Robot Localization

One of the foundational questions a robot must constantly answer is, "Where am I?

Robot localization is the process by which a mobile robot determines its position within a given environment relative to a known map. It is the robot's ability to establish its own location and subsequently update that location as it moves through space. Just as humans use landmarks, paths, and other cues to determine their location, robots use sensors and algorithms to compute their position.

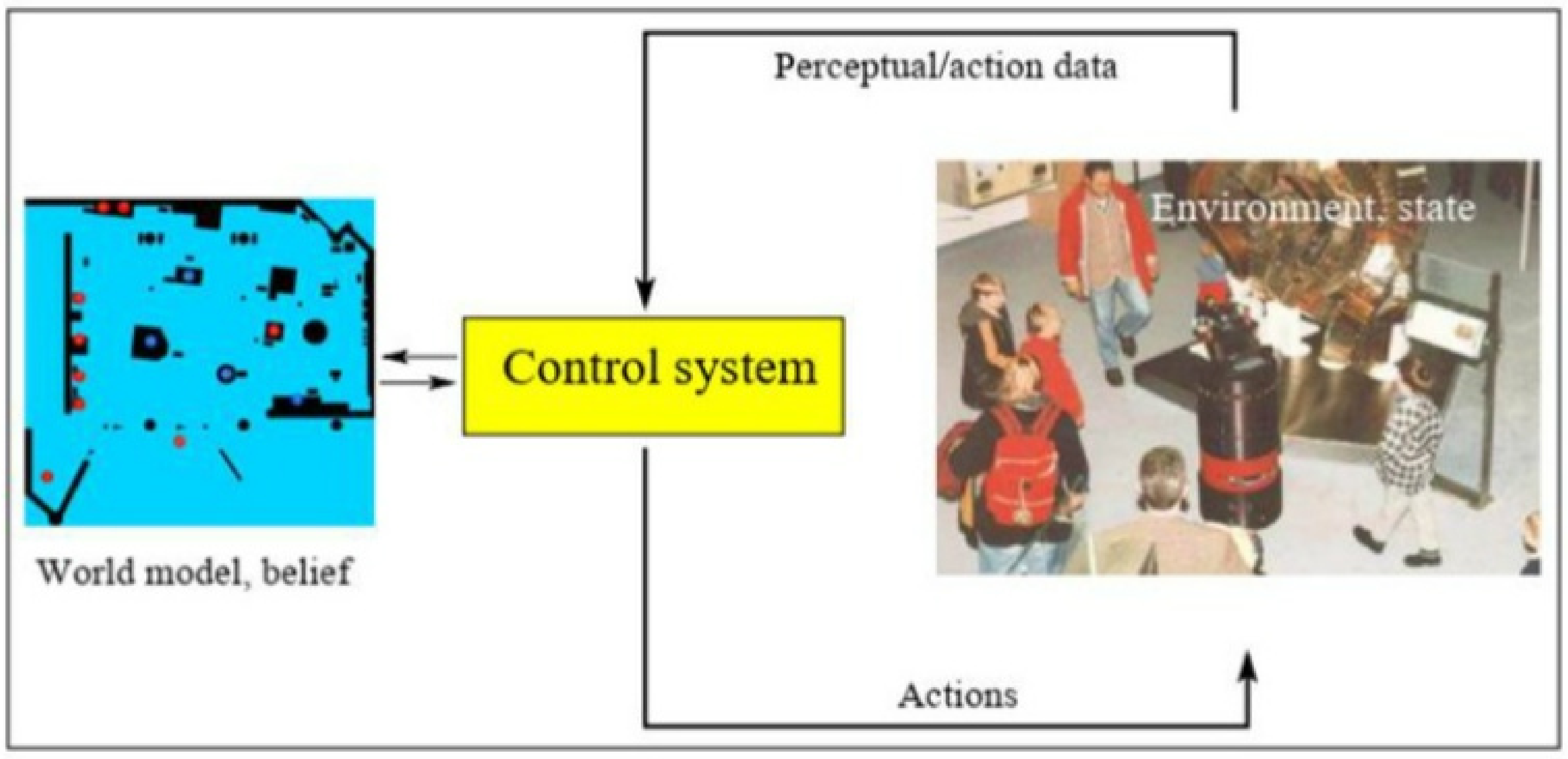
Why is Localization Important?

Safe Navigation: To safely and efficiently move from one point to another, a robot must be aware of its current position. A misjudgment can lead to collisions, getting lost, or failure in executing tasks.

Decision Making: Accurate localization allows robots to make informed decisions. For instance, an autonomous car needs to know its precise location before deciding to turn or overtake.

Multi-Robot Systems: In scenarios with multiple robots, like in warehouse automation or drone swarms, localization is crucial to coordinate actions and avoid inter-robot collisions.

Robot Environment Interaction



The environment, or world of a robot is dynamical system and has many parts that can affect what the robot does. The robot can acquire information about its environment using its sensors. However, sensors are noisy, and there are usually many things that cannot be sensed directly. As a consequence, the robot maintains an internal belief to the state of its environment. where everything is and what's happening. The robot can also influence its environment through its actuators.

Challenges in Localization

- Dynamic Environments: Environments can change, with obstacles moving or appearing unexpectedly. This introduces discrepancies between the robot's internal map and the actual world.
- Sensor Noise: No sensor is perfect. The inherent noise in measurements can lead to inaccurate position estimations.
- Complex Environments: Features like overhanging structures, reflective surfaces, or areas with similar appearances can confuse localization systems.

State

Environments are characterized by state. state (denoted x) is collection of all aspects of the robot and its environment that can impact the future. State may change over time, such as the location of people; or it may remain static throughout the robot's operation, such as the location of walls in (most) buildings. The state also includes variables regarding the robot itself, such as its pose, velocity, whether or not its sensors are functioning correctly, The location and features of surrounding objects in the environment and so on.

Markov Assumption

The idea is that the future is independent of the past given the present. For a state to be Markov, the probability of transitioning to any subsequent state depends solely on the current state and not on any sequence of states that preceded it.

$$P(X_{t+1} = a | X_1 = x_1, X_2 = x_2, \ldots, X_t = x_t) = P(X_{t+1} = a | X_t = x_t)$$

if state xt is complete, then knowing xt provides all the information needed to predict the future. Any additional data from previous states, measurements, or controls won't add any predictive power. if a state is complete as described, it adheres to the Markov property.

Environment Interaction

Sensor measurements. Perception is the process by which the robot uses its sensors to obtain information about the state of its environment. The result of such perceptual interaction is called a measurement zt . Examples of measurement data include camera images, range scans, and so on.

Control actions change the state of the world. Examples of control actions include robot motion and the manipulation of objects. Control data will be denoted ut

Probabilistic Generative Laws

The evolution of state and measurements is governed by probabilistic laws. In general, the state at time xt is generated stochastically. The probabilistic law characterizing the evolution of state:

$$p(x_t \mid x_{0:t-1}, z_{1:t-1}, u_{1:t})$$

$$p(x_t \mid x_{0:t-1}, z_{1:t-1}, u_{1:t}) = p(x_t \mid x_{t-1}, u_t)$$

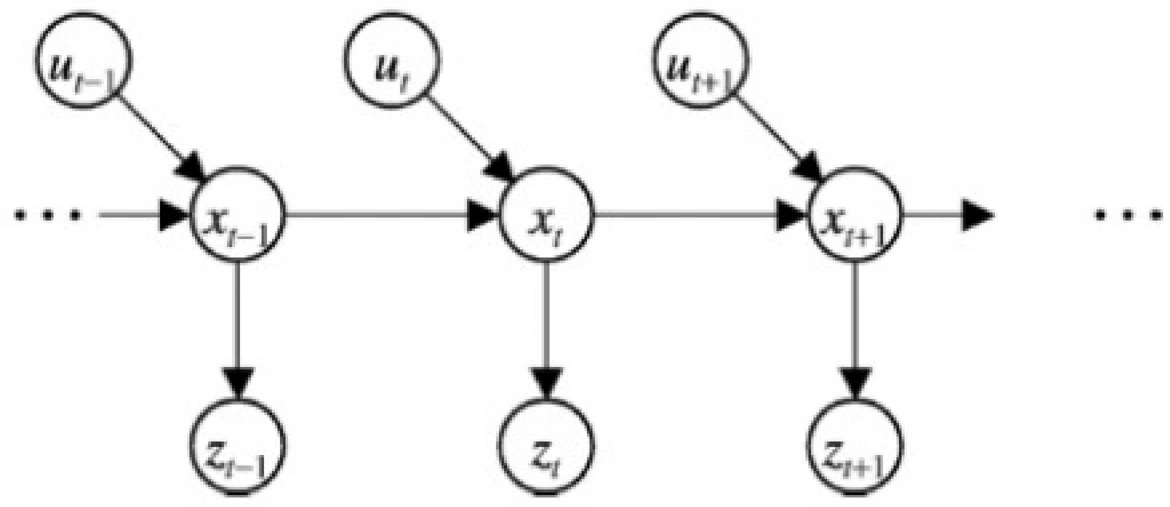
If state x is complete then x_{t-1} is a sufficient statistic of all previous controls and measurements up to this point, that is, $u_{1:t-1}$ and $z_{1:t-1}$.

The probability $p(x_t \mid x_{t-1}, u_t)$ is the **state transition probability**. It specifies how environmental state evolves over time as a function of robot actions ut is the action taken at time t, that influence the transition from x_{t-1} to the state x_t .

Similarly, one might want to model the process by which measurements are being generated. The probability p(zt|xt) is called the **measurement probability.**

$$p(z_t \mid x_{0:t}, z_{1:t-1}, u_{1:t}) \quad = \quad p(z_t \mid x_t)$$

the state x_t is sufficient to predict the (potentially noisy) measurement zt.



Belief Distributions

A belief reflects the robot's internal knowledge about the state of the environment.

Belief distributions are posterior probabilities over state variables conditioned on the available data. We will denote belief over a state variable xt by bel(xt)

$$bel(x_t) \quad = \quad p(x_t \mid z_{1:t}, u_{1:t})$$

This posterior is the probability distribution over the state xt at time t, conditioned on all past measurements z1:t and all past controls u1:t. The most general algorithm for calculating beliefs is given by the **Bayes filter algorithm**.

Bayes Filters

Given:

- Stream of observations z and action data u:

  $$d_t = \{u_1, z_1 \ldots, u_t, z_t\}$$

- Sensor model P(z|x).
- Action model P(x|u,x').
- Prior probability of the system state P(x).

Wanted:

- Estimate of the state X of a dynamical system.
- The posterior of the state is also called Belief:

$$Bel(x_t) = P(x_t \mid u_1, z_1 \ldots, u_t, z_t)$$