Numerical Linear Algebra

Jörg Liesen - TU Berlin

Version of September 23, 2024

> "Thus finite linear systems stand at the heart of all mathematical computation. Moreover, as science and technology develop, and computers become more powerful, systems to be handled become larger and require techniques that are more refined and efficient."
>
> "In fact, our subject is more than just vectors and matrices, for virtually everything we do carries over to functions and operators. Numerical linear algebra is really functional analysis, but with the emphasis always on practical algorithmic ideas rather than mathematical technicalities."

The two quotes above reflect two main characteristics of numerical linear algebra: As a mathematical field it is closely related to functional analysis, and one of its major driving forces is the practical requirement to solve linear algebraic problems of rapidly increasing sizes. The quotes are taken from two excellent books, one modern and one classical, on numerical linear algebra:

- L. N. Trefethen and D. Bau III, *Numerical Linear Algebra*, SIAM, 1997,
- A. S. Householder, *The Theory of Matrices in Numerical Analysis*, Blaisdell, 1964.

Can you guess which quote is taken from which book?

Here is another quote about the nature of the field:

> "Quo Vadis? ... From the positive point of view, this huge variety in methods has provided us with a powerful toolbox from which we can select the appropriate tool for a given problem. From the point of view of the poor user, however, this toolbox is confusingly large. In many practical situations it is not clear at all what method to select. One is often faced with the question from outside the expert community which method is the best, but there is in general no best method."
>
> Gene H. Golub and Henk A. van der Vorst, *Closer to the Solution: Iterative Linear Solvers*, in The State of the Art in Numerical Analysis, I. S. Duff and G. A. Watson, eds., Oxford University Press, 1997.

As indicated in this quote, the field of Numerical Linear Algebra contains a huge, even confusing, variety of methods. The overall goal of this course, which focusses on the numerical solution of linear algebraic systems and eigenvalue problems, is to obtain a thorough understanding of some of the most important methods in order to make a well informed choice when solving practical problems. Always keep in mind that there is no single best method for solving all linear algebraic systems or all eigenvalue problems. Moreover, in the "real world" we always need to use the structure of the given problem for constructing problem-adapted modifications that improve the performance of the "textbook algorithms". Such modifications can only be found when the methods are well understood.

This course starts with a survey of matrix decompositions and a treatment of the fundamentals of perturbation theory, where the focus is on linear algebraic systems. We then study several direct and iterative methods for solving linear algebraic systems, including the Cholesky decomposition, the LU decomposition with partial pivoting, "classical" iterative methods, and Krylov subspace methods. Most of these methods can be derived from one of the matrix decompositions we studied at the beginning. The second part of the course will focus on the numerical solution of eigenvalue problems. First we will study the perturbation theory of eigenvalue problems. We will then derive and analyze several different methods for solving such problems numerically. Again, many of the derivations will be based on matrix decompositions.

The course is based in part on the following books:

- G. H. Golub and C. F. Van Loan, *Matrix Computations*, 4th ed., The Johns Hopkins University Press, 2013,
- N. J. Higham *Accuracy and Stability of Numerical Algorithms*, 2nd ed., SIAM, 2002,
- J. Liesen and Z. Strakoš, *Krylov Subspace Methods. Principles and Analysis*, Oxford University Press, 2013,
- Y. Saad, *Numerical Methods for Large Eigenvalue Problems*, revised ed., SIAM, 2011. See Saad's website:
  `http://www-users.cs.umn.edu/~saad/eig_book_2ndEd.pdf`
- G. W. Stewart and J.-g. Sun, *Matrix Perturbation Theory*, Academic Press, 1990,
- L. N. Trefethen and D. Bau III, *Numerical Linear Algebra*, SIAM, 1997.

Some further references are cited in the text.

Many thanks to Daniel Bardutzky, Felix Binkowski, Lana Luise Blaschke, Carlos Echeverría Serur, Davide Fantin, Luis García Ramos, Moritz Geist, Denis Güldner, Nora Heinrich, Valgerður Helgadóttir, Lennart Helmstädter, Alexander Hopp, Mathias Klare, Thorsten Lucke, Ekkehard Schnoor, Olivier Sète, Johannes Taraz, and Jan Zur for reading previous versions of this script and for providing comments and corrections.
Please send further corrections to me at `liesen@math.tu-berlin.de`.

Jörg Liesen, Berlin, September 23, 2024

Contents

Chapter 0

Preliminaries

In this chapter we collect useful notation, definitions and results, which is mostly standard material from Linear Algebra.

0.1 Matrices and important matrix classes

We will (mostly) consider complex matrices, or matrices over $\mathbb{C}$, i.e., matrices of the form

$$A = [a_{ij}] = \begin{bmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nm} \end{bmatrix} \in \mathbb{C}^{n\times m}$$

with $a_{ij} \in \mathbb{C}$, $i = 1, \ldots, n$, $j = 1, \ldots, m$. If $m = 1$ we will usually write $\mathbb{C}^n$ (instead of $\mathbb{C}^{n\times 1}$). If $n = m$, we have a *square matrix* A. In this case we define $\operatorname{trace}(A) := \sum_{i=1}^{n} a_{ii}$. For two matrices $A \in \mathbb{C}^{n\times m}$ and $B \in \mathbb{C}^{m\times k}$ we then have

$$\operatorname{trace}(AB) = \sum_{i=1}^{n}\sum_{j=1}^{m} a_{ij}b_{ji} = \sum_{j=1}^{m}\sum_{i=1}^{n} b_{ji}a_{ij} = \operatorname{trace}(BA).$$

The matrix $I_n := [\delta_{ij}] \in \mathbb{C}^{n\times n}$ is called the *identity matrix*. Here

$$\delta_{ij} := \begin{cases} 1, & i = j, \\ 0, & i \neq j, \end{cases}$$

is the *Kronecker delta*. When the size is clear or irrelevant we write I. The matrix $0_{n\times m} := [0] \in \mathbb{C}^{n\times m}$ is called the *zero matrix*. Usually we just write 0.

If $A = [a_{ij}] \in \mathbb{C}^{n\times n}$ satisfies

$$a_{ij} = 0 \text{ for } \left\{ \begin{array}{l} i \neq j \\ i > j \\ i < j \end{array} \right\} \text{ then } A \text{ is called } \begin{cases} \textit{diagonal}, \\ \textit{upper triangular}, \\ \textit{lower triangular}. \end{cases}$$

A matrix of size 1×1 is trivially diagonal, upper triangular, and lower triangular. Moreover, a (square) matrix is diagonal if and only if it is both upper and lower triangular.
If $A = [a_{ij}] \in \mathbb{C}^{n\times n}$ and $B = [b_{ij}] \in \mathbb{C}^{n\times n}$ are both diagonal, upper triangular, or lower triangular matrices, then it is easy to see that the product $AB = [c_{ij}]$ is again diagonal, upper triangular, or lower triangular, respectively, with $c_{ii} = a_{ii}b_{ii}$ for $i = 1, \ldots, n$.
We sometimes write diagonal matrices as $A = \operatorname{diag}(a_{11}, \ldots, a_{nn})$. An upper or lower triangular matrix with $a_{ii} = 1$ for $i = 1, \ldots, n$ is called *unit* upper or lower triangular, respectively.

If for $A \in \mathbb{C}^{n\times n}$ there exists a matrix $B \in \mathbb{C}^{n\times n}$ with $AB = BA = I_n$, then A is called *nonsingular*. Otherwise A is called *singular*. It is well known that A is nonsingular if and only if $\det(A) \neq 0$, which holds if and only if $\operatorname{rank}(A) = n$.

Lemma 0.1. *For every $A \in \mathbb{C}^{n\times n}$ the following assertions hold:*

(1) If A is nonsingular, then there exists only one matrix $B \in \mathbb{C}^{n\times n}$ with $AB = BA = I_n$. We call this matrix the inverse *of A and denote it by A^{-1}.*

(2) If $AB = I_n$ or $BA = I_n$ holds for some $B \in \mathbb{C}^{n\times n}$, then A is nonsingular and $B = A^{-1}$.

Proof. (1) Suppose that $AB = BA = I_n$ and $AC = CA = I_n$. Then $C = CI_n = CAB = I_nB = B$.
(2) If $AB = I_n$, then $n = \operatorname{rank}(I_n) = \operatorname{rank}(AB) \leq \operatorname{rank}(A)$. Hence A is nonsingular with a unique inverse A^{-1}. Then $A^{-1} = A^{-1}I_n = A^{-1}AB = I_nB = B$. A similar argument applies when $BA = I_n$. □

Item (2) of Lemma 0.1 shows that only one of the equations $AB = I_n$ or $BA = I_n$ needs to be verified in order to show that a given matrix $B \in \mathbb{C}^{n\times n}$ is the (unique) inverse of a nonsingular matrix $A \in \mathbb{C}^{n\times n}$.
An upper triangular matrix $R = [r_{ij}] \in \mathbb{C}^{n\times n}$ is nonsingular if and only if $r_{ii} \neq 0$ for $i = 1, \ldots, n$. In this case $R^{-1} = [\widetilde{r}_{ij}]$ is again upper triangular with $\widetilde{r}_{ii} = r_{ii}^{-1}$ for $i = 1, \ldots, n$. In particular, the inverse of a unit upper triangular matrix is again unit upper triangular. The analogous statements hold for lower triangular matrices.

For $A = [a_{ij}] \in \mathbb{C}^{n\times m}$ the matrices

$$A^T = [b_{ij}] \in \mathbb{C}^{m\times n} \text{ with } b_{ij} := a_{ji}, \text{ and}$$
$$A^H = [b_{ij}] \in \mathbb{C}^{m\times n} \text{ with } b_{ij} := \overline{a}_{ji}$$

are called the *transpose* and *Hermitian transpose* of A. If $A = A^T$ or $A = A^H$, then A is called *symmetric* or *Hermitian*, respectively. Note that if $A \in \mathbb{C}^{n\times n}$ is Hermitian, then

$$x^HAx = x^HA^Hx = (x^HAx)^H = \overline{x^HAx},$$

and thus $x^HAx \in \mathbb{R}$ for all $x \in \mathbb{C}^n$.

If $A \in \mathbb{C}^{n\times n}$ is Hermitian and

$$x^H Ax > 0 \text{ for all } x \in \mathbb{C}^n \setminus \{0\}, \text{ or}$$
$$x^H Ax \geq 0 \text{ for all } x \in \mathbb{C}^n,$$

then A is called *Hermitian positive definite (HPD)* or *Hermitian positive semidefinite (HPSD)*, respectively. If the reverse inequalities hold, A is *Hermitian negative (semi)definite*.

In the next lemma we collect some useful properties of HPSD and HPD matrices.

Lemma 0.2. *If $A \in \mathbb{C}^{n\times n}$ is HPSD, then the following assertions hold:*

(1) $\lambda \geq 0$ for every eigenvalue λ of A (and $\lambda > 0$ if A is HPD).

(2) $X^H AX$ is HPSD for all $X \in \mathbb{C}^{n\times k}$, and if A is HPD and $\operatorname{rank}(X) = k$, *then $X^H AX$ is also HPD.*

(3) $A(1{:}k, 1{:}k) := \begin{bmatrix} a_{11} & \cdots & a_{1k} \\ \vdots & \ddots & \vdots \\ a_{k1} & \cdots & a_{kk} \end{bmatrix}$ *is HPSD for all $k = 1, \ldots, n$, and if A is HPD, then $A(1{:}k, 1{:}k)$ is also HPD.*

Proof. (1) If $Ax = \lambda x$ with $x \neq 0$, then $0 \leq x^H Ax = \lambda x^H x$, giving $\lambda \geq 0$. If A is HPD we have strict inequalities.
(2) Let $X \in \mathbb{C}^{n\times k}$, $y \in \mathbb{C}^k$ and write $x := Xy$. Then $y^H(X^H AX)y = x^H Ax \geq 0$ since A is HPSD. If A is HPD and $\operatorname{rank}(X) = k$, then for every $0 \neq y \in \mathbb{C}^k$ we have $x := Xy \neq 0$ and thus $y^H(X^H AX)y = x^H Ax > 0$.
(3) This follows from (2) using the matrix $X = [e_1, \ldots, e_k] \in \mathbb{C}^{n\times k}$, which has rank k. □

If $A \in \mathbb{C}^{n\times n}$ satisfies $A^H A = I_n$, then A is called *unitary*. Item (2) of Lemma 0.1 implies that then A is nonsingular with $A^{-1} = A^H$, and that also $AA^H = I_n$. Analogously, a matrix $A \in \mathbb{R}^{n\times n}$ with $A^T A = I_n$, and hence also $AA^T = I_n$, is called *orthogonal*.
If we write $A = [a_1, \ldots, a_n] \in \mathbb{C}^{n\times n}$ with $a_j \in \mathbb{C}^n$, $j = 1 \ldots, n$, then $A^H A = [a_i^H a_j] = I_n = [\delta_{ij}]$ means that the n columns of A are pairwise orthonormal with respect to the *Euclidean inner product* on $\mathbb{C}^n$, which is defined by

$$\langle v, w\rangle := w^H v \text{ for all } v, w \in \mathbb{C}^n.$$

Hence $A \in \mathbb{C}^{n\times n}$ is unitary if and only if the columns $a_1, \ldots, a_n$ form an orthonormal basis of $\mathbb{C}^n$ with respect to the inner product $\langle \cdot, \cdot \rangle$. The equation $AA^H = I_n$ means the same holds for the (transposed) rows of A. We have the analogous observations in the real case for an orthogonal matrix and the Euclidean inner product $\langle v, w\rangle := w^T v$ on $\mathbb{R}^n$. (See Definition 0.7 below for the general definition of an inner product on a complex vector space.)

If $A \in \mathbb{C}^{n\times m}$ with $n > m$ satisfies $A^H A = I_m$, then A has pairwise orthonormal columns with respect to $\langle\cdot,\cdot\rangle$, but A is *not* a unitary matrix. In this case $P := AA^H$ is a *projection* (i.e., $P^2 = P$) with $\operatorname{rank}(P) = m$.

If $A \in \mathbb{C}^{n\times n}$ satisfies $A^H A = AA^H$ then A is called *normal*. Note that Hermitian and unitary matrices are normal.

0.2 Norms

We now review the most important definitions and concepts in the context of norms.

Definition 0.3. *A function $\|\cdot\| : \mathcal{V} \to \mathbb{R}$ is called a* norm *on a real or complex vector space $\mathcal{V}$ when the following hold:*

(1) $\|v\| \geq 0$ for all $v \in \mathcal{V}$ with equality if and only if $v = 0$,

(2) $\|\alpha v\| = |\alpha|\|v\|$ for all scalars α and $v \in \mathcal{V}$,

(3) $\|v_1 + v_2\| \leq \|v_1\| + \|v_2\|$ for all $v_1, v_2 \in \mathcal{V}$ (so-called triangle inequality).

For example, given a real number $p \geq 1$, the *p-norm on* $\mathbb{C}^n$ is defined by

$$\|x\|_p := \left(\sum_{j=1}^{n} |x_j|^p\right)^{1/p}, \tag{0.1}$$

where $x = [x_1, \ldots, x_n]^T \in \mathbb{C}^n$. Important special cases are

$$\begin{aligned}
\|x\|_1 &= \sum_{j=1}^{n} |x_j|, \\
\|x\|_2 &= \left(\sum_{j=1}^{n} |x_j|^2\right)^{1/2} = (x^H x)^{1/2} \quad \text{(Euclidean or 2-norm)}, \\
\|x\|_\infty &= \max_{1\leq j\leq n} |x_j| \quad \text{(maximum or } \infty\text{-norm)},
\end{aligned}$$

where the last norm is obtained by taking the limit of $\|x\|_p$ for $p \to \infty$.
A frequently used norm on $\mathbb{C}^{n\times m}$ is the *Frobenius norm*, which is defined by

$$\|A\|_F := \left(\sum_{i=1}^{n}\sum_{j=1}^{m} |a_{ij}|^2\right)^{1/2},$$

for $A = [a_{ij}] \in \mathbb{C}^{n\times m}$.

Definition 0.4. *Let $|\cdot|$, $\|\cdot\|$, $\|\cdot\|_*$ be norms on $\mathbb{C}^{n\times m}$, $\mathbb{C}^{m\times k}$, $\mathbb{C}^{n\times k}$, respectively. These norms are called* consistent, *when*

$$\|AB\|_* \leq |A| \cdot \|B\|$$

holds for all $A \in \mathbb{C}^{n\times m}$ and $B \in \mathbb{C}^{m\times k}$. A single norm $\|\cdot\|$ on $\mathbb{C}^{n\times n}$ is called consistent *if $\|AB\| \leq \|A\|\,\|B\|$ holds for all $A, B \in \mathbb{C}^{n\times n}$.*

We will now show that each given (vector) norm on $\mathbb{C}^n$ can be used to defined a (matrix) norm on $\mathbb{C}^{n\times n}$, so that the two norms are consistent.

Lemma 0.5. *If $\|\cdot\|$ is any norm on $\mathbb{C}^n$, then the function*

$$\|\cdot\|_* : \mathbb{C}^{n\times n} \to \mathbb{R} \quad \textit{with} \quad \|A\|_* := \max_{\substack{x\in\mathbb{C}^n \\ \|x\|=1}} \|Ax\| = \max_{0\neq x\in\mathbb{C}^n} \frac{\|Ax\|}{\|x\|}$$

is a norm on $\mathbb{C}^{n\times n}$, which is called the matrix norm induced by $\|\cdot\|$. Moreover, the norms $\|\cdot\|$ and $\|\cdot\|_*$ are consistent.

Proof. We first prove that $\|\cdot\|_*$ satisfies properties (1)–(3) from Definition 0.3:
(1) It is clear that $\|A\|_* \geq 0$ holds for all $A \in \mathbb{C}^{n\times n}$. Moreover, by definition, $\|A\|_* = 0$ holds if and only if $\|Ax\| = 0$ for all $x \in \mathbb{C}^n$. Since $\|\cdot\|$ is a norm on $\mathbb{C}^n$, property (1) from Definition 0.3 tells us that this hold if and only if $Ax = 0$ for all $x \in \mathbb{C}^n$, and this is equivalent with $A = 0$.
(2) For all $A \in \mathbb{C}^{n\times n}$ and $\alpha \in \mathbb{C}$ we have

$$\|\alpha A\|_* = \max_{\substack{x\in\mathbb{C}^n \\ \|x\|=1}} \|\alpha Ax\| = |\alpha| \max_{\substack{x\in\mathbb{C}^n \\ \|x\|=1}} \|Ax\| = |\alpha|\|A\|_*,$$

where we have used property (2) from Definition 0.3 for the norm $\|\cdot\|$.
(3) For all $A, B \in \mathbb{C}^{n\times n}$ we have

$$\|A+B\|_* = \max_{\substack{x\in\mathbb{C}^n \\ \|x\|=1}} \|(A+B)x\| \leq \max_{\substack{x\in\mathbb{C}^n \\ \|x\|=1}} (\|Ax\| + \|Bx\|) \leq \|A\|_* + \|B\|_*,$$

where we have used property (3) from Definition 0.3 for the norm $\|\cdot\|$.

The norms $\|\cdot\|$ and $\|\cdot\|_*$ are consistent, since for all $A \in \mathbb{C}^{n\times n}$ and $y \in \mathbb{C}^n \setminus \{0\}$ we have

$$\frac{\|Ay\|}{\|y\|} \leq \max_{0\neq x\in\mathbb{C}^n} \frac{\|Ax\|}{\|x\|} = \|A\|_*,$$

which implies $\|Ay\| \leq \|A\|_*\|y\|$. For $y = 0$ this inequality holds trivially. □

For example, the (vector) p-norm on $\mathbb{C}^n$ induces (matrix) norm on $\mathbb{C}^{n\times n}$, and both are usually denoted by the same symbol. In particular, we have

$$\|A\|_1 = \max_{1\le j\le n} \sum_{i=1}^{n} |a_{ij}|,$$
$$\|A\|_\infty = \max_{1\le i\le n} \sum_{j=1}^{n} |a_{ij}|.$$

With an analogous proof as for Lemma 0.5 it can be shown that two norms $|\cdot|$ and $\|\cdot\|$ on $\mathbb{C}^n$ and $\mathbb{C}^m$, respectively, induce a norm on $\mathbb{C}^{n\times m}$ given by

$$\|\cdot\|_* : \mathbb{C}^{n\times m} \to \mathbb{R} \quad \text{with} \quad \|A\|_* := \max_{\substack{x\in\mathbb{C}^m \\ \|x\|=1}} |Ax| = \max_{0\neq x\in\mathbb{C}^m} \frac{|Ax|}{\|x\|}.$$

Now the three norms $|\cdot|$, $\|\cdot\|$, $\|\cdot\|_*$ are consistent, since for all $A \in \mathbb{C}^{n\times m}$ and $y \in \mathbb{C}^m \setminus \{0\}$ we have

$$\frac{|Ay|}{\|y\|} \le \max_{0\neq x\in\mathbb{C}^m} \frac{|Ax|}{\|x\|} = \|A\|_*,$$

which implies $|Ay| \le \|A\|_*\|y\|$, and again this inequality holds trivially for $y = 0$.

Next we will show that each given consistent (matrix) norm on $\mathbb{C}^{n\times n}$ can be used to define a (vector) norm on $\mathbb{C}^n$ so that the two norms are consistent.

Lemma 0.6. *If $\|\cdot\|_*$ is any consistent norm on $\mathbb{C}^{n\times n}$, and $y \in \mathbb{C}^n \setminus \{0\}$, then the function*

$$\|\cdot\| : \mathbb{C}^n \to \mathbb{R} \quad \textit{with} \quad \|x\| := \|xy^H\|_*$$

is a norm on $\mathbb{C}^n$. Moreover, the norms $\|\cdot\|$ and $\|\cdot\|_*$ are consistent.

Proof. We prove that $\|\cdot\|$ satisfies properties (1)–(3) from Definition 0.3:
(1) It is clear that $\|x\| = \|xy^H\|_* \ge 0$ for all $x \in \mathbb{C}^n$. Moreover, $\|xy^H\|_* = 0$ holds if and only if $xy^H = 0$, since $\|\cdot\|_*$ is a norm on $\mathbb{C}^{n\times n}$. Multiplying from the right by $y \neq 0$ gives $x(y^Hy) = 0$, and since $y^Hy > 0$, we must have $x = 0$.
(2) For all $x \in \mathbb{C}^n$ and $\alpha \in \mathbb{C}$ we have $\|\alpha x\| = \|\alpha xy^H\|_* = |\alpha|\|xy^H\|_* = |\alpha|\|x\|$, where we have used property (2) from Definition 0.3 for the norm $\|\cdot\|_*$.
(3) For all $x_1, x_2 \in \mathbb{C}^n$ we have $\|x_1 + x_2\| = \|(x_1 + x_2)y^H\|_* \le \|x_1y^H\|_* + \|x_2y^H\|_* = \|x_1\| + \|x_2\|$, where we have used property (3) from Definition 0.3 for the norm $\|\cdot\|_*$.

The norms $\|\cdot\|$ and $\|\cdot\|_*$ are consistent, since for each $A \in \mathbb{C}^{n\times n}$ and $x \in \mathbb{C}^n$ we have

$$\|Ax\| = \|A(xy^H)\|_* \le \|A\|_*\|xy^H\|_* = \|A\|_*\|x\|,$$

where we have used that $\|\cdot\|_*$ is consistent. □

A norm $\|\cdot\|$ on $\mathbb{C}^{n\times m}$ is called *unitarily invariant* if it is invariant under left and right multiplication with unitary matrices, i.e.,

$$\|A\| = \|UAV\| \quad \text{holds for all unitary matrices } U \in \mathbb{C}^{n\times n} \text{ and } V \in \mathbb{C}^{m\times m}.$$

For example, the 2-norm on $\mathbb{C}^{n\times m}$ which is induced by the 2-norms on $\mathbb{C}^m$ and $\mathbb{C}^n$, i.e.,

$$\|A\|_2 = \max_{\substack{x\in\mathbb{C}^m \\ \|x\|_2=1}} \|Ax\|_2,$$

is unitarily invariant, since

$$\begin{aligned}\|UAV\|_2^2 &= \max_{\substack{x\in\mathbb{C}^m \\ \|x\|_2=1}} \|UAVx\|_2^2 = \max_{\substack{x\in\mathbb{C}^m \\ \|x\|_2=1}} x^H V^H A^H U^H UAVx \quad (\text{set } y := Vx) \\ &= \max_{\substack{y\in\mathbb{C}^m \\ \|y\|_2=1}} y^H A^H Ay = \max_{\substack{y\in\mathbb{C}^m \\ \|y\|_2=1}} \|Ay\|_2^2 = \|A\|_2^2.\end{aligned}$$

Note that here we have used $\|x\|_2 = \|Vx\|_2 = \|y\|_2$, i.e., the unitary invariance of the 2-norm on $\mathbb{C}^m$.

In Lemma 0.5 we have induced a (matrix) norm on $\mathbb{C}^{n\times n}$ by a given (vector) norm on $\mathbb{C}^n$. There is another way of inducing norms on $\mathbb{C}^{n\times n}$, namely via an inner product.

Definition 0.7. *A function $\langle\cdot,\cdot\rangle : \mathcal{V}\times\mathcal{V} \to \mathbb{C}$ is called an* inner product *on a complex vector space $\mathcal{V}$ when the following hold:*

(1) $\langle \lambda u + \mu v, w\rangle = \lambda\langle u, w\rangle + \mu\langle v, w\rangle$ for all $\lambda, \mu \in \mathbb{C}$ and $u, v, w \in \mathcal{V}$ (linearity in the first component).

(2) $\langle v, w\rangle = \overline{\langle w, v\rangle}$ for all $u, v \in \mathcal{V}$ (conjugate symmetry).

(3) $\langle v, v\rangle \geq 0$ for all $v \in \mathcal{V}$ with equality if and only if $v = 0$ (positive definiteness).

For a real vector space $\mathcal{V}$ we have the analogous definition, but here $\langle\cdot,\cdot\rangle : \mathcal{V}\times\mathcal{V} \to \mathbb{R}$, and the conjugate symmetry becomes (real) symmetry, i.e., $\langle v, w\rangle = \langle w, v\rangle$ for all $u, v \in \mathcal{V}$.

Lemma 0.8. *If $\langle\cdot,\cdot\rangle$ is an inner product on a real or complex vector space $\mathcal{V}$, then the function*

$$\|\cdot\| : \mathcal{V} \to \mathbb{R} \quad \textit{with} \quad \|v\| := \langle v, v\rangle^{1/2}$$

is a norm on $\mathcal{V}$, which is called the norm induced by $\langle\cdot,\cdot\rangle$.

In particular, every inner product on $\mathbb{C}^{n\times n}$ induces a norm on $\mathbb{C}^{n\times n}$. An important example is the *trace inner product* on $\mathbb{C}^{n\times n}$, i.e., the function

$$\langle A, B\rangle := \operatorname{trace}(B^H A).$$

This inner product induces the Frobenius norm on $\mathbb{C}^{n\times n}$ since

$$\langle A, A\rangle^{1/2} = \operatorname{trace}(A^H A)^{1/2} = \left(\sum_{i=1}^{n}\sum_{j=1}^{n} |a_{ij}|^2\right)^{1/2} = \|A\|_F.$$

For any $A \in \mathbb{C}^{n\times n}$ and any unitary matrices $U, V \in \mathbb{C}^{n\times n}$ we have

$$\|UAV\|_F^2 = \operatorname{trace}(V^H A^H AV) = \operatorname{trace}(AVV^H A^H) = \operatorname{trace}(AA^H) = \operatorname{trace}(A^H A) = \|A\|_F^2,$$

which shows that the Frobenius norm is unitarily invariant. The analogous results hold for the real case, where we consider the inner product $\langle A, B\rangle := \operatorname{trace}(B^T A)$ on $\mathbb{R}^{n\times n}$ and obtain the orthogonal invariance of the Frobenius norm on $\mathbb{R}^{n\times n}$.
The following is a selection of inequalities that hold for $A \in \mathbb{C}^{n\times n}$ and the matrix norms mentioned above:

$$\frac{1}{\sqrt{n}}\|A\|_1 \le \|A\|_2 \le \sqrt{n}\|A\|_1, \qquad \|A\|_2 \le \sqrt{\|A\|_1\|A\|_\infty},$$
$$\frac{1}{\sqrt{n}}\|A\|_\infty \le \|A\|_2 \le \sqrt{n}\|A\|_\infty, \qquad \|A\|_2 \le \|A\|_F \le \sqrt{n}\|A\|_2.$$

Note that the entry-based matrix norms $\|\cdot\|_1$, $\|\cdot\|_\infty$, and $\|\cdot\|_F$ can not decrease when we increase the absolute value of some entries a_{ij}. The matrix 2-norm, on the other hand, may behave somewhat counterintuitively. For example,

$$\left\|\begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix}\right\|_2 \approx 1.6180 \quad \text{but} \quad \left\|\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}\right\|_2 \approx 1.4142.$$

Chapter 1

A Survey of Matrix Decompositions

In the January/February 2000 issue of *Computing in Science & Engineering*, a joint publication of the American Institute of Physics and the IEEE Computer Society, a list of the "Top Ten Algorithms of the Century" was published. Among the Top Ten Algorithms (ordered by year, so there is no "No. 1 Algorithm") is

> "1951: Alston Householder of Oak Ridge National Laboratory formalizes the decompositional approach to matrix computations."

In his introduction to the topic, Stewart [39] wrote that

> "the introduction of matrix decomposition into numerical linear algebra revolutionized matrix computations[1]".

A matrix decomposition is nothing but a factorization of the original matrix into "simpler" factors. Householder started with a systematical analysis of methods for inverting matrices (or solving linear algebraic systems) from the viewpoint of matrix decomposition in 1950 [19]. In 1957 he wrote [20]:

> "Most, if not all closed [i.e. direct] methods can be classified as methods of factorizations and methods of modification [...] these methods of factorization aim to express A as a product of two factors, each of which is readily inverted, or, equivalently, to find matrices P and Q such that $PA = Q$ and Q is easily inverted."

Major advantages of the decompositional approach as stated by Stewart [39] are:

[1] Stewart and many others use the term *matrix computations* as a synonym for *numerical linear algebra*. This has been done at least since Stewart's book *Introduction to Matrix Computations*, Academic Press, 1973. Stewart wrote in 1987 [37, p. 211]: "It is customary to identify the beginnings of modern *numerical linear algebra* with the introduction of the digital computer in the mid nineteen forties. ... Wherever one chooses to place the beginnings of *matrix computations*, it is certain that by the mid forties it had entered an expansive phase, from which it has not yet emerged." (My emphasis.)

- A matrix decomposition, which is generally expensive to compute, can be reused to solve new problems involving the original matrix.
- The decompositional approach often shows that apparently different algorithms are actually computing the same object.
- The decompositional approach facilitates rounding error analysis.
- Many matrix decompositions can be updated, sometimes with great savings in computation.
- By focusing on a few decompositions instead of a host of specific problems, software developers have been able to produce highly effective matrix packages.

We will now discuss several important matrix decompositions from a mathematical point of view, i.e., we will be mostly interested in their existence and uniqueness. In later chapters we will derive algorithms for computing the decompositions, analyze their numerical stability, and apply them in order to solve problems of numerical linear algebra.

Theorem 1.1 (LU decomposition)**.** *The following assertions are equivalent for every matrix $A \in \mathbb{C}^{n\times n}$:*

(1) There exist a unit lower triangular matrix $L \in \mathbb{C}^{n\times n}$, a nonsingular diagonal matrix $D \in \mathbb{C}^{n\times n}$, and a unit upper triangular matrix $U \in \mathbb{C}^{n\times n}$, such that

$$A = LDU.$$

(Here unit lower and unit upper triangular means that all diagonal entries of the respective triangular matrices are 1.)

(2) For each $k = 1, \ldots, n$ the matrix $A(1{:}k, 1{:}k) \in \mathbb{C}^{k\times k}$ is nonsingular.

If (2) holds, then there exists only one set of matrices L, D, U with the properties stated in (1) and $A = LDU$.

Proof. (1) $\Longrightarrow$ (2): If $A = LDU$, where L, D, U have the stated properties, we can consider any fixed k between 1 and n, and partition

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{bmatrix} \begin{bmatrix} D_{11} & 0 \\ 0 & D_{22} \end{bmatrix} \begin{bmatrix} U_{11} & U_{12} \\ 0 & U_{22} \end{bmatrix},$$

where $A_{11} = A(1{:}k, 1{:}k) \in \mathbb{C}^{k\times k}$. We see that $A_{11} = L_{11}D_{11}U_{11}$, and since L_{11} and U_{11} are unit lower and upper triangular, respectively, we obtain

$$\det(A_{11}) = \det(L_{11}) \det(D_{11}) \det(U_{11}) = \det(D_{11}) \neq 0.$$

(2) $\Longrightarrow$ (1): Induction on n. For $n = 1$ we can take $L = [1]$, $D = [a_{11}]$, $U = [1]$. Now suppose that the statement is true for all matrices up to order $n-1$ for some $n \geq 2$ and let $A \in \mathbb{C}^{n\times n}$. With the nonsingular matrix $A_{11} := A(1{:}n-1, 1{:}n-1)$ we can write

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} I_{n-1} & 0 \\ A_{21}A_{11}^{-1} & 1 \end{bmatrix} \begin{bmatrix} A_{11} & 0 \\ 0 & s \end{bmatrix} \begin{bmatrix} I_{n-1} & A_{11}^{-1}A_{12} \\ 0 & 1 \end{bmatrix} =: L_n \begin{bmatrix} A_{11} & 0 \\ 0 & s \end{bmatrix} U_n,$$

where $s := A_{22} - A_{21}A_{11}^{-1}A_{12} \in \mathbb{C}$ is the *Schur complement* of A_{11} in A. From

$$0 \neq \det(A) = \underbrace{\det(L_n)}_{=1}\underbrace{\det(A_{11})}_{\neq 0}\ s\ \underbrace{\det(U_n)}_{=1},$$

we get $s \neq 0$. By the induction hypothesis, the matrix A_{11} has a factorization $A_{11} = L_{n-1}D_{n-1}U_{n-1}$, where L_{n-1}, D_{n-1} and U_{n-1} have the required properties. Then

$$A = \underbrace{L_n \begin{bmatrix} L_{n-1} & 0 \\ 0 & 1 \end{bmatrix}}_{=:L} \underbrace{\begin{bmatrix} D_{n-1} & 0 \\ 0 & s \end{bmatrix}}_{=:D} \underbrace{\begin{bmatrix} U_{n-1} & 0 \\ 0 & 1 \end{bmatrix} U_n}_{=:U}$$

is the required decomposition.
Finally, suppose that $A = L_1D_1U_1 = L_2D_2U_2$ with $L_k = [l_{ij}^{(k)}]$ and $U_k = [u_{ij}^{(k)}]$, $k = 1, 2$, unit lower and upper triangular, respectively. Then $L_2^{-1}L_1D_1 = D_2U_2U_1^{-1}$ is lower and upper triangular, and hence diagonal. From $l_{ii}^{(1)} = l_{ii}^{(2)} = 1 = u_{ii}^{(1)} = u_{ii}^{(2)}$ and the structure of the matrices we immediately obtain $L_2^{-1}L_1 = I_n$ and $U_2U_1^{-1} = I_n$, hence $L_1 = L_2$, $U_1 = U_2$, and $D_1 = D_2$. □

A simple example where the condition (2) in Theorem 1.1 does not hold is given by the upper triangular matrix

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

which trivially has a the decomposition $A = LDU$ with $L = D = I$ and $U = A$. Note that here U is not unit upper triangular. Moreover, requiring that L is unit lower triangular and D is a nonsingular diagonal matrix not determine the factors uniquely, since

$$\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\alpha\beta & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & \alpha \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 0 & \beta \end{bmatrix},$$

for any choice of the parameters α and β.
Another example where the condition (2) does not hold is given by the matrix

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

This matrix does not have any decomposition of the form $A = LDU$ with D diagonal, L lower, and U upper triangular. But if we exchange (i.e., permute) the rows of A, then the resulting matrix

$$PA = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \quad P := \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix},$$

has an (obvious) LU decomposition. Allowing row exchanges is also important for the numerical stability of numerical methods for computing an LU decomposition; see Chapter 3.

Corollary 1.2 ($\mathrm{LDL^H}$ decomposition.)**.** *If $A = A^H \in \mathbb{C}^{n\times n}$ and $A(1:k, 1:k)$ is nonsingular for all $k = 1, \ldots, n$, then there exist a uniquely determined unit lower triangular matrix $L \in \mathbb{C}^{n\times n}$ and a uniquely determined nonsingular diagonal matrix $D \in \mathbb{R}^{n\times n}$, such that*

$$A = LDL^H.$$

Proof. Let $A = LDU$ be the uniquely determined factorization from Theorem 1.1. Since $A = A^H$ we have $A = LDU = (LDU)^H = U^H D^H L^H$. Here U^H and L^H are unit lower and upper triangular, respectively, and D^H is diagonal and nonsingular. The uniqueness of the factorization now implies that $U^H = L$ and $D = D^H$, and hence in particular $D \in \mathbb{R}^{n\times n}$. □

Corollary 1.3 (Cholesky decomposition)**.** *If $A \in \mathbb{C}^{n\times n}$ is HPD, then there exists a uniquely determined lower triangular matrix $L \in \mathbb{C}^{n\times n}$ with positive diagonal entries, such that*

$$A = LL^H.$$

1st Proof. If A is HPD, then by Corollary 1.2 there exists a uniquely determined factorization $A = \widetilde{L}D\widetilde{L}^H$, where $\widetilde{L} \in \mathbb{C}^{n\times n}$ is unit lower triangular and $D = [d_{ij}] \in \mathbb{R}^{n\times n}$ is nonsingular. By (2) in Lemma 0.2 the matrix $D = \widetilde{L}^{-1}A\widetilde{L}^{-H}$ is HPD and hence $d_{ii} > 0$ for $i = 1, \ldots, n$. We set $L := \widetilde{L}D^{1/2}$, where $D^{1/2} := \mathrm{diag}(d_{11}^{1/2}, \ldots, d_{nn}^{1/2}) \in \mathbb{R}^{n\times n}$, then $A = LL^H$. □

2nd Proof. Induction on n. If $n = 1$, we have $A = [a_{11}]$ with $a_{11} > 0$, and we set $L := [a_{11}^{1/2}]$. Suppose the statement is true for matrices up to order $n-1$ for some $n \geq 2$. Let $A \in \mathbb{C}^{n\times n}$ be HPD, and let $A_{n-1} := A(1{:}n-1, 1{:}n-1) \in \mathbb{C}^{(n-1)\times(n-1)}$, which is HPD; cf. (3) in Lemma 0.2. By the induction hypothesis, there exists a uniquely determined lower triangular matrix $L_{n-1} \in \mathbb{C}^{(n-1)\times(n-1)}$ with positive diagonal entries, such that $A_{n-1} = L_{n-1}L_{n-1}^H$. We thus can write

$$A = \begin{bmatrix} A_{n-1} & b \\ b^H & a_{nn} \end{bmatrix} = \begin{bmatrix} I_{n-1} & 0 \\ b^H A_{n-1}^{-1} & 1 \end{bmatrix} \begin{bmatrix} A_{n-1} & 0 \\ 0 & s \end{bmatrix} \begin{bmatrix} I_{n-1} & A_{n-1}^{-1}b \\ 0 & 1 \end{bmatrix},$$

where $s := a_{nn} - b^H A_{n-1}^{-1} b$. Taking determinants yields

$$0 < \det(A) = \det(A_{n-1})s,$$

and hence $s > 0$. Now define $c := L_{n-1}^{-1}b$, then

$$\|c\|_2^2 = b^H L_{n-1}^{-H} L_{n-1}^{-1} b = b^H A_{n-1}^{-1} b < a_{nn}.$$

Let α be the positive square root of $a_{nn} - \|c\|_2^2 = s > 0$, then the lower triangular matrix $L := \begin{bmatrix} L_{n-1} & 0 \\ c^H & \alpha \end{bmatrix}$ has positive diagonal entries and satisfies $A = LL^H$.

If $\widetilde{L} := \begin{bmatrix} L_{n-1} & 0 \\ d^H & \beta \end{bmatrix}$ with $\beta > 0$ satisfies $\widetilde{L}\widetilde{L}^H = LL^H$, then

$$\begin{bmatrix} A_{n-1} & L_{n-1}d \\ d^H L_{n-1}^H & \|d\|_2^2 + \beta^2 \end{bmatrix} = \begin{bmatrix} A_{n-1} & L_{n-1}c \\ c^H L_{n-1}^H & \|c\|_2^2 + \alpha^2 \end{bmatrix}.$$

Since L_{n-1} is nonsingular, we must have $d = c$, and hence $\beta^2 = \alpha^2$, giving $\beta = \alpha$, since α and β are both positive. □

If $A \in \mathbb{C}^{n\times m}$ has (full) rank m, then the *Gram–Schmidt algorithm* stated in Algorithm 1 yields a matrix $Q \in \mathbb{C}^{n\times m}$ with orthonormal columns and an upper triangular matrix $R = [r_{ij}] \in \mathbb{C}^{m\times m}$ with $r_{ii} > 0$, $i = 1, \ldots, m$, such that $A = QR$. If $n = m$, then Q is unitary.

Algorithm 1 (Classical) Gram–Schmidt algorithm

Input: Matrix $A = [a_1, \ldots, a_m] \in \mathbb{C}^{n\times m}$ with (full) rank m
Output: Matrix $Q \in \mathbb{C}^{n\times m}$ with orthonormal columns and upper triangular matrix $R \in \mathbb{C}^{m\times m}$ with positive diagonal entries
Set $q_1 = a_1/r_{11}$, where $r_{11} = \|a_1\|_2$
for $j = 1, \ldots, m-1$ **do**
  $\widehat{q}_{j+1} = a_{j+1} - \sum_{i=1}^{j} r_{i,j+1} q_i$, where $r_{i,j+1} = \langle a_{j+1}, q_i \rangle$
  $q_{j+1} = \widehat{q}_{j+1}/r_{j+1,j+1}$, where $r_{j+1,j+1} = \|\widehat{q}_{j+1}\|_2$
end for

Let us show by induction that Algorithm 1 indeed generates m orthonormal vectors: By construction, the vector q_1 has unit norm. Suppose that for some $j \in \{1, \ldots, m-1\}$ the vectors $q_1, \ldots, q_j$ are orthonormal, i.e., $\langle q_i, q_\ell \rangle = \delta_{i\ell}$. Then for each $\ell = 1, \ldots, j$ we have

$$\begin{aligned} \langle \widehat{q}_{j+1}, q_\ell \rangle &= \Big\langle a_{j+1} - \sum_{i=1}^{j} \langle a_{j+1}, q_i \rangle q_i, q_\ell \Big\rangle = \langle a_{j+1}, q_\ell \rangle - \sum_{i=1}^{j} \langle a_{j+1}, q_i \rangle \langle q_i, q_\ell \rangle \\ &= \langle a_{j+1}, q_\ell \rangle - \langle a_{j+1}, q_\ell \rangle = 0. \end{aligned}$$

Since additionally $\|q_{j+1}\|_2 = 1$, the vectors $q_1, \ldots, q_{j+1}$ are orthonormal.
Now suppose that $a_1 \in \mathbb{C}^n$ is any nonzero vector. Then a well known and important result of Linear Algebra says that we can complement this vector to obtain a basis $a_1, a_2, \ldots, a_n$ of $\mathbb{C}^n$. Applying Algorithm 1 to the full rank matrix $A = [a_1, \ldots, a_n] \in \mathbb{C}^{n\times n}$ then yields $A = QR$, where Q is unitary and has $a_1/\|a_1\|_2$ as its first column.
In the proof of the following result we will use that for any nonzero vector $a_1 \in \mathbb{C}^n$ there exists a unitary matrix Q that has $a_1/\|a_1\|_2$ as its first column.

Theorem 1.4 (QR decomposition). *Let $A \in \mathbb{C}^{n\times m}$ with $n \geq m$. Then there exist a unitary matrix $Q \in \mathbb{C}^{n\times n}$ and an upper triangular matrix $R = [r_{ij}] \in \mathbb{C}^{m\times m}$ with $r_{ii} \geq 0$ for $i = 1, \ldots, m$, such that*

$$A = Q \begin{bmatrix} R \\ 0 \end{bmatrix}.$$

If $\operatorname{rank}(A) = m$*, then* $r_{ii} > 0$*,* $i = 1, \ldots, m$*. Moreover, denoting* $Q = [Q_1, Q_2]$ *with* $Q_1 \in \mathbb{C}^{n\times m}$*, the matrices* Q_1 *and* R *with* $r_{ii} > 0$*,* $i = 1, \ldots, m$*, are uniquely determined.*

Proof. Induction on m. If $m = 1$ we have $A = [a] \in \mathbb{C}^n$. If $a = 0$, set $Q := I$ and $R := [0]$. If $a \neq 0$, let $Q \in \mathbb{C}^{n\times n}$ be a unitary matrix with first column $a/\|a\|_2$ and $R = [\|a\|_2]$, then

$$A = Q \begin{bmatrix} R \\ 0 \end{bmatrix}$$

as required.
Now suppose that the result is true for all matrices with $m-1$ columns for some $m \geq 2$, and at least $m-1$ rows. Let $A \in \mathbb{C}^{n\times m}$ with $n \geq m$, and write $A = [a, A_1]$, where $A_1 \in \mathbb{C}^{n\times(m-1)}$. If $a = 0$, set $Q_1 := I_n$. If $a \neq 0$, let $Q_1 \in \mathbb{C}^{n\times n}$ be a unitary matrix with first column $a/\|a\|_2$. Then

$$A = Q_1 \begin{bmatrix} \|a\|_2 & b^T \\ 0 & C \end{bmatrix}$$

for some $b^T \in \mathbb{C}^{1\times(m-1)}$ and $C \in \mathbb{C}^{(n-1)\times(m-1)}$. Applying the induction hypothesis to C, we obtain the decomposition $C = Q_2 \begin{bmatrix} R_2 \\ 0 \end{bmatrix}$, where $Q_2 \in \mathbb{C}^{(n-1)\times(n-1)}$ is unitary and $R_2 \in \mathbb{C}^{(m-1)\times(m-1)}$ is upper triangular with nonnegative diagonal entries. Hence with the unitary matrix

$$Q := Q_1 \begin{bmatrix} 1 & 0 \\ 0 & Q_2 \end{bmatrix},$$

we have

$$A = Q_1 \begin{bmatrix} 1 & 0 \\ 0 & Q_2 \end{bmatrix} \begin{bmatrix} \|a\|_2 & b^T \\ 0 & R_2 \\ 0 & 0 \end{bmatrix} =: Q \begin{bmatrix} R \\ 0 \end{bmatrix},$$

where R is upper triangular with nonegative diagonal entries.
If $\operatorname{rank}(A) = m$, then $\operatorname{rank}(R) = m$, which implies $r_{ii} > 0$ for $i = 1, \ldots, m$. If

$$A = Q_1 R = \widetilde{Q}_1 \widetilde{R},$$

where $Q_1, \widetilde{Q}_1 \in \mathbb{C}^{n\times m}$ have orthonormal columns and $R_1, \widetilde{R}_1 \in \mathbb{C}^{m\times m}$ are nonsingular upper triangular matrices with positive diagonal entries. Then $A^H A = R^H R = \widetilde{R}^H \widetilde{R}$, and hence

$$\underbrace{R\widetilde{R}^{-1}}_{\text{upper triangular}} = \underbrace{R^{-H}\widetilde{R}^H}_{\text{lower triangular}} = ((R\widetilde{R}^{-1})^H)^{-1},$$

which implies that $R\widetilde{R}^{-1} =: D$ is diagonal with positive diagonal entries. But then $D = (D^H)^{-1}$ shows that $D = I_m$, from which we see $R = \widetilde{R}$ and thus $Q_1 = \widetilde{Q}_1$. □

There is a close relation between the QR and the Cholesky decomposition. Let $A \in \mathbb{C}^{n\times m}$ have full rank m. If

$$A = Q \begin{bmatrix} R \\ 0 \end{bmatrix} \text{ with } r_{ii} > 0$$

is the (uniquely determined) QR decomposition of A, then $A^HA = R^HR$ is the (uniquely determined) Cholesky decomposition of the HPD matrix A^HA.
On the other hand, let $A^HA = LL^H$ be the (uniquely determined) Cholesky decomposition. Then the matrix $Q := AL^{-H}$ satisfies $Q^HQ = L^{-1}A^HAL^{-H} = I_m$, and hence $A = QL^H$ is the (uniquely determined) QR decomposition of A.

Theorem 1.5 (Schur decomposition)**.** *For every matrix $A \in \mathbb{C}^{n\times n}$ there exist a unitary matrix $U \in \mathbb{C}^{n\times n}$ and an upper triangular matrix $R \in \mathbb{C}^{n\times n}$, such that*

$$A = URU^H,$$

i.e., A can be unitarily triangularized.

Proof. Induction on n. If $n = 1$, set $U = I_1$, $R = A$. Suppose the statement is true for matrices up to order $n-1$ for some $n \geq 2$, and let $A \in \mathbb{C}^{n\times n}$. Suppose that λ is an eigenvalue of A with corresponding unit norm eigenvector x, i.e., $Ax = \lambda x$ with $\|x\|_2^2 = x^Hx = 1$. Let $Y \in \mathbb{C}^{n\times(n-1)}$ be any matrix such that $X := [x, Y] \in \mathbb{C}^{n\times n}$ is unitary. Then

$$X^HAX = \begin{bmatrix} \lambda & x^HAY \\ 0 & Y^HAY \end{bmatrix},$$

where $Y^HAY \in \mathbb{C}^{(n-1)\times(n-1)}$. By the induction hypothesis, there exists a unitary matrix $Z \in \mathbb{C}^{(n-1)\times(n-1)}$ such that $Z^H(Y^HAY)Z = \widetilde{R}$ is upper triangular. Then a straighforward computation shows that

$$A = X \begin{bmatrix} 1 & 0 \\ 0 & Z \end{bmatrix} \begin{bmatrix} \lambda & x^HAY \\ 0 & \widetilde{R} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & Z^H \end{bmatrix} X^H = URU^H,$$

where $U := X \begin{bmatrix} 1 & 0 \\ 0 & Z \end{bmatrix}$ is unitary and $R := \begin{bmatrix} \lambda & x^HAY \\ 0 & \widetilde{R} \end{bmatrix}$ is upper triangular. □

The matrix R in this theorem is called a *Schur form* of A. Its diagonal entries are the eigenvalues of A. As indicated by the proof, they can be chosen in any order. The uniqueness of the strictly upper triangular part of R as well as numerous further results on unitary similarity of matrices are discussed in [34].
The Schur decomposition has been called "[p]erhaps the most fundamentally useful fact of elementary matrix theory" [18, p. 79]. It has the following important corollary, which characterizes some fundamental classes of matrices.

Corollary 1.6 (Spectral decomposition of normal matrices)**.**

(1) $A \in \mathbb{C}^{n\times n}$ is normal if and only if A can be unitarily diagonalized, i.e., there exists a unitary matrix U such that U^HAU is diagonal.

(2) $A \in \mathbb{C}^{n\times n}$ is Hermitian if and only if A can be unitarily diagonalized and all the eigenvalues of A are real.

(3) $A \in \mathbb{C}^{n\times n}$ *is unitary if and only if* A *can be unitarily diagonalized and all eigenvalues* λ *of* A *satisfy* $|\lambda| = 1$.

Proof. (1) Let A be normal and let $A = URU^H$ be a Schur decomposition. Then $A^HA = AA^H$ implies that $R^HR = RR^H$. If we write $R = \begin{bmatrix} R_1 & r \\ 0 & \rho \end{bmatrix}$ with $R_1 \in \mathbb{C}^{(n-1)\times(n-1)}$, we obtain the equality

$$R^HR = \begin{bmatrix} R_1^HR_1 & R_1^Hr \\ r^HR_1 & \|r\|_2^2 + |\rho|^2 \end{bmatrix} = \begin{bmatrix} R_1R_1^H + rr^H & \overline{\rho}r \\ \rho r^H & |\rho|^2 \end{bmatrix} = RR^H.$$

A comparison of the $(2,2)$ entries shows that $r = 0$, and hence

$$R = \begin{bmatrix} R_1 & 0 \\ 0 & \rho \end{bmatrix}, \quad \text{where} \quad R_1^HR_1 = R_1R_1^H.$$

Inductively it follows that R must be diagonal.

On the other hand, if $A = UDU^H$ with U unitary and D diagonal, then

$$A^HA = (UD^HU^H)(UDU^H) = UD^HDU^H = UDD^HU^H = AA^H.$$

(2) If A is Hermitian, then A is normal, and hence $A = UDU^H$ with a diagonal matrix D. Now

$$A = UDU^H = A^H = UD^HU^H$$

shows $D = D^H$.

On the other hand, if $A = UDU^H$ with $D \in \mathbb{R}^{n\times n}$, then $A^H = (UDU^H)^H = UD^HU^H = UDU^H = A$.

(3) If A is unitary, then A is normal, and hence $A = UDU^H$ with a unitary matrix U and a diagonal matrix D. Now

$$I_n = A^HA = (UD^HU^H)(UDU^H) = UD^HDU^H$$

implies $D^HD = I_n$, and thus $|d_{ii}| = 1$.

On the other hand, if $A = UDU^H$ with a unitary matrix U and $D^HD = I_n$, then

$$A^HA = (UD^HU^H)(UDU^H) = I_n,$$

and hence A is unitary.

□

Note that the decomposition $A = UDU^H$ with $U = [u_1, \ldots, u_n]$ and $D = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$ can be written as

$$A = \sum_{j=1}^{n} \lambda_j u_j u_j^H.$$

The matrix $u_j u_j^H$ is Hermitian and satisfies $(u_j u_j^H)^2 = u_j u_j^H$. Thus, a normal matrix can be decomposed into the sum of n rank-one matrices, where each such matrix is an *orthogonal projection* onto the subspace spanned by an eigenvector of A.
A general matrix A can also be decomposed into the sum of rank-one matrices, but in general these matrices are not orthogonal projections onto eigenspaces of A.

Theorem 1.7 (Singular value decomposition, SVD)**.** *If $A \in \mathbb{C}^{n\times m}$ has rank r, then there exist unitary matrices $U \in \mathbb{C}^{n\times n}, V \in \mathbb{C}^{m\times m}$ and a diagonal matrix $\Sigma_+ = \operatorname{diag}(\sigma_1, \ldots, \sigma_r)$ with $\sigma_1 \geq \cdots \geq \sigma_r > 0$, such that*

$$A = U \begin{bmatrix} \Sigma_+ & 0 \\ 0 & 0 \end{bmatrix} V^H, \qquad \text{or} \qquad A = \sum_{j=1}^{r} \sigma_j u_j v_j^H. \tag{1.1}$$

The numbers $\sigma_1 \geq \cdots \geq \sigma_r > 0$ in (1.1) are called the (nonzero) singular values *of A, and the columns of the unitary matrices U and V are called* left *and* right singular vectors *of A, respectively.*

Proof. The matrix $A^H A \in \mathbb{C}^{m\times m}$ is HPSD, since $x^H A^H A x = \|Ax\|_2^2 \geq 0$ for all $x \in \mathbb{C}^m$. Hence $A^H A$ can be unitarily diagonalized with nonnegative real eigenvalues (cf. (1) in Lemma 0.2 and (2) in Corollary 1.6). Denote the $r = \operatorname{rank}(A) = \operatorname{rank}(A^H A)$ positive eigenvalues by $\sigma_1^2 \geq \cdots \geq \sigma_r^2 > 0$ and $\Sigma_+^2 := \operatorname{diag}(\sigma_1^2, \ldots, \sigma_r^2)$, and let the unitary diagonalization be

$$V^H A^H A V = \begin{bmatrix} \Sigma_+^2 & 0 \\ 0 & 0 \end{bmatrix},$$

for a unitary matrix $V \in \mathbb{C}^{m\times m}$. Then with $V = [V_1, V_2]$, where $V_1 \in \mathbb{C}^{m\times r}$, we see that $V_2^H A^H A V_2 = 0$, giving $AV_2 = 0$. Define $U_1 := AV_1\Sigma_+^{-1}$, then

$$U_1^H U_1 = \Sigma_+^{-1} V_1^H A^H A V_1 \Sigma_+^{-1} = I_r.$$

We therefore can choose a matrix U_2 so that $U := [U_1, U_2] \in \mathbb{C}^{n\times n}$ is unitary. Then, by construction, $U_2^H A V_1 = U_2^H U_1 \Sigma_+ = 0$, so that

$$U^H A V = \begin{bmatrix} U_1^H A V_1 & U_1^H A V_2 \\ U_2^H A V_1 & U_2^H A V_2 \end{bmatrix} = \begin{bmatrix} \Sigma_+ & 0 \\ 0 & 0 \end{bmatrix},$$

as required. □

If the matrix A in Theorem 1.7 is real, then we can show with essentially the same proof that the decomposition (1.1) holds with *orthogonal* matrices $U \in \mathbb{R}^{n\times n}$ and $V \in \mathbb{R}^{m\times m}$.

If $A \in \mathbb{C}^{n\times m}$ has an SVD as in (1.1), then the unitary invariance of the 2-norm shows that

$$\|A\|_2 = \left\| \begin{bmatrix} \Sigma_+ & 0 \\ 0 & 0 \end{bmatrix} \right\|_2 = \sigma_1.$$

If $n = m$ and A is nonsingular, then its SVD has the form $A = U\Sigma_+ V^H$, where

$$\Sigma_+ = \operatorname{diag}(\sigma_1, \ldots, \sigma_n) \quad \text{with} \quad \sigma_1 \geq \cdots \geq \sigma_n > 0.$$

Then $A^{-1} = V\Sigma_+^{-1}U^H$, and the unitary invariance of the 2-norm shows that

$$\|A^{-1}\|_2 = \frac{1}{\sigma_n}.$$

The SVD has a long history (see [38]), but its modern form as given in the theorem above appears to be due to Eckart and Young [8]. Because of its practical significance, it has been called the "Swiss Army Knife" as well as the "Rolls Royce" of matrix decompositions [11]. Applications of the SVD will be studied in Chapter 5.
As we have seen in the proof of Theorem 1.7, the singular values are the (positive) square roots of the nonzero eigenvalues of $A^H A$. Thus, the singular values of A are uniquely determined. Similar to eigenvectors, the singular vectors are not uniquely determined. In particular, for any $\phi_1, \ldots, \phi_r \in \mathbb{R}$ we can write

$$A = \sum_{j=1}^{r} \sigma_j u_j v_j^H = \sum_{j=1}^{r} \sigma_j (e^{i\phi_j} u_j)(e^{i\phi_j} v_j)^H,$$

where the sum on the right also is an SVD of A.

Corollary 1.8 (Polar decomposition)**.** *If $A \in \mathbb{C}^{n\times n}$ is nonsingular, there exist unitary matrices $U_1, U_2 \in \mathbb{C}^{n\times n}$ and HPD matrices $H_1, H_2 \in \mathbb{C}^{n\times n}$, such that $A = U_1 H_1 = H_2 U_2$.*

Proof. If A is nonsingular, it has an SVD of the form $A = U\Sigma V^H$ with $U, V \in \mathbb{C}^{n\times n}$ unitary and $\Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_n)$, where $\sigma_i > 0$ for $i = 1, \ldots, n$. Then $A = (UV^H)(V\Sigma V^H) =: U_1 H_1$ with U_1 unitary and H_1 HPD. Similarly, $A = (U\Sigma U^H)(UV^H) =: H_2 U_2$, with H_2 HPD and U_2 unitary. □

This result is a matrix analogue of the polar decomposition of a nonzero complex number $z = e^{i\phi}\rho = \rho e^{i\phi}$, where $\rho = |z| > 0$.

Chapter 2

Perturbation Theory

In this chapter we will give an introduction into the theory of errors in numerical analysis with a focus on numerical linear algebra problems, and into a field that is called *matrix perturbation theory*.

2.1 Errors and the condition number of a problem

Consider a function (or "problem") $f : \mathcal{X} \to \mathcal{Y}$, where $(\mathcal{X}, \|\cdot\|_{\mathcal{X}})$ and $(\mathcal{Y}, \|\cdot\|_{\mathcal{Y}})$ are finite dimensional normed vector spaces[1], called the input (or "data") and output (or "solution") space, respectively. We are interested in studying the behavior of f at a particular input point $x \in \mathcal{X}$. If $\widehat{y}$ is an approximation of $y = f(x)$, for example resulting from a numerical computation, then the accuracy of this approximation can be measured by

$$\begin{aligned} &\|\widehat{y} - y\|_{\mathcal{Y}}, && \text{the } \textit{absolute forward error}, \text{ or} \\ &\frac{\|\widehat{y} - y\|_{\mathcal{Y}}}{\|y\|_{\mathcal{Y}}}, && \text{the } \textit{relative forward error}. \end{aligned}$$

The forward error can be easily interpreted: If it is small, then the approximation $\widehat{y}$ is close to the solution with respect to the given norm (and vice versa). The main problem about the forward error is that it can only be computed when the solution of the given problem is known. This is usually not the case in practical computations.
We can also ask which input for the function f yields the output $\widehat{y}$, i.e., for which *perturbation* Δx of x we have $\widehat{y} = f(x + \Delta x)$. The quantity

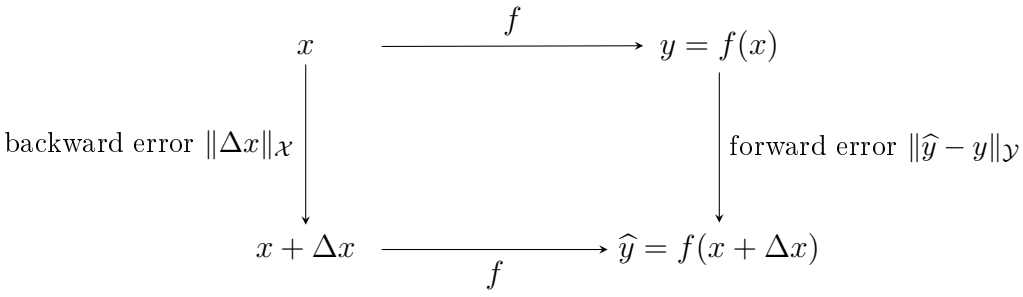
$$\|\Delta x\|_{\mathcal{X}} \text{ is called the } \textit{absolute backward error},$$

and

$$\frac{\|\Delta x\|_{\mathcal{X}}}{\|x\|_{\mathcal{X}}} \text{ is called the } \textit{relative backward error}.$$

This can be illustrated as follows:

[1] The development in this section can be generalized to infinite dimensional normed vector spaces, but the finite dimensional case is easier to treat and sufficient for our purpose.



The essential insight of the backward error idea is the following:

> If an approximation $\widehat{y}$ of the exact solution y has a small backward error, then it is the exact solution of the problem with nearby data.

Frequently in numerical computations the given data results from approximation processes which include some errors, e.g., discretizations, estimations, or measurements. In such cases, rather than solving the problem with the given data exactly (or even very precisely), it is sufficient to obtain a solution with a small backward error. In addition to this strong argument for considering the backward error in numerical computations, it is also important to note that the backward error of an approximate solution can often be computed or estimated a priori without knowing the solution. We will see many examples of this fact in later sections.

Example 2.1. *An important problem that we will study frequently is to solve a linear algebraic system $Ax = b$, where the nonsingular matrix $A \in \mathbb{C}^{n\times n}$ and the right hand side $b \in \mathbb{C}^n$ are given. Then the function we consider is*

$$f : \mathbb{C}^{n\times n} \times \mathbb{C}^n \to \mathbb{C}^n, \quad (A, b) \mapsto f(A, b) := A^{-1}b.$$

Thus, A and b are the input (or data) of the problem, and $x = f(A, b) = A^{-1}b$ is the output (or solution). Note that this common notation in the context of linear algebraic systems is a bit inconsistent with the notation used in the framework above, where x is the input any y is the output.
If $\widehat{x}$ is an approximate solution and $\|\cdot\|$ is a norm on $\mathbb{C}^n$, then $\|\widehat{x} - x\|$ is the forward error. We now ask which perturbed linear algebraic system is solved exactly by the approximate solution, i.e., for which $\Delta A \in \mathbb{C}^{n\times n}$ and $\Delta b \in \mathbb{C}^n$ we have $(A + \Delta A)\widehat{x} = b + \Delta b$, or equivalently $\widehat{x} = f(A + \Delta A, b + \Delta b) = (A + \Delta A)^{-1}(b + \Delta b)$ (if $A + \Delta A$ is nonsingular). This question is studied in detail in Section 2.3 below, and the most general answer about minimum norm perturbations is given in Theorem 2.26.

A function (or problem) is called *well-conditioned* at the input x, when small perturbations of x lead only to small changes in the resulting function values, i.e., a small $\|\Delta x\|_{\mathcal{X}}$ implies a small $\|\widehat{y} - y\|_{\mathcal{Y}}$. Here the word "small" needs to be interpreted in the given context. A function that is not well-conditioned at x is called *ill-conditioned* at x.

How can we determine whether a function is well-conditioned? For a motivation we consider a twice continuously differentiable function $f : \mathbb{R} \to \mathbb{R}$. For a given $x \in \mathbb{R}$ let $y = f(x)$ and suppose that $\widehat{y} = f(x + \Delta x)$. Then by Taylor's theorem

$$\widehat{y} - y = f(x + \Delta x) - f(x) = f'(x)\Delta x + O(|\Delta x|^2),$$

if $|\Delta x|$ is small enough, giving

$$|\widehat{y} - y| = |f'(x)||\Delta x| + O(|\Delta x|^2).$$

The quantity $|f'(x)|$ measures, for small $|\Delta x|$, the absolute change in the output for a given absolute change in the input. It therefore can be considered the *absolute condition number* of f at x. If $y = f(x) \neq 0$ we can divide the previous equation by $f(x)$ and write the result in the form

$$\left|\frac{\widehat{y} - y}{y}\right| = \left|f'(x)\frac{x}{f(x)}\right| \left|\frac{\Delta x}{x}\right| + O(|\Delta x|^2). \tag{2.1}$$

Here $|f'(x)x/f(x)|$ measures, for small $|\Delta x|$, the relative change in the output for a given relative change in the input. Hence $|f'(x)x/f(x)|$ is the *relative condition number* of f at x, and the equation (2.1) can be read as

$$\text{(relative) forward error} \lesssim \text{(relative) condition number} \times \text{(relative) backward error.} \tag{2.2}$$

This *rule of thumb* will appear frequently below.

We now generalize the concept of the relative condition number to a function between two finite dimensional normed vector spaces.

Definition 2.2. *Let $\mathcal{X}$ and $\mathcal{Y}$ be finite dimensional normed vector spaces. The* relative condition number *of a function $f : \mathcal{X} \to \mathcal{Y}$ at $x \in \mathcal{X}$ with $f(x) \neq 0$ is defined by*

$$\kappa_f(x) := \lim_{\delta \to 0} \sup_{\|\Delta x\|_{\mathcal{X}} \leq \delta} \frac{\|f(x + \Delta x) - f(x)\|_{\mathcal{Y}}}{\|f(x)\|_{\mathcal{Y}}} \frac{\|x\|_{\mathcal{X}}}{\|\Delta x\|_{\mathcal{X}}}. \tag{2.3}$$

Note that the first factor in the definition of $\kappa_f(x)$ in (2.3) is the relative forward error, and the second factor is the reciprocal of the relative backward error. Thus, the rule of thumb (2.2) is naturally built into the definition of the condition number. The relative forward error is not defined at $x \in \mathcal{X}$ with $f(x) = 0$ because we would then have a division by zero. (Similarly, the relative backward error does not exist for $x = 0$.) We will discuss below that the condition number $\kappa_f(x)$ may still be defined for such x.
A function $f : \mathcal{X} \to \mathcal{Y}$ is called *Fréchet differentiable at $x \in \mathcal{X}$*, if there exists a linear map $L_x : \mathcal{X} \to \mathcal{Y}$ such that

$$\lim_{\|\Delta x\|_{\mathcal{X}} \to 0} \frac{\|f(x + \Delta x) - f(x) - L_x(\Delta x)\|_{\mathcal{Y}}}{\|\Delta x\|_{\mathcal{X}}} = 0.$$

It can be shown that if such a map L_x exists, then it is uniquely determined. We then call this function the *Fréchet derivative of f at x*, and write $f'(x) := L_x$. For a differentiable function $f : \mathbb{R} \to \mathbb{R}$ this definition reduces to the usual derivative $f'(x)$.

If $f : \mathcal{X} \to \mathcal{Y}$ is linear we can choose $L_x = f$ for every $x \in \mathcal{X}$, which yields

$$\frac{\|f(x+\Delta x) - f(x) - L_x(\Delta x)\|_{\mathcal{Y}}}{\|\Delta x\|_{\mathcal{X}}} = \frac{\|f(x) + f(\Delta x) - f(x) - f(\Delta x)\|_{\mathcal{Y}}}{\|\Delta x\|_{\mathcal{X}}} = 0.$$

Hence in this case the unique Fréchet derivative of f is $f'(x) = f$ for every $x \in \mathcal{X}$. Since $\mathcal{X}$ and $\mathcal{Y}$ are finite dimensional, the linear map f can be represented by a matrix M (depending on f and the choice of bases of $\mathcal{X}$ and $\mathcal{Y}$), and this justifies to write $f'(x) = M$.

In general, f is Fréchet differentiable at x and $f(x) \neq 0$, then

$$\begin{aligned}
\kappa_f(x) &= \lim_{\delta \to 0} \sup_{\|\Delta x\|_{\mathcal{X}} \leq \delta} \frac{\|f(x+\Delta x) - f(x)\|_{\mathcal{Y}}}{\|f(x)\|_{\mathcal{Y}}} \frac{\|x\|_{\mathcal{X}}}{\|\Delta x\|_{\mathcal{X}}} \\
&= \left(\lim_{\delta \to 0} \sup_{\|\Delta x\|_{\mathcal{X}} \leq \delta} \frac{\|f(x+\Delta x) - f(x)\|_{\mathcal{Y}}}{\|\Delta x\|_{\mathcal{X}}} \right) \frac{\|x\|_{\mathcal{X}}}{\|f(x)\|_{\mathcal{Y}}} \\
&= \|f'(x)\|_* \frac{\|x\|_{\mathcal{X}}}{\|f(x)\|_{\mathcal{Y}}} \qquad \text{(cf. (2.1))}, \qquad (2.4)
\end{aligned}$$

where the norm $\|\cdot\|_*$ is induced by the norms on $\mathcal{X}$ and $\mathcal{Y}$, i.e.,

$$\|f'(x)\|_* = \max_{\|z\|_{\mathcal{X}}=1} \|(f'(x))(z)\|_{\mathcal{Y}}.$$

Using (2.4) we can extend the definition of the condition number to $x_0 \in \mathcal{X}$ with $f(x_0) = 0$ if the limit for $x \to x_0$ exists. In this case

$$\kappa_f(x_0) = \lim_{x \to x_0} \|f'(x)\|_* \frac{\|x\|_{\mathcal{X}}}{\|f(x)\|_{\mathcal{Y}}}.$$

Example 2.3. *For the function $f : \mathbb{R} \to \mathbb{R}$ with $f(x) = \alpha x$ for some nonzero $\alpha \in \mathbb{R}$ and the norm $\|\cdot\| = |\cdot|$ (absolute value) on $\mathbb{R}$ we obtain from (2.4) that*

$$\kappa_f(x) = |f'(x)| \frac{|x|}{|f(x)|} = |\alpha| \frac{|x|}{|\alpha x|} = 1 \quad \text{for all } x \in \mathbb{R} \setminus \{0\}.$$

For $x = 0$ we can use $\kappa_f(0) = \lim_{x \to 0} |\alpha x| / |\alpha x| = 1$.

Example 2.4. *For the function $f : \mathbb{R}_+ \to \mathbb{R}$ with $f(x) = \log(x)$ and the norm $\|\cdot\| = |\cdot|$ (absolute value) on $\mathbb{R}$ we obtain from (2.4) that*

$$\kappa_f(x) = |f'(x)| \frac{|x|}{|f(x)|} = \left| \frac{1}{x} \right| \frac{|x|}{|\log(x)|} = \frac{1}{|\log(x)|} \quad \text{for all } x \in \mathbb{R}_+ \setminus \{1\}.$$

Here $\lim_{x \to 1} |\log(x)|^{-1} = +\infty$, and we write $\kappa_f(1) = +\infty$. Clearly, the function f is ill-conditioned in the neighborhood of $x = 1$. In order to illustrate the ill-conditioning

numerically, let

$$x = 1 + 10^{-8}, \quad \Delta x = 10^{-10}.$$

An evaluation in MATLAB gives

$$\begin{aligned} \log(x) &= 9.999999889225291 \times 10^{-9}, \\ \log(x + \Delta x) &= 1.009999989649433 \times 10^{-8}, \end{aligned}$$

and hence

$$\begin{aligned} \frac{|\Delta x|}{|x|} &= 9.999999900000002 \times 10^{-11}, \\ \kappa_{\log}(x) &= 1.000000011077471 \times 10^{8}, \\ \frac{|\log(x + \Delta x) - \log(x)|}{|\log(x)|} &= 1.000000083767846 \times 10^{-2}. \end{aligned}$$

We observe that due to the large condition number of f at $x \approx 1$, a small relative perturbation of the input (or a small relative backward error) leads to a large change in the output (or a large relative forward error).

Example 2.5. *Consider integers $1 \leq k \leq n$ and an inner product $\langle \cdot, \cdot \rangle$ on $\mathbb{C}^n$. Let $v_1, \ldots, v_k \in \mathbb{C}^n$ be orthonormal vectors with respect to this inner product, i.e., $\langle v_i, v_j \rangle = \delta_{ij}$, and define $\mathcal{V}_k := \operatorname{span}\{v_1, \ldots, v_k\}$ and $V_k = [v_1, \ldots, v_k] \in \mathbb{C}^{n \times k}$. Let $\| \cdot \| = \langle \cdot, \cdot \rangle^{1/2}$ be the induced norm on $\mathcal{V}_k \subseteq \mathbb{C}^n$; cf. Lemma 0.8.*
We consider the function $f : \mathbb{C}^k \to \mathcal{V}_k$ with $f(x) = V_k x$ and the 2-norm $\| \cdot \|_2$ on $\mathbb{C}^k$. Then for every $x \in \mathbb{C}^k \setminus \{0\}$ Definition 2.2 yields

$$\kappa_f(x) = \lim_{\delta \to 0} \sup_{\|\Delta x\|_2 \leq \delta} \frac{\|f(x + \Delta x) - f(x)\|}{\|f(x)\|} \frac{\|x\|_2}{\|\Delta x\|_2} = \lim_{\delta \to 0} \sup_{\|\Delta x\|_2 \leq \delta} \frac{\|V_k \Delta x\|}{\|V_k x\|} \frac{\|x\|_2}{\|\Delta x\|_2}.$$

Since the vectors $v_1, \ldots, v_k$ are orthonormal we obtain, for every $x = [\xi_1, \ldots, \xi_k]^T \in \mathbb{C}^k$,

$$\|V_k x\| = \Big\langle \sum_{i=1}^{k} \xi_i v_i, \sum_{j=1}^{k} \xi_j v_j \Big\rangle^{1/2} = \left(\sum_{i=1}^{k} \sum_{j=1}^{k} \xi_i \overline{\xi}_j \langle v_i, v_j \rangle \right)^{1/2} = \|x\|_2,$$

which shows that $\kappa_f(x) = 1$ for every $x \in \mathbb{C}^k \setminus \{0\}$.
The same result can be obtained using the Frechét derivative and (2.4). Since f is linear we have $f'(x) = f$ for every $x \in \mathbb{C}^k$. Using the norm $\| \cdot \|_*$ that is induced by

the norms $\|\cdot\|_2$ *on* $\mathbb{C}^k$ *and* $\|\cdot\|$ *on* $\mathcal{V}_k$ *we obtain*

$$\|f'(x)\|_* = \max_{\substack{z\in\mathbb{C}^k\\ \|z\|_2=1}} \|f(z)\| = \max_{\substack{z\in\mathbb{C}^k\\ \|z\|_2=1}} \|V_k z\| = \max_{\substack{z\in\mathbb{C}^k\\ \|z\|_2=1}} \|z\|_2 = 1,$$
$$\|f(x)\| = \|V_k x\| = \|x\|_2,$$

and therefore

$$\kappa_f(x) = \|f'(x)\|_* \frac{\|x\|_2}{\|f(x)\|} = 1\frac{\|x\|_2}{\|x\|_2} = 1,$$

which holds for every $x \in \mathbb{C}^k$.

Example 2.6. *For a given matrix* $A \in \mathbb{C}^{n\times n}$ *we consider the function* $f : \mathbb{C}^n \to \mathbb{C}^n$ *with* $f(x) = Ax$*, which is Frechét differentiable at every* $x \in \mathbb{C}^n$ *with* $f'(x) = f$*.*
Let $\|\cdot\|$ *be any given norm on* $\mathbb{C}^n$ *as well as the induced matrix norm on* $\mathbb{C}^{n\times n}$*; cf. Lemma 0.5. Using the norm* $\|\cdot\|_*$ *induced by these two norms yields*

$$\|f'(x)\|_* = \max_{\substack{z\in\mathbb{C}^n\\ \|z\|=1}} \|f(z)\| = \max_{\substack{z\in\mathbb{C}^n\\ \|z\|=1}} \|Az\| = \|A\|,$$

for every $x \in \mathbb{C}^n$*. From* (2.4) *we thus obtain*

$$\kappa_f(x) = \|f'(x)\|_* \frac{\|x\|}{\|Ax\|} = \|A\| \frac{\|x\|}{\|Ax\|} \quad \text{for every } x \notin \ker(A) := \{y \in \mathbb{C}^n \,:\, Ay = 0\},$$

where $\ker(A)$ *is called the* kernel *(or* null space*) of* A*.*
If A *is nonsingular, then* $\ker(A) = \{0\}$ *and*

$$\max_{x\in\mathbb{C}^n\setminus\{0\}} \frac{\|x\|}{\|Ax\|} = \max_{x\in\mathbb{C}^n\setminus\{0\}} \frac{\|A^{-1}Ax\|}{\|Ax\|} = \max_{x\in\mathbb{C}^n\setminus\{0\}} \left\| A^{-1}\frac{Ax}{\|Ax\|} \right\| = \max_{\substack{z\in\mathbb{C}^n\\ \|z\|=1}} \|A^{-1}z\| = \|A^{-1}\|,$$

which shows that $\max_{x\in\mathbb{C}^n\setminus\{0\}} \kappa_f(x) = \|A\|\,\|A^{-1}\|$.

Example 2.6 motivates the following definition.

Definition 2.7. *If* $A \in \mathbb{C}^{n\times n}$ *is nonsingular, and* $\|\cdot\|$ *is a norm on* $\mathbb{C}^{n\times n}$*, then*

$$\kappa(A) := \|A\|\,\|A^{-1}\|$$

is called the condition number *of* A *with respect to the norm* $\|\cdot\|$.

Note that $\kappa(A)$ is an upper bound on $\kappa_f(x)$ for the function $f(x) = Ax$ over all nonzero $x \in \mathbb{C}^n$. Thus it represents a *worst case*, and it may happen that $\kappa_f(x) \ll \kappa(A)$ for some x.

For any nonsingular $A \in \mathbb{C}^{n\times n}$ and consistent norm $\|\cdot\|$ we have

$$\|I\| = \|AA^{-1}\| \le \|A\|\|A^{-1}\| = \kappa(A).$$

If A has a "large" condition number, i.e., $\kappa(A) \gg \|I\|$, then A is called *ill-conditioned*, and A is called *well-conditioned* when it has a "small" condition number, i.e., $\kappa(A) \approx \|I\|$.
The terms ill- and well-conditioning are defined rather vaguely, since the meaning of "$\gg$" and "$\approx$" depends on the context. In Example 2.18 below we will illustrate the rule of thumb that a condition number $\kappa(A) \approx 10^k$ may lead to a loss of k significant digits in an approximate solution of a linear algebraic system $Ax = b$ that is computed in finite precision arithmetic. Thus, if we compute in IEEE double precision and our floating point numbers have 16 significant digits (see Example 3.2), and our goal is to solve $Ax = b$ up to an accuracy of 8 significant digits, then matrices with a condition number of 10^7 or even 10^8 may still be "sufficiently well-conditioned" in our context.

Example 2.8. *Let $A \in \mathbb{C}^{n\times n}$ be nonsingular, and let $A = U\Sigma_+V^H$ be an SVD in the notation of Theorem 1.7. Then $\kappa_2(A) = \|A\|_2\,\|A^{-1}\|_2 = \sigma_1/\sigma_n$.*
We have $A^H = V\Sigma_+U^H$ and $A^{-H} = U\Sigma_+^{-1}V^H$, which shows that $\kappa_2(A^H) = \kappa_2(A)$ and

$$\|A^HA\|_2 = \|AA^H\|_2 = \|\Sigma_+^2\|_2 = \sigma_1^2,$$
$$\|(A^HA)^{-1}\|_2 = \|(AA^H)^{-1}\|_2 = \|\Sigma_+^{-2}\|_2 = \frac{1}{\sigma_n^2},$$
$$\kappa_2(A^HA) = \kappa_2(AA^H) = \frac{\sigma_1^2}{\sigma_n^2} = \kappa_2(A)^2.$$

The 2-norm condition number can be generalized to non-square matrices which have full rank. If $A \in \mathbb{C}^{n\times m}$ has (full) rank $m \le n$, then A has an SVD of the form

$$A = U\begin{bmatrix}\Sigma_+\\ 0\end{bmatrix}V^H, \quad \Sigma_+ = \operatorname{diag}(\sigma_1,\ldots,\sigma_m),$$

where $\sigma_1 \ge \cdots \ge \sigma_m > 0$, and we define $\kappa_2(A) := \sigma_1/\sigma_m$. This is consistent with the original definition for the case $m = n$. Note that

$$A^HA = V\Sigma_+^2V^H,$$

and hence $\kappa_2(A^HA) = \sigma_1^2/\sigma_m^2 = \kappa_2(A)^2$.
If $A \in \mathbb{C}^{n\times m}$ has (full) rank $n \le m$, then A has an SVD of the form

$$A = U[\Sigma_+\ 0]V^H, \quad \Sigma_+ = \operatorname{diag}(\sigma_1,\ldots,\sigma_n),$$

where $\sigma_1 \ge \cdots \ge \sigma_n > 0$, and we define $\kappa_2(A) := \sigma_1/\sigma_n$. Again this is consistent with the original definition for the case $m = n$. Now we have

$$AA^H = U\Sigma_+^2U^H,$$

and hence $\kappa_2(AA^H) = \sigma_1^2/\sigma_n^2 = \kappa_2(A)^2$.

It is important to note that ill conditioning may not be related to the distribution of the eigenvalues of A, which is illustrated by the following example.

Example 2.9. *A classical example due to Wilkinson is given by the nonsingular matrix*

$$A = \begin{bmatrix} 0.501 & 1 & & & & \\ & 0.502 & 1 & & & \\ & & \ddots & \ddots & & \\ & & & & 0.599 & 1 \\ & & & & & 0.600 \end{bmatrix} \in \mathbb{R}^{100\times 100},$$

which has equidistantly distributed eigenvalues in the real interval $[0.501, 0.6]$*, and the 2-norm condtition number* $\kappa_2(A) \approx 2.2 \times 10^{26}$*. Here* $\|A\|_2 \approx 1.6$*, so that the large condition number is caused by* $\|A^{-1}\|_2 \approx 10^{27}$*, and hence by the smallest singular value of* A*.*
Recall that we have defined the condition number of a function locally, and that $\|A\|\|A^{-1}\|$ *forms a global upper bound on the condition numbers* $\kappa_f(x)$ *and* $\kappa_g(x)$*, where* $f(x) = Ax$ *and* $g(x) = A^{-1}x$*; see the derivation in Example 2.6. In particular, we have* $\kappa_g(x) = \|A^{-1}\|\|x\|/\|A^{-1}x\|$*, and evaluating this expression with the Wilkinson matrix* A *and the 2-norm for the first and last unit basis vector in MATLAB gives*

$$\kappa_g(e_1) \approx 6.97 \times 10^{25} \quad \textit{and} \quad \kappa_g(e_n) \approx 1.2493.$$

Thus, the function g *is ill-conditioned at* $x = e_1$*, and well-conditioned at* $x = e_n$*.*

The condition number of matrices occurs naturally in many perturbation results. We give a few examples in the following section.

2.2 Perturbation results for matrices

We consider the function f from (a dense subset of) $\mathbb{C}^{n\times n}$ to $\mathbb{C}^{n\times n}$ defined by $f(A) = A^{-1}$, and we are interested in bounds on the norm of the forward error when we perturb the input of f. Thus, we consider a perturbed matrix $\widehat{A} = A + E$ and try to bound $\|f(A+E) - f(A)\| = \|\widehat{A}^{-1} - A^{-1}\|$.

Theorem 2.10. *Let* $A \in \mathbb{C}^{n\times n}$ *and* $\widehat{A} = A + E \in \mathbb{C}^{n\times n}$ *be nonsingular. Then for any consistent norm* $\|\cdot\|$ *on* $\mathbb{C}^{n\times n}$ *we have*

$$\frac{\|\widehat{A}^{-1} - A^{-1}\|}{\|\widehat{A}^{-1}\|} \le \|A^{-1}E\| \le \kappa(A)\frac{\|E\|}{\|A\|}. \tag{2.5}$$

Moreover, if $\kappa(A)\frac{\|E\|}{\|A\|} < 1$*, then*

$$\frac{\|\widehat{A}^{-1} - A^{-1}\|}{\|A^{-1}\|} \le \frac{\kappa(A)\frac{\|E\|}{\|A\|}}{1 - \kappa(A)\frac{\|E\|}{\|A\|}}. \tag{2.6}$$

Proof. Using $A^{-1}\widehat{A} = I_n + A^{-1}E$ and $\widehat{A}^{-1} - A^{-1} = (I_n - A^{-1}\widehat{A})\widehat{A}^{-1} = -A^{-1}E\widehat{A}^{-1}$ we obtain

$$\|\widehat{A}^{-1} - A^{-1}\| \leq \|A^{-1}E\|\|\widehat{A}^{-1}\| \leq \|A^{-1}\|\|A\|\frac{\|E\|}{\|A\|}\|\widehat{A}^{-1}\|.$$

Dividing by $\|\widehat{A}^{-1}\|$ yields the two inequalities in (2.5).
In order to show (2.6), we start with $\widehat{A}^{-1} = A^{-1} - A^{-1}E\widehat{A}^{-1}$, which yields

$$\|\widehat{A}^{-1}\| \leq \|A^{-1}\| + \|A^{-1}\|\|E\|\|\widehat{A}^{-1}\| = \|A^{-1}\| + \kappa(A)\frac{\|E\|}{\|A\|}\|\widehat{A}^{-1}\|,$$

and hence

$$\frac{\|\widehat{A}^{-1}\|}{\|A^{-1}\|} \leq \frac{1}{1 - \kappa(A)\frac{\|E\|}{\|A\|}},$$

where we have used that $\kappa(A)\frac{\|E\|}{\|A\|} < 1$. Rewriting (2.5) in the form

$$\frac{\|\widehat{A}^{-1} - A^{-1}\|}{\|A^{-1}\|} \leq \frac{\|\widehat{A}^{-1}\|}{\|A^{-1}\|}\kappa(A)\frac{\|E\|}{\|A\|},$$

and using the previous inequality yields (2.6). □

Note that if $\kappa(A)\frac{\|E\|}{\|A\|} \ll 1$, then

$$\frac{\|\widehat{A}^{-1} - A^{-1}\|}{\|A^{-1}\|} \leq \frac{\kappa(A)\frac{\|E\|}{\|A\|}}{1 - \kappa(A)\frac{\|E\|}{\|A\|}} = \kappa(A)\frac{\|E\|}{\|A\|} + O\Big(\Big(\kappa(A)\frac{\|E\|}{\|A\|}\Big)^2\Big),$$

which is another instance of our rule of thumb (2.2).

Example 2.11. *For any real or complex $\varepsilon \neq 0$ the matrix*

$$A = \begin{bmatrix} 1+\varepsilon & 1 \\ 1 & 1 \end{bmatrix},$$

is nonsingular with its inverse given by

$$A^{-1} = \begin{bmatrix} 1/\varepsilon & -1/\varepsilon \\ -1/\varepsilon & 1+1/\varepsilon \end{bmatrix}.$$

A MATLAB computation with

$$\varepsilon = 10^{-6}, \quad E = \begin{bmatrix} 10^{-7} & 0 \\ 0 & 0 \end{bmatrix}, \quad \widehat{A} = A + E,$$

and the matrix 2-norm yields

$$\begin{aligned}
\frac{\|\widehat{A}^{-1} - A^{-1}\|_2}{\|\widehat{A}^{-1}\|_2} &= 9.999997250000006 \times 10^{-2},\\
\kappa_2(A)\frac{\|E\|_2}{\|A\|_2} &= 2.000000500000125 \times 10^{-1},\\
\|A^{-1}E\|_2 &= 1.414213562373095 \times 10^{-1},\\
\frac{\|\widehat{A}^{-1} - A^{-1}\|_2}{\|A^{-1}\|_2} &= 9.090906818181821 \times 10^{-2},\\
\frac{\kappa_2(A)\frac{\|E\|_2}{\|A\|_2}}{1 - \kappa_2(A)\frac{\|E\|_2}{\|A\|_2}} &= 2.500000781250244 \times 10^{-1}.
\end{aligned}$$

We observe that the two bounds (2.5) *and* (2.6) *are quite tight in this case.*

We will next show how the condition number of a nonsingular matrix is related to the "distance to singularity" of the matrix.

Lemma 2.12. *If* $\|\cdot\|$ *is a consistent norm on* $\mathbb{C}^{n\times n}$*, then for each* $A \in \mathbb{C}^{n\times n}$ *we have*

$$\|A\| \geq \rho(A),$$

where $\rho(A) := \max\{\,|\lambda| : \lambda$ *is an eigenvalue of* $A\}$ *is called the* spectral radius *of* A*.*

Proof. We have shown that for any consistent norm $\|\cdot\|$ on $\mathbb{C}^{n\times n}$ there exists a norm $\|\cdot\|_*$ on $\mathbb{C}^n$ so that $\|\cdot\|$ and $\|\cdot\|_*$ are consistent; see Lemma 0.6 and the comments below that result. If $Ax = \lambda x$, $x \neq 0$, then

$$|\lambda|\|x\|_* = \|\lambda x\|_* = \|Ax\|_* \leq \|A\|\|x\|_*,$$

and hence $|\lambda| \leq \|A\|$. □

The lower bound in Lemma 2.12 is attained for the 2-norm on $\mathbb{C}^{n\times n}$ and any normal matrix $A \in \mathbb{C}^{n\times n}$. This can be seen from the unitary diagonalization $A = UDU^H$, where $D = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$, and the unitary invariance of the 2-norm, which gives

$$\|A\|_2 = \|UDU^H\|_2 = \|D\|_2 = \max_{1\leq j\leq n} |\lambda_j| = \rho(A).$$

Theorem 2.13. *If* $A \in \mathbb{C}^{n\times n}$ *is nonsingular and* $E \in \mathbb{C}^{n\times n}$ *is such that* $A + E \in \mathbb{C}^{n\times n}$ *is singular, then for any consistent norm* $\|\cdot\|$ *on* $\mathbb{C}^{n\times n}$ *we have*

$$\frac{\|E\|}{\|A\|} \geq \frac{1}{\kappa(A)}.$$

Proof. Since A is nonsingular we can write $A + E = A(I_n + A^{-1}E)$. Since $A + E$ is singular, the matrix $I_n + A^{-1}E$ must be singular, and thus -1 must be an eigenvalue of $A^{-1}E$. Lemma 2.12 then gives

$$1 \leq \rho(A^{-1}E) \leq \|A^{-1}E\| \leq \|A^{-1}\|\|A\|\frac{\|E\|}{\|A\|},$$

which yields the desired inequality. □

This theorem shows that "well-conditioned matrices are far from singular", since the norm of a perturbation E (relative to the norm of A) which makes a nonsingular matrix singular must be at least $1/\kappa(A)$.

Example 2.14. *Let $A \in \mathbb{C}^{n\times n}$ be nonsingular, and let $A = U\Sigma_+ V^H$ be an SVD in the notation of Theorem 1.7. For the matrix $E := -\sigma_n u_n v_n^H$ we have $\|E\|_2 = \sigma_n$, and*

$$A + E = \sum_{j=1}^{n-1} \sigma_j u_j v_j^H = U \begin{bmatrix} \sigma_1 & & & \\ & \ddots & & \\ & & \sigma_{n-1} & \\ & & & 0 \end{bmatrix} V^H$$

has rank $n-1$, and thus is singular. Moreover,

$$\frac{\|E\|_2}{\|A\|_2} = \frac{\sigma_n}{\sigma_1} = \frac{1}{\kappa_2(A)},$$

and therefore E is a perturbation with minimal (relative) 2-norm such that the perturbed matrix is singular. Note that for the matrix A in Example 2.9 we have $\|E\|_2 \approx 10^{-27}$.

We end this section with a perturbation result for the LU decomposition that is taken (with small modifications) from [2]. The matrix U in this theorem corresponds to the matrix DU in Theorem 1.1.

Theorem 2.15. *Suppose that $A \in \mathbb{C}^{n\times n}$ has the LU decomposition $A = LU$, where $L \in \mathbb{C}^{n\times n}$ is unit lower triangular, and $U \in \mathbb{C}^{n\times n}$ is nonsingular and upper triangular. Furthermore, suppose that $\Delta A \in \mathbb{C}^{n\times n}$ and that the perturbed matrix $A + \Delta A$ has the LU decomposition*

$$A + \Delta A = (L + \Delta L)(U + \Delta U),$$

where $L + \Delta L$ is unit lower triangular, and $U + \Delta U$ is nonsingular and upper triangular. If $\|L^{-1}\Delta A U^{-1}\|_F < 1$, then

$$\max\left\{\frac{\|\Delta L\|_F}{\|L\|_F}, \frac{\|\Delta U\|_F}{\|U\|_F}\right\} \leq \frac{\|L^{-1}\Delta A U^{-1}\|_F}{1 - \|L^{-1}\Delta A U^{-1}\|_F}.$$

Proof. First note that

$$A + \Delta A = LU + \Delta A = (L + \Delta L)(U + \Delta U) = LU + L\Delta U + \Delta L(U + \Delta U),$$

which gives

$$L\Delta U + \Delta L(U + \Delta U) = \Delta A.$$

Multiplying from the left with L^{-1} and from the right with $(U + \Delta U)^{-1}$ yields

$$\Delta U(U + \Delta U)^{-1} + L^{-1}\Delta L = L^{-1}\Delta A(U + \Delta U)^{-1}.$$

Now note that since both L and $L + \Delta L$ are unit lower triangular, the matrix ΔL must be strictly lower triangular (i.e., it is lower triangular with a has a zero diagonal). Therefore $L^{-1}\Delta L$ also is strictly lower triangular, and we have

$$\|\Delta U(U + \Delta U)^{-1}\|_F^2 + \|L^{-1}\Delta L\|_F^2 = \|L^{-1}\Delta A(U + \Delta U)^{-1}\|_F^2.$$

In particular,

$$\begin{aligned} \|\Delta U(U + \Delta U)^{-1}\|_F &\leq \|L^{-1}\Delta A(U + \Delta U)^{-1}\|_F, \\ \|L^{-1}\Delta L\|_F &\leq \|L^{-1}\Delta A(U + \Delta U)^{-1}\|_F. \end{aligned}$$

Moreover, $\|\Delta L\|_F = \|LL^{-1}\Delta L\|_F \leq \|L\|_F\|L^{-1}\Delta L\|_F$, which yields

$$\frac{\|\Delta L\|_F}{\|L\|_F} \leq \|L^{-1}\Delta L\|_F \leq \|L^{-1}\Delta A(U + \Delta U)^{-1}\|_F.$$

Next, we use $(U + \Delta U)^{-1} = U^{-1} - U^{-1}\Delta U(U + \Delta U)^{-1}$, which can be verified with a straightforward computation, and obtain

$$\begin{aligned} \|L^{-1}\Delta A(U + \Delta U)^{-1}\|_F &= \|L^{-1}\Delta AU^{-1} - L^{-1}\Delta AU^{-1}\Delta U(U + \Delta U)^{-1}\|_F \\ &\leq \|L^{-1}\Delta AU^{-1}\|_F + \|L^{-1}\Delta AU^{-1}\|_F\|\Delta U(U + \Delta U)^{-1}\|_F \\ &\leq \|L^{-1}\Delta AU^{-1}\|_F + \|L^{-1}\Delta AU^{-1}\|_F\|L^{-1}\Delta A(U + \Delta U)^{-1}\|_F. \end{aligned}$$

Since $\|L^{-1}\Delta AU^{-1}\|_F < 1$, this can be rewritten as

$$\|L^{-1}\Delta A(U + \Delta U)^{-1}\|_F \leq \frac{\|L^{-1}\Delta AU^{-1}\|_F}{1 - \|L^{-1}\Delta AU^{-1}\|_F}.$$

The proof that the same bound holds for $\|\Delta U\|_F/\|U\|_F$ is analogous. □

2.3 Perturbation results for linear algebraic systems

In the next result the vector $\widehat{x}$ should be interpreted as an approximate solution of the given linear algebraic system.

Theorem 2.16. *Let $A \in \mathbb{C}^{n\times n}$ be nonsingular, $x \in \mathbb{C}^n \setminus \{0\}$ and $b = Ax$. Then for consistent norms and every $\widehat{x} \in \mathbb{C}^n$ we have*

$$\frac{\|\widehat{x} - x\|}{\|x\|} \leq \kappa(A)\frac{\|r\|}{\|b\|}, \tag{2.7}$$

where $r := b - A\widehat{x}$ is the residual *and $\|r\|/\|b\|$ is the* relative residual norm.

Proof. Using $x = A^{-1}b$ and the definition of the residual we get

$$\|\widehat{x} - x\| = \|A^{-1}(A\widehat{x} - b)\| \leq \|A^{-1}\|\|r\| = \kappa(A)\frac{\|r\|}{\|A\|}.$$

Moreover, $\|b\| \leq \|A\|\|x\|$ gives

$$\frac{1}{\|x\|} \leq \frac{\|A\|}{\|b\|},$$

which implies the desired inequality. □

We call the inequality (2.7) a *residual-based forward error bound*. It implies that for an ill-conditioned matrix A a small relative residual norm does not guarantee a small relative forward error. And, vice versa, if A is well-conditioned and the relative residual norm is small, then the relative forward error is small as well.

Example 2.17. *We set up a* 12×12 Vandermonde matrix A *in MATLAB, define the vector $x = [1, \ldots, 1]^T$ (the solution) and use it to compute the right hand side as $b = Ax$. The matrix A is very ill-conditioned, $\kappa_2(A) \approx 2.1 \times 10^{24}$, and MATLAB's backslash computes an approximate solution of $Ax = b$ with a large relative forward error, but a small relative residual:*

```
>> v=[1:12].^2; A=vander(v); x=ones(12,1); b=A*x; xhat=A\b;
>> norm(x-xhat)/norm(x)
ans =
   25.2364
>> norm(b-A*xhat)/norm(b)
ans =
   2.8482e-23
```

Example 2.18. *We consider*

$$A = \begin{bmatrix} 1+\varepsilon & 1 \\ 1 & 1 \end{bmatrix}, \quad x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad b = Ax = \begin{bmatrix} 2+\varepsilon \\ 2 \end{bmatrix}, \quad \widehat{x} = \begin{bmatrix} 0 \\ 2 \end{bmatrix};$$

cf. Example 2.11. Then

$$r = b - A\widehat{x} = \begin{bmatrix} \varepsilon \\ 0 \end{bmatrix}, \quad \|r\|_2 = |\varepsilon|, \quad \textit{while} \quad \frac{\|\widehat{x} - x\|_2}{\|x\|_2} = 1.$$

For $\varepsilon = 10^{-6}$ *we have* $\kappa_2(A) \approx 4 \times 10^6$.
If we try to solve $Ax = b$ *using* `xhat=inv(A)*b` *in MATLAB, we get the computed approximation*

$$\widehat{x} = \begin{bmatrix} 1.000000000232831 \\ 0.999999999767169 \end{bmatrix} \quad \textit{with the relative forward error} \quad \frac{\|\widehat{x} - x\|_2}{\|x\|_2} \approx 2.33 \times 10^{-10}.$$

Since the machine epsilon is $\epsilon_M \approx 2.22 \times 10^{-16}$ *(see Example 3.2), we have lost six significant digits in the computed solution. Similarly, if we compute* `[L,U]=lu(A)` *and then* `xhat=U\(L\b)`*, we obtain the computed approximation*

$$\widehat{x} = \begin{bmatrix} 1.000000000200923 \\ 0.999999999799077 \end{bmatrix} \quad \textit{with the relative forward error} \quad \frac{\|\widehat{x} - x\|_2}{\|x\|_2} \approx 2.01 \times 10^{-10}.$$

Finally, with `[Q,R]=qr(A)` *and* `xhat=R\(Q'*b)` *we obtain*

$$\widehat{x} = \begin{bmatrix} 1.000000000301700 \\ 0.999999999698300 \end{bmatrix} \quad \textit{with the relative forward error} \quad \frac{\|\widehat{x} - x\|_2}{\|x\|_2} \approx 3.02 \times 10^{-10}.$$

This example suggests the following rule of thumb:

> If $\kappa(A) \approx 10^k$, then expect a loss of k significant digits in a computed (i.e., approximate) solution of $Ax = b$.

The loss of significant digits (or large relative forward error) is a consequence of the ill-conditioning of the problem, and hence it is independent of the numerical algorithm that is used for computing the approximation[2]. The best approach to deal with this situation is to avoid ill-conditioning of the problem in the first place.

Let $\widehat{x}$ be an approximation of the solution of $Ax = b$, and let $r = b - A\widehat{x}$ be the residual. Then

$$A\widehat{x} = b - r,$$

which shows that $\widehat{x}$ exactly solves the *perturbed* linear algebraic system

$$Ay = b + \Delta b, \quad \text{where} \quad \Delta b = -r.$$

Here only a perturbation of the input b is considered (and not of A), and hence in terms of the framework discussed in Section 2.1 the quantity $\|\Delta b\|/\|b\| = \|r\|/\|b\|$ is the relative

[2] Of course, with a poor algorithm we will likely lose even more significant digits, while if A^{-1} is known explicitly, we may be able to compute $x = A^{-1}b$ very accurately despite a large $\kappa(A)$.

backward error. The bound (2.7) can be written as

$$\frac{\|\widehat{x} - x\|}{\|x\|} \leq \kappa(A) \frac{\|\Delta b\|}{\|b\|},$$

and we recognize our rule of thumb (2.2). The backward error approach shows that if the relative backward error $\|r\|/\|b\|$ is small, then $\widehat{x}$ solves a nearby system with a slightly perturbed right hand side.

We now ask about perturbations of the matrix A such that a nonzero approximate solution $\widehat{x}$ solves the perturbed linear algebraic system. Let us define $y := \widehat{x}/\|\widehat{x}\|_2^2$, then $y^H \widehat{x} = 1$ and consequently $(I - \widehat{x}y^H)\widehat{x} = 0$. Hence for every $Z_B \in \mathbb{C}^{n \times n}$ the matrix

$$B := by^H + Z_B(I - \widehat{x}y^H)$$

satisfies $B\widehat{x} = b$, i.e.,

$$(A + \Delta A)\widehat{x} = b \quad \text{for} \quad \Delta A := B - A.$$

Thus, $\widehat{x}$ solves a whole family of perturbed linear algebraic systems, which is parameterized by the choice of Z_B. For the special case $Z_B = A$ we obtain $\Delta A = B - A = r\widehat{x}^H/\|\widehat{x}\|_2^2$, and the following result shows that this perturbation is minimal with respect to the 2-norm.

Theorem 2.19. *Let $A \in \mathbb{C}^{n \times n}$, $x \in \mathbb{C}^n$ and $b = Ax$. Let $\widehat{x} \in \mathbb{C}^n \setminus \{0\}$, then $(A + E)\widehat{x} = b$ for $E := r\widehat{x}^H/\|\widehat{x}\|_2^2$, and*

$$\frac{\|E\|_2}{\|A\|_2} = \frac{\|r\|_2}{\|A\|_2\|\widehat{x}\|_2} = \min\left\{ \frac{\|\Delta A\|_2}{\|A\|_2} \ : \ (A + \Delta A)\widehat{x} = b \right\}. \tag{2.8}$$

Proof. The assertion is trivial for $r = 0$, since then $A\widehat{x} = b$ and $E = 0$. Let $r \neq 0$, then for $E = r\widehat{x}^H/\|\widehat{x}\|_2^2$ we have

$$(A + E)\widehat{x} = A\widehat{x} + r\frac{\widehat{x}^H\widehat{x}}{\|\widehat{x}\|_2^2} = b \quad \text{and} \quad E = \frac{\|r\|_2}{\|\widehat{x}\|_2} \frac{r}{\|r\|_2} \left(\frac{\widehat{x}}{\|\widehat{x}\|_2} \right)^H. \tag{2.9}$$

The first equation shows that $\widehat{x}$ solves the perturbed linear algebraic system. In the second equation we have written the matrix E in the form σuv^H with two unit norm vectors u and v. This is actually an SVD of E, which in particular shows that $\|E\|_2 = \|r\|_2/\|\widehat{x}\|_2$, and hence we obtain the first equality in (2.8).
If $\Delta A \in \mathbb{C}^{n \times n}$ is any matrix such that $(A + \Delta A)\widehat{x} = b$, then $r = b - A\widehat{x} = \Delta A\widehat{x}$. Hence $\|r\|_2 \leq \|\Delta A\|_2\|\widehat{x}\|_2$, which yields

$$\frac{\|\Delta A\|_2}{\|A\|_2} \geq \frac{\|r\|_2}{\|A\|_2\|\widehat{x}\|_2}.$$

The lower bound is attained for $\Delta A = E$, which shows the second equality in (2.8). □

One of the main implications of Theorem 2.19 is that if the relative backward error $\|E\|_2/\|A\|_2$ is small, then $\widehat{x}$ solves a nearby system with a slightly perturbed matrix.

We next study the backward error of a nonzero approximate solution $\widehat{x}$ when allowing perturbations in both A and b. Let $y \in \mathbb{C}^n$ be any vector with $y^H\widehat{x} = 1$, and let $\alpha \in \mathbb{C}$ be arbitrary. Then a simple computation shows that $\Delta A_\alpha := \alpha r y^H$ and $\Delta b_\alpha := (\alpha - 1)r$ satisfy $(A + \Delta A_\alpha)\widehat{x} = b + \Delta b_\alpha$, i.e., $\widehat{x}$ exactly solves a whole family of perturbed linear algebraic systems, which is parameterized by α.
For $\alpha = 1$ we have $\Delta A_\alpha := r y^H$ and $\Delta b_\alpha = 0$, and choosing $y = \widehat{x}/\|\widehat{x}\|_2^2$ (hence $y^H\widehat{x} = 1$) we get the matrix E from Theorem 2.19. This result only works for the 2-norm, since its definition $\widehat{x}^H\widehat{x} = \|\widehat{x}\|_2^2$ is used in (2.9). In order to characterize the minimum norm backward perturbations for general norms (and when allowing perturbations in both A and b) we need some additional theory.

Lemma 2.20. *Let $\|\cdot\|$ be a norm on $\mathbb{C}^n$ and define*

$$\|\cdot\|^D : \mathbb{C}^n \to \mathbb{R} \quad \textit{with} \quad \|x\|^D := \max_{\|z\|=1} |x^H z| \ \textit{ for all } x \in \mathbb{C}^n.$$

Then $\|\cdot\|^D$ is a norm on $\mathbb{C}^n$ which is called the dual norm *of $\|\cdot\|$.*

Proof. Clearly $\|x\|^D \geq 0$ with equality if and only $x = 0$. For $\lambda \in \mathbb{C}$ we have $\|\lambda x\|^D = \max_{\|z\|=1} \|(\lambda x)^H z\| = |\lambda| \|x\|^D$. If $x_1, x_2 \in \mathbb{C}^n$, then

$$\begin{aligned}\|x_1 + x_2\|^D &= \max_{\|z\|=1} |(x_1 + x_2)^H z| \leq \max_{\|z\|=1} (|x_1^H z| + |x_2^H z|) \leq \max_{\|z\|=1} |x_1^H z| + \max_{\|z\|=1} |x_2^H z| \\ &= \|x_1\|^D + \|x_2\|^D,\end{aligned}$$

which completes the proof. □

Let us give some examples of dual norms.

Example 2.21. *On $\mathbb{C}^n$ we have*

$$\|\cdot\|_2^D = \|\cdot\|_2, \quad \|\cdot\|_\infty^D = \|\cdot\|_1, \quad \textit{and} \quad \|\cdot\|_1^D = \|\cdot\|_\infty.$$

(1) If $x \in \mathbb{C}^n$ is any nonzero vector, then

$$\begin{aligned}\|x\|_2^D &= \max_{\|z\|_2=1} |x^H z| \geq \left| x^H \frac{x}{\|x\|_2} \right| = \|x\|_2, \\ \|x\|_2^D &= \max_{\|z\|_2=1} |\langle z, x\rangle| \leq \max_{\|z\|_2=1} (\|z\|_2 \|x\|_2) = \|x\|_2,\end{aligned}$$

and thus $\|x\|_2 = \|x\|_2^D$, where in the upper bound we have used the Cauchy–Schwarz inequality in $\mathbb{C}^n$ equipped with the Euclidean inner product.

(2) For the second equality we note that

$$\|x\|_\infty^D = \max_{\|z\|_\infty=1} |x^H z| = \max_{\|z\|_\infty=1} \left|\sum_{j=1}^n \overline{x}_j z_j\right| \le \max_{\|z\|_\infty=1} \sum_{j=1}^n |x_j|\,|z_j| \le \sum_{j=1}^n |x_j| = \|x\|_1.$$

Each entry of x is of the form $x_j = r_j e^{i\varphi_j}$ for some $r_j \ge 0$ and $\varphi_j \in \mathbb{R}$. The vector $z \in \mathbb{C}^n$ with entries $z_j = e^{i\varphi_j}$ for $j = 1, \ldots, n$ satisfies $\|z\|_\infty = 1$ and $\overline{x}_j z_j = r_j$, and for this vector we have $|x^H z| = \|x\|_1$, so that the upper bound is attained.
(3) Finally, suppose that x_k is an entry of x with maximum modulus. Then

$$\|x\|_1^D = \max_{\|z\|_1=1} |x^H z| \le \max_{\|z\|_1=1} \sum_{j=1}^n |x_j|\,|z_j| \le |x_k| \max_{\|z\|_1=1} \sum_{j=1}^n |z_j| = \|x\|_\infty,$$

and for $z = e_k$ we have $\|z\|_1 = 1$ and $|x^H z| = |x_k| = \|x\|_\infty$, so that the upper bound is attained.

More generally, for $1 \le p \le \infty$ and the p-norm on $\mathbb{C}^n$ (see (0.1)) one can show that $\|\cdot\|_p^D = \|\cdot\|_q$, where $q = (1 - \frac{1}{p})^{-1}$, so that $\frac{1}{p} + \frac{1}{q} = 1$.
Basic properties of the dual norm are shown in the next result.

Lemma 2.22. *For all $x, y \in \mathbb{C}^n$ we have*

$$|x^H y| \le \|x\|^D \|y\| \quad \textit{and} \quad |x^H y| \le \|x\| \|y\|^D.$$

Proof. Both inequalities are obvious for $y = 0$. If $y \ne 0$, then for each $x \in \mathbb{C}^n$ we have

$$\frac{1}{\|y\|}\,|x^H y| = \left|x^H \frac{y}{\|y\|}\right| \le \max_{\|z\|=1} |x^H z| = \|x\|^D, \text{ and hence } |x^H y| \le \|x\|^D \|y\|.$$

The second inequality follows from the first by using $|x^H y| = |y^H x|$. □

For the 2-norm $\|\cdot\|_2$, both inequalities in Lemma 2.22 read

$$|x^H y| = |y^H x| = |\langle x, y\rangle| \le \|x\|_2 \|y\|_2,$$

which is the Cauchy–Schwarz inequality in $\mathbb{C}^n$ equipped with the Euclidean inner product. More generally, for the p- and q-norm on $\mathbb{C}^n$, where $\frac{1}{p} + \frac{1}{q} = 1$, the second inequality in Lemma 2.22 reads

$$|x^H y| = |y^H x| = |\langle x, y\rangle| \le \|x\|_p \|y\|_q, \text{ or } \left|\sum_{i=1}^n x_i \overline{y}_i\right| \le \left(\sum_{i=1}^n |x_i|^p\right)^{1/p} \left(\sum_{i=1}^n |y_i|^q\right)^{1/q}, \tag{2.10}$$

which is called the *Hölder inequality*.

For the 2-norm $\|\cdot\|_2$ and its dual norm $\|\cdot\|_2^D = \|\cdot\|_2$ we know from the properties of the Cauchy–Schwarz inequality that $|x^H y| = \|x\|_2\|y\|_2^D$ holds if and only if x, y are linearly dependent. In general we may have $|x^H y| = \|x\|\|y\|^D$ also for linearly independent vectors. For example, $x = e_1$ and $y = e_1 + e_2$ in $\mathbb{C}^n$, $n \geq 2$, yield $1 = |x^H y| = \|x\|_1\|y\|_1^D$, where $\|\cdot\|_1^D = \|\cdot\|_\infty$.

If $\|\cdot\|^{DD}$ denotes the dual norm of $\|\cdot\|^D$, i.e.,

$$\|x\|^{DD} = \max_{\|z\|^D=1} |x^H z| \quad \text{for all } x \in \mathbb{C}^n,$$

then Lemma 2.22 gives

$$\|x\|^{DD} = \max_{\|z\|^D=1} |x^H z| \leq \max_{\|z\|^D=1} (\|x\|\|z\|^D) = \|x\|.$$

With some more effort one can also show the reverse inequality, which gives the following important Duality Theorem.

Theorem 2.23. *If $\|\cdot\|$ is a norm on $\mathbb{C}^n$, $\|\cdot\|^D$ is the dual norm of $\|\cdot\|$, and $\|\cdot\|^{DD}$ is the dual norm of $\|\cdot\|^D$, then $\|x\| = \|x\|^{DD}$ for all $x \in \mathbb{C}^n$, i.e., $\|\cdot\| = \|\cdot\|^{DD}$.*

Corollary 2.24. *Let $\|\cdot\|$ be a norm on $\mathbb{C}^n$, let $\|\cdot\|^D$ be the dual norm and let $\widehat{x} \in \mathbb{C}^n \setminus \{0\}$. Then there exists a vector $y \in \mathbb{C}^n \setminus \{0\}$ with*

$$1 = y^H \widehat{x} = \|\widehat{x}\|\|y\|^D.$$

Such a vector y is called a dual vector *of $\widehat{x}$ (with respect to the norm $\|\cdot\|$).*

Proof. For the given vector $\widehat{x}$ we know that

$$\|\widehat{x}\| = \|\widehat{x}\|^{DD} = \max_{\|z\|^D=1} |\widehat{x}^H z|.$$

The maximum is attained for some vector $\widetilde{z}$ with $\|\widetilde{z}\|^D = 1$. Set $y := \widetilde{z}/\|\widehat{x}\|$, then $\|y\|^D = \|\widetilde{z}\|^D/\|\widehat{x}\|$, giving

$$1 = \|\widetilde{z}\|^D = \|\widehat{x}\|\|y\|^D.$$

Moreover, using $\widetilde{z} = \|\widehat{x}\|y$ we get

$$\|\widehat{x}\| = |\widehat{x}^H \widetilde{z}| = \|\widehat{x}\||\widehat{x}^H y| = \|\widehat{x}\||y^H \widehat{x}|,$$

so that $|y^H \widehat{x}| = 1$. Without loss of generality we can assume that $y^H \widehat{x} = 1$, since y can be multiplied by a suitable constant $e^{i\theta}$. □

Remark 2.25. Corollary 2.24 is a finite dimensional version of the following corollary of the Hahn–Banach Theorem:
If $(\mathcal{X}, \|\cdot\|_\mathcal{X})$ is a normed linear space, and $(\mathcal{X}^*, \|\cdot\|_{\mathcal{X}^*})$ is the dual space with $\|\ell\|_{\mathcal{X}^*} = \sup_{\|x\|_\mathcal{X} \leq 1} |\ell(x)|$, then for each nonzero $x_0 \in \mathcal{X}$ there exists a nonzero $\ell_0 \in \mathcal{X}^*$ with $\ell_0(x_0) = \|\ell_0\|_{\mathcal{X}^*}\|x_0\|_\mathcal{X}$.

Consider the 2-norm $\|\cdot\|_2$ and $\widehat{x} \in \mathbb{C}^n \setminus \{0\}$. We want to construct a dual vector y of $\widehat{x}$. Since such a vector must satisfy $y^H\widehat{x} = 1$, it must be a nonzero scalar multiple of some vector $\widetilde{y} \in \mathbb{C}^n \setminus \{0\}$ with $\widetilde{y}^H\widehat{x} \neq 0$. Thus, $y = \alpha\widetilde{y}$ for some nonzero $\alpha \in \mathbb{C}$. From the condition

$$1 = y^H\widehat{x} = \overline{\alpha}\widetilde{y}^H\widetilde{x}$$

we obtain $\alpha = 1/\overline{\widetilde{y}^H\widehat{x}}$, so that

$$\|\widehat{x}\|_2\|y\|_2 = \frac{\|\widehat{x}\|_2\|\widetilde{y}\|_2}{|\widetilde{y}^H\widehat{x}|},$$

which must be equal to 1. From the Cauchy–Schwarz inequality we know that $|\widetilde{y}^H\widehat{x}| = \|\widehat{x}\|_2\|\widetilde{y}\|_2$ holds if and only if $\widehat{x}, \widetilde{y}$ are linearly dependent, i.e., $\widetilde{y} = \beta\widehat{x}$ for some $\beta \in \mathbb{C} \setminus \{0\}$. But then

$$y = \frac{\widetilde{y}^H}{\widetilde{y}^H\widehat{x}} = \frac{\widehat{x}^H}{\|\widehat{x}\|_2^2},$$

which shows that the dual vector of $\widehat{x}$ with respect to the 2-norm is uniquely determined and given by $y = \widehat{x}/\|\widehat{x}\|_2^2$.
In general, a dual vector may not be uniquely determined. For example, the vectors $x = e_1$ and $y = e_1 + e_2$ in $\mathbb{C}^n$, $n \geq 2$, satisfy

$$1 = x^Hx = \|x\|_1\|x\|_\infty \quad \text{and} \quad 1 = y^Hx = \|x\|_1\|y\|_\infty,$$

which shows that the linearly independent vectors x and y are both dual vectors of x with respect to the 1-norm.

Using the uniquely determined dual vector of $\widehat{x}$ with respect to the 2-norm, the perturbation matrix E in Theorem 2.19 can be written as $E = ry^H$. The generalization to general norms appears in the following important result of Rigal and Gaches [32].

Theorem 2.26. *Let $A \in \mathbb{C}^{n\times n}$, $x \in \mathbb{C}^n$, $b = Ax$, and $\widehat{x} \in \mathbb{C}^n \setminus \{0\}$. Let $\|\cdot\|$ be any norm on $\mathbb{C}^n$ as well as the induced matrix norm on $\mathbb{C}^{n\times n}$. Let $E \in \mathbb{C}^{n\times n}$ and $f \in \mathbb{C}^n$ be given, and suppose that $y \in \mathbb{C}^n$ is a dual vector of $\widehat{x}$. Then the* normwise backward error *of the approximate solution $\widehat{x}$ of $Ax = b$ is given by*

$$\begin{aligned}\eta_{E,f}(\widehat{x}) &:= \min\{\varepsilon : (A + \Delta A)\widehat{x} = b + \Delta b \text{ with } \|\Delta A\| \leq \varepsilon\|E\| \text{ and } \|\Delta b\| \leq \varepsilon\|f\|\} \\ &= \frac{\|r\|}{\|E\|\|\widehat{x}\| + \|f\|},\end{aligned}$$

and the second equality is attained by the perturbations

$$\begin{aligned}\Delta A_{\min} &:= \frac{\|E\|\|\widehat{x}\|}{\|E\|\|\widehat{x}\| + \|f\|} ry^H, \\ \Delta b_{\min} &:= -\frac{\|f\|}{\|E\|\|\widehat{x}\| + \|f\|} r.\end{aligned}$$

Proof. First note that if ΔA and Δb are arbitrary perturbations with $(A+\Delta A)\widehat{x} = b+\Delta b$ and $\|\Delta A\| \le \varepsilon\|E\|$, $\|\Delta b\| \le \varepsilon\|f\|$, then $r = b - A\widehat{x} = \Delta A\widehat{x} - \Delta b$, and hence

$$\|r\| \le \|\Delta A\|\|\widehat{x}\| + \|\Delta b\| \le \varepsilon(\|E\|\|\widehat{x}\| + \|f\|),$$

which shows that ε is bounded from below as

$$\varepsilon \ge \frac{\|r\|}{\|E\|\|\widehat{x}\| + \|f\|} =: \varepsilon_{\min}.$$

Next, for the perturbations $\Delta A_{\min}$ and $\Delta b_{\min}$ stated in the theorem we have

$$\begin{aligned}(A + \Delta A_{\min})\widehat{x} &= A\widehat{x} + \frac{\|E\|\|\widehat{x}\|}{\|E\|\|\widehat{x}\| + \|f\|} r \underbrace{y^H\widehat{x}}_{=1} \\ &= \frac{\|E\|\|\widehat{x}\|}{\|E\|\|\widehat{x}\| + \|f\|} b + \frac{\|f\|}{\|E\|\|\widehat{x}\| + \|f\|} A\widehat{x} \\ &= b + \Delta b_{\min},\end{aligned}$$

i.e., $\widehat{x}$ solves the system that is perturbed by $\Delta A_{\min}$ and $\Delta b_{\min}$.
Finally, we show that the value $\varepsilon_{\min}$ is attained by the perturbations $\Delta b_{\min}$, $\Delta A_{\min}$:

$$\begin{aligned}\|\Delta b_{\min}\| &= \frac{\|f\|}{\|E\|\|\widehat{x}\| + \|f\|}\|r\| = \varepsilon_{\min}\|f\|, \\ \|\Delta A_{\min}\| &= \max_{z\neq 0} \frac{\|\Delta A_{\min} z\|}{\|z\|} = \frac{\|E\|\|\widehat{x}\|}{\|E\|\|\widehat{x}\| + \|f\|} \max_{z\neq 0} \frac{\|r y^H z\|}{\|z\|} \\ &= \frac{\|E\|\|\widehat{x}\|}{\|E\|\|\widehat{x}\| + \|f\|}\|r\| \underbrace{\max_{z\neq 0} \frac{|y^H z|}{\|z\|}}_{=1/\|\widehat{x}\|} = \varepsilon_{\min}\|E\|,\end{aligned}$$

where we have used that $1 = y^H\widehat{x} = \|\widehat{x}\|\|y\|^D = \|\widehat{x}\| \cdot \max_{\|z\|=1} |y^H z|$. □

As a special case of Theorem 2.26 consider the 2-norm $\|\cdot\|_2 = \|\cdot\|_2^D$, so that $y = \widehat{x}/\|\widehat{x}\|_2^2$ is the unique dual vector of $\widehat{x}$. Then for $E = A$ and $f = 0$ we obtain

$$\eta_{A,0}(\widehat{x}) = \frac{\|r\|_2}{\|A\|_2\|\widehat{x}\|_2}, \quad \Delta A_{\min} = \frac{r\widehat{x}^H}{\|\widehat{x}\|_2^2}, \quad \Delta b_{\min} = 0,$$

and hence we recover Theorem 2.19. Similarly, for $E = 0$ and $f = b$ we obtain

$$\eta_{0,b}(\widehat{x}) = \frac{\|r\|_2}{\|b\|_2}, \quad \Delta A_{\min} = 0, \quad \Delta b_{\min} = -r;$$

cf. the bound (2.7).

For $E = A$ and $f = b$, the resulting quantity

$$\begin{aligned}\eta_{A,b}(\widehat{x}) &= \min\{\varepsilon : (A+\Delta A)\widehat{x} = b + \Delta b \text{ with } \|\Delta A\| \le \varepsilon\|A\|, \|\Delta b\| \le \varepsilon\|b\|\} \\ &= \frac{\|r\|}{\|A\|\|\widehat{x}\| + \|b\|}\end{aligned}$$

is called the *normwise relative backward error* of the approximation $\widehat{x}$. A numerical method for solving $Ax = b$ with $A \in \mathbb{C}^{n\times n}$ nonsingular is called *normwise backward stable* if executed in floating precision arithmetic it produces a computed approximation $\widehat{x}$ of x with $\eta_{A,b}(\widehat{x})$ on the order of the unit roundoff u (or somewhat larger, when the context allows). For the formal definition of the unit roundoff see Section 3.1.
The main point is that a normwise backward stable method yields an exact solution $\widehat{x}$ of a (slightly) perturbed system, i.e., $(A + \Delta A)\widehat{x} = b + \Delta b$ with $\|\Delta A\|/\|A\| \le \varepsilon$ and $\|\Delta b\|/\|b\| \le \varepsilon$, where ε is "small". When our original system $Ax = b$ contains uncertainties (e.g., measurement errors), then the perturbed system may just be the system we wanted to solve!

A numerical method for solving $Ax = b$ with $A \in \mathbb{C}^{n\times n}$ nonsingular is called *normwise forward stable* when the computed approximation $\widehat{x}$ satisfies

$$\frac{\|\widehat{x} - x\|}{\|x\|} = O(\kappa(A)u);$$

again recall our rule of thumb (2.2).
The next result shows that under some reasonable assumptions normwise backward stability implies normwise forward stability.

Theorem 2.27. *Let $A \in \mathbb{C}^{n\times n}$ be nonsingular, $x \in \mathbb{C}^n \setminus \{0\}$, $b = Ax$, $\varepsilon > 0$, and suppose that $\widehat{x} \in \mathbb{C}^n$ satisfies*

$$(A + \Delta A)\widehat{x} = b + \Delta b,$$

where the perturbations satisfy $\|\Delta A\| \le \varepsilon\|A\|$ and $\|\Delta b\| \le \varepsilon\|b\|$. Suppose further that $\varepsilon\kappa(A) < 1$. Then for consistent norms we have

$$\frac{\|\widehat{x} - x\|}{\|x\|} \le \frac{2\varepsilon\kappa(A)}{1 - \varepsilon\kappa(A)}.$$

Thus, if a numerical method produces $\widehat{x}$ with $\eta_{A,b}(\widehat{x}) = u$, i.e., the method is normwise backward stable, and $u\kappa(A) < 1$, then

$$\frac{\|\widehat{x} - x\|}{\|x\|} = O(\kappa(A)u),$$

i.e., the method is normwise forward stable.

Proof. From $A(\widehat{x} - x) = \Delta b - \Delta A\widehat{x}$ we obtain $\widehat{x} - x = A^{-1}(\Delta b - \Delta A x + \Delta A(x - \widehat{x}))$. Taking norms yields

$$\begin{aligned}\|\widehat{x} - x\| &\leq \|A^{-1}\|(\varepsilon\|b\| + \varepsilon\|A\|\|x\| + \varepsilon\|A\|\|\widehat{x} - x\|) \\ &= \varepsilon(\|A^{-1}\|\|b\| + \kappa(A)\|x\|) + \varepsilon\kappa(A)\|\widehat{x} - x\| \\ &\leq \varepsilon(\kappa(A)\|x\| + \kappa(A)\|x\|) + \varepsilon\kappa(A)\|\widehat{x} - x\|,\end{aligned}$$

where we have used that $\|b\| \leq \|A\|\|x\|$, and which implies

$$\frac{\|\widehat{x} - x\|}{\|x\|} \leq \frac{2\varepsilon\kappa(A)}{1 - \varepsilon\kappa(A)}.$$

If $\widehat{x}$ satisfies $\eta_{A,b}(\widehat{x}) = u$ and $u\kappa(A) < 1$, then

$$\frac{\|\widehat{x} - x\|}{\|x\|} \leq \frac{2u\kappa(A)}{1 - u\kappa(A)} = O(\kappa(A)u).$$

□

Chapter 3

Direct Methods for Solving Linear Algebraic Systems

In this chapter we consider linear algebraic systems $Ax = b$ with a nonsingular matrix $A \in \mathbb{C}^{n\times n}$, so that $x = A^{-1}b$ is well defined.
Direct methods for solving $Ax = b$ are based (at least implicitly) on a decomposition or factorization of A into easily invertible factors, and the subsequent solution of the systems involving these factors. A computed approximation $\widehat{x}$ of the exact solution x is available only at the end of this process. *Iterative methods*, on the other hand, generate a sequence of intermediate approximations, and can be stopped once a user-specified accuracy of the approximate solution is attained.
For an example of a direct method we consider an HPD matrix A and its uniquely determined Cholesky decomposition $A = LL^H$, so that $x = (LL^H)^{-1}b = L^{-H}(L^{-1}b)$. In practice ones does not invert the matrices L and L^H but rather solves the two triangular systems

$$Ly = b \quad \text{and} \quad L^H x = y,$$

which in exact arithmetic gives $x = L^{-H}y = L^{-H}(L^{-1}b) = A^{-1}b$. In finite precision arithmetic, however, all computations are affected by rounding errors. The direct method of solving $Ax = b$ based on the Cholesky decomposition therefore generates an approximation $\widehat{x}$ of x, and the quality of this approximation depends on the errors made in the computation of the decomposition, and in solving the two triangular systems.

3.1 The floating point numbers

In order to analyze the errors in finite precision computations, we will have a brief look at the computer arithmetic. In this arithmetic we do not have all real numbers, but only a finite subset, the *floating point numbers*. A nonzero floating point number is of the form

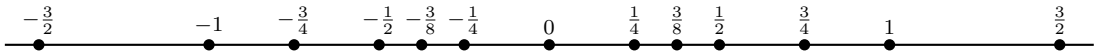
$$y = \pm\beta^e \left(\frac{d_1}{\beta} + \frac{d_2}{\beta^2} + \cdots + \frac{d_t}{\beta^t} \right), \tag{3.1}$$

where the integer parameters defining y are the *base* $\beta \geq 2$, the *precision* $t \geq 1$, and the *exponent* e, where $e_{\min} \leq e \leq e_{\max}$. The *digits* of the floating point number satisfy $0 \leq d_i \leq \beta - 1$, and we require that $d_1 \neq 0$ for normalization. The number $y = 0$ is then considered extra. We denote the set of the floating point numbers by

$$F = F(\beta, t, e_{\min}, e_{\max}) \subset \mathbb{R}.$$

Alternative ways to write these numbers include $y = \pm m \times \beta^{e-t}$ and $y = \pm .d_1 d_2 \ldots d_t \times \beta^e$.

Example 3.1. *If $\beta = 2$, $t = 2$, $e_{\min} = -1$ and $e_{\max} = 1$, then F contains $y = 0$ and the 12 nonzero elements $\pm 1/4$, $\pm 3/8$, $\pm 1/2$, $\pm 3/4$, ± 1, $\pm 3/2$:*



The example illustrates that the absolute spacing between the floating point numbers increases by a factor of β with each increase or decrease of the exponent. We have

$$\beta \left(\frac{1}{\beta} + \frac{0}{\beta^2} + \cdots + \frac{0}{\beta^t} \right) = 1 \ \in F,$$

and the next larger floating point number is

$$\beta \left(\frac{1}{\beta} + \frac{0}{\beta^2} + \cdots + \frac{0}{\beta^{t-1}} + \frac{1}{\beta^t} \right) = 1 + \beta^{1-t} \ \in F.$$

We call

$$\epsilon_M := \beta^{1-t}$$

the *machine epsilon*.

Example 3.2 (Different floating point systems)**.** *The* double precision *floating point numbers in the IEEE 754 standard (for the latest edition from 2019 see* `https://ieeexplore.ieee.org/document/8766229`*) are given by*

$$F(\beta, t, e_{\min}, e_{\max}) = F(2, 53, -1022, 1023),$$

so that $\epsilon_M = 2^{-52} \approx 2.22 \times 10^{-16}$. This standard is also used in MATLAB, and the command `eps` *shows the machine epsilon:*

```
>> eps
ans =
     2.220446049250313e-16
>> 2^(-52)
ans =
     2.220446049250313e-16
```

Note that while every double precision floating point number has $t = 53$ *digits with respect to the base* $\beta = 2$*, called binary digits or* bits*, it has only* $t \cdot \log_{10} \beta = 53 \cdot \log_{10} 2 \approx 15.95 \approx 16$ *decimal digits. The fact that this is a very fine "discretization" of the real numbers that is sufficient for many practical purposes, is nicely illustrated by the following quote from Nick Trefeten [43]:*

"Thus, roughly speaking, A gas or solid has around 10^9 particles per meter. *This is how fine the discretization is in our physical world. It's interesting to compare it with the floating-point arithmetic on our computers. In the IEEE double-precision standard that has prevailed since the 1980s, the real line is discretized by* $2^{52} \approx 10^{16}$ *numbers between* 1 *and* 2*, the same between* 2 *and* 4*, and so on. Thus we find:* Computer arithmetic is a million times finer than physics.*"*

Storing a nonzero double precision floating point number requires one bit for the sign, and 52 bits for the digits $d_i \in \{0, 1\}$*,* $i = 2, \ldots, 53$*, where* $d_1 = 1$ *because of the normalization. The exponent requires 11 bits, where one bit is used for the sign, and 10 bits for* $e \leq 1023 = 2^{10} - 1$*. The exponent* $1024 = 2^{10}$ *is used to represent the special values* `Inf` *(infinity) and* `NaN` *(not a number). Thus, each nonzero double precision floating point number consists of* $1 + 52 + 11 = 64$ *bits.*

Many applications do not require highly accurate computations, and hence "shorter" floating point numbers that give a lower resolution of the real numbers can be used. For example, the single precision *floating point numbers in the IEEE 754 standard have only 32 bits: One for the sign, 23 for the digits (*$t = 24$*), and 8 for the exponent (including one for the sign). Here* $\epsilon_M = 2^{-23} \approx 1.19 \times 10^{-7}$*,* $e_{\min} = -126$*, and* $e_{\max} = 127 \, (= 2^7 - 1)$*.*

There also exists the IEEE half precision *floating point format fp16 with only 16 bits. Here one is used for the sign, 10 for the digits, and 5 for the exponent (including one for the sign). In fp16 we thus have* $\epsilon_M = 2^{-10} \approx 9.77 \times 10^{-4}$ *and* $e_{\max} = 2^4 = 16$*. The fp16 format is implemented on many graphic processing units (GPUs), where computations with high precision are usually not required.*

Researchers in the area of deep neural networks recently have used low precision arithmetic based on 8 bits, and even just 4 bits [41]. In the ultra-low precision *FP4 format, the base is* $\beta = 4$*, one bit is used for the sign of the number, three bits are used for the exponent (one again is for the sign), and in the notation above we have no digits, i.e.,* $t = 0$*. Thus, the nonzero numbers are of the form*

$$y = \pm 4^{\pm e_1 e_2} \quad \text{or} \quad y = \pm 4^{\pm e},$$

where $e_1 e_2 \in \{00, 10, 01, 11\}$ *(binary) or* $e \in \{0, 1, 2, 3\}$*. The set of the 15 FP4 numbers (including zero) is given by*

$$F = \{0, \pm 2^{-6}, \pm 2^{-4}, \pm 2^{-2}, \pm 2^{0}, \pm 2^{2}, \pm 2^{4}, \pm 2^{6}\}.$$

The absolute values of the nonzero numbers are between $2^{-6} = 0.015625$ *and* $2^6 = 64$.

Using the single or half precision, or even "shorter" formats can lead to significant speedups of numerical algorithms in comparison with double precision, since moving "shorter" numbers simply is faster than moving "longer" numbers. This fact is becoming increasingly important in modern (very) large scale computations, where communication or data movement is more expensive than performing the actual floating point computations. Currently we are approaching exascale computing, *where hardware will be capable of executing* 10^{18} *(i.e., a quintillion) floating point operations per second. Of course, the speedup achieved with "shorter" numbers comes at the cost of accuracy. In practical applications one therefore needs to determine the fewest number of bits (or the "shortest possible" numbers) for which the given problem can be solved with an acceptable accuracy.*

Let G be the set of all numbers of the form (3.1) without restriction on the exponent. Then for every nonzero $x \in \mathbb{R}$ we denote by $fl(x)$ the element of G closest to x, i.e.,

$$fl(x) := \arg\min_{z \in G} |x - z|.$$

The mapping $x \mapsto fl(x)$ is called *rounding*, and the error made when working with $fl(x)$ instead of the exact real number x is called a *rounding error*. There are several ways to break ties when rounding, which we will not discuss here.
The absolute values of the nonzero floating point numbers satisfy

$$\beta^{e_{\min}-1} \le |y| \le \beta^{e_{\max}} \left(\frac{\beta-1}{\beta} + \frac{\beta-1}{\beta^2} + \cdots + \frac{\beta-1}{\beta^t} \right) = \beta^{e_{\max}}(1 - \beta^{-t}).$$

We are interested only in numbers that after rounding lie in this *range* of F. We say that the (nonzero) number $fl(x)$ *underflows* when

$$0 < |fl(x)| < \min\{|y| \,:\, 0 \neq y \in F\},$$

and it *overflows* when

$$|fl(x)| > \max\{|y| \,:\, y \in F\}.$$

Example 3.3. *For the floating point numbers* $F = F(2, 2, -1, 1)$ *in Example 3.1 we have*

$$\beta^{e_{\min}-1} = \frac{1}{4} \le |y| \le \frac{3}{2} = \beta^{e_{\max}}(1 - \beta^{-t})$$

for every nonzero $y \in F$. *For every nonzero* $x \in \mathbb{R}$, *the number* $fl(x)$ *is the element of*

$$G = \left\{ \pm 2^e \left(\frac{1}{2} + \frac{d_2}{4} \right) \,:\, e \in \mathbb{Z},\, d_2 \in \{0, 1\} \right\}$$

that is closest to x. We have $F \subset [-3/2, 3/2]$, while $\pm 2 \in G$. Thus, for every $x \in \mathbb{R}$ with $|x| > 7/4$ we get $|fl(x)| \geq 2 \notin F$, so that $fl(x)$ overflows.

Example 3.4. *In the IEEE 754 double precision arithmetic (see Example 3.2) we have $\beta = 2$ and $e_{\max} = 1023$. Hence $y = 2^{1024}$ is outside the range of the corresponding floating point numbers. A computation in MATLAB gives*

```
>> 2^1023
ans =
  8.9885e+307
>> 2^1024
ans =
   Inf
```

The absolute value range of the double precision floating point numbers is given by

```
>> realmin
ans =
  2.2251e-308
>> realmax
ans =
  1.7977e+308
```

If $x \in \mathbb{R}$ is nonzero and $fl(x)$ does not underflow or overflow, then the definition of the machine epsilon yields

$$\frac{|x - fl(x)|}{|x|} = \left|1 - \frac{fl(x)}{x}\right| \leq \frac{1}{2}\epsilon_M =: u,$$

where u is called the *unit roundoff*. Hence there exists a $\delta \in \mathbb{R}$, depending on x, such that

$$fl(x) = x(1 + \delta), \quad |\delta| \leq u.$$

Suppose that $\circledast$ is any of the arithmetic operations $+, -, *, \div$. Then in the *standard model for computing with floating point numbers* we assume that for any two numbers $y_1, y_2 \in F$ there exists some $\delta \in \mathbb{R}$, depending on $y_1 \circledast y_2$, such that

$$fl(y_1 \circledast y_2) = (y_1 \circledast y_2)(1 + \delta), \quad |\delta| \leq u. \tag{3.2}$$

Usually it is also assumed that the square root of a floating point number can be computed with the same accuracy. Computations with floating point numbers are frequently referred to as computations in *finite precision arithmetic*, and the term *exact arithmetic* is used for (mathematical) computations without rounding errors (e.g., with real or complex numbers).

If $y_1, y_2 \in \mathbb{R}$ are not floating point numbers, but are in the range of F, then in order to compute $y_1 \circledast y_2$ we first need to round y_1 and y_2. Let us consider this for the multiplication, where we skip the sign for simplicity:

$$\begin{aligned} fl(fl(y_1)fl(y_2)) &= fl(y_1(1+\delta_1)y_2(1+\delta_2)) = (y_1(1+\delta_1)y_2(1+\delta_2))(1+\delta_3) \\ &= y_1y_2(1+\delta_1+\delta_2+\delta_3+\delta_1\delta_2+\delta_1\delta_3+\delta_2\delta_3+\delta_1\delta_2\delta_3), \end{aligned}$$

where $|\delta_j| \leq u$, $j = 1, 2, 3$. Neglecting the terms containing two or three factors of δ_i, which are (much) smaller than the unit roundoff, we can write

$$|y_1y_2 - fl(fl(y_1)fl(y_2))| \approx |y_1y_2|\,\delta, \quad \text{where } |\delta| = |\delta_1+\delta_2+\delta_3| \leq 3u.$$

Consequently, if we first need to round the numbers, then the overall (relative) error in an operation may be larger than the unit roundoff. In the rounding error analysis of algorithms in the following sections we will usually assume, for simplicity, that the input is already given by floating point numbers. This is justified since our goal will be to analyze which rounding errors are produced by the actual algorithm, independently of a potential rounding of the input.

Example 3.5. *The unit roundoff is $u = \frac{1}{2}\epsilon_M$, and a MATLAB computation with IEEE 754 double precision arithmetic gives*

```
>> eps
ans =
     2.220446049250313e-16
>> u=eps/2
u =
     1.110223024625157e-16
```

By definition of the unit roundoff we have $fl(1+u) = 1$, which we can check in MATLAB by computing

```
>> x=1+u; x==1
ans =
  logical
   1
```

If $y_1 = y_2 = 1+u$, then the product $fl(y_1)fl(y_2) = 1 \cdot 1 = 1$ can be computed exactly:

```
>> x=(1+u)*(1+u); x==1
ans =
  logical
   1
```

Note that $fl(1+u) = 1 = (1+u)(1+\delta)$ *for* $\delta := -u/(1+u)$. *This gives*

$$fl(fl(y_1)fl(y_2)) = fl(y_1)fl(y_2) = (1+u)(1+\delta)(1+u)(1+\delta) = (1+u)^2(1+\delta)^2$$
$$= y_1y_2\left(1 - \frac{2u}{1+u} + \frac{u^2}{(1+u)^2}\right),$$

and hence $y_1y_2 - fl(fl(y_1)fl(y_2)) \approx y_1y_2\frac{2u}{1+u}$.

Example 3.6. *This example computed in MATLAB shows that rounding errors may occur even when computing with small integers:*

```
>> 1-(1/49)*49
ans =
     1.110223024625157e-16
```

The number $x = 49$ *is the smallest positive integer for which evaluating* $1 - (1/x)x$ *in MATLAB (and hence IEEE 754 double precision arithmetic) does not give exactly zero.*

Example 3.7. *Here is another example*[a] *computed in MATLAB:*

```
>> a=57055; b=339590; c=340126;
>> a^3+b^3-c^3
ans =
     0
```

The result of this computation contradicts Fermat's Last Theorem[b]*, so clearly there must be something wrong.*
When evaluating the term $a^3 + b^3 - c^3$, *MATLAB evaluates from left to right, i.e., it first computes* $a^3 + b^3$ *and then subtracts* c^3. *Exact computations would give*

$$a^3 + b^3 = 39.347.712.995.520.375,$$
$$c^3 = 39.347.712.995.520.376,$$

so that in fact $a^3 + b^3 - c^3 = -1$. *However, we see above that the numbers* $a^3 + b^3$ *and* c^3 *both have 17 digits. In the IEEE 754 double precision arithmetic with 16 decimal digits (see Example 3.2) the closest floating point number to both these numbers is*

```
fl(a^3+b^3)=fl(c^3)=3.934771299552038e+16
```

where the last digit is obtained by rounding up. Consequently, taking the difference of the rounded numbers gives exactly zero. Here are the floating point numbers that MATLAB uses for the computation:

```
>> a^3, b^3, c^3
ans =
     1.857296024413750e+14
ans =
     3.916198339307900e+16
ans =
     3.934771299552038e+16
>> a^3+b^3
ans =
     3.934771299552038e+16
```

The error we have made is tiny, but a "blind trust" in the results of finite precision computations and ignoring the existence of rounding errors would in this case produce a false counterexample to one of the most celebrated results of mathematics!

[a] Numbers taken from a tweet concerning computations in Javascript of Vitalik Buterin on February 7, 2018: `https://twitter.com/VitalikButerin/status/961259770090008576`.

[b] For integers $n > 2$ the equation $a^n + b^n = c^n$ cannot be solved with positive integers a, b, c.

Example 3.8. *The numbers in Example 3.7 can also be used to illustrate that the operations in finite precision arithmetic are in general* not associative. *The numbers*

$$\begin{aligned} a^3 &= \quad 185.729.602.441.375, \\ b^3 - c^3 &= -185.729.602.441.376 \end{aligned}$$

both have only 15 digital digits, and their sum is evaluated correctly in IEEE 754 double precision arithmetic:

```
>> a^3+(b^3-c^3)
ans =
    -1
```

Thus, in the finite precision computation we have $(a^3 + b^3) - c^3 \neq a^3 + (b^3 - c^3)$. *The non-associativity is even more striking in the following computation:*

```
>> (a^3+b^3)-(c^3-4)
ans =
     8
```

```
>> a^3+(b^3-c^3+4)
ans =
     3
```

3.2 Basic results about rounding errors

In addition to understanding the (small) rounding errors that potentially occur in each finite precision computation, it is important to understand how these errors can "add up" when performing many such computations. A basic but nevertheless useful example, which occurs in numerous algorithms, is the computation of the inner product of two real vectors, i.e.,

$$y^T x = \sum_{i=1}^{n} x_i y_i, \quad x, y \in \mathbb{R}^n.$$

The number $y^T x$ can be computed the following algorithm:

$s = 0$
for $i = 1, \ldots, n$ **do**
  $s = s + x_i y_i$
end for

Since we are interested in the accumulation of rounding errors due to the execution of this algorithm, *we assume that the entries of x and y are floating point numbers* (see the discussion above).
Let $\widehat{s}_k$, $k = 1, \ldots, n$, denote the computed partial sum of the first k terms. Then there exist $|\delta_i| \leq u$ such that

$$\begin{aligned}
\widehat{s}_1 &= fl(x_1 y_1) = x_1 y_1 (1 + \delta_1), \\
\widehat{s}_2 &= fl(\widehat{s}_1 + fl(x_2 y_2)) = (\widehat{s}_1 + x_2 y_2 (1 + \delta_2))(1 + \delta_3) \\
&= x_1 y_1 (1 + \delta_1)(1 + \delta_3) + x_2 y_2 (1 + \delta_2)(1 + \delta_3).
\end{aligned}$$

In order to reduce the technicalities we will write $1 \pm \delta$ instead of $1 + \delta_i$, where $|\delta| \leq u$. Thus,

$$\begin{aligned}
\widehat{s}_2 &= x_1 y_1 (1 \pm \delta)^2 + x_2 y_2 (1 \pm \delta)^2, \\
\widehat{s}_3 &= fl(\widehat{s}_2 + fl(x_3 y_3)) = (\widehat{s}_2 + x_3 y_3 (1 \pm \delta))(1 \pm \delta) \\
&= x_1 y_1 (1 \pm \delta)^3 + x_2 y_2 (1 \pm \delta)^3 + x_3 y_3 (1 \pm \delta)^2,
\end{aligned}$$

and inductively we get

$$\widehat{s}_n = x_1 y_1 (1 \pm \delta)^n + x_2 y_2 (1 \pm \delta)^n + x_3 y_3 (1 \pm \delta)^{n-1} + \cdots + x_n y_n (1 \pm \delta)^2. \tag{3.3}$$

The derivation of an error estimate is based on the following elementary, yet important result; see [16, Lemma 3.1].

Lemma 3.9. *If $|\delta_i| \le u$ for $i = 1, \dots, n$, and $nu < 1$, then*

$$\prod_{i=1}^{n}(1+\delta_i) = 1+\theta_n, \quad \textit{where} \quad |\theta_n| \le \frac{nu}{1-nu} =: \gamma_n.$$

Proof. We have

$$\prod_{i=1}^{n}(1+\delta_i) \le \prod_{i=1}^{n}(1+u) = (1+u)^n \le \frac{1}{1-nu} = 1 + \frac{nu}{1-nu},$$
$$\prod_{i=1}^{n}(1+\delta_i) \ge \prod_{i=1}^{n}(1-u) = (1-u)^n \ge 1 - \frac{nu}{1-nu},$$

where in each case the last inequality can be shown by induction on n under the assumption that $nu < 1$. □

Note that

$$\gamma_n = \frac{nu}{1-nu} = nu(1 + nu + (nu)^2 + \cdots) = nu + O(n^2u^2). \tag{3.4}$$

We will now use the notation

$$fl(y^T x) := \widehat{s}_n$$

for the computed result in (3.3). Using Lemma 3.9, we then can write (3.3) as

$$\begin{aligned} fl(y^T x) &= x_1y_1(1+\theta_n) + x_2y_2(1+\widetilde{\theta}_n) + x_3y_3(1+\theta_{n-1}) + \cdots + x_ny_n(1+\theta_2) \\ &= y^T x + (x_1y_1\theta_n + x_2y_2\widetilde{\theta}_n + x_3y_3\theta_{n-1} + \cdots + x_ny_n\theta_2), \end{aligned}$$

and hence

$$fl(y^T x) = (y+\Delta y)^T x, \quad \text{where} \quad \Delta y := [y_1\theta_n, y_2\widetilde{\theta}_n, y_3\theta_{n-1}, \dots, y_n\theta_2]^T. \tag{3.5}$$

The vector Δy is a backward error, and it satisfies the *entrywise* inequality

$$|\Delta y| \le \gamma_n |y|.$$

We see that the relative backward error of the algorithm for computing $y^T x$, which uses n multiplications and additions, is on the order of nu. We therefore call this algorithm *backward stable*. For the forward error we have the following result.

Lemma 3.10. *If $nu < 1$ and $|x| := [|x_1|, \dots, |x_n|]^T$, then*

$$|y^T x - fl(y^T x)| \le \gamma_n \sum_{i=1}^{n} |x_i y_i| = \gamma_n |y|^T |x| = nu|y|^T|x| + O(n^2u^2). \tag{3.6}$$

If $A \in \mathbb{R}^{m\times n}$, $x \in \mathbb{R}^n$ and $z = Ax$, then the ith entry of z is given by $z_i = a_i^T x$, where a_i^T is the ith row of A. As above, let $fl(z_i)$ denote the result of the computation in finite precision arithmetic (where we assume that the entries of A and x are floating point numbers), then (3.5) shows that

$$fl(z_i) = (a_i + \Delta a_i)^T x,$$

for some Δa_i, $i = 1, \ldots, m$, which satisfy $|\Delta a_i| \leq \gamma_n |a_i|$ entrywise. If ΔA is the matrix with rows $(\Delta a_i)^T$, $i = 1, \ldots, m$, then we have the backward error result

$$fl(z) = (A + \Delta A)x, \quad \text{where} \quad |\Delta A| \leq \gamma_n |A|,$$

which justifies to say that the computation of $z = Ax$ is backward stable.

3.3 Stability and cost of the Cholesky decomposition

We will next study the numerical stability of computing the Cholesky decomposition. For simplicity of notation, we will consider a real symmetric (rather than complex Hermitian) positive definite matrix A. Its Cholesky decomposition is given by

$$A = LL^T = \begin{bmatrix} l_{11} & & & \\ l_{21} & \ddots & & \\ \vdots & \ddots & \ddots & \\ l_{n1} & \cdots & l_{n,n-1} & l_{nn} \end{bmatrix} \begin{bmatrix} l_{11} & l_{21} & \cdots & l_{n1} \\ & \ddots & \ddots & \vdots \\ & & \ddots & l_{n,n-1} \\ & & & l_{nn} \end{bmatrix}, \tag{3.7}$$

where $L = [l_{ij}] \in \mathbb{R}^{n\times n}$ with $l_{ii} > 0$, $i = 1 \ldots, n$, is uniquely determined (cf. Theorem 1.3). If we equate the columns in (3.7), we immediately obtain the recursive Algorithm 2 for computing the entries of L.

Algorithm 2 Cholesky decomposition

Input: Symmetric positive definite matrix $A = [a_{ij}] \in \mathbb{R}^{n\times n}$
Output: Lower triangular matrix $L \in \mathbb{R}^{n\times n}$ with positive diagonal entries and $A = LL^T$
for $j = 1, \ldots, n$ **do**
  $l_{jj} = (a_{jj} - \sum_{k=1}^{j-1} l_{jk}^2)^{1/2}$
  $l_{ij} = \frac{1}{l_{jj}}(a_{ij} - \sum_{k=1}^{j-1} l_{ik} l_{jk})$ for $i = j+1, \ldots, n$
end for

Since the Cholesky decomposition exists, we know that Algorithm 2 *in exact arithmetic* will run to the final step n when the symmetric matrix A is positive definite. If A is not positive definite, then the algorithm must fail at some step j with $l_{jj} = a_{jj} - \sum_{k=1}^{j-1} l_{jk}^2 \leq 0$. This fact is often used as a test for positive definiteness of symmetric matrices[1].
Here is a straightforward MATLAB implementation of Algorithm 2 (including a test for (numerical) positive definiteness), which we will use in some numerical examples below:

[1]For example, a tip in the MATLAB documentation is: "Use `chol` (instead of `eig`) to efficiently determine whether a matrix is symmetric positive definite.".

```
function L=Cholesky(A)
% Computes the Cholesky factor L in the decomposition A=L*L'
n=length(A); L=zeros(n,n);
if A(1,1)<=0
    error('The matrix is (numerically) not positive definite')
end
L(1,1)=sqrt(A(1,1));
L(2:n,1)=(1/L(1,1))*A(2:n,1);
for j=2:n
    u=A(j,j)-sum(L(j,1:j-1).^2);
    if u<=0
        error('The matrix is (numerically) not positive definite')
    else
        L(j,j)=sqrt(u);
        L(j+1:n,j)=(1/L(j,j))*(A(j+1:n,j)-L(j+1:n,1:j-1)*L(j,1:j-1)');
    end
end
```

Example 3.11. *The* $n \times n$ Hilbert matrix *is given by* $A = [a_{ij}] = [1/(i+j-1)] \in \mathbb{R}^{n \times n}$. *In MATLAB we have, for example,*

```
>> A=hilb(4)
A =
    1.0000    0.5000    0.3333    0.2500
    0.5000    0.3333    0.2500    0.2000
    0.3333    0.2500    0.2000    0.1667
    0.2500    0.2000    0.1667    0.1429
```

The Hilbert matrices are symmetric and positive definite for all n. *However, with increasing* n *they become increasingly ill-conditioned. For example:*

```
>> A=hilb(14); cond(A)
ans =
    2.551498848378212e+17
>> norm(A)
ans =
    1.830594695920394
```

Thus, for the 14×14 *Hilbert matrix* A *we have* $\|A^{-1}\|_2 = 1/\lambda_{\min}(A) \approx 10^{18}$, *and therefore* $\lambda_{\min}(A) \approx 10^{-18}$, *which is significantly below the unit roundoff. Applying MATLAB's Cholesky factorization algorithm to* A *yields an error, although* A *is (mathematically) positive definite:*

```
>> L=chol(A,'lower')
Error using chol
Matrix must be positive definite.
```

Example 3.12. *A large condition number alone does not necessarily lead to a failure of Algorithm 2. As an example consider the* Pascal matrix $A \in \mathbb{R}^{n\times n}$, *which is symmetric positive definite with integer entries taken from Pascal's triangle. For example, the* 4×4 *Pascal matrix is*

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 \\ 1 & 3 & 6 & 10 \\ 1 & 4 & 10 & 20 \end{bmatrix}.$$

With increasing n *these matrices become increasingly ill-conditioned, but nevertheless the MATLAB implementation of Algorithm 2 given above produces a very accurate result. For example:*

```
>> A=pascal(20); cond(A)
ans =
     2.246502328748667e+21
>> L=Cholesky(A); norm(A-L*L')
ans =
     0
```

Suppose that A_R *is the transposed of the Pascal matrix* A *with respect to its anti-diagonal, i.e.,* $A_R = I_R A I_R$ *where* $I_R = [\delta_{i,n-i+1}]$ *is the* reverse identity matrix. *For example, in the* 4×4 *case we have*

$$A_R = I_R A I_R = \begin{bmatrix} 20 & 10 & 4 & 1 \\ 10 & 6 & 3 & 1 \\ 4 & 3 & 2 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

Clearly, A *and* A_R *have the same eigenvalues and condition number. Note, however, that the sizes of the entries of* A_R *decay similarly as those of a Hilbert matrix; see Example 3.11.*
Computations with the MATLAB implementation of Algorithm 2 given above and with MATLAB's Cholesky factorization algorithm appear to give a less accurate result than for the original Pascal matrix A*:*

```
>> L=Cholesky(A_R); norm(A_R-L*L')
ans =
     2.221091901490078e-06
```

```
>> L=chol(A_R,'lower'); norm(A_R-L*L')
ans =
     1.984839037382334e-06
```

The relative (backward) error norm, however, is still on the order of the machine epsilon:

```
>> L=Cholesky(A_R); norm(A_R-L*L')/norm(A_R)
ans =
     4.726246478141171e-17
>> L=chol(A_R,'lower'); norm(A_R-L*L')/norm(A_R)
ans =
     4.223525601895171e-17
```

This outcome will be confirmed by the rounding error analysis of Algorithm 2 in the following; see in particular Corollary 3.14.

The rounding error analysis of Algorithm 2 can be based on the same ideas that led to (3.6). Let us first consider the second of the two main steps of the algorithm, i.e.,

$$l_{ij} = \Big(a_{ij} - \sum_{k=1}^{j-1} l_{ik} l_{jk}\Big)/l_{jj},$$

which can be computed as follows:

$s = a_{ij}$
for $k = 1, \ldots, j-1$ **do**
  $s = s - l_{ik} l_{jk}$
end for
$l_{ij} = s/l_{jj}$

We are interested in the computed value $\widehat{l_{ij}}$, and at this step in the algorithm we have already computed all entries of the Cholesky factor that appear in the for-loop. Thus, in a finite precision computation this loop uses the computed values $\widehat{l}_{ik}$ and $\widehat{l}_{jk}$. If $\widehat{s}_k$ denotes the computed term after $k = 1, 2, \ldots$ steps of the loop, then analogously to the derivation of (3.6) and under the assumption that the entries of A are floating point numbers, there exist $|\delta_i|, |\epsilon_i| \leq u$ such that

$$\begin{aligned}
\widehat{s}_1 &= (a_{ij} - \widehat{l}_{i1}\widehat{l}_{j1}(1+\epsilon_1))(1+\delta_1), \\
\widehat{s}_2 &= (\widehat{s}_1 - \widehat{l}_{i2}\widehat{l}_{j2}(1+\epsilon_2))(1+\delta_2) \\
&= a_{ij}(1+\delta_1)(1+\delta_2) - \widehat{l}_{i1}\widehat{l}_{j1}(1+\epsilon_1)(1+\delta_1)(1+\delta_2) - \widehat{l}_{i2}\widehat{l}_{j2}(1+\epsilon_2)(1+\delta_2).
\end{aligned}$$

After the $j-1$ steps of the loop we have

$$\widehat{s}_{j-1} = a_{ij}(1+\delta_1)\cdots(1+\delta_{j-1}) - \sum_{k=1}^{j-1} \widehat{l}_{ik}\widehat{l}_{jk}(1+\epsilon_k)(1+\delta_k)\cdots(1+\delta_{j-1}). \tag{3.8}$$

Consequently,

$$\widehat{l}_{ij} = fl(\widehat{s}_{j-1}/\widehat{l}_{jj}) = \frac{\widehat{s}_{j-1}}{\widehat{l}_{jj}}(1+\delta_j),$$

so that

$$\widehat{l}_{jj}\widehat{l}_{ij}\frac{1}{(1+\delta_1)\cdots(1+\delta_j)} = a_{ij} - \sum_{k=1}^{j-1}\widehat{l}_{ik}\widehat{l}_{jk}\frac{1+\epsilon_k}{(1+\delta_1)\cdots(1+\delta_{k-1})}.$$

A simple variation of Lemma 3.9 shows that there exist some $|\theta_k| \le \gamma_k$ for $k = 1, \ldots, j$, such that

$$\widehat{l}_{jj}\widehat{l}_{ij}(1+\theta_j) = a_{ij} - \sum_{k=1}^{j-1}\widehat{l}_{ik}\widehat{l}_{jk}(1+\theta_k).$$

This can be rewritten as

$$a_{ij} - \sum_{k=1}^{j}\widehat{l}_{ik}\widehat{l}_{jk} = \sum_{k=1}^{j}\widehat{l}_{ik}\widehat{l}_{jk}\theta_k,$$

and analogously to (3.6) we obtain

$$|a_{ij} - \widehat{l}_i^T\widehat{l}_j| \le \gamma_j|\widehat{l}_i|^T|\widehat{l}_j|, \quad \text{for } i = j+1, \ldots, n, \tag{3.9}$$

where $\widehat{l}_i^T$ is the ith row of the computed Cholesky factor $\widehat{L}$, and $\widehat{l}_j$ is the jth column of $\widehat{L}^T$. In the first of the two main steps of Algorithm 2 we compute the diagonal entries of the Cholesky factor,

$$l_{jj} = \Big(a_{jj} - \sum_{k=1}^{j-1} l_{jk}^2\Big)^{1/2}.$$

The term in brackets on the right is computed by a loop as the one studied above, and after $j-1$ steps we obtain the computed value

$$\widehat{s}_{j-1} = a_{jj}(1+\delta_1)\cdots(1+\delta_{j-1}) - \sum_{k=1}^{j-1}\widehat{l}_{jk}^2(1+\epsilon_k)(1+\delta_k)\cdots(1+\delta_{j-1}); \tag{3.10}$$

see (3.8). We then take a square root to obtain the diagonal entry. Assuming that this operation like the operations in (3.2) we obtain

$$\widehat{l}_{jj} = fl(\widehat{s}_{j-1}^{1/2}) = \widehat{s}_{j-1}^{1/2}(1+\delta_j),$$

so that

$$\widehat{l}_{jj}^2 = \widehat{s}_{j-1}(1+\delta_j)^2.$$

Now (3.10) and similar arguments as above give

$$\widehat{l}_{jj}^2\frac{1}{(1+\delta_1)\cdots(1+\delta_{j-1})(1+\delta_j)^2} = a_{jj} - \sum_{k=1}^{j-1}\widehat{l}_{jk}^2\frac{1+\epsilon_k}{(1+\delta_1)\cdots(1+\delta_{k-1})},$$

and

$$\widehat{l}_{jj}^2(1+\theta_{j+1}) = a_{jj} - \sum_{k=1}^{j-1} \widehat{l}_{jk}^2(1+\theta_k).$$

Finally,

$$|a_{jj} - \widehat{l}_j^T \widehat{l}_j| \le \gamma_{j+1} |\widehat{l}_j|^T |\widehat{l}_j|, \quad \text{for } j = 1, \ldots, n. \tag{3.11}$$

Note that here we have the factor γ_{j+1} (instead of γ_j as in (3.9)). Combining (3.9) and (3.11) gives the following result; see [16, Theorem 10.3].

Theorem 3.13. *If $nu < 1$ and Algorithm 2 applied to a symmetric positive definite matrix $A \in \mathbb{R}^{n\times n}$ (with floating point entries) runs to completion, then the computed Cholesky factor $\widehat{L}$ satisfies*

$$A = \widehat{L}\widehat{L}^T + \Delta A, \quad \textit{where} \quad |\Delta A| \le \gamma_{n+1} |\widehat{L}|\,|\widehat{L}^T|.$$

The inequality on the right is meant entrywise.

The bound on the entries of the backward error matrix ΔA in Theorem 3.13 depends on the entries of the computed Cholesky factor $\widehat{L}$, which appears to be not very useful at first sight. However, we will now show that this bound implies that the Cholesky factorization algorithm is *backward stable* in the sense that the (normwise) relative backward error $\|\Delta A\|/\|A\|$ is close to the unit roundoff.
For $1 \le p \le \infty$, let $\|\cdot\|_p$ be the norm on $\mathbb{C}^{n\times n}$ induced by the p-norm on $\mathbb{C}^n$ (see (0.1)), i.e.,

$$\|A\|_p = \max_{\|x\|_p = 1} \|Ax\|_p.$$

Then, in particular,

$$\|A\|_1 = \max_{1\le j\le n} \sum_{i=1}^{n} |a_{ij}| \le n^{1-1/p}\|A\|_p \quad \text{and} \quad \|A\|_\infty = \max_{1\le i\le n} \sum_{j=1}^{n} |a_{ij}| \le n^{1/p}\|A\|_p.$$

Moreover, for each $A \in \mathbb{C}^{n\times n}$ we have

$$\|A\|_p \le \|A\|_1^{1/p} \|A\|_\infty^{1-1/p},$$

which reminds of the Hölder inequality (2.10). Since $\|A\|_1 = \||A|\|_1$ and $\|A\|_\infty = \||A|\|_\infty$, we get

$$\||A|\|_p \le \||A|\|_1^{1/p} \||A|\|_\infty^{1-1/p} = \|A\|_1^{1/p} \|A\|_\infty^{1-1/p} \le n^{2(1-1/p)/p} \|A\|_p,$$

and in particular $\||A|\|_2 \le n^{1/2}\|A\|_2$.
If $M \in \mathbb{R}^{n\times n}$ has an SVD of the form $M = U\Sigma V^T$, then $MM^T = U\Sigma^2 U^T$, from which we see that $\|MM^T\|_2 = \|M\|_2^2$. Thus,

$$\begin{aligned}\||\widehat{L}||\widehat{L}^T|\|_2 = \||\widehat{L}|\|_2^2 \le n\|\widehat{L}\|_2^2 = n\|\widehat{L}\widehat{L}^T\|_2 = n\|A - \Delta A\|_2 \le n(\|A\|_2 + \|\Delta A\|_2) \\ \le n(\|A\|_2 + \gamma_{n+1}\||\widehat{L}||\widehat{L}^T|\|_2),\end{aligned}$$

from which we obtain

$$\||\widehat{L}||\widehat{L}^T|\|_2 \leq \frac{n}{1-n\gamma_{n+1}}\|A\|_2, \tag{3.12}$$

and hence, using (3.4),

$$\|\Delta A\|_2 \leq \||\Delta A|\|_2 \leq \gamma_{n+1}\||\widehat{L}||\widehat{L}^T|\|_2 \leq \frac{n\gamma_{n+1}}{1-n\gamma_{n+1}}\|A\|_2 = \left(n^2u + O(n^3u^2)\right)\|A\|_2.$$

We thus have shown the following result for the (normwise) relative backward error.

Corollary 3.14. *The computed Cholesky factor $\widehat{L}$ satisfies*

$$A = \widehat{L}\widehat{L}^T + \Delta A, \quad \textit{where} \quad \frac{\|\Delta A\|_2}{\|A\|_2} \leq \frac{n\gamma_{n+1}}{1-n\gamma_{n+1}} = n^2u + O(n^3u^2).$$

If A does not have floating point entries, then its entries are first rounded, and Algorithm 2 is applied to

$$\widehat{A} = fl([a_{ij}]) = [fl(a_{ij})] = [a_{ij}(1+\varepsilon_{ij})] = A + [a_{ij}\varepsilon_{ij}], \quad \text{where } |\varepsilon_{ij}| \leq u.$$

We then have computed $\widehat{A} = \widehat{L}\widehat{L}^T + \Delta A$, and with $E := -[\varepsilon_{ij}]$ we obtain

$$A = \widehat{L}\widehat{L}^T + (\Delta A + A \circ E),$$

where $A \circ E$ denotes the entrywise (or Hadamard) product. We have $\|E\|_1 \leq nu$ and $\|E\|_\infty \leq nu$. Using $\|E\|_2 \leq \|E\|_1^{1/2}\|E\|_\infty^{1/2}$, and $\|A \circ E\|_2 \leq \|A\|_2\|E\|_2$ (see, e.g., [17]), we can bound

$$\|A \circ E\|_2 \leq nu\|A\|_2,$$

which yields

$$\frac{\|A \circ E\|_2}{\|A\|_2} \leq nu < n^2u.$$

We see that (potential) errors introduced by the initial rounding are negligible in comparison with the (potential) rounding errors in the computation. Consequently, the backward error bound in Corollary 3.14 also applies to an SPD matrix A with real entries (in the range of the given floating point numbers) that first have to be rounded.

Similar bounds as in Corollary 3.14 can be derived for other matrix norms. All these bounds will be of the form $\|\Delta A\|/\|A\| \leq p(n)u$, where $p(n)$ is a polynomial of small degree in n. *These bounds show that the Algorithm 2 for computing the Cholesky decomposition is (normwise) backward stable.*

Note that, in particular, the upper bound on the backward error does not depend on the condition number of the given matrix A. Moreover, in the derivation of the bound it is assumed that in every single operation a "largest possible" rounding error occurs. We can therefore expect that the bound is rather pessimistic in many examples, which is confirmed in the following experiment.

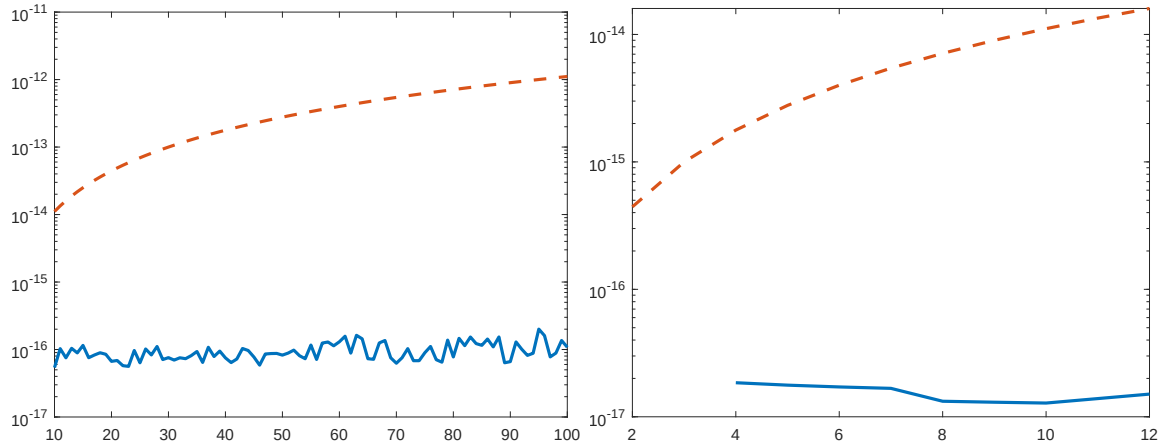


Figure 3.1: Relative backward error norms $\|\Delta A\|_2/\|A\|_2$ of the computed Cholesky factor (solid) and n^2u (dashed) for a sequence of random matrices of orders $n = 10, 11, , \ldots, 100$ (left) and Hilbert matrices of orders $n = 2, 3, \ldots, 12$ (right).

Example 3.15. *We apply the MATLAB implementation of Algorithm 2 given above to "random" symmetric positive definite matrices generated in MATLAB by* `A=randn(n); A=A*A';` *and to the Hilbert matrices of orders* $n = 1, 2, \ldots, 12$*; see Example 3.11. In Figure 3.1 we plot the relative backward error norms* $\|\Delta A\|_2/\|A\|_2$ *and the factors* n^2u *from the upper bound given in Corollary 3.14 by the dashed and solid lines, respectively. For the Hilbert matrices of orders* $n = 2$ *and* $n = 3$ *the backward error is zero. The algorithm runs to completion for all* $n \times n$ *Hilbert matrices up to* $n = 12$*. The (2-norm) condition number of the* 12×12 *Hilbert matrix is approximately* 1.6×10^{16}*; cf. the comments in Example 3.11. For all matrices the relative backward error norm is close to the unit roundoff* u *(approximately* 10^{-16}*), which is significantly below the term* n^2u *from the bound in Corollary 3.14*

As mentioned above, if we have given the Cholesky decomposition $A = LL^T$, we can compute an approximation of the solution of $Ax = b$ by solving the two nonsingular triangular systems

(1) $Ly = b$ (hence in exact arithmetic $y = L^{-1}b$),

(2) $L^Tx = y$ (hence in exact arithmetic $x = L^{-T}y = A^{-1}b$).

A lower (or upper) triangular system can be solved using forward (or back) substitution. The forward substitution algorithm used for solving $Ly = b$ can be written as

$$y_j = \frac{1}{l_{jj}}\Big(b_j - \sum_{k=1}^{j-1} l_{jk}y_k\Big), \quad j = 1, \ldots, n.$$

Evaluating the right hand side costs j multiplications and $j-1$ subtractions, and hence the total cost of the forward substitution algorithm is $\sum_{j=1}^{n}(2j-1) = n^2$.
In finite precision computations each operation is affected by rounding errors, and hence the algorithm does not yield the exact solution $y = [y_1, \ldots, y_n]^T$ but a computed approximation $\widehat{y} = [\widehat{y}_1, \ldots, \widehat{y}_n]^T$. Thus, for the rounding error analysis we must consider the recurrence

$$\widehat{y}_j = \frac{1}{l_{jj}}\Big(b_j - \sum_{k=1}^{j-1} l_{jk}\widehat{y}_k\Big), \quad j = 1, \ldots, n,$$

which can also be written as

$$b_j = \sum_{k=1}^{j} l_{jk}\widehat{y}_k, \quad j = 1, \ldots, n.$$

This version shows that the numerical stability analysis of the algorithm can be done as for the Algorithm 2. Obviously, the same analysis applies to the backward substitution, i.e., the solution of an upper triangular system, and this leads to the following result.

Theorem 3.16. *If the linear algebraic system $Ty = b$ with the lower (or upper) triangular matrix $T \in \mathbb{R}^{n\times n}$ and $b \in \mathbb{R}^n$ is solved by the forward (or back) substitution algorithm as stated above, then the computed approximation $\widehat{y}$ satisfies*

$$(T + \Delta T)\widehat{y} = b, \quad \textit{where} \quad |\Delta T| \le \gamma_n |T|.$$

Using the same analysis as above one can now show that the relative backward errors in the forward and back substitution algorithms satisfy bounds of the form $\|\Delta T\|/\|T\| \le p(n)u$, where $\|\cdot\|$ is an appropriate matrix norm, and $p(n)$ is a polynomial of small degree in n. Thus, both algorithms are (normwise) backward stable.
Consequently, when we solve $Ax = b$ with an SPD matrix $A \in \mathbb{R}^{n\times n}$ and $b \in \mathbb{R}^n$ (which both have floating point entries) in finite precision arithmetic by first computing $A = \widehat{L}\widehat{L}^T + \Delta A_1$, and then computing approximations to the solutions of $\widehat{L}y = b$ and $\widehat{L}^T x = y$ by forward and back substitution, we obtain an approximation $\widehat{x}$ such that

$$(A + \Delta A_2)\widehat{x} = b, \quad \text{where} \quad \frac{\|\Delta A_2\|}{\|A\|} \le p(n)u;$$

see, e.g., [16, Theorem 10.4 and equation (10.7)]. Thus, the norm of the residual satisfies

$$\|r\| = \|b - A\widehat{x}\| = \|\Delta A_2 \widehat{x}\| \le p(n)\|A\|\|\widehat{x}\|u,$$

giving

$$\frac{\|r\|}{\|A\|\|\widehat{x}\| + \|b\|} \le p(n)u\frac{\|A\|\|\widehat{x}\|}{\|A\|\|\widehat{x}\| + \|b\|} < p(n)u, \tag{3.13}$$

and hence this method for solving $Ax = b$ is normwise backward stable in the sense of Theorem 2.26.

We will now have a look at an alternative algorithm, which follows the second proof of the existence of the Cholesky decomposition in Corollary 1.3, and is well suited to analyse the computational cost for computing the decomposition. We write the SPD matrix $A \in \mathbb{R}^{n\times n}$ in the form

$$A = \begin{bmatrix} a_{11} & a_1^T \\ a_1 & A_1 \end{bmatrix}.$$

Since $e_1^T A e_1 = a_{11} > 0$, we can perform the following basic factorization step:

$$A = \begin{bmatrix} a_{11} & a_1^T \\ a_1 & A_1 \end{bmatrix} = \underbrace{\begin{bmatrix} a_{11}^{1/2} & 0 \\ a_{11}^{-1/2} a_1 & I_{n-1} \end{bmatrix}}_{=:L_1} \begin{bmatrix} 1 & 0 \\ 0 & S_1 \end{bmatrix} \underbrace{\begin{bmatrix} a_{11}^{1/2} & a_{11}^{-1/2} a_1^T \\ 0 & I_{n-1} \end{bmatrix}}_{=:L_1^T},$$

where $S_1 := A_1 - \frac{a_1 a_1^T}{a_{11}} \in \mathbb{R}^{(n-1)\times(n-1)}$ is called the *Schur complement* of a_{11} in A.

The symmetric matrices A and $\begin{bmatrix} 1 & 0 \\ 0 & S_1 \end{bmatrix}$ are congruent and hence have the same inertia (i.e., number of positive, negative and zero eigenvalues). Since A is SPD, S_1 must be SPD as well, so that we can perform the basic factorization step on S_1:

$$S_1 = \begin{bmatrix} b_{11} & b_1^T \\ b_1 & B_1 \end{bmatrix} = \underbrace{\begin{bmatrix} b_{11}^{1/2} & 0 \\ b_{11}^{-1/2} b_1 & I_{n-2} \end{bmatrix}}_{=:\widetilde{L}_2} \begin{bmatrix} 1 & 0 \\ 0 & S_2 \end{bmatrix} \underbrace{\begin{bmatrix} b_{11}^{1/2} & b_{11}^{-1/2} b_1^T \\ 0 & I_{n-2} \end{bmatrix}}_{=:\widetilde{L}_2^T},$$

where $S_2 := B_1 - \frac{b_1 b_1^T}{b_{11}} \in \mathbb{R}^{(n-2)\times(n-2)}$ is the Schur complement of b_{11} in S_1. With

$$L_2 := \begin{bmatrix} 1 & 0 & 0 \\ 0 & b_{11}^{1/2} & 0 \\ 0 & b_{11}^{-1/2} b_1 & I_{n-2} \end{bmatrix}$$

we then have a factorization of the form

$$A = L_1 L_2 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & S_2 \end{bmatrix} L_2^T L_1^T,$$

The Schur complement S_2 must also be SPD, so that we can again apply the basic factorization step. After n steps we obtain the Cholesky decomposition

$$A = (L_1 L_2 \cdots L_n)(L_n^T \cdots L_2^T L_1^T) =: L L^T.$$

Note that in the nth step we only take one square root and do not form a Schur complement.

By construction, each matrix L_j has the form

$$L_j = \begin{bmatrix} I_{j-1} & 0 & 0 \\ 0 & * & 0 \\ 0 & * & I_{n-j} \end{bmatrix} = I_n + \begin{bmatrix} 0_{j-1} \\ l_j \end{bmatrix} e_j^T,$$

where $0_{j-1} \in \mathbb{R}^{j-1}$ has zero entries, and $l_j \in \mathbb{R}^{n-j+1}$, $j = 1, \ldots, n$. Therefore, for example,

$$L_1 L_2 = (I_n + l_1 e_1^T)\left(I_n + \begin{bmatrix} 0 \\ l_2 \end{bmatrix} e_2^T\right) = I_n + l_1 e_1^T + \begin{bmatrix} 0 \\ l_2 \end{bmatrix} e_2^T,$$

and inductively

$$L = L_1 L_2 \cdots L_n = I_n + \sum_{j=1}^{n} \begin{bmatrix} 0_{j-1} \\ l_j \end{bmatrix} e_j^T.$$

This shows that when computing the factorization using the Schur complement approach we do not have to compute the product of the matrices L_j. Instead, we form L by simply writing the vectors l_j into the lower triangle for $j = 1, \ldots, n$. (A similar observation will be made in the computation of the LU decomposition; see the following section.)

The main computational cost in computing the factorization is in forming the Schur complements S_j, which are of the form $M - \frac{1}{\alpha} v v^T \in \mathbb{R}^{(n-j)\times(n-j)}$. This requires the following types of operations in steps $j = 1, \ldots, n-1$:

(1) $\frac{1}{\alpha} v$ with $v \in \mathbb{R}^{n-j}$: $n - j$ multiplications.

(2) $(\frac{1}{\alpha} v) v^T$: $\frac{(n-j)(n-j+1)}{2}$ multiplications. (Since this matrix is symmetric, only its upper (or lower) triangular part needs to be computed.)

(3) $S_j = M - (\frac{1}{\alpha} v v^T)$ with $M \in \mathbb{R}^{(n-j)\times(n-j)}$: $\frac{(n-j)(n-j+1)}{2}$ operations. (Again, this matrix is symmetric.)

Disregarding the square roots, the total cost for computing the Cholesky decomposition using the factorization approach described above is

$$\begin{aligned}
\sum_{j=1}^{n-1} [(n-j)(n-j+1) + (n-j)] &= \sum_{j=1}^{n-1} (n-j)(n-j+2) \\
&= \sum_{k=1}^{n-1} k(k+2) = \sum_{k=1}^{n-1} k^2 + 2\sum_{k=1}^{n-1} k \\
&= \frac{(n-1)n(2(n-1)+1)}{6} + 2\frac{(n-1)n}{2} \\
&= \frac{1}{6}(n(n-1)(2n-1) + 6(n^2-n)) \\
&= \frac{1}{6}(2n^3 + 3n^2 - 5n) \approx \frac{1}{3} n^3 \ \text{(for large } n\text{)}.
\end{aligned}$$

Many applications involve *sparse* matrices, i.e., matrices with a significant number of zero entries. Zero entries need not be stored, and they do not take part in numerical evaluations (multiplication, addition, subtraction). When A is sparse, we also would like to have a sparse Cholesky factor L.

When $a_{ij} = 0$ but $l_{ij} \neq 0$, the element l_{ij} is called a *fill-in* element. When A is SPD, then $P^T AP$ is SPD for any permutation matrix P. An important line of research in the context of the (sparse) Cholesky decomposition is concerned with finding permutations P so that the Cholesky factor of $P^T AP$ has the least possible number of nonzero entries. In this context we speak of the *sparse Cholesky decomposition.*

3.4 Computing the LU decomposition

Let us now consider a general (nonsingular) matrix $A \in \mathbb{C}^{n\times n}$. If the leading principal minors $A(1:k, 1:k) \in \mathbb{C}^{k\times k}$ are nonsingular for all $k = 1, \ldots, n$, then the decomposition

$$A = LU$$

with a unit lower triangular matrix $L \in \mathbb{C}^{n\times n}$ and a nonsingular upper triangular matrix $U \in \mathbb{C}^{n\times n}$ exists; see Theorem 1.1. Then $x = A^{-1}b = U^{-1}(L^{-1}b)$, so that we can again compute x (or rather an approximation $\widehat{x}$) using the forward and back substitution algorithms, which are normwise backward stable.
The LU decomposition can be computed by *Gaussian elimination*:
At step $j = 1, \ldots, n-1$, multiples of the j-th row are subtracted from rows $j+1, \ldots, n$ in order to introduce zeros in the column j below the entry in position (j, j), and the result is the upper triangular matrix U. Schematically:

$$A = \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} \to \begin{bmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & \times & \times & \times \end{bmatrix} \to \begin{bmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & \times & \times \end{bmatrix}$$
$$\to \begin{bmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times \end{bmatrix} = U$$

Each step j in the process can be considered one left-multiplication of A by a suitable unit lower triangular matrix L_j. In step $j = 1$ we have

$$\underbrace{\begin{bmatrix} 1 & 0 \\ \frac{-1}{a_{11}} v & I_{n-1} \end{bmatrix}}_{=:L_1} \underbrace{\begin{bmatrix} a_{11} & w^H \\ v & A_1 \end{bmatrix}}_{=A} = \underbrace{\begin{bmatrix} a_{11} & w^H \\ 0 & S_1 \end{bmatrix}}_{=:U_1}, \quad \text{where} \quad S_1 := A_1 - \frac{1}{a_{11}} v w^H.$$

By the assumption on A, we are guaranteed that $a_{11} \neq 0$ and that the leading principal minors of S_1 are nonsingular. Hence the process can be continued with the matrix S_1. After $n-1$ steps we obtain

$$L_{n-1} \cdots L_2 L_1 A = U \quad \text{or} \quad A = (L_1^{-1} L_2^{-1} \cdots L_{n-1}^{-1}) U =: LU,$$

where $L_1, \ldots, L_{n-1} \in \mathbb{C}^{n\times n}$ and hence $L = L_1^{-1}L_2^{-1}\cdots L_{n-1}^{-1}$ are unit lower triangular, and $U \in \mathbb{C}^{n\times n}$ is nonsingular and upper triangular.
Similar to what we have seen for the Schur complement approach for computing the Cholesky factorization, each matrix L_j is of the form

$$L_j = \left[\begin{array}{c|c|c} I_{j-1} & & \\ \hline & 1 & \\ \hline & l_j & I_{n-j} \end{array}\right] = I_n + \begin{bmatrix} 0_j \\ l_j \end{bmatrix} e_j^T,$$

where $0_j \in \mathbb{R}^j$ has zero entries, and $l_j \in \mathbb{C}^{n-j}$, and hence

$$L_j^{-1} = I_n - \begin{bmatrix} 0_j \\ l_j \end{bmatrix} e_j^T, \tag{3.14}$$

which shows that $L = L_1^{-1}L_2^{-1}\cdots L_{n-1}^{-1}$ can be easily computed from $L_1, \ldots, L_{n-1}$. Moreover,

$$L_1^{-1}L_2^{-1} = \left(I_n - \begin{bmatrix} 0_1 \\ l_1 \end{bmatrix} e_1^T\right)\left(I_n - \begin{bmatrix} 0_2 \\ l_2 \end{bmatrix} e_2^T\right) = I_n - \begin{bmatrix} 0_1 \\ l_1 \end{bmatrix} e_1^T - \begin{bmatrix} 0_2 \\ l_2 \end{bmatrix} e_2^T,$$

and inductively

$$L = I_n - \sum_{j=1}^{n-1} \begin{bmatrix} 0_j \\ l_j \end{bmatrix} e_j^T. \tag{3.15}$$

In [42, pp. 149–151], the simple form (3.14) of the inverse L_j^{-1} and the simple form (3.15) of the product of these inverses are called (the first) "two strokes of luck" of Gaussian elimination.

The main cost in the algorithm described above is in forming the Schur complement matrices $S_j \in \mathbb{C}^{(n-j)\times(n-j)}$, $j = 1, \ldots, n-1$. For a nonsymmetric (or non-Hermitian) matrix this is (approximately) twice as expensive as forming the symmetric (or Hermitian) Schur complement in the algorithm for computing the Cholesky decomposition. The cost for computing the LU decomposition (for large n) therefore is approximately $\frac{2}{3}n^3$.

Assuming that the decomposition $A = LU$ exists, and that the above algorithm runs to completion, we can perform a rounding error analysis similar to the analysis for the Cholesky decomposition. This analysis can be based on writing the decomposition in the (inner product) form

$$a_{ij} = \sum_{k=1}^{i} l_{ik}u_{kj}, \tag{3.16}$$

which leads, analogously to (3.9) and hence Theorem 3.13, in a backward error result of the form

$$A = \widehat{L}\widehat{U} + \Delta A, \quad \text{where} \quad |\Delta A| \le \gamma_n |\widehat{L}||\widehat{U}|;$$

see, e.g., [16, Theorem 9.3].

However, on the contrary to the Cholesky decomposition, the sizes of the entries in the factors $|\widehat{L}|$ and $|\widehat{U}|$ are not bounded by $\|A\|$, and we can not derive a bound analogous to (3.12). For example, the matrix

$$A = \begin{bmatrix} \varepsilon & 1 \\ 1 & 1 \end{bmatrix}, \quad \varepsilon \notin \{0, 1\},$$

has nonsingular leading principal minors, and the first step of the algorithm gives

$$\begin{bmatrix} 1 & 0 \\ -\varepsilon^{-1} & 1 \end{bmatrix} \begin{bmatrix} \varepsilon & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} \varepsilon & 1 \\ 0 & 1-\varepsilon^{-1} \end{bmatrix},$$

so that

$$\begin{bmatrix} \varepsilon & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \varepsilon^{-1} & 1 \end{bmatrix} \begin{bmatrix} \varepsilon & 1 \\ 0 & 1-\varepsilon^{-1} \end{bmatrix} =: LU.$$

For $|\varepsilon| \to 0$ the largest entries in the factors L and U grow unboundedly, hence $\|L\|$ and $\|U\|$ become arbitrarily large, while the largest entry in the matrix A is 1. This can lead to numerical instabilities:

Example 3.17. *We use $\varepsilon = 10^{-16}$ and compute the product LU in MATLAB:*

```
>> format longe
>> e=1e-16; L=[1 0;1/e 1]; U=[e 1; 0 1-1/e]; L*U
ans =
     1.000000000000000e-16     1.000000000000000e+00
     1.000000000000000e+00                         0
```

The (2,2) entry of the product is evaluated as $\varepsilon^{-1} + (1 - \varepsilon^{-1})$, and the finite precision result is (exactly) zero.

In order to control the sizes of the entries in the factors one can use *pivoting*[2]. We assume that A is nonsingular. (It will turn out that the nonsingularity assumption on the leading principal minors is not required when pivoting is used.) After $j \geq 0$ steps of the above algorithm we have

$$L_j \cdots L_1 A = \left[\begin{array}{c|ccc} U_j & * & & \\ \hline 0 & u_{j+1,j+1}^{(j)} & * \cdots & * \\ & \vdots & \vdots & \vdots \\ & u_{n,j+1}^{(j)} & * \ldots & * \end{array}\right], \tag{3.17}$$

[2]According to the Merriam-Webster Dictionary, a *pivot* is "a person, thing, or factor having a major or central role, function, or effect". This real-life definition fits well to the mathematical meaning of the pivot in Gaussian elimination, which is described in this paragraph.

where $U_j \in \mathbb{C}^{j\times j}$ is upper triangular. (Here U_0 is the empty matrix.) Since A is nonsingular, at least one of the entries $u^{(j)}_{j+1,j+1}, \ldots, u^{(j)}_{n,j+1}$ must be nonzero. If not, then first $j+1$ columns of $L_j \cdots L_1 A$ are linearly dependent, so that this matrix and hence A is singular. In the strategy of *partial (or row) pivoting*, we select an entry $u^{(j)}_{k,j+1}$, $1 \le j+1 \le k \le n$, of maximum modulus among these entries. If this is not the entry $u^{(j)}_{j+1,j+1}$, i.e. if $j+1 < k$, then we exchange the rows $j+1$ and k, and form the elimination matrix L_{j+1} using the submatrix with the exchanged rows. When forming L_{j+1} we divide the first column of the corresponding submatrix by the *pivot* $u^{(j)}_{k,j+1}$, which has the largest magnitude in that column. Consequently, all entries in the matrix L_{j+1}, and hence all entries in L^{-1}_{j+1}, are bounded in modulus by 1.
In matrix notation, the exchange of rows $j+1$ and k, where $1 \le j+1 < k \le n$, corresponds to a left-multiplication by the permutation matrix

$$P_{j+1,k} := [e_1, \ldots, e_j, e_k, e_{j+2}, \ldots, e_{k-1}, e_{j+1}, e_{k+1}, \ldots, e_n].$$

For any nonsingular matrix $A \in \mathbb{C}^{n\times n}$, the Gaussian elimination algorithm with partial pivoting therefore produces a factorization of the form

$$L_{n-1} P_{n-1,k_{n-1}} \cdots L_2 P_{2,k_2} L_1 P_{1,k_1} A = U,$$

where $L_j = [l^{(j)}_{st}]$ is unit lower triangular with $|l^{(j)}_{st}| \le 1$ for all s, t, and U is upper triangular. A permutation matrix P_{j,k_j} only appears when a row exchange was made.

For example, if $0 < |\varepsilon| < 1$, then

$$\underbrace{\begin{bmatrix} 1 & 0 \\ -\varepsilon & 1 \end{bmatrix}}_{=:L_1} \underbrace{\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}}_{=P_{12}} \underbrace{\begin{bmatrix} \varepsilon & 1 \\ 1 & 1 \end{bmatrix}}_{=A} = \underbrace{\begin{bmatrix} 1 & 1 \\ 0 & 1-\varepsilon \end{bmatrix}}_{=:U},$$

or $A = PLU$, where $L = L_1^{-1}$ and $P = P_{12}^T = P_{12}^{-1}$.

If a row exchange is made in step 2, then $1 < 2 < k_2 \le n$ and we have $e_1^T P_{2,k_2} = e_1^T$, which gives

$$P_{2,k_2} L_1 = P_{2,k_2} + P_{2,k_2} \begin{bmatrix} 0_1 \\ l_1 \end{bmatrix} e_1^T = \left(I_n + \begin{bmatrix} 0_1 \\ \widetilde{l}_1 \end{bmatrix} e_1^T \right) P_{2,k_2},$$

where $\widetilde{l}_1$ and l_1 have the same entries, except for two permuted ones. More generally, it follows that

$$\begin{aligned} L_{n-1} P_{n-1,k_{n-1}} \cdots L_2 P_{2,k_2} L_1 P_{1,k_1} &= \widetilde{L}_{n-1} \cdots \widetilde{L}_2 \widetilde{L}_1 P_{n-1,k_{n-1}}, \cdots P_{2,k_2} P_{1,k_1} \\ &= \widetilde{L} \widetilde{P}, \end{aligned}$$

where $\widetilde{L} = [\widetilde{l}_{ij}]$ is unit lower triangular with $|\widetilde{l}_{ij}| \le 1$ and the same structure as the matrix L in (3.15), and P is a permutation matrix. In [42, pp. 159–160] this is called the "third stroke of luck" of Gaussian elimination. From $\widetilde{L}\widetilde{P}A = U$ we now obtain $A = \widetilde{P}^T \widetilde{L}^{-1} U$, and thus the following important result about the LU decomposition with partial pivoting.

Theorem 3.18. *Each nonsingular matrix $A \in \mathbb{C}^{n\times n}$ can be factorized $A = PLU$, where $P \in \mathbb{C}^{n\times n}$ is a permutation matrix, $L = [l_{ij}] \in \mathbb{C}^{n\times n}$ is unit lower triangular with $|l_{ij}| \le 1$, and $U \in \mathbb{C}^{n\times n}$ is nonsingular and upper triangular.*

Let us write $P^T A = [\widetilde{a}_{ij}] = LU$, where $L = [l_{ij}] \in \mathbb{C}^{n\times n}$ is unit lower triangular with $|l_{ij}| \le 1$, and U is upper triangular. Then from (3.13) we obtain

$$\begin{aligned} \widetilde{a}_{1j} = l_{11}u_{1j} \quad &\Rightarrow \quad u_{1j} = \widetilde{a}_{1j}, \\ \widetilde{a}_{2j} = l_{21}u_{1j} + l_{22}u_{2j} \quad &\Rightarrow \quad |u_{2j}| \le 2\max\{|\widetilde{a}_{1j}|, |\widetilde{a}_{2j}|\}, \\ \widetilde{a}_{3j} = l_{31}u_{1j} + l_{32}u_{2j} + l_{33}u_{3j} \quad &\Rightarrow \quad |u_{3j}| \le 4\max\{|\widetilde{a}_{1j}|, |\widetilde{a}_{2j}|, |\widetilde{a}_{3j}|\}, \end{aligned}$$

and inductively we see that

$$|u_{ij}| \le 2^{i-1} \max_{1\le k\le i} |\widetilde{a}_{kj}|, \quad \text{for all } i, j = 1, \ldots, n. \tag{3.18}$$

Thus, when using the partial pivoting strategy, the entries of the upper triangular factor U are bounded in terms of the entries of A, where the upper bound contains the (inconvenient) constant 2^{i-1}. A closer analysis of this situation is based on the following definition.

Definition 3.19. *Let $A \in \mathbb{C}^{n\times n}$ be nonsingular, and let $A = PLU$ be an LU factorization of A with partial pivoting as in Theorem 3.18. Then the* growth factor *is defined as*[3]

$$\rho_n := \frac{\max_{i,j} |u_{ij}|}{\max_{i,j} |a_{ij}|}.$$

An alternative definition of the growth factor in Gaussian elimination with partial pivoting is given by maximizing over the values of all submatrices on the right hand side of (3.17) that occur in the elimination process, i.e., $\max_{i,j,k} |u_{ij}^{(k)}| / \max_{ij} |a_{ij}|$.

Note that in the above description of the partial pivoting strategy the pivot is "an entry $u_{k,j+1}^{(j)}$, $1 \le j+1 \le k \le n$, of maximum modulus among these entries". If there are several entries of the same maximum modulus, then some tiebreaking rule must be applied. For a given matrix A, different tiebreaking decisions may lead to different entries in U and hence to different values of the growth factor. For any of these values, however, we see from (3.18) that

$$\rho_n \le 2^{n-1}.$$

Wilkinson [46, p. 212] showed that for any $n \ge 2$ there exists a matrix $A \in \mathbb{R}^{n\times n}$ for which this upper bound is attained. For example,

$$A = \begin{bmatrix} 1 & & & 1 \\ -1 & 1 & & 1 \\ -1 & -1 & 1 & 1 \\ -1 & -1 & -1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & & & \\ -1 & 1 & & \\ -1 & -1 & 1 & \\ -1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 1 \\ & 1 & 0 & 2 \\ & & 1 & 4 \\ & & & 8 \end{bmatrix},$$

[3] The standard notation ρ for the growth factor is unfortunately in conflict with the standard notation for the spectral radius.

and we can easily see how this example generalizes to larger n.

Despite the existence of such extreme examples, Gaussian elimination with partial pivoting is "utterly stable in practice" in the sense that matrices A with large growth factors rarely occur. For discussions of this fact also from a historical point of view, see [42, pp. 166–170] or [16, Section 9.4].

Example 3.20. *MATLAB's LU decomposition algorithm with partial pivoting computes the factors P, L, U such that $PA = LU$, or actually $PA \approx LU$. Applying this algorithm to the Pascal matrix from Example 3.12 gives the following result:*

```
>> A=pascal(20); [L,U,P]=lu(A); norm(P*A-L*U)
ans =
     1.163781868564185e-05
>> norm(P*A-L*U)/norm(A)
ans =
     2.476403589575018e-16
```

The absolute backward error is large, but the relative backward error is close to the unit roundoff.
The LU decomposition algorithm is also MATLAB's way to compute the determinant of matrices, i.e., MATLAB computes the factors P, L, U with a unit lower triangular factor L, and then returns $det(A) = \pm det(U)$, where the sign is determined by $\det(P)$ *and* $\det(U)$ *is the product of the diagonal entries. However, the LU decomposition algorithm does not take the structure of A into account, which in this example has positive integer entries. It is clear that then* $\det(A)$ *must be an integer (in fact, here* $\det(A) = 1$*), but due to pivoting and rounding this property is lost:*

```
>> det(A)
ans =
     2.782628795196159e+01
>> det(P)
ans =
    -1
>> det(L)
ans =
     1
>> det(U)
ans =
    -2.782628795196159e+01
```

Using similar techniques as for the Cholesky factorization, one can now show that if the LU factorization algorithm with partial pivoting runs to completion, the computed factors

satisfy

$$A = \widehat{P}\widehat{L}\widehat{U} + \Delta A, \quad \text{where} \quad \frac{\|\Delta A\|}{\|A\|} \leq p(n)\rho_n u,$$

and $p(n)$ is some low-degree polynomial in n that depends on the chosen norm $\|\cdot\|$. When we subsequently apply the computed LU decomposition for solving the linear algebraic system $Ax = b$ using the normwise backward stable forward and back substitution algorithms, we obtain a computed approximation $\widehat{x}$ with

$$(A + \Delta A)\widehat{x} = b, \quad \text{where} \quad \frac{\|\Delta A\|}{\|A\|} \leq p(n)\rho_n u;$$

see, e.g., [16, p. 165]. Analogously to (3.13) we now obtain

$$\frac{\|r\|}{\|A\|\|\widehat{x}\| + \|b\|} \leq p(n)\rho_n u \frac{\|A\|\|\widehat{x}\|}{\|A\|\|\widehat{x}\| + \|b\|} < p(n)\rho_n u.$$

Apart from the growth factor ρ_n, the backward error bounds for the LU factorization algorithm with partial pivoting coincide with those for Hermitian positive definite matrices and the Cholesky decomposition.

Chapter 4

Iterative Methods for Solving Linear Algebraic Systems

As indicated in the introduction to Chapter 3, an essential difference between iterative and direct solution methods for solving linear algebraic systems is that the former generate a sequence of intermediate approximations (called *iterates*), while the latter yield an approximation of the exact solution only at the very end of the computation. Using the iterates it is possible to estimate the error (or residual) norm, which allows to stop the iteration when a desired accuracy is reached. This can be a significant advantage in practical applications, where we usually do not require a highly accurate approximation of the exact solution.

4.1 Classical iterative methods

In the following we consider a linear algebraic system $Ax = b$ with $A \in \mathbb{C}^{n\times n}$ nonsingular. Many "classical" iterative methods are based on a splitting $A = M - N$, where the matrix M should be nonsingular. For a computationally efficient method we also require that M^{-1} can be easily computed, or linear algebraic systems with M can be (approximately) solved at low cost. This is satisfied, for example, when M is a diagonal or a triangular matrix.
If $A = M - N$, where M is nonsingular, then $Ax = b$ can be written as $(M - N)x = b$, and hence

$$x = M^{-1}Nx + M^{-1}b.$$

This fixed point equation suggests the iterative method

$$x_{k+1} = M^{-1}Nx_k + M^{-1}b, \quad k = 0, 1, 2, \ldots, \tag{4.1}$$

where x_0 is a given initial approximation.
The kth *error* of the iteration is given by $e_k := x - x_k$. Using

$$M^{-1}b = M^{-1}(M - N)x = x - M^{-1}Nx,$$

we obtain

$$e_k = x - x_k = x - (M^{-1}Nx_{k-1} + x - M^{-1}Nx) = M^{-1}Ne_{k-1},$$

and by induction

$$e_k = (M^{-1}N)^k e_0, \quad k = 0, 1, 2, \ldots. \tag{4.2}$$

The matrix $M^{-1}N$ is called the *iteration matrix* of (4.1). The iteration converges to the exact solution x, when $e_k \to 0$ for $k \to \infty$. Here and in the following, the convergence to zero is meant entrywise in the vector or matrix sequences we consider.
For consistent norms $\|\cdot\|$ we have the error bound

$$\|e_k\| = \|(M^{-1}N)^k e_0\| \leq \|M^{-1}N\|^k \|e_0\|.$$

Thus, if there exists some consistent norm $\|\cdot\|$ with $\|M^{-1}N\| < 1$, then the iteration (4.1) converges *for any* x_0.
More generally, from (4.2) we see that the iteration converges for any given x_0, if and only if $(M^{-1}N)^k \to 0$ for $k \to \infty$. In order to analyze this situation we consider the Jordan decomposition

$$M^{-1}N = XJX^{-1}, \quad J = \operatorname{diag}(J_{d_1}(\lambda_1), \ldots J_{d_m}(\lambda_m)).$$

Then $(M^{-1}N)^k = XJ^kX^{-1}$, and $(M^{-1}N)^k \to 0$ holds if and only if

$$J^k = \operatorname{diag}((J_{d_1}(\lambda_1))^k, \ldots (J_{d_m}(\lambda_m))^k) \to 0.$$

The kth power of a Jordan block is given by

$$(J_d(\lambda))^k = (\lambda I_d + J_d(0))^k = \sum_{j=0}^{\min\{k,d-1\}} \binom{k}{j} \lambda^{k-j} (J_d(0))^j$$

$$= \sum_{j=0}^{\min\{k,d-1\}} \frac{1}{j!} \frac{k!\lambda^{k-j}}{(k-j)!} (J_d(0))^j,$$

and thus $(J_d(\lambda))^k \to 0$ holds if and only if

$$\left| \frac{k!\lambda^{k-j}}{(k-j)!} \right| = \frac{k!|\lambda|^{k-j}}{(k-j)!} \to 0, \quad k \to \infty$$

for every $j = 0, 1 \ldots, d-1$. It is clear that this holds when $\lambda = 0$, therefore we can assume that $\lambda \neq 0$.
For $j = 0$ the limit above reduces to $|\lambda|^k \to 0$, hence $|\lambda| < 1$ is necessary for $(J_d(\lambda))^k \to 0$. For a fixed $j \in \{0, 1, \ldots, d-1\}$ we define the sequence $a_k := k!|\lambda|^{k-j}/(k-j)!$ for $k \geq d$. Dividing two consecutive terms of this sequence yields

$$\frac{a_{k+1}}{a_k} = \frac{(k+1)!|\lambda|^{k+1-j}}{(k+1-j)!} \frac{(k-j)!}{k!|\lambda|^{k-j}} = |\lambda| \frac{k+1}{k+1-j} \leq |\lambda| \frac{k+1}{k+1-d} =: |\lambda| c_k,$$

where $c_k \to 1$ for $k \to \infty$. Thus, $a_{k+2} \leq |\lambda|^2 a_k c_{k+1} c_k$, and inductively we see that $|\lambda| < 1$ is also sufficient for $(J_d(\lambda))^k \to 0$. In summary, we have shown the following result.

Theorem 4.1. *The iteration* (4.1) *converges for any initial vector* x_0*, if and only if the spectral radius of the iteration matrix satisfies* $\rho(M^{-1}N) < 1$*. A sufficient condition for convergence for any initial vector* x_0 *is* $\|M^{-1}N\| < 1$ *for some consistent norm* $\|\cdot\|$*.*

For a simple example, suppose that

$$M^{-1}N = \begin{bmatrix} \frac{1}{2} & \alpha \\ 0 & \frac{1}{2} \end{bmatrix}.$$

Then $\rho(M^{-1}N) = \frac{1}{2}$ and $(M^{-1}N)^k \to 0$ for $k \to \infty$. However, for any common matrix norm we will have $\|M^{-1}N\| \gg 1$ if $|\alpha| \gg 1$. The iteration (4.1) will therefore converge for any given x_0, although the norm of the iteration matrix is larger than 1. Recall also Lemma 2.12, where we have shown that $\|A\| \geq \rho(A)$ holds for any consistent norm.

In order to construct specific examples of the iteration (4.1), we write A as

$$A = L + D + U,$$

where L and U are the *strictly* lower and upper triangular parts. This yields the following classical methods:

- Jacobi method: $M = D$, $N = -(L + U)$, hence $M^{-1}N = -D^{-1}(L + U) =: R_J$.
- Gauss–Seidel method: $M = L + D$, $N = -U$, hence $M^{-1}N = -(L + D)^{-1}U =: R_G$.

In both cases M^{-1} exists if and only if $A = [a_{ij}]$ has nonzero diagonal entries.
For the ∞-norm and the Jacobi method we then have

$$\|R_J\|_\infty = \max_{1 \leq i \leq n} \sum_{j \neq i} \frac{|a_{ij}|}{|a_{ii}|}.$$

A matrix that satisfies

$$\max_{1 \leq i \leq n} \sum_{j \neq i} \frac{|a_{ij}|}{|a_{ii}|} < 1, \quad \text{or equivalently} \quad |a_{ii}| > \sum_{j \neq i} |a_{ij}| \ \text{ for all } i = 1, \ldots, n,$$

is called *strictly (row) diagonally dominant*. Thus, the Jacobi method converges for any x_0 when applied to such matrices A.
For the Gauss-Seidel method and a strictly (row) diagonally dominant matrix A we consider the equation $R_G x = \lambda x$, or

$$-Ux = \lambda(L + D)x \quad \Leftrightarrow \quad \lambda Dx = -(\lambda L + U)x.$$

Assuming, without loss of generality, that the entry of largest magnitude in the vector x is $x_\ell = 1$, we obtain

$$|\lambda||a_{\ell\ell}| \leq |\lambda| \sum_{j=1}^{\ell-1} |a_{\ell j}| + \sum_{j=\ell+1}^{n} |a_{\ell j}|$$

and hence

$$|\lambda| \le \frac{\sum_{j=\ell+1}^{n} |a_{\ell j}|}{|a_{\ell\ell}| - \sum_{j=1}^{\ell-1} |a_{\ell j}|} = \frac{\sum_{j=\ell+1}^{n} |a_{\ell j}|}{\underbrace{|a_{\ell\ell}| - \sum_{j\neq\ell} |a_{\ell j}|}_{>0} + \sum_{j=\ell+1}^{n} |a_{\ell j}|} < 1,$$

so that $\rho(R_G) < 1$ (if A is strictly (row) diagonally dominant).

Although forming R_G is "more expensive" than forming R_J, we are not guaranteed that the Gauss–Seidel method performs better than the Jacobi method.

Example 4.2. *For the nonsingular matrix*

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

we have

$$R_J = \begin{bmatrix} 0 & -2 & 2 \\ -1 & 0 & -1 \\ -2 & -2 & 0 \end{bmatrix} \quad \text{and} \quad R_G = \begin{bmatrix} 0 & -2 & 2 \\ 0 & 2 & -3 \\ 0 & 0 & 2 \end{bmatrix}.$$

Since $R_J^3 = 0$ we have $\rho(R_J) = 0$ and for any initial vector x_0 the third error of the Jacobi method will satisfy $e_3 = R_J^3 e_0 = 0$. On the other hand, $\rho(R_G) = 2$, and the Gauss-Seidel method will not converge when $e_0 \neq [\alpha, 0, 0]^T$.

If $\rho(M^{-1}N)$ is close to 1, then the convergence of (4.1) can be very slow. In order to improve the speed of convergence we can introduce a (real) *relaxation parameter* $\omega > 0$ and consider $\omega Ax = \omega b$ instead of $Ax = b$. We then split

$$\omega A = \omega(L + D + U) = (D + \omega L) + (\omega U + (\omega - 1)D) =: M - N,$$

which results in the iteration

$$x_{k+1} = R_{SOR}(\omega)x_k + \omega M^{-1}b, \quad k = 0, 1, 2, \ldots, \tag{4.3}$$

where

$$R_{SOR}(\omega) := -(D + \omega L)^{-1}(\omega U + (\omega - 1)D). \tag{4.4}$$

In order to form $R_{SOR}(\omega)$, we still require that A has nonzero diagonal entries. For $\omega = 1$ this gives the Gauss-Seidel method, and for $\omega > 1$ this method is called the *Successive Over Relaxation (SOR) method*. For $0 < \omega < 1$ the resulting methods are called under-relaxation methods.

Numerous publications, in particular from the 1950s and 1960s, are concerned with choosing an "optimal" ω in the sense that $\rho(R_{SOR}(\omega))$ is minimal in different applications. The following result of Kahan [22] shows that one can restrict the search of an optimal (real and positive) ω to the interval $(0, 2)$.

Theorem 4.3. *The matrix $R_{SOR}(\omega)$ in (4.4) satisfies $\rho(R_{SOR}(\omega)) \geq |1-\omega|$, and hence the method (4.3)–(4.4) can converge for any initial vector $x_0 \in \mathbb{C}^n$ only if $\omega \in (0,2)$.*

Proof. Let $\lambda_1(\omega), \ldots, \lambda_n(\omega)$ be the eigenvalues of $R_{SOR}(\omega)$. Then the determinant multiplication theorem yields

$$\begin{aligned}\prod_{j=1}^{n} \lambda_j(\omega) &= \det(R_{SOR}(\omega)) = (-1)^n \det(D+\omega L)^{-1}(-1)^n \det((1-\omega)D - \omega U) \\ &= \det(D)^{-1}(1-\omega)^n \det(D) = (1-\omega)^n,\end{aligned}$$

where we have used that L and U are strictly lower and upper triangular, respectively. Now $\prod_{j=1}^{n} |\lambda_j(\omega)| = |1-\omega|^n$ implies $\rho(R_{SOR}(\omega)) = \max_{1\leq j\leq n} |\lambda_j(\omega)| \geq |1-\omega|$. □

The theorem above does *not* show that $\omega \in (0,2)$ is sufficient for the convergence of the method (4.3)–(4.4). This is sufficient, however, when A is HPD, as shown by the following theorem, which is a special case of a result of Ostrowski [29]. In particular, the theorem shows that the Gauss-Seidel method converges for HPD matrices.

Theorem 4.4. *If $A \in \mathbb{C}^{n\times n}$ is HPD, then $\rho(R_{SOR}(\omega)) < 1$ holds for each $\omega \in (0,2)$.*

More generally, we can consider a splitting $A = M - N$, a relaxation parameter $\omega > 0$, and write $\omega Ax = \omega b$ in the equivalent form

$$x = ((1-\omega)I + \omega M^{-1}N)x + \omega M^{-1}b.$$

This yields the iteration

$$x_{k+1} = R(\omega)x_k + \omega M^{-1}b, \quad k = 0, 1, 2, \ldots,$$

where

$$R(\omega) := (1-\omega)I + \omega M^{-1}N.$$

Now $e_k = R(\omega)^k e_0$, so that the convergence is determined by $\rho(R(\omega))$. The spectrum of the iteration matrix is given by

$$\Lambda(R(\omega)) = (1-\omega) + \omega\Lambda(M^{-1}N),$$

which can be used for determining an optimal ω when $\Lambda(M^{-1}N)$ is (approximately) known. In all such iterations the convergence is asymptotically (for large k) *linear*, with the average "reduction factor" per step given by $\|R(\omega)\|$.

Finally, we point out that linear algebraic system $x = M^{-1}Nx + M^{-1}b$, which suggested the iterative method (4.1) studied in this section, can be written in the fixed point form

$$(I - M^{-1}N)x = M^{-1}b. \tag{4.5}$$

If $M^{-1}N$ is "small" (in some sense), then the system matrix $I - M^{-1}N$ is close to the identity matrix I, and the system (4.5) may be easier to solve than the original system $Ax = b$. The system (4.5) arises from $Ax = b$ by left-multiplication with M^{-1}, and in this context M^{-1} is sometimes called a *preconditioner* for the original system. The preconditioned system (4.5) can then be solved using further iterative methods, for example those introduced in the following section.

4.2 Projection methods and Krylov subspace methods

In this section we consider iterative methods for solving linear algebraic systems $Ax = b$, where $A \in \mathbb{C}^{n,n}$ and $b \in \mathbb{C}^n$, which are based on projections onto subspaces. Initially we will not assume that A is nonsingular, but this assumption will appear in most of the results that we will show below.
A projection method starts with an initial approximation $x_0 \in \mathbb{C}^n$, and then constructs a sequence of approximations

$$x_k \in x_0 + \mathcal{S}_k := \{x_0 + s \,:\, s \in \mathcal{S}_k\}, \quad k = 1, 2, \ldots, \tag{4.6}$$

where $\mathcal{S}_k$ is a k-dimensional subspace of $\mathbb{C}^n$ called the *search space* (of course we have $k \leq n$). Since we have k degrees of freedom to construct x_k, we require k constraints in order to determine x_k. We impose these on the residual $r_k := b - Ax_k$ and require that

$$r_k \perp \mathcal{C}_k, \quad k = 1, 2, \ldots, \tag{4.7}$$

where $\mathcal{C}_k$ is a k-dimensional subspace of $\mathbb{C}^n$ called the *constraints space*. Here the orthogonality is meant with respect to the Euclidean inner product, i.e., $\langle r_k, c\rangle = c^H r_k = 0$ for all $c \in \mathcal{C}_k$. The condition (4.7) can therefore equivalently be written as

$$r_k \in \mathcal{C}_k^{\perp} := \{z \in \mathbb{C}^n \,:\, \langle z, c\rangle = 0 \text{ for all } c \in \mathcal{C}_k\}.$$

Note that $\dim(\mathcal{C}_k^{\perp}) = n - \dim(\mathcal{C}_k) = n - k$.

Suppose that $S_k, C_k \in \mathbb{C}^{n\times k}$ represent any bases of $\mathcal{S}_k$ and $\mathcal{C}_k$, respectively, then (4.6)–(4.7) can be written as

$$x_k = x_0 + S_k t_k \text{ for some } t_k \in \mathbb{C}^k, \tag{4.8}$$

and

$$0 = C_k^H r_k = C_k^H (b - Ax_0 - AS_k t_k), \quad \text{or} \quad C_k^H A S_k t_k = C_k^H r_0. \tag{4.9}$$

The matrix $C_k^H A S_k \in \mathbb{C}^{k,k}$ is called the *projected matrix*. In order to obtain t_k and hence x_k in step k, we thus need to solve a linear algebraic system of order k. This gives Algorithm 3. An example for a stopping criterion is $\|r_k\|_2 < \text{tol}$, where tol is some user-specified accuracy.

Algorithm 3 Prototype projection algorithm

Input: $A \in \mathbb{C}^{n\times n}$, $b, x_0 \in \mathbb{C}^n$, stopping criterion, maximal number of iterations $n_{\max}$
Output: Approximate solution x_k
Initialize: $r_0 = b - Ax_0$
for $k = 1, \ldots, n_{\max}$ **do**
  Determine full rank matrices $S_k, C_k \in \mathbb{C}^{n\times k}$
  Solve $C_k^H A S_k t_k = C_k^H r_0$ for t_k
  Set $x_k = x_0 + S_k t_k$ and stop if satisfied
end for

For a given subspace $\mathcal{U} \subseteq \mathbb{C}^n$ we define

$$A\mathcal{U} := \{Au \,:\, u \in \mathcal{U}\}.$$

Note that if A is nonsingular, then $\dim(A\mathcal{U}) = \dim(\mathcal{U})$.

The sum of two subspaces $\mathcal{U}_1, \mathcal{U}_2 \subseteq \mathbb{C}^n$ is defined as

$$\mathcal{U}_1 + \mathcal{U}_2 := \{u_1 + u_2 \,:\, u_1 \in \mathcal{U}_1,\, u_2 \in \mathcal{U}_2\}.$$

It is easy to see that $\mathcal{U}_1 + \mathcal{U}_2$ is also a subspace of $\mathbb{C}^n$. The sum is called *direct* when $\mathcal{U}_1 \cap \mathcal{U}_2 = \{0\}$, and the direct sum is then denoted by $\mathcal{U}_1 \oplus \mathcal{U}_2$. A sum $\mathcal{V} = \mathcal{U}_1 + \mathcal{U}_2$ is direct if and only if for each $v \in \mathcal{V}$ there exist uniquely determined $u_1 \in \mathcal{U}_1$ and $u_2 \in \mathcal{U}_2$ with $v = u_1 + u_2$.
For example, if $\{v_1, \ldots, v_n\}$ is a basis of $\mathbb{C}^n$, then for any $k \in \{1, \ldots, n\}$ we have

$$\mathbb{C}^n = \operatorname{span}\{v_1, \ldots, v_k\} \oplus \operatorname{span}\{v_{k+1}, \ldots, v_n\},$$

i.e., $\mathbb{C}^n$ is decomposed into the direct sum of a k-dimensional and an $(n-k)$-dimensional subspace. (For $k = n$ we have the trivial decomposition $\mathbb{C}^n = \mathbb{C}^n \oplus \{0\}$.)

Lemma 4.5. *Let $A \in \mathbb{C}^{n\times n}$ and let $S_k, C_k \in \mathbb{C}^{n\times k}$ represent bases of the k-dimensional subspaces $\mathcal{S}_k, \mathcal{C}_k \subseteq \mathbb{C}^n$. Then the following statements are equivalent:*

(1) The matrix $C_k^H A S_k \in \mathbb{C}^{k\times k}$ is nonsingular.

(2) $\mathbb{C}^n = A\mathcal{S}_k \oplus \mathcal{C}_k^\perp$.

Proof. We first note that $\dim(A\mathcal{S}_k) \le k$ and $\dim(\mathcal{C}_k^\perp) = n - k$. Therefore condition (2) holds if and only if $A\mathcal{S}_k \cap \mathcal{C}_k^\perp = \{0\}$ (i.e., the sum is direct) and $\dim(A\mathcal{S}_k) = k$ (so that the sum of the two dimensions is n).
(1) $\Rightarrow$ (2): Suppose that $C_k^H A S_k$ is nonsingular. Then AS_k must have (full) rank k, so that $\dim(A\mathcal{S}_k) = k$. Now suppose that $z \in A\mathcal{S}_k \cap \mathcal{C}_k^\perp$. Then $z = AS_k y$ for some $y \in \mathbb{C}^k$, and since $z \in \mathcal{C}_k^\perp$ we have $0 = C_k^H z = C_k^H A S_k y$. Since $C_k^H A S_k$ is nonsingular we have $y = 0$, and therefore $z = 0$.
(2) $\Rightarrow$ (1): Suppose that $\mathbb{C}^n = A\mathcal{S}_k \oplus \mathcal{C}_k^\perp$, i.e., $A\mathcal{S}_k \cap \mathcal{C}_k^\perp = \{0\}$, and $\dim(A\mathcal{S}_k) = k$ so that AS_k has (full) rank k. If $C_k^H A S_k z = 0$ for some $z \in \mathbb{C}^k$, then $AS_k z \in A\mathcal{S}_k \cap \mathcal{C}_k^\perp$, and hence $AS_k z = 0$. Since AS_k has rank k, we have $z = 0$ and hence $C_k^H A S_k$ is nonsingular. □

This lemma shows that the question whether t_k in (4.8)–(4.9) is uniquely determined depends only on $A, \mathcal{S}_k, \mathcal{C}_k$ but not on the choice of bases for $\mathcal{S}_k, \mathcal{C}_k$.

Definition 4.6. *If t_k in (4.8)–(4.9) is uniquely determined, i.e., $C_k^H A S_k$ is nonsingular, we call the projection method (4.6)–(4.7) well defined at step k.*

Let the projection method be well defined at step k. Then

$$t_k = (C_k^H A S_k)^{-1} C_k^H r_0$$

hence

$$x_k = x_0 + S_k t_k = x_0 + S_k (C_k^H A S_k)^{-1} C_k^H r_0,$$

and

$$r_k = b - A x_k = (I_n - P_k) r_0, \text{ where } P_k = A S_k (C_k^H A S_k)^{-1} C_k^H. \tag{4.10}$$

The matrix P_k is a projection since $P_k^2 = P_k$. For all $v \in \mathbb{C}^n$ we have

$$P_k v \in A\mathcal{S}_k, \text{ and } (I_n - P_k) v \in \mathcal{C}_k^{\perp},$$

and hence P_k projects onto $A\mathcal{S}_k$ orthogonally to $\mathcal{C}_k$. Equation (4.10) can be written as

$$r_0 = \underbrace{P_k r_0}_{\in A\mathcal{S}_k} + \underbrace{r_k}_{\in \mathcal{C}_k^{\perp}}.$$

If $A\mathcal{S}_k = \mathcal{C}_k$, then this decomposition of the initial residual is orthogonal since its two components, $P_k r_0$ and r_k, are mutually orthogonal. We therefore call the method an *orthogonal projection method*. When $A\mathcal{S}_k \neq \mathcal{C}_k$, we call the method an *oblique projection method*.
The following basic result, a generalization of the classical Pythagorean Theorem about the sides of a right triangle, will be useful below.

Lemma 4.7. *Suppose that $\langle \cdot, \cdot \rangle_*$ is an inner product on $\mathbb{C}^n$, and that $\| \cdot \|_*$ is the induced norm, i.e., $\|z\|_* = \langle z, z \rangle_*^{1/2}$ for all $z \in \mathbb{C}^n$. If $z_1, z_2 \in \mathbb{C}^n$ are orthogonal with respect to the given inner product, i.e., $\langle z_1, z_2 \rangle_* = 0$, then $\|z_1 + z_2\|_*^2 = \|z_1\|_*^2 + \|z_2\|_*^2$.*

Proof. For all $z_1, z_2 \in \mathbb{C}^n$ we have

$$\|z_1 + z_2\|_*^2 = \|z_1\|_*^2 + \langle z_1, z_2 \rangle_* + \langle z_2, z_1 \rangle_* + \|z_2\|_*^2,$$

and $\langle z_1, z_2 \rangle_* = 0$ implies the result. □

In an orthogonal projection method we therefore have

$$\|r_0\|_2^2 = \| \underbrace{P_k r_0}_{\in A\mathcal{S}_k} + \underbrace{r_k}_{\in (A\mathcal{S}_k)^{\perp}} \|_2^2 = \|P_k r_0\|_2^2 + \|r_k\|_2^2, \quad \text{or}$$
$$\|r_k\|_2^2 = \|r_0\|_2^2 - \|P_k r_0\|_2^2.$$

If $A\mathcal{S}_k \subseteq A\mathcal{S}_{k+1}$, then $\|P_k r_0\|_2 \leq \|P_{k+1} r_0\|_2$ and hence $\|r_{k+1}\|_2 \leq \|r_k\|_2$, i.e., the 2-norm of the residual is monotonically decreasing.

Theorem 4.8. *In the notation established above, a projection method is well defined at step k, if one of the following conditions hold:*

(i) A is HPD and $\mathcal{C}_k = \mathcal{S}_k$,

(ii) A is nonsingular and $\mathcal{C}_k = A\mathcal{S}_k$.

Proof. (i) If $\mathcal{C}_k = \mathcal{S}_k$, then for any bases C_k, S_k we have $C_k = S_k Z$ for some nonsingular $Z \in \mathbb{C}^{k\times k}$, and $C_k^H A S_k = Z^H S_k^H A S_k$, which is nonsingular since $S_k^H A S_k$ is nonsingular (even HPD).
(ii) Now we have $C_k = A S_k Z$ and $C_k^H A S_k = Z^H S_k^H A^H A S_k$. This matrix is nonsingular, since A is nonsingular, and hence $S_k^H A^H A S_k$ is HPD. □

After studying when the projection method (4.6)–(4.7) is well defined we now study when the method terminates (in exact arithmetic) with $r_k = 0$.

Lemma 4.9. *In the notation established above, let the projection method be well defined at step k. If $r_0 \in \mathcal{S}_k$ and $A\mathcal{S}_k = \mathcal{S}_k$, then $r_k = 0$.*

Proof. If $r_0 \in \mathcal{S}_k$ and $A\mathcal{S}_k = \mathcal{S}_k$, then $r_0 \in A\mathcal{S}_k$, and hence $r_k = r_0 - P_k r_0 = 0$; cf.(4.10). □

This lemma gives a sufficient condition for the finite termination property of a projection process. It motivates to use search spaces $\mathcal{S}_k$ with

$$\mathcal{S}_1 = \operatorname{span}\{r_0\} \quad \text{and} \quad \mathcal{S}_1 \subset \mathcal{S}_2 \subset \mathcal{S}_3 \subset \dots , \tag{4.11}$$

such that these spaces automatically satisfy

$$A\mathcal{S}_k = \mathcal{S}_k \quad \text{for some } k. \tag{4.12}$$

In order to find such spaces, consider the *Krylov sequence* of A and r_0, which is given by

$$r_0, Ar_0, A^2 r_0, \dots .$$

If $r_0 \neq 0$, then there exists a uniquely determined smallest integer $d \geq 1$ such that $r_0, Ar_0, \dots, A^{d-1} r_0$ are linearly independent, and $r_0, Ar_0, \dots, A^d r_0$ are linearly dependent. This integer $d = d(A, r_0)$ is called the *grade of the vector* r_0 with respect to the matrix A.

It is easy to see that $d(A, r_0) = 1$ holds if and only if r_0 is an eigenvector of A. If

$$M_A(z) = z^m + \alpha_{m-1} z^{m-1} + \dots + \alpha_0$$

is the minimal polynomial of A, then $M_A(A) = 0$, and hence

$$A^m r_0 + \alpha_{m-1} A^{m-1} r_0 + \dots + \alpha_0 r_0 = 0,$$

which shows that $r_0, Ar_0, \dots, A^m r_0$ are linearly dependent. The vectors $r_0, Ar_0, \dots, A^{m-1} r_0$ must be linearly independent, since otherwise we could find a nonzero polynomial p with $p(A) = 0$ and $\deg(p) < m$, contradicting the minimality of the degree of M_A. Consequently, $1 \leq d(A, r_0) \leq m \leq n$ holds for *any* vector $r_0 \in \mathbb{C}^n$.

If $A = J_n(0)$ is the Jordan block of size $n \times n$ and with eigenvalue 0, and e_j is the jth standard basis vector of $\mathbb{C}^n$, then $Ae_j = e_{j-1}$ for every $j = 1, \ldots, n$, where we set $e_0 = 0$. Hence for each $j = 1, \ldots, n$ the vectors

$$e_j, Ae_j, \ldots, A^{j-1}e_j$$

are linearly independent, and

$$e_j, Ae_j, \ldots, A^{j-1}e_j, A^j e_j$$

are linearly dependent (since $A^j e_j = 0$), which shows that $d(A, e_j) = j$. Using the same idea and the Jordan canonical form of a matrix $A \in \mathbb{C}^{n\times n}$, one can show that for each $j = 1, \ldots, \deg(M_A)$ there exists a vector $v_j \in \mathbb{C}^n$ with $d(A, v_j) = j$.

Using the vectors from the Krylov sequence, we now define the kth *Krylov subspace* of A and r_0 by

$$\mathcal{K}_k(A, r_0) := \operatorname{span}\{r_0, Ar_0, \ldots, A^{k-1}r_0\}, \quad k \geq 1. \tag{4.13}$$

Projection methods that use $\mathcal{S}_k = \mathcal{K}_k(A, r_0)$ (or some variant, for example $\mathcal{S}_k = A\mathcal{K}_k(A, r_0)$) are called *Krylov subspace methods*. If we use $\mathcal{S}_k = \mathcal{K}_k(A, r_0)$, then the condition (4.11) is automatically satisfied by construction.
Moreover, if r_0 is of grade d with respect to A, then $r_0, \ldots, A^{d-1}r_0$ are linearly independent, and $r_0, \ldots, A^{d-1}r_0, A^d r_0$ are linearly dependent. Thus, $A^d r_0 \in \mathcal{K}_d(A, r_0)$ and it easily follows by induction that $A^{d+j}r_0 \in \mathcal{K}_d(A, r_0)$ for all $j \geq 1$, so that

$$\mathcal{K}_1(A, r_0) \subset \ldots \subset \mathcal{K}_d(A, r_0) = \mathcal{K}_{d+j}(A, r_0) \quad \text{for all } j \geq 1.$$

The following result shows further properties of the Krylov subspaces, and in particular that they also satisfy the condition (4.12).

Lemma 4.10. *Let $A \in \mathbb{C}^{n\times n}$ be nonsingular and let $r_0 \in \mathbb{C}^n \setminus \{0\}$ be of grade d with respect to A, then the following hold:*

(i) $A\mathcal{K}_d(A, r_0) = \mathcal{K}_d(A, r_0)$.

(ii) If $\mathcal{S}_k = \mathcal{K}_k(A, r_0)$ in the projection method (4.6)–(4.7), and the method is well defined at step d, then $r_d = 0$.

Proof. (i) By construction $A^d r_0 \in \mathcal{K}_d(A, r_0)$, and therefore

$$A\mathcal{K}_d(A, r_0) = \operatorname{span}\{Ar_0, \ldots, A^d r_0\} \subseteq \mathcal{K}_d(A, r_0).$$

Consider the equation

$$0 = \gamma_1 Ar_0 + \gamma_2 A^2 r_0 + \cdots + \gamma_d A^d r_0.$$

Since A is nonsingular, we can multiply it from the left with A^{-1}, which gives

$$0 = \gamma_1 r_0 + \gamma_2 Ar_0 + \cdots + \gamma_d A^{d-1} r_0.$$

The linear independence of $r_0, Ar_0, \ldots, A^{d-1}r_0$ implies that $\gamma_1 = \cdots = \gamma_d = 0$. Hence $Ar_0, A^2 r_0, \ldots, A^d r_0$ are linearly independent, so that indeed $A\mathcal{K}_d(A, r_0) = \mathcal{K}_d(A, r_0)$.
(ii) follows from (i) and Lemma 4.9. □

We can now use Theorem 4.8 and Lemma 4.10 to obtain the mathematical characterizations of several well defined Krylov subspace methods. In order to state the result, we recall that if $A \in \mathbb{C}^{n,n}$ is HPD, then

$$\langle x, y \rangle_A := y^H A x = \langle Ax, y \rangle = \langle x, Ay \rangle$$

defines an inner product on $\mathbb{C}^n$, which is called the *A-inner product*. The induced *A-norm* (or *energy norm*) on $\mathbb{C}^n$ is given by

$$\|x\|_A := \langle x, x \rangle_A^{1/2} := (x^H A x)^{1/2}.$$

Two vectors $x, y \in \mathbb{C}^n$ are *A-orthogonal* when $\langle x, y \rangle_A = 0$, and we then write $x \perp_A y$. Moreover, $x \perp_A \mathcal{S}$ for some subspace $\mathcal{S} \subseteq \mathbb{C}^n$ means that $\langle x, s \rangle_A = 0$ for all $s \in \mathcal{S}$, which we also write as $x \in \mathcal{S}^{\perp_A}$. (Note that for $A = I$ we have the Euclidean inner product, the Euclidean norm (or 2-norm), and the Euclidean orthogonality $x \perp y$.)

Theorem 4.11. *Consider the projection method* (4.6)–(4.7) *for solving a linear algebraic system $Ax = b$ with initial approximation x_0 and let $r_0 = b - Ax_0$ be of grade $d \geq 1$ with respect to A.*

(i) If A is HPD, $\mathcal{S}_k = \mathcal{C}_k = \mathcal{K}_k(A, r_0)$, $k = 1, 2, \ldots$, then the projection method is well defined at every step k until it terminates with $r_d = 0$ at step d. It is characterized by the orthogonality property

$$r_k \perp \mathcal{K}_k(A, r_0) \quad \text{or} \quad x - x_k \perp_A \mathcal{K}_k(A, r_0),$$

and the equivalent optimality property

$$\|x - x_k\|_A = \min_{z \in x_0 + \mathcal{K}_k(A, r_0)} \|x - z\|_A.$$

(Mathematical description of the Conjugate Gradient (CG) method.)

(ii) If A is nonsingular, $\mathcal{S}_k = \mathcal{K}_k(A, r_0)$, $\mathcal{C}_k = A\mathcal{S}_k = A\mathcal{K}_k(A, r_0)$, $k = 1, 2, \ldots$, then the projection method is well defined at every step k until it terminates with $r_d = 0$ at step d. It is characterized by the orthogonality property

$$r_k \perp A\mathcal{K}_k(A, r_0) \quad \text{or} \quad x - x_k \perp_{A^H A} \mathcal{K}_k(A, r_0),$$

and the equivalent optimality property

$$\|r_k\|_2 = \min_{z \in x_0 + \mathcal{K}_k(A, r_0)} \|b - Az\|_2$$

(Mathematical description of the MINRES and the GMRES method.)

Proof. It only remains to prove the equivalent optimality properties.
(i) By construction we have $r_k \perp \mathcal{K}_k(A, r_0)$, i.e., $\langle r_k, z\rangle = 0$ for all $z \in \mathcal{K}_k(A, r_0)$. But since $r_k = b - Ax_k = A(x - x_k)$, this is equivalent with

$$0 = \langle A(x - x_k), z\rangle = \langle x - x_k, z\rangle_A \quad \text{for all } z \in \mathcal{K}_k(A, r_0),$$

i.e., $x - x_k \perp_A \mathcal{K}(A, r_0)$.
Let $x_k \in x_0 + \mathcal{K}_k(A, r_0)$ be the uniquely determined vector that satisfies $x - x_k \perp_A \mathcal{K}_k(A, r_0)$. Using Lemma 4.7 we then obtain, for any $z \in x_0 + \mathcal{K}_k(A, r_0)$,

$$\|x - z\|_A^2 = \| \underbrace{(x - x_k)}_{\in \mathcal{K}_k(A,r_0)^{\perp_A}} + \underbrace{(x_k - z)}_{\in \mathcal{K}_k(A,r_0)} \|_A^2 = \|x - x_k\|_A^2 + \|x_k - z\|_A^2 \geq \|x - x_k\|_A^2,$$

and hence x_k minimizes $\|x - z\|_A$ over $x_0 + \mathcal{K}_k(A, r_0)$. On the other hand, the above inequality is strict unless $z = x_k$, which shows that orthogonality and optimality properties are equivalent.
(ii) By construction we have $r_k \perp A\mathcal{K}_k(A, r_0)$, i.e., $\langle r_k, z\rangle = 0$ for all $z \in A\mathcal{K}_k(A, r_0)$, which is equivalent with $\langle r_k, Az\rangle = 0$ for all $z \in \mathcal{K}_k(A, r_0)$. Since $r_k = A(x - x_k)$, this is equivalent with

$$0 = \langle A(x - x_k), Az\rangle = \langle x - x_k, A^H Az\rangle = \langle x - x_k, z\rangle_{A^H A} \quad \text{for all } z \in \mathcal{K}_k(A, r_0),$$

i.e., $x - x_k \perp_{A^H A} \mathcal{K}(A, r_0)$. Moreover, for any $z \in \mathbb{C}^n$ we have

$$\begin{aligned} \|b - Az\|_2^2 &= \langle A(x - z), A(x - z)\rangle = \langle x - z, A^H A(x - z)\rangle \\ &= \langle x - z, x - z\rangle_{A^H A} = \|x - z\|_{A^H A}^2. \end{aligned}$$

Let $x_k \in x_0 + \mathcal{K}_k(A, r_0)$ be the uniquely determined vector that satisfies $x - x_k \perp_{A^H A} \mathcal{K}_k(A, r_0)$. Using again Lemma 4.7 we obtain, for any $z \in x_0 + \mathcal{K}_k(A, r_0)$,

$$\begin{aligned} \|b - Az\|_2^2 = \|x - z\|_{A^H A}^2 &= \| \underbrace{(x - x_k)}_{\in \mathcal{K}_k(A,r_0)^{\perp_{A^H A}}} + \underbrace{(x_k - z)}_{\in \mathcal{K}_k(A,r_0)} \|_{A^H A}^2 \\ &= \|x - x_k\|_{A^H A}^2 + \|x_k - z\|_{A^H A}^2 \geq \|x - x_k\|_{A^H A}^2 = \|r_k\|_2^2, \end{aligned}$$

and hence x_k minimizes $\|b - Az\|_2$ over $x_0 + \mathcal{K}_k(A, r_0)$. Again the inequality is strict unless $z = x_k$, which shows that orthogonality and optimality properties are equivalent. □

Theorem 4.11 shows that for an HPD matrix A the CG method minimizes the A-norm of the error over $x_0 + \mathcal{K}_k(A, r_0)$, and that for a general nonsingular matrix A the GMRES method minimizes the 2-norm of the residual over $x_0 + \mathcal{K}_k(A, r_0)$ (motivating the name Generalized Minimal Residual Method). Since the dimensions of the subspaces $\mathcal{K}_k(A, r_0)$ are growing with k, and the methods minimize over (affine) subspaces of growing dimensions, the norms $\|x - x_k\|_A$ in the CG method and $\|r_k\|_2$ in the GMRES method are monotonically decreasing with k. We already knew this fact for the GMRES method because it is an orthogonal projection method. It can be shown that the A-norm of the error

is always *strictly* monotonically decreasing (until the method terminates). However, there are examples in which the 2-norm of the residual in the GMRES can stagnate. Even complete stagnation, i.e., $\|r_0\|_2 = \|r_1\|_2 = \cdots = \|r_{n-1}\|_2$, is possible in the GMRES method for certain $(n \times n)$-matrices and special right hand sides.

4.3 The Arnoldi and Lanczos algorithms

In order to implement the Krylov subspace methods that are characterized in Theorem 4.11, we need bases of $\mathcal{S}_k = \mathcal{K}_k(A, r_0)$ and $\mathcal{C}_k$. The "canonical" basis $r_0, Ar_0, \ldots, A^{k-1}r_0$ of $\mathcal{K}_k(A, r_0)$ should not be used in practical computations, since the corresponding matrix usually is (very) ill-conditioned: For simplicity, assume that A is diagonalizable, $A = X \operatorname{diag}(\lambda_1, \ldots, \lambda_n) X^{-1}$, with a single dominant eigenvalue, $|\lambda_1| > |\lambda_j|$ for $j = 2, \ldots, n$, and suppose that $r_0 = X[\alpha_1, \ldots, \alpha_n]^T$ with $\alpha_1 \neq 0$. Then

$$A^k r_0 = \lambda_1^k X \begin{bmatrix} 1 & & & \\ & (\lambda_2/\lambda_1)^k & & \\ & & \ddots & \\ & & & (\lambda_n/\lambda_1)^k \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{bmatrix}.$$

It is easy to see that the sequence

$$v_k = A^{k-1}r_0 / \|A^{k-1}r_0\|, \quad k = 1, 2, \ldots$$

converges towards the (normalized) first column of X, i.e., an eigenvector of A corresponding to the dominant eigenvalue. Exploiting this behavior is actually the main idea of the power method for computing eigenvalues and eigenvectors, which we will analyze in Section 7.1.

For reasons of numerical stability it is advisable to use orthonormal bases of the Krylov subspaces in practical implementations. The most straightforward approach is to apply the Gram–Schmidt algorithm to the matrix $[r_0, Ar_0, \ldots, A^{d-1}r_0]$, which by assumption has full rank d; see Algorithm 4. As observed by Arnoldi [1], it is mathematically equivalent and in many applications advantageous to replace $A^j r_0$ in Algorithm 4 by Av_j, which gives Algorithm 5. Note that the Arnoldi algorithm does not explicitly require the matrix A. Only a function that implements the map $v \mapsto Av$ needs to be known. This can be a significant advantage in practical applications, particularly when A is sparse (i.e., has many zero entries) or has some other structure that allows to efficiently compute the matrix-vector product Av.
If the initial vector r_0 is of grade d with respect to A, then for each $k = 1, \ldots, d$ the vectors $v_1, \ldots, v_k$ generated by Algorithm 5 form an orthonormal basis of $\mathcal{K}_k(A, r_0)$ (in exact arithmetic). The algorithm terminates in step $k = d$, since in this step we must have

$$Av_d \in \operatorname{span}\{v_1, \ldots, v_d\},$$

Algorithm 4 Gram–Schmidt algorithm for orthonormal Krylov subspace basis

Input: Matrix $A \in \mathbb{C}^{n \times n}$ and (nonzero) initial vector $r_0 \in \mathbb{C}^n$ of grade d
Output: Matrix $V \in \mathbb{C}^{n \times d}$ with orthonormal columns, and upper triangular matrix $R \in \mathbb{C}^{d \times d}$ with positive diagonal entries, such that $[r_0, Ar_0, \ldots, A^{d-1}r_0] = VR$.
Set $v_1 = r_0 / r_{11}$, where $r_{11} = \|r_0\|_2$
for $k = 1, 2, \ldots$ **do**
    $\widehat{v}_{k+1} = A^k r_0 - \sum_{i=1}^{k} r_{i,k+1} v_i$, where $r_{i,k+1} = \langle A^k r_0, v_i \rangle$
    $r_{k+1,k+1} = \|\widehat{v}_{k+1}\|_2$
    if $r_{k+1,k+1} = 0$ **then**
        $v_{k+1} = 0$ and stop
    else
        $v_{k+1} = \widehat{v}_{k+1} / r_{k+1,k+1}$
    end if
end for

Algorithm 5 Arnoldi algorithm (Classical Gram–Schmidt variant)

Input: Matrix $A \in \mathbb{C}^{n \times n}$ and (nonzero) initial vector $r_0 \in \mathbb{C}^n$ of grade d
Output: Matrix $V_d \in \mathbb{C}^{n \times d}$ with orthonormal columns, and upper Hessenberg matrix $H_d \in \mathbb{C}^{d \times d}$ with positive subdiagonal entries, such that $AV_d = V_d H_d$
Set $v_1 = r_0 / \|r_0\|_2$
for $k = 1, 2 \ldots$ **do**
    $\widehat{v}_{k+1} = Av_k - \sum_{i=1}^{k} h_{ik} v_i$, where $h_{ik} = \langle Av_k, v_i \rangle$
    $h_{k+1,k} = \|\widehat{v}_{k+1}\|_2$
    if $h_{k+1,k} = 0$ **then**
        $v_{k+1} = 0$ and stop
    else
        $v_{k+1} = \widehat{v}_{k+1} / h_{k+1,k}$
    end if
end for

which implies that $\widehat{v}_{d+1} = 0$. Note that the grade d does not need to be known a priori. For each $k = 1, \ldots, d-1$ we have the equation

$$Av_k = h_{k+1,k} v_{k+1} + \sum_{i=1}^{k} h_{ik} v_i,$$

and writing the first k of these equations in matrix form gives

$$AV_k = V_k H_k + h_{k+1,k} v_{k+1} e_k^T, \quad k = 1, \ldots, d-1, \tag{4.14}$$

where $V_k = [v_1, \ldots, v_k] \in \mathbb{C}^{n \times k}$ has orthonormal columns, and

$$H_k = [h_{ij}] \in \mathbb{C}^{k \times k} \quad \text{with} \quad h_{j+1,j} \neq 0 \ \text{ for } j = 1, \ldots, k \quad \text{and} \quad h_{ij} = 0 \ \text{ for } i > j+1.$$

The matrix H_k is called a $k \times k$ *unreduced upper Hessenberg matrix*. The equation (4.14) can also be written in the form

$$AV_k = V_{k+1} H_{k+1,k}, \quad \text{where } H_{k+1,k} \in \mathbb{C}^{(k+1) \times k}, \quad k = 1, \ldots, d-1. \tag{4.15}$$

Since $Av_d \in \operatorname{span}\{v_1, \ldots, v_d\}$ we have

$$Av_d = \sum_{i=1}^{d} h_{id} v_i,$$

and combining this equation with (4.15) for $k = d-1$ shows that Algorithm 5 after d steps yields the *Arnoldi decomposition*

$$AV_d = V_d H_d, \tag{4.16}$$

where $V_d = [v_1, \ldots, v_d] \in \mathbb{C}^{n \times d}$ has orthonormal columns, and $H_d = [h_{ij}] \in \mathbb{C}^{d \times d}$ is a $d \times d$ unreduced upper Hessenberg matrix.

Strictly speaking, (4.16) is only a *decomposition* of A when $d = n$, since only then the matrix V_n is square and invertible. Then we can write $A = V_n H_n V_n^{-1} = V_n H_n V_n^H$, where the second inequality holds since V_n by construction is also unitary. Since for every vector $r_0 \in \mathbb{C}^n$ we have $d = d(A, r_0) \leq \deg(M_A) \leq n$, we can have $d = n$ only when $\deg(M_A) = n$, or, equivalently, when M_A is equal to the characteristic polynomial of A. Matrices with this property are called *nonderogatory*.

Because of numerical instabilities, the classical Gram–Schmidt variant of the Arnoldi algorithm is rarely used in practical computations. The most common implementation is the modified Gram–Schmidt (MGS) variant, where the orthogonalization is performed recursively; see Algorithm 6. Algorithms 5 and 6 are mathematically equivalent in the sense that in exact arithmetic they generate the same vectors $v_1, \ldots, v_d$ and the same unreduced upper Hessenberg matrix H_d. For a numerical illustration of the behavior of the two variants of the Arnoldi algorithm we consider the following examples.

Algorithm 6 Arnoldi algorithm (Modified Gram–Schmidt variant)

Input: $A \in \mathbb{C}^{n\times n}$ and (nonzero) initial vector $r_0 \in \mathbb{C}^n$ of grade d
Output: Matrix $V_d \in \mathbb{C}^{n\times d}$ with orthonormal columns, and upper Hessenberg matrix $H_d \in \mathbb{C}^{d\times d}$ with positive subdiagonal entries, such that $AV_d = V_d H_d$
Set $v_1 = r_0/\|r_0\|_2$
for $k = 1, 2, \ldots$ **do**
  $\widetilde{v}_{k+1,0} = Av_k$
  for $i = 1, \ldots, k$ **do**
    $\widetilde{v}_{k+1,i} = \widetilde{v}_{k+1,i-1} - h_{ik}v_i$, where $h_{ik} = \langle \widetilde{v}_{k+1,i-1}, v_i \rangle$
  end for
  $\widehat{v}_{k+1} = \widetilde{v}_{k+1,k}$
  $h_{k+1,k} = \|\widehat{v}_{k+1}\|_2$
  if $h_{k+1,k} = 0$ **then**
    $v_{k+1} = 0$ and stop
  else
    $v_{k+1} = \widehat{v}_{k+1}/h_{k+1,k}$
  end if
end for

Example 4.12. *We first use a real* 250×250 *random matrix generated in MATLAB by* `A=randn(250)*randn(250)`. *This matrix has a condition number of approximately* 2.3×10^4. *We run the two implementations with an initial vector* `v=randn(250)`. *In Figure 4.1 (left) we plot the loss of orthogonality of the computed basis vectors as a function of the iteration step* k, *i.e.,*

$$\|I_k - V_k^T V_k\|_2 \quad \text{for } k = 1, 2, \ldots.$$

The solid line shows these values for the classical Gram–Schmidt variant, and the dashed line for the modified Gram–Schmidt variant. Obviously, the loss of orthogonality is larger for classical Gram-Schmidt. Note however, that even for classical Gram-Schmidt we have $\|I_{200} - V_{200}^T V_{200}\|_2 < 10^{-13}$, *and hence the loss of orthogonality is not dramatic.*
Our next example is the Grcar matrix *generated by* `gallery('grcar',n,j)` *in MATLAB. This is an* $n \times n$ *upper Hessenberg matrix with values* -1 *on the subdiagonal and* $+1$ *on the diagonal and the first* j *superdiagonals. We use* `n=250` *and* `j=3`, *and obtain a matrix with* $\kappa(A) \approx 3.6211$, *i.e., the matrix is well conditioned. Nevertheless, both classical and modified Gram-Schmidt Arnoldi with the initial vector* `v=ones(250,1)` *suffer from a significant loss of orthogonality, as shown in Figure 4.1 (right). Around step* $k = 200$ *we even observe a complete loss of orthogonality, i.e.,* $\|I_k - V_k^T V_k\|_2 \approx 1$ *in both variants, which means that* V_k *will have a loss of rank.*

We now consider an important special case of the Arnoldi decomposition. Let $A \in \mathbb{C}^{n\times n}$

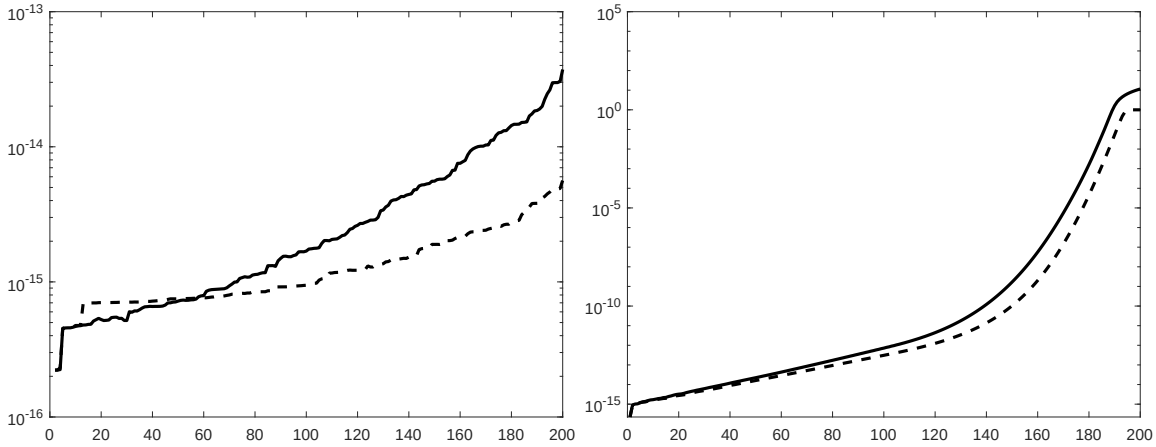


Figure 4.1: Loss of orthogonality of classical (solid) and modified (dashed) Gram-Schmidt Arnoldi for `A=randn(250)*randn(250)` (left) and `A=gallery('grcar',250,3)` (right).

be Hermitian, let $r_0 \in \mathbb{C}^n \setminus \{0\}$ be of grade d with respect to A, and let $AV_d = V_d H_d$ as in (4.16). Then, since V_d has orthonormal columns, $H_d = V_d^H A V_d$. Taking the Hermitian transpose and using $A^H = A$ gives

$$H_d^H = (V_d^H A V_d)^H = V_d^H A^H V_d = V_d^H A V_d = H_d.$$

The unreduced upper Hessenberg matrix $H_d = [h_{ij}]$ satisfies $h_{ij} = 0$ for $i > j+1$. Since H_d now also is Hermitian, this matrix must be tridiagonal. The decomposition $AV_d = V_d H_d$ for a Hermitian matrix A and with a tridiagonal matrix H_d is known as the *Lanczos decomposition*.
A comparison of the kth columns of $AV_d = V_d H_d$ with a tridiagonal matrix H_d gives

$$Av_k = h_{k+1,k} v_{k+1} + h_{kk} v_k + h_{k-1,k} v_{k-1}, \quad k = 1, \ldots, d,$$

where we set $h_{0,1} = h_{d+1,d} = 0$ and $v_0 = v_{d+1} = 0$. Thus,

$$h_{k+1,k} v_{k+1} = Av_k - h_{kk} v_k - h_{k-1,k} v_{k-1},$$

which means that the vector v_{k+1} satisfies a *3-term recurrence*. The resulting orthogonalization algorithm in the MGS variant is called the *Lanczos algorithm*; see Algorithm 7.
The decomposition computed by the Lanczos algorithm has the form

$$AV_d = V_d T_d, \quad \text{where} \quad T_d = \begin{bmatrix} \gamma_1 & \beta_1 & & \\ \beta_1 & \ddots & \ddots & \\ & \ddots & \ddots & \beta_{d-1} \\ & & \beta_{d-1} & \gamma_d \end{bmatrix}. \tag{4.17}$$

From now on we will use the notation T_d (instead of H_d) for the matrix in the Lanczos algorithm in order to stress that this matrix is tridiagonal. We point out that while for

Algorithm 7 Lanczos algorithm

Input: Hermitian matrix $A \in \mathbb{C}^{n\times n}$ and (nonzero) initial vector $r_0 \in \mathbb{C}^n$ of grade d
Output: Matrix $V_d \in \mathbb{C}^{n\times d}$ with orthonormal columns, and tridiagonal matrix $T_d \in \mathbb{C}^{d\times d}$ with positive sub- and superdiagonal entries
Set $v_0 = 0$, $\beta_0 = 0$, $v_1 = r_0/\|r_0\|_2$
for $k = 1, 2, \ldots$ **do**
  $u_k = Av_k - \beta_{k-1}v_{k-1}$
  $\widehat{v}_{k+1} = u_k - \gamma_k v_k$, where $\gamma_k = \langle u_k, v_k\rangle$
  $\beta_k = \|\widehat{v}_{k+1}\|_2$
  if $\beta_k = 0$ **then**
    $v_{k+1} = 0$ and stop
  else
    $v_{k+1} = \widehat{v}_{k+1}/\beta_k$
  end if
end for

a Hermitian matrix A we can compute orthonormal Krylov subspace bases by a 3-term recurrence, for a general matrix A this requires a full (Arnoldi) recurrence. The existence of short (3-term) recurrences for generating orthonormal Krylov subspace bases is essential for a low computational cost of methods that use such basis, and has been intensively analyzed since the early 1980s; see [23] for a survey of some results in this area.

4.4 Implementation and convergence analysis of CG

We now consider a linear algebraic system $Ax = b$, where $A \in \mathbb{C}^{n\times n}$ is HPD, and explain how to implement the CG method, which is mathematically characterized in item (i) of Theorem 4.11.
Using an initial guess x_0, we define $r_0 = b - Ax_0$ and apply the Lanczos algorithm to compute orthonormal bases of the Krylov subspaces $\mathcal{K}_k(A, r_0)$. In exact arithmetic this algorithm terminates at step $d = d(A, r_0)$ with a decomposition of the form $AV_d = V_dT_d$. The grade of r_0 is usually not known a priori, and we do not need to know it for implementing the method.
At any step $k = 1, \ldots, d-1$ of the Lanczos algorithm we have

$$AV_k = V_{k+1}T_{k+1,k}, \tag{4.18}$$

where $V_k = [v_1, \ldots, v_k] \in \mathbb{C}^{n\times k}$ has orthonormal columns, and $T_{k+1,k}$ consists of the first $k+1$ rows and k columns of T_d (similar to (4.15) for the Arnoldi algorithm). Note that since A is HPD, the tridiagonal matrix $T_k = V_k^H AV_k$ must be HPD as well.
Since T_k is Hermitian we have $\gamma_1, \ldots, \gamma_k \in \mathbb{R}$. Moreover, from the statement of the Lanczos algorithm we know that $\beta_j > 0$ for $j = 1, \ldots, k-1$. Thus, $T_k \in \mathbb{R}^{k\times k}$ is symmetric positive definite and tridiagonal with positive off-diagonal entries. Such a matrix is sometimes called a *Jacobi matrix*.

Using the mathematical characterization of the CG method we have

$$x_k = x_0 + V_k t_k$$

for some vector $t_k \in \mathbb{C}^k$ determined by the orthogonality property $r_k \perp \mathcal{K}_k(A, r_0)$, i.e.,

$$0 = V_k^H r_k = V_k^H (b - Ax_k) = V_k^H (r_0 - AV_k t_k) = \|r_0\|_2 e_1 - V_k^H A V_k t_k = \|r_0\|_2 e_1 - T_k t_k,$$

giving

$$t_k = T_k^{-1}(\|r_0\|_2 e_1) \quad \text{and} \quad x_k = x_0 + V_k T_k^{-1}(\|r_0\|_2 e_1).$$

Since T_k is symmetric positive definite, its Cholesky decomposition exists; cf. Theorem 1.3. We will now show how a clever use of this decomposition leads to simple update formulas with short recurrences for computing x_k.
Since T_k is tridiagonal, it can be easily shown that its Cholesky factorization has the form

$$T_k = L_k D_k L_k^T, \quad \text{where} \quad L_k \equiv \begin{bmatrix} 1 & & & \\ \mu_1 & 1 & & \\ & \ddots & \ddots & \\ & & \mu_{k-1} & 1 \end{bmatrix} \in \mathbb{R}^{k \times k},$$

with $\mu_j \neq 0$, $j = 1, \ldots, k-1$, and $D_k := \text{diag}(d_1, \ldots, d_k) \in \mathbb{R}^{k \times k}$ with $d_j > 0$, $j = 1, \ldots, k$. Consequently, we can write

$$x_k = x_0 + (V_k L_k^{-T})(\|r_0\|_2 D_k^{-1} L_k^{-1} e_1). \tag{4.19}$$

Let us define the matrix

$$\widehat{P}_k := [\widehat{p}_0, \ldots, \widehat{p}_{k-1}] := V_k L_k^{-T},$$

equivalently

$$V_k = \widehat{P}_k L_k^T = [\widehat{p}_0, \ldots, \widehat{p}_{k-1}] \begin{bmatrix} 1 & \mu_1 & & \\ & \ddots & \ddots & \\ & & 1 & \mu_{k-1} \\ & & & 1 \end{bmatrix}.$$

Hence the columns of the matrix $\widehat{P}_k$ are determined recursively as

$$\widehat{p}_j \;=\; v_{j+1} - \mu_j \widehat{p}_{j-1}, \quad j = 0, 1, \ldots, k-1, \tag{4.20}$$

where $\mu_0 := 0$ and $\widehat{p}_{-1} := 0$. Since $v_1 = r_0/\|r_0\|_2$, we see that

$$\text{span}\,\{\widehat{p}_0, \ldots, \widehat{p}_{k-1}\} = \text{span}\,\{v_1, \ldots, v_k\} = \mathcal{K}_k(A, r_0).$$

Moreover,

$$\widehat{P}_k^H A \widehat{P}_k = L_k^{-1} V_k^H A V_k L_k^{-T} = L_k^{-1} T_k L_k^{-T} = D_k, \tag{4.21}$$

which shows that the columns of $\widehat{P}_k$ form an A-orthogonal basis[1] of $\mathcal{K}_k(A, r_0)$.
Now we consider the vector

$$\widehat{c}_k = \begin{bmatrix} c_k^{(1)} \\ \vdots \\ c_k^{(k)} \end{bmatrix} := \|r_0\|_2 D_k^{-1} L_k^{-1} e_1$$

in (4.19), which is the unique solution of the linear algebraic system $L_k D_k \widehat{c}_k = \|r_0\|_2 e_1$. Since L_k is lower bidiagonal, this system can be written as

$$\left[\begin{array}{c|c} L_{k-1}D_{k-1} & 0 \\ \mu_{k-1}d_{k-1}e_{k-1}^T & d_k \end{array}\right] \left[\begin{array}{c} c_k^{(1)} \\ \vdots \\ c_k^{(k-1)} \\ \hline c_k^{(k)} \end{array}\right] = \|r_0\|_2 e_1.$$

The inverse of the matrix on the left hand side is given by

$$\begin{bmatrix} (L_{k-1}D_{k-1})^{-1} & 0 \\ -\frac{\mu_{k-1}d_{k-1}}{d_k} e_{k-1}^T (L_{k-1}D_{k-1})^{-1} & \frac{1}{d_k} \end{bmatrix}.$$

Since $\widehat{c}_{k-1} = (L_{k-1}D_{k-1})^{-1}(\|r_0\|_2 e_1)$, a straightforward computation yields

$$\widehat{c}_k = \begin{bmatrix} \widehat{c}_{k-1} \\ \hline c_k^{(k)} \end{bmatrix}, \quad \text{where} \quad c_k^{(k)} = -\frac{\mu_{k-1}d_{k-1}c_{k-1}^{(k-1)}}{d_k},$$

using this in (4.19) we obtain

$$\begin{aligned} x_k &= x_0 + \widehat{P}_k \widehat{c}_k = (x_0 + \widehat{P}_{k-1}\widehat{c}_{k-1}) + c_k^{(k)} \widehat{p}_{k-1} \\ &= x_{k-1} + c_k^{(k)} \widehat{p}_{k-1}. \end{aligned} \tag{4.22}$$

For the kth residual we then get

$$r_k = b - Ax_k = b - A(x_{k-1} + c_k^{(k)}\widehat{p}_{k-1}) = r_{k-1} - c_k^{(k)} A\widehat{p}_{k-1}. \tag{4.23}$$

The vectors $\widehat{p}_0, \ldots, \widehat{p}_{k-1}$ can be interpreted as *direction vectors* of the method, since at each step k we make a step from x_{k-1} into the direction of $\widehat{p}_{k-1}$.
Some further algebraic manipulations and simplifications (see, e.g., [24, Section 2.5.1] for details) yield an explicit form of the coefficient $c_k^{(k)}$ in (4.22) and (4.23). In these derivations we can use the A-orthogonality of the direction vectors and the orthogonality of the residual vectors, which follows from $r_k \perp \mathcal{K}_k(A, r_0)$. This yields the original implementation of the CG method due to Hestenes and Stiefel [15], which is stated in Algorithm 8.

[1] The A-orthogonal basis is sometimes called a *conjugate* basis of $\mathcal{K}_k(A, r_0)$, and this appears to be the original motivation for calling this the *Conjugate Gradient method*.

Algorithm 8 The CG method

Input: HPD matrix $A \in \mathbb{C}^{n\times n}$, vector $b \in \mathbb{C}^n$, initial vector $x_0 \in \mathbb{C}^n$, stopping criterion, maximal number of iterations $n_{\max}$
Output: Approximate solution x_k
Set $r_0 = b - Ax_0$, $p_0 = r_0$
for $k = 1, \ldots, n_{\max}$ **do**
  $\alpha_{k-1} = \frac{\|r_{k-1}\|_2^2}{p_{k-1}^H A p_{k-1}}$
  $x_k = x_{k-1} + \alpha_{k-1} p_{k-1}$
  $r_k = r_{k-1} - \alpha_{k-1} A p_{k-1}$
  Stop when the stopping criterion is satisfied
  $\omega_k = \frac{\|r_k\|_2^2}{\|r_{k-1}\|_2^2}$
  $p_k = r_k + \omega_k p_{k-1}$
end for

In Algorithm 8 we have not explicitly specified the stopping criterion. The CG method minimizes the A-norm of the error $\|x - x_k\|_A$ in every step, but this quantity is not computable, since it depends on the (unknown) solution vector x. Frequently the 2-norm of the residual, which is easily computable, is used for testing the convergence and for stopping the CG method. However, while the error norm $\|x - x_k\|_A$ is guaranteed to decrease monotonically with k, the residual 2-norms in the CG method do not necessarily decrease, and hence are often an unreliable measure of the convergence. Therefore, techniques to estimate the error norm $\|x - x_k\|_A$ have been developed, and it is recommended to use these for testing the convergence and stopping the CG method; see, e.g., [27] and the references given in that paper.

We will now study the convergence properties of the CG method. We know that the method is well defined for HPD matrices $A \in \mathbb{C}^{n\times n}$, and is mathematically characterized by $x_k \in x_0 + \mathcal{K}_k(A, r_0)$, and

$$\|x - x_k\|_A = \min_{z \in x_0 + \mathcal{K}_k(A, r_0)} \|x - z\|_A. \tag{4.24}$$

Since $r_0 = A(x - x_0)$, every $z \in x_0 + \mathcal{K}_k(A, r_0)$ can be written in the form

$$z = x_0 + \sum_{j=0}^{k-1} \gamma_j A^j r_0 = x_0 + \sum_{j=0}^{k-1} \gamma_j A^{j+1}(x - x_0)$$

for certain $\gamma_0, \gamma_1, \ldots, \gamma_{k-1} \in \mathbb{C}$ (or $\gamma_0, \gamma_1, \ldots, \gamma_{k-1} \in \mathbb{R}$ if A, b, x_0 are real). Therefore

$$x - z = x - x_0 + \sum_{j=0}^{k-1} \gamma_j A^j r_0 = \big(I - \sum_{j=0}^{k-1} \gamma_j A^{j+1}\big)(x - x_0) = p(A)(x - x_0),$$

where $p(z) = 1 - \sum_{j=0}^{k-1} \gamma_j z^{j+1}$ is a polynomial of degree k with $p(0) = 1$. If $\mathcal{P}_k(0)$ denotes the set of all such polynomials, then (4.24) can be written as

$$\|x - x_k\|_A = \min_{p \in \mathcal{P}_k(0)} \|p(A)(x - x_0)\|_A. \tag{4.25}$$

The HPD matrix A is unitarily diagonalizable with real positive eigenvalues, $A = X\Lambda X^H$ with $X^H X = I$, $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$, $\lambda_1 \geq \ldots \geq \lambda_n > 0$. We can thus define the square root

$$A^{1/2} := X\Lambda^{1/2}X^H, \quad \Lambda^{1/2} := \operatorname{diag}(\lambda_1^{1/2}, \ldots, \lambda_n^{1/2}),$$

which satisfies $(A^{1/2})^2 = A$. For every $v \in \mathbb{C}^n$ we thus get

$$\|v\|_A^2 = v^H A v = (v^H A^{1/2})(A^{1/2}v) = \|A^{1/2}v\|_2^2.$$

Using this result in (4.25) yields

$$\begin{aligned}
\|x - x_k\|_A &= \min_{p \in \mathcal{P}_k(0)} \|p(A)(x - x_0)\|_A \\
&= \min_{p \in \mathcal{P}_k(0)} \|A^{1/2}p(A)(x - x_0)\|_2 \\
&= \min_{p \in \mathcal{P}_k(0)} \|p(A)A^{1/2}(x - x_0)\|_2 \\
&\leq \min_{p \in \mathcal{P}_k(0)} \|p(A)\|_2 \|A^{1/2}(x - x_0)\|_2 \\
&= \min_{p \in \mathcal{P}_k(0)} \|p(\Lambda)\|_2 \|x - x_0\|_A \\
&= \min_{p \in \mathcal{P}_k(0)} \max_{1 \leq i \leq n} |p(\lambda_i)| \|x - x_0\|_A.
\end{aligned}$$

The only inequality in this derivation occurs when the matrix polynomial $p(A)$ and the vector $A^{1/2}(x - x_0)$ are separated. As shown in [12], this inequality is sharp in the sense that for every given HPD matrix $A \in \mathbb{C}^{n \times n}$ and iteration step k there exists an initial error $x - x_0$ for which equality holds.
Consequently, the quantity

$$\min_{p \in \mathcal{P}_k(0)} \max_{1 \leq i \leq n} |p(\lambda_i)|$$

describes the *worst-case* behavior of the (relative) error norms in the CG method. In this approximation problem we look for a polynomial $p(z)$ of degree at most k that has the minimal maximum value on the eigenvalues of A under the constraint $p(0) = 1$. Intuitively, it is clear that this value is small when the eigenvalues are contained in a "small" interval that is far away from zero. A computable bound is shown in the next result.

Theorem 4.13. *The relative A-norm of the error in step k of the CG method satisfies*

$$\begin{aligned}
\frac{\|x - x_k\|_A}{\|x - x_0\|_A} &\leq \min_{p \in \mathcal{P}_k(0)} \max_{1 \leq i \leq n} |p(\lambda_i)| \\
&\leq \min_{p \in \mathcal{P}_k(0)} \max_{z \in [\lambda_n, \lambda_1]} |p(z)| \\
&\leq 2\left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right)^k,
\end{aligned}$$

where $\kappa = \lambda_1/\lambda_n$ is the 2-norm condition number of A.

Proof. The first inequality was shown above. The second follows easily since the discrete set of eigenvalues in the min-max problem is replaced by the *inclusion set* $[\lambda_n, \lambda_1]$. The third inequality can be shown using suitably shifted and normalized *Chebyshev polynomials* as follows:
The Chebyshev polynomials of the first kind on the interval $[-1, 1]$ are given by

$$C_0(z) = 1, \quad C_1(z) = z, \quad \text{and} \quad C_{k+1}(z) = 2zC_k(z) - C_{k-1}(z), \quad k = 1, 2, \ldots . \tag{4.26}$$

For $z \in [-1, 1]$ we have $C_k(z) = \cos(k \cos^{-1}(z))$, which shows that $|C_k(z)| \leq 1$ for $z \in [-1, 1]$. The linear transformation

$$\rho(z) = \frac{2z - \lambda_1 - \lambda_n}{\lambda_n - \lambda_1}$$

maps the interval $[\lambda_n, \lambda_1]$ to the interval $[-1, 1]$, and thus $|C_k(\rho(z))| \leq 1$ for $z \in [\lambda_n, \lambda_1]$. Consequently,

$$\begin{aligned} \min_{p \in \mathcal{P}_k(0)} \max_{\lambda \in [\lambda_n, \lambda_1]} |p(\lambda)| &\leq \max_{z \in [\lambda_n, \lambda_1]} \left| \frac{C_k(\rho(z))}{C_k(\rho(0))} \right| \\ &\leq \frac{1}{|C_k(\rho(0))|}, \quad \text{where} \quad \rho(0) = \frac{-\lambda_1 - \lambda_n}{\lambda_n - \lambda_1} = \frac{\kappa + 1}{\kappa - 1}. \end{aligned}$$

A straightforward computation shows that $\rho(0)^2 - 1 = 4\kappa/(\kappa - 1)^2$. Since $\rho(0) > 1$ we can use the well known representation

$$C_k(z) = \frac{1}{2} \left((z + \sqrt{z^2 - 1})^k + (z - \sqrt{z^2 - 1})^k \right) \quad \text{for} \quad |z| \geq 1.$$

Since

$$\rho(0) + \sqrt{\rho(0)^2 - 1} = \frac{\sqrt{\kappa} + 1}{\sqrt{\kappa} - 1} \quad \text{and} \quad \rho(0) - \sqrt{\rho(0)^2 - 1} = \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1},$$

we obtain

$$C_k(\rho(0)) = \frac{1}{2} \left(\left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^k + \left(\frac{\sqrt{\kappa} + 1}{\sqrt{\kappa} - 1} \right)^k \right),$$

and finally

$$\frac{1}{|C_k(\rho(0))|} = 2 \left(\left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^k + \left(\frac{\sqrt{\kappa} + 1}{\sqrt{\kappa} - 1} \right)^k \right)^{-1} \leq 2 \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^k .$$

□

Main observations:

(1) The bounds in the theorem are *worst-case bounds* for the given matrix A, since they are independent of the choice of x_0 and the right hand side b.

(2) The last bound shows that the (worst-case) convergence of CG will be fast when the condition number of A is close to 1.

(3) The fact in (2) motivates the term *preconditioning*: Instead of $Ax = b$ we consider the (equivalent) preconditioned system

$$(L^{-1}AL^{-H})(L^H x) = L^{-1}b,$$

where L is an "easily invertible" matrix. In practical computations this means that solving linear algebraic systems with L should be "cheap". For example, L could be a lower triangular approximate Cholesky factor of A, i.e., $A \approx LL^H$. Then the matrix $L^{-1}AL^{-H}$ is HPD, and the worst-case convergence bound for the preconditioned system will involve $\kappa_2(L^{-1}AL^{-H})$ instead of $\kappa_2(A)$.

It is important to note that the worst-case bound does *not* imply that a smaller condition number will lead to a faster convergence of CG. Thus, even if we have $\kappa_2(L^{-1}AL^{-H}) < \kappa_2(A)$, the convergence of CG for the preconditioned system may be slower than for the original system. The general goal of preconditioning therefore is to obtain an equivalent system for which CG converges faster, rather than a preconditioned matrix which has a smaller condition number.

4.5 Implementation and convergence analysis of GMRES

We now consider a linear algebraic system $Ax = b$, where $A \in \mathbb{C}^{n\times n}$ is nonsingular, and explain how to implement the GMRES method, which is mathematically characterized in item (ii) of Theorem 4.11.
Using an initial guess x_0, we define $r_0 = b - Ax_0$ and apply the Arnoldi algorithm to compute orthonormal bases of the Krylov subspaces $\mathcal{K}_k(A, r_0)$. In exact arithmetic this algorithm terminates at step $d = d(A, r_0)$ with a decomposition of the form $AV_d = V_d H_d$. At any previous step $k = 1, \ldots, d-1$ we have

$$AV_k = V_{k+1}H_{k+1,k}, \tag{4.27}$$

where $V_k = [v_1, \ldots, v_k] \in \mathbb{C}^{n\times k}$ has orthonormal columns, and $H_{k+1,k}$ consists of the first $k+1$ rows and k columns of H_d; see (4.15).
Using (4.27) we have $x_k = x_0 + V_k t_k$ for some $t_k \in \mathbb{C}^k$, and hence

$$r_k = b - Ax_k = r_0 - AV_k t_k = r_0 - V_{k+1}H_{k+1,k}t_k = V_{k+1}(\|r_0\|_2 e_1 - H_{k+1,k}t_k).$$

The orthogonality condition $r_k \perp A\mathcal{K}_k(A, r_0)$ gives

$$0 = (AV_k)^H r_k = (V_{k+1}H_{k+1,k})^H r_k = H_{k+1,k}^H V_{k+1}^H V_{k+1}(\|r_0\|_2 e_1 - H_{k+1,k}t_k),$$

which is equivalent with

$$(H_{k+1,k}^H H_{k+1,k}) t_k = H_{k+1,k}^H (\|r_0\|_2 e_1).$$

By construction, for every $k = 1, \ldots, d-1$ the matrix $H_{k+1,k}$ has full rank k, so that $H_{k+1,k}^H H_{k+1,k} \in \mathbb{C}^{k\times k}$ is nonsingular. Therefore the uniquely determined solution of the linear algebraic system above is given by

$$t_k = H_{k+1,k}^+ (\|r_0\|_2 e_1), \quad \text{where} \quad H_{k+1,k}^+ := (H_{k+1,k}^H H_{k+1,k})^{-1} H_{k+1,k}^H. \tag{4.28}$$

The matrix $H_{k+1,k}^+ \in \mathbb{C}^{k\times(k+1)}$ is called the *Moore–Penrose pseudoinverse* of $H_{k+1,k}$. Note that $H_{k+1,k}^+ H_{k+1,k} = I_k$.
The equivalent optimality property in item (ii) of Theorem 4.11 is

$$\begin{aligned}\|r_k\|_2 &= \min_{z\in x_0+\mathcal{K}_k(A,r_0)} \|b - Az\|_2 = \min_{t\in\mathbb{C}^k} \|V_{k+1}(\|r_0\|_2 e_1 - H_{k+1,k} t)\|_2 \\ &= \min_{t\in\mathbb{C}^k} \|\,\|r_0\|_2 e_1 - H_{k+1,k} t\|_2.\end{aligned}$$

We will show in Chapter 5 that the unique minimizer of this *least squares problem* also is given by $t_k = H_{k+1,k}^+ (\|r_0\|_2 e_1)$.

The structure of the unreduced upper Hessenberg matrix $H_{k+1,k} \in \mathbb{C}^{(k+1)\times k}$ allows an efficient solution of this problem. Note that the QR decomposition of $H_{k+1,k}$ can be computed using k Givens rotations

$$G_j = \begin{bmatrix} I_{j-1} & & & \\ & c_j & s_j & \\ & -\overline{s}_j & c_j & \\ & & & I_{k-j} \end{bmatrix}, \quad c_j^2 + |s_j|^2 = 1, \quad j = 1, \ldots, k,$$

which yield

$$G_k \cdots G_1 H_{k+1,k} = \begin{bmatrix} R_k \\ 0 \end{bmatrix}, \quad \text{or} \quad H_{k+1,k} = Q_{k+1} \begin{bmatrix} R_k \\ 0 \end{bmatrix},$$

where $R_k \in \mathbb{C}^{k\times k}$ is upper triangular and nonsingular, and $Q_{k+1} = G_1^H \cdots G_k^H \in \mathbb{C}^{(k+1)\times(k+1)}$ is unitary and unreduced upper Hessenberg. In each step only one additional Givens rotation has to be computed. Using the QR decomposition we obtain

$$\begin{aligned}\|r_k\|_2 &= \min_{t\in\mathbb{C}^k} \|\,\|r_0\|_2 e_1 - H_{k+1,k} t\|_2 = \min_{t\in\mathbb{C}^k} \left\| Q_{k+1} \left(\|r_0\|_2 Q_{k+1}^H e_1 - \begin{bmatrix} R_k \\ 0 \end{bmatrix} t \right) \right\|_2 \\ &= \min_{t\in\mathbb{C}^k} \left\| \|r_0\|_2 Q_{k+1}^H e_1 - \begin{bmatrix} R_k \\ 0 \end{bmatrix} t \right\|_2,\end{aligned}$$

so that $t_k = \|r_0\|_2 R_k^{-1} (Q_{k+1}^H e_1)_{1:k}$, and $\|r_k\|_2 = \|r_0\|_2 |(Q_{k+1}^H e_1)_{k+1}|$. (This is a special case of the more general Theorem 5.3 that we will show below.)

The equation $\|r_k\|_2 = \|r_0\|_2 |(Q_{k+1}^H e_1)_{k+1}|$ shows that the residual norm $\|r_k\|_2$ is available without forming the approximation $x_k = x_0 + V_k t_k$ and the actual residual $r_k = b - Ax_k$. Due to rounding errors the quantity $\|r_0\|_2 |(Q_{k+1}^H e_1)_{k+1}|$, which is called the *updated residual norm*, may be (significantly) different from the *computed residual norm* $\|b - Ax_k\|_2$. In practice one should therefore not rely only on the size of the updated residual norm as a stopping criterion.

The GMRES method as implemented above (based on MGS Arnoldi and the QR decomposition of $H_{k+1,k}$) we first presented by Saad and Schultz in 1986 [33]. Because of the full recurrences in the Arnoldi algorithm, work and storage requirements in the GMRES method grow (linearly) with the iteration step k. Even for sparse matrices A, the Arnoldi basis vectors $v_1, v_2, \ldots$ usually will not be sparse. This may lead to storage problems in large applications and when many GMRES iteration steps need to be performed before a desired convergence tolerance is reached.

Algorithm 9 GMRES method

Input: Matrix $A \in \mathbb{C}^{n \times n}$, vector $b \in \mathbb{C}^n$, initial vector $x_0 \in \mathbb{C}^n$, convergence tolerance $\tau > 0$, maximal number of iterations $n_{\max}$
Output: Approximate solution x_k
Set $r_0 = b - Ax_0$
for $k = 1, \ldots, n_{\max}$ **do**
  Perform the kth step of MGS Arnoldi to generate V_k and $H_{k+1,k}$
  Update the QR decomposition of $H_{k+1,k}$
  Compute the (updated) residual norm $\|r_k\|_2 = \|r_0\|_2 (Q_{k+1}^H e_1)_{k+1}$
  if $\|r_k\|_2 < \tau$ **then**
    Compute the vector $t_k = H_{k+1,k}^+ (\|r_0\|_2 e_1)$
    Return the approximate solution $x_k = x_0 + V_k t_k$
  end if
end for

The *MINRES method* of Paige and Saunders [30] is an implementation of the projection method characterized in (ii) of Theorem 4.11 for Hermitian and nonsingular matrices, which is based on the Lanczos algorithm. It is mathematically equivalent to GMRES, but due to the Lanczos algorithm it uses 3-term instead of full recurrences. Hence work and storage requirements in the MINRES method remain constant throughout the iteration (similar to CG).

We now have a look at the convergence properties of GMRES. The method is mathematically characterized by $x_k \in x_0 + \mathcal{K}_k(A, r_0)$ and

$$\|r_k\|_2 = \|b - Ax_k\|_2 = \min_{z \in x_0 + \mathcal{K}_k(A, r_0)} \|b - Az\|_2.$$

For a diagonalizable matrix $A = X\Lambda X^{-1}$ we obtain

$$\begin{aligned}\|r_k\|_2 &= \min_{z\in\mathcal{K}_k(A,r_0)} \|r_0 - Az\|_2 = \min_{p\in\mathcal{P}_k(0)} \|p(A)r_0\|_2 \\ &\le \min_{p\in\mathcal{P}_k(0)} \|p(A)\|_2 \, \|r_0\|_2 \\ &= \min_{p\in\mathcal{P}_k(0)} \|Xp(\Lambda)X^{-1}\|_2 \, \|r_0\|_2 \\ &\le \kappa_2(X) \min_{p\in\mathcal{P}_k(0)} \|p(\Lambda)\|_2 \, \|r_0\|_2 \\ &= \kappa_2(X)\|r_0\|_2 \min_{p\in\mathcal{P}_k(0)} \max_{1\le i\le n} |p(\lambda_i)|,\end{aligned} \tag{4.29}$$

and thus

$$\frac{\|r_k\|_2}{\|r_0\|_2} \le \kappa_2(X) \min_{p\in\mathcal{P}_k(0)} \max_{1\le i\le n} |p(\lambda_i)|. \tag{4.30}$$

The right hand side of (4.30) is a worst-case bound on the relative residual norm in step k, similar to the worst-case bound for the CG method in Theorem 4.13. Since GMRES is well defined for any nonsingular matrix, we now have to take into account the condition number of the eigenvectors as well. Of course, for a normal matrix A we can choose eigenvectors so that $\kappa_2(X) = 1$.

When the eigenvectors of A are well-conditioned and the eigenvalues are in a single "cluster" that is far away from zero, then the bound (4.30) shows that GMRES will converge quickly. For example, suppose that the eigenvalues are contained in a disk with radius $\varrho > 0$ in the complex plane that is centered at $c \in \mathbb{C}$ and that does not contain zero, i.e., $\varrho < |c|$. Then $|c - \lambda_i| < \varrho$ for $i = 1, \ldots, n$, and using the polynomial $p(z) = (1 - z/c)^k \in \mathcal{P}_k(0)$ we obtain

$$\min_{p\in\mathcal{P}_k(0)} \max_{1\le i\le n} |p(\lambda_i)| \le \max_{1\le i\le n} \left|1 - \frac{\lambda_i}{c}\right|^k \le \left|\frac{\varrho}{c}\right|^k.$$

The right hand side decreases when the radius ϱ of the disk shrinks, and when the center c moves away from zero in the complex plane.

Note that if the eigenvector condition number $\kappa(X)$ is large, or the eigenvalues of A are not contained in a small disk, then the bound (4.30) does *not* imply that GMRES will converge slowly.

Example 4.14. *We apply the MATLAB implementation of GMRES with $x_0 = 0$ to linear algebraic systems $A_\alpha x = b$, where $A_\alpha \in \mathbb{R}^{n\times n}$ is generated by*

```
n = 500; ee = ones(n,1);
A = spdiags([ee,alpha*ee,ee,ee,ee,ee],-1:4,n,n);
```

and `b=randn(n,1)`. *The matrices A_α are well-conditioned when α is (a bit) larger than* 1, *but for moderate values of α they have very ill-conditioned eigenvectors:*

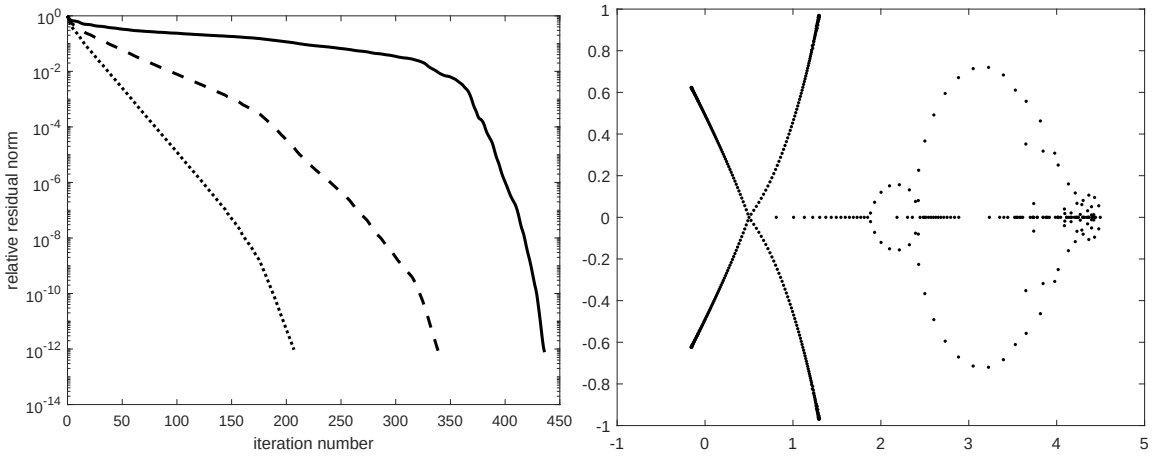


Figure 4.2: Left: Relative GMRES residual norms for $A_\alpha x = b$ with $\alpha = 1.5$ (solid), 1.75 (dashed), and 2.0 (dotted). Right: Eigenvalues of A_α for $\alpha = 1.5$ computed with MATLAB's `eig`.

α	$\kappa(A_\alpha)$	$\kappa(X_\alpha)$
1.5	58.5	8.96×10^{21}
1.75	27.9	6.19×10^{21}
2.0	16.9	2.87×10^{21}

The condition numbers were computed with MATLAB's `cond`*, and the eigendecompositions with* `eig`*.*
In Figure 4.2 (left) we plot the relative residual norms of GMRES for the three different values of α. For $\alpha = 1.5$ we observe a very slow convergence for almost 350 steps (note that the matrix is of size 500×500), while GMRES for $\alpha = 2.0$ converges quickly from the very first step. These differences can neither be explained by the condition number of A_α, nor by the condition number of the eigenvectors.
Figure 4.2 (right) shows the spectrum of A_α for $\alpha = 1.5$. Any other matrix $A_{\widehat{\alpha}}$ satisfies $A_{\widehat{\alpha}} = (\widehat{\alpha} - \alpha)I + A_\alpha$, and hence the spectrum of $A_{\widehat{\alpha}}$ is given by a linear shift of the spectrum of A_α. In particular, our choices of $\alpha = 1.75$ and $\alpha = 2.0$ correspond to a "shift to the right" compared with the spectrum Figure 4.2 (right). The increasing distance from the origin may be a reason why the the value of the polynomial approximation problem in the (4.30) *is potentially smaller with increasing α. However, because of the huge eigenvector condition numbers, the convergence bound* (4.30) *is practically useless.*

The minimization problem in (4.29) is called the *ideal GMRES approximation problem.* The name "ideal" is motivated by the fact that in this problem we have disentangled the "matrix essence of the process from the distracting effects of the initial vector" [14, p. 362].

Using the *field of values*

$$\mathcal{F}(A) := \{\langle Ax, x\rangle \,:\, \|x\|_2 = 1\}$$

we can derive a bound for the value of the ideal GMRES approximation problem [26, Theorem 3.1].

Theorem 4.15. *Let $A \in \mathbb{C}^{n\times n}$ be nonsingular and let $\nu(\mathcal{F}(A)) := \min_{z\in\mathcal{F}(A)} |z|$ be the distance of the field of values from the origin. Then for each $k \geq 0$ we have*

$$\min_{p\in\mathcal{P}_k(0)} \|p(A)\|_2 \leq \left(1 - \nu(\mathcal{F}(A))\nu(\mathcal{F}(A^{-1}))\right)^{k/2}. \tag{4.31}$$

Proof. Consider a given unit norm vector v with $Av \neq 0$ and the problem

$$\min_{\alpha\in\mathbb{C}} \|v - \alpha Av\|_2^2 = \min_{\alpha\in\mathbb{C}} \left(1 - 2\mathrm{Re}(\alpha\langle v, Av\rangle) + |\alpha|^2\langle Av, Av\rangle\right).$$

It is easy to show that the minimum is attained for

$$\alpha_* := \frac{\langle Av, v\rangle}{\langle Av, Av\rangle},$$

and that

$$\|v - \alpha_* Av\|_2^2 = 1 - \frac{\langle v, Av\rangle}{\langle Av, Av\rangle}\frac{\langle Av, v\rangle}{\langle v, v\rangle} = 1 - \frac{\langle A^{-1}w, w\rangle}{\langle w, w\rangle}\frac{\langle Av, v\rangle}{\langle v, v\rangle}, \quad \text{where } w := Av.$$

We now use α_* from above and an equality of min-max and max-min approximation problems that was shown by Joubert [21, Theorem 1] and Greenbaum and Gurvits [13, Theorem 2.5] to obtain

$$\begin{aligned}
\min_{p\in\mathcal{P}_k(0)} \|p(A)\|_2 &\leq \min_{\alpha\in\mathbb{C}} \|(I_n - \alpha A)^k\|_2 \leq \min_{\alpha\in\mathbb{C}} \|I_n - \alpha A\|_2^k = \min_{\alpha\in\mathbb{C}} \max_{\|v\|=1} \|v - \alpha Av\|_2^k \\
&= \max_{\|v\|=1} \min_{\alpha\in\mathbb{C}} \|v - \alpha Av\|_2^k = \max_{\|v\|=1} \left(\min_{\alpha\in\mathbb{C}} \|v - \alpha Av\|_2^2\right)^{k/2} \\
&= \max_{\|v\|=1} \left(1 - \frac{\langle v, Av\rangle}{\langle Av, Av\rangle}\frac{\langle Av, v\rangle}{\langle v, v\rangle}\right)^{k/2} \\
&\leq \left(1 - \min_{w\in\mathbb{C}^n} \left|\frac{\langle A^{-1}w, w\rangle}{\langle w, w\rangle}\right| \min_{v\in\mathbb{C}^n} \left|\frac{\langle Av, v\rangle}{\langle v, v\rangle}\right|\right)^{k/2} \\
&= \left(1 - \nu(\mathcal{F}(A))\nu(\mathcal{F}(A^{-1}))\right)^{k/2}.
\end{aligned}$$

□

In the derivation of (4.31) we have replaced the optimal polynomial of degree k by the polynomial $(1 - \alpha z)^k$, which has only one k-fold root $1/\alpha$. Thus, the bound (4.31) cannot be expected to be sharp in general. However, the bound (4.31) is one of the very few general purpose bounds on the value of the ideal GMRES approximation problem.

Chapter 5

Least Squares Problems and Low Rank Approximation

In the previous chapters we have studied the numerical solution of linear algebraic systems $Ax = b$, where usually A was a nonsingular matrix. Solving the linear algebraic system means to find a vector x, so that the right hand side b is *exactly equal* to a linear combination of the columns of A, with the coefficients in this linear combination given by the entries of x.
Our focus in this chapter is on *approximating* a given vector $b \in \mathbb{C}^n$ as closely as possible by a linear combination of the columns of a (non-square) matrix $A \in \mathbb{C}^{n\times m}$, where $m \leq n$ and usually $m < n$.

5.1 The full rank least squares problem

We are given a matrix $A \in \mathbb{C}^{n\times m}$ with $\text{rank}(A) = m$ (and hence $m \leq n$) and a vector $b \in \mathbb{C}^n$. We would like to approximate b as closely as possible by a linear combination of the columns of A. Of course, the problem on how to approximate b by the columns of A depends on the norm that we choose to measure the approximation. Here we consider the 2-norm. Thus, for the given A and b we want to solve the minimization problem

$$\arg\min_{x\in\mathbb{C}^m} \|b - Ax\|_2. \tag{5.1}$$

If $n = m$ and thus $A \in \mathbb{C}^{n\times n}$ is nonsingular, then $x = A^{-1}b$ is the uniquely determined solution. This case is included below, but we will mostly be interested in $m < n$.
Suppose that A and b are both real, i.e., $A \in \mathbb{R}^{n\times m}$ and $b \in \mathbb{R}^n$. Every vector $x \in \mathbb{C}^m$ can be written as $x = x_1 + ix_2$ with $x_1, x_2 \in \mathbb{R}^m$, and we have

$$\|b - Ax\|_2^2 = \|b - A(x_1 + ix_2)\|_2^2 = \|b - Ax_1\|_2^2 + \|x_2\|_2^2 \geq \|b - Ax_1\|_2^2,$$

with equality if and only if $x_2 = 0$, i.e., $x \in \mathbb{R}^m$. Thus, if A and b are both real, then the minimization problem is equivalent with

$$\arg\min_{x\in\mathbb{R}^m} \|b - Ax\|_2.$$

Example 5.1. *Suppose that we are given n data points $(\xi_j, \beta_j) \in \mathbb{R}^2$, $j = 1, \ldots, n$, where $\xi_1 < \xi_2 < \cdots < \xi_n$. For example, β_j could be the value of some measurement taken at time ξ_j for $j = 1, \ldots, n$. A polynomial $p \in \mathcal{P}_{m-1}$ (the (real) polynomials of degree at most $m-1$) approximates the given data points in the* least squares sense *when it minimizes the sum of the squares of the deviation from the data, i.e., when*

$$\sum_{j=1}^{n} |\beta_j - p(\xi_j)|^2 \leq \sum_{j=1}^{n} |\beta_j - q(\xi_j)|^2 \quad \text{for all } q \in \mathcal{P}_{m-1}.$$

Any $q \in \mathcal{P}_{m-1}$ has the form $q(z) = \alpha_0 + \alpha_1 z + \cdots + \alpha_{m-1} z^{m-1}$, and we need to determine the m coefficients $\alpha_0, \alpha_1, \ldots, \alpha_{m-1} \in \mathbb{R}$. If we define

$$b := \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix} \in \mathbb{R}^n, \quad A := \begin{bmatrix} 1 & \xi_1 & \cdots & \xi_1^{m-1} \\ 1 & \xi_2 & \cdots & \xi_2^{m-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \xi_n & \cdots & \xi_n^{m-1} \end{bmatrix} \in \mathbb{R}^{n \times m}, \quad x := \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{m-1} \end{bmatrix} \in \mathbb{R}^m,$$

then

$$\sum_{j=1}^{n} |\beta_j - q(\xi_j)|^2 = \|b - Ax\|_2^2.$$

Thus, finding a polynomial that approximates the data in a least squares sense is equivalent with solving the minimization problem (5.1). *The matrix A is called a* Vandermonde matrix. *It can be shown (by induction) that such a matrix has full rank if $m < n$ and $\xi_1, \ldots, \xi_n$ are pairwise distinct.*

Motivated by this example, the vector b and the matrix A in (5.1) are often called *observation vector* and *data matrix*, and the problem (5.1) is called a *least squares problem*. For any $x \in \mathbb{C}^m$, the vector $r(x) := b - Ax$ is the *residual* of the least squares problem.

Example 5.2. *Let us briefly discuss a generalization of Example 5.1 that is of interest in the area of Machine Learning; see [6, Chapter 3] for a significantly more detailed treatment. In this context we are given n vectors $\widetilde{x}_1, \ldots, \widetilde{x}_n \in \mathbb{R}^k$ that are interpreted as a* training set *consisting of n observations, and a vector $\widetilde{t} \in \mathbb{R}^n$ that contains n corresponding* target values. *The goal is to use this data for constructing a function that can "predict" the value $t \in \mathbb{R}$ for a new $x \in \mathbb{R}^k$.*
The function for making the prediction is constructed as a linear combination of m fixed (nonlinear) functions of the training set, i.e., we consider a function of the form

$$y(x, w) = \sum_{j=0}^{m-1} w_j \varphi_j(x) = w^T \varphi(x), \quad w = [w_0, w_1, \ldots, w_{m-1}]^T,$$

where $\varphi(x) = [\varphi_0(x), \varphi_1(x), \ldots, \varphi_{m-1}(x)]^T$ *contains the* basis functions $\varphi_j : \mathbb{R}^k \to \mathbb{R}$ *of the model, and we set* $\varphi_0(x) := 1$. *Our goal is to determine the so-called* bias parameter w_0 *and the parameters* $w_1, \ldots, w_{m-1}$ *in some appropriate way using the given training set and target values. A common type of basis functions is*

$$\varphi_j(x) = \exp\left(-\beta_j \|x - c_j\|_2^2\right), \quad j = 1, \ldots, m-1,$$

where the components of $c_j \in \mathbb{R}^k$ *and the values* β_j *determine the "location" and the spatial "scale" of the function, respectively. These functions* φ_j *are known as* radial basis functions.
Given the training values $\widetilde{x}_1, \ldots, \widetilde{x}_n \in \mathbb{R}^k$, *and the target* $\widetilde{t} = [t_1, \ldots, t_n]^T \in \mathbb{R}^n$, *we now determine the parameters* $w_0, w_1, \ldots, w_m$ *by minimizing the* error function

$$E(w) = \frac{1}{2}\sum_{j=1}^{n}(t_j - y(\widetilde{x}_j, w))^2 = \frac{1}{2}\sum_{j=1}^{n}(t_j - w^T\varphi(\widetilde{x}_j))^2 = \frac{1}{2}\|\widetilde{t} - \Phi w\|_2^2,$$

where

$$\Phi := \begin{bmatrix} \varphi_0(\widetilde{x}_1) & \varphi_1(\widetilde{x}_1) & \cdots & \varphi_{m-1}(\widetilde{x}_1) \\ \varphi_0(\widetilde{x}_2) & \varphi_1(\widetilde{x}_2) & \cdots & \varphi_{m-1}(\widetilde{x}_2) \\ \vdots & \vdots & & \vdots \\ \varphi_0(\widetilde{x}_n) & \varphi_1(\widetilde{x}_n) & \cdots & \varphi_{m-1}(\widetilde{x}_n) \end{bmatrix} \in \mathbb{R}^{n\times m}.$$

We see that minimizing $E(w)$ *for* $w \in \mathbb{R}^m$ *is equivalent to solving a least squares problem of the form* (5.1).

We now solve the least squares problem (5.1) using the QR decomposition of A.

Theorem 5.3. *Let* $A \in \mathbb{C}^{n\times m}$ *with full rank* m *and* $b \in \mathbb{C}^n$ *be given. Suppose that*

$$A = Q\begin{bmatrix} R \\ 0 \end{bmatrix} \quad \textit{with} \quad Q = [Q_1, Q_2]$$

is the QR decomposition of A *as in Theorem 1.4. Then* $\widehat{x} := R^{-1}Q_1^H b$ *is the uniquely determined solution of* (5.1), *and the residual is given by* $r(\widehat{x}) = (I - Q_1Q_1^H)b$.

Proof. Using the QR decomposition of A and the unitary invariance of the 2-norm we get

$$\begin{aligned} \|b - Ax\|_2^2 &= \left\| b - Q\begin{bmatrix} R \\ 0 \end{bmatrix} x \right\|_2^2 = \left\| Q\left(Q^H b - \begin{bmatrix} R \\ 0 \end{bmatrix} x\right)\right\|_2^2 = \left\| \begin{bmatrix} Q_1^H b - Rx \\ Q_2^H b \end{bmatrix}\right\|_2^2 \\ &= \|Q_1^H b - Rx\|_2^2 + \|Q_2^H b\|_2^2, \end{aligned}$$

which shows that $\|b - Ax\|_2 \geq \|Q_2^H b\|_2$ for any $x \in \mathbb{C}^m$. The lower bound is attained if and only if $\|Q_1^H b - Rx\|_2 = 0$, which is equivalent with $Rx = Q_1^H b$. Since R is nonsingular, the unique solution of (5.1) is given by $\widehat{x} = R^{-1}Q_1^H b$, and

$$r(\widehat{x}) = b - A\widehat{x} = b - [Q_1, Q_2]\begin{bmatrix} R \\ 0 \end{bmatrix} R^{-1}Q_1^H b = (I - Q_1Q_1^H)b$$

is the residual. □

Theorem 5.3 suggests Algorithm 10 for solving the least squares problem (5.1). Note that in practice, if a QR decompsition of A is available, then $\widehat{x}$ in step 2 of Algorithm 10 should be computed by solving the linear algebraic system $Rx = Q_1^H b$ with the upper triangular matrix R.

Algorithm 10 Least squares solution using the QR decomposition

Input: Matrix $A \in \mathbb{C}^{n\times m}$ with full rank m, vector $b \in \mathbb{C}^n$
Output: Solution $\widehat{x}$ of (5.1)
1. Compute the QR decomposition $A = Q_1 R$
2. Compute $\widehat{x} = R^{-1} Q_1^H b$

In (4.28) we already have seen the Moore–Penrose pseudoinverse of the matrix $H_{k+1,k}$. In general, if $A \in \mathbb{C}^{n\times m}$ has full rank m, then $A^H A \in \mathbb{C}^{m\times m}$ is HPD, and the matrix

$$A^+ := (A^H A)^{-1} A^H$$

is well defined. This matrix A^+ is called the Moore–Penrose pseudoinverse of A. Note that $A^+ A = I_m$, and hence A^+ is a left inverse of A. In general $AA^+ \neq I_n$, but we have the property $(AA^+)^H = AA^+$.
Suppose, as in Theorem 5.3, that

$$A = [Q_1, Q_2] \begin{bmatrix} R \\ 0 \end{bmatrix} = Q_1 R.$$

Since the columns of Q_1 are orthonormal, and $R \in \mathbb{C}^{m\times m}$ is nonsingular,

$$A^+ = (A^H A)^{-1} A^H = (R^H Q_1^H Q_1 R)^{-1} R^H Q_1^H = (R^H R)^{-1} R^H Q_1^H = R^{-1} Q_1^H.$$

Thus, the uniquely determined solution $\widehat{x}$ of the least squares problem (5.1) can be written as $\widehat{x} = A^+ b$, where $A^+ = R^{-1} Q_1^H$.

Since $\widehat{x} = A^+ b = (A^H A)^{-1} A^H b$, the uniquely determined solution of (5.1) is equal to the uniquely determined solution of the linear algebraic system

$$A^H A x = A^H b, \tag{5.2}$$

where the system matrix $A^H A \in \mathbb{C}^{m\times m}$ is HPD. The equations of the system (5.2) are called the *normal equations*. The motivation for this name is that if $\widehat{x}$ solves (5.2), then the residual $r(\widehat{x}) = b - A\widehat{x}$ is normal (or orthogonal) to the range of the columns of A, which is defined by

$$\mathrm{ran}(A) := \{Ay \,:\, y \in \mathbb{C}^m\}.$$

This fact can be seen from

$$\langle r(\widehat{x}), Ay\rangle = \langle A^H r(\widehat{x}), y\rangle = \langle A^H b - A^H A\widehat{x}, y\rangle = \langle 0, y\rangle = 0 \quad \text{for any } y \in \mathbb{C}^m.$$

On the other hand, suppose that $\widetilde{x} \in \mathbb{C}^m$ is any vector for which $r(\widetilde{x}) = b - A\widetilde{x} \perp \operatorname{ran}(A)$. Then for any $y \in \mathbb{C}^m$ we have

$$0 = \langle r(\widetilde{x}), Ay \rangle = \langle A^H r(\widetilde{x}), y \rangle = \langle A^H b - A^H A\widetilde{x}, y \rangle,$$

which implies that $A^H b - A^H A\widetilde{x} = 0$, and hence $\widetilde{x}$ solves the normal equations (5.2). We have thus shown the following result.

Lemma 5.4. *Let $A \in \mathbb{C}^{n\times m}$ with full rank m, $b \in \mathbb{C}^n$, and $\widehat{x} \in \mathbb{C}^m$ be given. Then the following are equivalent:*

(1) $\widehat{x}$ solves the least squares problem (5.1).

(2) $\widehat{x}$ solves the normal equations (5.2).

(3) $r(\widehat{x}) = b - A\widehat{x} \perp \operatorname{ran}(A)$.

Since $A^H A$ is HPD, we may solve the normal equations (5.2) and hence the least squares problem (5.1) with the Cholesky decomposition of $A^H A$. This is done in Algorithm 11. In a practical implementation of this algorithm, step 2 would consist of solving two linear algebraic systems with the triangular Cholesky factors, namely $Ly = A^H b$ and $L^H x = y$.

Algorithm 11 Least squares solution using the Cholesky decomposition

Input: Matrix $A \in \mathbb{C}^{n\times m}$ with full rank m, vector $b \in \mathbb{C}^n$
Output: Solution $\widehat{x}$ of (5.1)
1. Compute the Cholesky decomposition $A^H A = LL^H$
2. Compute $\widehat{x} = L^{-H} L^{-1} A^H b$

Now recall that a matrix $A \in \mathbb{C}^{n\times m}$ with full rank m has an SVD of the form

$$A = U \begin{bmatrix} \Sigma_+ \\ 0 \end{bmatrix} V^H = U_1 \Sigma_+ V^H, \tag{5.3}$$

where $U \in \mathbb{C}^{n\times n}$ and $V \in \mathbb{C}^{m\times m}$ are unitary, and $\Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_m)$ with $\sigma_1 \geq \cdots \geq \sigma_m > 0$; see Theorem 1.7. The matrix U_1 in (5.3) contains the first m columns of U.

Using the SVD we get

$$\kappa_2(A^H A) = \kappa_2(V\Sigma_+^2 V^H) = \|V\Sigma_+^2 V^H\|_2 \, \|V\Sigma_+^{-2} V^H\|_2 = \sigma_1^2 \sigma_m^{-2} = \kappa_2(A)^2.$$

Therefore the system of the normal equations (5.2) can be very ill conditioned, so that a numerical solution of this system can be sensitive to perturbations and rounding errors.

A straightforward computation using the SVD shows that

$$A^+ = (A^H A)^{-1} A^H = V\Sigma_+^{-1} U_1^H,$$

and we immediately get the following result from Theorem 5.3.

Corollary 5.5. *Let $A \in \mathbb{C}^{n\times m}$ with full rank m and $b \in \mathbb{C}^n$ be given. If (5.3) is an SVD of A, then $\widehat{x} := V\Sigma_+^{-1}U_1^H b$ is the uniquely determined solution of the least squares problem (5.1), and the corresponding residual is given by $(I - U_1U_1^H)b$.*

Corollary 5.5 suggests Algorithm 12 for solving the least squares problem (5.1) with the SVD of A. Note that the matrix Σ_+ in the SVD of A is diagonal. Therefore step 2 of Algorithm 12 does not require to solve a linear algebraic system (unlike step 2 of Algorithms 10 and 11).

Algorithm 12 Least squares solution using the SVD

Input: Matrix $A \in \mathbb{C}^{n\times m}$ with full rank m, vector $b \in \mathbb{C}^n$
Output: Solution $\widehat{x}$ of (5.1)
1. Compute the SVD (5.3)
2. Compute $\widehat{x} = V\Sigma_+^{-1}U_1^H b$

The uniquely determined solution $\widehat{x}$ in Corollary 5.5 is given by

$$\widehat{x} = V\Sigma_+^{-1}U_1^H b = \sum_{j=1}^{m} \frac{u_j^H b}{\sigma_j} v_j,$$

where $U_1 = [u_1, \ldots, u_m] \in \mathbb{C}^{n\times m}$ and $V = [v_1, \ldots, v_m] \in \mathbb{C}^{m\times m}$. In the sum on the right hand side, the small singular values of A potentially lead to large coefficients $(u_j^H b)/\sigma_j$. Therefore we can expect that the solution of the least squares problem is sensitive to perturbations when A has (very) small singular values. A backward error result will be shown in the next section.

5.2 The rank deficient least squares problem

In Section 5.1 we have considered the least squares problem (5.1) for a matrix $A \in \mathbb{C}^{n\times m}$ with full rank $m \leq n$. Now we consider the same problem, but $A \in \mathbb{C}^{n\times m}$ has rank $r \leq m \leq n$.
We first show a result analogous to Lemma 5.4, but without the full rank assumption on A.

Lemma 5.6. *Let $A \in \mathbb{C}^{n\times m}$ and $b \in \mathbb{C}^n$, and denote the set of the least squares solutions by*

$$\mathcal{L} := \{x \in \mathbb{C}^m \,:\, \|b - Ax\|_2 \leq \|b - Ay\|_2 \ \text{for all } y \in \mathbb{C}^m\}. \tag{5.4}$$

Then $\widehat{x} \in \mathcal{L}$ if and only if $r(\widehat{x}) = b - A\widehat{x} \perp \mathrm{ran}(A)$.

Proof. Suppose that $\widehat{x} \in \mathbb{C}^n$ satisfies $r(\widehat{x}) = b - A\widehat{x} \perp \mathrm{ran}(A)$, i.e., $A^H r(\widehat{x}) = 0$. Consider any $y \in \mathbb{C}^n$, then

$$b - Ay = b - A\widehat{x} + A\widehat{x} - Ay = r(\widehat{x}) + A(\widehat{x} - y).$$

Thus,

$$\begin{aligned}\|b - Ay\|_2^2 &= (b - Ay)^H(b - Ay) = (r(\widehat{x}) + A(\widehat{x} - y))^H(r(\widehat{x}) + A(\widehat{x} - y)) \\ &= \|r(\widehat{x})\|_2^2 + \|A(\widehat{x} - y)\|_2^2,\end{aligned}$$

which yields

$$\|r(\widehat{x})\|_2^2 = \|b - Ay\|_2^2 - \|A(\widehat{x} - y)\|_2^2 \leq \|b - Ay\|_2^2,$$

and therefore $\widehat{x} \in \mathcal{L}$.
On the other hand, suppose that $\widehat{x} \in \mathbb{C}^n$ satisfies $A^H r(\widehat{x}) = y \neq 0$. Let $\epsilon > 0$ be given and define $x := \widehat{x} + \epsilon y$. Then

$$b - Ax = b - A(\widehat{x} + \epsilon y) = r(\widehat{x}) - \epsilon Ay,$$

which yields

$$\begin{aligned}\|b - Ax\|_2^2 &= (b - Ax)^H(b - Ax) = (r(\widehat{x}) - \epsilon Ay)^H(r(\widehat{x}) - \epsilon Ay) \\ &= \|r(\widehat{x})\|_2^2 - 2\epsilon\|y\|_2^2 + \epsilon^2\|Ay\|_2^2.\end{aligned}$$

We have $y \neq 0$, and therefore choosing a small enough $\epsilon > 0$ shows that $\|b - Ax\|_2 < \|r(\widehat{x})\|_2$. Consequently, $\widehat{x} \notin \mathcal{L}$. □

Lemma 5.6 shows that $\widehat{x}$ is a least squares solution if and only if

$$0 = A^H r(\widehat{x}) = A^H b - A^H A\widehat{x},$$

i.e., if and only if $\widehat{x}$ solves the normal equations.

We next study the uniqueness of the least squares solution when A is rank deficient. Recall from Theorem 1.7 that the SVD of a matrix $A \in \mathbb{C}^{n\times m}$ with rank $r \leq m \leq n$ has the form

$$A = U \begin{bmatrix} \Sigma_+ & 0 \\ 0 & 0 \end{bmatrix} V^H = [U_1, U_2] \begin{bmatrix} \Sigma_+ & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^H \\ V_2^H \end{bmatrix} = U_1 \Sigma_+ V_1^H, \tag{5.5}$$

where $U \in \mathbb{C}^{n\times n}$ and $V \in \mathbb{C}^{m\times m}$ are unitary, and $\Sigma_+ = \operatorname{diag}(\sigma_1, \ldots, \sigma_r)$ with $\sigma_1 \geq \cdots \geq \sigma_r > 0$. In (5.5) we have $U_1 \in \mathbb{C}^{n\times r}$, $U_2 \in \mathbb{C}^{n\times(n-r)}$, $V_1^H \in \mathbb{C}^{r\times m}$, and $V_2^H \in \mathbb{C}^{(m-r)\times m}$.
The *rank-nullity theorem* of Linear Algebra tells us that

$$m = \dim(\operatorname{ran}(A)) + \dim(\ker(A)) = r + \dim(\ker(A)),$$

and hence $\dim(\ker(A)) = m - r$. The matrix $V_2 \in \mathbb{C}^{m\times(m-r)}$ in (5.5) has $m - r$ linearly independent columns. Therefore $AV_2 = U_1\Sigma_+ V_1^H V_2 = 0$ shows that $\ker(A) = \operatorname{ran}(V_2)$. Moreover, we see from (5.5) that $\operatorname{ran}(A) = \operatorname{ran}(U_1)$.

If $x \in \mathbb{C}^m$ and $z \in \ker(A)$, then

$$\|b - A(x + z)\|_2 = \|b - Ax\|_2.$$

Consequently, if $r < m$ and hence $\ker(A)$ is non-trivial, then the solution of (5.1) is not uniquely determined. Moreover, in this case solutions of (5.1) with an arbitrarily large norm exist, since for any given solution $\widehat{x}$ and any $z \in \ker(A)$, the vector $\widehat{x} + z$ is another solution. In order to control this situation it makes sense to search for *minimum norm solutions* of (5.1). Using the SVD we can show that the minimum norm constraint makes the least squares solution uniquely determined.

Theorem 5.7. *Let $A \in \mathbb{C}^{n\times m}$ with rank $r \leq m \leq n$ and the SVD (5.5), and $b \in \mathbb{C}^n$ be given. Then the constrained least squares problem*

$$\arg\min_{x\in\mathcal{L}} \|x\|_2,$$

where $\mathcal{L}$ is as in (5.4), has the uniquely determined solution $\widehat{x} = V_1\Sigma_+^{-1}U_1^H b_1$, and $b - A\widehat{x} = (I - U_1U_1^H)b$.

Proof. The proof is similar to the one of Theorem 5.3. Any $x \in \mathbb{C}^m$ can be written as $x = V\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$ for some $y_1 \in \mathbb{C}^r$ and $y_2 \in \mathbb{C}^{m-r}$. Thus, for any $x \in \mathbb{C}^m$ we have

$$\begin{aligned} \|b - Ax\|_2^2 &= \left\| U\left(U^H b - \begin{bmatrix} \Sigma_+ & 0 \\ 0 & 0 \end{bmatrix} V^H x\right)\right\|_2^2 \\ &= \left\| \begin{bmatrix} U_1^H b - \Sigma_+ y_1 \\ U_2^H b \end{bmatrix}\right\|_2^2 \\ &= \|U_1^H b - \Sigma_+ y_1\|_2^2 + \|U_2^H b\|_2^2, \end{aligned}$$

which yields $\|b - Ax\|_2 \geq \|U_2^H b\|_2$.
The lower bound is attained if and only if $\|U_1^H b - \Sigma_+ y_1\|_2 = 0$, which is equivalent with

$$y_1 = \Sigma_+^{-1}U_1^H b.$$

The vector y_2 does not play any role for the least squares solution. Since

$$\|x\|_2 = \left\| V\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}\right\|_2 = \left\| \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}\right\|_2,$$

we obtain the uniquely determined minimum norm solution of the least squares problem by setting $y_2 = 0$, so that indeed $\widehat{x} = V_1\Sigma_+^{-1}U_1^H b$. A straightforward computation using $A = U_1\Sigma_+V_1^H$ yields $b - A\widehat{x} = (I - U_1U_1^H)b$. □

Theorem 5.7 shows that the rank deficient least squares problem, i.e., the problem (5.1) with A having rank $r < m$, can also be solved by Algorithm 12. Moreover, the proof of Theorem 5.7 shows that $\mathcal{L} = \{V_1\Sigma_+^{-1}U_1^H b + V_2 y \,:\, y \in \mathbb{C}^{m-r}\}$.

We now look at backward error results for the least squares problem. We consider a given $A \in \mathbb{C}^{n\times m}$ and $b \in \mathbb{C}^n$. Let

$$\widetilde{x} \in \mathbb{C}^m \setminus \{0\} \quad \text{and} \quad \widetilde{r} := b - A\widetilde{x}.$$

Then for the matrix

$$E_1 := \frac{\widetilde{r}\widetilde{x}^H}{\|\widetilde{x}\|_2^2} \quad \text{with} \quad \|E_1\|_2 = \frac{\|\widetilde{r}\|_2}{\|\widetilde{x}\|_2}, \tag{5.6}$$

we get $b - (A + E_1)\widetilde{x} = b - A\widetilde{x} - \widetilde{r} = 0$, and hence $\widetilde{x}$ solves the least squares problem

$$\arg\min_{x \in \mathbb{C}^m} \|b - (A + E_1)x\|_2.$$

A simple modification of the proof of Theorem 2.19 shows that

$$\frac{\|E_1\|_2}{\|A\|_2} = \min\left\{ \frac{\|\Delta A\|_2}{\|A\|_2} \ : \ (A + \Delta A)\widetilde{x} = b \right\}.$$

A natural question is whether the given approximate solution $\widetilde{x} \in \mathbb{C}^m \backslash \{0\}$ solves perturbed least squares problems with A where the norm of the perturbation matrix is smaller than $\|E_1\|_2$. One suggestion, due to Stewart [35], is to use the matrix

$$E_2 := \frac{v\widetilde{x}^H}{\|\widetilde{x}\|_2^2}, \quad v := \widetilde{r} - r, \tag{5.7}$$

where $\widehat{x}$ is a solution of the least squares problem (5.1), and $r = b - A\widehat{x}$ is the corresponding residual. Then

$$v = \widetilde{r} - r = A(\widehat{x} - \widetilde{x}) \ \in \ \mathrm{ran}(A).$$

Since $r \perp \mathrm{ran}(A)$ by Lemma 5.6, we get

$$\|\widetilde{r}\|_2^2 = \|v + r\|_2^2 = \|v\|_2^2 + \|r\|_2^2,$$

and hence

$$\|E_2\|_2 = \frac{\|v\|_2}{\|\widetilde{x}\|_2} = \frac{(\|\widetilde{r}\|_2^2 - \|r\|_2^2)^{1/2}}{\|\widetilde{x}\|_2} \leq \frac{\|\widetilde{r}\|_2}{\|\widetilde{x}\|_2} = \|E_1\|_2,$$

with equality if and only if $r = 0$. Moreover,

$$b - (A + E_2)\widetilde{x} = b - A\widetilde{x} - (\widetilde{r} - r) = r.$$

Using $A^H r = 0$, which holds since $\widehat{x}$ is a least squares solution, we obtain

$$(A + E_2)^H r = A^H r + \frac{\widetilde{x}}{\|\widetilde{x}\|_2^2}(\widehat{x} - \widetilde{x})^H A^H r = 0,$$

i.e., $b - (A + E_2)\widetilde{x} \perp \mathrm{ran}(A + E_2)$. Lemma 5.6 now implies that $\widetilde{x}$ solves the least squares problem

$$\arg\min_{x \in \mathbb{C}^m} \|b - (A + E_2)x\|_2.$$

While $\|E_2\|_2$ is potentially much smaller than $\|E_1\|_2$, a significant disadvantage of the perturbation E_2 is that it depends on the solution $\widehat{x}$ of the unperturbed least squares problem, which usually is not known.
The following result of Stewart [36] gives an improvement over the perturbations E_1 and E_2 in (5.6) and (5.7), respectively.

Theorem 5.8. *Let $A \in \mathbb{C}^{n\times m}$, $b \in \mathbb{C}^n$, and $\widetilde{x} \in \mathbb{C}^m$ be given, and suppose that $\widetilde{r} := b - A\widetilde{x} \neq 0$. Define*

$$E_3 := -\frac{\widetilde{r}\widetilde{r}^H A}{\|\widetilde{r}\|_2^2}$$

then $\|E_3\|_2 = \|A^H\widetilde{r}\|_2/\|\widetilde{r}\|_2$, and $\widetilde{x}$ solves the least squares problem

$$\arg\min_{x\in\mathbb{C}^m} \|b - (A + E_3)x\|_2.$$

Proof. We first observe that

$$\|E_3\|_2 = \|E_3^H\|_2 = \frac{1}{\|\widetilde{r}\|_2}\left\|A^H\widetilde{r}\frac{\widetilde{r}^H}{\|\widetilde{r}\|_2}\right\|_2 = \frac{\|A^H\widetilde{r}\|_2}{\|\widetilde{r}\|_2};$$

cf. the proof of Theorem 2.19.
Lemma 5.6 shows that $\widetilde{x}$ solves the least squares problem with the matrix $A + E_3$ if and only if $b - (A + E_3)\widetilde{x} \perp \operatorname{ran}(A + E_3)$. For the given E_3 we have

$$E_3^H E_3 = \frac{A^H\widetilde{r}\widetilde{r}^H}{\|\widetilde{r}\|_2^2}\frac{\widetilde{r}\widetilde{r}^H A}{\|\widetilde{r}\|_2^2} = -A^H E_3.$$

Therefore,

$$\begin{aligned}(A + E_3)^H(b - (A + E_3)\widetilde{x}) &= (A + E_3)^H\widetilde{r} - (A + E_3)^H E_3\widetilde{x}\\ &= A^H\widetilde{r} - \frac{A^H\widetilde{r}\widetilde{r}^H}{\|\widetilde{r}\|_2^2}\widetilde{r} - A^H E_3\widetilde{x} + E_3^H E_3\widetilde{x}\\ &= 0,\end{aligned}$$

which shows the claim. □

Unlike the matrix E_2 in (5.7), the matrix E_3 in this theorem is easily computable because $\widetilde{x}$ is a given approximation. The backward error norm $\|E_3\|_2$ is small when the unit norm vector $\widetilde{r}/\|\widetilde{r}\|_2$ is close to being orthogonal to $\operatorname{ran}(A)$. We know from Lemma 5.6 that the exact orthogonality of the residual $\widetilde{r}$ to $\operatorname{ran}(A)$ is necessary and sufficient for $\widetilde{x}$ to be a solution of the unperturbed least squares problem. Now we see that any deviation of the exact orthogonality yields a backward perturbation whose norm corresponds to how much of the orthogonality is lost. Further backward error results for the least squares problem can be found in [40, Section III.5] and [45].

5.3 The SVD and low rank approximation

Consider a matrix $A \in \mathbb{C}^{n\times m}$ with rank $r \leq m \leq n$ and its SVD (5.5), then

$$A = \sum_{j=1}^{r} \sigma_j u_j v_j^H,$$

i.e., A is decomposed into a sum of the r rank-one matrices, where

$$\|\sigma_j u_j v_j^H\|_2 = \sigma_j, \quad j = 1, \ldots, r.$$

Since the singular values are ordered decreasingly, the "weight" or "importance" of the matrices in the sum potentially decreases with increasing j.
In many applications, in particular in the data sciences, it can be observed that data matrices A have singular values which decay quickly, i.e., there are only a few singular values that are much larger than the rest. In such cases only a few terms in the sum above are sufficient for a good approximation of A, i.e.,

$$A \;\approx\; A_k := \sum_{j=1}^{k} \sigma_j u_j v_j^H, \tag{5.8}$$

for some $k < r$ and with respect to some appropriate norm.
Let us make this more precise. Given the SVD (5.5) of A, it is clear that the matrix A_k in (5.8) has the SVD

$$A_k = U \begin{bmatrix} \Sigma_+^{(k)} & 0 \\ 0 & 0 \end{bmatrix} V^H.$$

where $\Sigma_+^{(k)} = \operatorname{diag}(\sigma_1, \ldots, \sigma_k)$. Since the 2-norm is unitarily invariant,

$$\|A - A_k\|_2 = \|U \operatorname{diag}(0, \ldots, 0, \sigma_{k+1}, \ldots, \sigma_r, 0, \ldots, 0) V^H\|_2 = \sigma_{k+1}.$$

This holds for all $k = 1, \ldots, r$ when we set $\sigma_{r+1} := 0$. If σ_{k+1} is very small (relative to σ_1) for some $k \ll r$, then the matrix A is *approximately low rank*; see [5, 44] for examples and applications. Note that storing the matrix A_k requires k scalars and $2k$ vectors, in general $k(n+m+1)$ numbers. For small k this can be much smaller than the nm numbers required (in general) to store the matrix A. In addition to the advantage of using less storage, operating (for example multiplying) with A_k costs fewer arithmetic operations than operating with A.
Note that we also have

$$\kappa_2(A_k) = \frac{\sigma_1}{\sigma_k} = \frac{\sigma_r}{\sigma_k} \kappa_2(A).$$

Since A_k potentially has a much smaller condition number than A, solving problems with A_k numerically may be more stable than solving problems with A.

The next result shows that the SVD not only yields *some* but *the best* low rank approximation of A with respect to the 2-norm.

Theorem 5.9. *The matrix A_k defined in* (5.8) *satisfies*

$$\sigma_{k+1} = \|A - A_k\|_2 \le \|A - B\|_2 \quad \textit{for every } B \in \mathbb{C}^{n \times m} \textit{ with rank } k.$$

Proof. If $B \in \mathbb{C}^{n\times m}$ has rank k, then $\dim(\ker(B)) = m - k$. Let $v_1, \ldots, v_m$ be the right singular vectors of A, then $\operatorname{span}\{v_1, \ldots, v_{k+1}\}$ has dimension $k+1$. We have

$$\ker(B) \cap \operatorname{span}\{v_1, \ldots, v_{k+1}\} \neq \{0\},$$

since these are two subspaces of $\mathbb{C}^m$ with the sum of their dimensions equal to $m+1$. Let $v \in \ker(B) \cap \operatorname{span}\{v_1, \ldots, v_{k+1}\}$ be a vector with unit 2-norm, then v can be written as

$$v = \sum_{j=1}^{k+1} \alpha_j v_j \quad \text{with} \quad \sum_{j=1}^{k+1} |\alpha_j|^2 = 1,$$

and since $Bv = 0$ we obtain

$$(A - B)v = Av = \sum_{j=1}^{k+1} \alpha_j A v_j = \sum_{j=1}^{k+1} \alpha_j \sigma_j u_j.$$

Consequently,

$$\begin{aligned}\|A - B\|_2 = \max_{\|x\|_2=1} \|(A-B)x\|_2 \geq \|(A-B)v\|_2 = \left\| \sum_{j=1}^{k+1} \alpha_j \sigma_j u_j \right\|_2 = \left(\sum_{j=1}^{k+1} |\alpha_j|^2 \sigma_j^2 \right)^{1/2} \\ \geq \sigma_{k+1} \left(\sum_{j=1}^{k+1} |\alpha_j|^2 \right)^{1/2} = \sigma_{k+1},\end{aligned}$$

and noting that $\sigma_{k+1} = \|A - A_k\|_2$ completes the proof. □

Theorem 5.9 shows that the relative error of the approximation of A by A_k, which is given by

$$\frac{\|A - A_k\|_2}{\|A\|_2} = \frac{\sigma_{k+1}}{\sigma_1},$$

is the smallest possible (with respect to the 2-norm) for all approximations of A with rank k.

Chapter 6

Perturbation of Eigenvalue Problems

In Chapter 2 we looked at the conditioning of a problem and we studied matrix perturbation theory with a focus on perturbations of linear algebraic systems. We will now complement that analysis by a closer look at the perturbation theory for eigenvalue problems

$$Ax = \lambda x, \quad A \in \mathbb{C}^{n\times n}.$$

In this chapter and the following ones on the numerical solution of eigenvalue problems, we will mostly use the 2-norm (or Euclidean norm) on $\mathbb{C}^n$ and the corresponding matrix 2-norm,

$$\|x\|_2 := (x^H x)^{1/2} \quad \text{and} \quad \|A\|_2 := \max_{\substack{x\in\mathbb{C}^n \\ \|x\|=1}} \|Ax\|.$$

We already know that $\|A\|_2 = \lambda_{\max}(A^H A)^{1/2} = \sigma_1$; see Theorem 1.7 and Example 2.6.

6.1 Basic concepts and definitions

Recall from Linear Algebra that for each $A \in \mathbb{C}^{n\times n}$ there exists a *Jordan decomposition* of the form

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

where $J = \operatorname{diag}\big(J_{d_1}(\lambda_1), \ldots, J_{d_m}(\lambda_m)\big) \in \mathbb{C}^{n\times n}$ is called a *Jordan canonical form* of A with *Jordan blocks* of the form

$$J_{d_i}(\lambda_i) = \begin{bmatrix} \lambda_i & 1 & & \\ & \ddots & \ddots & \\ & & & 1 \\ & & & \lambda_i \end{bmatrix} \in \mathbb{C}^{d_i\times d_i}, \quad i = 1, \ldots, m.$$

We can write

$$J_{d_i}(\lambda_i) = \lambda I_d + J_{d_i}(0),$$

where the Jordan block $J_{d_i}(0)$ is *nilpotent* of degree d_i, i.e.,

$$J_{d_i}(0)^{d_i-1} \neq 0 \quad \text{and} \quad J_{d_i}(0)^{d_i} = 0.$$

The Jordan canonical form is uniquely determined by A up to permutations of the Jordan blocks on the diagonal. If $d_1 = \cdots = d_m = 1$ (and thus $m = n$), then A is called *diagonalizable* and we sometimes write $A = X\Lambda X^{-1}$ with $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$.
The n numbers

$$\underbrace{\lambda_1, \ldots, \lambda_1}_{d_1}, \ldots, \underbrace{\lambda_m, \ldots, \lambda_m}_{d_m}, \quad d_1 + \ldots + d_m = n,$$

are the *eigenvalues* of A. Here $\lambda_1, \ldots, \lambda_m$ need not be pairwise distinct. If they are, i.e., if for each of the distinct eigenvalues of A there exists exactly one Jordan block, then A is called *nonderogatory*.
An eigenvalue of A is called *simple* when J contains only one corresponding Jordan block of size 1×1.
The size of the largest Jordan block corresponding to an eigenvalue λ of A is called the *index* of λ.
The following result gives basic information about the location of the eigenvalues of a matrix A in terms of its entries.

Theorem 6.1. *Each eigenvalue λ of $A = [a_{ij}] \in \mathbb{C}^{n\times n}$ satisfies*

$$\lambda \in \bigcup_{i=1}^{n} R_i, \quad R_i := \{z \in \mathbb{C} : |z - a_{ii}| \leq r_i\}, \quad r_i := \sum_{j\neq i} |a_{ij}|,$$
$$\lambda \in \bigcup_{j=1}^{n} C_j, \quad C_j := \{z \in \mathbb{C} : |z - a_{jj}| \leq c_j\}, \quad c_j := \sum_{i\neq j} |a_{ij}|.$$

Proof. Let $Ax = \lambda x$ with $[x_1, \ldots, x_n]^T \neq 0$, and choose x_i so that $|x_i| = \|x\|_\infty$. The ith equation of $(\lambda I - A)x = 0$ can be written as

$$(\lambda - a_{ii})x_i = -\sum_{j\neq i} a_{ij}x_j.$$

Dividing by $x_i \neq 0$ and taking absolute values yields

$$|\lambda - a_{ii}| = \left|\sum_{j\neq i} a_{ij}\frac{x_j}{x_i}\right| \leq \sum_{j\neq i} |a_{ij}| = r_i,$$

where we have also used that $|x_j/x_i| \leq 1$. Thus, $\lambda \in R_i$. Since the matrices A and A^T have the same eigenvalues, applying the same proof to A^T shows the result for the sets C_i. □

This result is called the *Gershgorin circle theorem*, and the sets R_i and C_i are called *Gershgorin disks*.

A nonzero vector x that satisfies $Ax = \lambda x$ is sometimes called a *right eigenvector* of A corresponding to the eigenvalue λ. A nonzero vector y with

$$y^H A = \lambda y^H$$

is a *left eigenvector* of A corresponding to the eigenvalue λ. If y is a left eigenvector, then

$$A^H y = \overline{\lambda} y.$$

Hence the left eigenvectors of A corresponding to the eigenvalue λ are the right eigenvectors of A^H corresponding to the eigenvalue $\overline{\lambda}$.
Let λ be a simple eigenvalue of A. If $x = Xe_i$ is a right eigenvector corresponding to λ, then each corresponding left eigenvector y is of the form $y = \alpha X^{-H} e_i$ for some $\alpha \neq 0$, so that $y^H x = \overline{\alpha} \neq 0$. (This property is not guaranteed when λ is not simple.) Consider an arbitrary perturbation matrix $\Delta A \in \mathbb{C}^{n \times n}$. It can be shown using classical results from analytic function theory that in a neighborhood of the origin there exist differentiable functions $x(t)$ and $\lambda(t)$ such that

$$(A + t\Delta A)x(t) = \lambda(t)x(t), \quad \text{where} \quad x(0) = x, \quad \lambda(0) = \lambda.$$

Differentiating with respect to t yields

$$(A + t\Delta A)x'(t) + \Delta A x(t) = \lambda'(t)x(t) + \lambda(t)x'(t),$$

and for $t = 0$ we have

$$Ax'(0) + \Delta A x = \lambda'(0)x + \lambda x'(0).$$

Multiplying from the left with y^H leads to

$$\lambda y^H x'(0) + y^H(\Delta A)x = \lambda'(0)y^H x + \lambda y^H x'(0),$$

hence

$$|\lambda'(0)| = \frac{|y^H(\Delta A)x|}{|y^H x|} \leq \|\Delta A\|_2 \kappa_2(\lambda), \quad \text{where} \quad \kappa_2(\lambda) := \frac{\|x\|_2 \|y\|_2}{|y^H x|}. \tag{6.1}$$

The quantity

$$|\lambda'(0)| = \lim_{t \to 0} \left| \frac{\lambda - \lambda(t)}{t} \right|$$

measures the sensitivity of the eigenvalue $\lambda = \lambda(0)$ under small perturbations of A. The inequality in (6.1) shows that ε–perturbations in A can lead to changes of order $O(\varepsilon\kappa(\lambda))$ in λ. The quantity $\kappa(\lambda)$ is called the *condition number* of the (simple) eigenvalue λ, and (6.1) is another example of the general rule of thumb

$$\text{forward error} \ \leq \ \text{condition number} \ \times \ \text{backward error};$$

see (2.2).

As already mentioned above, if $x = Xe_i$ is a right eigenvector corresponding to the simple eigenvalue λ, then each corresponding left eigenvector y is of the form $y = \alpha X^{-H} e_i$ for some $\alpha \neq 0$. We therefore get

$$\begin{aligned} |y^H x| &= |\alpha e_i^T X^{-1} X e_i| = |\alpha|, \\ \|x\|_2 &= \|X e_i\|_2 \leq \|X\|_2, \\ \|y\|_2 &= \|\alpha X^{-H} e_i\|_2 \leq |\alpha| \, \|X^{-H}\|_2 = |\alpha| \, \|X^{-1}\|_2, \end{aligned}$$

and hence

$$\kappa_2(\lambda) \leq \kappa_2(X).$$

The (forward) error when approximating an eigenvalue λ of A by $\widehat{\lambda}$ can be measured simply by the difference $|\lambda - \widehat{\lambda}|$. On the other hand, for measuring the error when approximating an eigenvector x of A by $\widehat{x}$, it makes no sense to consider $\|x - \widehat{x}\|$ (in any norm). Since every (nonzero) scalar multiple of x also is an eigenvector of A, we rather need to analyze how well $\operatorname{span}\{\widehat{x}\}$ approximates an eigenspace of A. For this analysis we need the concept of angles between subspaces. For any two subspaces $\mathcal{Y}, \mathcal{Z}$ of $\mathbb{C}^n$, the *minimal canonical angle* $\theta_{\min}(\mathcal{Y}, \mathcal{Z}) \in [0, \frac{\pi}{2}]$ between $\mathcal{Y}$ and $\mathcal{Z}$ is defined by

$$\cos \theta_{\min}(\mathcal{Y}, \mathcal{Z}) := \max_{\substack{y \in \mathcal{Y} \setminus \{0\} \\ z \in \mathcal{Z} \setminus \{0\}}} \frac{|z^H y|}{\|y\|_2 \|z\|_2}. \tag{6.2}$$

The angle $\theta_{\min}(\mathcal{Y}, \mathcal{Z})$ is well defined since any pair of vectors $y, z \in \mathbb{C}^n \setminus \{0\}$ satisfies the Cauchy–Schwarz inequality

$$|z^H y| \leq \|y\|_2 \|z\|_2,$$

and hence $0 \leq \frac{|z^H y|}{\|y\|_2 \|z\|_2} \leq 1$. For $\theta_{\min}(\mathcal{Y}, \mathcal{Z}) \in [0, \frac{\pi}{2}]$ we have

$$\sin \theta_{\min}(\mathcal{Y}, \mathcal{Z}) = \left(1 - \cos^2 \theta_{\min}(\mathcal{Y}, \mathcal{Z})\right)^{1/2},$$

and $\sin^2 \theta_{\min}(\mathcal{Y}, \mathcal{Z}) + \cos^2 \theta_{\min}(\mathcal{Y}, \mathcal{Z}) = 1$.
In the special case $\mathcal{Y} = \operatorname{span}\{y\}$ and $\mathcal{Z} = \operatorname{span}\{z\}$ for two nonzero vectors y and z, the minimal canonical angle is given by

$$\cos \theta_{\min}(\mathcal{Y}, \mathcal{Z}) = \frac{|z^H y|}{\|y\|_2 \|z\|_2}, \tag{6.3}$$

and we write

$$\cos \theta(y, z) \text{ instead of } \cos \theta_{\min}(\mathcal{Y}, \mathcal{Z}), \text{ and } \sin \theta(y, z) \text{ instead of } \sin \theta_{\min}(\mathcal{Y}, \mathcal{Z})$$

for simplicity. The two subspaces $\mathcal{Y}$ and $\mathcal{Z}$ are "close" when the angle $\theta(y, z)$ is "small", i.e., when $\cos \theta(y, z)$ is close to 1, which is equivalent with $\sin \theta(y, z)$ being close to 0. The largest possible angle $\theta(y, z) = \frac{\pi}{2}$ occurs when $y \perp z$, giving $\cos \theta(y, z) = 0$ and $\sin \theta(y, z) = 1$.
If λ is a simple eigenvalue of A with left and right eigenvectors y and x, then

$$\cos \theta(x, y) = \frac{1}{\kappa_2(\lambda)};$$

see (6.1).

6.2 Forward error bounds

In this section we will study bounds on the forward error for approximate eigenvalues and eigenvectors of a given matrix $A \in \mathbb{C}^{n\times n}$.
The following lemma is quite general, and in particular applies to Jordan blocks of the form $\lambda I_d + J_d(0)$ with $\lambda \neq 0$.

Lemma 6.2. *Let $A \in \mathbb{C}^{n\times n}$ be nonsingular and $B \in \mathbb{C}^{n\times n}$ be nilpotent of degree d. If $AB = BA$, then $A + B$ is nonsingular with*

$$(A+B)^{-1} = \sum_{j=1}^{d} A^{-j}(-B)^{j-1}.$$

Proof. Since A is nonsingular, the sum $\sum_{j=1}^{d} A^{-j}(-B)^{j-1}$ is well defined. A straightforward computation using the assumption that A and B commute now shows that

$$(A+B)\sum_{j=1}^{d} A^{-j}(-B)^{j-1} = I_n,$$

which implies the result. □

This lemma is used in the proof of the following theorem, which gives a general forward error bound for an approximate eigenvalue of A.

Theorem 6.3. *Suppose that $(\widehat{\lambda}, \widehat{x})$ is an approximate eigenpair of $A = XJX^{-1} \in \mathbb{C}^{n\times n}$ with $\|\widehat{x}\|_2 = 1$, and let $r := A\widehat{x} - \widehat{\lambda}\widehat{x}$ be the corresponding* residual. *Then there exists an eigenvalue λ of A with index d, such that*

$$\frac{|\lambda - \widehat{\lambda}|^d}{\sum_{j=0}^{d-1} |\lambda - \widehat{\lambda}|^j} \leq \kappa_2(X)\,\|r\|_2. \tag{6.4}$$

In particular, if A is diagonalizable, then there exists an eigenvalue λ of A, such that

$$|\lambda - \widehat{\lambda}| \leq \kappa_2(X)\,\|r\|_2. \tag{6.5}$$

Proof. When $\widehat{\lambda}$ is an eigenvalue of A both inequalities trivially hold for $\lambda = \widehat{\lambda}$, since then the left hand sides are zero.
We will now prove (6.4) under the assumption that $\widehat{\lambda}$ is not an eigenvalue of A. Then $A - \widehat{\lambda}I_n$ is nonsingular, and from $r = (A - \widehat{\lambda}I_n)\widehat{x}$ we obtain

$$\widehat{x} = (A - \widehat{\lambda}I_n)^{-1} r.$$

Hence

$$1 = \|\widehat{x}\|_2 = \|(A - \widehat{\lambda}I_n)^{-1} r\|_2 = \|X(J - \widehat{\lambda}I_n)^{-1}X^{-1}r\|_2 \leq \kappa_2(X)\,\|(J - \widehat{\lambda}I_n)^{-1}\|_2\,\|r\|_2,$$

so that

$$\kappa_2(X)\,\|r\|_2 \geq \frac{1}{\|(J-\widehat{\lambda}I_n)^{-1}\|_2} = \left(\max_{1\leq i\leq m}\|(J_{d_i}(\lambda_i)-\widehat{\lambda}I_{d_i})^{-1}\|_2\right)^{-1}$$
$$= \left(\max_{1\leq i\leq m}\|((\lambda_i-\widehat{\lambda})I_{d_i}+J_{d_i}(0))^{-1}\|_2\right)^{-1}$$
$$= \left(\|((\lambda_i-\widehat{\lambda})I_{d_i}+J_{d_i}(0))^{-1}\|_2\right)^{-1}, \tag{6.6}$$

for some $i \in \{1,\ldots,m\}$. In the first equality we have used that the 2-norm of a block diagonal matrix is equal to the maximal 2-norm of the diagonal blocks. The maximum in the last expression is attained for some eigenvalue λ_i. The corresponding block size d_i is the index of λ_i, since the norm of the diagonal blocks corresponding to the same λ_i strictly increases with the size of the blocks. Let us drop the index i for simplicity of notation, i.e., write $\lambda = \lambda_i$ and $d = d_i$.
Using Lemma 6.2 and $\|J_d(0)\|_2 = 1$ we now get

$$\|((\lambda-\widehat{\lambda})I_d+J_d(0))^{-1}\|_2 = \Big\|\sum_{j=1}^{d}(\lambda-\widehat{\lambda})^{-j}(-J_d(0))^{j-1}\Big\|_2$$
$$\leq \sum_{j=1}^{d}|\lambda-\widehat{\lambda}|^{-j} = |\lambda-\widehat{\lambda}|^{-d}\sum_{j=0}^{d-1}|\lambda-\widehat{\lambda}|^{j}.$$

Rearranging the inequality and inserting it in (6.6) shows that (6.4) holds.
For a diagonalizable matrix each eigenvalue has index 1, and thus (6.5) immediately follows from (6.4). □

Since

$$\sum_{j=0}^{d-1}|\lambda-\widehat{\lambda}|^{j} \;\leq\; (1+|\lambda-\widehat{\lambda}|)^{d-1},$$

the inequality (6.4) implies the weaker inequality

$$\frac{|\lambda-\widehat{\lambda}|^{d}}{(1+|\lambda-\widehat{\lambda}|_2)^{d-1}} \leq \kappa_2(X)\,\|r\|_2.$$

Both inequalities show that when $\kappa_2(X)$ is small, a small residual norm $\|r\|_2$ guarantees that the approximate eigenvalue $\widehat{\lambda}$ is close to an exact eigenvalue λ of A.
In particular, we have the following important corollary of Theorem 6.3 for normal matrices, which are unitarily diagonalizable.

Corollary 6.4. *In the notation of Theorem 6.3, suppose that $A \in \mathbb{C}^{n\times n}$ is normal and that $(\widehat{\lambda},\widehat{x})$ is an approximate eigenpair with $\|\widehat{x}\|_2 = 1$. Then there exists an eigenvalue λ of A, such that $|\lambda-\widehat{\lambda}| \leq \|r\|_2$.*

On the other hand, when $\kappa_2(X)$ is large, a small residual norm $\|r\|_2$ does *not* imply that $\widehat{\lambda}$ is close to an eigenvalue of A.

Example 6.5. *Let $0 < \varepsilon \ll 1$ and consider the matrix*

$$A = \begin{bmatrix} 1 & \varepsilon^{-1} \\ 0 & 1+\varepsilon \end{bmatrix}.$$

Then A has the two eigenvalues 1 and $1+\varepsilon$, and it is easily verified that

$$A = X \begin{bmatrix} 1 & 0 \\ 0 & 1+\varepsilon \end{bmatrix} X^{-1}, \quad \text{where} \quad X = \begin{bmatrix} 1 & 1 \\ 0 & \varepsilon^2 \end{bmatrix}, \quad X^{-1} = \begin{bmatrix} 1 & -\varepsilon^{-2} \\ 0 & \varepsilon^{-2} \end{bmatrix}.$$

We have $\|X\|_2 \approx 1$ and $\|X^{-1}\|_2 \approx \varepsilon^{-2}$, and hence $\kappa_2(X) \approx \varepsilon^{-2}$. For the approximate eigenpair

$$(\widehat{\lambda}, \widehat{x}) = \left(2, \begin{bmatrix} (1-\varepsilon^2)^{1/2} \\ \varepsilon \end{bmatrix}\right)$$

we have $\|\widehat{x}\|_2 = 1$ and

$$r = A \begin{bmatrix} (1-\varepsilon^2)^{1/2} \\ \varepsilon \end{bmatrix} - 2 \begin{bmatrix} (1-\varepsilon^2)^{1/2} \\ \varepsilon \end{bmatrix} = \begin{bmatrix} 1-(1-\varepsilon^2)^{1/2} \\ \varepsilon^2 - \varepsilon \end{bmatrix}.$$

Using $(1-\varepsilon^2)^{1/2} = 1 - \frac{1}{2}\varepsilon^2 + O(\varepsilon^4)$ we see that $\|r\|_2 \approx \varepsilon$. Hence $\|r\|_2 \to 0$ for $\varepsilon \to 0$, while the distance of $\widehat{\lambda}$ to the closest eigenvalue of A is given by $1-\varepsilon$. This distance even increases for $\varepsilon \to 0$.
We can also compute the condition numbers of the two simple eigenvalues of A:
Left and right eigenvectors corresponding to the eigenvalue 1 are $y = [1, -\varepsilon^{-2}]^T$ and $x = [1, 0]^T$, so that $y^T x = 1$ and

$$\kappa_2(1) = \|x\|_2 \|y\|_2 = (1+\varepsilon^{-4})^{1/2} \approx \varepsilon^{-2}.$$

Left and right eigenvectors corresponding to the eigenvalue $1+\varepsilon$ are $y = [0, \varepsilon^{-2}]^T$ and $x = [1, \varepsilon^2]^T$, so that $y^T x = 1$ and

$$\kappa_2(1+\varepsilon) = \|x\|_2 \|y\|_2 = \varepsilon^{-2}(1+\varepsilon^4)^{1/2} \approx \varepsilon^{-2}.$$

Here both individual condition numbers are on the same order as the condition number of the eigenvector matrix X. (Recall that in general $\kappa_2(\lambda) \le \kappa_2(X)$.)

Example 6.6. *We consider the* symmetric *Toeplitz matrix $A \in \mathbb{R}^{100\times 100}$ generated in MATLAB by*

```
n=100;ee=ones(n,1);
```

```
A=spdiags([ee,-4*ee,6*ee,-4*ee,ee],[-2:2],n,n);
```

Computing an eigendecomposition using `[X,D]=eig(full(A))` *in MATLAB yields matrices that satisfy*

```
cond(X)=1.000000000000001
norm(A*X-X*D)=1.957455527332561e-14
```

The computed approximations of the (real) eigenvalues of A are shown in Figure 6.1 (left). Since the problem is very well conditioned and the residuals are very small, the bound (6.5) *guarantees that the computed approximations of the eigenvalues of A are close to the actual eigenvalues.*
We now consider the nonsymmetric *Toeplitz matrix generated by*

```
n=100;ee=ones(n,1);
B=spdiags([ee,-4*ee,6*ee,-4*ee,ee],[-1:3],n,n);
```

Thus, B is generated by "shifting up" the nonzero diagonals of A. It can be shown analytically that B is diagonalizable with distinct real eigenvalues. Computing an eigendecomposition using `[X,D]=eig(full(B))` *in MATLAB yields matrices that satisfy*

```
cond(X)=1.073986146281143e+24
norm(B*X-X*D)=4.047349790474101e-13
```

The computed approximations of the eigenvalues of B are shown by the pluses in Figure 6.1 (right). Since the problem is highly ill conditioned, the bound (6.5) *is useless, and the relatively small residual norm does not guarantee that the computed approximations are close to the exact eigenvalues. Here they are, which is related to the Hessenberg structure of B. This structure is well suited for the application of the QR algorithm (see Algorithm 17), which is used in MATLAB's* `eig`.
The ill-conditioning of the problem becomes apparent when we consider the matrix B^T. Executing `[X,D]=eig(full(B'))` *yields matrices with*

```
cond(X)=6.122470693750833e+16
norm(B'*X-X*D)=3.163271579912171e-13
```

The computed approximations of the eigenvalues of B^T, which in theory coincide with those of B, are shown by the circles in Figure 6.1 (right).

The *Rayleigh quotient* of a matrix $A \in \mathbb{C}^{n\times n}$ and a nonzero vector $y \in \mathbb{C}^n$ is defined as

$$R_A(y) := \frac{y^H A y}{y^H y} = \frac{y^H A y}{\|y\|_2^2}. \tag{6.7}$$

It is easy to show the following properties of the Rayleigh quotient:

(1) Homogeneity: $R_A(\alpha y) = R_A(y)$ for all nonzero $\alpha \in \mathbb{C}$.

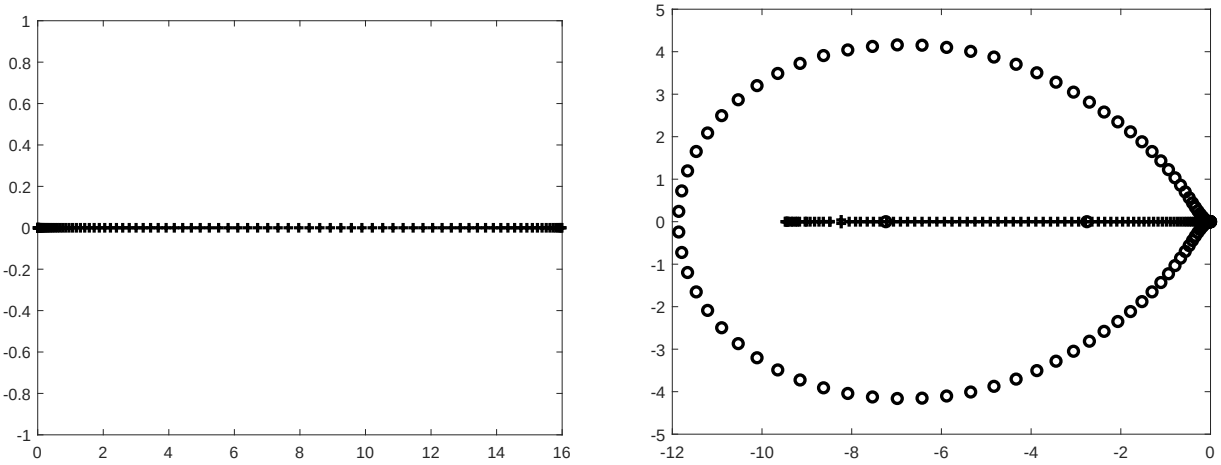


Figure 6.1: Computed approximations of the eigenvalues of some Toeplitz matrices.

(2) Unitary invariance: $R_A(y) = R_{UAU^H}(Uy)$ for every unitary matrix $U \in \mathbb{C}^{n\times n}$.

(3) Translation invariance: $R_{A-\alpha I_n}(y) = R_A(y) - \alpha$ for every $\alpha \in \mathbb{C}$.

(4) Orthogonality of the residual: For $r := Ay - R_A(y)y$ we have $y \perp r$.

(5) If $A = A^H$, then $R_A(y) \in \mathbb{R}$ (even when A and y are not real).

(6) If A is normal, then $R_A(y)$ is in the convex hull of the eigenvalues of A. In particular, $R_A(y) \in [\lambda_{\min}(A), \lambda_{\max}(A)]$ if $A = A^H$.

In order to see that (6) holds, let $A = UDU^H$ be a unitary diagonalization of the normal matrix A, where $D = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$, and let $x := U^H y/\|y\|_2 = [x_1, \ldots, x_n]^T$. Then $\|x\|_2 = 1$ and

$$R_A(y) = \frac{y^H U D U^H y}{\|y\|_2^2} = x^H D x = \sum_{j=1}^{n} \lambda_j |x_j|^2,$$

which shows that $R_A(y)$ is a convex combination of the eigenvalues of A.

If x is an eigenvector of A corresponding to the eigenvalue λ, then $R_A(x) = \lambda$. Thus, if $\widehat{x}$ is close to an eigenvector of A, the Rayleigh quotient $R_A(\widehat{x})$ appears to be a good candidate for the corresponding eigenvalue approximation. For Hermitian matrices this statement is made more precise in the next result, which improves the bound in Corollary 6.4 for the special choice $\widehat{\lambda} = R_A(\widehat{x})$.

Theorem 6.7. *Let $A = A^H \in \mathbb{C}^{n\times n}$, let $\widehat{x} \in \mathbb{C}^n$ be a unit norm vector, and define $\widehat{\lambda} := R_A(\widehat{x})$.*

(1) If $\widehat{\lambda} \in (\alpha, \beta)$ and (α, β) contains no eigenvalue of A, then

$$(\beta - \widehat{\lambda})(\widehat{\lambda} - \alpha) \leq \|r\|_2^2, \tag{6.8}$$

where $r = A\widehat{x} - \widehat{\lambda}\widehat{x}$.

(2) If λ *is an eigenvalue of* A *that is closest to* $\widehat{\lambda}$, *and* $\delta := \min\{|\lambda_i - \widehat{\lambda}| \,:\, \lambda_i \in \sigma(A)\backslash\{\lambda\}\}$ *is the distance of* $\widehat{\lambda}$ *to the rest of the spectrum of* A, *then*

$$|\lambda - \widehat{\lambda}| \,\leq\, \frac{\|r\|_2^2}{\delta}. \tag{6.9}$$

(If $\widehat{\lambda}$ *is exactly in the middle between two eigenvalues of* A, *then* λ *can be any of these two eigenvalues, and* $\delta = |\lambda - \widehat{\lambda}|$.*)*

Proof. (1) Consider the matrix $\widetilde{A} := (A - \alpha I_n)(A - \beta I_n)$. Since A is Hermitian, the matrix $\widetilde{A}$ is Hermitian and its eigenvalues are

$$(\lambda_i - \alpha)(\lambda_i - \beta), \quad i = 1, \ldots, n,$$

where $\lambda_1, \ldots, \lambda_n \in \mathbb{R}$ are the eigenvalues of A. By assumption, for each $i = 1, \ldots, n$ we have

$$\text{either } \lambda_i \leq \alpha < \beta, \text{ or } \alpha < \beta \leq \lambda_i.$$

Thus, all eigenvalues of $\widetilde{A}$ are nonnegative, so that $\widetilde{A}$ is positive semidefinite. For the unit norm vector $\widehat{x}$ we now obtain, using $\widehat{x} \perp r$,

$$\begin{aligned}
0 &\leq \widehat{x}^H \widetilde{A} \widehat{x} \\
&= \widehat{x}^H (A - \alpha I_n)(A - \beta I_n)\widehat{x} \\
&= \widehat{x}^H \left(A - \widehat{\lambda} I_n + (\widehat{\lambda} - \alpha) I_n\right) \left(A - \widehat{\lambda} I_n + (\widehat{\lambda} - \beta) I_n\right) \widehat{x} \\
&= \left(r^H + (\widehat{\lambda} - \alpha)\widehat{x}^H\right) \left(r + (\widehat{\lambda} - \beta)\widehat{x}\right) \\
&= \|r\|_2^2 + (\widehat{\lambda} - \alpha)(\widehat{\lambda} - \beta),
\end{aligned}$$

which shows (6.8).
(2) The inequality is trivial if $\widehat{\lambda} = \lambda$. Suppose that $\widehat{\lambda} < \lambda$, where λ is an eigenvalue of A that is closest to $\widehat{\lambda}$. Let $\lambda_i \in \sigma(A) \setminus \{\lambda\}$ with $|\lambda_i - \widehat{\lambda}| = \delta$. Then we have

$$\text{either } \lambda_i < \widehat{\lambda} < \lambda, \text{ or } \widehat{\lambda} < \lambda < \lambda_i.$$

In the first case the interval (λ_i, λ) does not contain an eigenvalue of A, and (6.8) with $\alpha = \lambda_i$ and $\beta = \lambda$ yields

$$(\lambda - \widehat{\lambda}) \underbrace{(\widehat{\lambda} - \lambda_i)}_{=\delta} \leq \|r\|_2^2,$$

which shows (6.9). In the second case the interval $(\widehat{\lambda} - \delta, \lambda)$ does not contain an eigenvalue of A, and (6.8) with $\alpha = \widehat{\lambda} - \delta$ and $\beta = \lambda$ yields

$$(\lambda - \widehat{\lambda}) \underbrace{(\widehat{\lambda} - (\widehat{\lambda} - \delta))}_{=\delta} \leq \|r\|_2^2,$$

which again shows (6.9). The proof for $\lambda < \widehat{\lambda}$ is analogous. □

A vector $\widehat{x}$ is a close approximation of an exact eigenvector x of A when the angle between $\text{span}\{\widehat{x}\}$ and $\text{span}\{x\}$ is small. This means that $\sin\theta(\widehat{x}, x)$ should be small. The following result gives a bound on $\sin\theta(\widehat{x}, x)$ for a normal matrix A.

Theorem 6.8. *Let $A \in \mathbb{C}^{n\times n}$ be normal, let $\widehat{x} \in \mathbb{C}^n$ be a unit norm vector, and define $\widehat{\lambda} := R_A(\widehat{x})$. Let λ be an eigenvalue of A that is closest to $\widehat{\lambda}$, suppose that λ is a simple eigenvalue of A, and let x be a unit norm eigenvector of A corresponding to λ. If $\delta := \min\{|\lambda_i - \widehat{\lambda}| : \lambda_i \in \sigma(A) \setminus \{\lambda\}\}$ is the distance of $\widehat{\lambda}$ to the rest of the spectrum of A, then*

$$\sin\theta(\widehat{x}, x) \leq \frac{\|r\|_2}{\delta}, \tag{6.10}$$

where $r = A\widehat{x} - \widehat{\lambda}\widehat{x}$.

Proof. The normal matrix A is unitarily diagonalizable, $A = X\text{diag}(\lambda_1, \ldots, \lambda_n)X^H$, with a unitary matrix $X = [x_1, \ldots, x_n] \in \mathbb{C}^{n\times n}$. Without loss of generality, we can consider that $\lambda = \lambda_1$ and $x_1 = x$. Expanding $\widehat{x}$ in the orthonormal basis $x_1, x_2, \ldots, x_n$ we have

$$\widehat{x} = \sum_{i=1}^{n} (x_i^H \widehat{x}) x_i = cx_1 + z,$$

where

$$c := x_1^H \widehat{x}, \quad z := \sum_{i=2}^{n} (x_i^H \widehat{x}) x_i.$$

Note that $|c| = |x_1^H \widehat{x}| = \cos\theta(\widehat{x}, x)$; see (6.3). By construction, $x_1 \perp z$ and hence $1 = \|\widehat{x}\|_2^2 = |c|^2 + \|z\|_2^2$, so that $\|z\|_2^2 = 1 - |c|^2 = \sin^2\theta(\widehat{x}, x)$.
The residual can be written as

$$r = (A - \widehat{\lambda}I_n)\widehat{x} = (A - \widehat{\lambda}I_n)(cx_1 + z) = c(\lambda - \widehat{\lambda})x_1 + (A - \widehat{\lambda}I_n)z.$$

Since $x_1^H(A - \widehat{\lambda}I_n)z = \lambda x_1^H z - \widehat{\lambda}x_1^H z = 0$, i.e., $x_1 \perp (A - \widehat{\lambda}I_n)z$, we obtain

$$\begin{aligned}
\|r\|_2^2 &= |c|^2 \, |\lambda - \widehat{\lambda}|^2 + \|(A - \widehat{\lambda}I_n)z\|_2^2 \\
&= |c|^2 \, |\lambda - \widehat{\lambda}|^2 + \left\| (A - \widehat{\lambda}I_n) \sum_{i=2}^{n} (x_i^H \widehat{x}) x_i \right\|_2^2 \\
&\geq \left\| \sum_{i=2}^{n} (x_i^H \widehat{x})(\lambda_i - \widehat{\lambda}) x_i \right\|_2^2 \\
&= \sum_{i=2}^{n} |x_i^H \widehat{x}|^2 \underbrace{|\lambda_i - \widehat{\lambda}|^2}_{\geq \delta^2} \\
&\geq \delta^2 \sum_{i=2}^{n} |x_i^H \widehat{x}|^2 \\
&= \delta^2 \|z\|_2^2.
\end{aligned}$$

Rearranging and using $\|z\|_2^2 = \sin^2\theta(\widehat{x}, x)$ shows the bound (6.10). □

For a Hermitian matrix A the bound of Theorem 6.8 can be compared to the one of Theorem 6.7 (2). We observe that the bound (6.9) on $|\widehat{\lambda} - \lambda|$ contains the quadratic factor $\|r\|_2^2$, while bound (6.10) on $\sin\theta(\widehat{x}, x)$ only contains the linear factor $\|r\|_2$. This indicates that (when the residual norm is small) the (forward) error in the eigenvalue approximation may be significantly smaller than the (forward) error in the eigenvector approximation.

6.3 Backward error perspective

Let $A \in \mathbb{C}^{n\times n}$ and an approximate eigenpair $(\widehat{\lambda}, \widehat{x})$ with $\|\widehat{x}\|_2 = 1$ be given. We can then ask about a minimum norm perturbation of A, so that the approximate eigenpair is an exact eigenpair of the perturbed matrix, i.e., about the *normwise backward error*

$$\eta(\widehat{\lambda}, \widehat{x}) := \min\left\{\|\Delta A\|_2 \,:\, (A + \Delta A)\widehat{x} = \widehat{\lambda}\widehat{x}\right\}. \tag{6.11}$$

Using the residual $r = A\widehat{x} - \widehat{\lambda}\widehat{x}$ we define the matrix $E := -r\widehat{x}^H$, which satisfies $\|E\|_2 = \|r\|_2$. Moreover,

$$(A + E)\widehat{x} = A\widehat{x} - r\widehat{x}^H\widehat{x} = \widehat{\lambda}\widehat{x},$$

and if ΔA is any matrix with $(A + \Delta A)\widehat{x} = \widehat{\lambda}\widehat{x}$, then $r = -\Delta A\widehat{x}$, which yields

$$\|r\|_2 \le \|\Delta A\|_2\|\widehat{x}\|_2 = \|\Delta A\|_2.$$

We thus have shown the following result, which is analogous to the residual-based backward error bound for linear algebraic systems in Theorem 2.19.

Theorem 6.9. *For a matrix $A \in \mathbb{C}^{n\times n}$ and an approximate eigenpair $(\widehat{\lambda}, \widehat{x})$ with $\|\widehat{x}\|_2 = 1$ the normwise backward error* (6.11) *is*

$$\eta(\widehat{\lambda}, \widehat{x}) = \|r\|_2,$$

and the minimum in (6.11) *is attaind by the matrix $-r\widehat{x}^H$.*

In particular, if A is diagonalizable we can use $\|E\|_2 = \|r\|_2$ in (6.5) and get

$$|\lambda - \widehat{\lambda}| \le \kappa_2(X)\|E\|_2, \tag{6.12}$$

where λ is the eigenvalue of A that is closest to $\widehat{\lambda}$. This inequality is another example of our rule of thumb

$$\text{forward error} \;\le\; \text{condition number} \;\times\; \text{backward error};$$

also cf. (6.1) for a similar bound using the condition number of an individual (simple) eigenvalue rather than the condition number of the eigenvector matrix X.

Now let $\lambda_1, \ldots, \lambda_n$ be the eigenvalues of the diagonalizable matrix $A \in \mathbb{C}^{n\times n}$. Let $\Delta A \in \mathbb{C}^{n\times n}$ be another given matrix, and denote the eigenvalues of the perturbed matrix $A + \Delta A$

by $\widehat{\lambda}_1, \ldots, \widehat{\lambda}_n$. (The perturbed matrix need not be diagonalizable.)
If $\widehat{x}_j$ is a unit norm eigenvector corresponding to $\widehat{\lambda}_j$, i.e., $(A + \Delta A)\widehat{x}_j = \widehat{\lambda}_j \widehat{x}_j$, then also $(A + E_j)\widehat{x}_j = \widehat{\lambda}_j \widehat{x}_j$ for $E_j := -r_j \widehat{x}_j^H$, where $r_j := A\widehat{x}_j - \widehat{\lambda}_j \widehat{x}_j$. The minimum norm property of the perturbation matrix E_j implies that $\|E_j\|_2 \leq \|\Delta A\|_2$, and (6.12) yields

$$\min_{1 \leq i \leq n} |\lambda_i - \widehat{\lambda}_j| \ \leq \ \kappa_2(X) \|E_j\|_2 \leq \kappa_2(X) \|\Delta A\|_2.$$

Maximizing the left hand side over j gives the following result.

Theorem 6.10 (Bauer & Fike [3]). *Let $A \in \mathbb{C}^{n \times n}$ be diagonalizable with the eigenvalues $\lambda_1, \ldots, \lambda_n$. If $\Delta A \in \mathbb{C}^{n \times n}$ and the eigenvalues of $A + \Delta A$ are $\widehat{\lambda}_1, \ldots, \widehat{\lambda}_n$, then*

$$\max_{1 \leq j \leq n} \min_{1 \leq i \leq n} |\lambda_i - \widehat{\lambda}_j| \ \leq \ \kappa_2(X) \|\Delta A\|_2.$$

This result motivates the following definition.

Definition 6.11. *Let $A \in \mathbb{C}^{n \times n}$ and $B \in \mathbb{C}^{n \times n}$ have the eigenvalues $\lambda_1, \ldots, \lambda_n$ and $\beta_1, \ldots, \beta_n$, respectively. The* spectral variation of B with respect to A *is defined as*

$$\mathrm{sv}(A, B) := \max_{1 \leq j \leq n} \min_{1 \leq i \leq n} |\lambda_i - \beta_j|.$$

Thus, for a diagonalizable matrix $A = X \Lambda X^{-1} \in \mathbb{C}^{n \times n}$ and *any* matrix $B \in \mathbb{C}^{n \times n}$ the bound in Theorem 6.10 can be written as

$$\mathrm{sv}(A, B) \leq \kappa_2(X) \, \|A - B\|_2.$$

If A is normal, then X can be chosen unitary and the bound becomes

$$\mathrm{sv}(A, B) \leq \|A - B\|_2. \tag{6.13}$$

It should be pointed out that the spectral variation is not symmetric, i.e., that we may have

$$\mathrm{sv}(A, B) \ \neq \ \mathrm{sv}(B, A).$$

As a simple example consider $\lambda_1 = 1$, $\lambda_2 = 2$, and $\beta_1 = \frac{3}{2}$, $\beta_2 = 3$. Then

$$\min_{1 \leq i \leq 2} |\lambda_i - \beta_1| = \frac{1}{2}, \quad \min_{1 \leq i \leq 2} |\lambda_i - \beta_2| = 1,$$
$$\min_{1 \leq j \leq 2} |\lambda_1 - \beta_j| = \frac{1}{2}, \quad \min_{1 \leq j \leq 2} |\lambda_2 - \beta_j| = \frac{1}{2},$$

and thus

$$\max_{1 \leq j \leq 2} \min_{1 \leq i \leq 2} |\lambda_i - \beta_j| = \mathrm{sv}(A, B) = 1,$$
$$\max_{1 \leq i \leq 2} \min_{1 \leq j \leq 2} |\lambda_i - \beta_j| = \mathrm{sv}(B, A) = \frac{1}{2}.$$

In order to prove a bound on the spectral variation of two general matrices A and B we need the *Hadamard inequality* for the determinant of a matrix.

Lemma 6.12. *If $A = [a_1, \ldots, a_n] \in \mathbb{C}^{n\times n}$ with $a_i \in \mathbb{C}^n$, $i = 1, \ldots, n$, then*

$$|\det(A)| \le \prod_{i=1}^{n} \|a_i\|_2. \tag{6.14}$$

Proof. Let $A = QR$ with a unitary matrix Q and an upper triangular matrix $R = [r_{ij}]$ be a QR decomposition of A. If $r_i = Q^H a_i$ is the ith column of R, then $|r_{ii}| \le \|r_i\|_2 = \|a_i\|_2$, and hence

$$|\det(A)| = |\det(QR)| = \underbrace{|\det(Q)|}_{=1} |\det(R)| = \prod_{i=1}^{n} |r_{ii}| \le \prod_{i=1}^{n} \|r_i\|_2 = \prod_{i=1}^{n} \|a_i\|_2.$$

□

Theorem 6.13 (Elsner [9])**.** *For any $A, B \in \mathbb{C}^{n\times n}$ we have*

$$\mathrm{sv}(A, B) \le (\|A\|_2 + \|B\|_2)^{(n-1)/n} \, \|A - B\|_2^{1/n}. \tag{6.15}$$

Proof. Let $\lambda_1, \ldots, \lambda_n$ and $\beta_1, \ldots, \beta_n$ be the eigenvalues of A and B, respectively.
Suppose that the maximum in the spectral variation is attained for the eigenvalue $\beta \in \{\beta_1, \ldots, \beta_n\}$, i.e,

$$\mathrm{sv}(A, B) = \min_{1\le i\le n} |\lambda_i - \beta|.$$

Let $\widehat{x}_1, \ldots, \widehat{x}_n$ be orthonormal vectors with $B\widehat{x}_1 = \beta\widehat{x}_1$. Using the Hadamard inequality we then get

$$\begin{aligned}
\mathrm{sv}(A, B)^n &= \min_{1\le i\le n} |\lambda_i - \beta|^n \le \prod_{i=1}^{n} |\lambda_i - \beta| \\
&= |\det(A - \beta I_n)| = |\det\left((A - \beta I_n)[\widehat{x}_1, \ldots, \widehat{x}_n]\right)| \\
&\le \prod_{i=1}^{n} \|(A - \beta I_n)\widehat{x}_i\|_2 = \|(A - \beta I_n)\widehat{x}_1\|_2 \prod_{i=2}^{n} \|(A - \beta I_n)\widehat{x}_i\|_2 \\
&= \|(A - B)\widehat{x}_1\|_2 \prod_{i=2}^{n} \|(A - \beta I_n)\widehat{x}_i\|_2 \\
&\le \|A - B\|_2 \, (\|A\|_2 + \|B\|_2)^{n-1},
\end{aligned}$$

where in the last inequality we used that $\|(A - \beta I_n)\widehat{x}_i\|_2 \le \|A\|_2 + |\beta| \le \|A\|_2 + \|B\|_2$. (Recall from Lemma 2.12 that the spectral radius of a square matrix M satisfies $\rho(M) \le \|M\|$ for any consistent norm.) Taking the nth root on both sides of the inequality above yields (6.15). □

The symmetric function

$$\mathrm{hd}(A, B) := \max\{\mathrm{sv}(A, B), \mathrm{sv}(B, A)\}$$

is called the *Hausdorff distance* between A and B. The next result follows immediately from the bound (6.13).

Corollary 6.14. *If $A, B \in \mathbb{C}^{n\times n}$ are normal, then* $\mathrm{hd}(A, B) \leq \|A - B\|_2$.

Moreover, since the right hand side of the inequality (6.15) is symmetric in A and B, we get the following corollary of Theorem 6.13.

Corollary 6.15. *For any $A, B \in \mathbb{C}^{n\times n}$ we have*

$$\mathrm{hd}(A, B) \leq (\|A\|_2 + \|B\|_2)^{(n-1)/n} \, \|A - B\|_2^{1/n}.$$

Chapter 7

Power iterations for solving eigenvalue problems

According to the classical Abel-Ruffini Theorem, there exists no algebraic solution to general polynomial equations $p(z) = 0$ when the degree of p is larger than 4. Since the eigenvalues of a matrix are the zeros of its characteristic polynomial, this fundamental theorem implies that no numerical method for solving eigenvalue problems

$$Ax = \lambda x, \quad A \in \mathbb{C}^{n \times n},$$

can be "direct" in the sense that it is guaranteed to give the exact solution after finitely many steps. Consequently, numerical methods for solving eigenvalue problems are necessarily of iterative character. In this chapter we will study methods that are based on so-called power iterations.

7.1 The power method

We start with the *power method*, which is the most basic method for computing eigenvector and eigenvalue approximations:

Algorithm 13 Power method

Input: $A \in \mathbb{C}^{n \times n}$, unit norm vector $q^{(0)} \in \mathbb{C}^n$, stopping criterion, maximal number of iterations $n_{\max}$
Output: Approximate eigenpair $(\lambda^{(k)}, q^{(k)})$ of A
for $k = 1, \ldots, n_{\max}$ **do**
  $\widehat{q}^{(k)} = Aq^{(k-1)}$
  $q^{(k)} = \widehat{q}^{(k)} / \|\widehat{q}^{(k)}\|_2$
  $\lambda^{(k)} = R_A(q^{(k)})$
  Test the approximate eigenpair for convergence and stop if satisfied
end for

From the backward error analysis (see Section 6.3) we know that each step of the power method generates an approximate eigenpair $(\lambda^{(k)}, q^{(k)})$ of A, which is an exact eigenpair of the perturbed matrix $A + E_k$ where

$$E_k := -r^{(k)}(q^{(k)})^H \quad \text{and} \quad r^{(k)} = Aq^{(k)} - \lambda^{(k)}q^{(k)}.$$

The residual norm $\|r^{(k)}\|_2 = \|E_k\|_2$ can be used as a stopping criterion for the iteration, but it needs to be kept in mind that a small residual norm does not necessarily imply a small forward error in the eigenvalue approximation (see Examples 6.5 and 6.6).
By induction we see that

$$\begin{aligned}\|A^k q^{(0)}\|_2 &= \|A^{k-1}(Aq^{(0)})\|_2 = \|A^{k-1}\widehat{q}^{(1)}\|_2 = \|\widehat{q}^{(1)}\|_2\|A^{k-1}q^{(1)}\|_2 = \|\widehat{q}^{(1)}\|_2\|A^{k-2}\widehat{q}^{(2)}\|_2 \\ &= \|\widehat{q}^{(1)}\|_2\|\widehat{q}^{(2)}\|_2\|A^{k-2}q^{(2)}\|_2 = \cdots = \|\widehat{q}^{(1)}\|_2\|\widehat{q}^{(2)}\|_2\cdots\|\widehat{q}^{(k)}\|_2,\end{aligned}$$

and

$$\begin{aligned}q^{(k)} &= \frac{\widehat{q}^{(k)}}{\|\widehat{q}^{(k)}\|_2} = \frac{Aq^{(k-1)}}{\|\widehat{q}^{(k)}\|_2} = \frac{A\widehat{q}^{(k-1)}}{\|\widehat{q}^{(k)}\|_2\|\widehat{q}^{(k-1)}\|_2} = \frac{A^2q^{(k-2)}}{\|\widehat{q}^{(k)}\|_2\|\widehat{q}^{(k-1)}\|_2} = \cdots \\ &= \frac{A^k q^{(0)}}{\|\widehat{q}^{(k)}\|_2\|\widehat{q}^{(k-1)}\|_2\cdots\|\widehat{q}^{(1)}\|_2} = \frac{A^k q^{(0)}}{\|A^k q^{(0)}\|_2}.\end{aligned}$$

In order to analyze the convergence behavior of this method we assume, for simplicity, that A is diagonalizable, $A = X\,\mathrm{diag}(\lambda_1, \ldots, \lambda_n)X^{-1}$. Furthermore, we assume that for some $j \in \{1, \ldots, n-1\}$ we have

$$\lambda_1 = \cdots = \lambda_j \quad \text{and} \quad |\lambda_j| > |\lambda_{j+1}| \geq \cdots \geq |\lambda_n|.$$

Let $q^{(0)} = X[\beta_1, \ldots, \beta_n]^T$, where at least one of the coefficients $\beta_1, \ldots, \beta_j$ is nonzero. Then

$$\begin{aligned}A^k q^{(0)} &= X\mathrm{diag}(\lambda_1^k, \ldots, \lambda_n^k)X^{-1}X[\beta_1, \ldots, \beta_n]^T = \sum_{i=1}^{n}\beta_i\lambda_i^k x_i \\ &= \lambda_1^k\left(\sum_{i=1}^{j}\beta_i x_i + \sum_{i=j+1}^{n}\beta_i\left(\frac{\lambda_i}{\lambda_1}\right)^k x_i\right) =: \lambda_1^k(\widetilde{x}_1 + y^{(k)}).\end{aligned}$$

We now have $\widetilde{x}_1 \neq 0$ by our assumption on $\beta_1, \ldots, \beta_j$. Clearly, $\widetilde{x}_1$ is an eigenvector of A corresponding to the eigenvalue λ_1.
Moreover, $y^{(k)} \to 0$ for $k \to \infty$ since $|\lambda_i/\lambda_1| < 1$ for $i = j+1, \ldots, n$, and hence

$$q^{(k)} = \frac{A^k q^{(0)}}{\|A^k q^{(0)}\|_2} = \frac{\widetilde{x}_1 + y^{(k)}}{\|x_1 + y^{(k)}\|_2} \to \frac{\widetilde{x}_1}{\|\widetilde{x}_1\|_2} \quad \text{for } k \to \infty,$$

where the rate of convergence depends on how quickly $y^{(k)}$ approaches zero, and hence, in general, on the ratio $|\lambda_{j+1}/\lambda_1|$. Since $q^{(k)}$ converges to an eigenvector of A corresponding to λ_1, we also see that $\lambda^{(k)} \to \lambda_1$ for $k \to \infty$.
For Hermitian matrices we can prove a more refined result, where for simplicity we assume that $j = 1$.

Theorem 7.1. *Suppose that $A \in \mathbb{C}^{n\times n}$ is Hermitian with the unitary eigendecomposition $A = X\,\mathrm{diag}(\lambda_1,\ldots,\lambda_n)X^H$ and (real) eigenvalues that satisfy $|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|$. Let $q^{(0)} \in \mathbb{C}^n$ be a unit norm vector with $|x_1^H q^{(0)}| = \cos\theta(q^{(0)}, x_1) \neq 0$. Then the power method applied to A and $q^{(0)}$ yields eigenvector and eigenvalue approximations, such that*

$$\sin\theta(q^{(k)}, x_1) \leq \frac{\sin\theta(q^{(0)}, x_1)}{\cos\theta(q^{(0)}, x_1)} \left|\frac{\lambda_2}{\lambda_1}\right|^k, \tag{7.1}$$

$$|\lambda_1 - \lambda^{(k)}| \leq \max_{2\leq j\leq n} |\lambda_1 - \lambda_j| \frac{\sin^2\theta(q^{(0)}, x_1)}{\cos^2\theta(q^{(0)}, x_1)} \left|\frac{\lambda_2}{\lambda_1}\right|^{2k}. \tag{7.2}$$

Proof. Let us write $q^{(0)} = X[\beta_1,\ldots,\beta_n]^T$, where by assumption

$$|\beta_1| = |x_1^H q^{(0)}| = \cos\theta(q^{(0)}, x_1) \neq 0.$$

Since the eigenvectors $x_1,\ldots,x_n$ are orthonormal, we have $1 = \|q^{(0)}\|_2^2 = \sum_{j=1}^n |\beta_j|^2$, and

$$\|A^k q^{(0)}\|_2^2 = \Big\|\sum_{j=1}^n \beta_j \lambda_j^k x_j\Big\|_2^2 = \sum_{j=1}^n |\beta_j|^2 \lambda_j^{2k},$$

where we have used that the eigenvalues are real. We then obtain

$$\begin{aligned}
\sin^2\theta(q^{(k)}, x_1) &= 1 - \cos^2\theta(q^{(k)}, x_1) = 1 - |x_1^H q^{(k)}|^2 = 1 - \frac{|x_1^H A^k q^{(0)}|^2}{\|A^k q^{(0)}\|_2^2} \\
&= 1 - \frac{|\beta_1|^2\lambda_1^{2k}}{\sum_{j=1}^n |\beta_j|^2\lambda_j^{2k}} = \frac{\sum_{j=2}^n |\beta_j|^2\lambda_j^{2k}}{\sum_{j=1}^n |\beta_j|^2\lambda_j^{2k}} \\
&\leq \frac{\sum_{j=2}^n |\beta_j|^2\lambda_j^{2k}}{|\beta_1|^2\lambda_1^{2k}} = \frac{1}{|\beta_1|^2}\sum_{j=2}^n |\beta_j|^2 \left|\frac{\lambda_j}{\lambda_1}\right|^{2k} \\
&\leq \frac{1}{|\beta_1|^2}\left(\sum_{j=2}^n |\beta_j|^2\right)\left|\frac{\lambda_2}{\lambda_1}\right|^{2k} = \frac{1-|\beta_1|^2}{|\beta_1|^2}\left|\frac{\lambda_2}{\lambda_1}\right|^{2k} \\
&= \frac{\sin^2\theta(q^{(0)}, x_1)}{\cos^2\theta(q^{(0)}, x_1)}\left|\frac{\lambda_2}{\lambda_1}\right|^{2k},
\end{aligned}$$

which shows (7.1). For the bound on the eigenvalue approximation we observe that, since A is Hermitian,

$$\begin{aligned}
\lambda^{(k)} &= (q^{(k)})^H A q^{(k)} = \frac{(q^{(0)})^H A^{2k+1} q^{(0)}}{\|A^k q^{(0)}\|_2^2} = \frac{(q^{(0)})^H A^{2k+1} q^{(0)}}{(q^{(0)})^H A^{2k} q^{(0)}} \\
&= \frac{\sum_{j=1}^n |\beta_j|^2\lambda_j^{2k+1}}{\sum_{j=1}^n |\beta_j|^2\lambda_j^{2k}}.
\end{aligned}$$

From this we obtain

$$|\lambda_1 - \lambda^{(k)}| = \left| \frac{\sum_{j=2}^n |\beta_j|^2 \lambda_j^{2k} (\lambda_1 - \lambda_j)}{\sum_{j=1}^n |\beta_j|^2 \lambda_j^{2k}} \right| \leq \max_{2 \leq j \leq n} |\lambda_1 - \lambda_j| \, \frac{\sum_{j=2}^n |\beta_j|^2 \lambda_j^{2k}}{\sum_{j=1}^n |\beta_j|^2 \lambda_j^{2k}}$$
$$\leq \max_{2 \leq j \leq n} |\lambda_1 - \lambda_j| \, \frac{\sin^2 \theta(q^{(0)}, x_1)}{\cos^2 \theta(q^{(0)}, x_1)} \left| \frac{\lambda_2}{\lambda_1} \right|^{2k},$$

where in the last inequality we have used the same argument as above. □

Example 7.2. *Suppose that*

$$A = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad \text{and} \quad q^{(0)} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix},$$

then the power method yields

$$\widehat{q}^{(1)} = Aq^{(0)} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = q^{(1)},$$
$$\widehat{q}^{(2)} = Aq^{(1)} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = q^{(0)}.$$

Hence, by induction, $q^{(k)} = q^{(k+2)}$ for all $k \in \mathbb{N}$. This illustrates why the power method may not converge when A has several distinct eigenvalues of the same maximal modulus.

As observed already in our comparison of Theorem 6.7 and Theorem 6.8, the bound (7.2) on the forward error in the eigenvalue is essentially the square of the bound (7.1) on the error in the eigenvector.
For each $A \in \mathbb{C}^{n \times n}$ we have

$$\|A\|_2 = \sigma_{\max}(A) = (\lambda_{\max}(A^H A))^{1/2}.$$

Applying the power method to the Hermitian positive semidefinite matrix $A^H A$ therefore yields an estimate of $\|A\|_2$. This approach is implemented in many numerical software packages[1].

Example 7.3. *We define $L_n := \text{tridiag}(-1, 2, -1) \in \mathbb{R}^{n \times n}$. The dashed lines in Figure 7.1 show the absolute error $\lambda_1 - \lambda^{(k)}$, $k = 1, 2, \ldots, 500$, in the power method for*

[1]For example, the documentation of `normest` in MATLAB R2023a reads: "n = normest(S) returns an estimate of the 2-norm of the matrix S. [...] The power iteration involves repeated multiplication by the matrix S and its transpose, S'. The iteration is carried out until two successive estimates agree to within the specified relative tolerance."

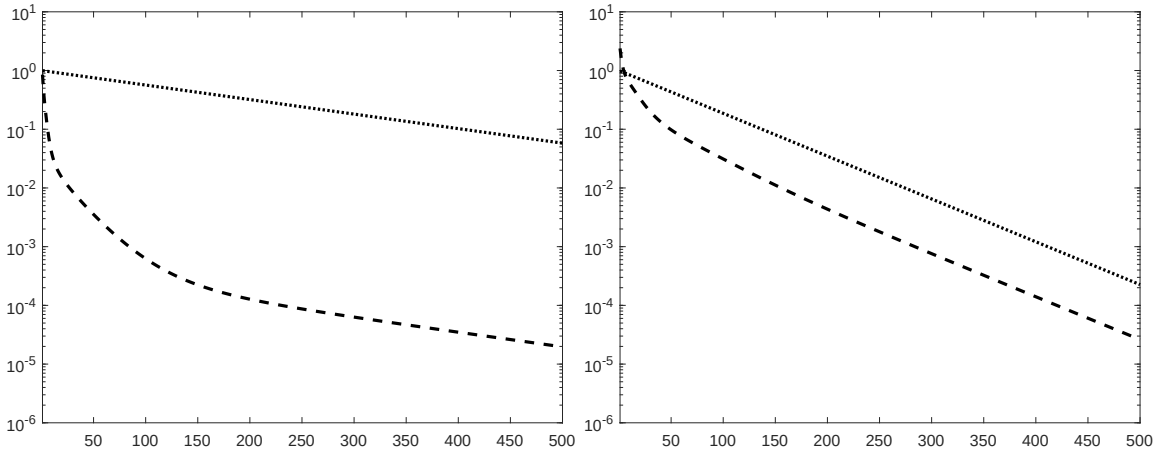


Figure 7.1: Absolute error $\lambda_1 - \lambda^{(k)}$ in the power method (dashed) and the values $|\lambda_2/\lambda_1|^{2k}$ for the two different matrices in Example 7.3.

the SPD matrices

$$A_1 = L_{50} \quad \textit{(left)} \quad \textit{and} \quad A_2 = L_{20} \otimes I_{20} + I_{20} \otimes L_{20} \in \mathbb{R}^{400\times 400} \quad \textit{(right)},$$

and a unit norm random initial vector generated with `randn` *in MATLAB. The dotted lines show the values* $|\lambda_2/\lambda_1|^{2k}$*. We observe that after an initial phase of about 50 steps, where the error decreases relatively quickly, the method enters a linear phase of convergence. For the matrix* A_2 *the linear rate of convergence is (approximately) predicted by the ratio* $|\lambda_2/\lambda_1|$*; cf. the bound* (7.2).

7.2 Inverse iteration and Rayleigh quotient iteration

If the assumption of Theorem 7.1 hold, then the power method converges towards an eigenvector corresponding to the largest eigenvalue λ_1 of a Hermitian matrix A. Suppose that A is nonsingular, i.e., $\lambda_n \neq 0$ in Theorem 7.1. Then λ_n^{-1} is an eigenvalue with largest modulus of A^{-1}. If that eigenvalue is simple, then the power method applied to A^{-1} will converge to λ_n^{-1} and a corresponding eigenvector of A^{-1}, which also is an eigenvector of A corresponding to the smallest eigenvalue (in modulus). This algorithm is called *inverse iteration*.

We can easily extend this idea by introducing a shift parameter $\sigma \in \mathbb{C}$, and by applying the power method to $(A - \sigma I_n)^{-1}$, where $A \in \mathbb{C}^{n\times n}$ is any given matrix. If $\lambda_1, \ldots, \lambda_n$ are the eigenvalues of A, then the eigenvalues of $(A-\sigma I_n)^{-1}$ are given by $(\lambda_1-\sigma)^{-1}, \ldots, (\lambda_n-\sigma)^{-1}$, and their largest modulus is

$$\left(\min_{1\leq j\leq n} |\lambda_j - \sigma| \right)^{-1}.$$

Consequently, if this method converges, then we obtain (an approximation of) an eigenvalue of A that is closest to σ, and an approximation of a corresponding eigenvector. The *shifted inverse iteration* is stated in Algorithm 14, where we (of course) implement the application of $(A - \sigma I_n)^{-1}$ as a linear algebraic system solve with $A - \sigma I_n$. Since this has to be done in every step, it can be beneficial to compute a decomposition of this matrix upfront (e.g. LU or Cholesky decomposition), and then use the factors of the decomposition in every step.

Algorithm 14 Shifted inverse iteration

Input: $A \in \mathbb{C}^{n\times n}$, unit norm vector $q^{(0)} \in \mathbb{C}^n$, shift parameter $\sigma \in \mathbb{C}$, stopping criterion, maximal number of iterations $n_{\max}$
Output: Approximate eigenpair $(\lambda^{(k)}, q^{(k)})$ of A
for $k = 1, \ldots, n_{\max}$ **do**
  Solve $(A - \sigma I_n)\widehat{q}^{(k)} = q^{(k-1)}$ for $\widehat{q}^{(k)}$
  $q^{(k)} = \widehat{q}^{(k)}/\|\widehat{q}^{(k)}\|_2$
  $\lambda^{(k)} = R_A(q^{(k)})$
  Test the approximate eigenpair $(\lambda^{(k)}, q^{(k)})$ for convergence and stop if satisfied
end for

From the convergence theory for the power method we expect that Algorithm 14 will converge linearly. A straightforward idea for obtaining a faster converging method is to vary the shift parameter in every step. This gives rise to the *Rayleigh quotient iteration*, where the shift parameter in every step is the Rayleigh quotient of the current eigenvector approximation.

Algorithm 15 Rayleigh quotient iteration

Input: $A \in \mathbb{C}^{n\times n}$, unit norm vector $q^{(0)} \in \mathbb{C}^n$, stopping criterion, maximal number of iterations $n_{\max}$
Output: Approximate eigenpair $(\rho_{k-1}, q^{(k-1)})$ of A
for $k = 1, \ldots, n_{\max}$ **do**
  $\rho_{k-1} = R_A(q^{(k-1)})$
  Test the approximate eigenpair $(\rho_{k-1}, q^{(k-1)})$ for convergence and stop if satisfied
  Solve $(A - \rho_{k-1} I_n)\widehat{q}^{(k)} = q^{(k-1)}$ for $\widehat{q}^{(k)}$
  $q^{(k)} = \widehat{q}^{(k)}/\|\widehat{q}^{(k)}\|_2$
end for

We now prove that for symmetric matrices the 2-norms of the residual vectors

$$r_k := (A - \rho_k I_n)q^{(k)}$$

in Algorithm 15 decrease monotonically.

Theorem 7.4. *If $A = A^T \in \mathbb{R}^{n\times n}$, then for every initial vector $q^{(0)} \in \mathbb{R}^n$ we have*

$$\|r_k\|_2 \leq \|r_{k-1}\|_2, \quad k = 1, 2, \ldots,$$

in Algorithm 15.

Proof. We first prove a minimality property of the Rayleigh quotient. Let $0 \neq y \in \mathbb{R}^n$, then for every $\alpha \in \mathbb{R}$ we have

$$\|(A - \alpha I_n)y\|_2^2 = y^T(A - \alpha I_n)^T(A - \alpha I_n)y = y^T A^T A y - 2\alpha y^T A y + \alpha^2 y^T y.$$

Taking the derivative of this expression with respect to α shows that $\|(A - \alpha I_n)y\|_2$ is minimal for

$$\alpha_* := \frac{y^T A y}{y^T y} = R_A(y).$$

According to Algorithm 15 we have

$$q^{(k-1)} = (A - \rho_{k-1} I_n)\widehat{q}^{(k)} = \|\widehat{q}^{(k)}\|_2 (A - \rho_{k-1} I_n) q^{(k)}.$$

Thus, the unit norm vector $q^{(k-1)}$ is a scalar multiple of $(A - \rho_{k-1} I_n)q^{(k)}$. In general, if $y = \alpha z$ with $\|y\|_2 = 1$ and hence $|\alpha| = 1/\|z\|_2$, then $|y^T z| = |\alpha| \|z\|_2^2 = \|z\|_2$. We will use this in the step $(*)$ below.
The residual vectors in Algorithm 15 satisfy

$$\begin{aligned}
\|r_k\|_2 &= \|(A - \rho_k I_n)q^{(k)}\|_2 \\
&\leq \|(A - \rho_{k-1} I_n)q^{(k)}\|_2 \quad \text{(by the minimalty property of } \rho_k) \\
&= |(q^{(k-1)})^T (A - \rho_{k-1} I_n) q^{(k)}| \quad (*) \\
&= |r_{k-1}^T q^{(k)}| \\
&\leq \|r_{k-1}\|_2 \|q^{(k)}\|_2 \quad \text{(by the Cauchy–Schwarz inequality)} \\
&= \|r_{k-1}\|_2,
\end{aligned}$$

which completes the proof. □

The proof of Theorem 7.4 shows that

$$\|r_k\|_2 = \|r_{k-1}\|_2$$

holds in Algorithm 15 if and only if

$$\rho_k = \rho_{k-1} \quad \text{and} \quad r_{k-1} = \beta q^{(k)} \ \text{ for some } \beta \in \mathbb{R}.$$

The equation $r_{k-1} = \beta q^{(k)}$ is equivalent with

$$(A - \rho_{k-1} I_n)^2 q^{(k)} = \frac{\beta}{\|\widehat{q}^{(k)}\|_2} q^{(k)},$$

which means that $q^{(k)}$ is an eigenvector of $(A - \rho_{k-1} I_n)^2$. A detailed convergence analysis of the Rayleigh quotient iteration, which shows that the algorithm usually converges cubically, is given in [31, Sections 4.6–4.9].

7.3 Orthogonal iteration and the QR algorithm

We can obtain a straightforward generalization of the power method (Algorithm 13) by considering an initial matrix with orthonormal columns instead of just a unit norm initial vector. The normalization step in the method is then replaced by a QR factorization, which gives the *orthogonal iteration*:

Algorithm 16 Orthogonal iteration

Input: $A \in \mathbb{C}^{n\times n}$ and $Q^{(0)} \in \mathbb{C}^{n\times r}$ with orthonormal columns, stopping criterion, maximal number of iterations $n_{\max}$
Output: r approximate eigenvalues of A
for $k = 1, \ldots, n_{\max}$ **do**
  $\widehat{Q}^{(k)} = AQ^{(k-1)}$
  $Q^{(k)}R^{(k)} = \widehat{Q}^{(k)}$ (QR decomposition)
  Compute the eigenvalues of $(Q^{(k)})^H AQ^{(k)} \in \mathbb{C}^{r\times r}$ and stop if satisfied
end for

It is easy to see that the orthogonal iteration (Algorithm 16) for $r = 1$ is the power method (Algorithm 13). Under the assumption

$$|\lambda_1| \geq \cdots \geq |\lambda_r| > |\lambda_{r+1}| \geq \cdots \geq |\lambda_n|$$

one can show that $\mathrm{ran}(Q^{(k)})$ in the orthogonal iteration converges to the "dominant" invariant subspace corresponding to the r largest eigenvalues of A, and that the rate of convergence is determined by the ratio $|\lambda_{r+1}/\lambda_r|$.
A straightforward MATLAB implementation of Algorithm 16 with a random initial matrix $Q^{(0)}$ having orthonormal columns is as follows:

```
function Ahat = orth_iteration(A,r,maxit)
n = length(A);
[Q,R] = qr(randn(n,r),0);  % random Q^{(0)}
for k = 1:maxit
    Qhat = A*Q;
    [Q,R] = qr(Qhat);
end
Ahat = Q'*A*Q;
```

In the orthogonal iteration (Algorithm 16) we perform one matrix multiplication and one QR factorization per step. At the same cost and for $r = n$ we obtain the *QR algorithm*, which is one of the most important general purpose algorithms for eigenvalue computations:

Each step of the QR algorithm requires one QR decomposition and one multiplication of $n \times n$ matrices, which in general both cost $\mathcal{O}(n^3)$ operations.

Algorithm 17 QR algorithm

Input: $A \in \mathbb{C}^{n\times n}$, $Q^{(0)} \in \mathbb{C}^{n\times n}$ unitary, stopping criterion, maximal number of iterations $n_{\max}$
Output: Approximate Schur factor $A^{(k)}$ of A
Initialize: $A^{(0)} = (Q^{(0)})^H A Q^{(0)}$
for $k = 1, \ldots, n_{\max}$ **do**
  $Q^{(k)} R^{(k)} = A^{(k-1)}$ (QR decomposition)
  $A^{(k)} = R^{(k)} Q^{(k)}$
  Test $A^{(k)}$ for convergence and stop if satisfied
end for

Implementations of eigenvalue solvers based on the QR algorithm sometimes perform an initial step in which the given matrix A is transformed to upper Hessenberg from. This can be achieved, for example, using the Arnoldi algorithm; see Algorithm 6. We know from the implementation of GMRES in Section 4.5 that an upper Hessenberg matrix H_n can be transformed to upper triangular form using $n-1$ Givens rotations, which overall costs only $\mathcal{O}(n)$ operations. By construction, the product of these $n-1$ Givens rotations is itself a unitary (lower) Hessenberg matrix. Thus, if we start with $Q^{(0)} = I_n$ and $A^{(0)} = H_n$ in Algorithm 17, and compute the QR decomposition of $A^{(0)}$ with $n-1$ Givens rotations as described in Section 4.5, then $Q^{(1)}R^{(1)} = A^{(0)}$, where $Q^{(1)}$ is a unitary upper Hessenberg matrix, and $R^{(1)}$ is upper triangular. Consequently, also $A^{(1)} = R^{(1)}Q^{(1)}$ and hence inductively every matrix $A^{(k)}$ in Algorithm 17 will be in upper Hessenberg form, so that each QR decomposition costs only $\mathcal{O}(n)$ operations.

The matrices generated in the QR algorithm satisfy

$$\begin{aligned} A^{(k)} &= R^{(k)}Q^{(k)} = (Q^{(k)})^H (Q^{(k)}R^{(k)})Q^{(k)} = (Q^{(k)})^H A^{(k-1)} Q^{(k)} = \cdots \\ &= (Q^{(0)}Q^{(1)}\cdots Q^{(k)})^H A (Q^{(0)}Q^{(1)}\cdots Q^{(k)}). \end{aligned} \tag{7.3}$$

Thus, the algorithm generates a sequence of matrices $A^{(k)}$ which are unitarily similar to A.

If we take $Q^{(0)} = I_n$ in the orthogonal iteration, we obtain

$$\widehat{Q}^{(1)} = A = Q^{(1)}R^{(1)} \quad \text{(QR factorization)},$$

and therefore

$$(Q^{(1)})^H A Q^{(1)} = R^{(1)}Q^{(1)}.$$

On the other hand, if we take $Q^{(0)} = I_n$ in the QR algorithm, we obtain

$$Q^{(1)}R^{(1)} = A^{(0)} = A \quad \text{(QR factorization)}, \quad A^{(1)} = R^{(1)}Q^{(1)}.$$

Hence the matrix $A^{(1)}$ in the QR algorithm is equal to the matrix $(Q^{(1)})^H A Q^{(1)}$ in the orthogonal iteration. It can be shown by induction that each matrix $A^{(k)}$ in the QR algorithm is equal to $(Q^{(1)}\cdots Q^{(k)})^H A (Q^{(1)}\cdots Q^{(k)})$ in the orthogonal iteration, so that

the methods are mathematically equivalent for $r = n$ and $Q^{(0)} = I_n$. Consequently, the convergence properties of the orthogonal iteration can be used to analyze the sequence of matrices $A^{(k)}$ in the QR algorithm.

We will now show that for diagonalizable matrices having n eigenvalues with n distinct absolute values the sequence of the matrices $A^{(k)}$ in the QR algorithm converges to an upper triangular matrix which has the eigenvalues of A on its diagonal (i.e., a *Schur form* of A). Therefore the QR algorithm is a method for computing approximations to all eigenvalues of A simultaneously.
The proof of the following Theorem 7.5, which is adapted from [7, Theorem 6.9.1] to our notation and which was already shown by Francis [10], uses Theorem 1.4 and the fact that a unitary upper triangular matrix $Q \in \mathbb{C}^{n\times n}$ must be diagonal. (You can prove this as an exercise.)

Theorem 7.5. *Let $A \in \mathbb{C}^{n\times n}$ be diagonalizable, $A = X\Lambda X^{-1}$, with $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$ and $|\lambda_1| > |\lambda_2| > \cdots > |\lambda_n| > 0$. Then the matrices $A^{(k)} = [a_{ij}^{(k)}]$ generated by the QR algorithm with $Q^{(0)} = I_n$ satisfy*

$$\lim_{k\to\infty} a_{ij}^{(k)} = \begin{cases} 0, & i > j, \\ \lambda_i, & i = j. \end{cases}$$

Proof. We first show by induction that

$$A^k = P_k U_k \quad \text{for all } k \geq 1, \tag{7.4}$$

where

$$P_k := Q^{(1)} \cdots Q^{(k)} \quad \text{and} \quad U_k := R^{(k)} \cdots R^{(1)}.$$

For $k = 1$ we have

$$A = A^{(0)} = Q^{(1)} R^{(1)} = P_1 U_1.$$

Suppose that (7.4) holds for some $k \geq 1$. Noting that (7.3) can be written as

$$AP_k = P_k A^{(k)} \tag{7.5}$$

we obtain

$$A^{k+1} = AA^k = AP_k U_k = P_k A^{(k)} U_k = P_k Q^{(k+1)} R^{(k+1)} U_k = P_{k+1} U_{k+1},$$

which finishes the induction.
By construction, each matrix P_k is unitary, and each matrix U_k is upper triangular. We assume (for simplicity of notation and without loss of generality) that the QR factorizations in the QR algorithm are computed so that each matrix $R^{(k)}$ has positive diagonal entries. Then (7.4) is the uniquely determined QR factorization of A^k in the sense of Theorem 1.4.

Suppose that $X^{-1} = LU$ is the (uniquely determined) LU decomposition of X^{-1}, where $L = [l_{ij}]$ is unit lower triangular and U is nonsingular and upper triangular[2]. Then we can write

$$A^k = X\Lambda^k X^{-1} = X(\Lambda^k L\Lambda^{-k})(\Lambda^k U). \tag{7.6}$$

The matrix

$$\Lambda^k L\Lambda^{-k} = \left[l_{ij}(\lambda_i/\lambda_j)^k\right]$$

is lower triangular and has a unit diagonal. Moreover, since $|\lambda_i/\lambda_j| < 1$ for $i > j$, the entries of $\Lambda^k L\Lambda^{-k}$ below the diagonal converge to zero for $k \to \infty$. We therefore can write

$$\Lambda^k L\Lambda^{-k} = I_n + E_k,$$

where E_k is strictly lower triangular and satisfies

$$E_k \to 0 \quad \text{for } k \to \infty.$$

Let

$$X = \widehat{Q}\widehat{R}$$

be the (uniquely determined) QR decomposition of X, where $\widehat{Q}$ is unitary, and $\widehat{R}$ is upper triangular and has positive diagonal entries. Then (7.6) can be written as

$$A^k = \widehat{Q}\widehat{R}(I_n + E_k)(\Lambda^k U) = \widehat{Q}(I_n + \widehat{R}E_k\widehat{R}^{-1})(\widehat{R}\Lambda^k U). \tag{7.7}$$

The matrix $\widehat{R}E_k\widehat{R}^{-1}$ is similar to the strictly lower triangular matrix E_k, and hence has only eigenvalues 0. Therefore $I_n + \widehat{R}E_k\widehat{R}^{-1}$ has only eigenvalues 1, and in particular is nonsingular. Let

$$I_n + \widehat{R}E_k\widehat{R}^{-1} = \widetilde{Q}_k\widetilde{R}_k$$

be the (uniquely determined) QR decomposition of $I_n + \widehat{R}E_k\widehat{R}^{-1}$, where $\widetilde{Q}_k$ is unitary, and $\widetilde{R}_k$ is upper triangular and has positive diagonal entries. Then a multiplication with $\widetilde{Q}_k^H$ and a reordering yields

$$\widetilde{R}_k - \widetilde{Q}_k^H = \widetilde{Q}_k^H\widehat{R}E_k\widehat{R}^{-1} \to 0 \quad \text{for } k \to \infty.$$

Thus, the matrices in the sequence $\widetilde{Q}_k^H$ converge to upper triangular form. Since the matrices $\widetilde{Q}_k^H$ are unitary, they must converge to a diagonal matrix. Therefore also the matrices in the sequence $\widetilde{R}_k$ must converge to a diagonal matrix, which simultaneously must be unitary. Since each matrix $\widetilde{R}_k$ has positive diagonal entries we obtain

$$\widetilde{R}_k \to I_n \quad \text{and} \quad \widetilde{Q}_k \to I_n \quad \text{for } k \to \infty.$$

[2]In general this decomposition has the form $X^{-1} = PLU$ with a permutation matrix P, but here we assume for simplicity of notation and without loss of generality that the LU decomposition can be obtained without pivoting.

We now write

$$\Lambda = \operatorname{diag}(|\lambda_1|, \ldots, |\lambda_n|) \operatorname{diag}(\lambda_1/|\lambda_1|, \ldots, \lambda_n/|\lambda_n|) =: |\Lambda| \, D_\Lambda.$$

Note that the matrix D_Λ is unitary. Let $D_U \in \mathbb{C}^{n\times n}$ be a unitary diagonal matrix so that the upper triangular matrix $D_U^H U$ has positive diagonal entries. We then find from (7.7) that

$$\begin{aligned} A^k &= \widehat{Q}(I_n + \widehat{R}E_k\widehat{R}^{-1})(\widehat{R}\Lambda^k U) \\ &= (\widehat{Q}\widetilde{Q}_k)(\widetilde{R}_k\widehat{R}\Lambda^k U) \\ &= \left(\widehat{Q}\widetilde{Q}_k D_U D_\Lambda^k\right)\left(D_\Lambda^{-k} D_U^H \widetilde{R}_k \widehat{R} |\Lambda|^k D_\Lambda^k D_U D_U^H U\right) \\ &= \left(\widehat{Q}\widetilde{Q}_k D_U D_\Lambda^k\right)\left((D_U D_\Lambda^k)^{-1} \widetilde{R}_k \widehat{R} (D_U D_\Lambda^k) |\Lambda|^k D_U^H U\right). \end{aligned}$$

The first matrix in this decomposition is unitary, and the second matrix is upper triangular and has positive diagonal entries. Because of the uniqueness of the QR decomposition of A^k in (7.4) we must have

$$P_k = \widehat{Q}\widetilde{Q}_k D_U D_\Lambda^k \quad \text{and} \quad U_k = (D_U D_\Lambda^k)^{-1} \widetilde{R}_k \widehat{R} (D_U D_\Lambda^k) |\Lambda|^k D_U^H U.$$

By construction,

$$\begin{aligned} A^{(k)} &= P_k^H A P_k = P_k^H (X\Lambda X^{-1}) P_k \\ &= \left((D_\Lambda^k)^H D_U^H \widetilde{Q}_k^H \widehat{Q}^H\right)\left(\widehat{Q}\widehat{R}\Lambda\widehat{R}^{-1}\widehat{Q}^H\right)\left(\widehat{Q}\widetilde{Q}_k D_U D_\Lambda^k\right) \\ &= \left(\widetilde{Q}_k D_U D_\Lambda^k\right)^H \left(\widehat{R}\Lambda\widehat{R}^{-1}\right)\left(\widetilde{Q}_k D_U D_\Lambda^k\right). \end{aligned}$$

Since $\widetilde{Q}_k \to I_n$ for $k \to \infty$, we see that the sequence of matrices $A^{(k)}$ converge to upper triangular form with the eigenvalues of A on the diagonal. □

Note that a nonsymmetric matrix $A \in \mathbb{R}^{n\times n}$ in general has non-real eigenvalues which appear in complex conjugate pairs. In this case the assumption $|\lambda_1| > |\lambda_2| > \cdots > |\lambda_n|$ in Theorem 7.5 does *not* hold. For such matrices or other cases with multiple eigenvalues of the same absolute value, a convergence analysis of the QR algorithm is still possible, but it will be significantly more involved than the proof of Theorem 7.5. In case of a real matrix with a complex conjugate eigenvalue pair, for example, the limiting matrix will not be upper triangular but will contain a 2×2 diagonal block corresponding to this pair.

Example 7.6. *The real matrix* $A = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \in \mathbb{R}^{2\times 2}$ *has the characteristic polynomial* $z^2 + 1$ *and hence complex eigenvalues* $\lambda_1 = i$ *and* $\lambda_2 = -i$. *The assumption* $|\lambda_1| > |\lambda_2|$ *does not hold, and we can not apply Theorem 7.5 to conclude that the matrices* $A^{(k)}$ *in the QR algorithm converge to upper triangular form.*

Starting the QR algorithm with $Q^{(0)} = I_2$ yields $A^{(0)} = A$, and then

$$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = Q^{(1)}R^{(1)} = A^{(0)} \text{ (QR decomposition)},$$
$$A^{(1)} = R^{(1)}Q^{(1)} = A^{(0)}.$$

Thus, the QR algorithm completely stagnates with $A^{(k)} = A$ for all $k \geq 0$.

As shown by the proof of Theorem 7.5, the speed of convergence of the matrices $A^{(k)}$ to upper triangular form depends on the speed of convergence of the matrices E_k to zero. This in turn depends on the ratios $|\lambda_i/\lambda_j|$ for $i > j$. Smaller ratios will lead to a faster convergence of the corresponding entries of E_k to zero. It is therefore reasonable to expect that "well separated" eigenvalues (in the absolute value sense) will be approximated faster by the QR algorithm than "tightly clustered" eigenvalues. This intuition will be confirmed in Example 7.7.

The example uses the following straightforward MATLAB implementation of Algorithm 17 with $Q^{(0)} = I$:

```
function Ahat = qr_algorithm(A,maxit);
n = length(A);
Q = eye(n);
Ahat = Q'*A*Q;
for k = 1:maxit
    [Q,R] = qr(Ahat);
    Ahat = R*Q;
end
```

Example 7.7. *We consider the 5×5 real symmetric matrix*

$$A = \begin{bmatrix} 3 & 2 & -1 & 0 & 0 \\ 2 & 3 & 2 & -1 & 0 \\ -1 & 2 & 3 & 2 & -1 \\ 0 & -1 & 2 & 3 & 2 \\ 0 & 0 & -1 & 2 & 3 \end{bmatrix},$$

for which MATLAB computes the following eigenvalues, sorted by absolute value:

$$5.5616,\ 5.4020,\ 4.2523,\ -1.6543,\ 1.4384.$$

(In this example we only display a few significant digits, but all MATLAB computations were done in double precision; see Example 3.2.)
We have the following ratios of subsequent eigenvalues:

i	2	3	4	5
$\lvert\lambda_i/\lambda_{i-1}\rvert$	0.9713	0.7871	0.3890	0.8695

Notable here is that λ_1 and λ_2 are the closest of the five eigenvalues, and that λ_3 is the "most separated" eigenvalue.

Applying the MATLAB implementations of the QR algorithm as stated above with `maxit=20` *yields the following (approximately diagonal) matrix:*

```
Ahat =
    5.5588   -0.0206    0.0089   -0.0000   -0.0000
   -0.0206    5.4044   -0.0181    0.0000    0.0000
    0.0089   -0.0181    4.2526    0.0000    0.0000
         0    0.0000    0.0000   -1.6503   -0.1112
         0         0    0.0000   -0.1112    1.4344
```

The computed eigenvalue approximations $\lambda_{20}^{(i)}$ for $i = 1, 2, 3, 4, 5$ are the diagonal entries of this matrix. These approximations have the following relative forward errors:

i	1	2	3	4	5
$\lvert\lambda_i - \lambda_{20}^{(i)}\rvert/\lvert\lambda_i\rvert$	0.0024	0.0028	0.0001	0.0005	0.0005

In Figure 7.2 we plot the relative forward errors $|\lambda_i - \lambda_k^{(i)}|/|\lambda_i|$ for $k = 1, 2, \ldots, 200$, and $i = 1$ (dashed), $i = 3$ (dotted), $i = 5$ (solid). The "most separated" eigenvalue λ_3 is approximated fastest, followed by λ_5, and the approximation of λ_1 is the slowest of the three. (The error curves for λ_2 and λ_4 are very similar to the ones of λ_1 and λ_5, respectively.)

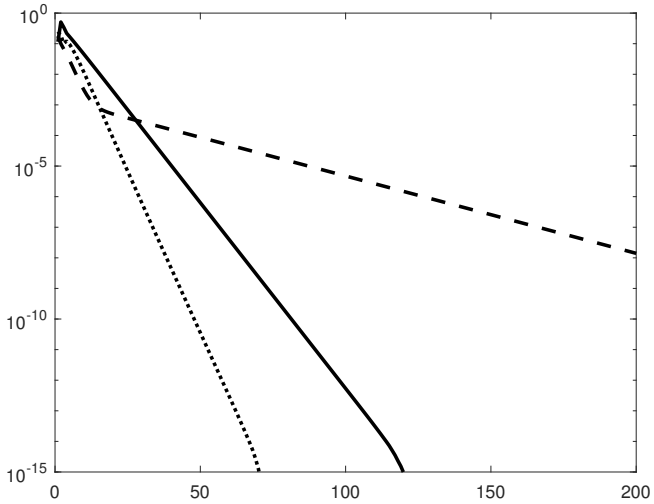


Figure 7.2: Relative forward errors of eigenvalue approximations computed by the QR algorithm in Example 7.7.

Chapter 8

Galerkin projection methods for eigenvalue problems

The general framework of the methods for the eigenvalue problem $Ax = \lambda x$ discussed in this chapter is closely related to the projection methods for solving linear algebraic systems $Ax = b$ that we studied in Chapter 4. Moreover, the specific methods that we construct are based on the Lanczos and Arnoldi algorithms, which are used also in the construction of the CG and MINRES/GMRES methods.

8.1 General framework

The idea of a Galerkin projection method for the eigenvalue problem $Ax = \lambda x$ is to compute a sequence of (approximate) eigenpairs

$$(\lambda_k, x_k) \in \mathbb{C} \times \mathcal{S}_k \setminus \{0\}, \quad k = 1, 2, \ldots, \tag{8.1}$$

where $\mathcal{S}_k \subset \mathbb{C}^n$ is a k-dimensional subspace, called the *search space*, such that the residual satisfies

$$Ax_k - \lambda_k x_k \perp \mathcal{S}_k. \tag{8.2}$$

The orthogonality condition in (8.2) is meant with respect to the Euclidean inner product. If (λ_k, x_k) satisfies (8.1)–(8.2), then λ_k and x_k are called a *Ritz value* and corresponding *Ritz vector* of the matrix A with respect to the subspace $\mathcal{S}_k$. Moreover, (λ_k, x_k) is called a *Ritz pair* of A with respect to the subspace $\mathcal{S}_k$.

Remark 8.1. Instead of $\mathcal{S}_k$ in (8.2) one could also consider a different k-dimensional subspace of $\mathbb{C}^n$. The projection process is then called a *Petrov-Galerkin projection method*, but we will not discuss such methods here.

In order to express the projection process (8.1)–(8.2) in matrix form, suppose that the columns of the matrix $S_k \in \mathbb{C}^{n \times k}$ form an orthonormal basis of the subspace $\mathcal{S}_k$. Then we have

$$x_k = S_k t_k$$

for some nonzero vector $t_k \in \mathbb{C}^k$, which is to be determined by the orthogonality condition

$$0 = S_k^H(Ax_k - \lambda_k x_k) = S_k^H A S_k t_k - \lambda_k S_k^H S_k t_k \quad \text{or} \quad S_k^H A S_k t_k = \lambda_k t_k. \tag{8.3}$$

Thus, each eigenpair (λ_k, t_k) of the matrix $S_k^H A S_k \in \mathbb{C}^{k\times k}$ gives a Ritz pair $(\lambda_k, S_k t_k)$ of A with respect to the subspace $\mathcal{S}_k$. In general there are up to k such pairs with linearly independent Ritz vectors. These Ritz pairs could be denoted by

$$(\lambda_1^{(k)}, S_k t_1^{(k)}),\ (\lambda_2^{(k)}, S_k t_2^{(k)}),\ \ldots,\ (\lambda_k^{(k)}, S_k t_k^{(k)}),$$

but for the moment we skip the double indices for notational simplicity.

We state the basic prototype of the Galerkin projection method in Algorithm 18.

Algorithm 18 Galerkin projection method

Input: $A \in \mathbb{C}^{n\times n}$, stopping criterion, maximal number of iterations $n_{\max}$
Output: approximate eigenpair(s) $(\lambda_k, S_k t_k)$ of A
for $k = 1, \ldots, n_{\max}$ **do**
  Determine $S_k \in \mathbb{C}^{n\times k}$ with orthonormal columns
  Solve the eigenvalue problem $S_k^H A S_k t_k = \lambda_k t_k$
  Test the Ritz pair(s) $(\lambda_k, S_k t_k)$ for convergence and stop if satisfied
end for

Lemma 8.2. *If the search space $\mathcal{S}_k$ is invariant under A, i.e., $A\mathcal{S}_k \subseteq \mathcal{S}_k$, then each Ritz pair of A with respect to $\mathcal{S}_k$ is an eigenpair of A.*

Proof. As above, let the columns of $S_k \in \mathbb{C}^{n\times k}$ form an orthonormal basis of $\mathcal{S}_k$. If $\mathcal{S}_k$ is invariant under A, then there exists a matrix $Z_k \in \mathbb{C}^{k\times k}$ such that $AS_k = S_k Z_k$, and hence $Z_k = S_k^H A S_k$. The Ritz pairs of A are of the form $(\lambda_k, S_k t_k)$, where (λ_k, t_k) is an eigenpair of the matrix Z_k. This gives

$$A S_k t_k = S_k Z_k t_k = \lambda_k S_k t_k,$$

so that the Ritz pair $(\lambda_k, S_k t_k)$ indeed is an eigenpair of A. □

This result implies that the iteration of a Galerkin projection method can be stopped once the subspace $\mathcal{S}_k$ becomes invariant under A, since then all Ritz pairs are exact eigenpairs of A (in exact arithmetic). Of course, in practice one still needs to compute the eigenvalues and corresponding eigenvectors of the matrix $S_k^H A S_k$. Even when no invariant subspace is reached, the approach can be efficient when for small k the eigenpairs of the $k \times k$ matrix $S_k^H A S_k$ closely approximate eigenpairs of A, and when from a computational point of view the structure of the matrix $S_k^H A S_k$ is "simpler" than the one of A. Examples of such methods are the *Lanczos method* and the *Arnoldi method*, where for both methods $\mathcal{S}_k$ is a Krylov subspace. Before we get to these specific methods, we will further study the general properties of the Galerkin projection method.

Let (λ_k, x_k) with $x_k = S_k t_k$ be a Ritz pair of A with respect to $\mathcal{S}_k$. In order to obtain bounds on how closely λ_k and x_k approximate exact eigenvalues and eigenvectors of A, we can apply our general results from Section 6.2. For example, if A is diagonalizable and $\|x_k\|_2 = 1$, then Theorem 6.3 implies that there exists an eigenvalue λ of A such that

$$|\lambda - \lambda_k| \leq \kappa_2(X)\|r\|_2, \quad \text{where} \quad r = Ax_k - \lambda_k x_k.$$

Moreover, if we multiply (8.3) from the left by t_k^H and use $S_k^H S_k = I$, we obtain the equation

$$t_k^H S_k^H A S_k t_k = \lambda_k t_k^H S_k^H S_k t_k,$$

or, equivalently,

$$\lambda_k = \frac{x_k^H A x_k}{x_k^H x_k} = R_A(x_k). \tag{8.4}$$

Thus, each Ritz value is a Rayleigh quotient of A and the corresponding Ritz vector. For Hermitian and normal matrices we can therefore also apply Theorem 6.7 and Theorem 6.8, respectively.
Most results from Section 6.2 involve the (usual) residual formed with the given matrix A and an approximate eigenpair (λ_k, x_k), i.e., $r = Ax_k - \lambda_k x_k$. Below we will study an alternative approach that considers a residual formed with an approximation of the matrix A and an exact eigenpair. In order to do this we multiply the equation (8.3) from the left with S_k and use that $S_k^H S_k = I$ to obtain the equation

$$(S_k S_k^H) A (S_k S_k^H) S_k t_k = \lambda_k S_k t_k,$$

or, equivalently,

$$A_k x_k - \lambda_k x_k = 0, \quad \text{where} \quad A_k := P_k A P_k, \quad P_k := S_k S_k^H, \quad x_k = S_k t_k.$$

The matrix $P_k \in \mathbb{C}^{n \times n}$ represents the orthogonal projection onto $\mathcal{S}_k$, i.e., we have $P_k^2 = P_k = P_k^H$ and $\operatorname{ran}(P_k) = \mathcal{S}_k$.
Each vector $x \in \mathbb{C}^n$ can be decomposed as $x = P_k x + (I_n - P_k)x$, where $P_k x \in \mathcal{S}_k$ and $(I_n - P_k)x \in \mathcal{S}_k^\perp$. Thus,

$$\|x\|_2^2 = \|P_k x + (I_n - P_k)x\|_2^2 = \|P_k x\|_2^2 + \|(I_n - P_k)x\|_2^2.$$

The norms $\|P_k x\|_2$ and $\|(I_n - P_k)x\|_2$ measure how much of x is contained in $\mathcal{S}_k$ and $\mathcal{S}_k^\perp$, respectively. Moreover, the following result shows that the vector $P_k x$ is the best approximation to x in the subspace $\mathcal{S}_k$.

Lemma 8.3. *For any $x \in \mathbb{C}^n$,*

$$\|x - P_k x\|_2 = \min_{y \in \mathcal{S}_k} \|x - y\|_2.$$

Proof. For each $y \in \mathcal{S}_k$,

$$\begin{aligned}\|x-y\|_2^2 &= \| \underbrace{(x-P_kx)}_{\in \mathcal{S}_k^\perp} + \underbrace{(P_kx-y)}_{\in \mathcal{S}_k} \|_2^2 = \|x-P_kx\|_2^2 + \|P_kx-y\|_2^2 \\ &\geq \|x-P_kx\|_2^2,\end{aligned}$$

with equality if and only if $P_kx = y$. □

If (λ, x) is an exact eigenpair of A, then the residual with respect to the matrix A_k, which can be considered the *Galerkin approximation of* A, is given by

$$\begin{aligned}A_kx - \lambda x &= (A_k - \lambda I_n)P_kx + (A_k - \lambda I_n)(I_n - P_k)x \\ &= (A_k - \lambda I_n)P_kx + (A_k - A_kP_k - \lambda I_n + \lambda P_k)x \\ &= (A_k - \lambda I_n)P_kx - \lambda(I_n - P_k)x,\end{aligned}$$

where we have used that $A_kP_k = P_kAP_k^2 = P_kAP_k = A_k$. The vectors $(A_k - \lambda I_n)P_kx$ and $(I_n - P_k)x$ are orthogonal, which can be seen from

$$\begin{aligned}x^H(I_n - P_k)^H(A_k - \lambda I_n)P_kx &= x^H(I_n - P_k)(A_k - \lambda I_n)P_kx \\ &= x^H(A_k - \lambda I_n - A_k + \lambda P_k)P_kx = 0.\end{aligned}$$

We therefore obtain

$$\|A_kx - \lambda x\|_2^2 = \|(A_k - \lambda I_n)P_kx\|_2^2 + |\lambda|^2\|(I_n - P_k)x\|_2^2.$$

Moreover,

$$\begin{aligned}(A_k - \lambda I_n)P_kx &= P_kAP_k^2x - P_k(\lambda x) = P_kAP_kx - P_kAx \\ &= P_kA(P_k - I_n)x = -P_kA(I_n - P_k)x.\end{aligned}$$

The derivations above can be summarized as follows.

Theorem 8.4. *As above, let P_k be the orthogonal projection onto the subspace $\mathcal{S}_k$, and let $A_k = P_kAP_k$. Then the residual of each eigenpair (λ, x) of A with respect to A_k satisfies*

$$\begin{aligned}\|A_kx - \lambda x\|_2^2 &= \|P_kA(I_n - P_k)x\|_2^2 + |\lambda|^2\|(I_n - P_k)x\|_2^2 \\ &\leq \left(\|P_kA(I_n - P_k)\|_2^2 + |\lambda|^2\right)\|(I_n - P_k)x\|_2^2. \qquad (8.5)\end{aligned}$$

In order to obtain the inequality we have used that $I_n - P_k = (I_n - P_k)^2$. On the right hand side of the bound (8.5) we have, in particular, the minimum distance between the eigenvector x and the subspace $\mathcal{S}_k$; see Lemma 8.3. For Krylov subspaces we will bound this distance in Theorem 8.15 below.

8.2 Implementation of the Lanczos method

The Lanczos method is a Galerkin projection method for computing approximations to the eigenvalues and eigenvectors of Hermitian matrices. Throughout this section we will therefore assume that

$$A = A^H \in \mathbb{C}^{n\times n}.$$

The Lanczos method is an implementation of Algorithm 18 with $\mathcal{S}_k = \mathcal{K}_k(A, v_1)$. We therefore need two main ingredients: (1) An algorithm to compute a matrix S_k whose columns form an orthonormal basis of the subspace $\mathcal{K}_k(A, v_1)$, and (2) a method to compute the Ritz pairs of the matrix $S_k^H A S_k$.
The orthonormal basis of the Krylov subspace is computed using the Lanczos algorithm, which also gives his name to the Lanczos *method* considered here. We use Algorithm 7 with the given Hermitian matrix A and some given initial vector v_1. (Note that here we do not consider a linear algebraic system, so v_1 is not an initial residual.)
After $k < d(A, v_1)$ steps of the Lanczos algorithm we obtain the (partial) decomposition

$$AV_k = V_k T_k + \beta_k v_{k+1} e_k^T, \tag{8.6}$$

where $V_k = [v_1, \ldots, v_k] \in \mathbb{C}^{n\times k}$ satisfies $V_k^H V_k = I_k$, and

$$T_k = V_k^H A V_k = \begin{bmatrix} \gamma_1 & \beta_1 & & \\ \beta_1 & \gamma_2 & & \\ & \ddots & \ddots & \beta_{k-1} \\ & & \beta_{k-1} & \gamma_k \end{bmatrix}.$$

Recall from Section 4.4 that T_k is a real symmetric tridiagonal matrix with positive off-diagonal entries, i.e., $\beta_1, \ldots, \beta_k > 0$. In summary, the Lanczos method can be stated as in Algorithm 19.

Algorithm 19 Lanczos method

Input: $A \in \mathbb{C}^{n\times n}$, unit norm vector $v_1 \in \mathbb{C}^n$, stopping criterion, maximal number of iterations $n_{\max}$
Output: approximate eigenvalues and/or eigenvectors of A
for $k = 1, \ldots, n_{\max}$ **do**
  Perform step k of the Lanczos algorithm and obtain V_k and T_k as in (8.6)
  Solve the eigenvalue problem $T_k z = \lambda z$
  Test the Ritz value(s) and/or Ritz vector(s) for convergence
end for

The symmetric tridiagonal matrix T_k is orthogonally diagonalizable,

$$T_k = Z_k \widehat{\Lambda}_k Z_k^T, \quad \text{where}$$
$$Z_k = [z_1^{(k)}, \ldots, z_k^{(k)}] \in \mathbb{R}^{k\times k}, \quad Z_k^T Z_k = I_k, \quad \widehat{\Lambda}_k = \operatorname{diag}(\widehat{\lambda}_1^{(k)}, \ldots, \widehat{\lambda}_k^{(k)}) \in \mathbb{R}^{k\times k}.$$

Each eigenpair $(\widehat{\lambda}_j^{(k)}, z_j^{(k)})$, $j = 1, \ldots, k$, of T_k gives a Ritz pair $(\widehat{\lambda}_j^{(k)}, \widehat{x}_j^{(k)})$, where $\widehat{x}_j^{(k)} = V_k z_j^{(k)}$, and the Galerkin conditions (8.1)–(8.2) are

$$\widehat{x}_j^{(k)} \in \mathcal{K}_k(A, v_1) \quad \text{and} \quad r_j^{(k)} = A\widehat{x}_j^{(k)} - \widehat{\lambda}_j^{(k)}\widehat{x}_j^{(k)} \perp \mathcal{K}_k(A, v_1).$$

Note that while the Lanczos *algorithm* has constant work and storage requirements in every step, the Ritz vectors $\widehat{x}_j^{(k)}$ in the Lanczos *method* require to store the entire matrix V_k. Thus, when eigenvector approximations are required, the storage requirements of the Lanczos method grow linearly with each iteration step of the Lanczos algorithm.
Let us have a closer look at the properties and the computation of the eigenvalues of T_k, i.e., the Ritz values. We define the polynomials

$$p_0(\mu) := 1 \quad \text{and} \quad p_j(\mu) := \det(T_j - \mu I_j), \quad j = 1, \ldots, k.$$

Thus, in particular, $p_1(\mu) = \gamma_1 - \mu$. Using the Laplace expansion of the determinant with respect to the last column it can be easily shown that

$$p_j(\mu) = (\gamma_j - \mu)p_{j-1}(\mu) - \beta_{j-1}^2 p_{j-2}(\mu), \quad j = 2, \ldots, k. \tag{8.7}$$

Thus, the characteristic polynomials of each Jacobi matrix T_j, $j = 1, \ldots, k$, can be computed using a 3-term recurrence. Since these matrices are real and symmetric, each polynomial $p_j(\mu)$ has j real roots. The following result, known as the *Cauchy interlacing theorem*, shows that these j roots are (pairwise) distinct, and that the roots of two consecutive polynomials interlace.

Theorem 8.5. *If $k \geq 2$, then for each $j = 1, \ldots, k-1$ the polynomial $p_j(\mu)$ has j distinct real roots which interlace (or separate) the $j+1$ distinct real roots of the polynomial $p_{j+1}(\mu)$.*

Proof. We first consider $p_1(\mu) = \gamma_1 - \mu$, which has the only real root $\mu = \gamma_1$. We have

$$p_2(\mu) = (\gamma_2 - \mu)(\gamma_1 - \mu) - \beta_1^2 = \mu^2 - (\gamma_1 + \gamma_2)\mu + (\gamma_1\gamma_2 - \beta_1^2),$$

and hence $p_2(\gamma_1) = -\beta_1^2 < 0$. The quadratic polynomial $p_2(\mu)$ satisfies $p_2(\mu) \to +\infty$ for $\mu \to -\infty$ and $\mu \to +\infty$, and hence it has one root strictly smaller and one root strictly larger than γ_1.
Suppose that the claim is true for all polynomials $p_1(\mu), \ldots, p_j(\mu)$, where $2 \leq j \leq k-2$. Let $\mu_1 < \mu_2$ denote two consecutive roots of $p_j(\mu)$. Then $p_{j-1}(\mu)$ has exactly one root between μ_1 and μ_2, and hence $p_{j-1}(\mu_1)$ and $p_{j-1}(\mu_2)$ have opposite signs. Using (8.7) we get

$$\begin{aligned} p_{j+1}(\mu_1) &= -\beta_j^2 p_{j-1}(\mu_1), \\ p_{j+1}(\mu_2) &= -\beta_j^2 p_{j-1}(\mu_2), \end{aligned}$$

which shows that $p_{j+1}(\mu_1)$ and $p_{j+1}(\mu_2)$ also have opposite signs. Thus, $p_{j+1}(\mu)$ has at least one root between μ_1 and μ_2, which shows that $j-1$ roots of $p_{j+1}(\mu)$ interlace the j real and distinct roots of $p_j(\mu)$.

Next observe that each $p_j(\mu)$, $j = 1, \ldots, k$, satisfies $p_j(\mu) \to +\infty$ for $\mu \to -\infty$. Furthermore, suppose that μ_1 is the smallest root of $p_j(\mu)$. By assumption, the roots of $p_{j-1}(\mu)$ interlace those of $p_j(\mu)$ and thus $p_{j-1}(\mu)$ has no root smaller than μ_1. We therefore must have $p_{j-1}(\mu_1) > 0$, and from $p_{j+1}(\mu_1) = -\beta_j^2 p_{j-1}(\mu_1)$ we see that $p_{j+1}(\mu_1) < 0$. This implies that $p_{j+1}(\mu)$ has one root strictly less than μ_1.
A similar argument shows that $p_{j+1}(\mu)$ also has one root strictly larger than the largest root of $p_j(\mu)$, which completes the proof. □

Using the interlacing property we will next show the *Sturm sequence property* of the polynomials $p_j(\mu)$, which can be used for computing the roots of $p_k(\mu)$ by bisection.

Theorem 8.6. *Let $\widetilde{\mu} \in \mathbb{R}$ be given and suppose that the sequence*

$$1, p_1(\widetilde{\mu}), \ldots, p_k(\widetilde{\mu})$$

contains no zero. Then the number of agreements in sign between consecutive numbers in this sequence is equal to the number of roots of $p_k(\mu)$ that are strictly larger than $\widetilde{\mu}$.

Proof. For the given $\widetilde{\mu} \in \mathbb{R}$ and $1 \leq j \leq k$ we denote by $s_j(\widetilde{\mu})$ the number of agreements in sign between consecutive numbers in the sequence

$$1, p_1(\widetilde{\mu}), \ldots, p_j(\widetilde{\mu}),$$

and by $g_j(\widetilde{\mu})$ the number of roots of $p_j(\mu)$ that are strictly larger than $\widetilde{\mu}$.
We prove the statement by induction on j.
For $j = 1$ we have the sequence $1, p_1(\widetilde{\mu})$. We know that $p_1(\mu) \to +\infty$ for $\mu \to -\infty$. If $p_1(\widetilde{\mu}) > 0$, and hence $s_1(\widetilde{\mu}) = 1$, then $\widetilde{\mu}$ is strictly less than the only root of $p_1(\mu)$, and hence $g_1(\widetilde{\mu}) = 1$. If $p_1(\widetilde{\mu}) < 0$, and hence $s_1(\widetilde{\mu}) = 0$, then $\widetilde{\mu}$ is strictly larger than the only root of $p_1(\mu)$, and hence $g_1(\widetilde{\mu}) = 0$.
Now suppose that the statement is true for the numbers $1, p_1(\widetilde{\mu}), \ldots, p_j(\widetilde{\mu})$, where $1 \leq j \leq k-1$. We need to show that $s_{j+1}(\widetilde{\mu}) = g_{j+1}(\widetilde{\mu})$. If $p_j(\widetilde{\mu})$ and $p_{j+1}(\widetilde{\mu})$ have the same sign, then $s_{j+1}(\widetilde{\mu}) = s_j(\widetilde{\mu}) + 1$. Otherwise, $s_{j+1}(\widetilde{\mu}) = s_j(\widetilde{\mu})$.
Suppose that $\widetilde{\mu} \in (\mu_1, \mu_2)$, where μ_1 and μ_2 are two consecutive roots of $p_j(\mu)$. From the interlacing property we know that there exists exactly one root of $p_{j+1}(\mu)$ in the interval (μ_1, μ_2). Denote that root by ρ. If $\widetilde{\mu} < \rho$, then $p_j(\widetilde{\mu})$ and $p_{j+1}(\widetilde{\mu})$ have the same sign, and both polynomials have the same number of roots that are smaller than $\widetilde{\mu}$. Thus, $p_{j+1}(\mu)$ must have one more root that is larger than $\widetilde{\mu}$, which gives $s_{j+1}(\widetilde{\mu}) = s_j(\widetilde{\mu}) + 1 = g_j(\widetilde{\mu}) + 1 = g_{j+1}(\widetilde{\mu})$.
A similar argument shows that $s_{j+1}(\widetilde{\mu}) = g_{j+1}(\widetilde{\mu})$ when $\rho < \widetilde{\mu}$. Moreover, this argument can be easily modified for the cases when $\widetilde{\mu}$ is larger than the largest or smaller than the smallest root of $p_j(\mu)$. □

The theorem above can easily be extended to the case that a zero value occurs in the

sequence $1, p_1(\widetilde{\mu}), \ldots, p_k(\widetilde{\mu})$. In this case we can define[1] the "sign" of the zero value $p_j(\widetilde{\mu})$ as the sign of $p_{j-1}(\widetilde{\mu})$. Note that $p_{j-1}(\widetilde{\mu}) = p_j(\widetilde{\mu}) = 0$ cannot occur. This would imply that also $p_{j-2}(\widetilde{\mu}) = \cdots = p_0(\widetilde{\mu}) = 0$, which is impossible since $p_0(\mu) = 1$ for all μ.
The following example shows how roots of $p_k(\mu)$ and thus eigenvalues of T_k can be computed using the Sturm sequence property and bisection. The approach is well suited for situations where only a few of the eigenvalues of T_k should be computed. This in turn is typical in many practical applications.

Example 8.7. *We apply $k = 5$ steps of the Lanczos algorithm to $A = \text{tridiag}(-1, 2, -1) \in \mathbb{R}^{50\times 50}$ and the initial vector $v_1 = 50^{-1/2}[1, \ldots, 1]^T$. Since A is positive definite, the resulting Jacobi matrix T_5 is positive definite as well.*
For $\widetilde{\mu}_1 = 2$ we obtain the following values $p_j(\widetilde{\mu}_1)$ for $j = 0, 1, \ldots, 5$:

$$1.0000 \quad -1.9600 \quad -0.0417 \quad 1.9565 \quad 0.0455 \quad -1.9524$$

There are two agreements in sign, and hence T_5 has three eigenvalues in the interval $(0, 2)$ and two eigenvalues larger than $\widetilde{\mu}_1$.
For $\widetilde{\mu}_2 = 1$ we obtain the following values $p_j(\widetilde{\mu}_2)$ for $j = 0, 1, \ldots, 5$:

$$1.0000 \quad -0.9600 \quad -1.0000 \quad -0.0435 \quad 0.9545 \quad 1.0000$$

There are three agreements in sign and hence T_5 has two eigenvalues in $(0, 1)$, one eigenvalue in $(1, 2)$ and two eigenvalues larger than $\widetilde{\mu}_1$.
For $\widetilde{\mu}_3 = 0.5$ we obtain the following values $p_j(\widetilde{\mu}_3)$ for $j = 0, 1, \ldots, 5$:

$$1.0000 \quad -0.4600 \quad -0.7292 \quad -0.6359 \quad -0.2273 \quad 0.2932$$

There are three agreements in sign and hence T_5 has two eigenvalues in $(0, 0.5)$, one eigenvalue in $(0.5, 2)$ and two eigenvalues larger than $\widetilde{\mu}_1$.
For $\widetilde{\mu}_4 = 0.25$ we obtain the following values $p_j(\widetilde{\mu}_4)$ for $j = 0, 1, \ldots, 5$:

$$1.0000 \quad -0.2100 \quad -0.4062 \quad -0.5020 \quad -0.4741 \quad -0.3297$$

There are four agreements in sign and hence T_5 has one eigenvalue in $(0, 0.25)$, one eigenvalue in $(0.25, 0.5)$, one eigenvalue in $(0.5, 2)$ and two eigenvalues larger than $\widetilde{\mu}_1$.

[1] This definition goes back (at least) to James M. Ortega (1932–2008) who found as a PhD student at Stanford University that an earlier definition given by the numerical linear algebra pioneer James Wallace Givens (1910–1993) was incorrect [28, p. 260]: "However, the classical theory of a Sturm sequence, expounded in [2], needs some extension to give signs to zero values in the sequence. We have noticed that the extension of Givens [namely $\text{sign}(p_j(\widetilde{\mu})) = +1$ when $p_j(\widetilde{\mu}) = 0$] in the text of [1] is not quite correct. The difficulty is a purely algebraic one and has nothing to do with the digital realization on a computer. Professor Givens [personal statement] concurs in this, but states that the machine codes in [1] are correct."

This process can be continued until inclusion intervals of sufficient accuracy for the desired eigenvalues are found.
(The eigenvalues of T_5 computed by MATLAB are 0.0090, 0.3993, 1.3947, 2.6247, 3.6199.*)*

8.3 Convergence analysis of the Lanczos method

We will now study the convergence of the Lanczos method and start with an *a posteriori* bound on the forward error in the Ritz values.

Corollary 8.8. *In the previous notation, let $(\widehat{\lambda}_j^{(k)}, \widehat{x}_j^{(k)})$, $j = 1, \ldots, k$, be the Ritz pairs of A with respect to $\mathcal{K}_k(A, v_1)$. Then for each Ritz value $\widehat{\lambda}_j^{(k)}$ there exists an eigenvalue λ of A such that*

$$|\lambda - \widehat{\lambda}_j^{(k)}| \leq \|r_j^{(k)}\|_2 = \beta_k |e_k^T z_j^{(k)}|. \tag{8.8}$$

Proof. Using (8.6) we get

$$r_j^{(k)} = A\widehat{x}_j^{(k)} - \widehat{\lambda}_j^{(k)}\widehat{x}_j^{(k)} = AV_k z_j^{(k)} - V_k T_k z_j^{(k)} = \beta_k (e_k^T z_j^{(k)}) v_{k+1},$$

and $\|r_j^{(k)}\|_2 = \beta_k |e_k^T z_j^{(k)}|$ follows by taking norms. (Recall that $\beta_k > 0$.) The existence of an eigenvalue λ of A with $|\lambda - \widehat{\lambda}_j^{(k)}| \leq \|r_j^{(k)}\|_2$ follows from Corollary 6.4 since the Hermitian matrix A is normal. □

Note that $|e_k^T z_j^{(k)}| < 1$ since $\|z_j^{(k)}\|_2 = 1$. The inequality (8.8) is valid for all $k = 1, \ldots, d(A, v_1)$. In particular, for $k = d(A, v_1)$ we have $\beta_k = 0$ in the Lanczos algorithm, and the right hand side of (8.8) is zero. This gives another proof that the Ritz values of A are exact eigenvalues when the given space (here $\mathcal{K}_k(A, v_1)$) is invariant under A; see Lemma 8.2.
In addition, the equality in (8.8) shows that the residual norm $\|r_j^{(k)}\|_2$ is available without explicit knowledge of the Ritz vector $\widehat{x}_j^{(k)}$. Consequently, when we are only interested in eigenvalue approximations and use (8.8) for checking the convergence, we do not have to store the matrix V_k which contains the Lanczos vectors.
Each Ritz pair $(\widehat{\lambda}_j^{(k)}, \widehat{x}_j^{(k)})$ in the Lanczos method satisfies

$$\|\widehat{x}_j^{(k)}\|_2^2 = (V_k z_j^{(k)})^H V_k z_j^{(k)} = 1,$$

and

$$\widehat{\lambda}_j^{(k)} = (z_j^{(k)})^T T_k z_j^{(k)} = (z_j^{(k)})^T V_k^H A V_k z_j^{(k)} = (\widehat{x}_j^{(k)})^H A \widehat{x}_j^{(k)} = R_A(\widehat{x}_j^{(k)}),$$

which is just a special case of (8.4). From Theorem 6.7 and Theorem 6.8 we now get the following a posteriori bounds on the eigenvalue and eigenvector approximations generated by the the Lanczos method.

Corollary 8.9. *Let $(\widehat{\lambda}_j^{(k)}, \widehat{x}_j^{(k)})$, $j = 1, \ldots, k$, be the Ritz pairs of A with respect to $\mathcal{K}_k(A, v_1)$.*

(1) If λ is the eigenvalue of A that is closest to $\widehat{\lambda}_j^{(k)}$, and $\delta_j := \min\{|\lambda_i - \widehat{\lambda}_j^{(k)}| : \lambda_i \neq \lambda\}$ is the distance of $\widehat{\lambda}_j^{(k)}$ to the rest of the spectrum of A, then

$$|\lambda - \widehat{\lambda}_j^{(k)}| \leq \frac{\|r_j^{(k)}\|_2^2}{\delta_j}.$$

(2) If the eigenvalue λ in (1) is simple and x is a corresponding unit norm eigenvector, then

$$\sin\theta(\widehat{x}_j^{(k)}, x) \leq \frac{\|r_j^{(k)}\|_2}{\delta_j}.$$

As shown by the easily computable forward error bounds in Corollary 8.8 and 8.9, a small residual norm $\|r_j^{(k)}\|_2$ means that the Ritz value $\widehat{\lambda}_j^{(k)}$ closely approximates *some* eigenvalue of A. In practice we usually apply the Lanczos method to (very) large and sparse matrices A, and we want to obtain good approximations of eigenvalues of A after only a few steps, i.e., for $k \ll n$. Thus, an essential question that is not answered by Corollary 8.8 and 8.9 is *which* eigenvalues of A are (well) approximated by Ritz values particularly for small k. The Ritz vectors generated by the Lanczos method are contained in the Krylov subspace

$$\mathcal{K}_k(A, v_1) = \operatorname{span}\{v_1, Av_1, \ldots, A^{k-1}v_1\},$$

and the Ritz values are the eigenvalues of $T_k = V_k^H A V_k$, where the columns of V_k form an orthonormal basis of this Krylov subspace. From the analysis of the power method in Chapter 7 we know that the sequence of vectors $A^k v_1 / \|A^k v_1\|_2$ converges towards an eigenvector of A corresponding to a dominant eigenvalue. We therefore expect, at least heuristically, that the Lanczos method converges faster for the larger eigenvalues of A and corresponding eigenvectors. The following example illustrates this effect.

Example 8.10. *We consider the same matrices and random initial vectors as in Example 7.3.*
The pluses (+) on the bottom in the left parts of Figures 8.1 and 8.2 show the eigenvalues of $A_1 = L_{50}$ (top) and $A_2 = L_{20} \otimes I_{20} + I_{20} \otimes L_{20}$ (bottom). They are contained in the intervals $(0, 4)$ and $(0, 8)$, respectively. In each figure, from the top to the bottom, each row of dots (.) shows one set of the Ritz values $\widehat{\lambda}_1^{(k)}, \ldots, \widehat{\lambda}_k^{(k)}$ for $k = 1, 2, \ldots, 40$. We observe that the Ritz values quickly approach the outer parts of the spectrum of A. The approximation of interior eigenvalues appears to be (much) poorer.
In the right parts of Figures 8.1 and 8.2 we show the difference between the largest eigenvalue of A and the largest Ritz value, i.e., $\lambda_1 - \widehat{\lambda}_1^{(k)}$, for $k = 1, 2, \ldots, 40$. The dashed lines, which are the same lines as in Figure 7.1, shows the error in the power method (where only one eigenvalue approximation is computed in each step). We observe that the Lanczos method convergences significantly faster than the power method.

We will next derive an *a priori* bound for the convergence of the largest Ritz value in the Lanczos method, which will also explain the differences between the Lanczos method and

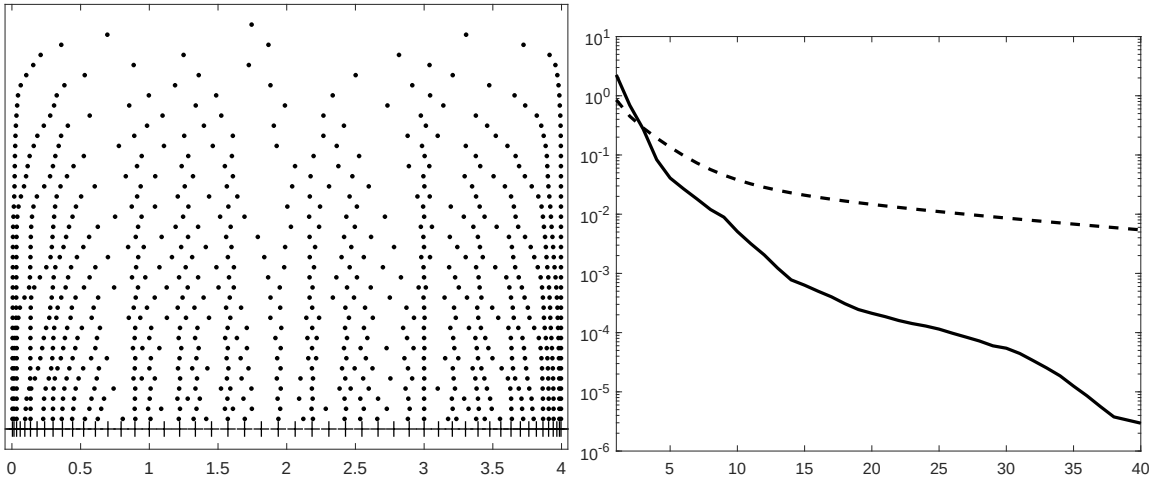


Figure 8.1: Left: Ritz value behavior for $A_1 = L_{50}$ and a random initial vector v_1. Right: Error in the largest eigenvalue for the Lanczos method (solid) and the power method (dashed) for A and v_1.

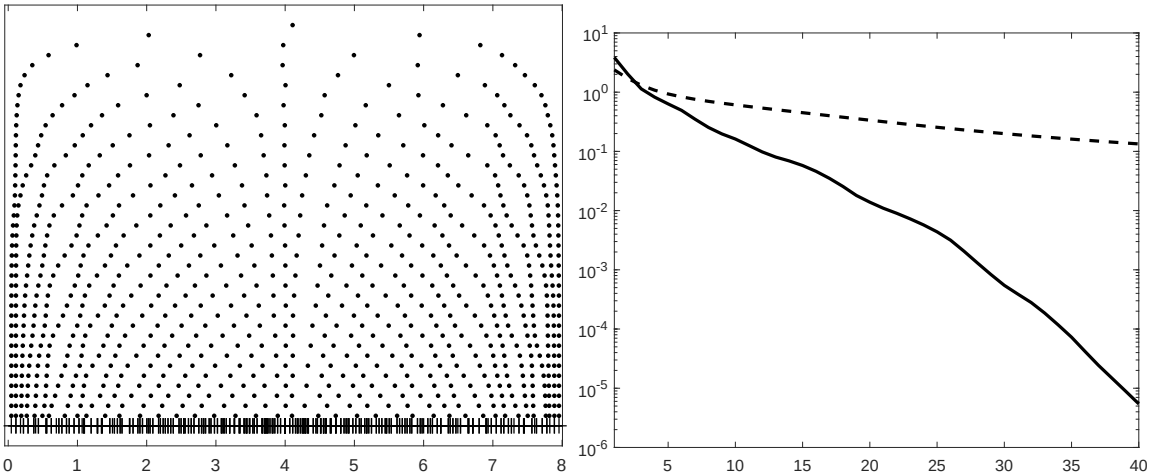


Figure 8.2: Left: Ritz value behavior for $A_2 = L_{20} \otimes I_{20} + I_{20} \otimes L_{20}$ and a random initial vector v_1. Right: Error in the largest eigenvalue for the Lanczos method (solid) and the power method (dashed) for A and v_1.

the power method illustrated in the right parts of Figures 8.1 and 8.2. In the proof we use that for any Hermitian matrix $M \in \mathbb{C}^{n\times n}$ we have

$$\lambda_{\max}(M) = \max_{0\neq z\in\mathbb{C}^n} R_M(z),$$

which is easy to prove.

Theorem 8.11. *Let $A = A^H \in \mathbb{C}^{n\times n}$ and a unit norm vector $v_1 \in \mathbb{C}^n$ be given. If $\lambda_1 > \lambda_2 \geq \cdots \geq \lambda_n$ are the (real) eigenvalues of A, x_1 is a unit norm eigenvector corresponding to λ_1, $|x_1^H v_1| = \cos\theta(x_1, v_1) \neq 0$, and $\widehat{\lambda}_1^{(k)}$ is the largest Ritz value, then*

$$0 \;\leq\; \lambda_1 - \widehat{\lambda}_1^{(k)} \;\leq\; (\lambda_1 - \lambda_n)\,\frac{\sin^2\theta(x_1, v_1)}{\cos^2\theta(x_1, v_1)} \min_{p\in\mathcal{P}_{k-1}(\lambda_1)} \max_{2\leq j\leq n} |p(\lambda_j)|^2\,, \tag{8.9}$$

where $\mathcal{P}_{k-1}(\lambda_1) := \{p \in \mathcal{P}_{k-1} \,:\, p(\lambda_1) = 1\}$.

Proof. If $\widehat{x}_1^{(k)}$ is a Ritz vector corresponding to $\widehat{\lambda}_1^{(k)}$, then

$$\begin{aligned}\widehat{\lambda}_1^{(k)} &= \max_{0\neq z\in\mathbb{C}^k} R_{T_k}(z) = \max_{0\neq z\in\mathbb{C}^k} \frac{z^H V_k^H A V_k z}{z^H z} = \max_{0\neq w\in\mathcal{K}_k(A,v_1)} R_A(w)\\ &\leq \max_{0\neq w\in\mathbb{C}^n} R_A(w) = \lambda_1,\end{aligned}$$

and hence $0 \leq \lambda_1 - \widehat{\lambda}_1^{(k)}$.
The Hermitian matrix A has a complete orthonormal set of eigenvectors, say $x_1, \ldots, x_n$, and we can write $v_1 = \sum_{i=1}^n \alpha_i x_i$, where $\alpha_i := x_i^H v_1$. By construction,

$$1 = \|v_1\|_2^2 = \sum_{i=1}^n |\alpha_i|^2 = |x_1^H v_1|^2 + \sum_{i=2}^n |\alpha_i|^2 = \cos^2\theta(x_1, v_1) + \sum_{i=2}^n |\alpha_i|^2,$$

giving $\sum_{i=2}^n |\alpha_i|^2 = 1 - \cos^2\theta(x_1, v_1) = \sin^2\theta(x_1, v_1)$.
Using that

$$-\widehat{\lambda}_1^{(k)} \;=\; -\max_{0\neq w\in\mathcal{K}_k(A,v_1)} R_A(w) \;=\; \min_{0\neq w\in\mathcal{K}_k(A,v_1)} R_{-A}(w)\,,$$

and $\mathcal{K}_k(A, v_1) = \{p(A)v_1 \,:\, p \in \mathcal{P}_{k-1}\}$, we obtain

$$\begin{aligned}\lambda_1 - \widehat{\lambda}_1^{(k)} &= \min_{0\neq w\in\mathcal{K}_k(A,v_1)} R_{\lambda_1 I_n - A}(w) = \min_{0\neq p\in\mathcal{P}_{k-1}} \frac{(p(A)v_1)^H\,(\lambda_1 I_n - A)\,p(A)v_1}{(p(A)v_1)^H\,p(A)v_1}\\ &= \min_{0\neq p\in\mathcal{P}_{k-1}} \frac{\sum_{j=2}^n (\lambda_1 - \lambda_j)|\alpha_j p(\lambda_j)|^2}{\sum_{j=1}^n |\alpha_j p(\lambda_j)|^2}\\ &\leq (\lambda_1 - \lambda_n) \min_{p\in\mathcal{P}_{k-1}(\lambda_1)} \frac{\sum_{j=2}^n |\alpha_j p(\lambda_j)|^2}{|\alpha_1|^2 + \sum_{j=2}^n |\alpha_j p(\lambda_j)|^2}\\ &\leq (\lambda_1 - \lambda_n) \min_{p\in\mathcal{P}_{k-1}(\lambda_1)} \frac{\sum_{j=2}^n |\alpha_j p(\lambda_j)|^2}{|\alpha_1|^2}\\ &\leq (\lambda_1 - \lambda_n) \min_{p\in\mathcal{P}_{k-1}(\lambda_1)} \left(\max_{2\leq j\leq n} |p(\lambda_j)|^2\, \frac{\sum_{i=2}^n |\alpha_i|^2}{|\alpha_1|^2}\right),\end{aligned}$$

which proves the second inequality in (8.9). □

The first inequality in (8.9) shows that the Ritz values $\widehat{\lambda}_1^{(k)}$, $k = 1, 2, \ldots$, must approach the largest eigenvalue of A from below. The upper bound on the (forward) error $\lambda_1 - \widehat{\lambda}_1^{(k)}$ contains (1) the "spread" of the eigenvalues of A, (2) a measure for the angle between x_1 and the initial vector v_1, and (3) the value of a polynomial minimization problem, where the polynomials are normalized at λ_1 (and not at 0 as in the convergence bounds for CG and GMRES; see Theorem 4.13 and equation (4.30)).
Suppose that $\lambda_1 > \lambda_2 \geq \cdots \geq \lambda_n \geq 0$, then the error bound (7.2) for the power method is given by

$$\lambda_1 - \lambda^{(k)} \leq (\lambda_1 - \lambda_n) \frac{\sin^2 \theta(q^{(0)}, x_1)}{\cos^2 \theta(q^{(0)}, x_1)} \left| \frac{\lambda_2}{\lambda_1} \right|^{2k}.$$

If we take the polynomial $p(z) = (z/\lambda_1)^{k-1} \in \mathcal{P}_{k-1}(\lambda_1)$ in (8.9), we obtain the bound

$$\lambda_1 - \widehat{\lambda}_1^{(k)} \;\leq\; (\lambda_1 - \lambda_n) \frac{\sin^2 \theta(x_1, v_1)}{\cos^2 \theta(x_1, v_1)} \left| \frac{\lambda_2}{\lambda_1} \right|^{2k-2},$$

which is almost the same as the bound for the power method. In the bound (8.9) we minimize over all polynomials of degree at most $k-1$ that are normalized at λ_1, and therefore the value of this bound can be significantly smaller than the value of the bound for the power method. Thus, we expect that the largest Ritz value in the Lanczos method converges much faster to the largest eigenvalue of A than the eigenvalue approximation generated by the power method; see the right parts of Figures 8.1 and 8.2 for a numerical example.
A computable upper bound on the value of the polynomial minimization problem in (8.9) can be derived using suitably shifted and normalized Chebyshev polynomials; cf. the proof of Theorem 4.13.
The Chebyshev polynomials of the first kind on the interval $[-1, 1]$ are given by $C_k(z) = \cos(k \cos^{-1}(z))$, which shows that $|C_k(z)| \leq 1$ for $z \in [-1, 1]$. The simple linear transformation

$$\rho(z) = \frac{2z - \lambda_2 - \lambda_n}{\lambda_2 - \lambda_n}$$

maps the interval $[\lambda_n, \lambda_2]$ to the interval $[-1, 1]$, and thus $|C_k(\rho(z))| \leq 1$ for $z \in [\lambda_n, \lambda_2]$. Consequently,

$$\begin{aligned}
\min_{p \in \mathcal{P}_{k-1}(\lambda_1)} \max_{2 \leq j \leq n} \left| \frac{p(\lambda_j)}{p(\lambda_1)} \right| &\leq \min_{p \in \mathcal{P}_{k-1}(\lambda_1)} \max_{z \in [\lambda_n, \lambda_2]} \left| \frac{p(z)}{p(\lambda_1)} \right| \\
&\leq \max_{z \in [\lambda_n, \lambda_2]} \left| \frac{C_{k-1}(\rho(z))}{C_{k-1}(\rho(\lambda_1))} \right| \\
&\leq \frac{1}{|C_{k-1}(\rho(\lambda_1))|}, \quad \text{where} \quad \rho(\lambda_1) = \frac{2\lambda_1 - \lambda_2 - \lambda_n}{\lambda_2 - \lambda_n} = 1 + 2\frac{\lambda_1 - \lambda_2}{\lambda_2 - \lambda_n}.
\end{aligned}$$

If $\lambda_1 \gg \lambda_2$ and the distance $\lambda_2 - \lambda_n$ is small, then $\rho(\lambda_1) \gg 1$ and the value $|C_{k-1}(\rho(\lambda_1))|$ will be very large. For example, if $\lambda_1 - \lambda_2 = 10$ and $\lambda_2 - \lambda_n = 1$, so that $\rho(\lambda_1) = 21$, then

we can use

$$C_0(z) = 1, \quad C_1(z) = z, \quad \text{and} \quad C_{k+1}(z) = 2zC_k(z) - C_{k-1}(z), \quad k = 1, 2, \ldots$$

to obtain

$$C_1(\rho(\lambda_1)) = 21, \;\; C_2(\rho(\lambda_1)) = 881, \;\; C_3(\rho(\lambda_1)) = 36.981, \;\; C_4(\rho(\lambda_1)) = 1.552.321.$$

In such cases the bound (8.9) predicts a rapid convergence of the largest Ritz value to the largest eigenvalue of A.
Deriving a bound for the interior eigenvalues and Ritz values is more difficult. Such bounds will be of a similar nature as (8.9), but now the polynomial will be normalized at an interior eigenvalue; cf. the bound in Theorem 8.15. The value of the corresponding polynomial minimization problem will decrease to zero much slower than for the case when the normalization point is one of the outermost eigenvalues. This reasoning explains why the convergence of the Lanczos method to interior eigenvalues of A (usually) is slower than to the extreme eigenvalues (in particular the largest ones). A numerical example for this phenomenon was shown above.

8.4 Implementation and convergence analysis of the Arnoldi method

We will now implement the Galerkin projection method (see Algorithm 18) with $\mathcal{S}_k = \mathcal{K}_k(A, v_1)$ for general non-Hermitian matrices. Instead of the Lanczos algorithm, which is well defined only for Hermitian matrices, we will use the Arnoldi algorithm (see Algorithm 6) for generating the required orthonormal Krylov subspace bases, and this will give the *Arnoldi method*.
If the initial vector v_1 has the grade $d = d(A, v_1)$ with respect to A, then the Arnoldi algorithm yields a decomposition of the form $AV_d = V_d H_d$; see (4.16). After $k < d(A, v_1)$ steps we obtain a partial decomposition of the form

$$AV_k = V_k H_k + h_{k+1,k} v_{k+1} e_k^T, \tag{8.10}$$

where $V_k^H V_k = I_k$ (in exact arithmetic) and hence $V_k^H A V_k = H_k$, which gives the Arnoldi method as stated in Algorithm 20.
Each eigenpair $(\widehat{\lambda}_j^{(k)}, z_j^{(k)})$ of H_k gives a Ritz pair $(\widehat{\lambda}_j^{(k)}, \widehat{x}_j^{(k)})$, where $\widehat{x}_j^{(k)} = V_k z_j^{(k)}$, and the Galerkin conditions (8.1)–(8.2) are

$$\widehat{x}_j^{(k)} \in \mathcal{K}_k(A, v_1) \quad \text{and} \quad r_j^{(k)} = A\widehat{x}_j^{(k)} - \widehat{\lambda}_j^{(k)} \widehat{x}_j^{(k)} \perp \mathcal{K}_k(A, v_1).$$

The matrix $H_k \in \mathbb{C}^{k \times k}$ is an unreduced upper Hessenberg matrix. The Hessenberg structure is "almost" upper triangular and hence well suited for applying the QR algorithm in order to solve the eigenvalue problem with H_k; see the discussion of Algorithm 17.

Algorithm 20 Arnoldi method

Input: $A \in \mathbb{C}^{n\times n}$, unit norm vector $v_1 \in \mathbb{C}^n$, stopping criterion, maximal number of iterations $n_{\max}$
Output: approximate eigenvalues and/or eigenvectors of A
for $k = 1, \ldots, n_{\max}$ **do**
  Perform step k of the Arnoldi algorithm and obtain V_k and H_k
  Solve the eigenvalue problem $H_k z = \lambda z$
  Test the Ritz value(s) and/or Ritz vector(s) for convergence
end for

While the Lanczos algorithm for Hermitian matrices is based on 3-term recurrences and therefore has constant work and storage requirements in every step (similar to CG), the Arnoldi algorithm in general requires full recurrences. Hence work and storage in the Arnoldi algorithm grow linearly with the number of iterations (similar to GMRES). This fact, and the more complicated reduced problem (upper Hessenberg instead of tridiagonal matrix) make the computations in the Arnoldi method (for general A) significantly more expensive than in the Lanczos method (for Hermitian A).

We will now analyze the convergence properties of the Arnoldi method. We start with a result analogous to Corollary 8.8.

Corollary 8.12. *If A is diagonalizable, $A = X\Lambda X^{-1}$, and $(\widehat{\lambda}_j^{(k)}, \widehat{x}_j^{(k)})$ is any Ritz pair with corresponding residual $r_j^{(k)}$, then there exists an eigenvalue λ of A, such that*

$$|\lambda - \widehat{\lambda}_j^{(k)}| \leq \kappa_2(X)\, \|r_j^{(k)}\|_2 = \kappa_2(X) h_{k+1,k}\, |e_k^T z_j^{(k)}|.$$

Proof. The inequality follows directly from (6.5) in Theorem 6.3, and the equality follows upon multiplying $AV_k - V_k H_k = h_{k+1,k} v_{k+1} e_k^T$ from the right with $z_j^{(k)}$. Note that $h_{k+1,k} > 0$ if $k < d(A, v_1)$. □

If $\kappa_2(X)$ is small, then the quantity $h_{k+1,k}\, |e_k^T z_j^{(k)}|$, which (in exact arithmetic) is equal to the residual norm, yields a computable a posteriori bound on the forward error in the eigenvalue approximation. For a general (diagonalizable) matrix A, however, the constant $\kappa_2(X)$ is potentially very large. Hence a small residual norm does not guarantee that the Ritz value $\widehat{\lambda}_j^{(k)}$ closely approximates any eigenvalue of A; see Example 6.5.

Let $k < d(A, v_1)$, so that $\mathcal{K}_k(A, v_1)$ is not invariant under A, let the columns of the matrix $V_k = [v_1, \ldots, v_k] \in \mathbb{C}^{n\times k}$ form an orthonormal basis of $\mathcal{K}_k(A, v_1)$, and let $P_k = V_k V_k^H$ be the corresponding orthogonal projection. Then for each $j = 1, \ldots, k-1$ we have

$$A^j v_1 = P_k A^j v_1 = P_k A(A^{j-1} v_1) = P_k A(P_k A^{j-1} v_1) = P_k A P_k A^{j-1} v_1 = A_k A^{j-1} v_1,$$

where $A_k = P_k A P_k$. Thus, by induction, $A^j v_1 = A_k^j v_1$ for $j = 0, 1, \ldots, k-1$. In addition,

$$P_k A^k v_1 = P_k A(A^{k-1} v_1) = P_k A(P_k A^{k-1} v_1) = A_k^k v_1.$$

This shows that

$$P_k p(A) v_1 = p(A_k) v_1 \tag{8.11}$$

holds for every polynomial p of degree at most k. Next note that

$$A_k^j = (P_k A P_k)^j = (V_k V_k^H A V_k V_k^H)^j = V_k (V_k^H A V_k)^j V_k^H,$$

and, therefore,

$$p(A_k) v_1 = V_k p(V_k^H A V_k) V_k^H v_1 = V_k p(V_k^H A V_k) e_1. \tag{8.12}$$

This will be used in the proof of the next result.

Theorem 8.13. *In the notation established above, if q_k is the (monic) characteristic polynomial of the matrix $V_k^H A V_k \in \mathbb{C}^{k\times k}$, then*

$$\|q_k(A) v_1\|_2 = \min_{\substack{p\in\mathcal{P}_k \\ p \text{ monic}}} \|p(A) v_1\|_2,$$

where $\mathcal{P}_k$ denotes the set of polynomials of degree at most k.

Proof. By the Cayley-Hamilton theorem, we have $q_k(V_k^H A V_k) = 0$. Using (8.11)–(8.12) we obtain

$$0 = V_k q_k(V_k^H A V_k) e_1 = V_k V_k^H q_k(A) v_1,$$

which means that $V_k^H q_k(A) v_1 = 0$, and thus

$$q_k(A) v_1 \perp \mathcal{K}_k(A, v_1). \tag{8.13}$$

If $s \in \mathcal{P}_{k-1}$ is arbitrary and we write $q_k(z) = z^k - s_{k-1}(z)$ with $s_{k-1} \in \mathcal{P}_{k-1}$, then the orthogonality condition (8.13) yields

$$\begin{aligned}
\|A^k v_1 - s(A) v_1\|_2^2 &= \| \underbrace{(A^k v_1 - s_{k-1}(A) v_1)}_{= q_k(A)v_1 \in \mathcal{K}_k(A,v_1)^\perp} + \underbrace{(s_{k-1}(A) v_1 - s(A) v_1)}_{\in \mathcal{K}_k(A,v_1)} \|_2^2 \\
&= \|q_k(A) v_1\|_2^2 + \|s_{k-1}(A) v_1 - s(A) v_1\|_2^2 \\
&\geq \|q_k(A) v_1\|_2^2,
\end{aligned}$$

with equality if $s = s_{k-1}$. (Actually, "if and only if", since we assume $k < d(A, v_1)$.) □

Note that the "optimality property" in Theorem 8.13 also applies to any Galerkin projection method with $\mathcal{S}_k = \mathcal{K}_k(A, v_1)$. However, it does not immediately lead to a convergence bound for these methods.
For a numerical illustration of the behavior of the Arnoldi method we consider two examples.

Example 8.14. *We first use a real* 250×250 *random matrix generated by* `A=randn(250)` *in MATLAB. The matrix A is diagonalizable with a modest eigenvector condition number; $\kappa(X) \approx 256.54$. In this case the bound in Corollary 8.12 can be used (a posteriori) for estimating how well Ritz values approximate eigenvalues of A. We apply the Arnoldi algorithm in the modified Gram-Schmidt variant and with a random initial vector generated by* `randn(250,1)` *in MATLAB. In Figure 8.3 we plot the eigenvalues of A by*

dots and the eigenvalues of H_k for $k = 10, 30, 50, 70$ by circles. The eigenvalues of A are (roughly) located in a disk centered at zero. We observe that the Ritz values tend to approximate the boundary of this disk. After 70 steps, most of the "outer" eigenvalues of A are approximated quite well by Ritz values.
The second matrix we consider is the Grcar matrix *generated by* `gallery('grcar',n,k)` *in MATLAB. This is an $n \times n$ upper Hessenberg matrix with values -1 on the subdiagonal and $+1$ on the diagonal and the first k superdiagonals. We use* `n=250` *and* `k=3`*, and obtain a matrix with $\kappa(A) \approx 3.6211$, which is diagonalizable but has severely ill-conditioned eigenvectors. A computation in MATLAB yields $\kappa(X) \approx 4.85 \times 10^{34}$. Clearly, for such a matrix Corollary 8.12 is useless. We again apply the Arnoldi algorithm in the modified Gram-Schmidt variant with a random initial vector. In Figure 8.4 we plot the eigenvalues of A by pluses and the eigenvalues of H_k for $k = 10, 30, 50, 70$ by circles. In this example the Ritz values also appear to approximate some curve enclosing the eigenvalues of A. This curve, however, is quite far away from the spectrum, so that even after 70 steps no eigenvalue of A is closely approximated by any Ritz value.*

A heuristic explanation of these experimental results can be given using Theorem 8.13. If q_k denotes the characteristic polynomial of H_k, then

$$\|q_k(A)v_1\|_2 = \min_{\substack{p \in \mathcal{P}_k \\ p \text{ monic}}} \|p(A)v_1\|_2.$$

In words, the polynomial q_k, whose k roots are the eigenvalues of H_k, minimizes $\|p(A)v_1\|_2$ over all monic polynomials p of degree k. If we disregard, for a moment, the initial vector v_1, we obtain the *ideal Arnoldi approximation problem*

$$\min_{\substack{p \in \mathcal{P}_k \\ p \text{ monic}}} \|p(A)\|_2; \tag{8.14}$$

cf. the ideal GMRES approximation problem in (4.29) and the corresponding bound in Theorem 4.15.
In the ideal Arnoldi problem, and hence (at least heuristically) in the actual Arnoldi method, we determine a (monic) polynomial q_k so that $\|q_k(A)\|$ is as small as possible. If A is diagonalizable, $A = X\Lambda X^{-1}$, then for each polynomial p we have

$$\|p(A)\|_2 = \|Xp(\Lambda)X^{-1}\|_2 \leq \kappa_2(X)\|p(\Lambda)\|_2.$$

If A is (close to) normal in the sense that $\kappa(X)$ is small, then a polynomial that is small at the eigenvalues of A guarantees that $\|p(A)\|_2$ is small. In this case the polynomial q_k typically has its roots close to the "outer" eigenvalues of A, as seen in Figure 8.3. If A is (highly) nonnormal, a small norm $\|p(\Lambda)\|_2$ may not be enough. Hence in such cases the roots of q_k may be quite far from the actual eigenvalues of A, as seen in Figure 8.4. A closer examination of matrix approximation problems like (8.14) is done in Chapter 10.

Finally, we will derive a bound on the (minimum) distance of an eigenvector of A to a Krylov subspace. For simplicity, let A be diagonalizable, $A = X \operatorname{diag}(\lambda_1, \ldots, \lambda_n)X^{-1}$, and let $v_1 = X[\alpha_1, \ldots, \alpha_n]^T$ with $\alpha_i \neq 0$ for some i.

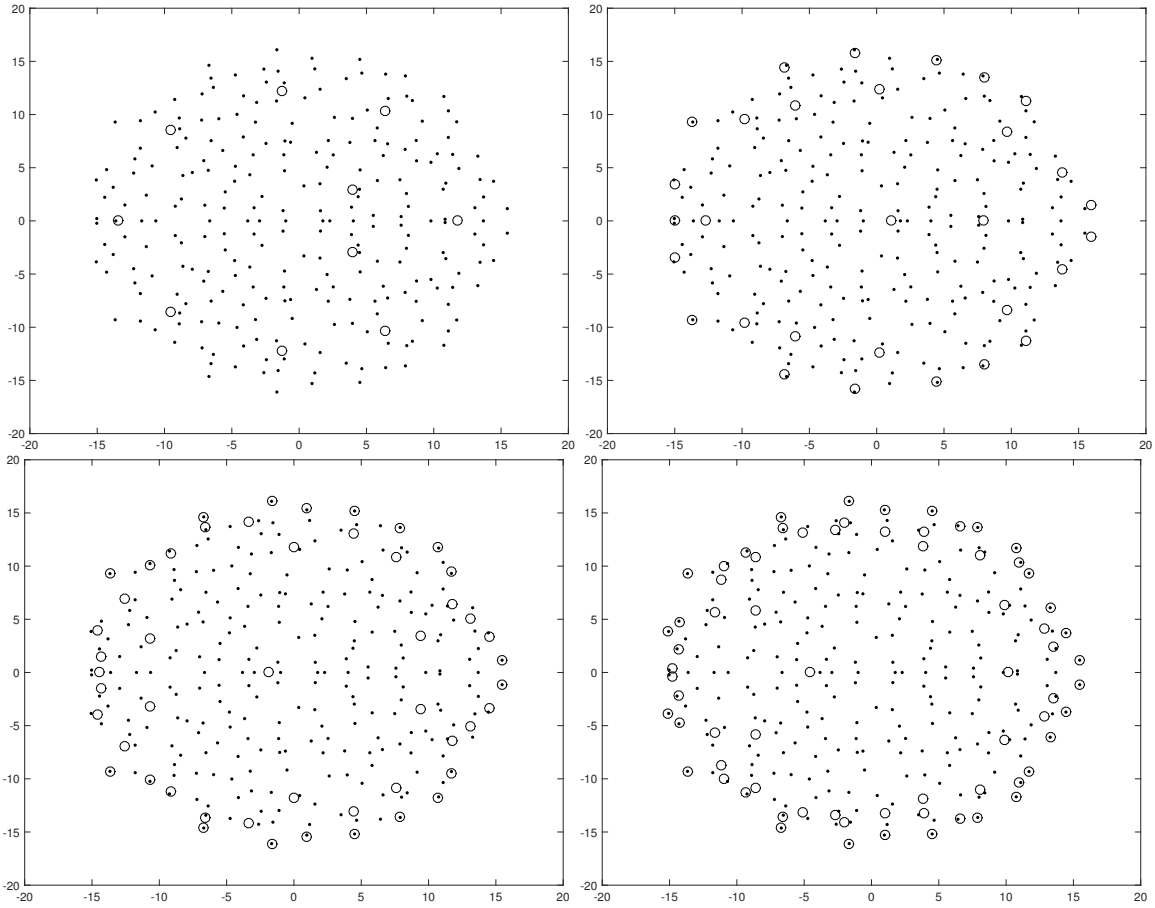


Figure 8.3: Eigenvalues of `A=randn(250)` (dots) and the corresponding eigenvalues of H_k (circles) for a random initial vector and $k = 10$ (top left), $k = 30$ (top right), $k = 50$ (bottom left), $k = 70$ (bottom right).

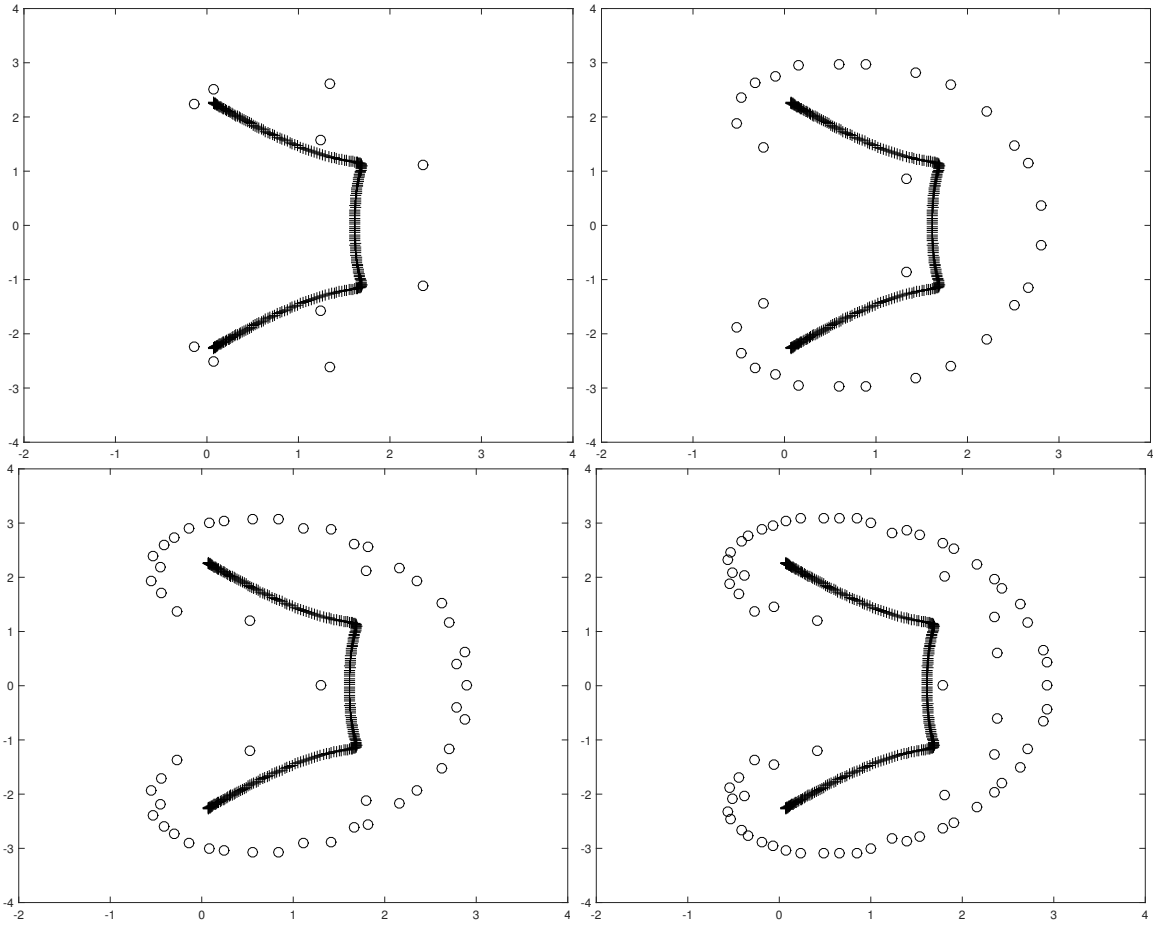


Figure 8.4: Eigenvalues of `A=gallery('grcar',250,3)` (pluses) and the corresponding eigenvalues of H_k (circles) for a random initial vector and $k = 10$ (top left), $k = 30$ (top right), $k = 50$ (bottom left), $k = 70$ (bottom right).

If the columns of V_k form an orthonormal basis of the Krylov subspace $\mathcal{K}_k(A, v_1) = \{p(A)v_1 : p \in \mathcal{P}_{k-1}\}$, then $P_k = V_k V_k^H$ is the corresponding orthogonal projection. Using Lemma 8.3 then we obtain

$$\begin{aligned}
\|(I_n - P_k)\alpha_i x_i\|_2 &= \min_{y \in \mathcal{K}_k(A,v_1)} \|\alpha_i x_i - y\|_2 \\
&= \min_{p \in \mathcal{P}_{k-1}} \|\alpha_i x_i - p(A)v_1\|_2 \\
&= \min_{p \in \mathcal{P}_{k-1}} \left\|\alpha_i x_i - \sum_{j=1}^{n} \alpha_j p(\lambda_j) x_j\right\|_2 \\
&\leq \min_{p \in \mathcal{P}_{k-1}(\lambda_i)} \left\| \sum_{\substack{j=1 \\ j \neq i}}^{n} \alpha_j p(\lambda_j) x_j\right\|_2 \\
&\leq \min_{p \in \mathcal{P}_{k-1}(\lambda_i)} \max_{\substack{\ell=1,\ldots,n \\ \ell \neq i}} |p(\lambda_\ell)| \left\| \sum_{\substack{j=1 \\ j \neq i}}^{n} \alpha_j x_j\right\|_2.
\end{aligned}$$

Assuming in addition that $\|x_j\|_2 = 1$ for $j = 1, \ldots, n$, and dividing by $|\alpha_i|$ yields the following result.

Theorem 8.15. *In the notation established above,*

$$\|(I_n - P_k)x_i\|_2 \leq \min_{p \in \mathcal{P}_{k-1}(\lambda_i)} \max_{\substack{\ell=1,\ldots,n \\ \ell \neq i}} |p(\lambda_\ell)| \sum_{\substack{j=1 \\ j \neq i}}^{n} \left|\frac{\alpha_j}{\alpha_i}\right|.$$

Thus, the distance of any given eigenvector x_i of A to the Krylov subspace $\mathcal{K}_k(A, v_1)$ is bounded by the value of a polynomial approximation problem on the (discrete) set $\{\lambda_1, \ldots, \lambda_n\} \setminus \{\lambda_i\}$, where the polynomials are normalized at the point λ_i. We can combine this result with Theorem 8.4 to obtain an alternative bound on the residual $A_k x_i - \lambda_i x_i$.

In the next chapter we will bound the "gap" between a desired eigenvector or invariant subspace of A and a computed Krylov subspace.

Chapter 9

Containment gap bounds for Krylov subspaces

As explained in Section 8.4, for a general nonnormal matrix $A = XJX^{-1}$ with a potentially large $\kappa(X)$, a small residual norm in the Arnoldi method does not guarantee that the Ritz values of method closely approximate any eigenvalue of A. Thus, the residual norm is in general not useful when analyzing the convergence or estimating the error in the Arnoldi method. Among the few alternative approaches to the convergence analysis of the Arnoldi method is the theory of Beattie, Embree and Sorensen [4]. They studied how well Krylov subspaces approximate an invariant subspace of A corresponding to "desired" eigenvalues. In this chapter we will derive the main results from [4]. We stress that results deal with the convergence of the Krylov subspaces, and thus are not tied to the Arnoldi method or to any other specific method.

9.1 The containment gap

The theory is based on the following concept.

Definition 9.1. *The* containment gap *between two subspaces $\mathcal{Y}, \mathcal{Z} \subseteq \mathbb{C}^n$ is given by*

$$\delta(\mathcal{Y}, \mathcal{Z}) := \max_{0 \neq y \in \mathcal{Y}} \min_{z \in \mathcal{Z}} \frac{\|y - z\|_2}{\|y\|_2}.$$

This definition can be interpreted as follows: The containment gap is the maximum distance of the unit norm vectors $y \in \mathcal{Y}$ to the set $\mathcal{Z}$. In particular, $0 \leq \delta(\mathcal{Y}, \mathcal{Z}) \leq 1$, and $\delta(\mathcal{Y}, \mathcal{Z}) = 0$ if and only if $\mathcal{Y} \subseteq \mathcal{Z}$. Thus, the containment gap can be interpreted as a measure for how well $\mathcal{Y}$ is contained in $\mathcal{Z}$.

Lemma 9.2. *If* $\dim(\mathcal{Y}) > \dim(\mathcal{Z})$, *then* $\delta(\mathcal{Y}, \mathcal{Z}) = 1$.

The proof of this lemma is left as an exercise.
Suppose that A has m *distinct* eigenvalues $\lambda_1, \ldots, \lambda_m$ and that we want to compute $\ell < m$ of them, say $\lambda_1, \ldots, \lambda_\ell$, which we call the *good eigenvalues*. The remaining $\lambda_{\ell+1}, \ldots, \lambda_m$

are called the *bad eigenvalues*. The (maximal) invariant subspaces corresponding to the good and bad eigenvalues are

$$\mathcal{X}_g := \bigoplus_{j=1}^{\ell} \ker(A - \lambda_j I)^{d_j} \quad \text{and} \quad \mathcal{X}_b := \bigoplus_{j=\ell+1}^{m} \ker(A - \lambda_j I)^{d_j},$$

respectively, where d_j is the index of λ_j, $j = 1, \ldots, m$. Clearly, $\mathbb{C}^n = \mathcal{X}_g \oplus \mathcal{X}_b$.

For a given unit norm initial vector $v_1 \in \mathbb{C}^n$ of grade d with respect to A we consider the (maximal) Krylov subspace

$$\mathcal{K}(A, v_1) := \mathcal{K}_d(A, v_1),$$

and we ask how well the invariant subspace $\mathcal{X}_g$ can possibly be approximated by vectors from $\mathcal{K}(A, v_1)$. Let us factor the minimal polynomial p_A of A as

$$p_A \;=\; p_g\, p_b,$$

where p_g and p_b have the good and the bad eigenvalues as roots, respectively. Since the two (non-constant) polynomials p_g and p_b have no common roots, the Lemma of Bézout implies that there exist polynomials q_g and q_b with

$$1 \;=\; p_g q_g + p_b q_b,$$

and hence $I = p_g(A)q_g(A) + p_b(A)q_b(A)$.

Lemma 9.3. *The matrices $P_g := p_b(A)q_b(A)$ and $P_b := p_g(A)q_g(A) = I_n - P_g$ satisfy*

$$\begin{aligned} &AP_g = P_g A, \quad P_g = P_g^2, \quad \operatorname{ran}(P_g) = \mathcal{X}_g, \quad \ker(P_g) = \mathcal{X}_b, \\ &AP_b = P_b A, \quad P_b = P_b^2, \quad \operatorname{ran}(P_b) = \mathcal{X}_b, \quad \ker(P_b) = \mathcal{X}_g. \end{aligned}$$

The proof of this lemma is left as an exercise.
Lemma 9.3 shows in particular that P_g and P_b are projections onto the good and bad invariant subspaces of A. In general these matrices are not Hermitian and hence the projection is oblique (and not orthogonal).

Example 9.4. *Consider a diagonalizable matrix $A = X\Lambda X^{-1} \in \mathbb{C}^{n\times n}$ with n distinct eigenvalues $\lambda_1, \ldots, \lambda_n$ and let $\ell = 1$. Then*

$$\mathcal{X}_g = \ker(A - \lambda_1 I_n) = \operatorname{span}\{x_1\} \quad \textit{and} \quad \mathcal{X}_b = \bigoplus_{j=2}^{n} \ker(A - \lambda_j I_n) = \operatorname{span}\{x_2, \ldots, x_n\},$$

$$p_g(\mu) = \mu - \lambda_1 \quad \textit{and} \quad p_b(\mu) = \prod_{j=2}^{n} (\mu - \lambda_j),$$

so that

$$P_g = p_b(A)q_b(A) = X p_b(\Lambda) X^{-1} X q_b(\Lambda) X^{-1} = X \operatorname{diag}(p_b(\lambda_1)q_b(\lambda_1), 0, \ldots, 0) X^{-1}.$$

Now $1 = p_g q_g + p_b q_b$ *and* $p_g(\lambda_1) = 0$ *imply that*

$$1 = p_g(\lambda_1)q_g(\lambda_1) + p_b(\lambda_1)q_b(\lambda_1) = p_b(\lambda_1)q_b(\lambda_1).$$

For $v_1 = \sum_{j=1}^{n} \alpha_j x_j = X[\alpha_1, \ldots, \alpha_n]^T$ *we therefore obtain*

$$\begin{aligned} P_g v_1 &= p_b(A)q_b(A)v_1 \\ &= X\mathrm{diag}(1, 0, \ldots, 0)X^{-1}X[\alpha_1, \ldots, \alpha_n]^T \\ &= X[\alpha_1, 0 \ldots, 0]^T \\ &= \alpha_1 x_1, \end{aligned}$$

which is the projection of v_1 *onto* $\mathcal{X}_g = \mathrm{span}\{x_1\}$. *On the other hand, for* $v_1 = X[0, \alpha_2, \ldots, \alpha_n]^T$ *we obtain* $P_g v_1 = 0$.

Lemma 9.5. *In the previous notation,*

$$\mathcal{K}(A, v_1) = \mathcal{K}(A, P_g v_1) \oplus \mathcal{K}(A, P_b v_1).$$

Proof. Using Lemma 9.3 we see that

$$P_g v_1 \in \mathcal{K}(A, P_g v_1) = P_g \mathcal{K}(A, v_1) \subseteq \mathrm{ran}(P_g) = \mathcal{X}_g$$

and

$$P_b v_1 \in \mathcal{K}(A, P_b v_1) = P_b \mathcal{K}(A, v_1) \subseteq \mathrm{ran}(P_b) = \mathcal{X}_b.$$

Moreover, $\mathcal{X}_g \cap \mathcal{X}_b = \{0\}$ by construction.
If $x \in \mathcal{K}(A, v_1)$ we can write $x = p(A)v_1$ for some polynomial p and hence

$$x = p(A)(P_g + P_b)v_1 = p(A)P_g v_1 + p(A)P_b v_1 \ \in \ \mathcal{K}(A, P_g v_1) \oplus \mathcal{K}(A, P_b v_1).$$

On the other hand, if $x \in \mathcal{K}(A, P_g v_1) \oplus \mathcal{K}(A, P_b v_1)$, then there exist polynomials s_g and s_b such that

$$x = s_g(A)P_g v_1 + s_b(A)P_b v_1 = (s_g(A)P_g + s_b(A)P_b)v_1 \ \in \ \mathcal{K}(A, v_1).$$

□

Lemma 9.5 shows that the subspace

$$\mathcal{U}_g := \mathcal{K}(A, P_g v_1) \subseteq \mathcal{X}_g$$

is the *maximal reachable* (good) invariant subspace for A and the intial vector v_1. The other subspace in the decomposition of $\mathcal{K}(A, v_1)$,

$$\mathcal{U}_b := \mathcal{K}(A, P_b v_1) \subseteq \mathcal{X}_b,$$

is the complementary maximal reachable invariant subspace. Ideally, we would like to have $\mathcal{U}_g = \mathcal{X}_g$, which would give $\delta(\mathcal{X}_g, \mathcal{K}(A, v_1)) = 0$. But since v_1 may be deficient in some (generalized) eigenvectors corresponding to a good eigenvalue, we may have $\mathcal{X}_g \setminus \mathcal{U}_g \neq \emptyset$, and thus $\delta(\mathcal{X}_g, \mathcal{K}(A, v_1)) > 0$.

Example 9.6. *(Continuation of Example 9.4.) If $\alpha_1 = 0$, then $P_g v_1 = \alpha_1 x_1 = 0$, giving $\mathcal{U}_g = \mathcal{K}(A, P_g v_1) = \{0\}$, and hence $\mathcal{X}_g \setminus \mathcal{U}_g \neq \emptyset$. On the other hand, if $\alpha_1 \neq 0$, then $P_g v_1 \neq 0$ and hence $\mathcal{U}_g = \mathcal{X}_g = \operatorname{span}\{x_1\}$.*

We can give a lower bound on the containment gap $\delta(\mathcal{X}_g, \mathcal{K}(A, v_1))$.

Lemma 9.7. *If $\mathcal{X}_g \setminus \mathcal{U}_g \neq \emptyset$, then $\delta(\mathcal{X}_g, \mathcal{K}(A, v_1)) \geq \|P_g\|_2^{-1}$.*

Proof. Since $\mathcal{X}_g \setminus \mathcal{U}_g \neq \emptyset$, there exists a nonzero vector $z \in \mathcal{X}_g \setminus \mathcal{U}_g$ with $z \perp \mathcal{U}_g$. Let this vector z be fixed. Then for any $x_g \in \mathcal{U}_g$ we have $\|z - x_g\|_2^2 = \|z\|_2^2 + \|x_g\|_2^2$, which implies $\|z - x_g\|_2 \geq \|z\|_2$.
Below we also use that every vector $x \in \mathcal{K}(A, v_1)$ can be uniquely decomposed as $x = x_g + x_b$ with $x_g \in \mathcal{U}_g$ and $x_b \in \mathcal{U}_b$; see Lemma 9.5. Thus,

$$\begin{aligned}
\delta(\mathcal{X}_g, \mathcal{K}(A, v_1)) &= \max_{0 \neq y \in \mathcal{X}_g} \min_{x \in \mathcal{K}(A, v_1)} \frac{\|y - x\|_2}{\|y\|_2} \geq \min_{x \in \mathcal{K}(A, v_1)} \frac{\|z - x\|_2}{\|z\|_2} \\
&= \min_{x_g \in \mathcal{U}_g, x_b \in \mathcal{U}_b} \frac{\|z - x_g - x_b\|_2}{\|z\|_2} \geq \min_{x_g \in \mathcal{U}_g, x_b \in \mathcal{U}_b} \frac{\|z - x_g - x_b\|_2}{\|z - x_g\|_2} \\
&= \min_{x_g \in \mathcal{U}_g, x_b \in \mathcal{U}_b} \frac{\|z - x_g - x_b\|_2}{\|P_g(z - x_g - x_b)\|_2} \\
&\geq \min_{0 \neq y \in \mathbb{C}^n} \frac{\|y\|_2}{\|P_g y\|_2} = \|P_g\|_2^{-1}.
\end{aligned}$$

□

Note that the matrix P_g and hence the lower bound on $\delta(\mathcal{X}_g, \mathcal{K}(A, v_1))$ are independent of the initial vector v_1.

9.2 Upper bounds for Krylov subspaces

Our next goal is to find an upper bound on the containment gap between the maximal reachable subspace $\mathcal{U}_g$ and the Krylov subspaces $\mathcal{K}_k(A, v_1)$. Let us define

$$\eta := \dim(\mathcal{U}_g).$$

In light of Lemma 9.2 we can assume that

$$\dim(\mathcal{U}_g) = \eta \leq k = \dim(\mathcal{K}_k(A, v_1)).$$

Each $y \in \mathcal{U}_g$ is of the form $y = q(A) P_g v_1$ for some polynomial $q \in \mathcal{P}_{\eta-1}$, and each $x \in \mathcal{K}_k(A, v_1)$ is of the form $x = p(A) v_1$ for some polynomial $p \in \mathcal{P}_{k-1}$. Using $P_g + P_b = I_n$

we therefore get

$$\begin{aligned}\delta(\mathcal{U}_g, \mathcal{K}_k(A, v_1)) &= \max_{0\neq q\in\mathcal{P}_{\eta-1}} \min_{p\in\mathcal{P}_{k-1}} \frac{\|p(A)v_1 - q(A)P_g v_1\|_2}{\|q(A)P_g v_1\|_2} \\ &= \max_{0\neq q\in\mathcal{P}_{\eta-1}} \min_{p\in\mathcal{P}_{k-1}} \frac{\|p(A)P_b v_1 + (p(A) - q(A))P_g v_1\|_2}{\|q(A)P_g v_1\|_2} \\ &\leq \max_{0\neq q\in\mathcal{P}_{\eta-1}} \min_{p\in\mathcal{P}^q_{k-1}} \frac{\|p(A)P_b v_1\|_2}{\|q(A)P_g v_1\|_2}, \end{aligned} \tag{9.1}$$

where for each given $q \in \mathcal{P}_{\eta-1}$ we define

$$\mathcal{P}^q_{k-1} := \{p \in \mathcal{P}_{k-1} \,:\, p(A)P_g v_1 = q(A)P_g v_1\}.$$

We can prove the following characterization of this set.

Lemma 9.8. *If $k \geq \eta$, then for any given $q \in \mathcal{P}_{\eta-1}$ we have*

$$\mathcal{P}^q_{k-1} = \{q - p\cdot\widetilde{p}_g \,:\, p \in \mathcal{P}_{k-\eta-1}\},$$

where $\widetilde{p}_g$ is the minimal polynomial of $P_g v_1$ with respect to A, i.e., the monic polynomial of lowest degree with $\widetilde{p}_g(A)P_g v_1 = 0$. (For $k = \eta$ we set $P_{-1} = \{0\}$.)

Proof. The polynomial $\widetilde{p}_g$ is defined as follows: Since $\mathcal{U}_g = \mathcal{K}(A, P_g v_1)$ and $\dim(\mathcal{U}_g) = \eta$, we know that

$$P_g v_1, AP_g v_1, \ldots, A^{\eta-1}P_g v_1$$

are linearly independent, and

$$P_g v_1, AP_g v_1, \ldots, A^{\eta-1}P_g v_1, A^{\eta}P_g v_1$$

are linearly dependent. Thus,

$$A^{\eta}P_g v_1 = \sum_{j=0}^{\eta-1} \alpha_j A^j P_g v_1, \quad \text{or} \quad \widetilde{p}_g(A)P_g v_1 = 0, \quad \text{where} \quad \widetilde{p}_g(\mu) := \mu^{\eta} - \sum_{j=0}^{\eta-1} \alpha_j \mu^j$$

is the monic polynomial of lowest degree with $\widetilde{p}_g(A)P_g v_1 = 0$.
If s is any polynomial with $s(A)P_g v_1 = 0$, then the polynomial division with remainder yields

$$s = \widetilde{q}\cdot\widetilde{p}_g + r$$

for some polynomials $\widetilde{q}$ and r (called the quotient and the remainder of the division of s by $\widetilde{p}_g$) with $\deg(r) < \deg(\widetilde{p}_g)$. Thus,

$$0 = s(A)P_g v_1 = \widetilde{q}(A)\widetilde{p}_g(A)P_g v_1 + r(A)P_g v_1 = r(A)P_g v_1.$$

The minimality assumption on $\deg(\widetilde{p}_g)$ shows that $r = 0$, and thus every nonzero polynomial s with $s(A)P_g v_1 = 0$ is a (polynomial) multiple of the minimal polynomial $\widetilde{p}_g$.

Finally, let $q \in \mathcal{P}_{\eta-1}$ be given. By construction, each element $\widehat{q} \in \mathcal{P}_{k-1}^q$ satisfies $(q(A) - \widehat{q}(A))P_g v_1 = 0$, and hence $q - \widehat{q}$ is a (polynomial) multiple of $\widetilde{p}_g$, i.e., $q - \widehat{q} = p \cdot \widetilde{p}_g$ for some polynomial p. Since $\deg(\widetilde{p}_g) = \dim(\mathcal{U}_g) = \eta$ we must have $\deg(p) \leq k - \eta - 1$. On the other hand, if $\widehat{q} = q - p \cdot \widetilde{p}_g$ for some $p \in \mathcal{P}_{k-\eta-1}$, then $\deg(\widehat{q}) \leq k-1$ and $\widehat{q}(A)P_g v_1 = q(A)P_g v_1$, so that $\widehat{q} \in \mathcal{P}_{k-1}^q$. □

Example 9.9. *(Continuation of Example 9.6.) If v_1 is given with $\alpha_1 \neq 0$, then $P_g v_1 = X[\alpha_1, 0, \ldots, 0]^T \neq 0$, and $\widetilde{p}_g = p_g = \mu - \lambda_1$, since*

$$\widetilde{p}_g(A)P_g v_1 = (A - \lambda_1 I_n)P_g v_1 = X\mathrm{diag}(0, \lambda_2 - \lambda_1, \ldots, \lambda_n - \lambda_1)X^{-1}X[\alpha_1, 0, \ldots, 0]^T = 0.$$

Obviously, there is no monic polynomial of lower degree that annihilates the nonzero vector $P_g v_1$.

We can now state and prove a variant of [4, Theorem 3.3].

Theorem 9.10. *Let $A \in \mathbb{C}^{n\times n}$ and unit norm initial vector $v_1 \in \mathbb{C}^n$ be given. Suppose that $k \geq 2\eta$, where $\eta := \dim(\mathcal{U}_g)$, and denote by Π_b the orthogonal projection onto $\mathcal{U}_b$. Then*

$$\delta(\mathcal{U}_g, \mathcal{K}_k(A, v_1)) \leq \left(\min_{p\in\mathcal{P}_{k-2\eta}} \|(I_n - p(A)\widetilde{p}_g(A))\Pi_b\|_2 \right) \left(\max_{0\neq q\in\mathcal{P}_{\eta-1}} \frac{\|q(A)P_b v_1\|_2}{\|q(A)P_g v_1\|_2} \right). \tag{9.2}$$

Proof. Using the characterization of Lemma 9.8 in (9.1) gives

$$\begin{aligned}\delta(\mathcal{U}_g, \mathcal{K}_k(A, v_1)) &\leq \max_{0\neq q\in\mathcal{P}_{\eta-1}} \min_{p\in\mathcal{P}_{k-1}^q} \frac{\|p(A)P_b v_1\|_2}{\|q(A)P_g v_1\|_2} \\ &= \max_{0\neq q\in\mathcal{P}_{\eta-1}} \min_{p\in\mathcal{P}_{k-\eta-1}} \frac{\|(q(A) - p(A)\widetilde{p}_g(A))P_b v_1\|_2}{\|q(A)P_g v_1\|_2}.\end{aligned}$$

Here we have a minimization over the set $\mathcal{P}_{k-\eta-1}$. We can get an upper bound by considering only polynomials of the form $p \cdot q$ for the polynomial $q \in \mathcal{P}_{\eta-1}$ that occurs in the maximization. If $\deg(p \cdot q) = k - \eta - 1$ and $\deg(q) = \eta - 1$, then $\deg(p) = k - 2\eta$, which is assumed nonnegative. Thus, for $k \geq 2\eta$ we obtain

$$\begin{aligned}\min_{p\in\mathcal{P}_{k-\eta-1}} \|(q(A) - p(A)\widetilde{p}_g(A))P_b v_1\|_2 &\leq \min_{p\in\mathcal{P}_{k-2\eta}} \|(q(A) - q(A)p(A)\widetilde{p}_g(A))P_b v_1\|_2 \\ &= \min_{p\in\mathcal{P}_{k-2\eta}} \|(I_n - p(A)\widetilde{p}_g(A))\Pi_b q(A)P_b v_1\|_2 \\ &\leq \left(\min_{p\in\mathcal{P}_{k-2\eta}} \|(I_n - p(A)\widetilde{p}_g(A))\Pi_b\|_2 \right) \|q(A)P_b v_1\|_2,\end{aligned}$$

which implies (9.2). □

Let us continue the development made in the examples above for a *normal* matrix $A = X\Lambda X^{-1}$, where $X^{-1} = X^H$, that has a simple eigenvalue λ_1, which is the only good eigenvalue, i.e., $\ell = 1$. With $\Omega_b = \{\lambda_2, \ldots, \lambda_n\}$ we get

$$\begin{aligned}\min_{p\in\mathcal{P}_{k-2\eta}} \|(I_n - p(A)\widetilde{p}_g(A))\Pi_b\|_2 &= \min_{p\in\mathcal{P}_{k-2}} \|X(I_n - p(\Lambda)\widetilde{p}_g(\Lambda))X^H\Pi_b\|_2 \\ &= \min_{p\in\mathcal{P}_{k-2}} \max_{2\le j\le n} |1 - p(\lambda_j)(\lambda_j - \lambda_1)| \\ &= \min_{p\in\mathcal{P}_{k-2}} \max_{\mu\in\Omega_b} |1 - p(\mu)(\mu - \lambda_1)| \\ &= \min_{p\in\mathcal{P}_{k-1}(\lambda_1)} \max_{\mu\in\Omega_b} |p(\mu)|.\end{aligned}$$

This is the same polynomial approximation problem as in Theorem 8.15 and in the bound (8.9) for the Lanczos method. Note, however, the squares on the right hand side of (8.9). Moreover,

$$P_g v_1 = X[\alpha_1, 0, \ldots, 0]^T \quad \text{and} \quad P_b v_1 = X[0, \alpha_2, \ldots, \alpha_n]^T.$$

Assuming, without loss of generality, that x_1 and v_1 have unit norm and that $v_1^H x_1 \neq 0$ or, equivalently, $\alpha_1 \neq 0$, we get $\mathcal{U}_g = \mathcal{K}(A, P_g v_1) = \operatorname{span}\{x_1\} = \mathcal{X}_g$ and

$$\begin{aligned}\max_{0\neq q\in\mathcal{P}_{\eta-1}} \frac{\|q(A)P_b v_1\|_2}{\|q(A)P_g v_1\|_2} = \max_{0\neq q\in\mathcal{P}_0} \frac{\|q(A)P_b v_1\|_2}{\|q(A)P_g v_1\|_2} = \max_{c\neq 0} \frac{\|cP_b v_1\|_2}{\|cP_g v_1\|_2} &= \left(\frac{\sum_{j=2}^n |\alpha_j|^2}{|\alpha_1|^2}\right)^{1/2} \\ = \frac{\sin\theta(x_1, v_1)}{\cos\theta(x_1, v_1)}.\end{aligned}$$

In summary, we have shown the following result.

Corollary 9.11. *Suppose that A is normal with a simple eigenvalue λ_1 and a corresponding unit norm eigenvector x_1. If v_1 is a unit norm vector with $v_1^H x_1 \neq 0$, then, in the notation established above, for all $k \geq 2$ we have*

$$\delta(\operatorname{span}\{x_1\}, \mathcal{K}_k(A, v_1)) \leq \frac{\sin\theta(x_1, v_1)}{\cos\theta(x_1, v_1)} \min_{p\in\mathcal{P}_{k-1}(\lambda_1)} \max_{\mu\in\Omega_b} |p(\mu)|.$$

This result holds in particular for Hermitian matrices. Note that the right hand side is essentially the square root of the right hand side in the bound (8.9) for the convergence of the largest Ritz value in the Lanczos method. This is consistent with the observation made by comparing Theorems 6.7 and 6.8: The error bound for the eigenvalue approximation is the square of the error bound for the eigenvector approximation (also cf. Theorem 7.1 for the power method).

For nonnormal matrices an analysis of the minimization problem in (9.2), i.e.,

$$\min_{p\in\mathcal{P}_{k-2\eta}} \|(I_n - p(A)\widetilde{p}_g(A))\Pi_b\|_2$$

is much more challenging. Beattie, Embree and Sorensen [4] introduced a clever trick for finding a bound on the value of this minimization problem in case of a general (nonnormal, non-diagonalizable) matrix. In order to explain this trick, we need the following definition.

Definition 9.12. *Let $A \in \mathbb{C}^{n\times n}$ have the Jordan decomposition $A = XJX^{-1}$ with*

$$J = \operatorname{diag}\big(J_{d_1}(\lambda_1), \ldots, J_{d_m}(\lambda_m)\big),$$

and let $\widetilde{\lambda}_1, \ldots, \widetilde{\lambda}_k$ be the distinct eigenvalues of A with their respective indices $\widetilde{d}_1, \ldots, \widetilde{d}_k$. A function f is defined on the spectrum of A, *if the values*

$$f^{(j)}(\widetilde{\lambda}_i), \quad i = 1, \ldots, k, \quad j = 0, 1, \ldots, \widetilde{d}_i - 1, \tag{9.3}$$

are defined[1]*. If f is defined on the spectrum of A, we define*

$$f(A) := Xf(J)X^{-1}, \quad \text{where} \quad f(J) := \operatorname{diag}\big(f(J_{d_1}(\lambda_1)), \ldots, f(J_{d_m}(\lambda_m))\big) \tag{9.4}$$

$$f(J_{d_i}(\lambda_i)) := \begin{bmatrix} f(\lambda_i) & f'(\lambda_i) & \cdots & \frac{f^{(d_i-1)}(\lambda_i)}{(d_i-1)!} \\ & \ddots & \ddots & \vdots \\ & & f(\lambda_i) & f'(\lambda_i) \\ & & & f(\lambda_i) \end{bmatrix}, \quad i = 1, \ldots, m. \tag{9.5}$$

In the literature, this definition of $f(A)$ is sometimes called the *primary matrix function*. The definition will be studied in more detail in Section 9.3 below.
Using this definition of $f(A)$, Beattie, Embree and Sorensen [4] defined the constant $\kappa(\Omega_b)$ as the smallest positive number such that

$$\|f(A)\Pi_b\|_2 \le \kappa(\Omega_b) \max_{z\in\Omega_b} |f(z)|$$

holds for all functions f that are analytic (and hence arbitrarily often differentiable) on Ω_b. Using this constant we obtain

$$\begin{aligned} \min_{p\in\mathcal{P}_{k-2\eta}} \|(I_n - p(A)\widetilde{p}_g(A))\Pi_b\|_2 &\le \kappa(\Omega_b) \min_{p\in\mathcal{P}_{k-2\eta}} \max_{z\in\Omega_b} |1 - p(z)\widetilde{p}_g(z)| \\ &\le \kappa(\Omega_b) \max_{z\in\Omega_b} |\widetilde{p}_g(z)| \min_{p\in\mathcal{P}_{k-2\eta}} \max_{z\in\Omega_b} \left| \frac{1}{\widetilde{p}_g(z)} - p(z) \right|, \end{aligned}$$

where in the second inequality we have used that $\widetilde{p}_g(z) \neq 0$ on Ω_b. The value of the min-max approximation problem depends on the distance of the good eigenvalues, which are the poles of the rational function $1/\widetilde{p}_g(z)$, to the set Ω_b. For a computable estimate one can replace the discrete set Ω_b by a nondiscrete set (e.g. the convex hull of the points in Ω_b) and use conformal mapping techniques. Details are beyond the course Numerical Linear Algebra, and interested students should look at [4, Section 3.4]. For an estimation of the constant $\kappa(\Omega_b)$ using pseudospectra of A see [4, Section 3.2].

[1] Here $f^{(j)}(\widetilde{\lambda}_i)$ is the value of the jth derivative of f at $\widetilde{\lambda}_i$.

9.3 More on functions of matrices

There are many different ways to define functions of matrices. For example, the determinant is a function that maps square real or complex matrices into the fields of real or complex numbers, or the rank is a function that maps rectangular real or complex matrices into the set of nonnegative integers.
Here we are interested in a definition that for a given $A \in \mathbb{C}^{n\times n}$ yields $f(A) \in \mathbb{C}^{n\times n}$. One possibility in this context is to define $f(A)$ by the entrywise application of f to $A = [a_{ij}]$, i.e., $f(A) := [f(a_{ij})]$. This definition, however, is not compatible with the matrix multiplication as shown by the example $A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$ and $f(z) = z^2$, which gives

$$\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} f(1) & f(1) \\ f(0) & f(1) \end{bmatrix} \neq A^2 = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix}.$$

From now on we will consider the primary matrix function $f(A)$ from Definition 9.12. First note that the eigenvalues of $f(A)$ are simply given by the values $f(\lambda_i)$, where the λ_i are the eigenvalues of A. However, the Jordan structure of $f(A)$ may be different from the Jordan structure of A.

Example 9.13. *If*

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

and $f(z) = \cos(z)$, then f is defined on the spectrum of A and

$$f(A) = \begin{bmatrix} f(\pi) & f'(\pi) \\ 0 & f(\pi) \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}.$$

Thus, $f(A)$ is diagonal, while A is not even diagonalizable.

If a given function f that is defined on the spectrum of A has several branches, we must use a single fixed branch for all Jordan blocks of A in order to be consistent with Definition 9.12.

Example 9.14. *Consider*

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

and the square root function $f(z) = z^{1/2}$. This function is defined on the spectrum of A and we can use either $f(1) = 1$ or $f(1) = -1$. If we denote the corresponding functions by f_1 and f_2, respectively, then

$$f_1(A) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \text{and} \quad f_2(A) = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$$

are consistent with Definition 9.12. On the other hand, the matrices

$$A_1 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad \text{and} \quad A_2 = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$$

satisfy $A_1^2 = A$ and $A_2^2 = A$. The matrices A_1 and A_2 can therefore be considered square roots of A, but they are not consistent with Definition 9.12.

Consider a single Jordan block $J_d(\lambda)$ and the function $f(z) = z^k$ for a given $k \geq 0$. Then we have

$$\begin{aligned} J_d(\lambda)^k = (\lambda I_d + J_d(0))^k &= \sum_{j=0}^{k} \binom{d}{j} \lambda^{d-j} J_d(0)^j = \sum_{j=0}^{k} \frac{\frac{k!}{(k-j)!}\lambda^{k-j}}{j!} J_d(0)^j \\ &= \sum_{j=0}^{k} \frac{f^{(j)}(\lambda)}{j!} J_d(0)^j = f(J_d(\lambda)). \end{aligned}$$

This shows, in particular, that the definition of $f(A)$ in (9.4)–(9.5) is compatible with the matrix multiplication: If $f(z) = z^2$, then the *definition* of $f(A)$ gives the same result as the *evaluation* of the product $A \cdot A = A^2$. More generally, we obtain the following result.

Lemma 9.15. *Let $A \in \mathbb{C}^{n\times n}$ and $f(z) = \alpha_k z^k + \cdots + \alpha_1 z + \alpha_0$ be a given polynomial. Then $f(A)$ as defined in (9.4)–(9.5) is equal to $\alpha_k A^k + \cdots + \alpha_1 A + \alpha_0 I_n$.*

The proof of this lemma is left as an exercise.
The definition of $f(A)$ only requires the existence of the function values in (9.3). It can be shown that there exists a uniquely determined polynomial p of degree at most $\sum_{i=1}^{k} \widetilde{d}_i - 1$ that solves the *Hermite interpolation problem*

$$p^{(j)}(\widetilde{\lambda}_i) = f^{(j)}(\widetilde{\lambda}_i), \quad i = 1, \ldots, k, \quad j = 0, 1, \ldots, \widetilde{d}_i - 1. \tag{9.6}$$

This polynomial satisfies $p(A) = f(A)$, and hence we obtain the following result.

Theorem 9.16. *Let $A \in \mathbb{C}^{n\times n}$ and let f be defined on the spectrum of A. Then $f(A) = p(A)$ for a polynomial p, and hence*

(1) $f(A)A = Af(A)$,

(2) if $S \in \mathbb{C}^{n\times n}$ satisfies $AS = SA$, then $f(A)S = Sf(A)$,

(3) $f(A^T) = f(A)^T$,

(4) $f(SAS^{-1}) = Sf(A)S^{-1}$ for any nonsingular $S \in \mathbb{C}^{n\times n}$.

The proof of this theorem is left as an exercise.
Using Theorem 9.16 we will now show that the definition of $f(A)$ in Definition 9.12 is independent of the choice of the Jordan canonical form of A. We know that the Jordan canonical form of A is unique up to the order of the Jordan blocks. If

$$\begin{aligned} J &= \operatorname{diag}(J_{d_1}(\lambda_1), \ldots, J_{d_m}(\lambda_m)) = X^{-1}AX, \\ \widetilde{J} &= \operatorname{diag}(J_{\widetilde{d}_1}(\widetilde{\lambda}_1), \ldots, J_{\widetilde{d}_m}(\widetilde{\lambda}_m)) = \widetilde{X}^{-1}A\widetilde{X} \end{aligned}$$

are two Jordan canonical forms of A, then $\widetilde{J} = P^T J P$ for a permutation matrix $P \in \mathbb{R}^{n,n}$, where the matrices J and $\widetilde{J}$ are the same up to the order of diagonal blocks. Hence

$$\begin{aligned} f(J) &= \operatorname{diag}(f(J_{d_1}(\lambda_1)), \ldots, f(J_{d_m}(\lambda_m))) \\ &= P\left(P^T \operatorname{diag}(f(J_{d_1}(\lambda_1)), \ldots, f(J_{d_m}(\lambda_m)))P\right)P^T \\ &= P\left(\operatorname{diag}(f(J_{\widetilde{d}_1}(\widetilde{\lambda}_1)), \ldots, f(J_{\widetilde{d}_m}(\widetilde{\lambda}_m)))\right)P^T \\ &= P f(\widetilde{J}) P^T. \end{aligned}$$

Theorem 9.16 applied to the matrix J yields the existence of a polynomial p with $f(J) = p(J)$, and thus we get

$$\begin{aligned} f(A) &= X f(J) X^{-1} = X p(J) X^{-1} = p(A) = p(\widetilde{X}\widetilde{J}\widetilde{X}^{-1}) = \widetilde{X} P^T p(J) P \widetilde{X}^{-1} \\ &= \widetilde{X} P^T f(J) P \widetilde{X}^{-1} = \widetilde{X} f(\widetilde{J}) \widetilde{X}^{-1}. \end{aligned}$$

Chapter 10

Matrix approximation theory

In this chapter we study approximation problems involving matrices such as the ideal Arnoldi approximation problem (8.14), which can be written as

$$\min_{\substack{p\in\mathcal{P}_k\\ p \text{ monic}}} \|p(A)\|_2 = \min_{p\in\mathcal{P}_{k-1}} \|A^k - p(A)\|_2. \tag{10.1}$$

In this problem we seek a best approximation (with respect to the 2-norm) of A^k from the finite dimensional subspace $\operatorname{span}\{I, A, \ldots, A^{k-1}\}$.
We start with a closer look at the existence and uniqueness of solutions of best approximation problems in general vector spaces. In the following, $\mathcal{V}$ is a real or complex vector space, which may be infinite-dimensional.

Theorem 10.1. *Let $\|\cdot\|$ be a norm on $\mathcal{V}$. If $\mathcal{U} \subseteq \mathcal{V}$ is a finite-dimensional subspace, then for each given $v \in \mathcal{V}$ there exists at least one $u_* \in \mathcal{U}$ of best approximation, i.e., at least one $u_* \in \mathcal{U}$ with*

$$\|v - u_*\| = \inf_{u\in\mathcal{U}} \|v - u\|.$$

The proof of this well known theorem uses basic tools of analysis, in particular compactness. The theorem justifies, for example, that in ideal Arnoldi approximation problem we were allowed to write "min" instead of "inf". We will show below that the solution of the ideal Arnoldi approximation problem is uniquely determined. In general, the question of uniqueness depends on the given norm.

Example 10.2.

(1) Let $\mathcal{V} = \mathbb{R}^2$ with the norm $\|v\| = \max\{|v_1|, |v_2|\}$ for all $v = [v_1, v_2]^T \in \mathbb{R}^2$, and let $\mathcal{U} = \operatorname{span}\{e_1\}$. For the vector $e_2 \in \mathcal{V}$ we have

$$\min_{u\in\mathcal{U}} \|e_2 - u\| = \min_{\alpha\in\mathbb{R}} \|e_2 - \alpha e_1\| = \min_{\alpha\in\mathbb{R}} \max\{|\alpha|, 1\} \geq 1.$$

The minimum is attained when $|\alpha| \leq 1$. Thus, each vector in $\mathcal{U}$ of the form αe_1 with $|\alpha| \leq 1$ is a best approximation of $e_2 \in \mathcal{V}$.

(2) *If we consider* $\mathcal{V} = \mathbb{R}^2$ *with the 2-norm and again seek the best approximation of* $e_2 \in \mathcal{V}$ *from* $\mathcal{U} = \operatorname{span}\{e_1\}$*, then*

$$\min_{u\in\mathcal{U}} \|e_2 - u\|_2 = \min_{\alpha\in\mathbb{R}} \|e_2 - \alpha e_1\|_2 = \min_{\alpha\in\mathbb{R}} (1+\alpha^2)^{1/2}.$$

Now the minimum is attained only for $\alpha = 0$*, and hence* $u = 0$ *is the uniquely determined best approximation of* e_2 *from the subspace* $\mathcal{U}$*.*

In order to study conditions under which a best approximation is unique, we introduce the following definition.

Definition 10.3. *A vector space* $\mathcal{V}$ *with a norm* $\|\cdot\|$ *is called* strictly convex *if for all* $v, w \in \mathcal{V}$ *we have*

$$\|v\| = \|w\| = \frac{1}{2}\|v+w\| \quad \Rightarrow \quad v = w.$$

The vector space $\mathbb{R}^2$ with the norm $\|v\| = \max\{|v_1|, |v_2|\}$ is not strictly convex. For example, $v = e_1$ and $w = [1,1]^T$ satisfy $1 = \|v\| = \|w\| = \frac{1}{2}\|v+w\|$, but $v \neq w$.

Theorem 10.4. *If* $\mathcal{V}$ *is strictly convex and* $\mathcal{U} \subseteq \mathcal{V}$ *is a finite-dimensional subspace, then for every* $v \in \mathcal{V}$ *there exists a uniquely determined* $u_* \in \mathcal{U}$ *of best approximation.*

Proof. Since $\mathcal{U}$ is finite-dimensional, we know from Theorem 10.1 that for each given $v \in \mathcal{V}$ there exists at least one best approximation in $\mathcal{U}$. If $u_1, u_2 \in \mathcal{U}$ are two such best approximations with

$$\gamma = \|v - u_1\| = \|v - u_2\| = \min_{u\in\mathcal{U}} \|v - u\|,$$

then

$$\|v - \frac{1}{2}(u_1+u_2)\| \leq \frac{1}{2}\|v-u_1\| + \frac{1}{2}\|v-u_2\| = \gamma.$$

Since $\frac{1}{2}(u_1+u_2) \in \mathcal{U}$, we must have $\|v - \frac{1}{2}(u_1+u_2)\| = \gamma$. Thus,

$$\|v-u_1\| = \|v-u_2\| = \frac{1}{2}\|(v-u_1)+(v-u_2)\|,$$

and the strict convexity implies $v - u_1 = v - u_2$, and hence $u_1 = u_2$. □

The next result explains why we have a uniquely determined best approximation in (2) in Example 10.2.

Theorem 10.5. *If the norm on* $\mathcal{V}$ *is induced by an inner product, i.e., if* $\|\cdot\| = \langle\cdot,\cdot\rangle^{1/2}$*, then* $\mathcal{V}$ *is strictly convex.*

Proof. If the norm is induced by an inner product, then the *parallelogram law*

$$\|v+w\|^2 + \|v-w\|^2 = 2(\|v\|^2 + \|w\|^2)$$

holds for all $v, w \in \mathcal{V}$. For any given $v, w \in \mathcal{V}$, let

$$\gamma = \|v\| = \|w\| = \frac{1}{2}\|v+w\|.$$

Then the parallelogram law implies

$$\gamma^2 = \frac{1}{4}\|v+w\|^2 = \frac{1}{4}(2(\|v\|^2 + \|w\|^2) - \|v-w\|^2) = \gamma^2 - \frac{1}{4}\|v-w\|^2,$$

hence $\|v-w\| = 0$, or $v = w$. □

We will now consider the 2-norm on $\mathbb{C}^{n\times n}$, which is defined using the Euclidean inner product on $\mathbb{C}^n$,

$$\|A\|_2 = \max_{v\in\mathbb{C}^n\setminus\{0\}} \frac{\|Av\|_2}{\|v\|_2} = \max_{v\in\mathbb{C}^n\setminus\{0\}} \frac{\langle Av, Av\rangle^{1/2}}{\langle v, v\rangle^{1/2}}.$$

However, the matrix 2-norm is *not* induced by an inner product on $\mathbb{C}^{n\times n}$. This follows from the fact that $\mathcal{V} = \mathbb{C}^{n\times n}$ with the matrix 2-norm is *not strictly convex.*

Example 10.6. *For a simple example consider the matrices $A_1, A_2 \in \mathbb{C}^{n\times n}$ of the form*

$$A_1 = \begin{bmatrix} B & 0 \\ 0 & C \end{bmatrix}, \quad A_2 = \begin{bmatrix} B & 0 \\ 0 & D \end{bmatrix},$$

and with $\|A_1\|_2 = \|A_2\|_2 = \|B\|_2 \geq \frac{1}{2}\|C+D\|_2$. Then

$$\|A_1\|_2 = \|A_2\|_2 = \frac{1}{2}\|A_1 + A_2\|_2 = \|B\|_2,$$

but whenever $C \neq D$ we have $A_1 \neq A_2$.

Despite the lack of strict convexity, many best approximation problems with respect to the matrix 2-norm have a uniquely determined solution. An important example is the problem

$$\min_{p\in\mathcal{P}_m} \|b(A) - p(A)\|_2, \tag{10.2}$$

where $b \in \mathcal{P}_{\ell+m+1}$ is a given polynomial (here $\ell, m \geq 0$ are given integers). Note that with $m = k-1$, $\ell = 0$ and $b(A) = A^k$, the problem (10.2) reduces to (10.1). Following [25] we will now show that the problem (10.2) has a uniquely determined solution.
Writing the given polynomial in the form

$$b = \sum_{j=0}^{\ell+m+1} \beta_j z^j \ \in \ \mathcal{P}_{\ell+m+1}, \quad \beta_{\ell+m+1} \neq 0.$$

the approximation problem (10.2) becomes

$$\begin{aligned}\min_{p\in\mathcal{P}_m} \|b(A)-p(A)\|_2 &= \min_{p\in\mathcal{P}_m} \left\| b(A) - \left(p(A) + \sum_{j=0}^{m} \beta_j A^j \right) \right\|_2 \\ &= \min_{p\in\mathcal{P}_m} \left\| \sum_{j=m+1}^{\ell+m+1} \beta_j A^j \; - \; p(A) \right\|_2 \\ &= \min_{p\in\mathcal{P}_m} \left\| A^{m+1} \sum_{j=0}^{\ell} \beta_{j+m+1} A^j \; - \; p(A) \right\|_2 . \end{aligned} \tag{10.3}$$

The polynomials in (10.3) are of the form $z^{m+1}g + h$, where the (nonzero) polynomial $g \in \mathcal{P}_\ell$ is given and $h \in \mathcal{P}_m$ is sought. Hence (10.2) is equivalent to the problem

$$\min_{p\in\mathcal{G}_{\ell,m}^{(g)}} \|p(A)\|_2, \quad \text{where } \mathcal{G}_{\ell,m}^{(g)} := \left\{ z^{m+1}g + h \mid h \in \mathcal{P}_m \right\}. \tag{10.4}$$

We can now state and prove [25, Theorem 2.2].

Theorem 10.7. *Let a matrix $A \in \mathbb{C}^{n\times n}$, integers $\ell, m \geq 0$, and a nonzero polynomial $g \in \mathcal{P}_\ell$ be given. If the value of* (10.4) *is positive, then this problem has a uniquely determined minimizer.*

Proof. We suppose that $q_1 = z^{m+1}g + h_1 \in \mathcal{G}_{\ell,m}^{(g)}$ and $q_2 = z^{m+1}g + h_2 \in \mathcal{G}_{\ell,m}^{(g)}$ are two distinct solutions to (10.4) and derive a contradiction. Suppose that the minimal norm attained by the two polynomials is

$$0 < \gamma = \|q_1(A)\|_2 = \|q_2(A)\|_2.$$

Define $q := \frac{1}{2}(q_1 + q_2) \in \mathcal{G}_{\ell,m}^{(g)}$, then

$$\|q(A)\|_2 \leq \frac{1}{2}(\|q_1(A)\|_2 + \|q_2(A)\|_2) = \gamma,$$

and hence $\|q(A)\|_2 = \gamma$. Denote an SVD of $q(A)$ by

$$q(A) = V \operatorname{diag}(\sigma_1, \ldots, \sigma_n) W^H. \tag{10.5}$$

Suppose that the maximal singular value $\sigma_1 = \gamma$ of $q(A)$ is J-fold, with left and right singular vectors given by $v_1, \ldots, v_J$ and $w_1, \ldots, w_J$, respectively.
We know from Theorem 10.5 that $\mathcal{V} = \mathbb{C}^n$ with the 2-norm is strictly convex. For each w_j, $1 \leq j \leq J$, we have

$$\gamma = \|q(A)w_j\|_2 \leq \frac{1}{2}\left(\|q_1(A)w_j\|_2 + \|q_2(A)w_j\|_2\right) \leq \gamma,$$

which implies

$$\|q_1(A)w_j\|_2 = \|q_2(A)w_j\|_2 = \gamma, \quad 1 \leq j \leq J,$$

and hence the strict convexity gives

$$q_1(A)w_j = q_2(A)w_j, \quad 1 \le j \le J.$$

Similarly, one can show that

$$q_1(A)^H v_j = q_2(A)^H v_j, \quad 1 \le j \le J.$$

Thus,

$$(q_2(A) - q_1(A))w_j = 0, \qquad (q_2(A) - q_1(A))^H v_j = 0, \qquad 1 \le j \le J. \tag{10.6}$$

By assumption, $q_2 - q_1 \in \mathcal{P}_m$ is a nonzero polynomial. By the division theorem for polynomials, there exist uniquely defined polynomials s and r, with $\deg(s) \le \ell + m + 1$ and $\deg(r) < \deg(q_2 - q_1) \le m$ (or $r = 0$), so that

$$z^{m+1} g = (q_2 - q_1) \cdot s + r.$$

Hence we have shown that for the given polynomials $q_2 - q_1$ and g there exist polynomials s and r such that

$$\widetilde{q} := (q_2 - q_1) \cdot s = z^{m+1} g - r \ \in \ \mathcal{G}_{\ell,m}^{(g)}.$$

Since $g \neq 0$, we must have $\widetilde{q} \neq 0$. For a fixed $\epsilon \in (0, 1)$, consider the polynomial

$$q_\epsilon \ = \ (1 - \epsilon) q + \epsilon \widetilde{q} \ \in \ \mathcal{G}_{\ell,m}^{(g)} .$$

By (10.6),

$$\widetilde{q}(A) w_j = 0, \quad \widetilde{q}(A)^H v_j = 0, \quad 1 \le j \le J,$$

and thus

$$\begin{aligned} q_\epsilon(A)^H q_\epsilon(A) w_j &= (1 - \epsilon) q_\epsilon(A)^H q(A) w_j = (1 - \epsilon)\gamma q_\epsilon(A)^H v_j \\ &= (1 - \epsilon)^2 \gamma q(A)^H v_j = (1 - \epsilon)^2 \gamma^2 w_j, \end{aligned}$$

which shows that $w_1, \ldots, w_J$ are right singular vectors of $q_\epsilon(A)$ corresponding to the singular value $(1 - \epsilon)\gamma$. Note that $(1 - \epsilon)\gamma < \gamma$ since $\gamma > 0$.
Now there are two cases: Either $\|q_\epsilon(A)\|_2 = (1 - \epsilon)\gamma$, or $(1 - \epsilon)\gamma$ is not the largest singular value of $q_\epsilon(A)$. In the first case we have a contradiction to the fact that γ is the minimal value of (10.4). Therefore, the second case must hold. In that case, none of the vectors $w_1, \ldots, w_J$ correspond to the largest singular value of $q_\epsilon(A)$. Using this fact and the SVD (10.5), we get

$$\begin{aligned} \|q_\epsilon(A)\|_2 &= \|q_\epsilon(A) W\|_2 \\ &= \|q_\epsilon(A)[w_{J+1}, \ldots, w_n]\| \\ &= \|(1 - \epsilon) q(A)[w_{J+1}, \ldots, w_n] \ + \ \epsilon \widetilde{q}(A)[w_{J+1}, \ldots, w_n]\|_2 \\ &\le (1 - \epsilon) \, \|[v_{J+1}, \ldots, v_n] \operatorname{diag}(\sigma_{J+1}, \ldots, \sigma_n)\|_2 \ + \ \epsilon \|\widetilde{q}(A)[w_{J+1}, \ldots, w_n]\|_2 \\ &\le (1 - \epsilon)\sigma_{J+1} \ + \ \epsilon \|\widetilde{q}(A)[w_{J+1}, \ldots, w_n]\|_2. \end{aligned} \tag{10.7}$$

Note that the norm $\|\widetilde{q}(A)[w_{J+1},\ldots,w_n]\|_2$ in (10.7) does not depend on the choice of ϵ and that (10.7) goes to σ_{J+1} as ϵ goes to zero. Since $\sigma_J > \sigma_{J+1}$, one can find a positive $\epsilon_* \in (0,1)$ such that (10.7) is less than σ_J for all $\epsilon \in (0,\epsilon_*)$. Any of the corresponding polynomials q_ϵ gives a matrix $q_\epsilon(A)$ whose norm is less than σ_J. This contradiction finishes the proof. □

Similarly to (10.4) we may consider, for a given nonzero polynomial $h \in \mathcal{P}_m$, the approximation problem

$$\min_{p \in \mathcal{H}_{\ell,m}^{(h)}} \|p(A)\|_2, \quad \text{where} \quad \mathcal{H}_{\ell,m}^{(h)} := \left\{ z^{m+1} g + h \mid g \in \mathcal{P}_\ell \right\}. \tag{10.8}$$

Setting $m = 0$ and $h = 1$ in (10.8), we obtain the problem

$$\min_{p \in \mathcal{P}_\ell} \|I_n - Ap(A)\|_2,$$

which occurs in the analysis of the GMRES method; see (4.29). As shown in [25, Theorem 2.3], the approximation problem (10.8) has a unique solution when its value is nonzero and A is nonsingular. When A is singular, however, the problem (10.8) might have more than one solution even when its value is positive.

Example 10.8. *Consider a normal matrix $A = U\Lambda U^H$, where $U^H U = I$ and $\Lambda = \operatorname{diag}(\lambda_1,\ldots,\lambda_n)$. Suppose that A is singular with n distinct eigenvalues, and $\lambda_1 = 0$. Furthermore, suppose that $h \in \mathcal{P}_m$ is any given polynomial that satisfies $|h(0)| > |h(\lambda_j)|$ for $j = 2,\ldots,n$. Then for* any *integer $\ell \geq 0$,*

$$\begin{aligned}
\min_{p \in \mathcal{H}_{\ell,m}^{(h)}} \|p(A)\|_2 &= \min_{g \in \mathcal{P}_\ell} \max_{1 \leq j \leq n} \left| \lambda_j^{m+1} g(\lambda_j) + h(\lambda_j) \right| \\
&= \min_{g \in \mathcal{P}_\ell} \max \left\{ |h(0)|, \max_{2 \leq j \leq n} \left| \lambda_j^{m+1} g(\lambda_j) + h(\lambda_j) \right| \right\} \\
&= |h(0)| > 0.
\end{aligned}$$

One solution of this problem is given by the polynomial $g = 0$. Moreover, the minimum value is attained for any polynomial $g \in \mathcal{P}_\ell$ that satisfies

$$\min_{g \in \mathcal{P}_\ell} \max_{2 \leq j \leq n} \left| \lambda_j^{m+1} g(\lambda_j) + h(\lambda_j) \right| \leq |h(0)|,$$

i.e., for any polynomial $g \in \mathcal{P}_\ell$ that is close enough to the zero polynomial.

The matrix approximation problem (10.2), which involves the best approximation of a given polynomial in A by polynomials in A, is more general than it may seem at first sight. In order to see this, recall from Theorem 9.16 that every well defined (primary) matrix function $f(A)$ is some polynomial in A We therefore have the following corollary of Theorem 10.7.

Corollary 10.9. *Let $A \in \mathbb{C}^{n\times n}$ and suppose that $f(A)$ is well defined in the sense of Definition 9.12. If $\min_{p\in\mathcal{P}_m} \|f(A) - p(A)\|_2 > 0$, then there exists a uniquely determined polynomial $p_* \in \mathcal{P}_m$ with*

$$\|f(A) - p_*(A)\|_2 = \min_{p\in\mathcal{P}_m} \|f(A) - p(A)\|_2.$$

The special matrix approximation problems discussed so far can be related to the general problem

$$\min_{p\in\mathcal{P}_m} \|f(A) - p(A)\|_2$$

for special choices of the function f. For example, $f(z) = z^{m+1}$ is defined on the spectrum of any matrix $A \in \mathbb{C}^{n\times n}$, and the resulting problem

$$\min_{p\in\mathcal{P}_m} \|A^{m+1} - p(A)\|_2$$

is the ideal Arnoldi approximation problem (8.14). If the value is positive and p_* is the unique solution, then $z^{m+1} - p_*(z)$ is called the $(m+1)st$ *Chebyshev polynomial of* A.
If A is nonsingular, then $f(z) = z^{-1}$ is defined on the spectrum of A, and

$$\min_{p\in\mathcal{P}_m} \|f(A) - p(A)\|_2 = \min_{p\in\mathcal{P}_m} \|A^{-1}(I_n - Ap(A))\|_2 \le \|A^{-1}\|_2 \min_{p\in\mathcal{P}_m} \|I_n - Ap(A)\|_2.$$

If $p_* \in \mathcal{P}_m$ is the unique solution (cf. the discussion following (10.8)), then $1 - zp_*(z) \in \mathcal{P}_{m+1}$ is the $(m+1)$st ideal GMRES polynomial of A.

In the analysis of matrix approximation problems we are of course not only interested in the existence and uniqueness of solutions, but also in estimating the value of the problem. Such estimates usually give convergence bounds for iterative methods; see, e.g., the bound on ideal GMRES using the field of values of A in Theorem 4.15.
In general, using the Jordan decompostion $A = XJX^{-1}$, where $J = \operatorname{diag}(J_{d_1}(\lambda_1), \ldots, J_{d_k}(\lambda_k))$, we have

$$\begin{aligned}\min_{p\in\mathcal{P}_m} \|f(A) - p(A)\|_2 &= \min_{p\in\mathcal{P}_m} \|X(f(J) - p(J))X^{-1}\|_2 \\ &\le \kappa_2(X) \min_{p\in\mathcal{P}_m} \max_{1\le i\le k} \|f(J_{\ell_i}(\lambda_i)) - p(J_{\ell_i}(\lambda_i))\|_2.\end{aligned}$$

In particular, for a diagonalizable matrix A (where $k = n$),

$$\min_{p\in\mathcal{P}_m} \|f(A) - p(A)\|_2 \le \kappa_2(X) \min_{p\in\mathcal{P}_m} \max_{1\le i\le n} |f(\lambda_i) - p(\lambda_i)|. \tag{10.9}$$

If A is diagonalizable by a well conditioned transformation, i.e., small $\kappa(X)$, then the bound (10.9) indicates that the *matrix* approximation problem is closely related to a *scalar* approximation problem on the (discrete) set $\{\lambda_1, \ldots, \lambda_n\}$ of the eigenvalues of A.
For a normal matrix A we can choose X with $\kappa_2(X) = 1$, and (10.9) is in fact an equality. In the special case $f(z) = z^{m+1}$, the unique solution p_* of the (matrix) approximation problem satisfies

$$\max_{1\le i\le m} |\lambda_i^{m+1} - p_*(\lambda_i)| = \min_{p\in\mathcal{P}_m} \max_{1\le i\le n} |\lambda_i^{m+1} - p(\lambda_i)|.$$

Thus, the polynomial $z^{m+1} - p_*(z)$ is the $(m+1)st$ *Chebyshev polynomial of the set* $\{\lambda_1, \ldots, \lambda_n\}$, i.e., the monic polynomial of degree $m+1$ that has minimal maximum norm on this set.

When $\kappa_2(X)$ is very large, however, the bound (10.9) may be useless in practice. Moreover, since the set of the eigenvalues is discrete, the value of the approximation problem on this set may be hard to compute or even estimate.

In order to simplify the problem or to get a smaller constant (or both) one can replace the discrete set of the eigenvalues by a larger (nondiscrete) set $\Omega \supset \{\lambda_1, \ldots, \lambda_n\}$. We have seen an example in the analysis of CG and the Lanczos algorithm, where the set of the eigenvalues was replaced by an inclusion interval $\Omega = [\lambda_{\min}(A), \lambda_{\max}(A)]$. Suitably scaled and shifted Chebyshev polynomials then give a computable estimate; see Theorems 4.13 and 8.11.

Bibliography

[1] W. E. Arnoldi, *The principle of minimized iteration in the solution of the matrix eigenvalue problem*, Quart. Appl. Math., 9 (1951), pp. 17–29.

[2] A. Barrlund, *Perturbation bounds for the LDL^H and LU decompositions*, BIT, 31 (1991), pp. 358–363.

[3] F. L. Bauer and C. T. Fike, *Norms and exclusion theorems*, Numer. Math., 2 (1960), pp. 137–141.

[4] C. A. Beattie, M. Embree, and D. C. Sorensen, *Convergence of polynomial restart Krylov methods for eigenvalue computations*, SIAM Rev., 47 (2005), pp. 492–515 (electronic).

[5] B. Beckermann and A. Townsend, *Bounds on the singular values of matrices with displacement structure*, SIAM Rev., 61 (2019), pp. 319–344. Revised reprint of "On the singular values of matrices with displacement structure" [MR3717820].

[6] C. M. Bishop, *Pattern Recognition and Machine Learning*, Information Science and Statistics, Springer-Verlag, 2006.

[7] E. K. Blum, *Numerical analysis and computation theory and practice*, Addison-Wesley Publishing Co., Reading, Mass.-London-Don Mills, Ont., 1972. Addison-Wesley Series in Mathematics.

[8] C. Eckart and G. Young, *A principal axis transformation for non-hermitian matrices*, Bull. Amer. Math. Soc., 45 (1939), pp. 118–121.

[9] L. Elsner, *An optimal bound for the spectral variation of two matrices*, Linear Algebra Appl., 71 (1985), pp. 77–80.

[10] J. G. F. Francis, *The QR transformation: a unitary analogue to the LR transformation. I*, Comput. J., 4 (1961/62), pp. 265–271.

[11] G. H. Golub, M. W. Mahoney, P. Drineas, and L.-H. Lim, *Bridging the gap between numerical linear algebra, theoretical computer science, and applications*, SIAM News, 39 (2006).

[12] A. Greenbaum, *Comparison of splittings used with the conjugate gradient algorithm*, Numer. Math., 33 (1979), pp. 181–193.

[13] A. Greenbaum and L. Gurvits, *Max-min properties of matrix factor norms*, SIAM J. Sci. Comput., 15 (1994), pp. 348–358.

[14] A. Greenbaum and L. N. Trefethen, *GMRES/CR and Arnoldi/Lanczos as matrix approximation problems*, SIAM J. Sci. Comput., 15 (1994), pp. 359–368. Iterative methods in numerical linear algebra (Copper Mountain Resort, CO, 1992).

[15] M. R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Research Nat. Bur. Standards, 49 (1952), pp. 409–436.

[16] N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second ed., 2002.

[17] R. A. Horn and C. R. Johnson, *Hadamard and conventional submultiplicativity for unitarily invariant norms on matrices*, Linear and Multilinear Algebra, 20 (1987), pp. 91–106.

[18] ——, *Matrix analysis*, Cambridge University Press, Cambridge, 1990. Corrected reprint of the 1985 original.

[19] A. S. Householder, *Some numerical methods for solving systems of linear equations*, Amer. Math. Monthly, 57 (1950), p. 453.

[20] A. S. Householder, *A class of methods for inverting matrices*, J. Soc. Ind. Appl. Math., 6 (1958), pp. 189–195.

[21] W. Joubert, *A robust GMRES-based adaptive polynomial preconditioning algorithm for nonsymmetric linear systems*, SIAM J. Sci. Comput., 15 (1994), pp. 427–439.

[22] W. Kahan, *Gauss–Seidel methods of solving large systems of linear equations*, PhD thesis, University of Toronto, 1958.

[23] J. Liesen and Z. Strakoš, *On optimal short recurrences for generating orthogonal Krylov subspace bases*, SIAM Rev., 50 (2008), pp. 485–503.

[24] J. Liesen and Z. Strakoš, *Krylov subspace methods. Principles and analysis*, Numerical Mathematics and Scientific Computation, Oxford University Press, Oxford, 2013.

[25] J. Liesen and P. Tichý, *On best approximations of polynomials in matrices in the matrix 2-norm*, SIAM J. Matrix Anal. Appl., 31 (2009), pp. 853–863.

[26] ——, *The field of values bound on ideal GMRES*, arXiv:1211.5969, (2012).

[27] G. Meurant, J. Papež, and P. Tichý, *Accurate error estimation in CG*, Numer. Algorithms, 88 (2021), pp. 1337–1359.

[28] J. M. Ortega, *On Sturm sequences for tridiagonal matrices*, J. Assoc. Comput. Mach., 7 (1960), pp. 260–263.

[29] A. M. Ostrowski, *On the linear iteration procedures for symmetric matrices*, Rend. Mat. e Appl., 14 (1954), pp. 140–163.

[30] C. C. Paige and M. A. Saunders, *Solution of sparse indefinite systems of linear equations*, SIAM J. Numer. Anal., 12 (1975), pp. 617–629.

[31] B. N. Parlett, *The symmetric eigenvalue problem*, vol. 20 of Classics in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1998. Corrected reprint of the 1980 original.

[32] J. L. Rigal and J. Gaches, *On the compatibility of a given solution with the data of a linear system*, Journal of the ACM (JACM), 14 (1967), pp. 543–548.

[33] Y. Saad and M. H. Schultz, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 856–869.

[34] H. Shapiro, *A survey of canonical forms and invariants for unitary similarity*, Linear Algebra Appl., 147 (1991), pp. 101–167.

[35] G. W. Stewart, *An inverse perturbation theorem for the linear least squares problem*, SIGNUM Newsletter, 10 (1975), pp. 39–40.

[36] ——, *Research, development, and LINPACK*, in Mathematical software, III (Proc. Sympos., Math. Res. Center, Univ. Wisconsin, Madison, Wis., 1977), Publ. Math. Res. Center, No. 39, Academic Press, New York, 1977, pp. 1–14.

[37] ——, *Review of Matrix Computations by Gene H. Golub and Charles F. Van Loan*, Linear Algebra Appl., 95 (1987), pp. 211–215.

[38] ——, *On the early history of the singular value decomposition*, SIAM Rev., 35 (1993), pp. 551–566.

[39] ——, *The decompositional approach to matrix computation*, Computing in Science & Engineering, 2 (2000), pp. 50–59.

[40] G. W. Stewart and J. G. Sun, *Matrix perturbation theory*, Computer Science and Scientific Computing, Academic Press, Inc., Boston, MA, 1990.

[41] X. Sun, N. Wang, C.-Y. Chen, J. Ni, A. Agrawal, X. Cui, S. Venkataramani, K. El Maghraoui, V. V. Srinivasan, and K. Gopalakrishnan, *Ultra-low precision 4-bit training of deep neural networks*, in Advances in Neural Information Processing Systems, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, eds., vol. 33, Curran Associates, Inc., 2020, pp. 1796–1807.

[42] L. N. Trefethen and D. Bau, III, *Numerical Linear Algebra*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997.

[43] N. Trefethen, *Notes of a Numerical Analyst*, London Mathematical Society Newsletter, 497 (2021), p. 36.

[44] M. Udell and A. Townsend, *Why are big data matrices approximately low rank?*, SIAM J. Math. Data Sci., 1 (2019), pp. 144–160.

[45] B. Waldén, R. Karlson, and J. G. Sun, *Optimal backward perturbation bounds for the linear least squares problem*, Numer. Linear Algebra Appl., 2 (1995), pp. 271–286.

[46] J. H. Wilkinson, *The algebraic eigenvalue problem*, Clarendon Press, Oxford, 1965.

Index