A General Calibration Algorithm for 3-Axis Compass/Clinometer Devices

Beat HEEB, Switzerland
Sept 2009

This article describes a general algorithm for the calibration of 3-axis electronic compass/clinometer systems and gives an analysis of its properties. The algorithm presented is fast and accurate and makes minimal assumptions about the calibration environment.

Introduction

Electronic 3-axis compass/clinometer systems proved to be very useful for compact survey instruments. They allow the determination of the precise orientation of the device in space relative to gravitation and the earth magnetic field. Such a system works for any direction with any orientation of the device. There are no levelling requirements and no restriction on steepness.

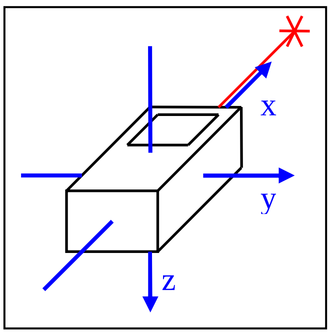
Due to manufacturing tolerances and external influences, such a system inevitably reveals certain errors. Among them are:
- Offset and gain errors of the sensors.
- Sensors mounted with incorrect angles.
- Angular errors between the sensors and the Laser beam.
- Influences of the battery and other metal parts on the magnetic field.

Fortunately, all these errors can be eliminated relatively easily by a linear correction function applied to the sensor values before evaluation. The remainder of this article describes a method to calculate the coefficients of this correction function from a set of calibration measurements.

The Calibration Function

The calculation of the calibration function unavoidably includes a lot of vector and matrix algebra. In the following bold capital letters (**A**) are used for three dimensional matrices and small letters with an arrow ($\vec{a}$) for 3 dimensional (column) vectors. Appendix A contains a summary of the vector and matrix operations used.

For the analysis of the problem, it is convenient to work in the local coordinate system of the device instead of a fixed, earth-referenced system. We generally use a coordinate system with the x-axis in forward (Laser beam) direction, y to the right, and z down (Fig. 1).



If the gravity and magnetic field sensors are mounted along these axes, the values read from them naturally form two vectors in our coordinate system: The gravity vector $\vec{gs}_i$ and the magnetic field vector $\vec{ms}_i$, where i runs over all measurements used for the calibration.

To apply calibration corrections we introduce the result vectors $\vec{gr}_i$ and $\vec{mr}_i$ connected to the sensor values by a linear function:

$$\begin{aligned} \vec{gr}_i &= \mathbf{G} \circ \vec{gs}_i + \vec{gd} \\ \vec{mr}_i &= \mathbf{M} \circ \vec{ms}_i + \vec{md} \end{aligned} \qquad 1)$$

For the final direction measurements, the result vectors are used to compute the azimuth and inclination angles.

During calibration, the goal is to find values for the two matrices **G** and **M** and the two offset vectors $\vec{g}d$ and $\vec{m}d$ such that the errors are minimized. To get a measure for the quality of a calibration we introduce the 'true' vectors $\vec{g}t_i$ and $\vec{m}t_i$. These are the theoretical exact gravity and magnetic field vectors for a given orientation of the device. The magnitude of the two vectors has no influence on the calibration and the final direction calculations. It is therefore arbitrarily set to one.

To characterize the orientation of the device in space we use the yaw, pitch, and roll angles (also known as Tait-Bryan rotations or z-y-x Euler angles) given as follows:
- Start with the device in normal position, z axis down and x axis pointing to north.
- Turn the device around the local z-axis by the yaw angle ψ (= azimuth).
- Turn the device around the local y-axis by the pitch angle θ (= inclination).
- Turn the device around the local x-axis by the roll angle φ.

In normal position the gravity vector points down along the z-axis:

$$\vec{g}t_{normal} = \vec{z} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$$

The magnetic field is slightly more complicated. At the equator it points in the direction of the x-axis. At all other places it includes a vertical part. In general it can be written as:

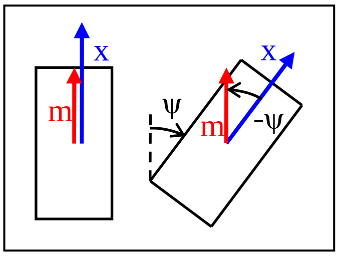
$$\vec{m}t_{normal} = \mathbf{R}_{\vec{y}}(\alpha) \circ \vec{z}$$

Where α is the angle between the magnetic field vector and the gravity vector. It is connected to the dip angle, the (negated) inclination of the magnetic field lines as follows:

$$\alpha = 90° - dip$$

At the equator dip is 0 and α is 90°. In the northern hemisphere dip and α are both between 0 and 90°. In the southern hemisphere dip is negative and α is between 90° and 180°.

Now if we turn the device around a given axis by a given angle both vectors turn around the same axis but in the opposite direction because it is the coordinate system which turns, the vectors are globally fixed (Fig. 2). If we apply all rotations as stated above we get the following general form of the true vectors:



$$\vec{g}t_i = \mathbf{R}_{\vec{x}}(-\varphi_i) \circ \mathbf{R}_{\vec{y}}(-\theta_i) \circ \vec{z}$$
$$\vec{m}t_i = \mathbf{R}_{\vec{x}}(-\varphi_i) \circ \mathbf{R}_{\vec{y}}(-\theta_i) \circ \mathbf{R}_{\vec{z}}(-\psi_i) \circ \mathbf{R}_{\vec{y}}(\alpha) \circ \vec{z} \qquad 2)$$

Solving these equations for θ and Ψ gives the well known equations needed to get the final azimuth and inclination angles from the measured vectors:

$$\theta = -\arctan(\vec{g}r_x, \sqrt{\vec{g}r_y^{\ 2} + \vec{g}r_z^{\ 2}})$$
$$\psi = \arctan((\vec{g}r_y \cdot \vec{m}r_z - \vec{g}r_z \cdot \vec{m}r_y) \cdot |\vec{g}r|, \vec{m}r_x \cdot |\vec{g}r|^2 - \vec{g}r_x \cdot (\vec{g}r \bullet \vec{m}r)) \qquad 3)$$

Arctan is used here in the sense of the computer function arctan2(y, x) = argument of the vector [x, y] in the range ±π. For an alternate derivation see [1].

For the calibration the true vectors can be used to define the error of a set of calibration measurements as the RMS average of the distances between the calculated and the true vectors:

$$E^2 = \frac{1}{n} \cdot \sum_i |\vec{g}r_i - \vec{g}t_i|^2 + |\vec{m}r_i - \vec{m}t_i|^2 \qquad 4)$$

Now suppose for the moment we have a calibration environment which allows us to measure the device orientation angles exactly for each of a set of calibration measurements. This allows us to calculate the true vectors for each measurement using equations 2). To get an optimal calibration we simply search for **G**, **M**, $\vec{g}d$, and $\vec{m}d$ such that the squared error gets minimal.

The minimum can be found algebraically by setting all the partial derivatives to zero:

$$\begin{aligned}
\nabla_{\mathbf{G}} E^2 &= \tfrac{2}{n} \cdot \sum_i (\mathbf{G} \circ \vec{g}s_i + \vec{g}d - \vec{g}t_i) \otimes \vec{g}s_i = \mathbf{0} \\
\nabla_{\mathbf{M}} E^2 &= \tfrac{2}{n} \cdot \sum_i (\mathbf{M} \circ \vec{m}s_i + \vec{m}d - \vec{m}t_i) \otimes \vec{m}s_i = \mathbf{0} \\
\nabla_{\vec{g}d} E^2 &= \tfrac{2}{n} \cdot \sum_i \mathbf{G} \circ \vec{g}s_i + \vec{g}d - \vec{g}t_i = \vec{0} \\
\nabla_{\vec{m}d} E^2 &= \tfrac{2}{n} \cdot \sum_i \mathbf{M} \circ \vec{m}s_i + \vec{m}d - \vec{m}t_i = \vec{0}
\end{aligned} \qquad 5)$$

The Nabla or Del operator ∇ means a vector or matrix consisting of the derivatives to the elements of the given vector or matrix. There is no magic behind this operator; it is just a compact representation of the large number of derivatives.

The equation system can be solved algebraically resulting in:

$$\begin{aligned}
\mathbf{Gs} &= \overline{\vec{g}s_i \otimes \vec{g}s_i} - \overline{\vec{g}s_i} \otimes \overline{\vec{g}s_i} \\
\mathbf{Ms} &= \overline{\vec{m}s_i \otimes \vec{m}s_i} - \overline{\vec{m}s_i} \otimes \overline{\vec{m}s_i} \\
\mathbf{G} &= (\overline{\vec{g}t_i \otimes \vec{g}s_i} - \overline{\vec{g}t_i} \otimes \overline{\vec{g}s_i}) \circ \mathbf{Gs}^{-1} \\
\mathbf{M} &= (\overline{\vec{m}t_i \otimes \vec{m}s_i} - \overline{\vec{m}t_i} \otimes \overline{\vec{m}s_i}) \circ \mathbf{Ms}^{-1} \\
\vec{g}d &= \overline{\vec{g}t_i} - \mathbf{G} \circ \overline{\vec{g}s_i} \\
\vec{m}d &= \overline{\vec{m}t_i} - \mathbf{M} \circ \overline{\vec{m}s_i}
\end{aligned} \qquad 6)$$

Averages ($\overline{\ldots}$) over i are used here instead of sums to avoid the numerous $1/n$ terms.

We get a result only if the **Gs** and **Ms** matrices are invertible. This can easily be guaranteed if we have a large number of measurements and the corresponding $\vec{g}s_i$ and $\vec{m}s_i$ vectors are evenly spread over all possible directions. In this case the average $\overline{\vec{g}s_i}$ and $\overline{\vec{m}s_i}$ vectors as well as the non-diagonal elements of the $\overline{\vec{g}s_i \otimes \vec{g}s_i}$ and $\overline{\vec{m}s_i \otimes \vec{m}s_i}$ matrices cancel out statistically. The sum of the diagonal elements of either of these matrices is equal to the square of the length of the corresponding sensor vector. The average matrices can therefore be approximated as:

$$\mathbf{Gs} \cong \tfrac{1}{3} \cdot \overline{|\vec{g}s_i|^2} \cdot \mathbf{I} \quad \mathbf{Ms} \cong \tfrac{1}{3} \cdot \overline{|\vec{m}s_i|^2} \cdot \mathbf{I}$$

with the identity matrix **I**.

If the spreading is not perfect, the matrices are no longer diagonal but the diagonal elements are still dominant and the matrices are still invertible.

In reality not all of the orientation angles of the calibration measurements are known. In this case we simply treat all not a priori known angles as additional unknown of the optimization problem. This results in a much larger but still manageable equation system.

Three types of measurements are common in practice:

1) Known Direction

The traditional approach to calibration is to measure a given course of precisely known survey stations. A calibration procedure for such a setup is for instance described in [1].

For given survey stations, we know the direction and therefore the ψ and θ angles but not the roll angle φ. This gives us a new unknown φ_i and a new equation for each measurement:

$$\forall i: \frac{\partial E^2}{\partial \varphi_i} = \tfrac{2}{n} \cdot [\vec{g}t_i, \vec{g}r_i, \vec{x}] + \tfrac{2}{n} \cdot [\vec{m}t_i, \vec{m}r_i, \vec{x}] = 0 \qquad 7)$$

To get a replacement for the no longer known true vectors we define the partially turned vectors:

$$\vec{g}p_i = \vec{g}t_i(\varphi_i = 0) = \mathbf{R}_{\vec{y}}(-\theta_i) \circ \vec{z}$$
$$\vec{m}p_i = \vec{m}t_i(\varphi_i = 0) = \mathbf{R}_{\vec{y}}(-\theta_i) \circ \mathbf{R}_{\vec{z}}(-\psi_i) \circ \mathbf{R}_{\vec{y}}(\alpha) \circ \vec{z}$$

The true vectors $\vec{g}t_i$ and $\vec{m}t_i$ can now be approximated using an estimated φ_i:

$$\vec{g}t_i = \mathbf{R}_{\vec{x}}(-\varphi_i) \circ \vec{g}p_i$$
$$\vec{m}t_i = \mathbf{R}_{\vec{x}}(-\varphi_i) \circ \vec{m}p_i$$

Substitution in 7) gives

$$[\mathbf{R}_{\vec{x}}(-\varphi_i) \circ \vec{g}p_i, \vec{g}r_i, \vec{x}] + [\mathbf{R}_{\vec{x}}(-\varphi_i) \circ \vec{m}p_i, \vec{m}r_i, \vec{x}] = 0 \qquad 8)$$

This can be solved as

$$\tan(\varphi_i) = \frac{[\vec{g}p_i, \vec{g}r_i, \vec{x}\,] \;+\; [\vec{m}p_i, \vec{m}r_i, \vec{x}]}{[\vec{g}p_i \times \vec{x}, \vec{g}r_i, \vec{x}\,] + [\vec{m}p_i \times \vec{x}, \vec{m}r_i, \vec{x}]}$$
$$= \frac{\vec{g}r_z \cdot \vec{g}p_y - \vec{g}r_y \cdot \vec{g}p_z + \vec{m}r_z \cdot \vec{m}p_y - \vec{m}r_y \cdot \vec{m}p_z}{\vec{g}r_y \cdot \vec{g}p_y + \vec{g}r_z \cdot \vec{g}p_z + \vec{m}r_y \cdot \vec{m}p_y + \vec{m}r_z \cdot \vec{m}p_z} \qquad 9)$$

We assume here that $\vec{g}r_i$ and $\vec{m}r_i$ are given vectors. However, their values depend on the calibration coefficients which in turn depend on true vectors and thus on the φ_i. So we have to solve the system consisting of equations 5) and 8) simultaneously. This can no longer be done algebraically but it can easily be done iteratively by applying 6) and 9) in turn until the result stabilizes.

2) Free Measurements

The method mentioned above is only useful if we have access to a set of precisely known survey stations. In practice it would be very desirable to have a way to do a calibration any time at any place. To approach this goal we examine the case of a set of free measurements without any known direction angles. This means all angles ψ, θ, and φ need to be treated as unknown. In contrast to equation 7) where we search the optimal turn angle around the x axis, we now search for an optimal rotation around any axis. Equation 7) is still valid but in addition we can substitute $\vec{x}$ by any other vector, in particular by $\vec{y}$ and $\vec{z}$:

$$[\vec{g}t_i, \vec{g}r_i, \vec{x}] + [\vec{m}t_i, \vec{m}r_i, \vec{x}] = 0$$
$$[\vec{g}t_i, \vec{g}r_i, \vec{y}] + [\vec{m}t_i, \vec{m}r_i, \vec{y}] = 0$$
$$[\vec{g}t_i, \vec{g}r_i, \vec{z}] + [\vec{m}t_i, \vec{m}r_i, \vec{z}] = 0$$

The three equations above can be combined to:

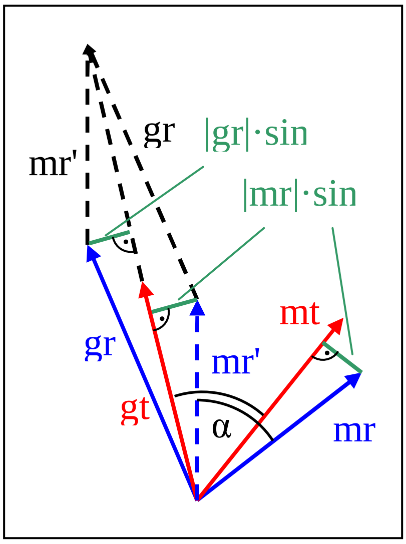
$$\vec{g}t_i \times \vec{g}r_i + \vec{m}t_i \times \vec{m}r_i = \vec{0} \qquad 10)$$

In addition we have the general properties:

$$\left|\vec{g}t_i\right| = \left|\vec{m}t_i\right| = 1 \quad \angle\vec{g}t_i, \vec{m}t_i = \alpha \tag{11}$$

Together equations 10) and 11) uniquely define the vectors $\vec{g}t_i$ and $\vec{m}t_i$ (except for a mirrored solution $\vec{g}t_i' = -\vec{g}t_i$ and $\vec{m}t_i' = -\vec{m}t_i$). To satisfy equation 10) they have to reside in the same plane as $\vec{g}r_i$ and $\vec{m}r_i$. Inside this plane they must be rotated such that:

$$\left|\vec{g}t_i \times \vec{g}r_i\right| = \left|\vec{g}r_i\right| \cdot \sin(\angle\vec{g}t_i, \vec{g}r_i) = \left|\vec{m}t_i \times \vec{m}r_i\right| = \left|\vec{m}r_i\right| \cdot \sin(\angle\vec{m}t_i, \vec{m}r_i)$$



They can be constructed geometrically (Fig. 3) or calculated algebraically using the plane normal vector $\vec{n}_i$:

$$\begin{aligned}
\vec{n}_i &= \left\langle \vec{g}r_i \times \vec{m}r_i \right\rangle \\
\vec{m}r_i' &= \vec{m}r_i \cdot \cos(\alpha) + (\vec{m}r_i \times \vec{n}_i) \cdot \sin(\alpha) \\
\vec{g}t_i &= \left\langle \vec{g}r_i + \vec{m}r_i' \right\rangle \\
\vec{m}t_i &= \vec{g}t_i \cdot \cos(\alpha) + (\vec{n}_i \times \vec{g}t_i) \cdot \sin(\alpha)
\end{aligned} \tag{12}$$

Calibration using free measurements has one major drawback: since the Laser direction is not used at all, it cannot be used to calibrate the angular error between the sensors and the Laser. We therefore have to do at least part of the measurements in a given direction. However, as we will see in the next section, there is no need to know this direction in advance.

3) Unidirectional Groups

An alternative to free measurements and measurements of known directions is to measure a fixed but not a priori known direction several times with various roll angles. Such a set of measurements with common ψ and θ but varying φ is called a unidirectional group. Similar to the case of known directions we have an unknown φ for each measurement and equations 7) and 9) still apply. In addition we have two unknown angles ψ and θ for each group to fix its direction. The corresponding equations are equivalent to the case of free measurements except that they are summed up over all measurements of the group:

$$\forall k : \sum_{i \in Sk} \vec{g}t_i \times \vec{g}r_i + \vec{m}t_i \times \vec{m}r_i = \vec{0} \tag{13}$$

Here k runs over all groups and *Sk* means the set of measurements belonging to group *k*.

As in the case of known directions, we define partially turned vectors for each group:

$$\begin{aligned}
\vec{g}p_k &= \mathbf{R}_{\vec{y}}(-\theta_k) \circ \vec{z} \\
\vec{m}p_k &= \mathbf{R}_{\vec{y}}(-\theta_k) \circ \mathbf{R}_{\vec{z}}(-\psi_k) \circ \mathbf{R}_{\vec{y}}(\alpha) \circ \vec{z}
\end{aligned}$$

and write the true vectors as:

$$\begin{aligned}
\forall i \in Sk : \vec{g}t_i &= \mathbf{R}_{\vec{x}}(-\varphi_i) \circ \vec{g}p_k \\
\forall i \in Sk : \vec{m}t_i &= \mathbf{R}_{\vec{x}}(-\varphi_i) \circ \vec{m}p_k
\end{aligned} \tag{14}$$

The $\vec{g}p_k$ and $\vec{m}p_k$ can be seen as adapted versions of the $\vec{g}t_i$ and $\vec{m}t_i$ turned to a common roll angle. To be able to find the unknown direction, we introduce similarly adapted versions of the $\vec{g}r_i$ and $\vec{m}r_i$ vectors:

$$\vec{g}a_i = \mathbf{R}_{\vec{x}}(\varphi_i) \circ \vec{g}r_i$$
$$\vec{m}a_i = \mathbf{R}_{\vec{x}}(\varphi_i) \circ \vec{m}r_i \qquad 15)$$

The choice of the common roll angle is arbitrary and has no influence on the result. Instead of using φ = 0, we can as well conform to the roll angle of the first measurement in each group by replacing the φ_i in 14) and 15) with:

$$\delta_i = \varphi_i - \varphi_{ik}$$

where *ik* is the first $i \in Sk$.

This makes the algorithm independent of the global reference system and avoids special handling of the roll angle ambiguities at $\theta = \pm 90°$ (gimbal lock).

Applying $\mathbf{R}_{\vec{x}}(\delta_i)$ to all terms in 13) and substituting the partially turned and adapted vectors we get:

$$\vec{g}p_k \times \vec{g}c_k + \vec{m}p_k \times \vec{m}c_k = \vec{0}$$
$$\vec{g}c_k = \sum_{i \in Sk} \vec{g}a_i$$
$$\vec{m}c_k = \sum_{i \in Sk} \vec{m}a_i$$

This has the same form as 10) and allows us to work out $\vec{g}p_k$ and $\vec{m}p_k$ from the vectors $\vec{g}c_k$ and $\vec{m}c_k$ similar to equations 12):

$$\vec{n}_i = \left\langle \vec{g}c_k \times \vec{m}c_k \right\rangle$$
$$\vec{g}p_k = \left\langle \vec{g}c_k + \vec{m}c_k \cdot \cos(\alpha) + (\vec{m}c_k \times \vec{n}_i) \cdot \sin(\alpha) \right\rangle \qquad 16)$$
$$\vec{m}p_k = \vec{g}p_k \cdot \cos(\alpha) + (\vec{n}_i \times \vec{g}p_k) \cdot \sin(\alpha)$$

Equations 16) and 9) must be solved simultaneously because of their circular dependency. Fortunately iteration between the two equations converges so fast that a single iteration suffices to solve it with high accuracy. In practice we do the following:

- Get an approximation of the δ_i from the relative roll angles of the result vectors $\vec{g}r_i$ and $\vec{m}r_i$.
- Get the adapted vectors from the δ_i using equation 15).
- Get $\vec{g}p_k$ and $\vec{m}p_k$ from $\vec{g}c_k$ and $\vec{m}c_k$ using equations 16).
- Get the final δ_i using equation 9).
- Get $\vec{g}t_i$ and $\vec{m}t_i$ using equation 14).

The Influence of the Magnetic Field

The shape of the magnetic field, in particular its inclination, is a critical value in our calculations. The precision of the corresponding α angle has a direct influence on the accuracy of the resulting calibration coefficients. In principle α is known for any location on earth. In practice, however, α is hard to figure out because it changes over time and there is an uncertainty because of local magnetic anomalies. It is therefore more convenient and more accurate to treat α as an additional unknown of the optimization process. This gives us the following new equation:

$$\frac{\partial E^2}{\partial \alpha} = \tfrac{2}{n} \cdot \sum_i [\vec{m}r_i, \mathbf{R}_{\vec{x}}(-\varphi_i) \circ \mathbf{R}_{\vec{y}}(-\theta_i) \circ \mathbf{R}_{\vec{z}}(-\psi_i) \circ \vec{y}, \vec{m}t_i] = 0$$

Isolation of α gives:

$$[\sum_i \mathbf{R}_{\vec{z}}(\psi_i)\circ\mathbf{R}_{\vec{y}}(\theta_i)\circ\mathbf{R}_{\vec{x}}(\varphi_i)\circ\vec{m}r_i,\vec{y},\mathbf{R}_{\vec{y}}(\alpha)\circ\vec{z}]=0$$

This can be solved as:

$$\tan(\alpha)=\frac{(\sum_i \mathbf{R}_{\vec{z}}(\psi_i)\circ\mathbf{R}_{\vec{y}}(\theta_i)\circ\mathbf{R}_{\vec{x}}(\varphi_i)\circ\vec{m}r_i)\bullet\vec{x}}{(\sum_i \mathbf{R}_{\vec{z}}(\psi_i)\circ\mathbf{R}_{\vec{y}}(\theta_i)\circ\mathbf{R}_{\vec{x}}(\varphi_i)\circ\vec{m}r_i)\bullet\vec{z}}=\frac{\sum_i(\vec{g}t_i\times\vec{m}r_i)\bullet\langle\vec{g}t_i\times\vec{m}t_i\rangle}{\sum_i\vec{m}r_i\bullet\vec{g}t_i}$$

For free measurements where $\vec{g}t_i$, $\vec{m}t_i$, and $\vec{m}r_i$ reside in the same plane, it can be simplified to:

$$\tan(\alpha)=\frac{\sum_i|\vec{m}r_i\times\vec{g}t_i|}{\sum_i\vec{m}r_i\bullet\vec{g}t_i}$$

For unidirectional groups it can be expressed using the adapted vectors:

$$\tan(\alpha)=\frac{\sum_k|\vec{m}c_k\times\vec{g}p_k|}{\sum_k\vec{m}c_k\bullet\vec{g}p_k}$$

Again, these equations circularly depend on all other equations and their evaluation must be included in the main iteration loop.

Note that, although the inclination of the magnet field is a critical value during calibration, the final calibration function is independent of it and a calibrated device works equally well everywhere in the world.

The Roll Angle Ambiguity

A set of calibration measurements consisting of known directions, free measurements, and unidirectional groups does not uniquely define the roll angles. If **G** and **M** are part of a solution of the equation system, so are **G′** and **M′** given by:

$$\mathbf{G}'=R_{\vec{x}}(\omega)\circ\mathbf{G}$$

$$\mathbf{M}'=R_{\vec{x}}(\omega)\circ\mathbf{M}$$

for any angle ω. Since we are not interested in measuring roll angles, any of these solutions is equally well suited for our needs. However, the ambiguity destabilizes the convergence of the main iteration. It is therefore advantageous to define the roll angles somehow. A simple way is to enforce a y-z symmetry in the **G** matrix. This means the roll angle is defined by the mounting angles of the y and z acceleration sensors. The symmetry can easily be enforced by a replacement of the relevant elements of the **G** matrix at the end of each iteration step:

$$\mathbf{G}'_{yz}=\mathbf{G}'_{zy}=\tfrac{1}{2}(\mathbf{G}_{yz}+\mathbf{G}_{zy})$$

Convergence and Termination Condition

The question arises how many iterations we have to make in the main iteration loop until the values converge to the desired result. In general, according to the Banach fixed point theorem the error of an approximation x_n after *n* iterations compared to the precise solution *x* can be calculated as follows:

$$\|x_n-x\|\le\frac{q}{1-q}\cdot\|x_n-x_{n-1}\|$$

The Lipschitz constant q is the improvement of the precision from step to step. For our algorithm, q is mostly independent of the exact values used and can be assumed to be between 0.8 and 0.9 in practice. If we repeat the iteration step as long as

$$\max(\|\mathbf{G}_n - \mathbf{G}_{n-1}\|, \|\mathbf{M}_n - \mathbf{M}_{n-1}\|) > 10^{-6}$$

we can be sure all elements of the final matrices are within 10^{-5} of the exact values. This assures the error introduced is negligible compared to the uncertainty of the measurements. The errors of the offset vectors are ignored here because the elements of these vectors always converge faster than those of the matrices.

The Main Iteration

The main iteration can be summarized as follows:

1) Calculate **Gs** and **Ms** and their inverse and a first estimation of α .

2) Set **G** = **M = I** and $\vec{gd} = \vec{md} = \vec{0}$.

3) Get the $\vec{gr}_i$ and $\vec{mr}_i$ vectors from **G**, **M**, $\vec{gd}$, $\vec{md}$ and the $\vec{gs}_i$ and $\vec{ms}_i$.

4) Get the $\vec{gt}_i$ and $\vec{mt}_i$ from $\vec{gr}_i$, $\vec{mr}_i$ and α.

5) Get a new α from $\vec{mr}_i$ and $\vec{gt}_i$ or $\vec{mc}_k$ and $\vec{gp}_k$.

6) Get new **G**, **M**, $\vec{gd}$, and $\vec{md}$ from the $\vec{gt}_i$ and $\vec{mt}_i$.

7) Enforce $\mathbf{G}_{yz} = \mathbf{G}_{zy}$.

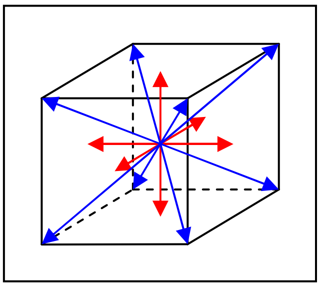
8) Loop to 3) as long as the changes in **G** and **M** $> 10^{-6}$.

Remarks:
- The matrix inversions of **Gs** and **Ms** need to be done only once because the two matrices depend on the sensor values only.
- Measurements with known direction, free measurements, and unidirectional groups may be mixed freely inside a set of calibration measurements.
- Free measurements can be handled as unidirectional groups consisting of a single measurement.

A pseudo code version of the full algorithm is included in appendix B.

The Calibration Procedure

As stated earlier, the set of measurements used to calculate a calibration should be taken such that the device orientations are evenly spread over the rotation group. Just using randomly selected orientations is not a good idea because it turns out to be badly spread in practise. A good choice for a well spread set is for instance the rotational symmetry group of a cube. A cube can be placed with any of the 6 faces up and in each case any of the 4 side faces may be in front, giving a total of 24 orientations. Unfortunately it turns out that 24 measurements are not enough for a good calibration. A perfect set of 60 orientations is contained in the symmetry group of the dodecahedron or icosahedron. However, this set of orientations is not useful in practice because it is too complex to be reproduced in the field.



A proven set of measurements is the following:
Use the 14 directions given by the middle of the 6 faces and the 8 vertices of a cube as seen from its centre. Measure each direction with four evenly spread roll angles, giving a total of 56 measurements (Fig. 4).

Representative numerical values:

	Azimuth (ψ)	*Inclination (θ)*	*Roll Angle (φ)*
Directions 1-4:	0º, 90º, 180º, 270º	0º	0º, 90º, 180º, 270º
Directions 5-6:	0º	90º, -90º	0º, 90º, 180º, 270º
Directions 7-14:	45º, 135º, 225º, 315º	35.3º, -35.3º	0º, 90º, 180º, 270º

How Precise is it?

Generally, a measuring device is useless if we have no idea how precise it is. It is therefore necessary to find a way to quantify the error of the device and the influence of the calibration on it.

Assume for the moment all the sensors are perfect except for the proposed linear errors and we are able to execute the calibration measurements without introducing any additional error. In this case a calibration function exists which exactly compensates all the errors leading to a total error E of zero. This is obviously the minimum we are running into when the algorithm converges.

In practice there are additional errors, namely random errors (noise), nonlinearities, and sighting errors during the measurements. Since these are all small errors on the input variables of our algorithm, we can use error propagation to analyze their influence on the resulting precision.

Let us first look at a single measurement ignoring calibration for the moment. Equations 3) are used to get θ and ψ directly from the measured sensor values. If we assume the sensor errors are random and independent, we can use statistical error propagation:

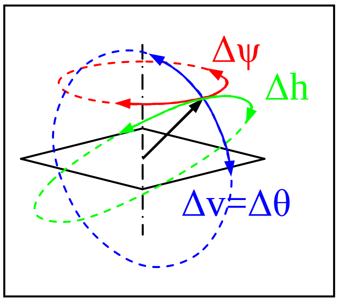
$$\Delta\theta^2 = \sum_i \left(\frac{\partial\theta}{\partial\vec{g}_i} \cdot \Delta\vec{g}_i \right)^2 = \left|\nabla_{\vec{g}}\theta\right|^2 \cdot \Delta g^2 = \frac{\Delta g^2}{|\vec{g}|^2} = \Delta\tilde{g}^2$$

$$\Delta\psi^2 = \sum_i \left(\frac{\partial\psi}{\partial\vec{g}_i} \cdot \Delta\vec{g}_i \right)^2 + \sum_i \left(\frac{\partial\psi}{\partial\vec{m}_i} \cdot \Delta\vec{m}_i \right)^2 = \left|\nabla_{\vec{g}t}\psi\right|^2 \cdot \Delta g^2 + \left|\nabla_{\vec{m}t}\psi\right|^2 \cdot \Delta m^2$$

$$= \frac{\sin(2\cdot\alpha)\cdot\tan(\theta)\cdot\cos(\psi) + \cos(\alpha)^2 + \sin(\alpha)^2\cdot\tan(\theta)^2}{\sin(\alpha)^2} \cdot \Delta\tilde{g}^2 + \frac{1}{\sin(\alpha)^2} \cdot \Delta\tilde{m}^2$$

Where $\Delta\theta$ and $\Delta\psi$ are the errors of the resulting angles, Δg and Δm are the absolute errors of the individual sensors, and $\Delta\tilde{g}$ and $\Delta\tilde{m}$ are the relative errors of the sensors. All of them must be interpreted as standard deviations of the corresponding values.

The error of ψ approaches infinity when θ gets near 90°. This is not a bad behaviour of the compass; it is a property of the Euler angles. When the angles are used to get a direction vector, the influence of ψ decreases when θ approaches 90°, and the error of the vector remains small for all directions. It is therefore appropriate to use the horizontal and vertical components of the error of a unit direction vector instead of the errors of the angles (Fig. 5):



$$\Delta h = \cos(\theta)\cdot\Delta\psi$$

$$\Delta v = \Delta\theta$$

The horizontal error depends slightly on the direction of the measurement. To get a general characteristic value, we use the RMS average of the error over all directions:

$$\overline{\Delta h^2} = \frac{1}{4\pi}\iint \Delta h^2 \cdot \cos(\theta)\cdot d\theta \cdot d\psi = \frac{2}{3\cdot\sin(\alpha)^2}\cdot\left(\Delta\tilde{g}^2 + \Delta\tilde{m}^2\right) - \frac{1}{3}\cdot\Delta\tilde{g}^2$$

Not surprisingly, the error increases with decreasing α, i.e. towards the poles. Note, however, that this is partially compensated by the fact that the magnitude of the magnetic field ($|\vec{m}|$) increases at the same time.

The sighting error is another independent directional error which adds to the sensor induced errors. In summary we have:

$$\overline{\Delta h^2} = \frac{2}{3 \cdot \sin(\alpha)^2} \cdot \left(\Delta\tilde{g}^2 + \Delta\tilde{m}^2\right) - \frac{1}{3} \cdot \Delta\tilde{g}^2 + \Delta d^2$$
$$\overline{\Delta v^2} = \Delta\tilde{g}^2 + \Delta d^2$$

The Influence of the Calibration

If we add calibration, there is no relevant change to the measurement errors above, but we introduce an additional error because of the limited precision of the calibration coefficients. The errors of the coefficients are induced by the sensor and sighting errors of the calibration measurements.

Again they can be calculated using statistical error propagation. To keep things clear, a vector $\vec{x}s$ is used here as a short for all sensor values of all the calibration measurements ($\vec{g}s$ & $\vec{m}s$). Another vector $\vec{C}$ represents all 24 calibration coefficients given by **G**, **M**, $\vec{g}d$, and $\vec{m}d$:

$$\Delta h_{cal}{}^2 = \sum_i \left(\frac{\partial h}{\partial \vec{x}_i}\right)^2 \cdot \Delta\vec{x}_i{}^2 = \sum_i \left(\cos(\theta) \cdot \sum_k \left(\frac{\partial \psi}{\partial \vec{C}_k} \cdot \frac{\partial \vec{C}_k}{\partial \vec{x}_i}\right)\right)^2 \cdot \Delta\vec{x}_i{}^2$$
$$\Delta v_{cal}{}^2 = \sum_i \left(\frac{\partial v}{\partial \vec{x}_i}\right)^2 \cdot \Delta\vec{x}_i{}^2 = \sum_i \left(\sum_k \left(\frac{\partial \theta}{\partial \vec{C}_k} \cdot \frac{\partial \vec{C}_k}{\partial \vec{x}_i}\right)\right)^2 \cdot \Delta\vec{x}_i{}^2$$

Reordering gives:

$$\Delta h_{cal}{}^2 = \sum_{ikl} ch_{kl} \cdot \frac{\partial \vec{C}_k}{\partial \vec{x}_i} \cdot \frac{\partial \vec{C}_l}{\partial \vec{x}_i} \cdot \Delta\vec{x}_i{}^2 \qquad ch_{kl} = \cos(\theta)^2 \cdot \frac{\partial \psi}{\partial \vec{C}_k} \cdot \frac{\partial \psi}{\partial \vec{C}_l}$$
$$\Delta v_{cal}{}^2 = \sum_{ikl} cv_{kl} \cdot \frac{\partial \vec{C}_k}{\partial \vec{x}_i} \cdot \frac{\partial \vec{C}_l}{\partial \vec{x}_i} \cdot \Delta\vec{x}_i{}^2 \qquad cv_{kl} = \frac{\partial \theta}{\partial \vec{C}_k} \cdot \frac{\partial \theta}{\partial \vec{C}_l}$$

ch_{kl} and cv_{kl} are universal coefficients given by equations 3) and 1). To remove their dependency on the device orientation during the final measurement, we use the RMS average over all orientations:

$$\overline{\Delta h_{cal}{}^2} = \sum_{ikl} \overline{ch_{kl}} \cdot \frac{\partial \vec{C}_k}{\partial x_i} \cdot \frac{\partial \vec{C}_l}{\partial x_i} \cdot \Delta\vec{x}_i{}^2 \qquad \overline{ch_{kl}} = \frac{1}{8\pi^2} \iiint ch_{kl} \cdot \cos(\theta) \cdot d\theta \cdot d\psi \cdot d\varphi$$
$$\overline{\Delta v_{cal}{}^2} = \sum_{ikl} \overline{cv_{kl}} \cdot \frac{\partial \vec{C}_k}{\partial x_i} \cdot \frac{\partial \vec{C}_l}{\partial x_i} \cdot \Delta\vec{x}_i{}^2 \qquad \overline{cv_{kl}} = \frac{1}{8\pi^2} \iiint cv_{kl} \cdot \cos(\theta) \cdot d\theta \cdot d\psi \cdot d\varphi$$

Fortunately, most of these coefficients evaluate to zero and some can be eliminated using symmetries.

The remaining derivatives of *C* to *x* are given by the calibration algorithm and depend on the calibration procedure used, but not on the final measurement. They cannot be calculated algebraically because we do not have a closed representation of the algorithm. They can, however, easily be evaluated numerically for any given calibration procedure and a given α angle. To do this, we evaluate the full calibration algorithm several times with one of the input values offset by a small amount at each run.

In summary, both the measurement error and the influence of the calibration can be written in a simple form using 6 coefficients given for a specific calibration:

$$\overline{\Delta h}^2 = hg \cdot \Delta\tilde{g}^2 + hm \cdot \Delta\tilde{m}^2 + hd \cdot \Delta d^2$$
$$\overline{\Delta v}^2 = vg \cdot \Delta\tilde{g}^2 + vm \cdot \Delta\tilde{m}^2 + vd \cdot \Delta d^2 \qquad 17)$$

The following table shows the values of these coefficients for the measurement errors and some selected calibration procedures:

M: Error of individual measurement.
Ug4: Influence of calibration using 4 unidirectional groups (horizontal directions).
Ug6: Influence of calibration using 6 unidirectional groups (horizontal and vertical directions).
Uga: Influence of calibration using unidirectional groups for all directions.
Kd: Influence of calibration using known directions.
Kd+: Influence of error of known directions.

	α =90°						α =30°					
	hg	*hm*	*hd*	*vg*	*vm*	*vd*	*hg*	*hm*	*hd*	*vg*	*vm*	*vd*
M	0.33	0.67	1.00	1.00	0.00	1.00	2.33	2.67	1.00	1.00	0.00	1.00
Ug4	0.07	0.12	0.08	0.13	0.05	0.08	0.30	0.35	0.12	0.13	0.04	0.11
Ug6	0.05	0.09	0.06	0.11	0.02	0.06	0.26	0.31	0.09	0.11	0.02	0.07
Uga	0.04	0.07	0.05	0.09	0.01	0.04	0.22	0.26	0.07	0.09	0.01	0.04
Kd	0.03	0.06	0.05	0.07	0.01	0.06	0.18	0.22	0.09	0.07	0.01	0.06
Kd+			0.04			0.06			0.06			0.06

The main usage of these numbers is to assure the error induced by the calibration is small compared to the error of the measurement itself. As can be seen, this is achieved even if we use unidirectional groups for the 4 horizontal directions only and use free measurements otherwise (row Ug4). All calibration coefficients are less than 20% of the corresponding coefficients of the measurement errors. Since the errors are squared in the equations, we can be sure the calibration adds less than 10% to the total error.

In the case of measurements of known directions an additional error is introduced because the given directions are themselves not precisely known. The coefficients belonging to these additional errors are given in the row Kd+. It can be seen that known directions do not lead to significantly better results compared to unidirectional groups. Even worse, if the predefined directions are not very precisely known, the additional errors easily outweigh the improvements.

Quantifying the Error

So far, we still have no absolute measure for the resulting error. To get a hint for the achieved precision, we calculate the error value *E* given by equation 4) during evaluation of the calibration algorithm. To relate *E* to the precision, we work out the standard deviation of a set of values d_i consisting of the components of the differences $\vec{g}r_{ij} - \vec{g}t_{ij}$ and $\vec{m}r_{ij} - \vec{m}t_{ij}$ such that:

$$E^2 = \frac{1}{n} \cdot \sum_i \left|\vec{g}r_i - \vec{g}t_i\right|^2 + \left|\vec{m}r_i - \vec{m}t_i\right|^2 = \frac{1}{n} \cdot \sum_i d_i^{\,2}$$

The standard deviation of this set can be computed statistically:

$$S(d_i)^2 = \frac{1}{6n} \cdot \sum_i d_i^{\,2} - \overline{d_i}^2 = \frac{1}{6n} \cdot \sum_i d_i^{\,2} = \frac{1}{6} \cdot E^2$$

Or analytically using error propagation:

$$S(d_i)^2 = \frac{1}{6n} \cdot \sum_i \Delta d_i^{\,2} = \frac{1}{6n} \cdot \sum_{ij} \left(\frac{\partial d_i}{\partial \vec{x}_j} \right)^2 \cdot \Delta \vec{x}_j^{\,2} = \frac{1}{6} \cdot \sum_j \left(\frac{\partial E}{\partial \vec{x}_j} \right)^2 \cdot \Delta \vec{x}_j^{\,2}$$

Comparison of the two representations gives:

$$E^2 = \sum_j \left(\frac{\partial E}{\partial \vec{x}_j} \right)^2 \cdot \Delta \vec{x}_j^{\,2}$$

As above, the derivatives of *E* can be evaluated numerically for any given calibration procedure, resulting in an equation for the error function *E* similar to equations 17):

$$E^2 = eg \cdot \Delta \tilde{g}^2 + em \cdot \Delta \tilde{m}^2 + ed \cdot \Delta d^2$$

This equation does not suffice to calculate the errors exactly but it gives us some constraints and in particular it can be used to a get an upper bound of the relevant errors:

$$\Delta h^2 \le E^2 \cdot \max\left(\frac{hg}{eg}, \frac{hm}{em}, \frac{hd}{ed}\right) \qquad \Delta v^2 \le E^2 \cdot \max\left(\frac{vg}{eg}, \frac{vm}{em}, \frac{vd}{ed}\right)$$

For *hg*, *hm*, *hd*, *vg*, *vm*, and *vd* we have to use the sum of the measurement and calibration errors.

For the standard procedure with 4 unidirectional groups we get the following numbers:

	α =90°						α =30°					
	eg	*em*	*ed*				*eg*	*em*	*ed*			
Ug4	1.46	1.54	0.45				1.50	1.50	0.37			
	hg/eg	*hm/em*	*hd/ed*	*vg/eg*	*vm/em*	*vd/ed*	*hg/eg*	*hm/em*	*hd/ed*	*vg/eg*	*vm/em*	*vd/ed*
Ug4	0.27	0.51	2.38	0.77	0.03	2.38	1.76	2.01	3.01	0.76	0.03	2.99

The table shows a maximal quotient of 3 and therefore we have a maximal error of $\sqrt{3} \cdot E$. In angular measures this corresponds to an error of 1° for an *E* of 1%.

Be aware that this relation must be used with care. It is not a strict limit for the error of each individual measurement. Instead it is a statistical measure for the average error over many measurements. In addition we postulated independent random errors which is not necessarily the case, especially for nonlinearities and systematic errors during calibration.

Conclusions

It can be shown that 3-axis compass/clinometer devices can be calibrated without relying on a given calibration course. This allows recalibration of the device at any time in the field.

The precision degradation introduced by improper calibration measurements can be analyzed mathematically. For the presented standard calibration procedure, the additional errors are negligible compared to the error of the measurement itself.

The algorithm presented is used in the “paperless caving” system [2] to calibrate the DistoX, an all-in-one electronic cave surveying device [3].

Experience shows that calibration must be repeated from time to time to avoid performance degradation due to component drift and aging. In devices using primary batteries, a calibration is needed after each battery change because the battery is unavoidably the main source of magnetic disturbance and new batteries never have exactly the same behaviour as the old ones.

References

[1] Phil Underwood: Calibrating a combined electronic compass/clinometer; Compass Points 37, 5-7; BCRA Cave Surveying Group; http://shetlandattackpony.co.uk.

[2] Beat Heeb: Paperless Caving - An Electronic Cave Surveying System; Proceedings of the IV European Speleological Congress 2008, p. 130-133; http://paperless.bheeb.ch.

[3] Beat Heeb: An All-In-One Electronic Cave Surveying Device**;** CREG Journal 72; BCRA Cave Radio and Electronics Group.

Appendix A: Vector and Matrix Operations Used

Operators:

$\cdot$: Standard Product of scalar numbers

$\circ$: Matrix Product ($(\mathbf{M}\circ\mathbf{N})_{ij} = \sum \mathbf{M}_{ik}\cdot\mathbf{N}_{kj}$, $(\mathbf{M}\circ\vec{a})_i = \sum \mathbf{M}_{ik}\cdot\vec{a}_k$)

$\bullet$: Dot or Scalar Product ($\vec{a}\bullet\vec{b} = \vec{a}^T\circ\vec{b} = \sum \vec{a}_i\cdot\vec{b}_i$)

$\times$: Cross or Vector Product ($\vec{a}\times\vec{b} = \begin{bmatrix}\vec{a}_y\cdot\vec{b}_z - \vec{a}_z\cdot\vec{b}_y & \vec{a}_z\cdot\vec{b}_x - \vec{a}_x\cdot\vec{b}_z & \vec{a}_x\cdot\vec{b}_y - \vec{a}_y\cdot\vec{b}_x\end{bmatrix}$)

$\otimes$: Outer or Kronecker Product ($\vec{a}\otimes\vec{b} = \vec{a}\circ\vec{b}^T$, $(\vec{a}\otimes\vec{b})_{ij} = \vec{a}_i\cdot\vec{b}_j$)

$[,,]$: Scalar Triple Product ($[\vec{a},\vec{b},\vec{c}] = (\vec{a}\times\vec{b})\bullet\vec{c} = (\vec{b}\times\vec{c})\bullet\vec{a} = (\vec{c}\times\vec{a})\bullet\vec{b}$)

$|\;|$: Norm or Length of Vector ($|\vec{a}| = \sqrt{\vec{a}\bullet\vec{a}}$)

$\langle\;\rangle$: NormalizedVector ($\langle\vec{a}\rangle = 1/_{|\vec{a}|}\cdot\vec{a}$)

$\angle$: Angle between two vectors ($\angle\vec{a},\vec{b} = \arccos(\langle\vec{a}\rangle\bullet\langle\vec{b}\rangle)$)

$\|\;\|$: Max Norm of Matrix ($\|\mathbf{M}\| = \max_{ij}(\mathbf{M}_{ij})$)

Rotation matrices:

$$\mathbf{R}_{\vec{x}}(\omega) = \begin{bmatrix}1 & 0 & 0\\ 0 & \cos\omega & -\sin\omega\\ 0 & \sin\omega & \cos\omega\end{bmatrix}\quad \mathbf{R}_{\vec{y}}(\omega) = \begin{bmatrix}\cos\omega & 0 & \sin\omega\\ 0 & 1 & 0\\ -\sin\omega & 0 & \cos\omega\end{bmatrix}\quad \mathbf{R}_{\vec{z}}(\omega) = \begin{bmatrix}\cos\omega & -\sin\omega & 0\\ \sin\omega & \cos\omega & 0\\ 0 & 0 & 1\end{bmatrix}$$

Useful Identities:

$$\vec{a}\times(\vec{b}\times\vec{c}) = \vec{b}\cdot(\vec{a}\bullet\vec{c}) - \vec{c}\cdot(\vec{a}\bullet\vec{b})$$

$$(\vec{a}\times\vec{b})\times\vec{c} = \vec{b}\cdot(\vec{a}\bullet\vec{c}) - \vec{a}\cdot(\vec{b}\bullet\vec{c})$$

$$(\vec{a}\times\vec{b})\bullet(\vec{c}\times\vec{d}) = [\vec{a}\times\vec{b},\vec{c},\vec{d}] = [\vec{a},\vec{b},\vec{c}\times\vec{d}] = ((\vec{a}\times\vec{b})\times\vec{c})\bullet\vec{d} = (\vec{a}\bullet\vec{c})\cdot(\vec{b}\bullet\vec{d}) - (\vec{a}\bullet\vec{d})\cdot(\vec{b}\bullet\vec{c})$$

$$(\mathbf{R}_{\vec{v}}(\omega)\circ\vec{a})\bullet(\mathbf{R}_{\vec{v}}(\omega)\circ\vec{b}) = \vec{a}\bullet\vec{b}$$

$$(\mathbf{R}_{\vec{v}}(\omega)\circ\vec{a})\times(\mathbf{R}_{\vec{v}}(\omega)\circ\vec{b}) = \mathbf{R}_{\vec{v}}(\omega)\circ(\vec{a}\times\vec{b})$$

$$[\mathbf{R}_{\vec{v}}(\omega)\circ\vec{a}, \mathbf{R}_{\vec{v}}(\omega)\circ\vec{b}, \mathbf{R}_{\vec{v}}(\omega)\circ\vec{c}] = [\vec{a},\vec{b},\vec{c}]$$

Derivatives:

$$\nabla_{\vec{v}}|\vec{v}|^2 = 2\cdot\vec{v}$$

$$\nabla_{\mathbf{M}}(\vec{a}^T\circ\mathbf{M}\circ\vec{b}) = \vec{a}\circ\vec{b}^T = \vec{a}\otimes\vec{b}$$

$$\frac{\partial}{\partial\omega}\mathbf{R}_{\vec{v}}(\omega)\circ\vec{a} = \langle\vec{v}\rangle\times(\mathbf{R}_{\vec{v}}(\omega)\circ\vec{a})$$

Appendix B: The Calibration Algorithm

// *Get calibration coefficients from a set of sensor values*

Calibrate($\vec{g}s_i, \vec{m}s_i \rightarrow \mathbf{G}, \mathbf{M}, \vec{g}d, \vec{m}d$) {

$avg\vec{g}s = av\vec{m}s = \vec{0}$; $\mathbf{avG} = \mathbf{avM} = \mathbf{0}$; $sa = ca = 0$;

for (all i) {

$sa = sa + |\vec{g}s_i \times \vec{m}s_i|$; // *sum up sine of angle*

$ca = ca + \vec{g}s_i \bullet \vec{m}s_i$; // *sum up cosine of angle*

$av\vec{g}s = av\vec{g}s + \vec{g}s_i$; // *sum up g values*

$av\vec{m}s = av\vec{m}s + \vec{m}s_i$; // *sum up m values*

$\mathbf{avG} = \mathbf{avG} + \vec{g}s_i \otimes \vec{g}s_i$; // *sum up outer product of g*

$\mathbf{avM} = \mathbf{avM} + \vec{m}s_i \otimes \vec{m}s_i$; // *sum up outer product of m*

}

$av\vec{g}s = 1/n \cdot av\vec{g}s$; $av\vec{m}s = 1/n \cdot av\vec{m}s$; // *build average*

$\mathbf{avG} = 1/n \cdot \mathbf{avG}$; $\mathbf{avM} = 1/n \cdot \mathbf{avM}$;

$\alpha = \arctan(sa, ca)$; // *first estimate of* α

$\mathbf{Gi} = \text{inverse}(\mathbf{avG} - av\vec{g}s \otimes av\vec{g}s)$; // *inverted matrices*

$\mathbf{Mi} = \text{inverse}(\mathbf{avM} - av\vec{m}s \otimes av\vec{m}s)$;

$\mathbf{G} = \mathbf{M} = \mathbf{1}$; $\vec{g}d = \vec{m}d = \vec{0}$; // *first estimate of coefficients*

do {

for (all i) { // *get result vectors using current cofficients*

$\vec{g}r_i = \mathbf{G} \circ \vec{g}s_i + \vec{g}d$;

$\vec{m}r_i = \mathbf{M} \circ \vec{m}s_i + \vec{m}d$;

}

$sa = ca = 0$;

for (all unidirectional groups k) {

$\vec{g}c = \vec{m}c = \vec{0}$;

for (all i in group k) { // *adapt vectors to first vector*

AdaptPhi($\vec{g}r_i, \vec{m}r_i, \vec{g}r_{ik}, \vec{m}r_{ik} \rightarrow \vec{g}a, \vec{m}a$);

$\vec{g}c = \vec{g}c + \vec{g}a$; // *sum up vectors*

$\vec{m}c = \vec{m}c + \vec{m}a$;

}

GetTrueVectors($\vec{g}c, \vec{m}c, \alpha \rightarrow \vec{g}p, \vec{m}p$); // *get true vector*

$sa = sa + |\vec{m}c \times \vec{g}p|$; // *sum up for* α *caculation*

$ca = ca + \vec{m}c \bullet \vec{g}p$;

for (all i in group k) { // *get individual roll angles*

AdaptPhi($\vec{g}p, \vec{m}p, \vec{g}r_i, \vec{m}r_i \rightarrow \vec{g}t_i, \vec{m}t_i$);

}

}

for (all free measurements i) {

GetTrueVectors$(\vec{g}r_i, \vec{m}r_i, \alpha \rightarrow \vec{g}t_i, \vec{m}t_i)$; // *get true vector*

$sa = sa + |\vec{m}r_i \times \vec{g}t_i|$; // *sum up for α caculation*

$ca = ca + \vec{m}r_i \bullet \vec{g}t_i$;

}

$\alpha = \arctan(sa, ca)$; // *get new α*

$\mathbf{avG} = \mathbf{avM} = \mathbf{0}$; $av\vec{g}t = av\vec{m}t = \vec{0}$;

for (all i) {

$av\vec{g}t = av\vec{g}t + \vec{g}t_i$; // *sum up true vectors*

$av\vec{m}t = av\vec{m}t + \vec{m}t_i$;

$\mathbf{avG} = \mathbf{avG} + \vec{g}t_i \otimes \vec{g}s_i$; // *sum up outer products*

$\mathbf{avM} = \mathbf{avM} + \vec{m}t_i \otimes \vec{m}s_i$;

}

$av\vec{g}t = 1/n \cdot av\vec{g}t$; $av\vec{m}t = 1/n \cdot av\vec{m}t$; // *build average*

$\mathbf{avG} = 1/n \cdot \mathbf{avG}$; $\mathbf{avM} = 1/n \cdot \mathbf{avM}$;

$\mathbf{oldG} = \mathbf{G}$; $\mathbf{G} = (\mathbf{avG} - av\vec{g}t \otimes av\vec{g}s) \circ \mathbf{Gi}$; // *get new matrices*

$\mathbf{oldM} = \mathbf{M}$; $\mathbf{M} = (\mathbf{avM} - av\vec{m}t \otimes av\vec{m}s) \circ \mathbf{Mi}$;

$\mathbf{G}_{yz} = \mathbf{G}_{zy} = 1/2 \cdot (\mathbf{G}_{yz} + \mathbf{G}_{zy})$; // *enforce symmetry*

$\vec{g}d = (av\vec{g}t - \mathbf{G} \circ av\vec{g}s)$; // *get new vectors*

$\vec{m}d = (av\vec{m}t - \mathbf{M} \circ av\vec{m}s)$;

} while $(\max(\|\mathbf{G} - \mathbf{oldG}\|, \|\mathbf{M} - \mathbf{oldM}\|) > 10^{-6})$ // *termination condition*

}

// *Get estimated true vectors for given result vectors*

GetTrueVectors$(\vec{g}r, \vec{m}r, \alpha \rightarrow \vec{g}t, \vec{m}t)$ {

$\vec{n} = \text{normalized}(\vec{g}r \times \vec{m}r)$; // *plane normal*

$\vec{g}t = \text{normalized}(\vec{g}r + \vec{m}r \cdot \cos(\alpha) + (\vec{m}r \times \vec{n}) \cdot \sin(\alpha))$;

$\vec{m}t = \vec{g}t \cdot \cos(\alpha) + (\vec{n} \times \vec{g}t) \cdot \sin(\alpha)$;

}

// *Turn $\vec{g}a / \vec{m}a$ to the roll angle of $\vec{g}b / \vec{m}b$*

AdaptPhi$(\vec{g}a, \vec{m}a, \vec{g}b, \vec{m}b \rightarrow \vec{g}x, \vec{m}x)$ {

$s = \vec{g}a_y \cdot \vec{g}b_z - \vec{g}a_z \cdot \vec{g}b_y + \vec{m}a_y \cdot \vec{m}b_z - \vec{m}a_z \cdot \vec{m}b_y$;

$c = \vec{g}a_y \cdot \vec{g}b_y + \vec{g}a_z \cdot \vec{g}b_z + \vec{m}a_y \cdot \vec{m}b_y + \vec{m}a_z \cdot \vec{m}b_z$;

$\delta = \arctan(s, c)$; // *roll angle difference*

$\vec{g}x = T_{\vec{x}}(\delta) \circ \vec{g}a$;

$\vec{m}x = T_{\vec{x}}(\delta) \circ \vec{m}a$;

}