# On Orientation Estimation using Iterative Methods in Euclidean Space

Martin A. Skoglund*

*Division of Automatic Control
Department of Electrical Engineering
Linköping University
Linköping, Sweden
Email: ms@isy.liu.se

Zoran Sjanic*,**

**Saab AB
Linköping
Sweden
Email: zoran.sjanic@saabgroup.com

Manon Kok†

†Department of Engineering
University of Cambridge
Cambridge, United Kingdom
Email: mk930@cam.ac.uk

*Abstract*—This paper presents three iterative methods for orientation estimation. The first two are based on iterated Extended Kalman filter (IEKF) formulations with different state representations. The first is using the well-known unit quaternion as state (q-IEKF) while the other is using orientation deviation which we call IMEKF. The third method is based on nonlinear least squares (NLS) estimation of the angular velocity which is used to parametrise the orientation. The results are obtained using Monte Carlo simulations and the comparison is done with the non-iterative EKF and multiplicative EKF (MEKF) as baseline. The result clearly shows that the IMEKF and the NLS-based method are superior to q-IEKF and all three outperform the non-iterative methods.

## I. INTRODUCTION

Orientation estimation is of concern for a wide range of applications. Examples include estimating the movement of a camera for visual-inertial simultaneous localisation and mapping (SLAM) [1–4] and estimation of human motion [5–7]. In this paper, we consider orientation estimation in three dimensions, i.e., we consider orientations that are described by the special orthogonal group SO(3). More specifically, we focus on filtering, where a state $x_t$ representing the orientation is estimated using the measurements $y_1, \ldots, y_t$.

A commonly used estimation method is the Extended Kalman filter (EKF), see e.g., [8]. EKFs are computationally quite efficient, but because they are based on first-order Taylor expansions of nonlinear models, they can suffer from linearisation errors. An issue with using EKFs for orientation estimation is that they assume that the unknown states $x_t$ reside in Euclidean space, $\mathbb{R}^n$. This is particularly the case for the measurement updates where additivity between states and the state correction is assumed. Since orientations reside in SO(3) instead, additional approximations are needed to estimate orientation using EKFs. If the estimation errors are small, these approximations are usually good enough and the estimated rotation will be good. However, if the correction term is large, the approximations applied on the corrected state can lead to large state estimation errors, since the correction term will take the state far away from the manifold.

In this paper we will consider two well-known strategies for using EKFs for orientation estimation. The first uses unit quaternions as a state [9, 10]. After each measurement
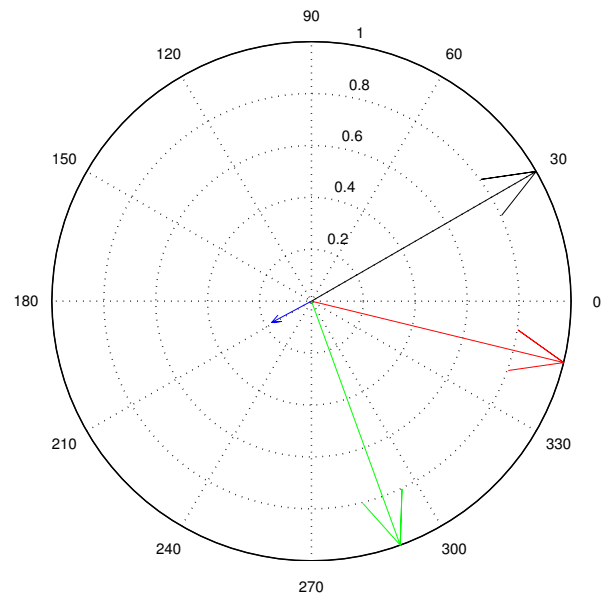


Fig. 1: Illustration of the measurement update problem with EKF and large residuals. True rotation is in black, prior or predicted rotation (i.e., before measurement update) is in green, rotation after measurement update with EKF is in blue and rotation after measurement update with MEKF is in red.

update, the quaternion will be unnormalised and an additional renormalisation step needs to be included. We refer to this approach as the q-EKF. An alternative approach is the so-called MEKF, where a three-dimensional orientation deviation is used as the state in the EKF and a linearisation point, parametrised for instance as a quaternion, is updated each time the state becomes non-zero. Since the orientation deviation is assumed to be small, the state can be seen as Euclidean. This approach is called a multiplicative EKF (MEKF) and is frequently used in aeronautics, see e.g., [11, 12].

For both the q-EKF and the MEKF, the measurement model is nonlinear and the measurement update of the EKF does not necessarily result in good estimates. This is specifically of

concern in case of large residuals, i.e., when large corrections need to be made in a single recursion of the algorithm. This is illustrated in Figure 1 for a one-dimensional example. Here, the true angle is depicted in black and the prior or equivalently the prediction is depicted in green. The angle is parametrised in terms of a two-dimensional vector with unit norm (the two-dimensional equivalent of a unit quaternion). The prior can be seen to deviate significantly from the true angle. This fact is reflected in a large prior covariance. Now assume that accurate measurements of the true angle are obtained. We would hope for the EKF measurement to produce an angle close to the black line. The angle obtained from the measurement update of the q-EKF, is depicted in blue while the angle obtained from the MEKF is depicted in red. Both can be seen to deviate significantly from the true angle. The angle estimated by the q-EKF is actually further away from the true angle than the prior. Furthermore, the magnitude of the resulting quaternion is less than 0.2.

There are different attempts to remedy this kind of undesired behaviour. Some of them still use rotation states in the filtering but explicitly take into account that these states reside in SO(3). Examples include the Lie Algebra-EKF [13] and approaches making use of distributions on manifolds [14, 15]. In this paper, however, we consider a quite natural modification of the above-mentioned EKF implementations. The idea is to iteratively improve the state estimate during measurement update by re-linearising the Jacobians in the current state estimate. This leads to the Iterated EKF with unit quaternions as states (q-IEKF) and the Iterated MEKF (IMEKF). These approaches are comparable to the Gauss-Newton method for finding the minimum of a function [16]. For an introduction to iterated Extended Kalman filtering see e.g., [8]. We also present a novel formulation of the orientation estimation problem in terms of a nonlinear least squares problem which we refer to as ROT-OPT. Instead of estimating the orientation directly, the angular velocity – which resides in Euclidean space – is considered to be the optimisation variable in ROT-OPT.

To allow for systematic comparison between the different approaches, we perform a number of simulation studies and show that the iterative methods outperform the EKF approaches. The down-side of the q-IEKF is that the estimation is still performed in the rotation space, i.e., on the manifold, and approximations are therefore still present during the iterations. This is shown to negatively influence the estimation accuracy. The state of the IMEKF, however, almost resides in Euclidean space (except the $2\pi$-wrap issue), making the iterations more meaningful and resulting in a much higher estimation accuracy. The IMEKF and ROT-OPT perform similarly in our simulation studies. These results show that iterative estimation methods in (almost) Euclidean space clearly outperform the other methods.

## II. MODELS FOR ORIENTATION ESTIMATION

In this section, models used for the orientation estimation are introduced. Different ways of parametrising rotations are described as well as dynamical and measurement models used in filtering.

### A. Parametrisation of Orientation

The following sections consider coordinate frames to be right-handed and as such, positive rotations are counter-clockwise for vectors pointing to an observer. The rotation space in 3 dimensions is called Special Orthogonal in dimension 3 or SO(3) and is spanned by all proper $3 \times 3$ rotation matrices with the following properties

$$\mathsf{R} \in \mathbb{R}^{3\times 3}, \qquad \mathsf{R}^{-1} = \mathsf{R}^{T}, \qquad \det(\mathsf{R}) = 1. \tag{1}$$

In practical applications, these rotation matrices express the rotation of different 3-dimensional vectors relative to each other. Using the full rotation matrix parametrisation to represent SO(3) is often impractical when estimating orientation, since the (non-convex) constraints are difficult to handle. Common alternatives are Euler angles, unit quaternions or the axis-angle representation, which each have different properties that make them more or less suitable for the estimation applications.

The axis-angle representation encodes the rotation as a vector $\eta \in \mathbb{R}^3$. Its direction, denoted $\bar{\eta}$, is the axis of rotation, and its length, denoted $\varphi$, is the rotation angle around that axis. The axis-angle parametrisation can be obtained from the rotation matrix as

$$\varphi \triangleq \|\eta\| = \arccos\left(\frac{\operatorname{trace}(\mathsf{R}) - 1}{2}\right), \tag{2a}$$

$$\bar{\eta} \triangleq \frac{\eta}{\varphi} = \frac{1}{2\sin(\varphi)} \begin{bmatrix} \mathsf{R}(2,3) - \mathsf{R}(3,2) \\ \mathsf{R}(3,1) - \mathsf{R}(1,3) \\ \mathsf{R}(1,2) - \mathsf{R}(2,1) \end{bmatrix}, \tag{2b}$$

where special care must be taken for the case when $\varphi = 0$, i.e., no rotation ($\mathsf{R} = I$), for which $\eta = 0$. The rotation matrix is obtained from axis-angle via Rodrigues' formula [17]

$$\mathsf{R} = I + \sin(\varphi)\widehat{\bar{\eta}} + (1 - \cos(\varphi))\left(\widehat{\bar{\eta}}\right)^2, \tag{3}$$

where we have used the widehat operator $\widehat{\cdot}$ to transform a vector in $\mathbb{R}^3$ to a skew-symmetric matrix in $\mathbb{R}^{3\times 3}$

$$\widehat{\eta} = \begin{bmatrix} 0 & \eta_3 & -\eta_2 \\ -\eta_3 & 0 & \eta_1 \\ \eta_2 & -\eta_1 & 0 \end{bmatrix}. \tag{4}$$

This sign-convention describes a rotation of the object w.r.t. an inertial frame as opposed to describing the rotation of the inertial frame w.r.t. the object. Also note that for small $\varphi$, (3) can be approximated as

$$R \approx I + \widehat{\eta}. \tag{5}$$

It is worth noting that the axis-angle representation is ambiguous, since $2\pi k\eta$ for integer $k$ represents the same rotation as $\eta$ [18].

Another orientation parametrisation is the unit quaternion [19], denoted $q = [q_0 \ q_1 \ q_2 \ q_3]^T \in \mathbb{S}^3 \subset \{\mathbb{R}^4 | q^T q = 1\}$.

The corresponding rotation matrix can be computed from a unit quaternion as

$$\mathsf{R} = \tag{6}$$
$$\begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 + q_0q_1) \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 - q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}.$$

From (6) we also note that $q$ and $-q$ represent the same rotation, i.e., the unit quaternion has antipodal properties. The relation between the unit quaternion and the axis-angle representation is defined as

$$q = \begin{bmatrix} \cos\left(\frac{\varphi}{2}\right) \\ \bar{\eta}\sin\left(\frac{\varphi}{2}\right) \end{bmatrix}, \tag{7}$$

The reverse transformation can be obtained by inverting the equation above (with the same treatment as in (2) for the case of no rotation)

$$\varphi = 2\arccos(q_0), \qquad \bar{\eta} = \frac{1}{\sin\left(\frac{\varphi}{2}\right)}\begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix}. \tag{8}$$

The transformation from the unit quaternion to the rotation matrix can be done via the axis-angle representation i.e., by using (8) and (3).

We will not describe the Euler angle representation in more detail here but only mention that it requires three parameters (called yaw, pitch and roll) and that it suffers from $2\pi k$ / $\pi k$ ambiguity as well as a phenomenon called gimbal lock, which the other representation do not have. For more details, see for example [18, 20].

### B. Rotation Dynamics

As mentioned, we would like to estimate the rotation sequence that is changing in time, $\alpha_t$, $t = 0,\dots,N$, where $\alpha$ represents any rotation parametrisation described above. Evolution of the rotation in time can be represented by a differential equation of the form

$$\dot{\alpha}_t = f(\alpha_t, \omega_t), \tag{9}$$

where $\omega_t = [\omega_{X,t}\ \omega_{Y,t}\ \omega_{Z,t}]^T \in \mathbb{R}^3$ represents the rotational velocity around object's $X$-, $Y$- and $Z$-axis respectively. The function $f$ has different forms depending on which parametrisation of the rotations we use. Some parametrisations are more attractive to use since they have a linear representation of the dynamics. For example, rotation matrix dynamics can be represented as

$$\dot{\mathsf{R}}_t = \widehat{\omega}_t\,\mathsf{R}_t, \tag{10}$$

where $\widehat{\omega}_t$ is the skew-symmetric matrix defined in (4), see e.g., [20] for a complete derivation.

The unit quaternion representation has a similar form

$$\dot{q}_t = S(\omega_t)q_t = \widetilde{S}(q_t)\omega_t, \tag{11}$$

where $S(\omega_t)$ is also a skew-symmetric matrix defined as

$$S(\omega) = \frac{1}{2}\begin{bmatrix} 0 & -\omega_X & -\omega_Y & -\omega_Z \\ \omega_X & 0 & \omega_Z & -\omega_Y \\ \omega_Y & -\omega_Z & 0 & \omega_X \\ \omega_Z & \omega_Y & -\omega_X & 0 \end{bmatrix}, \tag{12}$$

while $\widetilde{S}(q)$ is

$$\widetilde{S}(q) = \frac{1}{2}\begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & q_1 & q_0 \end{bmatrix}. \tag{13}$$

The price to pay for the linear dynamics is the number of parameters and the constraints. The dynamics of the axis-angle parametrisation (which is minimal with three parameters) is more complicated and is given by [20]

$$\dot{\varphi}_t = \bar{\eta}_t^T\omega_t, \qquad \dot{\bar{\eta}}_t = \frac{1}{2}\left(\widehat{\bar{\eta}}_t - \cot\left(\frac{\varphi_t}{2}\right)\widehat{\bar{\eta}}_t^2\right)\omega_t. \tag{14}$$

The descriptions above are expressed in continuous time, but for a filter estimation application a discrete time description is much more convenient. If we assume that angular velocities, $\omega$, are constant between sampling intervals (which is a good approximation for most gyroscopes because of their high sampling frequency), the discrete time representation is given by exact sampling [21] for the linear representations as

$$\mathsf{R}_{t+T} = e^{T\widehat{\omega}_t}\,\mathsf{R}_t, \tag{15a}$$
$$q_{t+T} = e^{TS(\omega_t)}q_t, \tag{15b}$$

where $T$ is the sampling interval. For the nonlinear dynamics (14), there is no exact expression and approximate methods, like Euler approximations, must be used.

### C. Measurements

Except gyroscopes measuring rotational velocities, other sensors are usually present measuring other entities that can be related to the orientation. Typical sensors that can be used for such purposes are accelerometers and magnetometers. Accelerometers measure the specific force that is acting on them, which includes the earth's gravitation, and magnetometers measure the local magnetic field. These two measurements can be related to the rotation of the object as

$$y_t^a = \mathsf{R}_t\,g + e_t^a, \tag{16a}$$
$$y_t^m = \mathsf{R}_t\,m + e_t^m, \tag{16b}$$

where we have assumed that the sensors are strapped to the rotating object and rotating together with it, as well as that no other forces than gravity are acting on the rotating object. The vectors $g$ and $m$ are typically expressed in the local fixed frame, called the earth frame, and assumed to be known, constant and non-parallel. We have also included the sensor noise terms $e_t^a$ and $e_t^m$ which are white, i.i.d. and $e_t^a \sim \mathcal{N}(0, \Sigma^a)$ and $e_t^m \sim \mathcal{N}(0, \Sigma^m)$.

Other measurements can be used depending on the sensors that are present, but the important property is of course that the measurements and rotations must be related in some way. Note

also that measurements can come in at different frequencies, which can potentially be different from angular rates' frequency. That fact is not a problem for the algorithms presented later, since all of them work in a multi-rate measurement update fashion.

## III. Recursive Estimation of Orientation

In this section, two iterative filters based on the Iterated Extended Kalman filter (IEKF) [8] and an optimisation based method are discussed. As noted in, e.g., [22, 23], the EKF is a special case of Gauss-Newton (GN) optimisation with only one iteration. The performance is therefore likely to improve using the IEKF when the linearisation errors are significant. The IEKF can be derived as the maximum a posteriori (MAP) estimate resulting in a nonlinear least-squares problem which is efficiently solved using GN. The corresponding MAP cost function can furthermore be used to monitor the performance and to adjust step-size while the optimisation view-point can be used to device other Quasi-Newton methods [16]. The generic iterated measurement update for the IEKF is

$$x_{i+1} = \hat{x}_{t|t-1} + K_i\big(y_t - h(x_i) - H_i(\hat{x}_{t|t-1} - x_i)\big) \quad (17)$$

where components depending on $i$ are recalculated at each iteration. Here the measurement equation $h(x)$ is for instance (16), $x$ is the state of interest, in this case some rotation parametrisation, and $H$ is the Jacobian of $h(x)$ w.r.t. $x$. When the iterations have terminated, the state and covariance are updated as

$$\hat{x}_{t|t} = x_{i+1}, \quad (18a)$$
$$P_{t|t} = (I - K_i H_i)P_{t|t-1}. \quad (18b)$$

where $K_i$ is the last iterated Kalman gain matrix.

### A. Filtering using the unit quaternion as state

The parametrisation alternatives discussed all have their strengths and weaknesses. For example, the individual coordinates of the unit quaternion are Euclidean for which it is simple to apply nonlinear filtering schemes but additive corrections typically result in violation of the unit constraint. The idea is to compute an additive measurement update correction using maximum a posteriori and then project the estimate onto the unit-sphere by normalisation

$$q := \frac{q}{\|q\|}. \quad (22)$$

This approximation can yield severe error if the correction is large. The quaternion covariance is propagated using the Jacobian of (22) as

$$J = \left.\frac{\partial \frac{q}{\|q\|}}{\partial q}\right|_{q=\hat{q}} = \frac{1}{\|\hat{q}\|_2^3}\hat{q}\hat{q}^T, \quad (23a)$$
$$P := JPJ^T, \quad (23b)$$

as in [24]. There are other suggestions for covariance propagation in [25] but these are not investigated here. Furthermore, when four-dimensional unit quaternions are used as states, in

---

**Algorithm 1: Iterated Extended Kalman Filter with quaternion state (q-IEKF)**

INPUT: Inertial and magnetometer data $\{y_t^a, \omega_t, y_t^m\}_{t=1}^N$, with the corresponding covariance matrices $\Sigma^a$, $\Sigma^\omega$, $\Sigma^m$. At time $t = 0$ require an initial state, $\hat{q}_{t|t-1}$ and its covariance, $P_{t|t-1}$.

OUTPUT: An orientation estimate $\hat{q}_{t|t}$ and covariance $P_{t|t}$ for $t = 1 \dots N$.

1. Measurement update iterations
Let $i = 1$, $q_i = \hat{q}_{t|t-1}$ and $\Sigma_t = \text{diag}(\Sigma^a, \Sigma^m)$
**while** termination criterion not fulfilled **do**

$$h_i = [\mathsf{R}_i\, g, \ \mathsf{R}_i\, m]^T \quad \text{Using (6) and (16)} \quad (19a)$$
$$H_i = \left.\frac{\partial h_i(q)}{\partial q}\right|_{q=q_i} \quad (19b)$$
$$K_i = P_{t|t-1}H_i^T(H_i P_{t|t-1}H_i^T + \Sigma_t)^{-1} \quad (19c)$$
$$q_{i+1} = \hat{q}_{t|t-1} + K_i\big(y_t - h_i - H_i(\hat{q}_{t|t-1} - q_i)\big) \quad (19d)$$
$$J = \frac{1}{\|q_{i+1}\|_2^3}q_{i+1}q_{i+1}^T \quad \text{Using (23a)} \quad (19e)$$
$$q_{i+1} := \frac{q_{i+1}}{\|q_{i+1}\|} \quad (19f)$$
$$i := i + 1 \quad (19g)$$

**end while**
2. Update the state and the covariance

$$\hat{q}_{t|t} = q_i \quad (20a)$$
$$P_{t|t} = (I - K_{i-1}H_{i-1})P_{t|t-1} \quad (20b)$$
$$P_{t|t} := JP_{t|t}J^T \quad \text{Using (23b)} \quad (20c)$$

3 Time update

$$\hat{q}_{t+1|t} = e^{TS(\omega_t)}\hat{q}_{t|t}, \quad \text{Using (15b)} \quad (21a)$$
$$F_t = e^{TS(\omega_t)} \quad (21b)$$
$$G_t = T\widetilde{S}(\hat{q}_{t|t}) \quad (21c)$$
$$P_{t+1|t} = F_t P_{t|t}F_t^T + G_t\Sigma^\omega G_t^T \quad (21d)$$

---

state space models, filter tuning becomes somewhat difficult since the dynamics (11) only has 3-DoF as can be seen in (12). This means that the unit quaternion only has 3-DoF, thus its true covariance is rank 3. A practical method is to introduce process noise on all four components. Instead of having additive noise directly on the quaternion it is perhaps more intuitive to introduce noise $w_t \sim \mathcal{N}(0, \Sigma^\omega) \in \mathbb{R}^3$ on the rotational velocity which gives the perturbed dynamics (15b)

$$q_{t+T} = e^{TS(\omega_t + w_t)}q_t. \quad (24)$$

In order to propagate the noise covariance in the time update, the first order Taylor approximation of (24), resulting in the bi-linear expression

$$q_{t+T} \approx (I + TS(\omega_t + w_t))q_t = q_t + T\widetilde{S}(q_t)(\omega_t + w_t), \quad (25)$$

is used to obtain the required Jacobians.

We now have what is necessary to introduce the q-IEKF. In the simplest of q-IEKF implementations the measurement update is computed in an iterative fashion and terminated using some criterion, such as, when the increment is small or a fixed number of iterations are performed. For more details see e.g., [16]. The q-IEKF is summarised Algorithm 1. Note that if only one iteration the performed the q-IEKF becomes a q-EKF as it should.

### B. Filtering using the orientation deviation as state

The multiplicative Extended Kalman Filter (MEKF) [11, 12] uses a global orientation parametrisation such as the rotation matrix or a semi-global one such as the unit quaternion. The corresponding measurement update is, however, performed using an axis-angle state $\eta$ encoding a deviation from the unit quaternion. Another difference is that the covariance of the orientation in MEKF is a full rank $3 \times 3$ matrix while the EKF with quaternion state uses a $4 \times 4$ matrix with rank 3 due to the state dimension. As in [11] we express $\eta$ in the body frame but let our quaternion denote a rotation from the body frame to the navigation frame, expressed in the navigation frame. This implies that the new rotation is obtained by the following expression

$$q_{\text{new}} = q_{\text{old}} \odot q(\eta), \tag{26}$$

where quaternion parametrisation using $\eta$ is established using (7). The quaternion multiplication $\odot$ using two quaternions $p$ and $q$ is defined as

$$p \odot q \triangleq \begin{bmatrix} p_0 q_0 - p_v q_v^T \\ p_0 q_v + q_0 p_v + p_v \times q_v \end{bmatrix}, \tag{27}$$

where subscript 0 denote the first component of the quaternion while $v$ denote the last three components, the so-called vector part. Note that (26) preserves the quaternion unit length. In a numerical implementation it is, however, necessary to renormalise the quaternion regularly due to numerical errors using (22).

In the MEKF, the state $\eta$ is reset after each measurement update since it expresses a local orientation deviation. This deviation is computed from the measurements using the standard Kalman update, (19c) and (19d). After the updated quaternion is calculated, using (26), the covariance matrix needs to be transformed to represent uncertainty in the new orientation. This is done using

$$J = \mathsf{R}(\eta), \quad \text{Using (3)} \tag{28a}$$
$$P := JPJ^T. \tag{28b}$$

Note that this is not performed in q-IEKF since there the iterates are calculated from the initial state $\hat{q}_{t|t-1}$.

The time update of the quaternion is performed exactly as for q-EKF/q-IEKF, while the time update of the covariance is done with respect to the dynamics of the deviation state [11]. We have now derived what is needed for an MEKF algorithm, however a complete and more detailed derivation of the MEKF can be found in [24].

---

**Algorithm 2:** Iterated Multiplicative Extended Kalman Filter with orientation deviation state (IMEKF)

---

INPUT: Inertial and magnetometer data $\{y_t^a, \omega_t, y_t^m\}_{t=1}^N$, with the corresponding covariance matrices $\Sigma^a$, $\Sigma^\omega$, $\Sigma^m$. At time $t = 0$ require an initial orientation estimate, $\hat{q}_{t|t-1}$ and an orientation deviation state covariance, $P_{t|t-1}$.
OUTPUT: An orientation estimate $\hat{q}_{t|t}$ and covariance $P_{t|t}$ for $t = 1 \ldots N$
1. Measurement update iterations
Let $i = 1$, $q_i = \hat{q}_{t|t-1}$, $P_i = P_{t|t-1}$ and $\Sigma_t = \text{diag}(\Sigma^a, \Sigma^m)$
**while** termination criterion not fulfilled **do**

$$h_i = [\mathsf{R}_i\, g, \ \mathsf{R}_i\, m]^T \quad \text{Using (6) and (16)} \tag{29a}$$
$$H_i = \left.\frac{\partial h_i(\eta)}{\partial \eta}\right|_{\eta=0} = \begin{bmatrix} \widehat{\mathsf{R}_i\, g} \\ \widehat{\mathsf{R}_i\, m} \end{bmatrix} \tag{29b}$$
$$K_i = P_i H_i^T (H_i P_i H_i^T + \Sigma_t)^{-1} \tag{29c}$$
$$\eta = K_i(y_t - h_i) \tag{29d}$$
$$\tilde{P} = (I - K_i H_i) P_i \tag{29e}$$

2 Propagate the quaternion and the covariance

$$q_{i+1} := q_i \odot q(\eta) \tag{30a}$$
$$J = \mathsf{R}(\eta) \tag{30b}$$
$$P_{i+1} := J P_i J^T \tag{30c}$$
$$i := i + 1 \tag{30d}$$

**end while**
3 Update the quaternion estimate and the covariance

$$\hat{q}_{t|t} = q_i, \tag{31a}$$
$$P_{t|t} = J \tilde{P} J^T \tag{31b}$$

4 Time update

$$\hat{q}_{t+1|t} = e^{TS(\omega_t)} \hat{q}_{t|t}, \quad \text{Using (15b)} \tag{32a}$$
$$F_t = e^{T\widehat{\omega}_t} \tag{32b}$$
$$G_t = TI \tag{32c}$$
$$P_{t+1|t} = F_t P_{t|t} F_t^T + G_t \Sigma^\omega G_t^T \tag{32d}$$

---

From the MEKF it is straightforward to introduce measurement update iterations using (17) and (18). A slight difference to IEKF is that the covariance matrix is updated during iterations. This is because the IMEKF calculates the new orientation deviation from a current quaternion iterate, while q-IEKF calculates the correction from the prior quaternion, $\hat{q}_{t|t-1}$, in each iteration. The IMEKF is summarised in Algorithm 2. Notice that, exactly as for the q-IEKF case, if only one iteration is done we obtain regular MEKF.

## C. Filtering using the iterative optimisation approach

An alternative improvement would be to actually use proper Euclidean variables in an iterative optimisation scheme, since the steps made in the iterations are always correct. One way of posing this problem is to use the measurement equations in (16) and combine them with dynamics equation (15). Basically, we will replace the rotation term in the measurement equations with the right-hand side of the dynamics obtaining

$$y_t^a = e^{T\widehat{\omega}_{t-1}}\hat{\mathsf{R}}_{t-1}g + e_t^a, \tag{33a}$$

$$y_t^m = e^{T\widehat{\omega}_{t-1}}\hat{\mathsf{R}}_{t-1}m + e_t^m. \tag{33b}$$

Here, we have used the rotation estimate from previous time $\hat{\mathsf{R}}_{t-1}$. Now we can see the angular velocity, $\omega_{t-1}$, as a parameter to be estimated, called $\theta_{t-1}$, and pose the following unconstrained nonlinear least squares (NLS) problem to be solved in each time step

$$\hat{\theta}_{t-1} = \underset{\theta_{t-1}}{\arg\min} \|y_t^a - e^{T\hat{\theta}_{t-1}}\hat{\mathsf{R}}_{t-1}g\|^2_{(\Sigma^a)^{-1}}+$$
$$\|y_t^m - e^{T\hat{\theta}_{t-1}}\hat{\mathsf{R}}_{t-1}m\|^2_{(\Sigma^m)^{-1}}+$$
$$\|\omega_{t-1} - \theta_{t-1}\|^2_{(\Sigma^\omega)^{-1}}, \tag{34}$$

where the last term was added in order to use the angular rate signal as a direct measurement of the parameter as

$$\omega_{t-1} = \theta_{t-1} + e_{t-1}^\omega, \tag{35}$$

where the sensor noise is white and $e_t^\omega \sim \mathcal{N}(0, \Sigma^\omega)$. After the parameter is estimated, the rotation is simply updated with expression in (15) with $\hat{\theta}$ plugged in instead of $\omega$. We will call this optimisation based approach ROT-OPT.

The formulation in (34) is viable if the rotation error is small, since all of the error terms will be of the similar order of magnitude. If the error is large, it will be interpreted as a large angular velocity needed to compensate for the angular error. For that reason we need to account for this kind of bias by adding the bias parameter, here called $\mu^b$, to the angular rates error term, as in

$$\|\omega_{t-1} - \theta_{t-1} - \mu_{t-1}^b\|^2_{(\Sigma^\omega)^{-1}}, \tag{36}$$

and estimating for it as well. Basically, this formulation would be the same as saying that noise of the angular rates is distributed as $e_t^\omega \sim \mathcal{N}(\mu_t^b, \Sigma^\omega)$. Now, this new problem has six variables to solve in each step (both parameters are three-dimensional) and they are both Euclidean which means that optimisation is done in the proper space, $\mathbb{R}^6$ in this case. Note that in this problem formulation, unlike q-EKF/MEKF and their iterative modifications, no state covariance is propagated in time, everything is just seen as parameters. However, if the covariance of the rotation state is wanted, it can be obtained by a Gauss approximation from the covariance of the parameters. The parameter covariance, in turn, can be estimated with the Jacobian of the loss function in (34), $J(\theta)$, evaluated at the solution $\hat{\theta}$, $\mathrm{Cov}(\hat{\theta}) = (J(\hat{\theta})^T J(\hat{\theta}))^{-1}$. This optimisation problem can be solved by an arbitrary NLS solver.

## IV. RESULTS

For the evaluation of the filtering methods presented above, a simulation experiment is performed. In the simulation, one minute of sensor data has been simulated, angular rates with the sampling frequency of 50 Hz and accelerometer and magnetometer measurements, as in (16), with varying sampling frequency, between 25 and 0.05 Hz. It is assumed that the accelerometer and magnetic measurements have low noise levels while the angular rates have about 90 times higher ones. On top of that, a large bias (order of magnitude tenth of degrees) has been added to the angular rates measurements in order to obtain large estimation errors between measurement updates when only time updates are done (which is simply dead-reckoning of the angular rates). This kind of data is quite unrealistic, since most gyroscopes do not have this kind of large bias, but it is done here in order to illustrate the case of large rotation errors.

Monte Carlo simulations are performed, where 100 different noise realisations are simulated for 5 different measurement frequencies and all of the filters have been applied. The results intend to compare mainly the three iterative schemes, while single step updates are there solely for baseline reference. Measurement and process noise covariance values for all of the filters are set to the same value as the simulated noise covariances. Also, for all iterative schemes the termination criterion was simply set to perform 8 iterations. This value was chosen based on the empirical observations of the average number of iterations that ROT-OPT needed to converge, which was between 5 and 10.

The results are depicted in Figures 2 and 3, where the RMSE value as a function of time has been shown only for those times where measurement updates are performed and only for the two measurement intervals, 0.04 s and 10 s.

In the figures it can be seen that q-EKF and MEKF have very similar performance and that q-IEKF performs better sometimes, but not always, like in the case for the yaw angle for 2 s measurement interval, see Figure 3. Both IMEKF and ROT-OPT achieve exactly the same error, which is much smaller then other methods, and is more or less constant in time as the measurement intervals (and hence the measurement residuals) increase. The oscillatory behaviour of the RMSE for some of the estimators in the left column in Figure 2 depends on Signal-to-Noise ratio (SNR) which in turn depends on the rotation itself. Also note that the obtained RMSE is worse for the yaw angle which is the typical behaviour for combination of gravity and magnetic field measurements. In summary the iterative estimators, in combination with an Euclidean parametrisation, obtain smaller estimation errors when the measurement residuals are large.

## V. CONCLUSIONS AND FUTURE WORK

In this paper three filtering methods with iterative measurement updates for estimating orientation are presented and discussed. Two of the approaches (IMEKF and ROT-OPT), which are the main contributions, use Euclidean variables to optimise over, while the third one (q-IEKF) uses quaternion
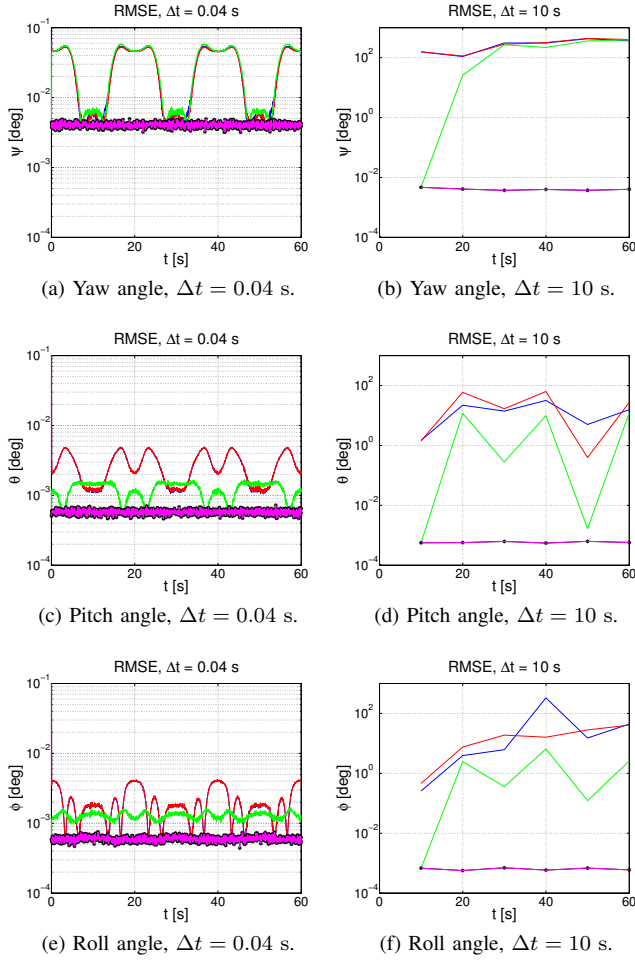
(a) Yaw angle, $\Delta t = 0.04$ s.

(b) Yaw angle, $\Delta t = 10$ s.

(c) Pitch angle, $\Delta t = 0.04$ s.

(d) Pitch angle, $\Delta t = 10$ s.

(e) Roll angle, $\Delta t = 0.04$ s.

(f) Roll angle, $\Delta t = 10$ s.

Fig. 2: RMSE of the rotation angles as a function of time for two time intervals between measurement updates, $0.04$ s ($25$ Hz) and $10$ s ($0.1$ Hz). Blue line is q-EKF, red line is MEKF, green line is q-IEKF, black line with asterisks is IMEKF and magenta line is ROT-OPT. Log-scale on the $y$-axis.

as states. The obtained results show that both IMEKF and ROT-OPT have superior performance to q-IEKF, particularly for the case with large measurement residuals. All three approaches perform better than single iteration versions, q-EKF and MEKF, which is expected, but q-IEKF is actually not so much better. This shows that using iterative measurement updates and Euclidean variables gains great improvement in the orientation estimation.

Since both IMEKF and ROT-OPT achieve basically the same RMSE, it would be interesting to study their similarities and differences. One common thing for both methods is that they minimise a certain loss function (but not necessarily the same one) in order to achieve the solution. ROT-OPT's loss function in (34) is a straightforward formulation of a standard NLS problem that can be solved by any NLS solver. The solution is interpreted as the total angular velocity needed to
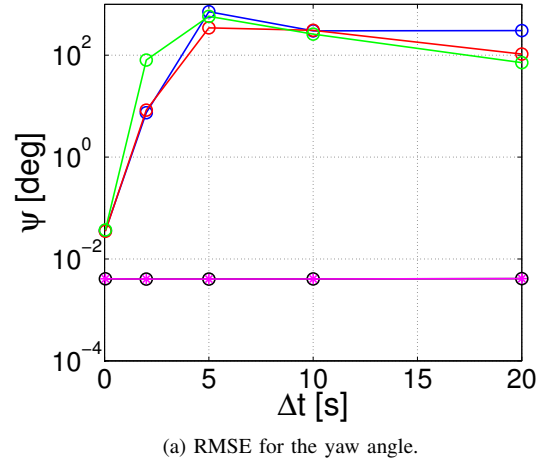


(a) RMSE for the yaw angle.
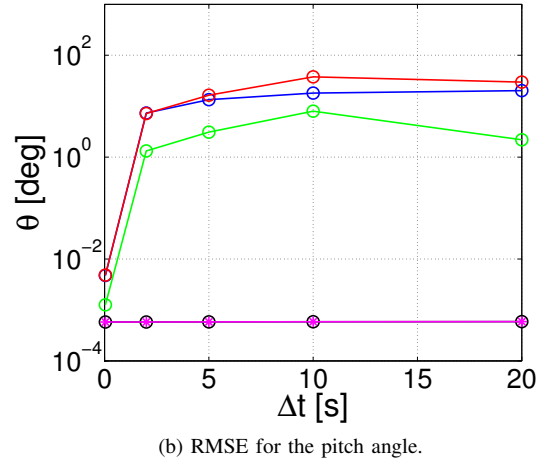
(b) RMSE for the pitch angle.
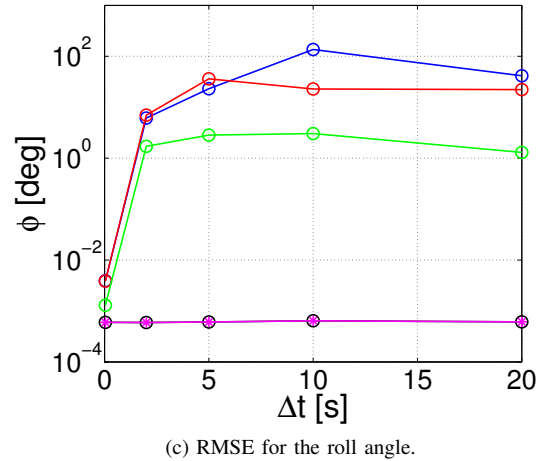
(c) RMSE for the roll angle.

Fig. 3: RMSE for the rotation angles as a function of time interval between measurement updates. Blue line is q-EKF, red line is MEKF, green line is q-IEKF, black line is IMEKF and magenta line with asterisks is ROT-OPT. Log-scale on the $y$-axis.

achieve the orientation best explained by the measurements. IMEKF, on the other hand, updates the current orientation estimate after each iteration as well as propagates its covariance. In the iterated EKF context this can be seen as constantly changing the prior orientation estimate implying that the loss function is actually changed after each iteration. However, these differences are simply of technical nature and the end result is the same estimation error.

As an extension to this work a case where some constraints are present can be studied and IMEKF and ROT-OPT will be compared. The big difference is that the constraints are straight forward to introduce in the ROT-OPT formulation, while it is not obvious how to do that for the IMEKF. Also, an extension to 6-DoF, i.e., adding 3D positions, can be considered as well and different approaches' performance compared.

## ACKNOWLEDGMENTS

## REFERENCES

[1] M. Bryson and M. Johnson-Roberson and S. Sukkarieh, "Airborne smoothing and mapping using vision and inertial sensors," in *Proceedings of the International Conference on Robotics and Automation (ICRA)*.  Kobe, Japan: IEEE Press, 2009, pp. 3143–3148.

[2] T. Lupton and S. Sukkarieh, "Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 61–76, Feb. 2012.

[3] S.-H. Jung and C. J. Taylor, "Camera trajectory estimation using inertial sensor measurements and structure from motion results," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2001, pp. II–732–II–737.

[4] Z. Sjanic, M. A. Skoglund, and F. Gustafsson, "EM-SLAM with inertial/visual applications," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 1, pp. 273–285, Apr. 2017.

[5] M. Kok, J. D. Hol, and T. B. Schön, "An optimization-based approach to human body motion capture using inertial sensors," in *Proceedings of the 19th World Congress of the International Federation of Automatic Control*, Cape Town, South Africa, Aug. 2014, pp. 79–85.

[6] M. Miezal, B. Taetz, and G. Bleser, "On inertial body tracking in the presence of model calibration errors," *Sensors*, vol. 16, no. 7, p. 1132, 2016.

[7] T. Seel, J. Raisch, and T. Schauer, "IMU-based joint angle measurement for gait analysis," *Sensors*, vol. 14, no. 4, pp. 6891–6909, 2014.

[8] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*, ser. Mathematics in Science and Engineering.  Academic Press, Inc, 1970, vol. 64.

[9] A. M. Sabatini, "Quaternion-based extended Kalman filter for determining orientation by inertial and magnetic sensing," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 7, pp. 1346–1356, 2006.

[10] J. L. Marins, X. Yun, E. R. Bachmann, R. B. McGhee, and M. J. Zyda, "An extended Kalman filter for quaternion-based orientation estimation using MARG sensors," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 4, Maui, Hawaii, USA, 2001, pp. 2003–2011.

[11] L. F. Markley, "Attitude error representations for Kalman filtering," *Journal of Guidance, Control, and Dynamics*, vol. 26, no. 2, pp. 311–317, Mar. 2003.

[12] J. L. Crassidis, F. L. Markley, and Y. Cheng, "A survey of nonlinear attitude estimation methods," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 1, pp. 12–28, 2007.

[13] G. Bourmaud, R. Mégret, M. Arnaudon, and A. Giremus, "Continuous-discrete extended Kalman filter on matrix Lie groups using concentrated Gaussian distributions," *Journal of Mathematical Imaging and Vision*, vol. 51, no. 1, pp. 209–228, 2015.

[14] G. Kurz, I. Gilitschenski, and U. D. Hanebeck, "Recursive bayesian filtering in circular state spaces," *IEEE Aerospace and Electronic Systems Magazine*, vol. 31, no. 3, pp. 70–87, Mar. 2016.

[15] G. Kurz, I. Gilitschenski, S. Julier, and U. D. Hanebeck, "Recursive Bingham filter for directional estimation involving 180 degree symmetry," *Journal of Advances in Information Fusion*, vol. 9, no. 2, pp. 90–105, Dec. 2014.

[16] M. A. Skoglund, G. Hendeby, and D. Axehill, "Extended Kalman filter modifications based on an optimization view point," in *Proceedings of 18th IEEE International Conference on Information Fusion*, Washington, DC, USA, Jul. 6–9 2015.

[17] O. Rodrigues, "Des lois géométriques qui régissent les déplacements d'un système solide dans l'espace, et de la variation des coordonnées provenant de ces déplacements considérés indépendamment des causes qui peuvent les produire," *Journal de Mathématiques Pures et Appliquées*, vol. 5, pp. 380–440, 1840.

[18] Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry, *An Invitation to 3-D Vision: From Images to Geometric Models*.  Springer Verlag, 2003.

[19] S. W. Hamilton, "On quaternions; or on a new system of imaginaries in algebra," *Philosophical Magazine*, vol. xxv, pp. 10–13, Jul. 1844.

[20] M. D. Shuster, "Survey of attitude representations," *Journal of the Astronautical Sciences*, vol. 41, pp. 439–517, Oct. 1993.

[21] W. J. Rugh, *Linear System Theory*.  Prentice Hall, Engelwood Cliffs, NJ, 2nd ed., 1996.

[22] D. Bertsekas, "Incremental least squares methods and the extended Kalman filter," in *Proceedings of the 33rd IEEE Conference on Decision and Control*, vol. 2, Dec. 1994, pp. 1211–1214.

[23] P. J. G. Teunissen, "On the local convergence of the iterated extended Kalman filter," in *Proceeding of the XX General Assembly of the IUGG, IAG Section IV*.  Vienna: IAG, Aug. 1991, pp. 177–184.

[24] M. Kok, J. D. Hol, and T. B. Schön, "Using inertial sensors for position and orientation estimation," *ArXiv e-prints*, Apr. 2017, arXiv:1704.06053.

[25] S. J. Julier and J. J. LaViola, "On Kalman filtering with nonlinear equality constraints," *IEEE Transactions on Signal Processing*, vol. 55, no. 6, pp. 2774–2784, June 2007.