ON ITERATIVE REFINEMENT/IMPROVEMENT OF THE SOLUTION TO AN ILL CONDITIONED LINEAR SYSTEM

By

Abdramane Sermé

A dissertation submitted to the Graduate Faculty in Mathematics in partial fulfillment of the requirements for the degree of Doctor of Philosophy

The City University of New York

2008

This manuscript has been read and accepted for the
Graduate Faculty in Mathematics in satisfaction of the
dissertation requirement for the degree of Doctor of Philosophy.

Professor Victor Y. Pan

Date

Chair of Examining Committee

Professor Józef Dodziuk

Date

Executive Officer

Professor Alvany Rocha

Professor Attila Maté

Supervisory Committee

THE CITY UNIVERSITY OF NEW YORK

Abstract

ON ITERATIVE REFINEMENT/IMPROVEMENT OF

THE SOLUTION TO AN ILL CONDITIONED LINEAR

SYSTEM

By

Abdramane Sermé

Adviser: Professor Victor Y. Pan

We study additive preconditioning $A \rightarrow C = A + UV^H$ for preconditioner UV^H of a smaller rank r. The SMW formula $A^{-1} = (C - UV^H)^{-1} = C^{-1} + C^{-1}U(I_r - V^HC^{-1}U)^{-1}V^HC^{-1}$ reduces the solution of a linear system $Ax = b$ to better conditioned linear systems with the matrices $S = I_r - V^HC^{-1}U$ and C. The computations preserve the structure and the sparseness of the input matrix A. We compute the matrix $W = C^{-1}U$ with a higher precision by applying iterative refinement/improvement

to approximate the matrix W closely as a sum $W_0+W_1+...+W_k$, where the matrices W_i are filled with low precision values.

We prove that if $\frac{\|C^{-1}F_k\|}{1-\|C^{-1}F_k\|} < 1$, where $F_k = C_k - C$, $X_k = W_0 + ... + W_k$ and $X = W$, then $\|X_k - X\| \leq \mathcal{O}(\bar{u})$. By applying forward error analysis, we prove that $\frac{\|X_k-X\|}{\|X\|} \leq \mathcal{O}(u)$, and by applying backward error analysis that $\lim_{k\longrightarrow\infty} \frac{\|U_k-CW_k\|}{\|C\|\|W_k\|} = \frac{4c_1(k)}{1-c_1'(k)cond_2Cu}\bar{u}$, where $c_1(k)$ and $c_1'(k)$ are linear functions in k.

To my wife Lisa,

my son Yusuf,

and my parents.

Acknowledgements

I would like to thank my advisor Professor Victor Y. Pan without whom this work would have been impossible. It was an honor and a privilege to be his student and benefit from his generosity, warm personality and invaluable support.

I would like also to thank the other members of my defense committee; Professor Alvany Rocha and Professor Attila Maté.

I am grateful to all professors and personnel of the department of mathematics of the City University of New York Graduate Center. In particular, many thanks to the office assistant Robert Landsman.

I am deeply indebted to my wife Lisa and my parents, in particular my elderly brother Youssoufou Sermé and my cousin Losseni Zemé.

Table of Contents

Introduction

0.1 Multiplicative and additive preconditioners

Solving linear system $Ax = b$ is the central problem of matrix computations. Its accurate solution $x = A^{-1}b$ can be computed readily if the system is well conditioned but this requires special care if the system is ill conditioned. We study preconditioning, a popular tool that improves conditioning of linear systems.

All mathematical concepts used in this section are formally defined in Chapter 1.

0.1.1 Multiplicative preconditioners

Definition 0.1.1. (Multiplicative Preprocessors (MPPs) and Multiplicative Preconditioners (MPCs))

$Cond_2 A = \|A\|_2 \|A^{-1}\|_2$ is the condition number of the matrix A in the 2-norm (cf. Definition 1.1.12, page 15).

Any pair of nonsingular matrices M of size $m \times m$ and C of size $n \times n$ is an MPP for a matrix A of size $m \times n$, whereas the transition $A \leftarrow MAC$ is its M-preprocessing.

Such an MPP is a Multiplicative Preconditioner (MPC) and the M-preprocessing is an M-preconditioning if $cond_2 A$ is large, whereas $cond_2 (MAC)$ is not (see page 15 for the definition of $cond_2 A$). Such an MPP is a multiplicative compressor if the matrix A is rank deficient, whereas the matrix MAC turns into a full rank matrix after the deletion of its zero rows and columns.

The multiplicative preconditioning is the transformation of an ill conditioned linear system into a better conditioned one by pre- and/or post-multiplying the coefficient matrix by some fixed matrices. In other words, we replace the linear system $Ax = b$ with an equivalent one $BACy = Bb$, where BAC is a better conditioned matrix and $x = Cy$. The matrices B and C are multiplicative preconditioners.

Remark 0.1.1. The problem with the use of multiplicative preconditioners for an ill conditioned input matrix A is that they are mostly defined by the Singular Value Decomposition (SVD) of A (cf. Definition 1.1.7, page 13), whose computation is an expensive task. Furthermore, the help of random multiplicative preconditioning against ill conditioning is limited because $cond_2 A \leq \Pi_i cond_2 F_i$ if $A = \Pi_i F_i$. Lastly, the multiplicative preconditioning may destroy the structure of the original input matrix.

To get around these problems, we use additive preconditioning to transform the original ill conditioned linear system into a better conditioned one. For the transition to this better conditioned linear system, some high precision matrix computations are

required. We perform them by using iterative refinement/improvement.

0.1.2 Additive preconditioners

Definition 0.1.2. (Additive Preprocessors (APPs))

For a pair of matrices U of size $m \times r$ and V of size $n \times r$, both having full rank $r > 0$, the matrix UV^H of rank r is an APP (of rank r) for any $m \times n$ matrix A, the matrix $C = A + UV^H$ is its A-modification. The matrices U and V are the generators of the APP, and the transition $A \leftarrow C$ is an A-preprocessing of rank r for the matrix A. An APP UV^H for a matrix A is an additive preconditioning (APC) and an A-preprocessing is an A-preconditioning if $cond_2 A \gg cond_2 C$ (cf. Definition 1.1.10, page 14). The condition number associated with the 2-norm, $cond_2$, is defined in Definition 1.1.12. An APP is an additive compressor (AC) and an A-preprocessing is an A-complementation if the matrix A is rank deficient, whereas the A-modification C has full rank. An APP UV^H is unitary if the matrices U and V are unitary.

Additive preconditioning consists in adding a matrix UV^H, usually of a smaller rank, to the input matrix A, to decrease its condition number. The A-modification is supposed to generate a well conditioned matrix C. To compute the A-modification $C = A + UV^H$ error-free, we fill the generators U and V with short binary numbers.

Remark 0.1.2. Suppose UV^H has rank r. Then, we expect that $cond_2 C = \sigma_1(C)/\sigma_n(C)$

has the order $\sigma_1(A)/\sigma_{n-r}(A)$, where A is an $n \times n$ matrix if the additive preconditioner UV^H is

i) random

ii) well conditioned, and

iii) properly scaled, that is $\|A\| \,/\, \left\|UV^H\right\|$ is not large and not small.

$\sigma_j(A)$ (or simply σ_j) for $j = 1, \ldots, n$ is the j^{th} largest singular value of the matrix A (for the definition of the singular values of a matrix see Section 1.1.6 on page 13).

Let $C = A + UV^H$. Then, we apply the Sherman-Morrison-Woodbury (SMW) formula to the original linear system $Ax = b$ and transform it into better conditioned linear systems of small sizes, with well conditioned matrices $V^H C^{-1}$, $C^{-1}U$, and $S = I - V^H C^{-1} U$ called aggregates. To compute the Schur aggregate $S = I - V^H C^{-1} U$ with high precision, we compute $W = C^{-1}U$ using the iterative refinement/improvement algorithm. We prove that we can get very close to the solution W of the linear system $CW = U$. We closely approximate it by working with numbers rounded to the IEEE (Institute of Electrical and Electronics Engineers) standard double precision (cf. Table 2.2, page 33). The proof of that theorem is provided in Chapter 3. We apply additive preconditioning and the SMW formula to prove the convergence of the iterative refinement/improvement algorithm. We also use error-free summation algorithm to derive a more accurate solution of the original linear system $Ax = b$.

Our (forward and backward) error analysis in Chapter 3 proves some reasonable error bounds for the solution.

Chapter 1

Additive preconditioning and the Schur aggregation

In this chapter we recall some basic definitions of matrix computation and perturbation theory.

1.1 Matrix computations: Preliminaries

Most of the definitions and the theorems reproduce or slightly modify the customary definitions in [1], [2], [4], [6]. $\mathcal{C}$ is the field of complex numbers. Let Δ be a set in $\mathcal{C}$ and let m and n be a pair of positive integers.

1.1.1 Vectors and matrices

$A = (a_{i,j})_{i=1,j=1}^{m,n} \in \Delta^{m\times n}$ is a $m \times n$ matrix with the entries in the set Δ. $\mathbf{v} = (v_i)_{i=1}^{n}$ $\in \Delta^{n\times 1}$ is a column vector of dimension n with the coordinates in Δ. A is called a square matrix if $m = n$.

A^T and $\mathbf{v}^T$ are the transposes of the matrix A and the vector $\mathbf{v}$, and A^H, $\mathbf{v}^H$ are their Hermitian or complex conjugate transposes. For a real matrix A and a vector $\mathbf{v}$, we have $A^H = A^T$ and $\mathbf{v}^H = \mathbf{v}^T$.

I_k or simply I is the $k \times k$ identity matrix for $k \in \mathcal{Z}^+$. e_i is the i^{th} column vectors. The zero matrix O is the matrix whose elements are zero, $O_{k,k}$ is the $k\times k$ zero matrix.

The $m \times n$ system of linear equations

$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1$

$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2$

$\cdots$

$a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n = b_m$

can be written in a compact form as $\sum_{j=1}^{n} a_{ij}x_j = b_i, i = 1, 2, ..., m$. This last form is equivalent to $Ax = b$, if we define A, x and b by

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots a_{1n} \\ a_{21} & a_{22} & \dots a_{2n} \\ a_{m1} & a_{m2} & \dots a_{mn} \end{bmatrix}, x = \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{bmatrix} \text{ and } b = \begin{bmatrix} b_1 \\ b_2 \\ \dots \\ b_m \end{bmatrix}.$$

The matrix A is a *unitary matrix* if $A^H A = I$ and $AA^H = I$. An $m \times n$ matrix A has full rank if its rank equals $min\{m, n\}$. Otherwise the matrix is rank deficient. *Rank A* is the rank of the matrix A. *Det A* is the determinant of a square matrix A. A matrix is singular if and only if its determinant is zero.

1.1.2 Random matrices

Random sampling of elements from a finite set Δ is their selection from this set at random, independently of each other, and under the uniform probability distribution on Δ. A matrix is random if its entries are randomly sampled from a fixed finite set Δ.

Lemma 1.1.1. [16].

For a finite set Δ of cardinality $|\Delta|$, and a polynomial p in m variables of total degree d, suppose p does not vanish identically on the set Δ^m, and let the values of its variables be randomly sampled from the set Δ. Then the polynomial vanishes with a probability of at most $d/|\Delta|$.

1.1.3 Rank and nullity

The linear space generated by a set of vectors is their span. $Range(A)$ is the range of a matrix A, that is the span of its column vectors.

Definition 1.1.1. Let $A \in \Delta^{m\times n}$. The column span of A is the subspace $\mathcal{R}(A) = \{Ax : x \in \Delta^n\}$. The row space of A is the space $\mathfrak{R}(A^T)$. $Rank(A)$, the rank of A is the integer $\rho = rank(A) = dim[\mathfrak{R}(A)] = dim[\mathfrak{R}(A^T)]$.

Definition 1.1.2. Let $A \in \Delta^{m\times n}$. The space $\mathfrak{N}(A) = \mathfrak{RN}(A) = \{x\colon Ax = 0\}$ is the right null space of the matrix A.

The space $\mathfrak{LN}(A) = \{x\colon x^T A = 0^T\} = \mathfrak{N}(A^H)$ is its left null space. If $\rho = rank(A)$, its left nullity $lnul\, A = n - \rho$ and its right nullity $rnul\, A = m - \rho$ are the dimensions of its left and right null spaces $\mathfrak{N}(A)$ and $\mathfrak{L}(A)$, respectively.

Definition 1.1.3. Let $A \in \Delta^{m\times n}$, the nullity of A is denoted

$nul\, A = min\,\{m, n\} - \rho = min\,\{lnul\, A, rnul\, A\}$.

We have $nul\, A = lnul\, A = n - \rho$ for $n \leq m$ and $nul\, A = rnul\, A = m - \rho$ for $m \leq n$. The numerical nullity of the matrix A is the number of its small singular values (cf. Definition 1.1.7 on page 13). The numerical nullity of the matrix A is denoted $nnul\, A$.

Remark 1.1.1. If a linear system $Ax = b$ has a solution x_0, then any solution can be expressed as $x_0 + \mathfrak{N}(A) = \{x_0 + x\colon\ x \in \mathfrak{N}(A)\}$.

1.1.4 Vector and matrix norms

The definition and properties of a norm and consistent norm can be found in [4].

Theorem 1.1.2. *The following three functions on* $\mathcal{C}^n$ *with* $x = (x_1, x_2, \ldots, x_n)$ *are vector norms:*

1) $\|x\|_1 = \sum_i |x_i|$ *called the* 1*-norm.*

2) $\|x\|_2 = (\sum_i |x_i|^2)^{1/2} = (x^T x)^{1/2}$ *called the Euclidean norm or* 2*-norm.*

3) $\|x\|_\infty = max_i |x_i|$ *called the* ∞*-norm.*

They all verify the following inequalities,

$\|x + y\| \leq \|x\| + \|y\|$ *(called the triangular inequality),*

$\|x - y\| \geq \|x\| - \|y\|$, *and*

if $x \neq 0$, *then* $\left\| \frac{x}{\|x\|} \right\| = 1$, *for any pair of vectors* x *and* y.

Definition 1.1.4. A matrix $A \in \Delta^{m \times n}$ has

1) the Frobenius norm $\|A\|_F = (trace(A^H A))^{1/2} = (\sum_{i,j} |a_{ij}|^2)^{1/2}$,

2) the 1-norm $\|A\|_1 = max_{1 \leq j \leq n} (\sum_{i=1}^m |a_{ij}|)$,

3) the ∞-norm $\|A\|_\infty = max_{1 \leq i \leq n} (\sum_{j=1}^n |a_{ij}|)$,

4) the 2-norm $\|A\|_2 = sup_{\|x\|=1} \|Ax\| = \sigma_1(A) = \sigma_{max}(A)$, also called the spectral norm, where $\sigma_j(A)$ denotes the j^{th} largest singular value of the matrix A (see the definition of singular values in Section 1.1.6, page 13).

Remark 1.1.2. We have

1) $\|A\|_2 \leq \|A\|_F \leq \sqrt{n}\,\|A\|_2$ for $A \in \Delta^{m\times n}$, $m \geq n$,

2) $\|A\|_F^2 = \sum_j \sigma_j^2$,where σ_j for $j = 1, 2, \ldots$ is the j^{th} largest singular value of the matrix A,

3) $\|A\|_2 \leq \|A\|_1 \leq \sqrt{n}\,\|A\|_2$.

Theorem 1.1.3. *An $n \times n$ matrix A is nonsingular if and only if it satisfies one of the following conditions*

1) $rank(A) = n$,

2) $nul\, A = 0$,

3) for any vector b, the system $Ax = b$ has a solution,

4) if a solution of the system $Ax = b$ exists, it is unique,

5) for all x, $Ax = 0 \Rightarrow x = 0$,

6) the columns (rows) of A are linearly independent,

7) there is a matrix A^{-1} such that $A^{-1}A = AA^{-1} = I$,

8) $detA \neq 0$,

9) all singular values of A are nonzero (cf. Section 1.1.6, *page* 13*).*

Part 5 in the above theorem implies the following corollary.

Corollary 1.1.4. *The product of square matrices A and B is nonsingular if and only if A and B are nonsingular. In that case $(AB)^{-1} = B^{-1}A^{-1}$.*

Theorem 1.1.5. *The 2-norm has the following properties*

1) $\|A\|_2 = max_{\|x\|=\|y\|=1} \left|y^H A x\right| = \sigma_1(A)$.

2) $\|A\|_2^2 = max_{\|x\|=1} x^H (A^H A) x$.

3) $\|A\|_2 = \left\|A^H\right\|_2$.

4) $\|A\|_2 \leq \|A\|_F$, *with equality if and only if* $rank(A) = 1$.

5) $\|A\|_2{}^2 \leq \|A\|_1 \|A\|_\infty$.

1.1.5 Absolute norms

If the absolute values of the entries of a vector x are not greater than the absolute values of the entries of a vector y, that is, if $|x| \leq |y|$, then should we have $\|x\| \leq \|y\|$ for any norm $\|.\|$? Unfortunately, there are simple counterexamples to this appealing conjecture. For example, the function $\|x\| = x_1^2 - \frac{1}{2}x_1 x_2 + x_2^2$ (with $x \in \mathbb{R}^2$) is a norm but $\left\|\begin{bmatrix} 1 \\ 0 \end{bmatrix}\right\| = 1$ and $\left\|\begin{bmatrix} 1 \\ 0.1 \end{bmatrix}\right\| = 0.96$.
The norms that are monotone in the elements are useful in componentwise error analysis, and so we recall the following definition.

Definition 1.1.5. A norm $\|.\|$ is an absolute norm if

$|x| \leq |y| \Rightarrow \|x\| \leq \|y\|$ or equivalently if $\|x\| \leq \| \, |x| \|$. Here $|x| = (|x_1|, |x_2|, \ldots, |x_n|)$ and $|x_i| \leq |y_i|$ for all $i = 1, \ldots, n$.

Definition 1.1.6. For any nonzero matrix $P \in \mathcal{C}^{n \times n}$, $|P| = (|p_{ij}|)_{i=1,j=1}^{n,n}$.

Remark 1.1.3. Among the most popular vector norms, the 1-, 2-, and ∞-norms are clearly absolute. So are the 1-, ∞-, and the Frobenius norms of matrices. Unfortunately, the matrix 2-norm is not absolute. However, it does satisfy the relation $|A| \leq B \Rightarrow \|A\|_2 \leq \||A|\|_2 \leq \|B\|_2$ where $|A| = (|a_{ij}|)_{i=1,j=1}^{m,n}$, $|a_{ij}| \leq b_{ij}$ for $i = 1, \ldots, m$ and $j = 1, \ldots, n$.

1.1.6 The Singular Value Decomposition (SVD)

Definition 1.1.7. [14]. The compact singular value decomposition or SVD of an $m \times n$ matrix A of a rank ρ is the decomposition:

$A = S^{(\rho)} \sum^{(\rho)} T^{(\rho)H} = \sum_{j=1}^{\rho} \sigma_j \mathbf{s}_j \mathbf{t}_j^H$ where $S^{(\rho)} = (\mathbf{s}_j)_{j=1}^{\rho}$ and $T^{(\rho)} = (\mathbf{t}_j)_{j=1}^{\rho}$ are unitary matrices, that is, $S^{(\rho)H} S^{(\rho)} = I_\rho$, $T^{(\rho)H} T^{(\rho)} = I_\rho$, $\sum^{(\rho)} = diag(\sigma_j)_{j=1}^{\rho}$ is a diagonal matrix, $\mathbf{s}_j$ and $\mathbf{t}_j$ are m- and n-dimensional vectors, respectively, and $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_\rho > 0$. σ_j or $\sigma_j(A)$ for $j = 1, \ldots, \rho$ is the j^{th} largest singular value of the matrix A.

Definition 1.1.8. [14]. Singular Value Decomposition (SVD)

Let us write $l = lnul\, A = m - \rho$, $r = rnul\, A = n - \rho$. The pair $S^{(nul)} = (\mathbf{s}_j)_{j=\rho+1}^{m}$ and $T^{(nul)} = (\mathbf{t}_j)_{j=\rho+1}^{n}$ of left and right unitary null matrix bases for a matrix A define the square unitary matrices $S = (S^{(\rho)}, S^{(nul)}) = (\mathbf{s}_j)_{j=1}^{m}$ and $T = (T^{(\rho)}, T^{(nul)}) = (\mathbf{t}_j)_{j=1}^{n}$ and the $m \times n$ matrix $\sum = diag(\sum^{(\rho)}, O_{l,r})$. The equation $A = S \sum T^H$ is the Singular Value Decomposition of the matrix A, also called its SVD and full SVD.

The scalars σ_j for $j \geq 1$ are the singular values of the matrix A.

Hereafter, we write $\sigma_j = 0$ for $j > \rho$ and $\sigma_j = +\infty$ for $j < 1$.

Definition 1.1.9. The vectors $\mathbf{s}_j$ for $j = 1, \ldots, m$ and $\mathbf{t}_j$ for $j = 1, \ldots, n$ are the left and right singular vectors, associated with a singular value σ_j, respectively.

The associated right and left singular spaces are the vectors spaces of all the right and left singular vectors of that singular value, respectively.

The null vectors, whose entries are all zero, are the singular vectors associated with the singular value zero.

Theorem 1.1.6. *We have* $A\mathbf{t}_j = \sigma_j \mathbf{s}_j$ *and* $\mathbf{s}_j^H A = \sigma_j \mathbf{t}_j^H$ *for all* j.

$A^H = T \sum^T S^H, A^H A = T \sum^T \sum T^H$, *and* $AA^H = S \sum \sum^T S^H$.

Definition 1.1.10. We write $n \gg d$ when the ratio n/d is large.

Definition 1.1.11. The Moore-Penrose generalized inverse (also called the pseudoinverse) of an $m \times n$ matrix A of a rank ρ is the matrix A^- such that

$A^- = (A^H A)^{-1} A^H$ if $m \geq n = \rho$

$A^- = A^H (AA^H)^{-1}$ if $m = \rho \leq n$

$A^- = A^{-1}$ if $m = n = \rho$.

We write A^{-H} for $(A^H)^- = (A^-)^H$

Definition 1.1.12. The condition number, $cond_2 A$ of a matrix A of a rank ρ is $cond_2 A = \sigma_1(A)/\sigma_\rho(A) = \|A\|_2 \|A^-\|_2$. A matrix is said to be ill conditioned if its condition number is large, that is if $\sigma_1(A) \gg \sigma_\rho(A)$, and is called well conditioned otherwise.

Remark 1.1.4. A matrix A of a rank $\rho > 1$ can be ill conditioned where $\sigma_1(A) \approx \sigma_j(A) \gg \sigma_{j+1}(A) \approx \sigma_\rho(A)$ for some j, $1 \leq j < \rho$. If ρ is large, the matrix A can be ill conditioned even if $\sigma_j(A)/\sigma_{j+1}(A) \leq c$ for all j and smaller bound $c > 1$. For example $cond_2(A) = 2^{100}$ with $c = 2$ and $\rho = 101$ if $\sigma_j(A)/\sigma_{j+1}(A) = 2$ for all j.

We recall the following simple theorem [4, 14].

Theorem 1.1.7. *The matrices A^H and A^- have the same right and left singular spaces. Furthermore $\sigma_j(A^H) = \sigma_j(A) = 1/\sigma_{\rho+1-j}(A^-)$ for $j = 1, \ldots, \rho$ with $\rho = rank(A)$, whereas $\sigma_j(A^H) = \sigma_j(A) = \sigma_j(A^-) = 0$ for $j > rank(A)$.*

1.2 Convergence and perturbation theory

1.2.1 Convergence

There is a natural way to extend the notion of limit from $\mathcal{C}$ to $\mathcal{C}^n$.

Definition 1.2.1. Let $\left\{x_k = (x_1^{(k)}, x_2^{(k)}, \ldots, x_n^{(k)})^T\right\}_1^\infty$ be a sequence of vectors in $\mathcal{C}^n$ and let $x \in \mathcal{C}^n$. The sequence $\{x_k\}_1^\infty$ converges componentwise to x and we write

$\{x_k\}_1^\infty \longrightarrow x$ if

$$\lim_{k\longrightarrow\infty} x_i^{(k)} = x_i$$

for $i = 1, 2, \ldots, n$.

Here is another way to define convergence in $\mathcal{C}^n$.

Definition 1.2.2. Let $\left\{x_k = (x_1^{(k)}, x_2^{(k)}, \ldots, x_n^{(k)})^T\right\}_1^\infty$ be a sequence of vectors in $\mathcal{C}^n$ and let $x \in \mathcal{C}^n$. The sequence $\{x_k\}_1^\infty$ converges normwise to x, that is

$$\lim_{k\longrightarrow\infty} x_k = x,$$

if and only if

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

Remark 1.2.1. There is no compelling reason to expect the two notions of convergence to be equivalent. In fact for infinite-dimensional vector space, they are not, as the following example shows.

Example: Let l_∞ be the set of all infinite row vectors X satisfying $\|X\|_\infty = Sup_i |x_i| < \infty$, where $\|.\|_\infty$ is the norm on l_∞. Now consider the following infinite sequence:

$x_1 = (1, 0, 0, 0, \ldots)$

$x_2 = (0, 1, 0, 0, \ldots)$

$x_3 = (0, 0, 1, 0, \ldots)$

$x_4 = (0, 0, 0, 1, \dots)$.

Clearly this sequence converges to the zero vector componentwise since each component converges to zero. However, $\|x_k - 0\|_\infty = 1$ for each k. Hence the sequence does not converge to zero in the ∞-norm.

Not only componentwise and normwise convergence sometimes disagree, but different norms can generate different notions of convergence. Fortunately, we only deal with finite dimensional spaces, in which all notions of convergence coincide. We recall here the equivalence of the 1- and 2-norms with the inequality $\|x\|_2 \leq \|x\|_1 \leq \sqrt{n}\,\|x\|_2$, where $x \in \mathcal{C}^n$.

1.2.2 Perturbation of the identity matrix

The basic matrix operations, that is, multiplication by a scalar, matrix addition, and matrix multiplication are continuous. The matrix inversion, however, is not continuous and needs further investigation.

Definition 1.2.3. Let $\|.\|_{ln}$, $\|.\|_{lm}$, and $\|.\|_{mn}$ be norms on $\mathcal{C}^{l\times n}$, $\mathcal{C}^{l\times m}$, $\mathcal{C}^{m\times n}$. Then these norms are consistent if $\|AB\|_{ln} \leq \|A\|_{lm}\,\|B\|_{mn}$ for all $A \in \mathcal{C}^{l\times m}$ and $B \in \mathcal{C}^{m\times n}$.

Since we identify $\mathcal{C}^{n\times 1}$ with $\mathcal{C}^n$, the above definition also defines consistency between matrix and vector norms.

Theorem 1.2.1. [4]. *Let $\|.\|$ be a matrix norm on $\mathcal{C}^{n\times n}$ consistent with a vector norm (also denoted $\|.\|$) and let a matrix $X \in \mathcal{C}^{n\times n}$. Let P be a square matrix such that*

$\|P\| < 1$. *Then*

(*i*) *the matrix* $I - P$ *is nonsingular,*

(*ii*) $\|(I-P)^{-1}X\| \leq \frac{\|X\|}{1-\|P\|}$, *and*

(*iii*) $\|(I-P)^{-1} - I\| \leq \frac{\|P\|}{1-\|P\|}$.

Proof. Define a vector norm $\|.\|$ consistent with the matrix norm $\|.\|$. Now, let $x \in \mathcal{C}^n$ be a nonzero vector. Then

$\|(I-P)x\| = \|x - Px\| \geq \|x\| - \|Px\| \geq \|x\| - \|P\|\,\|x\| = (1-\|P\|)\,\|x\| > 0$. Hence, in virtue of Theorem 1.1.3, the matrix $I - P$ is nonsingular. This proves Part (*i*). To establish Part (*ii*), write $G = (I-P)^{-1}X$. Then, $X = (I-P)G = G - PG$. Hence, $\|X\| \geq \|G\| - \|P\|\,\|G\|$, so $\|G\| \leq \frac{\|X\|}{1-\|P\|}$, and we deduce (*ii*). To establish Part (*iii*), set $H = (I-P)^{-1} - I$. Then, by multiplying by $I - P$, we find that $H - PH = P$. Hence, $\|P\| \geq \|H\| - \|P\|\,\|H\|$, and (*iii*) follows by solving for $\|H\|$. □

The following corollary extends Theorem 1.2.1.

Corollary 1.2.2. *Let* $A, E \in \mathcal{C}^{n\times n}$. *If* $\|A^{-1}E\| \leq 1$, *then* $\|(A+E)^{-1}\| \leq \frac{\|A^{-1}\|}{1-\|A^{-1}E\|}$.
Moreover, $(A+E)^{-1} - A^{-1} = [(I - A^{-1}E) - I]A^{-1}$,
so that $\|(A+E)^{-1} - A^{-1}\| \leq \frac{\|A^{-1}\|\|A^{-1}E\|}{1-\|A^{-1}E\|}$.

The corollary remains valid if all occurrences of $\|A^{-1}E\|$ are replaced by $\|EA^{-1}\|$. The corollary is closely related to the results on the perturbation of the solution of a linear system presented in the second chapter.

1.2.3 Neumann series

Theorem 1.2.3. [4]. *Let $P \in \mathcal{C}^{n\times n}$ and suppose that*

$$\lim_{k\longrightarrow\infty} P^k = 0.$$

Then $I - P$ is nonsingular and $(I-P)^{-1} = \sum_{k=0}^{\infty} P^k$ (this sum is called Neumann sum). A sufficient condition for $P^k \longrightarrow 0$ is that $\|P\| < 1$ in some consistent norm, in which case

$\left\|(I + P + P^2 + \cdots + P^k) - (I-P)^{-1}\right\| \leq \frac{\|P\|^{k+1}}{1-\|P\|}$.

Corollary 1.2.4. *If $P^k \longrightarrow 0$, then $(I - |P|)^{-1}$ is nonnegative and $|(I-P)^{-1}| \leq (I-|P|)^{-1}$.*

Proof. Since $\left|P^k\right| \leq |P|^k$, P^k approaches zero along with $|P|^k$. The nonnegativity of $(I-|P|)^{-1}$ and the inequality in the theorem now follow by taking the limits in $\left|I + P + P^2 + \cdots + P^k\right| \leq I + |P| + |P|^2 + \cdots + |P|^k$. □

Corollary 1.2.5. *Let $X, E \in \mathcal{C}^{n\times p}$. Then $|\sigma_i(X+E) - \sigma_i(X)| \leq \|E\|_2, i = 1, 2, \ldots, p$.*

Theorem 1.2.6. [4, 14]. *Let $\|.\|$ denote a matrix norm and a consistent vector norm, and $A, E \in \mathcal{C}^{n\times n}$. If the matrix A is nonsingular, $Ax = b$ and*

(i) $\widetilde{A}\widetilde{x} = b$, *where $\widetilde{x}$ is an approximate value of x, then*

(ii) $\frac{\|\widetilde{x}-x\|}{\|\widetilde{x}\|} \leq \|A^{-1}E\|$.

In addition if $\|A^{-1}E\| < 1$, then $\widetilde{A} = A + E$ is nonsingular and

(iii) $\frac{\|\widetilde{x}-x\|}{\|x\|} \leq \frac{\left\|A^{-1}E\right\|}{1-\|A^{-1}E\|}$.

Proof. From (i), it follows that $\widetilde{x} - x = -A^{-1}E\widetilde{x}$, and (ii) follows. Now, assume that $\|A^{-1}E\| < 1$. Then, the matrix $\widetilde{A}$ is nonsingular if and only if the matrix $A^{-1}\widetilde{A} = I + A^{-1}E$ is nonsingular. Hence, by Theorem 1.2.1 $\widetilde{A}$ is nonsingular, and (iii) follows immediately from (ii) and the following lemma. □

Lemma 1.2.7. *For a vector norm* $\|.\|$, *suppose that* $\widetilde{\rho} = \frac{\|\widetilde{x}-x\|}{\|\widetilde{x}\|} < 1$.

Then $\frac{\|\widetilde{x}-x\|}{\|x\|} \leq \frac{\widetilde{\rho}}{1-\widetilde{\rho}}$.

1.3 Additive preconditioners and the Schur aggregation

Suppose A is an ill conditioned nonsingular $n \times n$ input matrix, and $C = A + UV^H$ is its A-modification (cf. Definition 0.1.2, page 3) with a well conditioned additive preconditioner (APC) UV^H of a rank $r < n$. Consider a linear system $Ax = b$.

1.3.1 Sherman-Morrison-Woodbury (SMW) formula

Definition 1.3.1. [4, 14]. Consider a 2×2 block matrix $B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}$.

The matrices $S_{22} = B_{22} - B_{21}B_{11}^{-1}B_{12}$ and $S_{11} = B_{11} - B_{12}B_{22}^{-1}B_{21}$ are called the *Schur complement* of the northwestern block B_{11} and the *Schur complement* of the southeastern block B_{22} in the matrix B, respectively, provided $B_{11}^{-1}B_{11} = I$ and $B_{11}B_{11}^{-1} = I$ with its counterpart $B_{22}^{-1}B_{22} = I$ and $B_{22}B_{22}^{-1} = I$ [14, page 103].

Lemma 1.3.1. *Let the above block matrix B be nonsingular and let* $B^{-1} = \begin{bmatrix} F & X \\ Y & Z \end{bmatrix}$, *for some square matrices F, X, Y and Z of the same sizes as B_{ij}, respectively. Then $F = S_{11}^{-1}$ and $Z = S_{22}^{-1}$ if the blocks B_{11} and B_{22} are nonsingular respectively.*

Theorem 1.3.2. [4, 14]. *For $n \times r$ matrices U and V and $n \times n$ matrix A, let the matrix $C = A + UV^H$ be nonsingular. Then, the matrices $A = C - UV^H$ and $S = I_r - V^H C^{-1} U$ are the respective Schur complements of the block I_r and C in the matrix* $W = \begin{bmatrix} C & U \\ V^H & I_r \end{bmatrix}$ *such that (i) $\det W = \det A = (\det C)(\det S)$.*
Furthermore if the matrix A is nonsingular, then so is the matrix S, and we have the Sherman-Morrison-Woodbury (SMW) formula

(ii) $(C - UV^H)^{-1} = C^{-1} + C^{-1} U S^{-1} V^H C^{-1}$.

Proof. [1]. Consider the factorization $\begin{bmatrix} C & U \\ V^H & I_r \end{bmatrix} =$

$$\begin{bmatrix} I_n & U \\ 0 & I_r \end{bmatrix} \begin{bmatrix} A & 0 \\ 0 & I_r \end{bmatrix} \begin{bmatrix} I_n & 0 \\ V^H & I_r \end{bmatrix} = \begin{bmatrix} I_n & 0 \\ V^H C^{-1} & I_r \end{bmatrix} \begin{bmatrix} C & 0 \\ 0 & S \end{bmatrix} \begin{bmatrix} I_n & C^{-1}U \\ 0 & I_r \end{bmatrix},$$

and we obtain (i). Inverting the above block factorization and using Lemma 1.3.1, we obtain that

$$\begin{bmatrix} A^{-1} & X \\ Y & Z \end{bmatrix} = \begin{bmatrix} I_n & 0 \\ -V^H & I_r \end{bmatrix} \begin{bmatrix} A^{-1} & 0 \\ 0 & I_r \end{bmatrix} \begin{bmatrix} I_n & -U \\ 0 & I_r \end{bmatrix}$$

$$= \begin{bmatrix} I_n & -C^{-1}U \\ 0 & I_r \end{bmatrix} \begin{bmatrix} C^{-1} & 0 \\ 0 & S^{-1} \end{bmatrix} \begin{bmatrix} I_n & 0 \\ -V^H C^{-1} & I_r \end{bmatrix}$$

$$= \begin{bmatrix} C^{-1} + C^{-1}US^{-1}V^H C^{-1} & -C^{-1}US^{-1} \\ -S^{-1}V^H C^{-1} & S^{-1} \end{bmatrix} \text{ for some matrices } X,\ Y, \text{ and } Z.$$

The equation (ii) follows from the matrix equality

$$\begin{bmatrix} A^{-1} & X \\ Y & Z \end{bmatrix} = \begin{bmatrix} C^{-1} + C^{-1}US^{-1}V^H C^{-1} & -C^{-1}US^{-1} \\ -S^{-1}V^H C^{-1} & S^{-1} \end{bmatrix}. \qquad \square$$

1.3.2 The Schur aggregation

The aggregation method consists of transforming an original linear system $Ax = b$ into linear systems of smaller sizes with well conditioned coefficients matrices $V^H C^{-1}$, $C^{-1}U$, and $S = I_r - V^H C^{-1}U$.

Remark 1.3.1. Aggregation is a well known technique (cf. e.g., [13]), but aggregation used here both decreases the size of the input matrix and improves its conditioning. One may remark that aggregation can be applied recursively until no ill conditioned matrix appears in the computation.

Remark 1.3.2. For an ill conditioned matrix A, the span of the rows of the matrix $V^H C^{-1}$ (respectively, columns of the matrix $C^{-1}U$) approximates the left (respectively, the right) singular space associate with the r smallest singular values of the matrix A. The larger the ratio $\frac{\sigma_{n-r}(A)}{\sigma_{n-r+1}(A)}$, the closer the approximation.

Definition 1.3.2. The r smallest singular values and the associate singular spaces

form the r-tail of the SVD of the matrix A. The computation of and with the tail aggregates $V^H C^{-1}$ and $C^{-1}U$ is called the tail aggregations.

Definition 1.3.3. (The Schur aggregation [1].)

The Schur aggregation is the process of reducing the linear system $Ax = b$ by using the SMW (Sherman-Morrison-Woodbury) formula

$$A^{-1} = (C - UV^H)^{-1} = C^{-1} + C^{-1}U(I_r - V^H C^{-1}U)^{-1}V^H C^{-1}.$$

The matrix $S = I_r - V^H C^{-1}U$, which is the Schur complement (Gauss transform) of the block C in the block matrix $\begin{bmatrix} C & U \\ V^H & I_r \end{bmatrix}$, is called the Schur aggregate.

Remark 1.3.3. If the A-modification $C = A + UV^H$ and the Schur aggregate S are well conditioned, then the numerical problems in the inversion of the matrix A are confined to the computation of the Schur aggregate S.

Definition 1.3.4. Suppose A denotes an $n \times n$ singular matrix of rank $n - r$, UV^H is its APC of a rank r, and the A-modification $C = A + UV^H$ is nonsingular. The null aggregate $V^H C^{-1}$ and $C^{-1}U$ are the left and right null matrix bases for the matrix A, that is, their columns span the left and right null spaces of the matrix A, respectively. The computation of the null aggregates $V^H C^{-1}$ and $C^{-1}U$ is the null aggregation.

Definition 1.3.5. The nullity of A, $nul\ A = n - rank\ A$, is the smallest integer r for which a rank r APC UV^H can define a nonsingular A-modification $C = A + UV^H$. The nullity of A, which is defined as the dimension of the null space can also be defined

as the large integer r for which we have $AC^{-1}U = 0$ or $V^HC^{-1}A = 0$, provided C is a nonsingular matrix. In this case, $C^{-1}U$ and V^HC^{-1} are the right and left null matrix bases for the matrix A.

Remark 1.3.4. [1]. If $rank(UV^H) = nul\, A$ where A is ill conditioned, then the A-modification $C = A + UV^H$ is ill conditioned too. In this case, if we compute the dimension in Definition 1.3.7 numerically with rounding to a finite precision, then we would output $nnul\, A$, which is the numerical nullity of the matrix A, that is the number of its small singular values. The null matrix bases $C^{-1}U$ and V^HC^{-1} would turn into two tail aggregates whose column spans would approximate the associated left and right singular spaces. These spaces form the tail of the SVD of the matrix A.

Chapter 2

Rounding errors and iterative refinement/improvement

2.0.3 Rounding errors

We recall some basics of perturbation and error analysis [4, 6, 14, 15].

Definition 2.0.6. Let $\hat{x}$ be an approximation of the scalar x. The absolute error in $\hat{x}$ approximating x is the number $\varepsilon = |\hat{x} - x|$.

Definition 2.0.7. Let $\hat{x}$ be an approximation of a scalar $x \neq 0$. The absolute and relative errors of this approximation are the numbers $|\hat{x} - x|$ and $\rho = \frac{|\hat{x}-x|}{|x|}$, respectively. If $\hat{x}$ is an approximation to x with relative error ρ, then there is a number $r = \frac{\hat{x}-x}{x}$ such that 1) $|r| = \rho$ and

2) $\hat{x} = x(1 + r)$.

Remark 2.0.5. The relative error ρ is independent of scaling, that is the scaling $x \longrightarrow \alpha x$ and $\hat{x} \longrightarrow \alpha\hat{x}$ leave ρ unchanged.

Theorem 2.0.3. [6]. *Assume that $\hat{x}$ approximates x with relative error $\rho < 1$. Then $\hat{x}$ is nonzero and $\rho = \frac{|x-\hat{x}|}{|\hat{x}|} \leq \frac{\rho}{1-\rho}$.*

Proof. From the definition of the relative error, we have $\rho|x| = |\hat{x} - x| \geq |x| - |\hat{x}|$. It follows that $|\hat{x}| \geq (1 - \rho)|x| > 0$. Hence, from these inequalities, it follows that $\frac{\rho}{1-\rho} = \frac{|\hat{x}-x|}{(1-\rho)|x|} \geq \frac{|x-\hat{x}|}{|\hat{x}|}$. □

Definition 2.0.8. The significant digits in a finite precision number are the first nonzero digit and all succeeding digits. Thus 1.7320 has five significant digits while 0.0491 has only three.

Definition 2.0.9. (The correct significant digits in $\hat{x}$ approximating x)
An approximation $\hat{x}$ to x is said to have p correct significant digits if $\hat{x}$ and x agree when rounded to p significant digits. Thus if $x = 0.9948$ and $\hat{x} = 0.9952$, then $\hat{x}$ does not have two correct significant digits ($x \longrightarrow 0.99, \hat{x} \longrightarrow 1$) but does have one and three correct significant digits.

Remark 2.0.6. If the relative error of x with respect to $\hat{x}$ is ρ, then x and $\hat{x}$ agree to roughly $-\log_2(\rho)$ correct significant digits. For binary system, if x and $\hat{x}$ have relative error of approximately 2^{-t-1}, then x and $\hat{x}$ agree to about t bits.

Definition 2.0.10. The componentwise relative error is defined as: $max_i \frac{|x_i - \hat{x}_i|}{|x_i|}$ for $x = (x_1, x_2, \ldots, x_i, \ldots)$.

Remark 2.0.7. In numerical computation, one has three main sources of errors.

1. Rounding errors, which are unavoidable consequences of working in finite precision arithmetic.

2. Uncertainty in the input data, which is always a possibility when we are solving practical problems.

3. Truncation errors, which are due to the omitted terms. One can avoid truncation errors by using Maple or Mathematica.

Definition 2.0.11. Precision is the number of digits in the representation of a real number. It defines the accuracy with which the computations and in particular the basic arithmetic operations $+, -, \cdot, /$ are performed. For floating point arithmetic, precision is measured by the unit roundoff or machine precision, for which we write u in single precision and $\bar{u}$ in double precision. The values of the unit roundoff are given in Table 2.1 in Section 2.1.

Remark 2.0.8. Accuracy refers to the absolute or relative error of an approximation.

Definition 2.0.12. Let $\hat{y}$ be an approximation of $y = f(y)$ computed with a precision u where f is a real function of a real scalar variable.

$\min\{|\Delta x| : \hat{y} = f(x + \Delta x)\}$ is called the (absolute) backward error, whereas the absolute or relative errors of $\hat{y}$ are called forward errors. The process of bounding the

backward error of a computed solution is called backward error analysis. Δx is the perturbation of x.

Remark 2.0.9. 1. Backward error analysis interprets rounding errors as perturbations of the data. Therefore, if the backward error is no larger than the uncertainties in the data, which is in $\mathcal{O}(u)$, then the computed solution should be the solution we are seeking. This is what we show in the third part of this thesis.

2. Backward error analysis reduces estimating errors to perturbation theory, which for many problems is well understood.

Definition 2.0.13. (Backward stability)

A method of computing $y = f(x)$ is called backward stable if for any x, it computes $\hat{y}$ with a small backward error, that is if $\hat{y} = f(x + \Delta x)$ for some small perturbation Δx.

Examples:

1. The operations $x \pm y$ are backward stable operations.

2. Most algorithms for computing the cosine function do not satisfy $\hat{y} = cos(x + \Delta x)$ with a relatively small perturbation Δx.

Definition 2.0.14. A mixed forward-backward error is defined by the equation $\hat{y} + \Delta\hat{y} = f(x + \Delta x)$ where $|\Delta\hat{y}| \leq \varepsilon |y|$, $|\Delta x| \leq \eta |x|$ with ε and η small constants.

Remark 2.0.10. This definition implies that the computed value $\hat{y}$ differs little from the value $\hat{y} + \Delta\hat{y}$ that would have been produced by an input $x + \Delta x$ little different

from the actual input x. Simpler, $\hat{y}$ is almost the right answer for almost the right data.

Definition 2.0.15. An algorithm is called numerically stable if it is stable in the mixed forward and backward error sense. Clearly, backward stable algorithm is numerically stable.

Remark 2.0.11. One may use the following rule of thumb:
Forward error $\lessapprox$ condition number $\times$ backward error, with approximate equality possible. Therefore the computed solution to an ill conditioned problem can have a large forward error even if the computed solution has a small backward error. The latter error can be amplified by the condition number in the transition to forward error. This is one of the motivation for decreasing the condition number of the matrix.

Definition 2.0.16. An algorithm is called forward stable if it produces answers with forward errors of similar magnitude to those produced by backward stable method.

Remark 2.0.12. A backward stability implies a forward stability but the converse is not true.

2.0.4 Impact on solving a linear system of equations

Definition 2.0.17. For an approximation $\hat{x}$ to a solution of a linear system $Ax = b$, the forward error is the ratio $\frac{\|x-\hat{x}\|}{\|x\|}$.

Definition 2.0.18. For a system of linear equations $Ax = b$, $\rho(x) = \frac{\|b-Ax\|}{\|A\|\|x\|}$ is called the relative residual. The relative residual gives us an indication on how closely Ax represents b and is scaling independent.

Lemma 2.0.4. [6]. $\rho(x) = min\left\{\frac{\|\Delta A\|_2}{\|A\|_2} : (A+\Delta A)X = b\right\}$ *where* ΔA *is the perturbation of the matrix* A.

Proof. If $(A+\Delta A)x = b$, then by substituting $r = b - Ax = \Delta A\, x$ we obtain that $\|r\|_2 \leq \|\Delta A\|_2 \|x\|_2$, and this implies the following bound (i) $\frac{\|\Delta A\|_2}{\|A\|_2} \geq \frac{\|r\|_2}{\|A\|_2\|x\|_2} = \rho(x)$. On the other hand, we have $(A+\Delta A)x = b$ for $\Delta A = rx^T/x^Tx$, and then $\|\Delta A\|_2 = \|r\|_2 / \|x\|_2$. So the bound (i) is attainable. □

Remark 2.0.13. Lemma 2.0.4 says that $\rho(x)$ measures how much the matrix A (not the vector b) must be perturbed in order for x to be the exact solution to the perturbed system, that is, $\rho(x)$ equals the normwise relative backward error. Bounding it is a challenge in our third chapter. If the data A and b are uncertain and $\rho(x)$ is no larger than this uncertainty, i.e., $\rho(x) = \mathcal{O}(\bar{u})$ (or $\rho(x) = \mathcal{O}(u)$), then the approximate solution $\hat{y}$ must be regarded as very satisfactory.

2.1 Floating-point number system

Definition 2.1.1. [6]. A floating-point number system F is a subset of the real numbers whose elements have the form $y = \pm m \times \beta^{e-t}$.

The system F is characterized by four integer parameters:

1) The base β (sometimes called the radix)

2) The precision t

3) The exponent range $e_{\min} \leq e \leq e_{\max}$

4) The mantissa m, that is an integer satisfying $0 \leq m \leq \beta^t - 1$.

To ensure a unique representation for each $y \in F$, it is assumed that $m \geq \beta^{t-1}$ if $y \neq 0$, so that the system is normalized.

The range of the nonzero floating-point numbers in F is given by

$$\beta^{e_{min}-1} \leq |y| \leq \beta^{e_{\max}}(1 - \beta^{-t}).$$

Table 2.1

Machine and arithmetic	β	t	$e_{\min}$	$e_{\max}$	unit roundoff u
IEEE Single	2	24	-125	128	$2^{-24} \approx 5.96 \times 10^{-8}$
IEEE Double	2	53	-1021	1024	$2^{-53} \approx 1.11 \times 10^{-16}$

Remark 2.1.1. Any floating-point number $y \in F$ can be written in the form

$$\begin{aligned} Y &= \left(\frac{d_1}{\beta} + \frac{d_2}{\beta^2} + \ldots + \frac{d_l}{\beta^t}\right) \times (\pm\beta^e) \\ &= .d_1 d_2 \ldots d_t \times (\pm\beta^e) \end{aligned}$$

where each digit d_i satisfies $0 \leq d_i \leq \beta - 1$ and $d_1 \neq 0$ for normalized numbers. d_1 is called the most significant digit and d_t the least significant digit.

Definition 2.1.2. The subnormal numbers (also called denormalized numbers) are numbers that can be written in the form

$$y = \pm m \times \beta^{e_{\min}-t}, \quad 0 < m < \beta^{t-1}$$

which have the minimum exponent and are not normalized.

The system F can be extended to include subnormal numbers.

Any subnormal number can be written in the form

$$y = \pm \beta^{e_{\min}} \times .d_1 d_2 \ldots d_t$$

with $d_1 = 0$.

Subnormal numbers have fewer digits of precision than the normalized numbers.

Definition 2.1.3. The mapping

$$\begin{aligned} \mathbf{R} &\rightarrow F \subset \mathbf{R} \\ x &\rightarrow fl(x) \end{aligned}$$

is called rounding. $fl(x)$ denotes an element of F nearest to x; it is the rounded value of the number x.

Remark 2.1.2. There are several ways to break ties when x is equidistant from two floating-point numbers e.g., taking $fl(x)$ to be the number of larger magnitude (round

away from zero) or the one with an even last digit d_t (called round to even).

Definition 2.1.4. The unit roundoff error u is the quantity $u = \frac{1}{2}\beta^{1-t}$.

Table 2.2

The two main floating-point formats.

Type	Size	Mantissa	Exponent	Unit Roundoff u	Range
IEEE Single	32 bits	23 + 1 bits	8 bits	$2^{-24} \approx 5.96 \times 10^{-8}$	$10^{\pm 38}$
IEEE Double	64 bits	52 + 1 bits	11 bits	$2^{-53} \approx 1.11 \times 10^{-16}$	$10^{\pm 308}$

The widely used IEEE standard binary arithmetic has $\beta = 2$. For the single binary precision $t = 24, e_{\min} = -125, e_{\max} = 128$, and $u = 2^{-24}$, and for the double binary precision $t = 53, e_{\min} = -1021, e_{\max} = 1024$, and $u = 2^{-53}$. The IEEE arithmetic rounds to even.

Remark 2.1.3. The error analysis consists in bounding the error of the computations (forward error or backward error), eventually in terms of u.

The following theorem shows that every real number x lying in F can be approximated by an element of F with a relative error no larger than u.

Theorem 2.1.1. [6, *page* 42]. *If* $x \in \mathbf{R}$ *lies in* F *then* $fl(x) = x(1+\delta)$ *with* $|\delta| < u$.

Proof. We may assume that $x > 0$. Clearly, the real number $x = \mu \times \beta^{e-t}$, $\beta^{t-1} \leq \mu \leq \beta^t - 1$ lies between the adjacent floating-point numbers $y_1 = \lfloor \mu \rfloor \beta^{e-t}$ and $y_2 = \lceil \mu \rceil \beta^{e-t}$.

Then $fl(x) = y_1$ or $fl(x) = y_2$, and we have

$$|fl(x) - x| \leq \frac{|y_2 - y_1|}{2} \leq \frac{\beta^{e-t}}{2}.$$

Therefore

$$\left|\frac{fl(x) - x}{x}\right| \leq \frac{\frac{1}{2}\beta^{e-t}}{\mu\beta^{e-t}} \leq \frac{1}{2}\beta^{1-t} = u.$$

The last inequality is strict unless $\mu = \beta^{t-1}$, in which case $x = fl(x)$. Hence, the inequality of the theorem is strict. □

Theorem 2.1.1 says that $fl(x)$ is equal to x multiplied by a factor very close to 1. The representation $1 + \delta$ for the factor is a standard choice, but it is not the only possibility. The following modified version of this theorem is also useful.

Theorem 2.1.2. [6]. *If $x \in \mathbf{R}$ lies in F, then*

$$fl(x) = \frac{x}{1+\delta} \text{ with } |\delta| \leq u.$$

Both theorems 2.1.1 and 2.1.2 give estimates for $fl(x)$. To carry out the rounding error analysis of an algorithm, we need to make some assumptions about the accuracy of the basic arithmetic operations.

Definition 2.1.5. (Standard Model) Let $x, y \in F$ and let op be one of $+, -, \cdot, /$. Then $fl(x \text{ op } y) = (x \text{ op } y)(1 + \delta)$ with $|\delta| \leq u$.

It is usually assumed that this property also holds for the square root operation. We are using this standard model and so, from now on $fl(.)$, for an argument that is an arithmetic expression, denotes the computed value of that expression. op represents floating-point operation in F.

Remark 2.1.4. Under the model above the computed value of x op y is "as good as" the rounded exact answer, in the sense that the relative error bound is the same in both cases. However, the model does not require that $\delta = 0$ when x op $y \in F$, a condition which obviously holds for the rounded exact answer. The model is valid for most computers, and, in particular, it holds for the IEEE standard arithmetic.

Definition 2.1.6. Hereafter, γ_n denotes the positive number $\frac{nu}{1-nu}$, where u is the unit roundoff and $n < \frac{1}{u}$.

Lemma 2.1.3. [6]. *If* $|\delta_i| \leq u$ *and* $\rho_i = \pm 1$ *for* $i = 1, \ldots, n$ *and* $nu < 1$, *then*

$$\prod_{i=1}^{n}(1+\delta_i)^{\rho_i} = 1 + \theta_n, \text{ where } |\theta_n| \leq \frac{nu}{1-nu} = \gamma_n.$$

In the following theorems, ΔA is the perturbation of the matrix A, which is perturbed into $\hat{A} = A + \Delta A$.

Theorem 2.1.4. [6]. *If G.E. (Gaussian Elimination) applied to* $A \in \mathbf{R}^{m\times n}$ $(m \geq n)$ *runs to completion, then the computed LU factors* $\hat{L} \in \mathbf{R}^{m\times n}$ *and* $\hat{U} \in \mathbf{R}^{m\times n}$ *satisfy* $\hat{L}\hat{U} = A + \Delta A$, $|\Delta A| \leq \gamma_n |\hat{L}||\hat{U}|$.

Theorem 2.1.5. [6, *page* 175]. *Let* $A \in \mathbf{R}^{n\times n}$ *and suppose the G.E. produces computed LU factors* $\hat{L}$ *and* $\hat{U}$*, and a computed solution* $\hat{x}$ *to* $Ax = b$.

Then $(A + \Delta A)\hat{x} = b,\ |\Delta A| \leq \gamma_{3n}|\hat{L}||\hat{U}|$, *where* $3nu < 1$ *and* $\gamma_{3n} > 0$.

Proof. We have $\hat{L}\hat{U} = A+\Delta A_1$, $|\Delta A_1| \leq \gamma_n|\hat{L}||\hat{U}|$ by Theorem 2.1.4. By substitution, and using Theorem 8.5 [6, *page* 154], we produce $\hat{y}$ and $\hat{x}$ satisfying

$$(\hat{L} + \Delta\hat{L})\hat{y} = b,\ |\Delta\hat{L}| \leq \gamma_n|\hat{L}|,$$

$$(\hat{U} + \Delta\hat{U})\hat{x} = \hat{y},\ |\Delta\hat{U}| \leq \gamma_n|\hat{U}|.$$

Thus

$$\begin{aligned} b &= (\hat{L} + \Delta\hat{L})(\hat{U} + \Delta\hat{U})\hat{x} = (A + \Delta A_1 + \hat{L}\Delta\hat{U} + \Delta\hat{L}\,\hat{U} + \Delta\hat{L}\Delta\hat{U})\hat{x} \\ &= (A + \Delta A)\hat{x}, \end{aligned}$$

where

$$\Delta A = \Delta A_1 + \hat{L}\Delta\hat{U} + \Delta\hat{L}\,\hat{U} + \Delta\hat{L}\Delta\hat{U}\ , and$$

$$|\Delta A| \leq (3\gamma_n + \gamma_n^2)|\hat{L}||\hat{U}| \leq (\gamma_n + \gamma_{2n})|\hat{L}||\hat{U}| \leq \gamma_{3n}|\hat{L}||\hat{U}|.$$

□

Remark 2.1.5. Ideally, we would like to have $|\Delta A| \leq u|A|$, which corresponds to the uncertainty introduced by rounding the elements of A, but each entry of A is involved into up to n arithmetic operations, and so we cannot expect any better bound than $|\Delta A| \leq c_n u|A|$, where c_n is a constant of order n. Such a bound holds if $\hat{L}$ and $\hat{U}$

satisfy $|\hat{L}||\hat{U}| = |\hat{L}\hat{U}|$. This certainly holds if $\hat{L}$ and $\hat{U}$ are nonnegative because then Theorem 2.1.4 gives us

$$|\hat{L}||\hat{U}| = |\hat{L}\hat{U}| = |A + \Delta A| \leq |A| + \gamma_n |\hat{L}||\hat{U}|.$$

Therefore,

$$|\hat{L}||\hat{U}| \leq \frac{1}{1-\gamma_n}|A|.$$

Substituting into the bound in Theorem 2.1.5, we obtain

$$(A + \Delta A)\hat{x} = b, \ |\Delta A| \leq \frac{\gamma_{3n}}{1-\gamma_n}|A|$$

where $\hat{L} \geq 0, \ \hat{U} \geq 0$.

This result says that $\hat{x}$ has a small componentwise relative backward error. In Chapter 3 we show that $\hat{x}$ has a small normwise relative backward error, which gives us a better understanding of the convergence $x \to \hat{x}$.

2.2 Error-free floating-point summation and multiplication

The MSAs, that is, the advanced algorithms for floating-point summation and multiplication, yield a high precision or even error-free output for a sequence of double precision additions, subtractions, and multiplications even where the leading significant bits of the output are canceled. In this section, we keep writing $fl(.)$ (with an

argument being an arithmetic expression) to denote floating-point numbers operations (cf. [1]).

2.2.1 Floating-point summation

Below we provide some summation algorithms with small error bound relative to the absolute value $|s_1 + s_2 + \ldots + s_h|$ of the output sum versus the error bounds relative to the generally larger sum $|s_1| + |s_2| + \ldots + |s_h|$ in other popular algorithms. This advantage is important where many most significant leading bits of the summands are lost in the sum, which is frequently the case in the Schur aggregation (Chapter 3).

We use the Matlab-like notation and assume the IEEE standard representation of floating-point numbers as $s = \sigma 2^e f$, where σ is equal to -1 or 1, e is an integer in a fixed range $[1 - r, r]$ for a fixed positive integer r, and f is either zero or a binary number in the range $[1, 2)$. f is represented with $p + 1$ bits, including the leftmost bit one, which is the leading and most significant bit.

In particular, $r = 127$ and $p = 23$ for single precision IEEE standard floating point numbers and $r = 1023$ and $p = 52$ for double precision IEEE standard floating point numbers.

Here is a summation algorithm due to D.E. Knuth [5].

Algorithm 2.2.1. *ERROR-FREE TRANSFORMATION OF THE SUM OF TWO*

FLOATING POINT NUMBERS

$$\begin{aligned} function[x, y] &= Twosum(a, b) \\ x &= fl(a + b) \\ z &= fl(x - a) \\ y &= fl\big((a - (x - z)) + (b - z)\big) \end{aligned}$$

The algorithm transforms two input-floating point numbers a and b into two output floating-point numbers x and y such that $a + b = x + y$ and $x = fl(a + b)$.

Below is the Kahan-Babušhka's [9] and Dekker's [10] classical algorithm which outputs the same solution for the same problem provided that $|a| \geq |b|$. It uses fewer ops but includes branches, which slow down the code optimization.

Algorithm 2.2.2. *COMPENSATED SUMMATION OF TWO FLOATING-POINT NUMBERS*

$$\begin{aligned} function[x, y] &= Fast\ Two\ Sum(a, b) \\ x &= fl(a + b) \\ y &= fl\big((a - x) + b\big) \end{aligned}$$

One can extend either of the two algorithms to the summation of h floating-point numbers for any h by applying the Kahan-Babušhka Cascaded Summation. Below is its variant using Knuth's basic algorithm.

Algorithm 2.2.3. [9, 5]. *THE CASCADED SUMMATION*

$$\begin{aligned} \textit{Function res} &= \textit{Sum } 2s(p) \\ \pi_1 &= p_1 \\ \sigma_1 &= 0 \\ \textit{For } i &= 2 : h \\ [\pi_i, q_i] &= \textit{Two Sum}(\pi_{i-1}, p_i) \\ \sigma_i &= fl(\sigma_{i-1} + q_i) \\ res &= fl(\pi_h + \sigma_h) \end{aligned}$$

The algorithm outputs the approximation res to the sum $\sum_{i=1}^{h} s_i$ of h numbers with an error of at most $\left(\frac{h}{(2^p - h)}\right) \sum_{i=1}^{h} |s_i|$.

2.2.2 Floating-point multiplication

The floating-point multiplication algorithm by Veltkamp employs a method of Dekker to split a floating-point number into two parts.

Here are the Dekker's and Veltkamp's algorithms. We assume that g is an integer such that $0 < g \leq p$.

Algorithm 2.2.4. [10]. *SPLITTING OF A FLOATING-POINT NUMBER INTO*

TWO PARTS

$$\begin{aligned} Function[x, y] &= split(a) \\ c &= fl(factor \cdot a) \quad factor = 2^g + 1 \\ x &= fl(c - (c - a)) \\ y &= fl(a - x) \end{aligned}$$

Remark 2.2.1. 1. The shorter precision numbers x and y satisfy the equation $a = x2^g + y$. Under the common assumption that $0 \leq \lceil \frac{p}{2} \rceil - g \leq 1$, these are the half-precision numbers. For any integer g, the output value $y = a \mod 2^g = \sum_{i<g} a_i$ is the residual modulo 2^g of a binary number $a = \sum_{i \leq e} a_i$, $a_e = 1$, obtained by zeroing all its bits that represent the powers 2^i for $i \geq g$.

2. Algorithm 2.2.4 also computes the remaining leading part $x = \frac{a-y}{2^g}$ in the binary floating-point representation of the number a. We have $a = 2^g x$ for $g < e - p - 1$, where $a = \sum_{i=e-p-1}^{e} a_i$ and $a = y$ for $g > e$.

Algorithm 2.2.5. [10]. *TRANSFORMATION OF THE PRODUCT OF TWO FLOATING-POINT NUMBERS*

$$\begin{aligned} function[x, y] &= Two\ Product(a, b) \\ x &= fl(a \cdot b) \\ [a_1, a_2] &= split(a) \\ [b_1, b_2] &= split(b) \\ y &= fl\Big(a_2 \cdot b_2 - \Big(\big((x - a_1 b_1) - a_2 b_1\big) - a_1 b_2\Big)\Big) \end{aligned}$$

The output floating-point numbers x and y satisfy the equation $a \cdot b = x + y$ and $x = fl(a \cdot b)$, so that the two latter algorithms reduce the multiplication to an addition. By combining the summation and the multiplication algorithm, one can handle a sequence of floating-point operations. In particular, one can use this method to calculate dot products.

2.3 Iterative refinement/improvement

All norms used in this section are the 2-norm unless otherwise stated.

Definition 2.3.1. The iterative refinement/improvement algorithm is a technique for improving the computed approximate solution $\hat{x}$ of a linear system $Ax = b$. The process consists of three steps [4], [6], [14]:

1) Compute the residual $r = b - A\hat{x}$.

2) Solve $Ad = r$ using some basic solution method such as the Gaussian Elimination with Partial Pivoting (GEPP).

3) Update: $y = \hat{x} + d$.

Repeat steps 1-3 with $\hat{x}$ replaced by y until the approximation to the solution $\hat{x}$ is accurate enough.

Remark 2.3.1. 1. If there were no rounding errors in the computation of r, d and y, then y would be the exact solution to the system. The idea behind the refinement/improvement algorithm is that if r and d are computed accurately enough, then some improvement in the accuracy of the solution is obtained.

2. The iterative refinement/improvement algorithm can reuse the factorization of A in every recursive step of the refinement.

3. The iterative refinement/improvement algorithm is commonly used with Gaussian Elimination (GE), where the residuals r are computed with extended precision. Iterative refinement/improvement for GE was used in the 1940s on desk calculators, but the first thorough analysis of the method was given by Wilkinson in 1963.

Describing the next algorithm, we write (u) and $(\bar{u})$ to denote the operations performed in single precision and in double precision, respectively.

Algorithm 2.3.1. *BASIC ITERATIVE REFINEMENT/IMPROVEMENT*

Input: An $n \times n$ matrix A, a computed solution $\hat{x} = x_1$ to $Ax = b$ and a vector b.

Output: A solution vector x_i approximating x in $Ax = b$ and an error bound $\frac{\|x-x_i\|}{\|x\|}$.

Initialize: $i \leftarrow 1$

Computations:

1) *Compute the residual* $r_i = b - Ax_i$ *in double precision* $(\bar{u})$

2) *Solve* $Ad_i = r_i$ *in single precision (u) using the GEPP*

3) *Update* $x_{i+1} = x_i + d_i$ *in double precision* $(\bar{u})$
$i \leftarrow i + 1$

Repeat stages 1-3 *until* x_i *is accurate enough.*

Output x_i *and an error bound.*

2.3.1 Error analysis of iterative refinement/improvement

The iterative refinement algorithm for GE/GEPP usually behaves as follows.

If double precision is used in the computation of r_i and x_{i+1} and if A is not too ill conditioned, then the iterative refinement/improvement algorithm produces a solution correct to the working precision, and the rate of convergence depends on the condition number of the matrix A. We assume that the computed solution $\hat{x}$ to a linear system $Ax = b$ satisfies

$$(A + \Delta A)\hat{x} = b, \ \|A^{-1}\Delta A\|_p < 1$$

with $p = 2$ or ∞. ΔA is the perturbation of the matrix A, which is perturbed into $\hat{A} = A + \Delta A$. The computation of r_i, d_i, and x_{i+1} from iterative refinement/improvement

in Algorithm 2.3.1 relies on the following expressions (cf. Theorem 2.1.5 in Section 2.1):

(i) $r_i = b - Ax_i + \Delta r_i$ where $\|\Delta r_i\|_\infty \leq n\bar{u}(\|b\|_\infty + \|A\|_\infty\|x_i\|_\infty)$ represents the error of computing r_i;

(ii) $d_i = (A + \Delta A_i)^{-1} r_i$ where $\|\Delta A_i\|_\infty \leq 3nu\|L\|_\infty\|U\|_\infty$ represents the error of inverting A;

(iii) $x_{i+1} = x_i + d_i + \Delta x_{i+1}$ where $\|\Delta x_{i+1}\|_\infty \leq \bar{u}\|x_{i+1}\|_\infty$ represents the error of computing x_{i+1}. (Recall that u and $\bar{u}$ are the unit roundoff or machine precision in single and in double precision, respectively).

As long as A is not too ill conditioned and the solver is not too unstable, we estimate (cf. [6]) that the limiting normwise accuracy, that is the minimum of the ratio $\frac{\|x-x_i\|_\infty}{\|x\|_\infty}$ is roughly $2n\bar{u}cond_\infty A + u$. We can obtain a componentwise relative error of order μu that is, $\min_i \|x - x_i\| \lesssim \mu u\|x\|_\infty$ for some constant μ.
In this thesis, we prove that the above bound can be attained with an ill conditioned matrix if we use additive preconditioning along with iterative refinement/improvement and error-free MSAs.

In case the solver uses LU factorization, we can derive the following theorem.

Theorem 2.3.2. [6, *page* 234]. *Let iterative refinement/improvement based on LU factorization be applied to a nonsingular linear system* $Ax = b$, *and let the residuals*

be computed in double precision. Let $\eta = u\| \, |A^{-1}| \, |\hat{L}| \, |\hat{U}| \, \|_\infty$ *be substantially smaller than one where* $\hat{L}$ *and* $\hat{U}$ *denote the computed LU factors of* A*. Then iterative refinement/improvement reduces the error by a factor of approximately* η *at each stage, until* $\frac{\|x-\hat{x}_i\|_\infty}{\|x\|_\infty} \approx u$.

Remark 2.3.2. This theorem is stronger than the standard results in the literature, which have $\text{cond}_\infty(A)u$ in place of η. We can have $\eta \ll \text{cond}_\infty(A)u$, since η is independent of the row scaling of A (up to the changes in the pivot sequence).

For example, if $|\hat{L}||\hat{U}| = |A|$ in the above theorem then $\eta \approx \text{cond}(A)u$, and cond A can be arbitrarily smaller than $\text{cond}_\infty A$.

2.3.2 Convergence theorem

We follow [4] and rewrite the iterative refinement/improvement algorithm. We write g_0 for the error in the computation of r_0 and write h_0 for the error in the computation of x_1. We write $A_0 = A + E_0$, where E_0 is the perturbation to the matrix A. x_0 is a computed solution of the linear system $Ax = b$.

1) $r_0 = b - Ax_0 + g_0$

2) $d_0 = A_0^{-1} r_0$

3) $x_1 = x_0 + d_0 + h_0$

Repeat stages 1-3.

The algorithm yields a sequence of approximate solutions $x_0, x_1, \ldots$. In general, the sequence does not converge, but the following theorem shows some sufficient conditions for convergence.

Theorem 2.3.3. [4]. *Let the above algorithm be applied iteratively to give the sequence* $x_0, x_1, \ldots$. *Let* $e_i = x_i - x$ *and* $E_i = A_i - A$. *If* $\frac{\|A^{-1}E_i\|}{1-\|A^{-1}E_i\|} \leq \rho < 1$, $\|g_i\| \leq \gamma_i$ *and* $\|h_i\| \leq \eta_i$, $i = 0, 1, \ldots$. *Then*

$$\|e_i\| \leq \rho^i \|e_0\| \; + \; (1+\rho)\|A^{-1}\|(\gamma_i + \rho\gamma_{i-1} + \ldots + \rho^i\gamma_0)$$
$$+(\eta_i + \rho\eta_{i-1} + \ldots + \rho^i\eta_0).$$

Proof. By Theorem 1.2.1 (i) of Chapter 1, we have

$$(I + A^{-1}E_i)^{-1} = I + F_i, \text{ where } \|F_i\| \leq \rho.$$

Hence $\|(A + E_i)^{-1}\| \leq (1+\rho)\|A^{-1}\|$.

$$\begin{aligned}
\text{Now,} \quad x_1 &= x_0 + (A + E_0)^{-1}(b - Ax_0 + g_0) + h_0 \\
&= x_0 + (I + A^{-1}E_0)^{-1}A^{-1}(b - Ax_0) + (A + E_0)^{-1}g_0 + h_0 \\
&= x_0 - (I + A^{-1}E_0)^{-1}e_0 + (A + E_0)^{-1}g_0 + h_0 \\
&= x_0 - (I + F_0)e_0 + (A + E_0)^{-1}g_0 + h_0 \\
&= x - F_0e_0 + (A + E_0)^{-1}g_0 + h_0.
\end{aligned}$$

Therefore, $e_1 = x_1 - x = -F_0 e_0 + (A + E_0)^{-1} g_0 + h_0$. Hence

$$\begin{aligned} \|e_1\| &\leq \|F_0\|\|e_0\| + \|(A + E_0)^{-1}\|\|g_0\| + \|h_0\| \\ &\leq \rho\|e_0\| + (1 + \rho)\|A^{-1}\|\gamma_0 + \eta_0. \\ \text{Similarly,} \quad \|e_2\| &\leq \rho\|e_1\| \ + \ (1 + \rho)\|A^{-1}\|\gamma_1 + \eta_1. \text{ So} \\ \|e_2\| &\leq \rho^2\|e_0\| + (1 + \rho)\|A^{-1}\|(\gamma_1 + \rho\gamma_0) + \eta_1 + \rho\eta_0. \end{aligned}$$

The proof is readily completed by means of induction.

□

Chapter 3

The Schur aggregation and linear systems, the convergence theorem and error analysis

3.1 The Schur aggregate and its condition number

Next we analyze how the singular values (and thereafter the norm and conditioned number) of the Schur aggregate depend on the singular values of the matrices A and C.

Theorem 3.1.1. [4, 14]. *Assume the matrices A, U, V, and $C = A + UV^H$ of sizes $m \times n$, $m \times r$, $n \times r$ and $m \times n$, respectively, with full rank where the ratio r/n is*

small. Then the generalized inverse of A can be expressed as follows.

$$A^- = C^- + C^- U (I_r - V^H C^- U)^{-1}\ V^H C^-$$

where $C = A + UV^H$, $A^- A = I_n$ for $m \geq n$, $AA^- = I_m$ for $m \leq n$ and $S = I_r - V^H C^- U$ is the Schur complement of the block C of the block matrix $\begin{pmatrix} C & V^H \\ U & I_r \end{pmatrix}$.

Proof. (Cf. [1].) For $m \leq n$, the matrix $I_m - UV^H C^-$ is nonsingular, we have $C = A + UV^H$,

$$A = C - UV^H = (I_m - UV^H C^-)C, \tag{3.1.1}$$

$$A^- = C^- (I_m - UV^H C^-)^{-1}, \tag{3.1.2}$$

whereas for $m \geq n$, the matrix $I_n - C^- UV^H$ is nonsingular and we have

$$A = C(I_n - C^- UV^H), \tag{3.1.3}$$

$$A^- = (I_n - C^- UV^H)^{-1} C^-. \tag{3.1.4}$$

For $m \leq n$, substitute $C \leftarrow I_m$ and $V^H \leftarrow V^H C^-$ into the SMW formula in Theorem 1.3.2 to get

$$(I_m - UV^H C^-)^{-1} = I_m + U(I_r - V^H C^- U)^{-1}\ V^H C^-. \tag{3.1.5}$$

For $m \geq n$, substitute $C \leftarrow I_n$ and $U \leftarrow C^- U$ into the SMW formula to get

$$(I_n - C^- UV^H)^{-1} = I_n + C^- U(I_r - V^H C^- U)^{-1} V^H. \tag{3.1.6}$$

We complete the proof by combining (3.1.2) and (3.1.5), and (3.1.4) and (3.1.6). We have $A^-A = I_n$ for $m \geq n$, and $AA^- = I_m$ for $n \geq m$. □

By post-multiplying $A^- = C^- + C^-U(I - V^HC^-U)^{-1}V^HC^-$ in the above theorem by a vector b, we express the solution of the linear system of equations $Ax = b$ via the solutions of some linear systems with the matrix C (A-modification) and the Schur aggregate $S = I_r - V^HC^-U$, that is

$$A^-b = C^-b + C^-U(I_r - V^HC^-U)^{-1}V^HC^-b$$

or equivalently

$$A^-b = C^-b + C^-US^{-1}V^HC^-b.$$

We consider the case where the matrices C and S are well conditioned, whereas the matrices U and V have small rank r, so that we can solve the above linear systems with the matrices C and S faster and more accurately than the systems with the matrix A (cf. [1]). In this case the original conditioning problems for a linear system $Ax = b$ are restricted to the computation of the Schur aggregate S.

Let us follow [1] to supply some technical details. First, we estimate the j^{th} largest singular value of the matrix S^{-1} for $j = 1, ..., r$ in terms of the singular values $\sigma_j(A^-)$, $\sigma_1(C)$, and $\sigma_\rho(C)$ of the matrices A^- and C.

Theorem 3.1.2. [1]. *Let W denote an $m \times n$ matrix of full rank $\rho = \min\{m, n\}$. Write $\sigma_+(W) = \sigma_1(W), \sigma_-(W) = \sigma_\rho(W)$. Then we have $\sigma_j(M)\sigma_-(W) \leq \sigma_j(MW) \leq$*

$\sigma_j(M)\sigma_+(W)$ *and*

$\sigma_j(M)\sigma_-(W) \leq \sigma_j(WN) \leq \sigma_j(N)\sigma_+(W)$, *for* $j = 1, \ldots, \rho$ *and* $\rho \times \rho$ *matrices* M *and* N.

Proof. Since the singular values are invariant in multiplication with a unitary matrix, it is sufficient to consider the case of a positive diagonal matrix W. In that case, the claimed bounds readily follow from the Courant-Fischer Minimax characterization. □

Theorem 3.1.3. *Under the assumption of Theorem* 3.1.2, *we have*

$\sigma_j(W) - 1 \leq \sigma_j(W + I_n) \leq \sigma_j(W) + 1$ *for* $n \times n$ *matrix* W *and* $j = 1, 2, \ldots, n$.

Proof. In [14], Theorem 3.3.3, take $E = I_n$. □

Theorem 3.1.4. [1]. *For positive integers* m, n, *and* r, *a normalized* $m \times n$ *matrix* A, *and a pair of unitary matrices* U *of size* $m \times r$ *and* V *of size* $n \times r$, *write* $C = A + UV^H$ *and* $S = I_r - V^H C^- U$. *Suppose the matrices* A *and* $C = A + UV^H$ *have full rank* $\rho \geq r$. *Then the matrix* S *is nonsingular, and we have*

$\sigma_j(A^-)\sigma_-^2(C) - \sigma_-(C) \leq \sigma_j(S^{-1}) \leq \sigma_j(A^-)\sigma_+^2(C) + \sigma_+(C)$

for $\sigma_-(C) = \sigma_\rho(C)$, $\sigma_+(C) = \sigma_1(C) \leq 2$, *and* $\sigma_j(A^-) = 1/\sigma_{\rho-j+1}(A)$ *with* $j = 1, \ldots, r$.

Proof. Let $m \geq n$, from Equations (3.1.3) and (3.1.4), $S_n = I_n - C^- UV^H$ is nonsingular. The matrix S is nonsingular as well because $\det S = \det S_n$. Next, combine

Equation (3.1.4) with Theorem 3.1.3 for $M = S_n^{-1}$, $W = C^-$ and $A^- = MW$, to obtain that

$\sigma_j(S_n^{-1})\sigma_-(C^-) \leq \sigma_j(A^-) \leq \sigma_j(S_n^{-1})\sigma_+(C^-)$ for $j = 1, ..., \rho$. We then substitute $\sigma_-(C^-) = 1/\sigma_+(C)$ and $\sigma_+(C^-) = 1/\sigma_-(C)$, and obtain that

$$\sigma_j(A^-)\sigma_-(C) \leq \sigma_j(S_n^{-1}) \leq \sigma_j(A^-)\sigma_+(C), for\, j = 1, \ldots, \rho. \tag{3.1.7}$$

Taking $W = C^-U$ and $N = S^{-1}$ in Theorem 3.1.3, and using the equation $\sigma_j(C^-US^{-1}V^H) = \sigma_j(C^-US^{-1})$ for $j = 1, \ldots, r$ along with the inequalities

$\sigma_-(C^-U) \geq \sigma_-(C^-) = 1/\sigma_+(C)$, and $\sigma_+(C^-U) \leq \sigma_+(C^-) = 1/\sigma_-(C)$, we get

$\sigma_j(S^{-1})/\sigma_+(C) \leq \sigma_j(C^-US^{-1}V^H) \leq \sigma_j(S^{-1})/\sigma_-(C)$ for $j = 1, ..., r$. Combine the latter bounds with Theorem 3.1.4 for $W = C^-US^{-1}V^H$ and Equation (3.1.6) to deduce that $\sigma_j(S^{-1})/\sigma_+(C) - 1 \leq \sigma_j(S_n^{-1}) \leq \sigma_j(S^{-1})/\sigma_-(C) + 1$. Therefore

$(\sigma_j(S_n^{-1}) - 1)\sigma_-(C) \leq \sigma_j(S^-) \leq (\sigma_j(S_n^{-1}) + 1)\sigma_+(C)$ for $j = 1, ..., r$. We obtain the claimed bounds in the case of $m \geq n$, by combining the latter bound with equation (3.1.7). For $m \leq n$, one proceeds similarly using the equations (3.1.2) and (3.1.5) instead of (3.1.4) and (3.1.6), replace S_n with $S_m = I_m - UV^HC^-$. Furthermore, recall Theorem 3.1.3 at the first and second time, replace $M = S_n^{-1}$ with $N = S_m^{-1}$, and replace $W = C^-U$ with $W = V^HC^-$. □

Corollary 3.1.5. [1]. *Under the assumption of Theorem* 3.1.4, *we have*

$cond_2\, S = cond_2\,(S^-) \leq (cond_2\, C)(\sigma_1(A^-)\sigma_+(C) + 1)/(\sigma_r(A^-)\sigma_-(C) - 1)$,

$\|S\| = \sigma_1(S) = 1/\sigma_j(S^{-1}) \leq 1/(\sigma_r(A^-)\sigma_-^2(C) - \sigma_-(C))$.

Remark 3.1.1. Suppose A is an $n \times n$ nonsingular matrix with $nnul A = r$ and UV^H is a random, well conditioned and properly scaled APP of rank r. Then the values $\sigma_{n-j+1}(A)/\sigma_1(A)$ are small for $j \leq r$ and not small for $j > r$, whereas the value $\sigma_n(C)$ is expected to be of the order of $\sigma_{n-r}(A) >> \sigma_{n-r+1}(A)$. Therefore, all the singular values $\sigma_{r-j+1}(S) = 1/\sigma_j(S^{-1})$ for $j = 1, ..., r$ are expected to be of the order of at most $\sigma_{n-j+1}(A)$. Furthermore (cf. Corollary 3.1.5), $cond_2 S$ is likely to be of the order of at most $(cond_2 C)^2 \sigma_{n-r+1}(A)/\sigma_n(A)$, whereas $\|S\| = \sigma_1(S)$ is likely to be at most of the order of $\sigma_{n-r+1}(A)/\sigma_n^2(C)$. We conclude that the matrix S is expected to have a small norm if C is well conditioned and $nnul A = r > 0$.

The following algorithm highlights the computation of the matrices C, S, and A^{-1}. We use the matrices U and V whose entries can be rounded to a fixed (small) number of bits to control or avoid rounding errors in computing the matrix $C = A + UV^H$. In each recursive step, we choose the rank of an APC according to some fixed policy RANK, e.g., in every step we set $r = 1$ or let r be the minimum rank of an APC UV^H for which the matrix $C = A + UV^H$ is well conditioned.

Algorithm 3.1.6. *Recursive A-preconditioning and Schur aggregation.*

INPUT: a nonsingular $n \times n$ matrix A and a policy RANK.

OUTPUT: the matrix A^{-1}.

COMPUTATION:

0. Choose a positive integer r according to the policy RANK.

1. *Generate the pair of normalized random* $n \times r$ *matrices* $\tilde{U}$ *and* V *such that* $\|\tilde{U}\| = \|V\| = 1$.

2. *Compute a crude estimate* ν *for the norm* $\|A\|$.

3. *Compute the matrix* $U = \nu\tilde{U}$.

4. *Compute the* $n \times n$ *matrix* $C = A + UV^H$ *and its inverse* C^{-1}. *If this matrix is ill conditioned, set* $A \leftarrow C$ *and reapply the algorithm.*

5. *Compute the* $r \times r$ *matrix* $S = I_r - V^H C^{-1} U$ *and its inverse. (The computation of the matrix* S *may require high precision due to the cancelation of the leading bits in the representation of the entries.) If this matrix is ill conditioned, set* $A \leftarrow S$ *and reapply the algorithm.*

6. *Compute and output the* $n \times n$ *matrix* $A^{-1} = C^{-1} + C^{-1}US^{-1}V^H C^{-1}$ *and stop.*

3.2 Solving linear systems with APC's and iterative refinement/improvement

Let A and C be nonsingular $n \times n$ matrices, and let UV^H be an APC of a smaller rank r filled with short binary numbers, such that the matrix A is ill conditioned and the matrix $C = A + UV^H$ is well conditioned. We use the equation $A^{-1} = C^{-1} + C^{-1}U(I_r - V^H C^{-1}U)^{-1}V^H C^{-1}$ to reduce the solution of the linear system $Ax = b$ to

the computation of the vector $C^{-1}b$ and the matrix $C^{-1}U(I_r - V^HC^{-1}U)^{-1}V^HC^{-1}b$ (or $C^{-1}US^{-1}V^HC^{-1}b$), for a well conditioned matrix C. Since the norm $\|S\|_2$ is small, we must compute the Schur aggregate $S = I_r - V^HC^{-1}U$ with a higher precision. We apply MSAs in this computation. We also use the extension in [1] of Wilkinson's iterative refinement/improvement to computing the matrix $W = C^{-1}U$ with extended precision. In its classical form (see Section 2.3) the algorithm is applied to a single system $Cw = u$. It is easy to extend it to the matrix equation $CW = U$, but in the classical version, also the refinement stops when the matrix $W = C^{-1}U$ is computed with at most double precision. We apply a variant in [1] where the residuals dynamically decrease, which is a must for us. We represent the output value as the sum of fixed-precision numbers. Furthermore, we use the error-free floating point summation and multiplication in Section 2.2 to eliminate the rounding errors when we compute the residuals and sum $W = W_0 + W_1 + \cdots + W_k$.

In the next sections we follow [1] to prove convergence of the sum $W_0+W_1+...+W_k$ to $W = C^{-1}U$.

3.2.1 Extended iterative refinement/improvement

Let us specify and analyze the iterative refinement/improvement extended to both solving the matrix equation $CW = U$ and extending the solution from double to extended precision. Now the goal is to compute the matrices $W = \sum_{i=1}^{k} W_i$ and

$S = I_r - V^H W = I_r + \sum_{i=1}^{k} F_i$ for a sufficiently large k. We write $U_0 = U$ and $S_0 = I_r$, and successively compute the matrices

$$\begin{aligned} W_i &\leftarrow C^{-1}U_i, \\ U_{i+1} &\leftarrow U_i - CW_i, \\ F_i &\leftarrow -V^H W_i, \\ S_{i+1} &\leftarrow S_i + F_i \; for \; i = 0, 1, \ldots, k. \end{aligned}$$

Remark 3.2.1. 1. Theorem 3.1.4 defines a small upper bound on the norm $||S||$ if the matrix A is ill conditioned and the matrix C is well conditioned. Therefore, we can have $S_i \approx 0$ for $i = 0, 1, \ldots, k$ and some positive integer k. At the i^{th} step of iterative refinement/ improvement for $i \leq k$, we need to store only the most recently computed matrix S_{i+1} overwriting S_i. Similarly, we can overwrite the matrices W_{i-1}, U_i, and F_{i-1} with their updates W_i, U_{i+1}, and F_i, to save memory space.

2. The matrices U and V are chosen up to a perturbation within a fixed small norm as long as this perturbation keeps the A-modification $C = A + UV^H$ well conditioned. Likewise, we require that the matrices C^{-1} and $W_i \leftarrow C^{-1}U_i$ be computed within an error bound that ensures the decrease of the residual norms $u_i = ||U_i||$ (and consequently the error norm $e_i = ||E_i||$ since $E_i = C^{-1}U$) by a fixed factor $1/\theta$ exceeding one in each iteration.

3. Within the allowed perturbation norm, we vary the matrices U, V, C^{-1}, and W_i for all i to decrease the number of bits in the binary representation of their entries.

First, we estimate from above the norm of the input perturbation and the precision of computing that ensure the output error norm within the fixed tolerance bound. Next, we perturb the input within the estimated error norm to represent it with fewer bits. In particular, we set to zero the absolutely smaller input entries and round every other entry to fewer bits. Finally, we perform the extended iterative refinement/improvement and verify that it converges as expected, or otherwise we correct our policy of input perturbation.

The following theorems give us an estimate for the errors E_i and the parameter $1/\theta$.

Theorem 3.2.1. [1]. *Consider the sub-iteration*

$$\begin{aligned} W_i &\leftarrow fl(C^{-1}U_i) = C^{-1}U_i - E_i \\ U_{i+1} &\leftarrow U_i - CW_i \end{aligned}$$

for $i = 0, 1, \ldots, k$ *and* $U = U_0$. *Then*

$C(W_0 + \cdots + W_k) = U - CE_k$.

Proof. We have $CW_i = U_i - U_{i+1}, i = 0, 1, \ldots, k-1$. Sum this latter equation to obtain that $C(W_0 + \cdots + W_{k-1}) = U_0 - U_k$. Substitute the equations $U_0 = U$ and $U_k = CW_k + CE_k$ and obtain the theorem. □

The theorem implies that the sum $W_0 + \cdots + W_k$ approximates the matrix $W = C^{-1}U$ with the error matrix $-E_k$. It remains to show that the error term E_i converges

to zero as $i \to \infty$.

Theorem 3.2.2. [1]. *Assume that* $W_i = (C - \tilde{E}_i)^{-1}U_i = C^{-1}U_i - E_i$ *for all* i. *Write* $e_i = \|E_i\|$, $u_i = \|U_i\|$, *and* $\theta_i = \delta_i \|C\|$ *where* $\delta_i = \delta(C, \tilde{E}_i) = 2\|\tilde{E}_i\|_F max\{\|C^{-1}\|^2, \|(C - \tilde{E}_i)^{-1}\|^2\}$. *Then we have* $e_i \leq \delta_i u_i$ *for all* i, *and* $e_{i+1} \leq \theta_i e_i$, $u_{i+1} \leq \theta_i u_i$ *for* $i = 0, 1, \ldots, k-1$.

Theorem 3.2.3. [1]. *We have* $U_{i+1} = CE_i$ *and consequently* $u_{i+1} \leq e_i \|C\|$ *for all* i.

Proof. Pre-multiply the matrix equation $C^{-1}U_i - W_i = E_i$ by C and add the resulting equation to the equation $U_{i+1} - U_i + CW_i = 0$. □

Lemma 3.2.4. *Let* C *and* $C + E$ *be two nonsingular matrices. Then*

$$\|(C+E)^{-1} - C^{-1}\| \leq \|(C+E)^{-} - C^{-}\|_F \leq 2\|E\|_F max\{\|C^{-1}\|^2, \|(C-E)^{-1}\|^2\}$$

Proof. See [14, Section 5.5.5]. □

Corollary 3.2.5. [1]. *Assume that* $W_i = (C - \tilde{E}_i)^{-1}U_i = C^{-1}U_i - E_i$. *Then* $e_i \leq \delta_i u_i$ *where* $\delta_i = \delta(C, \tilde{E}_i) = 2\|\tilde{E}_i\|_F max\{\|C^{-1}\|^2, \|(C - \tilde{E}_i)^{-1}\|^2\}$, $e_i = \|E_i\|$, *and* $u_i = \|U_i\|$.

Combining Theorem 3.2.3 and Corollary 3.2.5, we obtain $u_{i+1} \leq \theta_i u_i$ and $e_{i+1} \leq \theta_i e_i$ for $\theta_i = \delta_i \|C\|$ and for all i.

Remark 3.2.2. For $\theta = max_i \theta_i < 1$, the theorem shows linear convergence of the error norms e_i to zero as $i \to \infty$. This implies linear convergence of the sum $W_0 + W_1 + \ldots + W_k$ to W, $U_0 + U_1 + \ldots + U_k$ to U, $F_0 + F_1 + \cdots + F_k$ to F, and S_k to S.

It remains to estimate θ_i.

Corollary 3.2.6. [1]. *Assume that the matrix C is well conditioned. Then we have* $\lim_{i\to\infty} \theta_i = 0$.

Proof. The matrix C is well conditioned, therefore the ratios $r_i = \|\tilde{E}_i\|_F/\|C\|_F$ are small and $cond(C - \tilde{E}_i) \approx cond\, C$ (cf. [14, Section 3.3]). Then

$$\begin{aligned}
\theta_i &= \delta_i \|C\| \\
&= 2r_i max\{cond^2 C, cond^2(C - E_i)\} \|C\|_F / \|C\| \\
&\approx 2(cond\, C)^2 r_i \|C\|_F / \|C\| \\
&\leq 2(cond\, C)^2 r_i n < 1, \text{ for all } i.
\end{aligned}$$

□

3.3 The convergence of iterative refinement/improvement

Suppose A is an ill conditioned nonsingular $n \times n$ matrix with $nnul A = r$, UV^H is a random, well conditioned and properly scaled APC of rank $r < n$, and the A-modification $C = A + UV^H$ is well conditioned. Surely, a small norm perturbations of the generators U and V, caused by truncation of their entrees, keep the matrix

C well conditioned. We rewrite the iterative refinement/improvement algorithm to solve the linear system $CW = U$ with $U_0 = U$.

Algorithm 3.3.1.

$$CW_k = U_k \tag{3.3.1}$$

$$U_{k+1} = U_k - CW_k \tag{3.3.2}$$

$$X_k = W_0 + \cdots + W_k, \; for \; k = 0, 1, 2, ... \tag{3.3.3}$$

The solution W of the linear system $CW = U$ is computed by means of Gaussian Elimination with Partial Pivoting (hereafter GEPP). It is corrupted by rounding errors of the computation of W_k in (3.3.1), so that the computed matrix W_k turns into $(C + F_k)^{-1}U_k$. F_k is the perturbation to the matrix C. Another source of error is the computation in (3.3.2), which numerically turns into the equation $U_{k+1} = U_k - CW_k + E_k$, where E_k is an error matrix. (Recall that the summation (3.3.3) is error free (see Section 2.2.1).) Factoring in the above errors, Algorithm 3.3.1 is executed as follows.

Algorithm 3.3.2.

$$\begin{aligned} W_0 &= C_0^{-1}U_0 \quad (U_0 = U \; and \; C_0 = C + F_0) \\ W_k &= (C + F_k)^{-1}U_k \\ U_{k+1} &= U_k - CW_k + E_k \\ X_k &= W_0 + W_1 + \cdots + W_k, \; for \; k = 0, 1, 2, ... \end{aligned}$$

Algorithm 3.3.2 generates a sequence $X_0, X_1, X_2, ...$ of approximate solutions. The question that we address next is the following. Does the sequence of approximate solutions $X_0, X_1, X_2, ...$ converge to the solution W of $CW = U$? The following theorem answers this question. We call this theorem the convergence theorem of iterative refinement/improvement with ill conditioned matrix.

Theorem 3.3.3. *Solve the linear system $CW = U$, derived from the ill conditioned linear system $Ax = b$, by applying the following additive preconditioning algorithm.*

$$W_0 = C_0^{-1}U_0 \quad (U_0 = U \, and \, C_0 = C + F_0)$$

$$W_k = (C + F_k)^{-1}U_k \tag{3.3.4}$$

$$U_{k+1} = U_k - CW_k + E_k \tag{3.3.5}$$

$$X_k = W_0 + \cdots + W_k, \; for \; k = 0, 1, 2, ...$$

Denote $F_k = C_k - C$.

If

$$\frac{\|C^{-1}F_k\|}{1 - \|C^{-1}F_k\|} \leq \rho < 1 \, and \tag{3.3.6}$$

$$\|E_k\| \leq \gamma_k \, for \, k = 0, 1, ..., \tag{3.3.7}$$

then

$$\|X_k - X\| \leq \rho^k \|X_0 - X\| + (1 + \rho)\|C^{-1}\|(\gamma_k + \rho\gamma_{k-1} + \cdots + \rho^{k-1}\gamma_1).$$

In other words, $\|X_k - X\|$ is bounded by $\mathcal{O}(\gamma_k)$ for a certain integer k.

Proof. We have

$$(I + C^{-1}F_k)^{-1} = I + \acute{E}_k \, where \, \|\acute{E}_k\| \leq \rho \tag{3.3.8}$$

due to Theorem 1.2.1 in Chapter 1 and the hypothesis (3.3.6) $\frac{\|C^{-1}F_k\|}{1-\|C^{-1}F_k\|} \leq \rho < 1$. Next observe that $(I + C^{-1}F_k)^{-1} = [C^{-1}(C + F_k)]^{-1} = (C + F_k)^{-1}C$.

$$\begin{aligned}
\text{Since } (I + C^{-1}F_k)^{-1} &= I + \acute{E}_k \text{ from (3.3.8), we have} \\
(C + F_k)^{-1}C &= I + \acute{E}_k, \\
(C + F_k)^{-1} &= (I + \acute{E}_k)C^{-1} \\
&= C^{-1} + \acute{E}_k C^{-1}, \text{ and so} \\
\|(C + F_k)^{-1}\| &\leq \|C^{-1}\| + \|\acute{E}_k\|\|C^{-1}\|, \\
\|(C + F_k)^{-1}\| &\leq \|C^{-1}\| + \rho\|C^{-1}\| \text{ using } \|\acute{E}_k\| \leq \rho,
\end{aligned}$$

$$\|(C + F_k)^{-1}\| \leq (1 + \rho)\|C^{-1}\|. \tag{3.3.9}$$

$$X_0 = W_0 = C_0^{-1}U_0.$$

From (3.3.3), we have $X_1 = W_0 + W_1$ and successively obtain that

$$X_1 = W_0 + C_1^{-1}U_1 \text{ using (3.3.4)},$$

$$X_1 = X_0 + C_1^{-1}(U_0 - CW_0 + E_0),$$

$$F_k = C_k - C \text{ so } F_1 = C_1 - C \text{ and } C_1 = F_1 + C,$$

$$X_1 = X_0 + (C + F_1)^{-1}(U_0 - CW_0 + E_0),$$

$$X_1 = X_0 + (C + F_1)^{-1}(U_0 - CW_0) + (C + F_1)^{-1}E_0,$$

$$X_1 = X_0 + [C(I + C^{-1}F_1)]^{-1}(U_0 - CW_0) + (C + F_1)^{-1}E_0,$$

$$X_1 = X_0 + (I + C^{-1}F_1)^{-1}C^{-1}(U_0 - CW_0) + (C + F_1)^{-1}E_0,$$

$$X_1 = X_0 + (I + C^{-1}F_1)^{-1}(C^{-1}U_0 - W_0) + (C + F_1)^{-1}E_0.$$

We obtain by using (3.3.8) that

$$X_1 = X_0 + (I + \acute{E}_1)(C^{-1}U_0 - W_0) + (C + F_1)^{-1}E_0,$$

$$X_1 = X_0 + (I + \acute{E}_1)(X - X_0) + (C + F_1)^{-1}E_0,$$

$$X - X_0 = -(X_0 - X),$$

$$X_1 = X_0 - (I + \acute{E}_1)(X_0 - X) + (C + F_1)^{-1}E_0,$$

$$X_1 = X_0 - (X_0 - X) - \acute{E}_1(X_0 - X) + (C + F_1)^{-1}E_0,$$

$$X_1 = X_0 - X_0 + X - \acute{E}_1(X_0 - X) + (C + F_1)^{-1}E_0,$$

$$X_1 = X - \acute{E}_1(X_0 - X) + (C + F_1)^{-1}E_0.$$

Taking the norm on both sides we obtain that

$$\|X_1 - X\| = \| - \acute{E}_1(X_0 - X) + (C + F_1)^{-1}E_0\|,$$

$$\|X_1 - X\| \leq \|\acute{E}_1\|\|X_0 - X\| + \|(C + F_1)^{-1}\|\|E_0\|.$$

By using the hypothesis $\|\acute{E}_1\| \leq \rho$, we have

$\|X_1 - X\| \leq \rho\|X_0 - X\| + \|(C + F_1)^{-1}\|\|E_1\|$. Due to the bound

$$\|E_k\| \leq \gamma_k, \tag{3.3.7}$$

the inequality turns into

$$\|X_1 - X\| \leq \rho\|X_0 - X\| + \|(C + F_1)^{-1}\|\gamma_1, \text{ that is}$$

$$\|X_1 - X\| \leq \rho\|X_0 - X\| + (1 + \rho)\|C^{-1}\|\gamma_1, \text{ due to the inequality (3.3.9).}$$

The same argument leads to

$$\|X_2 - X\| \leq \rho\|X_1 - X\| + (1 + \rho)\|C^{-1}\|\gamma_2, \text{ so that}$$

$$\|X_2 - X\| \leq \rho(\rho\|X_0 - X\| + (1 + \rho)\|C^{-1}\|\gamma_1) + (1 + \rho)\|C^{-1}\|\gamma_2.$$

$$\|X_2 - X\| \leq \rho^2\|X_0 - X\| + (1 + \rho)\|C^{-1}\|(\gamma_2 + \rho\gamma_1) \tag{3.3.10}$$

and

$$\|X_3 - X\| \leq \rho\|X_2 - X\| + (1 + \rho)\|C^{-1}\|\gamma_3,$$

which gives us

$$\|X_3 - X\| \leq \rho^3\|X_0 - X\| + (1 + \rho)\|C^{-1}\|(\gamma_3 + \rho\gamma_2 + \rho^2\gamma_1)$$

due to the inequality (3.3.10). By induction, we yield the claimed result

$$\|X_k - X\| \leq \rho^k\|X_0 - X\| + (1 + \rho)\|C^{-1}\|(\gamma_k + \rho\gamma_{k-1} + ... + \rho^{k-1}\gamma_1). \tag{3.3.11}$$

□

Corollary 3.3.4. *Under the assumption of Theorem* 3.3.3, *if* $\gamma_k \leq \gamma$ *for all* k *then*

$$\|X_k - X\| \leq \rho^k \|X_0 - X\| + \frac{1+\rho}{1-\rho}\|C^{-1}\|\gamma.$$

In particular if $\bar{\gamma} = \limsup \gamma_k$ *then*

$$\limsup \|X_k - X\| \leq \rho^k \|X_0 - X\| + \frac{1+\rho}{1-\rho}\|C^{-1}\|\bar{\gamma}.$$

Proof. This corollary is a straight-forward consequence of Theorem 3.3.3 as well as Theorem 1.2.3 in Chapter 1, which states that if $\lim_{k\to\infty} \rho^k = 0$, then $(I-P)^{-1} = \sum_{k=o}^{\infty} \rho^k$. Using (3.3.11), we deduce that

$\|X_k - X\| \leq \rho^k \|X_0 - X\| + (1+\rho)\|C^{-1}\|\|(\gamma_k + \rho\gamma_{k+1} + ... + \rho^{k-1}\gamma_1)$

Applying the bound $\gamma_k \leq \gamma$, we obtain

$\|X_k - X\| \leq \rho^k \|X_0 - X\| + (1+\rho)\|C^{-1}\|(\gamma + \rho\gamma + ... + \rho^{k-1}\gamma)$. Factoring out γ, we yield

$$\|X_k - X\| \leq \rho^k \|X_0 - X\| + (1+\rho)\|C^{-1}\|\gamma(1 + \rho + ... + \rho^{k-1})$$

$$\|X_k - X\| \leq \rho^k \|X_0 - X\| + (1+\rho)\|C^{-1}\|\gamma\frac{1-\rho^k}{1-\rho}.$$

This inequality for larger k turns into

$$\|X_k - X\| \leq \rho^k \|X_0 - X\| + \frac{1+\rho}{1-\rho}\|C^{-1}\|\gamma.$$

Next recall that

$$\bar{\gamma} = \limsup \gamma_k.$$

For all k, we have $\gamma_k \leq \bar{\gamma} = \limsup \gamma_k$. Therefore, inequality (3.3.11) in Theorem 3.3.3 gives us

$$\begin{aligned} \|X_k - X\| &\leq \rho^k \|X_0 - X\| + (1+\rho)\|C'^{-1}\|(\bar{\gamma} + \rho\bar{\gamma} + \ldots + \rho^{k-1}\bar{\gamma}) \\ \|X_k - X\| &\leq \rho^k \|X_0 - X\| + (1+\rho)\|C'^{-1}\|\bar{\gamma}(1 + \rho + \ldots + \rho^{k-1}). \end{aligned}$$

By using

$$1 + \rho + \ldots + \rho^{k-1} = \sum_{i=0}^{k-1} \rho^i \leq \frac{1}{1-\rho},$$

we prove the claim that

$$\|X_k - X\| \leq \rho^k \|X_0 - x\| + \frac{1+\rho}{1-\rho}\|C^{-1}\|\bar{\gamma}.$$

□

Remark 3.3.1.

$$\|X_k - X\| \leq \rho^k \|X_0 - X\| + \frac{1+\rho}{1-\rho}\|C^{-1}\|\bar{\gamma}$$

states that the error $\|X_k - X\|$ with which X is approximated decreases geometrically by a factor of $\rho < 1$ until it gets bounded by $\|C^{-1}\|\bar{\gamma}$ or $O(\bar{\gamma})$. This fact will also be shown in the error analysis in the next section.

3.4 Error analysis

In the last part of this thesis, we provide the backward and forward error analysis.

Let us consider the linear system $CW = U$ where C is an $n \times n$ well conditioned nonsingular matrix and U is an $n \times r$ matrix (see the previous section).

We compute the matrices $U_0 = U$ and $W_0 = C^{-1}U_0 = X_0$ by applying Gaussian Elimination with Partial Pivoting (GEPP). We apply Algorithm 3.3.1, which we rewrite as follow.

(1) Solve $CW_k = U_k$ using the GEPP
(2) $U_k = U_{k-1} - CW_{k-1}$
(3) $X_k = W_0 + \cdots + W_k$, $for\, k = 1, 2, ...$

The residual U_k in (2) and the new approximate solution X_k in (3) are computed using double precision arithmetic $\bar{u}$. The matrix W_k in (1) is computed using single precision arithmetic u.

The algorithm in (1) is backward stable, which means that there exists a matrix E_k such that

$$(C + E_k)W_k = U_k \text{ where } \|E_k\| \leq c(k)u\|C\|, \tag{3.4.1}$$

that is, W_k is an exact solution for the approximated problem, $c(k)$ is a linear function in k. Here is our error bounds in step (2) performed in double precision arithmetic,

$$U_k = fl(U_{k-1} - CW_{k-1}) = U_{k-1} - CW_{k-1} + \Delta E_k \tag{3.4.2}$$

where

$$\|\Delta E_k\| \leq c_1(k)\bar{u}(\|C\|\|W_{k-1}\| + \|U_{k-1}\|). \tag{3.4.3}$$

$$X_k = fl(W_{k-1} + W_k) = W_{k-1} + W_k. \tag{3.4.4}$$

We prove the following proposition.

Proposition 3.4.1. *Let $C \in \Delta^{n\times n}$ be nonsingular and consider the linear system $CW_k = U_k$. If $c(k)cond\,Cu < \rho < 1$, then the matrix $(C + E_k)$ is nonsingular and*

$$(C + E_k)^{-1} = (I + F_k)C^{-1} \text{ where } \|F_k\| \leq \frac{c(k)cond\,Cu}{1 - c(k)cond\,Cu} \tag{3.4.5}$$

or equivalently.

$$(C + E_k)^{-1} = C^{-1}(I + F_k).$$

Proof. Corollary 1.2.2 states that if $\|A^{-1}E\| < 1$ then $A - E$ is nonsingular. Since $(C + E_k)$ is nonsingular, it remains to prove that $\|C^{-1}E_k\| < 1$, which is the case because $\|C^{-1}E_k\| \leq \|C^{-1}\|\|E_k\|$. Therefore

$$\|C^{-1}E_k\| \leq \|C^{-1}\|\ c(k)\ u\|C\|$$

due to (3.4.1). Consequently,

$$\begin{aligned} \|C^{-1}E_k\| &\leq c(k)\|C^{-1}\|\|C\|u, \\ \|C^{-1}E_k\| &\leq c(k)cond\,Cu,\ cond\,C = \|C^{-1}\|\|C\|, \end{aligned}$$

$$\|C^{-1}E_k\| \leq c(k)cond\,Cu < \rho < 1. \tag{3.4.6}$$

Let us now prove that $(C + E_k)^{-1} = (I + F_k)C^{-1}$ (or $(C + E_k)^{-1} = C^{-1}(I + F_k)$) where

$$\|F_k\| \leq \frac{c(k)cond\,Cu}{1 - c(k)\ cond\,Cu}.$$

First, we observe that

$$\begin{aligned}(C+E_k)^{-1} &= \Big(C(I+C^{-1}E_k)\Big)^{-1} \\ &= (I+C^{-1}E_k)^{-1}C^{-1}.\end{aligned}$$

Furthermore

$$\begin{aligned}(C+E_k)^{-1}-C^{-1} &= \Big(I-C^{-1}(C+E_k)\Big)(C+E_k)^{-1} \\ &= -C^{-1}E_k(C+E_k)^{-1}.\end{aligned}$$

Consequently,

$$\begin{aligned}(C+E_k)^{-1} &= C^{-1}-C^{-1}E_k(C+E_k)^{-1} \\ &= C^{-1}\Big(I-E_k(C+E_k)^{-1}\Big) \\ &= C^{-1}(I+F_k)\,\text{where}\,F_k=-E_k(C+E_k)^{-1}.\end{aligned}$$

Taking the norm on both sides we obtain

$$\|F_k\| = \|-E_k(C+E_k)^{-1}\| \leq \|E_k\|\ \|(C+E_k)^{-1}\|.$$

From Corollary 1.2.2, we deduce

$$(C+E_k)^{-1} = \frac{\|C^{-1}\|}{1-\|C^{-1}E_k\|}.$$

Therefore,

$$\begin{aligned} \|F_k\| &\leq \|E_k\| \frac{\|C^{-1}\|}{1 - \|C^{-1}E_k\|}, \\ \|F_k\| &\leq \frac{\|E_k\|\|C^{-1}\|}{1 - \|C^{-1}E_k\|}. \end{aligned}$$

Now (3.4.1) and (3.4.6) together imply that

$$\begin{aligned} \|F_k\| &\leq \frac{c(k)u\|C\|\|C^{-1}\|}{1 - c(k)cond\,Cu}, \\ \|F_k\| &\leq \frac{c(k)cond\,Cu}{1 - c(k)cond\,Cu}. \end{aligned}$$

The same argument is used to deduce that $(C + E_k)^{-1} = (I + F_k)C^{-1}$.

$$\begin{aligned} (C + E_k)^{-1} - C^{-1} &= (C + E_k)^{-1}\Big(I - (C + E_k)C^{-1}\Big) \\ &= (C + E_k)^{-1}(C - I - E_kC^{-1}), \\ (C + E_k)^{-1} - C^{-1} &= (C + E_k)^{-1}(-E_kC^{-1}). \text{ Hence} \\ (C + E_k)^{-1} &= -(C + E_k)^{-1}E_kC^{-1} + C^{-1} \\ &= \Big(-(C + E_k)^{-1}E_k + I\Big)C^{-1} \\ &= \Big(I - (C + E_k)^{-1}E_k\Big)C^{-1} \\ &= (I + F_k)C^{-1}, \end{aligned}$$

for $F_k = -(C + E_k)^{-1}E_k$. □

3.5 Forward error analysis

We obtain from Equation (3.4.1) that

$$W_k = (C + E_k)^{-1} U_k. \tag{3.5.1}$$

Therefore, $X - X_k = X - W_{k-1} - W_k$. We obtain

$X - X_k = X - W_{k-1} - (C + E_k)^{-1} U_k$ by using (3.4.1),

$X - X_k = X - W_{k-1} - (C + E_k)^{-1}(U_{k-1} - CW_{k-1} + \Delta E_k)$ by using (3.4.2),

$X - X_k = X - W_{k-1} - (I + F_k)C^{-1}(U_{k-1} - CW_{k-1} + \Delta E_k)$ by using (3.4.5) in Proposition 3.4.1,

$X - X_k = X - W_{k-1} - (I + F_k)(C^{-1}U_{k-1} - W_{k-1} + C^{-1}\Delta E_k)$,

$X - X_k = X - W_{k-1} - (I + F_k)(C^{-1}U_k + C^{-1}\Delta E_k)$ by using (3.4.1).

Therefore,

$$X_k = W_{k-1} + W_k,$$

$$X_k = W_{k-1} + C^{-1}U_k,$$

$$X_k - W_{k-1} = C^{-1}U_k,$$

so that $X - X_k = X - W_{k-1} - (I + F_k)(X_k - W_{k-1} + C^{-1}\Delta E_k)$,

$X - X_k = X - W_{k-1} - (X_k - W_{k-1}) - F_k(X_k - W_{k-1}) - (I + F_k)C^{-1}\Delta E_k$. Consequently

$X - X_k = X - W_{k-1} + X_k + W_{k-1} - F_k(X_k - W_{k-1}) - (I + F_k)C^{-1}\Delta E_k$,

$X - X_k = -(X - X_k) - F_k(X_k - W_{k-1}) - (I + F_k)C^{-1}\Delta E_k$, and so

$2(X - X_k) = -F_k(X_k - W_{k-1}) - (I + F_k)C^{-1}\Delta E_k$.

Without loss of generality we can assume that

$X - X_k = -F_k(X_k - W_{k-1}) - (I + F_k)C^{-1}\Delta E_k.$

Recall that $\|F_k\| < 1$, take the norm on both sides, and obtain

$$\|X - X_k\| \leq \|F_k\|\|X_k - W_{k-1}\| + 2\|C^{-1}\|\|\Delta E_k\|.$$

Recalling (3.4.3) and (3.4.4), we deduce that

$$\|X - X_k\| \leq \|F_k\|\|X_k - W_{k-1}\| + 2\|C^{-1}\|c_1(k)\bar{u}\big(\|C\|\|W_{k-1}\| + \|U_{k-1}\|\big).$$

Recall the following inequalities,

(3.4.5) $\|F_k\| \leq \frac{c(k)cond\,Cu}{1-c(k)cond\,Cu} < 1$

(3.4.4) $X_k = W_{k-1} + W_k$, so $W_k = X_k - W_{k-1}$

$W_k = X - X_{k-1} + W_k - X + X_{k-1}$

$\|W_k\| \leq \|X - X_{k-1}\| + \|X\|$

so $\|X_k - W_{k-1}\| \leq \|W_k\| \leq \|X - X_{k-1}\| + \|X\|$

(3.4.3) $\|\Delta E_k\| \leq c_1(k)\bar{u}(\|C\|\|W_{k-1}\| + \|U_{k-1}\|).$

We also have

$$\|W_{k-1}\| \leq \|X - X_{k-1}\| + \|X\|$$

$$\|U_{k-1}\| \leq \|C\|\|W_k\|$$

$$\|U_{k-1}\| \leq \|C\|\|X - X_{k-1}\| + \|C\|\|X\|.$$

$$
\begin{aligned}
\|X - X_k\| &\leq \frac{c(k) cond\, Cu}{1 - c(k) cond\, Cu}(\|X - X_{k-1}\|) \\
&\quad + 2\ \|C^{-1}\| c_1(k)\bar{u}(\|C\|\|X - X_{k-1}\|) \\
&+ \|C\|\|X\| + \|C\|\|X - X_{k-1}\| + \|C\|\|X\| \\
\|X - X_k\| &\leq \left[\frac{c(k)\, u}{1 - c(k) cond\, Cu} + 2\|C^{-1}\|\|C\| c_1(k)\bar{u} \cdot 2\right] \|X - X_{k-1}\| \\
&\quad + 4\|C^{-1}\|\|C\| c_1(k)\bar{u}\|X\| \\
\|X - X_k\| &\leq \left[\frac{c(k) cond\, Cu}{1 - c(k) cond\, Cu} + 4 cond\, C\ c_1(k)\bar{u}\right] \|X - X_{k-1}\| \\
&\quad + 4 cond C\ c_1(k)\bar{u}\|X\| \\
\|X - X_k\| &\leq \left[\frac{c(k)}{1 - c(k) cond\, Cu} + 4c_1(k)\right] cond\, Cu\|X - X_{k-1}\| \\
&\quad + 4 cond\, C\ c_1(k)\bar{u}\|X\| \\
\|X - X_k\| &\leq \left[\frac{c(k)}{1 - c(k) cond\, Cu} + 4c_1(k)\right] cond\, Cu\|X - X_{k-1}\| \\
&\quad + 4 cond\, C\ c_1(k)\bar{u}\|X\| \\
\|X - X_k\| &\leq \alpha_1\|X - X_{k-1}\| + \alpha_2\|X\|
\end{aligned}
$$

where

$$
\begin{aligned}
\alpha_1 &= \left(\frac{c(k)}{1 - c(k) cond\, Cu} + 4c_1(k)\right) cond\, Cu \text{ and} \\
\alpha_2 &= 4 cond\, C\ c_1(k)\bar{u}.
\end{aligned}
$$

Assuming $|\alpha_1| < 1$ and $\frac{b_2(k) cond\, C\bar{u}}{1 - b_1(k) cond\, Cu} < 1$ we deduce that

$$\|X - X_{k-1}\| \leq \alpha_1\|X - X_{k-2}\| + \alpha_2\|X\|.$$

Therefore,

$$\|X - X_k\| \leq \alpha_1^2 \|X - X_{k-2}\| + \alpha_1 \alpha_2 |X\| + \alpha_2 \|X\|.$$

We also recall that

$$\|X - X_{k-2}\| \leq \alpha_1 \|X - X_{k-3}\| + \alpha_2 \|X\|,$$

so that

$$\|X - X_k\| \leq \alpha_1^3 \|X - X_{k-3}\| + \alpha_1^2 \alpha_2 \|X\| + \alpha_2 \|X\|,$$

and

$$\|X - X_k\| \leq \alpha_1^4 \|X - X_{k-4}\| + \alpha_1^3 \alpha_2 \|X\| + \alpha_1 \alpha_2 \|X\| + \alpha_2 \|X\|.$$

$$\ldots$$

$$\begin{aligned}
\|X - X_k\| &\leq \alpha_1^{k-1} \|X - X_1\| + (\alpha_1^{k-2} + \alpha_1^{k-3} + \ldots + 1) \alpha_2 \|X\| \\
\|X - X_k\| &\leq \alpha_1^{k-1} \|X - X_0\| + (1 + \alpha_1 + \ldots + \alpha_1^{k-3} + \alpha_1^{k-2}) \alpha_2 \|X\| \\
\|X - X_k\| &\leq \alpha_1^{k-1} \|X - X_0\| + \frac{1 - \alpha_1^{k-1}}{1 - \alpha_1} \alpha_2 \|X\|.
\end{aligned}$$

Therefore,

$$\begin{aligned}
\lim_{k\to\infty} &\leq \frac{\alpha_2}{1-\alpha_1}\|X\| \\
\lim_{k\to\infty} \|X - X_k\| &\leq \frac{4cond\,C\ c_1(k)\bar{u}}{1-\alpha_1}\|X\| \\
\lim_{k\to\infty} \frac{\|X - X_k\|}{\|X\|} &\leq \frac{4cond\,C\ c_1(k)\bar{u}}{1-\alpha_1} \leq \frac{u}{1-\alpha_1} \\
\lim_{k\to\infty} \frac{\|X - X_k\|}{\|X\|} &\leq \frac{u}{1-\alpha_1} = \beta(k)u, \quad \beta(k) = \frac{1}{1-\alpha_1} \text{ is a constant} \\
\lim_{k\to\infty} \frac{\|X - X_k\|}{\|X\|} &\leq O(u).
\end{aligned}$$

These results are in line with Higham's results discussed in Section 2.3.1 in Chapter 2. We deduce normwise errors, which are superior to the componentwise errors discussed by Higham in [6]. (cf. [6], page 234, Theorem 11.1).

3.6 Backward error analysis

We recall from (3.4.1) that $(C + E_k)W_k = U_k$, and

(3.4.4) $X_k = W_{k-1} + W_k$. We have

$X - X_k = X - W_{k-1} + W_k$,

$X - X_k = X - W_{k-1} - (C + E_k)^{-1}U_k$. By combining these equations with (3.4.4) deduce that

$X - X_k = X - W_{k-1} - C^{-1}(I + F_k)(U_k - CW_{k-1} + \Delta E_k)$.

By pre-multiplying by C, obtain

$CX - CX_k = CX - CW_{k-1} - (I + F_k)(U_k - CW_{k-1} + \Delta E_k)$.

$$CX - CX_k = CX - CW_{k-1} - F_k(U_{k-1} - CW_{k-1}) - (I + F_k)\Delta E_k - U_{k-1} + CW_{k-1}.$$

After cancelation this gives us

$$-CX_k = -U_{k-1} - F_k(U_{k-1} - CW_{k-1}) - (I + F_k)\Delta E_k,$$

$$U_{k-1} - Cx_k = -F_k(U_{k-1} - CW_{k-1}) - (I + F_k)\Delta E_k.$$

By combining (3.4.3) with the latter equality, we deduce that

$$U_{k-1} - C(W_{k-1} + W_k) = -F_k(U_{k-1} - CW_{k-1}) - (I + F_k)\Delta E_k,$$

$$U_{k-1} - CW_{k-1} - CW_k = -F_k(U_{k-1} - CW_{k-1}) - (I + F_k)\Delta E_k.$$

$U_{k-1} - CW_{k-1} = U_k$ from (3.4.2), so that

$$U_k - CW_k = -F_k(U_{k-1} - CW_{k-1}) - (I - F_k)\Delta E_k.$$

Taking the norm on both side and using $\|F_k\| < 1$ obtain

$$\begin{aligned}
\|U_k - CW_k\| &\leq \|F_k\|\|U_{k-1} - CW_{k-1}\| + 2\|\Delta E_k\| \\
\|U_k - CW_k\| &\leq \|F_k\|\|U_{k-1} - CW_{k-1}\| + 2c_1(k)\bar{u}(\|C\|\|W_{k-1}\| + \|U_{k-1}\|) \\
\|U_k - CW_k\| &\leq \|F_k\|\|U_{k-1} - CW_{k-1}\| + 2c_1(k)\bar{u}\|C\|\|W_{k-1}\| + 2c_1(k)\bar{u}\|U_{k-1}\| \\
\|U_k - CW_k\| &\leq \|F_k\|\|U_{k-1} - CW_{k-1}\| + 2c_1(k)\bar{u}\|U_{k-1}\| + 2c_1(k)\bar{u}\|C\|\|W_{k-1}\|.
\end{aligned}$$

Let us write $U_{k-1} = U_{k-1} - CW_{k-1} + CW_{k-1}$. Taking the norm on both side we obtain

$$\begin{aligned}
\|U_{k-1}\| &\leq \|U_{k-1} - CW_{k-1}\| + \|CW_{k-1}\|, \\
\|U_{k-1}\| &\leq \|U_{k-1} - CW_{k-1}\| + \|C\|\|W_{k-1}\|,
\end{aligned}$$

so that

$$\begin{aligned}
\|U_k - CW_k\| &\leq \|F_k\|\|U_{k-1} - CW_{k-1}\| + 2c_1(k)\bar{u}(\|U_{k-1} - CW_{k-1}\| + \|C\|\|W_{k-1}\|) \\
&\quad + 2c_1(k)\bar{u}\|C\|\|W_{k-1}\| \\
\|U_k - CW_k\| &\leq \|F_k\|\|U_{k-1} - CW_{k-1}\| + 2c_1(k)\bar{u}\|U_{k-1} - CW_{k-1}\| \\
&\quad + 4c_1(k)\bar{u}\|C\|\|W_{k-1}\| \\
\|U_k - CW_k\| &\leq (\|F_k\| + 2c_1(k)\bar{u}\|U_{k-1} - CW_{k-1}\| \\
&\quad + 4c_1(k)\bar{u}\|C\|\|W_{k-1}\|.
\end{aligned}$$

Dividing the latter inequality by $\|C\|\|W_{k-1}\|$ and using the inequality $\|W_{k-1}\| \leq \|W_k\|$, we deduce that

$$\begin{aligned}
\frac{\|U_k - CW_k\|}{\|C\|\|W_k\|} &\leq (\|F_k\| + 2c_1(k)\bar{u})\frac{\|U_{k-1} - CW_{k-1}\|}{\|C\|\|W_{k-1}\|} + 4c_1(k)\bar{u} \\
\frac{\|U_k - CW_k\|}{\|C\|\|W_k\|} &\leq \left(\frac{c_1(k)cond\,Cu}{1 - c_1(k)cond\,Cu} + 2c_1(k)\bar{u}\right)\frac{\|U_{k-1} - CW_{k-1}\|}{\|C\|\|W_{k-1}\|} + 4c_1(k)\bar{u} \\
\frac{\|U_k - CW_k\|}{\|C\|\|W_k\|} &\leq \alpha_1'\frac{\|U_{k-1} - CW_{k-1}\|}{\|C\|\|W_{k-1}\|} + \alpha_2',
\end{aligned}$$

where

$$\alpha_1' = \frac{c_1(k)cond\,Cu}{1 - c_1(k)cond\,Cu} + 2c_1(k)\bar{u} = c_1'(k)cond\,Cu$$

and

$$\alpha_2' = 4c_1(k)\bar{u}.$$

Therefore,

$$\frac{\|U_k - CW_k\|}{\|C\|\|W_k\|} \leq \alpha_1'\frac{\|U_{k-1} - CW_{k-1}\|}{\|C\|\|W_{k-1}\|} + \alpha_2'$$

and

$$\frac{\|U_{k-1} - CW_{k-1}\|}{\|C\|\|W_{k-1}\|} \leq \alpha'_1 \frac{\|U_{k-2} - CW_{k-2}\|}{\|C\|\|W_{k-2}\|} + \alpha'_2.$$

By combining the last two inequalities we obtain

$$\frac{\|U_k - CW_k\|}{\|C\|\|W_k\|} \leq \alpha'^2_1 \frac{\|U_{k-2} - CW_{k-2}\|}{\|C\|\|W_{k-2}\|} + \alpha'_1\alpha'_2 + \alpha'_2$$

$$\ldots$$

$$\begin{aligned} \frac{\|U_k - CW_k\|}{\|C\|\|W_k\|} &\leq \alpha'^k_1 + \alpha'_2(\alpha'^{k-1}_1 + \alpha'^{k-2}_1 + \ldots + 1), \\ \frac{\|U_k - CW_k\|}{\|C\|\|W_k\|} &\leq \alpha'^k_1 + \alpha'_2 \frac{1 - \alpha'^k_1}{1 - \alpha'_1}. \end{aligned}$$

At the limit we obtain

$$\begin{aligned} \lim_{k\to\infty} \frac{\|U_k - CW_k\|}{\|C\|\|W_k\|} &\leq \frac{\alpha'_2}{1 - \alpha'_1} = \frac{4c_1(k)\bar{u}}{1 - c'_1(k)cond\,Cu} \\ \lim_{k\to\infty} \frac{\|U_k - CW_k\|}{\|C\|\|W_k\|} &\leq \frac{4c_1(k)}{1 - c'_1(k)cond\,Cu}\bar{u} \\ \lim_{k\to\infty} \frac{\|U_k - CW_k\|}{\|C\|\|W_k\|} &\leq O(\bar{u}). \end{aligned}$$

These inequalities confirm the result in Remark 2.0.13 on page 30 in Chapter 2. The approximate solution $W_0 + W_1 + W_2 + \ldots + W_k \to W = C^{-1}U$ can be regarded as very satisfactory.

Remark 3.6.1. Theorem 3.2.1 and Theorem 3.3.3 along with the error analysis imply that the sum $W_0 + W_1 + W_2 + \ldots + W_k$ approximates the matrix $W = C^{-1}U$, solution of the linear equation $CW = U$ with an error no larger than the uncertainty of the

data A and b.

We have $W = \sum_{i=1}^{k} W_i$. We successively compute the matrices $F_i \leftarrow -V^H W_i$, and $S \leftarrow I_r + \sum_{i=1}^{k} F_i$ (cf. Remark 3.2.1).

Finally, we compute the vectors $C^{-1}b$, $C^{-1}US^{-1}V^H C^{-1}b$ and the improved solution $x = A^{-1}b = C^{-1}b + C^{-1}US^{-1}V^H C^{-1}b$ of the ill conditioned system $Ax = b$.

3.7 Stopping criteria

The extended iterative refinement/improvement process should stop where

(1) $\frac{\|W_k - W_{k-1}\|}{\|W_k\|} \leq cu$, for $c \simeq \sqrt{k}$.

(2) $\|U_k\| < c\|C\|\|W_{k-1}\|\bar{u}$, for $c \simeq \sqrt{k}$ (this is based on $\|\Delta E_k\| \leq c_1(k)\bar{u}(\|C\|\|W_{k-1}\| + \|U_{k-1}\|)$.

(3) $\frac{\|U_k - CW_k\|}{\|C\|\|W_k\|} \leq c\bar{u}$ for $c \simeq \sqrt{k}$.

3.8 Final comment

Further research in the direction of our study is likely to reveal additional reduction of the upper bounds on the error norms proved in this thesis and thus would lead us to a smaller error norm bound $\|X_k - X\|$ *with which the sum* $X_k = W_0 + W_1 + \cdots + W_k \ldots$ *is approximated.*

Bibliography

[1] V. Y. Pan, D. Ivolgin, B. Murphy, R. E. Rosholt, Y Tang, X. Yan, Additive Preconditioning and Aggregation in Matrix Computions, Technical Report TR 2006 007, CUNY Ph.D. *Program in Computer Science, Graduate Center, City University of New York*, April 2007.

[2] V. Y. Pan, *Structured Matrices and Polynomials: Unified Superfast Algorithms*, Birkhäuser/Springer, Boston/New York, 2001.

[3] V. Y. Pan, Y. Yu, Certification of Numerical Computation of the Sign of the Determinant of a Matrix, *Algorithmica*, 30, 708-724, 2001.

[4] G. W. Stewart, *Matrix Algorithms, Vol. I: Basic Decompositions*, SIAM, Philadelphia, 1998.

[5] D. E. Knuth, *The Art of Computer Programming: Volume 2, Seminumerical Algorithms*, Addison-Wesley, Reading, Massachusetts, 1969 (first edition), 1981 (second edition), 1998 (third edition).

[6] N. J. Higham, *Accuracy and Stability in Numerical Analysis*, SIAM, Philadelphia, 2002 (second edition).

[7] T. Ogita, S. M. Rump, S. Oishi, Accurate Sum and Dot Product, SIAM *Journal on Scientific Computing*, 26, 6, 1955-1988, 2005.

[8] S. M. Rump, T. Ogita, S. Oishi, Accurate Floating-Point Summation, Tech. Report 05.12, *Faculty for Information and Communication Sciences, Hamburg University of Technology*, November 2005.

[9] I. Babuška, Numerical Stability in Mathematical Analysis, *Information Processing*, 68 (Proc. of IFIP Congress), 11-23, North-Holland, Amsterdam, 1969.

[10] T. J. Dekker, A Floating-Point Technique for Extending the Available Precision, *Numerische Math.*, 18, 224-242, 1971.

[11] N. J. Higham. The Accuracy of Floating Point Summation, SIAM J. *on Scientific Computing*, 14, 783-799, 1993.

[12] J. Demmel, Y. Hida, W. Kahan, X. S. Li, Soni Mukherjee, E. J. Riedy, Error Bound from Extra Precise Iterative Refinement, *Computer Science Division, Technical Report* UCB//CSD-04-1344, University of California, Berkeley, February 2005.

[13] W. L. Miranker, V. Y. Pan, Methods of Aggregations, *Linear Algebra and Its Application*, 29, 231-257, 1980.

[14] G. H. Golub, C. F. Van Loan, *Matrix Computations*, 3rd edition, The Johns Hopkins University Press, Baltimore, Maryland, 1996.

[15] G. W. Stewart, *Matrix Algoritms, Vol II: Eigensystems*, SIAM, Philadelphia, 1998.

[16] R. A. Demillo, R. J. Lipton, A Probabilistic Remark on Algebraic Program Testing, *Information Processing Letters*, 7, 4, 193-195, 1978.