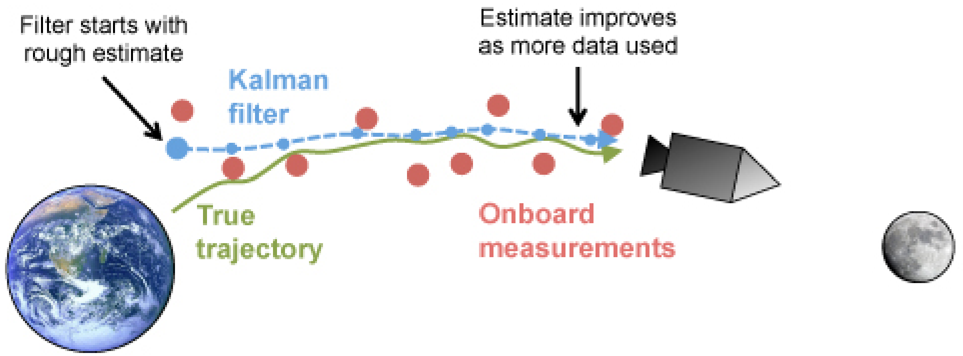
1 Rehabilitation Engineering

§1.7 *Sensor fusion*

Lorenzo Chiari, PhD

Dipartmento di Ingegneria dell'Energia Elettrica e dell'Informazione
Alma Mater Studiorum – Università di Bologna

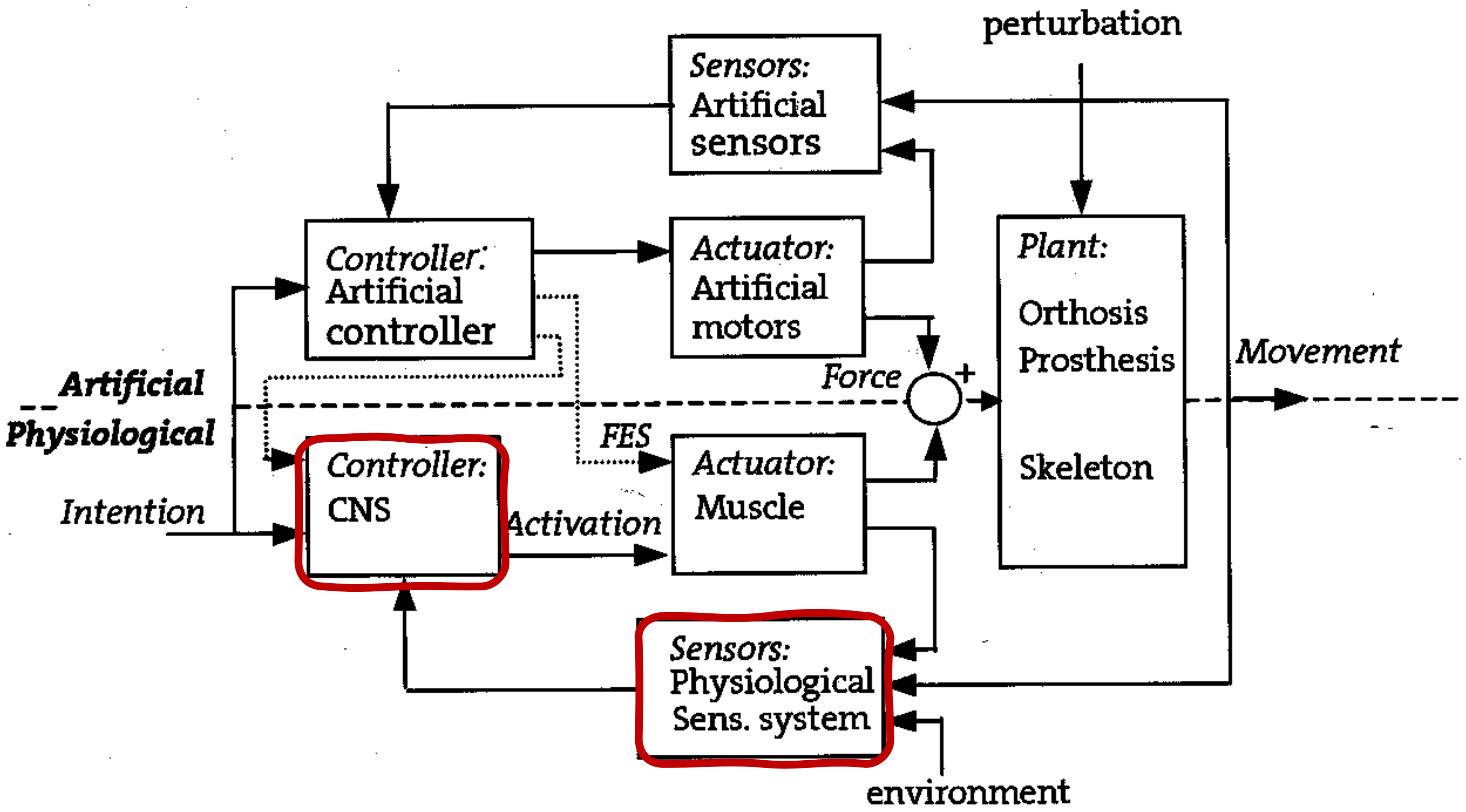
lorenzo.chiari@unibo.it



Outline

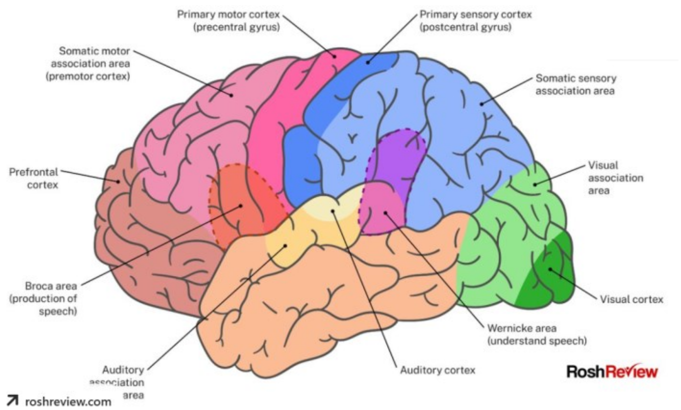
- ✓ Introduction to sensor fusion
- ✓ Kalman filter theory: linear and non-linear filters
- ✓ Sensor fusion for inertial-tracking
- ✓ Example of application: 2D Kalman Filter - Mechanical model-based application

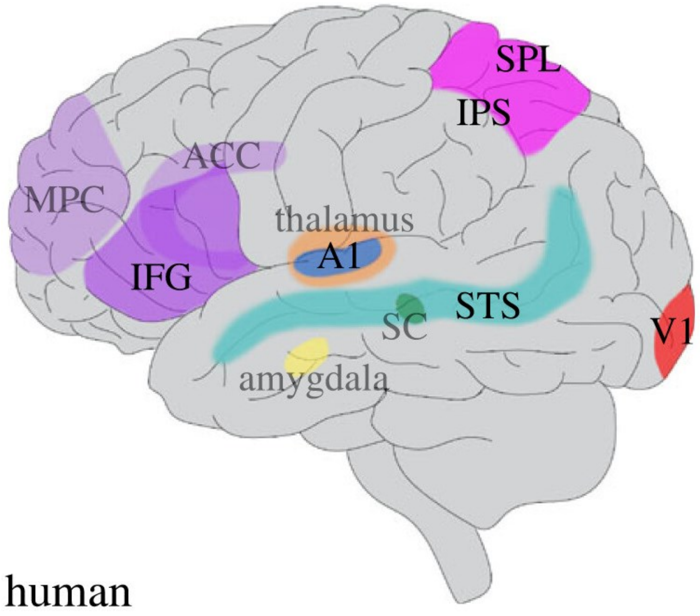
The biological inspiration



Multisensory integration (MSI) in the brain

On the way to deciphering the neural basis of MSI, anatomical tracing, electrophysiology and neuroimaging experiments revealed several key areas in the mammalian brain.
The first brain area where MSI was identified is the Superior Colliculus (SC). Early studies in cats and primates identified the role of the SC in the accurate orientation of the multisensory stimuli in space. Not only the SC but several cortical areas were also identified as hotspots for MSI. One is the association cortex, which receives convergent inputs from multiple sensory cortices. These association cortical areas are scattered across the parietal, temporal and frontal lobes.
Beyond the cortex, even sensory thalamic nuclei were involved in processing multisensory information. Therefore, MSI occurs in multiple brain areas in parallel, and this might be important for the flexible integration of sensory stimuli that constantly bombard an animal with distinct features in space and time.





Introduction to sensor fusion

Sensor fusion involves combining data from several sensors to obtain better information for perception.

Humans and animals process multiple sensory data to reason and act and the same principle is applied in multi-sensor data fusion. *Multi-sensor fusion combines data from different sensors into a common representation format*.

In developing robotic systems, multi-sensor fusion plays a crucial role since interaction with the environment is instrumental in successful execution of the task.

Significant applications of multi-sensor fusion can be found in applications such as mobile robots, defense systems (such as target tracking), transportation systems and industry, and of course medicine.

Chandrasekaran et al., SoutheastCon, 2017

Introduction to sensor fusion

The main goal of multi-sensor fusion is to achieve better operation of the system using the collective information from all sensors. This is also referred to as the **synergistic effect**. Combining the data from a single sensor at different time intervals can also produce this effect.

In order to have better spatial and temporal coverage multiple sensors can be used.

Also, with multiple sensors there is increased estimation accuracy and fault-tolerance.

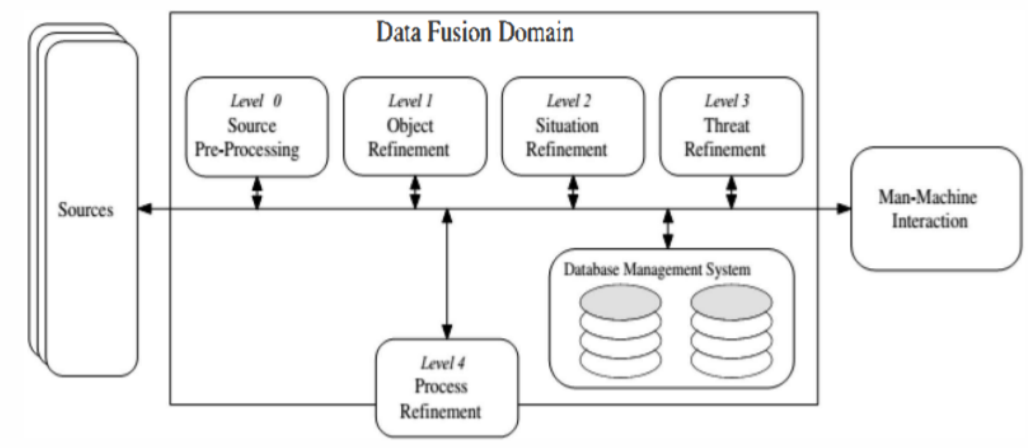
Chandrasekaran et al., SoutheastCon, 2017

Sensor fusion categories

A. Complementary
In this method, each sensor provides data about different aspects or attributes of the environment. By combining the data from each of the sensors we can arrive at a more global view of the environment or situation. Since there is no dependency between the sensors combining the data is relatively easy.
B. Competitive
In this method, several sensors measure the same or similar attributes. The data from several sensors is used to determine the overall value for the attribute under measurement. The measurements are taken independently and can also include measurements at different time instants for a single sensor. This method is useful in fault tolerant architectures to provide increased reliability of the measurement.
C. Co-operative
When the data from two or more independent sensors in the system is required to derive information, then co-operative sensor networks are used since a sensor individually cannot give the required information regarding the environment. A common example is stereoscopic vision.

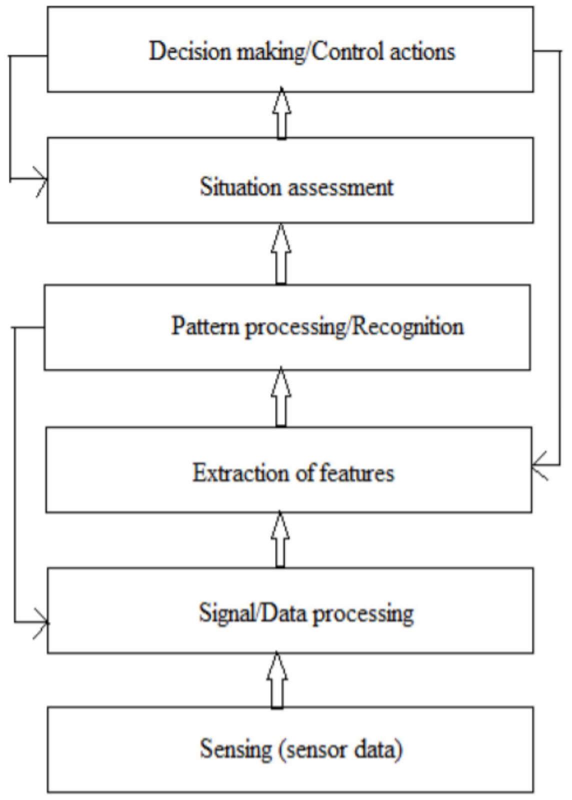
Chandrasekaran et al., SoutheastCon, 2017

Multi-sensor fusion models architectures



JDL (US Joint Directors of Laboratories, 1985)

Fusion levels:
- *signal level*
- decision level
- symbol level



mWFFM (modified Waterfall Fusion Model, 2009)

Chandrasekaran et al., SoutheastCon, 2017

Signal level fusion

In signal level fusion, data from multiple sources (sensors) are combined to obtain better quality data and higher understanding of the environment being observed. Signal level fusion often has either or both of the following goals:

• *Obtain a higher quality version of the input signals* i.e., higher signal-to-noise ratio. Sensor measurements from several sensors which have same physical properties are combined to determine the parameter being measured, more accurately. This minimizes and sometimes eliminates any uncertainty or inaccurate predictions caused by measurements from faulty sensors, measurement noise and state noise. For instance, readings from multiple temperature sensors in close proximity in a given space can be used for this kind of fusion.

• *Obtain a feature or mid-level information about the system that a single measuring node cannot reveal*. A feature is the first stage in understanding the state of the environment that helps the system in formulating a decision. Heterogeneous sensors are often employed for this process. For instance, signals from radar and images from cameras are used in target recognition.

Chandrasekaran et al., SoutheastCon, 2017

Signal level fusion: example in gait analysis

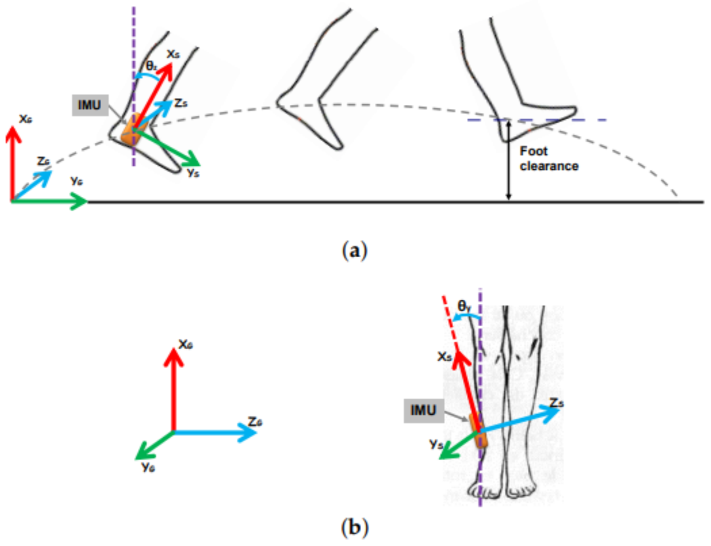


Figure 1. IMU placement and frame transformation in (**a**) the sagittal view and (**b**) the frontal view.

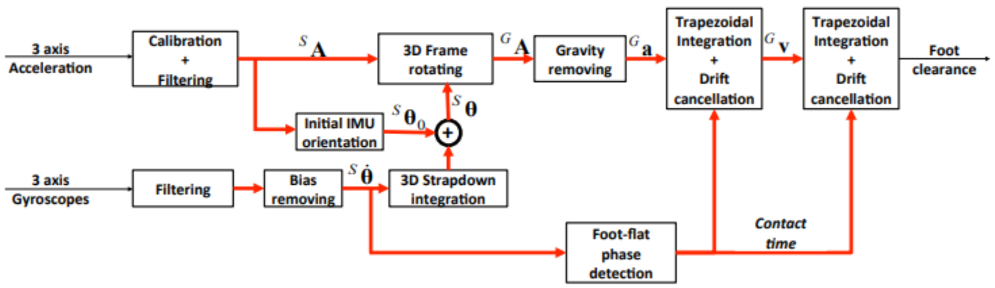
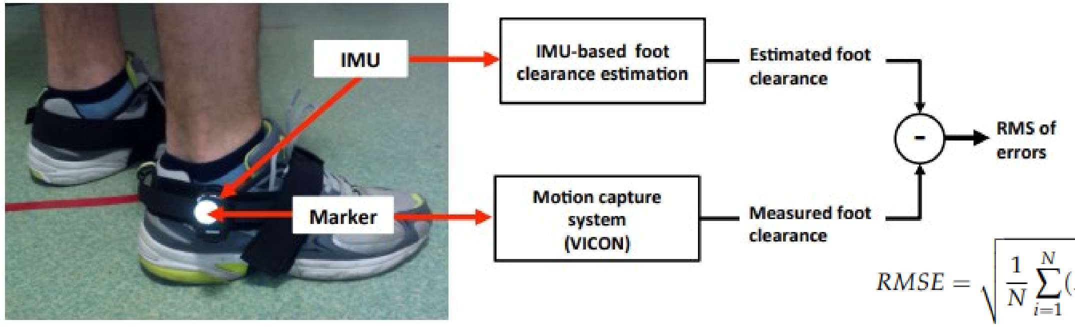


Figure 2. IMU-based foot clearance estimation algorithm.

They used the **Zero-Velocity Update** (ZVU) principle to limit the increase in integration drift between two successive strides. Then, they applied the above-mentioned acceleration correction method to each stride. For displacement estimation purposes (particularly foot clearance estimation), they adapted the same principle to vertical velocity integration and correction by generalizing to a **zero-displacement update** (ZDU) principle, where walking is assumed to occur on flat and horizontal ground.

Benoussaad et al., Sensors, 2016

Signal level fusion: example in gait analysis



$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(X_m - X_e)^2}$$

Figure 3. Experimental setup and validation of IMU-based foot clearance estimation.

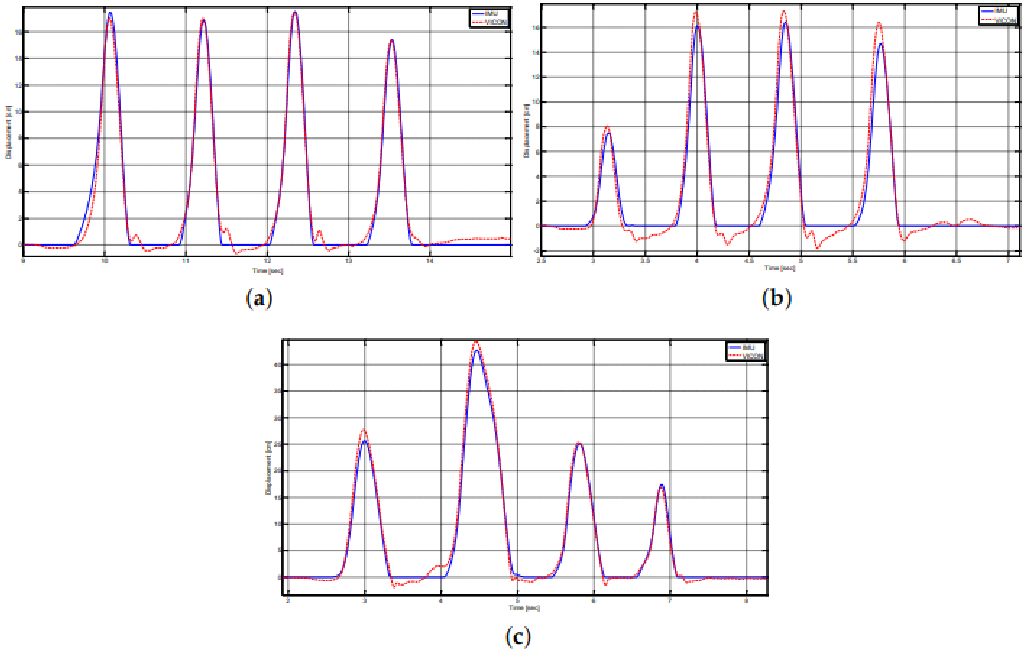
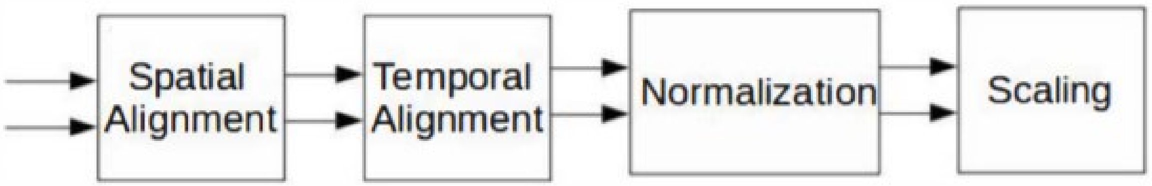


Figure 4. Foot clearance values measured by the VICON system and estimated by the IMU during: (**a**) Normal walking (Subject 1, left foot); (**b**) Fast walking (Subject 1, right foot); and (**c**) Walking with obstacles (Subject 10, right foot).

Benoussaad et al., Sensors, 2016

Signal level fusion

For sensor data to undergo signal level fusion, it is essential to condition the signals in the signal preprocessing phase. The signals must be in a common representation format. The stages involved in this process, as shown in Figure, include but not limited to: Signal alignment, normalization and scaling.



There are several methods by which signal level fusion can be achieved. The choice of method depends on various factors like the scenario and type of application, type of data or signal, relationship between the data or the state representation of the system.

Examples:

- ***Weighted averaging;***
- ***Neural networks;***
- ***Kalman filtering***

Chandrasekaran et al., SoutheastCon, 2017

Weighted averaging

Signal fusion can be achieved by taking an average of the various sensor signals measuring a particular parameter of the environment. If signals from some sensors can be *trusted* more than others, a higher weight is assigned to that sensor to increase its contribution towards the fused signal. The confidence level is a function of variance of the sensor signal.

$$x_{fused} = \sum_{i=0}^{n} w_i x_i$$

where $w_i = f(variance)$

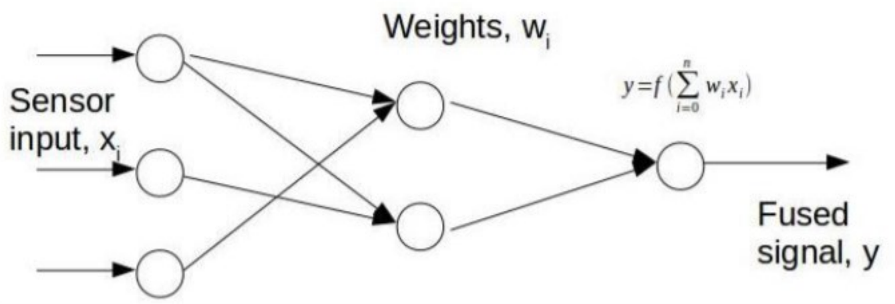
Chandrasekaran et al., SoutheastCon, 2017

Neural networks

An artificial neural network consists of interconnection of processing nodes called neurons. There is a pattern of interconnection between the neuronal layers that are weighted and the learning process that updates these weights. Data fusion models can be established using neural networks such that neurons and interconnecting weights are assigned based on the relationship between the multi-sensor data input and the signal output. The neural networks can be multilayer feed-forward or recurrent type. The fused output is a combination of input signal and corresponding weights calculated by the equation

$$y = f\left(\sum_{i=0}^{n} w_i x_i\right)$$

where w_i is the weight; x_i is the sensor data.



Example:
3-layer neural network with non-linear mapping

Chandrasekaran et al., SoutheastCon, 2017

Kalman filter

- The Kalman filter is a common adaptive method of sensor fusion to remove redundancy in the system and to predict the state of the system.
- The (linear) Kalman filter (KF) is an **estimator of the state of a linear dynamic system** starting from the measurement of the output in a stochastic environment or in the presence of disturbances (noise) on the state and on the output described by stochastic processes.
- It is an efficient **recursive filter**.

Applications

- The Kalman filter was used to estimate the position and speed of the Apollo 11 spacecraft in the first lunar mission as well as in subsequent missions to Mercury, Venus and Mars
- Navigation
- Control systems (e.g., autonomous driving, automatic pilots, attitude control, chemical and industrial processes)
- Fault diagnosis
- Signal and image processing
- Telecommunications
- …

Kalman filter theory

- We consider the application to a generic **linear and stationary dynamic system** with multiple inputs and multiple outputs (MIMO).
- It is assumed that the system is subject to process (state) noise $w(k)$ and measurement noise $v(k)$, uncorrelated over time, both Gaussians with zero mean and covariance matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$, respectively:

$$p(w) \propto N(0, \boldsymbol{Q})$$
$$p(v) \propto N(0, \boldsymbol{R})$$

- The state of the system is modeled as a Gaussian random variable $x(k)$ such that:

$x(0) = x_0$ - initial state

$E[x(0)] = \overline{x_0}$

$E[(x_0 - \overline{x_0})(x_0 - \overline{x_0})^T] = \boldsymbol{P}$

$\boldsymbol{P}(0) = \boldsymbol{P_0}$ - covariance of the initial state estimation error

Kalman filter theory: general equations

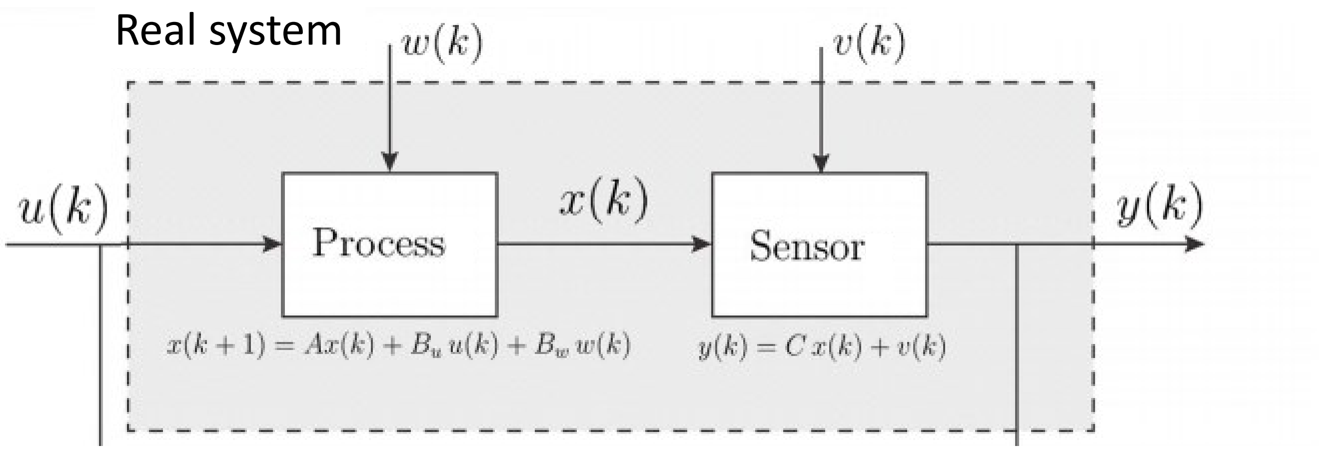
1. Equations of the model

$$x(k+1) = \boldsymbol{A}x(k) + \boldsymbol{B}u(k) + w(k)$$ *State equation*

$$y(k) = \boldsymbol{C}x(k) + v(k)$$ *Output equation*

Hp: $p(w) \propto N(0, \boldsymbol{Q})$

$p(v) \propto N(0, \boldsymbol{R})$



Kalman filter theory: general equations

1. Equations of the model

$$x(k+1) = \boldsymbol{A}x(k) + \boldsymbol{B}u(k) + w(k)$$ *State equation*

$$y(k) = \boldsymbol{C}x(k) + v(k)$$ *Output equation*

Hp: $p(w) \propto N(0, \boldsymbol{Q})$

$p(v) \propto N(0, \boldsymbol{R})$

2. Filter initialization

$x(0) = x_0$ *Initial state*

$\boldsymbol{P}(0) = \boldsymbol{P_0}$ *Covariance of the initial estimation error*

$$e(k) = x(k) - \hat{x}(k)$$

$$E[e(k)e^T(k)] = \boldsymbol{P}(k)$$

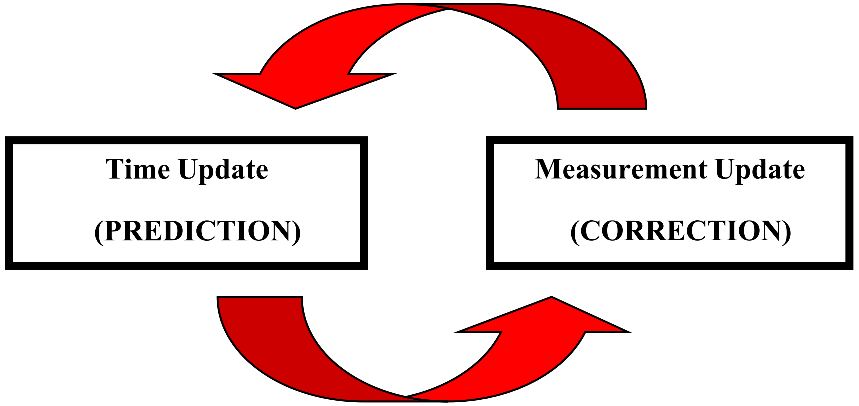
3. Noise covariance matrices

$$E[w(k)w^T(k-\tau)] = \boldsymbol{Q}(k)\delta(k-\tau)$$ *Covariance of the process noise*

$$E[v(k)v^T(k-\tau)] = \boldsymbol{R}(k)\delta(k-\tau)$$ *Covariance of the measurement noise*

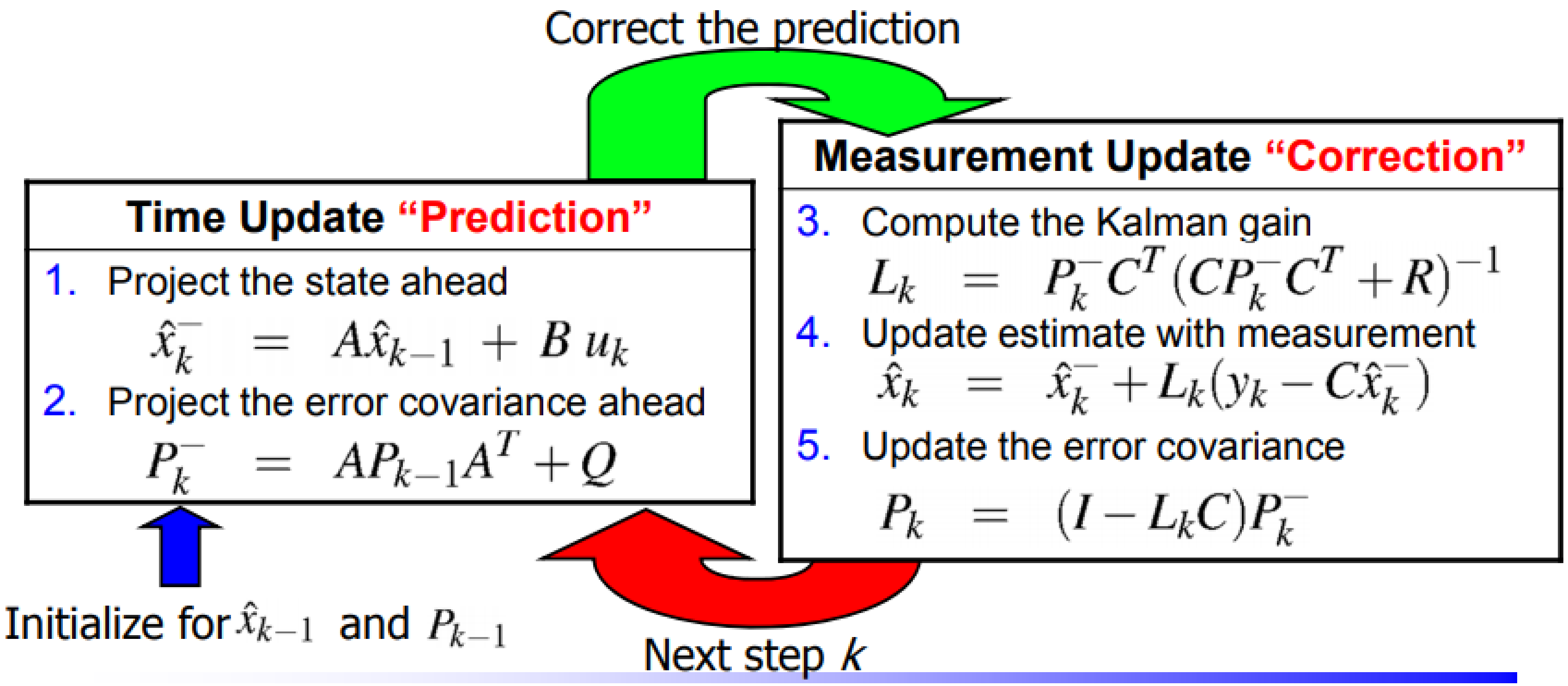
Kalman filter theory: general equations

- The KF provides an estimate of a process using a feedback control: at a certain instant of time the filter estimates the state and receives a feedback in the form of a measurement affected by noise;
- The KF equations fall into two groups: **time update** equations and **measurement update** equations.



- The former are responsible for forecasting the future state and the covariance of the filtering error as a function of the current state. The latter are responsible for the feedback: they are used to correct the estimate of the future state and the covariance of the filtering error through a new measurement.

Kalman filter theory: general equations

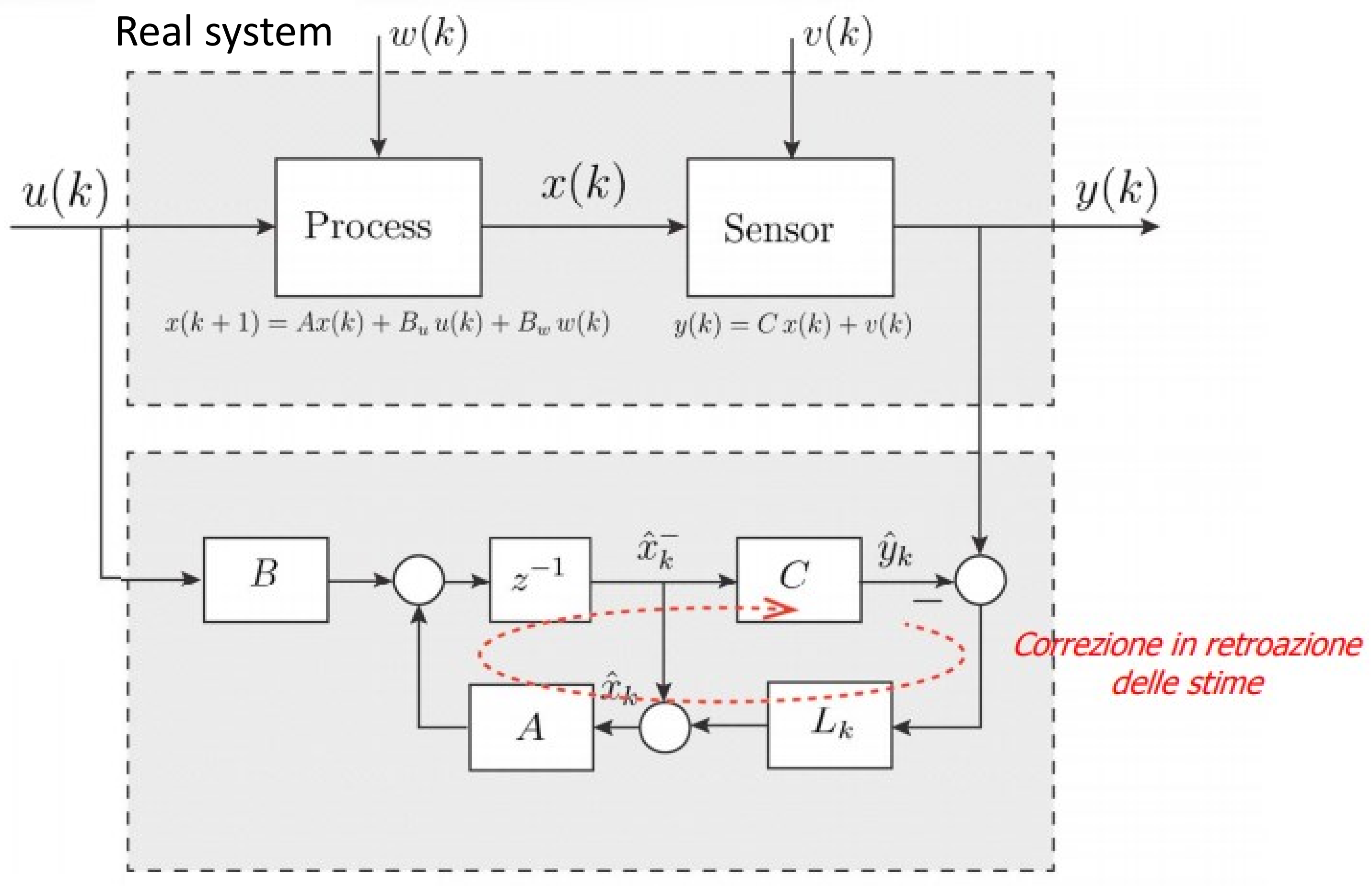


In the update or correction phase, the estimate from the predict phase is updated with the observation. If there are two sensors and both sending data simultaneously, then $y = [y_1, y_2]$. If the sensors are sending data one after the other, then the reading from first sensor can be used as a priori information before observation from second sensor is used to update the prediction.

Kalman filter theory: general equations ➔ pseudocode

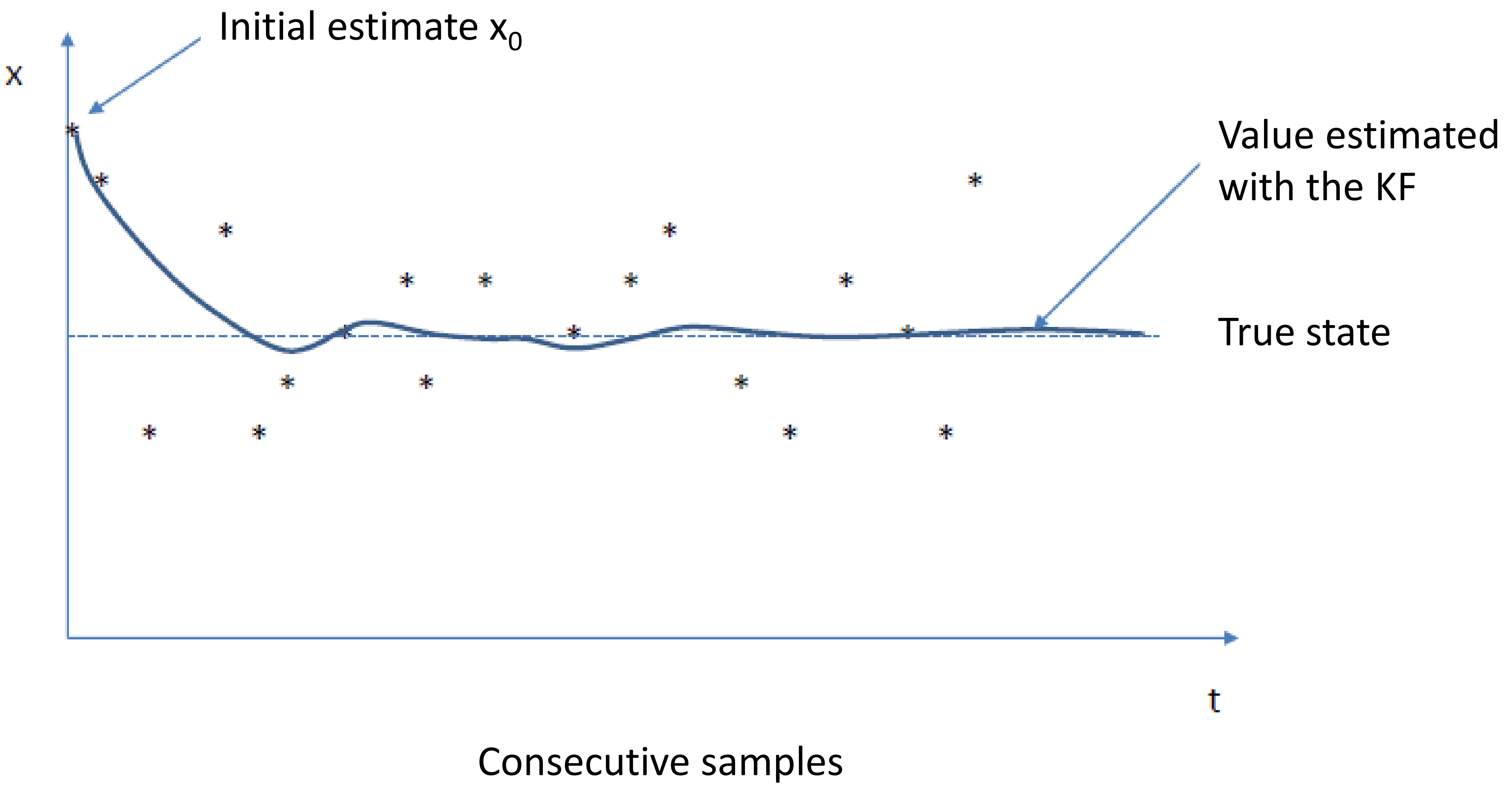
Time Update (Prediction)	Measurement Update (Correction)
1- Future state forecast	**1- Compute the Kalman gain L_K**
x(:,k) = A*x(:,k-1)	LK = P*C'*pinv(C*P*C'+R);
2. Compute the output	**2- Update estimate with real measurement y(k)**
ykalman(:,k) = C*x(:,k);	x(:,k)=x(:,k)+ LK * (yreal(:,k)-ykalman(:,k));
3- Predict the covariance of the initial state estimation error	**3- Update the error covariance**
P = A*P*A' + Q;	P = (eye(length(A))-LK*C)*P

Structure of the KF



$$L_k = P_k^- C^T (C P_k^- C^T + R)^{-1}$$

Example



Initial estimate x_0
x
Value estimated with the KF
True state
t
Consecutive samples

Extended Kalman filter (EKF) (Schmidt)

- If the process $x(k)$ to be estimated and/or the relationships between the measures and the process are not linear, the problem of state estimation must be faced with appropriate modifications with respect to the linear case;
- It is possible to linearize the estimate around the current estimate, using the partial derivatives of the measurement and process functions.

$$x(k+1) = f\big(x(k), u(k)\big) + w(k)$$ *State equation*

$$y(k) = h(x(k)) + v(k)$$ *Output equation*

- The noise considerations still apply:

$$E[w(k)w^T(k-\tau)] = \boldsymbol{Q}(k)\delta(k-\tau)$$ *Covariance of the process noise*

$$E[v(k)v^T(k-\tau)] = \boldsymbol{R}(k)\delta(k-\tau)$$ *Covariance of the measurement noise*

Extended Kalman filter (EKF) (Schmidt)

- From the linear case:

$$x(k+1) = \boldsymbol{A}x(k) + \boldsymbol{B}u(k) + w(k)$$ *State equation*

$$y(k) = \boldsymbol{C}x(k) + v(k)$$ *Output equation*

- To the non-linear case:

$$x(k+1) = f\big(x(k), u(k)\big) + w(k)$$ *State equation*

$$y(k) = h(x(k)) + v(k)$$ *Output equation*

$$\boldsymbol{A}(k) = \left.\frac{\delta f(x)}{\delta x}\right|_{x=x(k|k)}$$ *State matrix*

$$\boldsymbol{C}(k) = \left.\frac{\delta h(x)}{\delta x}\right|_{x=x(k|k-1)}$$ *Output matrix*

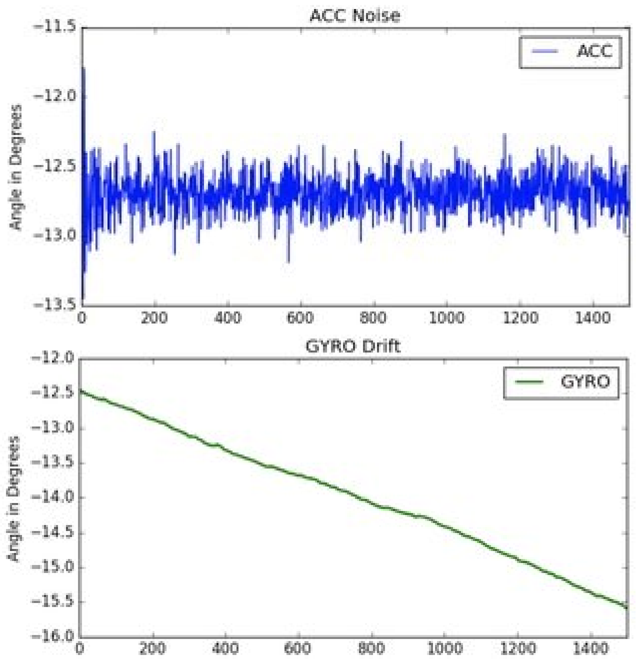
Sensor fusion for inertial-tracking

A calibrated IMU measures 3D angular velocity, 3D acceleration and gravity with respect to the reference frame integral with the sensor. Expressed in a non-rotating reference frame, the double integration of the acceleration provides the change of position, the integration of the angular velocity provides the angle of rotation. The accurate measurement of **orientation** plays a critical role in a range of fields, including motor rehabilitation.

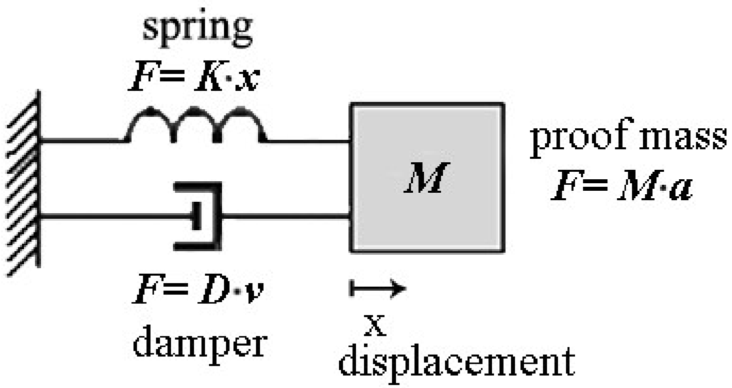
PROBLEM: the signals of accelerometric and gyroscopic sensors contain errors that make position and orientation estimates difficult due to the **integration of the drifts.**

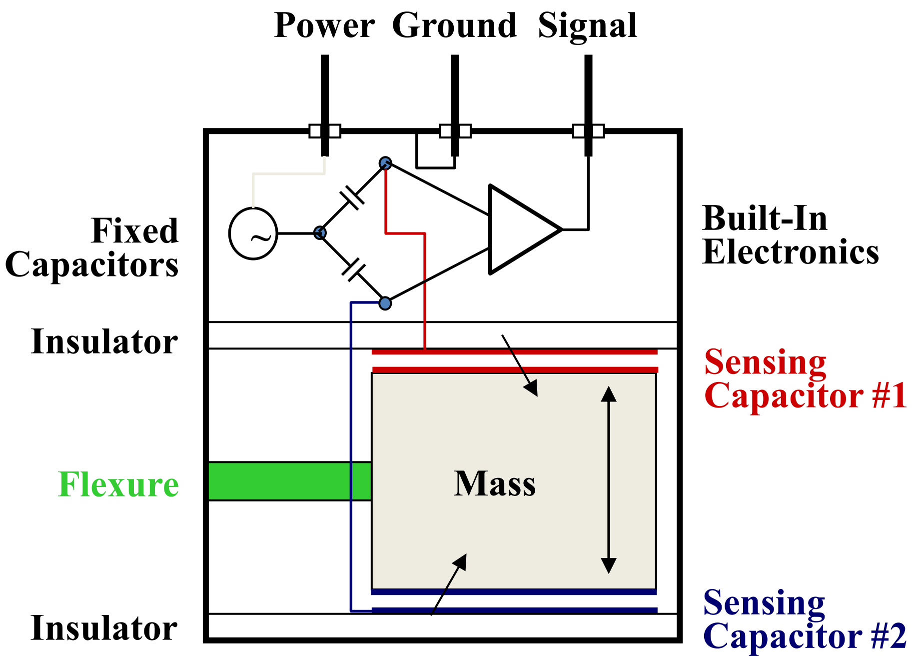
CAUSES

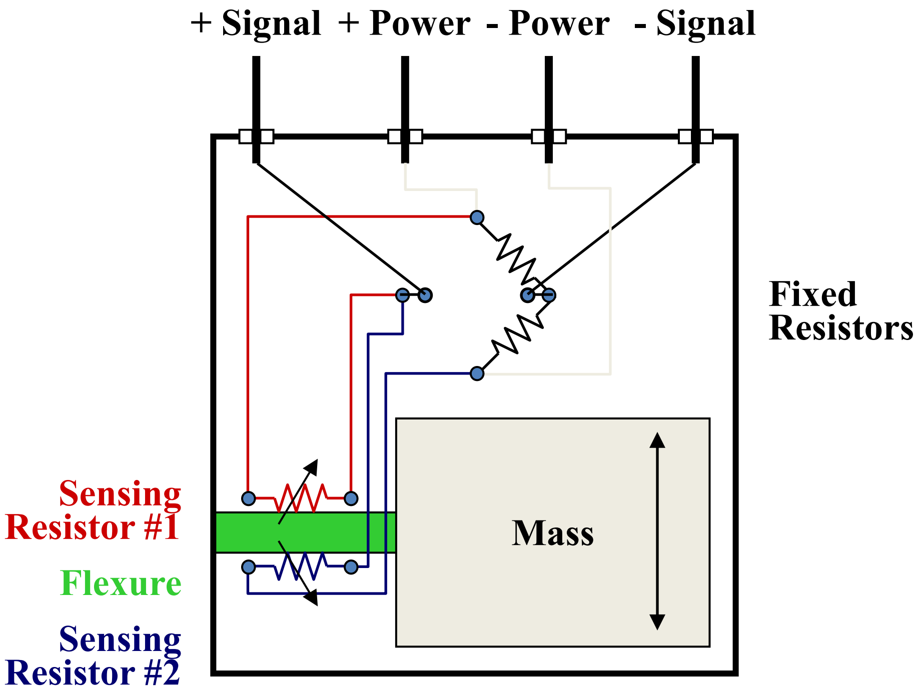
- Gyroscope offset and offset instability (temperature dependence);
- Accelerometer offset and offset instability;
- White noise;
- Calibration errors;
- Orientation error in gravity compensation;
- Double integration



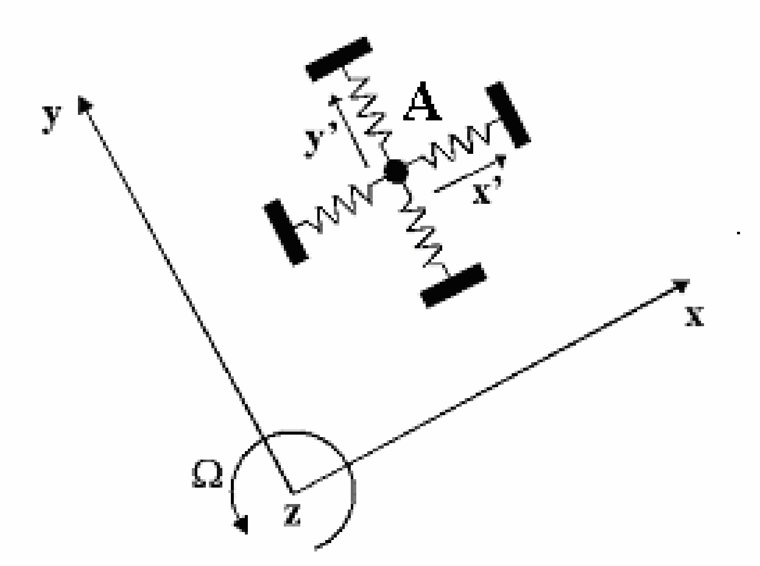
Accelerometer



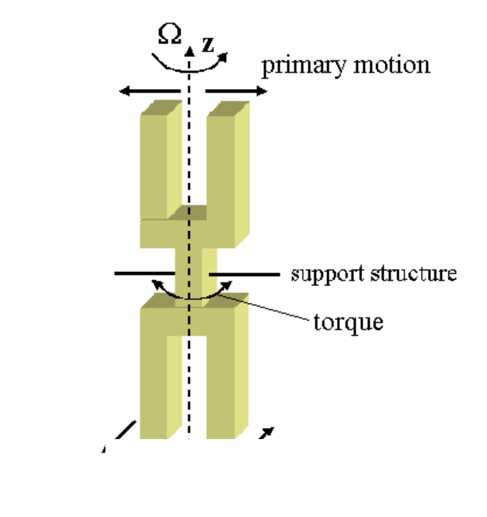


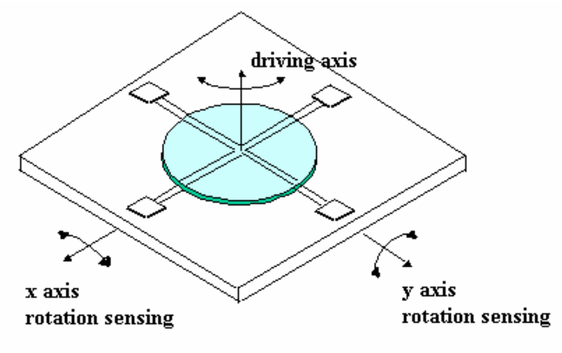


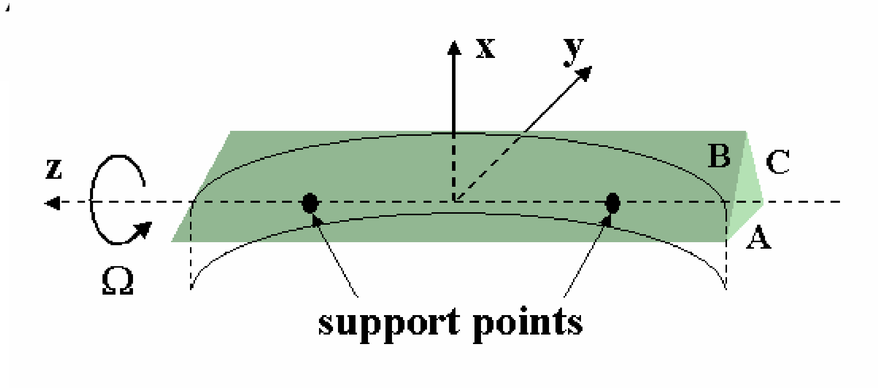
Gyroscope



$$\mathbf{F_c} = 2m\mathbf{v} \times \mathbf{\Omega}$$







Constant amplitude oscillation induced along x → **"Drive mode"**

Oscillation induced by Coriolis force along y → **"Sense mode"**

Sensor fusion for inertial-tracking

- The accurate measurement of **orientation** plays a critical role in a range of fields including: aerospace, robotics, navigation, and human motion analysis.
- Whilst a variety of technologies enable the direct measurement of orientation, inertial-based sensory systems have the advantage of being completely self-contained such that the measurement entity is constrained neither in motion nor to any specific environment or location.
- An IMU (Inertial Measurement Unit) consists of gyroscopes and accelerometers enabling the tracking of rotational and translational movements. In order to measure in three dimensions, tri-axis sensors consisting of 3 mutually orthogonal sensitive axes are required.
- A MIMU (Magneto-Inertial Measurement Unit) a.k.a. MARG (Magnetic, Angular Rate, and Gravity) or AHRS (Attitude and Heading Reference Systems) sensor is a hybrid IMU which incorporates a tri-axis magnetometer.
- An IMU alone can only measure an attitude relative to the direction of gravity which is sufficient for many applications.
- MARG systems are able to provide a complete measurement of orientation relative to the direction of gravity and the earth's magnetic field.

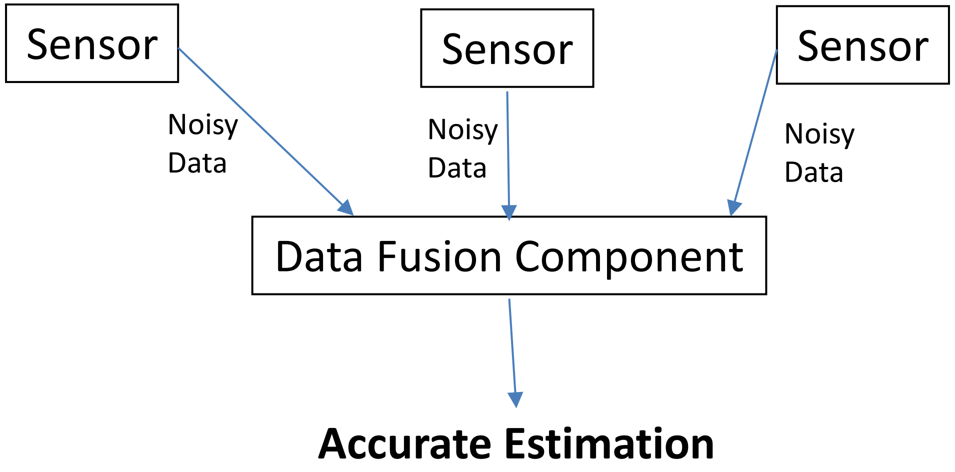
Sensor fusion for inertial-tracking

- A **gyroscope** measures angular velocity which, if initial conditions are known, may be integrated over time to compute the sensor's orientation. Precision gyroscopes are too expensive and bulky for most applications and so less accurate MEMS (Micro Electrical Mechanical System) devices are used in a majority of applications.
- The integration of gyroscope measurement errors will lead to an accumulating error in the calculated orientation. Therefore, *gyroscopes alone cannot provide an absolute measurement of orientation*.
- An **accelerometer** and **magnetometer** will measure the earth's gravitational and magnetic fields respectively and so provide an absolute reference of orientation. However, *they are likely to be subject to high levels of noise*; for example, accelerations due to motion will corrupt the measured direction of gravity.
- **The task of an orientation filter is to compute a single estimate of orientation through the optimal fusion of gyroscope, accelerometer and magnetometer measurements.**

Sensor fusion for inertial-tracking

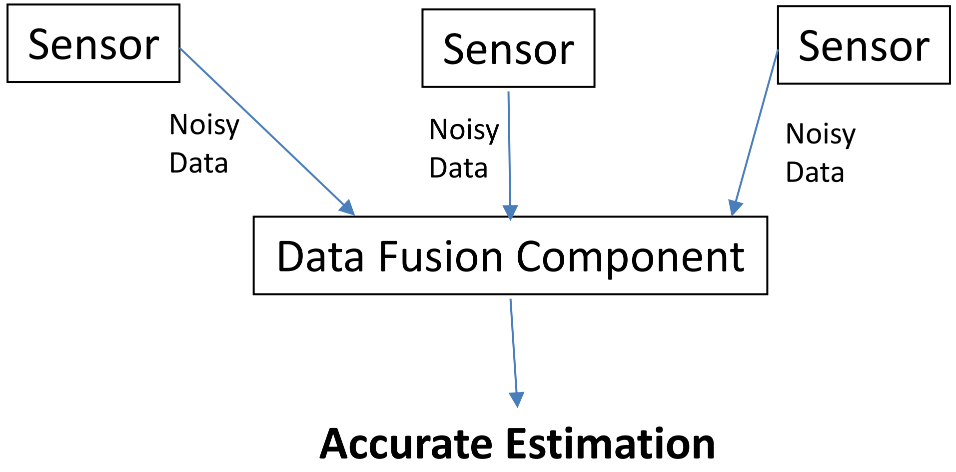
- Accelerometers and gyroscopes are used in many applications in the rehabilitation setting which include *monitoring of daily activities, evaluation of loads in ergonomic studies, virtual reality applications*.
- However, they have drifts in the position and orientation estimates that limit the long-term stable application of these sensors.
- The signals of these devices are therefore "combined" through sensor fusion algorithms to obtain the quantities of interest.

The Kalman filter is used to combine data from many indirect and noisy measurements. It weighs the information sources based on knowledge of the characteristics of the signal and the models that allow the best use of the data of each sensor.



Sensor fusion for inertial-tracking

- The Kalman filter (R. E. Kalman, A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82:35–45, 1960) has become the accepted basis for the majority of orientation filter algorithms and most commercial inertial orientation sensors (e.g., Xsens, Micro-strain, VectorNav, Intersense, PNI, APDM, and Crossbow) all produce systems founded on its use.
- **The Kalman filter is used to combine data from many indirect and noisy measurements. It weighs the information sources based on knowledge of the characteristics of the signal and the models that allow the best use of the data of each sensor.**

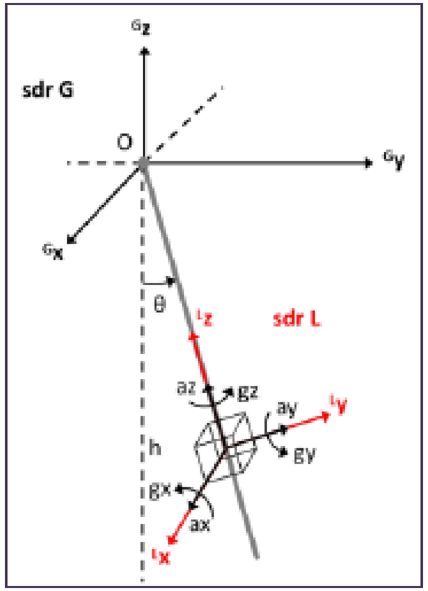


Example of application

2D KALMAN FILTER:
MECHANICAL MODEL-BASED APPLICATION

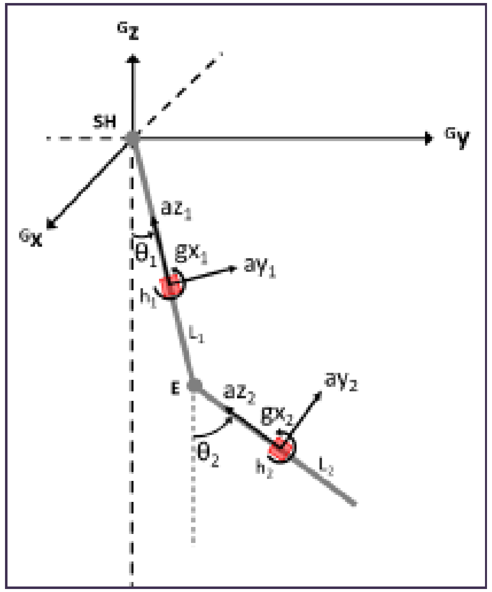
1-link model: theory

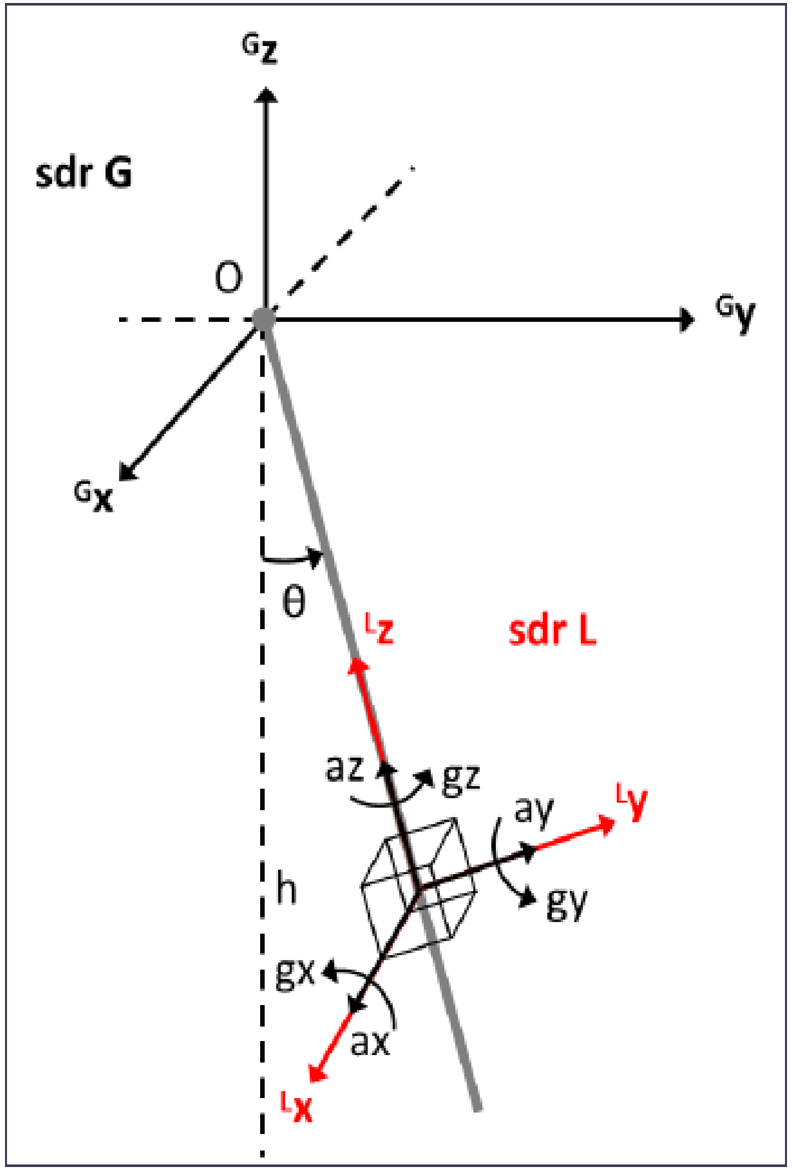
- Linear case
- Non-linear case



2-link model: application

- Non-linear case





1-link model

2D perspective: The oscillation occurs only in the sagittal plane yz, so around the x-axis.

The goal is to estimate the pendulum's kinematics, i.e., the trend of angle θ, through:

- the Kalman Filter (KF) in the linear case,
- the Extended Kalman Filter (EKF) in the non-linear case.

θ: the only degree of freedom of the system, defined with respect to the vertical line passing through the center of rotation and positive counterclockwise.

Suppose placing an IMU on the pendulum at a certain distance h from the pivot point O.

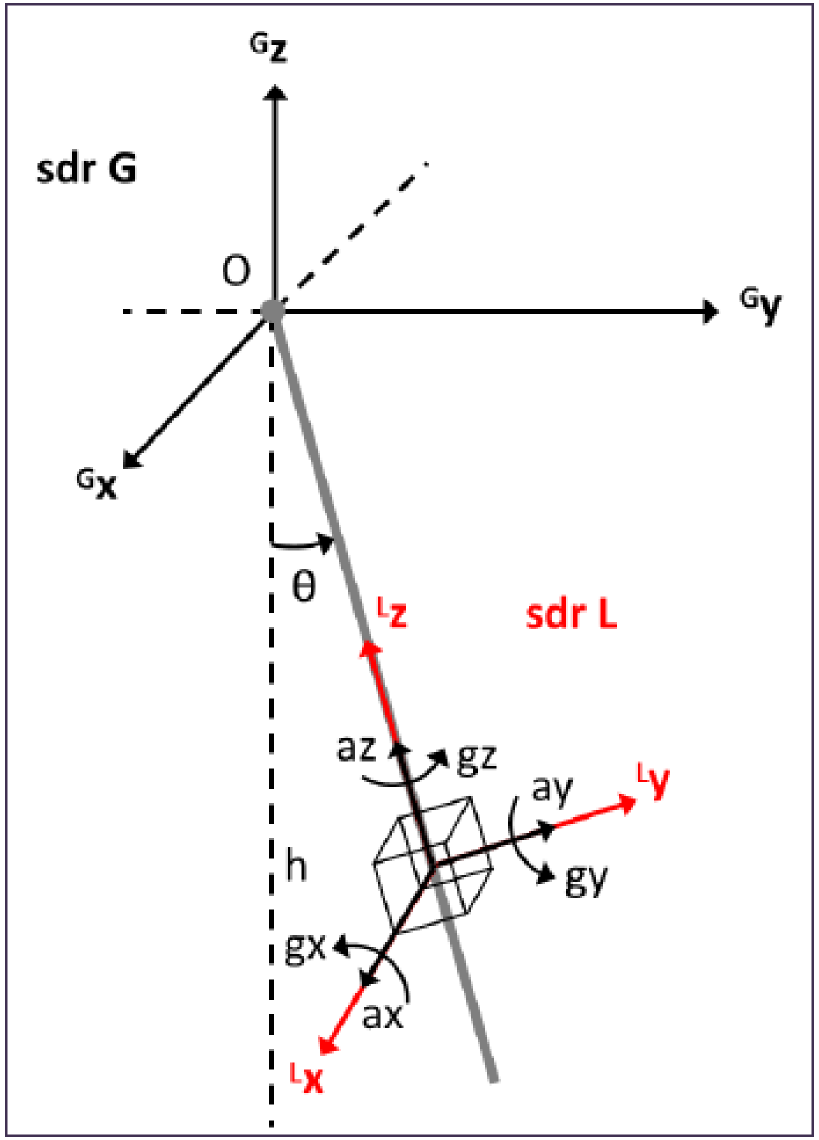
Rotation matrix G_{R_L}

Between the local reference system L, fixed with the IMU, and the global reference system G

$$^{G}\mathbf{R}_{\mathbf{L}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{bmatrix}$$

Cardano-Bryant angular convention:

$G_{R_L} = R_x * R_y * z$



ACCELEROMETERS output

$$\begin{bmatrix} ax \\ ay \\ az \end{bmatrix} = {}^G\mathbf{R_L}^{-1}\left({}^G\begin{bmatrix} \ddot{P}x \\ \ddot{P}y \\ \ddot{P}z \end{bmatrix} - {}^G\begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} \right) =$$

$$= \begin{bmatrix} 0 \\ h\ddot{\theta} + g\sin(\theta) \\ h\dot{\theta}^2 + g\cos(\theta) \end{bmatrix}$$

GYROSCOPES output

$$\begin{bmatrix} 0 & -gz & gy \\ gz & 0 & -gx \\ -gy & gx & 0 \end{bmatrix} = {}^G\mathbf{R_L}^{-1}\frac{d\,{}^G\mathbf{R_L}}{dt} \Rightarrow$$

$$\Rightarrow \begin{bmatrix} gx \\ gy \\ gz \end{bmatrix} = \begin{bmatrix} \dot{\theta} \\ 0 \\ 0 \end{bmatrix}$$

2D KALMAN FILTER: MECHANICAL MODEL-BASED APPLICATION

1-link
LINEAR CASE

THE LINEAR CASE ASSUMES THAT THE HYPOTHESIS OF SMALL OSCILLATION ANGLES MUST BE SATISFIED

STATE of the SYSTEM

$$\mathbf{x}(t) = \begin{bmatrix} \theta(t) \\ \omega(t) \\ \alpha(t) \end{bmatrix} = \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \\ \ddot{\theta}(t) \end{bmatrix}$$

State equations *continuous-time:*

$$\begin{cases} \dot{\theta}(t) = \omega(t) \\ \dot{\omega}(t) = \alpha(t) \\ \dot{\alpha}(t) = E(t) \end{cases}$$

State equations *discrete-time:*

$$\begin{cases} \theta_{K+1} = \theta_K + \omega_K T \\ \omega_{K+1} = \omega_K + \alpha_K T \\ \alpha_{K+1} = \alpha_K + E_K T \end{cases}$$

$$\mathbf{x}(k+1) = \mathbf{A}\,\mathbf{x}(k) + \mathbf{E}(k)$$

State Matrix: $\mathbf{A} = \begin{bmatrix} 1 & T & 0 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix}$

OUTPUT of the SYSTEM

Output equations *continuous-time:*

$$\begin{cases} y_1(t) = ay(t) + S(t) = h\ddot{\theta}(t) + g\theta(t) + S1(t) \\ y_2(t) = gx(t) + S(t) = \dot{\theta}(t) + S2(t) \end{cases}$$

Output equations *discrete-time:*

$$\begin{cases} y_{1,K} = h\alpha_K + g\theta_K + S_{1K} \\ y_{2,K} = \omega_K + S_{2K} \end{cases}$$

$$\mathbf{y}(k) = \mathbf{C}\,\mathbf{x}(k) + \mathbf{S}(k)$$

Output Matrix: $\mathbf{C} = \begin{bmatrix} g & 0 & h \\ 0 & 1 & 0 \end{bmatrix}$

Kalman Filter Algorithm

1) DEFINITION OF THE STATISTICAL VARIABLES FOR THE STATE ERROR E(K) AND THE MEASUREMENT ERROR S(K)

$$P(\mathbf{E}) \propto N(0, \mathbf{Q})$$

$$\mathbf{Q} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \varepsilon^2 \end{bmatrix}$$

The state error is white noise with zero mean and covariance **Q**, due to unknown disturbances and modeling errors; only the jerk is zero and noisy (with standard deviation of the noise equal to $\boldsymbol{\varepsilon}$)

$$P(\mathbf{S}) \propto N(0, \mathbf{R})$$

$$\mathbf{R} = \begin{bmatrix} \sigma_a^{\ 2} & 0 \\ 0 & \sigma_g^{\ 2} \end{bmatrix}$$

The measurement error is white noise with zero mean and covariance **R**, due to instrument noise and non-idealities in the measurement process; $\boldsymbol{\sigma}_a$ and $\boldsymbol{\sigma}_g$ are the standard deviations of the noises associated to the accelerometer and to the gyroscope

The **Q** and **R** variables constitute the design parameters of the KF, therefore they may need to undergo a potential tuning aimed at optimizing the filter.

2) FILTER INITIALIZATION

Initial state: $x_0 = X(0)$

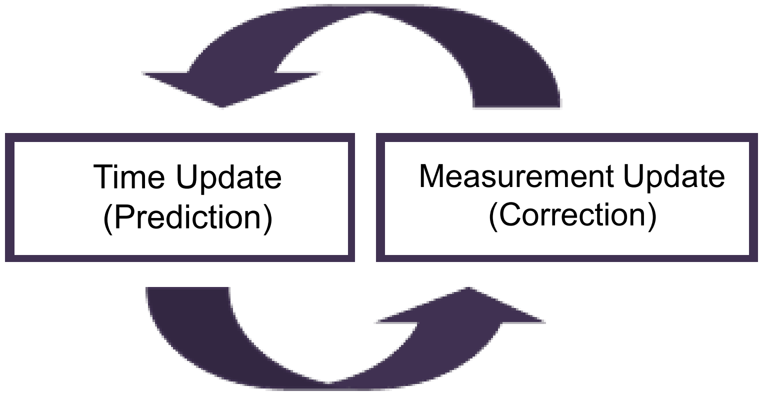
Initial covariance of the filtering error: $P_0 = P(0) = I$

3) DEFINITION OF THE MATRICES OF THE SYSTEM

State Matrix **A**

Output Matrix **C**

4) RECURSIVE CYCLE



1. Prediction of the future state

$$x(k+1) = \mathbf{A}\, x(k)$$

2. Prediction of covariance of the filtering error

$$\mathbf{P}(k+1) = \mathbf{A}\,\mathbf{P}(k)\,\mathbf{A}^T + \mathbf{Q}$$

3. Computation of the filter gain

$$\mathbf{K}(k+1) = \mathbf{P}(k+1)\,C^T\left(C\,\mathbf{P}(k+1)\,C^T + \mathbf{R}\right)^{-1}$$

4. Update of the state

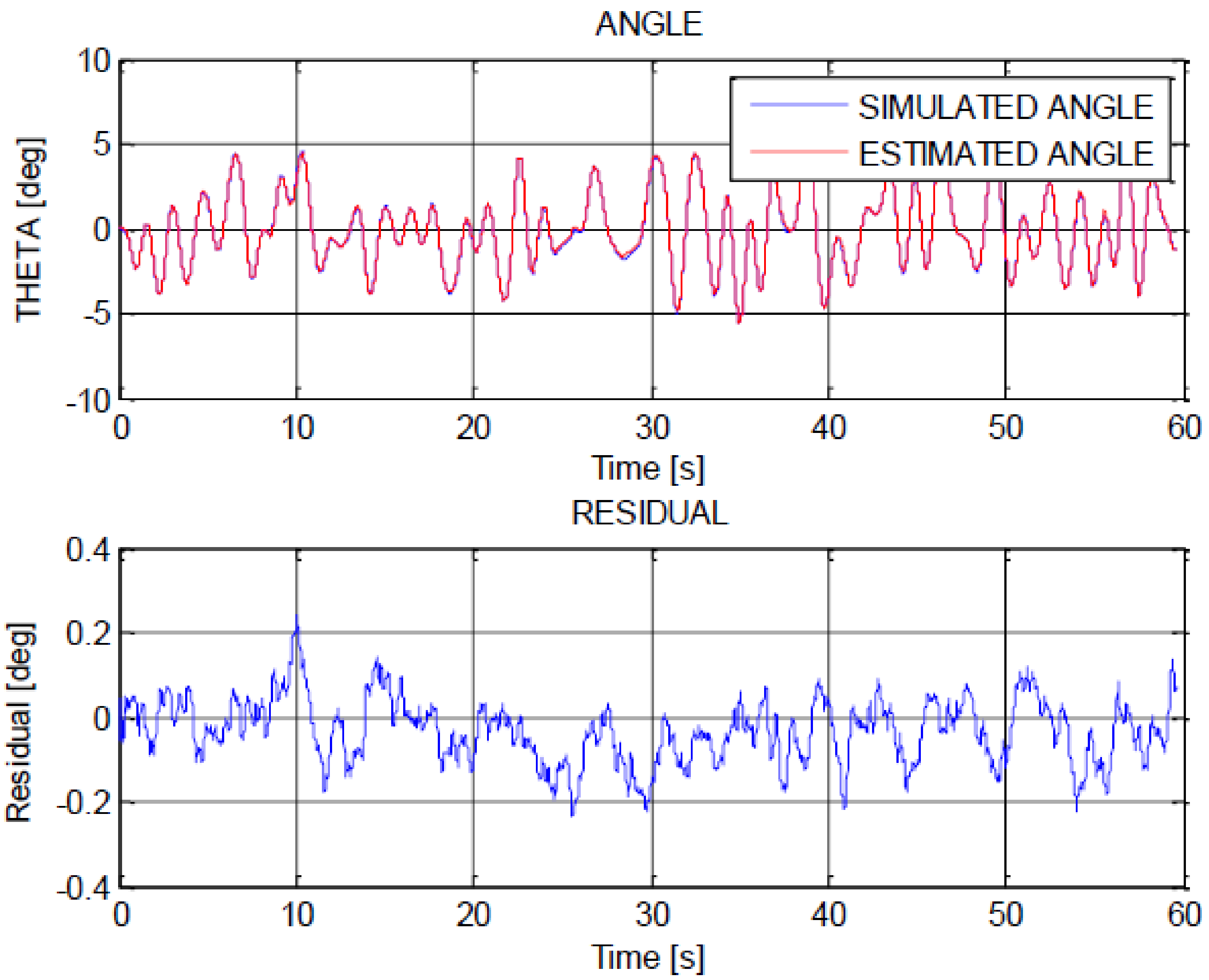
$$\mathbf{x}(k+1) = \mathbf{x}(k+1) + \mathbf{K}(k+1)\left[\mathbf{y}(k+1) - \mathbf{C}\,\mathbf{x}(k+1)\right]$$

5. Update of the covariance of the filtering error

$$\mathbf{P}(k+1) = \left(\mathbf{I} - \mathbf{K}(k+1)\,\mathbf{C}\right)\mathbf{P}(k+1)$$

2D KALMAN FILTER: MECHANICAL MODEL-BASED APPLICATION

1-link
LINEAR CASE



P-P = 12.35°
rmse = 0.08°

1-link NON-LINEAR CASE

THE NON-LINEAR CASE DOES NOT ASSUME THAT THE HYPOTHESIS OF SMALL OSCILLATION ANGLES MUST BE SATISFIED

STATE of the SYSTEM

$$\mathbf{x}(t) = \begin{bmatrix} \theta(t) \\ \omega(t) \\ \alpha(t) \end{bmatrix} = \begin{bmatrix} \theta(t) \\ \dot{\theta}(t) \\ \ddot{\theta}(t) \end{bmatrix}$$

State equations *continuous-time:*

$$\begin{cases} \dot{\theta}(t) = \omega(t) \\ \dot{\omega}(t) = \alpha(t) \\ \dot{\alpha}(t) = E(t) \end{cases}$$

State equations *discrete-time:*

$$\begin{cases} \theta_{K+1} = \theta_K + \omega_K T \\ \omega_{K+1} = \omega_K + \alpha_K T \\ \alpha_{K+1} = \alpha_K + E_K T \end{cases}$$

$$\mathbf{x}(k+1) = \mathbf{A}\,\mathbf{x}(k) + \mathbf{E}(k)$$

State Matrix:

$$\mathbf{A} = \begin{bmatrix} 1 & T & 0 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix}$$

OUTPUT of the SYSTEM

Output equations *continuous-time:*

$$\begin{cases} y_1(t) = ay(t) + S(t) = h\ddot{\theta}(t) + g\sin(\theta(t)) + S1(t) \\ y_2(t) = az(t) + S(t) = h\dot{\theta}^2(t) + g\cos(\theta(t)) + S2(t) \\ y_3(t) = gx(t) + S(t) = \dot{\theta}(t) + S3(t) \end{cases}$$

Output equations *discrete-time:*

$$\begin{cases} y_{1,K} = h\alpha_K + g\sin(\theta_K) + S_{1K} \\ y_{2,K} = h\omega_K^2 + g\cos(\theta_K) + S_{2K} \\ y_{3,K} = \omega_K + S_{3K} \end{cases}$$

$$\mathbf{y}(k) = c\,(\mathbf{x}(k)) + \mathbf{S}(k)$$

Output Matrix:

$$C_{i,j} = \frac{\partial c_i}{\partial x_j} = \begin{bmatrix} g\cos(\theta(k)) & 0 & h \\ -g\sin(\theta(k)) & 2h\omega(k) & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

Extended Kalman Filter Algorithm

1) DEFINITION OF THE STATISTICAL VARIABLES FOR THE STATE ERROR E(K) AND THE MEASUREMENT ERROR S(K)

$$P(\mathbf{E}) \propto N(0, \mathbf{Q})$$

$$\mathbf{Q} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \varepsilon^2 \end{bmatrix}$$

Covariance matrix of the state error

$$P(\mathbf{S}) \propto N(0, \mathbf{R})$$

$$\mathbf{R} = \begin{bmatrix} \sigma_a{}^2 & 0 & 0 \\ 0 & \sigma_a{}^2 & 0 \\ 0 & 0 & \sigma_g{}^2 \end{bmatrix}$$

Covariance matrix of the measurement error

2) FILTER INITIALIZATION

Initial state: $x_0 = X(0)$

Initial covariance of the filtering error: $P_0 = P(0) = I$

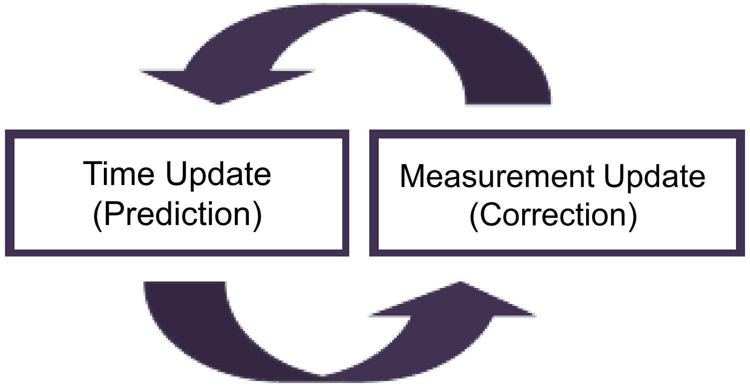
3) DEFINITION OF THE MATRICES OF THE SYSTEM

State Matrix **A**

2D KALMAN FILTER: MECHANICAL MODEL-BASED APPLICATION

1-link
NON-LINEAR CASE

4) RECURSIVE CYCLE



1. Prediction of the future state

$$\mathrm{x}(\mathrm{k}+1) = \mathbf{A}\,\mathrm{x}(\mathrm{k})$$

2. Output matrix

$$\mathbf{C}_{\mathrm{i,j}}(\mathrm{k}+1) = \frac{\partial c_{\mathrm{i}}}{\partial \mathrm{x}_{\mathrm{j}}(\mathrm{k}+1)}$$

3. Prediction of covariance of the filtering error

$$\mathrm{P}(\mathrm{k}+1) = \mathbf{A}\,\mathrm{P}(\mathrm{k})\,\mathbf{A}^{\mathrm{T}} + \mathbf{Q}$$

4. Computation of the filter gain

$$\mathbf{K}(\mathrm{k}+1) = \mathbf{P}(\mathrm{k}+1)\,\mathbf{C}^{\mathrm{T}}\left(\mathbf{C}\,\mathbf{P}(\mathrm{k}+1)\,\mathbf{C}^{\mathrm{T}} + \mathbf{R}\right)^{-1}$$

5. Update of the state

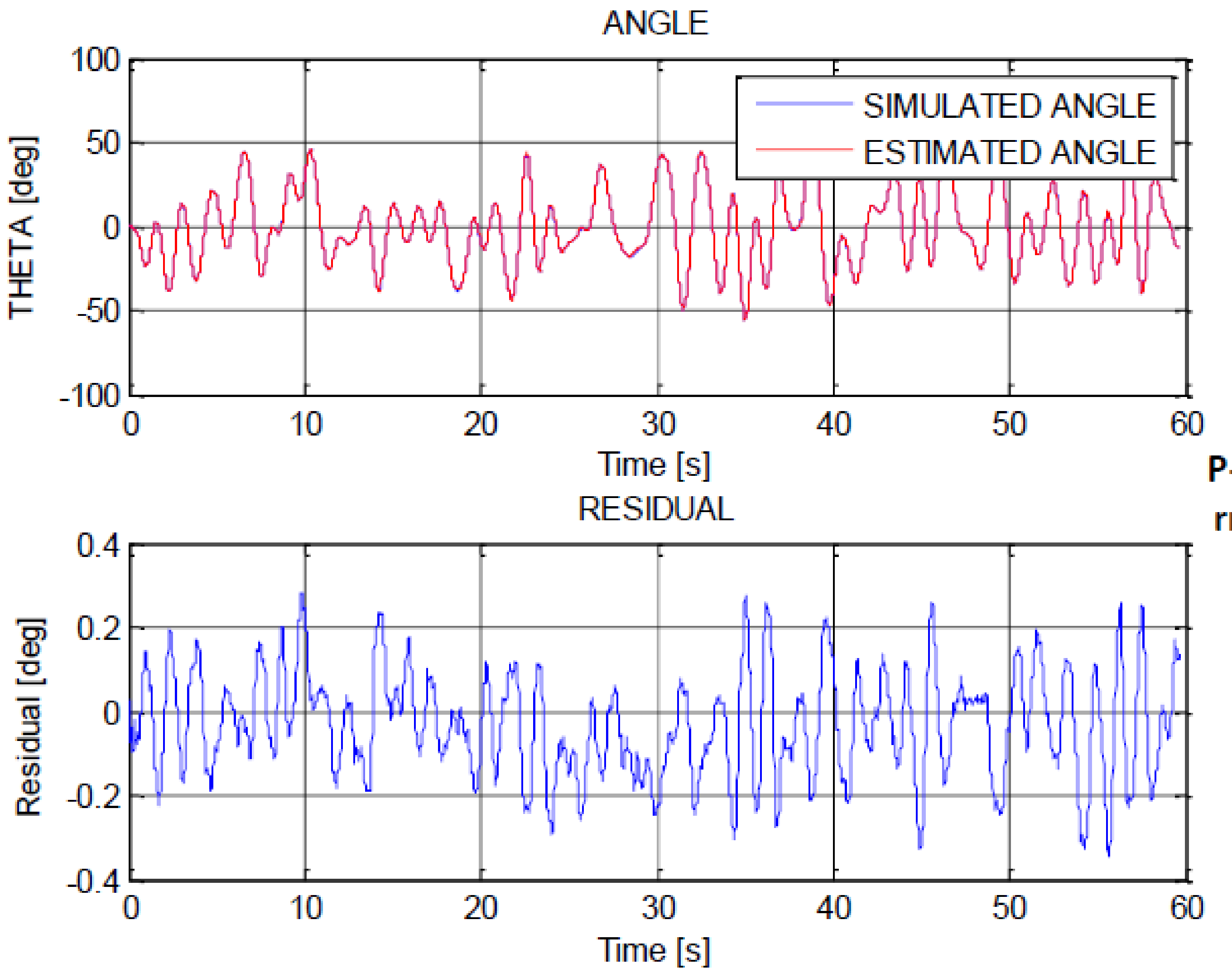
$$\mathbf{x}(\mathrm{k}+1) = \mathbf{x}(\mathrm{k}+1) + \mathbf{K}(\mathrm{k}+1)\left[\mathbf{y}(\mathrm{k}+1) - c(\mathbf{x}(\mathrm{k}+1))\right]$$

6. Update of the covariance of the filtering error

$$\mathbf{P}(\mathrm{k}+1) = \left(\mathbf{I} - \mathbf{K}(\mathrm{k}+1)\,\mathbf{C}\right)\mathbf{P}(\mathrm{k}+1)$$

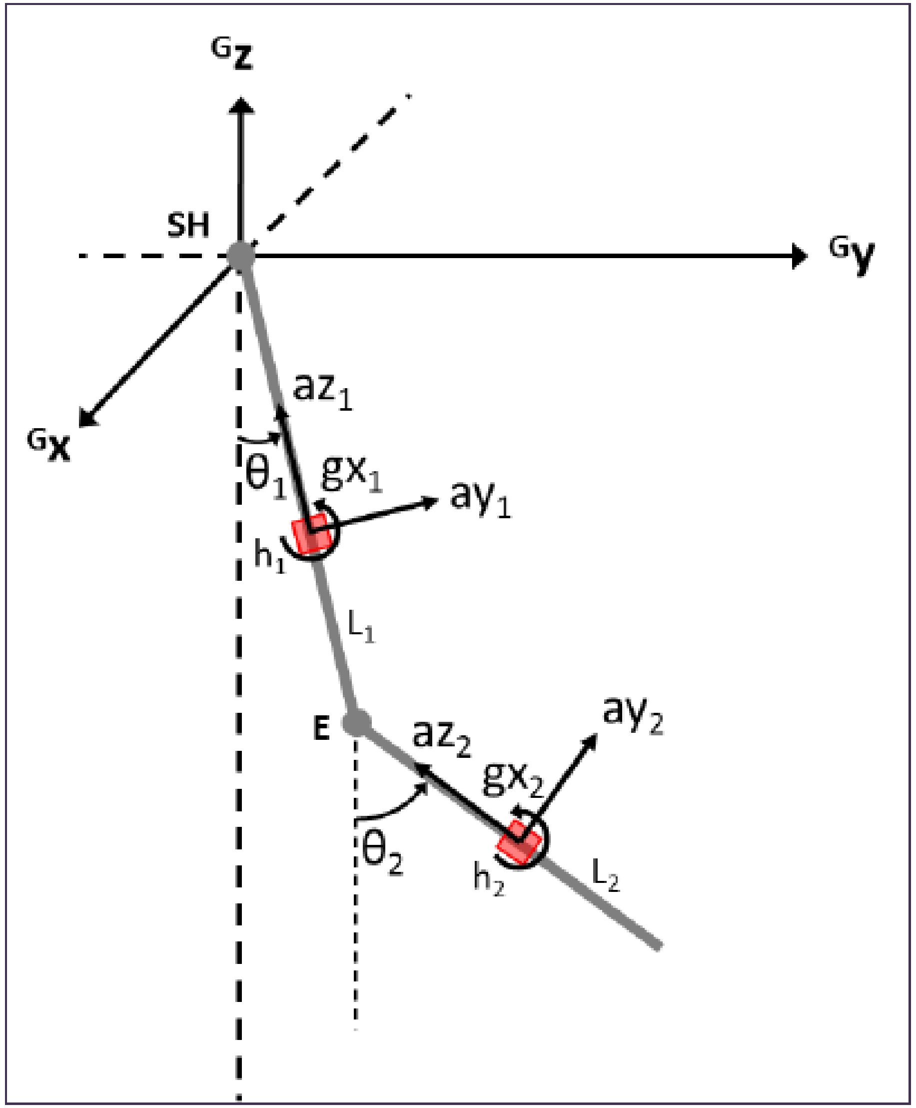
2D KALMAN FILTER: MECHANICAL MODEL-BASED APPLICATION

1-link
NON-LINEAR CASE



P-P = 123.49°
rmse = 0.13°

2D KALMAN FILTER: MECHANICAL MODEL-BASED APPLICATION



2-link model

Mechanical arm equipped with two IMUs from *Microstrain* and two high precision ENCODERS from *Gurley Precision Instruments*

2D perspective: The oscillation occurs only in the sagittal plane yz, so around the x-axis

The goal is to estimate the kinematics of the mechanical arm, i.e., the trends of the angles, through:

- the Extended Kalman Filter (EKF) -non-linear case-

θ_1, θ_2: degrees of freedom of the system, defined with respect to the vertical lines passing through the centers of rotation SH, E, and positive counterclockwise

The IMUs are placed on the pendulum at a certain distance h_1 and h_2 from the rotation centers SH and E

2D KALMAN FILTER: MECHANICAL MODEL-BASED APPLICATION

- ± 5g (accelerometer range),
- ± 300°/s (gyroscope range) ;
- A/D resolution 16 bit;
- RS232, RS422, USB 2.0 and wireless - 2.45 GHz, IEEE 802.15.4;
- 41 mm x 63 mm x 24 mm with enclosure,
- 32 mm x 36 mm x 15 mm without enclosure;
- 39 grams with enclosure,
  16 grams without enclosure.

Inertia-Link, Inertial Measurement Unit and Vertical Gyro, Microstrain

Virtual Absolute Encoder, Gurley Precision Instrument

- A/D resolution 18 bit

2D KALMAN FILTER: MECHANICAL MODEL-BASED APPLICATION

2-link
NON-LINEAR CASE

$$^{G}\mathbf{R_L}(\theta_i)=\begin{bmatrix}1 & 0 & 0\\ 0 & \cos(\theta_i) & -\sin(\theta_i)\\ 0 & \sin(\theta_i) & \cos(\theta_i)\end{bmatrix}$$

ACCELEROMETERS output

$$\begin{bmatrix}ax\\ ay\\ az\end{bmatrix}_i = {}^{G}\mathbf{R_L}^{-1}(\theta_i)\left({}^{G}\begin{bmatrix}\ddot{P}x\\ \ddot{P}y\\ \ddot{P}z\end{bmatrix}_i - {}^{G}\begin{bmatrix}0\\ 0\\ -g\end{bmatrix}\right) \Rightarrow \begin{cases}ax_i = 0\\ ay_i = h_i\ddot{\theta}_i + g\sin(\theta_i) + a_i^y\cos(\theta_i) - a_i^z\sin(\theta_i)\\ az_i = h_i\dot{\theta}_i^2 + g\cos(\theta_i) - a_i^y\sin(\theta_i) - a_i^z\cos(\theta_i)\end{cases}$$

Transport accelerations

$$a_i^y = \sum_{k=0}^{i-1} L_k \frac{d^2\sin(\theta_k)}{dt}$$

$$a_i^z = \sum_{k=0}^{i-1} L_k \frac{d^2\cos(\theta_k)}{dt}$$

GYROSCOPES output

$$\begin{bmatrix}0 & -gz & gy\\ gz & 0 & -gx\\ -gy & gx & 0\end{bmatrix}_i = {}^{G}\mathbf{R_L}^{-1}(\theta_i)\frac{d\,{}^{G}\mathbf{R_L}(\theta_i)}{dt} \Rightarrow \begin{cases}gx_i = \dot{\theta}_i\\ gy_i = 0\\ gz_i = 0\end{cases}$$

3-link
NON-LINEAR CASE

STATE of the SYSTEM

$$\mathbf{x}(t) = [\theta_1(t) \quad \omega_1(t) \quad \alpha_1(t) \quad \theta_2(t) \quad \omega_2(t) \quad \alpha_2(t)]'$$

$$\mathbf{x}(k+1) = \mathbf{A}\,\mathbf{x}(k) + \mathbf{E}(k)$$

The covariance matrix of the state error, Q, has size 6x6

OUTPUT of the SYSTEM

$$\mathbf{y}(t) = [ay_1(t) \quad az_1(t) \quad gx_1(t) \quad ay_2(t) \quad az_2(t) \quad gx_2(t)]'$$

$$\mathbf{y}(k) = c\,(\mathbf{x}(k)) + \mathbf{S}(k)$$

The covariance matrix of the measurement error, R, has size 6x6