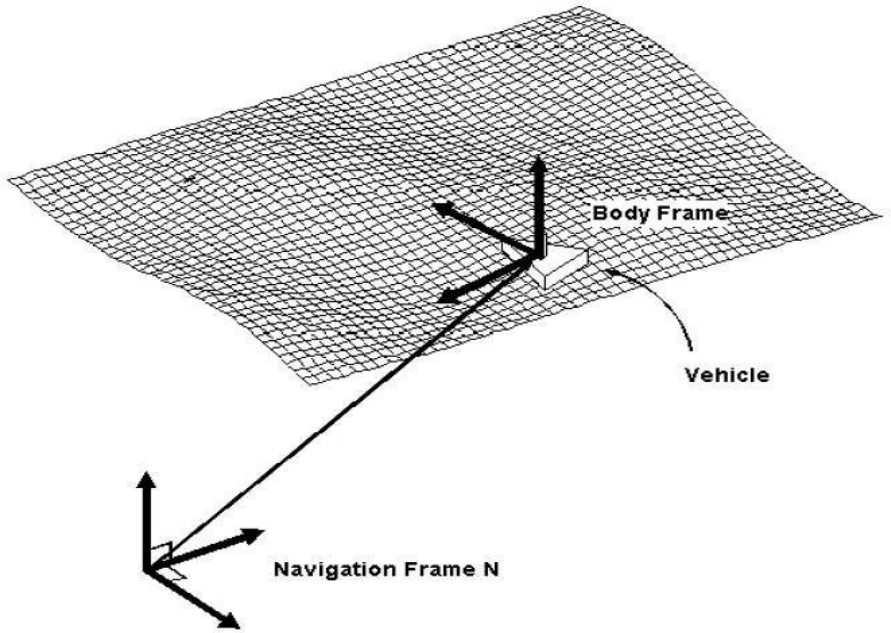
Coordinate frames

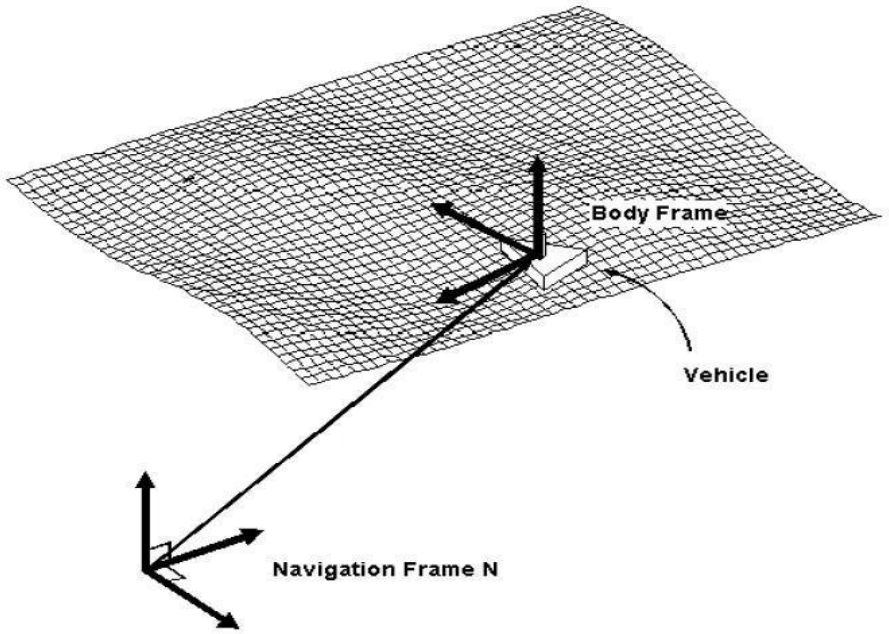
We have interest on two coordinate frames. A local one, defined respect to the platform's body; and a "global" one, which is defined respect to certain fixed reference



We call the global frame "navigation coordinate frame".

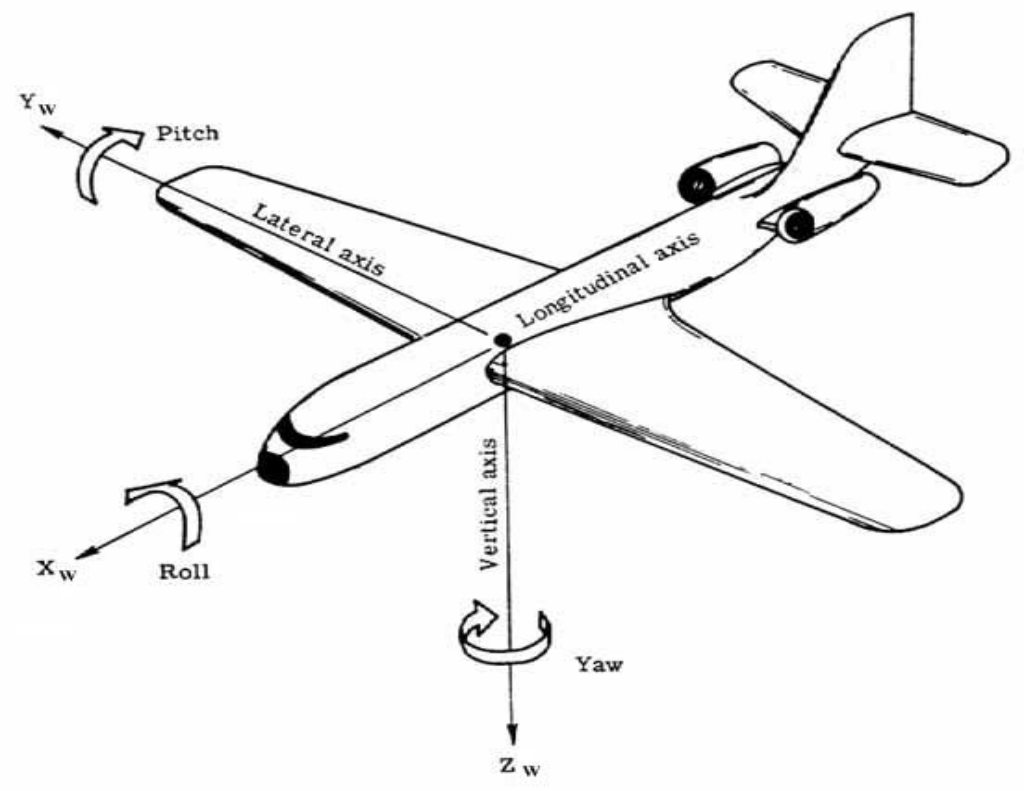
The local one, moves with the platform.

Many sensors that are installed on board platforms, do express their measurements using the “local coordinate frame”. Usual sensors of that type: RGB Cameras, Depth cameras, LiDARS, Inertial units.



Attitude Representation

There are different conventions to define orientation in 3D. The following convention for *Roll, Pitch* and *Yaw* is the one applied here.



Attitude Representation

The mathematical representation for this sequence can be seen as composed by three pure rotations (changes of coordinates):

a) Rotation about the *z*-axis by the yaw angle.
b) Rotation about the transformed (once rotated in step (a)) *y*-axis by the pitch angle.
c) Rotation about the transformed (twice-rotated, due to (a),(b)) *x*-axis by the roll angle.

The associated rotation matrix (to transform points from the body coordinate frame to the navigation frame) can be understood as a sequence of transformations, each one being a pure rotation.

Attitude Representation

Let’s navigate through these pure steps, to feel it.

We are in “CFA”, in which we rotate our orientation by an angle φ_z respect to our current (in CFA) **z** axis.

We are now in a new CF, let’s call it CFB. Now in CFB we rotate respect to its ***y*** axis, by φ_y (pitch), so that we are now in a new CF ("CFC", in which we apply the next pure rotation, respect to the current **x** axis, by an angle φ_x,

which results in our final CF, “CFD”

Attitude Representation

Now we see something here, at position **p**, in CFD.

We want to express it in CFA. HOW?

Let's do it in steps.

[1] We express **p** , seen in CFD, in CFC. For that we simply need to apply a 2D rotation respect to axis x, by φ_x. It will only affect coordinates y and z (coordinate x is not affected by that rotation). That is achieved by this rotation matrix. (here, we are actually transforming in 2D, as we did before, in week 1)

$$\mathbf{R}(\varphi_x, 0, 0) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\varphi_x) & -\sin(\varphi_x) \\ 0 & \sin(\varphi_x) & \cos(\varphi_x) \end{bmatrix}$$

$$\mathbf{p}^C = \mathbf{R}(\varphi_x, 0, 0) \cdot \mathbf{p}^D$$

Attitude Representation

We continue, in steps.

[2] Now we need to express the point in CFB. For that we simply need to apply a 2D rotation respect to axis y, by φ_y. It will only affect coordinates z and x (coordinate y is not affected by that rotation). That is achieved by this rotation matrix.

$$\mathbf{R}\left(0,\varphi_y,0\right)=\begin{bmatrix}\cos\left(\varphi_y\right) & 0 & \sin\left(\varphi_y\right)\\ 0 & 1 & 0\\ -\sin\left(\varphi_y\right) & 0 & \cos\left(\varphi_y\right)\end{bmatrix}$$

$$\mathbf{p}^B=\mathbf{R}\left(0,\varphi_y,0\right)\cdot\mathbf{p}^C$$

Attitude Representation

We continue, in steps.

[3] Now we need to express the point in CFA. For that we simply need to apply a 2D rotation respect to axis z, by φ_z. It will only affect coordinates x and y (coordinate z is not affected by that rotation). That is achieved by this rotation matrix.

$$\mathbf{R}(0,0,\varphi_z) = \begin{bmatrix} \cos(\varphi_z) & -\sin(\varphi_z) & 0 \\ \sin(\varphi_z) & \cos(\varphi_z) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{p}^A = \mathbf{R}(0,0,\varphi_z) \cdot \mathbf{p}^B$$

Attitude Representation

We concatenate all those individual steps:

$$\mathbf{p}^A = \mathbf{R}\left(0,0,\varphi_z\right)\cdot\mathbf{p}^B = \mathbf{R}\left(0,0,\varphi_z\right)\cdot\mathbf{R}\left(0,\varphi_y,0\right)\cdot\mathbf{p}^C$$

$$\Downarrow$$

$$\mathbf{p}^A = \mathbf{R}\left(0,0,\varphi_z\right)\cdot\mathbf{R}\left(0,\varphi_y,0\right)\cdot\mathbf{R}\left(\varphi_x,0,0\right)\cdot\mathbf{p}^D$$

$$\mathbf{p}^A = \mathbf{R}\left(\varphi_x,\varphi_y,\varphi_z\right)\cdot\mathbf{p}^D$$

$$\mathbf{R}\left(\varphi_x,\varphi_y,\varphi_z\right) = \mathbf{R}\left(0,0,\varphi_z\right)\cdot\mathbf{R}\left(0,\varphi_y,0\right)\cdot\mathbf{R}\left(\varphi_x,0,0\right)$$

That product of matrixes does not commute! It must be in that order.

It is usually expressed in just one matrix, having all its entries with expressions.

Attitude Representation ..

$$\mathbf{R}\left(\varphi_x,\varphi_y,\varphi_z\right)=\mathbf{R}\left(0,0,\varphi_z\right)\cdot\mathbf{R}\left(0,\varphi_y,0\right)\cdot\mathbf{R}\left(\varphi_x,0,0\right)$$

$$\mathbf{R}\left(\varphi_x,0,0\right)=\begin{bmatrix}1 & 0 & 0\\ 0 & \cos\left(\varphi_x\right) & -\sin\left(\varphi_x\right)\\ 0 & \sin\left(\varphi_x\right) & \cos\left(\varphi_x\right)\end{bmatrix},\quad \mathbf{R}\left(0,\varphi_y,0\right)=\begin{bmatrix}\cos\left(\varphi_y\right) & 0 & \sin\left(\varphi_y\right)\\ 0 & 1 & 0\\ -\sin\left(\varphi_y\right) & 0 & \cos\left(\varphi_y\right)\end{bmatrix}$$

$$\mathbf{R}\left(0,0,\varphi_z\right)=\begin{bmatrix}\cos\left(\varphi_z\right) & -\sin\left(\varphi_z\right) & 0\\ \sin\left(\varphi_z\right) & \cos\left(\varphi_z\right) & 0\\ 0 & 0 & 1\end{bmatrix}$$

Inverse?

$$\mathbf{R}\left(\varphi_x,\varphi_y,\varphi_z\right)^{-1}=\left(\mathbf{R}\left(0,0,\varphi_z\right)\cdot\mathbf{R}\left(0,\varphi_y,0\right)\cdot\mathbf{R}\left(\varphi_x,0,0\right)\right)^{-1}=$$

$$=\mathbf{R}\left(\varphi_x,0,0\right)^{-1}\cdot\mathbf{R}\left(0,\varphi_y,0\right)^{-1}\cdot\mathbf{R}\left(0,0,\varphi_z\right)^{-1}=$$

$$=\mathbf{R}\left(-\varphi_x,0,0\right)\cdot\mathbf{R}\left(0,-\varphi_y,0\right)\cdot\mathbf{R}\left(0,0,-\varphi_z\right)$$

$$\mathbf{R}\left(\varphi_x,\varphi_y,\varphi_z\right)^{-1}\neq\mathbf{R}\left(-\varphi_x,-\varphi_y,-\varphi_z\right):\quad\text{INVALID}$$

$$\mathbf{R}\left(\varphi_x,\varphi_y,\varphi_z\right)^{-1}=\mathbf{R}\left(\varphi_x,\varphi_y,\varphi_z\right)^{T}\quad\text{VALID}$$

If an observer that is located at position **T**, having an orientation $\left(\varphi_x, \varphi_y, \varphi_z\right)$

sees an object at position $\mathbf{p}^D$ in is local CF; that point, if expressed in the global CF, will be

$$\mathbf{p}^A = \mathbf{R}\left(\varphi_x, \varphi_y, \varphi_z\right) \cdot \mathbf{p}^D + \mathrm{T}$$

We see some examples in class

Now that we know how to express 3D attitude, we can discuss about IMU and Gyroscopes (in 3D).

We continue with IMUs, in the next part of the lecture.