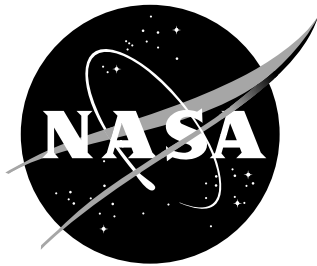


NASA/TP-2016-218956



Documenting the NASA Armstrong Flight Research Center Oblate Earth Simulation Equations of Motion and Integration Algorithm

*R. Clarke, L. Lintereur, and C. Bahm
Armstrong Flight Research Center, Edwards, California*

January 2016

NASA STI Program . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA scientific and technical information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI Program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI Program provides access to the NTRS Registered and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- **TECHNICAL PUBLICATION.**
Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers, but having less stringent limitations on manuscript length and extent of graphic presentations.
- **TECHNICAL MEMORANDUM.**
Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- **CONTRACTOR REPORT.**
Scientific and technical findings by NASA-sponsored contractors and grantees.

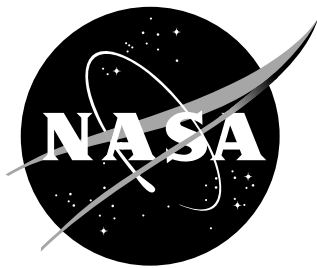
- **CONFERENCE PUBLICATION.**
Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.
- **SPECIAL PUBLICATION.**
Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- **TECHNICAL TRANSLATION.**
English- language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include organizing and publishing research results, distributing specialized research announcements and feeds, providing information desk and personal search support, and enabling data exchange services.

For more information about the NASA STI Program, see the following:

- Access the NASA STI program home page at <http://www.sti.nasa.gov>
- E-mail your question to help@sti.nasa.gov
- Phone the NASA STI Help Desk at 757-864-9658
- Write to:
NASA STI Information Desk
Mail Stop 148
NASA Langley Research Center
Hampton, VA 23681-2199

NASA/TP-2016-218956



Documenting the NASA Armstrong Flight Research Center Oblate Earth Simulation Equations of Motion and Integration Algorithm

*R. Clarke, L. Lintereur, and C. Bahm
Armstrong Flight Research Center, Edwards, California*

National Aeronautics and
Space Administration

Armstrong Flight Research Center
Edwards, California 93523-0273

January 2016

Acknowledgments

Thanks to the NASA Armstrong engineers, both past and present, who coded the Armstrong Flight Research Center core simulation.

The use of trademarks or names of manufacturers in this report is for accurate reporting and does not constitute an official endorsement, either expressed or implied, of such products or manufacturers by the National Aeronautics and Space Administration.

This report is available in electronic form at
<http://www.sti.nasa.gov/> and <http://ntrs.nasa.gov>.

Table of Contents

Abstract	2
Nomenclature	2
1 Introduction	5
2 Coordinate and Parameter Systems	6
3 States and Derived Parameters	9
4 Background for the Equations of Motion	13
5 Computation of Derived Parameters	13
5.1 Airmass Parameters	13
5.2 Exact Geodetic Parameters	17
6 Direction Cosine Matrix and Quaternions	20
7 Gravitational Model	22
8 Translational Equations	26
8.1 h -Frame Definition	27
8.2 Inertial Position Vector	27
8.3 Angular Velocity	27
8.4 Inertial Acceleration	28
8.5 Newton's Law	29
9 A Note on Rate of Change of Vehicle Mass	30
10 Rotational Equations	31
10.1 Effects of Rotating Mass	33
10.2 lv -Frame Orientation	34
11 Derivatives of Derived Parameters	35
12 Background for Integration Algorithm	38
13 Simulation Integration Algorithm	41
14 Oblate Earth Equations of Motion Considerations	42
15 Concluding Remarks	44
References	45

Abstract

A desire for more complete documentation of the National Aeronautics and Space Administration (NASA) Armstrong Flight Research Center (AFRC), Edwards, California legacy code used in the core simulation has led to this effort to fully document the oblate Earth six-degree-of-freedom equations of motion and integration algorithm. The authors of this report have taken much of the earlier work of the simulation engineering group and used it as a jumping-off point for this report. The largest addition this report makes is that each element of the equations of motion is traced back to first principles and at no point is the reader forced to take an equation on faith alone. There are no discoveries of previously unknown principles contained in this report; this report is a collection and presentation of textbook principles. The value of this report is that those textbook principles are herein documented in standard nomenclature that matches the form of the computer code `DERIVC`. Previous handwritten notes are much of the backbone of this work, however, in almost every area, derivations are explicitly shown to assure the reader that the equations which make up the oblate Earth version of the computer routine, `DERIVC`, are correct.

Nomenclature

a	speed of sound in feet per second
\mathbf{a}	general acceleration vector
AFRC	Armstrong Flight Research Center
b	wing span in feet
\bar{c}	mean aerodynamic chord in feet
cg	center of gravity
C_x^y	transformation matrix from general x -Frame to general y -Frame
$\frac{d}{dt}^x$	time derivative of a vector taken relative to the general x -Frame
$[D]$ or DCM	direction cosine matrix
$\frac{d}{dt}(\Delta R)$	vertical speed in feet per second of the vehicle along the radius vector
e_0	zeroth element of the lv to b attitude quaternion
\dot{e}_0	rate of change of the zeroth element of the lv to b attitude quaternion
e_1	first element of the lv to b attitude quaternion
\dot{e}_1	rate of change of the first element of the lv to b attitude quaternion
e_2	second element of the lv to b attitude quaternion
\dot{e}_2	rate of change of the second element of the lv to b attitude quaternion
e_3	third element of the lv to b attitude quaternion
\dot{e}_3	rate of change of the third element of the lv to b attitude quaternion
EOM	equations of motion
\mathbf{F}	general force vector
h	geocentric altitude above the oblate Earth
\dot{h}	altitude rate
\ddot{h}	altitude acceleration
h_{gd}	geodetic altitude of the position of the vehicle projected to the surface of the WGS-84 ellipsoid along the normal to that surface
\mathcal{I}	inertia tensor
I_{xx}	moment of inertia about the x_b axes in slug feet squared

I_{yy}	moment of inertia about the y_b axes in slug feet squared
I_{zz}	moment of inertia about the z_b axes in slug feet squared
I_{xy}	product of inertia about the $x - y_b$ axes in slug feet squared
I_{xz}	product of inertia about the $x - z_b$ axes in slug feet squared
I_{yz}	product of inertia about the $y - z_b$ axes in slug feet squared
JSC	Johnson Space Center
J_2	second zonal gravitational coefficient
J_3	third zonal gravitational coefficient
J_4	fourth zonal gravitational coefficient
J_5	fifth zonal gravitational coefficient
J_6	sixth zonal gravitational coefficient
J_7	seventh zonal gravitational coefficient
J_8	eighth zonal gravitational coefficient
m	mass of the vehicle in slugs
M	Mach number of the vehicle
NASA	National Aeronautics and Space Administration
NED	north-east-down coordinate system
\mathbf{p}	general linear momentum vector
p	roll rate
\dot{p}	roll acceleration
P_a	ambient static pressure in pounds per square foot
p_{lv}	Euler roll rate, angular rate around the $x - B$ axis with respect to the rotating lv -Frame
q	pitch rate
\dot{q}	pitch acceleration
\bar{q}	dynamic pressure in pounds per square foot
Q_c	impact pressure in pounds per square foot
q_{lv}	Euler pitch rate, angular rate around the $y - B$ axis with respect to the rotating lv -Frame
r	yaw rate
\dot{r}	yaw acceleration
r_{lv}	Euler yaw rate, angular rate around the $z - B$ axis with respect to the rotating lv -Frame
R	distance from the center of gravity of the vehicle to the center of the Earth in feet
RU_h	specific angular momentum of the vehicle
$\frac{d}{dt}(RU_h)$	rate of change of specific angular momentum of the vehicle
R_{local}	local radius of the Earth at the vehicle geocentric latitude in feet
R_{\oplus}	radius of the Earth at the equator (6378137 meters) (ref. 6)
R_p	radius of the Earth at the pole (6356752.3142 meters) (ref. 6)
S	reference wing area in feet squared
t	time
\dot{t}	rate of change of time
U_{bRA}	forward airspeed in feet per second in the b -Frame
U_g	forward gust speed in feet per second in the b -Frame
UEN	up-east-north coordinate system

\mathbf{V}	general velocity vector
\mathbf{v}	general vector
\mathbf{v}_x	vector in general x -Frame
V	speed of the vehicle with respect to the airmass
\dot{V}	rate of change of speed of the vehicle with respect to the airmass
VEAS	equivalent airspeed in knots (international nautical miles per hour)
V_{bRA}	side speed in feet per second in the b -Frame
V_e	rocket exhaust speed
V_g	side gust speed in feet per second in the b -Frame
V_i	inertial speed of the vehicle
w_h	inertial radial velocity in the h -Frame
\dot{w}_h	inertial radial acceleration of the h -Frame
W_{bRA}	down speed in feet per second in the b -Frame
W_g	down gust speed in feet per second in the b -Frame
WGS-84	World Geodetic System 1984
x	x -component of the ground track in fe -Frame
\dot{x}	x -component of ground track rate in fe -Frame
X_w	north component of the wind in feet per second in the lv -Frame
Y_w	east component of the wind in feet per second in the lv -Frame
y	y -component of the ground track in fe -Frame
\dot{y}	y -component of ground track rate in fe -Frame
Z_w	down component of the wind in feet per second in the lv -Frame
α	angle of attack of the vehicle with respect to the airmass
$\dot{\alpha}$	angle of attack rate
β	angle of sideslip of the vehicle with respect to the airmass
$\dot{\beta}$	angle of sideslip rate
ΔJ_2	permanent tidal deformation effect (9.3×10^{-9}) (ref. 6)
ΔR	height of vehicle in feet above a reference sphere with radius R_\oplus as measured along the vector from the center of the Earth to the cg of the vehicle
Δx	distance in feet in the x direction from the aerodynamic reference point to the cg of the vehicle measured in the b -Frame
Δy	distance in feet in the y direction from the aerodynamic reference point to the cg of the vehicle measured in the b -Frame
Δz	distance in feet in the z direction from the aerodynamic reference point to the cg of the vehicle measured in the b -Frame
θ	Euler pitch attitude in radians between the b -Frame and the lv -Frame
$\dot{\theta}$	Euler pitch attitude rate
λ	geocentric latitude of the vehicle in the e -Frame
$\dot{\lambda}$	geocentric latitude rate of the vehicle
ϑ	geocentric longitude of the vehicle in the e -Frame
$\dot{\vartheta}$	geocentric longitude rate of the vehicle
λ_{gc}	geocentric latitude of the position of the vehicle projected to the surface of the WGS-84 ellipsoid along the normal to that surface
λ_{gd}	geodetic latitude of the position of the vehicle projected to the surface of the WGS-84 ellipsoid along the normal to that surface

μ	gravitational constant of the Earth (3986005×10^8 meters cubed per second squared) (ref. 6)
ρ	air density in slugs per cubic foot
ρ_0	sea level air density (0.0023769 slugs per cubic foot)
ϕ	Euler roll attitude between the b -Frame and the lv -Frame
$\dot{\phi}$	Euler roll attitude rate
ψ	Euler heading attitude between the b -Frame and the lv -Frame
$\dot{\psi}$	Euler heading attitude rate
ψ_h	heading angle of the inertial velocity in the h -Frame with respect to the lv -Frame
$\dot{\psi}_h$	heading rate of the inertial velocity in the h -Frame with respect to the lv -Frame
ω^{xy}	the angular velocity of the y -Frame in the x -Frame
Ω_{\oplus}	angular velocity of the Earth (7292115×10^{-11} radians per second) (ref. 6)

1 Introduction

A review of the simulation engineering branch's handwritten documentation (ref. 1) of the National Aeronautics and Space Administration (NASA) Armstrong Flight Research Center (AFRC) core simulation was undertaken. The equations of motion used in this core simulation were built upon those presented in reference 2. This early documentation was expanded; this report is the result. Throughout this report the use of the typewriter font indicates reference to a computer program variable, routine name, or computer storage location. The handwritten notes included the following introduction:

A review of the EOM implemented by Bruce Powers for the Shuttle sim was undertaken in early 1985. The original implementation was undertaken after the Shuttle Sim was fully implemented with a flat Earth model. Compromises were made at the time to minimize impact on existing common blocks. The review was undertaken with the intent of establishing an understanding of the source of the EOM, cleaning up the common block interface, re-configuring the routines to be consistent with current Sim. Eng. practice, making corrections or additions as appropriate. The EOM were totally reworked, making rigorous use of naming conventions, and adding extensive comments. In the process, the following changes or corrections were made:

1. An oversight which resulted in single precision integration of the EOM was corrected to utilize double precision integration.
2. The expression for \dot{h} was expanded to include a term which includes oblate Earth effects.
3. \dot{x} and \dot{y} (velocities over the ground) were approximated by vehicle inertial velocities. These expressions were multiplied by the ratio $\frac{R_{local}}{R}$ to improve approximation.
4. Double precision is now used for many local variables.
5. The gravity terms have been upgraded based on input from Bill Lear, et. al. of JSC.

6. Calculations were added to determining geodetic latitude and altitude. *Still to be done.*

The last element listed, incorporation of equations to determine geodetic latitude and altitude, has now been accomplished using equations developed from several NASA reports, such as those cited as references 3 and 4, and an American Institute of Aeronautics and Astronautics (AIAA) journal article (ref. 5). Section 5.2 presents the background and derivation for this transformation. Several additional changes have been added to the simulation:

- All references to the simple Fisher ellipsoid parameters have been replaced with the newer World Geodetic System 1984 (WGS-84) ellipsoid parameters (ref. 6).
- Euler angles, ψ , θ , ϕ , have been replaced with quaternions to eliminate singularities. The Euler angles are still used for initialization and are available as computed parameters.
- The J_2 zonal harmonic gravity model has been replaced with a variable gravity model capable of running as a J_2 , J_4 or J_8 model. The parameters of this model were derived from the normalized gravitational coefficients defined in the WGS-84 Earth Gravitational Model (ref. 6).

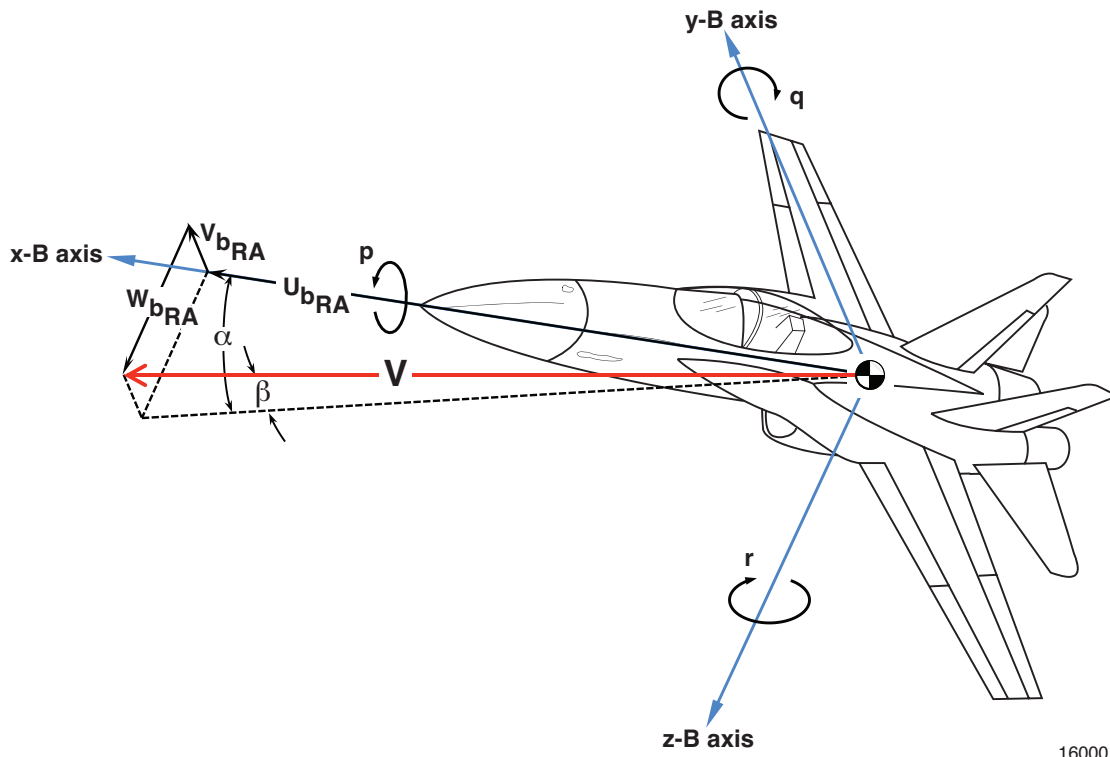
The algorithm used in the integration of the equations of motion has been used for many years at NASA AFRC. Comments in the FORTRAN code attribute the original implementation of the integration algorithm to Albert Myers in November 1972. The comments also indicate that circa 1978 the algorithm was modified by Albert Myers for the HiMAT program. The modification predicted the end-integration-step derivative based upon the start- and mid-integration-step values. The final modification was to integrate the values of many variables computed at the mid-integration-step to the end-integration-step. Several of these variables were originally states in the flat Earth equations of motion, but had become only computed parameters in the oblate Earth equations of motion.

2 Coordinate and Parameter Systems

Several coordinate or parameter systems are used in the equations of motion (EOM). Their definitions and descriptions follow:

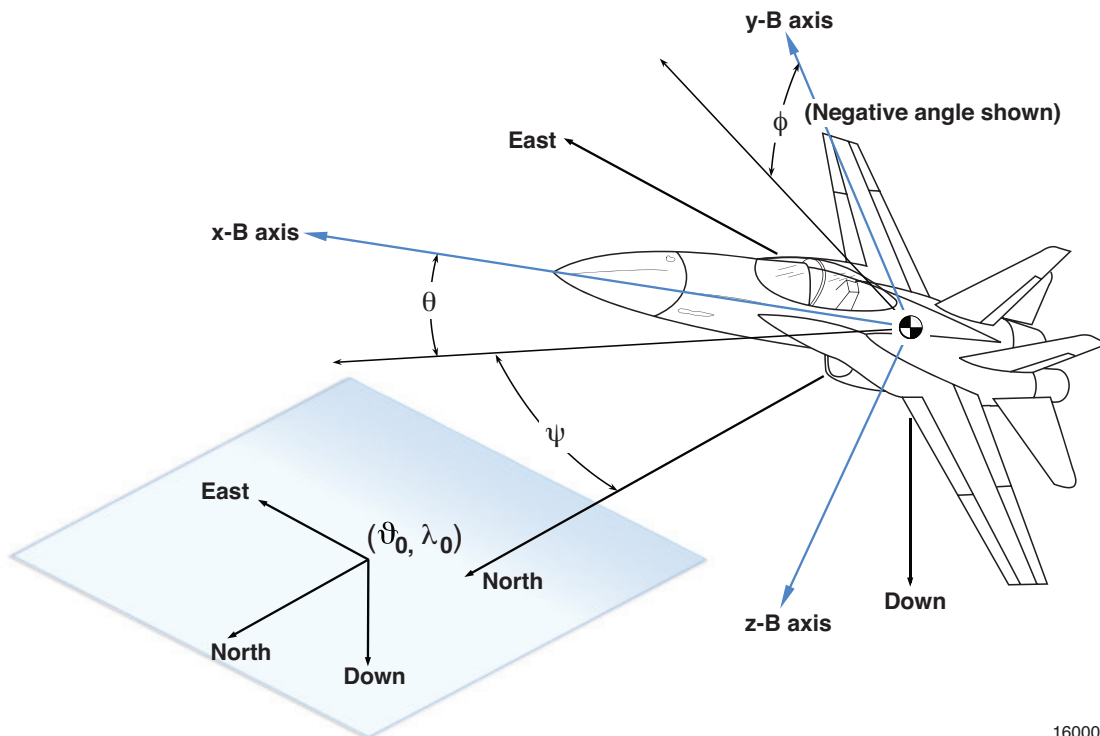
body axis coordinate system (b -Frame) The body axis system has its origin fixed at the center of gravity (cg) of the vehicle. The positive $x - B$ axis points out of the nose of the vehicle, the positive $y - B$ axis points out of the right wing of the vehicle, and the positive $z - B$ axis points through the bottom of the vehicle. The b -Frame is shown in figure 1.

local flat Earth coordinate system (fe -Frame) The local flat Earth coordinate system has its origin at an arbitrary location on the surface of the Earth at longitude ϑ_0 and latitude λ_0 . The positive $x - fe$ axis points north, the positive $y - fe$ axis points east, and the positive $z - fe$ axis points down. The fe -Frame is therefore a north-east-down, or NED, coordinate system. The fe -Frame is shown in figure 2.



160001

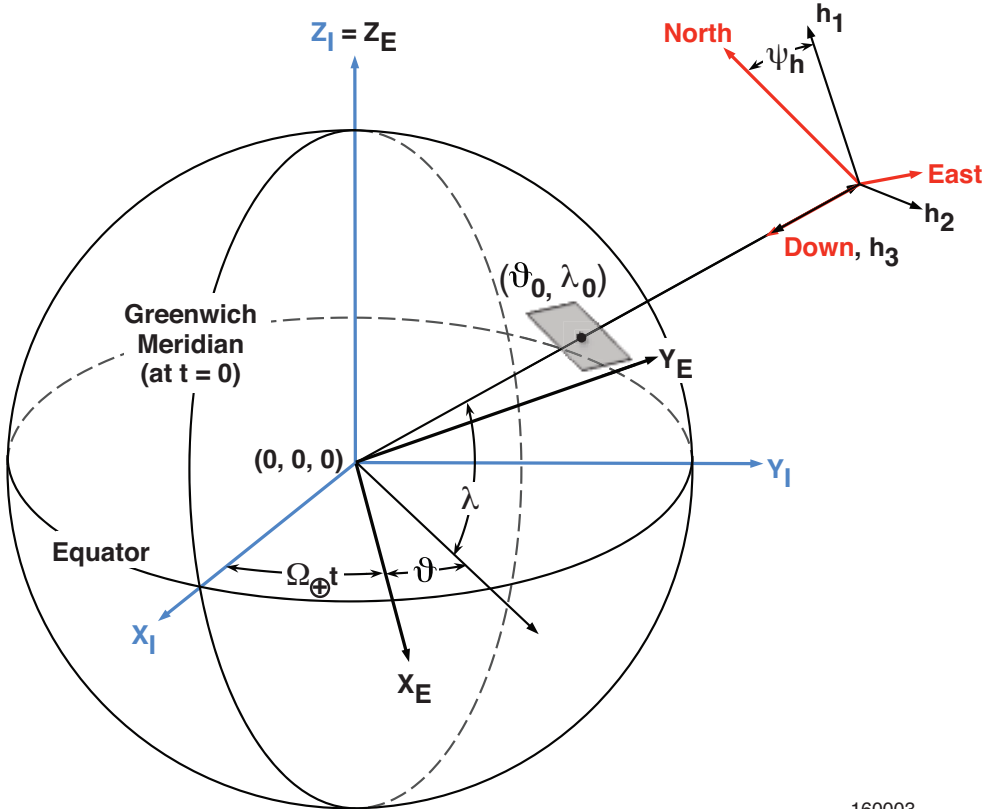
Figure 1. Body axis coordinate system (*b*-Frame).



160002

Figure 2. Local flat Earth coordinate system (*fe*-Frame).

local vertical coordinate system (*lv*-Frame) The local vertical coordinate system has its origin at the cg of the vehicle. The reference plane is perpendicular to the radius vector to the center of the Earth. The positive $x - lv$ axis points north, the positive $y - lv$ axis points east, and the positive $z - lv$ axis points through the center of the Earth. Therefore, *lv*-Frame is also an NED coordinate system. The Euler angles between the *lv*-Frame and *b*-Frame are the conventional heading, pitch, and roll angles. The Earth centric *lv*-, *h*-, *e*-, and *i*-Frames are shown in figure 3.



160003

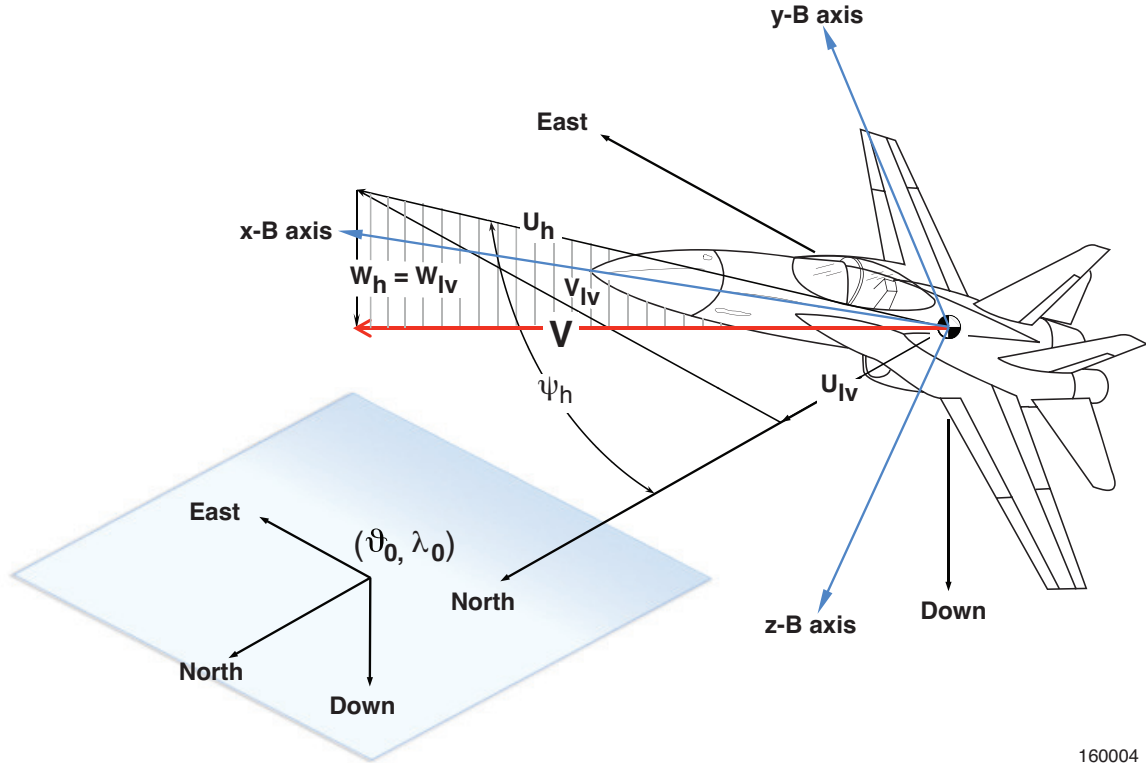
Figure 3. Local vertical coordinate system, heading coordinate system, Earth axis system, and inertial axis system (Earth centric view of *lv*-Frame, *h*-Frame, *e*-Frame, and *i*-Frame).

heading coordinate system (*h*-Frame) The heading coordinate system has its origin at the cg of the vehicle. The reference plane is perpendicular to the radius vector to the center of the Earth, and the reference direction (positive $x - H$) in the plane is the projection of the vehicle inertial velocity vector on the reference plane. The *h*-Frame differs from the *lv*-Frame by the single angle ψ_h , the heading angle of the inertial velocity vector. The vehicle centric definitions of the *lv*- and *h*-Frames are shown in figure 4.

Earth axis system (*e*-Frame) The Earth axis coordinate system has its origin at the center mass of the Earth and is fixed in the rotating Earth. $x - E$ and $y - E$ are in the equatorial plane of the Earth, with $x - E$ pointing through the Greenwich Meridian/equator intersection, $z - E$ points through the North Pole¹, and $y - E$ completes the right-hand

¹Strictly speaking, the $z - E$ points in the direction of the Conventional Terrestrial Pole for polar motion, as defined by the Bureau International de l'Heure for epoch 1984.

coordinate system. This e -Frame is equivalent to the WGS-84 coordinate system (ref. 6).



160004

Figure 4. Local vertical and heading coordinate systems (vehicle centric view of lv - and h -Frames).

inertial axis system (i -Frame) The inertial axis system (or Earth center inertial coordinate system) has its origin at the center of mass of the Earth. $x-I$ points through the time zero Greenwich Meridian/equator intersection, $z-I$ points through the North Pole of the Earth, and $y-I$ completes the right-hand coordinate system. Thus, the i -Frame represents the e -Frame coordinate system at time zero with the only difference being that it does not rotate with the Earth².

3 States and Derived Parameters

A list of the states and derivatives of the states used in the EOM and the definition of the state are presented below. These variables are all contained in the FORTRAN common block /DRVOUT/. The states used in the oblate Earth formulation of the EOM are different than those typically used in flat Earth EOM. The formulation of the states used in the oblate Earth EOM are particularly suited to airplanes or suborbital rocket flight as they define the vehicle location in easily understood latitude, longitude, and altitude³. The vehicle attitude is stored as

²This inertial frame is similar to those used by many other aircraft or spacecraft simulations, but because of the arbitrary choice made for time zero, it is only useful for simulation of a vehicle in the gravitational influence of the Earth. Thus, it cannot be used to simulate flights to Earth's moon.

³The only complication arises from the fact that global positioning system (GPS) geodetic coordinates must be converted to the geocentric quantities that the simulation uses. Several routines aid the user in this initialization

a quaternion, which is a simple extension from Euler angles. Use of the quaternion is required to ensure that the singularity associated with extreme pitch angles⁴ is avoided. The body rates are exactly the same as for flat Earth; however, if the vehicle is desired to remain at a constant attitude relative to a rotating Earth, then small angular rates are required of the vehicle to account for the rotation rate of the Earth⁵. The velocities of the vehicle are the only states that have no natural relationship to the flat Earth EOM states. To aid the user, several routines have been written to allow the user to specify airmass velocities (total speed, angle of attack, and angle of sideslip) and automatically convert these quantities to the oblate Earth simulation states.

t, T time in seconds since integration of the EOM started.

\dot{t} , TDOT rate of change of time in seconds per second.

p, P roll rate in radians per second, angular rate around the $x - B$ axis.

\dot{p} , PDOT roll acceleration in radians per second squared, angular acceleration around the $x - B$ axis.

q, Q pitch rate in radians per second, angular rate around the $y - B$ axis.

\dot{q} , QDOT pitch acceleration in radians per second squared, angular acceleration around the $y - B$ axis.

r, R yaw rate in radians per second, angular rate around the $z - B$ axis.

\dot{r} , RDOT yaw acceleration in radians per second squared, angular acceleration around the $z - B$ axis.

RU_h , RUH the distance to the center of the Earth multiplied by inertial velocity along the x_h axis in feet squared per second. This product represents the specific angular momentum of the vehicle in an orbit (which is constant while the vehicle is in an orbit acted only upon by a central gravitational field).

$\frac{d}{dt}(RU_h)$, RUHDOT rate of change of specific angular momentum of the vehicle in feet squared per second squared.

ψ_h , PSIH heading angle of the inertial velocity in the h -Frame with respect to the lv -Frame in radians. This angle defines the difference between the lv - and h -Frames. ψ_h is positive for clockwise rotation of the h -Frame with respect to the lv -Frame⁶.

$\dot{\psi}_h$, PSIHDT heading rate of the inertial velocity in the h -Frame with respect to the lv -Frame in radians per second.

phase.

⁴In aircraft, a *standard* Euler sequence is used: heading angle (0 to 360°); pitch attitude (-90 to +90°); and bank angle (-180 to +180°). Thus, a singularity exists at +90° and also at -90° pitch attitude.

⁵These small rates represent the rotation rate of the Earth projected onto the body axes of the vehicle, making the rates dependent upon the attitude of the vehicle. The simulation has helper routines which are used to set these Euler rates.

⁶This attribute is in keeping with the right-hand-rule for a positive rotation around the positive down vector of the lv -Frame.

w_h , WH inertial radial velocity in the h -Frame in feet per second.
 \dot{w}_h , WHDOT inertial radial acceleration of the h -Frame in feet per second squared.
 e_0 , E0 zeroth element of the lv to b attitude quaternion; it has no units.
 \dot{e}_0 , E0DOT rate of change of the zeroth element of the lv to b attitude quaternion.
 e_1 , E1 first element of the lv to b attitude quaternion; it has no units.
 \dot{e}_1 , E1DOT rate of change of the first element of the lv to b attitude quaternion.
 e_2 , E2 second element of the lv to b attitude quaternion; it has no units.
 \dot{e}_2 , E2DOT rate of change of the second element of the lv to b attitude quaternion.
 e_3 , E3 third element of the lv to b attitude quaternion; it has no units.
 \dot{e}_3 , E3DOT rate of change of the third element of the lv to b attitude quaternion.
 ΔR , DELR height of vehicle above a reference sphere with radius R_\oplus as measured along the vector from the center of the Earth to the cg of the vehicle in feet.
 $\frac{d}{dt}(\Delta R)$, DELRDT vertical velocity of the vehicle along the radius vector in feet per second.
 λ XLAT geocentric latitude of the vehicle in the e -Frame in radians.
 $\dot{\lambda}$, XLATDT geocentric latitude rate of the vehicle in radians per second.
 ϑ XLNG geocentric longitude of the vehicle in the e -Frame in radians.
 $\dot{\vartheta}$ XLNGDT geocentric longitude rate of the vehicle in radians per second.

Several important quantities are computed in the EOM. Some of these quantities represent variables which are normally considered to be states of the vehicle in a flat Earth EOM, but using this oblate Earth formulation of the EOM they are only derived parameters. The first group are contained in the common block /DRVOT2/. Their definitions are presented below:

V , V speed in feet per second of the vehicle with respect to the airmass.
 \dot{V} , VDOT rate of change of speed in feet per second squared.
 α , ALP angle of attack in radians of the vehicle with respect to the airmass.
 $\dot{\alpha}$, ALPDOT angle of attack rate in radians per second.
 β , BTA angle of sideslip in radians of the vehicle with respect to the airmass.
 $\dot{\beta}$, BTADOT angle of sideslip rate in radians per second.
 θ , THA Euler pitch attitude in radians between the b -Frame and the lv -Frame.
 $\dot{\theta}$, THADOT Euler pitch attitude rate in radians per second, measured between the b -Frame and the lv -Frame.

ϕ , PHI Euler roll attitude in radians between the b -Frame and the lv -Frame.

$\dot{\phi}$, PHIDOT Euler roll attitude rate in radians per second, measured between the b -Frame and the lv -Frame.

ψ , PSI Euler heading attitude in radians between the b -Frame and the lv -Frame.

$\dot{\psi}$, PSIDOT Euler heading attitude rate in radians per second, measured between the b -Frame and the lv -Frame.

h , H geocentric altitude in feet above the oblate Earth.

\dot{h} , HDOT altitude rate in feet per second.

x , X x -component of the ground track in feet, positive is north, measured in the fe -Frame.

\dot{x} , XDOT x -component of ground track rate in feet per second, measured in the fe -Frame.

y , Y y -component of the ground track in feet, positive is east, measured in the fe -Frame.

\dot{y} , YDOT y -component of ground track rate in feet per second, measured in the fe -Frame.

V_i , VI inertial speed of the vehicle in feet per second.

Several other quantities are also in the EOM to either aid the initialization of the simulation or to contain airmass related parameters. These are contained in six additional common blocks: /DRVOT3/, /EULERS/, /GSTOUT/, /SIMOUT/, /WINDAT/, and /WINDT2/. Their definitions are presented below:

λ_{gc} , LATGC geocentric latitude in radians of the position of the vehicle projected to the surface of the WGS-84 ellipsoid along the normal to that surface.

λ_{gd} , LATGD geodetic latitude in radians of the position of the vehicle projected to the surface of the WGS-84 ellipsoid along the normal to that surface.

h_{gd} , HGD geodetic altitude in feet of the position of the vehicle projected to the surface of the WGS-84 ellipsoid along the normal to that surface.

\ddot{h} , HACCEL altitude acceleration in feet per second squared (the time derivative of HDOT).

p_{lv} , PEULER Euler roll rate in radians per second, angular rate around the $x-B$ axis with respect to the rotating lv -Frame.

q_{lv} , QEULER Euler pitch rate in radians per second, angular rate around the $y-B$ axis with respect to the rotating lv -Frame.

r_{lv} , REULER Euler yaw rate in radians per second, angular rate around the $z-B$ axis with respect to the rotating lv -Frame.

$U_{b_{RA}}$, UBRA forward airspeed in feet per second in the b -Frame.

U_g , UG forward gust speed in feet per second in the b -Frame.

$V_{b_{RA}}$, VBRA side speed in feet per second in the b -Frame.

V_g , **VG** side gust speed in feet per second in the b -Frame.

$W_{b_{RA}}$, **WBRA** down speed in feet per second in the b -Frame.

W_g , **WG** down gust speed in feet per second in the b -Frame.

X_w , **XWIND** north component of the wind in feet per second in the lv -Frame.

Y_w , **YWIND** east component of the wind in feet per second in the lv -Frame.

Z_w , **ZWIND** down component of the wind in feet per second in the lv -Frame.

4 Background for the Equations of Motion

To determine the aerodynamic forces and moments which act on the vehicle it is necessary to determine the parameters which are inputs to the aerodynamic model. These are typically: α , $\dot{\alpha}$, β , $\dot{\beta}$, M , \bar{q} , Δx , Δy , Δz , S , \bar{c} , b , p , q , r , and all control surface positions.

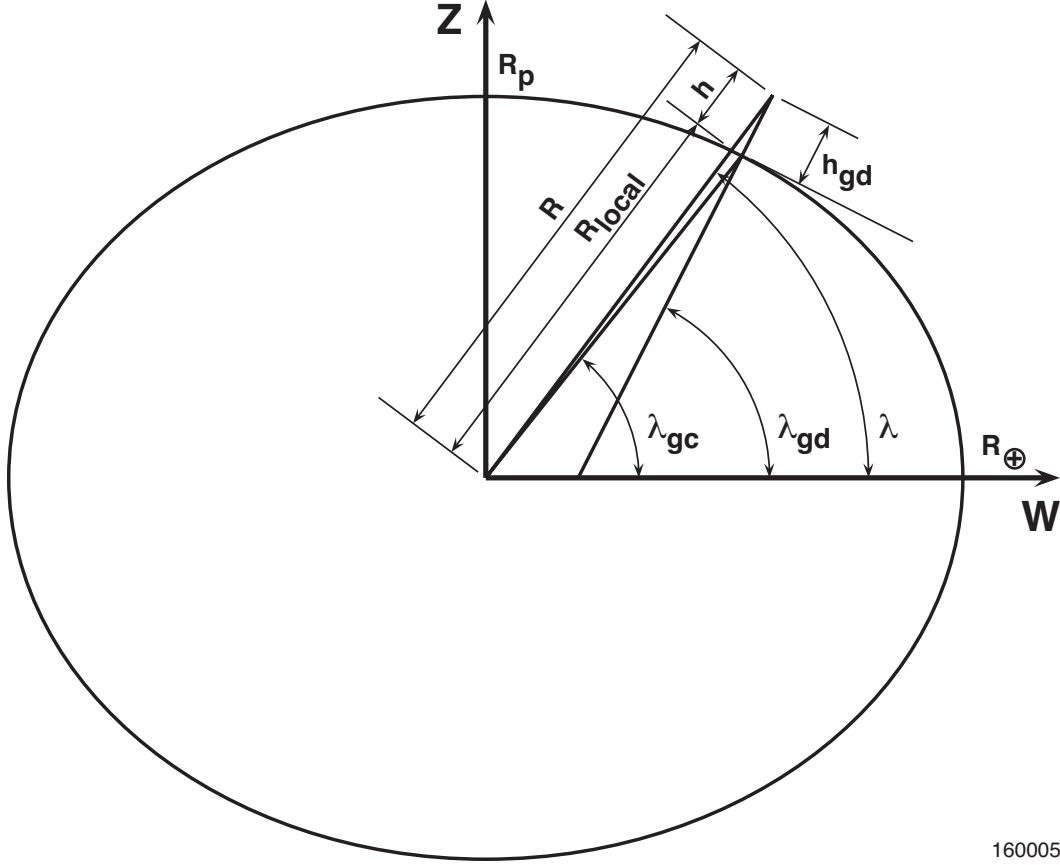
Several of these quantities are directly related to the atmospheric model. The EOM themselves do not depend upon a particular atmosphere model or its characteristics. Therefore, to keep this report simple, all discussion of the atmosphere will be of the simple atmosphere model defined in reference (7). This model defines several quantities which are used throughout the EOM. Speed of sound, a , is used in the computation of Mach number, M ; air density, ρ , is used to compute dynamic pressure, \bar{q} ; and both M and ambient static pressure, P_a , are used in the computation of impact pressure, Q_c . A simple wind, gust, and turbulence model will be assumed. The wind model defines three components of the wind vector, **XWIND**, **YWIND**, and **ZWIND**. These components are the north, east, and down components of the wind in feet per second. These quantities are added to airmass relative velocities in the lv -Frame. The lv -Frame is used because it is aligned with the NED reference frame of the wind model. The gust and turbulence models each help to define the quantities, **UG**, **VG**, and **WG**. These components are added to the airmass relative velocities, **UBRA**, **VBRA**, and **WBRA** in the b -Frame. Thus, all of these wind, gust, and turbulence components affect the computation of V , α , and β . The exact details of these computations is now discussed.

5 Computation of Derived Parameters

The EOM presented here use as states many quantities which have not been used in the development of aerodynamic models. Airmass relative velocities have typically been used in the development of these aerodynamic models. Navigation parameters such as geodetic latitude and geodetic altitude are also not available as vehicle states. The computation of these derived parameters from the vehicle state are the subject of the next few sections.

5.1 Airmass Parameters

The computation of vehicle angle of attack, sideslip, airmass speed, and altitude are all dependent upon the vehicle state (RU_h , ψ_h , W_h , λ , and ΔR). Altitude, h , is computed from the radius from the center of mass of the Earth to the cg of the vehicle, R , and local radius, R_{local} . We start with an ellipse representing a slice through the Earth along a meridian plane aligned with the



160005

Figure 5. Geocentric and geodetic parameters for the Earth ellipse.

longitude of the vehicle, λ . Figure 5 shows the resulting ellipse geometry including the various latitude and altitude quantities.

The local radius can be derived through simple mathematical manipulation of the equation for this ellipse:

$$\frac{w^2}{R_{\oplus}^2} + \frac{z^2}{R_p^2} = 1 \quad (1)$$

Making the following Cartesian to polar coordinate conversion:

$$w = R_{\text{local}} \cos(\lambda) \quad (2)$$

$$z = R_{\text{local}} \sin(\lambda) \quad (3)$$

we now have the following expression:

$$\frac{R_{\text{local}}^2 \cos^2(\lambda)}{R_{\oplus}^2} + \frac{R_{\text{local}}^2 \sin^2(\lambda)}{R_p^2} = 1 \quad (4)$$

Through simple rearrangement and trigonometric substitutions we have:

$$\begin{aligned}
R_{\text{local}}^2(\cos^2(\lambda)R_p^2 + \sin^2(\lambda)R_{\oplus}^2) &= R_p^2 R_{\oplus}^2 \\
R_{\text{local}}^2(\cos^2(\lambda)R_p^2 + \sin^2(\lambda)R_{\oplus}^2 + \cos^2(\lambda)R_{\oplus}^2 - \cos^2(\lambda)R_{\oplus}^2) &= R_p^2 R_{\oplus}^2 \\
R_{\text{local}}^2(\cos^2(\lambda)(R_p^2 - R_{\oplus}^2) + (\sin^2(\lambda) + \cos^2(\lambda))R_{\oplus}^2) &= R_p^2 R_{\oplus}^2 \\
R_{\text{local}}^2(\cos^2(\lambda)(R_p^2 - R_{\oplus}^2) + R_{\oplus}^2) &= R_p^2 R_{\oplus}^2 \\
R_{\text{local}}^2(\cos^2(\lambda)\left(\frac{R_p^2}{R_{\oplus}^2} - 1\right) + 1) &= R_p^2
\end{aligned} \tag{5}$$

Finally, solving for R_{local} and rearranging terms slightly we have:

$$R_{\text{local}} = R_p / \sqrt{1 - \left(1 - \left(\frac{R_p}{R_{\oplus}}\right)^2\right) \cos^2(\lambda)} \tag{6}$$

Defining ΔR by the following equation:

$$R = R_{\oplus} + \Delta R \tag{7}$$

We now have a definition for altitude:

$$h = R - R_{\text{local}} \tag{8}$$

The next step is to convert the h -Frame velocity states to the lv -Frame.

$$U_h = RU_h/R \tag{9}$$

$$U_{lv} = U_h \cos(\psi_h) \tag{10}$$

$$V_{lv} = U_h \sin(\psi_h) \tag{11}$$

$$W_{lv} = W_h \tag{12}$$

Once the velocities are in the lv -Frame we can compute the air mass relative quantities. The wind velocities in the lv -Frame have been computed with some process which typically uses h as its independent variable. These wind velocities have three components in the lv -Frame. The north component is X_w , the east component is Y_w , and finally the down component is Z_w (this last one is typically zero but is included for downdrafts or updrafts). This equation assumes that the atmosphere rotates with the Earth.

$$U_{lv_{RA}} = U_{lv} - X_w \tag{13}$$

$$V_{lv_{RA}} = V_{lv} - Y_w - R \cos(\lambda) \Omega_{\oplus} \tag{14}$$

$$W_{lv_{RA}} = W_{lv} - Z_w \tag{15}$$

Now that we have the air mass relative velocities we can transform them to the b -Frame using the direction cosine matrix (DCM) (which will be defined in the next section). This is also the point where we can add the gust velocities.

$$\mathbf{V}_{b_{RA}} = [D]\mathbf{V}_{lv_{RA}} + \mathbf{V}_g \tag{16}$$

Expanding these vectors we have:

$$\begin{Bmatrix} U_{b_{RA}} \\ V_{b_{RA}} \\ W_{b_{RA}} \end{Bmatrix} = [D] \begin{Bmatrix} U_{lv_{RA}} \\ V_{lv_{RA}} \\ W_{lv_{RA}} \end{Bmatrix} + \begin{Bmatrix} U_g \\ V_g \\ W_g \end{Bmatrix} \quad (17)$$

With these b -Frame velocities we can solve for V outright, and for α and β using the four-quadrant arctangent function, `atan2()`:

$$V = \sqrt{U_{b_{RA}}^2 + V_{b_{RA}}^2 + W_{b_{RA}}^2} \quad (18)$$

$$\tan(\alpha) = W_{b_{RA}}/U_{b_{RA}} \quad (19)$$

$$\tan(\beta) = \cos(\alpha)V_{b_{RA}}/U_{b_{RA}} \quad (20)$$

Inertial speed can be derived from either the h -Frame or lv -Frame inertial velocities. Using the h -Frame velocities we have:

$$V_i = \sqrt{U_h^2 + W_h^2} \quad (21)$$

Similarly, we can derive expressions for Mach number and dynamic pressure using standard definitions for these quantities and the previously defined airmass relative speed and local atmospheric characteristics.

$$M = V/a \quad (22)$$

$$\bar{q} = \frac{1}{2}\rho V^2 \quad (23)$$

Solving for equivalent airspeed requires only a little more work. The definition of equivalent airspeed is:

$$\text{VEAS} = V \sqrt{\frac{\rho}{\rho_0}} \quad (24)$$

Rearranging terms and using the previous definition of \bar{q} , we arrive at the following:

$$\begin{aligned} \text{VEAS} &= \sqrt{\frac{2}{\rho_0}} \sqrt{\frac{1}{2}\rho V^2} \\ &= \left(\sqrt{\frac{2}{\rho_0}}\right) \sqrt{\bar{q}} \end{aligned} \quad (25)$$

Since **VEAS** is usually expressed in knots (or nautical miles per hour) and not feet per second, we apply the following conversion using the number of seconds in an hour, the international standard definition for the nautical mile (ref. 6), and the sea level standard value for air density (ref. 7) .

$$\begin{aligned} \text{VEAS} &= \left(\frac{3600.0}{6076.11548556} \sqrt{\frac{2}{0.0023769}} \right) \sqrt{\bar{q}} \\ &= 17.1864\sqrt{\bar{q}} \end{aligned} \quad (26)$$

The next step is to derive the mass properties. This model typically computes the vehicle mass, cg location, moments of inertia, and products of inertia as a function of remaining propellant, landing gear location, et cetera. After these derived parameters have been computed we

call the aerodynamic model, the engine model, the reaction control system model, the landing gear model, gravity model, and any additional models that create forces or moments acting on the vehicle (such as a hold-down model or slosh model). The forces and moments acting on the vehicle can then be summed.

In the `DERIVC` routine these forces are all summed in the b -Frame except for the gravity forces. The landing gear model (and hold-down model if one is included) are treated in a special manner to ensure that the vehicle can “rest” on the gear without motion (or on the hold-down brackets).

$$\sum \mathbf{F}_b = \mathbf{F}_{\text{aero}} + \mathbf{F}_{\text{engine}} + \mathbf{F}_{\text{RCS}} + \mathbf{F}_{\text{gear}} + \dots \quad (27)$$

These forces are then divided by the mass to obtain the body accelerations.

$$\mathbf{a}_b = \mathbf{F}_b/m \quad (28)$$

These body accelerations are transformed from the b -Frame to the lv -Frame and the gravity accelerations are added. These gravity accelerations will be defined in a later section.

$$\mathbf{a}_{lv} = [D]^{-1}\mathbf{a}_b + \mathbf{a}_{glv} \quad (29)$$

Transforming these accelerations from the lv -Frame to the h -Frame:

$$\mathbf{a}_h = \begin{bmatrix} \cos(\psi_h) & \sin(\psi_h) & 0 \\ -\sin(\psi_h) & \cos(\psi_h) & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{a}_{lv} \quad (30)$$

Similarly, the moments are all summed in the b -Frame. Again the landing gear model (and hold-down model if one is included) are treated in a special manner to ensure that the vehicle can “rest” on the gear without rotation (or on the hold-down brackets).

$$\sum \mathbf{N}_b = \mathbf{N}_{\text{aero}} + \mathbf{N}_{\text{engine}} + \mathbf{N}_{\text{RCS}} + \mathbf{N}_{\text{gear}} + \dots \quad (31)$$

5.2 Exact Geodetic Parameters

The problem of computing the exact geodetic latitude and geodetic altitude of an arbitrary point given in the e -Frame coordinates is a complex problem. The technique used in the Armstrong simulation is built upon derivation taken from (refs. 3, 4, and 5). The algorithm and its derivation are presented below:

The distance of a vehicle to the Earth ellipsoid is given by:

$$d = \sqrt{(X - X_g)^2 + (Y - Y_g)^2 + (Z - Z_g)^2} \quad (32)$$

where (X_g, Y_g, Z_g) is the location of the vehicle and (X, Y, Z) is the location of a point on the Earth ellipsoid. For purposes of computing the geodetic parameters, the geometry can be simplified in the same manner as previously shown in figure 5.

By eliminating longitude, the distance of a vehicle to Earth ellipsoid is given by:

$$d = \sqrt{(W - W_g)^2 + (Z - Z_g)^2} \quad (33)$$

where (W_g, Z_g) is the location of the vehicle and (W, Z) is the location of a point on the Earth ellipsoid.

The minimum distance is the altitude of the vehicle, h_{gd} , and can be determined through the *Lagrange multiplier* method (ref. 8).

Scaling each of the simplified e -Frame vehicle coordinates with radius, R , we make the following substitutions.

$$a_g = R_{\oplus}/R \quad (34)$$

$$b_g = R_p/R \quad (35)$$

$$W_g = \cos(\lambda)R \quad (36)$$

$$Z_g = \sin(\lambda)R \quad (37)$$

$$W_{0_g} = \cos(\lambda) \quad (38)$$

$$Z_{0_g} = \sin(\lambda) \quad (39)$$

In the following formulation the performance index is $L(W, Z)$, and the constraint relationship is $f(W, Z) = 0$. Therefore, for our simple problem:

$$L(W, Z) = d^2 = (W - W_{0_g})^2 + (Z - Z_{0_g})^2 \quad (40)$$

since minimum distance requires minimum d^2 . The constraint equation is:

$$f(W, Z) = \frac{W^2}{a_g^2} + \frac{Z^2}{b_g^2} - 1 = 0 \quad (41)$$

Now we can form the adjoint, $H(W, Z)$ with a single multiplier, α :

$$\begin{aligned} H(W, Z, \alpha) &= L(W, Z) + \alpha f(W, Z) \\ &= (W - W_{0_g})^2 + (Z - Z_{0_g})^2 - \alpha \left(\frac{W^2}{a_g^2} + \frac{Z^2}{b_g^2} - 1 \right) \end{aligned} \quad (42)$$

Each of the partial derivatives of H are formed and equated to zero. Thus, the following:

$$\frac{\partial H}{\partial W} = 2(W - W_{0_g}) - \frac{2\alpha W}{a_g^2} = 0 \quad (43)$$

$$\frac{\partial H}{\partial Z} = 2(Z - Z_{0_g}) - \frac{2\alpha Z}{b_g^2} = 0 \quad (44)$$

$$\frac{\partial H}{\partial \alpha} = -\frac{W^2}{a_g^2} - \frac{Z^2}{b_g^2} + 1 = 0 \quad (45)$$

Solving for W and Z we have:

$$W = \frac{W_{0_g} a_g^2}{a_g^2 - \alpha} \quad (46)$$

$$Z = \frac{Z_{0_g} b_g^2}{b_g^2 - \alpha} \quad (47)$$

Finally, substituting these expressions for W and Z into equation (45) we form the following normalized fourth-order polynomial:

$$\begin{aligned} P_g(\alpha) &= \alpha^4 - 2a_g^2 b_g^2 \left(\left(\frac{1}{a_g}\right)^2 + \left(\frac{1}{b_g}\right)^2 \right) \alpha^3 + a_g^2 b_g^2 \left(4 + \left(\frac{a_g}{b_g}\right)^2 + \left(\frac{b_g}{a_g}\right)^2 - \left(\frac{W_{0_g}}{b_g}\right)^2 - \left(\frac{Z_{0_g}}{a_g}\right)^2 \right) \alpha^2 \\ &\quad + 2a_g^2 b_g^2 \left(W_{0_g}^2 + Z_{0_g}^2 - a_g^2 - b_g^2 \right) \alpha + a_g^2 b_g^2 \left(a_g^2 b_g^2 - W_{0_g}^2 b_g^2 - Z_{0_g}^2 a_g^2 \right) = 0 \end{aligned} \quad (48)$$

The piercing point is found by setting α_g to the smallest real root of $P_g(\alpha)$.

$$\alpha_g = \min(\Re(\mathcal{R}(P_g(\alpha)))) \quad (49)$$

Compute the coordinate of the piercing point on the surface of the reference ellipsoid.

$$W_{\text{pierce}} = R \frac{W_{0g} a_g^2}{a_g^2 - \alpha_g} \quad (50)$$

$$Z_{\text{pierce}} = R \frac{Z_{0g} b_g^2}{b_g^2 - \alpha_g} \quad (51)$$

Compute the altitude above (or below) the reference ellipsoid⁷.

$$h_{gd} = -\text{sgn}(\alpha_g) \sqrt{(W_{\text{pierce}} - W_g)^2 + (Z_{\text{pierce}} - Z_g)^2} \quad (52)$$

With the piercing point solved, we can immediately solve for the geocentric latitude, λ_{gc} , of the position of the vehicle projected to the surface of the Earth using the four-quadrant arctangent function, `atan2()`.

$$\tan(\lambda_{gc}) = Z_{\text{pierce}}/W_{\text{pierce}} \quad (53)$$

Lastly, we can solve for the geodetic latitude, λ_{gd} , using simple geometry. Using equation (1) and solving for z we have:

$$z = \frac{R_p}{R_{\oplus}} \sqrt{R_{\oplus}^2 - w^2} \quad (54)$$

For an arbitrary point (w, z) on the ellipse we can find the slope of a tangent to the surface by taking the derivative of expression z with respect to w , yielding the following equation:

$$\frac{dz}{dw} = -w \frac{R_p}{R_{\oplus}} \frac{1}{\sqrt{(R_{\oplus}^2 - w^2)}} \quad (55)$$

and a normal to the ellipse at this point has a slope defined by:

$$-\frac{dw}{dz} = \frac{1}{w} \frac{R_{\oplus}}{R_p} \sqrt{(R_{\oplus}^2 - w^2)} = \frac{R_{\oplus}^2}{R_p^2} \frac{z}{w} \quad (56)$$

but this slope is also the geodetic latitude, λ_{gd} , at the piercing point and we can solve for its value using the four-quadrant arctangent function, `atan2()`:

$$\begin{aligned} \tan(\lambda_{gd}) &= -\frac{dw}{dz} \\ &= \frac{R_{\oplus}^2}{R_p^2} \frac{Z_{\text{pierce}}}{W_{\text{pierce}}} = \frac{a_g^2}{b_g^2} \frac{Z_{\text{pierce}}}{W_{\text{pierce}}} \end{aligned} \quad (57)$$

⁷The sign of α_g is negative for altitude above the Earth and positive for below.

6 Direction Cosine Matrix and Quaternions

The direction cosine matrix, $[D]$ (ref. 9), is used to perform the transformation from $lv-$ to $b-$ Frame. The $[D]$ used in the AFRC simulation follows the standard (for airplanes) convention for the Euler rotation sequence, ψ , θ , and ϕ ⁸. Originally the $[D]$ was defined using the Euler angles (ψ , θ , and ϕ) with the following equation:

$$[D] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix} \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (58)$$

Multiplying these matrices together we arrive at this form:

$$[D] = \begin{bmatrix} \cos(\psi) \cos(\theta) & \sin(\psi) \cos(\theta) & -\sin(\theta) \\ \cos(\psi) \sin(\theta) \sin(\phi) - \sin(\psi) \cos(\phi) & \sin(\psi) \sin(\theta) \sin(\phi) + \cos(\psi) \cos(\phi) & \cos(\theta) \sin(\phi) \\ \cos(\psi) \sin(\theta) \cos(\phi) + \sin(\psi) \sin(\phi) & \sin(\psi) \sin(\theta) \cos(\phi) - \cos(\psi) \sin(\phi) & \cos(\theta) \cos(\phi) \end{bmatrix} \quad (59)$$

Now $[D]$ is defined from the quaternion e (refs. 9 and 10) with the following equation:

$$[D] = \begin{bmatrix} e_0^2 + e_1^2 - e_2^2 - e_3^2 & 2(e_1e_2 + e_0e_3) & 2(e_1e_3 - e_0e_2) \\ 2(e_1e_2 - e_0e_3) & e_0^2 - e_1^2 + e_2^2 - e_3^2 & 2(e_2e_3 + e_0e_1) \\ 2(e_1e_3 + e_0e_2) & 2(e_2e_3 - e_0e_1) & e_0^2 - e_1^2 - e_2^2 + e_3^2 \end{bmatrix} \quad (60)$$

The quaternion is defined as:

$$e = e_0 + e_1i + e_2j + e_3k \quad (61)$$

where the quantities e_0 , e_1 , e_2 , and e_3 are all real numbers and i , j , and k satisfy the following relationships:

$$\begin{aligned} i^2 &= j^2 = k^2 = -1 \\ ij &= -ji = k \\ jk &= -kj = i \\ ki &= -ik = j \end{aligned}$$

The quantity e_0 is the scalar part of the quaternion and $e_1i + e_2j + e_3k$ make up the vector portion. Since the quaternion is made up of four elements to parameterize a DCM made up from three Euler angles, a fourth equation constrains the quaternions. This equation is:

$$e_0^2 + e_1^2 + e_2^2 + e_3^2 = 1 \quad (62)$$

Thus, after some number of integration steps, the quaternion elements must be *normalized* by dividing by the square root of equation (62).

The quaternion elements are initialized from the initial vehicle Euler angle with the following relationships:

$$e = \begin{Bmatrix} e_0 \\ e_1 \\ e_2 \\ e_3 \end{Bmatrix} = \begin{Bmatrix} +\cos(\frac{\psi}{2})\cos(\frac{\theta}{2})\cos(\frac{\phi}{2}) + \sin(\frac{\psi}{2})\sin(\frac{\theta}{2})\sin(\frac{\phi}{2}) \\ +\cos(\frac{\psi}{2})\cos(\frac{\theta}{2})\sin(\frac{\phi}{2}) - \sin(\frac{\psi}{2})\sin(\frac{\theta}{2})\cos(\frac{\phi}{2}) \\ +\cos(\frac{\psi}{2})\sin(\frac{\theta}{2})\cos(\frac{\phi}{2}) + \sin(\frac{\psi}{2})\cos(\frac{\theta}{2})\sin(\frac{\phi}{2}) \\ -\cos(\frac{\psi}{2})\sin(\frac{\theta}{2})\sin(\frac{\phi}{2}) + \sin(\frac{\psi}{2})\cos(\frac{\theta}{2})\cos(\frac{\phi}{2}) \end{Bmatrix} \quad (63)$$

⁸In spacecraft literature this is referred to as a *Type 1: 3-2-1 Euler Angle Rotation* (ref. 11).

The derivatives of quaternion elements are taken from (ref. 10). Since the quaternion defines our attitude with respect to the lv -Frame, the equations have been modified to use the Euler angular rate vector (which will be defined in eq. (178)):

$$\dot{\mathbf{e}} = \begin{Bmatrix} \dot{e}_0 \\ \dot{e}_1 \\ \dot{e}_2 \\ \dot{e}_3 \end{Bmatrix} = \frac{1}{2} \begin{bmatrix} -e_1 & -e_2 & -e_3 \\ +e_0 & -e_3 & +e_2 \\ +e_3 & +e_0 & -e_1 \\ -e_2 & +e_1 & +e_0 \end{bmatrix} \begin{Bmatrix} p \\ q \\ r \end{Bmatrix}_{lv} \quad (64)$$

The Euler angles are derived from this quaternion by comparing elements of the Euler-DCM with the quaternion-DCM using trigonometric identities. The first angle to extract is θ . Looking at the Euler-DCM we can isolate the θ in several ways. The easiest is to equate the upper right-hand elements of the Euler-DCM and the quaternion-DCM. Doing this yields the following expression for θ :

$$-\sin(\theta) = 2(e_1e_3 - e_0e_2) \quad (65)$$

Unless the quaternion elements are normalized at every integration step this expression can exhibit numerical problems as θ approaches $\pm 90^\circ$. It is much safer, and only somewhat more computationally expensive, to find an expression utilizing $\arctan()$. In this case, the expression becomes:

$$\frac{\sin(\theta)}{\sqrt{\cos^2(\psi) \cos^2(\theta) + \sin^2(\psi) \cos^2(\theta)}} = \frac{-2(e_1e_3 - e_0e_2)}{\sqrt{(e_0^2 + e_1^2 - e_2^2 - e_3^2)^2 + (2(e_1e_2 + e_0e_3))^2}} \quad (66)$$

but through trigonometric identities this first expression reduces to $\tan(\theta)$. In the computer routine we use the four-quadrant `atan2()` routine to solve for these $\tan()$ angles. In the same manner, if θ is neither $+90$ or -90 deg, we can use simple trigonometric relationships to solve for ϕ and ψ using elements of the first row and third column of the Euler-DCM matrix.

In this general case we can determine that θ is not equal to either $+90$ or -90 deg by testing the sum of the squares of the first and second elements of the quaternion-DCM, $(e_0^2 + e_1^2 - e_2^2 - e_3^2)^2 + (2(e_1e_2 + e_0e_3))^2 \approx 0$. Therefore, if $(e_0^2 + e_1^2 - e_2^2 - e_3^2)^2 + (2(e_1e_2 + e_0e_3))^2 > 1.0 \times 10^{-14}$:

$$\tan(\theta) = \frac{-2(e_1e_3 - e_0e_2)}{\sqrt{(e_0^2 + e_1^2 - e_2^2 - e_3^2)^2 + (2(e_1e_2 + e_0e_3))^2}} \quad (67)$$

$$\tan(\phi) = \frac{2(e_0e_1 + e_2e_3)}{(e_0^2 - e_1^2 - e_2^2 + e_3^2)} \quad (68)$$

$$\tan(\psi) = \frac{2(e_1e_2 + e_0e_3)}{(e_0^2 + e_1^2 - e_2^2 - e_3^2)} \quad (69)$$

else the Euler-DCM reduces to the following:

$$[D] = \pm \begin{bmatrix} 0 & 0 & -1 \\ \cos(\psi) \sin(\phi) - \sin(\psi) \cos(\phi) & \sin(\psi) \sin(\phi) + \cos(\psi) \cos(\phi) & 0 \\ \cos(\psi) \cos(\phi) + \sin(\psi) \sin(\phi) & \sin(\psi) \cos(\phi) - \cos(\psi) \sin(\phi) & 0 \end{bmatrix} \quad (70)$$

where $[D]$ takes the positive value if $\theta = +90$ deg and takes the negative value if $\theta = -90$ deg. This expression can be further reduced to:

$$[D] = \pm \begin{bmatrix} 0 & 0 & -1 \\ \sin(\phi - \psi) & \cos(\phi - \psi) & 0 \\ \cos(\phi - \psi) & -\sin(\phi - \psi) & 0 \end{bmatrix} \quad (71)$$

which indicates that one of the angles, ϕ or ψ , can be set to an arbitrary value while the other angle is solved. We choose to set $\psi = 0$. The last angle is solved using another $\tan()$ function:

$$\sin(\phi)/\cos(\phi) = \tan(\phi) = \frac{\pm 2(e_1 e_2 - e_0 e_3)}{\pm(e_0^2 - e_1^2 + e_2^2 - e_3^2)} \quad (72)$$

Note: the decision logic for determining the correct quadrant is automatic if we use the `atan2()` function; however, this requires that we keep the correct signs on both of the arguments. Therefore if $\theta > 0$ both arguments are also positive, but if $\theta < 0$ both arguments must be multiplied by -1 .

7 Gravitational Model

The Earth gravitational model is based upon the spherical harmonic expansion of the gravitational potential (ref. 11).

$$\phi(X, Y, Z) = -\frac{\mu}{R} \left(1 - \sum_{n=2}^{\infty} \left(\frac{R_{\oplus}}{R} \right)^n J_n \mathcal{P}_n \left(\frac{Z}{R} \right) \right) \quad (73)$$

or more simply:

$$\phi(X, Y, Z) = -\frac{\mu}{R} - \sum_{n=2}^{\infty} -\frac{\mu}{R} J_n \left(\frac{R_{\oplus}}{R} \right)^n \mathcal{P}_n \left(\frac{Z}{R} \right) = -\frac{\mu}{R} - \sum_{n=2}^{\infty} \phi_n(X, Y, Z) \quad (74)$$

The notation has been put into the “standard” J_n notation where zonal harmonics only are included; sectoral and tesseral are not usually important unless the vehicle enters a geosynchronous orbit. The J_n coefficients can be computed from the C_{n0} coefficients (ref. 6) using the following equation:

$$J_n = -C_{n0} \sqrt{2n+1} \quad (75)$$

Table 1. The WGS-84 Earth Gravitational Model (selected coefficients copied from reference 6 table 5.2) and computed zonal harmonic gravitational coefficients.

Degree and Order		Normalized Gravitational Coefficients		Computed J_n Coefficients
n	m	C_{nm}	S_{nm}	J_n
2	0	$-0.48416685 \times 10^{-3}$	0.0	$+1.08262999 \times 10^{-3}$
3	0	$+0.95706390 \times 10^{-6}$	0.0	$-2.53215307 \times 10^{-6}$
4	0	$+0.53699587 \times 10^{-6}$	0.0	$-1.61098761 \times 10^{-6}$
5	0	$+0.71092048 \times 10^{-7}$	0.0	$-2.35785649 \times 10^{-7}$
6	0	$-0.15064821 \times 10^{-6}$	0.0	$+5.43169846 \times 10^{-7}$
7	0	$+0.85819217 \times 10^{-7}$	0.0	$-3.32376398 \times 10^{-7}$
8	0	$+0.42979835 \times 10^{-7}$	0.0	$-1.77210399 \times 10^{-7}$

No tesseral or sectoral harmonics are included. The gravity model can be run in the J_2 , J_4 , or J_8 mode, by truncating the polynomial series at the appropriate term.

With this gravitational potential model of the Earth, the e -Frame accelerations can be found from:

$$\mathbf{a} = \nabla\phi = \frac{\partial\phi}{\partial x}\vec{i} + \frac{\partial\phi}{\partial y}\vec{j} + \frac{\partial\phi}{\partial z}\vec{k} \quad (76)$$

Using a spherical harmonic expansion, the terms of the gravitational potential at a point in the e -Frame are defined by:

$$\phi_n(X, Y, Z) = -\frac{\mu J_n R_{\oplus}^n \mathcal{P}_n\left(\frac{Z}{R}\right)}{R^{n+1}} \quad (77)$$

where $\mathcal{P}_n\left(\frac{Z}{R}\right)$ is a Legendre polynomial and $R = \sqrt{X^2 + Y^2 + Z^2}$.

The recursive relationship which defines these Legendre polynomials is:

$$\mathcal{P}_{n+1}\left(\frac{Z}{R}\right) = \frac{(2n+1)\left(\frac{Z}{R}\right)\mathcal{P}_n\left(\frac{Z}{R}\right) - n\mathcal{P}_{n-1}\left(\frac{Z}{R}\right)}{(n+1)} \quad (78)$$

where:

$$\mathcal{P}_0\left(\frac{Z}{R}\right) = 1 \quad (79)$$

$$\mathcal{P}_1\left(\frac{Z}{R}\right) = \frac{Z}{R} \quad (80)$$

Thus, each of the terms of the partial derivative of the gravitational potential with respect to X , Y , and Z are:

$$\frac{\partial\phi_n}{\partial X} = -\mu J_n R_{\oplus}^n \left(R^2 \frac{\partial\mathcal{P}_n\left(\frac{Z}{R}\right)}{\partial X} - X(n+1)\mathcal{P}_n\left(\frac{Z}{R}\right) \right) / R^{n+3} \quad (81)$$

$$\frac{\partial\phi_n}{\partial Y} = -\mu J_n R_{\oplus}^n \left(R^2 \frac{\partial\mathcal{P}_n\left(\frac{Z}{R}\right)}{\partial Y} - Y(n+1)\mathcal{P}_n\left(\frac{Z}{R}\right) \right) / R^{n+3} \quad (82)$$

$$\frac{\partial\phi_n}{\partial Z} = -\mu J_n R_{\oplus}^n \left(R^2 \frac{\partial\mathcal{P}_n\left(\frac{Z}{R}\right)}{\partial Z} - Z(n+1)\mathcal{P}_n\left(\frac{Z}{R}\right) \right) / R^{n+3} \quad (83)$$

where the recursive relationships which define the partial derivatives of the Legendre polynomials with respect to X , Y , and Z are:

$$\frac{\partial\mathcal{P}_{n+1}\left(\frac{Z}{R}\right)}{\partial X} = \left((2n+1) \left(\frac{\partial\left(\frac{Z}{R}\right)}{\partial X} \mathcal{P}_n\left(\frac{Z}{R}\right) + (Z/R) \frac{\partial\mathcal{P}_n\left(\frac{Z}{R}\right)}{\partial X} \right) - n \frac{\partial\mathcal{P}_{n-1}\left(\frac{Z}{R}\right)}{\partial X} \right) / (n+1) \quad (84)$$

$$\frac{\partial\mathcal{P}_{n+1}\left(\frac{Z}{R}\right)}{\partial Y} = \left((2n+1) \left(\frac{\partial\left(\frac{Z}{R}\right)}{\partial Y} \mathcal{P}_n\left(\frac{Z}{R}\right) + (Z/R) \frac{\partial\mathcal{P}_n\left(\frac{Z}{R}\right)}{\partial Y} \right) - n \frac{\partial\mathcal{P}_{n-1}\left(\frac{Z}{R}\right)}{\partial Y} \right) / (n+1) \quad (85)$$

$$\frac{\partial\mathcal{P}_{n+1}\left(\frac{Z}{R}\right)}{\partial Z} = \left((2n+1) \left(\frac{\partial\left(\frac{Z}{R}\right)}{\partial Z} \mathcal{P}_n\left(\frac{Z}{R}\right) + (Z/R) \frac{\partial\mathcal{P}_n\left(\frac{Z}{R}\right)}{\partial Z} \right) - n \frac{\partial\mathcal{P}_{n-1}\left(\frac{Z}{R}\right)}{\partial Z} \right) / (n+1) \quad (86)$$

where:

$$\frac{\partial \mathcal{P}_0(\frac{Z}{R})}{\partial X} = 0 \quad (87)$$

$$\frac{\partial \mathcal{P}_0(\frac{Z}{R})}{\partial Y} = 0 \quad (88)$$

$$\frac{\partial \mathcal{P}_0(\frac{Z}{R})}{\partial Z} = 0 \quad (89)$$

$$\frac{\partial \mathcal{P}_1(\frac{Z}{R})}{\partial X} = -XZ/R^3 \quad (90)$$

$$\frac{\partial \mathcal{P}_1(\frac{Z}{R})}{\partial Y} = -YZ/R^3 \quad (91)$$

$$\frac{\partial \mathcal{P}_1(\frac{Z}{R})}{\partial Z} = (R^2 - Z^2)/R^3 \quad (92)$$

or, in normalized form:

$$\frac{\partial \phi_n}{\partial X} = -\frac{\mu}{R^2} J_n(R_\oplus/R)^n \left(R \frac{\partial \mathcal{P}_n(\frac{Z}{R})}{\partial X} - \left(\frac{X}{R} \right) (n+1) \mathcal{P}_n \left(\frac{Z}{R} \right) \right) \quad (93)$$

$$\frac{\partial \phi_n}{\partial Y} = -\frac{\mu}{R^2} J_n(R_\oplus/R)^n \left(R \frac{\partial \mathcal{P}_n(\frac{Z}{R})}{\partial Y} - \left(\frac{Y}{R} \right) (n+1) \mathcal{P}_n \left(\frac{Z}{R} \right) \right) \quad (94)$$

$$\frac{\partial \phi_n}{\partial Z} = -\frac{\mu}{R^2} J_n(R_\oplus/R)^n \left(R \frac{\partial \mathcal{P}_n(\frac{Z}{R})}{\partial Z} - \left(\frac{Z}{R} \right) (n+1) \mathcal{P}_n \left(\frac{Z}{R} \right) \right) \quad (95)$$

Summing the first eight terms of the derivative gravitational potential equation with respect to X we find:

$$\begin{aligned} \frac{\partial \phi}{\partial X} = & -(\mu/R^2) \left(\frac{X}{R} \right) (1 \\ & + \left(\frac{3}{2} (J_2 + \Delta J_2) (R_\oplus/R)^2 \left(-5 \left(\frac{Z}{R} \right)^2 + 1 \right) \right. \\ & + \left(\frac{5}{2} J_3 (R_\oplus/R)^3 \left(-7 \left(\frac{Z}{R} \right)^3 + 3 \left(\frac{Z}{R} \right) \right) \right. \\ & + \left(\frac{15}{8} J_4 (R_\oplus/R)^4 \left(-21 \left(\frac{Z}{R} \right)^4 + 14 \left(\frac{Z}{R} \right)^2 - 1 \right) \right. \\ & + \left(\frac{21}{8} J_5 (R_\oplus/R)^5 \left(-33 \left(\frac{Z}{R} \right)^5 + 30 \left(\frac{Z}{R} \right)^3 - 5 \left(\frac{Z}{R} \right) \right) \right. \\ & + \left(\frac{7}{16} J_6 (R_\oplus/R)^6 \left(-429 \left(\frac{Z}{R} \right)^6 + 495 \left(\frac{Z}{R} \right)^4 - 135 \left(\frac{Z}{R} \right)^2 + 5 \right) \right. \\ & + \left(\frac{9}{16} J_7 (R_\oplus/R)^7 \left(-715 \left(\frac{Z}{R} \right)^7 + 1001 \left(\frac{Z}{R} \right)^5 - 385 \left(\frac{Z}{R} \right)^3 + 35 \left(\frac{Z}{R} \right) \right) \\ & \left. \left. + \left(\frac{45}{128} J_8 (R_\oplus/R)^8 \left(-2431 \left(\frac{Z}{R} \right)^8 + 4004 \left(\frac{Z}{R} \right)^6 - 2002 \left(\frac{Z}{R} \right)^4 + 308 \left(\frac{Z}{R} \right)^2 - 7 \right) \right) \right) \end{aligned} \quad (96)$$

For the derivative of the gravitational potential with respect to y we have:

$$\frac{\partial \phi}{\partial Y} = \frac{\partial \phi}{\partial X} \frac{Y}{X} \quad (97)$$

and, finally, for the derivative of the gravitational potential with respect to Z we find:

$$\begin{aligned}
\frac{\partial\phi}{\partial Z} = & -(\mu/R^2)\left(\frac{Z}{R}\right) \\
& + \left(\frac{3}{2}(J_2 + \Delta J_2)(R_\oplus/R)^2\left(-5\left(\frac{Z}{R}\right)^3 + 3\left(\frac{Z}{R}\right)\right)\right) \\
& + \left(\frac{1}{2}J_3(R_\oplus/R)^3\left(-35\left(\frac{Z}{R}\right)^4 + 30\left(\frac{Z}{R}\right)^2 - 3\right)\right) \\
& + \left(\frac{5}{8}J_4(R_\oplus/R)^4\left(-63\left(\frac{Z}{R}\right)^5 + 70\left(\frac{Z}{R}\right)^3 - 15\left(\frac{Z}{R}\right)\right)\right) \\
& + \left(\frac{3}{8}J_5(R_\oplus/R)^5\left(-231\left(\frac{Z}{R}\right)^6 + 315\left(\frac{Z}{R}\right)^4 - 105\left(\frac{Z}{R}\right)^2 + 5\right)\right) \\
& + \left(\frac{7}{16}J_6(R_\oplus/R)^6\left(-429\left(\frac{Z}{R}\right)^7 + 693\left(\frac{Z}{R}\right)^5 - 315\left(\frac{Z}{R}\right)^3 + 35\left(\frac{Z}{R}\right)\right)\right) \\
& + \left(\frac{1}{16}J_7(R_\oplus/R)^7\left(-6435\left(\frac{Z}{R}\right)^8 + 12012\left(\frac{Z}{R}\right)^6 - 6930\left(\frac{Z}{R}\right)^4 + 1260\left(\frac{Z}{R}\right)^2 - 35\right)\right) \\
& + \left(\frac{9}{128}J_8(R_\oplus/R)^8\left(-12155\left(\frac{Z}{R}\right)^9 + 25740\left(\frac{Z}{R}\right)^7 - 18018\left(\frac{Z}{R}\right)^5 + 4620\left(\frac{Z}{R}\right)^3 - 315\left(\frac{Z}{R}\right)\right)\right)
\end{aligned} \tag{98}$$

which can be simplified somewhat by recognizing that:

$$\left(\frac{X}{R}\right) = X_{0_g} = \cos(\lambda) \cos(\vartheta) \tag{99}$$

$$\left(\frac{Y}{R}\right) = Y_{0_g} = \cos(\lambda) \sin(\vartheta) \tag{100}$$

$$\left(\frac{Z}{R}\right) = Z_{0_g} = \sin(\lambda) \tag{101}$$

Finally, we take these e -Frame accelerations and transform them into the lv -Frame. To accomplish this transformation we use the following steps:

1. Take the initial e -Frame and rotate it about the z -axis by $+\vartheta$;
2. rotate next about the y -axis by $-\lambda$ (negative rotation due to the right-hand rule and the definition of positive latitude); and
3. align the resulting coordinate frame with the lv -Frame through another rotation about the y -axis of -90° .

In matrix form we have:

$$\mathbf{a}_{lv} = \begin{bmatrix} \cos(-90^\circ) & 0 & -\sin(-90^\circ) \\ 0 & 1 & 0 \\ \sin(-90^\circ) & 0 & \cos(-90^\circ) \end{bmatrix} \begin{bmatrix} \cos(-\lambda) & 0 & -\sin(-\lambda) \\ 0 & 1 & 0 \\ \sin(-\lambda) & 0 & \cos(-\lambda) \end{bmatrix} \begin{bmatrix} \cos(\vartheta) & \sin(\vartheta) & 0 \\ -\sin(\vartheta) & \cos(\vartheta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{a}_e \tag{102}$$

Multiplying these matrices together and applying simple trigonometric relationships we arrive at this form:

$$\mathbf{a}_{lv} = \begin{bmatrix} -\sin(\lambda) \cos(\vartheta) & -\sin(\lambda) \sin(\vartheta) & \cos(\lambda) \\ -\sin(\vartheta) & \cos(\vartheta) & 0 \\ -\cos(\lambda) \cos(\vartheta) & -\cos(\lambda) \sin(\vartheta) & -\sin(\lambda) \end{bmatrix} \mathbf{a}_e \tag{103}$$

With the previously defined expressions for the gravitational potential, we can substitute and simplify to form the following expressions for the gravity accelerations in the lv -Frame:

$$a_{x_{g_{lv}}} = - \left(\frac{\partial \phi}{\partial X} \cos(\vartheta) + \frac{\partial \phi}{\partial Y} \sin(\vartheta) \right) \sin(\lambda) + \frac{\partial \phi}{\partial Z} \cos(\lambda) \quad (104)$$

$$a_{y_{g_{lv}}} = - \frac{\partial \phi}{\partial X} \sin(\vartheta) + \frac{\partial \phi}{\partial Y} \cos(\vartheta) \quad (105)$$

$$a_{z_{g_{lv}}} = - \left(\frac{\partial \phi}{\partial X} \cos(\vartheta) + \frac{\partial \phi}{\partial Y} \sin(\vartheta) \right) \cos(\lambda) - \frac{\partial \phi}{\partial Z} \sin(\lambda) \quad (106)$$

The y -axis acceleration in the lv -Frame can be shown to be zero through the following expansion and substitutions:

$$\begin{aligned} a_{y_{g_{lv}}} &= - \frac{\partial \phi}{\partial X} \sin(\vartheta) + \frac{\partial \phi}{\partial Y} \cos(\vartheta) \\ &= - \frac{\partial \phi}{\partial Y} \frac{X}{Y} \sin(\vartheta) + \frac{\partial \phi}{\partial Y} \cos(\vartheta) \\ &= - \frac{\partial \phi}{\partial Y} \frac{X R}{R Y} \sin(\vartheta) + \frac{\partial \phi}{\partial Y} \cos(\vartheta) \\ &= - \frac{\partial \phi}{\partial Y} \frac{\cos(\lambda) \cos(\vartheta)}{\cos(\lambda) \sin(\vartheta)} \sin(\vartheta) + \frac{\partial \phi}{\partial Y} \cos(\vartheta) \\ &= 0 \end{aligned} \quad (107)$$

Finally:

$$a_{x_{g_{lv}}} = - \left(\frac{\partial \phi}{\partial X} \cos(\vartheta) + \frac{\partial \phi}{\partial Y} \sin(\vartheta) \right) \sin(\lambda) + \frac{\partial \phi}{\partial Z} \cos(\lambda) \quad (108)$$

$$a_{y_{g_{lv}}} = 0 \quad (109)$$

$$a_{z_{g_{lv}}} = - \left(\frac{\partial \phi}{\partial X} \cos(\vartheta) + \frac{\partial \phi}{\partial Y} \sin(\vartheta) \right) \cos(\lambda) - \frac{\partial \phi}{\partial Z} \sin(\lambda) \quad (110)$$

8 Translational Equations

Equations of motion for any system can be derived using one of several vector and energy methods. The most common vector method is Newtonian mechanics. Applying Newton's second law of motion to a point mass requires the acceleration of the particle be computed with respect to an inertial reference frame. This acceleration is typically determined by twice time-differentiating the particle's position vector with respect to a fixed point in inertial space.

When performing vector derivatives, care must be taken to account for the frame from which the derivative is taken. For example, consider a vector \mathbf{v} . The same vector may be expressed in i -Frame or b -Frame coordinates and related by the transformation $\mathbf{v}_i = C_b^i \mathbf{v}_b$. If an i -Frame time derivative is taken of this vector, the chain rule gives:

$$\frac{d^i}{dt} \mathbf{v}_i = C_b^i \dot{\mathbf{v}}_b + \dot{C}_b^i \mathbf{v}_b \quad (111)$$

A common form of the above equation is the Theorem of Coriolis, which allows a vector expressed in one reference frame to be differentiated with respect to another. The theorem is:

$$\frac{d^i}{dt} \mathbf{v}_b = \frac{d^b}{dt} \mathbf{v}_b + \boldsymbol{\omega}^{ib} \times \mathbf{v}_b \quad (112)$$

8.1 h -Frame Definition

The reference frame chosen for writing the equations of motion is a local-level h -Frame. This frame is similar to the lv -Frame (NED) with an additional rotation of angle ψ_h about down. The i -Frame, lv -Frame and h -Frame are shown in figure 3.

The h -Frame is oriented so that the vehicle velocity vector always lies in the h_1 - h_3 plane. Therefore, the h -Frame velocity is given as:

$$\mathbf{v}_h = \left\{ \begin{array}{c} U_h \\ 0 \\ W_h \end{array} \right\}_h \quad (113)$$

8.2 Inertial Position Vector

The vehicle position vector is defined from an inertially-fixed point taken to be the center of the Earth, $o = (0, 0, 0)$, to a moving point, p , which is the cg of the vehicle. This vector is easily expressed in the h -Frame as:

$$\mathbf{r}_h^{op} = \left\{ \begin{array}{c} 0 \\ 0 \\ -R \end{array} \right\}_h \quad (114)$$

Point o is considered inertial, which is an assumption valid for aircraft and rocket vehicles in suborbital flight or orbital flights of short duration.

8.3 Angular Velocity

The angular velocity of the h -Frame relative to an Earth-centered inertial frame, i -Frame, is determined by adding the angular velocities of several intermediate reference frames. This process requires repeated application of the formula:

$$\boldsymbol{\omega}^{ac} = \boldsymbol{\omega}^{ab} + \boldsymbol{\omega}^{bc} \quad (115)$$

Beginning with the i -Frame, the following series of rotations are performed:

1. Rotate at Earth rate, Ω_{\oplus} , about $+\hat{Z}_i$ and form an intermediate Earth-centered Earth-fixed e -Frame.
2. Rotate at the longitude rate, $\dot{\vartheta}$, about $+\hat{z}_e$ and form an intermediate ϑ -Frame.
3. Rotate at the latitude rate, $\dot{\lambda}$, about $-\hat{y}_{\vartheta}$ and form an intermediate up-east-north (UEN) frame.
4. Rotate at heading rate, $\dot{\psi}_h$, about $-\hat{x}_{UEN}$.

The total angular velocity of the h -Frame relative to the i -Frame expressed in mixed coordinates may then be built up as:

$$\boldsymbol{\omega}^{ih} = \Omega_{\oplus} \hat{Z}_i + \dot{\vartheta} \hat{z}_e - \dot{\lambda} \hat{y}_{\vartheta} - \dot{\psi}_h \hat{x}_{UEN} \quad (116)$$

This angular velocity may be coordinatized in the UEN frame and then transformed to h -Frame coordinates using the following transformations:

$$\hat{Z}_i = \hat{z}_e = \hat{z}_\vartheta \quad (117)$$

$$\hat{z}_\vartheta = \sin(\lambda)\hat{x}_{UEN} + \cos(\lambda)\hat{z}_{UEN} \quad (118)$$

$$\hat{y}_\vartheta = \hat{y}_{UEN} \quad (119)$$

$$C_{UEN}^h = \begin{bmatrix} 0 & \sin(\psi_h) & \cos(\psi_h) \\ 0 & \cos(\psi_h) & -\sin(\psi_h) \\ 1 & 0 & 0 \end{bmatrix} \quad (120)$$

resulting in the following total angular velocity of the h -Frame relative to inertial space expressed in h -Frame coordinates:

$$\boldsymbol{\omega}^{ih} = \left\{ \begin{array}{c} -\dot{\lambda} \sin(\psi_h) + \left(\Omega_\oplus + \dot{\vartheta} \right) \cos(\lambda) \cos(\psi_h) \\ -\dot{\lambda} \cos(\psi_h) - \left(\Omega_\oplus + \dot{\vartheta} \right) \cos(\lambda) \sin(\psi_h) \\ \dot{\psi}_h - \left(\Omega_\oplus + \dot{\vartheta} \right) \sin(\lambda) \end{array} \right\}_h \quad (121)$$

This angular velocity expression may be simplified using the facts that the h -Frame is a local-level frame with \hat{z}_h pointed towards the center of the Earth and the vehicle velocity vector always lies in the \hat{x}_h - \hat{z}_h plane. These facts lead to the transformations for the latitude rate, longitude rate, and Earth rate terms:

$$\dot{\lambda} = U_h \cos(\psi_h)/R \quad (122)$$

$$\left(\Omega_\oplus + \dot{\vartheta} \right) \cos(\lambda) = U_h \sin(\psi_h)/R \quad (123)$$

Applying these transformations leads to the simplified expression of h -Frame angular velocity with respect to the i -Frame:

$$\boldsymbol{\omega}^{ih} = \left\{ \begin{array}{c} 0 \\ -U_h/R \\ \dot{\psi}_h - \left(\Omega_\oplus + \dot{\vartheta} \right) \sin(\lambda) \end{array} \right\}_h \quad (124)$$

8.4 Inertial Acceleration

An inertial acceleration must be obtained for Newton's laws to be applied. Inertial velocity, coordinatized in the h -Frame, is determined using the following Theorem of Coriolis:

$$\mathbf{v}_h^{ip} = \frac{d^h}{dt} \mathbf{r}_h^{op} + \boldsymbol{\omega}^{ih} \times \mathbf{r}_h^{op} \quad (125)$$

This computation yields:

$$\mathbf{v}_h^{ip} = \left\{ \begin{array}{c} U_h \\ 0 \\ -\dot{R} \end{array} \right\}_h \quad (126)$$

Inertial acceleration coordinatized in the h -Frame is found by applying the Theorem of Coriolis again to the inertial velocity. It is computed to be:

$$\mathbf{a}_h^{ip} = \left\{ \begin{array}{c} \dot{U}_h + U_h \dot{R}/R \\ U_h \left(\dot{\psi}_h - \left(\Omega_\oplus + \dot{\vartheta} \right) \sin(\lambda) \right) \\ -\dot{R} + U_h^2/R \end{array} \right\}_h \quad (127)$$

8.5 Newton's Law

Newton's second law of motion may now be applied to the inertial acceleration. The sum of all forces in the h -Frame are given by the vector:

$$\sum \mathbf{F}_h = \left\{ \begin{array}{c} X_h \\ Y_h \\ Z_h \end{array} \right\}_h \quad (128)$$

Applying these forces to the system using Newton's law yields:

$$X_h/m = \dot{U}_h + U_h \dot{R}/R \quad (129)$$

$$Y_h/m = U_h \left(\dot{\psi}_h - (\Omega_{\oplus} + \dot{\vartheta}) \sin(\lambda) \right) \quad (130)$$

$$Z_h/m = -\ddot{R} + U_h^2/R \quad (131)$$

The X_h force equation may be simplified by multiplying through by R and making the substitution:

$$\frac{d}{dt} (RU_h) = \dot{R}U_h + R\dot{U}_h \quad (132)$$

The Y_h force equation may be simplified by invoking the previously used transformation:

$$\Omega_{\oplus} + \dot{\vartheta} = \frac{U_h \sin(\psi_h)}{R \cos(\lambda)} \quad (133)$$

and noticing that the term $U_h \sin(\psi_h)$ is simply V_{lv} , or the eastern component of the vehicle velocity.

The Z_h force equation is simplified using the acceleration relationship inherent in the h -Frame: $-\ddot{R} = \dot{W}_h$.

The final h -Frame translational equations of motion then become:

$$\frac{d}{dt} (RU_h) = RX_h/m = Ra_{x_h} \quad (134)$$

$$\dot{\psi}_h = V_{lv} \tan(\lambda)/R + Y_h/(mU_h) = V_{lv} \tan(\lambda)/R + a_{y_h}/U_h \quad (135)$$

$$\dot{W}_h = Z_h/m - U_h^2/R = a_{z_h} - U_h^2/R \quad (136)$$

It is worth pointing out that as the latitude approaches 90 deg, $\dot{\psi}_h$ approaches infinity. Therefore, these EOM cannot be used for polar or near-polar flights and may exhibit numerical stability problems for extreme latitudes ($\lambda > 89.9^\circ$).

There are three additional equations that describe the rate of change of latitude, longitude, and altitude. These equations were presented previously and are restated here for completeness in their simplified forms:

$$\dot{\lambda} = U_{lv}/R \quad (137)$$

$$\dot{\vartheta} = V_{lv}/(R \cos \lambda) - \Omega_{\oplus} \quad (138)$$

$$\dot{R} = -W_{lv} \quad (139)$$

where:

$$U_{lv} = U_h \cos(\psi_h) \quad (140)$$

$$V_{lv} = U_h \sin(\psi_h) \quad (141)$$

$$W_{lv} = W_h \quad (142)$$

because $R = R_{\oplus} + \Delta R$ we also have:

$$\frac{d}{dt}(\Delta R) = \dot{R} = -W_{lv} \quad (143)$$

Equation (138) has a numerical stability problem as latitude approaches 90 deg. In much the same way that $\dot{\psi}_h$ approached infinity, $V_{lv}/(R \cos \lambda)$ will also approach infinity for extreme latitudes. This observation should be fairly obvious; the lines of longitude of the Earth collapse at the poles, and, therefore, for a given U_h the longitude rate increases at extreme latitudes.

9 A Note on Rate of Change of Vehicle Mass

The astute reader will notice that there is no \dot{m} term visible in the general translational equations of motion. Newton's second law of motion states that force equals the time rate of change of momentum with respect to an inertial reference frame:

$$\mathbf{F} = \frac{d^i}{dt} \mathbf{p} = m\dot{\mathbf{V}} + \dot{m}\mathbf{V} \quad (144)$$

It is commonly argued when dealing with rocket problems that the mass flow rate of fuel is large and therefore the \dot{m} term may not be neglected as it can be in aircraft problems. This argument is incorrect because the \dot{m} term that appears in Newton's law accounts for relativistic mass changes and does not apply to fuel mass changes in the rocket itself.

For example, consider the simple example of a rocket that expels mass through its nozzle over a short time period. Figure 6 shows the initial and final momentum states of the rocket and exhaust gases. Initially, the rocket and all its fuel are moving with a constant speed and

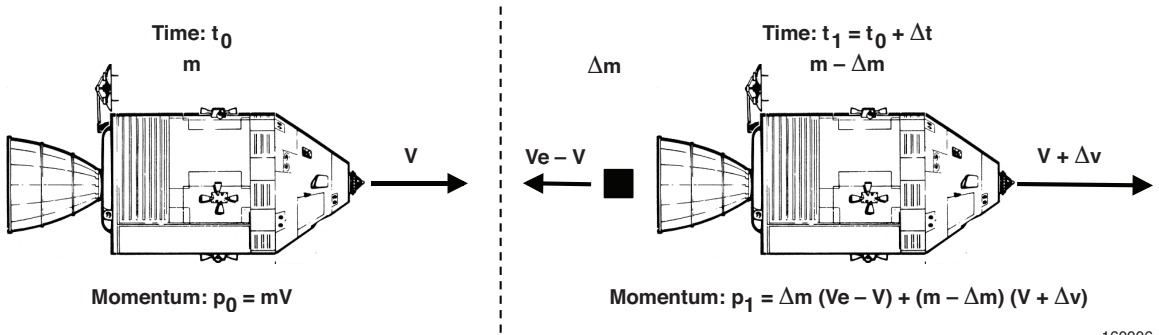


Figure 6. Simple rocket example.

have initial momentum, \mathbf{p}_0 . Over a small time period, Δt , the rocket has expelled an amount of exhaust with mass Δm at an exhaust speed of V_e relative to the rocket nozzle. The exhaust speed expressed relative to the same inertial point as the speed of the rocket is actually $V_e - V$. This action increases the speed of the rocket by ΔV and simultaneously decreases its mass by Δm . The momentum of the rocket-exhaust system in each time frame is given by:

$$\mathbf{p}_0 = mV \quad (145)$$

$$\mathbf{p}_1 = -\Delta m (V_e - V) + (V + \Delta V) (m - \Delta m) \quad (146)$$

$$\mathbf{p}_0 = \mathbf{p}_1 \quad (147)$$

Total rocket-exhaust system momentum over the time period remains constant because no mass has been allowed to enter or leave the system. Expanding the equation for the system momentum at time t_1 , and equating it with the initial momentum gives:

$$mV = -\Delta m V_e + \Delta m V + mV + m \Delta V - V \Delta m - \Delta V \Delta m \quad (148)$$

Several terms in the above equation cancel. The term $\Delta V \Delta m$ must be preserved for a finite stream of discrete impulses, but it may be dropped when considering a continuous stream of infinitesimal impulses because it becomes infinitely small with respect to the other terms. The latter case is considered here, and the resulting momentum conservation equation is:

$$\Delta m V_e = m \Delta V \quad (149)$$

The final differential equation may be found by dividing both sides by Δt and taking the limit as $\Delta t \rightarrow 0$:

$$\dot{m} V_e = m \dot{V} \quad (150)$$

This final equation contains no external force term because no external forces have been applied to the system. Newton's third law, however, does allow for equal and opposite internal forces. There are in fact equal and opposite thrust forces that accelerate the exhaust gases through the nozzle in one direction and accelerate the remaining mass of the rocket in the other. Two equations of motion result:

$$F_T = \dot{m} V_e \quad (151)$$

$$F_T = m \dot{V} \quad (152)$$

where:

$$F_T = \text{Thrust force}$$

The first equation in the above set relates the thrust force to the rate of fuel expenditure and exhaust speed. A variant of this equation would be found in an engine model where the thrust force is computed. The second equation describes the resultant motion of the rocket under the applied thrust force, and it is this equation that is used in derivation of the general simulation translational equations of motion. An \dot{m} term would only be present if the rocket or exhaust acquired or lost mass without any mass entering or leaving the closed system. A closed system, by definition, cannot admit or release mass, therefore, any mass change in the system may only occur by way of relativistic effects, which have been ignored in this analysis.

10 Rotational Equations

For rigid vehicles the rotational equations of motion are mostly simply presented in the following form (ref. 11):

$$\dot{\mathbf{L}} = \mathbf{N} - \boldsymbol{\omega} \times \mathbf{L} \quad (153)$$

but

$$\mathbf{L} = \mathcal{I}\boldsymbol{\omega} \quad (154)$$

and therefore in its simplest form we have:

$$\mathcal{I}\dot{\boldsymbol{\omega}} + \dot{\mathcal{I}}\boldsymbol{\omega} = \mathbf{N} - \boldsymbol{\omega} \times (\mathcal{I}\boldsymbol{\omega}) \quad (155)$$

If the coordinate frame is the b -Frame with its origin at the cg of the vehicle the $\dot{\mathcal{I}}$ term is identically zero for a rigid vehicle. Even if the vehicle is not rigid this term can generally be ignored if the rotation rate of the vehicle are small because of the relative size of this term with respect to the others. This term is the mechanism by which the rotational equations of motion enforce conservation of angular momentum. For the X-33 vehicle⁹, calculations show that this term is approximately one-hundred times smaller than the other pitch acceleration terms.

The fully populated inertia tensor is:

$$\mathcal{I} = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \quad (156)$$

solving for $\dot{\omega}$ and neglecting $\dot{\mathcal{I}}$ we have:

$$\mathcal{I}\dot{\omega} = (N - \omega \times I\omega) \quad (157)$$

After substitution and expansion of these vector quantities we have:

$$\mathcal{I} \begin{Bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{Bmatrix} = \left(\begin{Bmatrix} L_b \\ M_b \\ N_b \end{Bmatrix} - \begin{Bmatrix} p \\ q \\ r \end{Bmatrix} \times \mathcal{I} \begin{Bmatrix} p \\ q \\ r \end{Bmatrix} \right) \quad (158)$$

Substituting \mathcal{I} into the right-hand side of this equation:

$$\mathcal{I} \begin{Bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{Bmatrix} = \left(\begin{Bmatrix} L_b \\ M_b \\ N_b \end{Bmatrix} - \begin{Bmatrix} p \\ q \\ r \end{Bmatrix} \times \begin{Bmatrix} +I_{xx}p - I_{xy}q - I_{xz}r \\ -I_{xy}p + I_{yy}q - I_{yz}r \\ -I_{xz}p - I_{yz}q + I_{zz}r \end{Bmatrix} \right) \quad (159)$$

for simplification we can give this last vector the name \mathbf{H} . With this substitution we can expand the cross-product to arrive at:

$$\mathcal{I} \begin{Bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{Bmatrix} = \begin{Bmatrix} L_b & -qH_z & +rH_y \\ M_b & +pH_z & -rH_x \\ N_b & -pH_y & +qH_x \end{Bmatrix} = \begin{Bmatrix} T_1 \\ T_2 \\ T_3 \end{Bmatrix} \quad (160)$$

The inertia tensor has several useful properties. Chief among these is that for any *real* object, the inverse of the inertia tensor always exists (ref. 11). Thus, the inverse for \mathcal{I} can be solved using its adjoint, or matrix of cofactors, and determinant:

$$\text{adj}(\mathcal{I}) = \begin{bmatrix} I_{yy}I_{zz} - I_{yz}^2 & I_{xy}I_{zz} + I_{xz}I_{yz} & I_{xy}I_{yz} + I_{xz}I_{yy} \\ I_{xy}I_{zz} + I_{xz}I_{yz} & I_{xx}I_{zz} - I_{xz}^2 & I_{xx}I_{yz} + I_{xy}I_{xz} \\ I_{xy}I_{yz} + I_{xz}I_{yy} & I_{xx}I_{yz} + I_{xy}I_{xz} & I_{xx}I_{yy} - I_{xy}^2 \end{bmatrix} \quad (161)$$

and

$$\det \mathcal{I} = I_{xx}I_{yy}I_{zz} - I_{xx}I_{yz}^2 - I_{xy}^2I_{zz} - 2I_{xy}I_{xz}I_{yz} - I_{xz}^2I_{yy} \quad (162)$$

Thus, our original equation for the angular rates has become:

$$\mathcal{I}\dot{\omega} = \mathbf{T} \quad (163)$$

⁹The X-33 was a proposed vertically-launched research vehicle which was planned to be used to develop a single-stage-to-orbit rocket. It burned roughly eighty percent of its mass in little more than four minutes and underwent a 90-deg rotation in the pitch axis during this same period.

and after this inverse of the inertia tensor is substituted we have:

$$\dot{\boldsymbol{\omega}} = \text{adj}(\mathcal{I})\mathbf{T} / \det \mathcal{I} \quad (164)$$

solving for \dot{p} we have:

$$\dot{p} = \frac{(I_{yy}I_{zz} - I_{yz}^2)T_1 + (I_{xy}I_{zz} + I_{xz}I_{yz})T_2 + (I_{xy}I_{yz} + I_{xz}I_{yy})T_3}{\det \mathcal{I}} \quad (165)$$

Once \dot{p} is known we can substitute its value back into equation (163) and reduce the equation to:

$$\begin{bmatrix} +I_{yy} & -I_{yz} \\ -I_{yz} & +I_{zz} \end{bmatrix} \begin{Bmatrix} \dot{q} \\ \dot{r} \end{Bmatrix} = \begin{Bmatrix} T_2 + I_{xy}\dot{p} \\ T_3 + I_{xz}\dot{p} \end{Bmatrix} \quad (166)$$

We can again solve for one of the angular accelerations, this time \dot{q} , using the same techniques. The matrix equation becomes:

$$\begin{Bmatrix} \dot{q} \\ \dot{r} \end{Bmatrix} = \begin{bmatrix} +I_{zz} & +I_{yz} \\ +I_{yz} & +I_{yy} \end{bmatrix} \begin{Bmatrix} T_2 + I_{xy}\dot{p} \\ T_3 + I_{xz}\dot{p} \end{Bmatrix} / (I_{yy}I_{zz} - I_{yz}^2) \quad (167)$$

which gives us the following equation for \dot{q} :

$$\dot{q} = \frac{\dot{p}(I_{xy}I_{zz} + I_{xz}I_{yz}) + T_2I_{zz} + T_3I_{yz}}{(I_{yy}I_{zz} - I_{yz}^2)} \quad (168)$$

substituting both \dot{p} and \dot{q} yields the following equation for \dot{r} :

$$\dot{r} = (T_3 + I_{xz}\dot{p} + I_{yz}\dot{q}) / I_{zz} \quad (169)$$

10.1 Effects of Rotating Mass

In the previous section the angular momentum, L , was assumed to represent a single rigid body. For some flight vehicles this is not an accurate assumption due to rotating masses, for example, propellers and momentum wheels. The gyroscopic couples which account for the moments induced by these rotating masses can be summed up with the other external body axis moments. Adding these terms to the rotational equation of motion we have:

$$\dot{\mathbf{L}} = \mathbf{N} - \boldsymbol{\omega} \times \mathbf{L} - \boldsymbol{\omega} \times \mathbf{L}_{rm} \quad (170)$$

but

$$\boldsymbol{\omega} \times \mathbf{L}_{rm} = \begin{Bmatrix} p \\ q \\ r \end{Bmatrix} \times \begin{Bmatrix} l_x \\ l_y \\ l_z \end{Bmatrix}_{rm} = \begin{Bmatrix} ql_{z_{rm}} - rl_{y_{rm}} \\ rl_{x_{rm}} - pl_{z_{rm}} \\ pl_{y_{rm}} - ql_{x_{rm}} \end{Bmatrix} \quad (171)$$

which is the same equation as that presented in (ref. 12).

With the following assumptions:

1. The rotating mass is rotating around a principle axis with angular velocity Ωrm ;
2. The rotating mass is in stationary alignment with respect to vehicle body axes, that is, it is not allowed to freely wobble inside the vehicle; and
3. The inertia, I_{rm} , of the rotating mass is constant.

We can further reduce this gyroscopic coupling term to:

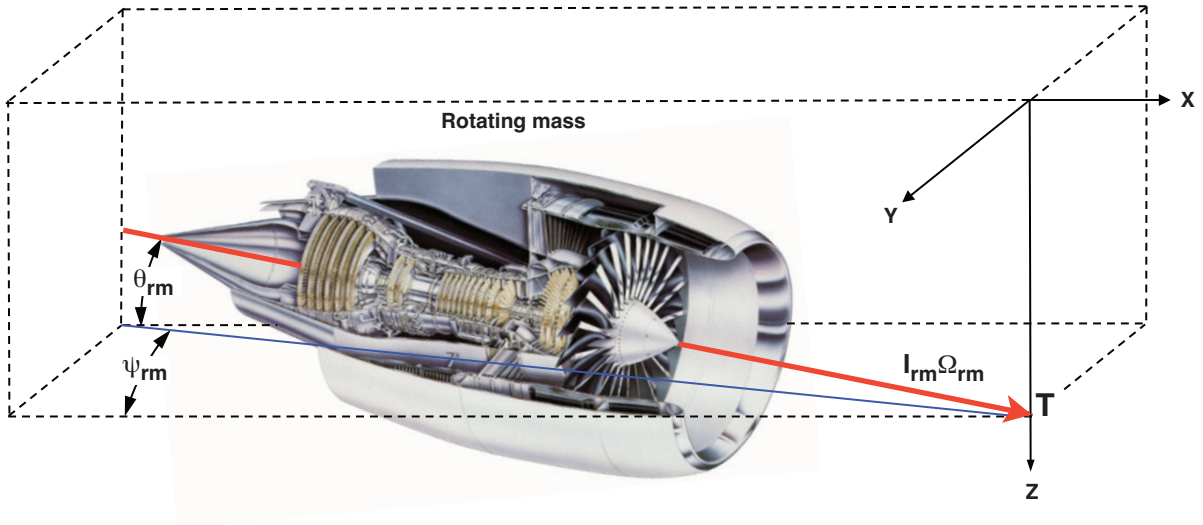
$$\boldsymbol{\omega} \times L_{rm} = \begin{Bmatrix} p \\ q \\ r \end{Bmatrix} \times \begin{Bmatrix} I_{rm}\Omega_{rm}x_{rm} \\ I_{rm}\Omega_{rm}y_{rm} \\ I_{rm}\Omega_{rm}z_{rm} \end{Bmatrix}_{rm} = I_{rm}\Omega_{rm} \begin{Bmatrix} qz_{rm} - ry_{rm} \\ rx_{rm} - pz_{rm} \\ py_{rm} - qx_{rm} \end{Bmatrix} \quad (172)$$

or if spherical coordinates are more advantageous to:

$$I_{rm}\Omega_{rm} \begin{Bmatrix} qz_{rm} - ry_{rm} \\ rx_{rm} - pz_{rm} \\ py_{rm} - qx_{rm} \end{Bmatrix} = I_{rm}\Omega_{rm} \begin{Bmatrix} q \sin(\theta_{rm}) + r \cos(\theta_{rm}) \sin(\psi_{rm}) \\ -r \cos(\theta_{rm}) \cos(\psi_{rm}) - p \sin(\theta_{rm}) \\ -p \cos(\theta_{rm}) \sin(\psi_{rm}) + q \cos(\theta_{rm}) \cos(\psi_{rm}) \end{Bmatrix} \quad (173)$$

It must be noted that this term is subtracted from the body moments, which switches the signs of all the terms (otherwise these equations are exactly the same as the equations presented in ref. 9).

Figure 7 shows how the torque due to rotating mass can be resolved into components along the vehicle body axes. It also shows the definitions of the alignment angles for the rotating mass¹⁰.



160007

Figure 7. Geometry of rotating mass with respect to the vehicle body axis.

10.2 lv -Frame Orientation

The attitude of the vehicle is described relative to the lv -Frame, but the lv -Frame constantly rotates with the vehicle position above the Earth. The angular velocity of the vehicle in this lv -Frame, $\boldsymbol{\Omega}_{lv}$, is equal to the angular velocity of the vehicle in the i -Frame, $\boldsymbol{\Omega}_i$, minus the angular rate of the lv -Frame, $\boldsymbol{\omega}_{lv}$ projected onto the vehicle body axes. Thus, the components of the angular velocity are:

$$\boldsymbol{\omega}_{lv} = \frac{V_{lv}}{R \cos(\lambda)} \vec{i}_e - \frac{U_{lv}}{R} \vec{j}_{lv} \quad (174)$$

¹⁰This figure is essentially figure 9 from ref. 9, except that the pitch angle shown is a negative angle.

but

$$\vec{i}_e = \cos(\lambda)\vec{i}_{lv} - \sin(\lambda)\vec{k}_{lv} \quad (175)$$

Thus,

$$\boldsymbol{\omega}_{lv} = \frac{V_{lv}}{R}\vec{i}_{lv} - \frac{U_{lv}}{R}\vec{j}_{lv} - V_{lv} \tan(\lambda)/R\vec{k} \quad (176)$$

subtracting this rate from the i -Frame angular velocity we have the following expression for the lv -Frame angular velocity:

$$\boldsymbol{\Omega}_{lv} = \boldsymbol{\Omega}_i - [D]\boldsymbol{\omega}_{lv} \quad (177)$$

or, after these vectors have been expanded:

$$\begin{Bmatrix} p \\ q \\ r \end{Bmatrix}_{lv} = \begin{Bmatrix} p \\ q \\ r \end{Bmatrix} - [D] \begin{Bmatrix} V/R \\ -U/R \\ -V \tan(\lambda)/R \end{Bmatrix}_{lv} \quad (178)$$

Equation (178) also has a numerical stability problem as latitude approaches 90 deg. In much the same way that $\dot{\psi}_h$ and $\dot{\vartheta}$ approach infinity, $\boldsymbol{\Omega}_{lv}$ will also approach infinity for extreme latitudes.

We now have the Euler angular rate vector of the vehicle required to compute the quaternion rates.

These Euler rates can also be used to define the time derivatives of the Euler angles. The standard definitions are used from ref. 9, which have a singularity at $\theta = \pm 90$.

$$\dot{\theta} = q_{lv} \cos(\phi) - r_{lv} \sin(\phi) \quad (179)$$

$$\dot{\psi} = (r_{lv} \cos(\phi) + q_{lv} \sin(\phi))/\cos(\theta) \quad (180)$$

$$\dot{\phi} = p_{lv} + \dot{\psi} \sin(\theta) \quad (181)$$

11 Derivatives of Derived Parameters

The time derivatives of many of the derived parameters are computed in the EOM. The derivation of these parameters will be presented in this section.

The first parameters are those related to position. From the previous definition of altitude in equation (8), yields the following definition for its derivative:

$$\dot{h} = \dot{R} - \dot{R}_{\text{local}} \quad (182)$$

but in equations (139) and (6) we defined \dot{R} and R_{local} respectively. Differentiating equation (6) with respect to time yields:

$$\begin{aligned} \dot{R}_{\text{local}} &= -\frac{(R_p \dot{\lambda} (1 - (\frac{R_p}{R_\oplus})^2) \cos(\lambda) \sin(\lambda))}{(1 - (1 - (\frac{R_p}{R_\oplus})^2) \cos^2(\lambda))^{\frac{3}{2}}} \\ &= -\frac{(R_p U_h \cos(\psi_h) e^2 \cos(\lambda) \sin(\lambda))}{R(1 - e^2 \cos^2(\lambda))^{\frac{3}{2}}} \end{aligned} \quad (183)$$

and with a little manipulation:

$$\dot{R}_{\text{local}} = -\frac{(R_{\text{local}} U_h \cos(\psi_h) e^2 \cos(\lambda) \sin(\lambda))}{R(1 - e^2 \cos^2(\lambda))} \quad (184)$$

Likewise, we can differentiate the previous expression for \dot{h} to obtain the following expression:

$$\ddot{h} = \ddot{R} - \ddot{R}_{\text{local}} \quad (185)$$

but we have previously shown:

$$\ddot{R} = U_h^2/R - a_{z_h} \quad (186)$$

Therefore, the only work left is to develop an expression for \ddot{R}_{local} . Differentiating equation (183) with respect to time we have:

$$\ddot{R}_{\text{local}} = \frac{-R_p e^2}{4} \left(\frac{2\ddot{\lambda} \sin(2\lambda) + \dot{\lambda}^2 \left(4 \cos(2\lambda) (1 - e^2 \cos^2(\lambda))^3 - 3e^2 \sin^2(2\lambda) (1 - e^2 \cos^2(\lambda))^{\frac{1}{2}} \right)}{(1 - e^2 \cos^2(\lambda))^3} \right) \quad (187)$$

where $\ddot{\lambda}$ is derived from equation (137):

$$\begin{aligned} \ddot{\lambda} &= \frac{d}{dt}(U_{lv}/R) \\ &= \frac{\dot{U}_{lv}R - \dot{R}U_{lv}}{R^2} \\ &= \frac{(a_{x_h} \cos(\psi_h)R - \dot{R}U_h \cos(\psi_h) - U_h \sin(\psi_h)\dot{\psi}_h R + W_{lv}U_{lv})}{R^2} \end{aligned} \quad (188)$$

The flat Earth ground track is generated assuming a spherical Earth with radius equal to R_{local} . Therefore, we have the following equations for x and y :

$$x = (\lambda - \lambda_0)R_{\text{local}} \quad (189)$$

$$y = (\vartheta - \vartheta_0)R_{\text{local}} \cos(\lambda) \quad (190)$$

where λ_0 and ϑ_0 represent an arbitrary reference location. The value of these flat Earth ground track parameters depends greatly upon appropriate choice of these reference longitude and latitude. In most cases, either the launch site or the landing site are good choices. The quantities \dot{x} and \dot{y} are computed through simple differentiation of these equations. Therefore:

$$\dot{x} = (\lambda - \lambda_0)\dot{R}_{\text{local}} + \dot{\lambda}R_{\text{local}} \quad (191)$$

$$\dot{y} = (\vartheta - \vartheta_0)(-R_{\text{local}} \sin(\lambda)\dot{\lambda} + \dot{R}_{\text{local}} \cos(\lambda)) + \dot{\vartheta}R_{\text{local}} \cos(\lambda) \quad (192)$$

The last parameters are those related to air mass velocities. From the definition of total speed, V , in equation (18) we derive the following definition for its derivative:

$$\dot{V} = \frac{U_{b_{RA}}\dot{U}_{b_{RA}} + V_{b_{RA}}\dot{V}_{b_{RA}} + W_{b_{RA}}\dot{W}_{b_{RA}}}{V} \quad (193)$$

Likewise for angle of attack and sideslip we have the following definitions:

$$\tan(\alpha) = W_{b_{RA}}/U_{b_{RA}} \quad (194)$$

$$\tan(\beta) = \cos(\alpha)V_{b_{RA}}/U_{b_{RA}} \quad (195)$$

which can be reduced to:

$$\tan(\alpha) = W_{b_{RA}}/U_{b_{RA}} \quad (196)$$

$$\tan(\beta) = V_{b_{RA}}/\sqrt{U_{b_{RA}}^2 + W_{b_{RA}}^2} \quad (197)$$

These expressions can be solved using the four-quadrant arctangent function, $\text{atan2}()$, but to solve for their derivatives we must use the $\arctan()$ function, which yields the following explicit equations:

$$\alpha = \arctan(W_{b_{RA}}/U_{b_{RA}}) \quad (198)$$

$$\beta = \arctan(V_{b_{RA}}/\sqrt{U_{b_{RA}}^2 + W_{b_{RA}}^2}) \quad (199)$$

Differentiating the expression for α yields the following:

$$\dot{\alpha} = \frac{U_{b_{RA}}\dot{W}_{b_{RA}} - \dot{U}_{b_{RA}}W_{b_{RA}}}{U_{b_{RA}}^2 + W_{b_{RA}}^2} \quad (200)$$

and for β :

$$\dot{\beta} = \frac{\dot{V}_{b_{RA}}(U_{b_{RA}}^2 + W_{b_{RA}}^2) - V_{b_{RA}}(U_{b_{RA}}\dot{U}_{b_{RA}} + W_{b_{RA}}\dot{W}_{b_{RA}})}{V^2\sqrt{U_{b_{RA}}^2 + W_{b_{RA}}^2}} \quad (201)$$

We now find that we need the derivatives of the component velocities in the b -Frame. To obtain these we start with the derivatives of the states and work our way through the coordinate transformations. Using the previous definition of $\frac{d}{dt}(RU_h)$ and solving for \dot{U}_h we have:

$$\dot{U}_h = \frac{\frac{d}{dt}(RU_h) - \dot{R}U_h}{R} = a_{x_h} + \frac{W_{lv}U_h}{R} \quad (202)$$

From the previous definitions of U_{lv} , V_{lv} , and W_{lv} we can solve for their derivatives:

$$\dot{U}_{lv} = -U_h \sin(\psi_h)\dot{\psi}_h + \dot{U}_h \cos(\psi_h) \quad (203)$$

$$\dot{V}_{lv} = U_h \cos(\psi_h)\dot{\psi}_h + \dot{U}_h \sin(\psi_h) \quad (204)$$

$$\dot{W}_{lv} = \dot{W}_h \quad (205)$$

By applying the chain rule we can differentiate equations (13), (14), and (15) to arrive at the following expressions for the airmass relative velocities:

$$\dot{U}_{lv_{RA}} = \dot{U}_{lv} - \dot{X}_w \quad (206)$$

$$\dot{V}_{lv_{RA}} = \dot{V}_{lv} - \dot{Y}_w + \Omega_{\oplus} \left(R \sin(\lambda)\dot{\lambda} - \dot{R} \cos(\lambda) \right) \quad (207)$$

$$\dot{W}_{lv_{RA}} = \dot{W}_{lv} - \dot{Z}_w \quad (208)$$

where the wind components are computed through simple differentiation:

$$\dot{X}_w \approx (X_w - X_{w_{\text{last}}})/\Delta t \quad (209)$$

$$\dot{Y}_w \approx (Y_w - Y_{w_{\text{last}}})/\Delta t \quad (210)$$

$$\dot{Z}_w \approx (Z_w - Z_{w_{\text{last}}})/\Delta t \quad (211)$$

Ignoring the gust contributions (because of the purely random zero mean nature of gusts) in the transformation from lv -Frame to b -Frame we arrive at the following:

$$\dot{\mathbf{V}}_{b_{RA}} = \frac{d}{dt}[D]\mathbf{V}_{lv_{RA}} + [D]\dot{\mathbf{V}}_{lv_{RA}} \quad (212)$$

where:

$$\frac{d}{dt}[D] = 2 \begin{bmatrix} e_0\dot{e}_0 + e_1\dot{e}_1 - e_2\dot{e}_2 - e_3\dot{e}_3 & e_1\dot{e}_2 + e_2\dot{e}_1 + e_0\dot{e}_3 + e_3\dot{e}_0 & e_1\dot{e}_3 + e_3\dot{e}_1 - e_0\dot{e}_2 - e_2\dot{e}_0 \\ e_1\dot{e}_2 + e_2\dot{e}_1 - e_0\dot{e}_3 - e_3\dot{e}_0 & e_0\dot{e}_0 - e_1\dot{e}_1 + e_2\dot{e}_2 - e_3\dot{e}_3 & e_2\dot{e}_3 + e_3\dot{e}_2 + e_0\dot{e}_1 + e_1\dot{e}_0 \\ e_1\dot{e}_3 + e_3\dot{e}_1 + e_0\dot{e}_2 + e_2\dot{e}_0 & e_2\dot{e}_3 + e_3\dot{e}_2 - e_0\dot{e}_1 - e_1\dot{e}_0 & e_0\dot{e}_0 - e_1\dot{e}_1 - e_2\dot{e}_2 + e_3\dot{e}_3 \end{bmatrix} \quad (213)$$

12 Background for Integration Algorithm

Throughout section 12, Background for Integration Algorithm, the equation symbology follows that used reference 13. In particular, chapter 8 of reference 13 presents analysis of several integration algorithms. This section summarizes many of the equations developed in that reference. These background integration algorithms are presented in simple scalar form. All of these equations can be extended to the vector formulation. Please refer to reference 13 for any assistance with understanding the symbology presented in the equations presented below.

Consider the standard form for a single simple first-order differential equation:

$$y' = f(x, y) \quad (214)$$

$$y(x_0) = y_0 \quad (215)$$

Most of the techniques used for numerical integration rely in some part upon a Taylor series of $y(x)$ about a point $x = x_0$.

$$y(x) = y_0 + (x - x_0)y'(x_0) + \frac{(x - x_0)^2}{2!}y''(x_0) + \dots \quad (216)$$

where

$$y' = f(x, y) \quad (217)$$

$$y'' = f' = f_x + f_y y' = f_x + f_y f \quad (218)$$

$$\begin{aligned} y''' = f'' &= f_{xx} + f_{xy}f + f_{yx}f + f_{yy}f^2 + f_y f_x + f_y^2 f \\ &= f_{xx} + 2f_{xy}f + f_{yy}f^2 + f_x f_y + f_y^2 f \end{aligned} \quad (219)$$

This Taylor series must be truncated at a finite number of terms, but the resulting truncated series must only be a good approximation over a step of $h = (x - x_0)$. After this step has been taken, the derivatives are reevaluated about the point $x_0 + h$ and additional steps can be taken in like manner.

This Taylor series algorithm leads to the following integration method and local error:

$$y_{n+1} = y_n + hT_k(x_n, y_n) \quad (220)$$

where

$$T_k(x, y) = f(x, y) + \frac{h}{2!}f'(x, y) + \dots + \frac{h^{k-1}}{k!}f^{(k-1)}(x, y) \quad (221)$$

$$\begin{aligned} E &= \frac{h^{k+1}f^{(k)}(\xi, y(\xi))}{(k+1)!} \\ &= \frac{h^{k+1}y^{(k+1)}(\xi)}{(k+1)!} \end{aligned} \quad (222)$$

where

$$k = 1, 2, \dots$$

and

$$x_n < \xi < x_n + h$$

These methods of integration, which require only information about the function y and its derivatives at a single point $x = x_n$, are called one-step methods. Among the simplest one-step integration techniques is Euler's method, which is Taylor's algorithm with $k = 1$. The following equations show the method and its local error:

$$y_{n+1} = y_n + hf(x_n, y_n) \quad (223)$$

$$E = \frac{h^2}{2} y''(\xi_n) \quad (224)$$

Several techniques exist to perform numerical integration with increased accuracy at a cost of additional derivative calculations or increased storage requirements for past states and past derivatives. These include both Runge-Kutta and multistep integration techniques.

The general formula for second-order Runge-Kutta integration:

$$y_{n+1} = y_n + ak_1 + bk_2 \quad (225)$$

$$k_1 = hf(x_n, y_n) \quad (226)$$

$$k_2 = hf(x_n + \alpha h, y_n + \beta k_1) \quad (227)$$

where a , b , α , and β are constants to be determined such that the equation (225) agrees with a Taylor series integration of as high an order as possible.

Expanding $y(x_{n+1})$ in a Taylor series through order h^3 we obtain the following:

$$y(x_{n+1}) = y(x_n) + hy'(x_n) + \frac{h^2}{2} y''(x_n) + \frac{h^3}{6} y'''(x_n) + \dots \quad (228)$$

$$\begin{aligned} &= y(x_n) + hf(x_n, y_n) + \frac{h^2}{2} (f_x + ff_y)_n \\ &\quad + \frac{h^3}{6} (f_{xx} + 2ff_{xy} + f_{yy}f^2 + f_x f_y + f_y^2 f)_n + \mathcal{O}(h^4) \end{aligned} \quad (229)$$

If we perform a Taylor expansion for functions of two variables on equation (227) we obtain the following expression for k_2/h :

$$\frac{k_2}{h} = f(x_n + \alpha h, y_n + \beta k_1) \quad (230)$$

$$\begin{aligned} &= f(x_n, y_n) + \alpha h f_x + \beta k_1 f_y + \frac{\alpha^2 h^2}{2} f_{xx} \\ &\quad + \alpha h \beta k_1 f_{xy} + \frac{\beta^2 k_1^2}{2} f_{yy} + \mathcal{O}(h^3) \end{aligned} \quad (231)$$

Substituting equations (226) and (231) into equation (225) and simplifying the notation slightly, we obtain the following:

$$\begin{aligned} y_{n+1} &= y_n + (a+b)hf + bh^2(\alpha f_x + \beta f f_y) \\ &\quad + bh^3 \left(\frac{\alpha^2}{2} f_{xx} + \alpha \beta f f_{xy} + \frac{\beta^2}{2} f^2 f_{yy} \right) + \mathcal{O}(h^4) \end{aligned} \quad (232)$$

For a second-order Runge-Kutta integration we find that we can make both the first- and second-order terms of h match by requiring:

$$a + b = 1 \quad (233)$$

$$b\alpha = b\beta = \frac{1}{2} \quad (234)$$

Since there are three equations with four unknowns we have many choices for these terms. A non-typical choice¹¹ of parameters are:

$$\begin{aligned} a &= 0 \\ b &= 1 \\ \alpha &= \frac{1}{2} \\ \beta &= \frac{1}{2} \end{aligned}$$

The resulting integration equation is:

$$y_{n+1} = y_n + k_2 \quad (235)$$

with

$$k_1 = hf(x_n, y_n) \quad (236)$$

$$k_2 = hf\left(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1\right) \quad (237)$$

The local error associated with this integration algorithm is¹²:

$$E = y(x_{n+1}) - y_{n+1} = \frac{h^3}{24}(f_{xx} + 2ff_{xy} + f^2f_{yy} + 4f_xf_y + 4ff_y^2) + \mathcal{O}(h^4) \quad (238)$$

Another series of more complex integration techniques are the multistep methods. The following equation shows the basis of these methods in general:

$$y_{n+1} = y_n + \int_{x_n}^{x_{n+1}} f(x, y(x))dx \quad (239)$$

If we introduce the notation

$$f(x_k, y(x_k)) = f_k$$

we can combine the Newton backward formula for polynomial interpolation and arrive at the following formula (known as the Adams-Bashforth formula):

$$y_{n+1} = y_n + h \int_0^1 \sum_{k=0}^m (-1)^k \binom{-s}{k} \Delta^k f_{n-k} ds \quad (240)$$

$$= y_n + h\{\gamma_0 f_n + \gamma_1 \Delta f_{n-1} + \dots + \gamma_m \Delta^m f_{n-m}\} \quad (241)$$

where

$$s = \frac{x - x_n}{h} \quad (242)$$

$$\Delta^i f_s = \begin{cases} f_s & i = 0 \\ \Delta(\Delta^{i-1} f_s) = \Delta^{i-1} f_{s+1} - \Delta^{i-1} f_s & i > 0 \end{cases} \quad (243)$$

¹¹The usefulness of this choice will become evident in the later discussion of the algorithm used in the AFRC simulation code.

¹²This equation differs from that presented in (ref. 13) because of the unconventional choice of a, b, α , and, β .

$$\binom{y}{i} = \begin{cases} 1 & i = 0 \\ \prod_{j=0}^{i-1} \frac{y-j}{j+1} = \frac{(y)(y-1)\dots(y-i+1)}{1\cdot 2\cdots i} & i > 0 \end{cases} \quad (244)$$

$$\gamma_k = (-1)^k \int_0^1 \binom{-s}{k} ds \quad (245)$$

A more general case is to consider that $f(x, y)$ can be integrated from x_{n-p} to x_{n+1} for some $p \geq 0$. Two of those special cases are presented below.

The first case takes $m = 1$ and $p = 1$:

$$y_{n+1} = y_{n-1} + 2hf_n \quad (246)$$

$$E = \frac{h^3}{3} y'''(\xi) \quad (247)$$

The second case, referred in the literature as Milne's formula, takes $m = 3$ and $p = 3$:

$$y_{n+1} = y_{n-3} + \frac{4h}{3}(2f_n - f_{n-1} + 2f_{n-2}) \quad (248)$$

$$E = \frac{14}{45} h^5 y^v(\xi) \quad (249)$$

13 Simulation Integration Algorithm

We have now finally developed enough background to describe the algorithm used in the AFRC simulation. The current algorithm uses a modification to the second-order Runge-Kutta technique presented earlier.

The notational translation of the Runge-Kutta algorithm yields:

$$x_{n+1} = x_n + k_2 \quad (250)$$

with

$$k_1 = \Delta t f(t_n, x_n) \quad (251)$$

$$k_2 = \Delta t f\left(t_n + \frac{1}{2}\Delta t, x_n + \frac{1}{2}k_1\right) \quad (252)$$

Step 1. Compute k_1 using the estimate of the derivative saved from the previous integration step.

$$k_1 = \Delta t f(t_n, x_n) \quad (253)$$

Step 2. Compute k_2 (at the mid-integration step) using a call to `DERIVC` to evaluate the function $f(t, x)$.

$$t_{n+\frac{1}{2}} = t_n + \frac{1}{2}\Delta t \quad (254)$$

$$x_{n+\frac{1}{2}} = x_n + \frac{1}{2}k_1 \quad (255)$$

$$k_2 = \Delta t f\left(t_{n+\frac{1}{2}}, x_{n+\frac{1}{2}}\right) \quad (256)$$

Step 3. Integrate over the full integration step.

$$x_{n+1} = x_n + k_2 \quad (257)$$

Step 4. Extrapolate the derivative computed at the estimated mid-integration step¹³ to the end-of-integration-step using one-half of the change in the first part of the integration step.

$$f_{n+1} = f_{n+\frac{1}{2}} + \frac{1}{2}(f_{n+\frac{1}{2}} - f_n) \quad (258)$$

$$= \frac{3}{2}f_{n+\frac{1}{2}} - \frac{1}{2}f_n \quad (259)$$

Step 5. Extrapolate needed parameters which have been computed at the mid-integration step to the end-of-integration-step boundary.

14 Oblate Earth Equations of Motion Considerations

Many of the states and parameters used in the AFRC simulation have limitations placed on their valid range. The definition of each of these states and parameters were provided previously in Section 3.

Time, t , has a valid range of $[0 : +\infty]$. It is initialized to 0.0 and can never become negative since its derivative, \dot{t} , is always 1.0.

Roll, pitch, and yaw rates, p , q , and r , have a valid range of $[-\infty : +\infty]$. No special attention is therefore required of them. Ridiculously high angular rates would, however, require an extremely small integration time.

The next parameters are RU_h and ψ_h . RU_h is the product of radius from the center of the Earth to the vehicle and its total horizontal speed in the h-frame. The valid range for this parameter is $[0 : +\infty]$. ψ_h is the direction of the U_h speed in the h-frame, hence its valid range is $[-\pi : +\pi]$. Since radius is always positive and cannot even take a value near zero¹⁴ and U_h is always positive, since it is the magnitude of the horizontal speed vector¹⁵. Since U_h and ψ_h are a polar representation of the velocities, x_{lv} and y_{lv} , care must be taken in the integration to assure that RU_h and ψ_h maintain the proper relationship. The variable RU_h shall be tested for becoming negative and if a negative value is detected the variable shall be made positive and π shall be added to ψ_h . Then ψ_h shall be limited to its valid range by adding $\pm 2\pi$ (whichever value is required to bring it into the valid range).

Vehicle vertical speed in the h-frame, w_h , has a valid range of $[-\infty : +\infty]$. No special attention is required.

¹³This is the modification step. To follow the second-order Runge-Kutta algorithm exactly, this extrapolation would be replaced by a second call to `DERIVC`. The choice of how to estimate the end-of-integration-step derivative from the start-of-integration-step and mid-integration-step derivative is completely arbitrary.

¹⁴Contrary to several popularized fictionalized accounts about journeys to the center of the Earth, the AFRC simulation does not believe that the Earth is hollow. Therefore, the smallest value of R is essentially R_p .

¹⁵ U_h can go to or through zero if the vehicle flies a trajectory which builds up speed in the westerly direction and cancels the speed of the rotation of the Earth. As a practical matter, U_h will never be exactly zero because even the slightest deviation in heading will create either a small northern or southern component speed. Simulation runs used in the Hyper-X batch simulation have proved this conjecture wrong; if you try hard enough you can confuse the AFRC simulation by flying directly west with no atmospheric disturbances. There is no simple fix, and the only solution is to change the initial conditions to prevent the degenerate condition from occurring; thus, don't intentionally fly directly west.

The next set of variables is the quaternion components, e_0 , e_1 , e_2 , and e_3 . These components each have the range of $[-1 : +1]$. To assure that the quaternion remains valid it is normalized after each integration step. The normalization step divides each element of the quaternion, e_i , by the quaternion magnitude, $\sqrt{e_0^2 + e_1^2 + e_2^2 + e_3^2}$.

The next state, ΔR , is the height of the vehicle above a spherical Earth with radius R_\oplus . It has a usable range of $[-(R_\oplus - R_p) : +\infty]$. No special care is required to keep this variable within its valid range.

The geocentric latitude of the vehicle, λ , has a valid range of $(-\pi/2 : +\pi/2)$ ¹⁶. If during the integration step it is determined that the parameter has exceeded this range it must be reflected back from the limit and 2π must be added to ϑ_{gc} . This treatment is analogous to the treatment that θ and ψ are given as θ passes through $\pm\pi/2$.

The geocentric longitude of the vehicle, ϑ_{gc} , has a range of validity of $[-\pi : +\pi]$. If ϑ_{gc} exceeds this range 2π shall be added or subtracted (whichever value is required to bring it into the valid range).

The airmass velocities of the vehicle, V , α , and β , have a valid range of $[0 : +\infty]$, $[-\pi : +\pi]$, and $[-\pi : +\pi]$ respectively. These velocities are computed parameters at the mid-point of the integration frame for use in the equations of motion and recomputed at the end-point. They are computed from the component body axis velocities relative to the moving airmass. Each parameter is computed in such a manner to assure that it stays within the valid range. Therefore no special care is required to keep these parameters within their proper ranges. Additional constraints may be placed on the angle of attack or sideslip due to limitations of the wind tunnels tests used in creation of the aerodynamic model. Because of this the simulation has a program control display page where the upper and lower limit for α can be controlled. In a similar manner the maximum limit for β can be set¹⁷. Exceeding any of these limits causes the simulation to enter the HOLD state.

The Euler angles, θ , ϕ , and ψ , have ranges of $[-\pi/2 : +\pi/2]$, $[-\pi : +\pi]$, and $[-\pi : +\pi]$ respectively. Since these parameters are derived at the end-point of the integration step from the quaternion elements they will remain within the appropriate range.

The altitude of the vehicle, h , has a valid range of $[0m : +1,000,000m]$ due to implementation of the 1976 U.S. Standard Atmosphere. The actual limitations of the 1976 U.S. Standard Atmosphere, if fully implemented, would allow a larger range of $[-5,000m : 1,000,000m]$. Other atmospheric models may have different lower or upper limits. The program control page allows the user to set a lower limit on altitude (below which the simulation enters the HOLD state).

The next variables approximate flat Earth coordinates around a local reference point. These variables are x and y . They have a valid range of $[-\infty : +\infty]$, but the approximation that the surface of the Earth is approximately a flat plane will have substantial errors long before either x or y reaches ∞ .

The parameter V_i is the total inertial speed of the vehicle and has a valid range of $[0 : +\infty]$. Since it is computed from U_h and W_h it has no special considerations or requirements to keep it within this range.

The geocentric and geodetic latitude parameters λ_{gc} and λ_{gd} have a valid range of $[-\pi/2 : +\pi/2]$. Because these parameters are computed directly from the vehicle's state using trigonometric functions there are no special considerations required to keep them within their proper ranges.

Likewise the geodetic altitude, h_{gd} , is a computed parameter and requires no special consid-

¹⁶To exclude the possibility of simulation problems with $\dot{\psi}_h$ the valid range is restricted to $[-89.9^\circ : +89.9^\circ]$.

¹⁷It is assumed that sideslip limits are symmetric.

eration to stay within its approximate range of $[-(R_{\oplus} - R_p) : +\infty]$.

The last parameters are the angular rates of the vehicle relative to the local horizon. These Euler rate parameters, p_{lv} , q_{lv} , and r_{lv} , have a useful range of $[-\infty : +\infty]$. They require no special consideration.

15 Concluding Remarks

The EOM shown in this report are general, neglecting only terms that are small, and precluding polar or near-polar trajectories¹⁸. Two major assumptions were made:

- The center of the Earth is the origin of an inertially fixed reference frame, i -Frame.
- The $\dot{\mathcal{I}}\omega$ term in the rotational EOM is small and can be neglected.

The simplicity of the EOM described in this report arises from the choice of the h -Frame.

The legacy NASA Armstrong Flight Research Center (AFRC) oblate Earth nonlinear simulation applies these translational, rotational, and helper equations in the routine `DERIVC`.

The algorithm used in the AFRC simulation has been presented. It is based upon a modification to the second-order Runge-Kutta integration method. The modification has been made to eliminate a second call to the routine `DERIVC`, which can be very expensive in terms of computation time. Additional algorithms that also do not require additional calls to `DERIVC` have also been presented.

The typical problems associated with startup and storage of these multistep algorithms are not significant for this application. Seeding the past derivatives with either Runge-Kutta or Euler methods while in `RESET` will eliminate the startup concern. The limited number of equations to integrate eliminates concern regarding storage requirements. Of more concern is the numerical stability of these integration methods. Runge-Kutta methods do not exhibit numerical instability if the time-step is sufficiently small, while multistep methods may be unstable for any value of the time-step.

Considerations of ranges for variables have also been presented. Limit functions and normalization routines should be used to assure that the integration states are maintained in their proper range.

¹⁸Care should be taken to look for numerical problems if the trajectory exceeds a geocentric latitude of 89.9° deg.

References

1. Schilling, Lawrence, "Oblate Earth EOM," NASA AFRC Internal Notes, May 7, 1985 (unpublished).
2. Fogarty, L.E. and R.M. Howe, "Flight Simulation of Orbital and Re-Entry Vehicles", *IRE Transactions on Electronic Computers*, Vol. EC-11, Issue 4, pp. 555-563.
3. Hedgley, David R., Jr., *An Exact Transformation from Geocentric to Geodetic Coordinates for Non-Zero Altitudes*, NASA TR R-458, 1976.
4. Haering, Edward A., Jr. and Stephen A. Whitmore, *FORTTRAN Program for Analyzing Ground-Based Radar Data: Usage and Derivations, Version 6.2*, NASA TP 3430, 1995.
5. Zhu, Jijie, "Exact Conversion of Earth-Centered, Earth-Fixed Coordinates to Geodetic Coordinates", *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 2, pp. 389-391.
6. DMA WGS 84 Development Committee, "Department of Defense World Geodetic System 1984 - Its Definition and Relationships with Local Geodetic Systems", DMA TR 8350.2, 1991.
7. National Oceanic and Atmospheric Administration, National Aeronautics and Space Administration, United States Air Force, "U.S. Standard Atmosphere, 1976", NOAA-S/T 76-1562, 1976.
8. Bryson, Arthur E., Jr. and Yu-Chi Ho, *APPLIED OPTIMAL CONTROL, OPTIMIZATION, ESTIMATION, AND CONTROL*, Hemisphere Publishing Corporation, Washington, D.C., 1975.
9. Gainer, Thomas G. and Sherwood Hoffman, *Summary of Transformation Equations and Equations of Motion Used in Free-Flight and Wind-Tunnel Data Reduction and Analysis*, NASA SP 3070, 1972.
10. Anonymous, "American National Standard – Recommended Practice for Atmospheric and Space Flight Vehicle Coordinate Systems", ANSI/AIAA R-004-1992, 1992.
11. Wertz, James R. ed., *Spacecraft Attitude Determination and Control*, Kluwer Academic Publishers, The Netherlands, 1978.
12. Etkin, Bernard, *DYNAMICS OF FLIGHT, STABILITY AND CONTROL*, John Wiley & Sons, Inc., London, 1963.
13. Conte, Samuel Daniel and Carl de Boor, *Elementary Numerical Analysis*, McGraw-Hill, Inc., 1980.
