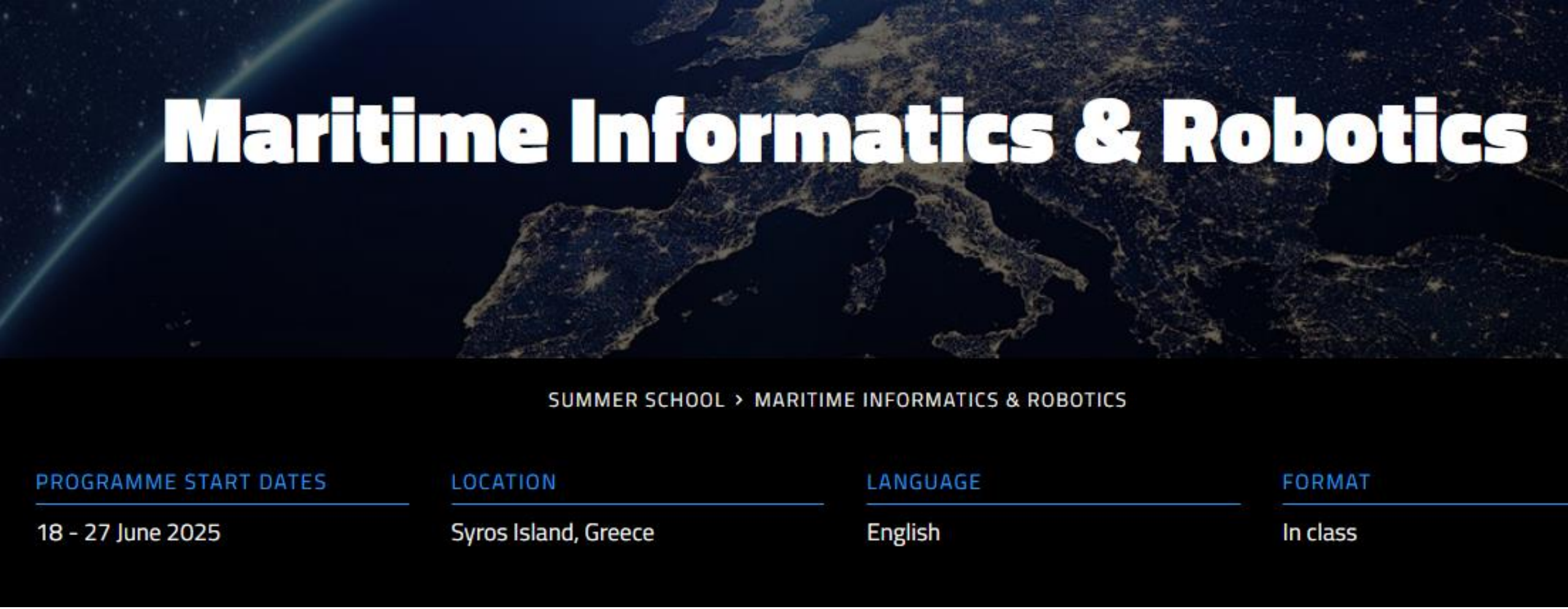


Robotic Motion Control

Xanthi S. Papageorgiou

Assistant Professor, University of the Aegean

UNIVERSITY OF THE AEGEAN

In this Lecture

- Introduction
- Motion Models
- Forward Kinematics, Inverse Kinematics
- Trajectory Tracking
- Open-loop vs Closed-loop Control
- Introduction to PID control
- Introduction to Model Predictive Control (MPC)
- Introduction to Marine Craft Model and Control

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Introduction

- **What is a robot?**
  - First defined by Czech writer Karel Čapek in his play R.U.R. (Rossum's Universal Robots) in 1920

Introduction

- **What is a robot?**
  - First defined by Czech writer Karel Čapek in his play R.U.R. (Rossum's Universal Robots) in 1920
- **Many alternative definitions but common features:**
  - Should be able to “move” (i.e. do physical work)
  - Should be programmable
  - “Intelligence” and autonomy are optional but desirable

Motion Planning and Control

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Motion Planning and Control

- **Motion planning:**
  - Where should I go and what motions should I go through?
  - Decides "what path" the robot should take to reach a goal while avoiding obstacles

  ✓Generates the trajectory (offline/online)
- **Motion control:**
  - How do I perform the desired motions?
  - Determines "how to follow" that path by computing real-time actuator commands

  ✓Executes the trajectory accurately (real-time)

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

What did we assume?

- **Perfect sensors?**
  - What information is available?
  - Uncertainty?
- **Perfect control?**
  - What controls are available?
  - Uncertainty?

Motion Planning and Control

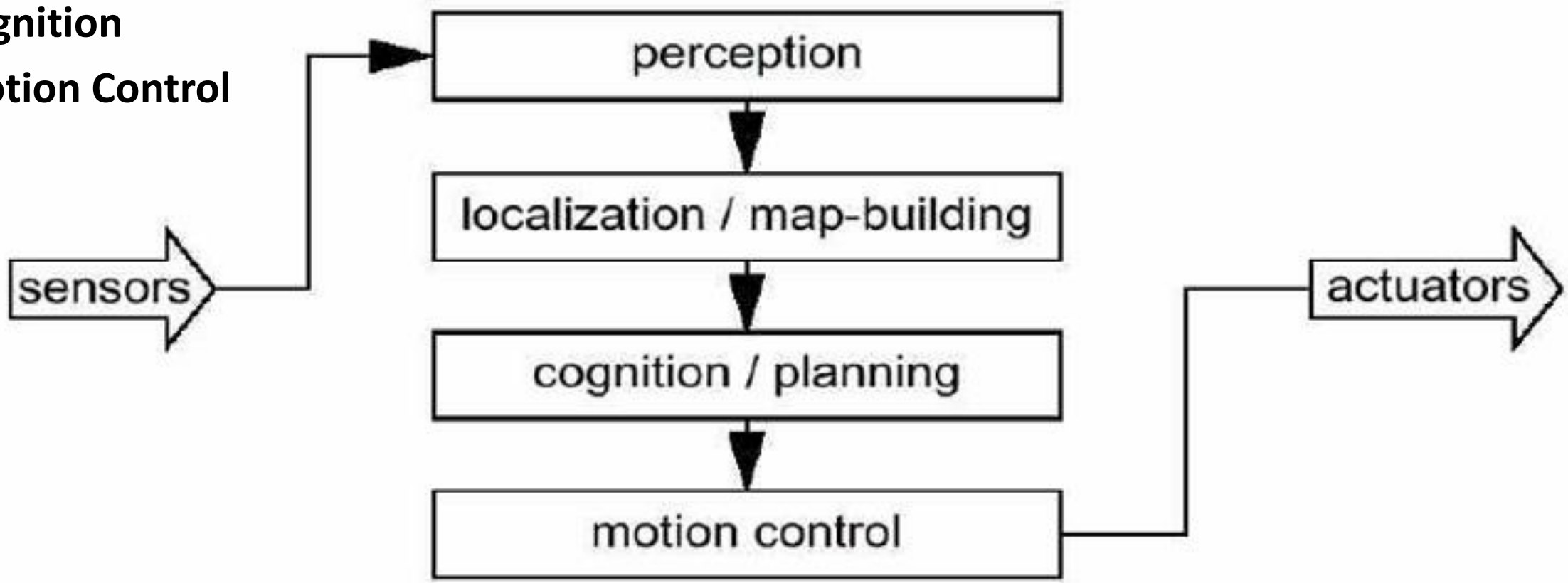
- Through **perception**, a model of the “real” world is captured in memory
- A goal is given, and a **plan** is generated, assuming the “real” world is not changing
- Then, the plan is **executed**, one (abstract) operation at a time

- What is “interesting” in the “real” world to be captured?
- At what level of details should we represent the “real” world?
  - The “real” world model must be at all times accurate (consistent and reliable)
- What if during plan execution, the “real” world changes?
  - Sudden changes in the world may not be reflected instantly in our model

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Robot Navigation

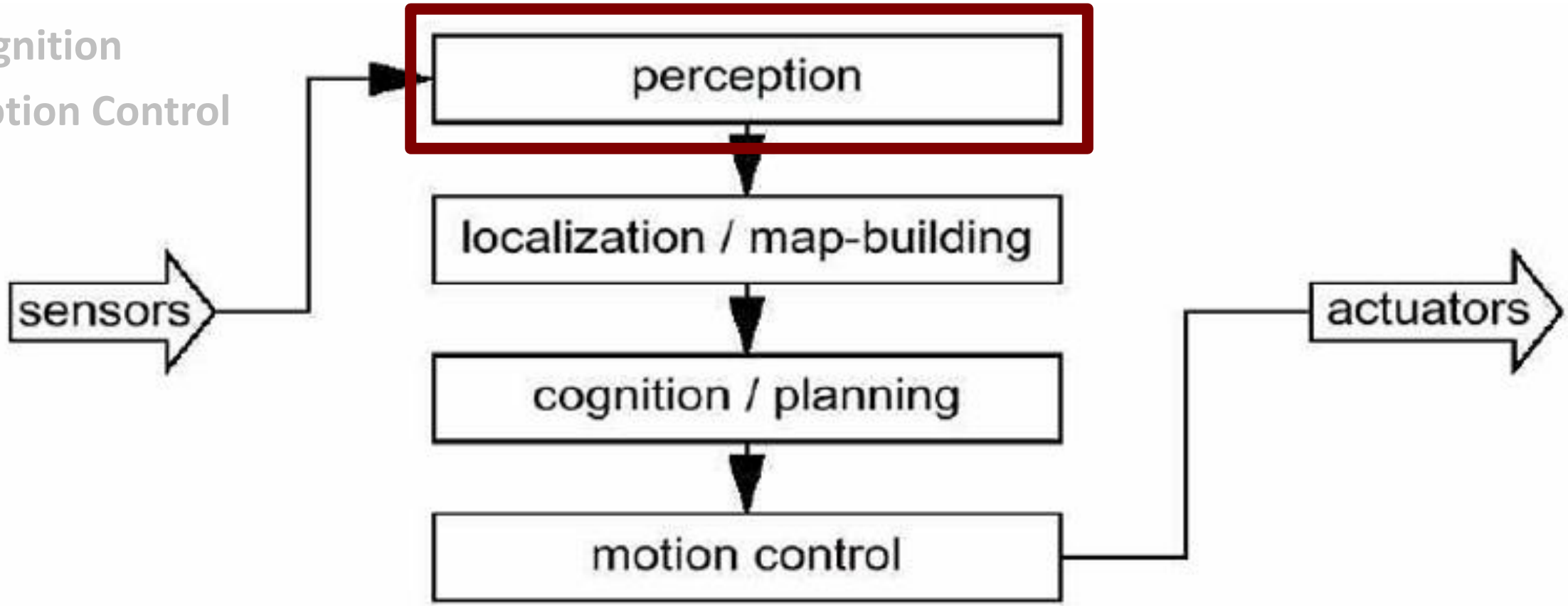
- Perception
- Localization
- Cognition
- Motion Control



Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Robot Navigation

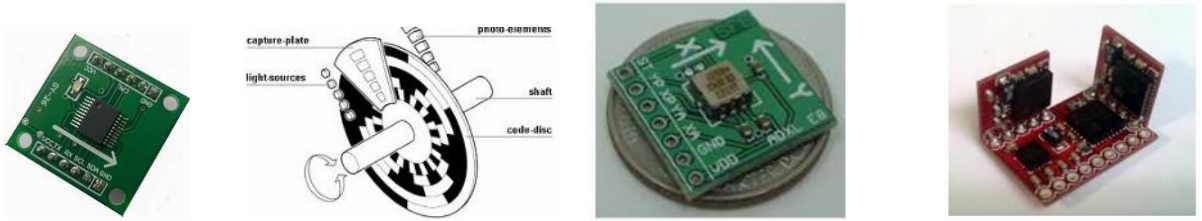
- **Perception**
- Localization
- Cognition
- Motion Control



Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

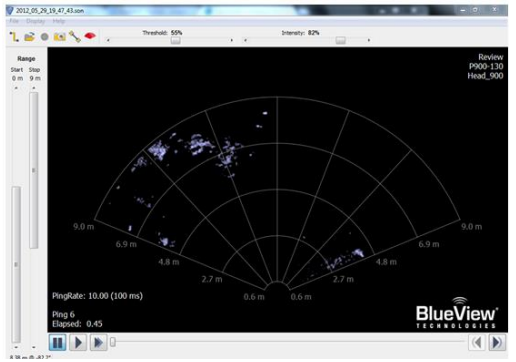
Robot Navigation - Perception

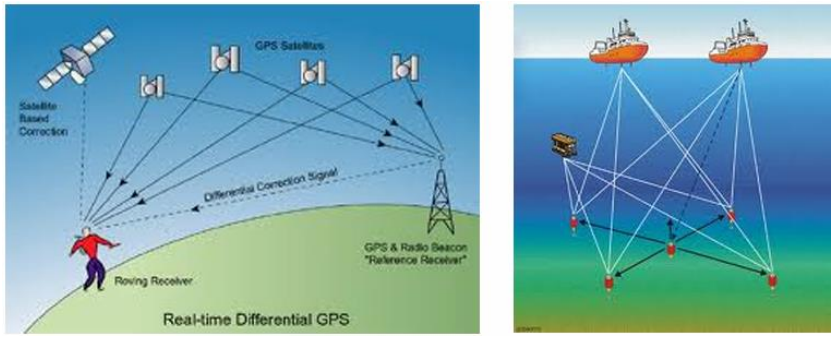
- **Proprioceptive**
  - Compass
  - Encoders
  - Accelerometers
  - IMU – Inertial Measurement Unit
- **Exteroceptive**
  - Range Sensors
  - Vision Systems
  - Sonar
  - Positioning Systems (e.g. GPS)



Images from elechouse.com, hightechsa.blogspot.com inmotion.pt, xkyle.com

Images from nauticexpo.com, acroname.com, robots.com, www8.garmin.com



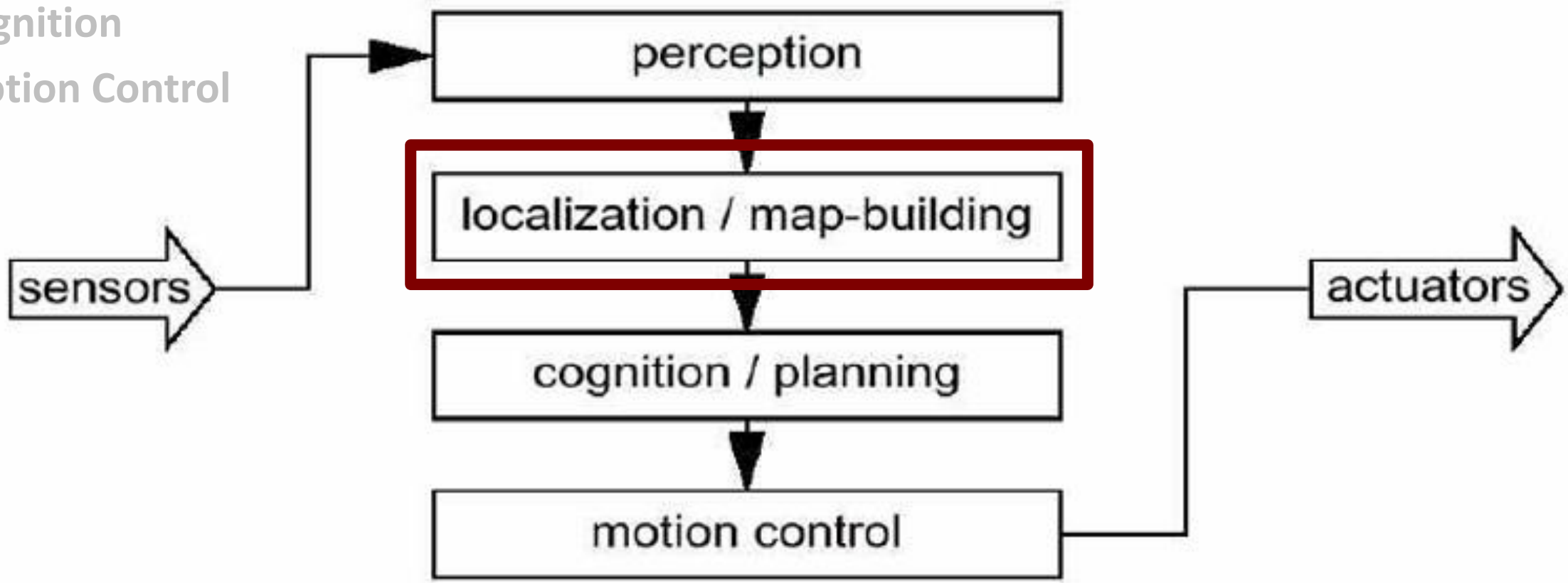


Images from 66oaudio.com, km.kongsberg.com

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Robot Navigation

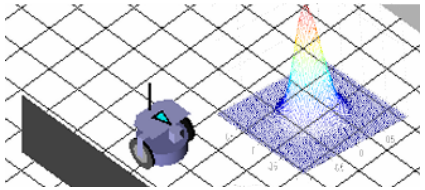
- Perception
- **Localization**
- Cognition
- Motion Control

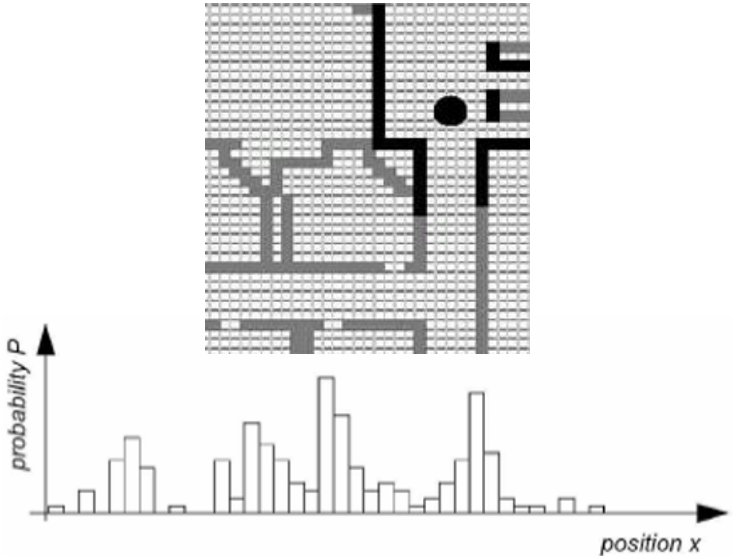


Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Robot Navigation - Localization

- **Probabilistic Representations**
  - Continuous
  - Discrete



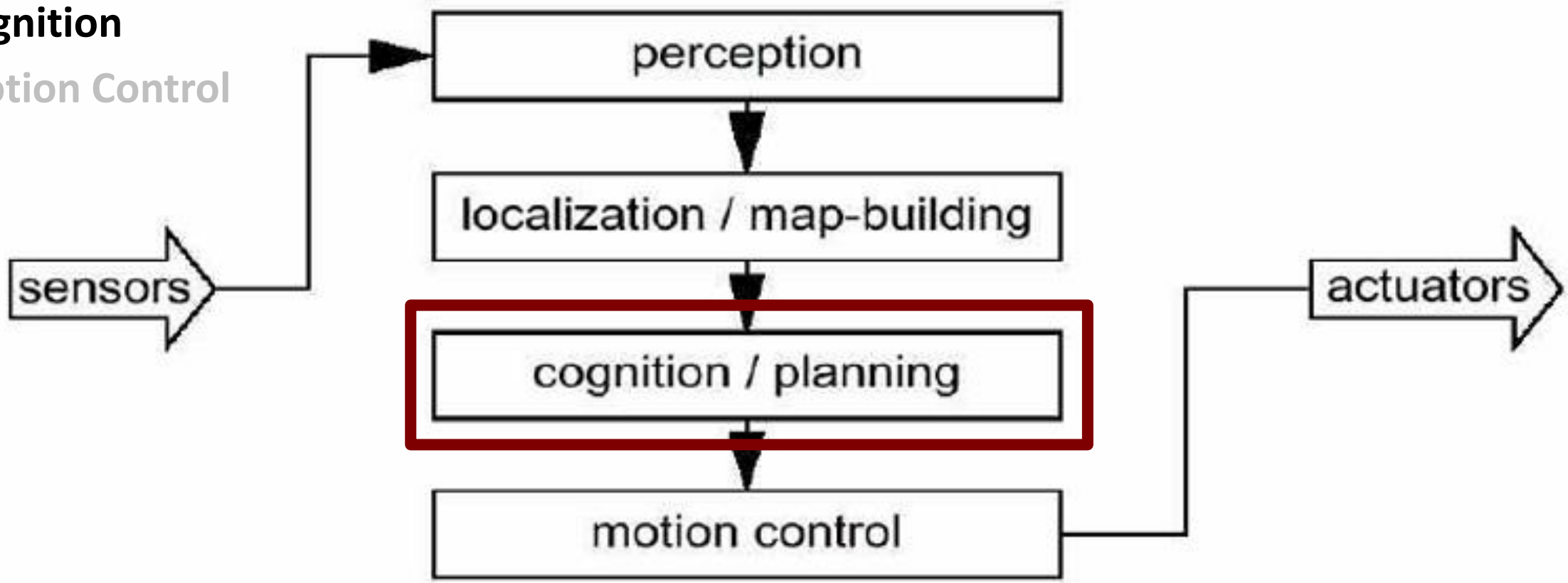


- **Probabilistic Algorithms**
  - Kalman Filter Based
    - Assumes Gaussian representation of robot state
    - Compact representation good for real time implementation
  - Particle Filter Based
    - Uses many particles to represent robot state, each particle is an estimate of the robot position with an associated weight

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Robot Navigation

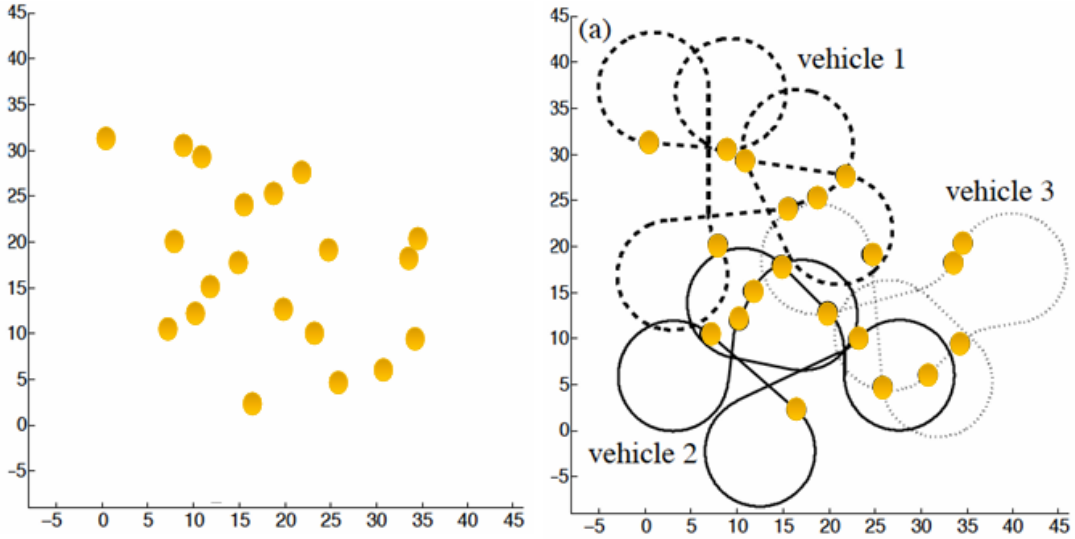
- Perception
- Localization
- **Cognition**
- Motion Control



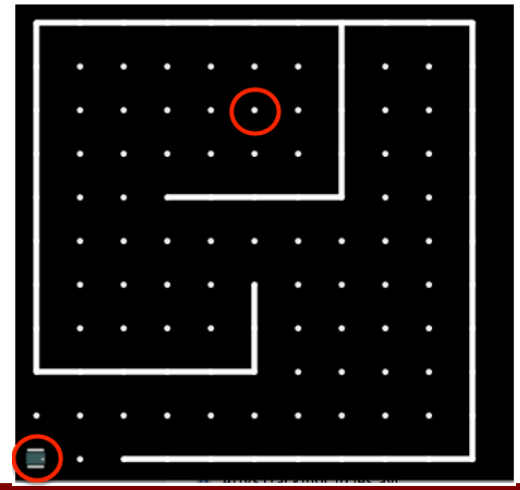
Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Robot Navigation - Cognition

- **Task Planning**
  - Given a set of tasks (e.g. task locations), identify ordering sequence for the tasks



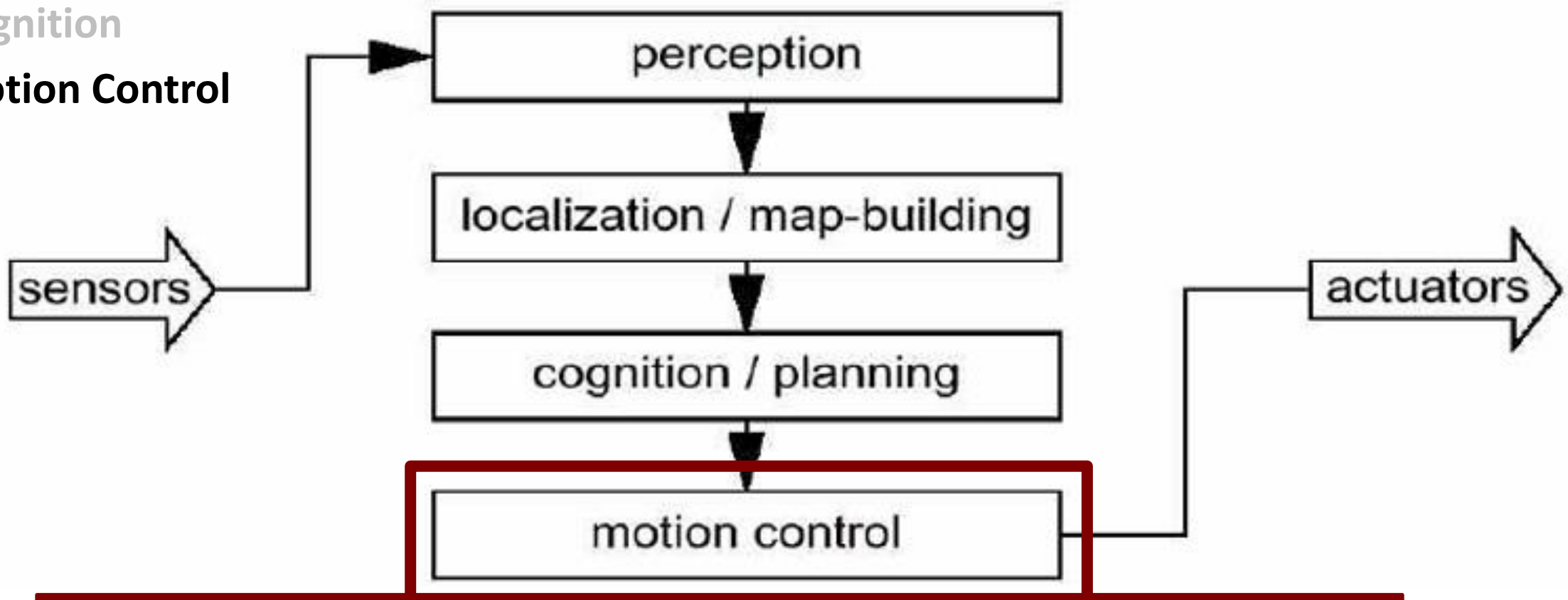
- **Motion Planning**
  - Given a robot's Start Configuration and Goal Configuration, construct a collision free trajectory from Start to Goal



Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Robot Navigation

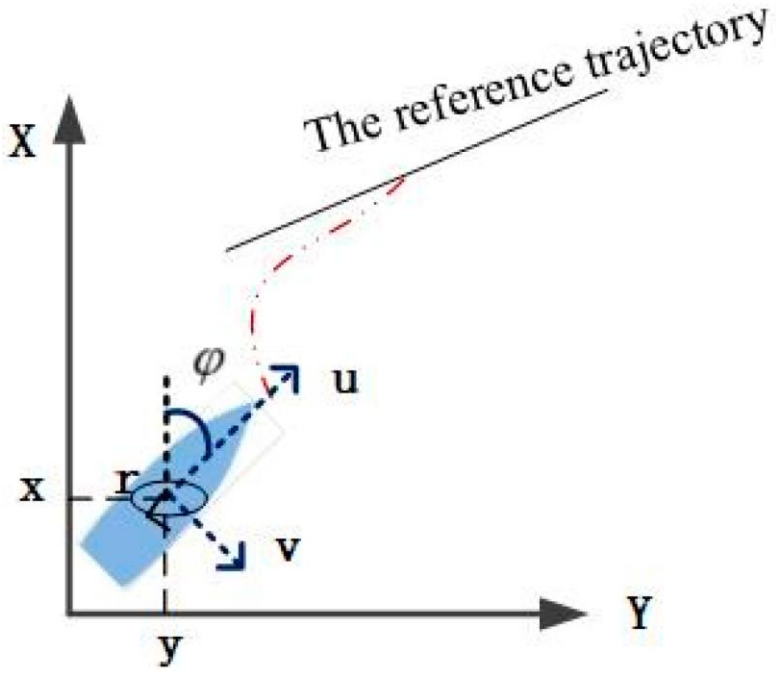
- Perception
- Localization
- Cognition
- **Motion Control**

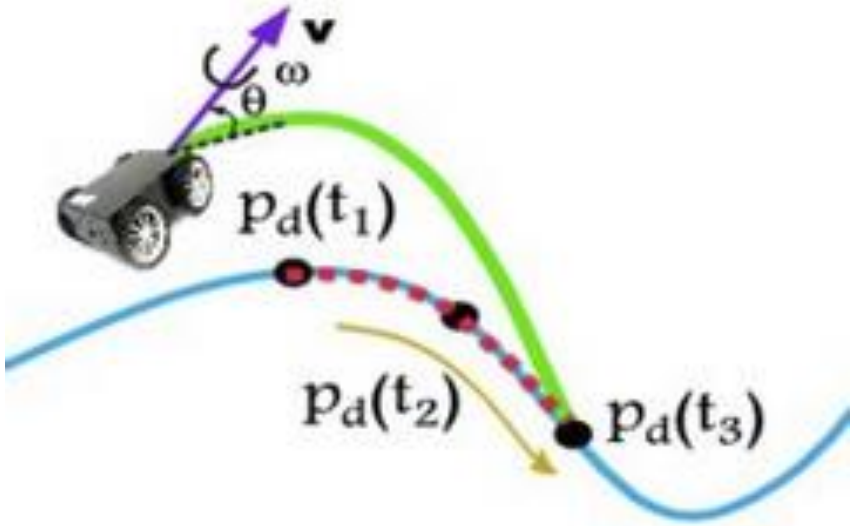


Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Robot Navigation - Motion Control

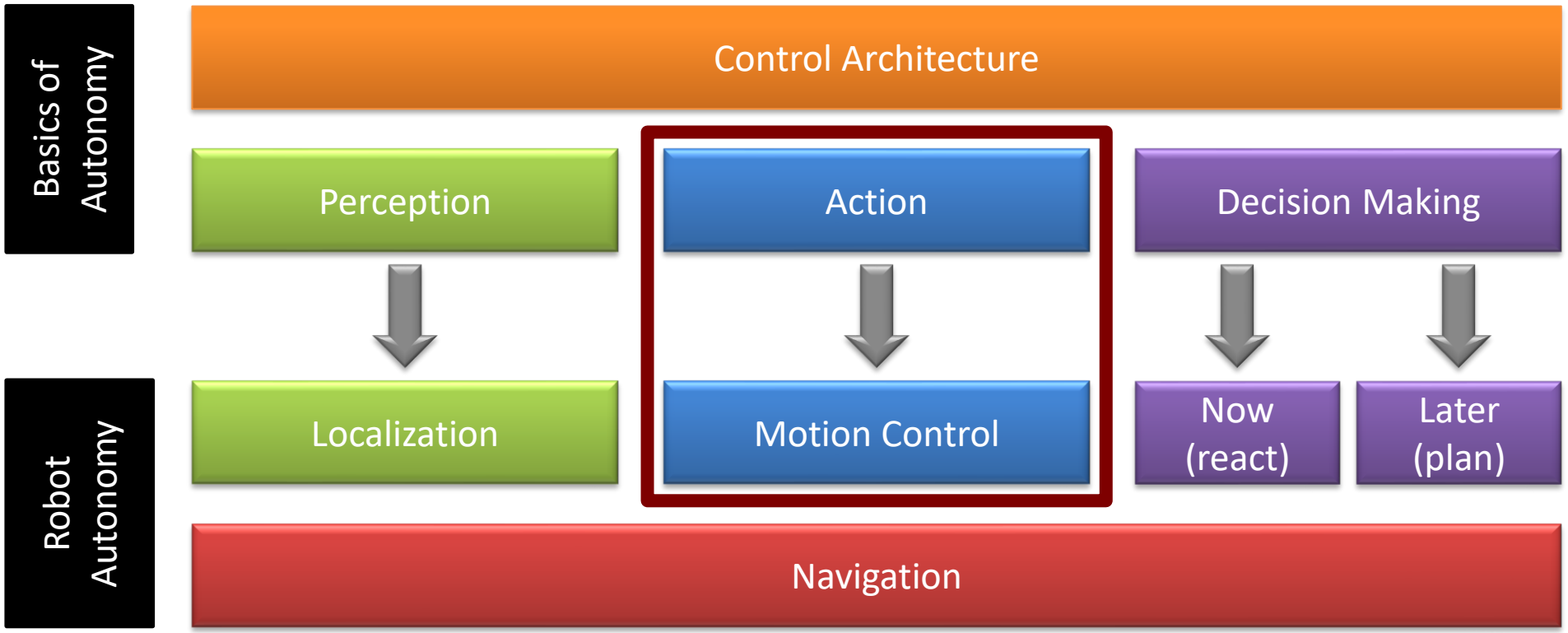
- **Trajectory Tracking Control**
  - Given a trajectory, determine the control signals sent to actuators that guarantee the robot will follow the trajectory





Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

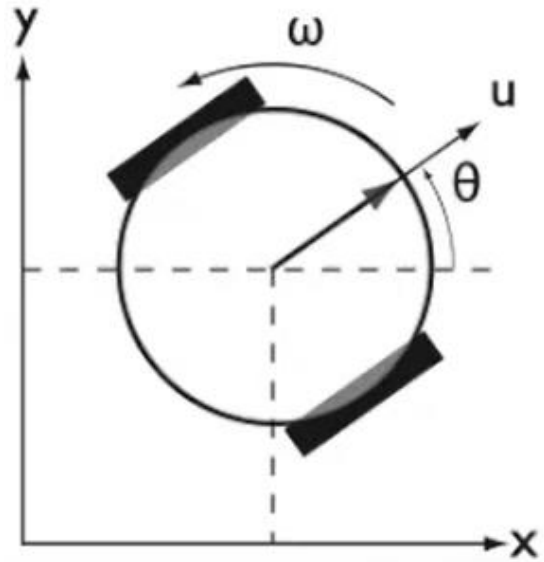
Control Architecture



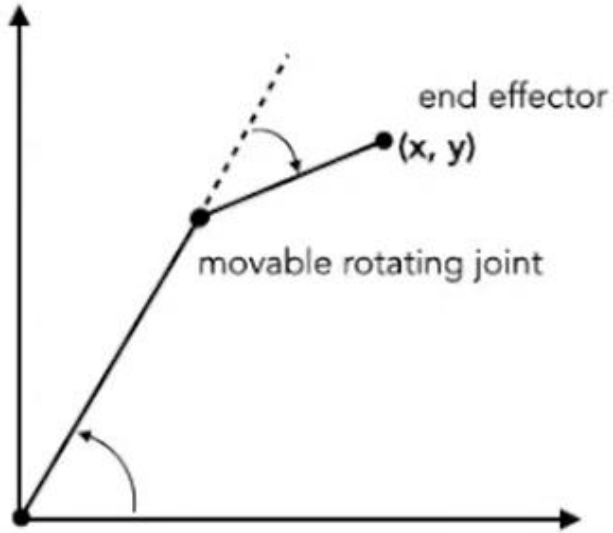
Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Degrees of Freedom

- Most actuators control a single **degree of freedom** (DOF)
- Every robot has a specific number of DOF
- If there is an actuator for every DOF, then all SOF are controllable



differential-drive robot



fixed rotating joint

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

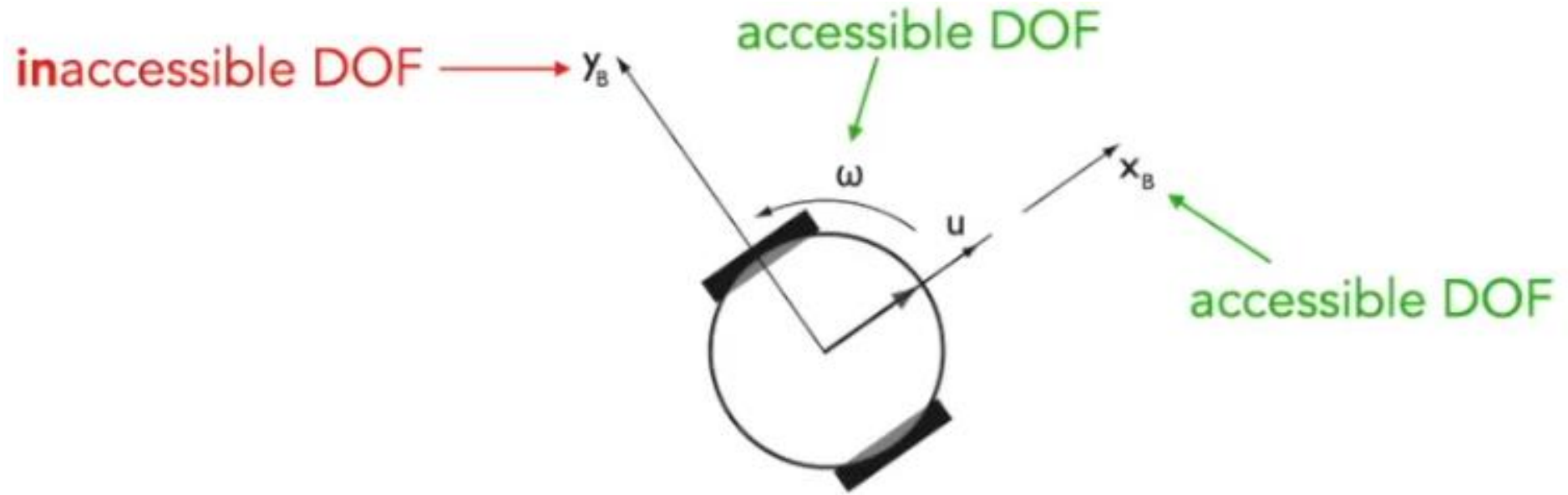
Holonomic Motion

- **Degree of mobility: DOM (differentiable DOF)**
  - Number of DOF that can be **directly accessed** by the actuators
  - A robot in the plane has at most 3 DOMs (position and heading)
- **Holonomic motion:**
  - **Holonomic robot**:
    - When the number of DOF is equal to robot's DOM
  - **Non-holonomic robot**:
    - When the number of DOF is greater than robot's DOM
  - When a robot's DOM it is larger than is DOF, the robot has **"redundant"** actuation

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Differential-Drive Robot

- Differential-drive robots can actuate left and right wheels (independently)



- DOF = 3, but DOM = 2: differential-drive robots are **non-holonomic**
- Impact of non-holonomicity: **motion constraint affect motion planning**

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Actuators

- Different purposes
  - Locomotion: e.g., wheeled, legged
  - Other motion: e.g., manipulation
  - Other type of actuation: e.g., heating, sound emission
- Examples of electrical-to-mechanical actuators:
  - DC motors, stepper motors, servo
  - **Control input** example: A driver can steer and accelerate/decelerate, so there are 2 control inputs
- Uncertainty/disturbance/noise:
  - Wheel slip, slack in mechanism, environmental factors (wind, friction, etc.)

Kinematics

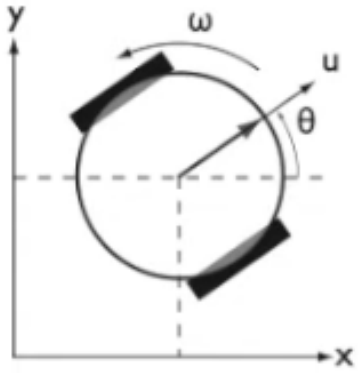
- **Forward** kinematics:
  - Given the control parameters (e.g., wheel velocities), and the time of movement *t*, **find the pose** (*x,y,ϑ*) reached by the robots

- **Inverse** kinematics:
  - Given the final desired pose (x,y,θ), **find the control parameters** to move the robot there at a given time *t*

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

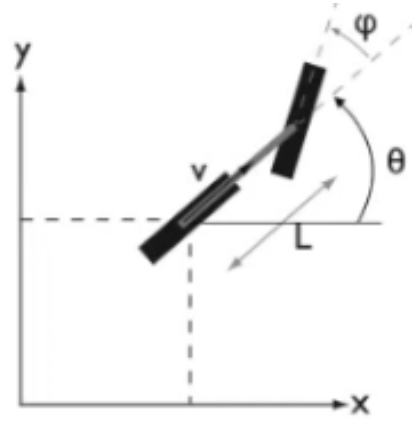
Forward Kinematics

- Differential equations describe robot motion
- How does robot state change over time as a function of control inputs?



$$\begin{cases} \dot{x} &= u \cdot \cos\theta \\ \dot{y} &= u \cdot \sin\theta \\ \dot{\theta} &= \omega \end{cases}$$

differential-drive model
3 DOF (2 controllable)



$$\begin{cases} \dot{x} &= v \cdot \cos\theta \\ \dot{y} &= v \cdot \sin\theta \\ \dot{\theta} &= v \cdot \frac{\tan\phi}{L} \end{cases}$$

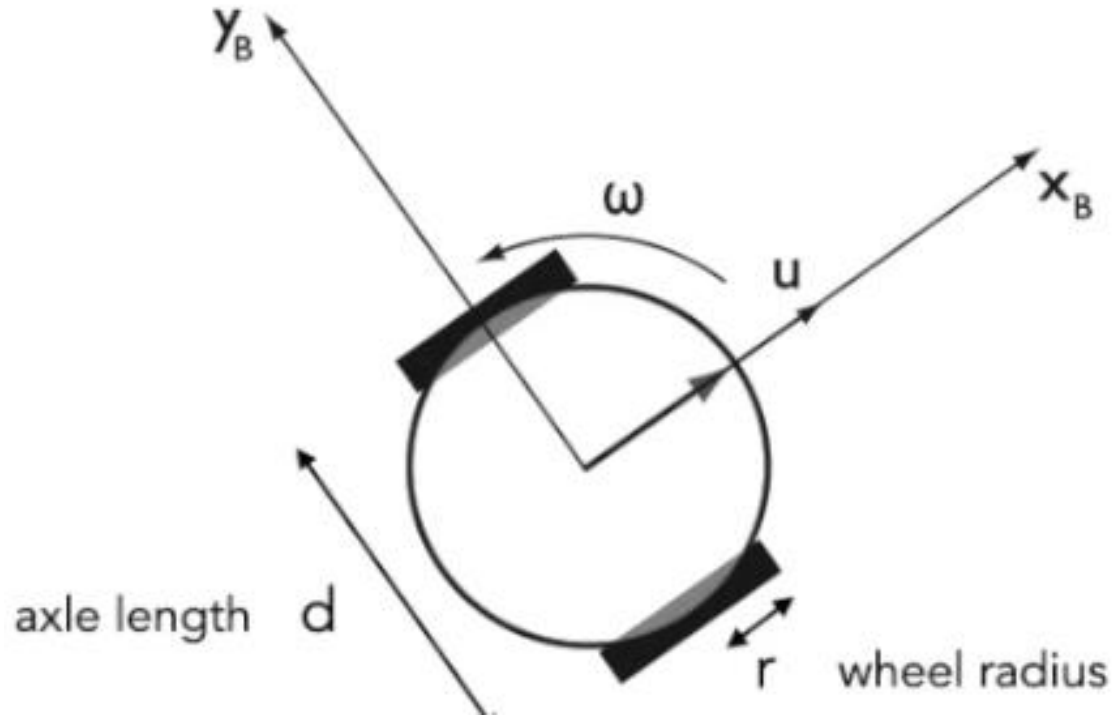
bicycle model
3 DOF (2 controllable)

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Forward Kinematics (body frame)

Actuators of differential-drive:

- Left wheel speed $\dot{\phi}_l$
- Right wheel speed $\dot{\phi}_r$



Forward velocity:

$$u = \frac{r\dot{\phi}_r}{2} + \frac{r\dot{\phi}_l}{2}$$

Rotational velocity:

$$\omega = \frac{r\dot{\phi}_r}{d} - \frac{r\dot{\phi}_l}{d}$$

Motion:

$$\dot{x}_B = u$$
$$\dot{y}_B = 0$$
$$\dot{\theta}_B = \omega$$

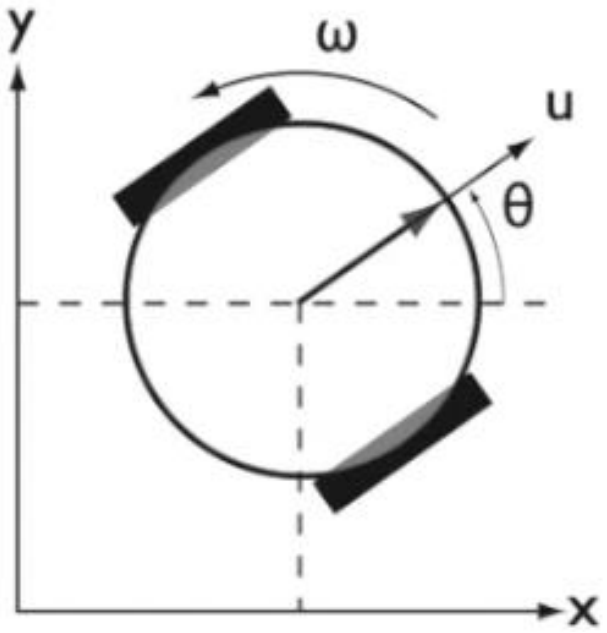
Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Forward Kinematics (world frame)

- Given known control inputs, how does the robot move w.r.t. a **global coordinate system**?
- Use a **rotation matrix**:
  - From body to world frames, the axes rotate by θ

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \underbrace{\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{T(\theta)} \begin{bmatrix} \dot{x}_B \\ \dot{y}_B \\ \dot{\theta}_B \end{bmatrix}$$

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



Inverse Kinematics

- We would like to control the robot motion in the world frame: $\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix}$
- We **invert** the previous equations to **find control inputs**:

$$\begin{bmatrix} u \\ 0 \\ \omega \end{bmatrix} = T^{-1}(\theta) \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix}$$

- yielding

$$\begin{aligned} u &= \dot{x}\cos\theta + \dot{y}\sin\theta \\ \omega &= \dot{\theta} \end{aligned}$$

- and finally

$$\begin{aligned} \dot{\phi}_l &= u - \frac{\omega d}{2r} \\ \dot{\phi}_r &= u + \frac{\omega d}{2r} \end{aligned} \quad \Longrightarrow \quad \begin{aligned} \dot{\phi}_l &= \dot{x}\cos\theta + \dot{y}\sin\theta - \frac{\dot{\theta} d}{2r} \\ \dot{\phi}_r &= \dot{x}\cos\theta + \dot{y}\sin\theta + \frac{\dot{\theta} d}{2r} \end{aligned}$$

we can now control the wheel speeds

- BUT: under the **constraint** (because this robot is non-holonomic)

$$\dot{x}\sin\theta = \dot{y}\cos\theta$$

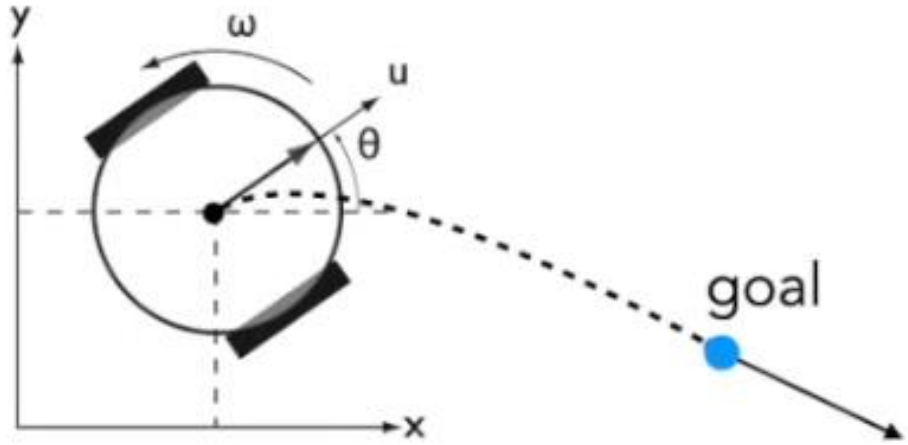
Inverse Kinematics

- We would like to control the robot to reach a goal pose: $\begin{bmatrix} x_G \\ y_G \\ \theta_G \end{bmatrix}$

- Ideally (if the robot would be holonomic), we would set:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = K \begin{bmatrix} x_G - x \\ y_G - y \\ \theta_G - \theta \end{bmatrix}$$

control gain



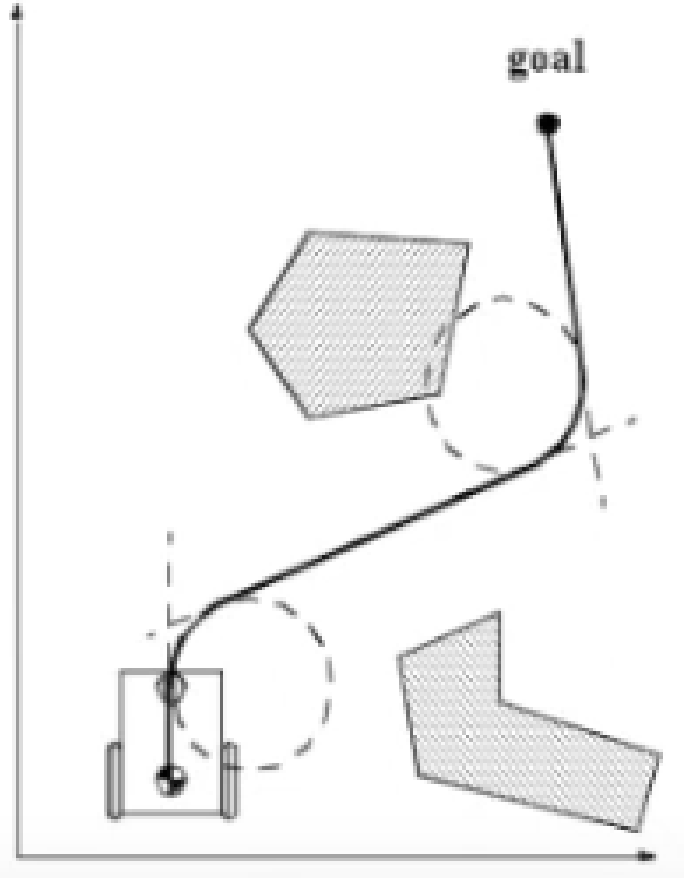
- However, we need to satisfy the non-holonomic constraint:

$$\dot{x} \sin\theta = \dot{y} \cos\theta$$

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Trajectory Tracking

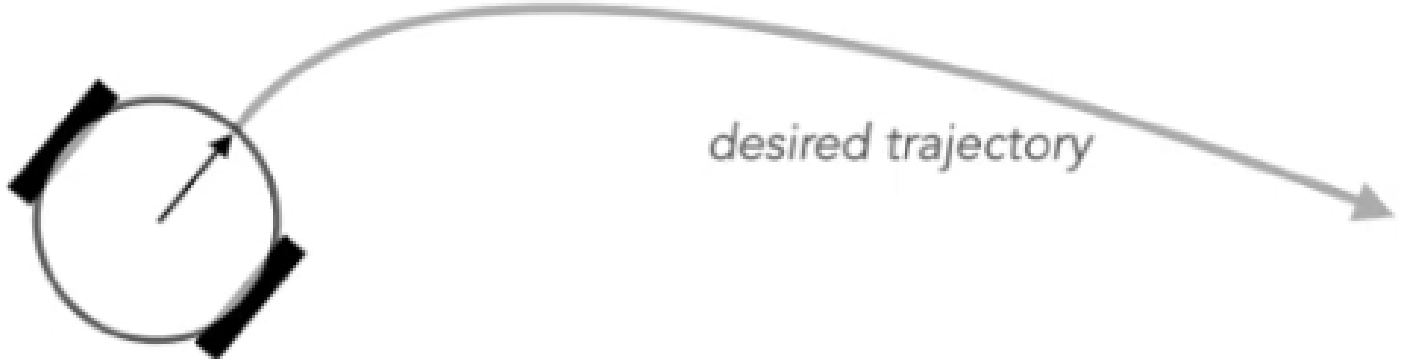
- Trajectory Tracking:
  - Pre-compute a smooth trajectory
  - Follow trajectory (in open-loop or closed-loop)
- Challenges
  - Feasibility of trajectory given motion constraints
  - Adaptation of trajectory in dynamical environments
  - Must guarantee smoothness of resulting trajectories (kinematic / dynamic feasibility): e.g., continuity of 1st derivative for 1st order control



Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

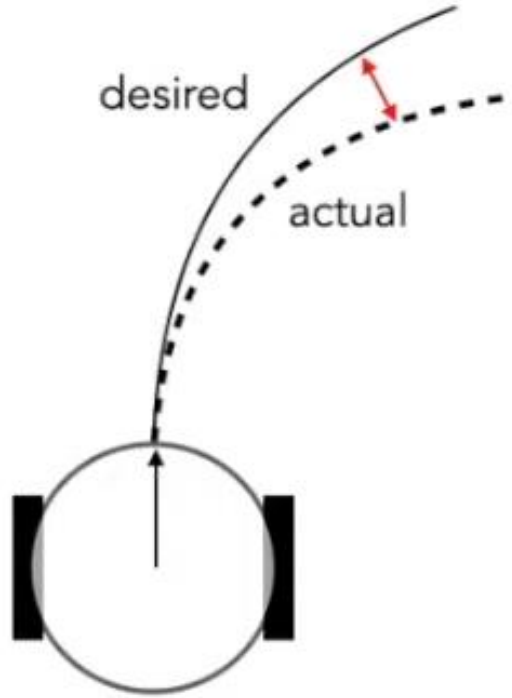
Open-Loop vs Closed-Loop

- Once we have a trajectory that enables the robot to reach its goal, we need to follow that trajectory
- There are two ways of doing this:
  - **Open-loop control:** Robot follows path blindly by applying the pre-computed control input
  - **Closed-loop control:** Robot can follow path for a small duration, then observe if anything changed in the world, recompute a new adapted path (repeatedly)



Open-Loop Controller

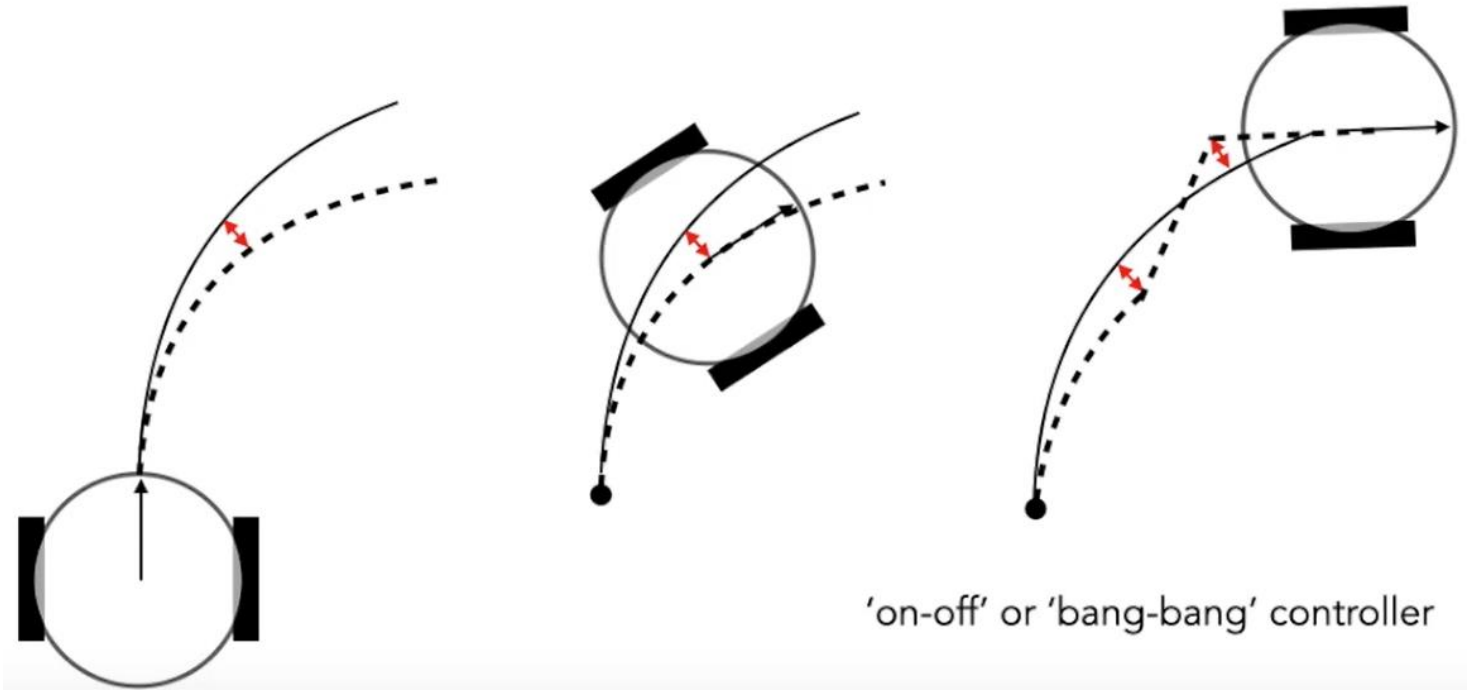
- Example: trajectory tracking
- In open-loop, the robot executes predefined control inputs



- Under imperfect conditions, the robot deviates from desires behavior

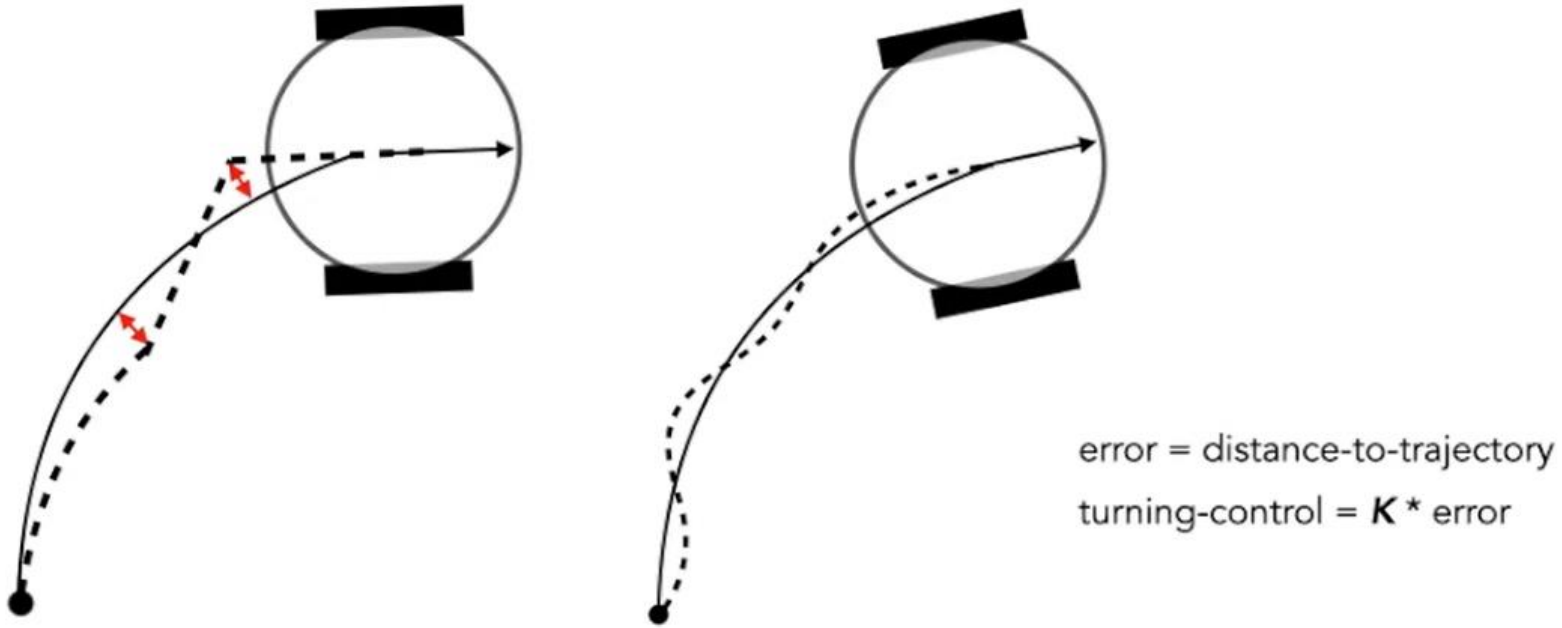
Closed-Loop Controller

- Example: trajectory tracking
- The robot uses **feedback** to maintain a desired set-point
- Assumption: robot receives **feedback** on distance to desired trajectory



Proportional Control (P-Control)

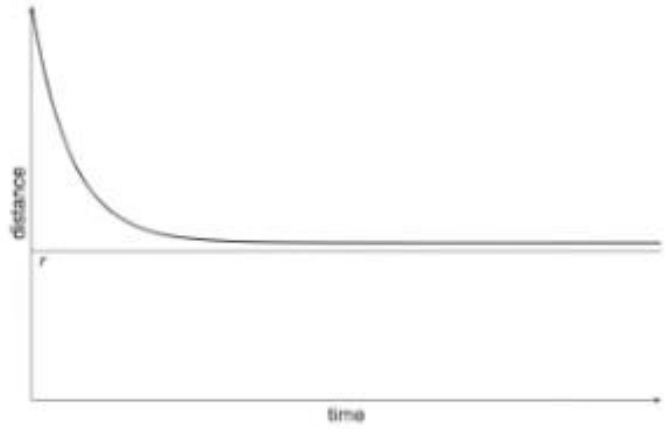
- Example: trajectory tracking
- The robot uses **feedback** to maintain a desired set-point
- Robot computes error, and adjust control **as a function of error**



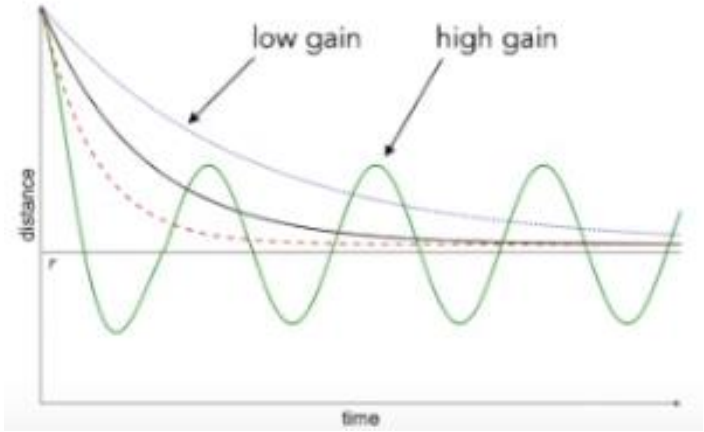
Proportional Control (P-Control)

- Behavior of P-controller:
  - Adapt control proportionally to your perceived error to set-point

$$u(t) = \kappa_p e(t)$$



- Behavior for varying gain values
- High gains not desirable! We call this an **unstable** controller



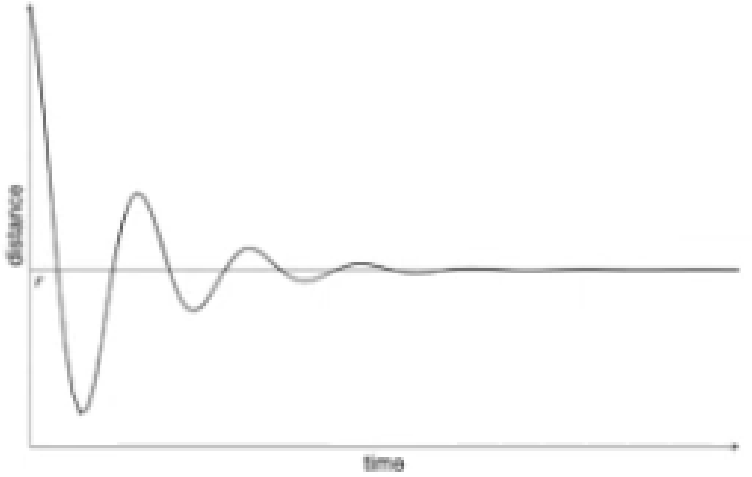
Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

PID Control

- PI-controller:
  - takes into account **accumulated error** over time

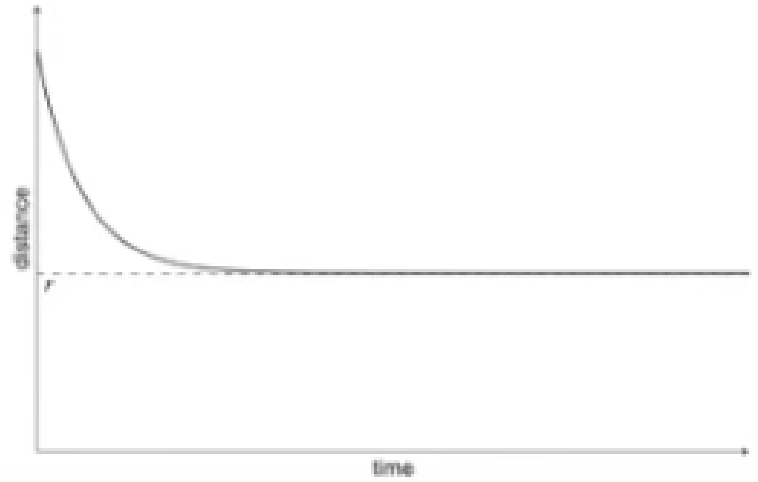
$$u(t) = \kappa_p e(t) + \kappa_i \int_0^t e(\tau)\, d\tau$$

  - e.g., in presence of friction, error will be integrated causing higher motor setting to overcome remaining delta



- PID-controller:
  - take into account **future error** by computing rate of error's change
  - act as a "dampener" on control effort

$$u(t) = \kappa_p e(t) + \kappa_i \int_0^t e(\tau)\, d\tau + \kappa_d \frac{de(t)}{dt}$$



Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Open-Loop vs Closed-Loop

- **Closed-loop** is much more robust to external perturbation:
  - Noisy sensors: wrong estimate of the goal position, wrong estimate of the robot position
  - Noisy actuation: robot does not move precisely
  - Unforeseen events, dynamic obstacles
- **Open-loop** is only useful when feedback is not possible:
  - Sensors cannot operate in certain circumstances
  - Limited bandwidth
  - Limited computational resources

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Model Predictive Control

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Model Predictive Control

- **Model Predictive Control (MPC)** is a control scheme where a model is used for predicting the future behavior of the system over finite time window, the horizon
- Based on these predictions and the current measured/estimated state of the system, the optimal control inputs with respect to a defined control objective and subject to system constraints is computed
- After a certain time interval, the measurement, estimation and computation process is repeated with a shifted horizon

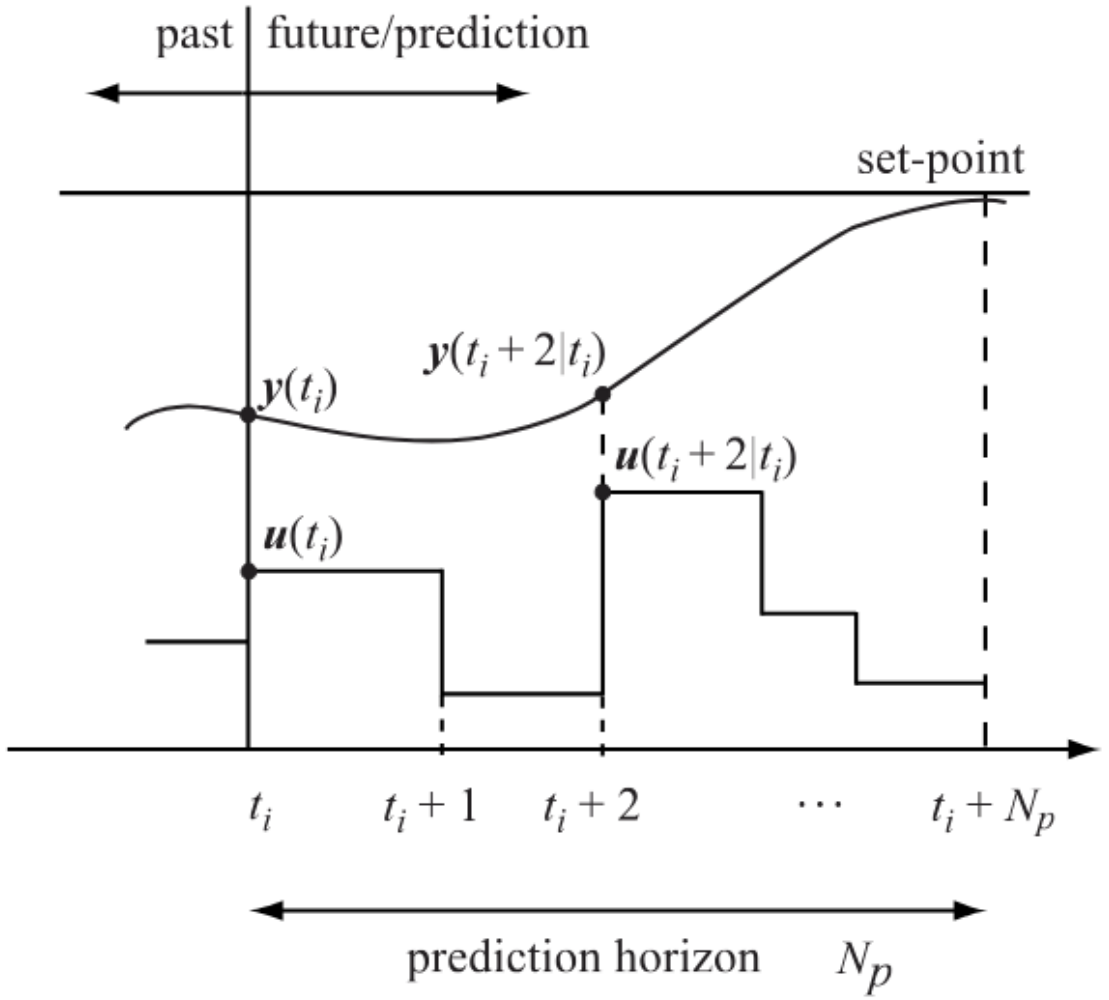
Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Model Predictive Control

- In the MPC approach, the current control action is computed on-line rather than using a pre-computed, off-line, control law
- A model predictive controller uses, at each sampling instant, the plant's current input and output measurements, the plant's current state, and the plant's model to
  - calculate, over a finite horizon, a future control sequence that optimizes a given performance index and satisfies constraints on the control action
  - use the first control in the sequence as the plant's input

Model Predictive Control

- N_p is the prediction horizon
- u(t + k|t) is the predicted control action at (t + k) given u(t)
- y(t + k|t) is the predicted output at (t + k) given y(t)



Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Model Predictive Control

- Major advantages of MPC in comparison to traditional reactive control approaches, e.g. PID, etc. are
  - **Proactive control action:** The controller is anticipating future disturbances, set-points etc.
  - **Non-linear control:** MPC can explicitly consider non-linear systems without linearization
  - **Constrained formulation:** Explicitly consider physical, safety or operational system constraints

Introduction to Marine Craft Model

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Introduction to Marine Craft Model

- **Fossen's marine craft model** provides a mathematical framework for expressing the nonlinear equations of motion of marine craft in a compact matrix–vector form
- **The matrix-vector representation** is particularly valuable for **designing Guidance, Navigation, and Control (GNC) systems**. It is widely applied to marine craft, including ships, floating offshore structures, submarines, autonomous underwater vehicles (AUVs), and **uncrewed surface vehicles (USVs)**.

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Introduction to Marine Craft Model

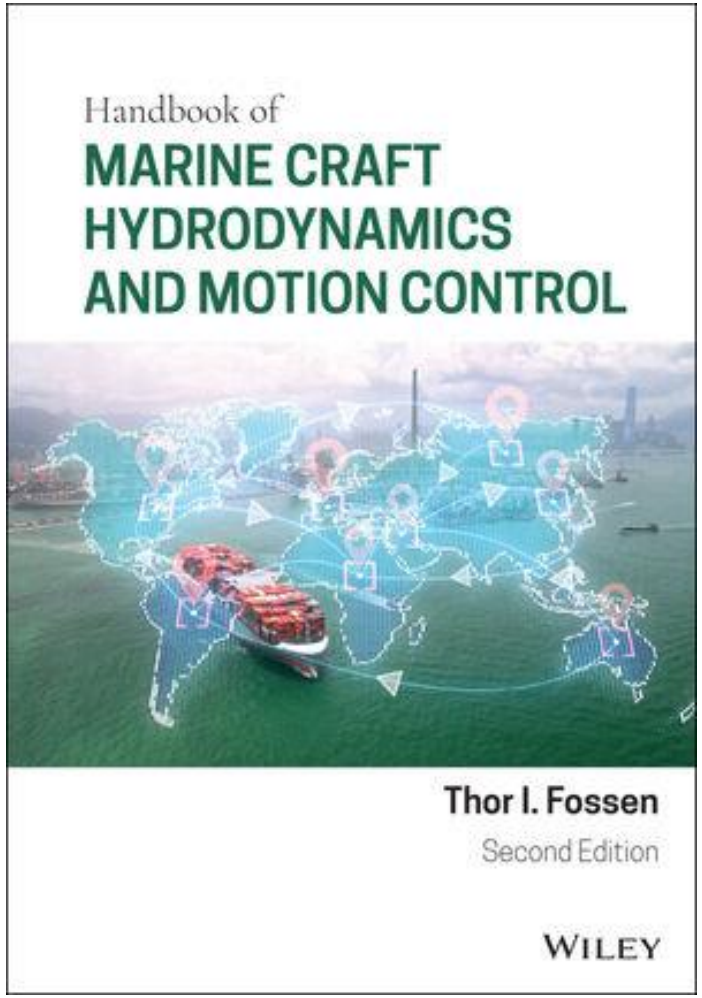
Handbook of Marine Craft Hydrodynamics and Motion Control (2nd ed.) Wiley

Fossen, T. I. (2021)

Textbook supplements: https://wiley.fossen.biz

MSS Toolbox: https://github.com/cybergalactic/MSS

Python Vehicle Simulator: https://github.com/cybergalactic/PythonVehicleSimulator



Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Marine Craft

In Encyclopedia Britannica, a ship and a boat are distinguished by their size through the following definition:

- **Ship:** any large floating vessel capable of crossing open waters, as opposed to a boat, which is generally a smaller craft

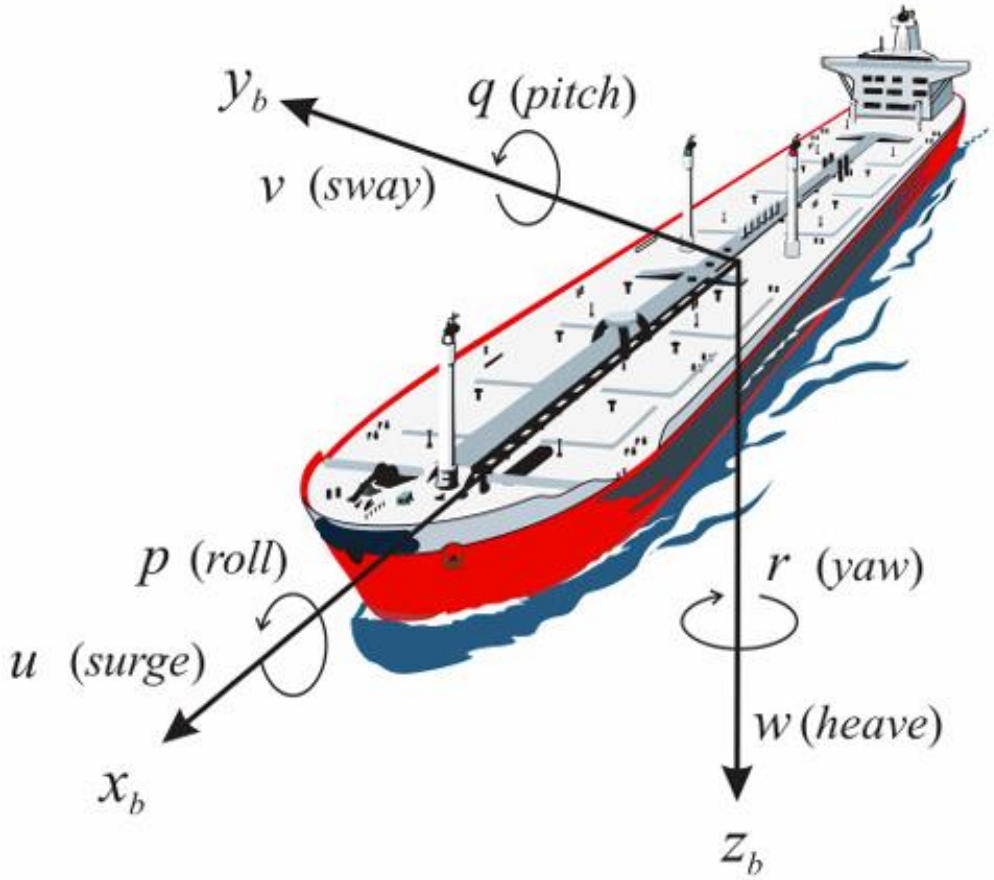
- **Submarine:** "any naval vessel that is capable of propelling itself beneath the water as well as on the water's surface

- **Underwater Vehicle:** "small vehicle that is capable of propelling itself beneath the water surface as well as on the water's surface
  - This includes unmanned underwater vehicles (UUV), remotely operated vehicles (ROV) and autonomous underwater vehicles (AUV)

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Degrees of Freedom (DOFs)

- For a marine craft, DOF is the set of **independent displacements and rotations** that completely specify the **displaced position and orientation** of the craft
- A craft that can move freely in the 3-D space has maximum **6 DOFs—three translational and three rotational** components
- Consequently, a fully actuated marine craft operating in 6 DOFs must be equipped with **actuators that can produce independent forces and moments in all directions**



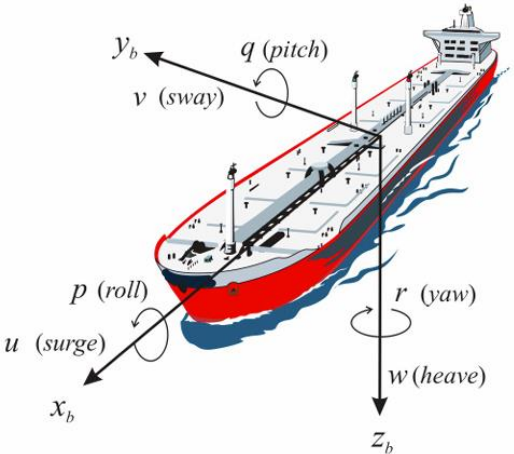
Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Degrees of Freedom (DOFs)

When designing **feedback control systems for marine craft,** reduced-order models are often used since most vehicles do not have actuation in all DOF

This is usually done by decoupling the motions of the vessel:

- **1-DOF** models can be used to design forward speed controllers **(surge),** heading autopilots **(yaw)** and roll damping systems **(roll)**



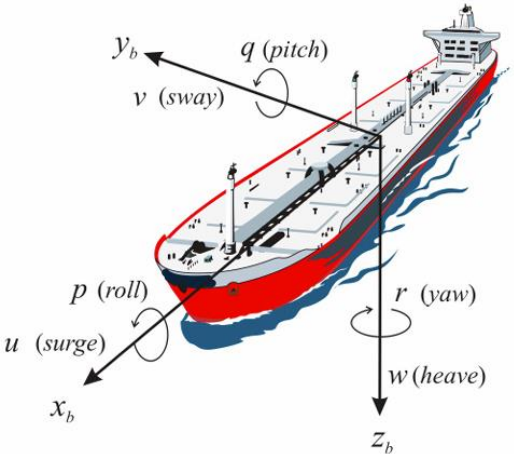
Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Degrees of Freedom (DOFs)

- **3-DOF** models are usually horizontal-plane models (**surge, sway and yaw**) for ships, semi-submersibles and underwater vehicles, trajectory-tracking control systems and path-following systems

  For slender bodies such as torpedo-shaped AUVs and submarines, it is also common to assume that the motions can be decoupled into longitudinal and lateral motions

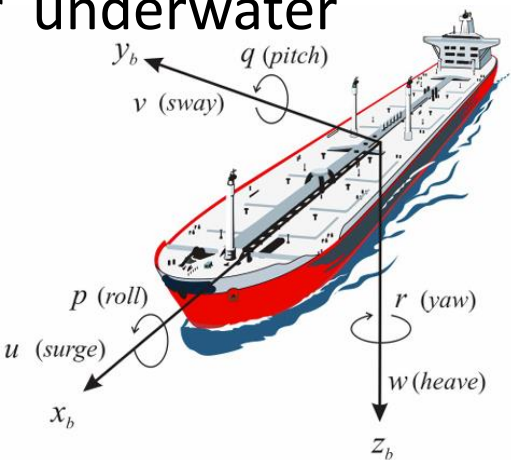
  - **Longitudinal models (surge, heave and pitch)** for forward speed, diving and pitch control
  - **Lateral model (sway, roll and yaw)** for turning and heading control



Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Degrees of Freedom (DOFs)

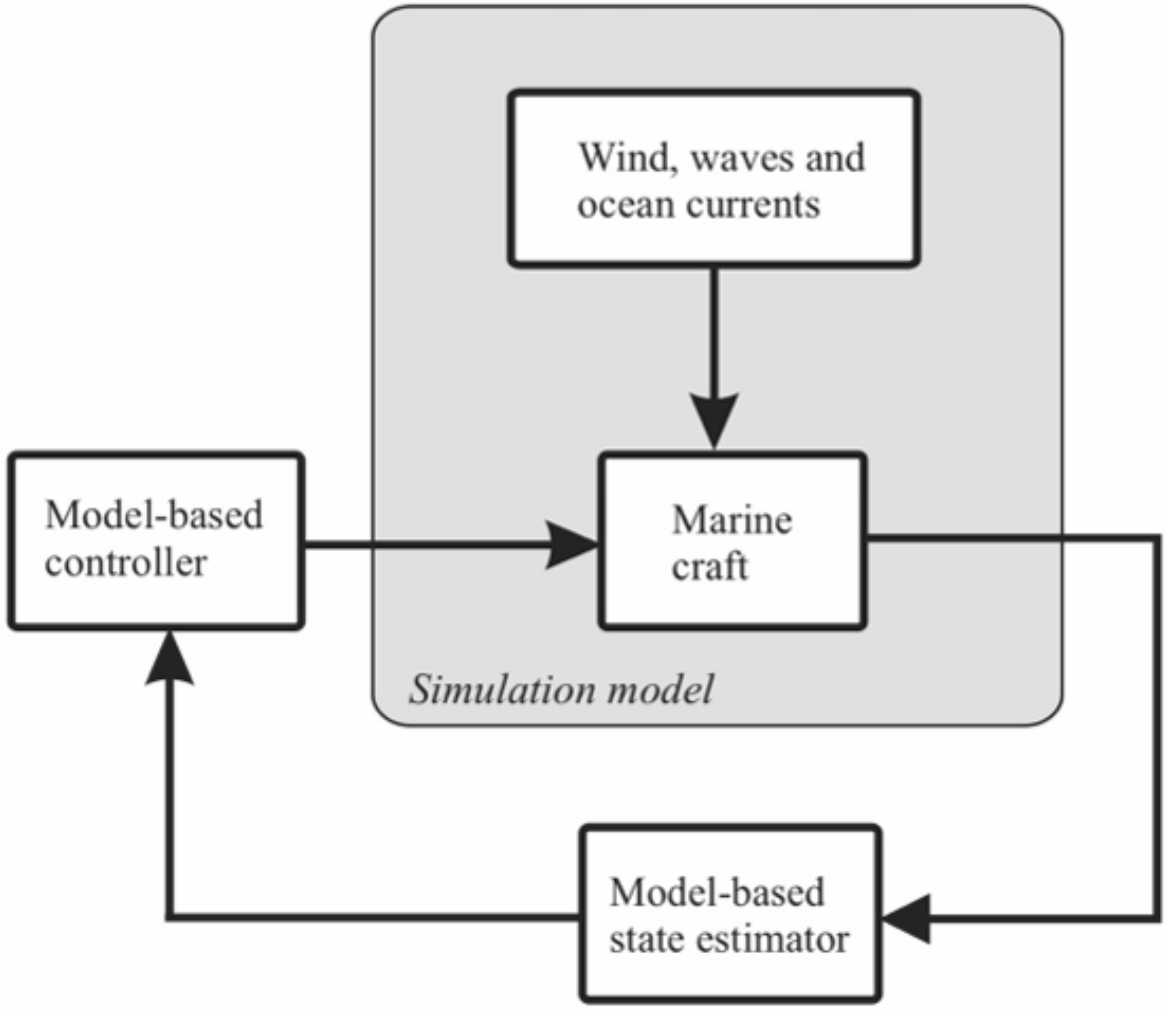
- **4-DOF models (surge, sway, roll and yaw)** are usually formed by adding the roll equation to the 3-DOF horizontal-plane model
  - These models are used in maneuvering situations where the purpose is to reduce roll by active control of fins, rudders or stabilizing liquid tanks
- **6-DOF models (surge, sway, heave, roll, pitch and yaw)** are fully coupled equations of motion used for simulation and prediction of coupled vessel motions
  - These models can also be used in advanced control systems for underwater vehicles, which are actuated in all DOFs



Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Classification of Models

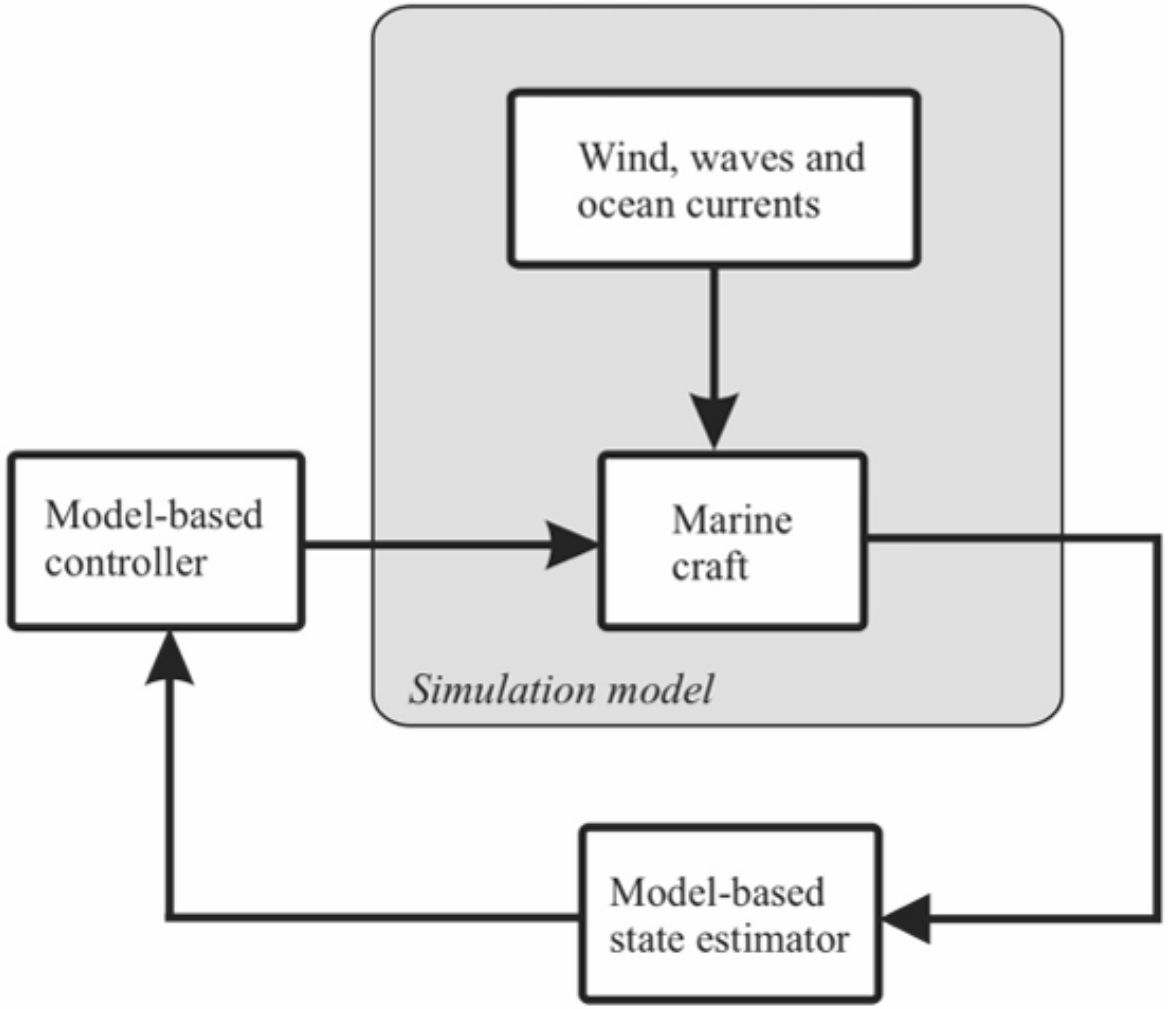
- **Simulation Model:** This model is the most accurate description of a system, for instance a 6-DOF high-fidelity model for simulation of coupled motions in the time domain



Classification of Models

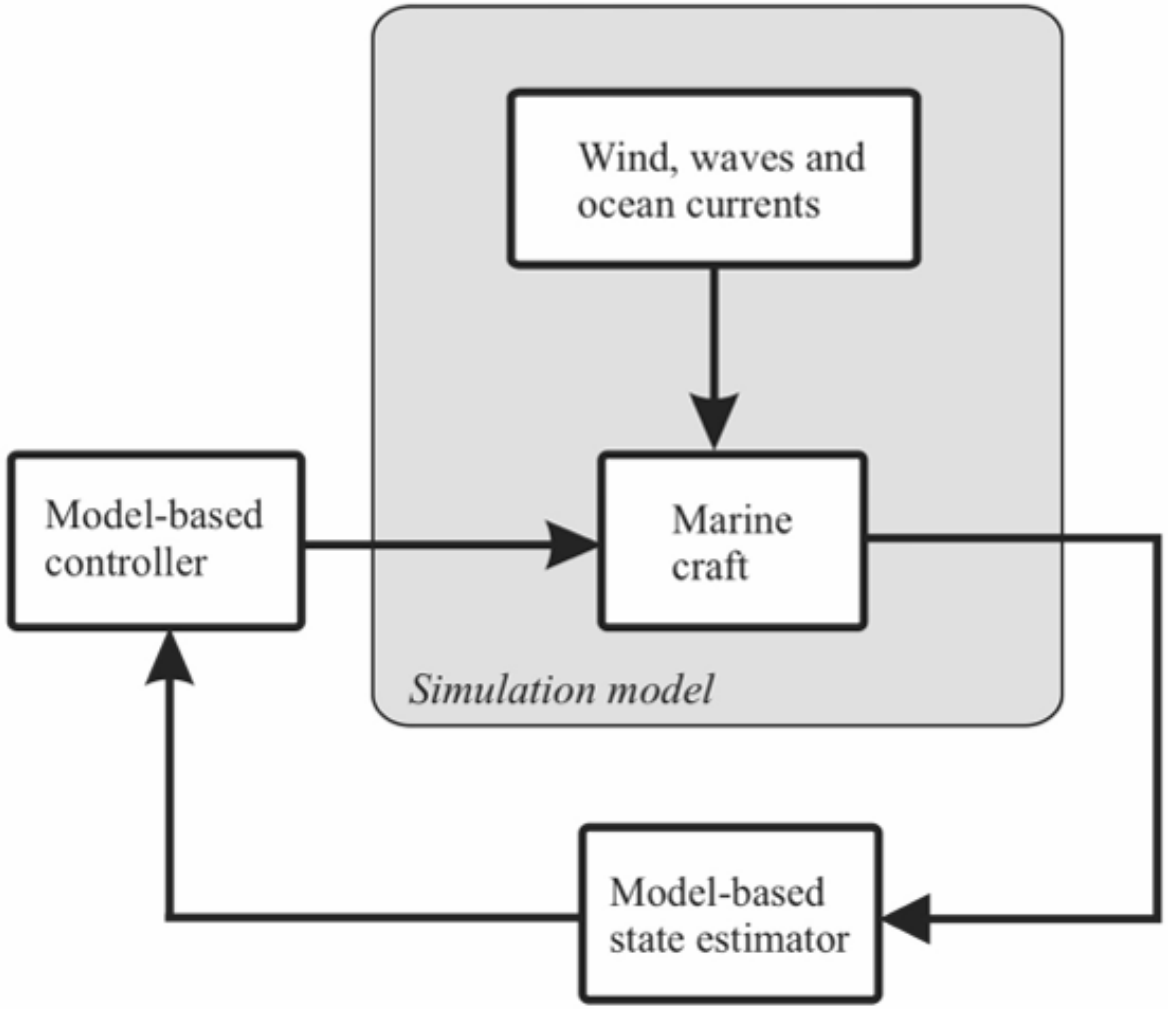
- **Control Design Model:** The controller model is a reduced-order or simplified simulation model that is used to design the motion control system

  In its simplest form, this model is used to compute a set of constant gains for a PID controller



Classification of Models

- **State Estimator Design Model:** Stochastic state estimators (Kalman filters) and deterministic state observers are both designed using **mathematical models** which are different from the models used in the simulator and **the controller** since the purpose is to capture the additional dynamics associated with the sensors and navigation system as well as disturbances



Guidance, Navigation and Control

Guidance *is the action or the system that continuously computes the reference (desired) position, velocity and attitude of a marine craft to be used by the motion control system. These data are usually provided to the human operator and the navigation system.*

How do I get there?
Actions required to reach the destination

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Guidance, Navigation and Control

Navigation *is the science of directing a craft by determining its position/attitude, course and distance traveled. In some cases velocity and acceleration are determined as well.*

Where am I?
Where do I want to go?
Which way should I go?
Planning a route

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Guidance, Navigation and Control

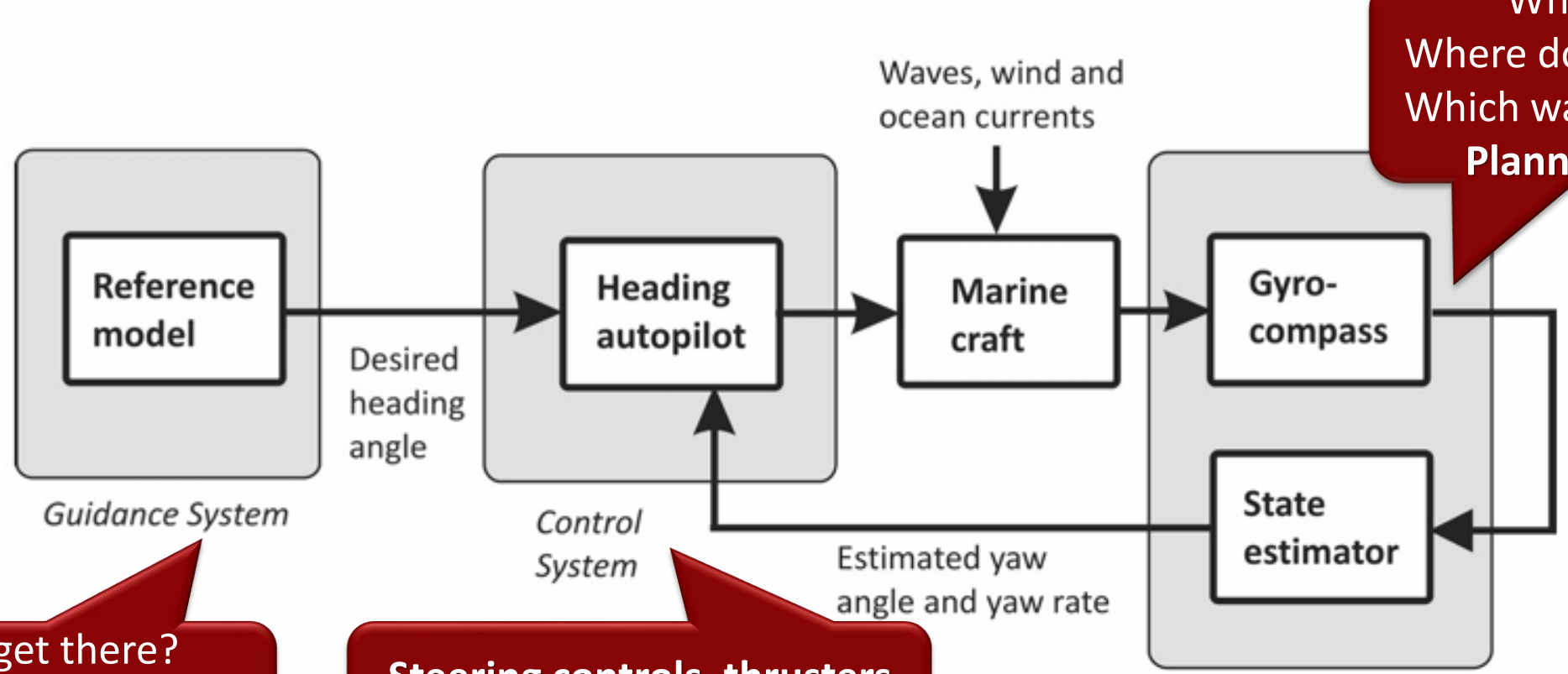
Control, *or more specifically motion control and control allocation, is the action of determining the necessary control forces and moments to be provided by the craft in order to satisfy a certain control objective.*

Steering controls, thrusters
Maintaining vehicle stability

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Open-Loop Guidance System

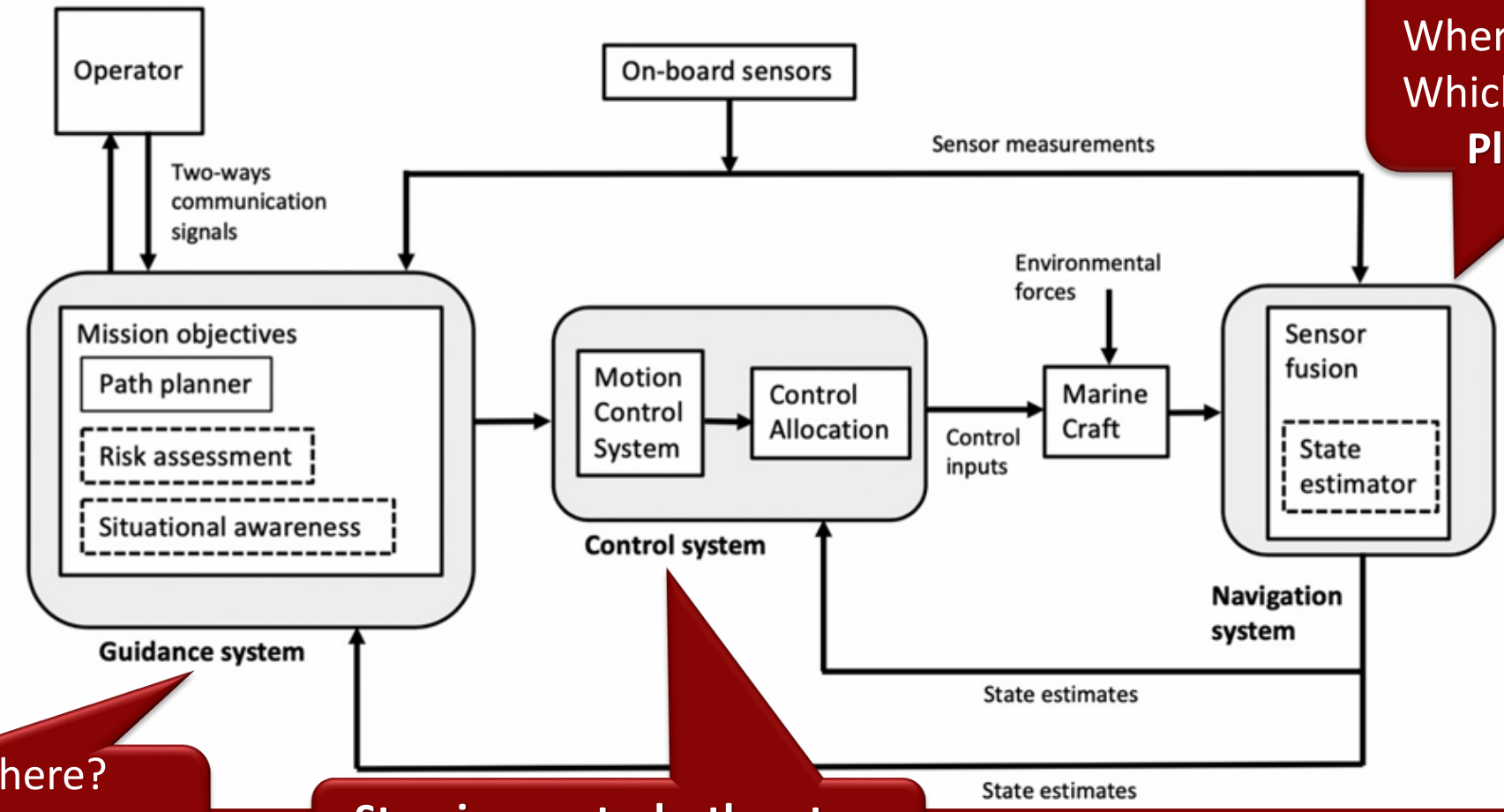
- GNC blocks with an open-loop guidance system represented by a reference model



Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr

Closed-Loop Guidance System

- GNC blocks where the guidance system makes use of the estimated alternatively measured positions and velocities



Where am I?
Where do I want to go?
Which way should I go?
Planning a route

How do I get there?
Actions required to reach the destination

Steering controls, thrusters
Maintaining vehicle stability

Thank you very much for your attention!

Prof. Xanthi S. Papageorgiou
xpapag@aegean.gr