ROGERS ENGINEERING & ASSOCIATES Technical Report 89-1

Final Report

for

SPACE SHUTTLE PROPULSION ESTIMATION DEVELOPMENT VERIFICATION

Volume I

by

Robert M. Rogers

Prepared under:

Contract NAS8-36152

for

GEORGE C. MARSHALL SPACE FLIGHT CENTER

Marshall Space Flight Center, Alabama 35812

28 February 1989

Abstract

The work described in this report details the application of extended Kalman filtering to estimating the Space Shuttle Propulsion performance, i.e. specific impulse, from flight data in a post-flight processing computer program. The flight data used includes inertial platform acceleration, SRB head pressure, SSME chamber pressure and flow rates, and ground based radar tracking data. The key feature in this application is the model used for the SRB's, which is a nominal or reference quasi-static internal ballistics model normalized to the propellant burn depth. Dynamic states of mass overboard and propellant burn depth are included in the filter model to account for real-time deviations from the reference model used. Aerodynamic, plume, wind and main engine uncertainties are also included for an integrated system model. Assuming uncertainty within the propulsion system model and attempts to estimate its deviations represent a new application of parameter estimation for rocket powered vehicles. Illustrations from the results of applying this estimation approach to several missions show good quality propulsion estimates.

Table of Contents

Table of Contents (cont'd)

Table of Contents (concluded)

List of Figures

List of Tables

1. Introduction

Volume I of this technical report summarizes the results of the Propulsion Estimation Development Verification performed under contract NAS8-36152 from March 1985 through January 1989. During this period, a computer program developed under a previous contract, NAS8-35324 [1][1], was modified to include improved models for the Solid Rocket Booster (SRB) internal ballistics, the Space Shuttle Main Engine (SSME) power coefficient model, the vehicle dynamics using quaternions, and an improved Kalman filter algorithm based on the U-D factorized algorithm. As additional output, the estimated propulsion performances, Isp's, for each device are computed with the associated 1-sigma bounds. The outputs of the estimation program are provided in graphical plots.

Since the previous program developments were based on using synthetic data, this verification effort uses real data. To accomplish this verification with real data, several additional programs were required to convert the real test data into data compatible with the usage in the estimation programs. In all, there are six additional preprocessing programs developed that also provide graphical output plots of the data for visual review prior to use. This preprocessing includes converting main engine pressurant volumetric flows into mass flows, units conversion for

[1]* numbers in brackets refer to the corresponding reference located in the REFERENCES

meteorological data, extracting selected ground radar tracking data from those available on the recorded magnetic tape, and providing preprocessed vehicle accelerations, attitudes and rates for later use in the propulsion and BET estimation programs.

The results obtained from the propulsion estimation programs are consistent with those obtained by using alternative methods. However, this process is an integrated methodology based on Kalman filtering theory.

An additional effort was expended to examine the use of the estimation approach to evaluating single engine test data. This approach used the model for the SSME used in the PFILTER program. The same Kalman filtering algorithms were also used. The results from this effort [2] showed that the linear algebraic power gain law model exhibits little sensitivity to model deviations and possesses little ability to correct for these model errors. Additionally, this model was used in an attempt to extract modeling parameters, coefficients of the power gain model. This effort also demonstrated little ability to estimate these parameters due to the limited measurements assumed available. These results used a single engine's flight data since ground data was not available.

In addition to the propulsion estimation program PFILTER, a program has been developed to produce a best estimate of trajectory (BET). The program, LFILTER, also uses the U-D factorized algorithm form of the Kalman filter as in the propulsion estimation program PFILTER. Several of the library routines are common between the programs. The BET system model is that of an inertially stabilized inertial measurement unit and the model is a linear dynamical system with additive system errors as contrasted with the nonlinear, equations of motion, dynamical model used in the PFILTER program. The PFILTER program is based on the extended Kalman filtering approach, while the LFILTER is based on a linearized Kalman filter. The BET program also uses a smoothing routine based on the Rauch-Tung-Striebel (RTS) algorithm. This smoothing routine improves the BET results by eliminating the filter induced time shift or lag, and by improving the state estimates and reducing the associated uncertainties for those estimates. This BET program is also used and a preprocessing program to improve the quality of some of the inputs used in the propulsion estimation program.

The necessary definitions and equations explaining the Kalman filtering approach for the PFILTER program are presented in section 2. The models used for this application are reviewed in section 3 for the dynamics and measurements. The program description is presented in section 4. The program operation is presented in section 5. Conclusions and recommendations are presented in section 6.

The preprocessing programs are described in Appendix A. The program which produces the output plots is described in Appendix B. Appendix C describes the BET LFILTER and RTSSMO filtering/smoothing programs respectively. Appendix D describes the library routines used for matrix manipulations and coordinate conversions. Appendix E contains the definition of coordinate frames used in these programs.

Volume II contains the program listings.

2. Kalman Filtering Equations/State Elements

The program PFILTER is based on the structure and operation of a Kalman filter. Within this structure and operation, a system is assumed to be described by a dynamical process. This process may also be "driven" by additive disturbances (noise), and by known inputs (controls). This process can be formulated into a state vector model given as

$$\dot{\underline{x}}(t) = \underline{f}(\underline{x},\underline{u},t) + \underline{w}(t) \tag{1}$$

where the dynamics of the state vector, $\underline{x}$, is a nonlinear function of itself, the control inputs, $\underline{u}$, and time, t. The additive disturbance vector, $\underline{w}(t)$, is assumed to be a zero mean uncorrelated gaussian noise process. If this disturbance were non-zero mean, additional elements are added to the state vector, $\underline{x}$, to accommodate biases associated with the non-zero mean disturbances. These disturbances' statistical properties are described by

$$E[\underline{w}(t)] = 0; \qquad Cov[\underline{w}(t)\underline{w}^T(\tau)] = Q(t)\delta(t-\tau) \tag{2}$$

where E[] and Cov[] denote the expected value (mean) and covariance respectively, and the superscript T denoted the matrix transpose operation. Since the assumption of gaussian processes is made, the properties in (2) are sufficient to describe the disturbances' statistical properties.

At discrete time instances, t_k, measurements are available that are functionally related to the state vector, $\underline{x}$. In general, these measurements can be represented by a nonlinear function of the state vector. These measurements are imperfect and are assumed to be corrupted by additive disturbances (noise). These disturbances are also assumed to zero mean uncorrelated gaussian sequences. The measurements are represented by

$$\underline{z}_k = \underline{h}[\underline{x}(t_k)] + \underline{v}_k; \; k = 1,2,... \qquad (3)$$

The statistical properties of the corrupting noise, $\underline{v}_k$, are described by

$$E[\underline{v}_k] = 0; \quad Cov[\underline{v}_k\underline{v}_j^T] = R_k\delta_{kj}. \qquad (4)$$

These equations and the extended Kalman filter algorithms are summarized in Table 2.-1 from [3]. In addition to the process and measurement noise definitions in (2) and (4), the initial state estimates are assumed to be governed by a gaussian distribution and the process noise (2) and measurement noise (4) are independent.

Table 2.-1

Extended Kalman Filter Algorithm
(continuous time - discrete measurements)

System Model	$\dot{\underline{x}} = \underline{f}[\underline{x}(t),\underline{u},t] + \underline{w}(t)$; $\underline{w}(t) = N[0,Q(t)]$
Measurement Model	$\underline{z}_k = \underline{h}_k[\underline{x}(t_k)] + \underline{v}_k$; $k = 1,2,\ldots$; $\underline{v}_k = N[0,R_k]$
Initial Conditions	$\underline{x}(0) = N[\underline{x}_o,P_o]$
Other Assumptions	$E[\underline{w}(t)\underline{v}_k^T] = 0$ for all t and k
State Estimate Propagation	$\dot{\hat{\underline{x}}} = \underline{f}[\hat{\underline{x}}(t),\underline{u},t]$
Error Covariance Propagation	$\dot{P}(t) = F[\hat{\underline{x}}(t),t]P(t) + P(t)F[\hat{\underline{x}}(t),t]^T + Q(t)$
State Estimate Update	$\hat{\underline{x}}_k(+) = \hat{\underline{x}}_k(-) + K_k\{\underline{z}_k - \underline{h}_k[\hat{\underline{x}}_k(-)]\}$
Error Covariance Update	$P_k(+) = \{I - K_kH_k[\hat{\underline{x}}_k(-)]\}P_k(-)$
Gain Matrix	$K_k = P_k(-)H_k[\hat{\underline{x}}(-)]^T\{H_k[\hat{\underline{x}}_k(-)]P_k(-)H_k[\hat{\underline{x}}_k(-)]^T + R_k\}^{-1}$
Definitions	$F[\hat{\underline{x}}(t),t] = \left.\dfrac{\partial\underline{f}[\underline{x}(t),t]}{\partial\underline{x}(t)}\right\vert_{\underline{x}(t) = \hat{\underline{x}}(t)}$ $H_k[\hat{\underline{x}}(-)] = \left.\dfrac{\partial\underline{h}[\underline{x}(t_k)]}{\partial\underline{x}(t_k)}\right\vert_{\underline{x}(t_k) = \hat{\underline{x}}_k(-)}$

The processing flow of the Kalman filter algorithm is one of propagating the state estimate from the previous measurement time, or initial state, up to the next measurement time instant, t_k. At the measurement time, each measurement available at that time is processed. The time propagation requires the integration of the state vector estimates and corresponding error covariance matrix. The state estimates and error covariance matrix are updated as each of the measurements are processed sequentially. At the conclusion of the measurement processing, for the time instant t_k, the resulting state estimates and error covariance matrix elements are used to reinitialize the state vector and covariance matrix integration variables for the time propagation up to the next measurement time instant.

The following matrices are defined which aid in assessing the quality of the filter's output results. The error covariance matrix is defined as the covariance of the error in the state estimate as

$$P(t) = \mathrm{Cov}[\underline{\tilde{x}}(t)\underline{\tilde{x}}^T(t)] \tag{5}$$

where

$$\underline{\tilde{x}}(t) = \underline{\hat{x}}(t) - \underline{x}(t). \tag{6}$$

The residual is defined as

$$\underline{r}_k = \underline{z}_k - \underline{h}_k[\underline{\hat{x}}(t_k)] \tag{7}$$

The residual covariance, or variance since the measurements are processed sequentially, is defined as

$$\mathrm{Cov}[\underline{r}_k \underline{r}_k^T] = H_k P_k(-) H_k^T + R_k. \tag{8}$$

The quality of the filter's outputs can be determined from these two matrices. The error covariance should be relatively small compared to the state estimate's value indicating low uncertainty associated with the estimate values. The residual variance should be zero mean and uncorrelated. If the system and measurements were linear and the noise processes were gaussian, the residual variance is theoretically zero mean and gaussian. These are the desired properties that should be exhibited by the filter's outputs.

This algorithm is applied to a system described by vehicle equations of motion describing position, body velocity and quaternions representing attitudes. The position is referenced in a boost reference inertial coordinate frame (see Appendix E or reference 4). These equations of motion are forced by the Space Shuttle Main Engines (SSME), the Solid Rocket Boosters (SRB), aerodynamic and plume forces, and winds. Each of these are described by mathematical models representing the nominal or reference conditions.

The state vector describing the system is composed of the vehicle motion variables and deviations in the forcing functions above. Additionally, bias states are included for the measurements described below. The state vector elements for this application are summarized in Table 2.-2.

The objective of the propulsion estimation program is to determine the propulsive contributions and to distinguish them from other forces on the vehicle. To accomplish this objective, measurements are selected to aid the filter algorithm. Measurements consisting of SSME chamber pressure, pressurant flows and fuel flow characteristics, SRB head pressure, IMU accelerometer, and radar azimuth, elevation and range measurements. These measurements are listed in Table 2.-3.

These measurements are processed one at a time; sequentially. Prior application [1] used the Kalman filter covariance update algorithm presented in Table 2.-1. This application resulted in numerical problems. As a result, this effort required the use of a factorized algorithm. The U-D factorized algorithm from [5] was selected. This algorithm is based on the covariance update equations presented in Table 2.-1; however, the covariance matrix is expressed as the product of three matrices as

$$P = UDU^T \tag{9}$$

The resultant factorization yields superior numerical qualities in providing equivalent double precision accuracy while using single precision programming providing less numerical sensitivity, and providing greater computational speed than other less sensitive algorithms.

Table 2.-2

State Vector Elements Modeled in PFILTER

Element	Units	Description
1	ft	x-position in boost reference inertial frame (BR)
2	ft	y-position in boost reference inertial frame
3	ft	z-position in boost reference inertial frame
4	ft/sec	u-component of body referenced velocity (B)
5	ft/sec	v-component of body referenced velocity
6	ft/sec	w-component of body referenced velocity
7	-	first quaternion element
8	-	second quaternion element
9	-	third quaternion element
10	-	fourth quaternion element
11		
12		not used
13		
14	lb	main engine 1 - mass overboard
15	-	main engine 1 - mixture ratio
16	lb/in^2	main engine 1 - chamber pressure deviation
17	lb/sec	main engine 1 - oxygen pressurant deviation
18	lb/sec	main engine 1 - hydrogen pressurant deviation
19	gal/min	main engine 1 - fuel volume flow bias
20	lb/in^2	main engine 1 - fuel flow pressure bias
21	deg R	main engine 1 - fuel flow temperature bias
22	lb/in^2	main engine 1 - chamber pressure bias
23-30	*	main engine 2 repeat for main engine 1 14-22
31-40	*	main engine 3 repeat for main engine 1 14-22
41	-	x-(B) component aerodynamic force deviation
42	-	y-(B) component aerodynamic force deviation
43	-	z-(B) component aerodynamic force deviation
44	lb	x-(B) component plume force deviation
45	lb	y-(B) component plume force deviation
46	lb	z-(B) component plume force deviation
47	ft/sec	north component wind velocity deviation
48	ft/sec	east component wind velocity deviation
49	ft/sec	down component wind velocity deviation
50	ft/sec^2	x-(BR) component accelerometer bias
51	ft/sec^2	y-(BR) component accelerometer bias
52	ft/sec^2	z-(BR) component accelerometer bias
53	deg	radar 1 - azimuth bias
54	deg	radar 1 - elevation bias
55	ft	radar 1 - range bias
56-58	*	radar 2 repeat for radar 1 53-55
59-61	*	radar 3 repeat for radar 1 53-55

Table 2.-2 (continued)

State Vector Elements Modeled in PFILTER

Element	Units	Description
62	lb	left motor - mass overboard
63	in	left motor - burn web thickness
64	**	left motor - burn rate coefficient deviation
65	-	left motor - nozzle coefficient deviation
66	lb/in^2	left motor - head pressure measurement bias
67-71	*	right motor repeat for left motor 62-66

* same units as for repeated elements
** $(in/sec)/((lb/in^2)^{**}(burn\ rate\ exponent))$

Table 2.-3

Measurement Vector Elements Modeled in PFILTER

Element	Units	Description
1	ft/sec^2	x-(BR) component acceleration
2	ft/sec^2	y-(BR) component acceleration
3	ft/sec^2	z-(BR) component acceleration
4		
5		not used
6		
7	lb/in^2	main engine 1 - chamber pressure
8	lb/sec	main engine 1 - oxygen pressurant flow
9	lb/sec	main engine 1 - hydrogen pressurant flow
10	gal/min	main engine 1 - fuel volume flow
11	lb/in^2	main engine 1 - fuel flow pressure
12	deg R	main engine 1 - fuel flow temperature
13-18	*	main engine 2 repeat for main engine 1 7-12
19-24	*	main engine 3 repeat for main engine 1 7-12
25	deg	radar 1 - azimuth
26	deg	radar 1 - elevation
27	ft	radar 1 - range
28-30	*	radar 2 repeat for radar 1 25-27
31-33	*	radar 3 repeat for radar 1 25-27
34	lb/in^2	left motor head pressure
35	lb/in^2	right motor head pressure

3. System Model and Measurements

As described in the previous section, this application uses the equations of vehicle motion as forced by propulsive, aerodynamic, plume and wind effects. The evolution of this motion is monitored by measurements processed by the filter algorithm. These measurements include propulsive, acceleration, and position as indicated by radar tracking. In this section, the equations describing the equations of motion, models for the forcing elements, and measurements will be presented. Later in the program descriptions presented in section 4., each of these models will be discussed in more detail in the routines in which they are implemented.

3.1 System Model

The system model includes the equations of motion, models for aerodynamic forces, plume forces, winds as they impact aerodynamic and plume forces, SSME and SRB thrust and mass overboard. If left uncorrected by the Kalman filtering algorithm, these models are sufficient to provide a reasonable approximation to the trajectory profile.

The modeling approach used in this development is an extension of those presented in references [6] and [7]. The extensions include; boost reference coordinate frame for position, plume forces, propulsion parameters and measurements, and using the U-D factorization form of the Kalman filter.

3.1.1 Equations of Motion

The equations of motion consist of position in boost reference frame, velocity in body frame, and body attitude relative to boost reference frame as derived from quaternions.

The vehicle position is maintained in the boost reference inertial coordinate frame (BR). This position is governed by the following

$$\dot{\underline{r}}^{(BR)} = {}^{BR}C^{B}\ \underline{v}^{(B)} \tag{10}$$

where

$\underline{r}^{(BR)}$ - BR position vector

${}^{BR}C^{(B)}$ - transformation matrix from body (B) frame to BR frame

$\underline{v}^{(B)}$ - velocity vector in body frame.

The vehicle velocity is formulated in body frame. This velocity is determined from

$$\dot{\underline{v}}^{(B)} = (\ q_{dyn} A \underline{c}_f + \underline{f}_T^{(B)} + \underline{f}_P^{(B)}\ )/m + {}^{B}C^{EF}\ \underline{g}^{(EF)}\ (\underline{r}^{(EF)}) - \underline{\omega} \times \underline{v}^{(B)} \tag{11}$$

where

q_{dyn} - dynamic pressure

A - aerodynamic reference area

$\underline{c}_f$ - aerodynamic force coefficient vector in body frame

$\underline{f}_T^{(B)}$ - thrust force vector in body frame

$\underline{f}_P$ - plume force vector in body frame

m - instantaneous vehicle mass

${}^{B}C^{EF}$ - transformation matrix from earth fixed (EF) to body frame

$\underline{g}^{(EF)}$ - gravity vector in earth fixed frame

$\underline{\omega}$ - body rotation vector relative to boost reference frame.

The vehicle mass in the difference between initial mass and the sum of mass overboard of each of the thrusting devices, SSME's and SRB's.

The vehicle attitude relative to the boost reference frame is governed by

$$\dot{\underline{q}} = f[p,q,r]\underline{q} \tag{12}$$

where

$\underline{q}$ - quaternion vector composed of 4 elements

f[p,q,r] - matrix whose elements are from body rate gyros.

3.1.2 Aerodynamic Forces

These forces are functions of vehicle attitude relative to the total velocity vector and the total vehicle speed. These angles are defined in Appendix E. The following equations define these angles in terms of body velocity, $\underline{v}^{(B)}$, from equation (11) and wind velocity, $\underline{v}_w$, obtained by meteorological measurements.

$$\underline{v}_r = \underline{v}^{(B)} - {}^{B}C^{LL}\, \underline{v}_w^{(LL)} \tag{13}$$

$$v_m = \sqrt{(v_{r1}^2 + v_{r2}^2 + v_{r3}^2)} \tag{14}$$

$$\alpha = \tan^{-1} (v_{r3} / v_{r1}) \tag{15}$$

$$\beta = \sin^{-1} (v_{r2} / v_m) \tag{16}$$

where

$\underline{v}_r$ = wind relative velocity vector

$\underline{v}_w^{(LL)}$ = wind velocity in local level coordinates

${}^{B}C^{LL}$ = transformation matrix from local level to body frame

$\underline{v}_m$ = magnitude of wind relative velocity; speed.

α = angle of attack of body relative to total velocity vector

β = angle of side slip of body relative to total velocity vector

v_{ri} = i^{th} component of wind relative velocity vector.

The aerodynamic force coefficients are expressed as linear coefficients of powers of these angles. Vehicle symmetry properties govern which powers are included. The aerodynamic coefficients are given as

$$C_A = C_{Ao} + C_{A\alpha}\, \alpha + C_{A\alpha^2}\, \alpha^2 + C_{A\beta^2}\, \beta^2 + C_{A\alpha\beta^2}\, \alpha\beta^2 \tag{17}$$

$$C_N = C_{No} + C_{N\alpha}\, \alpha + C_{N\alpha\beta^2}\, \alpha\beta^2 \tag{18}$$

$$C_Y = C_{Yo} + C_{Y\beta}\, \beta + C_{Y\alpha\beta}\, \alpha\beta + C_{Y\beta\alpha^2}\, \alpha^2\beta \tag{19}$$

where

C_A = axial force coefficient

C_N = normal force coefficient

C_Y = side force coefficient.

the force coefficient vector, $\underline{c}_f$, in equation (11) is formed from these coefficients as

$$\underline{c}_f = \begin{matrix} -C_A \\ C_Y \\ -C_N \end{matrix} \tag{20}$$

Each of the power coefficients given in equations (17), (18), and (19) are tabulated as functions of Mach number.

3.1.3 Plume Forces

The plume forces are also given as functions of the angle of attack and side slip. Additional corrections are applied as function of main engine power level and in situ dynamic pressure. The power level, angle and dynamic pressure effects are expressed as

$$\Delta P_L = 1.09 - P_L \tag{21}$$

$$f_{\Delta P} = K_2\ \Delta P_L^{\,2} + K_1\ \Delta P_L + 1 \tag{22}$$

$$f_{\alpha\beta} = K_\beta\ \beta^2 + K_\alpha\ \alpha + 1 \tag{23}$$

$$R_Q = q_{dyn}\ /\ q_{ref} \tag{24}$$

where

P_L = current main engine commanded power level

K_i = empirical correction coefficients

R_Q = ratio of in situ dynamic pressure to reference value.

The plume force vector, $\underline{f}_P$, is formed from these corrections as

$$\underline{f}_P = \begin{bmatrix} -(\Delta F_{Ab} - F_{Aorb} + F_{Aorb}\, f_{\Delta P})\, f_{\alpha\beta}\, R_Q \\ 0 \\ -\Delta F_N\, f_{\alpha\beta}\, f_{\Delta P}\, R_Q \end{bmatrix} \tag{25}$$

where

ΔF_{Ab} = incremental axial base force

F_{Aorb} = orbiter axial force

ΔF_N = incremental normal force.

Each of these and the K's in equations (21) through (24) are tabulated as functions of altitude.

3.1.4 Main Engine Thrust Forces

Each of the main engine's (and SRB's) thrust is transformed from its centerline, through which the thrust is assumed to act, to the vehicle's body frame using the nozzle deflections and mount structure angles. This transformed thrust is then summed to form the total thrust as

$$\underline{f}_T^{(B)} = \sum_{i=1}^{n} {}^{B}C_i^{CL} \begin{matrix} f_{Ti}^{CL} \\ 0 \\ 0 \end{matrix} \quad \underline{\Delta} \quad \sum_{i=1}^{n} {}^{B}C_i^{CL}\, \underline{f}_{Ti}^{(CL)} \tag{26}$$

where

f_{Ti}^{CL} = individual main engine (and SRB's) thrust

${}^{B}C^{CL}$ = transformation matrix from center line to body.

The thrust is corrected for atmospheric ambient pressure at the vehicle's altitude by

$$f_{Ti}^{CL} = f_{Ti\ vac} - P_s\, A_{ei} \tag{27}$$

where

$f_{Ti\ vac}$ = each main engine's vacuum thrust

A_{ei} = each main engine's nozzle exit area.

Each of the main engine's thrust is based on the individual engine's tag value, at 100% power level test stand data, plus a correction based on the current power level's and the 100% power level's nominal engine thrust.

The weight overboard for each of the main engines is the sum of the fuel and oxidizer less the gaseous oxygen and hydrogen pressurant weight flow. Fuel and Oxidizer weight flow rate's models are identical to the thrust described above. Pressurant flow rates are based on a nominal engine's characteristics.

3.1.5 SRB Thrust Forces

The rocket booster's thrust is transformed into body frame from the nozzle center line as in the main engine's. Each rocket booster's thrust is based on a quasi-static web burn rate model plus corrections derived from prior post flight data analyses. The thrust is given by

$$f_{Ts} = c_m \, c_T \, A_T \, P_O \, T_{SF} + \Delta T_{SRB} \tag{28}$$

where

c_m = nozzle coefficient

c_T = thrust coefficient

A_T = nozzle area

P_O = nozzle entrance pressure

T_{SF} = derived thrust scaling factor (function of web depth)

ΔT_{SRB} = derived thrust correction.

The nozzle entrance pressure, P_O, is computed from the following

$$P_O = \left(\frac{c^* \rho_P \, a \, A_B}{g \, A_T} \right)^{\frac{1}{(1-n)}} \tag{29}$$

where

c^* = characteristic exhaust velocity

ρ_P = propellant density

a = pressure scaled propellant burn rate coefficient

A_B = instantaneous propellant burn area

g = gravitational acceleration

A_T = nozzle throat area.

n = propellant burn rate exponent.

The mass overboard is given as

$$\dot{m}_s = \rho_P \, \dot{\tau} \, A_B \tag{30}$$

where

$$\dot{\tau} = a \, P_0^n \tag{31}$$

3.2 Measurement Equations

The measurements used in the Kalman filtering algorithm were selected to maximize the likelihood of discriminating between aerodynamic/plume and propulsive forces. External measurements, such as ground radar tracking and IMU accelerations, provide redundant information that characterized the vehicle motion and provide the opportunity to characterize each others' biases. Internal measurements, such as SRB head pressure and SSME flow rates, add to the redundancy by restricting the contributors to those measurements from being adjusted by the filter algorithm to account for vehicle motion deviations.

The measurements described in the following included radar tracking, IMU accelerations, SSME chamber pressure and flows, and SRB head pressure.

3.2.1 Radar Tracking

The measurement equations are identical to those described in section C.2.2 for the BET linearized Kalman filtering algorithm. The position coordinate frames are identical for the propulsion estimation program and the BET program; boost reference inertial. However, in this case, only three radars are used in the tracking. The updates provided by the radar measurements carry less weight as other measurements, i.e. accelerometer; therefore, mainly provide a gauge on the performance of the filter.

3.2.2 IMU Accelerations

The accelerations are assumed measured in the boost reference frame; sensed. These accelerations are modeled as contributions of aerodynamic, propulsion, plume and coriolis forces. Additional error is assumed to permit acceleration biases and additive random noise.

The measured acceleration in sensed frame is expressed as

$$\underline{a}_m^{(S)} = {}^{S}C^{B} \left[\left(q_{dyn} A \underline{c}_f + \underline{f}_T^{(B)} + \underline{f}_P^{(B)} \right)/m \right] + \underline{\omega} \times \left(\underline{\omega} \times \left(\underline{r}_S^{(B)} - \underline{r}_{cg}^{(B)}\right) \right) + \underline{b}_a^{(S)} + \underline{v}_a^{(S)} \tag{32}$$

where

${}^{S}C^{B}$ = body to sensed frame transformation matrix

$\underline{\omega}$ = body rotation rate from body rate gyros

$\underline{r}_S^{(B)}$ = body frame coordinates of IMU location

$\underline{r}_{cg}^{(B)}$ = body frame coordinates of vehicle center of gravity

$\underline{b}_a^{(S)}$ = acceleration bias vector

$\underline{v}_a^{(S)}$ = acceleration measurement noise.

Other terms involving aerodynamic, thrust and plume forces are defined in section 3.1.

3.2.3 SSME Measurements

The main engine measurements include chamber pressure, gaseous pressurant flow rates and fuel flow rate. The nominal chamber pressure is determined by the pressure associated at the rated power level for each individual engine multiplied by the commanded power level. The gaseous pressurant flow rates are determined from the nominal flow rates associated with the commanded power level.

3.2.4 SRB Measurements

The solid rocket booster measurement is the head pressure, at the opposite end from the exit nozzle. This measurement model is formed from the nozzle pressure equation (29) as

$$P_{O_{head}} = \frac{P_O}{2}\left(1 + \sqrt{\left(1 + 16\pi RT\left(\frac{A_T\, g}{c^* A_B}\right)^2 \right)}\; \frac{L^3}{V_P} \right) \qquad (33)$$

where

R = universal gas constant

T = gas absolute temperature

L = internal propellant grain port length

V_P = corresponding port volume

Port volume in the above equation is obtained from internal ballistics prediction programs and is adjusted as necessary to match the corresponding head pressure predictions.

4. Propulsion Estimation Program - PFILTER

This section describes the propulsion estimation program PFILTER. PFILTER, which mechanizes the Kalman filter algorithm and is written in the FORTRAN77 standard, provides estimates for the Shuttle's boost phase flight. This program is structured in parallel with the Kalman filter algorithm presented in Table 2.-1. There are routines that form the state vector time derivatives, and there are routines, in some cases the same routines, that form the measurements. The organization of the program's routines is shown in Figure 4.-1.

The program functions as follows. The main routine, FILTER, controls the operational flow within the structure of the Kalman filter algorithm. It opens and reads the data files, initially zeros arrays used later, ZERO, and initializes arrays, INITIL, based on data contained with the BLKDAT block data routine. The formation of the time derivatives, PROPAF, and their integration, RK4FIL, up to a measurement time is controlled in FILTER. At the measurement time, the measurement data is input by the GETDAT routine. For those measurements to be processed, the UPDATE routine calls the routine that forms the measurement and accomplishes the Kalman filter measurement update. Most of the routines just mentioned also call additional routines that contain the modeling for the function to be performed.

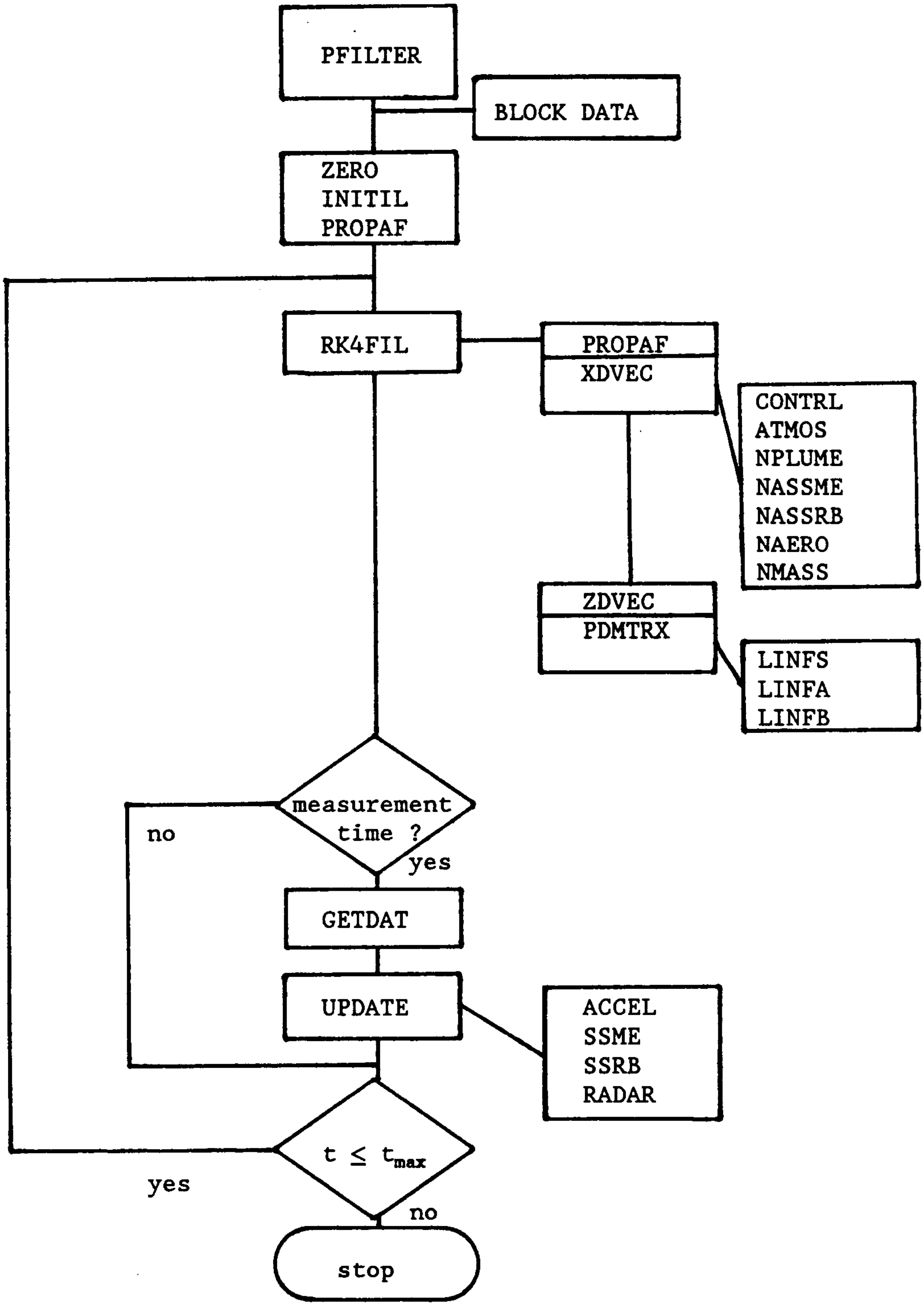


Figure 4.-1: PFILTER Program Flow Chart

Each of these routines, their function and the equations they form, will e discussed in the subsequent subsections. The correspondence between the mathematical notation and the symbology used in the FORTRAN code will also be presented.

Supporting the operation of this program are several input data files. These data files are produced by the preprocessing programs mentioned earlier. Also, this program produces an output file for plotting to assess the quality of programs other outputs. The plotting program then postprocesses the resulting outputs. These preprocessing and postprocessing programs are discussed in Appendix A.

4.1 Program FILTER

This main routine controls the Kalman filter flow as previously described as well as handling the input and output of data files, and array initializations.

Three data files are input. These are:

Data File Unit Number	Data File Name
1	CONTRL.DAT
2	REALMEA.DAT
3	METDAT.DAT

The first, CONTRL.DAT, contains the vehicle control input data. This data is as follows:

Symbol	Common	Units	Description
TTIME	NOZDAT	sec	time corresponding to control data
TRLD,TTLD	NOZDAT	in	left hand SRB rock and tilt displacements
TRRD,TTRD	NOZDAT	in	right hand SRB rock and tilt displacements
TPL	NOZDAT	-	SSME power level command
TPL1,TYL1	NOZDAT	deg	SSME-1 pitch and yaw gimbal angles
TPL2,TYL2	NOZDAT	deg	SSME-2 pitch and yaw gimbal angles
TPL3,TYL3	NOZDAT	deg	SSME-3 pitch and yaw gimbal angles
TRATER	NOZDAT	deg/sec	vehicle roll rate with respect to BR
TRATEP	NOZDAT	deg/sec	vehicle pitch rate with respect to BR
TRATEY	NOZDAT	deg/sec	vehicle yaw rate with respect to BR
TROLL	NOZDAT	deg	vehicle roll attitude with respect to BR
TPITCH	NOZDAT	deg	vehicle pitch attitude with respect to BR
TYAW	NOZDAT	deg	vehicle yaw attitude with respect to BR

These data are used in routine CONTRL to compute the specific time point control values as determined by a linear interpolation, performed by INTRP1.

The second data file, REALMEA.DAT, is read in by GETDAT. This file contains the measurements from the accelerometers, main engines, solid rocket motor head pressures, and ground tracking radars. Each set of measurements, i.e. accelerometers, are communicated via common to the appropriate update routines and will be discussed later.

The third data set, METDAT.DAT, contains tables of meteorological data at one thousand foot increments. The data is as follows:

Symbol	Common	Units	Description
TALT	SWIND	ft	altitude values for corresponding data
TSRHO	SATMOS	$lb\ sec^2/ft^4$	atmospheric density
TSP	SATMOS	lb/ft^2	static pressure
TSSUND	SATMOS	ft/sec	speed of sound
TVWX	SWIND	ft/sec	x component, north, of wind velocity
TVWY	SWIND	ft/sec	y component, east, of wind velocity

These data are used in routine ATMOS to determine the values at a specific altitude based on a linear interpolation by the INTRP1 routine.

The routine ZERO is called to zero arrays used for integration variables. INITIL initializes the state variables, i.e. position, velocity, and the values of the error covariance matrix used in the Kalman filter algorithm.

PROPAF forms the time derivatives of the state variables. Included in the state variables are; position, velocity, quaternions, SRB burn web thickness, mass overboard, and the upper triangular elements of the error covariance matrix since this matrix is symmetric. These derivatives are integrated numerically using a fourth order Runge-Kutta algorithm mechanized in the RK4FIL routine.

If the filter time does not exceed the specified maximum time, TMAX contained in the BLKDAT routine, the equations are integrated up to the next measurement time. At this instant, the measurement update is performed. The results are used to reinitialize the state variables and error covariance matrix elements for integration to the next measurement time. This process continues until the maximum time is exceeded. When this occurs, the output file, FILOUT.DATA, is closed.

4.2 BLKDAT Routine

The BLKDAT routine contains data to support the program flow control, i.e. TMAX for the maximum time, and modeling data that may frequently change to refine the filter's performance. Specific modeling data associated with a functional aspect of the system, i.e. mass properties, is contained within data statements for that function within its corresponding routine, i.e. NMASS. As a guide to determining where data resides, parameters that are frequently changed and program control data are located in BLKDAT. These include propulsion modeling data for the SRB's and SSME's.

The following program control data are specified in BLKDAT.

Symbol	Common	Units	Description
ISTAGE	STAGE	-	identifier for stage 1 or 2 data selection
TMAX	TIMDAT	sec	maximum filter time after liftoff
HSTEP	TIMDAT	sec	numerical integration step size
TSAMP	TIMDAT	sec	measurement sample time increment
TSTART	TIMDAT	sec	filter start time after liftoff
TRINT	TIMDAT	sec	time to start body rate integration
ITYPE	TYPE	-	<1 for synthetic data generation
NMEAS	NMEASR	-	number of measurement types, i.e. radar
IMEAS	NMEASR	-	specific measurement the types available
IATMOS	SATMOS	-	<1 for standard atmospheric model data
NALT	SWIND	-	number of tabulated altitude's
NS	LINFMT	-	number of vehicle motion states
NPAR	LINFMT	-	number of parameter state elements
NRMEAS	LINHMT	-	total number of measurements (all types)
ISSME	IPARAM	-	-1 for including SSME parameters
ISRB	IPARAM	-	-1 for including SRB parameters
IAERO	IPARAM	-	-1 for including aerodynamic parameters
IPLUME	IPARAM	-	-1 for including plume parameters
IWIND	IPARAM	-	-1 for including wind parameters
JACB	IPARAM	-	-1 for including accel. bias parameters
JRDR	IPARAM	-	-1 for including radar bias parameters

Data that specifies the geometrical relationships for the thrust points for each thrusting device, location of the IMU for the accelerometer, and data which represents structural deformation for the main engine mount structure due to load are specified in BLKDAT. These data and other geometrical data are:

Symbol	Common	Units	Description
RS	LAYOUT	ft	body coordinates of IMU location
RA	LAYOUT	ft	body coordinates of aerodynamic reference
RT1	LAYOUT	ft	body coordinates of SSME-1
RT2	LAYOUT	ft	body coordinates of SSME-2
RT3	LAYOUT	ft	body coordinates of SSME-3
RT4	LAYOUT	ft	body coordinates of left hand SRB
RT5	LAYOUT	ft	body coordinates of right hand SRB
TSDBTP	STRUCT	deg	pitch structure deflection coefficients due to power level and gimbal angles
TSDBTY	STRUCT	deg	yaw structure deflection coefficients due to power level and gimbal angles
TSDBAP	STRUCT	deg	pitch structure deflection coefficients due to vehicle acceleration
TSDBAY	STRUCT	deg	yaw structure deflection coefficients due to vehicle acceleration
AREA	GEODAT	ft^2	aerodynamic reference area
DIA	GEODAT	ft	aerodynamic reference length
PSMEPS	GEODAT	ft^2	main engine exit area
PSRBPS	GEODAT	ft^2	SRB exit area

Earth model specific data and initial vehicle orientation data are as follows:

Symbol	Common	Units	Description
RE	EDATA	ft	earth radius at the equator
FLAT	EDATA	-	earth model flattening factor
OMEGE	EDATA	rad/sec	earth angular rotation rate
XMU	EDATA	ft^3/sec^2	earth gravitational parameter
XJ2	EDATA	-	earth oblate spheroid gravity parameter

CRAD	EDATA	deg/rad	conversion from degrees to radians
CUENED	ASTRON	-	transformation matrix from UEN to NED
CBRIB	ASTRON	-	transformation matrix from BR to body

Launch points and radar site coordinates defined for the earth model are:

Symbol	Common	Units	Description
OLATD	LAUCOR	deg	launch point geodetic latitude
OLONG	LAUCOR	deg	launch point longitude
OHT	LAUCOR	ft	launch point altitude
RLAT	RDRDAT	deg	radar site geodetic latitude
RLONG	RDRDAT	deg	radar site longitude
RHT	RDRDAT	ft	radar site altitude
NRDR	RDRDAT	-	number of radar sites used

Stage 1 and 2 initial mass and stage 2 state vector data are:

Symbol	Common	Units	Description
XMASSI	IPROPR	lb	stage 1 initial weight
XMASS2	IPROPR	lb	stage 2 initial weight
R2	STGDAT	ft	stage 2 initial position in BR frame
VB2	STGDAT	ft/sec	stage 2 initial velocity in body frame
Q2	STGDAT	-	stage 2 initial quaternion elements

Values for the Kalman filter process noise "tuning" parameters and initial error magnitudes for initializing the error covariance matrix are:

Symbol	Common	Units	Description
A	CONST	(see Table 2.-2)	process noise magnitudes (1-sigmas)
R	CONST	(see Table 2.-3)	measurement noise variances
ER	APRIOR	ft	initial position uncertainties

EVB	APRIOR	ft/sec	initial body velocity uncertainties
EQ	APRIOR	-	initial quaternion uncertainties
ESME	APRIOR	(see Table 2.-2)	initial SSME model uncertainties
ESRB	APRIOR	(see Table 2.-2)	initial SRB model uncertainties
EARO	APRIOR	-	initial aero coefficient uncertainties
EPLM	APRIOR	lb	initial plume force uncertainties
EWND	APRIOR	ft/sec	initial wind velocity uncertainties
EACB	APRIOR	ft/sec^2	initial accelerometer bias uncertainties
ERDR	APRIOR	(see Table 2.-3)	initial radar bias uncertainties

A standard atmospheric model is used when the meteorological data is not specified as an option. The data used for this model is defined by the following tabular arrays:

Symbol	Common	Units	Description
THALT	ATMOSP	ft	tables of altitude
THRHO	ATMOSP	$lb\ sec^2/ft^4$	tables of density
THP	ATMOSP	lb/ft^2	tables of static pressure
THSUND	ATMOSP	ft/sec	tables of speed of sound

Tabular arrays containing modeling parameters for the SRB model are:

Symbol	Common	Units	Description
NTAU	SRBTBL	-	number of web thicknesses in SRB model
TFSA	SRBTBL	in^2	table of propellant surface areas
TPVOL	SRBTBL	in^3	table of internal port volumes
TAT	SRBTBL	in^2	table of throat areas
TCSTR	SRBTBL	in/sec	table of characteristic exhaust velocities
TTAU	SRBTBL	in	table of web thicknesses
CM	SRBTBL	-	SRB nozzle coefficient
G	SRBTBL	in/sec^2	gravitational constant
GAM	SRBTBL	-	ratio of specific heats
PE	SRBTBL	lb/in^2	SRB exit pressure
ABAR	SRBTBL	in/sec	burn rate coefficient
PBAR	SRBTBL	lb/in^2	corresponding reference pressure
EPBAR	SRBTBL	-	burn rate exponent
XLNGTH	SRBTBL	in	reference length for port volume

RBAR	SRBTBL	$sec^2/(R^\circ$ mole)	universal gas constant
XMBAR	SRBTBL	lb/mole	molecular constant
TEMP	SRBTBL	R°	combustion temperature
PI	SRBTBL	-	constant π

The tabular values above are used in NASSRB with the routine INTRP1 to determine specific values of the associated parameters.

The SSME power law modeling data are also included in BLKDAT. The constants for this model are given as follows:

Symbol	Common	Units	Description
PR	SMETBL		partials from gain model
YN	SMETBL		reference: thrust, fuel and oxidizer flows
XN	SMETBL		reference: pressurant flows
PRZ	SMETBL		additional partials for fuel flows
PRX	SMETBL		additional partials for pressurant flows
ZN	SMETBL		reference: fuel flows
RPLTAG	SMETBL	lb/in^2	individual engine tag pressure at PL=1
WDO2TU	SMETBL	lb/sec	individual engine tag oxidizer flow rate
WDH2TU	SMETBL	lb/sec	individual engine tag fuel flow rate
TVACTU	SMETBL	lb	individual engine tag vacuum thrust

Other data for atmospheric, aerodynamic, plume, mass properties and SRB models can be found in the routines ATMOS, NAERO, NPLUME, NMASS and NASSRB respectively.

4.3 INITIL Routine

The INITIL routine initializes the states and error covariance matrix diagonals for the Kalman filter algorithm. Also, since the states are numerically integrated, the storage arrays used for the numerical integration are also initialized.

For Stage 1, INITIL computes the initial BR position vector from the launch pad coordinates contained in BLKDAT. Also computed is the transformation matrix from earth centered inertial (ECI) to BR. This matrix is used later in the RADAR measurement update routine. Initial vehicle velocity and quaternions are also computed. Values for these states for stage 2 are stored in BLKDAT and are extracted from the BET filter/smoother output's described in Appendix C.

The states and filter variables initialized in INITIL are summarized below.

Symbol	Common	Units	Description
R	STATES	ft	initial BR position vector
VB	STATES	ft/sec	initial body velocity vector
Q	STATES	-	initial quaternion elements
CIBRI	ASTRON	-	transformation matrix from ECI to BR
XKM	PREUP	(see Table 2.-2)	state vector prior to measurement update
PKM	PREUP	(see Table 2.-2)	error covariance matrix prior to update
VAR	unlabeled		first element: current time, next "n" elements state vector, next "n(n+1)/2" elements upper triangle of PKM.
DER	unlabeled		corresponding time derivatives of VAR
TEMP	unlabeled		temporary storage array used by RK4FIL.

4.4 PROPAF Routine

This routine sets-up all of the time derivatives for the numerical integration routine RK4FIL. As a result, all of the dynamical model related routines are called from this routine. Those dynamical routines which have an effect on vehicle motion are called from XDVEC. The routine ZDVEC forms the dynamics associated external measurement biases. These two routines then form the time derivatives for the state vector elements.

The routine PDMTRX forms the time derivatives of the error covariance matrix whose upper triangular elements are used for numerical integration. This routine calls three routines; LINFS, LINFA, and LINFB. The first, LINFS, develops the partial derivatives of the vehicle motion with respect to the vehicle motion variables. The second, LINFA, develops the partial derivatives of the vehicle motion with respect to the aerodynamic, plume, wind and propulsion states. The last, LINFB, develops the linearized dynamics associated with the measurement biases.

In the next subsection, the XDVEC, LINFS and LINFA routines will be described. These routines represent the most significant features of the model's representing this application.

4.4.1 XDVEC Routine

To form the time derivatives of the vehicle motion, XDVEC calls other routines which contribute to these derivatives. The major routines called are CONTRL, ATMOS, NPLUME, NASSME, NASSRB and NAERO. These routines also form the time derivatives for the dynamics represented by each of their functional areas, partial derivatives that effect vehicle motion with respect to their model elements, and for NASSME and NASSRB, estimates of measurements based on the current state values and associated partial derivatives of these measurements with respect to the model elements.

The deferential equations formed within XDVEC were presented in section 3.1. For those equations, the FORTRAN symbol and originating routine is summarized as in the following. For the position vector, equation (10);

Math	Symbol	Units	Subroutine	Description
$\underline{r}^{(I)}$	R	ft	XDVEC	BR position vector
$\underline{v}^{(B)}$	VB	ft/sec	XDVEC	body velocity vector
${}^{I}C^{B}$	CBI	-	XDVEC	body to BR matrix

The body velocity, equation (11), is summarized as

Math	Symbol	Units	Subroutine	Description
$q_{dyn}A/m$	QAOM	ft/sec^2	XDVEC	product of dynamic pressure, reference area divided by mass
$\underline{c}_f$	CF	-	XDVEC	aerodynamic force coefficient
$\underline{g}^{(EF)}$	GRAV	ft/sec^2	AGRAV	gravitational vector in EF
$\underline{\omega} \times \underline{v}^{(B)}$	OCV	ft/sec^2	XDVEC	Coriolis acceleration

$\underline{f}_T^{(B)}$	THRUST	lb	XDVEC	total thrust vector
$\underline{f}_P^{(B)}$	FPLUME	lb	NPLUME	plume force vector
m	XMASS	lb sec/ft^2	NMASS	current vehicle mass
$^BC^I$	CIB	-	XDVEC	BR to body matrix

The quaternion elements, equation (12), is summarized as

Math	Symbol	Units	Subroutine	Description
$\underline{q}$	Q	-	XDVEC	quaternion vector
f[p,q,r]	QDMTRX	rad/sec	QMTRX	matrix of body rate components
p, q, r	OMEGA	rad/sec	CONTRL	body rates with respect to BR

The equations which incorporate winds and compute the aerodynamic angles and total velocity, equations (13) through (16), are summarized as

Math	Symbol	Units	Subroutine	Description
$^BC^{LL}$	CLLB	-	XDVEC	local level to body transformation matrix
$\underline{v}_w^{LL}$	VW	ft/sec	ATMOS	wind velocity vector
$\underline{v}_r^{(B)}$	VR	ft/sec	XDVEC	body relative velocity vector
v_m	VM	ft/sec	XDVEC	total wind relative velocity
α	ALPHA	deg	XDVEC	angle of attack
β	BETA	deg	XDVEC	angle of slide slip

The total thrust vector components, equations (26) through (31) are summarized as

Math	Symbol	Units	Subroutine	Description
${}^{B}C^{C}$	CCLB	-	CONTRL	thrust centerline to body transformation matrix
	TSSME	lb	XDVEC	main engine thrust
	TSRB	lb	XDVEC	SRB thrust
	TVACL	lb	NASSME	main engine vacuum thrust
	TVACS	lb	NASSRB	SRB vacuum thrust
P_s	PS	lb/ft^2	ATMOS	ambient static pressure
	PSMEPS	ft^2	BLKDAT	main engine exit area
	PSRBPS	ft^2	BLKDAT	SRB exit area

4.4.1.1 CONTRL Routine

This routine computes the thrust centerline to body transformation matrix ${}^{B}C^{C}$. For the main engines, this computation accounts for the engine mount structure cant angle, and the power level and inertia load induced structural deflections of the engine mount structure. The SRB nozzle entrance pressure induced structural deformation effects are accounted for in converting gimbal linear displacements to angular deflections.

The transformation matrix for the main engines is obtained from [13] and is given as

$$ {}^{B}C^{C} = \left[\frac{\partial P_g}{\partial P_L} P_L + \Delta\theta_D + P_g + P_{ref} \right]_P \left[\frac{\partial Y_g}{\partial P_L} P_L + \Delta\Psi_D + Y_g \right]_Y \tag{34} $$

Math	Symbol	Units	Subroutine	Description
${}^{B}C^{C}$	CCLB	-	CONTRL	thrust to body matrix
P_L	PL	-	CONTRL	commanded power level
$\frac{\partial P_g}{\partial P_L}$ PL		deg	CONTRL	pitch structural deflection due to thrust
$\frac{\partial Y_g}{\partial P_L}$ PL		deg	CONTRL	yaw structural deflection due to thrust
$\Delta\theta_D$		deg	CONTRL	pitch structural deflection due to inertia loads
$\Delta\Psi_D$		deg	CONTRL	yaw structural deflection due to inertia loads

P_g	GIMPTL	deg	CONTRL	main engine pitch gimbal
P_{ref}	DEF	deg	CONTRL	main engine structural cant
Y_g	GIMYWL	deg	CONTRL	main engine yaw gimbal
$[\ ]_P$		-	TMATP	rotation matrix about pitch
$[\ ]_Y$		-	TMATY	rotation matrix about yaw

For each of the main engines, the structural deflection partials with respect to power level are given as

engine no. 1:

$$\frac{\partial P_g}{\partial P_L} = .566 + .3537E\text{-}1\ P_g + .206E\text{-}3\ P_g^2 + .49E\text{-}3\ Y_g - .278E\text{-}3\ Y_g^2 \quad (35)$$

$$\frac{\partial Y_g}{\partial P_L} = -.3802 - .134E\text{-}2\ P_g - .408E\text{-}3\ P_g^2 + .863E\text{-}2\ Y_g + .238E\text{-}3\ Y_g^2 \quad (36)$$

engine no. 2:

$$\frac{\partial P_g}{\partial P_L} = .5052 + .2995E\text{-}1\ P_g - .382E\text{-}3\ P_g^2 - .563E\text{-}2\ Y_g + .66E\text{-}4\ Y_g^2 \quad (37)$$

$$\frac{\partial Y_g}{\partial P_L} = .0831 - .26E\text{-}3\ P_g - .276E\text{-}3\ P_g^2 - .347E\text{-}3\ Y_g + .158E\text{-}3\ Y_g^2 \quad (38)$$

engine no. 3:

$$\frac{\partial P_g}{\partial P_L} = 1.4776 + .9415E\text{-}1\ P_g - .7E\text{-}4\ P_g^2 - .413E\text{-}2\ Y_g - .43E\text{-}3\ Y_g^2 \quad (39)$$

$$\frac{\partial Y_g}{\partial P_L} = -.4265 - .102E\text{-}2\ P_g - .192E\text{-}3\ P_g^2 + .313E\text{-}2\ Y_g + .142E\text{-}3\ Y_g^2 \quad (40)$$

The changes in apparent gimbal angle due to inertia loads for each engine are given as

engine no. 1:

$$\Delta\Theta_D = -.1054\ \eta_x + .0368\ \eta_z \tag{41}$$

$$\Delta\Psi_D = .0104\ \eta_x - .0069\ \eta_z \tag{42}$$

engine no. 2:

$$\Delta\Theta_D = -.1143\ \eta_x + .0331\ \eta_z \tag{43}$$

$$\Delta\Psi_D = -.0102\ \eta_x - .0069\ \eta_z \tag{44}$$

engine no. 3:

$$\Delta\Theta_D = -.1645\ \eta_x + .1580\ \eta_z \tag{45}$$

$$\Delta\Psi_D = .0108\ \eta_x + .0082\ \eta_z \tag{46}$$

where the η_x and η_z are the body longitudinal (forward) and normal (upward) load factors respectively.

The following equations describe the transformation matrix from thrust center line to body for the SRB's

$$^{B}C^{C} = [\ P_g\ ]_P\ [\ Y_g\ ]_Y \qquad (47)$$

for the left hand SRB, P_g and Y_g are given in terms of rock, R, and tilt, T, angles by;

$$P_g = .707\ R - .707\ T \qquad (48)$$

$$Y_g = -.707\ R - .707\ T \qquad (49)$$

For the right hand SRB, P_g and Y_g are given by;

$$P_g = -.707\ R + .707\ T \qquad (50)$$

$$Y_g = .707\ R + .707\ T \qquad (51)$$

The rock and tilt angles above are obtained from rock and tilt displacements from the following;

$$R = .792\ R_D + .5\left(1 - \frac{P_{ON}}{595}\right) \qquad (52)$$

$$T = .792\ T_D + .5\left(1 - \frac{P_{ON}}{595}\right) \qquad (53)$$

The symbols for these equations used in CONTRL are summarized as

Math	Symbol	Units	Subroutine	Description
P_{ON}	PON	lb/in^2	NASSRB	nozzle entrance pressure
$R_D(L)$	RLD	in	CONTRL	left hand rock displacement
$T_D(L)$	TLD	in	CONTRL	left hand tilt displacement
$R_D(R)$	RRD	in	CONTRL	right hand rock displacement
$T_D(R)$	TRD	in	CONTRL	right hand tilt displacement
R(L)	RL	deg	CONTRL	left hand rock angle
T(L)	TL	deg	CONTRL	left hand tilt angle
R(R)	RR	deg	CONTRL	right hand rock angle
T(R)	TR	deg	CONTRL	right hand tilt angle
Y_G	GIMYWS	deg		yaw gimbal angle
P_G	GIMPTS	deg		pitch gimbal angle

4.4.1.2 ATMOS Routine

The ATMOS routine provides standard atmospheric properties if meteorological data is not specified as input. For either option, density for dynamic pressure, static pressure for altitude thrust correction, speed of sound for aerodynamic Mach number interpolation, and wind components in local level coordinates are the primary outputs. Additionally, gradients of these quantities with respect to altitude are provided.

The outputs of ATMOS when IWIND option is used are summarized as:

Symbol	Units	Common	Description
VW	ft/sec	VARIAB	vector components of wind velocity
VWGRAD	ft/sec/ft	VARIAB	gradient wind vector with altitude
F		LINFMT	elements 47 thru 49 of F matrix
S		CONST	process noise variance elements 47 thru 49
XDOT		DRVITV	elements 47 thru 49 of time derivatives

The units for the F, S and XDOT arrays correspond to those in Table 2.-1. Outputs of the ATMOS routine are used in XDVEC and LINFS routines.

4.4.1.3 NPLUME Routine

This routine forms the plume force vector in body axis for the XDVEC routine and the associated partial derivatives for use in the LINFA routine. The equations were presented in Section 3.1.3 for these forces.

The following summarizes the model and code symbols;

Math	Symbol	Units	Subroutine	Description
ΔP_L	DPL	-	NPLUME	difference between 1.09 and PL
K_1	XK1	-	NPLUME	linear power level coefficient
K_2	XK2	-	NPLUME	quadratic power coefficient
K_α	XKA	deg^{-1}	NPLUME	angle of attack coefficient
K_β	XKB	deg^{-1}	NPLUME	side slip coefficient
$f_{\Delta P}$	FDP	-	NPLUME	power level modeling factor
$f_{\alpha\beta}$	FAB	-	NPLUME	angle of attack and side slip modeling factor
R_Q	RQ	-	NPLUME	ratio of the dynamic pressure to the reference value
ΔF_{AB}	DFAB	lb	NPLUME	incremental base axial force
F_{Aorb}	FAORB	lb	NPLUME	orbiter axial force
ΔF_{Nb}	DFNB	lb	NPLUME	incremental normal force

These variables are derived from linear interpolation of data tables as a function of altitude for stage 1. Stage 2 plume axial and normal forces, DFAB and DFNB respectively, are specified only as functions of altitude and are obtained from the tabular arrays. The symbols used to define each variable in the tabular arrays is the same as the variable except with a "T" preceding the variable symbol, i.e. TALT for altitude tabular array, and with a "2" after the variable symbol for the second stage tabular arrays, i.e. TDFNB2 for the incremental normal force for the second stage.

Additional outputs of NPLUME are summarized as:

Symbol	Units	Common	Description
FPLUME	lb	PLUME	total plume force vector in body axis
F		LINFMT	elements 44 thru 46 of F matrix
S		CONST	process noise variance elements 44 thru 46
XDOT		DRVITV	elements 44 thru 46 of time derivatives

The units for the F, S and XDOT arrays correspond to the units indicated in Figure 2.-1.

4.4.1.4 NASSME Routine

This routine converts the coefficients in the power gain law modeling data from BLKDAT, using the function POW, into the SSME performance characteristics: thrust, fuel and oxidizer weight flow rates, hydrogen and oxygen pressurant weight flow rates. Estimates of measurements of chamber pressure, pressurant flow rates, and fuel volume, temperature and pressure flow parameters.

Partial derivatives are also computed for use in the linearized dynamics routine LINFA for propagating the error covariance matrix. Partial derivatives are also provided for the measurement update routines ACCEL and SSME for accelerometer and internal main engine measurements respectively.

The system model for the dynamics of the main engine is based on the weight overboard for each engine [8]. The net weight flow overboard is then given by

$$\dot{w} = \dot{w}_{H2} + \dot{w}_{O2} - \dot{w}_{gH2} - \dot{w}_{gO2} \tag{54}$$

Math	Symbol	Units	Subroutine	Description
$\dot{w}_{H2}$	WDH2H	lb/sec	NASSME	estimated fuel flow rate
$\dot{w}_{O2}$	WDO2H	lb/sec	NASSME	estimated oxidizer flow rate
$\dot{w}_{gH2}$	WDGH	lb/sec	NASSME	estimated hydrogen pressurant flow rate
$\dot{w}_{gO2}$	WDGO	lb/sec	NASSME	estimated oxygen pressurant flow rate

where the values of the individual weight flows given above are the sum of the tag values from the power gain law, plus the deviations due to changes with respect to (wrt) changes in mixture ratio, chamber pressure, and pressurant flow rates as given by

$$\dot{w}_{H2} = \dot{\bar{w}}_{H2} + \frac{\partial \dot{w}_{H2}}{\partial MR}\,\delta MR + \frac{\partial \dot{w}_{H2}}{\partial P_c}\,\delta P_c + \frac{\partial \dot{w}_{H2}}{\partial w_{gO}}\,\delta w_{gO} + \frac{\partial \dot{w}_{H2}}{\partial w_{gH}}\,\delta w_{gH} \qquad (55)$$

$$\dot{w}_{O2} = \dot{\bar{w}}_{O2} + \frac{\partial \dot{w}_{O2}}{\partial MR}\,\delta MR + \frac{\partial \dot{w}_{O2}}{\partial P_c}\,\delta P_c + \frac{\partial \dot{w}_{O2}}{\partial w_{gO}}\,\delta w_{gO} + \frac{\partial \dot{w}_{O2}}{\partial w_{gH}}\,\delta w_{gH} \qquad (56)$$

$$\dot{w}_{gH} = \dot{\bar{w}}_{gH} + \delta \dot{w}_{gH} \qquad (57)$$

$$\dot{w}_{gO} = \dot{\bar{w}}_{gO} + \delta \dot{w}_{gO} \qquad (58)$$

Math	Symbol	Units	Subroutine	Description
$\dot{\bar{w}}_{H2}$	WDH2	lb/sec	NASSME	nominal fuel flow rate
$\dot{\bar{w}}_{O2}$	WDO2	lb/sec	NASSME	nominal oxidizer flow rate
$\dot{\bar{w}}_{gO}$	WDGO	lb/sec	NASSME	nominal oxygen pressurant flow rate
$\dot{\bar{w}}_{gH}$	WDGH	lb/sec	NASSME	nominal hydrogen pressurant flow rate

$\frac{\partial \dot{w}_{H2}}{\partial MR}$	PWFMR	lb/sec	NASSME	partial fuel flow wrt mixture ratio
$\frac{\partial \dot{w}_{H2}}{\partial P_c}$	PWFPC	in^2/sec	NASSME	partial fuel flow wrt chamber pressure
$\frac{\partial \dot{w}_{H2}}{\partial \dot{w}_{gO}}$	PWFGO	-	NASSME	partial fuel flow wrt oxidizer pressurant flow rate
$\frac{\partial \dot{w}_{H2}}{\partial \dot{w}_{gH}}$	PWFGH	-	NASSME	partial fuel flow wrt fuel pressurant flow rate
$\frac{\partial \dot{w}_{O2}}{\partial MR}$	PWOMR	lb/sec	NASSME	partial oxidizer flow wrt mixture ratio
$\frac{\partial \dot{w}_{O2}}{\partial P_c}$	PWOPC	in^2/sec	NASSME	partial oxidizer flow wrt chamber pressure
$\frac{\partial \dot{w}_{O2}}{\partial \dot{w}_{gO}}$	PWOGO	-	NASSME	partial oxidizer flow wrt oxidizer pressurant flow rate
$\frac{\partial \dot{w}_{O2}}{\partial \dot{w}_{gH}}$	PWOGH	-	NASSME	partial oxidizer flow wrt fuel pressurant flow rate

The nominal values from the power gain law are based on the estimate of power level expressed by:

$$P_L = (P_{c\ nom} + \delta P_c)/RPLTAG \qquad (59)$$

Math	Symbol	Units	Subroutine	Description
P_L	PL	-	NASSME	power level estimate
δP_c	PCHAT	lb/in^2	NASSME	chamber pressure deviation estimate
RPLTAG	RPLTAG	lb/in^2	BLKDAT	rated power level tag value
$P_{c\ nom}$	PCTAG	lb/in^2	BLKDAT	chamber pressure tag value

Similarly, the vacuum thrust is computed from the power gain law based on the tag values, plus the deviations due to mixture ratio, chamber pressure, and pressurant flow rates. Thrust and specific impulse are given as:

$$T_{vac} = T_{vac\ nom} + \frac{\partial f_T}{\partial MR}\,\delta MR + \frac{\partial f_T}{\partial P_c}\,\delta P_c + \frac{\partial f_T}{\partial \dot{w}_{gO}}\,\delta \dot{w}_{gO} + \frac{\partial f_T}{\partial \dot{w}_{gH}}\,\delta \dot{w}_{gH} \qquad (60)$$

$$I_{sp} = T_{vac} / \dot{w} \qquad (61)$$

Math	Symbol	Units	Subroutine	Description
T_{vac}	TVACH	lb	NASSME	vacuum thrust estimate
$T_{vac\ nom}$	TVAC	lb	NASSME	vacuum thrust tag value

$\frac{\partial f_T}{\partial MR}$	PFTMR	lb	NASSME	partial thrust wrt mixture ratio
$\frac{\partial f_T}{\partial P_c}$	PFTPC	in^2	NASSME	partial thrust wrt chamber pressure
$\frac{\partial f_T}{\partial w_{gO}}$	PFTGO	sec	NASSME	partial thrust wrt oxidizer pressurant flow rate
$\frac{\partial f_T}{\partial w_{gH}}$	PFTGH	sec	NASSME	partial thrust wrt fuel pressurant flow rate
I_{sp}	XISP	sec	NASSME	vacuum specific impulse estimate
$\dot{w}$	WD	lb/sec	NASSME	total mass flow rate overboard estimate

Estimates of the thrust and mass overboard are provided to the XDVEC routine for the vehicle motion dynamics, and estimates of the specific impulse are provided to the output routine, OUTPUT, for output to the SSMEOT.DATA data file. These are given as:

Symbol	Units	Common	Description
TVACL	lb	SMEDAT	vacuum thrust estimates for each engine
OMASE	lb	SMEDAT	overboard mass estimates for each engine
XISPL	sec	SMEDAT	specific impulse estimates for each engine
CVISPL	sec^2	SMEDAT	variance uncertainties for specific impulse

The following partial derivatives are provided to ACCEL measurement update routine for filter estimate updates based on accelerometer measurements.

Symbol	Common	Description
PFTMR	METPRTL	partial of thrust wrt mixture ratio
PFTPC	METPRTL	partial of thrust wrt chamber pressure
PFTGO	METPRTL	partial of thrust wrt oxidizer pressurant flow rate
PFTGH	METPRTL	partial of thrust wrt fuel pressurant flow rate

Estimates of measurements and the partial derivatives are provided to the SSME measurement update routine for filter estimate updates based on the internal main engine measurements. Partial derivatives of the dynamics are provided to the routine LINFA and process noise parameters are computed for the error covariance matrix propagation between measurement update times. The time derivatives for the states are also provided. These are defined as:

Symbol	Common	Description
XLH	SMEMEA	estimates of chamber pressure, pressurant flows, and fuel measurements of volume, pressure and temperature for each engine
H	LINHMT	linearized measurement matrix rows 7-12, 13-18 and 19-24
F	LINFMT	linearized dynamics matrix rows 14-22, 23-31 and 32-40
S	CONST	process noise array elements 15-22, 24-31 and 33-40
XDOT	DRVITV	state time derivative elements 14-22, 23-31 and 32-40.

The units for the F, S and XDOT arrays correspond to the units indicated in Table 2.-2. The H array's units are indicated in Table 2.-3.

4.4.1.5 NASSRB Routine

The NASSRB routine used the data from BLKDAT to compute the SRB performance characteristics of thrust, mass flow, burn web thickness, and specific impulse. This model is based on a quasi-static internal ballistics representation for web thickness [9]. Estimates of the head pressure measurement for each SRB are computed assuming the one-dimensional flow correction from [10]. Also, included in this routine are the data tables correcting the vacuum thrust and nozzle coefficient correction based on previous NASA data analyses.

As with the NASSME routine, NASSRB computes partial derivatives for use in the linearized dynamics routine LINFA. The partial derivatives are computed from pseudo static equations given in equations (29-31). Partial derivatives are also provided for measurement update ACCEL routine, using equation (28), for the accelerometer measurements, and for the SRB routine, using equation (33), for head pressure measurements.

The following summarizes the dynamic model and code symbols;

Math	Symbol	Units	Subroutine	Description
$\dot{m}_s$	XMD	lb/sec	NASSRB	mass flow rate overboard
$\dot{\tau}$	TAUD	in/sec	NASSRB	web thickness burn rate
P_0	PO	lb/in^2	NASSRB	nozzle entrance pressure
a	ABAR	in/sec	BLKDAT	reference burn rate coefficient
P	PBAR	lb/in^2	BLKDAT	reference burn rate

				scale pressure
n	EP	-	BLKDAT	burn rate exponent
a	AN	$[a/P^n]$	NASSRB	nominal burn rate coefficient
$\frac{1}{1-n}$	OOMN	-	NASSRB	inverse of one minus EP
ρ_P	RHOP	lb/in^3	BLKDAT	propellant density
A_B	AB	in^2	NASSRB	propellant burn area
c^*	CSTAR	in/sec	NASSRB	propellant characteristic exhaust velocity
g	G	in/sec^2	BLKDAT	gravitational constant
A_T	AT	in^2	NASSRB	nozzle throat area

where propellant burn area, port volume, throat area, characteristic exhaust velocity, and thrust coefficient are linearly interpolated based on the estimated value of burn web thickness, τ.

The measurement model, given in equation (32), and code symbols are summarized as;

Math	Symbol	Units	Subroutine	Description
R	RBAR	**	BLKDAT	universal gas constant
π	PI	-	BLKDAT	constant "pi"
L	XLNGTH	in	BLKDAT	port volume reference length
V_P	PVOL	in^3	BLKDAT	port volume

The vacuum thrust, given in equation (28), derived corrections and specific impulse are summarized as;

$$T_{SF} = .9925 \qquad \Delta T_{SRB} = f(\tau) \tag{62}$$

$$I_{sp} = T_{vac} / \dot{m}_s \tag{63}$$

Math	Symbol	Units	Subroutine	Description
T_{vac}	TVACS	lb	NASSRB	estimated vacuum thrust
T_{SF}	TSF	-	NASSRB	nozzle coefficient correction
ΔT_{SRB}	DTSRB	lb	NASSRB	thrust correction
I_{sp}	XISPS	sec	NASSRB	estimated specific impulse

where the correction, ΔT_{SRB}, is obtained by linearly interpolating, using INTRP1, from data tables contained in the NASSRB routine based on the estimated value of τ.

Estimates of the thrust and mass overboard are provided to the XDVEC routine for the vehicle motion dynamics, and estimates of the specific impulse are provided to the routine, OUTPUT, for to the SRBOUT.DATA data file. These are summarized as;

Symbol	Common	Description
TVACS	SRBDAT	vacuum thrust estimates for each SRB
OMASS	SRBDAT	overboard mass estimates for each SRB
XISPS	SRBDAT	specific impulse estimates for each SRB
CVISPS	SRBDAT	variance uncertainties for specific impulse

The following partial derivatives are provided to the ACCEL measurement update routine for filter state estimate updates based on the accelerometer measurements:

Symbol	Units	Common	Description
PFT	lb/in	RBPRTL	partial of thrust wrt burn web thickness
PFA	**	RBPRTL	partial of thrust wrt burn rate coefficient
PFCM	lb	RBPRTL	partial of thrust wrt nozzle coefficient

The nozzle entrance pressure is computed to convert rock and tilt deflections into pitch and yaw gimbal angles. This value is communicated to the CONTRL routine via the common PONOZZ for each SRB.

Estimates of measurements and their partial derivatives are provided to the SRB measurement update routine for filter state estimate updates based on these internal measurements. Partial derivatives of the dynamics are provided to the routine LINFA and process noise parameters are computed to the error covariance matrix propagation. The time derivatives for the state elements are also provided. These are summarized as;

Symbol	Common	Description
POHHAT	SRBMEA	estimates of each SRB's chamber head pressure
H	LINHMT	linearized measurement matrix rows 34 and 35
F	LINFMT	linearized dynamics matrix rows 62-66 and 67-71
S	CONST	process noise array elements 64-66 and 69-71
XDOT	DRVITV	state time derivative elements 62-66 and 67-71.

The units for the F, S, and XDOT arrays correspond to the units indicated in Table 2.-2. The H array's units are indicated in Table 2.-3.

4.4.1.6 NAERO Routine

This routine, as with ATMOS, NPLUME, NASSME, and NASSRB, contains a reference model. The filter estimates deviations from this model. The aerodynamic coefficients are formed from a polynomial expression given in equations (17), (18), and (19). Filter estimates of angle of attack and side slip are used to form the total coefficient vector as given in equation (20). The resulting coefficient vector is supplied to the XDVEC routine. Linearized coefficients about angle of attack and side slip are supplied to the LINFA routine for propagating the error covariance matrix.

The coefficients in equations (17), (18), and (19) are defined as:

Math	Symbol	Units	Subroutine	Description
C_{A_o}	CA0	-	NAERO	zero alpha/beta coefficient
$C_{A\alpha}$	CAA	deg^{-1}	NAERO	axial coefficient of α
$C_{A\alpha}{}^2$	CAA2	deg^{-2}	NAERO	axial coefficient of α^2
$C_{A\beta}{}^2$	CAB2	deg^{-2}	NAERO	axial coefficient of β^2
$C_{A\alpha\beta}{}^2$	CAAB2	deg^{-3}	NAERO	axial coefficient of $\alpha\beta^2$
C_{N_o}	CN0	-	NAERO	zero alpha/beta coefficient
$C_{N\alpha}$	CNA	deg^{-1}	NAERO	normal coefficient of α
$C_{N\alpha\beta}{}^2$	CNAB2	deg^{-3}	NAERO	normal coefficient of $\alpha\beta^2$
C_{Y_o}	CY0	-	NAERO	zero alpha/beta coefficient
$C_{Y\beta}$	CYB	deg^{-1}	NAERO	side force coefficient of β
$C_{Y\alpha\beta}$	CYAB	deg^{-2}	NAERO	side force coefficient of $\alpha\beta$
$C_{Y\beta\alpha}{}^2$	CYA2B	deg^{-3}	NAERO	side force coefficient of $\alpha^2\beta$

The filter's estimates are corrections added onto the zero alpha/beta terms.

Each of the terms above are tabulated as a function of Mach number, and are linearly interpolated using INTRP1. The symbols used to define the tabular arrays are the same as the variable, except that the tabular array symbols have a "T" preceding the variable symbol, i.e. TCA0 for the zero alpha/beta axial force coefficient. A "2" is appended onto the variable for the second stage coefficients and tabular arrays, i.e. TCN20 for the aero alpha/beta normal force coefficient.

These coefficients and partials are equated to the appropriate vector elements and communicated to other routines. These body referenced vector components are defined as:

Symbol	Units	Common	Description
CF0	-	AERO	zero alpha/beta force coefficient
CFALP	deg^{-1}	AERO	force coefficient slope wrt alpha
CFBET	deg^{-1}	AERO	force coefficient slope wrt beta
CFQ	sec	AERO	partial force coefficient wrt body rates

Additional outputs from the NAERO routine are summarized below as:

Symbol	Common	Description
F	LINFMT	elements 41 thru 43 of linearized dynamics matrix
S	CONST	process noise elements 41 thru 43
XDOT	DRVITV	elements 41 thru 43 of state vector time derivatives

The units for the F, S, and XDOT arrays correspond to the units indicated in Table 2.-2.

4.4.1.7 NMASS Routine

This routine computes the mass properties as a function of mass overboard. This routine contains data for mass moment of inertia for stages 1 and 2 that are no longer used in the current estimation approach. The function of this routine is to compute the vehicle's current mass by subtracting the mass overboard from the initial weight initialized in the BLKDAT routine. The vehicle's center of gravity components are computed based on the mass overboard.

The initial mass properties, weight, for each state is communicated via a common statement:

Symbol	Units	Common	Description
XMASSI	lb	IPROPR	initial stage 1 weight
XMASS2	lb	IPROPR	initial stage 2 weight

Data contained in NMASS, not all currently used, is described below:

Symbol	Units	Subroutine	Description
XMASS	lb sec^2/ft	NMASS	current vehicle mass
OMASS	lb	XDVEC	total weight overboard, starting at 0, summed from all propulsive systems
XIMTRX	slug ft^2	NMASS	mass moment of inertia matrix
XIMATI	1/slug ft^2	NMASS	inverse of inertia matrix
RCG	ft	NMASS	body referenced coordinates of vehicle center of gravity

The values used in the inertia matrix and the center of gravity vector are obtained by linear interpolation, using INTRP1, from tabular arrays as a function of weight overboard. The following summarizes the description of the tabular array data contained in NMASS.

Symbol	Units	Subroutine	Description
DELWT	lb	NMASS	1st stage weight overboard
TIX	slug ft^2	NMASS	1st stage x axis inertia
TIY	slug ft^2	NMASS	1st stage y axis inertia
TIZ	slug ft^2	NMASS	1st stage z axis inertia
TIXY	slug ft^2	NMASS	1st stage x-y products of inertia
TIXZ	slug ft^2	NMASS	1st stage x-z products of inertia
TIYZ	slug ft^2	NMASS	1st stage y-z products of inertia
TXCG	ft	NMASS	1st stage x center of gravity
TYCG	ft	NMASS	1st stage y center of gravity
TZCG	ft	NMASS	1st stage z center of gravity
DELTWT	lb	NMASS	1st stage weight overboard for moments of inertia
AIXX	slug ft^2	NMASS	2nd stage x axis inertia
AIYY	slug ft^2	NMASS	2nd stage y axis inertia
AIZZ	slug ft^2	NMASS	2nd stage z axis inertia
AIXY	slug ft^2	NMASS	2nd stage x-y products of inertia
AIXZ	slug ft^2	NMASS	2nd stage x-z products of inertia
AIYZ	slug ft^2	NMASS	2nd stage y-z products of inertia
AXCG	ft	NMASS	2nd stage x center of gravity
AYCG	ft	NMASS	2nd stage y center of gravity
AZCG	ft	NMASS	2nd stage z center of gravity
DELTW2	lb	NMASS	2nd stage weight overboard for moments of inertia

4.4.1.8 Additional Routines

The routines described previously are the primary routines that contain modeling information. Several additional routines support computations within XDVEC. These are; AGRAV which computes the earth centered gravitational accelerations based on a second order oblate spheroid earth model, COOR which computes latitude, longitude and altitude from earth centered position, and QMTRX which computes the dynamics matrix, equation (12), for the quaternion state elements.

Additional library routines are used in XDVEC and elsewhere in the programs. These routines are described in Appendix D.

4.4.2 ZDVEC Routine

This routine provides the dynamic derivatives associated with the external measurement bias states. These states include the accelerometer and radar measurement biases.

The following derivatives are formed within ZDVEC:

Symbol	Common	Description
XDOT	DRVITV	elements 50 thru 52 for accelerometer bias elements 53 thru 55 for 1st radar biases elements 56 thru 58 for 2nd radar biases elements 59 thru 61 for 3rd radar biases

The units for the XDOT array correspond to the units indicated in Table 2.-2.

4.4.3 PDMTRX Routine

The Kalman filter algorithm requires not only the state time derivatives, but also the time derivatives for the error covariance matrix. These derivatives are established by forming the linearized dynamics matrices and completing the matrix products as shown in Table 2.-1. This routine forms the linearized dynamics matrices, using some partial derivatives formed when XDVEC routine is called.

The linearized dynamics matrix for the states listed in Table 2.-2 is shown in Figure 4.4.3-1. This linearized dynamics matrix, which represents the partial derivatives of the state differential equations wrt each state element, is partitioned into the dynamics matrices for; the vehicle motion dynamics formed in LINFS, i.e. quaternion elements, the system error parameters formed in LINFA, i.e. SRB burn rate coefficient, and the external measurement biases formed in LINFB, i.e. accelerometer bias. The submatrices indicated by $[\,]_{1,3}$ and $[\,]_{A,B}$ correspond to the SSME and SRB submatrices respectively. These submatrices are illustrated in Figure 4.4.3-2.

As the individual linearized dynamics matrices are formed, they are imbedded within the overall linearized system dynamics matrix, F, and communicated to PDMTRX via labeled common statement LINFMT to form the time derivative of the error covariance matrix. In the following sections, each routines that form the partitioned submatrices will be discussed.

ORIGINAL PAGE IS
OF POOR QUALITY

	$\delta \underline{r}^{(I)}$	$\delta \underline{v}^{(B)}$	$\delta \underline{q}$	$[\,]_{1,3}$	$\delta \underline{c}_f$	$\delta \underline{f}_p$	$\delta \underline{v}_w$	$\underline{a}_{c\,b}$	az_b	el_b	$range_b$	$[\,]_{2,3}$	$[\,]_{A,B}$
$\delta \underline{r}^{(I)}$	0	$\frac{\partial \dot{\underline{r}}^{(I)}}{\partial \underline{v}^{(B)}}$	$\frac{\partial \dot{\underline{r}}^{(I)}}{\partial \underline{q}}$		0	0	0						
$\delta \underline{v}^{(B)}$	$\frac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{r}^{(I)}}$	$\frac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{v}^{(B)}}$	$\frac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{q}}$		$\frac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{c}_f}$	$\frac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{f}_p}$	$\frac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{v}_w}$						
$\delta \underline{q}$	0	0	$\frac{\partial \dot{\underline{q}}}{\partial \underline{q}}$		0	0	0						
$[\,]_{1,3}$				$[\,]_{1,3}$									
$\delta \underline{c}_f$					$[\tau_a]$	0	0						
$\delta \underline{f}_p$					0	$[\tau_p]$	0						
$\delta \underline{v}_w$					0	0	$[\tau_w]$						
$\underline{a}_{c\,b}$								$[\tau_{ac}]$					
az_b													
el_b										$[\tau_r]$			
$range_b$													
$[\,]_{2,3}$												$[\,]_{2,3}$	
$[\,]_{A,B}$													$[\,]_{A,B}$

Figure 4.4.3-1: Linearized Dynamics Matrix - F

	δw_L	δMR	δP_c	δw_{gO}	δw_{gH}	$V_{F\,b}$	$P_{F\,b}$	$T_{F\,b}$	$P_{c\,b}$	$[\,]_{2,3}$
$\delta \underline{x}^{(I)}$	0	0	0	0	0	0	0	0	0	
$\delta \underline{y}^{(B)}$	$\frac{\partial \dot{\underline{y}}^{(B)}}{\partial w_L}$	$\frac{\partial \dot{\underline{y}}^{(B)}}{\partial MR}$	$\frac{\partial \dot{\underline{y}}^{(B)}}{\partial P_c}$	$\frac{\partial \dot{\underline{y}}^{(B)}}{\partial w_{gO}}$	$\frac{\partial \dot{\underline{y}}^{(B)}}{\partial w_{gH}}$	0	0	0	0	
$\delta \underline{q}$	0	0	0	0	0	0	0	0	0	
δw_L	0	$\frac{\partial \dot{w}_L}{\partial MR}$	$\frac{\partial \dot{w}_L}{\partial P_c}$	$\frac{\partial \dot{w}_L}{\partial w_{gO}}$	$\frac{\partial \dot{w}_L}{\partial w_{gH}}$	0	0	0	0	
δMR	0	τ_{MR}	0	0	0	0	0	0	0	
δP_c	0	0	τ_P	0	0	0	0	0	0	
δw_{gO}	0	0	0	τ_{gO}	0	0	0	0	0	
δw_{gH}	0	0	0	0	τ_{gH}	0	0	0	0	
$V_{F\,b}$	0	0	0	0	0	τ_V	0	0	0	
$P_{F\,b}$	0	0	0	0	0	0	τ_P	0	0	
$T_{F\,b}$	0	0	0	0	0	0	0	τ_T	0	
$P_{c\,b}$	0	0	0	0	0	0	0	0	τ_b	
$[\,]_{2,3}$										$[\,]_{2,3}$

	δm_s	$\delta \tau$	δa	δc_m	$P_{H\,b}$	$[\,]_B$
$\delta \underline{x}^{(I)}$	0	0	0	0	0	
$\delta \underline{y}^{(B)}$	$\frac{\partial \dot{\underline{y}}^{(B)}}{\partial m_s}$	$\frac{\partial \dot{\underline{y}}^{(B)}}{\partial \tau}$	$\frac{\partial \dot{\underline{y}}^{(B)}}{\partial a}$	$\frac{\partial \dot{\underline{y}}^{(B)}}{\partial c_m}$	0	
$\delta \underline{q}$	0	0	0	0	0	
δm_s	0	$\frac{\partial \dot{m}_s}{\partial \tau}$	$\frac{\partial \dot{m}_s}{\partial a}$	0	0	
$\delta \tau$	0	$\frac{\partial \dot{\tau}}{\partial \tau}$	$\frac{\partial \dot{\tau}}{\partial a}$	0	0	
δa	0	0	τ_a	0	0	
δc_m	0	0	0	τ_c	0	
P_{oH}	0	0	0	0	τ_H	
$[\,]_B$						$[\,]_B$

Figure 4.4.3-2: SSME and SRB Submatrices

4.4.3.1 LINFS Routine

This routine forms the partial derivatives of the vehicle motion states, position, velocity and attitude (quaternion), with respect to themselves. After these are formed, the routine IMBED stores the partial derivatives in the appropriated location with the linearized dynamics matrix F. The following describes the elements formed:

$$\frac{\partial \dot{\underline{r}}^{(I)}}{\partial \underline{v}^{(B)}} = {}^{I}C^{B} \tag{64}$$

$$\frac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{r}^{(I)}} = \frac{Av_m^2}{2m}\frac{\partial \rho}{\partial h}\underline{c}_f + \frac{\rho A v_m}{m}\underline{c}_f\frac{\partial v_m}{\partial h} + \frac{\rho A v_m^2}{m}\frac{\partial \underline{c}_f}{\partial \alpha}\frac{\partial \alpha}{\partial h} + \frac{\rho A v_m^2}{2m}\frac{\partial \underline{c}_f}{\partial \beta}\frac{\partial \beta}{\partial h}$$
$$+ \frac{1}{m}\left[{}^{B}C^{C}\frac{\partial \underline{f}_T^{(C)}}{\partial p_s}\frac{\partial p_s}{\partial h} + \frac{\partial \underline{f}_P}{\partial \alpha}\frac{\partial \alpha}{\partial h} + \frac{\partial \underline{f}_P}{\partial \beta}\frac{\partial \beta}{\partial h}\right]\left[\frac{\underline{r}^{(I)}}{|\underline{r}^{(I)}|}\right]^T \tag{65}$$
$$+ \frac{\partial\, {}^{B}C^{EF}\, \underline{g}^{(EF)}\, (\underline{r}^{(EF)})}{\partial \underline{r}^{(I)}}$$

$$\frac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{v}^{(B)}} = \frac{\rho A v_m}{m}\underline{c}_f\frac{\partial v_m}{\partial \underline{v}^{(B)}} + \frac{\rho A v_m^2}{2m}\frac{\partial \underline{c}_f}{\partial \alpha}\frac{\partial \alpha}{\partial \underline{v}^{(B)}} + \frac{\rho A v_m^2}{2m}\frac{\partial \underline{c}_f}{\partial \beta}\frac{\partial \beta}{\partial \underline{v}^{(B)}}$$
$$+ \frac{1}{m}\left[\frac{\partial \underline{f}_P}{\partial \alpha}\frac{\partial \alpha}{\partial \underline{v}^{(B)}} + \frac{\partial \underline{f}_P}{\partial \beta}\frac{\partial \beta}{\partial \underline{v}^{(B)}}\right] - [\underline{\omega}\, \mathrm{x}] \tag{66}$$

Math	Symbol	Units	Subroutine	Description
$\frac{\partial \dot{\underline{r}}^{(I)}}{\partial \underline{v}^{(B)}}$	CBI	-	XDVEC	partial of equation (10) wrt body velocity
$\frac{\partial \dot{\underline{r}}^{(I)}}{\partial \underline{q}}$	PCBIQ	ft/sec	AXCBIQ	partial of equation (10) wrt quaternions
$\frac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{r}^{(I)}}$	-	sec^{-2}	LINFS	partial of equation (11) wrt position
$\frac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{v}^{(B)}}$	-	sec^{-1}	LINFS	partial of equation (11) wrt body velocity
$\frac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{q}}$	-	sec^{-2}	LINFS	partial of equation (11) wrt quaternions
$\frac{\partial \dot{\underline{q}}}{\partial \underline{q}}$	QDMTRX	sec^{-1}	QMTRX	partial of equation (12) wrt quaternions

4.4.3.2 LINFA Routine

Partial derivatives of the vehicle motion state time derivatives with respect to the system error parameters are limited to the partial derivatives of body velocity partials. The routine IMBED is used to insert these partials into the appropriate elements of the linearized dynamics matrix F. In the order in which they are included in the state vector, below is a summary of the partials formed within LINFA:

Math	Symbol	Units	Subroutine	Description
$\dfrac{\partial \dot{\underline{v}}^{(B)}}{\partial(\ )}$	-	**	NASSME	partial of equation (11) wrt SSME states
$\dfrac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{c}_f^{(B)}}$	PVBDCF	ft/sec^2	LINFA	partial of equation (11) wrt aero force coefficients
$\dfrac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{f}_p^{(B)}}$	PVBDFP	ft/sec^2/lb	LINFA	partial of equation (11) wrt plume forces
$\dfrac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{v}_w^{(LL)}}$	PVBDVW	sec^{-1}	LINFA	partial of equation (11) wrt wind velocity
$\dfrac{\partial \dot{\underline{v}}^{(B)}}{\partial(\ )}$	-	**	NASSRB	partial of equation (11) wrt SRB states

** see the individual routines, i.e. NASSME, for the units definitions

4.4.3.3 LINFB Routine

This routine forms the elements of the linearized dynamics matrix associated with the external measurement bias states. These matrix elements are described as:

Symbol	Common	Description
F	LINFMT	elements 50,50 thru 52,52 for accelerometer elements 53,53 thru 55,55 for 1st radar elements 56,56 thru 58,58 for 2nd radar elements 59,59 thru 61,61 for 3rd radar

Also, formed are the process noise array elements which represent the continued uncertainty added to the error covariances associated with modeling these errors as exponentially correlated noises. The array elements for the bias states are described as:

Symbol	Common	Description
S	CONST	elements 50 thru 52 for accelerometer biases elements 53 thru 55 for 1st radar biases elements 56 thru 58 for 2nd radar biases elements 59 thru 61 for 3rd radar biases

The units for the F and S arrays correspond to the units indicated in Table 2.-2.

4.5 RK4FIL Routine

RK4FIL implements a fourth order Runge-Kutta numerical integration algorithm. This algorithm is used to integrate the time derivatives of the filter states and the upper triangular portion of the symmetric error covariance matrix.

The time derivatives are stored in the DER array in the unlabeled common. The resulting integrals are stored in the VAR array in this common. A work space required by this algorithm implementation uses the TEMP array in this common statement. The total number of variables to integrate is identified as NDER also contained in this common.

4.6 GETDAT Routine

This routine reads the measurement input data file, REALMEA.DAT, on logical unit 2. These data, all synchronized to the same time instances, are produced by the preprocessing program MERGE which merges data from other preprocessing programs. These programs are described in Appendix A.

At each measurement time, the following data are read and communicated to the applicable measurement update routines:

Symbol	Units	Common	Description
ACM	ft/sec^2	ACMEAS	IMU platform delta velocity components (3)
YMSME	**	MEMEAS	main engine measurements (6 per engine)
AZM	deg	RDMEAS	radar azimuth angular measurement (3 radars)
ELM	deg	RDMEAS	radar elevation angular measurement
RNGM	ft	RDMEAS	radar range measurement
YMSRB	lb/in^2	SRMEAS	SRB head pressure measurements (1 per motor)

** see the units as defined in Table 2.-3.

4.7 UPDATE Routine

Based on the flags set in the BLKDAT routine, the UPDATE routine calls the routines which accomplish the measurement updates for the corresponding measurements. Contained in the common NMEASR, are the number of measurement types to be processed, NMEAS, and the flag for each type contained in the array IMEAS. If the flag is set as indicated by a "1", then the appropriate routine to accomplish the update is called.

For each measurement type, i.e. radar, the processing sequence is as follows. The current filter states and upper triangular portion of the error covariance matrix, resulting either from time integration or from a previous measurement update, are extracted from the VAR array in the unlabeled common. These quantities are then stored in the XKM and PKM arrays, in the PREUP common, respectively. Each measurement update routine is then called and new values of the XKM and PKM arrays are generated. These new values are then stored back into the VAR array where they are available for numerical integration or for additional measurement updating.

The total number of estimation state elements is the sum of the number of vehicle motion states, NS, and the number of other model states, NPAR. These two numbers are communicated via the labeled common LINFMT.

Each of the measurement updating routines, ACCEL, SSME, RADAR and SRB, process measurements using the same sequence. This sequence is described as follows.

The Kalman filter algorithm described in section 2. is a implementation of the U-D factorized form of the Kalman filter [5]. This form of the algorithm processes each measurement, i.e. radar azimuth, individually. Therefore, each individual measurement's linearized measurement matrix, a row vector formed from the partial derivatives of that measurement wrt the state elements, is extracted from the full linearized measurement matrix for that measurement's updating. The outputs of the algorithm are the upper triangular elements of the error covariance matrix, the measurement residual covariance, and the unweighted Kalman gain vector. Since these quantities are used repeatedly in the UPDATE routines, a common storage UDWORK is used for these arrays.

Initially, in the ACCEL routine, the linearized measurement matrix, H, is initialized to zero. This matrix is not specifically needed in the U-D algorithm, but is formed to aid in the analysis of the "observability" of the system, for a check on the adequacy of the implementation of the mathematical linearizations versus results from numerical differentiation, and for temporary storage of the appropriate measurement's partial derivatives with respect to the state elements.

For each measurement, i.e. accelerometers 1 thru 3, the most recent values of the state variables and other related variables are computed by calls to the XDVEC and ZDVEC routines. Prior to this call, the state variables are extracted from the XKM array and stored into the X array where the XDVEC and ZDVEC routines expect the current variables to be located. Using these most recently computed values of the appropriate variables, the estimates of the measurements are formed. This includes the addition of the bias state elements associated with the measurement, i.e. accelerometer bias.

Next, the row vector corresponding to the measurement's partial derivatives wrt the state elements is extracted and stored in the RG array. The error covariance matrix, stored in upper triangular form, is extracted and stored in the U array. In this form, the routine COV2UD is called to convert the error covariance matrix into the U-D factorized form. At this stage, the necessary inputs are available to compute the measurement update to the error covariance matrix.

This update is accomplished by calling the UDMEAS routine. This routine computes the updated form of the U-D factorized error covariance matrix. Returning this array as U, the routine UD2COV is called to convert U into the upper triangular portion of the error covariance matrix, PO. This PO matrix is then used to re-establish the full form of the error covariance matrix PKM.

Also output from UDMEAS are the arrays used to form the Kalman gain for updating the state elements. This is accomplished by forming the residual, RESID, from the difference between the measurement and the estimate of the measurement formed from the state elements. This value is multiplied by the Kalman gain and this correction is then added to the previous value of the state estimates.

Were it not for the use of quaternions to describe the vehicle's attitude, this would complete the state update process for the measurement processed. Quaternions, with 4 elements, can be uniquely described by using only 3 elements. The fourth is determined from the constraint that the sum of the squares of all four elements must equal 1. The update to the fourth quaternion element can be obtained from the constraint relationship. This update is given by [6]

$$\delta q_3 = - (q_0 \, \delta q_0 + q_1 \, \delta q_1 + q_2 \, \delta q_2)/q_3 \tag{67}$$

The error covariance matrix row and column that contains this element are similarly updated deterministically.

After each measurement has been processed, the output routine, OUTPUT, is called to output the relevant data to an output disk file, FILOUT.DATA. The above sequence is repeated for each measurement processed.

The linearized measurement matrix, H, is illustrated in Figure 4.7-1, partitioned as in the case of the F matrix in Figure.4.4.3-1. The submatrices indicated by $[\,]_{1,3}$ and $[\,]_{A,B}$ correspond to the SSME and SRB linearizations respectively. These submatrices are illustrated in Figure 4.4.3-2. Each of the following sections describe the type of measurements processed.

	$\delta \underline{r}^{(I)}$	$\delta \underline{v}^{(B)}$	δq	$[\,]_{1,3}$	$\delta \underline{c}_f$	$\delta \underline{f}_P$	$\delta \underline{v}_w$	$\underline{a}_{c\,b}$	az_b	el_b	$range_b$	$[\,]_{2,3}$	$[\,]_{A,B}$
$\delta \underline{a}_m^{(S)}$	$\frac{\partial \underline{a}_m^{(S)}}{\partial \underline{r}^{(I)}}$	$\frac{\partial \underline{a}_m^{(S)}}{\partial \underline{v}^{(B)}}$	$\frac{\partial \underline{a}_m^{(S)}}{\partial \underline{q}}$		$\frac{\partial \underline{a}_m^{(S)}}{\partial \underline{c}_f}$	$\frac{\partial \underline{a}_m^{(S)}}{\partial \underline{f}_P}$	$\frac{\partial \underline{a}_m^{(S)}}{\partial \underline{v}_w}$	I					
$[\,]_{1,3}$				$[\,]_{1,3}$									
az_m	$\frac{\partial az}{\partial \underline{r}^{(I)}}$	0	0						1	0	0		
el_m	$\frac{\partial el}{\partial \underline{r}^{(I)}}$	0	0						0	1	0		
$range_m$	$\frac{\partial range}{\partial \underline{r}^{(I)}}$	0	0						0	0	1		
$[\,]_{2,3}$		$[\,]_{2,3}$										I	
$[\,]_{A,B}$													$[\,]_{A,B}$

Figure 4.7-1: Linearized Measurement Matrix - H

ORIGINAL PAGE IS
OF POOR QUALITY

	δw_L	δMR	δP_c	δw_{gO}	δw_{gH}	$V_{F\,b}$	$P_{F\,b}$	$T_{F\,b}$	$P_{c\,b}$	$[\,]_{2,3}$
$\delta \underline{a}_m^{(S)}$	$\frac{\partial \underline{a}_m^{(S)}}{\partial w_L}$	$\frac{\partial \underline{a}_m^{(S)}}{\partial MR}$	$\frac{\partial \underline{a}_m^{(S)}}{\partial P_c}$	$\frac{\partial \underline{a}_m^{(S)}}{\partial w_{gO}}$	$\frac{\partial \underline{a}_m^{(S)}}{\partial w_{gH}}$	0	0	0	0	
δP_c	0	0	1	0	0	0	0	0	1	
δw_{gO}	0	0	0	1	0	0	0	0	0	
δw_{gH}	0	0	0	0	1	0	0	0	0	
V_F	0	0	0	0	0	1	0	0	0	
P_F	0	0	0	0	0	0	1	0	0	
T_F	0	0	0	0	0	0	0	1	0	
$[\,]_{2,3}$										$[\,]_{2,3}$

	δm_s	δr	δa	δc_m	$P_{H\,b}$	$[\,]_B$
$\delta \underline{a}_m^{(S)}$	$\frac{\partial \underline{a}_m^{(S)}}{\partial m_s}$	$\frac{\partial \underline{a}_m^{(S)}}{\partial r}$	$\frac{\partial \underline{a}_m^{(S)}}{\partial a}$	$\frac{\partial \underline{a}_m^{(S)}}{\partial c_m}$	0	
δP_{oH}	0	$\frac{\partial P_{oH}}{\partial r}$	$\frac{\partial P_{oH}}{\partial a}$	0	1	
$[\,]_B$						$[\,]_B$

Figure 4.7-2: SSME and SRB Submatrices

ORIGINAL PAGE IS
OF POOR QUALITY

4.7.1 ACCEL Routine

The measurement updating process accomplished in this routine is the sequence just described. For the accelerometer measurement, equation (32), the following summarizes the mathematical and code symbology;

Math	Symbol	Units	Subroutine	Description
$\underline{a}_m^{(S)}$	ACMHAT	ft/sec^2	ACCEL	sensed acceleration
${}^{S}C^{B}$	CBS	-	CBIMXQ	transformation from body to sensed frame
	ACC	ft/sec^2	XDVEC	combined aerodynamic and plume accelerations
$\underline{\omega}$	OMEGA	sec^{-1}	CONTRL	body rotation rate from body rate gyros
$\underline{r}_S - \underline{r}_{cg}$	DRS	ft	ACCEL	IMU position relative to body center of gravity
$\underline{b}_a^{(S)}$		ft/sec^2	ACCEL	accelerometer bias

where the angular acceleration terms have been omitted.

Partial derivatives of the measurements are formed and communicated to ACCEL via several common statements. From LINFS, the following [] terms are provided to form the first two vector components of the motion dynamics:

$$\frac{\partial \underline{a}_m^{(S)}}{\partial \underline{r}^{(I)}} = {}^{S}C^{B} \left[\frac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{r}^{(I)}} \right] \tag{68}$$

$$\frac{\partial \underline{a}_m^{(S)}}{\partial \underline{v}^{(B)}} = {}^{S}C^{B} \left[\frac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{v}^{(B)}} \right] \tag{69}$$

Math	Symbol	Units	Common	Description
$\dfrac{\partial \underline{a}_m^{(S)}}{\partial \underline{r}^{(I)}}$	PAMBRI	sec^{-2}	APARTL	partial derivative of body acceleration wrt BR position
$\dfrac{\partial \underline{a}_m^{(S)}}{\partial \underline{v}^{(B)}}$	PAMBVB	sec^{-1}	APARTL	partial derivative of body acceleration wrt body velocity

The partial derivative of body acceleration with respect to the quaternion elements, the last motion dynamic vector components, is provided by AXCBIQ and is obtained from:

$$\frac{\partial \underline{a}_m^{(S)}}{\partial \underline{q}} = \frac{\partial\, {}^S C^B}{\partial \underline{q}} [\ \ ] \tag{70}$$

where the bracketed terms are those in equation (32).

The following summarizes the partial derivatives formed in LINFA for the aerodynamic, plume and wind contributors to acceleration:

Math	Symbol	Units	Common	Description
$\dfrac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{c}_f}$	PVBDCF	ft/sec^2	PARARO	partial of equation (11) wrt aerodynamic coefficient
$\dfrac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{f}_P}$	PVBDFP	$ft/sec^2/lb$	PARPLM	partial of equation (11) wrt plume forces

$\dfrac{\partial \dot{\underline{v}}^{(B)}}{\partial \underline{v}_w}$	PVBDVW	sec^{-1}	PARVWX	partial of equation (11) wrt wind velocity components

Individual contributors of the state elements from the main engines', described in section 4.4.1.4, and SRBs', described in section 4.4.1.5, are combined within ACCEL to form the partial derivatives of body acceleration with respect to these elements.

The partial derivative of acceleration measurement with respect to its bias is unity, UNIT.

4.7.2 SSME Routine

As with the ACCEL routine, the most recent values of the variables associated with the main engine are recomputed prior to using them in computing the estimate of the measurement to be processed. Then, as with ACCEL, each measurement is processed. The estimates of the states and measurements are computed within the NASSME routine rather than being formed in the SSME routine as they were in the ACCEL routine. Also, the partial derivatives of the measurements with respect to the main engine states are formed with the NASSME routine.

After updating, the outputs are provided via common to the OUTPUT routine.

4.7.3 SRB Routine

This routine's processing parallels that previously described for the SSME routine. All the model relevant computations are accomplished in the NASSRB routine with only the measurement updating being accomplished within the SRB routine.

4.7.4 RADAR Routine

The structure of this routine parallels the ACCEL routine in that the estimates of the measurements and the necessary partial derivatives are formed within this routine. The identical routine is used in the LFILTER program for providing trajectory reconstruction - BET. It is discussed in Appendix C.

4.8 OUTPUT Routine

After each of the measurement updates from the measurement update routines, OUTPUT is called to output the data to the output disk file FILOUT.DAT. In addition to the time, a count of which measurement is being processed output is contained in this file. The updated state estimates, variances, the measurement variance, and the measurement residual are also output. These data are later read by a plotting program which produces output plots of the filter's states and residuals with the associated uncertainties.

Other variables of special interest are also output. These are the SRB and main engine performance data, i.e. specific impulse.

The filter's outputs of the PFILTER program, as provided by the OUTPUT routine, are summarized as:

Symbol	Units	Common	Description
VAR(1)	sec	unlabeled	current time
TLAST	sec		last time output data written to disk
KO	-		current count of the measurements updated
NMEAS	-	LINHMT	total number of measurements to process
XKM	*	PREUP	current state vector estimate
PKM	*	PREUP	current error variances (diagonal of PKM)
COVZ	**	UDWORK	variance of measurement residual
RESID	**	UDWORK	measurement residual

* see Table 2.-2 for definition of units
** see Table 2.-3 for definition of units

Special outputs are provided for a quantitative evaluation of the propulsion performance estimates. For the SRB the following variables are output in the file SRBOUT.DATA:

Symbol	Units	Common	Description
VAR(1)	sec	unlabeled	current time
POHHAT	lb/in^2	SRBMEA	head pressure
XISPS	sec	SRBDAT	specific impulse
TVACS	lb	SRBDAT	vacuum thrust
XMD	lb/sec	SRBDAT	mass flow rate overboard
AEXIT	in^2	SRBDAT	exit area

These variables are output for each motor at each time step using the following format:

```
FORMAT( F7.3, 2F10.3, F13.1, 2F10.3 )
```

The main engine special output variables are written to the file SSMEOT.DATA at 27 words per record. These variables are summarized as:

Symbol	Units	Common	Description
VAR(1)	sec	unlabeled	current time
PLN	-	GIMBAL	commanded power level
TVACL{1-3}	lb	SMEDAT	vacuum thrust
XISPL{1-3}	sec	SMEDAT	specific impulse
WDO2H{1-3}	lb/sec	SMEDAT	oxidizer mass flow rate overboard
WDH2H{1-3}	lb/sec	SMEDAT	fuel mass flow rate overboard
XLH(2{1-3})	lb/sec	SMEMEA	oxygen pressurant mass flow rate
XLH(3{1-3})	lb/sec	SMEMEA	hydrogen pressurant mass flow rate
WD{1-3}	lb/sec	SMEDAT	total mass flow rate overboard

C-2

5. Program Operation and Results

The PFILTER program and the associated preprocessing programs are operational on the EADS IBM 3090 and CRAY X-MP computer systems. The preprocessing programs operate on the IBM where the data bases are located. The PFILTER programs' files are permanently located on the IBM but the program operates on the CRAY. Utilizing job control language (JCL), the necessary files and input data are retrieved with the program execution. The resulting output files are cataloged on the IBM for further processing. This further processing included plotting the filter's states and measurement residuals to assess the quality of the filter's results. The JCL for PFILTER's execution is presented in Figure 5.-1.

The program which produces the BET is located on the SAIL1 VAX computer system. The BET program was not transferred because the ease of access to the Radar tracking data on the SAIL1 versus the EADS IBM 3090. As soon as it arrives, the tracking data tape can be quickly read into the VAX computers and data processing can begin immediately afterwards. This program's execution parallels that presented earlier [12].

Iterations are usually required to refine the PFILTER program's output estimates. The need for these refinements is indicated by the quality of the residuals as discussed earlier, and the degree of agreement of the filter's estimates with other events observed flight data, i.e. SRB separation time.

```
//CRMR197 JOB (6EP55       ),ROGERS,CLASS=G,MSGLEVEL=(1,1),
//    TIME=0025
//   EXEC CRAY
//SYSIN  DD *
 CRSUBMIT F(INPUT) HOLD NOTIFY(CRMR197) SUB(CSS1)
//INPUT  DD *
JOB,JN=CRMR197,T=900.
ACCOUNT,AC=6EP551590294,US=CRMR197.
FETCH,DN=PFILTR,TEXT='DSN=CRMR197.PFILTER.FOR(PFILTR),DISP=SHR'.
COPYR,I=PFILTR,O=FILTER,NR=190.
FETCH,DN=BLKDAT,TEXT='DSN=CRMR197.PFILTER.FOR(BLKDAT),DISP=SHR'.
COPYR,I=BLKDAT,O=FILTER,NR=260.
FETCH,DN=INITIL,TEXT='DSN=CRMR197.PFILTER.FOR(INITIL),DISP=SHR'.
COPYR,I=INITIL,O=FILTER,NR=155.
FETCH,DN=PROPAF,TEXT='DSN=CRMR197.PFILTER.FOR(PROPAF),DISP=SHR'.
COPYR,I=PROPAF,O=FILTER,NR=106.
FETCH,DN=ERTHM,TEXT='DSN=CRMR197.PFILTER.FOR(ERTHM),DISP=SHR'.
COPYR,I=ERTHM,O=FILTER,NR=124.
FETCH,DN=XDVEC,TEXT='DSN=CRMR197.PFILTER.FOR(XDVEC),DISP=SHR'.
COPYR,I=XDVEC,O=FILTER,NR=323.
FETCH,DN=PDMTRX,TEXT='DSN=CRMR197.PFILTER.FOR(PDMTRX),DISP=SHR'.
COPYR,I=PDMTRX,O=FILTER,NR=668.
FETCH,DN=ATMOS,TEXT='DSN=CRMR197.PFILTER.FOR(ATMOS),DISP=SHR'.
COPYR,I=ATMOS,O=FILTER,NR=134.
FETCH,DN=NASSME,TEXT='DSN=CRMR197.PFILTER.FOR(NASSME),DISP=SHR'.
COPYR,I=NASSME,O=FILTER,NR=266.
FETCH,DN=RADAR,TEXT='DSN=CRMR197.PFILTER.FOR(RADAR),DISP=SHR'.
COPYR,I=RADAR,O=FILTER,NR=208.
FETCH,DN=NASSRB,TEXT='DSN=CRMR197.PFILTER.FOR(NASSRB),DISP=SHR'.
COPYR,I=NASSRB,O=FILTER,NR=269.
FETCH,DN=NAERO,TEXT='DSN=CRMR197.PFILTER.FOR(NAERO),DISP=SHR'.
COPYR,I=NAERO,O=FILTER,NR=497.
FETCH,DN=NPLUME,TEXT='DSN=CRMR197.PFILTER.FOR(NPLUME),DISP=SHR'.
COPYR,I=NPLUME,O=FILTER,NR=286.
FETCH,DN=NMASS,TEXT='DSN=CRMR197.PFILTER.FOR(NMASS),DISP=SHR'.
COPYR,I=NMASS,O=FILTER,NR=201.
FETCH,DN=AXMAT,TEXT='DSN=CRMR197.PFILTER.FOR(AXMAT),DISP=SHR'.
COPYR,I=AXMAT,O=FILTER,NR=424.
FETCH,DN=INTRP1,TEXT='DSN=CRMR197.PFILTER.FOR(INTRP1),DISP=SHR'.
COPYR,I=INTRP1,O=FILTER,NR=14.
FETCH,DN=KONTRL,TEXT='DSN=CRMR197.PFILTER.FOR(CONTRL),DISP=SHR'.
COPYR,I=KONTRL,O=FILTER,NR=123.
FETCH,DN=UPDATE,TEXT='DSN=CRMR197.PFILTER.FOR(UPDATE),DISP=SHR'.
COPYR,I=UPDATE,O=FILTER,NR=201.
FETCH,DN=NOISE,TEXT='DSN=CRMR197.PFILTER.FOR(NOISE),DISP=SHR'.
COPYR,I=NOISE,O=FILTER,NR=23.
FETCH,DN=MEASUR,TEXT='DSN=CRMR197.PFILTER.FOR(MEASUR),DISP=SHR'.
COPYR,I=MEASUR,O=FILTER,NR=592.
FETCH,DN=MLIB,TEXT='DSN=CRMR197.PFILTER.FOR(MLIB),DISP=SHR'.
COPYR,I=MLIB,O=FILTER,NR=296.
FETCH,DN=OUTPUT,TEXT='DSN=CRMR197.PFILTER.FOR(OUTPUT),DISP=SHR'.
COPYR,I=OUTPUT,O=FILTER,NR=71.
FETCH,DN=ZDVEC,TEXT='DSN=CRMR197.PFILTER.FOR(ZDVEC),DISP=SHR'.
COPYR,I=ZDVEC,O=FILTER,NR=28.
FETCH,DN=LINFB,TEXT='DSN=CRMR197.PFILTER.FOR(LINFB),DISP=SHR'.
COPYR,I=LINFB,O=FILTER,NR=43.
FETCH,DN=REFRAC,TEXT='DSN=CRMR197.PFILTER.FOR(REFRAC),DISP=SHR'.
COPYR,I=REFRAC,O=FILTER,NR=98.
REWIND,DN=FILTER.
CFT,I=FILTER.
FETCH,DN=CONTRL,TEXT='DSN=CRMR197.CTLIPT.DATA,DISP=SHR'.
FETCH,DN=REALME,TEXT='DSN=CRMR197.REALMEA.DATA,DISP=SHR'.
FETCH,DN=METDAT,TEXT='DSN=CRMR197.METIPT.DATA,DISP=SHR'.
LDR.
DISPOSE,DN=FILOUT,DC=ST,^
       TEXT='DSN=CRMR197.FILOUT.DATA,DISP=(NEW,CATLG),'^
            'UNIT=SYSDA,SPACE=(CYL,(20,10),RLSE),'^
            'DCB=(RECFM=FB,LRECL=80,BLKSIZE=6320)'.
DISPOSE,DN=SSMEOT,DC=ST,^
       TEXT='DSN=CRMR197.SSMEOT.DATA,DISP=(NEW,CATLG),'^
            'UNIT=SYSDA,SPACE=(TRK,(5,1)),'^
            'DCB=(RECFM=FB,LRECL=133,BLKSIZE=6251)'.
DISPOSE,DN=SRBOUT,DC=ST,^
       TEXT='DSN=CRMR197.SRBOUT.DATA,DISP=(NEW,CATLG),'^
            'UNIT=SYSDA,SPACE=(TRK,(5,1)),'^
            'DCB=(RECFM=FB,LRECL=133,BLKSIZE=6251)'.
```

Figure 5.-1: PFILTER.CJOB File

An example of the need for refinements is when the SRB separation time as estimated does not agree with the event time observed. The filter set the separation time based on the estimate of head pressure. Using the same criterion as onboard, time when the second motor's pressure falls below 50 psi plus a predetermined delay time, the program stops execution at this time and prints out the filter's time, state variables, and error variances of those estimates.

Two quantities are adjusted to better match this observed event. These are the burn rate coefficient and pressure bias uncertainty. The first is the most dominant during first stage, and, even though the program adjusts this parameter automatically, an adjustment is usually required which may be larger than that allowed by the program. Keeping the program's range of adjustment small for this parameter prevents large departures from the assumed linear deviations upon which the extended Kalman filtering is based. By monitoring the outputs, these adjustments bring the filter's operation more closely into the linear operating region about the prescribed reference models.

After the SRB burn time, as estimated by the filter, agrees with the event time, other adjustments may be required. Another of these is the balance between the accelerometer and radar measurements. The accelerometer measurements significantly effects all states. By requiring too close agreement with the measurements, small measurement uncertainty specified, the resulting vehicle's position may deviate from that required by the radar measurements. As a consequence, the use of radar measurements will not totally correct for accelerometer measurement update's inappropriate corrections. To achieve the desired quality of zero mean residuals, both accelerometer and radar, the measurements for these measurements may have to be adjusted until these qualities are achieved.

Other adjustments may include the uncertainties associated with the wind and plume forces. The need for these adjustments may be seen when evaluating the radar azimuth residuals when the range and elevation residuals are in good agreement. The azimuth residuals may indicate that the vehicle's estimated trajectory is drifting from an otherwise good flight path. In this case, "good" means that the vehicle's altitude and speed are in good agreement as evidenced by the radar elevation and range residuals.

The winds effect the lateral trajectory characteristics significantly during the first stage. The reference profile generated by the program, using the tabulated wind data, may not be representative of that actually encountered by the vehicle in flight and may need changes estimated by the program, as reflected by the uncertainty levels specified, to more closely match the lateral trajectory. Adjusting wind and lateral plume force component uncertainty levels can indicate the need and significance of such an adjustment.

Another significant parameter for first stage is the quaternion uncertainty levels. This parameter principally effects the degree of adjustment allowed by the filter in changing the vehicle's attitude to match accelerometer component measurements. Too large an uncertainty will produce large oscillatory deviations between the estimated trajectory and the indicated by the radar residuals. Too small an uncertainty will require the estimated attitude to more closely agree with the results from integrating the rate gyro outputs. This smaller uncertainty can produce large trajectory deviations observed in the radar residuals.

The adjustments above are based on experiences with processing the first stage flight. The second stage processing produces another set of possible adjustments. The most important quantity for second stage processing is the initialization for the vehicle position and velocity vector components. This initial state vector is obtained from the BET program described in Appendix C. The vector components are inserted into the BLKDAT routine's data statements.

Additional adjustments for the second stage follows those for the first stage except, of course, for those associated with the SRB and winds. Corrections afforded by the SSME elements are weaker than anticipated, thus placing additional emphasis on the initializations discussed above.

Second stage operation uses a larger integration step size than does first stage; however, as a result of its much longer duration, the processing time is longer.

Performance results, Isp, for two flights are illustrated; STS-61C and STS-26. Results for other flights were presented in Reference [12]. Shown in Figures 5.-2 and 5.-3 are the SRB Isp's for STS-61C and STS-26 respectively. Shown in Figures 5.-4 and 5.-5 are the SSME Isp's for STS-61C stage 1 and 2 respectively. These results are consistent with those obtained by using other NASA reconstruction techniques.

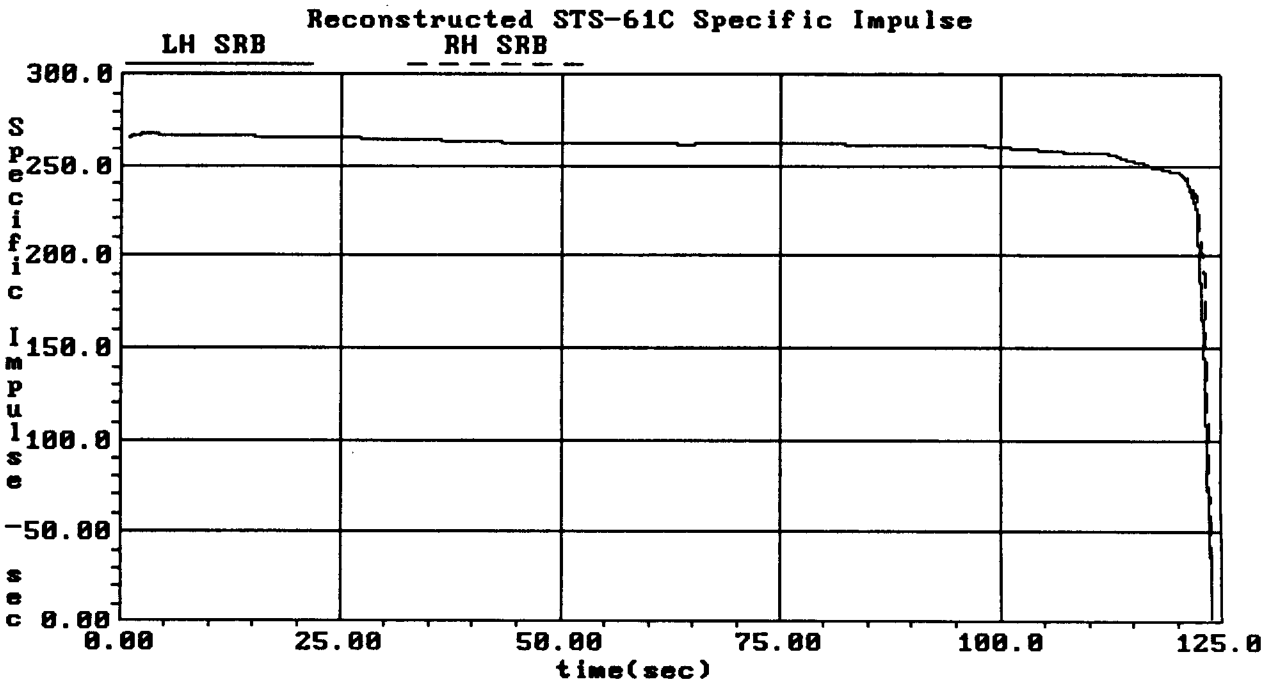


Figure 5.-2: STS-61C SRB Isp's

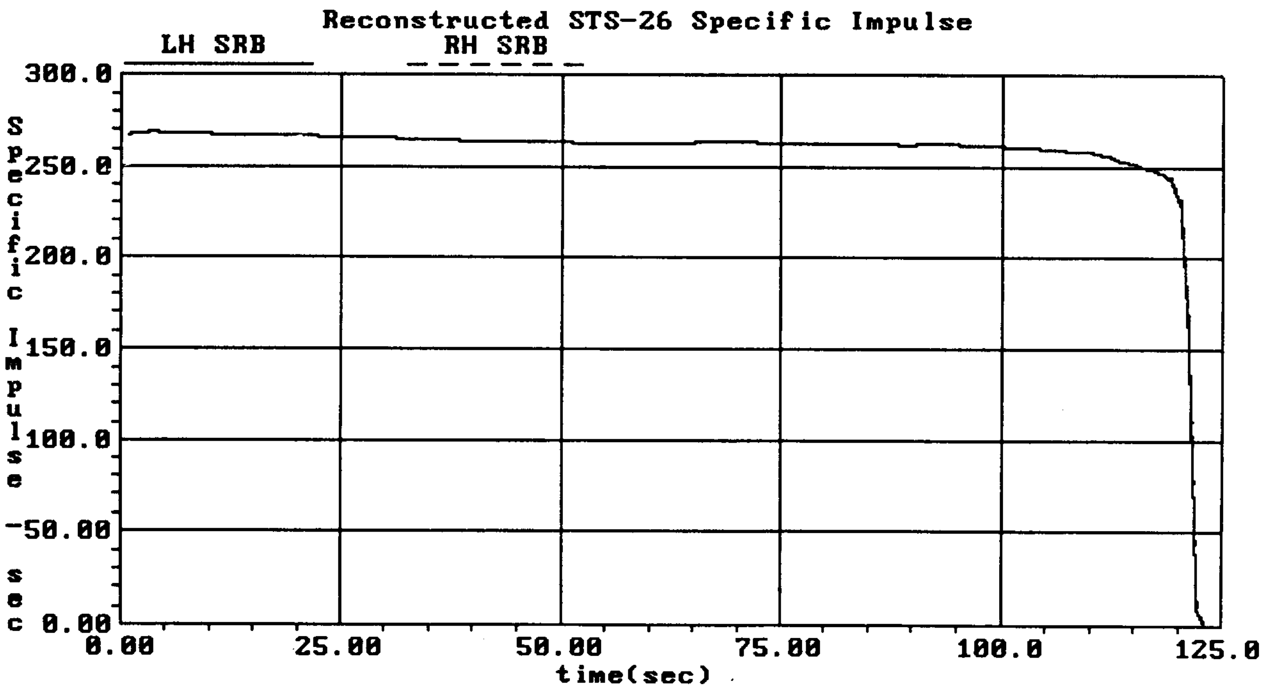


Figure 5.-3: STS-26 SRB Isp's

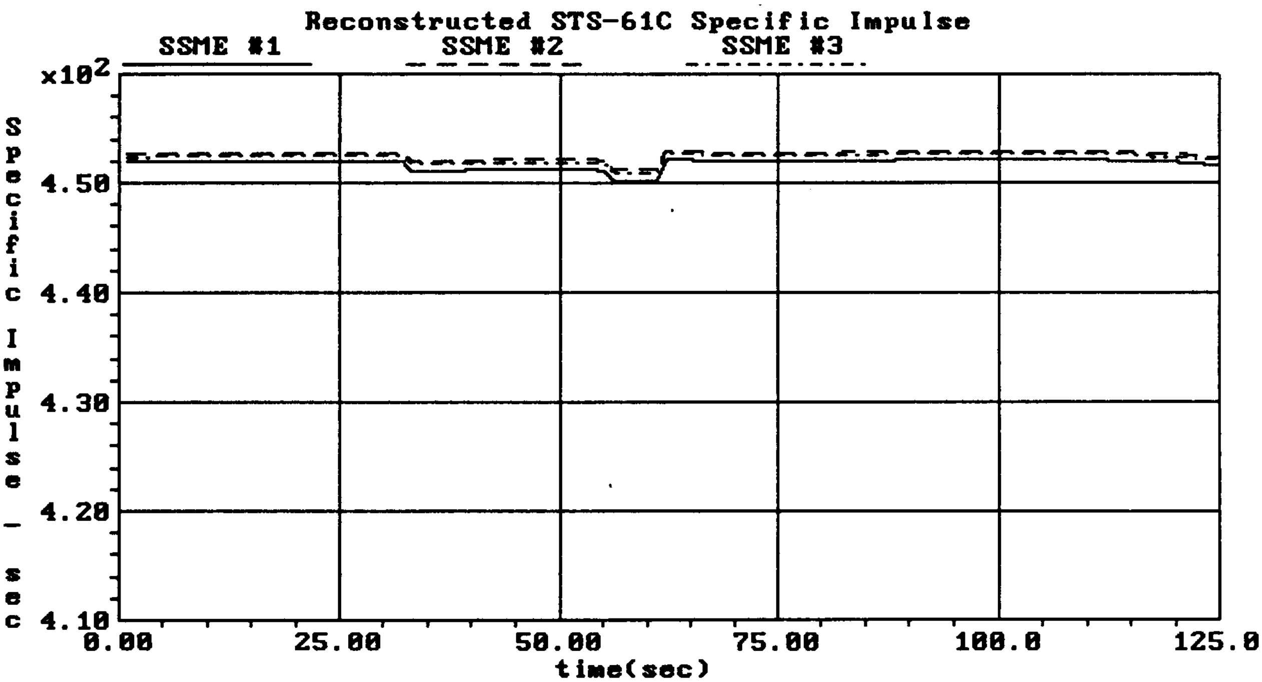


Figure 5.-4: STS-61C Stage I SSME Isp's

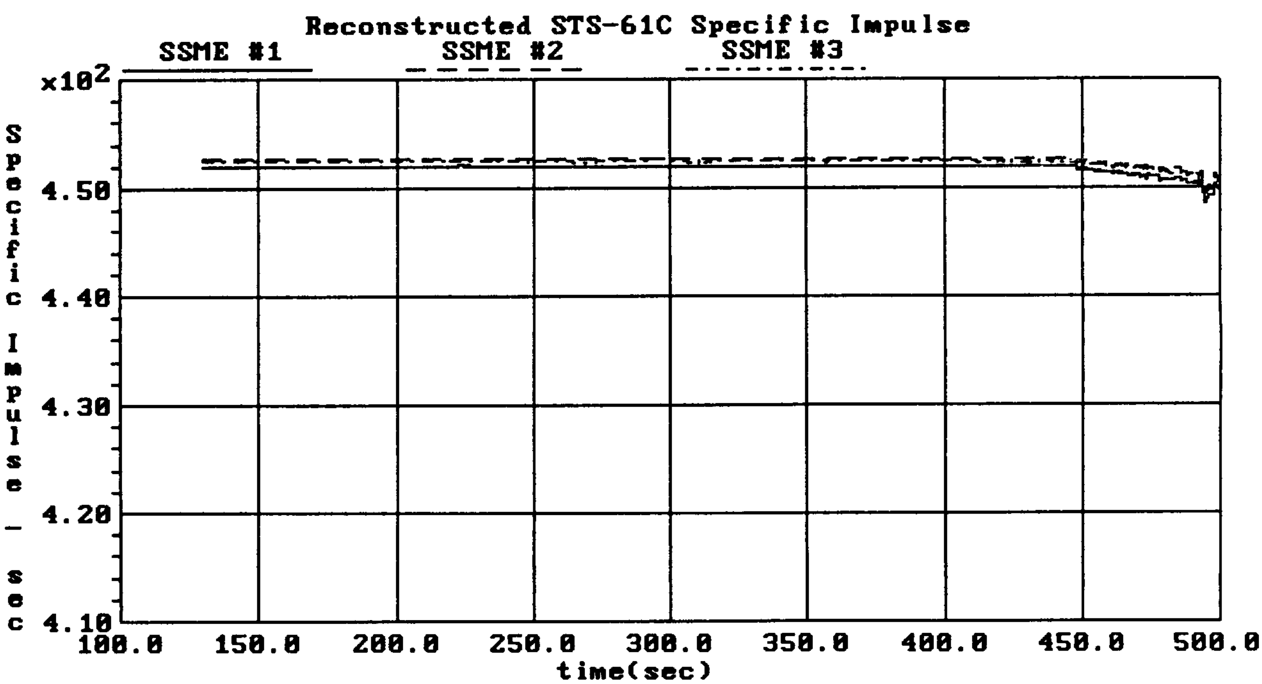


Figure 5.-5: STS-61C Stage II SSME Isp's

6. Conclusions and Recommendations

The computer programs developed during this contract can contribute to the flight test data analysis efforts. The first contribution is from the ability to generate a best estimate of trajectory (BET), and have this data shortly after a flight. This BET data should be available much sooner than was previously the case, and flight analysis efforts can begin with more time available prior to meeting scheduled completion dates.

The second contribution is from an alternative approach to analyzing the flight data. This alternative approach is through an integrated propulsion/trajectory reconstruction tool. This tool incorporates modeling from the propulsion disciplines, SRB and SSME, and incorporates the usual modeling from the trajectory disciplines, i.e. flight mechanics, aerodynamics, etc.

The new approaches included in this reconstruction tool include the internal ballistics modeling associated with the SRB. By including this model within the frame work of Kalman filtering, it is possible to resolve biases by processing redundant sources influenced by the internal SRB processes. Previous approaches relied on single measurements, i.e. head pressure, and engineering judgments were used to eliminate the errors associated with the pressure measurement sensor's biases. This approach allows the bias associated with the head pressure sensors to be resolved by processing accelerometer and head pressure measurements, both of which are influenced by the same internal SRB processes.

This new reconstruction tool does not contain the same modeling fidelity as other models that have been previously used for flight analysis. The modeling, by necessity, has been simplified in order to have an efficient and useful tool. These other approaches have merit in their more detailed models, and, as a result of using a different approach to the reconstruction, offer an independent evaluation of the flight data. This tool also has merit for the same reason, it offers an independent evaluation that uses an alternative methodology to arrive at the results.

It is recommended that this reconstruction tool be used as an additional analysis tool to complement the other analysis efforts. As a complementary tool, insights can be gained from its operation that can support judgments used in the previously used methodologies.

A. Preprocessing Programs

As seen in the discussions of the PFILTER program, several data files are required as input into this program. These data files are a logical separation of data types into measurements, deterministic control inputs, and environmental conditions. These data are further segmented into functional areas of main engine, solid rocket booster, inertial measurement unit, etc. Each of these data should be reviewed, using the data plots provided by the preprocessing programs, prior to their use in the PFILTER program to assess the quality and consistency of this data.

The preprocessing programs, to be discussed in subsequent sections, perform several functions. VEHREF edits inertial measurement accelerometer and attitude data, and transforms this data into the boost reference frame used by PFILTER. PREPRCS converts main engine gas pressurant flows into mass, or weight, flow rates and extracts the remainder of the propulsion measurements, i.e. SRB head pressure. REDRDR extracts the radar tracking data from the radar data file for those radars selected for use in PFILTER and LFILTER. The results of these three programs are merged as the basic input measurements by the program MERGE to form an input file for PFILTER.

The program CTRLTST extracts the deterministic control inputs, i.e. gimbal angles, for PFILTER.

The final preprocessing program is METTST. This program extracts the meteorological data that is supplied to the PFILTER and LFILTER programs. The primary purpose of this program is to convert the data into English system of units.

Each of these preprocessing programs, except for REDRDR and MERGE, use the ACCESS routine to extract the identified data from the "STS" formatted data bases located on the EADS.

A.1 MERGE Preprocessing Program

The program MERGE is simply a program to read data from three input files and merge them into a single output file. The three input files are:

Data File Unit Number	Data File Name
1	REFIPT.DATA
2	PROIPT.DATA
3	TRACK.DAT

The output file is:

Data File Unit Number	Data File Name
4	REALMEA.DATA

The input files are produced by VEHREF, PREPRCS and REDRDR respectively.

The following describes the program control variables and code symbol definitions:

Symbol	Units	Description
IMAX	-	maximum number of time points to be read
IBEGIN	-	time point to begin storing data for output
IEND	-	last time point for stored data
TIME	sec	current time for all data files
DIMU	*	IMU data; accelerometer, attitude and rates
DPROL	**	main engine data record; chamber pressure, pressurant flow rates, fuel volumetric flow rate, fuel temperature, and fuel pressure
DPROS	lb/in^2	SRB head pressures
DRDR	***	radar track azimuth, elevation and range

* units are; ft/sec^2, deg, and deg/sec respectively
** units are; lb/in^2, lb/sec, gal/min, deg R, and lb/in^2 respectively
*** units are; deg, deg, and ft respectively

A.2 VEHREF Preprocessing Program

VEHREF extracts IMU data for processing. The propulsion estimation and BET programs use this data. Prior to use in the PFILTER program, this data is merged with other measurement data in the MERGE preprocessing program.

VEHREF reads an input file and produces two output files. The input file is

Data File Unit Number	Data File Name
1	MSDREF.DAT

This file contains the MSID's for the data to be extracted from the STS data base. These MSID's are listed below

MSID	Units	Description
V95H0050C	rad	IMU 1 comp pitch resolver angle
V95H0051C	rad	IMU 1 comp azimuth resolver angle
V95H0053C	rad	IMU 1 comp inner roll resolver angle
V95H0054C	rad	IMU 1 comp outer roll resolver angle
V95L0065C	ft/sec	IMU 1 accum sensed change X vel
V95L0066C	ft/sec	IMU 1 accum sensed change Y vel
V95L0067C	ft/sec	IMU 1 accum sensed change Z vel
V95H1050C	rad	IMU 2 comp pitch resolver angle
V95H1051C	rad	IMU 2 comp azimuth resolver angle
V95H1053C	rad	IMU 2 comp inner roll resolver angle
V95H1054C	rad	IMU 2 comp outer roll resolver angle
V95L1065C	ft/sec	IMU 2 accum sensed change X vel
V95L1066C	ft/sec	IMU 2 accum sensed change Y vel
V95L1067C	ft/sec	IMU 2 accum sensed change Z vel
V95H2050C	rad	IMU 3 comp pitch resolver angle
V95H2051C	rad	IMU 3 comp azimuth resolver angle
V95H2053C	rad	IMU 3 comp inner roll resolver angle
V95H2054C	rad	IMU 3 comp outer roll resolver angle
V95L2065C	ft/sec	IMU 3 accum sensed change X vel
V95L2066C	ft/sec	IMU 3 accum sensed change Y vel
V95L2067C	ft/sec	IMU 3 accum sensed change Z vel

V95H3526C	deg/sec	pitch rate, flight body-inertial
V95H3527C	deg/sec	yaw rate, flight body-inertial
V95H3528C	deg/sec	roll rate, flight body-inertial
V95U0163C	g's	total load factor

The output files are

Data File Unit Number	Data File Name
3	REFPLT.DATA
9	REFIPT.DATA

The first of these files is a plot file containing line printer plots of the resulting output data in REFIPT.DAT.

Program flow control is accomplished using the following variables defined as:

Symbol	Units	Description
IST	-	number of time points between first time and corresponding first measurement time after lift-off
TSTART	sec	first time point and also time point for "snap" initialization prior to main engine ignition
TSTOP	sec	stopping time point for data outputs
SRATE	sec^{-1}	sample rate for measurement outputs

The program utilizes the resolver angles for each IMU and the cluster to Mean-50 transformation matrix to compute a transformation matrix from the body to boost reference frame. The approach is to take a "snap shot" of the resolver angles prior to main engine ignition, and, assuming the body orientation in the boost reference is exactly a 90° pitch angle, compute a Mean-50 (M50) to boost reference (BR) transformation matrix. This approach is approximate due to the earth rotation from the time of the snap shot until lift-off; however, the computed attitudes are in error by less than 0.1°.

The equation to compute these transformations are given as:

$$^{BR}C^{M50} = {}^{BR}T_{t=0}{}^{B}[-10.6]_P[-ROLO]_R[-PITCH]_P[-ROLI]_R[-YAW]_Y \qquad \text{A-1}$$

$$^{BR}C^{B} = {}^{BR}C^{M50}\ ({}^{B}C^{M50})^T \qquad \text{A-2}$$

Math	Symbol	Units	Description
$^{BR}T_{t=0}{}^{B}$	TBBRIO	-	transformation matrix from body to BR at t=0
-	REFSMT	-	transformation matrix from cluster to M50
-	PITCH	rad	pitch resolver angle
-	YAW	rad	yaw resolver angle
-	ROLI	rad	inner roll resolver angle
-	ROLO	rad	outer roll resolver angle
$^{B}C^{M50}$	TMP4/5	-	M50 to body transformation matrix
$^{BR}C^{M50}$	CM50BR	-	M50 to BR transformation matrix
$^{BR}C^{B}$	CBI	-	body to BR transformation matrix

From the CBI matrix, body attitude angles of roll, pitch and yaw are computed for each IMU. The CM50BR and REFSMT matrices are used to transform the platform sensed velocity changes into BR accelerations.

Before this program can be run, the REFSMT matrix data must be extracted from the data base. There are three of these matrices each containing 9 elements. This data is obtained using the STSDB program to tabulate the following MSID's

MSID	Units	Description
V99U3450C	-	
.	-	REFSMT's for IMU's 1, 2 & 3 (27 total)
V99U3476C	-	

These data are incorporated into the VEHREF program as FORTRAN data statements.

The accelerations are computed by subtracting the current accumulated sensed velocity from its previous time step value. For the three components of velocity, this is given as:

$$\Delta\underline{V} = \underline{V}_s - \underline{V}_{OLD} \qquad \text{A-3}$$

$$\underline{V}_{OLD} = \underline{V}_s \text{ for the next cycle} \qquad \text{A-4}$$

Math	Symbol	Units	Description
$\Delta\underline{V}$	DV	ft/sec	velocity difference (deltas)
$\underline{V}_s$	VAL	ft/sec	current value of accumulated sensed velocity
$\underline{V}_{OLD}$	VOLD	ft/sec	previous value of sensed velocity

The initial value of VOLD is set to zero.

The resulting accelerations, $\Delta\underline{V}$'s, from the three IMU's are averaged to yield a mean acceleration vector. A test is made to determine the quality of the data at the current time point. This test consists of comparing the components of the $\Delta\underline{V}$'s of two IMU's. If any of these components differ by greater than 0.15 ft/sec, then the previous time's computed mean value for that component is used for the current time point. A new mean vector is computed only when this test fails, indicating good quality data. The mean vector, either retained from a previous time point or from the current time point, is output. This editing is necessary as a result of extremely large and error sporadic spikes in the data.

The body attitude euler angles, roll, pitch and yaw, are computed from the body to boost reference inertial direction cosine matrix as:

$$\phi = \tan^{-1}({}^{BR}C_{3,2}{}^{B} \,/\, {}^{BR}C_{3,3}{}^{B}) \qquad \text{A-5}$$

$$\theta = \sin^{-1}(-{}^{BR}C_{3,1}{}^{B}) \qquad \text{A-6}$$

$$\psi = \tan^{-1}({}^{BR}C_{3,1}{}^{B} \,/\, {}^{BR}C_{1,1}{}^{B}) \qquad \text{A-7}$$

Mean values of these attitudes are also computed and tested as described for the acceleration components above.

The resulting outputs from VEHREF are described below.

Symbol	Units	Description
DVM	ft/sec	current or last "good" mean acceleration vector
THTM	deg	current or last "good" mean attitude components
OMEG	deg/sec	body-inertial rates.

A.3 PREPRCS Preprocessing Program

The PREPRCS program converts gas pressurant volumetric flows into mass, or weight, flow rates. Also, the program extracts other propulsion measurements used by PFILTER. As with the preceding program, there is a single input file:

Data File Unit Number	Data File Name
1	MSDMEA.DAT

This file contains the MSID's for the data to be extracted from the STS formatted data base files. These MSID's are listed below:

MSID	Units	Description
V41X1596E	event	MPS-GO2 press sov 1 close pwr on
V41X1598E	event	MPS-GO2 press sov 2 close pwr on
V41X1603E	event	MPS-GO2 press sov 3 close pwr on
V41X1661E	event	MPS-GH2 press sov 1 close pwr on
V41X1662E	event	MPS-GH2 press sov 2 close pwr on
V41X1663E	event	MPS-GH2 press sov 3 close pwr on
E41P1068D	psia	ME-1 ox tk pressurant press
E41P2068D	psia	ME-2 ox tk pressurant press
E41P3068D	psia	ME-3 ox tk pressurant press
V41P1160A	psia	MPS-ENG no 1 GH2 outlet temp
V41P1260A	psia	MPS-ENG no 2 GH2 outlet temp
V41P1360A	psia	MPS-ENG no 3 GH2 outlet temp
V41T1171A	deg F	MPS-ENG no 1 GOX press outlet temp
V41T1271A	deg F	MPS-ENG no 2 GOX press outlet temp
V41T1371A	deg F	MPS-ENG no 3 GOX press outlet temp
V41T1161A	deg F	MPS-ENG no 1 GH2 press outlet temp
V41T1261A	deg F	MPS-ENG no 2 GH2 press outlet temp
V41T1361A	deg F	MPS-ENG no 3 GH2 press outlet temp
E41P1016D	psia	ME-1 MCC pressure ch A
E41P2016D	psia	ME-2 MCC pressure ch A
E41P3016D	psia	ME-3 MCC pressure ch A
E41R1021D	gal/min	ME-1 fuel flowrate (avg)
E41R2021D	gal/min	ME-2 fuel flowrate (avg)

E41R3021D	gal/min	ME-3 fuel flowrate (avg)
E41P1018D	psia	ME-1 LPFT disch press (avg)
E41P2018D	psia	ME-2 LPFT disch press (avg)
E41P3018D	psia	ME-3 LPFT disch press (avg)
E41T1019D	deg R	ME-1 LPFT disch temp (avg)
E41T2019D	deg R	ME-2 LPFT disch temp (avg)
E41T3019D	deg R	ME-3 LPFT disch temp (avg)
B47P1300C	psia	LH press A SRM chamber
B47P2300C	psia	RH press A SRM chamber

The output files are:

Data File Unit Number	Data File Name
3	PROPLT.DATA
9	PROIPT.DATA

The first of these is a plot file containing line printer plots of the resulting output data PROIPT.

The conversion from volume to weight flow rates for the gas pressurants corresponds to the process used by the CONVRT quick look processing [8]. Standard routines are used to compute the density from temperature and pressure. In PREPRCS, these routines are supported by data from a block data routine. The routines used to perform this conversion are described as:

Routine	Description
BLKDATA	contains gas property and other constants
FINDD	computes density
PRESS	computes pressure from density and temperature
VPN	function to evaluate polynomial
DPDD	computes change in pressure with density

Program flow control is accomplished using the following variables:

Symbol	Units	Description
TSTART	sec	first measurement time
TSTOP	sec	stopping point for data output
SRATE	sec^{-1}	sample rate for measurement outputs
TSTAGE	sec	time to force to zero SRB head pressures

The following variables and descriptions correspond to those used in the CONVRT quick look program.

Symbol	Units	Description
FCV	-	flow control valve open (1) or closed (0)
PXM	psi	fuel and oxidizer pressurant pressures
TXM	deg F	fuel and oxidizer pressurant temperatures
WEHE	lb/sec	computed fuel and oxidizer pressurant flow rates

An iteration procedure is used to compute the WEHE variables above using the TOL parameter contained in BLKDATA. When the previously computed value of WEHE differs by less than or is equal to TOL, then the iteration is terminated and the last value of WEHE is assumed correct.

In addition to the last variable above for output, the following describes the PREPRCS outputs

Symbol	Units	Description
PCM	psi	measured chamber pressure
Y8	gal/min	fuel volumetric flow rate
PH	psi	fuel pressure

TH	deg R	fuel temperature
VAL 31/32	psi	left and right hand SRB head pressure respectively

A.4 REDRDR Preprocessing Program

REDRDR extracts data for specified radars from a data file containing the composite tracking data from ETR/KSC. Typically, there are 6 to 9 radars' data included on the composite. This program extracts either 5 of these for the BET processing or 3 for stage I and stage II propulsion estimation programs based on the user's specification and modification to the data statements in the program.

REDRDR reads the following data file

Data File Unit Number	Data File Name
1	RADAR.DAT

This file is generated by a VAX/VMS copy direct from a data tape supplied by ETR/KSC. At each time point, azimuth, elevation and range radar data are read, and from these, a new data file of the selected radar's data is created:

Data File Unit Number	Data File Name
3	TRACK.DAT

No plots are generated by this preprocessing program.

The following are program flow control variables:

Symbol	Description
NOUT	number of radar sites to output(3 for FILTER/5 for LFILTER)
KOUT	corresponding sequential number for site selected i.e.1,3,.
TINC	time increment for outputs.

The following are the input and output variables used in the program:

Symbol	Units	Description
TIME	sec	current time associated with each of the input radars' measurement arrays
AZM	deg	array of input azimuth measurements
ELM	deg	array of input elevation measurements
RNGM	ft	array of input range measurements
AZM2	deg	array of selected output azimuth measurements
ELM2	deg	array of selected output elevation measurements
RNGM2	ft	array of selected output range measurements

A.5 CTRLTST Preprocessing Program

This program extracts the main engine power level command, main engine gimbal angles, and SRB actuator displacements from the STS formatted data base files. Additionally, CTRLTST computes the vehicle attitudes, based on the approach presented in section A.2, and extracts the body rates as sensed by rate gyros. The input file is:

Data File Unit Number	Data File Name
1	MSDCTL.DAT

This file contains the MSID's for the data to be extracted. These MSID's are:

MSID	Units	Description
B58H1150C	in	LH position TVC rock actuator
B58H1151C	in	LH position TVC tilt actuator
B58H2150C	in	RH position TVC rock actuator
B58H2151C	in	RH position TVC tilt actuator
V90U1948C	-	commanded SSME throttle setting
V58H1100A	deg	MPS ENG 1 p actr posn
V58H1150A	deg	MPS ENG 1 y actr posn
V58H1200A	deg	MPS ENG 2 p actr posn
V58H1250A	deg	MPS ENG 2 y actr posn
V58H1300A	deg	MPS ENG 3 p actr posn
V58H1350A	deg	MPS ENG 3 y actr posn
V95H0051C	rad	
.		see section A.2
V95H2054C	rad	
V95H3526C	deg/sec	pitch rate, flight body-inertial
V95H3527C	deg/sec	yaw rate, flight body-inertial
V95H3528C	deg/sec	roll rate, flight body-inertial

The output files generated are:

Data File Unit Number	Data File Name
3	CTLIPT.DATA
9	CTLPLT.DATA

The first file also contains the control input data and the second contains plots of the data extracted from the data base files.

Program flow control is accomplished using the following variables:

Symbol	Units	Description
TSTART	sec	first measurement time
TSTOP	sec	stopping point for data output
SRATE	sec^{-1}	sample rate for measurement outputs

The power level in converted from a count value into a factor where 100 counts corresponds to 1.0.

A.6 METTST Preprocessing Program

METTST extracts data for processing. The propulsion estimation and BET programs use this data.

METTST reads the following input data file:

Data File Unit Number	Data File Name
1	MSDMET.DAT

This file contains the MSID's for the data extracted from the STS formatted Meteorological data files. These MSID's are listed below

MSID	Units	Description
WS	m/sec	wind speed
WD	deg	wind direction from north
DENS		ambient atmospheric density
PRESS		ambient atmospheric pressure
TEMP	C°	ambient atmospheric temperature
UNCWS	m/sec	uncertainty in wind speed
UNCWD	deg	uncertainty in wind direction
UNCD		uncertainty in density
UNCPR		uncertainty in pressure
UNCT	C°	uncertainty in temperature

The following output data file, including plots, is generated.

Data File Unit Number	Data File Name
3	METIPT.DATA

Program flow control is accomplished using the following variables, in metric units, defined as

Symbol	Units	Description
TSTART	m	first measurement altitude
TSTOP	m	stopping altitude for data output
SRATE	m^{-1}	sample rate measurement output

The data specified by the MSID's input the input file are converted to the following variables in English units

Symbol	Units	Description
ALT	ft	altitude, independent variable for data
DENS	$lb/sec^2/ft^4$	ambient atmospheric density
PRES	lb/ft^2	ambient atmospheric pressure
VSOUND	ft/sec	speed of sound
VWX	ft/sec	northward component of wind velocity
VWY	ft/sec	eastward component of wind velocity
UDENS	$lb/sec^2/ft^4$	uncertainty in density
UPRES	lb/ft^2	uncertainty in pressure
UTEMP	R°	uncertainty in temperature
UVWX	ft/sec	uncertainty in northward wind velocity
UVWY	ft/sec	uncertainty in eastward wind velocity

VSOUND is computed from TEMP as:

$$V_{sound} = 49.02 \sqrt{T}. \qquad \text{A-8}$$

WS and WD are resolved into north/south and east/west components by the following:

$$V_{Wx} = -V_W \sin(\psi_W) \qquad \text{A-9}$$

$$V_{Wy} = -V_W \cos(\psi_W). \qquad \text{A-10}$$

Math	Symbol	Units	Description
V_{Wx}	VWX	ft/sec	north component of wind velocity
V_{Wy}	VWY	ft/sec	east component of wind velocity
ψ_W	WD	deg	wind direction from north
V_W	WS	ft/sec	wind speed

B. Plotting Program - PLTFIL

One of the outputs of the propulsion estimation, FILTER, and the BET programs, LFILTER, is a data file containing the Kalman filter state variable estimates and measurement residuals. This output file is plotted using this line printer plot program for a quick assessment of the quality of the results generated. The discussion in this section concerns the outputs of the propulsion estimation program; however, a corresponding program is set-up for the BET program LFILTER.

The filter state variable estimate outputs from the FILTER program are written to an output file FILOUT.DATA as a function of time. This file and the file containing the y-axis plot labels, YLABFIL.DAT, is read by PLTFIL as

Data File Unit Number	Data File Name
1	FILOUT.DATA
2	YLABFIL.DAT

The output file is

Data File Unit Number	Data File Name
3	PLTFIL.DATA

This file contains the plot images for later printing on the line printer. This file can be viewed on a terminal screen prior to routing to the line printer.

Program flow in controlled from the input file assuming that there is no change from the standard number of states and measurements processed.

Data written on the file includes the state variable estimates and the associated variances which are used to establish the upper and lower bounds. The values of the state estimates are plotted with the square-root of the associated variance added to and subtracted from the estimate to produce three curves on one plot for each state variable.

Each measurement residual is similarly plotted. The residual and plus and minus values of residual variance square-root are plotted.

Three subroutines are needed to generate these plots. These are PLOT4, ADDCHR, and INCADD. The first, PLOT4, serves to generate the plot scaling, the placement of the points onto the plot, and to generate the plot labeling by entry calls to PLOT2, PLOT3 and PLOT4 respectively. Subroutines ADDCHR and INCADD fill in blanks or other characters as needed for each output line within a plot.

The subroutines above are used in all plotting programs developed. These include the data preprocessing and outputs of the estimation programs.

C. Best Estimate of Trajectory Program

C.1 Introduction

During the course of development and verification of the propulsion estimation program, results from a trajectory reconstruction program, TRW's LRBET5 [11], were used for comparison. Also, in some trials, the estimates from this program were used as inputs, i.e. accelerations, These estimates provided better data than the data from the STS data base, prompting the desire to have a BET and to have it more quickly than currently available. This objective was also an alternative path considered as part of the original contract if the propulsion estimation program effort was not successful. A BET program was developed initially to produce time history estimates of the following variables required by the propulsion estimation program:

1) acceleration
2) attitude

and

3) attitude rates.

These variables are inertially sensed inputs required by the propulsion estimation program. These data can be obtained from the STS data base, however, the resulting data is of poor quality.

As indicated above, the BET program can used as a preprocessing program, providing better input data into the propulsion estimation program, or as a tool to analyze the contributory trajectory profile data elements, i.e. radar tracking data, for data quality and consistency. In the latter case, this tool can be used to evaluate radar tracking site selection for the propulsion estimation program. With the availability of this tool, additional variables were identified for output from this BET program. These additional outputs include other trajectory variables, i.e. flight path angle, and atmospheric dependent quantities, i.e. dynamic pressure.

The sources of data for this BET program are the same as those for the more data intensive propulsion estimation program. The data sources include the STS data base, the meteorological data base and composite radar tracking data tapes from the Eastern Test Range (ETR).

This BET program uses some of the same routines as the propulsion estimation program. However, in this case, the dynamics associated with the system modeled are described by a linear process. The measurements processed in the BET program are nonlinear; however, the linearized Kalman filter is suitable for this application.

The dynamical process is the evolution of the vehicle position as indicated by the Inertial Measurement Unit (IMU). This position is based on sensed accelerations from accelerometers mounted on this inertially referenced and stabilized platform. These sensed accelerations are assumed corrupted by biases and small misalignments. The radar track measurements are used to correct the motion evolution via the Kalman filter which processes these measurements. The filter provides compensating corrections to the sensed accelerations accounting for accelerometer biases, platform misalignment, and radar azimuth, elevation and range biases. The inclusion of these error sources and the enhanced UDU^T form of the Kalman filtering algorithm are significant departures from the TRW LRBET5 approach.

The BET program is actually two programs operated in a two step processing sequence. The first is the Kalman filtering program, LFILTER, using the UDU^T algorithm, and the second is a smoothing program, RTSSMO, based on the Rauch-Tung-Striebel smoothing algorithm [3]. The filtering program is operated iteratively until the proper initial filter and process noise parameters are achieved. The combination is operated iteratively, with the outputs of the smoothing program used to form new inputs into the filtering program, until the resulting estimates no longer change significantly.

The filtering/smoothing process above is uncoupled from the computations involving atmospheric dependent variables. That is, atmospheric dependent variables are computed based on estimated vehicle states, i.e. velocity; however, the estimated states are not dependent on the atmospheric variables.

The general form of the linearized Kalman filter algorithm and the Rauch-Tung-Striebel smoothing algorithm are summarized in Table C-1 and C-2 respectively.

Table C-1

Linearized Kalman Filter Algorithm
(linear dynamics and nonlinear measurements)
(continuous time - discrete measurements)

System Model	$\dot{\underline{x}}(t) = F(t)\underline{x}(t) + \underline{w}(t)$; $\underline{w}(t) = N[0,Q(t)]$
Measurements	$\underline{z}_k = \underline{h}_k[\underline{x}(t_k)] + \underline{v}_k$; $k = 1,2,\ldots$; $\underline{v}_k = N[0,R_k]$
Initial Conditions	$\underline{x}(0) = N[\underline{x}_o,P_o]$
Other Assumptions	$E[\underline{w}(t)\underline{v}_k^T] = 0$ for all t and k
Auxiliary Relations	$\Phi \sim I + F(t)\Delta t$, $Q \sim Q(t)\Delta t$
State Estimate Propagation	$\hat{\underline{x}}_k(+) = \Phi\ \hat{\underline{x}}_{k-1}(+)$
Error Covariance Propagation	$P_k = \Phi\ P_{k-1}\ \Phi^T + Q$
State Estimate Update	$\hat{\underline{x}}_k(+) = \hat{\underline{x}}_k(-) + K_k\ \{\ \underline{z}_k - \underline{h}_k[\hat{\underline{x}}_k(-)]\ \}$
Error Covariance Update	$P_k(+) = \{\ I - K_k H_k[\hat{\underline{x}}_k(-)]\ \}\ P_k(-)$
Gain Matrix	$K_k = P_k(-)H_k[\hat{\underline{x}}_k(-)]^T\{H_k[\hat{\underline{x}}_k(-)]P_k(-)H_k[\hat{\underline{x}}_k(-)]^T+R_k\}^{-1}$
Linearized Measurement Matrix	$H_k = \left.\dfrac{\partial\underline{h}[\underline{x}(t_k)]}{\partial\underline{x}(t_k)}\right\vert_{\underline{x}(t_k) = \hat{\underline{x}}(t_k)}.$

Table C-2

Rauch-Tung-Striebel Smoothing Algorithm
(discrete time at measurement times)

State Estimate Propagation	$\hat{\underline{x}}_{k\|N} = \hat{\underline{x}}_k(+) + A_k \{ \hat{\underline{x}}_{k+1\|N} - \hat{\underline{x}}_{k+1}(-) \}$
Error Covariance Propagation	$P_{k\|N} = P_k(+) + A_k \{ P_{k+1\|N} - P_{k+1}(-) \} A_k^T$
Auxiliary Relations	$A_k = P_k(+)\Phi_k^T P_{k+1}^{-1}$ $\hat{\underline{x}}_{N\|N} = \hat{\underline{x}}_N(+)$ and $P_{N\|N} = P_N(+)$ for $k = N-1$

C.2 Mathematical Equation Descriptions

C.2.1 System Equations

The following vector differential equation describes the IMU indicated vehicle position in an inertial frame corrupted with the error sources assumed:

$$\ddot{\underline{R}}^{(BR)} = (\underline{a}_g + \underline{a}_S) + \delta\underline{\theta} \times \underline{a}_S + \underline{a}_b + [\underline{a}_S]_{diag}\, \underline{a}_{SF} \qquad \text{C-1}$$

where

$\underline{R}^{(BR)}$	= resulting boost reference inertial position vector
$(\underline{a}_g + \underline{a}_S)$	= net acceleration by gravity and sensed acceleration
$\delta\underline{\theta} \times \underline{a}_S$	= platform misalignment induced acceleration error
$\underline{a}_b$	= accelerometer bias error
$[\underline{a}_S]_{diag}\, \underline{a}_{SF}$	= accelerometer scale factor induced error

This equation is integrated twice to obtain the IMU indicated vehicle position. This indicated position would be in error as a result of the errors assumed.

The platform misalignment state, $\delta\underline{\theta}$, accelerometer bias state, $\underline{a}_b$, and the accelerometer scale factor error, $\underline{a}_{SF}$, are adjoined to the IMU indicated position and velocity to form the system error state vector for the Kalman filtering/smoothing algorithms. In addition to these states, error states associated with the radar measurements are also included. For each radar, bias states for azimuth, elevation and range are included in the state vector representation. These state vector elements are summarized in Table C-3.

The inertial frame chosen for the system above is the boost reference inertial frame (Appendix E). This frame is also that used for the propulsion estimation program. Each of the vector elements in equation C-1 are referenced in this frame.

The IMU acceleration data, assumed represented by equation C-1, is extracted from the STS data base by the preprocessing program VEHREF. In addition to the acceleration data, the vehicle attitude and attitude rates are also extracted by this preprocessing program. These data are used in the BET program.

Table C-3

State Vector Elements Modeled in LFILTER

Element	Units	Description
1	ft	x-position in boost reference inertial (BR)
2	ft	y-position in boost reference inertial frame
3	ft	z-position in boost reference inertial frame
4	ft/sec	x-velocity in boost reference inertial (BR)
5	ft/sec	y-velocity in boost reference inertial frame
6	ft/sec	z-velocity in boost reference inertial frame
7	rad	platform tilt error about x in BR frame
8	rad	platform tilt error about y in BR frame
9	rad	platform tilt error about z in BR frame
10	ft/sec^2	x-component accelerometer bias
11	ft/sec^2	y-component accelerometer bias
12	ft/sec^2	z-component accelerometer bias
13	-	x-component accelerometer scale factor error
14	-	y-component accelerometer scale factor error
15	-	z-component accelerometer scale factor error
16	deg	radar 1 azimuth bias
17	deg	radar 1 elevation bias
18	ft	radar 1 range bias
19	deg	radar 2 azimuth bias
20	deg	radar 2 elevation bias
21	ft	radar 2 range bias
22	deg	radar 3 azimuth bias
23	deg	radar 3 elevation bias
24	ft	radar 3 range bias
25	deg	radar 4 azimuth bias
26	deg	radar 4 elevation bias
27	ft	radar 4 range bias
28	deg	radar 5 azimuth bias
29	deg	radar 5 elevation bias
30	ft	radar 5 range bias

C.2.2 Measurement Equations

The measurements processed in the Kalman filtering algorithm are the azimuth, elevation and range measurements from the C-band radars operated by the ETR. These radars are located at Kennedy Space Center (KSC), on Wallops Island, and on the outlying islands of Bermuda and Bahamas.

The azimuth, elevation and range measurements, defined in a local level topographic frame, are nonlinear functions of the boose reference vehicle position. This functional dependence is defined by the following equations.

The components of the vehicle position vector relative to the radar site location in the local level topographic frame (Appendix E) is given by;

$$\begin{matrix} x \\ y \\ z \end{matrix} = \Delta \underline{R}_V^{(LL)} = {}^{LL}C^{ECF} \left[{}^{ECF}C^{ECI} \, \underline{R}^{(ECI)} - \underline{R}_{RDR}^{(ECF)} \right] \qquad \text{C-2}$$

where

$$ {}^{LL}C^{ECF} = \begin{matrix} -\sin(lat) & \cos(lat) & 0 \\ -\sin(lat)\cos(lon) & -\sin(lat)\sin(lon) & \cos(lat) \\ \cos(lat)\cos(lon) & \cos(lat)\sin(lon) & \sin(lat) \end{matrix} \qquad \text{C-3}$$

lat = radar site geodetic latitude (Appendix E)

lon = radar site longitude

$$^{ECF}C^{ECI} = \begin{matrix} \cos(\omega t) & \sin(\omega t) & 0 \\ \sin(\omega t) & \cos(\omega t) & 0 \\ 0 & 0 & 1 \end{matrix} \qquad \text{C-4}$$

ω = earth rotation rate (Appendix E)

t = elapsed time from liftoff

and,

$$\underline{R}^{(ECI)} = {}^{ECI}C^{BR}\ \underline{R}^{(BR)}. \qquad \text{C-5}$$

The superscripts refer to local level (LL), earth centered fixed (ECF), earth centered inertial (ECI), and boost reference (BR).

The range, azimuth and elevation (Appendix E) from the vehicle to the radar site then is given by;

$$\rho = \sqrt{(\ x^2 + y^2 + z^2\ )} + \rho_b + \Delta\rho \qquad \text{C-6}$$

$$A = \tan^{-1}(\ x/y\ ) + A_b \qquad \text{C-7}$$

$$E = \tan^{-1}(\ z/\sqrt{(\ x^2 + y^2\ )}\ ) + E_b + \Delta E \qquad \text{C-8}$$

where

ρ = range from radar site to vehicle

ρ_b = radar range bias

$\Delta\rho$ = radar range correction due to refraction

A = azimuth from radar site to vehicle

A_b - radar azimuth bias

E - elevation from radar site to vehicle

E_b - radar elevation bias

ΔE - radar elevation correction due to refraction

The bias terms are included in the state vector representation summarized in Table C-3 for each radar.

These radars begin tracking the vehicle shortly after launch, ~12 seconds, and continue tracking through main engine cutoff, ~520 seconds. Two of the KSC radars usually track the solid rocket boosters at the end of stage I, ~125 seconds.

Generally, 6 to 9 radars are available and tracking with the tracking intermittent and not continuous during the boost phase of flight. The BET program is setup to use 5 of these radars and each measurement, i.e. range, processed individually. The radar data used is extracted from the file containing the composite radar tracking data by the preprocessing program REDRDR. The Kalman filtering program requires the radar site location for each radar chosen and this is available with the data supplied by ETR/KSC. The measurements processed are summarized in Table C-4.

Table C-4

Measurement Vector Elements Modeled in LFILTER

Element	Units	Description
1	deg	radar 1 azimuth
2	deg	radar 1 elevation
3	ft	radar 1 range
4	deg	radar 2 azimuth
5	deg	radar 2 elevation
6	ft	radar 2 range
7	deg	radar 3 azimuth
8	deg	radar 3 elevation
9	ft	radar 3 range
10	deg	radar 4 azimuth
11	deg	radar 4 elevation
12	ft	radar 4 range
13	deg	radar 5 azimuth
14	deg	radar 5 elevation
15	ft	radar 5 range

C.3 Auxiliary Outputs

Additional variables from the filtering/smoothing processing are provided. These variables are formed from the state variables computed in the estimation process. The variables and their 1-sigma bounds are provided.

These variables are output in a 90 word element array, for each time point, which is then accessed by other interested users of the data. The entire 90 word array is not filled in the outputs provided by this BET. The outputs from this BET and the corresponding outputs from the TRW LRBET5 are provided in the same locations in this array. Table C-5 summarizes the output variables provided by the BET program.

For the variables indicated in Table C-5 that are not self explanatory, the following are definitions of these variables.

C.3.1 Body Velocity and Accelerations

These vectors are produced by transforming the inertial (boost reference) vectors into the body frame (Appendix E) as

$$\underline{v}^{(B)} = {}^{B}C^{BR}\ \underline{v}^{(BR)} \tag{C-9}$$

$$\underline{a}^{(B)} = {}^{B}C^{BR}\ \underline{a}^{(BR)} \tag{C-10}$$

$$a_{load} = \sqrt{(\ a_1^2 + a_2^2 + a_3^2\ )}. \tag{C-11}$$

C.3.2 Inertial Velocity Magnitude and Flight Path Angles

These variables use the inertial velocity components in boost reference (see Appendix E) as

$$V_M = \sqrt{(v_1^2 + v_2^2 + v_3^2)} \qquad \text{C-12}$$

where the v's are components of $\underline{v}^{(BR)}$.

$$\gamma_M = \sin^{-1} (\underline{r}^{(BR)} * \underline{v}^{(BR)} / |\underline{r}^{(BR)}| \; |\underline{v}^{(BR)}|) \qquad \text{C-13}$$

and

$$\psi_M = \tan^{-1} (v_{east} / v_{north}) \qquad \text{C-14}$$

where

$$\begin{matrix} v_{east} \\ v_{north} \\ v_{up} \end{matrix} = {}^{LL}C^{EF} \; {}^{EF}C^{BR} \; \underline{v}^{(BR)}. \qquad \text{C-15}$$

C.3.3 Wind Relative Velocity and Angles

The wind, from the meteorological inputs as a function of altitude, is first transformed into body axis (Appendix E) and differenced to obtain the wind relative velocity vector as

$$\underline{v}_R^{(B)} = \underline{v}^{(B)} - {}^{B}C^{LL}\underline{v}_W. \qquad \text{C-16}$$

Then

$$V_W = \sqrt{(v_1^2 + v_2^2 + v_3^2)} \qquad \text{C-17}$$

where the v's are components of $\underline{v}_R^{(B)}$.

Angle of attack and side slip (see Appendix E) are then given as

$$\alpha = \tan^{-1} (v_3 / v_1) \qquad \text{C-18}$$

and

$$\beta = \sin^{-1} (v_2 / V_W). \qquad \text{C-19}$$

C.3.4 Mach number and Dynamic pressure

Mach number and dynamic pressure computations use the input meteorological data of ambient temperature, T, and density, ρ, from the of estimated altitude as

$$M = V_W / a \qquad \text{C-20}$$

and

$$q = 1/2 \, \rho \, V_W^2 \qquad \text{C-21}$$

where speed of sound, a, is computed as

$$a = 49.02 \sqrt{T}. \qquad \text{C-22}$$

Table C-5

BET Output Variables

Location	Variable	Description	Units
1	Time	Elapsed time from liftoff	sec
2	XM	Boost Reference x-position	ft
3	YM	Boost Reference y-position	ft
4	ZM	Boost Reference z-position	ft
5	XDM	Boost Reference x-velocity	ft/sec
6	YDM	Boost Reference y-velocity	ft/sec
7	ZDM	Boost Reference z-velocity	ft/sec
8	XDDM	Boost Reference x-acceleration	ft/sec^2
9	YDDM	Boost Reference y-acceleration	ft/sec^2
10	ZDDM	Boost Reference z-acceleration	ft/sec^2
20	XDB	Body x-velocity component	ft/sec
21	YDB	Body y-velocity component	ft/sec
22	ZDB	Body z-velocity component	ft/sec
23	XDDB	Body x-acceleration component	ft/sec^2
24	YDDB	Body y-acceleration component	ft/sec^2
25	ZDDB	Body z-acceleration component	ft/sec^2
45	R	Boost Reference yaw body rate	deg/sec
46	Q	Boost Reference pitch body rate	deg/sec
47	P	Boost Reference roll body rate	deg/sec
49	PSID	Vehicle latitude	deg
50	ALAMDA	Vehicle longitude	deg
51	H	Vehicle altitude	ft
55	VM	Boost Reference velocity magnitude	ft/sec
56	GAMMAM	Vertical flight path angle	deg
57	PSIM	Lateral flight path angle	deg
68	VTD	Wind relative velocity magnitude	ft/sec
72	ALPHA	Wind relative angle of attack	deg
74	QBAR	Dynamic pressure	psf
75	QBALPH	QBAR and ALPHA product	deg psf
76	QBBETA	QBAR and BETA product	deg psf
77	AM	Mach number	-
79	T	Ambient air temperature	R°
80	PR	Ambient air pressure	psf
81	RHOAMB	Ambient air density	lb sec^2/ft^4
83	ALOAD	Sensed acceleration magnitude	g's
84	BETA	Wind relative angle of side slip	deg
85	EULERY	Boost Reference yaw Euler angle	deg
86	EULERP	Boost Reference pitch Euler angle	deg
87	EULERR	Boost Reference roll Euler angle	deg

C.4 Program Descriptions

C.4.1 LFILTER Program

The program LFILTER organization is illustrated in Figure C-1. Each of the routines shown, their function and equations implemented, will be discussed in turn in the subsequent subsections. The correspondence of the mathematical notation and the symbology used in the FORTRAN code will also be presented.

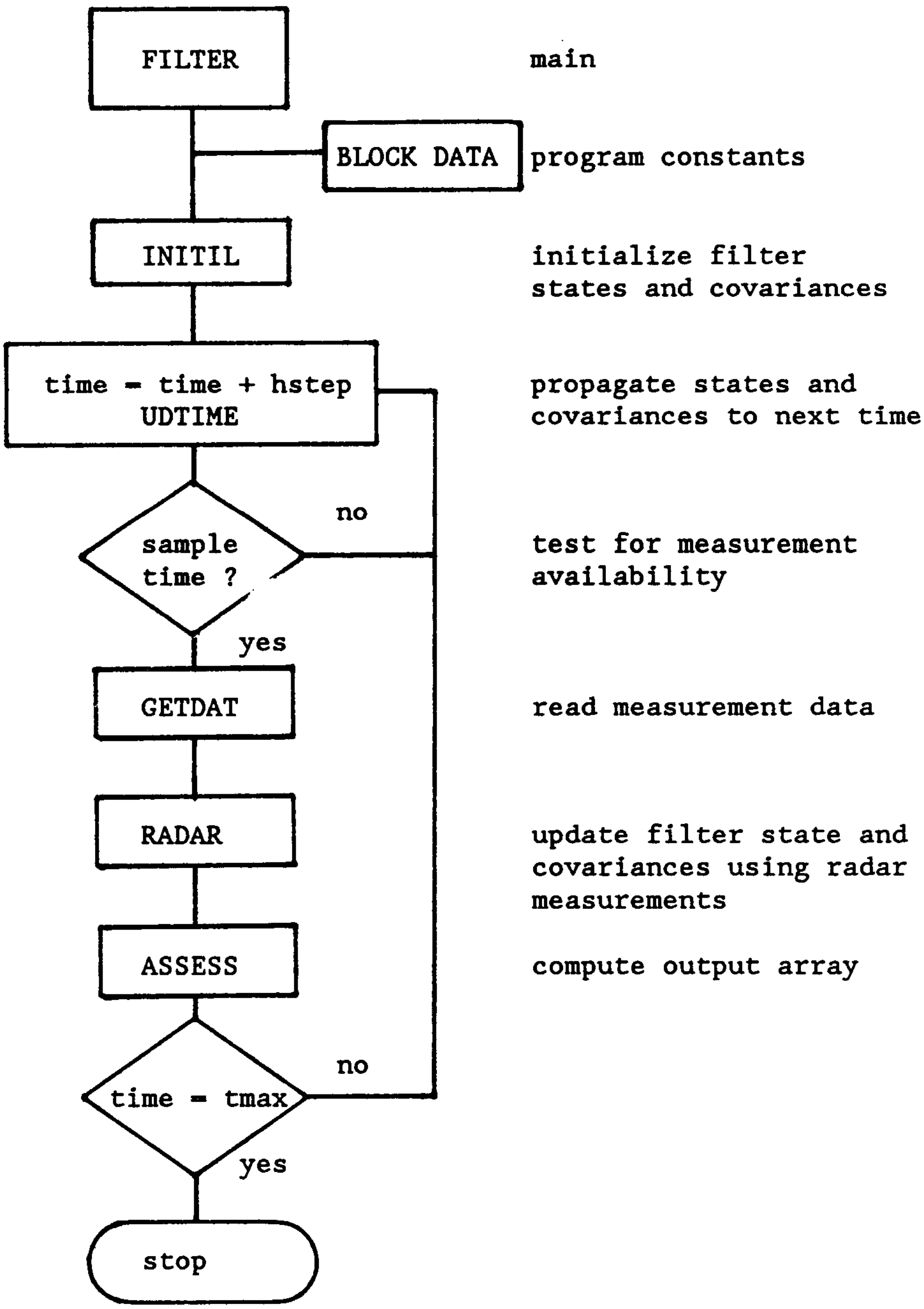


Figure C-1: LFILTER Program Flow

C.4.1.1 FILTER routine

The main routine, FILTER, controls the program flow. It opens and reads data files, initializes arrays based on data contained in the BLOCK DATA routine by calling the INITIL subroutine, integrates the state variables up to a measurement time with subroutine UDTIME, processes the measurements and updates the state variables in the subroutine RADAR, produces the output variables shown in Table C-3 in the ASSESS subroutine, and outputs filter quality data in subroutine OUTPUT.

The program flow in controlled by specifying a maximum or stop time. If the current filter time is less than the maximum time specified, the program continues the time integration and measurement processing. During the processing, data is stored for later plotting or processing by the smoothing program. Data outputs for the smoothing processing is activated by setting the ISMOOTH flag in BLOCK DATA to 1.

Data files produced by the preprocessing programs previously described are also used in this program. The input data files read are:

Data File Unit Number	Data File Name
2	TRACK.DAT
3	REFIPT.DAT
4	METDAT.DAT

TRACK.DAT contains the radar tracking measurements processed in the filter algorithm. REFIPT.DAT contains the reference IMU acceleration, attitude and attitude rate data. METDAT.DAT contains the meteorological data used for the atmospheric dependent variables computed.

Four files are produced by LFILTER. The output files written are:

Data File Unit Number	Data File Name
6	LFILOUT.DAT
7	SREFIPT.DAT
8	SMOIPT.DAT
9	LASSOUT.DAT

LFILOUT.DAT contains filter state variable estimates and measurement residuals for quality checks on the filter's outputs. SREFIPT.DAT contains the reference inputs required for the smoothing program. SMOIPT.DAT contains filter products for the smoothing processing cycle. LASSOUT.DAT contains the output variables summarized in Table C-3.

C.4.1.2 BLOCK DATA routine

The BLOCK DATA routine contains data to support the program flow control, i.e. TMAX for the maximum (stop) time, and modeling data that may be frequently changed to refine the filter's estimates and to evaluate aspects of the filter's output quality.

The following program control values are specified in BLOCK DATA.

Symbol	Units	Common	Description
TIME	sec	TIMDAT	current filter time after liftoff
TMAX	sec	TIMDAT	maximum operation time after liftoff
HSTEP	sec	TIMDAT	time integration step size
TSAMP	sec	TIMDAT	measurement sample time increment
N	-	LINFMT	number of dynamic and bias states
NP	-	NOISES	number of process noise(bias) states
NRMEAS	-	LINHMT	number of measurements processed

Earth model specific data and initial vehicle position data are as follows:

Symbol	Units	Common	Description
RE	ft	EDATA	earth model radius at the equator
FLAT	-	EDATA	earth model flattening factor
OMEGE	rad/sec	EDATA	earth angular rotation speed
XMU	ft^3/sec^2	EDATA	earth model gravity parameter
XJ2	-	EDATA	earth model oblate gravity parameter
OLATD	deg	LAUCOR	launch point coordinates
OLONG	deg	LAUCOR	launch point coordinates
OHT	ft	LAUCOR	launch point coordinates

Constants used for conversions are:

Symbol	Units	Common	Description
CRAD	deg/rad	CONST	conversion from radians to degrees
AGRAV	ft/sec^2	CONST	gravity acceleration constant
HRSEC	sec/hr	CONST	conversion from hours to seconds
PERCNT	-	CONST	conversion from factor to percent
XMRAD	-	CONST	conversion from radians to millirads

Radar site coordinates and atmospheric refraction correction data for each radar are specified by:

Symbol	Units	Common	Description
RLAT	deg	RDRDAT	radar site coordinates
RLONG	deg	RDRDAT	radar site coordinates
RHT	ft	RDRDAT	radar site coordinates
XNO	-	RDRDAT	atmospheric refraction index
NRDR	-	RDRDAT	number of radar sites used

Values for initializing the Kalman filter error covariance matrix and setting process noise parameters error magnitudes are;

Symbol	Units	Common	Description
R	**	NOISES	measurement noise array elements
ER	ft	APRIOR	initial position uncertainty
EV	ft/sec	APRIOR	initial velocity uncertainty
ETLT	deg	APRIOR	initial platform tilt uncertainty
EAB	ft/sec^2	APRIOR	initial accelerometer bias uncertain
EASF	-	APRIOR	initial accelerometer scale factor
ERDR	**	APRIOR	initial radar measurement uncertain
TAUT	sec	FPARAM	platform tilt process noise constant
TAUA	sec	FPARAM	accelerometer bias noise constant
TAUS	sec	FPARAM	accelerometer scale factor constant
TAUR	sec	FPARAM	radar measurement noise constant

UT	deg	FPARAM	platform tilt process noise level
UA	ft/sec^2	FPARAM	accelerometer bias noise level
US	-	FPARAM	accelerometer scale factor level
UR	**	FPARAM	radar measurement noise level

** deg for azimuth and elevation, and ft for range.

C.4.1.3 INITIL Routine

The INITIL routine initializes the states and error covariance matrix for the Kalman filter algorithm. This routine computes the initial BRI position vector from the launch coordinates in BLOCK DATA. Also, computed is the transformation matrix from earth centered inertial (ECI) to BRI. This matrix is used later in the RADAR measurement update routine.

The initial, time zero, computations are described as:

Symbol	Units	Common	Description
RBRI	ft	RSTATE	initial BRI position vector
VITO	ft/sec	RSTATE	initial BRI velocity vector
ABRI	ft/sec^2	RSTATE	initial BRI acceleration vector
CIBRI	-	RSTATE	transform matrix from ECI to BRI

The states, error covariance matrix, and U-D factored form of the error covariance matrix initialized in INITIL are as follows:

Symbol	Common	Description
XKM	PREUP	state vector prior to measurement update
PKM	PREUP	error covariance matrix prior to update
X	PROPAG	state vector between measurement updates
U	UDWORK	U-D factorization of PKM

The units for these arrays corresponds to those identified in Table C-1 for the state variables.

C.4.1.4 UDTIME Routine

This routine propagates (integrates or time updates) the system errors defined by equation C-1. This is accomplished by first forming the linearized system F matrix in the subroutine SYSTEM. The U-D time update subroutines XPHIU and WGS are used.

Referring to equation C-1, the following code symbols are defined.

Math	Symbol	Units	Subroutine	Description
$\underline{R}^{(BR)}$	RBRI	ft	UDTIME	BRI position vector
$\underline{a}_g + \underline{a}_g$	ANET	ft/sec^2	SYSTEM	net acceleration vector
$\underline{a}_b$	XKM	ft/sec^2	UDTIME	accelerometer bias error (elements 10-12 of XKM)
$\delta\underline{\theta}$ x $\underline{a}_S$	VEC2	ft/sec^2	UDTIME	platform tilt induced error
$[\underline{a}]_{diag}\ \underline{a}_{SF}$	VEC3	ft/sec^2	UDTIME	accelerometer scale factor induced error

The linearized dynamics matrix, F, and the process noise matrices are communicated from the subroutine SYSTEM via common arrays as:

Symbol	Common	Description
F	LINFMT	linearized dynamics matrix
Q	NOISES	process noise level matrix
G	NOISES	process noise distribution matrix

Units for these variables correspond to those identified in Table C-1.

C.4.1.5 RADAR routine

This routine computes the state (see Table C-1) estimate updates based on the linearized Kalman filter U-D factored algorithm by processing the radar measurements. In the linearized algorithm, the filter corrections are based on small departures from the current estimates of the measurements. Using the equations in section C.2, estimates of azimuth, elevation and range for each radar are computed using the current state estimates of vehicle position and the location of the radar site. These estimates of the measurements are then differenced from the actual measurements to form the residual. The residual is then multiplied by the gain to obtain the correction to the state estimates. The gain requires the computation of the linearized measurement matrix, H, which is effectively the gradient or direction to which the corrections are directed to the state elements.

Referring to equations C-2 through C-5, the following code symbols are defined:

Math	Symbol	Units	Subroutine	Description
$\underline{R}^{(BR)}$	RBRI	ft	UDTIME/RADAR	vehicle position in BRI
${}^{ECF}C^{BR}$	CBRIEF	-	RADAR	transformation from BRI to ECF
${}^{ECF}C^{ECI}\underline{R}^{(ECI)}$	REF	ft	RADAR	vehicle position in ECF
A	AZHAT	deg	RADAR	radar azimuth estimate
Δ_E	DEL	deg	REFRAC	elevation correction

$\Delta\rho$	DRANGE	ft	REFRAC	range refraction correction
E	ELHAT	deg	RADAR	radar elevation estimate
ρ	RNGHAT	ft	RADAR	radar range estimate

The following measurements are processed in the RADAR routine:

Symbol	Units	Subroutine	Description
AZM	deg	GETDAT	azimuth measurement
ELM	deg	GETDAT	elevation measurement
RNGM	ft	GETDAT	range measurement

To update the state estimates, the following are computed:

Symbol	Common	Description
H	LINHMT	linearized measurement matrix
RH	UDWORK	azimuth, elevation or range row of H
U	UDWORK	U-D factorization of error covariance matrix
PO	UDWORK	temporary storage for upper diagonal of error covariance matrix
SF	UDWORK	unused matrix in U-D routines
SG	UDWORK	unweighted Kalman gain vector
RESID	UDWORK	measurement and estimate difference
COVZ	UDWORK	residual variance

The units of these variables correspond to those indicated in Tables C-3 and C-4.

The routine REFRAC provides the corrections to elevation and range for atmospheric refraction. Routines UDMEAS and UD2COV accomplish the covariance matrix update and conversion from U-D to covariance matrix form for output respectively. The updates from each measurement are communicated to the OUTPUT routine.

Prior to the use of an update, or the completion of the measurement update, a test is made of the quality of the measurement. The measurement residual must be within a band of plus or minus 6 times the square-root of the measurement residual variance to be accepted. If this residual test fails and the state estimate update is bypassed.

C.4.1.6 ASSESS routine

This routine uses the resulting estimates from the Kalman filter algorithm to compute estimates of the variables summarized in Table C-5. These variables are formed from combinations of the state elements and meteorological data, tabular functions of altitude, using the filter estimate of altitude. The outputs from this routine are written to the output file, LASSOUT.DAT, for later processing by other NASA programs.

C.4.1.7 OUTPUT routine

This routine writes the output files for later processing. A quality assessment file, LFILOUT.DAT, is output. This file is later accessed by the plotting program PLTLFIL. Plotted are the filter estimates and measurement residuals from which the quality of the filter's estimates are determined.

If the smooth flag ISMOOTH is initialized, then the files SREFIPT.DAT and SMOIPT.DAT are output. These two files are required by the smoothing program RTSSMO discussed in the next section.

C.4.2 RTSSMO Program

The Rauch-Tung-Striebel smoothing algorithm reflects little about the system model or measurements processed. Its form and it's implementation discussed here is general and applicable to any linear or linearized problem definition.

The program RTSSMO organization is illustrated in Figure C-2. Each of the routines shown, their function and equations implemented, will be discussed in turn in the subsequent subsections. The correspondence of the mathematical notation and the symbology used in the FORTRAN code will also be presented.

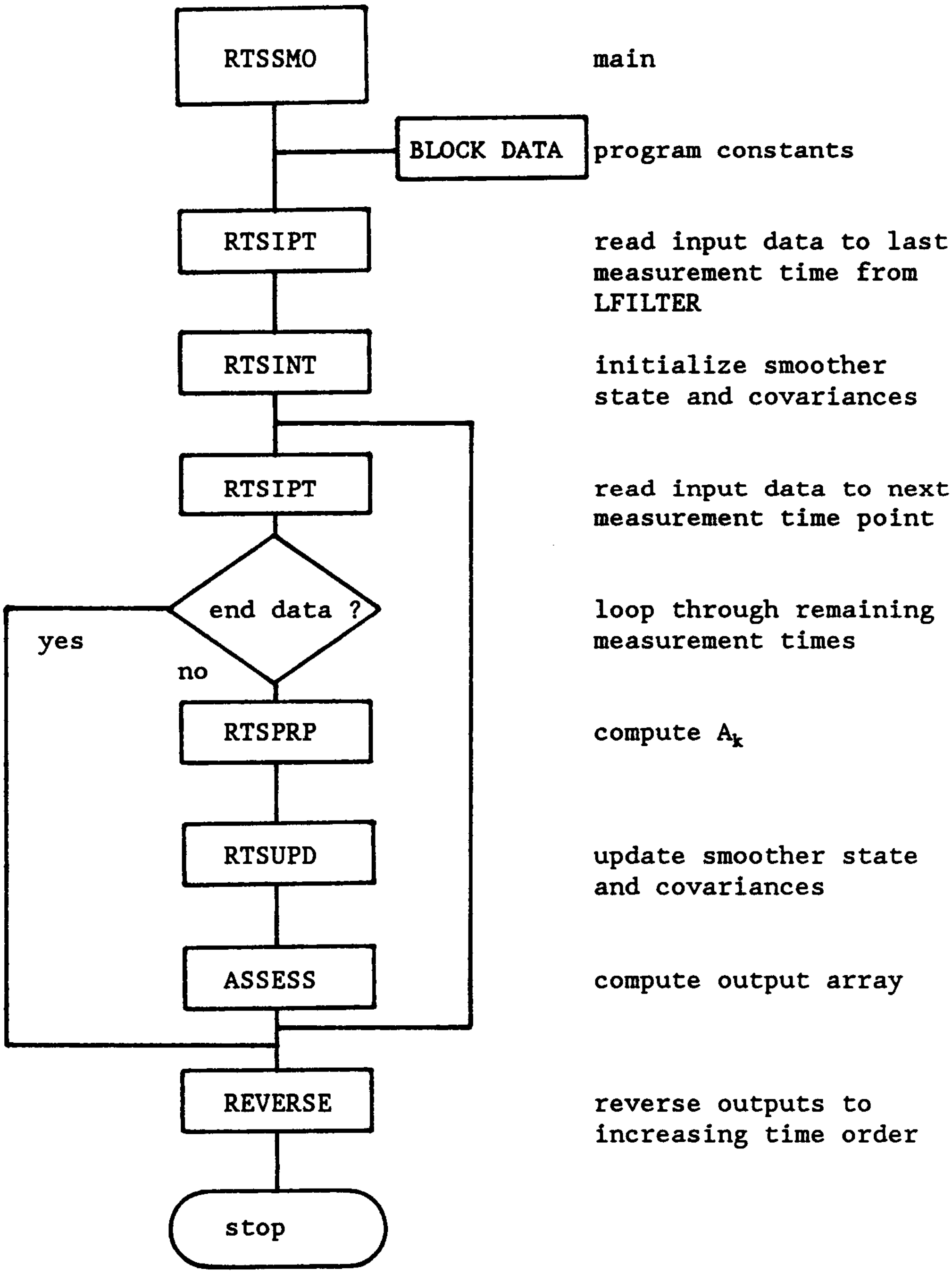


Figure C-2: RTSSMO Program Flow

C.4.2.1 RTSSMO Routine

The main routine, RTSSMO, controls the program flow. It initializes arrays based on data contained in the BLOCK DATA routine, opens and reads data files in RTSIPT, finds the last time specified in BLOCK DATA and initializes the state variables and error covariances in RTSINT, propagates the state variables and error covariances back to a measurement time with subroutine RTSPRP, produces the output variables shown in Table C-3 in the ASSESS subroutine, and reverses the outputs to the normal forward time sequence in REVERSE.

The program flow in controlled by specifying a maximum or beginning time. If the current time is greater than the zero time specified, the program continues the state and error covariance processing. During the processing, data is stored for later plotting.

This routine reprocesses the results of the LFILTER program by reading the two files generated. It also reads the meteorological data for computing the atmospheric dependent variables. The input data files are:

Data File Unit Number	Data File Name
2	SMOIPT.DAT
3	SREFIPT.DAT
4	METDAT.DAT

SMOIPT.DAT contains the filter states and covariances processed in the smoothing algorithm. SREFIPT.DAT contains the reference IMU acceleration, attitude and attitude rate data. METDAT.DAT contains the meteorological data used for the atmospheric dependent variables computed.

Four files are produced by RTSSMO. The output files written are:

Data File Unit Number	Data File Name
6	SMOOUT.DAT
7	SASSOUT.DAT
8	SCRATCH.DAT
9	NREFIPT.DAT

SMOOUT.DAT contains state variable estimates and covariances for quality checks on the smoother's outputs. SASSOUT.DAT contains the output variables summarized in Table C-3. NREFIPT.DAT contains new reference inputs required for the filtering program based on the current filtering/smoothing processing cycle. SCRATCH.DAT is temporary storage for data prior to resequencing the time order of the data.

C.4.2.2 BLOCK DATA routine

The BLOCK DATA routine contains data to support the program flow control, i.e. TMAX for the maximum (beginning) time.

The following program control values are specified in BLOCK DATA.

Symbol	Units	Common	Description
TIME	sec	TDATA	current time after liftoff
TMAX	sec	TDATA	maximum time after liftoff
TSAMP	sec	TDATA	measurement sample time increment
N	-	FILDAT	number of dynamic and bias states
NMEAS	-	FILDAT	number of measurements processed

Earth model specific data and initial vehicle position data are as follows:

Symbol	Units	Common	Description
RE	ft	EDATA	earth model radius at the equator
FLAT	-	EDATA	earth model flattening factor
OMEGE	rad/sec	EDATA	earth angular rotation speed
XMU	ft^3/sec^2	EDATA	earth model gravity parameter
XJ2	-	EDATA	earth model oblate gravity parameter
OLATD	deg	LAUCOR	launch point coordinates
OLONG	deg	LAUCOR	launch point coordinates
OHT	ft	LAUCOR	launch point coordinates

Constants used for conversions are:

Symbol	Units	Common	Description
CRAD	deg/rad	CONST	conversion from radians to degrees
AGRAV	ft/sec^2	CONST	gravity acceleration constant
HRSEC	sec/hr	CONST	conversion from hours to seconds
PERCNT	-	CONST	conversion from factor to percent
XMRAD	-	CONST	conversion from radians to millirads

C.4.2.3 RTSIPT routine

This routine reads the two input files from the filter program which are sequential in time. This routine reads each of these files until the time searched for is located. The data in these files are then communicated via common to other parts of the program.

The following filter output values are read in for each time point;

Symbol	Common	Description
XKM	FILDAT	state vector prior to measurement update
PKM	FILDAT	error covariance matrix prior to update
F	FILDAT	linear dynamics matrix
Q	FILDAT	filter process noise matrix

The units for these arrays corresponds to those identified in Table C-1

The following reference values are read as passed on by the LFILTER program;

Symbol	Units	Common	Description
ABRI	ft/sec^2	RSTATE	BRI acceleration vector
THTI	deg	RSTATE	BRI attitudes (roll, pitch & yaw)
OMEG	deg/sec	RSTATE	BRI attitude rates
RBRI	ft	RSTATE	BRI position vector
VBRI	ft/sec	RSTATE	BRI velocity vector
CIBRI	-	RSTATE	transformation from ECI to BRI
CBRIEF	-	RSTATE	transformation from BRI to ECF

C.4.2.4 RTSINT routine

If the time search for is the maximum (beginning) time, this routine is called. The smoothing algorithm is then initialized with the filter's state estimates and error covariances as specified by the algorithm summarized in Table C-2.

The following are initialized:

Math	Symbol	Common	Description
$\underline{x}_{k\|N}$	XKGN	SMTHER	state estimate at k given N points
$\underline{x}_{k+1\|N}$	XKP1GN	SMTHER	state estimate at k+1 given N points
$P_{k\|N}$	PKGN	SMTHER	covariance at k given N points
$P_{k+1\|N}$	PKP1GN	SMTHER	covariance at K+1 given N points

C.4.2.5 RTSUPD routine

This routine computes the state estimate and covariance updates as specified by the algorithm summarized in Table C-2.

C.4.2.6 RTSPRP routine

This routine computes the smoothing gain matrix used to update the state estimates and covariances:

Math	Symbol	Common	Description
Φ	PHI	SMTHER	state transition matrix
A_k	AK	SMTHER	smoother gain matrix at k

C.4.2.7 ASSESS routine

This routine uses the resulting estimates from the smoothing algorithm to compute estimates of the variables summarized in Table C-5. These variables are formed from combinations of the state elements and meteorological data, tabular functions of altitude, using the filter estimate of altitude. The outputs from this routine are written to the output file, SASSOUT.DAT, for later processing by other NASA programs.

D. Library Routines

Three libraries of routines have been implemented to perform repeated functions for all the programs previously described. This appendix describes these routines.

D.1 ERTHM Library

The ERTHM routines compute various quantities that are dependent on the earth model characteristics as follows:

Routine	Description
COOR	Computes the latitude, longitude and altitude from an earth centered earth fixed set of coordinates.
ECPOS	Computes the earth centered earth fixed set of coordinates from the latitude, longitude and altitude.
AGRAV	Computes the gravitational vector and partial derivative with respect to the earth centered earth fixed coordinates from those coordinates.
CIEFMX	Computes the earth centered inertial to earth centered earth fixed transformation matrix.

D.2 AXMAT Library

This library contains auxiliary matrix operation routines applicable to the aerodynamic and body attitude computations. The routines are:

Routine	Description
AXCBIQ	Computes the partial derivative of the body to inertial transformation matrix with respect to the quaternion elements.
AUXVAB	Computes the partial derivative of total velocity, angle-of-attack, and side-slip with respect to the velocity components.
AXCIBQ	Computes the partial derivative of the inertial to body transformation matrix with respect to the quaternion elements.
QMTRX	Forms the dynamical derivative of the quaternion dynamics.
PQOMEG	Forms the partial derivative of the quaternion dynamics with respect to the body rates.
AXVABQ	Computes the partial derivatives of total velocity, angle-of-attack, and side-slip with respect to the quaternion elements.
CBIMXQ	Forms the body to inertial transformation matrix using quaternion elements.
CBIMX	Forms the body to inertial transformation matrix using body attitude angles.
E2QUAT	Converts Euler angles into quaternion elements.
QUAT2E	Converts quaternion elements into Euler angles.
TMATY	Computes the transformation matrix based on a "yaw" rotation (rotation about the third axis).
TMATP	Computes the transformation matrix based on a "pitch" rotation.

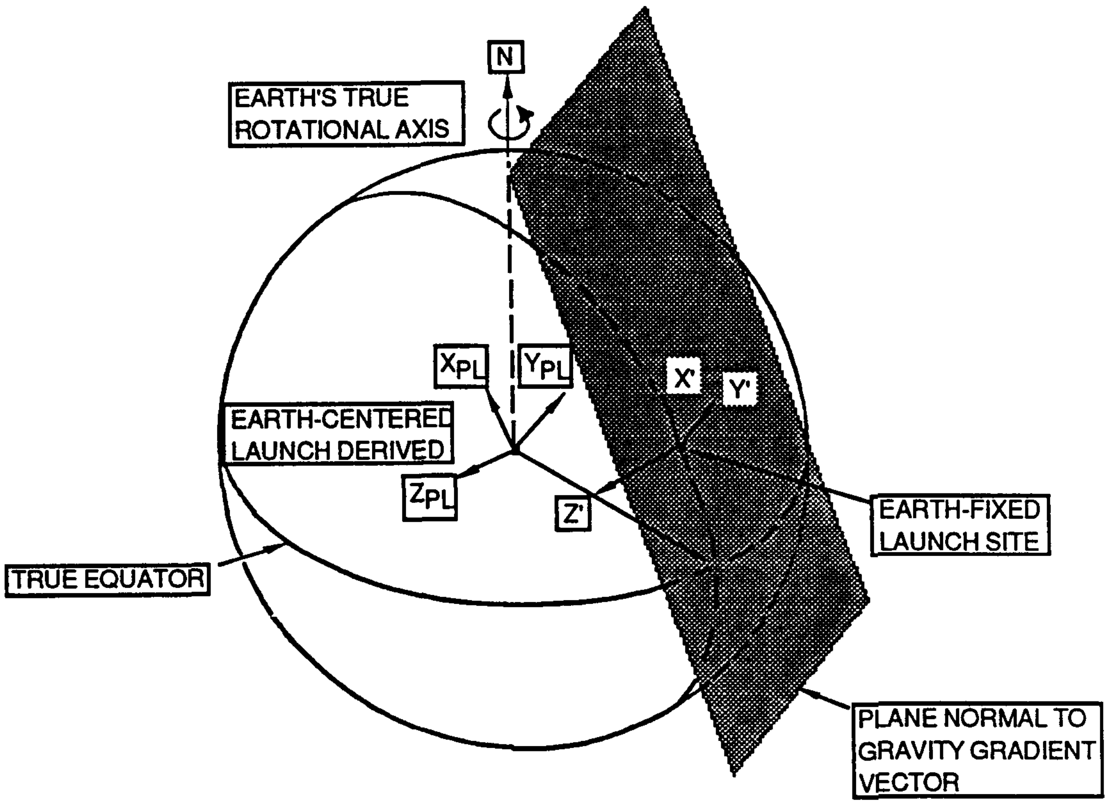
D.3 MLIB Library

The MLIB routines perform various matrix operations as:

Routine	Description
ADD	Adds two compatible subscripted arrays, each up to two dimensions.
SUBT	Subtracts two compatible subscripted arrays, each up to two dimensions.
MULT	Multiplies two compatible subscripted arrays.
TRANS	Transposes a two dimensional array (exchanges its rows and columns).
SMLT	Multiplies a subscripted array, up to two dimensions, by a scaler.
SWITCH	Equates one array, up to two dimensions, with another of the same dimension.
SYMTRK	Symetricizes a square matrix.
OUTER	Forms the outer product of two vectors, yielding a matrix.
INNER	Forms the inner product of two vectors, yielding a scaler.
SKEW	Forms a skew-symmetric matrix from a vector to perform an equivalent vector cross product.
IMBED	Imbeds a smaller matrix within a larger matrix.
INV2X2	Inverts a square two by two matrix.
INVNXN	Inverts a square "n" by "n" matrix.
INV3X3	Inverts a square three by three matrix.
ZEROM	Nulls the elements in a subscripted array with up to two dimensions.
CROSS	Forms a vector cross product, yielding a vector.

Appendix E

Coordinate Frame Definitions



NAME: Boost reference coordinate system.

ORIGIN: At the center of the earth.

ORIENTATION AND DIRECTIONS: The Z_{PL} -axis is parallel to the gravity gradient (Z') which passes through the launch site and is positive in the direction of gravity. The Z_{PL} -axis is fixed at a specifically stated time.
The X_{PL} -axis is paralled to X' which is along the launch site meridian and is positive northward.
The Y_{PL} -axis is parallel to the Y' and completes a standard right-handed system; i.e., positive east.
The X_{PL} - Y_{PL} plane is normal to the launch site gravity gradient vector.

CHARACTERISTICS: Inertial, right-handed, Cartesian.

Figure E.-1: Boost Reference Coordinate System

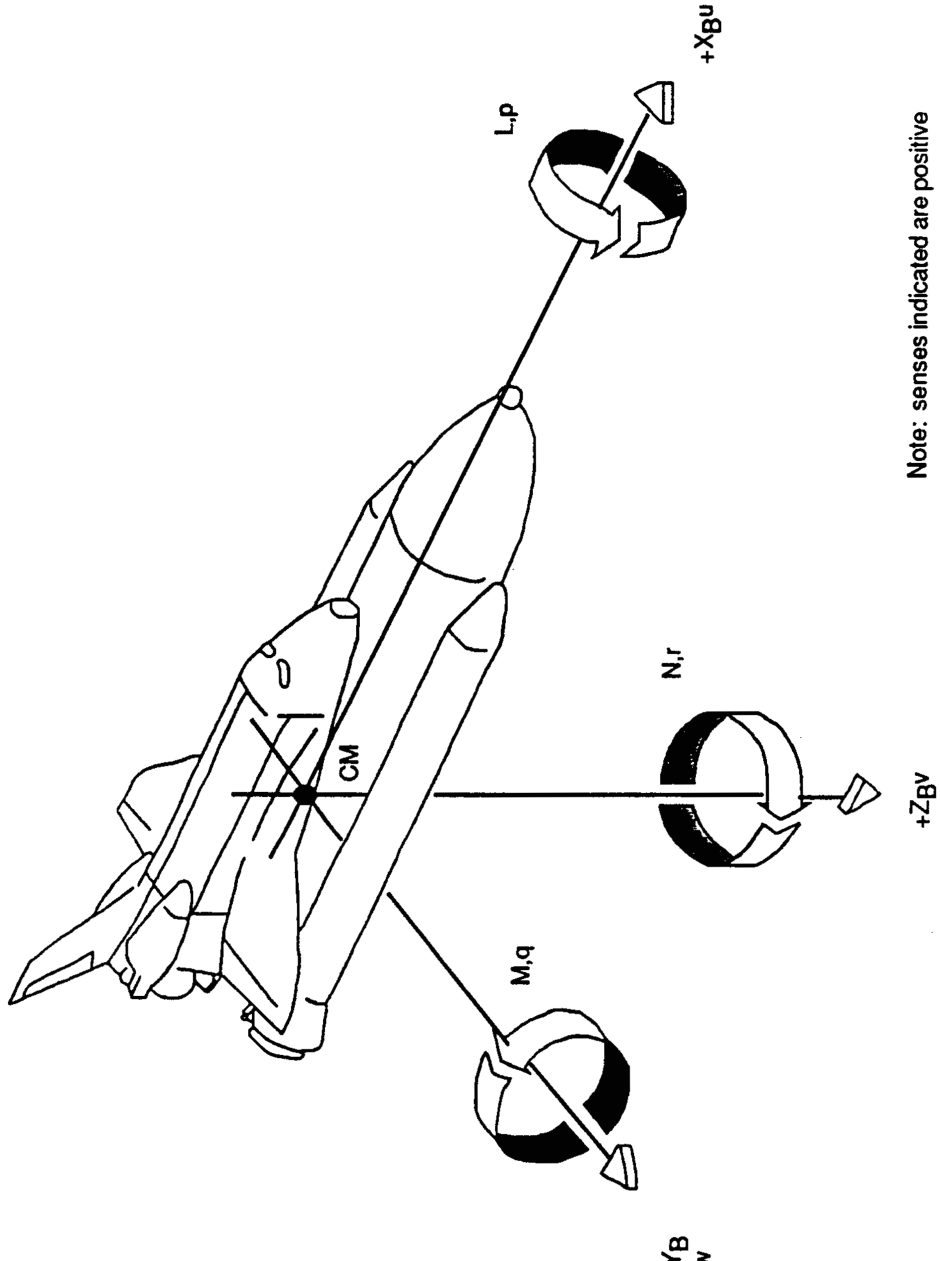


Figure E.-2: Body Axis Frame

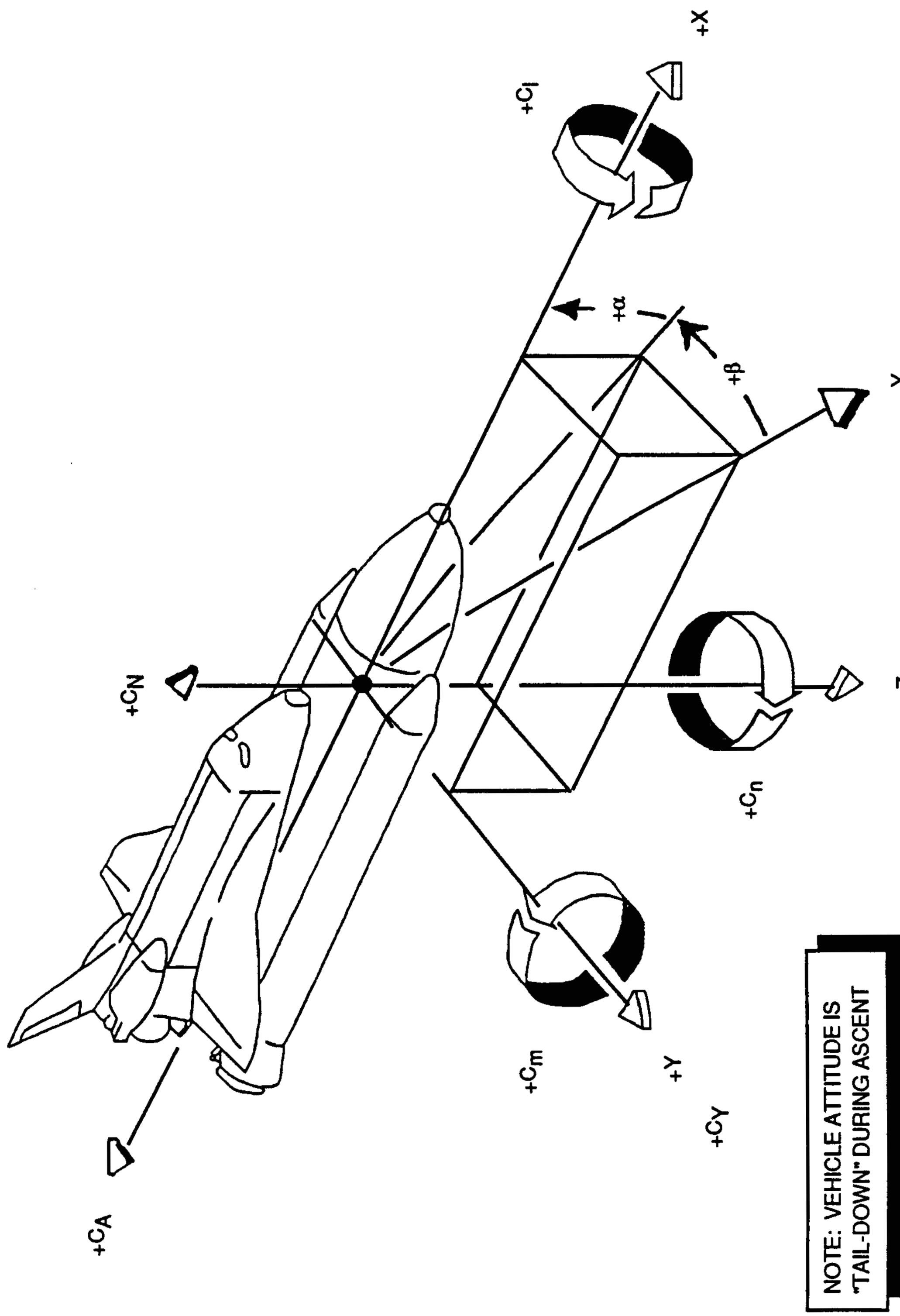


Figure E.-3: Aerodynamic and Plume Force Axes

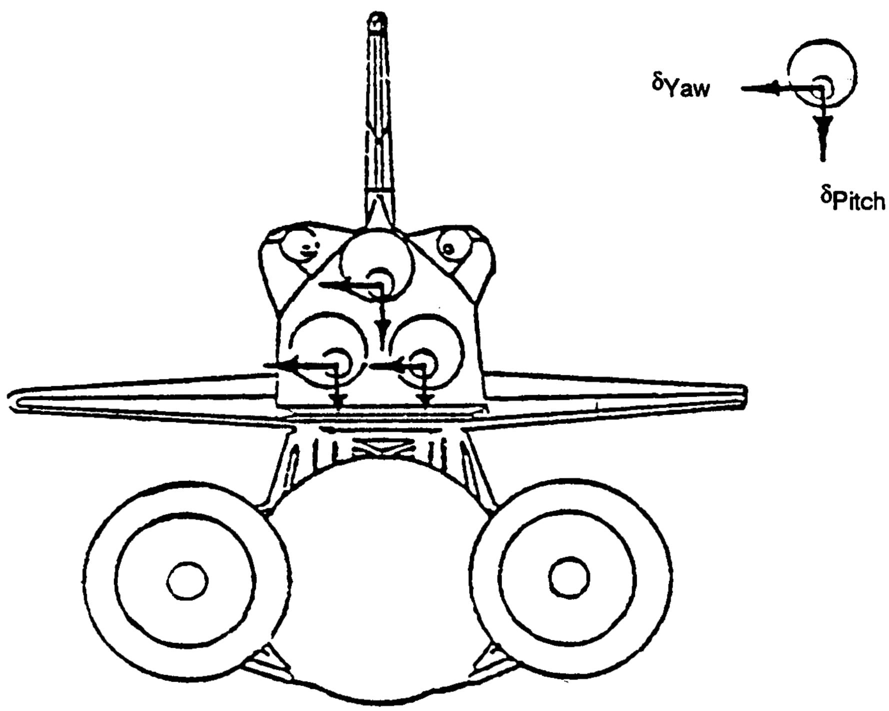


Figure E.-4: Main Engine Gimbal Deflections

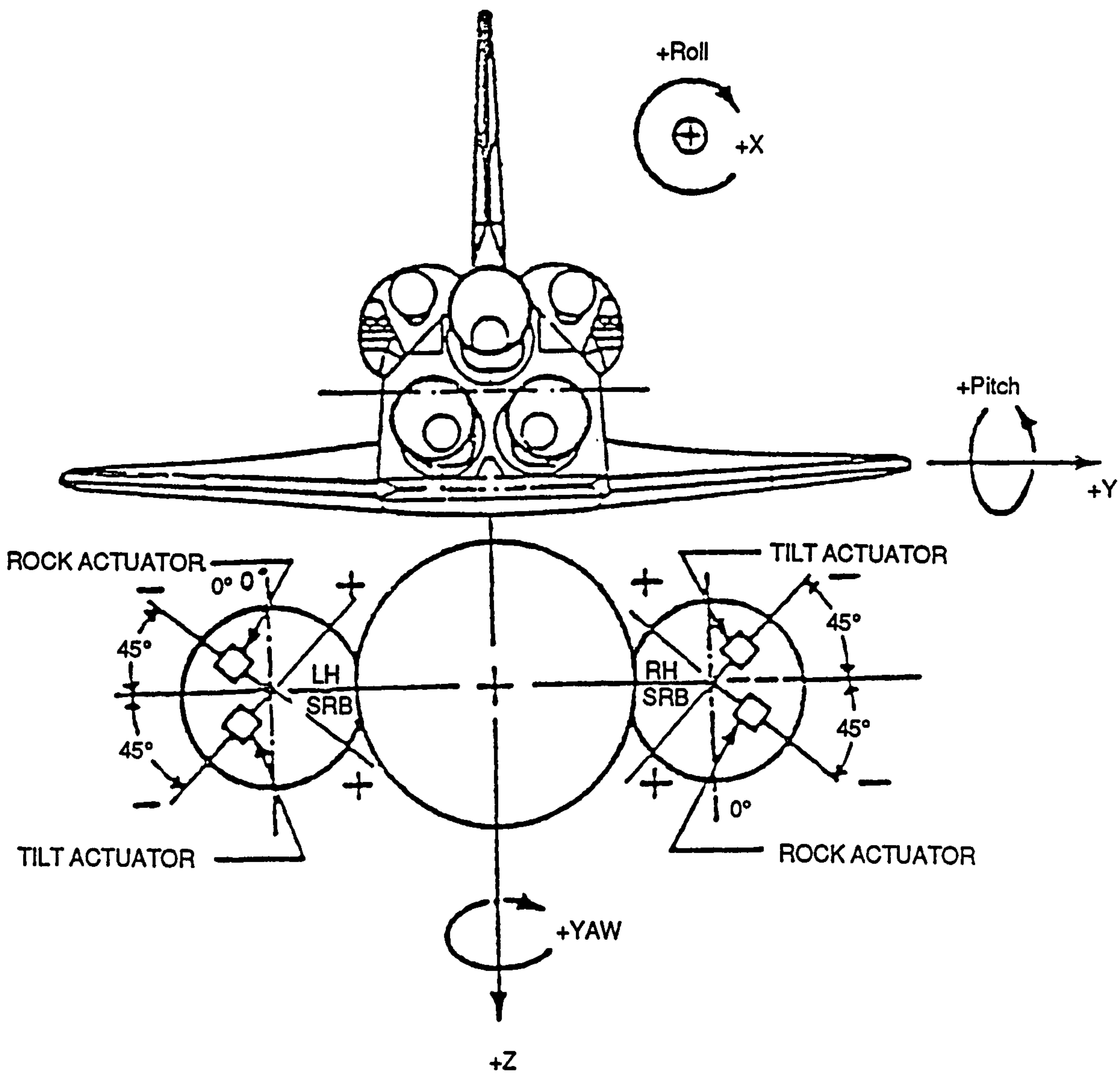


Figure E.-5: SRB Rock and Tilt Displacements

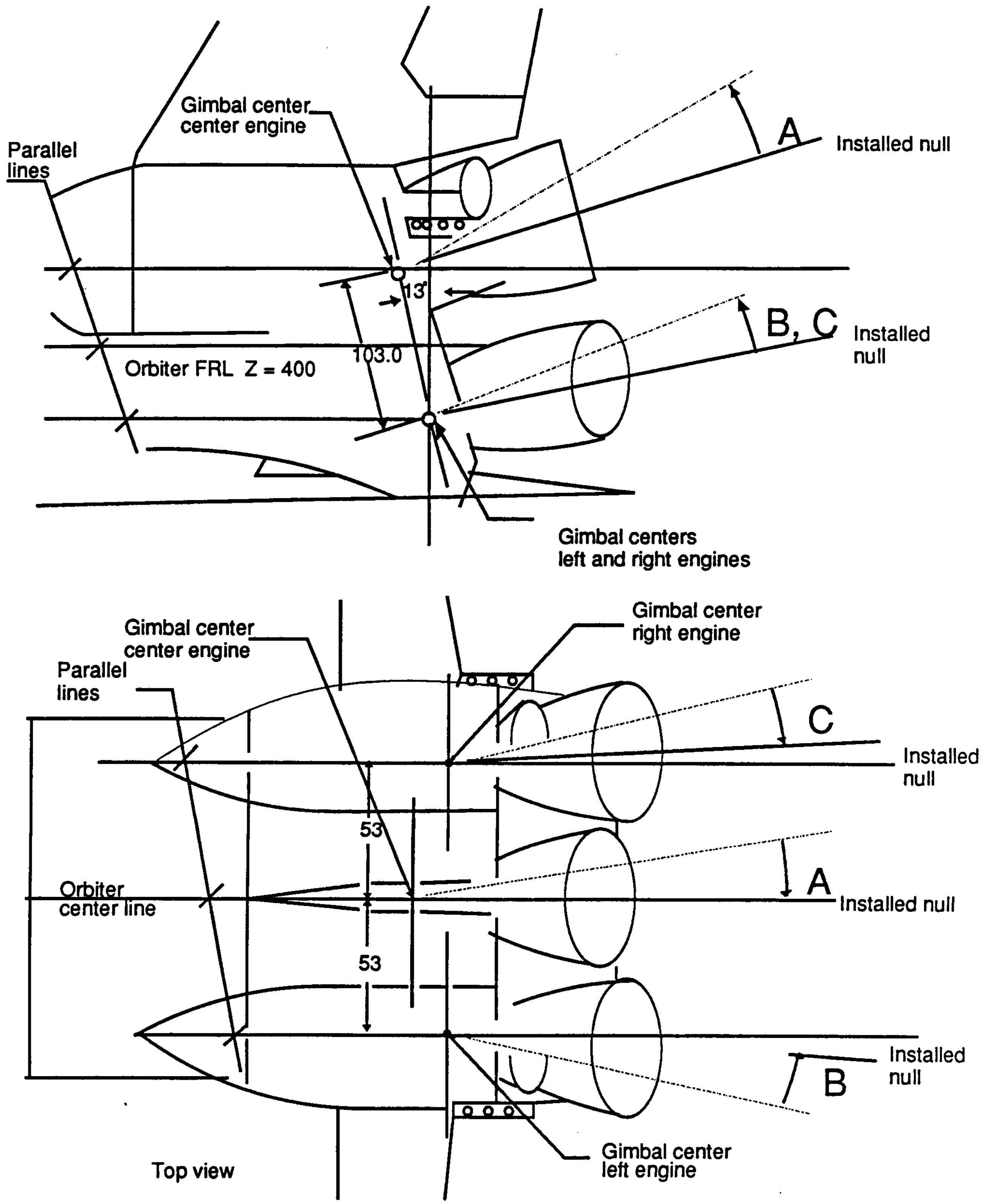


Figure E.-6: Main Engine Structural Deflections

References

1. Final Report, contract NAS8-35324, 1983.

2. Rogers, R. M., 38th progress report, contract NAS8-36152, 1988.

3. Gelb, A., Ed., Applied Optimal Estimation, MIT Press, Cambridge, 1974.

4. Davis, L. D., "Coordinate Systems for the Space Shuttle Program", NASA TM X-58153, 1974.

5. Bierman, G. J., "UDU^T Covariance Factorization for Kalman Filtering", Advances in Control and Dynamic Systems, Vol. 18, C. Leondes, Ed., Academic Press, New York, 1980.

6. Wagner, W. E. and Serold, A. C., "Formulation on Statistical Trajectory Estimation Programs", NASA CR-1482, 1970.

7. Kain, J. E., et al, "Aerodynamic Coefficient Estimation Program User's Manual", Final Report U.S. Air Force contract F08635-80-C-0070, 1980.

8. Internal NASA/MSFC note; CONVRT Q.L.

9. Redus, J. R., private communication.

10. Hill, P. G. and Peterson, C. R., Mechanics and Thermodynamics of Propulsion, Addison-Wesley, Reading, Massachusetts, 1965.

11. Lear, W., "The Ascent/Entry BET Program, LRBET5", NASA JSC report, JSC-19310, 1983.

12. Rogers, R. M., Interim Technical Report, contract NAS8-36152, May 1987.

13. Gerstner, B. A., letter documenting procedure for calculating SSME deflections due to elastic structural deflection, Rockwell International letter 84MA1054, Feb. 1984.

Distribution

	No. Copies
MSFC/AS24D	5
MSFC/AT01	1
MSFC/CC01/Wofford	1
MSFC/EM12A	1
MSFC/EP55	5+repro
NASA/Scientific and Technical Information Facility	1+repro