Determining *Tukka* for Attitude Determination Algorithm[*]

Rishav
ORION Space

June 27, 2021

Contents

[*] We invented the word ourselves. The term *Tukka* refers to the test input values used to validate any algorithm. This word was first used with this meaning in June 21, 2021, on one of the weekly meetings of team attending the *FloatSat* course. We have agreed to adopt the word ever since. *Note: Tukka is a noun and can be used for both singular and plural cases.*

1 Background

After implementation of attitude determination algorithm, it is very important to check if the software is working as expected. In order to do so, we can use some test data and see if it produces expected result.

If we consider the algorithm for tilt compensated heading determination, the input to the algorithm is the sensor readings of magnetometer $\boldsymbol{m}^B$ and accelerometer $\boldsymbol{a}^B$ and the output is tilt compensated yaw $\tilde{\psi}$. One of the ways of testing the implementation of this algorithm would be to take two arbitrary set of readings that represents $\boldsymbol{m}^B$ and $\boldsymbol{a}^B$ and then share the output $\tilde{\psi}$ with other team members. It is resonable to assume that the implementation is correct if every one get the result of $\tilde{\psi}$. But this approach is doomed to fail if every one in the team made same error in their software or if there is no team to test with.

This document presents a way to generate *Tukka* (i.e. $\boldsymbol{m}^B$ and $\boldsymbol{a}^B$) and see if the implementation is correct using more formal process. This method is applicable to any attitude determination method that uses vector measurements.

2 The Algorithm

2.1 Finding vectors in NED frame

We know that the rotation matrix $\boldsymbol{Q}(\psi, \theta, \phi)$ transforms the vectors in NED frame to vectors in body frame as follows.

$$\begin{aligned} \boldsymbol{a}^B &= \boldsymbol{Q}\,\boldsymbol{a}^N \\ \boldsymbol{m}^B &= \boldsymbol{Q}\,\boldsymbol{m}^N \end{aligned} \tag{1}$$

Where $\boldsymbol{a}^N$ and $\boldsymbol{m}^N$ are the acceleration and magnetic field intensity vectors in NED frame. The *Tukka* that we are looking for are the terms in left hand sides of the above equations which we can find if we can know values of $\boldsymbol{Q}$, $\boldsymbol{a}^N$ and $\boldsymbol{m}^N$

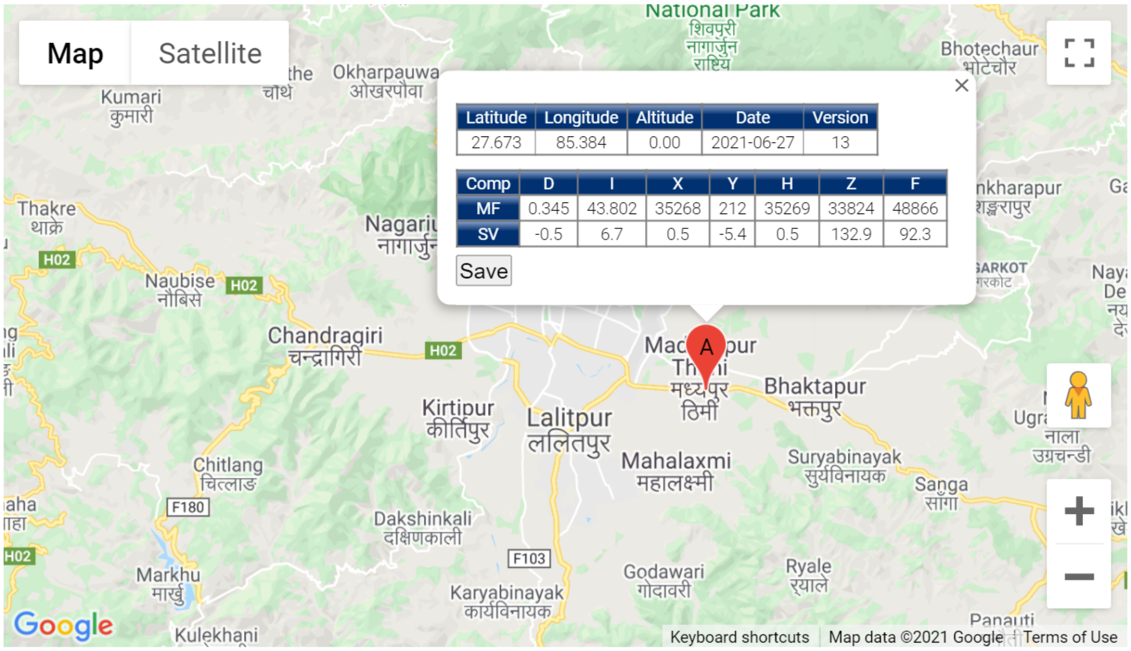
Based on our location on Earth, we can determine $\boldsymbol{a}^N$ and $\boldsymbol{m}^N$ because they can be written as follows.

$$\boldsymbol{a}^N = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \quad \text{and} \quad \boldsymbol{m}^N = \begin{bmatrix} \cos\delta \\ 0 \\ \sin\delta \end{bmatrix} \tag{2}$$

where, g is acceleration due to gravity and δ is angle of inclination of magnetic field.

It is fair to conclude that $\boldsymbol{a}^N$ and $\boldsymbol{m}^N$ are functions of g and δ respectively. So, we can determine these vectors based on where we are right now on the Earth. For instance, currently I am in Bhaktapur so I know that

$$\begin{aligned} &g = 9.8m/s^2, \quad \text{and} \\ &\delta = 43.802^\circ = 0.7675992957 \text{ radians} \end{aligned}$$



Latitude	Longitude	Altitude	Date	Version
27.673	85.384	0.00	2021-06-27	13

Comp	D	I	X	Y	H	Z	F
MF	0.345	43.802	35268	212	35269	33824	48866
SV	-0.5	6.7	0.5	-5.4	0.5	132.9	92.3

Figure 1: Geomag website allows us to find the value of inclination of magnetic field for given location.

The magnetic inclination for any location on Earth can be found from Geomag website.

2.2 Finding *Tukka* and testing software

Now that we have determined the vectors in NED frame, all we need to do is to use arbitrary roll ϕ, pitch ψ and yaw ψ to find $\boldsymbol{Q}$. Actually, later we will compare if output of our implementation $\tilde{\psi}$ equals to the ψ that we used now. We can use following equation to find $\boldsymbol{Q}$.

$$\boldsymbol{Q}(\psi,\theta,\phi) = \begin{bmatrix} \mathrm{c}\psi & \mathrm{c}\theta\,\mathrm{s}\psi & -\mathrm{s}\theta \\ \mathrm{s}\phi\,\mathrm{s}\theta\,\mathrm{c}\psi - \mathrm{c}\theta\,\mathrm{s}\psi & \mathrm{s}\phi\,\mathrm{s}\theta\,\mathrm{s}\psi + \mathrm{c}\phi\,\mathrm{c}\psi & \mathrm{s}\phi\,\mathrm{c}\theta \\ \mathrm{c}\phi\,\mathrm{s}\theta\,\mathrm{c}\psi + \mathrm{s}\phi\,\mathrm{s}\psi & \mathrm{c}\phi\,\mathrm{c}\theta\,\mathrm{s}\psi - \mathrm{s}\phi\,\mathrm{c}\psi & \mathrm{c}\phi\,\mathrm{c}\theta \end{bmatrix} \tag{3}$$

Now we know all the paramaters to find $\boldsymbol{m}^B$ and $\boldsymbol{a}^B$ from Eqn. (1).

Now this is where the advantage of this process comes into play. We can assign arbitrary ψ, θ and ϕ and compute $\boldsymbol{Q}$ using Eqn. (3). Using this $\boldsymbol{Q}$ we find input *Tukka* to our algorithm (i.e. $\boldsymbol{m}^B$ and $\boldsymbol{a}^B$). We will know if our algorithm is working if the value of tilt compensated heading computed by our algorithm is equal to the value of yaw that we assigned to find $\boldsymbol{Q}$. i.e.

$$\tilde{\psi} = \psi \tag{4}$$

3 Example

Let us suppose, from where I am right now, that $g = 9.8$ m/s^2 and $\delta = 43.802° = 0.7675992957$ radians. Similarly, let us imagine that we choose to assign the orientation of our *FloatSat* as $\psi = \pi/4$, $\theta = \pi/8$ and $\theta = \pi/4$ radians. Now, following steps are to be followed to test the implementation tilt compensation algorithm.

1. Use Eq. (3) to find $\boldsymbol{Q}$.
2. Establish the values of $\boldsymbol{a}^N$ and $\boldsymbol{m}^N$ using g and δ using Eq. (2).
3. Finally, use Eqn. (1) to find $\boldsymbol{m}^B$ and $\boldsymbol{a}^B$ i.e. *Tukka* for the tilt compensation algorithm.
4. Use these $\boldsymbol{m}^B$ and $\boldsymbol{a}^B$ to determine $\tilde{\psi}$ using tilt compensation algorithm.
5. If $\tilde{\psi} = \psi = \pi/4$, the software implementation has passed this test.