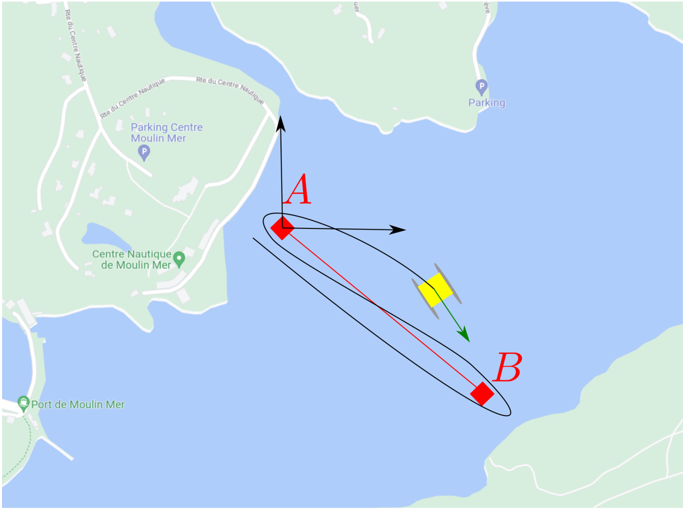
Calibrating magnetomers and accelerometers

L. Jaulin

February 2025
Guerlédan

1. Calibration of the magnetometers


Rte du Centre Nautique
Rte du Centre Nautique
Rte du Centre Nautique
Parking Centre Moulin Mer
Parking
Centre Nautique de Moulin Mer
Port de Moulin Mer
A
B

In Brest, the inclination of the magnetic field is: $I = 64°$.
The declination is 1.4° and its intensity is : $\beta = 46.0\mu T$.
See: franceipgp_Jussieu.html

For a dipole,

$$I = \arctan(2\tan\lambda)$$

For simplicity, we assume that the declination is 0, *i.e.*, the magnetic field is pointing toward the North.

We assume that

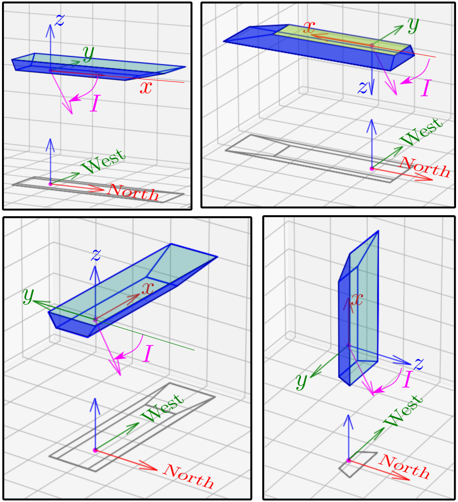
$$\mathbf{y} = \mathbf{A}^{-1}(\mathbf{x} + \mathbf{b})$$

where $\mathbf{x} \in \mathbb{R}^3$ is the vector of magnetic measurements, and $\mathbf{y}$ is the corrected vector.

The calibration step searches for an estimation of $\mathbf{A}$ and $\mathbf{b}$.

We take four measurements $\mathbf{x}_N, \mathbf{x}_S, \mathbf{x}_W, \mathbf{x}_U$, with the robot

- Direction N : The robot is laid flat toward the North
- Direction S : The robot is laid toward the South, upside down
- Direction W : The robot is laid flat toward the West
- Direction U : The robot is set vertically with its z axis toward the North


z
y
x
I
West
North
x
y
z
I
West
North
z
x
y
I
West
North
x
y
z
I
West
North

The four corresponding transformation matrices are

$$\mathbf{R}_N = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad \mathbf{R}_S = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

$$\mathbf{R}_W = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad \mathbf{R}_U = \begin{pmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

The corrected magnetic field should be

$$\mathbf{y}_N = \begin{pmatrix} \beta \cdot \cos I \\ 0 \\ -\beta \cdot \sin I \end{pmatrix}, \qquad \mathbf{y}_S = \mathbf{R}_S^{\mathsf{T}} \cdot \mathbf{y}_N = \begin{pmatrix} -\beta \cdot \cos I \\ 0 \\ \beta \cdot \sin I \end{pmatrix}$$

$$\mathbf{y}_W = \mathbf{R}_W^{\mathsf{T}} \cdot \mathbf{y}_N = \begin{pmatrix} 0 \\ -\beta \cdot \cos I \\ -\beta \cdot \sin I \end{pmatrix}, \quad \mathbf{y}_U = \mathbf{R}_U^{\mathsf{T}} \cdot \mathbf{y}_N = \begin{pmatrix} -\beta \cdot \sin I \\ 0 \\ \beta \cdot \cos I \end{pmatrix}$$

where $\beta = 46\mu T$.

Since

$$\begin{aligned} \mathbf{0} &= \mathbf{y}_N + \mathbf{y}_S \\ &= \mathbf{A}^{-1}(\mathbf{x}_N + \mathbf{b}) + \mathbf{A}^{-1}(\mathbf{x}_S + \mathbf{b}) \end{aligned}$$

we get

$$(\mathbf{x}_N + \mathbf{b}) + (\mathbf{x}_S + \mathbf{b}) = \mathbf{0}$$

i.e.,

$$\mathbf{b} = -\frac{1}{2}(\mathbf{x}_N + \mathbf{x}_S).$$

Moreover

$$\begin{aligned}\mathbf{A}\mathbf{y}_N &= \mathbf{x}_N + \mathbf{b}\\ \mathbf{A}\mathbf{y}_W &= \mathbf{x}_W + \mathbf{b}\\ \mathbf{A}\mathbf{y}_U &= \mathbf{x}_U + \mathbf{b}\end{aligned}$$

yields

$$\mathbf{A}\cdot\underbrace{\begin{pmatrix}\mathbf{y}_N & \mathbf{y}_W & \mathbf{y}_U\end{pmatrix}}_{\mathbf{Y}} = \underbrace{\begin{pmatrix}\mathbf{x}_N+\mathbf{b} & \mathbf{x}_W+\mathbf{b} & \mathbf{x}_U+\mathbf{b}\end{pmatrix}}_{\mathbf{X}}$$

i.e.,

$$\mathbf{A} = \mathbf{X} \cdot \mathbf{Y}^{-1}$$

Algorithm MagCalib(In: $\mathbf{x}_N, \mathbf{x}_S, \mathbf{x}_W, \mathbf{x}_U$; Out :$\mathbf{A}, \mathbf{b}$)

$I = 64 \cdot \frac{\pi}{180}; \beta = 46.0 \cdot 10^{-6} T.$

$\mathbf{b} = -\frac{1}{2}\left(\mathbf{x}_N + \mathbf{x}_S\right)$

$$\mathbf{Y} = \beta \cdot \begin{pmatrix} \cos I & 0 & -\sin I \\ 0 & -\cos I & 0 \\ -\sin I & -\sin I & \cos I \end{pmatrix}$$

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_N + \mathbf{b} & \mathbf{x}_W + \mathbf{b} & \mathbf{x}_U + \mathbf{b} \end{pmatrix}$$

$\mathbf{A} = \mathbf{X} \cdot \mathbf{Y}^{-1}$

Return $\mathbf{A}$, $\mathbf{b}$

2. Calibration of the accelerometers

We assume that

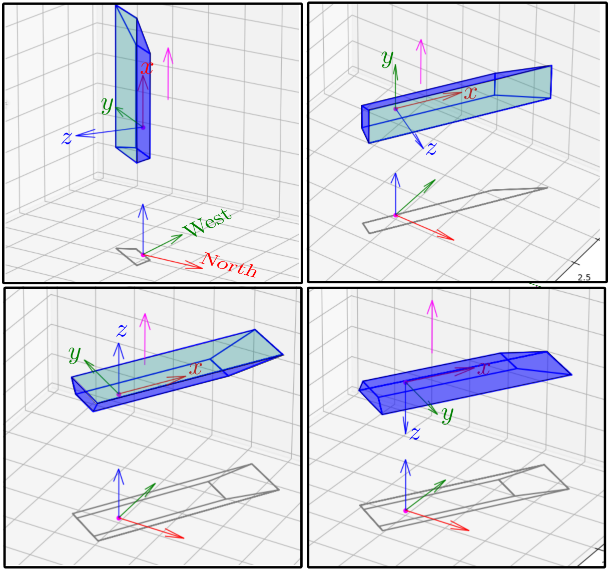
$$\mathbf{y} = \mathbf{A}^{-1}(\mathbf{x} + \mathbf{b})$$

where $\mathbf{x} \in \mathbb{R}^3$ is the acceleration vector which is measured, and $\mathbf{y}$ is gravity vector (oriented upward).
We assume the boat static so that the acceleration corresponds to the gravity.

The calibration step searches for an estimation of $\mathbf{A}$ and $\mathbf{b}$.

We take four measurements $\mathbf{x}_x, \mathbf{x}_y, \mathbf{x}_z, \mathbf{x}_{-z},$, with the robot. We do not care where the North is.

- Direction x : The robot is set vertically with its x axis looking upward
- Direction y : The robot is set with roll its y axis upward
- Direction z : The robot is laid flat with its z axis upward
- Direction $-z$: The robot is laid flat upside down with its z axis downward


x
y
z
West
North
y
x
z
2.5
z
y
x
x
y
z

The corresponding corrected acceleration vectors should be

$$\mathbf{y}_x = \begin{pmatrix} \beta \\ 0 \\ 0 \end{pmatrix}, \mathbf{y}_y = \begin{pmatrix} 0 \\ \beta \\ 0 \end{pmatrix}, \mathbf{y}_z = \begin{pmatrix} 0 \\ 0 \\ \beta \end{pmatrix}, \mathbf{y}_{-z} = \begin{pmatrix} 0 \\ 0 \\ -\beta \end{pmatrix}$$

where $\beta = 9.8 m \cdot s^{-2}$.

Since

$$\begin{aligned} \mathbf{0} &= \mathbf{y}_z + \mathbf{y}_{-z} \\ &= \mathbf{A}^{-1}(\mathbf{x}_z + \mathbf{b}) + \mathbf{A}^{-1}(\mathbf{x}_{-z} + \mathbf{b}) \end{aligned}$$

we get

$$(\mathbf{x}_z + \mathbf{b}) + (\mathbf{x}_z + \mathbf{b}) = \mathbf{0}$$

i.e.,

$$\mathbf{b} = -\frac{1}{2}(\mathbf{x}_z + \mathbf{x}_{-z}).$$

Moreover

$$\begin{aligned} \mathbf{A}\mathbf{y}_x &= \mathbf{x}_z + \mathbf{b} \\ \mathbf{A}\mathbf{y}_y &= \mathbf{x}_y + \mathbf{b} \\ \mathbf{A}\mathbf{y}_z &= \mathbf{x}_z + \mathbf{b} \end{aligned}$$

yields

$$\mathbf{A} \cdot \underbrace{\begin{pmatrix} \mathbf{y}_x & \mathbf{y}_y & \mathbf{y}_z \end{pmatrix}}_{\beta \mathbf{I}} = \underbrace{\begin{pmatrix} \mathbf{x}_x + \mathbf{b} & \mathbf{x}_y + \mathbf{b} & \mathbf{x}_z + \mathbf{b} \end{pmatrix}}_{\mathbf{X}}$$

Thus

$$\mathbf{A} = \frac{1}{\beta}\mathbf{X}$$

	Algorithm MagAccelero(In: $\mathbf{x}_x, \mathbf{x}_y, \mathbf{x}_z, \mathbf{x}_{-z}$; Out :$\mathbf{A}, \mathbf{b}$)
1	$\beta = 9.8 m \cdot s^{-2}$
2	$\mathbf{b} = -\frac{1}{2}(\mathbf{x}_z + \mathbf{x}_{-z})$
3	$\mathbf{X} = \begin{pmatrix} \mathbf{x}_x + \mathbf{b} & \mathbf{x}_y + \mathbf{b} & \mathbf{x}_z + \mathbf{b} \end{pmatrix}$
4	$\mathbf{A} = \frac{1}{\beta}\mathbf{X}$
5	Return $\mathbf{A}$, $\mathbf{b}$

3. IMU

The boat is moving in the waves.
An IMU estimates the Euler angles from a magnetometer and an accelerometer.

These Euler angles φ, θ, ψ have to be estimated from

1. the magnetic field $\mathbf{y}_1$ and
2. its acceleration $\mathbf{a}_1$, both given in the boat frame.

Estimate the vertical

Assumption. In average the accelerometers returns the gravity vector $\mathbf{g}_1$ is the robot frame, i.e.,

$$\mathbf{a}_1 = \mathbf{g}_1 + \beta$$

where β is a centered noise.
The assumption is false if the boat turns around a buoy for long.

The sensors return $\mathbf{a}_1$ and ω_1 in the robot frame.
The gravity vector $\mathbf{g}_1$ returns the vertical.

We have

$$\mathbf{g}_0 = \mathbf{R}\mathbf{g}_1$$

Differentiate with respect to t:

$$\mathbf{0} = \dot{\mathbf{R}}\mathbf{g}_1 + \mathbf{R}\dot{\mathbf{g}}_1$$

Thus

$$\begin{aligned} \dot{\mathbf{g}}_1 &= -\mathbf{R}^\mathsf{T}\dot{\mathbf{R}}\mathbf{g}_1 \\ &= -\omega_1 \wedge \mathbf{g}_1 \end{aligned}$$

We have

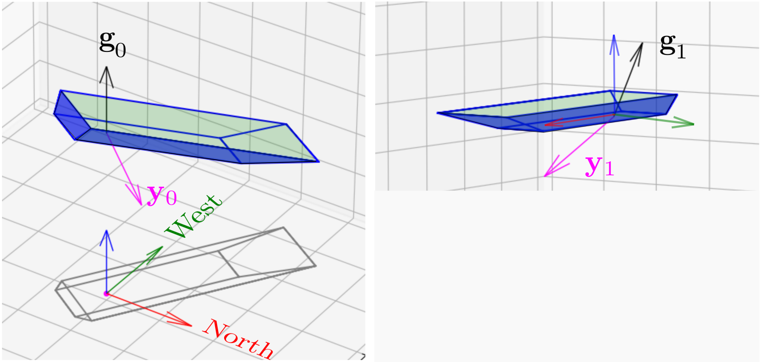
$$\begin{array}{rcll} \mathbf{g}_1(k+1) & = & \underbrace{(\mathbf{I} - dt\boldsymbol{\omega}_1(k)\wedge)}_{=\mathbf{A}(k)} \cdot \mathbf{g}_1(k) & \text{(prediction)} \\ \mathbf{a}_1(k) & = & \mathbf{g}_1(k) + \beta(k) & \text{(observation)} \end{array}$$

with a large variance for $\beta(k)$, which means that we measure $\mathbf{g}_1$ (through the acceleration $\mathbf{a}_1$), only in average.

A Luenberger observer could be

$$\hat{\mathbf{g}}_1(k+1) = \lambda \cdot (\mathbf{I} - dt\omega_1(k)\wedge) \cdot \hat{\mathbf{g}}_1(k) + (1-\lambda) \cdot \mathbf{a}_1(k+1)$$

with $\lambda = 0.99$.



We now assume that we are able to have a good estimation of $\mathbf{g}_1$

For simplicity, we assume that $\mathbf{y}$ and $\mathbf{g}$ have been normalized, *i.e.*, their norm are both equal to 1. Moreover, we assume that sensors are well calibrated.
In $\mathscr{R}_0$, we have

$$\mathbf{y}_0 = \begin{pmatrix} \cos I \\ 0 \\ -\sin I \end{pmatrix}, \ \mathbf{g}_0 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Step 1. From $\mathbf{g}_1$, we get an estimation of the bank φ (roll angle) and the elevation θ (pitch angle).

We have

$$\hat{\varphi} = \arcsin\left((0,1,0)\cdot \mathbf{g}_1\right)$$

and

$$\hat{\theta} = -\arcsin\left((1,0,0)\cdot \mathbf{g}_1\right)$$

Step 2. For the heading, we first compute the rotation which leads our boat to the nearest horizontal orientation.

If $\mathbf{u}$ and $\mathbf{v}$ are two unit vectors, the rotation $\mathbf{R}$ which minimizes $\|\mathbf{R}-\mathbf{I}\|$ and transforms $\mathbf{u}$ to $\mathbf{v}$ is given by

$$\mathbf{R}_{uv}(\mathbf{u},\mathbf{v}) = \mathrm{Exp}\left(\arccos(\mathbf{u}^\mathsf{T}\cdot\mathbf{v})\cdot\frac{\mathbf{u}\wedge\mathbf{v}}{\|\mathbf{u}\wedge\mathbf{v}\|}\right)$$

The rotation which transforms $\mathbf{g}_1$ to $\mathbf{g}_0$ is given by

$$\mathbf{R}_h = \mathbf{R}_{uv}(\mathbf{g}_1,\mathbf{g}_0).$$

Step 3. Provide an estimation $\hat{\psi}$ of the heading angle ψ.

We form the vector

$$\mathbf{y}_h = \mathbf{R}_h \mathbf{y}_1$$

which corresponds to the magnetic field measured by the horizontal avatar.
The heading is then obtained by

$$\hat{\psi} = -\mathrm{atan2}\left(y_{h2}, y_{h1}\right).$$

Algorithm EulerAngles(In: $\mathbf{a}_1, \mathbf{y}_1$; out: $\hat{\varphi}, \hat{\theta}, \hat{\psi}$)

$$\mathbf{g}_0 = (0,0,1)^\mathsf{T}, \mathbf{g}_1^n = \frac{\mathbf{a}_1}{\|\mathbf{a}_1\|}, \mathbf{y}_1^n = \frac{\mathbf{y}_1}{\|\mathbf{y}_1\|}$$

$$\hat{\varphi} = \arcsin\left((0,1,0)\cdot \mathbf{g}_1^n\right)$$

$$\hat{\theta} = -\arcsin\left((1,0,0)\cdot \mathbf{g}_1^n\right)$$

$$\mathbf{R}_h = \mathbf{R}_{uv}(\mathbf{g}_1^n, \mathbf{g}_0)$$

$$\mathbf{y}_h = \mathbf{R}_h \mathbf{y}_1^n$$

$$\hat{\psi} = -\mathrm{atan2}(y_{h2}, y_{h1})$$

With magnetometers, accelerometers and without gyro

Algorithm EulerAngles(In: $\mathbf{a}_1, \mathbf{y}_1, \omega_1, \hat{\mathbf{g}}_1$; out: $\hat{\varphi}, \hat{\theta}, \hat{\psi}, \hat{\mathbf{g}}_1$)

$\mathbf{g}_0 = (0,0,1)^\mathsf{T}$, $\mathbf{y}_1^n = \frac{\mathbf{y}_1}{\|\mathbf{y}_1\|}$

$\hat{\mathbf{g}}_1 = \lambda \cdot (\mathbf{I} - dt\omega_1 \wedge) \cdot \hat{\mathbf{g}}_1 + (1-\lambda) \cdot \mathbf{a}_1$, with $\lambda \simeq 1$

$\mathbf{g}_1^n = \frac{\hat{\mathbf{g}}_1}{\|\hat{\mathbf{g}}_1\|}$

$\hat{\varphi} = \arcsin\left((0,1,0) \cdot \mathbf{g}_1^n\right)$

$\hat{\theta} = -\arcsin\left((1,0,0) \cdot \mathbf{g}_1^n\right)$

$\mathbf{R}_h = \mathbf{R}_{uv}(\mathbf{g}_1^n, \mathbf{g}_0)$

$\mathbf{y}_h = \mathbf{R}_h \mathbf{y}_1^n$

$\hat{\psi} = -\mathrm{atan2}(y_{h2}, y_{h1})$

With magnetometers, accelerometers and gyro

4. Magnetometer autocalibration

Magnetometer autocalibration is the process of determining hard- and soft-iron calibration parameters from arbitrary 3D orientation data, without requiring known attitude or external references.

Equivalently, we want to find a transformation that converts the distorted ellipsoid into a sphere centered at the origin, to correct for hard- and soft-iron effects, without taking care of the rotation part which can be known in advance.

We have many points $(x_1(i), x_2(i), x_3(i))$, $i \geq 1$, collected by the magnetometer.

All belong to an ellipse:

$$\underbrace{p_1x_1^2 + p_2x_1^2 + p_3x_1^2 + p_4x_1x_2 + p_5x_1x_3 + p_6x_2x_3 + p_7x_1 + p_8x_2 + p_9x_3}_{\mathbf{f_p}(\mathbf{x})} = 1.$$

i.e.,

$$\left(x_1^2|x_2^2|x_3^2|x_1x_2|x_1x_2|x_1x_3|x_2x_3|x_1|x_2|x_3\right) \cdot \mathbf{p} = 1$$

We should have

$$\underbrace{\begin{pmatrix} x_1^2(1) & x_2^2(1) & x_3^2(1) & x_1(1)x_2(1) & x_1(1)x_3(1) & x_2(1)x_3(1) & x_1(1) & x_2(\\ x_1^2(2) & x_2^2(2) & x_3^2(2) & x_1(2)x_2(2) & x_1(2)x_3(2) & x_2(2)x_3(2) & x_1(2) & x_2(\\ x_1^2(3) & x_2^2(3) & x_3^2(3) & x_1(3)x_2(3) & x_1(3)x_3(3) & x_2(3)x_3(3) & x_1(3) & x_2(\\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}}_{\mathbf{M}}$$

A least-square solution is

$$\mathbf{p} = \left(\mathbf{M}^{\mathsf{T}}\mathbf{M}\right)^{-1} \cdot \mathbf{M}^{\mathsf{T}} \cdot \mathbf{1}$$

The ellipsoid can be written in matrix form

$$\underbrace{(\mathbf{x}-\mathbf{b})^{\mathsf{T}}\mathbf{Q}(\mathbf{x}-\mathbf{b})}_{\mathbf{f_p}(\mathbf{x})} = 1$$

The gradient for both expressions for $\mathbf{f_p}(\mathbf{x})$ should be equal.

$$\begin{aligned}\frac{\partial \mathbf{f_p}(\mathbf{x})}{\partial \mathbf{x}} &= p_1x_1^2 + p_2x_2^2 + p_3x_3^2 + p_4x_1x_2 + p_5x_1x_3 + p_6x_2x_3 + p_7x_1 + p_8x \\ &= (2p_1x_1 + p_4x_2 + p_5x_3 + p_7 | p_4x_1 + 2p_2x_2 + p_6x_3 + p_8 | p_5x_1 + p_6x_2 \\ &= (x_1|x_2|x_3)\begin{pmatrix} 2p_1 & p_4 & p_5 \\ p_4 & 2p_2 & p_6 \\ p_5 & p_6 & 2p_3 \end{pmatrix} + (p_7|p_8|p_9)\end{aligned}$$

And also

$$
\begin{aligned}
\frac{\partial \mathbf{f_p}(\mathbf{x})}{\partial \mathbf{x}} &= \frac{\partial (\mathbf{x}-\mathbf{b})^\mathsf{T}\mathbf{Q}(\mathbf{x}-\mathbf{b})}{\partial \mathbf{x}} \\
&= 2(\mathbf{x}-\mathbf{b})^\mathsf{T}\mathbf{Q} \\
&= 2\mathbf{x}^\mathsf{T}\mathbf{Q} - 2\mathbf{b}^\mathsf{T}\mathbf{Q}
\end{aligned}
$$

By identification, we get

$$\mathbf{Q} = \frac{1}{2}\begin{pmatrix} 2p_1 & p_4 & p_5 \\ p_4 & 2p_2 & p_6 \\ p_5 & p_6 & 2p_3 \end{pmatrix}$$

and

$$\mathbf{b} = -\frac{1}{2}\mathbf{Q}^{-1} \cdot \begin{pmatrix} p_7 \\ p_8 \\ p_9 \end{pmatrix}$$

To transform the ellipse to a sphere we take the transformation

$$\mathbf{y} = \sqrt{\mathbf{Q}} \cdot (\mathbf{x} - \mathbf{b})$$