

ESTIMATION OF THE PARAMETERS OF A SPACECRAFT'S ATTITUDE DYNAMICS MODEL USING FLIGHT DATA*

Mark L. Psiaki
Cornell University

ABSTRACT

An algorithm has been developed that uses flight data to estimate the parameters in an attitude dynamics model of a spacecraft. The new algorithm's estimates can enhance the fidelity of Euler equation models that are used to implement back-up mode attitude determination and control functions. The algorithm's estimation equation is an integrated version of Euler's equation expressed in inertial coordinates. It uses 3-axis attitude data and 3-axis rate-gyro data to yield a set of linear equations in the unknown dynamics parameters, which include moments and products of inertia and scale factors, alignments, and biases for all reaction wheels and magnetic torque rods. The estimation problem statement includes the statistics of unmodeled torques and sensor errors, and it incorporates a scalar quadratic constraint to overcome the usual unobservability of the overall scaling of the dynamic parameters. The sensor errors enter the model equation in a multiplicative fashion, which yields a total least-squares problem. The solution algorithm employs a guarded Newton iteration and special recursive factorizations that deal efficiently with the dynamic structure of the measurement error effects. The algorithm has been applied to the Microwave Anisotropy Probe (MAP) spacecraft, and the resulting parameter estimates show an ability to reduce torque modeling errors by a factor of 5 to 10.

INTRODUCTION

The goal of this research is to develop improved Euler attitude dynamics models for spacecraft by using flight data to estimate model parameters. Parameters that might be estimated include the independent elements of the moment-of-inertia matrix, reaction wheels' and magnetic torque rods' scale-factors and alignments, and residual spacecraft angular momentum and magnetic dipole moment. The latter quantities are the equivalent of biases in the reaction wheels and torque rods. The estimation procedure presumes the availability of very accurate attitude and rate data from star trackers and rate gyros. It also requires nominal reaction wheel angular momenta, nominal torque rod dipole moments, and magnetometer outputs, if relevant.

The primary motivation for this work is to improve the performance of back-up mode attitude determination systems. Such systems use a reduced set of sensors because of the failure of primary sensors. They must be able to operate without rate-gyro data or with incomplete rate-gyro information. This type of system relies on Euler's equation to propagate attitude rate estimates as in Refs. 1-7. Improved Euler models also could be used to develop back-up mode slewing and pointing controllers that could function with a reduced actuator set.

Back-up systems that rely on an Euler equation model exploit the property of observability or controllability of an attitude dynamics system in order to accomplish 3-axis attitude determination or control without the need for full 3-axis sensing or actuation. The stability and accuracy/pointing performance of all such systems are highly dependent on the accuracy with which the rate dynamics can be modeled by Euler's equations. System designers normally try to avoid relying on Euler's equation because there is usually a significant level of uncertainty about the relevant parameters and disturbance torques.

One may not be able to avoid reliance on Euler's equation when in back-up mode. Therefore, it would be good to have better parameter estimates. The scenario of this paper envisions the availability of highly accurate information from sensors during a time period before back-up mode operation is required. Its estimation algorithm can use this data to develop attitude parameter estimates which could be used as part of a back-up system that could operate later in the mission if any of the primary sensors failed.

A number of previous works have attempted to estimate attitude dynamics parameters. Reference 2 performed off-line estimation of moment-of-inertia parameters by considering the nutation mode oscillations of a spinning spacecraft. The direction, phasing, and frequency of these oscillations were used to correct 5 of the 6 inertia matrix elements. Other efforts estimated corrections to 5 of the 6 inertia matrix elements as part of an EKF whose primary function was to estimate rate only or attitude and rate^{4,6,7}. A common theme of these efforts is the attempt to improve parameter estimates in the Euler model

* This work was supported in part by NASA Goddard Spaceflight Center under grant number NAG5-11919.

based on limited sensor data of moderate to coarse accuracy. None of these systems attempted to estimate dynamics parameters other than elements of the moment-of-inertia matrix.

This paper's principal contributions are to define an attitude dynamics parameter estimation problem and to develop an algorithm to solve it. The problem is based on satisfying Euler's equation. A general set of Euler dynamics parameters can constitute the problem's unknowns. These parameters enter Euler's equation linearly. The problem formulation presumes that accurate 3-axis attitude and rate sensor time histories are available. The measurement errors, such as rate-gyro errors and errors in the reported reaction wheel angular momenta, enter as perturbations to the coefficient matrix that multiplies the unknown parameter vector in the estimation equation. This bilinear form of the model equation yields what is known as a total least-squares problem^{8,9}. The solution algorithm simultaneously estimates the unknown parameter vector and the multiplicative measurement errors.

There exist closed-form global solutions to certain total least-squares problems⁸, but such techniques are not applicable to the attitude dynamics parameter estimation problem because their measurement error models are not general enough. The philosophy taken here is that it is better to solve the desired problem using a less general, less robust algorithm than to use a more general, more robust algorithm that forces unrealistic modeling assumptions. The exact solution of the wrong problem is normally not very useful!

This paper's algorithm applies a mixture of linear and nonlinear optimization techniques in a divide-and-conquer approach. Recursive linear techniques are used to perform a global inner optimization that yields the best estimates of the multiplicative measurement errors for a fixed guess of the attitude dynamics parameter vector. The recursion takes advantage of the dynamic structure of the measurement error effects. An outer loop optimizes the least-squares cost via Newton's iterative numerical method. Safe-guards are employed to ensure convergence to at least a local minimum of the least-squares cost function.

This paper develops a technique for dealing with the unobservability of the overall parameter scaling, which manifests itself in the form of a homogeneous least-squares problem. A homogeneous problem is ill-posed, having the trivial solution $\mathbf{x} = 0$. This paper recovers observability and makes the problem well posed by adding a priori information in the form of a scalar constraint on a quadratic function of the unknown parameter vector. Such a constraint, if properly designed, does not significantly bias the parameter estimates.

The paper's other main contribution is to test its new algorithm on actual spacecraft data. These tests demonstrate the effectiveness of the method, and they illustrate difficulties that can arise when using actual data. In particular, the results illustrate the significant impact of certain types of measurement errors and the importance of using data that includes sufficiently rich variations of the control input and attitude response time histories.

The remaining 5 sections of this paper describe the parameter estimation problem, the solution algorithm, and the results that have been obtained for a test case. Section II sets up the total least-squares attitude parameter estimation problem. Section III develops the solution algorithm for the estimation problem. Section IV presents a recursive QR factorization algorithm that gets used to efficiently implement parts of Section III's algorithm. Section V applies the algorithm to data from the MAP spacecraft, and Section VI presents the paper's conclusions.

II. AN ATTITUDE PARAMETER ESTIMATION PROBLEM BASED ON EULER'S EQUATION

Euler's equation for the attitude dynamics of a rigid spacecraft with reaction/momentum wheels equates the time rate of change of angular momentum in inertial coordinates to the applied external torque. A trapezoidally integrated version of Euler's equation takes on the following form when expressed in inertial coordinates:

$$\begin{aligned} & A_{k+1}^T \{ I_m \mathbf{w}_{k+1} + C_{rw} h_{wk+1} + h_{bias} - \frac{1}{2} (t_{k+1} - t_k) [-b_{sck+1} \times (C_{mt} m_{k+1} + m_{bias}) + n_{gg} (A_{k+1}, G_{ink+1}, I_m)] \} \\ & = A_k^T \{ I_m \mathbf{w}_k + C_{rw} h_{wk} + h_{bias} + \frac{1}{2} (t_{k+1} - t_k) [-b_{sck} \times (C_{mt} m_k + m_{bias}) + n_{gg} (A_k, G_{ink}, I_m)] \} + \mathbf{D} h_{ink} \end{aligned} \quad (1)$$

The subscripts k and $k+1$ in this equation refer to values at the sample times of the beginning and end of the numerical integration interval, t_k and t_{k+1} . Other quantities in eq. (1) are A , the direction cosines matrix for the transformation from inertial coordinates to spacecraft coordinates, I_m , the moment-of-inertia matrix of the spacecraft body in body coordinates, \mathbf{w} , the angular velocity of the spacecraft with respect to inertial coordinates expressed in body coordinates, C_{rw} , a scale-factor/alignment matrix for the reaction wheels, h_w , the vector of nominal reaction wheel angular momenta caused by their rotation rates with respect to the spacecraft, h_{bias} , the net bias angular momentum in spacecraft coordinates of the reaction wheels and any other spacecraft instruments when $h_w = 0$, b_{sc} , the magnetic field vector in spacecraft coordinates, C_{mt} , a scale-factor/alignment matrix for the magnetic torque rods, m , the vector of nominal torque rod magnetic dipole moments, m_{bias} , the

net bias dipole moment in spacecraft coordinates of the torque rods and any other spacecraft instruments when $m = 0$, $n_{gg}(A, G_{in}, I_m)$, the gravity gradient torque in spacecraft coordinates, and Dh_{in} , the net inertial impulse due to unmodeled torques during the integration interval. The gravity gradient torque is a function of the attitude matrix, the gravity-gradient tensor in inertial coordinates, G_{in} , and the moment-of-inertia matrix.

The unknown parameters that get estimated are I_m , C_{rw} , h_{bias} , C_{mt} , and m_{bias} . It is assumed that the other quantities in the equation are all known, possibly with some measurement error. Star tracker data are used to determine A_k and A_{k+1} . Rate gyros provide w_k and w_{k+1} . A magnetometer measures b_{sck} and b_{sck+1} . The command and data handling system records h_{wk} , h_{wk+1} , m_k , and m_{k+1} . The orbital ephemerides and an Earth gravity model can be used to compute G_{ink} and G_{ink+1} .

The unknown parameters can be stacked into an estimation vector:

$$x = [I_{m11}, I_{m12}, I_{m13}, I_{m22}, I_{m23}, I_{m33}, c_{rw1}^T, c_{rw2}^T, \dots, c_{rwnw}^T, h_{bias}^T, c_{mt1}^T, \dots, c_{mnt}^T, m_{bias}^T]^T \quad (2)$$

where I_{mij} is the ij^{th} element of I_m , c_{rwj} is the j^{th} column of C_{rw} , and c_{mtj} is the j^{th} column of C_{mt} . The number of reaction wheels is n_w , and the number of magnetic torque rods is n_t . Thus, there are $n_x = 12 + 3(n_w + n_t)$ elements in this vector of unknown parameters.

The parameter estimation algorithm determines the parameter values that minimize the sum of the square errors in Euler's equation. It minimizes the weighted sum of the squared magnitudes of the residual unmodeled angular impulse time history $Dh_{in0}, Dh_{in1}, Dh_{in2}, \dots, Dh_{inN-1}$. Each Dh_{ink} gets divided by its per-axis standard deviation $S_{Dhk} = \sqrt{Dt_k q_{Dh}}$, where $Dt_k = t_{k+1} - t_k$. This operation normalizes the impulse errors before computing the sum of their squares. The quantity q_{Dh} is a continuous-time white process noise intensity and is expressed in $N^2\text{-m}^2\text{-sec}$ units. The unmodeled Dh_{ink} impulses arise due to solar and albedo radiation pressure torque, aerodynamic drag torque, and thermal radiation pressure torque.

An integrated version of Euler's equation is used in order to avoid the need to differentiate the measured angular velocity vector time history, w_k, w_{k+1}, \dots . Trapezoidal integration yields a reasonable approximation if the rotation $0.5(w_k + w_{k+1})(t_{k+1} - t_k)$ has a magnitude much smaller than 1. If the gravity-gradient and magnetic torques are negligible, then trapezoidal integration is exact. Inertial coordinates are used in order to avoid the squaring of angular velocity that appears in the $w \times I_m w$ term of the spacecraft-referenced version of Euler's equation.

Equation (1) could be generalized to include articulating appendages, such as solar arrays. The estimator would need articulation angle data in this case, and it could estimate corrections to a subset of the appendage's moment-of-inertia elements. If the spacecraft dynamics included significant flexibility or fuel slosh motions, then eq. (1) would be a poor dynamic model, and this paper's estimation algorithm would not produce useful results.

All of the unknown parameters enter eq. (1) linearly. This is obvious for all terms except the gravity gradient function. The gravity gradient formula is:

$$n_{gg}(A, G_{in}, I_m) = \begin{bmatrix} \{G_{sc23}(I_{m22} - I_{m33}) + (G_{sc33} - G_{sc22})I_{m23} + G_{sc13}I_{m12} - G_{sc12}I_{m13}\} \\ \{G_{sc13}(I_{m33} - I_{m11}) + (G_{sc11} - G_{sc33})I_{m13} + G_{sc12}I_{m23} - G_{sc23}I_{m12}\} \\ \{G_{sc12}(I_{m11} - I_{m22}) + (G_{sc22} - G_{sc11})I_{m12} + G_{sc23}I_{m13} - G_{sc13}I_{m23}\} \end{bmatrix} \quad (3)$$

where $G_{sc} = AG_{in}A^T$ is the gravity-gradient tensor in spacecraft coordinates. The elements of I_m enter eq. (3)'s gravity gradient torque model linearly.

Measurement errors and the integrated effects of unmodeled disturbance torques cause eq. (1) not to be satisfied exactly. Suppose that the true values, the measured values, and the measurement errors for the angular rate vector, the nominal wheel angular momenta, the magnetic field vector, and the nominal magnetic torque rod dipole strengths are:

$$w_k = w_{meas} + Dw_k \quad (4a)$$

$$h_{wk} = h_{wmeas} + Dh_{wk} \quad (4b)$$

$$b_k = b_{meas} + Db_k \quad (4c)$$

$$m_k = m_{meas} + Dm_k \quad (4d)$$

The quantities on the left-hand sides of these equations are the unknown true values that appear in eq. (1). The terms with the $(\)_{meas}$ subscript on the right-hand sides are the known measured quantities, and the terms with the D prefix are the unknown measurement errors. The effects of star tracker measurement errors on A_k are not considered because they are normally insignificant.

The estimation algorithm lumps all of the measurement errors into a single error vector that is re-scaled so that each of

its entries has unit variance:

$$v_k = [\mathbf{D}\mathbf{w}_k^T/\mathbf{s}_w, \mathbf{D}h_{wk}^T/\mathbf{s}_{hw}, \mathbf{D}b_k^T/\mathbf{s}_b, \mathbf{D}m_k^T/\mathbf{s}_m]^T \quad (5)$$

where \mathbf{s}_w , \mathbf{s}_{hw} , \mathbf{s}_b , and \mathbf{s}_m are the per-axis measurement error standard deviations of, respectively, the angular rate vector, the nominal wheel angular momenta, the magnetic field vector, and the nominal torque rod dipole moments. The estimation algorithm assumes that v_k is a zero-mean, identity-covariance, discrete-time, white-noise Gaussian random process, i.e., that $v_k \sim N(0, I)$ and that $E\{v_j v_k^T\} = 0$ if $j \neq k$. The dimension of v_k is $n_v = 6 + n_w + n_t$.

The definitions in eqs. (2) and (4a)-(5) and the gravity gradient torque formula in eq. (3) can be used to re-write eq. (1) in the following generalized form:

$$[E_{0k+1} + \sum_{i=1}^{n_v} (v_{k+1})_i E_{ik+1}]x = [D_{0k} + \sum_{i=1}^{n_v} (v_k)_i D_{ik}]x + \mathbf{h}_k \quad (6)$$

where the notation $(\)_i$ refers to the i^{th} element of the vector in question and where $\mathbf{h}_k = \mathbf{D}h_{ink}/\mathbf{s}_{dhk}$ is the normalized angular impulse error vector. The $3 \times n_x$ matrices D_{0k} , D_{ik} , E_{0k+1} and E_{ik+1} can be computed from the measurements. Their formulas can be derived based on eqs. (1)-(5). The formulas for several columns of these matrices are given below as examples:

$$\text{column 2 of } D_{0k} = \left(\frac{1}{\mathbf{s}_{dhk}} \right) A_k^T \left\{ \begin{bmatrix} (\mathbf{w}_{meask})_2 \\ (\mathbf{w}_{meask})_1 \\ 0 \end{bmatrix} + \frac{\mathbf{D}t_k}{2} \begin{bmatrix} (G_{sck})_{13} \\ -(G_{sck})_{23} \\ (G_{sck})_{22} - (G_{sck})_{11} \end{bmatrix} \right\} \quad (7a)$$

$$\text{columns 7 through 9 of } D_{4k} = \left(\frac{1}{\mathbf{s}_{dhk}} \right) A_k^T \mathbf{s}_{hw} \quad (7b)$$

$$\text{column 11 of } E_{0k+1} = \left(\frac{1}{\mathbf{s}_{dhk}} \right) A_{k+1}^T \begin{bmatrix} 0 \\ (h_{wmeask+1})_2 \\ 0 \end{bmatrix} \quad (7c)$$

$$\text{columns 6 and 7 of } E_{3k+1} = \left(\frac{1}{\mathbf{s}_{dhk}} \right) A_{k+1}^T \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \mathbf{s}_w & 0 \end{bmatrix} \quad (7d)$$

where the notation $(\)_{ij}$ refers to the ij^{th} element of the matrix in question. The remaining columns of the eq.-(6) matrices are straightforward to compute. Their formulas have been omitted for the sake of brevity.

Equation (6) fails to be an exact representation of eq. (1) on one point. It has no provision to include the products of magnetic field measurement error vectors $\mathbf{D}b$ and magnetic torque rod dipole moment measurement errors $\mathbf{D}m$. These error product terms have been neglected because their inclusion would unnecessarily complicate the measurement model in eq. (6). These errors are small relative to the corresponding measured values. Their product should be negligible.

A lumped total least-squares parameter estimation problem can be defined by considering eq. (6) for the N different intervals, $k = 0, 1, 2, \dots, N-1$. Suppose that one lumps the measurement errors into one large vector and the torque errors into another large vector:

$$v_{big} = [v_0^T, v_1^T, v_2^T, \dots, v_N^T]^T \quad (8a)$$

$$\mathbf{h}_{big} = [\mathbf{h}_0^T, \mathbf{h}_1^T, \mathbf{h}_2^T, \dots, \mathbf{h}_{N-1}^T]^T \quad (8b)$$

The length of the large measurement error vector is $n_{vbig} = n_v(N+1)$, and the length of the large torque error vector is $n_{hbig} = 3N$. One can lump the N copies of eq. (6) into the following large system of measurement equations:

$$[H_0 + \sum_{j=1}^{n_{vbig}} H_j (v_{big})_j]x = \mathbf{h}_{big} \quad (9)$$

where the large H matrices are:

$$H_0 = \begin{bmatrix} (E_{01} - D_{00}) \\ (E_{02} - D_{01}) \\ (E_{03} - D_{02}) \\ \vdots \\ (E_{0N} - D_{0N-1}) \end{bmatrix}, \quad H_1 = \begin{bmatrix} -D_{10} \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad H_2 = \begin{bmatrix} -D_{20} \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \dots, \quad H_{n_v} = \begin{bmatrix} -D_{n_v 0} \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix},$$

$$H_{nv+1} = \begin{bmatrix} E_{11} \\ -D_{11} \\ 0 \\ \vdots \\ 0 \end{bmatrix}, H_{nv+2} = \begin{bmatrix} E_{21} \\ -D_{21} \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \dots, H_{2nv} = \begin{bmatrix} E_{nv1} \\ -D_{nv1} \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \dots, H_{nvbig} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ -D_{nvN} \end{bmatrix} \quad (10)$$

Lumped measurement equation (9) and the statistical models of the random error vectors v_{big} and \mathbf{h}_{big} can be used to define the following nonlinear least-squares cost function:

$$J(x, v_{big}) = \frac{1}{2} x^T [H_0 + \sum_{j=1}^{n_{vbig}} H_j(v_{big})_j]^T [H_0 + \sum_{j=1}^{n_{vbig}} H_j(v_{big})_j] x + \frac{1}{2} v_{big}^T v_{big} \quad (11)$$

This paper's algorithm estimates x and v_{big} by finding the values that minimize this cost function subject to a constraint. This cost function equals a constant plus the negative log of the probability density function for the error vectors v_{big} and \mathbf{h}_{big} conditioned on the parameters given in x . Therefore, the minimizing x value is a maximum likelihood estimate, and the minimizing v_{big} value is a maximum a posteriori estimate.

Extra information must get included in the estimation problem in order for it to generate a non-trivial solution. The unconstrained global minimum of $J(x, v_{big})$ occurs at $x = 0$, $v_{big} = 0$ because eq. (9) lacks non-homogeneous terms. The physical reason for this is that all of the parameters in the x vector can be scaled up or down without affecting the dynamic response of the system. A large spacecraft and a small spacecraft will undergo the same dynamic response if the ratios of the parameters in the x vector are preserved.

This paper's problem definition adds scaling information in the form of a scalar quadratic constraint on x that is part of the optimization problem. This constraint takes the form:

$$0 = 1 - (Lx)^T (Lx) \quad (12)$$

where L is a non-zero matrix with n_x columns and at least 1 row. It is necessary to choose L wisely in order to get reasonable estimates from the algorithm. Most of the problems solved in this paper use an L matrix that has $3n_w$ rows. All of its columns are zero except for columns 7 through $(6 + 3n_w)$. These columns are set equal to the identity matrix divided by $\sqrt{n_w}$. Given this definition, the constraint in eq. (12) forces the square of the Frobenius norm of C_{rw} to equal n_w . This constraint controls scaling in a reasonable way: it forces the average of the squares of the reaction wheels' scale factor recalibrations to equal 1. If a given spacecraft has no reaction wheels, then a similar constraint could be applied to the square of the Frobenius norm of the magnetic torque rod scale-factor/alignment matrix C_{mr} . Alternatively, it would be appropriate to enforce a constraint on the square of the Frobenius norm of I_m for a spacecraft that has neither reaction wheels nor magnetic torque rods. The best choice of L is problem dependent.

The candidate L choices mentioned above have the common property that they act equally about all 3 spacecraft axes. One might be tempted to form a constraint that involves quantities only from one axis. In theory, such a constraint should serve to enforce a reasonable scaling, but in practice, a single-axis constraint often leads to poor estimates. Difficulties arise when there is only weak dynamic coupling between the axes. In this situation, the axis with the constraint has a reasonable scaling, but the other 2 axes have scalings that are too small. The estimator reduces the other 2 axes' contributions to $J(x, v_{big})$ simply by scaling down the x components associated with these axes. The resulting estimates can have nonsensical relative scalings between the axes, scalings which can produce moment-of-inertia estimates that fail to obey the physical constraint that the maximum eigenvalue of I_m must be less than the sum of the remaining two eigenvalues. If the L matrix weights all 3 axes equally, however, then the quadratic constraint will tend to balance the relative scaling between the axes in a reasonable manner.

Another type of scaling error can arise even when the quadratic constraint weights the 3 axes equally. Consider an example with weak coupling in which the measurements or unmodeled torques on one axis are much noisier than on another axis. The estimation algorithm will scale down the elements of x associated with the noisier axis in order to better minimize $J(x, v_{big})$. In this situation, the user would have to recognize that one axis is noisier than the others and re-scale that axis' measurement error or torque \mathbf{s} values so that the matrices of eq. (6) will properly account for the increased noise level.

There is an option to incorporate a priori information about the x parameter vector. This information takes the form of an a priori square-root information equation¹⁰:

$$R_{ap}x + z_{ap} = \mathbf{h}_{ap} \quad (13)$$

where R_{ap} and z_{ap} are the a priori square root information matrix and vector, respectively, and where \mathbf{h}_{ap} is a zero-mean, identity-matrix-covariance Gaussian random vector, $\mathbf{h}_{ap} \sim N(0, I)$.

The final form of the estimation problem, after the incorporation of the quadratic constraint and the a priori information, is:

$$\text{find: } \quad x \text{ and } v_{big} \quad (14a)$$

$$\begin{aligned} \text{to minimize: } \quad J(x, v_{big}) = & \frac{1}{2} x^T [H_0 + \sum_{j=1}^{n_{vbig}} H_j(v_{big})_j]^T [H_0 + \sum_{j=1}^{n_{vbig}} H_j(v_{big})_j] x \\ & + \frac{1}{2} [R_{ap} x + z_{ap}]^T [R_{ap} x + z_{ap}] + \frac{1}{2} v_{big}^T v_{big} \end{aligned} \quad (14b)$$

$$\text{subject to: } \quad 0 = 1 - (Lx)^T (Lx) \quad (14c)$$

The a priori information equation should be used only in two situations. The first is when there is very accurate a priori information about some of the elements of x . The second is when one needs to enforce an additional constraint in order to make the problem observable. In this latter situation, the constraining row of R_{ap} and the corresponding row of z_{ap} will take on very large values, normally orders of magnitude larger than the corresponding rows of the H matrices. This large row acts as a soft constraint on the solution. Note, however, that one must use numerically robust square-root-based techniques, as are developed in Sections III and IV, if one wants to mitigate the potential for adverse affects from the resulting wide variations in the sizes of different problem matrices.

One must be careful to apply an extra linear constraint in the form of an a priori information equation only where appropriate. One might think of trying to substitute a linear overall scaling constraint for the quadratic scaling constraint in eq. (14c). This may not be a good idea because a linear scaling constraint may bias the relative scalings between the different axes towards equality even when the true scalings are not equal.

An appropriate situation for adding a linear constraint occurs when one is also estimating the moment-of-inertia matrix of a rotating flexible appendage. Suppose that the appendage's center of mass lies on the rotation axis, that $[q_1, q_2, q_3]$ is an orthonormal triad in the main spacecraft's coordinate system, and that q_3 is directed along the appendage articulation axis. It is possible to make simultaneous unobservable changes to the moment-of-inertia matrix of the main spacecraft body I_m and the moment-of-inertia matrix of the appendage I_a . Given any scalar \mathbf{b} , the unobservable change adds $\mathbf{b}(q_1 q_1^T + q_2 q_2^T)$ to I_m while subtracting the same quantity from I_a . It is easy to show that these changes are unobservable because their effects on the system angular momentum exactly cancel regardless of the appendage articulation angle. The following a priori linear constraint removes this ambiguity:

$$(q_1^T I_m q_1 + q_2^T I_m q_2) - \mathbf{g}(q_1^T I_a q_1 + q_2^T I_a q_2) = 0 \quad (15)$$

where \mathbf{g} is a positive number that makes the a priori estimates of I_m and I_a satisfy this constraint.

III. ATTITUDE PARAMETER ESTIMATION ALGORITHM

A. Algorithm Overview

The solution algorithm for the problem in eqs. (14a)-(14c) exploits its bi-linear structure. For a fixed x , the cost function is a linear least-squares cost in the unconstrained variable v_{big} . Matrix factorizations can be used in an inner-loop optimization to solve for the global minimizer v_{big} as a function of x . Call this solution $v_{bigopt}(x)$. This solution can be substituted into the original cost function in eq. (14b) in order to define a new reduced-order estimation problem:

$$\text{find: } \quad x \quad (16a)$$

$$\text{to minimize: } \quad J_{ro}(x) = J[x, v_{bigopt}(x)] \quad (16b)$$

$$\text{subject to: } \quad 0 = 1 - (Lx)^T (Lx) \quad (16c)$$

This reduced-order problem is then solved numerically using the iterative Newton method in an outer optimization loop.

Newton's method applied to problem (16a)-(16c) starts with a guess x_0 that satisfies the quadratic constraint in eq. (16c). Next, it defines a nonlinear function $x_{curve}(x_0 + \mathbf{D}x)$ that is guaranteed to remain on the quadratic constraint as $\mathbf{D}x$ varies from zero and that takes on the initial value $x_{curve}(x_0) = x_0$. It uses this function to define the quadratic cost function approximation:

$$J_{newt}(\mathbf{D}x, \mathbf{D}v_{big}) \quad \cong \quad J[x_{curve}(x_0 + \mathbf{D}x), v_{bigopt}(x_0) + \mathbf{D}v_{big}]$$

$$= J[x_0, v_{bigopt}(x_0)] + g_x^T \mathbf{D}x + [\mathbf{D}x^T, \mathbf{D}v_{big}^T] \begin{bmatrix} W_{xx} & W_{xv} \\ W_{xv}^T & W_{vv} \end{bmatrix} \begin{bmatrix} \mathbf{D}x \\ \mathbf{D}v_{big} \end{bmatrix} \quad (17)$$

where the matrices W_{xx} , W_{xv} , and W_{vv} are blocks of the Hessian matrix. This approximate cost function gets minimized subject to the linearized approximation of the constraint in eq. (16c): $(Lx_0)^T L \mathbf{D}x = 0$. The solution is $[\mathbf{D}x_{opt}; \mathbf{D}v_{bigopt}]$, and $\mathbf{D}x_{opt}$ is the Newton step towards the solution of problem (16a)-(16c).

The Newton iteration finishes by performing a line search to approximately minimize the 1-dimensional cost function $J_{1D}(\mathbf{a}) = J\{x_{curve}(x_0 + \mathbf{a}\mathbf{D}x_{opt}), v_{bigopt}[x_{curve}(x_0 + \mathbf{a}\mathbf{D}x_{opt})]\}$. A step length of $\mathbf{a} = 1$ is tried in order to get the superlinear convergence of Newton's method when x_0 is near the solution, but \mathbf{a} gets decreased from 1 if necessary when far from the solution in order to guarantee a cost decrease: $J_{1D}(\mathbf{a}_{opt}) < J_{1D}(0)$. This forced decrease of the cost safe-guards the Newton algorithm so that it is guaranteed to converge at least to a local minimum. Enforcement of the cost-decrease constraint can be accomplished by an appropriate search process¹¹. The improved guess of the solution becomes the next Newton iterate, $x_1 = x_{curve}(x_0 + \mathbf{a}_{opt}\mathbf{D}x_{opt})$, and the process can be repeated until it converges to a local minimum.

It is possible, in theory, to reverse the roles of x and v_{big} in the algorithm, but this approach works poorly in practice. For a fixed guess of v_{big} , one can solve exactly for the x that globally minimizes of the cost function in eq. (14b) subject to the constraint in eq. (14c). The global minimization can be performed using an algorithm found in Ref. 12, or if $z_{ap} = 0$, by using two singular-value decompositions (SVDs) and several QR factorizations. This reversed method has been tried, but it does not work well. One problem with this approach is the difficulty of finding a reasonable initial guess of v_{big} . Seemingly reasonable values, such as $v_{big} = 0$, can produce very strange results when the measurement error standard deviations are large. Another problem with this approach is that the reduced-order cost function, $J_{ro}(v_{big}) = J[x_{opt}(v_{big}), v_{big}]$, can have strong nonlinearities that retard convergence. These strong nonlinearities may result from the fact that the inner optimization of x resembles an eigenvalue/eigenvector problem, and such problems can be very sensitive to their inputs when there are repeated eigenvalues.

B. Inner-Loop Optimization of v_{big}

The inner-loop optimization of v_{big} for x fixed at x_0 minimizes the linear least-squares cost function

$$J_{inner}(v_{big}) = \frac{1}{2} \left\{ \begin{bmatrix} e_v \\ 0 \end{bmatrix} + \begin{bmatrix} H_v \\ I \end{bmatrix} v_{big} \right\}^T \left\{ \begin{bmatrix} e_v \\ 0 \end{bmatrix} + \begin{bmatrix} H_v \\ I \end{bmatrix} v_{big} \right\} \quad (18)$$

which is equivalent to the cost in eq. (14b) without the a priori terms if e_v and H_v are

$$e_v = H_0 x_0 \quad \text{and} \quad H_v = [(H_1 x_0), (H_2 x_0), (H_3 x_0), \dots, (H_{nv_{big}} x_0)] \quad (19)$$

Although not explicitly stated, e_v and H_v depend on x_0 as shown in eq. (19).

The linear least-squares optimization problem in eq. (18) gets solved using standard QR factorization techniques¹¹. An orthogonal matrix Q_v , and a square, upper-triangular, non-singular matrix R_{vv} get computed via QR-factorization to satisfy

$$Q_v \begin{bmatrix} R_{vv} \\ 0 \end{bmatrix} = \begin{bmatrix} H_v \\ I \end{bmatrix} \quad (20)$$

and the optimal v_{big} gets computed as

$$v_{bigopt}(x_0) = - \begin{bmatrix} R_{vv}^{-1} & 0 \end{bmatrix} Q_v^T \begin{bmatrix} e_v \\ 0 \end{bmatrix} \quad (21)$$

The matrices involved in this calculation are large and sparse and have a dynamic programming structure. Section IV develops equivalent calculations that use an efficient recursion which exploits this structure.

C. Enforcement of the Curved Constraint

The purpose of the function $x_{curve}(x)$ is to generate a vector that satisfies the constraint in eq. (16c) while being close to x . It must have the property that $x_{curve}(x_0) = x_0$ if x_0 satisfies the constraint, and the difference $x_{curve}(x) - x$ must be small if x is near the constraint. These properties enable the line searches within the iterative Newton optimization procedure to exactly follow the curved constraint in eq. (16c).

A suitable function $x_{curve}(x)$ has been developed based on the singular value decomposition of L :

$$\begin{bmatrix} U_{L1} & U_{L2} \end{bmatrix} \begin{bmatrix} \mathbf{S}_L & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_{L1}^T \\ V_{L2}^T \end{bmatrix} = L \quad (22)$$

where the matrices U_{L1} , U_{L2} , V_{L1} , and V_{L2} are sets of columns of the orthogonal factors of the SVD and where \mathbf{S}_L is the positive-definite diagonal matrix that contains the non-zero singular values of L . The matrix V_{L1}^T projects x onto a subspace in which every component affects the quadratic constraint, and the matrix V_{L2}^T projects x onto the complementary subspace in which none of the components affect the constraint. Given these matrices, a suitable $x_{curve}(x)$ function is

$$x_{curve}(x) = \left(\frac{V_{L1}V_{L1}^T}{\sqrt{(Lx)^T(Lx)}} + V_{L2}V_{L2}^T \right) x \quad (23)$$

The idea of this function is to split x into its constrained and unconstrained components and then to re-scale the constrained component by $1/\sqrt{(Lx)^T(Lx)}$ in order to force satisfaction of the constraint. Given the SVD in eq. (22), it is straightforward to demonstrate that $x_{curve}(x)$ satisfies the quadratic constraint in eq. (16c), that $x_{curve}(x_0) = x_0$ when x_0 satisfies the constraint, and that $x_{curve}(x) - x$ is small when x is near the constraint.

D. Quadratic Problem and Solution for Newton Increment to x

The following quadratic approximate cost function is used for determining the Newton increment to x :

$$\begin{aligned} J_{newt}(\mathbf{D}x, \mathbf{D}v_{big}) = & \frac{1}{2} \left\{ \begin{bmatrix} e \\ v_{bigopt} \\ e_{ap} \end{bmatrix} + \begin{bmatrix} H_v \\ I \\ 0 \end{bmatrix} \mathbf{D}v_{big} + \begin{bmatrix} H_x \\ 0 \\ R_{ap} \end{bmatrix} \mathbf{D}x \right\}^T \left\{ \begin{bmatrix} e \\ v_{bigopt} \\ e_{ap} \end{bmatrix} + \begin{bmatrix} H_v \\ I \\ 0 \end{bmatrix} \mathbf{D}v_{big} + \begin{bmatrix} H_x \\ 0 \\ R_{ap} \end{bmatrix} \mathbf{D}x \right\} \\ & + \mathbf{D}x^T B \mathbf{D}v_{big} - \frac{1}{2} \mathbf{D}x^T \{ \mathbf{1} L^T L \} \mathbf{D}x \end{aligned} \quad (24)$$

where $e = H_x x_0$, $e_{ap} = R_{ap} x_0 + z_{ap}$, $H_x = H_0 + H_1(v_{bigopt})_1 + H_2(v_{bigopt})_2 + \dots + H_{nvbig}(v_{bigopt})_{nvbig}$, and $B = [(H_1^T e), (H_2^T e), \dots, (H_{nvbig}^T e)]$. Note, e is the error in the concatenated set of Euler equations, and e_{ap} is the error in the a priori information equation. Both errors are evaluated at the current solution guess $[x_0; v_{bigopt}(x_0)]$.

The quantity \mathbf{I} in eq (24) is like a Lagrange multiplier. It predicts the cost effects of the curvature in $x_{curve}(x_0 + \mathbf{D}x)$ under the conditions that x_0 satisfies the quadratic constraint in eq. (16c) and that $\mathbf{D}x$ satisfies the linearized approximation of the constraint: $a^T \mathbf{D}x = 0$ where $a = L^T L x_0$. The proper value of this multiplier depends on the actual form of the function $x_{curve}(x)$. For the function defined in eq. (23) the proper value is

$$\mathbf{I} = x_0^T V_{L1} V_{L1}^T [H_x^T e + R_{ap}^T e_{ap}] \quad (25)$$

The solution for the minimum of eq. (24) subject to the constraint $a^T \mathbf{D}x = 0$ starts with a transformation of the first cost term that enables unconstrained minimization with respect to $\mathbf{D}v_{big}$. Suppose that one performs the following QR factorization and associated transformation

$$Q_{big} \begin{bmatrix} R_{vv} & R_{vx} \\ 0 & R_{xx} \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} H_v & H_x \\ I & 0 \\ 0 & R_{ap} \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 0 \\ z_x \\ z_r \end{bmatrix} = Q_{big}^T \begin{bmatrix} e \\ v_{bigopt} \\ e_{ap} \end{bmatrix} \quad (26)$$

where Q_{big} is an orthogonal transformation, R_{vv} , R_{vx} , and R_{xx} are block matrices, with R_{vv} and R_{xx} square and upper-triangular and R_{vv} nonsingular, and z_x and z_r are vectors. The upper zero in the large vector that contains z_x and z_r is there by virtue of the fact that the cost function in eq. (24) is an expansion about an optimized value of v_{big} . The Q_{big} matrix can be used to transform each of the factors in the first cost term of eq. (24) so that the block matrices and vectors on the left-hand sides of eqs. (26) replace those on the right. After performing this transformation, the minimizing $\mathbf{D}v_{big}$ can be determined by solving the necessary condition that results from setting the partial derivative of $J_{newt}(\mathbf{D}x, \mathbf{D}v_{big})$ with respect to $\mathbf{D}v_{big}$ equal to zero. The result is

$$\mathbf{D}v_{big} = - R_{vv}^{-1} [R_{vx} + (B R_{vv}^{-1})^T] \mathbf{D}x \quad (27)$$

Note that the matrices in eqs. (26) and (27) are large, sparse, and structured. Section IV presents techniques that exploit the structure of these matrices in order to carry out the calculations of eqs. (26) and (27) in an efficient recursive manner.

The optimal solution for $\mathbf{D}v_{big}$ can be used to eliminate it from the transformed version of the cost function in eq. (24). This leaves a linearly constrained reduced-order quadratic optimization problem whose solution is the Newton search direction:

$$\text{find: } \mathbf{D}x \quad (28a)$$

$$\text{to minimize: } J_{newtro}(\mathbf{D}x) = \frac{1}{2} \mathbf{D}x^T \{ R_{xx}^T R_{xx} - (BR_{vv}^{-1})(BR_{vv}^{-1})^T - (BR_{vv}^{-1})R_{vx} - R_{vx}^T (BR_{vv}^{-1})^T - \mathbf{I}L^T L \} \mathbf{D}x \\ + (z_x^T R_{xx}) \mathbf{D}x + \frac{1}{2} (z_x^T z_x + z_r^T z_r) \quad (28b)$$

$$\text{subject to: } 0 = a^T \mathbf{D}x = (L^T L x_0)^T \mathbf{D}x \quad (28c)$$

The problem in eqs. (28a)-(28c) can be solved using QR factorizations and Cholesky factorizations. The first QR factorization is used to determine the null space of the constraint:

$$[q_{a1} \ Q_{a2}] \begin{bmatrix} r_a \\ 0 \end{bmatrix} = a \quad (29)$$

where q_{a1} is the first column of the resulting orthogonal matrix, Q_{a2} constitutes the last n_x-1 columns, and r_a is a non-zero scalar. This factorization can be used to transform the perturbation vector $\mathbf{D}x$ into a constrained component, $\mathbf{D}x_c$, and an unconstrained component, $\mathbf{D}x_u$:

$$\begin{bmatrix} \mathbf{D}x_c \\ \mathbf{D}x_u \end{bmatrix} = \begin{bmatrix} q_{a1}^T \\ Q_{a2}^T \end{bmatrix} \mathbf{D}x \quad (30)$$

One next uses Q_{a2} to project the square-root weighting matrix R_{xx} onto the unconstrained $\mathbf{D}x_u$ space and then one QR-factorizes the result to retrieve a square-upper triangular matrix:

$$Q_u \begin{bmatrix} R_{uu} \\ 0 \end{bmatrix} = R_{xx} Q_{a2} \quad (31)$$

where Q_u is an orthogonal matrix and R_{uu} is a square, upper-triangular, nonsingular matrix of dimension n_x-1 .

Next, one defines the change of coordinates $\mathbf{D}z_u = R_{uu} \mathbf{D}x_u$. The following unconstrained $\mathbf{D}z_u$ optimization problem is then equivalent to the estimation problem in eqs. (28a)-(28c):

$$\text{find: } \mathbf{D}z_u \quad (32a)$$

$$\text{to minimize: } J_{newtz}(\mathbf{D}z_u) = \frac{1}{2} \mathbf{D}z_u^T \{ \mathbf{r}I - \mathbf{D}W_{zz} \} \mathbf{D}z_u + g_z^T \mathbf{D}z_u + \frac{1}{2} (z_x^T z_x + z_r^T z_r) \quad (32b)$$

where

$$\mathbf{D}W_{zz} = (R_{uu}^{-1})^T Q_{a2}^T \{ (BR_{vv}^{-1})(BR_{vv}^{-1})^T + (BR_{vv}^{-1})R_{vx} + R_{vx}^T (BR_{vv}^{-1})^T + \mathbf{I}L^T L \} Q_{a2} R_{uu}^{-1} \quad (33a)$$

$$g_z = (R_{uu}^{-1})^T Q_{a2}^T R_{xx}^T z_x \quad (33b)$$

and where the positive scalar \mathbf{r} is normally set equal to 1. \mathbf{r} will get increased above 1 if necessary in order to ensure that the cost Hessian matrix $\mathbf{r}I - \mathbf{D}W_{zz}$ is positive definite. This modification option is part of the guarding technique that ensures the convergence of Newton's method to a local minimum. Its effect on $\mathbf{D}x$ guarantees that there exists a positive \mathbf{a} which yields a decrease of the line-search cost function $J_{ro}[x_{curve}(x_0 + \mathbf{a}\mathbf{D}x)]$.

The Cholesky factorization of the Hessian produces R_{zz} such that

$$R_{zz}^T R_{zz} = \mathbf{r}I - \mathbf{D}W_{zz} \quad (34)$$

where R_{zz} is a square, upper-triangular, nonsingular matrix of dimension n_x-1 . The Cholesky factorization process can be used to monitor the positive definiteness of the Hessian. One begins using the value $\mathbf{r} = 1$. If the Cholesky factorization process fails because the matrix on the left-hand side of eq. (34) is not positive definite, then \mathbf{r} gets increased, and the Cholesky factorization gets re-evaluated. A simple geometric progression of the form $\mathbf{r}_{new} = 1.05\mathbf{r}$ normally yields a positive definite Hessian matrix after a few Cholesky factorizations. The matrix dimensions involved are not very large. Therefore, it is acceptable to determine a reasonable value of \mathbf{r} via this brute-force heuristic iteration. One might try a larger increase factor if one finds that too many iterations are required to determine an acceptable \mathbf{r} value.

The final problem transformation from $\mathbf{D}x_u$ to $\mathbf{D}z_u$ is motivated by numerical considerations. One should avoid the squaring of matrices because squaring amplifies the adverse effects of poor matrix conditioning. The transformation to $\mathbf{D}z_u$ allows one to avoid squaring the matrix R_{uu} before computing the Cholesky factorization of the Hessian. For a small residuals estimation problem, i.e., for small e , the magnitude of $\mathbf{D}W_{zz}$ will be small. In this case, $\mathbf{r} = 1$ will yield a positive definite, well-conditioned Hessian matrix for the $\mathbf{D}z_u$ problem, and the only squaring that occurs, the squaring in eq. (33a), will have no adverse effects on the solution.

The matrix factorizations and problem transformations in eqs. (29)-(34) can be used to solve the original problem in eqs. (28a)-(28c). The solution is:

$$\mathbf{D}x = -Q_{a2}R_{uu}^{-1}R_{zz}^{-1}(R_{zz}^{-1})^T g_z \quad (35)$$

These matrix factorizations also provide a means of calculating the estimation error covariance matrix:

$$P_{xx} = Q_{a2}R_{uu}^{-1}R_{zz}^{-1}(Q_{a2}R_{uu}^{-1}R_{zz}^{-1})^T \quad (36)$$

In reality, this is only an approximation of the Cramer-Rao lower bound of the estimation error covariance. This covariance matrix is only valid at the termination of the optimization process, when x_0 is the final solution to the estimation problem and when $\mathbf{r} = 1$. Note that P_{xx} is singular. All of its eigenvalues are positive except for one that is zero. It corresponds to the eigenvector in the direction $a = L^T L x_0$, which is the normal to the quadratic constraint in eq. (16c). It makes sense for P_{xx} to have a zero eigenvalue in this direction because, according to the constraint, this component of x is known exactly.

E. Warning about Total Least Squares Problems

It is well known that total least squares problem can fail to have sensible solutions⁸. If the attitude parameter estimation problem in eqs. (14a) - (14c) does not have a sensible solution, then this fact will manifest itself in a failure of the guarded Newton method to converge. It will continue to decrease its cost at every iteration, but the usual algorithm termination criterion of $\|\mathbf{D}x\| \rightarrow 0$ will not be achieved. If the algorithm fails to converge even after hundreds of iterations, then one should consider whether the problem is well posed. It is often possible to transform an ill-posed problem into a well-posed problem by adding a priori information through R_{ap} and z_{ap} or by decreasing the measurement error standard deviations \mathbf{s}_w , \mathbf{s}_{hw} , \mathbf{s}_b , and \mathbf{s}_m .

IV. RECURSIVE QR MATRIX FACTORIZATIONS AND INVERSIONS

The QR factorizations in eqs. (20) and (26) and the inversions of R_{vv} in eqs. (21) and (27) can be carried out in an efficient recursive manner. The following analysis concentrates on recursive calculations for the large matrix operations in eqs. (26) and (27). The methods for eqs. (20) and (21) are almost identical and are discussed only in brief at the end of this section.

Suppose that one interchanges the order of the rows on the right-hand sides of the two equations in eq. (26). Then the error equations whose squares form the first term on the right-hand side of eq. (24) can be rewritten in the form:

$$\begin{bmatrix} I & 0 & 0 & \cdots & 0 & 0 & 0 \\ -D_{v0} & E_{v1} & 0 & \cdots & 0 & 0 & F_0 \\ 0 & I & 0 & \cdots & 0 & 0 & 0 \\ 0 & -D_{v1} & E_{v2} & \cdots & 0 & 0 & F_1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -D_{vN-1} & E_{vN} & F_{N-1} \\ 0 & 0 & 0 & \cdots & 0 & I & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & R_{ap} \end{bmatrix} \begin{bmatrix} \mathbf{D}v_0 \\ \mathbf{D}v_1 \\ \mathbf{D}v_2 \\ \vdots \\ \mathbf{D}v_{N-1} \\ \mathbf{D}v_N \\ \mathbf{D}x \end{bmatrix} + \begin{bmatrix} v_0 \\ e_0 \\ v_1 \\ e_1 \\ \vdots \\ e_{N-1} \\ v_N \\ e_{ap} \end{bmatrix} = 0 \quad (37)$$

where, based on eq. (6) and associated definitions,

$$\mathbf{D}_{vk} = [D_{1k}x_0, D_{2k}x_0, \dots, D_{m_k k}x_0] \text{ for } k = 0, \dots, N-1 \quad (38a)$$

$$E_{vk} = [E_{1k}x_0, E_{2k}x_0, \dots, E_{m_k k}x_0] \text{ for } k = 1, \dots, N \quad (38b)$$

$$F_k = [E_{0k+1} + \sum_{i=1}^{n_y} (v_{k+1})_i E_{ik+1}] - [D_{0k} + \sum_{i=1}^{n_y} (v_k)_i D_{ik}] \text{ and } e_k = F_k x_0 \text{ for } k = 0, \dots, N-1 \quad (38c)$$

An iterative QR factorization is then used to orthogonally transform eq. (37) into the following form:

$$\begin{bmatrix} R_{00} & R_{01} & 0 & \cdots & 0 & R_{0x} \\ 0 & R_{11} & R_{12} & \cdots & 0 & R_{1x} \\ 0 & 0 & R_{22} & \cdots & 0 & R_{2x} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & R_{NN} & R_{Nx} \\ 0 & 0 & 0 & \cdots & 0 & R_{xx} \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{D}v_0 \\ \mathbf{D}v_1 \\ \mathbf{D}v_2 \\ \vdots \\ \mathbf{D}v_{N-1} \\ \mathbf{D}v_N \\ \mathbf{D}x \end{bmatrix} + \begin{bmatrix} z_0 \\ z_1 \\ z_2 \\ \vdots \\ z_N \\ z_x \\ z_r \end{bmatrix} = 0 \quad (39)$$

Each R_{kk} , R_{kk+1} , and R_{kx} is a matrix block, with R_{kk} square, upper-triangular, and nonsingular, and each z_k is a vector. The matrix R_{xx} and the vectors z_x and z_r are exactly as defined in eq. (26). The upper right-hand blocks involving R_{kk} for $k = 0, \dots, N$ and R_{kk+1} for $k = 0, \dots, N-1$ constitute the R_{vv} matrix of eq. (26). The far-left blocks involving R_{kx} for $k = 0, \dots, N-1$ constitute the R_{vx} matrix of eq. (26). In accordance with eq. (26), the vectors z_0, \dots, z_N will be zero because eq. (26) is a linearization about optimized v_k values.

The transformation from eq. (37) to eq. (39) proceeds as follows. It temporarily stores intermediate results in three matrices, R_{t1k} , R_{t2k} , and R_{t2k+1} , and in two vectors, z_{t1k} and z_{t2k} . At $k = 0$ these temporary matrices and vectors get initialized to be empty arrays. Their values at later stages get computed by the factorization process. The transformation starts by recursively performing the following QR-factorization and transformation for stages $k = 0, \dots, N-1$:

$$Q_k \begin{bmatrix} R_{kk} & R_{kk+1} & R_{kx} \\ 0 & R_{t1k+1} & R_{t2k+1} \\ 0 & 0 & R_{t2k+1} \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} R_{t11k} & 0 & R_{t12k} \\ 0 & 0 & R_{t22k} \\ I & 0 & 0 \\ -D_{vk} & E_{v_{k+1}} & F_k \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} z_k \\ z_{t1k+1} \\ z_{t2k+1} \\ z_{rk} \end{bmatrix} = Q_k^T \begin{bmatrix} z_{t1k} \\ z_{t2k} \\ v_k \\ e_k \end{bmatrix} \quad (40)$$

At stage N the transformation takes the modified form.

$$Q_N \begin{bmatrix} R_{NN} & R_{Nx} \\ 0 & R_{xx} \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} R_{t11N} & R_{t12N} \\ 0 & R_{t22N} \\ I & 0 \\ 0 & R_{ap} \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} z_N \\ z_x \\ z_{rN} \end{bmatrix} = Q_N^T \begin{bmatrix} z_{t1N} \\ z_{t2N} \\ v_N \\ e_{ap} \end{bmatrix} \quad (41)$$

The vectors z_{r0}, \dots, z_{rN} are components of the large residual error vector z_r from eq. (26).

Given transformed eq. (39), the two multiplications by R_{vv}^{-1} in eq. (27) can be performed as follows. Suppose that the large matrix B in eq. (27) is broken down into blocks: $B = [B_0, B_1, \dots, B_N]$ and suppose that the matrix BR_{vv}^{-1} is also broken down into blocks: $BR_{vv}^{-1} = [C_0, C_1, \dots, C_N]$. Then the blocks of this latter matrix can be calculated by using the following recursion:

$$C_0 = B_0 R_{00}^{-1} \quad \text{and} \quad C_k = [B_k - C_{k-1} R_{k-1k}] R_{kk}^{-1} \quad \text{for } k = 1, \dots, N \quad (42)$$

Next, define the vector sequence $\mathbf{D}z_{vk} = (R_{kx} + C_k^T) \mathbf{D}x$ for $k = 0, \dots, N$. This vector sequence is used in the following backwards recursion in order to complete the evaluation of eq. (27):

$$\mathbf{D}v_N = -R_{NN}^{-1} \mathbf{D}z_{vN} \quad \text{and} \quad \mathbf{D}v_k = -R_{kk}^{-1} [\mathbf{D}z_{vk} + R_{kk+1} \mathbf{D}v_{k+1}] \quad \text{for } k = N-1, N-2, N-3, \dots, 0 \quad (43)$$

Note that the C_k values from eq. (42) can also be used to develop efficient summations that compute the terms involving BR_{vv}^{-1} in eq. (33a)'s formula for $\mathbf{D}W_{zz}$.

The evaluation of eqs. (20) and (21) proceeds in a similar manner. There is no need to include the columns associated with x in the equivalents of eqs. (37), (39), (40), and (41). The v_k vectors in the equivalents of eq. (37), (40), and (41) are all set to zero, and the vectors $e_k = (E_{0k+1} - D_{0k})x_0$ in these equations because they are linearized about $v_k = v_{k+1} = 0$. This causes the vectors z_0, \dots, z_N in the equivalent of eq. (39) to be non-zero. They replace the $\mathbf{D}z_{vk}$ vectors in the equivalent of eq. (43), and the outputs of that equation's recursion are $v_{0opt}(x_0), v_{1opt}(x_0), \dots, v_{Nopt}(x_0)$.

V. RESULTS USING DATA FROM THE MAP SPACECRAFT

A. Overview of Spacecraft and Data Sets

The new parameter estimation algorithm has been tested using flight data from the MAP spacecraft. MAP was launched on June 30, 2001 and reached its final orbit near the L2 Lagrange point of the Sun/Earth-Moon system on Oct. 1, 2001 after using gravity assists from the Moon¹³. The spacecraft carries two star trackers and a 3-axis rate-gyro package for attitude sensing, and it uses three nearly orthogonal reaction wheels to control attitude. Dynamics parameters have been estimated using data that was collected on July 1 and 2, 2001. Gravity-gradient and magnetic torques have been neglected because the spacecraft was far enough from the Earth during this time to render G_{sc} and b_{sc} negligible.

The estimation problem has been defined to estimate an 18-element x vector. It includes the 6 independent elements of the moment-of-inertia matrix, I_m , the 9 elements of the reaction wheel scale-factor/alignment matrix, C_{rw} , and the 3 elements of the reaction wheel angular momentum bias vector, h_{bias} . The quadratic constraint in eq. (14c) is defined to constrain the sum of the norms squared of the 3 columns of C_{rw} to equal 3.

The white-noise intensity of the unmodeled torque and the measurement error standard deviations that have been used in the estimator are $q_{Dh} = 2.3 \times 10^{-11}$ N²-m²-sec, $s_w = 2.5 \times 10^{-6}$ rad/sec, and $s_{hw} = 0.0244$ N-m-sec. Note that s_{hw} is relatively large, on the order of 2.5% of the total system angular momentum, which was about 1 N-m-sec during the time when flight data was collected. The limited resolution of h_{wmeask} provided the most significant noise source in the estimation problem. Its largeness stems from the limited number of bits that could be used to telemeter reaction wheel data to MAP's ground station. It is believed that the actual flight hardware keeps track of h_{wmeask} using a much higher precision.

Three different data sets have been used to do estimation. The first is about 6 hours long and starts at 15:19 UT on July 1, 2001. It includes 10 large-angle slew maneuvers that are controlled by the reaction wheels. The second data set is about 5 hours long and starts at 12:57 UT on July 2, 2001. It also includes 10 slew maneuvers that are similar in magnitude, but they are more abrupt. The third data set is only 1.6 hours long and starts at 19:22 UT on July 2, 2001. It operates in the MAP sky scanning mode. It rotates about its z axis at a nominal rate of 0.047 rad/sec (spin period = 134 sec). Super-imposed on the spin is a nutation that has a body-axis period of 129 sec and a coning half angle which ranges from 17 deg to 28 deg. This motion is controlled by the reaction wheels. They operate at a nominal angular momentum magnitude of $\|h_{wmeask}\| = 29.3$ N-m-sec in order to produce this motion, and the range of the dynamic variations of the wheel angular momentum for each axis is no more than 0.34 N-m-sec. Thus, this last data set lacks significant variability of its wheel speeds and its body-axis angular rates. The nominal sample interval for all data sets is $\Delta t_k = 10$ seconds. Data is available at 1 Hz, but a 0.1 Hz sampling rate has been used in the interests of conserving computation time.

B. Representative Results

Consider the results of a typical estimation case. This case uses all three data sets simultaneously in its estimation. Different data sets can be concatenated by zeroing out the D_{0k} , D_{ik} , E_{0k+1} , and E_{ik+1} matrices for the stage that falls on the boundary between two data sets. The estimation problem does not include any a priori information in the form of R_{ap} and z_{ap} . The algorithm converged to a well defined minimum of the least-squares cost function in about a dozen Newton iterations starting from the pre-flight estimates of the dynamic parameters.

The algorithm is able to improve its attitude dynamics parameter estimates significantly. Although there are no known truth values against which to measure success, there are several metrics that point to a significant improvement. One indication is that the final parameter estimates' least-squares cost is smaller than the initial cost by a factor of 9 even though the initial cost was calculated after partial optimization of the measurement error estimates in v_{big} . This large difference in cost occurs mostly in the third data set.

Consider the modeling errors shown in Fig. 1. The top graph plots norm time histories for the estimated wheel angular momentum measurement error $\|Dh_{wk}\|$, and the bottom graph plots norm time histories for the unmodeled disturbance torque $\|Dh_{in}/Dt_k\|$. The solid grey curves on the two graphs correspond to the nominal pre-launch dynamic parameter estimates, and the dotted black curves correspond to the new algorithm's best estimates of the parameters. Both sets of curves use optimal estimates of the measurement errors. The new estimates of the dynamic parameters yield measurement errors and unmodeled torques that, on average, are about 7 to 7.5 times smaller than those associated with the pre-flight parameters. Thus, the new parameter estimates model the MAP spacecraft's dynamics much better than do the pre-flight parameters.

The differences between the new parameter estimates and the pre-flight values are statistically significant. The maximum difference between the old and new estimates of the elements of I_m is 7.3% of the maximum principal inertia, but the maximum calculated estimation error standard deviation for an I_m element is only 0.28% of the maximum principal inertia.

Similarly, the maximum change in an element of the C_{rw} matrix is 0.035 while the maximum standard deviation for the estimation error of an element of C_{rw} is 0.0026. The differences between the pre-flight parameter estimates and the new estimates are small enough to seem reasonable, given the usual methods by which pre-flight estimates are determined, yet they are significant if one needs a very accurate model of the attitude dynamics.

A more comprehensive method of assessing the statistical significance of the differences between the pre-flight and optimal parameters is to calculate the scalar test statistic $(x_{opt}-x_{preflight})^T Q_{a2} R_{uu}^{-1} R_{zz}^{-1} R_{zz} R_{uu} Q_{a2}^T (x_{opt}-x_{preflight})$. This is the difference squared of the estimates normalized by the inverse of the estimation error covariance matrix as projected into the null space of the quadratic constraint in eq. (14c). This statistic will be a sample from a χ^2 distribution of degree n_x-1 if the differences between the two estimates are merely the result of random errors. The value of this statistic is 1.74×10^5 for this case. Given that $n_x-1 = 17$, there is virtually zero probability that these parameter differences are the result of random noise. Put differently, the average difference between the pre-flight parameter estimates and the optimal estimates is $\sqrt{1.74 \times 10^5 / 17} = 101$ standard deviations, which is very significant.

One might question how differences of I_m and C_{rw} that are on the order of 7% or less can produce the factor-of-7 differences in the error magnitudes shown on Fig. 1. The answer has to do with how MAP was being controlled during the third data set. The magnitude of the actual wheel angular momentum is 27 N-m-sec, and the magnitude of the rotational angular momentum of the main spacecraft body is 26 N-m-sec, but the total spacecraft angular momentum is only 1 N-m-sec. Thus, the large wheel angular momentum and the large rotational angular momentum almost cancel each other out to yield a total angular momentum that is less than 4% of either component's angular momentum. This cancellation has the potential to amplify the significance of parameter errors by a factor of 26 to 27, which is why a 7% change in an element of I_m can have such a large impact on the unmodeled torque magnitudes.

C. Prediction Capability

Another case has been run to more clearly illustrate the usefulness of the algorithm's parameter estimates for purposes of predicting the dynamic response of the MAP spacecraft. In this test, only the first 2 data sets have been used to estimate parameters, and the resulting parameter values have been used to estimate unmodeled torque errors and measurement errors for the third data set without further correction of the parameter estimates. The resulting wheel angular momentum errors and unmodeled torques are shown in Fig. 2. Note that the axis scales of Fig. 2 are the same as in Fig. 1. It is obvious from a comparison of the two figures that the parameter estimates from the first 2 data sets do a much better job of predicting the MAP attitude dynamics than do the pre-flight parameter estimates -- compare the solid grey curves on Fig. 1 with the curves on Fig. 2. The model errors when using the parameter estimates from the first two data sets are between 4.5 and 5 times smaller, on average, than the model errors when using the pre-flight parameters. Also obvious is the fact that the parameter estimates which use all three data sets do a slightly better job of modeling the attitude dynamics during the 3rd data set -- compare the dotted black curves on Fig. 1 with the curves on Fig. 2. This makes sense because the optimal parameter estimates from Fig. 1 interpolate into the third data set.

D. Degradation of Estimation Error with Limited Data

Another case has been run in an attempt to estimate parameters based only on the third data set. Recall that the third data set does not have a rich dynamic response. It just nutates with reaction wheel speeds and an angular velocity vector that are almost constant. If the only scaling information added to the problem is the quadratic constraint, then the estimation algorithm takes many iterations to converge, 117, and its estimates are nonsensical. The reaction wheel scale factors differ from the pre-launch estimates by as much as 47%, and the I_m estimate has two negative eigenvalues!

An attempt has been made to rescue the situation by adding a priori information. The R_{ap} and z_{ap} values have been set up to indicate that the a priori reaction wheel scale factors are accurate to within 0.2% 1- σ . This estimation run yields more reasonable estimation results. All of the eigenvalues of the I_m estimate are positive, and the estimates of the elements of C_{rw} differ from their pre-launch values by no more than 0.086. Nevertheless, the overall estimated parameter vector is not very accurate. For example, the minimum eigenvalue of the I_m estimate is only 10% of the pre-flight value. It is unrealistic to suppose that the pre-flight estimate would be in error by such a large amount.

The calculated estimation error standard deviations are very large even when the algorithm thinks that it has been given highly accurate a priori estimates for the scalings of the C_{rw} columns. The best estimation error standard deviation is 2 times larger than the corresponding standard deviation for the case that uses all 3 data sets. The worst one is almost 5000 times larger than its counterpart for the 3-data-set case. This is particularly telling when one considers that the case with 3 data sets did not even include a priori information beyond the quadratic constraint. Thus, it is critically important to use data with rich dynamic

variations in the control inputs and the attitude response if one wants to get good parameter estimates from this paper's algorithm.

E. Miscellaneous Results Discussion

A test has been made of what happens when one tries to pose the attitude estimation problem as a regular least-squares problem with a quadratic constraint. One might be tempted to model the problem in this way because the global solution can be determined simply by performing two SVDs and several QR factorizations. What one sacrifices is model fidelity. One must eliminate the multiplicative errors from the model, which eliminates the errors in w_k and h_{wk} . The difficulty with this approach is that these are the problem's most significant error sources. If one eliminates them, then the problem model becomes dubious. Although the problem is easy to solve, it is obvious that the resulting estimates contain significant errors. The minimum eigenvalue of the I_m estimate is only 22% of the pre-flight estimate, and the columns of the C_{rw} estimate differ in alignment from their pre-flight values by 20 to 27 deg. This failure highlights the importance of proper error modeling in parameter estimation problems.

As mentioned already, the dominant errors in the MAP attitude dynamics data are those in the reaction wheel telemetry stream, h_{wmeask} . This fact suggests two ideas for future consideration. The first is that it would be good for attitude estimation and control engineers to ask command and data handling engineers to deliver more accuracy in the h_{wmeask} telemetry stream on future missions. The current level of accuracy may suffice for normal "housekeeping" purposes, but increased accuracy could be critical to any attempt to implement back-up mode attitude determination or control functions in the event of a hardware failure.

The second idea has to do with the design of attitude estimation algorithms based on Euler's equations. This paper's parameter estimation algorithm obtains good results when it estimates corrections to the telemetered reaction wheel angular momentum time histories. This suggests that one should include estimation of reaction wheel corrections in an Euler-based attitude and rate estimation Kalman filter if the reaction wheel data is of low accuracy. Such an augmentation may make the difference between success and failure of the attitude filter.

VI. CONCLUSIONS

A new algorithm has been developed for estimating the parameters of a spacecraft's attitude dynamics model by using telemetered attitude and rate measurements. The estimated quantities include the moment-of-inertia matrix and, if present, alignments, scale factors, and biases for reaction wheels and magnetic torque rods. The input data includes 3-axis attitude from star trackers, 3-axis rate from rate gyros, nominal reaction wheel angular momenta, nominal torque rod dipole moment strengths, and magnetic field as measured by a magnetometer.

The algorithm uses a trapezoidally integrated version of Euler's equation in inertial coordinates as its basic estimation equation. The resulting estimation problem includes additive disturbance torque errors and multiplicative sensor measurement errors and is in the general form known as a total least-squares problem. A scalar quadratic constraint is added to the least-squares problem statement in order to make it well posed by constraining the overall scaling of the parameter estimates.

The algorithm uses inner and outer least-squares optimizations to estimate the dynamic model parameters. The inner optimization uses linear least-squares techniques to compute estimates of the sensor measurement errors for given values of the model parameters. The outer optimization uses a guarded Newton iterative numerical procedure to estimate the most likely model parameters.

The algorithm has been tested using data from the MAP spacecraft. The algorithm's best estimates of the moment-of-inertia matrix and the reaction wheel scale-factor/alignment matrix differ from the pre-flight estimates by 7% or less, but they can reduce the level of torque modeling error by a factor of 5 to 7 in certain modes of operation where spacecraft angular momentum and reaction wheel angular momentum almost cancel each other.

REFERENCES

1. Psiaki, M.L., Martel, F., and Pal, P.K., "Three-Axis Attitude Determination via Kalman Filtering of Magnetometer Data", *Journal of Guidance, Control, and Dynamics*, Vol. 13, No. 3, 1990, pp. 506-514.
2. Psiaki, M.L., Theiler, J., Bloch, J., Ryan, S., Dill, R.W., and Warner, R.E. "ALEXIS Spacecraft Attitude Reconstruction with Thermal/Flexible Motions Due to Launch Damage", *Journal of Guidance, Control, and Dynamics*, Vol. 20, No. 5, 1997, pp. 1033-1041.

3. Challa, M., Kotaru, S., and Natanson, G., "Magnetometer-Only Attitude and Rate Estimates During the Earth Radiation Budget Satellite 1987 Control Anomaly," *Proceedings of the AIAA Guidance, Navigation, and Control Conf.*, Vol. 2, American Institute of Aeronautics and Astronautics, Reston, VA, 1997, pp. 830-840.
4. M.L. Psiaki, E.M. Klatt, P.M. Kintner, Jr., and S.P. Powell, "Attitude Estimation for a Flexible Spacecraft in an Unstable Spin," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 1, 2002, pp. 88-95.
5. Challa, M., Natanson, G., and Ottenstein, N., "Magnetometer-Only Attitude and Rates For Spinning Spacecraft", *Proceedings of the AIAA/AAS Astrodynamics Specialists Conf.*, American Institute of Aeronautics and Astronautics, Reston, VA, 2000, pp. 311-321.
6. Psiaki, M.L., and Oshman, Y., "Spacecraft Attitude Rate Estimation From Geomagnetic Field Measurements," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 2, 2003, pp. 244-252.
7. Psiaki, M.L., "Global Magnetometer-Based Spacecraft Attitude and Rate Estimation," AIAA Paper No. 2003-5561, *Proceedings of the AIAA Guidance, Navigation, and Control Conf.*, American Institute of Aeronautics and Astronautics, Reston, VA, 2003, pp. unnumbered.
8. Golub, G.H., and Van Loan, C.F., "An Analysis of the Total Least Squares Problem," *SIAM Journal on Numerical Analysis*, Vol. 17, No. 6, 1980, pp. 883-893.
9. Van Huffel, S., and Vandewalle, J., "Analysis and Properties of the Generalized Total Least Squares Problem $AX \approx B$ when some or all Columns in A are Subject to Error," *SIAM Journal on Matrix Analysis and Applications*, Vol. 10, No. 3, 1989, pp. 294-315.
10. Bierman, G.J., *Factorization Methods for Discrete Sequential Estimation*, Academic Press, New York, 1977, pp. 69-76, 115-122.
11. Gill, P.E., Murray, W., and Wright, M.H., *Practical Optimization*, Academic Press, (New York, 1981), pp. 37-40, 88-92.
12. Psiaki, M.L., "Attitude-Determination Filtering via Extended Quaternion Estimation," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 2, 2000, pp. 206-214.
13. Bar-Itzhack, I.Y., and Harman, R.R., "The effect of Sensor Failure on the Attitude and Rate Estimation of the MAP Spacecraft," AIAA Paper No. 2003-5485, *Proceedings of the AIAA Guidance, Navigation, and Control Conf.*, American Institute of Aeronautics and Astronautics, Reston, VA, 2003, pp. unnumbered.

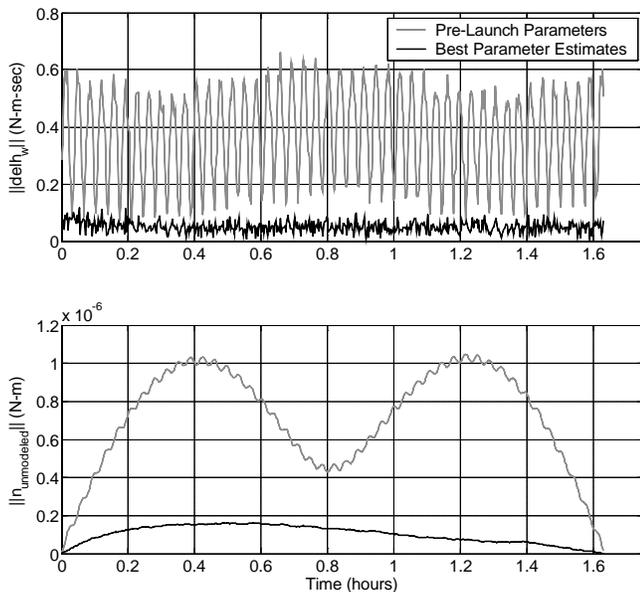


Fig. 1. Comparison of wheel angular momentum error magnitudes (top plot) and unmodeled disturbance torque magnitudes (bottom plot) for two different parameter estimates, 3rd data set.

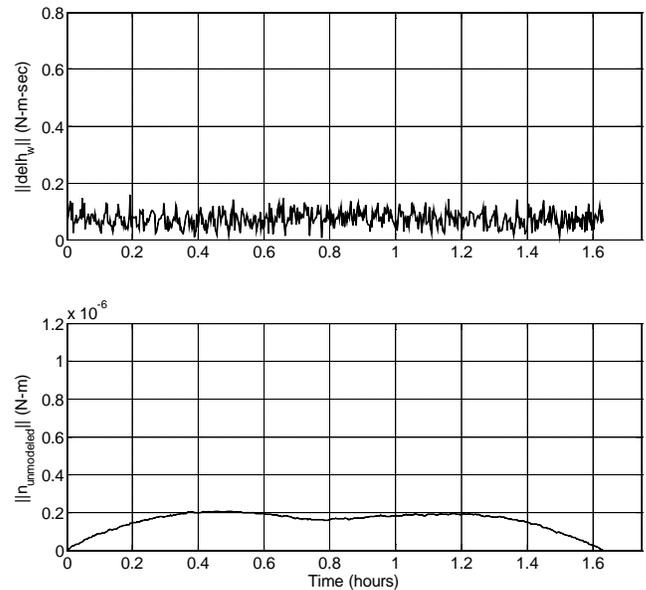


Fig. 2. Wheel angular momentum error magnitudes (top plot) and unmodeled disturbance torque magnitudes (bottom plot) for the 3rd data set when using parameter estimates based on the first 2 data sets.