Chapter 14

Special Topics

Background Material Needed

- Vector and Matrix Norms (Section 2.5)
- Rounding Errors in Basic Floating Point Operations (Section 3.3–3.7)
- Forward Elimination and Back Substitution Process (Algorithms 4.3 and 4.4)
- Gaussian Elimination with and without Pivoting (Sections 5.2.2 and 5.2.4)
- Householder QR Factorization Method (Section 7.2.2)

14.1 Introduction

In this final Chapter, we shall briefly discuss the following advanced (but important) topics:

- QR factorization with column pivoting;
- modifying QR factorization;
- a taste of round-off error analysis.

14.2 QR Factorization with Column Pivoting

If an $m \times n$ ($m \geq n$) matrix A has rank $r < n$, then the matrix R is singular. In this case the QR factorization cannot be employed to produce an orthonormal basis of $R(A)$.

To see this, just consider the following simple 2×2 example from Björck (1996, p. 21):

$$A = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} c & -s \\ s & c \end{pmatrix} \begin{pmatrix} 0 & s \\ 0 & c \end{pmatrix} = QR. \tag{14.1}$$

If c and s are chosen such that $c^2 + s^2 = 1$, $\text{rank}(A) = 1 < 2$, and the columns of Q do not form an orthonormal basis of $R(A)$, nor is one formed for its complement.

Fortunately, however, the process of QR factorization (for example, the Householder method) can be modified to yield an orthonormal basis. The idea here is to generate a permutation matrix P such that

$$AP = QR,$$

where

$$R = \begin{pmatrix} R_{11} & R_{12} \\ 0 & 0 \end{pmatrix}.$$

Here R_{11} is $r \times r$ upper triangular, r is the rank of A, and Q is orthogonal. The first r columns of Q will then form an orthonormal basis of $R(A)$. The following theorem guarantees the existence of such a factorization.

Theorem 14.1 (QR column pivoting theorem). *Let A be an $m \times n$ matrix with* $\operatorname{rank}(A) = r \le \min(m, n)$. *Then there exist an $n \times n$ permutation matrix P and an $m \times m$ orthogonal matrix Q such that*

$$Q^T AP = \begin{pmatrix} R_{11} & R_{12} \\ 0 & 0 \end{pmatrix},$$

where R_{11} is an $r \times r$ upper triangular matrix with nonzero diagonal entries.

Proof. Since $\operatorname{rank}(A) = r$, there exists a permutation matrix P such that

$$AP = (A_1, A_2),$$

where A_1 is $= m \times r$ and has linearly independent columns. Consider now the QR factorization of A_1, $Q^T A_1 = \left(\begin{smallmatrix} R_{11} \\ 0 \end{smallmatrix}\right)$, where by the uniqueness theorem (Theorem 7.14), Q and R_{11} are uniquely determined and R_{11} has positive diagonal entries. Then

$$Q^T AP = (Q^T A_1, Q^T A_2) = \begin{pmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{pmatrix}.$$

Since $\operatorname{rank}(Q^T AP) = \operatorname{rank}(A) = r$ and $\operatorname{rank}(R_{11}) = r$, we must have $R_{22} = 0$. □

Remark. Note that in practice, R_{22} can even be not small. See our discussion on $RRQR$ decomposition later.

Column Pivoting QR Factorization

The above factorization in known as **QR factorization with column pivoting**. The factorization in general is not unique. A standard algorithm can be briefly described as follows. The details can be found in Golub and Van Loan (1996) and Björck (1996).

Step 1. Find the column of A having the **maximum norm**. Permute now the columns of A so that the column of maximum norm becomes the first column.

This is equivalent to creating a permutation matrix P_1 such that the matrix AP_1 has the first column having the maximum norm. Create now a Householder matrix H_1 so that

$$A_1 = H_1 AP_1$$

has zeros in the first column below the (1,1) entry.

$$A_1 = \begin{pmatrix} \times & \times & \times & \cdots & \times \\ 0 & & & & \\ 0 & & & & \\ \vdots & & \hat{A}_1 & & \\ 0 & & & & \end{pmatrix}.$$

Step 2. Find the column with the maximum norm of the submatrix $\hat{A}_1$ obtained from A_1 by deleting the first row and the first column (as shown above). Permute the columns of this submatrix so that the column of maximum norm becomes the first column. This is equivalent to constructing a permutation matrix $\hat{P}_2$ such that the first column of $\hat{A}_1\hat{P}_2$ has the maximum norm. Now, construct a Householder matrix $\hat{H}_2$ such that the first column of $\hat{H}_2\hat{A}_1\hat{P}_2$ has zeros below its (1,1) entry. Form now P_2 and H_2 in the usual way; that is

$$P_2 = \text{ diag } (1, \hat{P}_2) \quad \text{and} \quad H_2 = \text{ diag } (1, \hat{H}_2).$$

Then

$$A_2 = H_2 A_1 P_2 = H_2 H_1 A P_1 P_2$$

has zeros in the second column of A_2 below the (2, 2) entry. The matrix A_2 has the following structure:

$$A_2 = \begin{pmatrix} \times & \times & \times & \cdots & \times \\ 0 & \times & \times & \cdots & \times \\ 0 & 0 & \times & \cdots & \times \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \times & \cdots & \times \end{pmatrix} = \begin{pmatrix} \times & \times & \times & \cdots & \times \\ 0 & \times & & & \\ 0 & 0 & & & \\ \vdots & \vdots & & \hat{A}_2 & \\ 0 & 0 & & & \end{pmatrix}.$$

The process is now continued with $\hat{A}_2$.

- The kth step can now easily be written down.
- The process is continued until the entries below the diagonal of the current matrix all become zero.

Suppose r steps are needed. Then at the end of the rth step, we have

$$\begin{aligned} A \equiv A_r &= H_r \cdots H_1 A P_1 \cdots P_r \\ &= Q^T A P = R = \begin{pmatrix} R_{11} & R_{12} \\ 0 & 0 \end{pmatrix}. \end{aligned}$$

Flop-count and storage consideration. The above method requires $4mnr - 2r^2 (m+n) + \frac{4r^3}{3}$ flops. The matrix Q, as in the Householder factorization, is stored in factored form in the subdiagonal part of A. The triangular part of A can be overwritten by the upper triangular part of R.

Example 14.2.

$$A = \begin{pmatrix} 0 & 0 & 0 \\ 0.5 & 0.5 & 1 \\ 1 & 0.5 & 1 \end{pmatrix} = (a_1, a_2, a_3).$$

Step 1. The column a_3 has the largest norm. Thus,

$$P_1 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix},$$

$$H_1 = \begin{pmatrix} 0 & -0.7071 & -0.7071 \\ -0.7071 & 0.5000 & 0.5000 \\ -0.7071 & -0.5000 & 0.50000 \end{pmatrix},$$

$$A_1 = H_1 A P_1 = \begin{pmatrix} -1.4142 & -0.7071 & -1.0607 \\ 0 & 0 & -0.2500 \\ 0 & 0 & 0.2500 \end{pmatrix}.$$

Step 2.

$$\hat{A}_1 = \begin{pmatrix} 0 & -0.2500 \\ 0 & 0.2500 \end{pmatrix},$$

$$\hat{P}_2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \hat{H}_2 = \begin{pmatrix} -0.7071 & 0.7071 \\ 0.7071 & 0.7071 \end{pmatrix},$$

$$P_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \quad H_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -0.7071 & 0.7071 \\ 0 & 0.7071 & 0.7071 \end{pmatrix},$$

$$A_2 = H_2 A_1 P_2 = \begin{pmatrix} -1.4142 & -1.0607 & -0.7071 \\ 0 & 0.3536 & 0 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} R_{11} & R_{12} \\ 0 & 0 \end{pmatrix}.$$

Note that $R \equiv A_2$, and $Q = H_1 H_2$, $P = P_1 P_2$. ∎

MATLAB Note: MATLAB command $[Q, R, E] = QR(A)$ produces a permutation matrix E such that $A * E = Q \times R$. E is chosen so that ABS(DIAG(R)) is decreasing.

Complete Orthogonal Factorization

It is easy to see that the submatrix (R_{11}, R_{12}) can further be reduced by using orthogonal transformations, yielding $\begin{pmatrix} T & 0 \\ 0 & 0 \end{pmatrix}$. Thus we have the following theorem.

Theorem 14.3 (complete orthogonalization theorem). *Given $A_{m\times n}$ with* rank$(A) = r$, *there exist orthogonal matrices $Q_{m\times m}$ and $W_{n\times n}$ such that*

$$Q^T A W = \begin{pmatrix} T & 0 \\ 0 & 0 \end{pmatrix},$$

where T is an $r \times r$ upper triangular matrix with positive diagonal entries.

Proof. The proof is left as an exercise (Exercise 14.2). □

The above decomposition of A is called the **complete orthogonal decomposition**.

A Rank-Revealing QR Factorization

The process of **QR factorization with column pivoting** was developed by Businger and Golub (1965). In exact arithmetic, it reveals the rank of matrix A which is the order of the nonsingular upper triangular matrix R_{11}. However, in the presence of rounding errors, we will actually have

$$R = \begin{pmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{pmatrix},$$

and if R_{22} is "small" in some measure (say, $\|R_{22}\|$ is of $O(\mu)$, where μ is the machine precision), then the reduction will be terminated. Thus, from the above discussion, we note that, given an $m \times n$ matrix A $(m \geq n)$, if there exists a permutation matrix P such that

$$Q^T AP = R = \begin{pmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{pmatrix},$$

where R_{11} is $r \times r$, and R_{22} is small in some measure, then we will say that A has numerical rank r. (For more on numerical rank, see Chapter 7 (Section 7.8.9).)

Unfortunately, the converse is not true.

A celebrated counterexample due to Kahan (1966) shows that a matrix can be nearly rank-deficient without having $\|R_{22}\|$ small at all.

Consider

$$A = \operatorname{diag}(1, s, \ldots, s^{n-1}) \begin{pmatrix} 1 & -c & -c & \cdots & -c \\ 0 & 1 & -c & \cdots & -c \\ \vdots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & -c \\ 0 & \cdots & \cdots & 0 & 1 \end{pmatrix} = R$$

with $c^2 + s^2 = 1,\ c, s > 0$. For $n = 100,\ c = 0.2$, it can be shown that this matrix is nearly singular (the smallest singular value is $O(10^{-8})$). On the other hand, $r_{nn} = s^{n-1} = 0.133$, which is not small, so R cannot be nearly singular.

The question whether at any stage R_{22} becomes really small for any matrix has been investigated by Chan (1987), Hong and Pan (1992), and others.

It can be shown that if $A \in R^{m\times n}$ $(m \geq n)$ and r is any integer $0 < r < n$, then there exists a permutation matrix Π such that QR factorization has the form

$$A\Pi = Q \begin{pmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{pmatrix},$$

where R_{11} is an $r \times r$ upper triangular matrix, with $\sigma_r(R_{11}) \geq \frac{1}{c}\sigma_r(A)$, $\|R_{22}\| \leq c\sigma_{r+1}(A)$, $c = \sqrt{r(n-r) + \min(r, n-r)}$. $\sigma_i(A)$ stands for the ith singular value of A.

A QR factorization of the above form is called a **rank-revealing QR (RRQR)** factorization. If $\sigma_{r+1} = 0$, then we have the decomposition in Theorem 14.1.

14.3 Modifying a QR Factorization

Suppose the QR factorization of an $m \times k$ matrix $A = (a_1, \ldots, a_k)$ $(m \geq k)$ has been obtained. A vector a_{k+1} is now appended to obtain a new matrix: $A' = (a_1, \ldots, a_k, a_{k+1})$.

It is natural to wonder how the QR factorization of A' can be obtained from the given QR factorization of A, without starting from scratch.

This is known as the *updating QR factorization*. *The downdating QR factorization* is similarly defined. The updating and downdating QR factorizations arise in a variety of practical applications, such as *signal and image processing*.

We present below a simple algorithm using Householder matrices to solve the updating problem.

ALGORITHM 14.1. Updating QR Factorization Using Householder Matrices.

Inputs: $A \in \mathbb{R}^{m\times k}$ $(m \geq k)$, an arbitrary column vector a_{k+1}, and Householder matrices H_1 through H_k such that

$$H_k H_{k-1} \cdots H_2 H_1 A = \begin{pmatrix} R \\ 0 \end{pmatrix}$$

Output: QR factorization of the augmented matrix $A' = (A, a_{k+1})$:

$$(Q')^T A' = R'.$$

Step 1. Compute $b_{k+1} = H_k \cdots H_1 a_{k+1}$.

Step 2. Compute a Householder matrix H_{k+1} so that $H_{k+1} b_{k+1} = r_{k+1}$ has zeros in entries $k+2, \ldots, m$.

Step 3. Form $R' = \left(\left(\begin{smallmatrix} R \\ 0 \end{smallmatrix}\right), r_{k+1}\right)$.

Step 4. Form $Q' = H_{k+1} \cdots H_1$.

Example 14.4.

$$a_2 = \begin{pmatrix} 1 \\ 4 \\ 5 \end{pmatrix}; \quad A = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix},$$

$$H_1 = Q^T = \begin{pmatrix} -0.2673 & -0.5345 & -0.8018 \\ -0.5345 & 0.7745 & -0.3382 \\ -0.8018 & -0.3382 & 0.4927 \end{pmatrix}, \quad R = \begin{pmatrix} -3.7417 \\ 0 \\ 0. \end{pmatrix}.$$

Step 1.

$$b_2 = H_1 a_2 = \begin{pmatrix} -6.4143 \\ 0.8227 \\ 0.3091 \end{pmatrix}.$$

Step 2.

$$H_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -0.9426 & -0.3339 \\ 0 & -0.3339 & 0.9426 \end{pmatrix}, \qquad r_2 = \begin{pmatrix} -6.4143 \\ -0.9258 \\ 0 \end{pmatrix}.$$

Step 3.

$$R' = (R, r_2) = \begin{pmatrix} -3.7417 & -6.4143 \\ 0 & -0.9258 \\ 0 & 0 \end{pmatrix},$$

$$Q' = H_2 H_1 = \begin{pmatrix} -0.2673 & -0.5345 & -0.8018 \\ 0.7715 & -0.6172 & 0.1543 \\ -0.5773 & -0.5774 & 0.5774 \end{pmatrix}.$$

Verification: $(Q')^T R' = \begin{pmatrix} 1 & 1 \\ 2 & 4 \\ 3 & 5 \end{pmatrix} = A'$. ∎

14.4 A Taste of Round-Off Error Analysis

Here we give the readers a taste of round-off error analysis in matrix computations by proving backward analyses of some standard matrix computations, such as solutions of *triangular systems, LU factorization using Gaussian elimination*, and *solution of a linear system*.

Let's recall that by *backward error analysis we mean an analysis that shows that the computed solution by the algorithm is an exact solution of a perturbed problem*. When the perturbed problem is close to the original problem, we say that the algorithm is backward stable.

Basic Laws of Floating Point Arithmetic

We first remind the reader of the basic laws of floating point arithmetic which will be used in what follows. These laws were obtained in Chapter 3. Let $|\delta| < \mu$, where μ is the machine precision. Then the following hold:

1. $\mathrm{fl}(x \pm y) = (x \pm y)(1 + \delta)$. (14.2)
2. $\mathrm{fl}(xy) = xy(1 + \delta)$. (14.3)
3. If $y \neq 0$, then $\mathrm{fl}(\frac{x}{y}) = (\frac{x}{y})(1 + \delta)$. (14.4)

Occasionally, we will also use

4. $\mathrm{fl}(x * y) = \frac{x * y}{1+\delta}$, (14.5)

where "*" denotes any of the arithmetic operations +, −, ×, /.

14.4.1 Backward Error Analysis for Forward Elimination and Back Substitution

Case 1. Lower Triangular System.

Consider solving the lower triangular system

$$Ly = b, \tag{14.6}$$

where

$$L = (l_{ij}), b = (b_1, \ldots, b_n)^T \tag{14.7}$$

and

$$y = (y_1, \ldots, y_n)^T,$$

using the **forward elimination** scheme.

We will use $\hat{s}$ to denote a computed quantity of s.

Step 1.

$$\hat{y}_1 = \mathrm{fl}\left(\frac{b_1}{l_{11}}\right) = \frac{b_1}{l_{11}(1+\delta_1)}, \quad |\delta_1| \le \mu \quad \text{(using 14.5)}.$$

This gives

$$l_{11}(1+\delta_1)\hat{y}_1 = b_1$$

or

$$\hat{l}_{11}\hat{y}_1 = b_1, \quad \text{where } \hat{l}_{11} = l_{11}(1+\delta_1).$$

That is, $\hat{y}_1$ is the exact solution of an equation whose coefficient is a number close to l_{11}.

Step 2.

$$\begin{aligned}\hat{y}_2 &= \mathrm{fl}\left(\frac{b_2 - l_{21}\hat{y}_1}{l_{22}}\right) = \mathrm{fl}\left(\frac{b_2 - \mathrm{fl}(l_{21}\hat{y}_1)}{l_{22}}\right)\\ &= \frac{(b_2 - l_{21}y_1(1+\delta_{11}))(1+\delta_{22})}{l_{22}(1+\delta_2)}\end{aligned} \tag{14.8}$$

(using both (14.3) and (14.5)), where $|\delta_{11}|$, $|\delta_{21}|$, and $|\delta_2|$ are all less than or equal to μ. Equation (14.8) can be rewritten as

$$l_{21}(1+\delta_{11})(1+\delta_{22})\hat{y}_1 + l_{22}(1+\delta_2)\hat{y}_2 = b_2(1+\delta_{22}) \tag{14.9}$$

That is,

$$l_{21}(1+\epsilon_{21})\hat{y}_1 + l_{22}(1+\epsilon_{22})\hat{y}_2 = b_2,$$

where

$$\epsilon_{21} = \delta_{11} \qquad \epsilon_{22} = \left(\frac{\delta_2 - \delta_{22}}{1+\delta_{22}}\right)$$

(neglecting $\delta_{11}\delta_{22}$, which is small). Thus, we can say that $\hat{y}_1$ and $\hat{y}_2$ satisfy

$$\hat{l}_{21}\hat{y}_1 + \hat{l}_{22}\hat{y}_2 = b_2, \text{ where } \hat{l}_{21} = l_{21}(1+\epsilon_{21}) \text{ and } \hat{l}_{22} = l_{22}(1+\epsilon_{22}). \tag{14.10}$$

Step *k*. The preceding can be easily generalized, and we can say that at the kth step, the unknowns y_1 through y_k satisfy

$$\hat{l}_{k1}\hat{y}_1 + \hat{l}_{k2}\hat{y}_2 + \cdots + \hat{l}_{kk}\hat{y}_k = b_k, \tag{14.11}$$

where $\hat{l}_{kj} = l_{kj}(1+\epsilon_{kj})$, $j = 1, \ldots, k$.

The process can be continued until $k = n$.

Thus, we see that the computed $\hat{y}_1$ through $\hat{y}_n$ satisfy the following perturbed triangular system:

$$\begin{aligned}\hat{l}_{11}\hat{y}_1 &= b_1,\\ \hat{l}_{21}\hat{y}_1 + \hat{l}_{22}\hat{y}_2 &= b_2,\\ &\vdots\\ \hat{l}_{n1}\hat{y}_1 + \hat{l}_{n2}\hat{y}_2 + \cdots + \hat{l}_{nn}\hat{y}_n &= b_n,\end{aligned}$$

where $\hat{l}_{kj} = l_{kj}(1+\epsilon_{kj}),\ k = 1,\dots,n,\ j = 1,\dots,k$. Note that $\epsilon_{11} = \delta_1$.

These equations can be written in matrix form,

$$\hat{L}\hat{y} = (L + \Delta L)\hat{y} = b, \tag{14.12}$$

where ΔL is a lower triangular matrix whose (i, j)th element $(\Delta L)_{ij} = l_{ij}\epsilon_{ij}$.

Knowing the bounds for ϵ_{ij}, the bounds for $(\Delta L)_{ij}$ can be easily computed. For example, if n is small enough so that $n\mu < \frac{1}{10}$, then $|\epsilon_{kj}| \leq 1.06(k - j + 2)\mu$ (see Chapter 3, Section 3.5). Then

$$|(\Delta L)_{ij}| \leq 1.06(i - j + 2)\mu|l_{ij}|. \tag{14.13}$$

The preceding discussions can be summarized in the following theorem.

Theorem 14.5. *The computed solution $\hat{y}$ to the $n \times n$ lower triangular system $Ly = b$, obtained by forward elimination, satisfies a perturbed triangular system:*

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

where the entries of ΔL are bounded by (14.13) *assuming that $n\mu < \frac{1}{10}$.*

Case 2. Upper Triangular System.

The round-off error analysis for solving an upper triangular system

$$Ux = c$$

using back substitution is similar to Case 1. In this case, we have the following theorem.

Theorem 14.6. *Let U be an $n \times n$ upper triangular matrix and let c be a vector. Then the computed solution $\hat{x}$ to the system*

$$Ux = c$$

using back substitution process satisfies

$$(U + \Delta U)\hat{x} = c, \tag{14.14}$$

where

$$|(\Delta U)_{ij}| \leq 1.06(i - j + 2)\mu|u_{ij}|, \tag{14.15}$$

assuming $n\mu < \frac{1}{10}$.

14.4.2 Backward Error Analysis for Triangularization by Gaussian Elimination

The treatment here follows very closely to the one given in Ortega (1990), and in Forsythe and Moler (1967).

Recall that the process of triangularization using Gaussian elimination consists of $(n-1)$ steps. At step k, matrix $A^{(k)}$ is constructed, which is triangular in the first k columns; that is,

$$A^{(k)} = \begin{pmatrix} a_{11}^{(k)} & \cdots & \cdot & \cdots & a_{ln}^{(k)} \\ & \ddots & & & \vdots \\ & & a_{kk}^{(k)} & \cdots & a_{kn}^{(k)} \\ & & \vdots & \ddots & \vdots \\ & & a_{nk}^{(k)} & \cdots & a_{nn}^{(k)} \end{pmatrix}. \tag{14.16}$$

The final matrix $A^{(n-1)}$ is triangular. We shall assume that the quantities $a_{ij}^{(k)}$ are the computed numbers.

Step 1. First, let's analyze the computations of the entries of $A^{(1)}$ from A in the first step. Let the computed multipliers be $-\hat{m}_{i1}$, $i = 2, 3, \ldots, n$. Then

$$\hat{m}_{i1} = \text{fl}\left(\frac{a_{i1}}{a_{11}}\right) = \frac{a_{i1}}{a_{11}}(1+\delta_{i1}), \quad |\delta_{i1}| \le \mu. \tag{14.17}$$

Thus, the error $e_{i1}^{(0)}$ in setting $a_{i1}^{(1)}$ to zero is given by

$$e_{i1}^{(0)} = a_{i1}^{(1)} - a_{i1} + \hat{m}_{i1}a_{11} \ = 0 - a_{i1} + \left(\frac{a_{i1}}{a_{11}}\right)(1+\delta_{i1})a_{11} \ = \delta_{i1}a_{i1}.$$

Let us now find the errors in computing the other elements $a_{ij}^{(1)}$ of $A^{(1)}$. The computed elements $a_{ij}^{(1)}$, $i, j = 2, \ldots, n$, are given by

$$\begin{aligned} a_{ij}^{(1)} = \text{fl}(a_{ij} - \text{fl}(\hat{m}_{i1}a_{ij})) &= (a_{ij} - \text{fl}(\hat{m}_{i1}a_{1j}))(1+\alpha_{ij}^{(1)}) \\ &= \left[a_{ij} - \hat{m}_{1j}a_{1j}(1+\beta_{ij}^{(1)})\right](1+\alpha_{ij}^{(1)}), \quad i, j = 2, \ldots, n, \end{aligned}$$

where $|\alpha_{ij}^{(1)}| \le \mu$, $|\beta_{ij}^{(1)}| \le \mu$.

The last equation can be rewritten as

$$a_{ij}^{(1)} = (a_{ij} - \hat{m}_{i1}a_{1j}) + e_{ij}^{(0)}, \quad i, j = 2, \ldots, n, \tag{14.18}$$

where

$$e_{ij}^{(0)} = \frac{\alpha_{ij}^{(1)}a_{ij}^{(1)}}{1+\alpha_{ij}^{(1)}} - \hat{m}_{i1}a_{1j}\beta_{ij}^{(1)}, \quad i, j = 2, \ldots, n. \tag{14.19}$$

From (14.18) and (14.19), we have (noting that the first row of $A^{(1)}$ is the same on the first row of A)

$$A^{(1)} = A - L^{(0)}A + E^{(0)}, \tag{14.20}$$

where

$$L^{(0)} = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ \hat{m}_{21} & 0 & \cdots & 0 \\ \vdots & & \ddots & \\ \hat{m}_{n1} & 0 & \cdots & 0 \end{pmatrix}, \qquad E^{(0)} = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ e_{21}^{(0)} & & \cdots & e_{2n}^{(0)} \\ \vdots & & \ddots & \\ e_{n1}^{(0)} & & \cdots & e_{nn}^{(0)} \end{pmatrix}.$$

Step 2. Analysis of computing $A^{(2)}$ from $A^{(1)}$ is similar.

Analogously, at the end of Step 2, we will have

$$A^{(2)} = A^{(1)} - L^{(1)}A^{(1)} + E^{(1)}, \tag{14.21}$$

where $L^{(1)}$ and $E^{(1)}$ are similarly defined.

Substituting (14.20) in (14.21), we have

$$\begin{aligned} A^{(2)} &= A^{(1)} - L^{(1)}A^{(1)} + E^{(1)} \\ &= A - L^{(0)}A + E^{(0)} - L^{(1)}A^{(1)} + E^{(1)}. \end{aligned} \tag{14.22}$$

Continuing in this way, we can write

$$A^{(n-1)} + L^{(0)}A + L^{(1)}A^{(1)} + \cdots + L^{(n-2)}A^{(n-2)} = A + E^{(0)} + E^{(1)} + \cdots + E^{(n-2)}. \tag{14.23}$$

Because

$$L^{(k-1)} = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ & 0 & & \\ \vdots & \hat{m}_{k+1,k} & \vdots & \\ \vdots & \vdots & & \vdots \\ 0 & \hat{m}_{n,k} & \cdots & 0 \end{pmatrix},$$

we have

$$L^{(k)}A^{(k)} = L^{(k)}A^{(n-1)}, \qquad k = 0, 1, 2, \ldots, n-2. \tag{14.24}$$

Thus from (14.23) and (14.24), we obtain

$$\begin{aligned} A^{(n-1)} + L^{(0)}A^{(n-1)} + L^{(1)}A^{(n-1)} + \cdots + L^{(n-2)}A^{(n-1)} \\ = A + E^{(0)} + E^{(1)} + \cdots + E^{(n-2)}. \end{aligned} \tag{14.25}$$

That is,

$$A + E^{(0)} + E^{(1)} + \cdots + E^{(n-2)} = (I + L^{(0)} + L^{(1)} + \cdots + L^{(n-2)})A^{(n-1)}. \tag{14.26}$$

Noting now that

$$I + L^{(0)} + L^{(1)} + \cdots + L^{(n-2)} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ \hat{m}_{21} & 1 & 0 & \cdots & 0 \\ \hat{m}_{31} & \hat{m}_{32} & 1 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \\ \hat{m}_{n1} & \hat{m}_{n2} & \cdots & \hat{m}_{n,n-1} & 1 \end{pmatrix} = \hat{L} \tag{14.27}$$

(the computed L) and $A^{(n-1)} = \hat{U}$ (computed U), and denoting $E^{(0)} + E^{(1)} + \cdots + E^{(n-2)}$ by E, we have from (14.26) and (14.27)

$$A + E = \hat{L}\hat{U}, \tag{14.28}$$

where the matrices $E^{(0)}, \ldots, E^{(n-2)}$ are given by

$$E^{(k-1)} = \begin{pmatrix} 0 & 0 & 0 & \cdots & \cdots & 0 \\ 0 & 0 & 0 & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & e_{k+1,k}^{(k-1)} & \cdots & e_{k+1,n}^{(k-1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & e_{n,k}^{(k-1)} & \cdots & e_{n,n}^{(k-1)} \end{pmatrix}, \quad k = 1, 2, \ldots, n-1, \tag{14.29}$$

$$e_{i,k}^{(k-1)} = a_{i,k}^{(k-1)} \delta_{i,k}, \quad i = k+1, \ldots, n \tag{14.30}$$

$$e_{ij}^{(k-1)} = \frac{\alpha_{ij}^{(k)}}{1 + \alpha_{ij}^{(k)}} a_{ij}^{(k)} - \hat{m}_{ik} a_{kj}^{(k-1)} \beta_{ij}^{(k)}, \quad i, j = k+1, \ldots, n, \tag{14.31}$$

and

$$|\delta_{ik}| \leq \mu, \qquad |\alpha_{ij}^{(k)}| \leq \mu, \tag{14.32}$$

and

$$|\beta_{ij}^{(k)}| \leq \mu. \tag{14.33}$$

We formalize the above discussion in the following theorem.

Theorem 14.7. *The computed upper and lower triangular matrices $\hat{L}$ and $\hat{U}$ produced by Gaussian elimination satisfy*

$$A + E = \hat{L}\hat{U},$$

where $\hat{U} = A^{(n-1)}$ and $\hat{L}$ is the unit lower triangular matrix of the computed multipliers given by

$$\hat{L} = \begin{pmatrix} 1 & 0 & \cdots & \cdots & 0 \\ \hat{m}_{21} & 1 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \hat{m}_{n1} & \hat{m}_{n2} & \cdots & \hat{m}_{n,n-1} & 1 \end{pmatrix}.$$

Example 14.8. Using two-digit arithmetic in the computations of $\hat{L}$ and $\hat{U}$, find the error matrix E such that $A + E = \hat{L}\hat{U}$:

$$A = \begin{pmatrix} 0.21 & 0.35 & 0.11 \\ 0.11 & 0.81 & 0.22 \\ 0.33 & 0.22 & 0.39 \end{pmatrix}.$$

Step 1.

$$\begin{aligned} & & a_{22}^{(1)} &= 0.81 - 0.52 \times 0.35 = 0.63, \\ \hat{m}_{21} &= \frac{-0.11}{0.21} = -0.52, & a_{23}^{(1)} &= 0.22 - 0.52 \times 0.11 = 0.16, \\ \hat{m}_{31} &= \frac{-0.33}{0.21} = -1.57, & a_{32}^{(1)} &= 0.22 - 1.57 \times .035 = -0.33, \\ & & a_{33}^{(1)} &= 0.39 - 1.57 \times 0.11 = 0.22, \end{aligned}$$

$$A^{(1)} = \begin{pmatrix} 0.21 & 0.35 & 0.11 \\ 0 & 0.63 & 0.16 \\ 0 & -0.33 & 0.22 \end{pmatrix},$$

$$e_{21}^{(0)} = 0 - [0.11 - 0.52 \times 0.21] = -0.0008,$$

$$e_{22}^{(0)} = 0.63 - [0.81 - 0.52 \times 0.35] = 0.0020,$$

$$e_{23}^{(0)} = 0.16 - [0.22 - 0.52 \times 0.11] = -0.0028,$$

$$e_{31}^{(0)} = 0 - [0.33 - 1.57 \times 0.21] = -0.0003,$$

$$e_{32}^{(0)} = -0.33 - [0.22 - 1.57 \times 0.35] = -0.0005,$$

$$e_{33}^{(0)} = 0.22 - [0.39 - 1.57 \times 0.11] = 0.0027,$$

$$E^{(0)} = \begin{pmatrix} 0 & 0 & 0 \\ -0.0008 & 0.0020 & -0.0028 \\ -0.0003 & -0.0005 & 0.0027 \end{pmatrix}.$$

Step 2.

$$m_{32} = \frac{0.33}{0.63} = 0.52, \qquad a_{33}^{(2)} = 0.22 + 0.52 \times 0.16 = 0.30,$$

$$A^{(2)} = \begin{pmatrix} 0.21 & 0.35 & 0.11 \\ 0 & 0.63 & 0.16 \\ 0 & 0 & 0.30 \end{pmatrix} = \hat{U}.$$

$$e_{32}^{(1)} = 0 - [-0.33 + 0.52 \times 0.63] = 0.0024,$$

$$e_{33}^{(1)} = 0.30 - [0.22 + 0.52 \times 0.16] = -0.0032,$$

$$E^{(1)} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0.0024 & -0.0032 \end{pmatrix}.$$

Thus

$$E = E^{(0)} + E^{(1)} = \begin{pmatrix} 0 & 0 & 0 \\ -0.0008 & 0.0020 & -0.0028 \\ -0.003 & 0.0019 & -0.0005 \end{pmatrix}.$$

Since

$$\hat{L} = \begin{pmatrix} 1 & 0 & 0 \\ 0.52 & 1 & 0 \\ 1.57 & -0.52 & 1 \end{pmatrix}$$

we can easily verify that $\hat{L}\hat{U} = A + E$. ∎

Bounds for the Elements of E

We now assess how large the entries of the error matrix E can be. For this purpose we assume that pivoting has been used in Gaussian elimination so that $|\hat{m}_{ik}| \le 1$. Recall that the growth factor ρ is defined by

$$\rho = \frac{\max_{i,j,k} |a_{ij}^{(k)}|}{\max_{i,j} |a_{ij}|}.$$

Let $a = \max_{i,j} |a_{ij}|$. Then from (14.29)–(14.33), we have

$$|e_{ik}^{(k-1)}| \le a\rho\mu, \quad k = 1, 2, \dots, n-1, \quad i = k+1, \dots, n,$$

and, for $i, j = k+1, \dots, n$ $(k = 1, 2, \dots, n-1)$,

$$|e_{ij}^{(k-1)}| \le \frac{\mu}{1-\mu}|a_{ij}^{(k)}| + \mu|a_{ij}^{(k-1)}| \le \frac{2}{1-\mu}a\rho\mu \qquad (\text{since } \hat{m}_{ik} \le 1).$$

Denote $\frac{\mu}{1-\mu}$ by η. Then

$$\begin{aligned}
|E| &= |E^{(0)} + \cdots + E^{(n-2)}| \le |E^{(0)}| + \cdots + |E^{(n-2)}| \\
&\le a\rho\eta \left[\begin{pmatrix} 0 & \cdot & \cdots & 0 \\ 1 & 2 & \cdots & 2 \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ 1 & 2 & \cdots & 2 \end{pmatrix} + \begin{pmatrix} 0 & \cdot & \cdot & \cdots & 0 \\ 0 & \cdot & \cdot & \cdots & 0 \\ 0 & 1 & 2 & \cdots & 2 \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ 0 & 1 & 2 & \cdots & 2 \end{pmatrix} + \cdots + \begin{pmatrix} 0 & 0 & \cdots & \cdot & 0 \\ 0 & 0 & \cdots & \cdot & 0 \\ \vdots & & & & \\ 0 & 0 & \cdots & \cdot & 0 \\ 0 & 0 & \cdots & 1 & 2 \end{pmatrix} \right] \\
&= a\rho\eta \begin{pmatrix} 0 & 0 & \cdot & \cdots & 0 \\ 1 & 2 & 2 & \cdots & 2 \\ 1 & 3 & 4 & \cdots & 4 \\ \vdots & & & & \\ 1 & 3 & 5 & \cdots & 2n-2 \end{pmatrix}.
\end{aligned} \tag{14.34}$$

Remark. Inequalities (14.34) hold elementwise. We can immediately obtain a bound in terms of norms. Thus,

$$\|E\|_\infty \le a\rho\eta(1 + 3 + \cdots + (2n-2)) \le a\rho n^2\eta\,. \tag{14.35}$$

Theorem 14.9 (round-off error analysis for GEPP). *The matrices $\hat{L}$ and $\hat{U}$, computed by Gaussian elimination with partial pivoting satisfy $A + E = \hat{L}\hat{U}$, where $\|E\|_\infty \le a\rho n^2\eta$, $a = \max_{i,j} |a_{ij}|$, and $\eta = \frac{\mu}{1-\mu}$.*

14.4.3 Backward Error Analysis for Solving $Ax = b$

We are now ready to give a backward round-off error analysis for solving $Ax = b$ using triangularization by Gaussian elimination, followed by forward elimination and back substitution.

First, from Theorem 14.7, we know that triangularization of A using Gaussian elimination yields $\hat{L}$ and $\hat{U}$ such that $A + E = \hat{L}\hat{U}$.

These $\hat{L}$ and $\hat{U}$ will then be used to solve

$$\hat{L}y = b, \quad \hat{U}x = y.$$

From Theorem 14.5 and Theorem 14.6, we know that computed solution $\hat{y}$ and $\hat{x}$ of the above two triangular systems satisfy

$$(\hat{L} + \Delta L)\hat{y} = b \quad \text{and} \quad (\hat{U} + \Delta U)\hat{x} = \hat{y}.$$

From these equations, we have

$$(\hat{U} + \Delta U)\hat{x} = (\hat{L} + \Delta L)^{-1}b$$

or

$$(\hat{L} + \Delta L)(\hat{U} + \Delta U)\hat{x} = b$$

or

$$(A + F)\hat{x} = b, \tag{14.36}$$

where

$$F = E + (\Delta L)\hat{U} + \hat{L}(\Delta U) + (\Delta L)(\Delta U). \tag{14.37}$$

(Note that $A + E = \hat{L}\hat{U}$.)

Bounds for F

From (14.37) we have

$$\|F\|_\infty \le \|E\|_\infty + \|\Delta L\|_\infty \|\hat{U}\|_\infty + \|\hat{L}\|_\infty \|\Delta U\|_\infty + \|\Delta L\|_\infty \|\Delta U\|_\infty.$$

We now obtain expressions for $\|\Delta L\|_\infty$, $\|\Delta U\|_\infty$, $\|\hat{L}\|_\infty$, and $\|\hat{U}\|_\infty$. Because

$$\hat{L} = \begin{pmatrix} 1 & & & \\ \hat{m}_{21} & \ddots & 0 & \\ \vdots & & \ddots & \\ \hat{m}_{n1} & \cdots & \hat{m}_{n,n-1} & 1 \end{pmatrix},$$

from (14.13), we obtain

$$|\Delta L| \le 1.06\mu \begin{pmatrix} 2 & & 0 & \\ 3|\hat{m}_{21}| & 2 & & \\ \vdots & & \ddots & \\ (n+1)|\hat{m}_{21}| & \cdots & 3|\hat{m}_{n,n-1}| & 2 \end{pmatrix}. \tag{14.38}$$

Assuming partial pivoting, i.e., $|\hat{m}_{ik}| \le 1,\ k = 1, 2, \ldots, n-1,\ i = k+1, \ldots, n$, we have

$$\|\hat{L}\|_\infty \le n \tag{14.39}$$

and

$$\|\Delta L\|_\infty \le \frac{n(n+3)}{2}(1.06\mu). \tag{14.40}$$

Similarly,

$$\|\hat{U}\|_\infty \leq na\rho \quad \text{(note that } U = A^{(n-1)}\text{)} \tag{14.41}$$

and using (14.15), we have

$$\|\Delta U\|_\infty \leq \frac{n(n+3)}{2} 1.06 a\rho\mu \quad \text{(note that } \max |u_{ij}| \leq a\rho\text{)}. \tag{14.42}$$

Also recall that

$$\|E\|_\infty \leq n^2 a\rho \frac{\mu}{\mu - 1}. \tag{14.43}$$

Assume that $n^2\mu \ll 1$ (which is a very reasonable assumption in practice). Then

$$\|\Delta L\|_\infty \cdot \|\Delta U\|_\infty \leq n^2 \rho a \mu. \tag{14.44}$$

Using (14.39)–(14.43) in (14.37), we have

$$\begin{aligned} \|F\|_\infty &\leq \|E\|_\infty + \|\Delta L\|_\infty \|\hat{U}\|_\infty + \|\hat{L}\|_\infty \|\Delta U\|_\infty \\ &\quad + \|\Delta L\|_\infty \|\Delta U\|_\infty \\ &\leq n^2 a\rho \frac{\mu}{\mu-1} + 1.06 n^2 (n+3) a\rho\mu + n^2 \rho a \mu. \end{aligned} \tag{14.45}$$

Since $\frac{\mu}{\mu-1} \leq 1$ and $a \leq \|A\|_\infty$, from (14.45) we can write

$$\|F\|_\infty \leq 1.06(n^3 + 5n^2)\rho \|A\|_\infty \mu. \tag{14.46}$$

Neglecting the terms involving $n^2\mu$, we have the following result.

Theorem 14.10. *The computed solution $\hat{x}$ to the linear system*

$$Ax = b$$

using Gaussian elimination with partial pivoting satisfies a perturbed system

$$(A + F)\hat{x} = b,$$

where F is defined by (14.37) *and $\|F\|_\infty \leq cn^3\rho\|A\|_\infty\mu$, where c is a small constant.*

Remarks.

1. The proceeding bound for F is grossly overestimated. In practice, this bound for F is very rarely attained.

   Wilkinson (1995) states that in practice $\|F\|_\infty$ is always less than or equal to $\mathbf{n}\mu\|\mathbf{A}\|_\infty$.

2. Making use of (14.13), (14.15), and (14.34), we can also obtain an elementwise bound for F (Exercise 14.7).

14.5 Review and Summary

In this chapter, some special topics have been discussed. Then include

- QR factorization with column pivoting;
- updating of a QR factorization;
- error analyses for LU factorization and solution of linear systems.

14.5.1 QR Factorization with Column Pivoting

If A is a rank-deficient matrix, its QR factorization can no longer be used to determine an orthonormal basis of $R(A)$. However, in this case a variation of QR factorization, called **QR factorization with column pivoting**, given by

$$Q^T AP = \begin{pmatrix} R_{11} & R_{12} \\ 0 & 0 \end{pmatrix},$$

can be used. See Theorem 14.1 for a proof.

In exact arithmetic this factorization would reveal the rank of A. However, if A is nearly rank-deficient, then a nonzero matrix R_{22} might appear in place of the last zero diagonal block in the above factorization. This is known as **rank-revealing QR factorization**. A bound of R_{22} in terms of the singular values of A has been provided in Section 14.2.

14.5.2 QR Updating

Given the QR factorization of A, the **QR updating problem** is that of finding the QR factorization of the augmented matrix (A, a_{k+1}), where a_{k+1} is an arbitrary column vector, by making use of the given QR factorization of A. A method for QR updating has been described in Algorithm 14.1.

14.5.3 Error Analysis

The backward error analysis for the following computation have been given:

1. lower and upper triangular systems using forward elimination and back substitution (Theorems 14.5 and 14.6);

2. LU factorizations using Gaussian elimination without and with partial pivoting (Theorems 14.7 and 14.9);

3. linear systems problem $Ax = b$ using Gaussian elimination with partial pivoting followed by forward elimination and back substitution (Theorem 14.10).

Bounds for the error matrix E in each case have been derived in (14.13), (14.15), (14.34), and (14.46).

We have merely attempted here to give the readers a taste of round-off error analysis, as the title of the section suggests.

The results of this chapter are already known to the reader. They have been stated earlier in the book without proofs. We have tried to give formal proofs here.

To repeat, these results say that the *forward elimination and back substitution methods for triangular systems are backward stable, whereas the stability of the Gaussian elimination process for LU factorization, and therefore for the linear system problem $Ax = b$ using the process, depends upon the size of the growth factor.*

14.5.4 Suggestions for Further Reading

For more on rank-revealing factorization, computational algorithms, and their applications, see Chan (1987), Chan and Hansen (1992), Foster (1986), and Chandrasekaran and Ipsen

(1994). Li and Zeng (2005) and Lee, Li, and Zeng (2009) have discussed rank-revealing methods with updating and downdating and their applications. See Daniel et al. (1976) for updating of the QR factorization using the Gram–Schmidt process; for downdating of the Cholesky factorization, see Bojanczyk et al. (1987) and Eldén and Park (1994). See also an earlier papers by Gill et al. (1974) and Nazareth (1989) for methods for modifying matrix factorization.

For details of round-off errors and backward stability, see Wilkinson's classics (1963, 1965) and the book by Higham (2002). Also see Ortega (1990) and Forsythe and Moler (1967).

Exercises on Chapter 14

14.1 Compute the QR factorization with column pivoting and find an orthonormal basis for $R(A)$ for each of the following matrices:

$$\text{(a)}\ A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}, \quad \text{(b)}\ A = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \quad \text{(c)}\ A = \begin{pmatrix} 1 & 2 \\ 0 & 0 \\ 1 & 2 \end{pmatrix},$$

$$\text{(d)}\ A = \begin{pmatrix} 1 & 2 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, \quad \text{(e)}\ A = \begin{pmatrix} 1 & 1 \\ 0 & 0 \\ 1 & 1 \\ 1 & 1 \end{pmatrix}.$$

14.2 Give a proof of the complete orthogonalization theorem (Theorem 14.3) starting from the QR column pivoting factorization theorem (Theorem 14.1).

14.3 Work out an algorithm to modify the QR factorization of a matrix A from which a column has been removed.

14.4 Consider the problems of solving linear systems

$$Ax = b$$

using Gaussian elimination with partial pivoting with each of the matrices from Exercise 14.1 and taking b to be the vector with all entries equal to 1 in each case. Find F in each case such that the computed solution x satisfies

$$(A + F)x = b.$$

Compare the bounds predicated by (14.46) with actual errors.

14.5 Using $\beta = 10$ and $t = 2$, compute the LU factorization using Gaussian elimination (without pivoting) for the following matrices, and find the error matrix E in each case such that $A + E = LU$:

$$\text{(a)}\ A = \begin{pmatrix} 3 & 4 \\ 5 & 6 \end{pmatrix}, \quad \text{(b)}\ A = \begin{pmatrix} 0.25 & 0.79 \\ 0.01 & 0.12 \end{pmatrix}, \quad \text{(c)}\ A = \begin{pmatrix} 10 & 9 \\ 8 & 5 \end{pmatrix},$$

$$\text{(d)}\ A = \begin{pmatrix} 4 & 1 \\ 1 & 2 \end{pmatrix}, \quad \text{(e)}\ A = \begin{pmatrix} 0.01 & 0.05 \\ 0.03 & 0.01 \end{pmatrix}.$$

14.6 Suppose now that partial pivoting has been used in computing the LU factorization of each of the above matrices of Exercise 14.5. Find again the error matrix E in each case, and compare the bounds of the entries in E predicted by (14.34) with the actual errors.

14.7 Making use of (14.13), (14.15), and (14.34), find an elementwise bound for F in Theorem 14.10 satisfying

$$(A + F)x = b.$$

14.8 From Theorems 14.5 and 14.6, show that the process of forward elimination and back substitution for lower and upper triangular systems, respectively, are backward stable.

14.9 From (14.35), conclude that the backward stability of Gaussian elimination is essentially determined by the size of the growth factor ρ.

14.10 Consider the problem of evaluating the polynomial

$$p(\alpha) = a_n\alpha^n + a_{n-1}\alpha^{n-1} + \cdots + a_0$$

by **synthetic division**:

$$\begin{aligned} p_n &= a_n, \\ p_{i-1} &= \mathrm{fl}(\alpha p_i + a_{i-1}), \qquad i = n, n-1, \ldots, 1. \end{aligned}$$

Then $p_0 = p(\alpha)$. Show that

$$p_0 = a_n(1 + \delta_n)\alpha^n + a_{n-1}(1 + \delta_{n+1})\alpha^{n-1} + \cdots + a_0(1 + \delta_0).$$

Find a bound for each $\delta_i,\ i = 0, 1, \ldots, n$. What can you say about the backward stability of the algorithm from your bounds?