

General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

GP-B ERROR MODELING AND ANALYSIS
(Annual Report)

(NASA-CR-170985) GP-B ERROR MODELING AND
ANALYSIS Annual Report (Tennessee Univ.)
59 p HC A04/MF A01 CSCI 12A

N84-18960

Unclass

G3/64 18458

Prepared for

National Aeronautics and Space Administration
George C. Marshall Space Flight Center
Marshall Space Flight Center, Alabama 35812

By

The University of Tennessee
Electrical Engineering Department
Knoxville, Tennessee 37996
Principal Investigator: James C. Hung

Under Contract NAS8-34426



10 February 1984

GP-B ERROR MODELING AND ANALYSIS
(Annual Report)

Prepared for

National Aeronautics and Space Administration
George C. Marshall Space Flight Center
Marshall Space Flight Center, Alabama 35812

By

The University of Tennessee
Electrical Engineering Department
Knoxville, Tennessee 37996
Principal Investigator: James C. Hung

Under Contract NAS8-34426

10 February 1984

TABLE OF CONTENTS

CHAPTER	PAGE
I. General Description	1
II. Finite-wordlength Induced Errors in Kalman Filtering Computation	2
III. Combating the Effect of $\frac{1}{f}$ -noise	45
Distribution List	56

CHAPTER I

GENERAL DESCRIPTION

This report contains the results of a continued study on the analysis and modeling for the Gravity Probe B (GP-B) experiment. The study was supported by the National Aeronautics and Space Administration, Marshall Space Flight Center, under contract NA58-34426. This report covers the study effort over a fifteen-month period from 1 September 1982 to 30 November 1983.

The results of two tasks are reported here. The first task is a refinement of a crude result, done in the last year, on the finite-wordlength induced errors in Kalman filtering computation. Errors in the crude result have been corrected, improved derivation steps are taken, and better justifications are given. The second task is to analyze the errors associated with the suppression of the $\frac{1}{f}$ -noise by rolling the spacecraft and then performing a derolling operation by computation. This second task could use a good deal more time for a more thorough study. The result reported here is what has been obtained to date.

CHAPTER II

FINITE-WORDLENGTH INDUCED ERRORS IN KALMAN FILTERING COMPUTATION

1. Introduction

The problem of finite-wordlength effect on digital computations has been investigated extensively during the past twenty years. Finite-wordlength property of a computer requires either rounding or chopping to be used to limit the wordlength of a number. Since most computers use rounding technique, only rounding errors will be considered in the sequel.

There are two approaches to analyze the rounding error, the first approach considers the statistical nature of rounding errors, and treats them as noise generated in the system. This approach has been widely used by those in the field of digital signal processing. In the statistical error analysis one is usually after the ensemble average and standard deviation of the final error based on the estimated characteristics of source errors and their propagation through computation steps. This approach does not seem to be sufficiently reliable for the analysis of GP-B data reduction errors for two reasons. First, GP-B's four experiment gyros represent only a small sample, their combined statistical characteristics may deviate a good deal from those of the population statistics. Thus the use of statistical analysis here may not give a reliable result. Secondly, the GP-B data reduction involves Kalman filtering and other rather

complex computations. The exact statistical nature of rounding error generation by and propagation through these computations is not easy to establish. Therefore a more conservative approach is needed.

The second approach is to establish bounds for the rounding errors involved in computation. This approach provides a very conservative, though rather pessimistic, result for rounding error analysis. This approach has often been used by those doing numerical analysis. Because of the unusual precision required of the GP-B and the expensiveness of the experiment the use of error bound approach provide a much more reliable results for the error analysis. Therefore this approach will be used for ensuing rounding error analysis. Since Kalman filtering is the main activity in GP-B data reduction, the present chapter is devoted to the analysis of rounding error in Kalman filtering computation.

2. Rounding Procedure in Floating Point Representation

Let x be a number

$$x = (\pm .d_1d_2 \dots) \times b^e \quad (1)$$

where b is the base of the number system used and e , an integer, is the exponent. In general the mantissa part of the number may have infinite number of digits for an exact representation, such as for $\sqrt{2}$. The number (1) may also be represented in the form

$$x = u \cdot b^e + v \cdot b^{e-t} \quad (2)$$

where $\frac{1}{b} \leq |u| < 1$, $0 < |v| < 1$, and u contains only t digits.

Examples: Base 10 numbers:

$$(a) 12.3456 = .1234 \times 10^2 + .56 \times 10^{-2}$$

Here $b = 10$, $t = 4$, and $e = 2$

$$(b) -.0123456 = -.1234 \times 10^{-1} + (-.56) \times 10^{-5}$$

Here $b = 10$, $t = 4$, and $e = -1$

The rounding procedure drops off the second term on the right side of (2) by appropriately adjusting the value of the first term. Thus, after rounding, x becomes \hat{x} which has a t -digit mantissa $.d_1d_2 \dots d_t$ and an exponent b^e . The conventional round-off procedure for any number is as follows:

$$\hat{x} = \begin{cases} u \cdot b^e & \text{if } |v| < \frac{1}{2} \\ u \cdot b^e + b^{e-t} & \text{if } v \geq \frac{1}{2} \\ u \cdot b^e - b^{e-t} & \text{if } v \leq -\frac{1}{2} \end{cases} \quad (3)$$

Note that u and v always have the same sign.

Examples: $b = 10$ and $t = 4$

$$(a) x = 765.4567 = .7654 \times 10^3 + .567 \times 10^{-1}$$

Here $v \geq \frac{1}{2}$ and $e = 3$, so

$$\hat{x} = u \cdot b^e + b^{e-t} = .7654 \times 10^3 + 10^{3-4} = .7655 \times 10^3$$

$$(b) x = 123.426 = .1234 \times 10^3 + .26 \times 10^{-1}$$

Here $v < \frac{1}{2}$, so

$$\hat{x} = u \cdot b^e = .1234 \times 10^3$$

$$(c) x = -765.4567 = -.765 \times 10^3 - .567 \times 10^{-1}$$

Here $v < -\frac{1}{2}$, so

$$\hat{x} = u \cdot b^e - b^{e-t} = -.7654 \times 10^3 - 10^{3-4} = -.7655 \times 10^3$$

These results are intuitively obvious. The reason for going through the formulations of Equations (1), (2) and (3) is to pave a way for the subsequent analysis of rounding errors.

3. Rounding Errors in Floating Pointing Representation

The "absolute rounding error" in \hat{x} is defined as

$$|\tilde{x}| = |\hat{x} - x| \geq 0 \quad (4)$$

From (2) and (3), it is clear that

$$|\tilde{x}| \leq \frac{1}{2} b^{e-t} \quad (5)$$

Examining (1) shows that $|u \cdot b^e| \geq b^{e-1}$ because $u \geq b^{-1}$; and

$|x| \geq |u \cdot b^e|$ because the second term, having similar sign, is dropped.

Hence

$$|x| \geq |u \cdot b^e| \geq b^{e-1} \quad (6)$$

Define the "absolute relative rounding error" ϵ as

$$\epsilon = \frac{|\hat{x}-x|}{|x|} = \frac{|\tilde{x}|}{|x|} \quad (7)$$

By (5) and (6), (7) gives

$$\epsilon \leq \frac{1}{2} b^{1-t} = \beta \quad (8)$$

The quantity β is called the "unit rounding error" which represents the absolute bound of rounding error in the floating point representation of a number of base b and having a t -digit mantissa. It is an important parameter in the analysis of rounding errors.

Example: Consider $b = 10$ and $t = 4$

$$\text{Then } \beta = \frac{1}{2} b^{1-t} = \frac{1}{2} 10^{-3}$$

$$\text{Let } x = 767.4567 = .7654 \times 10^3 + .567 \times 10^{-1}$$

$$\text{then } \hat{x} = .7655 \times 10^3$$

$$|\tilde{x}| = |\hat{x}-x| = .0433$$

$$\epsilon = \frac{.0433}{765.4567} = .56568 \times 10^{-4} < \beta$$

For the sake of comparison, the chopping error in floating point representation of a number will be analyzed next.

4. Chopping Error in Floating Point Representation

For a floating point number in the form of (2), a t -digit chopped number is given by

ORIGINAL PAGE IS
OF POOR QUALITY

$$x_c = u \cdot b^e \quad (9)$$

Define the "absolute chopping error" \tilde{x}_c as

$$|\tilde{x}_c| = |x_c - x| = |v| b^{e-t} \quad (10)$$

Since $|v| < 1$

$$|\tilde{x}_c| \leq b^{e-t} \quad (11)$$

Define the "absolute relative chopping error" as

$$\epsilon_c = \frac{|\tilde{x}_c|}{|x|} \quad (12)$$

Clearly,

$$\epsilon_c \leq \frac{b^{e-t}}{b^{e-1}} = b^{1-t} = \beta_c \quad (13)$$

where β_c is called the "unit chopping error." Comparing (13) and (8) shows

$$\beta_c = 2\beta \quad (14)$$

Example: $b = 10$ and $e = 4$

$$\text{Then } \beta_c = 10^{1-4} = 10^{-3}$$

$$\text{Let } x = 765.4567 = .7654 \times 10^3 + .567 \times 10^{-1}$$

$$\text{then } x_c = .7654 \times 10^3$$

$$|\tilde{x}_c| = |x_c - x| = .567 \times 10^{-1}$$

$$\epsilon_c = \frac{.0567}{765.4567} = .74073 \times 10^{-4} < \beta_c$$

5. Rounding Error in Basic Computer Arithmetic Operations

For the convenience of the subsequent analysis, notation for rounded floating point number is defined here in two equivalent forms. Let x be a floating point number. The rounded value of x is denoted by \hat{x} or $fl(x)$.

Let "*" denote any of the four basic arithmetic operations +, -, \times , and /. The computer value of $x*y$ is $fl(x*y)$, which is related to the exact value $x*y$ by

$$fl(x*y) = (x*y)(1+\epsilon) \quad (15)$$

where ϵ is the actual relative rounding error. The absolute relative error in $(x*y)$ is bounded by

$$|\epsilon| = \left| \frac{fl(x*y) - (x*y)}{(x*y)} \right| \leq \beta \quad (16)$$

where β is the unit rounding error.

6. Rounding Error in Composite Computer Arithmetic Operations

Repeated Additions and subtractions consider the sum

$$\begin{aligned} s &= x_1 + x_2 + x_3 + x_4 \\ &= ((x_1+x_2) + x_3) + x_4 \end{aligned}$$

The rounded value is

$$\begin{aligned} \hat{s} &= \{[(x_1+x_2)(1+\epsilon_1) + x_3] (1+\epsilon_2) + x_4\} (1+\epsilon_3) \\ &= (x_1+x_2)(1+\epsilon_1)(1+\epsilon_2)(1+\epsilon_3) + x_3(1+\epsilon_2)(1+\epsilon_3) + x_4(1+\epsilon_3) \end{aligned}$$

$$\approx (x_1+x_2)(1+\epsilon_1+\epsilon_2+\epsilon_3) + x_3(1+\epsilon_2+\epsilon_3) + x_4(1+\epsilon_3)$$

The rounding error is

$$\begin{aligned}\tilde{s} &= (x_1+x_2)(\epsilon_1+\epsilon_2+\epsilon_3) + x_3(\epsilon_2+\epsilon_3) + x_4(\epsilon_3) \\ &= (x_1+x_2+x_3+x_4)(\epsilon_1+\epsilon_2+\epsilon_3) - x_3\epsilon_1 - x_4(\epsilon_1+\epsilon_2)\end{aligned}$$

The absolute relative rounding error is bounded by

$$\epsilon = \left| \frac{\tilde{s}}{s} \right| \leq 3\beta + \beta \left| \frac{x_3}{s} \right| + 2\beta \left| \frac{x_4}{s} \right| \leq 3\beta + 3\beta \left| \frac{x_j}{s} \right|_{\max}$$

where $|x_j|_{\max}$ is the largest of all $|x_j|$. In general, for a sum of n terms

$$s = \sum_{j=1}^n x_j \quad (17)$$

the absolute relative error is bounded by

$$\epsilon = \left| \frac{\tilde{s}}{s} \right| \leq (n-1)\beta + \sum_{j=3}^n (j-2)\beta \left| \frac{x_j}{s} \right| \leq (n-1)\beta + \frac{(n-1)(n-2)}{2} \left| \frac{x_j}{s} \right|_{\max} \beta$$

Repeated Multiplication and Division. Consider the following combination of product and quotient

$$Q = \frac{x_1 x_2}{y_1} = (x_1 x_2) / y_1$$

the rounded value is

$$\hat{Q} = \frac{x_1 x_2 (1+\epsilon_1)}{y_1} (1+\eta_1) \approx \frac{x_1 x_2}{y_1} (1+\epsilon_1+\eta_1)$$

where ϵ_1 and η_1 are relative rounding errors due to multiplication and division, respectively. The rounding error and the absolute relative rounding error are, respectively,

$$\tilde{Q} = \frac{x_1 x_2}{y_1} (\epsilon_1 + \eta_1)$$

and

$$\epsilon = \left| \frac{\tilde{Q}}{Q} \right| = |\epsilon_1 + \eta_1| \leq 2\beta$$

For the general case

$$Q = \frac{x_1 x_2 \dots x_n}{y_1 y_2 \dots y_m} \quad (19)$$

The absolute relative rounding error is bounded by

$$\epsilon = \left| \frac{\tilde{Q}}{Q} \right| \leq (n+m+1)\beta \quad (20)$$

7. Norms of Vectors and Matrices

Norms of vectors and matrices are useful in the analysis of rounding errors in matrix operations. The following definitions of norm will be adopted in this study.

For an n -vector \underline{x} with elements x_j , define the vector norm as

$$\|\underline{x}\| = \max_j |x_j| \quad (21)$$

Clearly, the norm has the following properties:

- (i) $\|\underline{x}\| \geq 0$
- (ii) $\|\underline{x}\| = 0$ only if $\underline{x} = 0$
- (iii) $\|\underline{x} + \underline{y}\| \leq \|\underline{x}\| + \|\underline{y}\|$
- (iv) $\|a \underline{x}\| = |a| \cdot \|\underline{x}\|$ for any real a .

For a $m \times n$ matrix A with elements a_{ij} , define the matrix norm as

$$||A|| = \text{Max}_j \sum_{i=1}^m |a_{ij}| \quad (22)$$

This norm has the following properties:

- (i) $||A|| \geq 0$
- (ii) $||A|| = 0$ only if $A = 0$
- (iii) $||A+B|| \leq ||A|| + ||B||$
- (iv) $||a A|| = |a| \cdot ||A||$ for any real a
- (v) $||AB|| \leq ||A|| \cdot ||B||$

8. Rounding Error in Matrix Addition

Let A and B be two $m \times n$ matrices, the rounded sum of them is

$$fl[A+B] = A+B+R \quad (23)$$

where R is the rounding error matrix. By the definition of matrix norm (22) and in view of (15), the norm of the rounding error matrix is bounded by

$$||R|| \leq \beta ||A+B|| \quad (24)$$

where $\beta = \frac{1}{\epsilon} b^{1-t}$ as given by (8). The relative norm of $||R||$ is bounded by

$$\epsilon = \frac{||R||}{||A+B||} \leq \beta \quad (25)$$

which is the same as the relative rounding error of the sum of two numbers as shown in (16).

9. Rounding Error in Matrix Multiplication

Since elements of a matrix product are inner products of vector pairs, the rounding error associated with an inner product will be analyzed first. The result will then be used to analyze the rounding error in a matrix product.

9.1. Rounding Error in Inner Product

Consider the inner product of two 3-vectors \underline{a} and \underline{b}

$$I = \underline{a}^T \underline{b} = a_1 b_1 + a_2 b_2 + a_3 b_3$$

The rounded value of $\underline{a}^T \underline{b}$ is

$$\begin{aligned} I &= fl[\underline{a}^T \underline{b}] \\ &= \{[a_1 b_1 (1 + \epsilon_1) + a_2 b_2 (1 + \epsilon_2)](1 + \epsilon_3) + a_3 b_3 (1 + \epsilon_4)\}(1 + \epsilon_5) \\ &\approx a_1 b_1 (1 + \epsilon_1 + \epsilon_3 + \epsilon_5) + a_2 b_2 (1 + \epsilon_2 + \epsilon_3 + \epsilon_5) + a_3 b_3 (1 + \epsilon_4 + \epsilon_5) \end{aligned}$$

where ϵ_j 's are relative rounding errors associated with basic arithmetic operations. The rounding error in I is

$$\tilde{I} = a_1 b_1 (\epsilon_1 + \epsilon_3 + \epsilon_5) + a_2 b_2 (\epsilon_2 + \epsilon_3 + \epsilon_5) + a_3 b_3 (\epsilon_4 + \epsilon_5)$$

The absolute value of this rounding error is bounded by

$$|\tilde{I}| \leq 3\beta |a_1 b_1| + 3\beta |a_2 b_2| + 2\beta |a_3 b_3|$$

In general, the absolute rounding error of the inner product of two n -vectors is bounded by

$$|\tilde{I}| \leq \beta \{n |a_1 b_1| + \sum_{j=2}^n (n+2-j) |a_j b_j|\} \quad (26)$$

The expression for the absolute relative rounding error for an inner product appears cumbersome and is not given here.

$$||\tilde{C}|| \leq n\beta ||A|| \cdot ||B|| = \frac{n}{2} b^{1-t} ||A|| \cdot ||B|| \quad (28)$$

Generalize the above result to a product of N matrices

$$P = M_1 M_2 - - - M_N \quad (29)$$

with interface dimensions $d_1, d_2, - - - d_{N-1}$. Let

$$P_i = M_1 M_2 - - - M_i$$

Then the result of (28) implies the following rounded matrices, with ϵ being the worst error.

$$\hat{P}_2 = fl[M_1 M_2] = M_1 M_2 (1 + d_1 \epsilon)$$

and

$$\begin{aligned} \hat{P}_3 &= fl[\hat{P}_2 M_3] = \hat{P}_2 M_3 (1 + d_2 \epsilon) \\ &= M_1 M_2 M_3 (1 + d_1 \epsilon) (1 + d_2 \epsilon) \approx M_1 M_2 M_3 [1 + (d_1 + d_2) \epsilon] \end{aligned}$$

and

$$\hat{P}_N \approx M_1 M_2 - - - M_N [1 + (d_1 + - - - + d_{N-1}) \epsilon] \quad (30)$$

Rounding errors in P_N is

$$\hat{P}_N = M_1 M_2 - - - M_N (d_1 + d_2 + - - - + d_{N-1}) \epsilon \quad (31)$$

The norm of this error matrix is therefore

$$||\tilde{P}_N|| \leq \beta \left(\sum_{i=1}^{N-1} d_i \right) \prod_{j=1}^N ||M_j|| \quad (32)$$

The results of (24) and (31) can be used jointly to handle the matrix equation containing both products and sums. This will be demonstrated by the following two examples, assuming the worst error ϵ at every computation.

Example 1 Compute

$$R = ABC + D$$

where all matrices are $n \times n$. The rounded R is

$$\hat{R} = ABC(1+2n\epsilon)(1+\epsilon) + D(1+\epsilon)$$

The rounding error of R is

$$\tilde{R} \approx [(2n+1)ABC + D]\epsilon$$

and its norm is bounded by

$$\|\tilde{R}\| \leq \beta[(2n+1)\|ABC\| + \|D\|]$$

Example 2 Compute

$$R = ABC + D$$

where A is $n \times m$, B is $m \times r$, C is $r \times s$, and D is $n \times s$. Then

$$\hat{R} = ABC[1+(m+r)\epsilon](1+\epsilon) + D(1+\epsilon)$$

$$\tilde{R} \approx ABC(1+m+r)\epsilon + D\epsilon$$

$$\|\tilde{R}\| \leq \beta[(1+m+r)\|ABC\| + \|D\|]$$

These two examples show that the rounding error norm of matrix addition does not involve the dimension of the matrices, but that of matrix product involves all the interface dimensions.

10. Rounding Error in Matrix Inversion, First Approach

Let A be a nonsingular $n \times n$ matrix, its inverse A^{-1} satisfies the relationship

$$A A^{-1} = I, \text{ the identity matrix}$$

Let \underline{u}_j be the j th column vector of I and \underline{h}_j be the column of A^{-1} .

Then \underline{h}_j is the solution of

$$Ax = \underline{u}_j \quad j = 1 \text{ to } n \quad (33)$$

Thus A^{-1} can be obtained by solving (33) n times using different \underline{u}_j each time. The solution is usually done by a method based on the Gaussian elimination with partial pivoting. The present concern is the rounding error associated with the computation of A^{-1} . The analysis will be done in two steps: First, find error in A^{-1} computed from the exact A . Second, find error in A^{-1} computed from $A' = A + \Delta A$ where ΔA is the error in A .

Rounding Error in A^{-1} when A is exact. Let \hat{h}_j be computer solution of (33). Define the "residue" associated with \underline{h}_j as

$$\underline{r}_j = A \hat{h}_j - \underline{u}_j \quad (34)$$

The error in \hat{h}_j is

$$\tilde{h}_j = \hat{h}_j - h_j = A^{-1} \underline{r}_j \quad (35)$$

The rounding error matrix for the computer inverse of A is

$$E = [\tilde{h}_1 \quad \tilde{h}_2 \quad - \quad - \quad \tilde{h}_n] \quad (36)$$

Define the "residue matrix" for the computer inverse

$$R = [\underline{r}_1 \quad \underline{r}_2 \quad - \quad - \quad \underline{r}_n] \quad (37)$$

Then

$$E = A^{-1}R \quad (38)$$

The norm of E is bounded by

$$\|E\| \leq \|A^{-1}\| \cdot \|R\| \quad (39)$$

and the relative norm of E is bounded by

$$\epsilon \leq ||R|| \quad (40)$$

Rounding Error in A^{-1} when $A+\Delta A$ is inverted. Let A be erred to $A+\Delta A$, then the computer solution \hat{h}_j for

$$[A+\Delta A]\underline{x} = \underline{u}_j \quad j = 1 \text{ to } n \quad (41)$$

must satisfy

$$[A+\Delta A]\hat{h}_j = \underline{u}_j + \underline{r}_j \quad j = 1 \text{ to } n$$

where \underline{r}_j is the residue. Then

$$\hat{h}_j + A^{-1}\Delta A\hat{h}_j = A^{-1}\underline{u}_j + A^{-1}\underline{r}_j = \underline{h}_j + A^{-1}\underline{r}_j$$

The error in \hat{h}_j is

$$\tilde{h}_j = \hat{h}_j - \underline{h}_j = A^{-1}[\underline{r}_j - \Delta A \hat{h}_j] \quad (42)$$

Using the notation defined in (36) and (37), (42) gives the error matrix

$$E = A^{-1}R - A^{-1}\Delta A f_2[A^{-1}] \approx A^{-1}R - A^{-1}\Delta A A^{-1}$$

The norm of the error matrix is bounded by

$$||E|| \leq ||A^{-1}|| \cdot ||R|| + ||A^{-1}||^2 \cdot ||\Delta A|| \quad (43)$$

the relative error norm is

$$\epsilon = \frac{||E||}{||A^{-1}||} = ||R|| + ||A^{-1}|| \cdot ||\Delta A|| \quad (44)$$

Comparing (44) to (40) shows that the latter is a special case of the former where $\Delta A = 0$. Eq. (44) appears elegant, but its practical

usefulness is in doubt. The problem is that the residue matrix R cannot easily be obtained. In addition, both (43) and (44) do not explicitly depend on any wordlength related parameter, such as, the unit rounding error β .

11. Rounding Error in Matrix Inversion, Second Approach

The usual method of matrix inversion by a computer is based on a repeated use of Gaussian elimination procedure. The procedure consists of two parts, namely, triangularization of a matrix and back substitution. The rounding error for each part will be analyzed first, followed by the analyze of the resultant error. In the following analysis ϵ will denote the worst value of any rounding error ϵ_i . Thus $|\epsilon| \leq \beta$.

11.1. Rounding Error in Matrix Triangularization

Consider a 3x3 Matrix Equation

$$\underbrace{\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}}_{\text{A matrix}} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \quad (45)$$

Let $a_{ij}(0) = a_{ij}$ and $b_i(0) = b_i$ for $i, j = 1$ to n . The first step is to condition the first column. Let

$$m_{21} = -\frac{a_{21}}{a_{11}} = -\frac{a_{21}(0)}{a_{11}(0)} \quad (46)$$

then let

$$a_{22}(1) = a_{22}(0) + m_{21}a_{12}(0) = a_{22}(0) - \frac{a_{21}(0)}{a_{11}(0)} a_{12}(0)$$

$$\hat{a}_{22}(1) = f_{\epsilon} \left[a_{22}(0) - \frac{a_{21}(0)}{a_{11}(0)} a_{12}(0) \right]$$

$$= a_{22}(0)(1+\epsilon) - \frac{a_{21}(0)}{a_{11}(0)} a_{12}(0) (1+3\epsilon)$$

$$= \left[a_{22}(0) - \frac{a_{21}(0)}{a_{11}(0)} a_{12}(0) \right] + a_{22}(0)\epsilon - \frac{a_{21}(0)}{a_{11}(0)} a_{12}(0)(3\epsilon)$$

$$= a_{22}(1) + a_{22}(1)(3\epsilon) - a_{22}(0)(2\epsilon)$$

Similarly

$$\hat{a}_{23}(1) = a_{23}(1) + a_{23}(1) (3\epsilon) - a_{23}(0) (2\epsilon)$$

$$\hat{a}_{32}(1) = a_{32}(1) + a_{32}(1) (3\epsilon) - a_{32}(0) (2\epsilon)$$

$$\hat{a}_{33}(1) = a_{33}(1) + a_{33}(1) (3\epsilon) - a_{33}(0) (2\epsilon)$$

After the first step, the error in the new A, designated $\hat{A}(1)$, is given by $\tilde{A}(1)$ whose elements are

$$\tilde{a}_{ij}(1) = \begin{cases} 0 & i=1; j=1,2,3 \\ 0 & j=1; i=2,3 \\ a_{ij}(1) (3\epsilon) - a_{ij}(0) (2\epsilon) & i,j=2,3 \end{cases} \quad (47)$$

Define

$$\hat{A}_1(0) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & a_{22}(0) & a_{23}(0) \\ 0 & a_{32}(0) & a_{33}(0) \end{bmatrix}$$

and

$$A_1(1) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & a_{22}(1) & a_{23}(1) \\ 0 & a_{32}(1) & a_{33}(1) \end{bmatrix}$$

Then

$$\begin{aligned} \tilde{A}(1) = [\tilde{a}_{ij}(1)] &= 3\epsilon A_1(1) - 2\epsilon \hat{A}_1(0) \\ &\approx \epsilon [3\hat{A}_1(1) - 2\hat{A}_1(0)] \end{aligned} \quad (48)$$

Expressing \tilde{A} in terms of \hat{A} rather than A is important since \hat{A} is available from the computer but not A .

For the b_i coefficients we have

$$b_i(1) = b_i(0) - m_{i1} b_i(0) = b_i(0) - \frac{a_{i1}(0)}{a_{11}(0)} b_i(0)$$

Then

$$\begin{aligned} \hat{b}_i(1) = f_2[b_i(1)] &= b_i(0)(1+\epsilon) - \frac{a_{i1}(0)}{a_{11}(0)} b_i(0)(1+3\epsilon) \\ &= b_i(1) + \epsilon b_i(0) - 3\epsilon \frac{a_{i1}(0)}{a_{11}(0)} b_i(0) \\ &= b_i(1) + 3\epsilon b_i(1) - 2\epsilon b_i(0) \end{aligned}$$

Define

$$\hat{\underline{b}}_1(0) = \begin{bmatrix} 0 \\ b_2(0) \\ b_3(0) \end{bmatrix} \quad \text{and} \quad \underline{b}_1(1) = \begin{bmatrix} 0 \\ b_2(1) \\ b_3(1) \end{bmatrix}$$

then

$$\tilde{\underline{b}}(1) = 3\epsilon \underline{b}_1(1) - 2\epsilon \hat{\underline{b}}_1(0) \approx \beta[3\hat{\underline{b}}_1(1) - 2\hat{\underline{b}}_1(0)] \quad (49)$$

Again, expressing \underline{b} in terms of $\hat{\underline{b}}$ rather than \underline{b} is important since $\hat{\underline{b}}$ is available from the computer but \underline{b} is not.

After the first reduction step, one has

$$\begin{bmatrix} a_{11}(0) & a_{12}(0) & a_{13}(0) \\ 0 & \hat{a}_{22}(1) & \hat{a}_{23}(1) \\ 0 & \hat{a}_{32}(1) & \hat{a}_{33}(1) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1(0) \\ \hat{b}_2(1) \\ \hat{b}_3(1) \end{bmatrix}$$

where \hat{a}_{ij} and \hat{b}_j are rounded quantities. Their errors will be compounded to the new rounded quantities in the next step of the reduction process.

The second step of the reduction concerns the second column of the matrix. Let

$$m_{32} = -\frac{\hat{a}_{32}(1)}{\hat{a}_{22}(1)} \quad (50)$$

and let

$$a_{33}(2) = \hat{a}_{33}(1) + m_{32} \hat{a}_{23}(1) = \hat{a}_{33}(1) - \frac{\hat{a}_{32}(1)}{\hat{a}_{22}(1)} \hat{a}_{23}(1)$$

The rounded value is

$$\begin{aligned}\hat{a}_{33}(2) &= fl[a_{33}(2)] = \hat{a}_{33}(1)(1+\epsilon) - \frac{\hat{a}_{32}(1)}{\hat{a}_{22}(1)} \hat{a}_{23}(1)(1+3\epsilon) \\ &= a_{33}(2) + 3\epsilon a_{33}(2) - 2\epsilon \hat{a}_{33}(1)\end{aligned}$$

Let

$$\hat{A}_2(1) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \hat{a}_{33}(1) \end{bmatrix}$$

and

$$A_2(2) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & a_{33}(2) \end{bmatrix}$$

Then

$$\tilde{A}(2) = 3\epsilon A_2(2) - 2\epsilon \hat{A}_2(1) = \epsilon [3\hat{A}_2(2) - 2\hat{A}_2(1)] \quad (51)$$

For the b_i coefficients in the second reduction step,

$$b_3(2) = \hat{b}_3(1) + m_{32} \hat{b}_2(1) = \hat{b}_3(1) - \frac{\hat{a}_{32}(1)}{\hat{a}_{22}(1)} \hat{b}_2(1)$$

The rounded value is

$$\begin{aligned}\hat{b}_3(2) &= fl[b_3(2)] = \hat{b}_3(1)(1+\epsilon) - \frac{\hat{a}_{32}(1)}{\hat{a}_{22}(1)} \hat{b}_2(1)(1+3\epsilon) \\ &= \hat{b}_3(2) + 3\epsilon b_3(2) - 2\epsilon \hat{b}_3(1)\end{aligned}$$

Let

$$\hat{\underline{b}}_2(1) = \begin{bmatrix} 0 \\ 0 \\ \hat{b}_3(1) \end{bmatrix} \quad \text{and} \quad \underline{b}_2(2) = \begin{bmatrix} 0 \\ 0 \\ b_3(2) \end{bmatrix}$$

Then

$$\tilde{\underline{b}}(2) = [3 \underline{b}_2(2) - 2 \hat{\underline{b}}_2(2)] \epsilon = \beta [3 \hat{\underline{b}}_2(2) - 2 \hat{\underline{b}}_2(1)] \quad (52)$$

The resultant errors,

$$\begin{aligned} \tilde{\underline{A}} &= \tilde{\underline{A}}(1) + \tilde{\underline{A}}(2) = \beta [3 \hat{\underline{A}}_1(1) - 2 \hat{\underline{A}}_1(0)] + \beta [3 \hat{\underline{A}}_2(2) - 2 \hat{\underline{A}}_2(1)] \\ &= \left\{ \sum_{i=1}^2 [3 \hat{\underline{A}}_i(i) - 2 \hat{\underline{A}}_i(i-1)] \right\} \beta \end{aligned} \quad (53)$$

$$\begin{aligned} \tilde{\underline{b}} &= \tilde{\underline{b}}(1) + \tilde{\underline{b}}(2) = \beta [3 \hat{\underline{b}}_1(1) - 2 \hat{\underline{b}}_1(0)] + \beta [3 \hat{\underline{b}}_2(2) - 2 \hat{\underline{b}}_2(1)] \\ &= \beta \left\{ \sum_{i=1}^2 [3 \hat{\underline{b}}_i(i) - 2 \hat{\underline{b}}_i(i-1)] \right\} \end{aligned} \quad (54)$$

Generalization to an nxn matrix A

$$\underline{A} = \underline{A}(0) = \hat{\underline{A}}(0) = \begin{bmatrix} a_{11} & - & - & - & a_{1n} \\ | & & & & | \\ | & & & & | \\ a_{n1} & - & - & - & a_{nn} \end{bmatrix} \quad (55)$$

$$\hat{a}_{ij}(0) = a_{ij}(0) = a_{ij} \quad (56)$$

The matrix obtained after the kth reduction step is

$$A(k) = \begin{bmatrix} a_{11}(k) & - & - & a_{1n}(k) \\ | & & & | \\ | & & & | \\ a_{n1}(k) & - & - & a_{nn}(k) \end{bmatrix} \quad (57)$$

$$A_j(k) = \left[\begin{array}{c|c} \text{○} & \text{○} \\ \hline \text{○} & \text{X} \end{array} \right] \quad (58)$$

The $(n-i) \times (n-i)$ lower right
diagonal block matrix from $A(k)$

The resultant reduction or triangularization errors in A and \underline{b} are, respectively,

$$\tilde{A} = \beta \left\{ \sum_{i=1}^{n-1} [3\hat{A}_i(i) - 2\hat{A}_i(i-1)] \right\} \quad (59)$$

$$\tilde{\underline{b}} = \beta \left\{ \sum_{i=1}^{n-1} [3\hat{\underline{b}}_i(i) - 2\hat{\underline{b}}_i(i-1)] \right\} \quad (60)$$

Finally, the norms of errors due to triangularization are given by

$$\|\tilde{A}\| = \beta \left\{ \sum_{i=1}^{n-1} [3\|\hat{A}_k(k)\| + 2\|\hat{A}_k(k-1)\|] \right\} \quad (61)$$

$$\|\tilde{\underline{b}}\| = \beta \left\{ \sum_{k=1}^{n-1} [3\|\hat{\underline{b}}_k(k)\| + 2\|\hat{\underline{b}}_k(k-1)\|] \right\} \quad (62)$$

Note that after the (n-1)th reduction step, the original matrix A has been reduced to an upper triangular form. Denote it by $A_T = A(n-1)$. Thus,

$$\hat{A}_T = \begin{bmatrix} \hat{a}_{11}(0) & \hat{a}_{12}(0) & \dots & \hat{a}_{1k}(0) & \dots & \hat{a}_{1n}(0) \\ & \hat{a}_{22}(1) & \dots & \hat{a}_{2k}(0) & \dots & \hat{a}_{2n}(0) \\ & & \dots & \vdots & \dots & \vdots \\ & & & \hat{a}_{kk}(k-1) & \dots & \hat{a}_{kn}(0) \\ & & & & \dots & \vdots \\ & & & & & \hat{a}_{nn}(n-1) \end{bmatrix} \quad (63)$$

Similarly, the original \underline{b} vector has been reduced to $\hat{\underline{b}}_T$ given by

$$\hat{\underline{b}}_T = \hat{\underline{b}}(n-1) = [\hat{b}_1(0) \hat{b}_2(1) \dots \hat{b}_n(n-1)]^T \quad (64)$$

11.2 Rounding Error in Back Substitution

This problem is approached as follows: Consider the equation

$$A \underline{x} = \underline{b} \quad (65)$$

where A is an nxn upper triangular matrix. Let $\hat{\underline{x}}$ be the solution of this equation which contains rounding errors, then find \hat{A} and $\hat{\underline{b}}$ such that

$$\hat{A} \hat{\underline{x}} = \hat{\underline{b}} \quad (66)$$

has error-free solution $\hat{\underline{x}}$. Thus the rounding error problem has been transformed to an error problem caused by perturbations in A and \underline{b} .

Define

$$\Delta A = \hat{A} - A, \quad \Delta \underline{x} = \hat{\underline{x}} - \underline{x}, \quad \Delta \underline{b} = \hat{\underline{b}} - \underline{b} \quad (67)$$

Then (65) gives

$$(\hat{A} - \Delta A) \hat{x} - \Delta x = \hat{b} - \Delta b$$

$$\Delta x = \hat{A}^{-1} [\Delta b - \Delta A \hat{x}] \quad (68)$$

In (68), Δx is the rounding error in \hat{x} , \hat{A}^{-1} and \hat{x} are computed by the computer while solving (65), and Δb and ΔA are computed from formulas to be developed. Since A is an upper triangular matrix, the solution of (65) involves only the back substitution operations. Rounding error due to back substitution will now be analyzed.

3x3 Triangular Matrix Equation. Consider the equation

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22} & a_{23} \\ 0 & 0 & a_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \quad (69)$$

The equivalent perturbed equation for evaluating rounding errors is

$$\begin{bmatrix} \hat{a}_{11} & \hat{a}_{12} & \hat{a}_{13} \\ 0 & \hat{a}_{22} & \hat{a}_{23} \\ 0 & 0 & \hat{a}_{33} \end{bmatrix} \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \end{bmatrix} = \begin{bmatrix} \hat{b}_1 \\ \hat{b}_2 \\ \hat{b}_3 \end{bmatrix} \quad (70)$$

Write (69) as

$$\left. \begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 &= b_1 & (a) \\ a_{22}x_2 + a_{23}x_3 &= b_2 & (b) \\ a_{33}x_3 &= b_3 & (c) \end{aligned} \right\} \quad (71)$$

From (71c)

$$x_3 = \frac{b_3}{a_{33}}$$

The rounded value is

$$\hat{x}_3 = \frac{b_3}{a_{33}} (1+\epsilon) = x_3(1+\epsilon) \quad (72)$$

giving

$$x_3 = \frac{\hat{x}_3}{1+\epsilon} \quad (73)$$

Substituting into (71c) and rearranging terms, give

$$\underbrace{a_{33}}_{\hat{a}_{33}} \hat{x}_3 = \underbrace{b_3}_{\hat{b}_3} (1+\epsilon) \quad (74)$$

where \hat{a}_{33} and \hat{b}_3 are also defined. Next, from (71b)

$$x_2 = \frac{1}{a_{22}} [b_2 - a_{23}x_3]$$

Its rounded value is

$$\begin{aligned} \hat{x}_2 &= \frac{1}{a_{22}} [b_2 - a_{23}\hat{x}_3(1+\epsilon)] (1+2\epsilon) \\ &= \frac{1}{a_{22}} [b_2 - a_{23}x_3(1+2\epsilon)] (1+2\epsilon) \\ &= x_2 + \frac{2\epsilon b_2 - 4\epsilon a_{23}x_3}{a_{22}} \end{aligned} \quad (75)$$

giving

$$\begin{aligned} x_2 &= \hat{x}_2 - \frac{2\epsilon b_2 - 4\epsilon a_{23}x_3}{a_{22}} \\ &= \hat{x}_2 - \frac{2\epsilon b_2 - 4\epsilon a_{23}\hat{x}_3/(1+\epsilon)}{a_{22}} \end{aligned} \quad (76)$$

Substituting (76) and (73) into (71b) and rearranging terms, give

$$\underbrace{a_{22}\hat{x}_2}_{\hat{a}_{22}} + \underbrace{a_{23}(1+3\epsilon)x_3}_{\hat{a}_{23}} = \underbrace{b_2(1+2\epsilon)}_{\hat{b}_2} \quad (77)$$

where \hat{a}_{22} , \hat{a}_{23} , and \hat{b}_2 are also defined. Next, from (71a)

$$x_1 = \frac{1}{a_{11}} [b_1 - a_{12}\hat{x}_2 - a_{13}\hat{x}_3]$$

Its rounded value is

$$\hat{x}_1 = \frac{1}{a_{11}} [b_1(1+3\epsilon) - a_{12}\hat{x}_2(1+4\epsilon) - a_{13}\hat{x}_3(1+3\epsilon)]$$

With the help of (72) and (75),

$$\hat{x}_1 = x_1 + \frac{1}{a_{11}} \left\{ 3\epsilon b_1 - 4\epsilon a_{12}x_2 - 2\epsilon \frac{a_{12}}{a_{22}} b_2 + 4\epsilon \frac{a_{12}a_{23}}{a_{22}} x_3 - 4\epsilon a_{13}x_3 \right\}$$

Using the approximations $x_2 \approx \hat{x}_2$ and $x_3 \approx \hat{x}_3$, x_1 is expressed in terms of \hat{x}_1 , \hat{x}_2 , and \hat{x}_3 as

$$x_1 = \hat{x}_1 - \frac{1}{a_{11}} \left\{ 3\epsilon b_1 - 2\epsilon b_2 \frac{a_{12}}{a_{22}} - 4\epsilon a_{12}\hat{x}_2 + \frac{4\epsilon}{1+\epsilon} \left(\frac{a_{12}a_{23}}{a_{22}} - a_{13} \right) \hat{x}_3 \right\}$$

(78)

Substituting (78), (76), and (73) into (71a) and rearranging terms, give

$$\underbrace{a_{11}}_{\hat{a}_{11}} \hat{x}_1 + \underbrace{a_{12}(1+4\epsilon)}_{\hat{a}_{12}} \hat{x}_2 + \underbrace{a_{13}(1+3\epsilon)}_{\hat{a}_{13}} \hat{x}_3 = \underbrace{b_1(1+3\epsilon)}_{\hat{b}_1} \quad (79)$$

where \hat{a}_{11} , \hat{a}_{12} , \hat{a}_{13} , and \hat{b}_1 are also defined. Finally, put (74), (77) and (79) into a single matrix equation.

$$\underbrace{\begin{bmatrix} \hat{a}_{11}=a_{11} & \hat{a}_{12}=a_{12}(1+4\epsilon) & \hat{a}_{13}=a_{13}(1+3\epsilon) \\ 0 & \hat{a}_{22}=a_{22} & \hat{a}_{23}=a_{23}(1+3\epsilon) \\ 0 & 0 & \hat{a}_{33}=a_{33} \end{bmatrix}}_{\hat{A} = A + \Delta A} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}}_{\hat{x}} = \underbrace{\begin{bmatrix} \hat{b}_1=b_1(1+3\epsilon) \\ \hat{b}_2=b_2(1+2\epsilon) \\ \hat{b}_3=b_3(1+\epsilon) \end{bmatrix}}_{\hat{b} = b + \Delta b} \quad (80)$$

from which one easily gets

$$\Delta A = \begin{bmatrix} 0 & 4\epsilon a_{12} & 3\epsilon a_{13} \\ 0 & 0 & 3\epsilon a_{23} \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & a_{12} & a_{13} \\ 0 & 0 & a_{23} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 3 \end{bmatrix} \quad (81)$$

$$\Delta \underline{b} = \begin{bmatrix} 3b_1 \\ 2b_2 \\ b_3 \end{bmatrix} \epsilon = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \epsilon \quad (82)$$

Generalization to nxn Triangular Matrix Equation. Consider

$$\underbrace{\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ & a_{22} & \dots & a_{2n} \\ & & \ddots & | \\ & & & | \\ & & & | \\ & & & a_{nn} \end{bmatrix}}_A \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ | \\ | \\ | \\ x_n \end{bmatrix}}_{\underline{x}} = \underbrace{\begin{bmatrix} b_1 \\ b_2 \\ | \\ | \\ | \\ b_n \end{bmatrix}}_{\underline{b}} \quad (83)$$

Its equivalent perturbation equation for evaluating rounding errors can be obtained by generalizing the result of (80), which is

$$\underbrace{\begin{bmatrix} \hat{a}_{11} & \hat{a}_{12} & \dots & \hat{a}_{1n} \\ & \hat{a}_{22} & \dots & \hat{a}_{2n} \\ & & \ddots & | \\ & & & | \\ & & & | \\ & & & \hat{a}_{nn} \end{bmatrix}}_{\hat{A}} \underbrace{\begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ | \\ | \\ | \\ \hat{x}_n \end{bmatrix}}_{\hat{\underline{x}}} = \underbrace{\begin{bmatrix} \hat{b}_1 \\ \hat{b}_2 \\ | \\ | \\ | \\ \hat{b}_n \end{bmatrix}}_{\hat{\underline{b}}} \quad (84)$$

where

$$\hat{a}_{kj} = \begin{cases} a_{kj} & k=j \\ a_{kj}[1+(n-j+3)\epsilon] & 1 \leq k \leq j \leq n \\ 0 & \text{otherwise} \end{cases} \quad (85)$$

$$\hat{b}_k = b_k[1+(n-k+1)\epsilon] \quad k=1 \text{ to } n \quad (86)$$

Thus,

$$\Delta a_{kj} = \begin{cases} a_{kj}(n-j+3)\epsilon & 1 \leq k < j \leq n \\ 0 & \text{otherwise} \end{cases} \quad (87)$$

$$\Delta b_k = b_k(n-k+1)\epsilon \quad k = 1 \text{ to } n \quad (88)$$

$$\Delta A = \begin{bmatrix} 0 & (n+1)a_{12} & na_{13} & \dots & 4a_1(n-1) & 3a_{1n} \\ & 0 & na_{23} & \dots & 4a_2(n-1) & 3a_{2n} \\ & & & \ddots & & \vdots \\ & & & & & 3a_{(n-1)n} \\ & & & & & 0 \end{bmatrix} \epsilon$$

$$\Delta A = \underbrace{\begin{bmatrix} 0 & a_{12} & a_{13} & \dots & a_{1n} \\ & 0 & a_{23} & \dots & a_{2n} \\ & & & \ddots & \\ & & & & a_{(n-1)n} \\ & & & & 0 \end{bmatrix}}_{A_D} \underbrace{\begin{bmatrix} 0 & & & & \\ & n+1 & & & \\ & & & & \\ & & & & 4 \\ & & & & 3 \end{bmatrix}}_{M_A} \quad (89)$$

$$\underline{\Delta b} = \begin{bmatrix} nb_1 \\ (n-1)b_2 \\ | \\ | \\ b_n \end{bmatrix} \epsilon \leq \underbrace{\begin{bmatrix} n & & & & \\ & n-1 & & & \\ & & \circ & & \\ & & & \ddots & \\ \circ & & & & 1 \end{bmatrix}}_{M_b} \underbrace{\begin{bmatrix} b_1 \\ b_2 \\ | \\ | \\ b_n \end{bmatrix}}_{\underline{b}} \beta \quad (90)$$

where matrices A_D , M_A , and M_b are also defined.

11.3 Resultant Rounding Error in the Inverse Matrix

Two sets of equivalent perturbations for A and \underline{b} have been obtained to account for rounding errors. One set, \tilde{A} and $\tilde{\underline{b}}$ as given by (59) and (60), account for errors from triangularization. The second set, ΔA and $\underline{\Delta b}$ as given by (89) and (90), account for errors from back substitution. The resultant equivalent perturbations for A and \underline{b} are given by the sums

$$\delta A = \tilde{A} + \Delta A = \hat{A} - A \quad (91)$$

$$\delta \underline{b} = \tilde{\underline{b}} + \underline{\Delta b} = \hat{\underline{b}} - \underline{b} \quad (92)$$

The resultant rounding error in $\hat{\underline{x}}$ in the solution of

$$A \underline{x} = \underline{u}_j \quad j = 1 \text{ to } n, \quad (93)$$

where \underline{u}_j is the j th column vector of the identity matrix I , is given by

$$\delta \underline{x}_j = \hat{A}^{-1} [\delta \underline{b}_j - \delta A \hat{\underline{x}}_j] \quad (94)$$

**ORIGINAL PAGE IS
OF POOR QUALITY**

This equation is obtained in a way similar to that of (68). Rounding error in \hat{A}^{-1} is then given by

$$\begin{aligned}\delta(A^{-1}) &= [\delta\underline{x}_1 \text{ - - - } \delta\underline{x}_n] \\ &= \hat{A}^{-1}[(\delta\underline{b}_1 - \delta A \hat{\underline{x}}_1) \text{ - - - } (\delta\underline{b}_n - \delta A \hat{\underline{x}}_n)]\end{aligned}\tag{95}$$

The error norm of the computer's inverse matrix of A is therefore

$$||\delta(A^{-1})|| \leq ||\hat{A}^{-1}|| \cdot ||E||\tag{96}$$

where the matrix E has been defined in (95). The relative error norm is

$$\epsilon \leq \frac{||\delta(A^{-1})||}{||\hat{A}^{-1}||} = ||E||\tag{97}$$

It is obvious that the evaluation of (96) or (97) involves a good deal of computation and should be done by a computer. Figure III-1 is a computation block diagram for this purpose.

11.4. A Numerical Example

Consider inverting the following matrix

$$A = \begin{bmatrix} 3.235 & -1.234 & 3.256 \\ 1.023 & -5.235 & 0.921 \\ 1.336 & 2.120 & -8.235 \end{bmatrix}$$

using computer of different finite decimal wordlength. Then evaluate the corresponding error norms using the procedure of Figure 1. The

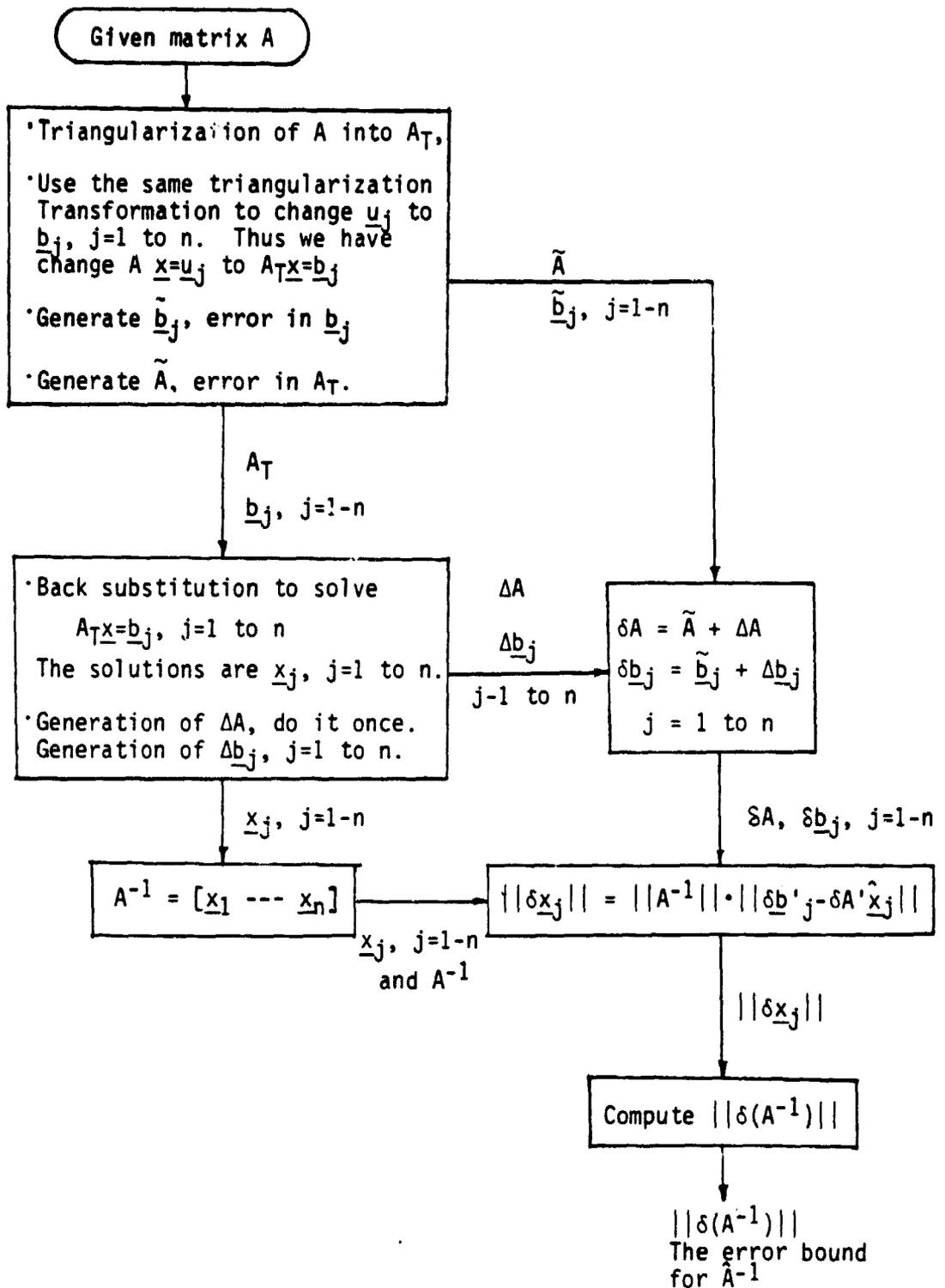


Figure III-1. Flow chart for evaluating rounding error bound of matrix inverse

effectiveness of the procedure is examined by comparing these error norms to the corresponding actual error norms. The actual norms are approximately obtained by using a computer having a much longer decimal wordlength. The result is given in Table 1, which shows that error norms obtained by using the proposed procedure are indeed very conservative. Notice that error norm decreases with increasing worklength. It is interesting to note that when the proposed method is used all error norms have the same mantissa.

Table 1

Matrix Inversion Error Norms

Wordlength: No. of places after decimal point	Error norm	
	By proposed method	Actual value
3	3.706×10^{-2}	7.782×10^{-5}
5	3.706×10^{-4}	4.530×10^{-7}
8	3.706×10^{-7}	4.672×10^{-10}

12. Rounding Error Bound for Kalman Filtering

Consider a process modeled by the following set of equations.

$$\underline{x}_k = \Phi_{k-1} \underline{x}_{k-1} + \omega_{k-1} \quad \dim \underline{x}_k = n$$

$$\underline{z}_k = H_k \underline{x}_k + \underline{v}_k \quad \dim \underline{z}_k = m$$

$$E \underline{x}(0) = \underline{x}_0 \quad E[\underline{\tilde{x}}(0) \underline{\tilde{x}}(0)^T] = P_0$$

$$\underline{w}_k \sim N(0, Q_k) \quad V_k \sim N(0, R_k)$$

$$E[\underline{w}_k \underline{v}_j] = 0 \quad \text{all } j, k$$

The Kalman Filter algorithm consists of the following equations.

$$\underline{x}_k^* = \phi_{k-1} \underline{x}_{k-1}^* + K_k [z_k - H_k \phi_{k-1} \underline{x}_{k-1}^*], \quad \underline{x}^*(0) = \underline{x}_0 \quad (98)$$

$$K_k = P_{kp} H_k^T [H_k P_{kp} H_k^T + R_k]^{-1} \text{ or } K_k = P_k H_k^T R_k^{-1} \quad (99)$$

$$P_{kp} = \phi_{k-1} P_{k-1} \phi_{k-1}^T + Q_k \quad (100)$$

$$P_k = P_{kp} - K_p H_k P_{kp} \quad (101)$$

where \underline{x}^* is the estimate of \underline{x} . The present interest is to find the bound of the rounding error norm for \underline{x}^* . For the sake of convenience, the asterisk "*" will be dropped, and \underline{x} will denote the rounded value of the estimate.

Error in Rounded P_{kp} . Recall (100), that is,

$$P_{kp} = \phi_{k-1} P_{k-1} \phi_{k-1}^T + Q_{k-1}$$

Its rounded result is

$$\begin{aligned} \hat{P}_{kp} &= \phi_{k-1} P_{k-1} \phi_{k-1}^T (1 + \overline{2n+1}\epsilon) + Q_{k-1} (1 + \epsilon) \\ &= P_{kp} (1 + \overline{2n+1}\epsilon) - 2n\epsilon Q_{k-1} \end{aligned} \quad (102)$$

The rounding error is

$$\tilde{P}_{kp} = (2n+1)\epsilon P_{kp} - 2n\epsilon Q_{k-1} \leq (2n+1)\epsilon P_{kp} \quad (103)$$

Error in Rounded P_k . Recall (101), that is,

$$P_k = [I - K_k H_k] P_{kp}$$

Its rounded value is

$$\begin{aligned} \hat{P}_k &= [I(1+\epsilon) - K_k H_k (1+\overline{m+1}\epsilon)] P_{kp} (1+n\epsilon) \\ &= [I(1+\overline{n+1}\epsilon) - K_k H_k (1+\overline{n+m+1}\epsilon)] \hat{P}_{kp} \\ &= [I(1+\overline{n+1}\epsilon) - K_k H_k (1+\overline{n+m+1}\epsilon)] [P_{kp} + \tilde{P}_{kp}] \\ &\approx P_k + [(n+1)\epsilon I - K_k H_k (n+m+1)\epsilon] P_{kp} + [I - K_k H_k] \tilde{P}_{kp} \\ &= P_k [1 + (n+m+1)\epsilon] - m\epsilon P_{kp} + [I - K_k H_k] \tilde{P}_{kp} \end{aligned} \quad (104)$$

$$\begin{aligned} \tilde{P}_k &= (n+m+1)\epsilon P_k - m\epsilon P_{kp} + [I - K_k H_k] \tilde{P}_{kp} \\ &\leq (n+m+1)\epsilon P_k - m\epsilon P_{kp} + \frac{[I - K_k P_k](2n+1) P_{kp}}{(2n+1)\epsilon P_k} \\ &\leq (3n+m+2)\epsilon P_k \end{aligned} \quad (105)$$

Error in Rounded K_k . Recall (99), that is,

$$K_k = P_k H_k^T R_k^{-1}$$

Its rounded form is

$$\hat{K}_k = \hat{P}_k H_k^T \hat{R}_k^{-1} (1 + \overline{n+m}\epsilon) \quad (106)$$

By (104) and (103)

$$\begin{aligned} \hat{P}_k &\leq (1 + \overline{n+m+1}\epsilon) P_k - m\epsilon P_{kp} + \frac{[I - K_k H_k](2n+1)\epsilon P_{kp}}{(2n+1)\epsilon P_k} \\ &= (1 + \overline{3n+m+2}\epsilon) P_k - m\epsilon P_{kp} \end{aligned}$$

Assume R_k diagonal

$$\hat{R}_k^{-1} = R_k^{-1} (1 + \epsilon)$$

Then (106) becomes

$$\begin{aligned} \hat{K}_k &= [(1 + \overline{3n+m+2}\epsilon) P_k - m\epsilon P_{kp}] H_k^T R_k^{-1} (1 + \epsilon) (1 + \overline{n+m}\epsilon) \\ &= (1 + \overline{4n+2m+3}\epsilon) P_k H_k^T R_k^{-1} - m\epsilon P_{kp} H_k^T R_k^{-1} \end{aligned} \quad (107)$$

The rounding error is

$$\begin{aligned} \tilde{K}_k &= (4n+2m+3)\epsilon K_k - m\epsilon P_{kp} H_k^T R_k^{-1} \\ &\leq (4n+2m+3)\epsilon K_k \end{aligned} \quad (108)$$

Error in Rounded \underline{x}_k . Recall (98), which is,

$$\begin{aligned} \underline{x}_k &= \Phi_{k-1} \underline{x}_{k-1} + K_k [z_k - H_k \Phi_{k-1} \underline{x}_{k-1}] \\ &= [I - K_k H_k] \Phi_{k-1} \underline{x}_{k-1} + K_k z_k \end{aligned} \quad (98)$$

**ORIGINAL PAGE IS
OF POOR QUALITY**

Let

$$F_k = [I - K_k H_k] \phi_{k-1} \quad (109)$$

$$\underline{u}_k = K_k \underline{z}_k \quad (110)$$

Then (98) becomes

$$\underline{x}_k = F_k \underline{x}_{k-1} + \underline{u}_k \quad (111)$$

Develop the following rounded quantities.

$$\begin{aligned} \hat{F}_k &= [I(1+\epsilon) - \hat{K}_k H_k(1+\overline{m+1}\epsilon)] \phi_{k-1} (1+n\epsilon) \\ &= [I(1+\overline{n+1}\epsilon) - K_k(1+\overline{4n+2m+3}\epsilon) H_k(1+\overline{n+m+1}\epsilon)] \phi_{k-1} \\ &= [I(1+\overline{n+1}\epsilon) - K_k H_k(1+\overline{5n+3m+4}\epsilon)] \phi_{k-1} \\ &= F_k(1+\overline{5n+3m+4}\epsilon) - (4n+3m+3)\epsilon \phi_{k-1} \end{aligned} \quad (112)$$

$$\begin{aligned} \hat{\underline{u}}_k &= \hat{K}_k \underline{z}_k (1+m\epsilon) \\ &= K_k(1+\overline{4n+2m+3}\epsilon) \underline{z}_k (1+m\epsilon) \\ &= K_k \underline{z}_k (1+\overline{4n+3m+3}\epsilon) \\ &= \underline{u}_k (1+\overline{4n+3m+3}\epsilon) \end{aligned} \quad (113)$$

$$\begin{aligned} \hat{\underline{x}}_k &= \hat{F}_k \underline{x}_{k-1} (1+\overline{n+1}\epsilon) + \hat{\underline{u}}_k (1+\epsilon) \\ &\leq (1+\overline{5n+3m+4}\epsilon) F_k \underline{x}_{k-1} (1+\overline{n+1}\epsilon) + \underline{u}_k (1+\overline{4n+3m+3}\epsilon) (1+\epsilon) \\ &= (\overline{6n+3m+5}\epsilon+1) F_k \underline{x}_{k-1} + (\overline{4n+3m+4}\epsilon+1) \underline{u}_k \end{aligned} \quad (114)$$

Define

$$\hat{F}_k = (1 + \overline{6n+3m+5\epsilon})F_k \quad (115)$$

$$\hat{u}_k = (1 + \overline{4n+3m+4\epsilon})u_k \quad (116)$$

Then (114) can be written as

$$\hat{x}_k = \hat{F}_k x_{k-1} + \hat{u}_k \quad (117)$$

Case of k = 3 Eq. (111) gives the exact x_3 as

$$x_3 = F_3 F_2 F_1 x_0 + F_3 F_2 u_1 + F_3 u_2 + u_3 \quad (118)$$

The rounded x_3 is

$$\begin{aligned} \hat{x}_3 = & \hat{F}_3 \hat{F}_2 \hat{F}_1 x_0 (1 + \overline{3n+3\epsilon}) + \hat{F}_3 \hat{F}_2 \hat{u}_1 (1 + \overline{2n+3\epsilon}) \\ & + \hat{F}_3 \hat{u}_2 (1 + \overline{n+2\epsilon}) + \hat{u}_3 (1 + \epsilon) \end{aligned}$$

Using (115) and (116), and combining terms,

$$\begin{aligned} \hat{x}_3 = & F_3 F_2 F_1 x_0 [1 + 3(7n+3m+6)\epsilon] + F_3 F_2 u_1 [1 + \overline{18n+9m+17\epsilon}] \\ & + F_3 u_2 [1 + \overline{11n+6m+11\epsilon}] + u_3 [1 + \overline{4n+3m+5\epsilon}] \\ = & x_3 [1 + 3(7n+3m+6)\epsilon] - (3m+1)\epsilon F_3 F_2 u_1 \\ & - (10n+3m+7)\epsilon F_3 u_2 - (17n+6m+13)\epsilon u_3 \end{aligned} \quad (119)$$

The rounding error is

$$\begin{aligned} \tilde{x}_3 = & 3(7n+3m+6)\epsilon x_3 - (3m+1)\epsilon F_3 F_2 u_1 \\ & - (10n+3m+7)\epsilon F_3 u_2 - (17n+6m+13)\epsilon u_3 \end{aligned} \quad (120)$$

Its norm is bounded by

$$\begin{aligned} \|\tilde{x}_3\| \leq & 3(7n+3m+6)\epsilon \|x_3\| - (3n+1)\epsilon \|F_3 F_2 u_1\| \\ & - (10n+3m+7)\epsilon \|F_3 u_2\| - (17n+6m+13)\epsilon \|u_3\| \end{aligned} \quad (121)$$

Assume $\|F_k\| \leq F$, $\|u_k\| \leq u$, and $F^i u \leq B$ for $i = 0$ to 2 , then

$$\begin{aligned} \|\tilde{x}_3\| \leq & 3(7n+3m+6)\epsilon \|x_3\| + (30n+9m+21)\epsilon B \\ \leq & B\{3(7n+3m+6)\|x_3\| + [3(3n+1) + \frac{3(3-1)}{2}(7n+3m+6)]B\} \end{aligned} \quad (122)$$

Case of $k=4$. Again, by (111),

$$\underline{x}_4 = F_4 F_3 F_2 F_1 \underline{x}_0 + F_4 F_3 F_2 \underline{u}_1 + F_4 F_3 \underline{u}_2 + F_4 \underline{u}_3 + \underline{u}_4 \quad (123)$$

Following similar derivation, gives the norm of rounding error as

$$\|\tilde{x}_4\| \leq B\{4(7n+3m+6)\|x_4\| + [4(3n+1) + \frac{4(4-1)}{2}(7n+3m+6)]B\} \quad (124)$$

The general case k . From the equation pattern of (122) and (124) for $k=3$ and 4 , the general case is found to be

$$\|\tilde{x}_k\| \leq B\{k(7n+3m+6)\|x_k\| + [k(3n+1) + \frac{k(k-1)}{2}(7n+3m+6)]B\} \quad (125)$$

Eq. (125) is the main result of this chapter, which gives the bound of error norm for the rounded state estimate. The bound

depends on β , the unit rounding error; k , the number of interactions; n and m , the dimension parameters of the process; $\|\underline{x}_k\|$, the norm of the estimated state; and B , a quantity depends on K_k , H_k , Φ_k and \underline{u}_k . The usefulness of this equation is at providing a general idea on the desired number of digits for the mantissa of the computer's floating number system. The following example will illustrate this point.

Example. Consider a one year GP- β operation where relativistic data are taken every 10 seconds. Assume that the Kalman filtering involved in data reduction is also operated at 10 second interaction period. Then, at the end of one year period, the value of k would be $k=365 \times 24 \times 3600/10 = 3.1536 \times 10^6$. Assume \underline{x}_k be a 10-vector and \underline{z}_k be a 2-vector, so $n=10$ and $m=2$. then,

$$k(7n+3m+b) = 2.6490 \times 10^8$$

$$K(3n+1) + 0.5k(k-1)(7n+3m+6) = 1.2949 \times 10^{16}$$

Just for the sake of discussion, assume only one term at the right-hand side of (125) dominates the result. If the first term dominates, one may estimate the desired β from

$$K(7n+3m+6)\beta = 1$$

so

$$\beta = 0.3775 \times 10^8$$

Comparing to (8), the formula $\beta = 0.5 \times 10^{1-t}$, gives $t \approx 9$, therefore 9 digits are desired for the mantissa of the floating point number system. On the other hand, if the second term of (125) dominates, one may estimate the desired β from

$$[k(3n+1) + 0.5k(k-1)(7n+3m+6)]\beta = 1$$

so

$$\beta = .7723 \times 10^{-16}$$

comparing to (8), gives $t \approx 17$. Hence 17 digits are desired for the mantissa.

13. Remarks

1. The main result of this chapter, given by (125), can probably be further refined to tighten the predicted bound while maintaining its reliability. This result was obtained after several different approaches to the problem had been attempted.

2. It is desirable to find out the effectiveness of (125) by a computer emulation of the GP-B data reduction. This has not been done.

14. References

1. J. H. Wilkinson, Rounding Errors in Algebraic Process, Prentice-Hall, Englewood Cliffs, New Jersey, 1963.
2. G. E. Forsythe, M. A. Malcolm, and C. B. Moler, Computer Methods for Mathematical Computations, Prentice-Hall, Englewood Cliffs, New Jersey, 1977.

3. J. B. Mankin and J. C. Hung, "On Rounding Errors in the Computation of Transition Matrices," Proceeding of the Joint Automatic Control Conference, American Society of Chemical Engineers, 1969, pp.60-64.
4. Pat H. Sterbenz, Floating-Point Computation, Prentice-Hall, Englewood Cliffs, New Jersey, 1974.
5. James S. Vandergraft, Introduction to Numerical Computation, Academic Press, New York, 1978.
6. D. I. Steinberg, Computational Matrix Algebra, McGraw-Hill, New York, 1974.
7. J. C. Hung, "GP-B Error Modeling and Analysis," an annual report for NASA Contract NA58-34426, The University of Tennessee, Electrical Engineering Department, 30 September, 1982.

CHAPTER III

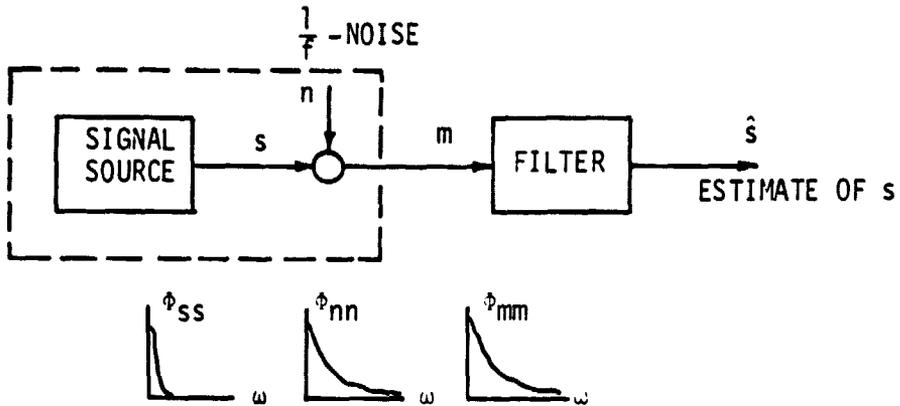
COMBATING THE EFFECT OF $\frac{1}{f}$ -NOISE

This chapter contains a discussion of the effectiveness of spacecraft rolling and the use of filtering for eliminating the $\frac{1}{f}$ -noise from the measured relativistic drift data.

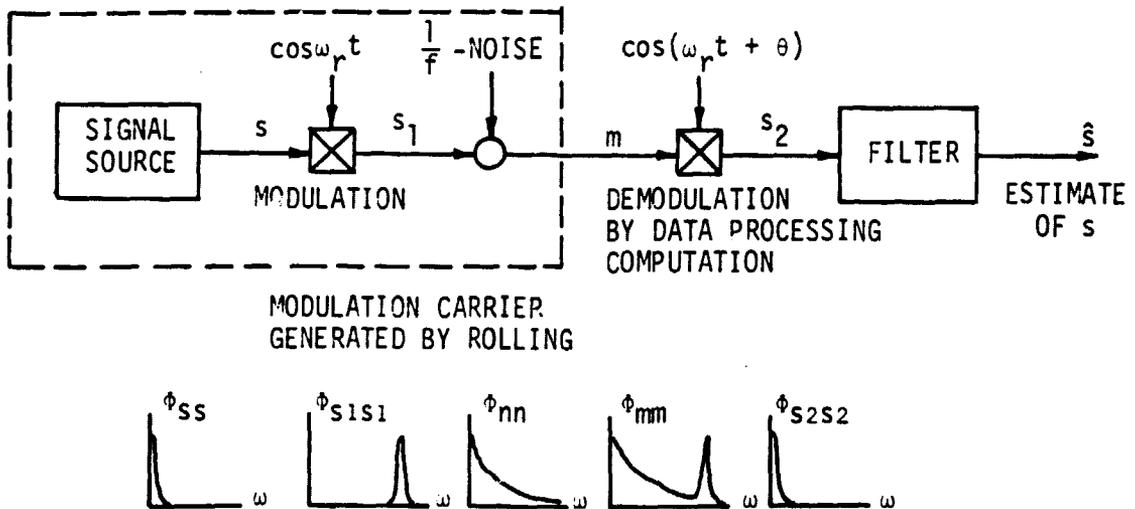
1. Rolling and Derolling

The effect of rolling the spacecraft on the relativistic signal is equivalent to the modulation operation in communication engineering, while the derolling the signal by software coordinate transformation is equivalent to the demodulation operation. Figure III-1 helps to show the effect. The $\frac{1}{f}$ -noise may be considered as an additive noise whose power spectral density ϕ_{nn} is inversely proportional to the frequency f . Therefore its power density is the heaviest near the zero frequency where the power spectrum, ϕ_{SS} , of the relativistic signal is lying. This is shown in Figure III-1(a). Under this condition, it is very difficult to extract the true signal from the noise-contaminated signal by filtering.

By rolling the spacecraft, a carrier signal is generated on each GP-B gyro and is being modulated by the relativistic signal. The modulation process takes place before the true relativistic signal has been contaminated by the $\frac{1}{f}$ -noise. Thus the true signal's power spectrum ϕ_{SS} is transformed to a power spectrum $\phi_{S_1S_1}$ situated in a higher frequency region centered at the roll frequency ω_r where



(a) Without roll and deroll operations.



(b) With roll and deroll operations.

Figure III-1. Effect of roll and deroll on signal and noise.

the power density of the $\frac{1}{f}$ -noise is much less. This is shown in Figure III-1(b). After receiving the signal by measurement, the signal is derolled and filtered by digital computation. An analytical description of the entire process is given below.

Referring to Figure III-1(b), let $s(t)$ be the true signal. The rolling motion of the spacecraft produces a carrier signal $c(t) = \cos \omega_r t$, where ω_r is the radian frequency of the rolling. The modulated signal is

$$s_1(t) = s(t) c(t) = s(t) \cos \omega_r t \quad (1)$$

The available measured signal is

$$m(t) = s_1(t) + n(t) = s(t) \cos \omega_r t + n(t) \quad (2)$$

where $n(t)$ is the $\frac{1}{f}$ -noise. The derolling by software is to multiply $m(t)$ by $d(t) = \cos(\omega_r t + \theta)$ where θ accounts for the phase difference between $c(t)$ and $d(t)$. The resulting signal is

$$\begin{aligned} s_2(t) &= m(t) d(t) \\ &= s(t) \cos(\omega_r t + \theta) + n(t) \cos(\omega_r t + \theta) \\ &= \frac{1}{2} s(t) \cos \theta + \frac{1}{2} s(t) \cos(2\omega_r t + \theta) + \\ &\quad n(t) \cos(\omega_r t + \theta) \end{aligned} \quad (3)$$

After a filtering process the low frequency component of $s_2(t)$ is retained, giving,

$$\hat{s} = \frac{1}{2} s(t) \cos \theta \quad (4)$$

For maximum strength of the recovered signal, $\theta = 0$, meaning a synchronous demodulation is desired.

For the GP-B experiment the relativistic signal consists of a pair of signals $s_x(t)$ and $s_y(t)$. The process of rolling, derolling and filtering is shown in Figure III-2. The spacecraft rolling transform $s_x(t)$ and $s_y(t)$ to $s_1(t)$ and $s_2(t)$ modeled by

$$\begin{bmatrix} s_1(t) \\ s_2(t) \end{bmatrix} = \begin{bmatrix} \cos \omega_r t & \sin \omega_r t \\ -\sin \omega_r t & \cos \omega_r t \end{bmatrix} \begin{bmatrix} s_x(t) \\ s_y(t) \end{bmatrix} \quad (5)$$

where ω_r is the radian frequency of rolling. The measured signals are $m_1(t)$ and $m_2(t)$ which contain the $\frac{1}{f}$ -noise $n_1(t)$ and $n_2(t)$

$$\begin{bmatrix} m_1(t) \\ m_2(t) \end{bmatrix} = \begin{bmatrix} s_1(t) + n_1(t) \\ s_2(t) + n_2(t) \end{bmatrix} \quad (6)$$

The ideal software deroll is a recursively computed coordinate transformation given by

$$\begin{aligned} \begin{bmatrix} s_3(t) \\ s_4(t) \end{bmatrix} &= \begin{bmatrix} \cos \omega_r t & -\sin \omega_r t \\ \sin \omega_r t & \cos \omega_r t \end{bmatrix} \begin{bmatrix} m_1(t) \\ m_2(t) \end{bmatrix} \\ &= \begin{bmatrix} s_x \\ s_y \end{bmatrix} + \begin{bmatrix} \cos \omega_r t & -\sin \omega_r t \\ \sin \omega_r t & \cos \omega_r t \end{bmatrix} \begin{bmatrix} n_1(t) \\ n_2(t) \end{bmatrix} \end{aligned} \quad (7)$$

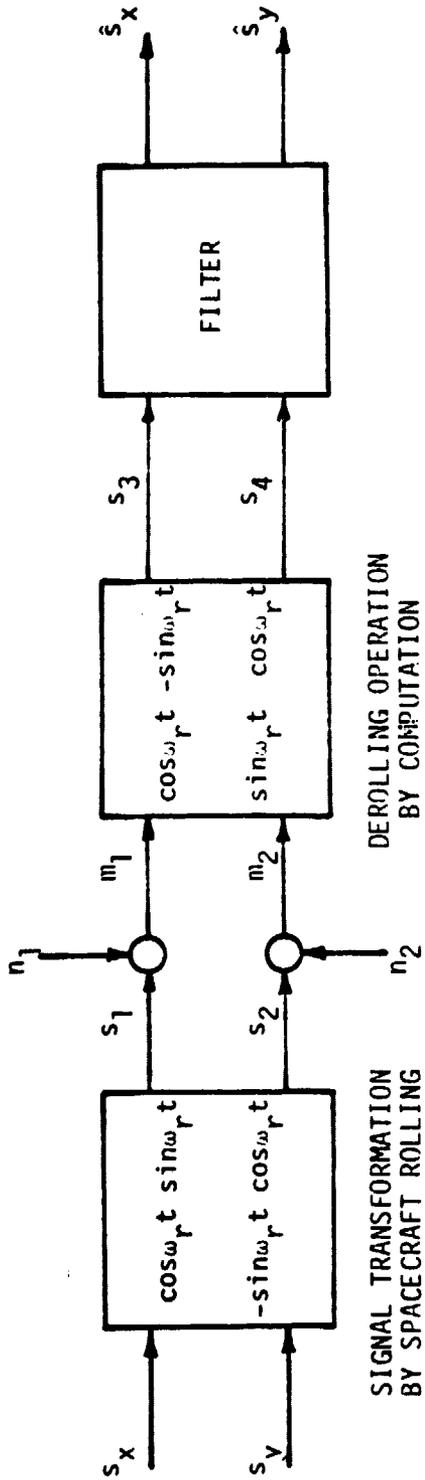


Figure III-2. A two-dimensional roll and deroll model.

After a filtering process to retain the low frequency component of the signal, one gets

$$\begin{bmatrix} \hat{s}_x \\ \hat{s}_y \end{bmatrix} = \begin{bmatrix} s_x \\ s_y \end{bmatrix} \quad (8)$$

2. Potential Error Sources

The potential error sources for the roll-deroll process consist of the following:

1. Phase difference between the roll signal and deroll signal.
2. Off-tuned deroll frequency.
3. Rectification error due to the $\frac{1}{T}$ -noise component at roll frequency.
4. Imperfect filtering of noise.
5. Variation in roll rate.

The first three of these error sources will be further discussed below. Study of the last two error sources is not yet complete.

3. Effect of Nonzero Phase Between Roll and Deroll Signals

A nonzero phase angle θ between the roll and deroll signals does not directly introduce any error. However, it does reduce the strength of the measured true signal by a factor of $\cos \theta$ as shown in Eq. (3). The effect is a reduction of the signal-to-noise ratio, making the extraction of the true signal by filtering harder. If the absolute value of θ can be kept below one degrees, then

$$1 - \cos^2 5^\circ = 1 - .9998^2 = .0004,$$

the reduction of signal-to-noise ratio will be less than .04%. Synchronizing the two signals to within one degree is not hard to do, since the roll motion is tracked by the signal from a star blipper sensor.

4. Off-Tuned Deroll Frequency

Consider a one-dimensional true signal $s(t)$. Let ω_r be the roll frequency and $\omega_r + \Delta\omega$ be the deroll frequency. Assuming zero phase difference, that is $\theta=0$, the error caused by $\Delta\omega$, the off-tuned deroll frequency, can be analyzed as follows:

The rolled signal is $s(t) \cos \omega_r t$ and the measured signal is

$$m(t) = s(t) \cos \omega_r t + n(t) \quad (9)$$

The derolled signal is

$$\begin{aligned} s_2(t) &= m(t) \cos (\omega_r t + \Delta\omega t) \\ &= \{s(t) \cos \omega_r t + n(t)\} \cos(\omega_r t + \Delta\omega t) \\ &= \frac{1}{2} s(t) \cos \Delta\omega t + \frac{1}{2} s(t) \cos (2\omega_r t + \Delta\omega t) + \\ &\quad n \cos (\omega_r t + \Delta\omega t) \end{aligned} \quad (10)$$

The low frequency component of $s_2(t)$ will be extracted by filtering, giving

$$\hat{s}(t) = \frac{1}{2} s(t) \cos \Delta\omega t \quad (11)$$

Assuming $s(t)$ is a constant signal, then the extracted signal $\hat{s}(t)$ can be affected by $\Delta\omega$ to various degrees depending on the size of $\Delta\omega$.

Ideally, $\Delta\omega=0$, that is, the deroll frequency equals the roll frequency. The roll frequency is monitored repeatedly by the use of a star blipper sensor. However, the finite wordlength limitation of the computer register will cause a finite wordlength induced $\Delta\omega$. Two sample calculations will illustrate this point.

Sample Calculation 1. Consider a roll period of $T = 600$ seconds. The roll frequency is

$$\omega_r = \frac{2\pi}{T} = .1047 \dots \times 10^{-1} \text{ rad/sec}$$

For a register having six places for mantissa,

$$\Delta\omega \approx .0000005 \times 10^{-1} = .5 \times 10^{-7} \text{ rad/sec}$$

Therefore Eq. (11) has a period of

$$T_{\Delta\omega} = \frac{2\pi}{\Delta\omega} = 1.2566 \times 10^8 \text{ sec} \approx 4 \text{ years}$$

During a one year period $\Delta\omega t$ begins with a value of 0 and ends with a value of $\frac{\pi}{2}$. Correspondingly, $\cos \Delta\omega t$ changes from 1 to 0, resulting in an extracted signal value of

$$\hat{s}(t) = s(t) \quad \text{to} \quad \hat{s}(t) = 0$$

which involves very severe errors.

Sample Calculation 2. Same sample roll frequency as the previous case is used, but use a register having ten places for mantissa. Now

$$\Delta\omega = .0000000005 \times 10^{-1} = .5 \times 10^{-11} \text{ rad/sec}$$

Then Eq. (11) has a period of

$$T_{\Delta\omega} = \frac{2\pi}{\Delta\omega} = 1,2566 \times 10^{12} \text{ sec} \approx 40,000 \text{ years}$$

Thus for a time averaging process over an one-year time, the effect of round-off induced $\Delta\omega$ is negligible. This is because that at the end of one year

$$\cos\left(\frac{2\pi}{40,000}\right) = .999999987$$

giving an error in $\hat{s}(t)$ of only $1.234 \times 10^{-6}\%$.

In conclusion, if good synchronization is maintained between the roll and deroll signals and if the wordlength of the computer registers are sufficiently long, the problem of off-turned deroll frequency can be eliminated.

5. Rectification Error Due to the $\frac{1}{f}$ -noise Components Within the Frequency Spread of the Signal

The relativistically drifted gyro position may be considered a ramp function of time, that is

$$s(t) = kt \quad t \geq 0$$

Hence it has a Fourier transform of

$$s(\omega) = \frac{k}{\omega^2}$$

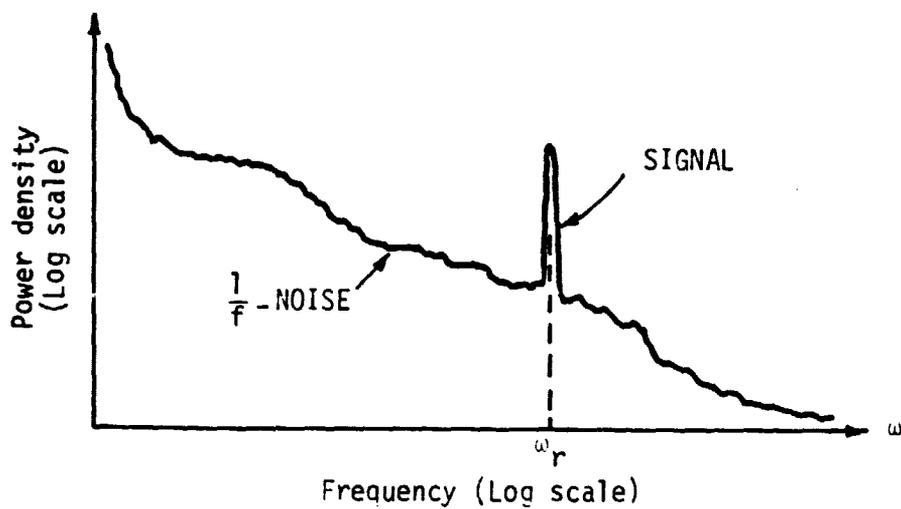
and a power spectral density function of

$$\phi_{SS}(\omega) = \frac{k^2}{\omega^4}$$

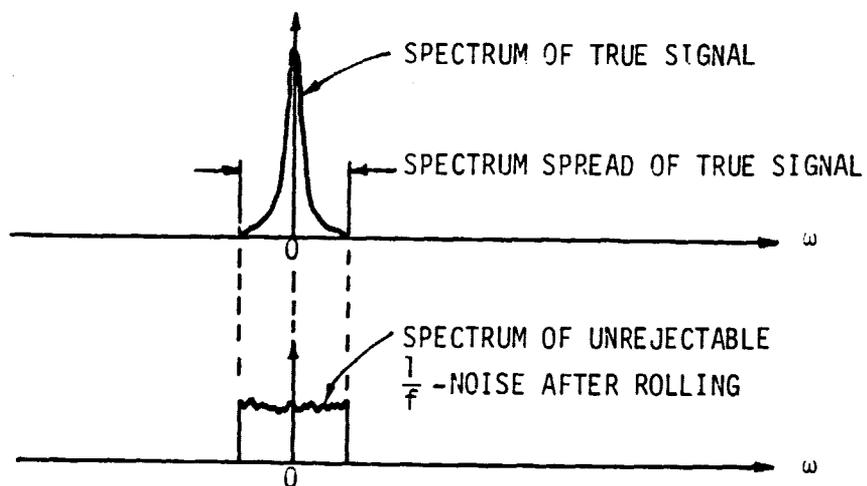
The signal has a spectrum spread of $\delta\omega$ which is nonzero. This spectrum spread is shifted to a higher frequency centered at the roll frequency ω_r , and is contaminated by the power spectrum of the $\frac{1}{f}$ -noise as shown in Figure III-3(a). After derolling not only the power spectrum of the true signal be shifted back to low frequency band but also the inphase components of the noise spectrum lying in the same frequency spread, as shown in Figure III-3(b). These components of noise cannot be suppressed completely, but an optimum choice of the bandwidth of a low-pass filter can provide a maximum signal-to-noise ratio. A quantitative treatment of this subject requires further study.

6. References

1. C. W. F. Everitt, "Report on a Program to Develop a Gyro Test of General Relativity in a Satellite and Associated Control Technology," Stanford University, June 1980. (The Green Book.)
2. P. Z. Peebles, Jr., Communication System Principles, Addison-Wesley, 1976.



(a)



(b)

Figure III-3. Spectra of signal and noise.