Consiglio Nazionale delle Ricerche

ISTITUTO DI ELABORAZIONE
DELLA INFORMAZIONE

PISA

STABILITY AND GOOD-BEHAVIOUR OF ALGORITHMS FOR SOLVING LINEAR SYSTEMS

M. Arioli, F. Romani

Nota interna B4-35
Luglio 1986

STABILITY AND GOOD-BEHAVIOUR
OF ALGORITHMS FOR SOLVING LINEAR SYSTEMS

Mario Arioli and Francesco Romani
IEI-CNR, Via S. Maria 46, 56100 Pisa, ITALY

Abstract

The behaviours of both the error and the residual in the solution of a linear system are studied, by assuming representation and roundoff errors to be random variables. Two quantities which measure the mean value of the linear part of the error and the mean value of the linear part of the residual are introduced, giving stability and good behaviour criteria.

These criteria are applied to various algorithms (Gaussian elimination with different types of pivoting, ortogonalization techniques). In addition the influence of row-scaling is studied.

1. Introduction and preliminaries.

Let us consider a linear system

$$A\,\mathbf{x} = \mathbf{b}, \tag{1.1}$$

where A is a square nonsingular real matrix of order n. Let $\mathcal{A}$ be an algorithm for solving (1.1). In this paper the behaviours of both the error and the residual in the solution of linear systems are investigated. Using probabilistic techniques, we introduce two quantities $\mathfrak{e}(A)$ and $\mathfrak{w}(A)$ which measure the mean values of the linear part of the error and residual vectors resulting from the application of $\mathcal{A}$ for solving (1.1).

We also introduce a new condition number similar to that defined in [Fletcher (1985)] and strictly related to the Skeel condition number [Skeel (1979, 1981)]. This conditioning measure is invariant under row scaling and can assume values much lower than the ones assumed by classical conditions numbers.

Statistical Stability and Good-Behaviour of an algorithm $\mathcal{A}$ are defined in section 2.

In order to study the algorithmic error in some classical methods we assume that the only significant errors in the algorithmic process are due to the representation of intermediate data. This is in good agreement with the results of the studies of [Kulish-Miranker (1986)] on arithmetic operations, and can be achieved, for example, by using multiple precision in intermediate computations.

The results of the analysis explain conveniently the well known experimental behaviour of classical methods (Gaussian Elimination, Orthogonalization method, scaling procedures).

Finally numerical experiments are given using special classes of matrices which have a high gap among the new condition number and the classical ones.

The notation $A^{-1} = Z = (z_{ij})$ is used, moreover $\mathbb{A}$ denotes the three way array $\mathbb{A} = (t_{ijk})$, $t_{ijk} = z_{ij}\, a_{jk}$. The symbol * denotes the Hadamard product (i.e. componentwise multiplication) between two arrays of the same size. The symbol $\| \,.\, \|_p$ denotes the Holder p-norm of vectors and the corresponding induced norm for matrices and $\| \,.\, \|_F$ denotes the Frobenius norm of matrices, and three-way arrays; $|A|$ denotes the array of the absolute values of the entries of A; $\mathbb{E}(\xi)$ denotes the expected value of the random variable ξ.

Let us define the domain $\mathbf{B}_n$ (the unitary ball) and its measure Γ as

$$\mathbf{B}_n = \{ \mathbf{x} \in \mathbf{R}^n \mid \| \mathbf{x} \|_2 = 1 \}, \qquad \Gamma = \int_{\mathbf{B}_n} d\mathbf{x} .$$

The mean of a function $f : \mathbf{R}^n \to \mathbf{R}$, is defined as

$$\operatorname*{Mean}_{\| \mathbf{x} \|_2 = 1} f(\mathbf{x}) = \Gamma^{-1} \int_{\mathbf{B}_n} f(\mathbf{x})\, d\mathbf{x} .$$

The classical condition number of a nonsingular matrix is $k_p(A)=\|A^{-1}\|_p\|A\|_p$. The quantities $k_F(A)=\|A^{-1}\|_F\|A\|_F$ and $k_2(A)$ are called the *Frobenius* and *spectral condition numbers* of A. The two condition numbers are connected by the relation $n^{-1} k_F(A) \le k_2(A) \le k_F(A)$.

The quantity $\|\mathbb{A}\|_F = [\sum_{r=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} z_{ri}^2\, a_{ij}^2]^{1/2}$ is called the *tensorial condition number* of A. It is easy to see that $\| \mathbb{A} \|_F \le k_F(A)$. Moreover $\| \mathbb{A} \|_F$ can be arbitrarily smaller than $k_F(A)$; e.g. for diagonal matrices the tensorial conditioning is $n^{1/2}$ and the spectral and Frobenius conditioning can be arbitrarily larger. On the other hand the if the columns of A have the same length the two condition numbers are equivalent.

Definition 1.1. A matrix A is said to be *row-equilibrated* if all its rows have the same euclidean norm, i.e. $\sum_{j=1}^{n} a_{ij}^{\,2} = \gamma^2$, $i=1,2, \dots ,n$. □

Proposition 1.1. If the matrix A is row-equilibrated its Frobenius condition number and the tensorial one are connected by the relation:

$$\| \mathbb{A} \|_F = n^{1/2} k_F(A).$$

Proof. One has

$$\| \mathbb{A} \|_F^2 = \sum_{i=1}^{n} (\sum_{r=1}^{n} z_{ri}^2) (\sum_{j=1}^{n} a_{ij}^2) = \gamma^2 \sum_{i=1}^{n} \sum_{r=1}^{n} z_{ri}^2 = \gamma^2 \| A^{-1} \|_F^2 . \qquad □$$

Another condition number was introduced by Skeel (1979), namely $C_p(A) = \| \, |A^{-1}| \, |A| \, \|_p$, this measure is strictly related with the tensorial condition, in fact the quantities $C_\infty(A)$ and $\| \mathbb{A} \|_F$ are the maximum and Frobenius norms of the same rectangular matrix. Let $B = (b_{rk})$, $b_{rk} = |z_{ij}|\,|a_{jr}|$, $1 \le i,j,r \le n$, $k=i+(j-1)n$, one has $C_\infty(A) = \| B \|_\infty$ and $\| \mathbb{A} \|_F = \| B \|_F$.

It is remarkable that all these conditioning measures are only of theoretical interest as their computation requires the inverse of A.

For test purposes, in order to evidentiate the dependence of algorithmic errors on the various condition numbers, we will use test matrices for which tensorial and Skeel condition numbers are much smaller than classical ones.

A non-trivial class of matrices with this property is the class of the *Vandermonde column scaled matrices* which depend on a parameter λ: the generic matrix of the class, say $A_\lambda=(a_{ij})$ is defined by $a_{ij} = \lambda^{(i-1)(j-1)} d_j$, with d_j chosen to minimize $C_\infty(A_\lambda)$.

In Fig.1 you can see the classical condition $k_\infty(A_\lambda)$ and the tensorial condition $\|\mathbb{A}_\lambda\|_F$ plotted versus the Skeel condition $C_\infty(A_\lambda)$ in a logarithmic scale, for some matrices of this class (with n=5). It is remarkable that as λ tends to 0 the tensorial condition tends to $n^{1/2}$, the Skeel condition tend to 1 and the classical condition numbers tend to the infinity. E.g. the matrix A_λ with λ = 0.02149 and n=5, is the following:

$$A_\lambda = \begin{bmatrix} 0.263E-11 & 0.311E-05 & 0.529E-01 & 0.145E+02 & 0.571E+02 \\ 0.263E-11 & 0.668E-07 & 0.244E-04 & 0.144E-03 & 0.122E-04 \\ 0.263E-11 & 0.144E-08 & 0.113E-07 & 0.143E-08 & 0.260E-11 \\ 0.263E-11 & 0.309E-10 & 0.521E-11 & 0.142E-13 & 0.555E-18 \\ 0.263E-11 & 0.663E-12 & 0.241E-14 & 0.141E-18 & 0.118E-24 \end{bmatrix},$$

its inverse is

$$A_\lambda^{-1} = \begin{bmatrix} 0.153E-04 & -0.390E+02 & 0.395E+07 & -0.854E+10 & 0.389E+12 \\ -0.334E-04 & 0.159E+03 & -0.160E+08 & 0.339E+11 & -0.339E+11 \\ 0.197E-03 & -0.940E+03 & 0.927E+08 & -0.440E+10 & 0.431E+10 \\ -0.155E-02 & 0.726E+04 & -0.160E+08 & 0.747E+09 & -0.730E+09 \\ 0.179E-01 & -0.184E+04 & 0.399E+07 & -0.186E+09 & 0.182E+09 \end{bmatrix},$$

and $k_\infty(A_\lambda)$ = 0.2270E+14, $C_\infty(A_\lambda)$ = 1.681, $\|\mathbb{A}_\lambda\|_F$ = 2.427.

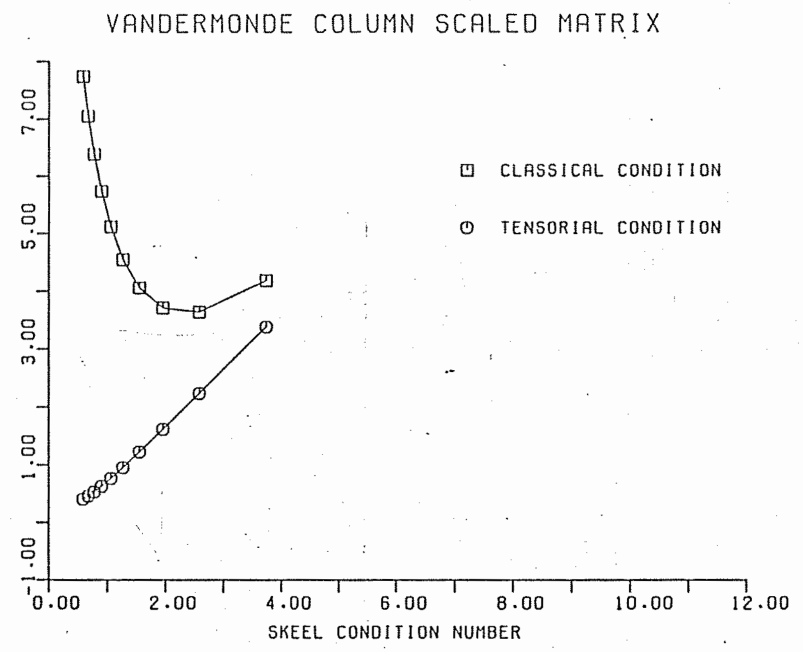

Fig. 1. Classical and tensorial condition numbers plotted versus the Skeel condition number for a 5x5 Vandermonde column scaled matrix.

2. Error Analysis.

Let $\tilde{A}$ and $\tilde{b}$ be slightly perturbed values of A and **b**, the matrix $\tilde{A}$ can be expressed as $A + (A*E')$ where $E' = (e'_{ij})$ is the matrix of the relative error terms of the entries of A. We assume $e'_{ij} = 0$ when $a_{ij} = 0$. Analogously we write $\tilde{\mathbf{b}} = \mathbf{b} + \mathbf{b}*\mathbf{e}''$ with $e''_i = 0$ when $b_i = 0$. The perturbed system will be $\tilde{A}\tilde{\mathbf{x}} = \tilde{\mathbf{b}}$ or, equivalently,

$$A[I + A^{-1}(A*E')]\,\tilde{\mathbf{x}} = \mathbf{b} + \mathbf{b}*\mathbf{e}''.$$

In the following we assume $\| A^{-1}(A*E') \|_2 < 1$ in order to ensure $\tilde{A}$ to be nonsingular. With this assumption the linearized part $\Delta\mathbf{x}$ of the error $\mathbf{x} - \tilde{\mathbf{x}}$ has the form

$$\Delta\mathbf{x} = -A^{-1}(A*E)\,\mathbf{x}, \qquad \text{where } E = (e_{ij}),\ e_{ij} = e'_{ij} - e''_i .$$

An algorithm $\mathcal{A}$ to solve a linear system by a direct method can be considered as a sequence of transformations which operate on the coefficient matrix and on the right hand sides [Broyden (1977), Stoer, Burlisch (1980)], namely

$$A|\mathbf{b} \longrightarrow A^{(1)}|\mathbf{b}^{(1)} \longrightarrow \dots \longrightarrow A^{(t-1)}|\mathbf{b}^{(t-1)} \longrightarrow I|\mathbf{x}, \quad \text{with } A^{(k)}\mathbf{x} = \mathbf{b}^{(k)}.$$

If the accumulation of local errors at step k produces a perturbation $E^{(k)}$ on $A^{(k)}$ and $\mathbf{b}^{(k)}$, this perturbation will influence the final error with a linear term $\Delta\mathbf{x}^{(k)} = -A^{(k)-1}(A^{(k)}*E^{(k)})\,\mathbf{x}$ and the total linear part of the error will be

$$\Delta\mathbf{x} = \sum_{k=1}^{t} \Delta\mathbf{x}^{(k)} = -\sum_{k}^{t} A^{(k)-1}(A^{(k)}*E^{(k)})\,\mathbf{x}. \qquad (2.1)$$

It readily follows that

$$\underset{\|\mathbf{x}\|_2=1}{\text{mean}}\ \mathbf{E}(\|\Delta\mathbf{x}\|_2^2) = \sum_{k=1}^{t}\ \underset{\|\mathbf{x}\|_2=1}{\text{mean}}\ \mathbf{E}(\|\Delta\mathbf{x}^{(k)}\|_2^2).$$

Let *eps* be the relative computer precision. For t-digit β-base floating point arithmetic with rounding one has *eps* = $\beta^{1-t}/2$ (see [15, p. 6]). When local errors are due to the representation of real numbers in the computer or an arithmetic operation, the quantity *eps* is related with to the mean μ and the variance σ^2 of the resulting errors. More in detail, using a floating point arithmetic with rounding it is common to assume that local representation and roundoff relative errors are independent random variables, uniformely distributed between [- *eps*/2 and *eps*/2], [Liu, Kaneko (1970), Oppenheim (1972)]. Then μ = 0 and σ^2 = *eps*^{2}/12. As noted by Oppenheim (1972), empirical studies have shown that the distribution is not quite uniform, so that σ^2 is proportional to *eps*^{2} with a proportionality constant slightly less than 1/12.

In performing matrix operations the accuracy of the result is limited by the finite precision of the arithmetic. The choice of word length influences both the amount of space required to store the matrices and the time spent in computations. The trivial choice is to use the same word length both to store the matrices and to perform the operations. In this case, usually many digits of the intermediate matrices involved in the computation and of the result are less of significance.

A classical alternative consists in using multiple precision arithmetic to perform the most critical operations (e.g. the accumulation of scalar products) [Brent (1976), Wilkinson (1963)]. Recently some authors proposed a technique which allows computing arithmetic expressions to least significant bit accuracy at the expense of a little computational overhead [Kulish, Miranker (1981, 1986), Rump, Bhoem (1983)]. Both these techniques allow a proper use of the computer storage and a better control of the errors by reducing the number of roundings in the computational processes, moreover on the modern computers, the resulting overhead is not too high.

On the basis of these considerations we will assume in the following that elementary operations on matrices are performed in multiple precision or with maximal accuracy arithmetic so that all the digits in the intermediate matrices representation are accurate. Therefore, the error matrices $E^{(k)}$ will be considered as matrices of independent random variables with mean $\mu = 0$ and variance $\sigma^2 = eps^2/12$.

The following lemma, whose proof is in the appendix A, will be useful to prove the statistical error bounds.

Lemma 2.1. Let A,C be nxn matrices, let $E = (e_{ij})$, $e_{ij} = e'_{ij} - e''_{i}$ with e'_{ij}, e''_{i} independent random variables with mean $\mu = 0$ and variance σ^2, then

$$\underset{\|\mathbf{x}\|_2 = 1}{\text{mean}} \ \mathbf{E}(\| C\,[A*E]\,\mathbf{x}\|_2^2) = \frac{2}{n}\,\sigma^2\,\Big(\sum_{r=1}^{n} \ \sum_{i,j \mid e_{ij}\neq 0} c_{ri}^2\, a_{ij}^2\Big)$$

□

The computed solution $\tilde{\mathbf{x}}$ can be compared with the exact one either by estimating the relative error $\|\mathbf{x} - \tilde{\mathbf{x}}\|_2/\|\mathbf{x}\|_2 \approx \|\Delta\mathbf{x}\|_2/\|\mathbf{x}\|_2$ or the relative residual $\|\mathbf{b} - A\tilde{\mathbf{x}}\|_2/\|\mathbf{x}\|_2 \approx \|A\,\Delta\mathbf{x}\|_2/\|\mathbf{x}\|_2$, these two quantities can behave quite differently, as shown in the following example.

Example 2.1. Let A be the 2x2 matrix $\begin{bmatrix} 2 & 0 \\ 0 & 2/5 \end{bmatrix}$, let $\mathbf{x} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$.

Fig.2 shows the sets $\mathbf{A} = \{x \in R^2 \mid \|x\|_2 = \epsilon\}$, $\mathbf{B} = \{x \in R^2 \mid \|A\,x\|_2 = \epsilon\}$. □

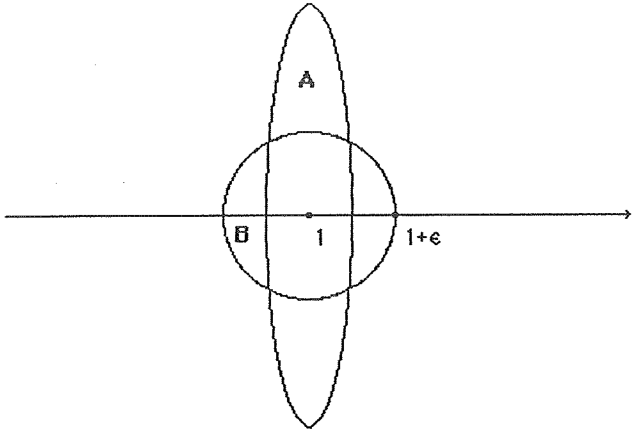

Fig.2. the sets **A** and **B** of example 2.1.

In the classical theory, stability and good-behaviour of an algorithm $\mathcal{A}$ are defined as follows.

Definition 2.1. [Jankowski, Wozniakowski (1977)] Let $\Delta\mathbf{x}$ be the linear part of the algorithmic error after the application of an algorithm $\mathcal{A}$. An algorithm $\mathcal{A}$ is said to be *classically numerically stable* if there exist a constant c_1 independent of A such that

$$\|\Delta\mathbf{x}\|_2 \le c_1\, eps\, k_2(A)\, \|\mathbf{x}\|_2 .$$

□

Definition 2.2. [Jankowski, Wozniakowski (1977)] Let $\Delta\mathbf{x}$ be the linear part of the algorithmic error after the application of an algorithm $\mathcal{A}$. An algorithm $\mathcal{A}$ is said to be *classically well-behaved* if there exist a constant c_2 independent of A such that

$$\|A\Delta x\|_2 \le c_2 \; eps \; \|A\|_2 \, \|x\|_2 \, . \qquad \square$$

It easy to show that classical good-behaviour implies numerical stability, but not vice versa. These definitions suggest to introduce two quantities, namely the *statistical stability factor* and the *statistical good-behaviour factor*, respectively.

$$\mathfrak{s}(A) = \underset{\|x\|_2 = 1}{\text{mean}} \; \mathbf{E}(\|\Delta x\|_2^2), \qquad \text{(statistical stability factor)},$$

$$\mathfrak{w}(A) = \underset{\|x\|_2 = 1}{\text{mean}} \; \mathbf{E}(\|A\,\Delta x\|_2^2), \qquad \text{(statistical good-behaviour factor)}.$$

Proposition 2.2. Taking into account data perturbations only, one has

$$\mathfrak{s}(A) = \frac{2}{n}\,\sigma^2 \, \|\mathbb{A}\|_F^2, \qquad \text{and} \qquad \mathfrak{w}(A) = \frac{2}{n}\,\sigma^2 \, \|A\|_F^2.$$

Proof. The thesis follows from the relations

$$\Delta x = -A^{-1}(A*E)\,x, \qquad \text{and}$$

$$A\Delta x = -A\;A^{-1}(A*E)\,x = -(A*E)\,x,$$

and from Lemma 2.1, with $C = A^{-1}$ and $C = I$. □

Now it is possible to define the statistical stability and good-behaviour of an algorithm.

Definition 2.3. Let Δx be the linear part of the algorithmic error after the application of an algorithm $\mathcal{A}$. Let $\mathfrak{s}(A)$ the corresponding stability factor. The algorithm $\mathcal{A}$ is said to be *numerically stable* if there exist a constant c_3 not depending on A such that $\mathfrak{s}(A) \le c_3 \; eps^2 \, \|\mathbb{A}\|_F^2$ □

Definition 2.4. Let Δx be the linear part of the algorithmic error after the application of an algorithm $\mathcal{A}$. Let $\mathfrak{w}(A)$ the corresponding good-behaviour factor. The algorithm $\mathcal{A}$ is said to be *well-behaved* if there exist a constant c_4 not depending on A such that $\mathfrak{w}(A) \le c_4 \; eps^2 \, \|A\|_F^2$. □

Proposition 2.3. Taking into account algorithmic errors, the stability and good-behaviour factors can be computed by the following relations

$$\mathfrak{s}(A) = \frac{2}{n}\,\sigma^2 \sum_{k=1}^{t} \sum_{r=1}^{n} \sum_{i,j \,|\, e_{ij} \ne 0} {z_{pi}^{(k)}}^2 \, {a_{ij}^{(k)}}^2 \, .$$

$$\mathfrak{w}(A) = \frac{2}{n}\,\sigma^2 \sum_{k=1}^{t} \sum_{r=1}^{n} \sum_{i,j \,|\, e_{ij} \ne 0} \Big(\sum_{p=1}^{n} a_{rp}\, z_{pi}^{(k)}\Big)^2 \, {a_{ij}^{(k)}}^2 \, .$$

Proof. From relation 2.1, we get

$$\Delta x = \sum_{k=1}^{t} \Delta x^{(k)} = -\sum_{k=1}^{t} {A^{(k)}}^{-1} (A^{(k)} * E^{(k)})\, x \, ,$$

$$A\,\Delta x = \sum_{k=1}^{t} A\,\Delta x^{(k)} = -\sum_{k=1}^{t} A\,{A^{(k)}}^{-1} (A^{(k)} * E^{(k)})\, x \, .$$

The thesis follows from lemma 2.1 with $C = A^{(k)-1}$ and $C = AA^{(k)-1}$. □

Corollary 2.1. $\mathfrak{w}(A) \le \|A\|_F^2 \; \mathfrak{s}(A)$. □

3. Influence of diagonal scaling on the error.

Scaling is one of the most commonly used preconditioning techniques [e.g. see Bauer (1963), Skeel (1979, 1981), van der Sluis (1969)]. It consists in multiplying rows and/or columns of the matrix A by suitable factors before solving the system with the algorithm $\mathcal{A}$. In the following, the influence of row scaling on numerical stability and good-behaviour of algorithms is studied. The following preconditioned algorithm can be derived.

Let U be the diagonal positive nxn matrix for which all the rows of the the matrix F = U A have euclidean length equal to 1, then F is row-equilibrated. The system A**x**=**b** can be written UA**x**=U**b** and solved in two steps:

Algorithm $\mathcal{A}$ row-scaled.

compute $\mathbf{z} = U\mathbf{b}$ and $F = U A$;

solve $F\mathbf{x} = \mathbf{z}$ with the algorithm $\mathcal{A}$.

It easy to see that tensorial condition is invariant under row-scaling , then from proposition 1.1 one has:

$$\| \mathbb{A} \|_F = \| \mathbb{F} \|_F = n^{1/2} k_F(F), \qquad (3.1)$$

Some questions naturally arise about the numerical behaviour of the scaling:

i) How the algorithm used to solve the system changes due to the scaling?

ii) How much is the error on the solution of the problem sensitive to the scaling itself?

iii) How numerical stability and well behaviour are affected by the scaling.

Answering to question (i) need the knowledge of the properties of the algorithm $\mathcal{A}$. When the algorithm is influenced by the numerical values of the quantities involved in the computation the algorithm itself changes with the scaling. For example in Gaussian elimination with column or total pivoting the choice of the pivots is influenced by the scaling, conversely QR and LQ algorithms without pivoting are not influenced by the scaling.

We want now to answer question (ii). The use of scaling obviously does not affect the inherent error, if the entries of the diagonal matrices are integers powers of the base arithmetic no roundoff error is introduced, moreover the scaling can used to change the pivoting strategy without affecting the entries involved in the computation [Stoer, Burlisch (1980)], therefore in the following we assume that scaling does not introduce additional roundoff errors.

Moreover the following sufficient conditions for the invariance of the algorithmic error under row scaling can be stated.

Proposition 3.1. Given a diagonal scaling A → UA, if the following conditions are satisfied

a) the intermediate matrices $(A^{(k)}|\mathbf{b}^{(k)})$ are transformed into $(UA^{(k)}|U\,\mathbf{b}^{(k)})$;

b) the statistical distribution of $e'^{(k)}_{ij}$, $e''^{(k)}_{ij}$ does not change;

then the error $\Delta\mathbf{x}$ remains unchanged.

The proof follows by elementary calculus. □

A similar theorem has been proved by Bauer (1963) for Gaussian elimination, this means that the only influence of scaling in Gaussian elimination is on the choice of pivots. It is easy to see that proposition 3.1 holds for LQ algorithm but not for QR.

It is possible to answer question (iii) by showing that row-scaling can improve the numerical stability of the algorithms without affecting the good-behaviour.

Proposition 3.2. Let $\Delta\mathbf{x}$ be the linear part of the algorithmic error after the application of an algorithm $\mathcal{A}$. Let $\varepsilon(A)$ the corresponding stability factor. If there exist a constant c_5 not depending on A such that $\varepsilon(A) \le c_5$ $\mathit{eps}^2\, k_F(A)^2$ then the algorithm $\mathcal{A}$ **row-scaled** is numerically stable.

Proof. Disregarding any roundoff error introduced with the scaling the total error on the solution is the same of the application of $\mathcal{A}$ to the system $F\mathbf{x}=Ub$. Then $\varepsilon(A) \le c_5\ \mathit{eps}^2\, k_F(F)^2$ and using (3.1) one has

$$\varepsilon(A) \le c_5\, n^{-1/2}\ \mathit{eps}^2\ \|\mathbb{A}\|_F . \qquad \square$$

Proposition 3.3. Let $\Delta\mathbf{x}$ be the linear part of the algorithmic error after the application of an algorithm $\mathcal{A}$. Let $w(A)$ the corresponding good-behaviour factor. If the algorithm $\mathcal{A}$ is well-behaved then the algorithm $\mathcal{A}$ **row-scaled** is well-behaved too.

Proof. It easy to see that $\|F\|_F^2 = n$, and $\|U^{-1}\|_2 \le \|U^{-1}\|_F = \|A\|_F$. Then, from the relation $\|A\Delta\mathbf{x}\|_2 \le \|U^{-1}\|_2\ \|F\Delta\mathbf{x}\|_2$ and from the good-behaviour of $\mathcal{A}$ one has

$$w(A) \le \|U^{-1}\|_2^2\, w(F) \le \|A\|_F^2\, c_4\ \mathit{eps}^2\, \|F\|_F^2 \le n\, c_4\ \mathit{eps}^2\, \|A\|_F^2 \qquad \square$$

4. Stability and Good-behaviour in Gaussian elimination.

We assume that the pivoting strategy has been already applied, then the Gaussian elimination algorithm can be considered as a sequence of elementary transformations

$$A^{(0)} = A, \qquad A^{(k)} = M^{(k)} A^{(k-1)}, \quad k = 1,2, \dots ,n-1 \quad \text{with} \quad M^{(k)} = (m_{ij}^{(k)}),$$

$$m_{ij}^{(k)} = \begin{cases} 0 & \text{if } i<j,\ i>j \ne k, \\ 1 & \text{if } i=j, \\ -a_{ij}^{(k-1)}/a_{kk}^{(k-1)} & \text{if } i>j=k. \end{cases}$$

Then $\quad A^{(k)} = M^{(k)} M^{(k-1)} \dots M^{(1)} A \quad$ and

$$A\, A^{(k)^{-1}} = M^{(1)^{-1}} M^{(2)^{-1}} \dots M^{(k)^{-1}} = L^{(k)},$$

the matrix $L^{(k)} = (l_{ij}^{(k)})$ has the following structure:

$$l_{ij}^{(k)} = \begin{cases} 0 & \text{if } i<j,\ i>j>k, \\ 1 & \text{if } i=j, \\ -m_{ij}^{(j)} & \text{if } i>j=k, \end{cases}$$

Moreover, let $E^{(k)} = (e_{ij}^{(k)})$, then $e_{ij}^{(k)} = 0$ if $i \le k$ or $j \le k$., i.e.

$$L^{(k)} = \begin{array}{c} \left[\begin{array}{cccccc} 1 & 0 & 0 & & \dots & 0 \\ * & 1 & 0 & 0 & \dots & 0 \\ * & * & 1 & & \dots & 0 \\ * & * & * & & \dots & 0 \\ * & * & * & 0 & \dots & 1 \end{array}\right] \\ 1 \dots k \quad \dots \qquad n \end{array}, \qquad E^{(k)} = \begin{array}{c} \left[\begin{array}{cccccc} 0 & 0 & & \dots & & 0 \\ 0 & 0 & & \dots & & 0 \\ 0 & 0 & \dots & * & * & * \\ 0 & 0 & \dots & * & * & * \\ 0 & 0 & \dots & * & * & * \end{array}\right] \\ 1 \quad \dots \quad k+1 \dots \quad n \end{array}$$

From $A^{(k)^{-1}} = A^{-1}\ R^{(k)^{-1}}$ we see that the matrix $A^{(k)^{-1}}$ differs from A^{-1} in the first k columns. Hence

$$e(A) = \frac{2}{n}\sigma^2 \sum_{k=1}^{n-1} \sum_{r=1}^{n} \sum_{i=k+1}^{n} \sum_{j=k+1}^{n} z_{ri}^{(k)^2} a_{ij}^{(k)^2} = \frac{2}{n}\sigma^2 \sum_{k=1}^{n-1} \sum_{r=1}^{n} \sum_{i=k+1}^{n} \sum_{j=k+1}^{n} z_{ri}^{2}\ a_{ij}^{(k)^2},$$

and

$$w(A) = \frac{2}{n}\sigma^2 \sum_{k=1}^{n-1} \sum_{r=1}^{n} \sum_{i=k+1}^{n} \sum_{j=k+1}^{n} l_{ri}^{(k)^2} a_{ij}^{(k)^2} = \frac{2}{n}\sigma^2 \sum_{k=1}^{n-1} \sum_{i=k+1}^{n} \sum_{j=k+1}^{n} a_{ij}^{(k)^2} \le$$

$$\le \frac{2}{n}\ \sigma^2 \sum_{k=1}^{n-1} \| A^{(k)} \|_F^2 \quad .$$

These equations exactly relate the quanties $e(A)$ and $w(A)$ to the growth of the elements of the intermediate matrices. Upper bounds for $e(A)$ and $w(A)$ can be derived using the bounds for $|a_{ij}^{(k)}|$ which in turn derive from the used pivoting strategy. It is well known that with suitable pivoting techniques (e.g. column pivoting or total pivoting, [Wilkinson (1961)]) there exists a function g(n) independent of A such that

$$|a_{ij}^{(k)}| \le \alpha\ g(n), \quad i=1,2,\dots,n, \quad j=1,2,\dots,n, \quad k=1,2,\dots,n-1,$$

where $\alpha = \max_{i,j} |a_{ij}|$. Then we have

$$e(A) \le \frac{2}{n}\sigma^2 \sum_{k=1}^{n-1} \sum_{r=1}^{n} \sum_{i=k+1}^{n} z_{ri}^2\,(n-k)\,\alpha^2\, g(n)^2 \le \sigma^2 n^2 g(n)^2\ \ \alpha^2 \|A^{-1}\|_F^2 ,$$

$$w(A) \le \frac{2}{n}\sigma^2 \sum_{k=1}^{n-1} \sum_{i=k+1}^{n} \sum_{j=k+1}^{n} \alpha^2 g(n)^2 = \frac{2}{n}\sigma^2 \alpha^2 g(n)^2 \sum_{k=1}^{n-1} (n-k)^2 \le \sigma^2\ n^2 g(n)^2 \alpha^2.$$

From $\alpha \le \| A \|_F$ it readily follows that

$$e(A) \ \le\ \sigma^2\ c_6(n)\, \| A \|_F^2\ \| A^{-1} \|_F^2, \tag{4.1}$$

$$w(A) \ \le\ \sigma^2\ c_7(n)\, \| A \|_F^2 , \tag{4.2}$$

where $c_6(n)$ and $c_7(n)$ do not depend on A. Therefore Gaussian elimination algorithms with column or total pivoting are well behaved.

The following examples show that Gaussian elimination (even with column pivoting) is not numerically stable and Gaussian elimination without pivoting is not well behaved.

Example 4.1. Let A be the 2x2 matrix $\begin{bmatrix} 1 & \beta \\ 1/\beta & 0 \end{bmatrix}$, $\beta>1$, Gaussian elimination with column pivoting reduces A in upper triangular form, i.e.

$$A^{(1)} = \begin{bmatrix} 1 & \beta \\ 0 & -1 \end{bmatrix}, \qquad A^{(1)-1} = A^{(1)},$$

Hence

$$e(A) = \sigma^2 \sum_{r=1}^{2} \sum_{i=1}^{2} \sum_{j=1}^{2} z_{ri}^{(1)2}\ a_{ij}^{(1)2} =$$

$$= \sigma^2\ [1 + 0 + \beta^2 + \beta^2 + 0 + 0 + 0 + 1] = 2\,\sigma^2\ [1 + \beta^2].$$

On the other hand, $A^{-1} = \begin{bmatrix} 0 & \beta \\ 1/\beta & -1 \end{bmatrix}$, the tensorial condition is

$$\| \mathbb{A} \|_F^2 = 0 + 1 + 0 + 0 + 1/\beta^2 + 1/\beta^2 + 1 + 0 = 2\,[1 + 1/\beta^2],$$

and Gaussian elimination with column pivoting is not numerically stable. □

Example 4.2. Let A be the 2x2 matrix $\begin{bmatrix} 1 & \beta \\ \beta & 0 \end{bmatrix}$, $\beta>1$. Gaussian elimination reduces A in upper triangular form, i.e.

$$A^{(1)} = \begin{bmatrix} 1 & \beta \\ 0 & -\beta^2 \end{bmatrix}, \qquad A^{(1)-1} = \begin{bmatrix} 1 & 1/\beta \\ 0 & -1/\beta^2 \end{bmatrix} ,$$

$$C = AA^{(1)-1} = \begin{bmatrix} 1 & 0 \\ \beta & 1 \end{bmatrix}.$$ Hence

$$w(A) = \sigma^2 \sum_{r=1}^{2} \sum_{i=1}^{2} \sum_{j=1}^{1} c_{ri}^2\, a_{ij}^{(1)2} = \sigma^2 [c_{11}^2 a_{11}^{(1)2} + c_{12}^2 a_{21}^{(1)2} +$$

$$+ c_{12}^2 a_{22}^{(1)2} + c_{21}^2 a_{11}^{(1)2} + c_{22}^2 a_{21}^{(1)2} + c_{22}^2 a_{22}^{(1)2}] =$$

$$= \sigma^2 [1 + 0 + \beta^2 + 0 + \beta^2 + 0 + \beta^4 + \beta^4] = \sigma^2 [2\beta^4 + 2\beta^2 + 1].$$

Then, since $\| A \|_F^2 = 1 + 2\beta^2$, Gaussian elimination without pivoting is not well-behaved. □

In Gaussian elimination the scaling can be used to modify the algorithm by affecting the choice of pivots. E.g. the matrix can be row-equilibrated before applying the algorithm or a weighted pivoting strategy can be used at any step of the elimination process without actually affecting the entries of the matrices [Stoer (1980)]. From proposition 3.2 and 3.3 and relation (4.1) and (4.2) it follows that row-scaled Gaussian elimination is numerically stable and well-behaved.

5. Stability and Good-behaviour of QR algorithm.

The solution of linear systems using Orthogonalization techniques consists in reducing the system to triangular form by multiplying the matrix of coefficients by appropriate orthogonal matrices.

We have $A^{(0)} = A$, $A^{(k)} = P^{(k)} A^{(k-1)}$, $k = 1,2, \dots, n-1$ with $P^{(k)}$ unitary and $A^{(n-1)} = R$ upper triangular. The matrices $P^{(k)}$ can be elementary Householder matrices or a product of plane rotations as in the Givens method [e.g. see Golub, Van Loan (1983)].

Let $Q^{(k)}$ be the product of the elementary matrices of QR algorithm after k steps. One has:

$$A^{(k)} = Q^{(k)} A, \quad A^{(k)^{-1}} = A^{-1} Q^{(k)^T}, \qquad A\, A^{(k)^{-1}} = Q^{(k)^T}.$$

Then

$$\Delta x = - \sum_{k=1}^{n-1} A^{-1} Q^{(k)^T} ([Q^{(k)} A] * E^{(k)})\, x \quad \text{and} \quad A\, \Delta x = - \sum_{k=1}^{n-1} Q^{(k)^T} ([Q^{(k)} A] * E^{(k)})\, x ,$$

By applying Lemma 2.1. we get:

$$e(A) \le \frac{2}{n} \sigma^2 \sum_{k=1}^{n-1} \| A^{-1} Q^{(k)^T} \|_F^2 \, \| Q^{(k)} A \|_F^2 \le 2 \sigma^2 \| A^{-1} \|_F^2 \, \| A \|_F^2,$$

$$w(A) \le \frac{2}{n} \sigma^2 \sum_{k=1}^{n-1} \| Q^{(k)} A \|_F^2 \le 2 \sigma^2 \| A \|_F^2.$$

Therefore, QR algorithm is well-behaved. On the other hand the following example shows that QR algorithm is not numerically stable.

Example 5.1. Let A be the 2x2 matrix $\begin{bmatrix} 1 & 0 \\ 1 & \beta \end{bmatrix}$, $\beta > 1$. QR reduction put A in upper triangular form with an unitary transformation $Q=\begin{bmatrix} c & -c \\ c & c \end{bmatrix}$, with $c^2 = 1/2$. Then

$$A^{(1)} = Q\,A = \begin{bmatrix} 1/c & c\beta \\ 0 & -c\beta \end{bmatrix}, \qquad A^{(1)\,-1} = \begin{bmatrix} c & c \\ 0 & -(c\beta)^{-1} \end{bmatrix},$$

Hence $\quad \mathfrak{e}(A) = \sigma^2 \sum_{r=1}^{2} \sum_{i=1}^{2} \sum_{j=1}^{2} z_{ri}{}^{(1)2}\, a_{ij}{}^{(1)2} =$

$$= \sigma^2\,[1 + 0 + \beta^2/2 + \beta^2/2 + 0 + 0 + 0 + 1] = \sigma^2\ [2 + \beta^2].$$

On the other hand $\quad A^{-1} = \begin{bmatrix} 1 & 0 \\ -1/\beta & 1/\beta \end{bmatrix}$, the tensorial condition is

$\| \mathbb{A} \|_F^2 = 1 + 0 + 0 + 0 + 1/\beta^2 + 1/\beta^2 + 0 + 1 = 2 + 2/\beta^2$, and the QR algorithm is not numerically stable. □

Propositions 3.1 and 3.2 can be applied to show that QR row-scaled is numerically stable and well-behaved.

6. Stability and Good-behaviour of LQ algorithm.

By applying QR algorithm to the transpose of A, a similar decomposition can be derived which reduces A to lower triangular form. This algorithm is denoted as LQ algorithm.

We have $A^{(0)} = A$, $A^{(k)} = A^{(k-1)}P^{(k)}$, $k = 1,2, \dots ,n-1$ with $P^{(k)}$ unitary and $A^{(n-1)} = L$ lower triangular. Let $Q^{(k)}$ be the product of the elementary matrices of LQ algorithm after k steps. One has:

$$A^{(k)} = A\,Q^{(k)}, \qquad A^{(k)^{-1}} = Q^{(k)^T} A^{-1}.$$

Then

$$\Delta x = -\sum_{k=1}^{n-1} Q^{(k)^T} A^{-1}([AQ^{(k)}]*E^{(k)})\,x\,, \qquad A\,\Delta x = -\sum_{k=1}^{n-1} A\,Q^{(k)^T} A^{-1}([AQ^{(k)}]*E^{(k)})\,x\,,$$

Since the transformation matrices do not change the lengths of both the rows of A and the columns of A^{-1}, by applying Lemma 2.1 we get:

$$\mathfrak{e}(A) \le 2\,\sigma^2\, \| \mathbb{A} \|_F^2,$$

and the LQ algorithm is numerically stable.

On the other hand, for $\mathfrak{w}(A)$ we can get only the trivial bound of corollary 2.1. The following example shows that LQ algorithm is not well-behaved.

Example 6.1. Let A be the 2x2 matrix $\begin{bmatrix} 1 & \beta \\ 0 & 1 \end{bmatrix}$. The LQ reduction put A in lower triangular form with an unitary transformation $Q=\begin{bmatrix} c & -s \\ s & c \end{bmatrix}$, $s^2+c^2 = 1$, $\beta = s/c$. Then

$$A^{(1)} = A\,Q = \begin{bmatrix} (1+\beta^2)^{1/2} & 0 \\ s & c \end{bmatrix}, \qquad A^{(1)\,-1} = \begin{bmatrix} (1+\beta^2)^{-1/2} & 0 \\ -\beta(1+\beta^2)^{-1/2} & 1/c \end{bmatrix},$$

$$C = AA^{(1)-1} = \begin{bmatrix} (1-\beta^2)(1+\beta^2)^{-1/2} & \beta/c \\ -\beta(1+\beta^2)^{-1/2} & 1/c \end{bmatrix}.$$ Hence

$$w(A) = \sigma^2 \sum_{r=1}^{2} \sum_{i=1}^{2} \sum_{j=1}^{i} c_{ri}^2\, a_{ij}^{(1)2} = \sigma^2 [c_{11}^2 a_{11}^{(1)2} + c_{12}^2 a_{21}^{(1)2} +$$

$$+ c_{12}^2 a_{22}^{(1)2} + c_{21}^2 a_{11}^{(1)2} + c_{22}^2 a_{21}^{(1)2} + c_{22}^2 a_{22}^{(1)2}] =$$

$$= \sigma^2 [(1-\beta^2)^2 + (\beta s/c)^2 + \beta^2 + \beta^2 + (s/c)^2 + 1] = \sigma^2 [2\beta^4 + \beta^2 + 2].$$

Then, since $\| A \|_F^2 = 2 + \beta^2$ the LQ algorithm is not well-behaved. □

7. Numerical experiments.

The test matrices introduced in section 1 were used to verify the numerical stability of several algorithms. The mean algorithmic error was computed and compared to the Skeel and classical condition numbers; m linear systems were solved with the solution vectors x_j randomly chosen in the unitary ball with uniform distribution. In our experiments m=100.

In order to evaluate the mean algorithmic error, the following quantity was computed and plotted for any algorithm.

$$\Gamma(\mathcal{A},A_\lambda) = \mathit{eps}^{-1} \Big(m^{-1} \sum_{j=1}^{m} \| \eta^{(j)} \|_2^2 / \|x_j\|_2^2 \Big)^{1/2} = \mathit{eps}^{-1} \Big(m^{-1} \sum_{j=1}^{m} \sum_{i=1}^{n} \eta_i^{(j)\,2} \Big)^{1/2},$$

where $\eta^{(j)} = (\eta_i^{(j)})$ is the vector of the errors of the solution of the j-th system, and *eps* is the machine precision related to the word length used to represent the matrices.

In the following graphs the abscissas represent the base 10 logarithm of the Skeel condition number $C_\infty(A_\lambda)$, and the ordinates, in logarithmic scale as well, represent the classical condition number $k_\infty(A_\lambda)$ and the quantity $\Gamma(\mathcal{A},A_\lambda)$ resulting from the application of the algorithm $\mathcal{A}$ to A_λ. Data points are connected by straight lines to evidentiate the behaviour of the matrices in the same class.

In Fig.3 the results concerning Gaussian Elimination and in Fig.4 those concerning Orthogonalization techniques are plotted, it is self evident how this class of test matrices allows to investigate the numerical stability of algorithms.

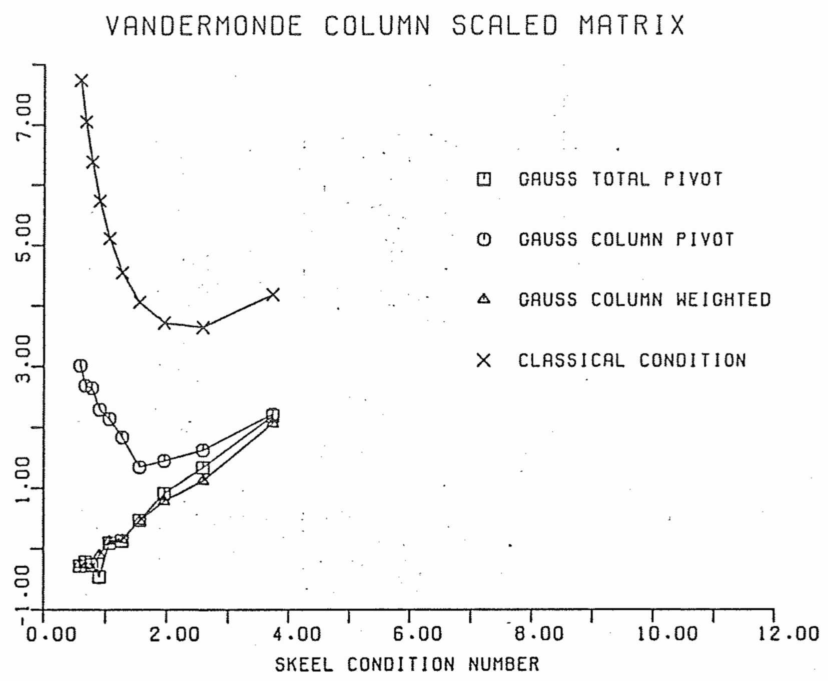

Fig. 3. The classical condition number and the quantities $\Gamma(\mathcal{A},A_\lambda)$ for various Gaussian elimination techniques plotted versus the Skeel condition number for a 5x5 Vandermonde column scaled matrix.

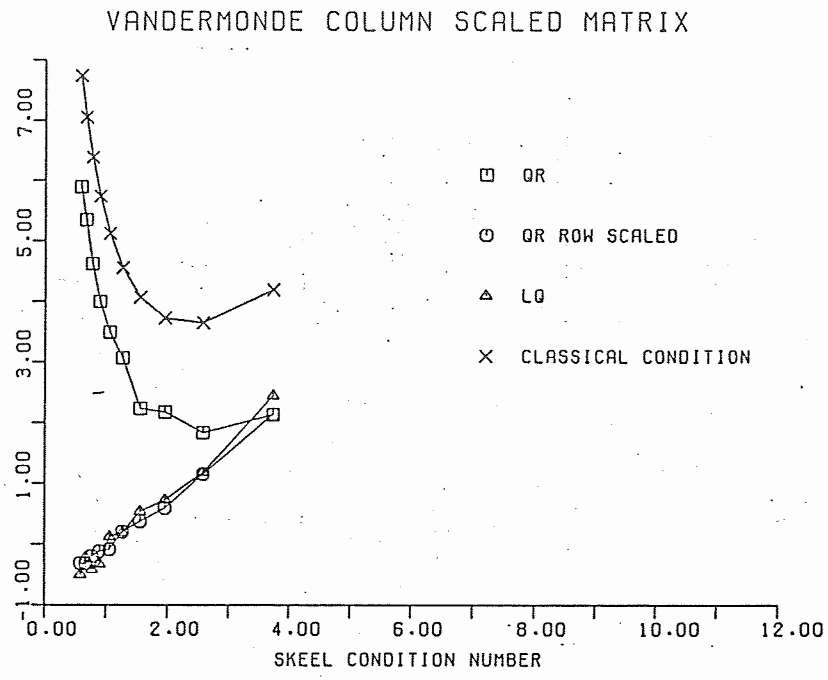

Fig. 4. The classical condition number and the quantities $\Gamma(\mathcal{A},A_\lambda)$ for various orthogonalization techniques plotted versus the Skeel condition number for a 5x5 Vandermonde column scaled matrix.

8. Concluding remarks.

Tables I summarizs the numerical stability and good-behaviour properties proved in this work.

Table I.

Algorithm	numerically stable	well-behaved
Gaussian elimination	no	no
Gaussian elimination with column pivoting	no	yes
Gaussian elimination with total pivoting	?	yes
Row-scaled Gaussian elimination with column or total pivoting	yes	yes
Q R	no	yes
Row-scaled Q R	yes	yes
L Q	yes	no

Appendix A.

In this appendix the proof of lemma 2.1 is given. Two preliminary lemmas are needed.

Lemma A.1 The following equalities hold

$$\Gamma^{-1} \int_{\mathbf{B}_n} x_h \, dx = 0, \qquad h=1,2, \dots, n;$$

$$\Gamma^{-1} \int_{\mathbf{B}_n} x^2_h \, dx = 1/n, \qquad h=1,2, \dots, n;$$

$$\Gamma^{-1} \int_{\mathbf{B}_n} x_h x_k \, dx = \delta_{hk}/n, \qquad h,k=1,2, \dots, n.$$

The proof follows from elementary calculus. □

Lemma A.2. Let $E = (e_{ij})$, $e_{ij} = e'_{ij} - e''_i$ with e'_{ij}, e''_i independent random variables with mean $\mu = 0$ and variance σ^2, then

$$\mathbf{E}(e_{ij} \; e_{pq}) = \sigma^2 \, (\delta_{ip} \, \delta_{jq} + \delta_{ip}) .$$

Proof. We have

$$\mathbf{E}(e_{ij} \; e_{pq}) = \mathbf{E}(e'_{ij} \; e'_{pq}) + \mathbf{E}(e''_i \, e''_p) - \mathbf{E}(e'_{ij} \; e''_p) - \mathbf{E}(e'_{pq} \; e''_i) =$$

$$= \mathbf{E}(e'_{ij} \; e'_{pq}) + \mathbf{E}(e''_i \, e''_p) ,$$

and the thesis follows from the relations

$$\mathbf{E}(e'_{ij} \; e'_{pq}) = \begin{cases} \mathbf{E}(e'_{ij}{}^2) = \mu^2 + \sigma^2 & \text{if } i=p \text{ and } j=q; \\ \mathbf{E}(e'_{ij}) \, \mathbf{E}(e'_{pq}) = \mu^2 & \text{otherwise;} \end{cases}$$

$$\mathbf{E}(e''_i \, e''_p) = \begin{cases} \mathbf{E}(e''_i{}^2) = \mu^2 + \sigma^2 & \text{if } i = p; \\ \mathbf{E}(e''_i) \, \mathbf{E}(e''_p) = \mu^2 & \text{otherwise.} \end{cases}$$

□

Lemma 2.1. Let A,C be nxn matrices, let $E = (e_{ij})$, $e_{ij} = e'_{ij} - e''_i$ with e'_{ij}, e''_i independent random variables with mean $\mu = 0$ and variance σ^2, then

$$\operatorname*{mean}_{\|x\|_2 = 1} \; \mathbf{E}(\| C \, [A*E] \, \mathbf{x} \|_2^2) = \frac{2}{n} \, \sigma^2 \, (\sum_{r=1}^{n} \; \sum_{i,j \,|\, e_{ij} \neq 0} c_{ri}^2 \, a_{ij}^2)$$

Proof. We have

$$\mathbf{E}(\| C \, [A*E] \, \mathbf{x} \|_2^2) = \sum_{r=1}^{n} \, [\sum_{i,j \,|\, e_{ij} \neq 0} c_{ri} \, a_{ij} \, e_{ij} \, x_j \,]^2 =$$

$$= \mathbf{E}(\sum_{r=1}^{n} \; \sum_{i,j \,|\, e_{ij} \neq 0} \; \sum_{p,q \,|\, e_{pq} \neq 0} c_{ri} \, a_{ij} \, e_{ij} \, x_j \;\; c_{rp} \, a_{pq} \, e_{pq} \, x_q) =$$

$$= \sum_{r=1}^{n} \; \sum_{i,j \,|\, e_{ij} \neq 0} \; \sum_{p,q \,|\, e_{pq} \neq 0} c_{ri} \, a_{ij} \; x_j \;\; c_{rp} \, a_{pq} \, x_q \, \mathbf{E}(e_{ij} \, e_{pq}) =$$

$$= \sigma^2 \sum_{r=1}^{n} \; \sum_{i,j \,|\, e_{ij} \neq 0} \; \sum_{p,q \,|\, e_{pq} \neq 0} c_{ri} \, a_{ij} \; x_j \;\; c_{rp} \, a_{pq} \, x_q \, (\delta_{ip} \, \delta_{jq} + \delta_{ip}) =$$

$$= \sigma^2 \sum_{r=1}^{n} \; \sum_{i,j \,|\, e_{ij} \neq 0} \; \sum_{q \,|\, e_{iq} \neq 0} c_{ri}^2 \, a_{ij} \, a_{iq} \, (1 + \delta_{jq}) \, x_j \, x_q \, .$$

Hence

$$\operatorname*{mean}_{\|x\|_2 = 1} \; \mathbf{E}(\|C \, [A*E] \, \mathbf{x} \|_2^2) = \Gamma^{-1} \int_{\mathbf{B}_n} \mathbf{E}(\|C \, [A*E] \, \mathbf{x} \|_2^2) \, d\mathbf{x} =$$

$$= \sigma^2 \sum_{r=1}^{n} \; \sum_{i,j \,|\, e_{ij} \neq 0} \; \sum_{q \,|\, e_{iq} \neq 0} c_{ri}^2 \, a_{ij} \, a_{iq} \, (1 + \delta_{jq}) \;\; \Gamma^{-1} \int_{\mathbf{B}_n} x_j \, x_q \; d\mathbf{x} =$$

$$= \sigma^2 \sum_{r=1}^{n} \; \sum_{i,j \,|\, e_{ij} \neq 0} \; \sum_{q \,|\, e_{iq} \neq 0} c_{ri}^2 \, a_{ij} \, a_{iq} \, (1 + \delta_{jq}) \, \delta_{jq} \, /n \, .$$

and the thesis follows by using the relation $(1 + \delta_{jq}) \, \delta_{jq} = 2 \, \delta_{jq}$. □

REFERENCES

BAUER, F.L. 1963 Optimally Scaled Matrices . *Numer. Math.* 5, 73-87.

BRENT, R.P. 1976 Fast Multiple Precision Evaluation of Elementary Functions . *J. Assoc. Comput. Mach.* **23**, 242-251.

BROYDEN, C.G.1980 Error Analysis, in "D. Jacobs ed. *The State of the art in Numerical Analysis*". London: Academic Press.

FLETCHER, R. 1985 Expected Conditioning. *IMA J. Numer. Anal.* **5**, 247-273.

GOLUB, G.H., VAN LOAN, C.F. 1983 *Matrix Computations.* North Oxford Academic: Oxford.

JANKOWSKI, M., WOZNIAKOWSKI, H. 1977 Iterative Refinement Implies Numerical Stability. *BIT* **17**, 303-311.

KULISH, U.W., MIRANKER W.L. 1986 The Arithmetic of the Digital Computer: A New Approach. *SIAM Review* **28**, 1-40.

LIU, B., KANEKO, T. 1970 Accumulation of Roundoff Errors in Fast Fourier Transforms. *J. Assoc. Comput. Mach.* **17**, 637-654.

OPPENHEIM, A.V., WEINSTEIN, C.J. 1972 Effects of Finite Register Length in Digital Filtering and the Fast Fourier Transform. *Proc. IEEE* **60**, 957-976.

RUMP, S.M., BOEHM, H.: Least Significant Bit Evaluation of Arithmetic Expressions in Single Precision. Computing **30**, 189-199 (1983).

SKEEL, R.D. 1979 Scaling for Numerical Stability in Gaussian Elimination. *J. Assoc. Comput. Mach.* **26**, 494-526.

SKEEL, R.D. 1981 Effects of Equilibration on Residual Size for Partial Pivoting. *SIAM J. Numer. Anal.* **18**, 449-454 .

SLUIS, VAN DER, A. 1969 Condition numbers and equilibration of matrices. *Numer. Math.* **14**, 14-23.

STOER, J., BURLISCH, R. 1980 *Introduction to Numerical Analysis.* New York: Springer.

WILKINSON, J.H. 1961 Error analysis of direct methods of matrix inversion. *J. Assoc. Comput. Mach.* . **8**, 281-330.

WILKINSON, J.H. 1963 *Rounding Errors in Algebraic Processes.* Englewood Cliffs : Prentice Hall.