In Proceedings of 14th IEEE-RAS International Conference on Humanoid Robots (Humanoids), Madrid, Spain, November 2014.

Robust Sensor Fusion for Robot Attitude Estimation

Philipp Allgeuer and Sven Behnke

***Abstract*— Knowledge of how a body is oriented relative to the world is frequently invaluable information in the field of robotics. An attitude estimator that fuses 3-axis gyroscope, accelerometer and magnetometer data into a quaternion orientation estimate is presented in this paper. The concept of fused yaw, used by the estimator, is also introduced. The estimator, a nonlinear complementary filter at heart, is designed to be uniformly robust and stable—independent of the absolute orientation of the body—and has been implemented and released as a cross-platform open source C++ library. Extensions to the estimator, such as quick learning and the ability to deal dynamically with cases of reduced sensory information, are also presented.**

I. Introduction

Attitude estimation is the task of constructing an estimate of the full 3D orientation of a body relative to some global fixed frame, based on a finite history of sensor measurements. The body in question is often a robot, but in total generality it can correspond to any object that is equipped with the sensors necessary for the estimation task. With the advent of low cost inertial sensors—particularly those based on microelectromechanical systems (MEMS)—the field of application for attitude estimation techniques has greatly widened, extending into the field of low cost robotics. With low cost sensors and processors however, it is crucial that any estimation algorithms are able to run computationally efficiently, and are able to function with high noise inputs without excessively sacrificing estimator response. In addition to low estimator latency, orientation-independent mathematical and numerical stability is also desirable.

An attitude estimator that aims to fulfil the aforementioned criteria is presented in this paper. The estimator has been implemented as a generic portable C++ library, and is freely available online [1]. All of the algorithms and cases discussed in this paper are implemented in the release, and have been tested both in simulation and on a real humanoid platform, the NimbRo-OP [2], developed by the University of Bonn.

Much effort has been made in the past to develop algorithms for the reconstruction of attitude in aeronautical environments. This work was largely in relation to the attitude and heading reference systems (AHRS) required for aeronautical applications, with examples being the works of Gebre-Egziabher et al. [3] and Munguía and Grau [4]. Other works in the area of attitude estimation, such as Vaganay et al. [5] and Balaram [6], have focused more on robotics and control applications, but do not specifically address the issues encountered with low cost inertial measurement unit (IMU) systems. A comprehensive survey of modern nonlinear filtering methods for attitude estimation was undertaken by Crassidis et al. [7]. Almost all of the surveyed advanced filtering techniques relied on some form of the Extended Kalman Filter (EKF), with various modifications being used to improve particular characteristics of the filter—often convergence. Such EKF filters can be seen to be computationally expensive however, when considering implementation on embedded targets such as microcontrollers. It is often also difficult to provide a guarantee of filter robustness [8].

Alternative to the general stream of development of EKF filtering is the concept of complementary filtering. This builds on the well-known linear single-input single-output (SISO) complementary filters, and extends these in a nonlinear fashion to the full 3D orientation space. Such filters have favourable frequency response characteristics, and seek to fuse low frequency attitude information with high frequency attitude rate data. Prominent examples of generalised complementary filtering include the works of Jensen [9] and Mahony et al. [10].

The problem addressed in this paper relates specifically to the design of an attitude estimator that can function with noisy low cost sensors and is simple and efficient enough to be implemented at high loop rates on low power embedded targets, such as microcontrollers. To this end, the work presented by Mahony et al. in [10] was used as a basis for the attitude estimator developed in this paper. A central problem in applying this work however, is that a method is required for reconstructing an instantaneous 3D orientation 'measurement' directly from the sensor measurements. This is a complex optimisation problem that generally requires a suboptimal solution algorithm for computational feasibility reasons [10]. Literature does not elucidate a clear solution to this problem—in particular not in an explicit form—and not in a way that can function robustly in all cases. The contribution of this paper lies predominantly in the presentation of an algorithm for robust calculation of such instantaneous orientation measurements. Other contributions include the novel use of fused yaw (Section II-B), the integration of quick learning (Section V-A), and the explicit extension of the attitude estimator to cases of reduced sensory information (Section V). A summary of the notation and identities used in this paper is provided in the appendix.

II. Preliminaries

A. Problem Definition

The goal of attitude estimation is to calculate an estimate of the rotation of a body relative to a global fixed frame,

All authors are with the Autonomous Intelligent Systems (AIS) Group, Computer Science Institute VI, University of Bonn, Germany. Email: pallgeuer@ais.uni-bonn.de. This work was partially funded by grant BE 2556/10 of the German Research Foundation (DFG).

based on observations acquired through sensory perception. Such sensory perception can include accelerometers, gyroscopes, magnetometers, Global Positioning System (GPS), visual perception and/or LIDAR. The types of sensors considered for the task in this paper are the ones that are typically found in IMU systems, and typically available in low cost variants for mobile robotic systems. These are the first three in the preceding list. Irrespective of which sensors are used however, it is a stringent requirement that the estimator always remains stable, and is able to function equally well throughout the entire orientation space.

We define $\{\mathrm{B}\}$ to be the body-fixed frame, which rotates with the body, and with the sensors that provide the observational input to the attitude estimator. It is assumed that $\{\mathrm{B}\}$ is defined such that its z-axis points 'upwards' relative to the body, and its x-axis points 'forwards'. We define $\{\mathrm{G}\}$ to be the global fixed frame, with the convention that the z-axis points 'upwards' relative to the world. Importantly, this means that the gravity vector can be written as $\mathbf{g} = (0, 0, -g)$ in global coordinates.

Using these definitions, the problem considered in this paper can be more precisely reformulated as being the task of robustly calculating an estimate for ${}^{G}_{B}q$ (or ${}^{G}_{B}R$), given arbitrary 3-axis gyroscope, accelerometer and magnetometer data. The format and type of data provided by each of these sensors is assumed to be modelled as follows.

1) Gyroscope: This sensor is assumed to provide a measure ${}^{B}\boldsymbol{\Omega}_{y} \in \mathbb{R}^3$ of the angular velocity of the body, in the coordinates of frame $\{\mathrm{B}\}$. The measurement is assumed to be affected by a largely time-invariant gyroscope bias $\mathbf{b}_{\Omega}$, as well as zero mean sensor noise $\mathbf{v}_{\Omega}$. That is,

$$ {}^{B}\boldsymbol{\Omega}_{y} = {}^{B}\boldsymbol{\Omega} + \mathbf{b}_{\Omega} + \mathbf{v}_{\Omega} \in \mathbb{R}^3, \tag{1} $$

where ${}^{B}\boldsymbol{\Omega}$ is the true angular velocity of the body. The measurement ${}^{B}\boldsymbol{\Omega}_{y}$ must be expressed in $\mathrm{rad\,s^{-1}}$.

2) Accelerometer: This sensor is assumed to provide a measure ${}^{B}\tilde{\mathbf{a}} \in \mathbb{R}^3$ of the proper acceleration of the body. This is the inertial acceleration being experienced by the body, together with the effect of gravitational acceleration. The latter term is assumed to dominate the measured proper acceleration. Cases where this assumption is violated, such as in collisions, are implicitly filtered out by the low-pass dynamics of the estimator. Merging the inertial acceleration components into the noise term $\mathbf{v}_a$ gives

$$ {}^{B}\tilde{\mathbf{a}} = {}^{B}_{G}R\,{}^{G}\mathbf{g} + \mathbf{b}_a + \mathbf{v}_a \in \mathbb{R}^3, \tag{2} $$

where ${}^{G}\mathbf{g} = (0, 0, -g)$ is the global gravity vector, and $\mathbf{b}_a$ is a time-invariant accelerometer bias. It is assumed that an estimate $\hat{\mathbf{b}}_a$ of the bias is available from an accelerometer calibration, and can be used to unbias ${}^{B}\tilde{\mathbf{a}}$. Normalising the unbiased ${}^{B}\tilde{\mathbf{a}}$ with zero noise, and recalling (28), yields

$$ {}^{B}\mathbf{z}_{G} = -\frac{{}^{B}\tilde{\mathbf{a}} - \hat{\mathbf{b}}_a}{\|{}^{B}\tilde{\mathbf{a}} - \hat{\mathbf{b}}_a\|} \in S^2. \tag{3} $$

Thus, with the assumption that the accelerometer measurement points in the direction of gravity, an instantaneous measurement of ${}^{B}\mathbf{z}_{G}$ can be derived for use in the filter.

3) Magnetometer: This sensor is assumed to provide a measure ${}^{B}\tilde{\mathbf{m}} \in \mathbb{R}^3$ of the strength and direction of ${}^{G}\mathbf{m}_{e}$, the Earth's magnetic field, in the coordinates of frame $\{\mathrm{B}\}$. The measurement is assumed to be affected by a largely time invariant bias $\mathbf{b}_m$, induced by local magnetic disturbances, as well as zero mean sensor noise $\mathbf{v}_m$. Therefore

$$ {}^{B}\tilde{\mathbf{m}} = {}^{B}_{G}R\,{}^{G}\mathbf{m}_{e} + \mathbf{b}_m + \mathbf{v}_m \in \mathbb{R}^3, \tag{4} $$

where ${}^{B}_{G}R$ is the inverse of the current true orientation of the body. It is the purpose of a hard-iron magnetometer calibration to derive an estimate $\hat{\mathbf{b}}_m$ for $\mathbf{b}_m$, which is then subtracted from all future measurements. Assuming a non-zero unbiased field strength, subsequent normalisation yields

$$ {}^{B}\mathbf{m} = \frac{{}^{B}\tilde{\mathbf{m}} - \hat{\mathbf{b}}_m}{\|{}^{B}\tilde{\mathbf{m}} - \hat{\mathbf{b}}_m\|} \approx {}^{B}_{G}R \cdot \frac{{}^{G}\mathbf{m}_{e}}{\|{}^{G}\mathbf{m}_{e}\|} \in S^2. \tag{5} $$

B. Yaw of a Rotation

There are many different possible definitions for the *yaw* of a rotation. Most of these definitions are equivalent to the first parameter of one of the twelve Euler angle rotation representations. Various different conventions of Euler angles exist, some alternately referred to as Tait-Bryan angles, and as such many different definitions of yaw can be derived. However, as yaw should intuitively correspond to some notion of how rotated a frame is about the global axis that points 'upwards'—in this case the z-axis—the natural choice here is the Z-Y′-X″ Euler angle convention. The Z component of the ZYX Euler angles representation of a rotation is henceforth referred to as the *ZYX yaw* of that rotation, and denoted ψ_E.

Analysis of the definition of ZYX yaw leads to a particularly useful characterisation thereof—the ZYX yaw of a rotation from frame $\{\mathrm{A}\}$ to frame $\{\mathrm{B}\}$ is equivalent to the angle about $\mathbf{z}_A$ from $\mathbf{x}_A$ to the projection of $\mathbf{x}_B$ onto the $\mathbf{x}_A\mathbf{y}_A$ plane. Thus, two frames $\{\mathrm{B}\}$ and $\{\mathrm{C}\}$ have the same ZYX yaw relative to $\{\mathrm{A}\}$ if the projections of their respective x-axes onto the $\mathbf{x}_A\mathbf{y}_A$ plane are parallel. Note that 'parallel' here is a stronger assertion than pure collinearity (refer to the appendix). From this characterisation it can be seen that the ZYX yaw goes undefined when $\mathbf{x}_B$ is collinear with $\mathbf{z}_A$. This corresponds to the well-known gimbal lock phenomenon, and is a singularity of this definition of yaw. For the sample application of a humanoid robot however, it is not uncommon that this configuration is reached, which can be problematic depending on implementation. ZYX yaw also does not possess some properties that can be useful in a definition of yaw, such as negation through rotation inversion.

In light of these issues, the notion of *fused yaw* is proposed as an alternative definition of yaw. Given two frames $\{\mathrm{A}\}$ and $\{\mathrm{B}\}$, in general there is a unique rotation that maps $\mathbf{z}_B$ onto $\mathbf{z}_A$, such that the axis of rotation is perpendicular to both $\mathbf{z}_B$ and $\mathbf{z}_A$. $\{\mathrm{C}\}$ is defined to be the frame that results from applying this rotation to $\{\mathrm{B}\}$. The fused yaw of the rotation from $\{\mathrm{A}\}$ to $\{\mathrm{B}\}$, denoted ψ_F, is defined as the angle from $\mathbf{x}_A$ to $\mathbf{x}_C$ about $\mathbf{z}_A$. This definition is only ambiguous if $\mathbf{z}_A$ is antiparallel to $\mathbf{z}_B$. This corresponds to a point of

singularity of fused yaw—unavoidable in general definitions of yaw—and can be thought of as the frame $\{\mathrm{B}\}$ being 'upside down' relative to $\{\mathrm{A}\}$. Note that if $\mathbf{z}_A$ and $\mathbf{z}_B$ are parallel, then $\{\mathrm{C}\}$ is unambiguously taken to be $\{\mathrm{B}\}$, and the fused yaw is still well-defined. Although beyond the scope of this paper, the definition of fused yaw is consistent, well-defined, and satisfies the axiomatic conditions one would expect of an expression of yaw. Fused yaw also has some useful properties, such as negation through rotation inversion.

C. 1D Linear Complementary Filter

A simple preliminary approach to the attitude estimation problem is to separate the problem into each of its independent axes of rotation. This can work well for body rotations close to the upright identity pose, but does not extend well to the whole orientation space. Nevertheless, the 1D filtering approach demonstrates well the concept of linear complementary filtering. Taking for example the pitch direction of rotation, one can express the filter equations as

$$\dot{\hat{\theta}} = (\omega_y - \hat{b}_\omega) + k_p(\theta_y - \hat{\theta}) \tag{6}$$

$$\dot{\hat{b}}_\omega = -k_i(\theta_y - \hat{\theta}), \tag{7}$$

where $\hat{\theta}$ is the pitch angle estimate, θ_y is an instantaneous measure of the pitch angle based solely on the accelerometer, ω_y is the gyroscope measurement in the pitch direction, $\hat{b}_\omega$ is an estimate of the bias thereof, and k_p and k_i are PI compensator gains. A similar expression can be formulated for the roll and yaw directions, where in the latter case the $\theta_y - \hat{\theta}$ error term is left to zero. The PI compensator closes the loop on the type I system, forming a linear second order system with zero theoretical steady state error to step inputs. The linear complementary filter combines the high-pass rate data with the low-pass position data to form a high bandwidth estimate of the system state. However, despite possessing positive filter attributes, the assumption that each axis behaves independently places a severe limitation on the usability of the filter for attitude estimation. A core issue is that the angular velocity about one axis generally affects the rotation about all axes, and to differing amounts depending on the orientation of the body. The 1D filter also does not lead unambiguously to some notion of a total 3D orientation.

III. 3D Nonlinear Passive Complementary Filter

A. Motivation and Filter Type

In light of the limitations of the 1D complementary filter, it is desirable to formulate a complementary filter that operates on the full 3D rotation space, ideally retaining the positive frequency attributes of the linear filter. Mahony et al. [10] introduced three such nonlinear filters, the direct, passive and explicit complementary filters. The main difference between the three filters is that while the direct complementary filter uses the instantaneous inertial sensor data to transform the gyroscope measurements in the update equation, the passive complementary filter uses the current filter estimate, and the explicit complementary filter uses an update technique that operates directly on the sensor measurement vectors.

A key design decision of the attitude estimator presented here is that the magnetometer measurements should not have any direct influence on the attitude estimate, other than to resolve the yaw. The reason for this is to reduce instabilities in the output pitch and roll components, and to alleviate the requirement for a magnetometer calibration for these components of the estimate to function correctly. This is not possible to achieve with the explicit complementary filter, and so the only filter in [10] to provide a solution to the problem of constructing an instantaneous orientation measurement from sensor data was found to be unsuitable. Comparison of the direct and passive filters also led to the conclusion that the feed-forward nature of the direct formulation was unsuitable due to high frequency noise considerations. Consequently, the attitude estimator presented in this paper was built around the core of the nonlinear passive complementary filter.

B. Passive Complementary Filter Equations

We define frame $\{\mathrm{E}\}$ as the frame corresponding to the current estimate of the body's orientation, ${}^G_E\hat{q} \equiv \hat{q}$. Given the current sensor measurements ${}^B\mathbf{z}_G$ and ${}^B\mathbf{m}$, and if needed also $\hat{q}$, the first task is to construct a full 3D 'measured' orientation ${}^G_B q_y \equiv q_y$ that is consistent with these measurements. Frame $\{\mathrm{B}\}$ is implicitly defined via this measured orientation. The error in the current orientation estimate with respect to the sensor measurements is expressed as $\tilde{q} = \hat{q}^* q_y$, where $\tilde{q} \equiv (\tilde{q}_0, \tilde{\mathbf{q}}) \equiv {}^E_B\tilde{q}$. The axis of this rotation leads to the corrective error feedback term $\mathbf{\Omega}_e$, which is added to the unbiased measured angular velocity using the equations

$$\mathbf{\Omega}_e = 2\tilde{q}_0\tilde{\mathbf{q}} \tag{8}$$

$$\mathbf{\Omega} = \mathbf{\Omega}_y - \hat{\mathbf{b}}_\Omega + k_p\mathbf{\Omega}_e, \tag{9}$$

where $\hat{\mathbf{b}}_\Omega$ is the current estimate of the gyroscope bias, and k_p is a P gain. The filter equations are then

$$\dot{\hat{q}} = \tfrac{1}{2}\hat{q}\Omega \tag{10}$$

$$\dot{\hat{\mathbf{b}}}_\Omega = -k_i\mathbf{\Omega}_e, \tag{11}$$

where $\Omega = (0, \mathbf{\Omega}) \in \mathbb{H}$, and k_i is an I gain. Note that mathematically (10) simply converts the angular velocity Ω into a quaternion angular velocity that can then be integrated. Trapezoidal integration is recommended for numerical implementations of these equations. We recommend that the time increment used for the numerical integration be the measured time, coerced to a suitable range, such as $[0.8, 2.2]$ times the nominal update interval of the filter. This avoids large jumps in the estimator states when lags occur and ensures greater correctness of the gyroscope integration, leading to better estimation results. The PI gains of the filter should be tuned to provide non-oscillatory yet responsive transients, as limited by sensor noise. The similarities between the passive filter and the 1D complementary filter become apparent when comparing (6–7) and (8–11).

The stability of the passive complementary filter is discussed in detail in [10]. Theoretical analysis demonstrates that there is a measure zero set in the space of all possible

measured rotation and bias errors such that equilibrium exists despite lack of convergence. The equilibrium is unstable however, and the error is locally exponentially stable in all other cases. This set consists of all error states such that $\hat{\mathbf{b}}_\Omega$ is error-free, and $\tilde{q}$ is a rotation by π radians. This pathological set is of no concern however, as it is never reached in any practical situation. Even intentional initialisation of the filter to such an equilibrium state in simulated experiments did not prove to be a problem, as mere arithmetic floating point errors were enough for the divergent dynamics near the pathological set to take over.

IV. Measured Quaternion Orientation Resolution

A. General Case

The calculation of the estimation error quaternion $\tilde{q}$, requires knowledge of q_y, the instantaneous measured orientation best fitting the sensor measurements ${}^B\mathbf{z}_G$ and ${}^B\mathbf{m}$. In general, these two measurements suffice to construct a unique rotation q_y that best fits the given data. If not, $\hat{q}$ is taken as a further input, and one of the *resolution methods* described in the following sections is used. In absolutely all cases however, $q_y \equiv {}^G_B q_y$ must respect ${}^B\mathbf{z}_G$. This ensures that the magnetometer, and any assumptions made by the resolution methods, as desired do not affect the pitch and roll components of the output quaternion estimate.

In the general case, a value for ${}^G\mathbf{m}_e$ is required. This is easily obtained by physically rotating the body such that the $\{\mathrm{B}\}$ and $\{\mathrm{G}\}$ frames coincide, and setting ${}^G\mathbf{m}_e$ to ${}^B\mathbf{m}$. This vector is only used as a reference relative to which the yaw of the output quaternion is expressed. The goal is to find a suitable rotation matrix ${}^G_B R_y = \begin{bmatrix} {}^B\mathbf{x}_G & {}^B\mathbf{y}_G & {}^B\mathbf{z}_G \end{bmatrix}^T$, and convert it into the required quaternion ${}^G_B q_y$. Refer to the appendix for details of a known robust conversion algorithm.

Ideally we would wish to be able to find ${}^B\mathbf{x}_G$ and ${}^B\mathbf{y}_G$ such that ${}^G\mathbf{m}_e$ and ${}^B\mathbf{m}$ are equal, but as this is not necessarily possible, we instead minimise the angular difference between the two. This condition can be seen to be satisfied when the respective projections of the two vectors onto the plane perpendicular to ${}^B\mathbf{z}_G$ are parallel. We define ${}^B\hat{\mathbf{m}}$ to be the projection of ${}^B\mathbf{m}$ onto the $\mathbf{x}_G\mathbf{y}_G$ plane, and use the cross product to construct a suitable third basis vector ${}^B\hat{\mathbf{u}}$. The required ${}^B\mathbf{x}_G$ and ${}^B\mathbf{y}_G$ vectors are then calculated as linear combinations of these basis vectors, based on the condition that ${}^B\hat{\mathbf{m}}$ must be parallel to ${}^G\hat{\mathbf{m}}_e = (m_{ex}, m_{ey}, 0)$, the trivial projection of ${}^G\mathbf{m}_e$ onto the $\mathbf{x}_G\mathbf{y}_G$ plane. This algorithm can be summarised mathematically as

$$ {}^B\hat{\mathbf{m}} = {}^B\mathbf{m} - ({}^B\mathbf{m} \cdot {}^B\mathbf{z}_G){}^B\mathbf{z}_G \tag{12} $$

$$ {}^B\hat{\mathbf{u}} = {}^B\hat{\mathbf{m}} \times {}^B\mathbf{z}_G \tag{13} $$

$$ {}^B\tilde{\mathbf{x}}_G = m_{ex}{}^B\hat{\mathbf{m}} + m_{ey}{}^B\hat{\mathbf{u}} \tag{14} $$

$$ {}^B\tilde{\mathbf{y}}_G = m_{ey}{}^B\hat{\mathbf{m}} - m_{ex}{}^B\hat{\mathbf{u}} \tag{15} $$

$$ {}^B\mathbf{x}_G = \frac{{}^B\tilde{\mathbf{x}}_G}{\|{}^B\tilde{\mathbf{x}}_G\|}, \quad {}^B\mathbf{y}_G = \frac{{}^B\tilde{\mathbf{y}}_G}{\|{}^B\tilde{\mathbf{y}}_G\|} \tag{16} $$

$$ {}^G_B R_y = \begin{bmatrix} {}^B\mathbf{x}_G & {}^B\mathbf{y}_G & {}^B\mathbf{z}_G \end{bmatrix}^T \tag{17} $$

Note that m_{ez} is not required. This algorithm only fails with a division by zero if ${}^G\hat{\mathbf{m}}_e$ is degenerate, or ${}^B\mathbf{z}_G$ and ${}^B\mathbf{m}$ are collinear. Both these causes of failure correspond to measurements of the Earth's magnetic field being vertical in the global fixed frame, a generally unexpected case.

B. ZYX Yaw Orientation Resolution Method

If the general case fails to produce a valid output, the magnetometer measurement is discarded, and a measured orientation q_y is instead constructed from ${}^B\mathbf{z}_G$ and $\hat{q}$. The latter is required as the former alone is insufficient to be able to calculate a unique q_y, and the latter can be used to ensure that q_y is 'as close as possible' to $\hat{q}$, thereby only minimally affecting the estimate in uncontrolled dimensions (i.e. yaw).

We define the frame $\{\mathrm{H}\}$ to be the frame $\{\mathrm{B}\}$ rotated by the inverse of $\hat{q}$. That is, $\{\mathrm{H}\}$ corresponds to the current estimated orientation of the global fixed frame. Note that this will not be identical to $\{\mathrm{G}\}$ in general, as $\hat{q}$ and q_y generally differ, even if only slightly. The aim of the ZYX yaw resolution method is to find a suitable rotation matrix ${}^G_B R_y$, such that the ZYX yaw of $\{\mathrm{H}\}$ with respect to $\{\mathrm{G}\}$ is zero. This is equivalent to saying that ${}^B\mathbf{x}_G$ should be parallel, and hence equal to, the normalised projection of ${}^B\mathbf{x}_H$ onto the $\mathbf{x}_G\mathbf{y}_G$ plane. ${}^B\mathbf{y}_G$ is calculated to complete the orthogonal basis. Letting $\hat{q} = (\hat{w}, \hat{x}, \hat{y}, \hat{z})$, the algorithm is mathematically given as

$$ {}^B\mathbf{x}_H = (1 - \hat{w}^2 - \hat{z}^2,\ \hat{x}\hat{y} - \hat{w}\hat{z},\ \hat{x}\hat{z} + \hat{w}\hat{y}) \tag{18} $$

$$ {}^B\tilde{\mathbf{x}}_G = {}^B\mathbf{x}_H - ({}^B\mathbf{x}_H \cdot {}^B\mathbf{z}_G){}^B\mathbf{z}_G \tag{19} $$

$$ {}^B\tilde{\mathbf{y}}_G = {}^B\mathbf{z}_G \times {}^B\tilde{\mathbf{x}}_G \tag{20} $$

after which equations (16–17) are used as before. The obtained rotation matrix ${}^G_B R_y$ is converted into the required quaternion ${}^G_B q_y$. This algorithm only fails, in the form of a division by zero, if ${}^B\mathbf{z}_G$ and ${}^B\mathbf{x}_H$ are collinear. This is only the case if the error quaternion $\tilde{q} \equiv \hat{q}^* q_y$ is in gimbal lock in terms of ZYX Euler angles. It is important to note that failure of this method depends only on the *error quaternion*, and not in any way on the absolute rotations $\hat{q}$ and q_y. As a result, the algorithm is equally stable in all global orientations of the body, as desired.

If the algorithm fails, a backup algorithm that zeros the Euler ZXY yaw of $\{\mathrm{H}\}$ with respect to $\{\mathrm{G}\}$ is employed instead. Analogously to (18–20),

$$ {}^B\mathbf{y}_H = (\hat{x}\hat{y} + \hat{w}\hat{z},\ 1 - \hat{x}^2 - \hat{z}^2,\ \hat{y}\hat{z} - \hat{w}\hat{x}) \tag{21} $$

$$ {}^B\tilde{\mathbf{y}}_G = {}^B\mathbf{y}_H - ({}^B\mathbf{y}_H \cdot {}^B\mathbf{z}_G){}^B\mathbf{z}_G \tag{22} $$

$$ {}^B\tilde{\mathbf{x}}_G = {}^B\tilde{\mathbf{y}}_G \times {}^B\mathbf{z}_G \tag{23} $$

after which equations (16–17) are used the same as before. Given that the previous algorithm failed, this algorithm is guaranteed never to, hence completing the ZYX yaw method.

C. Fused Yaw Orientation Resolution Method

The fused yaw resolution method is quite similar in idea to the ZYX yaw method, only instead of zeroing the ZYX yaw of $\{\mathrm{H}\}$ with respect to $\{\mathrm{G}\}$, it zeros the fused yaw. The first notable distinction here to before is that having zero relative

fused yaw is in fact a mutual relationship, as the inverse of a rotation has the exact negative of its fused yaw. The second notable distinction is that the notion of fused yaw is more closely related to quaternions than ZYX yaw, and so a convenient direct quaternion formulation exists. Treating quaternions notationally in (25) as column vectors in $\mathbb{R}^4$, the algorithm can be summarised mathematically as

$$ {}^{H}\mathbf{z}_G = L_{\hat{q}}({}^{B}\mathbf{z}_G) = \hat{q}\,{}^{B}\mathbf{z}_G\,\hat{q}^* = (z_{Gx}, z_{Gy}, z_{Gz}) \tag{24} $$

$$ {}^{G}_{B}\bar{q}_y = \begin{bmatrix} 1+z_{Gz} & -z_{Gy} & z_{Gx} & 0 \\ z_{Gy} & 1+z_{Gz} & 0 & -z_{Gx} \\ -z_{Gx} & 0 & 1+z_{Gz} & -z_{Gy} \\ 0 & z_{Gx} & z_{Gy} & 1+z_{Gz} \end{bmatrix} {}^{G}_{E}\hat{q} \tag{25} $$

where (24) is calculated numerically as in (29), and ${}^{G}_{B}q_y$ is subsequently calculated as the normalisation of ${}^{G}_{B}\bar{q}_y$. The mathematical proof of the correctness of this algorithm is beyond the scope of this paper. From inspection it can be seen however, that the only case in which the algorithm fails is ${}^{H}\mathbf{z}_G = (0, 0, -1)$. This is the case if the *error quaternion* $\tilde{q}$ is a rotation by exactly π radians about an axis in the $\mathbf{xy}$ plane. This is a subset of the conditions on $\tilde{q}$ required for unstable equilibrium of the passive filter itself. As such, the use of the fused yaw resolution method ensures that there is only a single error condition for which any part of the total passive filter yields suboptimal results. Furthermore, this one error condition is when $\tilde{q}$ is at an exact antipode of the identity rotation—a case that in practical situations is never reached. Nevertheless, for reasons of completeness and robustness, the above algorithm falls back to zeroing the ZYX yaw if it fails. This is computed using (16–20), and is guaranteed not to fail if the fused yaw algorithm failed. It is important to note that this resolution algorithm is once again equally stable in all global orientations of the body.

V. Extensions to the Estimator

A. Quick Learning

It is desired for the attitude estimator to settle quickly from large estimation errors, yet simultaneously provide adequate general noise rejection. To this end, we propose *quick learning* as a method to help achieve this. Quick learning allows two sets of PI gains to be tuned—one set that provides suitably fast transient response, and one set that provides good tracking and noise rejection. Given a desired quick learning time, a parameter $\lambda \in [0, 1]$ is used to fade linearly between the two sets of gains, ending in the nominal setpoint tracking gains. The gain fading scheme is given by

$$ (k_p, k_i) = \lambda(k_p^{nom}, k_i^{nom}) + (1-\lambda)(k_p^{quick}, k_i^{quick}). $$

Quick learning can be triggered at any time, including automatically when the estimator starts, and is disabled when λ reaches 1.

B. Estimation with Two-Axis Acceleration Data

If only two-axis $\mathbf{xy}$ accelerometer data is available, then the missing z-component of ${}^{B}\tilde{\mathbf{a}}$ can be calculated by solving $\|{}^{B}\tilde{\mathbf{a}}\| = g$, where g is the magnitude of gravitational acceleration. Letting ${}^{B}\tilde{\mathbf{a}} = (a_x, a_y, a_z)$, this yields

$$ a_z = -\sqrt{\max\{g^2 - a_x^2 - a_y^2, 0\}}\,. \tag{26} $$

This however, only allows for attitude estimates in the positive z-hemisphere, as the sign of the missing a_z component has to be assumed. For many applications, like bipedal walking, this can be sufficient.

C. Estimation with Reduced Magnetometer Data

Two-axis $\mathbf{xy}$ magnetometer data can still be used for ${}^{B}\tilde{\mathbf{m}}$ if the third unknown component is left to zero. Due to the projection operation in (12), this in general still produces satisfactory results. The calibration process of ${}^{G}\mathbf{m}_e$, the Earth's magnetic field, also remains the same, as the z-component thereof is not required for the orientation resolution algorithm. If magnetometer data is only available in terms of relative heading angle ψ, then the required three-axis data can be constructed using ${}^{B}\mathbf{m} = (\cos\psi, \sin\psi, 0)$, and used as before.

D. Estimation without Magnetometer Data

If no magnetometer data is available in a system, then the attitude estimator can still be used without any degradation in the estimation quality in the pitch and roll dimensions by setting ${}^{B}\mathbf{m}$ and ${}^{G}\mathbf{m}_e$ to zero. In this case, the estimation relies solely on the selected yaw-based orientation resolution method. Due to the yaw-zeroing approach used, the open-loop yaw produced by the estimator remains stable with each update of q_y. The linear combination of gyroscope biases that corresponds to rotations in the instantaneous $\mathbf{x}_G\mathbf{y}_G$ plane however does not have feedback, so small constant global yaw velocities in $\hat{q}$ can result. This yaw drift is unavoidable as no yaw feedback is present without magnetometer data. A more stable output quaternion $\hat{q}_s$ can be obtained by removing the fused yaw component of the estimate. Letting $\hat{q} = (\hat{w}, \hat{x}, \hat{y}, \hat{z})$, this can be done using

$$ \tilde{\hat{q}}_s = (\hat{w}, 0, 0, -\hat{z})\,\hat{q}, \quad \hat{q}_s = \frac{\tilde{\hat{q}}_s}{\|\tilde{\hat{q}}_s\|}. \tag{27} $$

We do not recommend removing the ZYX yaw instead, as this leads to unexpected behaviour near the not uncommon scenario of pitch rotations by $\frac{\pi}{2}$ radians.

VI. Experimental Results

Thorough experimentation and testing of the proposed attitude estimator and corresponding C++ implementation has been performed in simulation and on multiple humanoid robots. Presented in Fig. 1 are the results of three parallel instances of the attitude estimator running on a NimbRo-OP robot. The same L3G4200D gyroscope measurements, LIS331DLH accelerometer measurements, and HMC5883L magnetometer measurements were made available to each of the estimators. One of the estimators was permitted to use the magnetometer data, while the other two were configured to only use the ZYX yaw and fused yaw orientation resolution methods respectively. It can be

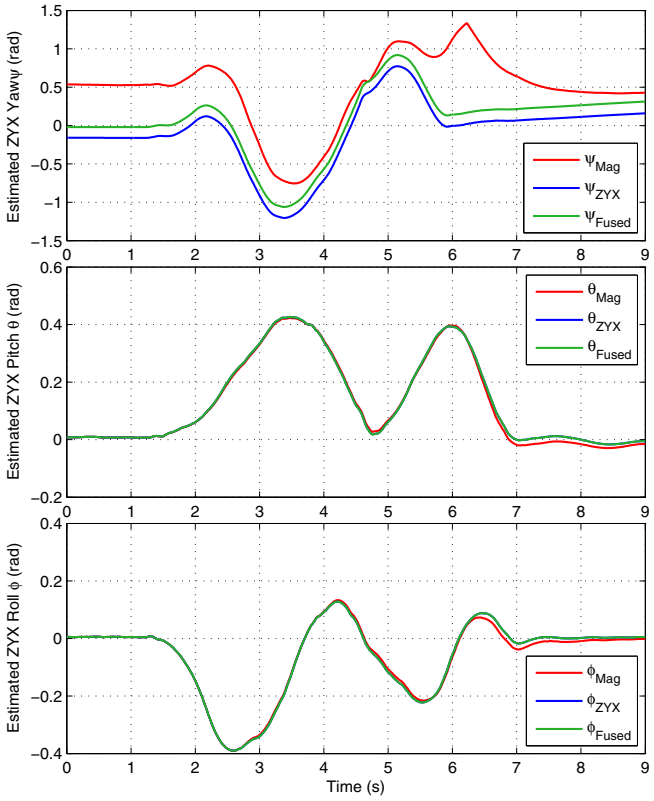

Fig. 1. Estimation results on the NimbRo-OP robot, using the magnetometer, ZYX yaw orientation resolution, and fused yaw orientation resolution.

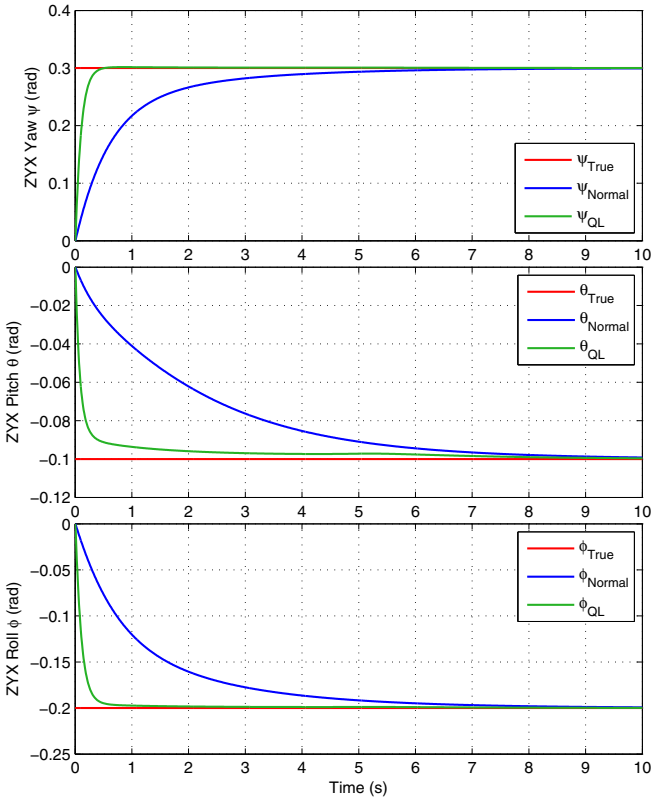

Fig. 2. Simulated attitude estimation results demonstrating the effect of quick learning on the filter's transient response.

seen that the results of the two yaw resolution methods are virtually indistinguishable, apart from in the ZYX yaw plot, where there is a vertical shift between the two curves. This can be expected to happen due to the lack of feedback in the yaw axis in these two methods. A (more approximate) vertical shift can also be seen between the non-magnetometer yaw curves and the yaw produced by the magnetometer method. An important observation is that while the use of the magnetometer allows the yaw estimation to be meaningful and absolute instead of just relative, it only minimally affects the pitch and roll, as desired. The effect on the pitch and roll is not exactly zero though, due to the interplay between the added yaw information and the measured angular velocities. Note that the magnetometer chip temporarily provided incorrect values shortly after 5.0 s. The estimator recovery, and performance in the pitch and roll directions despite this significant input disturbance, attests to the stability of the estimator. In applications where a yaw orientation estimate is required, the magnetometer method is the clear choice, but suitable magnetometer measurements must be available. Otherwise, the authors recommend the fused yaw resolution method for mathematical and performance reasons.

The effect of quick learning on the filter's transient response is shown in Fig. 2. For a simulated step in the true orientation of a body, the response of the filter with and without quick learning activated is shown. The quick learning time used was 3.0 s. The ψ_{QL}, θ_{QL} and ϕ_{QL} waveforms, using the quick learning feature, show clear improvement in rise time and settling time over the ψ_{Normal}, θ_{Normal} and ϕ_{Normal} waveforms, for which quick learning was disabled.

The attitude estimator was designed to be able to run at high loop rates on embedded hardware, so as to minimise estimation and possible control feedback latencies. The C++ attitude estimator library code was tested on a PC with a 2.40 GHz Intel i5-2430M processor. On a single CPU core, the average execution time of the estimator over 100 million cycles was found to be 127.6 ns for the magnetometer method, 144.3 ns for the ZYX yaw method, and 112.3 ns for the fused yaw method. It is to be expected that the fused yaw method takes comparatively less time, as it does not in general require a rotation matrix to quaternion conversion, unlike the other two methods. From these results it is confidently anticipated that the algorithm is efficient enough to be implemented at high execution rates on a low cost microcontroller, where floating and/or fixed point operations are comparatively more expensive than on a PC.

VII. Conclusions

A filter for attitude estimation, released online in the form of a C++ library [1], has been presented in this paper. The filter uses the technique of complementary filtering, and builds on the nonlinear passive complementary filter presented by Mahoney et al. in [10], providing robust algorithms for the reconstruction of a 'measured' orientation from instantaneous sensor data. The filter is equally stable in all global body orientations, and only demonstrates potential non-convergent behaviour on a pathological set that is of no practical concern. Extensions to the filter allow for reliable

attitude estimation in situations of reduced sensory data, and the advent of quick learning allows for quicker settling times from large estimation errors when required. The output of the presented attitude estimator can be used, for example, for the analysis and control of balance in a biped robot. For this task it can be beneficial to further decompose the estimated orientation into its components in each of the major planes. Future work includes a method for accounting for the inertial components of the accelerometer measurements based on other sensors and/or system information.

APPENDIX

This appendix introduces the notation, definitions and well-known identities that are used throughout this paper. The set of all unit vectors in $\mathbb{R}^3$, the 2-sphere, is denoted S^2. The set of all rotation matrices is called the special orthogonal group SO(3), and is defined as

$$\mathrm{SO}(3) = \{R \in \mathbb{R}^{3\times 3} : R^T R = \mathbb{I},\, \det(R) = 1\}.$$

Rotation of a vector $\mathbf{v} \in \mathbb{R}^3$ by a rotation matrix is given by matrix multiplication. For a rotation from coordinate frame $\{\mathrm{A}\}$ to frame $\{\mathrm{B}\}$, we have that

$$\overset{A}{B}R = \begin{bmatrix} {}^A\mathbf{x}_B & {}^A\mathbf{y}_B & {}^A\mathbf{z}_B \end{bmatrix} = \begin{bmatrix} {}^B\mathbf{x}_A & {}^B\mathbf{y}_A & {}^B\mathbf{z}_A \end{bmatrix}^T, \quad (28)$$

where for example ${}^A\mathbf{y}_B$ is the column vector corresponding to the y-axis of frame $\{\mathrm{B}\}$, expressed in the coordinates of $\{\mathrm{A}\}$. The term $\overset{A}{B}R$ refers to the rotation from $\{\mathrm{A}\}$ to $\{\mathrm{B}\}$.

The set of all quaternions $\mathbb{H}$, and the subset $\mathbb{Q}$ thereof of all quaternions that represent pure rotations, are defined as

$$\mathbb{H} = \{q = (q_0, \mathbf{q}) = (w, x, y, z) \in \mathbb{R}^4\}$$
$$\mathbb{Q} = \{q \in \mathbb{H} : |q| = 1\}.$$

The rotation of a vector $\mathbf{v} \in \mathbb{R}^3$ by a quaternion $q \in \mathbb{Q}$ is given by the function $L_q(\mathbf{v}) : \mathbb{R}^3 \to \mathbb{R}^3$, defined as

$$\begin{aligned} L_q(\mathbf{v}) &= q\mathbf{v}q^* \\ &= (q_0^2 - \|\mathbf{q}\|^2)\mathbf{v} + 2(\mathbf{q}\cdot\mathbf{v})\mathbf{q} + 2q_0(\mathbf{q}\times\mathbf{v}) \\ &= \mathbf{v} + q_0\mathbf{t} + \mathbf{q}\times\mathbf{t}, \end{aligned} \quad (29)$$

where $t = 2(\mathbf{q}\times\mathbf{v})$, and q^* is the quaternion conjugate of q. Note that (29) provides the computationally most efficient method for calculating $L_q(\mathbf{v})$. Relative quaternion rotations are denoted using symbols such as $\overset{A}{B}q$.

Two vectors that are linearly dependent and a positive multiple of each other are referred to as *parallel*. Two linearly dependent vectors that are a negative multiple of each other are referred to as *antiparallel*. Two vectors that are either parallel or antiparallel are referred to as *collinear*.

The conversion between the quaternion and rotation matrix representations of a rotation is often required, but not entirely numerically trivial. Given a unit quaternion $q = (w, x, y, z) \in \mathbb{Q}$, the equivalent rotation matrix is given by

$$R = \begin{bmatrix} 1-2(y^2+z^2) & 2(xy-wz) & 2(xz+wy) \\ 2(xy+wz) & 1-2(x^2+z^2) & 2(yz-wx) \\ 2(xz-wy) & 2(yz+wx) & 1-2(x^2+y^2) \end{bmatrix}$$

Depending on how the rotation matrix R is subsequently used, it may be necessary to coerce each of the matrix entries to $[-1, 1]$. Although for $|q| = 1$ it is impossible in a mathematical sense for one of the entries to exceed unity in magnitude, it can happen in practice due to floating point arithmetic. In such cases, subsequent calculations such as $\alpha = \mathrm{acos}(R_{33})$ can lead to unwanted numerical problems.

The reverse conversion is more difficult and is split into four cases, where each case corresponds to one of the four quaternion parameters being taken as the base of the conversion. Given a rotation matrix $R \in \mathrm{SO}(3)$ with matrix entries R_{ij}, if $\mathrm{tr}(R) \geq 0$,

$$r = \sqrt{1 + R_{11} + R_{22} + R_{33}}$$
$$q = \left(\tfrac{1}{2}r, \tfrac{1}{2r}(R_{32} - R_{23}), \tfrac{1}{2r}(R_{13} - R_{31}), \tfrac{1}{2r}(R_{21} - R_{12})\right)$$

Else if $R_{33} \geq R_{22}$ and $R_{33} \geq R_{11}$,

$$r = \sqrt{1 - R_{11} - R_{22} + R_{33}}$$
$$q = \left(\tfrac{1}{2r}(R_{21} - R_{12}), \tfrac{1}{2r}(R_{13} + R_{31}), \tfrac{1}{2r}(R_{32} + R_{23}), \tfrac{1}{2}r\right)$$

Else if $R_{22} \geq R_{11}$,

$$r = \sqrt{1 - R_{11} + R_{22} - R_{33}}$$
$$q = \left(\tfrac{1}{2r}(R_{13} - R_{31}), \tfrac{1}{2r}(R_{21} + R_{12}), \tfrac{1}{2}r, \tfrac{1}{2r}(R_{32} + R_{23})\right)$$

And otherwise,

$$r = \sqrt{1 + R_{11} - R_{22} - R_{33}}$$
$$q = \left(\tfrac{1}{2r}(R_{32} - R_{23}), \tfrac{1}{2}r, \tfrac{1}{2r}(R_{21} + R_{12}), \tfrac{1}{2r}(R_{13} + R_{31})\right)$$

This implementation of the rotation matrix to quaternion conversion is extremely robust, as it always chooses as the base of the conversion the quaternion parameter that provides the most well-conditioned problem to solve.

REFERENCES

[1] P. Allgeuer. (2014, Jul) Attitude Estimator. [Online]. Available: https://github.com/AIS-Bonn/attitude_estimator/

[2] P. Allgeuer, M. Schwarz, J. Pastrana, S. Schueller, M. Missura, and S. Behnke, "A ROS-based software framework for the NimbRo-OP humanoid open platform," in *Proc. 8th Workshop on Humanoid Soccer Robots, Int. Conf. on Humanoid Robots*, Atlanta, USA, 2013.

[3] D. Gebre-Egziabher, R. Hayward, and J. Powell, "Design of multi-sensor attitude determination systems," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 40, no. 2, pp. 627–649, 2004.

[4] R. Munguía and A. Grau, "A practical method for implementing an attitude and heading reference system," *Int. Journ. Adv. Robotic Systems*, vol. 11, no. 62, pp. 1–12, 2014.

[5] J. Vaganay, J. Aldon, and A. Fournier, "Mobile robot attitude estimation by fusion of inertial data," in *Proc. IEEE Int. Conf. Robotics Automation (ICRA)*, vol. 1, 1993, pp. 277–282.

[6] J. Balaram, "Kinematic observers for articulated rovers," in *Proc. IEEE Int. Conf. Robotics Automation (ICRA)*, 2000, pp. 2597–2604.

[7] J. Crassidis, F. 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.

[8] M. Euston, P. Coote, R. Mahony, K. Jonghyuk, and T. Hamel, "A complementary filter for attitude estimation of a fixed-wing uav," in *Int. Conf. Intelligent Robots and Systems (IROS)*, 2008, pp. 340–345.

[9] K. Jensen, "Generalized nonlinear complementary attitude filter," *J. of Guidance, Cont. and Dynamics*, vol. 34, no. 5, pp. 1588–1592, 2011.

[10] R. Mahony, T. Hamel, and J.-M. Pflimlin, "Nonlinear complementary filters on the special orthogonal group," *IEEE Trans. Automat. Contr.*, vol. 53, no. 5, pp. 1203–1218, 2008.