Università degli Studi di Padova

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE
Corso di Laurea Magistrale in Ingegneria dell'Automazione

Tesi di laurea magistrale

Motorcycle attitude estimation from IMU noisy measurements

Candidato:
Luca Caiaffa
Matricola 1164725

Relatore:
Ch.mo Prof. Alessandro Beghi

Correlatore:
Mattia Bruschetta

Anno Accademico 2018-2019

Ai miei genitori

Ringraziamenti

Desidero ringraziare coloro che hanno contribuito allo sviluppo di questo progetto e alla stesura di questa tesi.

In particolare, un sentito ringraziamento va al Professore Beghi Alessandro, per avermi dato l'opportunità di sviluppare questa tesi.
Un grande ringraziamento va a Bruschetta Mattia e Picotti Enrico, per avermi sempre assistito e motivato nello sviluppo di questo progetto. Fondamentale è stata la loro completa disponibilità al dialogo per ogni possibile dubbio.
Ringrazio inoltre Maran Fabio e Benedetti Samuele per l'appoggio e la collaborazione datami in questo periodo.

Ringrazio tutti i miei amici, con i quali ho condiviso molte esperienze e che mi hanno aiutato a divertirmi in questi anni di studio.

Ringrazio mia Mamma che ha sempre creduto in me, ed è grazie a lei che ho potuto raggiungere questo obiettivo.

Ringrazio infine mio Papà, che sarebbe stato fiero di questo traguardo.

Abstract

Rigid body attitude estimation is the problem of finding the relative orientation between two reference frames, that are respectively world and body.
In the case of a motorcycle, this task is performed by using Inertial Measurement Unit sensors, like accelerometers and gyroscopes. Information about these sensors involves angular velocities and body kinematics, which are related to body angles.
Since accelerometers and gyroscopes have complementary characteristics, their measures have to be combined in order to improve the accuracy of the estimation.
Attitude estimation based on different sensor is known as Sensor Fusion, and in this thesis it is performed using Extended Kalman Filter.
The accuracy of this Sensor Fusion filter is tested under simulative and experimental datasets, in order to understand its properties and robustness.

Contents

List of Figures

Chapter 1

Introduction

The problem of motorcycle attitude estimation is fundamental nowadays because it is the base for vehicle control.
Attitude estimation is defined as the problem of estimating the three angles of a body, regarding to a specific convention.
Tasks like traction control, anti-wheelie systems are based on the angles of the vehicle, and more in general they are based on vehicle state. While the complexity of the state of a multi-body system like a motorcycle depends on the choice of its dynamical model, its angles are always involved because they define the relative orientation of the vehicle with respect to the world.
Attitude estimation techniques have been studied with all type of vehicles, in particular in the aeronautic field. While car and motorcycles do not strictly need to know their attitude, air vehicles are based on Inertial Navigation Systems (INS), that are a series of techniques needed to estimate position, velocity and orientation from on-board sensors, without an external reference. Similar techniques can be applied to other vehicles, with an adaptation to the particular case.
Attitude estimation is based on on-board sensors, that are Inertial Measurement Unit (IMU), GPS, and other sensors capable of producing information about the vehicle, like star-tracking sensors, cameras and lasers.
There are two main methods to proceed with attitude estimation, that are:

- Sensor Fusion methods: they combine information from different sensors to compute the attitude estimation without using any dynamical equation of the body. These methods are independent from the particular vehicle since they treat it as a generic body, and so they are suited for each vehicle.

- Vehicle related methods: in addition with the combination of different sensors, these methods includes dynamical equations related to the vehicle, defining several vehicle states. These methods are related to the particular vehicle because they are based on its dynamics.

The problem of Sensor Fusion has been addressed in cases like [1],[2],[3], that are based on IMU sensors, that coincides with gyroscopes, accelerometers and magnetometers.
Other sensor fusion methods use different sensors like optical sensors, as in [4], cameras, as in [5].
Regarding to motorcycle related methods, several dynamical models with different complexity have been studied. In [6], the motorcycle is studied in its longitudinal and vertical dynamics, while in [7] the motorcycle is studied in its lateral dynamics. In other cases, like [8], the motorcycle is studied as multi-body system, dividing the vehicle in main frame and wheels.
In this thesis Sensor Fusion methods will be discussed, and they will be applied to motorcycle, even if they could be applied also with other vehicles with same sensor equipment.

Sensors that will be used are accelerometer, gyroscope, together with the measure of velocities. In particular, attitude estimation will be analyzed with sensors singularly, and then they will be combined.
Regarding to Sensor Fusion methods, there are several techniques to combine information from different sensors. In [3] and [9], two complementary filter are presented, while in [10] and [11], statistical filter like Kalman Filter (KF) or Particle Filter (PF) are used.
Regarding to convention, methods are often implemented by resorting to quaternions, as in [12], but in this case angles and rotation matrices will be used because they are related to the physical system.
In this thesis, statistical sensor fusion methods will be analyzed and they will be tested on simulative and experimental datasets.

Chapter 2

Reference systems and rotation matrices

2.1 Reference systems

The motion of an object in the three dimensional space is uniquely described by six variables which depend on the chosen reference frame. These variables are three for position and three for orientation. The position of an object is referred to the one of its center of mass, while the orientation is a description of the whole rigid body.
The problem of estimating the position and orientation of a rigid body is known as *pose*, that is determined by the rotation matrix and the origin of the coordinate system.
We will consider two reference frames where data can be expressed, that are:

- **World reference frame** (x^w, y^w, z^w): the world reference frame, also known as global or general, is fixed in space, and it is the reference for the inertial point of view for the motion of a rigid body.

- **Body reference frame** (x^b, y^b, z^b): the body frame is rigidly attached to the body of the moving object, that means that it is the reference for the non-inertial point of view for the motion of the object.

Reference frames are completely described by the center of coordinate system $\mathbf{O}$ and by the triplet of axis versors, that is $(\mathbf{u}_x, \mathbf{u}_y, \mathbf{u}_z)$, expressed with respect to the fixed frame.
Later on we will need another reference frame, which is related to the vehicle, that is

- **Heading reference frame** (x^h, y^h, z^h): the heading reference frame is the one where velocities of the vehicle are acquired, that is a frame which has the same xy plane as the world frame, while rotates about z axis.

It is important to notice that while versors of the world reference are constant, versors of the other frames are attached to the body, and then they are related to the motion, becoming time variant. Having a point $\mathbf{p} \in \mathbb{R}^3$, we denote as

$$\mathbf{p}^w \doteq (p_x^w, p_y^w, p_z^w), \qquad \mathbf{p}^b \doteq (p_x^b, p_y^b, p_z^b) \tag{2.1}$$

the world and body frame representation of $\mathbf{p}$ respectively.
We now need to find the relation between the expression of the same point in two different reference frame, that is the rototranslation matrix. Two different reference system are shown in Fig.(2.1).

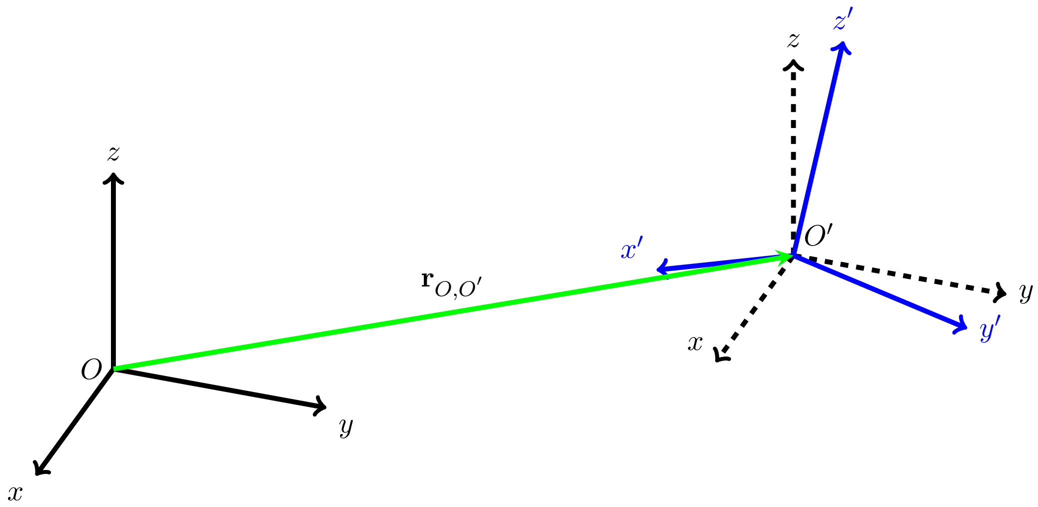


Figure 2.1: Reference systems and relative position and orientation

2.2 Relation between reference systems

2.2.1 Rotation matrix

Axis rotation consists of a transformation which has to preserve the orthogonality and the unitary norm of versors of the axis. This transformation is associated with a rotation matrix, that is a matrix whose multiplication with a vector rotates it preserving its length.
Rotation matrix are defined by the three dimensional special orthogonal group, that is

$$SO(3) = \{\mathbf{R} \in \mathbb{R}^{3\times 3} : \mathbf{R}\mathbf{R}^T = \mathbf{I}, |\mathbf{R}| = 1\} \tag{2.2}$$

The transformation between two different coordinate systems is unique, but since the matrix has 9 elements and 3 degrees of freedom it is possible to have several representation of the transformation, depending on the problem.
The relation between coordinate system can also be expressed by quaternions, whose depend on 4 variables and also avoid singularities problems. In our hypotheses we will be far from singularities and rotation matrices will ease the problem, since they are intuitive and related to the physical representation of the problem.

2.2.2 Coordinate transformation

In this thesis we define the rotation matrix as the matrix that pre-multiplied by a vector expressed in world reference frame gives its body reference frame expression.
Depending on the chosen convention, it is possible to inverse the definition, that translates into a transposition of the matrix.
Defining a vector $\mathbf{v} \in \mathbb{R}^3$ we have that

$$\mathbf{v}^b = \mathbf{R}_w^b \mathbf{v}^w \tag{2.3}$$

$$\mathbf{v}^w = \mathbf{R}_b^w \mathbf{v}^b \tag{2.4}$$

We denote $\mathbf{R} \doteq \mathbf{R}_w^b$ and then $\mathbf{R}^T = \mathbf{R}_b^w$. When it is necessary, the matrix $\mathbf{R}_a^b$ transforms the coordinate of a vector expressed in a-frame into b-frame.

Since we are dealing with vectors, we have no information about the vector absolute position, which means that we are rotating it from its point of application, but we are not translating it. Calling $\mathbf{O}_w, \mathbf{O}_b$ the origin of world and body coordinate systems respectively, we need to subtract them from vectors in order to eliminate offsets due to translation.
We obtain, starting from Eq.(2.3),(2.4)

$$\mathbf{x}^b = \mathbf{R}(\mathbf{x}^w - \mathbf{O}_w^w) = \mathbf{R}\mathbf{x}^w + \mathbf{O}_w^b \tag{2.5}$$

$$\mathbf{x}^w = \mathbf{R}^T(\mathbf{x}^b - \mathbf{O}_b^b) = \mathbf{R}^T\mathbf{x}^b + \mathbf{O}_b^w \tag{2.6}$$

It is common to define the augmented state vector as $[\mathbf{x}^T \quad 1]^T \in \mathbb{R}^4$ in order to obtain matrices representation of the rototranslation, that are

$$\begin{bmatrix} \mathbf{x}^b \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{O}_w^b \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{x}^w \\ 1 \end{bmatrix} \tag{2.7}$$

$$\begin{bmatrix} \mathbf{x}^w \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}^T & \mathbf{O}_b^w \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \mathbf{x}^b \\ 1 \end{bmatrix} \tag{2.8}$$

In this thesis, since we are solving the problem of attitude determination, we will only care about rotation matrices, while we will not need information about absolute position between frames.

2.2.3 Coordinate rotation

There are several ways to build the rotation matrix, depending on the adopted convention. The sequences are called Euler or Tait-Bryan rotations, depending on which axis we choose. A simple case is to build the matrix as a sequence of three rotations, one for each axis.
In this case we will consider the *Yaw-Pitch-Roll* sequence, that is defined as the sequence of rotation about z-axis, y-axis and x-axis. This sequence is a typical choice for vehicle attitude.
We can now define single rotations about axis, where each one is obtained by a rotation matrix.
The three coordinate rotations, that are shown in Fig.(2.2) are equal to:

- Yaw rotation ψ: rotation about z axis,

$$(x, y, z) \rightarrow (x', y', z) \qquad \mathbf{R}_z(\psi) = \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ \text{-}\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.9}$$

- Pitch rotation θ: rotation about y axis,

$$(x, y, z) \rightarrow (x', y, z') \qquad \mathbf{R}_y(\theta) = \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \tag{2.10}$$

- Roll rotation ϕ: rotation about x axis,

$$(x, y, z) \rightarrow (x, y', z') \qquad \mathbf{R}_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix} \tag{2.11}$$

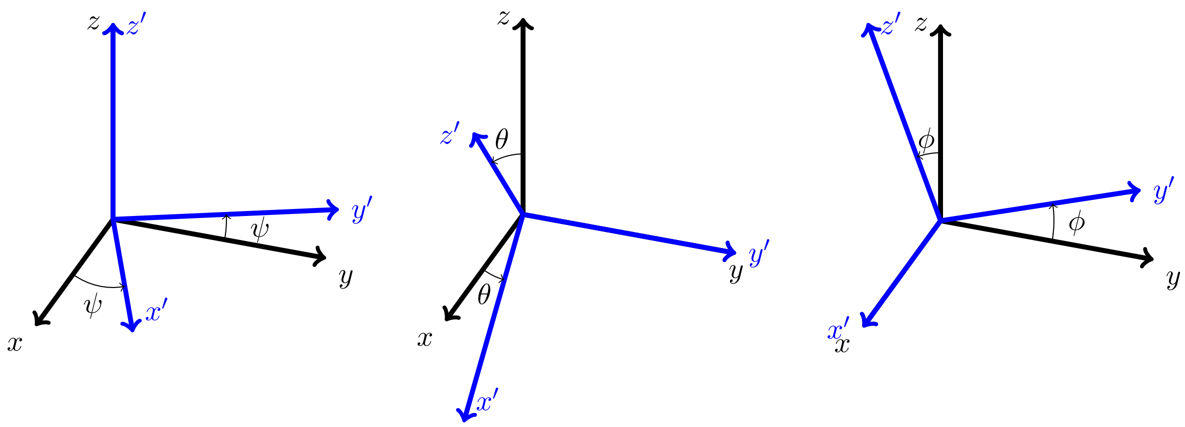


Figure 2.2: Elementary rotations

The sequence of rotation about different axis translates into matrix premultiplication, and then the *Yaw-Pitch-Roll* sequence gives

$$\mathbf{R}(\phi,\theta,\psi) = \mathbf{R}_x(\phi)\mathbf{R}_y(\theta)\mathbf{R}_z(\psi) \tag{2.12}$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix} \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ \text{-}\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.13}$$

Evaluating products in Eq.(2.13) we obtain the final relation between world and body frames, that is:

$$\mathbf{R}(\phi,\theta,\psi) = \begin{bmatrix} c_\theta c_\psi & c_\theta s_\psi & -s_\theta \\ s_\phi s_\theta c_\psi - c_\phi s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\theta s_\phi \\ c_\phi s_\theta c_\psi + s_\phi s_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi & c_\theta c_\phi \end{bmatrix} \tag{2.14}$$

The inverse relation, that is the one which expresses a body reference vector in world coordinates is given by the transpose of (2.14), that is

$$\mathbf{R}^T(\phi,\theta,\psi) = \begin{bmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta c_\psi - s_\phi c_\psi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{bmatrix} \tag{2.15}$$

Knowing matrix in Eq.(2.14), the single axis rotation angles can be obtained by

$$\phi = \text{atan2}(r_{23}, r_{33}) \tag{2.16}$$

$$\theta = \text{-asin}(r_{13}) \tag{2.17}$$

$$\psi = \text{atan2}(r_{12}, r_{11}) \tag{2.18}$$

where the entries of $\mathbf{R}$ are mapped as

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{2.19}$$

The main difference between different choice for the sequence of rotation is the inverse operation, that is the previous one. Since in physical problems the aim consists of obtaining angles that are capable of describing true quantities, it is necessary to choice a consistent representation. In the case of motorcycle there is a correspondence between these angles and vehicle attitude, that is:

- Yaw: rotation about z axis is the first one applied and it corresponds to the turn of a motorcycle, while staying upright.
- Pitch: rotation about y axis is applied when the vehicle has turned yet, and then when it is still in vertical position. The pitch rotation corresponds to work of suspensions and wheelie phenomena.
- Roll: rotation about x axis is applied when the vehicle has pitched yet. In this case the roll rotation corresponds to the lean condition.

Last, it is important to remember that rotations are not commutative and then the same angles but in a different sequence leads to different orientation, and so the sequence z-y-x is the most similar to the real angles we are dealing with. A representation of the sequence of rotations in show in Fig.(2.3).

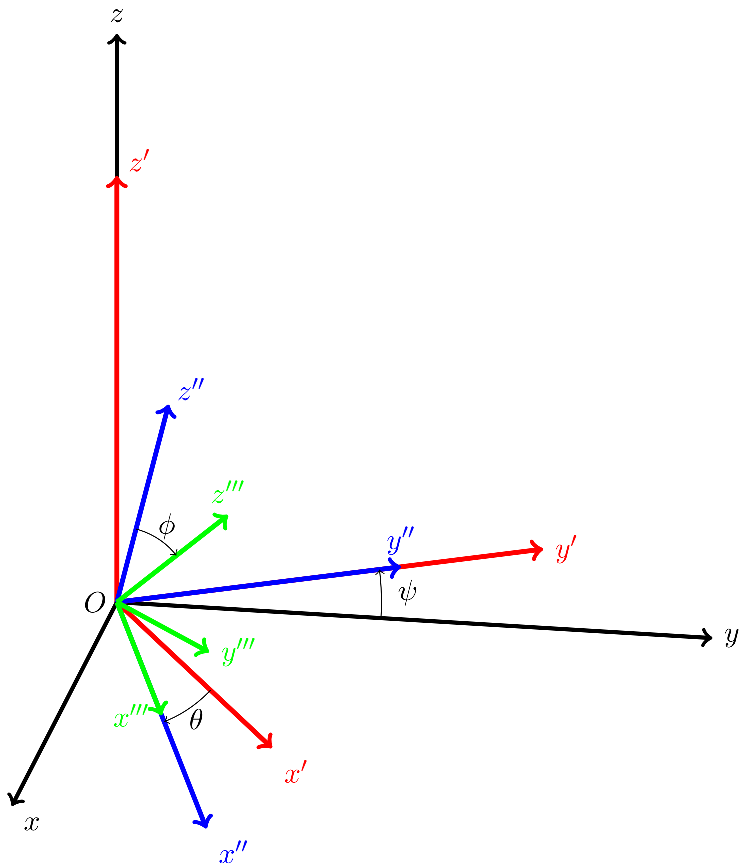


Figure 2.3: Yaw-Pitch-Roll rotation sequence

2.2.4 Heading frame

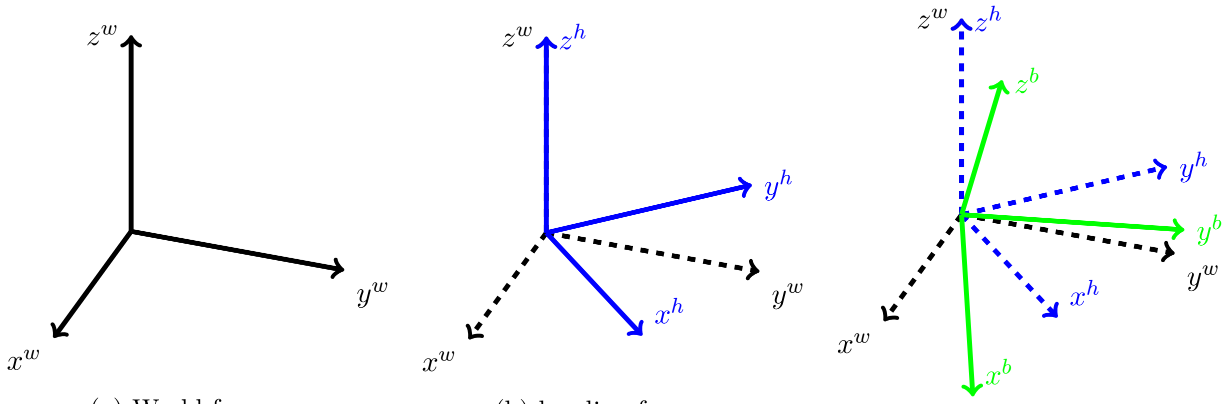
It is important to consider also the heading frame, which is the one where velocities are read. Indeed, longitudinal, lateral and vertical velocities of the vehicle are determined by sensors in this frame, whose x-axis always points the actual direction of the motorcycle.
The transformation between world and heading frame can be obtained from $\mathbf{R}$ by substituting $\phi = 0, \theta = 0$, obtaining

$$\mathbf{R}_w^h = \mathbf{R}(0, 0, \psi) = \begin{bmatrix} c_\psi & s_\psi & 0 \\ -s_\psi & c_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.20}$$

while the transformation between heading and body frame can be obtained from $\mathbf{R}$ by substituting $\psi = 0$, obtaining

$$\mathbf{R}_h^b = \mathbf{R}(\phi, \theta, 0) = \begin{bmatrix} c_\theta & 0 & -s_\theta \\ s_\phi s_\theta & c_\phi & c_\theta s_\phi \\ c_\phi s_\theta & -s_\phi & c_\theta c_\phi \end{bmatrix} \tag{2.21}$$

An example of relation between world, heading and sensor frame is shown in Fig.(2.4).



(a) World frame (b) heading frame (c) heading and Body frame

Figure 2.4: World, Heading and Body reference systems

2.3 Kinematics of rotation matrices

We now want to derive some kinematics properties of the rotation matrices, including their dependence on the relative angular velocity between two frames.
First, we derive the rotation matrix derivative, and then we find the relation between rates of angles and angular velocities.

2.3.1 Rotation matrix time derivative

Starting from rotation matrix definition, we have that

$$\mathbf{R}(t)\mathbf{R}^T(t) = \mathbf{I} \tag{2.22}$$

Taking the time derivative of both members of Eq.(2.22) we obtain

$$\frac{d}{dt}[\mathbf{R}(t)\mathbf{R}^T(t)] = \frac{d}{dt}\mathbf{I} \tag{2.23}$$

$$\dot{\mathbf{R}}(t)\mathbf{R}^T(t) + \mathbf{R}(t)\dot{\mathbf{R}}^T(t) = \mathbf{0} \tag{2.24}$$

Let's define

$$\mathbf{S}(t) \doteq \dot{\mathbf{R}}(t)\mathbf{R}^T(t) \tag{2.25}$$

Then, Eq.(2.24) becomes

$$\dot{\mathbf{R}}(t)\mathbf{R}^T(t) + \mathbf{R}(t)\dot{\mathbf{R}}^T(t) = \mathbf{0} \tag{2.26}$$

$$\dot{\mathbf{R}}(t)\mathbf{R}^T(t) + [\dot{\mathbf{R}}(t)\mathbf{R}^T(t)]^T = \mathbf{0} \tag{2.27}$$

$$\mathbf{S}(t) + \mathbf{S}^T(t) = \mathbf{0} \tag{2.28}$$

$\mathbf{S}$ matrix verifies the anti-symmetric property and then it appertains to $so(3)$ group, defined as

$$so(3) = \{\mathbf{S} \in \mathbb{R}^{3\times 3} : \mathbf{S} = -\mathbf{S}^T\} \tag{2.29}$$

Starting from a vector $\mathbf{s} \in \mathbb{R}^3$, its matrix $\mathbf{S} \in so(3)$ can be written as

$$[\mathbf{s}]_\times \doteq \mathbf{S} = \begin{bmatrix} 0 & -s_3 & s_2 \\ s_3 & 0 & -s_1 \\ -s_2 & s_1 & 0 \end{bmatrix} \tag{2.30}$$

Depending on the case, we will write $\mathbf{S}$ or $[\mathbf{s}]_\times$ for simplicity.
Antisymmetric matrices, also known as skew-symmetric, are directly related to the cross product. Having $\mathbf{s}, \mathbf{v} \in \mathbb{R}^3$, the cross product $\mathbf{s} \times \mathbf{v}$ can be written as

$$\mathbf{s} \times \mathbf{v} = [\mathbf{s}]_\times \mathbf{v} \tag{2.31}$$

where the previous definition has been applied. This is an important relation that will be useful later on since it allows to express the cross product as a matrix-vector multiplication.
From Eq.(2.25) we obtain

$$\dot{\mathbf{R}}(t) = \mathbf{S}(t)\mathbf{R}(t), \quad \text{with } \mathbf{S} \in so(3) \tag{2.32}$$

that is the rotation matrix derivative. We now want to find the expression for $\mathbf{S}(t)$.
If we consider a rotating vector $\mathbf{v}(t)$ we have that

$$|\mathbf{v}(t)| = \mathbf{R}(t)\mathbf{v}(t) \quad \Rightarrow \quad \mathbf{v}(t) = \mathbf{R}^T(t)|\mathbf{v}(t)| \tag{2.33}$$

since in the body frame this vector is constant, because it is attached to this moving frame. Taking the derivative of this vector we obtain

$$\dot{\mathbf{v}}(t) = \dot{\mathbf{R}}^T(t)|\mathbf{v}(t)| = \mathbf{R}^T(t)\mathbf{S}^T|\mathbf{v}(t)| = \mathbf{R}^T(t)[-\mathbf{s}]_\times|\mathbf{v}(t)| \tag{2.34}$$

The rotating vector derivative is also equal to

$$\dot{\mathbf{v}}(t) = \boldsymbol{\omega}^w_{b/w} \times \mathbf{v}(t) = [\boldsymbol{\omega}^w_{b/w}]_\times \mathbf{v}(t) \tag{2.35}$$

where $\boldsymbol{\omega}^w_{b/w}$ is the angular velocity of the body with respect to the world, expressed in world frame. From Prop.(A.12) we have that

$$\dot{\mathbf{v}}(t) = [\mathbf{R}^T\boldsymbol{\omega}^b_{b/w}]_\times \mathbf{v}(t) = \mathbf{R}^T[\boldsymbol{\omega}^b_{b/w}]_\times \mathbf{R}\mathbf{v}(t) = \mathbf{R}^T[\boldsymbol{\omega}^b_{b/w}]_\times|\mathbf{v}(t)| \tag{2.36}$$

Comparing Eq.(2.35)-(2.36) we have obtained that

$$\mathbf{S} = [\boldsymbol{\omega}^b_{b/w}]^T_\times = \begin{bmatrix} 0 & \omega_z & -\omega_y \\ -\omega_z & 0 & \omega_x \\ \omega_y & -\omega_x & 0 \end{bmatrix} \tag{2.37}$$

From this relation we have that the rotation matrix kinematics from world to body is equal to

$$\dot{\mathbf{R}}^b_w(t) = [\boldsymbol{\omega}^b_{b/w}]^T_\times \mathbf{R}^b_w(t) \tag{2.38}$$

2.4 Rates of angles and angular velocities relation

2.4.1 Relation between angular velocities and rates of angles from rotation matrix

Starting from the rotation matrix derivative

$$\dot{\mathbf{R}}(t) = [\boldsymbol{\omega}(t)]^T_\times \mathbf{R}(t) \tag{2.39}$$

it corresponds to a matrix first order differential equation, which solution is

$$\mathbf{R}(t) = e^{\int [\boldsymbol{\omega}(t)]^T_\times dt}\mathbf{R}(0) \tag{2.40}$$

Since the skew-symmetric matrix is time variant, we cannot find a closed form solution for the previous equation.
Since we need an implementable version to calculate angles we can approximate the matrix derivative. The simplest choice is to approximate the derivative with a first order difference, that is forward Euler.
With this approximation we are supposing to have constant angular velocities during each sample, and this results in

$$\mathbf{R}(k+1) = \mathbf{R}(k) + T[\boldsymbol{\omega}(k)]^T_\times \mathbf{R}(k) = [\mathbf{I} + T[\boldsymbol{\omega}(k)]^T_\times]\mathbf{R}(k) \tag{2.41}$$

If we start from Eq.(2.40) we can find a better approximation for the continuous time expression of $\mathbf{R}(t)$ that is

$$\mathbf{R}(k+1) = e^{T[\boldsymbol{\omega}(k)]^T_\times}\mathbf{R}(k) \tag{2.42}$$

Starting from Eq.(2.42), we can derive Eq(2.41) by taking the first order Taylor series expansion of the exponential. From Rodrigues formula, we have that

$$e^{T[\boldsymbol{\omega}(k)]_\times^T} = e^T \left[\mathbf{I} + \frac{\sin||\boldsymbol{\omega}(k)||}{||\boldsymbol{\omega}(k)||} [\boldsymbol{\omega}(k)]_\times^T + \frac{1-\cos||\boldsymbol{\omega}(k)||}{||\boldsymbol{\omega}(k)||^2} \left([\boldsymbol{\omega}(k)]_\times^T \right)^2 \right] \tag{2.43}$$

The update algorithm, which takes care of the entire series expansion of the matrix exponential is

Algorithm 1 Rotation matrix estimates

Require: $T, \phi_0, \theta_0, \psi_0, \omega_x, \omega_y, \omega_z$

for $k = 0$ to $t-1$ **do**

$\mathbf{R}_{k+1} = e^T \left[\mathbf{I} + \frac{\sin||\boldsymbol{\omega}(k)||}{||\boldsymbol{\omega}(k)||} [\boldsymbol{\omega}(k)]_\times^T + \frac{1-\cos||\boldsymbol{\omega}(k)||}{||\boldsymbol{\omega}(k)||^2} \left([\boldsymbol{\omega}(k)]_\times^T \right)^2 \right] \mathbf{R}_k$

subject to $\mathbf{R}_{k+1}\mathbf{R}_{k+1}^T = \mathbf{I}, \det[\mathbf{R}_{l+1}] = 1$

end for

return $\mathbf{R}_{1:t}$

Since we are dealing with an approximation of the matrix evolution, it is important to keep the matrix to maintain its properties of rotation matrix, that are unitary determinant and orthogonality. After we obtain its matrix rotation properties, we can find angles at time k by the relations Eq.(2.16)-(2.17)-(2.18).

It is possible to derive a direct relation between the time derivatives of angles and angular velocities expanding Eq.(2.39).

2.4.2 Direct relation between angular velocities and rate of angles

Let's write the matrix

$$\mathbf{R}(\phi, \theta, \psi) = \mathbf{R}_x(\phi)\mathbf{R}_y(\theta)\mathbf{R}_z(\psi) \tag{2.44}$$

with a compact notation, where the single matrices become

$$\mathbf{R}_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_\phi & s_\phi \\ 0 & -s_\phi & c_\phi \end{bmatrix}, \quad \mathbf{R}_y(\theta) = \begin{bmatrix} c_\theta & 0 & -s_\theta \\ 0 & 1 & 0 \\ s_\theta & 0 & c_\theta \end{bmatrix}, \quad \mathbf{R}_z(\psi) = \begin{bmatrix} c_\psi & s_\psi & 0 \\ -s_\psi & c_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.45}$$

Recalling the matrix derivative equation, we have that

$$\dot{\mathbf{R}}(t) = [-\boldsymbol{\omega}(t)]_\times \mathbf{R}(t) \tag{2.46}$$

$$[-\boldsymbol{\omega}(t)]_\times = \dot{\mathbf{R}}(t)\mathbf{R}^T(t) \tag{2.47}$$

Evaluating the explicit version of $\dot{\mathbf{R}}(t)$ we obtain

$$\dot{\mathbf{R}}(t) = \dot{\mathbf{R}}_x\mathbf{R}_y\mathbf{R}_z + \mathbf{R}_x\dot{\mathbf{R}}_y\mathbf{R}_z + \mathbf{R}_x\mathbf{R}_y\dot{\mathbf{R}}_z \tag{2.48}$$

from which

$$[-\boldsymbol{\omega}(t)]_\times = (\dot{\mathbf{R}}_x\mathbf{R}_y\mathbf{R}_z + \mathbf{R}_x\dot{\mathbf{R}}_y\mathbf{R}_z + \mathbf{R}_x\mathbf{R}_y\dot{\mathbf{R}}_z)\mathbf{R}_z^T\mathbf{R}_y^T\mathbf{R}_x^T \tag{2.49}$$

$$= \dot{\mathbf{R}}_x\mathbf{R}_x^T + \mathbf{R}_x\dot{\mathbf{R}}_y\mathbf{R}_y^T\mathbf{R}_x^T + \mathbf{R}_x\mathbf{R}_y\dot{\mathbf{R}}_z\mathbf{R}_z^T\mathbf{R}_y^T\mathbf{R}_x^T \tag{2.50}$$

Evaluating $\dot{\mathbf{R}}_x\mathbf{R}_x^T, \dot{\mathbf{R}}_y\mathbf{R}_y^T, \dot{\mathbf{R}}_z\mathbf{R}_z^T$ we obtain

$$\dot{\mathbf{R}}_x\mathbf{R}_x^T = \dot{\phi}\begin{bmatrix}0 & 0 & 0\\ 0 & -s_\phi & c_\phi\\ 0 & -c_\phi & -s_\phi\end{bmatrix}\begin{bmatrix}1 & 0 & 0\\ 0 & c_\phi & -s_\phi\\ 0 & s_\phi & c_\phi\end{bmatrix} = \dot{\phi}\begin{bmatrix}0 & 0 & 0\\ 0 & 0 & 1\\ 0 & -1 & 0\end{bmatrix} = \dot{\phi}[-\mathbf{e}_1]_\times \tag{2.51}$$

$$\dot{\mathbf{R}}_y\mathbf{R}_y^T = \dot{\theta}\begin{bmatrix}-s_\theta & 0 & -c_\theta\\ 0 & 1 & 0\\ c_\theta & 0 & -s_\theta\end{bmatrix}\begin{bmatrix}c_\theta & 0 & s_\theta\\ 0 & 1 & 0\\ -s_\theta & 0 & c_\theta\end{bmatrix} = \dot{\theta}\begin{bmatrix}0 & 0 & -1\\ 0 & 0 & 0\\ 1 & 0 & 0\end{bmatrix} = \dot{\theta}[-\mathbf{e}_2]_\times \tag{2.52}$$

$$\dot{\mathbf{R}}_z\mathbf{R}_z^T = \dot{\psi}\begin{bmatrix}-s_\psi & c_\psi & 0\\ -c_\psi & -s_\psi & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}c_\psi & -s_\psi & 0\\ s_\psi & c_\psi & 0\\ 0 & 0 & 1\end{bmatrix} = \dot{\psi}\begin{bmatrix}0 & 1 & 0\\ -1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix} = \dot{\psi}[-\mathbf{e}_3]_\times \tag{2.53}$$

Inserting the previous terms in Eq.(2.50) we obtain

$$[-\boldsymbol{\omega}(t)]_\times = \dot{\phi}[-\mathbf{e}_1]_\times + \mathbf{R}_x\dot{\theta}[-\mathbf{e}_2]_\times\mathbf{R}_x^T + \mathbf{R}_x\mathbf{R}_y\dot{\psi}[-\mathbf{e}_3]_\times\mathbf{R}_y^T\mathbf{R}_x^T \tag{2.54}$$

Writing explicitly the previous terms of the sum we have

$$\mathbf{R}_x\dot{\theta}[-\mathbf{e}_2]_\times\mathbf{R}_x^T = \dot{\theta}\begin{bmatrix}1 & 0 & 0\\ 0 & c_\phi & s_\phi\\ 0 & -s_\phi & c_\phi\end{bmatrix}\begin{bmatrix}0 & 0 & -1\\ 0 & 0 & 0\\ 1 & 0 & 0\end{bmatrix}\begin{bmatrix}1 & 0 & 0\\ 0 & c_\phi & -s_\phi\\ 0 & s_\phi & c_\phi\end{bmatrix} \tag{2.55}$$

$$\mathbf{R}_x\mathbf{R}_y\dot{\psi}[-\mathbf{e}_3]_\times\mathbf{R}_y^T\mathbf{R}_x^T = \dot{\psi}\begin{bmatrix}1 & 0 & 0\\ 0 & c_\phi & s_\phi\\ 0 & -s_\phi & c_\phi\end{bmatrix}\begin{bmatrix}c_\theta & 0 & -s_\theta\\ 0 & 1 & 0\\ s_\theta & 0 & c_\theta\end{bmatrix}\begin{bmatrix}0 & 1 & 0\\ -1 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}1 & 0 & 0\\ 0 & c_\phi & -s_\phi\\ 0 & s_\phi & c_\phi\end{bmatrix}\begin{bmatrix}c_\theta & 0 & s_\theta\\ 0 & 1 & 0\\ -s_\theta & 0 & c_\theta\end{bmatrix} \tag{2.56}$$

$$= \dot{\psi}\begin{bmatrix}1 & 0 & 0\\ 0 & c_\phi & s_\phi\\ 0 & -s_\phi & c_\phi\end{bmatrix}\begin{bmatrix}0 & c_\theta & 0\\ -c_\theta & 0 & -s_\theta\\ 0 & s_\theta & 0\end{bmatrix}\begin{bmatrix}1 & 0 & 0\\ 0 & c_\phi & -s_\phi\\ 0 & s_\phi & c_\phi\end{bmatrix} = \tag{2.57}$$

$$= \dot{\psi}\begin{bmatrix}0 & c_\theta c_\phi & -c_\theta s_\phi\\ -c_\theta c_\phi & 0 & -s_\theta\\ s_\phi c_\theta & s_\theta & 0\end{bmatrix} \tag{2.58}$$

The skew-symmetric representation of $\boldsymbol{\omega}$ results

$$[-\boldsymbol{\omega}(t)]_\times = \begin{bmatrix}0 & \omega_z & -\omega_y\\ -\omega_z & 0 & \omega_x\\ \omega_y & -\omega_x & 0\end{bmatrix} = \begin{bmatrix}0 & -s_\phi\dot{\theta} + c_\theta c_\phi\dot{\psi} & -c_\phi\dot{\theta} - c_\theta s_\phi\dot{\psi}\\ s_\phi\dot{\theta} - c_\theta c_\phi\dot{\psi} & 0 & \dot{\phi} - s_\theta\dot{\psi}\\ c_\phi\dot{\theta} + c_\theta s_\phi\dot{\psi} & -\dot{\phi} + s_\theta\dot{\psi} & 0\end{bmatrix} \tag{2.59}$$

from which it is possible to derive the angles rates-angular velocities relation, that is

$$\begin{bmatrix}\omega_x\\ \omega_y\\ \omega_z\end{bmatrix} = \begin{bmatrix}+\dot{\phi} - s_\theta\dot{\psi}\\ +c_\phi\dot{\theta} + c_\theta s_\phi\dot{\psi}\\ -s_\phi\dot{\theta} + c_\theta c_\phi\dot{\psi}\end{bmatrix} = \begin{bmatrix}1 & 0 & -s_\theta\\ 0 & +c_\phi & +c_\theta s_\phi\\ 0 & -s_\phi & +c_\theta c_\phi\end{bmatrix}\begin{bmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{bmatrix} \tag{2.60}$$

It is possible to invert the previous relation in order to obtain the rates of angles as an affine function of the angular velocities of the system, that is

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi / c_\theta & c_\phi / c_\theta \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \quad \text{with} \quad c_\theta \neq 0 \tag{2.61}$$

The inversion of Eq.(2.60), has a singularity when $c_\theta = 0, \theta = \pm 90°$, but in our case, since θ represents the pitch angle, $-90° < \theta < +90°$.
In order to compute the estimate of the attitude from angular velocities we need to discretize Eq.(2.61), as we did for Eq.(2.39).
The simplest case to approximate the derivative operator is to consider a first order difference equation. The equivalent first order discretized system is equal to

$$\begin{bmatrix} \phi(k+1) \\ \theta(k+1) \\ \psi(k+1) \end{bmatrix} = \begin{bmatrix} \phi(k) \\ \theta(k) \\ \psi(k) \end{bmatrix} + T \begin{bmatrix} 1 & \sin[\phi(k)]\tan[\theta(k)] & \cos[\phi(k)]\tan[\theta(k)] \\ 0 & \cos[\phi(k)] & -\sin[\phi(k)] \\ 0 & \sin[\phi(k)]/\cos[\theta(k)] & \cos[\phi(k)]/\cos[\theta(k)] \end{bmatrix} \begin{bmatrix} \omega_x(k) \\ \omega_y(k) \\ \omega_z(k) \end{bmatrix} \tag{2.62}$$

In this case we have no constraints that have to be respected, and so it is possible to direct implement the previous equation.
We can then construct the rotation matrix a posteriori once we have the estimates of angles, keeping all rotation matrix properties satisfied.

Algorithm 2 Rates of angles and rotation matrix estimates

Require: $T, \phi_0, \theta_0, \psi_0, \omega_x, \omega_y, \omega_z$
  for $k = 0$ to t **do**
    $\phi_{k+1} = \phi_k + T(\omega_x + \sin[\phi_k]\tan[\theta_k]\omega_y + \cos[\phi_k]\tan[\theta_k]\omega_z)$
    $\theta_{k+1} = \theta_k + T(\cos[\phi_k]\omega_y - \sin[\phi_k]\omega_z)$
    $\psi_{k+1} = \psi_k + T(\sin[\phi_k]/\cos[\theta_k]\omega_y + \cos[\phi_k]/\cos[\theta_k]\omega_z)$
    $\mathbf{R}_{k+1} = \mathbf{R}(\phi_{k+1}, \theta_{k+1}, \psi_{k+1})$
  end for
  return $\phi_{1:t}, \theta_{1:t}, \psi_{1:t}, \mathbf{R}_{1:t}$

Chapter 3

Gyroscope and Acceleration measure characterization

The aim of this chapter is to derive the relations between gyroscope and accelerometer measurements and body angles.
From these relation it is possible to solve attitude estimation problem with single IMU sensor.

3.1 Attitude estimation from gyroscope

The relation between rates of angles and angular velocities is

$$\dot{\boldsymbol{\zeta}}(t) = \mathbf{f}(\boldsymbol{\zeta}(t), \boldsymbol{\omega}(t)) \tag{3.1}$$

where $\boldsymbol{\zeta} = [\phi, \theta, \psi]^T$. The discretization with forward euler gives

$$\frac{\boldsymbol{\zeta}_{k+1} - \boldsymbol{\zeta}_k}{T} = \mathbf{f}(\boldsymbol{\zeta}_k, \boldsymbol{\omega}_k) \quad \Rightarrow \quad \boldsymbol{\zeta}_{k+1} = \boldsymbol{\zeta}_k + T[\mathbf{f}(\boldsymbol{\zeta}_k, \boldsymbol{\omega}_k)] \tag{3.2}$$

where the explicit version of previous equation is the one derived in Ch.(2), that is

$$\phi_{k+1} = \phi_k + T[\omega_{k,x} + \sin(\phi_k)\tan(\theta_k)\omega_{k,y} + \cos(\phi_k)\tan(\theta_k)\omega_{k,z}] \tag{3.3}$$

$$\theta_{k+1} = \theta_k + T[\cos(\phi_k)\omega_{k,y} - \sin(\phi_k)\omega_{k,z}] \tag{3.4}$$

$$\psi_{k+1} = \psi_k + T[\sin(\phi_k)/\cos(\theta_k)\omega_{k,y} - \cos(\phi_k)/\cos(\theta_k)\omega_{k,z}] \tag{3.5}$$

Since gyroscope measure at time k is $\boldsymbol{\omega}_k$, the previous relation consists of the attitude estimation algorithm based on gyroscope.

3.2 Attitude estimation from accelerometer

In the case we only have access to accelerometer, we deal with a measure of body accelerations, with the gravity term in addition. Accelerometer readings are equal to

$$\mathbf{a} = \ddot{\mathbf{p}} + \mathbf{R}\mathbf{g} \tag{3.6}$$

where $\mathbf{g} = [0, 0, g]^T, g = 9.81m/s^2$, $\ddot{\mathbf{p}}$ includes all acceleration terms beside gravity and $\mathbf{R}$ is the rotation matrix from world to body frame.
Writing Eq.(3.6) in explicit form leads to

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} \ddot{p}_x \\ \ddot{p}_y \\ \ddot{p}_z \end{bmatrix} + \begin{bmatrix} c_\theta c_\psi & c_\theta s_\psi & -s_\theta \\ s_\phi s_\theta c_\psi - c_\phi s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\theta s_\phi \\ c_\phi s_\theta c_\psi + s_\phi s_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi & c_\theta c_\phi \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} \ddot{p}_x \\ \ddot{p}_y \\ \ddot{p}_z \end{bmatrix} + \begin{bmatrix} -\sin(\theta)g \\ \sin(\phi)\cos(\theta)g \\ \cos(\phi)\cos(\theta)g \end{bmatrix} \tag{3.7}$$

From the previous relation it is possible to derive exact values of ϕ and θ. The yaw angle ψ cannot be estimated using only accelerometer, because the projection of gravity is invariant with respect to z-axis rotations.
The estimates consist of a static relation, which does not depend on time and there are not approximation due to discretization. From Eq.(3.7) we can state that

$$\theta = \text{asin}\left[\frac{\ddot{p}_x - a_x}{g}\right] \tag{3.8}$$

$$\phi = \text{atan}\left[\frac{a_y - \ddot{p}_y}{a_z - \ddot{p}_z}\right] \tag{3.9}$$

It is important to notice that in order to have a correct estimate for θ, we need to know $\ddot{p}_x$, that is the body longitudinal acceleration, while for ϕ, we need to know $\ddot{p}_y$ and $\ddot{p}_z$, that are lateral and vertical accelerations. In general, acceleration terms heavily depend on body dynamics, and then in order to characterize this term we need a description of how the vehicle is moving.

3.2.1 Acceleration characterization

We will insert the knowledge about external forces by finding the acceleration that a point in the body is subject to, through kinematics analysis.

Curvilinear motion

Our aim is to completely characterize the acceleration of a point moving with a curvilinear motion, as shown in Fig.(3.1). The motion of the position of the center of mass of a rigid body can be seen as a moving frame of reference described by $\mathbf{p}(t)$.

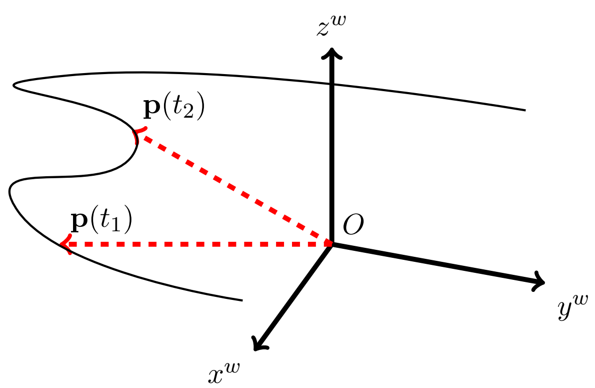


Figure 3.1: Curvilinear motion

The position of $\mathbf{p}(t)$ can be described by the position vector

$$\mathbf{r}^w_{b/w} = x_{b/w}\mathbf{u}^w_x + y_{b/w}\mathbf{u}^w_y + z_{b/w}\mathbf{u}^w_z \tag{3.10}$$

where $\mathbf{u}^w_x, \mathbf{u}^w_y, \mathbf{u}^w_z$ are world reference frame versors, which are constant by definition, and the subscript b/w means that we are referring to distances between center of mass of body and world frame. Since all vectors are expressed in world frame we omit the superscript w in order to ease the notation. With these coordinates we are supposing to observe the moving point in

an inertial fixed frame.
The velocity of the point, from derivative definition, is equal to

$$\mathbf{v}_{b/w} = \frac{d\mathbf{r}_{b/w}}{dt} \tag{3.11}$$

and inserting Eq.(3.10) in the previous one we obtain

$$\mathbf{v}_{b/w} = \frac{dx_{b/w}}{dt}\mathbf{u}_x + \frac{dy_{b/w}}{dt}\mathbf{u}_y + \frac{dz_{b/w}}{dt}\mathbf{u}_z + x_{b/w}\frac{d\mathbf{u}_x}{dt} + y_{b/w}\frac{d\mathbf{u}_y}{dt} + z_{b/w}\frac{d\mathbf{u}_z}{dt} \tag{3.12}$$

Since we have constant versors, the expression of the velocity becomes

$$\mathbf{v}_{b/w} = \frac{dx_{b/w}}{dt}\mathbf{u}_x + \frac{dy_{b/w}}{dt}\mathbf{u}_y + \frac{dz_{b/w}}{dt}\mathbf{u}_z \tag{3.13}$$

In the case we have access to angular velocity of rotation of the point and the position vector with respect the world frme origin, we can express the velocity vector in closed form as

$$\mathbf{v}_{b/w} = \boldsymbol{\omega}_{b/w} \times \mathbf{r}_{b/w} \tag{3.14}$$

where $\boldsymbol{\omega}_{b/w}$ is the angular velocity between the center of mass on the body frame and world, expressed in world frame.
As we did for the velocity derivation, we can proceed with acceleration terms. Applying the derivative definition to the velocity of Eq.(3.13), we obtain

$$\mathbf{a}_{b/w} = \frac{d^2x_{b/w}}{dt^2}\mathbf{u}_x + \frac{d^2y_{b/w}}{dt^2}\mathbf{u}_y + \frac{d^2z_{b/w}}{dt^2}\mathbf{u}_z \tag{3.15}$$

Differentiating Eq.(3.14) we obtain

$$\begin{align}
\mathbf{a}_{b/w} &= \frac{d\mathbf{v}_{b/w}}{dt} = \frac{d}{dt}[\boldsymbol{\omega}_{b/w} \times \mathbf{r}_{b/w}] \tag{3.16}\\
&= \frac{d\boldsymbol{\omega}_{b/w}}{dt} \times \mathbf{r}_{b/w} + \boldsymbol{\omega}_{b/w} \times \frac{d\mathbf{r}_{b/w}}{dt} \tag{3.17}\\
&= \boldsymbol{\alpha}_{b/w} \times \mathbf{r}_{b/w} + \boldsymbol{\omega}_{b/w} \times (\boldsymbol{\omega}_{b/w} \times \mathbf{r}_{b/w}) \tag{3.18}\\
&= \boldsymbol{\alpha}_{b/w} \times \mathbf{r}_{b/w} + \boldsymbol{\omega}_{b/w} \times \mathbf{v}_{b/w} \tag{3.19}
\end{align}$$

where $\boldsymbol{\alpha}$ is the derivative of the angular velocity vector. The acceleration term is made of two components :

- $\boldsymbol{\alpha}_{b/w} \times \mathbf{r}_{b/w}$: longitudinal acceleration.
- $\boldsymbol{\omega}_{b/w} \times \mathbf{v}_{b/w}$: centripetal acceleration term, generated by the curvilinear motion.

Relative motion

We start considering the point $\mathbf{p}(t)$ as seen by two different frames, that are the world frame and the body frame, as shown in Fig.(3.2). In this case we treat $\mathbf{p}(t)$ as the location of the sensors in the vehicle. Then, the position vector of $\mathbf{p}(t)$ can be decomposed in

$$\mathbf{r}_{s/w} = \mathbf{r}_{b/w} + \mathbf{r}_{s/b} \tag{3.20}$$

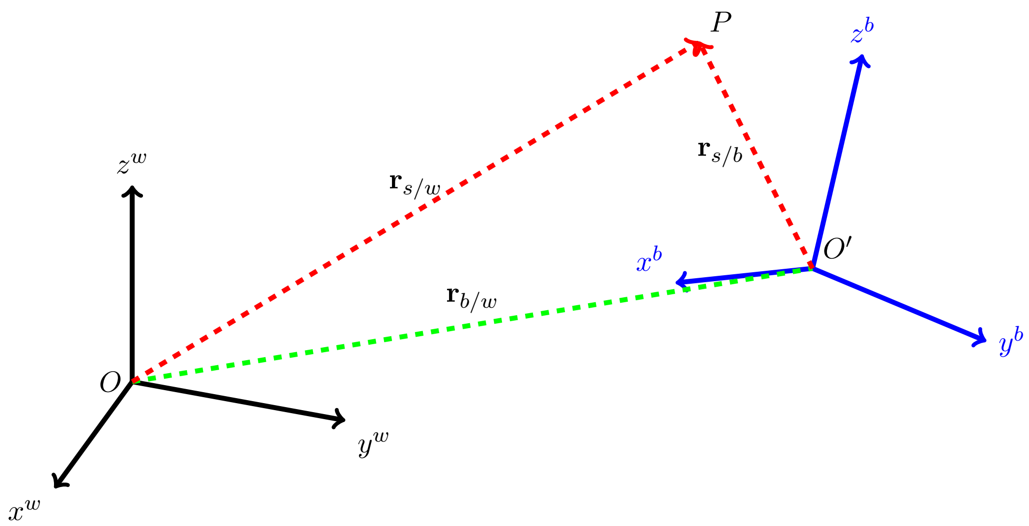


Figure 3.2: Relative motion frames

The three members of previous equations are then

$$\mathbf{r}_{s/w} = x_{s/w}\mathbf{u}_x + y_{s/w}\mathbf{u}_y + z_{s/w}\mathbf{u}_z \tag{3.21}$$

$$\mathbf{r}_{b/w} = x_{b/w}\mathbf{u}_x + y_{b/w}\mathbf{u}_y + z_{b/w}\mathbf{u}_z \tag{3.22}$$

$$\mathbf{r}_{s/b} = x_{s/b}\mathbf{u}_{x_b} + y_{s/b}\mathbf{u}_{y_b} + z_{s/b}\mathbf{u}_{z_b} \tag{3.23}$$

Recalling that body frame is moving, we have that its versors are time variant.
The respective velocities are equal to

$$\mathbf{v}_{s/w} = \frac{d\mathbf{r}_{s/w}}{dt} = \frac{dx_{s/w}}{dt}\mathbf{u}_x + \frac{dy_{s/w}}{dt}\mathbf{u}_y + \frac{dz_{s/w}}{dt}\mathbf{u}_z \tag{3.24}$$

$$\mathbf{v}_{b/w} = \frac{d\mathbf{r}_{b/w}}{dt} = \frac{dx_{b/w}}{dt}\mathbf{u}_x + \frac{dy_{b/w}}{dt}\mathbf{u}_y + \frac{dz_{b/w}}{dt}\mathbf{u}_z \tag{3.25}$$

$$\mathbf{v}_{s/b} = \frac{d\mathbf{r}_{s/b}}{dt} = \frac{dx_{s/b}}{dt}\mathbf{u}_{x_b} + \frac{dy_{s/b}}{dt}\mathbf{u}_{y_b} + \frac{dz_{s/b}}{dt}\mathbf{u}_{z_b} + x_{s/b}\frac{d\mathbf{u}_{x_b}}{dt} + y_{s/b}\frac{d\mathbf{u}_{y_b}}{dt} + z_{s/b}\frac{d\mathbf{u}_{z_b}}{dt} \tag{3.26}$$

Defining

$$\bar{\mathbf{v}}_{s/b} \doteq \frac{dx_{s/b}}{dt}\mathbf{u}_{x_b} + \frac{dy_{s/b}}{dt}\mathbf{u}_{y_b} + \frac{dz_{s/b}}{dt}\mathbf{u}_{z_b} \tag{3.27}$$

the velocity of a point which moves on a trajectory that is $\mathbf{r}(t)$ is equal to

$$\mathbf{v}_{s/w} = \mathbf{v}_{b/w} + \mathbf{v}_{s/b} = \mathbf{v}_{b/w} + \bar{\mathbf{v}}_{s/b} + x_{s/b}\frac{d\mathbf{u}_{x_b}}{dt} + y_{s/b}\frac{d\mathbf{u}_{y_b}}{dt} + z_{s/b}\frac{d\mathbf{u}_{z_b}}{dt} \tag{3.28}$$

From Poisson relation, we have that

$$\frac{d\mathbf{u}}{dt} = \boldsymbol{\omega} \times \mathbf{u} \tag{3.29}$$

where $\boldsymbol{\omega}$ is the rate of change of versors, that is the angular velocity of body with respect to world frame.
Substituting Eq.(3.29) for each derivative in velocity equation we obtain

$$\mathbf{v}_{s/w} = \mathbf{v}_{b/w} + \bar{\mathbf{v}}_{s/b} + x_{s/b}(\boldsymbol{\omega}_{b/w} \times \mathbf{u}_{x_b}) + y_{s/b}(\boldsymbol{\omega}_{b/w} \times \mathbf{u}_{y_b}) + z_{s/b}(\boldsymbol{\omega}_{b/w} \times \mathbf{u}_{z_b}) \tag{3.30}$$

$$= \mathbf{v}_{b/w} + \bar{\mathbf{v}}_{s/b} + \boldsymbol{\omega}_{b/w} \times (x_{s/b}\mathbf{u}_{x_b} + y_{s/b}\mathbf{u}_{y_b} + z_{s/b}\mathbf{u}_{z_b}) \tag{3.31}$$

$$= \mathbf{v}_{b/w} + \bar{\mathbf{v}}_{s/b} + \boldsymbol{\omega}_{b/w} \times \mathbf{r}_{s/b} \tag{3.32}$$

The previous equation depends on three terms, that are:

- $\mathbf{v}_{b/w}$: Relative velocity between two frames, that is the velocity of the body frame center with respect to the world fixed coordinates
- $\bar{\mathbf{v}}_{s/b}$: Relative velocity of the point with respect to the body frame in world reference frame
- $\boldsymbol{\omega}_{s,w} \times \mathbf{r}_{s/b}$: Rotational velocity of the point due to rotation of the body reference frame.

From velocity term, it is possible to derive the acceleration term of this point.
Differentiating Eq.(3.32) we obtain

$$\mathbf{a}_{s/w} = \frac{d\mathbf{v}_{b/w}}{dt} + \frac{d\bar{\mathbf{v}}_{s/b}}{dt} + \frac{d}{dt}[\boldsymbol{\omega}_{b/w} \times \mathbf{r}_{s/b}] \tag{3.33}$$

$$= \frac{d\mathbf{v}_{b/w}}{dt} + \frac{d\bar{\mathbf{v}}_{s/b}}{dt} + \frac{d\boldsymbol{\omega}_{b/w}}{dt} \times \mathbf{r}_{s/b} + \boldsymbol{\omega}_{b/w} \times \frac{d\mathbf{r}_{s/b}}{dt} \tag{3.34}$$

Since previous equation is made of several derivatives, we proceed by evaluating each term separately.
For the term relative to $\mathbf{v}_{b/w}$, we have that

$$\mathbf{a}_{b/w} = \frac{d\mathbf{v}_{b/w}}{dt} = \frac{d^2 x_{b/w}}{dt^2}\mathbf{u}_x + \frac{d^2 y_{b/w}}{dt^2}\mathbf{u}_y + \frac{d^2 z_{b/w}}{dt^2}\mathbf{u}_z \tag{3.35}$$

that coincides with the acceleration of the center of body frame.
Regarding to $\bar{\mathbf{v}}_{s/b}$, we have that

$$\frac{d\bar{\mathbf{v}}_{s/b}}{dt} = \frac{d^2 x_{s/b}}{dt^2}\mathbf{u}_{x_{s/b}} + \frac{d^2 y_{s/b}}{dt^2}\mathbf{u}_{y_{s/b}} + \frac{d^2 z_{s/b}}{dt^2}\mathbf{u}_{z_{s/b}} + \boldsymbol{\omega}_{b/w} \times \mathbf{v}_{s/b} \tag{3.36}$$

Defining

$$\bar{\mathbf{a}}_{s/b} \doteq \frac{d^2 x_{s/b}}{dt^2}\mathbf{u}_{x_{s/b}} + \frac{d^2 y_{s/b}}{dt^2}\mathbf{u}_{y_{s/b}} + \frac{d^2 z_{s/b}}{dt^2}\mathbf{u}_{z_{s/b}} \tag{3.37}$$

we can substitute Eq.(3.36)-(3.37) in Eq.(3.34) obtaining

$$\mathbf{a}_{s/w} = \mathbf{a}_{b/w} + \bar{\mathbf{a}}_{s/b} + \boldsymbol{\omega}_{b/w} \times \mathbf{v}_{s/b} + \frac{d\boldsymbol{\omega}_{b/w}}{dt} \times \mathbf{r}_{s/b} + \boldsymbol{\omega}_{b/w} \times \frac{d\mathbf{r}_{s/b}}{dt} \tag{3.38}$$

$$= \mathbf{a}_{b/w} + \bar{\mathbf{a}}_{s/b} + \boldsymbol{\omega}_{b/w} \times \mathbf{v}_{s/b} + \boldsymbol{\alpha}_{b/w} \times \mathbf{r}_{s/b} + \boldsymbol{\omega}_{b/w} \times \mathbf{v}_{s/b} + \boldsymbol{\omega}_{b/w} \times \boldsymbol{\omega}_{b/w} \times \mathbf{r}_{s/b} \tag{3.39}$$

$$= \mathbf{a}_{b/w} + \bar{\mathbf{a}}_{s/b} + \boldsymbol{\alpha}_{b/w} \times \mathbf{r}_{s/b} + 2\boldsymbol{\omega}_{b/w} \times \mathbf{v}_{s/b} + \boldsymbol{\omega}_{b/w} \times \boldsymbol{\omega}_{b/w} \times \mathbf{r}_{s/b} \tag{3.40}$$

The previous expression is the complete acceleration under relative motion. Previous equation includes all relative acceleration terms, where:

- $\boldsymbol{\alpha}_{b/w} \times \mathbf{r}_{s/b}$ is the transverse acceleration with respect to body frame.
- $2\boldsymbol{\omega}_{b/w} \times \mathbf{v}_{s/b}$ is the Coriolis acceleration.
- $\boldsymbol{\omega}_{b/w} \times \boldsymbol{\omega}_{b/w} \times \mathbf{r}_{s/b}$ is the centripetal acceleration with respect to body frame.

Starting from relative acceleration equations in Eq.(3.40), we can include the explicit version of the relative acceleration between world and body frame, that is $\mathbf{a}_{b/w}$, which yields

$$\mathbf{a} = \boldsymbol{\alpha}_{b/w} \times \mathbf{r}_{b/w} + \boldsymbol{\omega}_{b/w} \times \mathbf{v}_{b/w} + \bar{\mathbf{a}}_{s/b} + \boldsymbol{\alpha}_{b/w} \times \mathbf{r}_{s/b} + 2\boldsymbol{\omega}_{b/w} \times \mathbf{v}_{s/b} + \boldsymbol{\omega}_{b/w} \times \boldsymbol{\omega}_{b/w} \times \mathbf{r}_{s/b} \tag{3.41}$$

In the case of a rigid body, we have that the relative position between sensor frame and a generic point on the rigid body does not change over time, which means that $\mathbf{v}_{s/b} = 0$.
With this assumption, the acceleration of a fixed point on a rigid body is equal to

$$\mathbf{a} = \boldsymbol{\alpha}_{b/w} \times \mathbf{r}_{b/w} + \boldsymbol{\omega}_{b/w} \times \mathbf{v}_{b/w} + \boldsymbol{\alpha}_{b/w} \times \mathbf{r}_{s/b} + \boldsymbol{\omega}_{b/w} \times \boldsymbol{\omega}_{b/w} \times \mathbf{r}_{s/b} \tag{3.42}$$

Body frame representation

Each of the previous relation can be obtained in body frame through the rotation matrix $\mathbf{R}$. The most important equations that we need to project into body frame are Eq.(3.19)-(3.42). It is fundamental to project the acceleration terms in body frame, since we have access to body measurements. Since positions, velocities and accelerations are referred to the center of mass of the body we will omit the subscript b/w to ease the notation. In particular, we have:

- $\boldsymbol{\omega}^b_{b/w}$ is the body angular velocity vector, which coincides with gyroscope readings
- $\mathbf{a}^b$ is the acceleration vector of the sensor mounting position
- $\mathbf{v}^b$ is the velocity vector of the center of mass of the body
- $\mathbf{r}^b_{s/b}$ is the radius vector which consists of the relative position of the sensor mounting position with respect to the the center of mass

Applying distributive property of rotation matrix with respect to the cross product, Eq.(3.19) becomes

$$\mathbf{a}^b = \mathbf{R}[\boldsymbol{\alpha}^w_{b/w} \times \mathbf{r}^w + \boldsymbol{\omega}^w_{b/w} \times \mathbf{v}^w] \tag{3.43}$$
$$= \mathbf{R}\boldsymbol{\alpha}^w_{b/w} \times \mathbf{R}\mathbf{r}^w + \mathbf{R}\boldsymbol{\omega}^w_{b/w} \times \mathbf{R}\mathbf{v}^w \tag{3.44}$$
$$= \boldsymbol{\alpha}^b_{b/w} \times \mathbf{r}^b + \boldsymbol{\omega}^b_{b/w} \times \mathbf{v}^b \tag{3.45}$$

With gyroscope readings we can write the expression of the body acceleration as

$$\mathbf{a}^b = \dot{\mathbf{v}}^b + \boldsymbol{\omega}^b_{b/w} \times \mathbf{v}^b \tag{3.46}$$

The evaluation of $\dot{\mathbf{v}}^b$ yields to

$$\dot{\mathbf{v}}^b = \dot{\mathbf{R}}\mathbf{v}^w + \mathbf{R}\dot{\mathbf{v}}^w = -[\boldsymbol{\omega}^b_{b/w}]_\times \mathbf{R}\mathbf{v}^w + \mathbf{R}\dot{\mathbf{v}}^w \tag{3.47}$$

The body acceleration expression becomes

$$\mathbf{a}^b = -[\boldsymbol{\omega}^b_{b/w}]_\times \mathbf{R}\mathbf{v}^w + \mathbf{R}\dot{\mathbf{v}}^w + \boldsymbol{\omega}^b_{b/w} \times \mathbf{v}^b \tag{3.48}$$
$$= -\boldsymbol{\omega}^b_{b/w} \times \mathbf{v}^b + \mathbf{R}\dot{\mathbf{v}}^w + \boldsymbol{\omega}^b_{b/w} \times \mathbf{v}^b \tag{3.49}$$
$$= \mathbf{R}\dot{\mathbf{v}}^w \tag{3.50}$$

that is the acceleration of the body expressed in body frame.
We can apply the same properties with Eq.(3.42), from which we obtain the body frame acceleration expression when the sensor placement is out of the center of mass of the vehicle.

$$\mathbf{a}^b = \mathbf{R}[\boldsymbol{\alpha}^w_{b/w} \times \mathbf{r}^w + \boldsymbol{\omega}^w_{b/w} \times \mathbf{v}^w + \boldsymbol{\alpha}^w_{b/w} \times \mathbf{r}^w_{s/b} + \boldsymbol{\omega}^w_{b/w} \times \boldsymbol{\omega}^w_{b/w} \times \mathbf{r}^w_{s/b}] \tag{3.51}$$
$$= \boldsymbol{\alpha}^b_{b/w} \times \mathbf{r}^b + \boldsymbol{\omega}^b_{b/w} \times \mathbf{v}^b + \boldsymbol{\alpha}^b_{b/w} \times \mathbf{r}^b_{s/b} + \boldsymbol{\omega}^b_{b/w} \times \boldsymbol{\omega}^b_{b/w} \times \mathbf{r}^b_{s/b} \tag{3.52}$$
$$= \dot{\mathbf{v}}^b + \boldsymbol{\omega}^b_{b/w} \times \mathbf{v}^b + \boldsymbol{\alpha}^b_{b/w} \times \mathbf{r}^b_{s/b} + \boldsymbol{\omega}^b_{b/w} \times \boldsymbol{\omega}^b_{b/w} \times \mathbf{r}^b_{s/b} \tag{3.53}$$

While the first two terms of previous equations are the ones related to the acceleration of the center of mass, the others are the additive terms due to relative position with respect to the center of mass.
In the vehicle case, we have that the mounting position of the sensors does not coincide with the center of mass of the vehicle, and then the measured acceleration is not the body one and needs to be corrected.
In a real scenario, we can compute $\mathbf{r}^b_{s/b}$ based on an estimate of the vehicle center of mass and correct the acceleration.
With this formula, we can also describe which acceleration is applied in every point of the body, to understand the forces that a particular part of the vehicle is subjected to.

3.2.2 Accelerometer description

Recalling the equation which describes the acceleration of the vehicle with respect to the body frame, we have that the accelerometer readings are modeled as

$$\mathbf{a}^b = \boldsymbol{\alpha}^b_{b/w} \times \mathbf{r}^b + \boldsymbol{\omega}^b_{b/w} \times \mathbf{v}^b + \mathbf{R}\mathbf{g}^w = \dot{\mathbf{v}}^b + \boldsymbol{\omega}^b_{b/w} \times \mathbf{v}^b + \mathbf{R}\mathbf{g}^w \tag{3.54}$$

where we suppose our sensor to be on the center of mass.
For an easier notation, we omit the notation referred to body frame, remembering that all quantities are expressed in body frame. The explicit version of the acceleration term coincides with

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} \dot{v}_x \\ \dot{v}_y \\ \dot{v}_z \end{bmatrix} + \begin{bmatrix} -\omega_z v_y + \omega_y v_z \\ +\omega_z v_x - \omega_x v_z \\ -\omega_y v_x + \omega_x v_y \end{bmatrix} + \begin{bmatrix} -\sin(\theta) g \\ \sin(\phi)\cos(\theta) g \\ \cos(\phi)\cos(\theta) g \end{bmatrix} \tag{3.55}$$

Starting from these model, the estimates of roll and pitch angles from Eq.(3.55) are equal to

$$\hat{\theta} = \text{asin}\left(\frac{\dot{v}_x - \omega_z v_y + \omega_y v_z - a_x}{g}\right) \tag{3.56}$$

$$\hat{\phi} = \text{atan}\left(\frac{a_y - \dot{v}_y - \omega_z v_x + \omega_x v_z}{a_z - \dot{v}_z + \omega_y v_x - \omega_x v_y}\right) \tag{3.57}$$

From the previous equations it is possible to find estimates for pitch and roll angles, based on gyroscope measurements and velocity measurements.

3.2.3 IMU noise characterization

In order to better characterize IMU sensors, it is useful to introduce the main noise sources IMU are subjected to.
Following the analysis in [13], main IMU errors are:

- Scale and misalignment: these errors are intrinsic in IMU sensors, and depend on their manufacturing technique. They will not be considered in this analysis.

- Constant bias: this error consists of a constant term which is added in every measure. This is relevant in gyroscopes, because their measures are related to angles through non-linear integration, making the constant bias to grow linearly with time. Since this type of error can be estimated when no rotations are applied to gyroscope, it can be compensated, and it will not be considered.

- Random source noise: this class includes all errors that arise due to random sources, and so that are not predictable or directly compensated. The two major contributes are:

  – Random additive gaussian noise: this error consists of an additive high frequency noise that is always present in all sensors, caused by thermomechanical events.

  – Random walk bias instability: the bias instability consist of a slowly changes over time due to flicker noise in the electronics and other effects. For this reason it is modeled as a random walk.

According to [13], simple continuous time model for an IMU sensor is considered, that is

$$\begin{cases} \tilde{\mathbf{y}}(t) = \mathbf{y}(t) + \mathbf{r}(t) + \mathbf{n_y}(t) \\ \dot{\mathbf{r}}(t) = \mathbf{n_r}(t) \end{cases} \tag{3.58}$$

where $\mathbf{y}$ is the ideal measure, $\tilde{\mathbf{y}}$ is the noisy measure, $\mathbf{r}$ is a random walk process and $\mathbf{n_y}, \mathbf{n_r}$ are gaussian random vectors.
The gaussian random vectors are equal to

$$\mathbf{n_y}(t) \sim \mathcal{N}(\mathbf{0}, \mathbf{R}), \quad \mathbf{n_r}(t) \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}) \tag{3.59}$$

It is assumed to have independence between different axis and also to have the same variance in all axis. With this assumption all covariance matrices are diagonal and depend only on one parameter.
The dynamical system of errors, that is useful in order to simulate noise, is

$$\begin{cases} \dot{\mathbf{r}}(t) = \mathbf{n_r}(t) \\ \mathbf{e}(t) = \mathbf{r}(t) + \mathbf{n_e}(t) \end{cases} \tag{3.60}$$

It is possible to write the previous system in explicit form, that is

$$\begin{cases} \begin{bmatrix} \dot{r}_x(t) \\ \dot{r}_y(t) \\ \dot{r}_z(t) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r_x(t) \\ r_y(t) \\ r_z(t) \end{bmatrix} + \begin{bmatrix} n_{r,x}(t) \\ n_{r,y}(t) \\ n_{r,z}(t) \end{bmatrix} \\ \begin{bmatrix} e_x(t) \\ e_y(t) \\ e_z(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_x(t) \\ r_y(t) \\ r_z(t) \end{bmatrix} + \begin{bmatrix} n_{y,x}(t) \\ n_{y,y}(t) \\ n_{y,z}(t) \end{bmatrix} \end{cases} \tag{3.61}$$

Since this information is used to simulate data, the discretized version of the system is needed. Sampling the system with sampling time T, and calling the state transition matrix and the output matrix as $\mathbf{A}, \mathbf{C}$ the discretized version are

$$\mathbf{A}_D = e^{\mathbf{A}T} = \mathbf{I} \tag{3.62}$$

$$\mathbf{C}_D = \mathbf{C} = \mathbf{I} \tag{3.63}$$

$$\mathbf{R}_D = \mathbf{R} \tag{3.64}$$

Regarding to covariance matrix of the state, this corresponds to

$$\mathbf{Q}_D = \int_{\tau=0}^{T} e^{\mathbf{A}\tau} \mathbf{Q} e^{\mathbf{A}^T T} d\tau = \int_{\tau=0}^{T} \mathbf{Q} d\tau = \mathrm{T}\mathbf{Q} \tag{3.65}$$

The discrete time propagation equations are equal to

$$\begin{cases} \mathbf{r}_{k+1} & = \mathbf{r}_k + \mathbf{n}_r, \quad \mathbf{n}_r \sim \mathcal{N}(\mathbf{0}, \mathrm{T}\mathbf{Q}) \\ \mathbf{e}_k & = \mathbf{r}_k + \mathbf{n}_e, \quad \mathbf{n}_r \sim \mathcal{N}(\mathbf{0}, \mathbf{R}) \end{cases} \tag{3.66}$$

Using this discretized system, measures from noisy sensors can be simulated by adding to ideal values this simulation error. The algorithm used to simulate noisy IMU measures is the following

Algorithm 3 Generation of noisy measures from noiseless ones

Require: $T, y_x, y_y, y_z, \sigma_r^2, \sigma_e^2$
  for $k = 0$ to $t-1$ **do**
    $\mathbf{n}_r \sim \mathcal{N}(\mathbf{0}, \mathrm{T}\mathbf{Q})$;
    $\mathbf{n}_e \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$;
    $\mathbf{r}_{k+1} = \mathbf{r}_k + \mathbf{n}_r$;
    $\mathbf{e}_{k+1} = \mathbf{r}_{k+1} + \mathbf{n}_e$;
    $\tilde{\mathbf{y}}_{k+1} = \mathbf{y}_{k+1} + \mathbf{e}_{k+1}$;
  end for
  return $\tilde{\mathbf{y}}_{1:t}$

3.2.4 Consideration with single IMU sensor estimates

In this chapter we evaluated the differences about estimates using only gyroscope or only accelerometer readings. The main differences between these sensors has been

- Gyroscope involves a differential equations, the estimates from angular velocities are related to previous values, which makes the estimate to be reasonable, but at the same time it is affected by constant or slowly variant errors, which make the system to diverge.
- Accelerometer estimates in a static condition, and then it is affected by high frequency errors, since it has no continuity between estimates at different sample instants.
- While gyroscope estimates all three angles, since accelerometer estimates are based on gravity, it can only estimates roll and pitch angle. The lack of a yaw estimate is not an issue in our case, since we are not interested in knowing the vehicle rotation about z-axis. Since z-axis rotation is the first rotation in our reference system, pitch and roll rotation are not affected by yaw angle, and then they are still correct even if we are not able to estimates z-axis rotation.

The key idea is then to fuse gyroscope and accelerometer readings in order to compute the attitude estimation based on a combination of the sensors, exploiting their properties.
In all methods we will give more importance to gyroscope readings: gyroscope measurements can be affected by errors but their continuous time equations are mathematically exacted, while accelerometer readings are affected by the position of the sensor with respect to the center of mass and they also depend on velocity measurements, that in a real scenario are estimated.
For this reasons, accelerometer readings are affected by errors, and then they will be used in order to correct the estimates provided by gyroscopes.

Chapter 4

Sensor Fusion Estimation

In previous chapter we evaluated the main characteristics of attitude estimation based on single IMU measurement. Since gyroscope and accelerometer present complementary properties, the idea is to combine information about both sensors to compute the attitude estimation.
These methods, known as sensor fusion estimation, are well described in literature, as they are implemented following different approaches.
It is important to notice that all these methods are computed under the assumption of having null external forces. This translates in a strong constraints about the motion that the body can do, since in order to have negligible external forces it has to move mainly under uniform motion. With this approximation the accelerometer only measures the gravity force, and then it is used as an absolute direction of the gravity vector.
In [2] the problem is solved using a complementary filter which first solves the problem for accelerometer and gyroscope separately, and after it weights each estimate to obtain a fused one.
The complementary filter is the simplest fuse algorithm, since it only combines information obtained by different IMU in a static way.
The general equation of a complementary filter is

$$\hat{\boldsymbol{\xi}}_k = \beta \hat{\boldsymbol{\xi}}_k^g + (1-\beta)\hat{\boldsymbol{\xi}}_k^a, \quad \beta \in [0,1] \tag{4.1}$$

where $\hat{\boldsymbol{\xi}}_k^g$ and $\hat{\boldsymbol{\xi}}_k^a$ are the estimates from gyroscope and accelerometer respectively, and β is a filter coefficient which weights the measures, and has to be optimized.
This type of filter is simple but it does not consider any correction method, and so it is strongly dependent on the value of the weight. Since accelerometer measures are difficult to describe, the weighting parameter will be set very close to 1.
In [14],[15], attitude estimation problem is solved by finding an optimal rotation matrix, instead of finding directly the estimates of the angles.
These methods are based on finding an optimal matrix in SO(3) under gyroscope and accelerometer constraints. This translates into an optimization problem that is

$$\hat{\mathbf{R}} = \underset{\mathbf{R}\in SO(3)}{\arg\min}(\lambda_1 ||\mathbf{a} - \mathbf{R}\mathbf{g}||^2) \tag{4.2}$$

$$s.t. \quad \dot{\mathbf{R}} = [-\boldsymbol{\omega}]_\times \mathbf{R} \tag{4.3}$$

The problem is solved in suboptimal manner, applying gyroscope and accelerometer condition in sequence. Since the accelerometer is able to describe only two variables with respect to the three that a rotation matrix depends on, the problem is ill-defined.
In these papers, this issue is solved by inserting a third IMU sensor, that is the magnetometer.

The magnetometer is used because it describes the absolute direction of the earth magnetic field, and then it has information about z-axis rotation, which is the one that accelerometer is invariant to.
With the addition of the magnetometer the problem becomes well-defined, and then the optimization problem is solved as

$$\hat{\mathbf{R}} = \underset{\mathbf{R}\in SO(3)}{\arg\min}(\lambda_1||\mathbf{a} - \mathbf{R}\mathbf{g}||^2 + \lambda_2||\mathbf{m} - \mathbf{R}\mathbf{e}||^2) \tag{4.4}$$

$$s.t. \quad \dot{\mathbf{R}} = [-\boldsymbol{\omega}]_\times \mathbf{R} \tag{4.5}$$

where $\mathbf{m}$ is the measure from magnetometer, and $\mathbf{e}$ is the earth magnetic field direction.
Since magnetometer measures are sensitive to external electromagnetic fields, in the case of motorcycle, we will not assume to have this sensor, because it would be affected by too high noise.
The aim is then to develop a sensor fusion algorithm which has the following properties:

- It is based only on gyroscope and accelerometer measurements, while magnetometer sensor is assumed to be absent.
- It is based on velocity measures, it includes information about acceleration, and then it is not based on the assumption of treating accelerometer data as gravity direction vector.
- It has to combine IMU information in a more complete way with respect to complementary filter.

This problem is analyzed in [10], where the accelerometer outputs are described adding the longitudinal and centripetal acceleration beside gravity. The chosen fusion algorithm is an Extended Kalman Filter, which implicitly combines IMU data.
Sensor fusion problem solution can start from this method, with some advances regarding the addition of more information about acceleration characterization, and different fusion algorithms.
The fusion algorithm will be derived always with an optimization approach, in particular:

- The maximum a posteriori problem is derived, based on a non-linear system.
- Kalman filter and Extended Kalman Filter equations are derived as an optimization problem.

4.1 MAP estimation

Consider a discrete time non linear system with additive white gaussian noise,

$$\begin{cases} \mathbf{x}_{k+1} & = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{w}_k, \quad \mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}) \\ \mathbf{y}_k & = \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{n}_k, \quad \mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}) \end{cases} \tag{4.6}$$

The aim is to estimate the state based on state propagation and measurements, which means that a full information estimator is needed, which translates into finding the maximum of the posterior distribution, that is

$$\{\hat{\mathbf{x}}_0, \hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_t\} = \underset{\{\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_t\}}{\arg\max}\ p(\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_t | \mathbf{y}_0, \mathbf{y}_1, \dots, \mathbf{y}_t) \tag{4.7}$$

The posterior distribution can be written in compact form, defining

$$p(\mathbf{x}_{0:t}|\mathbf{y}_{0:t}) \doteq p(\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_t | \mathbf{y}_0, \mathbf{y}_1, \dots, \mathbf{y}_t) \tag{4.8}$$

Applying Bayes rule it is possible to express the posterior distribution as

$$p(\mathbf{x}_{0:t}|\mathbf{y}_{0:t}) = \frac{p(\mathbf{x}_{0:t}, \mathbf{y}_{0:t})}{p(\mathbf{y}_{0:t})} = \frac{p(\mathbf{y}_t|\mathbf{x}_{0:t}, \mathbf{y}_{0:t-1})p(\mathbf{x}_{0:t}, \mathbf{y}_{0:t-1})}{p(\mathbf{y}_{0:t})} \tag{4.9}$$

$$= \frac{p(\mathbf{y}_t|\mathbf{x}_{0:t}, \mathbf{y}_{0:t-1})p(\mathbf{y}_{t-1}|\mathbf{x}_{0:t}, \mathbf{y}_{0:t-2})p(\mathbf{x}_{0:t}, \mathbf{y}_{0:t-2})}{p(\mathbf{y}_{0:t})} \tag{4.10}$$

$$= \frac{p(\mathbf{y}_t|\mathbf{x}_{0:t}, \mathbf{y}_{0:t-1})p(\mathbf{y}_{t-1}|\mathbf{x}_{0:t}, \mathbf{y}_{0:t-2}) \dots p(\mathbf{y}_0|\mathbf{x}_{0:t})p(\mathbf{x}_{0:t})}{p(\mathbf{y}_{0:t})} \tag{4.11}$$

$$= \frac{p(\mathbf{y}_t|\mathbf{x}_{0:t}, \mathbf{y}_{0:t-1})p(\mathbf{y}_{t-1}|\mathbf{x}_{0:t}, \mathbf{y}_{0:t-2}) \dots p(\mathbf{y}_0|\mathbf{x}_{0:t})p(\mathbf{x}_t|\mathbf{x}_{0:t-1})p(\mathbf{x}_{0:t-1})}{p(\mathbf{y}_{0:t})} \tag{4.12}$$

$$= \frac{p(\mathbf{y}_t|\mathbf{x}_{0:t}, \mathbf{y}_{0:t-1})p(\mathbf{y}_{t-1}|\mathbf{x}_{0:t}, \mathbf{y}_{0:t-2}) \dots p(\mathbf{y}_0|\mathbf{x}_{0:t})p(\mathbf{x}_t|\mathbf{x}_{0:t-1})p(\mathbf{x}_{t-1}|\mathbf{x}_{0:t-2}) \dots p(\mathbf{x}_0)}{p(\mathbf{y}_{0:t})} \tag{4.13}$$

Previous relation can be expressed by a product of sequence as

$$p(\mathbf{x}_{0:t}|\mathbf{y}_{0:t}) = \frac{p(\mathbf{y}_0|\mathbf{x}_{0:t})p(\mathbf{x}_0)\prod_{k=1}^{t}[p(\mathbf{y}_k|\mathbf{x}_{0:k}, \mathbf{y}_{0:k-1})p(\mathbf{x}_k|\mathbf{x}_{0:k-1})]}{p(\mathbf{y}_{0:t})} \tag{4.14}$$

It is now necessary to evaluate the previous expression, based on system in Eq.(4.6). Since noise at different time instants are independent, the conditioned probabilities are equal to

$$p(\mathbf{y}_k|\mathbf{x}_{0:k}, \mathbf{y}_{0:k-1}) = p(\mathbf{y}_k|\mathbf{x}_k) \tag{4.15}$$

$$p(\mathbf{x}_k|\mathbf{x}_{0:k-1}) = p(\mathbf{x}_k|\mathbf{x}_{k-1}) \tag{4.16}$$

Based on these properties, the posterior distribution of Eq.(4.14) is

$$p(\mathbf{x}_{0:t}|\mathbf{y}_{0:t}) = \frac{p(\mathbf{x}_0)\prod_{k=0}^{t} p(\mathbf{y}_k|\mathbf{x}_k)\prod_{k=0}^{t-1} p(\mathbf{x}_{k+1}|\mathbf{x}_k)}{p(\mathbf{y}_{0:t})} \tag{4.17}$$

In maximizing the posterior distribution over $\mathbf{x}$, the denominator of Eq.(4.17) does not affect the maximization, and then only its numerator is considered. Then, defining

$$\tilde{p}(\mathbf{x}_{0:t}|\mathbf{y}_{0:t}) \doteq p(\mathbf{x}_0)\prod_{k=0}^{t} p(\mathbf{y}_k|\mathbf{x}_k)\prod_{k=0}^{t-1} p(\mathbf{x}_{k+1}|\mathbf{x}_k) \propto p(\mathbf{x}_{0:t}|\mathbf{y}_{0:t}) \tag{4.18}$$

the maximization problem becomes

$$\underset{\{\mathbf{x}_0,\mathbf{x}_1,\ldots,\mathbf{x}_t\}}{\arg\max}\ \tilde{p}(\mathbf{x}_{0:t}|\mathbf{y}_{0:t}) = \underset{\{\mathbf{x}_0,\mathbf{x}_1,\ldots,\mathbf{x}_t\}}{\arg\max}\ p(\mathbf{x}_{0:t}|\mathbf{y}_{0:t}) \tag{4.19}$$

Applying *log* function to $\tilde{p}(\mathbf{x}_{0:t}|\mathbf{y}_{0:t})$ it becomes

$$\begin{aligned}\log[\tilde{p}(\mathbf{x}_{0:t}|\mathbf{y}_{0:t})] &= \log[p(\mathbf{x}_0)\prod_{k=0}^{t} p(\mathbf{y}_k|\mathbf{x}_k)\prod_{k=0}^{t-1} p(\mathbf{x}_{k+1}|\mathbf{x}_k)] &(4.20)\\ &= \log[p(\mathbf{x}_0)] + \sum_{k=0}^{t}\log[p(\mathbf{y}_k|\mathbf{x}_k)] + \sum_{k=0}^{t-1}\log[p(\mathbf{x}_{k+1}|\mathbf{x}_k)] &(4.21)\end{aligned}$$

The previous relation is based on conditioned probabilities expressions, and then it is necessary to evaluate them. Starting from noise Gaussian distribution one has that

$$\begin{aligned}\mathbf{y}_k &\sim \mathcal{N}(\mathbf{h}_k(\mathbf{x}_k,\mathbf{u}_k),\mathbf{R}) &(4.22)\\ \mathbf{x}_{k+1} &\sim \mathcal{N}(\mathbf{f}_k(\mathbf{x}_k,\mathbf{u}_k),\mathbf{Q}) &(4.23)\end{aligned}$$

which means that

$$\begin{aligned}p(\mathbf{y}_k|\mathbf{x}_k) &= \frac{1}{\sqrt{\det(2\pi\mathbf{R})}}e^{-\frac{1}{2}(\mathbf{y}_k-\mathbf{h}_k(\mathbf{x}_k,\mathbf{u}_k))^T\mathbf{R}^{-1}(\mathbf{y}_k-\mathbf{h}_k(\mathbf{x}_k,\mathbf{u}_k))} &(4.24)\\ p(\mathbf{x}_{k+1}|\mathbf{x}_k) &= \frac{1}{\sqrt{\det(2\pi\mathbf{Q})}}e^{-\frac{1}{2}(\mathbf{x}_{k+1}-\mathbf{f}_k(\mathbf{x}_k,\mathbf{u}_k))^T\mathbf{Q}^{-1}(\mathbf{x}_{k+1}-\mathbf{f}_k(\mathbf{x}_k,\mathbf{u}_k))} &(4.25)\\ p(\mathbf{x}_0) &= \frac{1}{\sqrt{\det(2\pi\mathbf{P}_0)}}e^{-\frac{1}{2}(\mathbf{x}_0-\boldsymbol{\mu}_0)^T\mathbf{P}_0^{-1}(\mathbf{x}_0-\boldsymbol{\mu}_0)} &(4.26)\end{aligned}$$

The log function applied to previous densities gives

$$\begin{aligned}\log[p(\mathbf{y}_k|\mathbf{x}_k)] &= -\frac{1}{2}(\mathbf{y}_k-\mathbf{h}_k(\mathbf{x}_k,\mathbf{u}_k))^T\mathbf{R}^{-1}(\mathbf{y}_k-\mathbf{h}_k(\mathbf{x}_k,\mathbf{u}_k)) + c &(4.27)\\ \log[p(\mathbf{x}_{k+1}|\mathbf{x}_k)] &= -\frac{1}{2}(\mathbf{x}_{k+1}-\mathbf{f}_k(\mathbf{x}_k,\mathbf{u}_k))^T\mathbf{Q}^{-1}(\mathbf{x}_{k+1}-\mathbf{f}_k(\mathbf{x}_k,\mathbf{u}_k)) + c &(4.28)\\ \log[p(\mathbf{x}_0)] &= -\frac{1}{2}(\mathbf{x}_0-\boldsymbol{\mu}_0)^T\mathbf{P}_0^{-1}(\mathbf{x}_0-\boldsymbol{\mu}_0) + c &(4.29)\end{aligned}$$

where c is made of terms which does not depend on $\mathbf{x}$ and then they are considered as constants is the maximum problem.
With these results, taking the -log function, the maximization problem is equal to

$$\underset{\{\mathbf{x}_0,\mathbf{x}_1,\ldots,\mathbf{x}_t\}}{\arg\max}\ \tilde{p}(\mathbf{x}_{0:t}|\mathbf{y}_{0:t}) = \underset{\{\mathbf{x}_0,\mathbf{x}_1,\ldots,\mathbf{x}_t\}}{\arg\max}\ \log[\tilde{p}(\mathbf{x}_{0:t}|\mathbf{y}_{0:t})] = \underset{\{\mathbf{x}_0,\mathbf{x}_1,\ldots,\mathbf{x}_t\}}{\arg\min}\ -\log[\tilde{p}(\mathbf{x}_{0:t}|\mathbf{y}_{0:t})] \tag{4.30}$$

and Eq.(4.21) becomes

$$\underset{\{\mathbf{x}_0,\mathbf{x}_1,\ldots,\mathbf{x}_t\}}{\arg\min}\ \frac{1}{2}||\mathbf{x}_0-\boldsymbol{\mu}_0||^2_{\mathbf{P}_0^{-1}} + \frac{1}{2}\sum_{k=0}^{t}||\mathbf{y}_k-\mathbf{h}(\mathbf{x}_k,\mathbf{u}_k)||^2_{\mathbf{R}^{-1}} + \frac{1}{2}\sum_{k=0}^{t-1}||\mathbf{x}_{k+1}-\mathbf{f}(\mathbf{x}_k,\mathbf{u}_k)||^2_{\mathbf{Q}^{-1}} \tag{4.31}$$

This means that maximizing the posterior distribution translates into solving a non linear least square problem.

This state estimation is known as full information estimation, which is associated with cost index

$$J_{0:t}(\mathbf{X}) = \frac{1}{2}||\mathbf{x}_0 - \boldsymbol{\mu}_0||^2_{\mathbf{P}_0^{-1}} + \frac{1}{2}\sum_{k=0}^{t}||\mathbf{y}_k - \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k)||^2_{\mathbf{R}^{-1}} + \frac{1}{2}\sum_{k=0}^{t-1}||\mathbf{x}_{k+1} - \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)||^2_{\mathbf{Q}^{-1}} \quad (4.32)$$

Depending on the assumption and the form of the system, the problem can be solved in different ways. The main problems that arise are:

- Kalman Filter (KF): if the system is linear both in state and measure equations, then it is possible to find a recursive closed form algorithm that update the estimate based only on previous and actual step variables.

- Extended Kalman Filter (EKF): if the system is nonlinear we can proceed as the standard kalman filter whose matrices are the linearization of the system at the current step. Even in this case, it is possible to derive a recursive algorithm that is based only on previous and actual step variables.

- Moving Horizon Estimation (MHE): if the system is nonlinear we can solve the optimization problem in a longer horizon, instead of consider only one step before, as EKF does. This procedure cannot be solved in closed form, in fact the minimization is done by using a mathematical programming technique. In the particular case of a linear system, moving horizon estimation coincides with smoothed Kalman filter.

4.2 Kalman Filter from MAP problem

Starting from the quadratic cost index as in Eq.(4.32), it is possible to find the closed form solution in the case of linear system. This can be an alternative derivation of the Kalman filter, based on an optimization approach, instead of the statistical derivation.
Consider the system

$$\begin{cases} \mathbf{x}_{k+1} &= \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k \\ \mathbf{y}_k &= \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{n}_k \end{cases} \quad (4.33)$$

where $\mathbf{x} \in \mathbb{R}^n, \mathbf{u} \in \mathbb{R}^m, \mathbf{y} \in \mathbb{R}^p$, and

$$E[\mathbf{w}(k)] = 0, \quad E[\mathbf{w}(k)\mathbf{w}(k)^T] = \mathbf{Q}_k > 0 \quad (4.34)$$

$$E[\mathbf{v}(k)] = 0, \quad E[\mathbf{v}(k)\mathbf{v}(k)^T] = \mathbf{R}_k > 0 \quad (4.35)$$

$$E[\mathbf{x}(0)] = \boldsymbol{\mu}_0, \quad E[\mathbf{x}(0)\mathbf{x}(0)^T] = \mathbf{P}_0 > 0 \quad (4.36)$$

together with independence of noise at different time instants. Let's assume to have the same number of inputs and observations, that are

$$\{\mathbf{y}_1, \ldots, \mathbf{y}_k\}, \quad \{\mathbf{u}_0, \ldots, \mathbf{u}_{k-1}\} \quad (4.37)$$

Starting from the initial condition, it is possible to write a recursion for the state and the output, that is

$$\mathbf{x}_0 = \boldsymbol{\mu}_0 + \mathbf{w}_{-1} \quad \Rightarrow \quad \boldsymbol{\mu}_0 = \mathbf{x}_0 - \mathbf{w}_{-1} \tag{4.38}$$

$$\mathbf{x}_1 = \mathbf{F}\mathbf{x}_0 + \mathbf{G}\mathbf{u}_0 + \mathbf{w}_0 \quad \Rightarrow \quad \mathbf{G}\mathbf{u}_0 = \mathbf{x}_1 - \mathbf{F}\mathbf{x}_0 - \mathbf{w}_0 \tag{4.39}$$

$$\mathbf{y}_1 = \mathbf{H}\mathbf{x}_1 + \mathbf{D}\mathbf{u}_1 + \mathbf{v}_1 \quad \Rightarrow \quad \mathbf{y}_1 - \mathbf{D}\mathbf{u}_1 = \mathbf{H}\mathbf{x}_1 + \mathbf{v}_1 \tag{4.40}$$

$$\vdots \tag{4.41}$$

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{G}\mathbf{u}_{k-1} + \mathbf{w}_{k-1} \quad \Rightarrow \quad \mathbf{G}\mathbf{u}_{k-1} = \mathbf{x}_k - \mathbf{F}\mathbf{x}_{k-1} - \mathbf{w}_{k-1} \tag{4.42}$$

$$\mathbf{y}_k = \mathbf{H}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{v}_k \quad \Rightarrow \quad \mathbf{y}_k - \mathbf{D}\mathbf{u}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k \tag{4.43}$$

From the previous series of equations, it is possible to write the right part as a large linear system, that is

$$\mathbf{b}_k = \boldsymbol{\Phi}_k \mathbf{z}_k + \boldsymbol{\epsilon}_k \tag{4.44}$$

where

$$\mathbf{b}_k = \begin{bmatrix} \boldsymbol{\mu}_0 \\ \mathbf{G}\mathbf{u}_0 \\ \mathbf{y}_1 - \mathbf{D}\mathbf{u}_1 \\ \vdots \\ \mathbf{G}\mathbf{u}_{k-2} \\ \mathbf{y}_{k-1} - \mathbf{D}\mathbf{u}_{k-1} \\ \mathbf{G}\mathbf{u}_{k-1} \\ \mathbf{y}_k - \mathbf{D}\mathbf{u}_k \end{bmatrix} \quad \boldsymbol{\Phi}_k = \begin{bmatrix} 1 & 0 & 0 & & & & \\ -\mathbf{F} & 1 & 0 & & \cdots & & \\ 0 & \mathbf{H} & 0 & & & & \\ & & & \ddots & & & \\ & & & & -\mathbf{F} & 1 & 0 \\ & & & & 0 & \mathbf{H} & 0 \\ & \vdots & & & 0 & -\mathbf{F} & 1 \\ & & & & 0 & 0 & \mathbf{H} \end{bmatrix} \quad \mathbf{z}_k = \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_{k-1} \\ \mathbf{x}_k \end{bmatrix} \quad \epsilon_k = \begin{bmatrix} \mathbf{w}_{-1} \\ \mathbf{w}_0 \\ \mathbf{v}_1 \\ \vdots \\ \mathbf{w}_{k-2} \\ \mathbf{v}_{k-1} \\ \mathbf{w}_{k-1} \\ \mathbf{v}_k \end{bmatrix} \tag{4.45}$$

and the covariance matrix of $\boldsymbol{\epsilon}$ is

$$\mathbf{W}_k = \operatorname{diag}(\mathbf{P}_0^{-1}, \mathbf{Q}_0^{-1}, \mathbf{R}_1^{-1}, \ldots, , \mathbf{Q}_{k-1}^{-1}, \mathbf{R}_k^{-1}) \tag{4.46}$$

Since Eq.(4.44) is a linear system, its least square solution can be found by minimizing the cost index

$$J(\mathbf{z}_k) = \frac{1}{2} ||\boldsymbol{\Phi}_k \mathbf{z}_k - \mathbf{b}_k||_{\mathbf{W}}^2 \tag{4.47}$$

The LS solution for linear system, corresponds to

$$\hat{\mathbf{z}}_k = (\boldsymbol{\Phi}_k^T \mathbf{W}_k \boldsymbol{\Phi}_k)^{-1} \boldsymbol{\Phi}_{\boldsymbol{k}}^T \mathbf{W}_k \mathbf{b}_k \tag{4.48}$$

where the inverse term in Eq.(4.48) is possible because $\boldsymbol{\Phi}_k$ never loses rank.
The solution of this LS problem yields to smoothed estimates, given all observation until k, that means that we obtain

$$[\hat{\mathbf{x}}_{0|k}, \hat{\mathbf{x}}_{1|k}, \ldots, \hat{\mathbf{x}}_{k|k}]^T = (\boldsymbol{\Phi}_k^T \mathbf{W}_k \boldsymbol{\Phi})_k^{-1} \boldsymbol{\Phi}_{\boldsymbol{k}}^T \mathbf{W}_k \mathbf{b}_k \tag{4.49}$$

In [16] it is proven that in the case of linear system, the solution of this minimization corresponds to the smoothed Kalman filter estimation.
With this results we have a derivation for the Kalman filter which does not use any statistical property. This means that the Kalman filter is the best unbiased linear estimator, which translates into minimum error covariance, and at the same time is the optimal filter in the case that we have Gaussian distribution.
Since Eq.(4.48) is quite complex to solve due to its linear time dependent dimension, it is possible to derive a recursive version for solving the problem in (4.32), obtaining the same equation as the standard Kalman filter.

4.2.1 Kalman Filter derivation from quadratic cost function

Recursive Kalman filter equations will be derived, where we suppose to have a discrete time linear system, that is

$$\begin{cases} \mathbf{x}_{k+1} &= \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k \\ \mathbf{y}_k &= \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{n}_k \end{cases} \tag{4.50}$$

The idea is to recursively solve the LS problem, adding one term to the cost index at each iteration, knowing that previous estimates are optimal for the previous cost index. This method is known as Recursive Least Squares (RLS).
It is important to notice that solving an RLS problem does not solve the entire cost index: indeed, while with the RLS the Kalman filter equations are derived, with the entire LS problem Kalman smoother equations are obtained.
The minimization of the LS problem at each time instant is performed by Gauss-Newton method, that is

$$\mathbf{x}_{k+1} = \mathbf{x}_k - [\nabla^2 J(\mathbf{x}_k)]^{-1} \nabla J(\mathbf{x}_k) \tag{4.51}$$

Since the cost function J is quadratic, a single iteration of Gauss-Newton descent yields to the minimum. The complete solution of RLS problem is shown in [17].
The cost function with linear system becomes

$$J_{t|t} = \frac{1}{2}||\boldsymbol{\mu}_0 - \mathbf{x}_0||^2_{\mathbf{Q}_0^{-1}} + \frac{1}{2}\sum_{k=1}^{t}||\mathbf{y}_k - \mathbf{C}\mathbf{x}_k - \mathbf{D}\mathbf{u}_k||^2_{\mathbf{R}_k^{-1}} + \frac{1}{2}\sum_{k=1}^{t}||\mathbf{x}_k - \mathbf{A}\mathbf{x}_{k-1} - \mathbf{B}\mathbf{u}_{k-1}||^2_{\mathbf{Q}_k^{-1}} \tag{4.52}$$

The kalman filter will be derived following the two steps approach as in [17], that are prediction and correction. Let's define

$$F = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ \mathbf{A} \end{bmatrix}, \quad \mathbf{z}_k = \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_k \end{bmatrix} \tag{4.53}$$

The system index can be propagated through

$$J_{k|k-1} = J_{k-1|k-1} + \frac{1}{2}||\mathbf{x}_k - \mathbf{F}\mathbf{z}_{k-1} - \mathbf{B}\mathbf{u}_k||^2_{\mathbf{Q}_k^{-1}} \tag{4.54}$$

where the cost index depends on $\mathbf{z}_{k-1}$ and $\mathbf{x}_k$. The computation of both gradient and hessian of J yields to

$$\nabla J_{k|k-1}(\mathbf{z}_k) = \begin{bmatrix} \nabla J_{k-1|k-1}(\mathbf{z}_{k-1}) + \mathbf{F}_k^T\mathbf{Q}_k^{-1}(\mathbf{x}_k - \mathbf{F}\mathbf{z}_{k-1} - \mathbf{B}\mathbf{u}_k) \\ -\mathbf{Q}_k^{-1}(\mathbf{x}_k - \mathbf{F}\mathbf{z}_{k-1} - \mathbf{B}\mathbf{u}_k) \end{bmatrix} \tag{4.55}$$

$$\nabla^2 J_{k|k-1}(\mathbf{z}_k) = \begin{bmatrix} \nabla^2 J_{k-1|k-1}(\mathbf{z}_{k-1}) + \mathbf{F}_k^T\mathbf{Q}_k^{-1}\mathbf{F}_k & -\mathbf{F}_k^T\mathbf{Q}_k^{-1} \\ -\mathbf{Q}_k^{-1}\mathbf{F}_k & \mathbf{Q}_k^{-1} \end{bmatrix} \tag{4.56}$$

Since $\nabla^2 J_{k|k-1}(\mathbf{z}_k)$ is positive definite, a single iteration of Newton's method makes the cost function to converge to the minimum, which is

$$\hat{\mathbf{z}}_{k|k-1} = \mathbf{z}_k - \nabla^2 J_{k|k-1}^{-1}(\mathbf{z}_k)\nabla J_{k|k-1}(\mathbf{z}_k) \tag{4.57}$$

A good initial guess for the Newton's method is

$$\mathbf{z}_k = \begin{bmatrix} \hat{\mathbf{z}}_{k-1|k-1} \\ \mathbf{A}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}\mathbf{u}_k \end{bmatrix} \tag{4.58}$$

The optimal estimate given the measurements $\mathbf{y}_0, \dots, \mathbf{y}_{k-1}$, is

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{A}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}\mathbf{u}_k \tag{4.59}$$

The estimate of the covariance matrix coincides with the bottom right block of the hessian matrix, which is, from matrix inversion property

$$\mathbf{P}_{k|k-1} = [\nabla^2 J_{k|k-1}(\mathbf{z}_k)]^{-1}_{2,2} \tag{4.60}$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_k \mathbf{P}_{k-1|k-1} \mathbf{F}_k^T + \mathbf{Q} \tag{4.61}$$

After we measure $\mathbf{y}_k$ we can update and correct the estimate with the new measure. Let's define $\mathbf{H} = [0 \dots 0 \ \mathbf{C}]^T$.
The update objective function becomes

$$J_{k|k} = J_{k|k-1} + \frac{1}{2}||\mathbf{y}_k - \mathbf{H}\mathbf{z}_k - \mathbf{D}\mathbf{u}_k||^2_{\mathbf{R}_k^{-1}} \tag{4.62}$$

The gradient and Hessian of the correction step index are

$$\nabla J_{k|k}(\mathbf{z}_k) = \nabla J_{k|k-1}(\mathbf{z}_k) + \mathbf{H}^T\mathbf{R}^{-1}(\mathbf{y}_k - \mathbf{H}\mathbf{z}_k - \mathbf{D}\mathbf{u}_k) \tag{4.63}$$

$$\nabla^2 J_{k|k}(\mathbf{z}_k) = \nabla^2 J_{k|k-1}(\mathbf{z}_k) + \mathbf{H}^T\mathbf{R}^{-1}\mathbf{H} \tag{4.64}$$

Also this time the Hessian is positive definite and then a single Newton's iteration is enough to converge to the minimum. Again, if we choose the initial guess as $\mathbf{z}_{k|k} = \mathbf{z}_{k|k-1}$ we obtain that

$$\mathbf{P}_{k|k} = (\mathbf{P}^{-1}_{k|k-1} + \mathbf{C}^T\mathbf{R}^{-1}\mathbf{C})^{-1} \tag{4.65}$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{P}_{k|k}\mathbf{C}^T\mathbf{R}^{-1}(\mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_{k|k-1} - \mathbf{D}\mathbf{u}_k) \tag{4.66}$$

We proved that Kalman filter equations can also be derived by a quadratic optimization approach, in the case of linear system. It is important to notice that no probabilistic model has been used, in fact the derivation is not dependent from the noise source.

4.3 EKF from quadratic optimization

From the previous derivation of the Kalman filter it is possible also to derive the Extended Kalman Filter, recalling that we approximate the covariance matrix with the one of the linearized system.
We now derive the EKF recursive algorithm starting from Eq.(4.52) extended to the non-linear case. Also in this case we divide the procedure in two parts, that are propagation and correction, with their associated cost functions.
Starting from the prediction step, we consider the cost function

$$J_{k|k-1}(\mathbf{z}_k) = J_{k-1|k-1}(\mathbf{z}_{k-1}) + \frac{1}{2}||\mathbf{x}_k - \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_k)||^2_{\mathbf{Q}_k^{-1}} \tag{4.67}$$

Computing the gradient and hessian of $J_{k|k-1}(\mathbf{z}_k)$ yields

$$\nabla J_{k|k-1}(\mathbf{z}_k) = \begin{bmatrix} \nabla J_{k-1|k-1}(\mathbf{z}_{k-1}) - \nabla \mathbf{f}^T(\mathbf{x}_{k-1}, \mathbf{u}_k)\mathbf{Q}^{-1}(\mathbf{x}_k - \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_k)) \\ \mathbf{Q}^{-1}(\mathbf{x}_k - \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_k)) \end{bmatrix} \tag{4.68}$$

$$\nabla^2 J_{k|k-1}(\mathbf{z}_k) = \begin{bmatrix} \nabla^2 J_{k-1|k-1}(\mathbf{z}_{k-1}) + \mathbf{\Theta}(\mathbf{z}_k) & \nabla \mathbf{f}^T(\mathbf{x}_{k-1}, \mathbf{u}_k)\mathbf{Q}^{-1} \\ \mathbf{Q}^{-1}\nabla \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_k) & \mathbf{Q}^{-1} \end{bmatrix} \tag{4.69}$$

where

$$\mathbf{\Theta}(\mathbf{z}_k) = \nabla^2 \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_k)\mathbf{Q}^{-1}(\mathbf{x}_k - \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_k)) + \nabla \mathbf{f}^T(\mathbf{x}_{k-1}, \mathbf{u}_k)\mathbf{Q}^{-1}\nabla \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_k) \tag{4.70}$$

We want to find $\hat{\mathbf{z}}_{k|k-1}$ such that $\nabla J_{k|k-1}(\mathbf{z}_k) = 0$. This clearly happens when

$$\hat{\mathbf{z}}_{k|k-1} = \begin{bmatrix} \hat{\mathbf{z}}_{k-1|k-1} \\ \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_k) \end{bmatrix} \tag{4.71}$$

which means that the propagated state is equal to

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_k) \tag{4.72}$$

The covariance state estimate is the lower right block of hessian inverse, that is

$$\mathbf{P}_{k|k-1} = \mathbf{Q}^{-1} + \nabla \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_k)\mathbf{P}_{k-1|k-1}\nabla \mathbf{f}^T(\mathbf{x}_{k-1}, \mathbf{u}_k) \tag{4.73}$$

Regarding to correction step, we have to minimize the cost function

$$J_{k|k} = J_{k|k-1} + \frac{1}{2}||\mathbf{y}_k - \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k)||^2_{\mathbf{R}_k^{-1}} \tag{4.74}$$

Again, we compute the gradient and hessian, which are equal to

$$\nabla J_{k|k}(\mathbf{z}_k) = \nabla J_{k|k-1}(\mathbf{z}_k) - \nabla \mathbf{h}^T(\mathbf{x}_k, \mathbf{u}_k)\mathbf{R}^{-1}(\mathbf{y}_k - \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k)) \tag{4.75}$$

$$\nabla^2 J_{k|k}(\mathbf{z}_k) = \nabla^2 J_{k|k-1}(\mathbf{z}_k) - \nabla^2 \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k)\mathbf{R}^{-1}(\mathbf{y}_k - \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k)) + \nabla \mathbf{h}^T(\mathbf{x}_k, \mathbf{u}_k)\mathbf{R}^{-1}\nabla \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \tag{4.76}$$

Since we have that, after the prediction step we have $\mathbf{z}_k = \hat{\mathbf{z}}_{k|k-1}$, then, from Newton's method we obtain

$$\hat{\mathbf{z}}_{k|k} = \hat{\mathbf{z}}_{k|k-1} - [\nabla^2 J_{k|k}(\hat{\mathbf{z}}_{k|k-1})]^{-1}\nabla J_{k|k}(\hat{\mathbf{z}}_{k|k-1}) \tag{4.77}$$

while the updated covariance matrix is given by the hessian inverse, which is

$$\mathbf{P}_{k|k} = [\mathbf{P}^{-1}_{k|k-1} + \nabla \mathbf{h}^T(\mathbf{x}_k, \mathbf{u}_k)\mathbf{R}^{-1}\nabla \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k)]^{-1} \tag{4.78}$$

Finally, the corrected estimate, that is the last row of the Newton equation is

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{P}_{k|k}\nabla \mathbf{h}^T(\mathbf{x}_k, \mathbf{u}_k)\mathbf{R}^{-1}(\mathbf{y}_k - \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k)) \tag{4.79}$$

This is the standard formulation for a discrete time EKF, but since our system will evolve in continuous time we need some variant of EKF, which are able to handle hybrid systems.

4.3.1 Extended Kalman Filter Estimation

Consider a nonlinear discrete time system with additive gaussian noise

$$\begin{cases} \mathbf{x}_{k+1} & = f(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{w}_k, \quad \mathbf{w}_k \sim \mathcal{N}(0, \mathbf{Q}_k) \\ \mathbf{y}_k & = h(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{v}_k, \quad \mathbf{v}_k \sim \mathcal{N}(0, \mathbf{R}_k) \end{cases} \tag{4.80}$$

The problem of state estimation can be solved via Extended Kalman Filter (EKF) as derived in previous section as an approximation of the solution of the MAP problem. The final formulation is

$$E[\mathbf{x}_0] = \boldsymbol{\mu}_0, \quad \mathrm{Var}[\mathbf{x}_0] = \mathbf{P}_0 \tag{4.81}$$

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{f}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}) \tag{4.82}$$

$$\hat{\mathbf{y}}_k = \mathbf{h}(\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k) \tag{4.83}$$

$$\mathbf{A}_{k-1} = \frac{\partial \mathbf{f}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1})}{\partial \mathbf{x}}, \quad \mathbf{C}_k = \frac{\partial \mathbf{h}(\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k)}{\partial \mathbf{x}} \tag{4.84}$$

$$\hat{\mathbf{P}}_{k|k-1} = \mathbf{A}_{k-1}\hat{\mathbf{P}}_{k-1|k-1}\mathbf{A}_{k-1}^T + \mathbf{Q}_{k-1} \tag{4.85}$$

$$\mathbf{K}_k = \hat{\mathbf{P}}_{k|k-1}\mathbf{C}_k^T(\mathbf{C}_k\hat{\mathbf{P}}_{k|k-1}\mathbf{C}_k^T + \mathbf{R}_k)^{-1} \tag{4.86}$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{y}_k - \hat{\mathbf{y}}_k) \tag{4.87}$$

$$\hat{\mathbf{P}}_{k|k-1} = (\mathbf{I} - \mathbf{K}_k\mathbf{C}_k)\hat{\mathbf{P}}_{k|k-1} \tag{4.88}$$

The paradigm is assumed to have a discrete time system, but in real systems the general the state equation is continuous, while the output one is discrete.
This conditions, that is called an hybrid system, corresponds to

$$\begin{cases} \dot{\mathbf{x}}_t = \mathbf{f}_c(\mathbf{x}_t, \mathbf{u}_t) + \mathbf{w}_t, \quad \mathbf{w}_t \sim \mathcal{N}(0, \mathbf{Q}_t) \\ \mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{v}_k, \quad \mathbf{v}_k \sim \mathcal{N}(0, \mathbf{R}_k) \end{cases} \tag{4.89}$$

Since the EKF attitude estimation can be solved only in discrete time, there are two different strategies:

- Apply Kalman filter equations with an hybrid implementation, that propagates the differential equation for predict part, while using discrete equation for correction part. The issue with this implementation are the difficulties in resolving differential non linear equations.

- First discretize the dynamical relation, obtaining a full discrete system, and then apply EKF. The issue with this method is that the continuous additive noise may be not be additive when discretized, and in this way we will need the non-additive noise formulation. It is possible then to suppose to have the continuous system without noise, and add it after the discretization, in order to ensure the additivity of it.

Forward Euler integration EKF

A special case of discretization is the one when forward euler (FE) integration is applied to the original system, obtaining a discretized one that is

$$\begin{cases} \mathbf{x}_{k+1} & = \mathbf{x}_k + \mathrm{T}\mathbf{f}_c(\mathbf{x}_k, \mathbf{u}_k) + \mathrm{T}\mathbf{w}_k, \quad \mathbf{w}_k \sim \mathcal{N}(0, \mathbf{Q}_k) \\ \mathbf{y}_k & = \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{v}_k, \quad \mathbf{v}_k \sim \mathcal{N}(0, \mathbf{R}_k) \end{cases} \tag{4.90}$$

With this method the noise remains additive even in discrete time, with $\mathrm{T}\mathbf{w}_k \sim \mathcal{N}(0, \mathrm{T}^2\mathbf{Q})$ and then original formulation of EKF can be applied, obtaining

$$E[\mathbf{x}_0] = \boldsymbol{\mu}_0, \quad \mathrm{Var}[\mathbf{x}_0] = \mathbf{P}_0 \tag{4.91}$$

$$\hat{\mathbf{x}}_{k|k-1} = \hat{\mathbf{x}}_{k-1|k-1} + \mathrm{T}\mathbf{f}_c(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}) \tag{4.92}$$

$$\hat{\mathbf{y}}_k = \mathbf{h}(\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k) \tag{4.93}$$

$$\mathbf{A}_{k-1} = \mathbf{I} + \frac{\partial \mathbf{f}_c(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1})}{\partial \mathbf{x}}\mathrm{T}, \quad \mathbf{C}_k = \frac{\partial \mathbf{h}(\hat{\mathbf{x}}_{k|k-1}, \mathbf{u}_k)}{\partial \mathbf{x}} \tag{4.94}$$

$$\hat{\mathbf{P}}_{k|k-1} = \mathbf{A}_{k-1}\hat{\mathbf{P}}_{k-1|k-1}\mathbf{A}_{k-1}^T + \mathrm{T}^2\mathbf{Q}_{k-1} \tag{4.95}$$

$$\mathbf{K}_k = \hat{\mathbf{P}}_{k|k-1}\mathbf{C}_k^T(\mathbf{C}_k\hat{\mathbf{P}}_{k|k-1}\mathbf{C}_k^T + \mathbf{R}_k)^{-1} \tag{4.96}$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{y}_k - \hat{\mathbf{y}}_k) \tag{4.97}$$

$$\hat{\mathbf{P}}_{k|k-1} = (\mathbf{I} - \mathbf{K}_k\mathbf{C}_k)\hat{\mathbf{P}}_{k|k-1} \tag{4.98}$$

Since the linearized matrix $\mathbf{A}$ can also be obtained by exponential series expansion, if a better representation is needed higher order terms can be included, with an increment of computational cost.
A different numerical integration approach make the noise to become non-additive in discrete time. In this case EKF can still be applied, but with a linearization about the noise, in order to understand its equivalent linear covariance. In that case the formulation is known as non-additive noise EKF.

Chapter 5

Simulation Results

The aim of this chapter is to perform attitude estimation with different methods, using information from IMU sensors. In particular:

- Attitude estimation is performed by gyroscope and accelerometer singularly, by using their characterization obtained in Ch.(3).
- Gyroscope and accelerometer outputs are combined with the Extended Kalman Filter, building a sensor fusion estimation method.
- The sensor fusion method is applied under a simulated noisy version of IMU measures.

5.1 Simulation Setup

Simulation of a riding motorcycle are performed by Vi-Bike Simulator, a tool that is able to simulate the motion of a motorbike on a circuit. In this case of interest, it is able to generate references for the angles, and to read data from an IMU mounted on a certain position of the vehicle.
The simulation that will be used are:

- Vi-Track: this corresponds to a lap on the track in Fig.(5.1). All data is sampled at $T = 0.01s$.
- Slalom: this corresponds to a slalom condition, sampled at $T = 0.001s$.

In both datasets, the IMU mounting position is assumed to be the center of mass, and its measures are noiseless.
Regarding to conventions, the simulation returns the references for angles as a ZXY sequence, while the Yaw-Pitch-Roll convention corresponds to ZYX. For this reason, the references in ZYX convention are obtained by the ones in ZXY convention by Eq.(2.16)-(2.17)-(2.18), where the rotation matrix is $\mathbf{R}_{zxy} = \mathbf{R}_y\mathbf{R}_x\mathbf{R}_z$.
Regarding to noisy data, starting from ideal IMU measurements obtained by simulation, it is possible to insert additive noise a posteriori, as explained in Ch.(3). In this case, noise intensity is modeled as the one of the Xsens Mtx, that is a MEMS IMU containing three gyroscope, accelerometers and magnetometer. From its datasheet and from [13], its noise variance components are found. In Sec.(3) the noise value are described in detail.

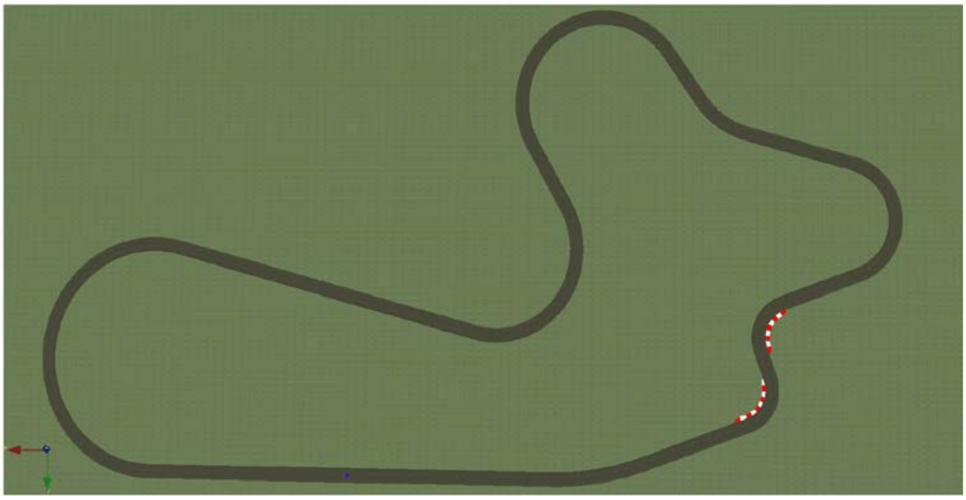


Figure 5.1: Vi-Track

5.2 Single IMU attitude estimation

5.2.1 Gyroscope attitude estimation

With the access to discrete time measurements of angular velocities it is possible to recursively estimate the body angles, once that initial conditions are set.
Regarding to this type of estimates, that are called dead reckoning, there are several issues, related to simplicity and lacking of information, that are:

- This attitude estimation method supposes to have perfect angular velocities values, but since in real scenario they coincide with gyroscope readings, they present different kind of errors. In particular, the two main contribution are:
  - Additive random noise: each measured value presents a random additive high frequency noise, which value depends on the type of gyroscope.
  - Bias instability: gyroscope measures can present some constant or slowly variant bias with respect to the real value.
- In general the initial condition $[\phi_0, \theta_0, \psi_0]^T$ is not known, which is fundamental for a good estimate. With a non perfect initial condition, the system evolves in open loop and cannot converge to real values, even if the gyroscope measures are not affected by any error.
- Without using any correction approach it is not possible to guarantee convergence of this method. This open-loop method is known as Dead Reckoning.
- Depending on the sampling time and bandwidth of the measured signal, the first order derivative approximation could not be enough correct, and so different numerical integration can be applied, like Runge-Kutta.

Different results of dead reckoning estimates are shown in Fig.(5.2)-(5.3)-(5.4). In particular, estimates are computed with *Vi-Track* dataset, and noise is generated as explained in Ch.(3).

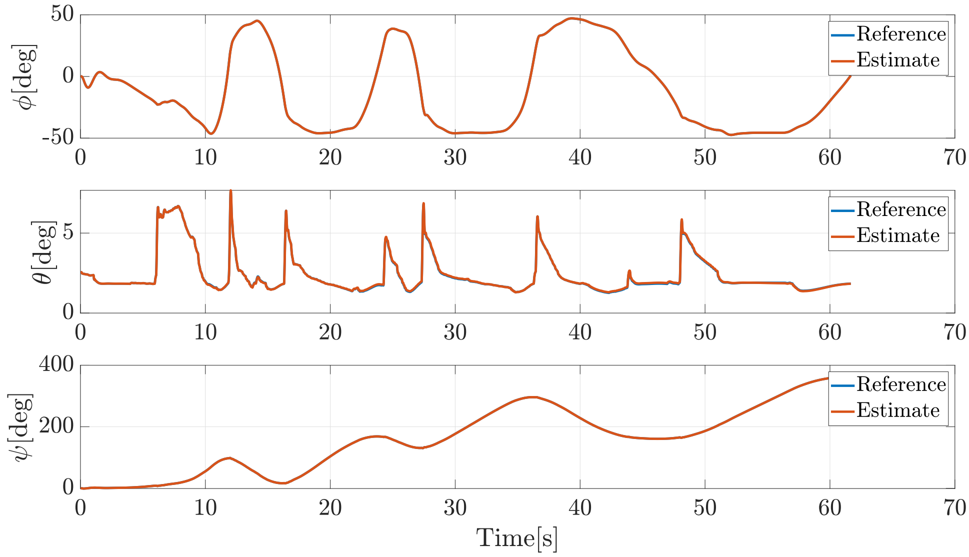


Figure 5.2: Dead reckoning method estimates with noiseless gyroscope measurements and correct initial condition

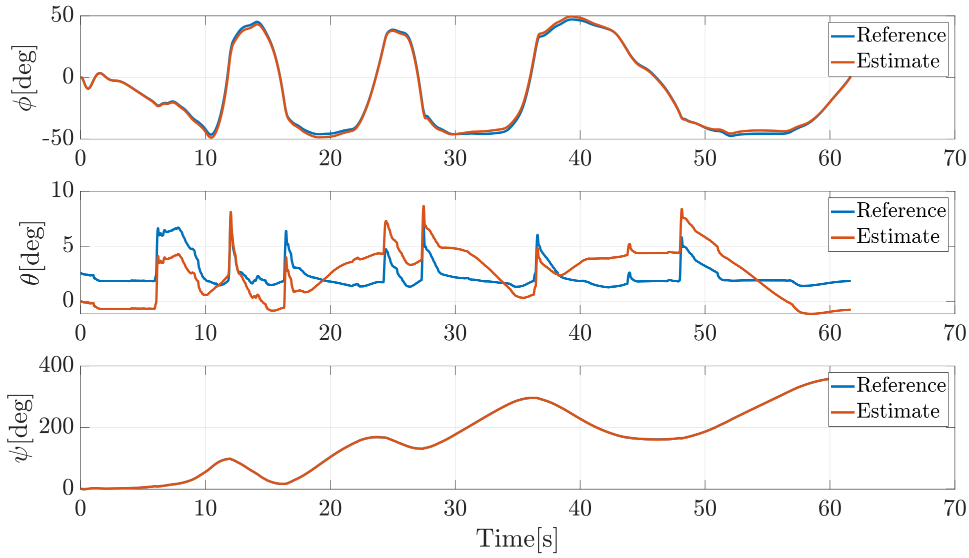


Figure 5.3: Dead reckoning method estimates with noiseless gyroscope measurements and wrong initial condition

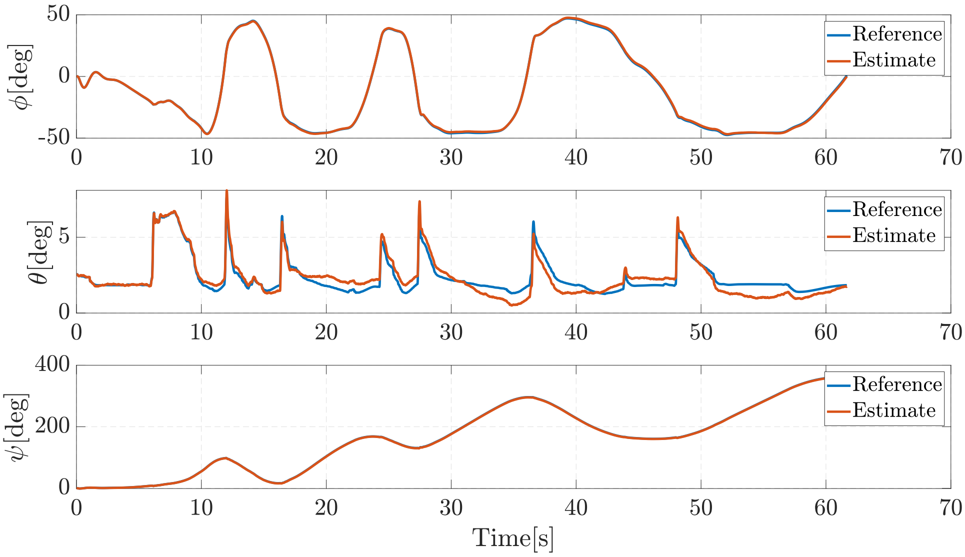


Figure 5.4: Dead reckoning method estimates with noisy gyroscope measurements and correct initial condition

From previous figures we can notice that

- Fig.(5.2) consists of dead reckoning estimate when initial conditions are correct and when gyroscope readings are ideal. This corresponds to the ideal case, where the only possible errors are due to discretization of non-linear system. In this case we have almost perfect estimates, and this means that the approach of first order derivative approximation is enough precise for this problem. Since the first order approximation depends on the sampling time, and in *Vi-Track* $T = 0.01s$, a sampling frequency of $f = 100Hz$ is great enough to make this approximation to be precise.

- Fig.(5.3) shows the error due to wrong initial condition, while the gyroscope readings are still ideal. In this case the wrong initial condition are not compensated, since the equations are evolving in open loop, and the system is not converging.

- Fig.(5.4) shows the error due to constant noises, even if we have correct initial condition. With the addition of noise components, the estimates does not necessarily converge because Eq.(3.5) consists of a discrete time integration.

5.2.2 Accelerometer attitude estimation

The attitude estimation from accelerometer is computed as in Eq.(3.8)-(3.9).
Starting from these equations, it is possible to understand which velocities better describe the acceleration by performing several tests having access to different velocities combinations.

Accelerometer attitude estimation under uniform motion assumption

In many cases, like [2],[3],[15], it is assumed to measure accelerometer data when external acceleration are not present or are negligible. This is a strong assumption, because the vehicle has to move mainly on uniform motion.
With these assumptions we have that

$$\mathbf{a} \simeq \mathbf{R}\mathbf{g} \tag{5.1}$$

and an estimate for roll and pitch angles can be directly found only by accelerometer readings, obtaining

$$\hat{\theta} = \text{asin}\Big[\frac{-a_x}{g}\Big], \quad \ddot{p}_x \simeq 0 \tag{5.2}$$

$$\hat{\phi} = \text{atan}\Big[\frac{a_y}{a_z}\Big], \quad \ddot{p}_y \simeq 0, \ddot{p}_z \simeq 0 \tag{5.3}$$

First, differences between the measured acceleration and the gravity projection are evaluated in order to understand if the assumption of negligible external forces can be made.
Fig.(5.5)-(5.6) show acceleration comparisons and estimates under assumptions of negligible external forces.

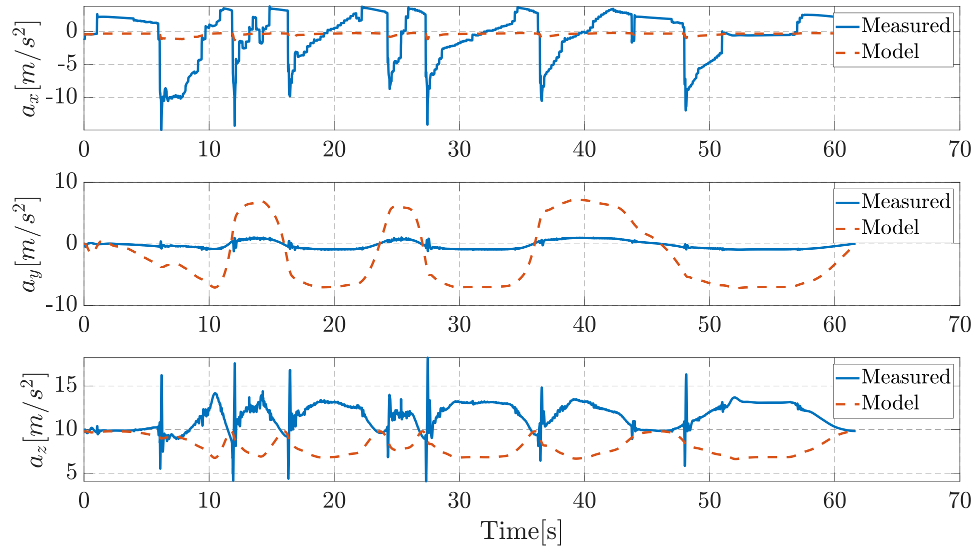


Figure 5.5: Acceleration model with null external forces

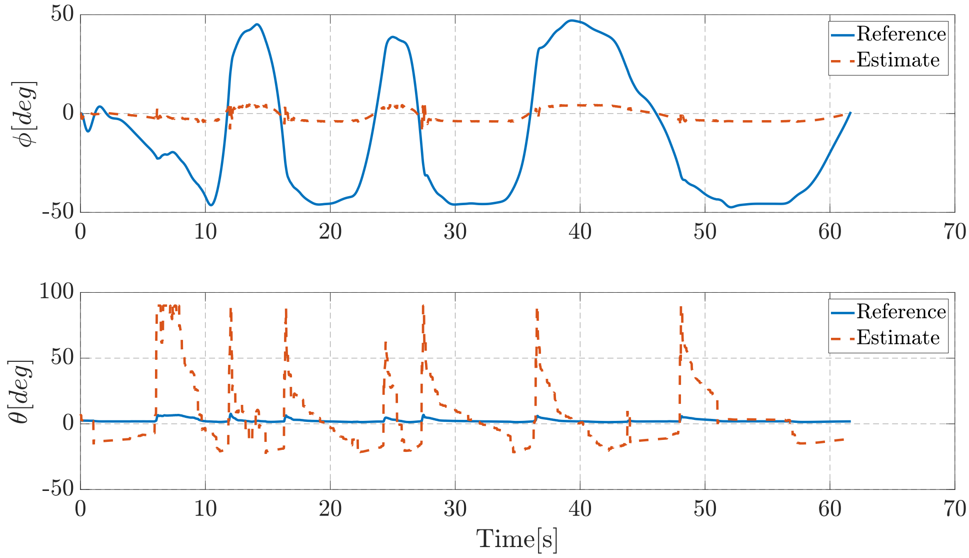


Figure 5.6: Accelerometer attitude estimation with null external forces

These figures clearly show that the gravity term is not prevalent with respect to external forces, in fact the characterization is completely different.
It is necessary to insert some knowledge about the external forces applied to the vehicle.

Accelerometer attitude estimation with complete model

The estimates can be computed through the complete model of Eq.(3.8)-(3.9), which is based on the assumption of having access to all velocities.
Since the axis accelerations are necessary, namely $\dot{v}$, but they are not measured, it is possible to compute them by approximating the derivative with the first order difference of the velocity, that is

$$\dot{v}(k) \simeq \frac{v(k) - v(k-1)}{T} \tag{5.4}$$

Fig.(5.7)-(5.8) show the results of complete acceleration characterization and related estimates.

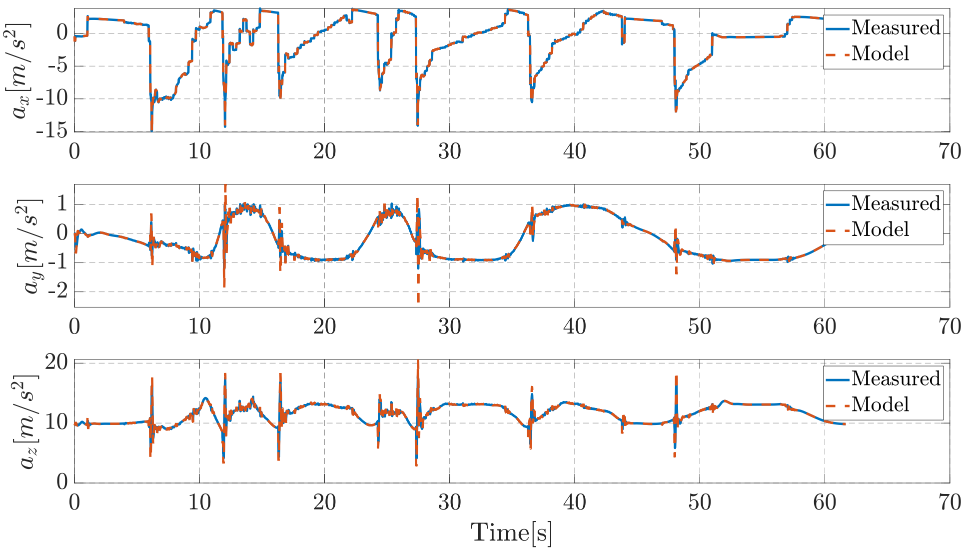


Figure 5.7: Acceleration model with complete information

From the comparison between measured and modeled acceleration, the knowledge of all velocities is able to describe the accelerometer readings, and this reflects into good estimates of roll and pitch angles.
The main errors are made of peaks in pitch angle estimates, they are generated by two assumptions that are:

- IMU on the center of mass: the dynamic motion of a motorcycle depends also on chassis and suspension. With the motion of the suspensions, the center of mass of the vehicle changes, and this translates into a different reading from accelerometer.
- First order velocity derivative: the approximation of the derivative with the first order difference can produce some errors when the velocity rapidly varies, when the sampling frequency is not high enough.

Previous results are obtained knowing velocities about all axis, which can be a strong assumption in a real scenario.
In this way, different cases will be considered, that are:

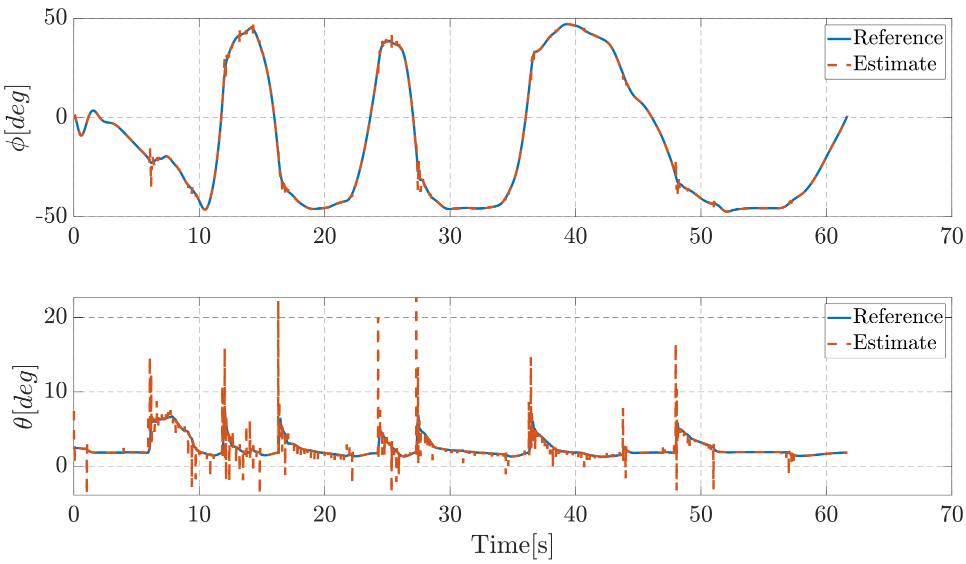


Figure 5.8: Accelerometer attitude estimation from complete information

- Only $\mathbf{v}_x$: this scenario is the most similar to the real one, since longitudinal velocity can be estimated through GPS sensors.
- A combination of $\mathbf{v}_x$ and $\mathbf{v}_x, \mathbf{v}_y$ in order to understand the main contribution for a good characterization.

For each case the acceleration expression and the estimate of the angles are evaluated.

Accelerometer attitude estimation with access only to $\mathbf{v}_x$

The characterization of accelerometer with measure of $\mathbf{v}_x$ is

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} \dot{v}_x \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ +\omega_z v_x \\ -\omega_y v_x \end{bmatrix} + \begin{bmatrix} -\sin(\theta)g \\ \sin(\phi)\cos(\theta)g \\ \cos(\phi)\cos(\theta)g \end{bmatrix} \tag{5.5}$$

while estimates are equal to

$$\hat{\theta} = \text{asin}\left(\frac{\dot{v}_x - a_x}{g}\right) \tag{5.6}$$

$$\hat{\phi} = \text{atan}\left(\frac{a_y - \omega_z v_x}{a_z + \omega_y v_x}\right) \tag{5.7}$$

The comparison of measured and modeled acceleration are shown in Fig.(5.9) and the estimates are shown in Fig.(5.10).

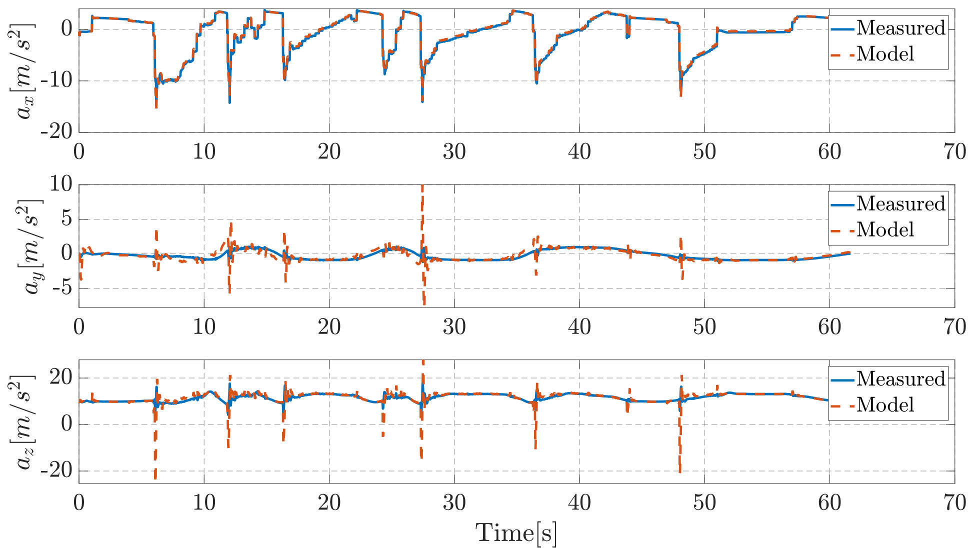


Figure 5.9: Acceleration model with access to $\mathbf{v}_x$

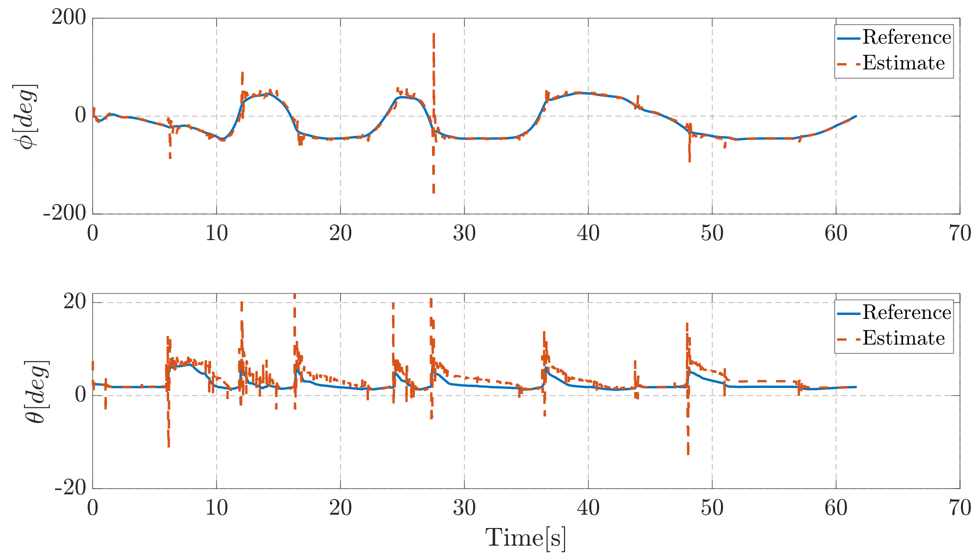


Figure 5.10: Accelerometer attitude estimation with knowledge of $\mathbf{v}_x$

From previous figures we can notice that the knowledge of v_x is fundamental in order to make a first characterization of the accelerations. With respect to the case with no access to velocity where the accelerations are totally wrong, in this case they present errors but they are shaped in a better way.
In particular, the longitudinal acceleration is shaped correct, while lateral and vertical acceleration present some errors, especially with peaks.
Body longitudinal velocity is the main variable that has to be taken into account to have a good estimates since it is directly related to effective velocity of the vehicle.

Accelerometer attitude estimation with access to $\mathbf{v}_x$ and $\mathbf{v}_y$

In the case of knowing $\mathbf{v}_x, \mathbf{v}_y$ the acceleration expression results

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} \dot{v}_x \\ \dot{v}_y \\ 0 \end{bmatrix} + \begin{bmatrix} -\omega_z v_y \\ +\omega_z v_x \\ -\omega_y v_x + \omega_x v_y \end{bmatrix} + \begin{bmatrix} -\sin(\theta)g \\ \sin(\phi)\cos(\theta)g \\ \cos(\phi)\cos(\theta)g \end{bmatrix} \tag{5.8}$$

while estimates are equal to

$$\hat{\theta} = \text{asin}\left(\frac{\dot{v}_x - \omega_z v_y - a_x}{g}\right) \tag{5.9}$$

$$\hat{\phi} = \text{atan}\left(\frac{a_y - \dot{v}_y - \omega_z v_x}{a_z + \omega_y v_x - \omega_x v_y}\right) \tag{5.10}$$

The comparison of measured and modeled acceleration is shown in Fig.(5.11)-(5.12) In this case, with respect to previous one, there is an improvement in lateral acceleration, but there are some errors during peaks phenomena.
It is important to notice that longitudinal acceleration characterization seems almost exact, and then this should be reflects into a correct pitch estimate, since it depends only on $\mathbf{a}_x$. From this figure pitch estimates is full of peaks, and this means that even a little error in longitudinal acceleration leads to errors in estimates, due to the asin function.

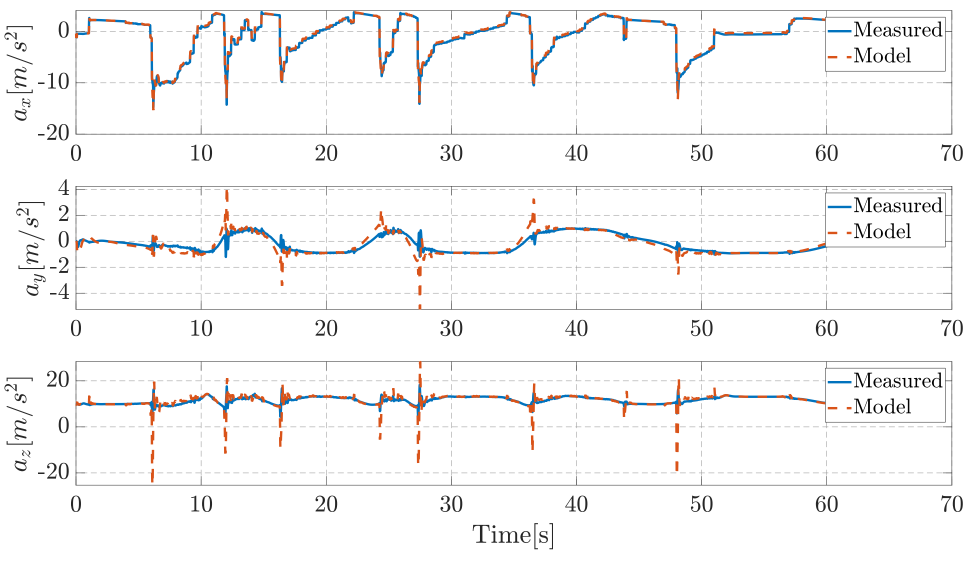


Figure 5.11: Acceleration model with complete information with access to $\mathbf{v}_x, \mathbf{v}_y$

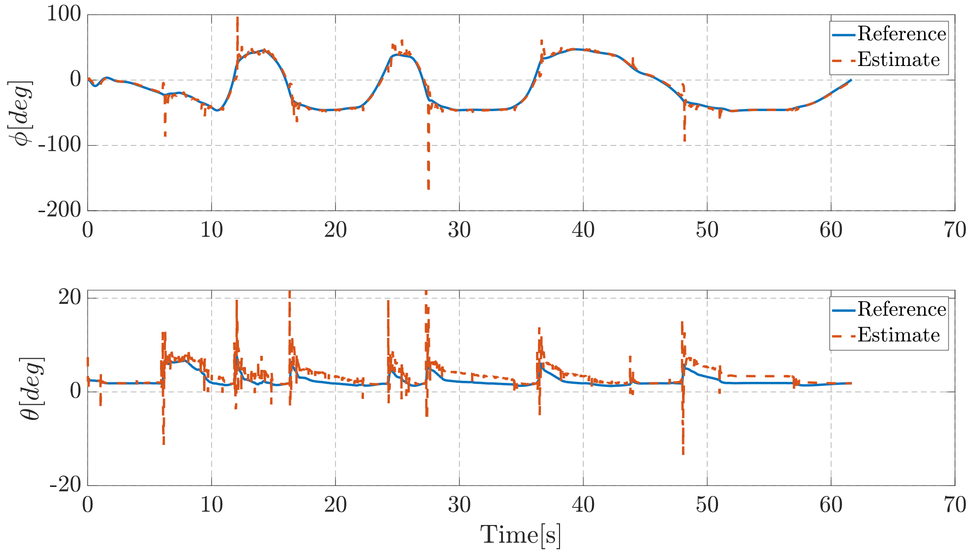


Figure 5.12: Accelerometer attitude estimation with access to $\mathbf{v}_x, \mathbf{v}_y$

Access to $\mathbf{v}_x$ and $\mathbf{v}_z$

In the case of knowing $\mathbf{v}_x, \mathbf{v}_z$ the acceleration expression

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} \dot{v}_x \\ 0 \\ \dot{v}_z \end{bmatrix} + \begin{bmatrix} +\omega_y v_z \\ +\omega_z v_x - \omega_x v_z \\ -\omega_y v_x \end{bmatrix} + \begin{bmatrix} -\sin(\theta) g \\ \sin(\phi)\cos(\theta) g \\ \cos(\phi)\cos(\theta) g \end{bmatrix} \tag{5.11}$$

while estimates are equal to

$$\hat{\theta} = \text{asin}\left(\frac{\dot{v}_x + \omega_y v_z - a_x}{g}\right), \quad \hat{\phi} = \text{atan}\left(\frac{a_y - \omega_z v_x + \omega_x v_z}{a_z - \dot{v}_z + \omega_y v_x}\right) \tag{5.12}$$

The comparison of measured and modeled acceleration is shown in Fig.(5.13)-(5.14)

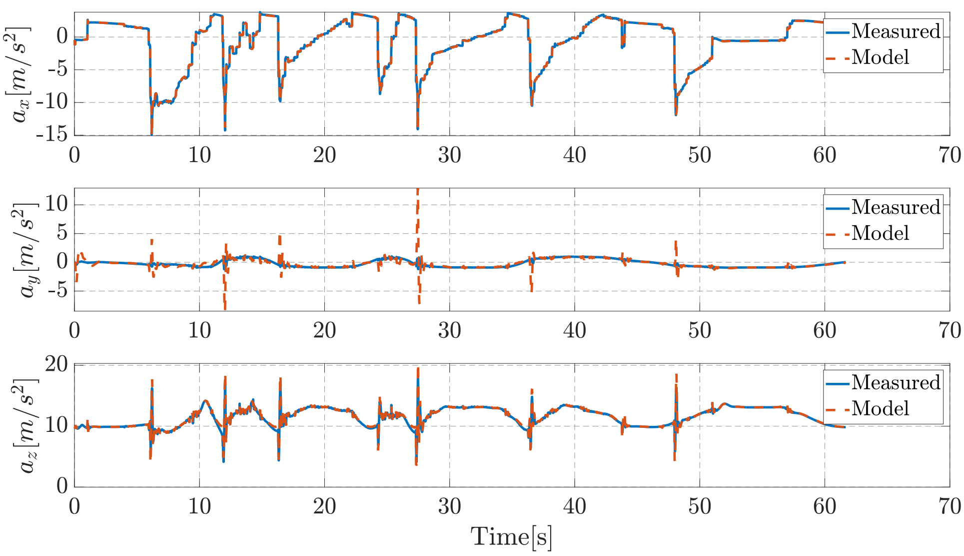


Figure 5.13: Acceleration model with complete information with access to $\mathbf{v}_x, \mathbf{v}_z$

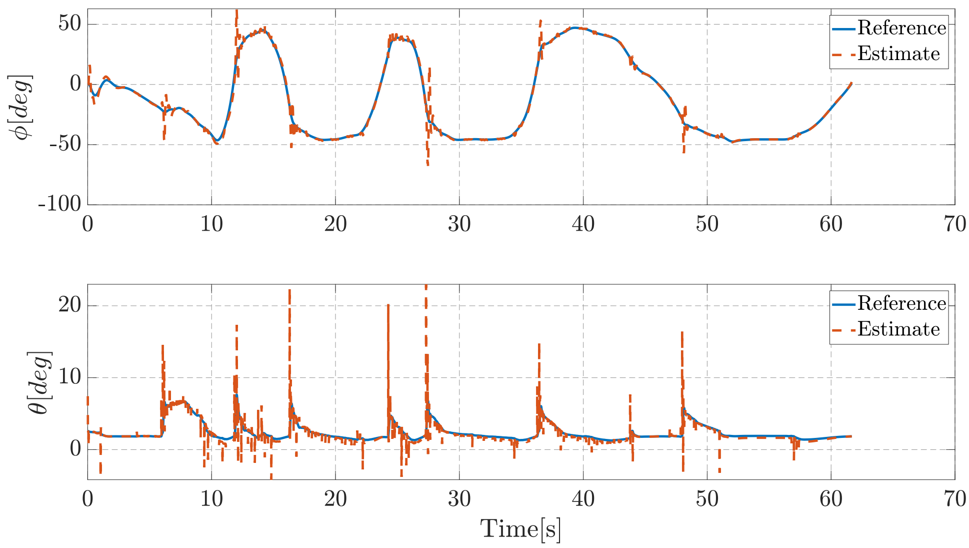


Figure 5.14: Accelerometer attitude estimation with knowledge of $\mathbf{v}_x, \mathbf{v}_z$

With the addition of vertical velocity with respect to the lateral one all peaks in $\mathbf{a}_z$ are shaped correctly, thanks to $\dot{\mathbf{v}}_z$ term.
Even if the vertical acceleration is shaped better with respect to the previous cases, there is a little improvement in estimates. This means that even if a large error in acceleration characterization is present it is not necessary related to error in estimates.

5.2.3 Only accelerometer estimates results

From the previous figures, one can state that:

- The assumption of having only gravity term is too strong, since the major contributions are centripetal and longitudinal acceleration which give totally different acceleration terms. Consequently, the estimates based on this assumptions are wrong, because they can be correct only when the motorcycle is running on uniform motion.

- In the case of having access to all velocities the acceleration is well characterized, but there are still some errors, because estimates are full of peaks. In the case of strong acceleration or in braking condition the approximation of considering numerical derivative of the velocity and the assumption of being in the center of mass produce some high frequency errors.

- The estimation equation is static, which means that it is not affected by motorcycle dynamics but only by instantaneous measurements. This can be a good property if the acceleration model is approximated, because an error on an estimate does not affect the next ones.

From the previous issues accelerometer attitude estimation can be reasonable under the knowledge of all body velocities and angular velocities, but it presents large error if some variables are not known or present uncertainties.

5.3 EKF with the original model

In this sectionr the EKF algorithm is tested in simulation dataset, as it has been done with attitude estimation from single IMU.
Fusion algorithm considers a system which has continuous time state, discrete time measurements and additive noise in discrete time.
Let's assume first to know all velocities, gyroscope outputs and accelerometer outputs. It is possible to formulate the problem as a nonlinear system, solvable via EKF.
The state equation is the relation between rates of angles and angular velocities, for which we assume angular velocities as inputs, while noise is added in discrete time.
Defining $\boldsymbol{\zeta} = [\phi, \theta, \psi]^T$, the continuous time state equation is

$$\dot{\boldsymbol{\zeta}} = f_c(\boldsymbol{\zeta}, \boldsymbol{\omega}) + \mathbf{w} = \begin{bmatrix} \omega_x + \sin(\phi)\tan(\theta)\omega_y + \cos(\phi)\tan(\theta)\omega_z \\ \cos(\phi)\omega_y - \sin(\theta)\omega_z \\ \sin(\phi)/\cos(\theta)\omega_y + \cos(\phi)/\cos(\theta)\omega_z \end{bmatrix} \tag{5.13}$$

The measurement equation, which is the one obtained by the characterization of the acceleration with additive white gaussian noise is equal to

$$\mathbf{a} = \mathbf{h}(\boldsymbol{\zeta}, \boldsymbol{\omega}, \mathbf{v}, \dot{\mathbf{v}}) + \mathbf{n} = \begin{bmatrix} \dot{v}_x + \omega_z v_y + \omega_y v_z - \sin(\theta)g \\ \dot{v}_y + \omega_z v_x - \omega_x v_z + \cos(\theta)\sin(\phi)g \\ \dot{v}_z - \omega_y v_x + \omega_x v_y + \cos(\theta)\cos(\phi)g \end{bmatrix} + \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}, \quad \mathbf{v} \in \mathcal{N}(\mathbf{0}, \mathbf{R}) \tag{5.14}$$

Writing the system in state space representation, we have

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{f}_c(\mathbf{x}, \mathbf{u}) \\ \mathbf{y} = \mathbf{h}(\mathbf{x}, \mathbf{u}) + \mathbf{v}, \quad \mathbf{v} \in \mathcal{N}(\mathbf{0}, \mathbf{R}) \end{cases} \tag{5.15}$$

where

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \\ u_9 \end{bmatrix} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \\ v_x \\ v_y \\ v_z \\ \dot{v}_x \\ \dot{v}_y \\ \dot{v}_z \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} \tag{5.16}$$

and

$$\mathbf{f}_c(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} u_1 + \sin(x_1)\tan(x_2)u_2 + \cos(x_1)\tan(x_2)u_3 \\ \cos(x_1)u_2 - \sin(x_2)u_3 \\ \sin(x_1)/\cos(x_2)u_2 + \cos(x_1)/\cos(x_2)u_3 \end{bmatrix} \tag{5.17}$$

$$\mathbf{h}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} +u_7 - u_3u_5 + u_2u_6 - \sin(x_2)g \\ +u_8 + u_3u_4 - u_1u_6 + \cos(x_2)\sin(x_1)g \\ +u_9 - u_2u_4 + u_1u_5 + \cos(x_2)\cos(x_1)g \end{bmatrix} \tag{5.18}$$

Since EKF algorithm first discretize and then linearize, applying Forward Euler the discretized system is equal to

$$\begin{cases} \mathbf{x}_{k+1} = \mathbf{x}_k + \mathrm{T}\mathbf{f}_c(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{w}_k, \quad \mathbf{w} \in \mathcal{N}(\mathbf{0}, \mathbf{Q}) \\ \mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{v}_k, \quad \mathbf{v} \in \mathcal{N}(\mathbf{0}, \mathbf{R}) \end{cases} \tag{5.19}$$

where the process noise has been added after discretization.
The discretized system, where EKF is applied to is

$$
\begin{cases}
\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} x_1 + \mathrm{T}(u_1 + \sin(x_1)\tan(x_2)u_2 + \cos(x_1)\tan(x_2)u_3) \\ x_2 + \mathrm{T}(\cos(x_1)u_2 - \sin(x_2)u_3) \\ x_3 + \mathrm{T}(\sin(x_1)/\cos(x_2)u_2 + \cos(x_1)/\cos(x_2)u_3) \end{bmatrix} + \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \\
\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} +u_7 - u_3u_5 + u_2u_6 - \sin(x_2)g \\ +u_8 + u_3u_4 - u_1u_6 + \cos(x_2)\sin(x_1)g \\ +u_9 - u_2u_4 + u_1u_5 + \cos(x_2)\cos(x_1)g \end{bmatrix} + \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}
\end{cases}
\tag{5.20}
$$

Before applying EKF it is important to understand how IMU measures are used:

- The system is adapted for sensor fusion since it includes gyroscope and accelerometer measures: in particular, gyroscope readings are present both in state and output equations, while accelerometer readings are present only in output equations.
- Gyroscope angular velocities are treated as inputs, and then they have to be considered exact. Since in a real system angular velocities are noisy, their uncertainties and noise are described by the additive noise $\mathbf{w}_k$. In this way, errors on gyroscopes are considered as model errors.
- Accelerometers are treated as outputs, which is correct since they are measured from IMU sensor. Since angular velocities are present also in accelerometer equations, then the additive noise of output, that is $\mathbf{v}_k$ has to include both accelerometer noise and gyroscope one.
- The differential equation of the system does not consider any dynamical relation about the vehicle, indeed it is independent from the rigid body this method is applied to.

The tuning variables of EKF are covariance matrix of model and output errors, that are called $\mathbf{Q}$ and $\mathbf{R}$, and initial state covariance $\mathbf{P}_0$. In [10], an optimization has been performed in order to capture the best weighting matrices to be chosen. In this case, covariance matrices values are taken similar to the one that has been found in [10] and then they will be tuned it for the specific cases.
Covariance matrices will be set as diagonal with starting values

$$
\mathbf{Q} = q\mathbf{I}, \quad q = 10^{-7}, \quad \mathbf{R} = r\mathbf{I}, \quad r = 10^2 \tag{5.21}
$$

It is important to notice that state and output covariance matrices has a difference of 10^9, which means that state equations are assumed to be almost perfect, while acceleration measures are considered as high noisy.
This choice of weights is correct since the state equation is correct, while the acceleration one is approximated.
Another important parameter to be chosen is the initial state covariance $\mathbf{P}_0$. This matrix includes the uncertainty about the initial condition, and weight how much the algorithm is based on the initial state. In the statistical version of the Kalman filter equation, this matrix is equal to

$$
\mathbf{P}_0 = \mathrm{E}[(\mathbf{x}_0 - \boldsymbol{\mu}_0)(\mathbf{x}_0 - \boldsymbol{\mu}_0)^T] \tag{5.22}
$$

where $\mathbf{x}_0$ is the true unknown initial state and $\boldsymbol{\mu}_0$ is the initial guess.
Since in real scenario there is little knowledge about initial state the high initial state covariance is chosen as

$$\mathbf{P}_0 = p\mathbf{I}, \quad p = 10^2 \tag{5.23}$$

Even if the initial covariance is high, it is updated by the EKF algorithm and decreases if the state is converging.
Regarding to initial state, since it is unknown, the algorithm will start from

$$\mathbf{x}_0 = [0, 0, 0]^T \tag{5.24}$$

Once that covariance matrices are set, which act as weight matrices, EKF is applied to *Slalom* and *Vi-Track* datsets.

5.3.1 EKF simulation results

As it has been done before in the case of accelerometer attitude estimation, estimates are computed assuming to have access only to some velocities, in order to understand how the algorithm works with lack of variables.
Since EKF works as a Kalman filter, is based on the two steps approach that are prediction and correction: prediction is based on the relation between rates of angles and angular velocities, while correction uses the predicted state and the output measure to adjust the state, through accelerometer readings.

Complete acceleration model

First it is considered the best case, that is when we read all velocities, namely $\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z$. In this case the acceleration are well described and the model is

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} x_1 + \mathrm{T}(u_1 + \sin(x_1)\tan(x_2)u_2 + \cos(x_1)\tan(x_2)u_3) \\ x_2 + \mathrm{T}(\cos(x_1)u_2 - \sin(x_2)u_3) \\ x_3 + \mathrm{T}(\sin(x_1)/\cos(x_2)u_2 + \cos(x_1)/\cos(x_2)u_3) \end{bmatrix} + \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \tag{5.25}$$

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} +u_7 - \sin(x_2)g \\ +u_3u_4 - +\cos(x_2)\sin(x_1)g \\ -u_2u_4 + +\cos(x_2)\cos(x_1)g \end{bmatrix} + \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \tag{5.26}$$

The associated estimates of the two datasets are shown in Fig.(5.15)-Fig.(5.16).

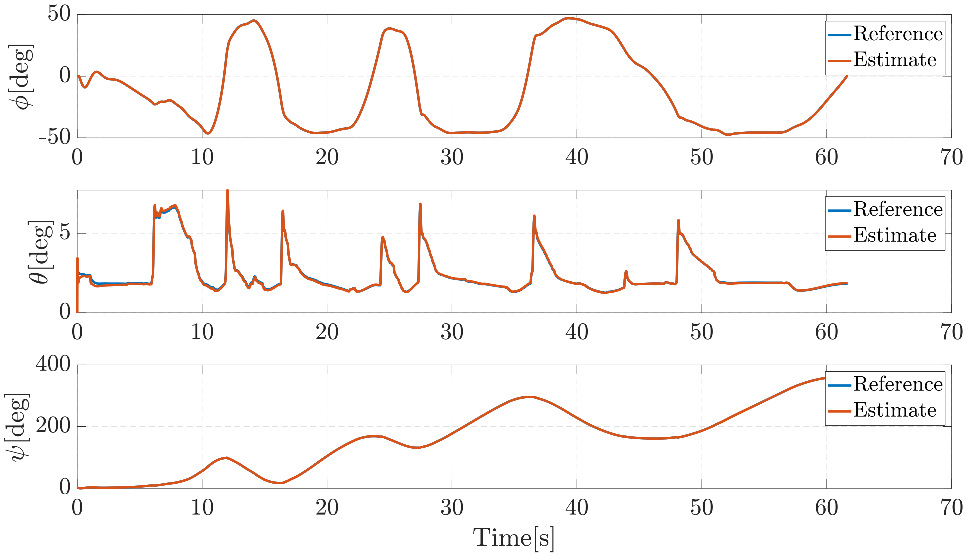


Figure 5.15: Vi-Track: EKF estimates with complete acceleration characterization

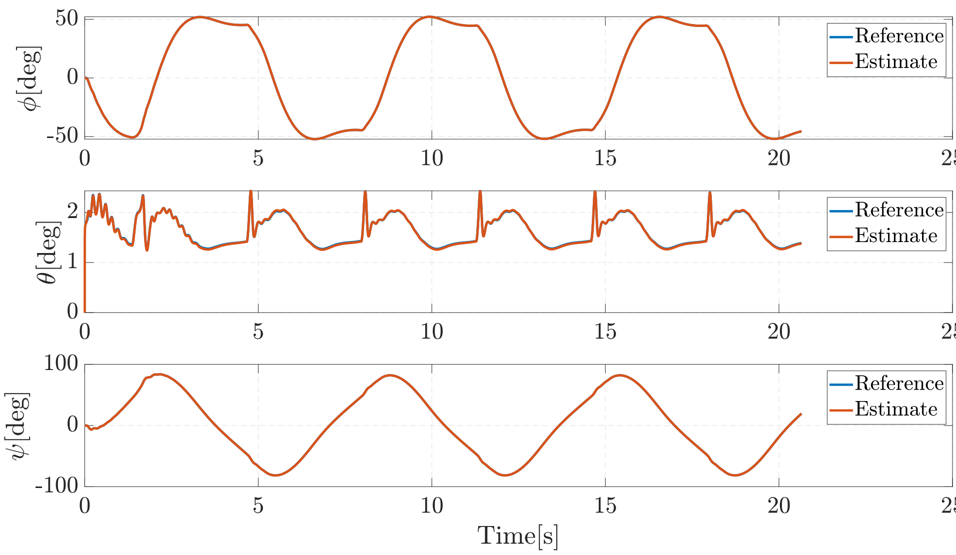


Figure 5.16: Slalom: EKF estimates with complete acceleration characterization

With the complete acceleration characterization the estimates are almost perfect with respect to the reference signal, even if the initial condition is not correct. This means that EKF is behaving as a good sensor fusion algorithm.
The estimate is also able to converge almost instantly to the true value, thanks to the initial state covariance matrix.
Since this is the ideal case, with access to all velocities, it is necessary understand how the algorithm behaves in the case of only $\mathbf{v}_x$ knowledge.

Reduced acceleration model: $\mathbf{v}_y = 0, \mathbf{v}_z = 0$

This case is the most important one, since the longitudinal velocity is the one we have access in a real scenario, and so it is fundamental to understand the behavior with this assumption.

The original model when $\mathbf{v}_x = 0, \mathbf{v}_y = 0$ becomes

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} x_1 + \mathrm{T}(u_1 + \sin(x_1)\tan(x_2)u_2 + \cos(x_1)\tan(x_2)u_3) \\ x_2 + \mathrm{T}(\cos(x_1)u_2 - \sin(x_2)u_3) \\ x_3 + \mathrm{T}(\sin(x_1)/\cos(x_2)u_2 + \cos(x_1)/\cos(x_2)u_3) \end{bmatrix} + \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \tag{5.27}$$

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} +u_7 - \sin(x_2)g \\ +u_3u_4 - +\cos(x_2)\sin(x_1)g \\ -u_2u_4 + +\cos(x_2)\cos(x_1)g \end{bmatrix} + \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \tag{5.28}$$

The associated estimates are shown in Fig.(5.17)-Fig.(5.18).

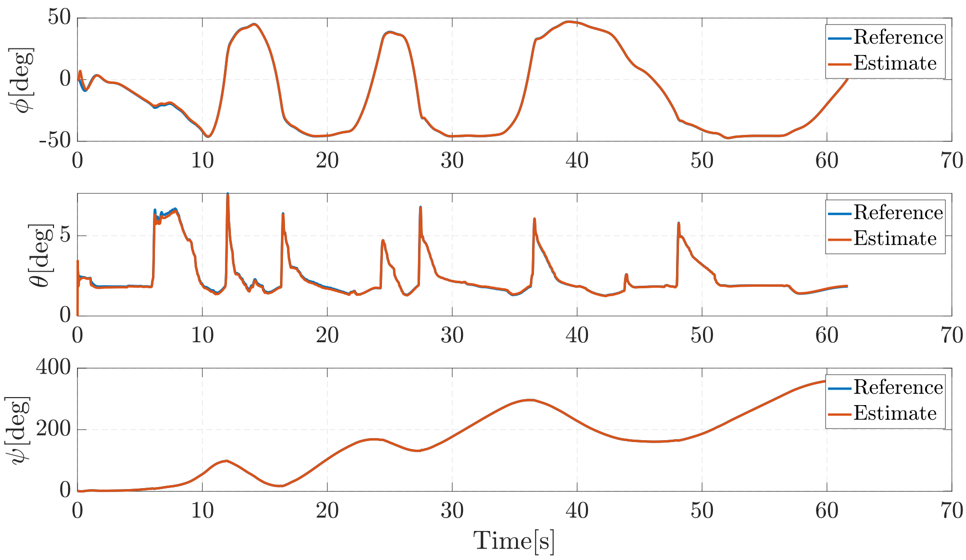


Figure 5.17: Vi-Track: EKF estimates with $\mathbf{v}_y = 0, \mathbf{v}_z = 0$

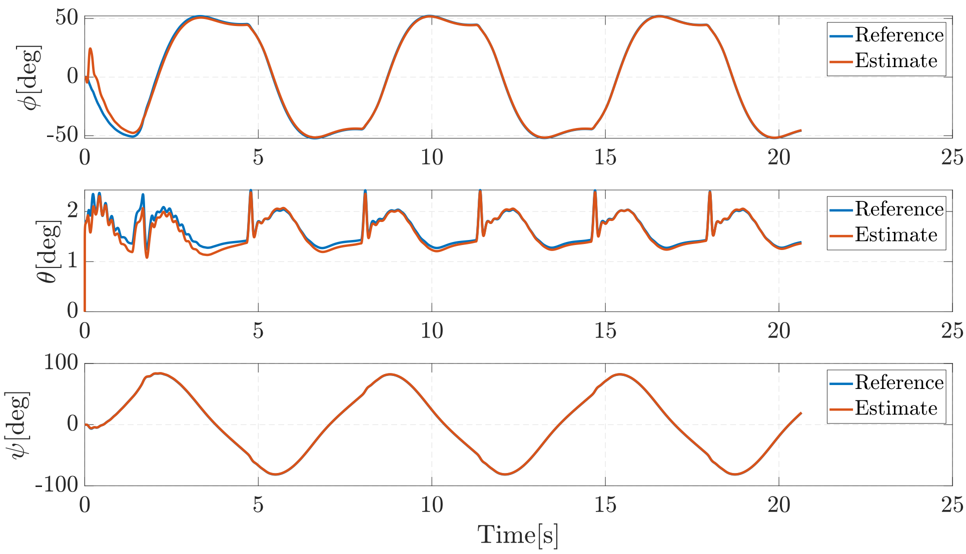


Figure 5.18: Vi-Track: EKF estimates with $\mathbf{v}_y = 0, \mathbf{v}_z = 0$

Even if we suppose $\mathbf{v}_y = 0, \mathbf{v}_z = 0$, the estimates are still good, which means that the main variable that characterize the accelerations in the longitudinal velocity, as we show in previous

chapter.
This is reasonable because the longitudinal velocity, which is solidal to body, is related to the effective velocity of the vehicle through the pitch angle. For small pitch angles, we have that $\mathbf{v}_x$ is almost equal to the effective velocity which generates the two main acceleration components, that are linear and centrifugal accelerations.
Linear and centrifugal acceleration can appear also on other axis in conditions like sideslip, but these phenomena are negligible with respect to the one generated by $\mathbf{v}_x$.
It is not necessary to show results in the case of a combination of $\mathbf{v}_x$ and other velocities, since the estimate is almost perfect yet.
Regarding to convergence velocity, a test is made by starting the algorithm from different initial conditions, up to the limit case for a motorcycle. The test is performed with the complete acceleration model.
In Fig.(5.19)-(5.20) the convergence of EKF is shown.

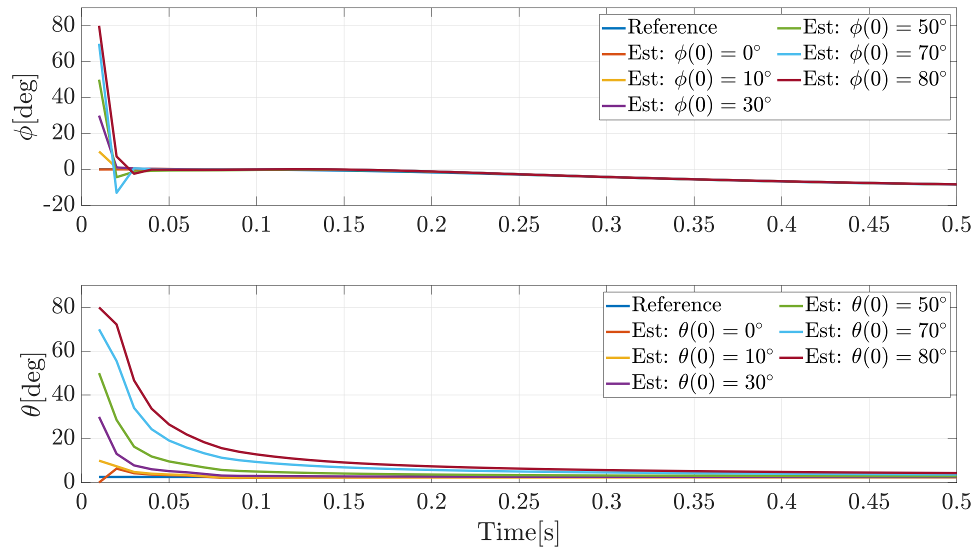


Figure 5.19: Vi-Track: convergence with different initial conditions

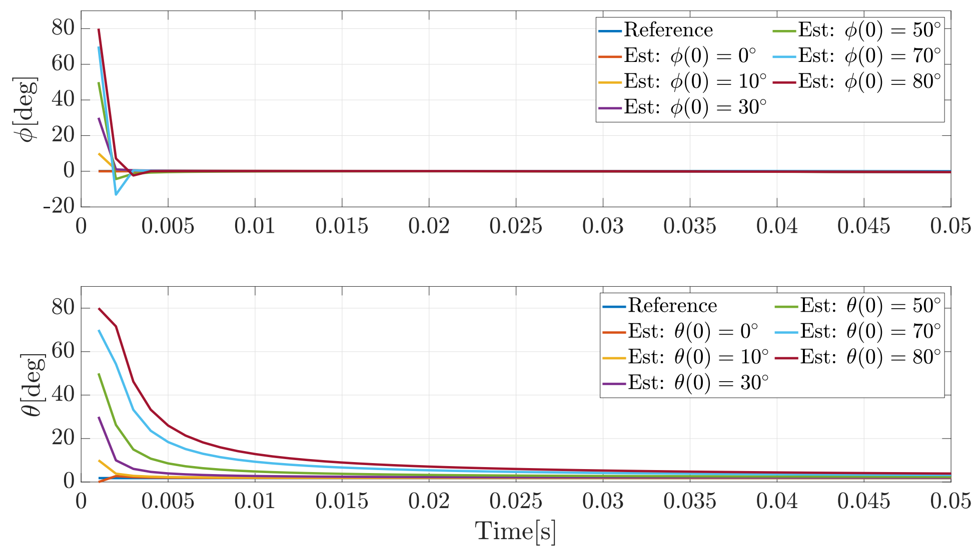


Figure 5.20: Slalom: convergence with different initial conditions

From previous figures it is visible that even in the case where the initial condition is wrong for the motorcycle case, like a roll angle of 80°, the algorithm is able to converge in about 50 steps. Since gyroscope and accelerometer measures are ideal in these datasets, the next step is to insert an additive noise to the sensors, in order to have a system that approximate a real one.

5.4 Extended Kalman Filter applied to Noisy IMU measurements

Since in the ideal case EKF estimate works well even with only $\mathbf{v}_x$ readings, its robustness is now tested, in particular under noisy measurements.

5.4.1 EKF results with noisy IMU measurements

From Algorithm(3) it is possible to generate noisy measurements, starting from ideal values.
Since it is necessary to understand the impact of noise in real IMU sensors, variance of errors will be taken from real sensors. In particular, the considered sensor is Xsens Mtx, that is a MEMS IMU containing three gyroscope, accelerometers and magnetometers. Magnetometers measures will not be considered.
From specifics of the components, the covariances that will be used to simulate noise are

$$\mathbf{Q} = q\mathbf{I}, \quad q = 11e - 4 \tag{5.29}$$
$$\mathbf{R} = r\mathbf{I}, \quad r = 4e - 2 \tag{5.30}$$

for gyroscopes and

$$\mathbf{Q} = q\mathbf{I}, \quad q = 2.7e - 6 \tag{5.31}$$
$$\mathbf{R} = r\mathbf{I}, \quad r = 2e - 2 \tag{5.32}$$

for accelerometers.
As what has been done for EKF estimates in previous section, two datasets are considered, that are *Vi-Track* and *Slalom*.

Vi-Track results

Comparisons of noisy and ideal gyroscope and accelerometers are shown respectively in Fig.(5.21)-(5.22).

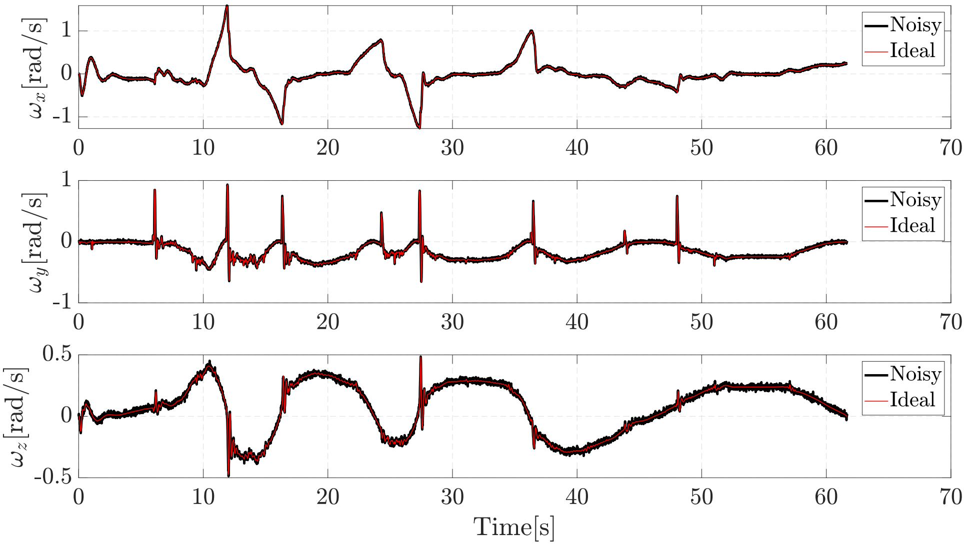


Figure 5.21: Vi-Track: Comparison between noisy and ideal gyroscope measures

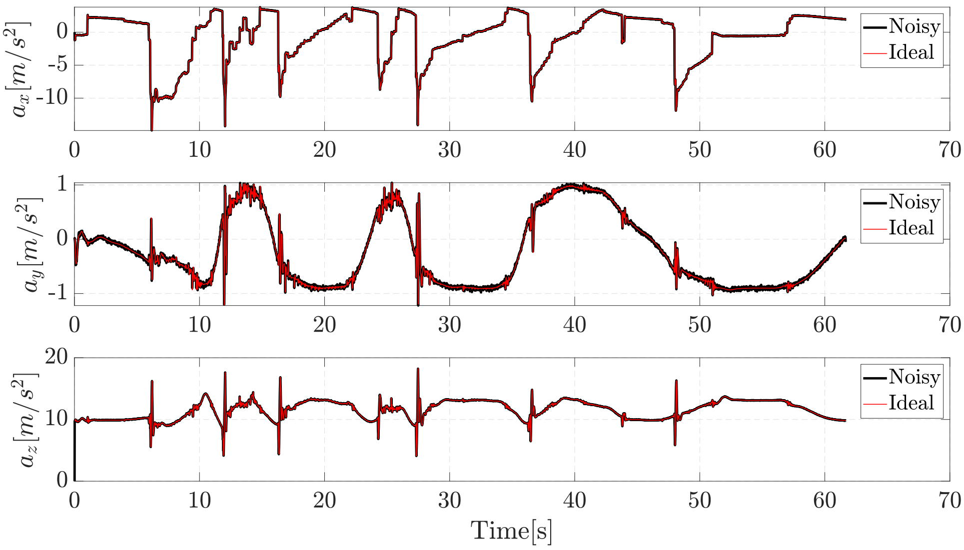


Figure 5.22: Vi-Track: Comparison between noisy and ideal accelerometer measures

Is is now possible to apply EKF with noisy gyroscope and accelerometer. Even in this case it is assumed to have access first to all velocities and after only to $\mathbf{v}_x$.
Extended Kalman Filter results with complete acceleration information and with knowledge of only $\mathbf{v}_x$ are shown in Fig.(5.23)-(5.24).

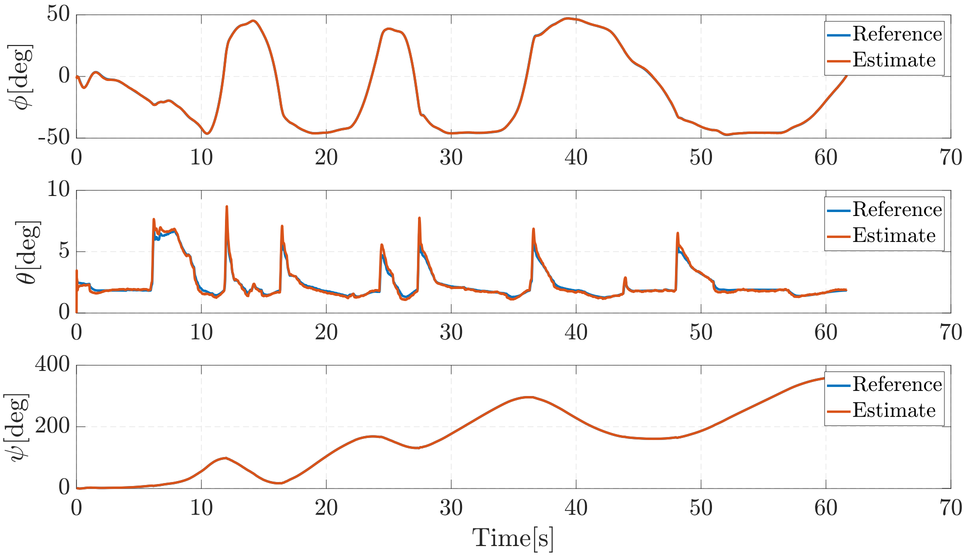


Figure 5.23: Vi-Track: EKF with complete acceleration information under noisy IMU measurements

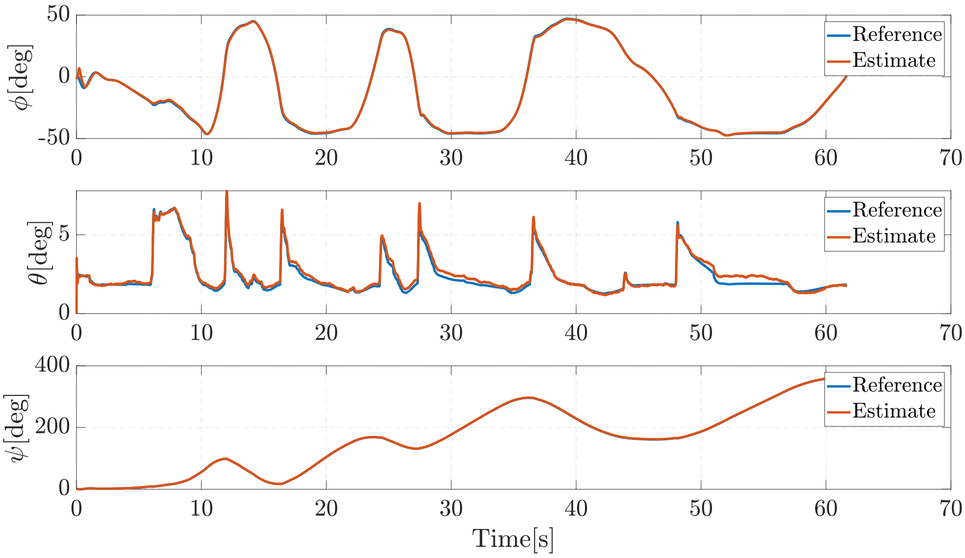


Figure 5.24: Vi-Track: EKF with only $\mathbf{v}_x$, under noisy IMU measurements

Slalom results

Comparisons of noisy and ideal gyroscope and accelerometers are shown respectively in Fig.(5.25)-(5.26).

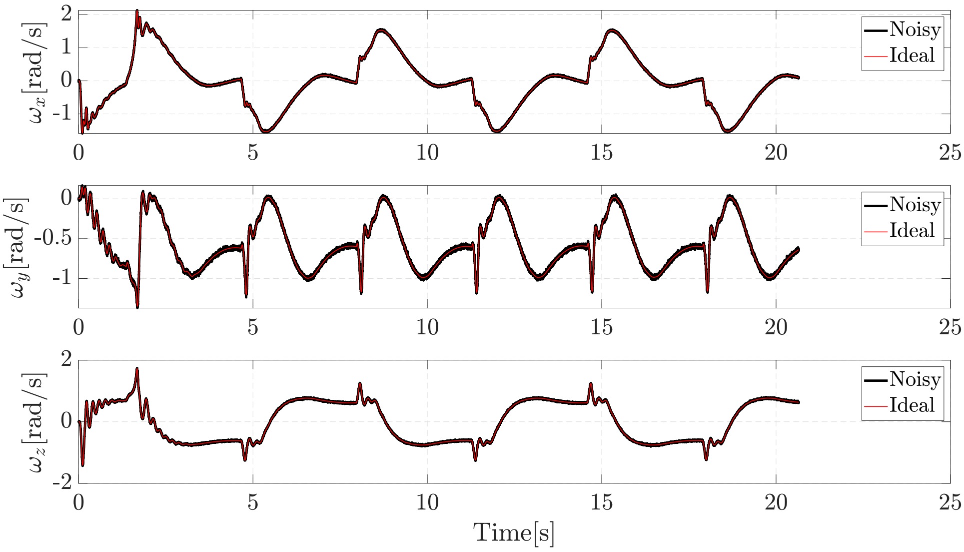


Figure 5.25: Slalom: Comparison between noisy and ideal gyroscope measures

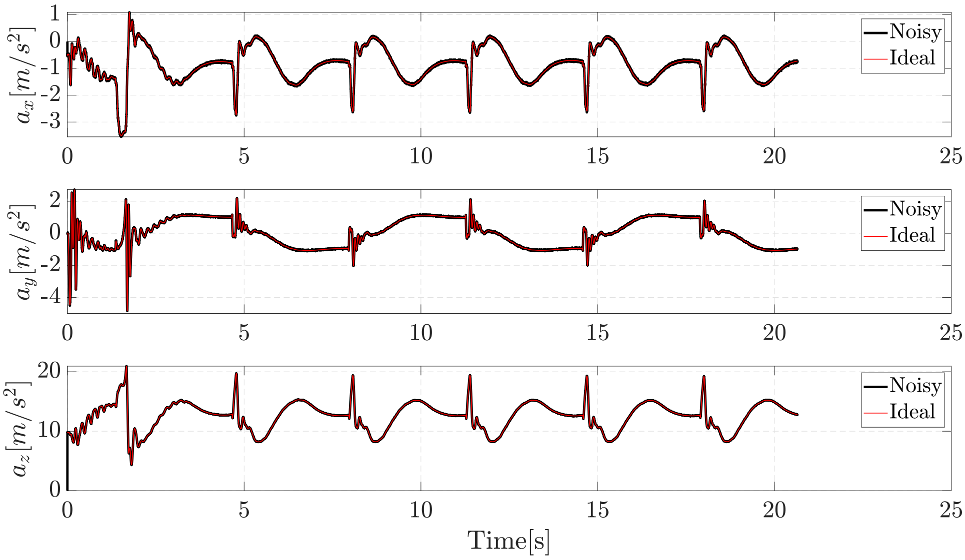


Figure 5.26: Slalom: Comparison between noisy and ideal accelerometer measures

Is is now possible to apply EKF with noisy gyroscope and accelerometer. Even in this case it is assumed to have access first to all velocities and after only to $\mathbf{v}_x$.
Extended Kalman Filter results with complete acceleration information and with knowledge of only $\mathbf{v}_x$ are shown in Fig.(5.27)-(5.28).

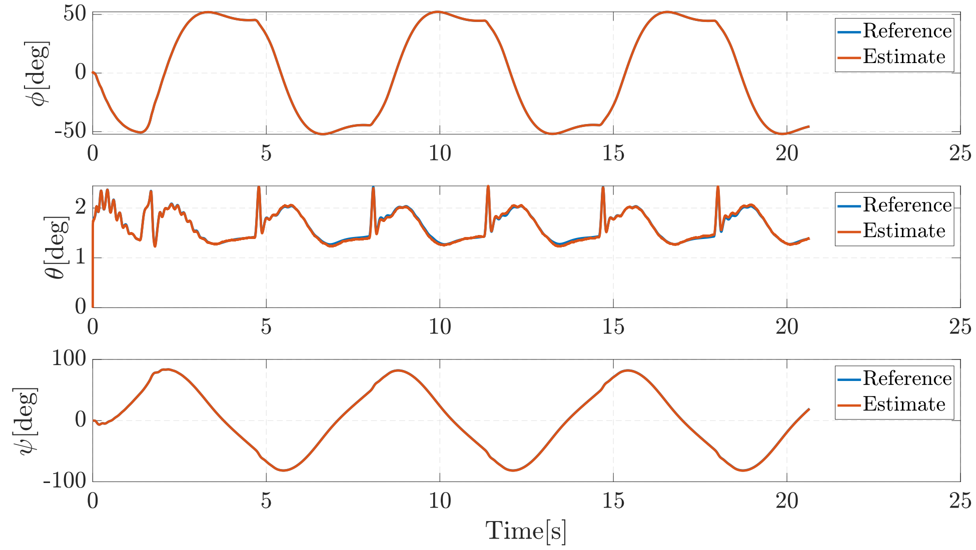


Figure 5.27: Slalom: EKF with complete acceleration information under noisy IMU measurements

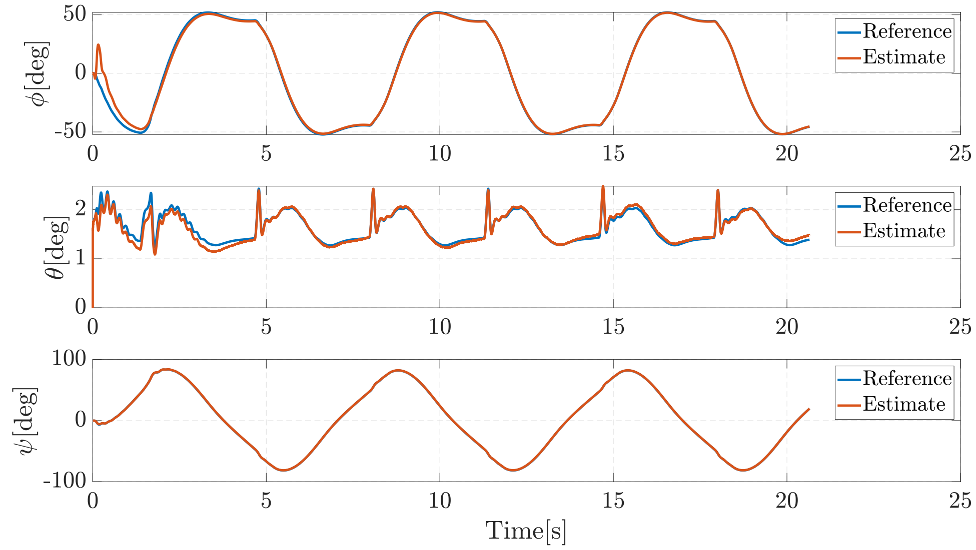


Figure 5.28: Slalom: EKF with only $\mathbf{v}_x$ characterization, under noisy IMU measurements

In both datasets it is shown that even the presence of noise in both IMU does not strongly affects the estimates. Since the additive white noise acts as an high frequency noise, EKF also acts as a smoothing filter, hence its estimates are not affected by high frequency noise.
This ability of smoothing estimates is important in particular for the gyroscope measure, because the filter has little weights on accelerometer.

5.5 Attitude estimation in simulation: conclusion

Different methods under different condition has been tested in simulation, and what has been obtained is that:

- Single IMU sensor attitude estimation is not enough robust to be implemented in real system: gyroscope is dependent on initial condition and low frequency errors, like bias, while accelerometer is sensitive to its characterization and to high frequency noise.

- Extended Kalman Filter can be used as a sensor fusion algorithm, because it handles wrong initial condition, reduced acceleration characterization and additive noise in IMU sensors.
- Each attitude estimation method presented does not include any dependence on the particular system to which the algorithm is applied to. This is clearly visible from state equation, which depends only on angular velocities which are treated as inputs, and does not include any dynamics.
  In the case of a motorcycle, a possible improvement in these methods is the insertion of a dynamical model depending on other variables of the particular vehicle, as done in [7],[6]. In that case, the state space equation consist of vehicle dynamics, while output equations are related to gyroscope and accelerometers.

Chapter 6

Experimental Results

The Extended Kalman Filter is applied with experimental data, in order to understand its robustness with real data.
The experimental dataset is made of a motorcycle riding on a track, and its values are not reported because they are property of the motorcycle manufacturing company.
The experimental dataset is made of readings of gyroscope and accelerometer, together with an estimate of the vehicle velocity. The references for angles have been obtained by laser measurements.
It is important to notice that the vehicle velocity that is estimated is not the body velocity, but it is the velocity in the heading frame.
For this reason it is necessary to modify EKF accelerometer equations in order to have the dependence on the heading velocity.
The accelerometer description with the dependence on the headings velocities is equal to

$$a_x = c_\theta \dot{v}_{x^i} - s_\theta \dot{v}_{z^i} - c_\phi \omega_z v_{y^i} - s_\phi \omega_y v_{y^i} - s_\theta g \tag{6.1}$$

$$a_y = s_\phi s_\theta \dot{v}_{x^i} + c_\phi \dot{v}_{y^i} + s_\phi c_\theta \dot{v}_{z^i} + \frac{s_\phi c_\phi}{c_\theta}\omega_y v_{x^i} + \frac{c_\phi^2}{c_\theta}\omega_z v_{x^i} - \frac{s_\phi^2 s_\theta}{c_\theta}\omega_y v_{y^h} - \frac{s_\phi c_\phi s_\theta}{c_\theta}\omega_z v_{y^h} + s_\phi c_\theta g \tag{6.2}$$

$$a_z = c_\phi s_\theta v_{x^h} - s_\phi v_{y^h} + c_\phi c_\theta v_{z^h} - \frac{s_\phi^2}{c_\theta}\omega_y v_{x^h} - \frac{s_\phi c_\phi}{c_\theta}\omega_z v_{x^h} - \frac{c_\phi s_\phi s_\theta}{c_\theta}\omega_y v_{y^h} - \frac{c_\phi^2 s_\theta}{c_\theta}\omega_z v_{x^h} + c_\phi c_\theta g \tag{6.3}$$

where the complete derivation is found in Appendix.
In this case, since we have the estimate of only $\mathbf{v}_{x^h}$, previous equations become

$$a_x = c_\theta \dot{v}_{x^h} - s_\theta g \tag{6.4}$$

$$a_y = s_\phi s_\theta \dot{v}_{x^h} + \frac{s_\phi c_\phi}{c_\theta}\omega_y v_{x^h} + \frac{c_\phi^2}{c_\theta}\omega_z v_{x^h} + s_\phi c_\theta g \tag{6.5}$$

$$a_z = c_\phi s_\theta v_{x^h} - \frac{s_\phi^2}{c_\theta}\omega_y v_{x^h} - \frac{s_\phi c_\phi}{c_\theta}\omega_z v_{x^h} + c_\phi c_\theta g \tag{6.6}$$

In the next section, main results are shown.

6.1 Experimental Results

In Fig.(6.1) gyroscope readings are shown. These readings presents some high noise that is clearly visible in particular in y-z axis. This readings are used as they are, without applying any filtering technique.
In Fig.(6.2) accelerometer readings are shown. This readings are supposed to be on the vehicle center of mass, and they are referred to the body frame. Even in this case, this readings are used without applying any filter.
With on-board measurements, it is possible to apply EKF to the system and compare its results with references. Weight matrices are chosen as in simulation, with values

$$\mathbf{Q} = q\mathbf{I}, \quad q = 1e-7 \quad \mathbf{R} = r\mathbf{I}, \quad q = 1e2 \quad \mathbf{P}_0 = p\mathbf{I}, \quad p = 1e2 \tag{6.7}$$

In Fig.(6.3) estimates are shown.
From this comparison it is possible to notice that:

- Roll angle is able to track the reference signal with little error.
- Pitch angle presents some errors with respect to the reference, and it is also affected by high frequency noise.
- Pitch reference presents some peaks that are not consistent with real phenomena, and so they are considered as outliers and we expect the algorithm not to follow them.
- References are treated as real data, but they are estimates, and so they can present some errors.

A possible improvement when dealing with real data is to filter all acquired signals, in order to reduce the high frequency components that are clearly visible in IMU measurements, which reflect into pitch estimate.
The applied filter is a second order Butterworth low pass filter with cut-off frequency of $5Hz$.
The EKF estimates, when IMU and velocity measurements are filtered are shown in Fig.(6.4).
Comparing estimates of Fig.(6.3)-(6.4) the main differences are present in pitch angle.
From filtered estimates the high frequency noise component is reduced, and this reflects into a better estimate.
From these results some issues with experimental data has been studied, that are:

- Real measured data is affected by errors, which means that the choice of the filter is fundamental to improve the estimate accuracy.
- Experimental references are estimated too, and so they are not an exact reference, because they depend on the technique that is used to generate them.
- The best weight matrices values are equal to the ones in simulation, which means that the importance that the algorithm gives in simulative and experimental scenario are the same.

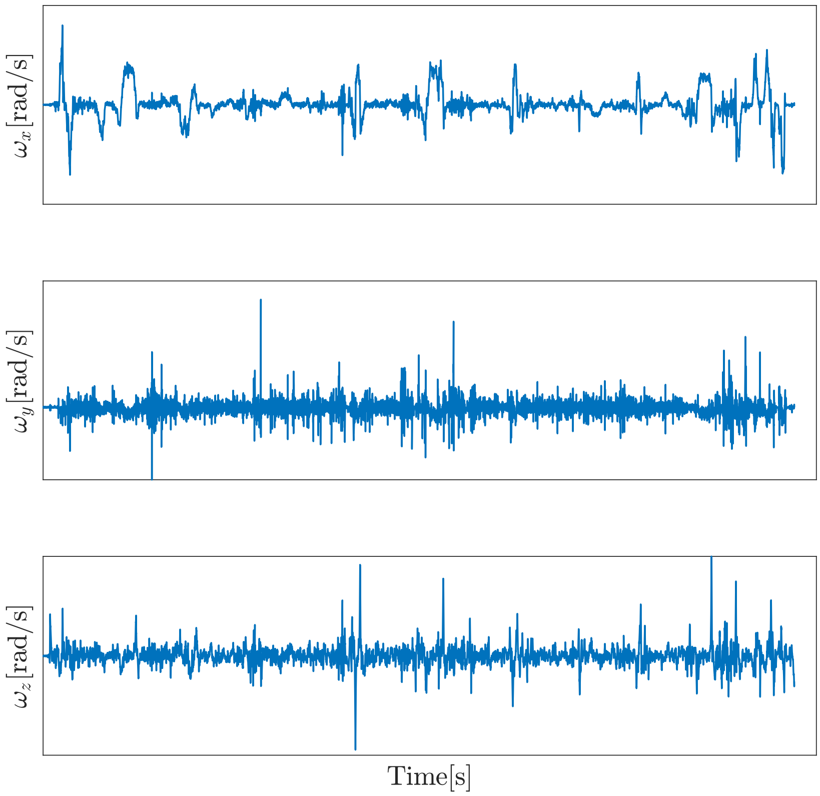


Figure 6.1: Gyroscope readings from on-board IMU

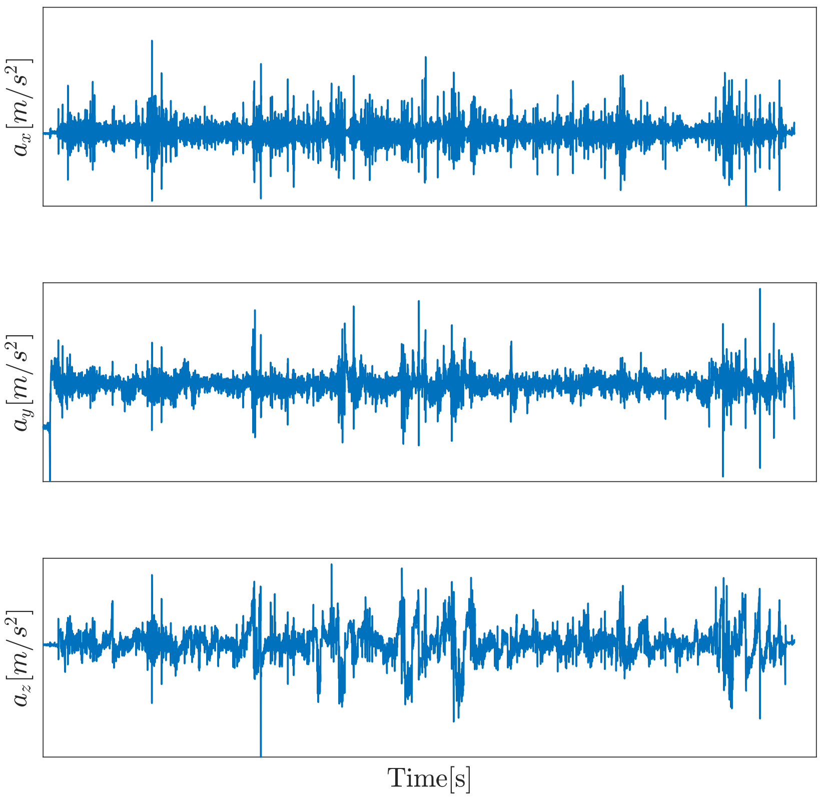


Figure 6.2: Accelerometer readings from on-board IMU

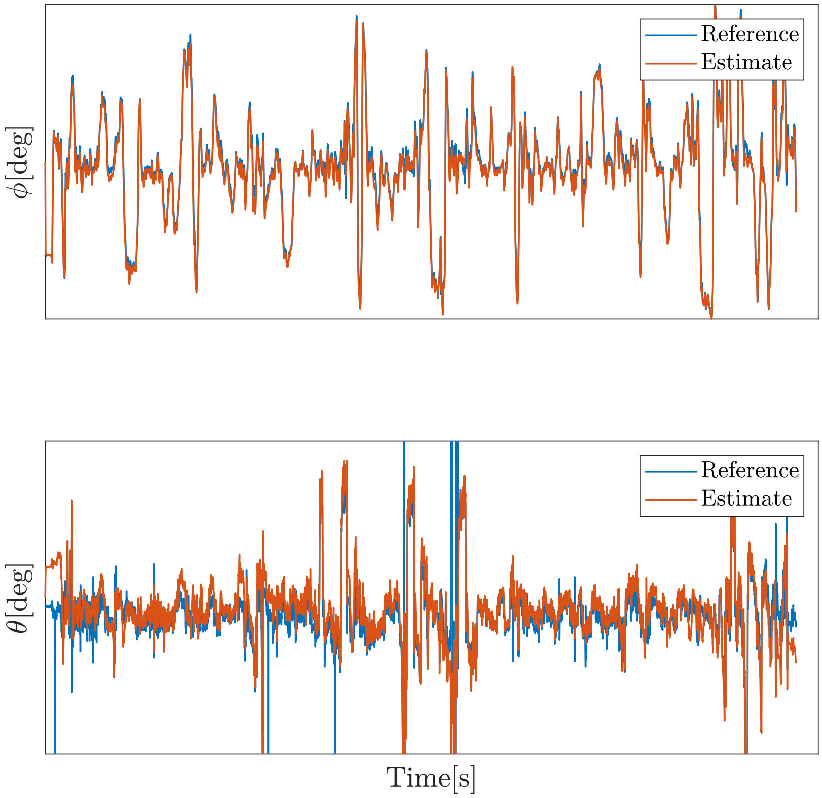


Figure 6.3: EKF attitude estimation from on-board readings

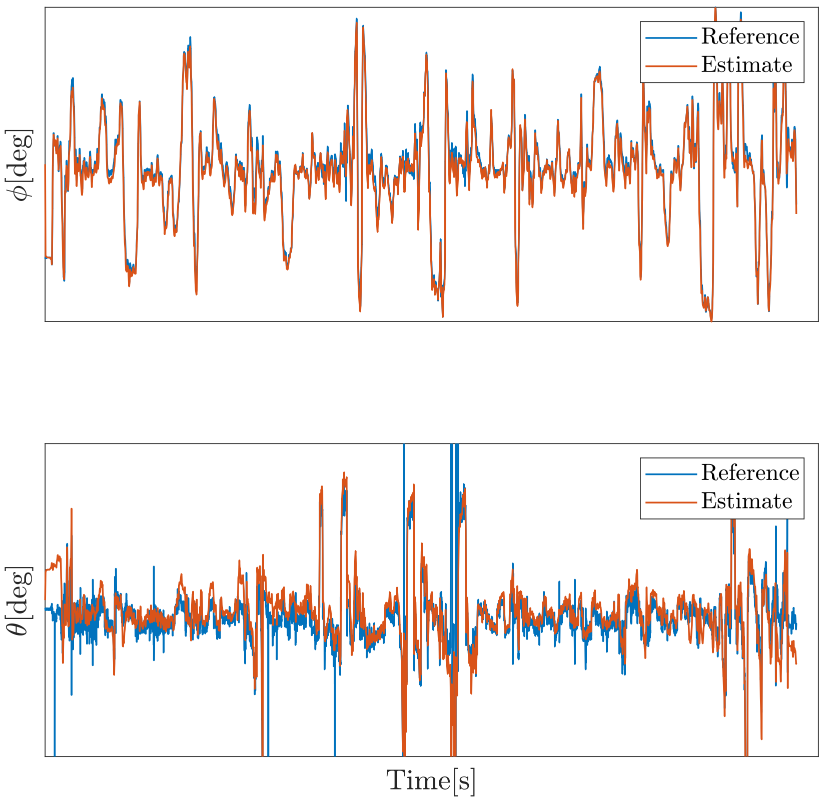


Figure 6.4: EKF attitude estimation from on-board filtered readings

Chapter 7

Conclusion

In this thesis, the attitude estimation problem has been studied. In particular, the accelerometer expression has been characterized based on velocity measurements.
Pros and cons of single IMU sensor attitude estimation have been evaluated, and the necessity of a sensor fusion algorithm has been proved.
From simulative results, Extended Kalman Filter worked as a good sensor fusion algorithm, because it took into account the equations of both gyroscope and accelerometer and it has been able to weight their contribution.
In particular, the filter has been tested with different assumption on accessibility to velocities, in order to prove its robustness with lack of variables.
In simulation it has been shown that the longitudinal velocity is the most important variable to have access to in order to obtain good estimates.
In experimental results, it has been shown that EKF still works well, but it presents some errors due to real values, in particular it is affected by high frequency noise.
Improvements can follow two different ways that are:

- The choice of a different sensor fusion algorithm, as the Moving Horizon Estimator, which follows an optimization approach and then it able to handle constraints. The challenge with this type of algorithm is its complexity because it has to solve the problem through mathematical programming techniques.

- The choice of a different dynamical system, which involves the specific vehicle dynamics. In this case, the main challenges are finding an efficient description of the motorcycle dynamics, which is a complex multi-body system.

Appendix A

Skew symmetric matrix and rotation matrix properties

A.1 Distributive property of rotation matrix under cross product

Prop. *Let* $\mathbf{R}$ *be a rotation matrix, that is* $\mathbf{R}\mathbf{R}^T = \mathbf{I}$, *det*$[\mathbf{R}] = 1$, *and* $\mathbf{a}, \mathbf{b} \in \mathbb{R}^3$. *Then*

$$\mathbf{R}(\mathbf{a} \times \mathbf{b}) = (\mathbf{R}\mathbf{a}) \times (\mathbf{R}\mathbf{b}) \tag{A.1}$$

Proof. Having $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{R}^3$, we define the triple product as

$$< \mathbf{c}, \mathbf{a} \times \mathbf{b} >= \mathbf{c}^T(\mathbf{a} \times \mathbf{b}) = \det \begin{bmatrix} \mathbf{c} & \mathbf{a} & \mathbf{b} \end{bmatrix} \tag{A.2}$$

where the second equivalence is due to the fact that

$$< \mathbf{c}, \mathbf{a} \times \mathbf{b} >= \det \begin{bmatrix} c_x & c_y & c_z \\ a_x & a_y & a_z \\ b_x & b_y & b_z \end{bmatrix} = \det \begin{bmatrix} \mathbf{c}^T \\ \mathbf{a}^T \\ \mathbf{b}^T \end{bmatrix} = \det \begin{bmatrix} \mathbf{c} & \mathbf{a} & \mathbf{b} \end{bmatrix} \tag{A.3}$$

Starting from the third definition of triple product we have that

$$\begin{aligned}
\det[\mathbf{c}, \mathbf{a}, \mathbf{b}] &= \det[\mathbf{R}^T\mathbf{R}]\det[\mathbf{c}, \mathbf{a}, \mathbf{b}] = \det[\mathbf{R}^T]\det[\mathbf{R}\mathbf{c}, \mathbf{R}\mathbf{a}, \mathbf{R}\mathbf{b}] && \text{(A.4)} \\
&= \det[\mathbf{R}^T] < \mathbf{R}\mathbf{c}, \mathbf{R}\mathbf{a} \times \mathbf{R}\mathbf{b} >=< \mathbf{R}\mathbf{c}, \det[\mathbf{R}](\mathbf{R}\mathbf{a} \times \mathbf{R}\mathbf{b}) > && \text{(A.5)} \\
&= (\mathbf{R}\mathbf{c})^T\det[\mathbf{R}](\mathbf{R}\mathbf{a} \times \mathbf{R}\mathbf{b}) = \mathbf{c}^T\mathbf{R}^T\det[\mathbf{R}](\mathbf{R}\mathbf{a} \times \mathbf{R}\mathbf{b}) && \text{(A.6)} \\
&=< \mathbf{c}, \det[\mathbf{R}]\mathbf{R}^T(\mathbf{R}\mathbf{a} \times \mathbf{R}\mathbf{b}) > && \text{(A.7)}
\end{aligned}$$

From this equivalence, exploiting the triple product definition we obtain

$$\begin{aligned}
&<\mathbf{x}, \mathbf{a} \times \mathbf{b} >=< \mathbf{x}, \det[\mathbf{R}]\mathbf{R}^T(\mathbf{R}\mathbf{a} \times \mathbf{R}\mathbf{b}) > && \text{(A.8)} \\
\Rightarrow \quad & \mathbf{a} \times \mathbf{b} = \det[\mathbf{R}]\mathbf{R}^T(\mathbf{R}\mathbf{a} \times \mathbf{R}\mathbf{b}) && \text{(A.9)} \\
\Rightarrow \quad & \mathbf{R}(\mathbf{a} \times \mathbf{b}) = \mathbf{R}\det[\mathbf{R}]\mathbf{R}^T(\mathbf{R}\mathbf{a} \times \mathbf{R}\mathbf{b}) && \text{(A.10)} \\
\Rightarrow \quad & \mathbf{R}(\mathbf{a} \times \mathbf{b}) = (\mathbf{R}\mathbf{a}) \times (\mathbf{R}\mathbf{b}) && \text{(A.11)}
\end{aligned}$$

□

Prop. *Let* $\mathbf{R}$ *be a rotation matrix, that is* $\mathbf{R}\mathbf{R}^T = \mathbf{I}, det[\mathbf{R}] = 1$*, and let* $[\mathbf{s}]_\times \in so(3)$ *be a skew-symmetric matrix. Then*

$$\mathbf{R}[\mathbf{s}]_\times \mathbf{R}^T = [\mathbf{R}\mathbf{s}]_\times \tag{A.12}$$

Proof. Multiplying both members of Eq.(A.12) by $\mathbf{v} \in \mathbb{R}^3$ yields

$$\mathbf{R}[\mathbf{s}]_\times \mathbf{R}^T \mathbf{v} = [\mathbf{R}\mathbf{s}]_\times \mathbf{v} \tag{A.13}$$

Exploiting the cross product in the first member of Eq.(A.13) leads to

$$\mathbf{R}\mathbf{s} \times (\mathbf{R}^T\mathbf{v}) = \mathbf{R}[\mathbf{s} \times (\mathbf{R}^T\mathbf{v})] \tag{A.14}$$

Applying Prop.(A.1) we have that

$$(\mathbf{R}\mathbf{s}) \times (\mathbf{R}\mathbf{R}^T\mathbf{v}) = (\mathbf{R}\mathbf{s}) \times \mathbf{v} = [\mathbf{R}\mathbf{s}]_\times \mathbf{v} \tag{A.15}$$

□

Prop. *Let* $\mathbf{R}_0^2 \in SO(3)$ *be a rotation matrix that represents a sequence of rotation, namely* $\mathbf{R}_0^2 = \mathbf{R}_1^2\mathbf{R}_0^1$*. Then, the angular velocity referred to* $\mathbf{R}_0^2$ *expressed in world frame is equal to*

$$\boldsymbol{\omega}_{2/0}^0 = \boldsymbol{\omega}_{2/1}^0 + \mathbf{R}_1^0\boldsymbol{\omega}_{1/0}^1 \tag{A.16}$$

Proof. From the sequence of rotation

$$\mathbf{R}_0^2 = \mathbf{R}_1^2\mathbf{R}_0^1 \tag{A.17}$$

we can compute its derivative, that is

$$\begin{align}\dot{\mathbf{R}}_0^2 &= \dot{\mathbf{R}}_1^2\mathbf{R}_0^1 + \mathbf{R}_1^2\dot{\mathbf{R}}_0^1 \tag{A.18}\\ &= [\boldsymbol{\omega}_{2/0}^2]_\times^T \mathbf{R}_0^2 \tag{A.19}\end{align}$$

Expanding the matrix derivative for both members of Eq.(A.18) and applying Prop.(A.12) we obtain

$$\begin{align}\dot{\mathbf{R}}_0^2 &= [\boldsymbol{\omega}_{2/1}^2]_\times^T \mathbf{R}_1^2\mathbf{R}_0^1 + \mathbf{R}_1^2[\boldsymbol{\omega}_{1/0}^1]_\times^T \mathbf{R}_0^1 \tag{A.20}\\ &= [\boldsymbol{\omega}_{2/1}^2]_\times^T \mathbf{R}_0^2 + \mathbf{R}_1^2[\boldsymbol{\omega}_{1/0}^1]_\times^T (\mathbf{R}_1^2)^T\mathbf{R}_1^2\mathbf{R}_0^1 \tag{A.21}\\ &= [\boldsymbol{\omega}_{2/1}^2]_\times^T \mathbf{R}_0^2 + [\mathbf{R}_1^2\boldsymbol{\omega}_{1/0}^1]_\times^T \mathbf{R}_0^2 \tag{A.22}\end{align}$$

By additivity of skew-symmetric matrices we have that

$$\dot{\mathbf{R}}_0^2 = [\boldsymbol{\omega}_{2/1}^2 + \mathbf{R}_1^2\boldsymbol{\omega}_{1/0}^1]_\times^T \mathbf{R}_0^2 \tag{A.23}$$

and from Eq.(A.16) we have that

$$\boldsymbol{\omega}_{2/0}^2 = \boldsymbol{\omega}_{2/1}^2 + \mathbf{R}_1^2\boldsymbol{\omega}_{1/0}^1 \tag{A.24}$$

If we express the previous relation in world frame we obtain

$$\begin{align}\mathbf{R}_2^0\boldsymbol{\omega}_{2/0}^2 = \boldsymbol{\omega}_{2/0}^0 &= \mathbf{R}_2^0\boldsymbol{\omega}_{2/1}^2 + \mathbf{R}_2^0\mathbf{R}_1^2\boldsymbol{\omega}_{1/0}^1 \tag{A.25}\\ &= \boldsymbol{\omega}_{2/1}^0 + \mathbf{R}_1^0\mathbf{R}_2^1\mathbf{R}_1^2\boldsymbol{\omega}_{1/0}^1 \tag{A.26}\\ &= \boldsymbol{\omega}_{2/1}^0 + \mathbf{R}_1^0\boldsymbol{\omega}_{1/0}^1 \tag{A.27}\end{align}$$

□

Appendix B

Acceleration expression with velocity on heading frame

In the case that we have access to velocity measured by GPS, namely v_{x^h}, this does not coincide with the usual v_x since it is not affected by pitch and roll effects.
As we can see in Fig.(B.1), the effective velocity of the vehicle is the vector that is always parallel to the ground, that is different from the general v_x, which direction coincides with the nose.
The difference between v_x and v_{x^h} is mainly visible in the case of pure wheeling, where v_{x^h} remains constant, while v_x decreases and v_z increases.
It is important then to find the dependence of body velocities with respect to the measured ones.

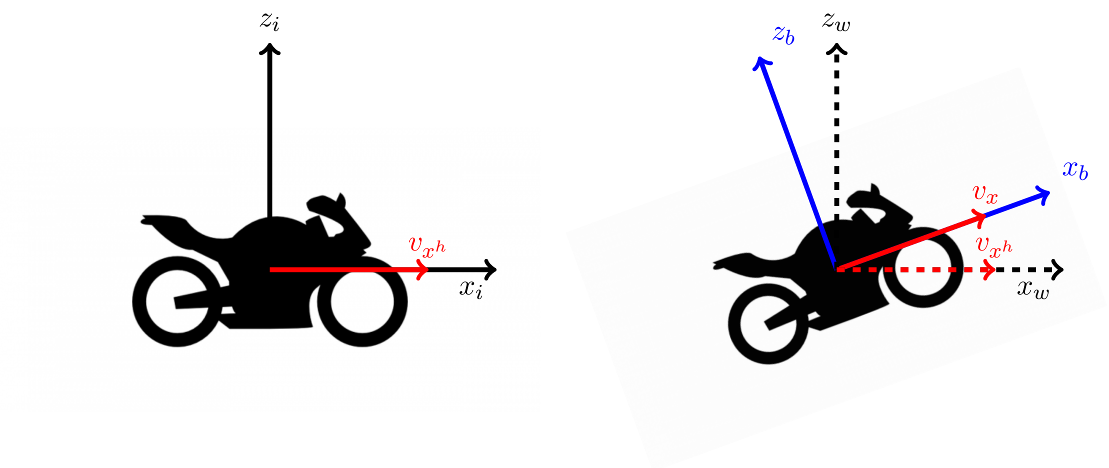


Figure B.1: Heading frame and Body frame

Defining an heading frame between the global and the body one as (x^h, y^h, z^h), we want to find the transformation that relates this frame to the other ones.
The three reference frames are shown in Fig.(B.2), where we can notice the sequence of rotation between global, heading and body frame.

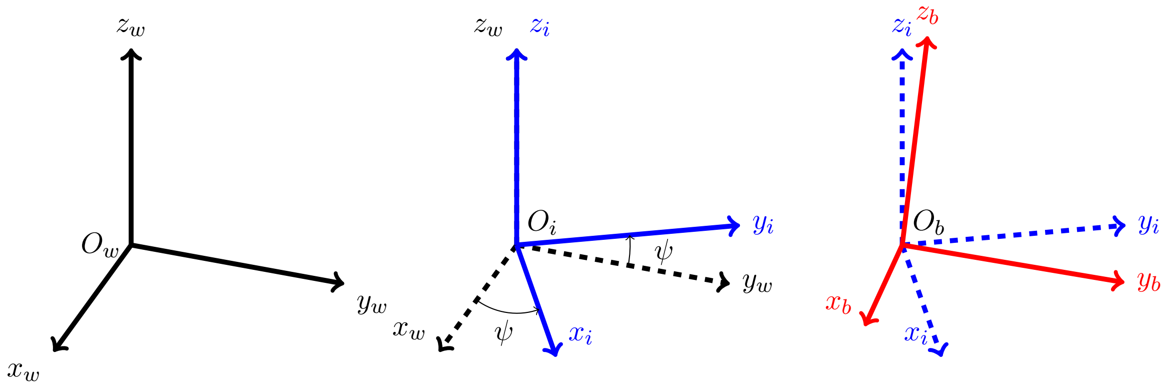


Figure B.2: World, Heading and Body reference systems

This heading frame always maintains the $x - y$ horizontal plane, and then it only rotates about the z axis with respect to the global frame. Then, the relations between global and body frame are equal to

$$\mathbf{R}_w^h(\psi) = \mathbf{R}_z(\psi) = \mathbf{R}_w^b(0,0,\psi) = \begin{bmatrix} c_\psi & s_\psi & 0 \\ -s_\psi & c_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{B.1}$$

$$\mathbf{R}_h^b(\phi,\theta) = \mathbf{R}_{xy}(\phi,\theta) = \mathbf{R}_w^b(\phi,\theta,0) = \begin{bmatrix} c_\theta & 0 & -s_\theta \\ s_\phi s_\theta & c_\phi & s_\phi c_\theta \\ c_\phi s_\theta & -s_\phi & c_\phi c_\theta \end{bmatrix} \tag{B.2}$$

Suppose that we have access to velocities of the heading frame $v_{x^h}, v_{y^h}, v_{z^h}$, that is the most suitable case, we want to find an expression for the body acceleration which depends on those velocities. In this case we are supposing that our IMU is mounted on the center of mass of the vehicle, accelerometer readings are in body frame, while velocities are read in heading frame. First, we express body velocities as a function of heading velocities through the heading rotation matrix.

$$\begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} = \mathbf{R}(\phi,\theta,0) \begin{bmatrix} v_x^h \\ v_y^h \\ v_z^h \end{bmatrix} = \begin{bmatrix} c_\theta & 0 & -s_\theta \\ s_\phi s_\theta & c_\phi & s_\phi c_\theta \\ c_\phi s_\theta & -s_\phi & c_\phi c_\theta \end{bmatrix} \begin{bmatrix} v_x^h \\ v_y^h \\ v_z^h \end{bmatrix} = \begin{bmatrix} c_\theta v_{x^h} - s_\theta v_{z^h} \\ s_\phi s_\theta v_{x^h} + c_\phi v_{y^h} + s_\phi c_\theta v_{z^h} \\ c_\phi s_\theta v_{x^h} - s_\phi v_{y^h} + c_\phi c_\theta v_{z^h} \end{bmatrix} \tag{B.3}$$

Recalling the acceleration expression

$$\mathbf{a} = \dot{\mathbf{v}} + \boldsymbol{\omega} \times \mathbf{v} \tag{B.4}$$

it depends on body velocities and their derivatives. It is possible to derive the expression for time derivative of the velocities differentiating the last term of Eq.(B.3), obtaining

$$\dot{v}_x = -s_\theta\dot{\theta}v_{x^h} + c_\theta\dot{v}_{x^h} - c_\theta\dot{\theta}v_{z^h} - s_\theta\dot{v}_{z^h} \tag{B.5}$$

$$\dot{v}_y = c_\phi\dot{\phi}s_\theta v_{x^h} + s_\phi c_\theta\dot{\theta}v_{x^h} + s_\phi s_\theta\dot{v}_{x^h} - s_\phi\dot{\phi}v_{y^h} + c_\phi\dot{v}_{y^h} + c_\phi\dot{\phi}c_\theta v_{z^h} - s_\phi s_\theta\dot{\theta}v_{z^h} + s_\phi c_\theta\dot{v}_{z^h} \tag{B.6}$$

$$\dot{v}_z = -s_\phi\dot{\phi}s_\theta v_{x^h} + c_\phi c_\theta\dot{\theta}v_{x^h} - c_\phi s_\theta\dot{v}_{x^h} - c_\phi\dot{\phi}v_{y^h} - s_\phi\dot{v}_{y^h} - s_\phi\dot{\phi}c_\theta v_{z^h} - c_\phi s_\theta\dot{\theta}v_{z^h} + c_\phi c_\theta\dot{v}_{z^h} \tag{B.7}$$

We can notice that derivatives depend on angle rate of ϕ and θ, while we only want to have dependence on angular velocities. Substituting the angle rates with their relation with the angular velocities we obtain

$$\begin{aligned}
\dot{v}_x =& - s_\theta(c_\phi\omega_y - s_\phi\omega_z)v_{x^h} + c_\theta\dot{v}_{x^h} - c_\theta(c_\phi\omega_y - s_\phi\omega_z)v_{z^h} - s_\theta\dot{v}_{z^h} && \text{(B.8)}\\
\dot{v}_y =& + c_\phi(\omega_x + s_\phi t_\theta\omega_y + c_\phi t_\theta\omega_z)s_\theta v_{x^h} + s_\phi c_\theta(c_\phi\omega_y - s_\phi\omega_z)v_{x^h} + s_\phi s_\theta\dot{v}_{x^h} - \dots \\
&\dots - s_\phi(\omega_x + s_\phi t_\theta\omega_y + c_\phi t_\theta\omega_z)v_{y^h} + c_\phi\dot{v}_{y^h} + c_\phi(\omega_x + s_\phi t_\theta\omega_y + c_\phi t_\theta\omega_z)c_\theta v_{z^h} - \dots \\
&\dots - s_\phi s_\theta(c_\phi\omega_y - s_\phi\omega_z)v_{z^h} + s_\phi c_\theta\dot{v}_{z^h} && \text{(B.9)}\\
\dot{v}_z =& - s_\phi(\omega_x + s_\phi t_\theta\omega_y + c_\phi t_\theta\omega_z)s_\theta v_{x^h} + c_\phi c_\theta(c_\phi\omega_y - s_\phi\omega_z)v_{x^h} - c_\phi s_\theta\dot{v}_{x^h} - \dots \\
&\dots c_\phi(\omega_x + s_\phi t_\theta\omega_y + c_\phi t_\theta\omega_z)v_{y^h} - s_\phi\dot{v}_{y^h} - s_\phi(\omega_x + s_\phi t_\theta\omega_y + c_\phi t_\theta\omega_z)c_\theta v_{z^h} - \dots \\
&\dots - c_\phi s_\theta(c_\phi\omega_y - s_\phi\omega_z)v_{z^h} + c_\phi c_\theta\dot{v}_{z^h} && \text{(B.10)}
\end{aligned}$$

Regarding to centrifugal acceleration, $\boldsymbol{\omega} \times \mathbf{v}$, again we substitute body velocities, resulting

$$[\boldsymbol{\omega} \times \mathbf{v}]_x = -\omega_z(s_\phi s_\theta v_{x^h} + c_\phi v_{y^h} + s_\phi c_\theta v_{z^h}) + \omega_y(c_\phi s_\theta v_{x^h} - s_\phi v_{y^h} + c_\phi c_\theta v_{z^h}) \tag{B.11}$$

$$[\boldsymbol{\omega} \times \mathbf{v}]_y = +\omega_z(c_\theta v_{x^h} - s_\theta v_{z^h}) - \omega_x(c_\phi s_\theta v_{x^h} - s_\phi v_{y^h} + c_\phi c_\theta v_{z^h}) \tag{B.12}$$

$$[\boldsymbol{\omega} \times \mathbf{v}]_z = -\omega_y(c_\theta v_{x^h} - s_\theta v_{z^h}) + \omega_x(s_\phi s_\theta v_{x^h} + c_\phi v_{y^h} + s_\phi c_\theta v_{z^h}) \tag{B.13}$$

$$\tag{B.14}$$

Combining longitudinal and centrifugal term of acceleration we obtain

$$\begin{aligned}
a_x =& - s_\theta(c_\phi\omega_y - s_\phi\omega_z)v_{x^h} + c_\theta\dot{v}_{x^h} - c_\theta(c_\phi\omega_y - s_\phi\omega_z)v_{z^h} - s_\theta\dot{v}_{z^h} \dots \\
&\dots - \omega_z(s_\phi s_\theta v_{x^h} + c_\phi v_{y^h} + s_\phi c_\theta v_{z^h}) + \omega_y(c_\phi s_\theta v_{x^h} - s_\phi v_{y^h} + c_\phi c_\theta v_{z^h}) && \text{(B.15)}\\
a_y =& + c_\phi(\omega_x + s_\phi t_\theta\omega_y + c_\phi t_\theta\omega_z)s_\theta v_{x^h} + s_\phi c_\theta(c_\phi\omega_y - s_\phi\omega_z)v_{x^h} + s_\phi s_\theta\dot{v}_{x^h} - \dots \\
&\dots - s_\phi(\omega_x + s_\phi t_\theta\omega_y + c_\phi t_\theta\omega_z)v_{y^h} + c_\phi\dot{v}_{y^h} + c_\phi(\omega_x + s_\phi t_\theta\omega_y + c_\phi t_\theta\omega_z)c_\theta v_{z^h} - \dots \\
&\dots - s_\phi s_\theta(c_\phi\omega_y - s_\phi\omega_z)v_{z^h} + s_\phi c_\theta\dot{v}_{z^h} + \omega_z(c_\theta v_{x^h} - s_\theta v_{z^h}) - \omega_x(c_\phi s_\theta v_{x^h} - s_\phi v_{y^h} + c_\phi c_\theta v_{z^h}) \\
& && \text{(B.16)}\\
a_z =& - s_\phi(\omega_x + s_\phi t_\theta\omega_y + c_\phi t_\theta\omega_z)s_\theta v_{x^h} + c_\phi c_\theta(c_\phi\omega_y - s_\phi\omega_z)v_{x^h} - c_\phi s_\theta\dot{v}_{x^h} - \dots \\
&\dots c_\phi(\omega_x + s_\phi t_\theta\omega_y + c_\phi t_\theta\omega_z)v_{y^h} - s_\phi\dot{v}_{y^h} - s_\phi(\omega_x + s_\phi t_\theta\omega_y + c_\phi t_\theta\omega_z)c_\theta v_{z^h} - \dots \\
&\dots - c_\phi s_\theta(c_\phi\omega_y - s_\phi\omega_z)v_{z^h} + c_\phi c_\theta\dot{v}_{z^h} - \omega_y(c_\theta v_{x^h} - s_\theta v_{z^h}) + \omega_x(s_\phi s_\theta v_{x^h} + c_\phi v_{y^h} + s_\phi c_\theta v_{z^h}) \\
& && \text{(B.17)}
\end{aligned}$$

The simplification of acceleration terms leads to the final form for the body acceleration depending on heading velocities, that is

$$a_x = c_\theta\dot{v}_{x^h} - s_\theta\dot{v}_{z^h} - c_\phi\omega_z v_{y^h} - s_\phi\omega_y v_{y^h} \tag{B.18}$$

$$a_y = s_\phi s_\theta\dot{v}_{x^h} + c_\phi\dot{v}_{y^h} + s_\phi c_\theta\dot{v}_{z^h} + \frac{s_\phi c_\phi}{c_\theta}\omega_y v_{x^h} + \frac{c_\phi^2}{c_\theta}\omega_z v_{x^h} - \frac{s_\phi^2 s_\theta}{c_\theta}\omega_y v_{y^h} - \frac{s_\phi c_\phi s_\theta}{c_\theta}\omega_z v_{y^h} \tag{B.19}$$

$$a_z = c_\phi s_\theta v_{x^h} - s_\phi v_{y^h} + c_\phi c_\theta v_{z^h} - \frac{s_\phi^2}{c_\theta}\omega_y v_{x^h} - \frac{s_\phi c_\phi}{c_\theta}\omega_z v_{x^h} - \frac{c_\phi s_\phi s_\theta}{c_\theta}\omega_y v_{y^h} - \frac{c_\phi^2 s_\theta}{c_\theta}\omega_z v_{x^h} \tag{B.20}$$

Eq.(B.18)-(B.19)-(B.20) can be written in matrix form, obtaining

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} c_\theta & 0 & -s_\theta \\ s_\phi s_\theta & c_\phi & s_\phi c_\theta \\ c_\phi s_\theta & -s_\phi & c_\phi c_\theta \end{bmatrix} \begin{bmatrix} \dot{v}_{x^h} \\ \dot{v}_{y^h} \\ \dot{v}_{z^h} \end{bmatrix} + \begin{bmatrix} 0 & -c_\phi\omega_z - s_\phi\omega_y & 0 \\ \frac{s_\phi c_\phi}{c_\theta}\omega_y + \frac{c_\phi^2}{c_\theta}\omega_z & -\frac{s_\phi^2 s_\theta}{c_\theta}\omega_y - \frac{s_\phi s_\theta c_\phi}{c_\theta}\omega_z & 0 \\ -\frac{s_\phi c_\phi}{c_\theta}\omega_z - \frac{s_\phi^2}{c_\theta}\omega_y & -\frac{c_\phi^2 s_\theta}{c_\theta}\omega_z - \frac{s_\phi s_\theta c_\phi}{c_\theta}\omega_y & 0 \end{bmatrix} \begin{bmatrix} v_{x^h} \\ v_{y^h} \\ v_{z^i} \end{bmatrix} \tag{B.21}$$

$$= \begin{bmatrix} c_\theta & 0 & -s_\theta \\ s_\phi s_\theta & c_\phi & s_\phi c_\theta \\ c_\phi s_\theta & -s_\phi & c_\phi c_\theta \end{bmatrix} \begin{bmatrix} \dot{v}_{x^h} \\ \dot{v}_{y^h} \\ \dot{v}_{z^h} \end{bmatrix} + \frac{s_\phi\omega_y + c_\phi\omega_z}{c_\theta} \begin{bmatrix} 0 & -1 & 0 \\ +c_\phi & -s_\phi s_\theta & 0 \\ -s_\phi & -c_\phi s_\theta & 0 \end{bmatrix} \begin{bmatrix} v_{x^h} \\ v_{y^h} \\ v_{z^h} \end{bmatrix} \tag{B.22}$$

In order to understand the meaning of Eq.(B.22), we can substitute back angle rates from the angular velocities, where we do not report derivation for simplicity.
The acceleration expression results

$$a_x = c_\theta \dot{v}_{x^h} - s_\theta \dot{v}_{z^h} - c_\theta \dot{\psi} v_{y^h} \tag{B.23}$$

$$a_y = s_\phi s_\theta \dot{v}_{x^h} + c_\phi \dot{v}_{y^h} + s_\phi c_\theta \dot{v}_{z^h} + c_\phi \dot{\psi} v_{x^h} - s_\phi s_\theta \dot{\psi} v_{y^h} \tag{B.24}$$

$$a_z = c_\phi s_\theta \dot{v}_{x^h} - s_\phi \dot{v}_{y^h} + c_\phi c_\theta \dot{v}_{z^h} - s_\phi \dot{\psi} v_{x^h} - c_\phi s_\theta \dot{\psi} v_{y^h} \tag{B.25}$$

It is possible to write the expression through rotation matrix, that is

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \mathbf{R}_h^b(\phi, \theta) \begin{bmatrix} \dot{v}_{x^h} \\ \dot{v}_{y^h} \\ \dot{v}_{z^h} \end{bmatrix} + \mathbf{R}_h^b(\phi, \theta) \begin{bmatrix} -\dot{\psi} v_{y^h} \\ +\dot{\psi} v_{x^h} \\ 0 \end{bmatrix} \tag{B.26}$$

The most important version of Eq.(B.26) is the one for which we have access only to v_{x^h}, that means that we suppose to have $v_{y^h}, v_{z^h} \simeq 0$.
Under this hypothesis Eq.(B.26) becomes

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \mathbf{R}_h^b(\phi, \theta) \begin{bmatrix} \dot{v}_{x^h} \\ 0 \\ 0 \end{bmatrix} + \mathbf{R}_h^b(\phi, \theta) \begin{bmatrix} 0 \\ +\dot{\psi} v_{x^h} \\ 0 \end{bmatrix} \tag{B.27}$$

Expanding the matrices we obtain the Eq.(B.18) with the knowledge of v_{x^h} only, that is

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} c_\theta & 0 & -s_\theta \\ s_\phi s_\theta & c_\phi & s_\phi c_\theta \\ c_\phi s_\theta & -s_\phi & c_\phi c_\theta \end{bmatrix} \begin{bmatrix} \dot{v}_{x^h} \\ 0 \\ 0 \end{bmatrix} + \frac{s_\phi \omega_y + c_\phi \omega_z}{c_\theta} \begin{bmatrix} 0 & -1 & 0 \\ +c_\phi & -s_\phi s_\theta & 0 \\ -s_\phi & -c_\phi s_\theta & 0 \end{bmatrix} \begin{bmatrix} v_{x^h} \\ 0 \\ 0 \end{bmatrix} \tag{B.28}$$

$$= \begin{bmatrix} c_\theta \dot{v}_{x^h} \\ s_\phi s_\theta \dot{v}_{x^h} + \frac{s_\phi c_\phi}{c_\theta} \omega_y v_{x^h} + \frac{c_\phi^2}{c_\theta} \omega_z v_{x^h} \\ c_\phi s_\theta v_{x^h} - \frac{s_\phi^2}{c_\theta} \omega_y v_{x^h} - \frac{s_\phi c_\phi}{c_\theta} \omega_z v_{x^h} \end{bmatrix} \tag{B.29}$$

In the case where we are in a frame which is out of the center of mass we have the additive terms described in Ch.(3), which are not related to the heading frame, since they do not depend on the body velocity. An example fo this case in shown in Fig.(B.3). The acceleration formula with the IMU out of the center of mass, combined with the headings velocities expression leads to

$$\mathbf{a} = \mathbf{R}\mathbf{v}^h + \boldsymbol{\omega} \times \mathbf{R}\mathbf{v}^h + \dot{\boldsymbol{\omega}} \times \mathbf{r} + \boldsymbol{\omega} \times [\boldsymbol{\omega} \times \mathbf{r}] \tag{B.30}$$

whose explicit version is

$$a_x = c_\theta \dot{v}_{x^h} - s_\theta \dot{v}_{z^h} - c_\phi \omega_z v_{y^h} - s_\phi \omega_y v_{y^h} - \dot{\omega}_z r_y + \dot{\omega}_y r_z - \omega_z^2 r_x - \omega_z \omega_y r_z - \omega_y^2 r_x + \omega_y \omega_x r_y \tag{B.31}$$

$$a_y = s_\phi s_\theta \dot{v}_{x^h} + c_\phi \dot{v}_{y^h} + s_\phi c_\theta \dot{v}_{z^h} + \frac{s_\phi c_\phi}{c_\theta} \omega_y v_{x^h} + \frac{c_\phi^2}{c_\theta} \omega_z v_{x^h} - \frac{s_\phi^2 s_\theta}{c_\theta} \omega_y v_{y^h} - \frac{s_\phi c_\phi s_\theta}{c_\theta} \omega_z v_{y^h} + \dots$$
$$\dots + \dot{\omega}_z r_x - \dot{\omega}_x r_z - \omega_z^2 r_y + \omega_z \omega_y r_z + \omega_x \omega_y r_x - \omega_x^2 r_y \tag{B.32}$$

$$a_z = c_\phi s_\theta v_{x^h} - s_\phi v_{y^h} + c_\phi c_\theta v_{z^h} - \frac{s_\phi^2}{c_\theta} \omega_y v_{x^h} - \frac{s_\phi c_\phi}{c_\theta} \omega_z v_{x^h} - \frac{c_\phi s_\phi s_\theta}{c_\theta} \omega_y v_{y^h} - \frac{c_\phi^2 s_\theta}{c_\theta} \omega_z v_{x^h} + \dots$$
$$\dots - \dot{\omega}_y r_x - \dot{\omega}_x r_y - \omega_y \omega_z r_y - \omega_y^2 r_z + \omega_x \omega_z r_x - \omega_x^2 r_z \tag{B.33}$$

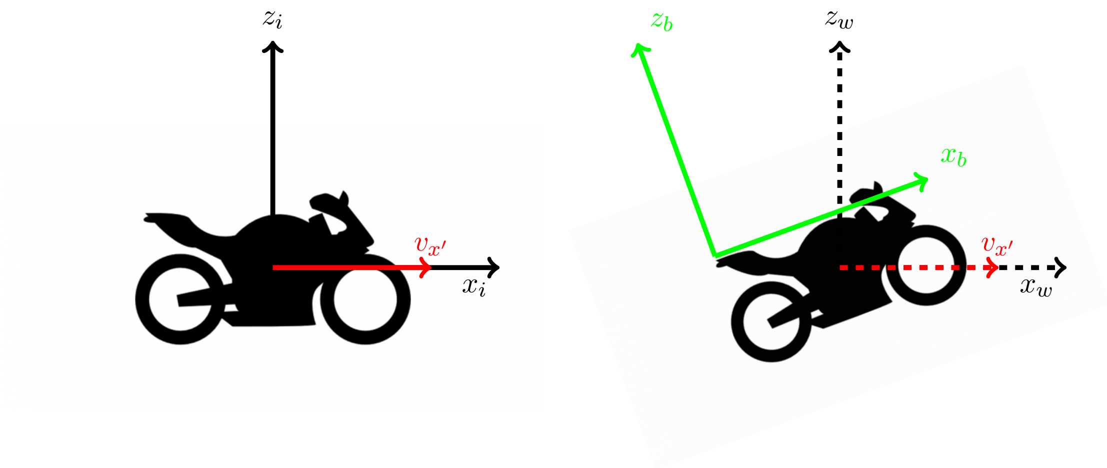


Figure B.3: Heading, Body frame and Sensor frame

Bibliography

[1] M.Kok, J.D. Holy, T.B. Schonz, "Using Inertial Sensors for Position and Orientation Estimation," *Foundations and Trends in Signal Processing*, 2017.

[2] Y. Guan, X. Song, "Sensor Fusion of Gyroscope and Accelerometer for Low-Cost Attitude Determination System," *IEEE*, 2018.

[3] P. Pounds, T. Hamel, R. Mahony, "Attitude Control of Rigid Body Dynamics From Biased IMU Measurements," *IEEE Conference on Decision and Control*, 2007.

[4] I. Boniolo, M. Norgia, M. Tanelli, C. Svelto, S. M. Savaresi, "Performance analysis of an optical distance sensor for roll angle estimation in sport motorcycles," *IFAC Proceedings Volumes*, 2008.

[5] L. Gasbarro, A. Beghi, R. Frezza, F. Nori, "Motorcycle trajectory reconstruction by integration of vision and MEMS accelerometers," *Proceedings of the IEEE Conference on Decision and Control*, 2004.

[6] D. V. Gowda, K. D. V. Shivashankar, A. C. Ramachandra, C. Pandurangappa, "Optimization of Motorcycle Pitch with Non Linear Control," *IEEE Internetional Conference On Recent Trends In Eletronics Information Communication Technology*, 2016.

[7] L. Nehaoua, D. Ichalal, H. Arioui, J. Davila, S. Mammar, "An Unknown Input HOSM Approach to Estimate Lean and Steering Motorcycle Dynamics," *IEEE Transaction on Vehicular Technology, Institute of Electrical and Electronics Engineers*, 2014.

[8] V. Cossalter, R. Lot, "A Motorcycle Multi-Body Model for Real Time Simulations Based on the Natural Coordinates Approach," *Vehicle System Dynamics*, 2002.

[9] S.O.H. Madgwick, "Estimation of IMU and MARG orientation using a gradient descent algorithm," *2011 IEEE International Conference on Rehabilitation Robotics*, 2011.

[10] I. Boniolo, S. Corbetta, S.M. Savaresi, "Attitude estimation of a motorcycle in a Kalman filtering framework," *IFAC Symposium Advances in Automotive Control*, 2010.

[11] I. Boniolo, S. Corbetta, S.M. Savaresi, "Attitude estimation of a motorcycle via Unscented Kalman Filter," *IFAC Symposium Advances in Automotive Control*, 2010.

[12] J. Crassidis, L. Markley, Y. Cheng, "Survey of Nonlinear Attitude Estimation Methods," *Journal of Guidance Control and Dynamics*, 2007.

[13] O. J. Woodman, "An introduction to inertial navigation," *Technical reports published by the University of Cambridge*, 2007.

[14] T. Hamel, R. Mahony, "Attitude estimation on SO(3) based on direct inertial measurements," *IEEE International Conference on Robotics and Automation*, 2006.

[15] N. Metni, J.Pflimlin, T. Hamel, P. Soueres, "Attitude and gyro bias estimation for a VTOL UAV," *Elsevier, Control Engineering Practice*, 2006.

[16] J. Humpherys, P. Redd, J. West, "A Fresh Look at the Kalman Filter," *SIAM Society for Industrial and Applied Mathematics Vol. 54*, 2012.

[17] J. Humpherys, J. West, "Kalman Filtering with Newton's Method," *IEEE Control System Magazine*, 2010.