



Tecnológico  
de Monterrey

# Reporte

## Semana 1 y 2

Andrea Valeria Pintor Valencia A01067424

Victoria Ramirez Castro - A01640824

Integración de robótica y sistemas inteligentes

Gpo 501

04 de abril de 2025

<b>Embedded Systems Module</b>	<b>3</b>
System Development Life Cycle (SDLC)	3
SDLC phases:	3
System Analysts	3
<b>Neurorobotics Model</b>	<b>3</b>
Probabilistic Methods for Robotics	3
Probability Theory	4
Basic Probability	4
Discrete Random Variables	5
Kalman Filter	5
<b>ROS2 Module</b>	<b>6</b>
Activity 1. Introduction to ROS2	6
Activity 2. ROS2 Nodes	6
<b>Autonomous Robots Module</b>	<b>7</b>
Linear Algebra	7
Eigenvalues and Eigenvector	7
Dynamic Systems	7
Equilibrium or Fixed Points	7
Flows	7
Linearization	8
Linear Control Theory	8
Activity Eigenvalues and Eigenvectors	9
Activity Fixed Points and Equilibrium	11
Activity Double Pendulum Model	12
System and linearization	12
System simulation	14
Linearization simulation	14
Activity Dynamic Model RDK X3	15
System and linearization	15
System simulation	16
Linearization simulation	16
Activity Lineal Control	17
Activity Rover Feedback Control	19
Simulation	21
ROS2 Nodes	21
<b>Vision Module</b>	<b>22</b>
Deep Learning	22
Training	22
Types of Neural Layers	22

# Embedded Systems Module

## System Development Life Cycle (SDLC)

The overall process for developing information systems from planning and analysis through implementation and maintenance.

SDLC phases:

- **Planning Phase:** Establishes a high-level plan of the intended project and determines project goals.
- **Analysis phase:** Involves analyzing end-user business requirements and refining project goals into defined functions and operations of the intended system.
- **Design phase:** Establishes descriptions of the desired features and operations of the system including screen layouts, business rules, process diagrams, pseudo code, and other documentation.
- **Development phase:** Involves taking all of the detailed design documents from the design phase and transforming them into the actual system.
- **Testing phase:** Involves bringing all the project pieces together into a special testing environment to eliminate errors and bugs, and verify that the system meets all of the business requirements defined in the analysis phase.
- **Implementation phase:** Involves placing the system into production so users can begin to perform actual business operations with it.
- **Maintenance phase:** Involves performing changes, corrections, additions, and upgrades to ensure the system continues to meet its business goals.

## System Analysts

They are key individuals in the systems development process. They are needed to determine how people, data, processes, communications, and information technology can best accomplish improvements for the business

## Neurorobotics Model

## Probabilistic Methods for Robotics

- Probabilistic robotics deals with uncertainties in robot perception and action.
  - Estimates state from sensor data
- **Belief distribution.** probabilistic state estimation.
- **State.** collection of all aspects of the robot.
  - Example: velocity, sensor status.
- **Measurements.** information about a momentary state of the environment.
  - Example: range scans.

- **Control Data.** information of state of change.
  - Example: velocity of a robot.
- **State transition probability**

$$p(x_t|x_{t-1}, u_t)$$

- **Measurement probability**

$$p(z_t|x_t)$$

- **Posterior probability** of the state.

$$p(x_t|z_{1:t}, u_{1:t})$$

## Probability Theory

- **Probability.** Chance that a given event occurs.
- **Random variable.** Rule (or function) that assigns a real number to every outcome of a random experiment.
- **Random process.** Rule (or function) that assigns a time function to every outcome of a random experiment.

Set theory	Probability theory	Probability symbol
Universe	Sample space (certain event)	$\mathcal{S}$
Element	Outcome (sample point)	$s$
Subset	Event	$A, B, \text{ or } C \dots$
Disjoint sets	Mutually exclusive events	$E_i \cap E_j = \emptyset$
Null set	Impossible event	$\emptyset$
Simple set	Simple event	$B = \{s\}$

## Basic Probability

- Develops a probabilistic model of a physical experiment.
- **Axioms.** Clear and obvious proposition accepted with no demonstrations.
- **Relative frequency.** Common sense, results of observations.
- **Trial.** Single realization of an experiment.
- **Outcome.** Result of a trial.
- **Probability of an event.** Sum of probabilities of each simple event.
- **Experiment:**
  - Procedure.
  - Observation.
  - Model.

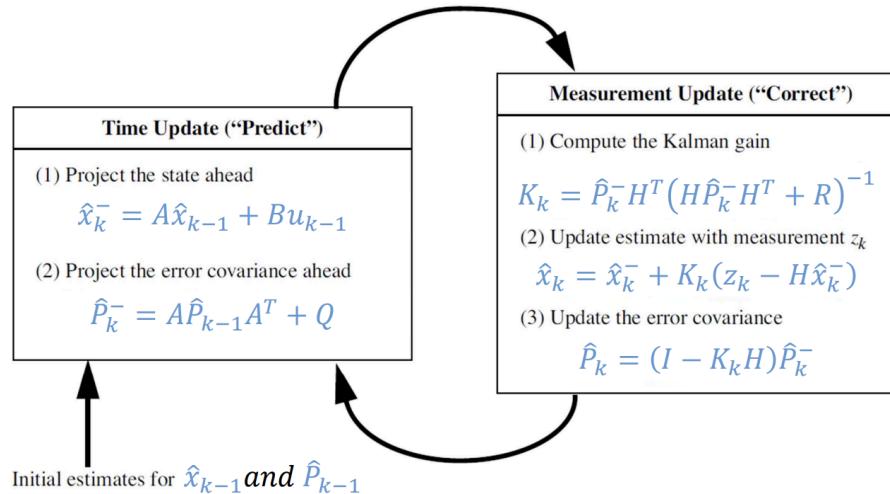
## Discrete Random Variables

- **Random variable.** function that assigns a real number to each outcome of a random experiment.

	Values	PMF	$E[X]$	$\text{var}(X)$
Uniform	$k = -M, \dots, M$	$\frac{1}{2M+1}$	0	$\frac{M(M+1)}{3}$
Bernoulli	$k = 0, 1$	$p^k(1-p)^{1-k}$	$p$	$p(1-p)$
Binomial	$k = 0, 1, \dots, M$	$\binom{M}{k} p^k(1-p)^{M-k}$	$Mp$	$Mp(1-p)$
Geometric	$k = 1, 2, \dots$	$(1-p)^{k-1}p$	$\frac{1}{p}$	$\frac{1-p}{p^2}$
Poisson	$k = 0, 1, \dots$	$\exp(-\lambda) \frac{\lambda^k}{k!}$	$\lambda$	$\lambda$

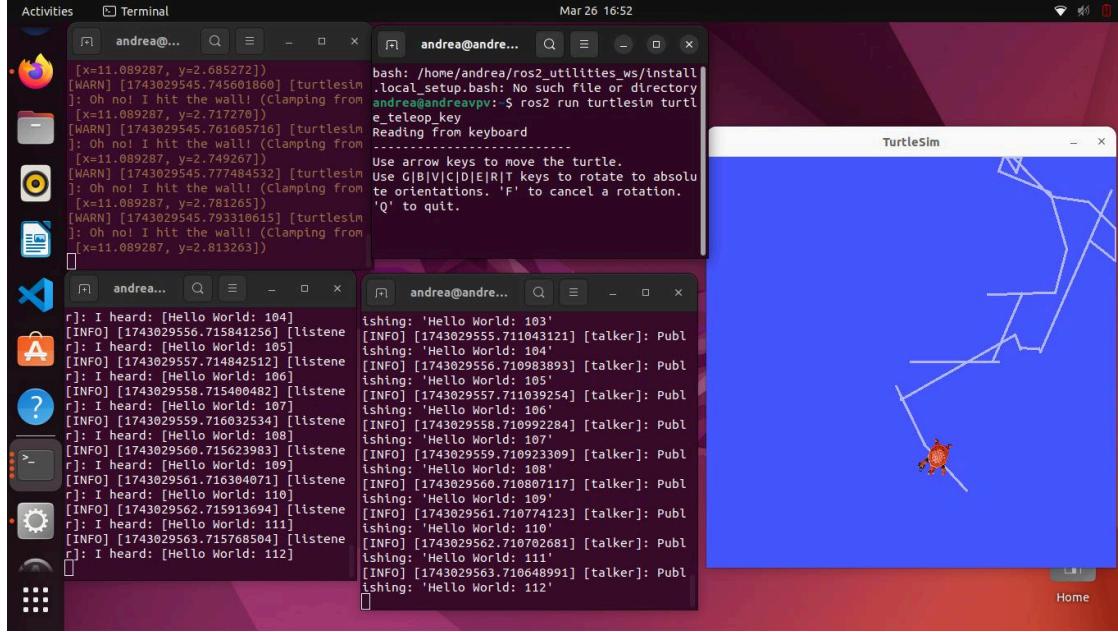
## Kalman Filter

Algorithm used to fuse measurements from sensors to estimate in real time the state of a dynamical system.

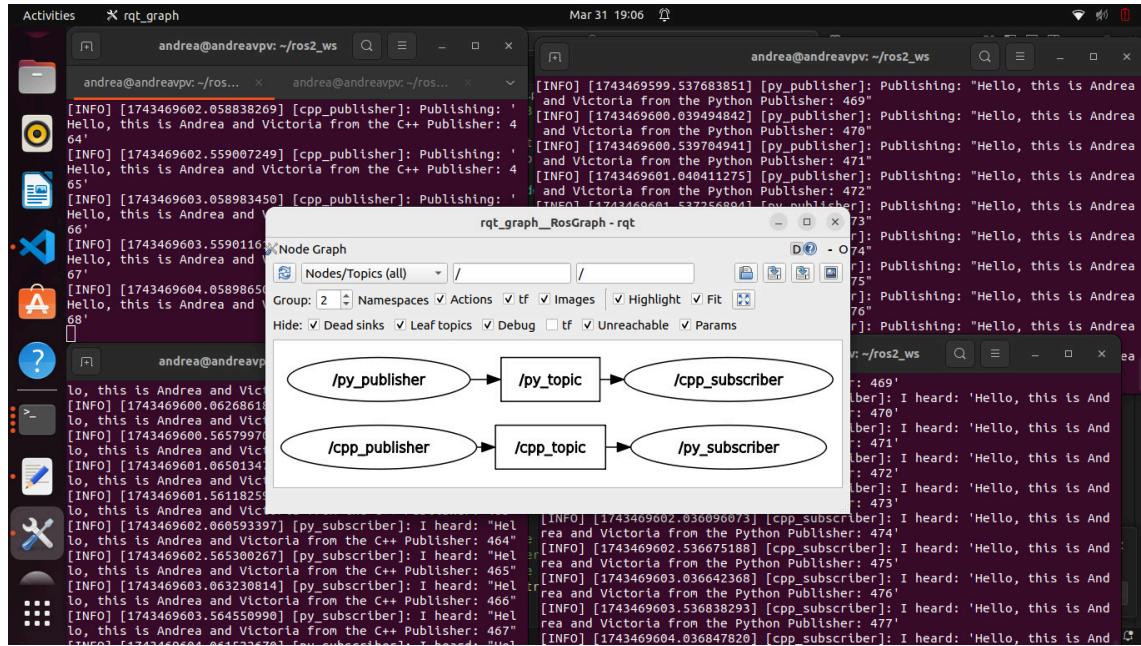


# ROS2 Module

## Activity 1. Introduction to ROS2



## Activity 2. ROS2 Nodes



# Autonomous Robots Module

## Linear Algebra

### Eigenvalues and Eigenvector

- **Eigenvalues.** Let  $A \in \mathbb{R}^{N \times N}$  then a scalar  $\lambda$  is an eigenvalue if there exists a vector  $x \neq 0$  such that

$$Ax = \lambda x \text{ for } x \neq 0$$

- $(A - \lambda I)$  no es invertible.
- $\det(A - \lambda I) = 0$
- If an eigenvalue exists, it means that an invariant subspace exists.
- For each eigenvalue, we have an eigenvector.
- When a matrix is non-invertible, there are  $\infty$  solutions.

- **Eigenspace.** Collection of all eigenvectors corresponding to  $\lambda$

$$E\lambda$$

### Dynamic Systems

- **Dynamics.** Study of motion through phase space.
- **Vector field.** is a function  $f: \mathbb{R} \times \mathbb{R} \times \dots \times \mathbb{R} \rightarrow \mathbb{R}^n$  that assigns a vector  $v \in \mathbb{R}^n$  to each point in space. That is

$$f(x_1, x_2, \dots, x_n)$$

- **Dynamical System.** Transformation or function  $f$  normally applied to a vector field  $x$ . This vector field is said to be the states of the dynamical system.

$$\begin{aligned} x' &= f(x), x \in \mathbb{R}^n \\ &\text{(autonomous system)} \end{aligned}$$

### Equilibrium or Fixed Points

- **Equilibrium point.** Particular point in phase space where the dynamical system stays stationary.

$$x' = f(x) = 0$$

### Flows

This is a way of viewing an equation in vector field space.

- **Flow.** Interpretation of a differential equation as a vector field. Where the vector field moves in an arbitrary initial condition.
  - The flow goes to the right if  $x' > 0$
  - The flow goes to the left if  $x' < 0$

- There is no flow if  $x' = 0$
- Fixed point **stable** if the flow goes towards it.
- Fixed point **unstable** if the flow goes outwards.

## Linearization

- Approximate a nonlinear function or a nonlinear system with a linear function at a specific fixed point
- **Jacobian Linearization**

$$A := \frac{\partial f(x)}{\partial x} \Big|_{\substack{x=x^* \\ u=u^*}}$$

$$\dot{\delta}_x(t) = A \delta_x(t) + B \delta_u(t)$$

$$B := \frac{\partial f(u)}{\partial u} \Big|_{\substack{x=x^* \\ u=u^*}}$$

Example:

EJEMPLO 1.  $\dot{x}_1 = \begin{bmatrix} x_2 \\ -\frac{g}{L} \sin(x_1) \end{bmatrix}$

$B = \emptyset$  ! U no existe  $\rightarrow$  derivada parcial es  $\emptyset$

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \end{bmatrix}$$

$f_1 = x_2$   
 $f_2 = -\frac{g}{L} \sin(x_1)$

$$A = \begin{bmatrix} 0 & 1 \\ -\frac{g}{L} \cos(x_1) & 0 \end{bmatrix}$$

Paso 1. Derivar con respecto a  $x_n$

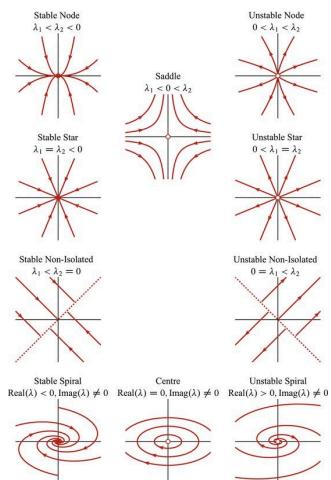
$$A = \begin{bmatrix} 0 & 1 \\ -\frac{g}{L} & \emptyset \end{bmatrix}$$

Paso 2. Evaluar en 1 punto fijo

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -\frac{g}{L} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Paso 3. Modelo Dinámico linearizado

## Linear Control Theory



Linear control theory is used to view the  $\mathbb{R}^2$  flow of linear systems. An uncoupled system is a system that can be developed in two equations. Equation 1 is only in terms of  $x$ , and equation 2 is only in terms of  $y$ . To determine whether a system is stable or not, consider the diagonal matrix or the eigenvalues. It is important to remember that in a diagonal matrix, the values on the diagonal are the eigenvalues.

If the values on the main diagonal are:

- **Negative**: the system is stable.
- **Positive**: the system is unstable.
- **Positive and negative**: the system is unstable.

## Activity Eigenvalues and Eigenvectors

a)  $A = \begin{bmatrix} 2 & 2 \\ 2 & -1 \end{bmatrix}$

$$A - \lambda I = \begin{bmatrix} 2 & 2 \\ 2 & -1 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} = \begin{bmatrix} 2-\lambda & 2 \\ 2 & -1-\lambda \end{bmatrix}$$

$$\begin{aligned} \det(A - \lambda I) &= (2-\lambda)(-1-\lambda) - 4 \\ &= -2 - 2\lambda + \lambda + \lambda^2 - 4 \\ &= \lambda^2 - \lambda - 6 = (\lambda - 3)(\lambda + 2) \end{aligned}$$

$$\lambda_1 = 3 \quad \lambda_2 = -2$$

para  $\lambda_1 = 3$

$$(A - \lambda I)x = 0$$

$$\begin{bmatrix} 2-\lambda_1 & 2 \\ 2 & -1-\lambda_1 \end{bmatrix}x = 0 \Rightarrow \begin{bmatrix} -1 & 2 \\ 2 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$-x_1 + 2x_2 = 0$$

$$2x_1 - 4x_2 = 0$$

$$x_1 = 2x_2$$

$$x = s \begin{bmatrix} 1 \\ 1/2 \end{bmatrix}, s \in \mathbb{R}$$

para  $\lambda_2 = -2$

$$(A - \lambda I)x = 0$$

$$\begin{bmatrix} 2-\lambda_2 & 2 \\ 2 & -1-\lambda_2 \end{bmatrix}x = 0 \Rightarrow \begin{bmatrix} 4 & 2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$4x_1 + 2x_2 = 0$$

$$2x_1 + x_2 = 0$$

$$x_2 = -2x_1$$

$$x_1 = -\frac{1}{2}x_2$$

$$x = \begin{bmatrix} 1 \\ -2 \end{bmatrix}, s \in \mathbb{R}$$

b)  $B = \begin{bmatrix} 0 & 4 \\ -1 & s \end{bmatrix}$

$$B - \lambda I = \begin{bmatrix} 0 & 4 \\ -1 & s \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} = \begin{bmatrix} -\lambda & 4 \\ -1 & s-\lambda \end{bmatrix}$$

$$\begin{aligned} \det(B - \lambda I) &= (-\lambda)(s-\lambda) - (-4) \\ &= -s\lambda + \lambda^2 + 4 = (\lambda - 4)(\lambda - 1) \end{aligned}$$

$$\lambda_1 = 4 \quad \lambda_2 = 1$$

para  $\lambda_1 = 4$

$$(B - \lambda I)x = 0$$

$$\begin{bmatrix} -\lambda_1 & 4 \\ -1 & s-\lambda_1 \end{bmatrix}x = 0 \Rightarrow \begin{bmatrix} -4 & 4 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$\begin{aligned} -4x_1 + 4x_2 &= 0 \\ -x_1 + x_2 &= 0 \\ x_2 &= x_1 \end{aligned}$$

$$x = \begin{bmatrix} 1 \\ s \end{bmatrix}, s \in \mathbb{R}$$

para  $\lambda_2 = 1$

$$(A - \lambda I)x = 0$$

$$\begin{bmatrix} -\lambda_2 & 4 \\ -1 & s-\lambda_2 \end{bmatrix}x = 0 \Rightarrow \begin{bmatrix} -1 & 4 \\ -1 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$-x_1 + 4x_2 = 0$$

$$-x_1 + 4x_2 = 0$$

$$x_2 = \frac{x_1}{4}$$

$$x = \begin{bmatrix} 1 \\ \frac{s}{4} \end{bmatrix}, s \in \mathbb{R}$$

c)  $C = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$

$$A - \lambda I = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} = \begin{bmatrix} -\lambda & -1 \\ 1 & -\lambda \end{bmatrix} = 0$$

$$\det(A - \lambda I) = (-\lambda)(-\lambda) - (-1)$$

$$= \lambda^2 + 1$$

$$\lambda_1 = i \quad \lambda_2 = -i$$

para  $\lambda_1 = i$

$$(A - \lambda I)x = 0$$

$$\begin{bmatrix} -\lambda_1 & -1 \\ 1 & -\lambda_1 \end{bmatrix}x = 0 \Rightarrow \begin{bmatrix} -i & -1 \\ 1 & -i \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$-ix_1 - x_2 = 0$$

$$x_1 - ix_2 = 0$$

$$x_1 = ix_2$$

$$x_2 = -ix_1$$

$$x = \begin{bmatrix} 1 \\ -i \end{bmatrix}, s \in \mathbb{R}$$

para  $\lambda_2 = -i$

$$(A - \lambda I)x = 0$$

$$\begin{bmatrix} -\lambda_2 & -1 \\ 1 & -\lambda_2 \end{bmatrix}x = 0 \Rightarrow \begin{bmatrix} i & -1 \\ 1 & i \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$ix_1 - x_2 = 0$$

$$x_1 + ix_2 = 0$$

$$x_1 = -ix_2$$

$$ix_1 = x_2$$

$$x = \begin{bmatrix} 1 \\ i \end{bmatrix}, s \in \mathbb{R}$$

d)  $D = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$

$$A - \lambda I = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} = \begin{bmatrix} 1-\lambda & 0 \\ 0 & -1-\lambda \end{bmatrix} = 0$$

$$\det(A - \lambda I) = (1-\lambda)(-1-\lambda) - 0$$

$$= -1 - \lambda + \lambda + \lambda^2$$

$$= \lambda^2 - 1 = (\lambda - 1)(\lambda + 1)$$

$$\lambda_1 = 1 \quad \lambda_2 = -1$$

para  $\lambda_1 = 1$

$$(A - \lambda I)x = 0$$

$$\begin{bmatrix} 1-\lambda_1 & 0 \\ 0 & -1-\lambda_1 \end{bmatrix}x = 0 \Rightarrow \begin{bmatrix} 0 & 0 \\ 0 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$-2x_2 = 0$$

$$x_2 = 0$$

$$x = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, s \in \mathbb{R}$$

para  $\lambda_2 = -1$

$$(A - \lambda I)x = 0$$

$$\begin{bmatrix} 1-\lambda_2 & 0 \\ 0 & -1-\lambda_2 \end{bmatrix}x = 0 \Rightarrow \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

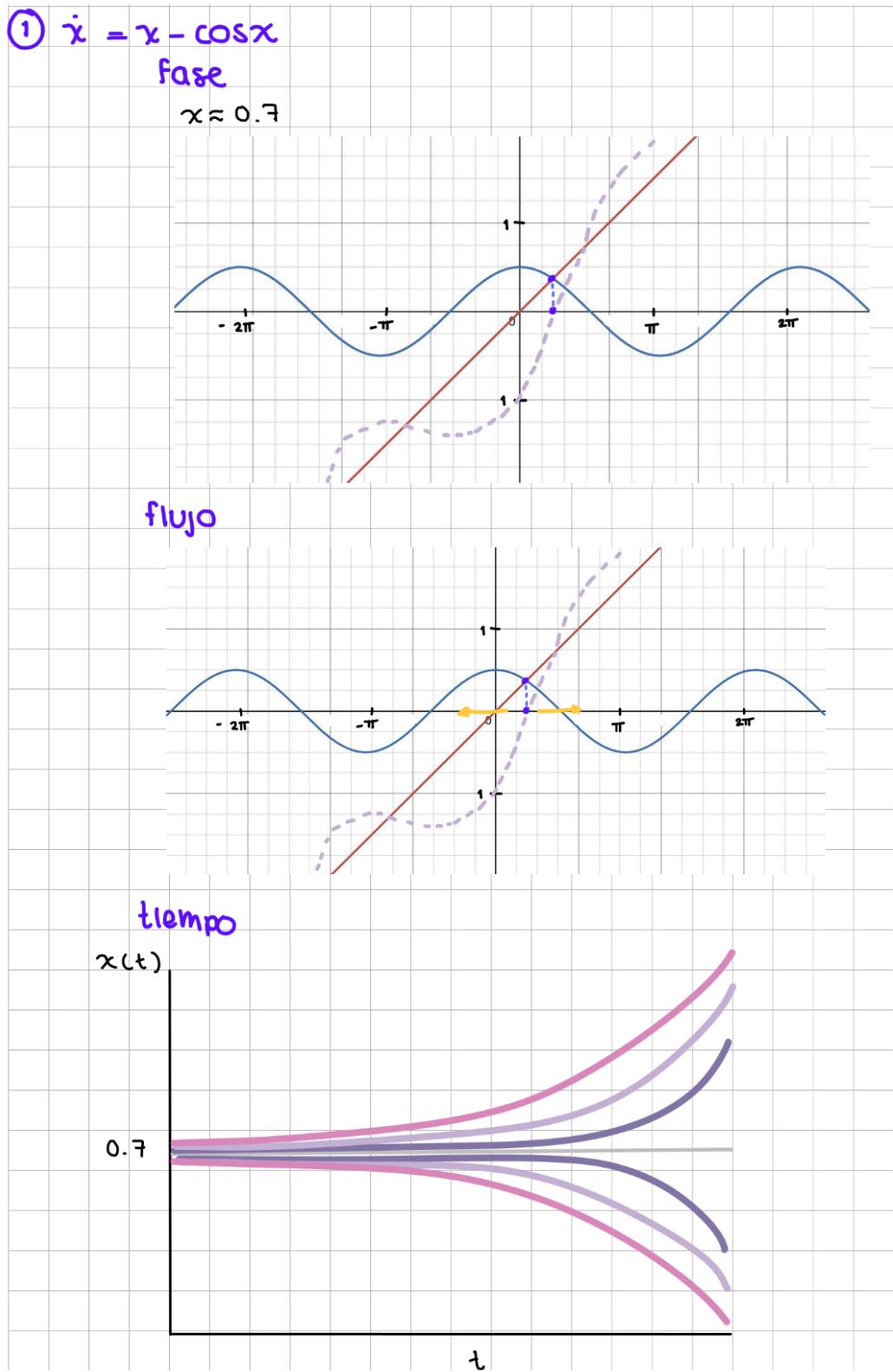
$$2x_1 = 0$$

$$x_1 = 0$$

$$x = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, s \in \mathbb{R}$$

## Activity Fixed Points and Equilibrium

### Flujo inestable



## Activity Double Pendulum Model

System and linearization

**Modelo Dinámico del Doble pendulo**

Lagrangian dynamic equations [from Newton's Eq.]

(1)  $\frac{d}{dt} \cdot \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = \tau_i$

(2)  $\begin{cases} p_1 = l_1 \begin{bmatrix} s_{q_1} \\ -c_{q_1} \end{bmatrix}, p_2 = p_1 + l_2 \begin{bmatrix} s_{q_1+q_2} \\ -c_{q_1+q_2} \end{bmatrix} \\ \dot{p}_1 = l_1 \dot{q}_1 \begin{bmatrix} c_{q_1} \\ s_{q_1} \end{bmatrix}, \dot{p}_2 = \dot{p}_1 + l_2 (\dot{q}_1 + \dot{q}_2) \begin{bmatrix} c_{q_1+q_2} \\ s_{q_1+q_2} \end{bmatrix} \end{cases}$

Kinetic & Potential Energy

$T = \frac{1}{2} m_1 \dot{p}_1^2 + \frac{1}{2} m_2 \dot{p}_2^2$

(3)  $T = \frac{1}{2} (m_1 + m_2) l_1^2 \dot{q}_1^2 + \frac{1}{2} m_2 l_2^2 (\dot{q}_1 + \dot{q}_2)^2 + m_2 l_1 l_2 \dot{q}_1 (\dot{q}_1 + \dot{q}_2) c_{q_2}$

(4)  $U = m_1 g y_1 + m_2 g y_2 = - (m_1 + m_2) g l_1 c_{q_1} - m_2 g l_2 c_{q_1+q_2}$

After partial derivatives & substitute in (1)

(5)  $(m_1 + m_2) l_1^2 \ddot{q}_1 + m_2 l_2^2 (\ddot{q}_1 + \ddot{q}_2) + m_2 l_1 l_2 (2\ddot{q}_1 + \ddot{q}_2) c_{q_2} - m_2 l_1 l_2 (2\dot{q}_1 + \dot{q}_2) \dot{q}_2 s_{q_2} + (m_1 + m_2) l_1 g s_{q_1} + m_2 g l_2 s_{q_1+q_2} = \tau_1$

(6)  $m_2 l_2^2 (\ddot{q}_1 + \ddot{q}_2) + m_2 l_1 l_2 \ddot{q}_1 c_{q_2} + m_2 l_1 l_2 \dot{q}_1^2 s_{q_2} + m_2 g l_2 s_{q_1+q_2} = \tau_2$

(5) & (6) are equations of Motion  
can be rewritten as

(7)  $M(q) \ddot{q} + C(q, \dot{q}) \dot{q} = \tau_g(q) + Bu$

vector de estados  $q[\theta_1, \theta_2]$  where

$M(q) = \begin{bmatrix} (m_1 + m_2) l_1^2 + m_2 l_2^2 + 2m_2 l_1 l_2 c_{q_2} & m_2 l_2^2 + m_2 l_1 l_2 c_{q_2} \\ m_2 l_2^2 + m_2 l_1 l_2 c_{q_2} & m_2 l_2 \end{bmatrix}$

$(C(q, \dot{q})) = \begin{bmatrix} 0 & -m_2 l_1 l_2 (2\dot{q}_1 + \dot{q}_2) s_{q_2} \\ \frac{1}{2} m_2 l_1 l_2 (2\dot{q}_1 + \dot{q}_2) s_{q_2} & -\frac{1}{2} m_2 l_1 l_2 \dot{q}_1 s_{q_2} \end{bmatrix}$

$\tau_g(q) = -g \begin{bmatrix} (m_1 + m_2) l_1 s_{q_1} + m_2 l_2 s_{q_1+q_2} \\ m_2 l_2 s_{q_1+q_2} \end{bmatrix}$        $B = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

reescribir (7)  
 $M\ddot{q} + C\dot{q} = \tau + Bu$

Cambio de variable

$$\dot{q}_1 = q_1$$

$$\dot{\dot{q}}_1 = \ddot{q}_1 = q_2$$

$$\ddot{q} = \ddot{q}_1 = q_2$$

despejar  $\ddot{q}$  de (7)

$$(8) \quad \ddot{q} = M^{-1}[\tau + Bu - C\dot{q}]$$

$$\ddot{q} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}$$

reescibir (8) en forma matricial

$$(9) \quad \ddot{q} = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = \begin{bmatrix} q_2 \\ M^{-1}[\tau + Bu - Cq_2] \end{bmatrix}$$

hacer multiplicación y hacer expansión de (9)

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = \begin{bmatrix} q_2 \\ M^{-1}\tau + M^{-1}Bu - M^{-1}Cq_2 \end{bmatrix}$$

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = \underbrace{\begin{bmatrix} q_2 \\ M^{-1}\tau - M^{-1}Cq_2 \end{bmatrix}}_{A} + \underbrace{\begin{bmatrix} 0 \\ M^{-1}Bu \end{bmatrix}}_{B}$$

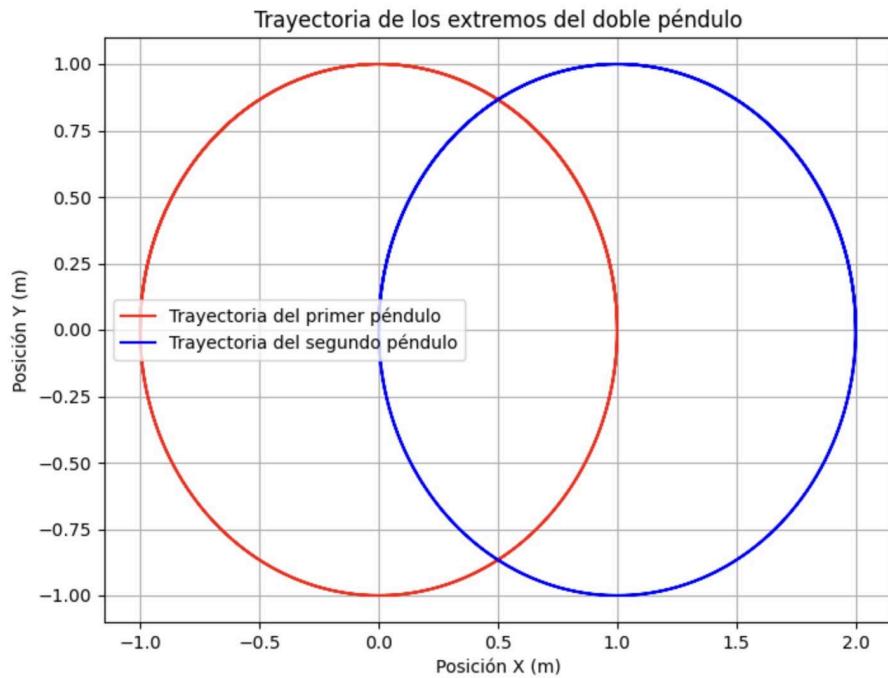
$$A = \begin{bmatrix} q_2 \\ M^{-1}\tau - M^{-1}Cq_2 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ M^{-1}Bu \end{bmatrix}$$

$$A = \begin{bmatrix} \frac{\partial q_2}{\partial q_1} & \frac{\partial q_2}{\partial q_2} \\ \frac{\partial M^{-1}\tau - M^{-1}Cq_2}{\partial q_1} & \frac{\partial M^{-1}\tau - M^{-1}Cq_2}{\partial q_2} \end{bmatrix} \quad B = \begin{bmatrix} \frac{\partial 0}{\partial u} \\ \frac{\partial M^{-1}Bu}{\partial u} \end{bmatrix}$$

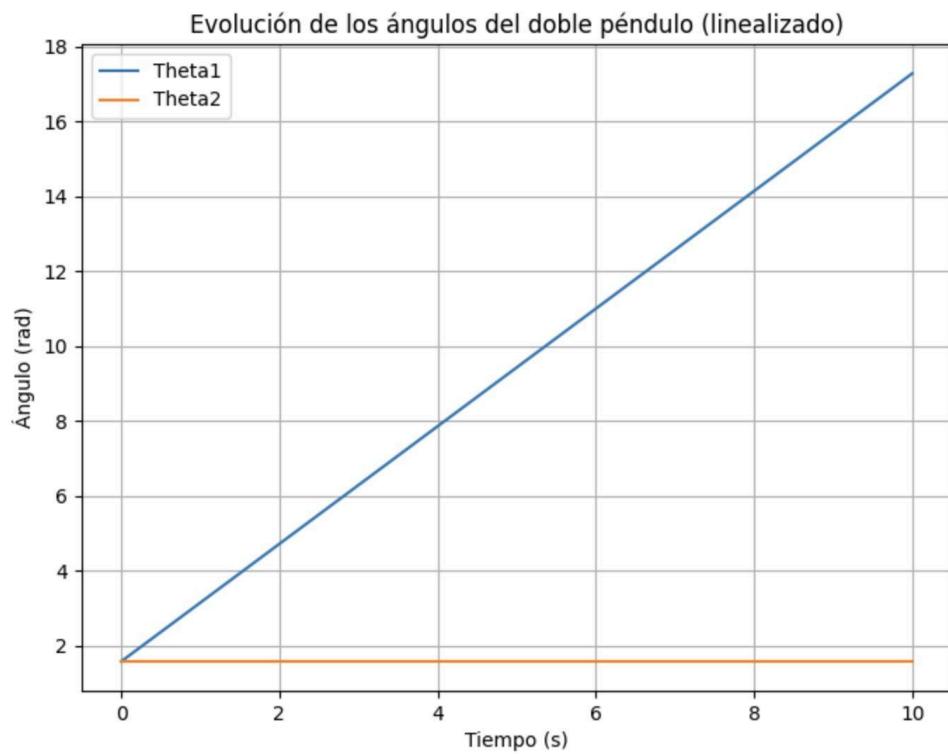
$$A = \begin{bmatrix} 0 & 1 \\ 0 & -M^{-1}C \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ M^{-1}B \end{bmatrix}$$

linealizado

## System simulation



## Linearization simulation



## Activity Dynamic Model RDK X3

System and linearization

### Modelo dinámico del robot

$$\dot{x} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \quad \ddot{x} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} v \cos \theta \\ v \sin \theta \\ \frac{v}{l} \tan \theta \end{bmatrix}$$

$v$ : Velocidad lineal  
 $l$ : Distancia entre ejes

### Espacio de estados

$$q = [x, y, \theta]^T$$

$$\dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} v \cos \theta \\ v \sin \theta \\ \frac{v}{l} \tan \theta \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 & 0 \\ \sin \theta & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ w \\ u \end{bmatrix}$$

$\checkmark$   
 $g(x)$

$\checkmark$   
 $u$

### Linearización

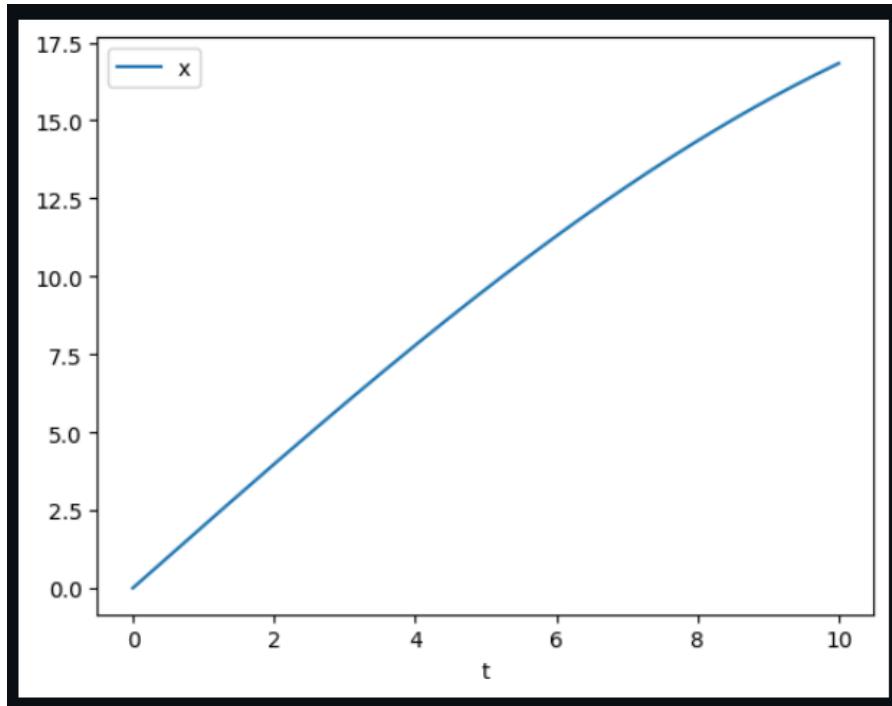
$$A = 0 \quad B = \begin{bmatrix} \frac{\partial f_1}{\partial v} & \frac{\partial f_1}{\partial w} \\ \frac{\partial f_2}{\partial v} & \frac{\partial f_2}{\partial w} \\ \frac{\partial f_3}{\partial v} & \frac{\partial f_3}{\partial w} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$q = g(x)u$$

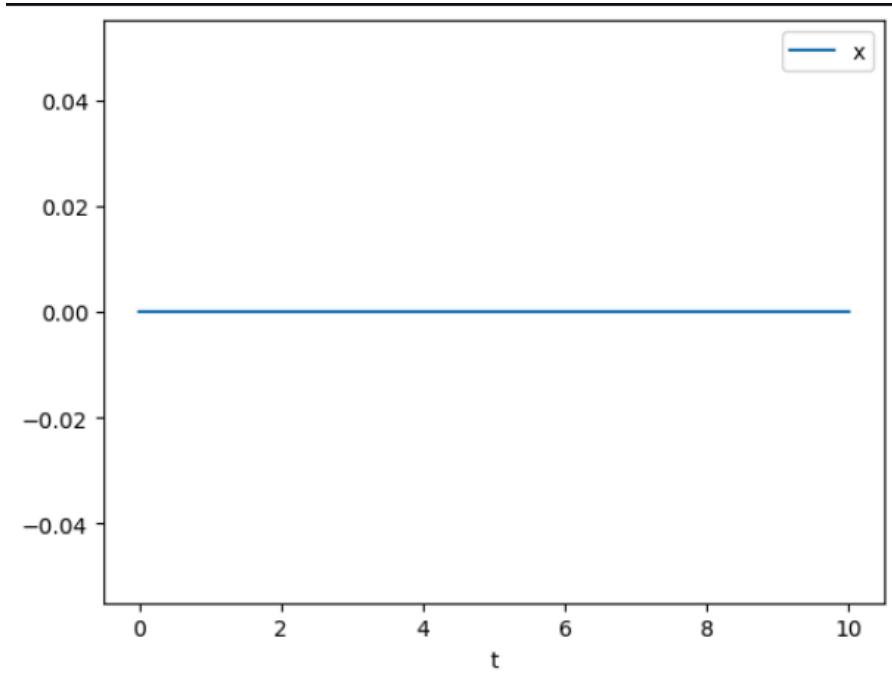
$$= \frac{q_k - q_{k-1}}{\tau} \rightarrow q_k = q_{k-1} + \int g(x)u$$

Odometría

### System simulation



### Linearization simulation



## Activity Lineal Control

$$A = \begin{bmatrix} 1 & 1 \\ 4 & -2 \end{bmatrix}$$

$$A - \lambda I = \begin{bmatrix} 1 & 1 \\ 4 & -2 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} = \begin{bmatrix} 1-\lambda & 1 \\ 4 & -2-\lambda \end{bmatrix}$$

$$\begin{aligned} \det(A - \lambda I) &= (1-\lambda)(-2-\lambda) - 4 \\ &= -2 - \lambda + 2\lambda + \lambda^2 - 4 \\ &= \lambda^2 + \lambda - 6 = (\lambda + 3)(\lambda - 2) \end{aligned}$$

$$\lambda_1 = -3 \quad \lambda_2 = 2$$

Para  $\lambda_1 = -3$

$$(A - \lambda_1 I)x = 0$$

$$\begin{bmatrix} 1-\lambda_1 & 1 \\ 4 & -2-\lambda_1 \end{bmatrix} x = 0 \Rightarrow \begin{bmatrix} 4 & 1 \\ 4 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$4x_1 + x_2 = 0$$

$$4x_1 + x_2 = 0$$

$$x_2 = -4x_1$$

$$x = s \begin{bmatrix} -\frac{1}{4} \\ 1 \end{bmatrix}, s \in \mathbb{R}$$

Para  $\lambda_2 = 2$

$$(A - \lambda_2 I)x = 0$$

$$\begin{bmatrix} 1-\lambda_2 & 1 \\ 4 & -2-\lambda_2 \end{bmatrix} x = 0 \Rightarrow \begin{bmatrix} -1 & 1 \\ 4 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

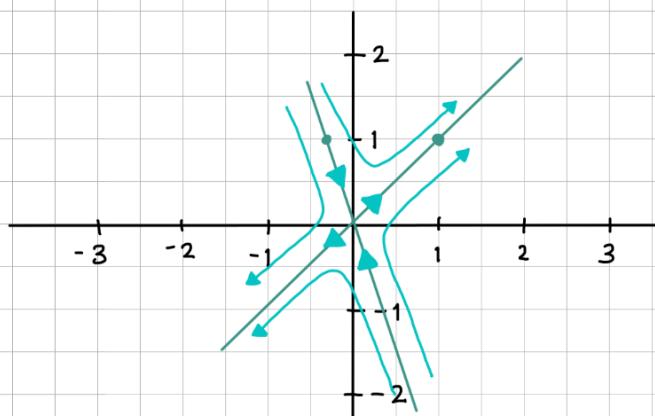
$$-x_1 + x_2 = 0$$

$$4x_1 - 4x_2 = 0$$

$$x_2 = x_1$$

$$x = s \begin{bmatrix} 1 \\ 1 \end{bmatrix}, s \in \mathbb{R}$$

→ Inestable



$$B = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

$$A - \lambda I = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} - \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} = \begin{bmatrix} 1-\lambda & 0 \\ 0 & -1-\lambda \end{bmatrix} = 0$$

$$\begin{aligned} \det(A - \lambda I) &= (1-\lambda)(-1-\lambda) - 0 \\ &= -1 - \lambda + \lambda + \lambda^2 \\ &= \lambda^2 - 1 = (\lambda - 1)(\lambda + 1) \end{aligned}$$

$\lambda_1 = 1 \quad \lambda_2 = -1$

para  $\lambda_1 = 1$

$$(A - \lambda_1 I)x = 0$$

$$\begin{bmatrix} 1-\lambda_1 & 0 \\ 0 & -1-\lambda_1 \end{bmatrix}x = 0 \Rightarrow \begin{bmatrix} 0 & 0 \\ 0 & -2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$-2x_2 = 0$$

$$x_2 = 0$$

$$x = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, s \in \mathbb{R}$$

para  $\lambda_2 = -1$

$$(A - \lambda_2 I)x = 0$$

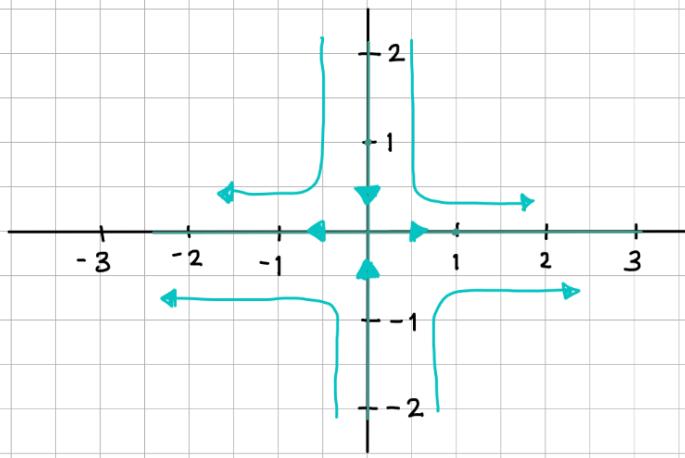
$$\begin{bmatrix} 1-\lambda_2 & 0 \\ 0 & -1-\lambda_2 \end{bmatrix}x = 0 \Rightarrow \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$2x_1 = 0$$

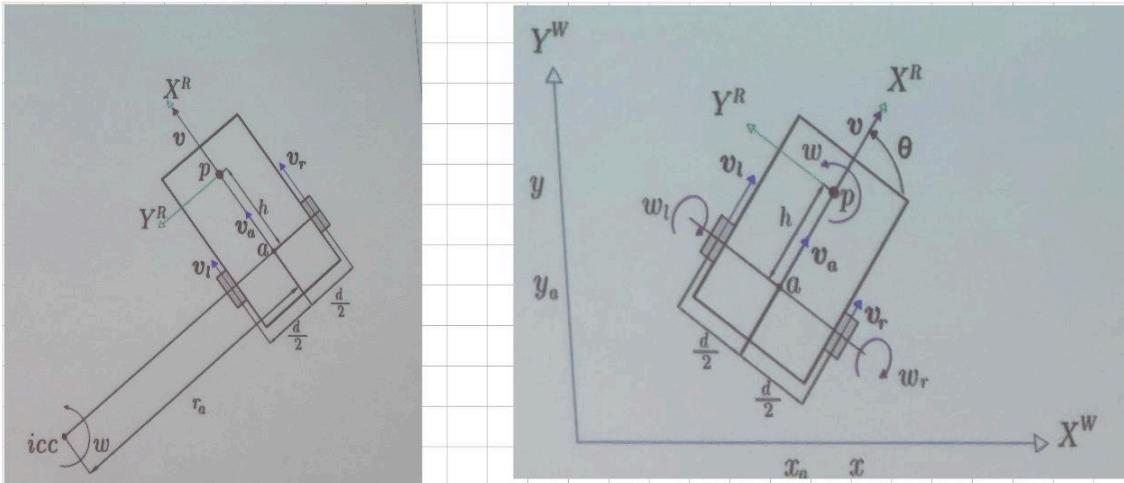
$$x_1 = 0$$

$$x = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, s \in \mathbb{R}$$

→ Inestable



## Activity Rover Feedback Control



$$x = x_a + h \cos \theta$$

$$\dot{\theta} = \omega$$

$$y = y_a + h \sin \theta$$

$$\dot{x} = \dot{x}_a - h \sin \theta \cdot \dot{\theta}$$

$$\dot{y} = \dot{y}_a + h \cos \theta \cdot \dot{\theta}$$

$$\dot{x}_a = v_a \cos \theta$$

$$\dot{y}_a = v_a \sin \theta$$

$\curvearrowleft$

$$\dot{x} = v \cos \theta - h \sin \theta \omega$$

$$\dot{y} = v \sin \theta + h \cos \theta \omega$$

$\rightarrow$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta & -h \sin \theta \\ \sin \theta & h \cos \theta \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}$$

Si  $h = 0$



$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}$$

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{1}{2} r & \frac{1}{2} r \\ \frac{r}{d} & -\frac{r}{d} \end{bmatrix} \begin{bmatrix} \omega_r \\ \omega_l \end{bmatrix}$$

r : radio de la llanta  
d : distancia entre llantas

$$U = \begin{bmatrix} v \\ w \end{bmatrix} \quad Z = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} = \begin{bmatrix} q \\ \Theta \end{bmatrix} \quad \dot{Z} = \begin{bmatrix} 0 \\ \emptyset^T \end{bmatrix} U$$

$$D = \begin{bmatrix} c\theta & -h\sin\theta \\ \sin\theta & h\cos\theta \end{bmatrix}$$

$$\emptyset^T = \begin{bmatrix} 0 & 1 \end{bmatrix}$$

Sistemas dinámicos:

$$\begin{aligned}\dot{q} &= Du \\ \dot{\Theta} &= \emptyset^T u\end{aligned}$$

Seguimiento de trayectoria:

$$e = q - q_d$$

$$\dot{e} = \dot{q} - \dot{q}_d = -Ke, \quad K > 0$$

$$= Du - q_d = -Ke$$

$$U = D^{-1} [-Ke + \dot{q}_d] \quad \leftarrow \text{Tiene que existir la inversa.}$$

\* La  $h$  de la  $D$  tiene que ser diferente de  $\emptyset$

$$A^{-1} = \frac{1}{\det A} [ ]$$

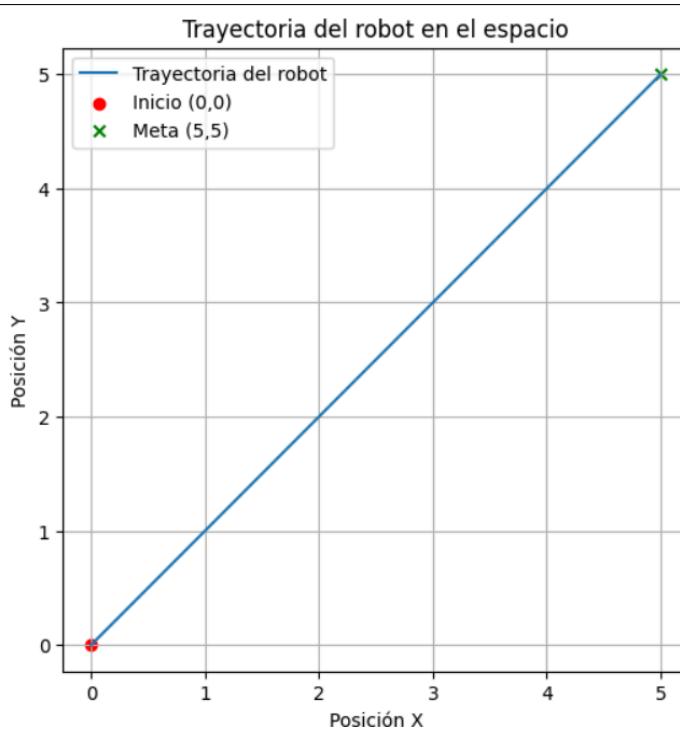
$$\det A \neq 0$$

Entonces:

$$\dot{\Theta} = \emptyset^T \cdot D^{-1} [-Ke + \dot{q}_d]$$

! No puedo controlar la orientación y la posición al mismo tiempo

## Simulation



## ROS2 Nodes

```
Windows PowerShell          root@ubuntu:~/project_ws      + - x
root@ubuntu:~/project_ws# ros2 run feedback_control control_node
[INFO] [1691668360.678086263] [Position_Node]: Actual Position: x:0.0, y
:0.0, z:0.0
[INFO] [1691668360.682619222] [Position_Node]: Actual Position: x:0.0, y
:0.0, z:0.0
[INFO] [1691668360.686798888] [Position_Node]: Actual Position: x:0.0, y
:0.0, z:0.0
[INFO] [1691668360.691103180] [Position_Node]: Actual Position: x:0.0, y
:0.0, z:0.0
[INFO] [1691668360.695368763] [Position_Node]: Actual Position: x:0.0, y
:0.0, z:0.0
[INFO] [1691668360.699582347] [Position_Node]: Actual Position: x:0.0, y
:0.0, z:0.0
[INFO] [1691668360.703677388] [Position_Node]: Actual Position: x:0.0, y
:0.0, z:0.0
[INFO] [1691668360.707923847] [Position_Node]: Actual Position: x:0.0, y
:0.0, z:0.0
[INFO] [1691668360.712045097] [Position_Node]: Actual Position: x:0.0, y
:0.0, z:0.0
[INFO] [1691668360.716149972] [Position_Node]: Actual Position: x:0.0, y
:0.0, z:0.0
|
```

```
super().__init__(executor=exec
utor)
File "/opt/tros/lib/python3.8/si
te-packages/rclpy/task.py", line 4
5, in __init__
    self._executor = None
KeyboardInterrupt
serial Close!
root@ubuntu:~/yahboomcar_ws# ros2
run yahboomcar_bringup Mcnamu_driv
er
Sunrise Robot Serial Opened! Baudr
ate=115200
-----create receive thr
eading-----
imu_link
1.0
1.0
1.0
|
```

```
23 from 192.168.8.90
root@ubuntu:~# ros2 topic list
/cmd_vel
/parameter_events
/pose
/rosout
root@ubuntu:~# ros2 run rqt_graph
rqt_graph
could not connect to display
This application failed to start b
ecause no Qt platform plugin could
be initialized. Reinstalling the
application may fix this problem.

Available platform plugins are: eg
lfs, linuxfb, minimal, minimalegl,
offscreen, vnc, xcb.
root@ubuntu:~# client_loop: send d
isconnect: Connection reset
PS C:\Users\avpv_>
|
```

ESP 17:29

```

[INFO] [1691668385.663476158] [Control_Node]: Actual Position: x:0.03649
327169782098, y:0.036755241092752085, z:0.4954185402158337
[INFO] [1691668385.668965817] [Control_Node]: Twist x: 0.085937855293018
31, Twist z: 0.5089947333796189
[INFO] [1691668385.673160984] [Control_Node]: Error:[[-0.06350673]
[-0.06324476]]
[INFO] [1691668385.677858775] [Control_Node]: Actual Position: x:0.03649
327169782098, y:0.036755241092752085, z:0.4954185402158337
[INFO] [1691668385.683126406] [Control_Node]: Twist x: 0.085937855293018
31, Twist z: 0.5089947333796189
[INFO] [1691668385.687458517] [Control_Node]: Error:[[-0.06350673]
[-0.06324476]]
[INFO] [1691668385.691977817] [Control_Node]: Actual Position: x:0.03649
327169782098, y:0.036755241092752085, z:0.4954185402158337
[INFO] [1691668385.6979646984] [Control_Node]: Twist x: 0.085937855293018
31, Twist z: 0.5089947333796189
[INFO] [1691668385.6981263658] [Control_Node]: Error:[[-0.06350673]
[-0.06324476]]
[INFO] [1691668385.6981897058] [Control_Node]: Actual Position: x:0.03655
693233105528, y:0.03681897055749953, z:0.49593460921271585
[INFO] [1691668385.6979646984] [Control_Node]: Twist x: 0.085864660450342
77, Twist z: 0.5075925146838761
[INFO] [1691668385.715771567] [Control_Node]: Error:[[-0.06344307]
[-0.06318031]]
[INFO] [1691668385.720196359] [Control_Node]: Actual Position: x:0.03655
693233105528, y:0.03681897055749953, z:0.49593460921271585
[INFO] [1691668385.725436234] [Control_Node]: Twist x: 0.085864660450342
77, Twist z: 0.5075925146838761
[INFO] [1691668385.729470525] [Control_Node]: Error:[[-0.06344307]
[-0.06318031]]
[INFO] [1691668385.732747692] [Control_Node]: Actual Position: x:0.03662
852954754721, y:0.03688263553681501, z:0.49644923859519785
[INFO] [1691668385.738855942] [Control_Node]: Twist x: 0.085791466649835
86, Twist z: 0.50619491174113587
[INFO] [1691668385.742967658] [Control_Node]: Error:[[-0.06337947]
[-0.06311736]]
[INFO] [1691668385.747337525] [Control_Node]: Actual Position: x:0.03662
852954754721, y:0.03688263553681501, z:0.49644923859519785
[INFO] [1691668385.753605775] [Control_Node]: Twist x: 0.085791466649835
86, Twist z: 0.50619491174113587
[INFO] [1691668385.758965990] [Control_Node]: Error:[[-0.06337947]
[-0.06311736]]
[INFO] [1691668385.764538775] [Control_Node]: Actual Position: x:0.03662
852954754721, y:0.03688263553681501, z:0.49644923859519785

```

Windows PowerShell  
Copyright (C) Microsoft Corporation. All rights reserved.  
Instale la versión más reciente de PowerShell para obtener nuevas características y mejoras. <https://aka.ms/PSWindows>

PS C:\Users\avpv-> ssh root@192.168.8.88  
root@192.168.8.88's password:  
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 4.14.57~19.101.1 arch64)  
\* Documentation: <https://help.ubuntu.com>  
\* Management: <https://landscape.canonical.com>  
\* Support: <https://ubuntu.com/advantage>

## Vision Module

### Deep Learning

Multilayer neural networks are used to model and solve complex problems. A piece of data enters a neuron multiplied by a weight indicating the importance of the data. All inputs are summed, and then an activation function is applied.

### Training

- Forward propagation: Inputs pass through the network to produce an output.
- Error calculation: Compares the predicted output with the actual value.
- Backpropagation: Adjusts weights using gradient descent to minimize error.

### Types of Neural Layers

- Input: Defines the shape of the data.
- Dense: Full connections for global learning.
- Convolutional: Extracts local features.
- Pooling: Reduces spatial dimensionality.
- Dropout: To avoid overfitting.
- Output: Generates the prediction.