MASSACHUSETTS INSTITUTE OF TECHNOLOGY

TE-36

HYBRID ALTIMETER USING A STRAPDOWN INERTIAL NAVIGATION SYSTEM

by

Philippe Charles Jankowski

NGR-22-009-229

MEASUREMENT SYSTEMS LABORATORY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
CAMBRIDGE 39, MASSACHUSETTS

1

TE-36

HYBRID ALTIMETER USING A STRAPDOWN INERTIAL NAVIGATION SYSTEM

by

Philippe Charles Jankowski
Ingenieur E.N.S.I. Besancon-France-1968

Submitted in Partial Fulfillment
of the Requirements for the
Degree of Master of
Science
at the
MASSACHUSETTS INSTITUTE OF
TECHNOLOGY
January , 1970

Signature of the Author . . . .
Department of Aeronautics and Astronautics, January 1970

Certified by . . . .
Thesis Supervisor

. . . .
Technical Advisor

Accepted by . . . .
Chairman , Departmental Committee on Graduate Students

HYBRID ALTIMETER USING A STRAPDOWN INERTIAL NAVIGATION SYSTEM

by ; Philippe Charles Jankowski

"Submitted to the Department of Aeronautics and Astronautics on January 1970 in partial fulfillment of the requirements for the degree of Master of Science . "

ABSTRACT

Several methods of combining the output of a strapdown inertial navigation system with an external measurement of altitude are investigated. This combined information is used in the computation of the gravity field vector of the earth. The computation uses the two souces of information in a manner which minimizes the error in the indicated position due to the component errors in the altimeters and the gyros. Two sub-optimal hybrid altimeter configurations are considered. The first altimeter mixes radar altitude,which contains no dynamic lag,with vertical acceleration;the second combines barometric altitude,which contains dynamic lag,with vertical acceleration.The steady state mean squared error of either hybrid system is not reduced by the inclusion of vertical acceleration data. The results of linear optimal filtering theory for discrete systems are reviewed. The linear optimal filtering theory is applied to the problem of hybrid altimetry. It is shown that a combination of a radio altimeter,barometric altimeter and vertical acceleration reduces the mean squared error in the estimate of altitude to a great extent.It is also shown that this accuracy is only very lightly sensible to the error angle in tracking the vertical.

Thesis Supervisor : Winston R. Markey

Title : Professor of Aeronautics and Astronautics

ACKNOWLEDGEMENTS

The author wisches to thank his thesis technical advisor , Ken FRITTING . Thanks are also due to Professor W. Vander Velde for his informative discussions and Ron Hirschorn , R. Penny , and Tim Ryan who corrected the manuscript .

Acknowledgement is also made to the M.I.T. Computation Center for its work , done as problem M4131 .

This thesis was made possible by a N.A.S.A.-E.S.R.O. fellowship .

This report was prepared under DSR Project No. 70343 sponsored by the National Aeronautics and Space Administration Electronic Research Center, Cambridge, Massachusetts, through NASA Grant No. NGR 22-009-229 .

TABLE OF CONTENTS

NOTATION

$|\underline{J}|$ denotes determinant of the matrix $\underline{J}$

$\underline{J}'$ denotes the transpose of the matrix $\underline{J}$

A bar over a variable indicates the expected or mean value of that variable.

Underlined lower or upper case letters are column vectors.

Underlined capital letters denote matrices.

1. Introduction

In recent years, the requirement for extremely accurate navigation has become more common. A determination of the aircraft altimetry errors is needed for assesment of the adequacy of vertical separation standards. For horizontal flight, the extent to which an airplane is known to deviate from its assigned altitude depends primarily on the instrument errors. For many years extensive efforts have been made to improve the accuracy of the airborne instruments.

The altimeters which will be considered herein consist of the radio altimeter and the barometric altimeter. The radio altimeter is capable of measuring the height with a precision of about 150 feet (ref 16) with negligible lag in the information. The barometric altimeter, on the other hand, is modeled mathematically as a first order system with an uncertainty in the output of 500 feet.

It is desired to obtain an extremely accurate hybrid altimeter, by combining these altimeters with an inertial navigation system, one of the most accurate and precise systems in the field of instrumentation.

The inertial navigation system (I.N.S.) considered is a strapdown I.N.S. But in this particular class of navigator, it is sometimes necessary to compute the gravity field vector using the computed and an external indication of altitude. Some errors (in the gravity field calculation) are induced in the computation scheme (due to the mathematical model of this field vector) and although a solution has been found (ref 1) to bound the error, it has not been determined whether a more efficient way of mixing the two types of information would reduce the resultant error.

The availability on most commercial and military aircraft of an on-board computer permits the use of the Kalman filter theory to process the data for the hybrid altimeter. It is known that this system is the optimal linear filter in the sense of minimizing the mean squared error. The recursive form of the filter is very convenient, since a new optimum estimate can be made very shortly after each new measurement is obtained and it is never necessary to store a great amount of data.

It is to be understood, however, that in this thesis, only the errors in the optimum estimate of height will be studied.

2. Analysis of Gravitational Field Computation

2.1 Introduction

In a strapdown inertial navigation system, it is sometimes necessary (depending on the frame in which the computations are performed) to compute the gravitational field vector. When this computation is involved, it has been found (ref 1) that a problem of instability along the vertical channel arises. A computation scheme which avoids this difficulty is known (ref 1). This technique uses both the computed position and the external information concerning the magnitude of the position vector to calculate the gravitational field vector. The external information is used to insure stability of the vector magnitude while the computed position is used to give the direction of the vector field. This computation scheme, however, introduces additional errors in the computed position due to the uncertainties in the external altitude information. The purpose of this chapter is to determine how the external information must be weighted with respect to the computed information in order to minimize the system error.

2.2 Gravitational Field Computation

It is known (ref 3) that the gravitational force field vector is derivable from a scalar function called the gravitational potential. The conventional form of the external potential is given by eq 1.6, ref 13:

$$V(r,\phi) = \frac{Gm}{r} [1 - \sum_{k=2}^{\infty} J_k \left(\frac{r_{eg}}{r}\right)^k \underline{P}_k (\cos \phi)]$$

where:

$V(r,\phi)$ = the gravitational potential

$r = |\underline{r}|$ = magnitude of the position vector

m = mass of the earth

ϕ = colatitude

r_{eg} = equatorial radius of the earth

$\underline{P}_k (\cos \phi)$ = Legendre polynomials

G = gravitational constant

The values of the coefficients J_k must be empirically obtained by suitable experiments such as the observation of satellite orbits. This formula, however, is only valid for bodies with axial symmetry. The earth may be assumed to have axial symmetry for inertial navigational purposes.

If the vector operator $\underline{\nabla}$ is defined as:

$$\underline{\nabla} = \{\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}\}$$

then the gravitational field vector is given by:

$$\underline{G} = \underline{\nabla}\, V(r,\phi)$$

which can be expressed (see derivation ref 1) as:

$$\underline{G} = - \frac{KE}{r^3}\underline{r} \qquad (2.1)$$

with $E = Gm$

$K = 1 - \frac{3}{4} J_2 (1 - 3 \cos 2L)$ (for the earth)

This expression was obtained by neglecting all the J_k's of higher order than two. This is valid since, as shown in ref 3, the largest neglected term is of the order $10^{-6}|\underline{G}|$ and state-of-the-art accelerometers are not sensitive to such small values.

In order to minimize the system error, it has been decided to investigate the weighting of the external information (r_a) and the computed position magnitude (r_c). Calling the weighting factor α, the computed gravitational field vector can be expressed as:

$$\underline{G}_c\ (\alpha) = - \frac{KE}{r_a^{(3-\alpha)}\ r_c^{\alpha}}\ \underline{r}_c \qquad (2.2)$$

In actuality the external information consists of the altitude h_a. To get the estimated position, h_a is added to the local geocentric earth radius magnitude. It has been proven (ref 4) that determining the position vector in this way is accurate to about a foot, assuming of course, that the information of altitude has no error by itself.

2.3 Derivation of the Error Equations in Computing the Gravitational Field Vector

In this section, the error equations will be derived for the following error sources:

1. Altimeter uncertainty (Δh_a)
2. Error ($\Delta \underline{r}$) in the computed vector position ($\underline{r}_c$).

In the analysis which follows, and all which will be done in this thesis, the perturbation method is used to linearize the nonlinear equations (or nonlinear system differential equations). The perturbation method involves the substitution:

$$\underline{a}_c = \underline{a} + \underline{\Delta a}$$

where $\underline{a}_c$ = computed dependent variable
$\underline{a}$ = errorless dependent variable
$\Delta \underline{a}$ = error in computed dependent variable

When this substitution is made, linear equations (or linear differential equations) involving only the error quantities emerge. Products of error quantities, being of second order, are generally neglected:

Defining:

$$\underline{r}_c = \underline{r} + \underline{\Delta r}$$

$$r_a = r + \Delta h_a$$

Substituting the above values in eq 2.2 gives:

$$\underline{G}_c\ (\alpha) = - \frac{KE \quad (\underline{r} + \underline{\delta r})}{(r + \delta h_a)^{3-\alpha}\ [(\underline{r} + \underline{\delta r})^T\ (\underline{r} + \underline{\delta r})]^{\alpha/2}}$$

but:

$$[(\underline{r} + \underline{\delta r})^T\ (\underline{r} + \underline{\delta r})]^{-\alpha/2} \simeq r^{-\alpha}\ (1 - \alpha\frac{\underline{r}^T}{r^2} \cdot \underline{\delta r})$$

(δr^2 has been neglected since it is of second order)

Also:

$$(r + \delta h)^{-(3-\alpha)} \cong r^{-(3-\alpha)}\ [1-(3-\alpha)\frac{\delta h}{r}]$$

Thus:

$$\underline{G}_c(\alpha) \cong - \frac{E \cdot K}{r^\alpha\ r^{+(3-\alpha)}}\ (1 - \alpha\frac{\underline{r}^T \cdot \underline{\delta r}}{r^2})\ [1 - (3 - \alpha)\ \frac{\delta h}{r}](\underline{r} + \underline{\delta r})$$

$$\cong - \frac{EK}{r^3}\ [\underline{r} - \alpha\frac{\underline{r}^T \cdot \underline{\delta r}}{r^2} \cdot \underline{r} - (3 - \alpha)\ \frac{\delta h}{r}\ \underline{r} + \underline{\delta r}]$$

but:

$$- \frac{EK}{r^3} \cdot \underline{r} = \underline{G} \quad \text{and} \quad \frac{E}{r^2} = \omega_s^2 = \text{Schuler period}$$

then:

$$\underline{G}_c(\alpha) = \underline{G} + \omega_s^2\ \alpha\frac{\underline{r}^T \cdot \underline{\delta r}}{r^2}\ \underline{r} + (3 - \alpha)\omega_s^2\ \frac{\delta h_a}{r}\underline{r} - \omega_s^2\ \underline{\delta r} \qquad (2.3)$$

Where the second order products $J_2\ \underline{\delta r}$ and $J_2\ \delta h_a$ have been neglected.

As pointed out before, it is necessary to study the errors in the computed longitude and latitude that result from uncertainties in the computation of the gravitational field vector (eq 2.3). Since these errors depend on the frame which is chosen for the computation (geocentric inertial or geographic frame), the two following sections will study the propagation of the errors in these two computation frames.

2.4 Geocentric Inertial Computation Frame

2.41 Derivation of Error Equations

A strapdown system which computes in geocentric inertial coordinates will be analyzed in this section. The functional diagram for such a system is shown in fig (2.1).

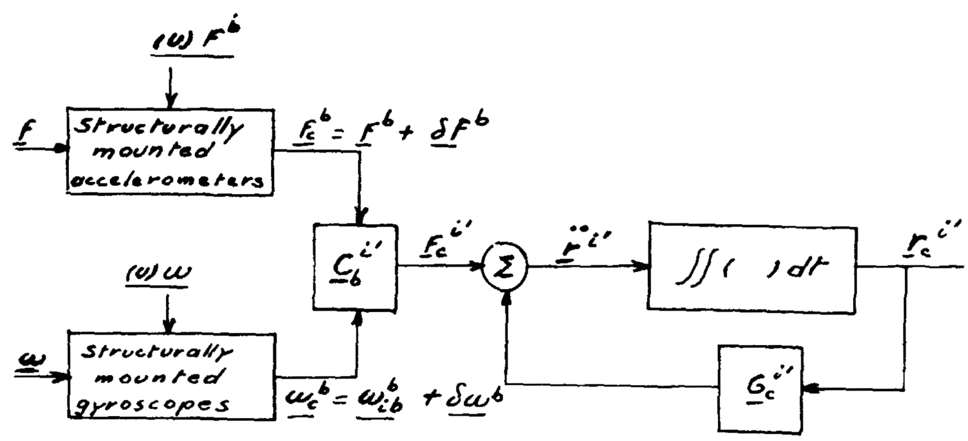


Fig 2.1

Strapdown system computing in geocentric inertial coordinates

Note that in fig 2.1 we have chosen to express the error in the transformation matrix as:

$$\underline{C}_b^{i'} = \underline{C}_i^{i'} \underline{C}_b^{i} = (\underline{I} - \underline{E}^{i}) \underline{C}_b^{i}$$

It is, of course, tacitly assumed that the attitude matrix $C_b^{i'}$ has been orthogonalitized through the application of the orthogonality relationship for coordinate transformation matrices:

$$\underline{C}_b^{i} (\underline{C}_b^{i})^T = \underline{I}$$

E^i has been defined as:

$$\underline{E}^i = \begin{bmatrix} 0 & -\varepsilon_z & \varepsilon_y \\ \varepsilon_z & 0 & -\varepsilon_x \\ -\varepsilon_y & \varepsilon_x & 0 \end{bmatrix}$$

Where the error angles, ε_k, result from positive rotations of the computed inertial axes relative to inertial axes.

Error Angle Computation:

The relationship between the inertial and navigational frames is given by:

$$\underline{r}^i = \underline{C}_n^i \, \underline{r}^n$$

Differentiating with respect to time:

$$\dot{\underline{r}}^i = \underline{C}_n^i \, \dot{r}^n + \dot{\underline{C}}_n^i \, \underline{r}^n = \underline{C}_n^i \, (\dot{\underline{r}}^n + \underline{C}_i^n \, \dot{\underline{C}}_n^i \, \underline{r}^n)$$

which is also equal, by the theorem of Coriolis:

$$\dot{\underline{r}}^i = \underline{C}_n^i \, (\dot{\underline{r}}^n + \underline{\Omega}_{in}^n \, \underline{r}^n)$$

$\underline{\Omega}$ is defined as the skew symmetric form of $\underline{\omega}$. From these two last equations, it is deduced:

$$\underline{C}_i^n \, \dot{\underline{C}}_n^i = \underline{\Omega}_{in}^n$$

$$\dot{\underline{C}}_n^i = \underline{C}_n^i \, \underline{\Omega}_{in}^n \qquad (2.4)$$

since

$$\underline{C}_b^{i'} = \underline{C}_i^{i'} \, \underline{C}_b^i$$

differentiation with respect to time yields:

$$\dot{\underline{C}}_b^{i'} = \dot{\underline{C}}_i^{i'} \underline{C}_b^i + \underline{C}_i^{i'} \dot{\underline{C}}_b^i$$

using eq 2.4

$$\dot{\underline{C}}_b^{i'} = \underline{C}_i^{i'} \underline{\Omega}_{i'i}^i \underline{C}_b^i + \underline{C}_i^{i'} \underline{C}_b^i \underline{\Omega}_{ib}^b \quad (2.5)$$

$\underline{\Omega}_{i'b}^b$ which updates the transformation matrix $\underline{C}_b^{i'}$ is given by the output of the gyroscopes.
The output of the gyro-triad is:

$$\underline{\omega}_{i'b}^b = \underline{\omega}_{ib}^b + \underline{\delta\omega}^b \qquad \text{or} \qquad \underline{\Omega}_{i'b}^b = \underline{\Omega}_{ib}^b + \underline{\delta\Omega}^b$$

Substituting into (2.5):

$$\dot{\underline{C}}_b^{i'} = \underline{C}_b^{i'} (\underline{\Omega}_{ib}^b + \underline{\delta\Omega}^b) \quad (2.6)$$

Equating eq (2.4) and eq (2.6):

$$\underline{C}_b^{i'} \underline{\Omega}_{ib}^b + \underline{C}_b^{i'} \underline{\delta\Omega}^b = \underline{C}_i^{i'} \underline{\Omega}_{i'i}^i \underline{C}_b^i + \underline{C}_i^{i'} \underline{C}_b^i \underline{\Omega}_{ib}^b$$

After premultiplying succesively by $\underline{C}_{i'}^i$, and $\underline{C}_i^b$, the following expression is obtained:

$$\underline{\delta\omega}^b = \underline{C}_i^b \underline{\omega}_{i'i}^i$$

From the previous definition $\underline{\omega}_{i'i}^i = - \dot{\varepsilon}^i$, then:

$$\underline{\delta\omega}^b = \underline{C}_i^b \underline{\omega}_{i'i}^i = - \underline{C}_i^b \dot{\underline{\varepsilon}}^i \quad (2.7)$$

It is assumed in this study that the sole error source for $\underline{\delta\omega}^b$ is the gyro drift; thus:

$$\underline{\delta\omega}^b = \underline{(u)\omega}^b$$

Eq (2.7) becomes: $\dot{\underline{\varepsilon}}^i = - \underline{C}_b^i \underline{\delta\omega}^b = - \underline{C}_b^i \underline{(u)\omega}^b$

Assuming $\underline{\varepsilon(o)} = \underline{o}$, $\underline{\varepsilon}^i$ is:

$$\underline{\varepsilon}^i = -\int_o^t \underline{C}_b^i \, \underline{(u)\omega}^b \, dt \tag{2.8}$$

Error Equations:

The output of the accelerometers, suitably coordinatized, are equal to the nonfield specific force coordinatized in geocentric inertial frame:

$$\underline{\ddot{r}}_c^{i'} = \underline{f}^{i'} + \underline{G}_c^{i'} \tag{2.9}$$

where $\underline{r}_c^{i'} = \underline{r}^i + \underline{\delta r}^i$

It must be noted that within a first order analysis, the error vector $\underline{\delta r}$ can be seen as coordinatized in either the inertial or computed inertial frame since:

$$\underline{\delta r}^i = \underline{C}_{i'}^i \, \underline{\delta r}^{i'} = (\underline{I} + \underline{E}^i) \, \underline{\delta r}^{i'} \simeq \underline{\delta r}^{i'}$$

where the second order product $\underline{E}^i \, \underline{\delta r}^i$ has been neglected. From eq (2.3) and eq (2.9), the following expression is obtained:

$$\underline{\ddot{r}}^i + \underline{\delta\ddot{r}}^i = \underline{f}^i + \underline{\delta f}^i - \underline{E}^i \underline{f}^i + \underline{G}^i + \omega_s^2 \alpha \cdot \frac{\underline{r}^T \cdot \underline{\delta r}}{r^2} \cdot \underline{r}^i$$

$$+ (3-\alpha)\omega_s^2 \frac{\delta h}{r} \underline{r}^i - \omega_s^2 \, \underline{\delta r}^i$$

since: $\underline{\ddot{r}}^i = \underline{f}^i + \underline{G}^i$

then:

$$\underline{\delta\ddot{r}}^i + \omega_s^2 \, \underline{\delta r}^i - \omega_s^2 \alpha \frac{\underline{r}^{i^T} \cdot \underline{\delta r}^i}{r^2} \cdot \underline{r}^i = \underline{\delta f}^i - E^i f^i + (3-\alpha)\omega_s^2 \frac{\delta h}{r} \cdot \underline{r}^i$$

but:

$$\frac{\underline{r}^{i^T} \cdot \underline{\delta r}^i}{r^2} \cdot \underline{r}^i = \frac{1}{r^2} \begin{bmatrix} r_x^2 & r_x r_y & r_x r_z \\ r_x r_y & r_y^2 & r_y r_z \\ r_x r_z & r_y r_z & r_z^2 \end{bmatrix} \underline{\delta r}^i$$

and: $\underline{r}^i = \{r \cos L \cos\lambda,\ r \cos L \sin\lambda,\ r \sin L\}$

The equation for $\underline{\delta r}^i$ is now given by:

$$\underline{B}\ \underline{\delta r}^i = \underline{\delta f}^i - \underline{E}^i \underline{f}^i + (3-\alpha)\omega_s^2 \frac{\delta h}{r} \cdot \underline{r}^i \tag{2.10}$$

where:

$$\underline{B} = \begin{bmatrix} p^2+\omega_s^2(1-\alpha)\frac{r_x^2}{r^2} & -\omega_s^2\alpha\frac{r_x r_y}{r^2} & -\omega_s^2\alpha\frac{r_x r_z}{r^2} \\ -\omega_s^2\alpha\frac{r_x r_y}{r^2} & p^2+\omega_s^2(1-\alpha)\frac{r_y^2}{r^2} & -\omega_s^2\alpha\frac{r_y r_z}{r^2} \\ -\omega_s^2\alpha\frac{r_x r_y}{r^2} & -\omega_s^2\alpha\frac{r_y r_z}{r^2} & p^2+\omega_s^2(1-\alpha)\frac{r_z^2}{r^2} \end{bmatrix}$$

It is now possible to determine the values of α for which the system is unstable by examining the determinant of the coefficient matrix $\underline{B}$.

After some manipulations, the determinant of $\underline{B}$ is expressed as:

$$|\underline{B}| = (p^2 + \omega_s^2)^2 \; [p^2 + \omega_s^2(1-\alpha)] \qquad (2.11)$$

It is seen that the system characteristic equation has poles in the right half plane if $\alpha > 1$, giving rise to exponential growth.

If $\alpha = 1$, the system equation gives rise to linear growth. To get a bounded error, it is seen that α must lie in the range:

$$\alpha < 1 \qquad (2.12)$$

where α can take on negative values.

Notice also that the only choice of α which uncouples the equations is $\alpha = 0$. However, (ref 1) the error induced by the altimeter error for this choice of α is quite large. Although a proper choice of α can decrease the altimeter error sensitivity, the question of what happens to the other error sensitivities remains.

Because of the coupled time-varying nature of the equations, computer solution is the only practical way of solving the system equations. In order to limit the number of trials on the computer using different values of α, the Laplace transformation techniques are used to obtain a "first cut" on the system behavior.

2.411 Error Equations Due to Altimeter Error

Because equation (2.10) is a linear differential equation, it is possible to study the effect of the different error sources one at a time, invoking the principle of superposition to combine the results (if L is a linear transformation L(x+y) = L(x) + L(y)). Equation (2.10) gives the resultant error due to altimeter uncertainty:

$$\underline{B}\; \underline{\delta r}^i = (3-\alpha)\omega_s^2 \frac{\delta h_a}{r} \cdot \underline{r}^i \qquad (2.13)$$

where $\underline{B}$ is defined by equation (2.10)

Since now, the stationary case will be studied. It allows to do some comparisons with some known results.

$$L = 45^\circ \quad \text{(a constant)}$$

$$\dot{\ell} = 0$$

$$\dot{\lambda} = \omega_{ie}$$

Taking the Laplace transform of equation (2.13), and considering only the behavior of the x channel, after some manipulations, the following expression is obtained:

$$L(\delta r_x) = \frac{(3-\alpha)\omega_s^2 \cos L \ \delta h_a}{[s^2 + \omega_s^2(1-\alpha)][s^2 + \omega_{ie}^2]}$$

The inverse of the Laplace transform gives δr_x as:

$$\delta r_x = \frac{(3-\alpha)\omega_s^2 \cos L}{\omega_s^2(1-\alpha) - \omega_{ie}^2} \frac{1}{\omega_{ie}} \sin\omega_{ie} t$$

$$- \frac{1}{\omega_{ie}^2[\omega_s^2(1-\alpha) - \omega_{ie}^2]} \sin\omega_s\sqrt{1-\alpha}\, t \quad \delta h_a$$

By the same method it is found that:

$$\delta r_y = (3-\alpha)\omega_s^2\omega_{ie}\cos L \ \frac{1}{\omega_{ie}^2\omega_s^2(1-\alpha)}$$

$$- \frac{1}{\omega_{ie}^2[\omega_s^2(1-\alpha)-\omega_{ie}^2]}\cos\omega_{ie} t - \frac{1}{\omega_s^2(1-\alpha)[\omega_{ie}^2-\omega_s^2(1-\alpha)]} \ \delta h_a$$

$$\delta r_z = \frac{(3-\alpha)\,\omega_s{}^2 \sin L}{\omega_s{}^2(1-\alpha)}\,(1-\cos\omega_s\sqrt{1-\alpha}\;t)\;\delta h_a$$

This study being conducted only to get some indication of the behavior of the system, it is perfectly admissible to do the approximation:

$$\omega_{ie}{}^2 << \omega_s{}^2.$$

In order to reduce the magnitude of the error vector, it appears that the equation $\frac{3-\alpha}{1-\alpha}$ must be minimized subject to the constraint $\alpha < 1$.

The function $A = \frac{3-\alpha}{1-\alpha}$ is plotted on graph 29. It is seen that α should be chosen as small as possible to reduce A. However, for $\alpha < -10$ the gain in the reduction of A is negligible. It must be noted that this is only an indication because changing α also changes the frequency of the response, and therefore leads to different ways of adding the components in δr_x, δr_y, δr_z.

2.412 Error Equations Due to Gyro Drift

Equation (2.10) gives: $\underline{B}\;\underline{\delta r}^i = -\underline{E}^i \underline{f}^i$

or:

$$\underline{f}^i = \underline{C}_n^i \underline{f}^n = \underline{C}_n^i \begin{bmatrix} o \\ o \\ -g \end{bmatrix} = g \begin{bmatrix} \cos L \cos\lambda \\ \cos L \sin\lambda \\ \sin L \end{bmatrix} = g\frac{\underline{r}^i}{r} - g\,D \begin{bmatrix} \sin L \cos\lambda \\ \sin L \sin\lambda \\ -\cos L \end{bmatrix}$$

where D, the deviation of the normal, is defined as:

$D = L - Lg$.

Since the maximum value of the deviation of the normal is of the order of the earth's ellipticity (1/297), the product of the deviation of the normal and the elements of the drift matrix are second order.

Also:

$$\frac{g}{r} = \omega_s^2 \ ; \quad \underline{f}^i = \omega_s^2 \ \underline{r}^i$$

Thus:

$$\underline{B} \ \underline{\delta r}^i = -\omega_s^2 \ \underline{E}^i \underline{r}^i$$

$\underline{E}^i$ being the skew symmetric form of $\underline{\varepsilon}^i$, $\underline{\varepsilon}^i$ must now be cal-culated.

Developping eq (2.8):

$$\dot{\underline{\varepsilon}}^i = \begin{bmatrix} -\sin L \cos\omega_{ie}t\omega_x - \sin\omega_{ie}t\omega_y - \cos L \cos\omega_{ie}t\omega_z \\ -\sin L \sin\omega_{ie}t\omega_x + \cos\omega_{ie}t\omega_y - \cos L \sin\lambda\omega_z \\ \cos L \ \omega_x - \sin L \ \omega_z \end{bmatrix}$$

Integrating

$$\underline{\varepsilon}^i = \begin{bmatrix} -\sin L\frac{\omega_x}{\omega_{ie}} \sin\omega_{ie}t + \frac{\omega_y}{\omega_{ie}}(\cos\omega_{ie}t-1) - \cos L\frac{\omega_z}{\omega_{ie}} \sin\omega_{ie}t \\ \sin L\frac{\omega_x}{\omega_{ie}}(\cos\omega_{ie}t-1) + \frac{\omega_y}{\omega_{ie}}\sin\omega_{ie}t - \frac{\omega_z}{\omega_{ie}}(\cos\omega_{ie}t-1) \\ \cos L \ \omega_x t - \sin L \ \omega_z t \end{bmatrix}$$

Which will be denoted:

$$\underline{\varepsilon}^i = \{\varepsilon_1 \ , \varepsilon_2, \ \varepsilon_3\}$$

Hand calculations, as done in the preceding section, lead to a very long formula from which no clear conclusion can be drawn.

From the two above sections, it is seen that α must be tried in the range $-10 \leq \alpha < 1$ for the error induced by the altimeter uncertainty. For the error induced by the gyros drift, it will be necessary to observe how the system error behaves when α is changed.

2.5 Geographic Computation Frame

The functional diagram for such a system is shown in Fig. 2.2

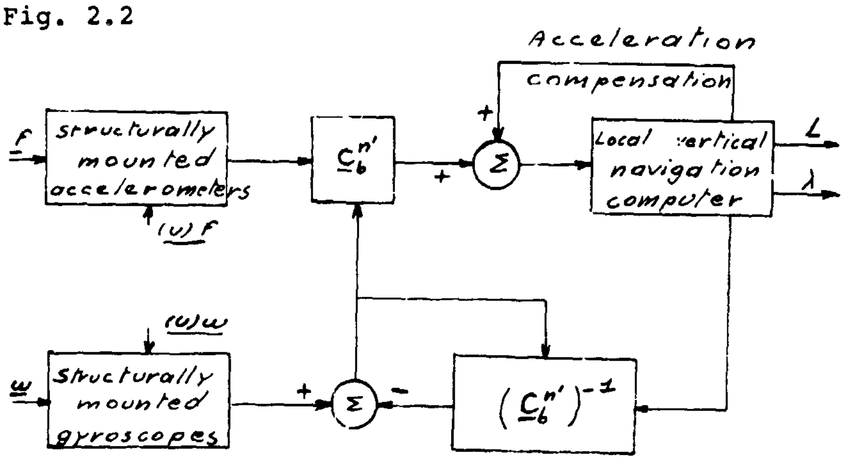


Fig 2.2

Strapdown system computing in geographic coordinates

Since the error analysis for this configuration has been performed in ref 5 (p. 130) it will not be repeated here. It is shown that $\underline{G}$ does not have to be computed, and so this preliminary study of the effect of $\underline{G}_c(\alpha)$ on δL, $\delta\lambda$ and δh is unecessary. Altitude information can be extracted from an independent channel, which will be done later.

2.6 Computer Program

The differential equation system error (2.10) is numerically solved for a six hour period. But the only existing programs for solving differential equations are for first order. Using the state vector approach, it is easy to convert eq (2.10) into a first order vector differential equation.

Eq (9) is expressed as:

$$\underline{\delta\ddot{r}}^i + \underline{C}\ \underline{\delta r}^i = -\omega_s^2\ \underline{E}^i\underline{r}^i + (3-\alpha)\,\omega_s^2\ \delta h_b \cdot \frac{\underline{r}_i}{r}$$

where $\underline{C}$ represents

$$\begin{bmatrix} \omega_s^2(1-\alpha\cos^2 L\cos^2\omega_{ie}t), & -\alpha\omega_s^2\cos^2 L\frac{\sin 2\omega_{ie}t}{2}, & -\omega_s^2\alpha\frac{\sin 2L}{2}\cos\omega_{ie}t \\ -\alpha\omega_s^2\cos^2 L\frac{\sin 2\omega_{ie}t}{2}, & \omega_s^2(1-\alpha\cos^2 L\sin^2\omega_{ie}t), & -\omega_s^2\alpha\frac{\sin 2L}{2}\sin\omega_{ie}t \\ -\alpha\omega_s^2\frac{\sin 2L}{2}\cos\omega_{ie}t, & -\alpha\omega_s^2\frac{\sin 2L}{2}\sin\omega_{ie}t, & \omega_s^2(1-\alpha\sin^2 L) \end{bmatrix}$$

and

$$\underline{E}^i\underline{r}^i = \begin{bmatrix} r(+\omega 3\sin\omega_{ie}t\cos L\ \ \omega 2\sin L) \\ r(\omega 1\sin L-\omega 3\cos L\cos\omega_{ie}t) \\ r\ \cos L(\omega 2\cos\omega_{ie}t-\omega 1\sin\omega_{ie}t) \end{bmatrix}$$

and

$$\frac{\underline{r}_i}{r} = \begin{bmatrix} \cos L\ \cos\omega_{ie}t \\ \cos L\ \sin\omega_{ie}t \\ \sin L \end{bmatrix}$$

If the following variables are defined:

$$\underline{Y} = \begin{bmatrix} \underline{\delta r} \\ \underline{\delta\dot{r}} \end{bmatrix}\ ;\quad \underline{Y} = [y(1),\ \ldots\ ,\ y(6)]$$

eq (9) can now be rewritten as:

$$\dot{\underline{Y}} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ & & & 0 & 0 & 0 \\ & -\underline{C} & & 0 & 0 & 0 \\ & & & 0 & 0 & 0 \end{bmatrix} \underline{Y} + (3-\alpha)\,\omega_s{}^2\,\delta h_b \begin{bmatrix} 0 \\ 0 \\ 0 \\ \cos L \cos\omega_{ie}t \\ \cos L \sin\omega_{ie}t \\ \sin L \end{bmatrix} +$$

$$+ \begin{bmatrix} 0 \\ 0 \\ 0 \\ r(+\omega 3 \sin\omega_{ie}t \cos L \omega 2 \sin L) \\ r(\omega 1 \sin L - \omega 3 \cos L \cos\omega_{ie}t) \\ r \cos L(\omega 2 \cos\omega_{ie}t - \omega 1 \sin\omega_{ie}t) \end{bmatrix}$$

This equation is of the type:

$$\dot{\underline{Y}}(t) = \underline{A}(t)\underline{Y}(t) + \underline{F}(t)$$

which can be solved by a subroutine using the Hamming predictor corrector method. (Program in appendix C, n $\doteq$ 1). It is assumed that at time 0:

$$\underline{\delta r}^{i}(o) = \underline{0}, \ \underline{\delta \dot{r}}^{i}_{(o)} = \underline{0}$$

The computation will be carried with:

$$\delta h_b = 1000 \text{ ft} \ ; \quad (\upsilon)\omega_k = 1 \text{ mer}\upsilon.$$

The program produces the root mean square value of $\delta L, \delta\lambda$, and δh due to δh_b. The ergodic property is assumed; under this condition, the R.M.S. is: (ref 2)

R.M.S. = $\sqrt{\frac{1}{T} \int_{o}^{T} x^2 (t) dt}$

It also produces the root sum square value of δL and $\delta\lambda$ due to $\underline{\delta\omega}$.

R.S.S. = $\sqrt{a_{\omega x}^2 + a_{\omega y}^2 + a_{\omega z}^2}$

and the outputs δL, $\delta\lambda$ and δh due to altimeter uncertainty.

2.7 Results

The computer program (no. 1) used to get the results of this paragraph is given in Appendix C. It was worked out on an I.B.M. 360 in the M.I.T. Computation Center.

Before exposing the results, it is better to give some explanations about the curves of appendix B.

Graphs 1 to 15 represent ΔL, $\Delta\lambda$, Δh due to Δh_b (altimeter uncertainty) for $\alpha = -10, -1, -0.5, 0., 0.5$. Graphs 16 to 21 represent the R.S.S. value of ΔL, $\Delta\lambda$ due to $\underline{(u)\omega}$ for $\alpha = -10, 0., 0.5$. Graphs 22 to 24 represent the R.M.S. value of ΔL, $\Delta\lambda$, Δh due to Δh_b for $\alpha = -10, -1., -0.5, 0., 0.5$.

For the R.S.S. value of δL and $\delta\lambda$ (due to $\underline{\delta\omega}$) it is seen that: there is no noticeable change when α changes from $\alpha = -10$ to $\alpha = +0.5$.

For the R.M.S. value of δL, $\delta\lambda$, δh (due to δh_b) it is seen that: as α approaches -10., the R.M.S. value of δL and $\delta\lambda$ increases (roughly doubles) while the R.M.S. value of δh decreases (roughly from 3 to 1) when α is changed from 0 to - 10.

The chart below gives some indication of the behavior of δL, $\delta\lambda$, δh due to δh_b:

α	δL max min	δλ max min	Time after which error in inertially computed altitude becomes $\geq h_b$ - min -
0.5	0.18	0.33	11
0	0.2	0.53	11
-0.5	0.18	0.30	11
-1	0.17	0.30	9.5
-10	0.37	0.80	6

2.8 Conclusion

It appears clearly from the results that only a compromise solution can be deduced since some errors grow while some others decrease for a change in α. For the errors studied, the best compromise appears to be: $\alpha = 0$. If $\alpha > 0$, the R.M.S. value of the error in inertially computed altitude grows excessively large. If $\alpha = -0.5$, no improvement in δL is achieved, while a slight decrease in δλ and in the R.M.S. value of the error in the inertially computed altitude is observed. On the other hand, the time after which the error in computed altitude is larger than the altimeter uncertainty is the same as for $\alpha = 0$ and the on board computer has many more operations to perform than for $\alpha = 0$. In addition, when the error due to gyro drift is considered, the improvement on the total error is insignificant. If $\alpha = -10$, the R.M.S. values of δL and $\delta\lambda$ due to the altimeter uncertainty are excessive.

3. Principles of inertial measurement height

3.1 Introduction

Inertial navigation techniques have been successfully used in the indication of position. In this chapter, inertial techniques for height measurement will be studied.

3.2 The Fundamental Model of Measurement Height

To measure height, the vertical acceleration given by the strapdown inertial system has to be integrated twice. But with such a system, any uncertainty in the accelerometers would give rise to a displacement error increasing with the square of the time. A proposed scheme (see ref. 7) is to bound the system using another source of altitude, in this case, barometric height. Fig. 3.1 shows the essential features.

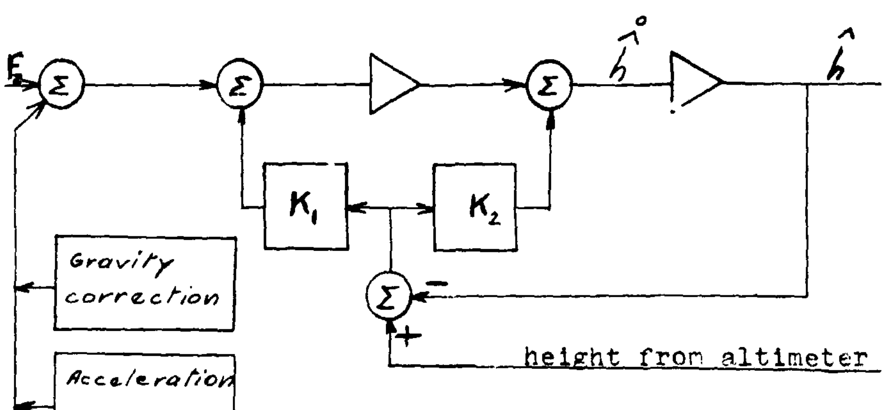


fig 3.1

Fundamental model of measurement height

3.2.1 Derivation of the Specific Force

The specific force on the z axis is

$$F_z^n = G_z^n - (Pi^2\overline{R_{EP}})_z^n$$

but $(Pi^2\overline{R_{EP}})^n$ is developed in ref 4:

$$(Pi^2\overline{R_{EP}})_z^n = -(2\mathring{R}\mathring{L}_c + R\overset{\circ\circ}{L}_c)\sin D + (R\mathring{L}_c^2 - \overset{\circ\circ}{R})\cos D + R$$

$$(\omega_{ie}^2 + 2\omega_{ie}\mathring{\ell} + \mathring{\ell}^2).\cos L_c \cos L$$

$$\approx R\mathring{L}_c^2 - \overset{\circ\circ}{R} + R\omega_{ie}^2 \cos L_c \cos L + R(2\omega_{ie}\mathring{\ell} + \mathring{\ell}^2)\cos L_c \cos L -$$

$$(2\overset{\circ\circ}{R}L_c + R\overset{\circ\circ}{L}_c)\ e \sin 2L$$

when the following approximations are made (ref 4)

$\cos D \simeq 1$

$\sin D \simeq e \sin 2L$

The specific force takes on the following expression:

$$F_z^n = [G_z^n - (Ro+h)\,\omega_{ie}^2 \cos L_c \cos L)] + \overset{\circ\circ}{h} - (Ro+h)$$

$$(2\omega_{ie}\mathring{\ell} + \mathring{\ell}^2)\cos L_c \cos L + [2\overset{\circ\circ}{R}L_c + (Ro+h)\overset{\circ\circ}{L}_c]e \sin 2L -$$

$$2e \cos 2L \frac{V_e^2}{a} - a\, e \sin 2L\, \overset{\circ\circ}{L} \quad (3.1)$$

where: $R = Ro + h$

$Ro = a\,(1 - e \sin^2 L)$

a = equatorial radius of the earth

$$\overset{\circ}{R} = - eV_E \sin 2L + \overset{\circ}{h}$$

$$\overset{\circ\circ}{R} = 2e \cos 2L \frac{V_E^2}{a} - ae \sin 2L \overset{\circ\circ}{L} + \overset{\circ\circ}{h}$$

$$V_E = R \cos L_c \overset{\circ}{\ell}$$

But H einskanen (see ref 18) has shown that:

$$Gz - (Ro+h) \cos L_c \cos L\, \omega_{ie}^2 = g_e \left[1 + (5/2\, m - e) \sin^2 L - 2 \frac{h}{Re} \right] = g_h$$

where:

g_h = value of the gravity at any point (including the centrifugal term due to earth's rotation)

g_e = value of gravity at sea level at equator

m = ratio of centrifugal to gravitational acceleration at equator (1/ 290)

e = ellipticity of the earth

Eg (3.1) becomes:

$F_z{}^n = \overset{\circ\circ}{h} + g_h$ - acceleration terms

As pointed out in ref (7), the gravity term will vary by 0.27 per cent for a latitude change of 45° and the height term by 0.67 per cent from sea level to 70,000 feet. The acceleration corrections can be 1.4 per cent of one g with an aircraft flying at mach 2 on an easterly course at low latitude. If this is compared with an accelerometer bias of 0.01 per cent g, it is seen that the terms, although important, do not have to be computed to a particularly high order of acaracy. This is an important point because the corrections terms are computed from the informations of the strapdown inertial navigation systems which contain some small errors.

One of these errors is the misalignment of the computed reference frame with respect to the vertical. The error in computing the gravity term will be unimportant because the gravity is modified as the **cosine** of the misalignment angle but the output of the accelerometers will contain a component of the lateral acceleration. It is necessary to compute how large the misalignment angle can be before the component of lateral acceleration **exceeds the accelerometer's uncertainty (g/10,000.) .** Assuming that the airplane has a maximum lateral acceleration of 0.5 g, the maximum permissible misalignment angle is

$$g / 10{,}000 \geq 0.5 \cdot \pi \, \theta / (180 \times 60) \quad (\theta \text{ misalignment angle in } \widehat{\text{min}})$$

$$\theta \leq 0.69 \ \widehat{\text{min}}$$

If the accelerometer uncertainty is g / 1000, this misalignment can be as large as 6.9 $\widehat{\text{min}}$.

3.3 Survey of altitude measuring methods

A brief discussion on altitude measurement techniques is presented in order to compare the various techniques for measuring height. The following resume is largely inspired from ref (16) (" Survey of altitude-measuring methods for the vertical separation of aircraft".).

It would appear that among the numerous techniques available; acceleration-measuring instruments, capacitance altimeters, magnetometers, pressure-measuring instruments, density altimeters, cosmic-ray altimeters, radar and radio altimeters, and sonic altimeters; only two classes of instruments are technically suitable. They are the class of pressure-measurement altimeters and the class of radar (or radio) altimeters.

3.31 Pressure-Measuring altimeters

The measurement of height is based on the variation of atmospheric pressure with height. However, for any type of pressure-measurement instrument the accuracy of the altitude measurement will depend not only on the accuracy of the instrument but also on the accuracy with which freestream static pressure is developed by the static-pressure sensor. The response of the instrument depends also on the speed of propagation of the pressure pulse through the connecting tubing. The mathematical model for such an instrument can be well approximated by a first order system with the time constant depending on the length of the connecting tubing and with an additive noise at the output due to the error induced by the static pressure sensor.

3.32 Radio and Radar Altimeters

The measurement of height is accomplished by transmitting a radio frequency wave from one antenna on the aircraft and receiving the reflected wave from the earth by a receiving antenna. The practical utility of the radar (or radio) altimeter, however, presupposes a relatively level terrain; the use of this type of altimeter, therefore, is generally restricted to over the ocean applications. This is not a drastic restriction, because the flight at 70,000 feet will be at supersonic speed and will not be allowed over ground areas because of noise considerations. This type of altimeter can be mathematically modeled as instantaneously giving the height plus some noise.

3.4 Error Analysis

From the discussion of sect 3.2, the following model can be assumed: see fig. 3.2

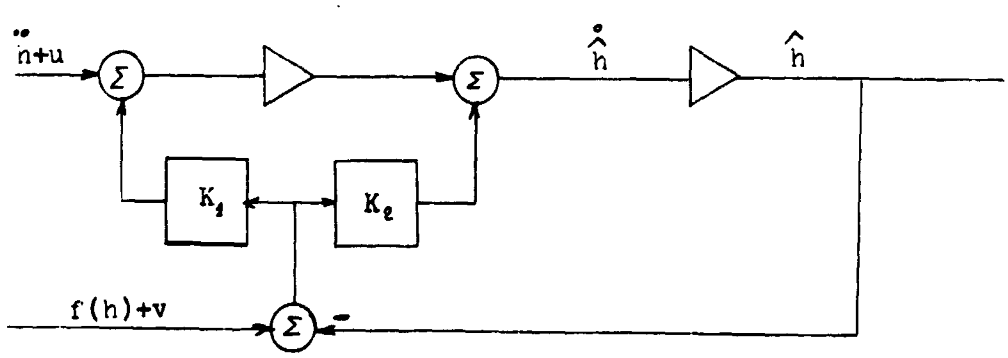


fig 3.2
Model of inertial measurement of height

where:

u: uncertainty in the accelerometers

v: uncertainty in the internal information of height.

It is desired to study the behavior of the system errors and to choose the two constants K_1 and K_2 in such a way to minimize the error in the indication of the estimate of altitude (h).

3.41 No Lag in the External Information of Height

The external source can be either a radar or a radio altimeter which gives the true height plus an error which is assumed, for this preliminary study, to belong to an ensemble of constant functions, similar to the accelerometer uncertainty.

The mean value of u is o and the mean squared value is $\overline{u_o^2}$. $E(v) = 0$, $\overline{v(t)^2} = \overline{v_o^2}$.

The system obeys the differential equation:

$$\overset{\circ}{\hat{h}} = \frac{1}{s}\left[s^2 h + u \right] - \frac{K_1}{s}\left[\hat{h} - h - v\right] - K_2\left[\hat{h} - h - v \right]$$

Defining the error in the estimated height as $\epsilon = \hat{h} - h$, there results:

$$\epsilon = \frac{u(s)}{s^2+K_2 s+K_1} + \frac{K_1 + K_2 s}{s^2 + K_2 s + K_1} v(s) \text{ or}$$

$$\varepsilon = \frac{(e^{r_1 t} - e^{r_2 t})\,u(t)}{r_1 - r_2} + \frac{e^{r_1 t}(r_2K_2+K_2) - (r_2K_2+K_2)\,e^{r_2 t}}{r_1 - r_2}\; v(t)$$

where :

$$r_1 = \frac{-K_2 + \sqrt{K_2^2 - 4\,K_1}}{2}$$

$$r_2 = \frac{-K_2 - \sqrt{K_2^2 - 4\,K_1}}{2}$$

Using the results of appendix A and the fact that $E(v) = E(u) = 0$, the steady state mean squared value of the error is given as :

$$\overline{\varepsilon^2(t)} = \frac{\overline{u_0^2}}{K_1^2} + \overline{v_0^2}$$

From this result , it is clear that it is not possible to reduce the system error to a value less than the radar error . At best , by choosing $K_1^2 \gg u_0^2$, the error in indicating the height will not be greater than the error of the radar .

3.42 Lag in the external information of height

The barometric altimeter is modeled as a first order system ; its output is :

$$h_b = \frac{h(s)}{\zeta s + 1} + v$$

where v belongs to an ensemble of constant functions . The differential equation governing the system is :

$$\mathring{\hat{h}} = \frac{1}{s}[\,s^2 h + u\,] - \frac{K_1}{s}\left[\,\hat{h} - \frac{h}{\zeta s + 1} - v\,\right] - \left[\,\hat{h} - \frac{h}{\zeta s + 1}\right.$$

$- v] K_2$

AS in the preceeding section the error is defined as :
$\varepsilon = \hat{h} - h$, so the error of the system is :

$$\varepsilon(s) = -\frac{K_1 s + K_2 s^2}{(\zeta s + 1)(s^2+K_2 s+K_1)} \zeta\, h(s) + \frac{u(s)}{(s^2+K_2 s+K_1)} +$$

$$\frac{K_1 + K_2 s}{(s^2+K_2 s+K_1)} v(s)$$

It can be seen that for the steady state case (h=cst) the error will be the same as for the case of no lag in the external information of height . Using the fact that u and v are zero mean variables , the mean squared value of the error will be :

$$\overline{\varepsilon^2(t)} = \overline{v_0^2} + \frac{\overline{u_0^2}}{K_1^2} + \text{error due to lag}$$

TO compute the error due to lag , the autocorrelation function for the altitude is assumed to be :

$$\Phi_{hh} = \overline{h^2} \frac{e^{-\lambda|\zeta|}}{\lambda^2 - s^2}$$

where $\frac{1}{\lambda} = 10$ seconds and $\overline{h^2} = 50\ ft^2$. These numbers correspond to a helicopter configuration of flight . Because h does not belong to an ensemble of constant functions , it is possible to compute $\overline{\varepsilon_h^2}$ in the following way :

$\varepsilon_h = \int_{-\infty}^{t} W(t-\zeta)h(\zeta)d\zeta$ = error due to change in altitude

$\Phi_{\varepsilon\varepsilon}(s) = W(s)W(-s)\Phi_{hh}(s)$

$$\overline{\varepsilon_h^2} = \frac{1}{2\pi j} \int_{-j\infty}^{+j\infty} W(s)W(-s)\Phi_{hh}(s)ds$$

(from eq 5.3.12 and 5.3.13 in ref 2)

After some algebric manipulations and the use of the tables in ref 8 which give the closed form solution for these types of integral , the total mean squared value of the error is :

$$\overline{\varepsilon^2} = \frac{\overline{u_0^2}}{K_1^2} + \overline{v_0^2} + 2\zeta\overline{h^2}\frac{K_2^3K_1\lambda^2\zeta + K_2^2K_1^2(\zeta\lambda+1) + K_1^3\lambda\zeta(\zeta\lambda+1) + K_1^3K_2\lambda\zeta^2}{(\zeta\lambda+1)[-2K_1K_2\lambda\zeta+K_2\lambda^2+K_1^2K_2\zeta^2]}$$

The values of K_1 and K_2 are to be determined such that they minimize the mean squared valed value of the error . It must be remembered that K_1 and K_2 have to be positive to ensure a bounded error . The minimum of $\overline{\varepsilon^2}$ will be , at best , equal to zero .

The problem of finding the coordinates of the minimum of a positive function is the same as finding the coordinates of the minimum of the square of the same function . Finding the minimum of the function squared facilitates the computer search . The search program , using a subroutine finding the unconstrained minimum of a function of several variables , is based on the Davidon (ref 15) method . Once the coordinates are found , the variance $\overline{\varepsilon^2}$, is computed (program No. 2 , appendix C) .

Results

The noises have been taken as :

$\overline{u_0^2} = 50\ ft^2$ and $\overline{v_0^2} = (g/100.)^2$

And two time constants for the barometric altimeter have been examined : $\zeta = 1s.$ and $\zeta = 10s.$

It was found that for $\zeta = 10s.$,

$$K_1 = 0.1 \ , \ K_2 = 0.3 \ , \ \varepsilon^2 = 65.2 \text{ feet}^2$$

While for $\zeta = 1s.$,

$$K_1 = 0.4 \ , \ K_2 = 0.8 \ , \ \varepsilon^2 = 80.5 \text{ feet}^2$$

The resulting error is larger than that of the barometric altimeter . At best , in an horizontal flying path , the error in the indication of height will be equal to the error of the barometric altimeter .

3.5 Conclusion

It is clear that this fundamental model of inertial measurement of height has a limited range of application . The barometric altimeter gives low frequency information on the change of height while the I.N.S.

gives high frequency information . Thus the resultant information possesse a wide bandwidth . Still , the accuracy of the steady state estimate of height is not improved over that of the barometric altimeter alone (or radar altimeter)

It is thus necessary to use the information from two different kinds of altimeter in order to get the possibility of improving the accuracy of the steady state estimate of height (actually , what is needed is any one altimeter with zero bias) .

Also , a more elaborate scheme of filtering is needed to extract the best estimate of altitude in a dynamic situation . The Kalman filter is the best method in a mean squared sense for linear filtering and the following chapter presents the results which are needed in this thesis . It must be noted that in the stationary case , the results of the Kalman theory are equivalent to those of the Wiener filter . The Kalman approach avoids the necessity of making approximations in obtaining the realizable filter . For the Wiener approach , mathematical approximations must be used to obtain a physically realizable filter from the theorical formula .

4. The Kalman Filtering Theory

Principals results and special cases .

Only those theorical results which are needed in this paper are presented . Their derivations are not reproduced and can be found in the listed references .

4.1 Linear filtering theory

The model of the message process is given by a linear dynamical system excited by white noise:

$$\dot{\underline{x}}(t) = \underline{F}(t)\ \underline{x}(t) + \underline{G}(t)\ \underline{u}(t) \qquad (4.1)$$

where $\underline{x}(t)$ is the state vector of dimension n ; its coordinates x are the state variables .
The observed signal contains an additive white noise:

$$\underline{m}(t) = \underline{H}(t)\ \underline{x}(t) + \underline{v}(t) \qquad (4.2)$$

$\underline{u}$ and $\underline{v}$ are white noise (Gaussian random processes) with zero means and covariances :

$$\underline{R}_u(t) = E(\ \underline{u}(t)\ \underline{u}'(s)\ ) = \underline{Q}(t)\ \delta(t-s) \qquad (4.3)$$

$$\underline{R}_v(t) = E(\ \underline{v}(t)\ \underline{v}'(s)\ ) = \underline{R}(t)\ \delta(t-s) \qquad (4.4)$$

$\delta(s)$ is the delta function .
The estimation problem consists of computing a function of $\underline{m}$ which in some manner approximates $\underline{x}$.
This function is denoted by $\hat{\underline{x}}$ .The estimation is

denoted by : $\tilde{\underline{x}} = \underline{x} - \hat{\underline{x}}$.

The function $\hat{\underline{x}}$ is computed in such a way as to minimize the variance of each component of $\tilde{\underline{x}}$.

4.11 Continuous filtering

If the measurement is taken continuously , Kalman (ref 9) has proven that the canonical form of the filter is :

$$\mathring{\hat{\underline{x}}} = \underline{F}\,\hat{\underline{x}} + \underline{K}\,(\,\underline{m} - \underline{H}\,\hat{\underline{x}}\,) \qquad (4.5)$$

where the optimal gain is :

$$\underline{K} = \underline{P}\,\underline{H}'\underline{R}^{-1} \qquad (4.6)$$

with $\underline{P} = \text{cov}(\,\underline{x}(t/t),\underline{x}(t/t)\,)$

$\underline{P}$ is the solution of the matrix differential equation:

$$\mathring{\underline{P}} = \underline{F}\,\underline{P} + \underline{P}\,\underline{F}' - \underline{P}\,\underline{H}'\,\underline{R}^{-1}\underline{H}\,\underline{P} + \underline{G}\,\underline{Q}\,\underline{G}' \qquad (4.7)$$

4.12 Discrete filtering

Usually it is desirable to compute the optimal gains as they are needed in real time with the on board computer . A change in the model or a change in the number of measurement devices may necessitate a change in the optimal gains which can be accomplished most expediently if the sequences are computed in real time . The measurements are taken only at discrete times . In a second paper , Kalman (ref 10) has shown that the system has the following behavior :

- at the time of the measurement , the estimate is updated with the equations:

$$\hat{\underline{x}} = \hat{\underline{x}}^{b} + \underline{K} (\underline{m} - \underline{H} \hat{\underline{x}}^{b}) \qquad (4.8)$$

where the superscript b indicates conditions that exist before the measurement .

$$\underline{K} = \underline{P}^{b} \underline{H}' (\underline{H} \underline{P} \underline{H}' + \underline{R})^{-1} \qquad (4.9)$$

The covariance matrix is updated at the time of the measurement by the equation :

$$\dot{\hat{\underline{x}}} = \underline{F} \hat{\underline{x}} \qquad (4.11)$$

and the covariance equation by :

$$\dot{\underline{P}} = \underline{F} \underline{P} + \underline{P} \underline{F}' + \underline{Q} \qquad (4.12)$$

If the system is stationary ($\underline{F}$ constant) and the sampling interval uniform , Kalman has shown (ref 10) that it is convenient to use the transition matrix ($\dot{\underline{\Phi}} = \underline{F} \underline{\Phi}$; $\underline{\Phi}_0 = \underline{I}$) . The state is then extrapolated betwen the measurements by :

$$\hat{\underline{x}}(t_1) = \underline{\Phi}(t_1, t_0) \hat{\underline{x}}(t_0) \qquad (4.13)$$

and $\underline{P}$ by :

$$\underline{P}(t_1) = \underline{\Phi}(t_1, t_0) \underline{P}(t_0) \underline{\Phi}'(t_1, t_0) + \int_{t_0}^{t_1} \underline{\Phi}(t_1, \tau) \underline{Q} \underline{\Phi}'(t_1, \tau) d\tau \qquad (4.14)$$

These two equations will be used by the computer program which simulates the filter .
The block diagram for the discrete case is shown in fig 4.1 .

4.2 Measurement contamined by colored noise

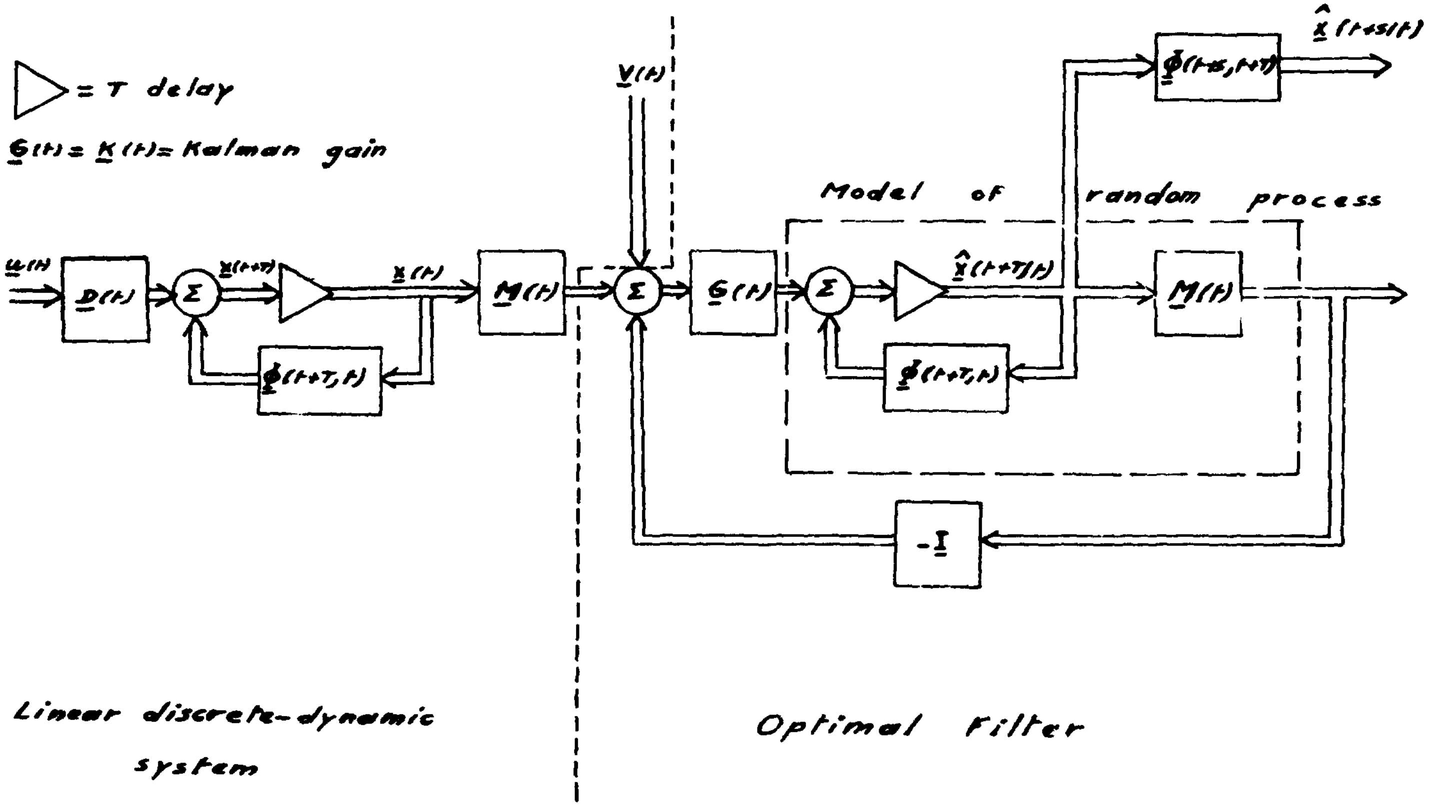
Practical systems exist in which the correlation times of the random measurement errors are not short compared to times of interest in the system (colored noise) . Also , a measurement may be so accurate that it is reasonable to assume it contains no error . These two cases are singular problems of the Kalman theory . Often , it is possible to simulate the colored noise by a linear dynamical system excited by white noise . The colored noise becomes part of an augmented state variable vector and the measurements contain a linear combination of the new state variables .The system appears mathematically as a system with perfect measurements .

In this section , the ensemble of constant functions is the colored measurement noise .This special noise can be generated by :

$\dot{x} = 0$ or in the discrete case $x_{m+1} = x_m$

Bryson and Johansen (ref 11) have found the solution of this singular problem in the discrete case and in the continuous case . They have shown that in the

Fig 4.1



Block diagram of the Kalman Filter for the discrete-case discrete case , by regarding all measurement noise variables as additional state variables , it is possible to use the method presented by Kalman (ref 10) for filtering in the discrete case . Discrete filtering being the only possible solution for aircraft application (limited capacity of the on board computer) , the continuous case will not be considered .

4.3 The filtering problem for hybrid altimetry

From the measurement of the vertical acceleration , and the two measurements of the height, it is desired to get the best estimate of the altitude The measurements are contamined by noise :

- white noise corrupts the indication of the position radar .
- white noise is added to the output of the accelerometers .
- ensemble of constant functions is added in the indication of the barometric altimeter

It is necessary to define a mathematical model representing as accurately as possible the physical system. Only steady-state cruise conditions are studied . During non steady-state cruise conditions , the vertical maneuver accelerations would have to be considered completly random . But from ref 12 , it is known that if the maneuver accelerations are included as completely random signals , a linear filter makes almost no

improvement in the performance of a navigation system over that of a deterministic system .
When the mathematical model is obtained , it is then possible to apply the results of the optimum linear filtering theory .

5. Optimal estimation of altitude f om strapdown I.N.S. , barometric altimetry and radar data

5.1 Introduction

In chapter 3 , it was shown that two different kinds of altimeter (different in the sense of having complementary error statistics) are needed in order to be able to improve the accuracy of the altitude estimation over that obtainable through the use of only one altimeter . In this section , a model simulating the dynamics of the aircraft and the measuring instruments is developed . The results of chapter 4 are then used in order to find what accuracy can be achieved in the estimation of height . Furthermore , only the errors in the optimal estimate are to be studied ; consequently the computer program solves only the variance equation .

5.2 Model of the filter

The model simulates the vertical accelerations encountered by an aircraft flying at 60,000 feet in a horizontal path at a speed of mach 2 , elevator locked in position . Under these conditions , it is known (ref 13) that airplanes have one long period lightly damped mode (called phugoid mode) , and one with a short period and relatively heavy damping . The short period mode will not be considered since all the transients die out in a very short period of time during steady state cruise conditions . Ref 13 shows that the airplane may be modeled as a second order system whose natural frequency is the phugoid frequency and which is forced by gusts of wind at the acceleration level . The gusts of wind usually encountered by an aircraft (ref 14) can be modeled as a first order system . The representation of the model is now rather straightforward (fig 5.1) . Following the error analysis of chapter 3 , the error introduced by the vertical error angle of the I.N.S. is included in the accelerometer uncertainty . The state variables are :

X_1 : output of the barometric altimeter without the noise

X_2 : variation of altitude of the aircraft around the horizontal flying path

X_3 : vertical speed of the plane

X_4 : gusts of wind (ft/sec^2)

X_5 : unknown ensemble of constant functions

X_6 : altitude of the theorical horizontal flying path

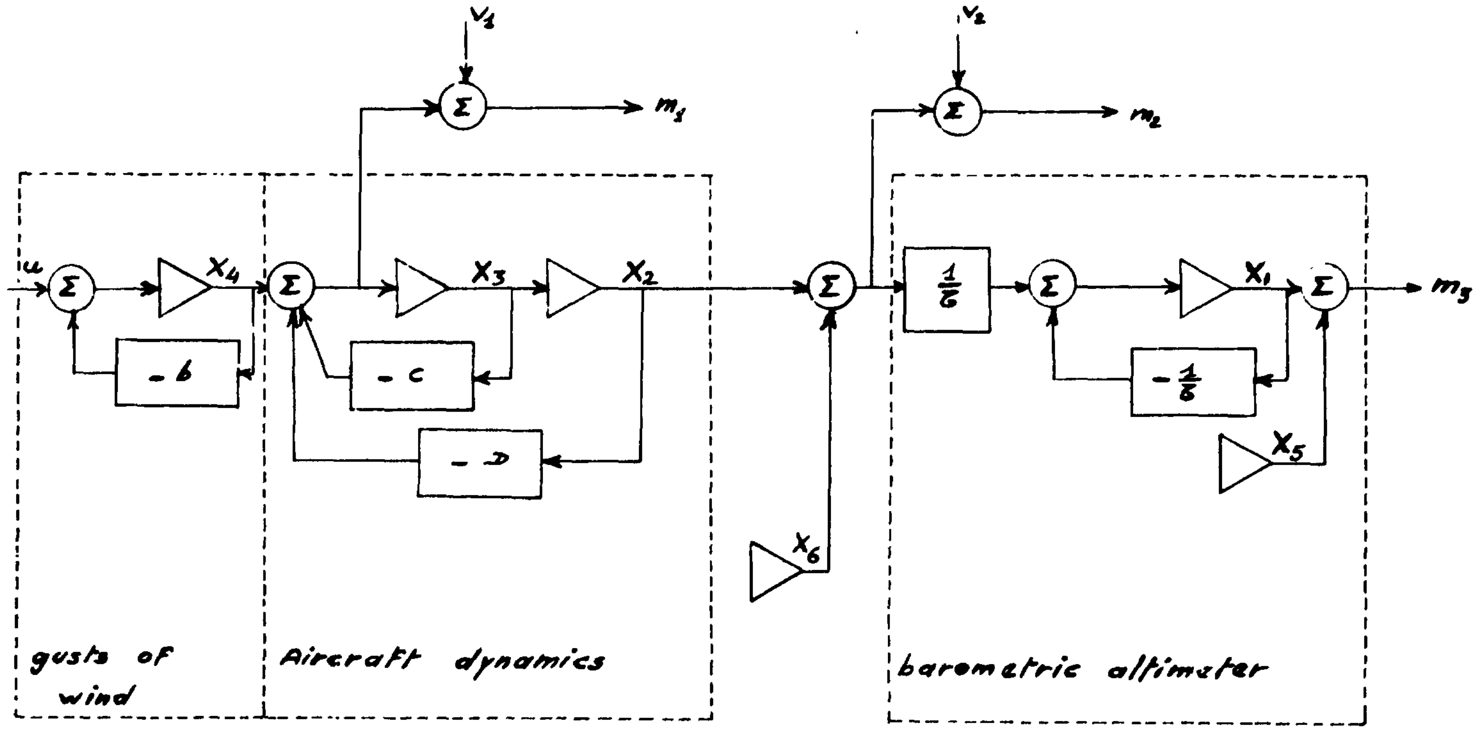
$X_2 + X_6$: true altitude

For an S.S.T. flying at mach 2 at 60,000 ft , the period of the phugoid can be assumed : T = 10 minits and the damping ratio : $\xi = 0.1$. Then :

$$d = \omega^2 = \frac{4\pi^2}{(10 \times 60)^2} \quad ; \quad c = 2\xi \frac{2}{10 \times 60}$$

The value of b = 0.25 is given by ref 14 and the value of the driving noise (white) is choosen such that the R.M.S. value of X_2 is 500 ft^2 :

Model of the system dynamics for the Kalman Filter



m_1 = accelerometer output

m_2 = radio-altimeter output

m_3 = barometric altimeter output

Fig 5.1

$$\overline{X_2^2} = 500 = \frac{1}{2\pi j}\int_{-j\infty}^{+j\infty} \left| \frac{1}{\tau_e s + 1} \times \frac{1}{s^2 + cs + d} \right|^2 \overline{u^2}\, ds$$

This choice of u which seems arbitrary is done in order to ensure that the amplitude of the phugoid mode is not unrealistically small . If the simulated amplitude of the phugoid were unrealistic , the simulated vertical acceleration would then be small and some wrong conclusions could be drawn about the necessity of having vertical acceleration information . The state variables equations are :

$$\begin{bmatrix} \dot{X}_1 \\ \dot{X}_2 \\ \dot{X}_3 \\ \dot{X}_4 \\ \dot{X}_5 \\ \dot{X}_6 \end{bmatrix} = \begin{bmatrix} -1/\tau & 1/\tau & 0 & 0 & 0 & -1/\tau \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -d & -c & 1 & 0 & 0 \\ 0 & 0 & 0 & -b & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \\ X_5 \\ X_6 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ u \\ 0 \\ 0 \end{bmatrix} \qquad (5.1)$$

or : $\dot{\underline{x}} = \underline{F}\ \underline{x} + \underline{G}\ \underline{u}$

The observed signal is :

$\underline{m} = \underline{H}\ \underline{x} + \underline{v}$

$$\begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} = \begin{bmatrix} 0 & -d & -c & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} \underline{x} + \begin{bmatrix} u_1 \\ 0 \\ v_1 \end{bmatrix} \qquad (5.2)$$

In the case where less than three different measurements are taken , the line corresponding to the missing measurement must be skipped in eq 5.2 . Equation 5.1 remains unchanged in this situation .

5.3 Filter equations

As mentioned before , the discrete filter (the only one realizable) will be studied . From the results of chapter 4 , the equations for this filter are :

dynamical system :

$\dot{\underline{x}} = \underline{F}\ \underline{x} + \underline{G}\ \underline{u}$ (given by eq 5.1)

observed signal :

$\underline{m} = \underline{H}\ \underline{x} + \underline{v}$ (given by eq 5.2)

Between the measurements , the state is extrapolated by eq 4.13 . The sampling time being constant and the dynamical system constant :

$\underline{\Phi}(t+T,t) = \underline{\Phi}(t+T-t) = \underline{\Phi}(T) = cst$

$\underline{x}(t+T) = \underline{\Phi}(T)\ \underline{x}(t)$

The transition matrix is given by :

$\underline{\Phi} = L^{-1}(\ \underline{I}\ s - \underline{F}\ )$ (this matrix is explicitcly written out on program No. 3 , appendix c)

The variance equation obeys equation 4.14 .

At a measurement :

the variance equation is updated by eq (4.10) .

the Kalman gain is updated by eq (4.9) .

the state is updated by eq (4.8) .

The results of eq 4.10 and 4.8 become initial conditions for eq 4.14 and 4.13 until the next measurement is taken .

5.4 Computer program

The computer program (program No. 3 , appendix C) solves , as previously mentionned , only the variance equation . The covariance of the error of the best estimate of the true altitude ($x_2 + x_6$) must be computed from the covariance matrix , $\underline{P} = \operatorname{cov} \underline{\tilde{x}}$. The covariance of the estimation error ($\underline{\tilde{x}} = \underline{x} - \underline{\hat{x}}$) is denoted by cov $\underline{\tilde{x}}$. Kalman has shown than the covariance equation for the error is the same as the covariance equation for the state .

$$\operatorname{cov}(\widetilde{x_2 + x_6}) = \overline{\tilde{x}_2^2 + \tilde{x}_6^2 + 2\tilde{x}_2\tilde{x}_6} - (\overline{\tilde{x}_2 + \tilde{x}_6})^2$$

$$= \overline{\tilde{x}_2^2} + \overline{\tilde{x}_6^2} + 2\overline{\tilde{x}_2\tilde{x}_6} - \bar{\tilde{x}}_2^2 - 2\bar{\tilde{x}}_2\bar{\tilde{x}}_6 - \bar{\tilde{x}}_6^2$$

$$= (\overline{\tilde{x}_2^2} - \bar{\tilde{x}}_2^2) + (\overline{\tilde{x}_6^2} - \bar{\tilde{x}}_6^2) + (\overline{\tilde{x}_2\tilde{x}_6} - \bar{\tilde{x}}_2\bar{\tilde{x}}_6) \times 2$$

$$= \operatorname{cov} \tilde{x}_2 + \operatorname{cov} \tilde{x}_6 + 2 \operatorname{cov}(\tilde{x}_2\tilde{x}_6)$$

These three values are elements of the covariance matrix

P . The behavior of the P matrix is stored on punched cards and plotted versus time using the subroutine " PICTUR " stored in the computer memory .

5.5 Results

The computer program (No. 3) used to get the results is given in appendix C .

The operating time studied has been 2.5 seconds. (From a practical point of view , the operating time must be at most , one fourth of the smallest time constant of the transition matrix , which in this case is 10 seconds .)

The white noise value of the radio altimeter (ref 17) has been taken to be 150 feet . The mean squared value of the ensemble of constant function uncertainty of the barometric altimeter was (500 feet $)^2$ and the time constant 10 seconds .

Two values for the uncertainty in the output of the accelerometers have been studied (g/1000. and g/10,000.) . It is to be remembered (from chapter 3) that the accelerometer uncertainty includes the effect of the error angle in tracking the vertical .

In appendix B , all the graphs represent the R.M.H. (root mean squared error - in feet - in altitude indication) as functions of time in seconds . Graph No. 25 gives the R.M.H. when only the radio altimeter and the barometric altimeter are available .

Graph NO. 26 gives the R.M.H. for the same case at a larger scale for a short period of time in order to clarify the short period behavior in graph No. 25 . Graph No. 27 and No. 28 give the R.M.H. when the radio altimeter , the barometric altimeter and the strapdown I.N.S. are available for the two values of accelerometer uncertainty .

It can be seen that between the measurements , in the case where the I.N.S. is not available , the R.M.H. reaches a steady state value of 131 feet in 2.5 seconds ; at the measurement instant, the R.M.H. value is 24 feet .

Also , when all the three measurements are available , the R.M.H. for the two accelerometer uncertainty values is exactly the same (12 feet) when the steady state conditions are reached . This means (from chapter 3) that the misalignment angle can be as large as 7 $\widehat{\text{min}}$ (under 0.5 g lateral acceleration) without significant degradation in the estimation of altitude .

5.6 Conclusions

1-If only one external source of information is available for use with the I.N.S. , the altitude can not be estimated with an accuracy better than the accuracy of the external aid (from chapter 3) .

2-Optimal mixing of the information from two different altimeters(no inertial information available) leads to an error in the estimation of height which grows (in 2.5 seconds) during the operating time up to the accuracy of the more accurate altimeter .

3-Mixing the information of the two altimeters with inertial information leads to a small error in the estimation of altitude (12 feet at 60,000 feet) which does not degrade during the operating time . The strapdown I.N.S. vertical error angle can be as large as 7 $\widehat{\text{min}}$ (for lateral acceleration of the airplane up to 0.5 g) without degrading the accuracy of the estimation of altitude .

In summary , a hybrid altimeter using a strapdown I.N.S. needs two external indications having different error statistics of height in order to improve the accuracy of the estimation of altitude with respect to the accuracy of the external aid . The strapdown I.N.S. need not be very accurate by itself .

APPENDIX A

STATISTICS OF THE ENSEMBLE OF CONSTANT FUNCTIONS (ref 6)

The random process under consideration consists of the ensemble of constant functions with mean squared value x_0^2 . The process is stationary since it does not vary with time but non-ergodic since there is no representative member . The autocorrelation function will also be constant at the mean squared value since once a particular constant is choosen , it remains fixed for all times . The above random process is assumed to be the input to a system which is presumed to be unexcited until t = 0 . In general , the output of a linear system in response to an input $x(\tau)$ applied at $t = \tau$ is :

$$e(t) = \int_{-\infty}^{t} W(t,\tau)x(\tau)d\tau \qquad \text{ref 2}$$

where $W(t,\tau)$ is the unit impulse response at time t after application at $t = \tau$. $W(t,\tau) = 0$ for $t<\tau$ for physically realizable systems .

The autocorrelation of the output is :

$$e(t_1)e(t_2) = \int_{-\infty}^{t_1} W(t_1,\tau_1)d\tau_1 \int_{-\infty}^{t_2} W(t_2,\tau_2)d\tau_2 \; x(\tau_1)x(\tau_2)$$

where it was possible to make an ensemble average under the integral signs since the weighting functions are independent of the probability distribution functions .

But : $x(\tau_1)x(\tau_2) = \phi_{xx}(\tau_1,\tau_2) = \phi_{xx}(\tau) = x_0^2$

and : $e(t_1)e(t_2) = \phi_{ee}(t_1,t_2) = x_0^2\int_{-\infty}^{t_1}W(t_1,\tau_1)d\tau_1\int_{-\infty}^{t_2}W(t_2,\tau_2)d\tau_2$

Finally if the system is at rest prior $t = 0$,

$\phi_{ee}(t_1,t_2) = x_0^2\int_0^{t_1}W(t_1,\tau_1)d\tau_1\int_0^{t_2} W(t_2,\tau_2)d\tau_2$

The mean squared error at time t is given by :

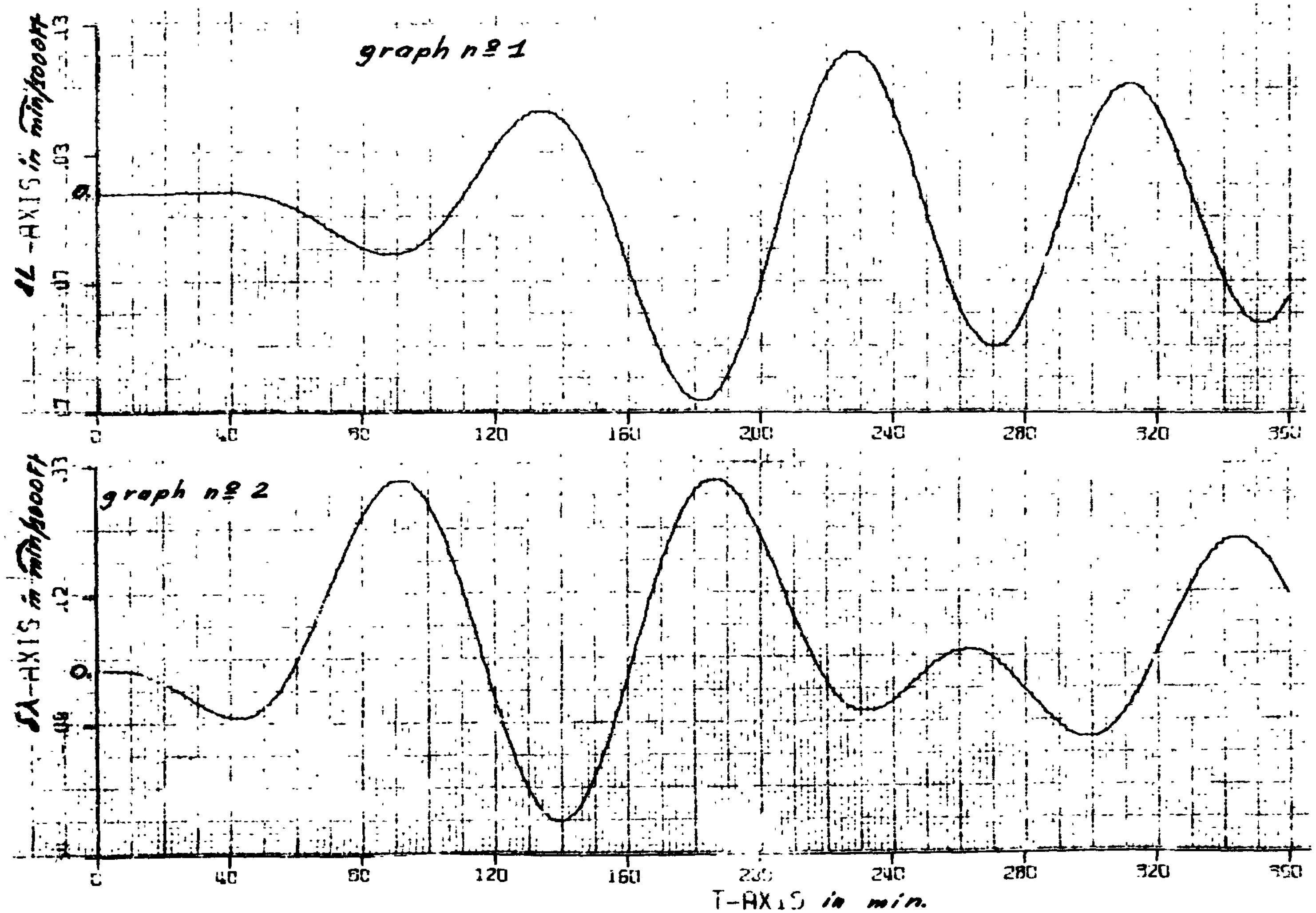
$\phi_{ee}(t,t) = x_0^2\int_0^t d\tau_1W(t,\tau_1) \int_0^t d\tau_2W(t,\tau_2)$ or

$e^2(t) = x_0^2 \int_0^t\int_0^t W(t,\tau_1)W(t,\tau_2)d\tau_1d\tau_2$

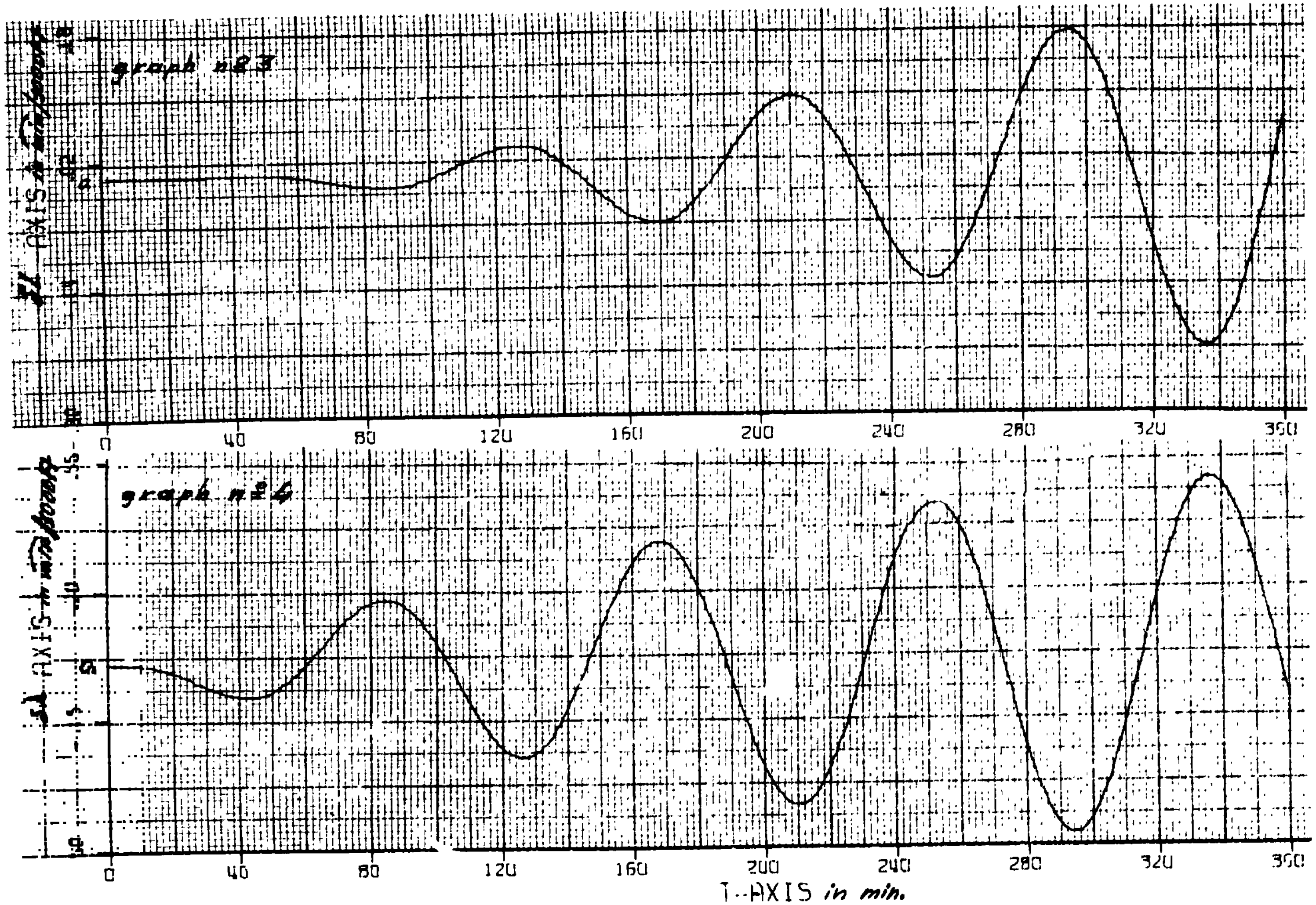
If the system under consideration is describable by constant coefficients :

$$W(t,\tau) \rightarrow W(t-\tau) \text{ , thus :}$$

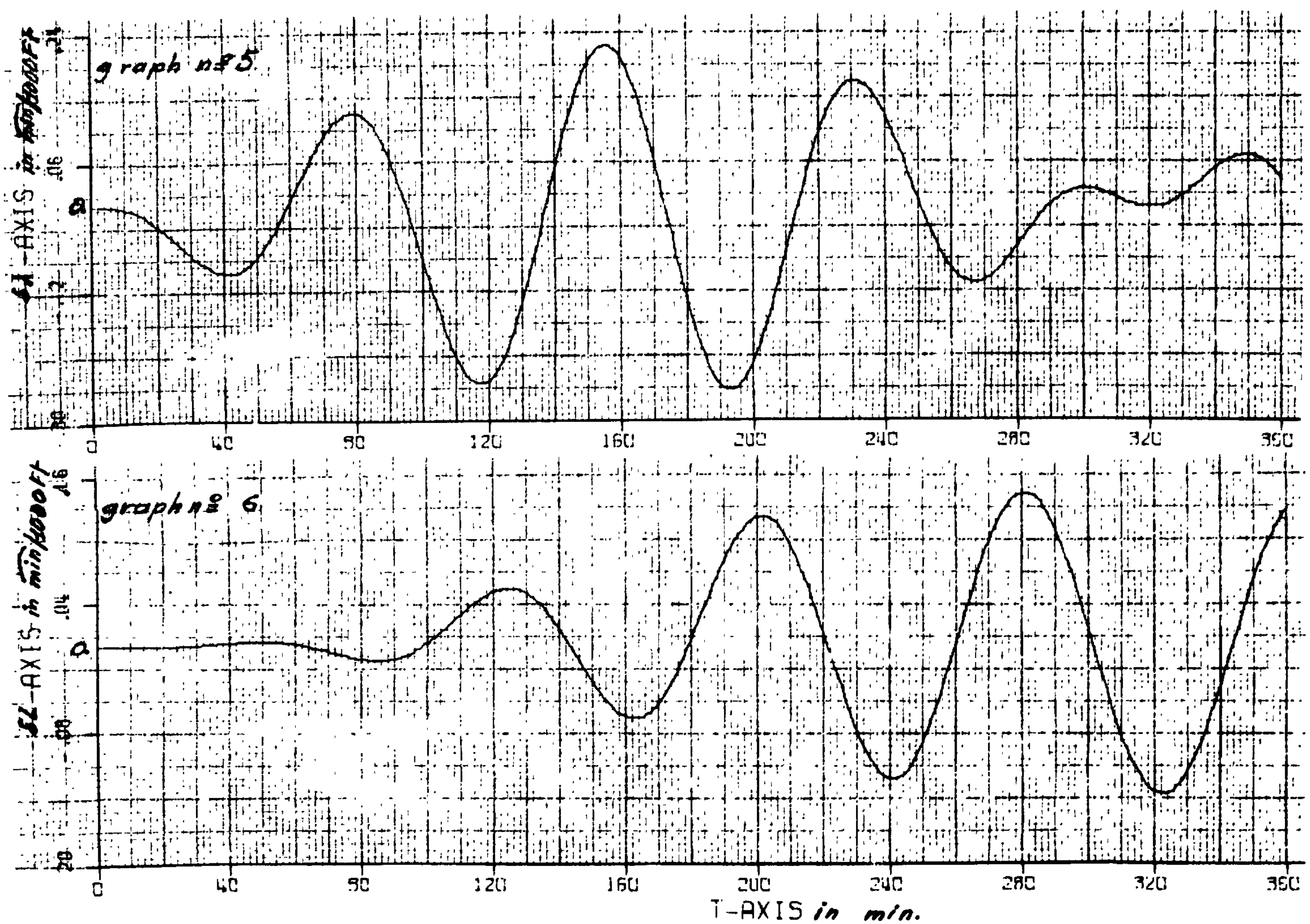
$e^2(t) = x_0^2\int_0^t\int_0^t W(t-\tau_1)W(t-\tau_2)d\tau_1d\tau_2$



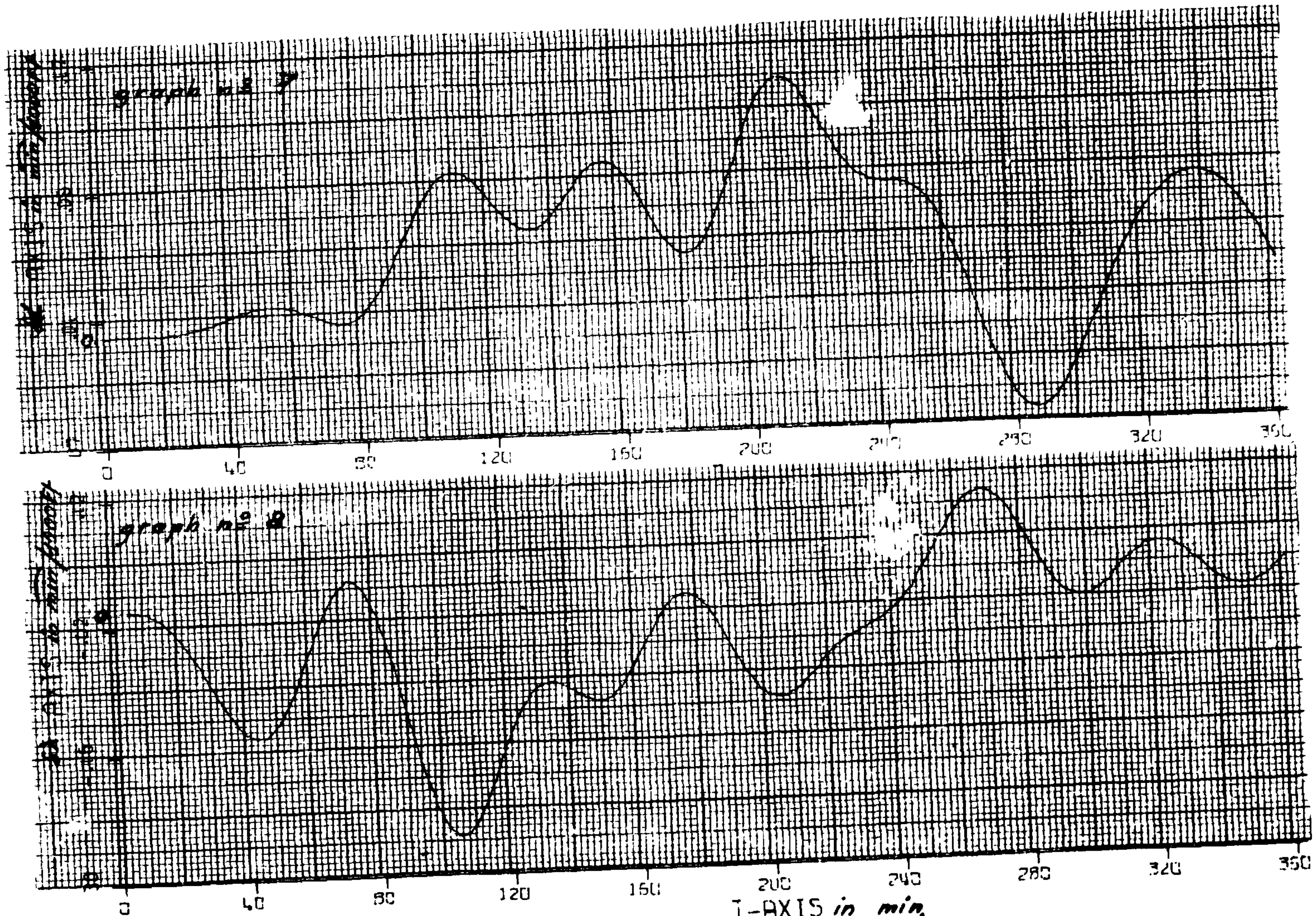
Navigation system errors due to altimeter uncertainty – $\alpha = +0.5$; $L = 45°$



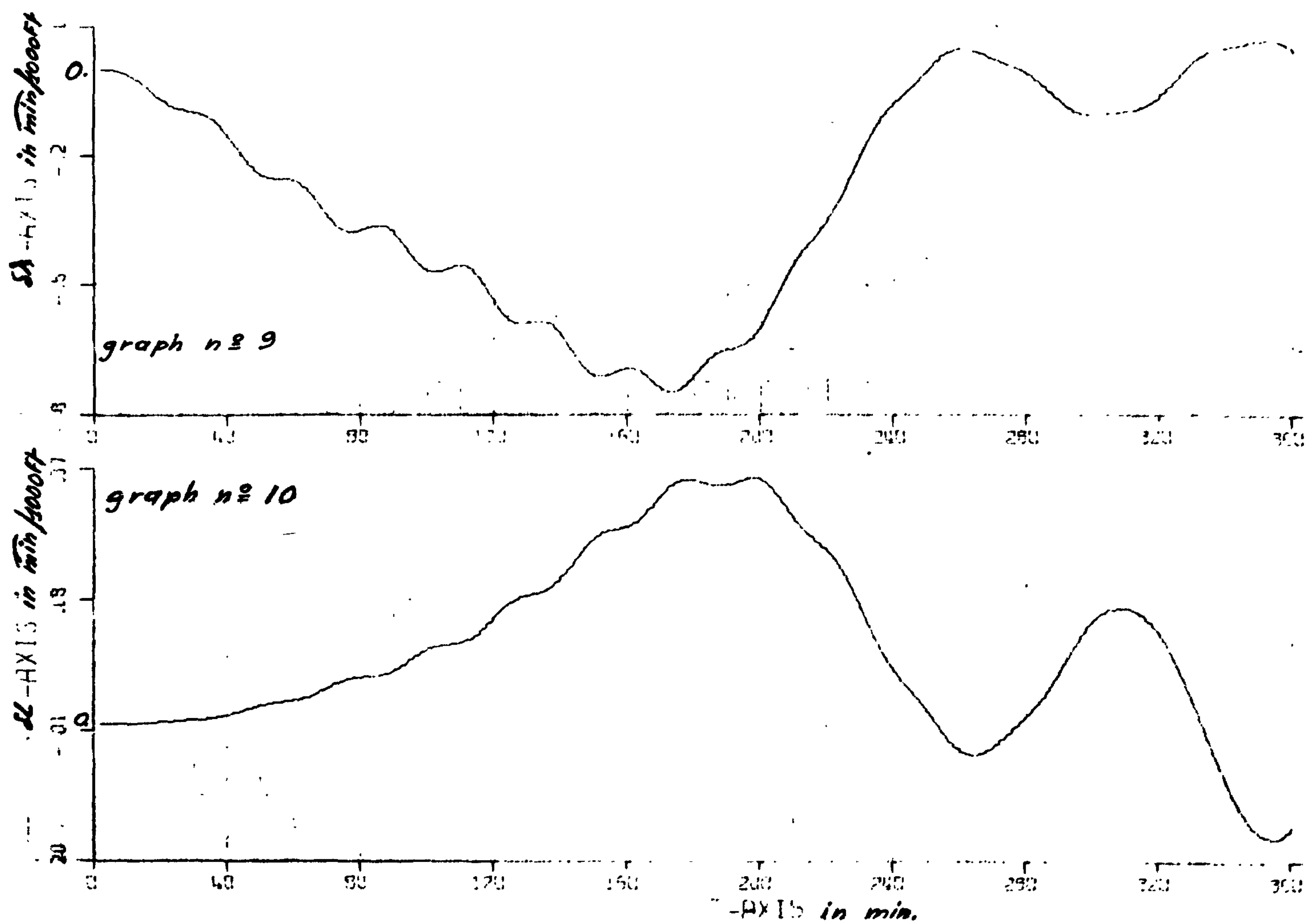
Navigation system errors due to altimeter uncertainty – $\alpha = 0$; $L = 45°$



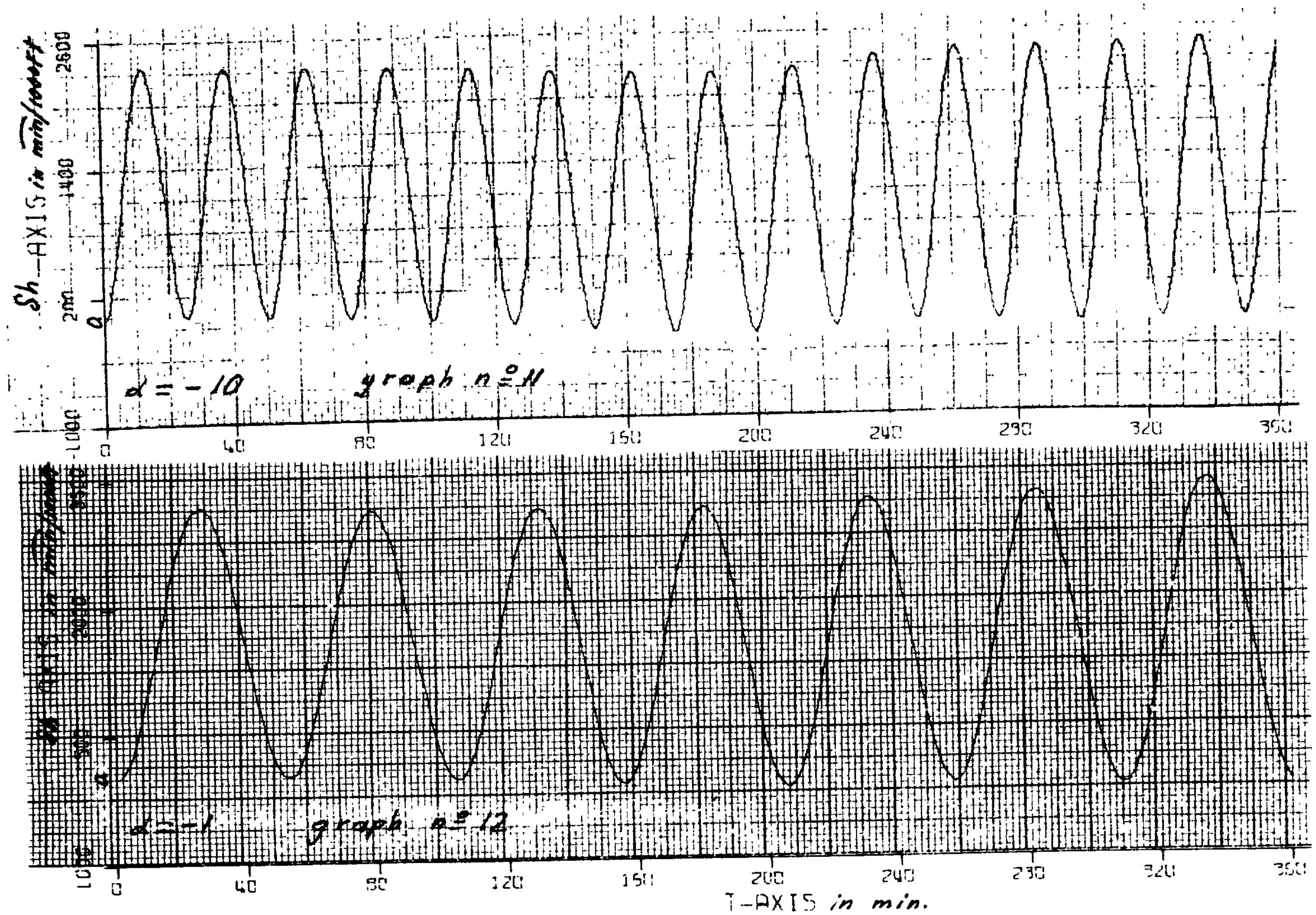
Navigation system errors due to altimeter uncertainty – $\alpha = -0.5$; $L = 45°$



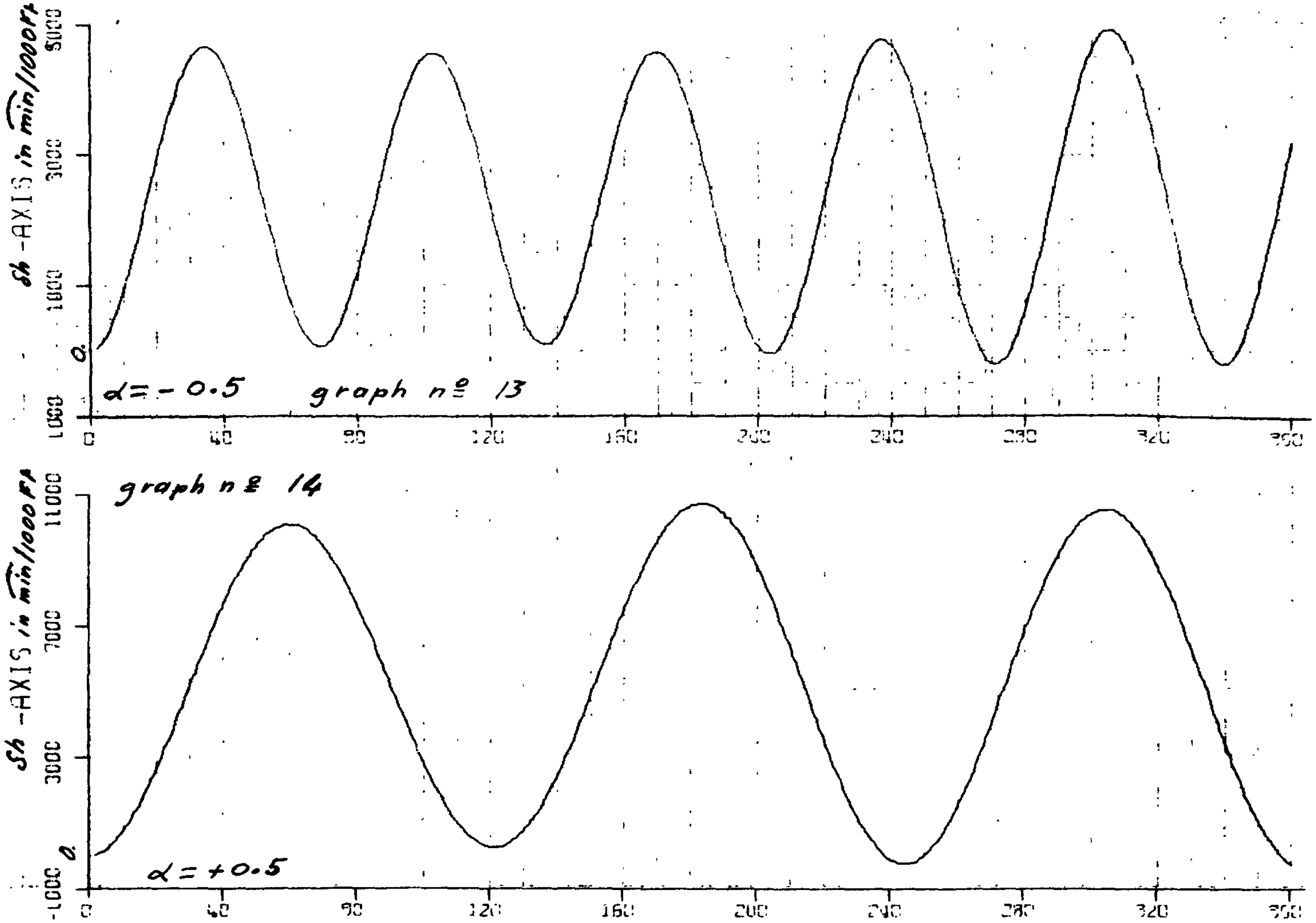
Navigation system errors due to altimeter uncertainty- $\alpha = -1$; $L = 45°$



Navigation system errors due to altimeter uncertainty - $\alpha = -10$; $L = 45°$

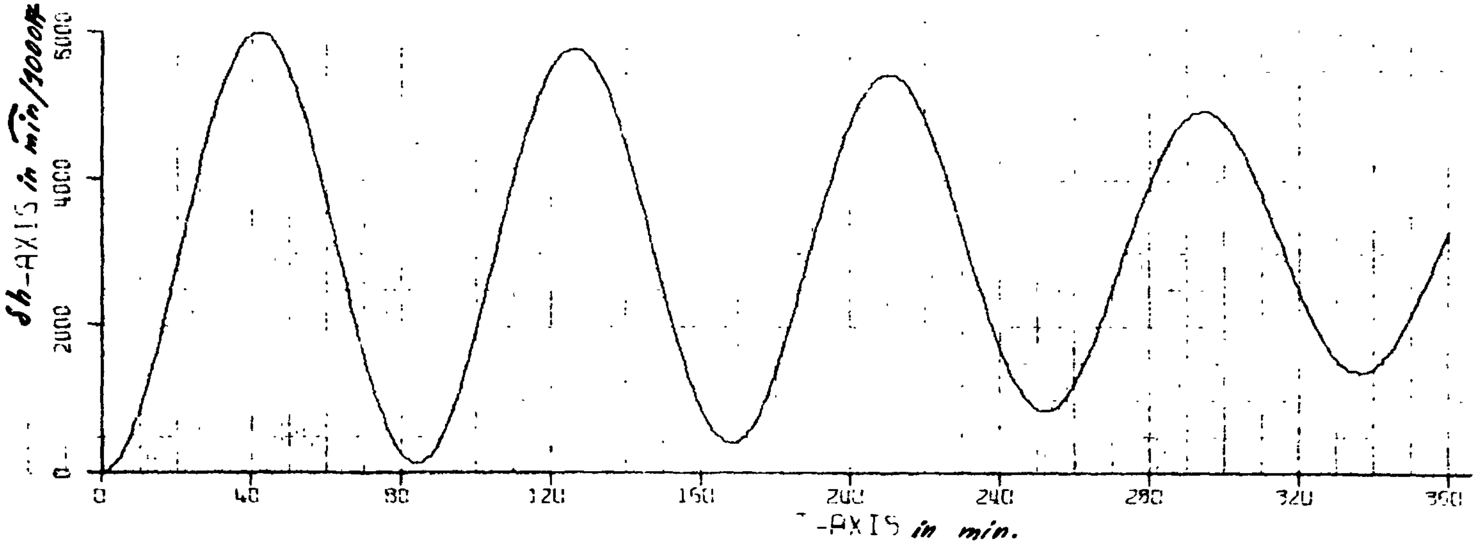


Navigation system errors due to altimeter uncertainty - L = 45°

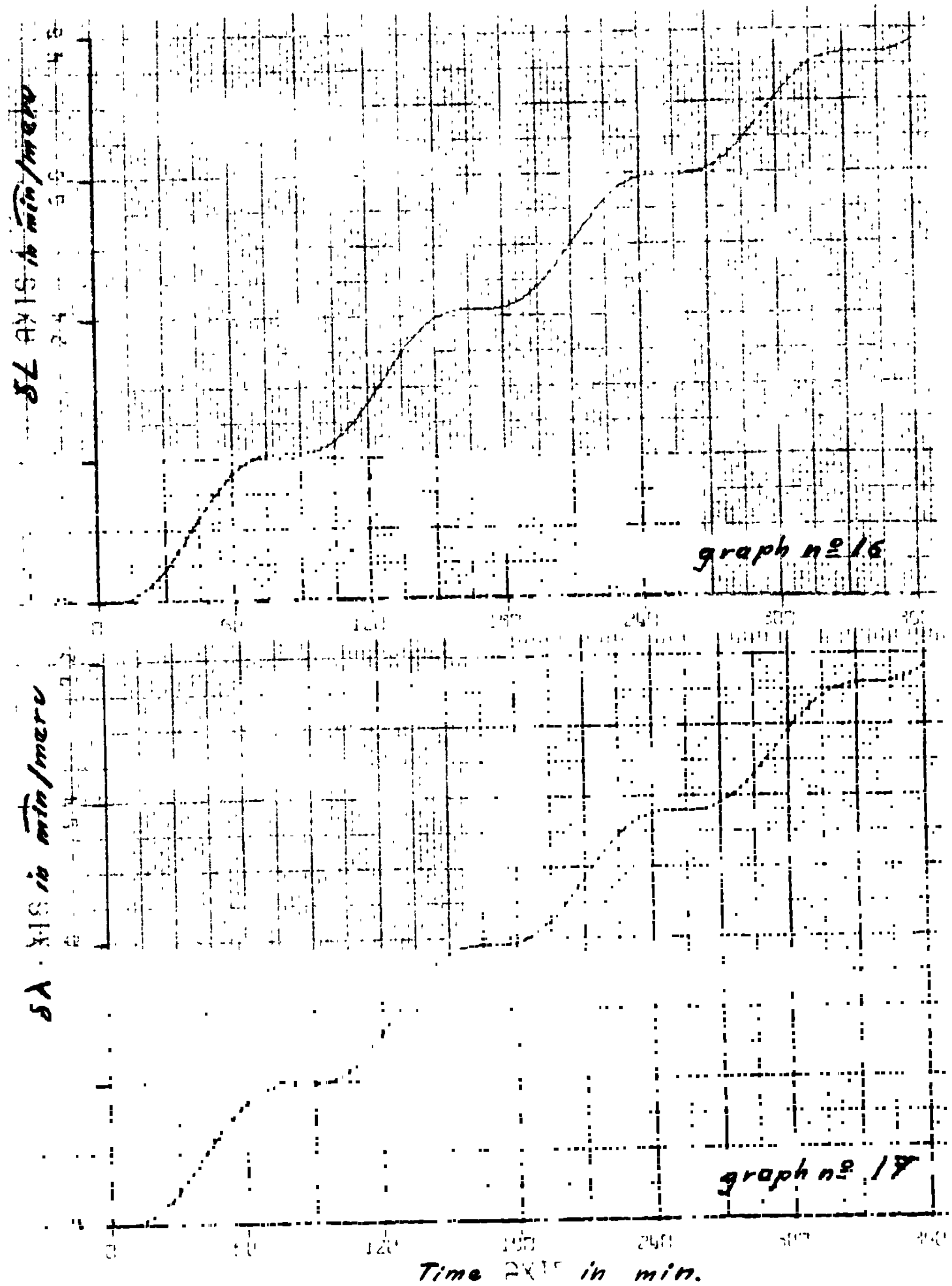


Navigation system errors due to altimeter uncertainty - L = 45°

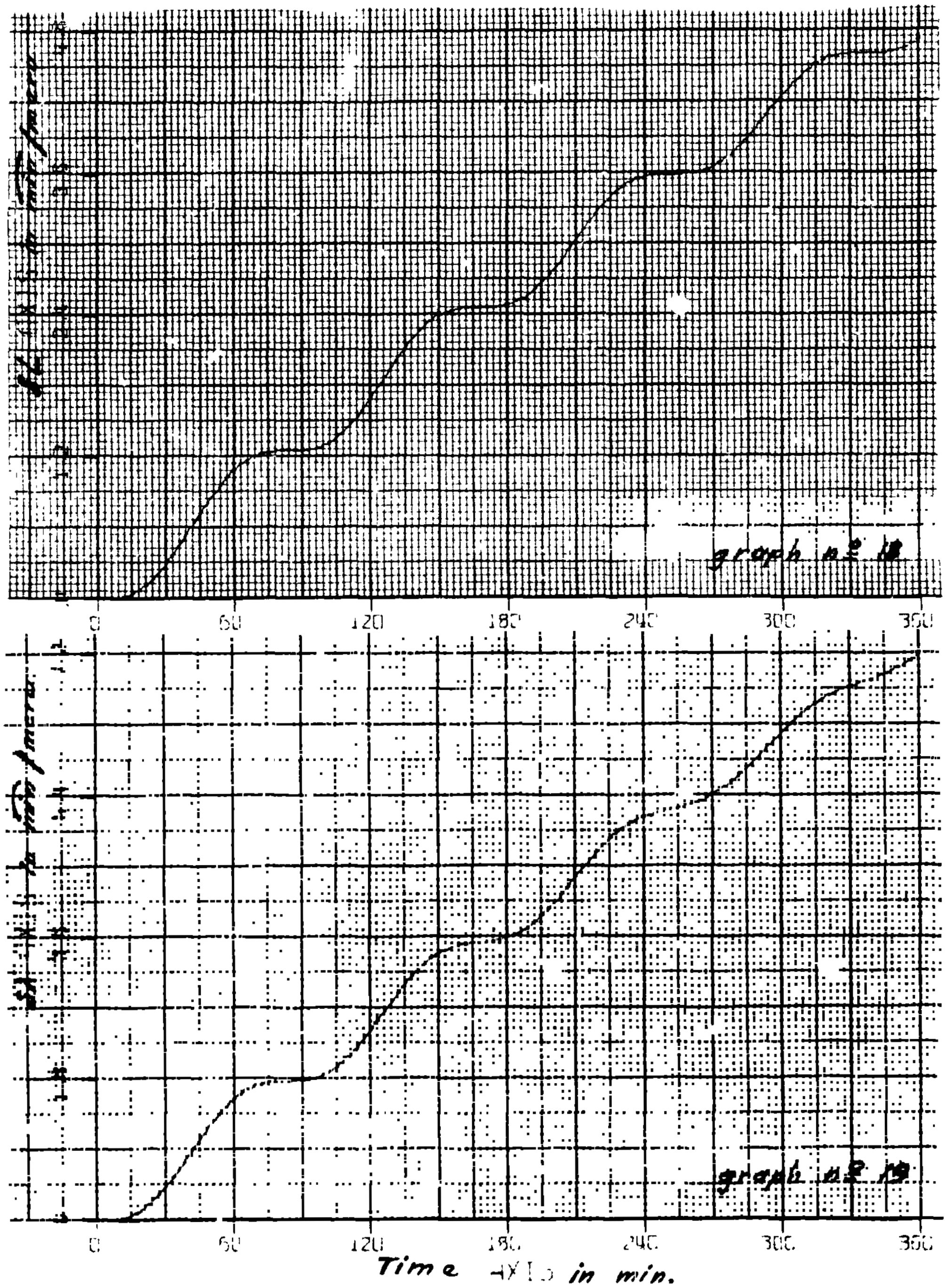
graph nº 15



Navigation system errors due to altimeter uncertainty - α=0 ; L=45°

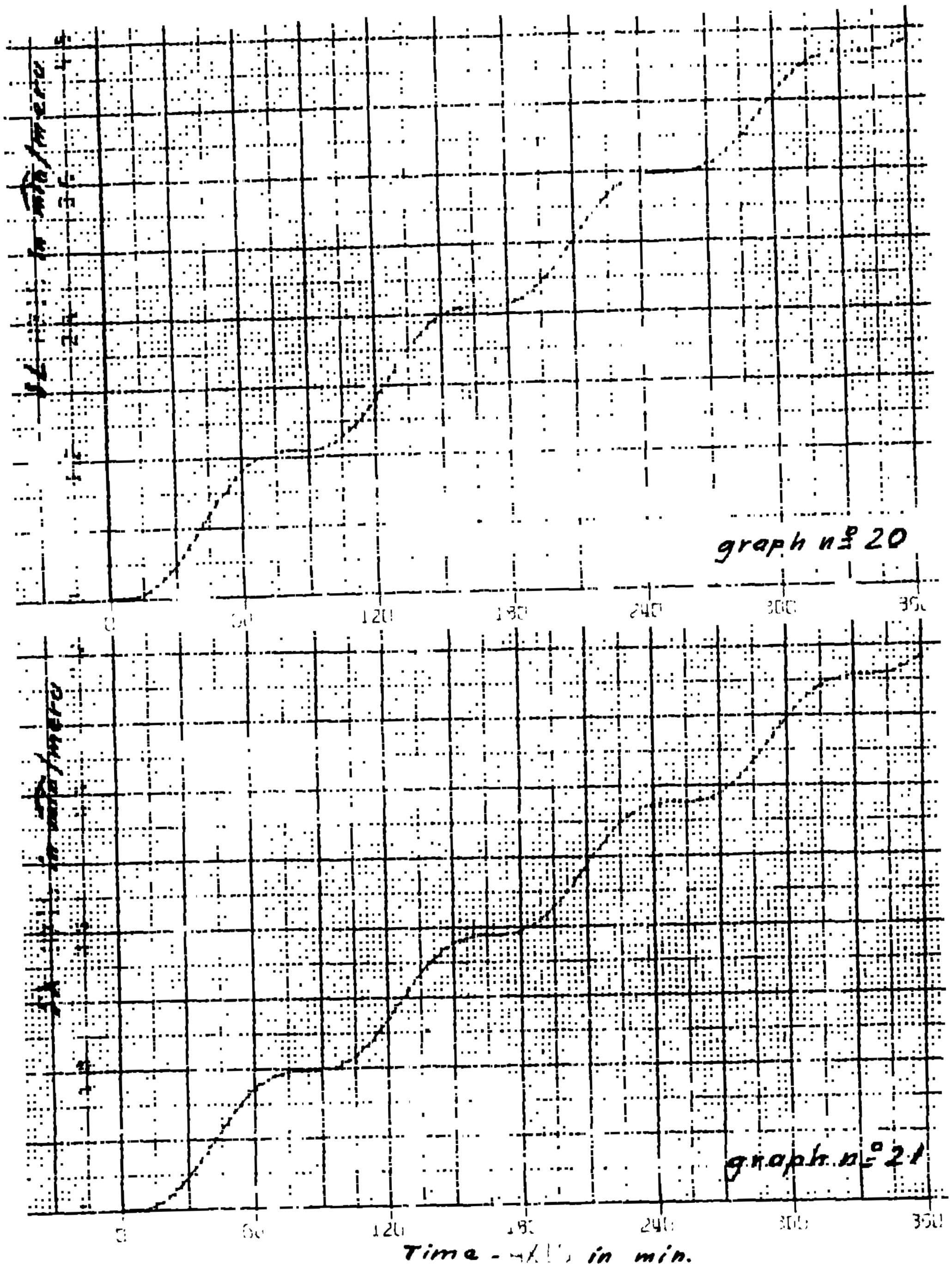


Navigation system errors due to gyro drift
$\alpha = -10$; $L = 45^\circ$; Root sum squared

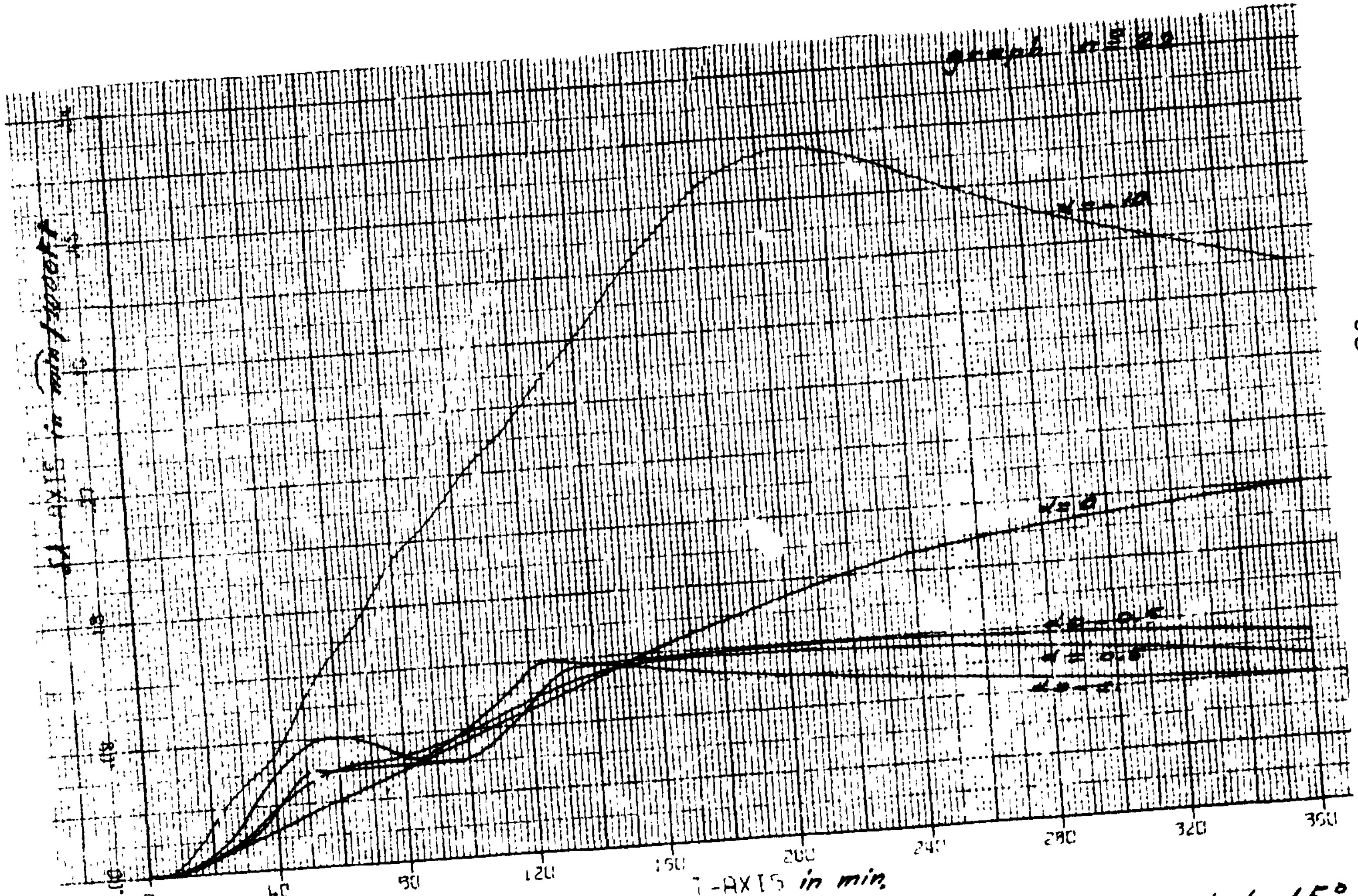


Navigation system errors due to gyro drift

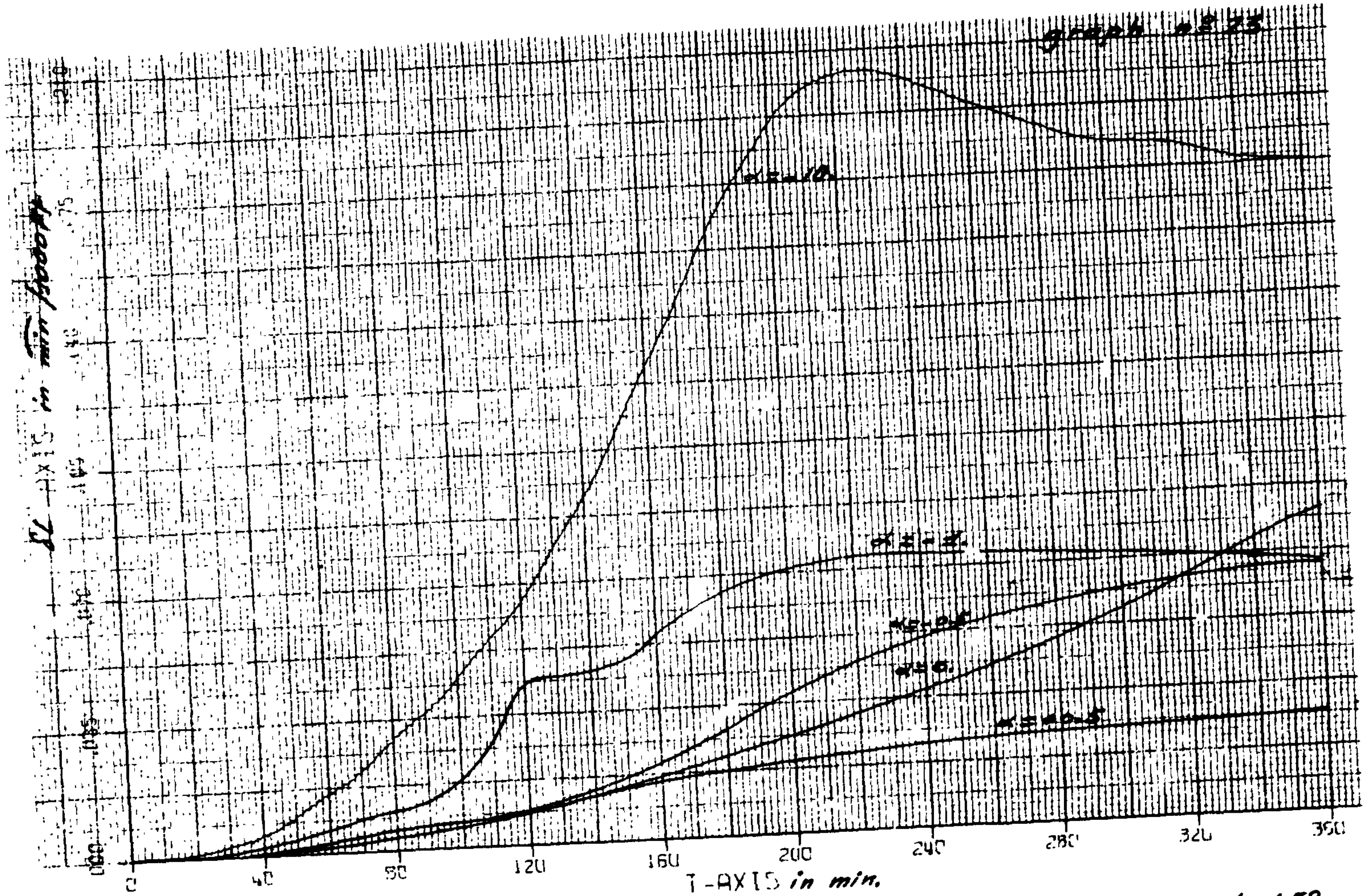
$\alpha = 0$; $L = 45°$; Root sum squared



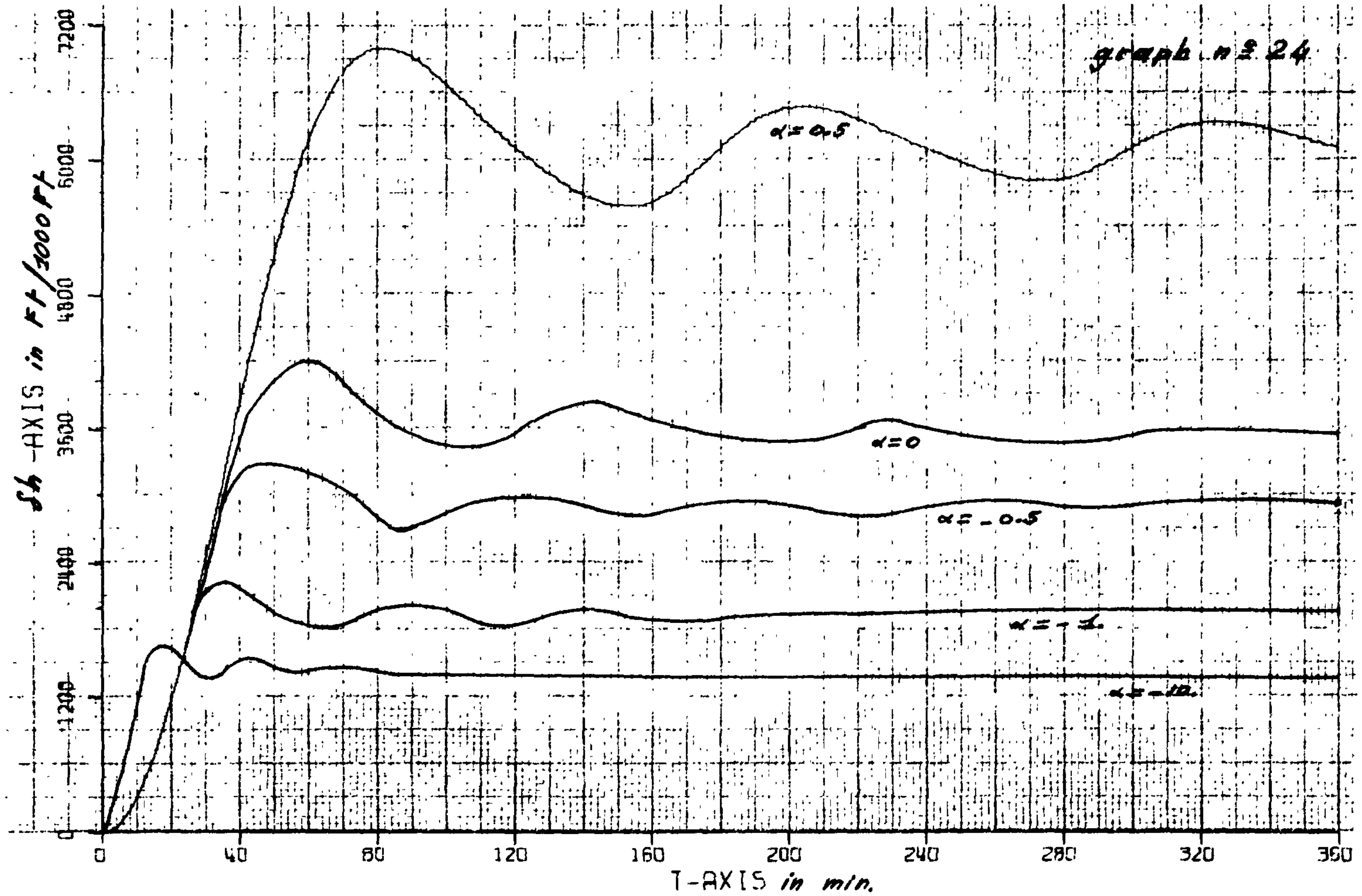
Navigation system errors due to gyro drift - α = 0.5
L = 45° ; Root sum squared



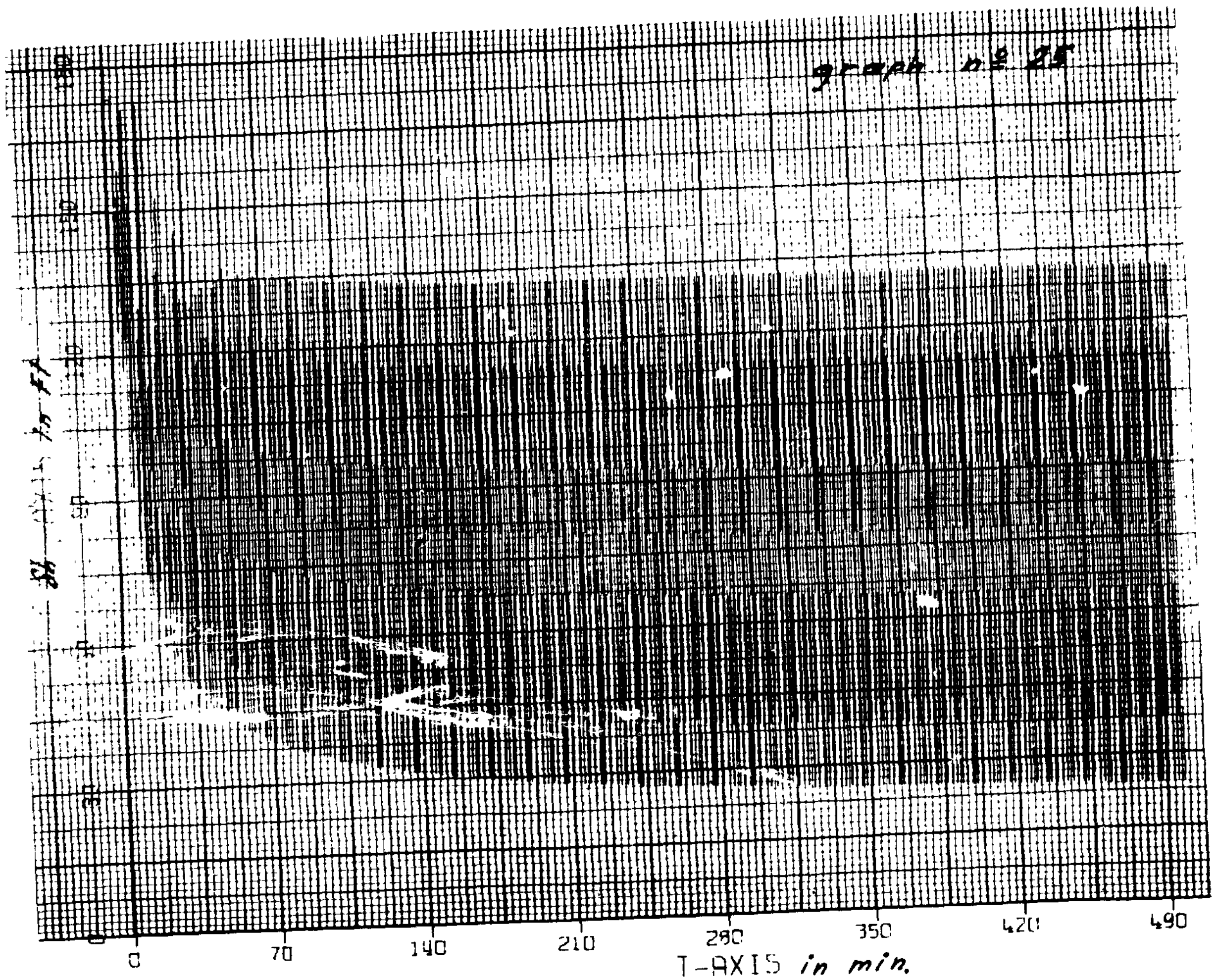
Navigation system errors due to altimeter uncertainty - Root mean squared - $L = 45°$



Navigation system errors due to altimeter uncertainty - Root mean squared - $L=45°$

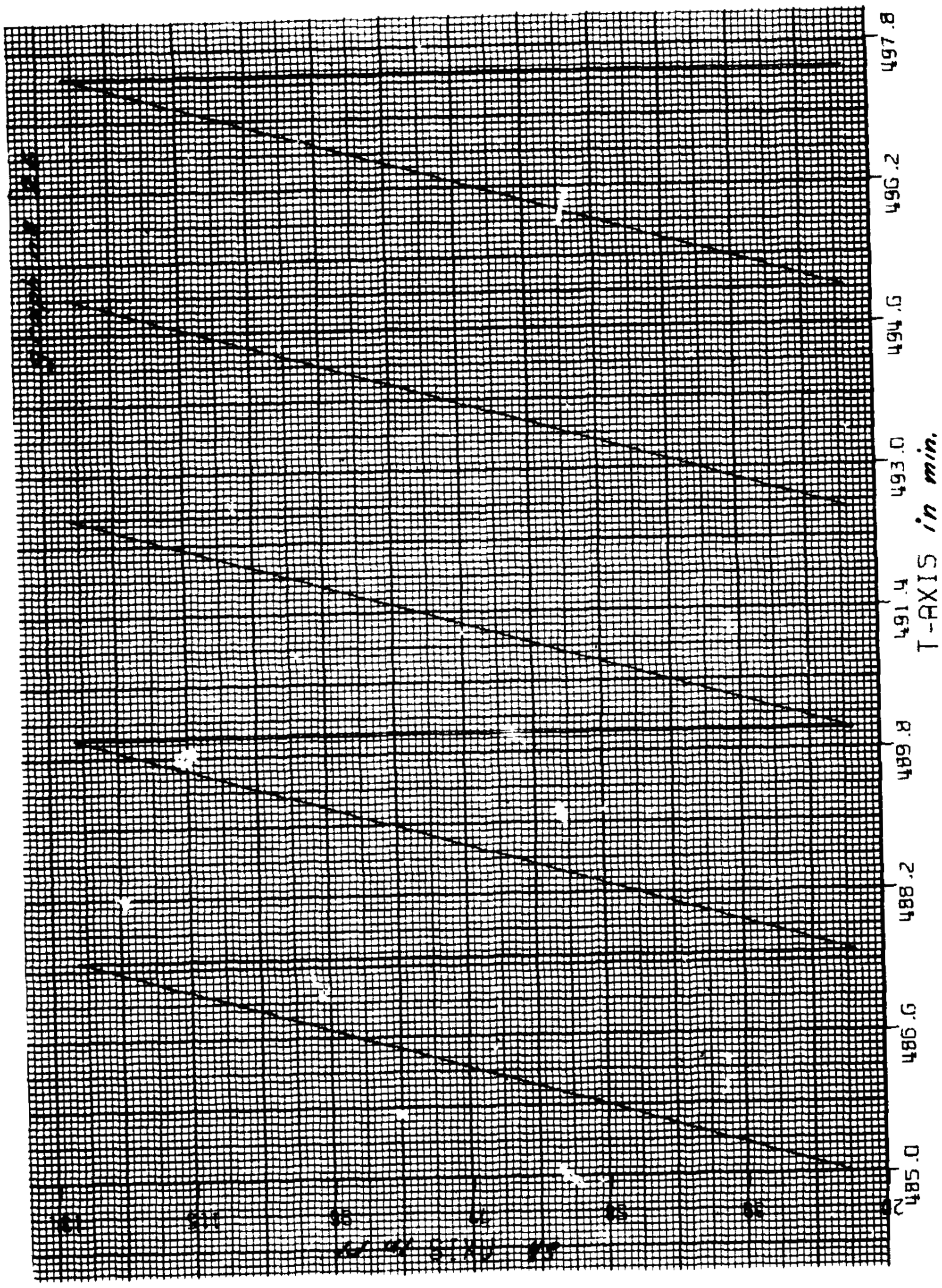


Navigation system errors due to altimeter uncertainty - Root mean squared - $L = 45°$

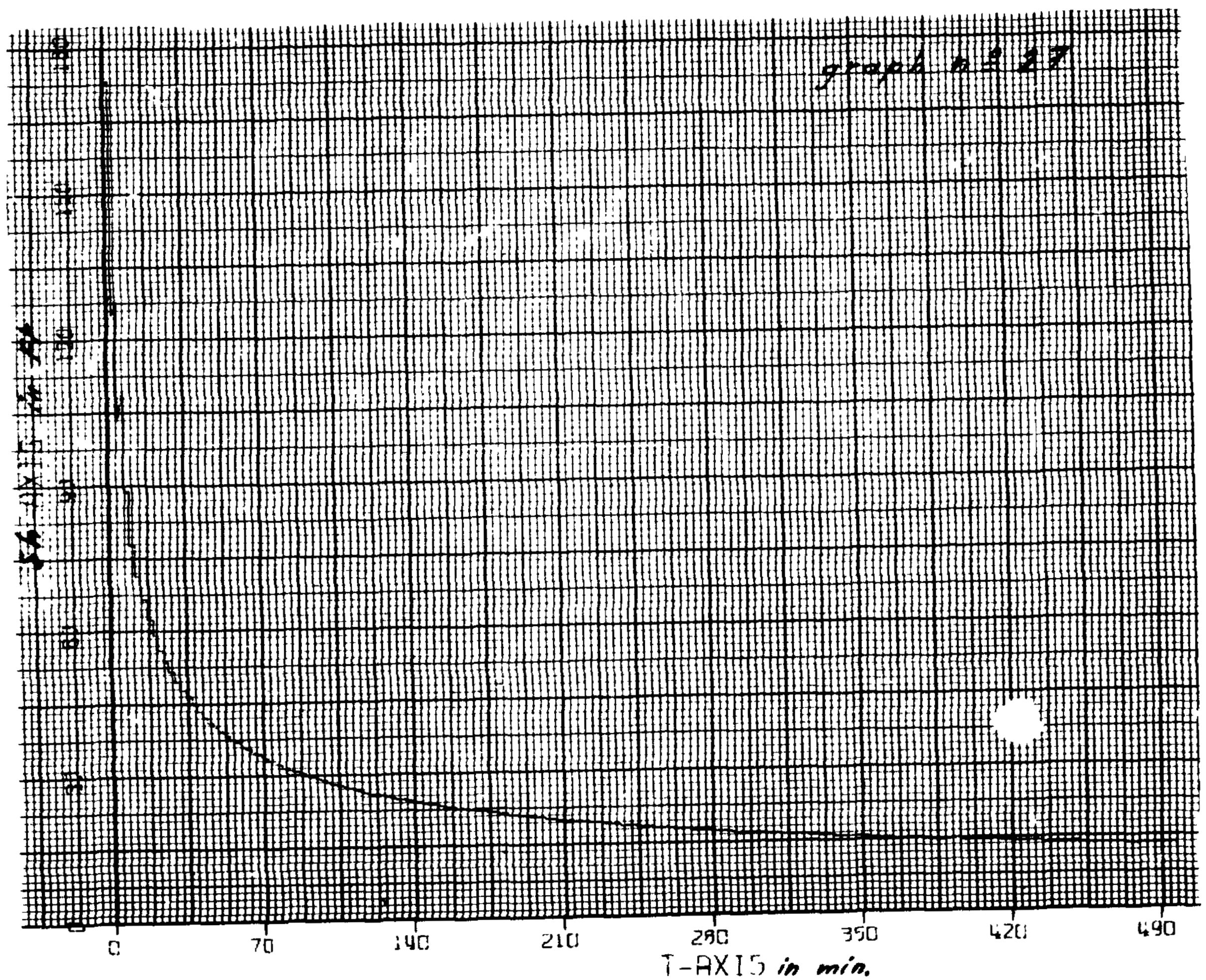


Optimal mixing of radio-altimeter and barometric altimeter informations

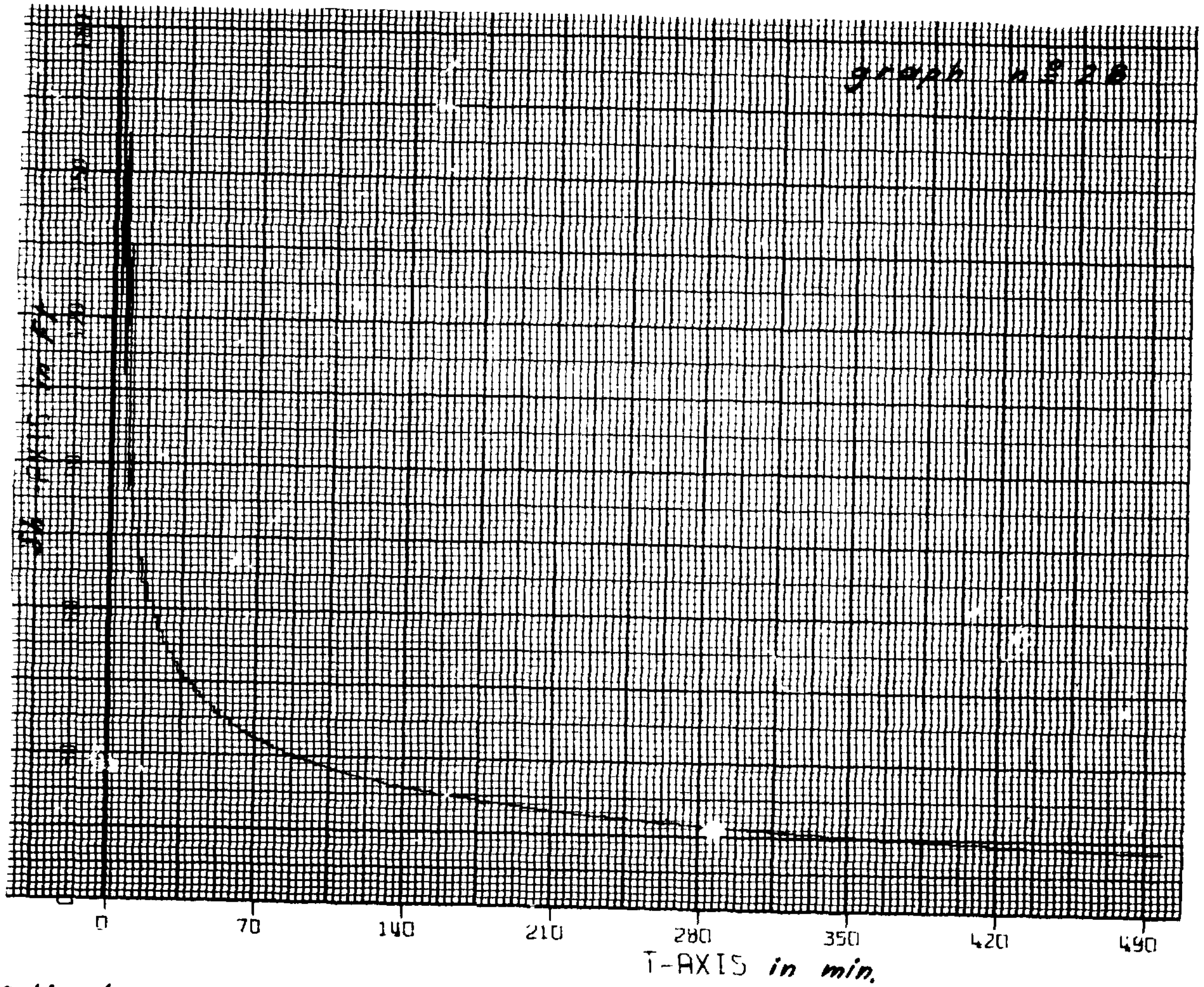
Root mean squared of error in altitude indication -



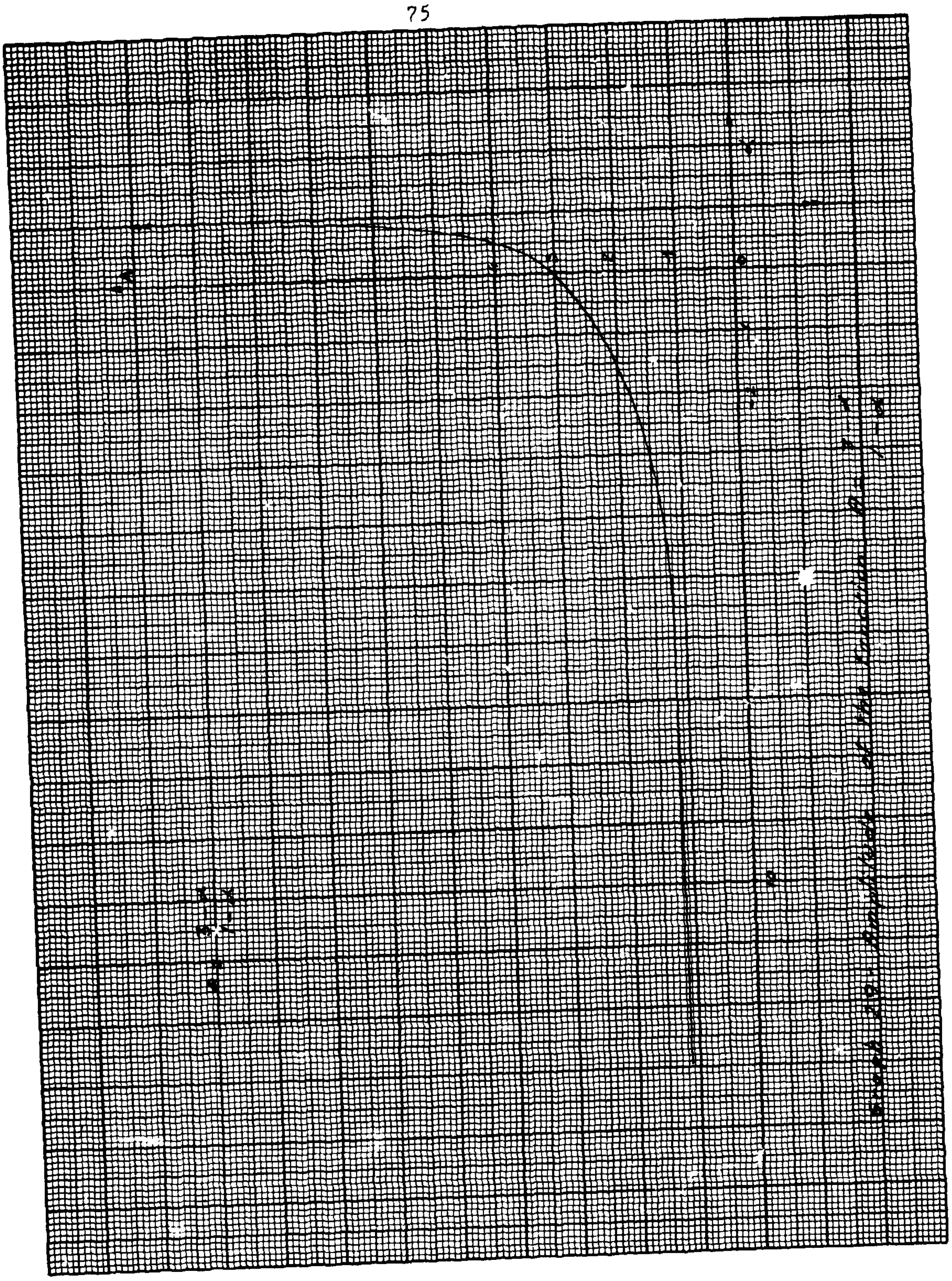
Enlargement of graph n° 25



Optimal mixing of radio-altimeter, barometric altimeter and I.N.S. informations
Root mean squared of error in altitude indication. Accelerometer uncertainty = g/10,000.



Optimal mixing of radio-altimeter, barometric altimeter and I.N.S. informations
Root mean squared of error in altitude indication - Accelerometer uncertainty=g/1000.



APPENDIX C

```
C                    PROGRAM NC. 1

C     CCMPUTATICN OF LONGITUDE,LATITUDE,AND ALTITUDE ERRORS FOR A
C     STRAPDOWN SYSTEM COMPUTING IN GEOCENTRIC INERTIAL FRAME
C

     OREAL R,CAP,YO,XO,MEMY,MEMX,X,Y,T,LLL,    DRI,CNI,QOT,QTT,CAL,
     1CAN,F,ATT,PDR,PDG,WIE,WS,WX,WY,WZ,DERY,PRMT,A,   SAVE,
     2NNN,AUX
      DIMENSION A(6,6),Y(6),YO(3,1),F(6),QOT(1,3),QTT(1,3),R(3,1),
     1SAVE(3,1),LLL(1),NNN(1),AUX(16,6),PRMT(5),T(6000),DERY(6),
     2D(3,3),B(3,1)
      COMMON ACP,ATT,ACF,ACO,WIE,CAP,WS,INIT,DRI(3,1),CNI(3,3),CAL(3,1),
     1CAN(3,1),MEMX(6000),MEMY(3,6000)
      CAP=10800./3.1416
      ACP=0.70711
      WIE=3.1416/(12.*3600.)
      WS=3.1416/2520.
      QOT(1,1)=1.
      QOT(1,2)=0.
      QOT(1,3)=0.
      QTT(1,1)=0.
      QTT(1,2)=1.
      QTT(1,3)=0.
      ATT=WS**2
      ACF=0.5
      ACO=1.
      EXTERNAL AFCT,FCT,OUTP
      PRMT(1)=0.
      PRMT(2)=21600.
      PRMT(3)=60.
      PRMT(4)=30.000/(3600.*180.)
      Y(1)=0.
      Y(2)=0.
      Y(3)=0.
      Y(4)=0.
      Y(5)=0.
      Y(6)=0.
      NDIM=6
      DERY(1)=0.15
      DERY(2)=0.15
      DERY(3)=0.15
      DERY(4)=0.2
      DERY(5)=0.2
      DERY(6)=0.15
      INIT=0
      CALL HCPL(PRMT,Y,DERY,NDIM,IHLF,AFCT,FCT,OUTP,AUX,A)
      WRITE (6,196)
  196 FORMAT(17HINTEGRATION DONE   )
      WRITE(6,900)INIT
  900 FORMAT(3X,I5)
```

```
      DO 27 KL=1,INIT
      DO 26 KN =1,3
   26 YO(KN,1)=MEMY(KN,KL)
      XO=MEMX(KL)
      CALL CNII(XO)
      CALL GMPRD(CNI,YO,R,3,3,1)
      DO 32 JJ=1,3
   32 SAVE(JJ,1)=R(JJ,1)
      T(KL)=XO/60.
      LLL(KL)=SAVE(1,1)*CAP/6400000.
      NNN(KL)=SAVE(2,1)*CAP/(0.70711*6400000.)
   27 PUNCH 699,T(KL),LLL(KL),NNN(KL)
  699 FORMAT(3X,F8.2,E15.5,E15.5,E15.5)
      CALL EXIT
      END
C
C
C     *******************************************
C
C
      SUBROUTINE GMPRD(D,B,R,N,M,L)
      DIMENSION C(1),B(1),R(1)
      COMMON ACP,ATT,ACF,ACO,WIE,CAP,WS,INIT,DRI(3,1),CNI(3,3),CAL(3,1),
     1CAN(3,1),MEMX(6000),MEMY(3,6000)
      IR=0
      IK=-M
      DO 10 K=1,L
      IK=IK+M
      DO 10 J=1,N
      IR=IR+1
      JI=J-N
      IB=IK
      R(IR)=0
      DO 10 I=1,M
      JI=JI+N
      IB=IB+1
   10 R(IR)=R(IR)+D(JI)*B(IB)
      RETURN
      END
C
C     *******************************************
C
      SUBROUTINE CNII(XO)
      COMMON ACP,ATT,ACF,ACO,WIE,CAP,WS,INIT,DRI(3,1),CNI(3,3),CAL(3,1),
     1CAN(3,1),MEMX(6000),MEMY(3,6000)
      PDR=COS(WIE*XO)
      PDG=SIN(WIE*XO)
      CNI(1,1)=-ACP*PDR
      CNI(1,2)=-ACP*PDG
      CNI(1,3)=ACP
      CNI(2,1)=-PDG
      CNI(2,2)=PDR
      CNI(2,3)=0.
      CNI(3,1)=-ACP*PDR
      CNI(3,2)=-ACP*PDG
```

```
      CNI(3,3)=-ACP
      RETURN
      END
C
C     ******************************************
C
      SUBROUTINE FCT(X,F)
      DIMENSION F(6)
      COMMON ACP,ATT,ACF,ACO,WIE,CAP,WS,INIT,DRI(3,1),CNI(3,3),CAL(3,1),
     1CAN(3,1),MEMX(6000),MEMY(3,6000)
      PDR=SIN(WIE*X)
      PDG=CCS(WIE*X)
      WX=0.
      WY=0.
      WZ=3.1416/(12.*3600.*1000.)
      W1=-ACP*WX*PDR/WIE+WY*(PDG-1.)/WIE-ACP*WZ*PDR/WIE
      W2=ACP*WX*(PDG-1.)/WIE+WY*PDR/WIE+WZ*(PDG-1.)*ACP/WIE
      W3=ACP*WX*X-ACP*WZ*X
      F(1)=0.
      F(2)=0.
      F(3)=0.
      F(4)=(6400000.*ACP*ATT)*(PDR*W3-W2)
      F(5)=(6400000.*ACP*ATT)*(W1-W3* PDG)
      F(6)=(6400000.*ACP*ATT)*(W2*PDG-W1*PDR)
      RETURN
      END
C
C     ******************************************
C
      SUBROUTINE AFCT(X,A)
      DIMENSION A(6,6)
      COMMON ACP,ATT,ACF,ACO,WIE,CAP,WS,INIT,DRI(3,1),CNI(3,3),CAL(3,1),
     1CAN(3,1),MEMX(6000),MEMY(3,6000)
      PDR=SIN(WIE*X)
      PDG=CCS(WIE*X)
      DO 410 I=1,3
      DO 410 J=1,3
  410 A(I,J)=0.
      A(1,5)=0.
      A(1,6)=0.
      A(2,4)=0.
      A(2,6)=0.
      A(3,4)=0.
      A(3,5)=0.
      DO   420 K=4,6
      DO 420 L=4,6
  420 A(K,L)=0.
      A(1,4)=1.
      A(2,5)=1.
      A(3,6)=1.
      A(4,1)=-ATT*(ACF*(PDG**2)*10.+1.)
      A(4,2)=-ATT*(5.*ACF*SIN(2.*WIE*X))
      A(4,3)=-ATT*(5.*ACO*PDG)
      A(5,1)=-ATT*(5.*SIN(2.*WIE*X))*0.5
```

```
      A(5,2)=-ATT*(1.+10.*ACF*PDR**2)
      A(5,3)=-ATT*(5.*ACO*PDR)
      A(6,1)=-ATT*( 5.*ACO*PDG)
      A(6,2)=-ATT*(5.*ACO*PDR)
      A(6,3)=-ATT*(1.+10.*ACP**2)
      RETURN
      END
C
C     ***************************************************************
C     ********************************************
C
      SUBROUTINE HCPL(PRMT,Y,DERY,NDIM,IHLF,AFCT,FCT,OUTP,AUX,A)
C
C     THE FOLLOWING PART OF SUBROUTINE HCPL (UNTIL FIRST BREAK-
C     POINT FOR LINKAGE) HAS TO STAY IN CORE DURING THE WHOLE
C     COMPUTATION
C
      DIMENSION PRMT(1),Y(1),DERY(1),AUX(16,1),A(1)
      COMMON ACP,ATT,ACF,ACO,WIE,CAP,WS,INIT,DRI(3,1),CNI(3,3),CAL(3,1),
     1CAN(3,1),MEMX(6000),MEMY(3,6000)
      GOTO 100
C
C     THIS PART OF SUBROUTINE HCPL COMPUTES THE RIGHT HAND SIDE DERY OF
C     THE GIVEN SYSTEM OF LINEAR DIFFERENTIAL EQUATIONS
    1 CALL AFCT(X,A)
      CALL FCT(X,DERY)
      DO 3 M=1,NDIM
      LL=M-NDIM
      HS=0.
      DO 2 L=1,NDIM
      LL=LL+NDIM
    2 HS=HS+A(LL)*Y(L)
    3 DERY(M)=HS+DERY(M)
      GOTO(105,202,204,206,115,122,125,308,312,327,329,128),ISW2
C     POSSIBLE BREAK POINT FOR LINKAGE
C
  100 N=1
      IHLF=0
      X=PRMT(1)
      H=PRMT(3)
      PRMT(5)=0.
      DO 101 I=1,NDIM
      AUX(16,I)=0.
      AUX(15,I)=DERY(I)
  101 AUX(1,I)=Y(I)
      IF(H*(PRMT(2)-X))103,102,104
C
C     ERROR RETURN
  102 IHLF=12
      GOTO 104
  103 IHLF=13
C
C     COMPUTATION OF DERY FOR STARTING VALUES
```

```
  104 ISW2=1
      GOTO 1
C
C     RECORDING OF STARTING VALUES
  105 CALL CUTP(X,Y,CERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))1C7,106,107
  106 IF(IHLF)1C8,108,107
  107 RETURN
  108 DC 109 I=1,NDIM

  109 AUX(8,I)=CERY(I)
C
C     CCMPUTATION OF AUX(2,I)
      ISW1=1
      GOTO 200
C
  110 X=X+H
      DO 111 I=1,NDIM
  111 AUX(2,I)=Y(I)
C
C     INCREMENT H IS TESTEC BY MEANS OF BISSECTION
  112 IHLF=IHLF+1
      X=X-H
      DO 113 I=1,NDIM
  113 AUX(4,I)=AUX(2,I)
      H=.5*H
      N=1
      ISW1=2
      GOTO 200
C
  114 X=X+H
      ISW2=5
      GOTO 1
  115 N=2
      DO 116 I=1,NDIM
      AUX(2,I)=Y(I)
  116 AUX(9,I)=DERY(I)
      ISW1=3
      GOTO 200
C
C     CCMPUTATION OF TEST VALUE DELT
  117 DELT=0.
      DO 118 I=1,NDIM
  118 DELT=DELT+AUX(15,I)*ABS(Y(I)-AUX(4,I))
      DELT=.06666667*DELT
      IF(DELT-PRMT(4))121,121,119
  119 IF(IHLF-1C)112,120,120
C
C     NO SATISFACTORY AFTER 10 BISSECTIONS.ERROR MESSAGE.
  120 IHLF=11
      X=X+H
      GOTO 104
C
C     SATISFACTORY ACCURACY AFTER LESS THAN  11 BISSECTIONS
```

```
  121 X=X+H
      ISW2=6
      GOTO 1
  122 DO 123 I=1,NDIM
      AUX(3,I)=Y(I)
  123 AUX(10,I)=DERY(I)
      N=3
      ISW1=4
      GOTO 200
C
  124 N=1
      X=X+H
      ISW2=7
      GOTO 1
  125 X=PRMT(1)
      DO 126 I=1,NDIM
      AUX(11,I)=DERY(I)
  126 Y(I)=AUX(1,I)+H*(.375*AUX(8,I)+.7916667*AUX(9,I)
     1-.2083333*AUX(10,I)+.04166667*DERY(I))
  127 X=X+H
      N=N+1
      ISW2=12
      GOTO 1
  128 CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))107,129,107
  129 IF(N-4)130,300,300
  130 DO 131 I=1,NDIM
      AUX(N,I)=Y(I)
  131 AUX(N+7,I)=DERY(I)
      IF(N-3)132,134,300
C
  132 DO 133 I=1,NDIM
      DELT=AUX(9,I)+AUX(9,I)
      DELT=DELT+DELT
  133 Y(I)=AUX(1,I)+.3333333*H*(AUX(8,I)+DELT+AUX(10,I))
      GOTO 127
C
  134 DO 135 I=1,NDIM
      DELT=AUX(9,I)+AUX(10,I)
      DELT=DELT+DELT+DELT
  135 Y(I)=AUX(1,I)+.375*H*(AUX(8,I)+DELT+AUX(11,I))
      GOTO 127
C
C     THE FOLLOWING PART OF SUBROUTINE HCPL COMPUTES BY MEANS OF
C     RUNGE-KUTTA METHOD STARTING VALUES FOR THE NOT SELF-STARTING
C     PREDICTOR-CORRECTOR METHOD
  200 Z=X
      DO 201 I=1,NDIM
      X=H*AUX(N+7,I)
      AUX(5,I)=X
  201 Y(I)=AUX(N,I)+.4*X
C     X IS AN AUXILLIARY STORAGE LOCATION
C
      X=Z+.4*X
```

```
      ISW2=2
      GOTO 1
  202 DO 203 I=1,NDIM
      X=H*DERY(I)
      AUX(6,I)=X
  203 Y(I)=AUX(N,I)+.2969776*AUX(5,I)+.1587596*X
C
      X=Z+.4557372*H
      ISW2=3
      GOTO 1
  204 DO 205 I=1,NDIM
      X=H*DERY(I)
      AUX(7,I)=X
  205 Y(I)=AUX(N,I)+.2181004*AUX(5,I)-3.050965*AUX(6,I)+3.832865*X
C
      X=Z+H
      ISW2=4
      GOTO 1
  206 DO 207 I=1,NDIM
  207OY(I)=AUX(N,I)+.1747603*AUX(5,I)-.5514807*AUX(6,I)
     1+1.205536*AUX(7,I)+.1711848*H*DERY(I)
      X=Z
      GOTO(110,114,117,124),ISW1
C
C     POSSIBLE BREAK-POINT FOR LINKAGE
C
C     STARTING VALUES ARE COMPUTED
C     NOW START HAMMINGS MODIFIED PREDICTOR-CORRECTOR METHOD
  300 ISTEP=3
  301 IF(N-8)304,302,304
C
C     N=8 CAUSES THE ROWS OF AUX TOCHANGE THE.R STORAGE LOCATIONS
  302 DO 303 N=2,7
      DO 303 I=1,NDIM
      AUX(N-1,I)=AUX(N,I)
  303 AUX(N+6,I)=AUX(N+7,I)
      N=7
C
C     N LESS THAN 8 CAUSESN+1 TO GET N
  304 N=N+1
C
C     COMPUTATICN OF NEXT VECTOR Y
      DO 305 I=1,NDIM
      AUX(N-1,I)=Y(I)
  305 AUX(N+6,I)=DERY(I)
      X=X+H
  306 ISTEP=ISTEP+1
      DO 307 I=1,NDIM
     ODELT=AUX(N-4,I)+1.333333*H*(AUX(N+6,I)+AUX(N+6,I)-AUX(N+5,I)+
     1AUX(N+4,I)+AUX(N+4,I))
      Y(I)=DELT-.9256198*AUX(16,I)
  307 AUX(16,I)=DELT
C     PREDICTOR IS NOW GENERATED IN ROW 16 OF AUX, MODIFIED PREDICTOR
C     IS GENERATED IN Y. DELT MEANS AN AUXILLIARY STORAGE.
```

```
      ISW2=8
      GOTO 1
C     DERIVATIVE OF MODIFIED PREDICTOR IS GENERATED IN DERY
C
  308 DO 309 I=1,NDIM
     0DELT=.125*(9.*AUX(N-1,I)-AUX(N-3,I)+3.*H*(DERY(I)+AUX(N+6,I)+
     1AUX(N+6,I)-AUX(N+5,I)))
      AUX(16,I)=AUX(16,I)-DELT
  309 Y(I)=DELT+.07438017*AUX(16,I)
C
C     TEST WHETHER H MUST BE HALVED OR DOUBLED
      DELT=0
      DO 310 I=1,NDIM
  310 DELT=DELT+AUX(15,I)*ABS(AUX(16,I))
        IF(DELT-PRMT(4))311,324,324
C
C     H MUST NOT BE HALVED. THAT MEANS Y(I) ARE GOOD.
  311 ISW2=9
      GOTO 1
  312 CALL OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      IF(PRMT(5))314,313,314
  313 IF(IHLF-11)315,314,314
  314 RETURN
  315 IF(H*(X-PRMT(2)))316,314,314
  316 IF(ABS(X-PRMT(2))-.1*ABS(H))314,317,317
  317 IF(DELT-.02*PRMT(4))318,318,301
C
C
C     H COULD BE DOUBLED IF ALL PREECEEDING VALUES ARE
C     AVAILABLE
  318 IF(IHLF)301,301,319
  319 IF(N-7)301,320,320
  320 IF(ISTEP-4)301,321,321
  321 IMOD= ISTEP/2
      IF(ISTEP -IMOD-IMOD)301,322,301
  322 H=H+H
      IHLF=IHLF-1
      ISTEP=0
      DO 323 I=1,NDIM
      AUX(N-1,I)=AUX(N-2,I)
      AUX(N-2,I)=AUX(N-4,I)
      AUX(N-3,I)=AUX(N-6,I)
      AUX(N+6,I)=AUX(N+5,I)
      AUX(N+5,I)=AUX(N+3,I)
      AUX(N+4,I)=AUX(N+1,I)
      DELT=AUX(N+6,I)+AUX(N+5,I)
      DELT=DELT+DELT+DELT
  3230AUX(16,I)=8.962963*(Y(I)-AUX(N-3,I))-3.361111*H*(DERY(I)+DELT
     1+AUX(N+4,I))
      GOTO 301
C
C
C     H MUST BE HALVED
  324 IHLF=IHLF+1
```

```
      IF(IHLF-10)325,325,311
  325 H=.5*H
      ISTEP=0
      DO 326 I=1,NDIM
     0Y(I)=.00390625*(80.*AUX(N-1,I)+135.*AUX(N-2,I)+40.*AUX(N-3,I)+
     1AUX(N-4,I))-.1171875*(AUX(N+6,I)-6.*AUX(N+5,I)-AUX(N+4,I))*H
     0AUX(N-4,I)= .00390625*(12.*AUX(N-1,I)+135.*AUX(N-2,I)+
     1108.*AUX(N-3,I)+AUX(N-4,I))-.0234375*(AUX(N+6,I)+18.*AUX(N+5,I)-
     29.*AUX(N+4,I))*H
      AUX(N-3,I)=AUX(N-2,I)
  326 AUX(N+4,I)=AUX(N+5,I)
      DELT=X-H
      X=DELT-(H+H)
      ISW2=10
      GOTO 1
  327 DO 328 I=1,NDIM
      AUX(N-2,I)=Y(I)
      AUX(N+5,I)=DERY(I)
  328 Y(I)=AUX(N-4,I)
      X=X-(H+H)
      ISW2=11
      GOTO 1
  329 X=DELT
      DO 330 I=1,NDIM
      DELT=AUX(N+5,I)+AUX(N+4,I)
      DELT=DELT+DELT+DELT
     0AUX(16,I)=8.962963*(AUX(N-1,I)-Y(I))-3.361111*H*(AUX(N+6,I)+DELT
     1+DERY(I))
  330 AUX(N+3,I)=DERY(I)
      GOTO 306
      END
C
C     *********************************************
C
      SUBROUTINE OUTP(X,Y,DERY,IHLF,NDIM,PRMT)
      REAL MEMX,MEMY
      DIMENSICN Y(6),DERY(6),PRMT(5)
      COMMON ACP,ATT,ACF,ACO,WIE,CAP,WS,INIT,DRI(3,1),CNI(3,3),CAL(3,1),
     1CAN(3,1),MEMX(6000),MEMY(3,6000)
      INIT=INIT+1
      MEMX(INIT)=X
      DO 400 K=1,3
  400 MEMY(K,INIT)=Y(K)
      IF(INIT-2)678,677,677
  677 PRMT(4)=(ABS(Y(1))+ABS(Y(2))+ABS(Y(3)))/300.
  678 CONTINUE
      RETURN
      END
```

```
C             PROGRAM  NO. 2

C      SEARCH FOR THE MINIMUM OF THE SQUARED FUNCTION

       DIMENSION H(9),X(2),G(2),ARG(2),GRAD(2)
       EXTERNAL FUNCT
       EPS=1./100000.
       LIMIT=400
       X(1)=150.
       X(2)=150.
       EST=10.
       N=2
       CALL FMFP(FUNCT,2,X,F,G,EST,EPS,LIMIT,IER,H)
       CALL EXIT
       END
C      ********************************************
       SUBROUTINE FUNCT(N,X,VAL,GRAD)
       DIMENSION X(2),GRAD(2)
       UU2=64./10000.
       HH2=50.
       VV2=50.
       T=1.
       P=0.1
       A=1.+P*T
       WS=3.1416/2520.
       WS2=WS**2
       E2=X(1)*X(1)
        E3=E2*X(1)
       F2=X(2)*X(2)
       F3=F2*X(2)
       UN=F3*X(1)*P*P*T+F2*E2*P*T*A+E3*P*T*A+F3*X(2)*P*T*T
       DE=-2.*X(1)*X(2)*A*P*T+X(2)*A*P*P+E2*X(2)*A*T*T
       UN1=F3*P*P*T+2.*X(1)*F2 *P*T*A+3.*E2*P*T*A+3.*E2*X(2)*P*T*T
       UN2=2.*X(2)*E2*P*T*A+E3*P*T*T
       DE1=-2.*X(2)*A*P*T+2.*X(1)*X(2)*A*T*T
       DE2=-2.*X(1)*A*P*T+A*P*P+E2*A*T*T
       VVV=2.*T*HH2*UN/DE+UU2/E2+VV2*(1.+2.*WS2/X(1))**2
       GGG1=2.*T*HH2*(UN1*DE-DE1* UN)/(DE**2)-2.*UU2/E3-4.*VV2*(1.+
      12.*WS2/X(1))*WS2/E2
       GGG2=2.*T*HH2*(UN2*DE-DE2*UN)/(DE**2)
       VAL=VVV**2
       GRAD(1)=2.*VVV*GGG1
       GRAD(2)=2.*VVV*GGG2
       F=VAL
       WRITE(6,699)X(1),X(2),F,VVV
   699 FORMAT(3X,E15.5,E15.5,E17.7,E17.7)
       RETURN
       END

C      ********************************************
```

96

```
      SUBROUTINE FMFP(FUNCT,N,X,F,G,EST,EPS,LIMIT,IER,H)
C
C     ..................................................................
C
C        SUBROUTINE FMFP
C
C        PURPOSE
C           TO FIND A LOCAL MINIMUM OF A FUNCTION OF SEVERAL VARIABLES
C           BY THE METHOD OF FLETCHER AND POWELL
C
C        USAGE
C           CALL FMFP(FUNCT,N,X,F,G,EST,EPS,LIMIT,IER,H)
C
C        DESCRIPTION OF PARAMETERS
C           FUNCT  - USER-WRITTEN SUBROUTINE CONCERNING THE FUNCTION TO
C                    BE MINIMIZED. IT MUST BE OF THE FORM
C                    SUBROUTINE FUNCT(N,ARG,VAL,GRAD)
C                    AND MUST SERVE THE FOLLOWING PURPOSE
C                    FOR EACH N-DIMENSIONAL ARGUMENT VECTOR  ARG,
C                    FUNCTION VALUE AND GRADIENT VECTOR MUST BE COMPUTED
C                    AND, ON RETURN, STORED IN VAL AND GRAD RESPECTIVELY
C           N      - NUMBER OF VARIABLES
C           X      - VECTOR OF DIMENSION N CONTAINING THE INITIAL
C                    ARGUMENT WHERE THE ITERATION STARTS. ON RETURN,
C                    X HOLDS THE ARGUMENT CORRESPONDING TO THE
C                    COMPUTED MINIMUM FUNCTION VALUE
C           F      - SINGLE VARIABLE CONTAINING THE MINIMUM FUNCTION
C                    VALUE ON RETURN, I.E. F=F(X).
C           G      - VECTOR OF DIMENSION N CONTAINING THE GRADIENT
C                    VECTOR CORRESPONDING TO THE MINIMUM ON RETURN,
C                    I.E. G=G(X).
C           EST    - IS AN ESTIMATE OF THE MINIMUM FUNCTION VALUE.
C           EPS    - TESTVALUE REPRESENTING THE EXPECTED ABSOLUTE ERROR.
C                    A REASONABLE CHOICE IS 10**(-6), I.E.
C                    SOMEWHAT GREATER THAN 10**(-D), WHERE D IS THE
C                    NUMBER OF SIGNIFICANT DIGITS IN FLOATING POINT
C                    REPRESENTATION.
C           LIMIT  - MAXIMUM NUMBER OF ITERATIONS.
C           IER    - ERROR PARAMETER
C                    IER = 0 MEANS CONVERGENCE WAS OBTAINED
C                    IER = 1 MEANS NO CONVERGENCE IN LIMIT ITERATIONS
C                    IER =-1 MEANS ERRORS IN GRADIENT CALCULATION
C                    IER = 2 MEANS LINEAR SEARCH TECHNIQUE INDICATES
C                    IT IS LIKELY THAT THERE EXISTS NO MINIMUM.
C           H      - WORKING STORAGE OF DIMENSION N*(N+7)/2.
C
C        REMARKS
C            I) THE SUBROUTINE NAME REPLACING THE DUMMY ARGUMENT  FUNCT
C               MUST BE DECLARED AS EXTERNAL IN THE CALLING PROGRAM.
C           II) IER IS SET TO 2 IF , STEPPING IN ONE OF THE COMPUTED
C               DIRECTIONS, THE FUNCTION WILL NEVER INCREASE WITHIN
C               A TOLERABLE RANGE OF ARGUMENT.
```

```
C                IER = 2 MAY OCCUR ALSO IF THE INTERVAL WHERE F
C                INCREASES IS SMALL AND THE INITIAL ARGUMENT WAS
C                RELATIVELY FAR AWAY FROM THE MINIMUM SUCH THAT THE
C                MINIMUM WAS OVERLEAPED. THIS IS DUE TO THE SEARCH
C                TECHNIQUE WHICH DOUBLES THE STEPSIZE UNTIL A POINT
C                IS FOUND WHERE THE FUNCTION INCREASES.
C
C        SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED
C           FUNCT
C
C        METHOD
C           THE METHOD IS DESCRIBED IN THE FOLLOWING ARTICLE
C           R. FLETCHER AND M.J.D. POWELL, A RAPID DESCENT METHOD FOR
C           MINIMIZATION,
C           COMPUTER JOURNAL VOL.6, ISS. 2, 1963, PP.163-168.
C
C     ..................................................................
C
C     SUBROUTINE FMFP(FUNCT,N,X,F,G,EST,EPS,LIMIT,IER,H)
C
C        DIMENSIONED DUMMY VARIABLES
      DIMENSION H(1),X(1),G(1)
C
C        COMPUTE FUNCTION VALUE AND GRADIENT VECTOR FOR INITIAL ARGUMENT
      CALL FUNCT(N,X,F,G)
C
C        RESET ITERATION COUNTER AND GENERATE IDENTITY MATRIX
      IER=0
      KOUNT=0
      N2=N+N
      N3=N2+N
      N31=N3+1
    1 K=N31
      DO 4 J=1,N
      H(K)=1.
      NJ=N-J
      IF(NJ)5,5,2
    2 DO 3 L=1,NJ
      KL=K+L
    3 H(KL)=0.
    4 K=KL+1
C
C        START ITERATION LOOP
    5 KOUNT=KOUNT +1
C
C        SAVE FUNCTION VALUE, ARGUMENT VECTOR AND GRADIENT VECTOR
      OLDF=F
      DO 9 J=1,N
      K=N+J
      H(K)=G(J)
      K=K+N
      H(K)=X(J)
C
C        DETERMINE DIRECTION VECTOR H
```

```
      K=J+N3
      T=0.
      DO 8 L=1,N
      T=T-G(L)*H(K)
      IF(L-J)6,7,7
    6 K=K+N-L
      GO TO 8
    7 K=K+1
    8 CONTINUE
    9 H(J)=T
C
C        CHECK WHETHER FUNCTION WILL DECREASE STEPPING ALONG H.
      DY=0.
      HNRM=0.
      GNRM=0.
C
C        CALCULATE DIRECTIONAL DERIVATIVE AND TESTVALUES FOR DIRECTION
C        VECTOR H AND GRADIENT VECTOR G.
      DO 10 J=1,N
      HNRM=HNRM+ABS(H(J))
      GNRM=GNRM+ABS(G(J))
   10 DY=DY+H(J)*G(J)
C
C        REPEAT SEARCH IN DIRECTION OF STEEPEST DESCENT IF DIRECTIONAL
C        DERIVATIVE APPEARS TO BE POSITIVE OR ZERO.
      IF(DY)11,51,51
C
C        REPEAT SEARCH IN DIRECTION OF STEEPEST DESCENT IF DIRECTION
C        VECTOR H IS SMALL COMPARED TO GRADIENT VECTOR G.
   11 IF(HNRM/GNRM-EPS)51,51,12
C
C        SEARCH MINIMUM ALONG DIRECTION H
C
C        SEARCH ALONG H FOR POSITIVE DIRECTIONAL DERIVATIVE
   12 FY=F
      ALFA=2.*(EST-F)/DY
      AMBDA=1.
C
C        USE ESTIMATE FOR STEPSIZE ONLY IF IT IS POSITIVE AND LESS THAN
C        1. OTHERWISE TAKE 1. AS STEPSIZE
      IF(ALFA)15,15,13
   13 IF(ALFA-AMBDA)14,15,15
   14 AMBDA=ALFA
   15 ALFA=0.
C
C        SAVE FUNCTION AND DERIVATIVE VALUES FOR OLD ARGUMENT
   16 FX=FY
      DX=DY
C
C        STEP ARGUMENT ALONG H
      DO 17 I=1,N
   17 X(I)=X(I)+AMBDA*H(I)
C
C        COMPUTE FUNCTION VALUE AND GRADIENT FOR NEW ARGUMENT
```

```
      CALL FUNCT(N,X,F,G)
      FY=F
C
C        COMPUTE DIRECTIONAL DERIVATIVE DY FOR NEW ARGUMENT.  TERMINATE
C        SEARCH, IF DY IS POSITIVE. IF DY IS ZERO THE MINIMUM IS FOUND
      DY=0.
      DO 18 I=1,N
   18 DY=DY+G(I)*H(I)
      IF(DY)19,36,22
C
C        TERMINATE SEARCH ALSO IF THE FUNCTION VALUE INDICATES THAT
C        A MINIMUM HAS BEEN PASSED
   19 IF(FY-FX)20,22,22
C
C        REPEAT SEARCH AND DOUBLE STEPSIZE FOR FURTHER SEARCHES
   20 AMBDA=AMBDA+ALFA
      ALFA=AMBDA
C        END OF SEARCH LOOP
C
C        TERMINATE IF THE CHANGE IN ARGUMENT GETS VERY LARGE
      IF(HNRM*AMBDA-1.E10)16,16,21
C
C        LINEAR SEARCH TECHNIQUE INDICATES THAT NO MINIMUM EXISTS
   21 IER=2
      RETURN
C
C        INTERPOLATE CUBICALLY IN THE INTERVAL DEFINED BY THE SEARCH
C        ABOVE AND COMPUTE THE ARGUMENT X FOR WHICH THE INTERPOLATION
C        POLYNOMIAL IS MINIMIZED
   22 T=0.
   23 IF(AMBDA)24,36,24
   24 Z=3.*(FX-FY)/AMBDA+DX+DY
      ALFA=AMAX1(ABS(Z),ABS(DX),ABS(DY))
      DALFA=Z/ALFA
      DALFA=DALFA*DALFA-DX/ALFA*DY/ALFA
      IF(DALFA)51,25,25
   25 W=ALFA*SQRT(DALFA)
      ALFA=DY-DX+W+W
      IF(ALFA) 250,251,250
  250 ALFA=(DY-Z+W)/ALFA
      GO TO 252
  251 ALFA=(Z+DY-W)/(Z+DX+Z+DY)
  252 ALFA=ALFA*AMBDA
      DO 26 I=1,N
   26 X(I)=X(I)+(T-ALFA)*H(I)
C
C        TERMINATE, IF THE VALUE OF THE ACTUAL FUNCTION AT X IS LESS
C        THAN THE FUNCTION VALUES AT THE INTERVAL ENDS. OTHERWISE REDUCE
C        THE INTERVAL BY CHOOSING ONE END-POINT EQUAL TO X AND REPEAT
C        THE INTERPOLATION.  WHICH END-POINT IS CHOOSEN DEPENDS ON THE
C        VALUE OF THE FUNCTION AND ITS GRADIENT AT X
C
      CALL FUNCT(N,X,F,G)
      IF(F-FX)27,27,28
```

```
   27 IF(F-FY)36,36,28
   28 DALFA=0.
      DO 29 I=1,N
   29 DALFA=DALFA+G(I)*H(I)
      IF(DALFA)30,33,33
   30 IF(F-FX)32,31,33
   31 IF(DX-DALFA)32,36,32
   32 FX=F
      DX=DALFA
      T=ALFA
      AMBDA=ALFA
      GO TO 23
   33 IF(FY-F)35,34,35
   34 IF(DY-DALFA)35,36,35
   35 FY=F
      DY=DALFA
      AMBDA=AMBDA-ALFA
      GO TO 22
C
C        TERMINATE, IF FUNCTION HAS NOT DECREASED DURING LAST ITERATION
   36 IF(OLDF-F+EPS)51,38,38
C
C        COMPUTE DIFFERENCE VECTORS OF ARGUMENT AND GRADIENT FROM
C        TWO CONSECUTIVE ITERATIONS
   38 DO 37 J=1,N
      K=N+J
      H(K)=G(J)-H(K)
      K=N+K
   37 H(K)=X(J)-H(K)
C
C        TEST LENGTH OF ARGUMENT DIFFERENCE VECTOR AND DIRECTION VECTOR
C        IF AT LEAST N ITERATIONS HAVE BEEN EXECUTED. TERMINATE, IF
C        BOTH ARE LESS THAN  EPS
      IER=0
      IF(KOUNT-N)42,39,39
   39 T=0.
      Z=0.
      DO 40 J=1,N
      K=N+J
      W=H(K)
      K=K+N
      T=T+ABS(H(K))
   40 Z=Z+W*H(K)
      IF(HNRM-EPS)41,41,42
   41 IF(T-EPS)56,56,42
C
C        TERMINATE, IF NUMBER OF ITERATIONS WOULD EXCEED  LIMIT
   42 IF(KOUNT-LIMIT)43,50,50
C
C        PREPARE UPDATING OF MATRIX H
   43 ALFA=0.
      DO 47 J=1,N
      K=J+N3
      W=0.
```

```
      DO 46 L=1,N
      KL=N+L
      W=W+H(KL)*H(K)
      IF(L-J)44,45,45
   44 K=K+N-L
      GO TO 46
   45 K=K+1
   46 CONTINUE
      K=N+J
      ALFA=ALFA+W*H(K)
   47 H(J)=W
C
C        REPEAT SEARCH IN DIRECTION OF STEEPEST DESCENT IF RESULTS
C        ARE NOT SATISFACTORY
      IF(Z*ALFA)48,1,48
C
C        UPDATE MATRIX H
   48 K=N31
      DO 49 L=1,N
      KL=N2+L
      DO 49 J=L,N
      NJ=N2+J
      H(K)=H(K)+H(KL)*H(NJ)/Z-H(L)*H(J)/ALFA
   49 K=K+1
      GO TO 5
C        END OF ITERATION LOOP
C
C        NO CONVERGENCE AFTER  LIMIT  ITERATIONS
   50 IER=1
      RETURN
C
C        RESTORE OLD VALUES OF FUNCTION AND ARGUMENTS
   51 DO 52 J=1,N
      K=N2+J
   52 X(J)=H(K)
      CALL FUNCT(N,X,F,G)
C
C        REPEAT SEARCH IN DIRECTION OF STEEPEST DESCENT IF DERIVATIVE
C        FAILS TO BE SUFFICIENTLY SMALL
      IF(GNRM-EPS)55,55,53
C
C        TEST FOR REPEATED FAILURE OF ITERATION
   53 IF(IER)56,54,54
   54 IER=-1
      GOTO 1
   55 IER=0
   56 RETURN
      END

C     ******************************************


C     SYSTEMATIC SEARCH ARONC THE MINIMUM PREVIOUSLY FOUND
```

```
      DIMENSION X(2)
      UU2=64./10000.
      HH2=50.
      VV2=50.
      T=1.
      X(1)=0.
      P=0.1
      A=1.+P*T
      WS=3.1416/2520.
      WS2=WS**2
      DO 54 I=1,9
      X(1)=X(1)+0.2
      E2=X(1)*X(1)
       E3=E2*X(1)
      X(2)=0.
       DO 53 J=1,9
      X(2)=X(2)+0.2
      F2=X(2)*X(2)
      F3=F2*X(2)
      UN=F3*X(1)*P*P*T+F2*E2*P*T*A+E3*P*T*A+E3*X(2)*P*T*T
      DE=-2.*X(1)*X(2)*A*P*T+X(2)*A*P*P+E2*X(2)*A*T*T
      VVV=2.*T*HH2*UN/DE+UU2/E2+VV2*(1.+2.*WS2/X(1))**2
      WRITE(6,52)X(1),X(2),VVV
 52   FORMAT(3X,E17.7,E17.7,E17.7)
 53   CONTINUE
  54  CONTINUE
      CALL EXIT
      END
```

```
C
C      PROGRAM NC. 3
C
C
C      SOLUTION OF THE KALMAN FILTER EQUATIONS FOR THE DISCRETE CASE
C
C
C      PHI=TRANSITION MATRIX
C      H=MEASUREMENT MATRIX
C      COVN=COVARIANCE MATRIX OF THE DRIVING NOISE
C      P=COVARIANCE MATRIX OF THE STATE AFTER A MEASUREMENT
C      PP=CVARIANCE MATRIX OF THE STATE BEFORE A MEASUREMENT
C      R=COVARIANCE MATRIX OF THE MEASUREMENT NOISE
C      N=DIMENSICN OF THE STATE VARIABLES
C      M=DIMENSICN OF THE MEASUREMENT
C      G=KALMAN GAIN
C
C
      DIMENSION PHI(36),COVN(21),H(18),P(21),PP(21),G(18),R(6)
      DOUBLE PRECISION PHI,COVN,H,P,PP,G,R,TO1,TO2,TO3,TO4,TO5,TO6,TO8,
     1BBB,CCC,DDD,BET,GR,U,T,TO,DSQRT,DSIN,DCOS,DEXP,DABS
      INTEGER TEST
 1000 FORMAT(6D13.6)
 1001 FORMAT(/,6(2X,1PD12.5))
  102 FORMAT(/,2X,1PD12.5,/,2(2X,1PD12.5),/,3(2X,1PD12.5),/,
     14(2X,1PD12.5),/,5(2X,1PD12.5),/,6(2X,1PD12.5))
  106 FORMAT(/,' KALMAN GAIN ')
  109 FORMAT(10(2X,1PD12.5))
      CALL TRAPS(0,0,100000)
      N=6
      M=3
      BBB=0.25
      DDD=(3.1416/300.)**2
      CCC=3.1416*0.2/300.
      BET=(DSQRT(4.*DDD-CCC**2))/2.
      GR= 981./(12.*2.54)
      T=2.5
      TO=10.
      R(1)=(GR/1000.)**2
      R(2)=0.
      R(3)=R(1)/1000.
      R(4)=0.
      R(5)=0.
      R(6)=32500.
      U=250000.*(2.*DDD*(-2.*DDD+(1.+2.*CCC)*(CCC+2.*DDD)))/(DDD*
     12.*(1.+2.*CCC))
      U=U*2.5
      DO 107 JJJ=1,21
  107 COVN(JJJ)=0.
      COVN(10)=U
      DO 695 K=1,18
  695 H(K)=0.
      H(1)=1.
      H(2)=1.
```

```
      H(4)=-DDD
      H(6)=1.
      H(7)=-CCC
      H(10)=1.
      H(14)=1.
      H(18)=1.
      WRITE(6,1001)H
      TEST=-1
      DO 101 K=1,21
  101 P(K)=0.
      P(1)=250000.
      P(3)=32500.
      P(6)=20.
      P(10)=10.
      P(15)=250000.
      P(21)=32500.
      PHI(8)=DDD*DEXP(-CCC*T/2.)*DSIN(BET*T)/BET
      PHI(7)=(DCOS(BET*T)+(-CCC/(2.*BET))*DSIN(BET*T))*DEXP(-CCC*
     1T/2.)
      PHI(6)=(DEXP(-T/TO)+(DEXP(-CCC*T/2.))*(-DCOS(BET*T)+(-CCC/2.+1./TO
     1)*DSIN(BET*T)/BET))*(1./TO-CCC)/((-CCC/2.+1./TO)**2+BET**2)-
     2PHI(8)/DDD
      PHI(12)=-PHI(8)/DDD
      PHI(13)=(DCOS(BET*T)-CCC/(2.*BET)*DSIN(BET*T))*DEXP(-CCC*T/2.)
      PHI(11)=(PHI(6)+PHI(8)/DDD)/(1./TO-CCC)
      PHI(17)=(-DEXP(-BBB*T)-(DEXP(-CCC*T/2.))*(-DCOS(BET*T)+(-CCC/2.+
     1BBB)*DSIN(BET*T)/BET))/((-CCC/2.+BBB)**2+BET**2)
      PHI(18)=PHI(12)+BBB*PHI(17)
      PHI(16)=-DEXP(-T/TO)/((BBB-1./TO)*((-CCC/2.+1./TO)**2+BET**2))-
     1DEXP(-BBB*T)/((1./TO-BBB)*((BBB-CCC/2.)**2+BET**2))-DEXP(-CCC*T/2.
     2)*((-CCC-BBB-1./TO)*DCOS(BET*T)+(1./BET*(1./TO-CCC/2.)*(BBB-CCC/
     32.)-BET**2)*DSIN(BET*T))/(((BBB-CCC/2.)**2+BET**2)*((1./TO-CCC/2.
     4)**2+BET**2))
      TO1=PHI(6)
      TO2=PHI(7)
      TO3=PHI(8)
      TO4=PHI(13)
      TO5=PHI(16)
      TO6=PHI(17)
      TO8=PHI(11)
      DO 108 K=2,36
  108 PHI(K)=0.
      PHI(1)=DEXP(-T/TO)
      PHI(7)=-TO1/TO
      PHI(8)=TO2
      PHI(9)=-TO3
      PHI(13)=TO8/TO
      PHI(14)=-PHI(9)/DDD
      PHI(15)=TC4
      PHI(19)=-TO5/TO
      PHI(20)=TO6
      PHI(21)=TO6
      PHI(22)=DEXP(-BBB*T)
      PHI(29)=1.
```

```
      PHI(31)=(1.-DEXP(-T/TO))/TO
      PHI(36)=1.
      WRITE(6,1001)PHI
      TTT=0.
      PP(21)=1.
      PP(3)=1.
      PP(17)=1.
      P(17)=-17500.
      DO 100 III=1,200
      CALL              POCDE(N,M,PHI,COVN,H,R,P,PP,G,TEST)
      TEST=1
      WRITE(6,102)P
      QQQ=      DSQRT(PP(3)+PP(21)+2.*PP(17))
      PPP=      DSQRT(P(3)+P(21)+2.*P(17))
      PUNCH 699,TTT,QQQ
      PUNCH 699,TTT,PPP
  699 FORMAT(2(2X,1PE12.5))
      TTT=TTT+T
  100 CONTINUE
      WRITE(6,106)
      WRITE(6,1001)G
      CALL EXIT
      END
C
C     ******************************************
C     ******************************************
C
       SUBROUTINE POCDE(N,M,PHI,COVN,H,R,P,PP,G,TEST)
       INTEGER TEST
       DIMENSION PHI(1),COVN(1),H(1),R(1),P(1),PP(1),G(1)
C
C      THE FOLLOWING DUMMY ARRAYS ARE USED ONLY WITHIN THIS SUBROUTINE
C
      DIMENSICN AA(36),BB(36),CC(21),PHIP(36),PPHIP(36),F(36),FT(36)
      DIMENSION PPHIT(36)
      DOUBLE PRECISION PHI,COVN,H,R,P,PP,G,AA,BB,CC,PHIP,PPHIP,F,FT
      DOUBLE PRECISION DABS,PPHIT
   71  FORMAT('IERC = ',I4)
   72  FORMAT('THE MATRIX  HPPHT + R  WAS NOT INVERTED PROPERLY')
       NN = N*N
       LN = N*(N+1)/2
       MM = M*M
       NM = N*M
       LM = M*(M+1)/2
      EPS=0.000001
      IERC=1
       IF(TEST) 13,28,27
   27 EPS=0.000001
       CALL MPRD(PHI,P,PHIP,N,N,0,1,N)
       CALL MTRA(PHIP,PPHIT,N,N,0)
       CALL MPRD(PHI,PPHIT,AA,N,N,0,0,N)
C
C      THE PRODUCT  PHI*P*PHIT  IS TEMPORARILY STORRED IN AA.  THIS
C      RESULT SHOULD BE SYMMETRIC.  HOWEVER, DUE TO THE EFFECT OF
```

```
C       ROUND OFF ERRORS THIS MAY NOT BE THE CASE.  THE MATRIX IS MADE
C       SYMMETRICAL BELOW.
C
        CALL MTRA(AA,BB,N,N,0)
        DO 21 I = 1,NN
   21   BB(I) = .5*(AA(I) + BB(I))
C
C       STORE MATRIX  PHI*P*PHIT  WHICH IS NOW IN BB IN SYMMETRIC
C       STORAGE MODE AS PP.
C
        CALL MSTR(BB,PP,N,0,1)
C
C       COMPUTE THE COVARIANCE MATRIX JUST BEFORE A MEASUREMENT.
C
        DO 31 I = 1,LN
   31   PP(I) = PP(I) + COVN(I)
C
C       THE COMPUTATION OF THE OPTIMAL FILTER GAIN MATRIX FOLLOWS.
C
   28   CALL MPRD(H,PP,FT,M,N,0,1,N)
        CALL MTRA(FT,F,M,N,0)
        CALL MPRD(H,F,AA,M,N,0,0,M)
C
C       MAKE  AA  A SYMMETRIC MATRIX.
C
        CALL MTRA(AA,BB,M,M,0)
        DO 22 I = 1,MM
   22   BB(I) = .5*(BB(I) + AA(I))
        CALL MSTR(BB,CC,M,0,1)
        DO 32 I = 1,LM
   32   CC(I) = CC(I) + R(I)
      CALL DSINV(CC,M,EPS,IERC)
        IF(IERC) 33,34,33
   33   WRITE(6,71) IERC
        WRITE(6,72)
        GO TO 13
   34   CONTINUE
        CALL MPRD(F,CC,G,N,M,0,1,M)
C
C       COMPUTE THE COVARIANCE MATRIX FOLLOWING A MEASUREMENT.
C
        CALL MPRD(G,H,AA,N,M,0,0,N)
        DO 29 I=1,NN
   29   AA(I) = -AA(I)
        DO 25 I= 1,N
        J = (I-1)*N + I
   25   AA(J) = 1. + AA(J)
        CALL MPRD(AA,PP,BB,N,N,0,1,N)
        CALL MTRA(BB,F,N,N,0)
        CALL MPRD(AA,F,BB,N,N,0,0,N)
        CALL MPRD(G,R,F,N,M,0,1,M)
        CALL MTRA(F,FT,N,M,0)
        CALL MPRD(G,FT,AA,N,M,0,0,N)
        DO 26 I=1,NN
```

```
   26  AA(I) = AA(I) + BB(I)
C
C      MAKE  AA  A SYMMETRIC MATRIX.
C
       CALL MTRA(AA,BB,N,N,0)
       DO 23 I = 1,NN
   23  BB(I) = .5*(BB(I) + AA(I))
       CALL MSTR(BB,P,N,0,1)
      RETURN
   13 IBCD=1/(1+IERC)
      RETURN
       END
C
C     ******************************************
C
      SUBROUTINE DSINV(A,N,EPS,IER)
       DIMENSION A(1)
      DOUBLE PRECISION A,DIN,WORK
      CALL DMFSD(A,N,EPS,IER)
       IF(IER) 9,1,1
    1  IPIV = N*(N+1)/2
       IND = IPIV
       DO 6 I=1,N
      DIN=1.D0/A(IPIV)
       A(IPIV) = DIN
       MIN = N
       KEND = I - 1
       LANF = N - KEND
       IF(KEND) 5,5,2
    2  J = IND
       DO 4 K=1,KEND
       WORK = 0.D0
       MIN = MIN - 1
       LHOR = IPIV
       LVER = J
       DO 3 L=LANF,MIN
       LVER = LVER + 1
       LHOR = LHOR + L
    3  WORK = WORK +        A(LVER)*A(LHOR)
       A(J) = -WORK*DIN
    4  J = J - MIN
    5  IPIV = IPIV - MIN
    6  IND = IND - 1
       DO 8 I=1,N
       IPIV = IPIV + I
       J = IPIV
       DO 8 K=I,N
       WORK = 0.D0
       LHOR = J
       DO 7 L=K,N
       LVER = LHOR + K - I
       WORK = WORK +        A(LHOR)*A(LVER)
    7  LHOR = LHOR + L
       A(J) = WORK
```

```
    8  J = J + K
    9  RETURN
       END
C
C      ******************************************
C
       SUBROUTINE DMFSD(A,N,EPS,IER)
       DIMENSION A(1)
       DOUBLE PRECISION DPIV,DSUM,A,DSQRT
       IF(N-1) 12,1,1
    1  IER = 0
       KPIV = 0
       DO 11 K=1,N
       KPIV = KPIV + K
       IND = KPIV
       LEND = K - 1
      TOL=ABS(EPS*SNGL(A(KPIV)))
       DO 11 I=K,N
       DSUM = 0.D0
       IF(LEND) 2,4,2
    2  DC 3 L=1,LEND
       LANF = KPIV - L
       LIND = IND - L
    3  DSUM = DSUM +       A(LANF)*A(LIND)
    4  DSUM =       A(IND)   - DSUM
       IF(I-K) 10,5,10
    5  IF(SNGL(DSUM) - TOL) 6,6,9
    6  IF(DSUM) 12,12,7
    7  IF(IER) 8,8,9
    8  IER = K - 1
    9  DPIV = DSQRT(DSUM)
       A(KPIV) = DPIV
       DPIV = 1.D0/DPIV
       GO TO 11
   10  A(IND) = DSUM*DPIV
   11  IND = IND + I
       RETURN
   12  IER = -1
       RETURN
       END
C      ******************************************
C
C      ******************************************
C
      SUBROUTINE MPRD(A,B,R,N,M,MSA,MSB,L)
      DIMENSION A(1),B(1),R(1)
      DOUBLE PRECISION A,B,R
      MS=MSA*10+MSB
      IF(MS-22)30,10,30
   10 DO 20 I=1,N
   20 R(I)=A(I)*B(I)
      RETURN
   30 IR=1
      DO 90 K=1,L
```

```
      DO 90 J=1,N
      R(IR)=0
      DO 80 I=1,M
      IF(MS)40,60,40
   40 CALL LOC(J,I,IA,N,M,MSA)
      CALL LCC(I,K,IB,M,L,MSB)
      IF(IA)50,80,50
   50 IF(IB)70,80,70
   60 IA=N*(I-1)+J
      IB=M*(K-1)+I
   70 R(IR)=R(IR)+A(IA)*B(IB)
   80 CONTINUF
   90 IR=IR+1
      RETURN
      END
C
C     ******************************************
C
      SUBROUTINE MTRA(A,R,N,M,MS)
      DIMENSION A(1),R(1)
      DOUBLE PRECISICN A,R
      IF(MS)10,20,10
   10 CALL MCPY(A,R,N,N,MS)
      RETURN
   20 IR=0
      DO 30 I=1,N
      IJ=I-N
      DO 30 J=1,M
      IJ=IJ+N
      IR=IR+1
   30 R(IR)=A(IJ)
      RETURN
      END
C
C     ******************************************
C
      SUBROUTINE MSTR(A,R,N,MSA,MSR)
      DIMENSION A(1),R(1)
      DOUBLE PRECISION A,R
      DO 20 I=1,N
      DO 20 J=1,N
      IF(MSR)5,10,5
    5 IF(I-J)10,10,20
   10 CALL LOC(I,J,IR,N,N,MSR)
      IF(IR)20,20,15
   15 R(IR)=0.0
      CALL LOC(I,J,IA,N,N,MSA)
      IF(IA)20,20,18
   18 R(IR)=A(IA)
   20 CONTINUE
      RETURN
      END
C
C     ******************************************
```

```
C
      SUBROUTINE MCPY(A,R,N,M,MS)
      DIMENSION A(1),R(1)
      DOUBLE PRECISION A,R
      CALL LOC(N,M,IT,N,M,MS)
      DO 1 I=1,IT
    1 R(I)=A(I)
      RETURN
      END
C
C     *********************************************
C
C     SUBROUTINE LOC : I.B.M. SSP SUBROUTINE
C
C     *********************************************
C
```

LIST OF REFERENCES

1 BROXMEYER , C. " Inertial navigation systems" Mc Graw Hill , 1964

2 LANING ,J.H. and BATTIN ,R.H. , " Random processes in automatic control " , Mc Graw Hill, 1956

3 BATTIN , R.H. " Astronautical guidance " Mc Graw Hill , 1964

4 BRITTING ,K.R. " Airborne gravimeter specific compensation " Report RE-35 . Experimental Astronomy Laboratory , July 1966

5 BRITTING , K.R. " Class reference notes ; 16.42 Inertial Guidance " Massachusetts Institute of Technology , Spring 1969

6 BRITTING , K.R. " Unpublished notes "

7 WHALLEY , R. " Inertial techniques in height measurement " Height measurement in supersonic aircraft - Institute of navigation journal 1964

8 NEWTON , G., GOULD ,L., KAISER ,J., " Analytical design of linear feedback controls " New - York , Wiley 1957

9 KALMAN , R.E., BUCY ,R.S., " New results in linear filtering and prediction theory " Journal of basic engineering - March 1961

10 KALMAN , R.E., " A new approach to linear filtering and prediction problems " Journal of basic engineering , March 1960

11 BRYSON , A.E., JOHANSEN , D.H. , " Linear filtering for time varying systems using measurements containing colored noise " IEEE Transactions on automatic control , January 1965

12 BROCK , L.D., " Application of statistical estimation to navigation systems " Ph.D Thesis - T414 - Course 16 - M.I.T. - 1965

13 BLACKLOCK , J., " Automatic control of aircraft and missiles " New-York , Wiley 1965

14 SEAMANS,R.C., BARNES,F.A., GARBER,T.B., HOWARD,V.W., " Recent developements in aircraft control " Journal of the aeronautical sciences , March 1955

15 FLETCHER ,R., POWELL ,M.J.D., " A rapidly convergent descent method for minimization " The computer journal, Vol. 6 , July 1963

16 GRACEY , W., " Survey of altitude measuring methods for the vertical separation of aircraft " N.A.S.A. TND-738 , March 1961

17 GREMILLET , M.J., " Navigation systems for aircraft and space vehicle " - Un nouveau radioaltimetre pour hautes altitudes - The Macmillan

Company - New-York 1962 -

18 HEISKANEN ,W.A., VENING MEINESZ ,F.A., " The earth and its gravity field " Mc Graw Hill , New - York 1958