IMU Integration for ATRIAS

by
Johnathan Van Why

A THESIS

submitted to

Oregon State University

University Honors College

in partial fulfillment of
the requirements for the
degree of

Honors Baccalaureate of Science in Mathematics
(Honors Scholar)

Presented February 11, 2016
Commencement June 2016

AN ABSTRACT OF THE THESIS OF

Johnathan Van Why for the degree of Honors Baccalaureate of Science in Mathematics presented on February 11, 2016. Title: IMU Integration for ATRIAS.

Abstract approved:

____

Jonathan Hurst

An Inertial Measurement Unit (IMU) is an important part of a freestanding bipedal robot's state estimation system. IMUs return translational accelerations and rotational velocities, which must be numerically integrated to obtain a robot's orientation. This thesis documents the implementation of the IMU alignment and integration systems for the ATRIAS robot. As a result of this work, ATRIAS has responsive and reliable orientation estimation, which allowed it to achieve freestanding walking.

Key Words: IMU, bipedal locomotion

Corresponding e-mail address: vanwhyj@oregonstate.edu

IMU Integration for ATRIAS

by
Johnathan Van Why

A THESIS

submitted to

Oregon State University

University Honors College

in partial fulfillment of
the requirements for the
degree of

Honors Baccalaureate of Science in Mathematics
(Honors Scholar)

Presented February 11, 2016
Commencement June 2016

Honors Baccalaureate of Science in Mathematics project of Johnathan Van Why presented on February 11, 2016.

APPROVED:

Jonathan Hurst, Mentor, representing MIME

Ross Hatton, Committee Member, representing MIME

Fuxin Li, Committee Member, representing EECS

Toni Doolen, Dean, University Honors College

I understand that my project will become part of the permanent collection of Oregon State University, University Honors College. My signature below authorizes release of my project to any reader upon request.

Johnathan Van Why, Author

Contents

List of Figures

Figure 1: The ATRIAS bipedal robot

1 Introduction

The control of a dynamic bipedal robot requires extensive state feedback, including feedback on the position and velocity of the robot. If the position and velocity of any one link on the robot is known, then position sensors placed throughout the robot may be used to determine the position and velocity of every link on the robot. While this works for many robots – including robots where one or more links have fixed positions – this does not work for freestanding robots. ATRIAS, a dynamic bypedal robot, has no link of a known position and therefore requires another source of position information for control. An Inertial Measurement Unit provides this additional information.

Modern strapdown Inertial Measurement Units contain 3 gyroscopic sensors and 3 accelerometers, giving their angular velocity and translational acceleration at each moment in time. Assuming that the IMU is a perfect sensor and we have perfect knowledge of its initial state, these sensor readings may be integrated to obtain the IMU's current orientation and position at any moment in time. In practice, however, obtaining a precise measurement of the IMU's initial state (a process known as "alignment") and accurately integrating the IMU readings is difficult. As a result, IMU-based orientation and position data tends to "drift", or accumulate error over time. [1]

Work has been done on combining IMU gyroscope readings with readings from other sensors to correct IMU drift. Techniques based on complementarity filters [4] and Kalman filters [2] [6] have been developed to correct orientation errors and remove IMU drift. However, we will argue that using the dynamics for correction is not advisable for the ATRIAS robot.

2 Background

2.1 Robot Orientation Sensing

There are several methods for sensing a bipedal robot's orientation. The goals of the ATRIAS project guided the decision on how to sense the robot's orientation. ATRIAS is designed to operate on rough, non-level terrain, and is therefore unable to use the ground's slope to determine its orientation. Simultaneous Localization and Mapping (SLAM) methods based on visual, LIDAR, or other environmental sensing can be used to track the robot's orientation, but are complicated to implement and rely on relatively unobscured sensing of the environment [9]. In practice, environmental conditions such as dense smoke or water can obscure cameras and LIDAR sensors, rendering SLAM unreliable. Compared to the other orientation estimation options, an IMU is a relatively simple and highly reliable orientation sensor that is not affected by environmental conditions and was therefore selected as the orientation sensor of choice for ATRIAS.

Figure 2: The KVH 1750 IMU used in the ATRIAS robot

2.2 Inertial Measurement Units

Different IMUs return different sets of measurements. Most IMUs contain both accelerometers (which measure translational motion) and gyroscopic sensors (which measure rotational motion). Basic accelerometers provide translational acceleration measurements, while integrating accelerometers contain electromechanical integration mechanisms in order to directly provide velocity measurements. In addition, any accelerometer whose sense axis is not orthogonal to gravitational acceleration will detect a gravity-induced acceleration. Gyroscopic sensors may provide angular velocity signals or (in the case of mechanical gyroscopes) directly provide orientation. Further, mechanical gyroscopes may provide a stable platform for the accelerometers, allowing them to measure world-relative accelerations rather than vehicle-relative accelerations. [1] The RHex robot uses MEMS gyroscopes and distributed MEMS accelerometers to directly provide translational acceleration, angular velocity, and angular acceleration. [3] The KVH 1750 IMU in ATRIAS contains MEMS accelerometers and fiber optic gyroscopes. Each millisecond, the KVH IMU reports sensed translational accelerations (in a reference frame fixed to the IMU) as well as accumulated delta angle readings (integrated outputs from angular rate gyroscopes located in the same reference frame as the accelerometers).

2.3 Error Sources

There is a large amount of material on error sources in IMU measurements and measurements obtained by integrating IMU outputs. For our purposes, the most significant error sources are biases – errors in the form of constant offsets between the sensor readings and the true acceleration and angular velocity. Gyroscope biases cause the orientation estimation error to grow over time. In addition, gravitational acceleration must be removed from accelerometer measurements before being integrated into the robot's velocity, and orientation error causes an error in the gravity correction that leads to a growing acceleration error. This error is then compounded when acceleration is integrated once or twice to obtain velocity or position, respectively. [1] The error buildup present due to the integration of biased sensor readings means that error management and the correct implementation of IMU integration is very important for obtaining accurate state estimation in ATRIAS.

2.4 Error Correction Literature

Much of the literature on robotic IMU drift correction has focused on using gravitational accelerations sensed by the accelerometers to correct the orientation estimate. A simple example is the complementary filter, which combines the low-frequency stability of acceleration-derived orientation estimates with the accurate high-frequency transient estimates derived from gyroscope data to obtain a responsive and stable orientation estimate. [4] Similar research has been done on using Kalman filters to combine the accelerometer and gyroscope readings in a similar fashion. Although more complex, the Kalman filter approach has the advantage of being extensible; it can be modified to use a model of a robot's dynamics to improve the estimation accuracy. [6] However, both approaches make the assumption that the detected gravity vector averages to the world's gravity vector in the presence of orientation error, which may not be true for robotic systems in which orientation is controlled as part of a closed-loop control system.

2.5 Quaternions

A recurring topic in the IMU integration literature is quaternions. The quaternions may be seen as a 4-dimensional extension of the complex numbers. Any quaternion q may be represented as the sum of four parts:

$$q = a + b \cdot i + c \cdot j + d \cdot k$$

where $a, b, c, d \in \mathbb{R}$, $i^2 = j^2 = k^2 = ijk = -1$, and i, j, k are distinct. These definitions give rise to a noncommutative ring structure – quaternions are associative but quaternion multiplication is not commutative. This associativity combined with the definitions of i, j, and k define quaternion multiplication. Additionally, the conjugate of q is defined by:

$$\bar{q} = a - b \cdot i - c \cdot j - d \cdot k$$

[5] The unit quaternion sphere is homomorphic to $SO(3)$ (the three dimensional rotation group), and unit quaternions may be used to represent the orientation of a rigid body. [8]

3 Methods

3.1 Filtering vs Direct Integration

The most important decision in the IMU integration implementation is whether or not a filtering method should be used to correct drift in the orientation estimate. While using a filter has the obvious advantage of eliminating drift, there are several roadblocks to their implementation on ATRIAS.

Complementary filters as well as many Kalman filters for fusing gyroscope and accelerometer data are designed under the assumption that orientation estimation error does not affect the IMU's translational acceleration [4][2]. In ATRIAS, these translational accelerations are a function of the control inputs, which are themselves a function of the estimated orientation. As a result, the stability and convergence of the filter depends not only on the filter's design and implementation, but on the robot's dynamics and the controller as well. Therefore an otherwise stable orientation estimation filter can fail (provide divergent estimates) as a result of a change made to ATRIAS's feedback control software. In the presence of modeling error, model-based methods such as full-body nonlinear Kalman filters suffer from the same controller-dependent stability issue. As the IMU is ATRIAS's only method for sensing orientation, diagnosing an instability in an orientation estimation filter would be difficult, so using a non-model-based orientation filter risks significantly delaying controller development.

In addition, ATRIAS is a hybrid system; the dynamics and number of degrees of freedom of the robot vary as the feet make and break contact with the ground. Further, this contact is difficult to detect and may not be stable; slipping feet leads to unpredictable contact dynamics. This makes it difficult to apply model-based filtering methods to ATRIAS.

Last, the IMU integration software needed to be implemented and tested while ATRIAS was mounted to its planarizing boom. However, the boom significantly alters ATRIAS's dynamics (reducing the number of degrees of freedom), so any model-based filter would need to be implemented for both the ATRIAS-on-boom model and the freestanding ATRIAS model.

For these reasons, it was decided that no bias correction filter would be used; instead, effort was put into making the IMU alignment and integration routines highly

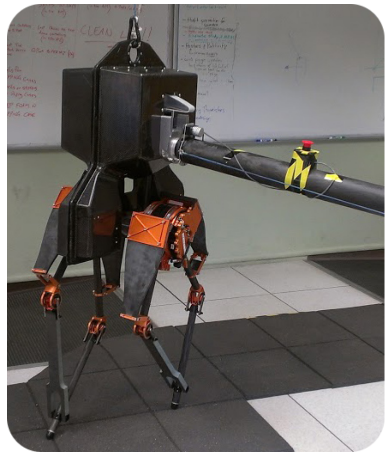

Figure 3: ATRIAS mounted to its planarizing boom

accurate so that alignment error and drift would be negligible.

3.2 Coordinate Systems

As shown in Figure 4, there are three coordinate systems used in the IMU integration system.

The first is the "world" coordinate system. The world coordinate system is defined relative to the Earth's surface near ATRIAS. Its x coordinate points East, its y coordinate points North, and its z coordinate points upwards.

The second is the "ATRIAS" coordinate system, which is fixed to ATRIAS's torso. Its x vector points forward, its y vector points to ATRIAS's left, and its z vector points upwards.

The last is the "IMU" coordinate system, which is defined by the axes of the KVH 1750 sensors. Its x vector points 45° forwards from the downwards direction and its y vector points 45° backwards off the downwards direction. Its z vector points to ATRIAS's left, coinciding with the y vector of the ATRIAS coordinate system.

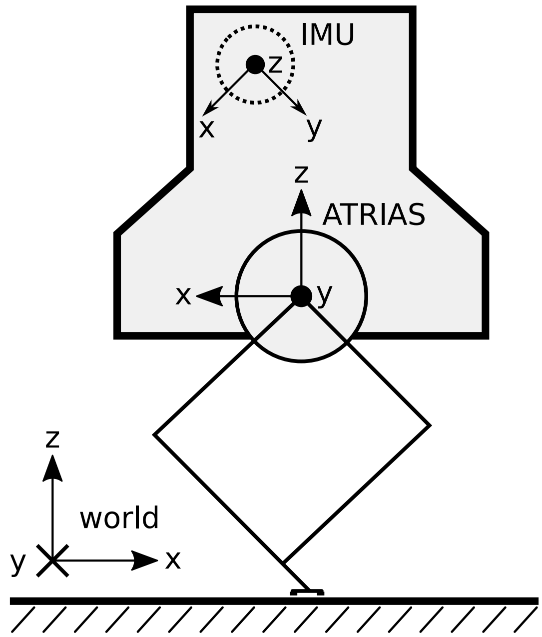

Figure 4: Coordinate systems relevant to IMU orientation integration

3.3 Rotation Quaternions

As mentioned previously, the unit quaternion sphere is homomorphic to $SO(3)$. Although the literature contains two opposing homomorphisms between unit quaternions and rotations, we will use a mapping defined as follows: Given a unit quaternion $q = a+b \cdot i+c \cdot j+d \cdot k$ and vector $v = (x, y, z) \in \mathbb{R}^3 = x \cdot i+y \cdot j+z \cdot k$, the rotation of v by q is $v' = qv\bar{q}$. This mapping is 2-1; each rotation corresponds with exactly two unit quaternions (which are negatives of each other). Further, because this mapping is a group homomorphism between the unit quaternions (as a multiplicative group) and $SO(3)$, quaternion multiplication corresponds to the composition of rotations. In other words, rotating a vector by q_1 then by q_2 is equivalent to rotating the vector by $q_2 q_1$. Last, the conjugate of a unit quaternion represents the inverse rotation of that quaternion. [8]

Quaternions were chosen as the orientation representation for several reasons. Unlike any 3-coordinate representation, quaternions are free from "gimbal lock" (i.e. they do not have any singularities as a representation of $SO(3)$, unlike Euler angles). However, they only have one constraint, that rotation quaternions are unit quaternions (whereas rotation matrices have 6 constraints, 3 of which are normality constraints and 3 of which are orthogonality constraints). [8] Any orientation representation with more than 3 coordinates will suffer from numerical drift due to roundoff errors, but with quaternions it is clear that a simple normalization suffices to restore constraint satisfaction. Last, as we will shortly see, it is very easy to convert IMU readings into rotation quaternions.

Each millisecond, the KVH 1750 IMU on ATRIAS provides a "delta angles" output. This output is the integral of the sensed angular velocity over the previous millisecond. Making the approximation that the change in angular velocity over the integration timestep is small, it has been shown that the direction and magnitude of this delta angles output are equal to the angle-axis representation of the IMU's rotation over this timestep. Given this, we can derive the incremental rotation quaternion from the IMU readings. Letting $\vec{d}$ refer to the vector of delta angles returned by the IMU (expressed as a quaternion), the corresponding rotation quaternion is:

$$d_r = \cos\left(\frac{\|d\|}{2}\right) + \sin\left(\frac{\|d\|}{2}\right) \cdot \frac{d}{\|d\|} \tag{1}$$

[7]

3.4 Alignment

At startup, the IMU integration software needs to determine the robot's orientation, a process known as alignment. The alignment routine for ATRIAS is based on gyrocompass alignment routines for gimballed inertial navigation systems. [1] The alignment consists of a period of data gathering (accumulation of accelerometer readings) followed by a computation which determines the robot's orientation as well as the Earth rotation correction vector. During the data gathering step, the accelerometer readings are integrated to obtain a vector that points opposite of the direction of gravitational acceleration in the IMU's local reference frame. ATRIAS is kept stationary during alignment and faces a known heading. The computation then consists of a few steps:

1. Use the sensed gravity vector to "level" the orientation estimate

2. Rotate the orientation estimate to match the known alignment heading

3. Compute the Earth's angular velocity vector using the known alignment latitude

In describing the alignment calculations, I'll let $\vec{g}_m$ refer to the measured acceleration from the data gathering step and q refer to the current orientation quaternion estimate. The alignment routine is designed to correct an initial, incorrect, estimate of ATRIAS's orientation.

To "level" the orientation estimate, we first rotate the accumulated acceleration vector into world coordinates then normalize it:

$$\hat{g} = q\vec{g}_m\bar{q}$$

$$\vec{u}_g = \frac{\hat{g}}{\hat{g}}$$

If the initial orientation was correct, $\vec{u}_g = (0,0,1)$. Any deviation in this represents an error in the orientation estimate. To correct this, we need to rotate $\vec{u}_g$ onto $(0,0,1)$. We rotate about an axis orthogonal to both $\vec{u}_g$ and $(0,0,1)$:

$$\vec{a} = \vec{u}_g \times (0,0,1)$$

Note that the choice of order in the cross product means that the correction rotation should be a positive rotation about axis $\vec{a}$.

Since $\vec{u}_g \quad = \quad (0,0,1) \quad = 1$:

$$\vec{a} \quad = \sin(\theta)$$

where θ is the necessary rotation angle for the correction.

Assuming the robot is aligned while upright, θ should lie in $[0, \pi/2)$. We can check this by verifying the following condition:

$$\vec{u}_g \cdot (0,0,1) > 0$$

We may then compute θ using

$$\theta = \arcsin(\ \vec{a} \ )$$

and normalize $\vec{a}$, giving us the angle-axis representation of the rotation quaternion for the correction. We then use Equation 1 to convert rotation vector $\theta \cdot \vec{a}$ into a rotation quaternion r. The final leveling computation is then:

$$\hat{q} = rq$$

To rotate the orientation estimate to match the known alignment heading without invalidating the leveling step of aligment, we need to rotate the orientation estimate about the world Z vector. To compute this rotation, we rotate the IMU's Z axis, which points towards ATRIAS's left side, into the world frame:

$$p = qk\bar{q}$$

then use trigonometry to determine the necessary rotation to correct the orientation estimate:

$$\theta = \text{atan2}(p_y, -p_x) - h$$

where h is the known heading at alignment.

Again, we use Equation 1 to convert rotation vector θk into a rotation quaternion r then apply the update using:

$$\hat{q} = rq$$

The last step in the alignment process is to compute the effect of the Earth's rotation so it may be cancelled out during integration. The Earth's rotation vector can be computed solely using the Earth's latitude and rotation rate. Letting l refer to the current latitude of ATRIAS and ω_e to the Earth's rotation speed, the Earth's angular velocity is:

$$\vec{\omega}_e = \omega_e \cdot \begin{bmatrix} 0 \\ \cos(l) \\ \sin(l) \end{bmatrix}$$

3.5 Integration

While ATRIAS is operating, after the alignment process is complete, we need to consistently update the orientation quaternion (q). To do this, we use the following sequence of steps:

1. Rotate the IMU's delta angles values (d) into the world reference frame:

$$\hat{d} = q \cdot d \cdot \bar{q}$$

2. Remove the Earth's rotation from the world-relative delta angles value:

$$\hat{d} := \hat{d} - \omega_e \cdot \Delta t$$

3. Compute the quaternion representing the rotation delta $\hat{d}$ using Equation 1, call this quaternion d_r

4. Use quaternion multiplication to concatenate this rotation with all previous rotations, yielding the new orientation quaternion:

$$q := d_r \cdot q$$

3.6 Conversion

Using $q = a + b \cdot i + c \cdot j + d \cdot k$, we can convert q into its equivalent rotation matrix using the formula:

$$\begin{bmatrix} a^2 + b^2 - c^2 - d^2 & 2(-ad + bc) & 2(ac + bd) \\ 2(ad + bc) & a^2 - b^2 + c^2 - d^2 & 2(cd - ab) \\ 2(bd - ac) & 2(ab + cd) & a^2 - b^2 - c^2 + d^2 \end{bmatrix}$$

[8]. This rotation matrix may then be used to compute more convenient representations of ATRIAS's orientation, such as Euler angles.

4 Results

In order to evaluate the accuracy of the IMU-based orientation tracking for ATRIAS, two experiments were run. Both experiments were conducted while ATRIAS was connected to its planarizing boom, which provides orientation measurements. For modularity and evaluation purposes, the IMU orientation was expressed using the same coordinate system as ATRIAS's boom so that the values from the two measurement systems are directly comparable.

For the first test, ATRIAS's walking controller was fed position and velocity signals from the boom while the IMU was simultaneously used to derive the same values. The position data from this experiment is visible in Figure 5, and the velocity data is visible in Figure 6. Note that no yaw data is available from the boom as that encoder was disconnected during the tests. Also, at the end of the test ATRIAS trips and falls on an obstacle. Due to a poor electrical connection, this fall shut down the IMU and the IMU integration software held the last known values from the IMU after the fall.

In general, the boom and IMU-derived values are very similar, but there are a couple of interesting features. Most importantly, there is no visible drift in the IMU's measurements over the 5-minute test duration. However, a noticeable sinusoidal variation may be seen in the relative values of the pitch and roll measurements; the boom and IMU values have a difference that varies as ATRIAS moves around the room. This variation may be caused by IMU alignment error or by an error in the boom's construction, installation, or calibration. Due to the small magnitude of the variation, it was deemed insignificant for control purposes and ATRIAS's controller was reconfigured to use IMU-derived orientation data for a second test.

For the second test, ATRIAS walked for over 6 minutes using the same controller as for the boom-based tests. The position data from this experiment is visible in Figure 7, and the velocity data is visible in Figure 8. Again, no boom-derived yaw data is available during this test.

Again, the IMU data stayed near to the boom data and no significant drift was observed in the IMU measurements. Further, no noticeable degredation in controller performance or stability was observed, further indicating that drift during the experiment was negligible.

Figures 9 and 10 give zoomed-in views of the IMU and boom data for several consecutive steps during this experiment. These figures demonstrate that the boom and IMU-derived orientation data closely match each other on shorter timescales.

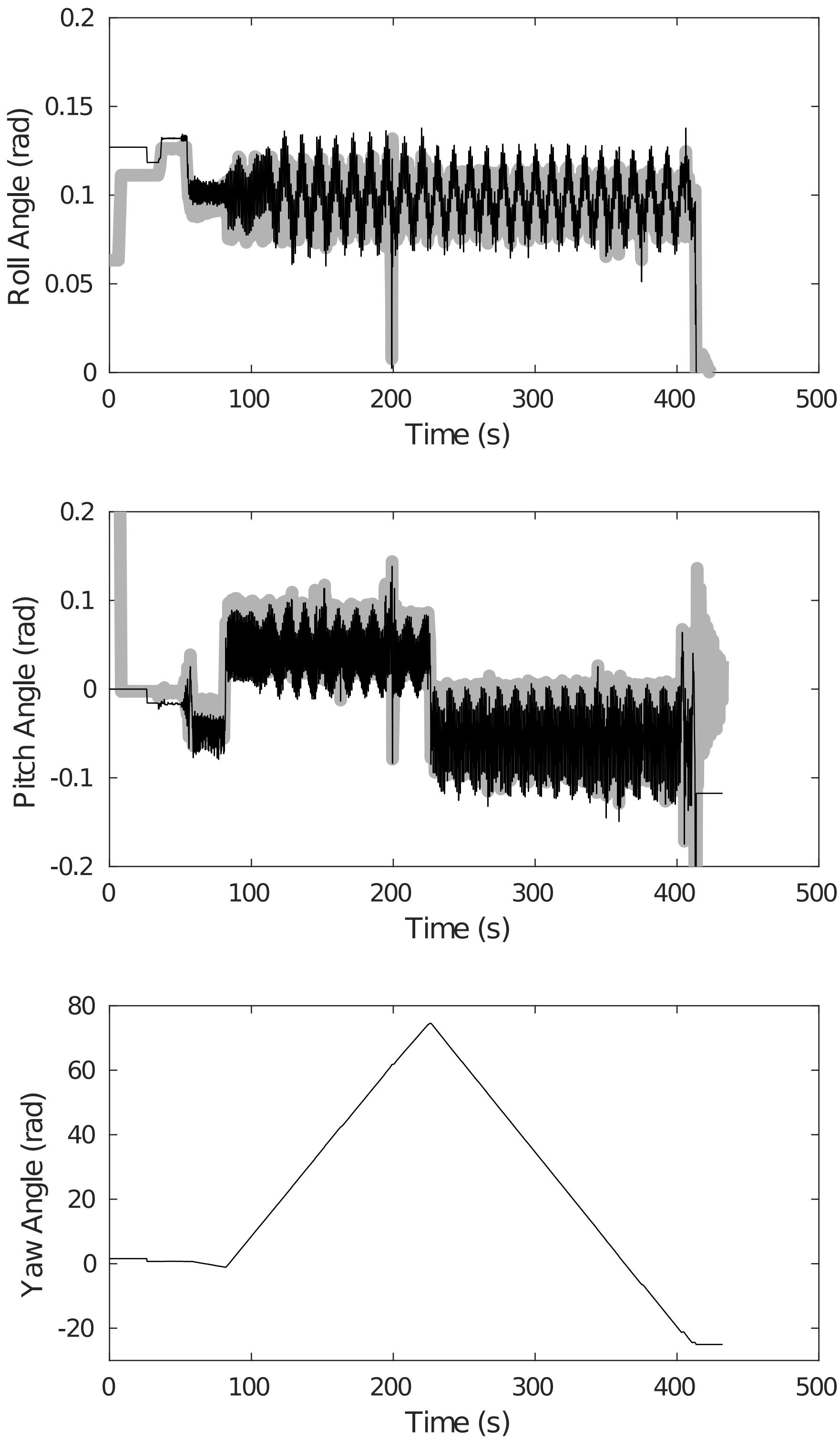

Figure 5: Orientation values from ATRIAS's boom and IMU while walking using boom data, demonstrating accuracy over a period of time.

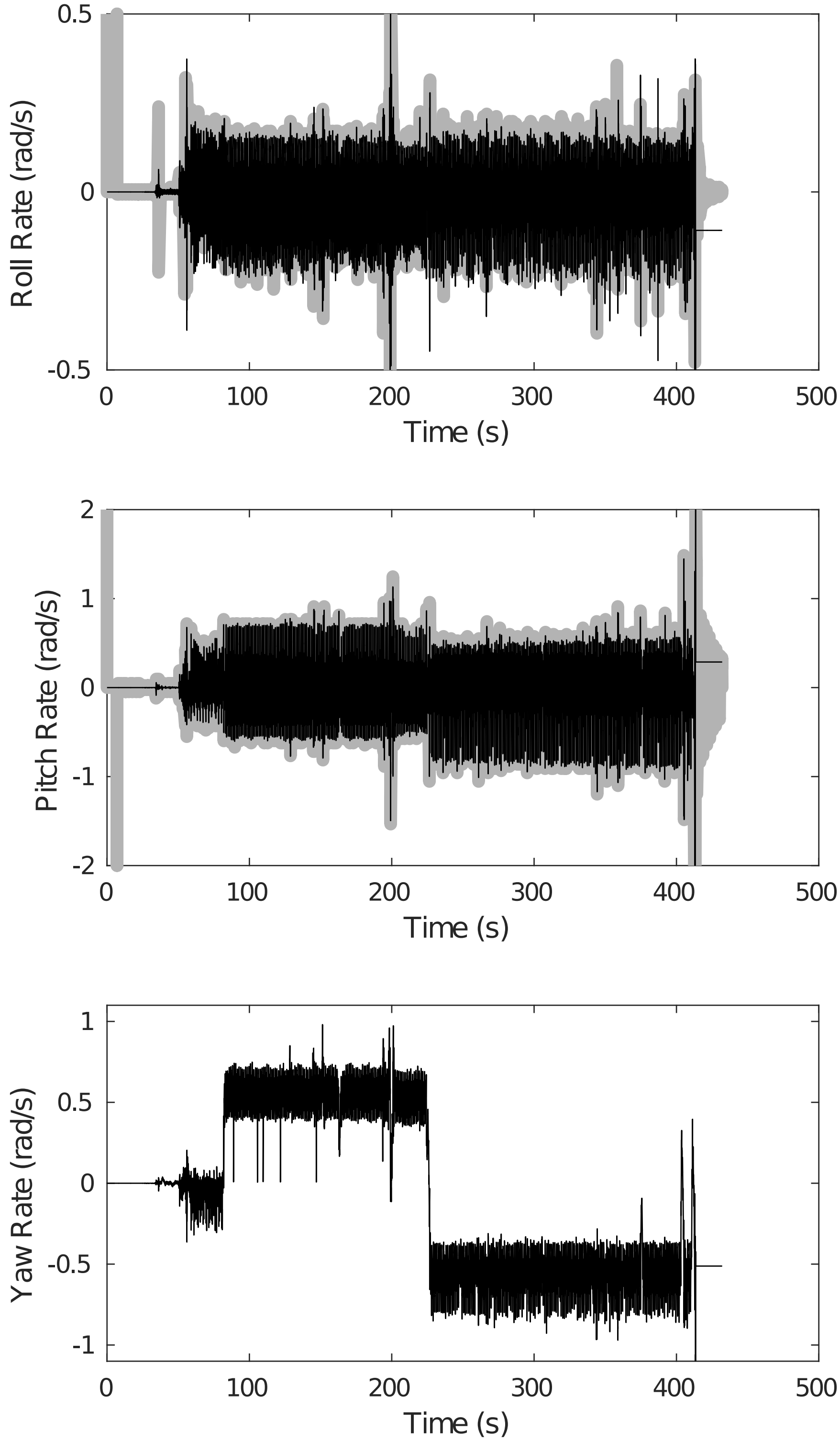

Figure 6: Orientation velocities from ATRIAS's boom and IMU while walking using boom data, again demonstrating accuracy over a period of time.

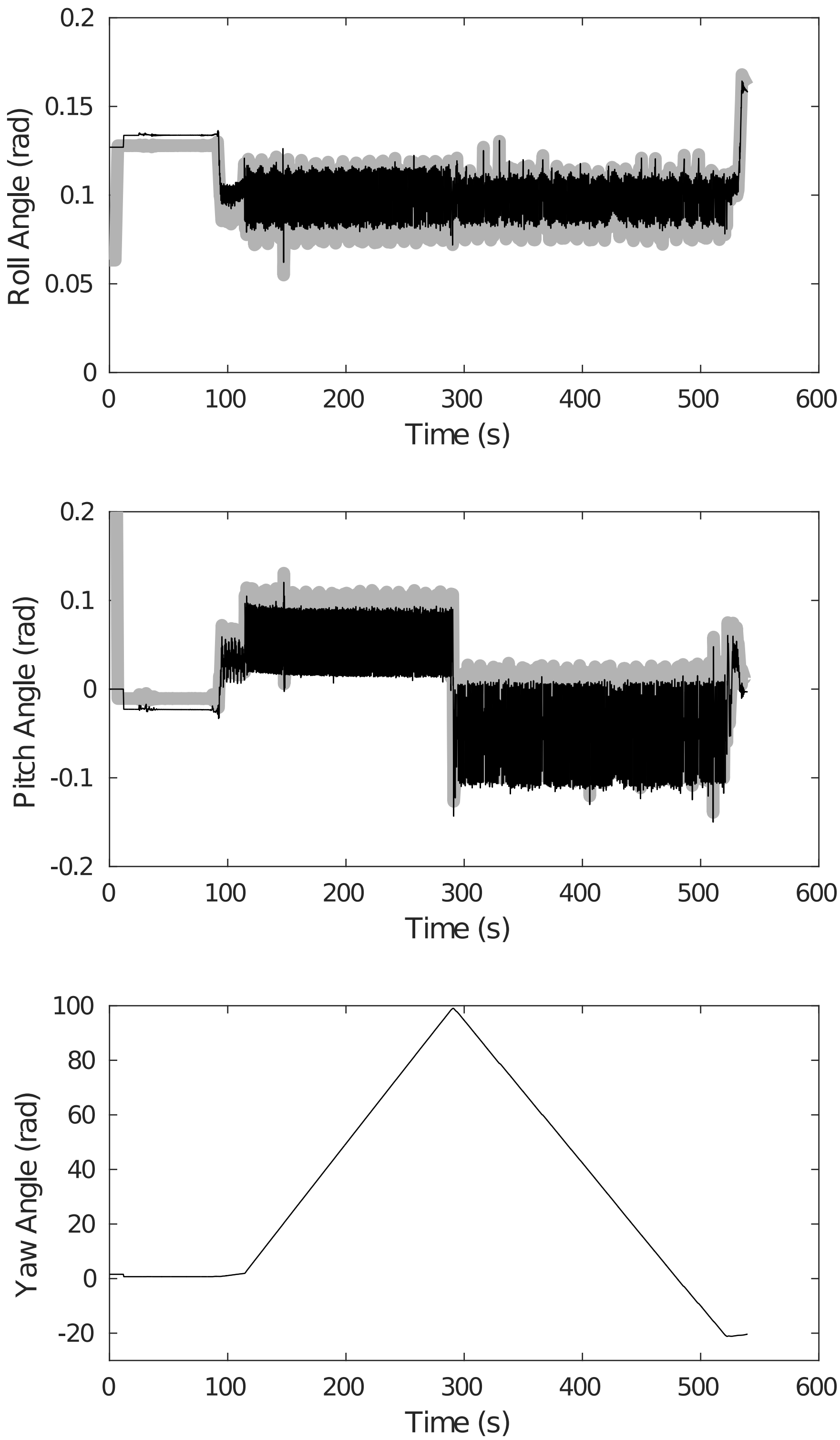

Figure 7: Orientation values from ATRIAS's boom and IMU while walking using IMU data, proving that the IMU orientation data is usable for walking.

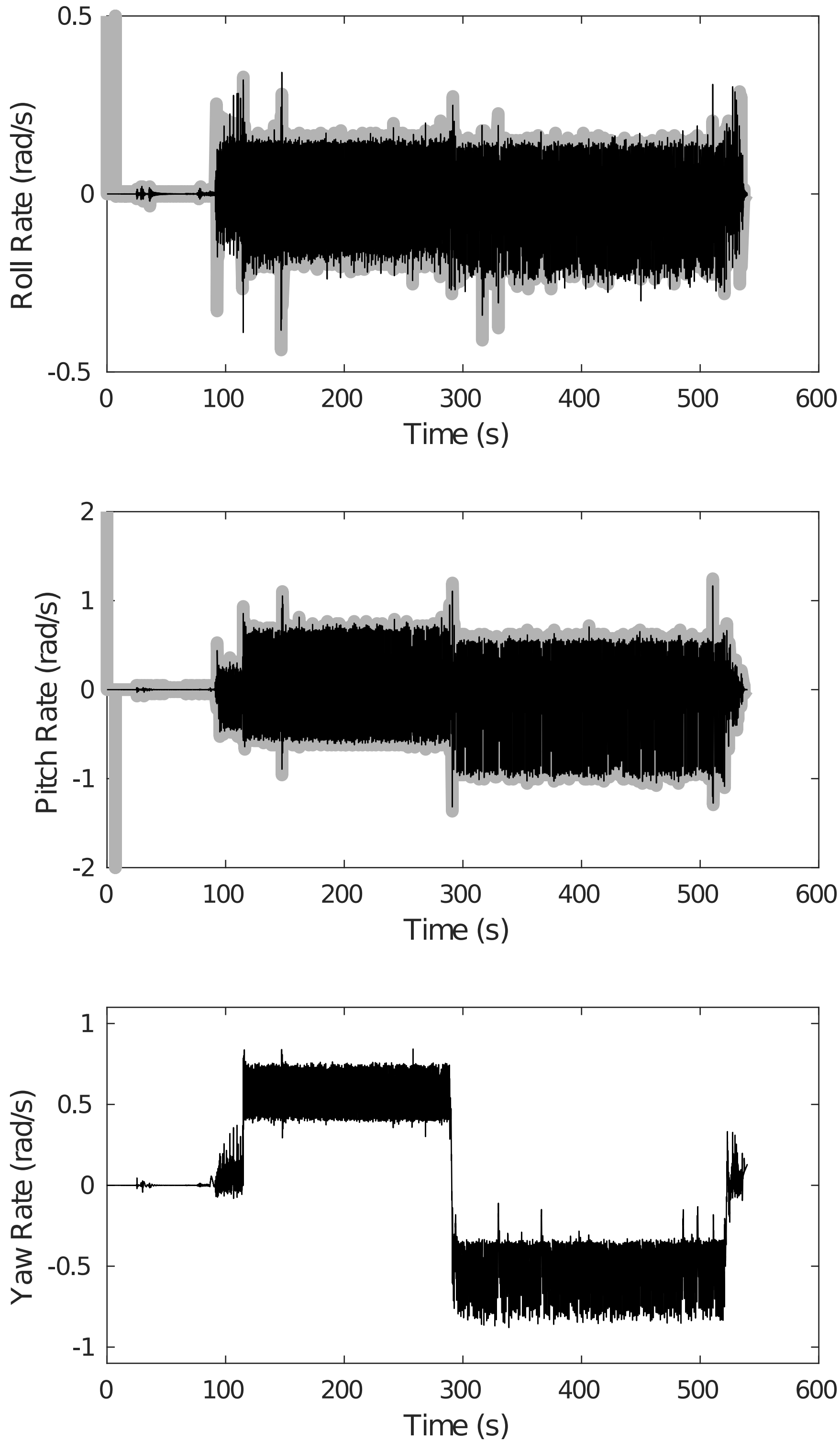

Figure 8: Orientation velocities from ATRIAS's boom and IMU while walking using IMU data

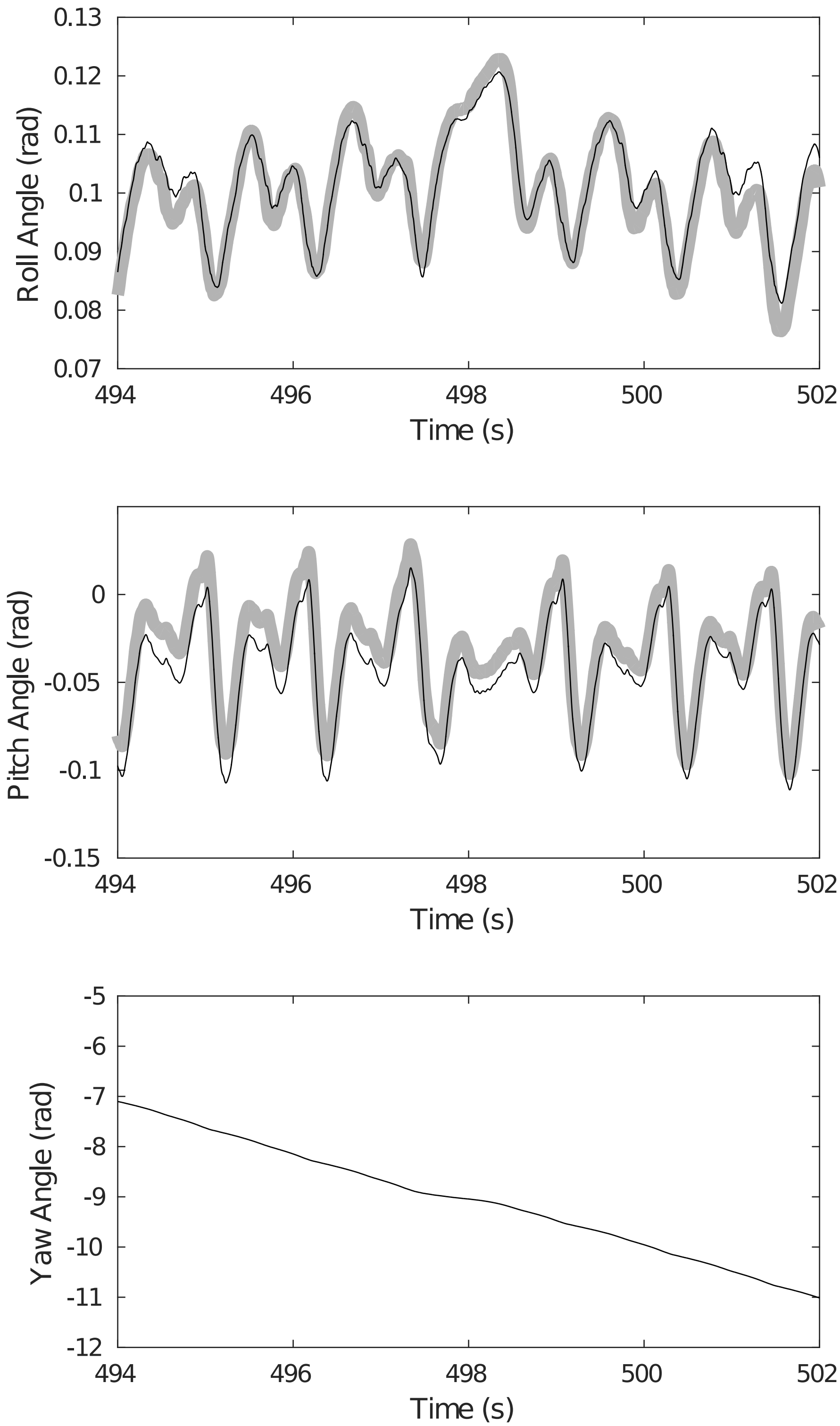

Figure 9: A closer view of the Orientation values from ATRIAS's boom and IMU while walking using IMU data. The IMU data closely represents the motion of the ATRIAS robot, though there is an offset between the boom data and the IMU data.

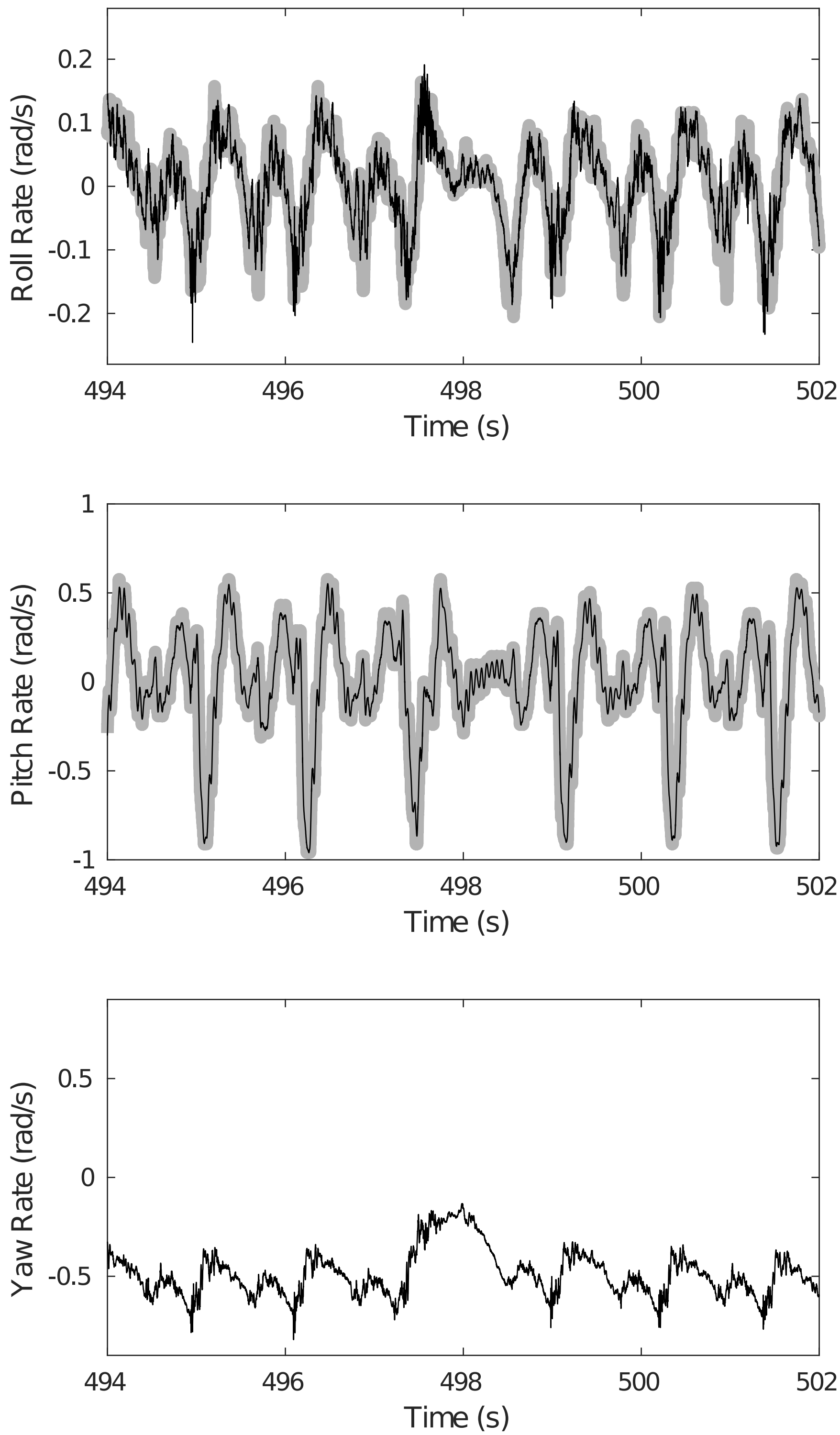

Figure 10: A closer view of the Orientation velocities from ATRIAS's boom and IMU while walking using IMU data. The velocity signals do not have the same offset as the position signals, and are a much more responsive measurement of ATRIAS's motions than the boom encoder readings.

The last relevant experimental results is a long-duration standing test in 3D (i.e. with ATRIAS freestanding rather than mounted on the boom). ATRIAS "stood" (dynamically stood in place while continuously switching between its left and right foot) for over 49 consecutive minutes before falling. Therefore, it is known that the IMU drift rate is sufficiently low for ATRIAS to remain stable for at least 49 minutes, which was considered an acceptably low drift rate.

5 Conclusions

The IMU implementation for ATRIAS gave reliable orientation sensing. In all cases, the integration performed acceptably and allowed the goals of the ATRIAS project to be achieved.

Looking forward, there are a few ideas that would be worth trying. As described in Section 3, several methods for drift elimination were excluded because they would have an uncertain effect on controller stability or because they could not be fully tested while ATRIAS was mounted to its boom. However, we now have walking controllers that are known to be stable if accurate IMU data is available, and filtering strategies for eliminating drift may be evaluated using the current controllers. Additionally, the standing controllers give us the ability to test the filter designs while ATRIAS is off the boom. Investigating filter design may allow for ATRIAS (or another similar robot) to operate for longer without re-aligning, operate on a less accurate (faster) alignment, or utilize a lower-cost IMU with larger biases.

References

[1] Mohinder S Grewal, Lawrence R Weill, and Angus P Andrews. *Global positioning systems, inertial navigation, and integration.* John Wiley & Sons, 2007.

[2] Edgar Kraft. A quaternion-based unscented kalman filter for orientation tracking. In *Proceedings of the Sixth International Conference of Information Fusion*, volume 1, pages 47–54, 2003.

[3] Pei-Chun Lin, Haldun Komsuoglu, and Daniel E Koditschek. Sensor data fusion for body state estimation in a hexapod robot with dynamical gaits. *Robotics, IEEE Transactions on*, 22(5):932–943, 2006.

[4] Robert Mahony, Tarek Hamel, and Jean-Michel Pflimlin. Nonlinear complementary filters on the special orthogonal group. *Automatic Control, IEEE Transactions on*, 53(5):1203–1218, 2008.

[5] Alfred C Robinson. On the use of quaternions in simulation of rigid-body motion. Technical report, DTIC Document, 1958.

[6] S Šlajpah, R Kamnik, and M Munih. Kinematics based sensory fusion for wearable motion assessment in human walking. *Computer methods and programs in biomedicine*, 116(2):131–144, 2014.

[7] Sara Stančin and Sašo Tomažič. Angle estimation of simultaneous orthogonal rotations from 3d gyroscope measurements. *Sensors*, 11(9):8536–8549, 2011.

[8] John Stuelpnagel. On the parametrization of the three-dimensional rotation group. *SIAM review*, 6(4):422–430, 1964.

[9] Sebastian Thrun and John J Leonard. Simultaneous localization and mapping. In *Springer handbook of robotics*, pages 871–889. Springer, 2008.