Decentralized Relative Attitude Estimation for Three-Spacecraft Formation Flying Applications

Christopher K. Nebelecky*, John L. Crassidis,† William D. Banas,‡ Yang Cheng§
University at Buffalo, State University of New York, Amherst, NY 14260-4400

Adam M. Fosbury¶
Air Force Research Laboratory, Space Vehicles Directorate, Kirtland Air Force Base, NM 87117

This paper investigates the problem of relative spacecraft attitude estimation between three vehicles from a decentralized point of view. Decentralized attitude estimation is achieved through the use of local extended Kalman filters and a data fusion process known as the Covariance Intersection algorithm. Because the global attitude parameterization is the quaternion, the Covariance Intersection algorithm is modified in order to handle the quaternion norm constraint using the method of Lagrange multipliers. A multivariate Newton-Raphson iteration is developed so that state vectors containing multiple quaternions may be fused. A formation flying simulation shows that the results of state fusion via the Covariance Intersection algorithm provides substantially better results than available from any one local source.

I. Introduction

An exciting new area of research involves using multiple small vehicles in cooperation to achieve a common mission objective, where in the past a single larger vehicle would have been employed. Many benefits arise from this new approach. Several applications using multiple vehicles have begun to take advantage of this technology. The robotics community is using cooperative robots which have the potential to revolutionize the manufacturing industry.[1,2] While other applications include underwater[3] and unmanned air vehicles,[4,5] the most profound impact of cooperative vehicles seems to be in the space industry—here, the technology may realize its fullest potential.

Unlike most terrestrial based applications, the space industry is highly limited by the size and mass of the spacecraft that can be launched. This limitation is motivation enough to prompt the development of cooperative vehicles for space missions. The use of multiple spacecraft together is typically referred to as formation flying and less-commonly known as distributed space systems (DSS) or fractionated spacecraft. One of the most common and well documented examples of using formation flying spacecraft is for the formation of synthetic apertures for high resolution space imaging.[6] An additional advantage of formation flying is the ability to distribute data and information among processors, or nodes spread throughout the formation. With limited computational processing available on modern-day space missions, the ability to distribute the computational burden allows for more complex missions as well as better system performance.[7,8] The distribution of data for control or estimation purposes results in what is called a decentralized system. Also, by using decentralized schemes, the entire formation is less vulnerable to failures of individual spacecraft.[9]

The benefits that arise from a cooperation of vehicles also comes with the cost of more complex dynamics, estimation and control algorithms. References [10] and [11] provide an in-depth review of the guidance

*Graduate Student, Department of Mechanical & Aerospace Engineering. E-mail: ckn@buffalo.edu. Student Member AIAA.
†Professor, Department of Mechanical & Aerospace Engineering. E-mail: johnc@eng.buffalo.edu. Associate Fellow AIAA.
‡Graduate Student, Department of Mechanical & Aerospace Engineering. E-mail: bill@banasclan.com. Member AIAA.
§Research Assistant Professor, Department of Mechanical & Aerospace Engineering. E-mail: cheng3@buffalo.edu. Senior Member AIAA.
¶Aerospace Engineer. E-mail: afrl.rvsv@kirtland.af.mil. Member AIAA.

and control issues associated with spacecraft formation flying applications. It is often the case in these applications that relative position and attitude information is more important than the inertial information. Reference [12] develops an extended Kalman filter for the relative attitude estimation of a pair of spacecraft using line-of-sight measurements. When the formation includes three spacecraft, a deterministic solution exists for relative attitudes within the formation.[13] This approach assumes all the necessary information is present to determine the attitude. When all the information is not present a deterministic solution is not possible. In this case an extended Kalman filter is used to estimate pertinent system states, presumably onboard each vehicle in the formation. In order to achieve the best possible estimates given this decentralized scheme, it is desirable to combine all available estimates in some optimal manner. Because of the nature of the formation, each set of state estimates is correlated, with the correlation being unknown. Thus simple averaging techniques can lead to inconsistent estimates. Here, we adopt the definition of consistency to mean that the estimated error is always greater than or equal to the actual error.

One solution to the consistency problem is the Covariance Intersection (CI) algorithm.[14,15] The CI algorithm provides a means to consistently fuse data with unknown correlations. The CI method has previously been applied to decentralized spacecraft attitude estimation onboard a single spacecraft.[16] Additional complexity arises in relative attitude estimation for a formation due to the additional attitude quaternions within the state vectors. For spacecraft attitude estimation, the four-dimensional quaternion is the attitude paramerization of choice for several reasons.[17] However, since a four-dimensional vector is used to describe three dimensions, the quaternion components cannot be independent of one another. For this reason the quaternion must maintain a unit norm. This norm provides increased complexity when combining quaternion estimated with the CI algorithm.

The paper is arranged as follows. First the CI algorithm is reviewed. A modification to the CI algorithm in order to include state vectors containing an arbitrary number of quaternions is presented. A solution to the CI algorithm is then given in terms of a Newton-Raphson iteration. Next, an extended Kalman filter is developed to estimate the relative attitudes within a formation of three spacecraft using line-of-sight and gyro measurements. A formation flying simulation with results completes the paper by showing the benefit of using the CI algorithm.

II. Decentralized Attitude Estimation

Since each estimator in a decentralized estimation scheme only processes a subset of the total measurements, the resulting estimates will be suboptimal. By utilizing information from each individual estimate, we strive to come up with a better representation of the estimated quantities. However, because some measurements may be utilized by more than one estimator, the resulting estimates will in general be correlated, with no means to determine the correlation. This redundancy can cause naive data fusion schemes to result in inconsistent estimates. The Covariance Intersection algorithm is a data fusion algorithm which preserves consistency when fusing estimates with unknown correlations.

A. The Covariance Intersection Algorithm

The CI algorithm is routed in the fundamentals of Gaussian intersection.[15] Appreciation for the CI algorithm can be gained by considering a simple two-dimensional state vector, estimated by two separate sources. In $\Re^2$ the estimate covariances can be represented by ellipses as shown in Fig. 1. When the correlations between the two estimates are exactly known, the optimal fused covariance lies completely within the intersection of the two individual covariance ellipses. As seen, the CI solution covariance ellipse passes through the intersections of the individual covariances.

The CI algorithm can be formally stated by saying that a consistent estimate $\mathbf{x}_{\text{CI}}$ and associated covariance P_{CI} can be constructed by fusing consistent estimates $\{\mathbf{x}_i,\ P_i\},\ i \in (1,\ n)$ by

$$P_{\text{CI}}^{-1} = \sum_{i=1}^{n} \gamma_i P_i^{-1} \tag{1a}$$

$$\mathbf{x}_{\text{CI}} = P_{\text{CI}} \sum_{i=1}^{n} \gamma_i P_i^{-1} \mathbf{x}_i \tag{1b}$$

where $\gamma_i \in [0,\ 1]$ is a scalar weight that satisfies $\sum_i \gamma_i = 1$. The conditions on the weights ensure that the

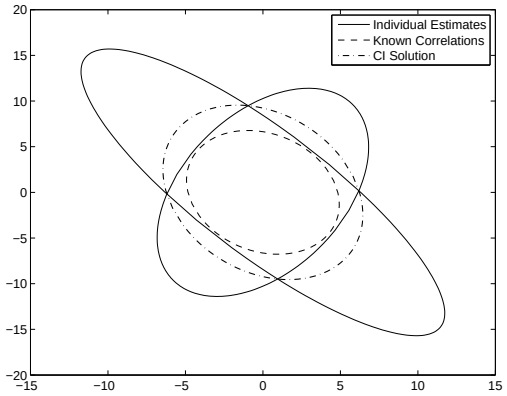


Figure 1. Geometric Interpretation of CI Algorithm

CI solution maintains consistency. The weights are somewhat arbitrary but typically chosen to minimize the trace or determinant of the fused covariance matrix P_{CI}.

B. Covariance Intersection for Attitude Estimation

The form of the CI algorithm shown in Eq. (1b) is limited to fusion of state vectors containing only unconstrained quantities. For attitude estimation it is typical that the four-dimensional quaternion will be used to parameterize the attitude. Because the quaternion is subject to a norm constraint, it is advantageous to express the CI algorithm as an optimization problem. This allows the constraints to be handled using the method of Lagrange multipliers.[18] Reference [16] has shown that the optimization problem for the CI algorithm can be written as

$$\min\ J(\mathbf{x}) = \sum_{i=i}^{n} \gamma_i \Delta\mathbf{x}_i^T P_i^{-1} \Delta\mathbf{x}_i + \sum_{j=1}^{N} \lambda_j \left(\mathbf{q}_j^T \mathbf{q}_j - 1\right) \tag{2}$$

where

$$\Delta\mathbf{x}_i = \begin{bmatrix} \Xi^T(\mathbf{q}_{1,i})\mathbf{q}_1 \\ \Xi^T(\mathbf{q}_{2,i})\mathbf{q}_2 \\ \vdots \\ \Xi^T(\mathbf{q}_{N,i})\mathbf{q}_N \\ \mathbf{b} - \mathbf{b}_i \end{bmatrix} \tag{3a}$$

is the i^{th} error vector associated with the state vector $\mathbf{x}$ defined as

$$\mathbf{x} = \begin{bmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \\ \vdots \\ \mathbf{q}_N \\ \mathbf{b} \end{bmatrix} \tag{4}$$

and the quaternions is defined as $\mathbf{q} = \begin{bmatrix} \boldsymbol{\varrho}^T & q_4 \end{bmatrix}^T$. The vector $\mathbf{b}$ includes any unconstrained quantities such as gyro biases. The matrix P_i is the covariance associated with the state vector $\mathbf{x}_i$ and is assumed to be the reduced order form associated with the small half-angle attitude errors and all remaining quantities.[19] The quaternion errors, given by $\Xi^T(\mathbf{q}_i)\mathbf{q}$ represent the small roll, pitch and yaw half-angle errors between the

quaternion estimates ($\mathbf{q}_i$) and the fused quaternion ($\mathbf{q}$). The matrix $\Xi(\mathbf{q})$ is given by

$$\Xi(\mathbf{q}) = \begin{bmatrix} q_4 I_{3\times 3} + [\boldsymbol{\varrho}\times] \\ -\boldsymbol{\varrho}^T \end{bmatrix} \tag{5a}$$

$$[\boldsymbol{\varrho}\times] = \begin{bmatrix} 0 & -\varrho_3 & \varrho_2 \\ \varrho_3 & 0 & -\varrho_1 \\ -\varrho_2 & \varrho_1 & 0 \end{bmatrix} \tag{5b}$$

It is convenient to express the error vector as

$$\begin{aligned} \Delta\mathbf{x}_i &= \begin{bmatrix} \Xi^T(\mathbf{q}_{1,i}) & 0_{3\times 4} & \dots & 0_{3\times 4} & 0_{3\times n_b} \\ 0_{3\times 4} & \Xi^T(\mathbf{q}_{2,i}) & \dots & 0_{3\times 4} & 0_{3\times n_b} \\ 0_{3\times 4} & 0_{3\times 4} & \ddots & \vdots & \vdots \\ \vdots & \vdots & \dots & \Xi^T(\mathbf{q}_{N,i}) & \vdots \\ 0_{n_b\times 4} & 0_{n_b\times 4} & \dots & 0_{n_b\times 4} & I_{n_b\times n_b} \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{0}_{3N\times 1} \\ \\ -\mathbf{b}_i \end{bmatrix} \\ &\equiv \boldsymbol{\Xi}_i \mathbf{x} + \mathbf{y}_{b_i} \end{aligned} \tag{6}$$

where n_b is the length of the unconstrained vector $\mathbf{b}$. Substituting Eq. (6) into (2) gives the following quadratic form for the cost function:

$$J(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{x}^T (F + \Pi)\, \mathbf{x} - 2\mathbf{x}^T \mathbf{g} + J^* - \sum_{j=1}^{N} \lambda_j \tag{7}$$

where

$$F \equiv \sum_{i=1}^{n} \gamma_i \boldsymbol{\Xi}_i^T P_i^{-1} \boldsymbol{\Xi}_i \tag{8a}$$

$$\mathbf{g} \equiv -\sum_{i=1}^{n} \gamma_i \boldsymbol{\Xi}_i^T P_i^{-1} \mathbf{y}_{b_i} \tag{8b}$$

$$J^* \equiv \sum_{i=1}^{n} \gamma_i \mathbf{y}_{b_i}^T P_i^{-1} \mathbf{y}_{b_i} \tag{8c}$$

$$\Pi \equiv \begin{bmatrix} \lambda_1 I_{4\times 4} & 0_{4\times 4} & \dots & 0_{4\times 4} & 0_{4\times n_b} \\ & \lambda_2 I_{4\times 4} & \dots & 0_{4\times 4} & 0_{4\times n_b} \\ & & \ddots & \vdots & \vdots \\ & & & \lambda_N I_{4\times 4} & \vdots \\ \text{sym} & & & & 0_{n_b\times n_b} \end{bmatrix} = \begin{bmatrix} \Lambda & 0_{4N\times n_b} \\ 0_{n_b\times 4N} & 0_{n_b\times n_b} \end{bmatrix} \tag{8d}$$

After taking the appropriate partials, the necessary conditions for optimality become

$$(F + \Pi)\mathbf{x} = \mathbf{g} \tag{9a}$$

$$\mathbf{q}_j^T \mathbf{q}_j = 1 \quad \forall j \in N \tag{9b}$$

Equation (9) represents $5N + n_b$ equations which must be simultaneously solved in order to determine the optimal fused estimate. Because $\mathbf{b}$ is unconstrained, it can be expressed in terms of the constrained variables. Begin by expressing Eq. (9a) as the block equation

$$\left(\begin{bmatrix} F_{qq} & F_{qb} \\ F_{qb}^T & F_{bb} \end{bmatrix} + \begin{bmatrix} \Lambda & 0_{4N\times n_b} \\ 0_{n_b\times 4N} & 0_{n_b\times n_b} \end{bmatrix} \right) \begin{bmatrix} \mathbf{Q} \\ \mathbf{b} \end{bmatrix} = \begin{bmatrix} \mathbf{g_q} \\ \mathbf{g_b} \end{bmatrix} \tag{10}$$

where $\mathbf{Q}$ is given the obvious definition of the vector containing all of the optimal quaternions. The lower n_b equations in Eq. (10) can be solved for the optimal $\mathbf{b}$ as

$$\mathbf{b} = F_{bb}^{-1} \left(\mathbf{g_b} - F_{qb}^T \mathbf{Q} \right) \tag{11}$$

Substituting Eq. (11) back into Eq. (10) results in

$$(\mathcal{F} + \Lambda)\,\mathbf{Q} = \mathbf{h} \tag{12}$$

where

$$\mathcal{F} \equiv F_{qq} - F_{qb}F_{bb}^{-1}F_{qb}^{T} \tag{13a}$$

$$\mathbf{h} \equiv \mathbf{g_q} - F_{qb}F_{bb}^{-1}\mathbf{g_b} \tag{13b}$$

Equation (12) along with (9b) now represent the $5N$ equations which must be solved to determine the optimal fused quaternions. The strategy here is to determine the optimal values of the Lagrange multipliers, after which the optimal quaternions may be computed directly from Eq. (12). Once the optimal quaternions are determined it is a trivial manner to compute the optimal $\mathbf{b}$ with Eq. (11).

Through this entire development we have only concerned ourselves with expressions for the optimal state vectors. The optimal covariance is calculated using Eq. (1a) just as one would do assuming the state vectors were unconstrained. The covariance is typically calculated before fusing the states owing to the fact that, in most cases, the weights γ_i are dependent upon the covariance calculation. Further discussion of the covariance and weight calculations can be found in Refs. [14–16].

C. A Newton-Raphson Solution to the CI Equations

While an analytic solution to Eq. (12) has yet to be found, it can be solved easily via Newton-Raphson iteration. The multivariate form of the Newton-Raphson iteration is given by[20]

$$\begin{bmatrix}\lambda_1\\ \vdots\\ \lambda_N\end{bmatrix}_{k+1} = \begin{bmatrix}\lambda_1\\ \vdots\\ \lambda_N\end{bmatrix}_{k} - \mathcal{J}^{-1}\begin{bmatrix}f_1\\ \vdots\\ f_N\end{bmatrix}\Bigg|_{\boldsymbol{\lambda}_k} \tag{14}$$

where f_j is the j^{th} constraint equation from Eq. (9b) expressed as

$$f_j(\boldsymbol{\lambda}) = \mathbf{q}_j^T\mathbf{q}_j - 1 = 0 \tag{15}$$

and $\mathcal{J}$ is the Jacobian matrix:

$$\mathcal{J} = \begin{bmatrix}\frac{\partial f_1}{\partial \lambda_1} & \cdots & \frac{\partial f_N}{\partial \lambda_1}\\ \vdots & \ddots & \vdots\\ \frac{\partial f_N}{\partial \lambda_1} & \cdots & \frac{\partial f_N}{\partial \lambda_N}\end{bmatrix} \tag{16}$$

Realization of the constraints begins by solving for the optimal quaternion vector from Eq. (12)

$$\mathbf{Q} = \begin{bmatrix}\mathbf{q}_1\\ \vdots\\ \mathbf{q}_N\end{bmatrix} = (\mathcal{F} + \Lambda)^{-1}\,\mathbf{h} \tag{17}$$

The individual quaternions can be extracted from the vector $\mathbf{Q}$ as

$$\mathbf{q}_1 = \begin{bmatrix}I_{4\times4} & 0_{4\times4} & \dots & 0_{4\times4}\end{bmatrix}\mathbf{Q} \tag{18a}$$

$$\mathbf{q}_2 = \begin{bmatrix}0_{4\times4} & I_{4\times4} & \dots & 0_{4\times4}\end{bmatrix}\mathbf{Q} \tag{18b}$$

$$\vdots$$

$$\mathbf{q}_N = \begin{bmatrix}0_{4\times4} & 0_{4\times4} & \dots & I_{4\times4}\end{bmatrix}\mathbf{Q} \tag{18c}$$

Defining $\mathcal{Y}(\boldsymbol{\lambda}) \equiv (\mathcal{F} + \Lambda)$ and using Eqs. (18) and (17) in Eq. (15) allows the constraints to be expressed as

$$f_j(\boldsymbol{\lambda}) = \mathbf{h}^T\mathcal{Y}(\boldsymbol{\lambda})^{-1}\mathcal{I}(j)\mathcal{Y}(\boldsymbol{\lambda})^{-1}\mathbf{h} - 1 = 0 \tag{19}$$

where $\mathcal{I}(j)$ is defined as the $[4N \times 4N]$ matrix of zeros with its (j,j) $[4 \times 4]$ sub-block equal to $I_{4\times4}$. As an example, when $N = 3$, $\mathcal{I}(2)$ would be equal to

$$\mathcal{I}(2) = \begin{bmatrix} 0_{4\times4} & 0_{4\times4} & 0_{4\times4} \\ 0_{4\times4} & I_{4\times4} & 0_{4\times4} \\ 0_{4\times4} & 0_{4\times4} & 0_{4\times4} \end{bmatrix} \tag{20}$$

The Jacobian is evaluated by utilizing the chain rule. The $(i,j)^{\text{th}}$ term in the Jacobian, $\mathcal{J}_{ij}$, is given by

$$\mathcal{J}_{ij} = \mathbf{h}^T \left(\frac{\partial \mathcal{Y}(\boldsymbol{\lambda})^{-1}}{\partial \lambda_j} \mathcal{I}(i) \mathcal{Y}(\boldsymbol{\lambda})^{-1} + \mathcal{Y}(\boldsymbol{\lambda})^{-1} \mathcal{I}(i) \frac{\partial \mathcal{Y}(\boldsymbol{\lambda})^{-1}}{\partial \lambda_j} \right) \mathbf{h} \tag{21}$$

The partial derivatives are easily computed using the following identity[21]

$$\frac{\partial Y^{-1}}{\partial x} = -Y^{-1} \frac{\partial Y}{\partial x} Y^{-1} \tag{22}$$

where Y is any $[m \times n]$ matrix and x is any scalar. Application of the identity leads to

$$\mathcal{J}_{ij} = \mathbf{h}^T \left(\mathcal{Y}^{-1}(\boldsymbol{\lambda}) \mathcal{I}(j) \mathcal{Y}^{-1}(\boldsymbol{\lambda}) \mathcal{I}(i) \mathcal{Y}^{-1}(\boldsymbol{\lambda}) + \mathcal{Y}^{-1}(\boldsymbol{\lambda}) \mathcal{I}(i) \mathcal{Y}^{-1}(\boldsymbol{\lambda}) \mathcal{I}(j) \mathcal{Y}^{-1}(\boldsymbol{\lambda}) \right) \mathbf{h} \tag{23}$$

which because of symmetry can be reduced to

$$\mathcal{J}_{ij} = 2\mathbf{h}^T \mathcal{Y}^{-1}(\boldsymbol{\lambda}) \; \mathcal{I}(j) \; \mathcal{Y}^{-1}(\boldsymbol{\lambda}) \; \mathcal{I}(i) \; \mathcal{Y}^{-1}(\boldsymbol{\lambda}) \mathbf{h} \tag{24}$$

1. Newton-Raphson Computational Considerations

Initialization of the Newton-Raphson iteration requires the initial guesses for the Lagrange multipliers to be sufficiently close to their true values. One method for attaining a good initial guess is to determine the average quaternions from the set of individual estimates. The j^{th} average quaternion is given by the eigenvector corresponding the the maximum eigenvalue of the matrix[22]

$$\mathcal{M}_j = -\sum_{i=1}^{n} \Xi(\mathbf{q}_{j,i}) \mathcal{P}_{jj_i} \Xi^T(\mathbf{q}_{j,i}) \tag{25}$$

where $\mathcal{P}_{jj_i}$ is the $(j,j)^{\text{th}}$ $[3 \times 3]$ sub-block of the inverse of P_i. Once the average quaternions are determined the Lagrange multipliers can be determined by first rearranging Eq. (12)

$$\Lambda \mathbf{Q}_{\text{avg}} = \mathbf{h} - \mathcal{F} \mathbf{Q}_{\text{avg}} \tag{26}$$

Equation (26) is then premultiplied by the following $[N \times 4N]$ matrix:

$$\mathcal{Q}_{\text{avg}} = \begin{bmatrix} \mathbf{q}_{1,\text{avg}}^T & 0_{1\times4} & \cdots & 0_{1\times4} \\ 0_{1\times4} & \mathbf{q}_{2,\text{avg}}^T & \cdots & 0_{1\times4} \\ \vdots & & \ddots & \vdots \\ 0_{1\times4} & \cdots & 0_{1\times4} & \mathbf{q}_{\text{N,avg}}^T \end{bmatrix} \tag{27}$$

Evaluating the multiplication leads values for the "average" Lagrange multipliers

$$\boldsymbol{\lambda}_{\text{avg}} = \mathcal{Q}_{\text{avg}} \left(\mathbf{h} - \mathcal{F} \mathbf{Q}_{\text{avg}} \right) \tag{28}$$

These values can then be used to initialize the Newton-Raphson iteration. In many cases, choosing starting values of $\boldsymbol{\lambda}(0) = \mathbf{0}$ is also a good approximation.[23]

The Newton-Raphson iteration uses a linear projection of the constraint errors in order to update the Lagrange multipliers at iteration. Because the constraint functions f_j are highly nonlinear, care must be taken so that the Newton update does not overstep the zero by an inordinate amount. While quasi-higher order iterative techniques exist, they are very complex and difficult to implement.[24,25] A more simplistic solution to this problem is to incorporate a contraction factor, $\varepsilon_k < 0$ into the Newton-Raphson iteration as

$$\boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k - \varepsilon_k \mathcal{J}^{-1} \mathbf{f} \big|_{\boldsymbol{\lambda}_k} \tag{29}$$

The proper selection of ε_k will ensure that the Newton-Raphson iteration will converge given an appropriate initialization. However, because the step size is reduced the rate of convergence will, in general, be much slower.

III. Three-Spacecraft Formation

The motivation behind the derivation of the CI algorithm presented is to apply the technique to the relative attitude estimation of a formation of three-spacecraft using a decentralized approach. In this scenario, two deputy spacecraft orbit about a chief spacecraft in simplified circular orbits. The formation can be seen in Fig. 2. Each spacecraft is equipped with two wide field-of-view (WFOV) optical line-of-sight (LOS) sensors housing focal plane detectors. In addition each spacecraft is equipped with inertial three-axis gyros. Communications links exist between neighboring spacecraft and have the necessary bandwidth to transmit measurement and state data. While unnecessary for the presented analysis, it is assumed that the chief has the required hardware to maintain knowledge about its inertial attitude and estimate its gyro bias.

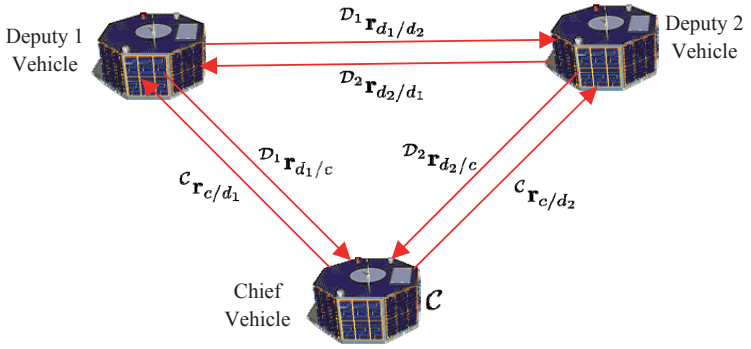


Figure 2. Three-Spacecraft Formation with LOS Definitions

In the following sections, an extended Kalman filter (EKF) for estimating the relative attitudes of the deputies with respect to the chief will be derived. To avoid confusion, we now introduce some of the notation that will be used. The body reference frame for the chief is denoted by $\mathcal{C}$ while the two deputy body frames are denoted by $\mathcal{D}_1$ and $\mathcal{D}_2$, respectively. LOS vectors are defined using the following convention: ${}^{z}\mathbf{r}_{x/y}$ is a unit vector originating at x and terminating at y, expressed in coordinate frame z. A LOS vector can be expressed in a different coordinate system by applying the proper coordinate transformation. This constitutes pre-multiplying a vector by the appropriate attitude matrix. For example, ${}^{y}\mathbf{r}_{x/y} = A_x^y \; {}^{x}\mathbf{r}_{x/y}$ where A_x^y is the attitude matrix which maps the x coordinate frame to y. For completion, we note that the attitude matrix is a proper orthogonal matrix and so has the following property, $A_x^y = (A_y^x)^T$. Similar to the attitude matrix, quaternions are given the notation $\mathbf{q}_x^y$ is the quaternion corresponding to the attitude matrix A_x^y. The attitude matrix can be written as the function of the quaternion by

$$A(\mathbf{q}) = \left(q_4^2 - \boldsymbol{\varrho}^T\boldsymbol{\varrho}\right) I_{3\times 3} + 2\boldsymbol{\varrho}\boldsymbol{\varrho}^T - 2q_4\left[\boldsymbol{\varrho}\times\right] \tag{30}$$

IV. Kalman Filtering for Spacecraft Relative Attitude Estimation

This section reviews the derivation of a multiplicative EKF for relative attitude estimation using LOS and gyro measurements. A more complete and detailed derivation of the presented material can be found in Refs. [12] and [26]. The goal of the EKF is to estimate the relative attitudes between the Chief and Deputy 1, $\mathbf{q}_{\mathcal{C}}^{\mathcal{D}_1}$, and between the Chief and Deputy 2, $\mathbf{q}_{\mathcal{C}}^{\mathcal{D}_2}$, as well as the gyro biases onboard the Deputy 1 and 2 spacecraft given by $\boldsymbol{\beta}_{d_1}$ and $\boldsymbol{\beta}_{d_2}$, respectively.

A. Kinematic Model

The multiplicative EKF[27] for attitude estimation uses a multiplicative representation of the error quaternion in order to calculate state errors and propagate the covariance. This form was first introduced in 1982 by Lefferts, Markley and Shuster.[19] The error quaternion between an estimated quaternion and its truth value is given by

$$\delta\mathbf{q} = \mathbf{q}\otimes\hat{\mathbf{q}}^{-1} \approx \begin{bmatrix}\frac{1}{2}\delta\boldsymbol{\alpha}\\ 1\end{bmatrix} = \begin{bmatrix}\Xi^T(\hat{\mathbf{q}})\mathbf{q}\\ \hat{\mathbf{q}}^T\mathbf{q}\end{bmatrix} \tag{31}$$

where $\delta\boldsymbol{\alpha}$ is the three-vector of the small roll, pitch and roll errors as discussed previously. For the case of relative quaternions, Ref. [12] has shown that the small angle error kinematics follow

$$\delta\dot{\boldsymbol{\alpha}}_{\mathcal{C}}^{\mathcal{D}_1} = -\left[\hat{\boldsymbol{\omega}}_{d_1}\times\right]\delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_1} + \delta\boldsymbol{\omega}_{d_1} - \hat{A}_{\mathcal{C}}^{\mathcal{D}_1}\delta\boldsymbol{\omega}_c \tag{32}$$

where $\hat{\boldsymbol{\omega}}_{d_1}$ is the estimate of the angular rate of the Deputy 1 spacecraft with respect to the inertial frame, $\delta\boldsymbol{\omega} \equiv \boldsymbol{\omega} - \hat{\boldsymbol{\omega}}$ is the error between the true and estimated angular rates. Assuming Farrenkopf's gyro model[28] the angular rate errors of the Deputy 1 spacecraft can be expressed as

$$\delta\boldsymbol{\omega}_{d_1} = -(\Delta\boldsymbol{\beta}_{d_1} + \boldsymbol{\eta}_{v_{d_1}}) \tag{33}$$

where $\Delta\boldsymbol{\beta}_{d_1} \equiv \boldsymbol{\beta}_{d_1} - \hat{\boldsymbol{\beta}}_{d_1}$ is the error between the true and estimated gyro biases, and $\boldsymbol{\eta}_{v_{d_1}}$ is a zero-mean Gaussian white noise process with covariance $Q_{v_{d_1}}$ (also stated as $\boldsymbol{\eta}_{v_{d_1}} \sim \mathcal{N}(\mathbf{0}, Q_{v_{d_1}})$). Assuming that the Chief gyro biases are known exactly, i.e. $\delta\boldsymbol{\omega}_c = \mathbf{0}$, and substituting Eq. (33) into (32) gives the following for the linearized small error kinematics

$$\delta\dot{\boldsymbol{\alpha}}_{\mathcal{C}}^{\mathcal{D}_1} = -\left[\hat{\boldsymbol{\omega}}_{d_1}\times\right]\delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_1} - \Delta\boldsymbol{\beta}_{d_1} - \boldsymbol{\eta}_{v_{d_1}} \tag{34}$$

The Deputy 1 gyro biases are driven by another zero-mean Gaussian white noise process

$$\dot{\boldsymbol{\beta}}_{d_1} = \boldsymbol{\eta}_{u_{d_1}}, \quad \boldsymbol{\eta}_{u_{d_1}} \sim \mathcal{N}(\mathbf{0}, Q_{u_{d_1}}) \tag{35}$$

The EKF however, assumes no kinematics on the gyro biases, $\dot{\hat{\boldsymbol{\beta}}} = \mathbf{0}$, and therefore Eq. (35) is also valid for the gyro bias error kinematics. Similar expressions to Eqs. (34) and (35) hold true for the Deputy 2 kinematics:

$$\delta\dot{\boldsymbol{\alpha}}_{\mathcal{C}}^{\mathcal{D}_2} = -\left[\hat{\boldsymbol{\omega}}_{d_2}\times\right]\delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_2} - \Delta\boldsymbol{\beta}_{d_2} - \boldsymbol{\eta}_{v_{d_2}} \tag{36a}$$

$$\Delta\dot{\boldsymbol{\beta}}_{d_2} = \boldsymbol{\eta}_{u_{d_2}}, \quad \boldsymbol{\eta}_{u_{d_2}} \sim \mathcal{N}(\mathbf{0}, Q_{u_{d_2}}) \tag{36b}$$

The linearized error kinematics are then expressed as the following linear time-varying state space model

$$\Delta\dot{\mathbf{x}}(t) = \mathcal{A}(\hat{\mathbf{x}}(t), t)\Delta\mathbf{x}(t) + \mathcal{B}(t)\mathbf{w}(t) \tag{37a}$$

$$\mathbf{x}(t) = \begin{bmatrix} \mathbf{q}_{\mathcal{C}}^{\mathcal{D}_1}(t) \\ \mathbf{q}_{\mathcal{C}}^{\mathcal{D}_2}(t) \\ \boldsymbol{\beta}_{d_1}(t) \\ \boldsymbol{\beta}_{d_2}(t) \end{bmatrix}, \quad \Delta\mathbf{x}(t) = \begin{bmatrix} \delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_1}(t) \\ \delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_2}(t) \\ \Delta\boldsymbol{\beta}_{d_1}(t) \\ \Delta\boldsymbol{\beta}_{d_2}(t) \end{bmatrix}, \quad \mathbf{w}(t) = \begin{bmatrix} \boldsymbol{\eta}_{v_{d_1}}(t) \\ \boldsymbol{\eta}_{u_{d_1}}(t) \\ \boldsymbol{\eta}_{v_{d_2}}(t) \\ \boldsymbol{\eta}_{u_{d_2}}(t) \end{bmatrix} \tag{37b}$$

and

$$\mathcal{A}(\hat{\mathbf{x}}(t), t) = \begin{bmatrix} -[\hat{\boldsymbol{\omega}}_{d_1}(t)\times] & 0_{3\times3} & -I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & -[\hat{\boldsymbol{\omega}}_{d_2}(t)\times] & 0_{3\times3} & -I_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \end{bmatrix} \tag{38a}$$

$$\mathcal{B}(t) = \begin{bmatrix} -I_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & -I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & I_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & I_{3\times3} \end{bmatrix} \tag{38b}$$

B. WFOV Sensor Model

The optical sensors located on each spacecraft determine a LOS vector from a set of focal plain detector measurements. The sensors work by measuring two angles ϑ and φ, which are the projections of the LOS vector onto the focal plane, $\mathbf{m} \equiv [\vartheta \;\; \varphi]^T$. The measurements, $\tilde{\mathbf{m}} = \mathbf{m} + \mathbf{w}_{\mathbf{m}}$, are corrupted by zero-mean Gaussian white noise where

$$\mathbf{w}_{\mathbf{m}} \sim \mathcal{N}(0, R_{\text{focal}}) \tag{39a}$$

$$R_{\text{focal}} = \frac{\sigma^2}{1 + d\left(\vartheta^2 + \varphi^2\right)} \begin{bmatrix} \left(1 + d\vartheta^2\right) & \left(d\vartheta\varphi\right)^2 \\ \left(d\vartheta\varphi\right)^2 & \left(1 + d\varphi^2\right) \end{bmatrix} \tag{39b}$$

where σ is the standard deviation of the noise associated with $\mathbf{m}$ and d is on the order of 1. Assuming a focal length of unity, the true and measured LOS's can be determined from the focal plane measurements as

$$^{\mathcal{S}}\mathbf{r} = \frac{1}{\sqrt{1+\mathbf{m}^T\mathbf{m}}}\begin{bmatrix}\mathbf{m}\\1\end{bmatrix} \tag{40a}$$

$$^{\mathcal{S}}\tilde{\mathbf{r}} = \frac{1}{\sqrt{1+\tilde{\mathbf{m}}^T\tilde{\mathbf{m}}}}\begin{bmatrix}\tilde{\mathbf{m}}\\1\end{bmatrix} \tag{40b}$$

where the superscript $\mathcal{S}$ denotes the sensor coordinate frame. Reference [29] has shown that the covariance of the LOS measurement can be constructed using a first order Taylor-Series expansion about the focal plane axis:

$$^{\mathcal{S}}\mathcal{R} = \Omega R_{\text{focal}}\Omega^T \tag{41a}$$

$$\Omega = \frac{\partial\ ^{\mathcal{S}}\mathbf{r}}{\partial\mathbf{m}} = \frac{1}{\sqrt{1+\mathbf{m}^T\mathbf{m}}}\begin{bmatrix}1&0\\0&1\\0&0\end{bmatrix} - \frac{1}{1+\mathbf{m}^T\mathbf{m}}\ ^{s}\mathbf{r}\ \mathbf{m}^T \tag{41b}$$

Similar to the QUEST Measurement Model,[30] the WFOV covariance is singular with an eigenpair $(0,\ ^{\mathcal{S}}\mathbf{r})$. For filtering, the measurement covariance is required to be non-singular. This is achieved through a rank-one update of the covariance matrix

$$^{\mathcal{S}}R =^{\mathcal{S}}\mathcal{R} + \frac{1}{2}\text{Tr}\left[^{\mathcal{S}}\mathcal{R}\right]\ ^{\mathcal{S}}\mathbf{r}^{\mathcal{S}}\mathbf{r}^T \tag{42}$$

where Tr[·] is the matrix trace operator. Finally the covariance from Eq. (42) must be rotated to the appropriate vehicle coordinate frame via

$$^{B}R = \left(A_B^{\mathcal{S}}\right)^T\ ^{\mathcal{S}}R\ A_B^{\mathcal{S}} \tag{43}$$

The body frame to sensor attitude, $A_B^{\mathcal{S}}$, will be known prior to launch or can be calibrated during on-orbit checkout. With the above results, the six LOS measurements along with their associated error covariance can be expressed as

$$^{\mathcal{C}}\tilde{\mathbf{r}}_{c/d_1} =\ ^{\mathcal{C}}\mathbf{r}_{c/d_1} + \mathbf{v}_{c/d_1} \quad \mathbf{v}_{c/d_1} \sim \mathcal{N}(\mathbf{0}, R_{c/d_1}) \tag{44a}$$

$$^{\mathcal{C}}\tilde{\mathbf{r}}_{c/d_2} =\ ^{\mathcal{C}}\mathbf{r}_{c/d_2} + \mathbf{v}_{c/d_2} \quad \mathbf{v}_{c/d_2} \sim \mathcal{N}(\mathbf{0}, R_{c/d_2}) \tag{44b}$$

$$^{\mathcal{D}_1}\tilde{\mathbf{r}}_{d_1/c} =\ ^{\mathcal{D}_1}\mathbf{r}_{d_1/c} + \mathbf{v}_{d_1/c} \quad \mathbf{v}_{d_1/c} \sim \mathcal{N}(\mathbf{0}, R_{d_1/c}) \tag{44c}$$

$$^{\mathcal{D}_1}\tilde{\mathbf{r}}_{d_1/d_2} =\ ^{\mathcal{D}_1}\mathbf{r}_{d_1/d_2} + \mathbf{v}_{d_1/d_2} \quad \mathbf{v}_{d_1/d_2} \sim \mathcal{N}(\mathbf{0}, R_{d_1/d_2}) \tag{44d}$$

$$^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/c} =\ ^{\mathcal{D}_2}\mathbf{r}_{d_2/c} + \mathbf{v}_{d_2/c} \quad \mathbf{v}_{d_2/c} \sim \mathcal{N}(\mathbf{0}, R_{d_2/c}) \tag{44e}$$

$$^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/d_1} =\ ^{\mathcal{D}_2}\mathbf{r}_{d_2/d_1} + \mathbf{v}_{d_2/d_1} \quad \mathbf{v}_{d_2/d_1} \sim \mathcal{N}(\mathbf{0}, R_{d_2/d_1}) \tag{44f}$$

C. Measurement Model

Clearly, the six LOS vectors cannot be independent of one another. Using the relative attitudes we can formulate the three vehicle-paired truths

$$^{\mathcal{D}_1}\mathbf{r}_{d_1/c} = -A_{\mathcal{C}}^{\mathcal{D}_1}\ ^{\mathcal{C}}\mathbf{r}_{c/d_1} \tag{45a}$$

$$^{\mathcal{D}_2}\mathbf{r}_{d_2/c} = -A_{\mathcal{C}}^{\mathcal{D}_2}\ ^{\mathcal{C}}\mathbf{r}_{c/d_2} \tag{45b}$$

$$^{\mathcal{D}_1}\mathbf{r}_{d_1/d_2} = -A_{\mathcal{D}_2}^{\mathcal{D}_1}\ ^{\mathcal{D}_2}\mathbf{r}_{d_2/d_1} \tag{45c}$$

so that the measurement vector and output vectors are

$$\tilde{\mathbf{y}} = \begin{bmatrix}^{\mathcal{D}_1}\tilde{\mathbf{r}}_{d_1/c}\\ ^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/c}\\ ^{\mathcal{D}_1}\tilde{\mathbf{r}}_{d_1/d_2}\end{bmatrix}, \quad \mathbf{h}(\mathbf{x}_k) = \begin{bmatrix}-A_{\mathcal{C}}^{\mathcal{D}_1}\ ^{\mathcal{C}}\tilde{\mathbf{r}}_{c/d_1}\\ -A_{\mathcal{C}}^{\mathcal{C}\mathcal{D}_2}\ ^{\mathcal{C}}\tilde{\mathbf{r}}_{c/d_2}\\ -A_{\mathcal{D}_2}^{\mathcal{D}_1}\ ^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/d_1}\end{bmatrix} \tag{46}$$

While the true attitude $A_{\mathcal{C}}^{\mathcal{D}_1}$ is unknown, it can be expressed in terms of the estimated attitude and error quaternion by utilizing the quaternion property of successive rotations:

$$A(\mathbf{q}) = A(\delta\mathbf{q} \otimes \hat{\mathbf{q}}) = A(\delta\mathbf{q})A(\hat{\mathbf{q}}) \tag{47}$$

The first order approximation to of the attitude matrix of the error quaternion is given by

$$A(\delta\mathbf{q}) \approx I_{3\times 3} - [\delta\boldsymbol{\alpha}\times] \tag{48}$$

Substituting Eqs. (47) and (48) into the first output term of Eq. (46) gives

$$\mathbf{h}_1(\hat{\mathbf{x}}_k) = -\hat{A}_{\mathcal{C}}^{\mathcal{D}_1}\ {}^{\mathcal{C}}\tilde{\mathbf{r}}_{c/d_1} + \left[\delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_1}\times\right]\hat{A}_{\mathcal{C}}^{\mathcal{D}_1}\ {}^{\mathcal{C}}\tilde{\mathbf{r}}_{c/d_1} \tag{49}$$

Noting that $[\mathbf{a}\times]\mathbf{c} = -[\mathbf{c}\times]\mathbf{a}$, Eq. (49) can be rearranged as

$$\mathbf{h}_1(\hat{\mathbf{x}}_k) = -\hat{A}_{\mathcal{C}}^{\mathcal{D}_1}\ {}^{\mathcal{C}}\tilde{\mathbf{r}}_{c/d_1} - \left[\left(\hat{A}_{\mathcal{C}}^{\mathcal{D}_1}\ {}^{\mathcal{C}}\tilde{\mathbf{r}}_{c/d_1}\right)\times\right]\delta\boldsymbol{\alpha}_{\mathcal{C}}^{D_1} \tag{50}$$

A similar expression holds true for $\mathbf{h}_2(\hat{\mathbf{x}}_k)$

$$\mathbf{h}_2(\hat{\mathbf{x}}_k) = -\hat{A}_{\mathcal{C}}^{\mathcal{D}_2}\ {}^{\mathcal{C}}\tilde{\mathbf{r}}_{c/d_2} - \left[\left(\hat{A}_{\mathcal{C}}^{\mathcal{D}_2}\ {}^{\mathcal{C}}\tilde{\mathbf{r}}_{c/d_2}\right)\times\right]\delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_2} \tag{51}$$

The third term, $\mathbf{h}_3(\hat{\mathbf{x}}_k)$ is a little more complicated due to the coupling of the attitudes. This term can be expressed as

$$\mathbf{h}_3(\mathbf{x}_k) = -A_{\mathcal{C}}^{\mathcal{D}_1}(A_{\mathcal{C}}^{\mathcal{D}_2})^T\ {}^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/d_1} \tag{52}$$

Using the first order approximation for each of the attitude matrices yields

$$\mathbf{h}_3(\hat{\mathbf{x}}_k) = -\left(I - \left[\delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_1}\times\right]\right)\hat{A}_{\mathcal{C}}^{\mathcal{D}_1}(\hat{A}_{\mathcal{C}}^{\mathcal{D}_2})^T\left(I + \left[\delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_2}\times\right]\right)\ {}^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/d_1} \tag{53}$$

where we have also used the fact that $[\mathbf{a}\times]^T = -[\mathbf{a}\times]$. Expanding results in

$$\begin{aligned}\mathbf{h}_3(\hat{\mathbf{x}}_k) = -\Big(&\hat{A}_{\mathcal{C}}^{\mathcal{D}_1}(\hat{A}_{\mathcal{C}}^{\mathcal{D}_2})^T\left[\delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_2}\times\right] - \left[\delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_1}\times\right]\hat{A}_{\mathcal{C}}^{\mathcal{D}_1}(\hat{A}_{\mathcal{C}}^{\mathcal{D}_2})^T \\ &+ \hat{A}_{\mathcal{C}}^{\mathcal{D}_1}(\hat{A}_{\mathcal{C}}^{\mathcal{D}_2})^T - \left[\delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_1}\times\right]\hat{A}_{\mathcal{C}}^{\mathcal{D}_1}(\hat{A}_{\mathcal{C}}^{\mathcal{D}_2})^T\left[\delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_2}\times\right]\Big)\ {}^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/d_1}\end{aligned} \tag{54}$$

After distributing ${}^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/d_1}$, neglecting the second order term and rearranging we end up with

$$\begin{aligned}\mathbf{h}_3(\hat{\mathbf{x}}_k) = -\left[\left(\hat{A}_{\mathcal{C}}^{\mathcal{D}_1}(\hat{A}_{\mathcal{C}}^{\mathcal{D}_2})^T\ {}^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/d_1}\right)\times\right]\delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_1} - \hat{A}_{\mathcal{C}}^{\mathcal{D}_1}(\hat{A}_{\mathcal{C}}^{\mathcal{D}_2})^T\left[\ {}^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/d_1}\times\right]\delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_2} \\ - \hat{A}_{\mathcal{C}}^{\mathcal{D}_1}(\hat{A}_{\mathcal{C}}^{\mathcal{D}_2})^T\ {}^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/d_1}\end{aligned} \tag{55}$$

The sensitivity matrix can then be easily calculated from Eqs. (50), (51) and (55):

$$H(\hat{\mathbf{x}}_k) = \begin{bmatrix} -\left[\left(\hat{A}_{\mathcal{C}}^{\mathcal{D}_1}\ {}^{\mathcal{C}}\tilde{\mathbf{r}}_{c/d_1}\right)\times\right] & 0_{3\times 3} & 0_{3\times 3} & 0_{3\times 3} \\ 0_{3\times 3} & -\left[\left(\hat{A}_{\mathcal{C}}^{\mathcal{D}_2}\ {}^{\mathcal{C}}\tilde{\mathbf{r}}_{c/d_2}\right)\times\right] & 0_{3\times 3} & 0_{3\times 3} \\ -\left[\left(\hat{A}_{\mathcal{C}}^{\mathcal{D}_1}(\hat{A}_{\mathcal{C}}^{\mathcal{D}_2})^T\ {}^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/d_1}\right)\times\right] & \hat{A}_{\mathcal{C}}^{\mathcal{D}_1}(\hat{A}_{\mathcal{C}}^{\mathcal{D}_2})^T\left[\ {}^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/d_1}\times\right] & 0_{3\times 3} & 0_{3\times 3} \end{bmatrix} \tag{56}$$

D. Kalman Update

The update and propagation equations for the relative attitude Kalman filter are equivalent as those used in a standard attitude estimation Kalman filter. They are presented here for completeness. The Kalman gain, K_k, is defined as

$$K_k = P_k^- H^T(\hat{\mathbf{x}}_k^-)\left[H(\hat{\mathbf{x}}_k^-)P_k^- H^T(\hat{\mathbf{x}}_k^-) + R\right]^{-1} \tag{57}$$

where R is the measurement noise covariance matrix. The measurement noise covariance matrix is block diagonal, $R = \text{diag}\left[{}^{\mathcal{D}_1}R_{d_1/c},\ {}^{\mathcal{D}_2}R_{d_2/c},\ {}^{\mathcal{D}_1}R_{d_1/d_2}\right]$. After determining the updated error states

$$\Delta\hat{\mathbf{x}}_k^+ = K_k[\tilde{\mathbf{y}} - \mathbf{h}(\hat{\mathbf{x}}_k^-)] \tag{58}$$

the quaternion and gyro biases are updated via

$$\hat{\mathbf{q}}_k^+ = \hat{\mathbf{q}}_k^- + \frac{1}{2}\Xi(\hat{\mathbf{q}}_k^-)\delta\boldsymbol{\alpha}_k^+ \tag{59a}$$

$$\hat{\boldsymbol{\beta}}_k^+ = \hat{\boldsymbol{\beta}}_k^- + \Delta\hat{\boldsymbol{\beta}}_k^+ \tag{59b}$$

The covariance update is given by

$$P_k^+ = [I - K_k H(\hat{\mathbf{x}}_k^-)]P_k^- \tag{60}$$

E. Discrete Propagation

The state and covariance from the relative attitude Kalman filter can be efficiently propagated in a discrete manner assuming that the spacecraft angular rates are constant over an interval Δt. The relative quaternions propagate as[12]

$$\hat{\mathbf{q}}_{\mathcal{C}}^{\mathcal{D}_{1,2}}\Big|_{k+1}^- = \bar{\Omega}\left(\hat{\boldsymbol{\omega}}_{d_{1,2}}\big|_k^+\right)\bar{\Gamma}\left(\hat{\boldsymbol{\omega}}_c|_k^+\right)\hat{\mathbf{q}}_{\mathcal{C}}^{\mathcal{D}_{1,2}}\Big|_k^+ \tag{61}$$

where

$$\bar{\Omega}(\boldsymbol{\omega}_k) = \begin{bmatrix} \cos\left(\frac{1}{2}||\boldsymbol{\omega}_k||\Delta t\right) I_{3\times 3} - [\boldsymbol{\psi}_k\times] & \boldsymbol{\psi}_k \\ -\boldsymbol{\psi}_k^T & \cos\left(\frac{1}{2}||\boldsymbol{\omega}_k||\Delta t\right) \end{bmatrix} \tag{62a}$$

$$\bar{\Gamma}(\boldsymbol{\omega}_k) = \begin{bmatrix} \cos\left(\frac{1}{2}||\boldsymbol{\omega}_k||\Delta t\right) I_{3\times 3} - [\boldsymbol{\psi}_k\times] & -\boldsymbol{\psi}_k \\ \boldsymbol{\psi}_k^T & \cos\left(\frac{1}{2}||\boldsymbol{\omega}_k||\Delta t\right) \end{bmatrix} \tag{62b}$$

$$\boldsymbol{\psi}_k = \frac{\sin\left(\frac{1}{2}||\boldsymbol{\omega}_k||\Delta t\right)\boldsymbol{\omega}_k}{||\boldsymbol{\omega}_k||} \tag{62c}$$

A numerical solution allows discrete propagation of the covariance matrix.[31] First define the $2n \times 2n$ matrix

$$\mathcal{K} = \begin{bmatrix} -\mathcal{A} & \mathcal{B}Q\mathcal{B}^T \\ 0_{n\times n} & \mathcal{A}^T \end{bmatrix} \tag{63}$$

where Q is the continuous time process noise covariance matrix, $Q = E\{\mathbf{w}\,\mathbf{w}^T\}$ and $E\{\cdot\}$ is the expectation operator. The matrix exponential of $\mathcal{K}$ is then computed and blocked as follows

$$\mathcal{L} = e^{\mathcal{K}} = \begin{bmatrix} \mathcal{L}_{11} & \Phi_k^{-1}\mathcal{Q}_k \\ 0_{n\times n} & \Phi_k^T \end{bmatrix} \tag{64}$$

The covariance then propagates as

$$P_{k+1}^- = \Phi_k P_k^+ \Phi_k^T + \mathcal{Q}_k \tag{65}$$

The summary of the relative attitude estimation EKF can be found in Table 1.

F. Decentralized Kalman Filtering

The communications links between two neighboring spacecraft only transmit measurement and state data pertaining to these two spacecraft only. Thus, information regarding the inter-neighbor link is unavailable. As an example, consider the Chief spacecraft with data links between itself and the two deputies. From these links the Chief receives the measurements ${}^{\mathcal{D}_1}\tilde{\mathbf{r}}_{d_1/c}$ and $\tilde{\boldsymbol{\omega}}_{d_1}$ from Deputy 1 and ${}^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/c}$ and $\tilde{\boldsymbol{\omega}}_{d_2}$ from Deputy 2, while having no knowledge of the measurements ${}^{\mathcal{D}_1}\tilde{\mathbf{r}}_{d_1/d_2}$ and ${}^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/d_1}$. To accommodate the missing measurements the EKF displayed in Table 1 must be slightly altered. Table 2 shows the necessary alterations in order for the EKF to accommodate the missing measurements. Similar changes are made for the filters running aboard the Deputy 1 and Deputy 2 spacecraft after omitting the appropriate missing measurements.

While each filter processes its own subset of the available measurements, they also utilize common information. By using the same gyro measurements within each filter, as well as some of the LOS's, the state estimates from each filter will be correlated. Improved state estimates are obtained by fusing the individual estimates with the CI algorithm as described in Section II.

Table 1. Extended Kalman Filter for Spacecraft Relative Attitude Estimation using LOS vectors

Initialize

$$\hat{\mathbf{q}}_{\mathcal{C}}^{\mathcal{D}_1}(t_0) = \hat{\mathbf{q}}_{\mathcal{C}_0}^{\mathcal{D}_1} \ , \ \hat{\mathbf{q}}_{\mathcal{C}}^{\mathcal{D}_2}(t_0) = \hat{\mathbf{q}}_{\mathcal{C}_0}^{\mathcal{D}_2} \ , \ \hat{\boldsymbol{\beta}}_{d_1} = \mathbf{0} \ , \ \hat{\boldsymbol{\beta}}_{d_2} = \mathbf{0} \ , \ P(t_0) = P_0$$

$$\mathbf{x}(t) = \begin{bmatrix} \mathbf{q}_{\mathcal{C}}^{\mathcal{D}_1}(t) \\ \mathbf{q}_{\mathcal{C}}^{\mathcal{D}_2}(t) \\ \boldsymbol{\beta}_{d_1}(t) \\ \boldsymbol{\beta}_{d_2}(t) \end{bmatrix}, \ \Delta\mathbf{x}(t) = \begin{bmatrix} \delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_1}(t) \\ \delta\boldsymbol{\alpha}_{\mathcal{C}}^{\mathcal{D}_2}(t) \\ \Delta\boldsymbol{\beta}_{d_1}(t) \\ \Delta\boldsymbol{\beta}_{d_2}(t) \end{bmatrix}$$

Gain

$$K_k = P_k^- H^T(\hat{\mathbf{x}}_k^-) \left[H(\hat{\mathbf{x}}_k^-) P_k^- H^T(\hat{\mathbf{x}}_k^-) + R \right]^{-1}$$

$$H(\hat{\mathbf{x}}_k) = \begin{bmatrix} -\left[\left(\hat{A}_{\mathcal{C}}^{\mathcal{D}_1} \ {}^{\mathcal{C}}\tilde{\mathbf{r}}_{c/d_1} \right) \times \right] & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & -\left[\left(\hat{A}_{\mathcal{C}}^{\mathcal{D}_2} \ {}^{\mathcal{C}}\tilde{\mathbf{r}}_{c/d_2} \right) \times \right] & 0_{3\times3} & 0_{3\times3} \\ \left[\left(\hat{A}_{\mathcal{C}}^{\mathcal{D}_1} (\hat{A}_{\mathcal{C}}^{\mathcal{D}_2})^T \ {}^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/d_1} \right) \times \right] & -\hat{A}_{\mathcal{C}}^{\mathcal{D}_1} (\hat{A}_{\mathcal{C}}^{\mathcal{D}_2})^T \left[\ {}^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/d_1} \times \right] & 0_{3\times3} & 0_{3\times3} \end{bmatrix} \Bigg|_{t_k}$$

Update

$$P_k^+ = [I - K_k H(\hat{\mathbf{x}}_k^-)] P_k^-$$

$$\Delta\hat{\mathbf{x}}_k^+ = K_k [\tilde{\mathbf{y}} - \mathbf{h}(\hat{\mathbf{x}}_k^-)]$$

$$\mathbf{h}(\mathbf{x}_k) = \begin{bmatrix} -A_{\mathcal{C}}^{\mathcal{D}_1} \ {}^{\mathcal{C}}\tilde{\mathbf{r}}_{c/d_1} \\ -A_{\mathcal{C}}^{\mathcal{D}_2} \ {}^{\mathcal{C}}\tilde{\mathbf{r}}_{c/d_2} \\ -A_{\mathcal{D}_2}^{\mathcal{D}_1} \ {}^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/d_1} \end{bmatrix}, \ \tilde{\mathbf{y}} = \begin{bmatrix} {}^{\mathcal{D}_1}\tilde{\mathbf{r}}_{d_1/c} \\ {}^{\mathcal{D}_2}\tilde{\mathbf{r}}_{d_2/c} \\ {}^{\mathcal{D}_1}\tilde{\mathbf{r}}_{d_1/d_2} \end{bmatrix}$$

$$\hat{\mathbf{q}}_k^+ = \hat{\mathbf{q}}_k^- + \tfrac{1}{2}\Xi(\hat{\mathbf{q}}_k^-)\delta\hat{\boldsymbol{\alpha}}_k^+, \text{ re-normalize quaternions}$$

$$\hat{\boldsymbol{\beta}}_k^+ = \hat{\boldsymbol{\beta}}_k^- + \Delta\hat{\boldsymbol{\beta}}_k^+$$

Propagate

$$\hat{\mathbf{q}}_{k+1}^- = \bar{\Omega}(\hat{\boldsymbol{\omega}}_{d_k})\bar{\Gamma}(\hat{\boldsymbol{\omega}}_{c_k})\hat{\mathbf{q}}_k^+ \ , \quad \hat{\boldsymbol{\beta}}_{k+1}^- = \hat{\boldsymbol{\beta}}_k^+ \ , \quad P_{k+1}^- = \Phi_k P_k^+ \Phi_k^T + \mathcal{Q}_k$$

V. Formation Flying Simulation Results

In this section the results of a simple simulation involving a three-spacecraft formation are shown. Three spacecraft equipped with LOS sensors and gyros are arranged as shown in Fig. 2. The deputies are assumed to be in circular orbit about the chief in a stationary formation with $\omega_{\mathrm{f}} = 0.011\frac{\mathrm{rad}}{\mathrm{s}}$. The true body angular rates of the three spacecraft are assumed arbitrarily to be

$$\boldsymbol{\omega}_c = \begin{bmatrix} 0 \\ 0 \\ \omega_{\mathrm{f}} \end{bmatrix}, \qquad \boldsymbol{\omega}_{d_1}(t) = \boldsymbol{\omega}_{d_2}(t) = \begin{bmatrix} 0 \\ 0.05\omega_{\mathrm{f}}\sin(0.1t) \\ \omega_{\mathrm{f}} + 0.25\omega_{\mathrm{f}}\cos(0.1t) \end{bmatrix}$$

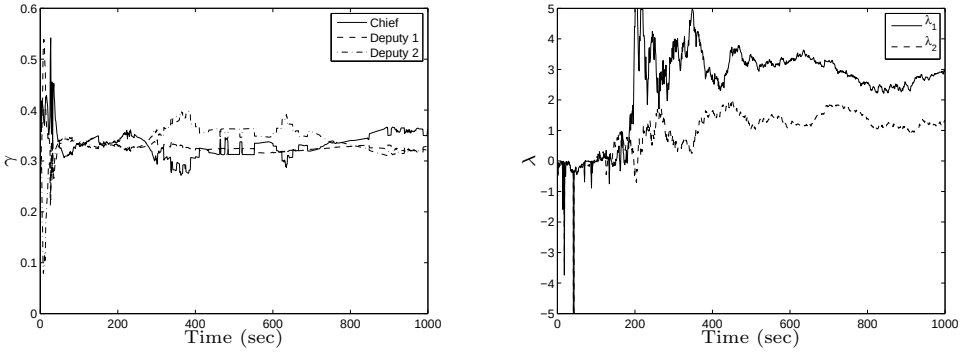
The WFOV-LOS sensors aboard the Chief spacecraft has a noise standard deviation of 3.5 arc-sec, while the Deputies each have WFOV-LOS sensors with noise standard deviations of 35 arc-sec. It is assumed that, once the LOS establish contact with the neighboring spacecraft's beacon, that the connection is never lost. Local EKFs as described in Sec. IV are run onboard each spacecraft resulting in a full state estimate and associated covariance.

The CI algorithm was used to fuse the local state estimates. The weights γ_i were determined using a CVX bounded optimization routine[32] while minimizing the trace of the fused covariance matrix. The values of the weights can be seen in Fig. 3(a). Because of stiff numeric difficulties associated with the Newton-Raphson iteration, the CI algorithm was carried out using multi-precision computing. Multi-precision computing was done in C++ and interfaced with MATLAB using *mex* files.[33] Multi-precision computing allows the user to select an arbitrary number of bits with which to compute and store variables. For this simulation an arbitrary number of 500 bits was selected for the multi-precision variables. It was found that the Newton-

Table 2. Changes to Global EKF to act as Chief Local EKF

Gain	$H_c(\hat{\mathbf{x}}_k) = \begin{bmatrix} -\left[\left(\hat{A}_c^{d_1}\ {}^c\tilde{\mathbf{r}}_{c/d_1}\right)\times\right] & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & -\left[\left(\hat{A}_c^{d_2}\ {}^c\tilde{\mathbf{r}}_{c/d_2}\right)\times\right] & 0_{3\times3} & 0_{3\times3} \end{bmatrix}\Bigg\vert_{t_k}$
Update	$\mathbf{h}_c(\mathbf{x}_k) = \begin{bmatrix} -A_c^{d_1}\ {}^c\tilde{\mathbf{r}}_{c/d_1} \\ -A_c^{d_2}\ {}^c\tilde{\mathbf{r}}_{c/d_2} \end{bmatrix}$, $\tilde{\mathbf{y}}_c = \begin{bmatrix} {}^{d_1}\tilde{\mathbf{r}}_{d_1/c} \\ {}^{d_2}\tilde{\mathbf{r}}_{d_2/c} \end{bmatrix}$

Raphson iteration converged when $\varepsilon_k = \mathcal{O}\left(\frac{1}{100\tau}\right)$ where $\tau = \text{argmax}\left(abs[\mathcal{J}^{-1}\mathbf{f}]\right)$. It should be noted that neither the number of multi-precision bits nor the contraction factor ε_k were optimized in order to find the minimum values necessary for computation. All iterations were initialized with $\lambda_1(0) = \lambda_2(0) = 0$ and given a stopping criteria of $\|\mathbf{f}\| \leq 10^{-4}$. The determined values of the Lagrange multiplier, which can be seen in Fig. 3(b), were generally obtained within 100 iterations.



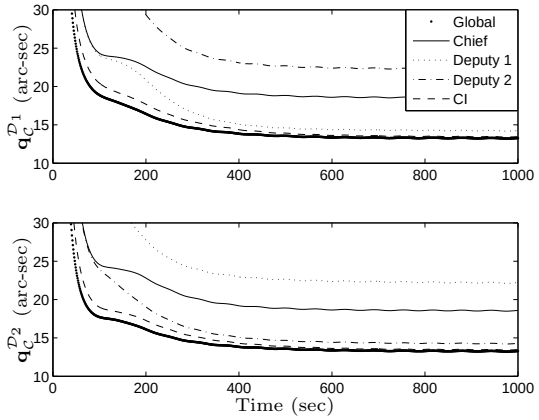
(a) CI Weights (b) Lagrange Multipliers

Figure 3. CI weights and Lagrange Multiplier Values

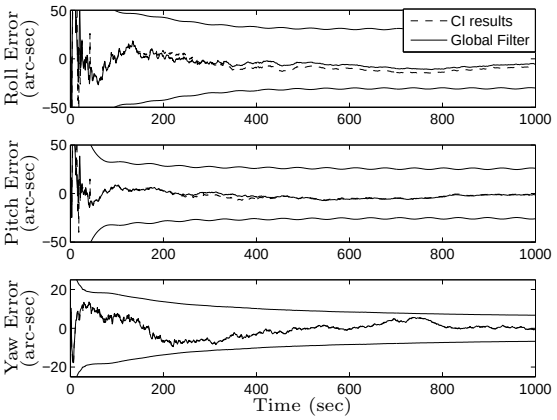
The 3σ attitude bounds for each of the local filters and CI algorithm are found in Fig. 4(a). Note how the results from the CI algorithm quickly approach the results from a global EKF processing all LOS and gyro measurements. The Deputy 1 local filter performs the best when estimating $\mathbf{q}_{\mathcal{C}}^{\mathcal{D}_1}$ but is the poorest in estimating $\mathbf{q}_{\mathcal{C}}^{\mathcal{D}_2}$. This is because the Deputy 1 filter has independent information regarding $\mathbf{q}_{\mathcal{C}}^{\mathcal{D}_1}$ but only coupled information about $\mathbf{q}_{\mathcal{C}}^{\mathcal{D}_2}$. The converse is true for the Deputy 2 filter. Since the Chief filter has only independent knowledge of $\mathbf{q}_{\mathcal{C}}^{\mathcal{D}_1}$ and $\mathbf{q}_{\mathcal{C}}^{\mathcal{D}_2}$, it provides the median estimate of each. Also because the Chief filter does not view the attitudes as coupled, it predicts zero cross-correlations between the two. As can be seen, the CI algorithm utilizes all of the available knowledge to attain better estimates than any of the local filters. The 3-axis attitude errors for $\mathbf{q}_{\mathcal{C}}^{\mathcal{D}_1}$ and $\mathbf{q}_{\mathcal{C}}^{\mathcal{D}_2}$ can be found in Figs. 4(b) and 4(c), respectively, and show that the error estimated are well contained in the 3σ bounds. These plots show the 3-axis attitude errors from the CI algorithm juxtaposed with those expected from the global filter. Note that for most of the simulation the two are indiscernible. Although not shown here, swift convergence was also achieved for the deputy gyro bias estimates.

VI. Conclusions

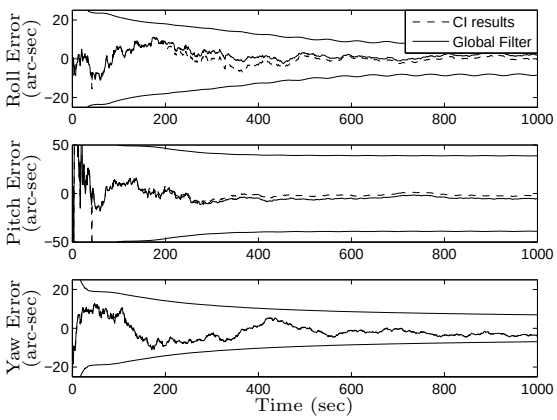
In order to overcome the obstacle of unknown correlations in decentralized estimation systems, the CI algorithm can be used to consistently fuse state estimates. When applying the CI algorithm to attitude estimation, there is an increased level of complexity due to equality constraints within the state vector.



(a) 3σ Attitude Bounds



(b) $\mathbf{q}_{\mathcal{C}}^{\mathcal{D}_1}$ 3-axis Attitude Errors and 3σ Bounds



(c) $\mathbf{q}_{\mathcal{C}}^{\mathcal{D}_2}$ 3-axis Attitude Errors and 3σ Bounds

Figure 4. Relative Attitude Estimation Results

A novel approach uses the method of Lagrange multipliers to handle these state constraints. When state vectors contain more than one constraint, an analytic solution is not possible. The development of a multivariate Newton-Raphson iteration provides a means for fusing state vectors containing an arbitrary number of equality constraints. However, because of persistent numeric problems, multi-precision computing was necessary in order to obtain convergence of the Newton-Raphson iteration.

The benefit of the CI algorithm is showcased via a simple formation flying simulation. Using LOS and gyro measurements, each spacecraft estimates the relative attitudes within the formation using EKFs. These estimates are then combined using the CI algorithm in order to obtain estimates better than any of the individual estimates. The CI results proved nearly indiscernible from the global, optimal solution.

References

[1]Cao, Y. U., Fukunaga, A. S., and Kahng, A. B., "Cooperative Mobile Robotics: Antecedents and Directions," *Autonomous Robots*, Vol. 4, No. 1, March 1997, pp. 7–27.

[2]Bhatt, R. M., Tang, C. P., and Krovi, V., "Formation Optimization for a Fleet of Wheeled Mobile Robots - A Geometric Approach," *Robotics and Autonomous Systems*, 2008.

[3]Stilwell, D. and Bishop, B., "Platoons of Underwater Vehicles," *IEEE Control Systems Magazine*, Vol. 20, No. 6, December 2000, pp. 45–52.

[4]Fosbury, A. M. and Crassidis, J. L., "Kalman Filtering for Relative Inertial Navigation of Uninhabited Air Vehicles," *AIAA Guidance, Navigation, and Control Conference*, Keystone, CO, August 2006.

[5]Fosbury, A. M. and Crassidis, J. L., "Relative Navigation of Air Vehicles," *Journal of Guidance, Control, and Dynamics*, Vol. 2, No. 5, 2008, pp. 824–834.

[6]Sholomitsky, G. B., Prilulsky, O. F., and Rodin, V. G., "Infra-Red Space Interferometer," *28th International Astronautical Congress of the International Astronautical Federation*, Prague, Czechoslovakia, Sept. 1977, IAF-77-68.

[7]Carpenter, J., Leitner, J., Folta, D., and Burns, R., "Benchmark Problems for Spacecraft Formation Flying Missions," *AIAA Guidance, Navigation, and Control Conference*, Austin, TX, August 2003.

[8]Leitner, J. and et al, "Formation Flight in Space: Distributed Spacecraft Systems Develop New GPS Capabilities," *GPS World*, February 2001.

[9]Carpenter, J. R., "Decentralized Control of Satellite Formations," *International Journal of Robust and Nonlinear Control*, Vol. 12, No. 2-3, 2002, pp. 141–161.

[10]Scharf, D. P., Hadaegh, F. Y., and Ploen, S. R., "A Survey of Spacecraft Formation Flying Guidance and Control (Part I): Guidance," *Proceedings of the American Control Conference*, Vol. 2, Denver, CO, June 2004, pp. 1733–1739.

[11]Scharf, D. P., Hadaegh, F. Y., and Ploen, S. R., "A Survey of Spacecraft Formation Flying Guidance and Control (Part II): Control," *Proceedings of the American Control Conference*, Vol. 4, Boston, MA, June-July 2004, pp. 2976–2985.

[12]Kim, S.-G., Crassidis, J. L., Cheng, Y., Fosbury, A. M., and Junkins, J. L., "Kalman Filtering for Relative Spacecraft Attitude and Position Estimation," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 1, Jan.-Feb. 2007, pp. 133–143.

[13]Andrle, M. S., Hyun, B., Crassidis, J. L., Linares, R., and Cheng, Y., "Deterministic Relative Attitude Determination of Three-Vehicle Formations," *Journal of Guidance, Control, and Dynamics*, 2008, to appear.

[14]Julier, S. J. and Uhlmann, J. K., "A Non-Divergent Estimation Algorithm in the Presence of Unknown Correlations," *Proceedings of the American Control Conference*, Vol. 4, Albuquerque, NM, June 1997, pp. 2369–2373.

[15]Uhlmann, J. K., "General Data Fusion for Estimates with Unknown Cross Covariances," *SPIE AeroSense Conference*, Vol. 2755, 1996, pp. 165–173.

[16]Crassidis, J. L., Cheng, Y., Fosbury, A. M., and Nebelecky, C., "Decentralized Attitude Estimation Using a Quaternion Covariance Intersection Approach," *Journal of the Astronautical Sciences*, Vol. Paper AAS 08-284, 2008.

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

[18]Vanderplaats, G. N., *Numerical Optimization Techniques for Engineering Design*, Colorado Springs, CO: VR&D, 3rd ed., 2001.

[19]Lefferts, E. J., Markley, F. L., and Shuster, M. D., "Kalman Filtering for Spacecraft Attitude Estimation," *Journal of Guidance, Control, and Dynamics*, Vol. 5, No. 5, Sept.-Oct. 1982, pp. 417–429.

[20]Chapra, S. C., *Applied Numerical Methods with MATLAB for Engineers and Scientists*, McGraw-Hill, New York, 1st ed., 2005.

[21]Beyer, W. H., editor, *Standard Mathematical Tables and Formulae*, CRC Press, Boca Raton, FL, 29^{th} ed., 1991.

[22]Markley, F. L., Cheng, Y., Crassidis, J. L., and Oshman, Y., "Averaging Quaternions," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 4, July-Aug. 2007, pp. 1193–1196.

[23]Zhang, Z. and Huang, Y., "A Projection Method for Least Squares Problems with A Quadratic Equality Constraint," *SIAM Journal of Matrix Analysis and Applications*, Vol. 25, No. 1, 2003, pp. 188–212.

[24]Levin, Y. and Ben-Israel, A., "Directional Halley and Quasi-Halley Methods in n Variables." *Inherently parallel algorithms in feasibility and optimization and their applications*, Studies in Computational Mathematics 8, 2000.

[25]Cuyt, A. A. M. and Rall, L. B., "Computational Implementation of the Multivariate Halley Method for Solving Nonlinear Systems of Equations," *ACM Transactions on Mathematical Software*, Vol. 11, No. 1, March 1985, pp. 20–36.

[26]Crassidis, J. L. and Junkins, J. L., *Optimal Estimation of Dynamic Systems*, Boca Raton, Fl: Chapman & Hall/CRC, 2004.

[27]Markley, F. L., "Multiplicative vs. Additive Filtering for Spacecraft Attitude Estimation," *Proceding of the* 6^{th} *Conference on Dynamics and Control of Systems and Structures in Space*, 2004, pp. 467–474.

[28]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. pp. 282–284.

[29]Cheng, Y., Crassidis, J. L., and Markley, F. L., "Attitude Estimation for Large Field-of-View Sensors," *The Journal of the Astronautical Sciences*, Vol. Paper AAS 05-462, 2005.

[30]Shuster, M. D. and Oh, S. D., "Three-Axis Attitude Determination from Vector Observations," *Journal of Guidance and Control*, Vol. 4, No. 1, Jan.-Feb. 1981, pp. 70–77.

[31]van Loan, C. F., "Computing Integrals Involving Matrix Exponentials," *IEEE Transactions on Automatic Control*, Vol. 23, No. 3, June 1978, pp. 395–404.

[32]Grant, M. and Boyd, S., "CVX: MATLAB Software for Disciplined Convex Programming," (web page and software). http://stanford.edu/ boyd/cvx.

[33]Banas, W. D., *Micro-Arcsecond Line-Of-Sight Filtered Performance for Spacecraft Formation Flying*, Master's thesis, The State University of New York at Buffalo, May 2008.