MAV Attitude Determination by Vector Matching

DEMOZ GEBRE-EGZIABHER, Member, IEEE
University of Minnesota–Twin Cities Campus

GABRIEL H. ELKAIM, Member, IEEE
University of California, Santa Cruz

An attitude determination algorithm suitable for micro aerial vehicle (MAV) applications is developed. The algorithm uses Earth's magnetic and gravity field vectors as observations. The magnetic field vector measurements are obtained from a magnetometer triad. The gravity field vector is measured by fusing information from an accelerometer triad with GPS/WAAS (wide area augmentation system) velocity measurements. Two linearization and estimator designs for implementing the algorithm are discussed. Simulation and experimental flight test results validating the algorithm are presented.

Manuscript received July 11, 2006; revised December 18, 2006; released for publication August 18, 2007.

IEEE Log No. T-AES/44/3/929450.

Refereeing of this contribution was handled by A. Helfrul.

Authors' addresses: D. Gebre-Egziabher, 110 Union St. SE, 107 Ackerman Hall, University of Minnesota, Twin Cities Campus, Minneapolis, MN 55455, E-mail: (gebre@aem.umn.edu); G. H. Elkaim, Computer Engineering Dept., University of California at Santa Cruz, 1156 High St., SOE3, Santa Cruz, CA 95064.

INTRODUCTION

Attitude is the term used to describe a rigid body's orientation in three-dimensional space. In a more general sense, it is the description of the relative orientation of two coordinate frames. In vehicle guidance, navigation, and control (GNC) applications near Earth's surface (e.g., applications involving airplanes, marine vessels, etc.) the two coordinate frames of interest are sometimes referred to as the body and navigation reference frames. The body frame is rigidly attached to and moves with the vehicle. The navigation frame is normally a locally level (or tangent) coordinate frame. That is, it has an origin attached to Earth's surface and located directly below the vehicle's current position. Its x-y-z axes are lined-up with North, East, and Down (along the local vertical) directions, respectively. Attitude determination systems are used to measure or estimate the relative orientation of these two frames. The information generated by attitude determination systems is indispensable in many GNC applications. A few examples of applications requiring attitude information include pilot-in-the-loop control of manned aircraft, accurate payload pointing on remote sensing platforms, and autonomous navigation and guidance of uninhabited aerial, ground, and marine vehicles.

The recent interest in high performance, micro aerial vehicles (MAVs) has necessitated the design of compact, accurate, and inexpensive attitude determination systems. MAVs are designed to be disposable and are very small in size and weight (largest dimensions no greater than 15 cm) [1, 2]. At these scales, the avionics and sensor payloads can represent a significant fraction of the overall vehicle dimension and weight. To address this need, many inexpensive rate-gyro based attitude determination systems have been developed [3–5]. However, inexpensive and miniature solid-state rate gyros ($<$ \$1000 per axis) tend to be low-performance sensors which have outputs subject to wideband noise and rate instabilities on the order of 10 to 100°/hr [6, 7]. In order to determine attitude, the rate gyro outputs must be integrated to give attitude and this leads to unbounded attitude errors. Thus, successful implementation of an attitude determination system that relies solely on rate gyros requires the use of sensors with exceptionally accurate and stable outputs. These types of gyros would have output errors less than 0.1°/hr and tend to be 1) prohibitively expensive, 2) have high power consumption, and/or 3) be physically too large for many miniature vehicle applications.

An alternative to relying on accurate and expensive rate gyros is to devise a system which fuses miniature, low-performance (and low-cost and low-power) gyros with a gyro-free aiding system using a complementary

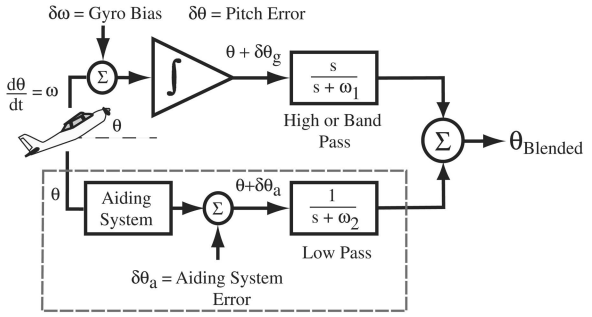


Fig. 1. Complementary filter for blending information from aiding system (like the one discussed in this paper) with rate gyros. Dashed box encloses system generating drift-free attitude but perhaps noisy attitude solution.

filter architecture as shown in Fig. 1. The gyro-free aiding system provides either one of two types of information. It can provide 1) a noisy but unbiased direct attitude measurement periodically (i.e., a low bandwidth attitude update), or 2) it can provide indirect measurements from which attitude can be extracted using an observer. The aiding system's measurements are used to arrest error growth due the integration of the gyro biases and also to estimate these gyro biases in real time. A few examples of aiding system which provide direct measurements include multi-antenna GPS attitude determination systems [9], accelerometer- or inclinometer-based leveling systems [7] and the novel pseudoattitude system described in [10]. An example of aiding systems providing indirect attitude measurements is a system, shown in Fig. 2, which uses an extended Kalman filter (EKF) to extract attitude information from GPS position estimates.

All of the above mentioned aiding systems have drawbacks which make them unsuitable for MAV applications. For example, multi-antenna GPS attitude determination systems require separation distances between antennas which are too large (> 16 cm) for use on MAVs [11]. Accelerometer- or inclinometer-based systems provide useful attitude information only when the vehicle is in unaccelerated flight [7]. EKF-based systems which extract attitude information indirectly from GPS position estimates suffer from conditional observability problems, and they cannot guarantee bounded attitude errors unless the vehicle follows a prescribed acceleration history. In certain scenarios the required acceleration history may be beyond the maneuvering capabilities of small MAVs.

The purpose of this paper is to present an algorithm and system design for a gyro-free attitude determination system suitable for MAV applications. It is an extension of ideas first presented in [8] and addresses all the shortcomings of the above mentioned aiding systems. The hardware and system architecture addresses the size, weight and power constraints unique to MAVs while the attitude determination algorithm and its implementation address issues encountered when using low-cost (or performance) inertial sensors. This system derives attitude without integrating sensor outputs and, as such, provides a solution which has bounded errors. In applications where the required bandwidth is small or output latencies are not problematic (e.g., postmission data analysis), the algorithm can be used as a stand-alone attitude solution. In applications where high bandwidths are required and latencies are not acceptable, it can be fused with rate gyros and serve as an aiding system which continuously calibrates rate gyro biases. The system developed uses magnetometers, accelerometers, and GPS (velocity measurements only) as the primary sensors. The attitude algorithm is mechanized in terms of quaternions, and the attitude quaternion is determined using a minimum variance estimator by processing of two noncollinear vectors measurements. The two vectors measured are Earth's magnetic field and gravitational acceleration vectors.

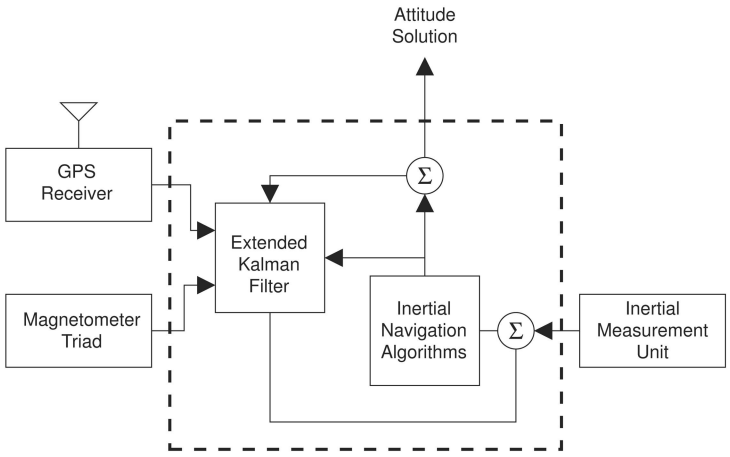


Fig. 2. An EKF architecture for blending INS with GPS information. GPS position solution provides indirect attitude aiding information.

BACKGROUND

Determining attitude from two or more vector measurements is a problem that has been a subject of interest for some time [12] and, thus, prior work in this area is extensive. Over the years, many algorithms to solve this problem have been developed and a representative (but not exhaustive) list of such algorithms are described in [9], [13]–[24]. In general, these algorithms can be classified into three groups. The first group consists of deterministic algorithms such as the ones described in [14]. The second group consists of algorithms which take a classical least squares approach to the problem as articulated by Wahba in [12]. That is, given a set of vectors $\vec{u}_i^n$ for $i = 1,\ldots,N$ known in the n coordinate frame and measurements of these vectors denoted $\vec{u}_i^b$ for $i = 1,\ldots,N$ in the b coordinate frame, find the n to b transformation matrix $\overset{n\to b}{C}(\mathbf{q})$ (or the attitude quaternion $\mathbf{q}$) which minimizes the following cost function J:

$$J = \frac{1}{2}\sum_{i=1}^{N}\left(\vec{u}_i^b - \overset{n\to b}{C}(\mathbf{q})\vec{u}_i^n\right)^T\left(\vec{u}_i^b - \overset{n\to b}{C}(\mathbf{q})\vec{u}_i^n\right) \quad (1)$$

subject to the constraint:

$$\|\mathbf{q}\| = 1. \quad (2)$$

A batch solution to this constrained least squares problem was given in [17] while a recursive one was presented in [18]. The third group of algorithms takes a filtering approach to the problem. In this approach, the attitude determination problem is cast in the form of an observer or filter. Specific examples of this approach are given in [19] (Euler angle filter), [21] (direction cosine matrix filter), [22] (Rodrigues parameter filter), and [20], [23] (quaternion filters).

While the classical approaches to the problem provide an exact solution, they cannot easily accommodate a dynamic model or sensor errors into their formulation. On the other hand, while the filtering approach allows accommodation of dynamic models and sensor errors, the solutions they provide are not necessarily optimal because they are derived from a linearization of the nonlinear attitude equations. An algorithm which combines both approaches in a Kalman filtering framework was developed in [24].

The contributions of this paper are twofold. First, it adds another algorithm for attitude determination from two-vector observation to the literature. The algorithm developed in this paper belongs to the third group of two-vector attitude algorithms described above; it is an EKF solution to the nonlinear equation relating attitude parameters to vector measurements. The algorithm is fundamentally similar to the one developed in [20] but instead of an additive quaternion update it uses a multiplicative one along the lines of [16], [25], and [26]. More importantly, however, we show that the choice of the coordinate frame in which to linearize the attitude equations has a significant bearing on implementation of the attitude filter in MAV applications. For example, it is shown that one particular linearization scheme results in a time-invariant EKF measurement equation which can be directly integrated into the psi angle INS/GPS filter mechanization architecture [27]. This means that this attitude determination system and algorithm can be easily incorporated into an INS/GPS fusion filter like the one depicted in Fig. 2. This helps to address the issues of condition observability which plagues these systems especially in MAV application where low-cost/performance inertial sensors are used.

The second contribution of this paper is that it demonstrates the design and operation of a vector matching attitude determination system suitable for use in small MAVs. The two vectors used are measured Earth's magnetic field vector $\vec{h}$, and a synthetic Earth's gravitational field vector $\vec{g}$. The synthetic vector is constructed by subtracting specific force measurements provided by a triad accelerometer from a GPS-derived vehicle acceleration vector. The acceleration measurement from GPS is derived by numerically differentiating GPS velocity estimate from a wide area augmentation system (WAAS) corrected signal. We show that the acceleration signal derived by differentiating the WAAS-corrected GPS velocity is normally a surprisingly clean and usable measurement.

The remainder of the paper is organized as follows: First, the general attitude determination problem is posed and the measurement equation relating vector observations to attitude errors is derived. Next, we present the design of estimators to solve the attitude equations. Following this, the stability, convergence, and estimation error characteristics of the attitude algorithm are verified in simulation for both the static and dynamic cases. Extended Kalman filtering and implementation issues of the algorithm are discussed next. The hardware description of the system follows, then a presentation of experimental results and validation of the attitude determination algorithm and systems using postprocessed flight test data. Finally, conclusions and a summary close the paper.

MEASUREMENT EQUATION DERIVATION

Attitude determination from two vector measurements requires knowledge of the components of two noncollinear, non-zero vectors in two separate coordinate frames. Let us denote these two vectors

as $\vec{u}$ and $\vec{v}$. The superscript "b" and "n" are used to denote body frame or navigation frame resolution of these vectors, respectively. For example, $\vec{u}^b$ represents the vector $\vec{u}$ with its components expressed (or coordinatized) in the body frame. First, the relations involving the vector $\vec{u}$ are derived. The relations involving $\vec{v}$ are a repeat of those derived for $\vec{u}$.

The transformation which maps the vector $\vec{u}$ expressed in the body frame to its resolution in the navigation frame is

$$\vec{u}^b = \overset{n\to b}{C}(\mathbf{q})\vec{u}^n. \tag{3}$$

The navigation-to-body frame transformation matrix $\overset{n\to b}{C}(\mathbf{q})$ is a function of the attitude quaternion $\mathbf{q}$ and can be expressed as

$$\overset{n\to b}{C}(\mathbf{q}) = \begin{bmatrix} 1-2({q_2}^2+{q_3}^2) & 2(q_1q_2+q_3q_0) & 2(q_1q_3-q_2q_0) \\ 2(q_1q_2-q_3q_0) & 1-2({q_1}^2+{q_3}^2) & 2(q_2q_3+q_0q_1) \\ 2(q_1q_3+q_2q_0) & 2(q_2q_3-q_1q_0) & 1-2({q_1}^2+{q_2}^2) \end{bmatrix}. \tag{4}$$

In this paper we adopt the following notation convention for the attitude quaternion [28]:

$$\mathbf{q} = \begin{bmatrix} q_0 \\ \vec{q} \end{bmatrix}. \tag{5}$$

It is composed of a scalar component q_0 defined as

$$q_0 = \cos\left(\frac{\Phi}{2}\right) \tag{6}$$

and a vector component given by

$$\vec{q} = \sin\left(\frac{\Phi}{2}\right)\hat{e} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix}. \tag{7}$$

The angle Φ is the rotation angle from Euler's or Chales' theorem [29] and $\hat{e}$ is the eigenvector of $\overset{n\to b}{C}$ corresponding to the eigenvalue of unity. Alternately, (3) can be written in terms of quaternions as follows:

$$\mathbf{u}^b = \mathbf{q}^* \otimes \mathbf{u}^n \otimes \mathbf{q} \tag{8}$$

where $\otimes$ represents quaternion multiplication and $\mathbf{q}^*$ is the complementary rotation of the quaternion $\mathbf{q}$ and is defined as

$$\mathbf{q}^* = [q_0 \quad -\vec{q}]^T. \tag{9}$$

Quaternion multiplication is defined as follows:

$$\mathbf{r} \otimes \mathbf{s} = \begin{bmatrix} s_0r_0 - \vec{r}^T\vec{s} \\ \vec{r}\times\vec{s} + r_3\vec{s} + s_3\vec{r} \end{bmatrix}. \tag{10}$$

The quaternion equivalent of $\vec{u}$ is denoted $\mathbf{u}$ and is equal to $[0 \;\; \vec{u}^T]^T$.

The attitude determination algorithm developed in this paper is fundamentally a linearization of (3) or its quaternion equivalent given by (8). Given an initial guess of the attitude along with body and navigation frame measurements of two vectors the algorithm computes the difference between the true attitude and this initial guess. The difference is used to correct the initial guess, and the process is repeated until convergence is achieved. Even though the final attitude determination algorithm will be in terms of quaternions, we derive the equations starting from (3), because this approach provides more insight and allows the use of matrix algebra in lieu of quaternion math.

The difference between the assumed attitude (or initial guess of attitude) and the true attitude is captured in a transformation matrix error $(\overset{n\to b}{\delta C})$, or equivalently the quaternion error $\mathbf{q}_e$. Both the matrix and quaternion errors represent the error between the coordinate frame chosen for linearization and the actual or true orientation of that frame. In this regard, there are two choices for the coordinate frames about which to linearize: the body frame or the navigation frame. The equations for the two linearization schemes are derived separately.

Linearization in the Navigation Frame

Let us define $\mathbf{q}_e$ to be the small rotation error between the estimated attitude $\hat{\mathbf{q}}$ and the true attitude $\mathbf{q}$. Throughout this paper we use carets above variables to denote that they are estimated quantities. The error quaternion is small but non-zero. It is non-zero because errors in the various sensors result in attitude errors. The relationship is expressed in terms of quaternion multiplication as follows:

$$\mathbf{q} = \hat{\mathbf{q}} \otimes \mathbf{q}_e. \tag{11}$$

That is, the estimated attitude is rotated a small amount further in order to arrive at the true attitude quaternion (the definition of quaternion multiplication). Since the error quaternion $\mathbf{q}_e$ is assumed to represent a small rotation, it can be approximated as [29]

$$\mathbf{q}_e = \begin{bmatrix} 1 \\ \vec{q}_e \end{bmatrix}. \tag{12}$$

An alternative way to view the relation between the estimated and actual attitude is using the transformation (or direction cosine) matrix. Since the error quaternion $\mathbf{q}_e$ is nothing more than a perturbation to the direction cosine matrix $\overset{n\to b}{C}(\mathbf{q})$, we can write

$$\overset{n\to b}{\delta C}(\mathbf{q}) \triangleq \overset{n\to b}{C}(\mathbf{q}_e). \tag{13}$$

Noting that the vector components of $\mathbf{q}_e$ are small, the perturbation to the transformation matrix in (4) can be rewritten as

$$\overset{n\to b}{C}(\mathbf{q}_e) = \begin{bmatrix} 1 & 2q_{e3} & -2q_{e2} \\ -2q_{e3} & 1 & 2q_{e1} \\ 2q_{e2} & -2q_{e1} & 1 \end{bmatrix} \tag{14}$$

where

$$\vec{\mathbf{q}}_e = [q_{e1} \;\; q_{e2} \;\; q_{e3}]^T. \tag{15}$$

Let the n' coordinate frame be the computed (and, thus, erroneous) navigation frame while n represents the actual true navigation frame. The superscript b represents the body frame. Using these definitions we write the n to b transformation matrix as

$$\overset{n\to b}{C} = \overset{n'\to b}{C}\overset{n\to n'}{C} \tag{16}$$

$$= \overset{n'\to b}{C}\,\delta C \tag{17}$$

$$= \overset{n'\to b}{C}\overset{n\to b}{C}(\mathbf{q}_e) \tag{18}$$

where

$$\overset{n\to b}{\delta C} = \overset{n'\to n}{C} = \overset{n\to b}{C}(\mathbf{q}_e). \tag{19}$$

This implies that these matrices can be written in terms of small rotations about the navigation frame's coordinate axes. These rotation errors are denoted as ϵ_N (rotation about the north axis), ϵ_E (rotation about the east axis) and ϵ_D (rotation about the down/vertical axis). In terms of these angles the attitude error matrix can be written as

$$\overset{n\to b}{C}(\mathbf{q}_e) = \begin{bmatrix} 1 & \epsilon_D & -\epsilon_E \\ -\epsilon_D & 1 & \epsilon_N \\ \epsilon_E & -\epsilon_N & 1 \end{bmatrix}. \tag{20}$$

These attitude errors are similar to the so-called psi-angle error models used in error analysis of inertial navigation systems [27] and they are also used as accuracy metrics later in the paper. Since $\overset{n\to b}{C}(\mathbf{q}_e)$ represents a small rotation between the n' and n frames, using (14) it can be written as

$$\overset{n\to b}{C}(\mathbf{q}_e) = I_{3\times 3} - 2[\vec{q}_e\times] \tag{21}$$

where $I_{3\times 3}$ is the three-by-three identity matrix and $[\vec{q}_e\times]$ is a skew-symmetric matrix composed of the entries of $\vec{q}_e$. That is,

$$[\vec{q}_e\times] = \begin{bmatrix} 0 & q_{e3} & -q_{e2} \\ -q_{e3} & 0 & q_{e1} \\ q_{e2} & -q_{e1} & 0 \end{bmatrix}. \tag{22}$$

Substituting this back into (3) leads to

$$\vec{u}^b = \overset{n\to b}{C}\,\vec{u}^n \tag{23}$$

$$= \overset{n'\to b}{C}\overset{n\to n'}{C}\,\vec{u}^n \tag{24}$$

$$= \overset{n'\to b}{C}\,[I_{3\times 3} - 2[\vec{q}_e\times]]\vec{u}^n. \tag{25}$$

Recalling that $\hat{}$ is used to denote estimated quantities and that $\overset{n'\to b}{C} = \overset{n\to b}{\hat{C}}$, the previous equation can be premultiplied by $\overset{b\to n'}{C}$ to yield

$$\overset{b\to n'}{C}\,\vec{u}^b = [I_{3\times 3} - 2[\vec{q}_e\times]]\vec{u}^n \tag{26}$$

$$\hat{\vec{u}}^n = \vec{u}^n - 2[\vec{u}^n\times]\vec{q}_e \tag{27}$$

using the well-known identity that $\vec{a}\times\vec{b} = -\vec{b}\times\vec{a}$. Let us define $\delta\vec{u}^n = -\vec{u}^n\hat{\vec{u}}^n$. Using this definition leads to the following:

$$\delta\vec{u}^n = -2[\vec{u}^n\times]\vec{q}_e. \tag{28}$$

If the above derivation is repeated for the second vector, $\vec{v}$, and the equations for both $\vec{u}$ and $\vec{v}$ are put together, the following linear measurement equation is obtained

$$\begin{bmatrix} \delta\vec{u}^n \\ \delta\vec{v}^n \end{bmatrix} = \begin{bmatrix} -2[\vec{u}^n\times] \\ -2[\vec{v}^n\times] \end{bmatrix}\vec{q}_e. \tag{29}$$

This equation can be used in an estimator or observer to determine attitude. The design of two such estimators is discussed later in the paper after we present an alternate body frame linearization of the attitude equations.

Linearization in the Body Frame

The procedure for linearization of the attitude equations in the body frame is similar to that for the navigation frame, above. The major difference is that the derivation starts by expanding the following

$$\overset{b\to n}{C} = \overset{b'\to n}{C}\overset{b\to b'}{C} \tag{30}$$

$$= \overset{b\to n}{\hat{C}}\overset{b\to n}{C}(\mathbf{q}_e). \tag{31}$$

Note that now the b' coordinate frame is the computed body frame while b is the true body frame. Similar to what was done earlier, note the following equivalence of notations:

$$\overset{b\to n}{\delta C} = \overset{b\to b'}{C} = \overset{b\to n}{C}(\mathbf{q}_e). \tag{32}$$

These matrices represent the error in the body to navigation frame transformation matrix and now $\overset{b\to n}{C}(\mathbf{q}_e)$ represents a small rotation between the b' and

b frames. It can be written as

$$\overset{b\to n}{C}(\mathbf{q}_e) = I_{3\times 3} - 2[\vec{q}_e\times]. \tag{33}$$

Even though the above equation for $\overset{b\to n}{C}(\mathbf{q}_e)$ looks identical to the expression for $\overset{n\to b}{C}(\mathbf{q}_e)$ given by (21), there is a subtle, albeit significant difference: in (21) and (33), the quaternion $\mathbf{q}$ (and its error $\mathbf{q}_e$) are not the same. The quaternions in (33) are the complements of the quaternions in (21). Equation (33) can be used to write the b to n transformation as

$$\overset{b\to n}{C} = \overset{\hat{b}\to n}{C}[I_{3\times 3} - 2[\vec{q}_e\times]]. \tag{34}$$

This relationship is used to write the transformation of $\vec{u}^b$ to $\vec{u}^n$ in the following manner:

$$\vec{u}^n = \overset{b\to n}{C}\vec{u}^b \tag{35}$$

$$= \overset{\hat{b}\to n}{C}[I_{3\times 3} - 2[\vec{q}_e\times]]\vec{u}^b. \tag{36}$$

Premultiplying both sides by $\overset{\hat{n}\to b}{C}$ and rearranging leads to

$$\delta\vec{u}^b = -\hat{\vec{u}}^b\,\vec{u}^b \tag{37}$$

$$= -2[\hat{\vec{u}}^b\times]\vec{q}_e. \tag{38}$$

Once again, repeating the above derivation for the second vector, $\vec{v}$, and assembling the equations for both $\vec{u}$ and $\vec{v}$ yields the following measurement equation:

$$\begin{bmatrix}\delta\vec{u}^b\\ \delta\vec{v}^b\end{bmatrix} = \begin{bmatrix}-2[\vec{u}^b\times]\\ -2[\vec{v}^b\times]\end{bmatrix}\vec{q}_e. \tag{39}$$

ATTITUDE ESTIMATOR DESIGN

Either (29) or (39) can be used in an estimator to solve the attitude determination problem. To do this we note that both (29) and (39) can be written as

$$\vec{z} = H\vec{q}_e \tag{40}$$

where $\vec{z}$ is the observation vector and H is the observation (or measurement) matrix. Assuming linearization in the navigation frame ((29)), then the observation vector $\vec{z}$ becomes

$$\vec{z} = \begin{bmatrix}\delta\vec{u}^n\\ \delta\vec{v}^n\end{bmatrix} \tag{41}$$

and the observation matrix H becomes

$$H = \begin{bmatrix}-2[\vec{u}^n]^\times\\ -2[\vec{v}^n]^\times\end{bmatrix}. \tag{42}$$

We present two minimum variance solutions to the attitude determination problem. The first is an iterated least squares solution and the second is an EKF implementation. The two vectors ($\vec{u}$ and $\vec{v}$), the way they are measured, and the error models for the measurement of these vectors is the same for both estimators. Thus, before discussing the details of the two estimators, we present details of the vectors used in the physical system, how they are measured, and error models for the vector measurements.

Vector Measurement Models

The physical implementation of this MAV attitude determination system uses Earth's magnetic field and gravitational acceleration vectors as the two vector observations. The vector $\vec{u}^n$ in the algorithm above is defined to be equal to Earth's magnetic field vector in the navigation frame $\vec{h}^n$. This vector is a function of geographic location and can be computed easily using the International Geomagnetic Reference Field (IGRF) model [30]. Similarly, $\vec{v}^n$ is defined to be equal to Earth's gravitational acceleration vector in the navigation frame $\vec{g}^n$. It is also a function of geographic location and can be computed using a model such as the World Geodetic System 1984 (WGS-84) gravity model [31]. In the implementation here, the values of $\vec{h}^n$ and $\vec{g}^n$ computed using [30] and [31] are assumed to be exact.

The vector $\vec{u}^b$ is defined to be the body frame measurement of Earth's magnetic field vector and is measured by a triad of magnetometers. This measurement is modeled as

$$\hat{\vec{h}}^b(t_k) = \vec{h}^b(t_k) + \vec{n}_h(t_k). \tag{43}$$

The caret above the vector on the left hand side of the equation indicates a measured quantity. The measured field value is equal to the true value (i.e., measurement that would be made by an error-free magnetometer triad) corrupted by an additive error. We assume that the magnetometers have been calibrated so that correlated sensor errors such as null-shift or Markov biases have been removed. Thus, it is reasonable to assume that the remaining sensor error term n_h can be modeled as a white noise sequence with covariance R_h given as

$$R_h = \mathcal{E}\{\vec{n}_h(t_j)\vec{n}_h^T(t_k)\} = \begin{cases}\sigma_h^2 I_{3\times 3} & j = k\\ 0_{3\times 3} & j \neq k\end{cases} \tag{44}$$

where σ_h^2 is the noise power.

The vector $\vec{v}^b$ is defined to be the body frame measurement of Earth's gravitational acceleration. In a nonaccelerating vehicle, a triad of accelerometers would provide a direct measurement of this vector. However, accelerometers are fundamentally sensors for measuring specific force and not accelerations. The output of an accelerometer $\vec{f}$ is related to the gravitational acceleration vector $\vec{g}$ and the vehicle's

acceleration vector $\vec{a}$ by the following relation:

$$\vec{f}^b = \vec{a}^b - \vec{g}^b \tag{45}$$

where the superscript b indicates body frame expression of the vectors. Thus,

$$\vec{g}^b = \vec{a}^b - \vec{f}^b \equiv \vec{v}^b. \tag{46}$$

When the vehicle is not accelerating ($\vec{a}^b = 0$), the accelerometers provide a direct measure of $-\vec{g}^b$. However, when the vehicle is accelerating, a correction equal to $\vec{a}^b$ must be made to the accelerometer readings.

The output error model for the acceleormeters has the same form as the output error model for the magnetometers:

$$\hat{\vec{f}}^b(t_k) = \vec{f}^b(t_k) + \vec{n}_f(t_k). \tag{47}$$

The sampling noise term $\vec{n}_f$ is also assumed to be a white noise sequence with covariance R_f given by

$$R_f = \mathcal{E}\{\vec{n}_f(t_j)\vec{n}_f^T(t_k)\} = \begin{cases} \sigma_f^2 I_{3\times 3} & j = k \\ 0 & j \neq k \end{cases} \tag{48}$$

where σ_f^2 is the noise power. The error model for $\vec{a}$ and the magnitude of its covariance depends on how $\vec{a}$ is computed. In the implementation discussed here, $\vec{a}^b$ is determined by using the measured vehicle velocity $\dot{\hat{x}}$ in the following difference equation:

$$\vec{a}^b(t_k) = \frac{1}{\Delta t}(\dot{\vec{x}}^b(t_k) - \dot{\vec{x}}^b(t_{k-1})) \tag{49}$$

where $\Delta t = t_k - t_{k-1}$. The measurements of $\dot{\vec{x}}$ comes from GPS. GPS provides a measurement of the vehicle's velocity in the navigation frame and not the body frame. That is, it provides $\dot{\vec{x}}^n$ and not $\dot{\vec{x}}^b$. This is not a problem, however, because linearization in the navigation frame only requires that we use $\hat{\vec{v}}^b$ to compute $\hat{\vec{v}}^n$. Thus,

$$\hat{\vec{v}}^n \equiv \hat{\vec{g}}^n = (\hat{C}^{n\to b})^T \hat{\vec{g}}^b \tag{50}$$

$$= \hat{\vec{a}}^n - (\hat{C}^{n\to b})^T \vec{f}^b. \tag{51}$$

In the equation above

$$\hat{\vec{a}}^n(t_k) = \frac{1}{\Delta t}(\dot{\hat{\vec{x}}}^n(t_k) - \dot{\hat{\vec{x}}}^n(t_{k-1})). \tag{52}$$

We model the velocity measurements as

$$\dot{\hat{\vec{x}}} = \dot{\vec{x}}^n_{\text{GPS}} = \dot{\vec{x}}^n + \vec{n}_{\dot{x}}. \tag{53}$$

Since, in this work, we are using differentially corrected GPS (corrections provided by the WAAS) it is reasonable to assume that most of the correlated velocity errors have been removed and what remains is essentially wideband noise. Therefore, the velocity error $\vec{n}_{\dot{x}}$ is modeled as an uncorrelated white noise sequence where the covariance of $\vec{n}_{\dot{x}}$ (denoted $R_{\dot{x}}$) is given by

$$R_{\dot{x}} = \mathcal{E}\{\vec{n}_{\dot{x}}(t_j)\vec{n}_{\dot{x}}^T(t_k)\} = \begin{cases} \begin{bmatrix} \sigma_{\dot{x}_N}^2 & 0 & 0 \\ 0 & \sigma_{\dot{x}_E}^2 & 0 \\ 0 & 0 & \sigma_{\dot{x}_D}^2 \end{bmatrix} & j = k \\ 0_{3\times 3} & j \neq k \end{cases} \tag{54}$$

where the subscripts N, E, and D refer to the noise power in the North, East, and Down directions, respectively. Thus, R_a which is the covariance of $\hat{\vec{a}}^n$ is given by

$$R_a = \mathcal{E}\{\vec{n}_{\dot{x}}(t_j)\vec{n}_{\dot{x}}^T(t_k)\} = \begin{cases} \frac{2}{\Delta t^2} R_{\dot{x}} I_{3\times 3} & j = k \\ 0_{3\times 3} & j \neq k, \end{cases} \tag{55}$$

The covariance matrices R_h, R_f, and R_a are used in computing error bounds on the attitude estimate later in the paper.

Iterated Least Squares Estimator

The iterated least squares solution does not leverage a priori information about the vehicle's attitude. Instead, at each time step it performs a global search for the attitude quaternion. The iterated least squares algorithm is used in the simulation studies evaluating the convergence characteristics of the attitude algorithms developed in this paper. Given the above error models and assuming linearization in the navigation frame ((29)), the iterated least squares solution is implemented as follows.

1) Initialize the attitude quaternion as $\hat{\mathbf{q}} = [1\ 0\ 0\ 0]^T$, and the attitude error quaternion as $\hat{\mathbf{q}}_e = [1\ 0\ 0\ 0]^T$.

2) Use $\hat{\mathbf{q}}$ to map the body measurement of $\vec{u}$ and $\vec{v}$ to the navigation frame. That is, compute $\hat{\mathbf{u}}^n = \hat{\mathbf{q}} \otimes \hat{\mathbf{u}}^b \otimes \hat{\mathbf{q}}^*$ and likewise for $\hat{\mathbf{v}}^n = \hat{\mathbf{q}} \otimes \hat{\mathbf{v}}^b \otimes \hat{\mathbf{q}}^*$.

3) Formulate the errors $\delta\hat{\vec{u}}^n = \vec{u}^n - \hat{\vec{u}}^n$ and $\delta\hat{\vec{v}}^n = \vec{v}^n - \hat{\vec{v}}^n$.

4) Form H ((29)) and take its pseudoinverse: $[H^T\ H]^{-1}H^T = H^\dagger$.

5) Estimate the quaternion error,

$$\hat{\vec{q}}_e = H^\dagger \begin{bmatrix} \delta\hat{\vec{u}}^n \\ \delta\hat{\vec{v}}^n \end{bmatrix}.$$

6) Update the quaternion estimate as follows: $\hat{\mathbf{q}} = \hat{\mathbf{q}} \otimes \mathbf{q}_e$.

7) Normalize the updated quaternion estimate in the following way: $\hat{\mathbf{q}}^+ = \hat{\mathbf{q}}^+ / \|\hat{\mathbf{q}}^+\|$.

8) Return to step (2) and repeat until convergence is achieved.

Note that step 6 represents an incremental rotation towards the correct solution, as represented by quaternion multiplication. This is different from the algorithm in [20] where an incremental update to the quaternion error is added to the prior estimate.

Given the algorithm developed above, the following two questions are of interest: 1) How accurate is the attitude estimate and what factors influence the accuracy? 2) Does the algorithm converge, and what are its convergence characteristics? To qualitatively answer the first question, we develop a covariance expression for the attitude estimate and simulation studies are used to quantitatively answer both accuracy and convergence issues.

The quality of the attitude estimate generated by this algorithm is captured in the state covariance matrix P defined as

$$P = \mathcal{E}\{\vec{q}_e \vec{q}_e^T\} = \mathcal{E}\{H^\dagger \vec{z}\vec{z}^T H^{\dagger T}\}. \tag{56}$$

For linearization in the navigation frame $H^\dagger$ is independent of $\hat{\mathbf{q}}$ (or $\mathbf{q}_e$). Thus, (56) can be written as

$$P = H^\dagger \mathcal{E}\{\vec{z}\vec{z}^T\} H^{\dagger T} = H^\dagger R H^{\dagger T} \tag{57}$$

where R_z is the covariance of the navigation frame measurement of $\vec{u}$ and $\vec{v}$. The magnitude of its entries depends on the quality of the magnetometers, accelerometers and GPS velocity measurements used to construct the two vectors $\hat{\vec{g}}^n$ and $\hat{\vec{h}}^n$. It also depends on the attitude of the body frame relative to the navigation frame. From (44), (48), (51), and (55) it can be shown that R_z is given by

$$R_z = \begin{bmatrix} (\hat{C}^{n\to b}) R_n (\hat{C}^{n\to b})^T & 0_{3\times 3} \\ 0_{3\times 3} & R_a + (\hat{C}^{n\to b}) R_f (\hat{C}^{n\to b})^T \end{bmatrix}. \tag{58}$$

The square root of the diagonal entries of P are the standard deviations of the quaternion estimates. It is more intuitive to describe attitude errors in terms of the small rotation angles about the navigation frame axes, rather than quaternions. From (20), the variances of these rotation angles are approximated by

$$\sigma^2_{\epsilon_N} = 4P_{11} \tag{59}$$

$$\sigma^2_{\epsilon_E} = 4P_{22} \tag{60}$$

$$\sigma^2_{\epsilon_D} = 4P_{33}. \tag{61}$$

Note that this interpretation of attitude errors is only valid for linearization in the navigation frame. If the linearization is carried out in the body frame, the physical meaning or interpretation of the entries in the covariance matrix (P_{ii}) will be different.

Extended Kalman Filter Solution

The EKF approach is more efficient than the iterated least squares solution because it leverages prior attitude information. Unlike the iterated least squares soltuion which performs a global search at each time step, the EKF just refines the attitude solution from the previous time step using the most current vector observations. As such, it is better suited for real-time applications.

In order to implement the EKF, equations accounting for the dynamics must be included in the formulation. If angular rate information is available (from rate gyros), a dynamic model for $\vec{q}_e$ based on the kinematics of the attitude problem can be included [7]. The dynamic model for $\hat{\vec{q}}_e$ can be written in the standard state space form for dynamics systems as follows:

$$\dot{\hat{\vec{q}}}_e = F\hat{\vec{q}}_e + G\vec{w}. \tag{62}$$

The matrix F is the system dynamic matrix. The vector $\vec{w}$ is the process noise vector which models uncertainties in the dynamic model for $\hat{\vec{q}}_e$ and the rate gyros used. It is a white noise sequence with covariance R_w. The matrix G is the process noise mapping matrix. The structures of F and G depend on the frame used for the linearization (navigation or body) and the types of rate gyros used. Given the dynamic model in (62) and assuming navigation frame linearization of the measurement equation, the EKF attitude estimator is implemented as follows.

1) At time step t_{k-1} sample gyros to propagate $\hat{\mathbf{q}}$.

2) Propagate the state covariance forward in time using the following:

$$P_k^- = \Phi P_{k-1}^+ \Phi^T + C_d. \tag{63}$$

P_{k-1}^- is the a priori state error covariance, Φ is the discrete equivalent of F, and C_d is the discrete equivalent GR_wG^T.

3) Use the vector observation of Earth's magnetic and gravity fields to construct $\vec{z}$, H, and R_z.

4) Compute the Kalman gain matrix $L = P_k^- H^T (HP_k^- H^T + R_z)^{-1}$.

5) Compute $\hat{\vec{q}}_e = L\vec{z}$.

6) Similar to the iterated least squares approach, update $\hat{\mathbf{q}}$ using $\hat{\mathbf{q}}_e$ and normalize.

7) Update the state error covariance by $P_k^+ = (I_{3\times 3} - LH)P_k^-$.

8) Return to step 1 and repeat.

The dynamic model given in (62) can be augmented to include sensor errors as well. In this instance, the elements of the process noise mapping matrix G may include parameters for shaping filters which map the white noise sequence $\vec{w}$ into correlated sensor errors. More details on this and, in general, on how to incorporate rate gyros into an attitude estimator are given in [7] and [25]. We, therefore,

forgo further discussion of these details and instead focus on a gyro-free implementation of the EKF estimator.

The gyro-free form of the EKF estimator can be implemented if the frequency content of the vehicle's attitude dynamics is smaller than the frequency at which the vector observation are being made. In this instance, the time evolution of the quaternion error $\vec{q}_e$ between vector observations can be modeled as a time-correlated random process. A mathematically tractable random process that is suited for this case would be that of an exponentially correlated (or first order Gauss-Markov) process. Noting that using such a random process is nothing more than a mathematical restatement of the fact that the vehicle's attitude does not change appreciably between vector observations, we can write the dynamics and process noise mapping matrices as

$$F = -\frac{1}{\tau} I_{3\times 3} \tag{64}$$

$$G = -\tau F \tag{65}$$

where τ is the correlation time for the quaternion error $\vec{q}_e$. The correlation time τ and the process noise covariance matrix R_w are parameters that can be used to tune the response of this gyro-free dynamic model. In the simulation studies that follow and the experimental validation we present the performance of this gyro-free EKF implementation.

It should be noted that practical considerations will require that the algorithm for the gyro-free implementation described above be modified when used in real-time applications. Specifically, the algorithm will have to be modified to deal with issues such as sensor data dropouts, latencies in transmission of the data from the sensor to the processing computer and asynchronous measurements of Earth's magnetic and gravity field vectors. In the simulation and experimental data (postprocessed) that are presented next, all these issues had to be considered.

If the attitude solution is used in applications where solution latency is not an issue (e.g., postmission data analysis), then the issues of sensor data latency and asynchronism can be dealt with by intelligent data processing. Data dropouts, however, require some knowledge of the vehicle's dynamics to propagate the attitude solution during the intervals of missing sensor data. All three of these issues can be dealt with successfully by incorporating rate gyros to provide data at a higher rate than the Earth magnetic and gravity field measurements [7, 25].

ALGORITHM PERFORMANCE

A series of simulations are used to evaluate the performance of the attitude estimators developed above. A vehicle equipped with a WAAS capable GPS receiver, a triad of magnetometers, and a triad of accelerometers, is simulated. In the first series of simulations we assume the vehicle is static and located at the Minneapolis–St.Paul International Airport, USA (N44.9° latitude, E93.2° longitude and 256 m above mean sea level). At this location, $\vec{h}^n$ and $\vec{g}^n$ are given by [30, 31]

$$\vec{h}^n = [0.1770 \;\; 0.0075 \;\; 0.5525]^T \text{ Gauss} \tag{66}$$

$$\vec{g}^n = [0 \;\; 0 \;\; 9.8069]^T \text{ m/s}^2. \tag{67}$$

These vectors are assumed to be exact and the error in the attitude estimation is introduced primarily by the sensor used to measure these vectors. This is a reasonable assumption since we are dealing with MAV applications where low-cost sensors will be used. In the second set of simulations we assume the vehicle is maneuvering in the vicinity of the Minneapolis–St. Paul International Airport.

Static Simulations

The static simulations consist of one million Monte-Carlo runs. The objective of these is to analyze the convergence and accuracy performance of the attitude algorithm. For each Monte Carlo run, an Euler angle triad was picked randomly from a uniformly distributed population that spanned the 3-2-1 Euler angle space. The starting yaw angle ψ came from a uniform population between $\pm 180°$, the pitch angle θ from a uniform population between $\pm 80°$, and the roll angle ϕ from a uniform population between $\pm 180°$. The pitch angle was limited to $\pm 80°$ in order to avoid the well-known Euler angle singularity at $\theta = \pm 90°$. This does not mean that the attitude algorithm has a singularity; the attitude algorithm is, in fact, singularity free. Since we are, however, using Euler angles for generating and visualizing simulation data, a pitch angle $\theta = \pm 90°$ would introduce unnecessary complications. Once the Euler angle triad was picked, the body-fixed sensor measurements corresponding to this attitude were generated. These body-fixed measurements were corrupted with appropriate levels of sensor wideband noise. For the magnetometer measurements, the measurement noise σ_h was 1 milli-Gauss and for the accelerometer measurements, σ_f was 1 milli-g. These are reasonable error magnitude for inexpensive magnetometers and accelerometers [6]. Since the vehicle is known to be static, the GPS derived acceleration measurements were not used in the attitude solution. That is, R_a is set equal to zero. For each Monte Carlo run the algorithm is always initialized with a starting guess for $\hat{\mathbf{q}}$ of $[1 \;\; 0 \;\; 0 \;\; 0]^T$ which corresponds to a 3-2-1 Euler angle triad of $(\psi, \theta, \phi) = (0, 0, 0)$. The algorithm was given 10 iterations to converge.

The overall convergence characteristics of the algorithm are very good; convergence is rapid and

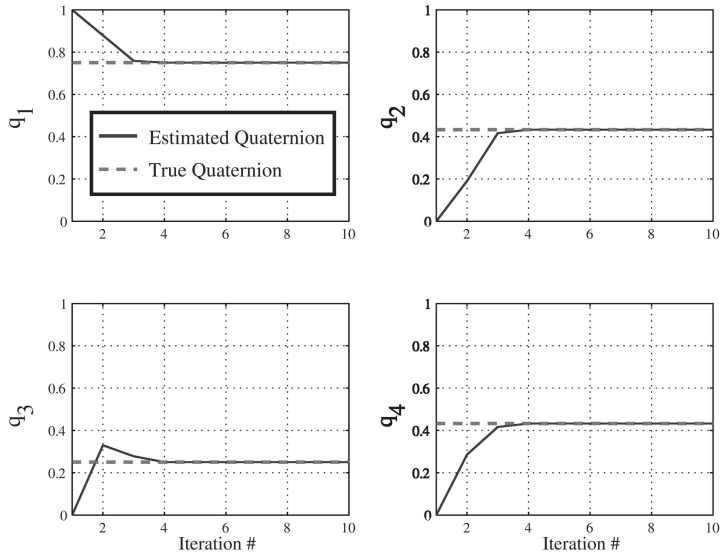


Fig. 3. Quaternion convergence history.

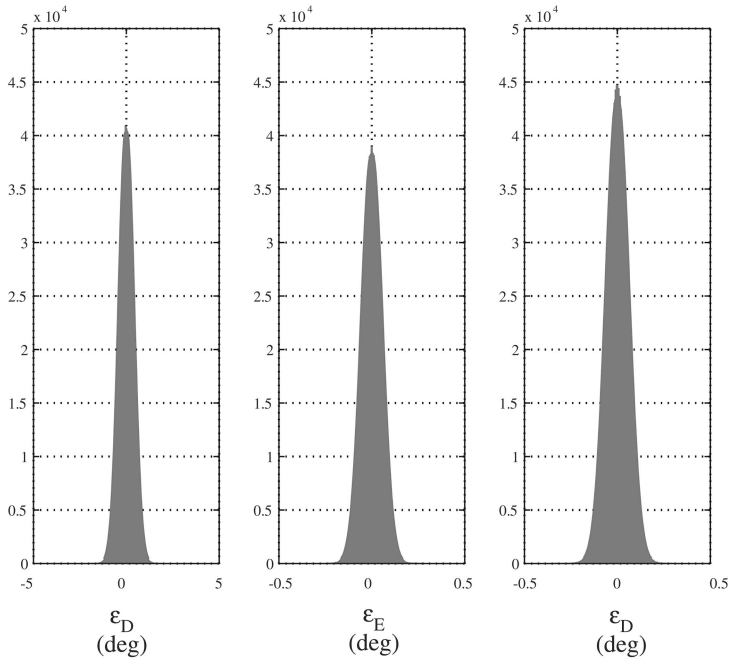


Fig. 4. Histogram of attitude errors for 1,000,000 Monte Carlo runs.

assured. The rapid convergence is shown in Fig. 3. This figure is a time history for the attitude quaternion components (as a function of iteration) during one of the Monte Carlo runs. As can be seen, the algorithm converges the correct solution in less than five iterations. Note that the assumed initial condition of $\hat{\mathbf{q}} = [1\ 0\ 0\ 0]^T$ (which is equivalent to the 3-2-1 Euler angle triad of $(\psi,\theta,\phi) = (0,0,0)$ is not always close to the initial guess. Thus, although the formulation of this solution methodology assumed a small q_e, the algorithm never diverged in all of the 1 million Monte Carlo runs.

The accuracy characteristics of the algorithm are captured in Fig. 4 which shows a histogram of attitude errors for all one million Monte Carlo runs. These errors are computed from the product $\overset{n\to b}{C}\ \overset{b\to n'}{C}$ (a rearrangement of (16)) where $\overset{n\to b}{C}$ is computed from the known true attitude and $\overset{b\to n'}{C}$ is computed using the attitude estimates generated by the algorithm after 10 iterations. The 1 σ standard deviation for all the angles is less than 1 deg. From Fig. 4 it is apparent that ϵ_D is the largest error. This is a result of the degree of observability of certain angles from the vectors selected; it is not an inherent limitation of the algorithm itself. When Earth's gravitational acceleration vector is one of the vectors used for

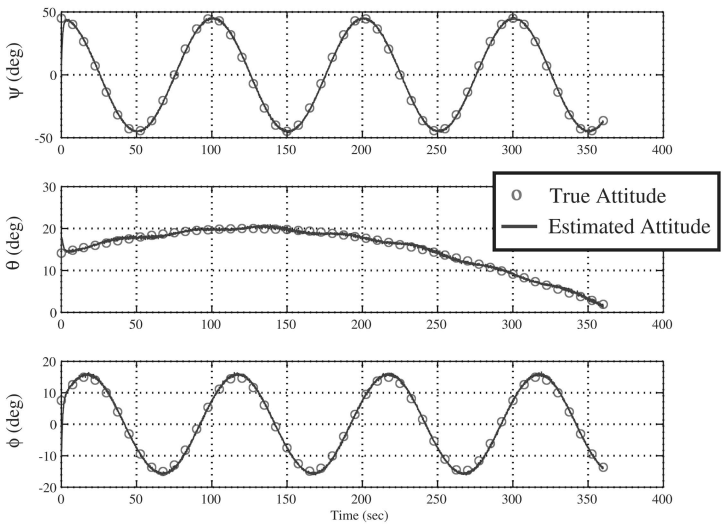


Fig. 5. Time history for true and estimated (EKF implementation) attitude. Dynamic simulation.

the solution of the attitude problem, errors about the down axis (ϵ_D) are not observable from this vector. That is, using measurements of only the gravitational acceleration, heading information cannot be extracted. All information about ϵ_D comes from the magnetic field vector $\vec{h}$. The other two error angles, on the other hand, are observable from both vectors and the estimation algorithm uses this redundant information.

Dynamic Simulations

The EKF formulation of the attitude algorithm is tested on a simulation of the same vehicle in motion. The vehicle is simulated performing maneuvers that include accelerations and attitude changes. The magnetometer and accelerometer triad error statistics are the same as was used in the static simulation. The values used in the covariance matrix $R_{\dot{x}}$ (which is used to compute R_a) are based on work in [33] which characterized the WAAS velocity errors velocity, and is set to be:

$$R_{\dot{x}} = \begin{bmatrix} (0.037)^2 & 0 & 0 \\ 0 & (0.042)^2 & 0 \\ 0 & 0 & (0.098)^2 \end{bmatrix} \text{ m/s}^2. \tag{68}$$

The time constant for the quaternion error correlation was set to 1 s (i.e., $\tau = 1$). The process noise covariance matrix was varied to adjust the performance of the attitude estimator. More specifically, R_w was set equal to

$$R_w = \alpha I_{3\times 3}. \tag{69}$$

The parameter α was used at the tuning parameter. With α set to a small number (low process noise), the estimator places less importance on the vector observations. To place more importance on the vector observations, the value of α is increased (high process noise).

Figs. 5–9 show the results for the dynamic simulations. Fig. 5 shows the true attitude history and the performance of the EKF formulation with the process noise matrix set high ($\alpha = 0.08727$). This value was established by trial and error and it is up to the designer of the estimator to select an α value that is appropriate. Fig. 6 shows the error in the attitude estimate or error transient for the first 10 s. It is clear that that the attitude errors (or residuals) converge to zero very quickly. Fig. 7 shows the errors 100 s into the simulation. That attitude residual is noisy and is time varying. The magnitude of the noise on the residuals is larger than the static case. This is due to the GPS derived acceleration signal used in the dynamic case and it should be noted that this is the measurement that introduces most of the noise into the attitude solution. The time-varying nature of the errors is due to the lag in the attitude solution. That is, the dynamic model used in the EKF formulation assumes a correlation time for the quaternion errors but that does not necessarily reflect the actual attitude dynamics of the vehicle.

The noise on the residuals can be reduced by using a smaller process noise matrix. Figs. 8 and 9 show the performance of the EKF estimator when $\alpha = 1.5708 \times 10^{-3}$. From these two figures we see that the wideband noise on the attitude residual is reduced. However, this comes at the expense of introducing a larger lag into the attitude solution. This lag manifests itself as correlated component in the attitude residual as shown in Fig. 9. If rate gyros were used to capture the true attitude dynamics, then this lag can be eliminated. In this case, the attitude estimator will also provide a means by which the gyro errors can be estimated and, thus, result in an overall higher performance attitude determination system [7].

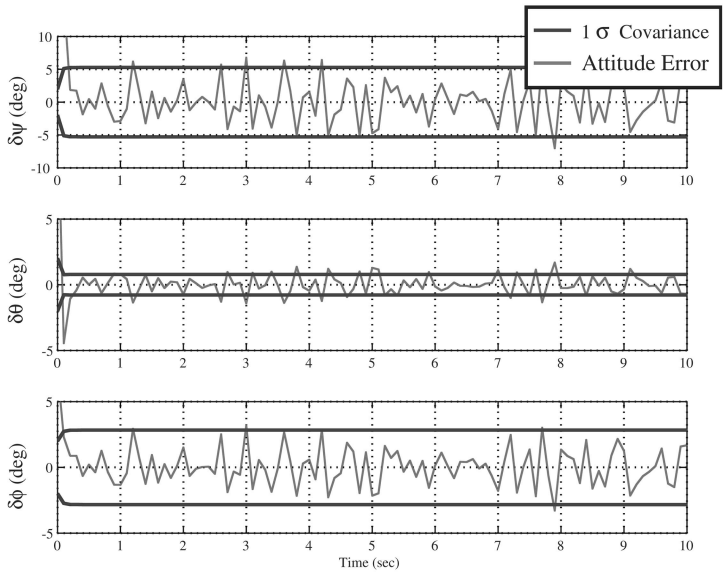


Fig. 6. Initial transient of attitude error history for simulation shown in Fig. 5. High process noise ($\alpha = 0.08727$).

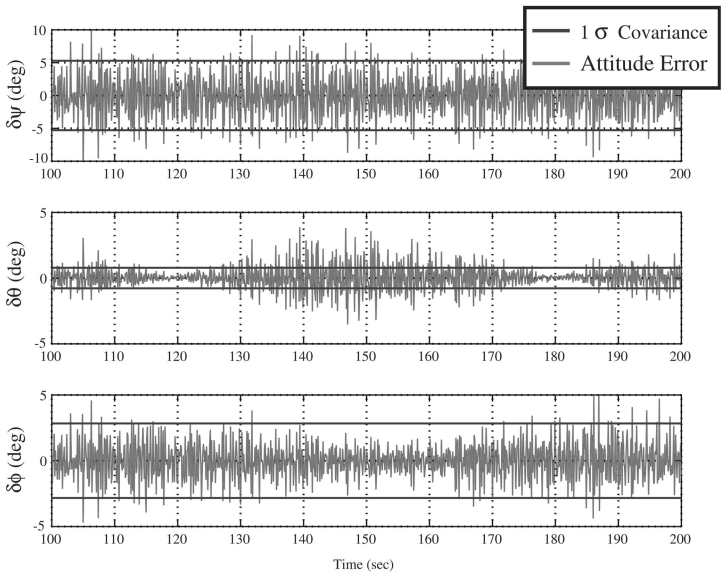


Fig. 7. Steady state attitude error history for simulation shown in Fig. 5. High process noise ($\alpha = 0.08727$).

EXPERIMENTAL VALIDATION

Experimental validation of the algorithm was accomplished by postprocessing data collected from a WAAS enabled GPS receiver, a magnetometer triad and an accelerometer triad flown on a general aviation test aircraft. A general aviation test aircraft and not an MAV was used because the test aircraft was large enough to carry several high-quality sensors that were used as a truth references. The truth reference for attitude was a Honeywell HG-1150 navigation-grade inertial reference unit (IRU) (1 nm/hr drift). The attitude accuracy of this system was approximately 0.05 deg in pitch and roll and 0.1 deg in yaw. The body-fixed accelerometer measurements ($\hat{\vec{g}}^b$) were obtained from inexpensive solid state accelerometers in a Crossbow DMU-FOG. The local level acceleration $\vec{a}$ was computed by differencing velocities derived from GPS augmented by the WAAS. A low-cost magnetometer triad (Honeywell HMR 2300) was used to measure the Earth's magnetic field vector in the body frame $\hat{\vec{h}}^b$. The magnetometer required calibration for misalignment errors, hard and soft iron errors (bias), and scale factors errors. This was done after the flight using the algorithm discussed in [32]. The data from all the sensors was time-tagged and recorded at 1 Hz for subsequent processing.

A 10 min portion of the flight test trajectory is shown in Fig. 10. From the trajectory it can be seen that the aircraft experienced accelerations and decelerations (several turns) as well as attitude

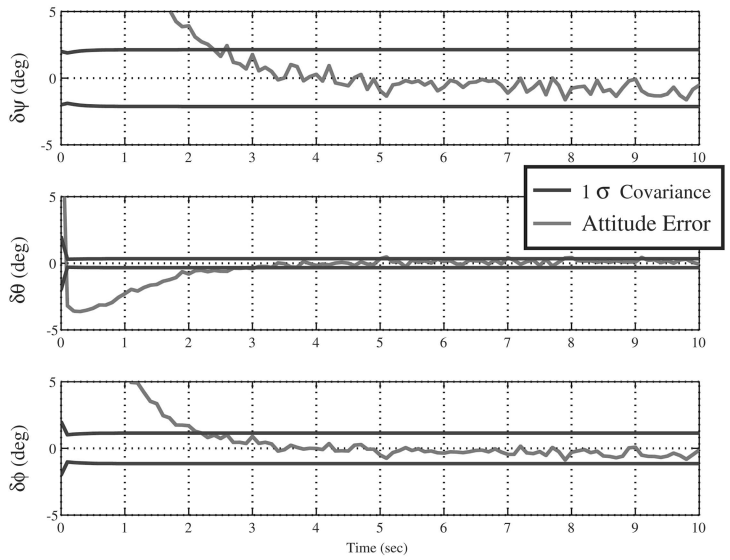


Fig. 8. Initial transient of attitude error history for simulation shown in Fig. 5. Low process noise ($\alpha = 1.5708 \times 10^{-3}$).

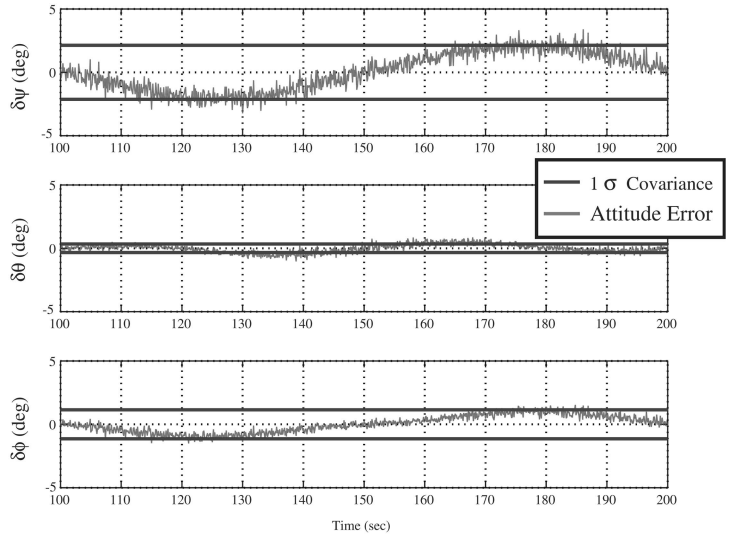


Fig. 9. Steady state attitude error history for simulation shown in Fig. 5. Low process noise ($\alpha = 1.5708 \times 10^{-3}$). Oscillations of errors are due to lag in attitude solution.

changes about all three axes. The 3-2-1 Euler angle history corresponding to this segment of flight is shown in Fig. 11. The true attitude trajectory was recorded by the high quality Honeywell IRU. The attitude solution determined by EKF implementation of the attitude algorithm is plotted along with the true attitude in Fig. 11. As can be seen the EKF implementation generated an attitude solution that was generally in agreement with the truth reference. For this flight, the means and standard deviations for the attitude errors (computed by taking the difference between the EKF solution and truth reference) are summarized in Table I in the row labeled "Nominal." This is the nominal performance of the EKF implementation of this estimator. There are a couple of points (between minutes 21 and 22 as well as between minutes 29 and 30) where the EKF solution appears to diverge. This divergence is due to the fact that the GPS derived acceleration becomes very noisy. This occurs in prolonged steep turns and results in the computed $\vec{g}^n$ being larger than its true value. This can be seen by examining the time history plot for the computed normalized gravity vector ($\hat{\vec{g}}^n / \|\hat{\vec{g}}^n\|$). The norm of this vector should always be unity but it is seen to exceed this value at the points where the attitude algorithm generates an estimate with large errors. If a gyro triad is included in this estimation scheme, it can easily coast through these momentary error spikes. If we remove the error spikes from the error statistics, we get the accuracies reported in Table I in the row labeled "Best." Since the attitude solution described above was generated

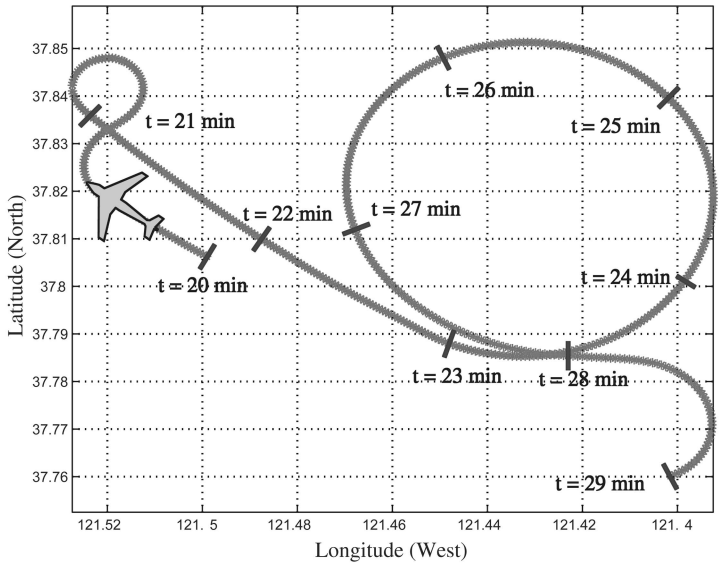


Fig. 10. Trajectory of test aircraft.

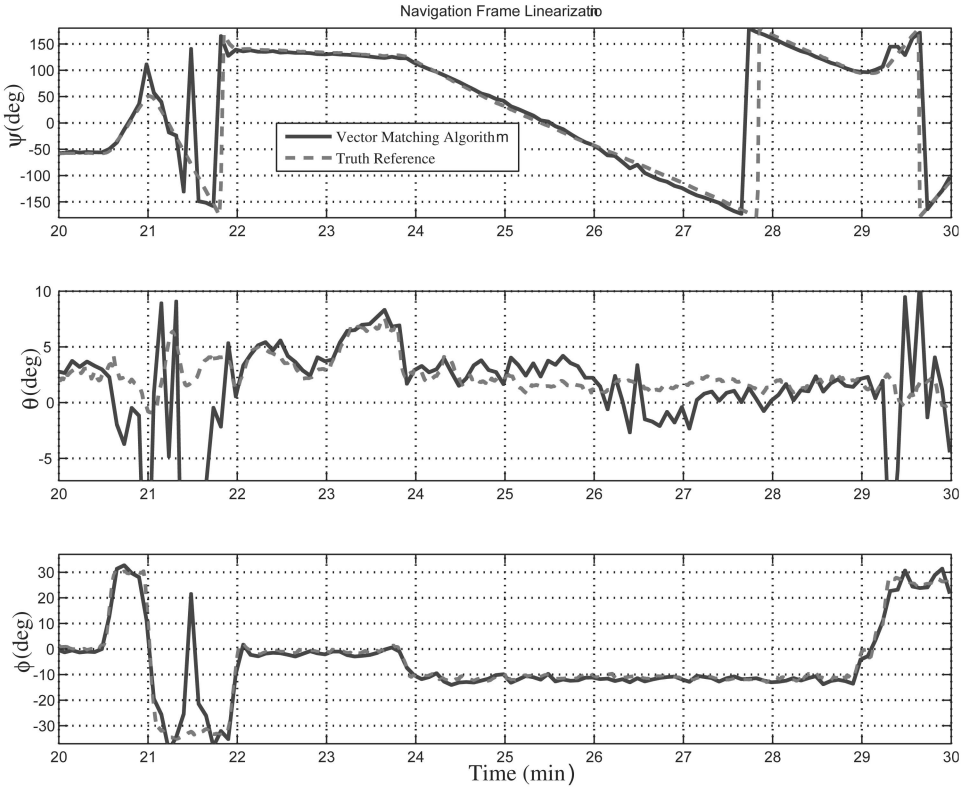


Fig. 11. Attitute history for postprocessed flight data.

in postprocess, sensor data latency or asynchronism was not an issue. In a real-time system where these are issues, the inclusion of rate gyros will also help mitigate their negative effects.

Incorporating rate gyros into the system architecture has another additional benefit. Considering the frequency content of the attitude history shown in Fig. 11, the trajectory shown in Fig. 10 is clearly benign. Maneuvers encountered in MAV operations will be more severe in that they will contain higher dynamics or frequency content. Thus, in real-time MAV applications, the gyro-free implementation of this algorithm may not be ideal. The effect of severe attitude dynamics can be dealt with by incorporating rate gyros into the system architecture. Architectures which fuse rate gyros with an aiding system are not new ([7] and [25]) and can be found in many commercially available systems

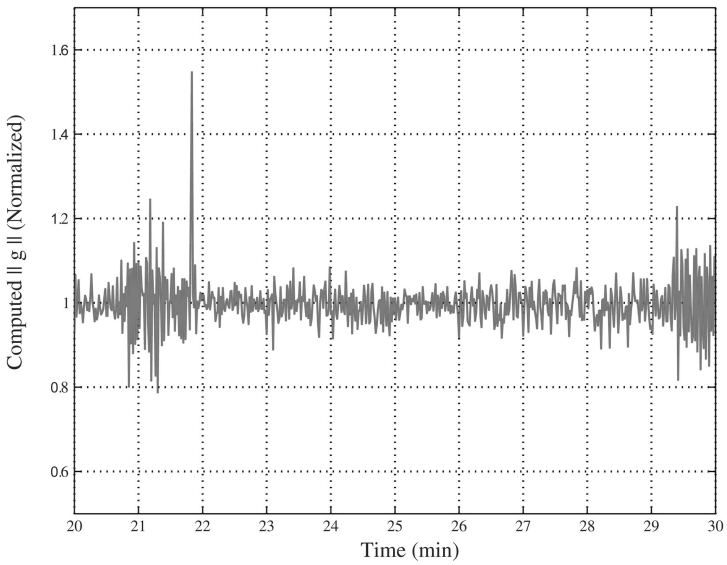


Fig. 12. Magnitude of $\vec{g}$ computed from accelerometer outputs and differentiation of GPS velocity.

TABLE I
Attitude Error Statistics from Flight Test Data
(Nominal and Best Performance)

Performance	σ_ψ (deg)	σ_θ (deg)	σ_ϕ (deg)
Nominal	18.2	5.04	5.48
Best	5.46	1.38	0.739

aimed at MAV or low-cost unmanned aerial vehicle (UAV) applications. However, most of these systems use an aiding which is based on accelerometer measurements only. Such systems perform poorly in prolonged accelerated maneuvers such as the turns shown between $t = 24$ min and $t = 29$ min of Fig. 11. Thus, using the algorithm developed in this paper as an aiding system for a triad of rate gyros will provide an attitude determination system capable of dealing with the high dynamics as well as maneuvers with prolonged accelerations encountered in many missions for which MAVs are being considered (e.g., surveillance).

CONCLUSIONS

In this work, we developed an algorithm and system for attitude determination from vector observation suitable for MAV applications. As we have shown it is particulary suited for MAV applications because it can be implemented using small, inexpensive sensors having low power consumption. The sensor used in the implementation presented were a WAAS enabled GPS receiver, a magnetometer triad, and an accelerometer triad. Simulation and flight test results show that the algorithm and system yield an attitude solution with errors approximately 5 deg (1 σ) in yaw and 1 deg (1 σ) in roll and pitch. Since it is a gyro-free implementation, the attitude errors do not grow with time.

The largest source of error was shown to be the noise from the GPS derived acceleration. However, the effect of this noise can be easily mitigated by incorporating rate gyros into the attitude estimation algorithm. Incorporating rate gyros can also help deal with issues of data dropout, latency and asynchronism. The rate gyros provide a dynamic model which allows coasting through the time periods of missing data. They also allow propagating the attitude solution forward in time until the latent or asynchronous Earth magnetic and gravity field vector measurements are processed. A framework for incorporating rate gyros was briefly discussed in this paper but is discussed in more detail in other published works such as [7] and [25]. Another possible method for mitigating the above noted errors would be to use other vector observations such as velocity measurements (from Doppler radars) or visual line of sight vectors to objects obtained from cameras.

ACKNOWLEDGMENTS

The work in this paper is the result of research sponsored by the Intelligent Transportation System (ITS) Institute at the University of Minnesota. The initial part of this research was conducted and the flight test data used to validate the algorithm's performance was obtained from research sponsored by the Federal Aviation Adminstration (FAA) Satellite Program Office under Grant 95-G-005. The authors gratefully acknowledge the FAA, the Principal Investigator of this project (Dr. Per Enge, Stanford University) and Co-Principal Investigators (Dr. J. David Powell and Dr. Bradford Parkinson both Stanford University) for support and sharing the flight test data.

REFERENCES

[1] McMichael, J. M., and Francis, M. S.
Micro air vehicles-Toward a new dimension in flight.
Defense Advanced Research Projects Agency (DARPA) Briefing, http://euler.aero.iitb.ac.in/docs/MAV/www.darpa.mil/tto/MAV/mavauvsi.html.

[2] Office of the Secretary of Defense
Unmanned aircraft systems roadmap 2005–2030.
Dec. 2002, http://www.acq.osd.mil/usd/uavroadmap.pdf.

[3] Microbotics Inc.
MIDG II specifications.
Microbotics Inc., Hampton, VA, Aug. 2005.

[4] Xsens Technologies
MT9 inertial 3D motion tracker specification sheet.
Xsens Technologies, B.V., Enschede, Netherlands, 2005.

[5] Cloud Cap Technology
Crista inertial measurement unit, interface/operation document.
Cloud Cap Technology Inc., Hood River, OR, Feb. 2008.

[6] Gebre-Egziabher, D.
Design and performance analysis of a low-cost aided-dead reckoning navigation system.
Ph.D. dissertation, Dept. of Aeronautics and Astronautics, Stanford University, Stanford, CA, Dec. 2001, 68–90.

[7] Gebre-Egziabher, D., Hayward, R. C., and Powell, J. D.
Design of Multi-sensor attitude determination systems.
IEEE Transactions on Aerospace and Electronic Systems, **40**, 2 (2004), 627–643.

[8] Gebre-Egziabher, D., Elkaim, G. H., Powell, J. D., and Parkinson, B. W.
A gyro-free quaternion based attitude determination system suitable for implementation using low-cost sensors.
In *Proceedings of the IEEE PLANS 2002*, San Diego, CA, 2002, 185–192.

[9] Cohen, C. E.
Attitude determination using GPS.
Ph.D. dissertation, Dept. of Aeronautics and Astronautics, Stanford University, Stanford, CA, 1992.

[10] Kornfeld, R., Hansman, R. J., and Deyst, J.
Single antenna GPS-based aircraft attitude determination.
Navigation: Journal of the Institute of Navigation, **45**, 1 (Spring 1998), 51–60.

[11] Gebre-Egziabher, D., Hayward, R. C., and Powell, J. D.
A low cost GPS/inertial attitude heading reference system (AHRS) for general aviation applications.
In *Proceeding of the IEEE PLANS 1998*. Rancho Mirage, CA, Apr. 21–23, 1998, 518–525.

[12] Wahba, G.
A least squares estimate of spacecraft attitude.
SIAM, Review, **7**, 3 (1965), 409.

[13] Wahba, G.
Problem 65-1 (solution).
SIAM, Review, **8** (1966), 384–386.

[14] Black, H. D.
A passive system for determining the attitude of a satellite.
AIAA Journal, **2**, 7 (1964), 1350–1351.

[15] Lerner, G. M.
Three-axis attitude determination.
In J. R. Werts and D. Reidel (Eds.), *Spacecraft Attitude Determination and Control*, D. Reidel Publishing Co., Dordrecht, Netherlands, 1978, 420–428.

[16] Thompson, I. C., and Quasius, G. R.
Attitude determination for the P80-1 satellite (AIAA paper 80-001).
In *Proceedings of AAS Guidance and Control Conference*, Keystone, CO, 1980.

[17] Shuster, M. D. and Oh, S. D.
Three-axis attitude determination from vector observations.
Journal of the Astronautical Sciences, **48**, 1 (1981), 70–77.

[18] Bar-Itzhack, I. Y.
REQUEST: A recursive QUEST algorithm for sequential attitude determination.
Journal of Guidance, Control and Dynamics, **19**, 5 (1996), 1034–1038.

[19] Bar-Itzhack, I. Y., and Idan, M.
Recursive attitude determination from vector observations: Euler angle estimation.
AIAA Journal of Guidance, Control and Dynamics, **AES-10**, 2 (1987), 152–157.

[20] Bar-Itzhack, I. Y., and Oshman, Y.
Attitude determination from vector observations: Quaternion estimation.
IEEE Transactions on Aerospace and Electronic Systems, **21**, 1 (1985), 128–136.

[21] Bar-Itzhack, I. Y. and Reiner, J.
Recursive attitude determination from vector observations: Direction cosine matrix identification.
AIAA Journal of Guidance, Control and Dynamics, **7**, 1 (1984), 51–56.

[22] Idan, M.
Estimation of Rodrigues parameters from vector observations.
IEEE Transaction on Aerospace and Electronic Systems, **32**, 2 (1996), 578–585.

[23] Choukroun, D., Bar-Itzhack, I. Y., and Oshman, Y.
Novel quaternion Kalman filter.
IEEE Transaction on Aerospace and Electronic Systems, **42**, 1 (2006), 174–190.

[24] Psiaki, M. L.
Attitude-determination filtering via extended quaternion estimation.
Journal of Guidance, Control and Dynamics, **23**, 2 (2000), 206–214.

[25] Creamer, G.
Spacecraft attitude determination using gyros and quaternion measurements.
The Journal of Astronautical Sciences, **44**, 3 (1996), 357–371.

[26] Lefferts, E. J., Markley, F. L., and Shuster, M. D.
Kalman filtering for spacecraft attitude estimation.
AIAA Journal of Guidance, Control and Dynamics, **5**, 5 (1982), 417–429.

[27] Siouris, G. M.
Aerospace Avionics Systems: A Modern Synthesis.
San Diego, CA: Academic Press, 1993, 231–241.

[28] Kuipers, J. B.
Quaternions and Rotation Sequences.
Princeton, NJ: Princeton University Press, 2002.

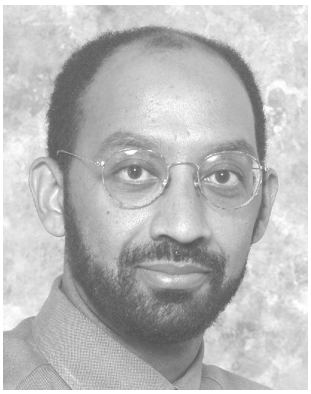
[29] Baruh, H.
Analytical Dynamics.
New York: McGraw-Hill, 1999.

[30] Barton, C. E.
International geomagnetic reference field: The seventh generation.
Journal of Geomagnetisim and Geoelectricity, **49**, 2 (1997), 123–148.

[31] National Geospastial Intelligence Agency
Department of Defense World Geodetic System 1984, Its Definition and Relationships With Local Geodetic Systems.
NIMA Technical Report TR8350.2, 3rd ed., July 4, 1997.

[32] Gebre-Egziabher, D., Elakaim, G. H., Powell, J. D., and Parkinson, B. W.
Calibration of strapdown magnetometers in magnetic field domain.
Journal of Aerospace Engineering, **19**, 2 (Apr. 2006), 87–102.

[33] Houck, S., and Powell, J. D.
Visual cruise formation flying dynamics.
AIAA Atmospheric Flight Mechanics Conference, Denver, CO, Aug. 14–17, 2000, Collection of Technical Papers (A00-39676 10-08), AIAA Paper AIAA-2000-4316.



Demoz Gebre-Egziabher holds a Ph.D. in aeronautics and astronautics from Stanford University.

He is an Assistant Professor of Aerospace Engineering and Mechanics at the University of Minnesota, Twin Cities Campus, Minneapolis, MN. His research is in the areas navigation, guidance and control with a particular emphasis on application of estimation theory to avionics sensor fusion and system integration issues.

Gabriel Hugh Elkaim received his B.S. degree in mechanical/aerospace engineering from Princeton University, Princeton, NJ, in 1990, and the M.S. and Ph.D. Degrees from Stanford University, Stanford, CA, in aeronautics and astronautics, in 1995 and 2002, respectively.

In 2003, he joined the faculty of the Jack Baskin School of Engineering, at the UC Santa Cruz where he is an assistant professor in the Computer Engineering Department. His research interests include control systems, sensor fusion, GPS, system identification, and autonomous vehicle systems. His research focuses on intelligent autonomous vehicles, with an emphasis on robust guidance, navigation, and control strategies. Specifically, he has founded the Autonomous Systems Lab at UC Santa Cruz, and is currently developing an autonomous wing-sailed marine surface vehicle and off-road autonomous ground vehicles.