

AERODYNAMIC PARAMETER ESTIMATION FOR CONTROLLED PARACHUTES

Robert M. Rogers, Associate Fellow
Rogers Engineering & Associates
Gainesville, FL

ABSTRACT

An extended Kalman filter algorithm implementation is used to estimate the aerodynamic, wind, mass property and measurement errors for controlled low-glide parachutes. This implementation incorporates two new approaches: 1) an alternate attitude error model formulation to overcome mathematical singularities associated with vertical flight, and 2) apparent mass characteristics used to describe a parachute's unsteady aerodynamic characteristics. Results based on simulated data show that aerodynamic characteristics and winds can be estimated separately from the apparent mass coefficients.

INTRODUCTION

Ongoing research and development, with recent flight tests of an Affordable Guided Airdrop System (AGAS), have shown that a parachute's trajectory can be controlled with contracting Pneumatic Muscle Actuators¹. Earlier development and demonstrations of Precision Guided Airdrop System (PGAS) for a ram-air parafoil incorporated an Inertial Navigation System aided with the Global Positioning System (INS/GPS) to provide navigation data for parafoil guidance². Low cost controlled parachutes and GPS for guidance data offers the potential for accurate low cost aerial cargo delivery³.

This paper addresses estimation of the aerodynamic characteristics of controlled low-glide parachutes from flight data. The approach presented is based on the Extended Kalman Filtering (EKF) algorithm that implements a non-linear system dynamic model, the equations of motion, and their associated linearizations. For the application of the EKF algorithm to parachute aerodynamic parameter estimation, two primary problems must be addressed and represent the primary contribution of this paper. The first is vertical flight and the mathematical singularities associated with it if the conventional Euler angle dynamics are used for the attitude dynamics portion of the system's dynamics⁴. This problem is addressed by using an attitude error formulation that avoids this singularity. The second to account for a parachute's unsteady motion. This problem is addressed by incorporating added or apparent mass terms in the velocity equations⁵.

The paper is organized as follows. First, the EKF algorithm's processing flow is summarized. Equations of motion are then presented to establish the nomenclature used in this paper. The attitude error formulation, to eliminate the mathematical singularity associated with vertical flight, and the approach to incorporate apparent mass, as part of the motion dynamics, are presented next. A summary of the system error dynamics is presented.

Using simulation, results of applying the algorithm to the aerodynamic parameter estimation for controlled parachutes is presented. Details of the linearizations of the nonlinear equations of motion and measurement equations are presented in the Appendices.

EXTENDED KALMAN FILTERING

The EKF algorithm processing flow is presented in Fig. 1. The algorithm is based on an assumed reference dynamic model - the non-linear equations for the parachute's dynamics with control inputs - \mathbf{u} . Observations of this motion (measurements) - \mathbf{z} are modeled as non-linear functions of the reference model states - \mathbf{x} . Supporting the reference dynamic and observation models are the *a priori* parachute mass properties and aerodynamic characteristics. The EKF algorithm estimates deviations, or errors, from the *a priori* reference model necessary to "match" the observations. Actual recorded control inputs, atmospheric conditions and observations are supplied to the algorithm. Error estimates provided by the EKF algorithm are used to correct the on-going computations. The EKF algorithm for a continuous dynamic system with discrete observations is summarized in Table 1.

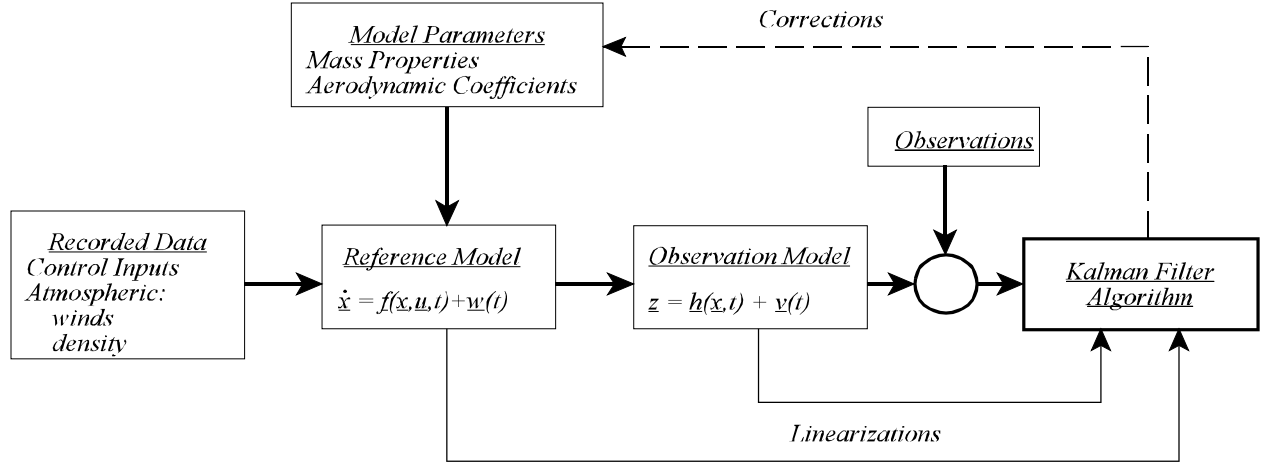


Figure 1: EKF Algorithm Processing Flow

Table 1: Continuous-Discrete Extended Kalman Filtering Algorithm

System Model	$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) + \mathbf{w}(t) \quad E[\mathbf{w}(t) \mathbf{w}(\tau)^T] = \mathbf{Q}\delta(t - \tau)$
Measurement Model	$\mathbf{z}_k = \mathbf{h}_k(\mathbf{x}(t_k)) + \mathbf{v}_k \quad E[\mathbf{v} \mathbf{v}^T] = \mathbf{R}_k$
Initial Conditions	$E[\mathbf{x}(t=0)] = \hat{\mathbf{x}}_0 \quad E[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T] = \mathbf{P}_0$
Assumptions	$E[\mathbf{w} \mathbf{v}^T] = 0 \quad \{\text{considered here}\}$
Propagate State	$\bar{\mathbf{x}} = \mathbf{f}(\bar{\mathbf{x}}, \mathbf{u}, t)$
Propagate Covariance	$\dot{\mathbf{P}}(t) = \mathbf{F}(\bar{\mathbf{x}}(t), t) \mathbf{P}(t) + \mathbf{P}(t) \mathbf{F}(\bar{\mathbf{x}}(t), t)^T + \mathbf{Q}(t)$
Measurement Update	$\hat{\mathbf{x}}_k = \bar{\mathbf{x}}(t_k) + \mathbf{K}_k [\mathbf{z}_k - \mathbf{h}(\bar{\mathbf{x}}(t_k))]$ $\hat{\mathbf{P}}_k = [\mathbf{I} - \mathbf{K}_k \mathbf{H}_k] \bar{\mathbf{P}}(t_k) [\mathbf{I} - \mathbf{K}_k \mathbf{H}_k]^T + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k^T$ $\mathbf{K}_k = \bar{\mathbf{P}}(t_k) \mathbf{H}_k^T (\mathbf{H}_k \bar{\mathbf{P}}(t_k) \mathbf{H}_k^T + \mathbf{R}_k)^{-1}$
Linearizations	
system	$\mathbf{F}(\bar{\mathbf{x}}(t), t) = \left. \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{u}, t)}{\partial \mathbf{x}(t)} \right _{\mathbf{x}(t) = \bar{\mathbf{x}}(t)}$
measurement	$\mathbf{H}_k = \left. \frac{\partial \mathbf{h}_k(\mathbf{x}(t_k))}{\partial \mathbf{x}(t_k)} \right _{\mathbf{x}(t_k) = \bar{\mathbf{x}}(t_k)}$

EQUATIONS OF MOTION

The EKF algorithm's system dynamic model in Table 1 are summarized below. In the following, the parachute is modeled as a rigid body with six degrees-of-freedom⁵.

Coordinate Axes:

The body referenced axes are shown in Fig. 2.

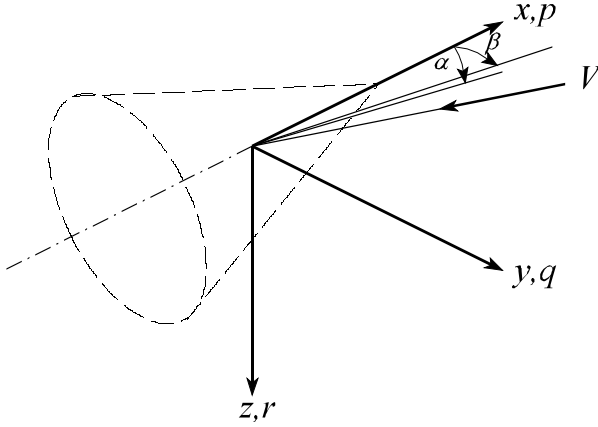


Figure 2: Body Axis Definitions

Position:

The rate of change of position in an inertial reference frame is the transformation of body referenced velocity as

$$\dot{\mathbf{r}}^i = \mathbf{C}_b^i \mathbf{v}^b \quad (1)$$

where

\mathbf{r}^i - inertial position vector (down range, cross range and altitude)

\mathbf{C}_b^i - body to inertial transformation matrix

\mathbf{v}^b - body referenced relative velocity vector

Velocity:

The conventional form⁴ for the rate of change of the body referenced velocity is given by

$$\dot{\mathbf{v}}^b = -\mathbf{\Omega}_{i/b}^b \mathbf{v}^b + \frac{1}{m} \mathbf{f}_{aero}^b + \mathbf{C}_i^b \mathbf{g}^i \quad (2)$$

where

$\mathbf{\Omega}_{i/b}^b$ - skew symmetric matrix equivalent of the vector cross product $(\mathbf{\omega}_{i/b}^b \times)$ where the vector $\mathbf{\omega}_{i/b}^b$ is the rotation of the body relative to the inertial reference coordinatized in the body axes.

\mathbf{f}_{aero}^b - aerodynamic forces in body axes

\mathbf{g}^i - gravity

This form will be modified later in this paper to incorporate apparent mass terms associated with the parachute's unsteady aerodynamics.

The body referenced aerodynamic force vector is

defined in terms of the atmospheric density/wind relative velocity magnitude product (dynamic pressure), aerodynamic reference area and aerodynamic force coefficient as

$$\mathbf{f}^b = \frac{1}{2} \rho |\Delta \mathbf{v}^b|^2 S \mathbf{c}_f = q S \mathbf{c}_f \quad (3)$$

where the density is assumed to comply with the exponential atmospheric model

$$\rho = \rho_0 e^{-h/H_s} \quad (4)$$

The wind relative velocity is the difference in body referenced velocity and winds

$$\Delta \mathbf{v}^b = \mathbf{v}^b - \mathbf{C}_i^b \mathbf{w}^i \quad (5)$$

The aerodynamic force coefficient vector is modeled as the following sum where the control inputs - \mathbf{u} in Fig. 1 are incorporated via the vector $\mathbf{\delta}$

$$\mathbf{c}_f = \mathbf{c}_{f_0} + \mathbf{c}_{f_\alpha} \alpha + \mathbf{c}_{f_\beta} \beta + [\mathbf{c}_{f_\delta}]_{diag} \mathbf{\delta} \quad (6)$$

Referring to Fig. 2, the angle-of-attack and side-slip are defined in terms of the body referenced wind relative velocity components as

$$\alpha = \tan^{-1} \left(\frac{\Delta v_z^b}{\Delta v_x^b} \right) \quad (7)$$

and

$$\beta = \tan^{-1} \left(\frac{\Delta v_y^b}{\Delta v_x^b} \right) \quad (8)$$

Attitude:

The rate of change of the body-to-inertial direction cosine matrix (DCM) is

$$\dot{\mathbf{C}}_b^i = -\mathbf{\Omega}_{i/b}^b \mathbf{C}_b^i \quad (9)$$

Only six of the nine elements in this DCM require numerical integration in the EKF algorithm since its rows and columns form orthogonal unit vectors.

Rotation Rate:

The rate of change of the body rotation rate is given by

$$\dot{\mathbf{\omega}}_{i/b}^b = -[\mathbf{I}]^{-1} \mathbf{\Omega}_{i/b}^b [\mathbf{I}] \mathbf{\omega}_{i/b}^b + [\mathbf{I}]^{-1} \mathbf{T}_{aero}^b \quad (10)$$

where

$$\boldsymbol{\omega}_{i/b}^b \equiv \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (11)$$

$[I]^{-1}$ - inertia matrix inverse

\mathbf{I}_{aero}^b - aerodynamic moments in body axes

The body referenced aerodynamic torque vector is defined in terms of the same dynamic pressure as above for the force vector, aerodynamic reference area and length, and the aerodynamic moment coefficient

$$\mathbf{I}_{aero}^b = q S d \boldsymbol{\epsilon}_m \quad (12)$$

The aerodynamic moment coefficient vector is modeled as the following sum

$$\begin{aligned} \boldsymbol{\epsilon}_m = & \boldsymbol{\epsilon}_{m_0} + \boldsymbol{\epsilon}_{m_\alpha} \alpha + \boldsymbol{\epsilon}_{m_\beta} \beta + \frac{d}{2|\Delta \mathbf{v}^b|} [\boldsymbol{\epsilon}_{m_\omega}]_{diag} \boldsymbol{\omega}_{i/b}^b \\ & + [\boldsymbol{\epsilon}_{m_\delta}]_{diag} \boldsymbol{\delta} \end{aligned} \quad (13)$$

ATTITUDE ERROR DYNAMICS

Using the matrix dynamic equation, Eq. (9), to evolve attitudes, the singularity associated with the Euler angle dynamics is avoided. The implementation of the EKF algorithm requires a linearized form for the attitude dynamics. The use of a small angle “external” attitude error representation to obtain this linearized form for application to aerodynamic parameter estimation represents one of the contributions of this paper. It is assumed that a computed form of the \mathbf{C}_b^i matrix, the result of integrating the DCM differential equation with incorrect initialization and/or body rates, can be related to the true matrix by the following

$$\bar{\mathbf{C}}_b^i = [I - (\boldsymbol{\Phi} \mathbf{x})] \mathbf{C}_b^i \quad (14)$$

The difference between the computed and true DCMs is the DCM error

$$\begin{aligned} \delta \mathbf{C}_b^i &= \bar{\mathbf{C}}_b^i - \mathbf{C}_b^i \\ &= -(\boldsymbol{\Phi} \mathbf{x}) \mathbf{C}_b^i \end{aligned} \quad (15)$$

The equation for the attitude error, $\boldsymbol{\Phi}$, is obtained by taking the derivative of both rows of this equation, then equating the results⁶. The computed DCM’s derivative satisfies the same differential equation form as the true DCM. Taking the derivative of the first row

$$\begin{aligned} \delta \dot{\mathbf{C}}_b^i &= \dot{\bar{\mathbf{C}}}_b^i - \dot{\mathbf{C}}_b^i \\ &= -\bar{\boldsymbol{\Omega}}_{i/b}^b \bar{\mathbf{C}}_b^i + \boldsymbol{\Omega}_{i/b}^b \mathbf{C}_b^i \\ &= -\bar{\boldsymbol{\Omega}}_{i/b}^b [I - (\boldsymbol{\Phi} \mathbf{x})] \mathbf{C}_b^i + \boldsymbol{\Omega}_{i/b}^b \mathbf{C}_b^i \\ &= -[\bar{\boldsymbol{\Omega}}_{i/b}^b - \boldsymbol{\Omega}_{i/b}^b - \bar{\boldsymbol{\Omega}}_{i/b}^b (\boldsymbol{\Phi} \mathbf{x})] \mathbf{C}_b^i \\ &\approx -[\bar{\boldsymbol{\Omega}}_{i/b}^b - \boldsymbol{\Omega}_{i/b}^b - \bar{\boldsymbol{\Omega}}_{i/b}^b (\boldsymbol{\Phi} \mathbf{x})] \mathbf{C}_b^i \end{aligned} \quad (16)$$

Taking the derivative of the second row

$$\begin{aligned} \delta \dot{\mathbf{C}}_b^i &= -(\dot{\boldsymbol{\Phi}} \mathbf{x}) \mathbf{C}_b^i - (\boldsymbol{\Phi} \mathbf{x}) \dot{\mathbf{C}}_b^i \\ &= -(\dot{\boldsymbol{\Phi}} \mathbf{x}) \mathbf{C}_b^i + (\boldsymbol{\Phi} \mathbf{x}) \boldsymbol{\Omega}_{i/b}^b \mathbf{C}_b^i \\ &= -[(\dot{\boldsymbol{\Phi}} \mathbf{x}) - (\boldsymbol{\Phi} \mathbf{x}) \boldsymbol{\Omega}_{i/b}^b] \mathbf{C}_b^i \end{aligned} \quad (17)$$

Equating the results above, and converting from the skew-symmetric matrix form to vector form, yields the following vector differential equation for attitude error

$$\dot{\boldsymbol{\Phi}} = -\mathbf{C}_b^i \delta \boldsymbol{\omega}_{i/b}^b \quad (18)$$

This form for attitude error dynamics requires only three elements as contrasted to four for quaternion implementations used in earlier applications.

UNSTEADY AERODYNAMICS - APPARENT MASS

Unsteady parachute aerodynamics are modeled as additional mass terms for each of the three force vector’s components in body axes⁵. From page 31 of Ref. 5, the following equations, excluding mass center of gravity offsets, are found

$$\Sigma F_x = (m + \alpha_{11})\dot{u} - (m + \alpha_{33})(rv - qw) \quad (19)$$

$$\Sigma F_y = (m + \alpha_{33})(\dot{v} - pw) + (m + \alpha_{11})ru \quad (20)$$

$$\Sigma F_z = (m + \alpha_{33})(\dot{w} + pv) - (m + \alpha_{11})qu \quad (21)$$

where

ΣF_i - sum of aerodynamic force and gravity terms

In these equations, the “y” and “z” components of the added mass α_{ii} are assumed to be equivalent.

The conventional form for the velocity dynamics above is modified to incorporate these added mass terms, in a more generalized form allowing differences in the individual axis components, and becomes

$$\dot{\mathbf{v}}^b = -M^{-1}(\boldsymbol{\omega}_{i/b}^b \mathbf{x}) M \mathbf{v}^b + M^{-1}[\mathbf{f}^b + m \mathbf{C}_i^b \mathbf{g}^i] \quad (22)$$

where

$$M \equiv \begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{bmatrix} \quad (23)$$

and

$$m_i = m + \Delta m_i, \forall i=1,3 \quad (24)$$

where

$$\Delta m_i = k_i V \rho \quad (25)$$

The added mass is proportional to the mass of air contained within the parachute's canopy volume - V with the proportionality dependent on the coefficient k_i .

SYSTEM ERROR DYNAMICS

The system's error dynamics equations are summarized in Table 2. Only the wind and the first term of the aerodynamic force coefficient vector are included in this table (see Appendix A for additional terms). This table summarizes the linearizations using the conventional form of velocity equation - Eq. (2) above. The modifications to this linearized velocity equation form, to incorporate added mass terms, is presented in second part of Appendix A. A complete list of the extended Kalman filter algorithm's error state vector is presented in Table 3.

Table 3: Kalman Filter Error State Vector Elements

error states	description
$\delta x, \delta y, \delta z$	inertial position
$\delta v_x^b, \delta v_y^b, \delta v_z^b$	body velocity
ϕ_x, ϕ_y, ϕ_z	attitude
$\delta \omega_{i/b}^b, \delta \omega_{i/b}^b, \delta \omega_{i/b}^b$	angular rotation rate
$\delta w_x^i, \delta w_y^i$	wind
$\delta c_{f_{a_x}}, \delta c_{f_{a_y}}, \delta c_{f_{a_z}}$	zero angle-of-attack force coef.
$\delta c_{m_{a_y}}, \delta c_{m_{a_z}}$	zero angle-of-attack moment coef.
$\delta c_{m_{x_a}}, \delta c_{m_{x_p}}$	angle-of-attack moment coef.
$\delta c_{m_{v_{x_p}}}, \delta c_{m_{v_{y_p}}}$	control input moment coef.
$\delta c_{f_{x_p}}, \delta c_{f_{x_a}}$	angle-of-attack force coef.
$\delta c_{f_{v_{x_p}}}, \delta c_{f_{v_{y_p}}}$	control input force coef.
$\delta c_{m_a}, \delta c_{m_r}$	moment coefficient damping
$\delta I_y, \delta I_z$	moment of inertia
δd	measurement displacement
δk	apparent mass coef.

Table 2: System Error Dynamics Matrix

	$\delta \underline{r}^i$	$\delta \underline{v}^b$	Φ	$\delta \omega_{i/b}^b$	$\delta \underline{w}^i$
$\delta \underline{r}^i$	$0_{3 \times 3}$	C_b^i	$(C_b^i \underline{v}^b x)$	$0_{3 \times 3}$	$0_{3 \times 3}$
$\delta \underline{v}^b$	$C_i^b \frac{\partial \underline{g}^i}{\partial \underline{r}^i} + \frac{1}{m} \frac{\partial \underline{f}^b}{\partial \underline{r}^i}$	$-(\omega_{i/b}^b x) + \frac{1}{m} \frac{\partial \underline{f}^b}{\partial \underline{v}^b}$	$-C_i^b (\underline{g}^i x) + \frac{1}{m} \frac{\partial \underline{f}^b}{\partial \Phi}$	$(\underline{v}^b x)$	$\frac{1}{m} \frac{\partial \underline{f}^b}{\partial \underline{w}^i}$
Φ	$0_{3 \times 3}$	$0_{3 \times 3}$	$0_{3 \times 3}$	$-C_b^i$	$0_{3 \times 3}$
$\delta \omega_{i/b}^b$	$[I]^{-1} \frac{\partial \underline{f}^b}{\partial \underline{r}^i}$	$[I]^{-1} \frac{\partial \underline{f}^b}{\partial \underline{v}^b}$	$[I]^{-1} \frac{\partial \underline{f}^b}{\partial \Phi}$	$-[I]^{-1} [(\omega_{i/b}^b x)[I] - ([I]\omega_{i/b}^b)x] + [I]^{-1} \frac{\partial \underline{f}^b}{\partial \omega_{i/b}^b}$	$[I]^{-1} \frac{\partial \underline{f}^b}{\partial \underline{w}^i}$

MEASUREMENTS

The observations assumed to be available for processing by the EKF algorithm include inertial position and attitude. A candidate source of position measurements is a GPS unit. Attitude measurements could be provided by either a Attitude Heading Reference System (AHRS) unit or an Inertial Navigation System which is the combination of GPS with a 3-axis navigation system (GPS/INS). With similar aerodynamic characteristics in the “y” and “z” axes, there is little difference in a controlled parachute’s dynamics to distinguish direction. The attitude measurements provides the mechanism to estimate attitude and distinguish the different axes’ contributors.

The linearized form of the observation equations for the EKF algorithm are presented in Appendix B.

SIMULATION RESULTS

The EKF algorithm is evaluated using simulated data. The process is segmented into 1) generating simulated observations and 2) applying the EKF algorithm. The simulation generates the observations using the dynamic equations described above. Known errors are introduced into the simulation, i.e., constant winds, that deviate from the *a priori* reference model. In the results presented below, the simulation includes a single added mass error, the apparent mass coefficients are the same for all axis components. A stable parachute with a constant roll rate is modeled.

It is the task of the EKF algorithm to reproduce the known errors introduced into the simulation. The accuracy of the algorithm can then be assessed by comparing the EKF estimates to these known errors. The EKF algorithm uses the same *a priori* modeling as does the simulation but without the interjected errors. Errors in all the aerodynamic coefficients, winds, mass properties and longitudinal position observation displacement are included in the simulation.

A comparison of observed altitude versus down range (extension of the initial flight direction) and that estimated by the EKF algorithm is presented in Fig. 3 (solid line for the observed and dashed line for the estimate). The companion cross range versus down range results are presented in Fig. 4. At approximately 100 sec into the 300 sec flight, the simulated wind in the down range direction is increased from 1 to 2 ft/sec. During the simulated flight, lateral control inputs combined with the parachute’s roll rate force the trajectory to wander as seen in these figures. No guidance is implemented in the simulation that would assist in controlling the trajectory to a desired impact point.

The wind error estimates are shown in Fig. 5. Down range (solid line) and cross range (dashed line) winds are included. Initially, the interjected down range and cross range wind errors are 1 ft/sec. The wind errors, including

the down range wind error change from 1 to 2 ft/sec, are estimated by the EKF algorithm. Also shown in this figure are the square root of the estimation error covariance matrix diagonal elements (lines with down arrows) corresponding to the down and cross range wind error state vector elements. These latter lines represent the estimation uncertainty associated with the EKF algorithms’ estimates.

Shown in Fig. 6 are estimates and corresponding uncertainties for the interjected zeroth term aerodynamic force coefficient, c_{f_0} , errors. The interjected error for all three elements of this term is 0.01. The EKF algorithm estimates these errors for the two lateral axis elements accurately and a significant portion of the “x” axis component.

The added mass estimate is shown in Fig. 7. The 0.1 interjected error is estimated accurately by EKF algorithm. Finally, the longitudinal position observation displacement 1.0 ft error is estimated by the EKF algorithm. This error represents an error in the assumed position of the position sensor’s, i.e., GPS unit, reference point and the center of gravity.

The results in Fig. 6 for the estimate of the aerodynamic force coefficient deviation, δc_{f_0} , indicates less than total convergence to the true error value of 0.01. A closer examination of the correlations between this error state and others reveals a high correlation with the body velocity deviation δv_x^b . The correlation shown in Fig. 9 is almost unity early in the simulation and is affected by the simulated control inputs. Other error states exhibit less correlation. This lack of correlation between other states is indicative of the algorithm’s estimates convergence to the true error values interjected into the simulation that generates the observation data.

CONCLUSIONS

This paper has presented an approach for the estimation of the aerodynamic characteristics of controlled low-glide parachutes from flight data. The approach presented is based on Extended Kalman Filtering (EKF) algorithm. Two primary problems were addressed and represent the contribution of this paper. The first was vertical flight and was addressed by using an external attitude error formulation. The second was the parachute’s unsteady motion and was addressed by incorporating added or apparent mass terms in the velocity equations. The EKF algorithm’s ability to reproduce known interjected errors was demonstrated using simulation.

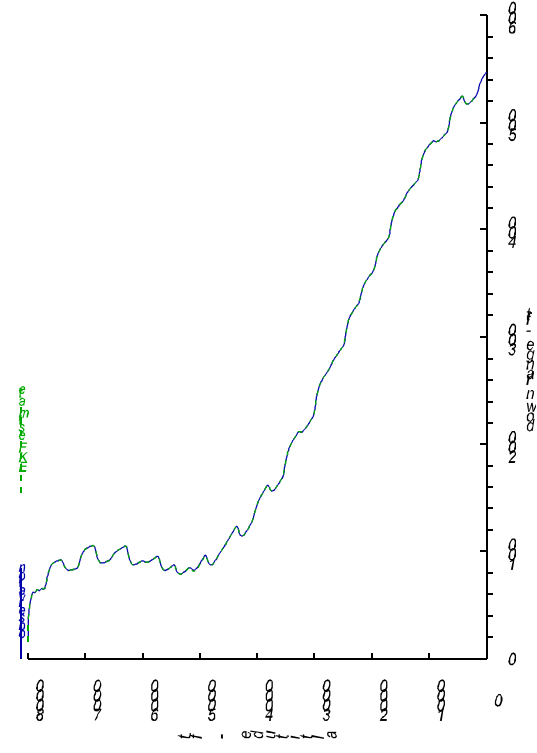


Figure 3: Down Range versus Altitude - Observed and Estimated

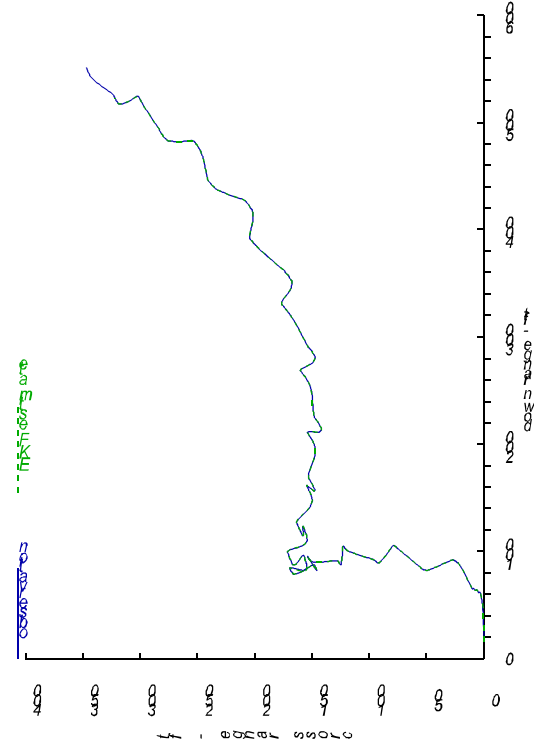


Figure 4: Down Range versus Cross Range - Observed and Estimated

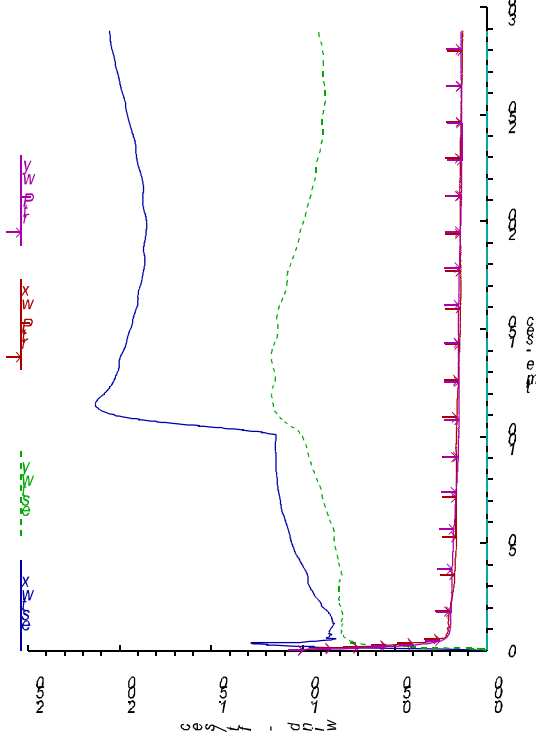


Figure 5: Wind Error Estimates and Uncertainties

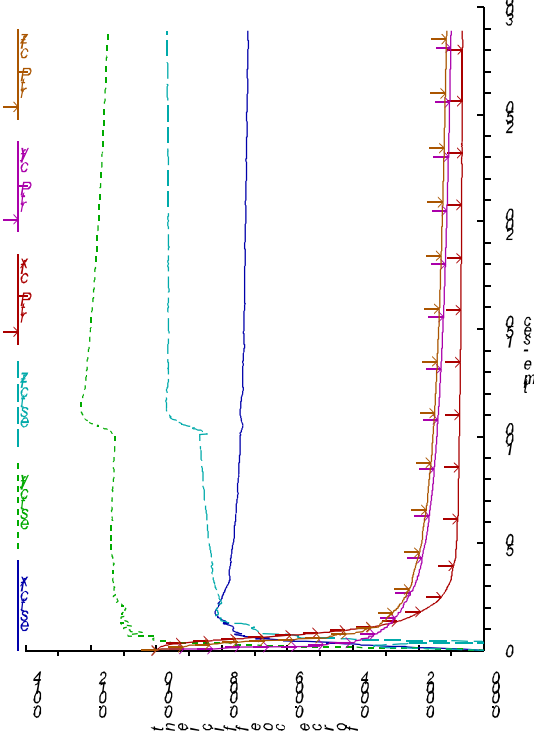


Figure 6: Aerodynamic Force Coefficient - Estimates and Uncertainties

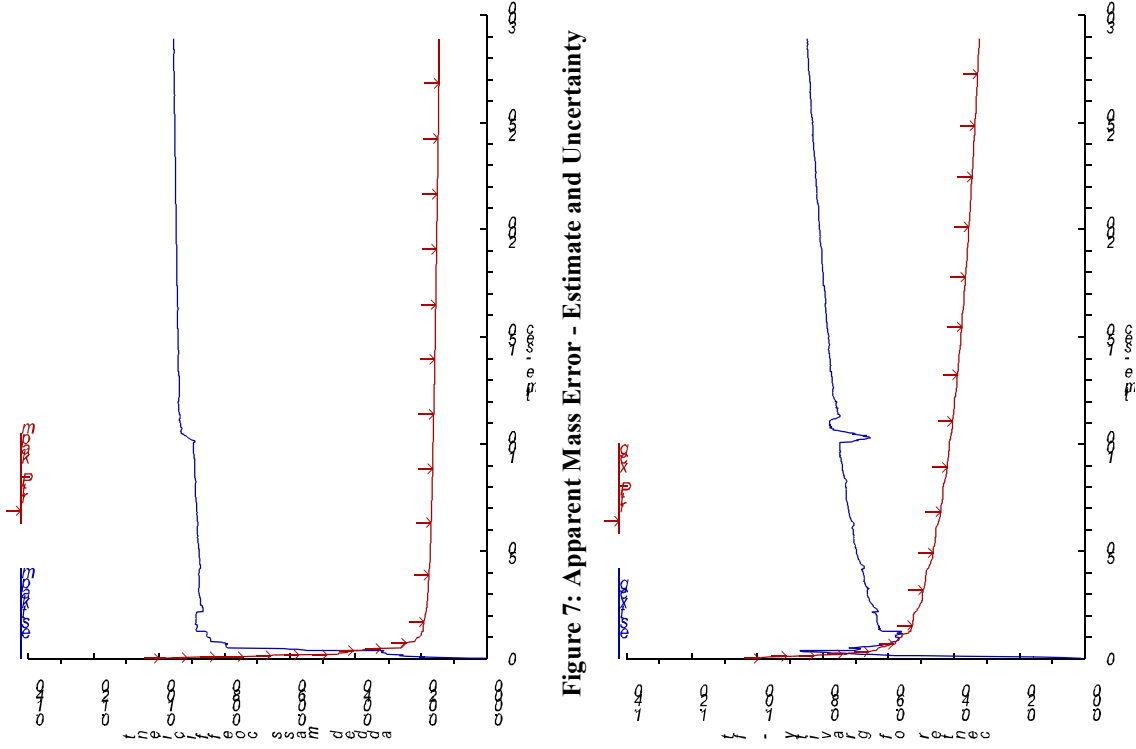


Figure 7: Apparent Mass Error - Estimate and Uncertainty

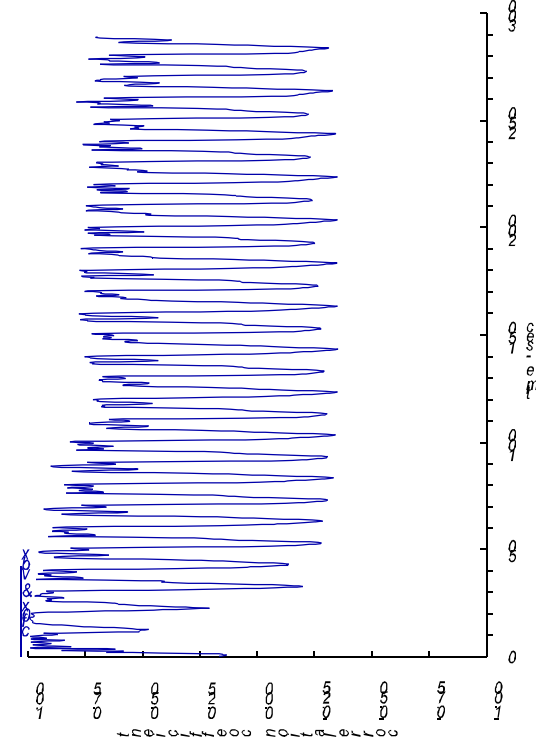


Figure 9: "x" Components of Aerodynamic Force Coefficient and Body Velocity Correlations.

Figure 8: Longitudinal Position Displacement - Estimate and Uncertainty

REFERENCES

1. Brown, G., Haddard, R., Almassy, R., Benney, R., and Dellicker, S., "The Affordable Guided Airdrop System (AGAS)", 15th CAES/AIAA Aerodynamic Decelerator Systems Technology Conference, June 1999.
2. Hattis, P. D., et al, "Final Report: Development and Demonstration Test of a Ram-Air Parafoil Precision Guided Airdrop System", Draper Laboratory Report CSDL-R-2752, October 1996.
3. AIAA Aerodynamic Decelerator Systems Technical Committee, Aerospace America Magazine, December 2000.
4. Etkin, B., Dynamics of Atmospheric Flight, Wiley, 1972.
5. Cockrell, D. J. Ed., The Aerodynamics of Parachutes, AGARD-AG-295, July 1987.
6. Rogers, R. M., "Space Shuttle Propulsion Performance Reconstruction from Flight Data", AIAA Flight Mechanics Conference Proceedings, August 1985, Boston, MA, pp 75-82.
7. Rogers, R. M., Applied Mathematics in Integrated Navigation Systems, AIAA Textbook Series, September 2000.

APPENDIX A

LINEARIZED DYNAMICS EQUATIONS

Velocity without Apparent Mass

Velocity Perturbation Error Dynamics:

$$\delta \dot{\mathbf{v}}^b = -(\boldsymbol{\omega}_{ib}^b \mathbf{x}) \delta \mathbf{v}^b + (\mathbf{v}^b \mathbf{x}) \delta \boldsymbol{\omega}_{ib}^b + \frac{1}{m} \delta \mathbf{f}^b - C_i^b (\mathbf{g}^i \mathbf{x}) \boldsymbol{\phi} + C_i^b \delta \mathbf{g}^i$$

Velocity Linearizations:

$\delta \mathbf{f}^b$: with respect to system variables;

$$\text{position} \quad \frac{\partial \mathbf{f}^b}{\partial \mathbf{r}^i} = \frac{\partial \mathbf{f}^b}{\partial \rho} \frac{\partial \rho}{\partial h} \frac{\partial h}{\partial \mathbf{r}^i}$$

$$\frac{\partial \mathbf{f}^b}{\partial \rho} = \frac{1}{2} |\Delta \mathbf{v}^b|^2 S c_f; \quad \frac{\partial \rho}{\partial h} = -\frac{\rho}{H_s}$$

$$\frac{\partial \mathbf{f}^b}{\partial \mathbf{r}^i} = \begin{bmatrix} 0 & 0 & -1 \end{bmatrix}$$

$$\text{velocity} \quad \frac{\partial \mathbf{f}^b}{\partial \mathbf{v}^b} = \frac{\partial \mathbf{f}^b}{\partial \Delta \mathbf{v}^b} \frac{\partial \Delta \mathbf{v}^b}{\partial \mathbf{v}^b}$$

$$\frac{\partial \mathbf{f}^b}{\partial \Delta \mathbf{v}^b} = \left(\frac{\partial \mathbf{f}^b}{\partial q} \frac{\partial q}{\partial \Delta \mathbf{v}^b} + \frac{\partial \mathbf{f}^b}{\partial \alpha} \frac{\partial \alpha}{\partial \Delta \mathbf{v}^b} + \frac{\partial \mathbf{f}^b}{\partial \beta} \frac{\partial \beta}{\partial \Delta \mathbf{v}^b} \right)$$

$$\frac{\partial \mathbf{f}^b}{\partial q} = S c_f; \quad \frac{\partial \mathbf{f}^b}{\partial \alpha} = q S c_{f_\alpha}; \quad \frac{\partial \mathbf{f}^b}{\partial \beta} = q S c_{f_\beta}$$

$$\frac{\partial q}{\partial \Delta \mathbf{v}^b} = \begin{bmatrix} \rho |\Delta \mathbf{v}_x| & \rho |\Delta \mathbf{v}_y| & \rho |\Delta \mathbf{v}_z| \end{bmatrix}; \quad \frac{\partial \Delta \mathbf{v}^b}{\partial \mathbf{v}^b} = I$$

$$\frac{\partial \alpha}{\partial \Delta \mathbf{v}^b} = \left[\frac{-\Delta v_z^b}{\Delta v_x^{b^2} + \Delta v_z^{b^2}} \mid 0 \mid \frac{\Delta v_x^b}{\Delta v_x^{b^2} + \Delta v_z^{b^2}} \right]$$

$$\frac{\partial \beta}{\partial \Delta \mathbf{v}^b} = \left[\frac{-\Delta v_y^b}{\Delta v_x^{b^2} + \Delta v_y^{b^2}} \mid \frac{\Delta v_x^b}{\Delta v_x^{b^2} + \Delta v_y^{b^2}} \mid 0 \right]$$

$$\text{attitude} \quad \frac{\partial \mathbf{f}^b}{\partial \boldsymbol{\phi}} = \frac{\partial \mathbf{f}^b}{\partial \Delta \mathbf{v}^b} \frac{\partial \Delta \mathbf{v}^b}{\partial \boldsymbol{\phi}}$$

$$\frac{\partial \Delta \mathbf{v}^b}{\partial \boldsymbol{\phi}} = C_i^b (\mathbf{w}^i \mathbf{x})$$

$\delta \mathbf{f}^b$: with respect to parameters

$$\text{wind} \quad \frac{\partial \mathbf{f}^b}{\partial \mathbf{w}^i} = \frac{\partial \mathbf{f}^b}{\partial \Delta \mathbf{v}^b} \frac{\partial \Delta \mathbf{v}^b}{\partial \mathbf{w}^i}$$

$$\frac{\partial \Delta \mathbf{v}^b}{\partial \mathbf{w}^i} = -C_i^b$$

$$\text{aerodynamics} \quad \frac{\partial f^b}{\partial c_f} = (q S) I$$

Rotation Rate

Rotation Rate Perturbation Error Dynamics:

$$\delta \dot{\omega}_{ib}^b = -[I]^{-1}[(\omega_{ib}^b \times) [I] - (([I] \omega_{ib}^b) \times)] \delta \omega_{ib}^b + [I]^{-1} \delta T^b$$

Rotation Rate Linearizations:

δT^b : with respect to system variables:

$$\text{position} \quad \frac{\partial T^b}{\partial \underline{r}^i} = \frac{\partial T^b}{\partial \rho} \frac{\partial \rho}{\partial h} \frac{\partial h}{\partial \underline{r}^i}$$

$$\frac{\partial T^b}{\partial \rho} = \frac{1}{2} |\Delta \underline{v}^b|^2 S d c_m$$

$$\text{velocity} \quad \frac{\partial T^b}{\partial \underline{v}^b} = \frac{\partial T^b}{\partial \Delta \underline{v}^b} \frac{\partial \Delta \underline{v}^b}{\partial \underline{v}^b}$$

$$\frac{\partial T^b}{\partial \Delta \underline{v}^b} = \left(\frac{\partial T^b}{\partial q} \frac{\partial q}{\partial \Delta \underline{v}^b} + \frac{\partial T^b}{\partial \alpha} \frac{\partial \alpha}{\partial \Delta \underline{v}^b} + \frac{\partial T^b}{\partial \beta} \frac{\partial \beta}{\partial \Delta \underline{v}^b} + \frac{\partial T^b}{\partial |\Delta \underline{v}^b|} \frac{\partial |\Delta \underline{v}^b|}{\partial \Delta \underline{v}^b} \right) \delta \underline{v}^b = -\delta M^{-1}(\omega_{ib}^b \times) M \underline{v}^b + M^{-1} \delta(\omega_{ib}^b \times) M \underline{v}^b + M^{-1}(\omega_{ib}^b \times) \delta M \underline{v}^b + M^{-1}(\omega_{ib}^b \times) M \delta \underline{v}^b + \delta M^{-1} f^b + M^{-1} \delta f^b + \delta M^{-1} m C_i^b g^i + M^{-1} m \delta C_i^b g^i + M^{-1} m C_i^b \delta g^i$$

$$\frac{\partial T^b}{\partial q} = S d c_m; \quad \frac{\partial T^b}{\partial \alpha} = q S d c_{m_\alpha}$$

$$\frac{\partial T^b}{\partial \beta} = q S d c_{m_\beta}$$

$$\frac{\partial T^b}{\partial |\Delta \underline{v}^b|} = -\frac{q S d^2}{2 |\Delta \underline{v}^b|^2} [c_{m_\omega}]_{diag} \omega_{ib}^b$$

$$\frac{\partial |\Delta \underline{v}^b|}{\partial \Delta \underline{v}^b} = \left[\frac{\Delta v_x}{|\Delta \underline{v}^b|} \mid \frac{\Delta v_y}{|\Delta \underline{v}^b|} \mid \frac{\Delta v_z}{|\Delta \underline{v}^b|} \right]$$

$$\text{attitude} \quad \frac{\partial T^b}{\partial \Phi} = \frac{\partial T^b}{\partial \Delta \underline{v}^b} \frac{\partial \Delta \underline{v}^b}{\partial \Phi}$$

$$\text{rate} \quad \frac{\partial T^b}{\partial \omega_{ib}^b} = \frac{q S d^2}{2 |\Delta \underline{v}^b|} [c_{m_\omega}]_{diag}$$

δT^b : with respect to parameters

$$\text{wind} \quad \frac{\partial T^b}{\partial \underline{w}^i} = \frac{\partial T^b}{\partial \Delta \underline{v}^b} \frac{\partial \Delta \underline{v}^b}{\partial \underline{w}^i}$$

Inertia

$$\delta \omega_{ib}^b = \dots \begin{bmatrix} -\frac{T_x^b}{I_x} & 0 & 0 \\ -\frac{pr}{I_y} & -\frac{1}{I_y^2} [T_y^b - pr(I_x - I_z)] & \frac{pr}{I_y} \\ \frac{pq}{I_z} & -\frac{pq}{I_z} & -\frac{1}{I_z^2} [T_z^b - pq(I_y - I_x)] \end{bmatrix} \begin{bmatrix} \delta I_x \\ \delta I_y \\ \delta I_z \end{bmatrix}$$

Velocity Error Equation Apparent Mass Terms

Error in apparent mass:

$$\delta m_i = \rho V \delta k_i - \frac{\rho V k_i}{H_s} \delta h$$

Velocity perturbation error dynamics:

$$\begin{aligned} & -\delta M^{-1}(\omega_{ib}^b \times) M \underline{v}^b + M^{-1} \delta(\omega_{ib}^b \times) M \underline{v}^b \\ & + M^{-1}(\omega_{ib}^b \times) \delta M \underline{v}^b + M^{-1}(\omega_{ib}^b \times) M \delta \underline{v}^b \\ & + \delta M^{-1} f^b + M^{-1} \delta f^b + \delta M^{-1} m C_i^b g^i \\ & + M^{-1} m \delta C_i^b g^i + M^{-1} m C_i^b \delta g^i \end{aligned}$$

first and third terms:

$$-\delta M^{-1}(\omega_{ib}^b \times) M \underline{v}^b + M^{-1}(\omega_{ib}^b \times) \delta M \underline{v}^b$$

$$= \begin{bmatrix} \frac{rm_2 v - qm_3 w}{m_1^2} & -\frac{rv}{m_1} & \frac{qw}{m_1} \\ \frac{ru}{m_2} & \frac{-rm_1 u + pm_3 w}{m_2^2} & -\frac{pw}{m_2} \\ -\frac{qu}{m_3} & \frac{pv}{m_3} & \frac{qm_1 u - pm_2 v}{m_3^2} \end{bmatrix} \begin{bmatrix} \delta m_1 \\ \delta m_2 \\ \delta m_3 \end{bmatrix}$$

second term:

$$M^{-1} \delta(\omega_{ib}^b \times) M \underline{v}^b = \begin{bmatrix} 0 & \frac{m_3 w}{m_1} & -\frac{m_2 v}{m_1} \\ -\frac{m_3 w}{m_2} & 0 & \frac{m_1 u}{m_2} \\ \frac{m_2 v}{m_3} & -\frac{m_1 u}{m_3} & 0 \end{bmatrix} \delta \omega_{ib}^b$$

fourth term:

$$M^{-1}(\omega_{ib}^b x) M \delta v^b = \begin{bmatrix} 0 & \frac{-m_2 r}{m_1} & \frac{m_3 q}{m_1} \\ \frac{m_1 r}{m_2} & 0 & \frac{-m_3 p}{m_2} \\ \frac{-m_1 q}{m_3} & \frac{m_2 p}{m_3} & 0 \end{bmatrix} \delta v^b$$

seventh term:

$$\delta M^{-1} a = - \begin{bmatrix} \frac{a_1}{m_1^2} & 0 & 0 \\ 0 & \frac{a_2}{m_2^2} & 0 \\ 0 & 0 & \frac{a_3}{m_3^2} \end{bmatrix} \begin{bmatrix} \delta m_1 \\ \delta m_2 \\ \delta m_3 \end{bmatrix}$$

where

$$a \equiv f^b + m C_i^b g^i$$

remaining terms:

$$M^{-1} \delta a = \begin{bmatrix} \frac{1}{m_1} & 0 & 0 \\ 0 & \frac{1}{m_2} & 0 \\ 0 & 0 & \frac{1}{m_3} \end{bmatrix} \delta a$$

where

$$\delta a = \delta f^b + m \delta C_i^b g^i + m C_i^b \delta g^i$$

APPENDIX B MEASUREMENTS

Position

Position measured is the sum of computed position, based on the mass center of gravity (c.g.), and a translation from the c.g. to the sensor providing the position, i.e., a GPS unit.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}_m = \begin{bmatrix} \bar{x} \\ \bar{y} \\ \bar{z} \end{bmatrix}_i + \begin{bmatrix} \overline{\Delta x} \\ \overline{\Delta y} \\ \overline{\Delta z} \end{bmatrix}_i$$

where the computed c.g. position is assumed to be composed of the true position and an error.

$$\begin{bmatrix} \bar{x} \\ \bar{y} \\ \bar{z} \end{bmatrix}_i = \begin{bmatrix} x \\ y \\ z \end{bmatrix}_i + \begin{bmatrix} \delta x \\ \delta y \\ \delta z \end{bmatrix}_i$$

The position from the c.g. to the sensor is translated by the product of the computed attitude DCM and an assumed distance along the “x” axis.

$$\begin{bmatrix} \overline{\Delta x} \\ \overline{\Delta y} \\ \overline{\Delta z} \end{bmatrix}_i = \begin{bmatrix} \overline{C}_{xx} & \overline{C}_{xy} & \overline{C}_{xz} \\ \overline{C}_{yx} & \overline{C}_{yy} & \overline{C}_{yz} \\ \overline{C}_{zx} & \overline{C}_{zy} & \overline{C}_{zz} \end{bmatrix} \begin{bmatrix} \bar{d} \\ 0 \\ 0 \end{bmatrix}_b$$

Then, the difference between the measured position and the true position becomes

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}_m - \begin{bmatrix} x \\ y \\ z \end{bmatrix}_i = \begin{bmatrix} \delta x \\ \delta y \\ \delta z \end{bmatrix}_i + \delta \begin{bmatrix} \overline{\Delta x} \\ \overline{\Delta y} \\ \overline{\Delta z} \end{bmatrix}_i$$

where

$$\delta \begin{bmatrix} \overline{\Delta x} \\ \overline{\Delta y} \\ \overline{\Delta z} \end{bmatrix}_i = d \begin{bmatrix} 0 & -C_{xx} & C_{yx} \\ C_{xx} & 0 & -C_{zx} \\ -C_{yx} & C_{xx} & 0 \end{bmatrix} \phi + \begin{bmatrix} C_{xx} \\ C_{yx} \\ C_{zx} \end{bmatrix} \delta d$$

Attitude

Attitude computed from integrating the DCM is post multiplied by the corresponding true DCM, transformed, to yield the attitude error.

$$(\overline{C}_b^i) (C_b^i)^T = I - (\Phi x)$$

then

$$\phi_x = \overline{C}_{yx} C_{zx} + \overline{C}_{yy} C_{zy} + \overline{C}_{yz} C_{zz}$$

$$\phi_y = \overline{C}_{zx} C_{xx} + \overline{C}_{zy} C_{xy} + \overline{C}_{zz} C_{xz}$$

$$\phi_z = \overline{C}_{xx} C_{yx} + \overline{C}_{xy} C_{yy} + \overline{C}_{xz} C_{yz}$$