JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 34, No. 2, March–April 2011

Adaptive Nonlinear Huber-Based Navigation for Rendezvous in Elliptical Orbit

Christopher D. Karlgaard*
Analytical Mechanics Associates, Inc., Hampton, Virginia 23666
and
Hanspeter Schaub†
University of Colorado at Boulder, Boulder, Colorado 80309

DOI: 10.2514/1.51939

This paper discusses the development of an adaptive discrete-time robust nonlinear filtering technique, and the application of this technique to the six-degree-of-freedom elliptical orbit rendezvous and docking problem. The proposed adaptive nonlinear filtering technique is based on a robust modification of the divided-difference filtering method based on Huber's generalized maximum likelihood estimation, blended with a robust covariance matching procedure based on a modified Myers–Tapley method. In particular, the adaptive scheme can estimate the process noise and measurement noise covariance matrices along with the state estimate and state estimate error covariance matrix, which uses a robust approach to estimating these covariances that can resist the effects of outliers in the stored buffer of residuals.

I. Introduction

THE development of navigation filters that make use of these robust techniques is important due to the sensitivity of the typical minimum ℓ_2 norm techniques, such as the Kalman filter, to deviations in the assumed underlying probability distribution. In particular, those distributions with thicker tails than the Gaussian distribution can give rise to erratic filter performance and inconsistency of results. A class of filtering techniques based on Huber's generalized maximum likelihood estimation theory have been previously developed which provide robustness with respect to these deviations from the Gaussian case [1]. Additionally, if the assumed parameters of the distribution differ greatly from the true parameters, then the filter can exhibit large errors and possibly divergence in nonlinear problems [2]. This behavior is possible even if the true error distributions are Gaussian. To remedy these problems, adaptive filtering techniques have been introduced in order to automatically tune the Kalman filter by estimating the measurement and process noise covariances [3].

The Huber-based formulations of the filtering problem also make some assumptions regarding the distribution, namely the Huber approach considers a class of contaminated densities in the neighborhood of the Gaussian density. Essentially the method assumes that the statistics of the main Gaussian density are known, as well as the ratio or percentage of the contamination (although the filter makes no assumption of the nature of the contaminating density other than it be symmetric with finite variance). The technique can be improved upon by the introduction of a method to adaptively estimate the noise statistics along with the state and state error covariance matrix.

One technique in common use for adaptively estimating the noise statistics is known as *covariance matching*. The covariance matching technique is an approach in which the measurement noise and process noise covariances are determined in such a way that the true residual covariance matches the theoretically predicted covariance. The true residual covariance is approximated in real time using the sample covariance, over some finite buffer of stored residuals. An explicit solution to the covariance matching problem is proposed by Myers and Tapley in [4] which uses the sample mean and sample covariance of the stored residuals in order to estimate the measurement and process noise covariance matrices. The drawback to this approach is that the presence of outliers and non-Gaussianity can create problems of robustness with the use of sample mean and sample covariance. Therefore some additional steps must be taken to identify the outliers before forming the covariance estimates. The particular outlier identification method proposed in this paper is based on quantities known as *projection statistics* [5], which use the sample median and median absolute deviation, and as a result are highly effective technique for outlier identification. These projection statistics are employed in this paper as weights in a new, robust covariance matching procedure in order to reduce the influence of the outliers.

The hybrid robust/adaptive nonlinear filtering methods introduced in this paper are applied to the problem of six-degree-of-freedom (DOF) rendezvous navigation in elliptical orbit. The equations of relative motion are formulated in spherical coordinates centered on the target orbit. A simple control law based on feedback linearization is used to track a desired rendezvous trajectory. The attitude dynamics are parameterized using modified Rodrigues parameters (MRPs) [6], which are advantageous for both control law development and estimation since they constitute a minimal three-parameter attitude description. A switching technique which exploits the stereographic projection properties of the MRP coordinate is used to avoid singularities which inevitably arise in minimal attitude descriptions. This switching, described in [7], includes the proper covariance transformations. An attitude control law based on backstepping is employed to track the target vehicle.

A sensor suite consisting of a generic light detection and ranging (LIDAR) or optical sensor, an inertial measurement unit (IMU), a star tracker, and a generic orbit sensor are used to provide measurement data to the navigation filters so that the chaser vehicle can estimate its relative state during the rendezvous maneuver. Several filters are implemented for comparison, including the extended Kalman filter, first and second-order divided-difference filters (DD1 and DD2, respectively) and Huber-based generalizations of these filters that include adaptive techniques for estimating the noise covariances. Monte Carlo simulations are presented which include both Gaussian

Presented as Paper 2010-7665 at the AIAA/AAS Astrodynamics Specialists Conference, Toronto, 2–5 August 2010; received 9 August 2010; revision received 25 October 2010; accepted for publication 25 October 2010.

*Senior Project Engineer, 303 Butler Farm Road, Suite 104A; karlgaard@ama-inc.com. Senior Member AIAA.

†Associate Professor, Aerospace Engineering Sciences Department; hanspeter.schaub@colorado.edu. Associate Fellow AIAA.

and non-Gaussian errors, including mismatches in the assumed noise covariances in the navigation filters in order to illustrate the benefits of the robust/adaptive nonlinear filters. Additionally, computational burdens of the various filters are compared.

II. Robust Huber-Based Nonlinear Filtering

This section discusses the problem of estimating the state of the system of ordinary differential equations:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{v}, t) \tag{1}$$

where $\boldsymbol{x}$ is the state vector, $\boldsymbol{u}$ are the deterministic inputs to the system, and $\boldsymbol{v}$ are random process noise inputs to the system. The mean value of $\boldsymbol{v}$ is $\bar{\boldsymbol{v}} = \boldsymbol{0}$ and its spectral density is $\mathcal{Q}$. It is assumed that the state of the system can be measured at discrete times in the form of a model given as

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k) + \boldsymbol{w}_k \tag{2}$$

where the subscript k refers to the value of the parameter at time t_k, $\boldsymbol{y}_k$ is the measurement at time t_k, and $\boldsymbol{w}_k$ is the measurement noise at time t_k. The mean value of $\boldsymbol{w}_k$ is $\bar{\boldsymbol{w}}_k = \boldsymbol{0}$ and its covariance is $\boldsymbol{R}_k$.

A. Robust Extended Kalman Filtering

The Kalman filter is a well-known technique for estimating the state of systems of differential equations described in the form provided in Eqs. (1) and (2). In this approach, the state and covariance predictions are given as

$$\bar{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}_{k-1} + \int_{t_{k-1}}^{t_k} \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, \bar{\boldsymbol{v}}, t)\,\mathrm{d}t \tag{3}$$

$$\bar{\boldsymbol{P}}_k = \hat{\boldsymbol{P}}_{k-1} + \int_{t_{k-1}}^{t_k} [\mathbf{A}(\hat{\boldsymbol{x}}(t), t)\bar{\boldsymbol{P}}(t) + \bar{\boldsymbol{P}}(t)\mathbf{A}(\hat{\boldsymbol{x}}(t), t)^T + \mathbf{B}(\hat{\boldsymbol{x}}(t), t)\mathcal{Q}(t)\mathbf{B}(\hat{\boldsymbol{x}}(t), t)^T]\,\mathrm{d}t \tag{4}$$

where $\bar{\boldsymbol{x}}_k$ is the predicted value of the state at time t_k, based on the estimated value of the state at time t_{k-1}, which is $\hat{\boldsymbol{x}}_{k-1}$. Similarly, $\bar{\boldsymbol{P}}_k$ is the predicted state error covariance matrix at time t_k and $\hat{\boldsymbol{P}}_{k-1}$ is the estimated state error covariance matrix and time t_{k-1}. Also, the matrices $\mathbf{A}$ and $\mathbf{B}$ are given by

$$\mathbf{A}(t) = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\bar{\boldsymbol{x}}(t), \boldsymbol{v}=\bar{\boldsymbol{v}}} \tag{5}$$

$$\mathbf{B}(t) = \left.\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{v}}\right|_{\boldsymbol{x}=\bar{\boldsymbol{x}}(t), \boldsymbol{v}=\bar{\boldsymbol{v}}} \tag{6}$$

The discrete-time covariance propagation is written as $\bar{\boldsymbol{P}}_{k+1} = \boldsymbol{\Phi}_k \hat{\boldsymbol{P}}_k \boldsymbol{\Phi}_k^T + \boldsymbol{Q}_k$ where $\boldsymbol{\Phi}_k$ is the state transition matrix and $\boldsymbol{Q}_k$ is the process noise covariance matrix. Both of these quantities are determined jointly through the relation [8,9]

$$\exp\left(\begin{bmatrix} -\boldsymbol{A} & \boldsymbol{B}\mathcal{Q}\boldsymbol{B}^T \\ \boldsymbol{0} & \boldsymbol{A}^T \end{bmatrix}\delta t\right) = \begin{bmatrix} \boldsymbol{X}_{11} & \boldsymbol{X}_{12} \\ \boldsymbol{0} & \boldsymbol{X}_{22} \end{bmatrix} = \begin{bmatrix} \boldsymbol{X}_{11} & \boldsymbol{\Phi}_k^{-1}\boldsymbol{Q}_k \\ \boldsymbol{0} & \boldsymbol{\Phi}_k^T \end{bmatrix} \tag{7}$$

leading to the result $\boldsymbol{\Phi}_k = \boldsymbol{X}_{22}^T$ and $\boldsymbol{Q}_k = \boldsymbol{\Phi}_k \boldsymbol{X}_{12}$, where $\delta t = t_{k+1} - t_k$. Note that these relationships and the use of the state transition matrix are approximations that are true only when the state dynamics matrix is constant. These approximations generally work well when the process is slowly varying and/or the sampling rate is sufficiently high for a particular problem.

Equations for the state and covariance updates result in the iteratively reweighted Kalman algorithm given by [10]

$$\hat{\boldsymbol{x}}_k^{(m+1)} = \bar{\boldsymbol{x}}_k + \boldsymbol{K}_k^{(m)}[\boldsymbol{y}_k - \boldsymbol{h}(\bar{\boldsymbol{x}}_k)] \tag{8}$$

$$\hat{\boldsymbol{P}}_k = (\mathbf{I} - \boldsymbol{K}_k \boldsymbol{H}_k)\bar{\boldsymbol{P}}_k^{1/2} \boldsymbol{\Psi}_x^{-1} \bar{\boldsymbol{P}}_k^{1/2} \tag{9}$$

where m is an iteration index and

$$\boldsymbol{K}_k^{(m)} = \bar{\boldsymbol{P}}_k^{1/2} \boldsymbol{\Psi}_x^{-1} \bar{\boldsymbol{P}}_k^{1/2} \boldsymbol{H}_k^T (\boldsymbol{H}_k \bar{\boldsymbol{P}}_k^{1/2} \boldsymbol{\Psi}_x^{-1} \bar{\boldsymbol{P}}_k^{1/2} \boldsymbol{H}_k^T + \boldsymbol{R}_k^{1/2} \boldsymbol{\Psi}_y^{-1} \boldsymbol{R}_k^{1/2})^{-1} \tag{10}$$

The quantities $\boldsymbol{\Psi}_x$ and $\boldsymbol{\Psi}_y$ are diagonal matrices computed from the Huber ψ function, given by $\psi(\zeta_i) = \phi(\zeta_i)/\zeta_i$, where $\phi(\zeta_i) = \rho'(\zeta_i)$ and

$$\rho(\zeta_i) = \begin{cases} \frac{1}{2}\zeta_i^2 & \text{for } |\zeta_i| < \gamma \\ \gamma|\zeta_i| - \frac{1}{2}\gamma^2 & \text{for } |\zeta_i| \geq \gamma \end{cases} \tag{11}$$

with residual vector

$$\boldsymbol{\zeta} = \begin{bmatrix} \boldsymbol{R}_k^{-1/2} & \boldsymbol{0} \\ \boldsymbol{0} & \bar{\boldsymbol{P}}_k^{-1/2} \end{bmatrix} \cdot \begin{Bmatrix} \boldsymbol{y}_k - \bar{\boldsymbol{y}}_k \\ \hat{\boldsymbol{x}}_k^{(0)} - \bar{\boldsymbol{x}}_k \end{Bmatrix} \tag{12}$$

B. Robust Divided-Difference Filtering

The divided-difference filtering equations rely upon a discrete representation of the system dynamics, in which the differential equation in Eq. (1) is replaced with a difference equation of the form [11]

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}(\boldsymbol{x}_k, \boldsymbol{v}_k, t_k) \tag{13}$$

and the measurement equation is given by

$$\boldsymbol{y}_k = \boldsymbol{G}(\boldsymbol{x}_k, \boldsymbol{w}_k, t_k) \tag{14}$$

where $\boldsymbol{y}_k$ is the measurement at time t_k, and $\boldsymbol{w}_k$ is the measurement noise at time t_k. The mean value of $\boldsymbol{w}_k$ is $\bar{\boldsymbol{w}}_k = \boldsymbol{0}$ and its covariance is the matrix $\boldsymbol{R}_k$.

The following square-root decompositions of the predicted state error covariance, $\bar{\boldsymbol{P}}_k$, corrected state error covariance, $\hat{\boldsymbol{P}}_k$, process noise covariance, $\mathbf{Q}_k$, and measurement noise covariance, $\boldsymbol{R}_k$, are defined as

$$\hat{\boldsymbol{P}}_k = \hat{\boldsymbol{S}}_{x_k} \hat{\boldsymbol{S}}_{x_k}^T, \qquad \bar{\boldsymbol{P}}_k = \bar{\boldsymbol{S}}_{x_k} \bar{\boldsymbol{S}}_{x_k}^T$$
$$\mathbf{Q}_k = \boldsymbol{S}_{v_k} \boldsymbol{S}_{v_k}^T, \qquad \boldsymbol{R}_k = \boldsymbol{S}_{w_k} \boldsymbol{S}_{w_k}^T \tag{15}$$

Also, the jth column of $\bar{\boldsymbol{s}}_{x_k}$ is referred to as $\bar{\boldsymbol{s}}_{x_{k_j}}$; likewise for the other matrices.

Robust Huber-based divided-difference filters have been developed in [1], including both first and second-order filters. The following sections summarize the second-order filter (DD2). The results on the first-order filter (DD1) are conceptually similar to the DD2 and can be found in [1]. Note that these filters are nonadaptive in nature.

The DD2 filter makes use of both first and second-order divided differences to approximate nonlinear transformation of the state and covariance. The following matrices of first-order divided differences are defined as

$$\boldsymbol{S}'_{x\hat{x}_{k_{i,j}}} = \frac{1}{2c}[\boldsymbol{F}_i(\hat{\boldsymbol{x}} + c\hat{\boldsymbol{s}}_{x_j}, \bar{\boldsymbol{v}}_k, t_k) - \boldsymbol{F}_i(\hat{\boldsymbol{x}}_k - c\hat{\boldsymbol{s}}_{x_j}, \bar{\boldsymbol{v}}_k, t_k)] \tag{16}$$

$$\boldsymbol{S}'_{xv_{k_{i,j}}} = \frac{1}{2c}[\boldsymbol{F}_i(\hat{\boldsymbol{x}}_k, \bar{\boldsymbol{v}}_k + c\boldsymbol{s}_{v_j}, t_k) - \boldsymbol{F}_i(\hat{\boldsymbol{x}}_k, \bar{\boldsymbol{v}}_k - c\boldsymbol{s}_{v_j}, t_k)] \tag{17}$$

$$\boldsymbol{S}'_{y\bar{x}_{k_{i,j}}} = \frac{1}{2c}[\boldsymbol{G}_i(\bar{\boldsymbol{x}}_k + c\bar{\boldsymbol{s}}_{x_j}, \bar{\boldsymbol{w}}_k, t_k) - \boldsymbol{G}_i(\bar{\boldsymbol{x}}_k - c\bar{\boldsymbol{s}}_{x_j}, \bar{\boldsymbol{w}}_k, t_k)] \tag{18}$$

$$\boldsymbol{S}'_{yw_{k_{i,j}}} = \frac{1}{2c}[\boldsymbol{G}_i(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{w}}_k + c\boldsymbol{s}_{w_j}, t_k) - \boldsymbol{G}_i(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{w}}_k - c\boldsymbol{s}_{w_j}, t_k)] \tag{19}$$

where c the divided-difference perturbing parameter and (i, j) refers to the (row, column) indices of the divided-difference matrices.

The matrices of second-order divided differences are defined as

$$\boldsymbol{S}''_{x\hat{x}_{k_{i,j}}} = \frac{\sqrt{c^2-1}}{2c^2}[\boldsymbol{F}_i(\hat{\boldsymbol{x}}_k + c\hat{\boldsymbol{s}}_{x_j}, \bar{\boldsymbol{v}}_k, t_k) + \boldsymbol{F}_i(\hat{\boldsymbol{x}}_k - c\hat{\boldsymbol{s}}_{x_j}, \bar{\boldsymbol{v}}_k, t_k) - 2\boldsymbol{F}_i(\hat{\boldsymbol{x}}_k, \bar{\boldsymbol{v}}_k, t_k)] \tag{20}$$

$$\boldsymbol{S}''_{x v_{k_{i,j}}} = \frac{\sqrt{c^2-1}}{2c^2}[\boldsymbol{F}_i(\hat{\boldsymbol{x}}_k, \bar{\boldsymbol{v}}_k + c\boldsymbol{s}_{w_j}, t_k) + \boldsymbol{F}_i(\hat{\boldsymbol{x}}_k, \bar{\boldsymbol{v}}_k - c\boldsymbol{s}_{w_j}, t_k) - 2\boldsymbol{F}_i(\hat{\boldsymbol{x}}_k, \bar{\boldsymbol{v}}_k, t_k)] \tag{21}$$

$$\boldsymbol{S}''_{y\bar{x}_{k_{i,j}}} = \frac{\sqrt{c^2-1}}{2c^2}[\boldsymbol{G}_i(\bar{\boldsymbol{x}} + c\bar{\boldsymbol{s}}_{x_j}, \bar{\boldsymbol{w}}_k, t_k) + \boldsymbol{G}_i(\bar{\boldsymbol{x}} - c\bar{\boldsymbol{s}}_{x_j}, \bar{\boldsymbol{w}}_k, t_k) - 2\boldsymbol{G}_i(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{w}}_k, t_k)] \tag{22}$$

$$\boldsymbol{S}''_{y w_{k_{i,j}}} = \frac{\sqrt{c^2-1}}{2c^2}[\boldsymbol{G}_i(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{w}}_k + c\boldsymbol{s}_{w_j}, t_k) + \boldsymbol{G}_i(\bar{\boldsymbol{x}}_k \bar{\boldsymbol{w}}_k - c\boldsymbol{s}_{w_j}, t_k) - 2\boldsymbol{G}_i(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{w}}_k, t_k)] \tag{23}$$

The state, state root-covariance, measurement, and measurement covariance predictions are given by

$$\bar{\boldsymbol{x}}_{k+1} = \left(\frac{c^2 - n_x - n_v}{c^2}\right)\boldsymbol{F}(\hat{\boldsymbol{x}}_k, \bar{\boldsymbol{v}}_k, t_k) + \frac{1}{2c^2}\sum_{j=1}^{n_x}[\boldsymbol{F}(\hat{\boldsymbol{x}}_k + c\hat{\boldsymbol{s}}_{x_{k_j}}, \bar{\boldsymbol{v}}_k, t_k) + \boldsymbol{F}(\hat{\boldsymbol{x}}_k - c\hat{\boldsymbol{s}}_{x_{k_j}}, \bar{\boldsymbol{v}}_k, t_k)] + \frac{1}{2c^2}\sum_{j=1}^{n_v}[\boldsymbol{F}(\hat{\boldsymbol{x}}_k, \bar{\boldsymbol{v}}_k + c\boldsymbol{s}_{v_{k_j}}, t_k) + \boldsymbol{F}(\hat{\boldsymbol{x}}_k, \bar{\boldsymbol{v}}_k - c\boldsymbol{s}_{v_{k_j}}, t_k)] \tag{24}$$

$$\bar{\boldsymbol{S}}_{x_{k+1}} = \mathcal{H}([\,\boldsymbol{S}'_{x\hat{x}_k} \quad \boldsymbol{S}'_{xv_k} \quad \boldsymbol{S}''_{x\hat{x}_k} \quad \boldsymbol{S}''_{xv_k}\,]) \tag{25}$$

$$\bar{\boldsymbol{y}}_k = \left(\frac{c^2 - n_x - n_w}{c^2}\right)\boldsymbol{G}(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{w}}_k, t_k) + \frac{1}{2c^2}\sum_{j=1}^{n_x}[\boldsymbol{G}(\bar{\boldsymbol{x}}_k + c\bar{\boldsymbol{s}}_{x_{k_j}}, \bar{\boldsymbol{w}}_k, t_k) + \boldsymbol{G}(\bar{\boldsymbol{x}}_k - c\bar{\boldsymbol{s}}_{x_{k_j}}, \bar{\boldsymbol{w}}_k, t_k)] + \frac{1}{2c^2}\sum_{j=1}^{n_w}[\boldsymbol{G}(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{w}}_k + c\boldsymbol{s}_{w_{k_j}}, t_k) + \boldsymbol{G}(\bar{\boldsymbol{x}}_k, \bar{\boldsymbol{w}}_k - c\boldsymbol{s}_{w_{k_j}}, t_k)] \tag{26}$$

$$\boldsymbol{S}_{y_k} = \mathcal{H}([\,\boldsymbol{S}'_{y\bar{x}_k} \quad \boldsymbol{S}'_{yw_k} \quad \boldsymbol{S}''_{y\bar{x}_k} \quad \boldsymbol{S}''_{yw_k}\,]) \tag{27}$$

where n_x is the size of the state dimension, n_v is the size of the process noise dimension, and n_w is the size of the measurement noise dimension.

The state and root-covariance update equations are given by

$$\boldsymbol{S}^{(1)}_{y_k} = \mathcal{H}([\,\boldsymbol{S}'_{y\bar{x}_k}\boldsymbol{\Psi}_x^{-1/2} \quad \boldsymbol{S}'_{yw_k}\boldsymbol{\Psi}_y^{-1/2} \quad \boldsymbol{S}''_{y\bar{x}_k}\boldsymbol{\Psi}_x^{-1/2} \quad \boldsymbol{S}''_{yw_k}\boldsymbol{\Psi}_y^{-1/2}\,]) \tag{28}$$

$$\boldsymbol{K}^{(1)}_k = \bar{\boldsymbol{S}}_{x_k}\boldsymbol{\Psi}_x^{-1}\boldsymbol{S}'^{T}_{y\bar{x}_k}(\boldsymbol{S}^{(1)}_{y_k}\boldsymbol{S}^{(1)^T}_{y_k})^{-1} \tag{29}$$

$$\hat{\boldsymbol{x}}_k = \bar{\boldsymbol{x}}_k + \boldsymbol{K}^{(1)}_k(\boldsymbol{y}_k - \bar{\boldsymbol{y}}_k) \tag{30}$$

$$\hat{\boldsymbol{S}}_{x_k} = \mathcal{H}([\,\bar{\boldsymbol{S}}_{x_k}\boldsymbol{\Psi}_x^{-1/2} - \boldsymbol{K}^{(1)}_k\boldsymbol{S}'_{yx_k}\boldsymbol{\Psi}_x^{-1/2} \quad \boldsymbol{K}^{(1)}_k\boldsymbol{S}'_{yw_k}\boldsymbol{\Psi}_y^{-1/2} \quad \boldsymbol{K}^{(1)}_k\boldsymbol{S}''_{yx_k}\boldsymbol{\Psi}_x^{-1/2} \quad \boldsymbol{K}^{(1)}_k\boldsymbol{S}''_{yw_k}\boldsymbol{\Psi}_y^{-1/2}\,]) \tag{31}$$

where $\boldsymbol{\Psi}_x$ and $\boldsymbol{\Psi}_y$ are diagonal matrices computed from the Huber ψ function, with residuals that take the form

$$\boldsymbol{\zeta} = \begin{bmatrix} \boldsymbol{S}_{w_k}^{-1} & \boldsymbol{0} \\ \boldsymbol{0} & \bar{\boldsymbol{S}}_{x_k}^{-1} \end{bmatrix} \cdot \begin{Bmatrix} \boldsymbol{y}_k - \bar{\boldsymbol{y}}_k \\ \hat{\boldsymbol{x}}_k^{(0)} - \bar{\boldsymbol{x}}_k \end{Bmatrix} \tag{32}$$

where the superscript (0) refers to the initial state estimate.

III. Adaptive Tuning for Unknown Noise Statistics

A. Myers–Tapley Method

An intuitive approach to adaptive state estimation is proposed by Myers and Tapley [4]. In this approach, the measurement residual sequence is mined to produce estimates of the measurement noise statistics, and state prediction residuals are mined to compute estimates of the process noise statistics. The estimators make use of a buffer of N stored measurement noise and process noise residuals to compute the noise statistics. In batch form, the estimator for the measurement noise covariance can be derived by first defining the empirical covariance matrix

$$\boldsymbol{C}_\zeta = \frac{1}{N-1}\sum_{j=1}^{N}(\boldsymbol{\zeta}_j - \bar{\boldsymbol{\zeta}})(\boldsymbol{\zeta}_j - \bar{\boldsymbol{\zeta}})^T \tag{33}$$

where $\boldsymbol{\zeta}_j$ is the jth stored measurement residual, and $\bar{\boldsymbol{\zeta}}$ is the sample mean of the residuals:

$$\bar{\boldsymbol{\zeta}} = (1/N)\sum_{j=1}^{N}\boldsymbol{\zeta}_j$$

The expected value of $\boldsymbol{C}_\zeta$ is [4]

$$E[\boldsymbol{C}_\zeta] = \boldsymbol{R} + \frac{1}{N}\sum_{j=1}^{N}\boldsymbol{H}_j\bar{\boldsymbol{P}}_j\boldsymbol{H}_j^T \tag{34}$$

By substituting Eq. (33) into Eq. (34), an estimate for the measurement noise covariance matrix is

$$\hat{\boldsymbol{R}} = \frac{1}{N-1}\sum_{j=1}^{N}\left[(\boldsymbol{\zeta}_j - \bar{\boldsymbol{\zeta}})(\boldsymbol{\zeta}_j - \bar{\boldsymbol{\zeta}})^T - \left(\frac{N-1}{N}\right)\boldsymbol{H}_j\bar{\boldsymbol{P}}_j\boldsymbol{H}_j^T\right] \tag{35}$$

To form estimates for the process noise statistics, the process noise sample is defined as $\boldsymbol{\lambda}_j = \hat{\boldsymbol{x}}_j - \bar{\boldsymbol{x}}_j$. Then, the empirical covariance matrix for $\boldsymbol{\lambda}$ is

$$\boldsymbol{C}_\lambda = \frac{1}{N-1}\sum_{j=1}^{N}(\boldsymbol{\lambda}_j - \bar{\boldsymbol{\lambda}})(\boldsymbol{\lambda}_j - \bar{\boldsymbol{\lambda}})^T \tag{36}$$

where

$$\bar{\boldsymbol{\lambda}} = (1/N)\sum_{j=1}^{N}\boldsymbol{\lambda}_j$$

The expected value of $\boldsymbol{C}_\lambda$ is [12] $E[\boldsymbol{C}_\lambda] = \boldsymbol{K}_k\boldsymbol{H}_k\bar{\boldsymbol{P}}_k = \bar{\boldsymbol{P}}_k^\star + \boldsymbol{Q} - \hat{\boldsymbol{P}}_k$, where $\bar{\boldsymbol{P}}_k^\star$ is the propagated covariance without the process noise component, given by $\bar{\boldsymbol{P}}_k^\star = \boldsymbol{\Phi}_{k-1}\hat{\boldsymbol{P}}_{k-1}\boldsymbol{\Phi}_{k-1}^T$ where $\boldsymbol{\Phi}_{k-1}$ is the state transition matrix. It follows that an estimator for the process noise covariance matrix is [4]

$$\hat{\boldsymbol{Q}} = \frac{1}{N-1}\sum_{j=1}^{N}\left[(\boldsymbol{\lambda}_j - \bar{\boldsymbol{\lambda}})(\boldsymbol{\lambda}_j - \bar{\boldsymbol{\lambda}})^T - \left(\frac{N-1}{N}\right)(\bar{\boldsymbol{P}}_j^\star - \hat{\boldsymbol{P}}_j)\right] \tag{37}$$

An adaptive filter can function by using some initial guess of the measurement noise and process noise matrices, storing the residuals for the first N frames, and then updating the covariance estimates

based on Eqs. (35) and (37) at each subsequent frame. It is important to note that the Myers–Tapley method for adaptively estimating the measurement noise and process noise covariance matrices make use of the sample mean and covariance of the stored residuals, which are nonrobust estimators. This lack of robustness implies that the performance Myers–Tapley adaptive method can degrade in the presence of non-Gaussianity. Therefore it is of interest to develop a modification of the Myers–Tapley approach that is robust with respect to non-Gaussian distributions, which is the subject of the following section.

B. Modified Myers–Tapley Method

1. Outlier Identification

In this section, two outlier identification methods are discussed. The first method is a classical technique that is based on the weighted Euclidean norm of the separation between a possible outlier and the sample mean, known as *Mahalanobis distances*. The weighting in this method is based on the sample covariance matrix. The second method, known as *projection statistics*, is a robust approach in which the sample mean and covariance are replaced by the sample median and the median absolute deviation, respectively. The latter method is said to be robust because it is insensitive to clusters of outliers, unlike the classical method.

Given a cloud of m points in n dimensions represented by the vectors $\boldsymbol{h}_i$ for $i = 1, \cdots, m$, the Mahalanobis distances are defined as

$$\mathcal{M}_i = \sqrt{(\boldsymbol{h}_i - \bar{\boldsymbol{h}})^T \boldsymbol{C}^{-1} (\boldsymbol{h}_i - \bar{\boldsymbol{h}})} \tag{38}$$

where $\bar{\boldsymbol{h}}$ is the sample mean and $\boldsymbol{C}$ is the sample covariance matrix, given by the equations

$$\bar{\boldsymbol{h}} = \frac{1}{m} \sum_{i=1}^{m} \boldsymbol{h}_i \tag{39}$$

$$\boldsymbol{C} = \frac{1}{m-1} \sum_{i=1}^{m} (\boldsymbol{h}_i - \bar{\boldsymbol{h}})(\boldsymbol{h}_i - \bar{\boldsymbol{h}})^T \tag{40}$$

respectively.

The Mahalanobis distances can also be expressed as the solution to a maximization problem of the form [5]

$$\mathcal{M}_i = \max_{\|\boldsymbol{v}\|=1} \left[\frac{\| \boldsymbol{h}_i^T \boldsymbol{v} - \frac{1}{m} \sum_{j=1}^{m} \boldsymbol{h}_j^T \boldsymbol{v} \|}{\sqrt{\frac{1}{m-1} \sum_{k=1}^{m} (\boldsymbol{h}_k^T \boldsymbol{v} - \frac{1}{m} \sum_{j=1}^{m} \boldsymbol{h}_j^T \boldsymbol{v})^2}} \right] \tag{41}$$

The Mahalanobis distances represent the surface of an n-dimensional ellipsoid centered at the sample mean. The square of the Mahalanobis distances follow a χ^2 distribution with n degrees of freedom, assuming that the input data are Gaussian. Therefore, an outlier identification method is to consider all points satisfying

$$\mathcal{M}_i > \sqrt{\chi^2_{n,\alpha}} \tag{42}$$

to be outliers, where α is the probability that a value falls inside the ellipse (for example, $\alpha = 0.95$).

While the Mahalanobis distances are simple to conceptualize and easy to compute, the method suffers from sensitivity to clusters of outliers. This sensitivity is due to the masking effect, in which it is possible to find groups of points with nonzero errors that can sum to produce very small residuals. The masking effect in the Mahalanobis distances is related to the sensitivity of the sample mean and covariance, which are not robust estimators. In the case of clustered outliers the sample mean can be pulled toward their direction and away from the main cluster of data, which serves also to increase the size of the sample covariance. In effect, these sensitivities serve to hide or mask the cluster of outliers since their associated Mahalanobis distances will not be larger than the main group of data. See [13,14] for further information on the masking effect and sensitivity of the Mahalanobis distances.

A robust approach to the problem of outlier identification is to replace the sample mean and covariance in the equation for the Mahalanobis distances, Eq. (41), with the sample median and the median absolute deviation from the median [5]. These estimators of location and scale are known to be robust with respect to outliers and therefore it is expected that the computation of a Mahalanobis-like quantity based on these parameters will also be robust with respect to outliers. These quantities are known as *projection statistics*, and are defined as the solution to the maximization problem

$$\mathcal{P}_i = \max_{\|\boldsymbol{v}\|=1} \left[\frac{\| \boldsymbol{h}_i^T \boldsymbol{v} - \text{median}(\boldsymbol{h}_j^T \boldsymbol{v}) \|}{c \cdot \text{median}(\| \boldsymbol{h}_k^T \boldsymbol{v} - \text{median}(\boldsymbol{h}_j^T \boldsymbol{v}) \|)} \right] \tag{43}$$

where $c = 1.4826$ in the denominator is a correction factor to ensure unbiasedness [15]. The maximization problem can be approximated by considering only the directions that correspond to the unit vectors of the individual data points relative to the median of the point cloud. An algorithm for computing the approximate projection statistics is given in [16].

2. Robust Covariance Estimation

The Myers–Tapley adaptive tuning method can be modified in order to account for non-Gaussianity by means of using the robust covariance estimates based on the projection statistics of the stored residuals. In particular, the measurement and process noise covariance estimates can be written as

$$\hat{\boldsymbol{R}}^{\star} = \left[\sum_{i=1}^{N} w_{\zeta_i} - 1 \right]^{-1} \cdot \left[\sum_{i=1}^{N} (w_{\zeta_i} \boldsymbol{\zeta}_i - \bar{\boldsymbol{\zeta}}_r)(w_{\zeta_i} \boldsymbol{\zeta}_i - \bar{\boldsymbol{\zeta}}_r)^T \right] - \text{median}_{j \in N} [(\boldsymbol{H}_i \bar{\boldsymbol{P}}_j^{1/2}) \boldsymbol{\Psi}_x^{-1} (\boldsymbol{H}_j \bar{\boldsymbol{P}}_j^{1/2})^T] \tag{44}$$

$$\hat{\boldsymbol{Q}} = \left[\sum_{i=1}^{N} w_{\lambda_i} - 1 \right]^{-1} \cdot \left[\sum_{i=1}^{N} (w_{\lambda_i} \boldsymbol{\lambda}_i - \bar{\boldsymbol{\lambda}}_r)(w_{\lambda_i} \boldsymbol{\lambda}_i - \bar{\boldsymbol{\lambda}}_r)^T \right] - \text{median}_{j \in N} (\bar{\boldsymbol{P}}_k^{\star} - \hat{\boldsymbol{P}}_j) \tag{45}$$

where w_{ζ_i} and w_{λ_i} are the weights based on the projection statistics of the measurement and process noise residuals, respectively, and

$$\bar{\boldsymbol{\zeta}}_r = \left[\sum_{i=1}^{N} w_{\zeta_i} \right]^{-1} \cdot \left[\sum_{i=1}^{N} w_{\zeta_i} \zeta_i \right]$$

$$\bar{\boldsymbol{\lambda}}_r = \left[\sum_{i=1}^{N} w_{\lambda_i} \right]^{-1} \cdot \left[\sum_{i=1}^{N} w_{\lambda_i} \lambda_i \right] \tag{46}$$

where w_i are weights computed from the statistics by means of $w_i = \min[1, \quad (\chi^2_{n,\alpha}/\mathcal{P}_i^2)]$, for some specified probability α.

The matrix $\hat{\boldsymbol{R}}^{\star}$ is related to the measurement noise covariance estimate as

$$\hat{\boldsymbol{R}}^{\star} = \hat{\boldsymbol{R}}^{1/2} \bar{\boldsymbol{\Psi}}_y^{-1} (\hat{\boldsymbol{R}}^{1/2})^T \tag{47}$$

where $\bar{\boldsymbol{\Psi}}_y^{-1}$ is the median value of the $\boldsymbol{\Psi}_y$ matrix across the buffer of stored observations and residuals. The estimate of the measurement noise covariance matrix can be determined from a square-root decomposition of Eq. (44). Specifically,

$$\hat{\boldsymbol{R}} = (\hat{\boldsymbol{R}}^{\star})^{1/2} \bar{\boldsymbol{\Psi}}_y (\hat{\boldsymbol{R}}^{\star})^{1/2\,T} \tag{48}$$

Note that in the DD1 and DD2 filter formulation, the quantity $(\boldsymbol{H}_j \bar{\boldsymbol{P}}_j^{1/2}) \boldsymbol{\Psi}_x^{-1} (\boldsymbol{H}_j \bar{\boldsymbol{P}}_j^{1/2})^T$ is replaced by $\boldsymbol{S}'_{y\bar{x}_k} \boldsymbol{\Psi}_x^{-1} \boldsymbol{S}'^{\,T}_{y\bar{x}_k}$ (DD1) and $\mathcal{H}([\, \boldsymbol{S}'_{y\bar{x}_k} \boldsymbol{\Psi}_x^{-1/2} \quad \boldsymbol{S}''_{y\bar{x}_k} \boldsymbol{\Psi}_x^{-1/2}\,])$ (DD2) for the measurement noise covariance estimation. Similarly, the quantity $\bar{\boldsymbol{P}}_k^{\star} = \boldsymbol{\Phi}_{k-1} \hat{\boldsymbol{P}}_{k-1} \boldsymbol{\Phi}_{k-1}^T$ is replaced by $\boldsymbol{S}'_{x\hat{x}_k} \boldsymbol{S}'^{\,T}_{x\hat{x}_k}$ (DD1) and $\mathcal{H}([\, \boldsymbol{S}'_{x\hat{x}_k} \quad \boldsymbol{S}''_{x\hat{x}_k}\,])$ (DD2) for the process noise covariance estimation.

As noted in [4], numerical issues can arise, particularly with small buffer sizes, in which the noise covariance estimates can become negative definite. Here, the same approach suggested in [4] can be

used to guarantee that numerical issues like these do not occur in practice. Namely, the diagonal elements of $\hat{\boldsymbol{R}}$ and $\hat{\boldsymbol{Q}}$ are reset to their absolute values.

3. Estimation of the Contamination Parameter

The modified Myers–Tapley approach discussed in the previous section can also offer a crude scheme for estimating the contamination parameter ϵ by using the weighting parameters relating to the stored residual data. In particular, a crude estimate of the contamination parameter is

$$\hat{\epsilon}_k = 1 - \frac{1}{N}\sum_{i=1}^{N} w_{\zeta_i} \tag{49}$$

At each frame where the measurement and process noise covariances are computed, the contamination parameter can be estimated directly from the weighting parameters. Then, the optimal tuning parameter can be calculated from the guidelines provided in [10], which is then used within the Huber filter at each measurement update. Note, however, that it should be expected that the estimated contamination parameter will in general be biased as in cases of large contamination there will be some portion of errors drawn from the contaminating distribution that appear to be drawn from the nominal distribution, in other words false-negatives or the so-called Type 2 errors in detection theory. Likewise in cases of small contamination there will be some nonzero quantity of data that appear as outliers when they are in fact perfectly valid, in other words the Type 1 error in detection theory.

The bias of the contamination parameter is not necessarily problematic, so long as upper and lower bounds are set on the value of the tuning parameter γ used in the Huber measurement update. Clearly, the Huber technique is by nature a suboptimal filter, since the purpose is to find a filter that is consistent across a range of distributions but not necessarily optimal at any one in particular. Therefore one could expect the performance of the technique to improve by having good estimates of the contamination parameter, but so long as the tuning parameter is bounded by some reasonable value then the robustness properties of the estimator will not be compromised by such a bias.

4. Fading Memory Filter

In the course of computing the estimates of the measurement and process noise covariance matrices, as well as the contamination parameter estimates, it is useful to introduction a "forgetting" factor, k_f, in order to smooth the estimate histories. The filter can be implemented as [12]

$$\tilde{\boldsymbol{R}}_k = k_f \tilde{\boldsymbol{R}}_{k-1} + (1-k_f)\hat{\boldsymbol{R}}_k \tag{50}$$

$$\tilde{\boldsymbol{Q}}_k = k_f \tilde{\boldsymbol{Q}}_{k-1} + (1-k_f)\hat{\boldsymbol{Q}}_k \tag{51}$$

$$\tilde{\epsilon}_k = k_f \tilde{\epsilon}_{k-1} + (1-k_f)\hat{\epsilon}_k \tag{52}$$

In this approach, the estimates based on the current set of stored residuals is averaged with the previous estimate, with k_f as a weighting parameter, where small values lead to rapid response of the estimates and larger values provide a smoother response. In general the recommended value of k_f depends on the particular application and can be treated as a tuning parameter.

IV. Application to Elliptical Orbit Rendezvous Navigation

A. Rendezvous Dynamics

The development of the 6-DOF rendezvous equations of motion requires the definition of several coordinate systems. First, a planet-centered inertial frame, $\mathcal{I}$, is introduced. This frame is aligned with the target spacecraft orbit (assuming no perturbations) such that the x axis is oriented towards the periapsis, the z axis is oriented along the positive orbit normal, and the y axis completes a right-handed system. The local, or $\mathcal{L}$, frame has its origin located at the position of the target spacecraft and is defined such that the z axis is oriented toward the center of the planet, the y axis is oriented along the negative orbit normal, and the x axis is in the transverse direction, completing the right-handed system. The $\mathcal{I}$ and $\mathcal{L}$ frames are illustrated in Fig. 1.

The chaser body frame is denoted by $\mathcal{B}$ with right-handed axes. It is assumed that the various sensors carried onboard the chaser spacecraft produce outputs referenced to this body frame. These sensors are introduced in later subsections, following the development of the equations of motion.

1. Rotational Dynamics

The rotational kinematics are represented using the MRPs, which are defined in terms of the quaternions (q_1, q_2, q_3, q_4) as [6,17–19]

$$\boldsymbol{\sigma} = \frac{\boldsymbol{q}}{1+q_4} = \boldsymbol{e}\tan\left(\frac{\theta}{4}\right) \tag{53}$$

where $\boldsymbol{q} = (q_1, q_2, q_3)$ is the vector part of the quaternion, q_4 is the scalar part of the quaternion, $\boldsymbol{e}$ is the principal rotation axis, and θ is the principal rotation angle. The MRPs have several useful properties for describing attitude kinematics for navigation and control problems. In particular, a simple singularity avoidance technique can be used to provide globally nonsingular attitude descriptions. This property is described in detail in [7], along with the covariance transformations that must accompany the singularity avoidance technique.

MRPs satisfy the kinematic differential equation [6]

$$\dot{\boldsymbol{\sigma}} = \frac{1}{4}\boldsymbol{B}(\boldsymbol{\sigma})\boldsymbol{\omega} = \frac{1}{4}[(1-\boldsymbol{\sigma}^T\boldsymbol{\sigma})\boldsymbol{I} + 2\boldsymbol{\sigma}^\times + 2\boldsymbol{\sigma}\boldsymbol{\sigma}^T]\boldsymbol{\omega} \tag{54}$$

where $\boldsymbol{\omega}$ is the angular velocity.

The dynamic equations describing the change in angular momentum are the well-known Euler equations

$$\dot{\boldsymbol{\omega}} = -\mathbb{I}^{-1}\boldsymbol{\omega}^\times\mathbb{I}\boldsymbol{\omega} + \mathbb{I}^{-1}\boldsymbol{\tau} \tag{55}$$

where $\boldsymbol{\omega}$ is the inertial angular velocity of the body, $\mathbb{I}$ is the inertia tensor of the vehicle, and $\boldsymbol{\tau}$ is the net control and/or disturbance torque. The relative angular velocity between the chaser and the $\mathcal{L}$ frame is $\delta\boldsymbol{\omega} = \boldsymbol{\omega} - \boldsymbol{\Omega}$ where $\boldsymbol{\Omega}$ is the angular velocity of the $\mathcal{L}$ frame (with all quantities represented in the spacecraft $\mathcal{B}$ frame). By solving for $\boldsymbol{\omega}$ and substituting into Eq. (55), the dynamic equation for the relative angular velocity becomes

$$\begin{aligned}\delta\dot{\boldsymbol{\omega}} = &-\mathbb{I}^{-1}\delta\boldsymbol{\omega}^\times\mathbb{I}\delta\boldsymbol{\omega} - \mathbb{I}^{-1}\delta\boldsymbol{\omega}^\times\mathbb{I}\boldsymbol{\Omega} - \mathbb{I}^{-1}\boldsymbol{\Omega}^\times\mathbb{I}\delta\boldsymbol{\omega}\\ &- \mathbb{I}^{-1}\boldsymbol{\Omega}^\times\mathbb{I}\boldsymbol{\Omega} + \dot{\boldsymbol{\Omega}} + \mathbb{I}^{-1}\boldsymbol{\tau}\end{aligned} \tag{56}$$

The angular velocity $\boldsymbol{\Omega}$ is given by $\boldsymbol{\Omega} = \boldsymbol{T}(\boldsymbol{\sigma})[0 \quad -\omega_0 \quad 0]^T$, where $\boldsymbol{\sigma}$ is the MRP representing the transformation from the $\mathcal{L}$ frame

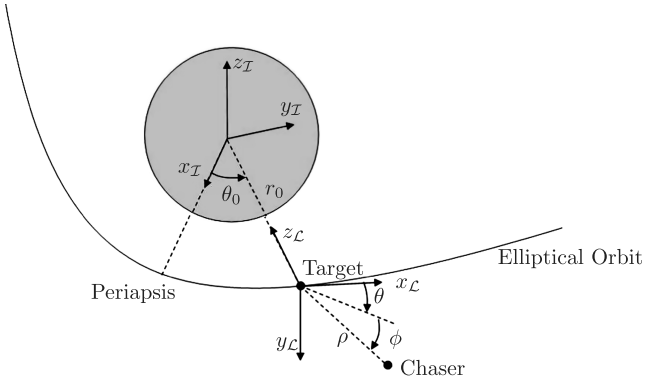


Fig. 1 Illustration of $\mathcal{I}$ and $\mathcal{L}$ coordinate frames.

to the $\mathcal{B}$ frame, $\boldsymbol{T}(\boldsymbol{\sigma})$ is the transformation matrix, and ω_0 is the angular velocity of the reference orbit.

2. Translational Dynamics

The translation equations of motion for a spacecraft relative to an elliptical Keplerian reference orbit are well known and can be written as [6]

$$\ddot{x} - 2\omega_0\dot{z} - \dot{\omega}_0 z - \omega_0^2 x + \frac{\mu x}{[x^2 + y^2 + (r_0 - z)^2]^{3/2}} = u_x + v_x \tag{57}$$

$$\ddot{y} + \frac{\mu y}{[x^2 + y^2 + (r_0 - z)^2]^{3/2}} = u_y + v_y \tag{58}$$

$$\ddot{z} + 2\omega_0\dot{x} + \dot{\omega}_0 x - \omega_0^2 z + \frac{\mu}{r_0^2} - \frac{\mu(r_0 - z)}{[x^2 + y^2 + (r_0 - z)^2]^{3/2}} = u_z + v_z \tag{59}$$

$$\ddot{r}_0 - r_0\omega_0^2 + \frac{\mu}{r_0^2} = 0 \tag{60}$$

$$\dot{\omega}_0 + \frac{2\dot{r}_0\omega_0}{r_0} = 0 \tag{61}$$

Here, x is the in-track position component, z is the position component along the negative radial direction, and y is the position along the negative orbit normal direction. The variables r_0 and ω_0 correspond to the radius and angular velocity of the reference orbit, and μ is the gravitational parameter of the planet. It is convenient to introduce a transformation of variables such that the equations of motion are in spherical coordinates centered at the reference point [20]. Such a transformation can be found by setting

$$x = \rho\cos\phi\cos\theta \tag{62}$$

$$y = \rho\sin\phi \tag{63}$$

$$z = -\rho\cos\phi\sin\theta \tag{64}$$

where ρ is the range between the target spacecraft and the maneuvering spacecraft, θ is an azimuth angle in the reference orbit plane measured from the x axis, positive toward the $-z$ direction, and ϕ is an out of plane angle measured from the x-z plane positive toward the y axis. After the substitution of this transformation the equations of motion become

$$\ddot{\rho} = (\omega_0 - \dot{\theta})^2\rho\cos^2\phi + \rho\dot{\phi}^2 + \frac{\mu}{r_0^2}\sin\theta\cos\phi - \frac{\mu(\rho + r_0\cos\phi\sin\theta)}{(r_0^2 + 2r_0\rho\cos\phi\sin\theta + \rho^2)^{3/2}} + u_\rho + v_\rho \tag{65}$$

$$\ddot{\theta} = \dot{\omega}_0 + 2(\omega_0 - \dot{\theta})\frac{\dot{\rho}}{\rho} - 2(\omega_0 - \dot{\theta})\dot{\phi}\tan\phi + \frac{\mu\cos\theta\sec\phi}{r_0^2\rho} - \frac{\mu r_0\cos\theta\sec\phi}{\rho(r_0^2 + 2r_0\rho\cos\phi\sin\theta + \rho^2)^{3/2}} + u_\theta + v_\theta \tag{66}$$

$$\ddot{\phi} = -\frac{1}{2}(\omega_0 - \dot{\theta})^2\sin 2\phi - \frac{2\dot{\phi}\,\dot{\rho}}{\rho} - \frac{\mu\sin\theta\sin\phi}{r_0^2\rho} + \frac{\mu r_0\sin\theta\sin\phi}{\rho(r_0^2 + 2r_0\rho\cos\phi\sin\theta + \rho^2)^{3/2}} + u_\phi + v_\phi \tag{67}$$

where the control inputs are

$$u_\rho = u_x\cos\theta\cos\phi + u_y\sin\phi - u_z\sin\theta\cos\phi \tag{68}$$

$$u_\theta = -u_x\left[\frac{2r_0\rho\sin^2\theta + (r_0^2 + \rho^2)\sin\theta\sec\phi}{\rho(r_0^2 + 2r_0\rho\cos\phi\sin\theta + \rho^2)}\right] - u_z\left[\frac{2r_0\rho\cos\theta\sin\theta + (r_0^2 + \rho^2)\cos\theta\sec\phi}{\rho(r_0^2 + 2r_0\rho\cos\phi\sin\theta + \rho^2)}\right] \tag{69}$$

$$u_\phi = -u_x\frac{\cos\theta\sin\phi}{\rho} + u_y\frac{\cos\phi}{\rho} + u_z\frac{\sin\theta\sin\phi}{\rho} \tag{70}$$

with similar relations for the process noise inputs.

The translation equations of motion can be written in the form

$$\ddot{\boldsymbol{\eta}} = \boldsymbol{f}(\boldsymbol{\eta}, \dot{\boldsymbol{\eta}}, t) + \boldsymbol{u} + \boldsymbol{v} \tag{71}$$

where $\boldsymbol{\eta} = [\,\rho \quad \theta \quad \phi\,]^T$, $\boldsymbol{u} = [\,u_\rho \quad u_\theta \quad u_\phi\,]^T$ and $\boldsymbol{v} = [\,v_\rho \quad v_\theta \quad v_\phi\,]^T$.

B. Rendezvous Guidance and Control

This section develops 6-DOF guidance and control schemes for rendezvous in elliptical orbit. These guidance and control schemes are fairly simple in nature, but they are adequate for the purposes of illustrating the navigation filter performance during rendezvous maneuvers.

1. Translational Guidance and Control

Given a commanded translational reference trajectory, $\boldsymbol{\eta}_c$, a translation error state can be defined as $\delta\boldsymbol{\eta} = \boldsymbol{\eta} - \boldsymbol{\eta}_c$. This error state obeys the differential equations

$$\delta\ddot{\boldsymbol{\eta}} = \ddot{\boldsymbol{\eta}} - \ddot{\boldsymbol{\eta}}_c = \boldsymbol{f}(\boldsymbol{\eta}, \dot{\boldsymbol{\eta}}, t) + \boldsymbol{u} + \boldsymbol{v} - \ddot{\boldsymbol{\eta}}_c \tag{72}$$

This system can be controlled by using a feedback linearization of the form

$$\boldsymbol{u} = \ddot{\boldsymbol{\eta}}_c - \boldsymbol{f}(\boldsymbol{\eta}, \dot{\boldsymbol{\eta}}, t) - \boldsymbol{K}_\eta\delta\boldsymbol{\eta} - \boldsymbol{K}_{\dot{\eta}}\delta\dot{\boldsymbol{\eta}} \tag{73}$$

By substituting Eq. (73) into Eq. (72), the closed-loop system becomes

$$\delta\ddot{\boldsymbol{\eta}} + \boldsymbol{K}_\eta\delta\boldsymbol{\eta} + \boldsymbol{K}_{\dot{\eta}}\delta\dot{\boldsymbol{\eta}} = \boldsymbol{v} \tag{74}$$

This closed-loop system is asymptotically stable in the absence of disturbance inputs for any gain matrices $\boldsymbol{K}_\eta > 0$ and $\boldsymbol{K}_{\dot{\eta}} > 0$. Note that nonzero disturbance accelerations result from both model error and feedback error due to erroneous state estimates from the navigation outputs. This mixture of deterministic and stochastic disturbances suggests a gain design using a suboptimal H_∞ approach, which is described in [20].

It is assumed here that a suitable reference trajectory $\boldsymbol{\eta}_c$ is available to provide the guidance commands. In general this trajectory can be specified using a multitude of approaches depending on the specific nature of the rendezvous and docking sequence and path constraints for a particular spacecraft application.

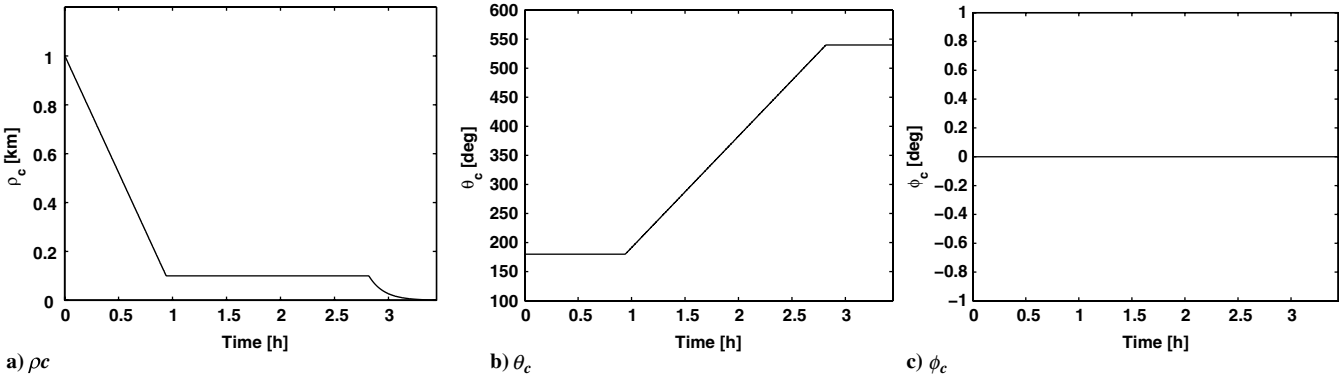
2. Attitude Guidance and Control

Attitude guidance commands are generated by solving for the MRP $\boldsymbol{\sigma}_c$ that provides zero azimuth and elevation angles so that the radar boresight is aligned with the target vehicle. The following system of nonlinear algebraic equations is solved at each guidance cycle:

$$\alpha_c(\boldsymbol{\sigma}_c) = 0 \tag{75}$$

$$\varepsilon_c(\boldsymbol{\sigma}_c) = 0 \tag{76}$$

$$\varphi_c(\boldsymbol{\sigma}_c) = 0 \tag{77}$$



a) ρc b) θ_c c) ϕ_c

Fig. 2 Guidance commands.

The quantity φ_c specifies the relative roll angle and is defined by

$$\varphi_c = \boldsymbol{y}_{\mathcal{L}} \cdot \boldsymbol{z}_{\mathcal{B}} = [0 \quad 1 \quad 0] \cdot \boldsymbol{T}(\boldsymbol{\sigma})^T \cdot [0 \quad 0 \quad 1]^T = \frac{8\sigma_2\sigma_3 + 4\sigma_1(1 - \boldsymbol{\sigma}^T\boldsymbol{\sigma})}{(1 + \boldsymbol{\sigma}^T\boldsymbol{\sigma})^2} \tag{78}$$

A zero-order hold is applied to the MRP attitude commands in order to generate a MRP rate command. The rate command is computed using a backward difference derivative between the current MRP command and the lagged MRP command. The commanded MRP and MRP rate are the inputs to the attitude control law.

Given the commanded MRP and MRP rate, $\boldsymbol{\sigma}_c$ and $\dot{\boldsymbol{\sigma}}_c$, respectively, an attitude control law to track these commands can be constructed using the multi-input backstepping method [21]. Here, the control design is split into a sequence of two subproblems. First, the angular velocity is assumed to be a control, which is chosen to stabilize the attitude kinematics. Next, torque commands are generated in order to track the desired angular velocity. To construct this controller, an error MRP is defined as $\delta\boldsymbol{\sigma} = \boldsymbol{\sigma} \ominus \boldsymbol{\sigma}_c$, where $\ominus$ indicates the MRP subtraction rule [6]. The error MRP obeys the kinematic differential equation

$$\delta\dot{\boldsymbol{\sigma}} = \frac{1}{4}\boldsymbol{B}(\delta\boldsymbol{\sigma})\delta\boldsymbol{\omega} \tag{79}$$

Assuming the relative angular rate $\delta\boldsymbol{\omega} = \boldsymbol{\nu}(\delta\boldsymbol{\sigma})$ can be chosen arbitrarily, a Lyapunov function can be constructed in order to stabilize the relative MRP kinematics. To this end, a candidate Lyapunov function is $V(\delta\boldsymbol{\sigma}) = 2k_v\,\ell n(1 + \delta\boldsymbol{\sigma}^T\delta\boldsymbol{\sigma})$ where $k_v > 0$ is a gain.

Making use of an identity in [6], the candidate Lyapunov function rate is

$$\dot{V} = \frac{4k_v}{1 + \delta\boldsymbol{\sigma}^T\delta\boldsymbol{\sigma}}\delta\boldsymbol{\sigma}^T\delta\dot{\boldsymbol{\sigma}} = \frac{k_v}{1 + \delta\boldsymbol{\sigma}^T\delta\boldsymbol{\sigma}}\delta\boldsymbol{\sigma}^T\boldsymbol{B}(\delta\boldsymbol{\sigma})\boldsymbol{\nu}(\delta\boldsymbol{\sigma}) = k_v\delta\boldsymbol{\sigma}^T\boldsymbol{\nu}(\delta\boldsymbol{\sigma}) \tag{80}$$

By choosing the function $\boldsymbol{\nu}(\delta\boldsymbol{\sigma}) = -\boldsymbol{K}_\sigma\delta\boldsymbol{\sigma}$, where $\boldsymbol{K}_\sigma > 0$, the Lyapunov function rate becomes

$$\dot{V} = -k_v\delta\boldsymbol{\sigma}^T\boldsymbol{K}_\sigma\delta\boldsymbol{\sigma} < 0 \tag{81}$$

According to Theorem 4.2 in [21], the relative kinematics are asymptotically stable with this choice of function $\boldsymbol{\nu}(\delta\boldsymbol{\sigma})$ since the function $V(\delta\boldsymbol{\sigma})$ is positive definite with negative definite rate.

Next, control torques must be provided that cause the spacecraft angular velocity to track the desired angular velocity $\delta\boldsymbol{\omega} = \boldsymbol{\nu}(\delta\boldsymbol{\sigma}) = -\boldsymbol{K}_\sigma\delta\boldsymbol{\sigma}$. An augmented Lyapunov function can be introduced as

$$V_a(\delta\boldsymbol{\sigma}, \delta\boldsymbol{\omega}) = V(\delta\boldsymbol{\sigma}) + \frac{1}{2}[\delta\boldsymbol{\omega} - \boldsymbol{\nu}(\delta\boldsymbol{\sigma})]^T[\delta\boldsymbol{\omega} - \boldsymbol{\nu}(\delta\boldsymbol{\sigma})] \tag{82}$$

The Lyapunov function rate is

$$\begin{aligned}\dot{V}_a(\delta\boldsymbol{\sigma}, \delta\boldsymbol{\omega}) &= \frac{\partial V}{\partial\delta\boldsymbol{\sigma}}\delta\dot{\boldsymbol{\sigma}} + [\delta\boldsymbol{\omega} - \boldsymbol{\nu}(\delta\boldsymbol{\sigma})]^T\left[\delta\dot{\boldsymbol{\omega}} - \frac{\partial\boldsymbol{\nu}}{\partial\delta\boldsymbol{\sigma}}\delta\dot{\boldsymbol{\sigma}}\right] \\ &= \frac{1}{4}\frac{\partial V}{\partial\delta\boldsymbol{\sigma}}\boldsymbol{B}(\delta\boldsymbol{\sigma})\boldsymbol{\nu}(\delta\boldsymbol{\sigma}) + \frac{1}{4}\frac{\partial V}{\partial\delta\boldsymbol{\sigma}}\boldsymbol{B}(\delta\boldsymbol{\sigma})[\delta\boldsymbol{\omega} - \nu(\delta\boldsymbol{\sigma})] \\ &\quad + [\delta\boldsymbol{\omega} - \boldsymbol{\nu}(\delta\boldsymbol{\sigma})]^T\Big[-\mathbb{I}^{-1}\delta\boldsymbol{\omega}^\times\mathbb{I}\delta\boldsymbol{\omega} - \mathbb{I}^{-1}\delta\boldsymbol{\omega}^\times\mathbb{I}\boldsymbol{\Omega} - \mathbb{I}^{-1}\boldsymbol{\Omega}^\times\mathbb{I}\delta\boldsymbol{\omega} \\ &\quad - \mathbb{I}^{-1}\boldsymbol{\Omega}^\times\mathbb{I}\boldsymbol{\Omega} + \dot{\boldsymbol{\Omega}} + \mathbb{I}^{-1}\boldsymbol{\tau} - \frac{1}{4}\frac{\partial\boldsymbol{\nu}}{\partial\delta\boldsymbol{\sigma}}\boldsymbol{B}(\delta\boldsymbol{\sigma})\delta\boldsymbol{\omega}\Big]\end{aligned} \tag{83}$$

By choosing the control input

$$\begin{aligned}\boldsymbol{\tau} &= \delta\boldsymbol{\omega}^\times\mathbb{I}\delta\boldsymbol{\omega} + \delta\boldsymbol{\omega}^\times\mathbb{I}\boldsymbol{\Omega} + \boldsymbol{\Omega}^\times\mathbb{I}\delta\boldsymbol{\omega} + \boldsymbol{\Omega}^\times\mathbb{I}\boldsymbol{\Omega} - \mathbb{I}\dot{\boldsymbol{\Omega}} \\ &\quad + \frac{1}{4}\mathbb{I}\frac{\partial\boldsymbol{\nu}}{\partial\delta\boldsymbol{\sigma}}\boldsymbol{B}(\delta\boldsymbol{\sigma})\delta\boldsymbol{\omega} - \frac{1}{4}\mathbb{I}\left[\frac{\partial V}{\partial\delta\boldsymbol{\sigma}}\boldsymbol{B}(\delta\boldsymbol{\sigma})\right]^T - \mathbb{I}\boldsymbol{K}_\omega[\delta\boldsymbol{\omega} - \boldsymbol{\nu}(\delta\boldsymbol{\sigma})]\end{aligned} \tag{84}$$

Table 1 Initial conditions

Initial state	Mean	Standard deviation
$\rho(0)$, km	1.0	0.005
$\theta(0)$, deg	180.0	0.25
$\phi(0)$, deg	0.0	0.25
$\dot{\rho}(0)$, m/s	−0.266	0.05
$\dot{\theta}(0)$, deg/s	0.0	0.003
$\dot{\phi}(0)$, deg/s	0.0	0.003
σ_1, rad	0.0	0.0175
σ_2, rad	0.0	0.0175
σ_3, rad	0.0	0.0175
β_1, deg/hr	0.0	1.0
β_2, deg/hr	0.0	1.0
β_3, deg/hr	0.0	1.0
r_0, km	1753.1	0.01
θ_0, deg	0.0	0.001
$\dot{r}_0$, m/s	0.0	$1.0 \cdot 10^{-5}$
$\dot{\theta}_0$, deg/s	$9.619 \cdot 10^{-5}$	$1.0 \cdot 10^{-5}$

Table 2 Sensor specifications

Measurement	Standard deviation
$\boldsymbol{\eta}_\omega$, deg/s	$1.8 \cdot 10^{-5}\boldsymbol{I}$
$\boldsymbol{\eta}_\beta$, deg/s^2	$1.8 \cdot 10^{-8}\boldsymbol{I}$
$\boldsymbol{\eta}_a$, m/s^2	$0.1\boldsymbol{I}$
$\boldsymbol{S}$, ppm	500
ϱ, m	0.1
α, deg	0.05
ε, deg	0.05
$\delta\boldsymbol{\sigma}$, deg	$0.05\boldsymbol{I}$
$\delta\boldsymbol{\sigma}_I$, deg	$0.05\boldsymbol{I}$
r_0, m	0.1
θ_0, deg	$3.3 \cdot 10^{-5}$

where $\boldsymbol{K}_\omega > 0$, the Lyapunov function rate becomes

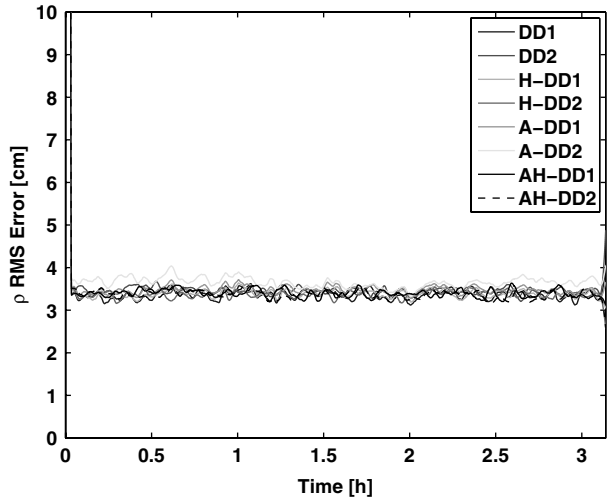
$$\dot{V}_a = -\delta\boldsymbol{\sigma}^T\boldsymbol{K}_\sigma\delta\boldsymbol{\sigma} - [\delta\boldsymbol{\omega} - \boldsymbol{\nu}(\delta\boldsymbol{\sigma})]^T\boldsymbol{K}_\omega[\delta\boldsymbol{\omega} - \boldsymbol{\nu}(\delta\boldsymbol{\sigma})] < 0 \quad (85)$$

Since the function V_s is positive definite and radially unbounded with a negative definite rate, it follows from Lyapunov's direct method (Theorem 4.2 in [21]) that the origin of the closed-loop system is globally asymptotically stable.

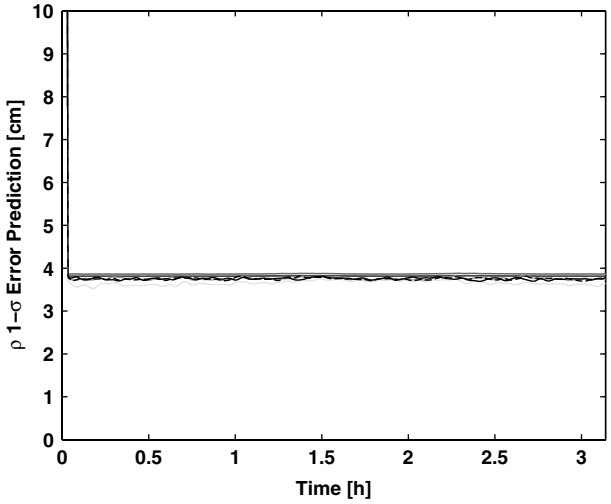
By substituting the appropriate expressions for $V(\delta\boldsymbol{\sigma})$ and $\boldsymbol{\nu}(\delta\boldsymbol{\sigma})$, the control torque τ is

$$\boldsymbol{\tau} = \delta\boldsymbol{\omega}^\times\mathbb{I}\delta\boldsymbol{\omega} + \delta\boldsymbol{\omega}^\times\mathbb{I}\boldsymbol{\Omega} + \boldsymbol{\Omega}^\times\mathbb{I}\delta\boldsymbol{\omega} + \boldsymbol{\Omega}^\times\mathbb{I}\boldsymbol{\Omega} - \mathbb{I}\dot{\boldsymbol{\Omega}} - \frac{1}{4}\mathbb{I}\boldsymbol{K}_\sigma\boldsymbol{B}(\delta\boldsymbol{\sigma})\delta\boldsymbol{\omega} - \frac{k_v}{2}\mathbb{I}\delta\boldsymbol{\sigma} - \mathbb{I}\boldsymbol{K}_\omega[\delta\boldsymbol{\omega} + \boldsymbol{K}_\sigma\delta\boldsymbol{\sigma}] \quad (86)$$

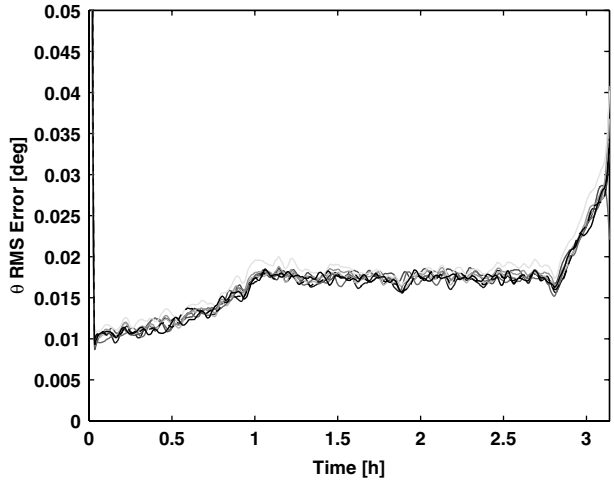
Substituting this control law into Eq. (56) and linearizing about $(\delta\boldsymbol{\sigma}, \delta\boldsymbol{\omega}) = (\boldsymbol{0}, \boldsymbol{0})$ yields



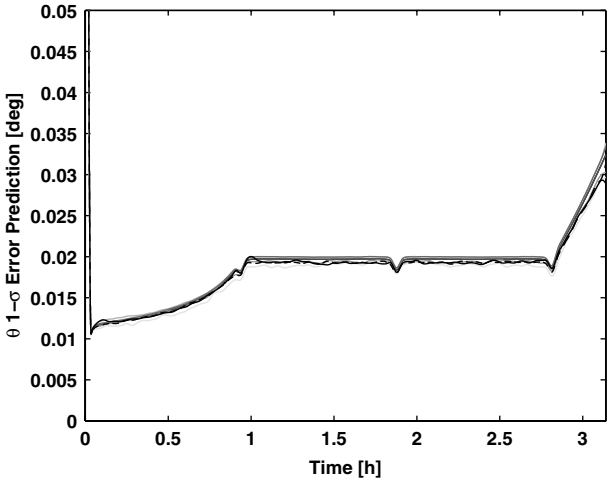
a) ρ RMS Error



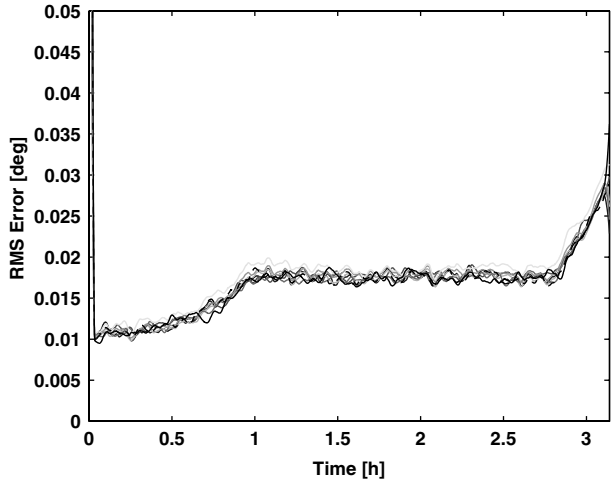
b) ρ RMS Error Prediction



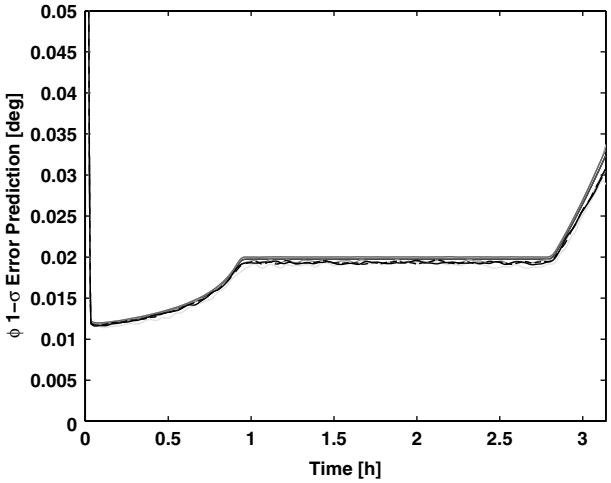
c) θ RMS Error



d) θ RMS Error Prediction



e) ϕ RMS Error



f) ϕ RMS Error Prediction

Fig. 3 Gaussian simulation results: position.

$$\delta\dot{\boldsymbol{\sigma}} = \frac{1}{4}\delta\boldsymbol{\omega} \tag{87}$$

$$\delta\dot{\boldsymbol{\omega}} = -\left(\frac{k_v}{2}\boldsymbol{I} + \boldsymbol{K}_\omega\boldsymbol{K}_\sigma\right)\delta\boldsymbol{\sigma} - \left(\boldsymbol{K}_\omega + \frac{1}{4}\boldsymbol{K}_\sigma\right)\delta\boldsymbol{\omega} \tag{88}$$

Following a similar logic as that discussed in the translational controller section, the gains can be chosen according to a suboptimal H_∞ controller development [20].

C. Rendezvous Sensors

Vehicles performing autonomous rendezvous maneuvers must make use of a wide variety of navigation sensors to estimate the trajectory and perhaps other relevant parameters in order to successfully complete the mission. This section discusses the modeling of sensor systems whose data can be processed to estimate the relative position, velocity, and orientation of the chaser vehicle.

1. *Inertial Measurement Unit*

An IMU is a device which measures applied accelerations and angular rates by using a system of accelerometers and gyroscopes. The gyroscope system can be represented mathematically by using Farrenkopf's model. In this model, the sensed angular velocity if expressed as the true angular velocity with an additive bias and white noise. The bias term is itself a slowly varying parameter driven by white noise. The model can be expressed as [22]

$$\tilde{\boldsymbol{\omega}} = \boldsymbol{\omega} + \boldsymbol{\beta} + \boldsymbol{\eta}_\omega \tag{89}$$

$$\dot{\boldsymbol{\beta}} = \boldsymbol{\eta}_\beta \tag{90}$$

where $\tilde{\boldsymbol{\omega}}$ is the sensed inertial angular velocity, $\boldsymbol{\omega}$ is the true inertial angular velocity, $\boldsymbol{\beta}$ is the measurement bias, and $\boldsymbol{\eta}_\omega$ and $\boldsymbol{\eta}_\beta$ are unbiased and uncorrelated random vectors with variances given by σ_ω^2 and σ_β^2, respectively. Discrete-time simulated gyroscope measurements can be generated according to this model by use of the equations [23]

$$\tilde{\boldsymbol{\omega}}_{k+1} = \boldsymbol{\omega}_k + \frac{1}{2}(\boldsymbol{\beta}_{k+1} + \boldsymbol{\beta}_k) + \left(\frac{\sigma_\omega^2}{\Delta t} + \frac{1}{12}\sigma_\beta^2\Delta t\right)^{1/2}\boldsymbol{n}_\omega \tag{91}$$

$$\boldsymbol{\beta}_{k+1} = \boldsymbol{\beta}_k + \sigma_\beta(\Delta t)^{1/2}\boldsymbol{n}_\beta \tag{92}$$

where k refers to the time increment, $\Delta t = t_{k+1} - t_k$ is the sampling interval, and $\boldsymbol{n}_\omega$ and $\boldsymbol{n}_\beta$ are unbiased, uncorrelated, unit-variance random vectors.

An accelerometer system measures the accelerations applied to the spacecraft. A model for acceleration measurements including scale factor and noise errors is

$$\tilde{\boldsymbol{a}}_m = (\boldsymbol{I} + \boldsymbol{S})\boldsymbol{a}_m + \boldsymbol{\eta}_a \tag{93}$$

where $\boldsymbol{a}_m$ is the true acceleration at the IMU location, $\tilde{\boldsymbol{a}}_m$ is the sensed acceleration, $\boldsymbol{\eta}_a$ is the measurement noise, and $\boldsymbol{S}$ is a diagonal matrix of constant scale factor errors. The accelerations at the vehicle center of mass can be calculated from $\boldsymbol{a} = \boldsymbol{a}_m - (\dot{\boldsymbol{\omega}}^\times + \boldsymbol{\omega}^\times\boldsymbol{\omega}^\times)\boldsymbol{r}_m$ where $\boldsymbol{a}$ is the center of mass acceleration and $\boldsymbol{r}_m$ is the position of the IMU with respect to the vehicle center of mass.

2. *Star Tracker Sensors*

It is assumed that a star tracker or some other attitude sensor is available to provide corrections to the attitude estimates formed by direct numerical integration of the angular velocity measurements. The star tracker is assumed to output an estimated MRP that relates the orientation of the body to the inertial frame. The estimates are assumed to be unbiased but with a superimposed random measurement noise. The output from such a sensor can be expressed as $\tilde{\boldsymbol{\sigma}}_I = \boldsymbol{\sigma}_I + \delta\boldsymbol{\sigma}$ where $\boldsymbol{\sigma}_I$ is the MRP representing the true orientation of the vehicle with respect to inertial space, $\tilde{\boldsymbol{\sigma}}_I$ is the "measured" MRP, and $\delta\boldsymbol{\sigma}$ is an error MRP with covariance matrix denoted by $\boldsymbol{R}$. For instance, the measured MRP could be an output from the algorithm described in [24], involving vector measurements. The inertial attitude MRP, $\boldsymbol{\sigma}_I$, can be written as a function of the relative attitude, $\boldsymbol{\sigma}$, and the inertial orbital position angle θ_0.

3. *Laser/Radar Navigation Sensors*

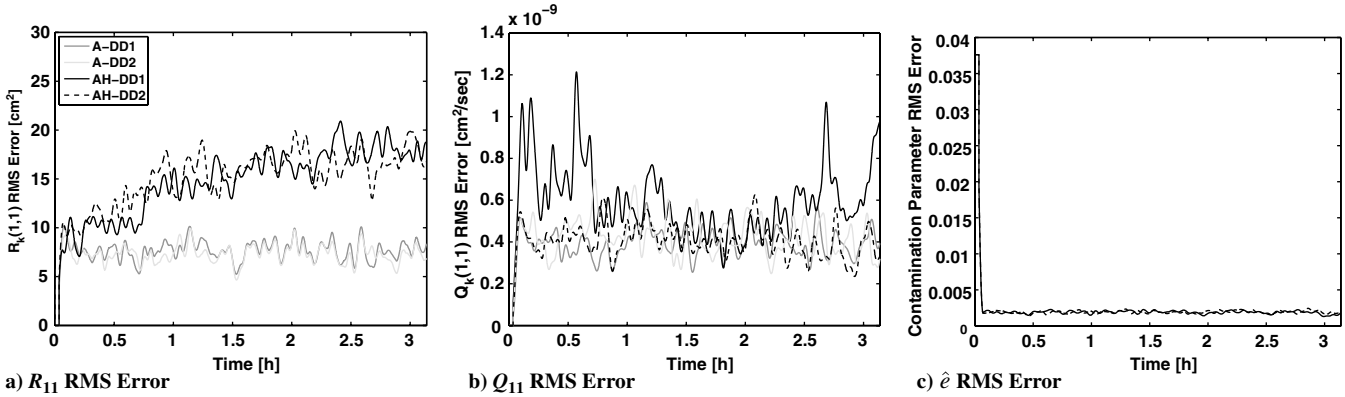
A radar or laser sensing system provides range, azimuth, elevation measurements between the sensor and the target. These measurements can be modeled as

$$\varrho\begin{Bmatrix}\cos\alpha\cos\varepsilon\\ \sin\alpha\cos\varepsilon\\ \sin\varepsilon\end{Bmatrix} = \boldsymbol{T}(\boldsymbol{\sigma})\boldsymbol{r} - \delta\boldsymbol{r}_s \tag{94}$$

where ϱ is the range, α is the azimuth angle, and ε is the elevation angle. Here, $\boldsymbol{r} = [x, \quad y, \quad z]^T$ is the Cartesian relative position in the local frame and $\delta\boldsymbol{r}_s$ is the sensor position in the body frame. It is also assumed that the relative attitude between the target and chaser can also be determined in addition to range and bearing by the laser sensing system. This relative attitude is modeled in the same manner as the inertial attitude sensor given in Sec. IV.C.2, namely that $\tilde{\boldsymbol{\sigma}} = \boldsymbol{\sigma} + \delta\boldsymbol{\sigma}$ where $\tilde{\boldsymbol{\sigma}}$ is the measured relative MRP, $\boldsymbol{\sigma}$ is the true relative attitude MRP and $\delta\boldsymbol{\sigma}$ is an error MRP.

4. *Orbit Sensor*

A wide variety of sensors can be used to estimate the orbit of the spacecraft. For example, Global Position System orbit determination sensors can be used to Earth orbiting and some lunar orbit cases [25]. In other cases, such as lunar orbit, autonomous orbit determination can be accomplished using optical sensors and landmark tracking [26]. For the purposes of this paper, it is assumed that some general orbit determination sensor is available for use in the state estimator. This sensor is assumed to provide position data in the form of radius r_0 and orbit angle θ_0 measurements.



a) $\boldsymbol{R}_{11}$ RMS Error b) $\boldsymbol{Q}_{11}$ RMS Error c) $\hat{e}$ RMS Error

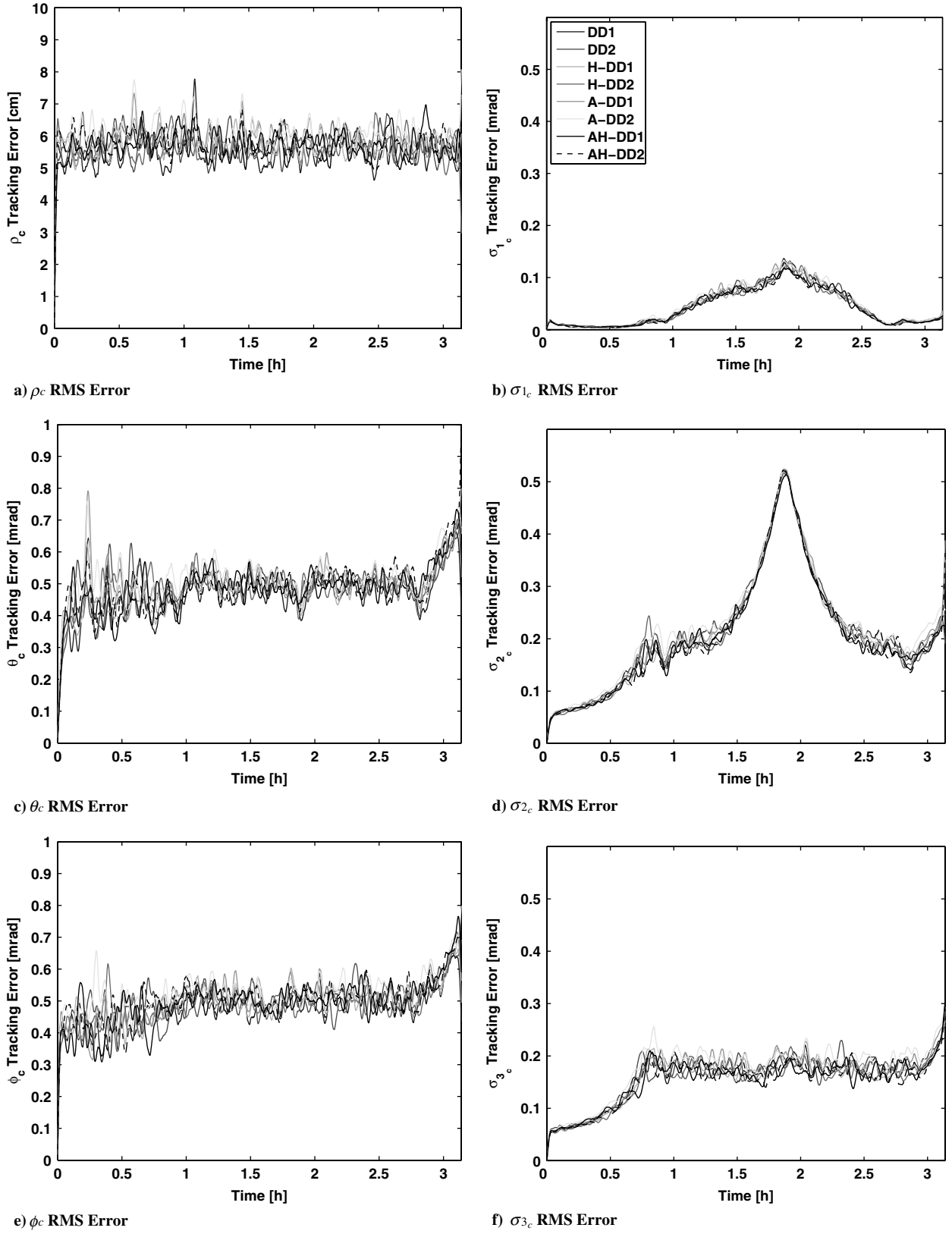
Fig. 4 Gaussian simulation results: covariance and contamination.

D. Numerical Simulations

1. Overview and Simulation Setup

This section described the application of the robust/adaptive filtering algorithms to the problem of 6-DOF rendezvous navigation and control in elliptical orbit. Here, the navigation filters are used inside the control loop as state observers.

The specific example discussed in this section involves terminal rendezvous maneuver in a 15 by 75 km altitude lunar orbit. This orbit is typical of intermediate phasing orbit during ascent from the lunar surface. Hypothetically, the lunar ascent vehicle could have failed, leaving it stranded in such an orbit. Therefore, a CEV/Orion-like vehicle must maneuver from a circular parking orbit, typically



a) ρ_c **RMS Error**

b) σ_{1_c} **RMS Error**

c) θ_c **RMS Error**

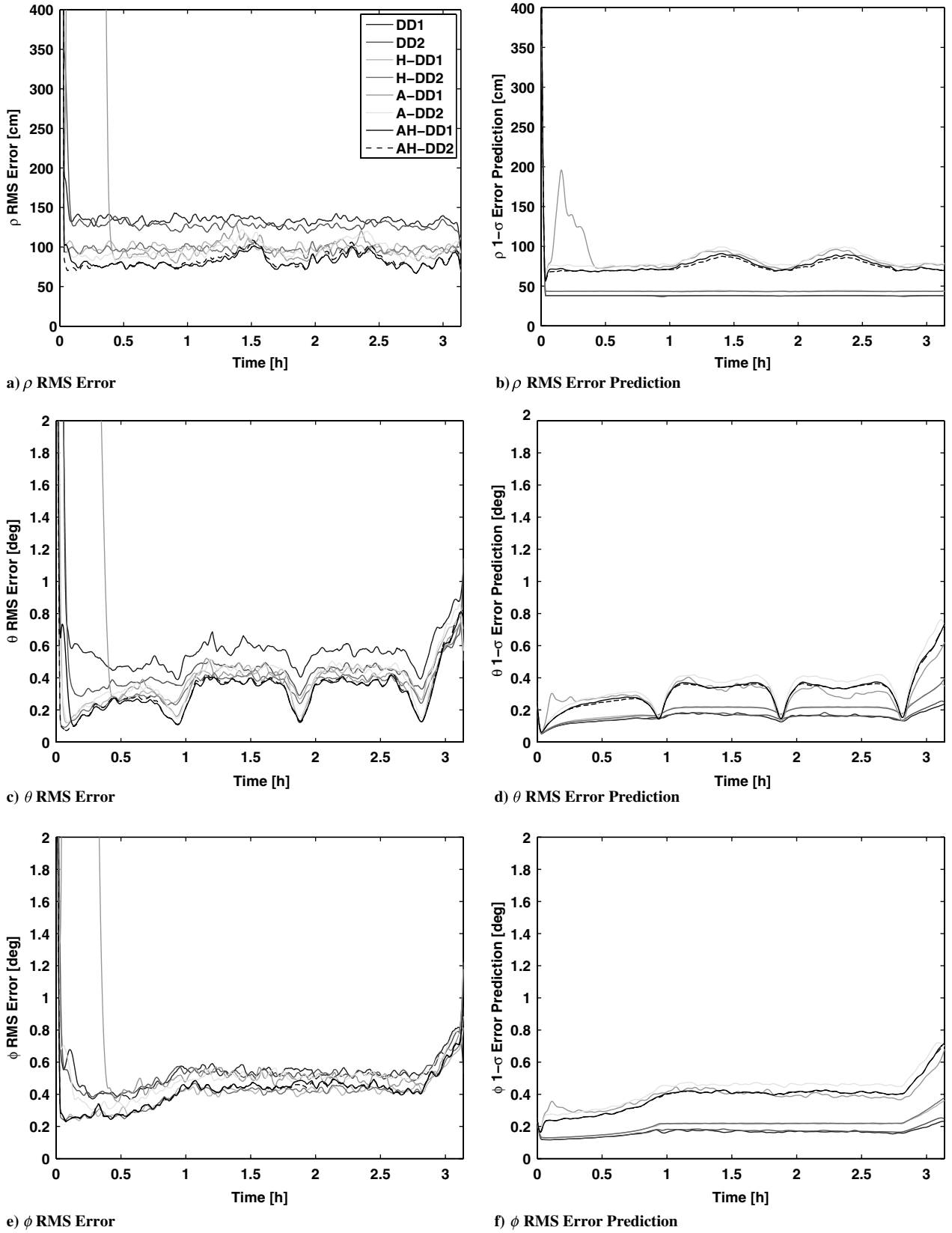
d) σ_{2_c} **RMS Error**

e) ϕ_c **RMS Error**

f) σ_{3_c} **RMS Error**

Fig. 5 Gaussian simulation results: closed-loop system performance.

100 km altitude, to rendezvous with the lunar ascent vehicle. This problem assumes that the midcourse maneuvering and phasing has already been accomplished such that the initial conditions of the maneuvering vehicle are 1 km behind the target vehicle in the in-track direction. The rendezvous trajectory begins with a constant closure rate from 1 km to 100 m over a duration of half an orbital period (in this problem, one orbital period is approximately 113 min). At this point, the vehicle is commanded to execute a circumnavigation of the target vehicle at a constant range of 100 m over one orbital period. Finally, the vehicle executes a glideslope maneuver using a Space Shuttle-based exponentially decaying range rate guidance scheme [27] with decay rate of 0.2% over a time span of half an orbital period.



a) ρ RMS Error

b) ρ RMS Error Prediction

c) θ RMS Error

d) θ RMS Error Prediction

e) ϕ RMS Error

f) ϕ RMS Error Prediction

Fig. 6 Non-Gaussian simulation results: position.

These guidance commands are shown in Fig. 2. Attitude commands are generated such that the vehicle remains pointed at the target throughout the maneuver. The guidance commands are generated at a rate of 4 Hz. The control laws and navigation filter update rates are at 4 Hz. The inertial navigation sensors operate at 20 Hz. Control force and torque limits were set to 1779.2 N and 5337.6 Nm, respectively.

The nominal vehicle mass properties correspond to a total mass of $1.1 \cdot 10^5$ kg with inertia tensor given by

$$\boldsymbol{I} = 10^4 \cdot \begin{bmatrix} 2.9441 & 0.0368 & 0.3680 \\ 0.0368 & 3.6801 & 0.0074 \\ 0.3680 & 0.0074 & 3.6801 \end{bmatrix} \text{kg} \cdot \text{m}^2 \qquad (95)$$

Monte Carlo simulations are conducted for this problem for several different navigation filters. In particular, the extended Kalman filter and first- and second-order divided-difference filters, including Huber implementations, Myers–Tapley adaptive implementations, and combined Huber–Myers–Tapley adaptive filtering methods. Results are shown in the following subsections for both Gaussian and non-Gaussian cases.

2. *Gaussian Simulation*

Rendezvous simulation conducted with pure Gaussian errors are described in this section. The initial conditions of the maneuvering vehicle are at the time of perilune passage of the target vehicle and are provided in Table 1. Table 1 shows the true value of the initial conditions, the estimate of the initial conditions for filter initialization, and the components of the initial variance matrix for filter initialization. The initial state errors are uncorrelated in this simulation. The vehicle mass properties are dispersed by multiplying the total mass by a Gaussian random variable with standard deviation of 0.5% and a inertia tensor uncertainty by the mass multiplier coupled with a uncorrelated random axis uncertainty with 0.5 deg standard deviation in yaw-pitch-roll Euler angles.

The navigation sensor uncertainties are summarized in Table 2, which shows the measurement standard deviations for the gyro parameters ($\boldsymbol{\eta}_\omega$ and $\boldsymbol{\eta}_\beta$), the accelerometer errors ($\boldsymbol{\eta}_a$ and $\boldsymbol{S}$), the rendezvous LIDAR sensor errors (ϱ, α, ε, and $\delta\boldsymbol{\sigma}$), the star tracker errors ($\delta\boldsymbol{\sigma}_I$) and the orbit sensor errors (r_0 and θ_0). For this simulation, the errors are cast as uncorrelated Gaussian random numbers with standard deviations given in Table 2. The filter measurement noise and process noise matrices are set according to the sensor errors in Table 2 without error in the assumed values. In this simulation, the accelerometer scale factor errors are not modeled in the filter formulation as state variables in order to provide a mismatch between the true measurements and the modeled measurements in the filter.

Results are shown from the first- and second-order divided-difference filters (DD1 and DD2), including Huber implementations (H-DD1 and H-DD2), Myers–Tapley adaptive implementations (A-DD1 and A-DD2), and combined Huber-Myers–Tapley adaptive filtering methods (AH-DD1 and AH-DD2). The adaptive filters use a buffer size of 500 samples in the procedures to estimate the measurement and process noise covariances. Results from the extended Kalman filter approaches are essentially identical to the first-order divided-difference filtering approaches, both being first-order filters, and therefore are omitted from the rms error plots in an effort to keep the results clear.

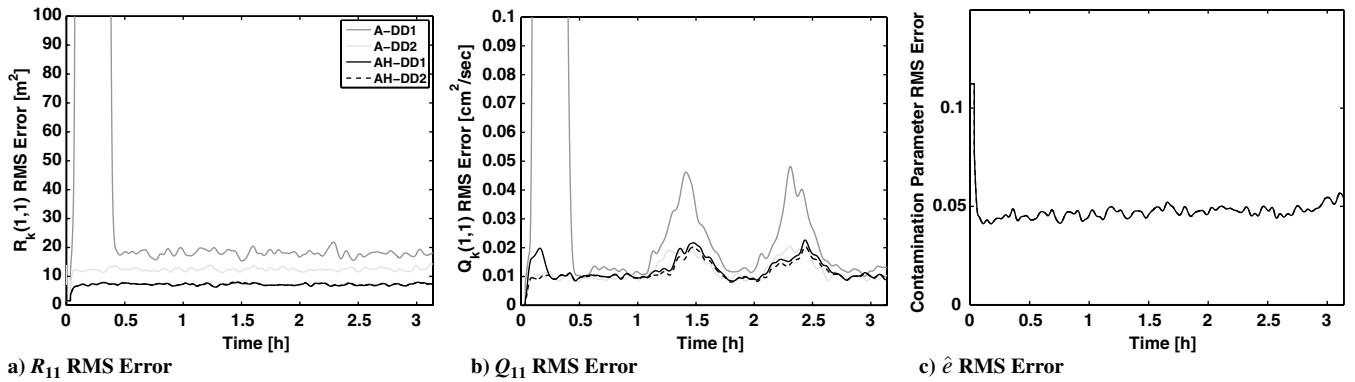
Figure 3 shows the position rms error results. The upper row shows the actual rms error plots for the position variables ρ, θ, and ϕ while the lower row shows the predicted rms errors based on the filter covariance matrix. Although the differences between the performance various filters in this case is small, it is possible to discern a slightly better performance in the Gaussian-based filters such as the DD1 and DD2. Note in this simulation that the measurement and process noise cases were initialized without error. Therefore, the adaptive filters do not perform as well as the nonadaptive filters since their measurement and process noise covariances are not exact estimates due to the finite sample sizes used in the buffering technique.

The predicted rms error results show the trend that the adaptive Gaussian filters (A-DD1 and A-DD2) exhibit the smallest error prediction. This trend is due to the adaptive tuning of the filter in real time, which has the effect of reducing the state error covariance matrix. This reduction is erroneous, since the actual rms error results are higher due to the introduction of error into the measurement and process noise covariances. The Huber-based filters (H-DD1 and H-DD2) correctly predict that the rms error should be higher than the other filters. The adaptive Huber filters (AH-DD1 and AH-DD2) have a slightly lower rms error prediction than the nonadaptive Huber filters due to the adaptive tuning.

Figure 4 shows a comparison of results for the adaptive filters investigated in this study. Figure 4a shows the rms estimation error of the $(1, 1)$ component of the measurement noise covariance matrix, which corresponds to the range measurement from the LIDAR sensor. Here, the A-DD2 exhibits the best performance, which is expected since the errors are purely Gaussian and since the DD2 filter has the benefits of capturing second-order terms in the state error covariance estimates. The Huber-based estimates exhibit higher errors due to the reduced statistical efficiency of the projection statistics algorithm for purely Gaussian errors. Figure 4b shows the $(1, 1)$ component of the process noise covariance estimate. In this case differences between the various adaptive filters is not as obvious, thought the A-DD2 filter shows slightly better performance than the others. Figure 4c shows the results of estimating the contamination parameter in the Huber-based adaptive filters. Here, both filters AH-DD1 and AH-DD2 are able to reduce the error in the assumed contamination parameter.

Finally, the closed-loop system performance results are shown in Fig. 5. These plots show the true position and attitude rms error results in the upper and lower rows, respectively. The relative performance of the system is not significantly different between the various filters, although Gaussian filters DD1 and DD2 show slight improvement. This result is expected since the errors are purely Gaussian and the filters have perfect knowledge of the measurement and process noise covariance matrices.

To summarize the results of the Gaussian simulation, the overall trends follow the expectation that the Gaussian DD1 and DD2 filters



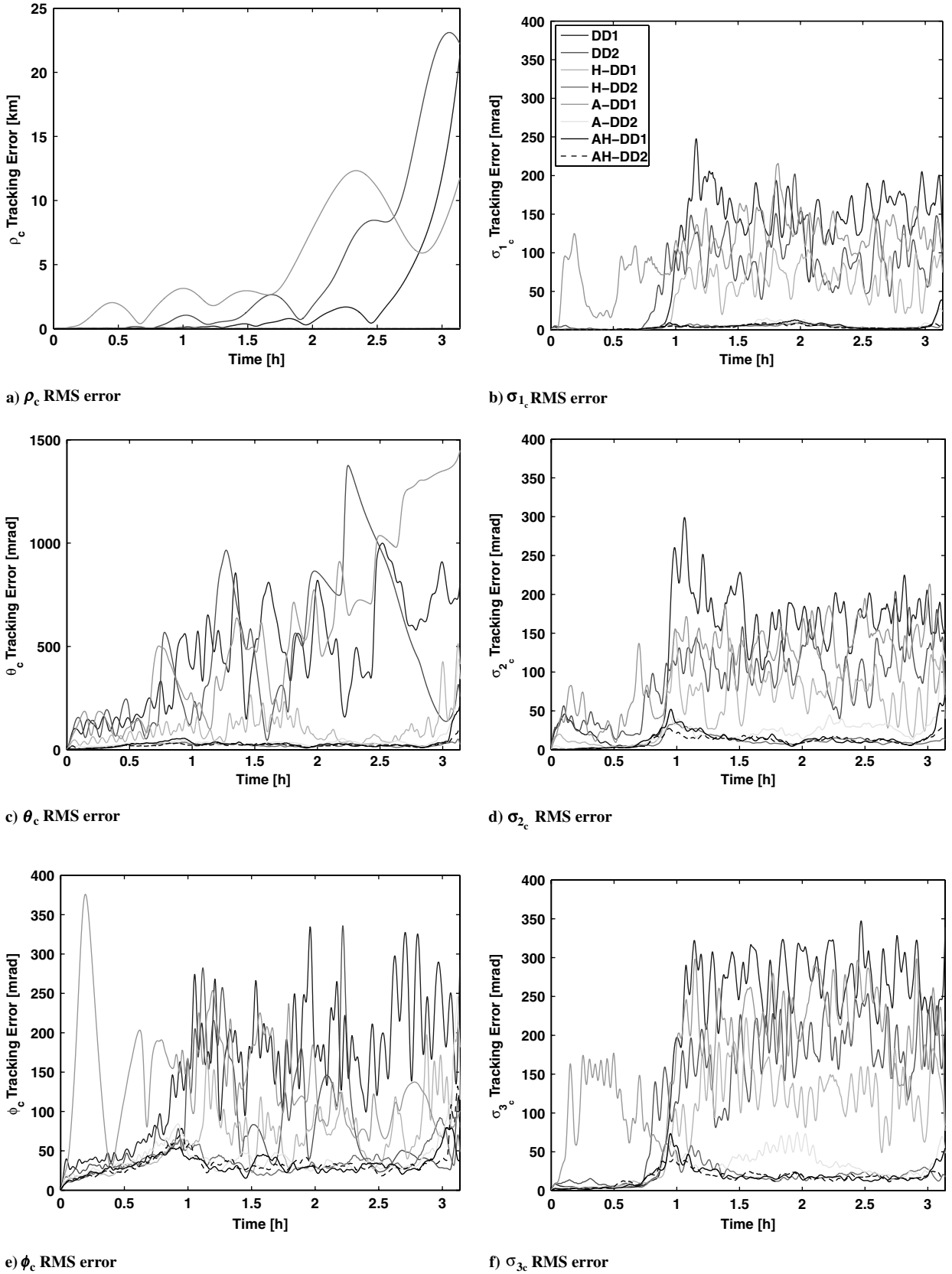
a) $\boldsymbol{R}_{11}$ RMS Error b) $\boldsymbol{Q}_{11}$ RMS Error c) $\hat{e}$ RMS Error

Fig. 7 Non-Gaussian simulation results: covariance and contamination.

should have the best overall performance since the errors in the simulation are purely Gaussian and these filters have perfect knowledge of the measurement and process noise covariance matrices. However, the differences between the various filters is fairly small, implying that for the benign Gaussian case the navigation filters are essentially identical in overall performance.

3. *Non-Gaussian Simulation*

In the Non-Gaussian case, the noise samples are drawn from a Gaussian mixture model with 15% contamination from the higher variance distribution. In an effort to stress the filters, the standard deviations of the primary Gaussian density of the mixture model were set to 10 times the values provided in Table 1, with the



a) ρ_c RMS error

b) σ_{1_c} RMS error

c) θ_c RMS error

d) σ_{2_c} RMS error

e) ϕ_c RMS error

f) σ_{3c} RMS error

Fig. 8 Non-Gaussian simulation results: closed-loop system performance.

contaminating densities an additional 5 times higher. Additionally, the true standard deviations are scaled according to a uniform distribution to be up to twice as high as the assumed standard deviations in the navigation filter, and uniform random correlation coefficients were assigned to the true errors on each run. The mean initial conditions of this simulation is the same as that given in Table 2, but the true random initial errors are generated according to a randomly correlated covariance matrix with standard deviations set to 10 times the value in Table 2.

The rms errors of the position variables are shown in Fig. 6. In this non-Gaussian problem, the differences between the various filters become much more apparent. In particular the nonrobust filters suffer greatly from the contamination present in this simulation. In the range variable ρ, the DD1 filters shows the worst overall performance, closely followed by the DD2 filter. Interestingly, the A-DD1 filters exhibits large errors over the first 0.5 hrs of the simulation before finally settling to the steady state error. The A-DD2 filter benefits from the inclusion of second-order terms in the system and measurement dynamics and as a result, does not exhibit the large initial errors. The A-DD2 shows similar performance to the Huber-based filters, H-DD1 and H-DD2, which are nearly identical in this problem, with the differences between these two filters being comparable to the differences between the Gaussian DD1 and DD2 filters, though the rms errors are smaller by roughly 25%. The best performing filters are the AH-DD1 and AH-DD2 filters, which are essentially identical in this variable. The predicted rms errors show that the Gaussian filters DD1 and DD2 produce highly optimistic error estimates, having the smallest error predictions yet the largest actual rms error. The nonadaptive Huber estimates, H-DD1 and H-DD2, correctly show slightly larger error estimates. The adaptive filter A-DD1 has large initial error estimates, which correctly reflect the large actual errors although the magnitudes are quite different, with the predicted error being much smaller in magnitude. The A-DD2, AH-DD1, and AH-DD2 filters produce the most accurate error predictions in this particular variable. These same trends can be observed in the other position variables θ and ϕ, as shown in Fig. 6.

Results of the covariance and contamination estimation for the adaptive filters are shown in Fig. 7. The rms errors in these plots indicate that the A-DD1 filter has relatively inaccurate covariance estimates over the first 0.5 hrs of simulation time for range measurement error and the range process noise. Once converged to its steady state error level the magnitudes remain higher than the A-DD2 filter, which is expected since the A-DD2 filter has the benefit of second-order terms in the process and measurement transformations. The AH-DD1 and AH-DD2 filters are nearly identical in performance with the AH-DD2 filter performing slightly better.

The closed-loop system performance results are shown in Fig. 8. Here it can be seen that the DD1, DD2, A-DD1, and H-DD1 filter results have poor performance and essentially diverge in the presence of the non-Gaussian noise. The A-DD2 filter also gives erratic performance, which is more apparent in the attitude response. The H-DD2, AH-DD1, and AH-DD2 filters exhibit the best performance and are able to successfully execute the rendezvous maneuver even in the presence of a high degree of non-Gaussian noise with uncertain statistics. This result is interesting since the H-DD2 filter is nonadaptive, it is still better able to cope with the errors than the other filters investigated in this study.

4. *Computational Comparisons*

The relative computational costs of the various filters is provided in Table 3. Here, the median computational time is computed for each filter and then normalized by the EKF median computational time to provide a relative cost comparison. The results show that DD1 and DD2 filters cost roughly twice that of the EKF. The DD2 filter costs only slightly more than the DD1 filter. The Huber-based filters each cost roughly 7% more than the standard Kalman implementations. In the adaptive case, the adaptive Huber filters cost roughly 80% more than the standard Myers–Tapley adaptive implementations.

Table 3 Computational time comparisons

Filter	EKF	DD1	DD2
Kalman	1.000	1.938	2.124
Huber–Kalman	1.066	2.086	2.272
Adaptive Kalman	1.526	2.302	2.526
Adaptive Huber–Kalman	2.961	4.330	4.659

V. Conclusions

This paper discusses the development of an adaptive discrete-time robust filtering technique based on a recursive form of Huber's mixed minimum ℓ_1/ℓ_2 norm approach, combined with a new robust form of the adaptive Myers–Tapley covariance matching technique. This new adaptation technique adopts a robust approach to estimating these covariances that can resist the effects of outliers, based on the use of projection statistics, which have been previously developed for robust outlier identification as generalizations of the classical Mahalanobis distance measures but have not been applied to the adaptive state estimation problem in such a manner before.

The hybrid robust/adaptive filtering approaches are applied to the 6 degrees of freedom elliptical orbit rendezvous navigation problem. Numerical simulations are conducted with both Gaussian and non-Gaussian error distributions in order to asses the performance of the filtering techniques. The results indicate that the robust filters provide the same closed-loop performance as their nonrobust counterparts for purely Gaussian noise simulations. However for the non-Gaussian problem, the hybrid robust/adaptive filters are superior to the nonrobust filters by a considerable margin. Furthermore, the robust/adaptive filters can estimate the noise covariances and the contamination ratio in the non-Gaussian case. Therefore, the adaptive Huber filtering techniques introduced in this paper have better consistency and are self-tuning.

References

[1] Karlgaard, C. D., and Schaub, H., "Huber-Based Divided Difference Filtering," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 3, 2007, pp. 885–891. doi:10.2514/1.27968

[2] Fitzgerald, R. J., "Divergence of the Kalman Filter," *IEEE Transactions on Automatic Control*, Vol. 16, No. 6, 1971, pp. 736–747. doi:10.1109/TAC.1971.1099836

[3] Mehra, R. K., "Approaches to Adaptive Filtering," *IEEE Transactions on Automatic Control*, Vol. 17, No. 5, 1972, pp. 693–698. doi:10.1109/TAC.1972.1100100

[4] Myers, K. A., and Tapley, B. D., "Adaptive Sequential Estimation with Unknown Noise Statistics," *IEEE Transactions on Automatic Control*, Vol. 21, No. 4, 1976, pp. 520–523. doi:10.1109/TAC.1976.1101260

[5] Gasko, M., and Donoho, D. L., "Influential Observations in Data Analysis," *Proceedings of the Business and Economic Statistics Section*, American Statistical Association, 1982, pp. 104–110.

[6] Schaub, H., and Junkins, J. L., *Analytical Mechanics of Space Systems*, AIAA Education Series, AIAA, Reston, VA, 2003, pp. 107–111, 598.

[7] Karlgaard, C. D., and Schaub, H., "Nonsingular Attitude Filtering Using Modified Rodrigues Parameters," AAS/AIAA Space Flight Mechanics Meeting, AAS Paper 09–130, Feb. 2009.

[8] Van Loan, C. F., "Computing Integrals Involving the Matrix Exponential," *IEEE Transactions on Automatic Control*, Vol. 23, No. 3, 1978, pp. 395–404. doi:10.1109/TAC.1978.1101743

[9] Crassidis, J. L., and Junkins, J. L., *Optimal Estimation of Dynamic Systems*, CRC, Boca Raton, FL, 2004, p. 274.

[10] Karlgaard, C. D., and Schaub, H., "Comparison of Several Nonlinear Filters for a Benchmark Tracking Problem," AIAA Guidance, Navigation, and Control Conference, AIAA Paper 2006–6243, Keystone, CO, Aug. 2006.

[11] Nørgaard, M., Poulsen, N. K., and Ravn, O., "New Developments in State Estimation for Nonlinear Systems," *Automatica*, Vol. 36, No. 11, 2000, pp. 1627–1638. doi:10.1016/S0005-1098(00)00089-3

[12] Maybeck, P. S., Jensen, R. L., and Harnly, D. A., "An Adaptive Extended Kalman Filter for Target Image Tracking," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-17,

No. 2, 1981, pp. 173–180.
doi:10.1109/TAES.1981.309143

[13] Rousseeuw, P. J., and van Zomeren, B. C., "Unmasking Multivariate Outliers and Leverage Points," *Journal of the American Statistical Association*, Vol. 85, No. 411, 1990, pp. 633–639.
doi:10.2307/2289995

[14] Rousseeuw, P. J., and van Zomeren, B. C., "Robust Distances: Simulations and Cutoff Values," *Directions in Robust Statistics and Diagnostics, Part 2*, edited by W. Stahel and S. Weisberg, Vol. 34, The IMA Volumes in Mathematics and its Applications, Springer–Verlag, New York, 1991, pp. 195–203.

[15] Rousseeuw, P. J., and Leroy, A. M., *Robust Regression and Outlier Detection*, Wiley, New York, 1987, pp. 158–174.

[16] Des Rosiers, A. P., Schoenig, G. N., and Mili, L., "Robust Space-Time Adaptive Processing Using Projection Statistics," International Conference on Radar Systems, Toulouse, France, Société de l'Electricité, de l'Electronique et des Technologies de l'Information et de la Communication Paper 10P-SP-222, Oct. 2004.

[17] Weiner, T. F., "Theoretical Analysis of Gimballess Inertial Reference Equipment Using Delta-Modulated Instruments," Sc.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, March 1962.

[18] Shuster, M. D., "A Survey of Attitude Representations," *Journal of the Astronautical Sciences*, Vol. 41, No. 4, 1993, pp. 439–517.

[19] Schaub, H., and Junkins, J. L., "Stereographic Orientation Parameters for Attitude Dynamics: A Generalization of the Rodrigues Parameters," *Journal of the Astronautical Sciences*, Vol. 44, No. 1, 1996, pp. 1–19.

[20] Karlgaard, C. D., "Robust Adaptive Estimation for Autonomous Rendezvous in Elliptical Orbit," Ph.D. Dissertation, Department of Aerospace and Ocean Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA, June 2010.

[21] Khalil, H. K., *Nonlinear Systems*, 3rd ed., Prentice–Hall, Upper Saddle River, NJ, 2002, p. 124.

[22] Farrenkopf, R. L., "Analytic Steady-State Accuracy Solutions for Two Common Spacecraft Attitude Estimators," *Journal of Guidance and Control*, Vol. 1, No. 4, 1978, pp. 282–284.
doi:10.2514/3.55779

[23] Crassidis, J. L., "Sigma-Point Kalman Filtering for Integrated GPS and Inertial Navigation," AIAA Paper 2005-6087, Aug. 2005.

[24] Bruccoleri, C., and Mortari, D., "MRAD: Modified Rodrigues Vector Attitude Determination," *Journal of the Astronautical Sciences*, Vol. 54, Nos. 3–4, 2006, pp. 383–390.

[25] Stadter, P. A., Duven, D. J., Kantsiper, B. L., Sharer, P. J., Finnegan, E. J., and Weaver, G. L., "A Weak-Signal GPS Architecture for Lunar Navigation and Communication Systems," Institute of Electrical and Electronics Engineers Aerospace Conference, Big Sky, MT, Institute of Electrical and Electronics Engineers Paper 1336, March 2008.

[26] Hur-Diaz, S., Bamford, B., and Gaylor, D., "Autonomous Lunar Orbit Navigation Using Optical Sensors," AAS Paper 07-312, Aug. 2007.

[27] Brody, A. R., "Evaluation of the '0.1% Rule' for Docking Maneuvers," *Journal of Spacecraft and Rockets*, Vol. 27, No. 1, 1990, pp. 7–8.
doi:10.2514/3.26097