III - Simulation

Representation Formalism

	Euler angles	Quaternions	Rotation Matrix
Position	$\mathbf{X} = [x \quad y \quad z]^T$		
Attitude	$\mathbf{\Phi}^{\text{eul}} = [\phi \quad \theta \quad \psi]^T$	$\mathbf{Q}$	$\mathbf{R}$
Velocities	$\mathbf{V}_{\boldsymbol{B}} = [u \quad v \quad w]^T, \boldsymbol{\omega}_B = [p \quad q \quad r]^T$		
State	$\boldsymbol{\chi} = \begin{bmatrix} \boldsymbol{\eta} = \begin{bmatrix} \mathbf{X} \\ \mathbf{\Phi}^{\text{eul}} \end{bmatrix} \\ \boldsymbol{\nu} = \begin{bmatrix} \mathbf{V_B} \\ \boldsymbol{\omega}_B \end{bmatrix} \end{bmatrix}$	$\boldsymbol{\chi} = \begin{bmatrix} \boldsymbol{\eta} = \begin{bmatrix} \mathbf{X} \\ \boldsymbol{Q} \end{bmatrix} \\ \boldsymbol{\nu} = \begin{bmatrix} \mathbf{V_B} \\ \boldsymbol{\omega}_B \end{bmatrix} \end{bmatrix}$	$\boldsymbol{\chi} = \{\mathbf{X}, \mathbf{R}, \mathbf{V}_{\boldsymbol{B}}, \boldsymbol{\omega}_B\}$
Kinematic Model $\dot{\boldsymbol{\eta}} = \mathbf{f}(\boldsymbol{\nu})$	$\begin{bmatrix} \dot{\mathbf{X}} \\ \dot{\mathbf{\Phi}}^{\text{eul}} \end{bmatrix} = \mathbf{R}_{\text{cin}}^{\text{eul}} \cdot \begin{bmatrix} \mathbf{V_B} \\ \boldsymbol{\omega}_B \end{bmatrix}$	$\begin{cases} [0, \dot{\mathbf{X}}^T]^T = \mathbf{Q} \otimes [0, \mathbf{V}_{\boldsymbol{B}}^T]^T \otimes \mathbf{Q}^* \\ \dot{\mathbf{Q}} = \frac{1}{2} \cdot \mathbf{Q} \otimes \underbrace{[0, \boldsymbol{\omega}_B^T]^T}_{\boldsymbol{\Omega}_B} \end{cases}$	$\begin{cases} \dot{\mathbf{X}} = \mathbf{R} \cdot \mathbf{V_B} \\ \dot{\mathbf{R}} = \mathbf{R} \cdot (\boldsymbol{\omega}_{\mathbf{B}} \wedge) \end{cases}$

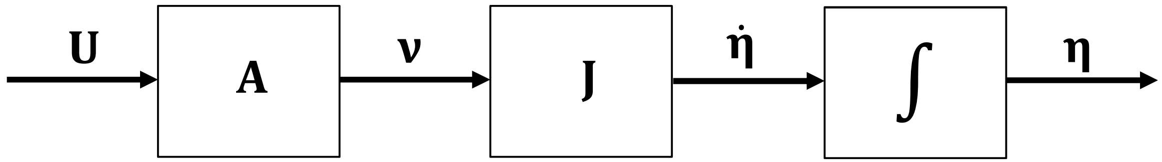
$$\mathbf{R}_{\text{cin}}^{\text{eul}} = \begin{bmatrix} \cos\psi \cdot \cos\theta & \cos\psi \cdot \sin\theta \cdot \sin\phi - \sin\psi \cdot \cos\phi & \cos\psi \cdot \sin\theta \cdot \cos\phi + \sin\psi \cdot \sin\phi & 0 & 0 & 0 \\ \sin\psi \cdot \cos\theta & \sin\psi \cdot \sin\theta \cdot \sin\phi + \cos\psi \cdot \cos\phi & \sin\psi \cdot \sin\theta \cdot \cos\phi - \cos\psi \cdot \sin\phi & 0 & 0 & 0 \\ -\sin\theta & \cos\theta \cdot \sin\phi & \cos\theta \cdot \cos\phi & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & \sin\phi \cdot \tan\theta & \cos\phi \cdot \tan\theta \\ 0 & 0 & 0 & 0 & \cos\phi & -\sin\phi \\ 0 & 0 & 0 & 0 & \frac{\sin\phi}{\cos\theta} & \frac{\cos\phi}{\cos\theta} \end{bmatrix}$$

Gimbal lock, if $\theta = \pm\frac{\pi}{2}$

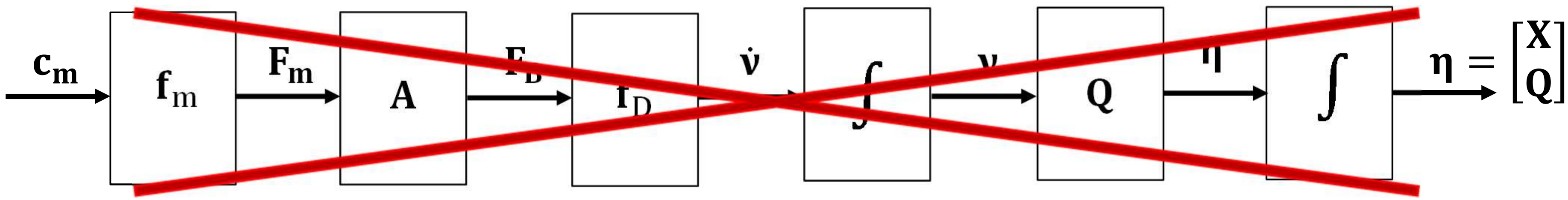
$$\boldsymbol{\omega}_B \wedge = \begin{pmatrix} 0 & -w_z & w_y \\ w_z & 0 & -w_x \\ -w_y & w_x & 0 \end{pmatrix}$$

Simulator

- Kinematics



- Dynamics

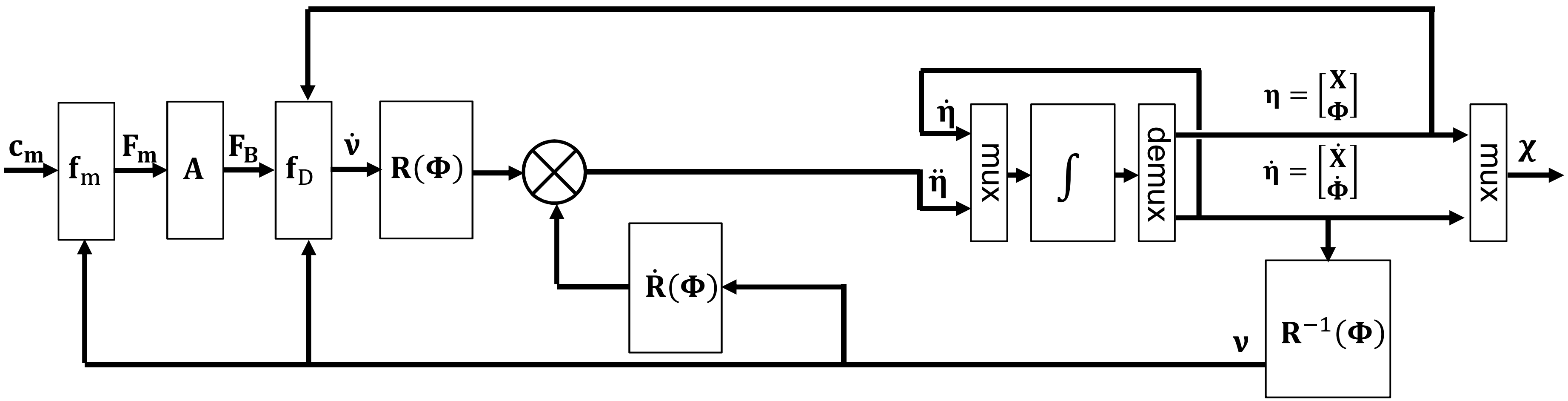


Simulator

- Dynamics (Euler formalism)

$$\boldsymbol{\eta} = \begin{bmatrix} \mathbf{X} \\ \boldsymbol{\Phi} \end{bmatrix} \overset{\frac{\mathrm{d}}{\mathrm{d}t}}{\Rightarrow} \dot{\boldsymbol{\eta}} = \begin{bmatrix} \dot{\mathbf{X}} \\ \dot{\boldsymbol{\Phi}} \end{bmatrix} = \mathbf{R}(\boldsymbol{\Phi}) \cdot \begin{bmatrix} \mathbf{V_B} \\ \boldsymbol{\omega}_B \end{bmatrix} \overset{\frac{\mathrm{d}}{\mathrm{d}t}}{\Rightarrow} \ddot{\boldsymbol{\eta}} = \begin{bmatrix} \ddot{\mathbf{X}} \\ \ddot{\boldsymbol{\Phi}} \end{bmatrix} = \mathbf{R}(\boldsymbol{\Phi}) \cdot \begin{bmatrix} \dot{\mathbf{V}}_\mathbf{B} \\ \dot{\boldsymbol{\omega}}_B \end{bmatrix} + \dot{\mathbf{R}}(\boldsymbol{\Phi}) \cdot \begin{bmatrix} \mathbf{V_B} \\ \boldsymbol{\omega}_B \end{bmatrix}$$

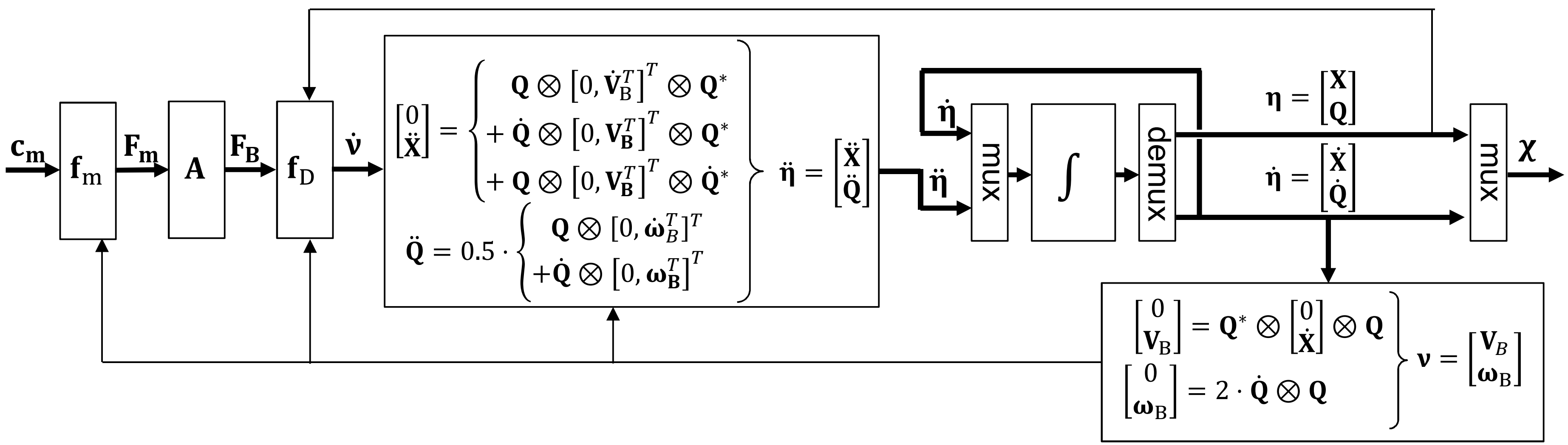
$$\boldsymbol{\nu} = \begin{bmatrix} \mathbf{V_B} \\ \boldsymbol{\omega}_B \end{bmatrix} = \mathbf{R}^{-1}(\boldsymbol{\Phi}) \cdot \begin{bmatrix} \dot{\mathbf{X}} \\ \dot{\boldsymbol{\Phi}} \end{bmatrix}$$



Simulator

- Dynamics (quaternion form.)

$$\boldsymbol{\eta} = \begin{bmatrix}\mathbf{X}\\ \mathbf{Q}\end{bmatrix} \overset{\frac{\mathrm{d}}{\mathrm{d}t}}{\Rightarrow} \dot{\boldsymbol{\eta}} = \begin{bmatrix}\dot{\mathbf{X}}\\ \dot{\mathbf{Q}}\end{bmatrix} : \begin{cases} \left[0, \dot{\mathbf{X}}^T\right]^T = \mathbf{Q} \otimes \left[0, \mathbf{V}_B^T\right]^T \otimes \mathbf{Q}^* \\ \dot{\mathbf{Q}} = \frac{1}{2} \cdot \mathbf{Q} \otimes [0, \boldsymbol{\omega}_B^T]^T \end{cases} \overset{\frac{\mathrm{d}}{\mathrm{d}t}}{\Rightarrow} \ddot{\boldsymbol{\eta}} = \begin{bmatrix}\ddot{\mathbf{X}}\\ \ddot{\mathbf{Q}}\end{bmatrix} : \begin{cases} \begin{bmatrix}0\\ \ddot{\mathbf{X}}\end{bmatrix} = \begin{cases} \mathbf{Q} \otimes \left[0, \dot{\mathbf{V}}_\mathrm{B}^T\right]^T \otimes \mathbf{Q}^* \\ + \dot{\mathbf{Q}} \otimes \left[0, \mathbf{V}_\mathrm{B}^T\right]^T \otimes \mathbf{Q}^* \\ + \mathbf{Q} \otimes \left[0, \mathbf{V}_\mathrm{B}^T\right]^T \otimes \dot{\mathbf{Q}}^* \end{cases} \\ \ddot{\mathbf{Q}} = 0.5 \cdot \begin{cases} \mathbf{Q} \otimes [0, \dot{\boldsymbol{\omega}}_B^T]^T \\ + \dot{\mathbf{Q}} \otimes \left[0, \boldsymbol{\omega}_\mathrm{B}^T\right]^T \end{cases} \end{cases}$$



Respect de la contrainte de normalisation ? → Utilisation des Contraintes de Lagrange

Lagrange Constraint

Let the following dynamical system with model : $\mathbf{F}_1 = \mathbf{M} \cdot \ddot{\mathbf{x}} + \mathbf{G}(\mathbf{x}, \dot{\mathbf{x}})$, where $\mathbf{x} = [x_1, \dots x_\mathrm{n}]^\mathrm{T}$ denotes its state vector. Considering $\mathbf{F} = \mathbf{F}_1 - \mathbf{G}(\mathbf{x}, \dot{\mathbf{x}})$, yields :

$$\mathbf{F} = \mathbf{M} \cdot \ddot{\mathbf{x}}$$

This system undergoes the following constraints:

$$\phi(\mathbf{x}) = 0$$

The constraint's derivation provides :

$$\underbrace{\left[\frac{\partial \phi(\mathbf{x})}{\partial x_1}, \dots, \frac{\partial \phi(\mathbf{x})}{\partial x_\mathrm{n}}\right]}_{\mathbf{A}} \cdot \dot{\mathbf{x}} = \mathbf{A} \cdot \dot{\mathbf{x}} = 0$$

$$\dot{\mathbf{A}} \cdot \dot{\mathbf{x}} + \mathbf{A} \cdot \ddot{\mathbf{x}} = 0$$

Hence the dynamics of the constrained system is, where $\mathbf{A}^\mathrm{T} \cdot \boldsymbol{\lambda}$ designs the forces of respect of the contraint:

$$\mathbf{F} + \mathbf{A}^\mathrm{T} \cdot \lambda = \mathbf{M} \cdot \ddot{\mathbf{x}}$$

The new constrained system is then written as:

$$\begin{cases} \mathbf{F} = \mathbf{M} \cdot \ddot{\mathbf{x}} - \mathbf{A}^\mathrm{T} \cdot \lambda \\ \dot{\mathbf{A}} \cdot \dot{\mathbf{x}} + \mathbf{A} \cdot \ddot{\mathbf{x}} = 0 \end{cases} \rightarrow \begin{bmatrix} \ddot{\mathbf{x}} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{M} & -\mathbf{A}^\mathrm{T} \\ \mathbf{A} & 0 \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{F} \\ -\dot{\mathbf{A}} \cdot \dot{\mathbf{x}} \end{bmatrix} \rightarrow \begin{bmatrix} \dot{\mathbf{x}} \\ \ddot{\mathbf{x}} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{M} & -\mathbf{A}^\mathrm{T} \\ \mathbf{0} & \mathbf{A} & 0 \end{bmatrix}^{-1} \cdot \begin{bmatrix} \dot{\mathbf{x}} \\ \mathbf{F} \\ -\dot{\mathbf{A}} \cdot \dot{\mathbf{x}} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{M} & -\mathbf{A}^\mathrm{T} \\ \dot{\mathbf{A}} & \mathbf{A} & \mathbf{0} \end{bmatrix}^{-1} \cdot \begin{bmatrix} \dot{\mathbf{x}} \\ \mathbf{F} \\ 0 \end{bmatrix}$$

Lagrange Constraint

Exemple : an object, that is assimilable to a ponctual mass m, is falling in the vertical plane of a terrestrial gravity field of magnitude g, with a linear viscous friction with coefficient f. Its initial position is denoted $\mathbf{X}(0) = [0,0]^{\mathrm{T}}$. This object is attached to a nonelastic rope of length l attached to the point $\mathbf{X}_{\mathrm{R}} = [l/2,0]^{\mathrm{T}}$.

Unconstrained dynamics : $\mathbf{F} = \mathbf{M} \cdot \ddot{\mathbf{X}} = \begin{bmatrix} 0 \\ -m \cdot g \end{bmatrix} - f \cdot \dot{\mathbf{X}}$, where $\mathbf{M} = \begin{bmatrix} m & 0 \\ 0 & m \end{bmatrix}$

$$\rightarrow \ddot{\mathbf{X}} = \mathbf{M}^{-1} \cdot \mathbf{F} \rightarrow \begin{bmatrix} \ddot{\mathbf{X}} \\ \dot{\mathbf{X}} \end{bmatrix} = \begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{\mathrm{d}} \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{F} \\ \dot{\mathbf{X}} \end{bmatrix}$$

Constraint : $\phi(\mathbf{X}) = (\mathbf{X} - \mathbf{X}_{\mathrm{R}})^{\mathrm{T}} \cdot (\mathbf{X} - \mathbf{X}_{\mathrm{R}}) - l^2 = 0$

$$\dot{\phi}(\mathbf{X}) = \mathbf{A} \cdot \dot{\mathbf{X}} \,;\, \mathbf{A} = 2 \cdot (\mathbf{X} - \mathbf{X}_{\mathrm{R}})^{\mathrm{T}} \,;$$

$$\ddot{\phi}(\mathbf{X}) = \dot{\mathbf{A}} \cdot \dot{\mathbf{X}} + \mathbf{A} \cdot \ddot{\mathbf{X}} \,;\, \dot{\mathbf{A}} = 2 \cdot \dot{\mathbf{X}}^{\mathbf{T}}$$

Constrained dynamics : $\mathbf{F} = \mathbf{M} \cdot \ddot{\mathbf{X}} - \mathbf{A}^{\mathrm{T}} \cdot \boldsymbol{\lambda} \,;$

$$\rightarrow \begin{bmatrix} \ddot{\mathbf{X}} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{M} & -\mathbf{A}^{\mathrm{T}} \\ \mathbf{A} & 0 \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{F} \\ -\dot{\mathbf{A}} \cdot \dot{\mathbf{x}} \end{bmatrix}$$

$$\rightarrow \begin{bmatrix} \ddot{\mathbf{X}} \\ \dot{\mathbf{X}} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{M} & \mathbf{0} & -\mathbf{A}^{\mathrm{T}} \\ \mathbf{0} & \mathbf{I}_{\mathrm{d}} & \mathbf{0} \\ \mathbf{A} & \dot{\mathbf{A}} & 0 \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{F} \\ \dot{\mathbf{X}} \\ 0 \end{bmatrix}$$

If $\phi(\mathbf{X}) < 0$, unconstrained dynamics : $\begin{bmatrix} \ddot{\mathbf{X}} \\ \dot{\mathbf{X}} \end{bmatrix} = \begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{\mathrm{d}} \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{F} \\ \dot{\mathbf{X}} \end{bmatrix}$

else, constrained dynamics : (at first instant : $\dot{\mathbf{X}}(2) = 0$) and $\begin{bmatrix} \ddot{\mathbf{X}} \\ \dot{\mathbf{X}} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{M} & \mathbf{0} & -\mathbf{A}^{\mathrm{T}} \\ \mathbf{0} & \mathbf{I}_{\mathrm{d}} & \mathbf{0} \\ \mathbf{A} & \dot{\mathbf{A}} & 0 \end{bmatrix}^{-1} \cdot \begin{bmatrix} \mathbf{F} \\ \dot{\mathbf{X}} \\ 0 \end{bmatrix}$

Lagrange Constraint

Application to Quaternion integration (1/2):

The position and attitude of a mobile system are denoted : $\boldsymbol{\eta} = \begin{bmatrix} \mathbf{X} \\ \mathbf{Q} \end{bmatrix}$, while its body-frame (absolute) velocities are : $\boldsymbol{\nu} = \begin{bmatrix} \mathbf{V}_\mathrm{B} = [u, v, w]^\mathrm{T} \\ \mathbf{W}_\mathrm{B} = [p, q, r]^\mathrm{T} \end{bmatrix}$. The kinematic model can be expressed as :

$$\dot{\boldsymbol{\eta}} = \begin{bmatrix} \dot{\mathbf{X}} \\ \dot{\mathbf{Q}} \end{bmatrix} = \begin{bmatrix} \mathbf{T}_\mathrm{Q}^\mathrm{v} \cdot \left(\mathbf{Q} \otimes \left[0, \mathbf{V}_{\boldsymbol{B}}^T\right]^T \otimes \mathbf{Q}^* \right) \\ \frac{1}{2} \cdot \mathbf{Q} \otimes \underbrace{[0, \boldsymbol{\omega}_B^T]^T}_{\boldsymbol{\Omega}_B} \end{bmatrix}$$

where $\mathbf{T}_\mathrm{Q}^\mathrm{v} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$, which allows to transform a pure imaginary quaternion into its equivalent vector.

The inverse transformations uses $\mathbf{T}_\mathrm{v}^\mathrm{Q} = \left(\mathbf{T}_\mathrm{Q}^\mathrm{v}\right)^\mathrm{T}$. Hence the inverse kinematic model is written as :

$$\boldsymbol{\nu} = \begin{bmatrix} \mathbf{V}_\mathrm{B} \\ \mathbf{W}_\mathrm{B} \end{bmatrix} = \begin{bmatrix} \mathbf{T}_\mathrm{v}^\mathrm{Q} \cdot \left(\mathbf{Q}^* \otimes \left[0, \dot{\mathbf{X}}^\mathrm{T}\right]^T \otimes \mathbf{Q} \right) \\ \mathbf{T}_\mathrm{v}^\mathrm{Q} \cdot \left(2 \cdot \mathbf{Q}^* \otimes \dot{\mathbf{Q}}\right) \end{bmatrix}$$

The inverse dynamic model of the system is expressed in $\{B\}$ as : $\dot{\boldsymbol{\nu}} = \begin{bmatrix} \dot{\mathbf{V}}_\mathrm{B} \\ \dot{\mathbf{W}}_\mathrm{B} \end{bmatrix} = \mathbf{M}^{-1} \cdot \left(\mathbf{F}_\mathrm{B}(\mathbf{c_m}) - \mathbf{f}(\boldsymbol{\nu}, \boldsymbol{\eta})\right)$,

Hence, system dynamics can be written as :

$$\ddot{\boldsymbol{\eta}} = \begin{bmatrix} \ddot{\mathbf{X}} = \mathbf{T}_\mathrm{v}^\mathrm{Q} \cdot \left(\mathbf{Q} \otimes \left[0, \dot{\mathbf{V}}_\mathrm{B}^T\right]^T \otimes \mathbf{Q}^* + \dot{\mathbf{Q}} \otimes \left[0, \mathbf{V}_\mathbf{B}^T\right]^T \otimes \mathbf{Q}^* + \mathbf{Q} \otimes \left[0, \mathbf{V}_\mathbf{B}^T\right]^T \otimes \dot{\mathbf{Q}}^* \right) \\ \ddot{\mathbf{Q}} = 0.5 \cdot \left(\mathbf{Q} \otimes [0, \dot{\boldsymbol{\omega}}_B^T]^T + \dot{\mathbf{Q}} \otimes \left[0, \boldsymbol{\omega}_\mathbf{B}^T\right]^T \right) \end{bmatrix}$$

Lagrange Constraint

Application to Quaternion integration (2/2):

The constraint to be considered concerns the quaternion normalisation, that can be written as :

$$\phi(\mathbf{Q}) = \mathbf{Q}^{\mathrm{T}} \cdot \mathbf{Q} - 1 = 0$$

First derivation yields : $\mathbf{Q}^{\mathrm{T}} \cdot \dot{\mathbf{Q}} = \mathbf{A} \cdot \dot{\mathbf{Q}} = 0$, where $\mathbf{A} = \mathbf{Q}^{\mathrm{T}}$

Second derivation : $\dot{\mathbf{A}} \cdot \dot{\mathbf{Q}} + \mathbf{A} \cdot \ddot{\mathbf{Q}} = 0$

The consideration of the normalisation contraint yields to the consideration of the following constrained dynamic system :

$$\ddot{\boldsymbol{\eta}} = \begin{bmatrix} \ddot{\mathbf{X}} \\ \ddot{\mathbf{Q}} \end{bmatrix} = \begin{bmatrix} \mathbf{T}_{\mathrm{V}}^{\mathrm{Q}} \cdot \left(\mathbf{Q} \otimes \left[0, \dot{\mathbf{V}}_{\mathrm{B}}^{T}\right]^{T} \otimes \mathbf{Q}^{*} + \dot{\mathbf{Q}} \otimes \left[0, \mathbf{V}_{\mathbf{B}}^{T}\right]^{T} \otimes \mathbf{Q}^{*} + \mathbf{Q} \otimes \left[0, \mathbf{V}_{\mathbf{B}}^{T}\right]^{T} \otimes \dot{\mathbf{Q}}^{*} \right) \\ 0.5 \cdot \left(\mathbf{Q} \otimes [0, \dot{\boldsymbol{\omega}}_{B}^{T}]^{T} + \dot{\mathbf{Q}} \otimes \left[0, \boldsymbol{\omega}_{\mathbf{B}}^{T}\right]^{T} \right) \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \lambda \cdot \mathbf{A}^{\mathrm{T}} \end{bmatrix}$$

We can then define the following system, to be integrated, with constraint respect :

$$\begin{bmatrix} \ddot{\boldsymbol{\eta}} \\ \dot{\boldsymbol{\eta}} \\ \lambda \end{bmatrix} = \underbrace{\begin{bmatrix} \underset{(7\times7)}{\mathbf{I}_{\mathrm{d}}} & \underset{(7\times7)}{\mathbf{0}} & \underset{(7\times1)}{\begin{bmatrix} \mathbf{0} \\ -\mathbf{A}^{\mathrm{T}} \end{bmatrix}} \\ \underset{(7\times7)}{\mathbf{0}} & \underset{(7\times7)}{\mathbf{I}_{\mathrm{d}}} & \underset{(1\times7)}{\mathbf{0}} \\ \begin{bmatrix} \underset{(1\times3)}{\mathbf{0}} & \underset{(1\times4)}{\mathbf{A}} \end{bmatrix} & \underset{(1\times7)}{\mathbf{0}} & \underset{(1\times1)}{\mathbf{I}_{\mathrm{d}}} \end{bmatrix}^{-1}}_{(15\times15)} \cdot \begin{bmatrix} \mathbf{T}_{\mathrm{V}}^{\mathrm{Q}} \cdot \left(\mathbf{Q} \otimes \left[0, \dot{\mathbf{V}}_{\mathrm{B}}^{T}\right]^{T} \otimes \mathbf{Q}^{*} + \dot{\mathbf{Q}} \otimes \left[0, \mathbf{V}_{\mathbf{B}}^{T}\right]^{T} \otimes \mathbf{Q}^{*} + \mathbf{Q} \otimes \left[0, \mathbf{V}_{\mathbf{B}}^{T}\right]^{T} \otimes \dot{\mathbf{Q}}^{*} \right) \\ 0.5 \cdot \left(\mathbf{Q} \otimes [0, \dot{\boldsymbol{\omega}}_{B}^{T}]^{T} + \dot{\mathbf{Q}} \otimes \left[0, \boldsymbol{\omega}_{\mathbf{B}}^{T}\right]^{T} \right) \\ \dot{\mathbf{X}} \\ \dot{\mathbf{Q}} \\ -\dot{\mathbf{A}} \cdot \dot{\mathbf{Q}} \end{bmatrix}$$

$$\begin{bmatrix} \dot{\boldsymbol{\chi}} \\ \lambda \end{bmatrix} = \boldsymbol{f}_{Ct}(\boldsymbol{\chi}, \mathbf{u})$$

Simulator : Integration

- Numerical solution of an ODE

$$x^{(n)}(t) = f\left(x^{(n-1)}(t), \dots, \dot{x}(t), x(t), u^{(m)}(t), \dots, \dot{u}(t), \mathrm{P}\right) \rightarrow x(t)?$$

Initial conditions at t_0 : $x^{(n-1)}(t_0), \dots, x^{(i)}(t_0), \dots, \dot{x}(t_0), x(t_0)$ et $u^{(m)}(t), \dots, \dot{u}(t), \mathrm{P}$ known

$\rightarrow$ Computat° of $x^{(n)}$ (t_0)= $f\left(x^{(n-1)}(t_0), \dots, \dot{x}(t_0), x(t_0), u^{(m)}(t_0), \dots, \dot{u}(t_0), \mathrm{P}\right)$

$\rightarrow$ Computation at $(t_0 + dt)$: $x^{(n-1)}(t_0 + dt), \dots, x^{(i)}(t_0 + dt), \dots, \dot{x}(t_0 + dt), x(t_0 + dt)$

Integration over an horizon dt :

$$x^{(n-1)}(t_0 + dt) = x^{(n-1)}(t_0) + \int_{t_0}^{t_0+dt} x^{(n)}(\tau)\mathrm{d}\tau$$

$$\vdots$$

$$x^{(i-1)}(t_0 + dt) = x^{(i-1)}(t_0) + \int_{t_0}^{t_0+dt} x^{(i)}(\tau)\mathrm{d}\tau$$

$$\vdots$$

$$x(t_0 + dt) = x(t_0) + \int_{t_0}^{t_0+d} \dot{x}(\tau)\mathrm{d}\tau$$

Simulator : Integration

- Numerical solution of an ODE

$$x^{(n)}(t) = f\left(x^{(n-1)}(t), \dots, \dot{x}(t), x(t), u^{(m)}(t), \dots, \dot{u}(t), \mathrm{P}\right) \rightarrow x(t)?$$

Initial conditions at t_0 : $x^{(n-1)}(t_0), \dots, x^{(i)}(t_0), \dots, \dot{x}(t_0), x(t_0)$ et $u^{(m)}(t), \dots, \dot{u}(t), \mathrm{P}$ known

$\rightarrow$ Computat° of $x^{(n)}$ (t_0)= $f\left(x^{(n-1)}(t_0), \dots, \dot{x}(t_0), x(t_0), u^{(m)}(t_0), \dots, \dot{u}(t_0), \mathrm{P}\right)$

$\rightarrow$ Computation at $(t_0 + dt)$: $x^{(n-1)}(t_0 + dt), \dots, x^{(i)}(t_0 + dt), \dots, \dot{x}(t_0 + dt), x(t_0 + dt)$

Integration over an horizon dt :

$$\underbrace{\begin{bmatrix} x^{(n-1)}(t_0+dt) \\ x^{(n-2)}(t_0+dt) \\ \vdots \\ x^{(i-1)}(t_0+dt) \\ \vdots \\ x(t_0+dt) \end{bmatrix}}_{\boldsymbol{\chi}(t_0+dt)} = \underbrace{\begin{bmatrix} x^{(n-1)}(t_0) \\ x^{(n-2)}(t_0) \\ \vdots \\ x^{(i-1)}(t_0) \\ \vdots \\ x(t_0) \end{bmatrix}}_{\boldsymbol{\chi}(t_0)} + \int_{t_0}^{t_0+d} \underbrace{\begin{bmatrix} x^{(n)}(\tau) \\ x^{(n-1)}(\tau) \\ \vdots \\ x^{(i)}(\tau) \\ \vdots \\ \dot{x}(\tau) \end{bmatrix}}_{\dot{\boldsymbol{\chi}}(\tau)} \cdot \mathrm{d}\tau$$

$$\boldsymbol{\chi}(t_0 + dt) = \boldsymbol{\chi}(t_0) + \int_{t_0}^{t_0+dt} \dot{\boldsymbol{\chi}}(\tau) \cdot \mathrm{d}\tau$$

Simulator : Integration

- Numerical solution of an ODE

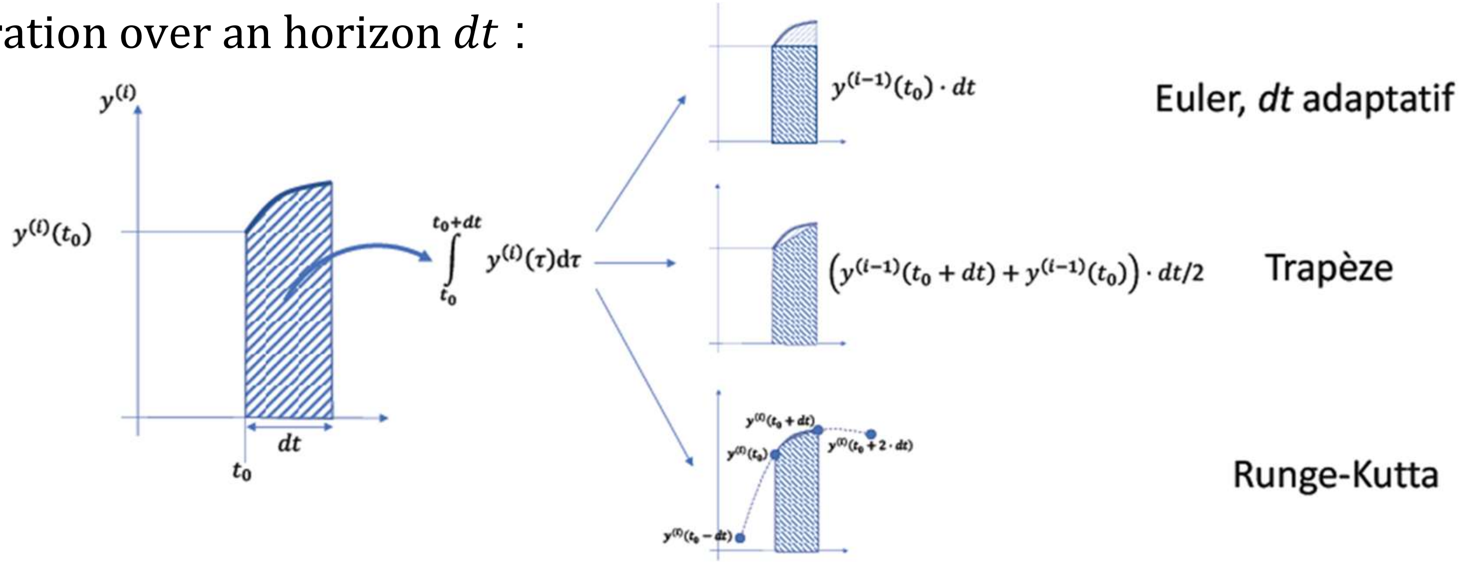
$$x^{(n)}(t) = f\left(x^{(n-1)}(t), \dots, \dot{x}(t), x(t), u^{(m)}(t), \dots, \dot{u}(t), \mathrm{P}\right) \rightarrow x(t)?$$

Initial conditions at t_0 : $x^{(n-1)}(t_0), \dots, x^{(i)}(t_0), \dots, \dot{x}(t_0), x(t_0)$ et $u^{(m)}(t), \dots, \dot{u}(t), \mathrm{P}$ known

→ Computat° of $x^{(n)}$ (t_0)= $f\left(x^{(n-1)}(t_0), \dots, \dot{x}(t_0), x(t_0), u^{(m)}(t_0), \dots, \dot{u}(t_0), \mathrm{P}\right)$

→ Computation at $(t_0 + dt)$: $x^{(n-1)}(t_0 + dt), \dots, x^{(i)}(t_0 + dt), \dots, \dot{x}(t_0 + dt), x(t_0 + dt)$

Integration over an horizon dt :



Simulator : Integration

- Numerical solution of an ODE

$$x^{(n)}(t) = f\left(x^{(n-1)}(t), \dots, \dot{x}(t), x(t), u^{(m)}(t), \dots, \dot{u}(t), \mathrm{P}\right) \rightarrow x(t)?$$

Initial conditions at t_0 : $x^{(n-1)}(t_0), \dots, x^{(i)}(t_0), \dots, \dot{x}(t_0), x(t_0)$ et $u^{(m)}(t), \dots, \dot{u}(t), \mathrm{P}$ known

→ Computat° of $x^{(n)}$ $(t_0)= f\left(x^{(n-1)}(t_0), \dots, \dot{x}(t_0), x(t_0), u^{(m)}(t_0), \dots, \dot{u}(t_0), \mathrm{P}\right)$

→ Computat° at $(t_0 + dt)$: $x^{(n-1)}(t_0 + dt), \dots, x^{(i)}(t_0 + dt), \dots, \dot{x}(t_0 + dt), x(t_0 + dt)$

→ Computat° of $x^{(n)}$ $(t_0 + dt)= f\left(x^{(n-1)}(t_0 + dt), \dots, \dot{x}(t_{0+dt}), x(t_0 + dt) \dots\right)$

→ Computat° at $(t_0 + 2dt)$: $x^{(n-1)}(t_0 + 2dt), \dots, x^{(i)}(t_0 + 2dt), \dots, \dot{x}(t_0 + 2dt), x(t_0 + 2dt)$

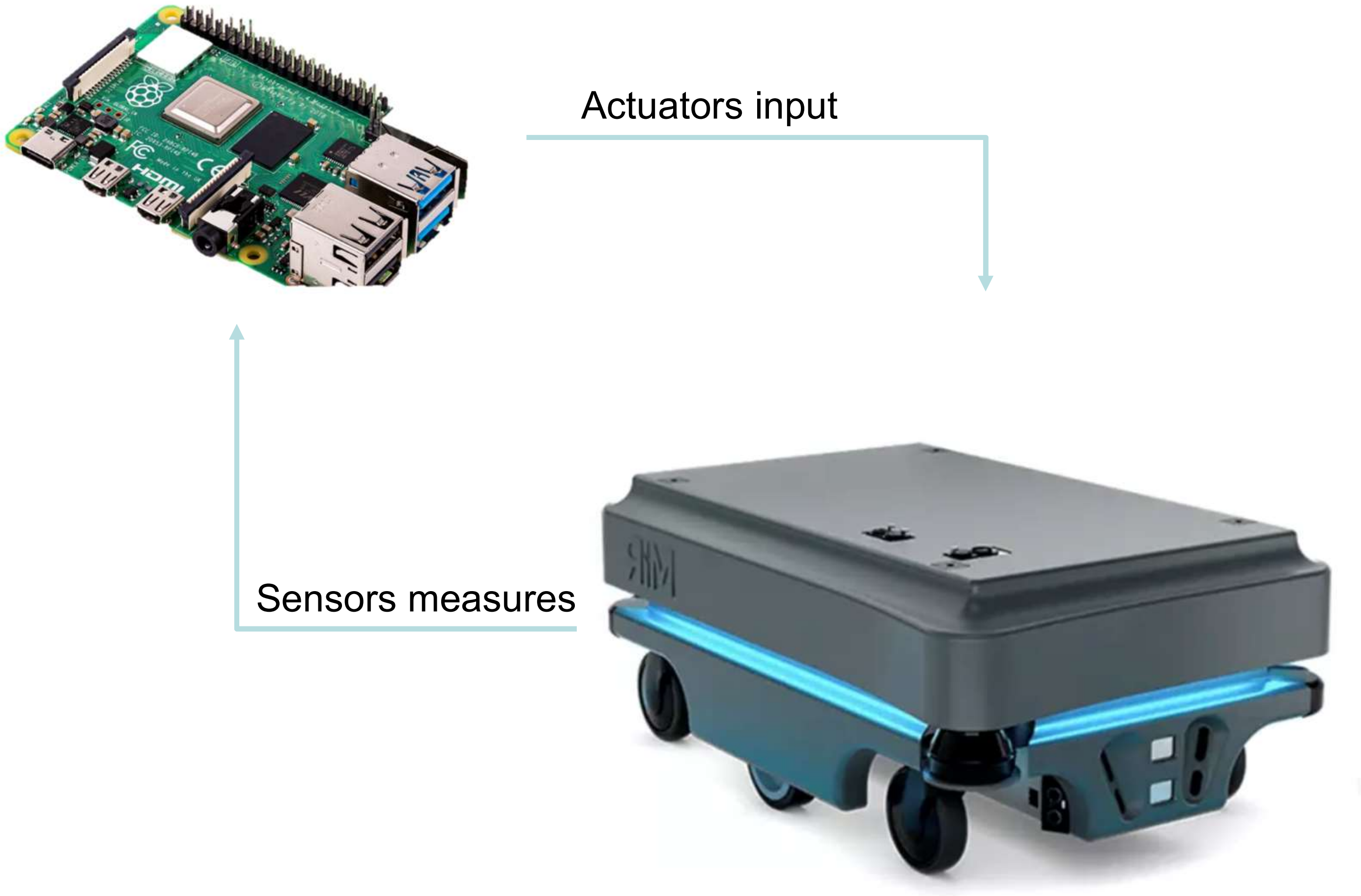
→ ⋯

→ Computat° at $(t_0 + kdt)$: $x^{(n-1)}(t_0 + kdt), \dots, x^{(i)}(t_0 + kdt), \dots, \dot{x}(t_0 + kdt), x(t_0 + kdt)$

$$\boldsymbol{x}(t) = \underbrace{\{x(t_0), x(t_0 + dt), \dots, x(t_0 + kdt), \dots, x(t_0 + ndt)\}}_{\text{Trajectory}}$$

Simulator : structure

- Target



Simulator : structure

- Hardware in the Loop : HIL

The Periods

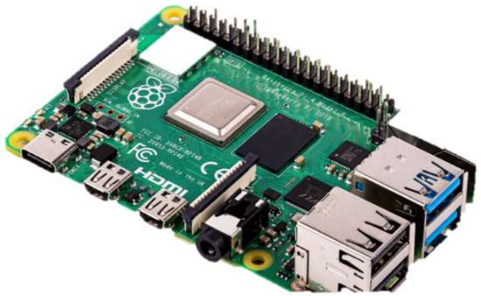
T_{CTRL}: Control period

T_{INTEG}: Integration period ($T_{INTEG} < \frac{T_{CTRL}}{100}$)

T_{S1} : Sampling period of $SENSOR_1$

T_{S2} : Sampling period of $SENSOR_2$

T_{Sn} : Sampling period of $SENSOR_n$



T_{CTRL}

Actuators input

Sensors measures

T_{INTEG}

$$\dot{x}(t) = f(x(t), u(t))$$

$$x(t+dt) = \int_t^{t+dt} \dot{x}(\tau) \cdot d\tau$$

T_{S1} $\quad SENSOR_1(t+dt) = g_1\big(x(t+dt)\big)$

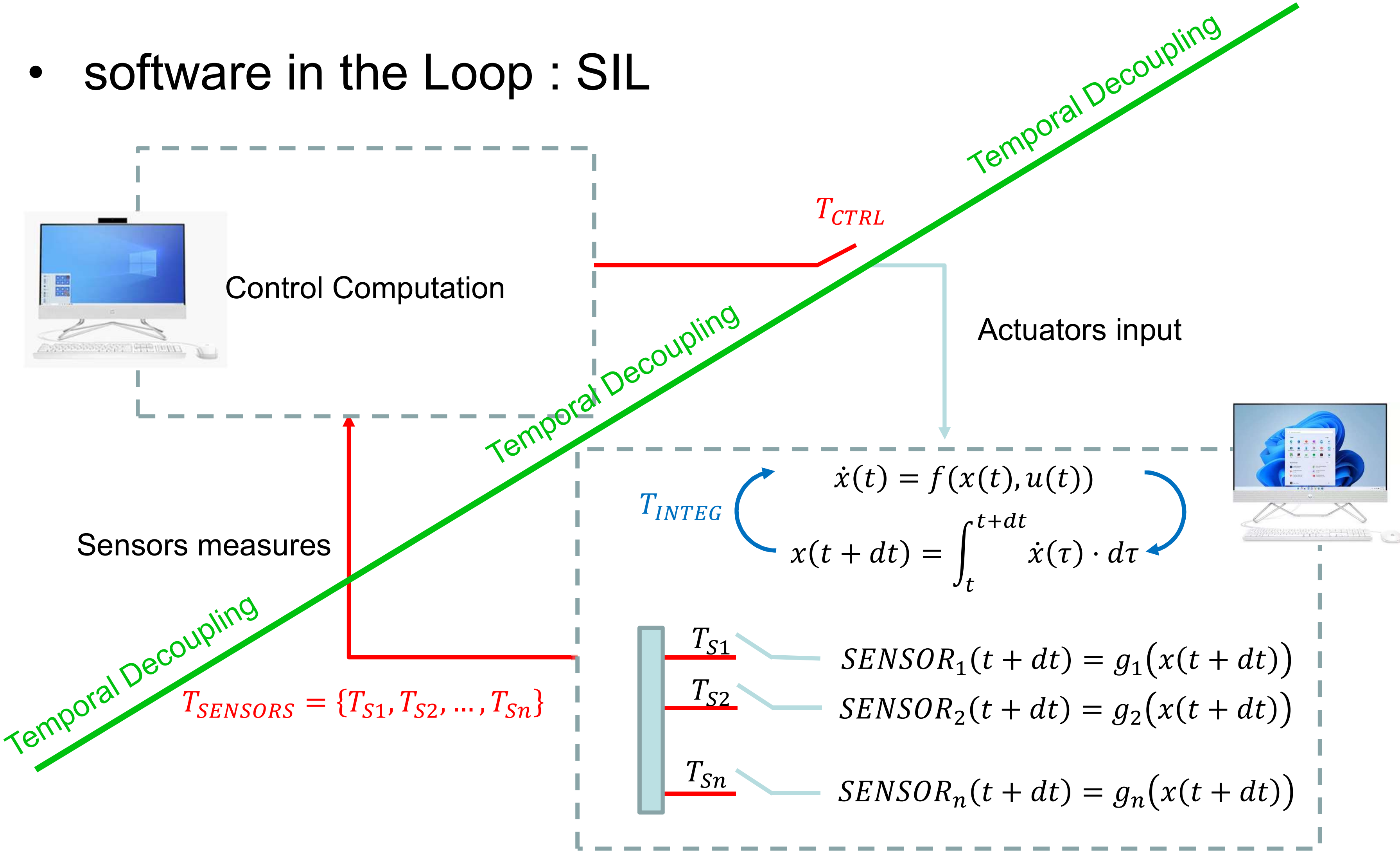
T_{S2} $\quad SENSOR_2(t+dt) = g_2\big(x(t+dt)\big)$

T_{Sn} $\quad SENSOR_n(t+dt) = g_n\big(x(t+dt)\big)$

$T_{SENSORS} = \{T_{S1}, T_{S2}, \dots, T_{Sn}\}$

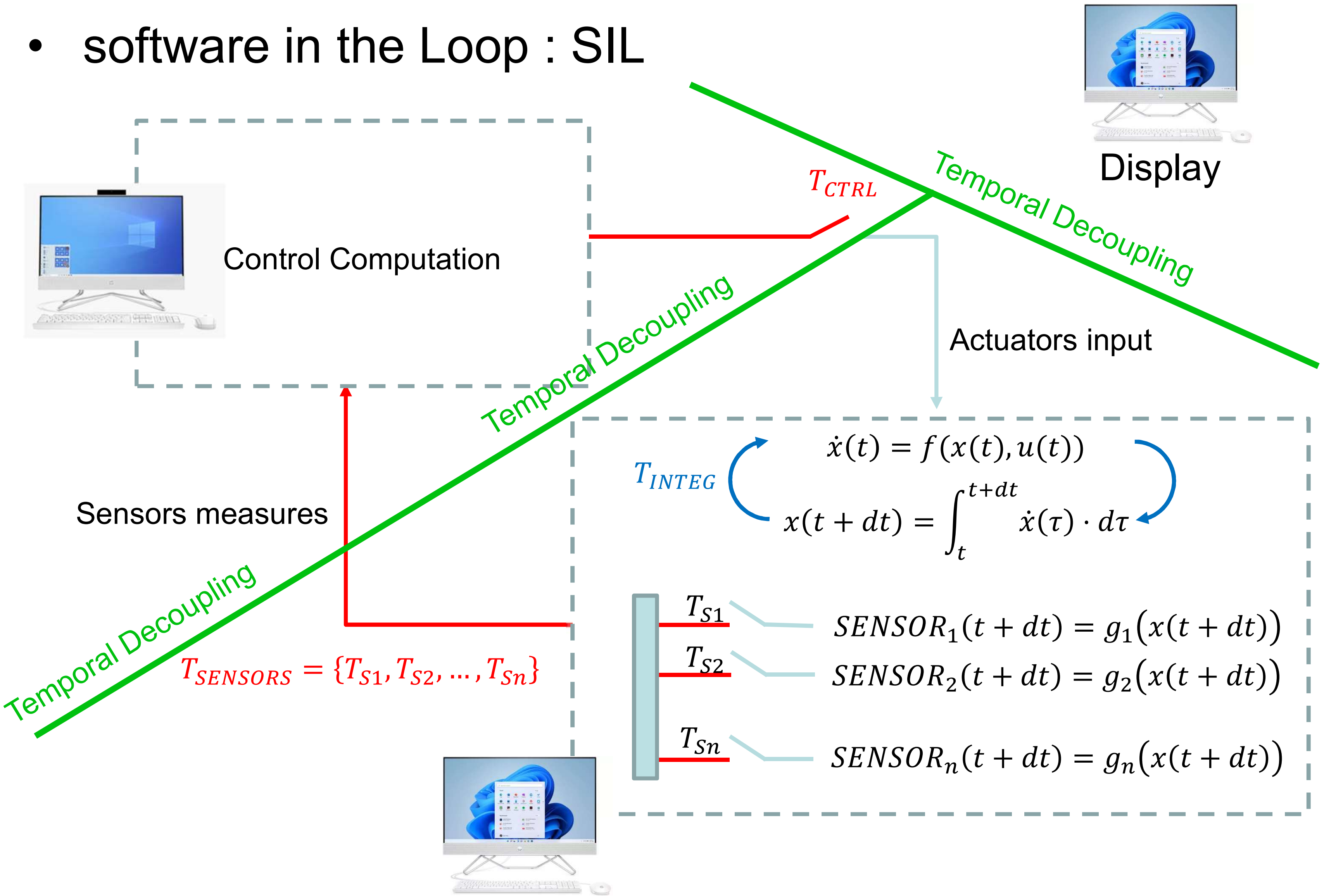
Simulator : structure

- software in the Loop : SIL



Simulator : structure

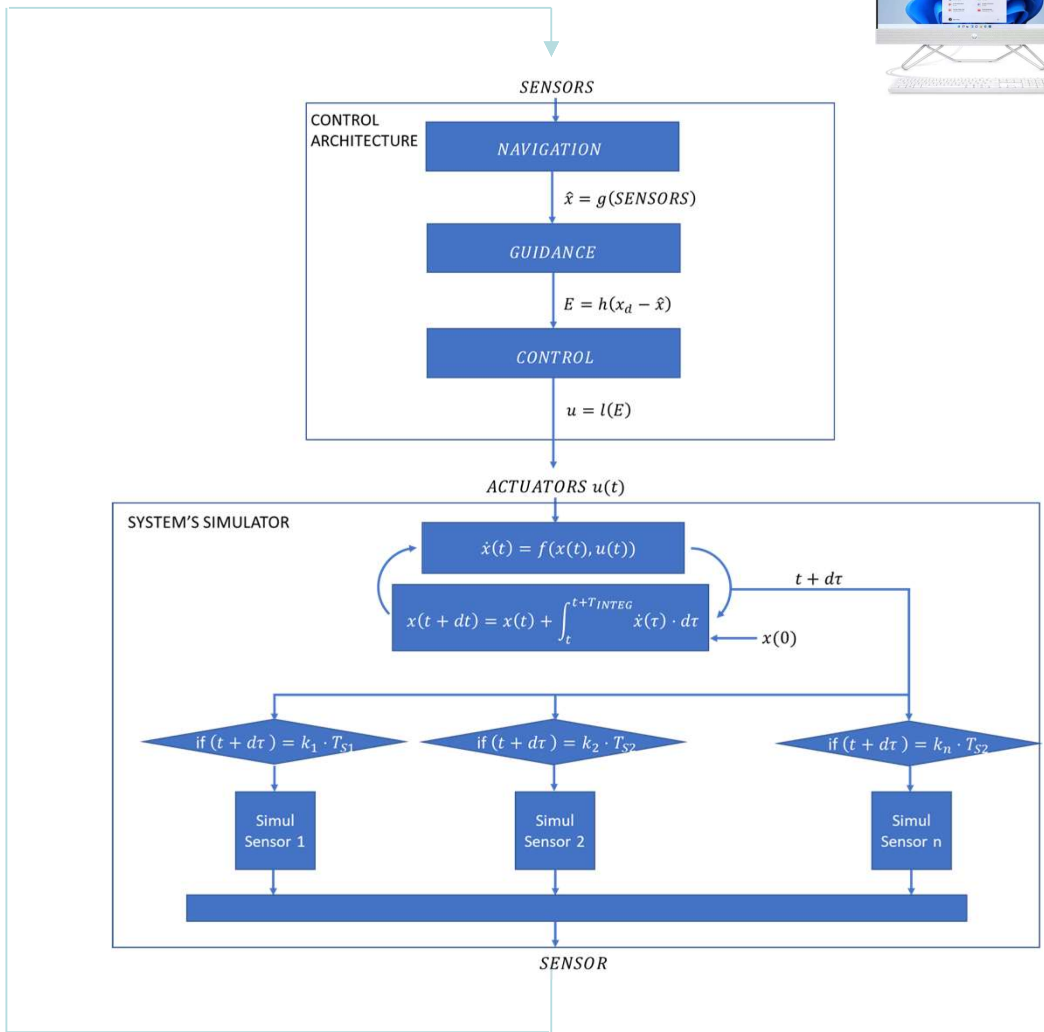
- software in the Loop : SIL



Simulator : structure

- Basic simulation

Control computation
Physics computation
Display



- Exercices:
- 1) develop a ‘classic’ simulator structure of :
  - Falling brick
  - AUV

AUV Parameters

Longueur_Body = 1;
Longueur_Nose = 0.2;
Longueur_Tail = 0.2;
Diametre_body = 0.1;

```
%% Generic Shape
% (3)
% ______
% |      \ |     1/3*longueur
% |        \
% |        |\
% |        |  \ (2)
% |        |  |
% |        |  |
% |        |  |
% |        |  | 1/2* longueur
% |        |  |
% |        |  |
% |        |  |
% |________|__|(1)
% (4)      |
% -2/3*largeur | 1/3*largeur
```

Largeur_Gouverne = 0.1;
Longueur_Gouverne = 0.12;
Gouverne_Vertices = [Largeur_Gouverne*(1/3),0,0;...
Largeur_Gouverne*(1/3),Longueur_Gouverne*(1/2),0;...
-Largeur_Gouverne*(2/3),Longueur_Gouverne,0;...
-Largeur_Gouverne*(2/3),0,0];
Gouverne_faces = [1 2 3 4];
Gouverne_Surface = 0.01;
Gouvernes(1).Type = 'Foil_Carract_NACA_0012.m';
Gouvernes(i).Surf=0.008;

Largeur_Prop = 0.05;
Diametre_Prop = 0.1;
Thruster.Pos_B = [-0.76,0,0];

mu=10;mv=10;mw=10;mp=1;mq=5;mr=5;
du=1;dv=5;dw=5;dp=0.1;dq=1;dr=1;
Yur=1;