Autonomous Systems
- Map-based localisation -

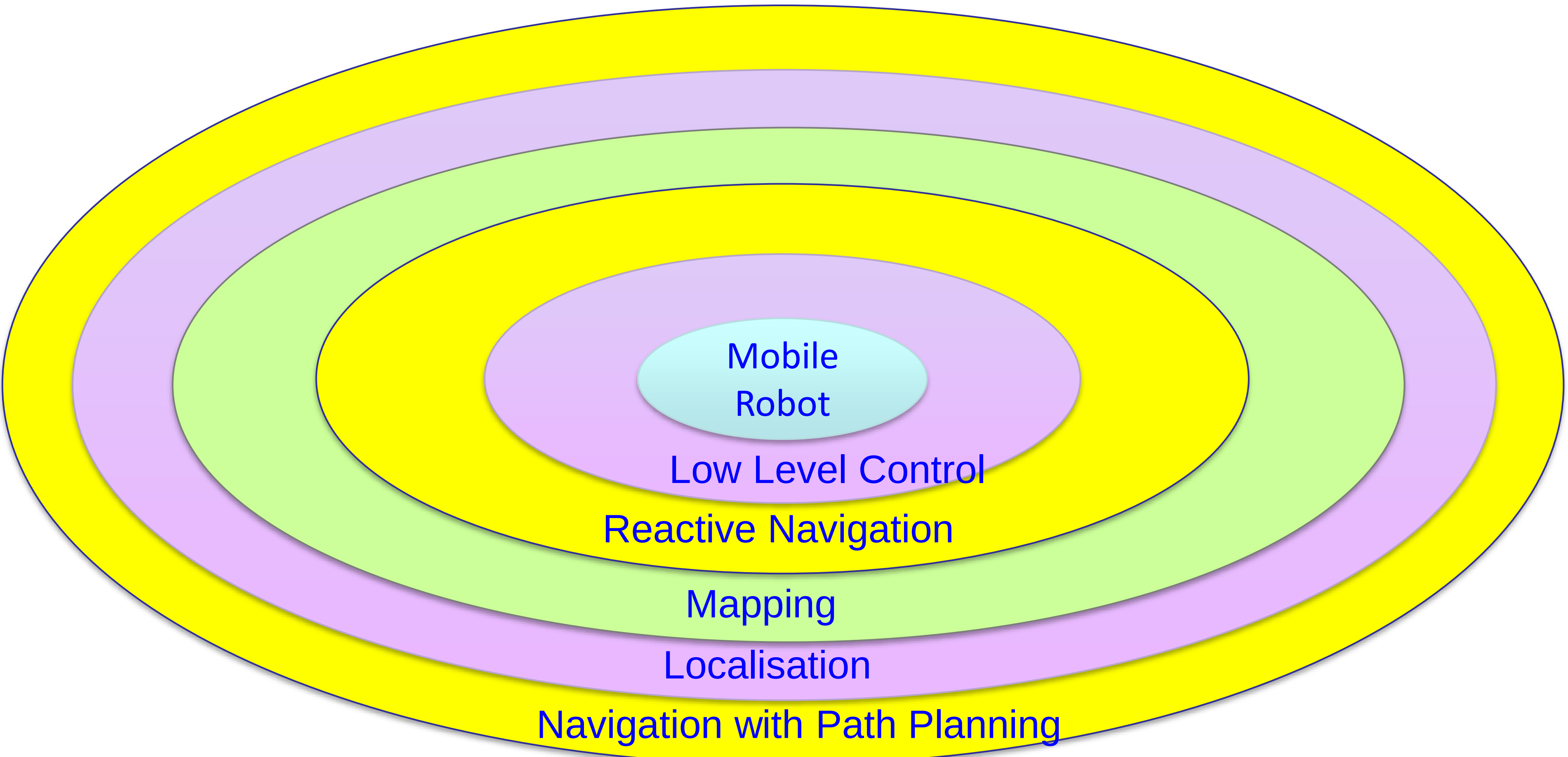
Dr Alexandru Stancu
Dr Mario Martinez

This week lectures

- Map-based localisation
  - Bayes filter
  - Kalman filter
- Case study: Map-based localisation using Kalman filter
- Worked examples

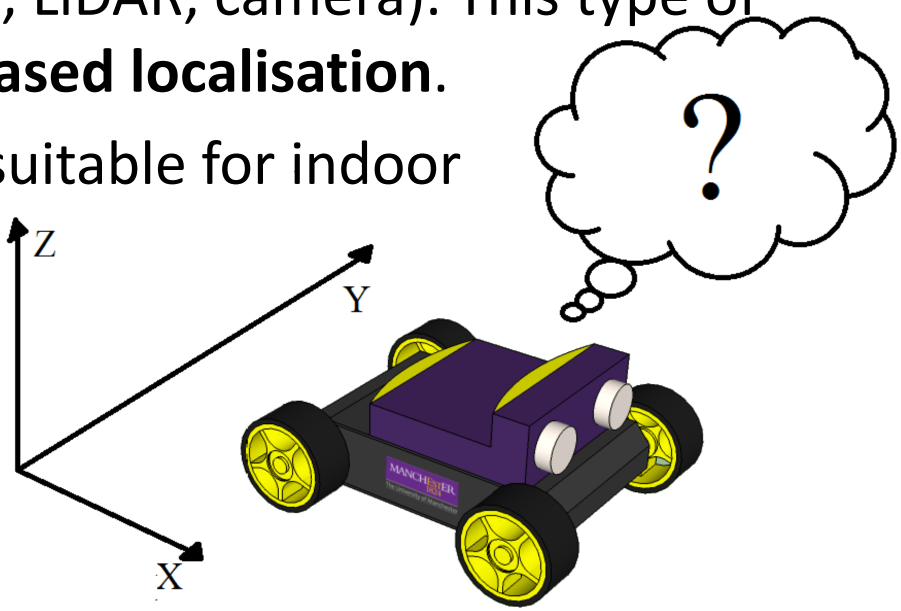
Where we are and where we are going

How much information and support must be provided by human to ensure that the robot is able to achieve its goals.

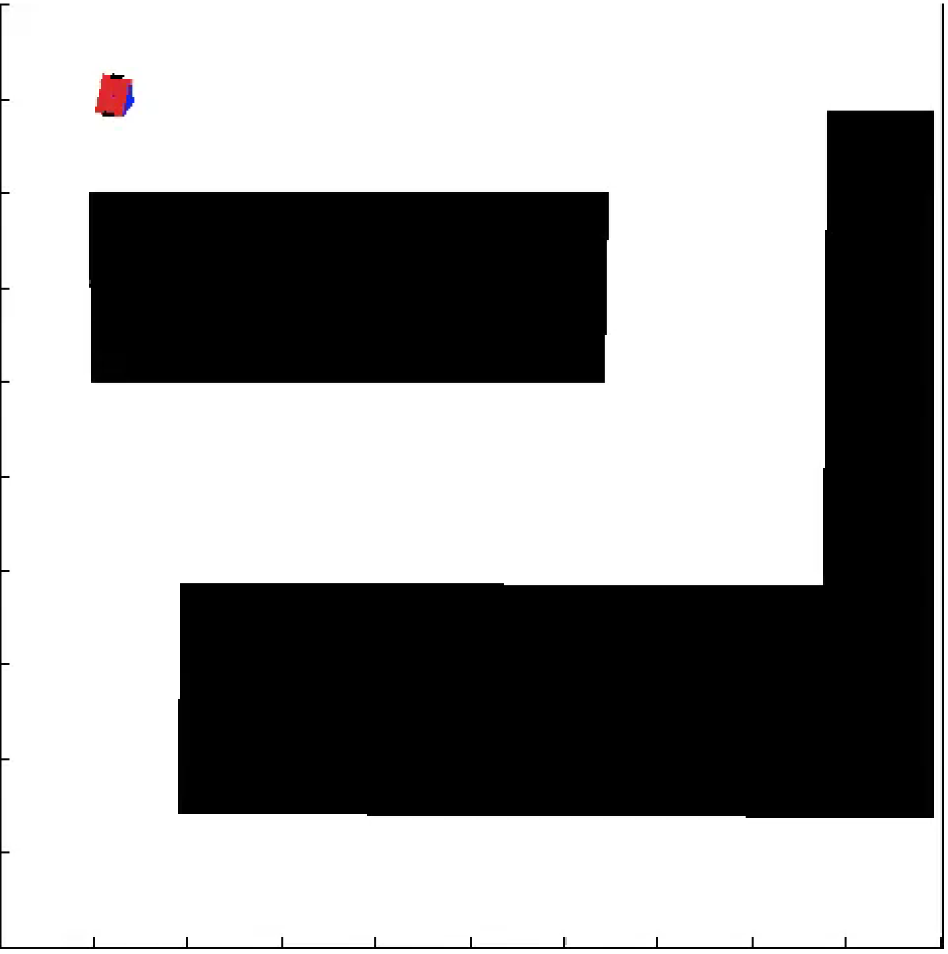


Localisation

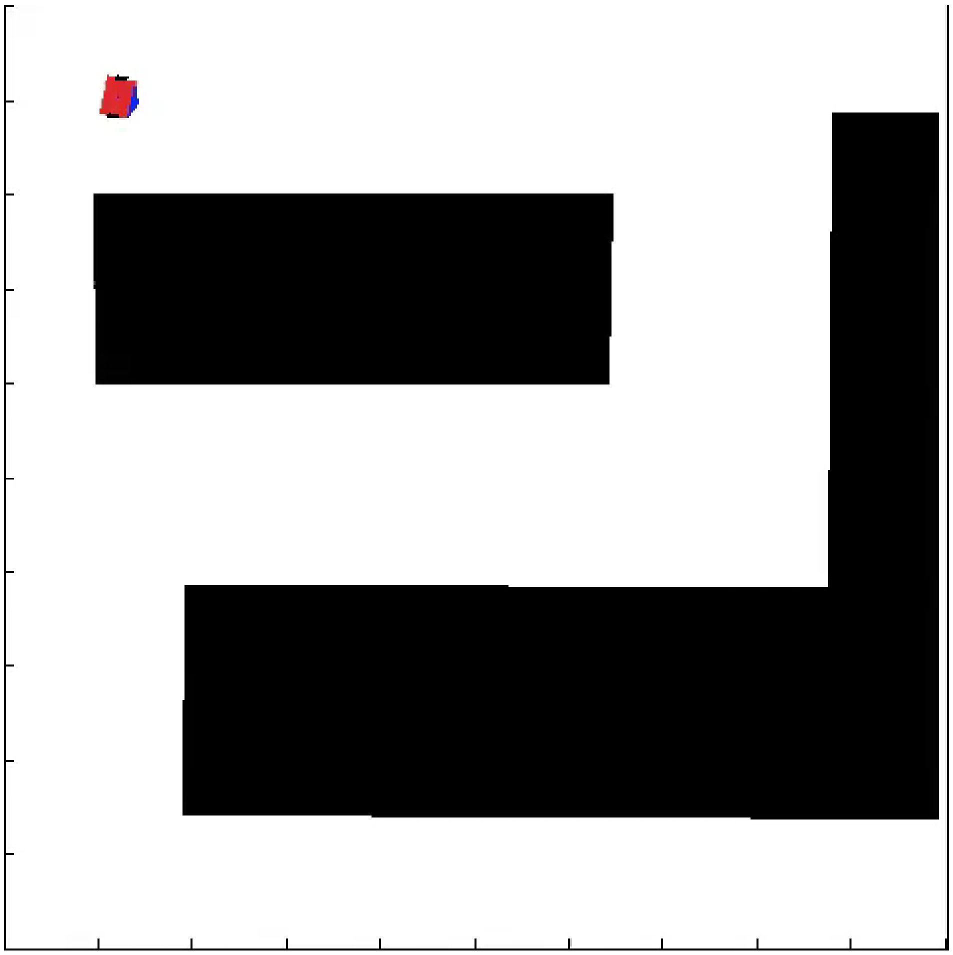
- The robot needs to know its location in the environment in order to make proper decisions.
- Localisation can be achieved using:
  - Proprioceptive sensors (encoders, IMU). This type of localisation is named **dead reckoning localisation**.
  - Exteroceptive sensors (sonar, LiDAR, camera). This type of localisation is named **map-based localisation**.
  - External sensors (GPS). Not suitable for indoor applications.



Localisation



Dead Reckoning Localisation



Extended Kalman Filter Localisation
The map is needed!!!

Bayes filter – theoretical foundation

Bayes filter is a probabilistic approach to recursively estimates some unknown PDFs as new information, such as measurements, becomes available.

1. State transition, or motion model probability:

$$p(x_k|x_{k-1}, u_k)$$

2. Measurement, or observation model probability:

$$p(z_k|x_k)$$

Bayes filter – theoretical foundation

The belief

$bel(x_k)$ is the probability distribution of the system state x_k

conditioned on all available data; control input: $U_k = \{u_1, \ldots, u_k\}$

and measurement $Z_k = \{z_1, \ldots, z_k\}$ such that:

$$bel(x_k) = p(x_k | U_k, Z_k)$$

Definition of the belief

Bayes filter – theoretical foundation

In the following, we will derive a recursive formula for the Bayes filter.

At time instant *k,* we have *k* known observations and *k* control signals have been executed.

$$bel(x_k) = p(x_k | u_1, u_2, \dots, u_k, z_1, z_2, \dots, z_k)$$

Recall the Bayes' Theorem for continuous random variables:

$$p_{\mathrm{X}|Y}(x|y) = \frac{p_{\mathrm{Y}|X}(y|x) p_X(x)}{p_Y(y)}$$

Bayes filter – theoretical foundation

Let's apply Bayes' Theorem to x_k and z_k

Therefore, we will swap the two variables and Bayes' Theorem tells us how to do that:

$$bel(x_k) = p(x_k|u_1, u_2, \dots, u_k, z_1, z_2, \dots, z_{k-1}, z_k) =$$

$$= \frac{p(z_k|x_k, u_1, u_2, \dots, u_k, z_1, z_2, \dots, z_{k-1}) \cdot p(x_k|u_1, u_2, \dots, u_k, z_1, z_2, \dots, z_{k-1})}{p(z_k|u_1, u_2, \dots, u_k, z_1, z_2, \dots, z_{k-1})}$$

Bayes filter – theoretical foundation

Considering the following notation:

$$\eta = \frac{1}{p(z_k | u_1, u_2, \dots, u_k, z_1, z_2, \dots, z_{k-1})}$$

where η is a normalisation constant that is independent of x_k.

Therefore $bel(x_k)$ can be expressed as following:

$$bel(x_k) = \eta \cdot p(z_k | x_k, u_1, u_2, \dots, u_k, z_1, z_2, \dots, z_{k-1}) \cdot p(x_k | u_1, u_2, \dots, u_k, z_1, z_2, \dots, z_{k-1})$$

Bayes filter – theoretical foundation

We consider the **Markov assumption** for the first term of $bel(x_k)$

⬇

If we know the probability of obtaining the measurement z_k

given that we know the state, then we can ignore the previous measurements and the previous commands executed. Therefore, given the state of the world we can ignore what happened in the past. Consequently, we can get rid of the observations and measurements. Therefore, considering the following assumption:

$$p(z_k|x_k, U_k, Z_{k-1}) = p(z_k|x_k)$$

Bayes filter – theoretical foundation

Hence:

$$bel(x_k) = \eta \cdot p(z_k|x_k) \cdot p(x_k|u_1, u_2, \dots, u_k, z_1, z_2, \dots, z_{k-1})$$

$$bel(x_k) = \eta \cdot p(z_k|x_k) \cdot p(x_k|U_k, Z_{k-1})$$

$p(z_k|x_k)$ is the observation model. We make the following notation:

$$\overline{bel}(x_k) = p(x_k|U_k, Z_{k-1})$$

$\overline{bel}(x_k)$ is the estimation of the current state of the system given the past observations up to the time step *k-1*, so the last observation is missing, but all commands executed. **So, having an estimate up to *k-1* and executing a motion command.**

Bayes filter – theoretical foundation

We apply **Marginal Distribution** for $\overline{bel}\,(x_k)$

Marginal distribution is applied for continuous random variables and it is equivalent with Law of total probability for discrete random variables.

$$\overline{bel}(x_k) = p(x_k | U_k, Z_{k-1})$$

$$= \int\limits_{x_{k-1}} p(x_k | x_{k-1}, U_k, Z_{k-1}) p(x_{k-1} | U_k, Z_{k-1}) dx_{k-1}$$

x_{k-1} is a new variable which represents the state of the system at previous time step.

Bayes filter – theoretical foundation

Therefore, a new variable is introduced and we integrate over this new variable, so the expression will stay the same. We apply again Markov assumption and the expression of $\overline{bel}(x_k)$ become:

$$\overline{bel}(x_k) = \int_{x_{k-1}} p(x_k|x_{k-1}, u_k)p(x_{k-1}|U_{k-1}, Z_{k-1})dx_{k-1}$$

Therefore, u_k is executed 'to go' from x_{k-1} to x_k. This tells us how the system should evolve form *k-1* to *k,* i.e., $\overline{bel}(x_k)$ represents the motion model. In other words, given the state x_{k-1} and the command *GO 2 METERS FORWARD* then we can have a probability distribution about the estimation of the position of the robot at current time.

Bayes filter – theoretical foundation

Therefore, we can express $bel(x_k)$ as:

$$bel(x_k) = \eta \cdot p(z_k|x_k) \int_{x_{k-1}} p(x_k|x_{k-1}, u_k) bel(x_{k-1}) dx_{k-1}$$

where $p(z_k|x_k)$ is the observation model and $bel(x_{k-1})$ is the previous believe, i.e., believe at *k-1*.

We have a recursive update scheme which allows us to estimate the state of the system based on the previous state x_{k-1} , the current motion command u_k and the current observation z_k .

Bayes filter – theoretical foundation

Therefore, if we have a probability distribution for x_{k-1} (the previous state of the system), and if we execute a motion command and we get a new measurement from the exteroceptive sensor (observation), then we can compute the state of the system at the current time instant k.

$$\Downarrow$$

Recursive Bayes filter

In conclusion, the Bayes filter consists of two main steps: **Prediction** and **Correction**.

Bayes filter – theoretical foundation

Recursive Bayes filter as a Prediction/Correction algorithm:

Prediction step:

$$\overline{bel}(x_k) = \int_{x_{k-1}} p(x_k | x_{k-1}, u_k) bel(x_{k-1}) dx_{k-1}$$

Correction step:

$$bel(x_k) = \eta \cdot p(z_k | x_k) \cdot \overline{bel}(x_k)$$

Because I don't know the exact state of the system at time instant *k-1*, I have to integrate over all possible states.

η makes the integral of all possible states to sum up to one.

Bayes filter – theoretical foundation

It is not possible to solve analytically the integral of system states at *k-1*.

The Bayes filter is a general method for state estimation. It is a theoretical method. However, in robotics, we need an algorithm which can be implemented in a computer.

Kalman filter

Particle filter

Bayes filter – theoretical foundation

If we can represent the Probability Density Function (PDF) as Gaussian (parametric distribution described by mean and variance)

Kalman filter

If the PDF has a non-parametric distribution (has no closed form), there are unlimited numbers of parameters to describe the PDF; hence we need some kind of sampling technique to represent the full distribution.

Particle filter

Next topic

Kalman filter