Automatic Calibration of a Three-Axis Magnetic Compass

Raj Sodhi
Wednesday, June 1, 2005

PNI's family of tilt compensated magnetic compasses, or TCM modules, are beginning to show widespread acceptance in the marketplace. The automatic calibration algorithm we shall describe allows the TCM to characterize its magnetic environment while the user rotates the unit through various orientations. This algorithm must also characterize and correct for a residual rotation between the magnetometer and accelerometer coordinate systems. This is an extremely difficult task, especially considering that the module will often be used in a noisy environment, and many times will not be given the full range of rotations possible during the calibration. Given a successful user calibration with its underlying factory calibration s, heading accuracies of 2° or better may be of achieved, even under severe pitch and roll conditions.

First, we shall introduce the concepts of soft and hard iron distortion, providing a little mathematical background. Then, we shall present the two stages of the algorithm, along with the procedure for recursive least squares. Finally, we shall present the results of a simulation putting all these to use.

Magnetic Distortion Introduction

For the magnetometers to be perfectly calibrated for magnetic compassing, they must only sense Earth's magnetic field, and nothing else. With an ideal triaxial magnetometer, the measurements will seem to ride along the surface of a sphere of constant radius, which we may call our "measurement space." The end-user may install a perfectly calibrated module from the factory into an application whose magnetic environment leads to distortions of this measurement space, and therefore a significant degradation in heading accuracy. The two most common impairments to the measurement space are hard and soft iron distortion. Hard iron distortion may be described as a fixed offset bias to the readings, effectively shifting the origin of our ideal measurement space. It may be caused by a permanent magnet mounted on or around the fixture of the compass installation. Soft iron distortion is a direction dependent distortion of the gains of the magnetometers, often caused by the presence of high permeability materials in or around the compass fixture. Figure 1 shows the unimpaired measurement space, or the locus of measurements for a perfectly calibrated magnetometer.

Figure 2 shows the magnetometer with a hard iron distortion impairment of 10 uT in the X direction, 20 uT in the Y direction, and 30 uT in the Z direction, and soft iron distortion impairments. One may observe that the locus of measurements has now changed from being spherical to ellipsoidal.

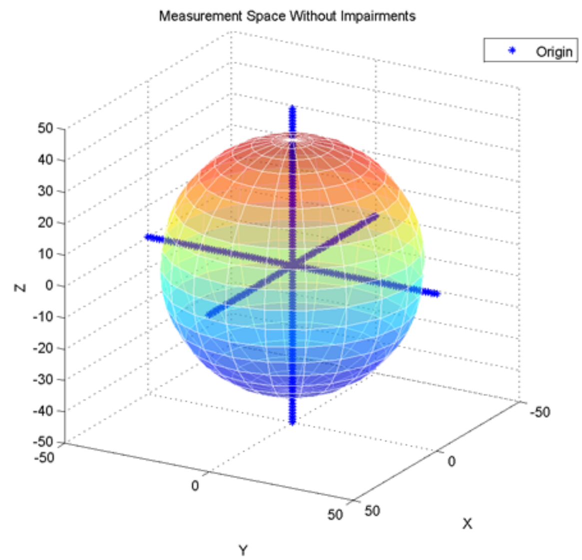


Figure 1. Measurement space for perfectly calibrated magnetometer without hard iron and soft iron distortions.

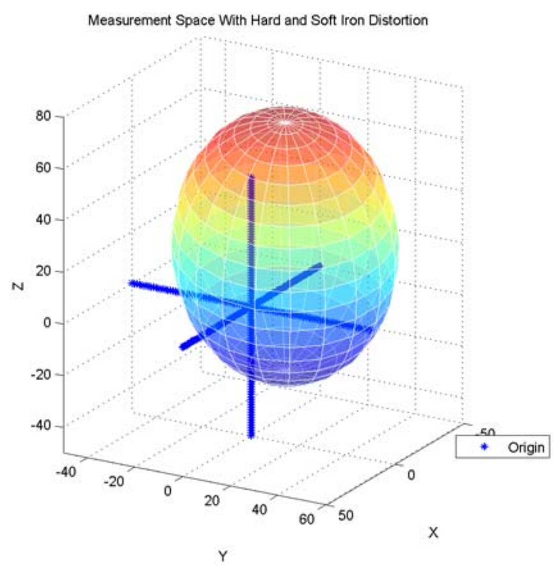


Figure 2. Measurement space with hard and soft iron distortions.

We will represent the undistorted Earth's magnetic field as “Be”, and what is measured as “Bm”. To go from one to the other, we may use the following equations.

(eq. 1) $$Bm = S^{-1} \cdot Be + H \qquad\qquad Be = S \cdot (Bm - H)$$

The soft iron matrix “S” is 3 x 3, and the hard iron vector “H” is 3 x 1. It will be our job to undo the effects of S and H in terms of heading accuracy.

Algorithm Derivation

From the three magnetometers and three accelerometers, we receive six streams of data. At regular intervals of roughly 30 ms, we put the readings from each sensor together to approximately take a snapshot of the modules orientation. We say "approximately" here because there is indeed a delay between when the X magnetometer has been queried and when the Z magnetometer has been queried. Assuming that we have an ideal module that can make error-free measurements of the local magnetic field and acceleration instantly, we may make the following two very important statements.

- The magnitude of Earth's magnetic field is constant, regardless the TCM's orientation.
- The angle formed between Earth's magnetic field vector and Earth's gravitational acceleration vector is constant, once again regardless the TCM's orientation.

The algorithm is founded on the above two statements. We use the first statement to mathematically bend and stretch the magnetometer axes such that they are orthogonal to each other and gain matched. At this stage, we also determine the hard iron offset vectors. We use the second statement to determine a rotation matrix that may be used to fine tune our estimate of the soft iron distortion, and to align the magnetometer coordinate system with the accelerometer coordinate system. Therefore, the algorithm comes in two stages. The first stage centers our ellipsoidal measurement space about the origin, and makes it spherical. The second stage rotates the sphere slightly.

We may express the magnitude of Earth's magnetic field in terms of our measurement vector Bm.

(eq. 2) $$(|Be|)^2 = [[S \cdot (Bm - H)]]^T \cdot [S \cdot (Bm - H)] = \left(Bm^T - H^T\right) \cdot S^T \cdot S \cdot (Bm - H)$$

The middle term may be expressed as a single 3 x 3 symmetrical matrix, C.

(eq. 3) $$C = S^T S$$

Multiplying these terms out, we get the following quadratic equation.

(eq. 4) $$(|Be|)^2 = Bm^T C \cdot Bm - 2 \cdot Bm^T C \cdot H + H^T C \cdot H$$

Stage One of Calibration Process

At this stage, we shall assume that our soft iron matrix is upper triangular, correcting for this assumption in the second stage of the calibration.

(eq. 5)

$$S\_ut^T S\_ut = \begin{pmatrix} 1 & 0 & 0 \\ s12 & s22 & 0 \\ s13 & s23 & s33 \end{pmatrix} \cdot \begin{pmatrix} 1 & s12 & s13 \\ 0 & s22 & s23 \\ 0 & 0 & s33 \end{pmatrix} = \begin{pmatrix} 1 & s12 & s13 \\ s12 & s12^2 + s22^2 & s12 \cdot s13 + s22 \cdot s23 \\ s13 & s12 \cdot s13 + s22 \cdot s23 & s13^2 + s23^2 + s33^2 \end{pmatrix}$$

$$C = S\_ut^T S\_ut = \begin{pmatrix} 1 & c12 & c13 \\ c12 & c22 & c23 \\ c13 & c23 & c33 \end{pmatrix}$$

(eq. 6)

If we assume that our measurement Bm may be expressed as a 3 x 1 vector of $[x, y, z]^T$, we obtain the following.

$$(x \;\; y \;\; z) \cdot \begin{pmatrix} 1 & c12 & c13 \\ c12 & c22 & c23 \\ c13 & c23 & c33 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} - 2 \cdot (x \;\; y \;\; z) \cdot C \cdot H + H^T \cdot C \cdot H = (|Be|)^2$$

(eq. 7)

There are three parts to this equation. We shall deal with each separately. The first part may be expanded to give terms that are second-order.

(eq. 8) $$T1 = x^2 + 2 \cdot x \cdot y \cdot c12 + 2 \cdot x \cdot z \cdot c13 + y^2 \cdot c22 + 2 \cdot y \cdot z \cdot c23 + z^2 \cdot c33$$

The second gives rise to terms that are linear with respect to x, y and z.

$$T2 = -2 \cdot (x \;\; y \;\; z) \cdot C \cdot H = -2 \cdot (x \;\; y \;\; z) \cdot \begin{pmatrix} Lx \\ Ly \\ Lz \end{pmatrix} = -2 \cdot x \cdot Lx - 2 \cdot y \cdot Ly - 2 \cdot z \cdot Lz$$

(eq. 9)

The constant terms may be grouped as follows.

(eq. 10) $$T3 = H^T \cdot C \cdot H - (|Be|)^2$$

After some algebraic manipulations, we may recast this equation as the inner product of two vectors: a changing input vector, and our estimation parameter vector.

$$\mathrm{obs}(n) = x^2 = u(n)^T \cdot w = \begin{pmatrix} -2\cdot x\cdot y \\ -2\cdot x\cdot z \\ -y^2 \\ -2\cdot y\cdot z \\ -z^2 \\ 2\cdot x \\ 2\cdot y \\ 2\cdot z \\ 1 \end{pmatrix}^T \cdot \begin{bmatrix} c12 \\ c13 \\ c22 \\ c23 \\ c33 \\ Lx \\ Ly \\ Lz \\ H^T\cdot C\cdot H - (|Be|)^2 \end{bmatrix}$$

(eq. 11)

Since our observation obs(n) and the input vector u(n) are linearly related by w, we may continue to make improvements on our estimates of w as these data are streaming by. Let us assume that using recursive least squares, we have obtained good estimates of the parameters c12, c13, c22, c23, c33, Lx, Ly and Lz. First we organize our the first five parameters into the C matrix as defined in equation 6. Then we may extract our upper triangular soft iron matrix by taking the Cholesky decomposition. The Cholesky decomposition may be thought of as taking the square root of a matrix. If a square matrix is "positive definite" (which we will define shortly), then we may factor this matrix into the product of U^T*U, where U is upper triangular, and U^T is lower triangular. Applying this concept to our C matrix...

$$\mathrm{S\_ut} = \mathrm{chol}(C) = \mathrm{chol}\left(\begin{pmatrix} 1 & c12 & c13 \\ c12 & c22 & c23 \\ c13 & c23 & c33 \end{pmatrix}\right) = \begin{pmatrix} 1 & s12 & s13 \\ 0 & s22 & s23 \\ 0 & 0 & s33 \end{pmatrix}$$

(eq. 12)

However, we cannot take the Cholesky decomposition of any matrix. As we are iteratively improving our estimates of the parameters in w, we may well stumble upon a C matrix where the Cholesky decomposition fails. To guarantee the success of this operation, we must test to see if the C matrix is positive definite. Strictly speaking, a positive definite matrix A may be pre-multiplied and post-multiplied by any vector x to give rise to a positive scalar constant. x^T*A*x > 0. But this is not very useful, as we do not have time to test this condition on hundreds upon thousands of test vectors. Fortunately, there is an easier way. The following are necessary but not sufficient conditions on our C matrix to be positive definite. [1]

1. c22>0 and c33>0.
2. The largest value of C lies on the diagonal.
3. c11+c22 > 2*c12 and c11+c33 > 2*c13 and c22+c33 > 2*c23.
4. The determinant of C is positive.

For our purposes, the above conditions have experimentally shown to be sufficient.

The hard iron vector may be extracted by simply inverting the equation L=C*H from equation 9.

(eq. 13)
$$H = C^{-1} \cdot L = \begin{pmatrix} 1 & c12 & c13 \\ c12 & c22 & c23 \\ c13 & c23 & c33 \end{pmatrix}^{-1} \cdot \begin{pmatrix} Lx \\ Ly \\ Lz \end{pmatrix}$$

At this point, we have established our first level of magnetic distortion correction, having determined the hard iron offset vector, and having found an upper triangular matrix that effectively makes the magnetometer sensors orthogonal and gain matched. We may apply these corrections to our data using equation 1.

(eq. 14)
$$Stage1 = S\_ut \cdot (Bm - H)$$

Stage Two of Calibration Process

Clearly, the assumption of our soft iron matrix being upper triangular is a gross one. The Z sensor may be skewed, or the X sensor may be gain-mismatched, yet neither of these are accounted for so far in our upper triangular matrix. We refine our soft iron matrix by estimating a rotation matrix that aligns our now orthogonal magnetometer coordinate system to the accelerometer coordinate system. In other words, we may left multiply equation 14 by a 3 x 3 matrix R as follows.

(eq. 15)
$$Stage2 = R \cdot Stage1 = R \cdot S\_ut \cdot (Bm - H)$$

Note that we are assuming that the accelerometer coordinate system is orthogonal, and the data received from the sensors is ideal. The vector result “Stage2” will always make the same angle to the gravitational acceleration direction vector, regardless the orientation of the TCM module. Going one step further, we may say that the cosine of this angle will be constant, and therefore the dot product between these two vectors should be constant. If we represent the accelerometer and stage one magnetometer readings as 3 x 1 vectors, we get the following.

$$\text{Accel} = \begin{pmatrix} ax \\ ay \\ az \end{pmatrix} \quad \text{and} \quad \text{Mag} = R \cdot \text{Stage1} = \begin{pmatrix} r11 & r12 & r13 \\ r21 & r22 & r23 \\ r31 & r32 & r33 \end{pmatrix} \cdot \begin{pmatrix} xc \\ yc \\ zc \end{pmatrix} \qquad \text{(eq. 16)}$$

The dot product may be expressed as follows.

$$\text{Accel}^T \cdot \text{Mag} = \begin{pmatrix} ax & ay & az \end{pmatrix} \cdot \begin{pmatrix} r11 & r12 & r13 \\ r21 & r22 & r23 \\ r31 & r32 & r33 \end{pmatrix} \cdot \begin{pmatrix} xc \\ yc \\ zc \end{pmatrix} = \text{constant} \qquad \text{(eq. 17)}$$

Dividing through by r11,

$$\begin{pmatrix} ax & ay & az \end{pmatrix} \cdot \begin{pmatrix} 1 & \frac{r12}{r11} & \frac{r13}{r11} \\ \frac{r21}{r11} & \frac{r22}{r11} & \frac{r23}{r11} \\ \frac{r31}{r11} & \frac{r32}{r11} & \frac{r33}{r11} \end{pmatrix} \cdot \begin{pmatrix} xc \\ yc \\ zc \end{pmatrix} = \frac{\text{constant}}{r11} \qquad \text{(eq. 18)}$$

As we had done in stage one (equation 11), we may cast our linear equation as the inner product of an input vector and a parameter vector.

$$\text{obs}(n) = ax \cdot xc = u(n)^T \cdot w = \begin{pmatrix} -ax \cdot yc \\ -ax \cdot zc \\ -ay \cdot xc \\ -ay \cdot yc \\ -ay \cdot zc \\ -az \cdot xc \\ -az \cdot yc \\ -az \cdot zc \\ 1 \end{pmatrix}^T \cdot \begin{pmatrix} \frac{r12}{r11} \\ \frac{r13}{r11} \\ \frac{r21}{r11} \\ \frac{r22}{r11} \\ \frac{r23}{r11} \\ \frac{r31}{r11} \\ \frac{r32}{r11} \\ \frac{r33}{r11} \\ \frac{-\text{Accel}^T \cdot \text{Mag}}{r11} \end{pmatrix} \qquad \text{(eq. 19)}$$

Once again, we find that our observation obs(n) is linearly related to the input vector u(n), and our estimates of the parameter vector w shall improve as we receive more and more data. Thanks to the recursive least squares algorithm, we may estimate the parameters in w, obtaining the terms r12/r11, r13/r11, etc. These terms may be used to construct a scaled rotation matrix T as follows.

(eq. 20)
$$T = \begin{pmatrix} 1 & \frac{r12}{r11} & \frac{r13}{r11} \\ \frac{r21}{r11} & \frac{r22}{r11} & \frac{r23}{r11} \\ \frac{r31}{r11} & \frac{r32}{r11} & \frac{r33}{r11} \end{pmatrix}$$

A big unknown here is r11. To find it, we will make use of an important property of rotation matrices: the determinant is always equal to one.

(eq. 21)
$$|T| = \frac{|R|}{r11^3} = \frac{1}{r11^3}$$

So only if the above determinant is positive, we may update our estimate of r11.

(eq. 22)
$$r11 = \left(\frac{1}{|T|}\right)^{\frac{1}{3}}$$

And now our rotation matrix has been solved.

(eq. 23)
$$R = r11 \cdot T$$

We may use equation 15 to correct our input data for the magnetic distortions.

Recursive Least Squares

At both stage one and stage two of our calibration process, we have said that if we had the solution to the parameter vector, we could use it to solve for S_ut, H and R, but we have not shown how. We shall now detail the process of recursive least squares. At each time sample, we receive six data values from six different sensors: x, y, z from the magnetometers, and ax, ay, az from the accelerometers. For both stage one and stage two, we are trying to track a "desired" signal, which is our observation, obs(n). We do this by iterating on our parameter vector w, which when multiplied by our input vector, minimizes the difference between our estimate and the observation. It will do so in a least squares sense, meaning that it will optimize our parameter vector in such a way that the sum of the squares of the errors between our estimates of the observations will be minimized.

To begin the algorithm, we need to make a number of initializations. Initially, we may assume no soft or hard iron distortion.

(eq. 24)
$$S = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad H = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

"P" is our error covariance matrix. For both stage one and stage two, we initialize it to be the identity matrix multiplied by some large number, like 1e5.

(eq. 25)
$$P = 10^5 \cdot \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

To give a reasonable starting point for our estimated parameter vectors on which we shall iterate, we may initialize as follows.

(eq. 26)
$$w1 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 10^4 \end{pmatrix} \qquad w2 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 50 \end{pmatrix}$$

Meanwhile, we shall have changing input vectors defined as follows.

(eq. 27)

$$u1(n) = \begin{pmatrix} -2\cdot x\cdot y \\ -2\cdot x\cdot z \\ -y^2 \\ -2\cdot y\cdot z \\ -z^2 \\ 2\cdot x \\ 2\cdot y \\ 2\cdot z \\ 1 \end{pmatrix} \qquad u2(n) = \begin{pmatrix} -ax\cdot yc \\ -ax\cdot zc \\ -ay\cdot xc \\ -ay\cdot yc \\ -ay\cdot zc \\ -az\cdot xc \\ -az\cdot yc \\ -az\cdot zc \\ 1 \end{pmatrix}$$

Our desired signal for the two stages are as follows.

(eq. 28) $obs1(n) = x^2 \qquad obs2(n) = ax\cdot xc$

"λ" is an adjustable parameter that adjusts the forgetting factor of the algorithm. Setting this parameter very close to one makes the algorithm behave very similar to the normal least squares algorithm, where all past data are equally considered. If it is set loosely, like to 0.5, the algorithm will adapt quickly to a changing environment, but will not be resistant to noise at all. We have experimentally found that setting it to 0.9 works quite well.

And now we are ready to present the algorithm. At each time interval, as new data arrives, we shall perform the following calculations.

(eq. 29) $k = \dfrac{P\cdot u(n)}{\lambda + u(n)^T\cdot P\cdot u(n)}$ calculate the Kalman gain

(eq. 30) $\alpha = obs(n) - w(n)^T\cdot u(n)$ calculate the a priori error estimate

(eq. 31) $w(n) = w(n-1) + k\cdot\alpha$ update state vector estimate

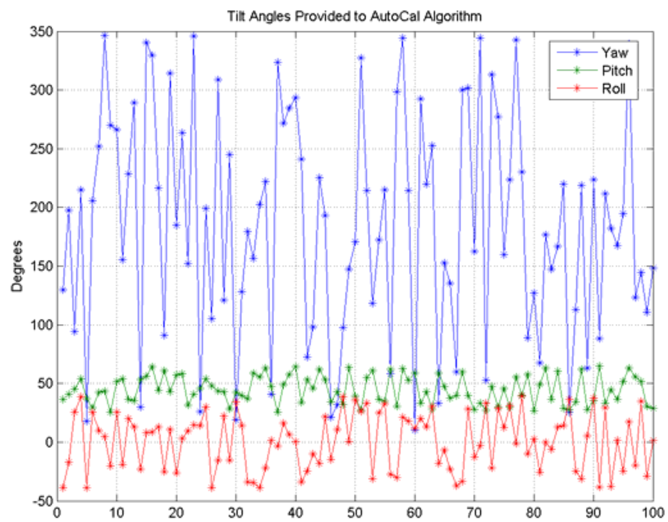
(eq. 32) $P = \dfrac{1}{\lambda}\cdot\left(I_9 - k\cdot u(n)^T\right)\cdot P$ update the error covariance matrix

I_9, in this case, is the 9 x 9 identity matrix.

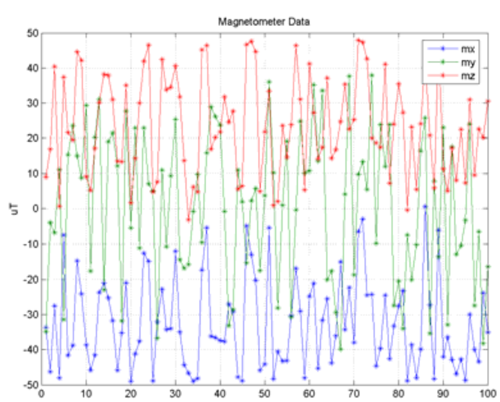
When the algorithm is first starting out, it may well output nonsensical values for the state vector estimate, which could easily give rise to imaginary numbers in the soft iron distortion matrix. To prevent this, we only update our estimates of soft and hard iron distortion if the C matrix is positive definite in stage one and if the determinant of T is positive in stage two.

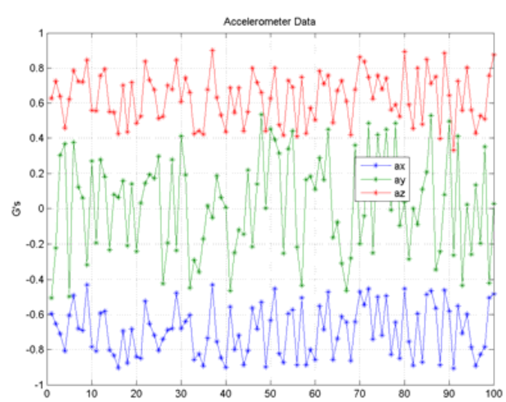
Algorithm Simulation

The algorithm described above was implemented in Matlab. In San Francisco, the magnitude of Earth's magnetic field is 49.338 uT, the dip angle is 61.292°, and the declination is 14.814°. This means that the X, Y and Z components of Earth's magnetic field are [22.9116 ; 6.0595 ; 43.2733] uT. The gravitational acceleration may be represented by [0 ; 0 ; 1]. We assume that there are 100 samples submitted to the algorithm. We randomize the yaw, pitch and roll so that they vary between 0 - 360°, 25 - 65°, and -40 – 40° respectively.



For each triad of yaw, pitch and roll, a 3 x 3 rotation matrix is calculated to move from our Earth inertial reference frame coordinate system to that of the compass. Once the transformation matrix has been calculated, it may be used to multiply the North vector and the gravity vector to simulate our ideal measurements.



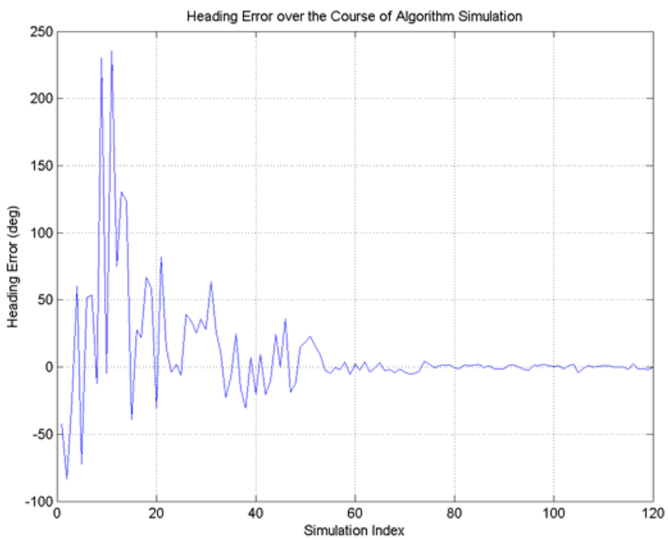


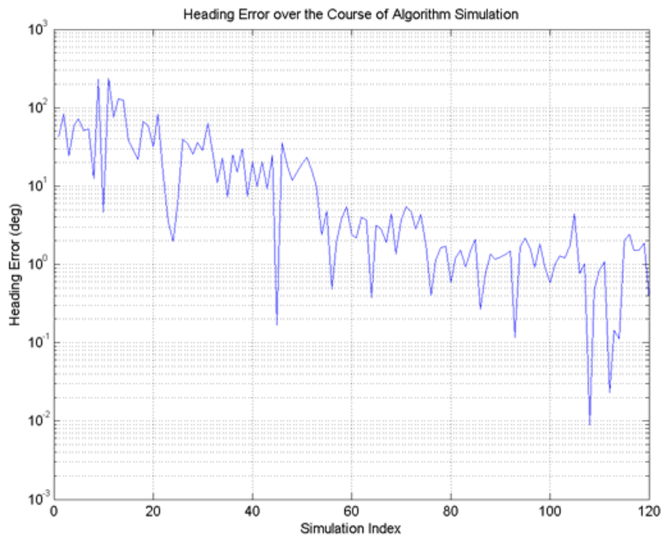
The soft iron distortion matrix was randomly generated, and yet designed to be close to the identity matrix.

Actual SI = [
   0.9567   0.0288   0.1189
  -0.1666   0.8854   -0.0038
   0.0125   0.1191   1.0327
] ;

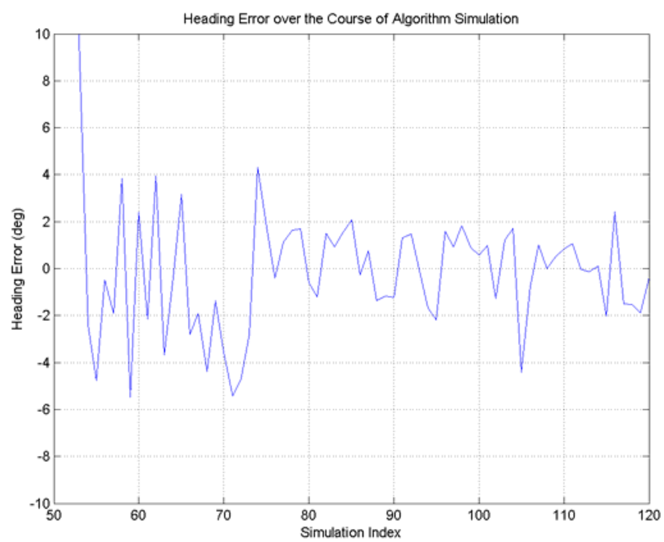
The hard iron vector was chosen as [10 ; 20 ; 30] uT.

These magnetic distortion impairments were applied to the data to simulate our measured data. To make this data seem more real, 30 nT RMS noise was added to the magnetometer data and 3 mG RMS noise was added to the accelerometer data. Finally, the data were submitted to the recursive least squares estimation algorithm. Initially, the algorithm did very poorly with regards to heading accuracy. But as the algorithm learned about its magnetic impairments, the heading error diminished.





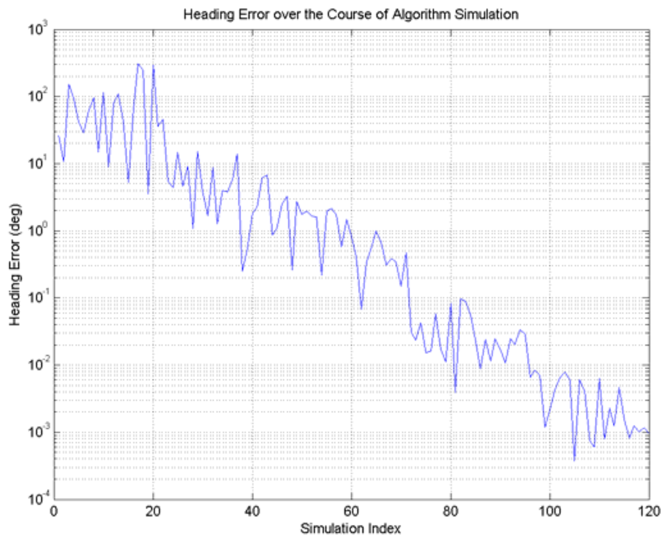
If we look at the heading error after its initial learning period, we may get a feel for the accuracy of our simulated compass.



The noise level of the heading directly depends on two things.

- the amount of coverage of the sphere in X, Y and Z during the calibration process.
- the amount of noise added to our ideal measurements.

If we allow for perfect coverage of the sphere, and if we turn off the noise, the heading error becomes vanishingly small.



Concluding Remarks

The three dimensional auto calibration algorithm presented above has enormous potential for magnetic compassing applications such as in cell phones, PDAs, etc. Based on how much noise is added to the raw measurements, and based on how much of the sphere has been covered over the course of the algorithm, we may see heading errors of 2° or less.

References.

[1]. http://mathworld.wolfram.com/PositiveDefiniteMatrix.html
[2]. Haykin, Simon, Adaptive Filter Theory, third edition, Prentice-Hall, 1996, chapters 11 and 13.
[3]. Lecture Notes by Professor Ioan Tabus at http://www.cs.tut.fi/~tabus/course/ASP/Lectures_ASP.html