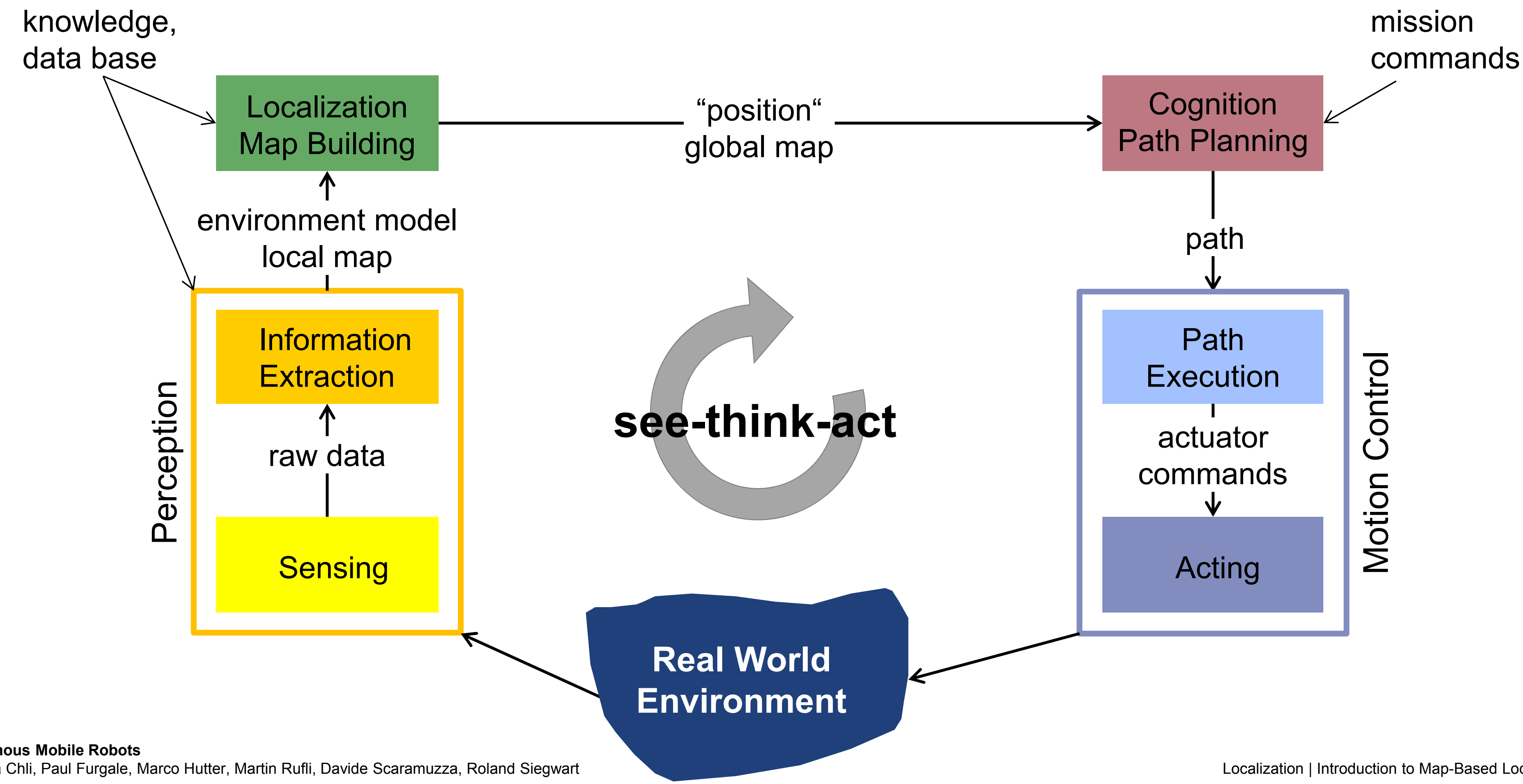
Localization | **Introduction to Map-Based Localization**

Autonomous Mobile Robots

Roland Siegwart

Margarita Chli, Paul Furgale, Marco Hutter, Martin Rufli, Davide Scaramuzza

Introduction | probabilistic map-based localization



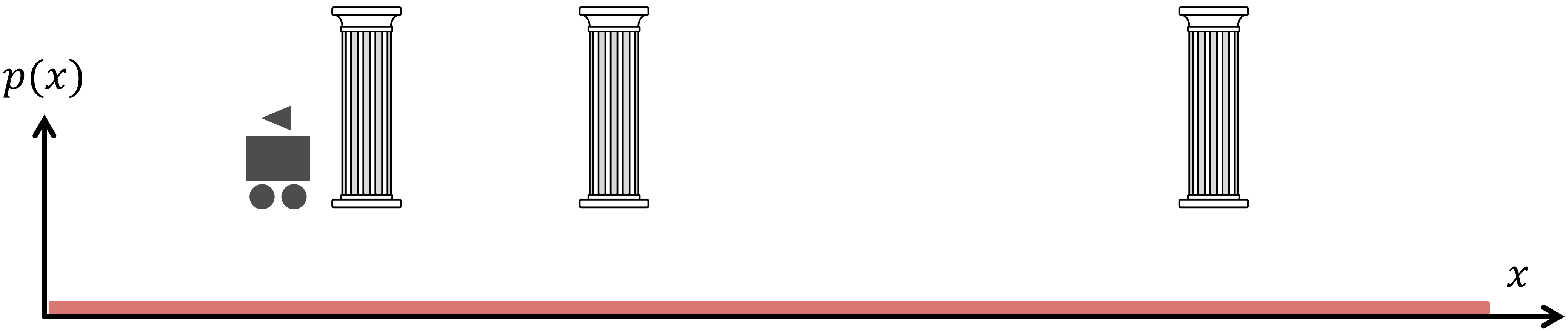
Localization | definition, challenges and approach

- Map-based localization
  - The robot estimates its position using perceived information and a map
  - The map
    - might be known (localization)
    - Might be built in parallel (simultaneous localization and mapping – SLAM)
- Challenges
  - Measurements and the map are inherently error prone
  - Thus the robot has to deal with uncertain information

  ⟶ Probabilistic map-base localization
- Approach
  - The robot estimates the belief state about its position through an ACT and SEE cycle

Concept | SEE and ACT to improve belief state

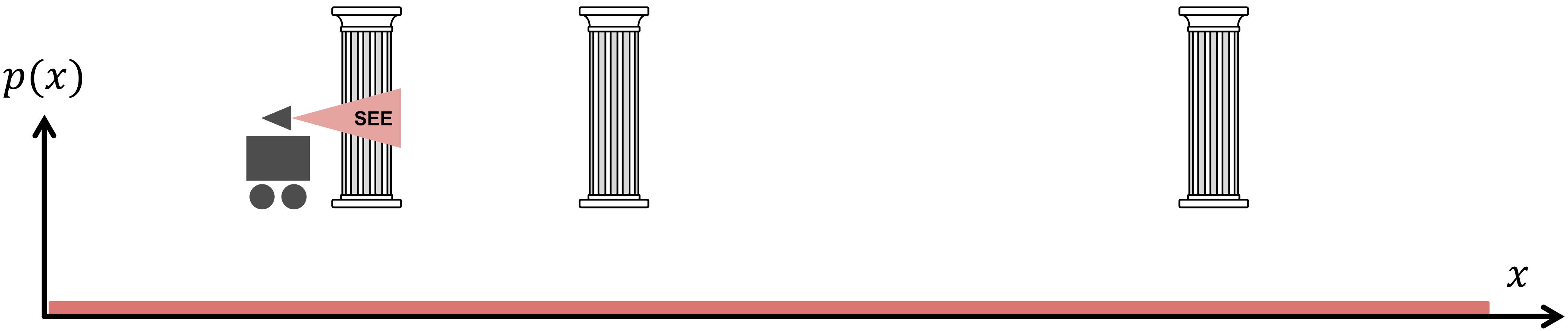
- Robot is placed somewhere in the environment → location unknown
- SEE: The robot queries its sensors → finds itself next to a pillar
- ACT: Robot moves one meter forward
  - motion estimated by wheel encoders
  - accumulation of uncertainty
- SEE: The robot queries its sensors again → finds itself next to a pillar
- Belief updates (information fusion)



Concept | SEE and ACT to improve belief state

- Robot is placed somewhere in the environment → location unknown
- SEE: The robot queries its sensors → finds itself next to a pillar

- ACT: Robot moves one meter forward
  - motion estimated by wheel encoders
  - accumulation of uncertainty
- SEE: The robot queries its sensors again → finds itself next to a pillar
- Belief updates (information fusion)



Concept | SEE and ACT to improve belief state

- Robot is placed somewhere in the environment → location unknown
- SEE: The robot queries its sensors
  → finds itself next to a pillar

- ACT: Robot moves one meter forward
  - motion estimated by wheel encoders
  - accumulation of uncertainty
- SEE: The robot queries its sensors again
  → finds itself next to a pillar
- Belief updates (information fusion)

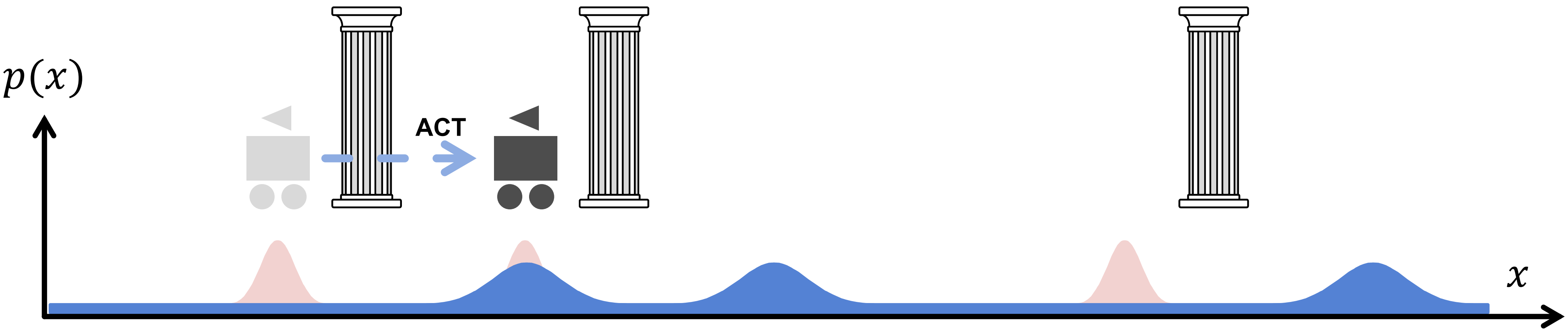
Concept | SEE and ACT to improve belief state

- Robot is placed somewhere in the environment → location unknown
- SEE: The robot queries its sensors → finds itself next to a pillar

- ACT: Robot moves one meter forward
  - motion estimated by wheel encoders
  - accumulation of uncertainty
- SEE: The robot queries its sensors again → finds itself next to a pillar
- Belief updates (information fusion)

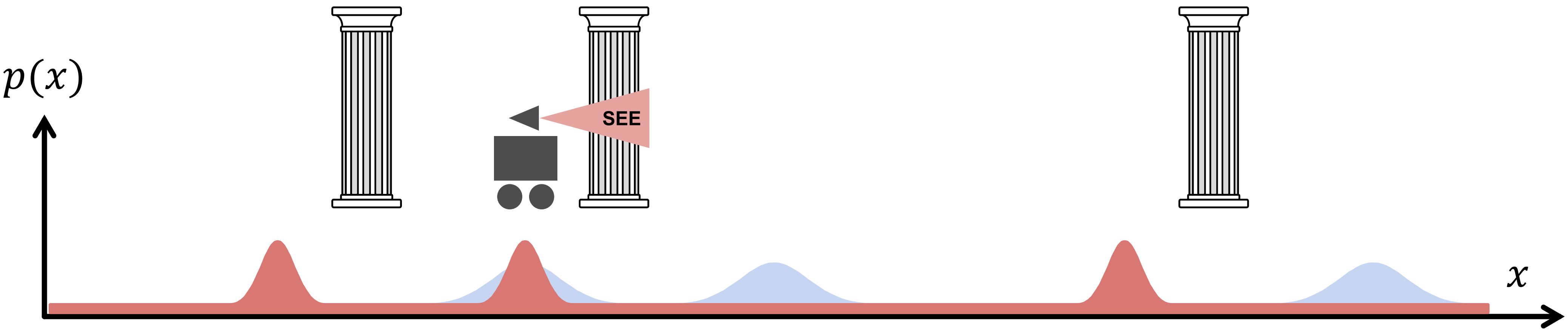
Concept | SEE and ACT to improve belief state

- Robot is placed somewhere in the environment → location unknown
- SEE: The robot queries its sensors → finds itself next to a pillar
- ACT: Robot moves one meter forward
  - motion estimated by wheel encoders
  - accumulation of uncertainty
- SEE: The robot queries its sensors again → finds itself next to a pillar
- Belief updates (information fusion)



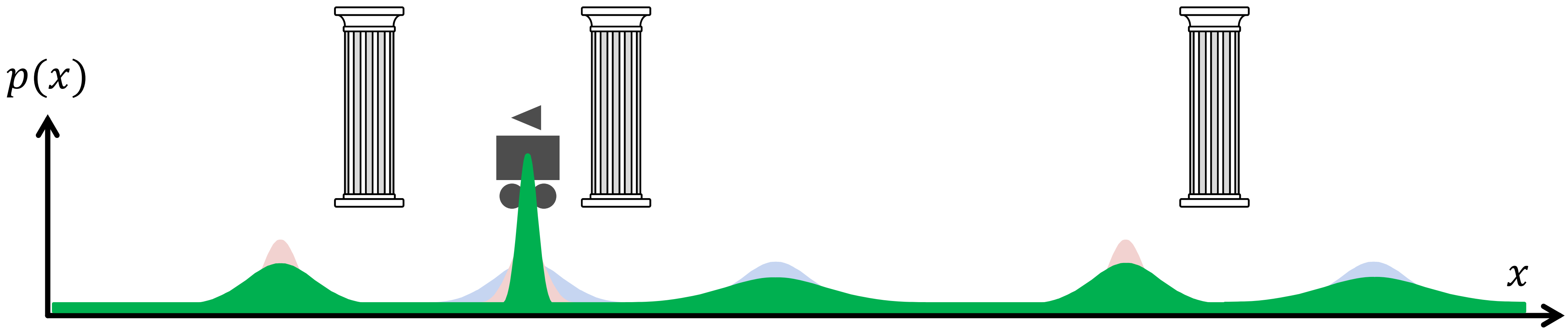
Concept | SEE and ACT to improve belief state

- Robot is placed somewhere in the environment → location unknown
- SEE: The robot queries its sensors → finds itself next to a pillar
- ACT: Robot moves one meter forward
  - motion estimated by wheel encoders
  - accumulation of uncertainty
- SEE: The robot queries its sensors again → finds itself next to a pillar
- Belief updates (information fusion)



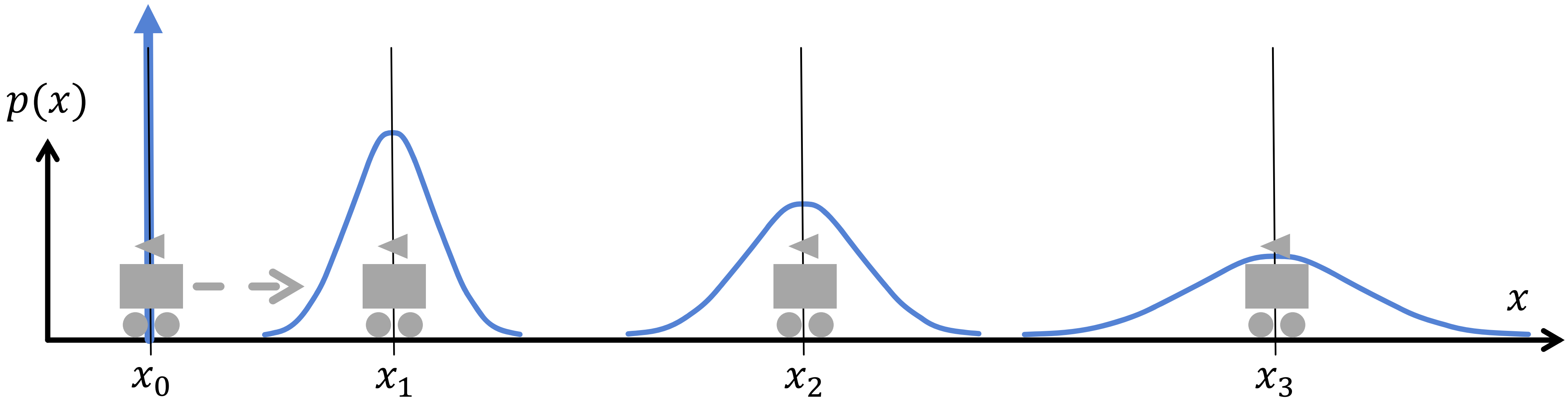
Concept | SEE and ACT to improve belief state

- Robot is placed somewhere in the environment → location unknown
- SEE: The robot queries its sensors → finds itself next to a pillar
- ACT: Robot moves one meter forward
  - motion estimated by wheel encoders
  - accumulation of uncertainty
- SEE: The robot queries its sensors again → finds itself next to a pillar
- Belief update (information fusion)



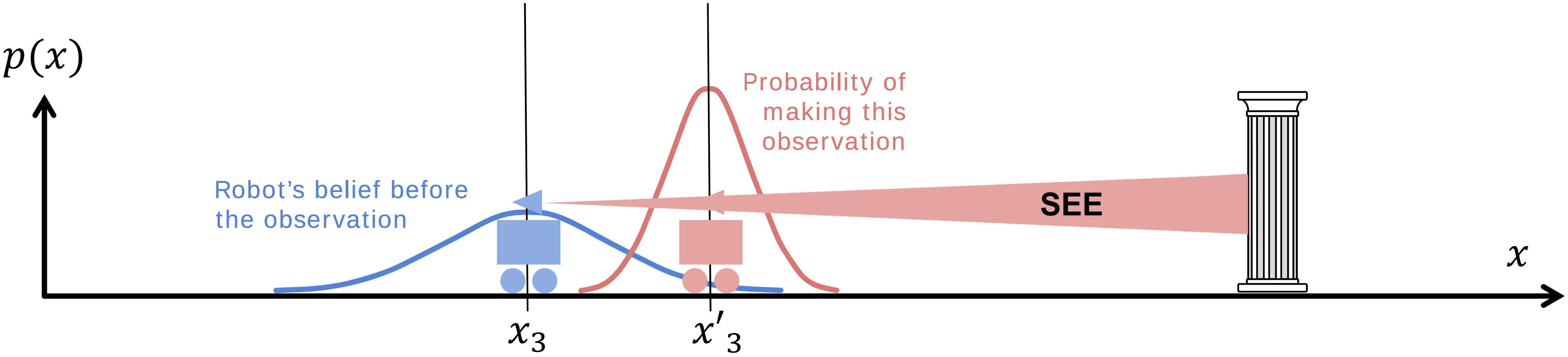
ACT | using motion model and its uncertainties

- The robot moves and estimates its position through its proprioceptive sensors
  - Wheel Encoder (Odometry)
- During this step, the robot's state uncertainty grows



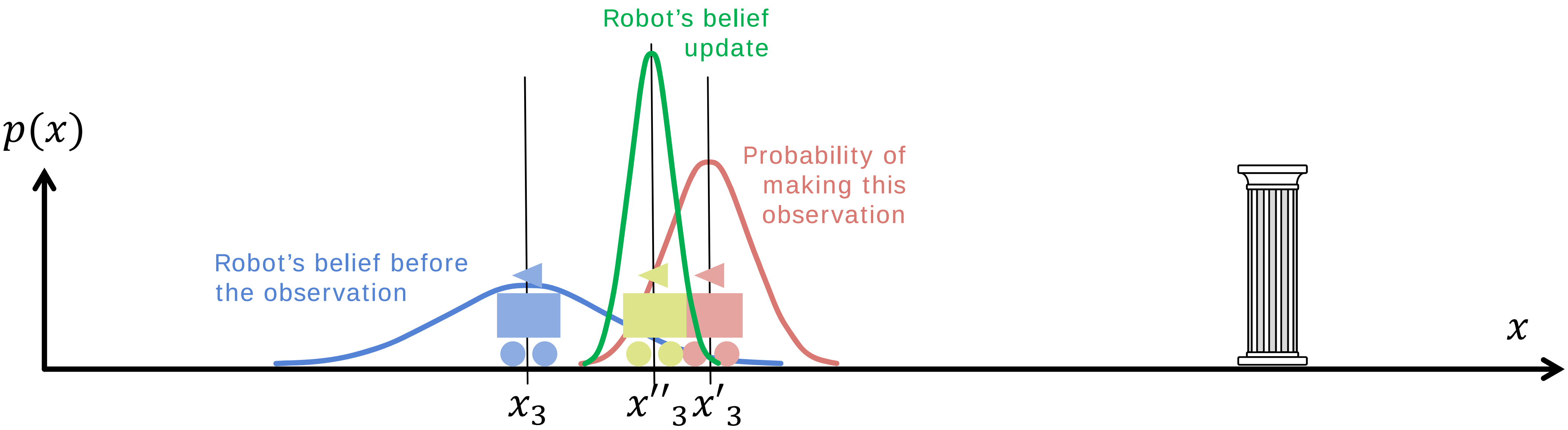
SEE | estimation of position based on perception and map

- The robot makes an observation using its exteroceptive sensors
- This results in a second estimation of the current position

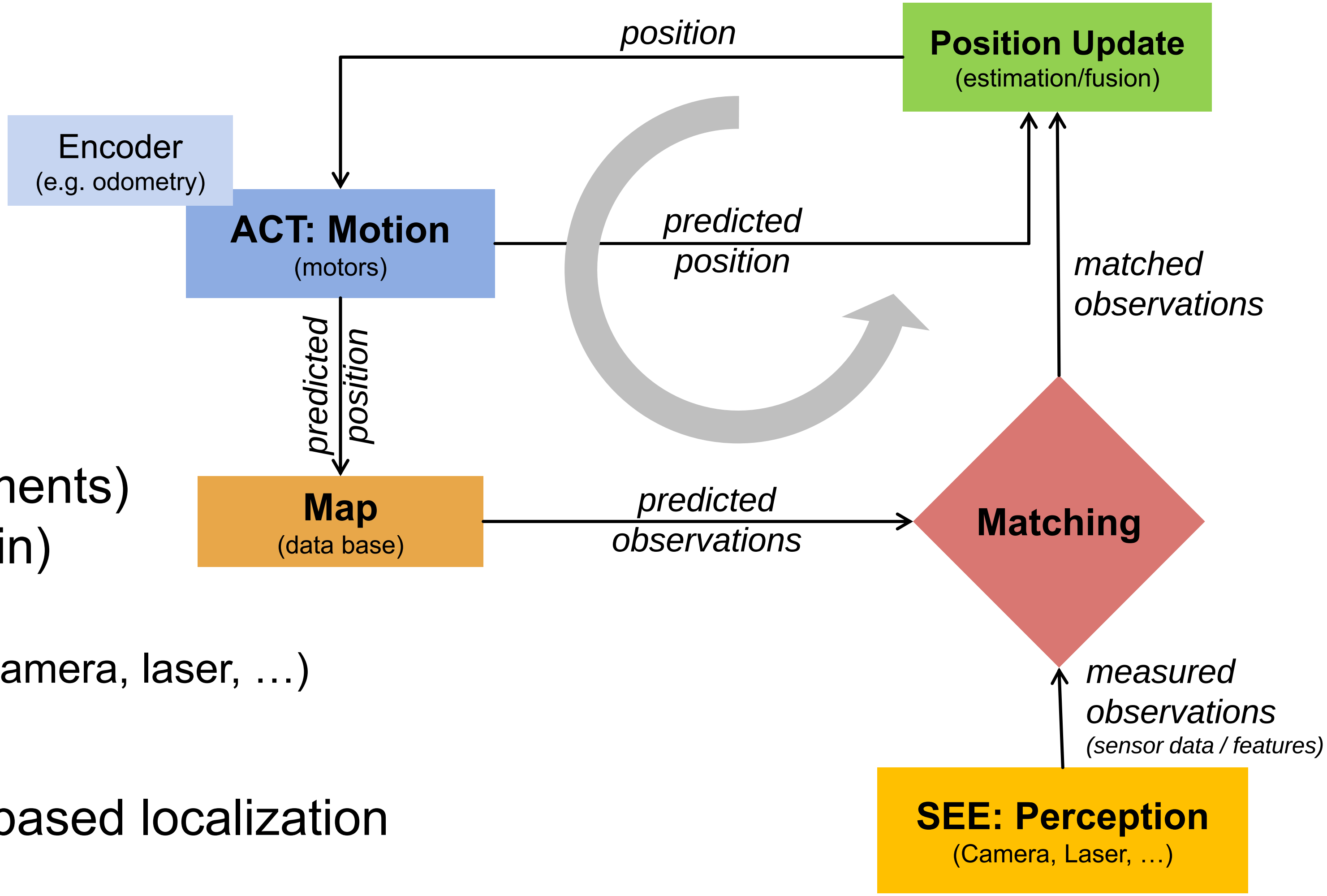


Belief update | fusion of prior belief with observation

- The robot corrects its position by combining its belief before the observation with the probability of making exactly that observation
- During this step, the robot's state uncertainty shrinks



Map-based localization | the estimation cycle (ACT-SEE)



- Information (measurements) is error prone (uncertain)
  - Odometry
  - Exteroceptive sensors (camera, laser, …)
  - Map

→ Probabilistic map-based localization

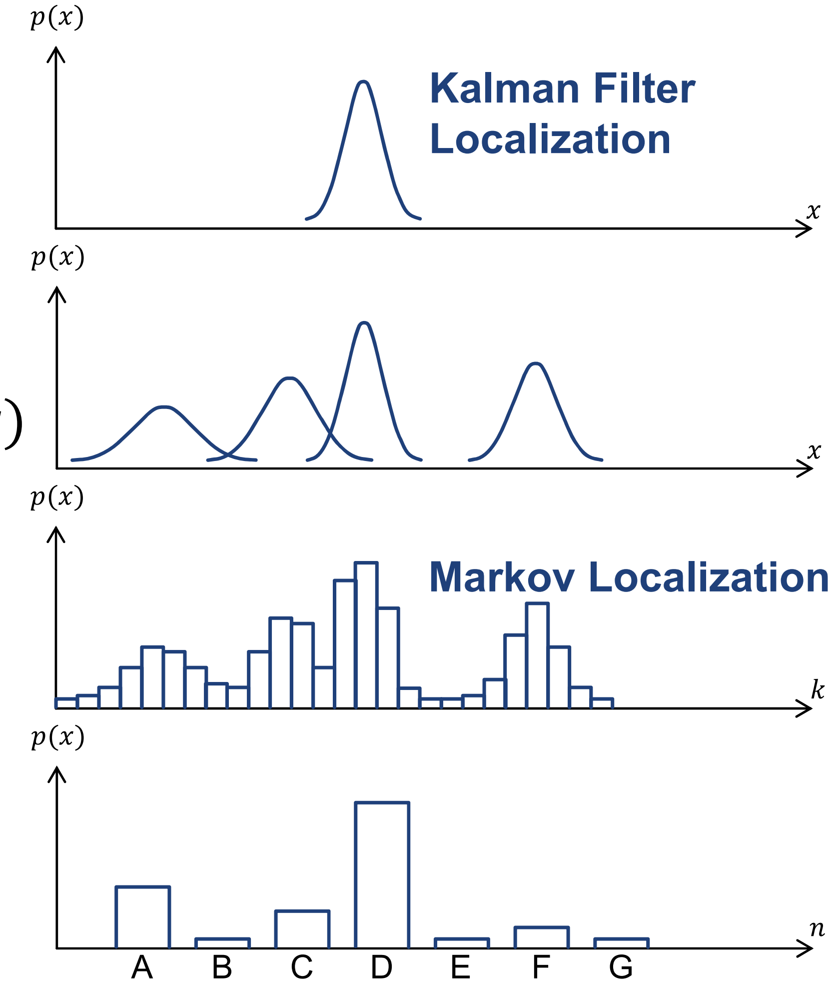
Probabilistic localization | belief representation

a) Continuous map with
single hypothesis probability distribution $p(x)$

b) Continuous map with
multiple hypotheses probability distribution $p(x)$

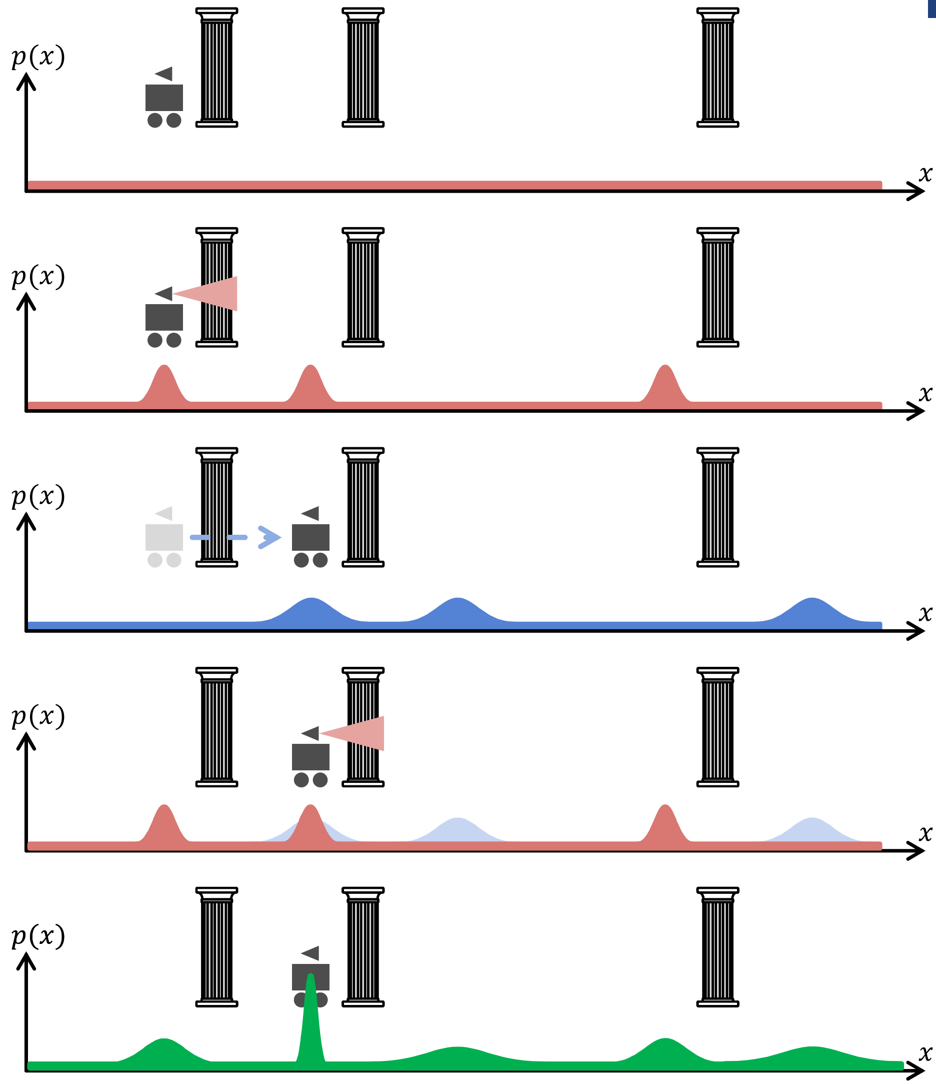
c) Discretized metric map (grid k) with
probability distribution $p(k)$

d) Discretized topological map (nodes n) with
probability distribution $p(n)$



Take home message | ACT - SEE Cycle for Localization

- SEE: The robot queries its sensors → finds itself next to a pillar
- ACT: Robot moves one meter forward
  - motion estimated by wheel encoders
  - accumulation of uncertainty
- SEE: The robot queries its sensors again → finds itself next to a pillar
- Belief update (information fusion)



Localization | **Refresher on Probability Theory**

Autonomous Mobile Robots

Roland Siegwart

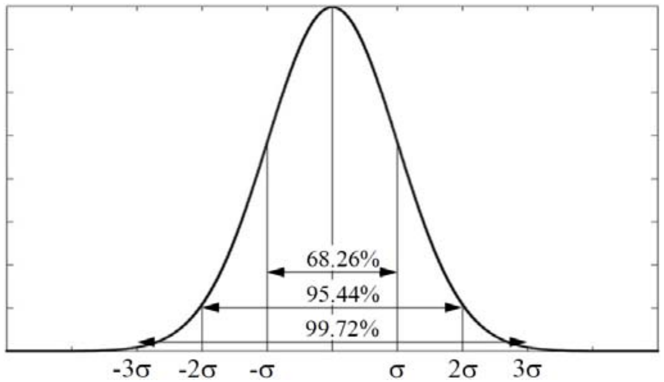
Margarita Chli, Paul Furgale, Marco Hutter, Martin Rufli, Davide Scaramuzza

Probability theory | how to deal with uncertainty

- Mobile robot localization has to deal with error prone information
- Mathematically, error prone information (uncertainties) is best represented by random variables and probability theory

- $p(x) = p(X = x)$: probability that the random variable X has value x (x is true).
  - X: random variable
  - x: a specific value that X might assume.
  - The **Probability Density Functions** (PDF) describes the relative likelihood for a random variable to take on a given value
  - PDF example: The Gaussian distribution:

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$



Basic concepts of probability theory | joint distribution

- $p(x, y)$: **joint distribution** representing the probability that the random variable X takes on the value x and that Y takes on the value y
  $\rightarrow$ x and y is true.

- If X and Y are independent we can write:

$$p(x, y) = p(x)p(y)$$

Basic concepts of probability theory | conditional probability

- $p(x|y)$: **conditional probability** that describes the probability that the random variable X takes on the value x *conditioned* on the knowledge that Y for sure takes y.

$$p(x|y) = \frac{p(x,y)}{p(y)}$$

and if X and Y are independent (uncorrelated) we can write:

$$p(x|y) = \frac{p(x)p(y)}{p(y)} = p(x)$$

Basic concepts of probability theory | theorem of total probability

- The **theorem of total probability** (*convolution*) originates from the axioms of probability theory and is written as:

$$p(x) = \sum_{y} p(x|y)p(y)$$ for discrete probabilities

$$p(x) = \int_{y} p(x|y)p(y)dy$$ for continuous probabilities

- This theorem is used by both *Markov* and *Kalman-filter* localization algorithms during the prediction update.

Basic concepts of probability theory | the Bayes rule

- The **Bayes rule** relates the conditional probability $p(x|y)$ to its inverse $p(y|x)$.
- Under the condition that $p(y) > 0$, the Bayes rule is written as:

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)}$$

$$p(x|y) = \eta p(y|x)p(x)$$

$\eta = p(y)^{-1}$ normalization factor ($\int p = 1$)

- This theorem is used by both *Markov* and *Kalman-filter* localization algorithms during the measurement update.

Usage | application of probability theory to robot localization

- Probability theory is widely and very successfully used for mobile robot localization

- In the following lecture segments, its application to localization will be illustration
  - Markov localization
    - Discretized pose representation
  - Kalman filter
    - Continuous pose representation and Gaussian error model

- Further reading:
  - "Probabilistic Robotics," Thrun, Fox, Burgard, MIT Press, 2005.
  - "Introduction to Autonomous Mobile Robots", Siegwart, Nourbakhsh, Scaramuzza, MIT Press 2011

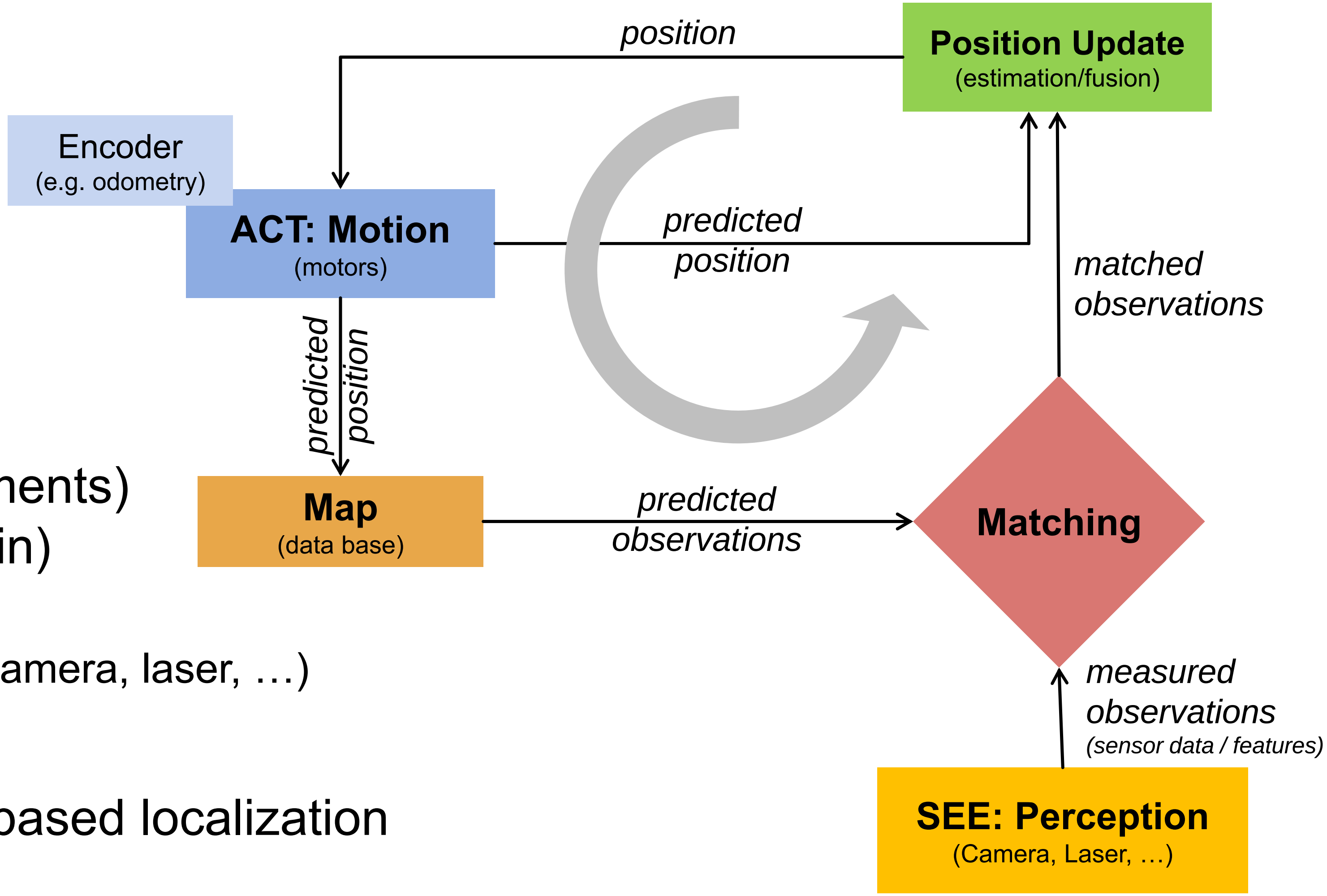
Localization | **the Markov Approach**

Autonomous Mobile Robots

Roland Siegwart

Margarita Chli, Paul Furgale, Marco Hutter, Martin Rufli, Davide Scaramuzza

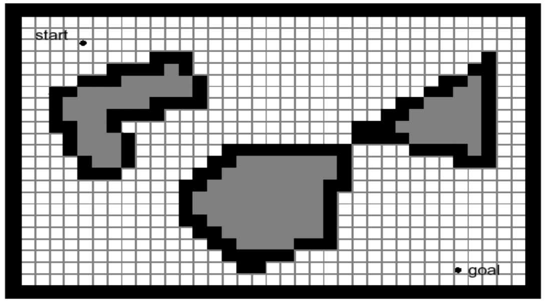
Markov localization | applying probability theory to localization



- Information (measurements) is error prone (uncertain)
  - Odometry
  - Exteroceptive sensors (camera, laser, …)
  - Map

→ Probabilistic map-based localization

Markov localization | basics and assumption



- Discretized pose representation $x_t \rightarrow$ grid map

- Markov localization tracks the robot's belief state $bel(x_t)$ using an arbitrary probability density function to represent the robot's position

- *Markov assumption*: Formally, this means that the output of the estimation process is a function x_t only of the robot's previous state x_{t-1} and its most recent actions (odometry) u_t and perception z_t.

$$p(x_t | x_0, u_t \cdots u_0, z_t \cdots z_0) = p(x_t | x_{t-1}, u_t, z_t)$$

- Markov localization addresses the *global localization problem*, the *position tracking problem*, and the *kidnapped robot problem*.

Markov localization | applying probability theory to localization

- **ACT** | probabilistic estimation of the robot's new belief state $\overline{bel}\ (x_t)$ based on the previous location $bel(x_{t-1})$ and the probabilistic motion model $p(x_t|u_t, x_{t-1})$ with action u_t (control input).

→ application of ***theorem of total probability*** / ***convolution***

$$\overline{bel}(x_t) = \int p(x_t|u_t, x_{t-1}) bel(x_{t-1})\, dx_{t-1}$$ for continuous probabilities

$$\overline{bel}(x_t) = \sum_{x_{t-1}} p(x_t|u_t, x_{t-1}) bel(x_{t-1})$$ for discrete probabilities

Markov localization | applying probability theory to localization

- **SEE** | probabilistic estimation of the robot's new belief state $bel(x_t)$ as a function of its measurement data z_t and its former belief state $\overline{bel}\ (x_t)$:

→ application of ***Bayes rule***

$$bel(x_t) = \eta p(z_t|x_t, M)\overline{bel}(x_t)$$

where $p(z_t|x_t, M)$ is the probabilistic measurement model (SEE), that is, the probability of observing the measurement data z_t given the knowledge of the map M and the robot's position x_t. Thereby $\eta = p(y)^{-1}$ is the normalization factor so that $\sum p = 1$.

Markov localization | the basic algorithms for Markov localization

For all x_t do

$\overline{bel}(x_t) = \sum_{x_{t-1}} p(x_t|u_t, x_{t-1}) bel(x_{t-1})$ (prediction update)

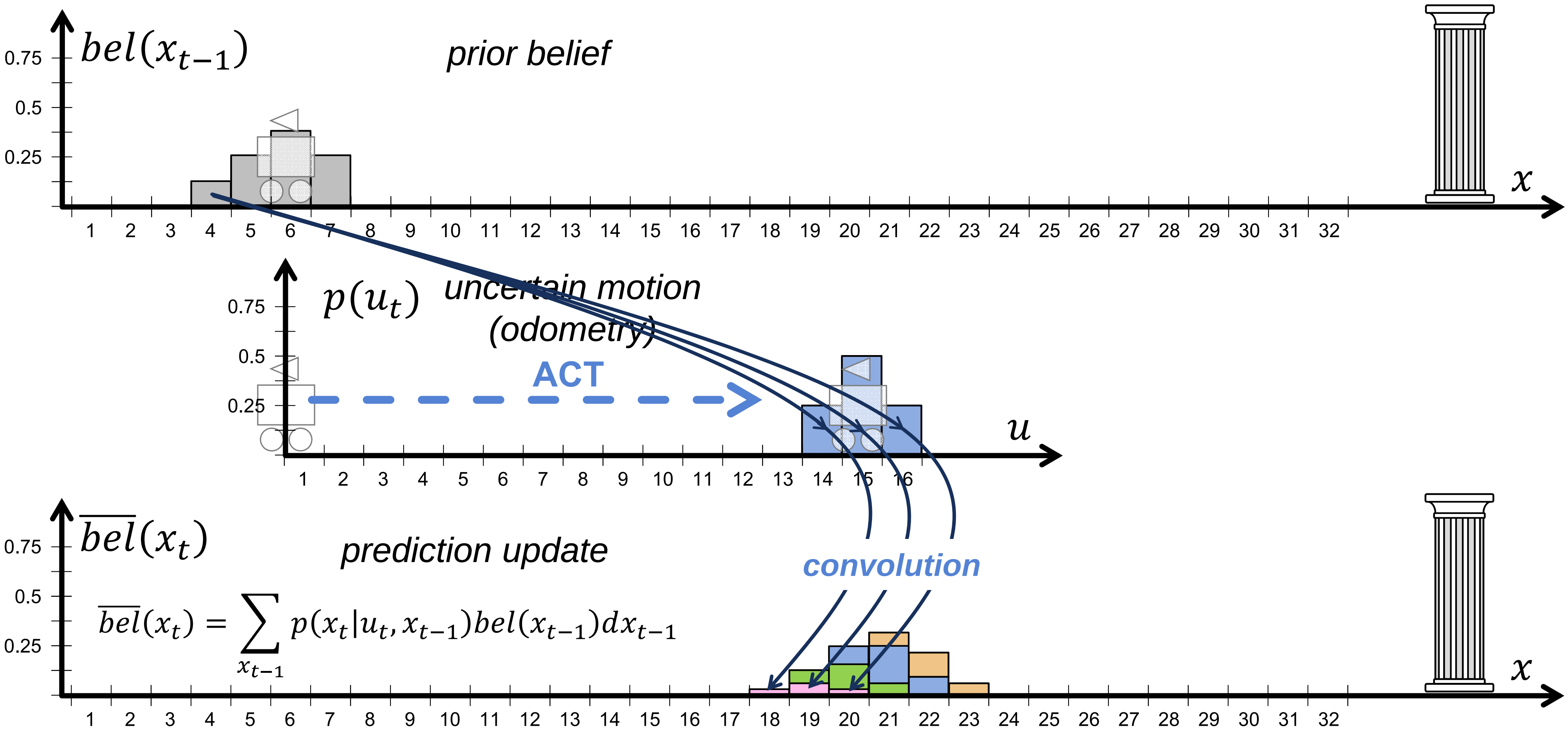
$bel(x_t) = \eta p(z_t|x_t, M)\overline{bel}(x_t)$ (measurement update)

endfor

Return $bel(x_t)$

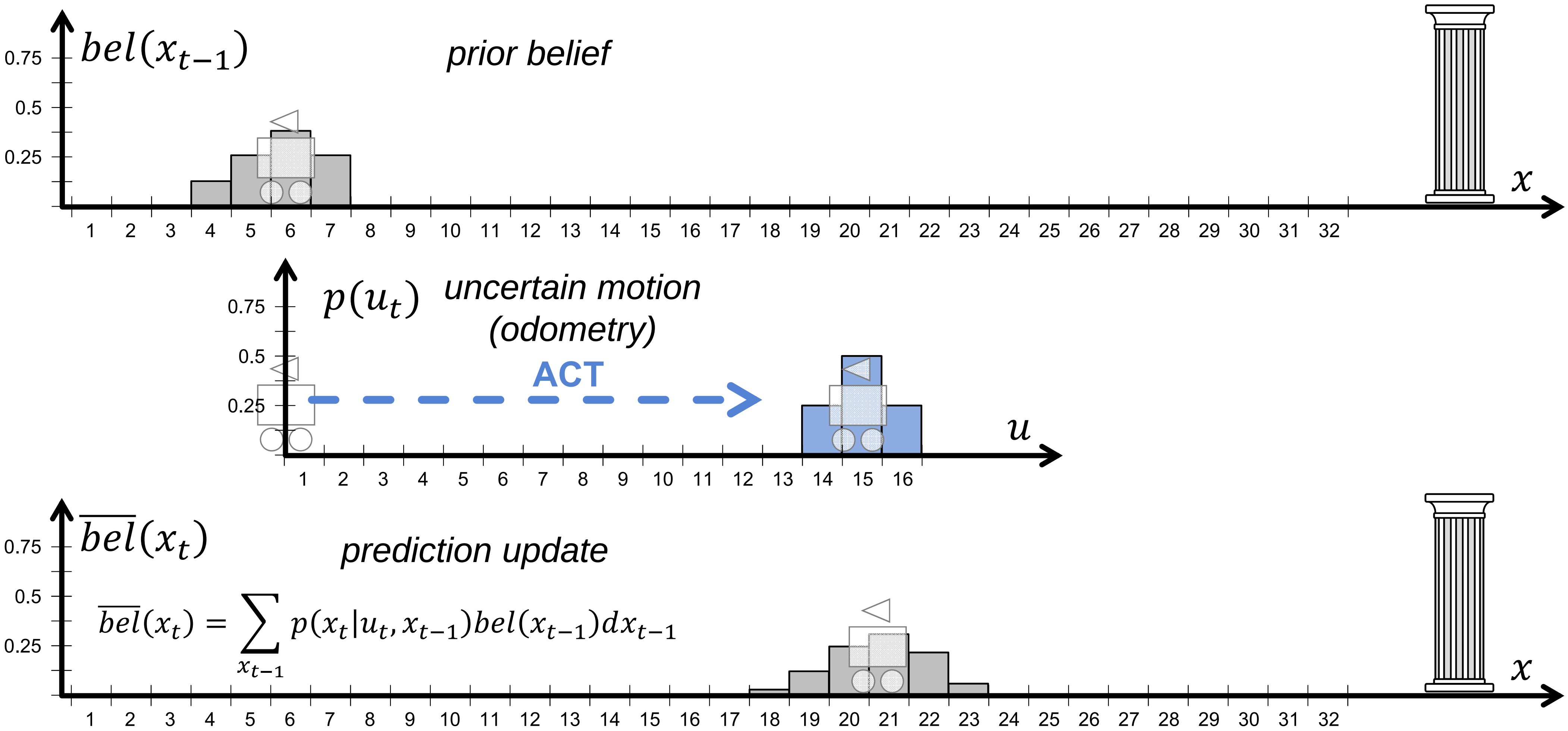
- ***Markov assumption***: Formally, this means that the output is a function x_t only of the robot's previous state x_t and its most recent actions (odometry) u_t and perception z_t.

ACT | using motion model and its uncertainties



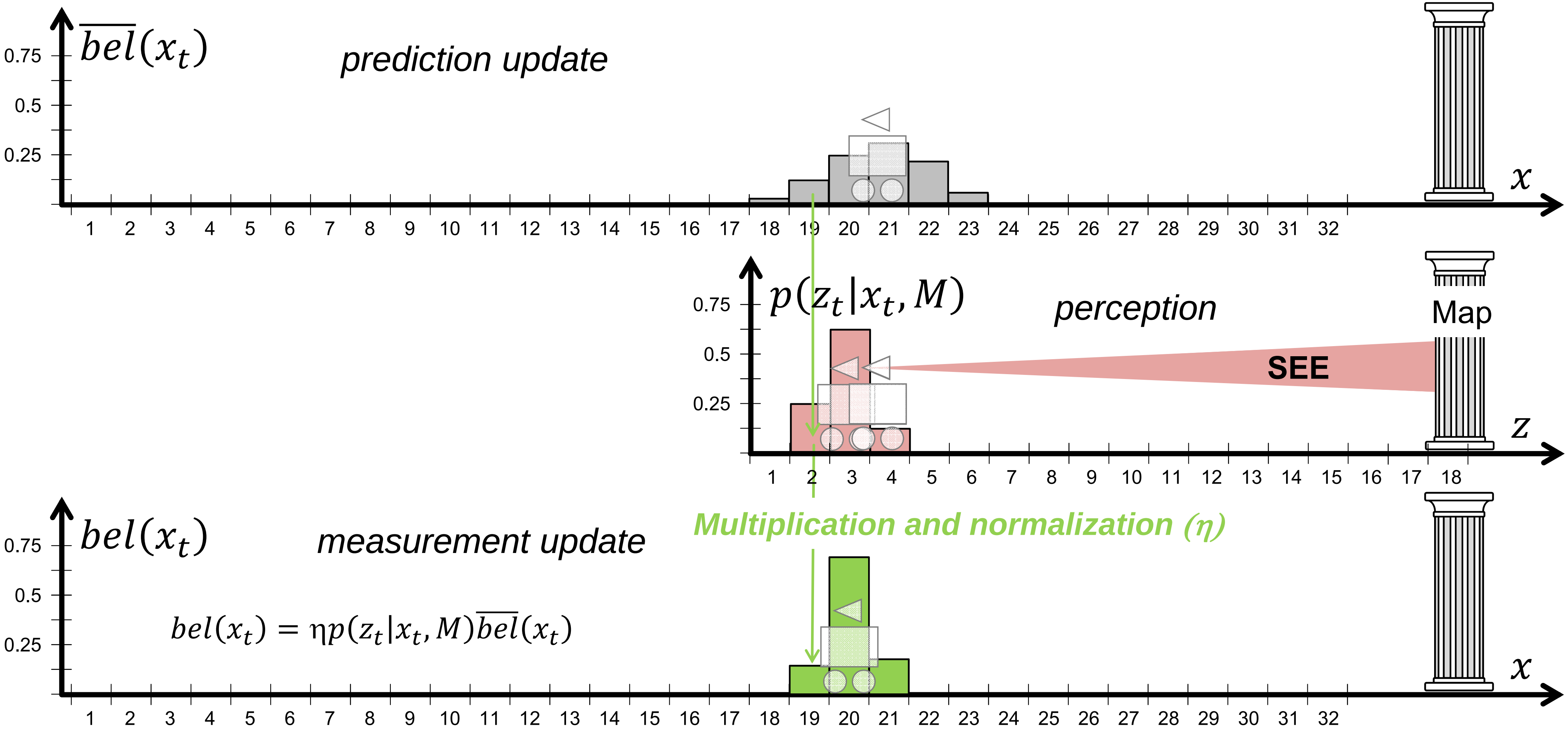
$$\overline{bel}(x_t) = \sum_{x_{t-1}} p(x_t|u_t, x_{t-1}) bel(x_{t-1}) dx_{t-1}$$

ACT | using motion model and its uncertainties



$$\overline{bel}(x_t) = \sum_{x_{t-1}} p(x_t|u_t, x_{t-1}) bel(x_{t-1}) dx_{t-1}$$

SEE | estimation of position based on perception and map



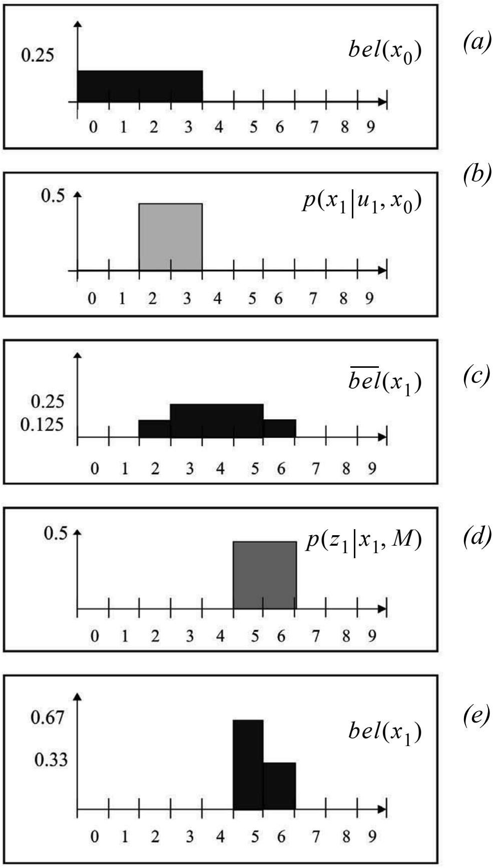


Figure 5.23 Markov localization using a grid-map.

$$p(x_1 = 2) = p(x_0 = 0)p(u_1 = 2) = 0.125, \tag{5.44}$$

$$p(x_1 = 3) = p(x_0 = 0)p(u_1 = 3) + p(x_0 = 1)p(u_1 = 2) = 0.25 \tag{5.45}$$

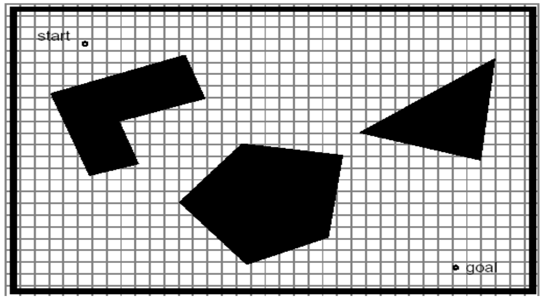
$$p(x_1 = 4) = p(x_0 = 1)p(u_1 = 3) + p(x_0 = 2)p(u_1 = 2) = 0.25 \tag{5.46}$$

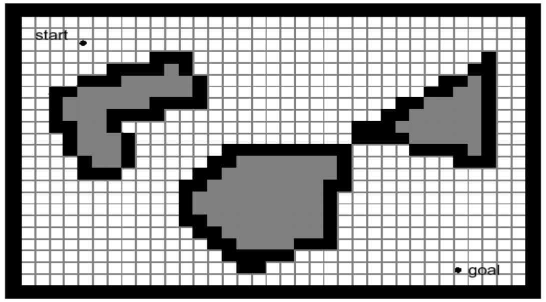
$$p(x_1 = 5) = p(x_0 = 2)p(u_1 = 3) + p(x_0 = 3)p(u_1 = 2) = 0.25 \tag{5.47}$$

$$p(x_1 = 6) = p(x_0 = 3)p(u_1 = 3) = 0.125 \tag{5.48}$$

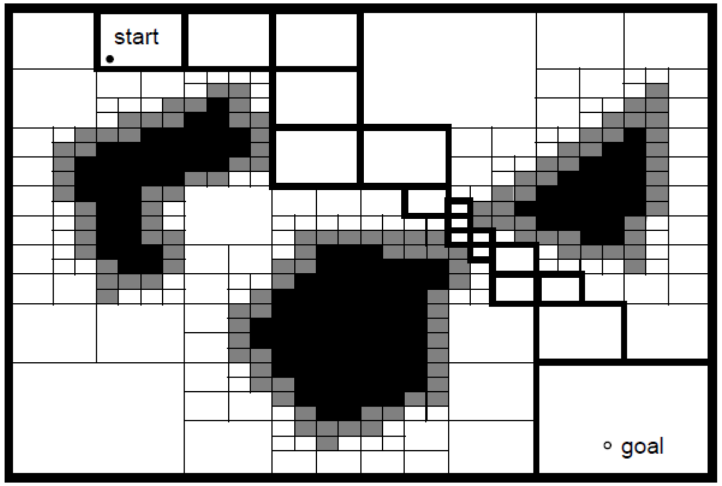
Markov localization | extension to 2D

- The real world for mobile robot is at least 2D (moving in the plane)
  - → discretized pose state space (grid) consists of x, y, θ
  - → Markov Localization scales badly with the size of the environment
- Space: 10 m x 10 m with a grid size of 0.1 m and an angular resolution of 1°
  - → $100 \cdot 100 \cdot 360 = 3.6\ 10^6$ grid points (states)
  - → prediction step requires in worst case $(3.6\ 10^6)^2$ multiplications and summations
- Fine fixed decomposition grids result in a huge state space
  - Very important processing power needed
  - Large memory requirement





Markov localization | reducing computational complexity

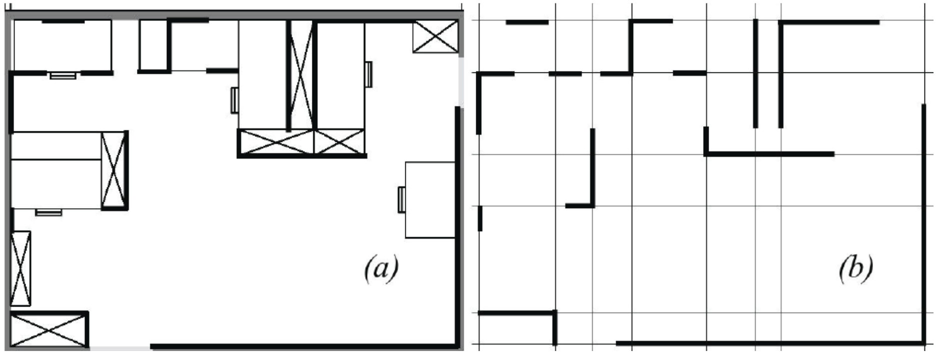


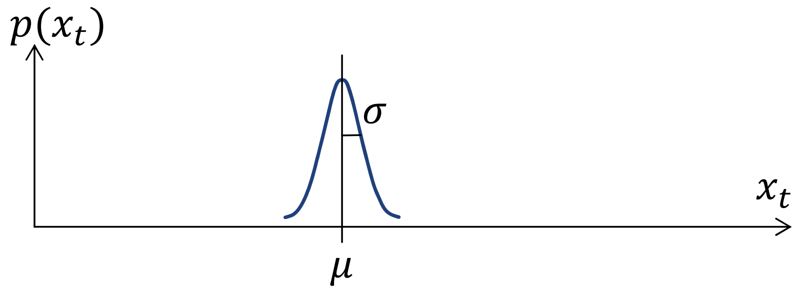
- Adaptive cell decomposition
- Motion model (Odomety) limited to a small number of grid points
- Randomized sampling
  - Approximation of belief state by a representative subset of possible locations
  - weighting the sampling process with the probability values
  - Injection of some randomized (not weighted) samples

  - randomized sampling methods are also known as particle filter algorithms, condensation algorithms, and Monte Carlo algorithms.

Kalman Filter Localization | Basics and assumption

- Continuous pose representation x_t
- Kalman Filter Assumptions:
  - Error approximation with normal distribution: $x = N(\mu, \sigma^2)$ (Gaussian model)
  - Output y_t distribution is a linear (or linearized) function of the input distribution: $y = Ax_1 + Bx_2$
- Kalman filter localization tracks the robot's belief state $p(x_t)$ typically as a single hypothesis with normal distribution.
- Kalman localization thus addresses the *position tracking problem*, but ***not*** the *global localization* or the *kidnapped robot problem*.





Kalman Filter Localization | in summery

1. **Prediction (ACT)** based on previous estimate and odometry
2. **Observation (SEE)** with on-board sensors
3. **Measurement prediction** based on prediction and map
4. **Matching** of observation and map
5. **Estimation** → position update (posteriori position)