Localization

CS 3630 Intro to Perception and Robotics

Frank Dellaert

Localization

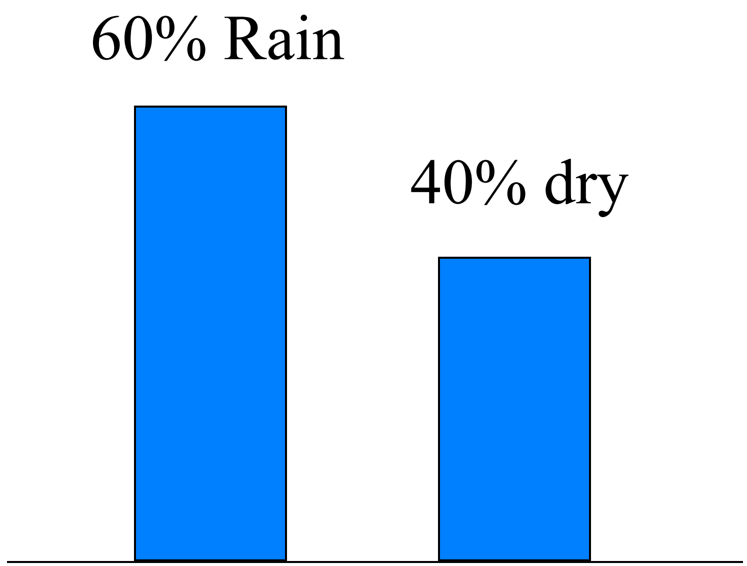
- Pretty old concept ☺

- Outdoors
  - Then:
    - Dead reckoning
    - sky
  - Earlier this century: radio
  - Now: GPS

- Indoors:
  - ?
  - Oculus Rift

The Bayesian Paradigm

- Knowledge as a probability distribution



Probabilities as Knowledge

Question:

Rain tomorrow? 20%, sunny today, just raining

(DATA) (PRIOR MODEL)

Rain OCT12? 70%, farmers *know*

PRIOR = learned from experience

This is a *Binary* Event:

100% = certainty

0% = will *not* rain

Bayes Rule

Conditional Probability

$$P(R \mid T) = \frac{P(R \cap T)}{P(T)}$$

$$P(T \mid R) = \frac{P(R \cap T)}{P(R)}$$

Bayes Rule

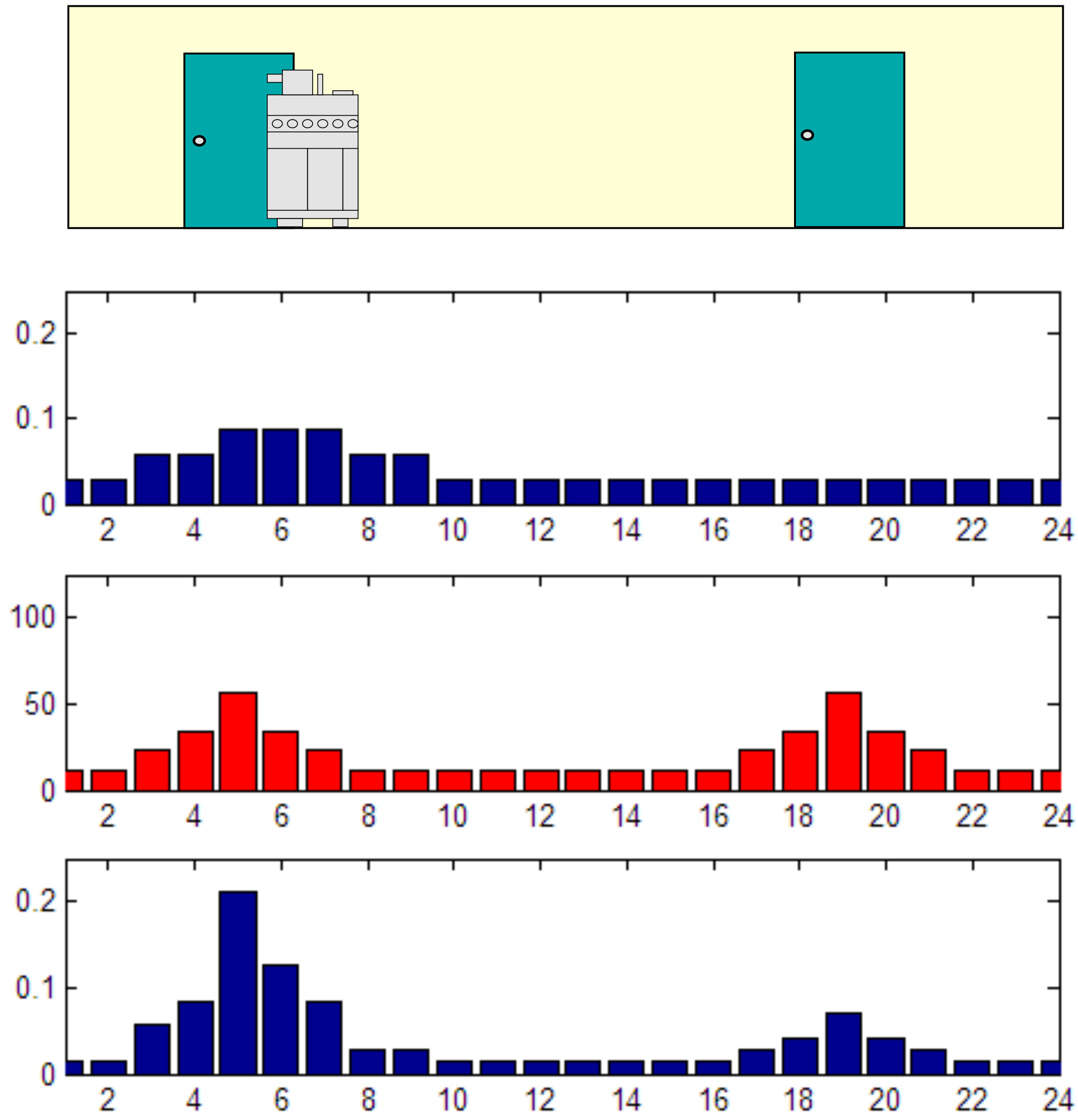
$$P(R \mid T) = \frac{P(T \mid R)P(R)}{P(T)}$$

X: state Z: measurement

$$P(X \mid Z) = \frac{P(Z,X)}{P(Z)} = \frac{P(Z \mid X)P(X)}{P(Z)}$$

$$= \frac{P(Z \mid X)P(X)}{\sum_{X'} P(Z \mid X')P(X')} \propto P(Z \mid X)P(X)$$

1D Robot Example



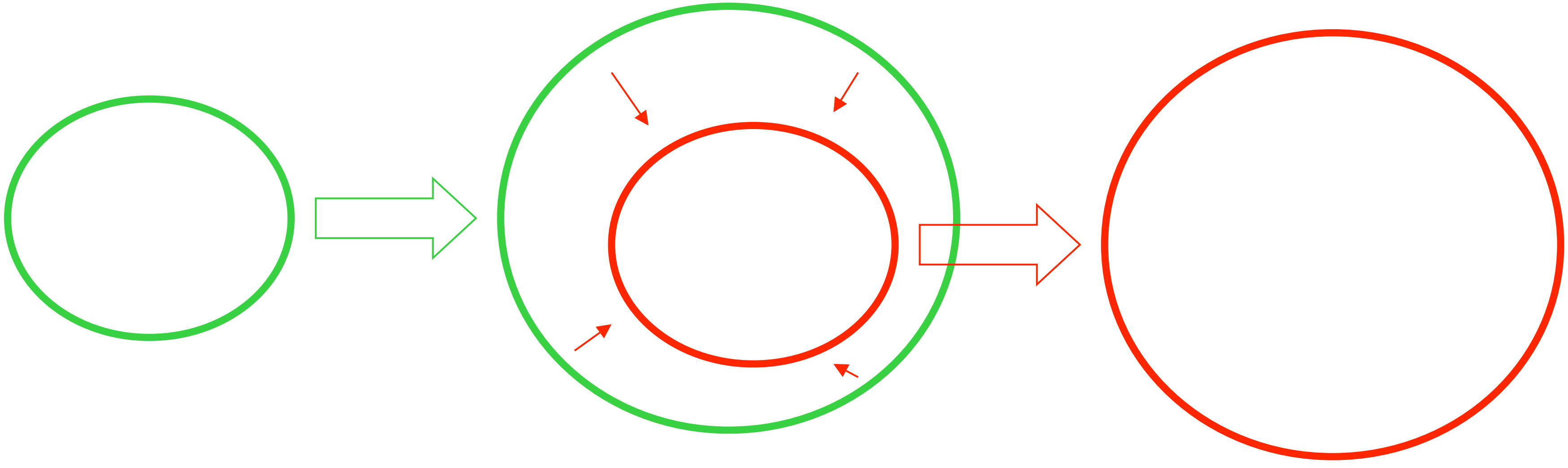
Prior P(X)

Likelihood L(X;Z)

Posterior P(X|Z)

Bayesian Filtering

- Two phases:
  - 1. Prediction Phase
  - 2. Measurement Phase



Bayes Filter Equations

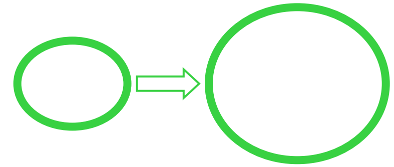
Motion Model

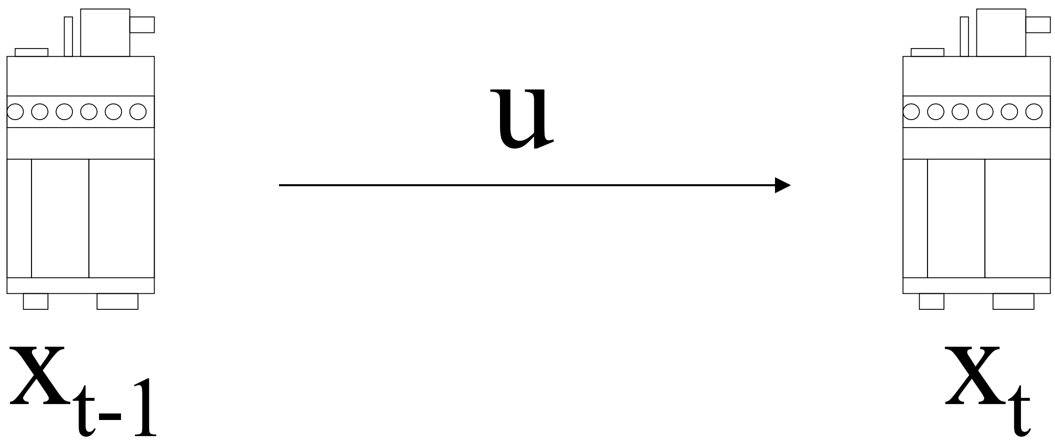
Recursive Bayes Filter Equation:

$$P(X_t|Z^t) = kP(Z_t|X_t)\int_{X_{t-1}} P(X_t|X_{t-1})P(X_{t-1}|Z^{t-1})$$

Predictive Density

1. Prediction Phase

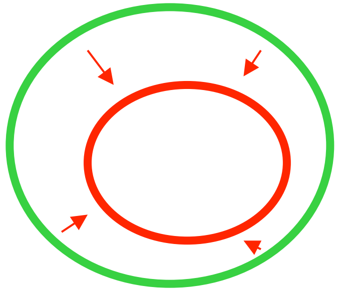


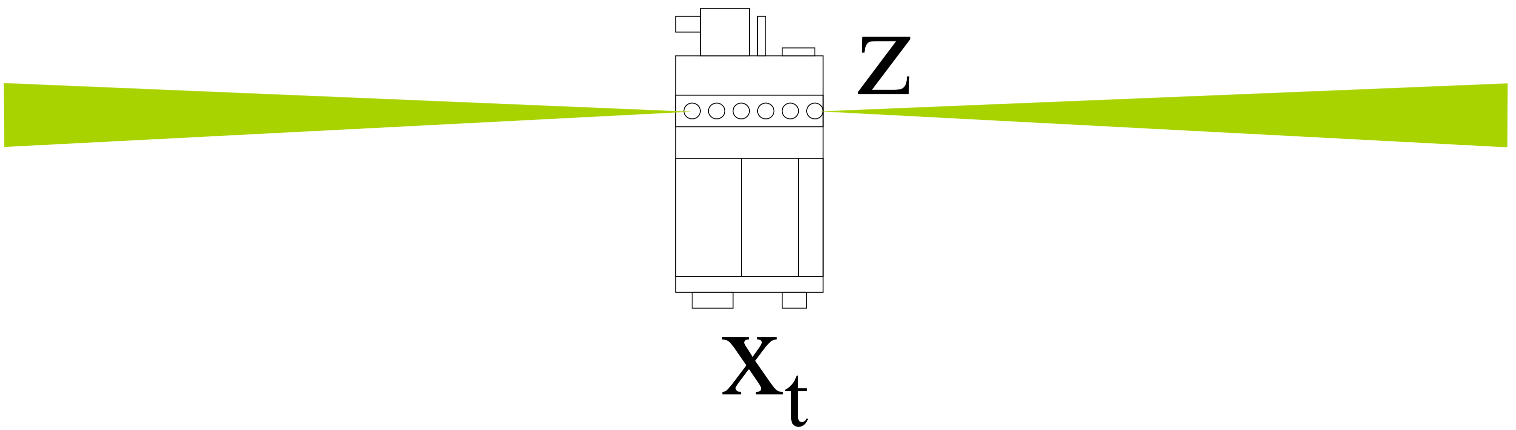


$$P(x_t) = \sum P(x_t|x_{t-1},u)\, P(x_{t-1})$$

Motion Model

2. Measurement Phase

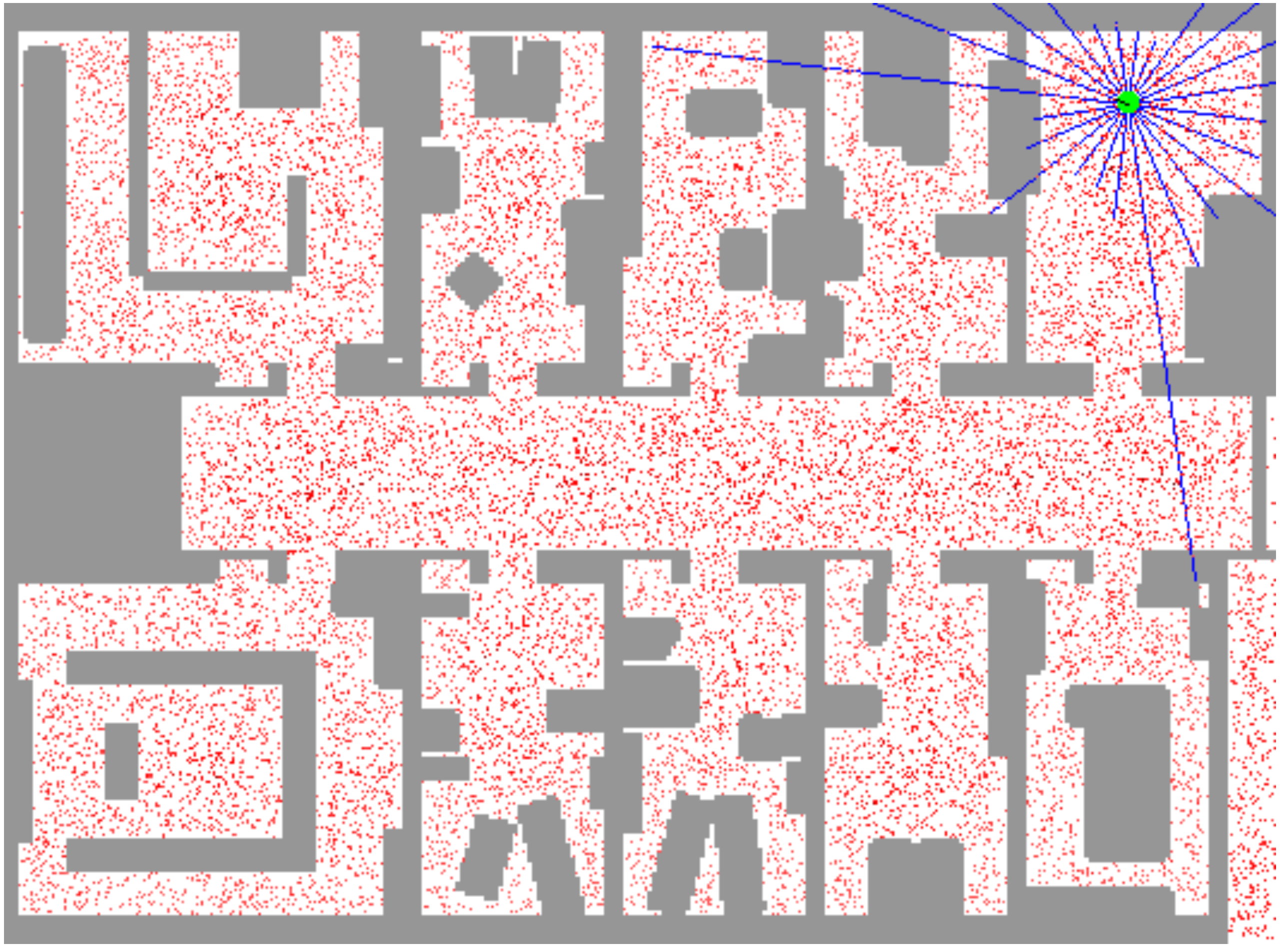




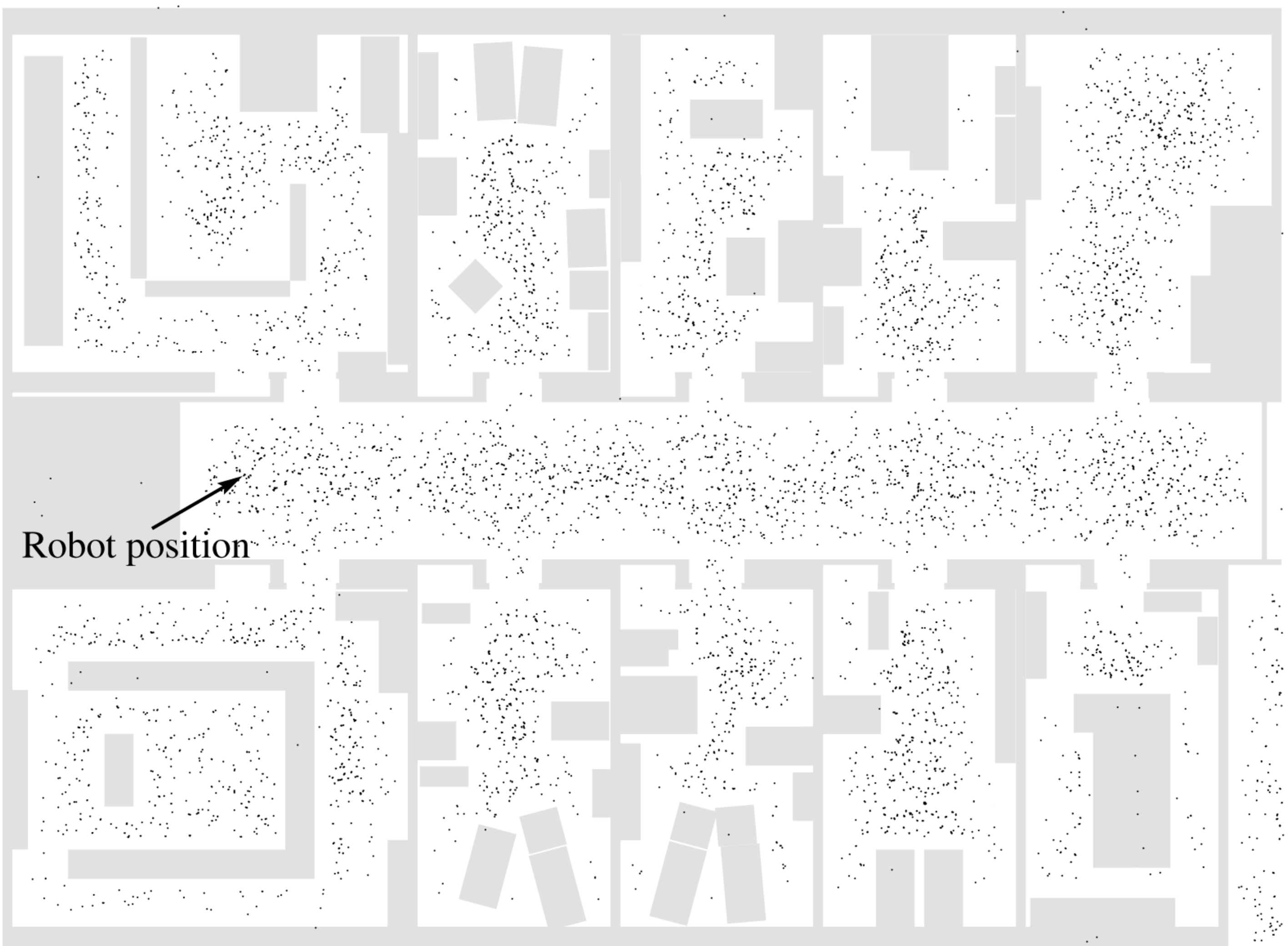
$$P(x_t|z) = k\, P(z|x_t)\, P(x_t)$$

Sensor Model

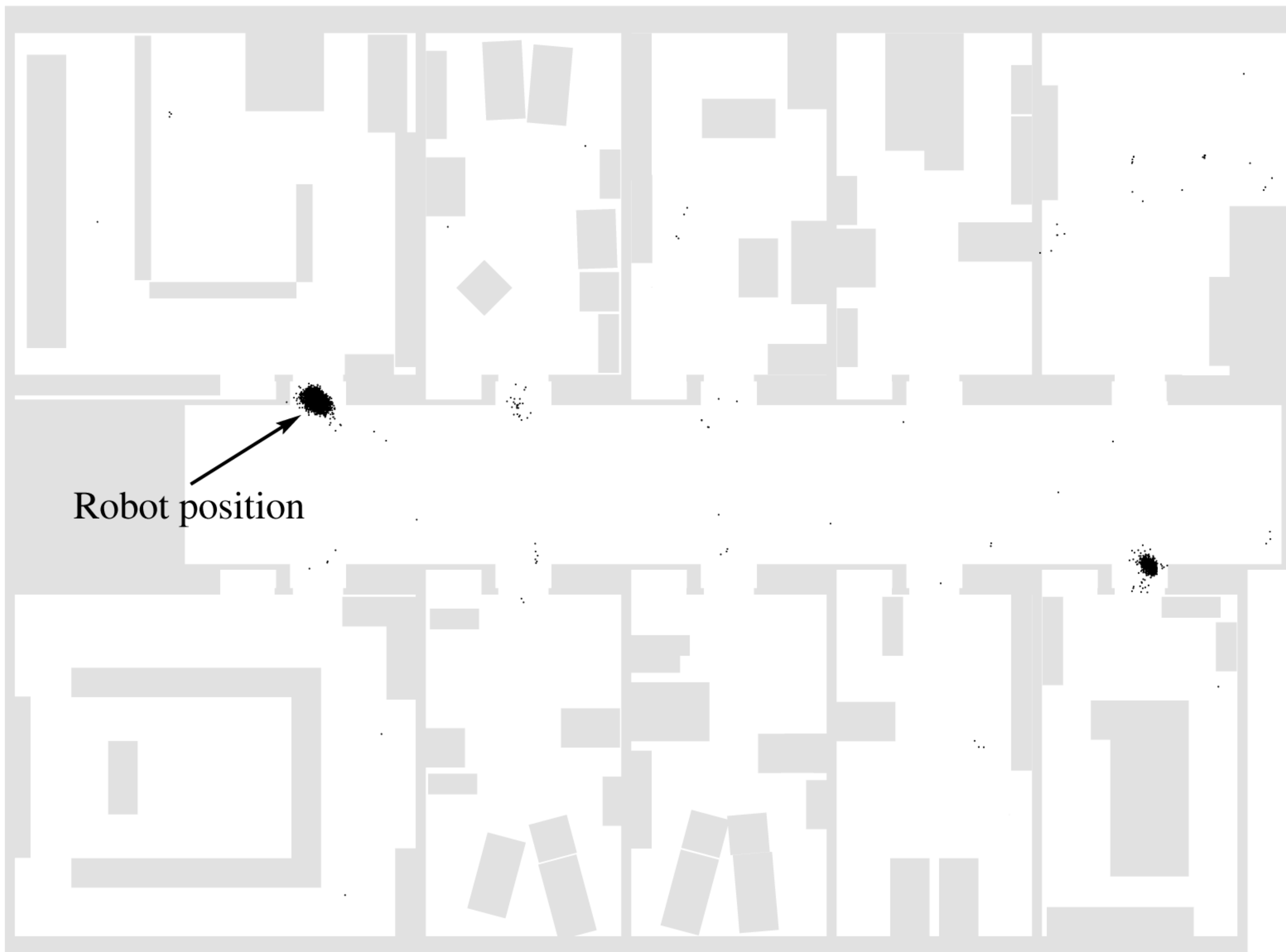
Animation



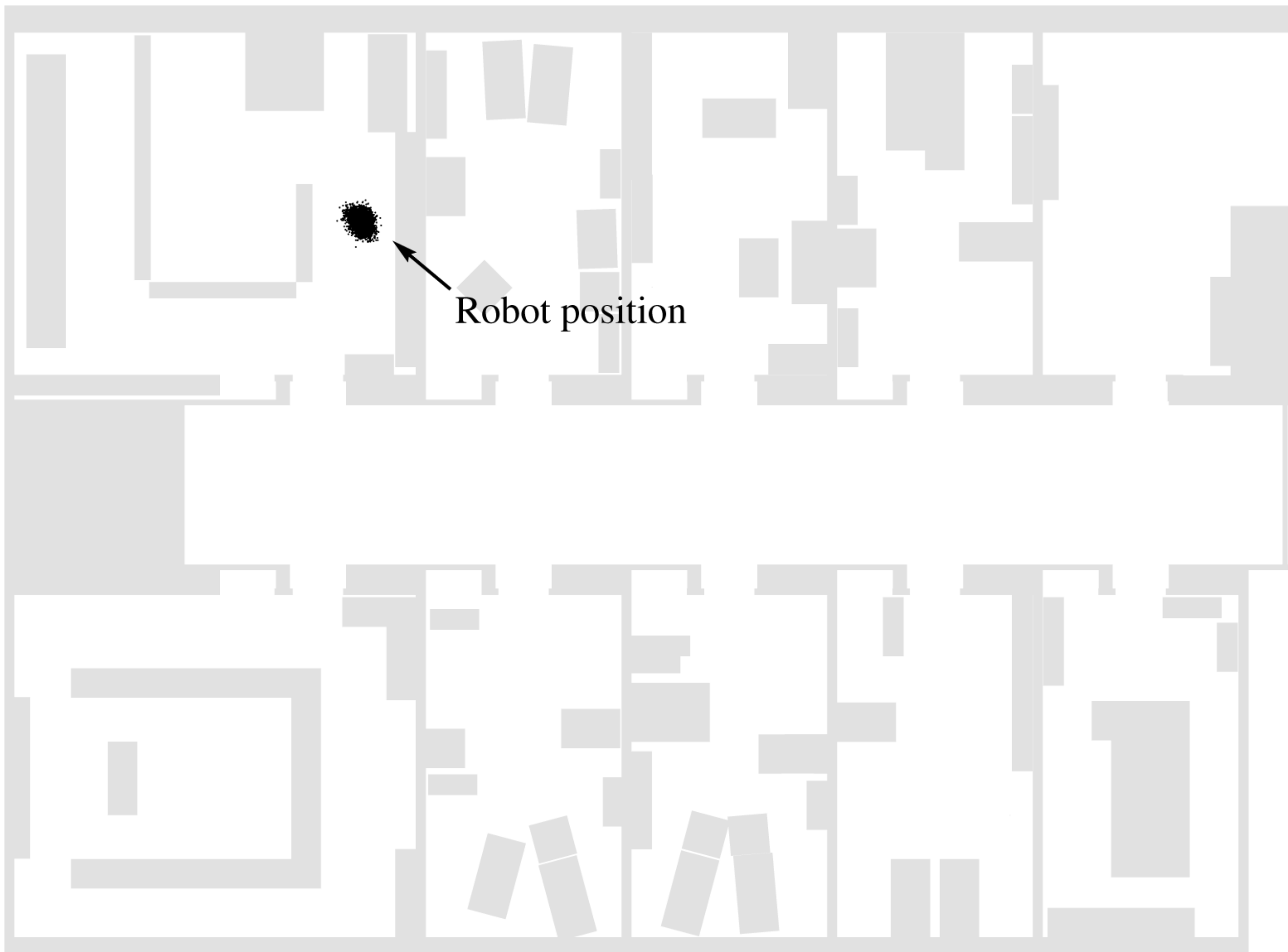
Global Localization



Global Localization (2)

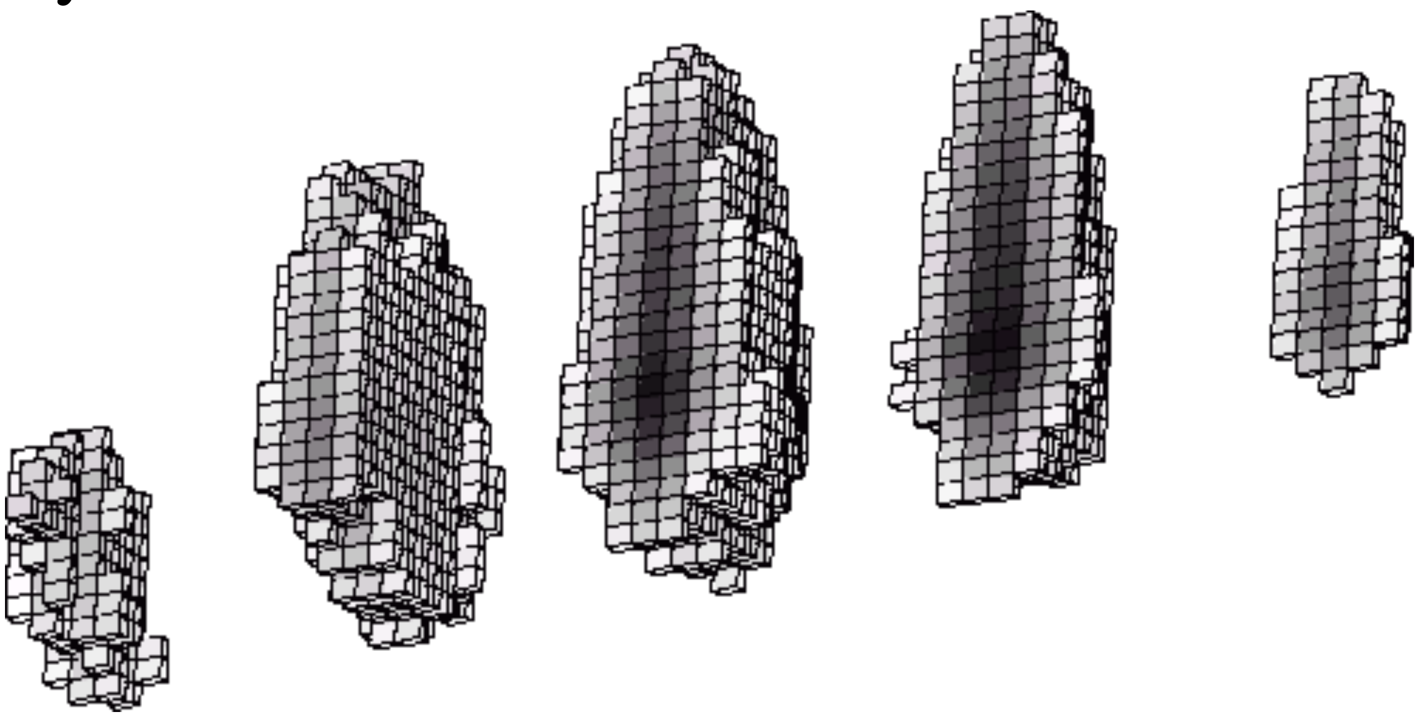


Global Localization (3)



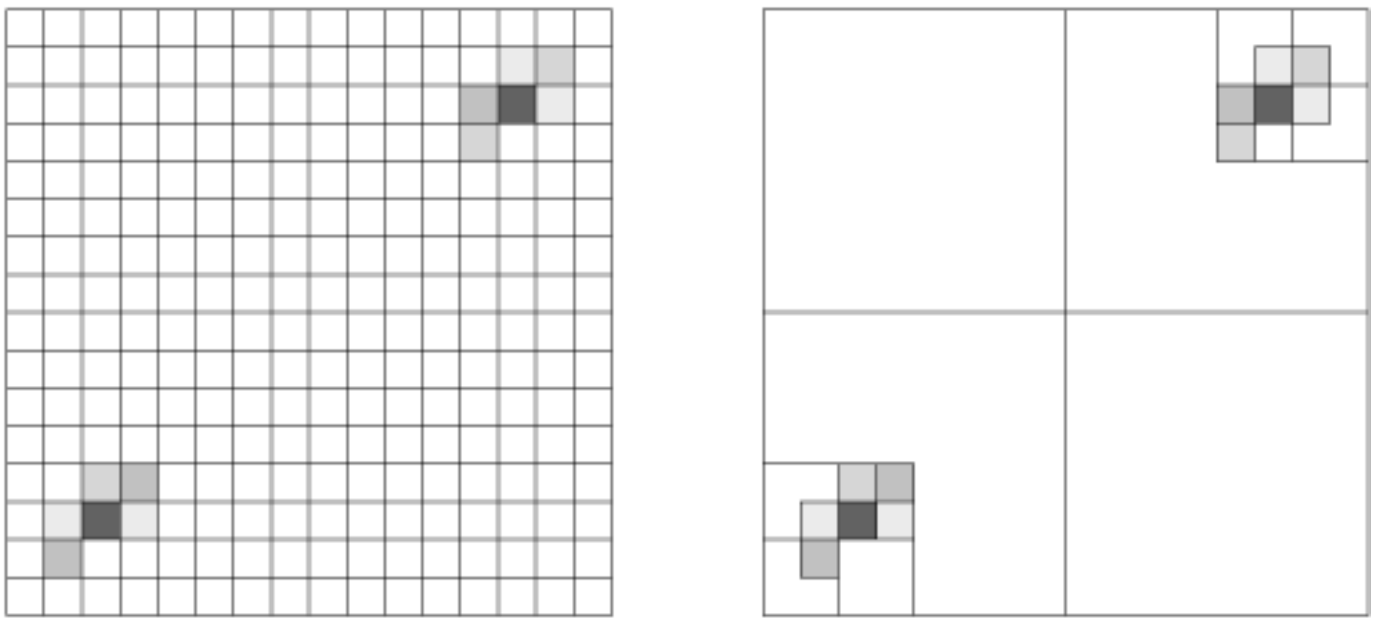
Markov Localization

- Fine discretization over {x,y,theta}
- Very successful: Rhino, Minerva, Xavier…

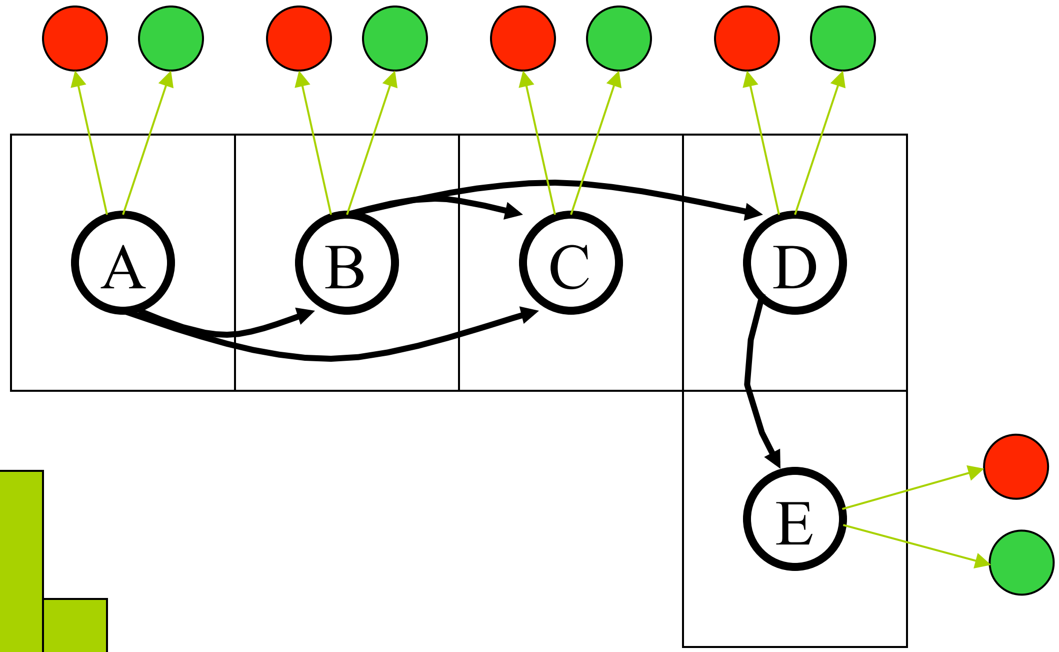


Dynamic Markov Localization

- Burgard et al., IROS 98
- Idea: use Oct-trees



Hidden Markov Models

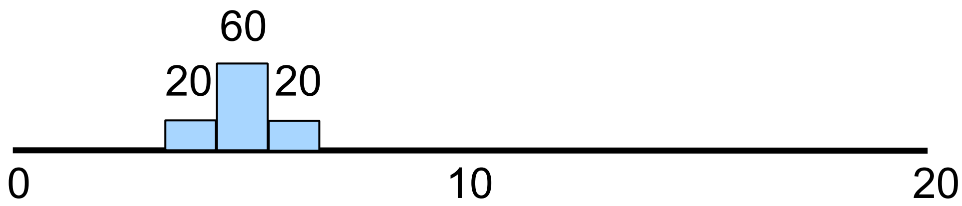


Discrete vs. Continuous

$$P(0 < x < 20) = 1$$

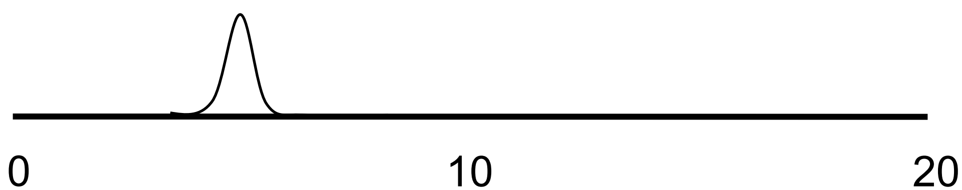
$$P(0 < x < 10) = 1$$

$$\sum_{i=1}^{20} P(I_i) = 1$$



P: Probability ***Distribution***

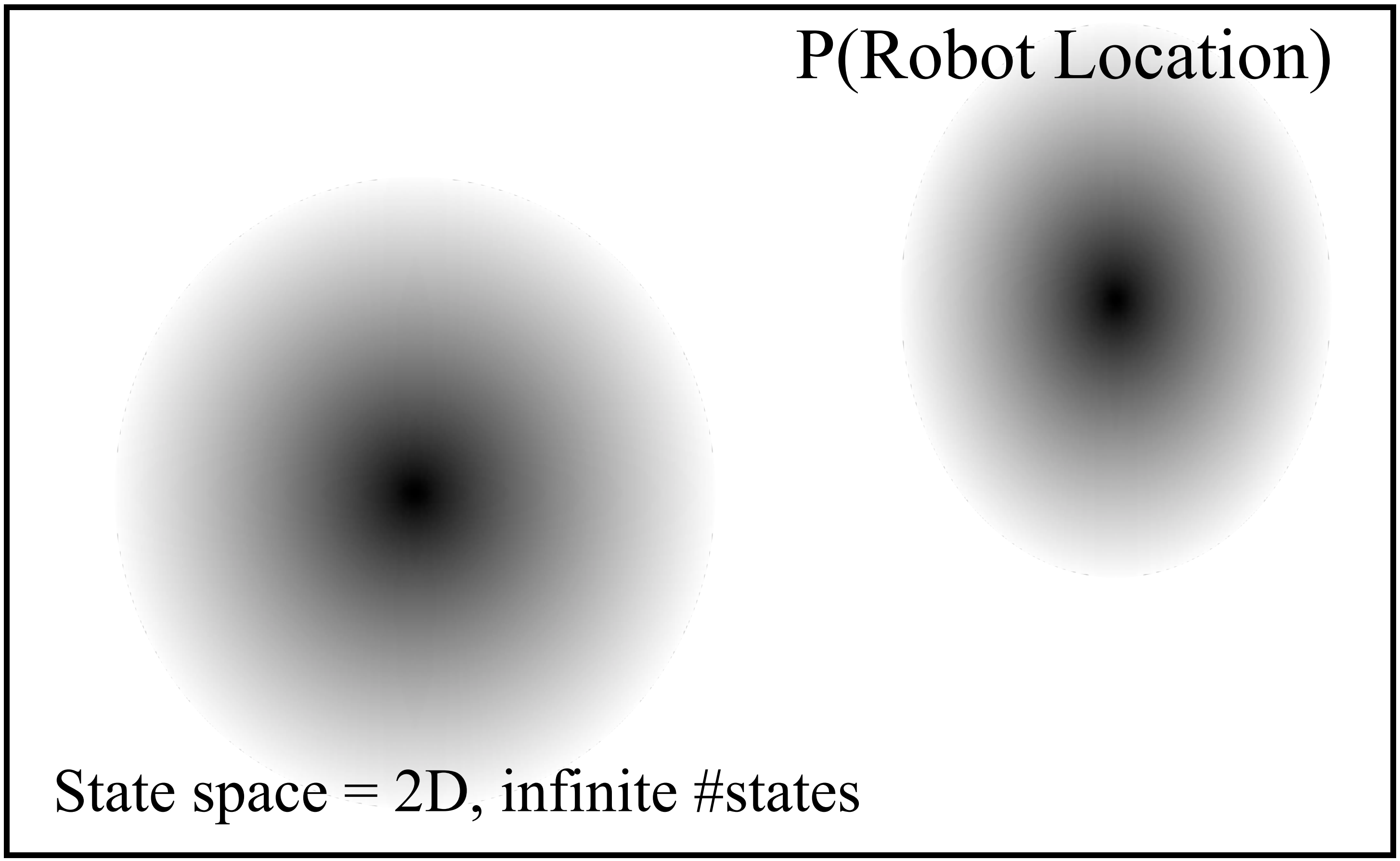
$$\int_{x=0}^{20} p(x)dx = 1$$



p: Probability ***Density***

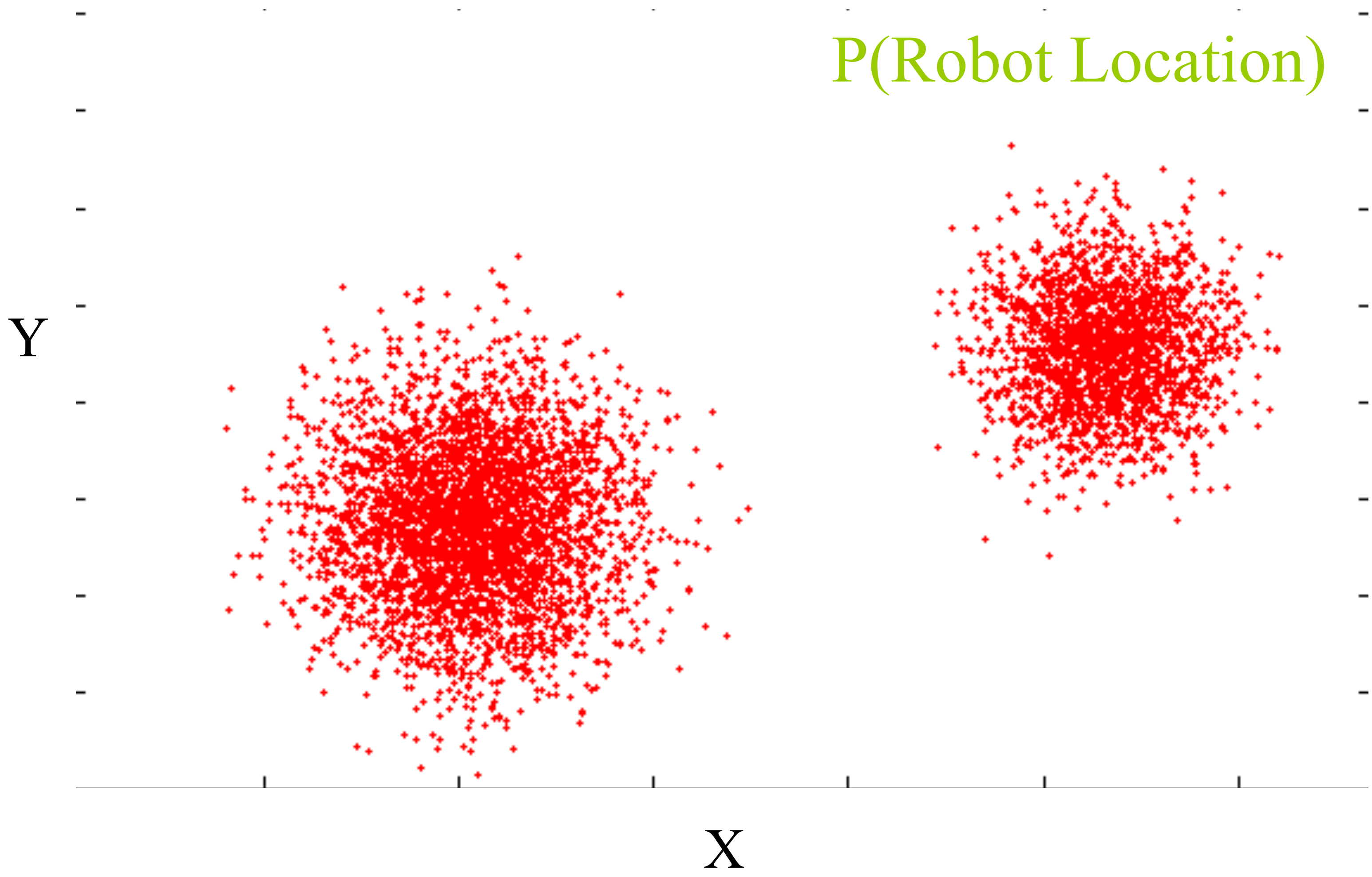
Probability of Robot Location

Y



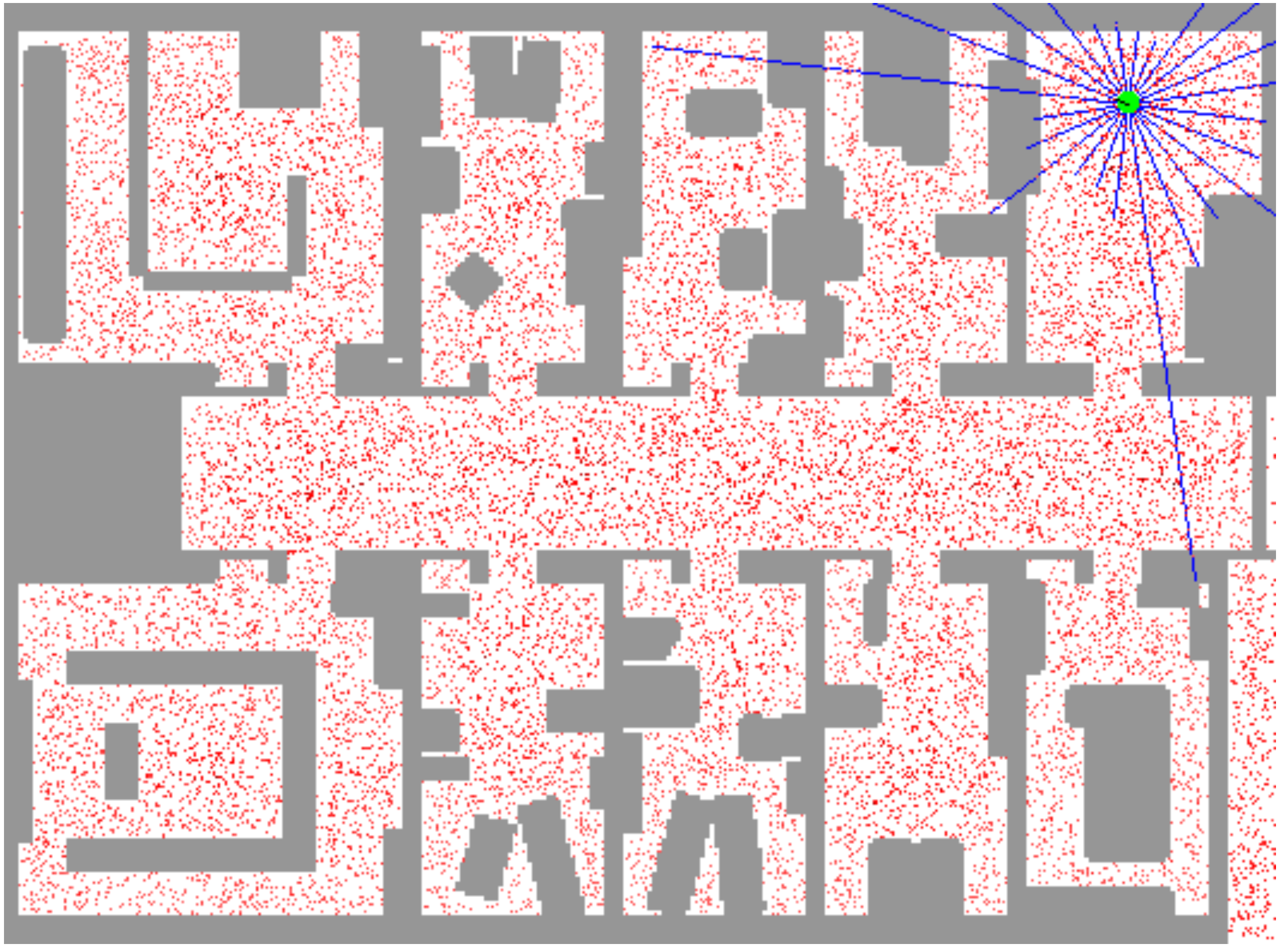
X

Sampling as Representation



3D Particle filter for robot pose: Monte Carlo Localization

Dellaert, Fox & Thrun ICRA 99



Sampling Advantages

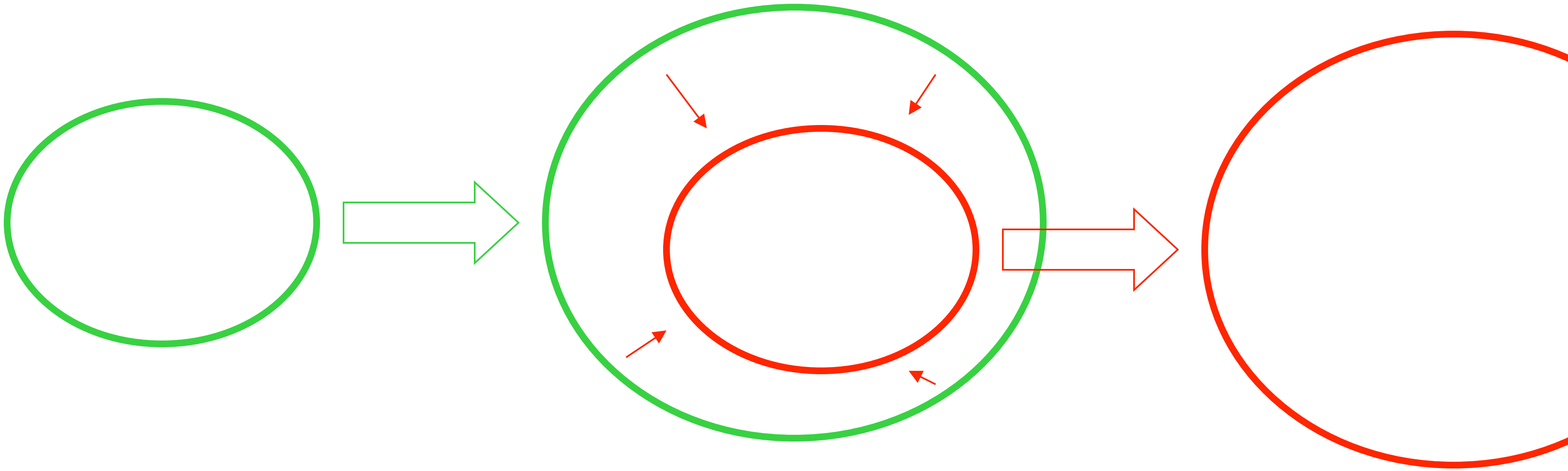
- Arbitrary densities
- Memory = O(#samples)
- Only in "Typical Set"
- Great visualization tool !

- minus: Approximate

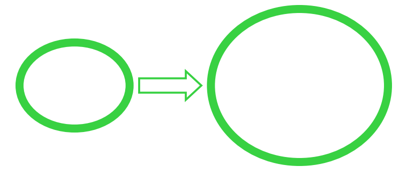
First appeared in 70's, re-discovered by Kitagawa, Isard & Blake in computer vision, Monte Carlo Localization in robotics

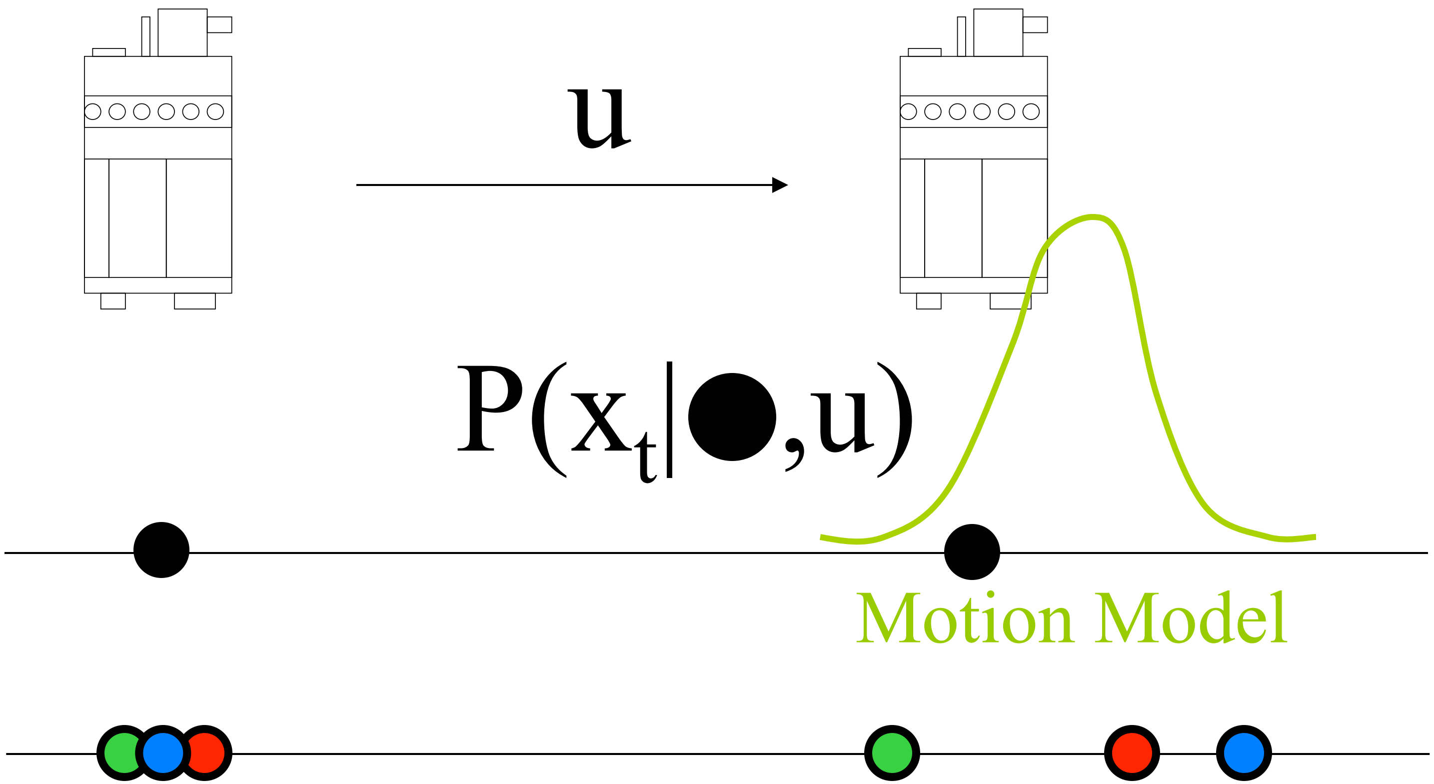
Bayesian Filtering

- Two phases: 1. Prediction Phase
  2. Measurement Phase

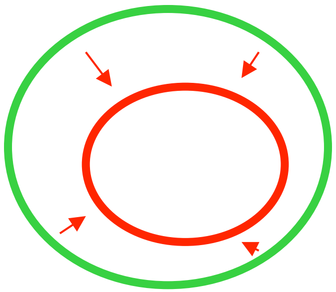


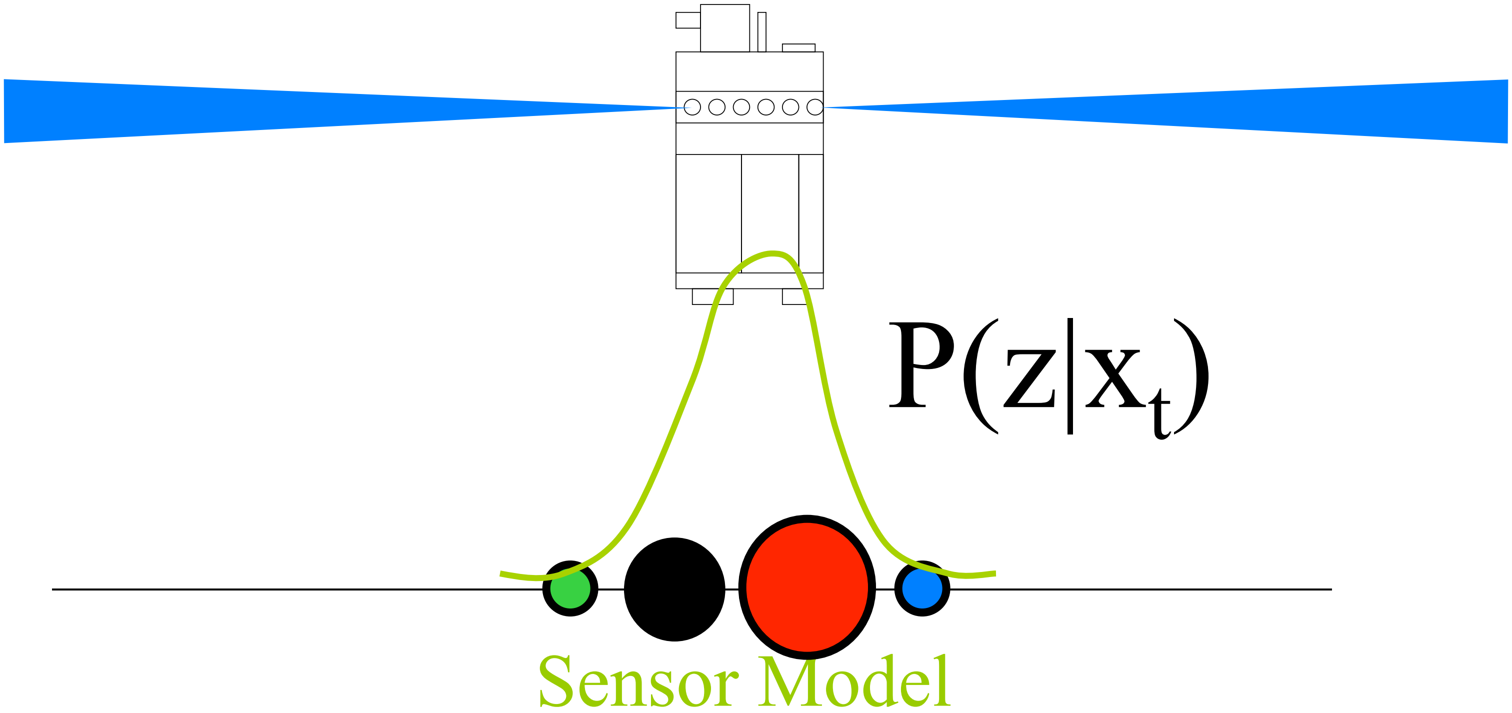
1. Prediction Phase



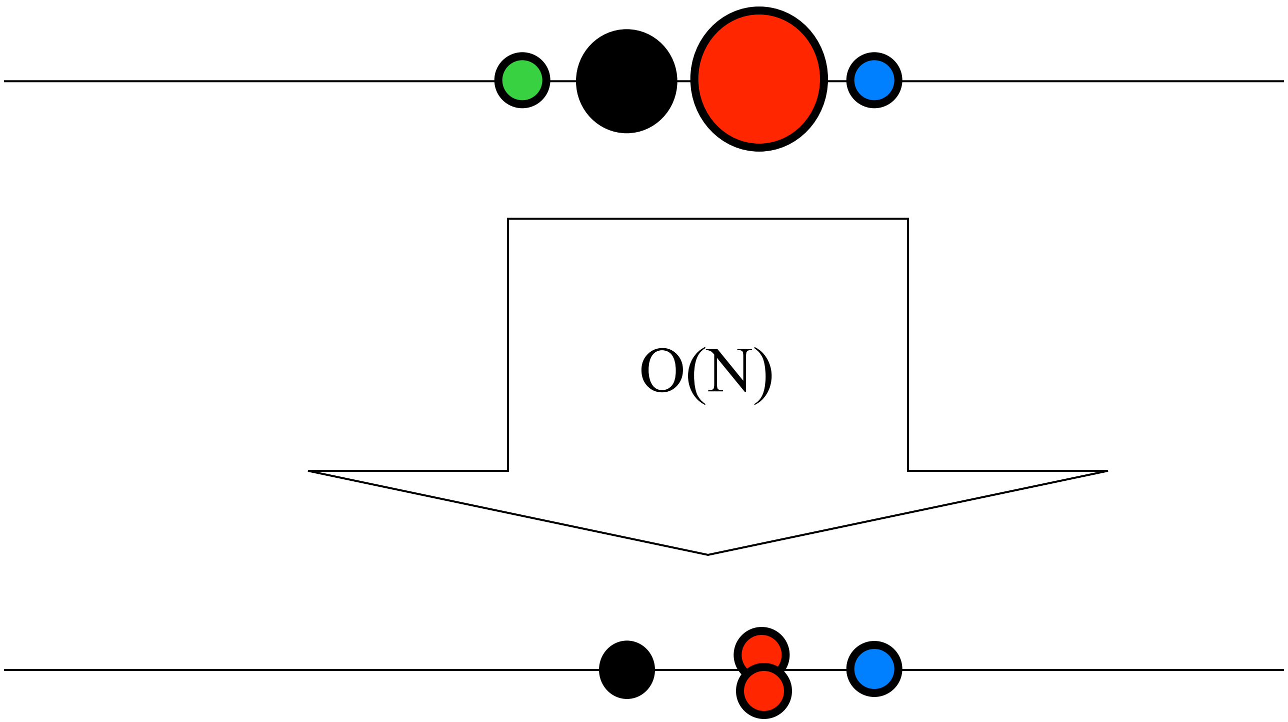


2. Measurement Phase



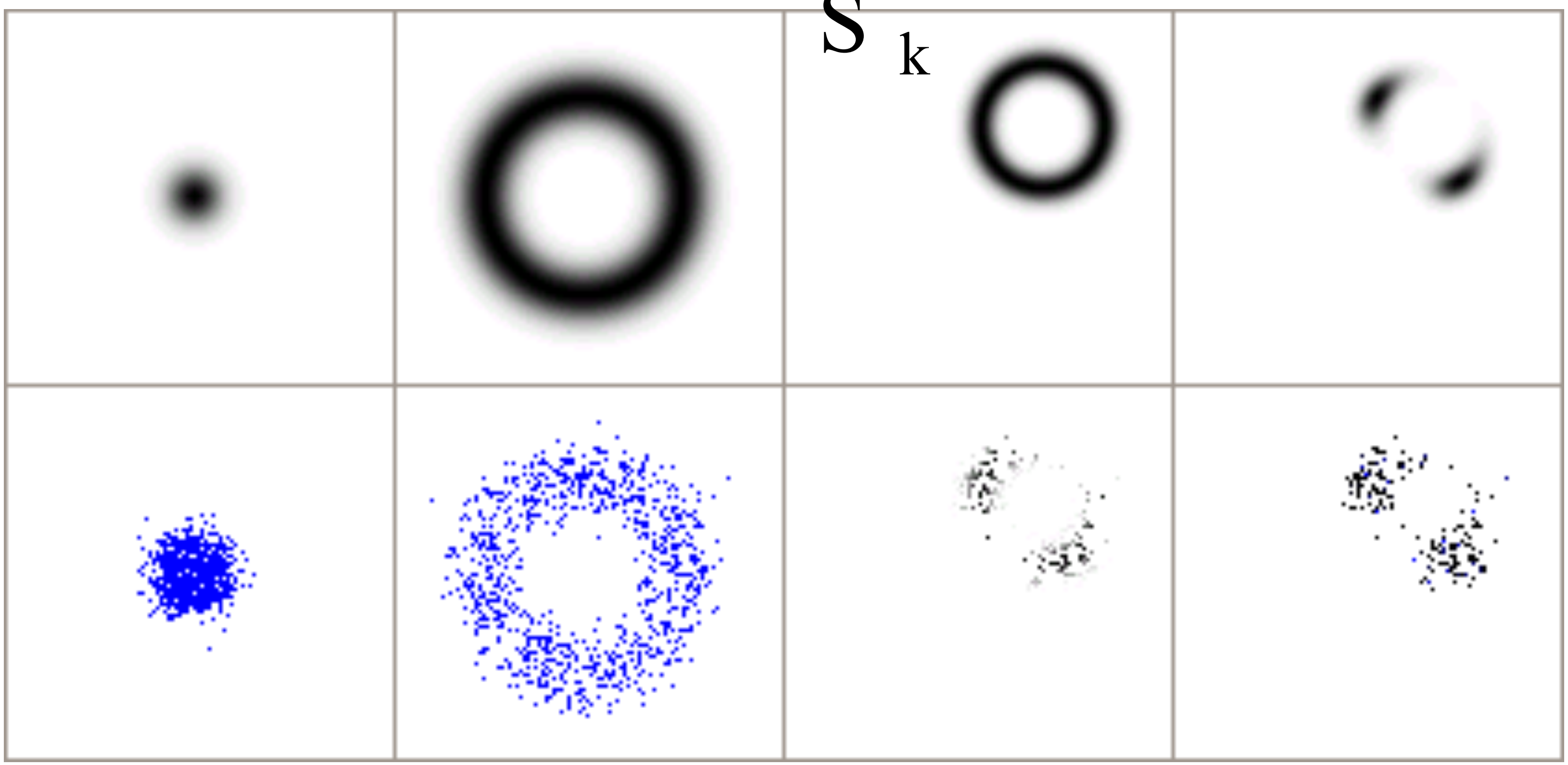


3. Resampling Step



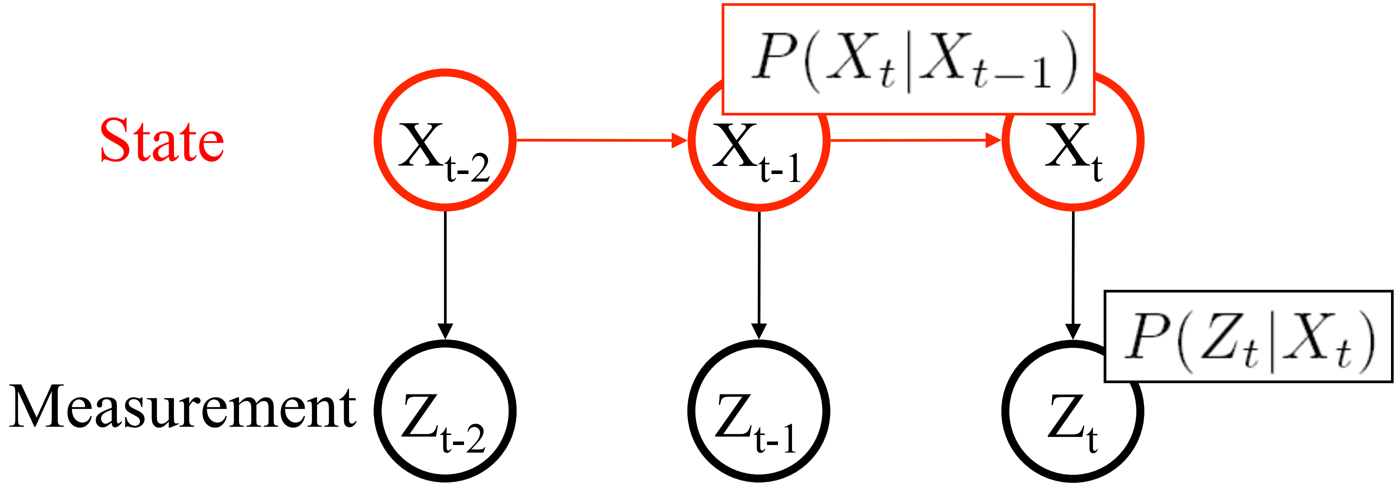
Monte Carlo Localization

S_{k-1} S'_k weighted S'_k S_k



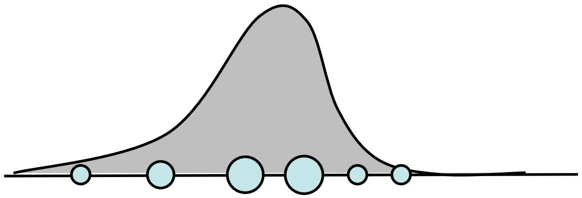
Predict Weight Resample

Particle Filter Tracking



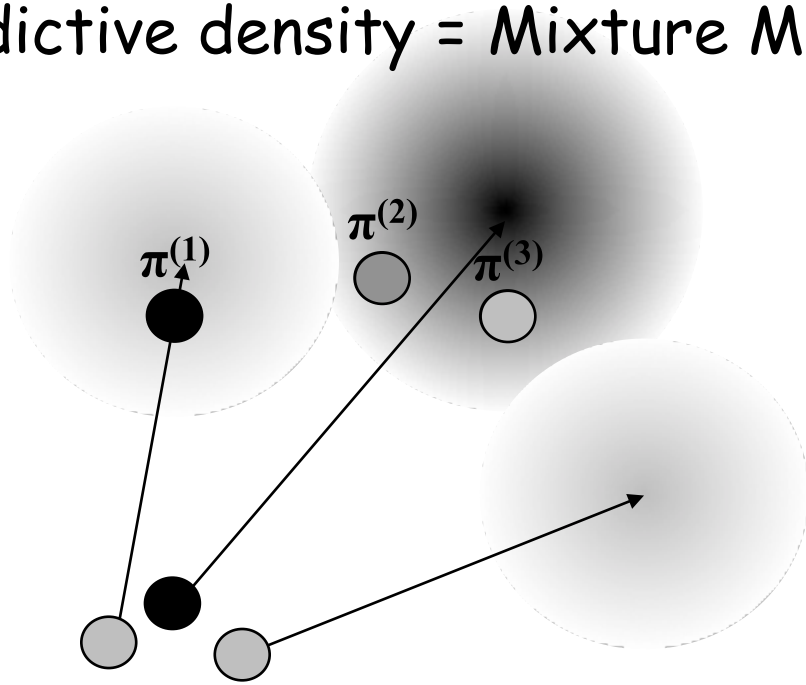
Monte Carlo Approximation of Posterior:

$$P(X_{t-1}|Z^{t-1}) \Longleftrightarrow \{X_{t-1}^{(i)}, \pi_{t-1}^{(i)}\}_{i=1}^{N}$$



Two-step View of the Particle Filter

Empirical predictive density = Mixture Model



$$\pi_t^{(s)} = P(Z_t | X_t^{(s)})$$

Bayes Filter and Particle Filter

Motion Model

Recursive Bayes Filter Equation:

$$P(X_t|Z^t) = kP(Z_t|X_t)\int_{X_{t-1}} P(X_t|X_{t-1})P(X_{t-1}|Z^{t-1})$$

Predictive Density

Monte Carlo Approximation:

$$P(X_t|Z^t) \approx kP(Z_t|X_t)\sum_r \pi_{t-1}^{(r)} P(X_t|X_{t-1}^{(r)})$$

Conclusions

- Monte Carlo Localization:
  Powerful yet efficient
  Significantly less memory and CPU
  Very simple to implement

Take Home Message

Representing uncertainty using samples is powerful, fast, and simple !