ERROR ANALYSIS OF GAUSSIAN ELIMINATION METHOD FOR SOLVING SYSTEM OF LINEAR ALGEBRAIC EQUATIONS

NAI-KUAN TSAO

APPLIED MATHEMATICS RESEARCH LABORATORY

DECEMBER 1971

PROJECT 7071

AEROSPACE RESEARCH LABORATORIES
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
WRIGHT-PATTERSON AIR FORCE BASE, OHIO

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1 ORIGINATING ACTIVITY *(Corporate author)*

Aerospace Research Laboratories
Applied Mathematics Research Laboratory
Wright-Patterson AFB, Ohio 45433

2a. REPORT SECURITY CLASSIFICATION

Unclassified

2b. GROUP

3 REPORT TITLE

Error Analysis of Gaussian Elimination Method for Solving System of Linear Algebraic Equations

4 DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

Scientific Interim

5 AUTHOR(S) *(First name, middle initial, last name)*

Nai-Kuan Tsao

6 REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
December 1971	28	3

8a. CONTRACT OR GRANT NO In-house Research	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO 7071-00-14	
c. DoD Element 61102F	9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
d. DoD Subelement 681304	ARL 71-0288

10 DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11 SUPPLEMENTARY NOTES	12 SPONSORING MILITARY ACTIVITY
Tech Other	Aerospace Research Laboratories (LB) Wright-Patterson AFB, Ohio 45433

13 ABSTRACT

A posteriori forward error analysis is applied to the Gaussian elimination method for solving system of linear algebraic equations of the type $Az = p$. By attributing the generated round-off errors properly to the matrices A and p, it is shown that the computed z satisfies a new perturbed system such that $(A + \delta A)z = p + \delta p$. For large system order n, the upper bounds for δA and δp in infinite norm are then shown to be proportional to n^2, instead of n^3 obtained by the usual backward error analysis where round-off errors are attributed totally to the system matrix A. This answers partially some questions raised concerning the discrepancy between the theoretical result and practical observation of the perturbations.

DD FORM 1 NOV 65 1473

UNCLASSIFIED

Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Error Analysis						
Matrix Decomposition						
Gaussian Elimination						

*U.S.Government Printing Office: 1972 — 759-082/368

ERRATA

ARL 71-0288 December 1971

Page 23 - Reference 3 should read:

3. Tsao, N. K., A Posteriori Forward Error Analysis, Aerospace Research Laboratories Technical Report ARL 71-0287, AFSC, Wright-Patterson AFB, Ohio, December 1971.

AEROSPACE RESEARCH LABORATORIES
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
WRIGHT-PATTERSON AIR FORCE BASE, OHIO

FOREWORD

This research was accomplished while the author held a National Research Council Postdoctoral Resident Research Associateship supported by the Applied Mathematics Research Laboratory of the Aerospace Research Laboratories. The author wishes to thank Dr. Paul J. Nikolai for several suggestions which helped to improve the draft of this paper. Also thanks are due to Mrs. Barbara Geiger for a carefully typed manuscript.

ABSTRACT

A posteriori forward error analysis is applied to the Gaussian elimination method for solving system of linear algebraic equations of the type $Az = p$. By attributing the generated round-off errors properly to the matrices A and p, it is shown that the computed z satisfies a new perturbed system such that $(A + \delta A)z = p + \delta p$. For large system order n, the upper bounds for δA and δp in infinite norm are then shown to be proportional to n^2, instead of n^3 obtained by the usual backward error analysis where round-off errors are attributed totally to the system matrix A. This answers partially some questions raised concerning the discrepancy between the theoretical result and practical observation of the perturbations.

TABLE OF CONTENTS

1. Introduction.

Consider a system of n linear algebraic equations in n unknowns, written as $Az = p$ where A is a square coefficient matrix of order n, whose elements are real numbers a_{ij} with a determinant $\det(A) \neq 0$; z and p are column vectors, and the components of p are given real numbers. It is desired to find the unique solution z. Among the classes of direct methods in solving the system $Az = p$, the most popular one is perhaps the class of methods based on Gauss's idea of a systematic elimination of variables. The usual approach of the Gaussian elimination methods consists of the following steps: first, forward elimination with pivoting is used to decompose A into two factors L and U such that $LU = A$ where L is a lower triangular matrix and U is an upper triangular matrix; secondly, substitution is then used to solve the decomposed system $LUz = p$ in the sequence $Lv = p$ and $Uz = v$.

The backward error analysis of this class of methods [1,2] shows that the computed z satisfies a perturbed system such that $(A + \Delta A)z = p$. For large system order n, the upper bound for ΔA in infinite norm is proportional to n^3. This is mainly due to the multiplicative accumulation of perturbations attributed to the matrices L and U in solving the triangular systems.

By attributing the generated round-off errors properly to both A and p, a posteriori forward error analysis [3] is carried out in this paper to analyze the Gaussian elimination method. The results show that in solving the triangular systems the accumulation of perturbations is additive instead of multiplicative. It is also shown that the computed

z satisfies a new perturbed system such that $(A + \delta A)z = p + \delta p$ where the upper bounds for δA and δp are proportional to n^2 for large n. This result is then used to explain some inconsistent interpretations of the results of backward error analysis.

2. Some basic lemmas.

Throughout this paper the infinite norm of a vector x is used as our vector norm. For simplicity it is denoted as $||x||$. In association with this vector norm, the infinite matrix norm is also defined. Thus we have, for any vector x and matrix A,

$$\begin{aligned} ||x|| &= \max_i |x_i|, \\ ||A|| &= \max_i \sum_j |a_{ij}|. \end{aligned} \tag{2.1}$$

We next define $|\cdot|$ as the result of replacing all elements of the argument by the corresponding absolute values. Thus for a scalar s, $|s|$ is simply its magnitude; for a vector $v = (v_i)$, $|v|$ is a vector with elements $|v_i|$; for a matrix $M = (m_{ij})$, $|M|$ is a matrix with elements $|m_{ij}|$. Furthermore the inequality $|A| \leq |B|$ implies $|a_{ij}| \leq |b_{ij}|$ for all i,j. We have the following lemma which can easily be proved:

Lemma 2.1. With respect to the norms defined in (2.1), we have

$$\begin{aligned} &\text{(i)} \quad ||X|| = || \, |X| \, ||, \\ &\text{(ii)} \quad ||A|| = || \, |A| \, ||, \\ &\text{(iii)} \quad ||AB|| \leq || \, |A| \cdot |B| \, ||, \\ &\text{(iv)} \quad |A| \leq |B| \rightarrow ||A|| \leq ||B||. \end{aligned} \tag{2.2}$$

Now we will only consider normalized floating-point computations with t bits allocated to the mantissa of a floating-point number. Given two floating-point numbers x, y, we shall denote by $fl(x*y)$ the correctly rounded result of any floating-point operation *. For a posteriori error analysis, we need the following lemma [1]:

Lemma 2.2. Let * denote any of the operators +, -, ×, /. Then

$$(1 + \delta)fl(x*y) = x*y, \qquad |\delta| \leq 2^{-t} = u. \tag{2.3}$$

We see that Lemma 2.2 indeed tells us the a posteriori error $(\delta)fl(x*y)$ which is the difference between the exact result x*y and the computed result $fl(x*y)$. Furthermore the bound for the error can easily be estimated for each operation. For algorithms with a finite number of these basic operations, the repeated use of Lemma 2.2 will enable us to monitor the error generated at each step of computation.

3. The triangular systems.

Consider a triangular system of linear equations defined as

$$Lv = p \tag{3.1}$$

where $L = (\ell_{ij})$ is a non-singular n-th order triangular matrix and p is a given n-vector. Let us now define L_{st} as an n-th order matrix with ℓ_{st} as its (s,t)-th element and 0 or 1 as the off-diagonal or diagonal elements, respectively. Thus for a 3 × 3 system, L_{21} and L_{33} will be

$$L_{21} = \begin{bmatrix} 1 & 0 & 0 \\ \ell_{21} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{3.2}$$

and

$$L_{33} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \ell_{33} \end{bmatrix}. \tag{3.3}$$

respectively. With L_{st} defined above, we have the following theorem:

Theorem 3.1. For the lower triangular matrix L defined in (3.1), let $L^{(k)}$ denote an n-th order identity matrix with its k-th row replaced by the k-th row of L. Then we have

$$\text{(i)} \quad L_{k1}L_{k2} \cdots L_{kk} = L^{(k)}, \qquad 1 \leq k \leq n, \tag{3.4}$$

$$\text{(ii)} \quad L^{(1)}L^{(2)} \cdots L^{(n)} = L. \tag{3.5}$$

Equations (3.4) and (3.5) can easily be proved by induction. From Theorem 3.1, we see that solving (3.1) is equivalent to solving a decomposed system

$$L^{(1)}L^{(2)} \cdots L^{(n)}v = p \tag{3.6}$$

which can be solved in the sequence

$$
\begin{aligned}
p^{(0)} &= p, \\
L^{(1)}p^{(1)} &= p^{(0)}, \\
L^{(2)}p^{(2)} &= p^{(1)}, \\
&\vdots \\
L^{(n)}p^{(n)} &= p^{(n-1)}, \\
v &= p^{(n)}.
\end{aligned}
\tag{3.7}
$$

Again by Theorem 3.1, each of the equations in (3.7), say $L^{(k)}p^{(k)} = p^{(k-1)}$, is equivalent to

$$
L_{k1}L_{k2} \cdots L_{kk}p^{(k)} = p^{(k-1)} \tag{3.8}
$$

which can also be solved in a new sequence

$$
\begin{aligned}
p^{(k),0} &= p^{(k-1)}, \\
L_{k1}p^{(k),1} &= p^{(k),0}, \\
L_{k2}p^{(k),2} &= p^{(k),1}, \\
&\vdots \\
L_{kk}p^{(k),k} &= p^{(k),k-1}, \\
p^{(k)} &= p^{(k),k}.
\end{aligned}
\tag{3.9}
$$

Expressing a specific equation of (3.9) in detail, say $L_{kj}p^{(k),j} = p^{(k),j-1}$, $1 \leq j < k$, we have

$$
\begin{array}{l}
\\
\text{k-th row} \rightarrow
\end{array}
\begin{bmatrix}
1 & & & & & & & \\
& \ddots & & & & & & \\
& & 1 & & & & & \\
& & & \ddots & & & & \\
0 & \cdots & 0 & \ell_{kj} & 0 & \cdots & 1 & \\
& & & & & & & \ddots \\
& & & & & & & 1
\end{bmatrix}
\begin{bmatrix} p_1^{(k),j} \\ \vdots \\ p_j^{(k),j} \\ \vdots \\ p_k^{(k),j} \\ \vdots \\ p_n^{(k),j} \end{bmatrix}
=
$$

$$
\begin{array}{c}
\uparrow \\
\text{j-th column}
\end{array}
\qquad (3.10)
$$

$$
\begin{bmatrix} p_1^{(k),j-1} \\ \vdots \\ p_j^{(k),j-1} \\ \vdots \\ p_k^{(k),j-1} \\ \vdots \\ p_n^{(k),j-1} \end{bmatrix}
$$

Equation (3.10) shows that the only non-trivial computation is that to obtain

$$
\begin{aligned}
p_k^{(k),j} &= -\ell_{kj} p_j^{(k),j} + p_k^{(k),j-1} \\
&= -\ell_{kj} p_j^{(k),j-1} + p_k^{(k),j-1}, \qquad 1 \leq j < k.
\end{aligned}
\qquad (3.11)
$$

For $j = k$ we simply have

$$p_k^{(k),k} = p_k^{(k),k-1}/\ell_{kk}. \tag{3.12}$$

Thus we have reduced the solution of a general lower triangular system to the solution of a sequence of decompositions in which at most two elementary operations are needed for each decomposition. If we define $s^{(k),j} = fl(-\ell_{kj}p_j^{(k),j})$, then computationally (3.11) and (3.12) become

$$\left.\begin{aligned} s^{(k),j} &= fl(-\ell_{kj}p_j^{(k),j}) \\ p_k^{(k),j} &= fl(s^{(k),j} + p_k^{(k),j-1}) \end{aligned}\right\}, \quad 1 \le j < k, \tag{3.13}$$

$$p_k^{(k),k} = fl(p_k^{(k),k-1}/\ell_{kk}). \tag{3.14}$$

Applying Lemma 2.2 to (3.13) and (3.14), we have

$$\begin{aligned} &\ell_{kj}p_j^{(k),j} + p_k^{(k),j} + p_k^{(k),j}\delta_{kj} + s^{(k),j}\delta'_{kj} = \\ &p_k^{(k),j-1}, \qquad |\delta_{kj}|, |\delta'_{kj}| \le u, \ 1 \le j \le k, \end{aligned} \tag{3.15}$$

$$\ell_{kk}p_k^{(k),k} + \ell_{kk}p_k^{(k),k}\delta_{kk} = p_k^{(k),k-1}, \qquad |\delta_{kk}| \le u. \tag{3.16}$$

In matrix formulation, we have

$$L_{kj}p^{(k),j} + e^{(k),j} = p^{(k),j-1}, \qquad 1 \le j < k, \tag{3.17}$$

and

$$L_{kk}p^{(k),k} + e^{(k),k} = p^{(k),k-1} \tag{3.18}$$

where $e^{(k),j}$ and $e^{(k),k}$ are n-vectors whose only non-zero elements are the k-th elements

$$e_k^{(k),j} = p_k^{(k),j}\delta_{kj} + s^{(k),j}\delta'_{kj}, \qquad 1 \leq j < k, \tag{3.19}$$

and

$$e_k^{(k),k} = \ell_{kk}p^{(k),k}\delta_{kk}. \tag{3.20}$$

Premultiplying both sides of (3.18) by $L_{k1}L_{k2} \cdots L_{k,k-1}$ and using (3.17), we have

$$L_{k1}L_{k2} \cdots L_{kk}p^{(k)} + \varepsilon^{(k)} = p^{(k-1)} \tag{3.21}$$

where

$$\begin{aligned}\varepsilon^{(k)} &= e^{(k),1} + L_{k1}e^{(k),2} + L_{k1}L_{k2}e^{(k),3} + \cdots + \\ &L_{k1}L_{k2} \cdots L_{k,k-1}e^{(k),k}.\end{aligned} \tag{3.22}$$

Now the only effect of premultiplying L_{kj} with $e^{(k),i}$ is to add an additional term $\ell_{kj}e_j^{(k),i}$ to the k-th element of $e^{(k),j}$; since $e_j^{(k),i}$ is zero for $j \neq k$, hence we have

$$L_{ki}e^{(k),j} = e^{(k),j}, \qquad i \neq j. \tag{3.23}$$

Applying (3.23) to (3.22), we have

$$\varepsilon^{(k)} = \sum_{i=1}^{k} e^{(k),i}. \tag{3.24}$$

Furthermore, the only non-zero element of $\varepsilon^{(k)}$ is the k-th element $\varepsilon_k^{(k)}$ which is equal to

$$\varepsilon_k^{(k)} = \sum_{j=1}^{k-1} \left(p_k^{(k),j}\delta_{kj} + s^{(k),j}\delta'_{kj}\right) + \ell_{kk}p_k^{(k),k}\delta_{kk}. \tag{3.25}$$

Equation (3.21) can also be expressed as

$$L^{(k)}p^{(k)} + \varepsilon^{(k)} = p^{(k-1)}. \tag{3.26}$$

Extending (3.26) to k = 1, 2, ···, n and combining these equations, we have

$$L^{(1)}L^{(2)} \cdots L^{(n)}v + e = p \tag{3.27}$$

where

$$\begin{aligned} e = {} & \varepsilon^{(1)} + L^{(1)}\varepsilon^{(2)} + L^{(1)}L^{(2)}\varepsilon^{(3)} + \cdots + \\ & L^{(1)}L^{(2)} \cdots L^{(n-1)}\varepsilon^{(n)}. \end{aligned} \tag{3.28}$$

Again we have

$$L^{(j)}\varepsilon^{(i)} = \varepsilon^{(i)}, \qquad j \leq i-1, \tag{3.29}$$

since the first i-1 elements of $\varepsilon^{(i)}$ are zero. Hence (3.28) simplifies to

$$e = \sum_{i=1}^{n} \varepsilon^{(i)}. \tag{3.30}$$

Now if we define

$$\rho_p = \max_{k,j}\left[|p_k^{(k),j}|, |s^{(k),j}|\right], \qquad 1 \leq k \leq n, \ 1 \leq j \leq k, \tag{3.31}$$

and

$$\sigma_L = \max_{k}|\ell_{kk}|, \tag{3.32}$$

Then an upper bound for the k-th element of e, or $\varepsilon_k^{(k)}$ in (3.25), can be estimated as

$$|\varepsilon_k^{(k)}| \leq [2(k-1) + \sigma_L]\rho_p u, \qquad 1 \leq k \leq n. \tag{3.33}$$

Thus we have proved the following theorem:

Theorem 3.2. In solving the triangular system of equations (3.1), the solution v computed by the sequential decompositions of p satisfies the equation

$$Lv + e = p \tag{3.34}$$

where e is defined by (3.30); furthermore

$$|e| \leq \begin{bmatrix} 2(0) + \sigma_L \\ 2(1) + \sigma_L \\ 2(2) + \sigma_L \\ \vdots \\ 2(n-1) + \sigma_L \end{bmatrix} \rho_p u, \tag{3.35}$$

$$||e|| \leq || \, |e| \, || \leq [2(n-1) + \sigma_L]\rho_p u. \tag{3.36}$$

Now we observe that (3.8) can also be written as

$$L^{(k)}p^{(k)} = \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & \ddots & & & \\ \ell_{k1} & \ell_{k2} & \cdots & \ell_{kk} & & \\ & & & & \ddots & \\ & & & & & 1 \end{bmatrix} \begin{bmatrix} p_1^{(k)} \\ p_2^{(k)} \\ \vdots \\ p_k^{(k)} \\ \vdots \\ p_n^{(k)} \end{bmatrix} = \tag{3.37}$$

$$\begin{bmatrix} p_1^{(k-1)} \\ p_2^{(k-1)} \\ \vdots \\ p_k^{(k-1)} \\ \vdots \\ p_n^{(k-1)} \end{bmatrix},$$

whose solutions are easily obtained as

$$p_k^{(k)} = f_\ell\left[\frac{1}{\ell_{kk}}\left(-\sum_{i=1}^{k-1}\ell_{ki}p_i^{(k)} + p_k^{(k-1)}\right)\right],$$

and (3.38)

$$p_j^{(k)} = p_j^{(k-1)}, \qquad j \neq k.$$

The algorithm expressed in equation (3.38) is exactly the substitution algorithm used in Gaussian elimination to solve the decomposed triangular systems. Furthermore, if the inner product in (3.38) is evaluated first, then the computations are executed in exactly the same sequence as that in (3.9). Thus computationally the decomposition algorithm expressed in (3.7) and (3.9) are equivalent to the conventional substitution algorithm. However, if we follow the usual backward error analysis, the computed v can be shown [1] to satisfy:

$$(L + \Delta L)v = p \tag{3.39}$$

where

$$||\Delta L|| \leq \frac{1.01}{2} n(n+1) \max_{i,j} |\ell_{ij}| u. \tag{3.40}$$

Comparing (3.36) and (3.40), we have the following comments:

(a) The bound for e in (3.36) is a function of σ_L, ρ_p and n in which ρ_p and σ_L are relatively stationary for computations with sufficient precision. Hence if the system order n is large, the bound is proportional to n. However, we see $||\Delta L||$ is proportional to n^2 for large n. Since these bounds are used to bound the relative error between the computed solution and exact solution, (3.40) is an overestimation when compared with practical results.

(b) Computationally, using (3.36) is not only practical as it enables us to monitor the round-off error step by step, but it is also realistic as it depends on both matrix L and the n-vector p. For example, if n = 1 and p = 0, then it is obvious that $||e|| \leq 0$ and this is what

happens in actual computation. On the other hand, (3.40) depends only on the matrix L, hence intuitively and computationally it is a "static" overestimation with very little information regarding what actually happens in the process of computation.

4. The general systems.

Now we can consider solving a general system of linear equations defined as

$$Az = p \tag{4.1}$$

where A is an n-th order non-singular matrix and p is an n-vector. It is rather trivial to show that by properly interchanging rows or columns, the permuted A, for simplicity we will still call it A, can be decomposed into a product of L, and U such that A = LU where L is a unit triangular matrix and U is an upper triangular matrix. The usual row-pivoting strategy makes the decomposition possible by proper row interchanges. We will consider the partial row-pivoting strategy in which a row is chosen as pivoting row such that it has the largest magnitude coefficient for the variable to be eliminated. We will also assume that row permutations are done in advance so that no pivoting is necessary.

Now the decomposition consists of computing a sequence of matrices $A^{(1)} = A, A^{(2)}, \cdots, A^{(n)}$, where the matrix $A^{(k)}$ is zero below the diagonal in the first k-1 columns. The matrix $A^{(k+1)}$ is obtained from $A^{(k)}$ by subtracting a multiple of the k-th row from each of the rows below it; the rest of $A^{(k)}$ is not changed. The multipliers are chosen

so that if there were no round-off errors, $A^{(k+1)}$ would have zeros below the diagonal in the k-th column. We do not calculate these elements but take them to be zero by definition. More precisely, let $A^{(k)}$ have elements $a_{ij}^{(k)}$. Then let

$$m_{ik} = fl\left(a_{ik}^{(k)}/a_{kk}^{(k)}\right), \qquad k+1 \leq i, \tag{4.2}$$

and

$$a_{ij}^{(k+1)} = \begin{cases} 0 & , \quad j = k,\ k+1 \leq i, \\ fl\left(a_{ij}^{(k)} - m_{ik}a_{kj}^{(k)}\right), & k+1 \leq j,\ k+1 \leq i, \\ a_{ij}^{(k)} & \text{otherwise.} \end{cases} \tag{4.3}$$

These steps are carried out for k = 1, 2, $\cdots$, n-1. Finally, let

$$U = A^{(n)}, \tag{4.4}$$

and

$$L = \begin{bmatrix} 1 & & & & \\ m_{21} & 1 & & & \\ m_{31} & m_{32} & \cdot & & \\ \cdot & \cdot & & \cdot & \\ \cdot & \cdot & & & \cdot \\ m_{n1} & m_{n2} & \cdot & \cdot & \cdot & 1 \end{bmatrix}. \tag{4.5}$$

To compute (4.3), let us further define

$$s_{ij}^{(k)} = fl\left(-m_{ik}a_{kj}^{(k)}\right), \qquad k+1 \leq j,\ k+1 \leq i. \tag{4.6}$$

So we have

$$a_{ij}^{(k+1)} = \begin{cases} 0 & , \quad j = k,\ k+1 \le i, \\ fl\left\{a_{ij}^{(k)} + s_{ij}^{(k)}\right\}, & k+1 \le j,\ k+1 \le i, \\ a_{ij}^{(k)} & \text{otherwise.} \end{cases} \tag{4.7}$$

Applying Lemma 2.2 to (4.2), (4.6), and (4.7), we have

$$(1 + \delta_{ik})m_{ik} = a_{ik}^{(k)}/a_{kk}^{(k)}, \qquad k+1 \le i, \tag{4.8}$$

$$(1 + \delta_{ij})s_{ij}^{(k)} = -m_{ik}a_{kj}^{(k)}, \qquad k+1 \le j,\ k+1 \le i, \tag{4.9}$$

$$a_{ik}^{(k+1)} = 0 \qquad , \qquad k+1 \le i, \tag{4.10}$$

$$(1 + \delta'_{ij})a_{ij}^{(k+1)} = a_{ij}^{(k)} + s_{ij}^{(k)}, \qquad k+1 \le j,\ k+1 \le i, \tag{4.11}$$

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} \qquad \text{otherwise.} \tag{4.12}$$

Combining (4.8) and (4.10), we have

$$a_{ik}^{(k+1)} = 0 = a_{ik}^{(k)} - m_{ik}a_{kk}^{(k)} - m_{ik}a_{kk}^{(k)}\delta_{ik}, \qquad k+1 \le i. \tag{4.13}$$

Combining (4.9) and (4.11), we obtain

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - m_{ik}a_{kj}^{(k)} - s_{ij}^{(k)}\delta_{ij} - a_{ij}^{(k+1)}\delta'_{ij}, \qquad k+1 \le j,\ k+1 \le i. \tag{4.14}$$

In matrix notation (4.13), (4.14), and (4.12) are combined to give

$$A^{(k+1)} = A^{(k)} - L^{(k)}A^{(k)} - E^{(k)} \tag{4.15}$$

where $E^{(k)} = \left(\varepsilon_{ij}^{(k)}\right)$ with

$$\varepsilon_{ij}^{(k)} = \begin{cases} m_{ik}a_{kk}^{(k)}\delta_{ik} & , \quad k+1 \leq i,\ j = k, \\ +s_{ij}^{(k)}\delta_{ij} + a_{ij}^{(k+1)}\delta_{ij}', & k+1 \leq i,\ k+1 \leq j, \\ 0 & \text{otherwise}, \end{cases} \tag{4.16}$$

and

$$L^{(k)} = \begin{bmatrix} & & 0 & & \\ & & 0 & & \\ & & \cdot & & \\ & & \cdot & & \\ 0 & & 0 & & 0 \\ & & m_{k+1,k} & & \\ & & m_{k+2,k} & & \\ & & \cdot & & \\ & & \cdot & & \\ & & m_{nk} & & \end{bmatrix} . \tag{4.17}$$

Adding (4.15) for $k = 1, 2, \cdots, n-1$, we have

$$A^{(n)} + \sum_{k=1}^{n-1} L^{(k)}A^{(k)} = A - \sum_{k=1}^{n-1} E^{(k)}. \tag{4.18}$$

Since the matrix $L^{(k)}A^{(k)}$ depends only upon the k-th row of $A^{(k)}$ which is equal to the k-th row of $A^{(n)}$, we thus have

$$\left(I + \sum_{k=1}^{n-1} L^{(k)}\right) A^{(n)} = A - \sum_{k=1}^{n-1} E^{(k)}. \tag{4.19}$$

That is,

$$LU = A - E \tag{4.20}$$

where L and U are defined by (4.5) and (4.4) and where

$$E = \sum_{k=1}^{n-1} E^{(k)}. \tag{4.21}$$

To bound E we observe that since $m_{ik} = fl\left(a_{ik}^{(k)}/a_{kk}^{(k)}\right)$, the use of pivoting implies that $|m_{ik}| \leq 1$ for all i,k. Furthermore from (4.6) we have

$$|s_{ij}^{(k)}| \leq |a_{kj}^{(k)}|, \qquad k+1 \leq j,\ k+1 \leq i. \tag{4.22}$$

Hence if we define

$$\sigma = \max_{i,j,k}\left[|s_{ij}^{(k)}|,\ |a_{ij}^{(k)}|\right] = \max_{i,j,k}\left[|a_{ij}^{(k)}|\right], \tag{4.23}$$

then from (4.16) we have

$$|\varepsilon_{ij}^{(k)}| \leq \begin{cases} \sigma u, & j = k,\ k+1 \leq i, \\ 2\sigma u, & k+1 \leq i,\ k+1 \leq j, \\ 0 & \text{otherwise.} \end{cases} \tag{4.24}$$

Following (4.21), we add the $\varepsilon_{ij}^{(k)}$ together to get E. And we have

$$|E| \leq \sigma u \begin{bmatrix} 0 & 0 & . & . & . & 0 \\ 1 & 2 & . & . & . & 2 \\ 1 & 3 & 4 & . & . & 4 \\ . & . & . & . & . & . \\ 1 & 3 & 5 & . & 2n-4 & 2(n-2) \\ 1 & 3 & 5 & . & 2n-3 & 2(n-1) \end{bmatrix} . \qquad (4.25)$$

From (4.25) we also have $|| \ |E| \ || \leq (1+3+5+\cdots+2n-3+2(n-1))\sigma u = (n^2-1)\sigma u$. Thus we have proved the following theorem:

Theorem 4.1. The matrices L and U computed by Gaussian elimination with row-pivoting, using floating-point arithmetic, satisfy

$$LU + E = A. \qquad (4.26)$$

Furthermore,

$$||E|| \leq || \ |E| \ || \leq (n^2-1)\sigma u. \qquad (4.27)$$

Once the matrix A has been decomposed, the results in section 3 can therefore be used to solve the decomposed triangular systems. Thus after decomposition we have

$$LU + E = A. \qquad (4.28)$$

Now in solving $LUz = p$ in the sequence $Lv = p$ and $Uz = v$ by substitution algorithm, Theorem 3.2 tells us that the computed v and z satisfy

$$Lv + e_1 = p, \qquad (4.29)$$

and

$$Uz + e_2 = v \tag{4.30}$$

where

$$|e_1| \le \begin{bmatrix} \sigma_L \\ 2+\sigma_L \\ 2(2)+\sigma_L \\ \cdot \\ \cdot \\ \cdot \\ 2(n-1)+\sigma_L \end{bmatrix} \rho_p u, \qquad |e_2| \le \begin{bmatrix} 2(n-1)+\sigma_U \\ \cdot \\ \cdot \\ \cdot \\ 2(2)+\sigma_U \\ 2+\sigma_U \\ \sigma_U \end{bmatrix} \rho_v u. \tag{4.31}$$

Combining (4.28), (4.29), and (4.30), we have

$$(A - E)z = p - e_1 - Le_2. \tag{4.32}$$

Thus the computed z satisfies a new system with perturbed A and perturbed p. Let $\delta A = -E$, $\delta p = -e_1 - Le_2$, the bound for δA is estimated as

$$||\delta A|| \le (n^2-1)\sigma u. \tag{4.33}$$

Applying Lemma 2.1 to δp, we have

$$\begin{aligned} ||\delta p|| &\le ||e_1|| + || \, |L| \cdot |e_2| \, || \\ &\le [2(n-1) + \sigma_L]\rho_p u + [n^2 - n + n\sigma_U]\rho_v u. \end{aligned} \tag{4.34}$$

We should note that $\sigma_L = 1$ and $\sigma_U \leq \sigma$ from the definition of L and U. Furthermore, if we denote $\rho = \max[\rho_p, \rho_v]$, then (4.34) can be simplified as

$$||\delta p|| \leq (n^2 + n - 1 + n\sigma)\rho u \tag{4.35}$$

where

$$\rho = \max[\rho_p, \rho_v]. \tag{4.36}$$

Thus we have proved the following theorem:

Theorem 4.2. The solution z computed by Gaussian elimination with row-pivoting and substitution satisfies the equation

$$(A + \delta A)z = p + \delta p \tag{4.37}$$

where $\delta A = -E$ and $\delta p = -e_1 - Le_2$. Furthermore,

$$||\delta A|| \leq (n^2 - 1)\sigma u, \tag{4.38}$$

$$||\delta p|| \leq (n^2 + n - 1 + n\sigma)\rho u. \tag{4.39}$$

We observe that δA is essentially in the same format as the perturbation matrix obtained by Forsythe and Moler [1] in the decomposition of A. This is no surprise to us since they have also used Lemma 2.2 in part of their analysis. However, the overall result is different. In fact, their result shows [1] that the computed z satisfies

$$(A + \Delta A)z = p \tag{4.40}$$

where

$$||\Delta A|| \leq 1.01(n^3 + 3n^2)\sigma u. \tag{4.41}$$

The upper bound for ΔA in (4.41) is therefore proportional to n^3 for large system order n. The comments at the end of section 3 also apply here.

We further note that the factor n^3 in (4.41) is due to the solution of the decomposed triangular systems. Hence if we use higher precision to solve the decomposed systems, this term should be reduced drastically and hence we should expect to have much more accurate results. However, this is not true in practical observations. Indeed, if the decomposition is already in error, the improvement to solution accuracy using high precision arithmetic in solving the triangular systems is very little, if not naught. The reason can be explained by the results of our analysis. We see that the perturbations due to decomposition of A is δA and the perturbations due to the solution of triangular systems is δp. The upper bounds for δA and δp, shown in (4.38) and (4.39), show that they are of the same order n^2 for large n. So unless higher precision arithmetic is used for both the decomposition of A and the decomposition of p, there is very little gain in using higher precision arithmetic in only one process.

5. Conclusions.

We have shown by using a posteriori error analysis that the perturbations due to decomposition process and due to solution of triangular matrices are of the same order n^2 for large n. This approach of

attributing generated errors to both matrices A and p is intuitively and computationally natural. In fact, the decomposed L and U are kept in computer memory and are not perturbed in solving the triangular systems. Hence the perturbations in the solution of triangular systems should be attributed to the vector p which is actually perturbed. There is of course another advantage of using a posteriori error analysis: that the "dynamic" behavior of the computational process can be monitored step by step.

We should also note that the "efficient" Gaussian process is essentially an "analytic" process [3]. In other words, this algorithm tries to decompose p such that $Az = p$ for given A, p. Algebraically z is unique whether it is obtained by satisfying $Az = p$ or by directly evaluating $z = A^{-1}p$. However, computationally the closeness of Az to p does not guarantee the closeness of z to $A^{-1}p$. Hence the results of the a posteriori error analysis can only tell us the difference between the computed decomposition LU and the exact decomposition A or the difference between the computed decomposition LUz and the "exact" decomposition p. In order to find the difference between computed solution z and the exact solution $A^{-1}p$, we need to know A^{-1} whose information has been inadvertently by-passed in the Gaussian process. Therefore "efficient" algorithms are not necessarily "good" algorithms in other respects.

REFERENCES

1. Forsythe, G. E. and C. B. Moler, Computer Solution of Linear Algebraic Systems, Prentice Hall, Englewood Cliffs, N. J., 1967.
2. Wilkinson, J. H., Rounding Errors in Algebraic Processes, Prentice Hall, Englewood Cliffs, N. J., 1963.
3. Tsao, N. K., A Posteriori Forward Error Analysis. (To be submitted to SIAM J. Numer. Analysis.)