \|A\|^k = (I - \|A\|)^{-1}$$

Thus proving the right hand inequality of (2.49)

$$\|(I - A)^{-1}\| \leqslant \frac{1}{(I - \|A\|)} \tag{2.50}$$

Furthermore, the equality $\|I\| = 1$ holds, so that

$$\begin{aligned} 1 = \|I\| = \|(I - A)(I - A)^{-1}\| &\leqslant \|(I - A)(I - A)^{-1}\| \\ &\leqslant \|I - A\|\|(I - A)^{-1}\| \\ &\leqslant \|(I + \|A\|)\|(1 - A)^{-1} \\ \frac{1}{(I + \|A\|)} &\leqslant \|(I - \|A\|)^{-1} \end{aligned} \tag{2.51}$$

which yields the left-hand inequality in (2.49). Hence,

$$\frac{1}{1 + \|A\|} \leqslant \|(I - A)^{-1}\| \leqslant \frac{1}{1 - \|A\|}$$

□

Lemma 2. *[?, p.212, Lemma 4.4.14] Assume that $A \in \mathbb{C}^{n\times n}$ satisfies $\|A\| < 1$ in some induced matrix norm. Then $I + A$ is invertible and*

$$\|(I+A)^{-1}\| \leqslant \frac{1}{1-\|A\|}$$

Proof. For all $x \in \mathbb{C}^n$, $x \neq 0$

$$\|(I+A)x\| = \|x + Ax\|$$

$$\geqslant \|x\| - \|Ax\| \geqslant \|x\| - \|A\|\|x\| = \|x\|(1-\|A\|) \geqslant 0$$

Thus $I + A$ is invertible.
Moreover, the equality $\|I\| = 1$ holds, so that

$$1 = \|I\| = \|(I+A)^{-1}(I+A)\|$$

$$= \|(I+A)^{-1} + (I+A)A\|$$

$$1 \geqslant \|(I+A)^{-1}\| - \|(I+A)^{-1}A\|\|A\|$$

$$1 \geqslant \|(I+A)^{-1}\|(1-\|A\|)$$

consequently,

$$\|(I+A)^{-1}\| \leqslant \frac{1}{1-\|A\|}.$$

□

Let us consider an example

Example 10. *Consider the matrix,*

$$B = \begin{pmatrix} 1 & 1/4 & 1/4 \\ 1/4 & 1 & 1/4 \\ 1/4 & 1/4 & 1 \end{pmatrix}$$

then

$$A = B - I = \begin{pmatrix} 0 & 1/4 & 1/4 \\ 1/4 & 0 & 1/4 \\ 1/4 & 1/4 & 0 \end{pmatrix}$$

Note: If $\|A\| \leqslant 1$ then $\|B^{-1}\| = \|(I+A)^{-1}\| \leqslant \frac{1}{1-\|A\|}$

Using the infinity norm, $\|A\|_\infty = \frac{1}{2} < 1$*, so*

$$\|B^{-1}\|_\infty \leqslant \frac{1}{1-1/2} = 2$$

and$\|B\|_\infty = 1.5$*. Hence,*

$$K_\infty(B) = \|B\|_\infty \|B^{-1}\|_\infty \leqslant 1.5 \times 2$$

or $K_\infty(B) \leqslant 3$*, which is quite modest.*

Theorem 9. [**?**, p. 106, Theorem 2.3.20] If A is nonsingular,

$$\frac{\|\delta A\|}{\|A\|} < \frac{1}{\kappa(A)}, \;\; Ax = b \;\; and \;\; (A+\delta A)(x+\delta x) = b + \delta b,$$

then

$$\frac{\|\delta x\|}{\|x\|} \leqslant \frac{\kappa(A)(\frac{\|\delta A\|}{\|A\|} + \frac{\|\delta b\|}{\|b\|})}{1 - \kappa(A)\frac{\|\delta A\|}{\|A\|}} \tag{2.52}$$

Proof. Consider $Ax = b$, and the perturbed linear system

$$(A+\delta A)(x+\delta x) = b + \delta b. \tag{2.53}$$

Note that

$$\|A^{-1}\delta A\| \leqslant \|A^{-1}\|\|\delta A\| \leqslant \kappa(A)\frac{\|\delta A\|}{\|A\|} < 1$$

Then due to Lemma 2, $(I + A^{-1}\delta A)$ is invertible and it follows that

$$\|I + A^{-1}\delta A\| \leqslant \frac{1}{1 - \|A^{-1}\delta A\|} \leqslant \frac{1}{1 - \|A^{-1}\|\|\delta A\|} \tag{2.54}$$

on the other hand, solving for δx in equation (2.53) and recalling that $Ax = b$ one gets.

$$\delta x = (1 + A^{-1}\delta A)A^{-1}(\delta b - \delta A x) \tag{2.55}$$

from which, passing to the norm and using equation (2.54) it follows that

$$\|\delta x\| \leqslant \frac{\|A^{-1}\|}{1 - \|A^{-1}\|\|\delta A\|}(\|\delta b\| + \|\delta A\|\|x\|)$$

finally dividing both sides by $\|x\|$ (which is nonzero since $\|b\| \neq 0$ and A is nonsingular) and noticing that $\|x\| \geqslant \frac{\|b\|}{\|A}$ and hence the result follows

$$\frac{\|\delta x\|}{\|x\|} \leqslant \frac{\kappa(A)(\frac{\|\delta b\|}{\|b\|} + \frac{\|\delta A\|}{\|A\|})}{1 - \kappa(A)\frac{\|\delta A\|}{A\|}}$$

□

Summary

1. $\kappa(A)$ is defined for non-singular matrix
2. $\kappa(A) \geqslant 1$ for all A
   since $1 = \|I\| = \|AA^{-1}\| \leqslant \|A\|\|A^{-1}\| = \kappa(A)$
3. A is a *well-conditioned* matrix if $\kappa(A)$ is small(close to 1), the relative error in x is not much larger than the relative error in b.
4. A is badly condition or ill-condition if $\kappa(A)$ is large, the relative error in x can be much larger than the relative error in b.
5. The matrix A is said to be perfectly conditioned if $\kappa(A) = 1$.
6. The condition number depends on the matrix A and particular operator norm being used.
7. it is usual to define $\kappa(A) = \infty$ whenever A is not invertible.
8. $\kappa(AB) \leqslant \kappa(A)\kappa(B)$ for any pair of $n \times n$ matrices A and B.
9. some quick facts about $\kappa(A)$
   (a) $\kappa(\alpha A) = \kappa(A)$ where α is a nonzero scalar.
   (b) $\kappa(A^{-1}) = \kappa(A)$
   (c) $\kappa(A^T) = \kappa(A)$
   (d) $\kappa(I) = 1$
   (e) for any diagonal matrix D
   $$\kappa(D) = \frac{max|d_i|}{min|d_i|}$$
   (f) In the 2-norm, orthogonal matrices are perfectly conditioned in that $\kappa_2(Q) = 1$ if Q is orthogonal. i.e $Q^TQ = I$
10. Any two condition numbers $\kappa_\alpha(.)$ and $\kappa_\beta(.)$ on $\mathbb{R}^{n\times n}$ are equivalent in that constants c_1 and c_2 can be found for which
    $$c_1\kappa_\alpha(A) \leqslant \kappa_\beta(A) \leqslant c_2\kappa_\alpha(A) \qquad A \in \mathbb{R}^{n\times n} \tag{2.56}$$

For example, on $\mathbb{R}^{n\times n}$ we have

$$\frac{1}{n}\kappa_2(A) \leqslant \kappa_1(A) \leqslant n\kappa_2(A) \tag{2.57}$$

$$\frac{1}{n}\kappa_\infty(A) \leqslant \kappa_2(A) \leqslant n\kappa_\infty(A) \tag{2.58}$$

$$\frac{1}{n^2}\kappa_1(A) \leqslant \kappa_\infty(A) \leqslant n^2\kappa_1(A) \tag{2.59}$$

Thus, if a matrix is ill-conditioned in the α-norm, it is ill-conditioned in the β-norm modulo the constants c_1 and c_2 above.

2.8 Eigenvalues and Eigenvectors

Many practical problems in engineering and physics lead to eigenvalue problems. In this section, we present the basic facts about eigenvalues and eigenvectors. From a geometrical viewpoint, the eigenvector indicate the directions of pure stretch and the eigenvalues the extent of stretching. Most matrices are complete, meaning that their (complex) eigenvectors form a basis of the underlying vector space. A particularly important class are the symmetric matrices whose eigenvectors form an orthogonal basis of $\mathbb{C}^n$. A non-square matrix A does not have eigenvalues. The numerical computation of eigenvalues and eigenvectors is a challenging issue in numerical linear algebra. In other words, to diagonalize a square matrix.

To find the root λ_i, we have to compute the characteristic polynomial $p(\lambda) = det(A-\lambda_i I)$ and then $(A-\lambda_i)x = 0$ for the eigenvectors. However, this procedure is not satisfactory. We discuss two methods. The power method and the QR iteration will find only the eigenvalues but for general matrices. For the eigenvectors we still have to solve the equations $(A - \lambda_i)x = 0$ one by one.

We inaugurate our discussion of eigenvalues and eigenvectors with the basic definition.

Definition 15. Let $A \in \mathbb{C}^{n\times n}(or\mathbb{R}^{n\times n})$ be a square matrix. A scalar λ is called an *eigenvalue* of A if there is a *non-zero* vector $x \neq 0$, called an *eigenvector,* such that

$$Ax = \lambda x \quad x \neq 0 \tag{2.60}$$

In other words, the matrix A stretches the eigenvector x by an amount specified by the eigenvalue λ. The requirement that the eigenvector x be nonzero is important, since $x = 0$ is a trivial solution to the eigenvalue equation (2.60) for any scalar λ. The eigenvalue equation (2.60) is a system of linear equations for the entries of the eigenvector x provided that the eigenvalue λ is specified in advance.

Let us begin by rewriting the equation in the form

$$(A - \lambda I)x = 0 \tag{2.61}$$

where I is the identity matrix of the correct size. Now, for given λ equation (2.61) is a homogeneous linear system for x and always has

the trivial zero solution $x = 0$. But we are specifically seeking a nonzero solution.

A homogeneous linear system has a nonzero solution x if and only if its coefficient matrix, which is in this case is $A - \lambda I$, is singular. This observation is the key to resolving the eigenvector equation. An eigenvector is a special vector that is mapped by A into a vector parallel to itself. The length is increased if $|\lambda| > 1$ and decreased if $|\lambda| < 1$. The set of distinct eigenvalues is called the spectrum of A and is denoted by $\sigma(A)$.

Theorem 10. [**?**, p. 205, Theorem 4.2.3] For any $A \in \mathbb{C}^{n\times n}$ we have $\lambda \in \sigma(A)$, if and only if

$$det(A - \lambda I) = 0.$$

Proof. Suppose (λ, x) is an eigenpair for A. The equation $Ax = \lambda x$ can be written $(A - \lambda I)x = 0$. Since x is nonzero the matrix $A - \lambda I$ must be singular with a zero determinant.

Conversely, if $det(A - \lambda I) = 0$ then $A - \lambda I$ is singular and $(A - \lambda I)x = 0$ for some nonzero $x \in \mathbb{C}^{n\times n}$. Thus $Ax = \lambda x$ and (λ, x) is an eigenpair for A. □

2.8.1 The characteristic polynomial

The eigenvalue-eigenvector equation(2.60) can be rewritten as

$$(A - \lambda I)x = 0, \;\; x \neq 0 \tag{2.62}$$

Thus, $\lambda \in \sigma(A)$ if and only if $A - \lambda I$ is a singular matrix, that is

$$det(A - \lambda I) = 0 \tag{2.63}$$

equation (2.63) is called the *characteristic equation* of A and $det(A-\lambda I)$ is called the *characteristic polynomial* of A, is defined by

$$p_A(\lambda) = det(A - \lambda I) \tag{2.64}$$

The fact that $p_A(\lambda)$ is a polynomial of degree n whose leading term is $(-1)^n\lambda^n$ comes from the diagonal product $\prod_{k=1}^{n}(a_{kk} - \lambda)$. Since $p_A(\lambda)$ has degree n, it follows from (2.64) and Fundamental Theorem of

Algebra that *every $n \times n$ matrix A has exactly n (possibly repeated and possibly complex) eigenvalues, namely the roots of $p_A(\lambda)$*. If we denote them by $\lambda_1, \lambda_2, ..., \lambda_n$, then $p_A(\lambda)$ can be factored as

$$p_A(\lambda) = (-1)^n(\lambda - \lambda_1)(\lambda - \lambda_2)...(\lambda - \lambda_n) = \prod_{j=1}^{n}(\lambda_j - \lambda) \qquad (2.65)$$

If we take $\lambda = 0$ in (2.64) and (2.65) we see that

$$det(A) = \lambda_1\lambda_2...\lambda_n = constant\ term\ of\ \ p_A(\lambda) \qquad (2.66)$$

So *A is invertible if and only if all eigenvalues are nonzero*, in which case multiplying (2.60) by $\lambda^{-1}A^{-1}$ gives $A^{-1}x = (1/\lambda)x$. Thus

$$(\lambda, x) is\, an\, eigenpair\, for\, A \iff (\frac{1}{\lambda}, x) is\, an\, eigenpair\, for\, A^{-1} \quad (2.67)$$

To illustrate the concept of eigenvalues and eigenvectors let us see the following examples.

Example 11. *Consider the 3×3 matrix*

$$A = \begin{pmatrix} 3 & -6 & -7 \\ 1 & 8 & 5 \\ -1 & -2 & 1 \end{pmatrix}$$

Then, expanding the determinant, we find

$$det(A - \lambda I) = 0$$

$$\lambda^3 - 12\lambda^2 - 44\lambda + 48 = 0$$

This can be factored as

$$\lambda^3 - 12\lambda^2 - 44\lambda + 48 = (\lambda - 2)(\lambda - 4)(\lambda - 6) = 0$$

so the eigenvalues are 2, 4 and 6.
For each eigenvalue, the corresponding eigenvectors are found by solving the associated homogenous linear system equ(2.15)*. For the first eigenvalue, the eigenvector equation is*

$$(A - 2I)x = \begin{pmatrix} 1 & -6 & -7 \\ 1 & 6 & 5 \\ -1 & -2 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

or

$$x_1 - 6x_2 - 7x_3 = 0$$
$$x_1 + 6x_2 + 5x_3 = 0$$
$$-x_1 - 2x_2 - x_3 = 0$$

The simplified form is

$$2x_1 - 2x_3 = 0$$
$$4x_2 + 4x_3 = 0$$

so

$$X_1 = \begin{pmatrix} x_3 \\ -x_3 \\ x_3 \end{pmatrix} = x_3 \begin{pmatrix} 1 \\ -1 \\ 1 \end{pmatrix}$$

and so as eigenvector for the eigenvalue $\lambda_1 = 2$ *is*

$$X_1 = \begin{pmatrix} 1 \\ -1 \\ 1 \end{pmatrix}$$

These steps can be done with MATLAB using poly and root. If A is a square matrix, the command poly(A) computes the characteristic polynomial, or rather, its coefficients.

```
>> A=[3 -6 -7; 1 8 5; -1 -2 1];
>> p=poly(A)

p =

    1.0000  -12.0000   44.0000  -48.0000

Recall that the coefficient of the highest power comes first.
The function roots takes as input a vector representing
the coefficients of a polynomial and returns the roots.

>> roots(p)

ans =

    6.0000
    4.0000
    2.0000
```

```
To find the eigenvector(s) for eigenvalue 2, we must
solve the homogenous equation (A-2I)x=0.
Recall that eye(n) is the n x n identity matrix I

>> rref(A-2*eye(3))

ans =

     1     0    -1
     0     1     1
     0     0     0
```

From this we can read off the solution

$$X_1 = \begin{pmatrix} 1 \\ -1 \\ 1 \end{pmatrix}$$

Similarly we find for $\lambda_2 = 4$ *and* $\lambda_3 = 6$ *that the corresponding eigenvectors are*

$$X_2 = \begin{pmatrix} -1 \\ -1 \\ 1 \end{pmatrix}, X_3 = \begin{pmatrix} -2 \\ 1 \\ 0 \end{pmatrix}$$

The three eigenvectors X_1,X_2 *and* X_3 *are lineraly independent and form a basis for* $\mathbb{R}^3$.

The MATLAB command for finding eigenvalues and eigenvectors is eig. The command eig(A) lists the eigenvalues

```
>> eig(A)

ans =

    4.0000
    2.0000
    6.0000
```

while the variant $[X, D] = eig(A)$ returns a matrix X whose columns are eigenvectors and a diagonal matrix D whose diagonal entries are the eigenvalues.

```
>> [X,D]=eig(A)

X =

    0.5774    0.5774   -0.8944
    0.5774   -0.5774    0.4472
   -0.5774    0.5774    0.0000


D =

    4.0000         0         0
         0    2.0000         0
         0         0    6.0000
```

Notice that the eigenvalues have been normalized to have length one. Also, since they have been computed numerically, they are not exactly correct.

2.9 The Multiplicities of an Eigenvalues

Let $A \in \mathbb{M}_n$ and the characteristic polynomial $p_A(\lambda)$ is

$$p_A(\lambda) = \prod_{j=1}^{n} (\lambda_j - \lambda) \tag{2.68}$$

where $\lambda_j \in \sigma(A) = \{\lambda_1, \lambda_2, ..., \lambda_n\}$

Definition 16. For a given $\lambda \in \sigma(A)$, the set $\{x \neq 0 | x \in N(A - \lambda I)\}$ of all vectors $x \in \mathbb{C}^n$ satisfying $Ax = \lambda x$ is called the *eigenspace* of A corresponding to the eigenvalue λ. Note that every nonzero element of this eigenspace is an eigenvector of A corresponding to λ.

Definition 17. The *algebriac multiplicity* of a λ is the number of times the factor $(\lambda_j - \lambda)$ appears in equation (2.68). In other words, $alg\, mult_A(\lambda_j) = \alpha_j$ if and only if $(\lambda_1 - \lambda)^{\alpha_1}(\lambda_2 - \lambda)^{\alpha_2}...(\lambda_n - \lambda)^{\alpha_n} = 0$ is the characteristic equation of A.

Definition 18. The *geometric multiplicity* of λ is $\dim N(A - \lambda I)$. In other word, $geo\, mult_A(\lambda)$ is the maximal number of linearly independent eigenvectors associated with λ.

Let us illustrate an example.

Example 12. *Consider the matrix*

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$$

we compute the characteristic polynomial $p_A(\lambda)$ of A is

$$p_A(\lambda) = det(A - \lambda I) = \lambda^2$$

has only one distinct eigenvalue, $\lambda = 0$, that is repeated twice, so

$$alg\, multiplicity_A(\lambda) = alg\, multiplicity_A(0) = 2.$$

But

$$dim N(A - \lambda I) = dim N(A - 0I) = dim N(A) = 1$$
$$\Rightarrow geo\, mult_A(0) = 1$$

Hence, in other words, there is only one linearly independent eigenvector associated with $\lambda = 0$ even though $\lambda = 0$ is repeated twice as an eigenvalue.

Chapter 3

Matrix Functions

The concept of a matrix function play a widespread role in many areas of linear algebra and has numerous applications in science and engineering, especially in control theory and, more generally, differential equations where $exp(At)$ play prominent role.

For example, *Nuclear magnetic resonance:* Solomon equations

$$dM/dt = -RM, \quad M(0) = I$$

where $M(t)$ =matrix of intensities and R =symmetric relaxation matrix. In the case $n = 1$ the series is $M = e^{-tR}$ with $M, R, I \in \mathbb{M}_n$. Where e^{-tR} is suitably defined for $R \in \mathbb{M}_n$.

Matrix functions in general are an interesting area in matrix analysis. A matrix function can have various meanings, but we will be only concerned with a definition that is based on a scalar function, f. Given a scalar function f of the scalar x defined in terms of $+, -, \times, \div$. We define a matrix value function of $A \in \mathbb{M}_n$, by replacing each occurrence of x by A. The resulting function of A is a $n \times n$ matrix.
For example,

1. if $f(x) = x^2$, we have $f(A) = A^2$;

2. if $f(x) = \frac{x+1}{x-1}$, $x \neq 0$ then we have $f(A) = (A + I)(A - I)^{-1}$ if $1 \notin \Lambda(A)$
3. if $f(x) = \frac{1+x^2}{1-x}$ $x \neq 0$ then we have $f(A) = (I + A^2)(I - A)^{-1}$ if $1 \notin \Lambda(A)$

where $\Lambda(A)$ denotes the set of eigenvalues of A, which is called the *spectrum* of A.

Similarly scalar functions defined by a power series extend to matrix functions, such as

$$If\ \ f(x) = log(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + ...,$$

Then

$$f(A) = log(1+A) = A - \frac{A^2}{2} + \frac{A^3}{3} - \frac{A^4}{4} + ...,$$

One can show that this series converges $\Leftrightarrow \rho(A) < 1$.

Many power series have an infinite radius of convergence such as

$$cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - ...$$

This generates

$$cos(x) = I - \frac{A^2}{2!} + \frac{A^4}{4!} - ...$$

Again one can show that this converges for all $A \in \mathbb{M}_n$.

This approach to defining a matrix function is sufficient for a wide rage of functions, but it does not provide a definition for a general matrix function, also it does not necessarily provide a good way to numerically evaluate $f(A)$. So will consider alternate definitions.

3.1 Definitions of $f(A)$

A function of a matrix can be defined in several ways, of which the following three are the most generally useful.

3.1.1 Jordan Canonical Form

Most problems related to a matrix A can be easily solved if the matrix is diagonalizable. But as we have seen with

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$$

not every square matrix is diagonalizable over $\mathbb{C}$ or ($\mathbb{R}$). However by using similarity transformations every square matrix can be transformed

to a matrix which is "nearly diagonal" in a certain sense. This nearly diagonal matrix is known as the *Jordan Canonical Form* and is important both for theoretical purpose and practical applications. Let us consider the following example.

Example 13. *Consider the matrix*

$$A = \begin{pmatrix} 2 & 3 \\ 0 & 2 \end{pmatrix}.$$

The eigenvalues are $\lambda_1 = \lambda_2 = 2$*, but only one linearly independent eigenvector.*

$$\begin{pmatrix} 2 & 3 \\ 0 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = 2 \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$$3x_2 = 0$$

and the corresponding eigenvector is

$$x = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Definition 19. A *Jordan block* $J_k(\lambda)$ is a $k \times k$ upper triangular matrix which has square matrices of various sizes placed across the diagonal and every diagonal entry is λ, for some $\lambda \in \mathbb{C}$, and every entry just above the main diagonal is 1. All other entries are 0.

The matrix

$$J_k(\lambda_k) = \begin{pmatrix} \lambda_k & 1 & & \\ & \lambda_k & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_k \end{pmatrix} \in \mathbb{C}^{m_k \times m_k} \tag{3.1}$$

is called a *Jordan block* of size m_k with eigenvalues λ_k and $n = \sum_{k=1}^{p} m_p$ The scalar λ appears k times on the main diagonal and $+1$ appears $(k-1)$ times on the superdiagonal. For example.

– a Jordan block of size 1 is simply the 1×1 matrix is

$$J_1(\lambda) = \begin{pmatrix} \lambda \end{pmatrix}$$

– Jordan block of size 2 is simply the 2×2 matrix is

$$J = \begin{pmatrix} \lambda & 1 \\ 0 & \lambda \end{pmatrix}$$

– a Jordan block of size 3 is simply the 3×3 matrix is

$$J = \begin{pmatrix} \lambda & 1 & 0 \\ 0 & \lambda & 1 \\ 0 & 0 & \lambda \end{pmatrix}.$$

Example 14. *The following are in Jordan Canonical Form:*

$$J_2 = \begin{pmatrix} 4 & 1 \\ 0 & 4 \end{pmatrix}, \quad J_2 = \begin{pmatrix} 2 & 0 \\ 0 & 6 \end{pmatrix}, \quad J_2 = \begin{pmatrix} 3 & 0 \\ 0 & 3 \end{pmatrix}$$

$$J_3 = \begin{pmatrix} 2 & 1 & 0 \\ 0 & 2 & 1 \\ 0 & 0 & 2 \end{pmatrix}, \quad J_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 10 \end{pmatrix}$$

$$J_3 = \begin{pmatrix} 7 & 0 & 0 \\ 0 & -8 & 0 \\ 0 & 0 & -8 \end{pmatrix}, \quad J_3 = \begin{pmatrix} 5 & 1 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & -9 \end{pmatrix}$$

Definition 20. Any matrix $A \in \mathbb{C}^{n \times n}$ (not necessarily diagonalizable) can be expressed in the **Jordan canonical form**

$$X^{-1}AX = J = diag(J_1(\lambda_1), J_2(\lambda_2), ...J_p(\lambda_p)) \tag{3.2}$$

$$J = \begin{pmatrix} J_1 & & & \\ & J_2 & & \\ & & \ddots & \\ & & & J_k \end{pmatrix}.$$

Where X is nonsingular and $m_1 + m_2 + ... + m_p = n$. Where

$$J_k = J_k(\lambda_k) = \begin{pmatrix} \lambda_k & 1 & & \\ & \lambda_k & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_k \end{pmatrix} \in \mathbb{C}^{m_k \times m_k} \tag{3.3}$$

and λ_k are the eigenvalues of A. The Jordan matrix J is unique up to the ordering of the blocks J_k, but the transforming matrix X is not unique.

Observations and applications

[8, p. 129 §3.2.1]

1. The number k of Jordan blocks is the number of linearly independent eigenvectors of J.
2. The matrix J is diagonalizble if and only if $k = n$.
3. The number and dimensions of the *Jordan matrix* $J \in \mathbb{C}^{m\times m}$ is a direct sum of Jordan blocks.
4. A Jordan matrix is not completely determined in general by a knowledge of the eigenvalues and the dimension of their generalized and standard eigenspaces. One must also know the sizes of the Jordan blocks corresponding to each eigenvalue.
5. The size of the largest Jordan block corresponding to an eigenvalue λ is the multiplicity of λ as a root of the minimal polynomial.
6. The size of the Jordan blocks corresponding to a given eigenvalue are determined by a knowledge of the ranks of certain powers. For example, if

$$J = \begin{pmatrix} 2 & 1 & 0 & & & & & \\ 0 & 2 & 1 & & & & & \\ 0 & 0 & 2 & & & & & \\ & & & 2 & 1 & & & \\ & & & 0 & 2 & & & \\ & & & & & 2 & 1 & \\ & & & & & 0 & 2 & \\ & & & & & & & 2 \end{pmatrix}$$

then

$$J - 2I = J = \begin{pmatrix} 0 & 1 & 0 & & & & & \\ 0 & 0 & 1 & & & & & \\ 0 & 0 & 0 & & & & & \\ & & & 0 & 1 & & & \\ & & & 0 & 0 & & & \\ & & & & & 0 & 1 & \\ & & & & & 0 & 0 & \\ & & & & & & & 0 \end{pmatrix}$$

$$(J-2I)^2 = \begin{pmatrix} 0 & 0 & 1 & & & & & \\ 0 & 0 & 0 & & & & & \\ 0 & 0 & 0 & & & & & \\ & & & 0 & 0 & & & \\ & & & 0 & 0 & & & \\ & & & & & 0 & 0 & \\ & & & & & 0 & 0 & \\ & & & & & & & 0 \end{pmatrix}$$

and $(J-2I)^3 = 0$. Thus we know that

$$(J-2I)^3 = 0 \qquad rank(J-2I)^2 = 1 \qquad rank(J-2I) = 4$$

This list of numbers is sufficient to determine the block structure of J. The fact that $(J-2I)^3$ tells us that the largest block has order 3. The rank of $(J-2I)^2$ will be the number of blocks of order 3, so there is only one. The rank of $(J-2I)$ is the twice the number of blocks of order 3 plus the number of blocks of order 2, so there are two of them. The number of blocks of order 1 is 8-(2×2)-3=1. A similar procedure can be applied to direct sums of Jordan blocks of any size. if J is such a direct sum corresponding to the eigenvalue λ, then the smallest integer k_1 such that $(J-\lambda I)^{k_1} = 0$ is the size of the largest block. The rank of $(J-\lambda I)^{k_1-1}$ is the number of blocks of order k_1, the rank of $(J-\lambda I)^{k_1-2}$ is twice the number of blocks of order k_1 plus the number of blocks of size k_1-1, and so forth.The sequence of ranks of $(J-\lambda I)^{k_1-i}, \quad i = 0, 1, 2, ..., k_1-1$, recursively determines the orders of all the blocks in J.

7. The Jordan canonical form of a matrix can be used to compute powers of a matrix, even if the matrix is not diagonalizable. For example, if

$$A = XJX^{-1}$$

$$A^2 = XJX^{-1}XJX^{-1} = XJ^2X^{-1}$$

and, in general,

$$A^m = XJ^mX^{-1}$$

It is easy to compute powers of a Jordan block J_k. We have

$$J_k(\lambda) = (\lambda I + N) \in \mathbb{M}_k$$

where

$$N \equiv J_k(0) = \begin{pmatrix} 0 & 1 & & 0 \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ 0 & & & 0 \end{pmatrix}$$

Since $N_k^m = 0$ for all $m \geqslant k$

$$J_k^m(\lambda) = (\lambda I + N_k)^m = \sum_{i=0}^{m} \frac{m!}{i!(m-i)!} \lambda^{m-i} N_k^i$$

Which yields for $m \geqslant n_k$.

$$J_k^m(\lambda) = \begin{pmatrix} \lambda^m & C_1^m \lambda^{m-1} & \cdots & C_{n_k-1}^{n_k-1} \lambda^{m-(n_k-1)} \\ & \lambda^m & \ddots & \vdots \\ & & \ddots & \vdots \\ & & & \lambda^m \end{pmatrix}$$

where $C_r^n = \frac{n!}{r!(n-r)!}$

For example

$$J_k^3(\lambda) = \begin{pmatrix} \lambda & 1 & 0 \\ 0 & \lambda & 1 \\ 0 & 0 & \lambda \end{pmatrix}^3 = \begin{pmatrix} \lambda^3 & 3\lambda^2 & 3\lambda \\ 0 & \lambda^3 & 3\lambda^2 \\ 0 & 0 & \lambda^3 \end{pmatrix}$$

8. Using the *Jordan Canonical Form* to solve Linear systems of ODEs. Let $A \in M_n$ be given, and consider the system of first order differential equations

$$y'(t) = Ay(t), \quad y(0) = y_0 \tag{3.4}$$

Using the *Jordan* form, we can rewrite eq(2) as

$$y'(t) = XJX^{-1}y(t).$$

Multiplying through by X^{-1} yields

$$X^{-1}y'(t) = JX^{-1}y(t),$$

which can be rewritten as

$$z'(t) = Jz(t) \tag{3.5}$$

where $z(t) = X^{-1}y(t)$. The new initial condition is

$$z(0) = z_0 = X^{-1}y_0.$$

If we assume that J is a diagonal matrix (because A has a full set of linearly independent eigenvectors), then the system decouples into scalar equations of the form

$$z_k^{'} = \lambda_k z_k(t) \tag{3.6}$$

where λ_k is an eigenvalue of A. This equation has the solution

$$z_k(t) = e^{\lambda_k t} z_k(0),$$

$$X^{-1}y_k(t) = e^{\lambda_k t} X^{-1} y_k(0)$$
$$y_k(t) = X e^{\lambda_k t} X^{-1} y_k(0)$$

$$y(t) = X \begin{pmatrix} e^{\lambda_1 t} & & \\ & \ddots & \\ & & e^{\lambda_k t} \end{pmatrix} X^{-1} y(0)$$

If the eigenvalue λ_k is real, this is a simple exponential, if $\lambda_k = a_k + ib_k$ is complex, it is an oscillatory term

$$y_k(t) = X e^{a_k t}[cos(b_k t) + isin(b_k t)] X^{-1} y_k(0) \tag{3.7}$$

Definition 21. A nonzero vector x is said to be a *generalized eigenvector* of A of rank k associated with the eigenvalue λ if

$$(A - \lambda I_n)^k x = 0 \ and (A - \lambda I_n)^{k-1} x \neq 0.$$

Note that if $k = 1$, this is the usual definition of an eigenvector. For a generalized eigenvector x of rank $k \geqslant 1$ belonging to an eigenvalue λ, define

$$x_k = x,$$
$$x_{k-1} = (A_\lambda I)x_k = (A_\lambda I)x,$$
$$x_{k-2} = (A_\lambda I)x_{k-1} = (A_\lambda I)^2 x,$$
$$\vdots$$
$$x_2 = (A_\lambda I)x_3 = (A_\lambda I)^{k-2} x,$$
$$x_1 = (A_\lambda I)x_2 = (A_\lambda I)^{k-1} x,$$

This approach also work for when the Jordan canonical form of A is not diagonalizable. Here is a simple example. Take

$$A = \begin{pmatrix} \lambda & 1 \\ 0 & \lambda \end{pmatrix}$$

Then $J = A$, $X = I$, and

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

$$y(t) = e^{tA}y(0)$$

$$y(t) = \begin{pmatrix} e^{\lambda t} & e^{\lambda t} \\ 0 & e^{\lambda t} \end{pmatrix} \begin{pmatrix} y_1(0) \\ y_2(0) \end{pmatrix}$$

$$y(t) = e^{\lambda t} \begin{pmatrix} y_1(0) + y_2(0) \\ y_2(0) \end{pmatrix}$$

For the computation of Jordan Canonical Form, we illustrate an example.

Example 15.

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & 4 \\ 0 & 0 & 1 \end{pmatrix}$$

The characteristic polynomial

$$det(A - \lambda I) = (1 - \lambda)^3 = 0$$

Then the eigenvalues are $\lambda_1 = \lambda_2 = \lambda_3 = \lambda = 1$
the eigenvector x_1 *for* $\lambda_1 = 1$ *is*

$$(A - \lambda I)x_1 = \begin{pmatrix} 0 & 2 & 3 \\ 0 & 0 & 4 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

$$x_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Using the generalized eigenvectors formula

$$(A - \lambda_i I)x_k = x_{k-1}$$

Find x_2,

$$(A - \lambda_2 I)x_2 = x_1$$

$$\begin{pmatrix} 0 & 2 & 3 \\ 0 & 0 & 4 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

$$x_2 = \begin{pmatrix} 1 \\ 1/2 \\ 0 \end{pmatrix}$$

generate x_3 *from* x_2,

$$\begin{pmatrix} 0 & 2 & 3 \\ 0 & 0 & 4 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 1 \\ 1/2 \\ 0 \end{pmatrix}$$

$$x_3 = \begin{pmatrix} 1 \\ 5/16 \\ 1/8 \end{pmatrix}$$

Thus

$$X = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1/2 & 5/16 \\ 0 & 0 & 1/8 \end{pmatrix}$$

Hand calculation shows that

$$X^{-1} = \begin{pmatrix} 1 & -2 & -3 \\ 0 & 2 & -5 \\ 0 & 0 & 8 \end{pmatrix}$$

Thus we have

$$X^{-1}AX = J = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

which is the required Jordan Canonical Form.

3.1.2 Definition of a Matrix Function via The Jordan Canonical Form

Definition 22. Let f be defined on a neighborhood of the spectrum of $A \in \mathbb{C}^{n\times n}$ and let A have the *Jordan canonical form*. Then

$$f(A) = Xf(J)X^{-1} = Xdiag(f(J_k(\lambda_k)))X^{-1}, \tag{3.8}$$

where

$$f(J_k) = f(J_k(\lambda_k)) = \begin{pmatrix} f(\lambda_k) & f'(\lambda_k) & \cdots & \frac{f^{(m_k-1)}(\lambda_k)}{(m_k-1)!} \\ & f(\lambda_k) & \ddots & \vdots \\ & & \ddots & f'(\lambda_k) \\ 0 & & & f(\lambda_k) \end{pmatrix} \tag{3.9}$$

The R.H.S of (3.8) is independent of the choice of X and J.

If A is not normal, its eigenvalues are not necessarily well conditioned. It is also important to realize that the size of the Jordan blocks may widely vary under infinitesimal perturbations in A. Therefore, if A is not normal, then the definition does not provide a numerically stable means for computing $f(A)$.

A simple example illustrates the definition.

Example 16. *Consider the matrix*

$$A = \begin{pmatrix} 6 & 2 & 2 \\ -2 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix}$$

and $f(x) = log(x)$,
Since

$$|A - \lambda I| = (2-\lambda)(4-\lambda)^2,$$

it follows that $\sigma(A) = \{2, 4\}$, *the nonsingular matrix* X *is*

$$X = \begin{pmatrix} 0 & 2 & 0 \\ -1 & -2 & 1 \\ 1 & 0 & 0 \end{pmatrix}, \quad X^{-1} = \begin{pmatrix} 0 & 0 & 1 \\ 1/2 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}$$

and the Jordan form is

$$J = \begin{pmatrix} 2 & 0 & 0 \\ 0 & 4 & 1 \\ 0 & 0 & 4 \end{pmatrix}, \quad f(J) = \begin{pmatrix} log2 & 0 & 0 \\ 0 & log4 & \frac{1}{4} \\ 0 & 0 & log4 \end{pmatrix}$$

$$A = XJX^{-1}$$
$$f(A) = Xf(J)X^{-1}$$

Hence,

$$f(A) = \begin{pmatrix} 0 & 2 & 0 \\ -1 & -2 & 1 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} ln2 & 0 & 0 \\ 0 & ln4 & \frac{1}{4} \\ 0 & 0 & 4 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 \\ 1/2 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}$$

$$f(A) = \begin{pmatrix} 1.8863 & 0.5000 & 0.5000 \\ -0.5000 & 0.8863 & 0.1931 \\ 0 & 0 & 0.6931 \end{pmatrix}$$

3.2 Polynomial Matrix Function

Two polynomials are associated with every square matrix: the characteristic polynomial and the minimal polynomial. These polynomials play an important role in various problems of the theory of matrices.

Definition 23. A scalar polynomial $\psi(\lambda)$ is called an annihilating polynomial of the square matrix A if

$$\psi(A) = 0. \tag{3.10}$$

An annihilating polynomial $\psi(\lambda)$ of least degree with highest coefficient 1 is called a *minimal polynomial* of A. In fact it is unique.

Let $\psi_1(\lambda)$ and $\psi_2(\lambda)$ be two minimal polynomials of one and the same matrix. Then each is divisible without remainder by the other, i.e., the polynomials differ by a constant factor. This constant factor must be 1, because the highest coefficients in $\psi_1(\lambda)$ and $\psi_2(\lambda)$ are 1. Thus we have proved the uniqueness of the minimal polynomial of a given matrix A.

By the Hamilton-Cayley Theorem the characteristic polynomial $\chi(\lambda)$ is an annihilating polynomial of A. However, as we shall show by example below, it is not, in general, a minimal polynomial.

Lemma 3. *[?, p. 224, Theorem 1] Every annihilating polynomial of a matrix is divisible without remainder by the minimal polynomial.*

Proof. Let us divide an arbitrary annihilating polynomial $f(\lambda)$ by a minimal polynomial

$$f(\lambda) = \psi(\lambda)q(\lambda) + r(\lambda) \tag{3.11}$$

where the degree of $r(\lambda)$ is less than that of $\psi(\lambda)$. Hence we have,

$$f(A) = \psi(A)q(A) + r(A) \tag{3.12}$$

Since $f(A) = 0$ and $\psi(A) = 0$, it follows that $r(A) = 0$. But the degree of $r(\lambda)$ is less than that of the minimal polynomial $\psi(\lambda)$. Therefore $r(\lambda) \equiv 0$. □

Lemma 4. *[8, p. 144] The minimal polynomial of the Jordan block of order m with eigenvalue λ is $(t - \lambda)^m$.*

Proof. Let A be the matrix then

$$N = A - \lambda I$$

$$A - \lambda I = \begin{pmatrix} 0 & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ & & & 0 \end{pmatrix} \in M_n$$

Then

$$(A - \lambda I)^m = 0, \quad (A - \lambda I)^{m-1} \neq 0,$$

so it follows that $(t-\lambda)^m$ is an annihilating polynomial and none of its divisor is such therefore $(t - \lambda)^m$ is the minimal polynomial. □

Lemma 5. *Let $A \in \mathbb{C}^{n\times n}$ and suppose that $\lambda_1, \lambda_2, ..., \lambda_s$ are the distinct eigenvalues of A. Then the minimal polynomial, the unique monic polynomial p of lowest degree such that $p(A) = 0$ is defined to be*

$$\psi(\lambda) = \prod_{i=1}^{s} (\lambda - \lambda_i)^{n_i} \tag{3.13}$$

where n_i is the size of the largest Jordan block in which λ_i appears.

The proof is similar to Lemma 4.

Example 17. *Find the minimal polynomial of the matrix A, where*

$$A = \begin{pmatrix} 2 & 1 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 5 \end{pmatrix}.$$

A is an upper triangular matrix, so, the characteristic polynomial of the given matrix A is,

$$p_A(\lambda) = det(A - \lambda I)$$

$$p_A(\lambda) = (2 - \lambda)^3(5 - \lambda)$$

there are three possibilities for the minimal polynomial

$$\psi_1(\lambda) = (\lambda - 2)(\lambda - 5)$$

$$\psi_2(\lambda) = (\lambda - 2)^2(\lambda - 5)$$

$$\psi_3(\lambda) = (\lambda - 2)^3(\lambda - 5)$$

Substituting $\lambda = A$ in the polynomial $\psi_1(\lambda)$ yields

$$\psi_1(\lambda) = (A - 2I)(A - 5I) \neq 0$$

Therefore, $\psi_1(\lambda)$ can not be the minimum polynomial of A. Whereas

$$\psi_2(\lambda) = (A - 2I)^2(A - 5I) = 0$$

Since $\psi(\lambda) = 0$, this shows that $\psi_2(\lambda) = (\lambda-2)^2(\lambda-5)$ is the minimum polynomial of A.

Theorem 11. [8, p. 86, Theorem 2.4.2] Let A be a square $n \times n$ matrix and let $p_A(\lambda)$ be its characteristic polynomial i.e $p_A(\lambda) = det(\lambda I - A)$ then $p_A(A) = 0$.

This theorem says essentially that *" A matrix satisfies its own characteristic equation."* To illustrate the Theorem 11, we consider the following example.

Example 18.

$$A = \begin{pmatrix} 2 & 1 \\ -1 & 3 \end{pmatrix}$$

$$p(\lambda) = det(\lambda I - A) = \lambda^2 - 5\lambda + 7$$

$$p(A) = A^2 - 5A + 7$$

$$p(A) = \begin{pmatrix} 3 & 5 \\ -5 & 8 \end{pmatrix} - 5\begin{pmatrix} 2 & 1 \\ -1 & 3 \end{pmatrix} + 7\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$p(A) = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} = 0$$

Now we have an alternate method to evaluate matrix polynomials. Evaluating a matrix polynomial is a common task when working with matrix functions. Horner's method(or synthetic division) is almost always used when evaluating a scalar polynomial, but in the matrix case we must also consider alternative methods.

In fact, any scalar polynomial

$$p(t) = a_0 + a_1 t + ... + a_{m-1}t^{m-1} + a_m t^m = \sum_{i=0}^{m} a_i t^i \tag{3.14}$$

gives rise to a matrix polynomial with scalar coefficients by simply substituting A for t in (3.11)

$$p(A) = a_m A^m + a_{m-1}A^{m-1} + ... + a_0 I = \sum_{i=0}^{m} a_i A^i \tag{3.15}$$

More generally, for functions f with a series representation on an open disk containing the eigenvalues of A, we are able to define the matrix function $f(A)$ via the Taylor series for f.

Theorem 12. [5, p. 565, Theorem 11.2.3] If $f(t)$ has a power series representation

$$f(t) = \sum_{i=0}^{\infty} a_i t^i$$

on an open disk containing $\lambda(A)$, then

$$f(A) = \sum_{i=0}^{\infty} a_i A^i$$

Proof. When A is diagonalizable. Suppose

$$X^{-1}AX = D = diag(\lambda_1, \lambda_2, ..., \lambda_n)$$

we have

$$f(A) = Xdiag(f(\lambda_1), f(\lambda_2), ..., f(\lambda_n))X^{-1}$$

$$= Xdiag(\sum_{i=0}^{\infty} a_i\lambda_1^i, ..., \sum_{i=0}^{\infty} a_i\lambda_n^i)$$

$$= X(\sum_{i=0}^{\infty} a_i D^i)X^{-1} = (\sum_{i=0}^{\infty} a_i (XDX^{-1})^i$$

$$f(A) = \sum_{i=0}^{\infty} a_i A^i$$

□

To evaluate a polynomial matrix function if $A \in \mathbb{M}_n$ is diagonalizable with a diagonalization $A = X\Lambda X^{-1}$ with $\Lambda = diag(\lambda_1, \lambda_2, ..., \lambda_n)$ is known, then (3.9) assumes the simple form

$$p(A) = p(X\Lambda X^{-1}) = Xp(\Lambda)X^{-1}$$

$$p(A) = X \begin{pmatrix} p(\lambda_1) & & 0 \\ & \ddots & \\ 0 & & p(\lambda_n) \end{pmatrix} X^{-1} \tag{3.16}$$

In deriving this formula, we have used the important property of polynomial functions $p(t)$ that $p(XAX^{-1}) = Xp(A)X^{-1}$. If A is not necessarily diagonalizable and has Jordan canonical form $A = XJX^{-1}$ with

$$J_k = J_k(\lambda_k) = \begin{pmatrix} \lambda_k & 1 & & \\ & \lambda_k & \ddots & \\ & & \ddots & \ddots \\ & & & \lambda_k \end{pmatrix}$$

as in [8, (3.1.12), p. 126, Theorem 3.1.11] then

$$p(A) = p(XJX^{-1}) = Xp(J)X^{-1}$$

$$p(A) = X \begin{pmatrix} p(J_1(\lambda_1)) & & 0 \\ & \ddots & \\ 0 & & p(J_p(\lambda_p)) \end{pmatrix} X^{-1} \tag{3.17}$$

Now write

$$J_k(\lambda) = \lambda I + N \quad where \quad N \equiv J_k(0).$$

Then

$$J_k^j(\lambda) = (\lambda I + N)^j = \sum_{i=0}^{j} C_i^j \lambda^{j-i} N^i$$

and all terms with $i \geqslant k$ are zero because $N^k = 0$. This gives

$$p(J_k(\lambda)) = \sum_{j=0}^{m} a_j J_k(\lambda)^j = \sum_{j=0}^{m} \sum_{i=0}^{j} a_j \frac{j!}{i!(j-i)!} \lambda^{j-i} N^i$$

$$= \sum_{i=0}^{m} \left\{ \sum_{j=i}^{m} a_j \frac{j!}{i!(j-i)!} \lambda^{j-i} \right\} N^i$$

$$\sum_{i=0}^{m} \frac{1}{i!} \left\{ \sum_{j=i}^{m} \frac{j!}{(j-i)!} a_j (\lambda)^{j-i} \right\} N^i$$

or

$$p(J_k(\lambda)) = \sum_{i=0}^{m} \frac{1}{i!} p^i(\lambda) N^i$$

$$= \sum_{i=0}^{\mu} \frac{1}{i!} p^i(\lambda) N^i, \qquad \mu = min\{m, k-1\}$$

$$p(J_k) = p(J_k(\lambda)) = \begin{pmatrix} p(\lambda) & p'(\lambda) & \frac{1}{2}p''(\lambda) & \cdots & \frac{p^{(k-1)}(\lambda)}{(k-1)!} \\ 0 & p(\lambda) & p'(\lambda) & \ddots & \vdots \\ 0 & 0 & p(\lambda) & \ddots & \frac{1}{2}p''(\lambda) \\ \vdots & \vdots & \cdots & \ddots & p'(\lambda) \\ 0 & 0 & \cdots & \cdots & p(\lambda) \end{pmatrix} \tag{3.18}$$

in which all entries in the *ith* superdiagonal are $p^{(i)}(\lambda)/i!$, the normalized *ith* derivative. Notice that only derivatives up to order $k-1$ are required.

Definition 24. The values

$$f^j(\lambda_i), \quad j = 0 : n_i - 1, \quad i = 1 : s$$

are called "the values of the function f and its derivatives on the spectrum of A", and if they exist f is said to be defined on the spectrum of A.

Note that the minimal polynomial ψ takes the values zero on the spectrum of A.

Theorem 13. [7, p. 5, Theorem 1.3] For polynomials p and q and $A \in \mathbb{C}^{n\times n}$, $p(A) = q(A)$ if and only if p and q take the same values on the spectrum of A.

Proof. Suppose, we have two polynomials p and q such that $p(A) = q(A)$ and let $d(\lambda) = p(\lambda) - q(\lambda)$, then obviously $d(A) = p(A) - q(A) = 0$, so the values of d on spectrum of A are all 0, and hence $d(\lambda)$ is an annihilating polynomial for A. Thus (3.11) $d(\lambda)$ is divisible by the minimal polynomial $\psi(\lambda)$ of A given by (3.13) and there exist a polynomial $q(\lambda)$ such that

$$d(\lambda) = q(\lambda)\psi(\lambda).$$

Computing the values of $d(\lambda)$ on the spectrum of A, then

$$d^{(j)}(\lambda_i) = 0 \quad for\ i = 1, ...s,\ \ j = 0, ...n_i - 1$$

where s is the number of distinct eigenvalues.

$$p^{(j)}(\lambda_i) - q^{(j)}(\lambda_i) = 0$$

$$p^{(j)}(\lambda_i) = q^{(j)}(\lambda_i)$$

for $i = 1, ...s,\ j = 0, ...n_i - 1$,
So,

$$p(A) = q(A) \ \Rightarrow\ p^{(j)}(\lambda_i) = q^{(j)}(\lambda_i)$$

Thus, the two polynomials $p(A)$ and $q(A)$ have the same values on the spectrum of A provided $p(A) = q(A)$.
Conversely, if $p^{(j)}(\lambda_i) = q^{(j)}(\lambda_i)$ holds, then

$$d^{(j)}(\lambda_i) = 0, \quad i = 1, ..., s, \quad j = 0, ..., n_i - 1$$

Hence $d(\lambda)$ must be divisible by $\psi(\lambda)$ in (3.13) and it follows that

$$d(A) = 0$$

$$p(A) - q(A) = 0 \Longrightarrow p(A) = q(A)$$

□

3.2.1 Matrix Function via Hermite interpolation

Definition 25. Let f be defined on the spectrum of A. Then $f(A) = r(A)$, where r is the unique *Lagrange-Sylvester interpolating polynomial* of degree less than

$$\sum_{i=1}^{s} n_i = deg\psi$$

that satisfies the interpolation conditions

$$r^j(\lambda_i) = f^j(\lambda_i) \quad j = 0 : n_i - 1, \quad i = 1 : s$$

Note that the polynomial r depends on A through the values the function f takes on the spectrum of A.

Let us consider the following example which is useful clarification.

Example 19. *[7, p. 5] Consider* $f(t) = \sqrt{t}$ *and*

$$A = \begin{pmatrix} 2 & 2 \\ 1 & 3 \end{pmatrix}$$

The eigenvalues are $\lambda_1 = 1, \quad \lambda_2 = 4, \quad s = 2 \quad and \quad n_1 = n_2 = 1$ *so that the minimal polynomial* $\psi(t)$ *of* A *is* $(t-1)(t-4)$. *The Lagrange interpolatory polynomial is*

$$p(1) = f(1) = \sqrt{1} = 1$$
$$p(4) = f(4) = \sqrt{4} = 2$$
$$p(t) = \sum_{i=1}^{k} f(\lambda_i) \prod_{j=1, j\neq i}^{k} \frac{t - \lambda_j}{\lambda_i - \lambda_j}$$
$$p(t) = f(1)\frac{t-4}{1-4} + f(4)\frac{t-1}{4-1} = \frac{1}{3}(t+2)$$

Hence,

$$f(A) = p(A) = \frac{1}{3}(A + 2I) = \frac{1}{3}\begin{pmatrix} 4 & 2 \\ 1 & 5 \end{pmatrix}$$

It is easily checked that $f(A) = p(A) = \sqrt{A} \Rightarrow f(A)^2 = A$, *which is true.*

Let us consider another example, where the matrix has the same eigenvalues, but a different Jordan Canonical Form.

Example 20. *Consider* $f(t) = \sqrt{t}$ *and*

$$B = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 4 \end{pmatrix}.$$

The eigenvalues are $\lambda_1 = 1$, $n_1 = 2$ *and* $\lambda_2 = 4$ $n_2 = 1$. B *has Jordan Canonical Form.*

$$= \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 4 \end{pmatrix}$$

Let $\tilde{p}$ *denote the unique Lagrange-Syvester interpolating polynomial of degree*$\leqslant n_1 + n_2 = 3$ *Then, we have*

$$f(t) = \sqrt{t} \quad f'(t) = \frac{1}{2}(t)^{-1/2}$$

$$\tilde{p}(t) = f(t) = \sqrt{t}$$
$$\tilde{p}(1) = f(1) = \sqrt{1} = 1$$
$$\tilde{p'}(t) = f'(1) = \frac{1}{2}(1)^{-1/2} = \frac{1}{2}$$
$$\tilde{p}(4) = f(4) = \sqrt{4} = 2$$

Let

$$\tilde{p}(t) = a_0 + a_1 t + a_2 t^2.$$

To solve for the values a_0, a_1 *and* a_2 *we have,*

$$a_0 + a_1 + a_2 = 1$$

$$a_1 + 2a_2 = \frac{1}{2}$$
$$a_0 + 4a_1 + 16a_2 = 2.$$

After solving the system of equations, we have,

$$a_0 = \frac{8}{18}, a_1 = \frac{11}{18}, a_2 = \frac{-1}{18}$$

$$\tilde{p}(t) = \frac{8}{18} + \frac{11}{18}t - \frac{1}{18}t^2$$

Hence,

$$\tilde{p}(B) = \frac{8}{18} + \frac{11}{18}B - \frac{1}{18}B^2$$

$$\tilde{p}(B) = f(B) = \frac{1}{18}(8I + 11B - B^2)$$
$$= \frac{1}{18}\begin{pmatrix} 18 & 9 & 5 \\ 0 & 18 & 6 \\ 0 & 0 & 36 \end{pmatrix}$$

Since $\tilde{p}$ satisfies the same conditions as p, so with A as in Example 19. We have

$$\tilde{p}(A) = A^{1/2}.$$

But p satisfies two conditions not all three conditions that $\tilde{p}$ satisfies. So we can not expect that $p(B) = \tilde{p}(B)$, and in fact, we have

$$p(B) = \frac{1}{3}(B + 2I) = \begin{pmatrix} 1 & 1/3 & 1/3 \\ 0 & 1 & 1/3 \\ 0 & 0 & 2 \end{pmatrix}$$

$$p(B)^2 = \begin{pmatrix} 1 & 2/3 & 10/9 \\ 0 & 1 & 1 \\ 0 & 0 & 4 \end{pmatrix} \neq B$$

so

$$p(B) \neq B^{1/2} = \tilde{p}(B)$$

Example 21. *Consider $f(t) = e^{2t}$ and*

$$A = \begin{pmatrix} 6 & -1 \\ 3 & 2 \end{pmatrix}$$

The eigenvalues are $\lambda_1 = 3, \quad \lambda_2 = 5, \quad s = 2 \quad$ and $\quad n_1 = n_2 = 1$ so that the minimal polynomial $\psi(t)$ of A is $(t-3)(t-5)$. The Lagrange interpolating polynomial:

$$p(t) = \sum_{i=1}^{k} f(\lambda_i) \prod_{j=1, j\neq i}^{k} \frac{t - \lambda_j}{\lambda_i - \lambda_j}$$
$$p(t) = f(3)\frac{t-5}{3-5} + f(5)\frac{t-3}{5-3}$$
$$p(t) = e^6(\frac{t-5}{-2}) + e^{10}(\frac{t-3}{2})$$

Hence,

$$p(A) = f(A) = -\frac{1}{2}e^6(A - 5I) + \frac{1}{2}e^{10}(A - 3I)$$
$$e^{2A} = \frac{1}{2}\begin{pmatrix} 3e^{10} - e^6 & -e^{10} + e^6 \\ 3e^{10} - 3e^6 & -e^{10} + 3e^6 \end{pmatrix}$$

We now mention two important properties of matrix functions that are discussed by [**?**, p. 310, Theorem 1, Theorem 2]

Lemma 6. *[10, p. 310, Theorem 2] If $A, B, X \in \mathbb{C}^{n\times n}$, where $B = XAX^{-1}$, and $f(\lambda)$ is defined on the spectrum of A, then*

$$f(B) = Xf(A)X^{-1}. \tag{3.19}$$

Proof. Since A and B are similar, they have the same Jordan canonical form. So the polynomial p that interpolate f on the spectrum of A also interpolate f on the spectrum of B. Thus $f(A) = p(A)$ and $f(B) = p(B)$.

Since $B = XAX^{-1}$, we have

$$B^k = (XAX^{-1})^k = XA^kX^{-1}.$$

Let $p(\lambda) = \sum_{i=0}^{n} \alpha_i \lambda^i$. Then

$$f(B) = p(B) = \sum_{i=0}^{n} \alpha_i B^i = \sum_{i=0}^{n} \alpha_i X A^i X^{-1}$$

$$= X(\sum_{i=0}^{n} \alpha_i A^i)X^{-1} = Xp(A)X^{-1}$$

$$f(B) = Xf(A)X^{-1}$$

□

Lemma 7. *[10, p. 310, Theorem 1] If $A \in \mathbb{C}^{n\times n}$ is a block diagonal matrix*

$$A = diag(A_1, A_2, ..., A_s)$$

where $A_1, A_2, ...A_s$ are square matrices, then

$$f(A) = diag(f(A_1), f(A_2), ..., f(A_s)). \tag{3.20}$$

Proof. If $r(\lambda)$ is the interpolating polynomial of minimal degree that interpolates to $f(\lambda)$ on the spectrum of A then

$$f(A) = r(A) = diag(r(A_1), r(A_2), ..., r(A_s)) \tag{3.21}$$

Since, on the other hand, the minimal polynomial $\psi(\lambda)$ of A is an annihilating polynomial for each of the matrices $A_1, A_2, ..., A_s$. Therefore it follows from the equation

$$f(\Lambda_A) = r(\Lambda_A)$$

that

$$f(\Lambda_{A_1}) = diag(r(\Lambda_{A_1}), r(\Lambda_{A_2}), ..., r(\Lambda_{A_s}))$$

Therefore

$$f(A_1) = r(A_1), f(A_2) = r(A_2), ..., f(A_r) = r(A_s),$$

and eq(3.18) can be written as follows:

$$f(A) = diag(f(A_1), f(A_2), ..., f(A_s)$$

□

3.3 Matrix function via Cauchy Integral Formula

The Cauchy integral formula is an elegant result from complex analysis stating that the value of a function can be computed by an integral. Given a function $f(z)$ and a value $z = a$ we can compute $f(a)$ by

$$f(a) = \frac{1}{2\pi i} \int_\Gamma \frac{f(z)}{z - a} dz \tag{3.22}$$

where Γ is a contour in $\mathbb{C}$ such that Γ encloses a and $f(z)$ is analytic on and inside Γ. Generalizing this formula to the matrix case. We have

Definition 26. Let $\Omega \subset \mathbb{C}$ be a domain, $f : \Omega \longrightarrow \mathbb{C}$ analytic function. Let diagonalisable $A \in \mathbb{R}^{n\times n}$ such that all the eigenvalues of A lie in Ω. Denote the boundary of Ω by Γ. Then we define $f(A) \in \mathbb{R}^{n\times n}$ as a contour integral

$$f(A) = \frac{1}{2\pi i} \int_\Gamma f(z)(zI - A)^{-1} dz, \tag{3.23}$$

where $(zI - A)^{-1}$ is the resolvent of A at z and Γ is a closed contour lying in the region of analyticity of f and winding once around the spectrum $\Lambda(A)$ in the contour-clockwise direction.(Here $i = \sqrt{-1}$ and the contour integral of the matrix $f(z)(zI - A)^{-1}$ is the matrix of the contour integrals of each entry of $f(z)(zI - A)^{-1}$.)

Let us illustrate the following example.

Example 22. *Using the Cauchy integral formula to find $f(A)$ for*

$$A = \begin{pmatrix} -1 & 1 \\ 3 & 1 \end{pmatrix} \quad \textit{and} \quad f(z) = z^2.$$

Solution. We calculate

$$f(z)(zI - A)^{-1} = z^2 \begin{pmatrix} z+1 & -1 \\ -3 & z-1 \end{pmatrix}^{-1}$$

$$= \frac{z^2}{z^2-4} \begin{pmatrix} z-1 & 1 \\ 3 & z+1 \end{pmatrix}$$

The eigenvalues of A are $\lambda_1 = -2$ and $\lambda_1 = 2$. Since $f(z) = z^2$ is analytic on $\mathbb{C}$,
we consider

$$\Omega = \{z \in \mathbb{C} : |z| < 2 + \epsilon\}$$

for some $\epsilon > 0$, so that $\lambda_1, \lambda_2 \in \Omega$. Hence Ω is the circle with center the origin and radius $2 + \epsilon$. we have to calculate the contour integral

$$f(A) = \frac{1}{2\pi i} \int_\Gamma \frac{z^2}{z^2-4} \begin{pmatrix} z-1 & 1 \\ 3 & z+1 \end{pmatrix} dz$$

$$f(A) = \begin{pmatrix} \frac{1}{2\pi i}\int_\Gamma \frac{z^2(z-1)}{z^2-4}dz & \frac{1}{2\pi i}\int_\Gamma \frac{z^2}{z^2-4}dz \\ \frac{1}{2\pi i}\int_\Gamma \frac{3z^2}{z^2-4}dz & \frac{1}{2\pi i}\int_\Gamma \frac{z^2(z+1)}{z^2-4}dz \end{pmatrix}$$

for $z = (2+\epsilon)e^{i\theta}$. This implies $dz = (2+\epsilon)ie^{i\theta}d\theta$ and , calculating the contour integrals and letting $\epsilon \to 0$, we get

$$f(A) = \begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix}$$

The calculation of the integrals in $f(A)$ is hard to evaluate especially for $n > 2$.

Theorem 14. [9, p. 427, Theorem 6.2.28] Let $A \in \mathbb{R}^{n\times n}$ diagonalisable matrix and f analytic function on a domain that contains the eigenvalues of A. Then

$$f(A) = Xf(\Lambda)X^{-1}$$

where $A = X\Lambda X^{-1}$ is the eigenvalue decomposition of A, and f is defined by the Cauchy integral formula.

Proof. we have

$$(zI-A)^{-1} = (zI-X\Lambda X^{-1})^{-1} = (X(zI-\Lambda)X^{-1})^{-1} = X(zI-\Lambda)^{-1}X^{-1}$$

Hence,

$$f(A) = \frac{1}{2\pi i}\int_\Gamma f(z)X(zI-A)^{-1}dz$$

$$= \frac{1}{2\pi i}\int_\Gamma f(z)X(zI-\Lambda)^{-1}X^{-1}dz$$

$$= X(\frac{1}{2\pi i}\int_\Gamma f(z)(zI-\Lambda)^{-1}dz)X^{-1}$$

$$f(A) = Xf(\Lambda)X^{-1}$$

Such results are often called *spectral theorems.* □

We, therefore, conclude that:

1. the theorem above says that $f(A)$ is similar to the matrix $f(\Lambda)$;
2. if we calculate $f(\Lambda)$, the spectral theorem gives us a way of calculating $f(A)$

But what is $f(\Lambda)$ for Λ diagonal matrix.?

3.4 Functions of diagonal matrices

Definition 27. We denote a diagonal matrix $D \in \mathbb{M}_n$ with entries $d_i \quad i = 1, 2..., n$ by $diag(d_1, d_2, ..., d_n)$

Lemma 8. *[9, p. 407,(6.2.1)] Let $\Lambda \in \mathbb{R}^{n\times n}$ diagonal matrix with $\Lambda = diag(\lambda_1, \lambda_2, ..., \lambda_n)$ and f analytic function on a domain containing $\lambda_i, i = 1, 2, ..., n$. Then we have that*

$$f(\Lambda) = diag(f(\lambda_1), f(\lambda_2), ..., f(\lambda_n))$$

Proof. Since Λ and zI are diagonal, so is $zI - A$; hence,

$$(zI-\Lambda)^{-1} = diag(\frac{1}{z-\lambda_1}, ..., \frac{1}{z-\lambda_n})$$

Using the definition, we have

$$f(\Lambda) = \frac{1}{2\pi i}\int_\Gamma f(z)(zI-\Lambda)^{-1}dz = \frac{1}{2\pi i}\int_\Gamma diag(\frac{f(z)}{z-\lambda_1},...,\frac{f(z)}{z-\lambda_n})dz$$
$$= diag(\frac{1}{2\pi i}\int_\Gamma \frac{f(z)}{z-\lambda_1}dz,...,\frac{1}{2\pi i}\int_\Gamma \frac{f(z)}{z-\lambda_n})dz$$
$$= diag(f(\lambda_1), f(\lambda_2),..., f(\lambda_n))$$

from Cauchy's integral formula. □

In view of the above lemma, we conclude the following:

1. since $f(\Lambda)$ is diagonal, the eigenvalue decomposition of $f(A)$ reads $f(A) = Xf(\Lambda)X^{-1}$;
2. it is easy to calculate $f(\Lambda)$; just apply f to the eigenvalues of A.

Let us consider the following example.

Example 23. *Find $f(A)$, using the Diagonalisation for matrix A.*

$$A = \begin{pmatrix} 2 & 2 \\ 1 & 3 \end{pmatrix}$$

The diagonalisation form of A is $A = X\Lambda X^{-1}$.
The eigenvalues are $\lambda_1 = 1$, $\lambda_2 = 4$ where

$$X = \begin{pmatrix} -2 & 1 \\ 1 & 1 \end{pmatrix}, \quad X^{-1} = \frac{-1}{3}\begin{pmatrix} 1 & -1 \\ -1 & -2 \end{pmatrix}, \Lambda = \begin{pmatrix} 1 & 0 \\ 0 & 4 \end{pmatrix}$$

Then

$$f(A) = \frac{1}{2\pi i}\int_\Gamma f(z)(zI-A)^{-1}dz$$
$$f(A) = \frac{1}{2\pi i}\int_\Gamma f(z)(zI-X\Lambda X^{-1})^{-1}dz$$
$$f(A) = \frac{1}{2\pi i}\int_\Gamma f(z)X(zI-\Lambda)^{-1}X^{-1}dz$$
$$f(A) = X[\frac{1}{2\pi i}\int_\Gamma \begin{pmatrix} z-\lambda_1 & 0 \\ 0 & z-\lambda_2 \end{pmatrix}^{-1} f(z)d(z)]X^{-1}$$
$$f(A) = X\begin{pmatrix} f(\lambda_1) & 0 \\ 0 & f(\lambda_2) \end{pmatrix}X^{-1}$$
$$f(A) = Xf(\Lambda)X^{-1}$$

But not all matrices are diagonalisable, There are many other ways of calculating $f(A)$ without using the eigenvalue decomposition method.

3.5 Power Series Expansions

Frequently there is a basic requirement of computing the power of a matrix. We have to compute a sequence of successive powers A^2,A^3,..., in the obvious way i.e by repeated multiplication by A.

Theorem 15. [5, p. 565, Theorem 11.2.3] If f has a power series expansion

$$f(z) = \sum_{k=0}^{\infty} c_k z^k$$

on an open disc containing the eigenvalues of A, then

$$f(A) = \sum_{k=0}^{\infty} c_k A^k$$

Proof. First we consider the case when A is diagonalizable.
Suppose $X^{-1}AX = \Lambda = diag(\lambda_1, ..., \lambda_n)$ using theorem 14, we have

$$f(A) = Xdiag(f(\lambda_1), ..., f(\lambda_n))X^{-1}$$

$$= Xdiag(\sum_{k=0}^{\infty} c_k\lambda_1^k, ..., \sum_{k=0}^{\infty} c_k\lambda_n^k)X^{-1}$$

$$= X(\sum_{k=0}^{\infty} c_k\Lambda^k)X^{-1} = \sum_{k=0}^{\infty} c_k(X\Lambda X^{-1})^k$$

$$f(A) = \sum_{k=0}^{\infty} c_k A^k$$

Next we consider the case where A is not diagonalizable. In this case we can find a sequence of matrices $A_j \to A$, where each A_j has distinct eigenvalues. The A_j therefore are diagonalizable.
Hence

$$\frac{1}{2\pi i}\int_\Gamma z^k(zI - A)^{-1}dz = \lim_{j\to\infty} \frac{1}{2\pi i}\int_\Gamma z^k(zI - A_j)^{-1}dz$$

$$= \lim_{j\to\infty} A_j^k = A^k$$

□

Example 24. *Find $f(A)$ when*

1. $f(z) = e^z$. *We have*

$$e^z = \sum_{k=0}^{\infty} \frac{1}{k!} z^k$$

Hence

$$e^A = \sum_{k=0}^{\infty} \frac{1}{k!} A^k$$

2. $f(z) = \sin(z)$. *We have*

$$\sin(z) = \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k+1)!} z^{2k+1}$$

Hence

$$\sin(A) = \sum_{k=0}^{\infty} \frac{(-1)^k}{(2k+1)!} A^{2k+1}$$

A way to approximate $f(A)$ is truncate the power series expansions, i.e.,

$$f(A) \approx \sum_{k=0}^{m} c_k A^k.$$

This is possible to compute in practice, the bigger is m is the better the approximation. Since f is analytic, the coefficients c_k in the power series expansion get smaller and smaller in absolute value.

3.6 The Relationship of the Definitions of Matrix Function

If $A \in \mathbb{C}^{n\times n}$ and f is analytic function on a domain that contains the spectrum of A, there are many ways to define $f(A)$.

R.F.Rinehart [17] shows that all of our three definitions of a matrix function are equivalent.

Theorem 16. Let $A \in \mathbb{M}_n$. Let f be an analytic function defined on a domain containing the spectrum of A. Let

1. $f_J(A)$, denote the value of $f(A)$ computed the Jordan canonical form definition:

2. $f_p(A)$, denote the value of $f(A)$ computed the interpolating polynomial definition:
3. $f_c(A)$, denote the value of $f(A)$ computed the Cauchy integral definition:

Then

$$f_J(A) = f_p(A) = f_c(A) \tag{3.24}$$

To prove this theorem we will need to prove some preliminary results. First we consider the value of f on a diagonal matrix.

Lemma 9. *[9, p. 385] Let $A \in \mathbb{C}^{n\times n} = diag(\lambda_1, \lambda_2, ..., \lambda_n)$ and f be an analytic function defined on a domain containing the spectrum of A. Then*

$$f_c(A) = f_p(A) = f_J(A) = diag(f(\lambda_1), ..., f(\lambda_n)) \tag{3.25}$$

Proof. If A is diagonal then

$$A = \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix}$$

block diagonal matrix each block is 1×1, then

$$f_J(A) = \begin{pmatrix} f(J_1) & & \\ & \ddots & \\ & & f(J_n) \end{pmatrix}$$

but

$$J_1 = [\lambda_1] \quad 1 \times 1,$$

so then by definition

$$f(J_1) = [f(\lambda_1)],$$

and hence

$$f_J(A) = \begin{pmatrix} f(\lambda_1) & & \\ & \ddots & \\ & & f(\lambda_n) \end{pmatrix}. \tag{3.26}$$

Now consider the polynomial definition.

$$p_{A,f}(\lambda_i) = f(\lambda_i), \ \ for\ i = 1, 2, ..., n$$

$$p_{A,f}(t) = \sum_{i=0}^{n-1} \alpha_i t^i$$

$$f_p(A) = p_{A,f}(A) = \sum_{i=0}^{n-1} \alpha_i A^i$$

$$f_p(A) = \sum_{i=0}^{n-1} \alpha_i \begin{pmatrix} \lambda_1^i & & \\ & \ddots & \\ & & \lambda_n^i \end{pmatrix}$$

$$= \begin{pmatrix} \sum_{i=0}^{n-1} \alpha_i \lambda_1^i & & \\ & \ddots & \\ & & \sum_{i=0}^{n-1} \alpha_i \lambda_n^i \end{pmatrix}$$

$$= \begin{pmatrix} p_{A,f}(\lambda_1) & & \\ & \ddots & \\ & & p_{A,f}(\lambda_n) \end{pmatrix}$$

$$f_p(A) = \begin{pmatrix} f(\lambda_1) & & \\ & \ddots & \\ & & f(\lambda_n) \end{pmatrix} \tag{3.27}$$

Finally, consider the Cauchy Integral formula

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

$$f_c(A) = \frac{1}{2\pi i} \oint_\Gamma f(z) \begin{pmatrix} z - \lambda_1 & & \\ & \ddots & \\ & & z - \lambda_n \end{pmatrix}^{-1}$$

$$f_c(A) = \begin{pmatrix} \frac{1}{2\pi i} \oint_\Gamma f(z)(z - \lambda_1)^{-1} & & \\ & \ddots & \\ & & \frac{1}{2\pi i} \oint_\Gamma f(z)(z - \lambda_n)^{-1} \end{pmatrix}$$

$$f_c(A) = \begin{pmatrix} f(\lambda_1) & & \\ & \ddots & \\ & & f(\lambda_n) \end{pmatrix} \tag{3.28}$$

Hence from (3.26), (3.27), and (3.28), we have

$$f_J(A) = f_p(A) = f_c = diag(f(\lambda_1), ..., f(\lambda_n)),$$

as required. □

Now we show that the three definitions interact with similarities in the same way.

Lemma 10. *[9, p. 412, Theorem 6.2.9(c)] Let $A \in \mathbb{M}_n$ and f be an analytic function defined on a domain containing the spectrum of A. Then*

$$f_J(XAX^{-1}) = Xf_J(A)X^{-1} \tag{3.29}$$

$$f_p(XAX^{-1}) = Xf_p(A)X^{-1} \tag{3.30}$$

$$f_c(XAX^{-1}) = Xf_c(A)X^{-1} \tag{3.31}$$

for any nonsingular matrix $X \in \mathbb{M}_n$.

Proof. Now, assume that $B = XAX^{-1}$, where $X \in \mathbb{C}^{n\times n}$ is nonsingular, $B \in \mathbb{C}^{n\times n}$

1. If $A = SJS^{-1}$ is the Jordan canonical form of A, then

   $$B = XAX^{-1} = XSJS^{-1}X^{-1} = (XS)J(XS)^{-1},$$

   so by definition

   $$f_J(A) = f_J(SJS^{-1}) = Sf_J(J)S^{-1}$$

   and

   $$\begin{aligned} f_J(B) &= (XS)f_J(J)(XS)^{-1} \\ f_J(B) &= X(Sf_J(J)S^{-1})X^{-1} \\ f_J(B) &= Xf_J(A)X^{-1} \\ f_J(XAX^{-1}) &= Xf_J(A)X^{-1} \end{aligned} \tag{3.32}$$

   as required.

2. Now consider the power series definition Let

   $$f(t) = \sum_{k=0}^{\infty} \alpha_k t^k$$

   $$f_p(A) = \sum_{k=0}^{\infty} \alpha_k A^k$$

   Since

   $$B = XAX^{-1}$$

Then

$$f_p(B) = \sum_{k=0}^{\infty} \alpha_k B^k$$

$$f_p(B) = \sum_{k=0}^{\infty} \alpha_k (XAX^{-1})^k = \sum_{k=0}^{\infty} \alpha_k (XA^kX^{-1})$$

$$f_p(B) = X(\sum_{k=0}^{\infty} \alpha_k A^k)X^{-1} = Xf(A)X^{-1}$$

$$f_p(XAX^{-1}) = Xf_p(A)X^{-1} \tag{3.33}$$

as required.

3. Finally, consider the Cauchy Integral definition. Let

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

Now again

$$B = XAX^{-1}$$

Therefore

$$f(B) = \frac{1}{2\pi i} \int_\Gamma f(z)(zI - B)^{-1} dz$$

$$f(B) = \frac{1}{2\pi i} \int_\Gamma f(z)(zI - XAX^{-1})^{-1} dz$$

$$f(B) = \frac{1}{2\pi i} \int_\Gamma f(z)X(zI - A)^{-1}X^{-1} dz$$

$$f(B) = X(\frac{1}{2\pi i} \int_\Gamma f(z)(zI - A)^{-1} dz)X^{-1}$$

so

$$f_c(XAX^{-1}) = Xf_c(A)X^{-1} \tag{3.34}$$

□

We can generalize the idea for evaluating matrix functions using the relation $f(XAX^{-1}) = Xf(A)X^{-1}$.

Now we show $f_J(A) = f_p(A) = f_c(A)$ when A is diagonalizable.

Lemma 11. *[9, p. 407,] If $A \in \mathbb{M}_n$ is diagonalizable, then*

$$f_J(A) = f_p(A) = f_c(A)$$

holds.

Proof. Since $A \in \mathbb{M}_n$ is diagonalizable and

$$A = S\Lambda S^{-1}$$

with $\Lambda = diag(\lambda_1, ..., \lambda_n)$, then

$$f_J(A) = f_J(S\Lambda S^{-1})$$

$$= S f_J(\Lambda) S^{-1}$$

$$f_J(A) = S \begin{pmatrix} f(\lambda_1) & & \\ & \ddots & \\ & & f(\lambda_n) \end{pmatrix} S^{-1} \tag{3.35}$$

Since $f(\lambda)$ and $p_{A,f}(\lambda)$ assume the same values on the spectrum of A, they must also have the same values on the spectrum of A. Hence $p_{A,f}(\lambda_i) = f(\lambda_i), \quad i = 1, 2, ..., n$. Therefore

$$f(A) = p_{A,f}(A) = \sum_{i=0}^{n-1} \alpha_i A^i$$

$$f_p(A) = p(A) = \sum_{i=0}^{n-1} \alpha_i (S\Lambda S^{-1})^i$$

$$= S(\sum_{i=0}^{n-1} \alpha_i \Lambda^i) S^{-1},$$

$$f_p(A) = S \begin{pmatrix} \sum_{i=0}^{n-1} \lambda_1^i & & \\ & \ddots & \\ & & \sum_{i=0}^{n-1} \lambda_{n-1}^i \end{pmatrix} S^{-1}$$

$$= S \begin{pmatrix} p(\lambda_1) & & \\ & \ddots & \\ & & p(\lambda_n) \end{pmatrix} S^{-1}$$

$$f_p(A) = S \begin{pmatrix} f(\lambda_1) & & \\ & \ddots & \\ & & f(\lambda_n) \end{pmatrix} S^{-1}. \tag{3.36}$$

Now, consider

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

Since $A = S\Lambda S^{-1}$ then

$$f_c(A) = f(A) = \frac{1}{2\pi i}\oint_\Gamma f(z)(zI - S\Lambda S^{-1})^{-1}dz$$

$$f_c(A) = S(\frac{1}{2\pi i}\oint_\Gamma f(z)(zI - \Lambda)^{-1}dz)S^{-1}$$

$$f_c(A) = Sf(\Lambda)S^{-1} \tag{3.37}$$

Hence from equation(3.38), equation(3.39) and equation(3.40) we have

$$f_J(A) = f_p(A) = f_c(A)$$

□

Lemma 12. *[5, p. 316, Corollary 7.1.8] If $A \in \mathbb{M}_n$ has distinct eigenvalues then A is diagonalisable.*

Proof. If A has n linearly independent eigenvectors $x_1, x_2, ..., x_n$, form a nonsingular matrix X with them as columns. Then

$$Ax_i = \lambda_i x_i \ \ for\ i = 1, 2, ..., n$$

and λ_i are distinct eigenvalues.
Then

$$AX = [Ax_1|Ax_2|....|Ax_n] = [\lambda_1 x_1|\lambda_2 x_2...|\lambda_n x_n]$$

$$AX = [x_1|x_2|...x_n]\begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix} = X\Lambda \ \Rightarrow\ A = X\Lambda X^{-1}.$$

□

Note that not every matrix is diagonalizable. e.g.; Consider

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$$

has eigenvalues $\{0, 0\}$. Then,

$$X\Lambda X^{-1} = X\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}X^{-1} = 0.$$

Lemma 13. *[9, p. 408] Let $A \in \mathbb{M}_n$, $\epsilon > 0$. Then $\exists$ A_ϵ such that*

$$\|A_\epsilon - A\| \leqslant \epsilon$$

and A_ϵ has distinct eigenvalues and hence is diagonalizable.

Proof. Let $A = UTU^*$ be the Schur form of A. If T has distinct eigenvalues then A is diagonalizable. Take $A_\epsilon = A$. If not let $\delta = min\{|\lambda_i - \lambda_j|, \lambda_i \neq \lambda_j\}$
Let $\eta = min\{\delta, \epsilon\}/n > 0$ Consider the upper triangular matrix T with the diagonal entries $t_{11}, ..., t_{nn}$. Let $T_\epsilon = T + diag(0, \eta, ..., (n-1)\eta)$ Then T_ϵ has distinct eigenvalues.
Let

$$A_\epsilon = UT_\epsilon U^*.$$

$$\|A - A_\epsilon\| = \|UTU^* - UT_\epsilon U^*\| = \|T - T_\epsilon\|$$

$$\|A - A_\epsilon\| = (n-1)\eta < \epsilon.$$

□

We will establish continuity properties of f_p and f_c, so that we can use a continuity argument to show $f_p(A) = f_c(A)$ in the non-diagonalizable case.

Lemma 14. *[9, p. 396, Theorem 6.1.28][9, p. 427, Theorem 6.2.28] Let $A \in \mathbb{M}_n$ and f be analytic function on a domain containing the spectrum of A. Then f_p and f_c are continuous at A.*

Proof. When A is not diagonalizable. We will use a continuity argument.

1. $f_p(A)$ is a continuous function of A.

Let $t_{\epsilon,i}$, $i = 1, ...n$ be the spectrum of A_ϵ. Let r_{A_ϵ} be the polynomial of $deg \leqslant n$ that interpolates f at the spectrum of A_ϵ then

$$r_{A_\epsilon}(t) = \sum_{j=1}^{n} f[t_{\epsilon,1}, ..., t_{\epsilon,j}] \prod_{i=1}^{j-1} (t - t_{\epsilon,i})$$

Note that eigenvalues of a matrix are continuous functions of the matrix. As $\epsilon \to 0, t_{\epsilon,i} \to t_i$, also $f[t_1, ..., t_j]$ is a continuous function of

$t_1, ..., t_j$ provided f is $j-1$ terms continuously differentiable. Thus the coefficients of $r_{A_\epsilon}(t)$ are continuous functions of A_ϵ. So

$$r_{A_\epsilon}(t) = \sum_{j=0}^{n} a_{\epsilon,j} t^j$$

where $a_{\epsilon,j} \to a_{o,j}$ as $\epsilon \to 0$ and

$$r_A(t) = \sum_{j=0}^{n} a_{0,j} t^j$$

Thus

$$r_{A_\epsilon}(A_\epsilon) = \sum_{j=0}^{n} a_{\epsilon,j} A_\epsilon^j$$

since

$$a_{\epsilon,j} \to a_{0,j}$$
$$A_\epsilon^j \to A^j$$
$$r_{A_\epsilon}(A_\epsilon) \to r_A(A)$$

as required. Thus $f_p(A)$ is continuous.

1. $f_c(A)$ is a continuous function of A.

To show that $f_c(A)$ is continuous. Consider

$$f_c(A) = \frac{1}{2\pi i} \oint_\Gamma f(z)(zI - A)^{-1} \tag{3.38}$$

since $(zI - A)^{-1}$ is continuous for z not on eigenvalues of A, we might expect that equation (3.28) is continuous in A since the boundary of Γ is disjoint from the spectrum of A and is of finite length. The details are in [9, Theorem 6.2.28, p. 427]. □

With these results we are in a position to prove Theorem 16.

Proof. First consider the case that A is diagonal. Then by Lemma 9, we have that,

$$f_J(A) = f_p(A) = f_c(A)$$

Now consider the case that A is diagonalizable, i.e., $A = X\Lambda X^{-1}$. Then by Lemma 10, we have the

$$f_J(A) = f_J(X\Lambda X^{-1}) = X f_J(\Lambda) X^{-1}$$

$$f_p(A) = f_p(X\Lambda X^{-1}) = X f_p(\Lambda) X^{-1}$$

$$f_c(A) = f_c(X\Lambda X^{-1}) = X f_c(\Lambda) X^{-1}$$

Since, these equality yields the result,

$$f_J(A) = f_p(A) = f_c(A)$$

as desired.

Finally consider the case where $A \in \mathbb{M}_n$ is not necessarily diagonalisable. We will show.

1. $f_J(A) = f_p(A)$
2. $f_p(A) = f_c(A)$

Let

$$A = \begin{pmatrix} J_1 & & \\ & \ddots & \\ & & J_k \end{pmatrix} \in \mathbb{M}_n$$

then

$$f_J(A) = \begin{pmatrix} f(J_1) & & \\ & \ddots & \\ & & f(J_k) \end{pmatrix}$$

and

$$f_p(A) = \begin{pmatrix} p(J_1) & & \\ & \ddots & \\ & & p(J_k) \end{pmatrix}$$

We must show for a Jordan block

$$f(J_1) = p(J_1)$$

where p is the polynomial that interpolates f on the spectrum of $J_1 \oplus J_2 \oplus ... \oplus J_k$.

Note that from [9, Theorem 6.1.26, p. 395] we have: Let $t_1, t_2, ..., t_n \in \mathbb{C}^n$ and let the polynomial $r(t)$ be defined by the Newton formula

$$r(t) = f[t_1]+f[t_1,t_2](t-t_1)+f[t_1,t_2,t_3](t-t_1)(t-t_2)+...+f[t_1,...,t_n]\prod_{k=1}^{n-1}(t-t_k) \tag{3.39}$$

interpolate f at $t_1, t_2, ..., t_n$ and moreover $t_i = \lambda \; for \; i = 1, 2, ..., k$ we have

$$f[\lambda, \lambda, ..., \lambda] = \frac{1}{(k-1)!} f^{(k-1)}(\lambda) \tag{3.40}$$

Let

$$J_1 = \begin{pmatrix} \lambda_1 & 1 & & \\ & \lambda_1 & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_1 \end{pmatrix} = \lambda_1 I + N, \quad k_1 \times k_1$$

Since $\lambda_i = \lambda_1$ for $i > 1$. Now we will evaluate $p(J_1)$. Let $t_1, t_2, ..., t_n$ be the eigenvalues of $J_1, J_2, ..., J_k$.
We know $t_1 = t_2 = ... = t_{k_1} = \lambda_1$ by the formula (3.18)

$$p(t) = \sum_{j=1}^{n} f[t_1, ..., t_j] \prod_{i=1}^{j-1}(t - t_i)$$

so

$$p(J_1) = \sum_{j=1}^{n} f[t_1, ..., t_j] \prod_{i=1}^{j-1}(J - t_i I) \tag{3.41}$$

For $j = 1, ..., k_1 + 1$,

$$\prod_{i=1}^{j-1}(J_1 - t_i I) = \prod_{i=1}^{j-1}(\lambda_1 I + N - \lambda_1 I) = N^{j-1}$$

Since $N^{j-1} = 0 \quad for \; j = k_1 + 1$, so it must be zero for all $j \geqslant k_1 + 1$ so equation (3.41) is in fact

$$p(J_1) = \sum_{j=1}^{k_1} f[t_1, ..., t_j] N^{j-1}$$

$$p(J_1) = \sum_{j=1}^{k_1} \frac{f^{j-1}(\lambda_1)}{(j-1)!} N^{j-1}$$

$$p(J_1) = f(\lambda_1) + f'(\lambda_1)N + \frac{f''(\lambda_1)}{2!}N^2 + ... + \frac{f^{(k_1-1)}}{(k_1-1)!}N^{k_1-1}$$

$$p(J_1) = \begin{pmatrix} f(\lambda_1) & f'(\lambda_1) & \frac{f''(\lambda_1)}{2!} & \cdots & \frac{f^{(m_{k_1}-1)}(\lambda_1)}{(m_{k_1}-1)!} \\ 0 & f(\lambda_1) & f'(\lambda_1) & \ddots & \vdots \\ \vdots & 0 & f(\lambda_1) & \ddots & \frac{f''(\lambda_1)}{2!} \\ \vdots & \vdots & \vdots & \ddots & f'(\lambda_1) \\ 0 & 0 & 0 & & f(\lambda_1) \end{pmatrix}$$

Hence

$$p(J_1) = f_J(J_1)$$

So we have shown

$$f_J(A) = f_p(A)$$

We know (see Lemma 10) for any $A \in \mathbb{M}_n$, $\exists\ A_\epsilon$ such that $\|A - A_\epsilon\| \leqslant \epsilon$ and A_ϵ is diagonalizable.

Note that for $\epsilon > 0$, $f_p(A_\epsilon) = f_c(A_\epsilon)$ since A_ϵ is diagonalizable. Then

$$f_p(A) = \lim_{\epsilon \downarrow 0} f_p(A_\epsilon) \tag{3.42}$$

$$= \lim_{\epsilon \downarrow 0} f_c(A_\epsilon) \tag{3.43}$$

$$= f_c(A_\epsilon) \tag{3.44}$$

The equality in (3.45) is because f_p is continuous, the equality in (3.46) is because A_ϵ is diagonalizable and the equality in (3.47) is because f_c is continuous. Hence, we have proved that when A is not diagonalizable then

$$f_p(A) = f_c(A)$$

Hence, the desired result is

$$f_J(A) = f_p(A) = f_c(A)$$

This concludes the proof of the theorem. □

3.7 A Schur-Parlett Algorithm for Computation Matrix Functions

Many methods are based on the property $f(XAX^{-1}) = Xf(A)X^{-1}$. If X can be found such that

$$A = XBX^{-1}$$

then $f(A) = Xf(B)X^{-1}$. When for example, A is diagonalisable and $B = diag(\lambda_i)$, a simple calculation yields

$$f(A) = Xdiag(f(\lambda_i))X^{-1}. \tag{3.45}$$

Unfortunately, this approach is reliable only if X is well conditioned, that is, if the condition number $\kappa(X) = \|X\|\|X^{-1}\|$ is not too large. To overcome such complications the use of ill conditioned similarity transformations must be avoided. Ideally if X will be unitary then in the 2-norm $\kappa_2(X) = 1$. If for Hermitian A or more generally normal A, we have the spectral decomposition $A = QDQ^*$, where Q is unitary and D diagonal. If we compute this decomposition then $f(A) = Qf(D)Q^*$ gives an excellent way of computing $f(A)$.

Another way to do this is to take $A = XBX^{-1}$ to be the Schur decomposition. The Schur decompsoition of $A \in \mathbb{C}^{n\times n}$ can be written as

$$A = UTU^*, \tag{3.46}$$

where U is unitary and $T = (t_{ij})$ is upper triangular matrix ($t_{ij} = 0, \quad i > j$) and $\lambda_i = t_{ii} \quad i = 1, 2, ..., n$ appears on the diagonal of T. Since $U^* = U^{-1}$ hence $\kappa(U) = \|U\|\|U^{-1}\| = 1$ then

$$f(A) = Uf(T)U^* \tag{3.47}$$

In this way the computation of $f(A)$ is reduced to the computations of $F = f(T)$ for an upper triangular matrix T.

Let $T = (t_{ij})$ be an upper triangular matrix with $\lambda_i = t_{ii}$ and that $f(T)$ is defined by equation (3.47), that $f_{ii} = f(\lambda_i)$ for $1 \leqslant i \leqslant n$ and $f_{ij} = 0$ for $1 \leqslant j < i \leqslant n$ and for all $1 \leqslant i < j \leqslant n$, we have

$$f_{ij} = \sum_{(\eta_0,\eta_1,...,\eta_k)\in \mathrm{S}_{ij}} t_{\eta_0,\eta_1} t_{\eta_1,\eta_2} ... t_{\eta_{k-1},\eta_k} f[\lambda_{\eta_0}, ..., \lambda_{\eta_k}] \tag{3.48}$$

where S_{ij} is the set of distinct sequences of integers such that $i = \eta_0 < \eta_1 < ... < \eta_k = j$, $1 \leqslant k \leqslant j - i$ and $f[\lambda_{\eta_0}, ..., \lambda_{\eta_k}]$ is the kth order divided difference of f at $\{\lambda_{\eta_0}, ..., \lambda_{\eta_k}\}$. Computing the function $f(T)$ using the above method requires $O(2^n)$ arithmetic operations, which is computationally infeasible even for matrices of moderate size.

For example, let $A \in \mathbb{C}^{n\times n}$ for the case $n = 2$ and for $\lambda_1 \neq \lambda_2$ we have,

$$T = \begin{pmatrix} \lambda_1 & t_{12} \\ 0 & \lambda_2 \end{pmatrix}, \quad f(T) = \begin{pmatrix} f(\lambda_1) & t_{12}\frac{f(\lambda_2)-f(\lambda_1)}{\lambda_2-\lambda_1} \\ 0 & f(\lambda_2) \end{pmatrix}$$

For $\lambda_1 = \lambda_2 = \lambda$ we have

$$T = \begin{pmatrix} \lambda & t_{12} \\ 0 & \lambda \end{pmatrix}, \quad f(T) = \begin{pmatrix} f(\lambda) & t_{12}f'(\lambda) \\ 0 & f(\lambda) \end{pmatrix}$$

3.7.1 Parlett's Algorithm

A much more efficient and a faster algorithm for computing $F = f(T)$ was proposed by Parlett [7]. The recurrence formula for the Parlett's method is derived from the following commutativity relation

$$FT = TF. \tag{3.49}$$

Because F is a polynomial in T and we know that T and F commute. We write

$$T = \begin{pmatrix} t_{11} & \cdots & t_{1n} \\ & \ddots & \vdots \\ 0 & & t_{nn} \end{pmatrix}, \quad f(T) = \begin{pmatrix} f_{11} & \cdots & f_{1n} \\ & \ddots & \vdots \\ 0 & & f_{nn} \end{pmatrix}$$

$$f(T) = r_{f,T}(T)$$

is a polynomial so it can be written like

$$f(T) = \sum_{j=0}^{n-1} \alpha_j T^j$$

$$f(T) = \sum_{j=0}^{n-1} \alpha_j \begin{pmatrix} t_{11}^j & & * \\ & \ddots & \\ 0 & & t_{nn}^j \end{pmatrix}$$

the k, k elements of this sum is

$$\sum_{j=0}^{n-1} \alpha_j t_{kk}^j = r_{f,t}(t_{kk}) = f(t_{kk})$$

so

$$f_{kk} = f(t_{kk})$$

$$\begin{pmatrix} f_{11} & \cdots & f_{1n} \\ & \ddots & \vdots \\ 0 & & f_{nn} \end{pmatrix} \begin{pmatrix} t_{11} & \cdots & t_{1n} \\ & \ddots & \vdots \\ 0 & & t_{nn} \end{pmatrix} = \begin{pmatrix} t_{11} & \cdots & t_{1n} \\ & \ddots & \vdots \\ 0 & & t_{nn} \end{pmatrix} \begin{pmatrix} f_{11} & \cdots & f_{1n} \\ & \ddots & \vdots \\ 0 & & f_{nn} \end{pmatrix}$$

Where T and diagonal of $F = f(T)$ is known and we want to calculate the off-diagonal elements of $F = f(T)$. Now assume $i < j$ and equate (i, j) entries of the identity of equation(3.49). For example we consider $c_{1,2}$ element such that

$$f_{11}t_{12} + f_{12}t_{22} = t_{11}f_{12} + t_{12}f_{22}$$

$$f_{12} = (t_{12})\frac{f_{11} - f_{22}}{t_{11} - t_{22}}$$

then compute f_{13}, consider the $c_{1,3}$ we have

$$f_{13} = t_{13}\frac{f_{11} - f_{33}}{t_{11} - t_{33}} + \frac{f_{12}t_{23} - t_{12}f_{23}}{t_{11} - t_{33}}$$

Now by comparing (i, j) entries in this way, generally, we obtain the summation formula

$$\sum_{k=i}^{j} f_{ik}t_{kj} = \sum_{k=i}^{j} t_{ik}f_{kj}$$

which gives

$$f_{ij} = t_{ij}\frac{(f_{ii} - f_{jj})}{(t_{ii} - t_{jj})} + \sum_{k=i+1}^{j-1} \frac{(f_{ik}t_{kj} - t_{ik}f_{kj})}{(t_{ii} - t_{jj})} \tag{3.50}$$

which requires that $t_{ii} \neq t_{jj}$ for all $i \neq j$. This recurrence can be evaluated in $\frac{2n^3}{3}$ operations. From equation (3.50) we see that any element of F can be calculated as long as all the elements to the left

and below it are known. Thus the recurrence allows us to compute F a superdiagonal at a time, starting with the diagonal elements $f_{ii} = f(t_{ii})$. Once $F = f(T)$ is calculated, then $f(A)$ is given by $f(A) = Qf(T)Q^*$. The complete procedure is as follows.

Algorithm 1. (Parlett recurrence)[7] Given an upper triangular $T \in \mathbb{C}^{n\times n}$ with distinct eigenvalues $t_{11}, t_{22}, ..., t_{nn}$ and a function $f(x)$, the following algorithm computes $F = f(T)$ assuming that it is defined:

```
for i=1:n
    f(i,i)=f(t(i,i))
    for j=2:n
        for i=j-1:-1:1
            f(i,j) = (t(i,j)*(f(i,i)-f(j,j)))/(t(i,i)-t(j,j));
            for k=i+1:j-1
            f(i,j)=f(i,j)+(f(i,k)*t(k,j)-t(i,k)*f(k,j))/(t(i,i)-t(j,j));
            end
    end
end
```

Consider the following example.

Example 25. *[5, p. 560] If*

$$T = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 3 & 4 \\ 0 & 0 & 5 \end{pmatrix}$$

and

$$f(x) = \frac{1+x}{x} = 1 + \frac{1}{x}$$

then

$$F = (f_{ij}) = f(T)$$

is defined by

$$f_{11} = \frac{1+1}{1} = 2$$

$$f_{22} = \frac{1+3}{3} = \frac{4}{3}$$

$$f_{33} = \frac{1+5}{5} = \frac{6}{5}$$

$$f_{12} = t_{12}\frac{f_{22} - f_{11}}{t_{22} - t_{11}} = -\frac{2}{3}$$

$$f_{23} = t_{23}\frac{f_{33} - f_{22}}{t_{33} - t_{22}} = -\frac{4}{15}$$

$$f_{12} = [t_{13}(f_{33} - f_{11}) + (t_{12}f_{23} - f_{12}t_{23})]t_{33} - t_{11} = -\frac{1}{15}$$

The recurrence (3.50) breaks down, when $t_{ii} = t_{jj}$ for some $i \neq j$ and leads to poor results when $t_{ii} \approx t_{jj}$ for some $i \neq j$. In this case Parlett [7] advises using $T = (T_{ij})$ as a block matrix with square diagonal blocks, possibly of different sizes. The first step is to choose Q in the Schur decomposition such that close or multiple eigenvalues are clustered together in blocks T_{ii} along the diagonal of T. Then we must compute a partitioning

$$T = \begin{pmatrix} T_{11} & T_{12} & \cdots & T_{1p} \\ & T_{22} & \cdots & T_{2p} \\ & & \ddots & \vdots \\ 0 & & & T_{pp} \end{pmatrix} \quad F = \begin{pmatrix} F_{11} & F_{12} & \cdots & F_{1p} \\ & F_{22} & \cdots & F_{2p} \\ & & \ddots & \vdots \\ 0 & & & F_{pp} \end{pmatrix}$$

Where $\lambda(T_{ii}) \cap \lambda(T_{jj}) = \emptyset,\ i \neq j$. Next we compute the submatrices $F_{ii} = f(T_{ii})$ for $i = 1, 2, ..., p$. Once the diagonal blocks of F are known, the blocks in the strict upper triangle of F can be found. To derive the governing equations, we equate (i, j) blocks in $FT = TF$ for $i < j$. The required matrix is then computed one block superdiagonal at a time from

$$F_{ij}T_{jj} - T_{ii}F_{ij} = T_{ij}F_{jj} - F_{ii}T_{ij} + \sum_{k=i+1}^{j-1}(T_{ik}F_{kj} - F_{ik}T_{kj}) \quad i < j \ \ (3.51)$$

provided that it is possible to evaluate the blocks $F_{ii} = f(t_{ii})$ and solve the Sylvester equations (3.51) for the F_{ij}. For the Sylvester equation (3.51) to be nonsingular we need that T_{ii} and T_{jj} have no eigenvalues in common. Moreover, for the Sylvester equation to be well conditioned the eigenvalues of the block T_{ii} must be well separated from those of T_{ij} A reordering of the Schur decomposition $A = UTU^*$ is computed. We compute F_{ii} by some other method.eg; Taylor series. Once F_{ii} known

then compute F_{ij} for $i \neq j$ one diagonal at a time using (3.51). To obtain F_{ij} from (3.51) needs to solve a Sylvester equation.

This is easier that a general Sylvester, since T_{ii} and T_{jj} are upper triangular. The standard approach to this is the Bartels-Stewart algorithm[5, p. 367, Algorithm 7.6.2].

Here is a matlab implementation of Bartels-Stewart algorithm:

```
function [C] = bartels(F, G, C)
%
% solve the sylvester equation
% FZ-ZG = C
% with F, G upper triangular
% C is overwritten by Z
[p r] = size(C);
%
for k=1:r
    C(1:p,k) = C(1:p,k) + C(1:p,1:k-1)*G(1:k-1,k);
    z = (F - G(k,k)*eye(size(F)))\C(1:p,k);
    C(1:p,k) = z;
end
```

Using this, we can compute $f(A)$ using the Block Parlett algorithm. Here is an implementation in the case $f = exp$. By changing the value of $F(r, r)$ in the first loop we can compute $f(A)$ for general functions.

Here is the matlab implementation of Block Parlett algorithm:

```
function [F]= parlett_block_final(T, v)
%
% v = vector of block sizes
%
n = max(size(T));
m = length(v);
s(1) = 1; % start of block i
e(m) = n; % end of block i
for i=1:m-1, e(i) = sum(v(1:i)); end
for i=2:m, s(i) = e(i-1)+1; end
% ith col of w contain first and last index in ith block
w = [s; e];
%
```

```
% compute exp of diagonal blocks
%
for i=1:m,
    r = w(1,i):w(2,i);
    F(r,r) = expm(T(r, r));
% One may use any method to compute the exponential of T(r, r)
end
% compute dth block diag of exp(A)
%
for d=1:m-1
    for j = d+1:m
        i = j-d;
        % solve for F(i,j)
        r = w(1,i):w(2,i); % rows in F(i,j)
        c = w(1,j):w(2,j); % cols in F(i,j)
        rhs = T(r,c)*F(c,c) - F(r,r)*T(r,c);
        for k = i+1:j-1
            rc =w(1,k):w(2,k);
            rhs = rhs + (T(r,rc)*F(rc,c) - F(r,rc)*T(rc,c));
        end
         F(r,c) = -bartels(T(r,r), T(c,c), rhs);
    end
end
```

In section (5.2.2) we present examples to show the difference in accuracy between the Parlett and Block Parlett methods when the matrix has close eigenvalues. The examples also show the importance of choosing the blocks well.

Chapter 4

The Matrix Exponential Theory

How to calculate the exponential of matrices in an explicit manner is an important problem in many areas of science. There are many different methods which have been proposed for computing e^A the exponential of a matrix. The following well known, see for example [7, p. 234]:

$$Power\ series \quad I + A + \frac{A^2}{2!} + ... \tag{4.1}$$

$$Limit\ of\ Powers \quad \lim_{n\to\infty}(I + \frac{A}{n})^n \tag{4.2}$$

$$Scaling\ and\ squaring \quad (e^{\frac{A}{2^s}})^{2^s} \tag{4.3}$$

$$Cauchy\ integral\ \frac{1}{2\pi i}\int_\Gamma e^z(zI - A)^{-1}dz \tag{4.4}$$

$$Jordan\ Form\ Xdiag(e^{J_k})X^{-1} \tag{4.5}$$

$$Interpolation \quad \sum_{i=1}^{n} f[\lambda_1, ..., \lambda_i]\prod_{j=1}^{i-1}(A - \lambda_j I) \tag{4.6}$$

$$Schur\ form \quad Qdiag(e^T)Q^* \tag{4.7}$$

$$Pade\ approximation \quad p_{mn}(A)q_{mn}(A)^{-1} \tag{4.8}$$

None of these is entirely satisfactory from either a theoretical or a computational point of view, as we shall see.

4.1 Matrix exponential

The numerical evaluation of the exponential of a matrix is of some importance because of its occurrence in many physical, engineering, and economics applications. One of the reasons for the importance of the matrix exponential is that it can be used to solve system of linear first order constant coefficient ordinary differential equation,

$$\dot{x}(t) = Ax(t), \quad x(0) = x_0 \quad 0 \leqslant t < \infty \tag{4.9}$$

Where x and x_0 are n-dimensional column vectors functions with respect to t and $A \in \mathbb{C}^{n\times n}$ is a given, fixed, real or complex $n \times n$ matrix. It is well known that the theoretical solution to this equation is given by

$$x(t) = e^{At}x_0,$$

where e^{At} can be formally defined by the convergent power series.

Definition 28. For a square matrix $A \in \mathbb{M}_n$ we define the matrix exponential as

$$e^{At} = \sum_{k=0}^{\infty} \frac{(tA)^k}{k!} = I + tA + \frac{(tA)^2}{2!} + ... \tag{4.10}$$

where $A^0 = I_n$.
This converges for all A and uniformly in t, at $t = 1$, we have,

$$e^{A} = \sum_{k=0}^{\infty} \frac{(A)^k}{k!} = I + A + \frac{(A)^2}{2!} + ... \tag{4.11}$$

where $A^0 = I_n$

Note that this is the generalization of the Taylor series expansion of the standard exponential function. The series (4.11) converges absolutely for all $A \in \mathbb{C}^{n\times n}$ (has radius of convergence equal to $+\infty$), so the exponential of A is well-defined. To prove the convergence of the series, we have the following theorem.

Theorem 17. [1, p. 420 Prop. 11.1.2] The series (4.11) converges absolutely for all $A \in \mathbb{M}_n$. Furthermore, let $\|.\|$ be a normalized submultiplicative norm on $\mathbb{M}_n$. Then

$$\|e^{A}\| \leqslant e^{\|A\|} \tag{4.12}$$

Proof. The *nth* partial sum is

$$S_n = \sum_{k=0}^{n} \frac{A^k}{k!}$$

so

$$\begin{aligned} \|e^A - S_n\| = \|\sum_{k=0}^{\infty} \frac{A^k}{k!} - \sum_{k=0}^{m} \frac{A^k}{k!}\| &= \|\sum_{k=m+1}^{\infty} \frac{A^k}{k!}\| \\ &\leqslant \sum_{k=m+1}^{\infty} \frac{\|A^k\|}{k!} \\ &\leqslant \sum_{k=m+1}^{\infty} \frac{\|A\|^k}{k!} \end{aligned} \tag{4.13}$$

Since $\|A\|$ is a real number, the right-hand side is a part of the convergent series of real numbers

$$e^{\|A\|} = \sum_{k=0}^{\infty} \frac{1}{k!}\|A\|^k \tag{4.14}$$

Hence, since (4.14) is convergent, if $\epsilon > 0$, there is an N such that for $m \geqslant N$,

$$e^{\|A\|} = \sum_{k=m+1}^{\infty} \frac{\|A\|^k}{k!} < \epsilon$$

This is sufficient to prove that S_n is convergent.
Furthermore, note that

$$\|e^A\| = \|\sum_{k=0}^{\infty} \frac{1}{k!}A^k\| \leqslant \sum_{k=0}^{\infty} \frac{1}{k!}\|A^k\| \leqslant \sum_{k=0}^{\infty} \frac{1}{k!}\|A\|^k = e^{\|A\|}.$$

□

In some cases, it is a simple matter to express the matrix exponential. For example, suppose when A is a nilpotent matrix, i.e., $A^p = 0$ for some natural number p, the exponential is given by a matrix polynomial because some power of A vanishes.

$$e^A = I + A + \frac{1}{2!}A^2 + \frac{1}{3!}A^3 + ... + \frac{1}{(p-1)!}A^{p-1}$$

Consider the following example.

Example 26. *When*

$$A = \begin{pmatrix} 0 & 1 & 3 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{pmatrix}$$

then

$$exp(A) = \begin{pmatrix} 1 & 1 & 4 \\ 0 & 1 & 2 \\ 0 & 0 & 1 \end{pmatrix}$$

and $A^3 = 0$

If A is a 1×1 matrix $A = [t]$, then $e^A = e^t$, by the Maclaurin series formula for the function $y = e^t$. More generally [20], if A is a diagonal matrix having diagonal entries $(a_1, a_2, ..., a_n)$, then we have

$$A = \begin{pmatrix} a_1 & 0 & \cdots & 0 \\ 0 & a_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_n \end{pmatrix}$$

$$e^A = I + A + \frac{1}{2!}A^2 + ...$$

and its exponential is

$$e^A = exp(A) = \begin{pmatrix} e^{a_1} & 0 & \cdots & 0 \\ 0 & e^{a_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & e^{a_n} \end{pmatrix}$$

Example 27. *Consider the matrix*

$$A = \begin{pmatrix} 2 & 0 \\ 0 & 3 \end{pmatrix}$$

then

$$e^A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \begin{pmatrix} 2 & 0 \\ 0 & 3 \end{pmatrix} + \frac{1}{2!}\begin{pmatrix} 2^2 & 0 \\ 0 & 3^2 \end{pmatrix} + \frac{1}{3!}\begin{pmatrix} 2^3 & 0 \\ 0 & 3^3 \end{pmatrix} + ...$$

$$e^A = \begin{pmatrix} 1 + 2 + \frac{1}{2!}2^2 + \frac{1}{3!}2^3 + ... & 0 \\ 0 & 1 + 3 + \frac{1}{2!}3^2 + \frac{1}{3!}3^3 + ... \end{pmatrix}$$

$$= \begin{pmatrix} e^2 & 0 \\ 0 & e^3 \end{pmatrix}$$

This also allows one to exponentiate a diaonalizable matrix. If a matrix A is diagonalizable, then there exists an invertible S so that $A = S\Lambda S^{-1}$ where Λ is a diagonal matrix of eigenvalues of A, and S is a matrix having eigenvectors of A as its columns. then

$$e^A = \sum_{k=0}^{\infty} \frac{1}{k!}(S\Lambda S^{-1})^k = \sum_{k=0}^{\infty} \frac{1}{k!} S\Lambda^k S^{-1}$$

$$= S(\sum_{k=0}^{\infty} \Lambda^k) S^{-1} = Se^{\Lambda} S^{-1}$$

where $S = (s_1, s_2, ..., s_n), \quad \Lambda = diag(\lambda_1, \lambda_2, ..., \lambda_n),$
$e^{\Lambda} = diag(e^{\lambda_1}, e^{\lambda_2}, ..., e^{\lambda_n})$
Let us illustrate an example that compute e^A

Example 28.

$$A = \begin{pmatrix} 5 & 1 \\ -2 & 2 \end{pmatrix}$$

The characteristic equation for eigenvalues is

$$p(\lambda) = |A - \lambda I| = 0$$

and it yields the eigenvalues $\lambda_1 = 4$, $\lambda_2 = 3$, *with corresponding eigenvectors*

$$s_1 = \begin{pmatrix} 1 \\ -1 \end{pmatrix} \quad and \quad s_2 = \begin{pmatrix} 1 \\ -2 \end{pmatrix}.$$

It follows that,

$$A = SDS^{-1} = \begin{pmatrix} 1 & 1 \\ -1 & -2 \end{pmatrix} \begin{pmatrix} 4 & 0 \\ 0 & 3 \end{pmatrix} \begin{pmatrix} 2 & 1 \\ -1 & -1 \end{pmatrix}$$

so that

$$e^A = \begin{pmatrix} 1 & 1 \\ -1 & -2 \end{pmatrix} \begin{pmatrix} e^4 & 0 \\ 0 & e^3 \end{pmatrix} \begin{pmatrix} 2 & 1 \\ -1 & -1 \end{pmatrix}$$

$$e^A = \begin{pmatrix} 2e^4 - e^3 & e^4 - e^3 \\ 2e^3 - 2e^4 & 2e^3 - e^4 \end{pmatrix}$$

$$= \begin{pmatrix} 89.1108 & 34.5126 \\ -69.0252 & -14.4271 \end{pmatrix}$$

Generally a matrix $A \in \mathbb{M}_n$ has a decomposition known as *SN-Decomposition* or *Jordan-Chevalley decomposition* which is similar to the canonical decomposition, which can be written as

$$A = S + N$$

where

- S is diagonalizable,
- N is nilpotent, such that $N^k = 0$
- S commutes with N i.e $SN = NS$
  it means that the exponential of A can be written as

$$e^A = e^{S+N} = e^S e^N$$

We will compute $exp(A)$ using the SN-decomposition in the following example.

Example 29. *Consider the matrix*

$$A = \begin{pmatrix} \lambda & 1 \\ 0 & \lambda \end{pmatrix}$$

the SN-decomposition is

$$A = S + N$$

where

$$S = \begin{pmatrix} \lambda & 0 \\ 0 & \lambda \end{pmatrix} \quad and \quad N = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$$

Now, we have

$$exp(St) = \begin{pmatrix} e^{\lambda t} & 0 \\ 0 & e^{\lambda t} \end{pmatrix} \quad and \quad exp(Nt) = I + Nt = \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix}$$

so that

$$exp(At) = \begin{pmatrix} e^{\lambda t} & 0 \\ 0 & e^{\lambda t} \end{pmatrix} \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} e^{\lambda t} & te^{\lambda t} \\ 0 & e^{\lambda t} \end{pmatrix}$$

When it is not possible to find n linearly independent eigenvectors of A, the matrix A is not diagonalizable. In this case we can use a closely related method based on the *Jordan* form of A. Suppose J is the *Jordan* form of A, with X the transition matrix. Then

$$e^A = Xe^J X^{-1}$$

Also, since

$$J = diag(J_1(\lambda_1), J_2(\lambda_2), ..., J_p(\lambda_p))$$

$$J = J_1(\lambda_1) \oplus J_2(\lambda_2)... \oplus J_p(\lambda_p)$$

so

$$e^J = exp(J_1(\lambda_1)) \oplus exp(J_2(\lambda_2))... \oplus exp(J_p(\lambda_p))$$

Therefore, we need only know how to compute the matrix exponential of a *Jordan block*. But each *Jordan block* is of the form

$$J_k(\lambda) = \lambda_k I + N_k \in \mathbb{M}_k$$

where

$$N_k \equiv J_k(0) = \begin{pmatrix} 0 & 1 & & \\ & 0 & \ddots & \\ & & \ddots & 1 \\ & & & 0 \end{pmatrix}$$

where N is a special nilpotent matrix. Since $(N_k)^m = 0$ for all $m \geqslant k$. The matrix exponential of this block is given by

$$e^{\lambda_k I + N_k} = e^{\lambda_k} e^{N_k} = e^{\lambda_k}(I + N + \frac{1}{2!}N^2 + ... + \frac{1}{(k-1)!N^{k-1}})$$

Now consider the following example for computing the matrix exponential via *Jordan form*.

Example 30. *Compute e^A for the matrix*

$$A = \begin{pmatrix} -7 & -4 & -3 \\ 10 & 6 & 4 \\ 6 & 3 & 3 \end{pmatrix}$$

Then the eigenvalues are 0, 1, 1 *We have $A = XJX^{-1}$, where*

$$J = \begin{pmatrix} 0 \end{pmatrix} \oplus \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

and

$$X = \begin{pmatrix} 1 & -1 & -1 \\ -1 & 2 & 0 \\ -1 & 0 & 3 \end{pmatrix}$$

Hence, using the Jordan canonical form definition, we have

$$e^A = Xe^J X^{-1} = X \begin{pmatrix} 1 & 0 & 0 \\ 0 & e & e \\ 0 & 0 & e \end{pmatrix} X^{-1}$$

$$= \begin{pmatrix} 1 & -1 & -1 \\ -1 & 2 & 0 \\ -1 & 0 & 3 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & e & e \\ 0 & 0 & e \end{pmatrix} \begin{pmatrix} 6 & 3 & 2 \\ 2 & 2 & 1 \\ 2 & 1 & 1 \end{pmatrix}$$

$$e^A = \begin{pmatrix} 6-7e & 3-4e & 2-3e \\ -6+10e & -3+6e & -2+4e \\ -6+6e & -3+3e & -2+3e \end{pmatrix}$$

4.2 The Matrix Exponential as a Limit of Powers

From calculus we know that for any numbers a and t the exponential is

$$e^{at} = \lim_{n\to\infty}(1 + \frac{at}{n})^n \tag{4.15}$$

from equation (4.16) one can define the matrix exponential as a limit of powers as

$$e^{At} = \lim_{n\to\infty}(I + \frac{At}{n})^n \tag{4.16}$$

or

$$e^{A} = \lim_{n\to\infty}(I + \frac{A}{n})^n \tag{4.17}$$

This formula is the limit of the first order Taylor expansion of A/n raised to the power $n \in \mathbb{Z}$.

Example 31. *Consider the matrix,*

$$A = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}$$

$$(I + \frac{At}{n})^n = \begin{pmatrix} (1+\frac{t}{n})^n & 0 \\ 0 & (1+\frac{2t}{n})^n \end{pmatrix}$$

and so apply eq(4.17)

$$\lim_{n\to\infty}(I + \frac{At}{n})^n = \begin{pmatrix} \lim_{n\to\infty}(1+\frac{t}{n})^n & 0 \\ 0 & \lim_{n\to\infty}(1+\frac{2t}{n})^n \end{pmatrix}$$

$$e^{At} = \begin{pmatrix} e^t & 0 \\ 0 & e^{2t} \end{pmatrix}$$

4.3 The Matrix Exponential via Interpolation

This approach is well known. See for example [9, p. 391-92]

4.3.1 Lagrange Interpolation Formula

Let $\lambda_1, ..., \lambda_k$ be the distinct eigenvalues of a matrix $A \in \mathbb{M}_n$ and $f(t)$ is any function that is well defined at the eigenvalues of A, then the *Lagrange formula* for $f(A)$ is

$$f(t) = \sum_{i=1}^{k} f(\lambda_i) \prod_{j=1,\, j\neq i}^{k} \frac{t - \lambda_j}{\lambda_i - \lambda_j}$$

$$f(A) = \sum_{i=1}^{k} f(\lambda_i) \prod_{j=1,\, j\neq i}^{k} \frac{A - \lambda_j I}{\lambda_i - \lambda_j} \tag{4.18}$$

4.3.2 Newton's Divided Difference Interpolation

Let $A \in \mathbb{M}_n$ be a matrix with eigenvalues $\lambda(A) = \{\lambda_1, ..., \lambda_n\}$. Now we define $f(A)$ as follows:

$$f(A) = \sum_{i=1}^{n} f[\lambda_1 ..., \lambda_i] \prod_{j=1}^{i-1} (A - \lambda_j I), \tag{4.19}$$

where $f[\lambda_1 ..., \lambda_i]$ is the divided difference at $\lambda_1 ..., \lambda_i$. If $\lambda_1 ..., \lambda_i$ are distinct we compute the divided difference recursively by

$$f[\lambda_1] = f(\lambda_1)$$

$$f[\lambda_1, \lambda_2] = \frac{f(\lambda_2) - f(\lambda_1)}{\lambda_2 - \lambda_1}$$

$$f[\lambda_1, \lambda_2, ..., \lambda_n] = \frac{f[\lambda_2, ..., \lambda_n] - f[\lambda_1, ..., \lambda_{n-1}]}{\lambda_n - \lambda_1}, \quad n \geqslant 1$$

Note that

$$f(\lambda_i I) = f(\lambda_i)I \quad for\ i = 0, 1, 2, ..., n$$

Otherwise we use the fact that the value of a divided difference is independent of the order of the arguments, that is

$$f[\lambda_1, \lambda_2, ..., \lambda_k] = f[\lambda_{i_1}, ..., \lambda_{i_k}]$$

where $\{i_1, ..., i_k\} = \{1, ..., k\}$ e.g., $f[\lambda_1, \lambda_2, \lambda_3] = f[\lambda_2, \lambda_3, \lambda_1] = f[\lambda_3, \lambda_2, \lambda_1] =$ ... so we order $\lambda_1, ..., \lambda_n$ so that the identical eigenvalues are together. Then we may also use the identity

$$f[\lambda_1, ..., \lambda_k] = \frac{f^{(k-1)}\lambda_1}{(k-1)!}$$

if $\lambda_1 = \lambda_2 = ... = \lambda_k$. So for example if $\lambda_1 = 1$, $\lambda_2 = 2$, $\lambda_3 = 1$. Then

$$f[\lambda_1, \lambda_2, \lambda_3] = f[1, 2, 1] = f[1, 1, 2]$$

$$= \frac{f[1,1] - f[1,2]}{1-2} = \frac{f'(1) - (f(1) - f(2)/(1-2))}{1-2}$$

Let us illustrate the following examples.

Example 32.

$$A = \begin{pmatrix} -49 & 24 \\ -64 & 31 \end{pmatrix}$$

and $f(x) = e^x$

The eigenvalues are $\lambda(A) = \{-1, -17\}$

$f(-1) = e^{-1}$ *and* $f(-17) = e^{-17}$.

Let $(\lambda_0, f(\lambda_0)) = (-1, e^{-1})$ *and*
$(\lambda_1, f(\lambda_1)) = (-17, e^{-17})$

Then by definition

$$f(A) = f(\lambda_0)I + f[\lambda_0, \lambda_1](A - \lambda_0 I)$$

$$e^A = \begin{pmatrix} e^{-1} & 0 \\ 0 & e^{-1} \end{pmatrix} + \frac{e^{-17} - e^{-1}}{-17+1}\{\begin{pmatrix} -49 & 24 \\ -64 & 31 \end{pmatrix} - \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}\}$$

$$e^A = \begin{pmatrix} -0.735759 & 0.551819 \\ -1.471518 & 1.103638 \end{pmatrix}$$

Example 33.

$$A = \begin{pmatrix} 3 & -1 \\ 1 & 1 \end{pmatrix}$$

and $f(x) = e^x$

The repeated eigenvalues are $\lambda(A) = \{2, 2\}$

$f(2) = e^2$ *and* $f'(2) = e^2$.

Let $(\lambda, f(\lambda)) = (2, e^2)$ *and*

Then by definition

$$f(A) = f(\lambda)I + f[\lambda, \lambda](A - \lambda I)$$

$$f(A) = f(2)I + f'(2)(A - 2I)$$

$$e^A = \begin{pmatrix} e^2 & 0 \\ 0 & e^2 \end{pmatrix} + e^2\{\begin{pmatrix} 3 & -1 \\ 1 & 1 \end{pmatrix} - \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}\}$$

$$e^A = e^2 \begin{pmatrix} 2 & -1 \\ 1 & 0 \end{pmatrix}$$

4.4 Additional Theory

In this final part of this chapter we collect for reference additional important properties of the matrix exponential that are not needed in the development.

Theorem 18. [8, p. 435, Theorem 6.2.38] Let A, $B \in \mathbb{M}_n$ be given. If $AB = BA$, then $e^{A+B} = e^A e^B = e^B e^A$

Proof. We use the power series for e^t to compute

$$e^A e^B = (\sum_{r=0}^{\infty} \frac{1}{r!} A^r)(\sum_{s=0}^{\infty} \frac{1}{s!} B^s) = \sum_{r,s \geqslant 0}^{\infty} \frac{1}{r!s!} A^r B^s$$

$$= \sum_{n=0}^{\infty} \sum_{k=0}^{n} \frac{A^k B^{n-k}}{k!(n-k)!} = \sum_{n=0}^{\infty} \frac{1}{n!} \sum_{k=0}^{n} \frac{n!}{k!(n-k)!} A^k B^{n-k}$$

$$= \sum_{n=0}^{\infty} \frac{1}{n!} (A+B)^n = e^{A+B}$$

Hence follows that $e^{A+B} = e^A e^B = e^{B+A} = e^B e^A$. □

The conclusion fails for general A and B. Consider the following example.

Example 34.

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \quad \textit{and} \quad B = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$$

We use Matlab to compute the relevant quantities.

```
>> expm(A)

ans =

     1     1
     0     1

>> expm(B)

ans =

    2.7183         0
         0    1.0000

>> expm(A)*expm(B)

ans =

    2.7183    1.0000
         0    1.0000

>> expm(A+B)

ans =
```

```
    2.7183    1.7183
         0    1.0000

>> expm(B)*expm(A)

ans =

    2.7183    2.7183
         0    1.0000
```

Hence the above example shows that statement $e^A e^B \neq e^B e^A = e^{A+B}$ *does not hold for general* A *and* B

Theorem 19. [1, p. 421, Prop. 11.1.6] Let $A, B \in \mathbb{C}^{n\times n}$. Then, $AB = BA$ if and only if, for all $t \in \mathbb{R}$,

$$e^{(A+B)t} = e^{At}e^{Bt} \tag{4.20}$$

Proof. Suppose $AB = BA$. Then,

$$e^{(A+B)t} = I + t(A+B) + \frac{1}{2!}t^2(A+B)^2 + ...$$

$$e^{At} = I + tA + \frac{1}{2!}t^2A^2 + ...$$

$$e^{Bt} = I + tB + \frac{1}{2!}t^2B^2 + ...$$

$$e^{At}e^{tB} = (I + tA + \frac{1}{2!}t^2A^2 + ...)(I + tB + \frac{1}{2!}t^2B^2 + ...)$$

$$e^{At}e^{Bt} = I + t(A+B) + \frac{1}{2!}t^2(A^2 + 2BA + B^2) + ...$$

since $AB = BA$,

$$e^{At}e^{Bt} = I + t(A+B) + \frac{1}{2!}t^2(A+B)^2 + ...$$

it can be seen that the expansions are identical. Conversely, differentiating (4.20) twice with respect to t and setting $t = 0$, we have, $AB = BA$ □

Theorem 20. [1, p. 422, Prop. 11.1.8] Let I_n, $A \in \mathbb{C}^{n\times n}$ and I_m, $B \in \mathbb{C}^{m\times m}$. then,

$$e^{A\otimes I_m} = e^A \otimes I_m \tag{4.21}$$

$$e^{I_n\otimes B} = I_n \otimes e^B \tag{4.22}$$

$$e^{A\oplus B} = e^A \otimes e^B \tag{4.23}$$

where $A \oplus B = A \otimes I_m + I_n \otimes B$.

Proof. Since we have

$$e^{A\otimes I_m} = I_{nm} + A \otimes I_m + \frac{1}{2!}(A \otimes I_m)^2 + ...$$

$$= I_n \otimes I_m + A \otimes I_m + \frac{1}{2!}(A^2 \otimes I_m) + ...$$

$$= (I_n + A + \frac{1}{2!}A^2 + ...) \otimes I_m = e^A \otimes I_m$$

and similarly we can prove (4.22). To prove (4.23), note that

$$(A \otimes I_m)(I_n \otimes B)A \otimes B = (I_n \otimes B)(A \otimes I_m)$$

this shows that $A \otimes I_m$ and $I_n \otimes B$ commute. Thus, by Theorem(19),

$$e^{A\oplus B} = e^{A\otimes I_m + I_n\otimes B} = e^{A\otimes I_m} e^{I_n\otimes B}$$

$$= (e^A \otimes I_m)(I_n \otimes e^B) = e^A \otimes e^B.$$

□

Theorem 21. [11, p. 109, 11.1.1] Let $A \in \mathbb{M}_n$ be given. Then

1. $e^{(A^T)} = (e^A)^T$. It follows that if A is symmetric then e^A is also symmetric.
2. Let $A \in \mathbb{C}^{n\times n}$ and $s, t \in \mathbb{C}$. Then
$$e^{A(s+t)} = e^{As}e^{At} \tag{4.24}$$
3. e^A is nonsingular, and $(e^A)^{-1} = e^{-A}$.
4. If $A \in \mathbb{M}_n$ is skew-symmetric, then e^A is orthogonal.i.e.,
$$e^A(e^A)^T = I_n$$

.

Proof. 1. We know that

$$e^A = \sum_{k=0}^{\infty} \frac{A^k}{k!}$$

If A is symmetric i.e; $A^T = A$, then

$$e^{A^T} = \sum_{k=0}^{\infty} \frac{(A^T)^k}{k!} = \sum_{k=0}^{\infty} \frac{(A^k)^T}{k!}$$

$$= (\sum_{k0}^{\infty} \frac{A^k}{k!})^T = (e^A)^T$$

2. From the definition (4.10) we have

$$e^{As}e^{At} = (I + As + \frac{A^2s^2}{2!} + ...)(I + At + \frac{A^2t^2}{2!} + ...)$$

$$= (\sum_{j=0}^{\infty} \frac{A^j s^j}{j!})(\sum_{k=0}^{\infty} \frac{A^k t^k}{k!})$$

$$= \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} \frac{A^{j+k} s^j t^k}{j!k!} \tag{4.25}$$

Let $m = j + k$, then $j = m - k$. This follows from (4.24) and the binomial theorem that

$$e^{As}e^{At} = \sum_{m=0}^{\infty} \sum_{k=0}^{\infty} \frac{A^m s^{m-k} t^k}{(m-k)!k!}$$

$$= \sum_{m=0}^{\infty} \frac{A^m}{m!} \sum_{k=0}^{\infty} \frac{m!}{(m-k)!k!} s^{m-k} t^k$$

$$= \sum_{m=0}^{\infty} \frac{A^m (s+t)^m}{m!} = e^{A(s+t)}$$

3. setting $s = -1$ and $t = -1$ in (4.24), we have

$$e^A e^{-A} = e^{A(1+(-1))} = e^0 = I_n$$

It proves that the exponential matrix e^A is always invertible, and has inverse e^{-A}.

4. If A is skew-symmetric we have $A^T = -A$ and hence by part 1 and 3

$$(e^A)^T = e^{A^T} = e^{-A} = (e^A)^{-1}$$
$$(e^A)^T e^A = I_n = e^A (e^A)^T$$

Therefore, e^A is orthogonal matrix.

□

Chapter 5

The Matrix Exponential Functions: Algorithms

The focus of this chapter is to examine the accuracy and computational cost of methods to compute e^A. There are many models of physical application which involve the calculation of the matrix exponential. However, we are going to focus on the advantages and disadvantages of some truncated series methods.

In §5.1 we take a look at the Taylor series, Padé Approximation and Scaling and Squaring techniques.

5.1 Series Methods

In this section we will describe the various series methods to computing the matrix exponential which include; Taylors series, Padé approximations and scaling and squaring.

By using properly the Taylor series and Padé approximation one can develop efficient computational methods.[13]

5.1.1 Taylor Series

Let the exponential function $f(x)$ in the scalar case can be defined by its convergent infinite Taylor series

$$e^x = \sum_{k=0}^{\infty} \frac{x^k}{k!} = 1 + x + \frac{1}{2!}x^2 + ... + \frac{x^k}{k!} + ...$$

for $x \in \mathbb{C}$. In another analogy to the scalar case, let $A \in \mathbb{M}_n$. The matrix exponential can be formally defined by its convergent infinite Taylor series

$$e^A = \sum_{k=0}^{\infty} \frac{A^k}{k!} = I + A + \frac{1}{2!}A^2 + ..., \tag{5.1}$$

where $A^0 = I_n$ is the identity matrix. The above series always converges, so the exponential is well-defined.

Here is the m-file for the Taylor series method:

```
function [e] = exp_taylor(A, m)
%
% compute exp using taylor series m terms
%(alternatively can stop when norm(term) < 10-15*norm(S))
%
n = max(size(A));
S = eye(n);
%
term = S;
for k=1:m,
    term = term * A/k;
    S = S + term;
end
e = S;
```

To calculate the matrix exponential we would like to use computers and we will only be able to approximate the exponential with a truncated Taylor series of k terms. The truncated Taylor series is denoted by $T_k(A)$, where the subscript k represent the highest power of the matrix A in the truncated series

$$e^A = T_k(A) = \sum_{i=0}^{k} \frac{A^k}{k!}. \tag{5.2}$$

For the Taylor series, the order of the approximation is seen to be the highest power of the truncated Taylor series which will be denoted by k and can be find such that,

$$\|T_k(A) - e^A\| \leqslant (\frac{\|A\|^{k+1}}{(k+1)!})(\frac{1}{1 - \|A\|/(k+2)})$$

We use this quantity k to compare methods which are approximating the same function, but for approximations of different functions this comparison can not be made. There are many other factors that can affect the accuracy of a solution technique. Since the order of the approximation does not necessarily reflect the amount of work that required to obtain the approximation.

For accuracy and time efficiency, the result for the matrix exponential depends on the matrix norm, i.e., $\|A\|$. The matrix exponential calculation can have problems converging, i.e., the number of terms k in (5.2) may have to be very large, which in turn may cause inaccuracy due to numerical roundoff. This situation occurs when the entries of A are large, or equivalently the norm of A is large ($>> 1$), causing the numerator of the summand in (5.2) to increase rapidly with increasing powers n. On the other hand if $\|A\| < 1$, then all the terms of the product (A) are < 1, and the number of terms required to achieve reasonable accuracy is relatively small.

Let us examine the effect of rounding errors in evaluating

$$e^x = \sum_{k=0}^{\infty} \frac{x^k}{k!}.$$

Let $fl(z)$ denote the value of z computed in floating point arithmetic. It can be shown that

$$fl(\frac{x^k}{k!}) = \frac{x^k}{k!}(1 + \epsilon_k)$$

where $|\epsilon_k| \leqslant (2k-1)\epsilon_M$ and ϵ_M is *machine epsilon* or *unit round off*. Since $k \leqslant m$, we have

$$|\epsilon_k| \leqslant 2m\epsilon_M.$$

Thus

$$\begin{aligned} & |fl(\sum_{k=0}^{m} \frac{x^k}{k!}) - \sum_{k=0}^{m} \frac{x^k}{k!}| \\ = & |\sum_{k=0}^{m} fl(\frac{x^k}{k!}) - \sum_{k=0}^{m} \frac{x^k}{k!}| \\ \leqslant & \sum_{k=0}^{m} |fl(\frac{x^k}{k!}) - \frac{x^k}{k!}| \\ \leqslant \sum_{k=0}^{m} |\frac{x^k}{k!}(1+\epsilon_k) - \sum_{k=0}^{m} \frac{x^k}{k!}| & = \sum_{k=0}^{m} |\frac{x^k}{k!}\epsilon_k| = \sum_{k=0}^{m} \frac{|x^k|}{k!}|\epsilon_k| \approx 2m\epsilon_M e^{|x|} \end{aligned}$$

Thus

$$rel\ err = \frac{|fl(\sum_{k=0}^{m} \frac{x^k}{k!}) - \sum_{k=0}^{m} \frac{x^k}{k!}|}{|\sum_{k=0}^{m} \frac{x^k}{k!}|}$$

$$\leqslant \frac{2m\epsilon_M \sum_{k=0} m\frac{x^k}{k!}}{|\sum_{k=0} m\frac{x^k}{k!}} \approx 2m\epsilon_M \frac{e^{|x|}}{|e^x|}$$

$$= \begin{cases} 2m\epsilon_M & if x \geqslant 0 \\ 2m\epsilon_M e^{2|x|} & if x < 0 \end{cases}$$

So if $x \geqslant 0$, we have shown that the relative error will be small. However if $x < 0$, then the relative error can be large, especially if $|x|$ is large. For example, take $x = -10$. We compute e^{-10} with $m = 100$ and get $4.539992962303128e-005$ which has relative error of $9.703303908195806e-006$, even though we are using $\epsilon_M \approx 10^{-16}$. This agrees with the error bound $2m\epsilon_M e^{2|x|}$. The point of this example is that if $x \in \mathbb{R}$ is negative then $|e^x| < 1$, but $e^x = \sum_{k=0}^{\infty} \frac{x^k}{k!}$ has terms that are larger then e^x, so there is cancelation, which results in large relative errors. Notice that this scalar example shows that cancelation can cause a large relative error in the computation of the matrix exponential even when $n = 1$.

For example, let $A = [10] \in \mathbb{M}_n$. Then $\|A\| = 10$. To compute e^A to an accuracy of 10^{-10},

$$\sum_{k=m+1}^{\infty} \frac{A^k}{(m+1)!} \leqslant 10^{-10}$$

need

$$m + 1 = 43 \ \Rightarrow \ m = 42$$

It means that Taylor need $m = 42$, to compute $A, A^2, ..., A^{42}$, i.e., 41 matrix multiplications.

A table for the relative errors of the scalar exponential is given below. There are two cases to be discussed; first, $x = 0.1 < 1$ and second $x = 10 > 1$.

Table 3.

Relative Error for e^x		
m	$x = 0.1$	$x = 10$
1	*0.00467884016044*	*0.99950060077261*
2	*1.546530702647670e-004*	*0.99723060428449*
3	*3.846833925457031e-006*	*0.98966394932407*
4	*7.667801704962426e-008*	*0.97074731192304*
5	*1.274898869421281e-009*	*0.93291403712097*
10	*4.018285340183601e-016*	*0.41696024980701*
15	*4.018285340183601e-016*	*0.04874040330398*
20	*4.018285340183601e-016*	*0.00158826066186*
25	*4.018285340183601e-016*	*1.768027241765956e-005*
30	*4.018285340183601e-016*	*7.983794668242912e-008*

The table shows that the truncated Taylor series gives an accurate estimate for cases where the terms of x are small. The table also shows that the truncated Taylor series may require a very large amount of work and still not gives an acceptable accuracy. Where as in the second case it shows clearly that the truncated series is an inefficient method to compute the matrix exponential.

In the case of matrix where A is negative definite and large $\|A\|$ e.g $A = [-100]$ 100 terms of Taylor series does not produce a good estimate, need approximately 200 terms until relative error $\approx 10^{-16}$ a lot of computation is required.

Here is a matlab code to approximate $exp(A)$ based on the series definition. We prescribe a tolerance, which determines the number of terms retained in the partial sum approximating the series. We also compare the results to those obtained using the matlab routine *expm* by computing the difference in the l_∞ norm; this error is displayed at each iteration:

```
function X = exp_T_series(A,tol);
% Usage: X = exp_matrix_series(A,tol);
% Calculates the matrix exponential of A to required tolerance using the
% series definition.
% Input:
% A = matrix
% tol = absolute tolerance
% Output:
% X = approximate value of matrix exponential
[n,n]=size(A);
```

```
P = eye(n,n);
X = eye(n,n);
diff = 1000;
exact = expm(A); %Exact value for matrix exponential
i = 0;
while diff > tol
i = i+1;
P = P*(A/i);
X = X + P;
diff = norm(P,inf)/norm(X, inf);
err = norm(exact-X,inf);
rel_err = err/norm(expm(A));
display([sprintf('iteration = %2.0f, error = %e',i,err)]);
display([sprintf('iteration = %2.0f, rel_error = %e',i,rel_err)]);
%i
X
end
```

Note: The code is not indented so that it fits on the page.

Consider the following test matrices:

$$A1 = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}, \; A2 = \begin{pmatrix} -49 & 24 \\ -64 & 31 \end{pmatrix}$$

The matrix X is the current approximation to the exponential.

```
>> A1 = [2 -1; -1 2]

A1 =

     2    -1
    -1     2

>> exp_T_series(A1,1e-14)
iteration =  1, error = 1.608554e+001
iteration =  1, rel_error = 8.008517e-001

X =

     3    -1
    -1     3
```

```
iteration =  2, error = 1.158554e+001
iteration =  2, rel_error = 5.768099e-001

X =

    5.5000   -3.0000
   -3.0000    5.5000

iteration =  3, error = 7.085537e+000
iteration =  3, rel_error = 3.527681e-001

X =

    7.8333   -5.1667
   -5.1667    7.8333

iteration =  4, error = 3.710537e+000
iteration =  4, rel_error = 1.847368e-001

X =

    9.5417   -6.8333
   -6.8333    9.5417

iteration =  5, error = 1.685537e+000
iteration =  5, rel_error = 8.391794e-002

X =

   10.5583   -7.8417
   -7.8417   10.5583

iteration =  6, error = 6.730369e-001
iteration =  6, rel_error = 3.350854e-002

X =

   11.0653   -8.3472
   -8.3472   11.0653
```

```
iteration =  7, error = 2.391084e-001
iteration =  7, rel_error = 1.190450e-002

X =

   11.2823   -8.5641
   -8.5641   11.2823

iteration =  8, error = 7.638514e-002
iteration =  8, rel_error = 3.802992e-003

X =

   11.3637   -8.6454
   -8.6454   11.3637

iteration =  9, error = 2.214407e-002
iteration =  9, rel_error = 1.102488e-003

X =

   11.3908   -8.6726
   -8.6726   11.3908

iteration = 10, error = 5.871745e-003
iteration = 10, rel_error = 2.923370e-004

X =

   11.3990   -8.6807
   -8.6807   11.3990

iteration = 11, error = 1.433839e-003
iteration = 11, rel_error = 7.138663e-005

X =

   11.4012   -8.6829
   -8.6829   11.4012

iteration = 12, error = 3.243623e-004
```

```
iteration = 12, rel_error = 1.614905e-005

X =

   11.4017   -8.6835
   -8.6835   11.4017

iteration = 13, error = 6.832928e-005
iteration = 13, rel_error = 3.401915e-006

X =

   11.4019   -8.6836
   -8.6836   11.4019

iteration = 14, error = 1.346506e-005
iteration = 14, rel_error = 6.703859e-007

X =

   11.4019   -8.6836
   -8.6836   11.4019

iteration = 15, error = 2.492217e-006
iteration = 15, rel_error = 1.240802e-007

X =

   11.4019   -8.6836
   -8.6836   11.4019

iteration = 16, error = 4.348086e-007
iteration = 16, rel_error = 2.164784e-008

X =

   11.4019   -8.6836
   -8.6836   11.4019

iteration = 17, error = 7.173654e-008
iteration = 17, rel_error = 3.571552e-009
```

```
X =

   11.4019   -8.6836
   -8.6836   11.4019

iteration = 18, error = 1.122453e-008
iteration = 18, rel_error = 5.588367e-010

X =

   11.4019   -8.6836
   -8.6836   11.4019

iteration = 19, error = 1.670008e-009
iteration = 19, rel_error = 8.314481e-011

X =

   11.4019   -8.6836
   -8.6836   11.4019

iteration = 20, error = 2.368292e-010
iteration = 20, rel_error = 1.179103e-011

X =

   11.4019   -8.6836
   -8.6836   11.4019

iteration = 21, error = 3.208989e-011
iteration = 21, rel_error = 1.597661e-012

X =

   11.4019   -8.6836
   -8.6836   11.4019

iteration = 22, error = 4.169110e-012
iteration = 22, rel_error = 2.075677e-013
```

```
X =

   11.4019   -8.6836
   -8.6836   11.4019

iteration = 23, error = 5.275780e-013
iteration = 23, rel_error = 2.626656e-014

X =

   11.4019   -8.6836
   -8.6836   11.4019

iteration = 24, error = 7.283063e-014
iteration = 24, rel_error = 3.626024e-015

X =

   11.4019   -8.6836
   -8.6836   11.4019

iteration = 25, error = 1.953993e-014
iteration = 25, rel_error = 9.728356e-016

X =

   11.4019   -8.6836
   -8.6836   11.4019


ans =

   11.4019   -8.6836
   -8.6836   11.4019
```

One can see that the convergence is very slow. We have obtained an error of about 10^{-14} in 25 iterations.

Next, we apply `exp_matrix_series` to matrix

$$A2 = \begin{pmatrix} -49 & 24 \\ -64 & 31 \end{pmatrix}$$

```
>> A2=[-49 24; -64 31]

A2 =

   -49    24
   -64    31

>> exp_T_series(A2,1e-14)
iteration =  1, error = 9.342484e+001
iteration =  1, rel_error = 4.542887e+001

X =

   -48    24
   -64    32

iteration =  2, error = 7.700752e+002
iteration =  2, rel_error = 3.744576e+002

X =

  384.5000 -192.0000
  512.0000 -255.5000

iteration =  3, error = 4.141758e+003
iteration =  3, rel_error = 2.013976e+003

X =

  1.0e+003 *

   -2.0717    1.0360
   -2.7627    1.3817

iteration =  4, error = 1.673820e+004
iteration =  4, rel_error = 8.139135e+003
```

```
X =

  1.0e+004 *

    0.8368   -0.4184
    1.1157   -0.5578

iteration =  5, error = 5.425459e+004
iteration =  5, rel_error = 2.638190e+004

X =

  1.0e+004 *

   -2.7128    1.3564
   -3.6171    1.8086

iteration =  6, error = 1.468918e+005
iteration =  6, rel_error = 7.142777e+004

X =

  1.0e+004 *

    7.3445   -3.6722
    9.7926   -4.8963

iteration =  7, error = 3.416066e+005
iteration =  7, rel_error = 1.661100e+005

X =

  1.0e+005 *

   -1.7080    0.8540
   -2.2774    1.1387

iteration =  8, error = 6.964525e+005
iteration =  8, rel_error = 3.386578e+005
```

```
X =

  1.0e+005 *

    3.4823   -1.7411
    4.6430   -2.3215

iteration =  9, error = 1.264326e+006
iteration =  9, rel_error = 6.147925e+005

X =

  1.0e+005 *

   -6.3216    3.1608
   -8.4289    4.2144

iteration = 10, error = 2.068997e+006
iteration = 10, rel_error = 1.006073e+006

X =

  1.0e+006 *

    1.0345   -0.5172
    1.3793   -0.6897

iteration = 11, error = 3.082502e+006
iteration = 11, rel_error = 1.498901e+006

X =

  1.0e+006 *

   -1.5413    0.7706
   -2.0550    1.0275

iteration = 12, error = 4.215456e+006
iteration = 12, rel_error = 2.049812e+006

X =
```

```
  1.0e+006 *

    2.1077   -1.0539
    2.8103   -1.4052

iteration = 13, error = 5.328028e+006
iteration = 13, rel_error = 2.590813e+006

X =

  1.0e+006 *

   -2.6640    1.3320
   -3.5520    1.7760

iteration = 14, error = 6.260488e+006
iteration = 14, rel_error = 3.044232e+006

X =

  1.0e+006 *

    3.1302   -1.5651
    4.1737   -2.0868

iteration = 15, error = 6.873163e+006
iteration = 15, rel_error = 3.342152e+006

X =

  1.0e+006 *

   -3.4366    1.7183
   -4.5821    2.2911

iteration = 16, error = 7.081341e+006
iteration = 16, rel_error = 3.443381e+006

X =
```

```
  1.0e+006 *

    3.5407   -1.7703
    4.7209   -2.3604

iteration = 17, error = 6.873163e+006
iteration = 17, rel_error = 3.342152e+006

X =

  1.0e+006 *

   -3.4366    1.7183
   -4.5821    2.2911

iteration = 18, error = 6.306091e+006
iteration = 18, rel_error = 3.066406e+006

X =

  1.0e+006 *

    3.1530   -1.5765
    4.2041   -2.1020

iteration = 19, error = 5.485873e+006
iteration = 19, rel_error = 2.667567e+006

X =

  1.0e+006 *

   -2.7429    1.3715
   -3.6573    1.8286

iteration = 20, error = 4.537296e+006
iteration = 20, rel_error = 2.206311e+006

X =

  1.0e+006 *
```

```
    2.2686   -1.1343
    3.0249   -1.5124

iterations 21 to 56 are omitted.


iteration = 57, error = 4.679132e-007
iteration = 57, rel_error = 2.275280e-007

X =

   -0.7358    0.5518
   -1.4715    1.1036

iteration = 58, error = 1.250355e-007
iteration = 58, rel_error = 6.079988e-008

X =

   -0.7358    0.5518
   -1.4715    1.1036

iteration = 59, error = 4.581416e-008
iteration = 59, rel_error = 2.227764e-008

X =

   -0.7358    0.5518
   -1.4715    1.1036

iteration = 60, error = 1.048504e-008
iteration = 60, rel_error = 5.098467e-009

X =

   -0.7358    0.5518
   -1.4715    1.1036

iteration = 61, error = 1.089735e-008
iteration = 61, rel_error = 5.298959e-009
```

```
X =

   -0.7358    0.5518
   -1.4715    1.1036

iteration = 62, error = 7.198322e-009
iteration = 62, rel_error = 3.500264e-009

X =

   -0.7358    0.5518
   -1.4715    1.1036

iteration = 63, error = 8.196473e-009
iteration = 63, rel_error = 3.985626e-009

X =

   -0.7358    0.5518
   -1.4715    1.1036

iteration = 64, error = 7.931339e-009
iteration = 64, rel_error = 3.856702e-009

X =

   -0.7358    0.5518
   -1.4715    1.1036

iteration = 65, error = 8.000682e-009
iteration = 65, rel_error = 3.890420e-009

X =

   -0.7358    0.5518
   -1.4715    1.1036

iteration = 66, error = 7.982821e-009
iteration = 66, rel_error = 3.881735e-009
```

```
X =

   -0.7358    0.5518
   -1.4715    1.1036

iteration = 67, error = 7.987353e-009
iteration = 67, rel_error = 3.883939e-009

X =

   -0.7358    0.5518
   -1.4715    1.1036

iteration = 68, error = 7.986220e-009
iteration = 68, rel_error = 3.883388e-009

X =

   -0.7358    0.5518
   -1.4715    1.1036

iteration = 69, error = 7.986499e-009
iteration = 69, rel_error = 3.883524e-009

X =

   -0.7358    0.5518
   -1.4715    1.1036

iteration = 70, error = 7.986431e-009
iteration = 70, rel_error = 3.883491e-009

X =

   -0.7358    0.5518
   -1.4715    1.1036

iteration = 71, error = 7.986447e-009
iteration = 71, rel_error = 3.883498e-009

X =
```

```
   -0.7358    0.5518
   -1.4715    1.1036


ans =

   -0.7358    0.5518
   -1.4715    1.1036
```

We can see that the error is growing at first. In fact the error only starts to decrease after 16 iterations. After that point it decreases steadily until iterations 62 where the relative error is $3.500264e-009$. After this point the error stagnates, and we do not get close to the ideal relative error of 10^{-16}. The reason for this is that the signs of the elements of X^k alternates, and so there is considerable cancellation.

The series method is unstable because of cancellation error. Here both $A1$ and $A2$ have negative entries. The $(A1)^k$ has sign pattern

$$\begin{pmatrix} + & - \\ - & + \end{pmatrix}$$

for all k. Thus there is no cancellation error in adding the individual components of $(A1)^k$ in the series. On the other hand $(A2)^k$ has sign pattern

$$\begin{pmatrix} - & + \\ - & + \end{pmatrix}$$

for all k odd, and

$$\begin{pmatrix} + & - \\ + & - \end{pmatrix}$$

for k even. Thus when adding powers, there is cancellation in every component of the sum and explain why the series method results are so much worse for $A2$ than for $A1$.

5.1.2 Padé Approximation

The Taylor series is the simplest algorithm for the exponential of programme. It requires just one loop, containing one matrix multiplication, one scalar matrix multiplication and one matrix-matrix addition.

But we have discussed that this approach can be inefficient and not accurate. The purpose of a Taylor series is to expand a function as a power series, whereas a Padé approximation expands a function as the ratio of two polynomials.

Since the Padé approximation is defined as a rational function and is expressed as ratio of polynomials, it can be calculated numerically easily. A Padé approximation approximates a function in only one variables, an approximation of a function in two variables is called a *Chisholm approximation* and in multiple variables is called a *Canterbury approximation.*

Assume that e^x is to be approximated by

$$\frac{1+p_1x}{1+q_1x}.$$

Finding p_1 and q_1 requires two equations, which will come from the coefficients of x and x^2, so the leading error term will be x^3, hence

$$e^x = \frac{1+p_1x}{1+q_1x} + \alpha_3x^3 + \alpha_4x^4 + ...$$

$$(1+q_1x)(1+x+\frac{1}{2}x^2+\frac{1}{6}x^3+...) = 1+p_1x+(1+q_1x)(\alpha_3x^3+\alpha_4x^4+...)$$

Hence

$$(1+q_1-p_1)x+(\frac{1}{2}+q_1)x^2+(\frac{1}{6}+\frac{1}{2}q_1-\alpha_3)x^3+higher\ order\ terms = 0$$

This is satisfied uniquely to terms of order three by

$$p_1 = \frac{1}{2},\ q_1 = -\frac{1}{2}\ and\ \alpha_3 = -\frac{1}{13}$$

The rational approximation

$$\frac{(1+\frac{1}{2}x)}{(1-\frac{1}{2}x)}$$

is called the $(1,1)$ Padé approximation of order 2 to $exp(x)$ and has a leading error term of order 3.

In general, it is possible to approximate e^x by

$$e^x = r_{mn} + \alpha_{m+n+1}x^{m+n+1} + O(x^{m+n+2})$$

where

$$r_{mn}(x) = \frac{p_{mn}(x)}{q_{mn}(x)}$$

$$p_{mn}(x) = \sum_{i=0}^{m} p_i x^i = p_0 + p_1 x + p_2 x^2 + ... + p_m x^m$$

$$q_{mn}(x) = \sum_{i=0}^{n} q_i x^i = q_0 + q_1 x + q_2 x^2 + ... + q_n x^n$$

$$r_{mn}(x) = \frac{p_0 + p_1 x + p_2 x^2 + ... + p_m x^m}{1 + q_1 x + q_2 x^2 + ... + q_n x^n} \tag{5.3}$$

We can assume $q_0 = 1$, because, $q_0 \neq 0$ in order to have a nonzero denominator. The two polynomials $p_{mn}(x)$ and $q_{mn}(x)$ are constructed in such a way that $f(x)$ and $r_{mn}(x)$ agree at $x = 0$ and also their derivatives up to $m+n$. In the case where $q_0(x) = 1$, the approximation is just the Maclaurin expansion for $f(x)$. It can be shown that for a given value of $m + n$, the error is smallest when $p_{mn}(x)$, $q_{mn}(x)$ have the same degree or the degree of $p_{mn}(x)$ is one higher than $q_{mn}(x)$ for the value of $m + n$[7]. The rational function $r_{mn}(x)$ has $m + n + 1$ coefficients. We need to find these coefficients p_i and q_i such that the derivatives of the function are approximated as

$$f^{(k)}(0) = r^{(k)}(0),\ k = 0, 1, 2, ..., m + n$$

The error of the approximation is

$$f(x) - r(x) = f(x) - \frac{p(x)}{q(x)} = \frac{f(x)q(x) - p(x)}{q(x)}$$

Now, let us replace the function $f(x) = \sum_{k=0}^{\infty} c_k x^k$ and the approximated rational function into the error formula gives

$$f(x) - r_{mn}(x) = \sum_{k=0}^{\infty} c_k x^k - \frac{\sum_{i=0}^{m} p_i x^i}{\sum_{i=0}^{n} q_i x^i}$$

$$= \frac{\sum_{k=0}^{\infty} c_k x^k \sum_{i=0}^{n} q_i x^i - \sum_{i=0}^{m} p_i x^i}{q_{mn}(x)}$$

By expanding the sums and taking the advantage of the assumed value of $q_0 = 1$ and reordering produces the coefficient of the x^k, the numerator term as

$$\sum_{i=0}^{k} (c_i q_{k-i}) - p_k.$$

We select the coefficients such that this expression is zero for $k \leqslant m+n$. This assures that $f(x) - r_{mn}(x)$ has a zero of multiplicity $m+n+1$ at $x=0$. This results in a set of $m+n+1$ linear equations of the form

$$\sum_{i=0}^{k} c_i q_{k-i} - p_k = 0,\ k = 0, 1, ..., m+n$$

This is a homogeneous system of linear equations in $m+n+1$ unknowns p_k, $k = 0, 1, ..., m$ and q_k, $k = 1, 2, ..., n$. The Padé approximant $r_{mn}(x)$ to the exponential function $f(x) = e^x$ is given by

$$p_{mn}(x) = \sum_{j=0}^{m} \frac{m!(m+n-j)!}{(m+n)!(m-j)!} \frac{x^j}{j!}$$

$$q_{mn}(x) = \sum_{j=0}^{n} \frac{n!(m+n-j)!}{(m+n)!(n-j)!} \frac{(-x)^j}{j!}$$

Moreover,

$$e^x - r_{mn}(x) = (-1)^m \frac{m!n!}{(m+n)!(m+n+1)!} x^{m+n+1} + O(x^{m+n+2}) \quad (5.4)$$

The following table gives the first eight Padé approximants to $exp(x)$ and their leading error terms.

Table 4.

Padé Approximant to e^x		
(m,n)	$r_{mn}(x)$	*error term*
(1,0)	$1+x$	$\frac{1}{2}x^2$
(2,0)	$1+x+\frac{1}{2}x^2$	$\frac{1}{6}x^3$
(0,1)	$\frac{1}{1-x}$	$-\frac{1}{2}x^2$
(1,1)	$\frac{1+\frac{1}{2}x}{1-\frac{1}{2}x}$	$-\frac{1}{12}x^3$
(2,1)	$\frac{1+\frac{2}{3}x+\frac{1}{6}x^2}{1-\frac{1}{3}x}$	$-\frac{1}{72}x^4$
(0,2)	$\frac{1}{1-x+\frac{1}{2}x^2}$	$\frac{1}{6}x^3$
(1,2)	$\frac{1+\frac{1}{3}x}{1-\frac{2}{3}x+\frac{1}{6}x^2}$	$\frac{1}{72}x^4$
(2,2)	$\frac{1+\frac{1}{2}x+\frac{1}{12}x^2}{1-\frac{1}{2}x+\frac{1}{12}x^2}$	$\frac{1}{720}x^5$

Let us consider the following enlightening example.

Example 35. *Consider the approximation of*

$$f(x) = e^x$$

The Maclaurin series expansion is

$$e^x = 1 + x + \frac{1}{2}x^2 + \frac{1}{6}x^3 + O(x^4).$$

Consider the Padé approximation with $m = 2$ and $n = 1$.
Following the above theory we set up the equation

$$p_m(x) = p_2(x) = p_0 + p_1 x + p_2 x^2$$

$$q_n(x) = q_1(x) = 1 + q_1 x.$$

Form

$$f(x)q_n(x) - p_m(x) = 0$$

$$(1 + x + \frac{1}{2}x^2 + \frac{1}{6}x^3 + ...)(1 + q_1 x) - (p_0 + p_1 x + p_2 x^2) = 0.$$

$$1 - p_0 + (1 + q_1 - p_1)x + (\frac{1}{2} + q_1 - p_2)x^2 + (\frac{1}{6} + \frac{1}{2}q_1)x^3 = 0$$

since $m + n = 2 + 1 = 3$, we need four equations.

$$1 - p_0 = 0$$

$$1 + q_1 - p_1 = 0$$

$$\frac{1}{2} + q_1 - p_2 = 0$$

and

$$\frac{1}{6} + \frac{1}{2}q_1 = 0.$$

Then we have the values of the coefficients are

$$p_0 = 1,\ p_1 = \frac{2}{3},\ p_2 = \frac{1}{6},\ q_1 = -\frac{1}{3}$$

The Padé approximation of e^x is therefore of the form

$$r_{21}(x) = \frac{1 + \frac{2}{3}x + \frac{1}{6}x^2}{1 - \frac{1}{3}x}$$

Let us compare the accuracy of the Maclaurin series approximation with that of the Padé approximation on the interval $[0, 1]$. *Using Matlab we find*

$$max\{|e^x - (1 + x + \frac{x^2}{2} + \frac{x^3}{6})| : x \in [0, 1]\} = 0.05$$

and

$$max\{|e^x - r_{21}| : x \in [0, 1]\} = 0.03$$

Thus the Padé approximation is slightly more accurate. But more importantly, the highest power of x *in the Padé approximation is* x^2, *but in the Maclaurin approximation is* x^3. *Where* x *is scalar.*

The Padé approximation of matrix exponential can be extended in the same way if x is a matrix $A \in \mathbb{M}_n$. A Taylor series approach to matrix exponential approximation is generally slow and inaccurate. Our goal is to make the maximum error as small as possible. The Padé approximation to the matrix exponential can be defined as

$$e^A \approx r_{mn}(A) = \frac{p_{mn}(A)}{q_{mn}(A)} \tag{5.5}$$

$$= [q_{mn}(A)]^{-1} p_{mn}(A)$$

where

$$p_{mn}(A) = \sum_{j=0}^{m} \frac{(m+n-j)!m!}{(m+n)!j!(m-j)!} A^j \tag{5.6}$$

$$q_{mn}(A) = \sum_{j=0}^{n} \frac{(m+n-j)!n!}{(m+n)!j!(n-j)!} (-A)^j \tag{5.7}$$

In the case when $n = 0$, the approximation will take the form of Taylor(Maclaurin) expansion for $f(A)$. There are $m + 1$ unknown coefficient in $p_{mn}(A)$, and n unknown coefficients in $q_{mn}(A)$, hence the rational function $r_{mn}(A)$ has $m + n + 1$ unknown coefficients.

From [7] it was seen that the diagonal approximation ($m = n$) are preferred over the off diagonal approximation ($m \neq n$) for stability and economy of computation. The diagonal Padé approximation have some advantages over the Taylor series i.e., the Padé approximation is

$$r_{22}(A) = [q_{22}(A)]^{-1} p_{22}(A) \tag{5.8}$$

where p, q are as in (5.6) and (5.7). Notice that since matrix multiplications for $k \times k$ matrix is $O(k^3)$ flops and matrix addition is $O(k^2)$ flops, most of the work is evaluating equation (5.8) is in computing the powers of the matrix A.

Let us take $m = n$. Then the error in r_{mn} is $O(x^{2m+1})$. To evaluate $r_{mn}(A)$ for $A \in M_k$, requires that we compute $A, , A^2, ..., A^m$ and $p(A)$, $q(A)$. This requires $2k^3$ flops. To obtain an error of $O(x^{2m+1})$ using the Taylor series would require the computation of $A, A^2, ..., A^{2m}$. That is $(2m-1)2k^3$ flops, which is almost twice as much of for the Padé method.

Since most of work in computing the Padé and power series approximation is in computing the powers of A, for a given amount of work. The Padé method has twice the order of the power series method. The Padé is only accurate near the origin so that the approximation of e^A is not valid, when $\|A\|$ is too large. Like Taylor series there is a question that where the series in Padé approximation to terminate, what are the appropriate values of m. In this case, suppose Padé with $m = n$ i.e., the diagonal approximation. Then error is $o(\|A\|^{2m+1})$ we need same amount of work for $(m-1)$ matrix multiplication $p_{mn}(A)$, $q_{mn}(A)$. To compute $r_{mn}(A)$, $2n^3$ flops are required, same as 1 matrix multiplication, so the total work need m matrix multiplication. Now consider the Taylor series with m matrix multiplication. Then error is $o(\|A\|^{m+2})$. So if $m \geqslant 2$ Padé gives a smaller error.

The Taylor series and the Padé approximation are applicable to certain cases, but they require a lot of computation when $\|A\|$ is large. This problem will be solved when we introduce the so called Scaling and Squaring method.

Here is the m-file for the Padé approimation:

```
function [r] = diag_pade(A, m)
% diag pade est of exp using formula
% from higham
%
p = zeros(size(A));
q = p;
for j=0:m,
    % this is inefficient and could overflow
    % but is presented in this form to agree with theory
    coef = factorial(2*m-j)*factorial(m);
```

```
    coef = coef/(factorial(2*m)*factorial(j)*factorial(m-j));
    p = p+ coef*(A)^j;
    q = q+ coef*(-A)^j;
end
r=inv(q)*p;
```

See section (5.1.3) for an example.

5.1.3 Scaling and Squaring

The main problem, i.e., roundoff error difficulties and the computing costs of the Taylor Series and Padé approximation is that the accuracy decreases and efficiency increases as $\|A\|$ increases. To overcome these difficulties we use a fundamental property:

$$(e^{a/b})^b = e^a$$

where a and b are scalars unique to any exponential function.

This property can be applied to matrices:

$$e^A = (e^{A/m})^m$$

where $A \in \mathbb{M}_n$ and m is a positive integer. This idea will help to control the roundoff error and more importantly the computing costs it would take to find either by Taylor series or Padé approximation. The advantage of the scaling methods is that the scaled transition matrix can be made to have a norm less than unity, so by reducing the norm of the matrix. Scaling and Squaring improves the efficiency of the Taylor series and Padé methods there is one commonly used criteria of choosing m is to make it the smallest power of two, $m = 2^j$, such that $\|A\|/m \leqslant 1$. If m is too big then lose accuracy.(For example, taking $m = 10^{-17}$ in matlab we have $(e^{10^{-17}})^{(10)^{-17}} = 1$ rather that 2.7183). If too small, series takes too long to converge. So, with this restriction, approximate $e^{A/m} = e^{A/2^j} \approx r(A/2^j)^{2^j}$, where r is either Taylor or Padé approximant to the exponential and then take $e^A \approx r(A/2^j)^{2^j}$, where to form the matrix $(e^{A/m})^m$ by j repeated squarings.

Now we define the scaled matrix exponential or base matrix

$$M = e^{A/m}$$

hence

$$e^A = M^m.$$

Using the Taylor series definition of the exponential function, we have

$$M = I + A/m + \frac{(A/m)^2}{2!} + ... + \frac{(A/m)^i}{i!} + ...$$

Again, the base matrix must be approximated by the truncated series. The degree of Taylor approximation to the base matrix is denoted by M_k. We can also use the Padé approximation M_{mn} for the determination of the base matrix and for the diagonal Padé approximation where $m = n$ it is denoted by M_{mm}.

Since M^m depends on the integer m, as we said above that the most effective method is to choose $m = 2^j$; we can square the base matrix j times instead of multiplying $m = 2^j times$.

$$e^A = \underbrace{[[[[M^2]^2]^2]\ldots]^2}_{m\,times}$$

In this manner we need only m times matrix multiplications are necessary instead of the k matrix multiplications necessary for direct calculation. But we need to be careful not to scale too much. For instance we try to approximate e^1 by $(1 + \frac{1}{n})^n$, for $n = 10^{10}$ we get an accuracy of 2×10^{-7}, but for $n = 12$, we get an accuracy of 2×10^{-4}.

Let us look at the error introduced in the squaring phase. To compute e^x we can compute $z = fl(e^{x/2^j})$ and $\underbrace{[[[[M^2]^2]^2]\ldots]^2}_{j\,times}$

Consider

$$z = fl(e^{x/2^j})$$

Then

$$z = e^{x/2^j}(1 + \epsilon) \quad where\ |\epsilon| \leqslant \epsilon_M$$

Even if make no more rounding errors, we will get

$$z^{2^j} = e^{x/2^j}(1 + \epsilon)^{2^j} = e^x(1 + \epsilon)^{2^j}$$

$$\approx e^x(1 + 2^j\epsilon)$$

so

$$relerr = |\frac{(fl(e^{x/2^j}))^{2^j} - e^x}{e^x}| \approx |\frac{2^j e^x \epsilon}{e^x}| \leqslant 2^j \epsilon_M$$

For example if $A = [10] \in \mathbb{M}_n$ then we want to compute e^{A_1} to accuracy 10^{-16} after scaling $A = \frac{A}{32} = \frac{10}{32}$

$$\sum_{k=m+1}^{\infty} \frac{A_1}{(m+1)!} \leqslant 10^{-16}$$

$$\Leftrightarrow \ \frac{A_1^{m+1}}{(m+1)!} \leqslant 10^{-16}$$

need $m+1 = 13 \ \Rightarrow \ m = 12$ i.e., 11 matrix multiplication. In general, the squaring method is more efficient.

Here is the m-file for Scaling and squaring of Taylor's method:

```
function [e] = exp_taylor(A, tol, s)
%
% compute exp using taylor series with given tolerance
%                     and s scalings and squarings
%
n = max(size(A));
A0 = A;
for i=1:s,
    A = A/2;
end
%
e = exp_matrix_series(A,tol);
%
% square
for i=1:s,
    e = e*e;
end
```

Here is the matlab implementation code to the scaling and squaring of Padé method:

```
function [e] = exp_pade_sc_sq(A, m, s)
%
% compute exp usign taylor series with given tolerance
%                     and s scalings and squarings
%
```

```
n = max(size(A));
A0 = A;
for i=1:s,
    A = A/2;
end
%
e = diag_pade(A, m);
%
% square
for i=1:s,
    e = e*e;
end
```

Consider the following test matrix:

$$A2 = \begin{pmatrix} -49 & 24 \\ -64 & 31 \end{pmatrix}$$

```
>> A2 = [-49 24; -64 31]

A2 =

   -49    24
   -64    31

>> norm(A2)/2^8

ans =

    0.3501

>> et = exp_taylor_sc_sq(A2,10^-14, 8)
iteration =  1, error = 1.288799e-002
iteration =  1, rel_error = 1.121899e-002

i =

     1
```

```
ans =

    0.8086    0.0938
   -0.2500    1.1211

iteration =  2, error = 2.879704e-004
iteration =  2, rel_error = 2.506781e-004

i =

     2


ans =

    0.8152    0.0905
   -0.2412    1.1167

iteration =  3, error = 4.797634e-006
iteration =  3, rel_error = 4.176339e-006

i =

     3


ans =

    0.8150    0.0905
   -0.2414    1.1168

iteration =  4, error = 6.385968e-008
iteration =  4, rel_error = 5.558982e-008

i =

     4


ans =
```

```
    0.8150    0.0905
   -0.2414    1.1168

iteration =  5, error = 7.078917e-010
iteration =  5, rel_error = 6.162195e-010

i =

     5


ans =

    0.8150    0.0905
   -0.2414    1.1168

iteration =  6, error = 6.723871e-012
iteration =  6, rel_error = 5.853127e-012

i =

     6


ans =

    0.8150    0.0905
   -0.2414    1.1168

iteration =  7, error = 5.540013e-014
iteration =  7, rel_error = 4.822579e-014

i =

     7


ans =

    0.8150    0.0905
   -0.2414    1.1168
```

```
iteration =  8, error = 7.771561e-016
iteration =  8, rel_error = 6.765141e-016

i =

     8


ans =

    0.8150    0.0905
   -0.2414    1.1168

iteration =  9, error = 2.775558e-016
iteration =  9, rel_error = 2.416122e-016

i =

     9


ans =

    0.8150    0.0905
   -0.2414    1.1168


et =

   -0.7358    0.5518
   -1.4715    1.1036

>> (norm(et-expm(A2)))

ans =

  7.7242e-013

>> ep = exp_pade_sc_sq(A2,5, 8)
```

```
ep =

   -0.7358    0.5518
   -1.4715    1.1036

>> (norm(ep-expm(A2)))

ans =

  2.2308e-013
```

Notice that with $m = 5$ for the Padé method we obtain similar accuracy as with $i = 9$ powers of$A$ for the Taylor series method. This agrees with our theoretical observation in the discussion following equation (5.8).

5.2 Matrix Decomposition Methods

As we know the matrix decomposition methods are based on the similarity transformation of a matrix as

$$A = SBS^{-1}.$$

From the definition of e^{tA} implies that

$$e^{tA} = Se^{tB}S^{-1}$$

but if S is close to singular it means $\kappa(S)$ is large.

5.2.1 Eigenvalue-eigenvector method

Since $A \in \mathbb{C}^{n\times n}$ is a real symmetric matrix and $\exists$ a real unitary V and $\Lambda = diag(\lambda_i)$, with λ_i the eigenvalues of A which are real, such that

$$A = V\Lambda V^* \tag{5.9}$$

Then by JCF definition e^A we have

$$e^A = Ve^{\Lambda}V^* \tag{5.10}$$

For $\Lambda = diag(\lambda_1, ..., \lambda_n)$, the $e^{\Lambda} = diag(e^{\lambda_1}, e^{\lambda_2}, ..., e^{\lambda_n})$ is trivial to compute.

But, a related difficulty with this approach occurs when A may not be diagonalizable and thus is defective, because there is no invertible matrix of eigenvectors V. These observations serve to highlight the difficulties associated with ill-conditioned similarity transformations [5]. It can be shown that let $X\,Y \in \mathbb{M}_n$ then

$$fl(XY) = XY + E$$

where

$$|E| \leqslant 2n\epsilon_M|X||Y|, \;\; hence \, \|E\|_1 \leqslant 2n\epsilon_M\|X\|_1\|Y\|_1$$

Thus

$$\begin{aligned} fl(VDV^{-1}) &= Vfl(DV^{-1}) + E_1 \\ &= VDV^{-1} + VE_2 + E_1 \\ \|fl(VDV^{-1}) - VDV^{-1}\|_1 &= \|VE_2 + E_1\|_1 \\ &\leqslant \|VE_2\|_1 + \|E_1\|_1 \\ &\leqslant \|V\|_1\|E_2\|_1 + \|E_1\|_1, \end{aligned}$$

where

$$\begin{aligned} \|E_1\|_1 &\leqslant 2n\epsilon_M\|V\|_1 \, \|DV^{-1}\|_1 \\ &\leqslant 2n\epsilon_M\|V\|_1\|D\|_1\|V^{-1}\|_1 \\ \|E_2\|_1 &\leqslant 2n\epsilon_M\|D\|_1\|V^{-1}\|_1 \\ \|fl(VDV^{-1}) - VDV^{-1}\|_1 &\leqslant 2n\epsilon_M\|V\|_1\|D\|_1\|V^{-1}\|_1 + 2n\epsilon_M\|V\|_1\|D\|_1\|V^{-1}\|_1 \\ &\leqslant 4n\epsilon_M\|D\|_1\|V\|_1\|V^{-1}\|_1 \\ &\leqslant 4n\epsilon_M\|D\|_1\kappa(V) \\ \|fl(VDV^{-1}) - VDV^{-1}\|_1 &\leqslant 4n\epsilon_M\|D\|_1\kappa(V) \end{aligned} \tag{5.11}$$

If V is ill-condition the error will be large. even if we do not compute $e^{VDV^{-1}} = Ve^DV^{-1}$, just computing $fl(VDV^{-1})$ gives the equation (5.11) error. This method relies on diagonalizating the matrix. Thus if for example a defective matrix is

$$A = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}.$$

Which is not diagonalizable. Thus we can not use the method.

However, the following example suggests that ill-conditioned similarity transformations should be avoided when computing a function of a matrix.

Example 36. *Let*

$$A = \begin{pmatrix} 1+\alpha & 1 \\ 0 & 1-\alpha \end{pmatrix}$$

then

$$V = \begin{pmatrix} 1 & -1 \\ 0 & 2\alpha \end{pmatrix}$$

$$D = \begin{pmatrix} 1+\alpha & 0 \\ 0 & 1-\alpha \end{pmatrix}$$

and

$$cond(V) = O(\frac{1}{\alpha}).$$

If $\alpha = 10^{-13}$ *then*

$$\kappa(V) = 1.000244225956801e + 013$$

In Matlab:

```
>> A=[1+10^-13 1; 0 1-10^-13]

A =

   1.00000000000010   1.00000000000000
                  0   0.99999999999990

>> [V,D]=eig(A)

V =

   1.00000000000000  -1.00000000000000
                  0   0.00000000000020


D =

   1.00000000000010                   0
                  0   0.99999999999990

>> eA=V*expm(D)*inv(V)

eA =
```

```
   2.71828182845932   2.69921875000000
                   0   2.71828182845878

>> expm(A)

ans =

   2.71828182845931   2.71828182845905
                   0   2.71828182845878

>> relerr=norm(eA-expm(A))/norm(expm(A))

relerr =

   0.00433421961422=4.33421961422e-3
```

Note: $10^{-3} \approx relerr \approx \epsilon_M \times \kappa(V) \approx 10^{-16} \times 10^{13} \approx 10^{-3}$

Thus the error is large because V is ill-conditioned.

Here is the matlab code based on the eigenpair decomposition:

```
function f = exp_eigenpair(A);

% usage: f = exp_eigenpair(A);
% calculate the matrix exponential of A using
% eigenpair decomposition.
% Input:
% A = matrix
% Output:
% f = approximate value of matrix exponential
  [n,n]=size(A);
  [U,D]=eig(A);
  for i=1:n
      D(i,i)=exp(D(i,i));
  end
  f=U*D*inv(U);
  err = norm(expm(A)-f,inf);
  rel_err = err/norm(expm(A))
```

```
  display([sprintf('error=%e\n',err)]);
end
```

Let us consider the example to use the above matlab code:

```
>> A1=[2 -1; -1 2]

A1 =

     2    -1
    -1     2

>> exp_eigenpair(A1)

rel_err =

  2.6532e-016

error=5.329071e-015


ans =

   11.4019   -8.6836
   -8.6836   11.4019

>> A2=[-49 24; -64 31]

A2 =

   -49    24
   -64    31

>> exp_eigenpair(A2)

rel_err =

  1.0948e-013

error=2.251532e-013
```

```
ans =

   -0.7358    0.5518
   -1.4715    1.1036
```

5.2.2 Schur Parlett Method

We recall the Schur decomposition

$$A = UTU^t$$

with unitary U and upper-triangular T exists if A is real and has real eigenvalues. If A has complex eigenvalues, then it is necessary to allow 2×2 blocks on the diagonal of T or to make U and T complex (and replace U^t with U^*). Once matrix A has been decomposed, the matrix exponential is evaluated from

$$e^A = Ue^T U^* \tag{5.12}$$

Where T is a triangular or quasitriangular matrix. The computation of the matrix exponential e^T of an upper triangular matrix is performed by using an algorithm which was developed by Parlett [14, 15]. If T is upper triangular matrix with $\lambda_1, ..., \lambda_n$ on the diagonal, then e^T is upper triangular with $e^{\lambda_1}, ..., e^{\lambda_n}$ on the diagonal. Note that there is no need of eigenvectors of A. Again, when the eigenvalues are distinct almost equal, inaccuracy takes place in the computation of e^T. Let us consider the following examples.

In the case, 2×2 and when $\lambda_1 \neq \lambda_2$, we have

$$T = \begin{pmatrix} \lambda_1 & t_{12} \\ 0 & \lambda_2 \end{pmatrix}$$

The exponential of this matrix is

$$e^T = \begin{pmatrix} e^{\lambda_1} & t_{12}\frac{e^{\lambda_2} - e^{\lambda_1}}{\lambda_2 - \lambda_1} \\ 0 & e^{\lambda_2} \end{pmatrix}$$

When $\lambda_1 = \lambda_2 = \lambda$

$$e^T = \begin{pmatrix} e^\lambda & t_{12} \\ 0 & e^\lambda \end{pmatrix}$$

For $n = 3$, case for 3×3 upper triangular matrix T.

$$T = \begin{pmatrix} \lambda_1 & t_{12} & t_{13} \\ 0 & \lambda_2 & t_{23} \\ 0 & 0 & \lambda_3 \end{pmatrix}$$

Then the exponential of this matrix is

$$e^T = \begin{pmatrix} e^{\lambda_1} & t_{12}\alpha_{12} & t_{13}\alpha_{13} + t_{12}t_{23}\alpha_{123} \\ 0 & e^{\lambda_2} & t_{23}\alpha_{23} \\ 0 & 0 & e^{\lambda_3} \end{pmatrix}$$

where

$$\alpha_{ij} = \frac{e^{\lambda_i} - e^{\lambda_i}}{\lambda_i - \lambda_j} \; for\, i < j$$

and

$$\alpha_{123} = \frac{\alpha_{12} - \alpha_{23}}{\lambda_1 - \lambda_2}$$

Hence, from these examples we observed that e^T is an upper triangular matrix whose entries involve divided differences of e^{λ_i}.

Here is the m-file for the Schur decomposition method:

```
function [f] = parlett_exp(t)
%
% assume t is upper tri and has distinct diag elts
%
n=max(size(t))
% assign diag of f
for i=1:n
    f(i,i) = exp(t(i,i));
end
%
for d=2:n,
    for i=1:n-d+1
         j=i+d-1;
          f(i,j) = (t(i,j)*(f(i,i)-f(j,j)))/(t(i,i)-t(j,j));
         for k=i+1:j-1
```

```
  f(i,j) = f(i,j) + (f(i,k)*t(k,j)-t(i,k)*f(k,j))/(t(i,i)-t(j,j));
        end
    end
end
```

Let us consider the following examples to find the exponential of a matrix using Schur Decomposition for matrices.

Example 37. *$A = UTU^*$, where*

$$T = \begin{pmatrix} 1 & 1 \\ 0 & 1+10^{-5} \end{pmatrix} \quad \textit{and} \quad U = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

```
>> T=[1 1; 0 1+1e-5]

T =

   1.00000000000000   1.00000000000000
                   0   1.00001000000000

>> U = 1/sqrt(2)* [1 1; 1 -1]

U =

   0.70710678118655   0.70710678118655
   0.70710678118655  -0.70710678118655

>> A = U*T*U'

A =

   1.50000500000000  -0.50000500000000
   0.49999500000000   0.50000500000000

>> expmA=expm(A)

expmA =

   4.07744312989289  -1.35916130143385
   1.35913411847965   1.35914770997940
```

```
>> [U1 T1]=schur(A)

U1 =

   0.70711385222309  -0.70709971007930
   0.70709971007930   0.70711385222309


T1 =

   1.00001000000578  -1.00000000000000
                  0   0.99999999999422

>> t11 = T1(1,1); t22 = T1(2,2); t12 = T1(1,2);
>> f11 = exp(t11); f22=exp(t22); f12 = t12*(f11-f22)/(t11-t22);
>> F = [f11 f12 ; 0 f22]

F =

   2.71830901142895  -2.71829541993511
                  0   2.71828182844334

>> Eschur=U1*F*U1'

Eschur =

   4.07744312990370  -1.35916130144465
   1.35913411849046   1.35914770996859

>> norm(Eschur-expmA)

ans =

    2.162126034736025e-011

>> norm(Eschur-expmA)/norm(expmA)

ans =

    4.915828381315610e-012
```

The reason for the error is that $t_{11} \approx t_{22}$. If we redo the example with

$$T = \begin{pmatrix} 1 & 1 \\ 0 & 1 + 10^{-10} \end{pmatrix}$$

the error will be larger$\approx 10^{-5}$. So our approximation of the matrix exponential using Schur Decomposition shows that roundoff error is caused by nearly confluent eigenvalues λ_i.

Now we illustrate the behavior in the Block Parlett algorithm by considering the following example. Notice that the 1,1 and 3,3 entries of B are very close.

```
B =

    0.4751    0.3810    0.3077    0.2029    0.0289
         0    0.2282    0.3960    0.4677    0.1764
         0         0    0.4609    0.4585    0.4066
         0         0         0    0.2051    0.0049
         0         0         0         0    0.0694

>> B(3,3)=B(1,1)+1e-8;
>> norm(parlett_exp(B)-expm(B))/norm(expm(B))

ans =

  1.0814e-008

>> B(3,3)=B(1,1)+1e-12;
>> norm(parlett_exp(B)-expm(B))/norm(expm(B))

ans =

  6.7441e-005

>> v=[3 2]

v =

     3     2
```

```
>> norm(parlett_block_final(B, v)-expm(B))/norm(expm(B))

ans =

  3.0098e-015

>> v=[2 3];
>> norm(parlett_block_final(B, v)-expm(B))/norm(expm(B))

ans =

  1.6032e-004
```

This shows that as two eigenvalues of B get closer the error in the Parlett method increases. If we use the Block Parlett method with the close eigenvalues in the same block there is no loss of accuracy. This is the case when $v = [3, 2]$. The first block is 3×3 and contains the two close eigenvalues.

On the other hand, if we use block Parlett with $v = [2, 3]$. Then the first block is 2×2 and contains one of the close eigenvalues. The other is in the second block. In this case the error is very large similarly to the error in the non-block Parlett algorithm.

5.3 Cauchy's integral formula

Recall from chapter 3 section 3.3,

$$f(A) = \frac{1}{2\pi i} \int_\Gamma f(z)(zI - A)^{-1} dz, \tag{5.13}$$

where Γ is a closed contour lying in the region of analyticity of f and winding once around the spectrum $\sigma(A)$ in the counterclockwise direction. The simplest choice of the contour Γ is a circle with radius r centered at some point z_0, defined by

$$\Gamma = \{z(\theta) = z_0 + re^{i\theta} : 0 \leqslant \theta \leqslant 2\pi\}.$$

Then making the substitution $dz = z'(\theta)d\theta$, the Cauchy integral (5.13) of a function of the matrix A becomes

$$f(A) = \frac{1}{2\pi i} \int_0^{2\pi} f(z)(zI - A)^{-1} rie^{i\theta} d\theta,$$

$$f(A) = \frac{1}{2\pi}\int_0^{2\pi} (z(\theta) - z_0) f(z)(zI - A)^{-1} d\theta, \qquad (5.14)$$

The matlab programme below implements the mid-point rule to approximate this integral. We will see that it requires a large number of functions evaluations to obtain reasonable accuracy. Of course the mid point rule is one of the simplest numerical quadrature rules. We propose to study more sophisticated methods like those in [6] to make this method more efficient. One important aspect of the approximate the Cauchy Integral is that the matrix inversions can be computed in parallel.

Here is a matlab code to implement the Cauchy Integral Method:

```
function [ea] = exp_c(a, m)
%
s = zeros(size(a))
theta = 2*pi*i/m;
id = eye(size(a));
%
for j=1:m,
    z = exp((j+.5)*theta); % z at midpoint
    s = s + inv(z*id - a) * exp(z)*(exp((j+1)*theta)-exp(j*theta));
end
ea= s /(2*pi*i);
```

Example 38. *Let us consider the test example:*

```
>> a= triu(rand(5))

a =

    0.9501    0.7621    0.6154    0.4057    0.0579
         0    0.4565    0.7919    0.9355    0.3529
         0         0    0.9218    0.9169    0.8132
         0         0         0    0.4103    0.0099
         0         0         0         0    0.1389

>> norm(exp_cauchy_Int(a, 100)-expm(a))
```

```
ans =

    2.1561

>> norm(exp_cauchy_Int(a, 1000)-expm(a))

ans =

  1.0058e-005

>> norm(exp_cauchy_Int(a, 2000)-expm(a))

ans =

  2.5144e-006

>> norm(exp_cauchy_Int(a, 4000)-expm(a))

ans =

  6.2860e-007

>> norm(exp_cauchy_Int(a, 8000)-expm(a))

ans =

  1.5715e-007

>> A=a/2

A =

    0.0967    0.3489    0.2483    0.3301    0.3636
         0    0.1892    0.4499    0.1710    0.1546
         0         0    0.4108    0.1449    0.4192
         0         0         0    0.1706    0.2840
         0         0         0         0    0.1852

>> norm(exp_cauchy_Int(A, 1000)-expm(A))

ans =
```

```
  3.7095e-006

>> norm(exp_cauchy_Int(A, 2000)-expm(A))

ans =

  9.2738e-007
```

The convergence is slow. It takes 1000 matrix inversions to attain an accuracy of 10^{-5} in the first example. By contrast, using Padé with scaling and squaring we would obtain relative accuracy of about 10^{-14} using $m = 8$ for the Padé approximation, and 4 scaling and squaring. That is about 12 matrix multiplications to obtain much higher accuracy.

Notice when the number of integration nodes is doubled, the error is reduced by a factor of 4. Using this we extrapolate. Let $E1000$ and $E2000$ denote the approximate with 1000 and 2000 nodes. Then a better approximation is

$$extrapolate = (4E2000 - E1000)/3.$$

Evaluating this for the matrix A gives

```
>> E2000=exp_cauchy_Int(A, 2000);

>> E1000=exp_cauchy_Int(A, 1000);

>> extrap =(4*E2000-E1000)/3

>> norm(extrap-expm(A))

ans =

  5.1339e-013
```

Thus combing $E1000$ and $E2000$ that have errors about 10^{-5} and 10^{-6} gives us an approximation with much smaller error 5.1339×10^{-13}.

Using even more sophisticated ideas perhaps we can make the Cauchy Integral method competitive.

Chapter 6

Conclusion and Future work

6.1 Conclusion

There are many ways to define e^A and they lead to many algorithms to compute e^A. In this part must we deal with the obvious question: Which method is the best? To answer such a question is very risky, because, we do not know enough about the sensitivity of the original problem.

In summery, we have to look at the strengthen and weakness of each method:

1. **Taylor's Series**: This method always converges in theory. But we have seen that there can be large cancellation errors, and the convergence can be slow if $\|A\|$ is large. These problems can be avoided by careful Scaling and Squaring.
2. **Padé approximation**: This method does not always converges. But with suitable Scaling and Squaring it dose always converges.
3. **Schur-Parlett**:
   This is a finite algorithm in theory once Schur form is found, but does not work if eigenvalues are equal and shows inaccuracy if the eigenvalues are close.
4. **Eigenvalue-Eigenvector method**: This method is finite, once eigenvalues and eigenvectors are found. It does not work if A is not diagonalizable. It is inaccurate if A is close to a non-diagonalizable matrix or if the eigenvectors are ill-conditioned

5. **Cauchy Integral formula method**:
   This method has good parallel computation properties but needs eigenvalues to be inside a contour, and so we need to know something about eigenvalues. For larger spectral radii or more scattered eigenvalues the convergence will be slower. The efficiency depends also on a good quadrature rule and is helped by clustered eigenvalues.

We have seen that there is no uniformly best method for the computation matrix exponential. The choice of method depends on the application and the particular matrix. Here are two common situations where one would choose different methods. For example, if A has well separated eigenvalues and we need e^{At} for many t, then

$$A = UTU^*$$

$$e^{At} = Ue^{tT}U^*,$$

using Schur-Parlett is simple, and efficient. If however A has repeated eigenvalues or close eigenvalues the Schur-Parlett method does not work well. Series, Padé and Scaling-squaring always work. Here good method would be to use Scaling and Squaring with Padé approximation. Incidentally this is the default algorithm used by Matlab *expm*.

6.2 Future work

Our implementation of the Cauchy Integral method uses only the simplest contour a circle centered at the origin, and the simplest quadrature method, the mid point rule. We would like to investigate higher order quadrature rules and extrapolation methods as well as transformation of the problem as suggested in [6].

The reason for our interest in this apparently unpromising method is that the most of the work is in computing $(zI - A)^{-1}$ for different values of z. All these inversions can be carried out in parallel.

There is no good set of test matrices for the exponential. For example N.Hale, N.J. Higham and L. Trefethen in their paper [6] use 2×2 matrices. We intend to find a set of matrices that arise in practice whose exponential is necessary to compute. This will be useful to us in

comparing methods, and will be useful resource for other researchers also.

Finally we would like to investigate a new idea that has never been considered before. Namely using a dynamic stopping criterion based on $\|A^k\|_1$ rather that using the inequality $\|A^k\| \leqslant \|A\|^k$.

The idea is that the accuracy of the Taylor and Padé methods depends on $\|A^k\|_\infty$. The usual analysis replaces $\|A^k\|_\infty$ by the larger quantity $\|A\|_\infty^k$. This may result in computing more terms then necessary. We intend to investigate the Taylor and Padé methods, when we compute $\|A^k\|_\infty$, at a cost of $O(n^2)$ flops, once A^k is formed. The cost of computing an extra power of A is $O(n^3)$ while the cost of our check is $O(n^2)$. We will decide whether to terminate or not, based on $\|A^k\|_\infty$ rather than the usual $\|A\|_\infty^k$.

We hope to bring all these ideas together to produce improved algorithms for the matrix exponential.

Bibliography

[1] D. S. Bernstein, *Matrix Mathematics Theory*, Facts, And Formulas with application to Linear systems theory, Princeton University Press, 2005, xxxv+726 pp. ISBN 0-691-11802-7(acid-free paper).

[2] J. W. Demmel , *Applied Numerical Linear Algebra*, SIAM. Society for Industrial and Applied Mathematics, Philadelphia, (1997), xii+419 pp. ISBN 01-0-89871-389-7.

[3] J. H. Gallier, *Geometric Methods and Applications*, Springer, 2001. xxi+565 pp. ISBN 987-0-387-95044-0.

[4] F.R.Gantmacher, *The theory of Matrices*, AMS Chelsea Publshing, Vol.I,(1959,1960,1977), x+374 pp. ISBN 0-8218-1376-5(Vol.I).

[5] G.H. Golub and C.F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press 1996, xxvii+694 pp. ISBN 0-8018-5414-8.

[6] N. Hale, N. J. Higham, and L. N. Trefethen, *Computing A^α, Log(A), and Related Matrix Functions by Contour Integrals*, SIAM J. Numerical Analysis Vol. 46, No. 5, pp.2505-2523, June 11, 2008

[7] N. J. Higham, *Functions of Matrices Theory and Computation*, SIAM. Society for Industrial and Applied Mathematics, Philadelphia, PA. USA, 2008. ISBN 978-0-89871-646-7. xx+425 pp.

[8] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, 1985. xiii+561 pp. ISBN 0-521-30586-2.

[9] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, 1991. viii+607 pp. ISBN 978-0-521-46713-1.

[10] P. Lancaster and M. Tismenetsky, *The Theory of Matrices*, Second Edition With Application, Academic Press,(1984), xv+570 pp. ISBN 0-12-435560-9.

[11] A. J. Laub, *Matrix Analysis for Scientists and Engineers*, SIAM 2005, xiii+157 pp. ISBN 0-89871-576-8.

[12] M. L. Liou, *A novel method of evaluating transient response*, Proc. IEEE, 54 (1966), pp.20-23

[13] C. Moler and C. V. Loan, *Nineteen Dubious Ways to Compute the Exponential of a Matrix, Twenty-Five Years Later*, SIAM Review Vol. 45, No.1, pp.3-49, March., 2003

[14] B. N. Parlett, *A recurrence among the elements of functions of triangular matrices*, Linear Algebra Appl., 14 (1976), pp. 117-121.

[15] B. N. Parlett, *Computations of Functions of Triangular Matrices*, Electronics Research Laboratory, University of California-Berkeley Memorandum No. ERL-M481., (November 1974).

[16] Dr C. E. Parnell and Dr Stéphane Régnier, *Numerical Analysis*, Lecturer's Notes MT 3802, October 15, 2008.

[17] R.F. Rinehart *The equivalence of Definitions of a Matric Function*, American Mathematical Monthly, Vol. 62, pp.395-414, 1955.

[18] G. W. Stewart, *Matrix Algoritms*, SIAM, 2001, xix+469 pp. ISBN 0-89871-503-2(VOLUME II).

[19] J. Stoer, R. Bulirsch , *Introduction to Numerical Analysis, Third Edition*, Springer. (2002, 1980, 1997), xv+744pp. ISBN 0-387-95452-X.

[20] P. Waltman, *A Second Course in Elementary Differential Equations*, Dover Publications, 2004-03. ISBN 0486434788. pages 259.

[21] D. S. Watkins, *Fundamentals of MATRIX Computations*, John Wiley and Sons, 1991, xiii+449 pp. ISBN 0-471-61414-9 (cloth).