NAL TM-266

UDC 519.28:
629.783.05:
007.3

航空宇宙技術研究所資料

TECHNICAL MEMORANDUM OF NATIONAL AEROSPACE LABORATORY

TM-266

カルマンフィルターによる軌道および誘導誤差解析のためのシミュレーションプログラム

村田正秋・志甫　徹・吉田正廣

1974 年 11 月

航空宇宙技術研究所
NATIONAL AEROSPACE LABORATORY

目　　次

カルマンフィルターによる軌道および誘導誤差解析のためのシミュレーションプログラム*

村田正秋**・志甫 徹**・吉田正廣**

Simulation Program for Rocket Trajectory and Guidance Error Analysis Using Kalman Filter

By Masaaki MURATA, Toru SHIHO and Masahiro YOSHIDA

ABSTRACT

In this report we present one phase of a study on error analysis of space trajectories, that is to say, how to build an error analysis program based on the modern estimation theory. The proposed method plays an essential role in the post-flight evaluations of the GN&C system of a space vehicle, and is applicable as well to the mission planning phase.

The simulation program thus constructed has the following capabilities:

(1) generating any guided trajectories, noisy radar tracking data and IMU outputs (by DATA GENERATOR), and

(2) conducting sensitivity analysis and guidance error recovery with regard to almost all guidance parameters in the "consider" option as well as in the "solve-for" option (by TRAJECTORY RE-CONSTRUCTOR).

In the appendices, the basic subroutines and some of the application programs for space guidance are presented with the lists for user's information.

0. まえがき

筆者らは過去数ケ年にわたって制御を考慮した軌道推定の研究を行ってきた。[1] 一方筆者のひとりは，誘導方程式の開発[2]および最適軌道の研究[3]にも関与した。これらの研究活動を通して，筆者らは誘導制御系の設計とその評価のためのプログラムの構造は如何を追求してきた。本報告はこの問題に対する筆者らの回答であり，理論展開としてはKalman－Bucyの推定理論を基礎にしている。また筆者らが作製した一通りのプログラムも公表した。プログラムでは，所要 incore メモリーを節約し，また計算時間を短縮するためにアルゴリズムおよびソフトウェアの設計に若干の工夫がなされている。もうひとつの特徴は，incore の割当てと coding を容易にするため殆んど全体を通して 1 次元 ARRAY を採用している。

もとよりこのような大規模なシミュレーションプログラムは筆者らの手に余る部分があまりにも多い。われわれが必要と認めながらも作製されていないプログラムも数多い。pre－flight phase における IMU の caliblation プログラムはその 1 例である。また本報告では injection phase までに限定しているが同様の接近をして近い将来には interplanetary flight まで拡張する予定で現在検討中である。

本プログラムの特徴と使用上の注意を述べる。

(1) GNT, GDT および GRT（これらの説明は第 4 節を参照）を任意に計算できる。尚 GNT の計算においては，われわれの興味は 2 段目以降の active な guidance phase にあるので，空気力の効果は考慮されていない。しかし user は目的に応じてより高精度な軌道発生プログラムと置き換えることができ

＊昭和49年6月21日
＊＊計測部

る。

(2) 雑音に乱されたレーダートラッキングデータおよびIMU出力をシミュレートできる。不規則雑音としては，任意に与えられた統計量をもつ白色雑音あるいは有色雑音を印加できる。

(3) 殆んどすべての誘導パラメータ，例えばIMUハードウェア誤差，レーダーシステム誤差，重力ポテンシャル誤差等に関する感度解析あるいは飛翔後の実データに基づく誘導誤差解析が可能である。これはKalman−Bucyの推定理論に基づく統計的手法によっている。

(4) アルゴリズムとして，塔載型誘導計算機が実際に行うのと同様のG＆N計算をシミュレートできる。航法計算はstrapdown方式およびstable platform方式両方について可能である。また誘導計算はサターン型の3次元explicit IGM(Iterative Guidance Mode)によっている。制御系は殆んど理想的に機能するものと仮定されるが，まだ推力の大きさと方向(すなわち推力軸のミスアラインメント)のfluctuationsは適当な不規則雑音を付加してシミュレートしている。

(5) 附録として我々が開発し常時使用している一連のプログラムをuserのためにリストとともに掲載した。

我々が新しく開発している実用的フィルターに関しては結果だけを報告している。その応用例に関しては別途報告する予定である。

1. 座標系・力学系

1.1 座標系

基本的な座標系を説明する。

(1) 基準座標系

平均北極方向にZ_I軸，赤道面とLAUNCH MERIDIANとの交わりをX_I軸とする地球中心の右手慣性座標系$O_E-X_I\,Y_I\,Z_I$を基準座標系に選び，以後この座標系をI座標系と呼ぶ。

(2) L座標系(Launch site coordinate)

図1に示すようにLaunch siteに原点O_Lを有し，その接平面上真東にY_L軸，真北にZ_L軸，鉛直上方にX_L軸の慣性座標系$O_L-X_L\,Y_L\,Z_L$をL座標系という。I座標系からL座標系への座標変換行列C_{LI}は，

$$C_{LI} \triangleq \begin{pmatrix} \cos\phi_L & 0 & \sin\phi_L \\ 0 & 1 & 0 \\ -\sin\phi_L & 0 & \cos\phi_L \end{pmatrix} \qquad (1)$$

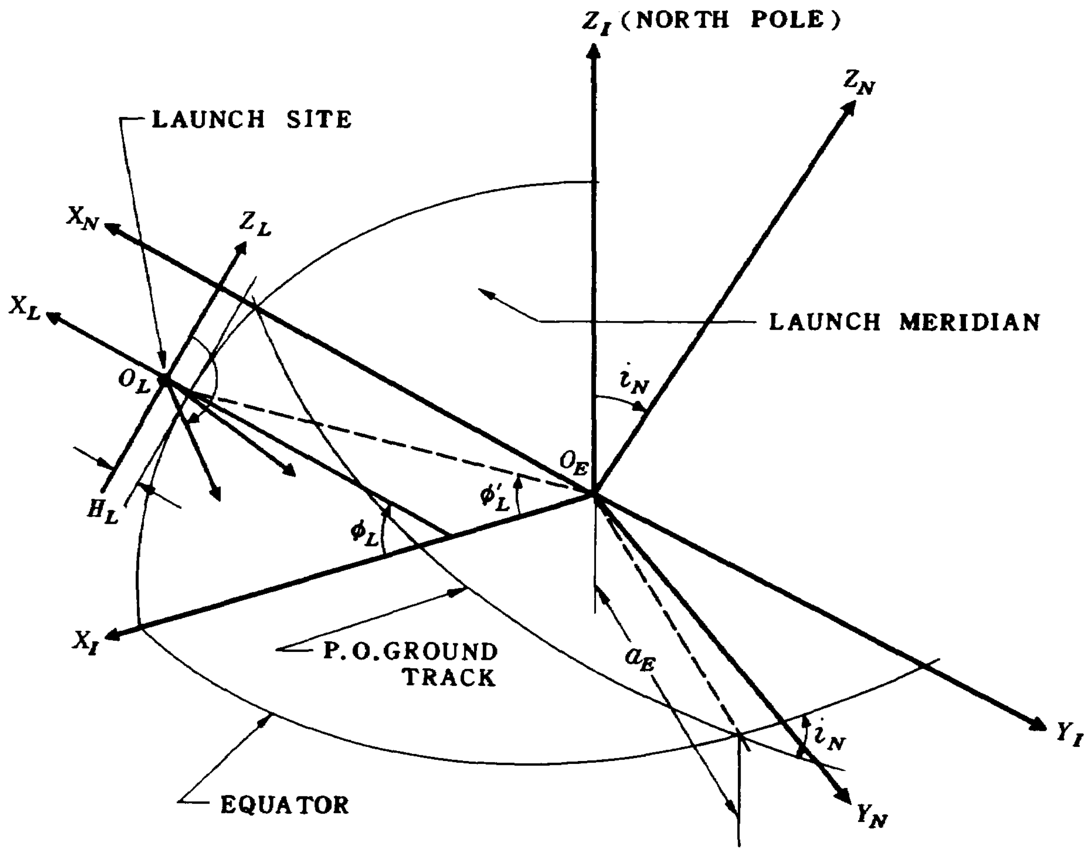

図1 座標系(その1)

で与えられる。ここでϕ_LはO_Lの測地学的緯度（geodetic latitude）であり，地理学的緯度（geograplric latitude）ϕ'_Lとは，地球の離心率をeとすると

$$\tan\phi_L = \frac{\tan\phi'_L}{1-e^2}$$

で結ばれている。ϕ'_Lを launch latitude ということもある。

(3) 航法座標系（Navigation coordinate）

航法計算の基準になる慣性座標系である。ミッションの各 phase に応じて種々の座標系が用いられるが，特に injection phase までは図1に示すように地球中心の，parking orbit（P.O.）面内でX_L方向にX_N軸，それに直交してY_N軸をもつ右手座標系$O_E-X_N\,Y_N\,Z_N$を使うと便利である。この座標系をN座標系という。L座標系からN座標系への座標変換行列C_{NL}は，

$$C_{NL}=\begin{pmatrix}1 & 0 & 0\\ 0 & \sin\Phi_L & \cos\Phi_L\\ 0 & -\cos\Phi_L & \sin\Phi_L\end{pmatrix} \tag{2}$$

である。ここにΦ_LはZ_L方向から測った方位角（Launch azimuth）である。P.O.面の傾斜（inclination）をi_Nとすると

$$\cos i_N = \sin\Phi_L \quad \cos\phi_L \tag{3}$$

が成り立つ。またI座標系からN座標系への座標変換行列C_{NI}は，$C_{NI}=C_{NL}\cdot C_{LI}$から

$$C_{NI}=\begin{pmatrix}\cos\phi_L & 0 & \sin\phi_L\\ -\cos\Phi_L\sin\phi_L & \sin\Phi_L & \cos\Phi_L\cos\phi_L\\ -\sin\Phi_L\sin\phi_L & -\cos\Phi_L & \sin\Phi_L\cos\phi_L\end{pmatrix} \tag{4}$$

で与えられる。

(4) 回転座標系

図2に示すように，機体重心に原点O_Rを有し，飛翔体の運動とともに回転する座標系$O_R-X_R\,Y_R\,Z_R$でこれをR座標系という。local horizon 面に対して，Z_R軸は垂直上方に，Y_R軸は面内でかつP.O.面に平行にとる。N座標系からR座標系への座標変換行列C_{RN}は，

$$C_{RN}=\begin{pmatrix}\sin\phi\cos\phi & \sin\phi\sin\phi & -\cos\phi\\ -\sin\phi & \cos\phi & 0\\ \cos\phi\cos\phi & \cos\phi\sin\phi & \sin\phi\end{pmatrix} \tag{5}$$

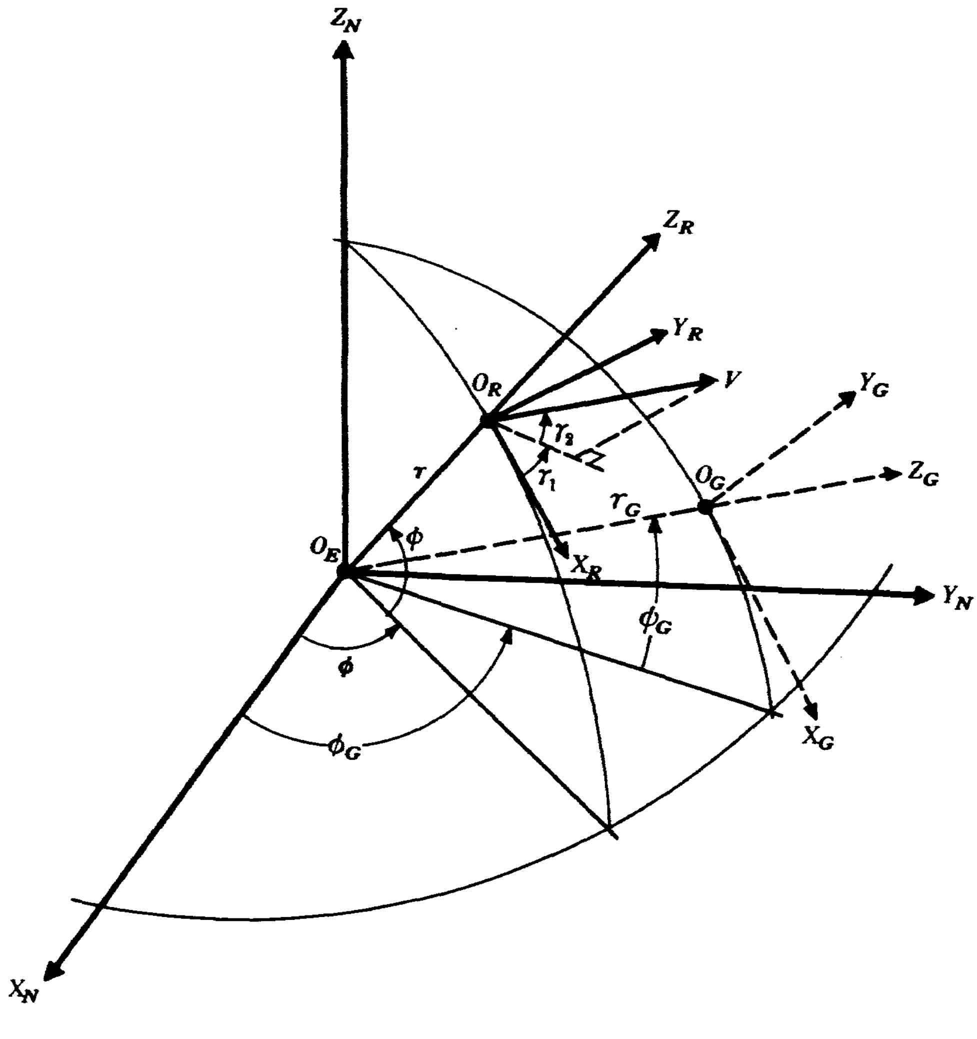

図2　座標系（その2）

となる。ここにϕは range angle，φはcross-range angle である。したがって飛翔体の位置ベクトル$\underline{r}$，速度ベクトル$\underline{v}$のR座標系からN座標系への変換は次式に従う。

$$\begin{aligned}\underline{r}_N &= C_{NR}\ \underline{r}_R \\ \underline{v}_N &= C_{NR}\ \underline{v}_R + \Omega_N\ C_{NR}\ \underline{r}_R\end{aligned} \qquad (6)$$

ここで

$$\underline{r}_R = \begin{pmatrix} 0 \\ 0 \\ r \end{pmatrix}, \qquad \underline{v}_R = \begin{pmatrix} 0 \\ 0 \\ \dot{r} \end{pmatrix}$$

$$\Omega_N = \begin{pmatrix} 0 & -\dot{\phi} & -\dot{\phi}\cos\varphi \\ \dot{\phi} & 0 & -\dot{\phi}\sin\varphi \\ \dot{\phi}\cos\varphi & \dot{\phi}\sin\varphi & 0 \end{pmatrix}$$

(5)　誘導座標系（Guidance Coordinate）

誘導を行う基準慣性座標系$O_G - X_G\,Y_G\,Z_G$でG座標系という。所望の injection（または stage separation）point O_Gに原点を有し，3軸の設定はR座標系と同様である。N座標系からG座標系への座標変換行列は

$$C_{GN} = \begin{pmatrix} \sin\varphi_G\cos\phi_G & \sin\varphi_G\sin\phi_G & -\cos\varphi_G \\ -\sin\phi_G & \cos\phi_G & 0 \\ \cos\varphi_G\cos\phi_G & \cos\varphi_G\sin\phi_G & \sin\varphi_G \end{pmatrix} \quad (7)$$

となる。ここでϕ_G，φ_Gはそれぞれ原点O_GのN座標系に関する range angle および cross－range angle である。

(6)　機体固定座標系（Body fixed coordinate）

図3に示すように機体軸（ロール軸）をZ_B，ヨー軸をY_Bとして機体に固定された座標系$O_B - X_B\,Y_B\,Z_B$をB座標系という。B座標系からN座標系（またはI座標系）への座標変換行列C_{NB}（またはC_{IB}）は常微分方程式

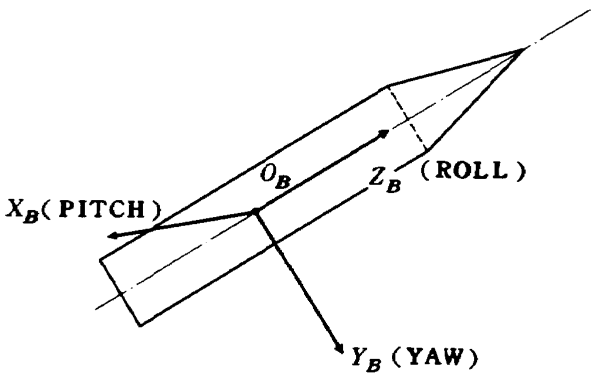

図3　座標系（その3）

$$\dot{C}_{NB}(t) = C_{NB}(t)\,[\,\underline{\omega}\times\,] \qquad (8)$$

に従う。ここで$\underline{\omega} = (\omega_x\ \omega_y\ \omega_z)'$は機体の角速度ベクトルの$B$座標での表現である。また一般に，任意の3次元ベクトル$\underline{a} = (a_x\ a_y\ a_z)'$に対して次式で定義される行列を$\underline{a}$のつくる回転行列とよぶ。

$$[\,\underline{a}\times\,] = \begin{pmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{pmatrix}$$

(7)　地球固定座標系（Earth fixed coordinate）

地球に固定され，自転とともに回転する座標系で打上げ瞬時にはI座標系に一致する回転座標系$O_E - X_E\,Y_E\,Z_E$をE座標系という。I座標系からE座標系への座標変換行列C_{EI}は次式で与えられる。

$$C_{EI} = \begin{pmatrix} \cos\omega_E t & \sin\omega_E t & 0 \\ -\sin\omega_E t & \cos\omega_E t & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad (9)$$

ここでω_Eは地球の自転速度とする。

(8)　レーダーサイト座標系

図4に示すようにレーダーサイトに原点O_Sをもち接平面内真東にX_S軸，真北方向にY_S軸および垂直上向きにZ_S軸をとった地球固定の回転座標系$O_S - X_S\ Y_S\ Z_S$をS座標系という。E座標系からS座標系への座標変換行列C_{SE}は次式で与えられる。

$$C_{SE} = \begin{pmatrix} -\sin\lambda_S & \cos\lambda_S & 0 \\ -\sin\phi_S\cos\lambda_S & -\sin\phi_S\sin\lambda_S & \cos\phi_S \\ \cos\phi_S\cos\lambda_S & \cos\phi_S\sin\lambda_S & \sin\phi_S \end{pmatrix} \quad (10)$$

ここにλ_S，ϕ_Sはそれぞれレーダーサイトの，E座標系のX_E軸から測った経度および測地学的緯度とする。またレーダーサイトのE座標系での位置ベクトル$\underline{R}_E$は

$$\underline{R}_E = \begin{pmatrix} G_1\ \cos\phi_E\ \cos\lambda_E \\ G_1\ \cos\phi_E\ \sin\lambda_E \\ G_2\ \sin\phi_E \end{pmatrix} \qquad (11)$$

ただし

$$G_1 = H_S + \frac{a_E}{\sqrt{1-e^2\sin^2\phi_E}}, \quad G_2 = H_S + \frac{a_E(1-e^2)}{\sqrt{1-e^2\sin^2\phi_E}}$$

ここでa_Eは地球の平均赤道半径を，H_Sはサイトの想定した規準回転楕円面からの垂直上向きの高さを表わし，また当然　$\phi_E = \phi_S$，$\lambda_E = \phi_E$　である。

(9)　IMU計測座標系（Instrumentation coordinate）

IMU，すなわち加速度計およびジャイロのそれぞれ入出力軸および pendulum軸（スピン軸）のつくる規準的（nominally）に直交する座標系をM座標系という。またIMUが stable platform を構成する場合に platform 上の3個の加速度計の入力軸がつくる座標系をP座標系という。（図5）。

(10)　軌道座標系（Trajectory coordinate）

適当な慣性座標系（すなわちN座標系またはI座標系）に関する飛翔体の位置ベクトル$\underline{r}$，速度ベクトル$\underline{v}$を使

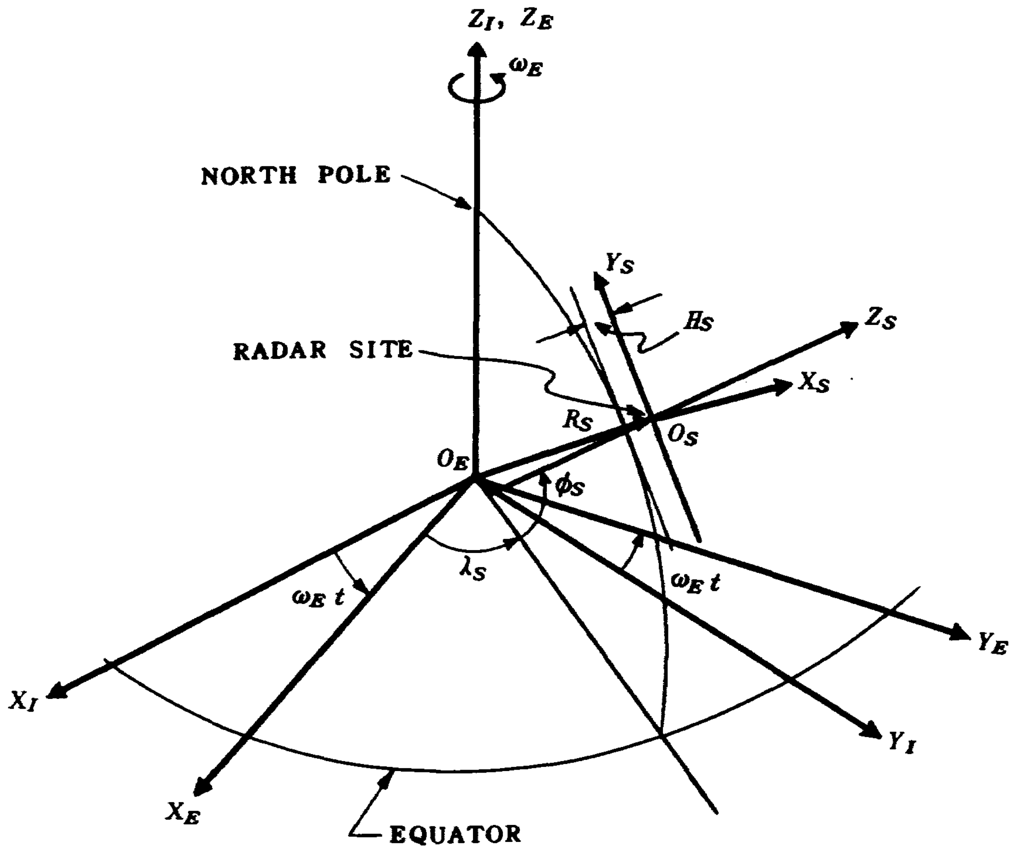

図4　座標系（その4）

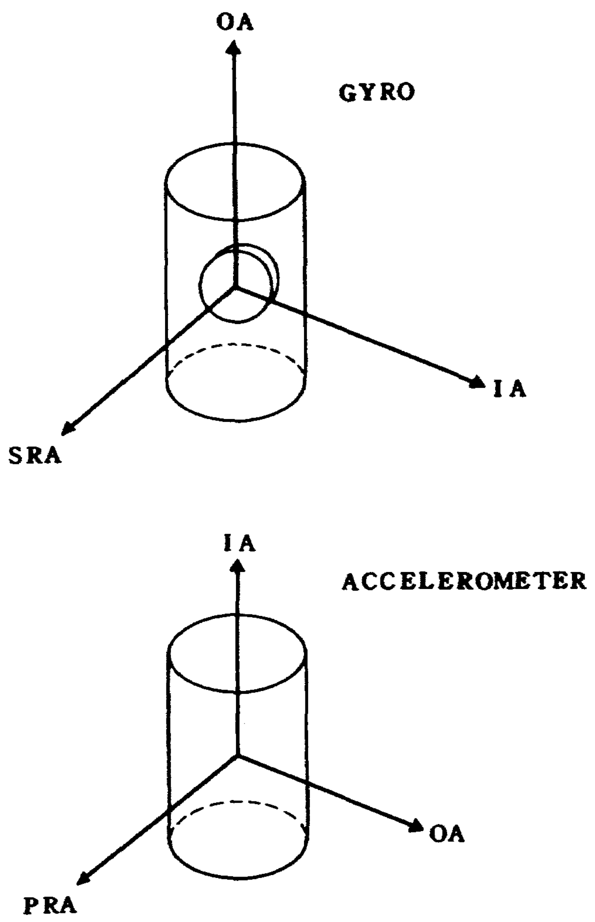

図5　座標系（その5）

って次のような直交座標系 $O_T-X_T\,Y_T\,Z_T$ を構成するとき，この座標系を T 座標系という。T 座標系では地球の扁球性を無視するならば，$\underline{X}_T$ が downrange 方向の，$\underline{Y}_T$ が cross-range 方向の，$\underline{z}_T$ が高度方向の単位ベクトルを与える。従ってたとえば I 座標系から T 座標系への座標変換行列 C_{TI} は

$$C_{TI}=\begin{pmatrix}\underline{X}'_T\\ \underline{Y}'_T\\ \underline{Z}'_T\end{pmatrix} \tag{12}$$

で与えられる。とくに経路角（flight path angle）を γ_T とすると

$$\cos\gamma_T=\frac{(\underline{v},\ \underline{x}_T)}{\|\underline{v}\|} \tag{13}$$

が成り立つ。

以上10項目の座標系は本報告で基本的に重要な役割を演じる。目的によっては別の座標系も必要になろうが誘導制御系のシンセシスおよびアナリシスを行う場合に最小限これだけは用意しておくべきであろう。

1.2　力 学 系

軌道発生　われわれのプログラムには多段大型ロケットの injection までの推力飛翔軌道を R 座標系において質点計算する軌道発生プログラムが補助的に用意されている。軌道発生は，目的に応じて2次元（pitch

面）および3次元ともに計算可能である。その際 IMU 入出力をシミュレートするため，ロケットの角速度ベクトルを発生するプログラムも内含されている。

2. 誘導制御系

2.1 航法系

航法（Navigation）とは，IMU，レーダートラッキングあるいは光学的センサーの計測データに基づいて，飛翔体の現時点における位置，速度および姿勢を決定することである。観測手段として何を用いるかに応じて当然航法計算のアルゴリズムは異なる。ここでは純慣性誘導の大型ロケットを想定して，IMU出力のみによる航法系について述べる。この場合航法計算の規準になる座標系の保持の仕方に応じて，stable platform方式とstrapdown方式のふたつがあることを既報告[1]の中でも説明した。

航法計算は N 座標系で行うのが普通である。両方式に対する運動方程式はそれぞれ次式で記述される。

stable platform方式

$$\ddot{\underline{R}}_N = \underline{g}_N + C_{NP}\ C_{PM}\ \underline{a}_M \tag{14}$$

ただし　$\underline{a}_M$：非重力加速度（M座標）

C_{NP}：$P \to N$ 座標変換行列（一定）

C_{PM}：$M \to P$ 座標変換行列*

$\underline{R}_N$：位置ベクトル

$\underline{v}_N$：速度ベクトル（$=\dot{\underline{R}}_N$）

$\underline{g}_N$：重力加速度（$=C_{NE}\ \underline{g}_E\ (\underline{R}_E)$，但し $\underline{R}_E = C_{EN}\ \underline{R}_N$）

strapdown 方式

$$\ddot{\underline{R}}_N = \underline{g}_N + C_{NB}\ C_{BM}\ \underline{a}_M \tag{15}$$

$$\dot{C}_{NB} = C_{NB}\ [\ \underline{\omega}_B \times\ ],\quad \underline{\omega}_B = C_{BM}\ \underline{\omega}_M \tag{16}$$

ただし　$\underline{a}_M$：非重力加速度（M座標）

$\underline{\omega}_M$：角速度　　（M座標）

C_{NB}：$B \to N$　座標変換行列

C_{BM}：$M \to B$　座標変換行列*

(14) 式の $\underline{a}_M$ はplatform上の加速度計によって計測される。(15) (16) 式の $\underline{a}_M$，$\underline{\omega}_M$ はstrapdown package内の加速度計およびジャイロによって計測される。後述するようにジャイロは出力軸に垂直な方向の加速度にセンシティブで大きな誤差を誘起するので両packageはできるだけこれによる誤差を小さくするように構成される。その一般的に最適な配置は既報告[1]で説明した。

さて航法計算は，補正されたIMU出力に基づいて (14) または (15) (16) を積分することにより行なわれる。得られた $\underline{R}_N$，$\underline{v}_N$ が誘導系への入力になり，同時にstable platform のジンバル角，あるいは C_{NB} が制御系に機体の姿勢情報を与える。

われわれのシミュレーションプログラムに用意されている航法計算アルゴリズムを次に示そう。いま

$$\delta\,\underline{v}_M = \int_t^{t+\Delta t} \underline{a}_M\ dt$$

$$\delta\,\underline{\theta}_M = \int_t^{t+\delta t} \underline{\omega}_M\ dt$$

とする。また

$$\left.\begin{aligned} \underline{R}_N(\,t+\Delta t\,) &= \underline{R}_N(t) + \delta\,\underline{R}_N\,\Delta t \\ \underline{V}_N(\,t+\Delta t\,) &= \underline{V}_N(t) + \delta\,\underline{V}_N\,\Delta t \end{aligned}\right\} \tag{17}$$

とするとき，

stable platform方式

$$\left.\begin{aligned} \delta\,\underline{R}_N &= \underline{V}_N(t) + \frac{1}{2}\left[\underline{g}_N(t)\,\Delta t + C_{NP}\ C_{PM}\ \delta\underline{v}_M\right] \\ \delta\,\underline{V}_N &= \frac{1}{2}\left[\underline{g}_N(t) + \underline{g}_N(t+\Delta t)\right] + C_{NP}\ C_{PM}\ \delta\underline{v}_M \end{aligned}\right\} \Big/ \Delta t \tag{18}$$

strapdown 方式

$$\left.\begin{aligned} \delta\,\underline{R}_N &= \underline{V}_N(t) + \frac{1}{2}\left[\underline{g}_N(t)\,\Delta t + \delta\underline{v}_N\right] \\ \delta\,\underline{V}_N &= \frac{1}{2}\left[\underline{g}_N(t) + \underline{g}_N(t+\Delta t)\right] + \delta\underline{v}_N / \Delta t \end{aligned}\right\} \tag{19}$$

$$\delta\,\underline{v}_N = C_{NB}\ C_{BM}\ \delta\,\underline{v}_M \tag{20}$$

$$C_{NB}(\,t+\delta t\,) = C_{NB}(t)\ \cdot\ A \tag{21}$$

$$A = I + [\,\delta\,\underline{\theta}_B \times\,] + \frac{1}{2}[\,\delta\,\underline{\theta}_B \times\,]^2 \tag{22}$$

$$\delta\,\underline{\theta}_B = C_{BM}\ \delta\,\underline{\theta}_M \tag{23}$$

(17) 式の update は，guidance cycle で行なえばよいが (21) 式の update は正確に計算する必要があって，通常 guidance cycle 内で 100 回程度繰り返す。$\underline{g}_E$ はその第3項以降が $10^{-4}g$ 程度で加速度計の精度と合致することから航法計算のためには高々 J_2 を含む項まで考慮すれば充分である。本プログラムでは第1項のみ組み込まれている。

2.2 誘導制御系

われわれのシミュレーションプログラムにはサターン型の3次元 explicit IGM（Iterative Guidance Mode）が用意されている。すなわち G 座標系で所望の injection（または separation）条件を指定すれば，それを終端条件として満足する最短時間（推力一定とすれば最少燃量）制御を実現するための推力指令角（steering command）を発生させることができる。IGMに関し

---

* C_{PM} および C_{BM} は規準的に直交し，しかも単位行列としてよい。

ては既に新田，大坪，松島および志甫による報告[2)]がある。

サターン型のIGMは，flat earthの仮定で変分法を使って得られる range angle free optimal steering であり次の2段階を経て決定される。

phase 1. range angle free & altitnde free

すなわち速度にのみ拘束を課した最短時間制御でその最適解は定推力角$\tilde{\chi}$を与える。$\tilde{\chi}$に微小な補正角δを附加して，終端でさらに与えられた位置（高度）をも満足するような最短時間制御を新たに考える。即ち

phase 2. range angle free only

この最適解$\tilde{\delta}$は一般に次の linear-tangent steering law として知られている。

$$\tan\tilde{\delta} = A + Bt \tag{24}$$

$\tilde{\delta}$は微小ゆえ

$$\tilde{\delta} \sim A + Bt$$

という近似が成り立ち，最終的に準最適な推力角

$$\chi = \tilde{\chi} + A + Bt \tag{25}$$

を得る。* ここで $0 \leq t \leq T$ とし，$\tilde{\chi}$，A，BおよびT（time-to-go）の値は後に導かれるように終端拘束条件および現時刻の位置および速度ベクトル（これらの諸値はIMU出力に基づく航法計算により知られる）の関数として表現される。

以上に述べた手続きに従って，われわれのシミュレーションプログラムに用意されている3次元誘導方程式を説明する。

誘導方程式はできるだけ簡単な表現を得るためにG座標系で導かれる。飛翔体の運動をG座標系で表現すると

$$\ddot{\underline{R}} = \underline{g}(\underline{R}) + \frac{F}{m}\underline{u} \tag{26}$$

$$\underline{u} = (\cos\alpha\ \sin\beta \quad \cos\alpha\ \cos\beta \quad \sin\alpha)' \tag{27}$$

となる（図6）。ここに

$\underline{R}$	；	位置ベクトル	(m)
$\underline{V}(=\dot{\underline{R}})$	；	速度ベクトル	(m/sec)
$\underline{g}(\underline{R})$	；	重力加速度	(m/sec)
$\underline{u}$	；	推力方向制御ベクトル	
α	；	ピッチ推力角	(rad)
β	；	ヨー推力角	(rad)
m	；	飛翔体の質量	(Kg)
F	；	飛翔体の推力	(Kg・重)

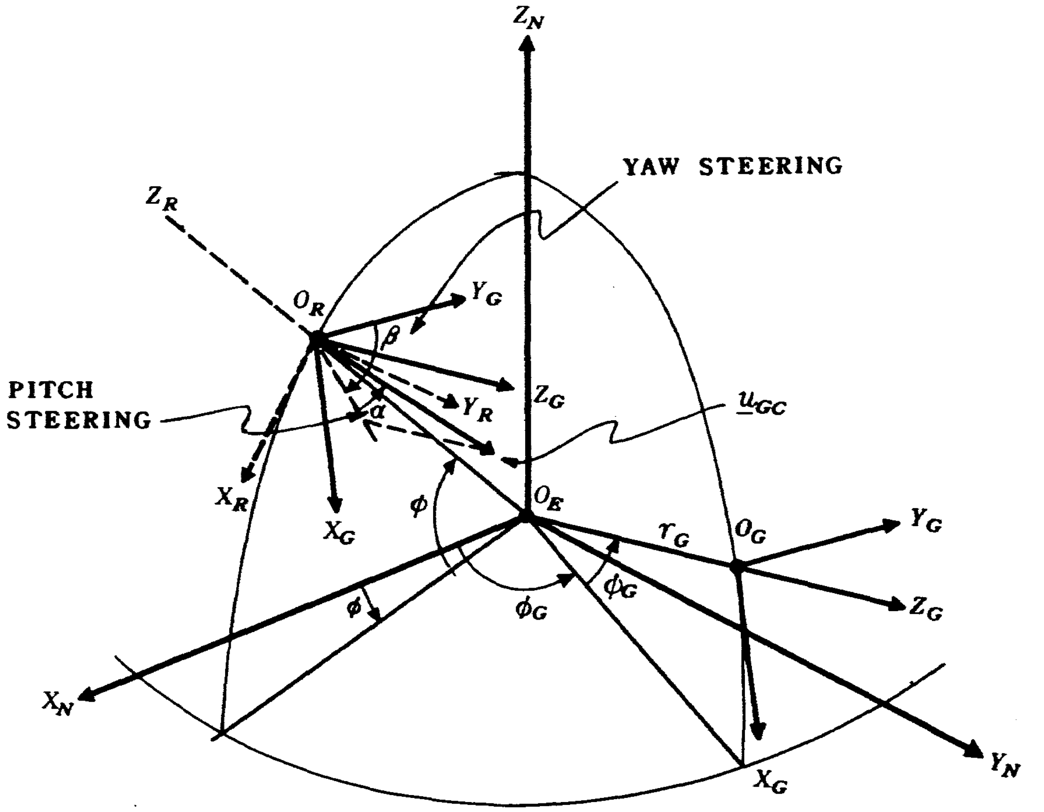

図6 G座標系における推力角

* 新田他による誘導方程式は phase 2. で $\int_0^T \delta^2\,dt$ を最小にするものである。シミュレーションによって誘導可能領域が(25)の場合よりも拡張されたと報告している。

$(\dot{\ })=\dfrac{d}{dt}$; $0\leq t\leq T$

通常誘導は飛翔体が空気抵抗の無視しうるに充分な高度に達してから開始されるので，運動方程式の非重力項としては(26)式のように推力加速度のみを考慮する。さらに $\underline{u}$ の解析的表現を得るために，高度の F への影響を無視し，また $\underline{g}(\underline{R})$ は第1項まで採用し平均化する（average $\underline{g}$ method）。

$$\underline{g}(\underline{R})=-\mu\ \underline{R}/|\underline{R}|^3 \tag{28}$$

$$\overline{\underline{g}}=\frac{1}{T}\int_0^T \underline{g}(\underline{R})\,dt \tag{29}$$

以上の簡略化の後最適推力角 $\underline{u}_{GC}$ を決める。まずphase 1. の最適解を $\tilde{\alpha}$, $\tilde{\beta}$（一定）として，(26)式を積分すると

$$\underline{V}(T)-\underline{V}(0)-\overline{\underline{g}}\cdot T=V_{ex}\,\tilde{\underline{u}}\int_0^T\frac{dt}{\tau-t} \tag{30}$$

ただし

V_{ex} ; 排気速度 (m/sec)

τ ; 質量／噴射率 (sec)

とする。(30)式を成分で書くと，

$$V_X(T)-V_X(0)-\overline{g}_X\cdot T=V_{ex}\cos\tilde{\alpha}\sin\tilde{\beta}\ \ln\frac{\tau}{\tau-T}$$

$$V_Y(T)-V_Y(0)-\overline{g}_Y\cdot T=V_{ex}\cos\tilde{\alpha}\cos\tilde{\beta}\ \ln\frac{\tau}{\tau-T}$$

$$V_Z(T)-V_Z(0)-\overline{g}_Z\cdot T=V_{ex}\sin\tilde{\alpha}\ \ln\frac{\tau}{\tau-T}$$

従って

$$\tan\tilde{\beta}=\frac{\Delta V_X}{\Delta V_Y} \tag{31}$$

$$\tan\tilde{\alpha}=\frac{\Delta V_Z}{(\Delta V_X^2+\Delta V_Y^2)^{\frac{1}{2}}} \tag{32}$$

により $\tilde{\alpha}$, $\tilde{\beta}$ が計算される。ここで

$$\Delta\underline{V}=\underline{V}(T)-\underline{V}(0)-\overline{\underline{g}}\cdot T$$

は現時刻から推力切断までに得るべき速度増分ベクトルで，

$$\Delta V=|\Delta\underline{V}|=(\Delta V_X^2+\Delta V_Y^2+\Delta V_Z^2)^{\frac{1}{2}}$$

は velocity-to-be-gained と呼ばれ，後に T を決定する際重要になる。つぎにphase 2. の最適解を

$$\alpha=\tilde{\alpha}+a+bt \tag{34}$$

$$\beta=\tilde{\beta}+c+dt \tag{35}$$

として，係数 a, b, c, d を飛翔体の現在位置および速度ベクトル，それに終端で課せられた拘束条件から決定する。すなわち(26)を2回積分して

$$\Delta\underline{V}=V_{ex}\int_0^T dt\ \frac{\underline{u}}{\tau-t} \tag{36}$$

$$\Delta\underline{R}=V_{ex}\int_0^T dT\int_0^T dt\ \frac{\underline{u}}{\tau-t} \tag{37}$$

を得る。ここで

$$\Delta\underline{R}=\underline{R}(T)-\underline{R}(0)-\underline{V}(0)\cdot T-\frac{1}{2}\overline{\underline{g}}\cdot T^2$$

と置いた。かくして

$$\Delta V_z=V_{ex}\int_0^T dt\ \frac{\sin\alpha}{\tau-t}$$

$$\Delta R_z=V_{ex}\int_0^T dT\int_0^T dt\ \frac{\sin\alpha}{\tau-t}$$

を連立的に解いて係数 a, b が定まり，

$$\Delta V_x=V_{ex}\int_0^T dt\ \frac{\cos\alpha\ \sin\beta}{\tau-t}$$

$$\Delta R_x=V_{ex}\int_0^T dT\int_0^T dt\ \frac{\cos\alpha\ \sin\beta}{\tau-t}$$

より係数 c, d が決定される。最終的に

$$a=-\tan\tilde{\alpha}+\left\{\left[T^2-T\left(\tau-\frac{1}{2}T\right)\ln\frac{\tau}{\tau-T}\right]\cos\tilde{\alpha}\right\}^{-1}\times\left\{\frac{\Delta V_z}{V_{ex}}\left[T\left(\tau-\frac{1}{2}T\right)-\tau(\tau-T)\ln\frac{\tau}{\tau-T}\right]-\frac{\Delta R_z}{V_{ex}}\left[-T+\tau\ \ln\frac{\tau}{\tau-T}\right]\right\} \tag{38}$$

$$c=-\tan\tilde{\beta}+a\tan\tilde{\beta}\tan\tilde{\alpha}+\left\{\left[T^2-T\left(\tau-\frac{1}{2}T\right)\ln\frac{\tau}{\tau-T}\right]\times\cos\tilde{\alpha}\cos\tilde{\beta}\right\}^{-1}\cdot\left\{-\frac{\Delta R_x}{V_{ex}}\left[-T+\tau\ \ln\frac{\tau}{\tau-T}\right]+\frac{\Delta V_x}{V_{ex}}\left[T\left(\tau-\frac{1}{2}T\right)-\tau(\tau-T)\ln\frac{\tau}{\tau-T}\right]\right\} \tag{39}$$

を得る。この際

$$\tilde{\alpha}\gg a+bt$$

$$\tilde{\beta}\gg c+dt$$

という近似を行ったが，これの妥当性はシミュレーションによっても確認されている。尚誘導計算は常に $t=0$ として行なわれるので，b, d の解析解を得ておく必要はないから省略する。残る問題は time-to-go の T の決定である。それはロケットの T 時間の間に得る特性速度増分

$$\int_0^T\frac{F}{m}dt=V_{ex}\ \ln\frac{\tau}{\tau-T} \tag{40}$$

が(33)式で表わされる velocity-to-be-gained ΔV に等しい，ということから決定できる。すなわち T の関数

$$f(T)=V_{ex}\ \ln\frac{\tau}{\tau-T}-\Delta V$$

の零点が所望の T となるが $f(T)$ は超越関数ゆえ，T の解析的表現は不可能である。そこで次の近似を行う。T を近似解として，真値を $T+\delta T$ と書くとき，近似的に

$$f(T)+\frac{df(T)}{dT}\delta T+\frac{1}{2}\frac{d^2f(T)}{dT^2}\delta T^2=0 \tag{41}$$

が成立する。したがって

$$\delta T=\frac{-B+\sqrt{B^2-4AC}}{2A} \tag{42}$$

ここで

$$2A=\frac{d^2f(T)}{dT^2}=\frac{V_{ex}}{(\tau-T)^2}+\frac{1}{\Delta V^3}\left\{(\Delta\underline{V},\ \underline{\bar{g}})^2-\Delta V^2\,|\underline{\bar{g}}|^2\right\}$$

$$B=\frac{df(T)}{dT}=\frac{V_{ex}}{\tau-T}+\frac{(\Delta\underline{V},\ \underline{\bar{g}})}{\Delta V}$$

$$C=f(T)$$

と置いた。guidance cycle で時々刻々 T を δT だけ修正してゆけば急速に最適な T に収束してゆく。

このように導出せられた誘導方程式は G 座標系で表現されるので，実効上推力方向制御ベクトル $\underline{u}_{GC}$ を R 座標系へ変換しなければならない。すなわち

$$\underline{u}_{RC}=C_{RN}\cdot C_{NG}\ \ \underline{u}_{GC} \tag{43}$$

ただし

$$\underline{u}_{RC}=\begin{bmatrix}\cos\theta_2\ \cos\theta_1\\ \cos\theta_2\ \sin\theta_1\\ \sin\theta_2\end{bmatrix}$$

(43)式を θ_1，θ_2 に関して解くと R 座標系での推力指令角が得られる。

推力制御系および姿勢制御系に関する詳細なソフトウェアは現時点では準備されていず，これらふたつの制御系は殆んど理想的に機能することが仮定されている。すなわち実効上は推力の大きさ，および推力軸のミスアライメントが一次のガウス・マルコフ性不規則雑音で fluctuate されると想定している。

3. 観測系誤差モデル

3.1 IMU誤差モデル

慣性機器として，1自由度ペンデュラム型加速度計（Single-Degree-of Freedom Pendulum type Accelerometer）および1自由度積分ジャイロ（Single-Degree-of-Freedom rate integrating Gyro）を使用するものと考える。これらの入出力関係を表わす数学的モデルは極めて複雑で一般には出力は入力のある非線型関数として無限個のパラメータを含むと考えられるが実効上は主要な誤差源として，次式を採用すれば充分であろう。即ち $j-th$ 入力軸（$j=x, y, z$）に対して，

加速度計

$$a_{mj}=a_j+\alpha_0\Delta a_j+\alpha_1\sum_{l=1}^{3}E_{jl}\ a_l+\alpha_2\,\underline{a}'K_{aj}\,\underline{a}+n_{aj} \tag{44}$$

但し　a_{mj}：$j-th$ 加速度計の出力　（m/sec²）

a_j：$j-th$ 加速度計の入力　（m/sec²）

Δa_j：$j-th$ 加速度計の定バイアス誤差（m/sec²）

E_{jl}：E_{jl}（$j=l$）はスケールファクタ誤差，E_{jl}（$j\neq l$）は，入力軸のミスアラインメント誤差（rad）

$$K_{aj}=\begin{pmatrix}K_{11}^{(j)} & K_{12}^{(j)}/2 & K_{13}^{(j)}/2\\ K_{12}^{(j)}/2 & K_{22}^{(j)} & K_{23}^{(j)}/2\\ K_{13}^{(j)}/2 & K_{23}^{(j)}/2 & K_{33}^{(j)}\end{pmatrix}$$

：$j-th$ 加速度計の入力自乗比例誤差係数（$\mu g/g^2$）

n_{aj}：量子化誤差および上記モデルで考慮されなかった誤差，例えば加速度の高次（3乗以上）非線型誤差などを総合したもので本プログラムでは平均値△，分散 σ_{aj}^2 の正規性白色雑音として擬似される。

ジャイロ

$$\omega_{mj}=\omega_j+\beta_0\Delta\omega_j+\beta_1\sum_{l=1}^{3}S_{jl}\ \omega_l+\beta_2\,\underline{\omega}'K_{\omega j}\,\underline{\omega}+\gamma_1(D_{I_j}\ D_{S_j}\ D_{O_j})\underline{f}_j+\gamma_2\,\underline{f}'D_{aj}\,\underline{f}_j+n_{\omega j} \tag{45}$$

但し

ω_{mj}：$j-th$ ジャイロの出力（rad/sec）

ω_j：$j-th$ ジャイロの入力（rad/sec）

$\Delta\omega_j$：$j-th$ ジャイロの定バイアス誤差（meru）

S_{jl}：$j-th$ ジャイロの入力比例誤差係数

$$K_{\omega j}=\begin{pmatrix}K_{\omega 11}^{(j)} & K_{\omega 12}^{(j)} & K_{\omega 13}^{(j)}\\ K_{\omega 21}^{(j)} & K_{\omega 22}^{(j)} & K_{\omega 23}^{(j)}\\ K_{\omega 31}^{(j)} & K_{\omega 32}^{(j)} & K_{\omega 33}^{(j)}\end{pmatrix}$$

：$j-th$ ジャイロの入力自乗比例誤差係数（対称行列）

$D_{I_j}\ D_{S_j}\ D_{O_j}$：加速度比例ドリストレート（mass unbalance drift rate）の誤差係数。添字 I_j，S_j，O_j は $j-th$ ジャイロの夫々入力軸，スピン軸，出力軸を表わす　（meru/g）

$$D_{a_j}=\begin{pmatrix}D_{II}^{(j)} & D_{IS}^{(j)} & D_{IO}^{(j)}\\ D_{SI}^{(j)} & D_{SS}^{(j)} & D_{SO}^{(j)}\\ D_{OI}^{(j)} & D_{OS}^{(j)} & D_{OO}^{(j)}\end{pmatrix}$$

；j-th ジャイロの加速度自乗比例誤差（aniso-elastic driftrate）の係数（対称行列）（meru／g^2）

n_{ω_j} ；n_{a_j} と同様に平均値 0，分散 $\sigma^2_{\omega_j}$ の正規性白色雑音として擬似される。

ジャイロを既報告の最適構成に配置すれば，D_{O_j}，$D^{(j)}_{OO}$は常に 0 としてよく，またその時 $\underline{f}_j$ は次式で定義されることになる。

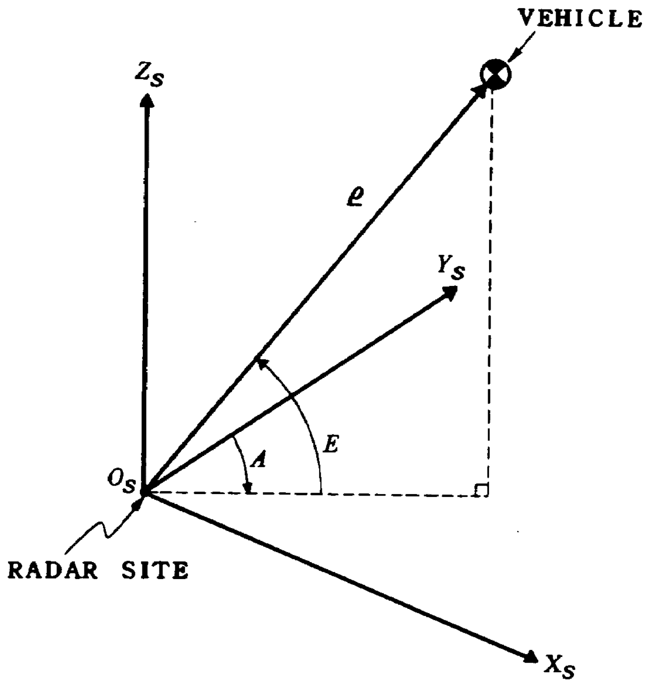

図 7　レーダートラッキング

$$\underline{f}_1=\begin{bmatrix} a_{x_B} \\ -a_{y_B} \\ a_{z_B} \end{bmatrix},\quad \underline{f}_2=\begin{bmatrix} a_{x_B} \\ a_{y_B} \\ a_{z_B} \end{bmatrix},\quad \underline{f}_3=\begin{bmatrix} a_{z_B} \\ -a_{x_B} \\ a_{y_B} \end{bmatrix}$$

なお(44)，(45)式で α_0，α_1，α_2，β_0，β_1，β_2，γ_1，γ_2 は適当な規格化パラメータである。

3.2　レーダートラッキング誤差モデル

レーダーによるトラッキングデータは電波誘導では勿論のこと，一般の誘導制御系の設計あるいは誘導誤差解析のために極めて重要な情報を提供する。通常ひとつのレーダーステーションでは図 7 に示すような 4 つの物理量，即ち ρ（slant range），$\dot{\rho}$（range-rate），A（azimuth）および E（elevation）に関する観測データが得られる。これらの物理量と飛翔体の状態ベクトルとの関係は次式で与えられる。

$$\left.\begin{aligned} \rho &= \|\underline{\rho}\| \\ \dot{\rho} &= \frac{(\dot{\underline{\rho}},\ \underline{\rho})}{\rho} \\ E &= \sin^{-1}\left[\frac{(\underline{\rho},\ \underline{Z}s)}{\rho}\right],\quad \delta \leq E \leq \frac{\pi}{2} \\ A &= \tan^{-1}\left[\frac{(\underline{\rho},\ \underline{X}s)}{(\underline{\rho},\ \underline{Y}s)}\right],\quad |A| \leq \frac{\pi}{2} \end{aligned}\right\} \tag{46}$$

但しレーダー観測は，飛翔体が図 8 に示すトラッキング可能領域（$E \geq \delta$，δ は観測できる最小の上下角）に在るときに限り可能である。

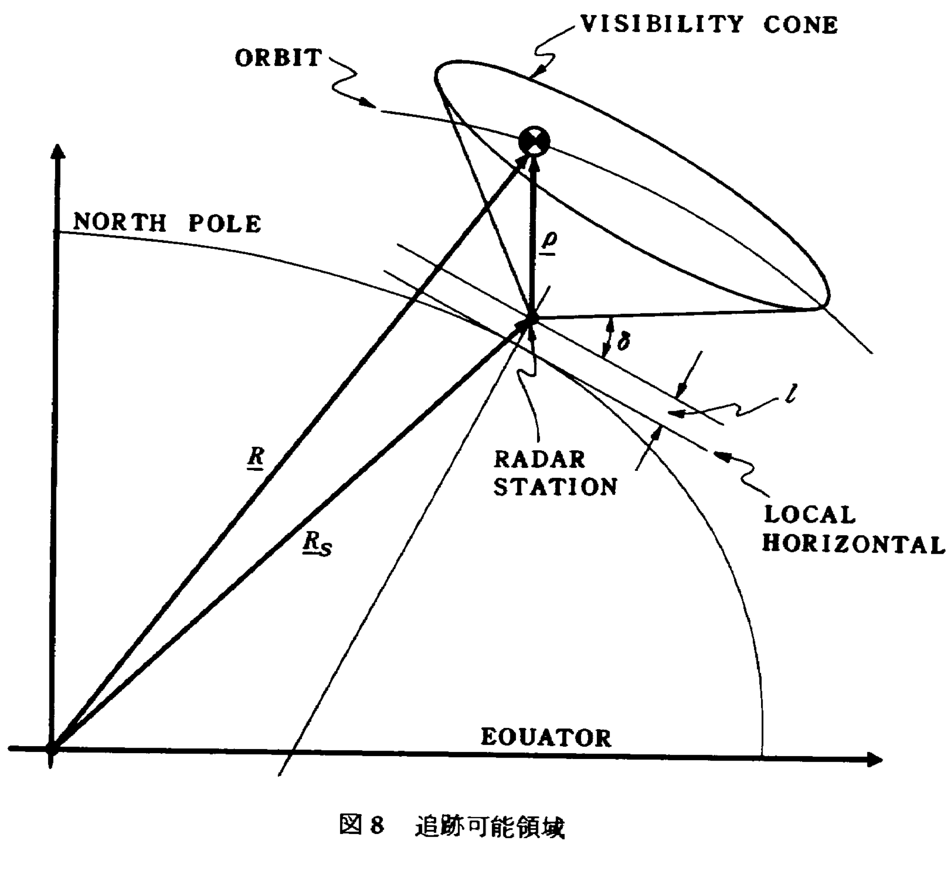

図 8　追跡可能領域

ここで

$$\underline{\rho}=\underline{R}_I-\underline{R}_{SI},\quad \dot{\underline{\rho}}=R_I-\dot{\underline{R}}_{SI}$$

かつ$\underline{R}_{SI}$はレーダーサイトのI座標系での位置ベクトル

$$\underline{R}_{SI} = C_{IE}\ \underline{R}_{SE}$$

である。また記号$\|\cdot\|$はユークリッドノルムを表わす。(46)式がいわゆる error-free の完全観測の方程式であるが実際には各種の系統的(systematic)誤差および偶然(random)誤差に汚された観測データしか入手できない。系統的誤差としては timing, refraction, dynamic lag, nonorthogonality, constant bias および scale factor error 等が考えられるが，本プログラムでは constant bias と，range 観測における scale factor のみを主誤差源とみなし，残りは偶然誤差に繰り込む。勿論レーダーサイトの位置誤差，即ちサイトの経度λ_S，緯度ϕ_Sおよび高度H_Sの誤差は重要な誤差源となる。かくして本プログラムに用意されているレーダートラッキングの最終的な観測方程式は次式で示される。

$$\left.\begin{aligned}\rho_{meas}&=\rho+\varepsilon\rho+\triangle\rho+n_\rho\\ \dot{\rho}_{meas}&=\dot{\rho}+\triangle\dot{\rho}+n_{\dot{\rho}}\\ E_{meas}&=E+\triangle E+n_E\\ A_{meas}&=A+\triangle A+n_A\end{aligned}\right\}\qquad(47)$$

ここで

ε ; scale factor
$\triangle\rho, \triangle\dot{\rho}, \triangle E, \triangle A$; constant bias
$n_{(\cdot)}$; random noise
$(\quad)_{meas}$; measured quantity

トラッキングにおける不規則雑音$\eta_{(\cdot)}$の power spectral density $S_{\eta(\cdot)}(\omega)$を

$$S_{\eta(\cdot)}(\omega)=\frac{2\beta_{(\cdot)}\sigma^2_{\eta(\cdot)}}{\beta^2_{(\cdot)}+\omega^2},\quad \beta_{(\cdot)}=\frac{1}{T_{(\cdot)}}\qquad(48)$$

と仮定しよう。但し$T_{(\cdot)}$は correlation time である。すなわち$\eta_{(\cdot)}$は次の1次の Gauss-Markov 過程に従う。

$$\frac{d}{dt}\eta_{(\cdot)}+\beta_{(\cdot)}\eta_{(\cdot)}=\sqrt{2\beta_{(\cdot)}}\ \zeta_{(\cdot)}\qquad(49)$$

ここで$\zeta_{(\cdot)}$は白色雑音でその power spectral density を

$$S_{\zeta(\cdot)}(\omega)=\sigma^2_{\eta(\cdot)}$$

とする。(49)式の離散形は

$$\eta_{(\cdot)}(t_{j+1})=e^{-\beta_{(\cdot)}\triangle t_j}\ \eta_{(\cdot)}(t_j)+n_{(\cdot)}(t_j)\qquad(50)$$

但し$\triangle t_j=t_{j+1}-t_j$，かつ$n_{(\cdot)}(t_j)$は平均0，分散$\sigma^2_{\eta(\cdot)}(1-e^{-2\beta_{(\cdot)}\triangle t_j})$の白色雑音である。1次の有色雑音の発生は(50)式に従っている。

4. KALMAN—BUCY推定理論とその応用

ある非線型な力学系および観測系の，基準軌道近傍に関する線型化方程式が次式に従うものとする。

$$d\underline{x}(t)=A(t)\,\underline{x}(t)\,dt+B(t)\,dt\,\underline{\mu}+\underline{\phi}(t)\,dt+G(t)\,d\underline{w}(t)\qquad(51)$$

$$\underline{y}(t_k)=C(t_k)\,\underline{x}(t_k)+D(t_k)\,\underline{\nu}+\underline{v}(t_k)\qquad(52)$$

ここで
$\underline{x}(t)$; n次元状態ベクトル
$A(t)$; $n\times n$ システム行列
$\underline{\mu}$; m_1次元パラメータベクトル
$B(t)$; $n\times m_1$行列
$\underline{\phi}(t)$; n次元 forcing ベクトル(既知)
$G(t)$; $n\times d$ noise intensity 行列
$\underline{w}(t)$; d次元 Brownian motion ベクトル過程で $\varepsilon\{\underline{w}(t)\,\underline{w}'(t)\}=Q|t-\tau|$, $\varepsilon\{\underline{w}(t)\}=0$
$\underline{y}(t_k)$; l次元観測値ベクトル
$C(t_k)$; $l\times n$ 観測感度行列
$D(t_k)$; $l\times m_2$行列
$\underline{\nu}$; m_2次元パラメータベクトル
$\underline{v}(t_k)$; l次元観測雑音ベクトル
t_k ; 観測時刻

とする。(51)式は伊藤型の確率微分方程式であるが，これに対する離散的表現は$n\times n$遷移行列

$$\frac{d}{dt}\Phi(t,s)=A(t)\ \Phi(t,s)\qquad(53)$$

$$\Phi(s,s)=I\quad(単位行列)$$

を用いて厳密に次式で与えられる。

$$\underline{x}(t)=\Phi(t,s)\,\underline{x}(s)+B(s)\,\underline{\mu}+\underline{\phi}(s)+\underline{\xi}(s)\qquad(54)$$

ただし

$$B(s)\triangleq\int_s^t d\alpha\,\Phi(t,\alpha)\,B(\alpha)$$

$$\underline{\phi}(s)\triangleq\int_s^t d\alpha\,\Phi(t,\alpha)\,\underline{\phi}(\alpha)$$

$$\underline{\xi}(s)\triangleq\int_s^t \Phi(t,\alpha)\,G(\alpha)\,d\underline{w}(\alpha)$$

と置いた。また明らかに$\varepsilon\{\underline{\xi}(s)\}=0$，かつ共分散行列は

$$\Xi(s)=\int_s^t d\alpha\ \Phi(t,\alpha)\ G(\alpha)\,Q\,G(\alpha)'\ \Phi'(t,\alpha)\qquad(55)$$

となる。既にTR-302でも概説したように観測雑音ベクトル$\underline{v}(t_k)$が理想的な白色雑音である場合には周知の KALMAN-BUCY フィルターがそのまま使える。その推定アルゴリズムを表1に示す。ただし簡単のためにB

$=D=0$とする。即ちパラメータベクトル$\underline{\mu}$，$\underline{\nu}$はもともと$\underline{x}(t)$の中に繰り込まれているものと考える。

表1 最適カルマンフィルターアルゴリズム

PREDICTION

$$\hat{\underline{x}}_{j+1/j}=\Phi_{j+1,j}\ \hat{\underline{x}}_{j/j}+\underline{\phi}_j$$

$$P_{j+1/j}=\Phi_{j+1,j}\ P_{j/j}\ \Phi^T_{j+1,j}+\Xi_j$$

FILTERING

$$\hat{\underline{x}}_{j+1/j+1}=\hat{\underline{x}}_{j+1/j}+K_{j+1}\ \tilde{\underline{y}}_{j+1/j}$$

$$P_{j+1/j+1}=P_{j+1/j}-K_{j+1}\ C_{j+1}\ P_{j+1/j}$$

$$\tilde{\underline{y}}_{j+1/j}=y_{j+1}-C_{j+1}\ \hat{\underline{x}}_{j+1/j}$$

$$K_{j+1}=P_{j+1/j}\ C^T_{j+1}\ (C_{j+1}\ P_{j+1/j}\ C^T_{j+1}+R_{j+1})^{-1}$$

where

$\varepsilon\{\underline{\nu}_{j+1}\}=0$，$\varepsilon\{\underline{\nu}_{j+1}\ \underline{\nu}^T_{j+1}\}=R_{j+1}$

$\hat{\underline{x}}_{k/j}$; conditional mean with respect to the set $\{\underline{y}(t_1),\cdots,\ \underline{y}(t_j)\}$

$P_{k/j}$; conditional covariance matrix

制御対象および観測系の，雑音の統計的性質まで含めた構造が既知の場合には，Table 1 に従って最適フィルターを設計できる。しかし殆んどすべての実在問題に対してこのような場合は期待できないし，一方数学的モデルを精密にすればするだけ未知パラメータの数が増加し，計算時間および所要記憶容量ともに増えることになる。また未知パラメータの中には推定の結果その不確定度が殆んど減少しないものがあって，これらのパラメータまで推定すべき状態ベクトルに繰り込むことは得策ではない。このような微小な影響しか持たないパラメータは無視するか，或いはあたかも偶然誤差のごとくにみなす考え方がありしばしば成功している。しかしこれは所謂“フィルターの発散（divergence）”を誘起する危険があるのでここではパラメータの a priori な統計量，換言すればパラメータの事前確率分布を積極的に考慮して設計された実用的フィルターのプログラムも用意されている。この実用的フィルターの導出とその応用例に関しては別稿で報告する予定なのでそのアルゴリズムのみを示しておこう。即ち Table 2. では，ある非線型な力学系および観測系の線型化方程式の離散的表現をそれぞれ

$$\underline{x}_{j+1}=A_j\ \underline{x}_j+B_j\ \underline{\mu}+\underline{\xi}_j \tag{56}$$

$$\underline{y}_{j+1}=C_{j+1}\ \underline{x}_{j+1}+D_{j+1}\ \underline{\nu}+\underline{\eta}_{j+1} \tag{57}$$

$$\underline{\eta}_{j+1}=L_j\ \underline{\eta}_j+\underline{\zeta}_j \tag{58}$$

とした場合の最小分散不偏推定を与える実用的フィルターが示されている。

表2 実用的カルマンフィルターアルゴリズム

PREDICTION

$$\hat{\underline{x}}_{j+1/j}=A_j\ \underline{x}_{j/j}+B_j\ \varepsilon\{\underline{\mu}\}$$

$$P_{j+1/j}=A_j\ P_{j/j}\ A^T_j+B_j\ \varepsilon\{\tilde{\underline{\mu}}\ \tilde{\underline{\mu}}^T\}\ B^T_j+Q_j+A_j\ M_{j/j}\ B^T_j+B_j\ M^T_{j/j}\ A^T_j$$

$$M_{j+1/j}=A_j\ M_{j/j}+B_j\ \varepsilon\{\tilde{\underline{\mu}}\ \tilde{\underline{\mu}}^T\}$$

$$N_{j+1/j}=A_j\ N_{j/j}+B_j\ \varepsilon\{\tilde{\underline{\mu}}\ \tilde{\underline{\nu}}^T\}$$

FILTERING

$$\hat{\underline{x}}_{j+1/j+1}=\hat{\underline{x}}_{j+1/j}+K_{j+1}\ \tilde{\underline{z}}_{j+1/j}$$

$$P_{j+1/j+1}=P_{j+1/j}-K_{j+1}\ X^T_{j+1}$$

$$M_{j+1/j+1}=(I-K_{j+1}\ C_{j+1})N_{j+1/j}+K_{j+1}\ C^{\#}_j\ M_{j/j}-K_{j+1}\ \tilde{D}_{j+1}\ \varepsilon\{\tilde{\underline{\nu}}\ \tilde{\underline{\mu}}^T\}$$

$$N_{j+1/j+1}=(I-K_{j+1}\ C_{j+1})N_{j+1/j}+K_{j+1}\ C^{\#}_j\ N_{j/j}-K_{j+1}\ \tilde{D}_{j+1}\ \varepsilon\{\tilde{\underline{\nu}}\ \tilde{\underline{\nu}}^T\}$$

wheve

$$K_{j+1}=X_{j+1}\ Y^{-1}_{j+1}$$

$$X_{j+1}=P_{j+1/j}\ C^T_{j+1}-(A_j\ P_{j/j}+B_j\ M^T_{j/j})C^{\#T}_j+N_{j+1/j}\ \tilde{D}^T_{j+1}$$

$$\begin{aligned}Y_{j+1}=&\ C_{j+1}\ P_{j+1/j}\ C^T_{j+1}+R_j-C_{j+1}(A_j\ P_{j/j}+B_j\ M^T_{j/j})\ C^{\#T}_j+C_{j+1}\ N_{j+1/j}\ \tilde{D}^T_{j+1}\\&+C^{\#}_j\ P_{j/j}\ C^{\#T}_j-C^{\#}_j(A_j\ P_{j/j}+B_j\ M^T_{j/j})\ C^T_{j+1}-C^{\#}_j\ N_{j/j}\ \tilde{D}^T_{j+1}+\tilde{D}_{j+1}\ N^T_{j+1/j}\ C^T_{j+1}\\&-\tilde{D}_{j+1}\ N^T_{j/j}\ C^{\#T}_j+\tilde{D}_{j+1}\ \varepsilon\{\tilde{\underline{\nu}}\ \tilde{\underline{\nu}}^T\}\ \tilde{D}^T_{j+1}\end{aligned}$$

観測雑音$\underline{\eta}_{j+1}$は有色雑音に仮定されているので，BRYSON，JOHANSEN および HENRIKSON による MEASUREMENT DIFFERENCE 法を適用して

$$\underline{z}_{j+1}=\underline{y}_{j+1}-L_j\ \underline{y}_j=C_{j+1}\ \underline{x}_{j+1}-C^{\#}_j\ \underline{x}_j+\tilde{D}_{j+1}\ \underline{\nu}+\underline{\zeta}_j \tag{59}$$

によって観測系(57)式を白色化する。その時観測残差ベクトルは

$$\tilde{\underline{z}}_{j+1/j}=\underline{z}_{j+1}-\hat{\underline{z}}_{j+1/j}=C_{j+1}\ \tilde{\underline{x}}_{j+1/j}-C^{\#}_j\ \tilde{\underline{x}}_{j/j}+\tilde{D}_{j+1}\ \tilde{\underline{\nu}}+\underline{\zeta}_j \tag{60}$$

となる。但し $\tilde{\underline{x}}_{j/k}=\underline{x}_j-\hat{\underline{x}}_{j/k}$, $C_j^{*}=C_{j+1}-\tilde{C}_{j+1}$
$\tilde{C}_{j+1}=C_{j+1}-L_j C_j$, $\tilde{D}_{j+1}=D_{j+1}-L_j D_j$, $\tilde{\underline{\mu}}=\underline{\mu}-\varepsilon\{\underline{\mu}\}$, $\tilde{\underline{\nu}}=\underline{\nu}-\varepsilon\{\underline{\nu}\}$ とし，さらに次の共分散行列を定義する。

$$\left.\begin{aligned}P_{j/k}&=\varepsilon\{\tilde{\underline{x}}_{j/k}\,\tilde{\underline{x}}'_{j/k}\}\\ M_{j/k}&=\varepsilon\{\tilde{\underline{x}}_{j/k}\,\tilde{\underline{\mu}}'\}\\ N_{j/k}&=\varepsilon\{\tilde{\underline{x}}_{j/k}\,\tilde{\underline{\nu}}'\}\end{aligned}\right\}$$

$\underline{x}$を“solve−for”parameter（推定すべきパラメータ），$\underline{\mu}$および$\underline{\nu}$を“consider”parameter（フィルターの設計の段階でそれらの事前確率分布は考慮されるが実際には推定されないパラメータ）と呼ぶことが多い。

規準軌道

規準軌道の近傍で線型化できることを仮定した。規準軌道としては通常次の 3 つが利用できる。即ち(1) GNT（Guided Noninal Trajectory），(2) GDT（Guidance Derived Trajectory），(3) GRT（Guidance Reconstructed Trajectory），である。(2)は機上計算機がIMU出力に基づき航法計算してつくる軌道で純慣性誘導方式ではこの軌道に従って実際の誘導がなされる。(3)はIMUのtelemetryデータに基づき地上の大型計算機によって高精度な航法計算をして得られる軌道である。(1)は通常の標準軌道で各種誘導パラメータの基準値を用いて軌道計算される。原理的にはいずれの軌道を使ってもよいが，飛翔後の誘導誤差解析の目的では(3)の GRT が最適であることを既に報告した。[1]

適用例

1 例として Kalman−Bucy フィルターによる誘導誤差解析の目的でstrapdown方式を対象にして構成された状態方程式を説明しよう。はじめにIMU誤差の系統的誤差ベクトルによる表式を得ておく。即ち加速度計に対して，

$$\triangle\underline{a}=(\triangle a_x\ \ \triangle a_y\ \ \triangle a_z)'$$
$$\underline{e}^{(j)}=(E_{j1}\ \ E_{j2}\ \ E_{j3})'$$
$$\underline{K}_a^{(j)}=(K_{a_{11}}^{(j)}\ K_{a_{12}}^{(j)}\ K_{a_{13}}^{(j)}\ K_{a_{22}}^{(j)}\ K_{a_{23}}^{(j)}\ K_{a_{33}}^{(j)})$$

但し $K_{a_{12}}^{(j)}\triangleq K_{a_{12}}^{(j)}+K_{a_{21}}^{(j)}$, $K_{a_{13}}^{(j)}\triangleq K_{a_{13}}^{(j)}+K_{a_{31}}^{(j)}$,
$K_{a_{23}}^{(j)}\triangleq K_{a_{23}}^{(j)}+K_{a_{32}}^{(j)}$

$$\underline{n}_a=(n_{a_x}\ n_{a_y}\ n_{a_z})'$$

さらに

$$\underline{e}_a=\begin{bmatrix}\underline{e}^{(1)}\\ \underline{e}^{(2)}\\ \underline{e}^{(3)}\end{bmatrix},\quad \underline{K}_a=\begin{bmatrix}\underline{K}_a^{(1)}\\ \underline{K}_a^{(2)}\\ \underline{K}_a^{(3)}\end{bmatrix}$$

と置く。そのとき，(44) 式は

$$\underline{a}_m=\underline{a}+\alpha_0\triangle\underline{a}+\alpha_1\begin{bmatrix}\underline{a}&&\\&\underline{a}&\\&&\underline{a}\end{bmatrix}\underline{e}+\begin{bmatrix}K_a&&\\&K_a&\\&&K_a\end{bmatrix}\underline{K}_a+\underline{n}_a \tag{62}$$

と書ける。ここで

$$K_a=(a_1^2\ \ a_1a_2\ \ a_1a_3\ \ a_2^2\ \ a_2a_3\ \ a_3^2)$$

とする。同様にジャイロに対しても，

$$\triangle\underline{\omega}=(\triangle\omega_x\ \ \triangle\omega_y\ \ \triangle\omega_z)'$$
$$\underline{s}^{(j)}=(S_{j1}\ \ S_{j2}\ \ S_{j3})'$$
$$\underline{K}_\omega^{(j)}=(K_{\omega11}^{(j)}\ K_{\omega12}^{(j)}\ K_{\omega13}^{(j)}\ K_{\omega22}^{(j)}\ K_{\omega23}^{(j)}\ K_{\omega33}^{(j)})'$$

但し $K_{\omega12}^{(j)}\triangleq K_{\omega12}^{(j)}+K_{\omega21}^{(j)}$, $K_{\omega13}^{(j)}\triangleq K_{\omega13}^{(j)}+K_{\omega31}^{(j)}$,
$K_{\omega23}^{(j)}\triangleq K_{\omega23}^{(j)}+K_{\omega32}^{(j)}$

$$\underline{n}_\omega=(n_{\omega_x}\ \ n_{\omega_y}\ \ n_{\omega_z})'$$
$$\underline{d}_a^{(j)}=(D_{aII}^{(j)}\ \ D_{aIS}^{(j)}\ \ D_{aIO}^{(j)}\ \ D_{aSS}^{(j)}\ \ D_{aSO}^{(j)})'$$

但し $D_{aIS}^{(j)}\triangleq D_{aIS}^{(j)}+D_{aSI}^{(j)}$, $D_{aIO}^{(j)}\triangleq D_{aIO}^{(j)}+D_{aOI}^{(j)}$,
$D_{aSO}^{(j)}=D_{aSO}^{(j)}+D_{aOS}^{(j)}$

$$\underline{D}^{(j)}=(D_{I_j}\ \ D_{S_j})'$$

さらに
$$\underline{K}_\omega=\begin{bmatrix}\underline{K}_\omega^{(1)}\\ \underline{K}_\omega^{(2)}\\ \underline{K}_\omega^{(3)}\end{bmatrix},\quad \underline{s}=\begin{bmatrix}\underline{s}^{(1)}\\ \underline{s}^{(2)}\\ \underline{s}^{(3)}\end{bmatrix},\quad \underline{d}_a=\begin{bmatrix}\underline{d}^{(1)}\\ \underline{d}^{(2)}\\ \underline{d}^{(3)}\end{bmatrix}$$

$$\underline{D}=\begin{bmatrix}\underline{D}^{(1)}\\ \underline{D}^{(2)}\\ \underline{D}^{(3)}\end{bmatrix}$$

と置く。そのとき (45) 式は

$$\underline{\omega}_m=\underline{\omega}+\beta_0\triangle\underline{\omega}+\beta_1\begin{bmatrix}\underline{\omega}'&&\\&\underline{\omega}'&\\&&\underline{\omega}'\end{bmatrix}\underline{s}+\beta_2\begin{bmatrix}K_\omega&&\\&K_\omega&\\&&K_\omega\end{bmatrix}\underline{K}_\omega+\gamma_1\begin{bmatrix}f^{(1)}&&\\&f^{(2)}&\\&&f^{(3)}\end{bmatrix}\underline{D}+\gamma_2\begin{bmatrix}D_a^{(1)}&&\\&D_a^{(2)}&\\&&D_a^{(3)}\end{bmatrix}\underline{d}_a+\underline{n}_\omega \tag{63}$$

ここで

$$K_\omega=(\omega_1^2\ \ \omega_1\omega_2\ \ \omega_1\omega_3\ \ \omega_2^2\ \ \omega_2\omega_3\ \ \omega_3^2)$$
$$f^{(j)}=(f_{I_j}\ \ f_{S_j})$$
$$D_a^{(j)}=(f_{I_j}^2\ \ f_{I_j}f_{S_j}\ \ f_{I_j}f_{O_j}\ \ f_{S_j}^2\ \ f_{S_j}f_{O_j})$$

とする。

以上の準備の後，先に述べたような適当な規準軌道およびパラメータの規準値からの deviation として，状態ベクトルを定義すると線型化方程式は次式で与えられることになる。

$$\frac{d}{dt}\,\delta\underline{x}=F(t)\,\delta\underline{x}+G(t)\,\underline{n} \tag{64}$$

ここで $\delta\underline{x}=\underline{x}-\overset{*}{\underline{x}}$，かつ

$$\underline{n}=\begin{bmatrix}\underline{n}_a\\ \underline{n}_\omega\end{bmatrix},\quad G(t)=\begin{bmatrix}\begin{array}{c:c}-C & \\ \hdashline & \\ \hdashline & -I \\ \hdashline & \end{array}\end{bmatrix}$$

$$F(t)=\begin{bmatrix}\begin{array}{c:c:c}F_{11} & F_{12} & \\ \hdashline & & \end{array}\end{bmatrix}$$

$$F_{11}=\begin{bmatrix}\begin{array}{c:c:c} & I & \\ \hdashline \dfrac{\partial\underline{g}}{\partial\underline{R}} & & -C(\underline{a}\times) \\ \hdashline & & -(\underline{w}\times)\end{array}\end{bmatrix} \tag{65}$$

$$F_{12}=\begin{bmatrix}\begin{array}{c:c:c:c} & & & \\ \hdashline \dfrac{\partial\underline{g}}{\partial\underline{J}} & -\alpha_0 C & -\alpha_1 C\begin{pmatrix}\underline{a}' & & \\ & \underline{a}' & \\ & & \underline{a}'\end{pmatrix} & -\alpha_2 C\begin{pmatrix}K_a & & \\ & K_a & \\ & & K_a\end{pmatrix} \\ \hdashline & & & \end{array}\end{bmatrix}$$

$$\begin{array}{c:c:c} & & \\ \hdashline & & \\ \hdashline -\beta_0 I & -\beta_1\begin{pmatrix}\underline{\omega}' & & \\ & \underline{\omega}' & \\ & & \underline{\omega}'\end{pmatrix} & -\beta_2\begin{pmatrix}K_\omega & & \\ & K_\omega & \\ & & K_\omega\end{pmatrix}\end{array}$$

$$\begin{array}{c:c} & \\ \hdashline & \\ \hdashline -\gamma_1\begin{pmatrix}\underline{f}^{(1)} & & \\ & \underline{f}^{(2)} & \\ & & \underline{f}^{(3)}\end{pmatrix} & -\gamma_2\begin{pmatrix}D_a^{(1)} & & \\ & D_a^{(2)} & \\ & & D_a^{(3)}\end{pmatrix}\end{array} \tag{66}$$

$$\delta\underline{x}=\begin{bmatrix}\delta\underline{R}\\ \delta\underline{V}\\ \delta\underline{\theta}\\ \delta\underline{J}\\ \delta(\triangle\underline{a})\\ \delta\underline{e}\\ \delta\underline{K}_a\\ \delta(\triangle\underline{\omega})\\ \delta\underline{S}\\ \delta\underline{K}_\omega\\ \delta\underline{D}\\ \delta\underline{d}_a\\ \delta\underline{T}\end{bmatrix}\begin{array}{l}\text{position}\\ \text{velocity}\\ \text{attitude}\\ \text{geopotential error}\\ \\ \text{accelerometer systematic error}\\ \\ \\ \text{gyro systematic error}\\ \\ \\ \text{radar system error}\\ \\ \end{array} \tag{67}$$

注意 1. (65)，(66) 式で各要素はすべて規準値をとる。

注意 2. F_{11}は (73)式の $A_{11}(t)$ に同等である。stable platform に対する F_{12} の表限はここでは省略する。

注意 3. $\varepsilon\{\underline{n}\}=0$，かつ

$$\varepsilon\{\underline{n}\,\underline{n}'\}=\begin{bmatrix}\sigma_{a_x}^2 & & & & & \\ & \sigma_{a_y}^2 & & & & \\ & & \sigma_{a_z}^2 & & & \\ & & & \sigma_{\omega_x}^2 & & \\ & & & & \sigma_{\omega_y}^2 & \\ & & & & & \sigma_{\omega_z}^2\end{bmatrix}$$

注意 4. かりに

$\underline{T}=(\lambda_s\ \phi_s\ H_s\ \varepsilon\ \triangle R\ \triangle E\ \triangle A)'$ (1ステーション)

$\underline{J}=(\mu\ \ J_2\ \ a_e)'$

とすると $\delta\underline{x}$ は概略 100 次元である。

5. 数値計算法に関する考察

Kalman－Bucy 理論による誘導制御系のシンセシスおよびアナリシスの計算には既に指摘されているように概略推定すべき状態ベクトル (solve－for parameter)の次元 n の 3 乗のオーダーで計算処理時間が増し，2 乗のオーダーで所要記憶容量が必要になってくる。そのため数値計算法とソフトウェアデザインには細心の注意を払わなければならない。後者については次節で考察することにして，本節では数値計算法に関する若干の工夫と注意について述べる。

(1)計算処理時間の主要部分は，線型化方程式 (51) 式をカルマンサイクル $[t_j,\ t_{j+1}]$ 間で離散化するプロセスに在る。初めに遷移行列 $\Phi(t,s)$ の計算法を考える。(53)式の解析解は

$$\Phi(t,s)=\exp\left(\int_s^t A(\tau)\,d\tau\right) \tag{69}$$

で与えられる。特に $A(t)=A$ (定行列) の場合厳密に

$$\Phi(t,s)=\exp[(t-s)A]=\sum_{n=0}^{\infty}(t-s)^n A^n/n! \tag{70}$$

即ち

$$\Phi(t,s)\sim\sum_{n=0}^{m-1}\frac{(t-s)^n A^n}{n!}+O(|t-s|^m) \tag{71}$$

が成り立つ。(71)式のように m 項まで考慮し，$(m+1)$ 項以降を切り捨てた場合の Φ の近似をここでは以後 m 次近似と呼ぶことにする。従って次のことが言える。一般に $A(t)$ が時変の場合，$|t-s|$ を充分小にして $A(t)$ を殆んど定数に見なし得るならば，(71)式による m 次近

似の誤差は概略$O(|t-s|^m)$で評価される。この事実から我々は，カルマンサイクル〔t_j, t_{j+1}〕の適当なsubdivision（部分々割）$t_{j+1}=t_{j(N+1)}>t_{j(N)}>\cdots>t_{j(1)}=t_j$に関して，(71)式の近似を採用することが出来る。

純慣性誘導による飛翔体の運動の線型化方程式に対する$A(t)$は次式で表現される。

$$A(t)=\begin{array}{c} \overleftrightarrow{\;n_s\;}\quad\overleftrightarrow{\;n_p\;} \\ \left[\begin{array}{c|c} A_{11}(t) & A_{12}(t) \\ \hline 0 & 0 \end{array}\right] \end{array}\begin{array}{c} \updownarrow n_s \\ \updownarrow n_p \end{array}$$

但し　$n_s+n_p=n$

で，状態ベクトル$\underline{x}(t)$の要素で，定パラメータの数をn_pとする。具体的には例えばstrapdown方式では$n_s=9$で

$$A_{11}(t)=\begin{bmatrix} 0 & I & 0 \\ \left(\dfrac{\partial \underline{g}}{\partial \underline{R}}\right)^* & 0 & -C^*[\underline{a}^*\times] \\ 0 & 0 & -[\underline{\omega}^*\times] \end{bmatrix} \tag{73}$$

となる。また stable platform 方式では$n_s=6$で

$$A_{11}(t)=\begin{bmatrix} 0 & I \\ \left(\dfrac{\partial \underline{g}}{\partial \underline{R}}\right)^* & 0 \end{bmatrix} \tag{74}$$

である。ここで$(\cdot)^*$は適当な規準軌道に関する値を示す。$A_{12}(t)$はパラメータの選択に依存して決まる。従って，この$A(t)$に応ずる遷移行列は，

$$\Phi_{11}(t,s)=A_{11}(t)\ \Phi_{11}(t,s),\quad \Phi_{11}(s,s)=I \tag{75}$$

を用いて

$$\Phi(t,s)=\begin{array}{c} \overleftrightarrow{\;n_s\;}\qquad\overleftrightarrow{\qquad n_p\qquad} \\ \left[\begin{array}{c|c} \Phi_{11}(t,s) & \int_s^t \Phi_{11}(t,\tau)A_{12}(\tau)\,d\tau \\ \hline 0 & I \end{array}\right] \end{array}\begin{array}{c} \updownarrow n_s \\ \updownarrow n_p \end{array} \tag{76}$$

で与えられる。(76)式の計算には，Φ_{11}のm次近似式

$$\Phi_{11}(t_{j(p+1)},t_{j(p)})\sim\sum_{n=0}^{m-1}\frac{A_{11}^n(t_{j(p)})}{n!}(t_{j(p+1)}-t_{j(p)})^n \tag{77}$$

を使う。実効上は$A_{11}(t)$の性質およびNの大きさに応じて$m=1$又は2を用いる。また$\Phi_{12}(t,s)$の計算は

$$\Phi_{12}(t,s)\sim\int_s^t \Phi_{11}(t,\tau)\,d\tau\cdot A_{12}(s) \tag{78}$$

による。

表3にΦの計算アルゴリズムをまとめておく。

(51)式の離散形は

$$\underline{x}(t_{j+1})=A_j\,\underline{x}_j+B_j\,\underline{u}+\underline{\phi}_j+\underline{\xi}_j \tag{79}$$

で与えられることを4節で見た(54)式)。ここで新たに

$A_j \triangleq \Phi(t_{j+1},t_j)$　と置いた。以下ではB_j, ϕ_jおよび$\underline{\xi}_j$の共分散行列Ξ_jの数値計算法について述べる。

いま区間〔t_j, t_{j+1}〕を$2m$等分して，それを

$$t_j=t_1<t_2<\cdots<t_{2m}<t_{2m+1}=t_{j+1}$$

とする。従って分割幅$\triangle t_j$は

$$2m\triangle t_j=t_{j+1}-t_j,\quad i.e.\ \triangle t_j=\frac{t_{j+1}-t_j}{2m}$$

となる。一般に，区間〔t_1, t_3〕,〔t_3, t_5〕, …,〔t_{2m-1}, t_{2m-1}〕にシンプソンの$\frac{1}{3}$則を使えば

$$\int_{t_1}^{t_{2m+1}} f(\alpha)d\alpha\approx\frac{1}{3}\triangle t_j\Big\{f(t_1)+f(t_{2m+1})+4\big[\underbrace{f(t_2)+\cdots+f(t_{2m})}_{m}\big]+2\big[\underbrace{f(t_3)+\cdots+f(t_{2m-1})}_{m-1}\big]\Big\} \tag{80}$$

となる。この近似誤差は$|f^{(4)}|\le M$として

$$\frac{(t_{j+1}-t_j)^5}{180\cdot(2m)^4}M \tag{81}$$

で与えられる。B_j, $\underline{\phi}_j$およびΞ_jの数値積分は(80)式により精度よく遂行できる。なおΞ_jの計算は煩雑を避けるため台形公式を用いて

$$\Xi_j\sim\frac{t_{j+1}-t_j}{2}\Big\{\Phi(t_{j+1},t_j)G(t_j)Q(t_j)G'(t_j)\Phi'(t_{j+1},t_j)+G(t_{j+1})Q(t_{j+1})G'(t_{j+1})\Big\} \tag{82}$$

によるのが得策である。

表3　遷移行列を計算するためのアルゴリズム

$$\Phi_{11}(t_{k+1},t_k)=I+A_{11}(t_k)\,\delta t_k+\frac{1}{2}A_{11}^2(t_k)\,\delta t_k^2$$

$$\Phi_{12}(t_{k+1},t_k)=\Big[I+\frac{1}{2}A_{11}(t_k)\,\delta t_k+\frac{1}{6}A_{11}^2(t_k)\,\delta t_k^2\Big]A_{12}(t_k)\,\delta t_k$$

$$\Phi_{12}(t_{k+1},t_k)=\Big[I+\frac{1}{2}A_{11}(t_k)\,\delta t_k\Big]A_{12}(t_k)\,\delta t_k$$

where　$\delta t_k=t_{k+1}-t_k$

〔2〕次に，表1に示した最適カルマンフィルターの計算時間の短縮化を図ろう。〔1〕でn_pは状態ベクトル$\underline{x}$の要素のうち，一定のパラメータ（即ちその時間微分は0）の数として定義された。さらに分割して

$$n_p = n_a + n_b$$

とする。ここでn_bは観測系の推定すべきパラメータ数とする。従って状態ベクトルを次のように分割して考える。

$$\underline{x} \triangleq \begin{array}{c} \left[\begin{array}{c} \underline{x} \\ \hline \underline{a} \\ \hline \underline{b} \end{array}\right] \end{array} \begin{array}{c} \updownarrow n_s \\ \updownarrow n_a \\ \updownarrow n_b \end{array} \tag{83}$$

このとき対応する遷移行列は明らかに次式で与えられる。

$$\begin{array}{cc} & \begin{array}{ccc} \leftarrow n_s \rightarrow & \leftarrow n_a \rightarrow & \leftarrow n_b \rightarrow \end{array} \\ \Phi(t,s) = & \left[\begin{array}{c|c|c} \Phi_{11}(t,s) & \Phi_{12}(t,s) & \\ \hline & I & \\ \hline & & I \end{array}\right] \begin{array}{c} \updownarrow n_s \\ \updownarrow n_a \\ \updownarrow n_b \end{array} \end{array} \tag{84}$$

以上のような分割を使って表1に示した最適カルマンフィルターを書き下してみよう(表4)。記述の便宜上ここでは update 前後の値を $(\cdot)^+$, $(\cdot)^-$ で表わす。また推定誤差 $\tilde{\underline{x}}_{j/k} = \underline{x}_j - \hat{\underline{x}}_{j/k}$ の共分散行列Pも次のように分割しておく。

$$P_{j/k} = \varepsilon\{\tilde{\underline{x}}_{j/k}\ \tilde{\underline{x}}_{j/k}'\}$$

$$= \begin{array}{c} \begin{array}{ccc} \leftarrow n_s \rightarrow & \leftarrow n_a \rightarrow & \leftarrow n_b \rightarrow \end{array} \\ \left[\begin{array}{c|c|c} P_{XX} & P_{XA} & P_{XB} \\ \hline P_{AX} & P_{AA} & P_{AB} \\ \hline P_{BX} & P_{BA} & P_{BB} \end{array}\right] \begin{array}{c} \updownarrow n_s \\ \updownarrow n_a \\ \updownarrow n_b \end{array} \end{array} \tag{85}$$

表4 最適カルマンフィルターの部分行列による表現

PREDICTION

$$\hat{\underline{x}}^{(+)} = \Phi_{11}\ \hat{\underline{x}}^{(-)} + \Phi_{12}\ \hat{\underline{a}}^{(-)}$$
$$\hat{\underline{a}}^{(+)} = \hat{\underline{a}}^{(-)}$$
$$\hat{\underline{b}}^{(+)} = \hat{\underline{b}}^{(-)}$$
$$P_{XX}^{(+)} = \Phi_{11}(P_{XX}^{(-)}\ \Phi_{11}^T + P_{XA}^{(-)}\ \Phi_{12}^T) + \Phi_{12}(P_{AX}^{(-)}\ \Phi_{11}^T + P_{AA}^{(-)}\ \Phi_{12}^T) + \Xi_X$$
$$P_{XA}^{(+)} = \Phi_{11}\ P_{XA}^{(-)} + \Phi_{12}\ P_{AA}^{(-)}$$
$$P_{XB}^{(+)} = \Phi_{11}\ P_{XB}^{(-)} + \Phi_{12}\ P_{AB}^{(-)}$$
$$P_{AA}^{(+)} = P_{AA}^{(-)}$$
$$P_{AB}^{(+)} = P_{AB}^{(-)}$$
$$P_{BB}^{(+)} = P_{BB}^{(-)}$$

FILTERING

$$\hat{\underline{x}}^{(+)} = \hat{\underline{x}}^{(-)} + K_X\ \underline{\nu}$$
$$\hat{\underline{a}}^{(+)} = \hat{\underline{a}}^{(-)} + K_A\ \underline{\nu}$$
$$\hat{\underline{b}}^{(+)} = \hat{\underline{b}}^{(-)} + K_B\ \underline{\nu}$$
$$P_{XX}^{(+)} = P_{XX}^{(-)} - K_X(H_X\ P_{XX}^{(-)} + H_B\ P_{XB}^{(-)T})$$
$$P_{XA}^{(+)} = P_{XA}^{(-)} - K_X(H_X\ P_{AX}^{(-)T} + H_B\ P_{AB}^{(-)T})$$
$$P_{XB}^{(+)} = P_{XB}^{(-)} - K_X(H_X\ P_{BX}^{(-)T} + H_B\ P_{BB}^{(-)})$$
$$P_{AA}^{(+)} = P_{AA}^{(-)} - K_A(H_X\ P_{AX}^{(-)T} + H_B\ P_{AB}^{(-)T})$$
$$P_{AB}^{(+)} = P_{AB}^{(-)} - K_A(H_X\ P_{BX}^{(-)T} + H_B\ P_{BB}^{(-)})$$
$$P_{BB}^{(+)} = P_{BB}^{(-)} - K_B(H_X\ P_{BX}^{(-)T} + H_B\ P_{BB}^{(-)})$$

where

$$K_X = (P_{XX}^{(-)}\ H_X^T + P_{XB}^{(-)}\ H_B^T)\ Y^{-1}$$
$$K_A = (P_{AX}^{(-)}\ H_X^T + P_{AB}^{(-)}\ H_B^T)\ Y^{-1}$$
$$K_B = (P_{BX}^{(-)}\ H_X^T + P_{BB}^{(-)}\ H_B^T)\ Y^{-1}$$
$$Y = R + H_X(P_{XX}^{(-)}\ H_X^T + P_{XB}^{(-)}\ H_B^T) + H_B(P_{BX}^{(-)}\ H_X^T + P_{BB}^{(-)}\ H_B^T)$$
$$H = (H_X\ \ 0\ \ H_B)$$
$$\underline{\nu} = \text{innovation processes}$$

明らかに prediction では $\hat{\underline{x}}$, P_{XX}, P_{XA}, P_{XB} のみを update すればよく残りは計算する必要がない。また filtering においても約2割記憶容量と計算時間が節約されることになる。表2に示した実用的フィルターにおいてもこのような partitioned form を使う。

〔3〕カルマンアルゴリズムは，表1に示すように電子計算機によるデータ処理に適した構造になっているが，不幸なことにこのアルゴリズムは数値的に不安定である。即ち共分散行列Pの positive semidefinite symmetric (PSDS)性が破れる場合に度々遭遇する。この欠点を防ぐために次の所謂安定化された(numerically stabilized)カルマンアルゴリズムが提案された(Joseph)。

$$P^{(+)} = (I - KH)\ P^{(-)}\ (I - KH)^T + KRK^T$$

従来のアルゴリズムに較べて，計算量が概略2倍になるという短所はあるものの suboptimal な情況下でカルマンフィルターを使う場合に適した有効なアルゴリズムである。

PSDS性を保証する他の手法として，共分散行列Pの Cholesky square root* S を update してゆく，所謂 square root filtering 法がある。ここでは説明を省略するが文献〔4〕,〔5〕に詳しい。

* n行n列のPSDS行列Pに対して

$$P = SS^T$$

を満たす行列Sは一意には定まらないが，特に

$$S = \begin{bmatrix} S_{11} & 0 \cdots\cdots & 0 \\ S_{21} & S_{22} & 0 \\ \cdots\cdots\cdots\cdots & & \vdots \\ & & 0 \\ S_{n1} & S_{n2} \cdots\cdots & S_{nn} \end{bmatrix}$$

なる下三角行列に制限すると一意的に決まる。このSをPのCholesky square root と呼び，次の手続きによって簡単に計算される。

$$S_{ii} = \left(P_{ii} - \sum_{j=1}^{i-1} S_{ij}^2\right)^{\frac{1}{2}} \qquad , \ 1 \leq i \leq n$$

$$S_{ji} = \begin{cases} 0 \ , & j < i \\ \dfrac{1}{S_{ii}}\left(P_{ji} - \displaystyle\sum_{k=1}^{i-1} S_{jk} S_{jk}\right), & i+1 \leq j < n \end{cases}$$

6. SOFTWARE DESIGNに関する考察

ここで報告するシミュレーションプログラムのように必要記憶容量および計算処理時間ともに極めて大規模なソフトウェアシステムを設計する場合にはSOFTWARE DESIGNにも充分な工夫がなされなければならない。特に限られたincoreの電子計算機（例えば HITAC 5020 Fのuser's available memoryは約55 K words ）ではareaのassignmentに最も苦労することは日常よく経験するところである。限られたusable memoryを効率よく使うために，われわれのシミュレーションプログラムでは全体を1次元ARRAYで統一している。即ちベクトルも行列も次の約束に従って同一視する。いまXをM行N列の2次元ARRAYとするとき，この行列Xの（i，j）要素を1次元ARRAY Yの（$i+M*(j-1)+4$）番目の要素に対応させる。この対応は one－to－one である。そして1次元ARRAY Yの最初の4要素は行列Xの大きさを特徴づけるパラメータを入れる。即ち

$$Y(1) = M$$
$$Y(2) = N$$
$$Y(3) = M * N$$
$$Y(4) = M * N + 4$$

として，Y(5)以降にXの要素が記憶させる。Y(3)はXの全要素数，Y(4)はYの実際の長さを表わす。また特にM次元ベクトルは，M行1列の行列と考える。その結果行列およびベクトルを含むすべての線型演算は一通りのサブルーチンを用意すれば実行できる。換言すれば行列用，ベクトル用と区別して作製する必要はない*。このような着想を基盤にすれば，サブルーチンの引数を減少させられるばかりか，ソフトウェアの設計上memory assignmentが容易，かつEQUIVALENCEによる結合が簡単というメリットもある。

* 例えば行列同志の積AB，および行列Aとベクトル$\underline{b}$の積は，付録AのL－1のサブルーチン"MULT"を使っていずれも

CALL MULT (C, A, B)

によって計算される。勿論MULTを呼ぶ前にA，Bは適切に定義されているものとする。同様に同じ大きさの行列A，Bの1次結合$pA+qB$（p，qはスカラー）および同じ次元のベクトル$\underline{a}$，$\underline{b}$の1次結合$p\underline{a}+q\underline{b}$は，付録AのL－2のサブルーチン"ADD"を使っていずれも

CALL ADD(C, P, A, Q, B)

によって計算される。

さらにまたMULT，ADDのごとく頻繁に使用するサブルーチンは，FORTRANによるよりもHISAPでコーディングする方がはるかに実行時間の短縮になる。実際80次元の正方行列に対して，1回当りの平均実行時間をHITAC 5020Fにより調べてみると次に示すようにFORTRANでコーディングされたサブルーチンがHISAPによるものに較べて，MULTで約4倍，ADDで約3倍かかるという著しい結果が得られた。われわれのシミュレーションプログラムにおいてもHISAPによるサブルーチンを採用した結果，計算時間を約1/2乃至1/3に短縮することができた。

サブルーチン名	HISAP	FORTRAN	割 合
MULT	14.8秒	58.6秒	3.95
ADD	0.2秒	0.6秒	3

付録Aにわれわれのプログラムに用意されている基本的なサブルーチン群を，付録Bに応用プログラムの1部を説明する。

7. あとがき

われわれは過去数年間にわたって，推定論を応用した誘導制御系の設計と評価に関する研究を行なってきた。その間にわれわれが研究開発したり，あるいは必要上やむをえなく作製したプログラムが膨大な量に達した。この報告ではその中から特に統計的な誘導誤差解析の手法とそのシミュレーションプログラムを重点的に解説した。ロケット軌道の高精度シミュレーションプログラムと併用することによって，殆んどすべてのミッションに対するinjection phaseまでの（pre－flightまたはpost－flightの）誘導誤差解析が可能になる。

終りに本研究を進めてゆく過程で極めて多くの方々か

ら有益なご教示を得た。とりわけ計算研究室磯部俊夫技官にはサブルーチンをHI SAPでコーディングしなおして頂いた他，プログラミング技術に関して得る処が多かった。これらの方に筆者らは深甚なる謝意を表する。

なお附録に示したプログラムは，HITAC 5020Fでテスト済みであることを附記しておく。

参考文献

1) 村田正秋：Kalman－Bucyフィルターによる誘導誤差の推定，航技研TR－302，1972年10月

2) 新田慶治ほか：宇宙飛翔体の直接式最適誘導法，航技研TR－161，1968年8月

3) 志甫徹：一般化Newton－Raphson法の計算機容量問題に対する改良最適化アルゴリズム，航技研 TR－249，1971年10月

4) G.J. Bierman; A comparison of Discrete Linear Filtering Algorithms, IEEE vol. AES-9, No. 1, January 1973.

5) P.G. Kaminski, A.E. Bryson, Jr., S.F. Schmidt; Discrete Square Root Filtering: A Survey of Current Techniques, IEEE vol. AC-16, No. 6, December 1971.

附録A BASIC SUBROUTINES

ここでは，SPACE GUIDANCE のための基本的なサブルーチンを説明し，併せて利用者のためにリストも呈示する。初めに付録Bにも共通する注意を述べる。

注1. 原則として2倍精度計算とする。

注2. 原則として引数の順序は出力値を初めに，入力値を続けてその次に書く。

注3. 物理定数，規格化常数などの不変データがそのサブルーチンに必要ならば，

DATA/ … / … /

の形式で組み込んでおく。

注4. 以下の説明では初めにサブルーチンのデック名を，続いて(1)計算目的・適用条件，(2)呼び方(CALLING SEQUENCE)，(3)引数(ARGUMENT LIST)，(4)使用サブルーチン，(5)プログラムリスト，が述べられている。

注5. デック名の前のL－，F－，E－はサブルーチンがそれぞれ線型計算，関数発生およびその他に属することを示す。

注6. 記号は本文に用いた記号に準ずる。

L－1 "MULT"

(1) 目　的

任意の行列A，Bの積

$$C = AB$$

を計算する。

(2) 呼び方

CALL MULT (C, A, B)

(3) 引　数

引数名	実行前	実行後	次元
C	(関係なし)	結果 C	$M \times L$
A	行列 A	(不変)	$M \times N$
B	行列 B	(不変)	$N \times L$

(4) 使用サブルーチン

なし

(5) リスト

HARP 5020 COMPILED LIST MUL

```
CMULT        HARP    64                                    MULT 005
      SUBROUTINE MULT(C,A,B)
      DIMENSION C(1),A(1),B(1)                             MULT 020
      C(1)=A(1)                                            MULT 030
      C(2)=B(2)                                            MULT 040
      II=A(1)+0.01                                         MULT 050
      JJ=B(2)+0.01                                         MULT 060
      KK=A(2)+0.01                                         MULT 070
      C(3)=II*JJ                                           MULT 080
      C(4)=C(3)+4.0                                        MULT 090
      DO 1 I=1,II                                          MULT 100
      DO 1 J=1,JJ                                          MULT 110
      IJ=I+II*(J-1)+4                                      MULT 120
      C(IJ)=0.0                                            MULT 130
      DO 2 K=1,KK                                          MULT 140
      IK=I+II*(K-1)+4                                      MULT 150
      KJ=K+KK*(J-1)+4                                      MULT 160
    2 C(IJ)=C(IJ)+A(IK)*B(KJ)                              MULT 170
    1 CONTINUE                                             MULT 180
      RETURN                                               MULT 190
      END                                                  MULT 200
```

L−2 “ADD”

(1) 目　的

任意の行列X, Yの和

$$Z = aX + bY$$

を計算する。ただしa, bはスカラー。

(2) 呼び方

CALL ADD (Z, A, X, B, Y)

(3) 引　数

引数名	実行前	実行後	次元
Z	(関係なし)	結果 Z′	M×N
A	スカラー a	(不変)	——
X	行列 X	(不変)	M×N
B	スカラー b	(不変)	——
Y	行列 Y	(不変)	M×N

備　考

CALL ADD(X, A, X, B, Y)という使い方もできる。

(4) 使用サブルーチン

なし

(5) リスト

HARP 5020 COMPILED LIST ADD

```
CADD        HARP   DECK                                   ADD 0010
      WORD LENGTH 64 BITS                                 ADD 0020
      BUBROUTINE ADD(Z,A,X,B,Y)                           ADD 0030
      DIMENSION Z(1),X(1),Y(1)                            ADD 0040
      DO 1 I=1,4                                          ADD 0050
    1 Z(I)=X(I)                                           ADD 0060
      II=Z(1)+0.01                                        ADD 0080
      JJ=Z(2)+0.01                                        ADD 0090
      DO 2 I=1,II                                         ADD 0100
      DO 2 J=1,JJ                                         ADD 0110
      IJ=I+II*(J-1)+4                                     ADD 0120
    2 Z(IJ)=A*X(IJ)+B*Y(IJ)                               ADD 0130
      RETURN                                              ADD 0140
      END                                                 ADD 0150
```

L−3 "MULTP"

(1) 目　的

任意の行列 A, B に対して

$$C = A B^T$$

を計算する。

(2) 呼び方

CALL MULTP (C, A, B)

(3) 引　数

引数名	実行前	実行後	次元
C	(関係なし)	結果 C	$M \times M$
A	行列 A	(不変)	$M \times N$
B	行列 B	(不変)	$M \times N$

(4) 使用サブルーチン

なし

(5) リスト

HARP 5020 COMPILED LIST MULTP

```
CMULTP        HARP   DECK                               MULTP 010
      WORD LENGTH 64 BITS                               MULTP 020
      SUBROUTINE MULTP(C,A,B)                           MULTP 030
      DIMENSION C(1),A(1),B(1)                          MULTP 040
      C(1)=A(1)                                         MULTP 050
      C(2)=B(1)                                         MULTP 060
      II=A(1)+0.01                                      MULTP 070
      JJ=A(2)+0.01                                      MULTP 080
      KK=B(1)+0.01                                      MULTP 090
      C(3)=II*KK                                        MULTP 100
      C(4)=C(3)+4.01                                    MULTP 110
      DO 1 I=1,II                                       MULTP 120
      DO 1 K=1,KK                                       MULTP 130
      IK=I+II*(K-1)+4                                   MULTP 140
      C(IK)=0.0                                         MULTP 150
      DO 2 L=1,JJ                                       MULTP 160
      IL=I+II*(L-1)+4                                   MULTP 170
      KL=K+KK*(L-1)+4                                   MULTP 180
    2 C(IK)=C(IK)+A(IL)*B(KL)                           MULTP 190
    1 CONTINUE                                          MULTP 200
      RETURN                                            MULTP 210
      END                                               MULTP 220
```

L-4 "TRANSP"

(1) 目 的

行列Aの転置行列

$$B = A^T$$

を計算する。

(2) 呼び方

CALL TRANSP (B, A)

(3) 引 数

引数名	実行前	実行後	次元
B	(関係なし)	結果 B'	$N\times M$
A	行列 A	(不変)	$M\times N$

(4) 使用サブルーチン

なし

(5) リスト

HARP 5020 COMPILED LIST TRANSP

```
CTRANSP     HARP    DECK                                          TRAN0010
      WORD LENGTH 64 BITS                                         TRAN0020
      SUBROUTINE TRANSP(B,A)                                      TRAN0030
      DIMENSTON B(1),A(1)                                         TRAN0040
      DO 1 I=1,4                                                  TRAN0050
    1 B(I)=A(I)                                                   TRAN0060
      B(1)=A(2)                                                   TRAN0070
      B(2)=A(1)                                                   TRAN0080
      II=B(1)+0.01                                                TRAN0090
      JJ=B(2)+0.01                                                TRAN0100
      DO 2 I=1,II                                                 TRAN0110
      DO 2 J=1,JJ                                                 TRAN0120
      IJ=I+II*(J-1)+4                                             TRAN0130
      JI=J+JJ*(I-1)+4                                             TRAN0140
    2 B(IJ)=A(JI)                                                 TRAN0150
      RETURN                                                      TRAN0160
      END                                                         TRAN0170
```

L-5 "MAKE"

(1) 目 的

行列を転送 (transfer) する

$$B = A$$

(2) 呼び方

CALL MAKE (B, A)

(3) 引 数

引数名	実行前	実行後	次元
B	(関係なし)	結果 B'	$M\times N$
A	行列 A	(不変)	$M\times N$

(4) 使用サブルーチン

なし

(5) リスト

HARP 5020 COMPILED LIST MAKE

```
CMAKE       HARP    DECK                                          MAKE0010
      WORD LENGTH 64 BITS                                         MAKE0020
      SUBROUTINE MAKE(B,A)                                        MAKE0030
      DIMENSION B(1),A(1)                                         MAKE0040
      DO 1 I=1,4                                                  MAKE0050
    1 B(I)=A(I)                                                   MAKE0060
      II=B(1)+0.01                                                MAKE0070
      JJ=B(2)+0.01                                                MAKE0080
      DO 2 I=1,II                                                 MAKE0090
      DO 2 J=1,JJ                                                 MAKE0100
      IJ=I+II*(J-1)+4                                             MAKE0110
    2 B(IJ)=A(IJ)                                                 MAKE0120
      RETURN                                                      MAKE0130
      END                                                         MAKE0140
```

L-6 " SET "

(1) 目　的

行列をクリア (clear) する

$A = 0$ (零行列)

(2) 呼び方

CALL SET (A)

(3) 引　数

引数名	実 行 前	実 行 後	次 元
A	(関係なし)	結 果 A	$M \times N$

(4) 使用サブルーチン

な　し

(5) リスト

HARP 5020 COMPILED LIST SET

```
CSET          HARP     DECK                                       SET 0010
      WORD LENGTH 64 EITS                                         SET 0020
      SUBROUTINE SET(A)                                           SET 0030
      DIMENSION A(1)                                              SET 0040
      II=A(1)+0.01                                                SET 0050
      JJ=A(2)+0.01                                                SET 0060
      DO 1 I=1,II                                                 SET 0070
      DO 1 J=1,JJ                                                 SET 0080
      IJ=I+II*(J-1)+4                                             SET 0090
    1 A(IJ)=0.0                                                   SET 0100
      RETURN                                                      SET 0110
      END                                                         SET 0120
```

L-7 " UNIT "

(1) 目　的

単位行列を発生する

(2) 呼び方

CALL UNIT (E)

(3) 引　数

引数名	実 行 前	実 行 後	次 元
E	(関係なし)	結 果 E'	$M \times M$

(4) 使用サブルーチン

な　し

(5) リスト

HARP 5020 COMPILED LIST UNIT

```
CUNIT         HARP     DECK                                       UNIT0010
      WORD LENGTH 64 BITS                                         UNIT0020
      SUBROUTINE UNIT(E)                                          UNIT0030
      DIMENSION E(1)                                              UNIT0040
      II=E(1)+0.01                                                UNIT0050
      DO 1 I=1,II                                                 UNIT0060
      DO 1 J=1,II                                                 UNIT0070
      IJ=I+II*(J-1)+4                                             UNIT0080
      E(IJ)=0.0                                                   UNIT0090
    1 IF(I.EQ.J) E(IJ)=1.0                                        UNIT0100
      RETURN                                                      UNIT0110
      END                                                         UNIT0120
```

L−8 "SSUM"

(1) 目 的

任意の行列A, Bに対して

$$C = AB^T + BA^T$$

を計算する。

(2) 呼び方

CALL SSUM(C, A, B)

(3) 引 数

引数名	実 行 前	実 行 後	次 元
C	(関係なし)	結 果 C'	$M \times M$
A	行 列 A	(不変)	$M \times N$
B	行 列 B	(不変)	$M \times N$

(4) 使用サブルーチン

な し

(5) リスト

HARP 5020 COMPILED LIST SSUM

```
CSSUM        HARP    64                               SSUM 110
      SUBROUTINE SSUM(C,A,B)                          SSUM 120
      DIMENSION C(1),A(1),B(1)                        SSUM 130
      II=A(1)+0.01                                    SSUM 140
      JJ=A(2)+0.01                                    SSUM 150
      C(1)=II                                         SSUM 160
      C(2)=II                                         SSUM 170
      C(3)=II**2                                      SSUM 180
      C(4)=C(3)+4.0                                   SSUM 190
      DO 1 I=1,II                                     SSUM 200
      DO 1 J=1,II                                     SSUM 210
      IJ=I+II*(J-1)+4                                 SSUM 220
      C(IJ)=0.0                                       SSUM 230
      DO 2 K=1,JJ                                     SSUM 240
      IK=I+II*(K-1)+4                                 SSUM 250
      JK=J+II*(K-1)+4                                 SSUM 260
    2 C(IJ)=C(IJ)+A(IK)*B(JK)+B(IK)*A(JK)             SSUM 270
    1 CONTINUE                                        SSUM 280
      RETURN                                          SSUM 290
      END                                             SSUM 295
```

L−9 "MATINV"

(1) 目 的

行列Aの逆行列，あるいは連立一次方程式

$$A\underline{x} = \underline{b}$$

の解を計算する。Aの次元は 50 以下。

(2) 呼び方

CALL MATINV(*A*, *B*, *M*, *D*)

(3) 引 数

引数名	実行前	実行後	次元
A	係数行列A	逆行列 A^{-1}	$M \times M$
B	ベクトル$\underline{b}$	解 $\underline{x}$	$M \times 1$
M	(関係なし)	(不変)	——
D	(関係なし)	行列式 $\|A\|$	——

備 考

$M = 0$ならば逆行列のみ計算

$M \neq 0$ならば逆行列および解を計算

(4) 使用サブルーチン

な し

(5) リスト

HARP 5020 COMPILED LIST MINV

```
CMINV        HARP    64                                          MINV 010
      WORD LENGTH 64 EITS
      SUBROUTINE MATINV(A,B,M,D)                                 MINV 020
      DIMENSTON A(1),B(1)                                        MINV 030
      DIMENSTON IPIVOT(50),INDEX(50,2),FIVOT(50)                 MINV 040
      EOUIVA ENCE (IROW,JROW),(IOOLUM,JOOLUM),                   MINV 050
        (AMAX,T,SWAP)
      D=1.0                                                      MINV 060
      N=A(1)+0.01                                                MINV 070
      DO 20 J=1,N                                                MINV 080
   20 IPIVOT(J)=0                                                MINV 090
      DO 555 I=1,N                                               MINV 100
      AMAX=0.0                                                   MINV 110
      DO 105 J=1,N                                               MINV 120
      IF(IPIVOT(J).EQ.1) GO TO 105                               MINV 130
      DO 100 K=1,N                                               MINV 140
      IF(IPIVOT(K)-1) 80,100,740                                 MINV 150
   80 JK=J+N*(K-1)+4                                             MINV 160
      IF(ABS(AMAX).GE.ABS(A(JK))) GO TO 100                      MINV 170
      IROW=J                                                     MINV 180
      ICOLUM=K                                                   MINV 190
      AMAX=A(JK)                                                 MINV 200
  100 CONTINUE                                                   MINV 210
  105 CONTINUE                                                   MINV 220
      IPIVOT(ICOLUM)=IPIVOT(ICOLUM)+1                            MINV 230
      IF(IROW.EQ.ICOLUM) GO TO 260                               MINV 240
      D=-D                                                       MINV 250
      DO 200 L=1,N                                               MINV 260
      IRL=IROW+N*(L-1)+4                                         MINV 270
      ICL=ICOLUM+N*(L-1)+4                                       MINV 280
      SWAP=A(IRL)                                                MINV 290
      A(IRL)=A(ICL)                                              MINV 300
  200 A(ICL)=SWAP                                                MINV 310
      IF(M.EQ.0) GO TO 260                                       MINV 320
      IR=IROW+4                                                  MINV 340
      IC=ICO UM+4                                                MINV 350
      SWAP=B(IR)                                                 MINV 360
      B(IR)=B(IC)                                                MINV 370
```

(続き) HARP 5020 COMPILED LIST MINV

```
      B(IC)=SWAP                                                MINV 380
  260 INDEX(I,1)=IROW                                           MINV 390
      INDEX(I,2)=ICOLUM                                         MINV 400
      IC=ICO UM+N*(ICOLUM-1)+4                                  MINV 410
      PIVOT(T)=A(IC)                                            MINV 420
      D=D*PIVOT(I)                                              MINV 430
      A(IC)=1.0                                                 MINV 440
      DO 350 L=1,N                                              MINV 450
      IL=ICO UM+N*(I-1)+4                                       MINV 460
  350 A(IL)=A(IL)/PIVOT(T)                                      MINV 470
      IF(M.EQ.0) GO TO 380                                      MINV 480
      IC=ICO UM+4                                               MINV 490
      B(IC)=B(IC)/PIVOT(I)                                      MINV 500
  380 DO 550 L1=1,N                                             MINV 510
      IF(L1.EQ.ICOLUM) GO TO 550                                MINV 520
      LI=L1+N*(ICOLUM-1)+4                                      MINV 530
      T=A(LI)                                                   MINV 540
      A(LI)=0.0                                                 MINV 550
      DO 450 L=1,N                                              MINV 560
      LL=L1+N*(L-1)+4                                           MINV 570
      IL=ICOLUM+N*(L-1)+4                                       MINV 580
  450 A(LL)=A(LL)-A(IL)*T                                       MINV 590
      IF(M.EQ.0) GO TO 550                                      MINV 600
      L=L1+4                                                    MINV 610
      IC=ICOLUM+4                                               MINV 620
      B(L)=B(L)-B(IC)*T                                         MINV 630
  550 CONTINUE                                                  MINV 640
  550 CONTINUE                                                  MINV 650
      DO 710 I=1,N                                              MINV 660
      L=N+1-I                                                   MINV 670
      IF(INDEX(L,1).EQ.INDEX(L,2)) GO TO 710                    MINV 680
      JROW=INDEX(L,1)                                           MINV 690
      JCOLUM=INDEX(L,2)                                         MINV 700
      DO 700 K=1,N                                              MINV 710
      KJ=K+N*(JROW-1)+4                                         MINV 720
      KJC=K+N*(JCOLUM-1)+4                                      MINV 730
      SWAP=A(KJ)                                                MINV 740
      A(KJ)=A(KJC)                                              MINV 750
  700 A(KJC)=SWAP                                               MINV 760
  710 CONTINUE                                                  MINV 770
  740 RETURN                                                    MINV 780
      END                                                       MINV 790
```

L−10 “TRACE”

(1) 目　的

正方行列Aのトレース (trace) を計算する

(2) 呼び方

CALL TRACE (S, A)

(3) 引　数

引数名	実 行 前	実 行 後	次 元
S	(関係なし)	結 果 S'	——
A	行 列 A	(不変)	$N \times N$

(4) 使用サブルーチン

な し

(5) リスト

HARP 5020　COMPILED LIST　TRACE

```
CTRACE      HARP    64                                    TRAC   10
      SUBROUTINE TRACE(S,A)                               TRAC   20
      DIMENSION A(1)                                      TRAC   30
      II=A(1)+0.01                                        TRAC   40
      S=0.0                                               TRAC   50
      DO 1  I=1,II                                        TRAC   60
      JJ=I+II*(I-1)+4                                     TRAC   70
    1 S=S+A(JJ)                                           TRAC   80
      RETURN                                              TRAC   90
      END                                                 TRAC  100
```

L−11 “MAKES”

(1) 目　的

スカラーaと行列Xの積

$$Y = aX$$

を計算する。

(2) 呼び方

CALL MAKES (Y, A, X)

(3) 引　数

引数名	実 行 前	実 行 後	次 元
Y	(関係なし)	結 果 Y'	$M \times N$
A	スカラー a	(不変)	——
X	行 列 X	(不変)	$M \times N$

備　考

CALL MAKES (X, A, X) という使い方もできる。

(4) 使用サブルーチン

な し

(5) リスト

HARP 5020　COMPILED LIST　MAKES

```
CMAKES      HARP    64                                    MAKES010
      WORD LENGTH 64 BITS                                 MAKES005
      SUBROUTINE MAKES(Y,A,X)                             MAKES020
      DIMENSION Y(1),X(1)                                 MAKES030
      DO 1  I=1,4                                         MAKES040
    1 Y(I)=X(I)                                           MAKES050
      II=X(1)+0.01                                        MAKES060
      JJ=X(2)+0.01                                        MAKES070
      DO 2  I=1,II                                        MAKES080
      DO 2  J=1,JJ                                        MAKES090
      IJ=I+II*(J-1)+4                                     MAKES100
    2 Y(IJ)=A*X(IJ)                                       MAKES110
      RETURN                                              MAKES120
      END                                                 MAKES130
```

L−12 “RMAT”

(1) 目 的

3次元ベクトル$\underline{w}$のつくる回転行列Rを発生する。

$$R=\begin{pmatrix} 0 & -w_3 & w_1 \\ w_3 & 0 & -w_2 \\ -w_1 & w_2 & 0 \end{pmatrix}$$

(2) 呼び方

CALL RMAT (*R*, *W*)

(3) 引 数

引数名	実 行 前	実 行 後	次 元
R	(関係なし)	結 果 R'	3×3
W	ベクトル $\underline{w}$	(不変)	3×1

(4) 使用サブルーチン

SET, LET

(5) リスト

```
      HARP 5020    COMPILED LIST    RMAT


CRMAT        HARP                                              RMAT 010
      WORD LENGTH 64 BITS                                      RMAT 020
      SUBROUTINE RMAT(R,W)                                     RMAT 030
      DIMENSION R(1),W(1)                                      RMAT 040
      CALL LET(R,3,3)                                          RMAT 050
      CALL SET(R)                                              RMAT 060
      R(6)=W(7)                                                RMAT 070
      R(7)=-W(6)                                               RMAT 080
      R(8)=-W(7)                                               RMAT 090
      R(10)=W(5)                                               RMAT 100
      R(11)=W(6)                                               RMAT 110
      R(12)=-W(5)                                              RMAT 120
      RETURN                                                   RMAT 130
      END                                                      RMAT 140
```

L−13 “INNERP”

(1) 目 的

*N*次元ベクトル $\underline{a}$, $\underline{b}$ の内積

$$c=(\underline{a},\ \underline{b})$$

を計算する。

(2) 呼び方

CALL INNERP (*C*, *A*, *B*)

(3) 引 数

引数名	実 行 前	実 行 後	次 元
C	(関係なし)	結 果 *c*	——
A	ベクトル $\underline{a}$	(不変)	*N*×1
B	ベクトル $\underline{b}$	(不変)	*N*×1

(4) 使用サブルーチン

```
      HARP 5020    COMPILED LIST    INNERF


CINNERP      HARP                                              INNE 010
      WORD LENGTH 64 EITS                                      INNE 020
      SUBROUTINE INNERP(C,A,B)                                 INNE 030
      DIMENSION A(1),B(1)                                      INNE 040
      C=0.0                                                    INNE 050
      DO 1 I=1,3                                               INNE 060
    1 C=C+A(I+4)*B(I+4)                                        INNE 070
      RETURN                                                   INNE 080
      END                                                      INNE 090
```

L-14 "OUTERP"

(1) 目 的

3次元ベクトル $\underline{a}$, $\underline{b}$ の外積

$$\underline{c}=\underline{a}\times\underline{b}$$

を計算する。

(2) 呼び方

CALL OUTERP (*C*, *A*, *B*)

(3) 引 数

引数名	実行前	実行後	次元
C	(関係なし)	結果 $\underline{c}$	3×1
A	ベクトル $\underline{a}$	(不変)	3×1
B	ベクトル $\underline{b}$	(不変)	3×1

(4) 使用サブルーチン

なし

(5) リスト

HARP 5020 COMPILED LIST OUTERP

```
COUTERP     HARP                                  OUTE 010
      WORD LENGTH 64 BITS                         OUTE 020
      SUBROUTINE OUTERP(C,A,B)                    OUTE 030
      DIMENSTION C(1),A(1),B(1)                   OUTE 040
      C(1)=3                                      OUTE 050
      C(2)=1                                      OUTE 060
      C(3)=3                                      OUTE 070
      C(4)=7                                      OUTE 080
      C(5)=A(6)*B(7)-E(6)*A(7)                    OUTE 090
      C(6)=A(7)*B(5)-A(5)*B(7)                    OUTE 100
      C(7)=A(5)*B(6)-E(5)*A(6)                    OUTE 110
      RETURN                                      OUTE 120
      END                                         OUTE 130
```

L-15 "VNORM"

(1) 目 的

*N*次元ベクトル $\underline{v}$ のユークリッドノルム $\|\underline{v}\|$ を計算する。

(2) 呼び方

CALL VNORM (*VN*, *V*)

(3) 引 数

引数名	実行前	実行後	次元
VN	(関係なし)	結果 $\|\underline{v}\|$	——
V	ベクトル $\underline{v}$	(不変)	*N*×1

(4) 使用サブルーチン

INNERP,

(5) リスト

HARP 5020 COMPILED LIST VNORM

```
CVNORM      HARP                                  VNOR 010
      WORD LENGTH 64 EITS                         VNOR 020
      SUBROUTINE VNORM(VN,V)                      VNOR 030
      DIMENSION V(1)                              VNOR 040
      CALL INNERP(C,V,V)                          VNOR 050
      VN=DSGRT(C)                                 VNOR 060
      RETURN                                      VNOR 070
      END                                         VNOR 080
```

L−16 “VUNIT”

(1) 目 的

N次元ベクトル $\underline{v}$ を規格化(normalize)する。

(2) 呼び方

CALL VUNIT(VU, V)

(3) 引 数

引数名	実行前	実行後	次元
VU	(関係なし)	結果 VU'	$N\times 1$
V	ベクトル $\underline{v}$	(不変)	$N\times 1$

(4) 使用サブルーチン

VNORM

(5) リスト

HARP 5020 COMPILED LIST VUNIT

```
CVUNIT      HARP                                        VUNI 010
       WORD LENGTH 64 BITS                              VUNI 020
       SUBROUTINE VUNIT(VU,V)                           VUNI 030
       DIMENSION VU(1),V(1)                             VUNI 040
       DO 1 I=1,4                                       VUNI 050
     1 VU(I)=V(I)                                       VUNI 060
       CALL VNORM(VN,V)                                 VUNI 070
       II=VU(3)+0.01                                    VUNI 080
       DO 2 I=1,II                                      VUNI 090
     2 VU(I+4)=V(I+4)/VN                                VUNI 100
       RETURN                                           VUNI 110
       END                                              VUNI 120
```

L−17 “MULT3”

(1) 目 的

行列A, Bに対して，積

$$C = ABA^T$$

を直接に計算する。

(2) 呼び方

CALL MULT3 (C, A, B)

(3) 引 数

引数名	実行前	実行後	次元
C	(関係なし)	結果 C'	$M\times M$
A	行列 A	(不変)	$M\times N$
B	行列 B	(不変)	$N\times N$

(4) 使用サブルーチン

なし

(5) リスト

HARP 5020 COMPILED LIST MULT3

```
CMULT3       HARP                                   MULT3010
      WORD LENGTH 64 BITS                           MULT3020
      SUBROUTINE MULT3(C,A,B)                       MULT3030
      DIMENSION C(1),A(1),B(1)                      MULT3040
      M=A(1)+0.01                                   MULT3050
      N=A(2)+0.01                                   MULT3060
      C(1)=M                                        MULT3070
      C(2)=M                                        MULT3080
      C(3)=M**2                                     MULT3090
      C(4)=C(3)+4.0                                 MULT3100
      DO 1 I=1,M                                    MULT3110
      DO 1 J=1,M                                    MULT3120
      IJ=I+M*(J-1)+4                                MULT3130
      C(IJ)=0.0                                     MULT3140
      DO 2 K=1,N                                    MULT3150
      DO 2 L=1,N                                    MULT3160
      KL=K+N*(L-1)+4                                MULT3170
      IK=I+M*(K-1)+4                                MULT3180
      JL=J+M*(L-1)+4                                MULT3190
    2 C(IJ)=C(IJ)+B(KL)*A(IK)*A(JL)                 MULT3200
    1 CONTINUE                                      MULT3210
      RETURN                                        MULT3220
      END                                           MULT3230
```

F−1 “TTI”

(1) 目 的

I 座標系から *T* 座標系への座標変換行列の発生

(2) 呼び方

CALL TTI (*CTI*, *FPA*, *R*, *V*)

(3) 引 数

引数名	実行前	実行後	次元
CTI	(関係なし)	変換行列	3×3
FPA	(関係なし)	γ_T (rad)	——
R	位置ベクトル $\underline{r}$	(不変)	3×1
V	速度ベクトル $\underline{v}$	(不変)	3×1

(4) 使用サブルーチン

VUNIT, OUTERP, VNORM, INNERP

(5) リスト

HARP 5020 COMPILED LIST TTI

```
CTTI        HARP
      WORD LENGTH 64 BITS
      SUBROUTINE TTI(CTI,FPA,R,V)
      DIMENSION ZT(7),YT(7),XT(7),CTI(1),R(1),V(1)
      CALL VUNIT(ZT,R)
      CALL OUTERP(YT,R,V)
      CALL VUNIT(YT,YT)
      CALL OUTERP(XT,YT,ZT)
      CTI(1)=3
      CTI(2)=3
      CTI(3)=9
      CTI(4)=13
      DO 1 J=1,3
      J1=3*J+2
      J2=J1+1
      J3=J2+1
      CTI(J1)=XT(J+4)
      CTI(J2)=YT(J+4)
    1 CTI(J3)=ZT(J+4)
      CALL VNORM(VN,V)
      CALL INNERP(A,V,XT)
      A=A/VN
      CALL INNERP(B,R,V)
      A=DSQRT(1,0-A**2)
      B=DSIGN(A,B)
      FPA=DATAN2(B,A)
      RETURN
      END
```

F-2 "TEI"

(1) 目　的

*I*座標系から*E*座標系への座標変換行列の発生

(2) 呼び方

CALL TEI (*CEI*, *TIME*)

(3) 引　数

引数名	実行前	実行後	次元
CEI	(関係なし)	変換行列	3×3
TIME	*t*	(不変)	——

(4) 使用サブルーチン

SET

(5) リスト

HARP 5020　COMPILED LIST　TEI

```
CTEI        HARP                                   TEI 010
      WORD LENGTH 64 BITS                          TEI 020
      SUBROUTINE TEI(CEI,TIME)                     TEI 030
      DIMENSION CEI(1)                             TEI 040
      DATA/OMEGA/7.292115E-5/                      TEI 045
      CEI(1)=3                                     TEI 050
      CEI(2)=3                                     TEI 060
      CEI(3)=9                                     TEI 070
      CEI(4)=13                                    TEI 080
      A=OMEGA*TIME                                 TEI 085
      CS=DCOS(A)                                   TEI 090
      SN=DSIN(A)                                   TEI 100
      CALL SET(CEI)                                TEI 110
      CEI(5)=CS                                    TEI 120
      CEI(6)=-SN                                   TEI 130
      CEI(8)=SN                                    TEI 140
      CEI(9)=CS                                    TEI 150
      CEI(13)=1.0                                  TEI 160
      RETURN                                       TEI 170
      END                                          TEI 180
```

F−3 “TSE”

(1) 目 的

*E*座標系から*S*座標系への座標変換行列の発生，およびレーダーサイトの位置ベクトル（*E*座標）の計算

(2) 呼び方

CALL TSE(*CSE, RE, HS, GLON, GLAT*)

(3) 引 数

引数名	実 行 前	実 行 後	次 元
CSE	（関係なし）	変換行列	3×3
RE	（関係なし）	位置ベクトル	3×1
HS	H_S	（不変）	——
GLON	λ_S (deg)	（不変）	——
GLAT	ϕ_S (deg)	（不変）	——

(4) 使用サブルーチン

な し

(5) リスト

HARP 5020 COMPILED LIST TSE

```
CTSE        HARP                                                 TSE 010
      WORD LENGTH 64 BITS                                        TSE 020
      SUBROUTINE TSE(CSE,RE,HS,GLON,GLAT)                        TSE 030
      DIMENSION CSE(1),R E(1)                                    TSE 040
      DATA/AE,EC,DR/0.0637816490E8,0.003352329,57.295779578/     TSE 050
      DOUBLE PRECISION LON,LAT                                   TSE 060
      LON=GLON/DR                                                TSE 070
      LAT=GLAT/DR                                                TSE 080
      A=DTAN(LAT)                                                TSE 090
      E=1.0-EC**2                                                TSE 100
      B=DSQRT(E**2+A**2)                                         TSE 110
      SLA=A/B                                                    TSE 120
      CLA=E/B                                                    TSE 130
      SLO=DSIN(LON)                                              TSE 140
      CLO=DCOS(LON)                                              TSE 150
      CSE(1)=3                                                   TSE 160
      CSE(2)=3                                                   TSE 170
      CSE(3)=9                                                   TSE 180
      CSE(4)=13                                                  TSE 190
      CSE(5)=-SLO                                                TSE 200
      CSE(6)=-SLA*CLO                                            TSE 210
      CSE(7)=CLA*CLO                                             TSE 220
      CSE(8)=CLO                                                 TSE 230
      CSE(9)=-SLA*SLO                                            TSE 240
      CSE(10)=CLA*SLO                                            TSE 250
      CSE(11)=0.0                                                TSE 260
      CSE(12)=CLA                                                TSE 270
      CSE(13)=SLA                                                TSE 280
      A=1.0-(EC*SLA)**2                                          TSE 290
      A=DSQRT(A)                                                 TSE 300
      G1=HS+AE/A                                                 TSE 310
      G2=HS+AE*E/A                                               TSE 320
      RE(1)=3                                                    TSE 330
      RE(2)=1                                                    TSE 340
      RE(3)=3                                                    TSE 350
      RE(4)=7                                                    TSE 360
      RE(5)=G1*CLA*CLO                                           TSE 370
      RE(6)=G1*CLA*SLO                                           TSE 380
      RE(7)=G2*SLA                                               TSE 390
      RETURN                                                     TSE 400
      END                                                        TSE 410
```

F-4 "TLI"

(1) 目　的

I 座標系から L 座標系への座標変換行列の発生

(2) 呼び方

CALL TLI (*CLI*, *GLL*)

(3) 引　数

引数名	実 行 前	実 行 後	次 元
CLI	(関係なし)	変換行列	3×3
GLL	ϕ_L' (deg)	(不変)	——

(4) 使用サブルーチン

SET

(5) リスト

```
        HARP 5020    COMPILED LIST     TLI

CTLI         HARP                                              TLI 010
       WORD LENGTH 64 BITS                                     TLI 020
       SUBROUTINE TLI(CLI,GLL)                                 TLI 030
       DIMENSION CLI(1)                                        TLI 040
       DATA/EC,DR/0.003352329,57.295779578/                    TLI 050
       E=1.0-EC**2                                             TLI 060
       G=GLL /DR                                               TLI 065
       A=DTAN(G)                                               TLI 070
       B=DSQRT(E**2+A**2)                                      TLI 080
       SLA=A/B                                                 TLI 090
       CLA=E/B                                                 TLI 100
       CLI(1)=3                                                TLI 110
       CLI(2)=3                                                TLI 120
       CLI(3)=9                                                TLI 130
       CLI(4)=13                                               TLI 140
       CALL SET(CLI)                                           TLI 150
       CLI(5)=CLA                                              TLI 160
       CLI(7)=-SLA                                             TLI 170
       CLI(9)=1.0                                              TLI 180
       CLI(11)=SLA                                             TLI 190
       CLI(13)=CLA                                             TLI 200
       RETURN                                                  TLI 210
       END                                                     TLI 220
```

F−5 “TNL”

(1) 目 的

L 座標系から *N* 座標系への座標変換行列の発生

(2) 呼び方

CALL TNL (*CNL*, *LA*)

(3) 引 数

引数名	実行前	実行後	次元
CNL	(関係なし)	変換行列	3×3
LA	Φ_L (deg)	(不変)	——

(4) 使用サブルーチン

SET

(5) リスト

HARP 5020 COMPILED LIST TNL

```
CTNL        HARP                                   TNL 010
       WORD LENGTH 64 EITS                         TNL 020
       SUBROUTINE TNL(CNL,LA)                      TNL 030
       DIMENSION CNL(1)                            TNL 040
       DOUBLE PRECISION LA                         TNL 050
       DATA/DR/57.295779578/                       TNL 060
       RA=LA/DR                                    TNL 070
       CNL(1)=3                                    TNL 090
       CNL(2)=3                                    TNL 100
       CNL(3)=9                                    TNL 110
       CNL(4)=13                                   TNL 120
       CALL SET(CNL)                               TNL 130
       CNL(5)=1.0                                  TNL 140
       CNL(9)=DSIN(RA)                             TNL 150
       CNL(10)=-DCOS(RA)                           TNL 160
       CNL(12)=-CNL(10)                            TNL 170
       CNL(13)=CNL(9)                              TNL 180
       RETURN                                      TNL 230
       END                                         TNL 240
```

F−6 “TNI”

(1) 目 的

I 座標系から *N* 座標系への座標変換行列の発生

(2) 呼び方

CALL TNI (*CNI*, *GLL*, *LA*)

(3) 引 数

引数名	実行前	実行後	次元
CNI	(関係なし)	変換行列	3×3
GLL	ϕ'_L (deg)	(不変)	——
LA	Φ_L (deg)	(不変)	——

(4) 使用サブルーチン

TLI, TNL, MULT

(5) リスト

HARP 5020 COMPILED LIST TNI

```
CTNI        HARP                                   TNI 010
       WORD LENGTH 64 BITS                         TNI 020
       SUBROUTINE TNI(CNI,GLL,LA)                  TNI 030
       DIMENSION CNI(1),CLI(13),CNL(13)            TNI 040
       DOUBLE PRECISION LA                         TNI 050
       CALL TLI(CLT,GLL)                           TNI 060
       CALL TNL(CNL,LA)                            TNI 070
       CALL MULT(CNI,CNL,CLI)                      TNI 080
       RETURN                                      TNI 090
       END                                         TNI 100
```

F−7 “TRN”

(1) 目　的

*R*座標系から*N*座標系への座標変換行列の発生，および状態ベクトルの変換

(2) 呼び方

CALL TRN (*CRN, RN, VN, RA, DRA, DCRA, R, V*)

(3) 引　数

引数名	実行前	実行後	次元
CRN	(関係なし)	変換行列	3×3
RN	(関係なし)	$\underline{r}_N$	3×1
VN	(関係なし)	$\underline{v}_N$	3×1
RA	ϕ (rad)	(不変)	——
DRA	$\dot{\phi}$ (rad／sec)	(不変)	——
CRA	ϕ (rad)	(不変)	——
DCRA	$\dot{\phi}$ (rad／sec)	(不変)	——
R	r	(不変)	——
V	$\dot{r}$	(不変)	——

(4) 使用サブルーチン

SET, TRANSP, RMAT, MULT, ADD

(5) リスト

HARP 5020　COMPILED LIST　TRN

```
CTRN        HARP                                                      TRN 010
      WORD LENGTH 64 BITS                                             TRN 020
      SUBROUTINE TRN(CRN,RN,VN,RA,DRA,CRA,DCRA,R,V)                   TRN 030
      DIMENSION CRN(1),RN(1),VN(1),RR(7),VR(7),CNR(13),A(13),W(7)     TRN 040
      SR=DSIN(RA)                                                     TRN 050
      CR=DCOS(RA)                                                     TRN 060
      SC=DSIN(CRA)                                                    TRN 070
      CC=DCOS(CRA)                                                    TRN 080
      CRN(1)=3                                                        TRN 090
      RR(1)=3                                                         TRN 005
      RR(2)=1                                                         TRN 015
      VR(1)=3                                                         TRN 025
      VR(2)=1                                                         TRN 035
      CALL SET(RR)                                                    TRN 045
      CALL SET(VR)                                                    TRN 055
      RR(7)=R                                                         TRN 065
      VR(7)=V                                                         TRN 075
      CRN(2)=3                                                        TRN 100
      CRN(3)=9                                                        TRN 110
      CRN(4)=13                                                       TRN 120
      CRN(5)=SC*CR                                                    TRN 130
      CRN(6)=-SR                                                      TRN 140
      CRN(7)=CC*CR                                                    TRN 150
      CRN(8)=SC*SR                                                    TRN 160
      CRN(9)=CR                                                       TRN 170
      CRN(10)=CC*SR                                                   TRN 180
      CRN(11)=-CC                                                     TRN 190
      CRN(12)=0.0                                                     TRN 200
      CRN(13)=SC                                                      TRN 210
      CALL TRANSP(CNR,CRN)                                            TRN 220
      W(1)=3                                                          TRN 230
      W(2)=1                                                          TRN 240
      W(3)=3                                                          TRN 250
      W(4)=7                                                          TRN 260
      W(5)=DCRA*SR                                                    TRN 270
      W(6)=DCRA*CR                                                    TRN 280
      W(7)=DRA                                                        TRN 290
      CALL RMAT(A,W)                                                  TRN 300
      CALL MULT(RN,CNR,RR)                                            TRN 310
      CALL MULT(VN,CNR,VR)                                            TRN 320
      CALL MULT(W,CNR,RR)                                             TRN 330
      CALL MULT(CNR,A,W)                                              TRN 340
      CALL ADD(VN,1.0,VN,1.0,CNR)                                     TRN 350
      RETURN                                                          TRN 360
      END                                                             TRN 370
```

F−8 " TGN "

(1) 目　的

N座標系からG座標系への座標変換行列の発生

(2) 呼び方

CALL TGN (*CGN, GRA, GCRA*)

(3) 引　数

引数名	実行前	実行後	次元
CGN	(関係なし)	変換行列	3×3
GRA	ϕ_G (deg)	(不変)	——
GCRA	φ_G (deg)	(不変)	——

(4) 使用サブルーチン

なし

(5) リスト

HARP 5020　COMPILED LIST　TGN

```
CTGN        HARP                                        TGN 010
      WORD LENGTH 64 BITS                               TGN 020
      SUBROUTINE TGN(CGN,GRA,GCRA)                      TGN 030
      DIMENSION CGN(1)                                  TGN 040
      DATA/DR/57.295779578/                             TGN 050
      PG=GCRA/DR                                        TGN 060
      FG=GRA/DR                                         TGN 070
      SP=DSIN(PG)                                       TGN 080
      CP=DCOS(PG)                                       TGN 090
      SF=DSIN(FG)                                       TGN 100
      CF=DCOS(FG)                                       TGN 110
      CGN(1)=3                                          TGN 120
      CGN(2)=3                                          TGN 130
      CGN(3)=9                                          TGN 140
      CGN(4)=13                                         TGN 150
      CGN(5)=SP*CF                                      TGN 160
      CGN(6)=-SF                                        TGN 170
      CGN(7)=CP*CF                                      TGN 180
      CGN(8)=SP*SF                                      TGN 190
      CGN(9)=CF                                         TGN 200
      CGN(10)=CP*SF                                     TGN 210
      CGN(11)=-CP                                       TGN 220
      CGN(12)=0.0                                       TGN 230
      CGN(13)=SP                                        TGN 240
      RETURN                                            TGN 250
      END                                               TGN 260
```

F−9 " RNORM "

(1) 目　的

任意の平均値μと分散σ^2を持つGauss型乱数を発生させるものである。発生方法としてはBox-Mullerの変換式を用いる。すなわち，〔0, 1〕の一様乱数x_1，x_2の2個を用いて

$$z_1 = (-2 \log x_1)^{1/2} \cos 2\pi x_2$$

$$z_2 = (-2 \log x_1)^{1/2} \sin 2\pi x_2$$

なる変換により2個の独立な正規乱数($\mu = 0$, $\sigma^2 = 1$)を発生させる。さらに指定した平均値μを分散σ^2を与えるために

$$y_i = \mu + \sigma z_i \quad (i = 1, 2)$$

の変換を行なう。

(2) 呼び方

CALL RNORM (*RNOR, RI1, RI2, ICY, RMEAN, RSIGM, N*)

(3) 引　数

引数名	実 行 前	実 行 後	次 元
RNOR		発生乱数の格納	*N*
*RI*1	一様乱数のた	次回のCALLの	——
*RI*2	めの初期値	ための初期値	——
ICY	出力サイクル	（不　変）	——
RMEAN	平 均 値	（不　変）	——
RSIGM	標準偏差	（不　変）	——
N	乱数の個数	（不　変）	——

備考1　引数*RI*1, *RI*2, *ICY*は，このサブルーチン内で一様乱数発生のサブルーチンRNUNIFを使用するために必要な引数である。

備考2　*RSIGM*は標準偏差 σ を入力データとするもので，分散 σ^2 と混同しないよう注意が必要である。

備考3　*RMEAN*＝0.0，*RSIGM*＝1.0と指定すれば*N*（0, 1^2）の分布の正規乱数が得られる。

備考4　必要なGauss型乱数の個数*N*は偶数で指定しなければならない。これは変換公式の性質上，一様乱数を偶数個必要とするためである。

(4) 使用サブルーチン

RNUNIF

(5) リスト

```
HARP 5020    COMPILED LIST     RNM


CRNM          HARP
       WORD LENGTH 64 BITS                                   RNM 010
       SUBROUTINE RNORM(RNOR,RI1,RI2,ICY,RMEAN,RSIGM,        RNM 020
         NUM)
C****  GAUSSIAN RANDOM NOISE OR NORMAL RANDOM NOISE          RNM 030
         GENERATOR
C      RI1,RI2 = INITIAL REAL NUMBER                         RNM 040
C      ON AND AFTER SECOND CALL, RI1,RI2 ARE REWRITTEN       RNM 050
         AUTOMATICALLY
C      ICY = OUTPUT CYCLE INDEX FOR ¥RUN                     RNM 060
C      RMEAN = MEAN VALUE                                    RNM 070
C      RSIGM = STANDARD DEVIATION                            RNM 080
C      NUM = NUMBER OF GAUSSIAN RANDOM NOISE (EVEN           RNM 090
         NUMBER)
C      RNOR(I) = STORAGE FOR GAUSSIAN RANDOM NOISE           RNM 100
C
       DIMENSION RNOR(NUM)                                   RNM 110
       IND=0                                                 RNM 120
       CALL RNUNIF(RI1,RI2,ICY,NUM,RNOR)                     RNM 130
       DO 100 I=1,NUM                                        RNM 140
       IF(IND.EQ.1) GO TO 200                                RNM 150
       R1=RNOR(I)                                            RNM 160
       R2=RNOR(I+1)                                          RNM 170
       X1=-2.0* DLOG(R1)                                     RNM 180
       X1=DSQRT(X1)                                          RNM 190
       X2=R2*6.283185307179586                               RNM 200
       Z1=X1*DCOS(X2)                                        RNM 210
       Z2=X1*DSIN(X2)                                        RNM 220
       Z1=Z1*RSIGM +RMEAN                                    RNM 230
       Z2=Z2*RSIGM +RMEAN                                    RNM 240
       RNOR(I)=Z1                                            RNM 250
       IND=1                                                 RNM 260
       GO TO 100                                             RNM 270
   200 RNOR(I)=Z2                                            RNM 280
       IND=0                                                 RNM 290
   100 CONTINUE                                              RNM 300
       RETURN                                                RNM 310
       END                                                   RNM 320
```

F−10 “RNUNIF”

(1) 目　的

区間〔0, 1〕の一様乱数（矩形乱数）を発生させることを目的とする。発生方法としてはFibonacci 数列

$$x_{n+2}=x_{n+1}+x_n$$

を利用する。プログラム上は x_{n+2} の値が1.0を越した場合は1.0を差引き，常に〔0, 1〕の範囲の値を発生するようにしてある。

(2) 呼び方

CALL RNUNIF（*RUNN*, *RI* 1, *RI* 2, *ICY*, *N*）

(3) 引　数

引数名	実行前	実行後	次元
RUNN		発生乱数の格納	*N*
RI 1	初期値	次回のCALLのための初期値	—
RI 2			—
ICY	出力のサイクル	（不変）	—
N	乱数の個数	（不変）	—

備考1　初期値*RI* 1, *RI* 2には，たとえば円周率 π とか自然対数の底 e などのような値を利用するとよい。2回目以降のCALLの際はサブルーチンの中で初期値が自動的に置換えられるから，初期値は第1回のCALLの時だけ与えればよい。

備考2　引数*ICY*は出力のサイクルを指定する引数で，*ICY*＝1のときはFibonacci 数列そのものである。*ICY*＝n,（$n\geq 2$）のときはFibonacci 数列を n 番目毎に，すなわち $n-1$ 個おきにサンプリングして取出して来ることを意味する。

備考3　*ICY*＝1の場合に，発生した乱数10,000個に対し，χ^2検定を利用して頻度検定，継次検定，ポーカー検定，およびギャップ検定の4種の検定を行なった結果，Fibonacci 数列によって発生させるものは有意水準5%，すなわち信頼係数95%で一様乱数とみなしうると判定された。

(4) 使用サブルーチン

なし

(5) リスト

HARP 5020　COMPILED LIST　RUM

```
CRUM          HARP
       WORD LENGTH 64 BITS
       SUBROUTINE RNUNIF(RUNN,RI1,RI2,ICY,NUM)
C****  UNIFORM RANDOM NOISE GENERATOR (0,1)
C      RI1,RI2 = INITIAL REAL NUMBER
C      ICY = OUTPUT CYCLE INDEX
C      NUM = NECESSARY NUMBERS OF UNIFORM RANDOM NOISE
C      RUNN = STORAGE FOR RANDOM NOISE
C      ADJUSTABLE DIMENSION
C      ON AND AFTER SECOND CALL,RI1,RI2 ARE REWRITTEN AUTOMATICALLY
C
       DIMENSION RUNN(NUM)
       XN1=RI1
       XN2=RI2
       DO 10 I=1,NUM
       J=1
   20  XN3=XN2+XN1
   25  IF(XN3.LE.1.0) GO TO 30
       XN3=XN3-1.0
       GO TO 25
   30  XN1=XN2
       XN2=XN3
       IF(J.EQ.ICY) GO TO 40
       J=J+1
       GO TO 20
   40  RUNN(I)=XN2
   10  CONTINUE
       RI1=XN1
       RI2=XN2
       RETURN
       END
```

F−11 “RCNG”

(1) 目　的

スペクトル密度が

$$\frac{2\beta\ \sigma^2}{\beta^2+\omega^2}\ ,\quad \beta=\frac{1}{T}$$

で与えられる有色雑音を発生する（3.2 参照）。

(2) 呼び方

CALL RCNG（*RCN, TC, SIGMA, SI, RI*1, *RI*2）

(3) 引　数

引数名	実　行　前	実　行　後	次元
RCN	旧有色雑音	新有色雑音	—
TC	*T*（秒）	（不変）	—
SIGMA	σ	（不変）	—
SI	サンプル間隔(秒)	（不変）	—
RI 1 *RI* 2	一様乱数のための初期値	次回のCALLののための初期値	—

(4) 使用サブルーチン

RNORM

(5) リスト

HARP 5020　COMPILED LIST　RONG

```
CRCNG      HARP   DECK                                                RONG05
           WORD LENGTH 64 BITS                                        RONG10
           SUBROUTINE RCNG(RCN,TO,SIGMA,SI)                           RONG15
C****  GENERATION OF RANDOM COLORED NOISE WITH GIVEN STATISTICS       RONG20
C      RCN   = COLORED NOISE (NEWLY REPLACED)                         RONG25
C      TC    = CORRELATION TIME                                       RONG30
C      SIGMA = STADARD DEVIATION                                      RONG35
C      SI    = SAMPLING INTERVAL                                      RONG40
           DIMENSION WN(2)                                            RONG45
           DATA/P,Q/3.141592653590,2.718281828459/                    RONG50
           ALPH=1.0/TO                                                RONG55
           A1=DEXP(-ALPH*SI)                                          RONG60
           A2=A1**2                                                   RONG65
           SA=SIGMA*DSQRT(1.0-A2)                                     RONG70
           CALL RNORM(P,Q,1,0.0,SA,2,WN)                              RONG75
           RCN=A1*RCN+WN(2)                                           RONG80
           RETURN                                                     RONG85
           END                                                        RONG90
```

E-1 " MATVEC "

(1) 目 的

2次元配列で与えられたM行N列の行列Aを1次元配列の行列Bに変換する(またはその逆)。

(2) 呼び方

CALL MATVEC (A, II, JJ, B, IND)

(3) 引 数

引数名	実行前	実行後	次元
A	行列 A	(不変)	$M\times N$
II	行数 M	(不変)	——
JJ	列数 N	(不変)	——
B	(関係なし)	結果 B'	$M\times N$
IND	(関係なし)	(不変)	——

備考

1° $IND<0$ (負整数)の場合に上記表に従う。

2° $IND\geq 0$ (非負整数)とすると,1次元配列の行列Bが2次元配列の行列Aに変換される。その場合もBは不変である。

(4) 使用サブルーチン

なし

(5) リスト

HARP 5020 COMPILED LIST MATVEC

```
CMATVEC       HARP     64                        MATV 010
      SUBROUTINE MATVEC(A,II,JJ,B,IND)           MATV 020
      DIMENSION A(1,1),B(1)                      MATV 030
      IF(IND) 1,2,2                              MATV 040
    1 II=B(1)+0.01                               MATV 050
      JJ=B(2)+0.01                               MATV 060
      DO 10 I=1,II                               MATV 070
      DO 10 J=1,JJ                               MATV 080
      IJ=I+II*(J-1)+4                            MATV 090
   10 A(I,J)=B(IJ)                               MATV 100
      GO TO 30                                   MATV 110
    2 B(1)=II                                    MATV 120
      B(2)=JJ                                    MATV 130
      B(3)=II*JJ                                 MATV 140
      B(4)=B(3)+4.0                              MATV 150
      DO 20 I=1,II                               MATV 160
      DO 20 J=1,JJ                               MATV 170
      IJ=I+II*(J-1)+4                            MATV 180
   20 B(IJ)=A(I,J)                               MATV 190
   30 RETURN                                     MATV 200
      END                                        MATV 210
```

E-2 "LET"

(1) 目 的

1次元配列の行列の構造を指定する

(2) 呼び方

CALL LET (A, M, N)

(3) 引 数

引数名	実行前	実行後	次元
A	(関係なし)	結果 A'	$M\times N$
M	行 数	(不変)	—
N	列 数	(不変)	—

備 考

この結果，$A(1)=M$，$A(2)=N$，$A(3)=M*N$，$A(4)=M*N+4$となり，$A(5)$以降は修正を受けない。

(4) 使用サブルーチン

なし

(5) リスト

HARP 5020 COMPILED LIST LET

```
CLET        HARP                                          LET 010
      WORD LENGTH 64 EITS                                 LET 020
      SUBROUTINE LET(A,M,N)                               LET 030
      DIMENSION A(1)                                      LET 040
      A(1)=M                                              LET 050
      A(2)=N                                              LET 060
      A(3)=M*N                                            LET 070
      A(4)=M*N+4                                          LET 080
      RETURN                                              LET 090
      END                                                 LET 100
```

E-3 “LOAD”

(1) 目 的

1次元配列の行列Aの(I, J)要素をBとする(あるいはその逆)。

(2) 呼び方

CALL LOAD(A, I, J, B, IND)

(3) 引 数

引数名	実行前	実行後	次元
A	行列 A	(不変)	$M\times N$
I	I 行	(不変)	—
J	J 列	(不変)	—
B	(関係なし)	結果 B'	—
IND	負整数	(不変)	—

備 考

$0\leq IND$(非負整数)ならば，スカラーBがAの(I, J)要素に記憶される。

(4) 使用サブルーチン

なし

(5) リスト

HARP 5020 COMPILED LIST LOAD

```
CLOAD        HARP    DECK                                        LOAD0010
       WORD LENGTH 64 BITS                                       LOAD0020
       SUBROUTINE LOAD(A,I,J,B,IND)                              LOAD0030
       DIMENSION A(1)                                            LOAD0040
       II=A(1)+0.01                                              LOAD0050
       IJ=I+II*(J-1)+4                                           LOAD0060
       IF(IND   ) 1,2,2                                          LOAD0070
     2 A(IJ)=B                                                   LOAD0080
       GO TO 3                                                   LOAD0090
     1 B=A(IJ)                                                   LOAD0100
     3 RETURN                                                    LOAD0110
       END                                                       LOAD0120
```

E-4 “MPRINT”

(1) 目 的

1次元配列のM行N列の行列Aをプリントする。

(2) 呼び方

CALL MPRINT (A)

(3) 引 数

引数名	実 行 前	実 行 後	次 元
A	行 列 A	(不変)	$M\times N$

(4) 使用サブルーチン

なし

(5) リスト

HARP 5020 COMPILED LIST MPRINT

```
CMPRINT     HARP                                          MPRI 010
      WORD LENGTH 64 BITS                                 MPRI 020
      SUBROUTINE MPRINT(A)                                MPRI 030
      DIMENSION A(1)                                      MPRI 040
      INTEGER KJ(6),FM(4)                                 MPRI 050
  100 FORMAT(1H1,10X,6(5X,I6,7X)/)                        MPRI 180
  200 FORMAT(4X,I6)                                       MPRI 210
      FM(1)=5H(1H+,                                       MPRI 260
      FM(3)=5HX,E15                                       MPRI 280
      FM(4)=5H.8)                                         MPRI 290
      II=A(1)+0.01                                        MPRI 060
      JJ=A(2)+0.01                                        MPRI 070
      IF(JJ.LT.6) GO TO 20                                MPRI 500
      C=JJ                                                MPRI 080
      C=C/6.0+0.01                                        MPRI 090
      L1=IDINT(C)                                         MPRI 100
      L2=JJ-6*L1                                          MPRI 110
      DO 1 L=1,L1                                         MPRI 120
      J1=6*L-5                                            MPRI 130
      J2=J1+5                                             MPRI 140
      DO 2 K=1,6                                          MPRI 150
    2 KJ(K)=J1+K-1                                        MPRI 160
      WRITE(6,100) (KJ(K),K=1,6)                          MPRI 170
      DO 3 I=1,II                                         MPRI 190
      WRITE(6,200) I                                      MPRI 200
      M=0                                                 MPRI 220
      DO 4 J=J1,J2                                        MPRI 230
      M=M+1                                               MPRI 240
      MM=13+18*(M-1)                                      MPRI 245
      IJ=I+II*(J-1)+4                                     MPRI 250
      FM(2)=IBTOD(MM)                                     MPRI 270
      WRITE(6,FM) A(IJ)                                   MPRI 300
    4 CONTINUE                                            MPRI 310
    3 CONTINUE                                            MPRI 320
    1 CONTINUE                                            MPRI 330
      IF(L2.EQ.0) GO TO 10                                MPRI 340
      J1=JJ-L2+1                                          MPRI 350
      DO 5 K=1,L2                                         MPRI 360
    5 KJ(K)=6*L1+K                                        MPRI 370
      WRITE(6,100) (KJ(K),K=1,L2)                         MPRI 380
      DO 6 I=1,II                                         MPRI 390
      WRITE(6,200) I                                      MPRI 400
      M=0                                                 MPRI 405
      DO 7 J=J1,JJ                                        MPRI 410
      M=M+1                                               MPRI 420
```

(続き)　HARP 5020　COMPILED LIST　MPRINT

```
      MM=13+18*(M-1)                                MPRI 430
      FM(2)=IBTOD(MM)                               MPRI 440
      IJ=I+II*(J-1)+4                               MPRI 450
    7 WRITE(6,FM) A(IJ)                             MPRI 460
    6 CONTIUE                                       MPRI 470
      GO TO 10                                      MPRI 510
   20 CONTINUE                                      MPRI 520
      DO 11 K=1,JJ                                  MPRI 530
   11 KJ(K)=K                                       MPRI 540
      WRITE(6,100) (KJ(K),K=1,JJ)                   MPRI 550
      DO 8 I=1,II                                   MPRI 560
      WRITE(6,200) I                                MPRI 570
      M=0                                           MPRI 580
      DO 9 J=1,JJ                                   MPRI 590
      M=M+1                                         MPRI 600
      MM=13+18*(M-1)                                MPRI 610
      FM(2)=IBTOD(MM)                               MPRI 620
      IJ=I+II*(J-1)+4                               MPRI 630
    9 WRITE(6,FM) A(IJ)                             MPRI 640
    8 CONTINUE                                      MPRI 650
   10 RETURN                                        MPRI 480
      END                                           MPRI 490
```

E-5 " RKG "

(1) 目　的

独立変数を x とすると n 次元ベクトル $\underline{y}$ に関する

$$\frac{d\underline{y}}{dx}=\underline{f}\,(\underline{y},\ x)$$

$$\underline{y}\,(x_0)=\underline{\eta}$$

というベクトル常微分方程式（すなわち連立 n 元常微分方程式）の初期値問題を解くことを目的とする。

ここでは丸め誤差の累積の自動修正という利点を持つ Runge-Kutta-Gill 法を用いる。すなわち,

$$\underline{y}_0=\underline{\eta},\qquad \underline{q}_{0,1}=0$$

$$h_n=x_{n+1}-x_n$$

$$\underline{k}_1=h_n\underline{f}\,(\underline{y}_n,\ x_n)$$

$$\underline{r}_1=\frac{1}{2}\underline{k}_1-\underline{q}_{n,1}$$

$$\underline{s}=\underline{y}_n$$

$$\underline{y}_{n,1}=\underline{y}_n+\underline{r}_1$$

$$\underline{r}_1'=\underline{y}_{n,1}-\underline{s}$$

$$\underline{q}_{n,2}=\underline{q}_{n,1}+3\underline{r}_1'-\frac{1}{2}\underline{k}_1$$

$$\underline{k}_2=h_n\underline{f}\,(\underline{y}_{n,1},\ x_n+\frac{h_n}{2})$$

$$\underline{r}_2=\left(1-\frac{1}{\sqrt{2}}\right)(\underline{k}_2-\underline{q}_{n,2})$$

$$\underline{s}=\underline{y}_{n,1}$$

$$\underline{y}_{n,2}=\underline{y}_{n,1}+\underline{r}_2$$

$$\underline{r}_2'=\underline{y}_{n,2}-\underline{s}$$

$$\underline{q}_{n,3}=\underline{q}_{n,2}+3\underline{r}_2'-\left(1-\frac{1}{\sqrt{2}}\right)\underline{k}_2$$

$$\underline{k}_3=h_n\underline{f}\left(\underline{y}_{n,2},\ x_n+\frac{h_n}{2}\right)$$

$$\underline{r}_3=\left(1+\frac{1}{\sqrt{2}}\right)(\underline{k}_3-\underline{q}_{n,3})$$

$$\underline{s}=\underline{y}_{n,2}$$

$$\underline{y}_{n,3}=\underline{y}_{n,2}+\underline{r}_3$$

$$\underline{r}_3'=\underline{y}_{n,3}-\underline{s}$$

$$\underline{q}_{n,4}=\underline{q}_{n,3}+3\underline{r}_3'-\left(1+\frac{1}{\sqrt{2}}\right)\underline{k}_3$$

$$\underline{k}_4=h_n\underline{f}\,(\underline{y}_{n,3},\ x_n+h_n)$$

$$\underline{r}_4=\frac{1}{6}\,(\underline{k}_4-2\underline{q}_{n,4})$$

$$\underline{s}=\underline{y}_{n,3}$$

$$\underline{y}_{n+1}=\underline{y}_{n,3}+\underline{r}_4$$

$$\underline{r}_4'=\underline{y}_{n+1}-\underline{s}$$

$$\underline{q}_{n+1\,1}=\underline{q}_{n,4}+3\underline{r}_4'-\frac{1}{2}\underline{k}_4$$

上式で $\underline{s}$ が関与している部分は一見冗長のようであるが，ここに丸め誤差の累積を避けるための仕掛けが含まれている。その理論的根拠については伊理，松谷の論文,* および Gillの原論文** を参照して頂きたい。

精しいことは上述の文献にゆずるとして，簡単に説明すれば，$\underline{r}$，$\underline{k}$，$\underline{q}$はx，$\underline{y}$に比べると小さく，一般にhの程度の数であり，$\underline{r}$を$\underline{y}$に加える時に$\underline{r}$の下の方の桁が丸められるので，そのとき生じた丸め誤差の分だけ値の変化を受けた$\underline{r}'$を用いて$\underline{q}$の計算を行なう必要があるということである。

* 伊理正夫，松谷泰行：Runge-Kutta-Gill 法について，情報処理，Vol.8，No.2，pp.103〜107，1967．

** S.Gill : A Process for the Step-by-Step Integration of Differential Equations in an Automatic Digital Computing Machine.，Proc. of the Cambridge Phil. Soc. Vol.47 (1951)，pp.96〜108．

備考1 上式群中で$\underline{s}$はベクトル表示をしているが，プログラム上では作業用の場所S一つでよい。

備考2 独立変数はxにしてあるが，これは時間tであってもさしつかえない。本報告では独立変数を時間tに取ることが多いので，プログラムではxの代りにtを用いている。

備考3 Runge-Kutta-Gill 法の公式は上述の通りであるが，プログラム上，問題に応じて変化する部分は微分方程式の右辺$\underline{f}$（$\underline{y}_n$，x_n）を計算する部分である。

(2) 呼び方

CALL RKG（YS, FS, N, HK, T）

(3) 引 数

引数名	実 行 前	実 行 後	次元
YS	変数（$t=t_n$）	変数($t=t_{n+1}$)	N
FS			N
N	変数の数	（不 変）	—
HK	積分刻み巾	（不 変）	—
T	独立変数(時間t_n)	t_{n+1}	—

備考4 このRKGサブルーチンの変数の数の制限は一応100としてあるが，プログラム中の作業用配列の次元を100以上に変更すれば変数の制限を拡げることができる。

(4) 使用サブルーチン

DAUX（YS, FS, T）

備考5 サブルーチンDAUXは微分方程式の右辺を計算するためのもので，与えられた方程式に応じて各自が作らなければならない。

DAUXの引数で，YS，Tはそれぞれ変数$\underline{y}$と独立変数x（このプログラムでは時間t）であり，FSは変数のそれぞれに対応する方程式の右辺の値を格納するもので，次元はYSと同じくNである。

(5) リスト

```
      HARP 5020    COMPILED LIST     RKG


CRKG          HARP
      WORD LENGTH 64 BITS
            SUBROUTINE RKG(YS,FS,NE,HK,T)
            DIMENSION YS(1),FS(1),Q(100)
      DATA/J/0/
      IF(J) 15,5,15
    5 DO 10 I=1,NE                                              RKG   020
   10 Q(I)=0.000                                                RKG   025
      B1=1.0D0                                                  RKG   030
      B2=2.0D0                                                  RKG   035
      B3=3.0D0                                                  RKG   040
      B4=B1/B2                                                  RKG   045
      B5=DSQRT(B4)                                              RKG   050
      B6=B1-B5                                                  RKG   055
      B7=B1+B5                                                  RKG   060
      B8=6.0
      J=1
   15       CALL DAUX (YS,FS,T)
      DO 30 I=1,NE                                              RKG   080
      AK=HK*FS(I)                                               RKG   085
      QI=Q(I)                                                   RKG   090
```

(続き) HARP 5020 COMPILED LIST RKG

```
      RY=B4*AK-QI                                              RKG  095
      SS=YS(I)                                                 RKGM   1
      YS(I)=SS+RY                                              RKGM   2
      RY=YS(I)-SS                                              RKGM   3
      Q(I)=QI+B3*RY-34*AK                                      RKG  100
   30 CONTINUE                                                 RKG  110
      T=T+B4*HK                                                RKG  115
          CALL DAUX (YS,FS,T)
      DO 40 I=1,NE                                             RKG  130
      AK=HK*FS(I)                                              RKG  135
      QI=Q(I)                                                  RKG  140
      RY=B6*(AK-QI)                                            RKG  145
      SS=YS(I)                                                 RKGM   1
      YS(I)=SS+RY                                              RKGM   2
      RY=YS(I)-SS                                              RKGM   3
      Q(I)=QI+B3*RY-36*AK                                      RKG  150
   40 CONTINUE                                                 RKG  160
          CALL DAUX (YS,FS,T)
      DO 50 I=1,NE                                             RKG  170
      AK=HK*FS(I)                                              RKG  175
      QI=Q(I)                                                  RKG  180
      RY=B7*(AK-QI)                                            RKG  185
      SS=YS(I)                                                 RKGM   1
      YS(I)=SS+RY                                              RKGM   2
      RY=YS(I)-SS                                              RKGM   3
      Q(I)=QI+B3*RY-B7*AK                                      RKG  190
   50 CONTINUE                                                 RKG  200
      T=T+B4*HK                                                RKG  205
          CALL DAUX (YS,FS,T)
      DO 60 I=1,NE                                             RKG  220
      AK=HK*FS(I)                                              RKG  225
      QI=Q(I)                                                  RKG  230
      RY=(AK-B2*QI)/B8                                         RKG  235
      SS=YS(I)                                                 RKGM   1
      YS(I)=SS+RY                                              RKGM   2
      RY=YS(I)-SS                                              RKGM   3
      Q(I)=QI+B3*RY-B4*AK                                      RKG  240
   60 CONTINUE                                                 RKG  250
      IF ACCUMULATOR OVERFLOW 100,200                          RKGN
  100 WRITE(6,900)                                             RKGN
  900 FORMAT(1H0,30H( **** ¥RKG OVERFLOW ***** )   )           RKGN
      STOP                                                     RKGN
  200 RETURN                                                   RKGN
      END
```

附録B　応用プログラム

ここでは，誘導制御系の設計と評価のための標準的な応用プログラムを解説する。ここで述べるシミュレーションプログラムは大別してふたつの部分から構成されており，その概要は次のようである。

DATA GENERATOR

GNT，GDTおよびGRTの計算，不規則雑音に汚されたIMUおよび電波追跡データのシミュレーション。

RECONSTRUCTOR

機能的にさらに次の3部分に分割する。

(1) INPUT PROCESSOR

DATA GENERATOR によってシミュレートされたIMUデータ，トラッキングデータおよびGRT等は通常数本の磁テープに格納されている。これら各種データに必要なPRE-CROCESSINGを施し，次段階のLINEAR ESTIMATORへの入力のデータ構造として適切なる再編集を行う。

(2) LINEAR ESTIMATOR

カルマンフィルターに基づき，軌道再編成および各種誘導パラメータの推定を行う。

(3) OUTPUT PROCESSOR

主として推定結果および観測残差の図式化を行う。

ここでは紙数の都合で軌道，IMUデータおよびトラッキングデータのシミュレーションプログラム，そしてカルマンフィルターによって軌道および誘導パラメータを推定するプログラムを紹介することにし，GRTおよび遷移行列を計算するプログラムは省略している。本論にその概要を説明しているので使用者は容易にプログラム化できるが必要ならば著者らに照会されたい。

1. 軌道，IMUデータおよびトラッキングデータの発生プログラム

メインプログラム

NE　：状態ベクトルの次元(19)
BH　：刻み巾(sec)
TF　：積分時間(sec)
C1　：発射点高度(*m*)
C2　：発射点経度(deg)
C3　：発射点緯度(deg)
C4　：方位角(deg)
C5　：上下角(deg)
RI　：位置ベクトル(*I*座標)
VI　：速度ベクトル(*I*座標)
CIB：方向余弦行列(*I*座標)
X(1)　：高度(m)
X(2)　：速度(m/sec)
X(3)　：径路角(rad)
X(4)　：レンジ角(rad)
X(5)〜X(13)　：*CIB*
X(14)〜X(16)：*RI*
X(17)〜X(19)：*VI*
TIME：飛翔時間(sec)
R1：レンジ観測におけるランダムノイズの1σ値
R2：レンジ観測におけるスケールファクタ
R3：レンジ観測におけるバイアス(m)
E1：上下角観測におけるランダムノイズの1σ値(rad)
E2：上下角観測におけるスケールファクタ
E3：上下角観測におけるバイアス(rad)
A1：方位角観測におけるランダムノイズの1σ値(rad)
A2：方位角観測におけるスケールファクタ
A3：方位角観測におけるバイアス(rad)
H1：ステーション高度(m)
H2：ステーション経度(deg)
H3：ステーション緯度(deg)
T1：レンジ観測におけるランダムノイズのコリレーションタイム(sec)
T2：上下角観測におけるランダムノイズのコリレーションタイム(sec)
T3：方位角観測におけるランダムノイズのコリレーションタイム(sec)
SI：サンプリング間隔(sec)
INDEX＝　1：ノイズフリー
　　　　2：白色雑音
　　　　3：有色雑音(1次マルコフ過程)
CTE：座標変換行列
RAM：レンジ観測値(m)
ELM：上下角観測値(rad)
AZM：方位角観測値(rad)
A　：推力加速度ベクトル(m/sec^2)
AM　：加速度計出力ベクトル(m/sec^2)
W　：回転角速度ベクトル(rad/sec)
WM　：ジャイロ出力ベクトル(rad/sec)

備考1．このプログラムはSS−3ヴィークルをモデルにして，その推力飛翔軌道とランダムノイズに乱されたIMUおよびレーダートラッキングデータを発生するものである。本プログラムでひとつのステーションにおけるトラッキングデータをシミュレートできるが，サブルーチン“TTE”および“NRTD”を繰り返し使用することによって任意に与えられたステーションをシミュレートできる。

備考2．サブルーチン“IMU”において慣性機器の各種ハードウェアの誤差パラメータは適当に規格化した後DATA/…/……/ という形式で与えている。パラメータの規準値は APOLLO IMUに拠っている。

備考3．本プログラムで発生した軌道，IMUデータおよびトラッキングデータは磁気テープに格納した後GRT，遷移行列等のオフライン計算を経て軌道を再構成し，誘導誤差を分離するために後述するプログラム“RECON-STRUCTOR”にかけられる。

備考4．メインプログラムのRUNには次のサブプログラムが必要である。

RADAR ： ρ，$\dot{\rho}$，E，Aの真値の計算
NRTD ： レーダー観測データの発生
IMU ： IMU出力の計算
THRUST： 推力制御系の模擬
DAUX ： RKGにおける$\underline{f}$の計算
INITIA： 初期条件の計算
GRAV ： 重力加速度の計算

```
        HARP 5020   COMPILED LIST

      WORD LENGTH 64 BITS                                        TRAJ 040
C                                                                TRAJ 045
C      SPACE TRAJECTORY, NOISY IMU AND RADAR TRACKING            TRAJ 050
         DATA SIMULATION
C      THIS PROGRAM NEEDS THE FOLLOWING SUBROUTINES:             TRAJ 055
         INITIA,VNORM,TTE,
C      NRTD,RKG,DAUX,RNORM,RMAT,GRAV,THRUST,ATTITU,IMU           TRAJ 056
C                                                                TRAJ 060
      DIMENSION RI(7),VI(7),CIB(13),X(19),F(19),C(3,3)           TRAJ 070
      DIMENSION CTE(13),RTE(7)                                   TRAJ 075
      COMMON/IMU/A(3),W(3),AM(3),WM(3),D(12,100)                 TRAJ 080
   10 FORMAT(6E10.3)                                             TRAJ 100
   20 FORMAT(I10)                                                TRAJ 105
    4 FORMAT(1H0,5X,17HFLIGHT TIME       ,17HSLANT RANGETRAJ 310
           ,17HVELOCIT
     1Y          ,17HPATH ANGLE       ,17HRANGE ANGLE            TRAJ 320
    6 FORMAT(5X,F10.2,7X,4(E15.8,2X)/)                           TRAJ 340
    8 FORMAT(10X,8HPOSITION,12X,8HVELOCITY,22X,                  TRAJ 400
         8HATTITUDE)
    9 FORMAT(5X,5(E15.8,5X))                                     TRAJ 420
   30 FORMAT(5X,10HRANGE     ,E15.8,10HELEVATION,E15.8,TRAJ 425
         10HAZIMUTH   ,
     1E15.8,)                                                    TRAJ 426
      READ(5,20) NE                                              TRAJ 106
      READ(5,20) INDEX                                           TRAJ 107
      READ(5,10) HH,TF                                           TRAJ 090
      READ(5,10) R1,R2,R3,E1,E2,E3                               TRAJ 111
      READ(5,10) A1,A2,A3,T1,T2,T3                               TRAJ 112
      READ(5,10) H1,H2,H3                                        TRAJ 113
      READ(5,10) C1,C2,C3,C4,C5                                  TRAJ 110
      CALL INITIA(RI,VI,CIB,C1,C2,C3,C4,C5)                      TRAJ 120
      DO 1 I=1,3                                                 TRAJ 130
      X(I+13)=RI(I+4)                                            TRAJ 140
      X(I+16)=VI(I+4)                                            TRAJ 150
      DO 5 J=1,3                                                 TRAJ 160
      IJ=I+3*(J-1)+4                                             TRAJ 170
```

（続き） HARP 5020 COMPILED LIST

```
    5 X(IJ)=CIB(IJ)                                                     TRAJ 180
    1 CONTINUE                                                          TRAJ 190
      CALL VNORM(X(1),RI)                                               TRAJ 200
      CALL VNORM(X(2),VI)                                               TRAJ 210
      X(3)=0.0                                                          TRAJ 220
      X(4)=0.0                                                          TRAJ 230
      J=0                                                               TRAJ 240
      SI=1                                                              TRAJ 245
  999 J=J+1                                                             TRAJ 250
      TIME=J-1                                                          TRAJ 260
      TIME=HH*TIME                                                      TRAJ 270
      JJ=MOD(J,100)                                                     TRAJ 280
      IF(JJ-1) 2,3,2                                                    TRAJ 290
    3 WRITE(6,4)                                                        TRAJ 300
      WRITE(6,6) TIME,(X(I),I=1,4)                                      TRAJ 330
      DO 7 I=1,3                                                        TRAJ 350
      DO 7 K=1,3                                                        TRAJ 360
      IK=I+3*(K-1)+4                                                    TRAJ 370
    7 C(I,K)=X(IK)                                                      TRAJ 380
      WRITE(6,8)                                                        TRAJ 390
      WRITE(6,9) (X(I+13),X(I+16),(C(I,J)          ,J=1,3),             TRAJ 410
        I=1,3)
      CALL TTE(CTE,RTE,G1,G2,G1F,G2F,H1,H2,H3)                          TRAJ 421
      CALL NRTD(RAM,ELM,AZM,TIME, OTE,RTE,X(13),X(16),R1,               TRAJ 422
        R2,R3,E1,E2,E3
     1,A1,A2,A3,T1,T2,T3,SI,INDEX)                                      TRAJ 423
      WRITE(6,30) RAM,ELM,AZM                                           TRAJ 424
    2 CONTINUE                                                          TRAJ 430
      CALL RKG(X,F,NE,HH,TIME)                                          TRAJ 440
      IF(TIME-TE) 999,999,99                                            TRAJ 450
   99 STOP                                                              TRAJ 460
      END                                                               TRAJ 470
```

HARP 5020 COMPILED LIST NRTD

```
CNRTD       HARP                                                  NRTD 010
      WORD LENGTH 64 BITS                                         NRTD 020
      SUBROUTINE NRTD(RAM,ELM,AZM,TIME,CTE,RTE,RI,                NRTD 030
        VI,SR,SFR,BR,SE,SFE,
     1BE,SA,SFA,BA,TCR,TCE,TCA,SI,INDEX)                          NRTD 040
      DIMENSION CTE(1),RTE(1),RI(1),VI(1),WN(2)                   NRTD 050
      DATA/P,Q/3.141592653590,2.718281828459/                     NRTD 060
      DATA/J/0/                                                   NRTD 065
      CALL RADAR(RA,EL,AZ,TIME,CTE,RI,VI)                         NRTD 070
      GO TO (1,2,3), INDEX                                        NRTD 080
    1 RN=0.0                                                      NRTD 090
      EN=0.0                                                      NRTD 100
      AN=0.0                                                      NRTD 110
      GO TO 10                                                    NRTD 120
    2 CALL RNORM(P,Q,1,0.0,SR,2,WN)                               NRTD 130
      RN=WN(2)                                                    NRTD 140
      CALL RNORM(P,Q,1,0.0,SE,2,WN)                               NRTD 150
      EN=WN(2)                                                    NRTD 160
      CALL RNORM(P,Q,1,0.0,SA,2,WN)                               NRTD 170
      AN=WN(2)                                                    NRTD 180
      GO TO 10                                                    NRTD 190
    3 IF(J) 100,200,100                                           NRTD 200
  200 CALL RNORM(P,Q,1,0.0,SR,2,WN)                               NRTD 210
      RN=WN(2)                                                    NRTD 220
      CALL RNORM(P,Q,1,0.0,SE,2,WN)                               NRTD 230
      EN=WN(2)                                                    NRTD 240
      CALL RNORM(P,Q,1,0.0,SA,2,WN)                               NRTD 250
      AN=WN(2)                                                    NRTD 260
      J=1                                                         NRTD 270
  100 CALL RCNG(RN,TCR,SR,SI)                                     NRTD 280
      CALL RCNG(EN,TCE,SE,SI)                                     NRTD 290
      CALL RCNG(AN,TCA,SA,SI)                                     NRTD 300
   10 RAM=RA*(1.0+SFR)+BR+RN                                      NRTD 310
      ELM=EL*(1.0+SFE)+BE+EN                                      NRTD 320
      AZM=AZ*(1.0+SFA)+BA+AN                                      NRTD 330
      RETURN                                                      NRTD 340
      END                                                         NRTD 350
```

HARP 5020 COMPILED LIST RADAR

```
CRADAR     HARP                                              RADA 010
      WORD LENGTH 64 BITS                                    RADA 020
      SUBROUTINE RADAR(RA,EL,AZ,TIME,CTE,RI,VI)              RADA 030
      DIMENSION CTE(1),RI(1),VI(1)                           RADA 040
      DIMENSION CEI(13),RE(7),RT(7)                          RADA 050
      DATA/P,EP,EQ/3.14159265359O,1.0,1.0E-8/                RADA 060
      DATA/RMAX,EMIN/5.0,0.0872664626/                       RADA 070
      CALL TEI(CEI,TIME)                                     RADA 080
      CALL MULT(RE,CEI,RI)                                   RADA 090
      CALL ADD(RE,1.0,RE,-1.0,RTE)                           RADA 100
      CALL MULT(RT,CTE,RE)                                   RADA 110
      CALL VNORM(RA,RT)                                      RADA 120
      R=RT(5)**2+RT(6)**2                                    RADA 130
      SNE=RT(7)/RA                                           RADA 140
      Q=DABS(SNE-1.0)                                        RADA 150
      IF(Q-EQ) 5,5,6                                         RADA 160
    5 EL=0.5*P                                               RADA 170
      GO TO 7                                                RADA 180
    6 CSE=DSQRT(1.0-SNE**2)                                  RADA 190
      EL=DATAN(SNE/CSE)                                      RADA 200
    7 IF(RT(6)-EP) 1,2,2                                     RADA 210
    2 AZ=DATAN(RT(5)/RT(6))                                  RADA 220
      GO TO 10                                               RADA 230
    1 IF(RT(6)+EP) 3,3,4                                     RADA 240
    3 AZ=DATAN(RT(5)/RT(6))+DSIGN(P,RT(5))                   RADA 250
      GO TO 10                                               RADA 260
    4 AZ=DSIGN(0.5*P,RT(5))                                  RADA 270
   10 CONTINUE                                               RADA 280
      IF(R-RMAX) 51,61,61                                    RADA 290
   51 WRITE(6,70)                                            RADA 320
   70 FORMAT(2X,45HAZIMUTH VALUE IS NOT AVAILABLE            RADA 330
         (WARNING ONLY))
   61 IF(EL-EMIN) 50,60,60                                   RADA 340
   50 WRITE(6,55)                                            RADA 360
   55 FORMAT(2X,47HELEVATION VALUE IS NOT AVAILABLE          RADA 370
         (WARNING ONLY))
   60 RETURN                                                 RADA 380
      END                                                    RADA 390
```

HARP 5020 COMPILED LIST IMU

```
CIMU          HARP   DECK
              WORD LENGTH 64 BITS
              SUBROUTINE IMU(AM,WM,A,W)
C       IMU I/O ERROR SIMULATION PROGRAM
C       PROGRAMED BY M.MURATA, NAL, OCTOBER 1,1973
              DIMENSION AM(1),WM(1),A(1),W(1)
              DIMENSION ASF(3,3),ASS(3,3,3),AB(3),ARS(3),
             .GSF(3,3),GB(3),GSS(3,3,3),DI(3),DS(3),ASSD
     1        (3,3,3),GRS(3),WN(4),F(3,3)
              DATA/A0,A1,A2/9.8E-4,1.0E-6,1.0E-6/
              DATA/B0,B1,B2,G1,G2/0.727E-7,1.0E-9,0E-9,
                1.0E-9,7.42E-9,7.57E-10/
              DATA/RN1,RN2/3.14159265359O,2.718281828459/
              DATA/GSF/100.0,10.0,10.0,10.0,100.0,10.0,
                10.0,10.0,100.0/
              DATA/GSS/10.0,1.0,1.0,1.0,10.0,1.0,1.0,1.0,
                10.0,10.0,1.0,1.0,
     1        1.0,10.0,1.0,1.0,1.0,10.0,10.0,1.0,1.0,1.0,
     2          10.0,1.0,1.0,1.0,10.0/
              DATA/GB/5.0,0.5.0/
              DATA/DI,DS/15.0,15.0,15.0,15.0,15.0,15.0/
              DATA/ASSD/2.0,0.5,0.5,0.5,2.0,0.5,0.5,0.5,
     1          2.0,0.5,0.5,0.5,2.0,0.5,0.5,0.5,0.0,2.0,
                0.5,0.5,0.5,0.0/
              DATA/JJ/1/
              DATA/ASF/100.0,50.0,50.0,50.0,100.0,50.0,
                50.0,50.0,100.0/
              DATA/ASS/10.0,1.0,1.0,1.0,10.0,1.0,1.0,1.0,
                10.0,10.0,1.0,1.0,
     1        1.0,10.0,1.0,1.0,1.0,10.0,10.0,1.0,1.0,1.0,
     2        10.0,1.0,1.0,1.0,10.0/
              DATA/AB/2.0,2.0,2.0/
              DO 1 I=1,3
              ARS(I)=A0
              GRS(I)=B0
     1        CONTINUE
              DO 2 J=1,3
              CALL RNORM(RN1,RN2,1,0.0,ARS(J),2,WN)
              AM(J)=A(J)+A0*AB(J)+WN(2)
              CALL RNORM(RN1,RN2,1,0.0,GRS(J),2,WN)
              WM(J)=W(J)+B0*GB(J)+WN(2)
              DO 3 L=1,3
              AM(J)=AM(J)+A1*ASF(J,L)*A(L)
              WM(J)=WM(J)+B1*GSF(J,L)*W(L)
     3        CONTINUE
              DO 4 M=1,3
              DO 5 N=1,3
              AM(J)=AM(J)+A2*ASS(M,N,J)*A(M)*A(N)
     5        WM(J)=WM(J)+B2*GSS(M,N,J)*W(M)*W(N)
     4        CONTINUE
     2        CONTINUE
              F(1,1)=A(1)
              F(2,1)=-A(2)
              F(1,2)=A(2)
              F(2,2)=A(1)
```

HARP 5020 COMPILED LIST IMU

```
      F(1,3)=A(3)
      F(2,3)=-A(1)
      F(3,1)=A(3)
      F(3,2)=A(3)
      F(3,3)=A(2)
      DO 6 J=1,3
      WM(J)=WM(J)+G1*(DI(J)*F(1,J)+DS(J)*F(2,J))
      DO 7 M=1,3
      DO 8 N=1,3
    8 WM(J)=WM(J)+G2*ASSD(M,N,J)*F(M,J)*F(N,J)
    7 CONTINUE
    6 CONTINUE
      IF(JJ-1) 10,9,10
    9 CONTINUE
      PRINT 20
   20 FORMAT(10X,39H***** LIST OF IMU HARDWARE ERRORS
       *****)
      PRINT 30,GSF,GSS,GB,DI,DS,ASSD,GRS
   30 FORMAT(20X,3(E15.8,5X))
      PRINT 40,ASF,ASS,AB,ARS
   40 FORMAT(20X,3(E15.8,5X))
   10 JJ=JJ+1
      RETURN
      END
```

HARP 5020 COMPILED LIST THRUST

```
CTHRUST HARP
      WORD LENGTH 64 BITS
      SUBROUTINE THRUST(AT,AD,TAB)
      DIMENSION AT(1),AD(1),SU(4),TC(4),WN(4),A(4)
      DATA/J,R,S/0,3.141592653590,2.718281828459/
      DATA/SU,TC/0.002,0.002,0.002,0.0025,200.0,200.0,200.0,200.0/
C
C     THRUST CONTROL FLUCTUATION BY RANDOM DRIFTS
C
      IF(J) 1,2,1
    2 DO 3 I=1,4
      CALL RNORM(R,S,1,0,0,SU(I),2,WN)
      A(I)=WN(2)
    3 CONTINUE
      J=1
    1 DO 4 I=1,4
    4 CALL RCNG(A(I),TC(I),SU(I),0.0025)
      AT(1)=TAB*(1.0+A(4))*(AD(1)+A(2)*AD(3)-A(3)*AD(2))
      AT(2)=TAB*(1.0+A(4))*(AD(2)+A(3)*AD(1)-A(1)*AD(3))
      AT(3)=TAB*(1.0+A(4))*(AD(3)+A(1)*AD(2)-A(2)*AD(1))
      RETURN
      END
```

HARP 5020 COMPILED LIST DAUX

```
CDAUX      HARP   DECK                                                   DAUX 000
           WORD LENGTH 64 BITS                                           DAUX 010
           SUBROUTINE DAUX(YS,FS,T)                                      DAUX 020
           DOUBLE PRECISION MU                                           DAUX 030
           DIMENSION YS(1),FS(1)                                         DAUX 040
           DIMENSION           U(6),FM(6),THETA(16),TA(16),TS(6)         DAUX 050
           DIMENSION G(7),R(7),B(13),UC(7)                               DAUX 060
           COMMON/IMJ/A(3),W(3),AM(3),WM(3),D(12,100)                    DAUX 070
           DATA/G(1),G(2),G(3),G(4)/3.0,1.0,3.0,7.0/                     DAUX 080
           DATA/U,FM/284.999,0.0,284.996,0.0,290.0,0.0,173172.0,38293.0, DAUX 090
     1     27672.0,6817.0,4754.0,1414.0/                                 DAUX 100
           DATA/TA/0.0,15.0,143.80,162.00,192.00,222.00,252.00,282.00,312.00,DAUX 110
     1     342.00,352.30,372.30,392.30,412.30,412.50,1000.00/            DAUX 120
           DATA/THETA/90.000,90.000,21.005,18.637,14.612,10.499,6.361,2.279,DAUX 130
     1     -1.649,-5.298,-6.476,-8.685,-10.785,-12.780,0.000,0.000/      DAUX 140
           DATA/MU,DR,G0/0.0398603068E+16,57.295779578,9.8/              DAUX 150
           DATA/TS/0.0,143.870,144.000,342.120,343.0,536.72/             DAUX 160
           DATA/TBO/412.0/                                               DAUX 170
           DATA/JJ,LJ/1,0/                                               DAUX 180
           IF(T-TBO) 1000,20,20                                          DAUX 190
 1000      CONTINUE                                                      DAUX 200
           DO 1 I=1,5                                                    DAUX 210
           II=I+1                                                        DAUX 220
           S=(T-TS(I))*(T-TS(II))                                        DAUX 230
           IF(S) 2,2,1                                                   DAUX 240
    1      CONTINUE                                                      DAUX 250
    2      CONTINUE                                                      DAUX 260
           DM=(FM(I)-FM(II))/(TS(I)-TS(II))                              DAUX 270
           GO TO 21                                                      DAUX 280
   20      DM=0.0                                                        DAUX 290
           I=5                                                           DAUX 300
           FMM=FMM                                                       DAUX 310
           GO TO 30                                                      DAUX 320
   21      CONTINE                                                       DAUX 330
           FMM=FM(I)+DM*(T-TS(I))                                        DAUX 340
   30      CONTINE                                                       DAUX 350
           TAB=-U(I)*GO*DM/FMM                                           DAUX 360
           DO 3 I=1,15                                                   DAUX 370
           II=I+1                                                        DAUX 380
           S=(T-TA(I))*(T-TA(II))                                        DAUX 390
           IF(S) 4,4,3                                                   DAUX 400
    3      CONTINUE                                                      DAUX 410
    4      CONTINUE                                                      DAUX 420
           DTH=(THETA(II)-THETA(I))/(TA(II)-TA(I))                       DAUX 430
           SK=THETA(I)*DTH*(T-TA(I))                                     DAUX 440
           TH=SK/DR                                                      DAUX 450
           ALPH=TH-YS(3)                                                 DAUX 460
           CAL=DCOS(ALPH)                                                DAUX 470
           SAL=DSIN(ALPH)                                                DAUX 480
           CGA=DCOS(YS(3))                                               DAUX 490
           SGA=DSIN(YS(3))                                               DAUX 500
           FS(1)=YS(2)*SGA                                               DAUX 510
           FS(2)=TAB*CAL-MU*SGA/YS(1)/YS(1)                              DAUX 520
           FS(3)=TAB*SAL/YS(2)-MU*CGA/YS(2)/YS(1)/YS(1)+YS(2)            DAUX 530
           *CGA/YS(1)
```

(続き) HARP 5020 COMPILED LIST DAUX

```
        FS(4)=YS(2)*CGA/YS(1)
        WD=DTH/DR-FS(4)
        CALL ATTITU(W,WD)
        DO 10 I=1,3
  10    G(I+4)=W(I)
        CALL RMAT(B,G)
        DO 6 I=1,3
        DO 6 J=1,3
        IJ=I+3*(J-1)
        FS(IJ+4)=0.0
        DO 7 K=1,3
        IK=I+3*(K-1)
        KJ=K+3*(J-1)+4
   7    FS(IJ+4)=FS(IJ+4)+YS(IK+4)*B(KJ)
   6    CONTINUE
        DO 5 I=1,3
        FS(I+13)=YS(I+16)
   5    R(I)=YS(I+13)
        CALL GRAVIT(G,R)
C
C     UC = THRUST CONTROL VECTOR
C
        UC(1)=0.0
        UC(2)=0.0
        UC(3)=1.0
        CALL THRUST(A,UC,TAB)
        DO 8 I=1,3
        FS(I+16)=G(I+4)
        DO 9 J=1,3
        IJ=I+3*(J-1)
   9    FS(I+16)=FS(I+16)+YS(IJ+4)*A(J)
   8    CONTINUE
        KK=MOD(JJ,4)
        IF(KK-1) 100,200,100
 200    CALL IMU(AM,WM,A,W)
        LJ=LJ+1
        DO 300 L=1,3
        D(L,LJ)=AM(L)
        D(L+3,LJ)=A(L)
        D(L+6,LJ)=WM(L)
 300    D(L+9,LJ)=W(L)
        IF(LJ-100) 55,60,55
  60    LJ=0
      PRINT 34
  34  FORMAT(10X,20HACCEL MEASURED      ,20HACCEL ACTUAL        ,
     120HANGUL MEASURED      ,20HANGUL ACTUAL        )
      PRINT 33,((D(L,1),D(L+3,1),D(L+6,1),D(L+9,1)),L=1,3)
  33  FORMAT(1H ,10X,4(E15.8,5X))
      ALPH=DR*ALPH
      WRITE(2) D,YS,FS,TAB,FMM,WD,DTH,SK,ALPH
  55    CONTINUE
 100    JJ=JJ+1
        RETURN
        END
```

HARP 5020 COMPILED LIST INITIAL

```
CINITIAL   HARP
      WORD LENGTH 64 BITS
      SUBROUTINE INITIA(R,V,C,LH,LON,GLA,A,E)
      DIMENSION R(1),V(1),C(1)
      DIMENSION D(13),H(13)
      DOUBLE PRECISION LH,LON
      DATA/RE,EC,OMEGA,DR/0.0637816490E8,0.003352329,7.292115E-
     15,57.295779578/
      LON=LON/DR
      GLA=GLA/DR
      TNG=DSIN(GLA)/OCOS(GLA)
      X=1.0-EC**2
      DLA=DATAN2(TNG,X)
      SD=DSIN(DLA)
      CD=DCOS(DLA)
      SL=DSIN(LON)
      CL=DCOS(LON)
      B=1.0-(EC*SD)**2
      B=DSQRT(B)
      G1=LH+RE/B
      G2=LH+RE*X/B
      CALL LET(R,3,1)
      CALL LET(V,3,1)
      CALL LET(C,3,3)
      R(5)=G1*CD*CL
      R(6)=G1*CD*SL
      R(7)=G2*SD
      V(5)=-OMEGA*R(6)
      V(6)=OMEGA*R(5)
      V(7)=0.0
      A=A/DR
      E=E/DR
      CA=DCOS(A)
      SA=DSIN(A)
      SE=DSIN(E)
      CE=DCOS(E)
      D(5)=CA
      D(6)=SA*SE
      D(7)=CE*SA
      D(8)=-SA
      D(9)=SE*CA
      D(10)=CE*CA
      D(11)=0.0
      D(12)=-CE
      D(13)=SE
      CALL LET(H,3,3)
      CALL SET(H)
      SF=DSIN(DLA)
      CF=DCOS(DLA)
      H(8)=1.0
      H(6)=-SF
      H(12)=CF
      H(7)=CF
      H(13)=SF
      CALL LET(D,3,3)
```

(続き) HARP 5020 COMPILED LIST INITIAL

```
      CALL MULT(C,D,H)                                          INIT 560
      CALL TRANSP(D,C)                                          INIT 570
      CALL MAKE(C,D)                                            INIT 580
      PRINT 10                                                  INIT 590
   10 FORMAT(5X,38HINITIAL VALUES IN INERTIAL COORDINATE)       INIT 600
      PRINT 20                                                  INIT 610
   20 FORMAT(10X,15HPOSITION VECTOR,/)                          INIT 620
      CALL MPRINT(R)                                            INIT 630
      PRINT 30                                                  INIT 640
   30 FORMAT(10X,15HVELOCITY VECTOR,/)                          INIT 650
      CALL MPRINT(V)                                            INIT 660
      PRINT 40                                                  INIT 670
   40 FORMAT(10X,16HDIRECTION MATRIX,/)                         INIT 680
      CALL MPRINT(C)                                            INIT 690
      RETURN                                                    INIT 700
      END                                                       INIT 710
```

HARP 5020 COMPILED LIST GRAVITY

```
CGRAVITY   HARP                                                 GRAV 110
      SUBROUTINE GRAN(G,R)                                      GRAV 120
      DIMENSION G(1),R(1),U(3),B(3)                             GRAV 130
      DOUBLE PRECISION MU,J2                                    GRAV 150
      DATA/MU,J2,A/398606.4E9,1082.32E-6,6371043.0/             GRAV 140
      CALL MAKE(G,R)                                            GRAV 160
      RA2=R(5)**2+R(6)**2+R(7)**2                               GRAV 170
      RA1=DSQRT(RA2)                                            GRAV 180
      DO 1 J=1,3                                                GRAV 190
    1 U(J)=R(J+4)/RA1                                           GRAV 200
      C1=(A/RA1)**2                                             GRAV 210
      C2=-MU/RA2                                                GRAV 220
      B(1)=1.0+1.5*J2*C1*(1.0-5.0*U(3)**2)                      GRAV 230
      B(2)=B(1)                                                 GRAV 240
      B(3)=1.0+1.5*J2*C1*(3.0-5.0*U(3)**2)                      GRAV 250
      DO 2 J=1,3                                                GRAV 260
    2 G(J+4)=C2*B(J)*U(J)                                       GRAV 270
      RETURN                                                    GRAV 280
      END                                                       GRAV 290
```

2. カルマンフィルターによる軌道および誘導パラメータの推定プログラム

メインプログラムに入る前に以下のオフライン計算が必要である。

(1) 各ステーションのトラッキングデータはカルマンフィルタールーチンへの入力として適切なデータ構造に変換されていなければならない。即ちメインプログラムの

READ(4) T, YM, YMM, HXX, HBB, DY

において，GRTに関する観測残差ベクトル$DY(I, J)$と観測感度行列$HXX(M, J)$, $HBB(N, J)$が磁気テープに格納されていなければならない。ここで

J ; 各ステーションにASSIGNされた番号

T ; 観測時間

$YM(I, J)$; 観測値

$YMM(I, J)$; GRTに関する観測規準値

$DY(I, J)$; $YM(I, J)-YMM(I, J)$

$HXX(M, J)$; H_X

$HBB(N, J)$; H_B

(2) カルマンサイクル間の遷移行列が磁気テープに格納されていなければならない。即ちメインプログラムの

READ(2) XJ, YJ, QJ

において

XJ ;	$\Phi_{11}(t_{j+1}, t_j)$	(76)式
YJ ;	$\Phi_{12}(t_{j+1}, t_j)$	(76)式
QJ ;	$\Xi(t_j)$	(82)式

が計算されていなければならない。

メインプログラム

NN ; 飛翔体の位置・速度および姿勢を規定するベクトルの次元。STABLE PLATFORM 方式の場合，$NN=6$, STRAPDOWN方式の場合 $NN=9$。

NA ; IMUハードウェアの系統的誤差ベクトルの次元。

NBK ; K番目のステーションにおける系統的観測誤差ベクトルの次元($K=1, 2$)。

NMK ; K番目のステーションにおける観測ベクトルの次元。

JF ; 計算終了時のステップ数

$JB(K)$; K番目のステーションの観測開始時のステップ数

PX, PA, PB ; それぞれ飛翔体の状態ベクトル，IMUハードウェア誤差ベクトル，およびステーション誤差ベクトルの初期推定誤差の1σ値。

RS ; 観測ランダムノイズの1σ値。

備考1. メインプログラムのRUNには次のサブプログラムが必要である。

$PRED$; カルマンサイクル間の予測値を計算

$KALFIL$; カルマンフィルターにより観測データを加工処理する。

HARP 5020 COMPILED LIST

```
C                                                                  EST 0000
C     TRAJECTORY RECONSTRUCTION AND GUIDANCE ERROR ANALYSIS PROGRAMEST 0010
C     BY CONVENTIONALT( OPTIMAL ) KALMAN FILTERING METHOD          EST 0020
C                                                                  EST 0030
      WORD LENGTH 64 BITS                                          EST 0040
      DRUM DIMENSION D(10000)                                      EST 0050
      DIMENSION HXX(31,2),HBB(25,2)                                EST 0060
      DIMENSION DBB(260),DAB(1300),DXB(150),DB(20)                 EST 0070
      DIMENSION PBB(260),PAB(1300),PXB(150)                        EST 0080
      DIMENSION PXX(85),PXA(740),PAA(6600),EX(13),EA(85),EB(20),DZ(12) EST 0090
      DIMENSION PX(9),PA(81),PB(14)                                EST 0100
      DIMENSION R(70),HX(80),HB(120),RS(8)                         EST 0110
      DIMENSION XJ(85),YJ(740),QJ(85)                              EST 0120
      DIMENSION JB(2)                                              EST 0130
      COMMON/RADAR/ DY(3,5),YM(3,5),YMM(3,5)                       EST 0140
      COMMON/KALF/FF(10000)                                        EST 0150
      COMMON/PRED/DA(1000)                                         EST 0160
      EQUIVALENCE (FF(1),DBB(1)),(FF(400),DAB(1)),(FF(4000),DXB(1)) EST 0170
     1,(FF(5000),DB(1))                                            EST 0180
      EQUIVALENCE (PA(1),DA(1)),(PX(1),DA(100)),(PB(1),DA(200)),(RS(1) EST 0190
     1,DA(300))                                                    EST 0200
      DATA/DR/57.29577951308232/                                   EST 0220
   10 FORMAT(4I10)                                                 EST 0230
   15 FORMAT(2I10)                                                 EST 0240
   20 FORMAT(8E10.3)                                               EST 0250
   30 FORMAT(3I10)                                                 EST 0260
   60 FORMAT(2X,6(E15.8,3X))                                       EST 0270
   85 FORMAT(1H0,5X,5HTIME=,F10.2/)                                EST 0280
  760 FORMAT(1H ,3(5X,E15.8))                                      EST 0290
      READ(5,10) NN,NA,NB1,NB2                                     EST 0300
      READ(5,15) NM1,NM2                                           EST 0310
      READ(5,30) JF,JB                                             EST 0320
      NM=NM1+NM2                                                   EST 0330
      NB=NB1+NB2                                                   EST 0340
      READ(5,20) (PX(I),I=1,NN)                                    EST 0350
      READ(5,20) (PA(I),I=1,NA)                                    EST 0360
      READ(5,20) (PB(I),I=1,NB )                                   EST 0370
      READ(5,20) (RS(K),K=1,NM )                                   EST 0380
      CALL LET(HX,NM ,NN)                                          EST 0390
      CALL SET(HX)                                                 EST 0400
      CALL LET(HB,NM ,NB)                                          EST 0410
      CALL SET(HB)                                                 EST 0420
      CALL LET(PXX,NN,NN)                                          EST 0430
      CALL LET(PXA,NN,NA)                                          EST 0440
      CALL LET(PXB,NN,NB )                                         EST 0450
      CALL LET(PAA,NA,NA)                                          EST 0460
      CALL LET(PAB,NA,NB )                                         EST 0470
      CALL LET(PBB,NB ,NB )                                        EST 0480
      CALL LET(EX,NN,1)                                            EST 0490
      CALL LET(EA,NA,1)                                            EST 0500
      CALL LET(EB,NB ,1)                                           EST 0510
      CALL LET(DZ,NM ,1)                                           EST 0520
      CALL MAKE(XJ,PXX)                                            EST 0560
      CALL MAKE(YJ,PXA)                                            EST 0540
      CALL MAKE(QJ,PXX)                                            EST 0550
```

(続き) HARP 5020 COMPILED LIST

```
      CALL SET(PXA)                                      EST 0560
      CALL SET(PXB)                                      EST 0570
      CALL SET(PAB)                                      EST 0580
      CALL SET(PXX)                                      EST 0590
      CALL SET(PAA)                                      EST 0600
      CALL SET(PBB)                                      EST 0610
      DO 1 I=1,NN                                        EST 0620
      II=I+NN*(I-1)+4                                    EST 0630
    1 PXX(II)=PX(I)**2                                   EST 0640
      DO 2 I=1,NA                                        EST 0650
      II=I+NA*(I-1)+4                                    EST 0660
    2 PAA(II)=PA(I)**2                                   EST 0670
      DO 3 I=1,NB                                        EST 0680
      II=I+NB *(I-1)+4                                   EST 0690
    3 PBB(II)=PB(I)**2                                   EST 0700
      CALL LET(R,NM ,NM)                                 EST 0710
      CALL SET(R)                                        EST 0720
      DO 4 I=1,NM                                        EST 0730
      II=I+NM *(I-1)+4                                   EST 0740
    4 R(II)=RS(I)**2                                     EST 0750
      J=0                                                EST 0820
  999 J=J+1                                              EST 0830
      T=J-1                                              EST 0840
      PRINT 85,T                                         EST 0850
      DO 21 I=1,NB                                       EST 0860
      II=I+NB*(I-1)+4                                    EST 0870
   21 PB(I)=PBB(II)                                      EST 0880
      DO 22 I=1,NA                                       EST 0890
      II=I+NA*(I-1)+4                                    EST 0900
   22 PA(I)=PAA(II)                                      EST 0910
      DO 23 I=1,NN                                       EST 0920
      II=I+NN*(I-1)+4                                    EST 0930
   23 PX(I)=PXX(II)                                      EST 0940
      WRITE(6,60) (PX(I),I=1,NN)                         EST 0950
      WRITE(6,60) (PA(I),I=1,NA)                         EST 0960
      WRITE(6,60) (PB(I),I=1,NB)                         EST 0970
      WRITE(6,60) (EX(I),I=5,NN+4)                       EST 0980
      WRITE(6,60) (EA(I),I=5,NA+4)                       EST 0990
      WRITE(6,60) (EB(I),I=5,NB+4)                       EST 1000
      KK=DZ(4)+0.01                                      EST 1010
      PRINT 760, (DZ(K),K=5,KK)                          EST 1020
      READ(4) T,YM,YMM,HXX,HBB,DY                        EST 1110
      IF(J-JB(1)) 92,93,93                               EST 1120
   93 IF(J-JB(2)) 94,95,95                               EST 1130
   95 CONTINUE                                           EST 1140
      CALL SET(HX)                                       EST 1150
      DO 51 N=1,NN                                       EST 1160
      DO 52 M=1,NM1                                      EST 1170
      MJ=M+NM*(N-1)+4                                    EST 1180
      MN=M+NM1*(N-1)+4                                   EST 1190
   52 HX(MJ)=HXX(MN,1)                                   EST 1200
      N1=NM1+1                                           EST 1210
      DO 53 M=N1,NM                                      EST 1220
      MJ=M-NM1+NM2*(N-1)+4                               EST 1230
      MN=M+NM*(N-1)+4                                    EST 1240
```

(続き) HARP 5020 COMPILED LIST

```
   53 HX(MN)=HXX(MJ,2)                                          EST 1250
   51 CONTINUE                                                  EST 1260
      CALL SET(HB)                                              EST 1270
      DO 54 M=1,NM1                                             EST 1280
      DO 54 N=1,NB1                                             EST 1290
      MN=M+NM1*(N-1)+4                                          EST 1300
      MJ=M+NM*(N-1)+4                                           EST 1310
   54 HB(MJ)=HBB(MN,1)                                          EST 1320
      N2=NB1+1                                                  EST 1330
      DO 55 M=N1,NM                                             EST 1340
      DO 55 N=N2,NB                                             EST 1350
      MJ=M-NM1+NM2*(N-NB1-1)+4                                  EST 1360
      MN=M+NM*(N-1)+4                                           EST 1370
   55 HB(MN)=HBB(MJ,2)                                          EST 1380
      DO 56 I=1,NM1                                             EST 1390
   56 DZ(I+4)=DY(I,1)/DR                                        EST 1400
      II=4+NM1                                                  EST 1410
      DO 57 I=1,NM2                                             EST 1420
      I1=I+II                                                   EST 1430
   57 DZ(I1  )=DY(I,2)/DR                                       EST 1440
      GO TO 99                                                  EST 1450
   94 CONTINUE                                                  EST 1460
      DO 61 K=1,NM1                                             EST 1470
   61 DZ(K+4)=DY(K,1)/DR                                        EST 1480
      DO 62 K=1,NM1                                             EST 1490
      DO 62 L=1,NB1                                             EST 1500
      KL=K+NM1*(L-1)+4                                          EST 1510
      KK=K+NM*(L-1)+4                                           EST 1520
   62 HB(KK)=HBB(KL,1)                                          EST 1530
      DO 63 K=1,NM1                                             EST 1540
      DO 63 L=1,NN                                              EST 1550
      KK=K+NM*(L-1)+4                                           EST 1560
      KL=K+NM1*(L-1)+4                                          EST 1570
   63 HX(KK)=HXX(KL,1)                                          EST 1580
   99 CONTINUE                                                  EST 1590
      CALL KALFIL(PXX,PXA,PXB,PAA,PAB,PBB,EX,EA,EB,R,DZ,HX,HB) EST 1600
      IF(J-421) 150,160,150                                     EST 1610
  160 CALL MPRINT(PXX)                                          EST 1620
      CALL MPRINT(PXA)                                          EST 1630
      CALL MPRINT(PXB)                                          EST 1640
      CALL MPRINT(PAA)                                          EST 1650
      CALL MPRINT(PAB)                                          EST 1660
      CALL MPRINT(PBB)                                          EST 1670
      CALL MPRINT(EX)                                           EST 1680
      CALL MPRINT(EA)                                           EST 1690
      CALL MPRINT(EB)                                           EST 1700
      CALL MPRINT(DZ)                                           EST 1710
  150 CONTINUE                                                  EST 1720
   92 CONTINUE                                                  EST 1730
      IF(J-JF) 990,1000,1000                                    EST 1740
  990 READ(2) XJ,YJ,QJ                                          EST 1800
      CALL PRED(PXX,PXA,PXB,PAA,PAB,PBB,EX,EA,EB,XJ,YJ,QJ)      EST 1810
      GO TO 999                                                 EST 1830
 1000 STOP                                                      EST 1840
      END                                                       EST 1850
```

HARP 5020 COMPILED LIST PREDICT

```
CPREDICT    HARP                                                       PRED  110
      SUBROUTINE PRED(PXX,PXA,PXE,FAA,PAB,PBE,EX,EA,EB,XJ,YJ,QJ)PRED  120
      DIMENSION PXX(1),PXA(1),PAA(1),EX(1),EA(1),XJ(1),YJ(1),GJ(1),     PRED  130
     1PXB(1),PAB(1),PBB(1),EB(1)                                        PRED  135
      DIMENSION A(85),B(740)                                            PRED  140
C     N IS LESS THAN 9                                                  PRED  160
C     NA IS LESS THAN 81                                                PRED  170
C     NB IS LESS THAN 7                                                 PRED  180
      COMMON/PRED/DA(1000)                                              PRED  190
      EQUIVALENCE ( B(1),DA(1)),( A(1),DA(801))                         PRED  195
      CALL MULT(A ,XJ,EX)                                               PRED  200
      CALL MULT(EX,YJ,EA)                                               PRED  210
      CALL ADD(EX,1.0,EX,1.0,A)                                         PRED  220
C                                                                       PRED  230
      CALL MULTP( A,PXX,XJ)                                             PRED  240
      CALL MULTP( B,PXA,YJ)                                             PRED  250
      CALL ADD( A,1.0, A,1.0,B)                                         PRED  260
C           COMPUTE PXA                                                 PRED  270
      CALL MULT( B,XJ,PXA)                                              PRED  280
      CALL MULT(PXA,YJ,PAA)                                             PRED  290
      CALL ADD(PXA,1.0,PXA,1.0,B)                                       PRED  300
C           COMPUTE PXB                                                 PRED  310
      CALL MULT( B,XJ,PXB)                                              PRED  320
      CALL MULT(PXB,YJ,PAB)                                             PRED  330
      CALL ADD(PXB,1.0,PXB,1.0,B)                                       PRED  340
C           COMPUTE PXX                                                 PRED  350
      CALL MULT(PXX,XJ,A )                                              PRED  360
      CALL MULTP(B ,YJ,PXA)                                             PRED  370
      CALL ADD(PXX,1.0,PXX,1.0,B )                                      PRED  380
      CALL ADD(PXX,1.0,PXX,1.0,QJ)                                      PRED  390
      RETURN                                                            PRED  400
      END                                                               PRED  410
```

HARP 5020 COMPILED LIST KALFIL

```
CKALFIL     HARP    64
      SUBROUTINE KALFIL(PXX,PXA,PXB,PAA,PAB,PBB,EX,EA,EB,R,DZ,HX,HB)
      DIMENSION PXX(1),PXA(1),PXB(1),PAA(1),PAB(1),PBB(1),EX(1),EA(1),
     1EB(1),R(1),DZ(1),HX(1),HB(1)
      DIMENSION C(85),A(68),DT(6565),DA(76),DM(76),DC(655),DN(655),DD
     1(132),DP(132),DJ(76),DK(655),DL(132),GM(8,8)
      COMMON/KALF/FF(10000)
      EQUIVALENCE (FF(1),DT(1)),(FF(6566),DC(1)),(FF(7221),DN(1)),(FF
     1(7876),DK(1)),(FF(8531),DD(1)),(FF(8663),DP(1)),(FF(8795),DL(1)),
     2(FF(8927),C(1)),(FF(9012),A(1)),(FF(9080),DA(1)),(FF(9156),DM(1)),
     3(FF(9232),DJ(1))
C     C(MAX(N,A,B,R*R )+4),A(R*R +4),DT(MAX(N**2,N*A,N*B,A*B,
C     A*A,B*B)+4),DA(R*N+4),DM(R*N+4),DC(R*A+4),DN(R*A+4),DD(R*B
C     +4),DP(R*B+4),DJ(N*R+4),DK(A*R+4),DL(B*R+4),GM(R,R)
C           COMPUTE MEASUREMENT RESIDUAL
      CALL MULT(DA,HX,EX)
      CALL MULT(DM,HB,EB)
      CALL ADD(DZ,1.0,DZ,-1.0,DA)
      CALL ADD(DZ,1.0,DZ,-1.0,DM)
C
      CALL MULT(DA,HX,PXX)
      CALL MULTP(DM,HB,PXB)
      CALL ADD(DA,1.0,DA,1.0,DM)
C
      CALL MULT(DC,HX,PXA)
      CALL MULTP(DN,HB,PAB)
      CALL ADD(DC,1.0,DC,1.0,DN)
C
      CALL MULT(DD,HX,PXB)
      CALL MULT(DP,HB,PBB)
      CALL ADD(DD,1.0,DD,1.0,DP)
C
      CALL MULTP(C,HX,DA)
      CALL MULTP(A,HB,DD)
      CALL ADD(C,1.0,C,1.0,A)
C
      CALL MAKE(A,R)
      CALL ADD(C,1.0,C,1.0,A)
C
      II=C(1)+0.01
      DO 1 I=1,II
      DO 1 J=1,II
      IJ=I+II*(J-1)+4
    1 GM(I,J)=C(IJ)
      CALL MATINV(GM,II,C,0,DET,II)
      DO 2 I=1,II
      DO 2 J=1,II
      IJ=I+II*(J-1)+4
    2 A(IJ)=GM(I,J)
C
      CALL TRANSP(DJ,DA)
      CALL TRANSP(DK,DC)
      CALL TRANSP(DL,DD)
C
C           COMPUTE KALMAN GAIN MATRIX
```

(続き) HARP 5020 COMPILED LIST KALFIL

```
C                                                          KALF 445
      CALL MULT(DM,DJ,A)                                   KALF 450
      CALL MULT(DP,DL,A)                                   KALF 460
      CALL MULT(DN,DK,A)                                   KALF 470
C                                                          KALF 475
      CALL MULT(DT,DM,DA)                                  KALF 480
      CALL ADD(PXX,1.0,PXX,-1.0,DT)                        KALF 490
      CALL MULT(DT,DM,DC)                                  KALF 500
      CALL ADD(PXA,1.0,PXA,-1.0,DT)                        KALF 510
      CALL MULT(DT,DM,DD)                                  KALF 520
      CALL ADD(PXB,1.0,PXB,-1.0,DT)                        KALF 530
      CALL MULT(DT,DN,DC)                                  KALF 540
      CALL ADD(PAA,1.0,PAA,-1.0,DT)                        KALF 550
      CALL MULT(DT,DN,DD)                                  KALF 560
      CALL ADD(PAB,1.0,PAB,-1.0,DT)                        KALF 570
      CALL MULT(DT,DP,DD)                                  KALF 580
      CALL ADD(PBB,1.0,PBB,-1.0,DT)                        KALF 590
C                                                          KALF 595
C           UPDATE ESTIMATE                                KALF 600
C                                                          KALF 605
      CALL MULT(C,DM,DZ)                                   KALF 610
      CALL ADD(EX,1.0,EX,1.0,C)                            KALF 620
      CALL MULT(C,DN,DZ)                                   KALF 630
      CALL ADD(EA,1.0,EA,1.0,C)                            KALF 640
      CALL MULT(C,DP,DZ)                                   KALF 650
      CALL ADD(EB,1.0,EB,1.0,C)                            KALF 660
C                                                          KALF 665
      RETURN                                               KALF 670
      END                                                  KALF 680
```

航空宇宙技術研究所資料266号

昭和49年11月発行

発行所　航空宇宙技術研究所
東京都調布市深大寺町1880
電話武蔵野三鷹(0422)47-5911(大代表)〒182

印刷所　株式会社　共進
東京都杉並区久我山4－1－7(羽田ビル)

Printed in Japan