N92-14092

QUATERNION NORMALIZATION IN ADDITIVE EKF FOR SPACECRAFT ATTITUDE DETERMINATION

by

I.Y. Bar-Itzhack*, J. Deutschmann+, and F.L. Markley#

Abstract

This work introduces, examines and compares several quaternion normalization algorithms, which are shown to be an effective stage in the application of the additive extended Kalman filter (EKF) to spacecraft attitude determination, which is based on vector measurements. Two new normalization schemes are introduced. They are compared with one another and with the known brute force normalization scheme, and their efficiency is examined. Simulated satellite data are used to demonstrate the performance of all three schemes. A fourth scheme is suggested for future research.

Although the schemes were tested for spacecraft attitude determination, the conclusions are general and hold for attitude determination of any three dimensional body when based on vector measurements, and use an additive EKF for estimation, and the quaternion for specifying the attitude.

---

* Professor, Faculty of Aerospace Engineering, Technion - Israel Institute of Technology, member Technion Space Research Institute, Haifa, Israel 32000. Tel. (972)4-293196.

+ Aerospace Engineer, Attitude Analysis Section, Flight Dynamics Analysis Branch, NASA Goddard Space Flight Center, Greenbelt, MD 20771. Tel. (301) 286-9033.

Assistant Head, Guidance and Control Branch, NASA Goddard Space Flight Center, Greenbelt, MD 20771. Tel. (301) 286-8333.

I. INTRODUCTION

Attitude determination of spacecraft usually utilizes vector measurements such as Sun, center of Earth, star, and magnetic field direction to update the quaternion which determines the spacecraft orientation with respect to some reference coordinates in the three dimensional space [1,2,3]. These measurements are usually processed by an extended Kalman filter (EKF) which yields an estimate of the attitude quaternion [4-8].

Two EKF versions for quaternion estimation were presented in the literature; namely, the multiplicative EKF [4-6] and the additive EKF [5,7,8]. In the multiplicative EKF it is assumed that the error between the correct quaternion and its *a-priori* estimate is, by itself, a quaternion that represents the rotation necessary to bring the attitude which corresponds to the *a-priori* estimate of the quaternion into coincidence with the correct attitude. The EKF basically estimates this quotient quaternion and then the updated quaternion estimate is obtained by the *product* of the *a-priori* quaternion estimate and the estimate of the difference quaternion. In the additive EKF it is assumed that the error between the *a-priori* quaternion estimate and the correct one is an algebraic difference between two four-tuple elements and thus the EKF is set to estimate this difference. The updated quaternion is then computed by *adding* the estimate of the difference to the *a-priori* quaternion estimate.

If the quaternion estimate converges to the correct quaternion, then, naturally, the quaternion estimate has unity norm. This fact was utilized in the past to obtain superior filter performance by applying normalization to the filter measurement update of the quaternion [7]. It was observed for the additive EKF that when the attitude changed very slowly between measurements, normalization merely resulted in a faster convergence [7,8]; however, when the attitude changed considerably between measurements, without filter tuning or normalization, the quaternion estimate diverged. However, when the quaternion estimate was normalized, the estimate converged faster and to a lower error than with tuning only.

In the next section we introduce the additive EKF for attitude determination. The role of quaternion normalization in the additive EKF is explained in Section III. In Section IV we discuss the brute force (BF) normalization scheme and examine its performance. In the following sections we introduce the quaternion pseudo-measurement (QPM), and the magnitude pseudo-measurement (MPM). Test results of the application of all normalization algorithms discussed in this work to simulated Earth Radiation Budget Satellite (ERBS) data is presented in Section VII. In Section VIII we introduce the linearized orthogonalized matrix (LOM) normalization scheme as a suggestion for future investigation. Finally, the conclusions of this work are discussed in Section IX.

II. THE ADDITIVE EKF FOR QUATERNION ESTIMATION

Attitude determination from vector observations using the additive EKF is explained as follows. Suppose that a sequence $\underline{v}_{bm,i}$ $i=0,1,2,...$ of vector measurements performed in body, **b**, coordinates are given. Given are also these vectors in the reference coordinate system **r**. Denote the latter vectors by $\underline{v}_{r,i}$ $i=0,1,2,....$. The vector $\underline{v}_{bm,i}$ is a column matrix whose elements are the components of a vector $\bar{v}$ measured at time t_i and coordinatized in the body coordinate system. Similarly, the corresponding $\underline{v}_{r,i}$ vector is a column matrix whose elements are the components of the same vector $\bar{v}$ coordinatized in the reference coordinate system. Our aim is to estimate the quaternion $\underline{q}$ which expresses the body attitude with respect to the reference coordinate system. To meet this end we define an *effective measurement* $\underline{y}$ as follows

$$\underline{y}_i = \underline{v}_{bm,i} - A(\hat{\underline{q}})\underline{v}_{r,i} \tag{2.1}$$

where A is the direction cosine matrix (DCM) which transforms vectors from r to b, and where $\hat{\underline{q}}$ is the latest estimate of $\underline{q}$. The vector $\underline{v}_{bm,i}$ which is a result of a measurement, contains all the error associated with the instrumentation, such as instrument misalignments, scale factor error, bias, white noise etc. The vector $\underline{v}_{r,i}$ is taken from the almanac and is assumed to be perfectly known. We observe that when the measurement is error free and when the quaternion estimate is accurate, $\underline{y}_i$ is zero. On the other hand, when these assumptions do not hold, then $\underline{y}_i$ is a, generally non-linear, function of the instrument and attitude errors.

The measured vector $\underline{v}_{bm,i}$ can be expressed as follows

$$\underline{v}_{bm,i} = \underline{v}_{b,i} + \left.\frac{\partial \underline{v}}{\partial \underline{e}}\right|_{\underline{v}_{b,i}} (\delta\underline{e} + \underline{n}_i) \tag{2.2}$$

where $\underline{v}_{b,i}$ is the error-free value of $\bar{v}$ when coordinatized in the b system. the Jacobian matrix

$$H_{e,i} \triangleq \left.\frac{\partial \underline{v}}{\partial \underline{e}}\right|_{\underline{v}_{b,i}} \tag{2.3}$$

is the sensitivity matrix of the error associated with the measurement $\underline{v}_{bm,i}$ as a function of the instrument errors. The latter are expressed as a sum of a narrow spectrum error vector, $\delta\underline{e}$, and a wide spectrum error vector $\underline{n}_i$, which is modeled as a white noise error vector. The vector $\underline{e}$ contains all the instrumentation errors mentioned before, while $\delta\underline{e}$ denotes the difference between $\underline{e}$ and its compensation value which is the latest estimate of $\underline{e}$ denoted by $\hat{\underline{e}}$.

Define $\delta\underline{q}$ as follows

$$\delta\underline{q} = \underline{q} - \hat{\underline{q}} \tag{2.4}$$

then

$$A(\underline{q}) = A(\hat{\underline{q}}+\delta\underline{q}) \tag{2.5}$$

therefore, based on the assumption that $\delta\underline{q}$ is small such that $\hat{\underline{q}}$ is close enough to $\underline{q}$, $A(\underline{q})$ can be approximated as follows

$$A(\underline{q}) = A(\hat{\underline{q}}) + \sum_{j=1}^{4} \left. \frac{\partial A(\underline{q})}{\partial q_j} \right|_{\hat{\underline{q}}} \delta q_j \tag{2.6}$$

consequently

$$A(\hat{\underline{q}})\underline{v}_{r,i} = A(\underline{q})\underline{v}_{r,i} - \sum_{j=1}^{4} \left. \frac{\partial A(\underline{q})}{\partial q_j} \right|_{\hat{\underline{q}}} \delta \underline{q}_j \tag{2.7}$$

Define

$$G_j = G_j(\hat{\underline{q}}) = \left. \frac{\partial A(\underline{q})}{\partial q_j} \right|_{\hat{\underline{q}}} \tag{2.8}$$

Since

$$A(\underline{q}) = \begin{bmatrix} q_1^2-q_2^2-q_3^2+q_4^2 & 2(q_1q_2+q_3q_4) & 2(q_1q_3-q_2q_4) \\ 2(q_1q_2-q_3q_4) & -q_1^2+q_2^2-q_3^2+q_4^2 & 2(q_2q_3+q_1q_4) \\ 2(q_1q_3+q_2q_4) & 2(q_2q_3-q_1q_4) & -q_1^2-q_2^2+q_3^2+q_4^4 \end{bmatrix} \tag{2.9}$$

then

$$G_1 = 2 \begin{bmatrix} \hat{q}_1 & \hat{q}_2 & \hat{q}_3 \\ \hat{q}_2 & -\hat{q}_1 & \hat{q}_4 \\ \hat{q}_3 & -\hat{q}_4 & -\hat{q}_1 \end{bmatrix} \quad (2.10a) \qquad G_2 = 2 \begin{bmatrix} -\hat{q}_2 & \hat{q}_1 & -\hat{q}_4 \\ \hat{q}_1 & \hat{q}_2 & \hat{q}_3 \\ \hat{q}_4 & \hat{q}_3 & -\hat{q}_2 \end{bmatrix} \quad (2.10b)$$

$$G_3 = 2 \begin{bmatrix} -\hat{q}_3 & \hat{q}_4 & \hat{q}_1 \\ -\hat{q}_4 & -\hat{q}_3 & \hat{q}_2 \\ \hat{q}_1 & \hat{q}_2 & \hat{q}_3 \end{bmatrix} \quad (2.10c) \qquad G_4 = 2 \begin{bmatrix} \hat{q}_4 & \hat{q}_3 & -\hat{q}_2 \\ -\hat{q}_3 & \hat{q}_4 & \hat{q}_2 \\ \hat{q}_2 & -\hat{q}_1 & \hat{q}_4 \end{bmatrix} \quad (2.10d)$$

Define

$$\underline{h}_j = G_j \underline{v}_{r,i} \tag{2.11}$$

and

$$H_{q,i} = \left[\underline{h}_1 | \underline{h}_2 | \underline{h}_3 | \underline{h}_4\right] \tag{2.12}$$

then using (2.12), (2.7) can be written as

$$A(\hat{\underline{q}})\underline{v}_{r,i} = A(\underline{q})\underline{v}_{r,i} - H_{q,i}\delta\underline{q} \tag{2.13}$$

Finally from (2.1), (2.2), (2.3) and (2.13) we obtain

$$\underline{y}_i = \underline{v}_{b,i} + H_{e,i}\delta\underline{e} + H_{e,i}\underline{n}_i - A(\underline{q})\underline{v}_{r,i} + H_{q,i}\delta\underline{q} \tag{2.14}$$

Since $\underline{v}_{b,i}$, $A(\underline{q})$ and $\underline{v}_{r,i}$ are error-free, it is clear that

$$\underline{v}_{b,i} = A(\underline{q})\underline{v}_{r,i} \tag{2.15}$$

therefore (2.14) can be written as

$$\underline{y}_i = H_{e,i}\delta\underline{e} + H_{q,i}\delta\underline{q} + \underline{n}_i^* \tag{2.16}$$

where

$$\underline{n}_i^* = H_{e,i}\underline{n}_i \tag{2.17}$$

Note that (2.16) can be written as

$$\underline{y}_i = \left[H_{q,i} | H_{e,i}\right] \begin{bmatrix} \delta\underline{q} \\ \delta\underline{e} \end{bmatrix} + \underline{n}_i^* \tag{2.18}$$

The propagation of the vector $[\delta\underline{q}^T | \delta\underline{e}^T]^T$ (where T denotes the transpose) in time can be expressed by the linear equation [8]

$$\frac{d}{dt}\begin{bmatrix} \delta\underline{q} \\ \delta\underline{e} \\ \delta\underline{p} \end{bmatrix} = \begin{bmatrix} & & \\ & F & \\ & & \end{bmatrix} \begin{bmatrix} \delta\underline{q} \\ \delta\underline{e} \\ \delta\underline{p} \end{bmatrix} + \underline{w} \tag{2.19}$$

where $\delta\underline{p}$ contains additional states necessary to express (2.19) as a linear equation driven by a white noise vector $\underline{w}$. For compatibility with (2.19), (2.18) is extended to include $\delta\underline{p}$ as follows

$$\underline{y}_i = [H_{q,i}|H_{e,i}|\ 0\ ] \begin{bmatrix} \delta\underline{q} \\ \delta\underline{e} \\ \delta\underline{p} \end{bmatrix} + \underline{n}_i \tag{2.20}$$

The set (2.19) and (2.20) can be written as

$$\dot{\underline{x}} = F\ \underline{x} + \underline{w} \tag{2.21a}$$

$$\underline{y}_i = H_i\underline{x} + \underline{n}_i^* \tag{2.21b}$$

where $H_i=[H_{q,i}|H_{e,i}|0]$. The latter equations can be used in an EKF to compute $\hat{\underline{x}}_i$, the estimate of $\underline{x}$ at time t_i.

Let $\underline{X}^T=[\underline{q}^T|\underline{e}^T|\underline{p}^T]$ then according to the EKF algorithm, $\hat{\underline{X}}_i(-)$, the a-priori estimate of $\underline{X}$ at time t_i is used to calculate H_i which is needed to obtain the a-posteriori estimate $\hat{\underline{x}}_i(+)$. The latter is then used to update the entire state estimate as follows

$$\hat{\underline{X}}_i(+) = \hat{\underline{X}}_i(-) + \hat{\underline{x}}_i(+) \tag{2.22}$$

Using (2.21a), $\hat{\underline{X}}_i(+)$ is propagated in time to become $\hat{\underline{X}}_{i+1}(-)$, the a-priori entire state estimate at time t_{i+1}. The dynamics matrix for the propagation of $\hat{\underline{X}}_i(+)$ is denoted by A (see (2.23a). The covariance which is needed for computing the Kalman gain necessary for evaluating $\hat{\underline{x}}_i(+)$, is computed according to the ordinary Kalman filter algorithm. To sum it up, the full EKF algorithm is as follows

Between measurements

Solve from t_i to t_{i+1}

$$\dot{\hat{\underline{X}}} = A[\hat{\underline{X}}(t),t]\hat{\underline{X}} \tag{2.23a}$$

$$\dot{P} = A[\hat{\underline{X}}(t),t]P + P\ A^T[\hat{\underline{X}}(t),t] + Q(t) \tag{2.23b}$$

with the initial conditions $\hat{X}_o=\hat{X}_i(+)$, $P_o=P_i(+)$. The solutions at t_{i+1} are denoted by $\hat{X}_{i+1}(-)$ and $P_{i+1}(-)$ respectively. Q(t) is the spectral density matrix [9] of $\underline{w}$.

Across measurements

$$K_{i+1} = P_{i+1}(-)H^T_{i+1}\left[H_{i+1}P_{i+1}(-)H^T_{i+1} + R_{i+1}\right]^{-1} \tag{2.23c}$$

$$\hat{\underline{x}}_{i+1}(+) = K_{i+1}\underline{y}_{i+1} \tag{2.23d}$$

$$\hat{\underline{X}}_{i+1}(+) = \hat{\underline{X}}_{i+1}(-) + \hat{\underline{x}}_{i+1}(+) \tag{2.23e}$$

$$P_{i+1}(+) = \left[I - K_{i+1}H_{i+1}\right] P_{i+1}(-)\left[I - K_{i+1}H_{i+1}\right]^T + K_{i+1}R_{i+1}K^T_{i+1} \tag{2.23f}$$

where R_{i+1} is the covariance of $\underline{n}^*_{i+1}$.

Compensation

In computing (2.23a) and (2.23b) we need to use the gyro output vector $\underline{w}$ which contains errors. Those errors are estimated as a part of $\hat{\underline{p}}$. Before each time (2.23a,b) are used, the errors have to be appropriately compensated using their estimate. Similarly, $\underline{v}_{bm,i}$, which is used in (2.1) to obtain $\underline{y}_i$, contains erros which constitute $\underline{e}$. Before each time $\underline{y}_i$ is computed (for use in (2.23d)), the errors in $\underline{v}_{bm,i}$ have to be compensated using their estimate.

III. THE ROLE OF QUATERNION NORMALIZATION

The state measurement update given in (2.23e) can be written in an explicit form as follows:

$$\begin{bmatrix} \hat{\underline{q}}(+) \\ \hat{\underline{e}}(+) \\ \hat{\underline{p}}(+) \end{bmatrix}_{i+1} = \begin{bmatrix} \hat{\underline{q}}(-) \\ \hat{\underline{e}}(-) \\ \hat{\underline{p}}(-) \end{bmatrix}_{i+1} + \begin{bmatrix} \delta\hat{\underline{q}}(+) \\ \delta\hat{\underline{e}}(+) \\ \delta\hat{\underline{p}}(+) \end{bmatrix}_{i+1} \tag{3.1}$$

Unless convergence has been attained, $\hat{\underline{q}}_{i+1}(+)$ is not necessarily normal even if $\hat{\underline{q}}_{i+1}(-)$ is. We know, however, that the quaternion which the algorithm is trying to estimate is necessarily normal. We can then enforce normalization on $\hat{\underline{q}}_{i+1}(+)$ with the hope that the enforcement of this quality of the correct quaternion will direct the estimated quaternion in the right track and will enhance its convergence. Indeed, it was found in the past [7,8] that normalization is helpful. In particular, it was found that when the attitude varies very slowly between measurements, normalization, although not necessary, resulted in a faster convergence; however, when the attitude changed rapidly between measurements either filter tuning or normalization were necessary to avoid divergence. The use of normalization is superior to tuning because, first, tuning involves a tedious trial and error process, second, tuning is not a robust solution, and third, with quaternion normalization the final attitude estimate is closer to the correct quaternion.

Four normalization schemes are discussed next.

IV. BRUTE FORCE (BF) NORMALIZATION

The BF normalization is performed as follows [7]. After $\hat{\underline{X}}_{i+1}(+)$ has been computed in (2.23e) the quaternion part of the state (see (3.1)) is normalized as follows

$$\hat{\underline{q}}^*_{i+1}(+) = \hat{\underline{q}}_{i+1}(+) / \| \hat{q}_{i+1}(+) \| \tag{4.1}$$

and then, the normal quaternion, $\hat{\underline{q}}^*_{i+1}(+)$, is used to re-form $\hat{\underline{X}}_{i+1}(+)$ as follows

$$\hat{\underline{X}}_{i+1}(+) = \begin{bmatrix} \hat{\underline{q}}^*(+) \\ \hat{\underline{e}}(+) \\ \hat{\underline{p}}(+) \end{bmatrix}_{i+1} \tag{4.2}$$

This straightforward mode of normalization constitutes an outside interference in the EKF algorithm which has to be accounted for in order to avoid filter divergence. It was shown [7] that the normalization operation of (4.1) is tantamount to the following computation of $\hat{q}^*_{i+1}(+)$

$$\hat{q}^*_{i+1}(+) = \left[\hat{\underline{q}}_{i+1}(-) + \delta\hat{\underline{q}}_{i+1}(+)\right] - \hat{\underline{q}}_{i+1}(-)\hat{\underline{q}}^T_{i+1}(-)\delta\hat{\underline{q}}_{i+1}(+) \tag{4.3}$$

Therefore, while the EKF algorithm presented in Section II assumes that the a-priori quaternion estimate is updated according to (2.23e) as follows

$$\hat{\underline{q}}_{i+1}(+) = \hat{q}_{i+1}(-) + \delta\hat{\underline{q}}_{i+1}(+) \tag{4.4}$$

in reality, due to the normalization, it is updated according to (4.3). The difference is then in the term $-\hat{q}_{i+1}(-)\hat{q}^T_{i+1}\delta\hat{\underline{q}}_{i+1}(+)$. Because of this residual term, the full reset implied by (4.4) does not hold anymore. Therefore, following the logic of the EKF algorithm, the residual term, $-\hat{q}_{i+1}(-)\hat{q}^T_{i+1}\delta\hat{\underline{q}}_{i+1}(+)$ has to be propagated in time. It was shown [7] that this mode of normalization does not affect the covariance computation of the EKF; therefore, only the state computation has to be modified. In view of the normalization operation of (4.1), the following changes have to be made in the EKF algorithm presented in Section II. Between measurements, in addition to the computation of $\hat{\underline{X}}_{i+1}(-)$ and $P_{i+1}(-)$, compute also $\delta\hat{q}_{i+1}(-)$ as follows. Solve from t_i to t_{i+1} the differential equation

$$\delta\dot{\hat{\underline{q}}} = F_q[\hat{\underline{X}}(t),t]\delta\hat{\underline{q}} \tag{4.5}$$

where F_q is the 1,1 submatrix of F, with the initial condition $\delta\hat{\underline{q}}_o = \hat{q}_i(-)\hat{q}^T_i(-)\delta\hat{\underline{q}}_i(+)$ and denote the solution at t_{i+1} by $\delta\hat{\underline{q}}_{i+1}(-)$. Then form

$$\hat{\underline{x}}^T_{i+1}(-) = [\delta\hat{\underline{q}}^T_{i+1}(-),\underline{0}^T,\underline{0}^T] \tag{4.6}$$

and change (2.23d) to read

$$\hat{\underline{x}}_{i+1}(+) = \hat{\underline{x}}_{i+1}(-) + K_{i+1}[\underline{y}_{i+1} - H_{i+1}\hat{\underline{x}}_{i+1}(-)] \tag{4.7}$$

The BF normalization algorithm has all the expected advantages mentioned in Section III; however, it is not elegant in the sense that the normalization constitutes an outside interference in the EKF algorithm which has to be compensated. This compensation adds a certain complication to the algorithm presented in Section II. Therefore we propose the following QPM normalization scheme.

V. QUATERNION PSEUDO-MEASUREMENT (QPM) NORMALIZATION

According to this algorithm the updated quaternion $\hat{\underline{q}}_{i+1}(+)$ is used to form a pseudo-measurement as follows

$$\underline{y}_{n,i+1} = \hat{\underline{q}}_{i+1}(+)/\,||\,\hat{\underline{q}}_{i+1}(+)\,|| \tag{5.1}$$

It is then assumed that the quaternion is measured by an imaginary device, say "quaternion-meter", and the output of this device is $\underline{y}_{n,i+1}$ plus a small white measurement error. Following this rationale a measurement update is performed which is based on the quaternion measurement. To accomplish that we realize from (3.1) that $\underline{y}_{n,i+1}$ is related to the state vector as follows

$$\underline{y}_{n,i+1} = H_{n,i+1}\underline{X}_{i+1} + n_{n,i+1} \tag{5.2}$$

where

$$\underline{X}^T_{i+1} = [\underline{q}^T|\underline{e}^T|\underline{p}^T]_{i+1} \tag{5.3}$$

$$H_{n,i+1} = \left[\begin{array}{cccc|c|c} 1 & 0 & 0 & 0 & & \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & & \\ 0 & 0 & 0 & 1 & & \end{array}\right] \tag{5.4}$$

and $R_{n,i+1}$, the covariance of $\underline{n}_{n,i+1}$, is the diagonal matrix

$$R_{n,i+1} = \text{diag}[r^2, r^2, r^2, r^2] \tag{5.5}$$

and where **r** is a small number. The a-priori state estimate for this measurement update is, of course, $\hat{\underline{X}}_{i+1}(+)$. Note that the output of this update is the full state vector and not just the estimate of, $\underline{x}$, the difference between $\underline{X}_{i+1}$ and its estimate $\hat{X}_{i+1}(+)$. This pseudo-measurement update is performed after the computation in (2.23f) has been carried out. The pseudo-measurement update algorithm is as follows. $K_{n,i+1}$ is computed according to (2.23c) where H_{i+1} and $P_{i+1}(-)$ and R_{i+1} are replaced by $H_{n,i+1}$, $P_{i+1}(+)$ and $R_{n,i+1}$ respectively. The state update is then re-computed as follows

$$\hat{\underline{X}}_{i+1}(+) = \hat{\underline{X}}_{i+1}(+) + K_{n,i+1}[\underline{y}_{n,i+1} - H_{n,i+1}\hat{\underline{X}}_{i+1}(+)] \tag{5.6}$$

and $P_{i+1}(+)$ is re-computed according to (2.23f) where K_{i+1}, H_{i+1} and $P_{i+1}(-)$ are replaced by $K_{n,i+1}$, $H_{n,i+1}$ and $P_{i+1}(+)$ respectively. The new estimate and its covariance are then propagated in time as before.

The QPM normalization performs quite well and achieves the expected benefits of quaternion normalization provided **r** is well tuned. If this is not the case, the attitude estimate may reach a wrong value, and if the attitude changes between vector measurements, it may even diverge. The reason for this is described next.

For the normalization to be effective one is tempted to choose a small **r** in which case the filter practically replaces the stored quaternion estimate by the normalized quaternion. However, the small "measurement noise", **r**, reduces the variance of the quaternion estimation error considerably. Therefore, the filter assigns a very high credibility to the normalized quaternion estimate even though it is not yet the correct quaternion. Consequently, the filter does not allow new vector measurements to alter the quaternion estimate and the latter is stuck on a wrong value. If the quaternion changes now due to attitude change then the quaternion estimate diverges. In order to avoid this phenomenon one has to tune the value of **r** which constitutes an additional design burden. Therefore although, unlike the BF normalization, the QPM normalization blends naturally into the EKF algorithm, the required tuning constitutes a considerable disadvantage. To alleviate this problem we proposed the following magnitude pseudo-measurement normalization scheme.

VI. MAGNITUDE PSEUDO-MEASUREMENT (MPM) NORMALIZATION

Unlike the previous scheme, where we assumed that we "measured" the normalized quaternion, here we assume that we "measure" the square of the quaternion Euclidean norm whose magnitude is assumed to be 1. This imaginary "norm meter" yields the reading z where

$$z_{n,i+1} = 1 + v_{n,i+1} \tag{6.1}$$

and where $\mathbf{v}_{n,i+1}$ is assumed to be a white measurement noise whose variance is **r**. Note that the "measured" quantity is a non-linear function of the quaternion components; therefore, we compute the *effective* measurement, $y_{n,i+1}$, as

$$y_{n,i+1} = z_{n,i+1} - \left(\hat{q}(+)^2_{1,i+1} + \hat{q}(+)^2_{2,i+1} + \hat{q}(+)^2_{3,i+1} + \hat{q}(+)^2_{4,i+1}\right) \tag{6.2}$$

Using (6.1) and (2.4), (6.2) can be written as

$$y_{n,i+1} = 1 - \sum_{j=1}^{4} \left(q_{j,i+1} - \delta q_{j,i+1}\right)^2 + v_{n,i+1} \tag{6.3}$$

Neglecting products of $\delta q_{j,n+1}$, (6.3) can be written as

$$y_{n,i+1} = 1 - \left[\sum_{j=1}^{4} q^2_{j,i+1} - 2\sum_{j=1}^{4} q_{j,i+1}\delta q_{j,i+1}\right] + v_{n,i+1} \tag{6.4}$$

and since

$$\sum_{j=1}^{4} q_{j,i+1}^2 = 1 \tag{6.5}$$

then (6.4) can be written as

$$y_{n,i+1} = \left[2q_{1,i+1} \middle| 2q_{2,i+1} \middle| 2q_{3,i+1} \middle| 2q_{4,i+1}\right] \begin{bmatrix} \delta q_{1,i+1} \\ \delta q_{2,i+1} \\ \delta q_{3,i+1} \\ \delta q_{4,i+1} \end{bmatrix} + v_{n,i+1} \tag{6.6}$$

Since $q_{j,i+1}$ j=1,2,3,4 is unknown, we follow the common practice of replacing the quaternion components by their estimate, thus

$$y_{n,i+1} = \left[2\hat{q}(+)_{1,i+1} \middle| 2\hat{q}(+)_{2,i+1} \middle| 2\hat{q}(+)_{3,i+1} \middle| 2\hat{q}(+)_{4,i+1}\right] \begin{bmatrix} \delta q_{1,i+1} \\ \delta q_{2,i+1} \\ \delta q_{3,i+1} \\ \delta q_{4,i+1} \end{bmatrix} + v_{n,i+1} \tag{6.7}$$

The latter is the measurement equation which is used to perform a magnitude pseudo-measurement normalization update. The sequence of operations is similar to that performed when the QPM normalization update is carried out (see the preceding section). The only difference is that now

$$y_{n,i+1} = 1 - \sum_{j=1}^{4} \hat{q}(+)_{j,i+1}^2 \tag{6.8}$$

$$H = \left[2\hat{q}(+)_{1,i+1} \middle| 2\hat{q}(+)_{2,i+1} \middle| 2\hat{q}(+)_{3,i+1} \middle| 2\hat{q}(+)_{4,i+1}\right] \tag{6.9}$$

and

$$R_{n,i+1} = \mathbf{r} \tag{6.10}$$

We realize that the fact that $\mathbf{r}$ is very small does not imply that the measurement of $\underline{q}$ is precise. It only implies that the measurement of $\| \underline{q} \|$ is precise. Therefore, now the variances of the quaternion states do not increase to a value close to **r** and thus the estimates of the quaternion components do not cling to wrong values and stay there like they do when the preceding QPM normalization is applied with a small **r**.

VII. TEST RESULTS

The algorithms presented in this paper were and still are being tested now. In these tests the EKF is applied to simulated as well as to real Earth Radiation Budget Satellite (ERBS) data. Partial results are presented as follows.

Quaternion normalization speeds up the convergence of the additive EKF when used to estimate spacecraft attitude from vector measurements. Moreover, if the attitude changes considerably between vector measurements, quaternion normalization replaces filter tuning which is necessary to avoid divergence. In the latter case, quaternion normalization also reduces the final attitude estimation error.

In Table 7.1 we see the final attitude estimation error when the EKF is applied to simulated ERBS data. The initial attitude error is 30° and the value of **r** used in the QPM and MPM algorithms is 10^{-5}.

Table 7.1: Final Attitude Error in Degrees at 100 sec, $\mathbf{r}=10^{-5}$

	Normalization Algorithm			
	Without Normalization	BP	QPM	MPM
Yaw	.0048	.0074	.0057	.0069
Roll	.0022	-.0002	.0019	.0039
Pitch	.0170	.0060	-.0009	-.0033
RMS	0.0178	0.0095	0.0061	0.0086

Note that the BF algorithm implies no measurement update therefore no **r** is used in this run. We turn to Table 7.2 to see the advantage of the MPM over the QPM algorithm. We realize that while for $r=10^{-5}$ both algorithms exhibit identical accuracy, the QPM algorithm fails when $r=10^{-11}$. The reason for this difference was mentioned at the end of Section V.

Table 7.2: Final Attitude Error in Degrees at 100 sec, $r=10^{-11}$

	Normalization Algorithm	
	QPM	MPM
Yaw	3.2387	.0045
Roll	10.3660	.0083
Pitch	-0.7451	.0127

VIII. SUGGESTED FUTURE RESEARCH

Although the MPM normalization performed satisfactorily we suggest to investigate an algorithm of implied normalization which does not really use normalization. This algorithm is presented next.

In Section II we presented the development of the additive EKF for quaternion estimation. In that development we derived the linearized relationship between the vector measurement error and the quaternion estimation error which are summarized in (2.16). To meet this end we differentiated the matrix $A(\underline{q})$ given in (2.9). The differentials were partial differentials with respect to the elements of $\underline{q}$. As a result of the differentiations we obtained the matrices G_j, $j=1,2,3,4$ which are listed in (2.10).

When $\underline{q}$ is indeed of unit length, $A(\underline{q})$ is an orthonormal matrix; that is, its columns (rows) are orthogonal to one another and are of unit length. If, however, $\underline{q}$ is not of unit length, then the columns (rows) of $A(\underline{q})$ are still orthogonal to one another, but their length is not a unit anymore. It was proven in [10] that the matrix $A^*(\hat{\underline{q}})$ computed as follows

$$A^*(\hat{\underline{q}}) = \frac{1}{\|\hat{\underline{q}}\|^2} A(\hat{\underline{q}}) \tag{8.1}$$

is not only orthonormal, but it is also the "closest" *orthonormal* matrix to $A(\underline{q})$; that is, of all possible orthonormal matrices, the distance between $A^*(\hat{\underline{q}})$ and $A(\hat{\underline{q}})$ is the smallest where by distance we mean the Euclidean norm of the difference matrix $A^*(\underline{q})-A(\underline{q})$. It can be argued that if we use $A^*(\hat{\underline{q}})$ rather than $A(\hat{\underline{q}})$, we practically enforce normalization. This is so because normalizing $\hat{\underline{q}}$ first and then using the normalized quaternion to compute $A(\hat{\underline{q}})$ is identical to the computation of $A^*(\hat{\underline{q}})$ as given in (8.1). The partial differentiation of (8.1) with respect to the quaternion components yields

$$\Gamma_j = \Gamma_j(\hat{\underline{q}}) = \left.\frac{\partial A^*(\underline{q})}{\partial q_j}\right|_{\hat{q}} = -\frac{2\hat{q}_j}{\|\hat{q}\|^4} A(\hat{\underline{q}}) + \frac{1}{\|\hat{q}\|^2} G_j(\hat{\underline{q}}) \tag{8.2}$$

where $G_j(\hat{\underline{q}})$ is given in (2.10). The final algorithm is as given in Section II with Γ_j replacing G_j in (2.11). We call this normalization scheme the linearized orthogonalized matrix (LOM) algorithm.

Finally, in the future we intend to apply all the normalization schemes discussed here to real ERBS data.

IX. CONCLUSIONS

It was found again that quaternion normalization in the additive EKF for attitude determination from vector measurement has the following advantages. If the attitude changes slowly, normalization speeds up estimation convergence. If attitude changes rapidly between measurements and no normalization is applied then filter tuning has to be used in order to avoid divergence. However, if normalization is applied, convergence is achieved without filter tuning. Moreover, the final attitude estimation error is smaller. There is then a clear advantage to quaternion normalization. Three quaternion normalization algorithms were tested. The conclusions with regard to the use of each one of them is listed next.

- The brute force (BF) normalization algorithm works well and exhibits the normalization benefits described before.
- The quaternion pseudo-measurement (QPM) algorithm performs well only after tuning.
- The magnitude pseudo-measurement (MPM) algorithm performs well and needs no tuning.

Finally, we suggest the investigation of the linearized orthogonal matrix (LOM) normalization whose development was presented in Section VII. All the normalization schemes will be tested on real ERBS data.

References

1. Wertz, J.R., *Spacecraft Attitude Determination and Control*, D. Reidel Publishing Company, Dordrecht, Holland, 1988.

2. Shuster, M.D., "Approximate Algorithm for Fast Optimal Attitude Computation," *AIAA Guidance and Control Conference*, Palo Alto, CA, Aug. 7-9, 1978, Paper 78-1249.

3. Shuster, M.D. and Oh, S.D., "Three Axis Attitude Determination from Vector Observation," *Journal of Guidance and Control*, Vol. 4, Jan.-Feb., 1981, pp. 70-77.

4. Murrell, J. W., "Precision Attitude Determination for Multimission Spacecraft," *AIAA Guidance and Control Conference*, Palo Alto, CA, Aug. 7-9, 1978, Paper 78-1248.

5. Lefferts, E.J., Markley, F.L., and Shuster, M.D., "Kalman Filtering for Spacecraft Attitude Estimation," *Journal of Guidance Control, and Dynamics*, Vol. 5, Sept-Oct. 1982, pp. 417-429.

6. Gai, E.G., Daly, K.C., Harrison, J.V., and Lamos, L., "Star-Sensor-Based Satellite Attitude/Attitude Rate Estimator," Journal of *Guidance, Control, and Dynamics*, Vol. 8, Sept.-Oct. 1985, pp. 560-565.

7. Bar-Itzhack, I.Y., and Oshman, Y., "Attitude Determination from Vector Observations: Quaternion Estimation," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-21, Jan., 1985, pp. 128-135.

8. Bar-Itzhack, I.Y. and Deutschmann, J., "Extended Kalman Filter for Attitude Estimation of the Earth Radiation Budget Satellite," *Astrodynamics Conference*, Portland, OR, Aug. 20-22, 1990, paper 90-2964.

9. Gelb, A., *Applied Optimal Estimation*, M.I.T. Press, Cambridge, MA, 1988, p. 75.

10. Giardina, C.R., Bronson, R., and Wallen, I., "An Optimal Normalization Scheme," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-11, July 1975, pp. 443-446.