Gyroscope Drift Correction Algorithm for Inertial Measurement Unit Applications in Robotics

Nonnarit O-larnnithipong, Sudarat Tangnimitchok, Armando Barreto

Department of Electrical and Computer Engineering, Florida International University
10555 W Flagler Street, Engineering Center 3900
Miami, Florida

ABSTRACT

In this paper, we propose an algorithm to correct the gyroscope drift in Inertial Measurement Units used in robotics applications. The drift in orientation measurement is caused by the accumulation of the bias offset error in the gyroscope reading. The algorithm consists of two parts, which are: bias offset estimation and quaternion correction using gravity vector. The bias offset estimation is performed during periods when the sensor is estimated to be static, when the gyroscope reading would provide only the bias offset error for prediction. The quaternion was calculated based on unbiased angular velocity and then used to rotate the gravity vector in the Earth’s frame resulting in the calculated gravity vector in the sensor’s frame. The error of the angle between calculated and measured gravity vector measured from the accelerometer is used to calculate the difference in quaternion to correct the previously obtained quaternion result. The result of the orientation estimation using this algorithm can be used to describe the orientation of the IMU module with less drift and improved orientation accuracy than without using gravity vector correction.

Keywords

Inertial Measurement Unit, Gyroscope Drift, Drift Correction Algorithm, Bias Offset Error Estimation, Quaternion Correction using Gravity Vector.

1. INTRODUCTION

Nowadays, Inertial Measurement Units or IMUs play key roles in many technologies such as smartphones, wearable devices and fitness tracking devices. An IMU consists of a group of sensing units that could determine the movement, position, magnetic field direction and orientation. IMUs are also popular in many applications in the field of robotics. For example, autonomous land vehicles, unmanned vehicles and navigation system [2], [3], [8]. Most of the applications in robotics use IMUs to determine the position and orientation of any working modules. Some applications replace the position tracking with optical sensor or GPS, which provide more accurate results than using the accelerometers in the IMU because position tracking requires double integration of acceleration readings and that produces large amounts of error even with small levels of sensor error.

IMUs contain gyroscopes, which provide angular velocity measurements that can be used to track orientation in robotics applications. However, orientation tracking also requires the mathematical integration of angular velocity readings from the gyroscopes. Most low-cost gyroscopes may output an erroneous non-zero measurement level even when they are not moving, called the bias offset error. This bias offset error can produce an orientation tracking error called drift, which causes many problems in navigation and other applications that use low-cost gyroscope to determine the orientation of moving parts [8].

Several studies and algorithms have been proposed to eliminate gyroscope drift error in inertial measurement systems. Most of the studies use Kalman-based processes to solve the error in inertial sensors [6], [9], [10]. But this approach is complex and can be complicated to implement [5]. To estimate the orientation, many studies use the idea of sensor fusion which is the method that combines more than one sensor to determine one measurement. For example, a sensor fusion approach may suggest using gyroscopes, accelerometers and magnetometers to determine orientation [1], [4]. It is particularly attractive to make use of sensors contained in a single Inertial Measurement Unit. IMUs then become a good choice to be used to determine the orientation in many applications.

The idea of sensor fusion is also the main method to determine the orientation in this work. We propose an algorithm to correct gyroscope drift by using the combination of gyroscope and accelerometer measurements. In order to combine two sensor’s measurements together, we need to understand the relationship between angular velocity and acceleration. The Earth’s gravity is the quantity that can be measured by accelerometer when the IMU is static. Even though we know that gravity is always pointing down to the ground, in the sensor frame of coordinates, the gravity is measured and distributed into three components in the sensor’s orthogonal axes while the sensor is in an oblique orientation. Therefore, the gravity vector measured from the accelerometer can be related to the measurement of the angle or the inclination of the sensor module. The algorithm in this work will use this main concept to correct the gyroscope reading and improve the performance in estimating the actual orientation of the sensor module. The algorithm consists of two parts: bias offset estimation, using simple linear regression model, and quaternion correction using the comparison with the gravity vector.
2. METHODOLOGY

2.1 Proposed algorithm

When the IMU is static, the gyroscope data, which is the measurement of angular velocity, should ideally provide the reading values of zero. But in reality, each static gyroscope can incorrectly generate a reading that deviates from zero, called the bias offset error. This type of error is the major cause that produces the drift, the significant error of the orientation measurement. The drift is the phenomenon that takes place when the bias offset error and noise in gyroscope measurements are accumulated by means of integration through time and yield unacceptable orientation results.

2.1.1 Bias offset estimation

To minimize the drift, an algorithm to correct the gyroscope bias offset has been proposed. With the idea that the IMU sensor should provide the reading of zero when there is no input or movement taking place, the bias offset error could be determined and used to subtract from future gyroscope readings, to obtain unbiased gyroscope data as a result. The predicted bias offset error (\( \hat{b} \)) of each gyroscope axis was determined during the sensor’s static periods by using a simple linear regression model \([7]\) as shown in equation (1). The model coefficients (\( \beta_0 \) and \( \beta_1 \)) can be found by using equations (2) and (3), where \( b_i \) and \( t_i \) are the measured gyroscope bias and time at \( i \)-th sample, respectively.

\[
\hat{b} = \beta_0 + \beta_1 t (1)
\]

\[
\beta_1 = \frac{\sum_{i=1}^{n}(t_i - \bar{t})(b_i - \bar{b})}{\sum_{i=1}^{n}(t_i - \bar{t})^2} (2)
\]

\[
\beta_0 = \bar{b} - \beta_1 \bar{t} (3)
\]

The predicted bias is subtracted from the measured gyroscope reading, yielding the unbiased angular rate (\( \omega_B \)). To avoid gimbal lock problems, the orientation representation used throughout this paper will be using quaternion notation.

The quaternion rate (\( \dot{q} \)) can be calculated by using the unbiased angular rate as in equation (4), where \( \dot{q}_0 \) is the previous quaternion estimation of the orientation.

\[
\dot{q} = \frac{1}{2} \dot{q}_0 \otimes \omega_B (4)
\]

The quaternion representation of the orientation (\( q \)) can be obtained by integrating the quaternion rate using equation (5), where \( \Delta t \) is the sampling period of the reading data. This quaternion (\( q \)) can be used to rotate points or vectors in three-dimensional space. In other words, it can be used to describe the orientation of the sensor module or the object that the IMU is attached to. To justify the orientation estimation using this method, we also proposed the method to compare this quaternion result with another referencing measurement from the accelerometer integrated in the same IMU.

\[
q = \exp((\Delta t)\dot{q}_0 \otimes \dot{q}_0) \otimes \dot{q}_0 (5)
\]

2.1.2 Quaternion correction using gravity vector

Each of the three accelerometers in the IMU provides the value of acceleration applied to the sensor module in its sensing direction. When the sensor is static, the accelerometers are assumed to measure only the acceleration due to gravity, which is always directed towards the Earth’s center and has only a vertical component in the Earth frame. When the IMU module is in an oblique orientation (not at right angle with the horizon), the acceleration due to gravity is decomposed into each orthogonal axis in the sensor frame. These components can be represented as the inclination of the sensor module in the Earth frame. Accordingly, we can use the quaternion (\( q \)) to rotate the referencing gravity vector in the Earth frame (\( m \)) to obtain the gravity vector in sensor frame as shown in equation (6). This calculated gravity vector (\( \vec{y}(q) \)) will further be used to compare with the measured gravity vector from the accelerometer readings.

\[
\vec{y}(q) = q^* \otimes m \otimes q (6)
\]

If the estimated orientation of the IMU, represented by the quaternion \( q \) had no error, \( \vec{y}(q) \) from eq. (6) would match the measured gravity vector from accelerometer readings. Otherwise, the error between these two vectors can be used to correct the orientation estimation by calculating the angular difference between these two vectors. The difference in angle of the calculated gravity vector (\( \vec{y}(q) \)) and measured gravity vector from accelerometer readings (\( \vec{y}_0 \)) can be calculated and then represented as a rotation in quaternion form, as in equation (7).

\[
\Delta q = q \| q \| (7)
\]

![Figure 1. Block diagram of the proposed drift correction algorithm](image-url)
Then, the estimated quaternion ($\hat{q}$) is calculated by multiplying the quaternion from equation (5) with the difference in quaternion ($\Delta q$) between the calculated gravity vector and the measured gravity vector from accelerometer readings, from equation (7). The estimated quaternion ($\hat{q}$) is described as in equation (10).

$$\Delta q = \mathcal{H}(\hat{q}_v, q_w)$$  
(7)

$$\hat{q}_v = \hat{y}_0 \times \hat{y}(q)$$  
(8)

$$q_w = ||\hat{y}_0|| ||\hat{y}(q)|| + \hat{y}_0 \cdot \hat{y}(q)$$  
(9)

$$\hat{q} = q \otimes \Delta q$$  
(10)

This resulting estimated quaternion ($\hat{q}$) represents the orientation of the sensor module and will be used to calculate the quaternion for the next iteration. Note that the quaternion result has to be normalized before proceeding with any calculations. This algorithm to estimate the orientation in quaternion form is also visually described in the block diagram shown in Figure 1.

### 2.2 Implementation

In this work, we used the YEI 3-Space Sensor module, from YEI Corporation, which is an ultra-miniature, high-precision, high-reliability, low-cost Inertial Measurement Unit (IMU). The module has the dimensions of 23mm x 23mm x 2.2mm (0.9 x 0.9 x 0.086 in.) and consists of tri-axial gyroscope, accelerometer, and magnetometer. The coordinate axes of the sensor’s frame are superimposed on the image of the sensor’s module shown in Figure 2. For now, we are not using the magnetometer as a part of sensor fusion algorithm because the measurements from the magnetometer can be easily disrupted by ferromagnetic materials in the testing environment.

![Image of YEI 3-Space Embedded IMU module](image)

**Figure 2.** Coordinate axes superimposed on the image of YEI 3-Space Embedded IMU module

The key element to effectively use the drift compensation algorithm of the orientation in a robotics application is to be capable of implementing the algorithm in real-time. For this work, we were streaming the angular velocity and acceleration measurement from the IMU and stored them in a text file with a sampling rate of 260 samples per second. The data was imported into the MATLAB workspace, and the algorithm was also implemented using MATLAB. In order to mimic the real-time situation, each iteration of the algorithm was aware of only one sample of data at a time. The reason of implementing the algorithm in MATLAB is to easily evaluate the result obtained from the algorithm. The result of the estimated orientation in quaternion form was also exported to a text file in order to be visually evaluated using OpenGL 3D visualization.

As shown in Figure 3, the implementation of the algorithm had begun by calculating the maximum value in three axis of the 50-sample window average of gyroscope data (gyroMaxAvg). This value is the condition that was used to consider the movement of the sensor module. If this value is less than a pre-defined threshold value (gyroThreshold), the module is considered to be not moving. To verify the static period of the sensor, this condition has to be true for 25 consecutive samples of gyroscope data and then the new predicted bias error will be calculated. Otherwise, the unbiased angular rate will be determined by using the previous predicted bias error. The quaternion was then calculated based on the unbiased angular rate.

![Flowchart showing the implementation of drift correction algorithm for one iteration with the condition that the module should be in a static period (Stillness)](flowchart)

**Figure 3.** Flowchart showing the implementation of drift correction algorithm for one iteration with the condition that the module should be in a static period (Stillness)
The second part of the algorithm, which is the quaternion correction using gravity vector, used a different value to determine the sensor’s static period called stillness. Stillness is the value that reflects the movement of the sensor module, which can be varied from 0 to 1. A high value of Stillness indicates that the sensor is in a static condition and thus, the accelerometer measurements will only or almost only contain the components of acceleration due to gravity. The Stillness value is a special feature provided by this IMU, which is more sensitive to acceleration measurements than angular velocity measurement.

In our algorithm, if the value of Stillness is greater than pre-defined threshold (stillnessThreshold), the calculated gravity vector using the quaternion that represents the current orientation will be determined, and the difference in quaternion (Δq) with the measured gravity vector from accelerometer will be determined.

In the sensor’s static period, where the Stillness is greater than stillnessThreshold, the estimated orientation will be updated by the accumulation of Δq to the pre-calculated quaternion (q) by means of quaternion multiplication. Otherwise, the estimated orientation will be equal to the pre-calculated quaternion (q).

The YEI 3-Space Embedded IMU module was connected to a PC host via USB connection. A C program was written to stream the data from the sensor and stored in a text file.

The experiment was performed by rotating the sensor module in different axes in order to verify the validity for any rotating movements. While the sensor module was rotating, angular velocity and acceleration were recorded with a sampling rate of 260 samples per second. The recorded data was processed by the proposed algorithm, following the flowchart in Figure 3.

3. RESULTS

The gyroscope and acceleration data recorded from the experiment have the data length of 17,267 samples from the recording duration of 66.33 seconds.

The first two plots in Figure 4 are the raw gyroscope data recording in units of radian per second and the predicted bias errors for three coordinate axes (x, y and z) of the gyroscope. The third plot in Figure 4 is the result of computed quaternion using unbiased angular rate, this computed quaternion is the result of the orientation representation of the sensor module before applying the correction using gravity vector.

The measured gravity vector from the accelerometer is shown in the first plot of Figure 5; its vertical axis is the strength of gravity components (in units of g) in each coordinate axis of the sensor frame. The second plot in Figure 5 shows the difference of the measured gravity vector and the computed gravity vector using the computed quaternion.

Figure 4. The plots of raw gyroscope data including bias offset error, predicted bias error, quaternion result (without gravity-vector correction)

Figure 5. The plots of measured gravity vector from accelerometer, error between measured and computed gravity vector, estimated quaternion result (with gravity-vector correction)
The resulting estimated orientation in the form of quaternion after gravity-vector correction is shown in the third plot of Figure 5. It consists of the four components of the quaternion, which can be used to describe the orientation of the sensor module. The horizontal axis in Figure 4 and Figure 5 is the time in seconds.

4. VERIFICATION
In this section, the result in section 3 will be verified through 3D visualization. The quaternion results both before and after gravity vector correction were also exported as text files and used in 3D visualization. Then they were verified and compared with the top view of the actual sensor movement as shown in Figure 6.

Figure 6(a) shows the sequence of 9 actual orientations of the sensor module captured during the recording. Figure 6(b) is the 3D visualization using the orientation quaternion before gravity-vector correction.

Figure 6(c) shows the estimated orientation sequence of the sensor module rotation using the estimated quaternion after gravity-vector correction. The 9-stage sequences of orientation in Figure 6 (b) and (c) are the visualization of the results in the third plots of Figures 4 and 5, respectively. The numbers (1-9) to the left of the stages in Figure 6 correspond with the (1-9) intervals labelled at the bottom of Figure 4 and Figure 5.

5. DISCUSSION
The results obtained by predicting the bias offset error shown in Figure 4 lead us to support the use of this approach every time the sensor module is static to remove the offset error in the gyroscope reading and produce less drift in orientation. We can see that during the sensor’s static period the algorithm calculated the predicted bias offset error but when the sensor was rotating the algorithm held the previous value of the bias offset.

The result of orientation in the form of quaternion \(q\) indicated that we can approximate the orientation of the sensor module but we can still observe the error that deviated from zero in some axes. This could be because the bias offset error prediction might be over-compensating, causing the error that still occurred in the orientation results. The second part in the algorithm that could help solving this problem is the quaternion estimation using gravity-vector correction.

The measured gravity vector, referenced in the sensor frame, is shown in Figure 5. This measurement indicates the direction of the gravity vector in the sensor frame, affected by the rotation of the sensor module. When the sensor is static, this measurement reflects only the acceleration due to gravity. The calculated gravity vector using the quaternion result in the first part can be used to compare with this gravity vector measurement. The error between the measured and calculated gravity vector shown in Figure 5 provides the mechanism to determine the error between two different sources of measurement (e.g. gyroscope and accelerometer). We can see that with this idea, we could also determine the sensor orientation using the measurement of acceleration due to gravity in the sensor frame. The difference between two vectors can be represented in the form of rotation. In other words, we could achieve the expected orientation of the sensor if we can determine how much we have to rotate the calculated gravity vector to match the measured gravity vector. The angular difference between these two vectors was determined and represented as quaternion (rotation). This angle difference was then included in the previous calculated quaternion and resulted in the estimated orientation of the sensor module as shown in Figure 5. The result shows that the method of using the gravity vector to correct the orientation estimation can help to improve the output. The inclination through time of the quaternion result, which is the indication of drift problem, was reduced using the gravity vector as the reference as can be seen in the quaternion result of Figure 5.

To verify these results, the quaternion output from both processes (before and after application of the gravity vector correction method) were applied to the rotation of the 3D model (IMU model) as shown in Figure 6(b) and 6(c). Comparing both results to the actual movement that had been captured as the pictures in Figure 6(a), leads us to support that the estimated quaternion after gravity-vector correction \(q\) provides a better result than before we applied the correction. This visualization shows that the 3D model using the original computed quaternion in Figure 6(b)
becomes misaligned after some rotations were applied to the sensor module, a common requirement in the context of robotics applications. Gyroscope drift causes a non-zero angular velocity to be measured while the sensor is static. The erroneous gyroscope output produced in the absence of movement is called bias offset error. In determination of the orientation, the angular velocity is mathematically integrated. Therefore, a small offset in the angular velocity reading can produce large amounts of error in orientation.

We proposed an algorithm to improve the IMU orientation estimation and it consists of two parts. The first part is the prediction of the bias offset error by using a simple linear regression model to determine the bias offset error of the gyroscope while the sensor module is in a static period. This is because in that period of time the gyroscope is supposed to produce the reading of zero. The actual measurement at that time can be used to predict the bias offset error. The result of predicted bias offset error shows that the algorithm can be used to predict the value while the sensor is not moving and hold the same bias offset error while the sensor is not stationary. The unbiased angular velocity was then calculated by subtracting the gyroscope measurement with the predicted bias offset error calculated. This unbiased angular rate was used to determine the orientation in the form of a quaternion representation. The result of the computed quaternion in this part can approximately represent the orientation of the sensor module. There was still some drift present in this result but it was better than integrating the angular velocity without removing the bias offset error. This quaternion result was then used in the second part of the algorithm, where it is utilized to compute the gravity vector described in the sensor frame. This will be compared with the acceleration measured from the module’s accelerometers during the next static period because the measurement will provide the acceleration due to gravity. The difference between calculated and measured gravity vectors was determined. This difference, expressed as a quaternion, was used to supplement the original quaternion result to obtain a corrected quaternion estimation of the module’s orientation. Comparing the orientation estimates indicated by the original quaternion and the corrected quaternion shows that the latter approximates the real orientation sequence better. The drift in rotation was corrected and the orientation estimation was improved as can be verified by the 3D visualization offered in Figure 6.

In conclusion, we believe that this proposed algorithm to correct the drift in gyroscope measurement will be useful in robotics applications when the Inertial Measurement Unit (IMU) is used to determine the orientation of a robot’s component or an unmanned vehicle. The algorithm is simple and easy to implement in any controlling platforms. Especially, this work was performed with the goal that the algorithm could be implemented in a real-time manner. This is a common requirement in the context of robotics applications. The idea of using two types of measurement (e.g., angular velocity from gyroscope and acceleration from accelerometer) to determine the orientation is applicable and useful since the Inertial Measurement Unit is capable of providing both types of signals from the same compact module.

7. ACKNOWLEDGMENT
This research was supported by National Sciences Foundation grants HRD-0833093 and CNS-1532061.

8. REFERENCES