REPORT SAMSO-TR-75-297
(SUPERSEDES TR-73-220)

1 NW

ADA040475

Titan IIIC Guidance with the Carousel VB Inertial Guidance System

T. E. DARNAUD and J. B. SHAUL
Guidance and Control Division
Engineering Science Operations
The Aerospace Corporation
El Segundo, Calif. 90245

15 July 1975

APPROVED FOR PUBLIC RELEASE:
DISTRIBUTION UNLIMITED

Prepared for

SPACE AND MISSILE SYSTEMS ORGANIZATION
AIR FORCE SYSTEMS COMMAND
Los Angeles Air Force Station
P.O. Box 92960, Worldway Postal Center
Los Angeles, Calif. 90009

Approved

J. R. Allder, Director
Guidance and Flight Dynamics
Subdivision
Guidance and Control Division
Engineering Science Operations

D. Baxter, Director
Titan III Directorate
Vehicle Systems Division
Systems Engineering Operations

Publication of this report does not constitute Air Force approval of the report's findings or conclusions. It is published only for the exchange and stimulation of ideas.

RICHARD E. WOLFSBERGER, Colonel, USAF
Assistant Program Director
Expendable Launch Vehicles SPO

This report supersedes and replaces TR-0073(3413-02)-1, dated 29 November 1972.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER (17) (18) SAMSO TR-75-297	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) (6) TITAN IIIC GUIDANCE WITH THE CAROUSEL VB INERTIAL GUIDANCE SYSTEM. Reissue A,	5. TYPE OF REPORT & PERIOD COVERED
	6. PERFORMING ORG. REPORT NUMBER (14) TR-0073(3413-02)-1-A Reissue A
7. AUTHOR(s) (10) T. E. Darnaud [redacted] J. B. Shaul	8. CONTRACT OR GRANT NUMBER(s) (15) F04701-72-C-0073, F04701-75-C-0076
9. PERFORMING ORGANIZATION NAME AND ADDRESS The Aerospace Corporation El Segundo, Calif. 90245	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Space and Missile Systems Organization Los Angeles Air Force Station Los Angeles, Calif. 90045	12. REPORT DATE (11) 15 Jul[redacted] [redacted]75
	13. NUMBER OF PAGES (12) 58 p.
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Guidance Equations
Inertial Navigation
Carousel VB
Rocket Steering Law
Vehicle-Borne Computer Program

3C

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The Titan IIIC Standard Space Launch Vehicle, starting with Vehicle 26, will use the Carousel VB Inertial Guidance System for navigation, guidance, and digital flight controls. This guidance system consists of a Carousel VB Inertial Measurement Unit (IMU) and a MAGIC 352 Missile Guidance Computer (MGC), both manufactured by Delco Electronics, General Motors Corporation. The C-VB IMU is a modification of the C-IVB inertial navigator currently in airline service as a primary navigation system.

DD FORM 1473 (FACSIMILE)

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

404068

19. KEY WORDS (Continued)

20. ABSTRACT (Continued)

A unique feature of the Carousel IMU is that two of the three gyro/accelerometer sets are revolved at 1 rpm with respect to the stable platform. The effect is to partially cancel certain instrument errors associated with the revolving instruments. Conversion of this instrument from commercial airline navigation service to guidance and control of the rocket boost vehicle presented a guidance software design task that is the subject of this paper.

PREFACE

The guidance equations derived in this paper were developed by the authors in a joint effort with D. L. Kleinbub and A. C. Liang of The Aerospace Corporation.

A summary of the first issue of this report was presented at the Sixth Hawaii International Conference on System Sciences January 11, 1973 and was published in the proceedings.

CONTENTS

CONTENTS (Continued)

FIGURES

TABLE

SECTION I

INTRODUCTION

Reference 1 is a complete specification of the guidance equations for the Titan IIIC launch vehicle. It includes all information necessary for programming the launch vehicle guidance computer. Reference 1 does not, however, include any discussion of the guidance algorithms or the derivation of equations. The equations specified by Ref. 1 and discussed in this document are applicable to SSLV C-26 and subsequent vehicles.

Section II, a very general description of a typical Flight Plan VII mission and of the Titan IIIC vehicle, is included as an aid to understanding the inflight equations, some aspects of which are vehicle- and/or mission-peculiar.

Section III is a discussion of the Titan IIIC navigation, guidance, and steering.

SECTION II

MISSION SUMMARY AND VEHICLE DESCRIPTION

A. GENERAL

The Flight Plan VII mission consists of injecting a payload(s) into a synchronous or near-synchronous equatorial orbit using the Titan IIIC SSLV. The combined vehicle Stages II and III, together with the payload, are injected directly into an elliptical parking orbit having a perigee of approximately 80 nmi and an apogee of approximately 235 nmi. At the first descending or ascending node, depending on mission options, a first burn of Stage III produces an elliptical orbit with an apogee of approximately 19,323 nmi and an orbital inclination reduced by approximately 2-1/4 deg. At apogee, a second Stage III burn produces the desired synchronous (or near synchronous) equatorial orbit.

B. VEHICLE DESCRIPTION

The Titan IIIC launch vehicle shown in Fig. 1 consists of a solid motor stage and three liquid engine stages plus a control module. The solid motor stage is referred to as Stage 0, and the three liquid engine stages are referred to as Core Stages I and II, and Stage III. The control module and Stage III are sometimes collectively called the transtage. The stages are briefly described in the following paragraphs.

1. STAGE 0

Stage 0 consists of two solid propellant rocket motors positioned parallel to the standard core in the yaw plane. Thrust vector control is accomplished by injecting oxidizer (N_2O_4), pressurized by gaseous nitrogen (N_2), into any of four quadrants of the nozzles.

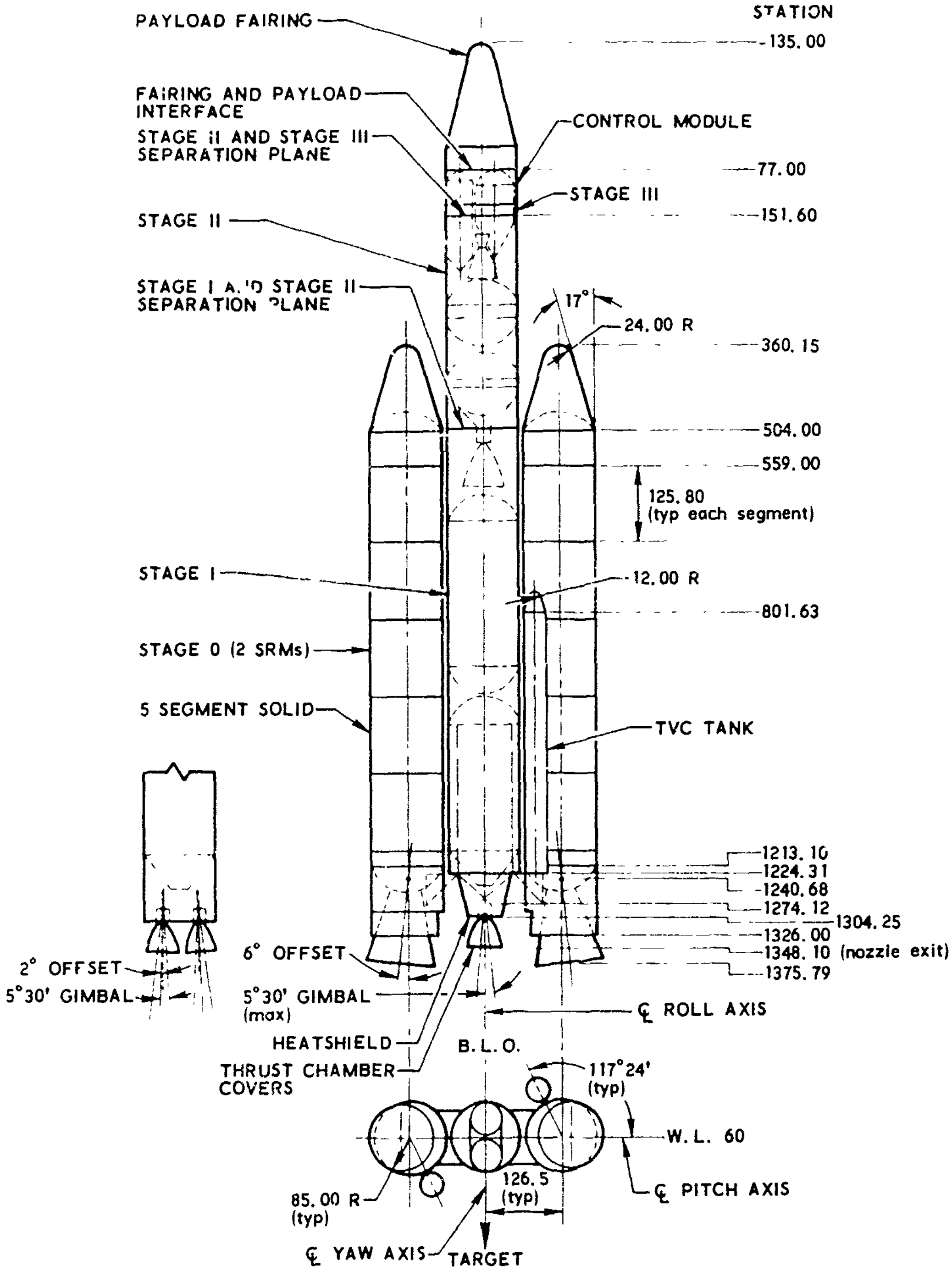


Figure 1. Titan IIIC External Profile

2. CORE STAGES I AND II

Stages I and II are powered, respectively, by dual and single thrust chamber, turbopump-fed, liquid propulsion systems. These systems utilize storable propellants, a 50:50 mixture of hydrazine and unsymmetrical dimethylhydrazine (UDMH) for fuel and nitrogen tetroxide (N_2O_4) for oxidizer.

3. STAGE III

The Stage III propulsion system consists of two pressure-fed engines. Stage III also contains an attitude control system (ACS), which provides attitude control during coast, ullage control prior to main engine burns, and velocity additions for vernier phases and for satellite ejections. The propellants used for the main propulsion system are the same as for Stages I and II. The attitude control system employs hydrazine monopropellant engines.

4. CONTROL MODULE

The control module contains the major elements of the flight control, inertial guidance, electrical, telemetry, tracking, and flight safety systems. The control module is attached to the forward end of Stage III and remains attached throughout flight.

C. <u>MISSION DESCRIPTION (FLIGHT PLAN VII)</u>

The following is a brief mission description of a typical Flight Plan VII. All values given are only approximate, and any discussion of the Flight Plan that does not contribute to understanding the guidance equations has been intentionally omitted. A complete description of a Flight Plan VII mission is given in Refs. 2 and 3. Figure 2 is an illustration of the Flight Plan.

1. ASCENT TO PARKING ORBIT

The vehicle is launched vertically from Air Force Eastern Test Range (ETR) Pad 41 or 40; shortly after liftoff, it is rolled from a pad azimuth of 100.2 deg to a flight azimuth of 93 deg. Following the roll maneuver, an open-loop pitchover, with load relief during the region of max Q, is performed for the remainder of Stage 0 burn. Stages I and II are closed-loop guided;

STAGE	~THRUST (lbs)	~BURN DURATION (sec)	~ $H_P \times H_A$ (nmi)	~INCLINATION (deg)
O	2,000,000	120		
I	500,000	140		
II	100,000	205	100 x 100	28.6
TRAN, A	16,000	300	100 x 19,323	26.4
TRAN, B	16,000	100	19,323 x 19,323	0

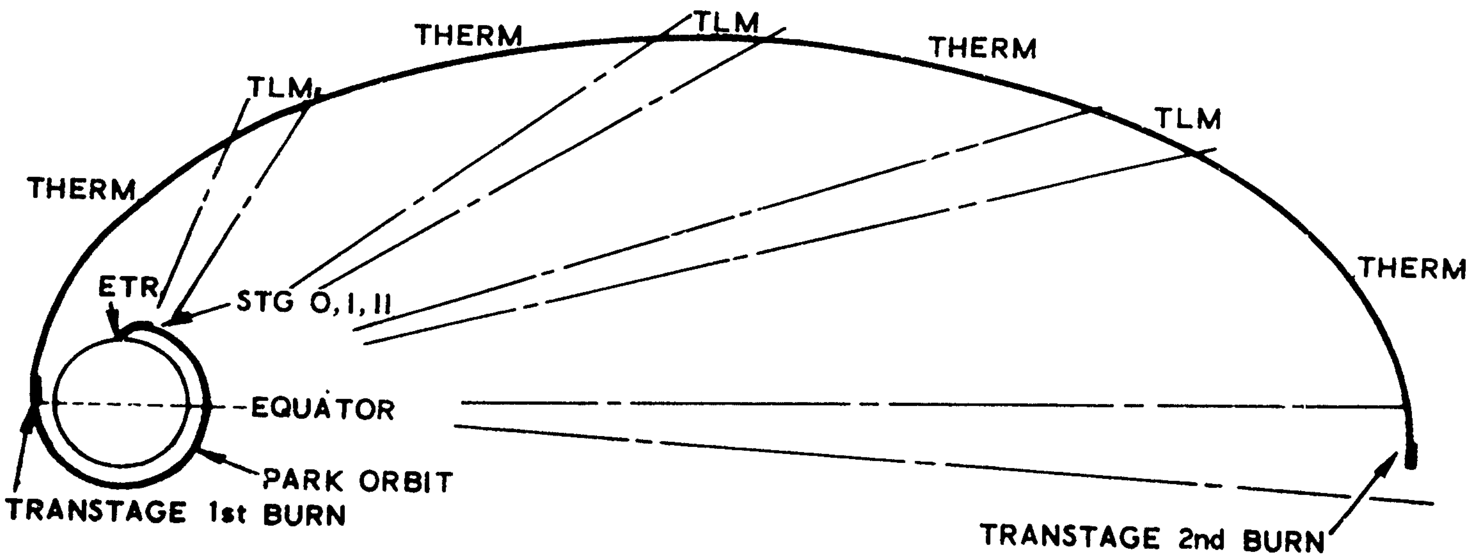


Figure 2. Typical Titan IIIC Mission Profile

the engines burn for approximately 149 and 207 sec, respectively, and the stages burn to propellant exhaustion. At Stage II propellant depletion, the vehicle is nominally in an elliptical orbit 80 × 235 nmi. Further, in the case of a low performing vehicle, the minimum orbit is 80 × 95 nmi.

2. PARKING ORBIT

Immediately after injection into the parking orbit, the vehicle attitude is adjusted to the standard orbital orientation: vehicle longitudinal axis normal to the geocentric radius vector, the nose of the vehicle pointed approximately along the velocity vector, and the vehicle pitch plane coincident with the orbital plane.

After coasting in the parking orbit to near the first descending or ascending node, the vehicle is reoriented in yaw as a startup attitude for the first burn of the transtage. Concurrently, a roll maneuver is performed so that the vehicle telemetry antenna is pointed toward the desired ground tracking station upon completion of the reorientation maneuver.

3. STAGE III FIRST BURN

Close to the equatorial crossing, the Stage III engines are ignited and burn for approximately 300 sec. The vehicle is injected into an elliptical orbit with an apogee of 19,323 nmi. The first burn includes an orbital plane change maneuver of approximately 2.25 deg. If required, the ACS of Stage III can adjust the orbital parameters by a vernier velocity addition. Nominally, shutdown of the Stage III engines is controlled so that a 6-sec ACS vernier phase is required.

4. TRANSFER ORBIT

During the transfer orbit, the transtage performs certain maneuvers to meet thermal control requirements. Currently, three options are being considered for thermal maneuver. The first, called rotisserie, is an oscillatory roll maneuver of approximately ±115 deg at a rate of 1 deg/sec, with a

1-min dwell time at each extreme position. In the second maneuver, toasting, the vehicle is simply turned back and forth in space at widely-spaced time intervals. The third is a continuous roll maneuver between 1 and 2 deg/sec. In all three options, the relationship of the sun vector to the vehicle or spacecraft axes is specified by the payload thermal requirements. In addition to the thermal maneuvers, the vehicle is oriented several times during the transfer orbit to an attitude that permits reliable reception of telemetry.

Finally, shortly before reignition of the Stage III engines, the vehicle is reoriented to a startup attitude for the second burn, which also points the telemetry antenna earthward.

5. FINAL ORBIT INJECTION

The second Stage III burn, of approximately 104-sec duration, injects the payload and vehicle into a circular orbit at 19,323-nmi attitude, with a near zero-deg inclination.

6. PAYLOAD SEPARATION

After the second Stage III shutdown, the vehicle is reoriented for the satellite separation phase. This orientation can vary depending on the selected mission. After sufficient time for stabilization at the desired attitude, the ACS is switched to the payload release coast mode; after sufficient time for stabilization in this mode, the payload is released. At this point, the Titan IIIC mission can be ended with a transtage shutdown sequence. Alternatively, the equations provide an option for performing a short propellant settling burn, another payload release sequence, and, finally, the transtage shutdown sequence.

In both of these payload release options, the equations issue a variety of required discretes and can perform multiple reorientations as specified by payload requirements.

SECTION III

GUIDANCE, STEERING, AND NAVIGATION

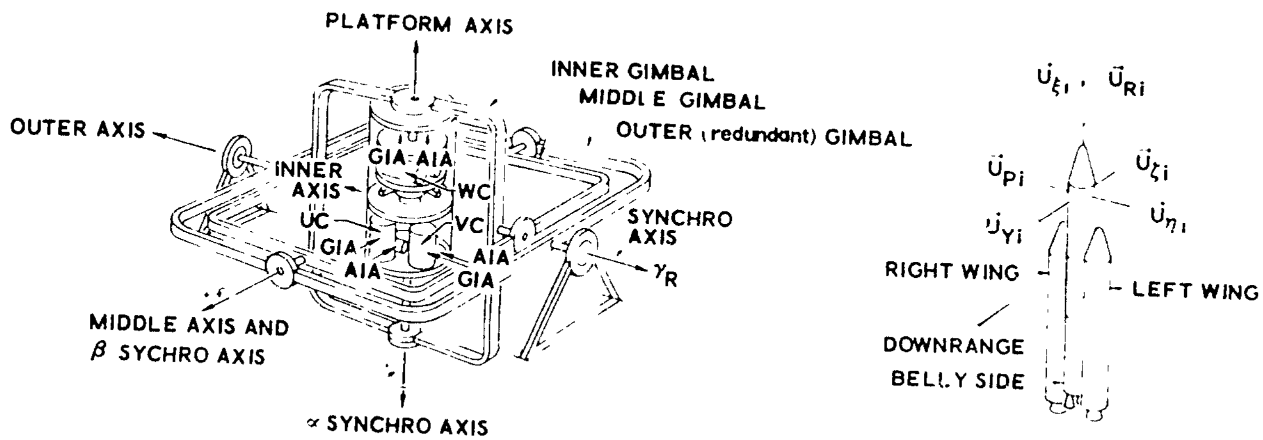
A. GENERAL

The vehicleborne digital computer program is divided into three distinct sections: digital flight control equations, ground equations, and guidance equations. This document treats only the guidance equations; the other programs are mentioned only when they interface with the guidance equations. The carousel sensor and gimbal geometry is given in Fig. 3, while principal computer and sensor characteristics are tabulated in Table 1. Figure 3 and Table 1 are modifications of data taken from Ref. 9.

The guidance equations are organized into various subprograms. A main control or executive program controls logical routing to appropriate calculations, depending on the phase of flight and/or significant flight events. The navigation equations accept inertial measurement unit (IMU) data and compute current vehicle position and velocity. Booster steering equations calculate open-loop steering for Stage 0. Powered flight equations provide vehicle steering as a function of navigational quantities and desired end conditions. Several coast phase subprograms provide computations for desired vehicle attitudes and event initiations during nonpowered flight. Finally, the specific flight plan and the multistage Titan IIIC vehicle necessitate numerous subprograms to repeatedly reinitialize the guidance equations for each phase of flight and to issue discretes to control events such as engine shutdown, engine ignition, and stage separation. The following paragraphs describe the guidance philosophy and overall logic flow of the guidance equations.

B. GUIDANCE

The guidance philosophy governing the guidance equations presented in this document is commonly termed 'explicit'' guidance. Generally, an



AT GO INERTIAL, THE FOLLOWING ORIENTATIONS EXIST;

- THE α ORIENTATION IS ARBITRARY; β AND γ_R SYNCHRO ANGLES ARE SMALL, BUT NOT NECESSARILY ZERO DUE TO ERECTION TOLERANCES.
- ORIENTATION OF THE CAROUSELLING INSTRUMENTS WITH RESPECT TO BOTH THE GIMBALS AND THE TURRET IS ARBITRARY.

AIA AND GIA ARE THE ACCELEROMETER AND GYRO INPUT AXES.

FOR ALL THE FOLLOWING VECTORS, THE i SUBSCRIPTS INDICATE ORIENTATION AT GO INERTIAL:

- $\vec{U}_{\xi i}$, $\vec{U}_{Ri}$ VEHICLE (guidance and control) ROLL AXIS
- $\vec{U}_{\eta i}$ VEHICLE (guidance) PITCH AXIS
- $\vec{U}_{Pi}$ VEHICLE (control) PITCH AXIS
- $\vec{U}_{\zeta i}$ VEHICLE (guidance) YAW AXIS
- $\vec{U}_{Yi}$ VEHICLE (control) YAW AXIS

Figure 3. Carousel VB IMU Geometry

Table 1. Carousel VB IGS Characteristics

Inertial Measurement Unit

- Carouselling platform (1 rpm), X-Y instruments
- 4 gimbal, all attitude
- Synchro attitude readout (sine/cosine)
- AC651G stabilization gyros - SDF, floated
- AC653A accelerometers - force rebalanced

Computer - Magic 352

- Random access memory
- 24-bit word length
- 16-k memory, 2-k erasible
- Fixed point arithmetic
- 57 instructions
  - Add/sub - 6 μs
  - Multiply - 30 μs
  - Divide - 36 μs

explicit guidance scheme is one in which an explicit solution to the powered flight dynamics is computed in flight and the desired end result of guidance is explicitly defined in the guidance equations. The Flight Plan VII guidance equations, depending upon the flight phase, define different desired end conditions. The burns of Core Stages I and II are planned for a specific position magnitude, velocity magnitude, radial velocity, and preplanned orbit planes. The end result of guidance for the first burn of Stage III (transtage) is injection onto a transfer ellipse whose apogee, semi-major axis orientation, and orbital plane are specified. Finally, during the second burn, the transtage guides to a near-circular orbit at the transfer ellipse apogee radius and the orbit plane is specified (approximately equatorial).

References 4, 5, and 6 all contain general discussions of explicit guidance algorithms.

1. THRUST ACCELERATION PREDICTION

A basic requirement of the guidance equations presented here is a prediction of vehicle performance, particularly of thrust acceleration. For this report, "thrust acceleration" is used as an abbreviation for "acceleration (of the vehicle) due to thrust." The following paragraphs derive a time-dependent expression for thrust acceleration.

Consider the variable T_r, defined as

$$T_r = W/\dot{W} \tag{1}$$

where

W - total weight of vehicle at any time
$\dot{W}$ - rate of change of vehicle weight

T_r can be interpreted as the time remaining to zero vehicle weight (mass intercept), if a constant $\dot{W}$ is assumed.

Since specific impulse is

$$I_{sp} = \text{Thrust}/\dot{W} \tag{2}$$

and thrust acceleration is

$$A_{TH} = \text{Thrust}/(W/g) \tag{3}$$

by solving for thrust and weight as

$$\begin{aligned} W &= T_r \dot{W} \\ \text{Thrust} &= \dot{W} I_{sp} \end{aligned} \tag{4}$$

one can express A_{TH} as

$$A_{TH} = gI_{sp}/T_r \tag{5}$$

Further, since the effective exhaust velocity V_e is given as

$$V_e = gI_{sp} \tag{6}$$

then it follows that

$$A_{TH} = V_e/T_r \tag{7}$$

Finally, since T_r is a linear function of time, then A_{TH} as a function of time t can be expressed as

$$A_{TH}(t) = V_e/(T_{r0} - t) \tag{8}$$

where T_{r0} = value of T_r computed at t = 0.

2. PITCH STEERING

In polar coordinates, the two-body equations of motion for a vehicle in a central force field gives the radial acceleration as:

$$R = A_{THr} - (\mu / R^2) + \omega^2 R \tag{9}$$

where

A_{THr} = radial thrust acceleration

$\omega^2 R$ = centrifugal acceleration

$-\mu / R^2$ = acceleration due to gravity

If a time-dependent guidance steering law for the radial thrust pointing direction is assumed to be

$$A_{THr} / A_{TH} = A + Bt + \left[(\mu / R^2 - \omega^2 R) / A_{TH} \right] \tag{10}$$

then, upon substituting for A_{THr}, one finds that Eq. (9) becomes

$$\dot{R} = (A + Bt) A_{TH} \tag{11}$$

This steering law, the familiar "linear sine" law, has been applied to numerous guidance programs. It has the advantages of simplicity and very near fuel optimality for current rocket-propelled vehicles.

Since explicit guidance equations require a prediction of vehicle motion, it is desirable to time-integrate Eq. (11) twice to estimate future vehicle radial velocity and position. However, before these integrations are attempted, Eq. (11) can be simplified to reduce integration complexities.

The linear assumption of T_r can be written as

$$t = T_{r0} - T_r \tag{12}$$

Substituting Eq. (12) into Eq. (11) yields

$$\ddot{R} = (A + BT_{r0} - BT_r)(A_{TH}) \tag{13}$$

And, after some algebraic manipulation,

$$\begin{aligned} \ddot{R} &= -BT_r A_{TH} + (A + BT_{r0})(A_{TH}) \\ &= -BV_e + (A + BT_{r0})(A_{TH}) \end{aligned} \tag{14}$$

In Eq. (14), the quantities A, B, V_e, and T_{r0} are assumed constant; therefore, Eq. (14) can be rewritten as

$$\ddot{R} = A'_r + B_r A_{TH} \tag{15}$$

Time integration of Eq. (15) from a given time ($t = 0$) to cutoff ($t = T_g$) yields

$$\begin{aligned} V_{rf} &= V_r + \int_0^{T_g} A'_r dt + B_r \int_0^{T_g} A_{TH} dt \\ R_f &= R + V_r T_g + \int_0^{T_g} \left[\int_0^t A'_r ds \right] dt + B_r \int_0^{T_g} \left[\int_0^t A_{TH} ds \right] dt \end{aligned} \tag{16}$$

where

V_r = present radial velocity

V_{rf} = radial velocity at cutoff

R = present radial position

R_f = radial position at cutoff

The integrals of thrust acceleration in Eq. (16) can be evaluated using the time-dependent expression for A_{TH} given in Eq. (12):

$$a_{11} = \int_0^{T_g} A_{TH} dt = V_g$$

$$a_{12} = \int_0^{T_g} \left[\int_0^t A_{TH} ds \right] dt = V_g T_g - V_g T_{r0} + V_e T_g \qquad (17)$$

In these integrals, it is assumed that V_g is known. A derivation of the second integral (a_{12}) can be found in practically any discussion of an explicit guidance technique, such as Ref. 4.

To solve for the steering coefficients A'_r and B_r, one must substitute the values for the integrals in Eq. (17) into Eqs. (16):

$$V_{rf} = V_r + A'_r T_g + B_r V_g$$

$$R_f = R + V_r T_g + A'_r (T_g^2/2) + B_r (a_{12}) \qquad (18)$$

Solving for B_r yields

$$B_r = \left[R_f - R - (T_g/2)(V_{rf} + V_r)\right] / \left[-V_g\left[T_{r0} - (T_g/2)\right] + V_e T_g\right] \tag{19}$$

and for A'_r,

$$A'_r = (1/T_g)(V_{rf} - V_r - V_g B_r) \tag{20}$$

Finally, the expression for the desired radial acceleration A_r is obtained by substitution into Eq. (15):

$$A_r = \ddot{R} = (1/T_g)(V_{rf} - V_r - V_g B_r) + B_r A_{TH} \tag{21}$$

3. YAW STEERING

The yaw steering is derived in a manner analogous to the pitch steering scheme. In the yaw case, a desired orbital plane is specified by a vector normal to the plane. Normal error parameters R_n and V_n can be computed as the dot product of vehicle inertial position and velocity with the specified normal vector. Since the desired value of both error parameters is zero at cutoff, the following equations can be used to determine the yaw steering coefficients A_n and B_n:

$$\begin{aligned} 0 &= V_n + A_n T_g + B_n V_g \\ 0 &= R_n + V_n T_g + A_n T_g^2/2 + B_n a_{12} \end{aligned} \tag{22}$$

4. ESTIMATE OF TIME TO GO

The estimate of time to go is derived from ΔM, the vehicle angular momentum to be gained, which is given as

$$\Delta M = M_f - M \tag{23}$$

where

M_f = desired vehicle angular momentum* magnitude at cutoff

M = present vehicle angular momentum* magnitude

Angular momentum to be gained is also given by the integral equation

$$\Delta M = \int_0^{T_g} \dot{M} dt \tag{24}$$

where T_g is the time until rocket engine cutoff.

To estimate time to go, solve the integral Eq. (24) for T_g by a numerical integration technique. The magnitude of angular momentum is given as

$$M = R v_t \tag{25}$$

where

v_t = velocity in the tangential direction

R = position magnitude

---

*Per unit mass

Differentiating Eq. (25) produces

$$\dot{M} = \dot{R}v_t + R\dot{v}_t \tag{26}$$

Another expression based on orbital and tangential thrust acceleration is

$$\dot{v}_t = -v_t\dot{R}/R + A_{THt} \tag{27}$$

where A_{THt} is the vehicle thrust acceleration magnitude in the tangential direction. Substituting the expression for $\dot{v}_t$ into Eq. (26) yields for $\dot{M}$ the simple expression

$$\dot{M} = \dot{R}v_t + R(-v_t\dot{R}/R + A_{THt}) = RA_{THt} \tag{28}$$

The integral of Eq. (24) is evaluated using Simpson's integration formula, with the integrand computed at $t = 0$, $t = T_g$, and three equally-spaced points in between.

The first factor in the integrand, position magnitude, is computed at the five desired time points by a linear interpolation between the present vehicle position magnitude R and the final position magnitude R_f; thus,

$$R(t_j) = R + (j - 1)(R_f - R)/4 \qquad j = 1, \ldots 5 \tag{29}$$

Thrust acceleration magnitude as a function of time has been derived in Paragraph III. B. 1. Thus, it follows that

$$A_{TH}(t_j) = V_e/\left[T_{r0} - (j - 1)T_g'/4\right] \qquad j = 1, \ldots 5 \tag{30}$$

Where T_g' is the time-to-go from the last major cycle decremented by 1 sec

The tangential acceleration A_{TH} is obtained from a square root as

$$A_{THt}(t_j) = \left[A^2_{TH}(t_j) - A^2_{THr}(t_j) - A^2_{THn}(t_j) \right]^{1/2} \tag{31}$$

All that remains is to compute the normal and radial thrust accelerations A_{THn} and A_{THr} at the required time points.

The present radial components of thrust acceleration have already been given as

$$A_{THr} = A'_r + B_r A_{TH} - (-\mu/R^2 + \omega^2 R) \tag{32}$$

and at cutoff as

$$A_{THrf} = A'_r + B_r A_{THf} - (-\mu/R_f^2 + \omega_f^2 R_f) \tag{33}$$

where the subscript f designates final values at cutoff. The centrifugal acceleration terms in Eqs. (32) and (33) can be rewritten, respectively, as

$$\begin{aligned} \omega^2 R &= M^2/R^3 \\ \omega_f^2 R_f &= M_f^2/R_f^3 \end{aligned} \tag{34}$$

Finally, by linearly interpolating the accelerations due to gravity and centrifugal force and by employing the value of acceleration magnitude derived previously in Eq. (30), one obtains for A_{THr} at the required time points

$$\begin{aligned} A_{THr}(t_j) = {} & A'_r + B_r A_{TH}(t_j) - (-\mu/R^2 + M^2/R^3) \\ & - (j-1)\left\{\left[(-\mu/R_f^2 + M_f^2/R_f^3) - (-\mu/R^2 + M^2/R^3)\right]/4\right\} \quad j = 1, \ldots 5 \end{aligned} \tag{35}$$

The normal component of thrust acceleration is given simply as

$$A_{THn}(t_j) = A_n + B_n A_{TH}(t_j) \qquad j = 1, \ldots 5 \tag{36}$$

By combining the results of Eqs. (29) and (31), one obtains the integrand of Eq. (24) at the specified time points

$$\dot{M}(t_j) = R(t_j) A_{THt}(t_j) \tag{37}$$

and, using Simpson's integration rule, one can compute an estimate of ΔM:

$$\Delta M' = [\dot{M}(t_1) + 4\dot{M}(t_2) + 2\dot{M}(t_3) + 4\dot{M}(t_4) + \dot{M}(t_5)] T'_g/12 \tag{38}$$

where $\Delta M'$ is the angular momentum if the initial estimate of time to go is assumed. By comparing ΔM, the true angular momentum to be gained, with $\Delta M'$, one can compute an adjustment to the time to go T_a.

$$T_a = (\Delta M - \Delta M')/\dot{M}(t_5) \tag{39}$$

Finally, the time to go becomes

$$T_g = T'_g + T_a \tag{40}$$

With time to go T_g and the time to mass depletion T_r, the acceleration integral V_g is computed as

$$V_g = \int_0^{T_g} \left| V_e/(T_{r0} - t) \right| dt = -V_e \ln(1 - T_g/T_{r0}) \tag{41}$$

The guidance scheme described here is iterative. The steering coefficient calculations require an estimate of time to go T_g, while the time-to-go calculations presuppose a knowledge of the steering coefficients. Fortunately, experience has shown that with reasonable startup values and appropriate updating between major guidance cycles, the equations described here converge quite rapidly for the mission considered.

5. INTEGRAL CONTROL

In Paragraphs III.B.2 and III.B.3, expressions were derived for desired radial and normal thrust accelerations. If these accelerations were translated directly into vehicle axis pointing-direction commands, a thrust acceleration pointing error would result because of vehicle and IMU properties, such as engine misalignments, vehicle center of gravity effects, and IMU gimbal misalignments. Integral control, the method for overcoming these pointing errors, is used to measure differences between desired and actual guidance-computed accelerations. These differences are numerically time-integrated with an appropriate weighting factor for stability reasons, i.e., a digital filter, and finally subtracted from the desired radial and normal accelerations. The final steady state effect is to bias the vehicle-pointing commands so that the observed misalignments are canceled. Reference 7 contains a complete description of integral control. The equations are mechanized as

$$\Delta A_n \overset{*}{\leftarrow} \Delta A_n + (C_{F6}) \left[(\Delta V_n/\Delta t) - A_n\right] \tag{42}$$

$$\Delta A_r \leftarrow \Delta A_r + (C_{F6}) \left[(\Delta V_r/\Delta t) - A_r\right] \tag{43}$$

---

*The arrow indicates that the results are stored in the original memory location.

where

ΔA_n and ΔA_r - integral control terms

C_{F6} integral weighting factor

A_n = desired normal component of total vehicle acceleration

A_r = desired radial component of total vehicle acceleration

$\Delta V_n/\Delta t$ = guidance-computed normal component of total vehicle acceleration

$\Delta V_r/\Delta t$ guidance-computed radial component of total vehicle acceleration

Δt 1-second major cycle interval

The corrected desired normal and radial equations are then given as

$$A_{nc} = A_n - \Delta A_n \tag{44}$$

$$A_{rc} = A_r - \Delta A_r - A_g - \omega^2 r \tag{45}$$

Equation (45) includes terms for gravity and centrifugal accelerations, these terms were added so that A_{rc}, excluding the effect of ΔA_r, would represent the desired radial thrust acceleration.

Dividing Eqs. (44) and (45) by thrust acceleration magnitude yields the desired unit vehicle longitudinal axis in the radial, normal, and tangential coordinates

$$U_{\xi n} = A_{nc}/A_{TH} \tag{46}$$

$$U_{\xi r} = A_{rc}/A_{TH} \tag{47}$$

$$U_{\xi t} = \left(1 - U_{\xi n}^2 - U_{\xi r}^2\right)^{1/2} \tag{48}$$

Then the unit desired vehicle longitudinal roll axis vector $\bar{U}_\xi$ is converted to inertial coordinates by a matrix multiplication as

$$\begin{bmatrix} U_{\xi x} \\ U_{\xi y} \\ U_{\xi z} \end{bmatrix} = \begin{bmatrix} N \end{bmatrix} \begin{bmatrix} U_{\xi t} \\ U_{\xi n} \\ U_{\xi r} \end{bmatrix} \tag{49}$$

where N is the matrix that relates tangential, normal, and radial coordinates to ECI (x, y, z) coordinates.

The desired unit vehicle pitch axis $\bar{U}_\eta$ is arbitrary from a guidance standpoint and is selected by the guidance equations to conform to vehicle and/or telemetry antenna constraints. Finally, the desired unit yaw axis $\bar{U}_\zeta$ is formed from $\bar{U}_\xi \times U_\eta$ to complete a right-hand orthogonal coordinate system.

6. MANEUVERING EQUATIONS

This section presents a derivation of the maneuvering equations. In this paper, maneuvering is considered the application of rate limits to the guidance commands (desired vehicle axes) and, if the desired maneuver is large, the computation of an efficient maneuver. In addition, the maneuvering equations calculate values to interpolate between successive major cycle commands on a minor cycle basis. The attitude error equations, performed on a minor cycle basis, take the output commands from the maneuvering equations in the form of desired vehicle axis vectors. The desired vehicle axes are compared with the actual present vehicle axes (read from the IMU attitude sensors), and vehicle attitude errors are computed. The flow is from guidance equations to maneuvering equations to attitude error equations.

The initial calculations made by the maneuvering equations are a coordinate conversion from earth centered inertial (ECI) to drifted launch-centered inertial (LCI), as follows:

$$[RG]^T = [\psi]^T [\phi]^T [\psi][CG]$$

$$U_{\xi c} = [RG] [\bar{U}_{\xi}]$$

$$U_{\zeta c} = [RG] [\bar{U}_{\zeta}] \tag{50}$$

$$\bar{U}_{\eta c} = \bar{U}_{\zeta c} \times \bar{U}_{\xi c}$$

where $\bar{U}_{\xi c}$, $\bar{U}_{\eta c}$, and $\bar{U}_{\zeta c}$ are the final desired commanded-vehicle roll, pitch, and yaw axes in drifted LCI coordinates; $[CG]$ is a coordinate transformation matrix from ECI to LCI coordinates; $[\phi]$ is the present compensable IMU drift matrix; and ψ is a matrix chosen to rearrange the rows of ϕ so as to be compatible with the LCI coordinate system.

The drifted LCI coordinate system is used throughout the following equations, since it is the most convenient coordinate system for the guidance minor cycle calculations; the IMU attitude readouts are referenced to the drifted LCI coordinate system. The drifted LCI system (called a, b, g) can also be considered an ideal gimbal angle system.

Following the coordinate transformation, one can calculate the transformation matrix of the desired maneuver:

$$[BG] = \begin{bmatrix} \bar{U}_{\xi c} \cdot \bar{U}_{\xi a} & \bar{U}_{\xi c} \cdot \bar{U}_{\eta a} & \bar{U}_{\xi c} \cdot \bar{U}_{\zeta a} \\ \bar{U}_{\eta c} \cdot \bar{U}_{\xi a} & \bar{U}_{\eta c} \cdot \bar{U}_{\eta a} & \bar{U}_{\eta c} \cdot \bar{U}_{\zeta a} \\ \bar{U}_{\zeta c} \cdot \bar{U}_{\xi a} & \bar{U}_{\zeta c} \cdot \bar{U}_{\eta a} & \bar{U}_{\zeta c} \cdot \bar{U}_{\zeta a} \end{bmatrix} \tag{51}$$

where $\bar{U}_{\xi a}$, $\bar{U}_{\eta a}$, and $\bar{U}_{\zeta a}$ are the present desired commanded-vehicle roll, pitch, and yaw axes. The trace of the BG matrix can be related to the commanded maneuver as

$$\text{Trace} = 1 - 2 \cos \theta \tag{52}$$

or

$$\cos \theta \quad (1 - \text{Trace})/2$$

where θ is the magnitude of the maneuver.

Cos θ can be tested to determine whether to command a rate-limited maneuver. If $\cos \theta$ is sufficiently close to 1.0, the maneuver is small and the following calculations are performed:

$$d\bar{U}_{\xi L} = \bar{U}_{\xi c} - \bar{U}_{\xi a}$$

and (53)

$$d\bar{U}_{\eta L} = \bar{U}_{\eta c} - \bar{U}_{\eta a}$$

The variables $d\bar{U}_{\xi L}$ and $d\bar{U}_{\eta L}$ represent the motion of the commanded-body axes $\bar{U}_{\xi a}$ and $\bar{U}_{\eta a}$ during the next major cycle. If $d\bar{U}_{\xi L}$ and $d\bar{U}_{\eta L}$ are multiplied by the reciprocal of the number of minor cycles per major cycle, C_{10}, the commanded-body axes can be interpolated on a minor cycle basis. With

$$\begin{aligned} \delta \bar{R}_{cm} &= d\bar{U}_{\xi L} \, C_{10} \\ \delta \bar{P}_{cm} &= d\bar{U}_{\eta L} \, C_{10} \end{aligned} \tag{54}$$

computed on a major cycle basis, the following minor cycle calculations yield the interpolated axes:

$$U_{\xi dm} \leftarrow \bar{U}_{\xi dm} + \delta \ddot{R}_{cm}$$
$$U_{\eta dm} \leftarrow \bar{U}_{\eta dm} + \delta \ddot{P}_{cm} \tag{55}$$

If $\cos \theta$ is not close to 1.0, a large maneuver has been commanded and rate-limiting is desired. The maneuvering strategy selected consists of maneuvering about a single inertially fixed axis at a constant angular rate. It can be shown (Euler's theorem) that this single-rotation axis is an eigenvector of the maneuver transformation matrix given in Eq. (51). The following paragraphs describe an algorithm for computing a command "eigenvector" maneuver.

To solve for an eigenvector, $\overline{E}$, of BG, the maneuver transformation matrix, consider a skew-symmetric matrix D, calculated as

$$D = BG - BG^T \tag{56}$$

The D matrix has the properties

$$[D][\overline{E}] = [0] \tag{57}$$

$$D_{ij} = 0 \qquad \text{if } i = j \tag{58}$$

From these properties of the D matrix, the following set of equations is derived:

$$D_{12}E_2 + D_{13}E_3 = 0$$
$$-D_{12}E_1 + D_{23}E_3 = 0 \tag{59}$$
$$-D_{13}E_1 - D_{23}E_2 = 0$$

Solving for the eigenvector in Eqs. (59) yields

$$
\begin{aligned}
E_1 &= D_{23} = BG_{23} - BG_{32} \\
E_2 &= -D_{13} = BG_{31} - BG_{13} \\
E_3 &= D_{12} = BG_{12} - BG_{21}
\end{aligned}
\tag{60}
$$

After unitization, the eigenvector is transformed from body coordinates to the LCI coordinate system with the following matrix multiplication:

$$
\overline{RL} = [A] \begin{bmatrix} UE_1 \\ UE_2 \\ UE_3 \end{bmatrix}
\tag{61}
$$

where

$$
\left[A\right] = \left[\overline{U}_{\xi a}, \; \overline{U}_{\eta a}, \; \overline{U}_{\zeta a}\right] \text{ the body axes as columns.}
$$

The magnitude of the eigenvector computed in Eq. (60) is proportional to the sine of the maneuver angle θ. There is a singularity wherein the magnitude of the eigenvector approaches zero when the commanded maneuver approaches 180 deg. When $\cos \theta$ is sufficiently close to -1.0, the $\overline{RL}$ vector is replaced by a unit-commanded body axis according to the following strategy.

The largest diagonal element of the BG matrix is determined.

Then, if

$$
\begin{aligned}
BG_{11} \text{ largest}, \ \overline{RL} &= \overline{U}_{\xi a} \\
BG_{22} \text{ largest}, \ \overline{RL} &= \overline{U}_{\eta a} \\
BG_{33} \text{ largest}, \ \overline{RL} &= \overline{U}_{\zeta a}
\end{aligned}
\tag{62}
$$

The above procedure ensures that the initial maneuver away from the 180-deg region is not made about an axis which is perpendicular to the desired maneuver axis.

$\overline{RL}$ is considered the unit command rotation vector, and the variables $d\overline{U}_{\xi L}$ and $d\overline{U}_{\eta L}$ are computed as

$$
\begin{aligned}
d\overline{U}_{\xi L} &= \left(\overline{RL} \times \overline{U}_{\xi a}\right) \text{MLIM} \\
d\overline{U}_{\eta L} &= \left(\overline{RL} \times \overline{U}_{\eta a}\right) \text{MLIM}
\end{aligned}
\tag{63}
$$

and the interpolated body axes, on a minor cycle basis, are computed as described in Eqs. (54) and (55). MLIM is the magnitude of the rate limit.

The method used to interpolate the commanded-body vector throughout the major cycle results in a very small nonorthonormality in the commanded vehicle axes. To prevent this nonorthonormality from growing, one can perform an orthonormalizing process for each major cycle, as follows:

$$
\begin{aligned}
\overline{U}_{\xi a} &= (\overline{U}_{\xi a} + d\overline{U}_{\xi L}) / |\overline{U}_{\xi a} + d\overline{U}_{\xi L}| \\
\overline{U}_{\zeta a} &= \overline{U}_{\xi a} \times (\overline{U}_{\eta a} + d\overline{U}_{\eta L}) / |\overline{U}_{\xi a} \times (\overline{U}_{\eta a} + d\overline{U}_{\eta L})| \\
\overline{U}_{\eta a} &= \overline{U}_{\xi a} \times \overline{U}_{\eta a}
\end{aligned}
\tag{64}
$$

This process resets the command-vehicle coordinates to orthonormality every major cycle.

C. NAVIGATION

The guidance navigation calculations compute the present vehicle position and velocity using as inputs accumulated incremental velocity pulses from a triad of force rebalance integrating accelerometers. One of the accelerometers (WC) is mounted on the IMU stable platform (turret); the other two (UC and VC) are mounted on a platform that rotates, or carousels, at 1 rpm. Figure 4 is a block diagram of the major and minor cycle navigation.

1. ACCELEROMETER RESOLUTION AND COMPENSATION

For the navigation function to be performed, the initial azimuth (at "go inertial") of the rotating platform must be determined. Further, whenever the incremental velocities are sampled, this azimuth must be updated.

Four coordinate systems are used in the accelerometer resolution and compensation:

[uc, vc, wc]	Actual carouselling accelerometer input axes coordinate system. The coordinate system is, in general, nonorthogonal and rotating due to IMU drift and carouselling.
$[\bar{1}, \bar{2}, \bar{3}]$	Ideal carouselling coordinate system. This coordinate system of convenience is defined as follows. $\bar{3}$ is colinear with the carouselling axis of rotation. $\bar{2}$ is perpendicular to $\bar{3}$; it is in the plane of $\bar{3}$ and the vc accelerometer input axis. $\bar{1}$ completes the orthogonal right-hand set.
[u, v, w]	Drifted launch site coordinate system. Initially equal to the initial vehicle roll, pitch, yaw coordinate system. After go inertial, related to the initial vehicle coordinate system by the IMU drift matrix.
[x, y, z]	Earth centered inertial coordinate system. Defined as follows: $\bar{x}$ = unit vector in the equatorial and prime meridian planes at go-inertial time. $\bar{z}$ = unit earth spin vector, i.e., North Pole $\bar{y}$ = $\bar{z} \times \bar{x}$ to complete a right-hand set.

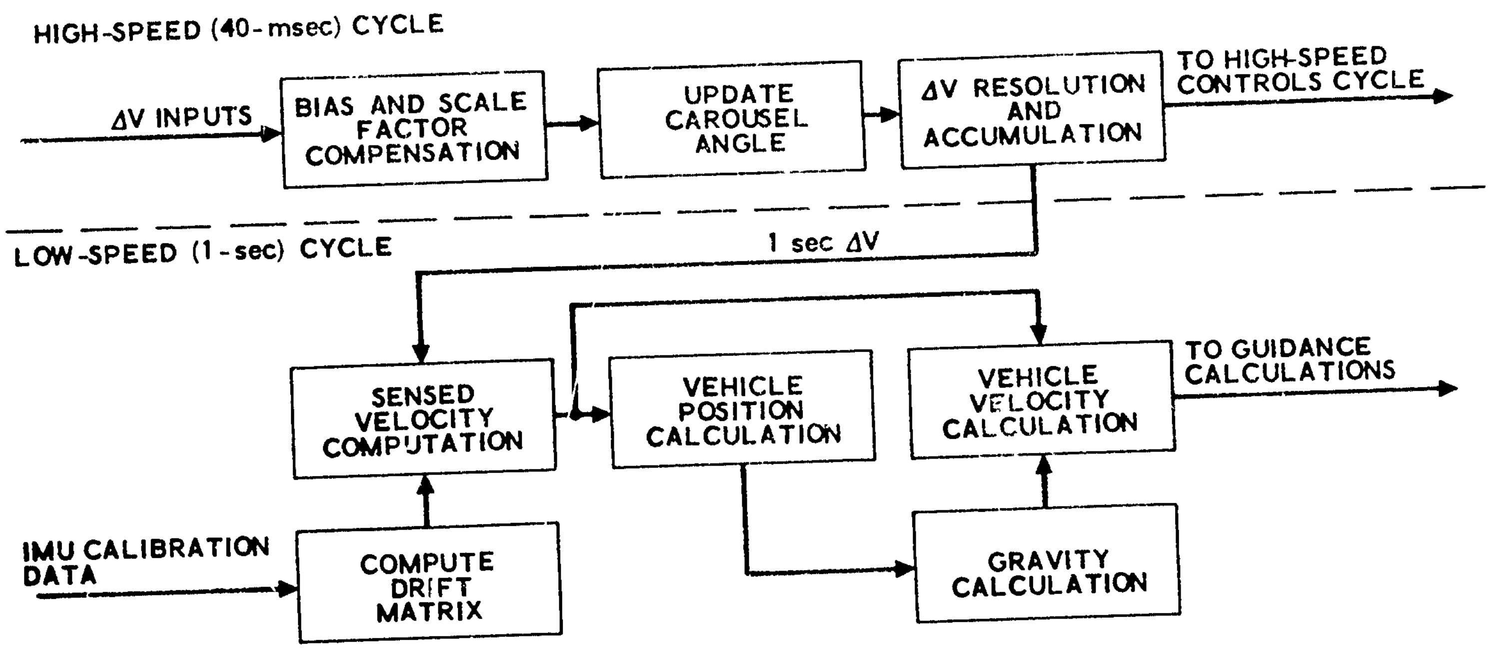


Figure 4. Navigation Block Diagram

An initial azimuth, or phase angle, is computed in the ground-to-flight interface program as follows:

$$FANG = HTHYN + CFANG + CZZANG \tag{65}$$

where

HTHYN = the angle, computed in the ground alignment program, from north to the VC accelerometer input axis plus gyro misalignment errors

CZZANG = gyro misalignment IMU compensation parameter

CFANG = parameter to convert the phase angle from northeast to downrange-crossrange coordinates and to time synchronize with the accelerometer readouts.

The sine and cosine of the initial phase angle are then taken for resolution purposes.

$$\begin{aligned} SS &= \sin(FANG) \\ BS &= \cos(FANG) \end{aligned} \tag{66}$$

After initialization, the carousel phase angle sine and cosine is updated every 40 msec by a trigonometric identity as follows:

$$\begin{aligned} SS_{n+1} &= (SS_n)(\cos 0.24^\circ) + (BS_n)(\sin 0.24^\circ) \\ BS_{n+1} &= (BS_n)(\cos 0.24^\circ) - (SS_n)(\sin 0.24^\circ) \end{aligned} \tag{67}$$

where 0.24 deg is the angle "carousselled" through in 40 msec. Also, the carousel phase angle is reinitialized to SS and BS at 1-minute intervals.

The accelerometer incremental counts are modified with the calibration constants determined as part of factory calibration and pad test procedures.

The 40-msec frequency accelerometer compensation equations are given as

$$\begin{aligned}
\Delta V_{uc} &= (CKAX)(\Delta N_{uc} - CKBXG) + (CKNX)(\Delta N_{uc})^2 \\
\Delta V_{vc} &= (CKAY)(\Delta N_{vc} - CKBYG) + (CKNY)(\Delta N_{vc})^2 \\
\Delta V_{wc} &= (HZAF)(\Delta N_{wc} - HBZ) + (CKNZ)(\Delta N_{wc})^2
\end{aligned} \tag{68}$$

where

ΔN_{uc}, ΔN_{vc}, ΔN_{wc} = the raw counts from the carouselling uc and vc and the stationary wc accelerometers over the last 40-msec interval

CKBXG, CKBYG = the uc and vc accelerometer factory bias calibration

HBZ = the wc accelerometer bias calibration computed during final align

CKAX, CKAY = the uc and vc factory scale factor calibration

HZAF = the wc accelerometer scale factor calibration computed during final align

CKNX, CKNY, CKNZ = the uc, vc, and wc nonlinearity calibration

Continuing, one finds that

$$\begin{aligned}
ST1 &= \Delta V_{uc} - (CZZB2U)(\Delta V_{vc}) - (CZZB3U)(\Delta V_{wc}) \\
ST2 &= \Delta V_{vc} - (CZZB3V)(\Delta V_{wc})
\end{aligned} \tag{69}$$

Equation (69) shows small angle approximations to misalignment rotation compensation where

ST1, ST2 = velocity increments in $\overline{1}, \overline{2}, \overline{3}$ coordinates

CZZB2U = misalignment of the u axis toward $\overline{2}$

CZZB3U = misalignment of the u axis toward $\overline{3}$

CZZB3V = misalignment of the v axis toward $\overline{3}$

ST1 and ST2 are then resolved into u, v, w coordinates as follows:

$$\Delta V_{um} = (BS)(ST1) - (SS)(ST2)$$
$$\Delta V_{vm} = (SS)(ST1) + (BS)(ST2) \quad (70)$$

The resolved minor cycle velocity increments are finally accumulated as follows:

$$\Delta V_{up} \leftarrow \Delta V_{up} + \Delta V_{um}$$
$$\Delta V_{vp} \leftarrow \Delta V_{vp} + \Delta V_{vm} \quad (71)$$
$$\Delta V_{wap} \leftarrow \Delta V_{wap} + \Delta V_{wc}$$

Once per second, the accumulated velocity increments are sampled for use as inputs to the major cycle navigation equations

$$\Delta V_{u} \leftarrow \Delta V_{up}$$
$$\Delta V_{v} \leftarrow \Delta V_{vp} \quad (72)$$
$$\Delta V_{wa} \leftarrow \Delta V_{wap}$$

The variables ΔV_{up}, ΔV_{vp} and ΔV_{wap} are then zeroed in preparation for further accumulation in the next major cycle interval.

$$\Delta V_{up} \leftarrow 0$$
$$\Delta V_{vp} \leftarrow 0 \quad (73)$$
$$\Delta V_{wap} \leftarrow 0$$

The w-accelerometer misalignments are performed on a major cycle basis, since the w instruments do not carousel. These misalignments do rotate, because the w-accelerometer is rotating at earth rate at liftoff. During guidance initialization, misalignments are computed in the u, v, w coordinate system as a function of the initial gimbal angle

$$
\begin{aligned}
BW1 &= (CKB4W)(\sin \alpha_i) + (CKB5W)(\cos \alpha_i) \\
BW2 &= -(CKB4W)(\cos \alpha_i) + (CKB5W)(\sin \alpha_i)
\end{aligned}
\tag{74}
$$

where

BW1 = w-accelerometer misalignment away from u axis

BW2 = w-accelerometer misalignment away from v axis

CKB4W, CKB5W = factory-calibrated w-accelerometer misalignments in an α gimbal-oriented coordinate system

The w-accelerometer input axis also "cones" after go inertial as a function of the inner gimbal angle, because of misalignment of the inner gimbal axis and the carouselling axis. The complete equation for w-accelerometer misalignment compensation is given in Eq. (75):

$$
\begin{aligned}
\Delta V_w = \Delta V_{wa} &+ [\,BW1 + (CKBSW)(\cos \alpha) - (CKBCW)(\sin \alpha)\,]\Delta V_u \\
&+ [\,BW2 - (CKBCW)(\cos \alpha) - (CKBSW)(\sin \alpha)\,]\Delta V
\end{aligned}
\tag{75}
$$

where

CKBSW, CKBCW = calibrated carouselling axis misalignments in an α gimbal angle coordinate system

2. GYRO DRIFT COMPENSATION

Gyro drift compensation is accomplished by computing the platform drift rate with calibration values obtained during hangar and pad tests. Then, by time-integration of the drift rate, a matrix ϕ can be computed that transforms the present (u, v, w) axes to the initial nondrifting axes. Further matrix operations then transform to the desired ECI coordinate system.

$$\begin{bmatrix} \Delta V_{sx} \\ \Delta V_{sy} \\ \Delta V_{sz} \end{bmatrix} = [CG]^T [\psi]^T [\phi] \begin{bmatrix} \Delta V_u \\ \Delta V_v \\ \Delta V_w \end{bmatrix} \tag{76}$$

where

$\Delta V_{sx, sy, sz}$ = measured (sensed) values of velocity increments in the ECI coordinate system

$\Delta V_{u, v, w}$ measured value of velocity increments along the present drifted launch site coordinate system

ϕ = drift matrix

ψ row rearrangement matrix

CG = coordinate transformation matrix from LCI to ECI coordinates

Note that ϕ, ψ and CG are shared by guidance (Paragraph III.B.6) and navigation.

If the instantaneous drift rate vector is given as ($\dot{\phi}_u$, $\dot{\phi}_v$, $\dot{\phi}_w$), the derivative of each column of the ϕ matrix is the cross-product of the drift rate vector with that column vector. Calculating the cross products and combining terms, one obtains the matrix form

$$\dot{\phi} = [\phi] \begin{bmatrix} 0 & -\dot{\phi}_w & \dot{\phi}_v \\ \dot{\phi}_w & 0 & -\dot{\phi}_u \\ -\dot{\phi}_v & \dot{\phi}_u & 0 \end{bmatrix} \tag{77}$$

Integration of $\dot{\phi}$ is accomplished by using a second order Runge-Kutta algorithm. First, the ϕ matrix is integrated a half cycle with an initial derivative:

$$\tilde{\phi}_n = \phi_n + (\Delta t/2)\dot{\phi}_n \tag{78}$$

The drift rate is then averaged over two cycles:

$$\begin{aligned}
\Delta\tilde{\phi}_u &= (\Delta\phi_{un} + \Delta\phi_{un+1})/2 \\
\Delta\tilde{\phi}_v &= (\Delta\phi_{vn} + \Delta\phi_{vn+1})/2 \\
\Delta\tilde{\phi}_w &= (\Delta\phi_{wn} + \Delta\phi_{wn+1})/2
\end{aligned} \tag{79}$$

A final step advances ϕ as follows:

$$\phi_{n+1} = \phi_n + [\tilde{\phi}_n] \quad \begin{bmatrix} 0 & -\Delta\tilde{\phi}_w & \Delta\tilde{\phi}_v \\ \Delta\tilde{\phi}_w & 0 & -\Delta\tilde{\phi}_u \\ -\Delta\tilde{\phi}_v & \Delta\tilde{\phi}_u & 0 \end{bmatrix} \tag{80}$$

Computation of the drift vector $\dot{\phi}$ in u, v, w coordinates is complicated by the rotation of the carouselling instruments. Drifts of each gyro must be located spatially due to the arbitrary location of the IMU turret and the rotation of the platform at 1 rpm.

All drift matrix calculations are performed once per second. A central assumption for the drift calculations is that the drift occurred approximately centered in the 1 sec compute cycle. To locate the carouselling instruments, one must compute backed-up values of the variables BS and SS.

$$SSB = -(BS)(\sin 3°) + (SS)(\cos 3°)$$
$$BSB = (BS)(\cos 3°) + (SS)(\sin 3°) \tag{81}$$

The variables SSB and BSB are then used to resolve ΔV_u and ΔV_v into coordinates representing the "average" position of the carouselling gyros over the last second.

$$\Delta V_{g1} = (\Delta V_u)(BSB) + (\Delta V_v)(SSB)$$
$$\Delta V_{g2} = -(\Delta V_u)(SSB) + (\Delta V_v)(BSB) \tag{82}$$

The fixed-torque and unbalance drifts of the carouselling gyros are then computed as

$$DRFTU = FTDU + (CKU1)(\Delta V_{g1}) + (CKU2)(\Delta V_{g2})$$
$$DRFTV = FTDV + (CKU3)(\Delta V_{g2}) + (CKU4)(\Delta V_{g1}) \tag{83}$$

where

FTDU, V = fixed-torque drifts

CKU1, 3 = spin-axis unbalance drifts

CKU2, 4 = input-axis unbalance drifts

In addition to the usual error sources associated with an IMU, carouselling itself introduces gyro drifts. The compensable ones are called gimbal-oriented bias (GOB), turret-oriented bias (TOB), and turret-oriented eta (TOE). GOB is a drift in the carouselling plane, which rotates as the inner gimbal rotates; i.e., a drift fixed to the inner gimbal. Initially, GOB is computed in launch-centered coordinates as

$$GOBC1 = (CK20)(\cos CK21)$$

$$GOBC2 = (CK20)(\sin CK21) \tag{84}$$

where

CK20 = GOB magnitude

CK21 = GOB phase angle

Later, in flight, the computation of GOB reflects the rotation about the inner gimbal as follows:

$$GOB1 = -(GOB\ C2)(\cos \alpha) + (GOB\ C1)(\sin \alpha)$$

$$GOB2 = (GOB\ C1)(\cos \alpha) + (GOB\ C2)(\sin \alpha) \tag{85}$$

Turret-oriented bias is a drift of the turret (perpendicular to the carouselling plane) that is a function of several harmonics of the turret present position with respect to the inner gimbal. To locate the turret during guidance initialization, one must compute the angle of the turret revolution from a zero inner gimbal angle value:

$$\alpha_i = \tan^{-1}(\sin \alpha_i / \cos \alpha_i) \tag{86}$$

where

α_i = initial (at go inertial) inner gimbal angle

$\sin \alpha_i$ = sine of initial α

$\cos \alpha_i$ = cosine of initial α

Then, in flight, this angle is updated as

$$\alpha_t = \tan^{-1}(\sin \alpha/\cos \alpha) + \alpha_i \tag{87}$$

The equation for TOB is

$$\mathrm{TOB} = \sum_{i=1}^{4} K_i [\cos (n_i \alpha_t - \theta_i)] \tag{88}$$

In Eq. (88), n_i represents integers specifying the harmonics of α_t to be included in the TOB calculation, θ_i represents the phase angle of each harmonic; and K_i represents magnitudes.

TOE is a drift in the carouselling plane that is a function of the location of the turret with respect to the launch-centered coordinate system. It is computed at guidance initialization as

$$\begin{aligned} \mathrm{TOE1} &= -(\mathrm{CK22})[(\sin \alpha_i)(\cos \mathrm{CK23}) + (\cos \alpha_i)(\sin \mathrm{CK23})] \\ \mathrm{TOE2} &= (\mathrm{CK22})[(\cos \alpha_i)(\cos \mathrm{CK23}) - (\sin \alpha_i)(\sin \mathrm{CK23})] \end{aligned} \tag{89}$$

where

CK22 - TOE bias drift magnitude

CK23 - TOE bias drift phase angle

Finally, the derivative of ϕ is computed as shown in Eq. (90):

$$\begin{aligned} \Delta\phi_u &= (\mathrm{BSB})(\mathrm{DRFTU}) - (\mathrm{SSB})(\mathrm{DRFTV}) + \mathrm{GOB1} + \mathrm{TOE1} \\ \Delta\phi_v &= (\mathrm{SSB})(\mathrm{DRFTU}) + (\mathrm{BSB})(\mathrm{DRFTV}) + \mathrm{GOB2} + \mathrm{TOE2} \\ \Delta\phi_w &= \mathrm{HRZ} + (\mathrm{DIAU})(\Delta V_w) - (\mathrm{CZZU6})[(\Delta V_v)(\sin \alpha_i) + (\Delta V_u)(\cos \alpha_i)] + \mathrm{TOB} \end{aligned} \tag{90}$$

where

BSB, SSB = sine and cosine of the average carousel angle during last compute cycle

DRFTU, DRFTV = fixed-torque and g-sensitive drifts of the UC and VC carouselling gyros

GOB1, GOB2 = GOB drift [Eq. (84)]

TOE1, TOE2 = TOE drift [Eq. (89)]

TOB = TOB drift [Eq. (88)]

HRZ = noncarouselling Z gyro fixed-torque drift

DIAU = spin-axis unbalance drift

CZZU6 = input-axis unbalance drift, Z gyro

3. TRANSFORMATION OF VELOCITY TO INERTIAL x,y,z COORDINATE SYSTEM

The inertial x, y, z coordinate system is defined as follows:

$\overline{x}$ = unit vector in the equatorial and prime meridian planes at go inertial time

$\overline{z}$ = unit earth spin vector, i.e., North Pole

$\overline{y} = \overline{z} \times \overline{x}$ to complete a right-hand set

4. VEHICLE INERTIAL POSITION

The following trapezoidal integration formulas are used to obtain the vehicle position (X, Y, Z) in the inertial coordinate system:

$$
\begin{aligned}
X &\leftarrow X + V_x(\Delta t) + (1/2)(\Delta t)^2 (\Delta V_{sx} + \Delta V_{gx}) \\
Y &\leftarrow Y + V_y(\Delta t) + (1/2)(\Delta t)^2 (\Delta V_{sy} + \Delta V_{gy}) \qquad (91) \\
Z &\leftarrow Z + V_z(\Delta t) + (1/2)(\Delta t)^2 (\Delta V_{sz} + \Delta V_{gz})
\end{aligned}
$$

where

$$V_x, V_y, V_z = \text{x, y, z components of vehicle velocity}$$

$$\Delta t = \text{major compute cycle interval} = 1 \text{ second}$$

$\Delta V_{sx}, \Delta V_{sy}, \Delta V_{sz}$ = x, y, z components of sensed vehicle velocity increment from Eq. (76)

$\Delta V_{gx}, \Delta V_{gy}, \Delta V_{gz}$ = x, y, z components of velocity increments due to gravity

Vehicle position magnitude is then calculated as

$$R = (X^2 + Y^2 + Z^2)^{1/2} \tag{92}$$

5. GRAVITY COMPUTATIONS

An approximation for the gravitational potential of the earth is

$$U = (\mu / R)[1 + (Ja^2/3R^2)(1 - 3 \sin^2 \lambda)] \tag{93}$$

where

μ = gravitational parameter of the earth

R = distance from center of the earth

λ = latitude

a = equatorial radius of the earth

J = a characteristic constant that is a function of the moments of inertia with respect to the polar and equatorial axes

A derivation of Eq. (93) is contained in Ref. 7. This approximation, which is a truncated series, contributes negligible inaccuracies to the overall navigation function.

Then, from the definition of the Z component of vehicle position, it follows that

$$\sin^2 \lambda = Z^2/R^2 \tag{94}$$

which, upon substitution into Eq. (93), yields

$$U = (\mu/R)(1 + Ja^2/3R^2 - Ja^2Z^2/R^4) \tag{95}$$

The x, y, z components of the acceleration due to gravity are then obtained by the partial differentiation of Eq. (95) with respect to each axis, and

$$\begin{aligned} g_x &= -\partial U/\partial x = -(\partial U/\partial R)(\partial R/\partial x) \\ g_y &= -\partial U/\partial y = (-\partial U/\partial R)(\partial R/\partial y) \\ g_z &= -\partial U/\partial z = (-\partial U/\partial R)(\partial R/\partial z) \end{aligned} \tag{96}$$

The negative sign is added by convention.

An evaluation of the common term $\partial U/\partial R$ yields

$$\partial U/\partial R = (-\mu/R^2)(1 + Ja^2/R^2 - 5Ja^2Z^2/R^4) \tag{97}$$

and, by factoring out $A_g = -(\mu/R^2)$, one obtains

$$\partial U/\partial R = A_g(1 - Ja^2A_g/\mu + 5Ja^2A_gZ^2/\mu R^2) \tag{98}$$

In addition, one finds that

$$
\begin{aligned}
\partial R/\partial x &= X/R \\
\partial R/\partial y &= Y/R \\
\partial R/\partial z &= Z/R
\end{aligned}
\qquad (99)
$$

and

$$\partial U/\partial z = 2\mu J a^2 Z/R^5 = +2 J a^2 A_g^2 Z/\mu R \qquad (100)$$

The velocity increments over a major compute cycle due to gravitational acceleration are given in Eq. (101).

$$
\begin{aligned}
\Delta V_{gx} &= g_x = -(\partial U/\partial R)(\partial R/\partial x) \\
\Delta V_{gy} &= g_y = -(\partial U/\partial R)(\partial R/\partial x) \\
\Delta V_{gz} &= g_z = -(\partial U/\partial R)(\partial R/\partial z)
\end{aligned}
\qquad (101)
$$

6. VEHICLE INERTIAL VELOCITY

The inertial velocity of the vehicle is computed as

$$(V_i)_{n+1} = (V_i)_n + (1/2)[(\Delta V_{gi})_{n+1} + (\Delta V_{gi})_n]\,[\Delta t] + (\Delta V_{si})(\Delta t) \qquad (102)$$

$$i = x, y, z$$

Averaging ΔV_g over two cycles produces a trapezoidal integration of acceleration due to gravity.

D. ATTITUDE ERRORS

Attitude errors are crucial outputs of the guidance equations utilized by the Digital Flight Controls Equation in stabilizing the vehicle to a desired attitude. They represent differences between the present actual vehicle attitude and the present desired vehicle attitude. The desired attitude is computed by the guidance maneuvering equations (Paragraph III. B. 6); the actual attitude is derived from synchros attached to each gimbal.

1. SYNCHROS

The synchro signals supplied to the inflight computer are of the form

$$V1 = K \sin(\theta - 120°) \text{ and } V2 = K \sin(\theta - 60°) \tag{103}$$

where θ can be any of three gimbal angles, α, β, or γ_R, and K is a scaling factor common to V1 and V2.

By trigonometric identities, one obtains

$$\begin{aligned} V1 - V2 &= K \sin(\theta - 120°) - K \sin(\theta - 60°) \\ &= K\left[(\sin\theta \cos 120° - \cos\theta \sin 120°) - (\sin\theta \cos 60° - \cos\theta \sin 60°)\right] \\ &= K \sin\theta (\cos 120° - \cos 60°) \\ &= -K \sin\theta \end{aligned} \tag{104}$$

and

$$
\begin{aligned}
V1 + V2 &= K \sin (\theta - 120^\circ) + K \sin (\theta - 60^\circ) \\
&= K \left[(\sin \theta \cos 120^\circ - \cos \theta \sin 120^\circ) \right. \\
&\quad \left. + (\sin \theta \cos 60^\circ - \cos \theta \sin 60^\circ) \right] \\
&= K \cos \theta (- \sin 120^\circ - \sin 60^\circ) \\
&= -K\sqrt{3} \cos \theta
\end{aligned}
\tag{105}
$$

Thus, by simple sum and difference, one obtains the sine and cosine of the gimbal angles, except for a common multiplier -K and the constant $\sqrt{3}$.

Since the sine and cosine of the gimbal angles are desired on a minor cycle basis as a convenience for the computation of vehicle attitude errors, it is necessary to determine the scaling factor K.

At go inertial, the following initial calibration is performed for each of the three gimbal angles:

$$
\begin{aligned}
X &= V2 - V1 \\
Y &= (V1 + V2)(C1) \\
K^2 &= (X)(X) + (Y)(Y) \\
K &= \sqrt{K^2}
\end{aligned}
\tag{106}
$$

In this series of equations, C1 is a constant equal to $-1.0/\sqrt{3}$, and K is the calibrated value of the synchro scale factor.

On succeeding minor cycles, the sine and cosine of each gimbal angle are computed as follows:

$$
\begin{aligned}
X_m &= V2 - V1 \\
Y_m &= (V1 + V2)(C1) \\
K^2_m &= (X_m)(X_m) + (Y_m)(Y_m) \\
K_m &= (1/2)\ (K^2_m / K_m + K_m) \\
\sin\theta &= X/K_m \\
\cos\theta &= Y/K_m
\end{aligned}
\tag{107}
$$

These minor cycle computations use Newton's square-root algorithms, which converge quite rapidly if the initial value is relatively accurate.

Since the three gimbals, α, β, and γ_R, are not necessarily zero at go inertial, they are read initially; the initial values are used to compute the three angles through which the vehicle rotates during the flight. Taking α as an example, trigonometric identities yield

$$
\begin{aligned}
\sin\alpha &= \sin\alpha_\rho \cos\alpha_i - \cos\alpha_\rho \sin\alpha_i \\
\cos\alpha &= \cos\alpha_\rho \cos\alpha_i + \sin\alpha_\rho \sin\alpha_i
\end{aligned}
\tag{108}
$$

where

$\sin\alpha$, $\cos\alpha$ = sine and cosine of vehicle angular rotation

$\sin\alpha_\rho$, $\cos\alpha_\rho$ = present gimbal synchro sine and cosine readings

$\sin\alpha_i$, $\cos\alpha_i$ = initial gimbal synchro sine and cosine readings

An identical procedure is followed to compute the sine and cosine of the β and γ_R vehicle rotations.

The fourth gimbal γ has very limited travel (< 10 deg), and the following simplifying approximations are made:

$$
\begin{aligned}
\cos \gamma &= 1.0 \\
\sin \gamma &= (\text{CKDSCA})(\sin \gamma_\rho) - \sin \gamma_i
\end{aligned}
\qquad (109)
$$

where CKDSCA = calibrated scale factor for $\sin \gamma_\rho$

2. ATTITUDE ERROR COMPUTATIONS

The IMU gimbal angles (α, β, γ and γ_R) ideally establish the relationship between two vehicle coordinate systems: the present roll, pitch, and yaw (ξ, η, ζ) body axes; and the initial body axes (ξ_i, η_i, ζ_i). Initial alignment is such that, at launch, a positive roll results in a negative gimbal angle α; a positive yaw results in a positive gimbal angle β; and a positive pitch results in a positive gimbal angle γ_R. The fourth gimbal γ, in normal circumstances, is zeroed. Its sign convention is the same as γ_R

$$
\left\{\bar{U}_\xi, \bar{U}_\eta, \bar{U}_\zeta\right\} \begin{bmatrix} \cos\gamma_R & 0 & -\sin\gamma_R \\ 0 & 1 & 0 \\ \sin\gamma_R & 0 & \cos\gamma_R \end{bmatrix} \begin{bmatrix} \cos\beta & \sin\beta & 0 \\ -\sin\beta & \cos\beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\gamma & 0 & -\sin\gamma \\ 0 & 1 & 0 \\ \sin\gamma & 0 & \cos\gamma \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \left\{\bar{U}_{\xi_i}, \bar{U}_{\eta_i}, \bar{U}_{\zeta_i}\right\} \qquad (110)
$$

Performing the matrix multiplication with $\cos \gamma = 1.0$ yields

$$
M = \begin{bmatrix}
(\cos\gamma_R \cos\beta - \sin\gamma_R \sin\gamma) & (\cos\gamma_R \sin\beta\cos\alpha & (-\cos\gamma_R \sin\beta\sin\alpha \\
 & -\cos\gamma_R\cos\beta\sin\gamma\sin\alpha & -\cos\gamma_R\cos\beta\sin\gamma\cos\alpha \\
 & -\sin\gamma_R\sin\alpha) & -\sin\gamma_R\cos\alpha) \\
(-\sin\beta) & (\cos\beta\cos\alpha + & (-\cos\beta\sin\alpha + \sin\beta\sin\gamma\cos\alpha) \\
 & \sin\beta\sin\gamma\sin\alpha & \\
(\sin\gamma_R\cos\beta + \cos\gamma_R\sin\gamma & (\sin\gamma_R\sin\beta\cos\alpha & (-\sin\gamma_R\sin\beta\sin\alpha \\
 & -\sin\gamma_R\cos\beta\sin\gamma\sin\alpha & -\sin\gamma_R\cos\beta\sin\gamma\cos\alpha \\
 & +\cos\gamma_R\sin\alpha) & +\cos\gamma_R\cos\alpha)
\end{bmatrix} \tag{111}
$$

or $M = [\bar{U}_\xi, \bar{U}_\eta, \bar{U}_\zeta]$

If the matrix of initial vehicle axes is assumed to be identity, the rows of the M matrix comprise the present measured vehicle axes in a gimbal-oriented coordinate system (a, b, g), where initially

$$
\begin{aligned}
\bar{a} &= \bar{U}_{\xi i} \\
\bar{b} &= \bar{U}_{\eta i} \\
\bar{g} &= \bar{U}_{\zeta i}
\end{aligned} \tag{112}
$$

From the actual vehicle axes and the present desired vehicle axes, attitude errors can now be computed.

Consider a three-axis (roll, pitch, yaw) maneuver to reorient the vehicle from its actual attitude to the desired attitude, as three Euler rotations in the following order: roll, pitch, yaw. In matrix form, this is

$$B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos R & \sin R \\ 0 & -\sin R & \cos R \end{bmatrix} \begin{bmatrix} \cos P & 0 & -\sin P \\ 0 & 1 & 0 \\ \sin P & 0 & \cos P \end{bmatrix} \begin{bmatrix} \cos Y & \sin Y & 0 \\ -\sin Y & \cos Y & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{113}$$

where R, P, and Y are the respective roll, pitch, and yaw rotations.

The matrix multiplication yields

$$B = \begin{bmatrix} \cos P \cos Y & \sin R \sin P \cos Y + \cos R \sin Y & -\cos R \sin P \cos Y + \sin R \sin Y \\ -\cos P \sin Y & \sin R \sin P \sin Y + \cos R \cos Y & \cos R \sin P \sin Y + \sin R \cos Y \\ \sin P & -\sin R \cos P & \cos R \cos P \end{bmatrix} \tag{114}$$

Simplifying B with small angle approximations yields

$$B \approx \begin{bmatrix} 1 & Y & -P \\ -Y & 1 & R \\ P & -R & 1 \end{bmatrix} \tag{115}$$

Another expression for B in terms of body axes is

$$B = \begin{bmatrix} \bar{U}_{\xi dm} \cdot \bar{U}_{\xi} & \bar{U}_{\xi dm} \cdot \bar{U}_{\eta} & \bar{U}_{\xi dm} \cdot \bar{U}_{\zeta} \\ \bar{U}_{\eta dm} \cdot \bar{U}_{\xi} & \bar{U}_{\eta dm} \cdot \bar{U}_{\eta} & \bar{U}_{\eta dm} \cdot \bar{U}_{\zeta} \\ \bar{U}_{\zeta dm} \cdot \bar{U}_{\xi} & \bar{U}_{\zeta dm} \cdot \bar{U}_{\eta} & \bar{U}_{\zeta dm} \cdot \bar{U}_{\zeta} \end{bmatrix} \tag{116}$$

Finally, an expression is obtained for roll, pitch, and yaw attitude error by comparing like element B in Eqs. (115) and (116):

$$\begin{aligned} \text{Roll error} &= \overline{U}_{\eta dm} \cdot \overline{U}_{\zeta} \\ \text{Pitch error} &= -\overline{U}_{\xi dm} \cdot \overline{U}_{\zeta} \\ \text{Yaw error} &= \overline{U}_{\xi dm} \cdot \overline{U}_{\eta} \end{aligned} \tag{117}$$

REFERENCES

1. T. E. Darnaud, D. L. Kleinbub, and J. B. Shaul, Guidance, Control and Ground Equations for SSLV C-26, Aerospace Corporation Report No. TOR-0172(2112-02)-11, Reissue B (1 November 1972), Vol. I.

2. Program 624A Mission Specification for Flight Plan VII-J, Aerospace Corporation Report No. TOR-0172(2112-02)-6, Rev. 1 (28 September 1972).

3. H. Sokoloff, Program 624A Discrete List for Flight Plan VII-J, Aerospace Corporation Report No. TOR-0172(2112-02)-16, Reissue A (24 November 1972).

4. George W. Cherry, "A General Explicit Optimizing Guidance Law for Rocket Propelled Spacecraft," Proceedings of the AIAA/ION Astrodynamics Guidance and Control Conference, AIAA Paper No. 64-638 (24-26 August 1964).

5. D. MacPherson, An Explicit Solution to the Powered Flight Dynamics of a Rocket Vehicle, Aerospace Corporation Report No. TOR-169(3126)TN-2 (31 October 1962).

6. C. W. Pittman, The Design of Explicit Guidance Equations for Rocket Ascent, Aerospace Corportation Report No. TDR-469(5540-10)-4 (24 May 1965).

7. T.E. Darnaud, A Technique Allowing Continuous Operation of Integral Control, Aerospace Corporation Report No. TOR-1001(2116-60)-8 (13 September 1966).

8. K. A. Ehricke, "Environmental and Celestial Mechanics," Principles of Guided Missile Design, Vol. I: Space Flight (D. Van Nostrand Company, Inc., Princeton, N.J., 1960).

9. Carousel V Inertial Navigation System, System Technical Description EP0137, Delco Electronics Division of General Motors, Milwaukee, Wisconsin, 1969.