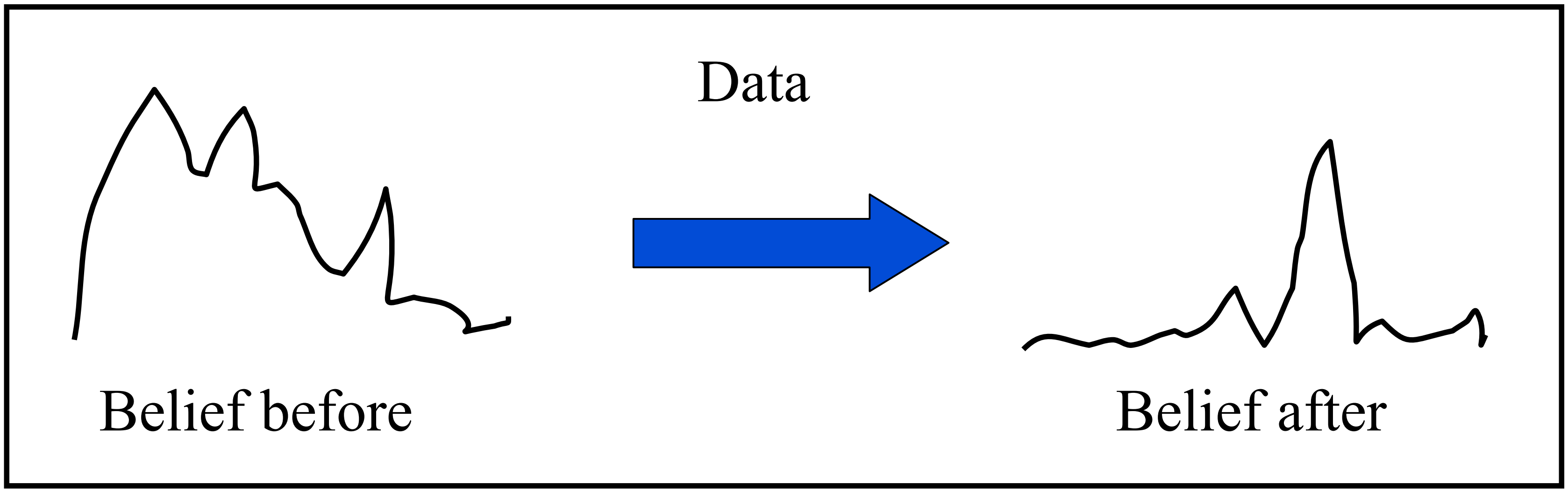
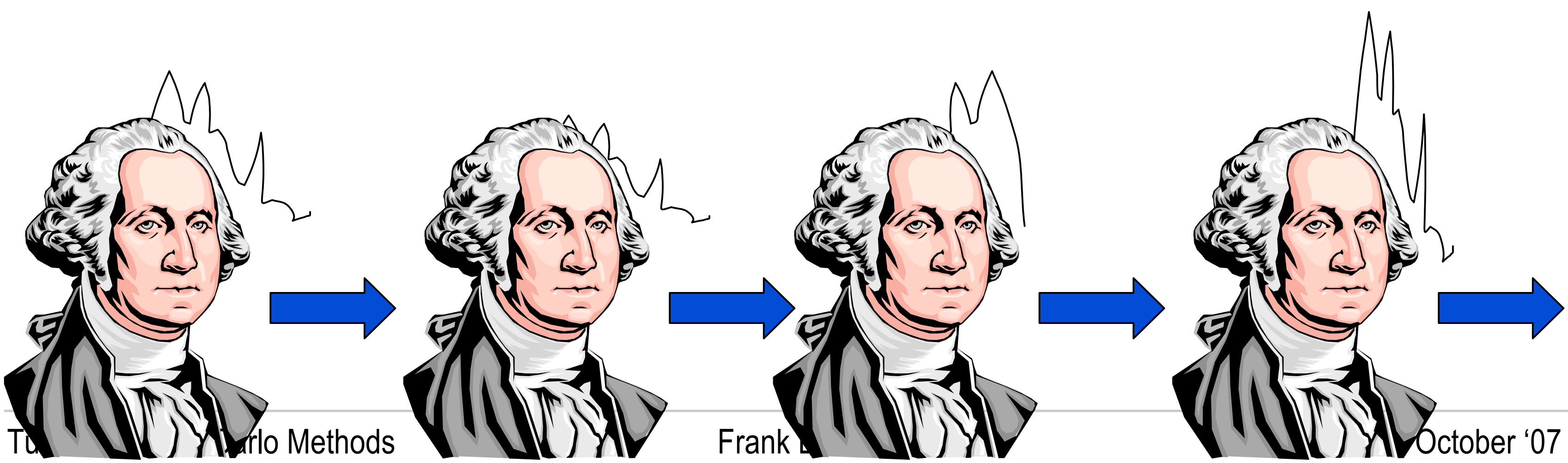
Frank Dellaert, Fall 07

Particle Filtering

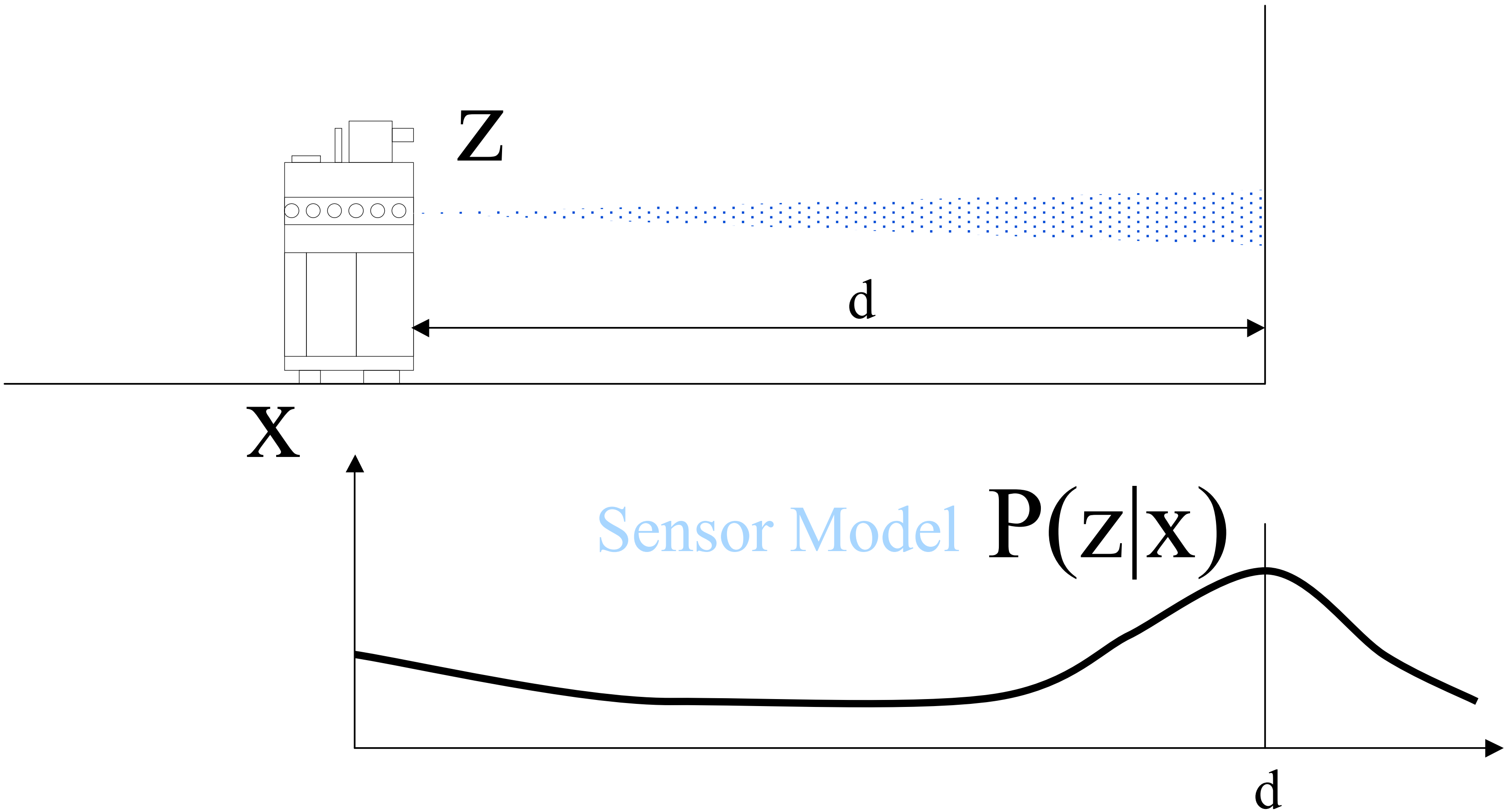
for Tracking and Localization

Bayesian Inference





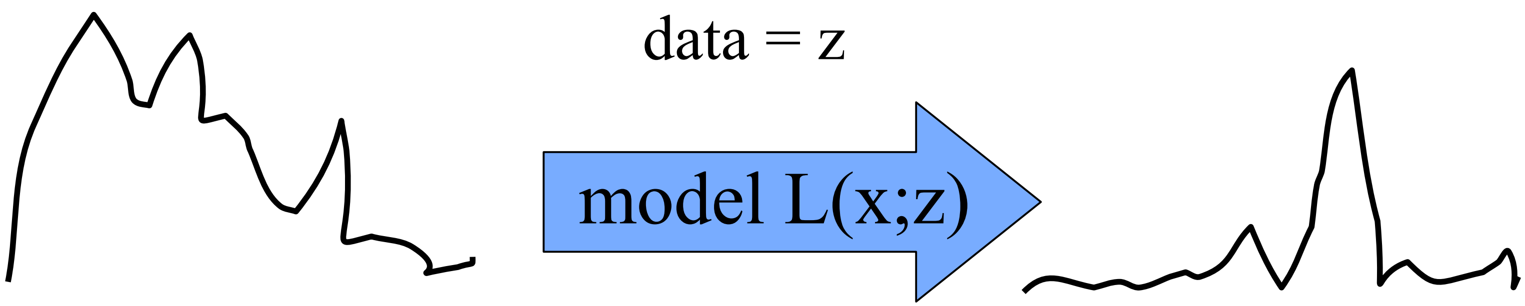
World Knowledge



Most often analytic expression, can be learned

Recap: Bayes Law

$$P(x|z) \sim L(x;z)P(x)$$



Belief before = P(x)

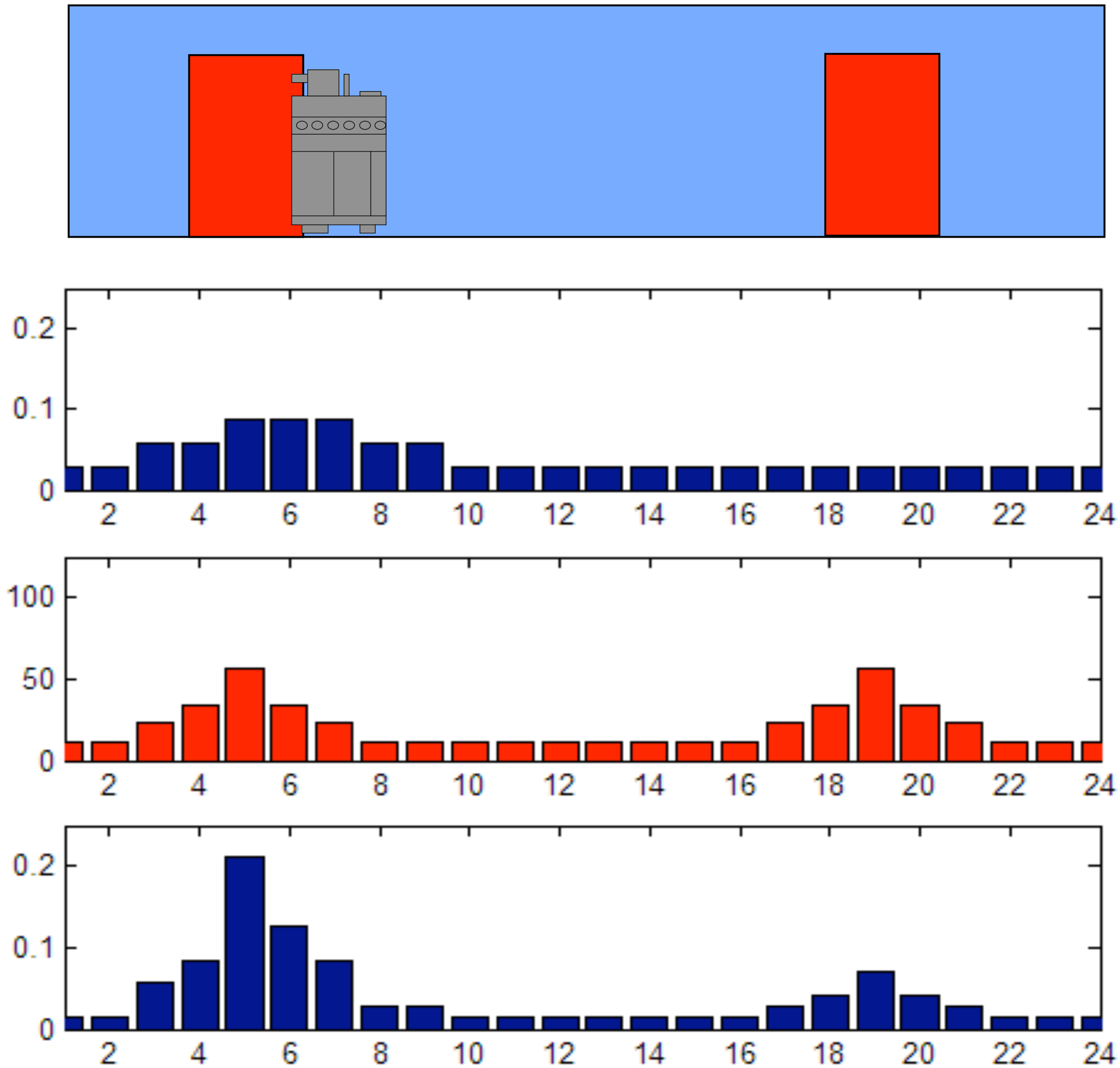
Belief after = P(x|z)

Prior Distribution
of x

Likelihood
of x given Z

Posterior Distribution
of x given Z

Example: 1D Robot Localization



Prior $P(x)$

Likelihood $L(x;z)$

Posterior $P(x|z)$

Problem: Large Open Spaces

- Walls and obstacles out of range
- Sonar and laser have problems
- Horizontally mounted sensors have problems

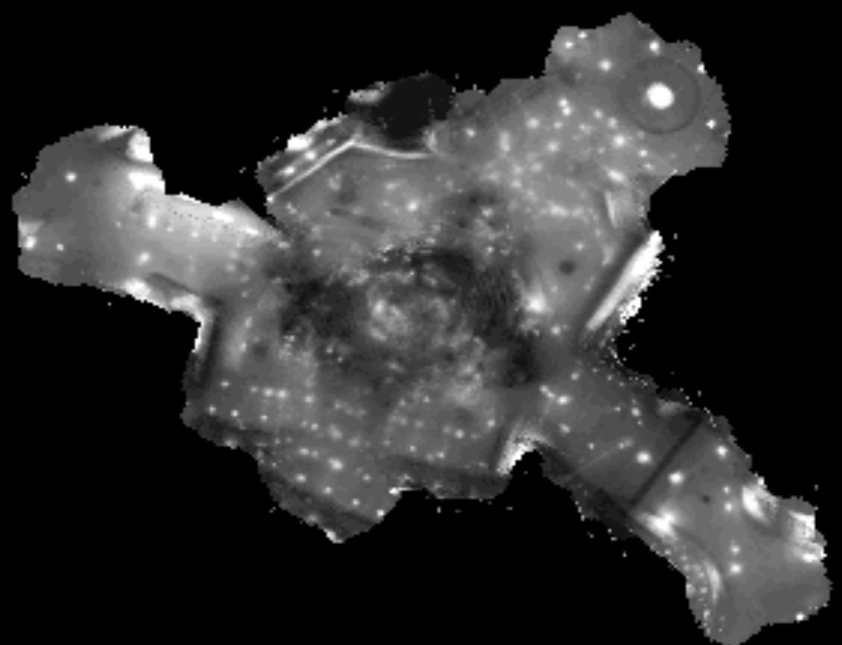
Problem: Large Crowds

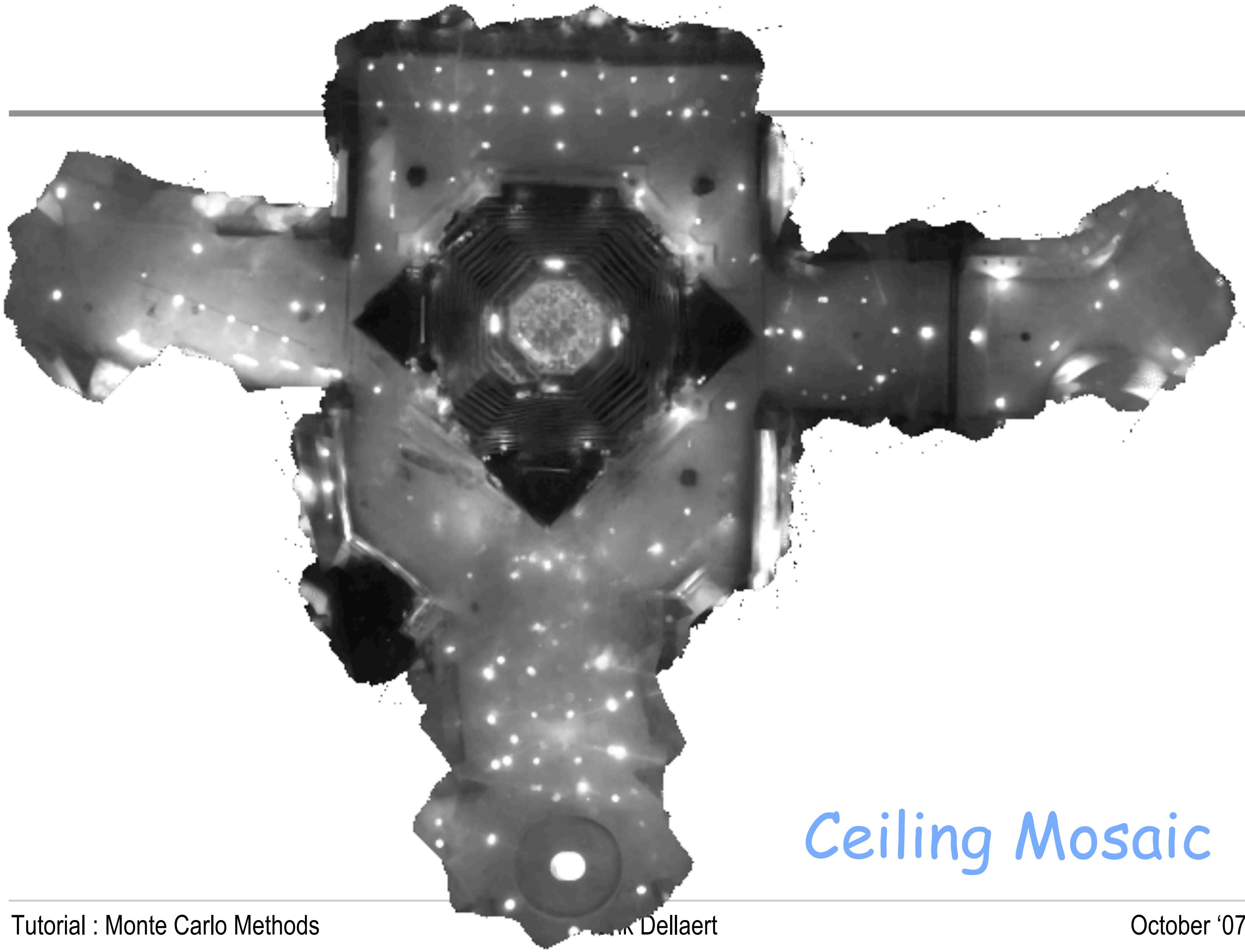
- One solution: Robust filtering

Solution: Ceiling Camera

- Upward looking camera
- Model of the world = Ceiling Mosaic

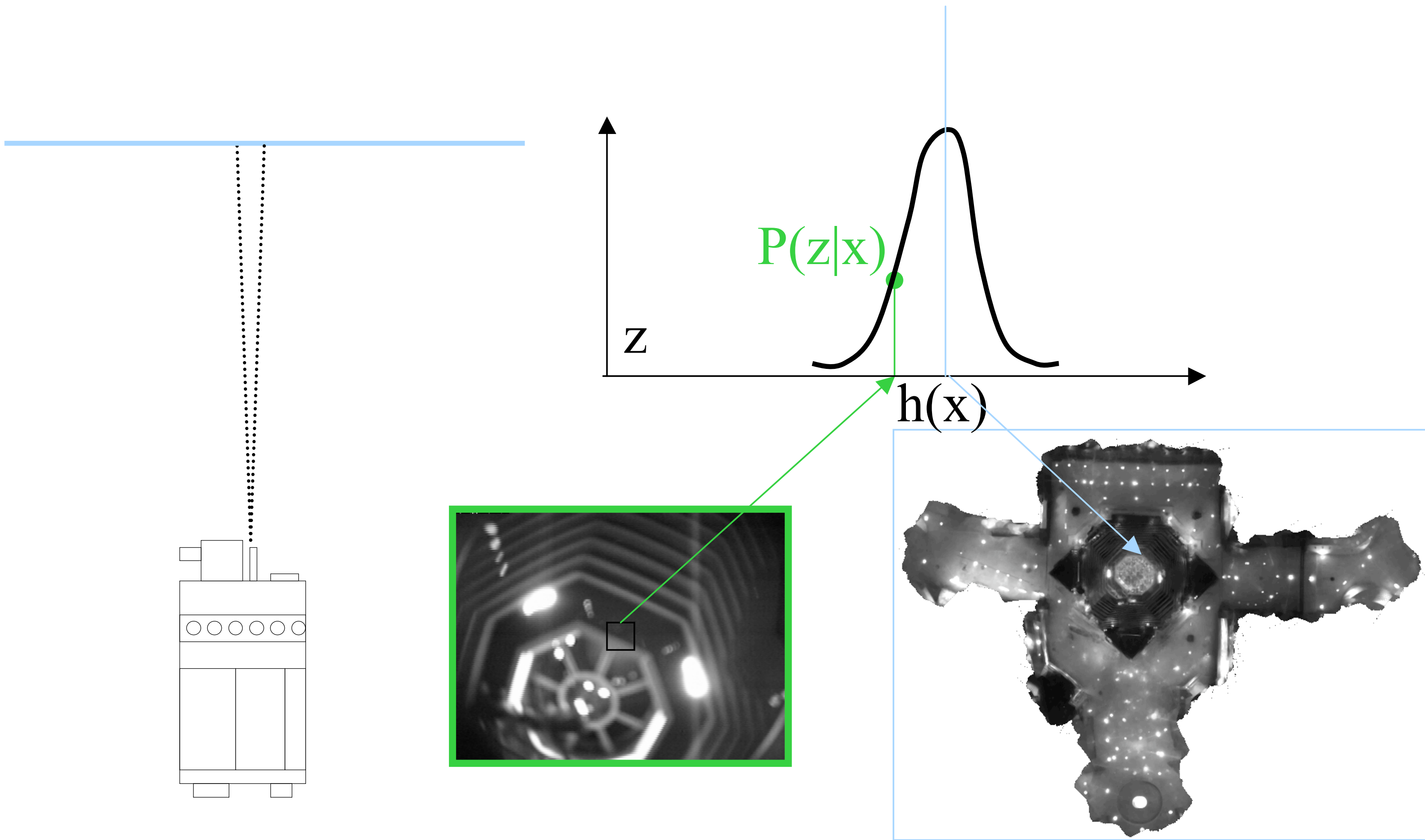
Global Alignment (other talk)





Ceiling Mosaic

Vision based Sensor



Under Light

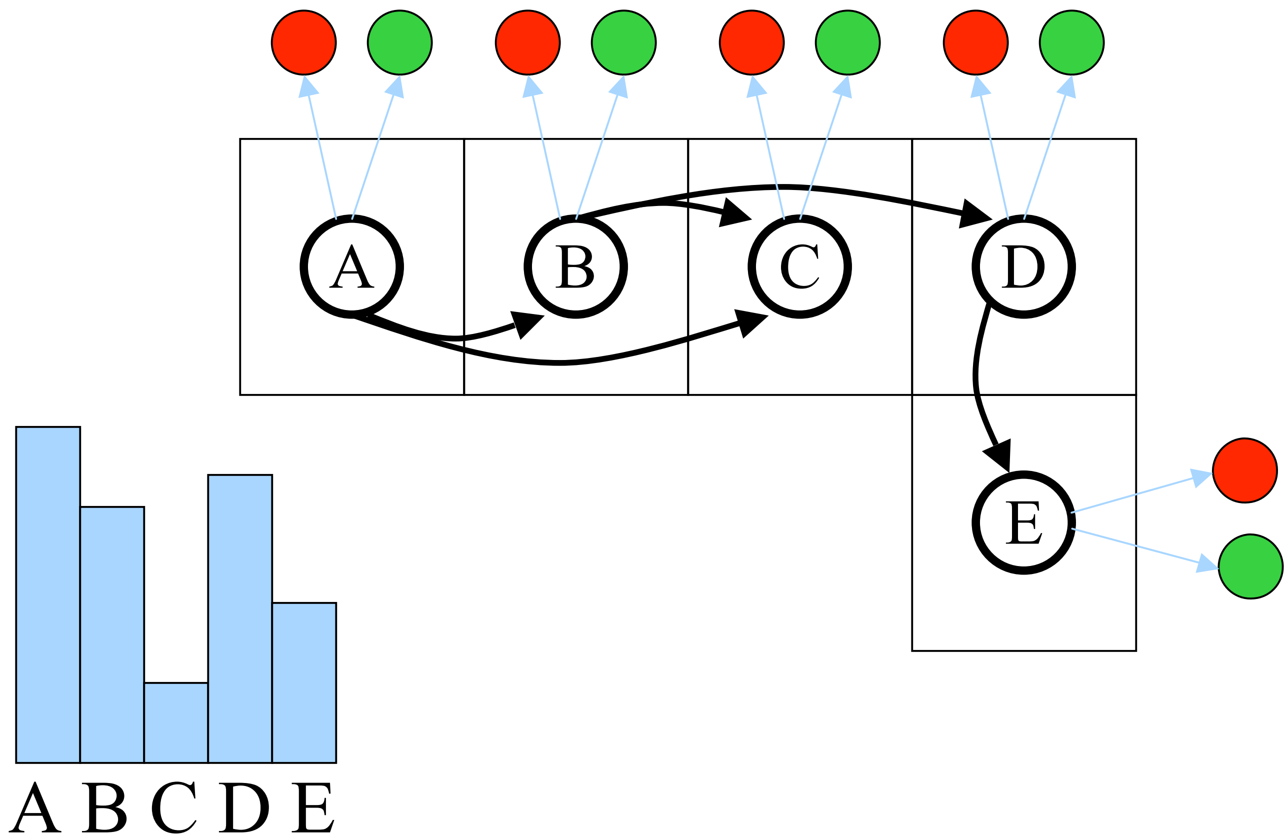
Next to Light

Elsewhere

Various Density Representations

- Gaussian centered around mean x,y
- Mixture of Gaussians
- Finite element i.e. histogram
- Does not scale to large state spaces encountered in computer vision & robotics

Hidden Markov Models

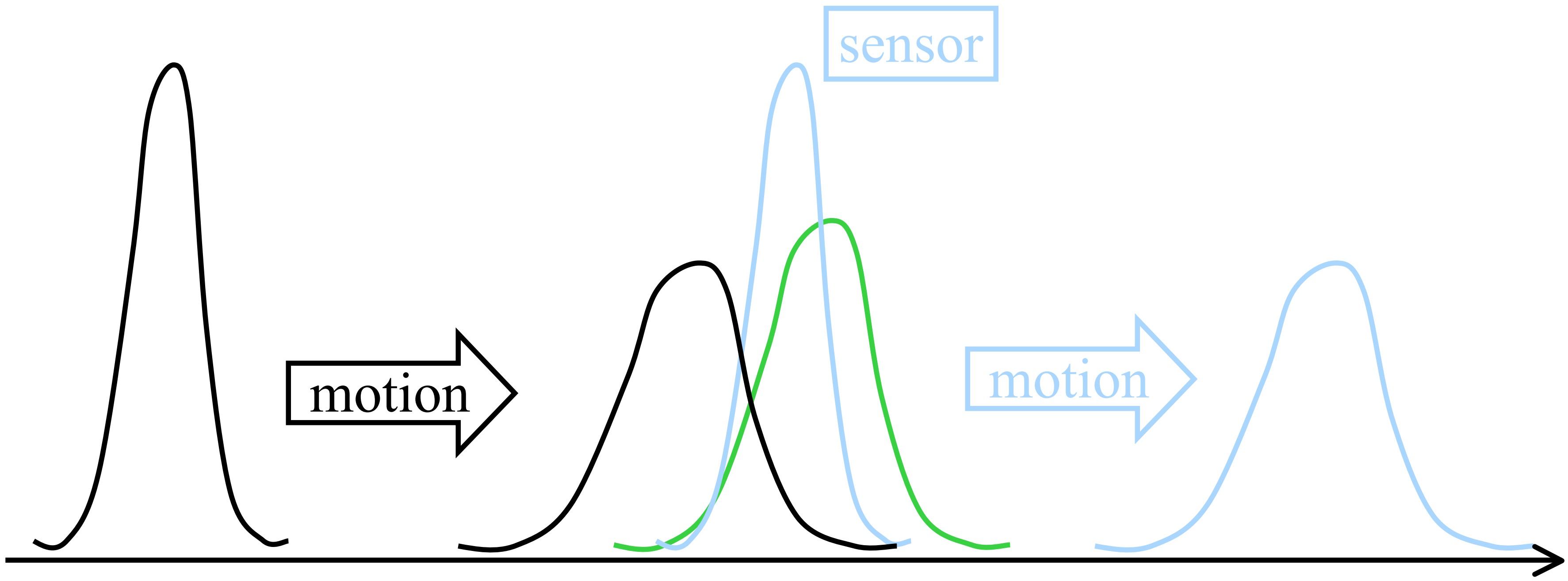


Kalman Filter = Very Easy

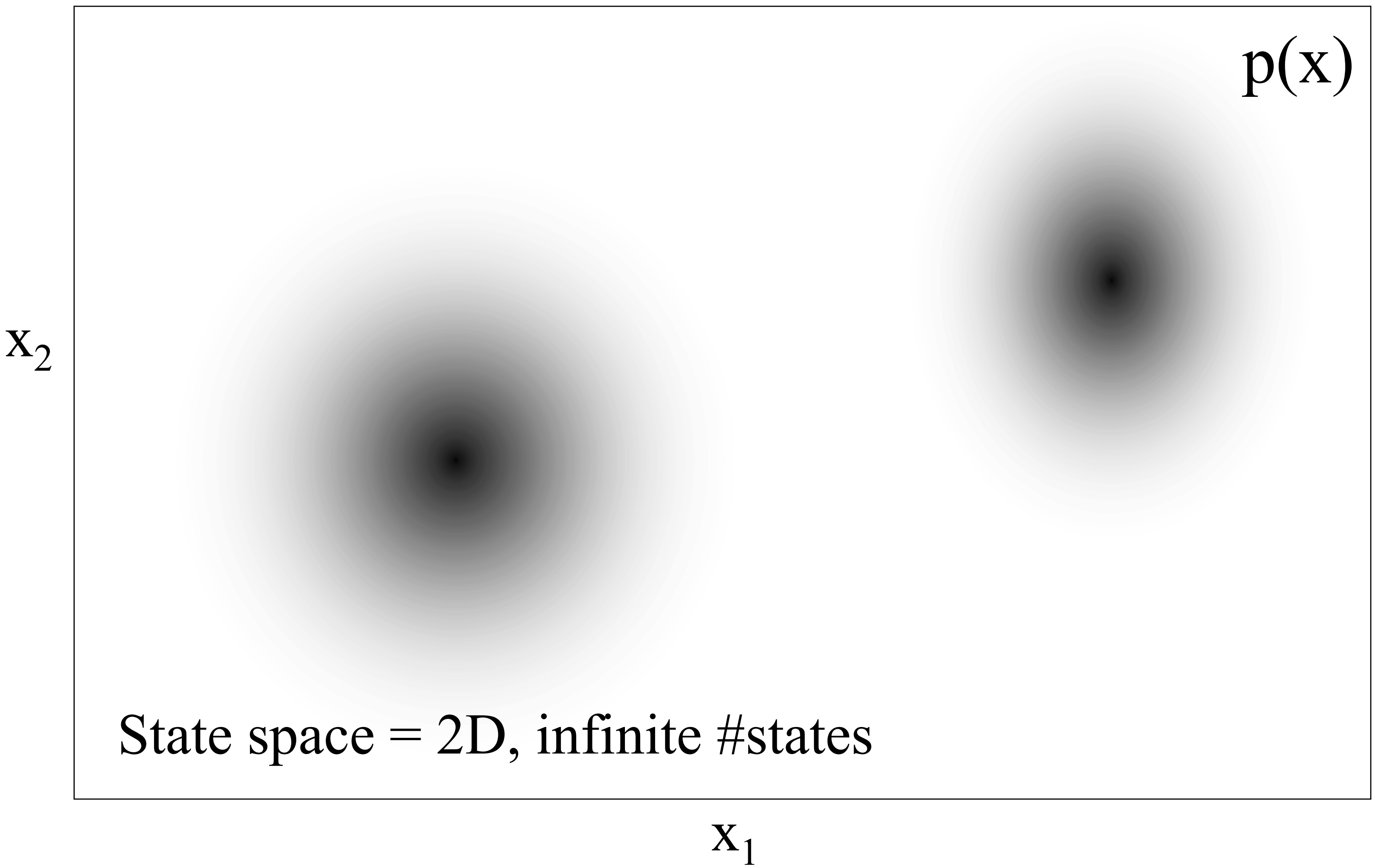
- Think adding quadratics
- Then Minimize
- Dynamics = Enlarge Quadratic

Kalman Filter

- Very powerful
- Gaussian, unimodal

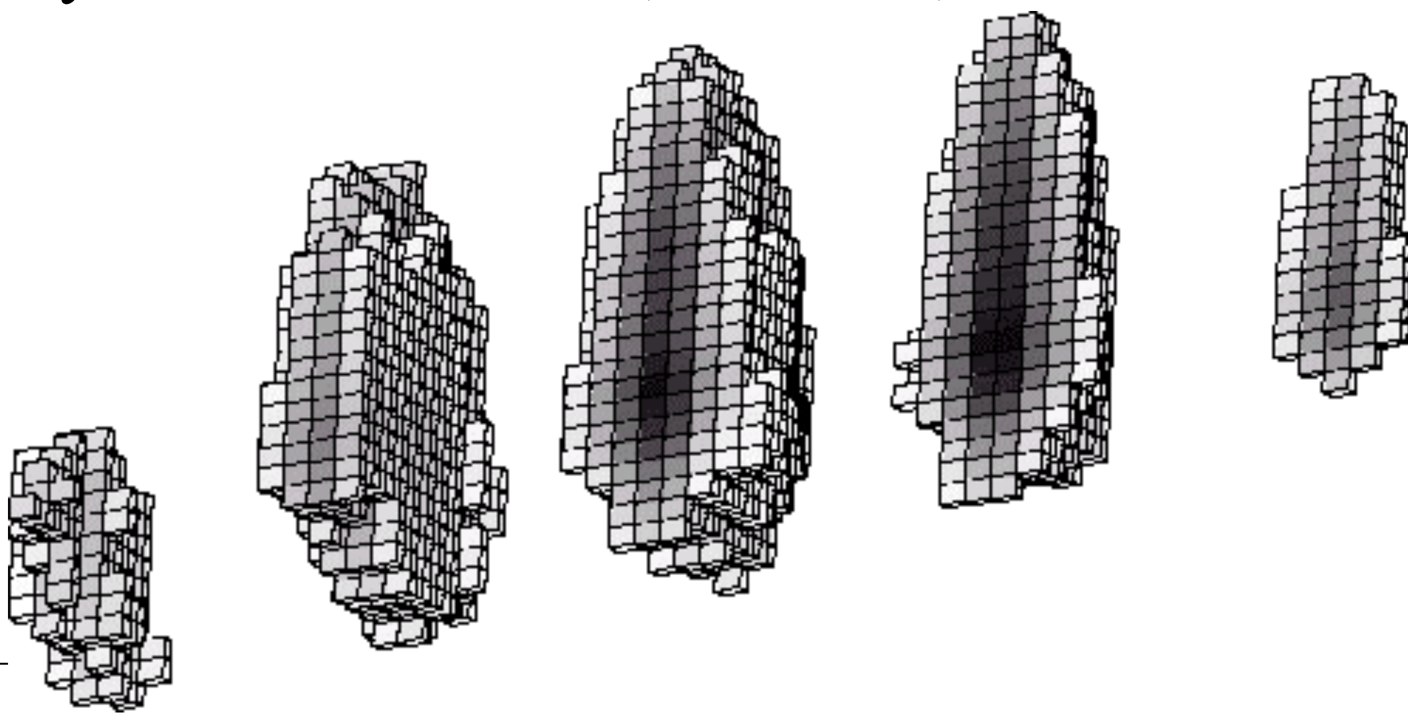


Example: 2D Robot Location



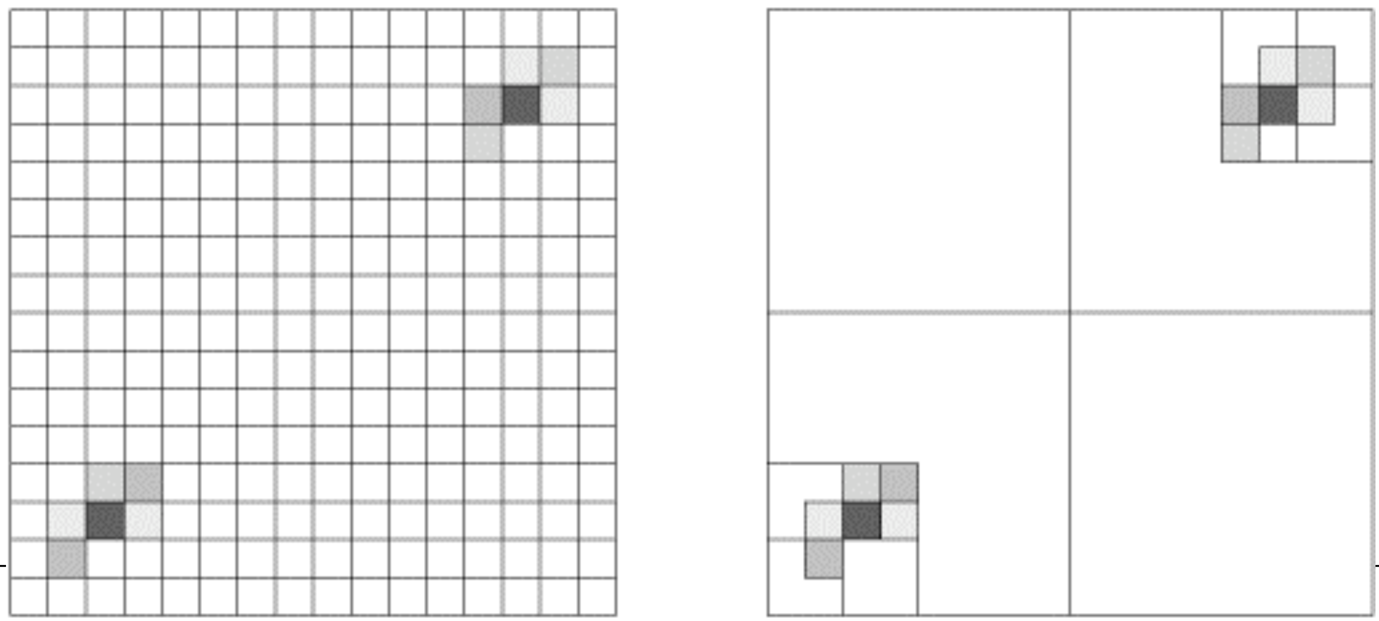
Markov Localization

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

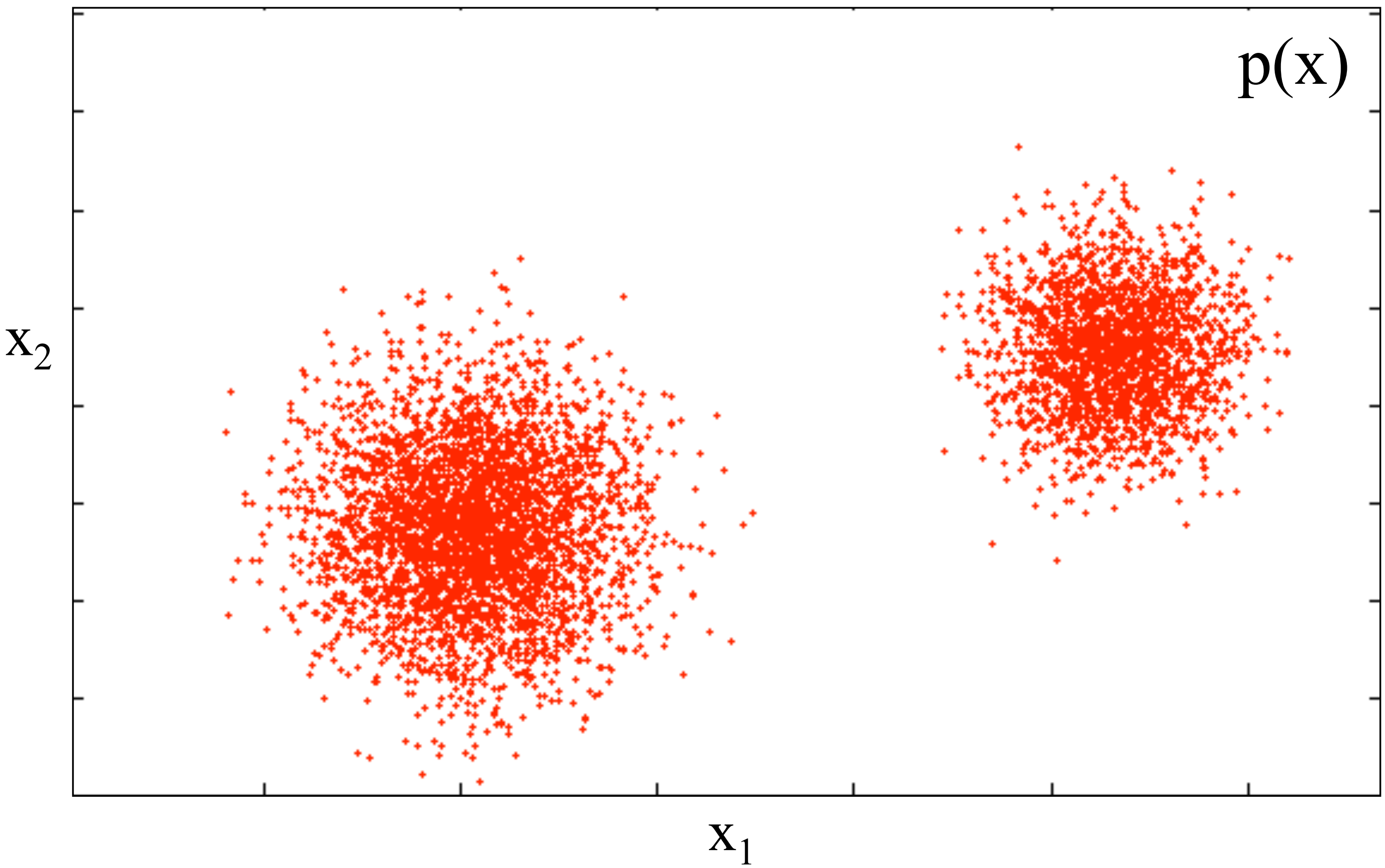


Dynamic Markov Localization

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



Sampling as Representation



Sampling Advantages

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

- minus: Approximate

Mean and Variance of a Sample

Mean

$$\mu = \int_x x P(x) dx$$

$$\mu \approx \frac{1}{R} \sum_{r=1}^{R} x^{(r)}$$

Variance (1D)

$$\sigma^2 = \int_x (x - \mu)^2 P(x) dx$$

$$\sigma^2 \approx \frac{1}{R} \sum_{r=1}^{R} (x^{(r)} - \hat{\mu})^2$$

Inference = Monte Carlo Estimates

- Estimate expectation of any function f:

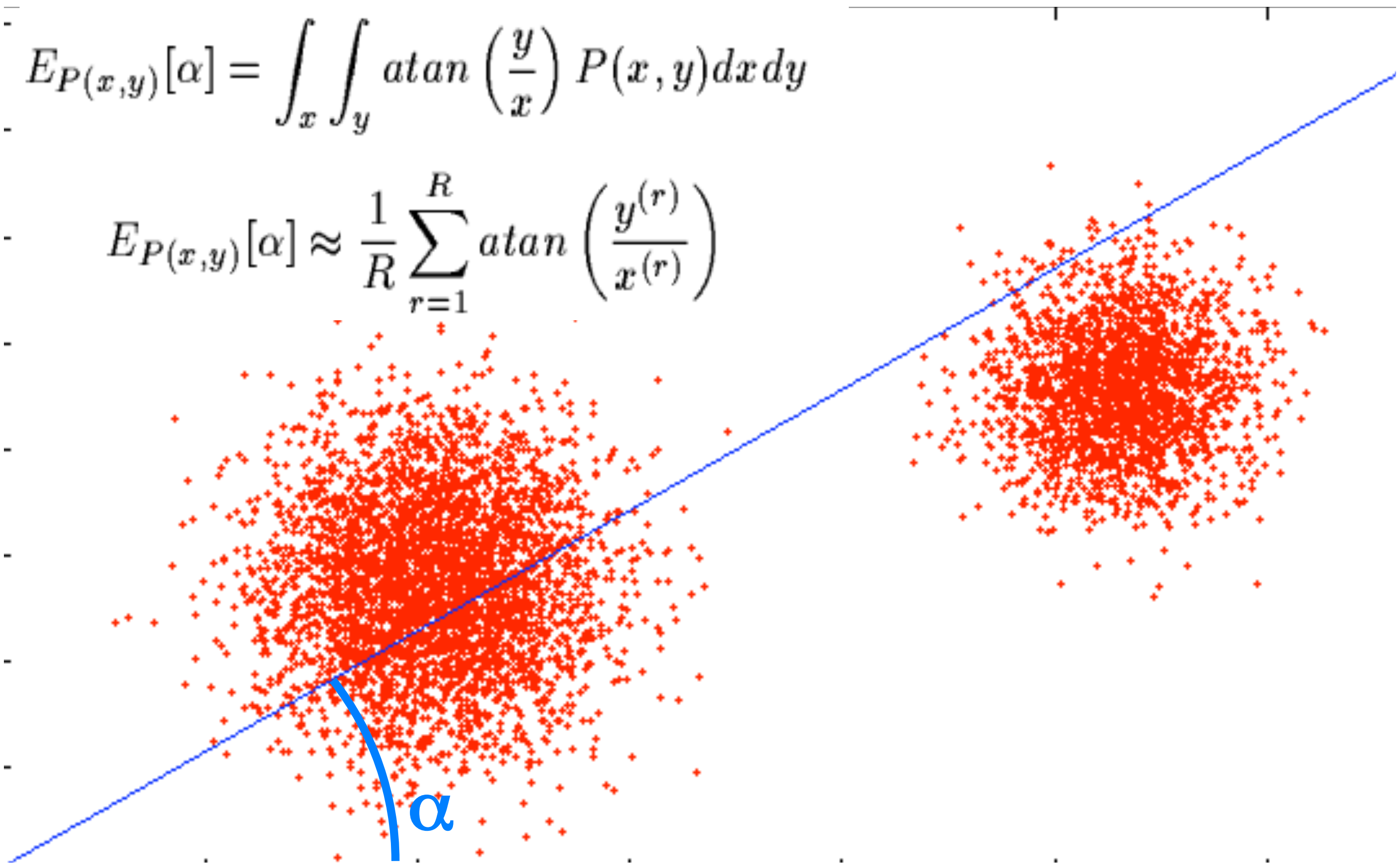
$$E_{P(x)}[f(x)] = \int_x f(x)P(x)d^N x$$

$$E_{P(x)}[f(x)] \approx \frac{1}{R}\sum_{r=1}^{R} f(x^{(r)})$$

Monte Carlo Expected Value

$$E_{P(x,y)}[\alpha] = \int_x \int_y atan\left(\frac{y}{x}\right) P(x,y)\,dx\,dy$$

$$E_{P(x,y)}[\alpha] \approx \frac{1}{R}\sum_{r=1}^{R} atan\left(\frac{y^{(r)}}{x^{(r)}}\right)$$



Expected angle = 30°

How to Sample ?

- Target Density $\pi(x)$
- Assumption: we can evaluate $\pi(x)$ up to an arbitrary multiplicative constant

- Why can't we just sample from $\pi(x)$??

How to Sample ?

- Numerical Recipes in C, Chapter 7
- Transformation method: Gaussians etc…
- Rejection sampling
- Importance sampling

Rejection Sampling

- Target Density $\pi(x)$
- Proposal Density $q(x)$

- π and q need only be known up to a factor

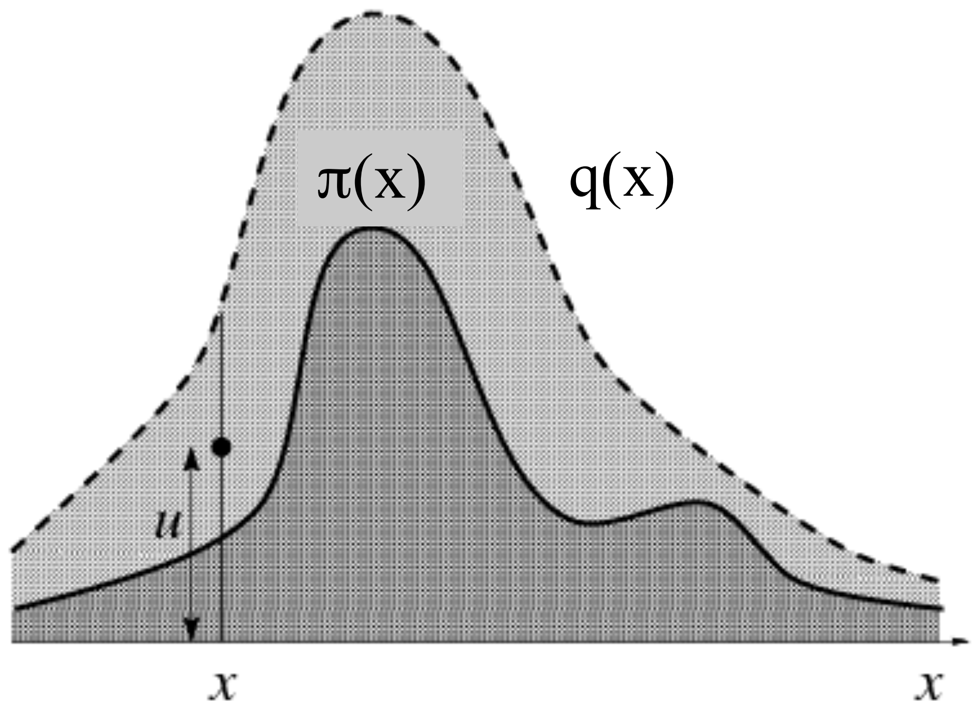
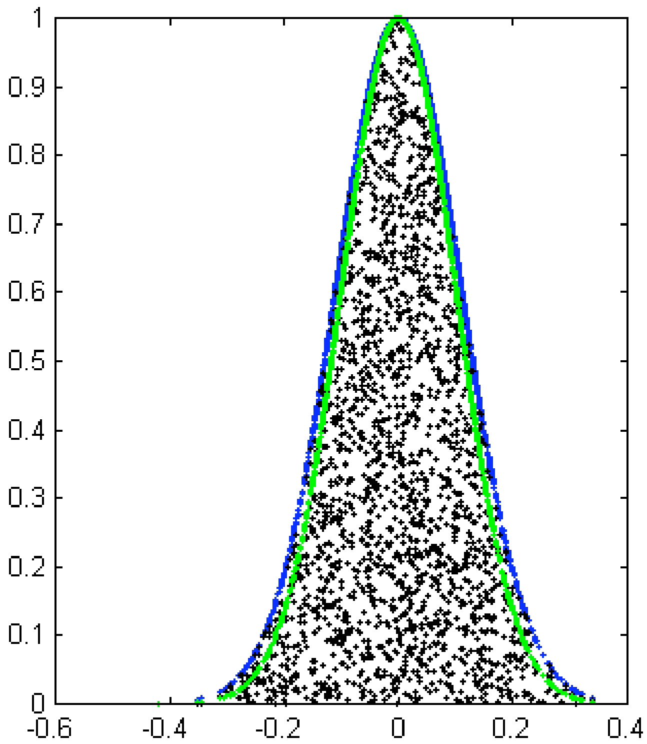
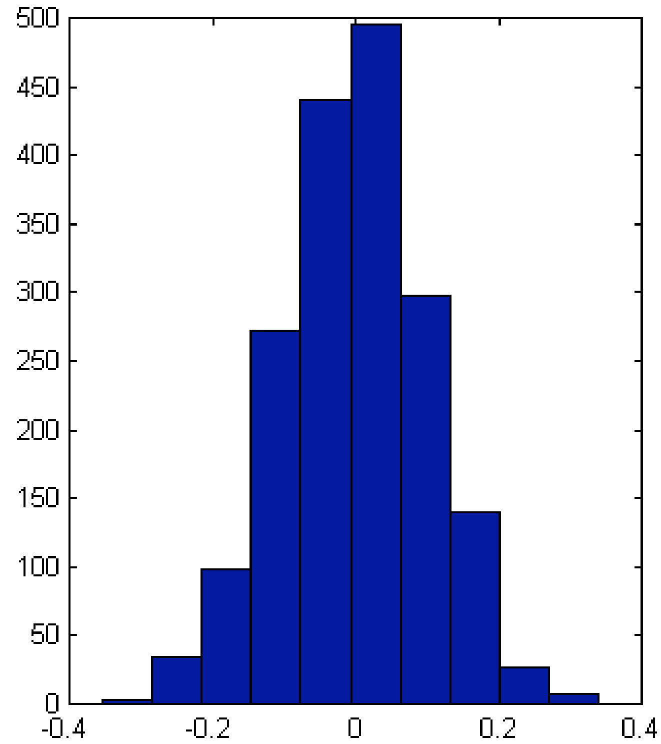


Image by MacKay

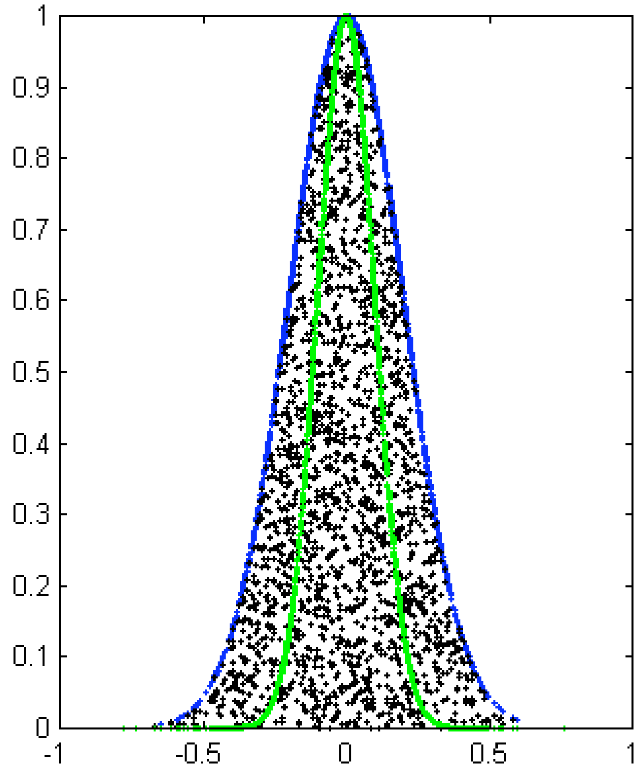
The Good...

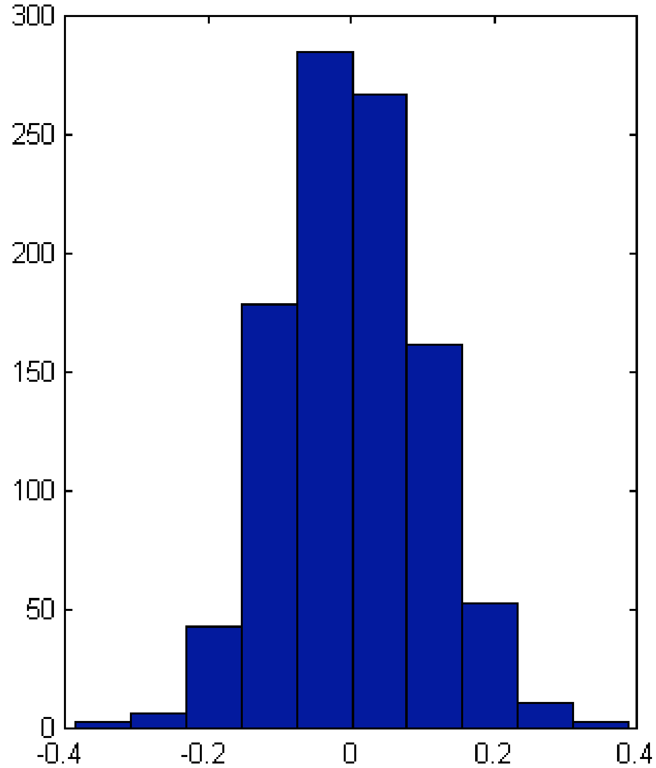




9% Rejection Rate

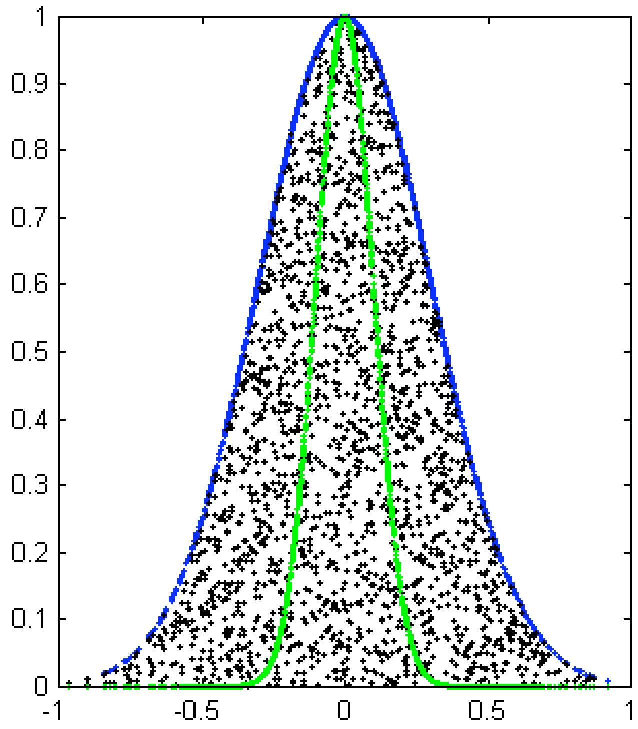
…the Bad…

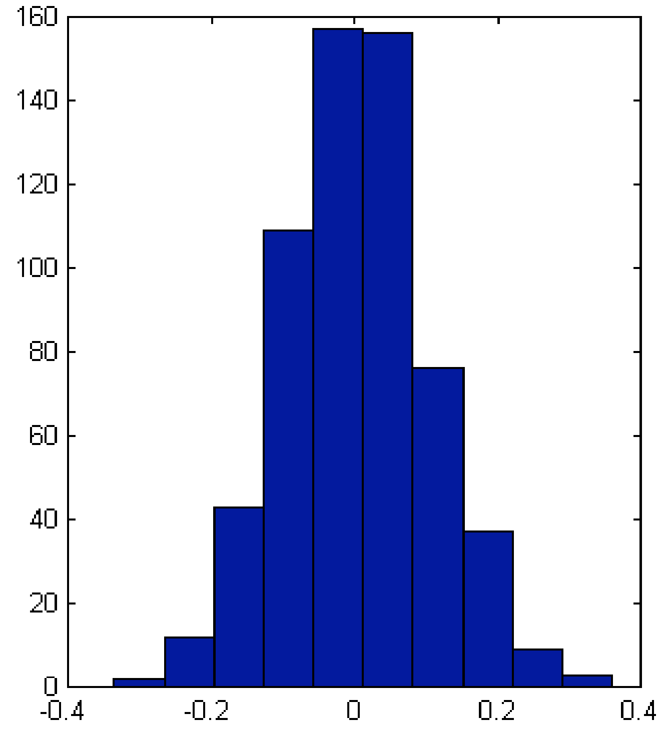




50% Rejection Rate

…and the Ugly.

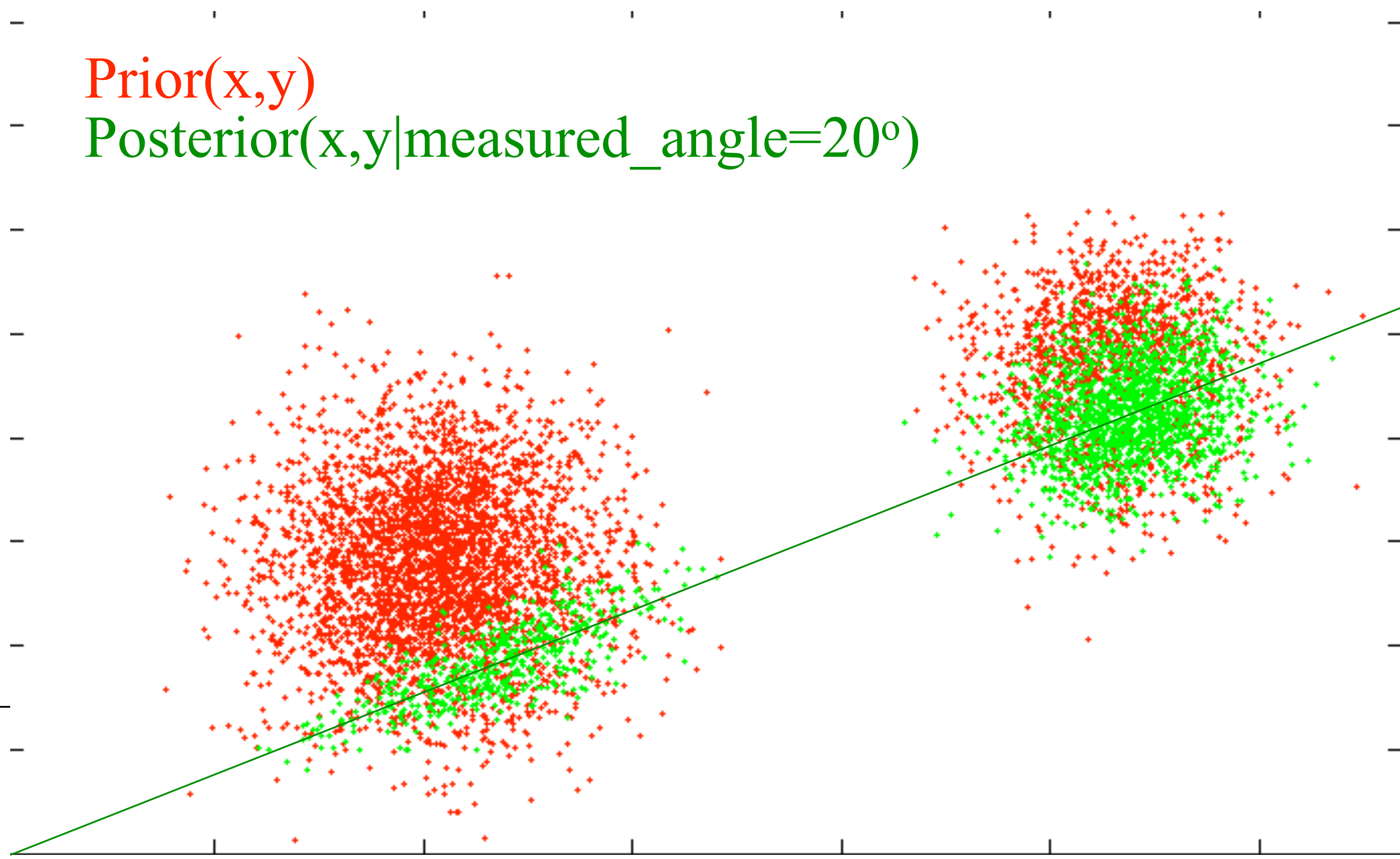




70% Rejection Rate

Inference by Rejection Sampling

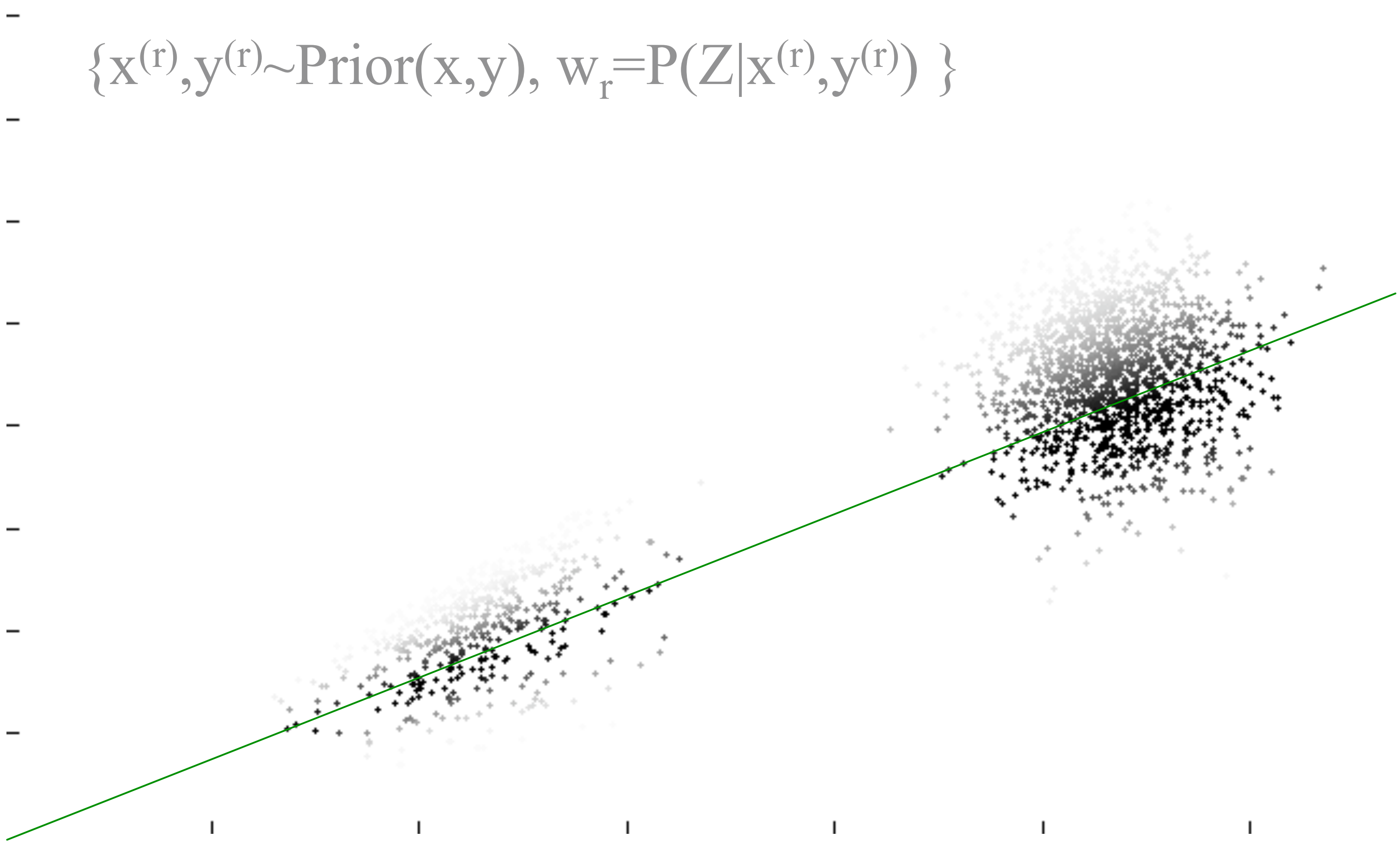
- P(measured_angle|x,y) = N(predicted_angle,3 degrees)



Importance Sampling

- Good Proposal Density would be: prior !
- Problem:
  - No guaranteed c s.t. c $P(x) >= P(x|z)$ for all x
- Idea:
  - sample from $P(x)$
  - give each sample $x^{(r)}$ a importance weight equal to $P(Z|x^{(r)})$

Example Importance Sampling



Importance Sampling

- Sample $x^{(r)}$ from $q(x)$
  - $w_r = \pi(x^{(r)})/q(x^{(r)})$

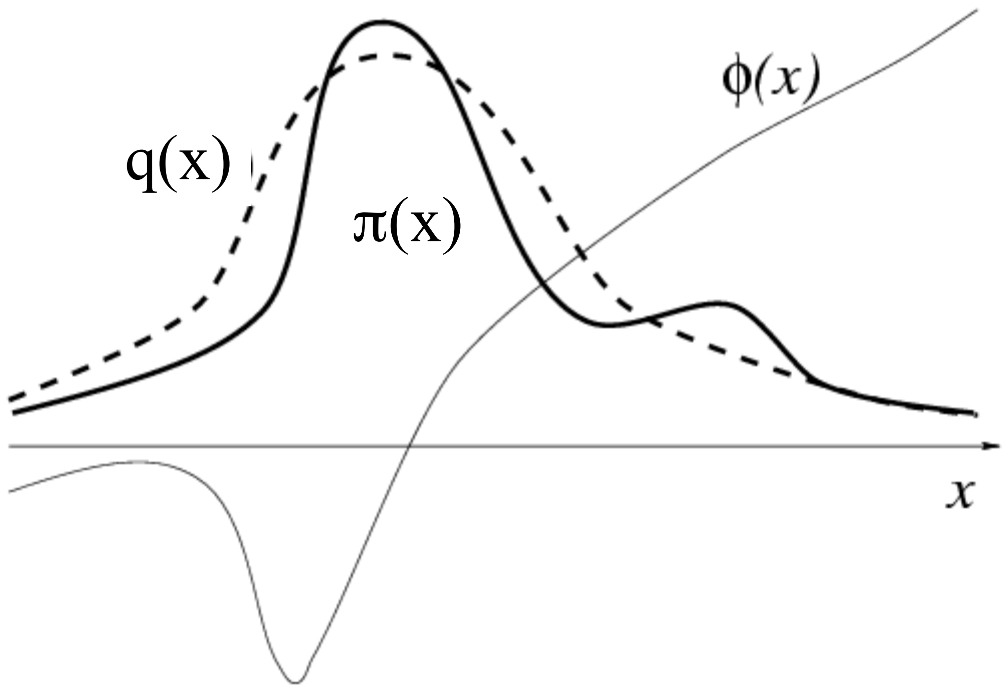
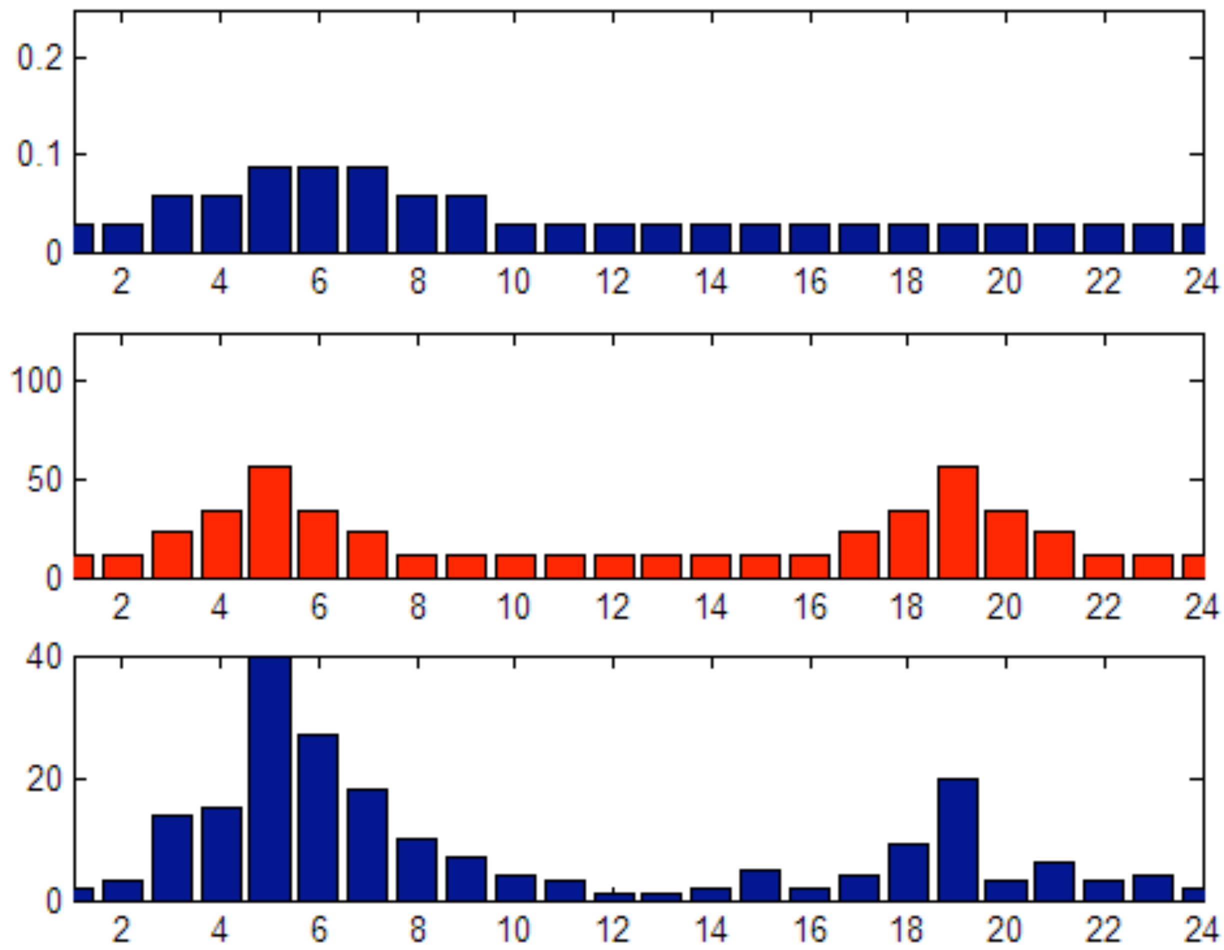


Image by MacKay

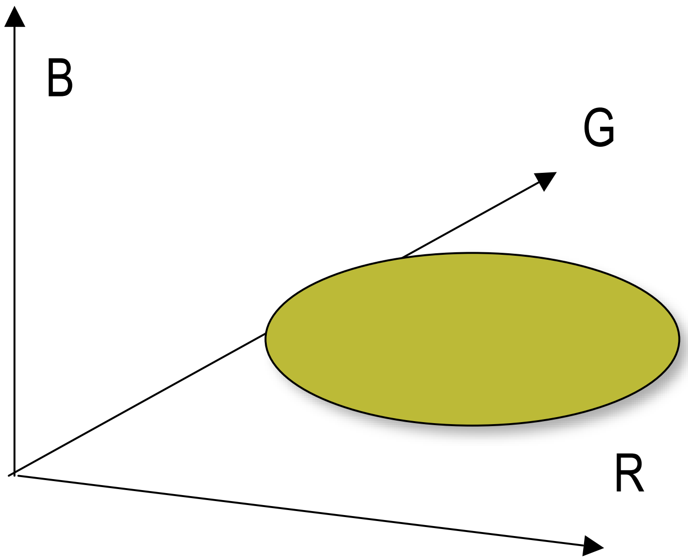
1D Importance Sampling



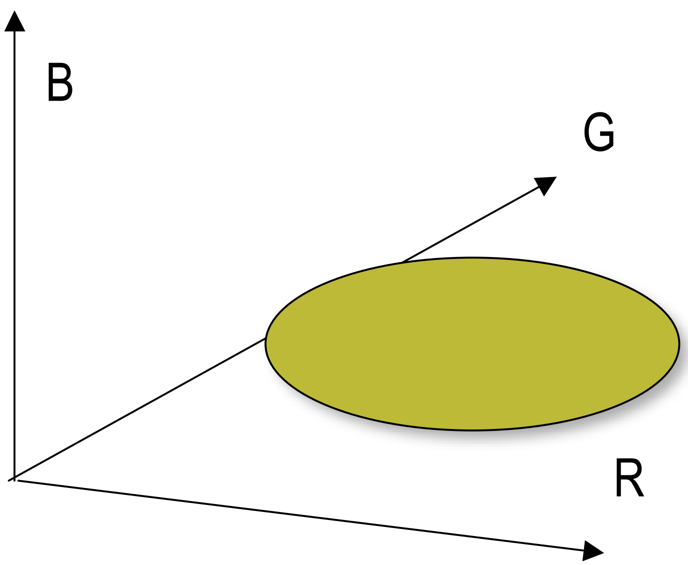
Example 1

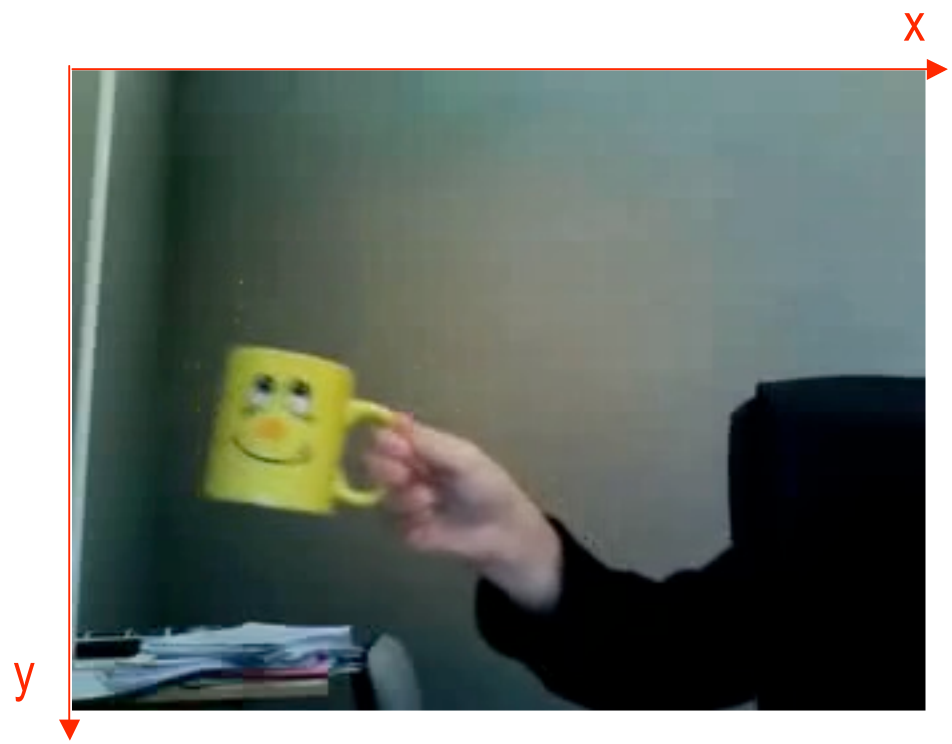
- Learn Color Model
- Implement Rejection Sampling
- Implement Importance Sampling
- Add a spatial prior

A Simple Color Model



Likelihood





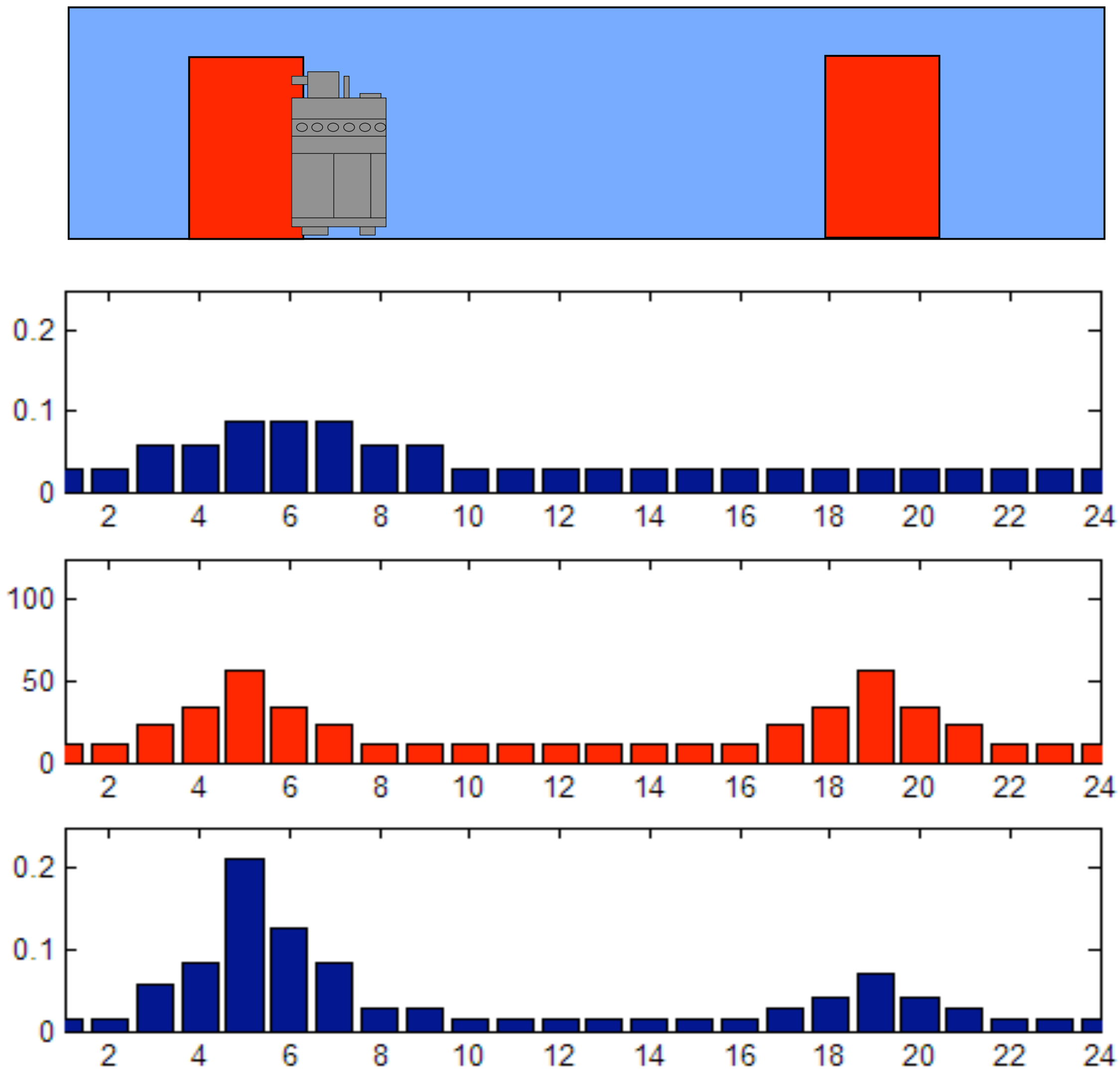
$$P(x, y \mid color) \propto P(color \mid x, y) P(x, y)$$

$$P(color \mid x, y) = \prod_{c=r,g,b} N(\mu_c, \sigma_c{}^\wedge 2)$$

References

- **Isard & Blake 98**, Condensation -- conditional density propagation for visual tracking
- **Dellaert, Fox, Burgard & Thrun 99**, Monte Carlo Methods Localization for Mobile Robots
- **Khan, Balch & Dellaert 04** A Rao-Blackwellized Particle Filter for EigenTracking

Example: 1D Robot Localization

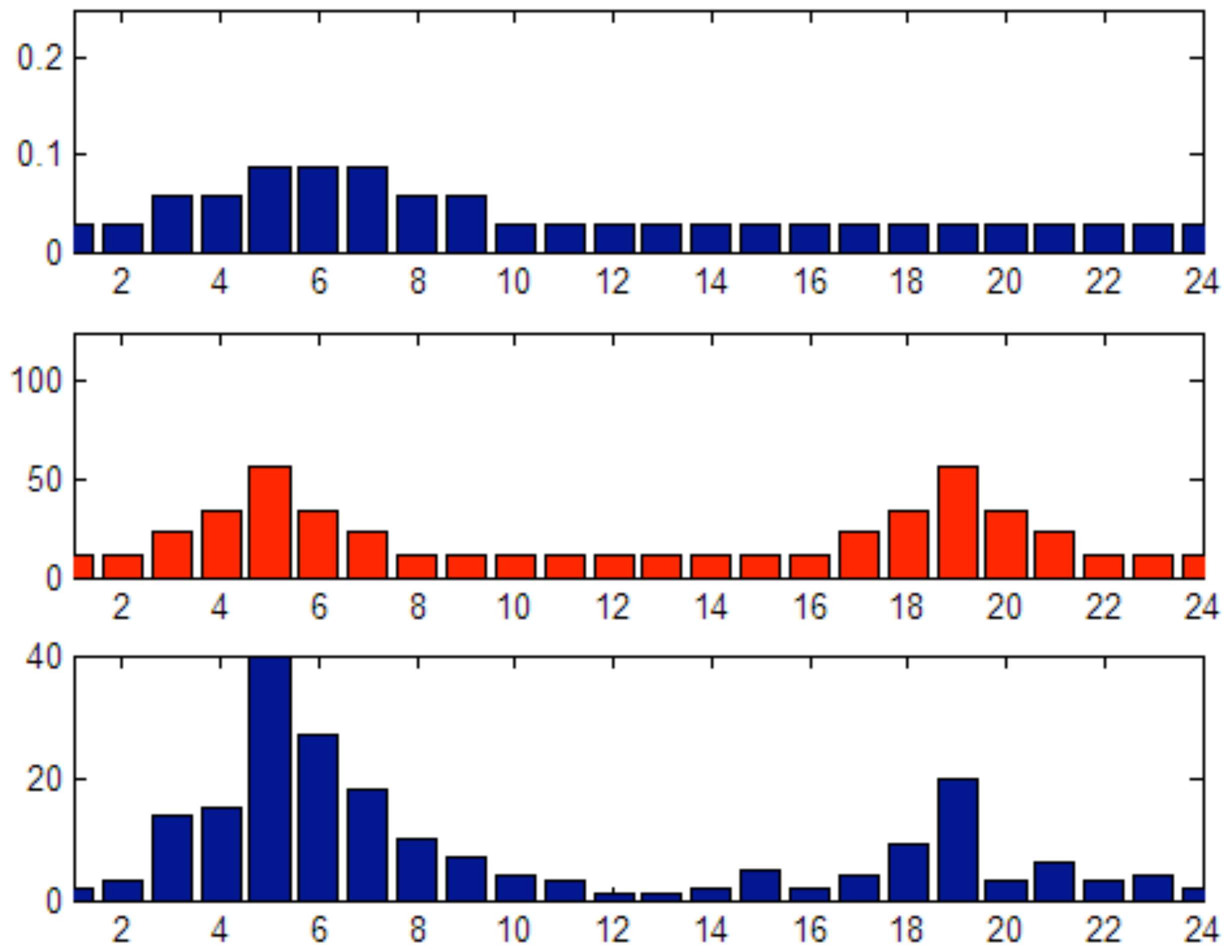


Prior P(x)

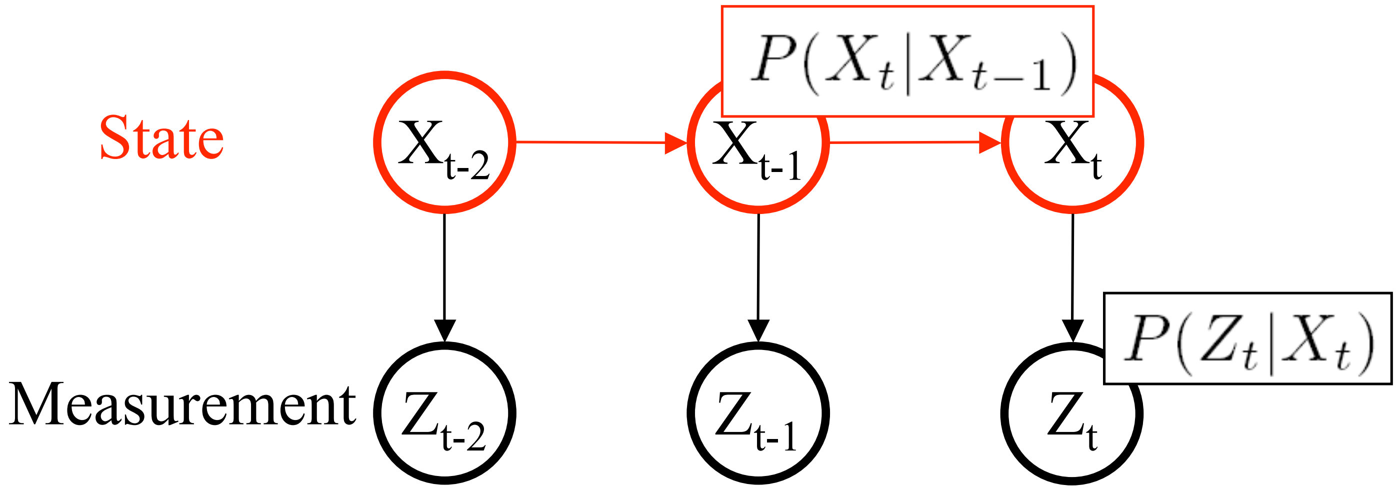
Likelihood L(x;z)

Posterior P(x|z)

1D Importance Sampling

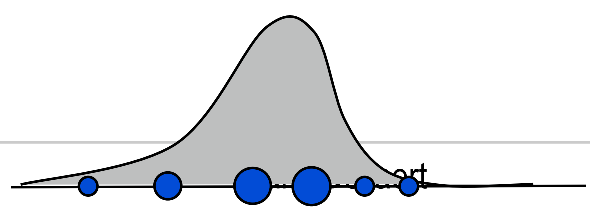


Particle Filter Tracking



Monte Carlo Approximation of Posterior:

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



Bayes Filter and Particle Filter

Motion Model

Recursive Bayes Filter Equation:

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

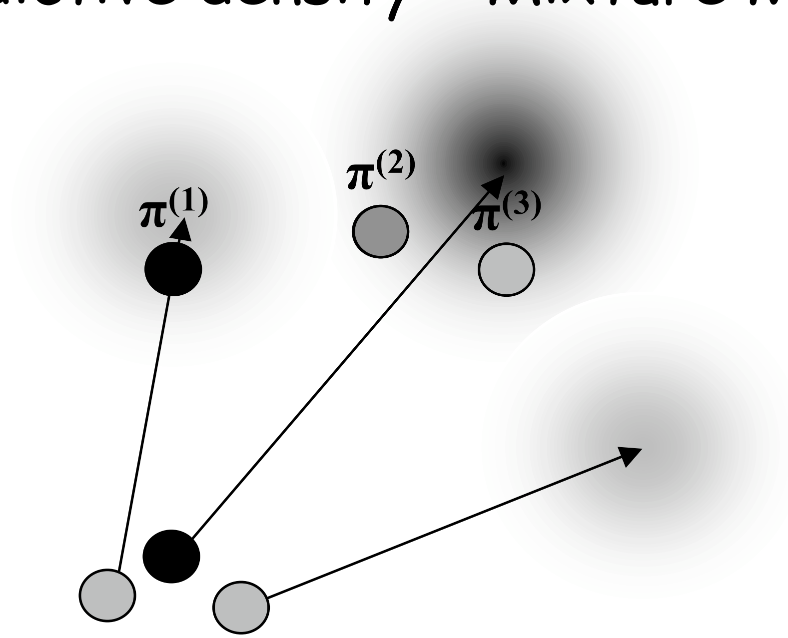
Predictive Density

Monte Carlo Approximation:

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

Particle Filter

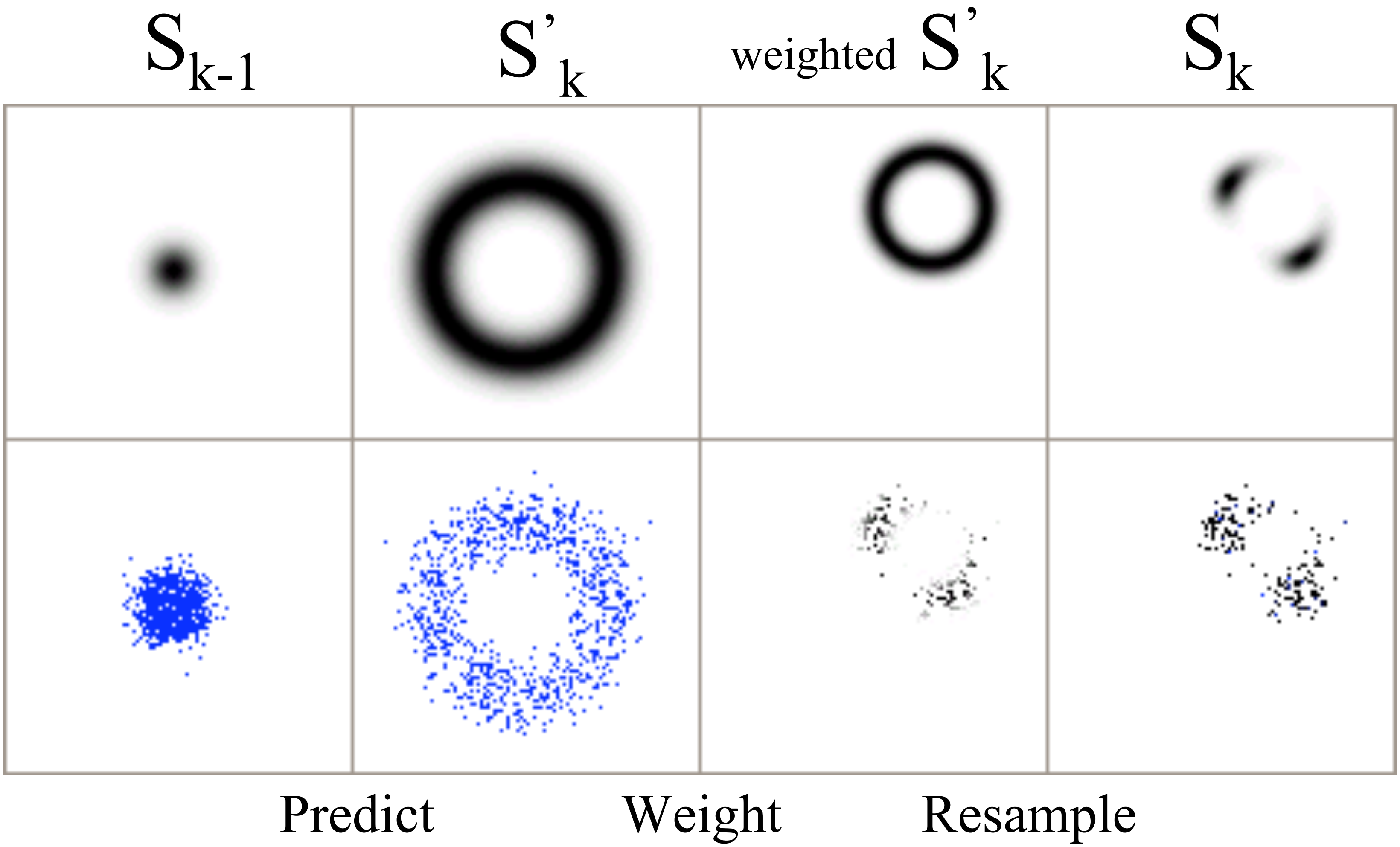
Empirical predictive density = Mixture Model



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

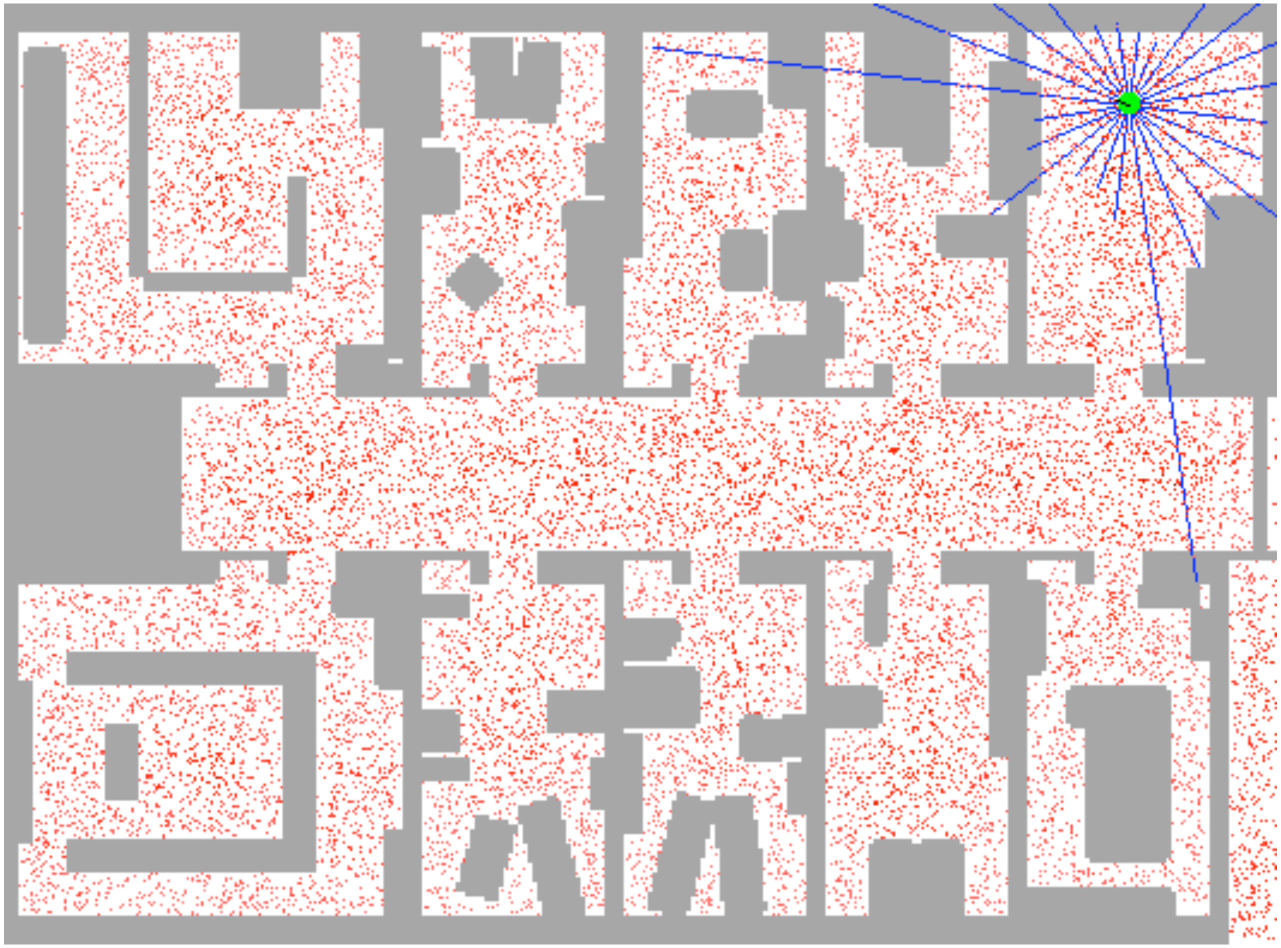
First appeared in 70's, re-discovered by Kitagawa, Isard, …

Monte Carlo Localization



3D Particle filter for robot pose: Monte Carlo Localization

Dellaert, Fox, Burgard & Thrun ICRA 99



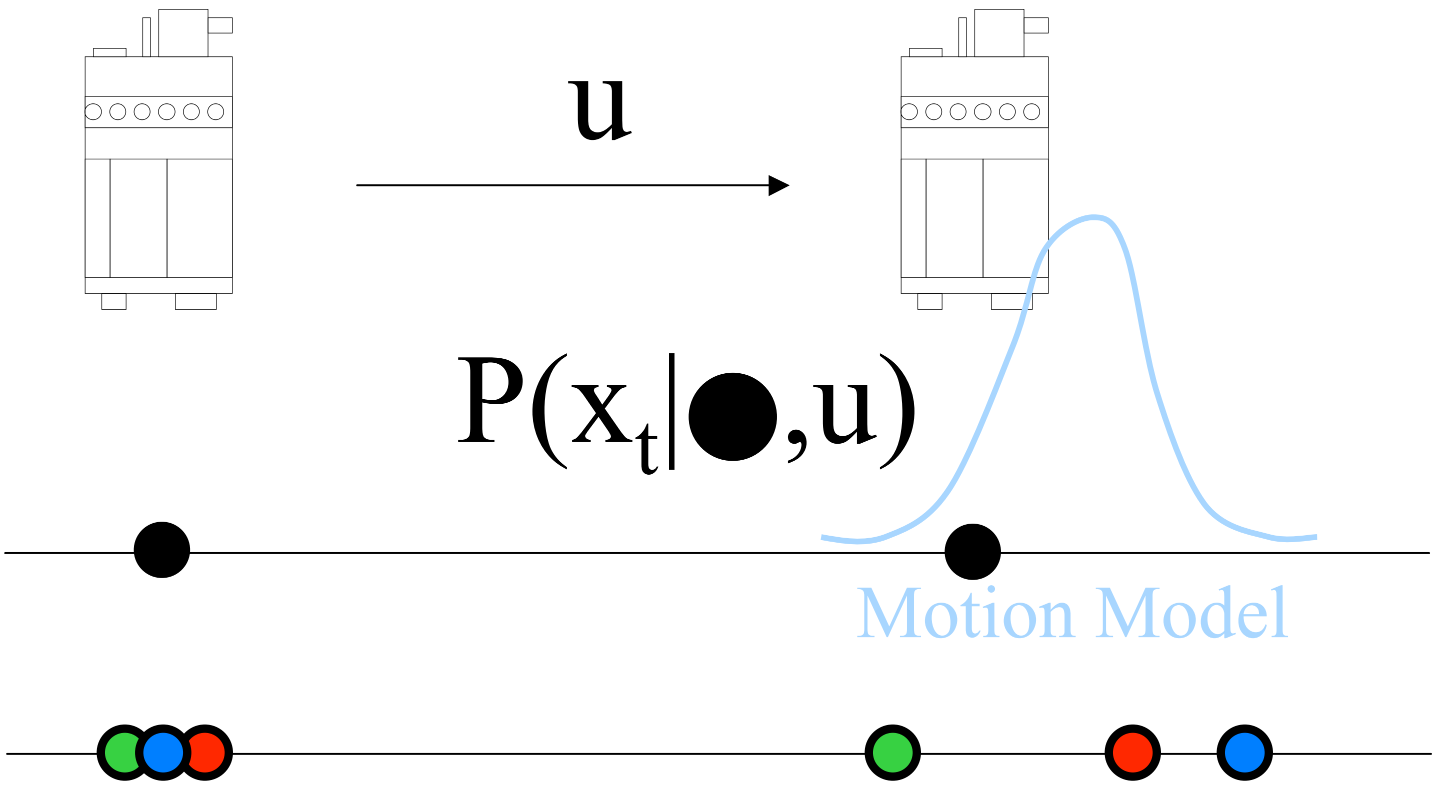
Resampling

- Importance Sampling => weighted
- To get back a fair sample:
  - Resample from the weighted samples according to the importance weights
  - efficient O(N) algorithms exist

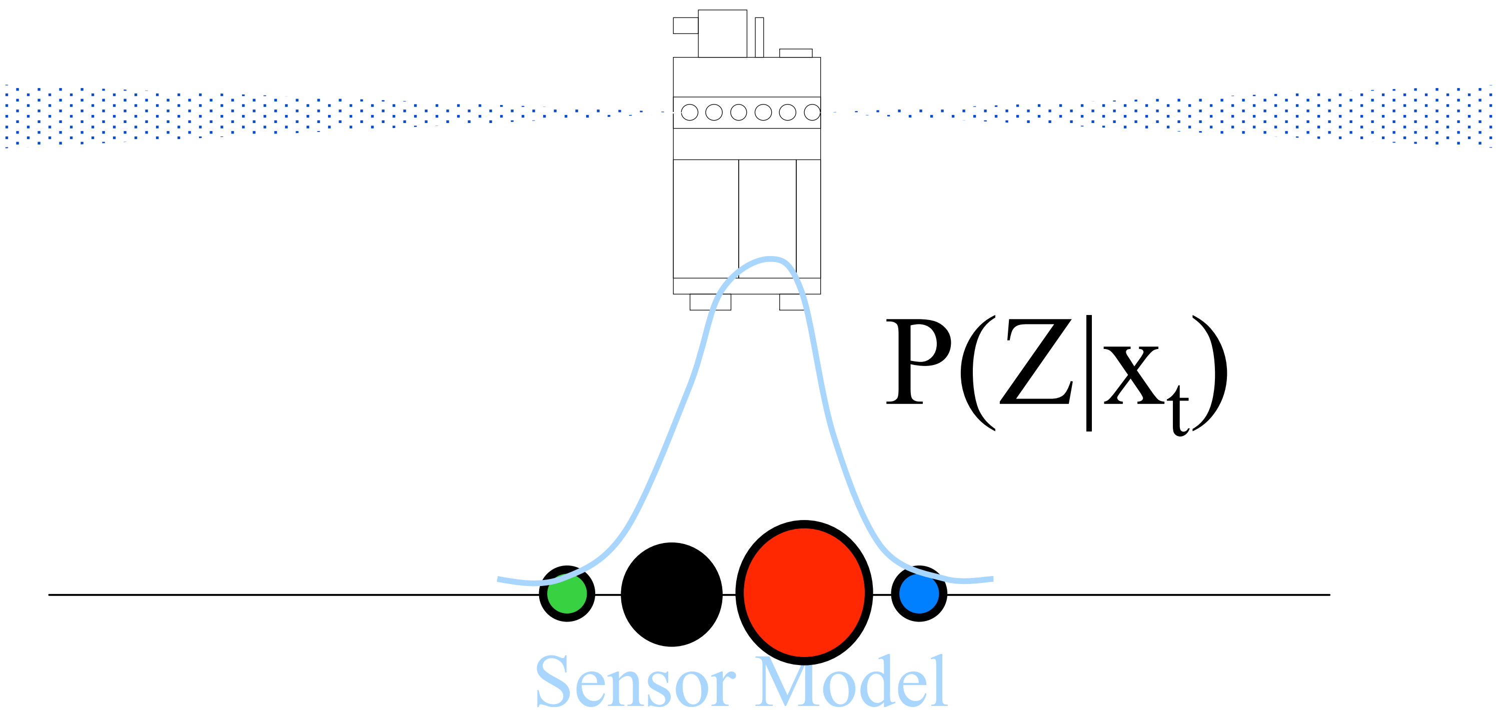
Condensation Algorithm

- Sequential Estimation
- Iterates over:
  - Prediction with motion model
  - Importance Sampling for Inference
  - Resampling

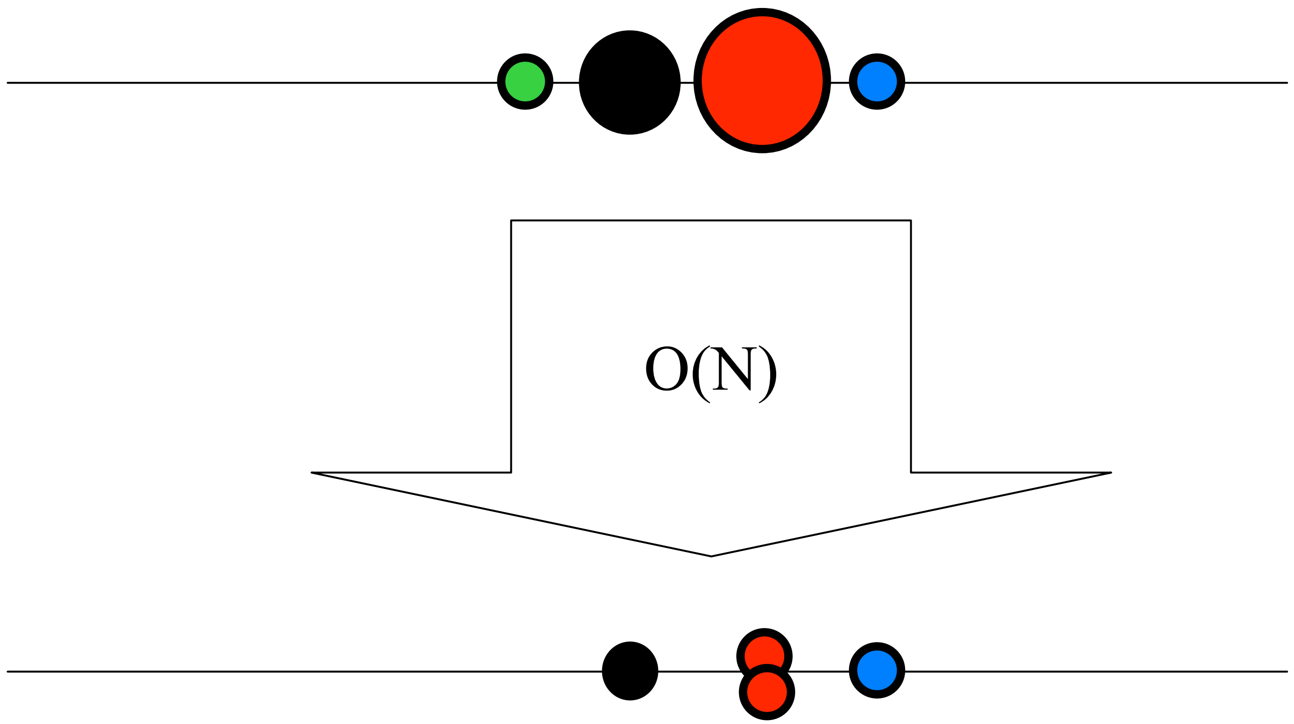
1. Prediction Phase



2. Measurement Phase



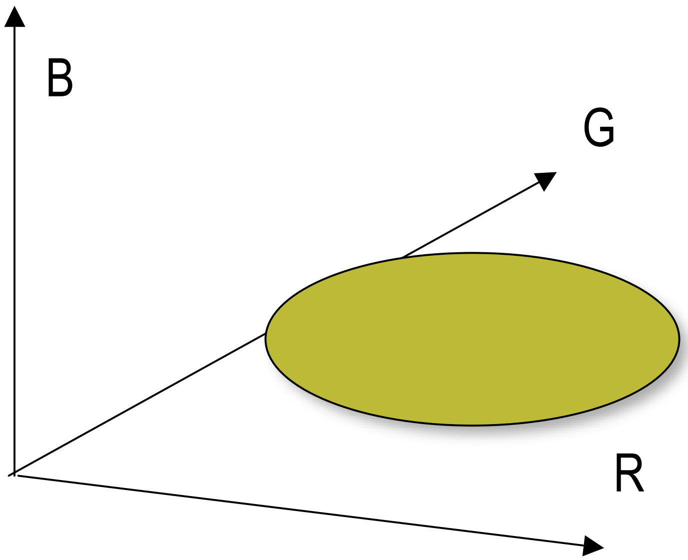
3. Resampling Step



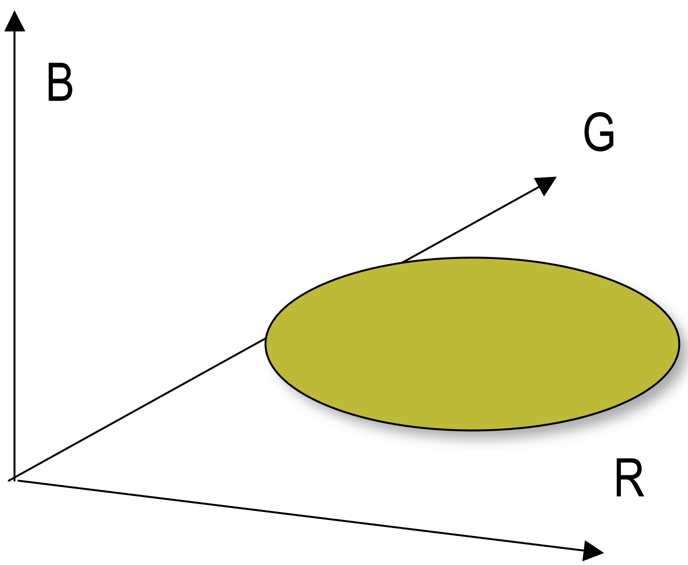
Example 1

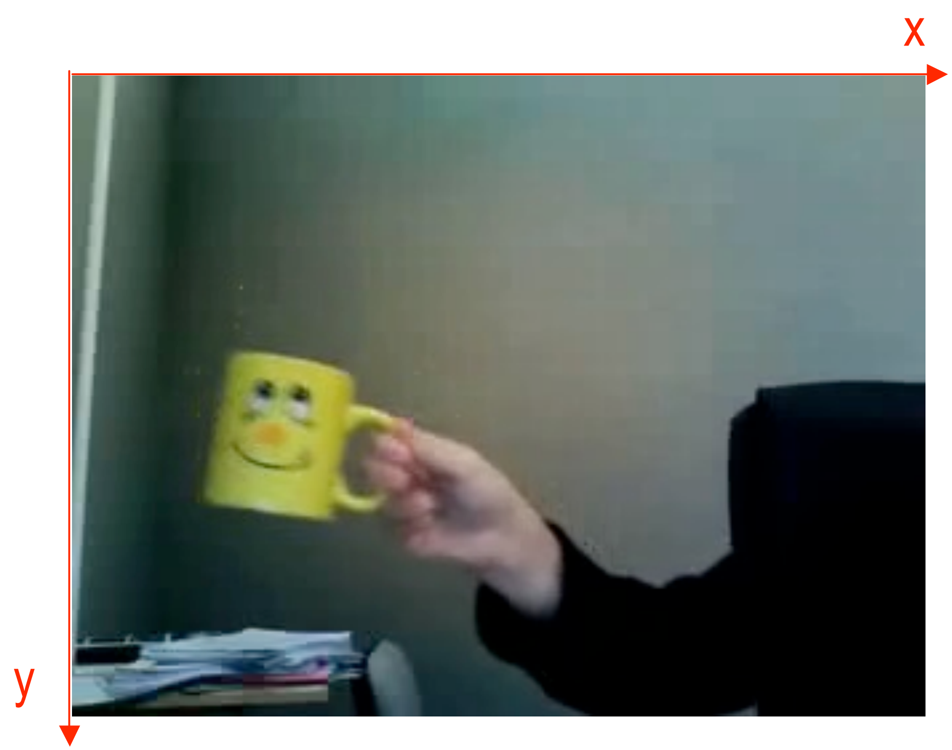
- Start from Importance Sampling w Prior
- Implement Sample Mean
- Try increasing nrSamples
- Implement Resampling Step
- Implement Particle Motion Model

A Simple Color Model



Likelihood





$$P(x, y \mid color) \propto P(color \mid x, y) P(x, y)$$

$$P(color \mid x, y) = \prod_{c=r,g,b} N(\mu_c, \sigma_c \wedge 2)$$