CS 3630!

Lecture 8:
Monte Carlo Inference

Topics

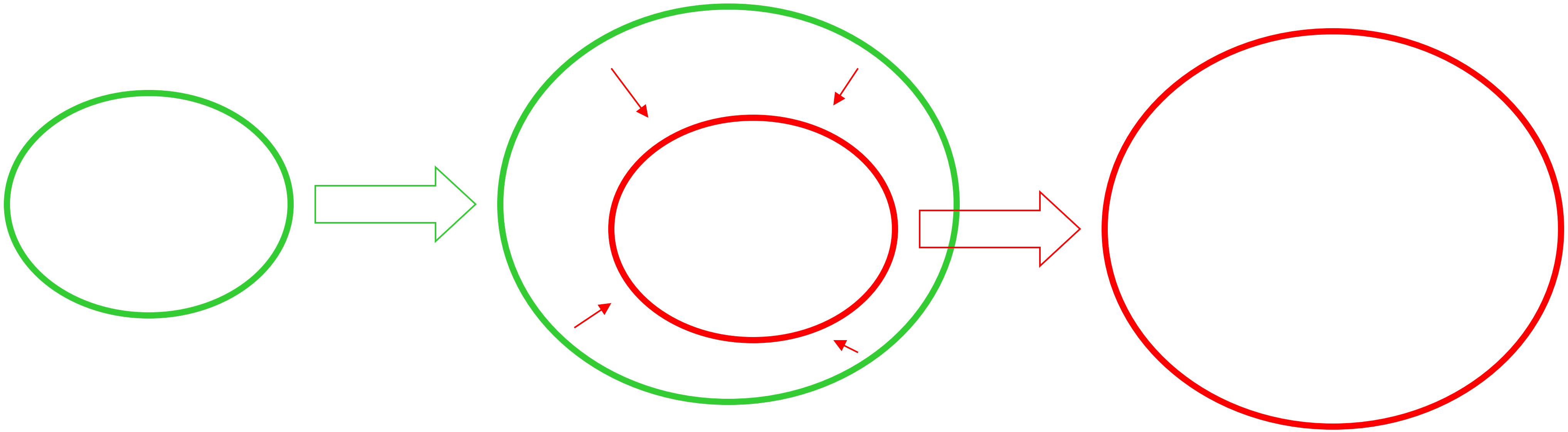
- **1. Continuous Densities**
- **2. Gaussian Densities**
- **3. Bayes Nets & Mixture Models**
- **4. Cont. Measurement Models**
- **5. Cont. Motion Models**
- **6. Simulating Cont. Bayes Nets**
- **7. Sampling as Approximation**
- **8. Importance Sampling**
- **9. Particle Filters & MCL**
- **10. Monte Carlo & Elimination**

Motivation

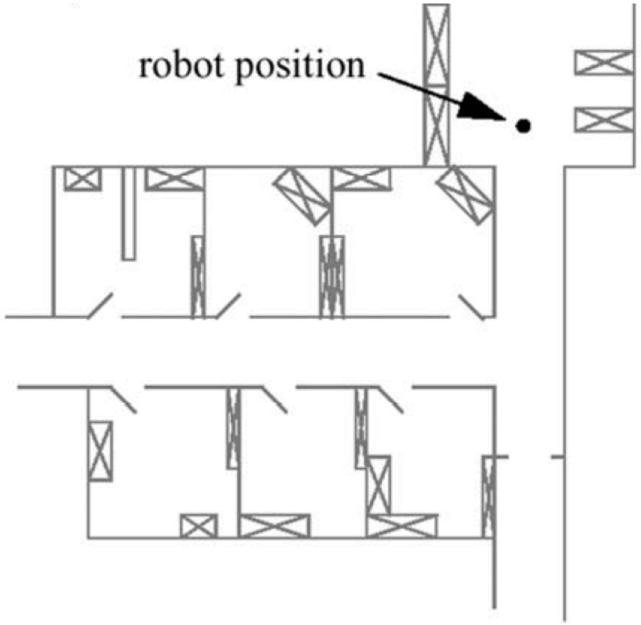
- Robots live in a continuous world
- To localize the robot, we need probabilistic inference
- Many of the concepts we discussed before generalize
- In many cases exact inference is intractable -> sampling
- A popular class of algorithm: Particle filters & Monte Carlo Localization

Remember: the Bayes Filter

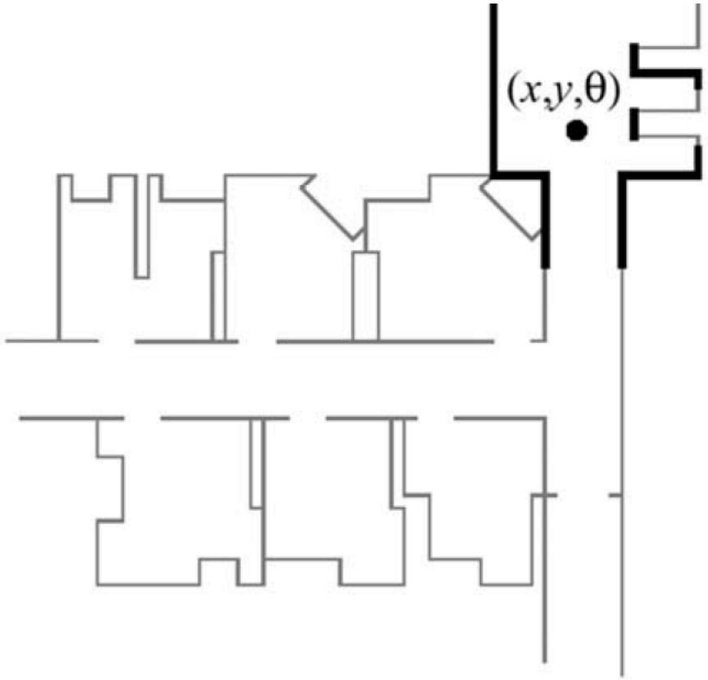
- Two phases: a. Prediction Phase
  b. Measurement Phase



Representations

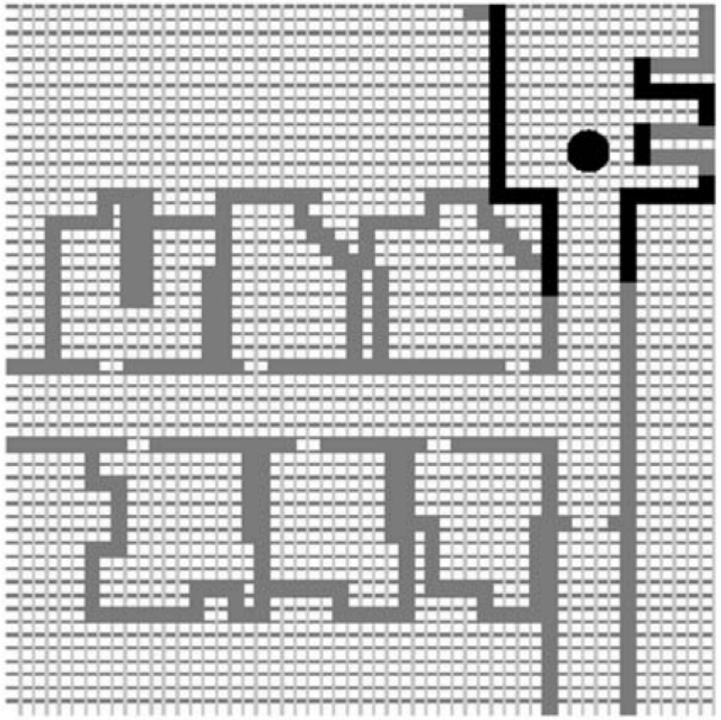


The real map with walls, doors and furniture.

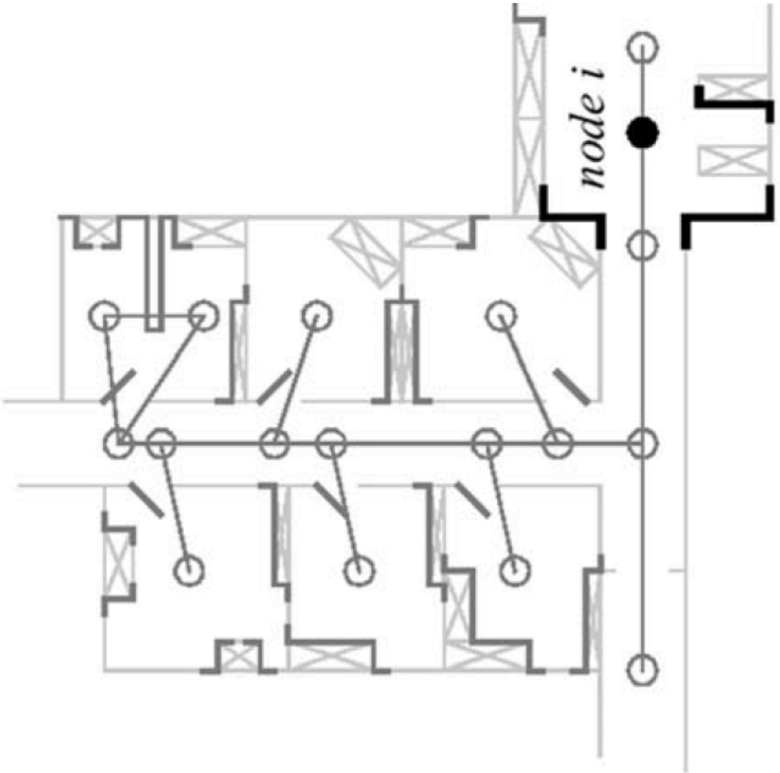


Line-based map (~100 lines)

Representations



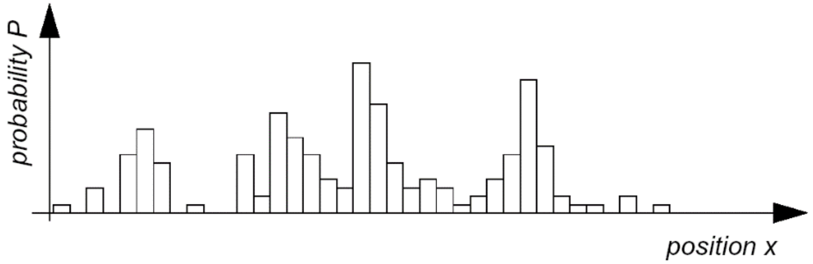
Grid-based map (3000 cells, each 50cm x 50cm)



Topological map (50 features, 18 nodes)

Belief representation: how do we represent our belief of where the robot is located?

Discretized map with multiple hypotheses probability distribution



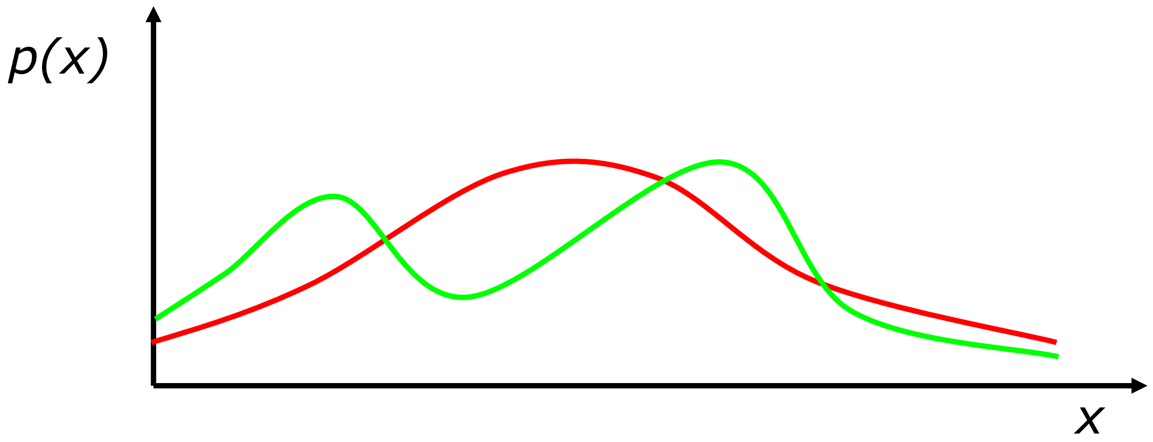
Discretized topological map with with multiple hypotheses probability distribution

1. Continuous Probability Densities

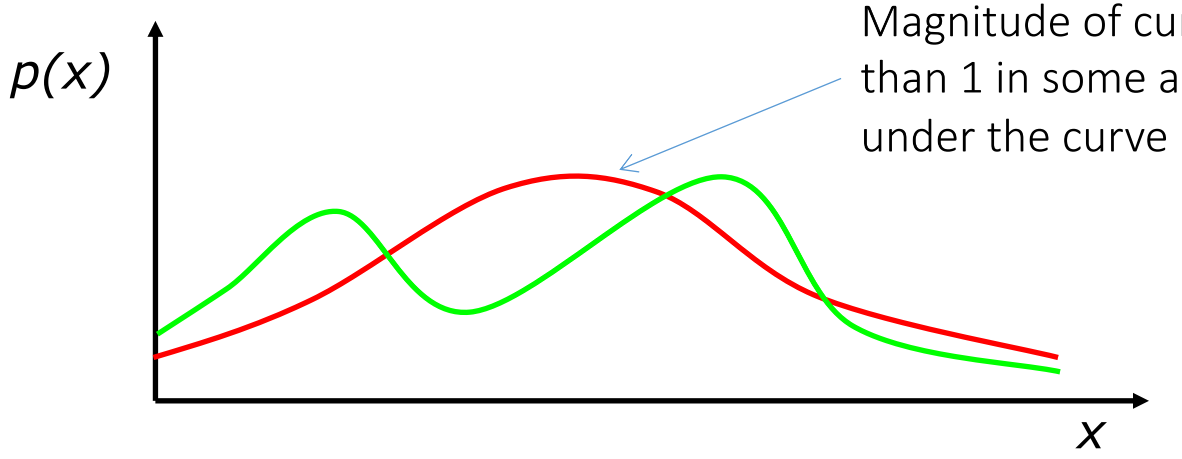
- X takes on values in the continuum.
- $p(X = x)$, or $p(x)$, is a probability density function.

$$P(x \in (a,b)) = \int_a^b p(x)dx$$

- E.g.



Probability Density Function



Magnitude of curve could be greater than 1 in some areas. The total area under the curve must add up to 1.

Since continuous probability functions are defined for an infinite number of points over a continuous interval, the probability at a single point is always 0.

2. Gaussian Densities

A Gaussian probability density is given by

$$\mathcal{N}(\theta; \mu, \Sigma) = \frac{1}{\sqrt{|2\pi\Sigma|}} \exp\left\{ -\frac{1}{2} \|\theta - \mu\|_\Sigma^2 \right\},$$

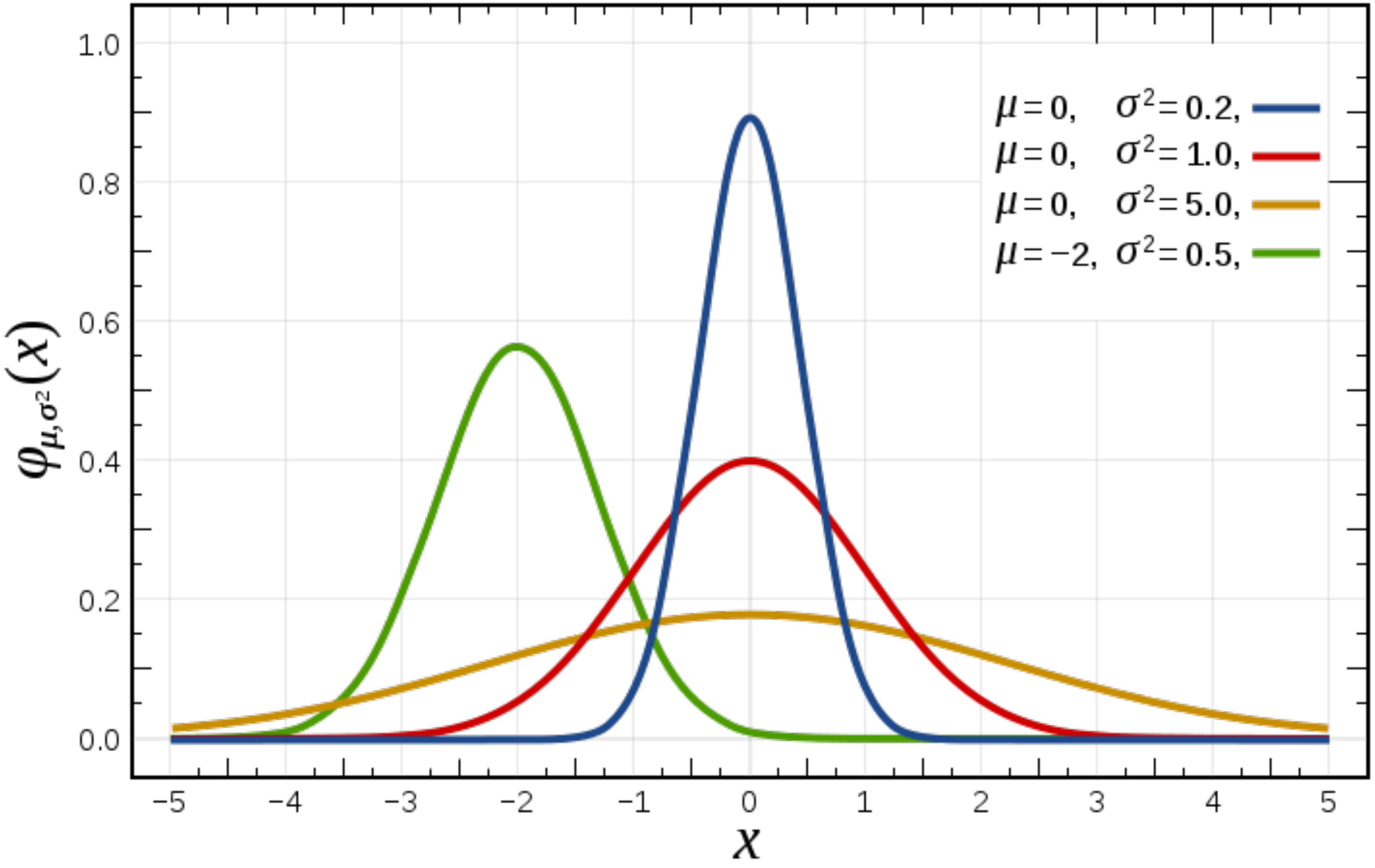
where $\mu \in \mathbb{R}^n$ is the mean, Σ is an $n \times n$ covariance matrix, and

$$\|\theta - \mu\|_\Sigma^2 \stackrel{\Delta}{=} (\theta - \mu)^\top \Sigma^{-1} (\theta - \mu)$$

denotes the squared Mahalanobis distance.

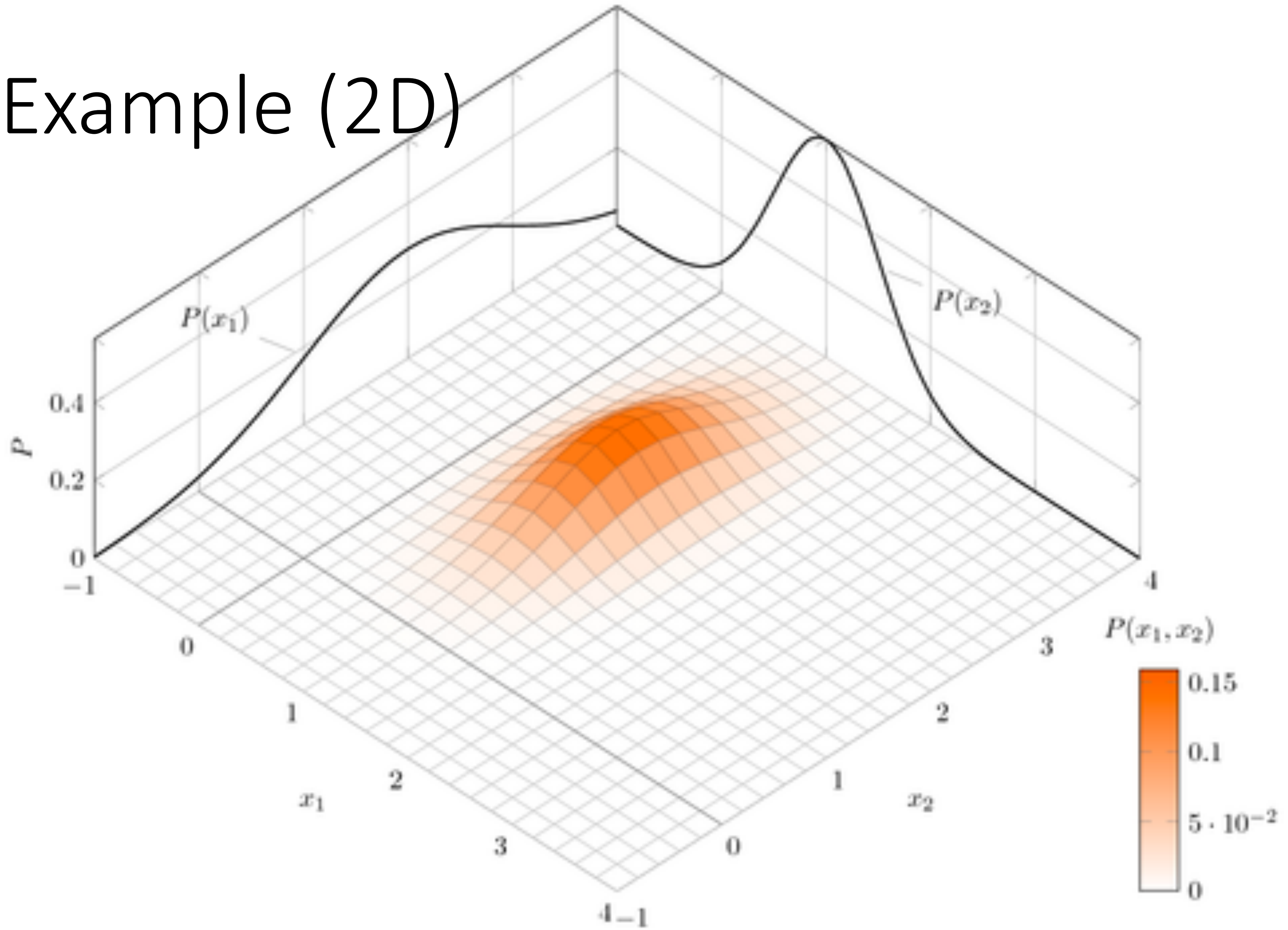
- Easy: negative log is quadratic
- Also known as the "bell curve"
- One of a few densities for which sampling is *easy*

1D examples



- From Wikipedia!

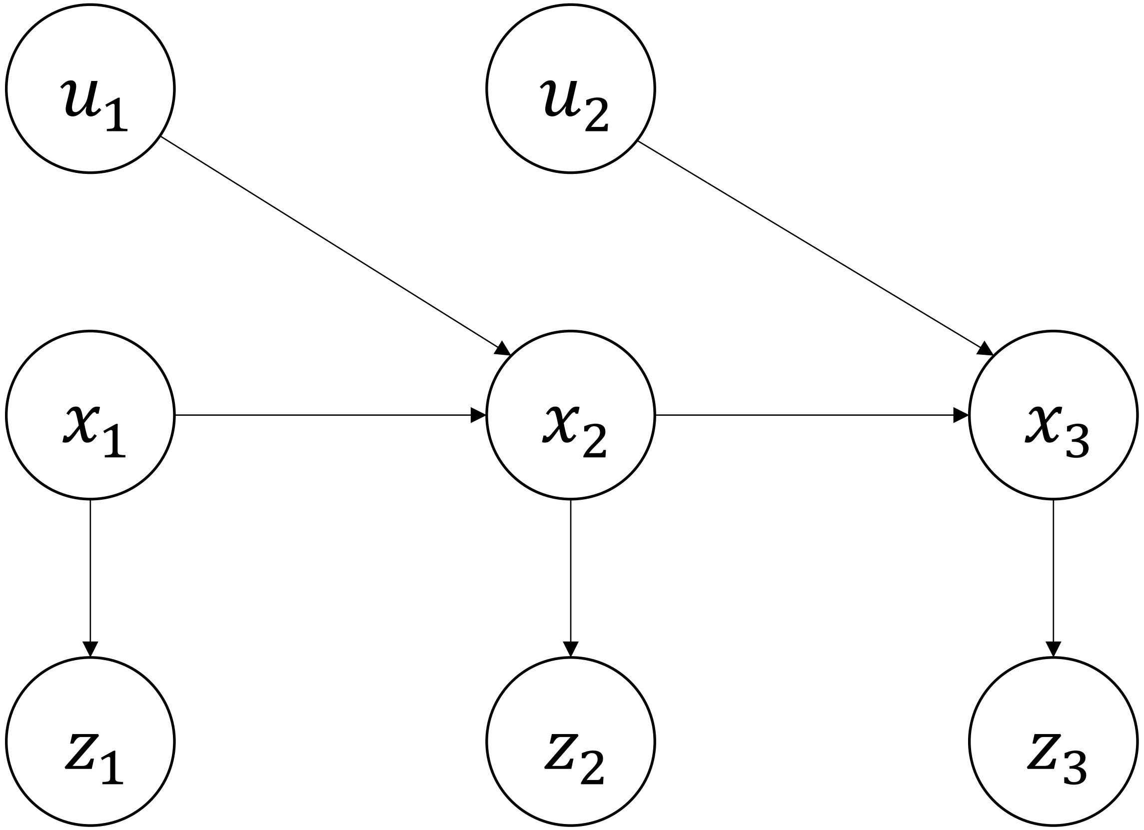
Multivariate Example (2D)



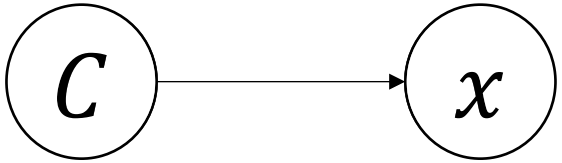
- http://pgfplots.net/tikz/examples/bivariate-normal-distribution/

2. Continuous Bayes Nets

- As before, but now states S, observations O, and action A can all be continuous.
- Terminology: *x, z, u*
- Hence: measurement models and state transition models are continuous.

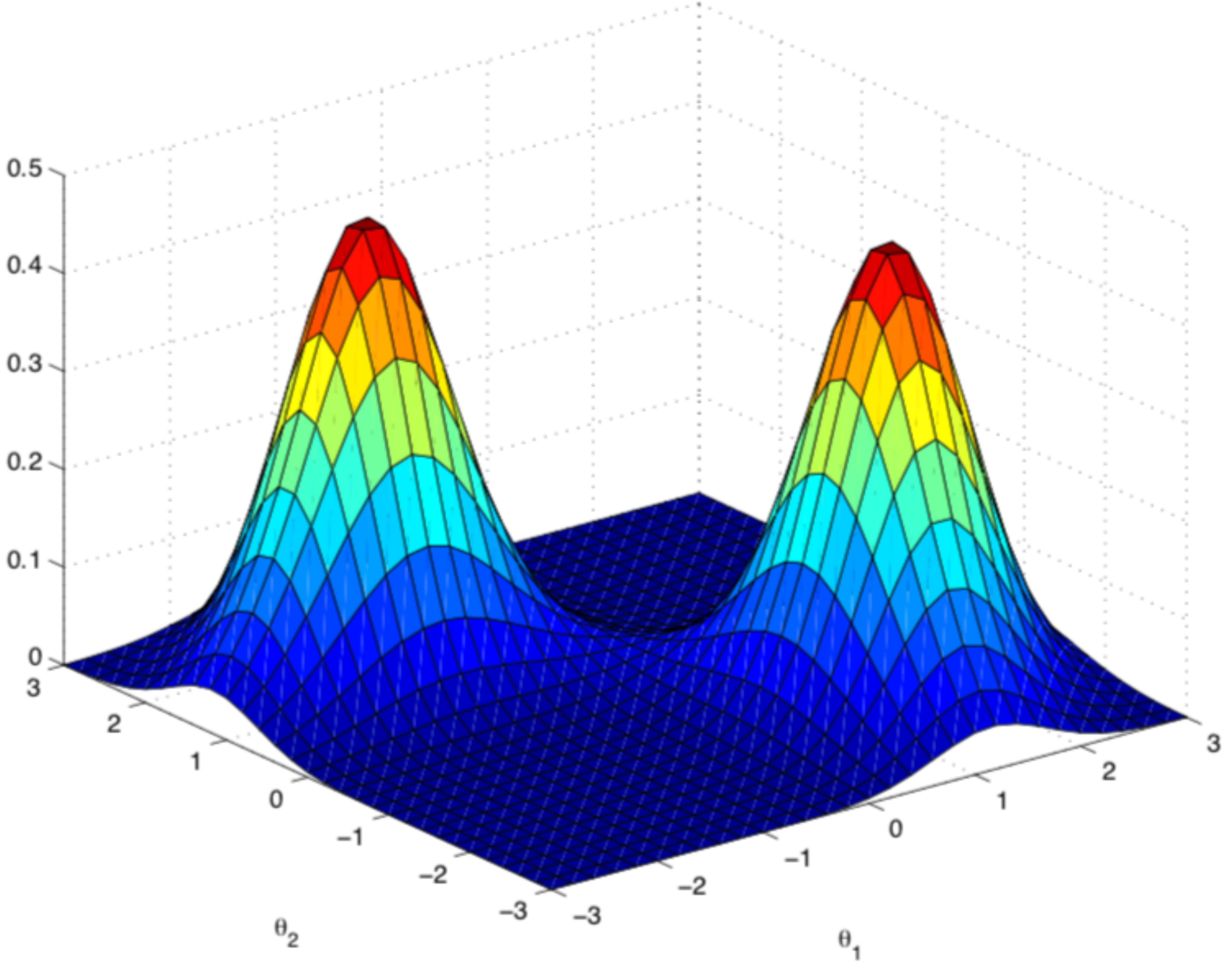


Important aside: Mixture Models



- We can mix discrete and continuous
- Most important example: mixture of continuous densities
- Example: Gaussian mixture model
- Sampling: sample **component**, then sample from Gaussian:

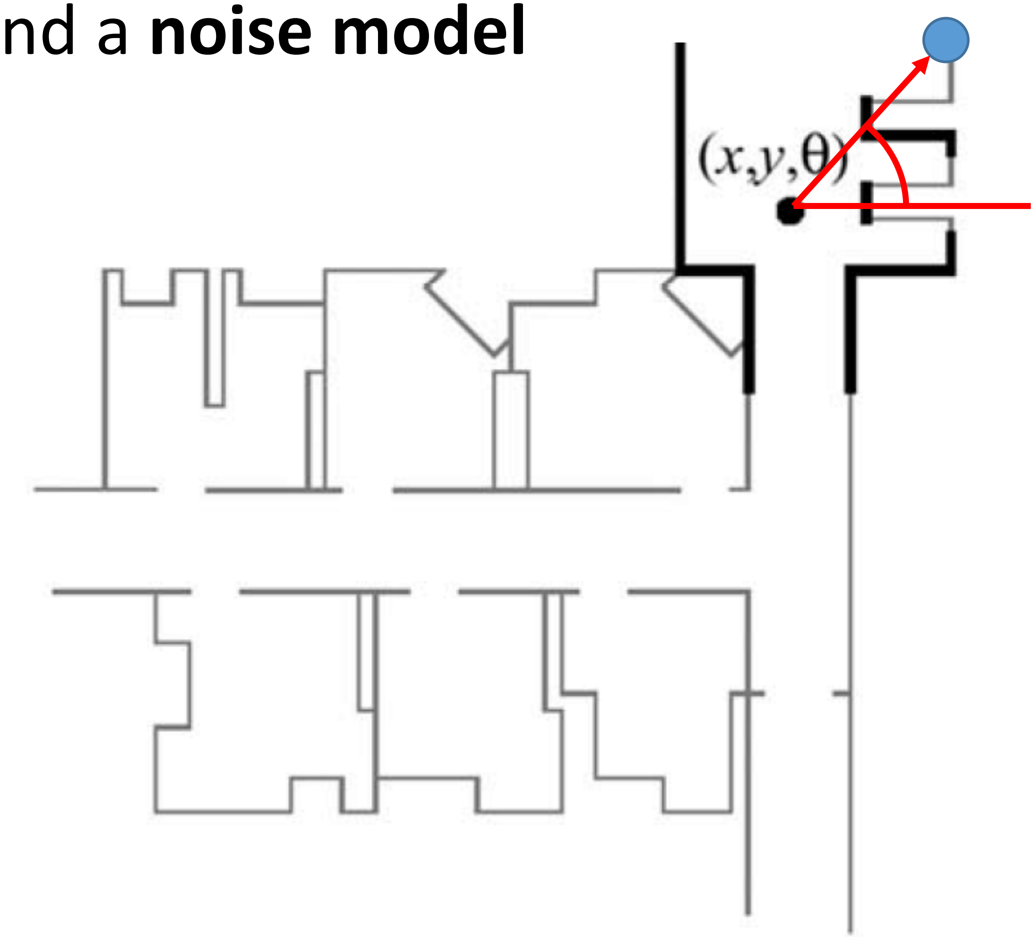
$$p(x,C)=p(x|C)P(C)$$



4. Continuous Measurement Models

- We need a **measurement function** and a **noise model**
- Example: bearing to a landmark l:

$$h(x, l) = atan2(l_y - x_y, l_x - x_x)$$



Adding a noise model

- Generative model of measuremen $z = h(x, l) + \eta,$
- Assuming Gaussian noise:

$$p(z|x,l) = \mathcal{N}(z; h(x,l), R) = \frac{1}{\sqrt{|2\pi R|}} \exp\left\{-\frac{1}{2}\|h(x,l) - z\|_R^2\right\}$$

Adding a noise model

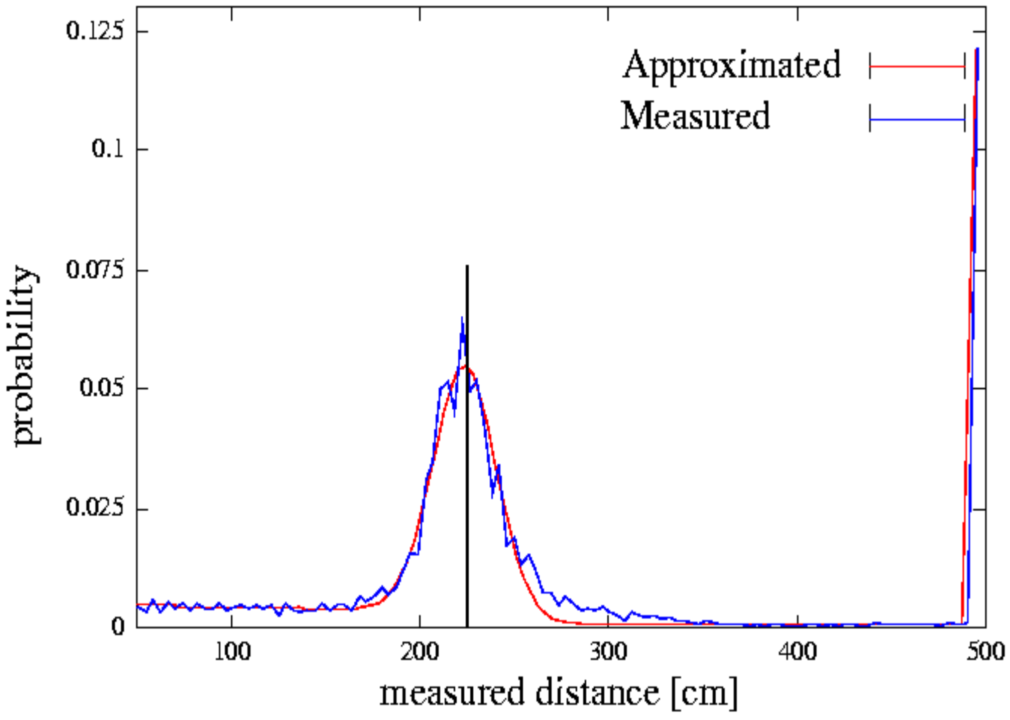
- Generative model of measuremen $z = h(x, l) + \eta,$
- Assuming Gaussian noise:

$$p(z|x,l) = \mathcal{N}(z; h(x,l), R) = \frac{1}{\sqrt{|2\pi R|}} \exp\left\{-\frac{1}{2}\|h(x,l) - z\|_R^2\right\}$$

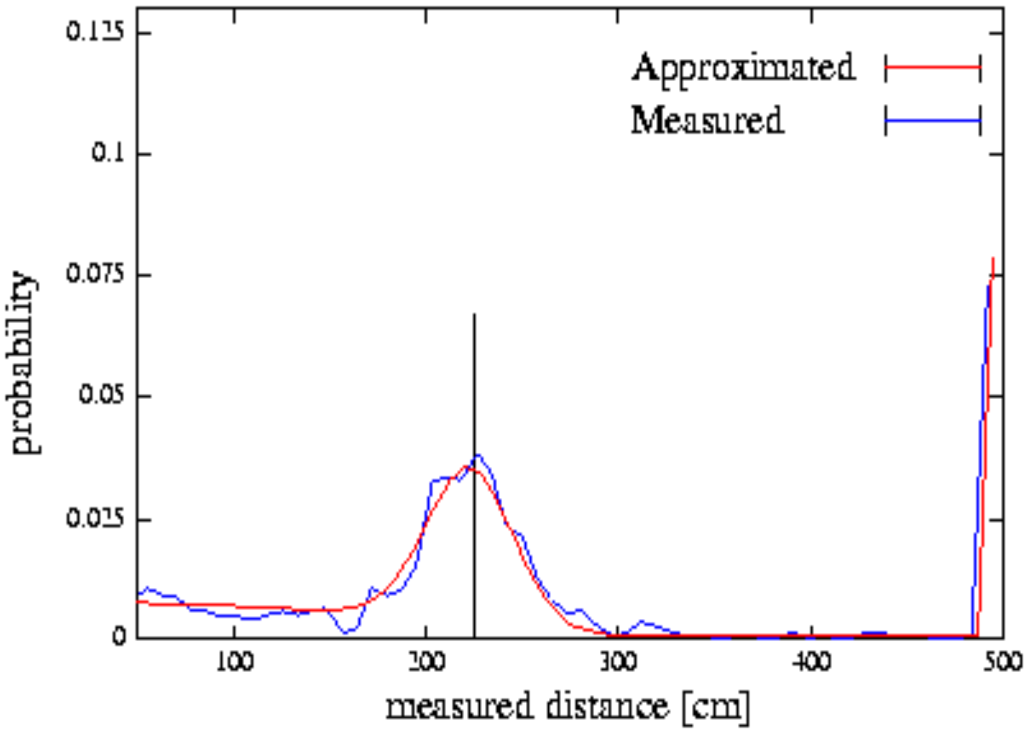
- Putting it together:

$$p(z|x,l) = \frac{1}{\sqrt{|2\pi R|}} \exp\left\{-\frac{1}{2}\|atan2(l_y - x_y, l_x - x_x) - z\|_R^2\right\}$$

Other sensor models



Laser sensor



Sonar sensor

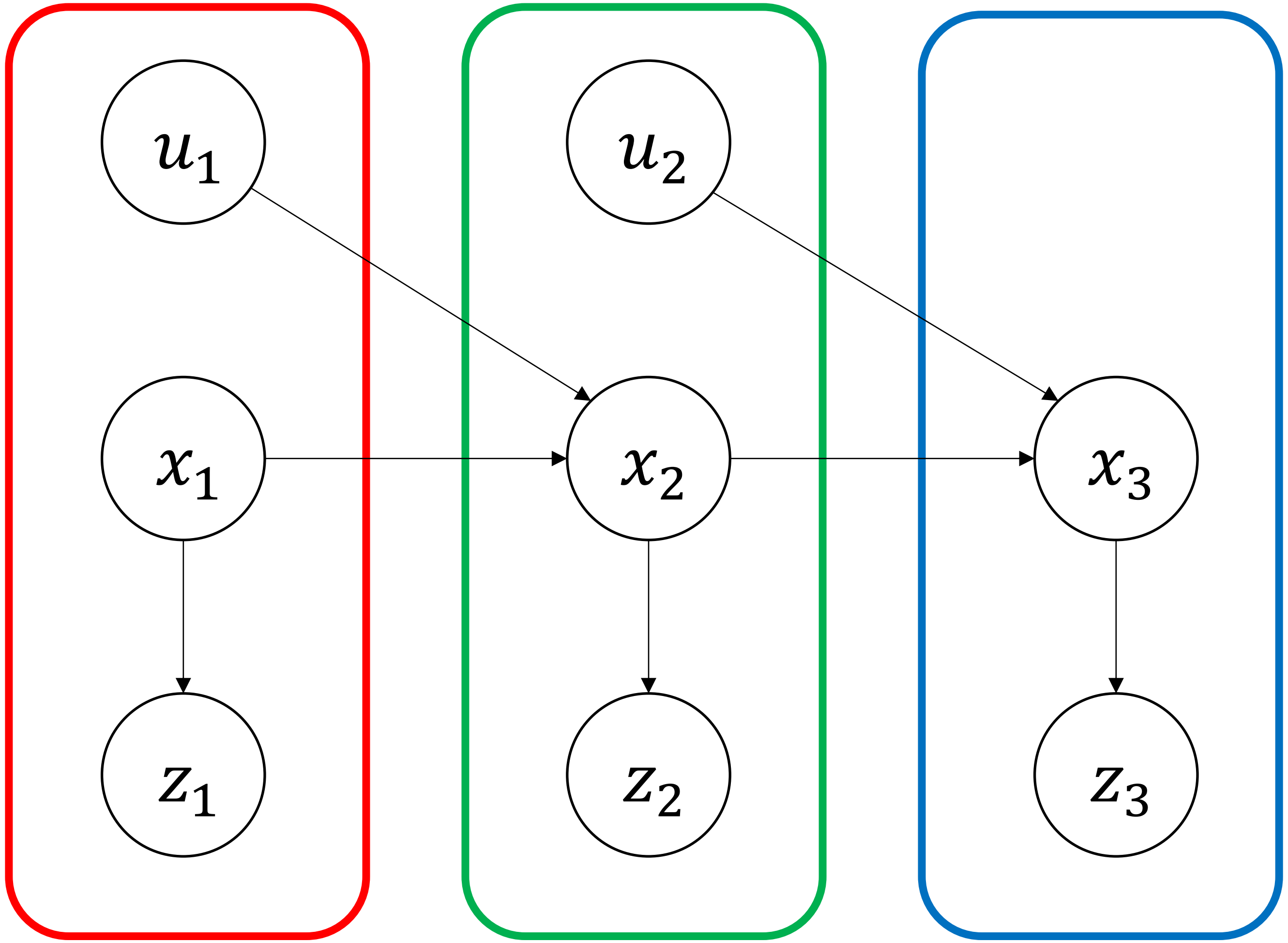
5. Continuous Motion Models

- Similar for state transition, but we now have a motion model
- Motion model $g(x,u)$ takes state x and control u
- Multivariate noise model with covariance Q:

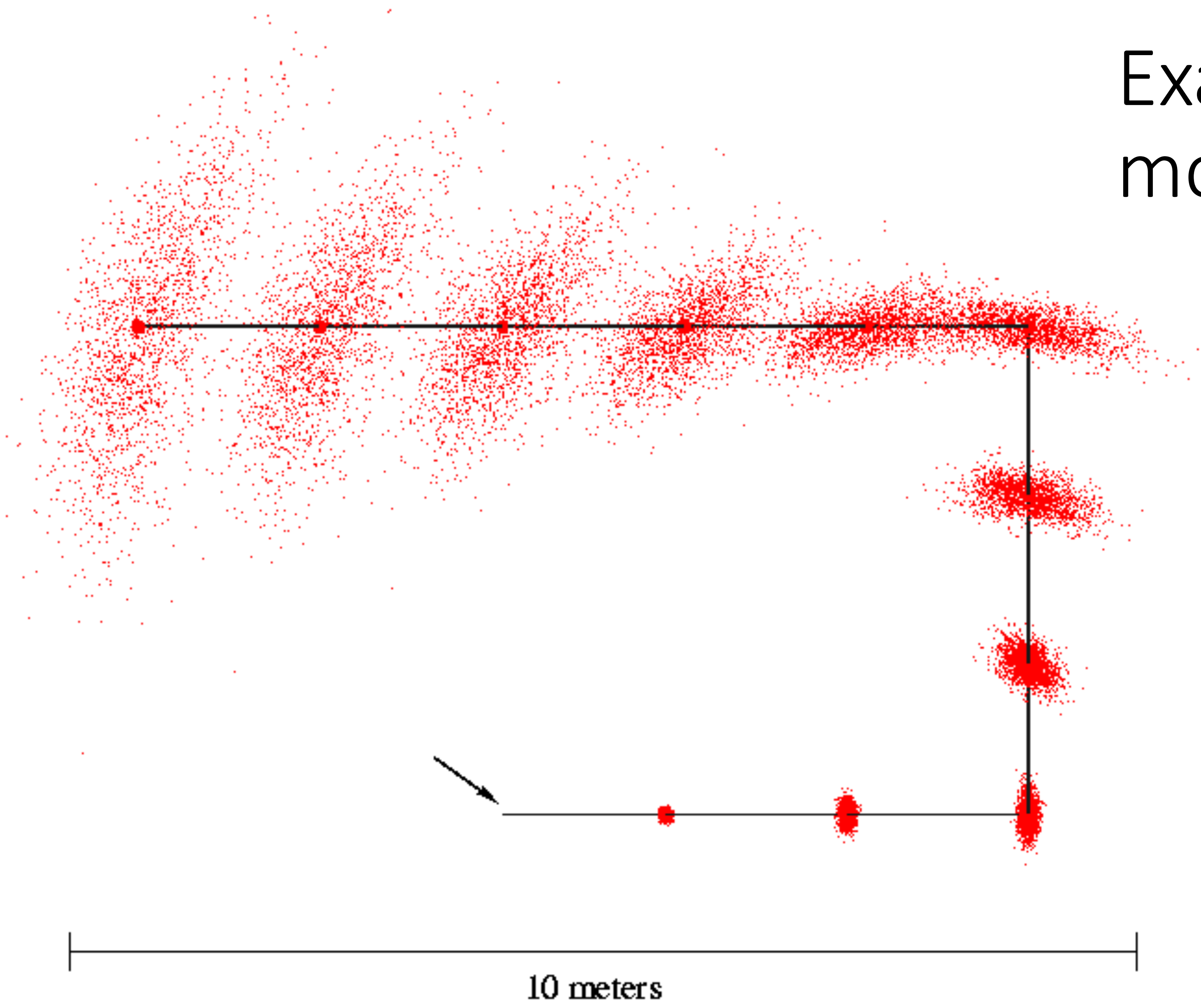
$$p(x_{t+1}|x_t, u_t) = \frac{1}{\sqrt{|2\pi Q|}} \exp\left\{-\frac{1}{2}\|g(x_t, u_t) - x_{t+1}\|_Q^2\right\}$$

6. Simulating from a Continuous Bayes Net

1. Slice 1:
   a) Sample from $p(x_1)$
   b) Sense $p(z_1|x_1)$
   c) Sample from $p(u_1)$
2. Slice 2:
   a) Act $p(x_2|x_1, u_1)$
   b) Sense $p(z_2|x_2)$
   c) Sample from $p(u_2)$
3. Slice 3:
   a) …



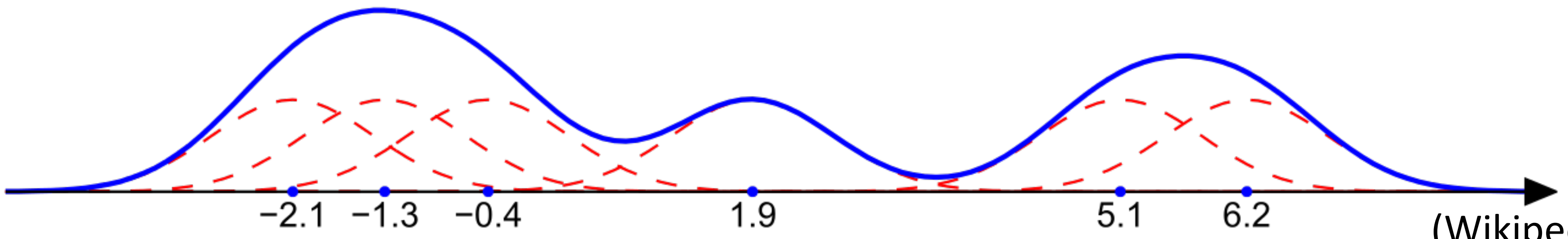
Example: motion model only



- The infamous "banana density"
- Happens because we also sample heading \theta
- Clearly non-Gaussian!

7. Sampling to Approximate Densities

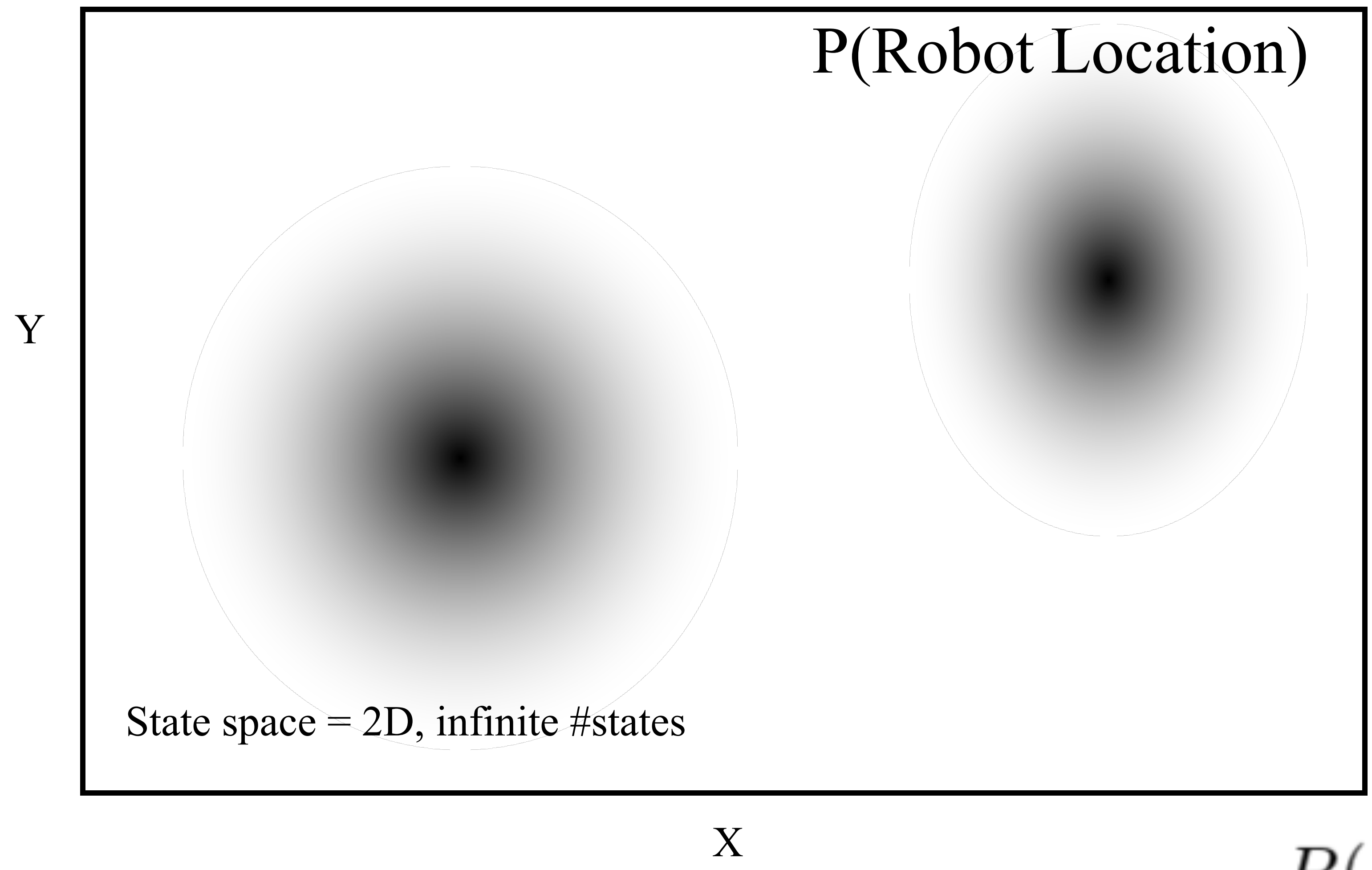
- As banana distribution illustrates, densities can become arbitrarily complex, even when noise models are Gaussian
- Issue is nonlinear measurement and noise models
- One way out: **Parzen window density estimation** (mixtures!)



(Wikipedia)

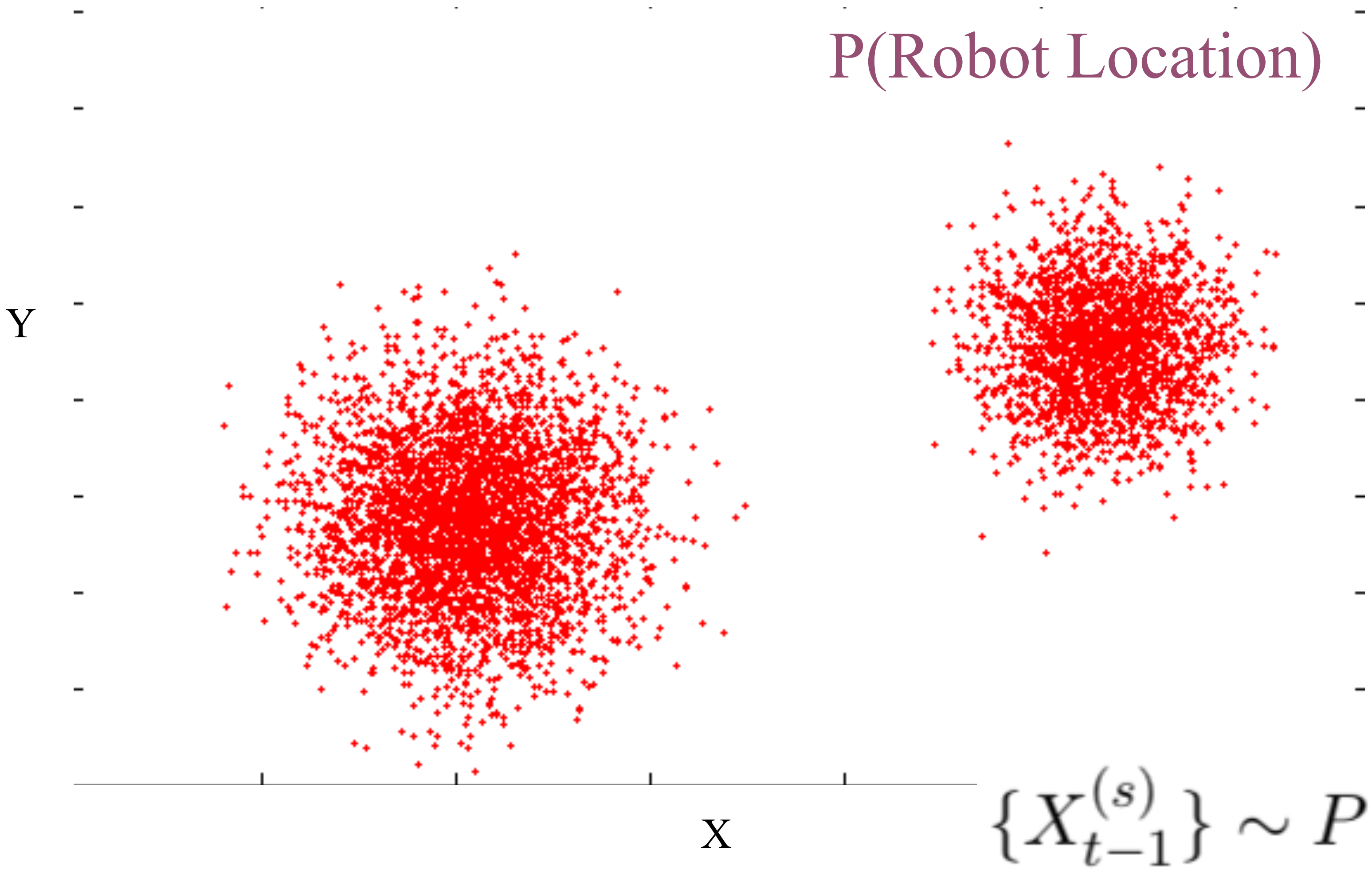
- Other way out: ***sampling!***

Probability of Robot Location



$$P(X_{t-1}|Z^{t-1})$$

Sampling as Representation



$$\{X_{t-1}^{(s)}\} \sim P(X_{t-1} | Z^{t-1})$$

Sampling Advantages

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

- minus: Approximate

8. Importance Sampling

- Additionally use weights to represent a density

$$\{X_{t-1}^{(r)}, \pi_{t-1}^{(r)}\} \sim P(X_{t-1}|Z^{t-1})$$

- Generic importance sampling idea:
  - We want to sample from *p(x)*, but we don't know how
  - sample $x^{(r)}$ from *q(x)*, which some way we can sample from
  - give each sample $x^{(r)}$ an importance weight equal to *p(x)/q(x)*

Importance Sampling

- Sample $x^{(r)}$ from $q(x)$
  - $\pi_r = p(x^{(r)})/q(x^{(r)})$

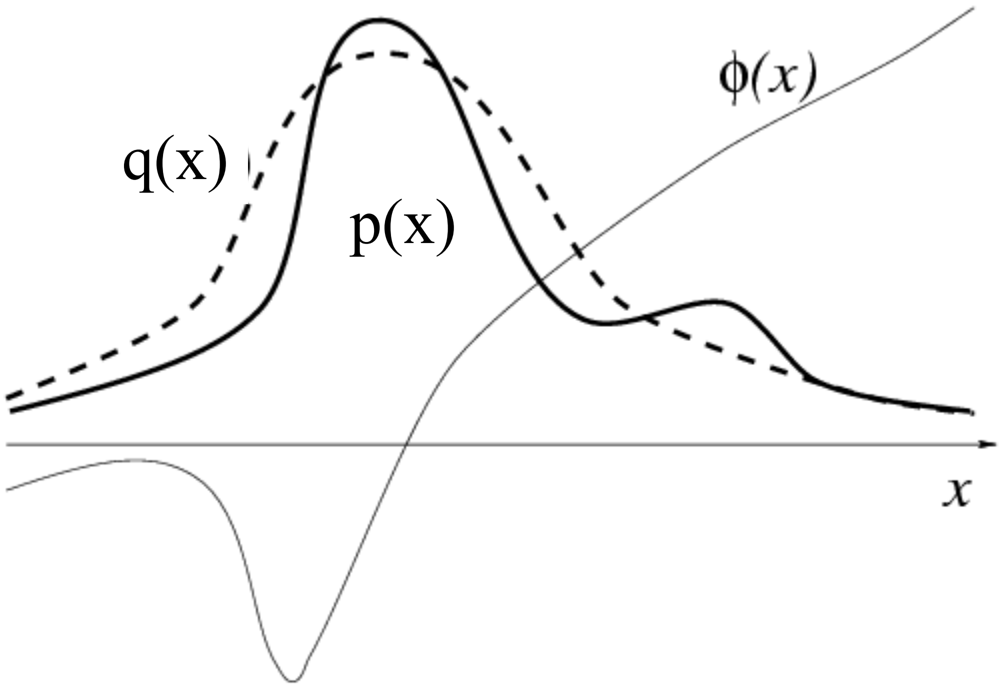
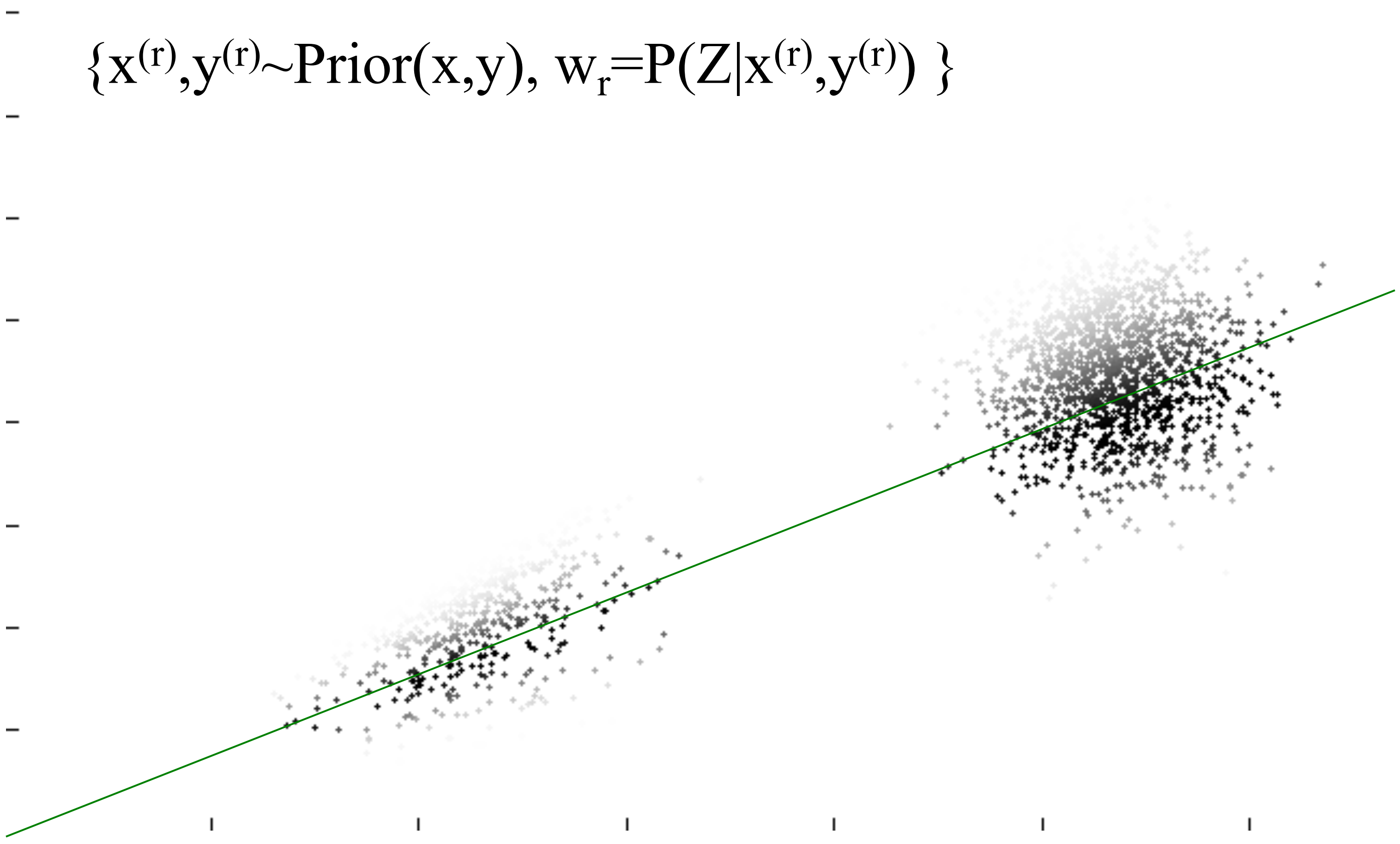


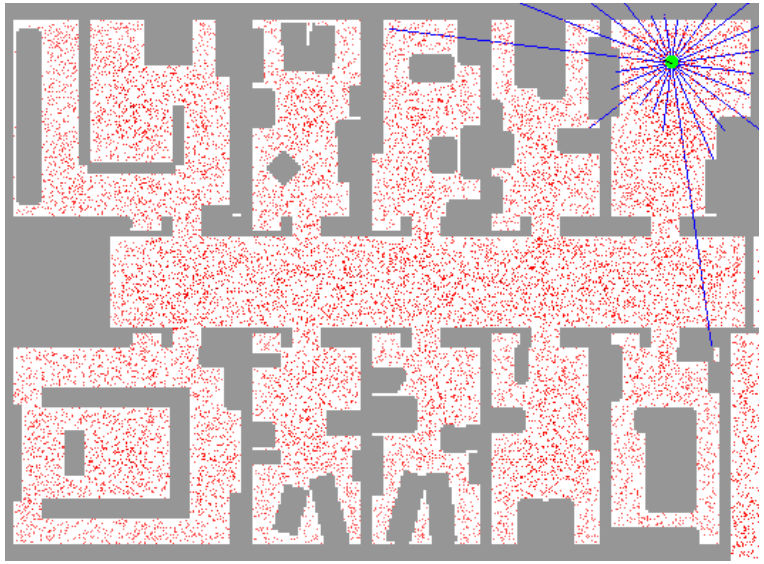
Image by MacKay

Example: Bayes law via importance sampling

$\{x^{(r)}, y^{(r)} \sim \mathrm{Prior}(x,y),\ w_r = P(Z|x^{(r)}, y^{(r)})\ \}$



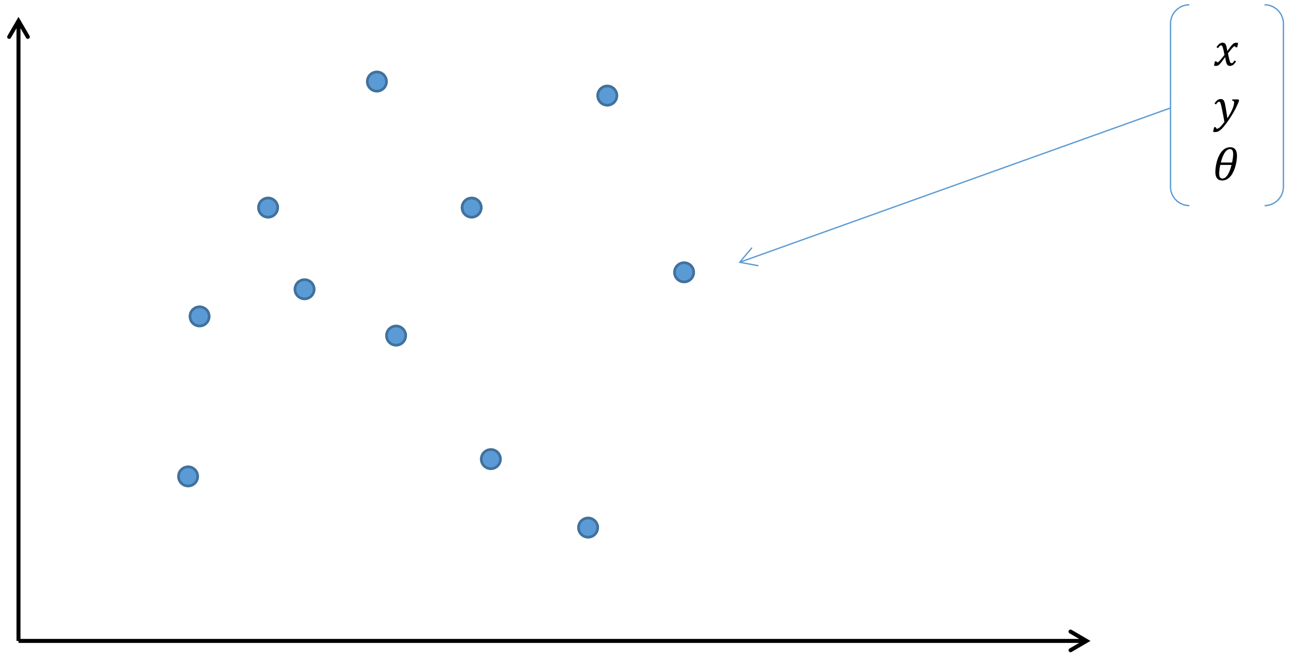
9. Particle Filters & Monte Carlo Localization



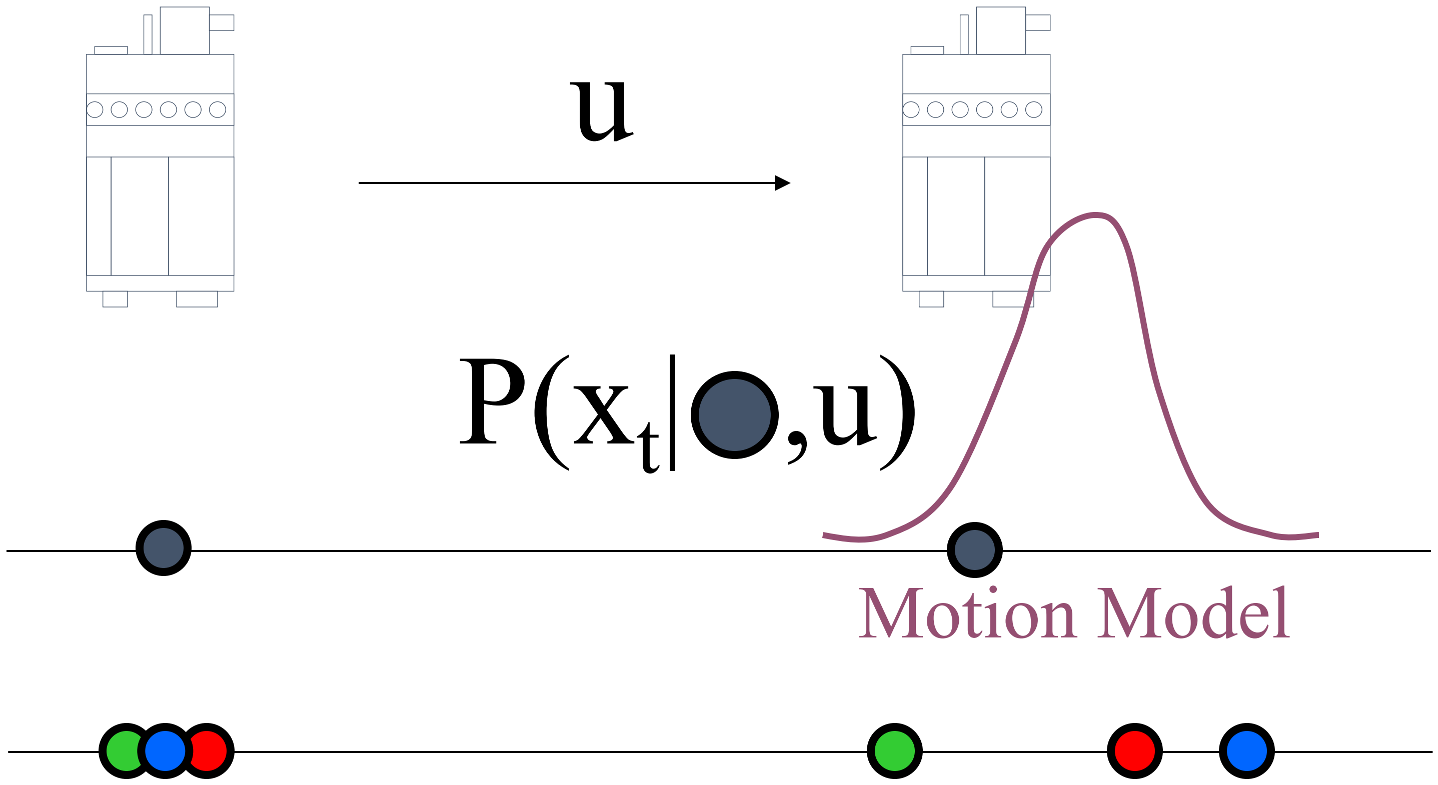
- Bayes filter using importance sampling for Bayes law
- First appeared in 70's, re-discovered by Kitagawa,
- Isard & Blake rediscovered in computer vision, as CONDENSATION
- Monte Carlo Localization in robotics

Particles

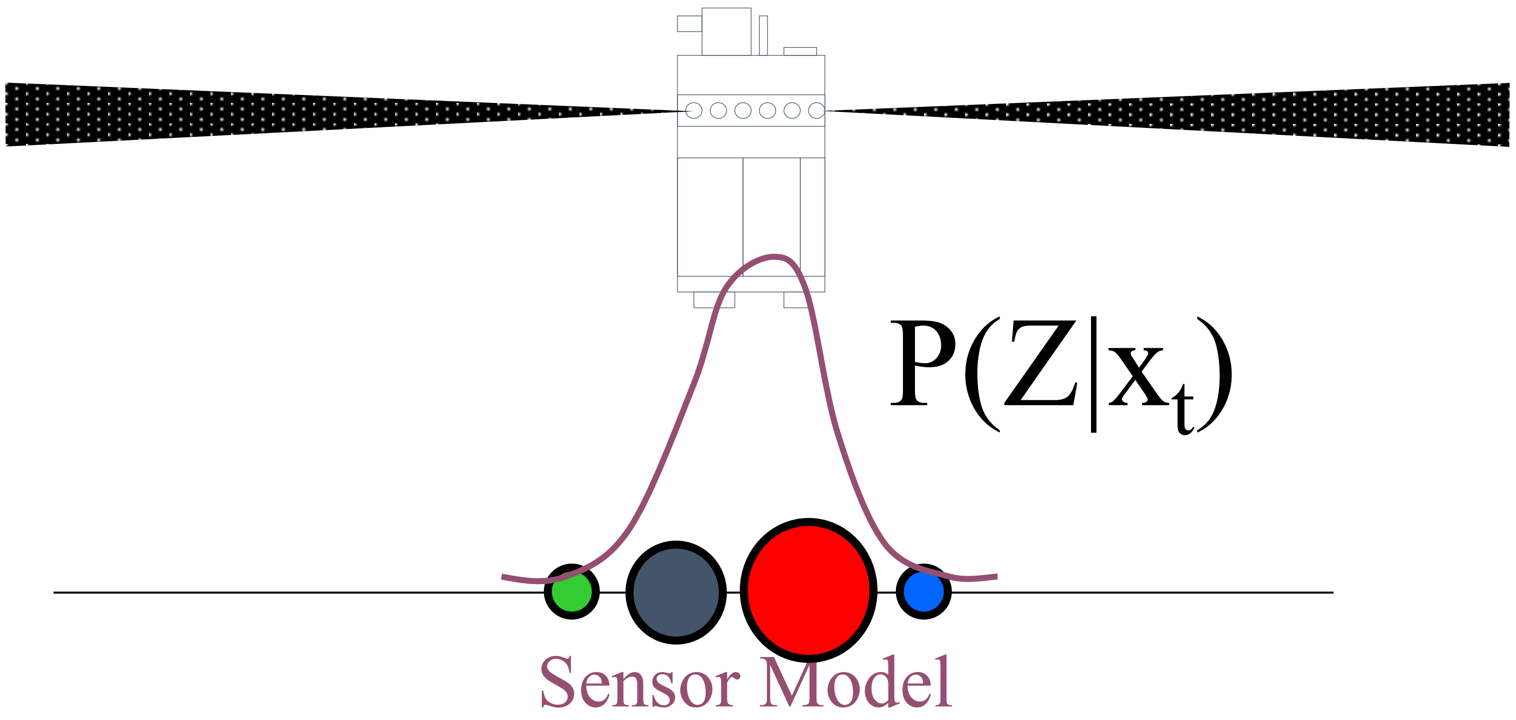
- Each particle is a guess about where the robot might be



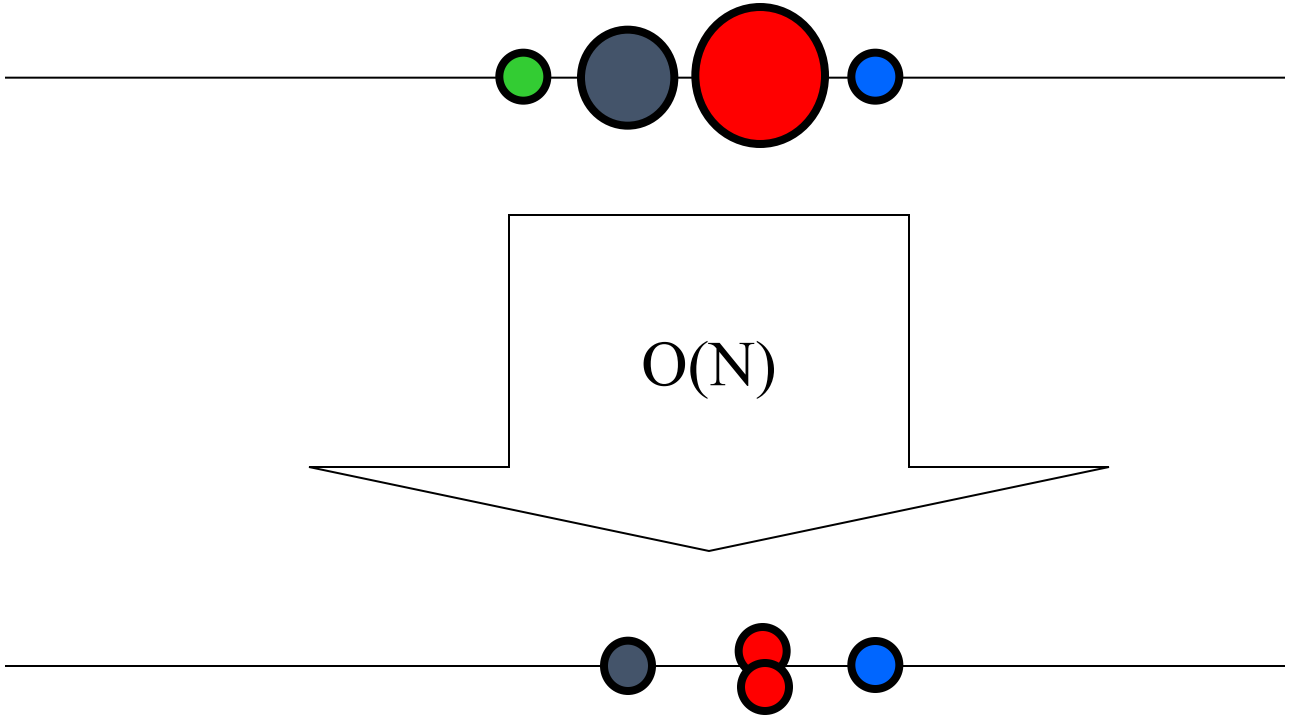
1. Prediction Phase

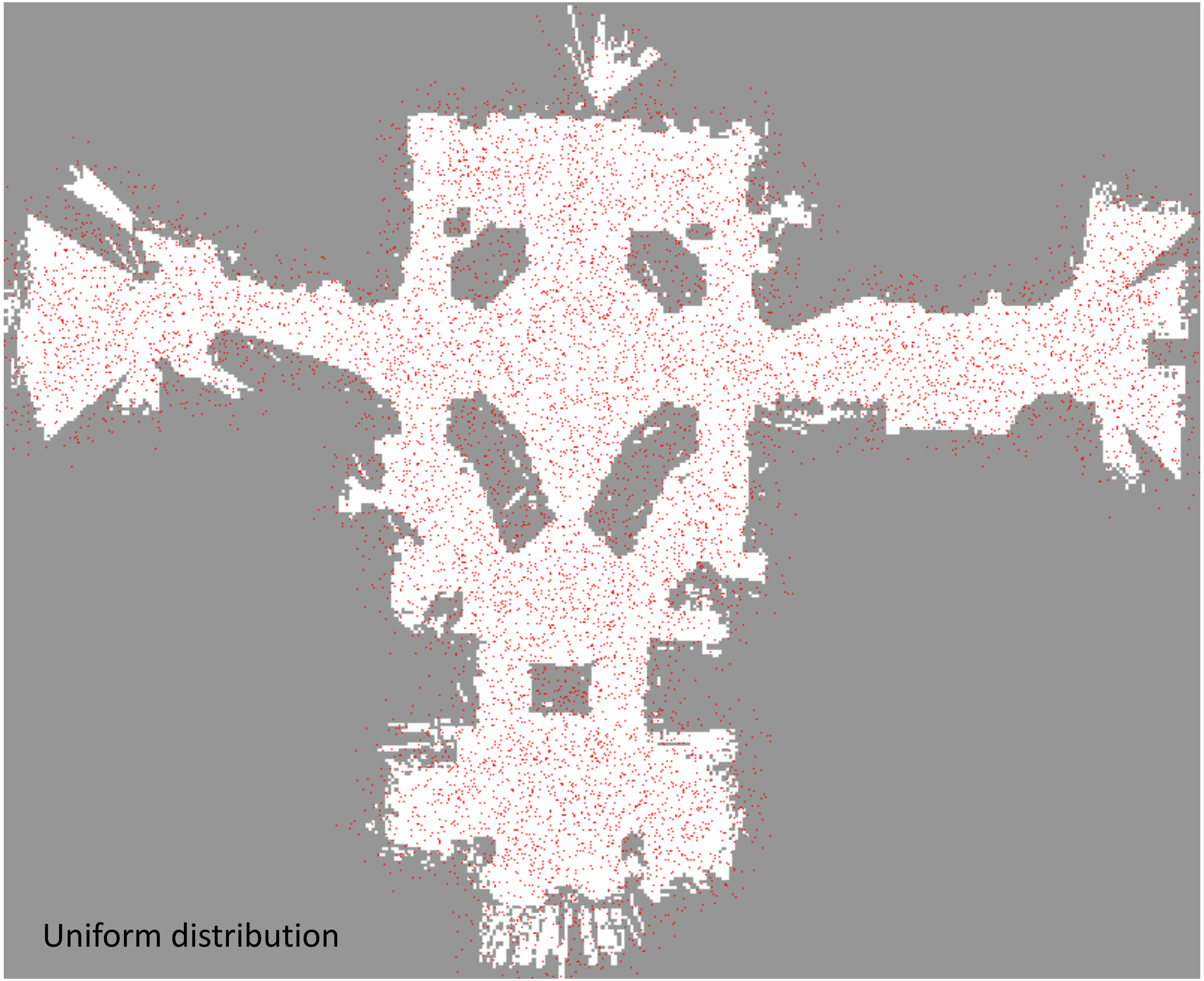


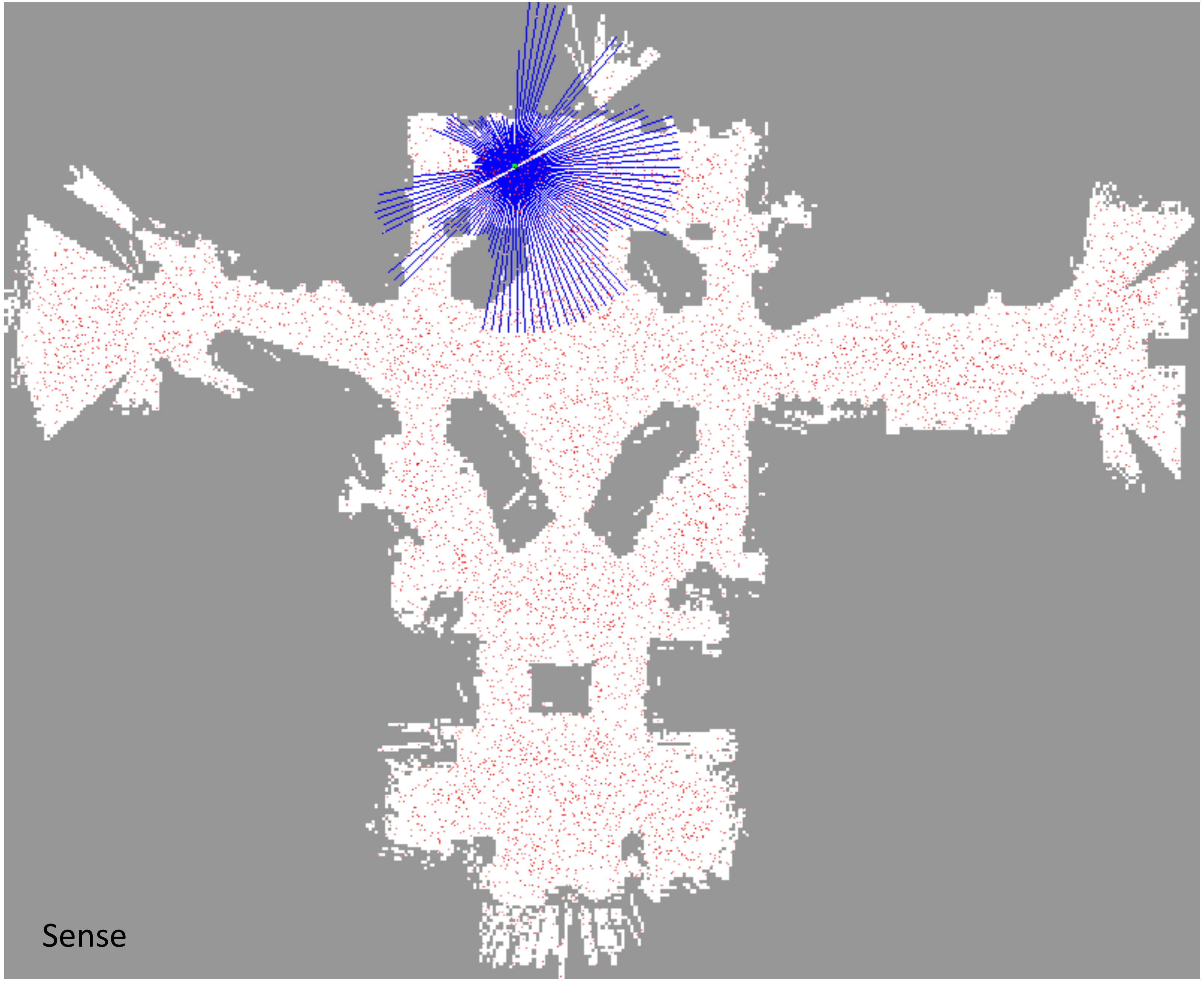
2. Measurement Phase

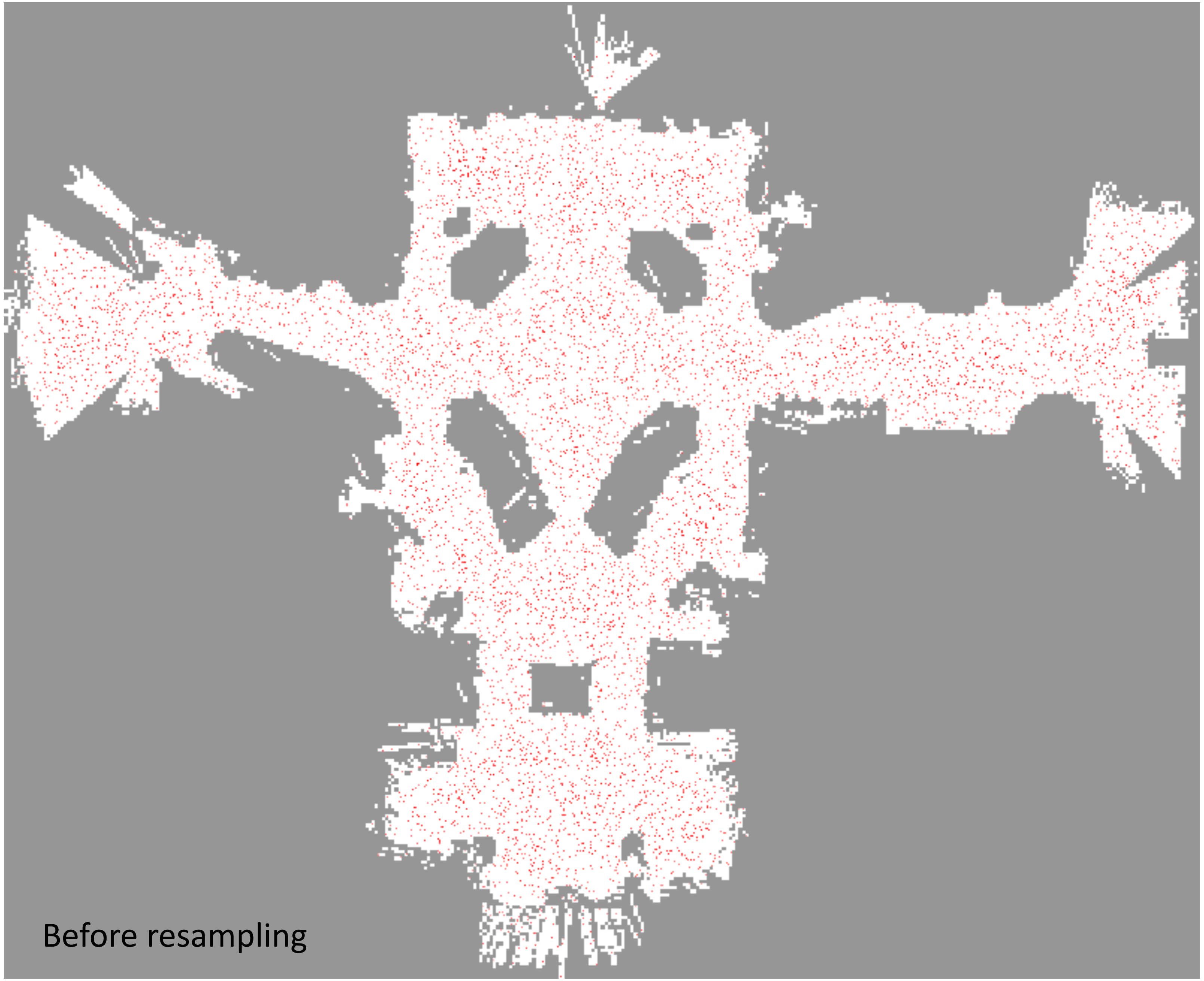


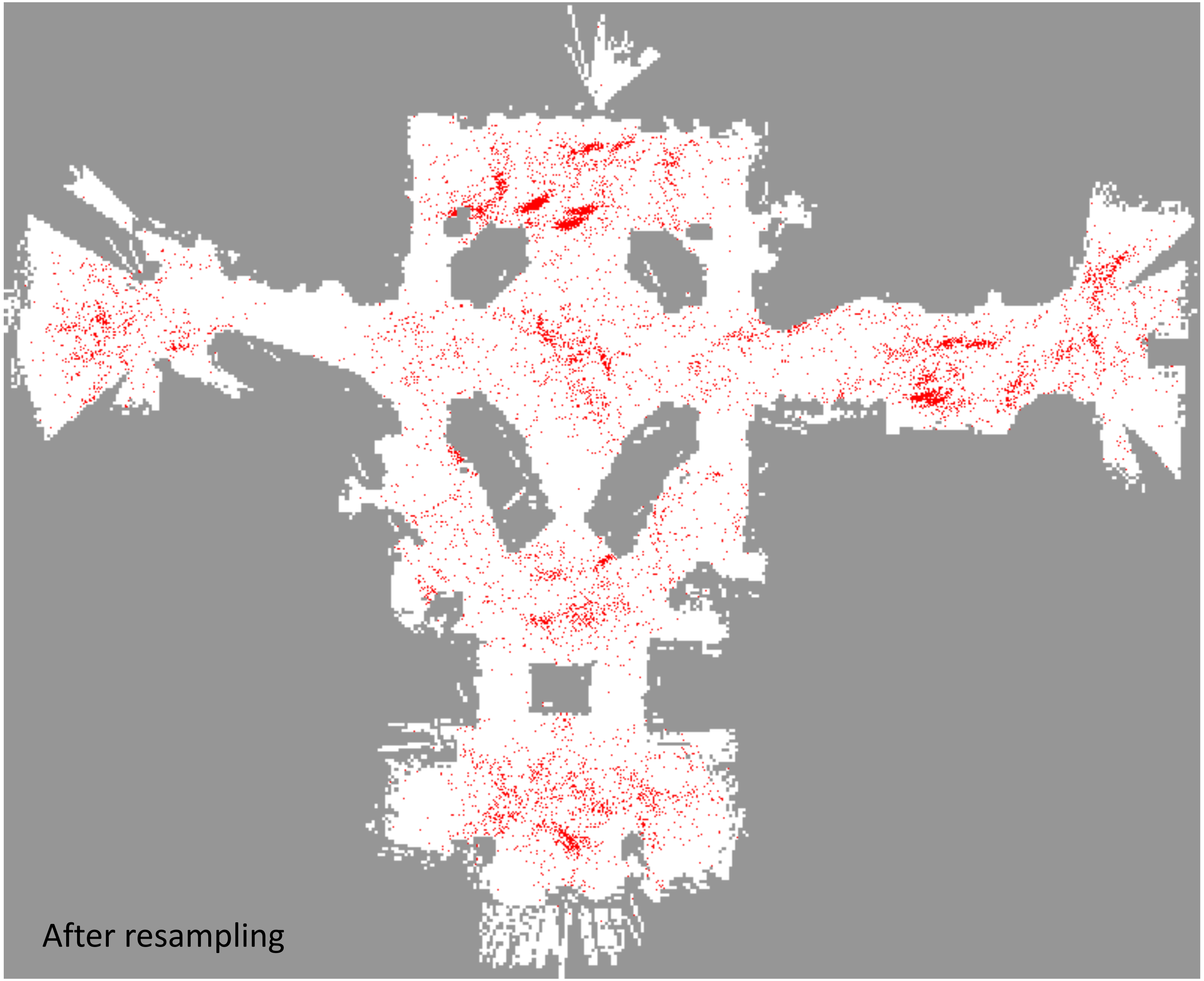
3. Resampling Step

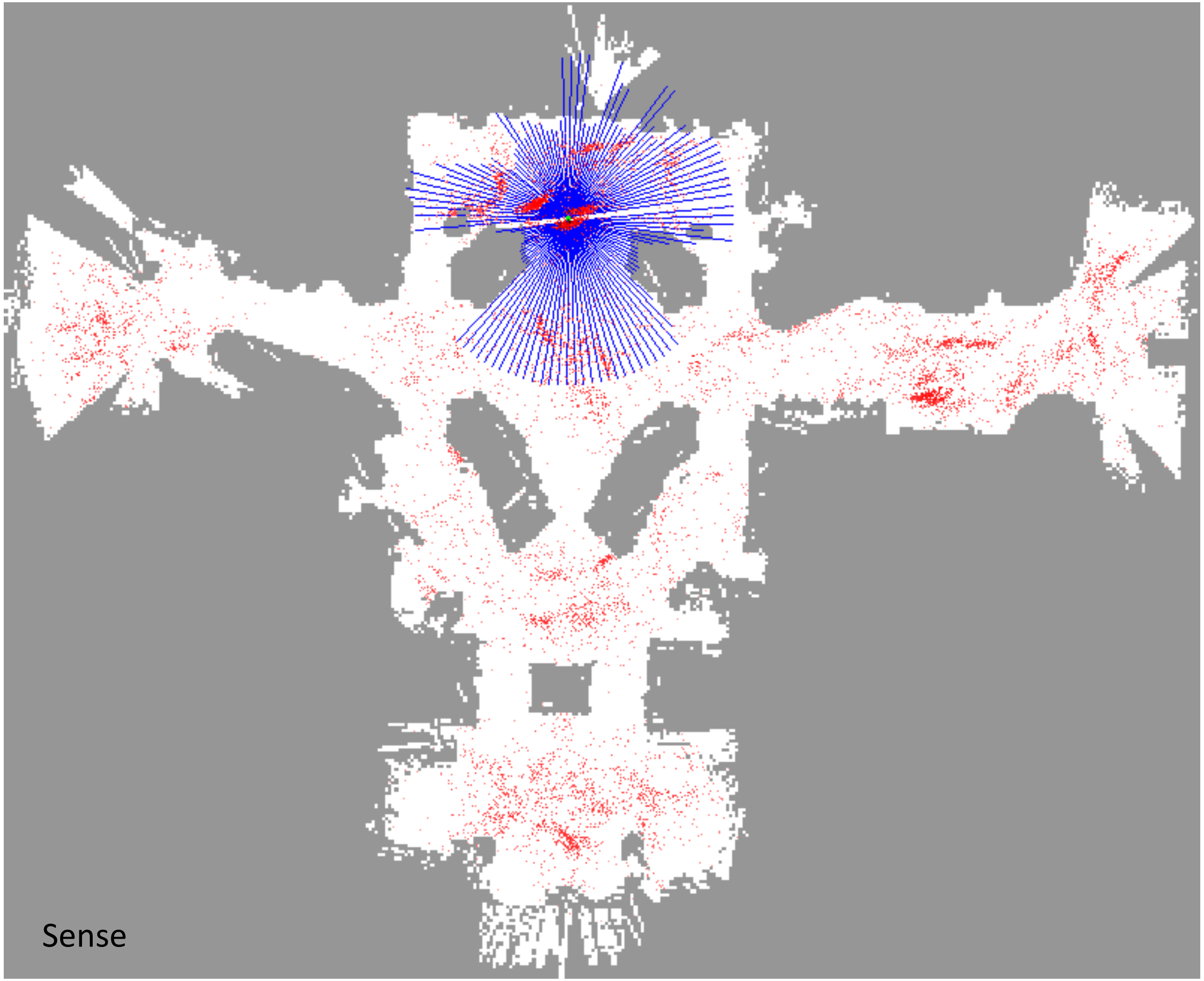


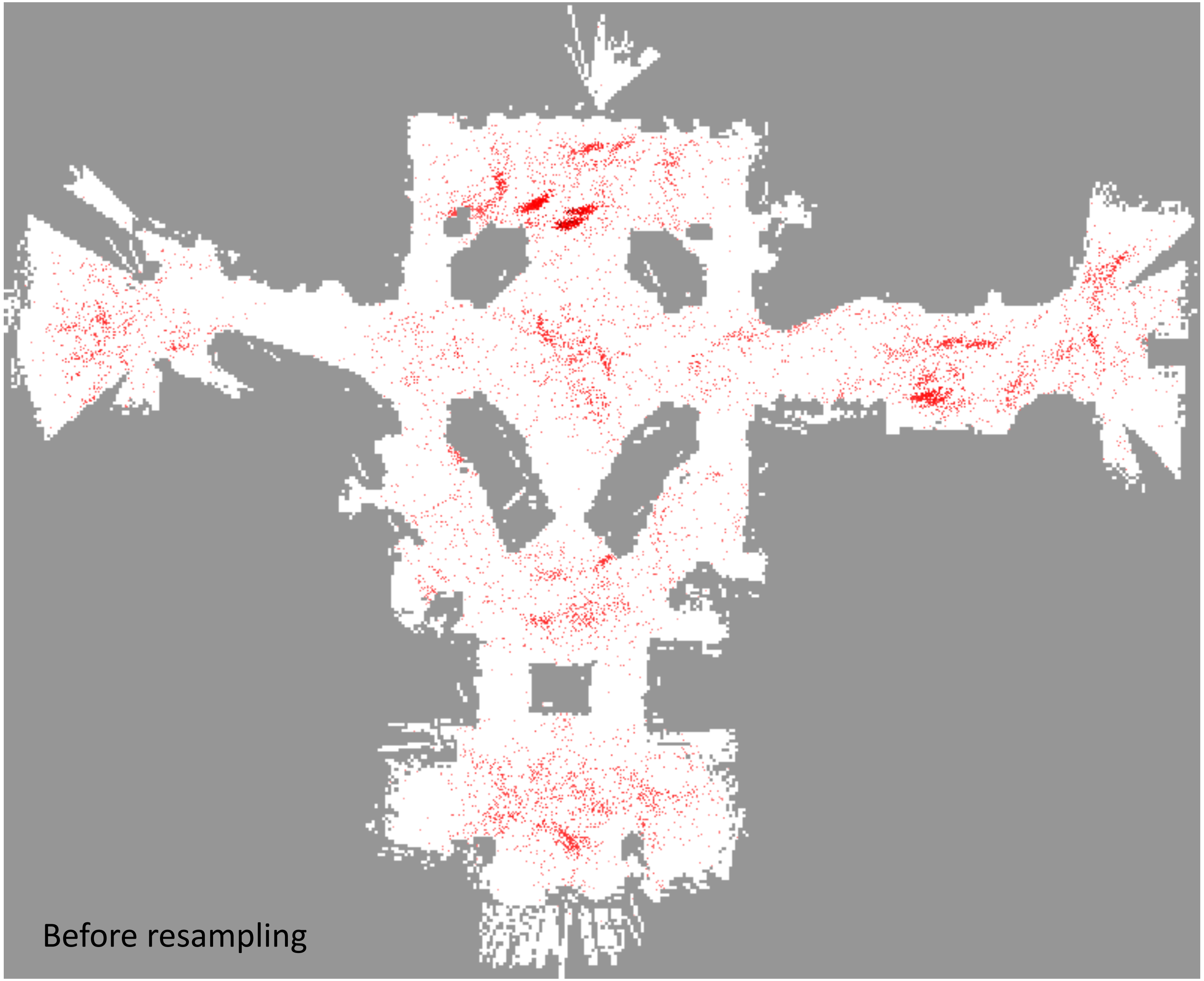

Uniform distribution

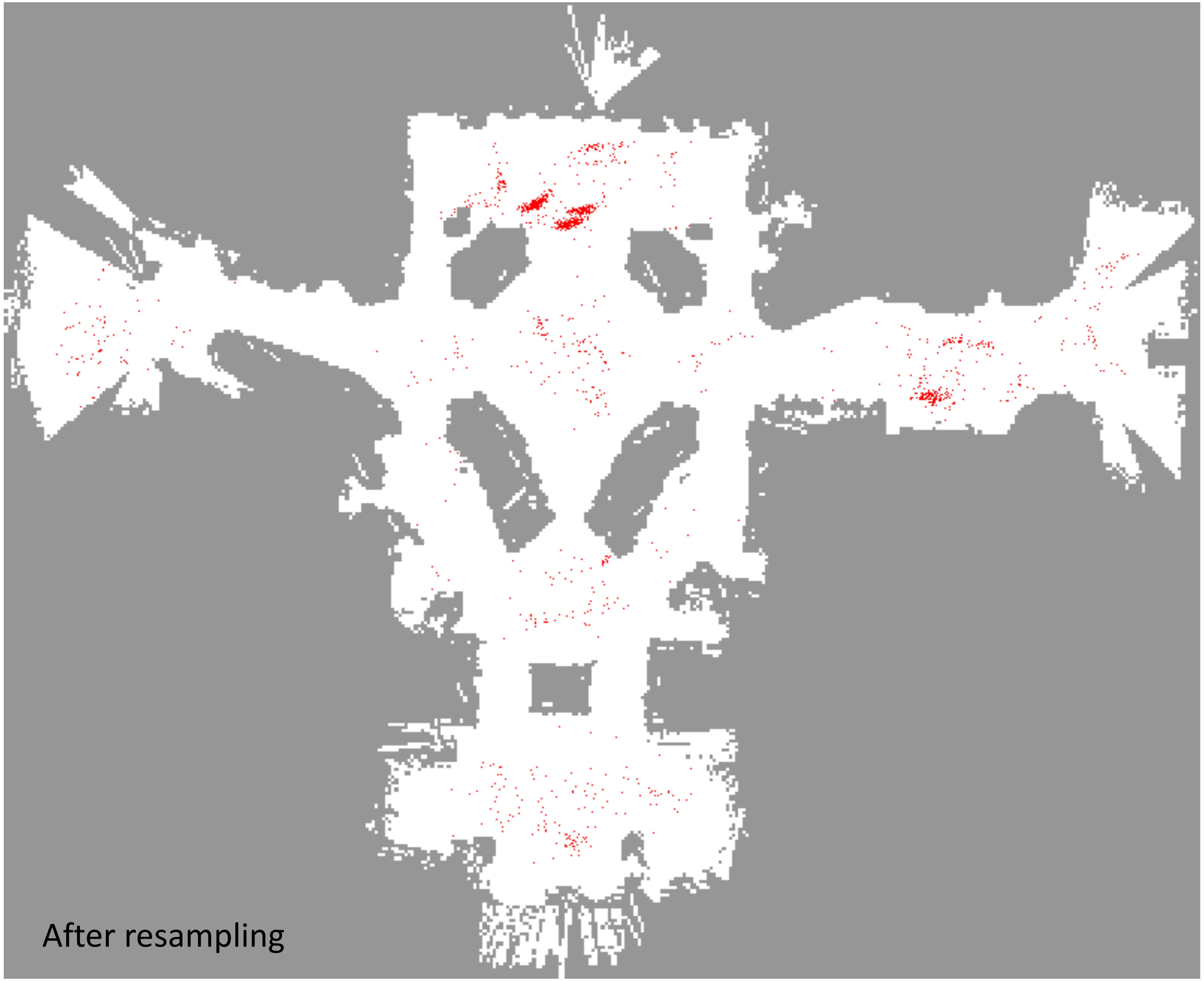

Sense


Before resampling

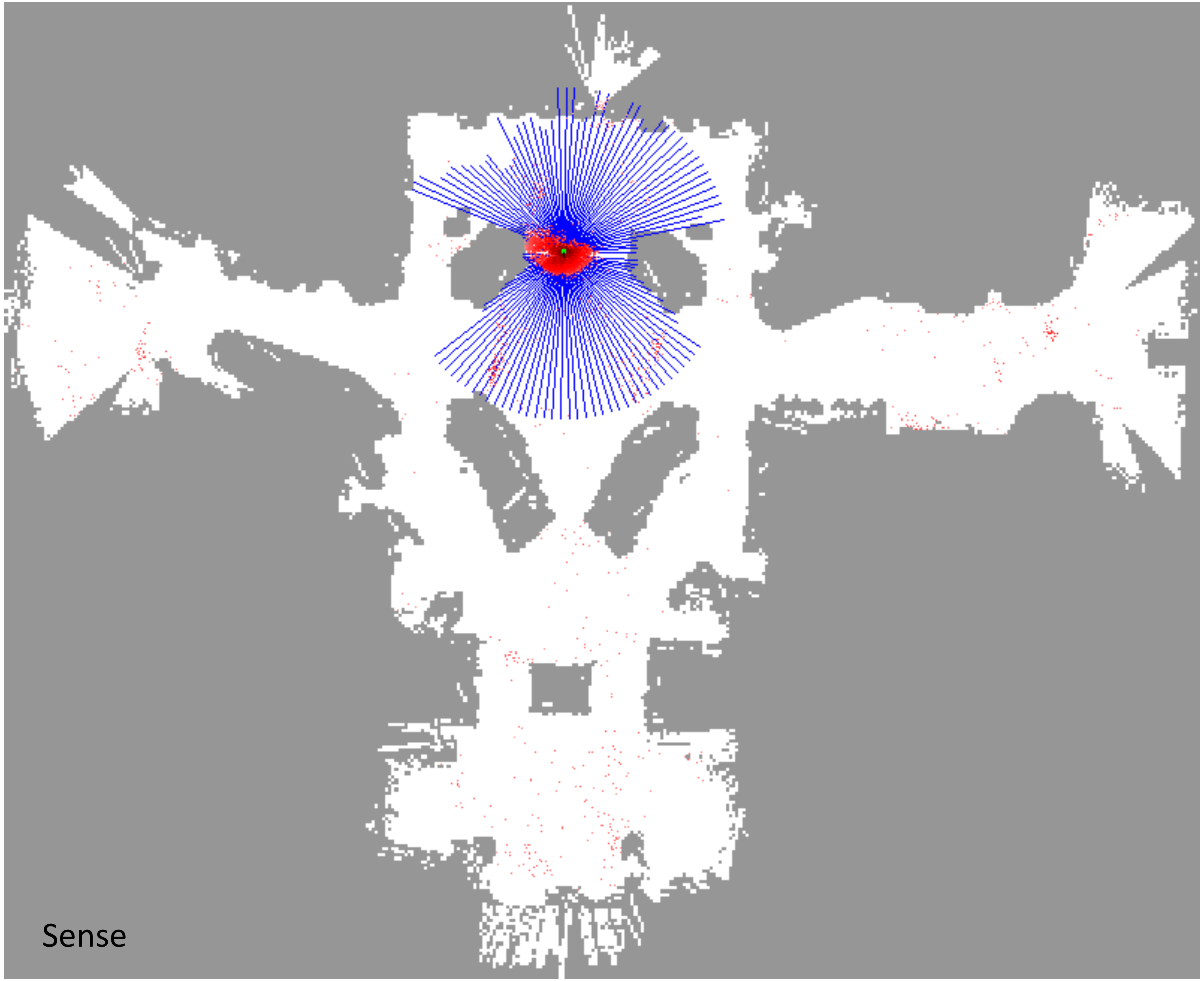

After resampling


Sense

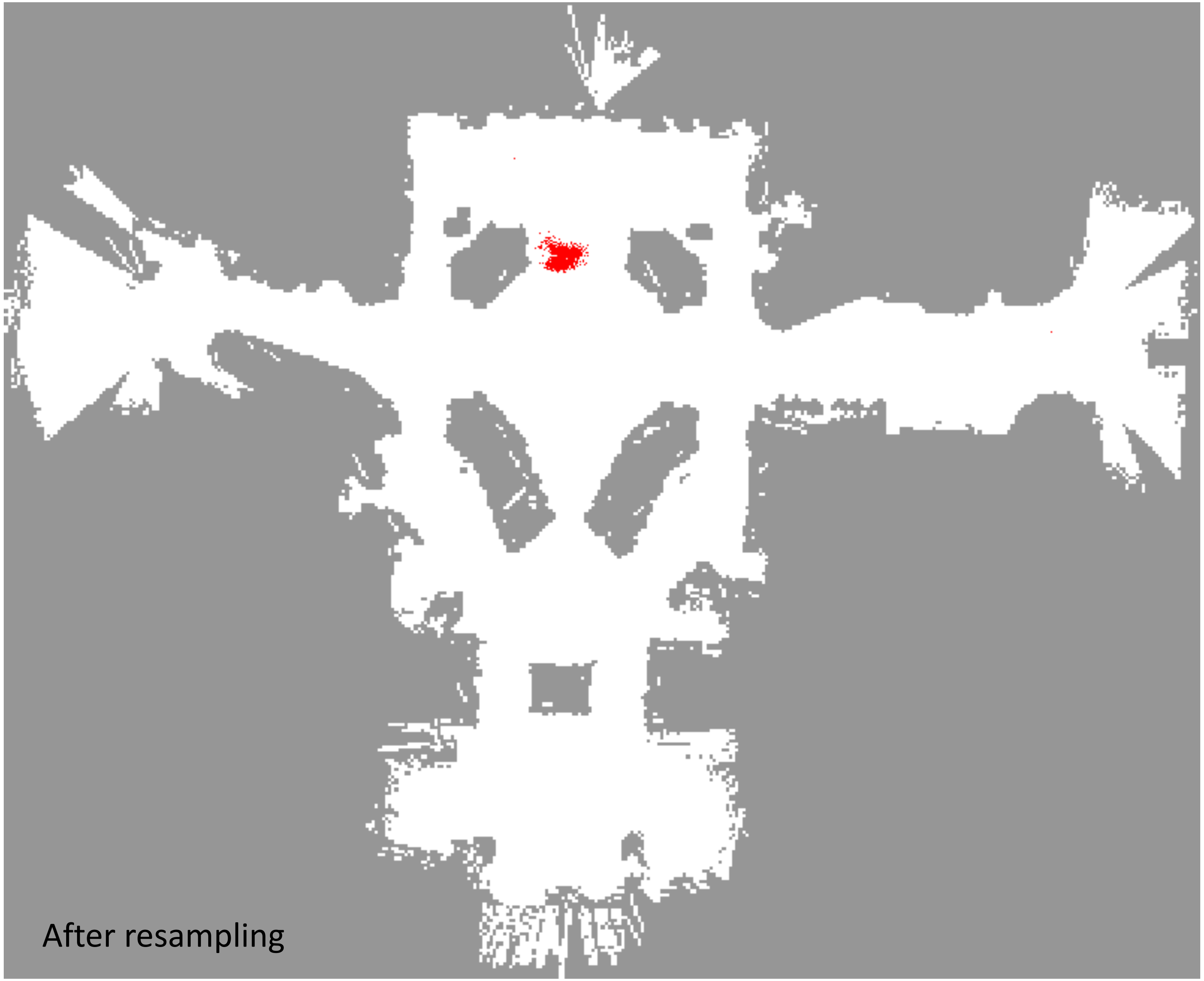

Before resampling

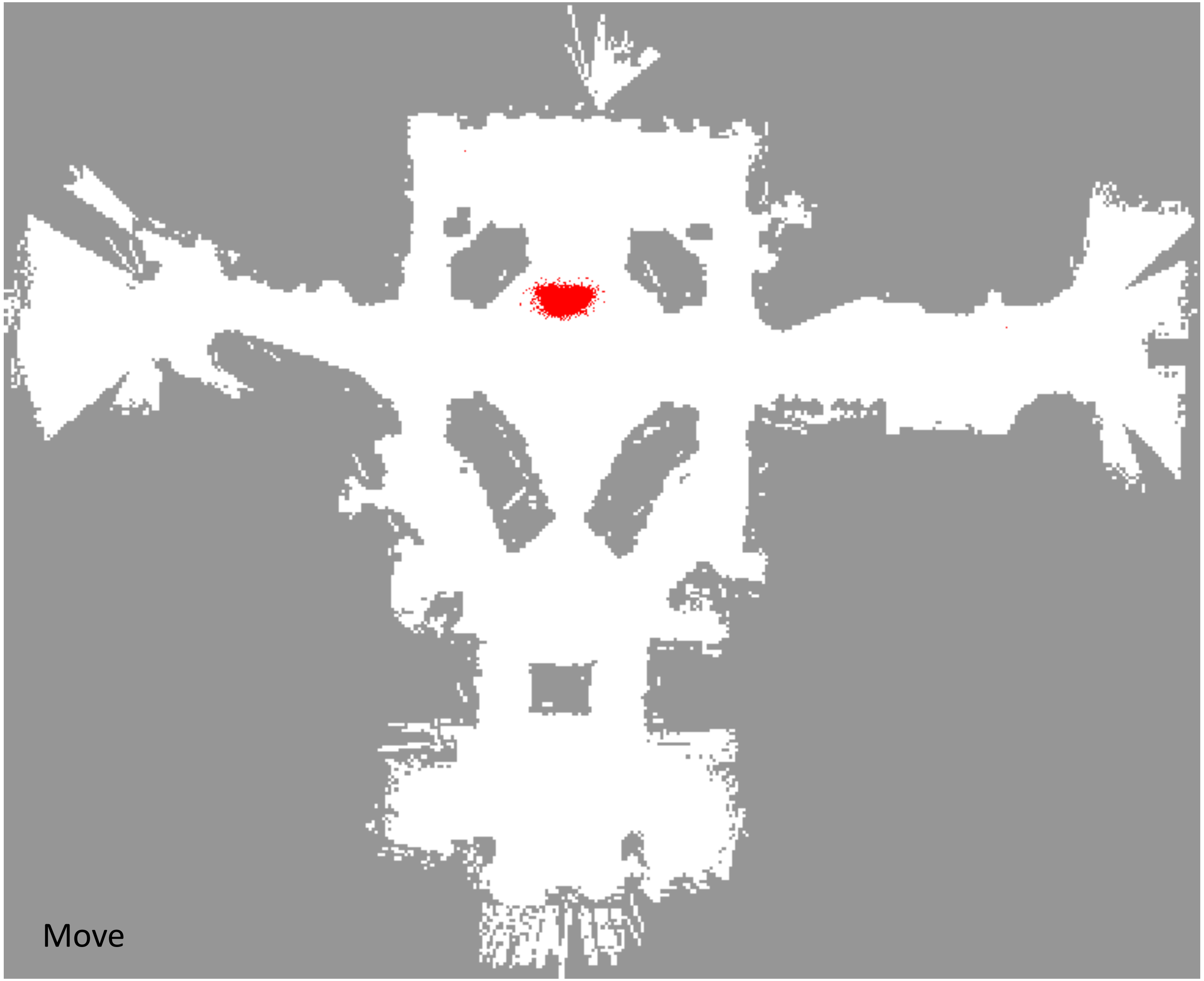

After resampling

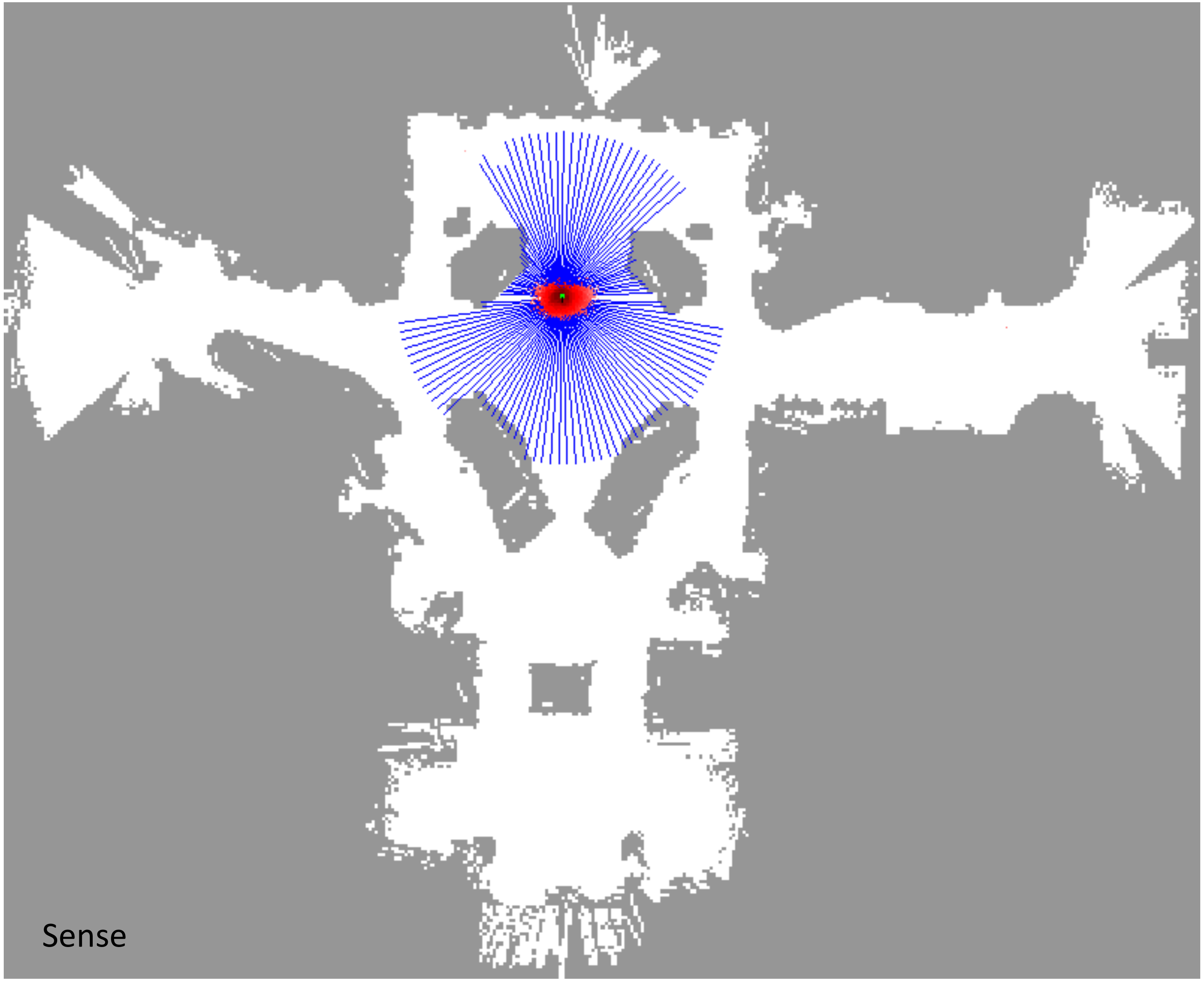

Move

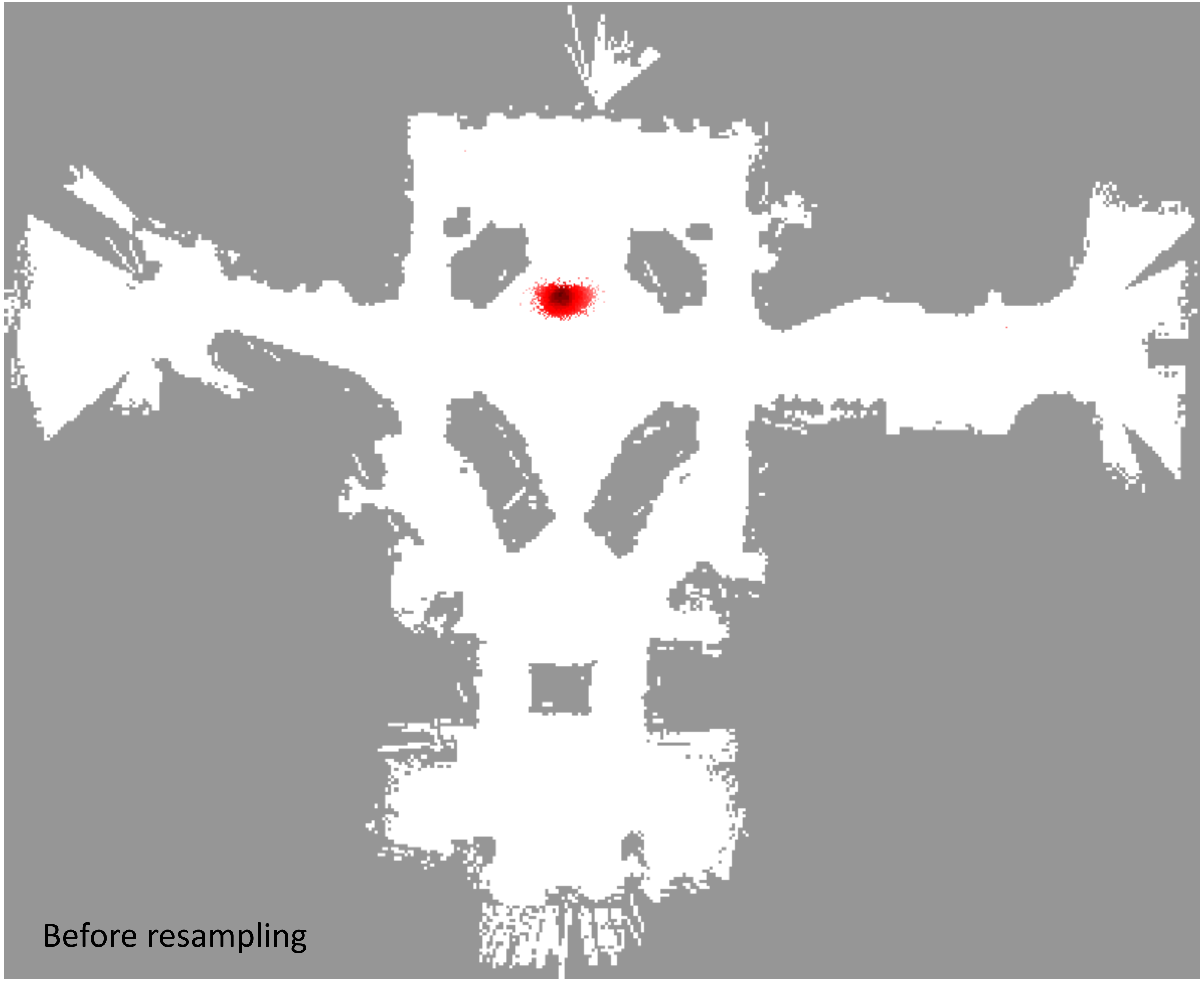

Sense

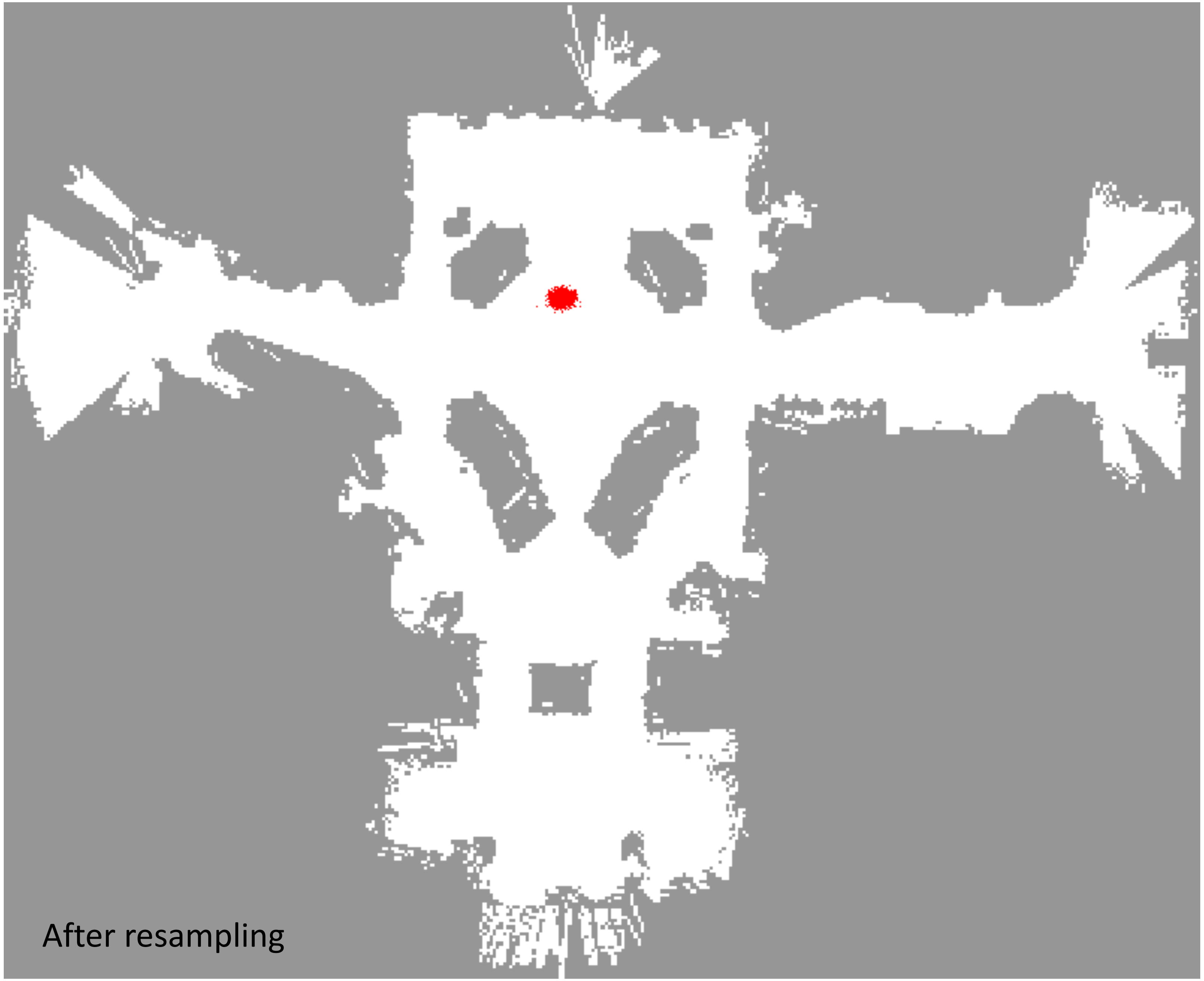

Before resampling

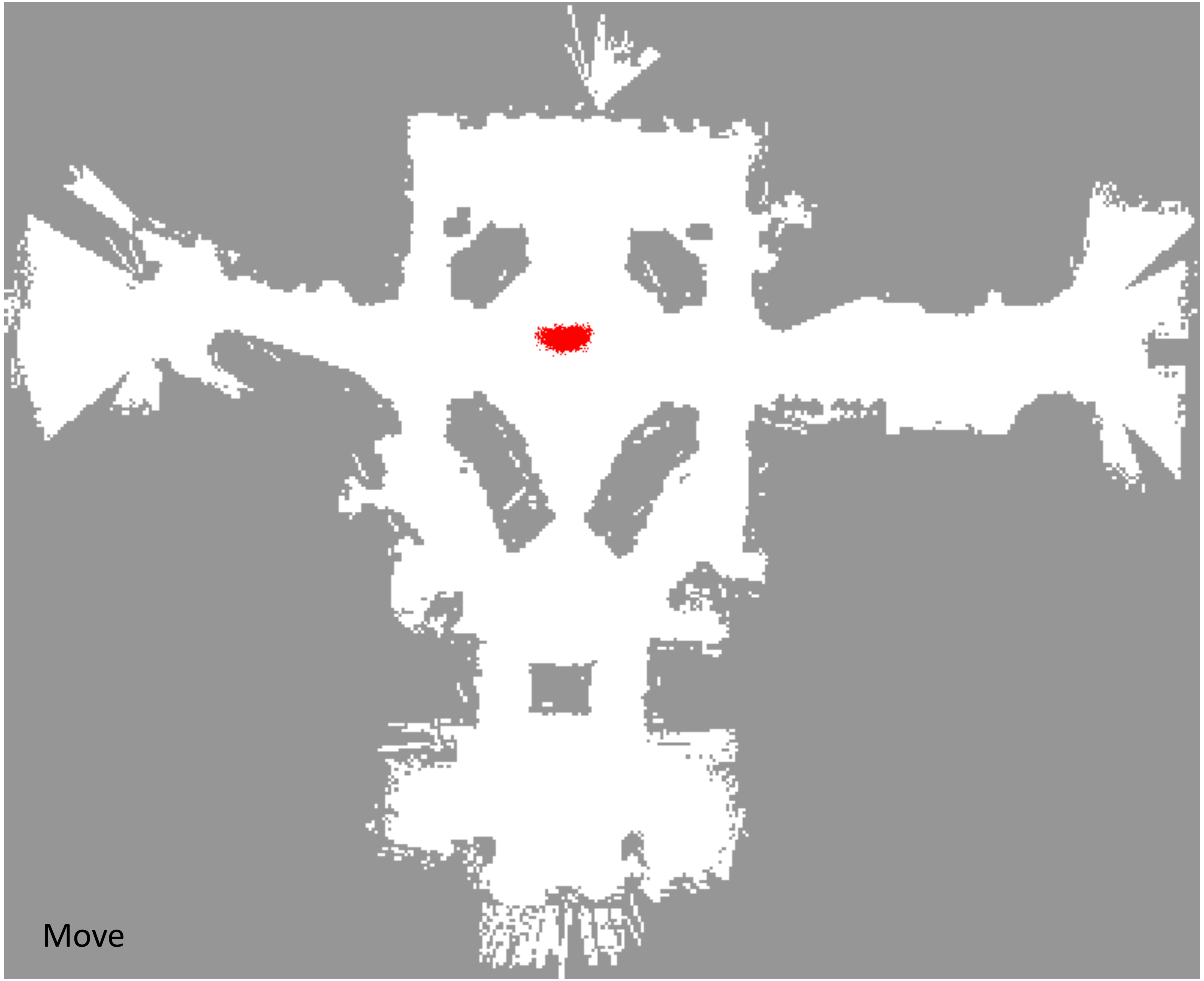

After resampling


Move

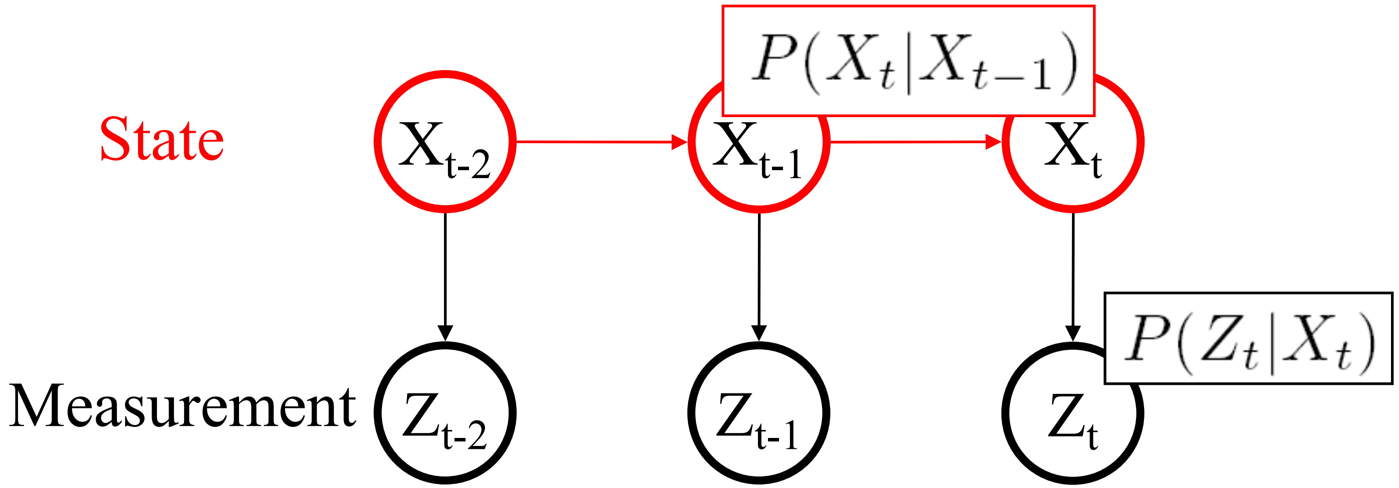

Sense


Before resampling


After resampling

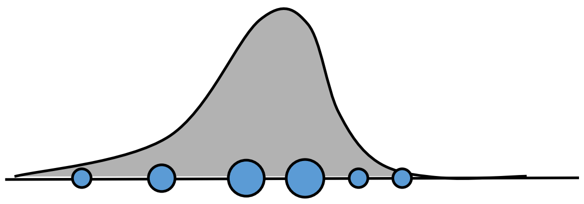

Move

Particle Filter Tracking



Monte Carlo Approximation of Posterior:

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



Bayes Filter and Particle Filter

Motion Model

Recursive Bayes Filter Equation:

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

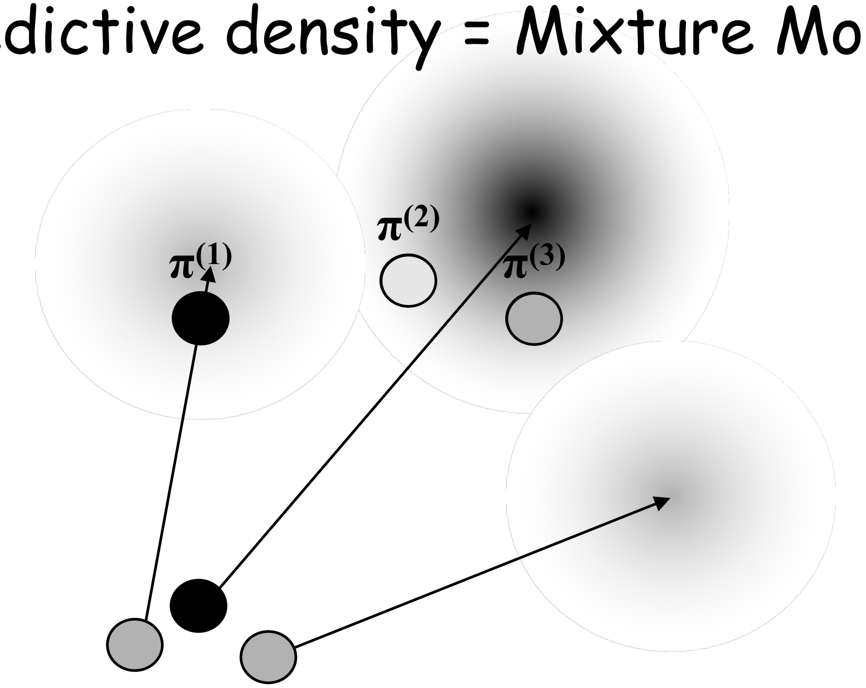
Predictive Density

Monte Carlo Approximation:

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

Particle Filter

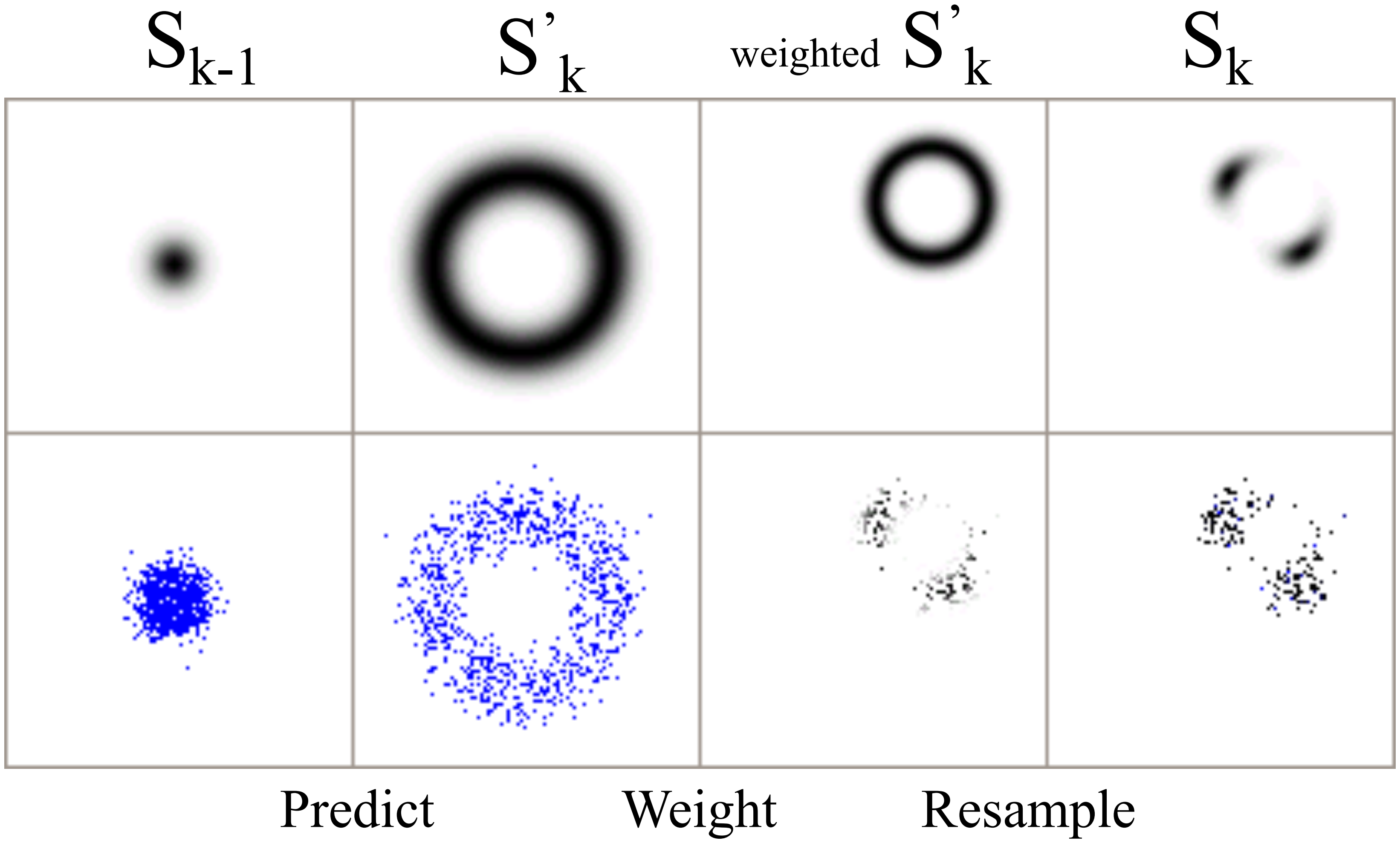
Empirical predictive density = Mixture Model



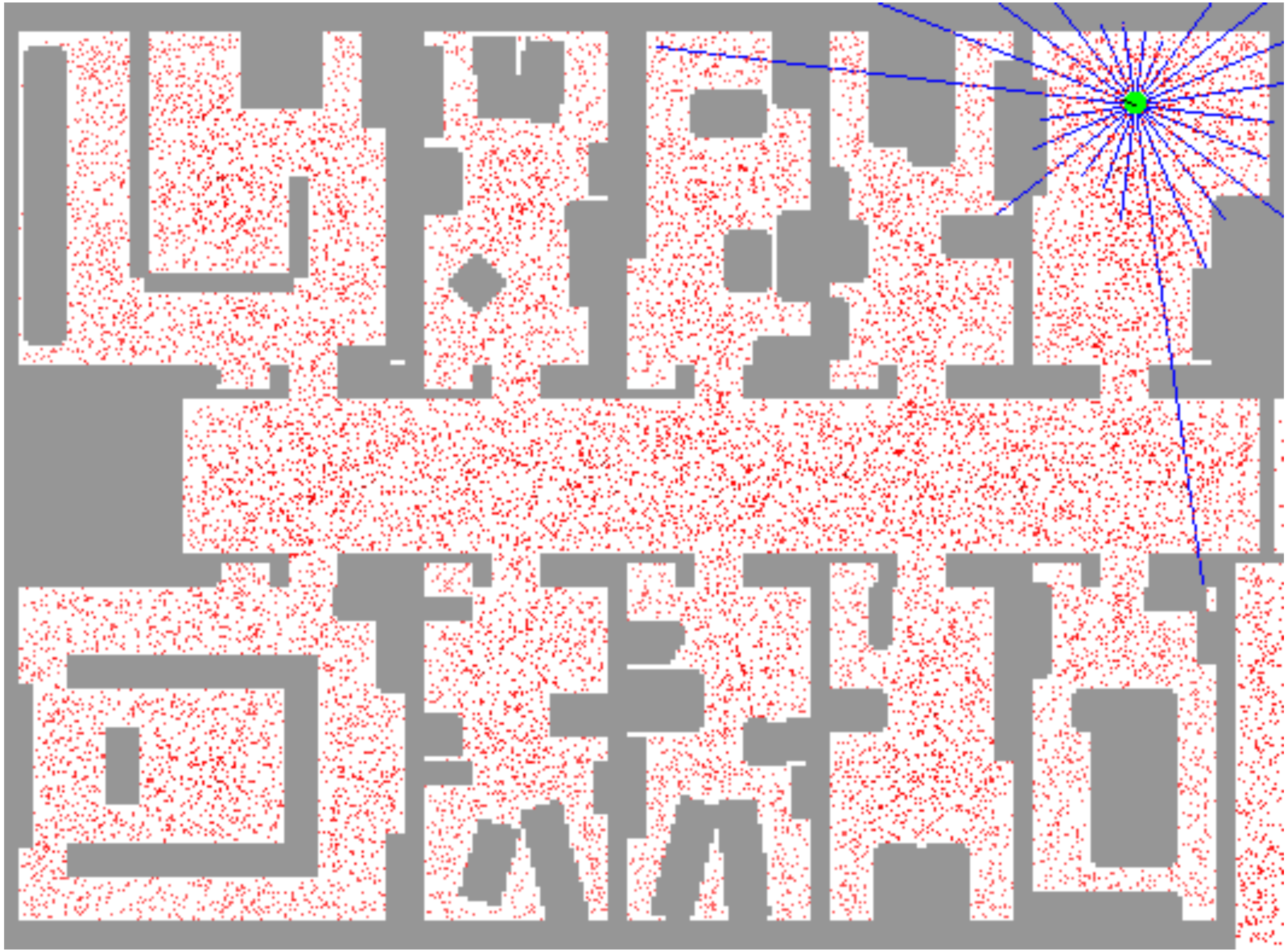
$$\pi_t^{(s)} = P(Z_t | X_t^{(s)})$$

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

Monte Carlo Localization



Monte Carlo Localization (99)



10. Connection with Elimination Algorithm*

- In class, if time remains…

Summary

- Continuous Densities
- Gaussian Densities
- Bayes Nets & Mixture Models
- Cont. Measurement Models
- Cont. Motion Models
- Simulating Cont. Bayes Nets
- Sampling as Approximation
- Importance Sampling
- Particle Filters and Monte Carlo Localization
- Monte Carlo & Elimination