“Magic” Madgwick Filter for Attitude Estimation

Team `sudo rm -rf *`

Abhishek Shastry
Department of Aerospace Engineering
University of Maryland
College Park 20742
Email: shastry@umd.edu

Animesh Shastry
Department of Aerospace Engineering
University of Maryland
College Park 20742
Email: animeshs@umd.edu

Nicholas Rehm
Department of Aerospace Engineering
University of Maryland
College Park 20742
Email: nrehm@umd.edu

***Abstract*—In this report the Madgwick Filter is implemented on an IMU dataset that was given as a part of the first week's assignment of the course ENAE788M: Hands on Autonomous Aerial Robotics. The performance of the filter is compared with attitude estimations from the IMU's Accelerometer, Gyroscope and a complementary filter fusion of both the Accelerometer and Gyroscope data. Additionally, Vicon data is used as the ground truth to show the remarkable accuracy of the Madgwick Filter's attitude estimates.**

I. Introduction

In many aerospace and robotics applications, accurately measuring the system's orientation plays a vital role for supplying feedback to the autopilot or the controllers. A very popular sensor for measuring orientation is an IMU (Inertial Measurement unit). It consists of tri-axis gyroscopes and accelerometers. Gyroscopes measures the angular velocity, which can be integrated over time from some known initial condition to estimate the sensor's orientation. The numerical integration technique is prone to accumulation of errors over time which leads to a drift in the estimated orientation from the true orientation. On the other hand, an accelerometer measures the earth's gravitational field and therefore can provide orientation estimates from an absolute frame of reference. However, any translational motion will result in corrupted measurements of earth's gravity and consequently the orientation estimates suffers. This particular problem can be handled by using an orientation filter that estimates a single estimate of orientation by fusing the accelerometer and gyroscope measurements provided by an IMU.
Link to the result videos: Click Here

II. Mathematical Details

A. IMU Data Pre-Processing

Data gathered from the IMU must first be pre-processed to convert into physical units and to negate bias inherent to the instrumentation. The following expression describes the conversion of raw accelerometer readings $\boldsymbol{a} = [a_x \; a_y \; a_z]^T$ to acceleration data in m/s^2.

$$\tilde{a_i} = 9.81(a_i * s_i + b_{a,i}) \tag{1}$$

where s_i is a scale factor of the accelerometer for each axis and $b_{a,i}$ is the bias for each axis.

To convert the raw gyro angular velocity reading $\boldsymbol{\omega} = [\omega_x \; \omega_y \; \omega_z]^T$ into angular velocity data in rad/s, the following expression is used.

$$\tilde{\omega_i} = \frac{3300}{1023} * \frac{\pi}{180} * .3 * (\omega_i - b_{g,i}) \tag{2}$$

where $b_{g,i}$ is calculated as the average of the first 200 raw angular velocity reading samples for each axis of rotation.

B. Attitude Quaternions

The representation of attitude/orientation using Euler Angles is associated with the problem of singularity. This can be mitigated by the use of unit length four-dimensional complex numbers called attitude quaternions. Quaternions comprise of a single real element (represented by the subscript $_0$) and three imaginary elements (represented by the subscripts $_1$, $_2$ and $_3$). The following expression describes the attitude quaternion.

$$\begin{aligned} \boldsymbol{q} &= [q_0 \;\; q_1 \;\; q_2 \;\; q_3]^T \\ ||\boldsymbol{q}||_2 &= \left(\sum_{i=0}^{3} q_i^2\right)^{1/2} = 1 \end{aligned} \tag{3}$$

A quaternion conjugate operator, denoted by * is used to swap the relative frames described by an orientation.

$$\boldsymbol{q}^* = [q_1 \;\; -q_2 \;\; -q_3 \;\; -q_4]^T \tag{4}$$

The quaternion product operator, denoted by $\otimes$ is used to define compound orientations. For two quaternions. $\boldsymbol{a}$ and $\boldsymbol{b}$. the quaternion product can be computed using the Hamilton rule as follows.

$$\begin{aligned} \boldsymbol{a} \otimes \boldsymbol{b} &= [a_0 \;\; a_1 \;\; a_2 \;\; a_3]^T \otimes [b_0 \;\; b_1 \;\; b_2 \;\; b_3]^T \\ &= \begin{bmatrix} a_0b_0 - a_1b_1 - a_2b_2 - a_3b_3 \\ a_0b_1 + a_1b_0 + a_2b_3 - a_3b_2 \\ a_0b_2 - a_1b_3 + a_2b_0 + a_3b_1 \\ a_0b_3 + a_1b_2 - a_2b_1 + a_3b_0 \end{bmatrix} \end{aligned} \tag{5}$$

It is also common to represent quaternions with scalar and vector parts, in which case, the quaterion multiplication formulation is written in a simplified manner as follows.

$$\begin{gathered}\boldsymbol{q} = (r, \vec{v}),\ \boldsymbol{q} \in \mathbb{H},\ r \in \mathbb{R},\ \vec{v} \in \mathbb{R}^3 \\ (r_1, \vec{V_1}) \otimes (r_2, \vec{V_2}) = (r_1 r_2 - \vec{v}_1 \cdot \vec{v}_2,\ r_1 \vec{v}_2 + r_2 \vec{v}_1 + \vec{v}_1 \times \vec{v}_2)\end{gathered} \tag{6}$$

To rotate a vector described in frame A to frame B, using a quaterion that describes the orientation of frame B relative to frame A, the following expression is typically used.

$$[0\ {}^{B}\boldsymbol{v}]^T = \boldsymbol{q} \otimes [0\ {}^{A}\boldsymbol{v}]^T \otimes \boldsymbol{q}^* \tag{7}$$

The z-x-y Euler angle representation of $\boldsymbol{q}$ is defined by equation (8).

$$\begin{aligned}\psi &= arc\tan 2(2q_1q_2 - 2q_0q_3,\ 2q_0^2 + 2q_1^2 - 1) \\ \theta &= -\sin^{-1}(2q_1q_3 + 2q_0q_2) \\ \phi &= arc\tan 2(2q_2q_3 - 2q_0q_1,\ 2q_0^2 + 2q_3^2 - 1)\end{aligned} \tag{8}$$

C. Madgwick Filter

A detailed description with mathematical derivation of the Madgwick Filter can be found in [1]. The filter uses the quaternion representation of orientation/attitude to avoid the singularities that are inherent in the three-dimensional Euler angle representation. A summary of the filter derivation steps that are necessary to implement the filter digitally is provided below.

1) Orientation from gyroscpe: The rate of change of the attitude quaternion $\dot{\boldsymbol{q}}_\omega$ can be calculated from the measured 3-axis gyroscope angular velocities ω_x, ω_y, ω_z and the current quaternion estimate $\hat{\boldsymbol{q}}$ as follows.

$$\begin{aligned}\boldsymbol{\omega} &= [0\ \omega_x\ \omega_y\ \omega_z]^T \\ \dot{\boldsymbol{q}}_\omega &= \frac{1}{2}\hat{\boldsymbol{q}} \otimes \boldsymbol{\omega}\end{aligned} \tag{9}$$

Rather than integrating the quaternion derivative using lie algebra (which is computationally costly), an approximation of the quaternion integration step over time interval/sampling time Δt is formulated as follows.

$$\boldsymbol{q}_\omega = \hat{\boldsymbol{q}} + \dot{\boldsymbol{q}}_\omega \Delta t \tag{10}$$

2) Orientation from accelerometer: To calculate the attitude quaternion $\hat{\boldsymbol{q}}$ from the earth's gravity $\boldsymbol{g}$ and 3-axis accelerometer sensor which provides the gravity vector data in the body frame a_x, a_y, a_z, an optimization problem is formulated as follows.

$$\min_{\hat{\boldsymbol{q}} \in \mathbb{R}^4} \boldsymbol{f}(\hat{\boldsymbol{q}}, \boldsymbol{g}, \boldsymbol{a}) \tag{11}$$

$$\boldsymbol{f}(\hat{\boldsymbol{q}}, \boldsymbol{g}, \boldsymbol{a}) = \hat{\boldsymbol{q}}^* \otimes \boldsymbol{g} \otimes \hat{\boldsymbol{q}} - \boldsymbol{a} \tag{12}$$

where

$$\begin{aligned}\boldsymbol{a} &= [0\ a_x\ a_y\ a_z]^T \\ \boldsymbol{g} &= [0\ 0\ 0\ 1]\end{aligned} \tag{13}$$

The solution to the optimization problem can be computed using gradient descent algorithm. Representing the step size as μ, the quaternion estimate at every iteration step k can be computed as follows.

$$\boldsymbol{q}_{k+1} = \hat{\boldsymbol{q}}_k - \mu \frac{\nabla \boldsymbol{f}(\hat{\boldsymbol{q}}, \boldsymbol{g}, \boldsymbol{a})}{||\nabla \boldsymbol{f}(\hat{\boldsymbol{q}}, \boldsymbol{g}, \boldsymbol{a})||} \tag{14}$$

$$\nabla \boldsymbol{f}(\hat{\boldsymbol{q}}, \boldsymbol{g}, \boldsymbol{a}) = \boldsymbol{J}^T(\hat{\boldsymbol{q}}, \boldsymbol{g}) \boldsymbol{f}(\hat{\boldsymbol{q}}, \boldsymbol{g}, \boldsymbol{a}) \tag{15}$$

$$\boldsymbol{f} = \begin{bmatrix} 2(q_2q_4 - q_1q_3) - a_x \\ 2(q_1q_2 + q_3q_4) - a_y \\ 1 - q_2^2 - q_3^2 - a_z \end{bmatrix} \tag{16}$$

$$\boldsymbol{J} = \begin{bmatrix} -2q_3 & 2q_4 & -2q_1 & 2q_2 \\ 2q_2 & 2q_1 & 2q_4 & 2q_3 \\ 0 & -4q_2 & -4q_3 & 0 \end{bmatrix} \tag{17}$$

Assuming that μ is set at such a value that the convergence rate governed by it is equal or greater than the physical rate of change of orientation, one iteration is sufficient for the optimization problem's solution to converge within an acceptable tolerance. The equation (14) can now be re-written with a subscript $_\nabla$ to indicate the use of gradient descent as follows.

$$\boldsymbol{q}_\nabla = \hat{\boldsymbol{q}}_k - \mu \frac{\nabla \boldsymbol{f}}{||\nabla \boldsymbol{f}||} \tag{18}$$

3) Filter Fusion: The orientation estimates from gyroscope/angular velocities q_w and from the accelerometer q_∇ can now be fused together using a weighted summation approach as follows.

$$\hat{\boldsymbol{q}} = \beta \boldsymbol{q}_\nabla + (1 - \beta)\boldsymbol{q}_\omega \tag{19}$$

It must be noted that in (10), (18) and (19) the resultant quaternion on the left-hand side of the equations move out of the unit quaternion space and thus they no longer represent the attitude/orientation of the body. To rectify this they must be normalized after every time they are calculated.

III. RESULTS

Datasets 1 to 6 include Vicon measurements treated as ground truth to compare the performance of the Madgwick Filter. The Datasets 7 to 10 are the evaluation/test datasets and do not contain the Vicon measurements.

A. Dataset 1

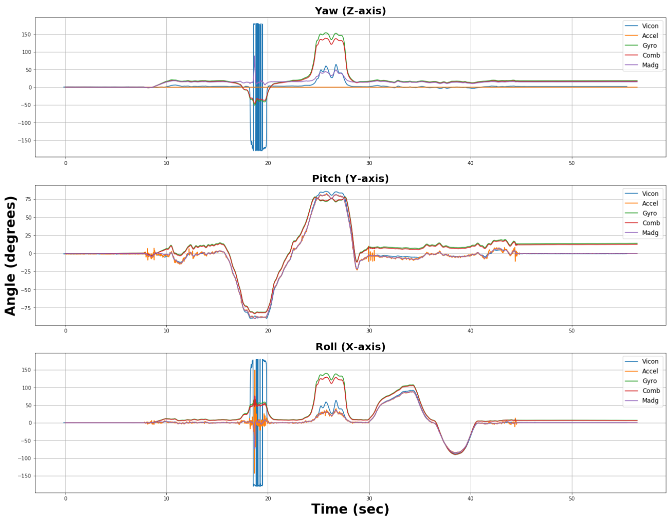

Fig. 1. Comparision of Attitude Estimation using the various filters with the Vicon Ground Truth for the first datatset.

B. Dataset 2

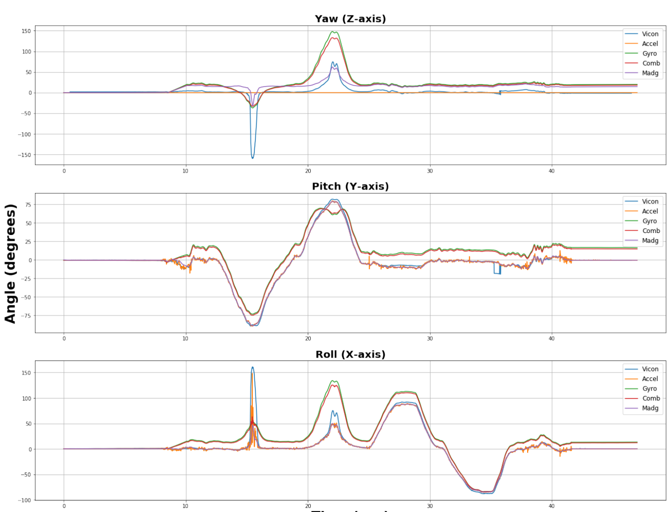

Fig. 2. Comparision of Attitude Estimation using the various filters with the Vicon Ground Truth for the second datatset.

C. Dataset 3

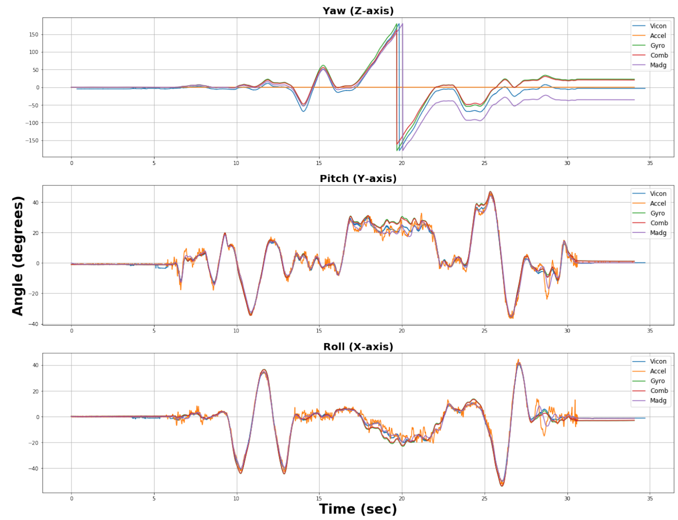

Fig. 3. Comparision of Attitude Estimation using the various filters with the Vicon Ground Truth for the third datatset.

D. Dataset 4

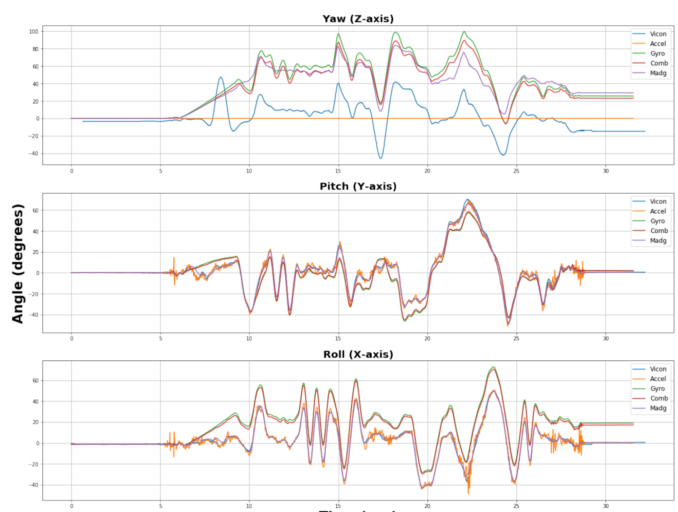

Fig. 4. Comparision of Attitude Estimation using the various filters with the Vicon Ground Truth for the fourth datatset.

E. Dataset 5

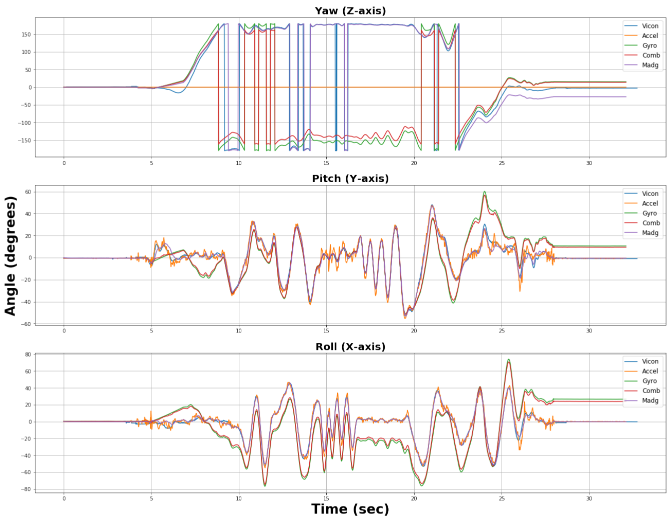

Fig. 5. Comparision of Attitude Estimation using the various filters with the Vicon Ground Truth for the fifth datatset.

F. Dataset 6

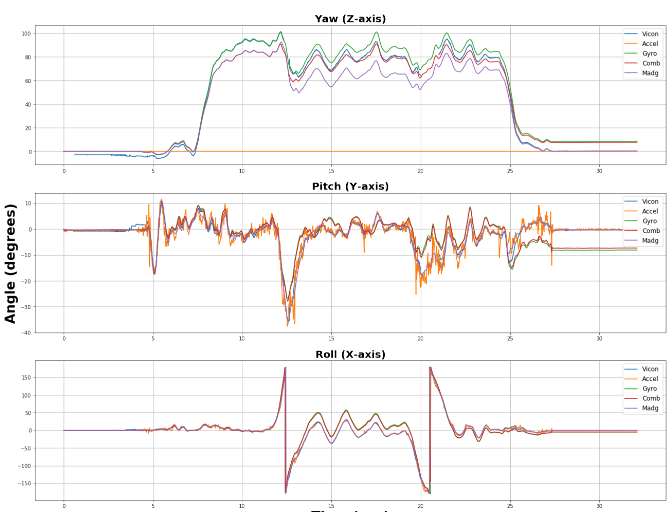

Fig. 6. Comparision of Attitude Estimation using the various filters with the Vicon Ground Truth for the sixth datatset.

G. Dataset 7

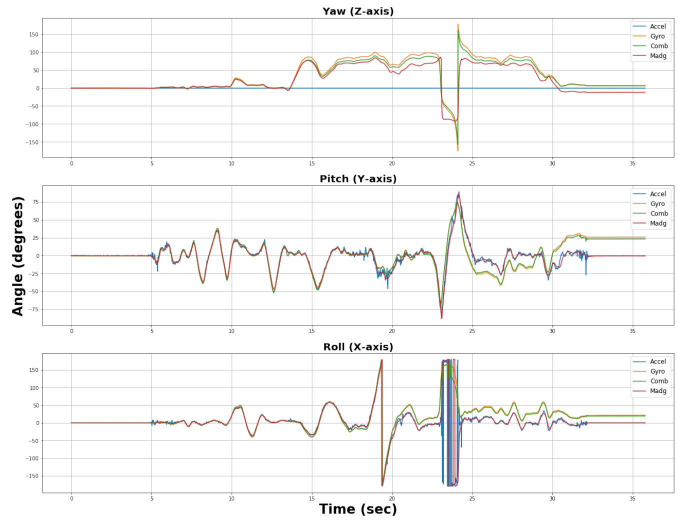

Fig. 7. Comparision of Attitude Estimation using the various filters for the seventh datatset.

H. Dataset 8

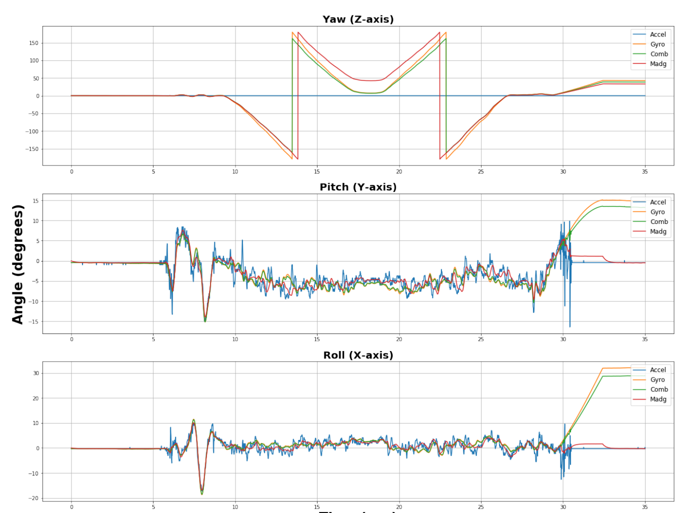

Fig. 8. Comparision of Attitude Estimation using the various filters for the eighth datatset.

I. Dataset 9

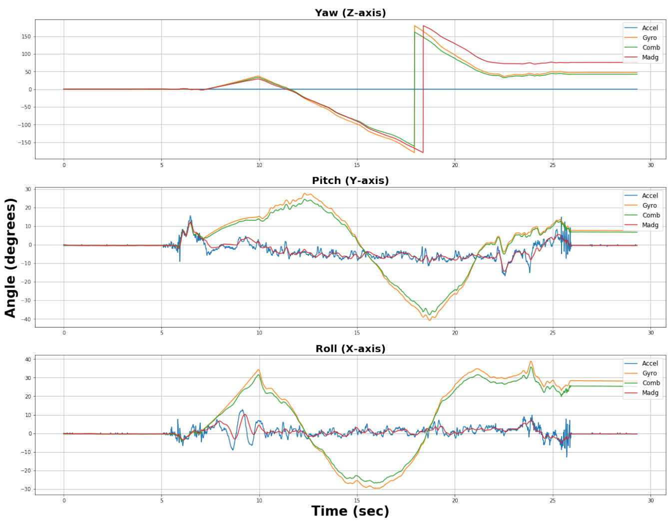

Fig. 9. Comparision of Attitude Estimation using the various filters for the ninth datatset.

J. Dataset 10

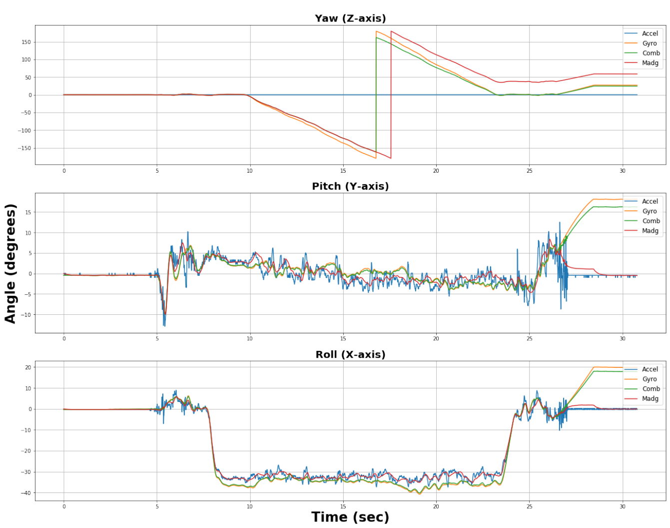

Fig. 10. Comparision of Attitude Estimation using the various filters for the tenth datatset.

IV. Conclusion

The results show that each method of computing attitude provides an estimation of attitude varying in accuracy, with the Madgwick filter providing the best estimate to the vicon truth data. Roll and pitch is best estimated by all of the filters, while yaw estimates suffer due to the accelerometer's inherent inability to measure accelerations about the yaw axis. The gyro attitude estimation was the most prone to drift as integration error accumulated with time. This led to varying performance of the combined and Madgwick filters, particularly in yaw where the accelerometer was unable to provide any estimate of orientation to correct gyro drift. While the Madgwick filter does not provide a perfect attitude estimate, it is still robust in that error from gyro readings does not significantly accumulate and discontinuities in raw data do not prevent the filter from tracking the orientation accurately. Furthermore, the implementation of the Madgwick filter is relatively simple and computationally inexpensive, making it superior to the other methods of attitude estimation studied in this project.

References

[1] Sebastian O.H. Madgwick, *An efficient orientation filter for inertial and inertial/magnetic sensor arrays*