

# Principles of Robot Autonomy I

Non-parametric filtering



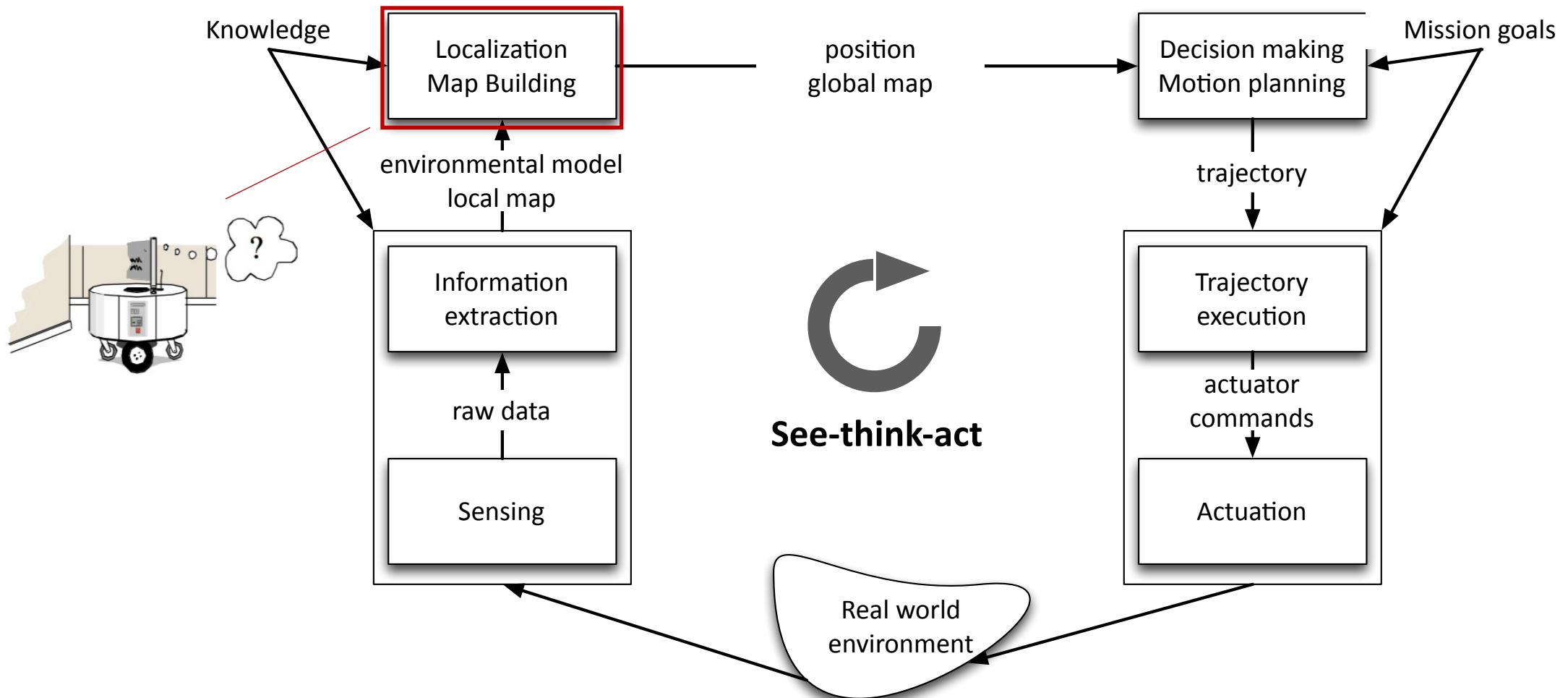
**Stanford**  
University



# Logistics

- HW3 due tomorrow (Friday, 11/5)
- HW4, final project description out later today/early tomorrow
- Note on group problems: intended to be completed collaboratively (i.e., not parted out/parallelized)

# Module 3



# Today's lecture

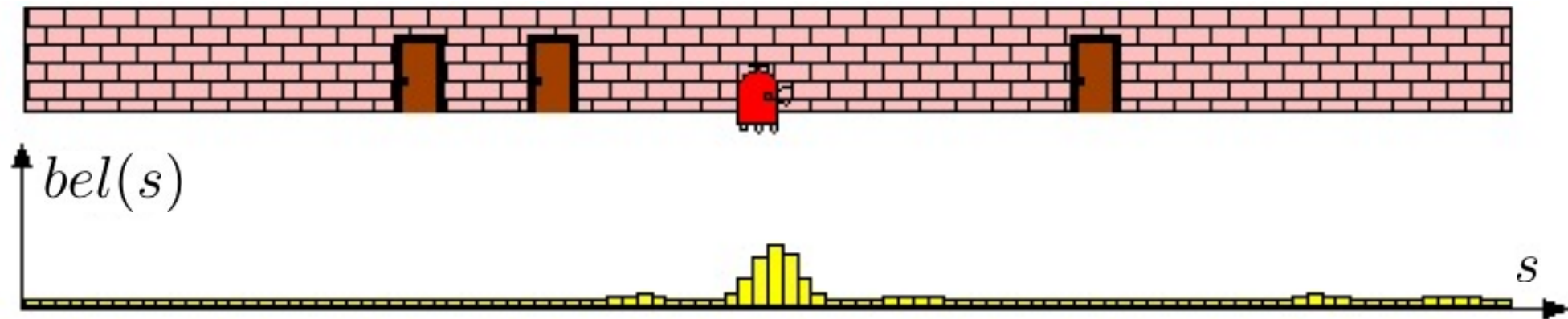
- Aim
  - Learn about non-parametric filters
- Readings
  - S. Thrun, W. Burgard, and D. Fox. Probabilistic robotics. MIT press, 2005. Sections 3.1 – 3.4, 4.1, 4.3, 7.1

# Instantiating the Bayes' filter

- Tractable implementations of Bayes' filter exploit structure and / or approximations; two main classes
  - Parametric filters: e.g., **KF**, **EKF**, UKF, etc.
  - Non parametric filters: e.g., **histogram filter**, **particle filter**, etc.

# Histogram filter

- **Key idea:** use *discrete* Bayes' filter as an approximate inference tool for *continuous* state spaces



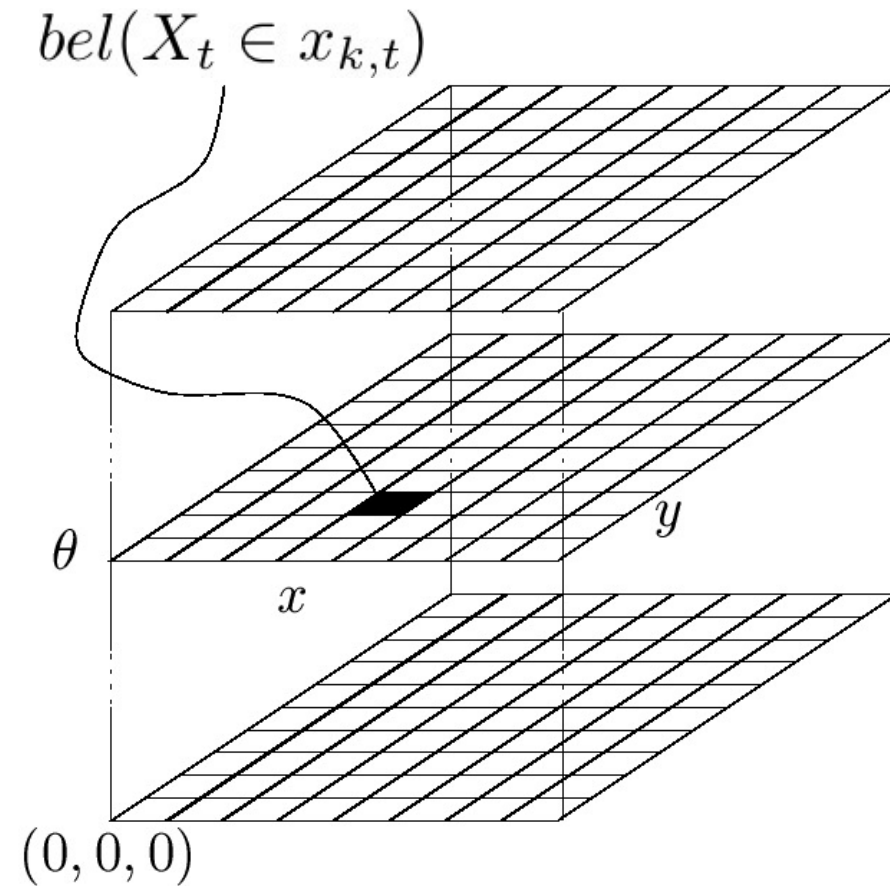
- Step #1: histogram filters decompose a continuous space into finitely many bins

$$\text{dom}(X_t) = x_{1,t} \cup x_{2,t} \cup \dots \cup x_{K,t}$$

↖ State space

$\{x_{k,t}\}$ : convex regions forming a partition of state space (e.g., grid cell)

# Example



# Histogram filter

- Step #2: assign to each region  $x_{k,t}$  a probability  $p_{k,t}$ ; probabilities are then approximated according to a piecewise scheme

$$p(x_t) \equiv \frac{p_{k,t}}{|x_{k,t}|}, \quad \text{for all } x_t \in x_{k,t} \quad \Rightarrow \quad p(X_t \in x_{k,t}) = \int_{x_{k,t}} \frac{p_{k,t}}{|x_{k,t}|} dx_t = p_{k,t}$$



# Histogram filter

- Step #3: discretize motion and measurements models, i.e.,

$$p(x_t | u_t, x_{t-1}) \quad \text{and} \quad p(z_t | x_t)$$

1. Select mean state as representative state

$$\hat{x}_{k,t} = |x_{k,t}|^{-1} \int_{x_{k,t}} x_t dx_t$$

2. Approximate measurement model

$$p(z_t | x_{k,t}) \approx p(z_t | \hat{x}_{k,t})$$

3. Approximate transition model

$$p(x_{k,t} | u_t, x_{i,t-1}) \approx \eta |x_{k,t}| p(\hat{x}_{k,t} | u_t, \hat{x}_{i,t-1})$$

- Step #4: execute discrete Bayes' filter with discretized probabilities

# Histogram filter

- Then one can run the usual discrete Bayes' filter

- Belief  $bel(x_t)$   
represented as pmf  
 $\{p_{k,t}\}$

**Data:**  $\{p_{k,t-1}\}, u_t, z_t$

**Result:**  $\{p_{k,t}\}$

**foreach**  $k$  **do**

$$\left| \bar{p}_{k,t} = \sum_i p(X_t = x_k | u_t, X_{t-1} = x_i) p_{i,t-1}; \right.$$

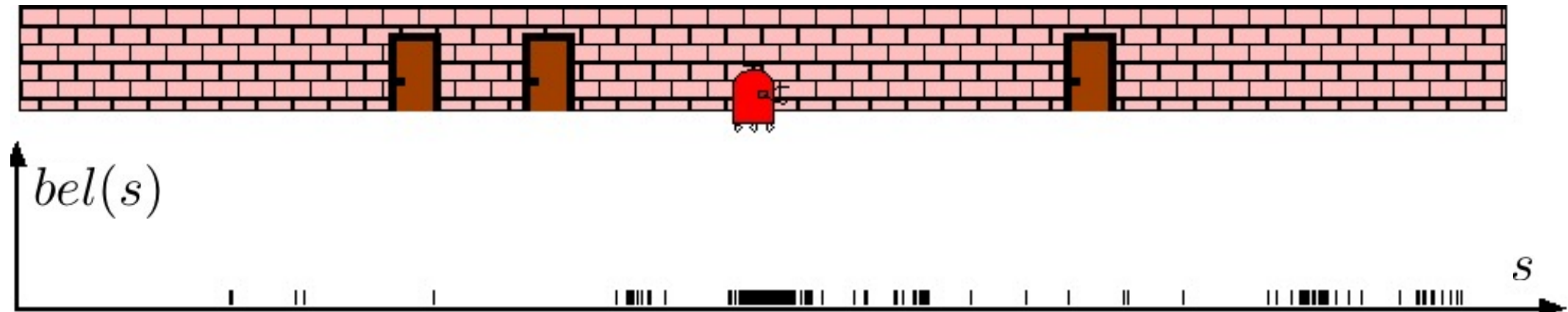
$$\left| p_{k,t} = \eta p(z_t | X_t = x_k) \bar{p}_{k,t}; \right.$$

**end**

**Return**  $\{p_{k,t}\}$

# Particle filter

- **Key idea:** represent posterior  $bel(x_t)$  by a set of random samples



- Allows one to represent non-Gaussian distributions and handle nonlinear transformations in a direct way

# Particle filter

- Samples of posterior distribution are called *particles*, denoted as

$$\mathcal{X}_t := x_t^{[1]}, x_t^{[2]}, \dots, x_t^{[M]}$$

- A particle represents a hypothesis about what the true world state might be at time  $t$
- Ideally, particles should be distributed according to

$$x_t^{[m]} \sim p(x_t | z_{1:t}, u_{1:t}) = \text{bel}(x_t)$$

- Matching exact only as  $M \rightarrow \infty$ , but  $M \approx 1000$  usually good enough
- A particle filter constructs the particle set  $\mathcal{X}_t$  from the particle set  $\mathcal{X}_{t-1}$  recursively, with the goal of matching the distribution  $\text{bel}(x_t)$

# Particle filter: algorithm

- The temporary particle set  $\bar{\mathcal{X}}_t$  represents the belief  $\overline{bel}(x_t)$
- The particle set  $\mathcal{X}_t$  represents the belief  $bel(x_t)$
- Importance factors are used to incorporate measurement  $z_t$  in the particle set
- After resampling, particles are (as  $M \rightarrow \infty$ ) distributed as

$$bel(x_t) = \eta p(z_t | x_t^{[m]}) \overline{bel}(x_t)$$

$bel(x_{t-1})$

**Data:**  $\mathcal{X}_{t-1}, u_t, z_t$   
**Result:**  $\mathcal{X}_t$   
 $\bar{\mathcal{X}}_t = \mathcal{X}_t = \emptyset;$

**for**  $i = 1$  **to**  $M$  **do**

Prediction:  $\overline{bel}(x_t)$  {

Sample  $x_t^{[m]} \sim p(x_t | u_t, x_{t-1}^{[m]});$

Importance factor  $w_t^{[m]} = p(z_t | x_t^{[m]});$

$\bar{\mathcal{X}}_t = \bar{\mathcal{X}}_t \cup (x_t^{[m]}, w_t^{[m]});$

**end**

Correction:  $bel(x_t)$  {

**for**  $m = 1$  **to**  $M$  **do**

Draw  $i$  with probability  $\propto w_t^{[i]};$

Add  $x_t^{[i]}$  to  $\mathcal{X}_t;$

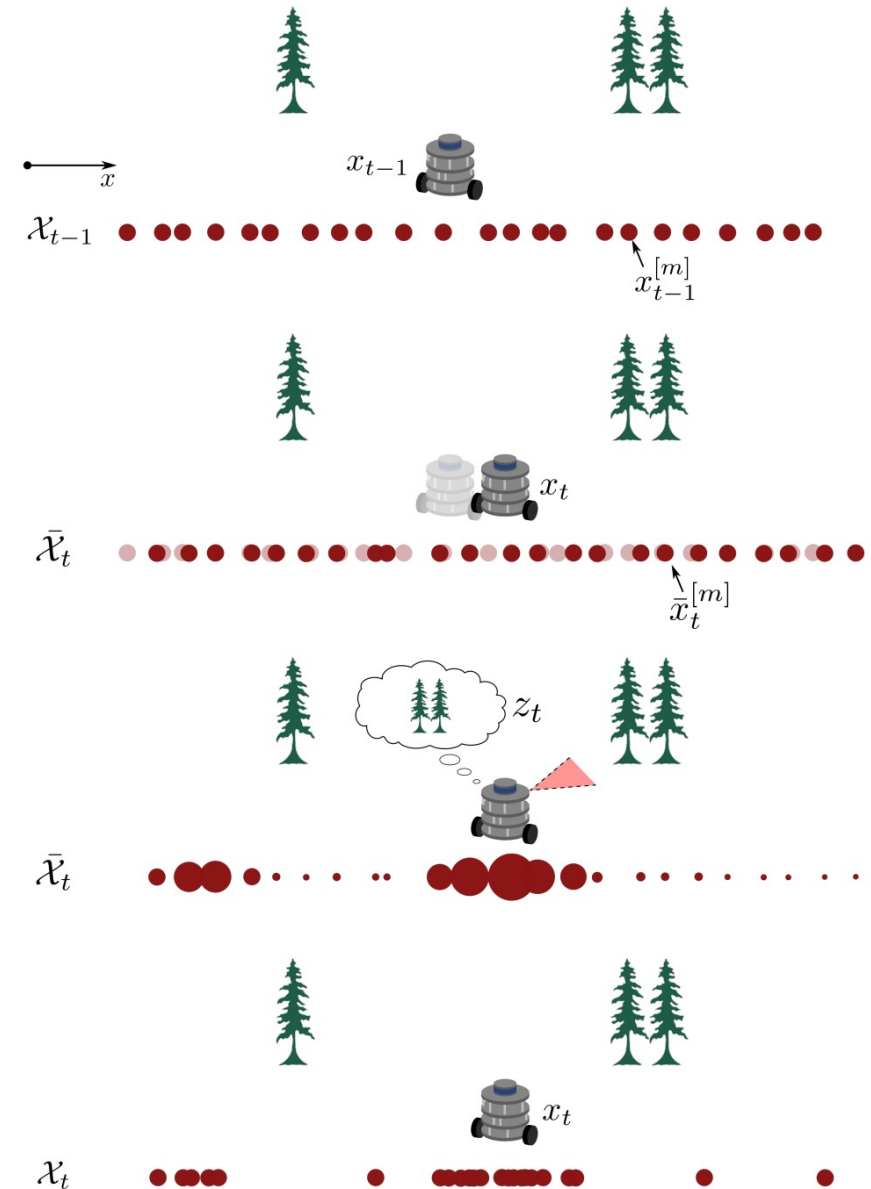
**end**

Return  $\mathcal{X}_t$

$bel(x_t)$

# Particle filter: example

- Resampling can be a high variance process (e.g., “weight collapse” can be a problem) motivating the development of lower variance schemes and/or recovery processes
- Many extensions/variants (e.g., Gaussian Sum Particle Filtering in which belief is represented as a Gaussian Mixture Model)



# Next time

