Building Accurate Maps using Rao-Blackwellized Particle Filters

Cyrill Stachniss

University of Freiburg, Germany



ALBERT-LUDWIGS-
UNIVERSITÄT FREIBURG

Thanks and partial slide courtesy of Mike Montemerlo and Dirk Haehnel

Research Lab for Autonomous Intelligent Systems

- Headed by Prof. Dr. Wolfram Burgard
- 1 academic advisor & 1 post-doc
- 14 Ph.D. students

Key Projects

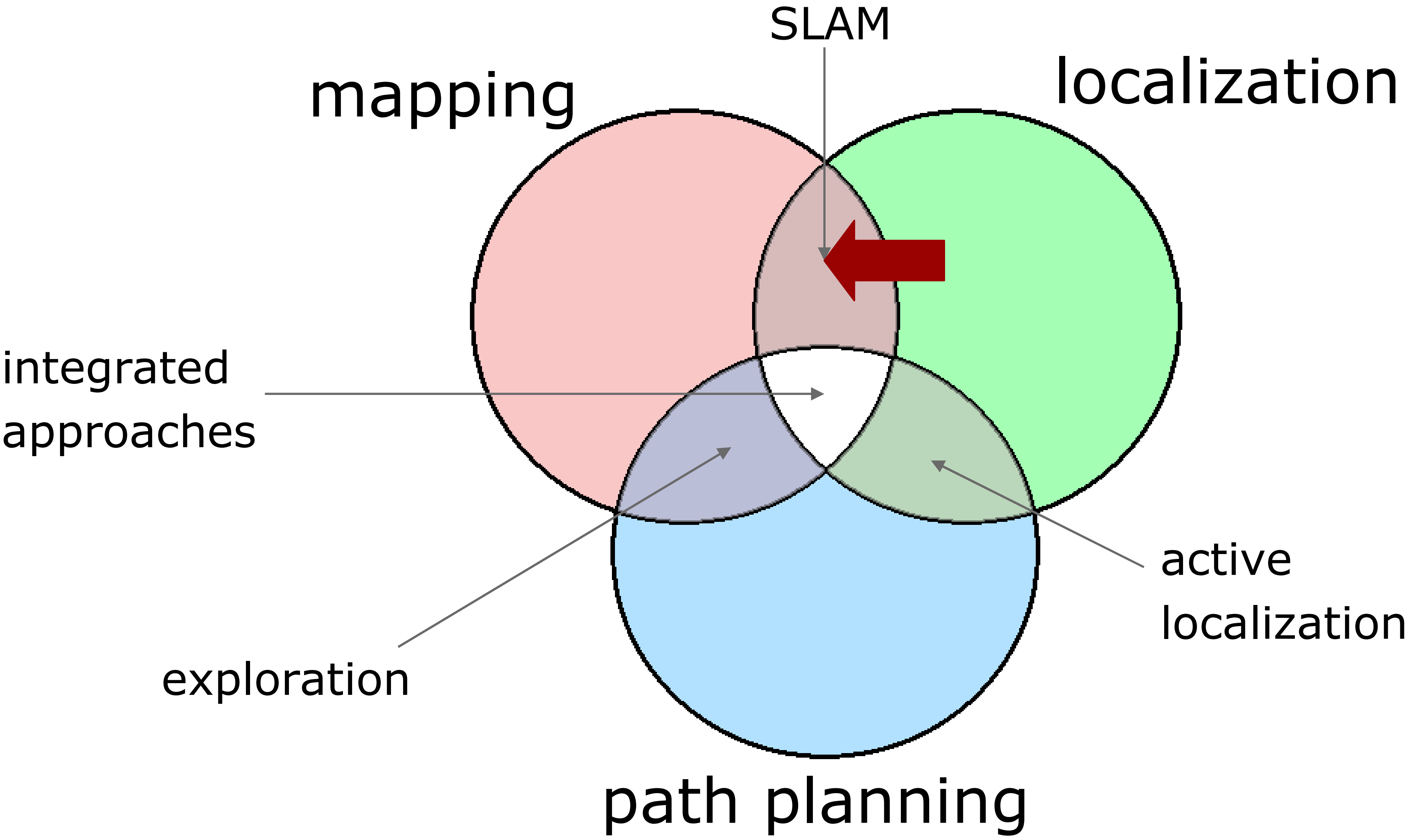
- 1 SFB/TR-8
- 3 European projects
- 1 DFG graduate school
- 1 BMBF project
- 3 projects funded by industry

Fields of Research

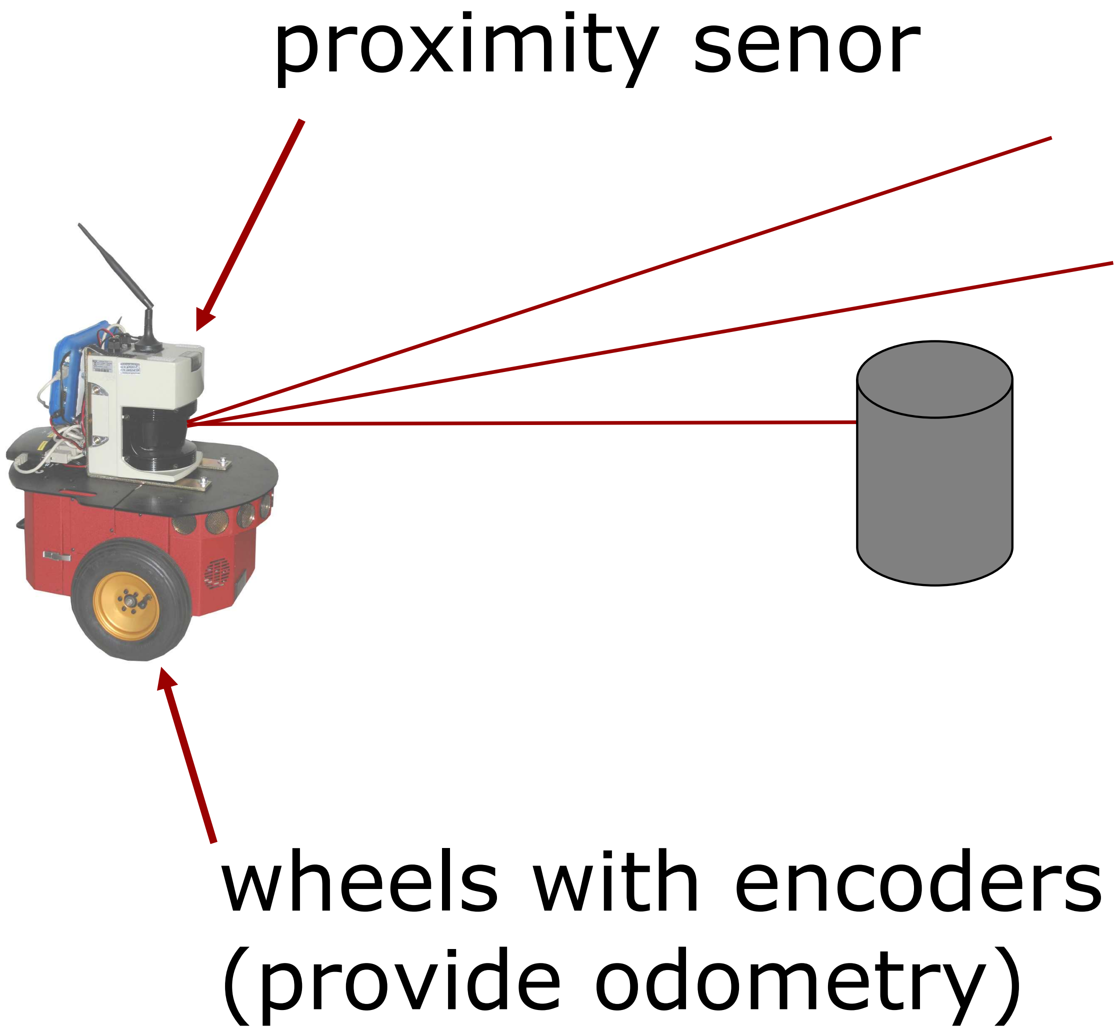
- Mobile robots
- State and model estimation
- Adaptive techniques and learning
- Multi-robot coordination
- Decision-theoretic approaches
- Scene understanding
- Manipulation
- Autonomous cars
- Humanoid robots
- Flying vehicles
- …

Probabilistic Robotics

What is this Talk About?

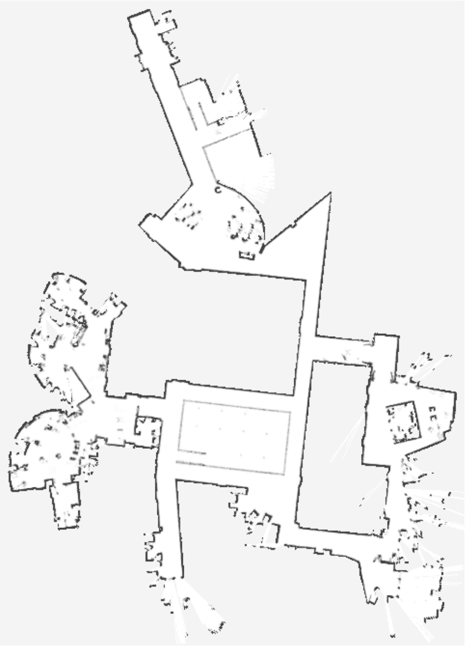


A Typical Robot...



Maps are Important for Robots

- Maps model the environment of the robot
- Localization is impossible without a map ("Where am I?")
- Efficient motion planning requires maps ("How to reach a target location?")
- Reasoning about the state of the world
- …



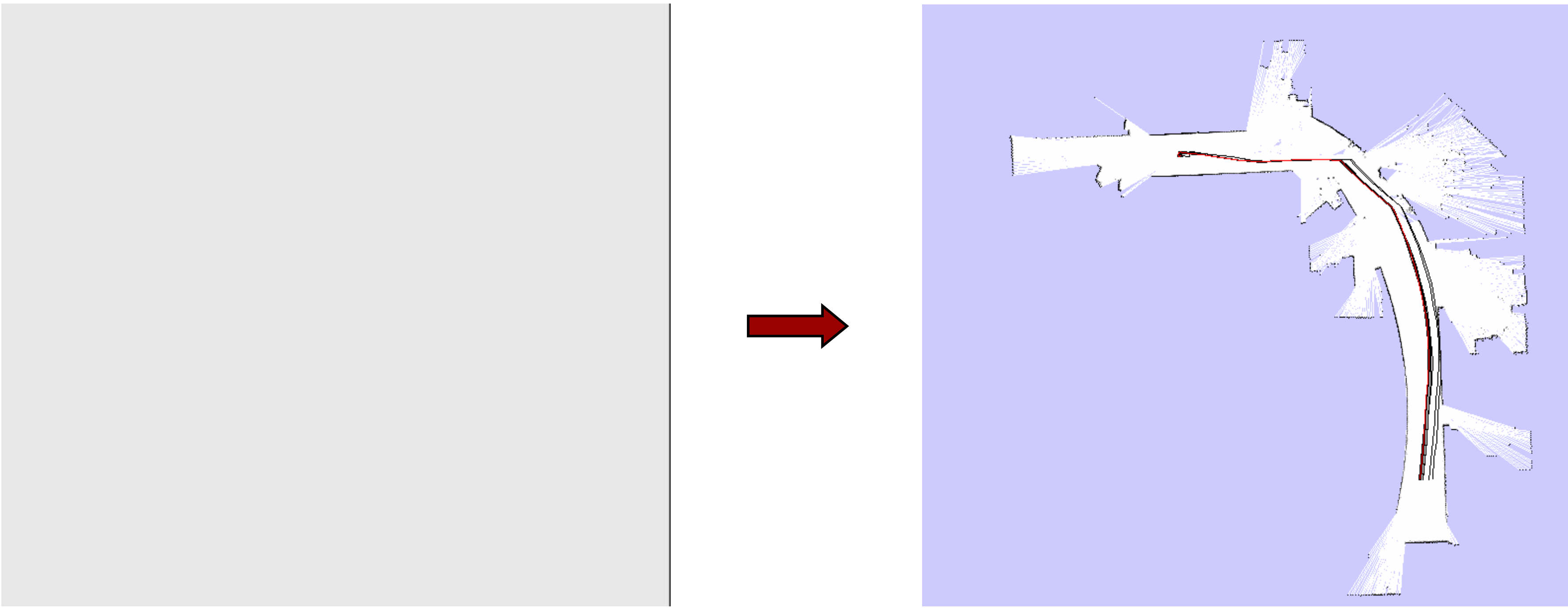
Maps are important for efficiently solving standard robotic problems

What is "SLAM" ?

- Estimate the pose and the map of a mobile robot at the same time

$$p(x, m \mid z, u)$$

poses map observations & movements



Courtesy of Dirk Haehnel

Mapping using Raw Odometry

- Why is SLAM hard? Chicken-or-egg problem:
  - a map is needed to localize the robot and
  - a pose estimate is needed to build a map

Courtesy of Dirk Haehnel

Particle Filters

Who knows how a particle filter works

?

Explain Particle Filters

Skip Explanation

Brief Introduction to Particle Filters

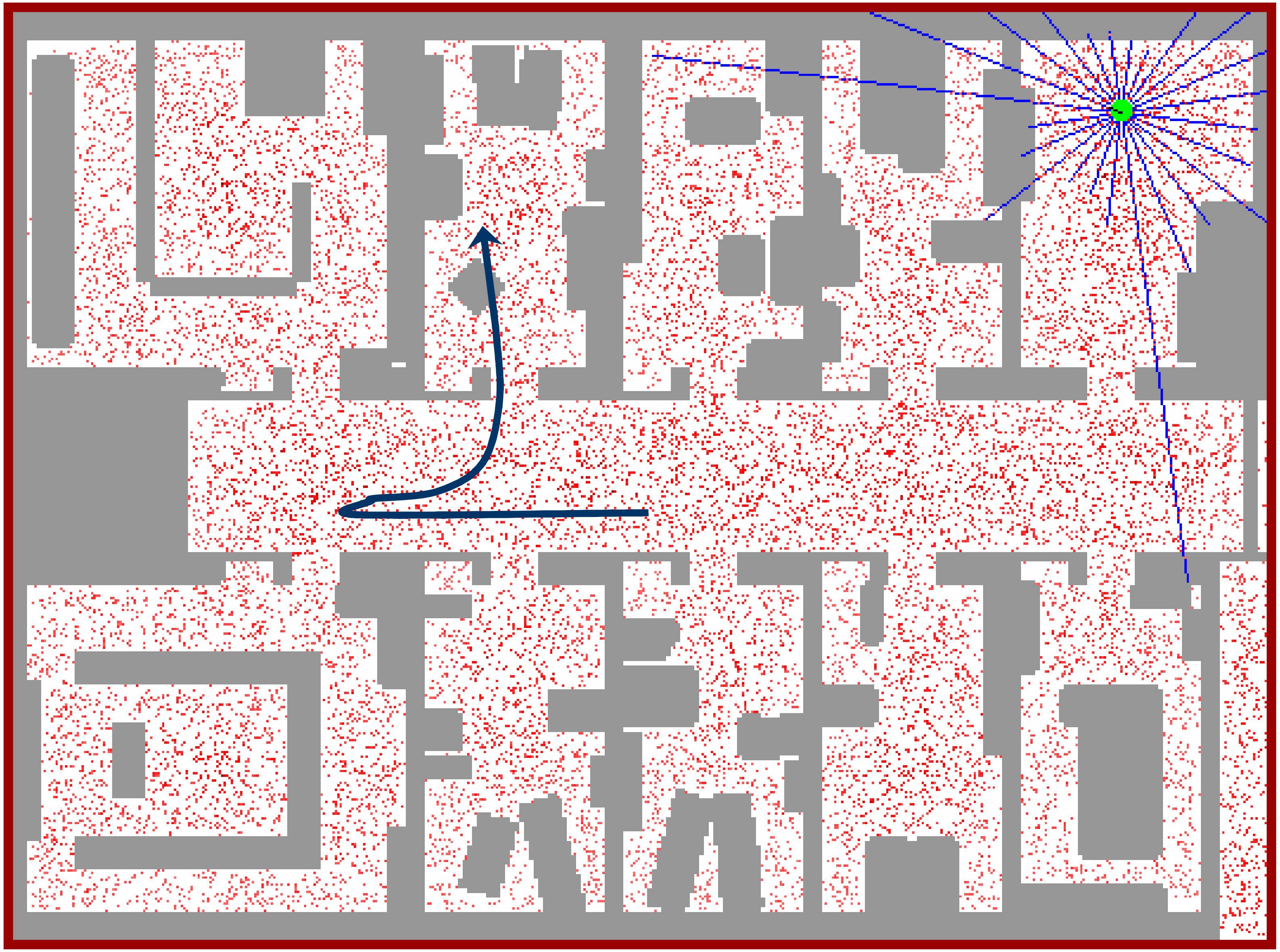
What is a particle filter?

- It is a Bayes filter
- Particle filters are a way to efficiently represent non-Gaussian distribution

Basic principle

- Set of state hypotheses (“particles”)
- Informally speaking: “survival-of-the-fittest”

Sample-based Localization (sonar)



Video courtesy of Dieter Fox

$p(x \mid m, z, u) =$ where is the robot?

Sample-based Posteriors

- Set of weighted samples

$$S = \left\{ \left\langle s^{(i)}, w^{(i)} \right\rangle \mid i = 1, \dots, N \right\}$$

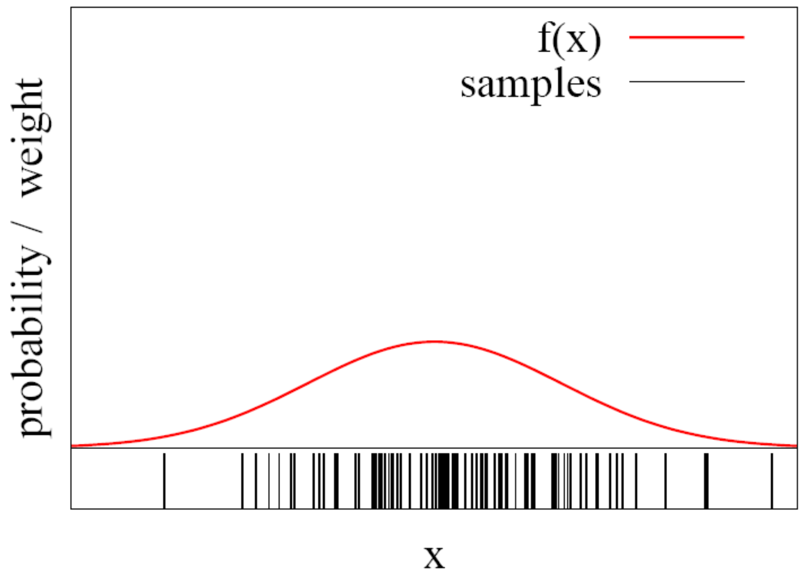
State hypothesis — Importance weight

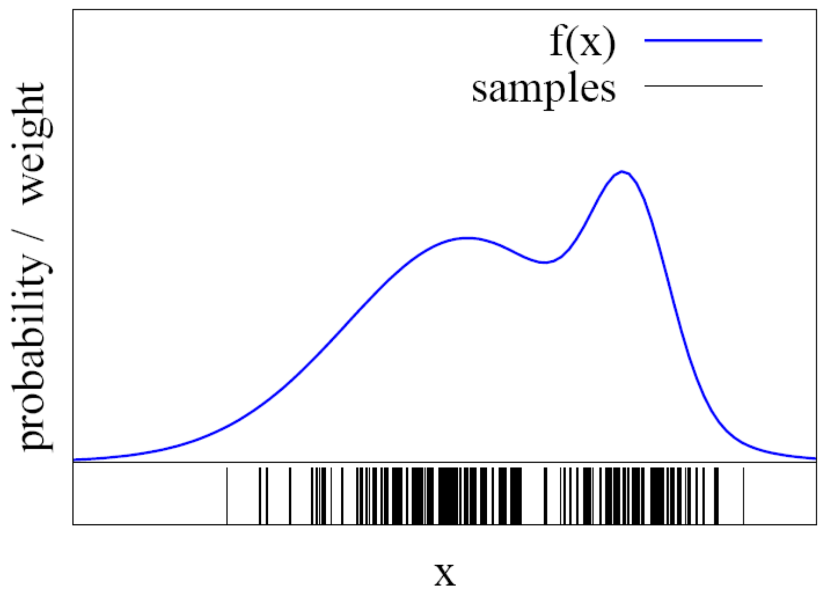
- The samples represent the posterior

$$p(x) = \sum_{i=1}^{N} w^{(i)} \cdot \delta_{s^{(i)}}(x)$$

Posterior Approximation

- Particle sets can be used to approximate functions

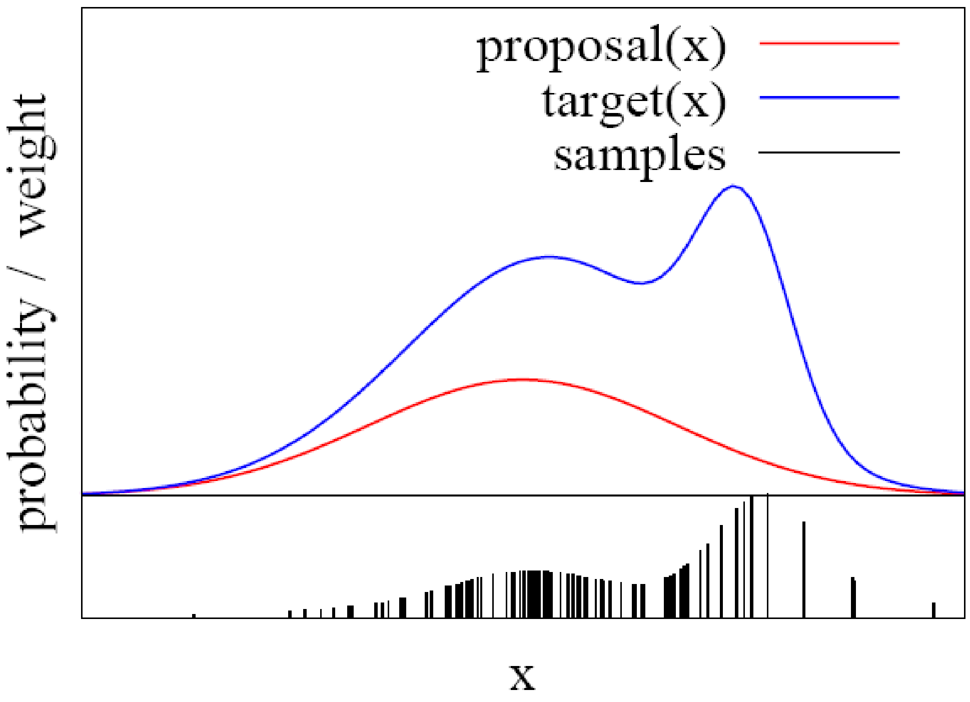




- The more particles fall into an interval, the higher the probability of that interval
- We can easily draw samples from a Gaussian but not from general distributions

Importance Sampling Principle

- We can even use a different distribution g to generate samples from f
- By introducing an importance weight w, we can account for the "differences between g and f"
- $w = f / g$
- f is called target
- g is called proposal



From Sampling to a Particle Filter

- Set of samples describes the posterior
- Updates are based on actions and observations

Three sequential steps:

1. Sampling from the proposal distribution
   (Bayes filter: prediction step)

2. Compute the particle weight (importance sampling)
   (Bayes filter: correction step)

3. Resampling

Monte-Carlo Localization (MCL)

- For each **motion** Δ do:
  - **Sampling**: Generate from each sample in a new sample according to the motion model

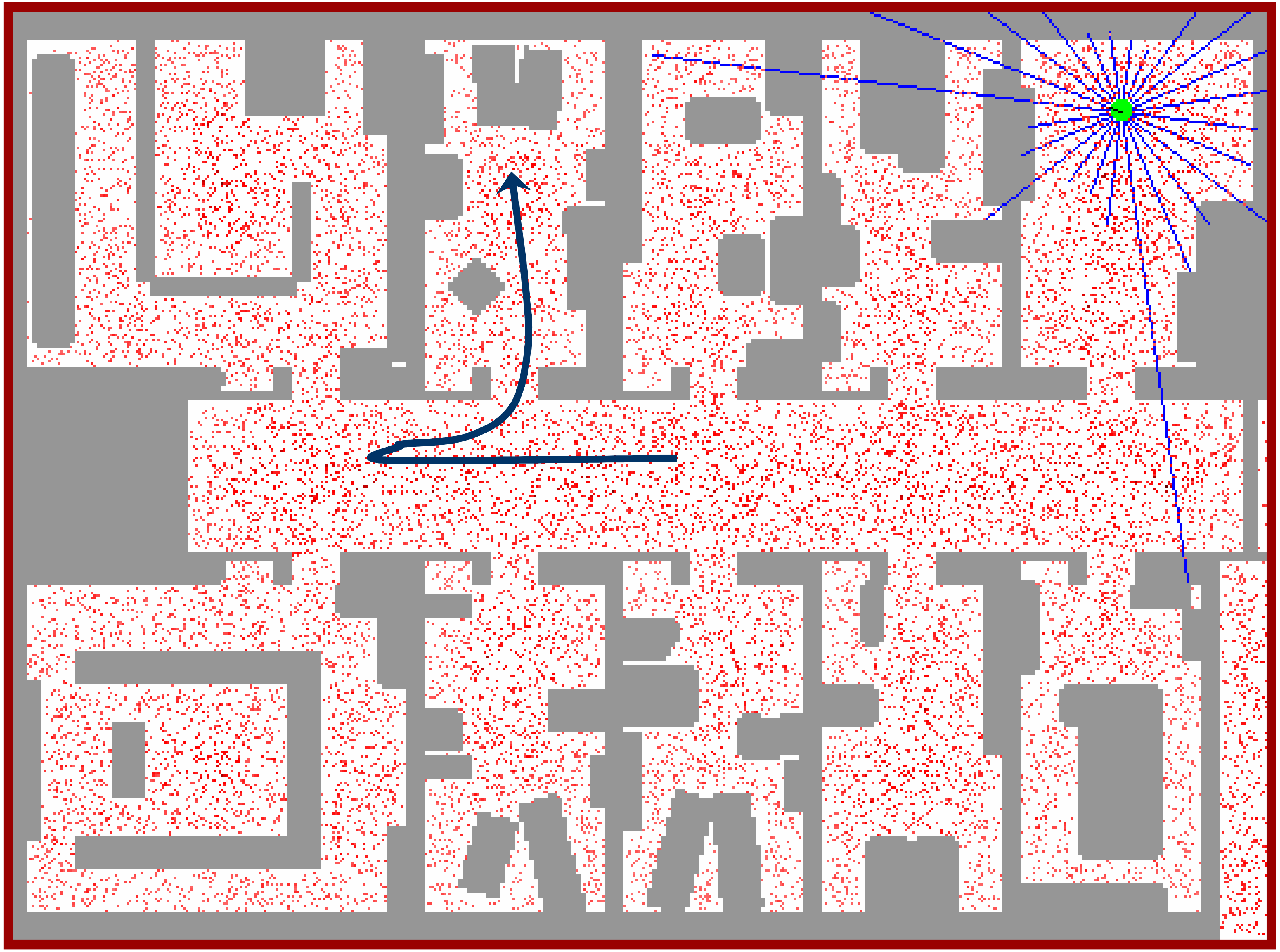
$$x^{(i)} \leftarrow x^{(i)} + \Delta'$$

- For each **observation** do:
  - **Weight** the samples with the observation likelihood

$$w^{(i)} \leftarrow p(z \mid m, x^{(i)})$$

  - **Resampling**

Sample-based Localization (sonar)



Courtesy of Dieter Fox

SLAM with Particle Filters

- Particle filters have successfully been applied to localization, can we use them to solve the SLAM problem?

- Posterior over poses x and maps m

$$p(x \mid m, z, u) \Rightarrow p(x, m \mid z, u)$$

(localization) (SLAM)

Observations:

- The map depends on the poses of the robot during data acquisition
- If the poses are known, mapping is easy

Rao-Blackwellization

poses map observations & movements

$$p(x_{1:t}, m \mid z_{1:t}, u_{0:t-1}) =$$

$$p(x_{1:t} \mid z_{1:t}, u_{0:t-1}) \cdot p(m \mid x_{1:t}, z_{1:t})$$

Factorization first introduced by Murphy in 1999

Rao-Blackwellization

poses map observations & movements

$$p(x_{1:t}, m \mid z_{1:t}, u_{0:t-1}) = p(x_{1:t} \mid z_{1:t}, u_{0:t-1}) \cdot p(m \mid x_{1:t}, z_{1:t})$$

SLAM posterior

Robot path posterior

Mapping with known poses

Factorization first introduced by Murphy in 1999

Rao-Blackwellization

$$p(x_{1:t}, m \mid z_{1:t}, u_{0:t-1}) =$$
$$p(x_{1:t} \mid z_{1:t}, u_{0:t-1}) \cdot p(m \mid x_{1:t}, z_{1:t})$$

This is localization, use MCL

Use the pose estimate from the MCL and apply mapping with known poses

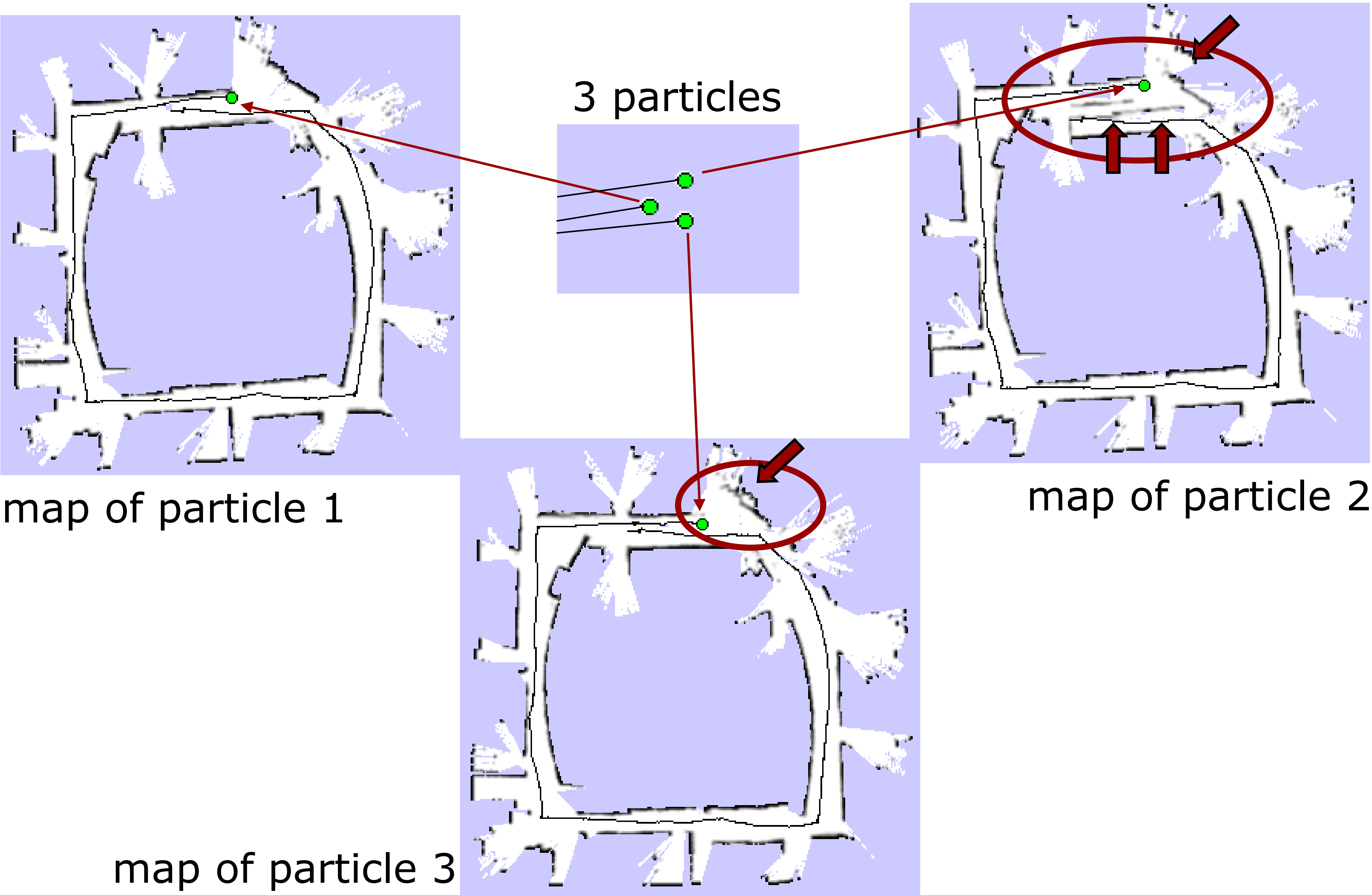
A Solution to the SLAM Problem

- **Problem:** Build a map and localize the robot in that map under pose and sensor uncertainty

- **Mapping with Rao-Blackwellized particle filters**
  - Use a particle filter to **represent potential trajectories of the robot**
  - **Each particle** carries its **own map**
  - “Survival of the fittest” based on the likelihood of observations given the map built so far

[Murphy, 99; Montemerlo et al., 03; Haehnel et al., 03; Eliazar and Parr, 03; Grisetti et al., 05]

Example: Particle Filter for Mapping



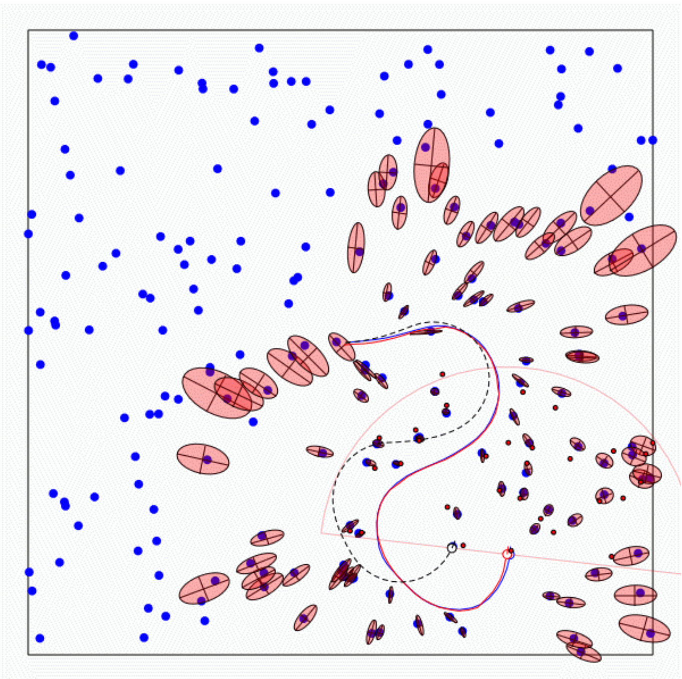
Map Representations

Typical models are:

- Feature maps
- Grid maps (occupancy or reflection probability maps)

Part 1

Part 2



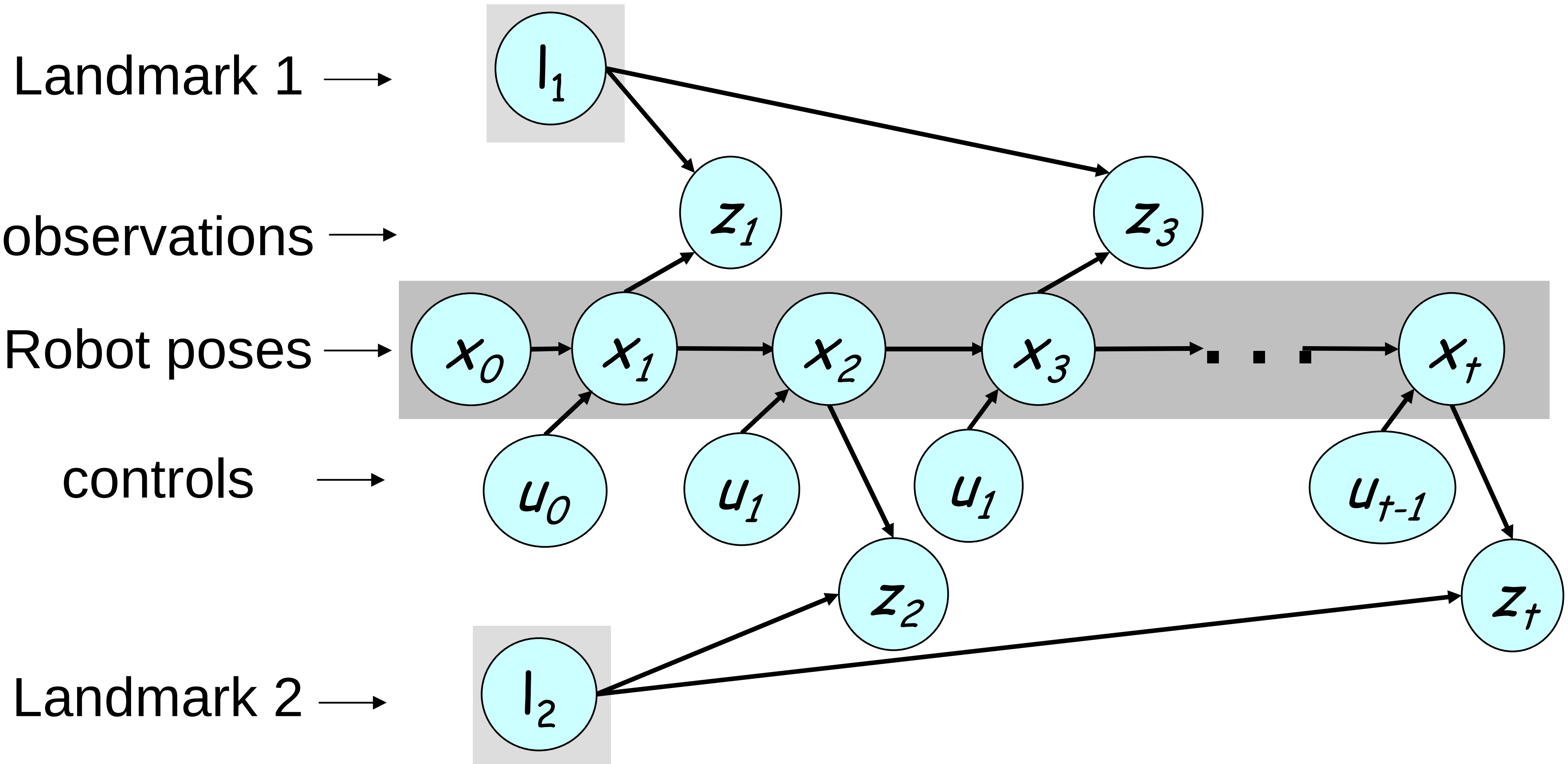
Part 1: FastSLAM

- Operates on landmarks
- Work of Mike Montemerlo et al., 2002/3

Partial slide courtesy of Mike Montemerlo!

[Montemerlo et al., 2002/2003]

Mapping using Landmarks



Knowledge of the robot's true path renders landmark positions conditionally independent

Factored Posterior

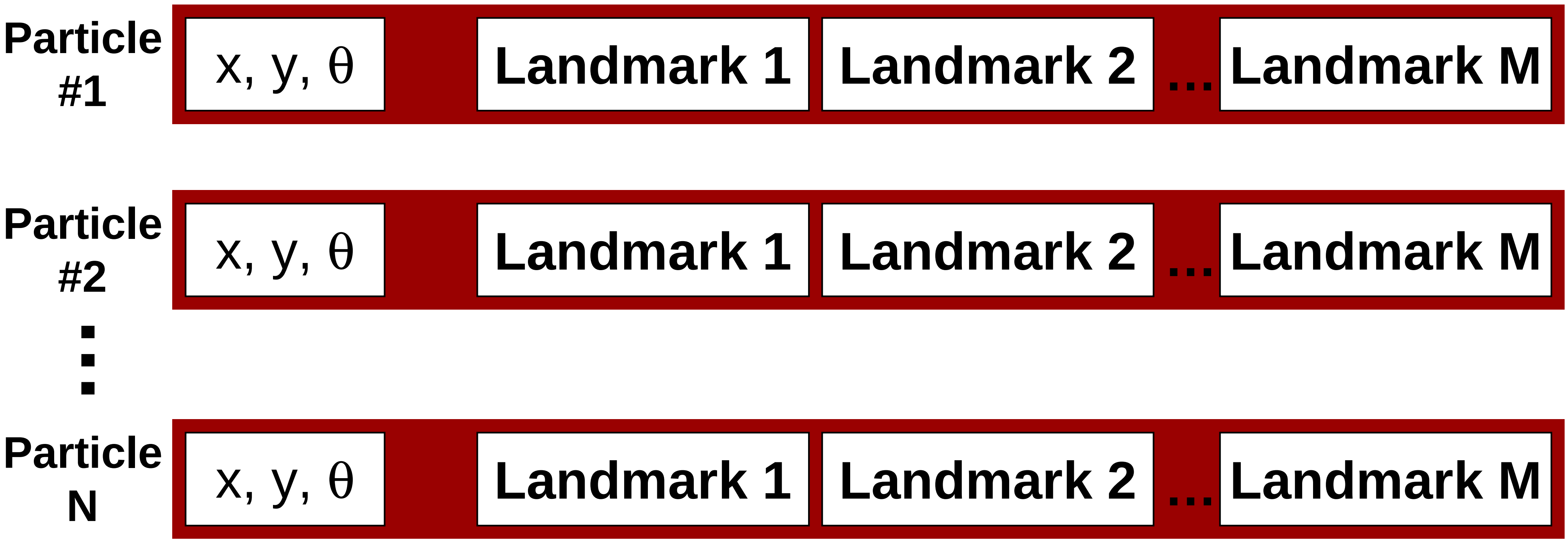
$$
\begin{aligned}
& p(x_{1:t}, l_{1:m} \mid z_{1:t}, u_{0:t-1}) \\
&= p(x_{1:t} \mid z_{1:t}, u_{0:t-1}) \cdot p(l_{1:m} \mid x_{1:t}, z_{1:t}) \\
&= p(x_{1:t} \mid z_{1:t}, u_{0:t-1}) \cdot \prod_{i=1}^{M} p(l_i \mid x_{1:t}, z_{1:t})
\end{aligned}
$$

Robot path posterior (localization problem)

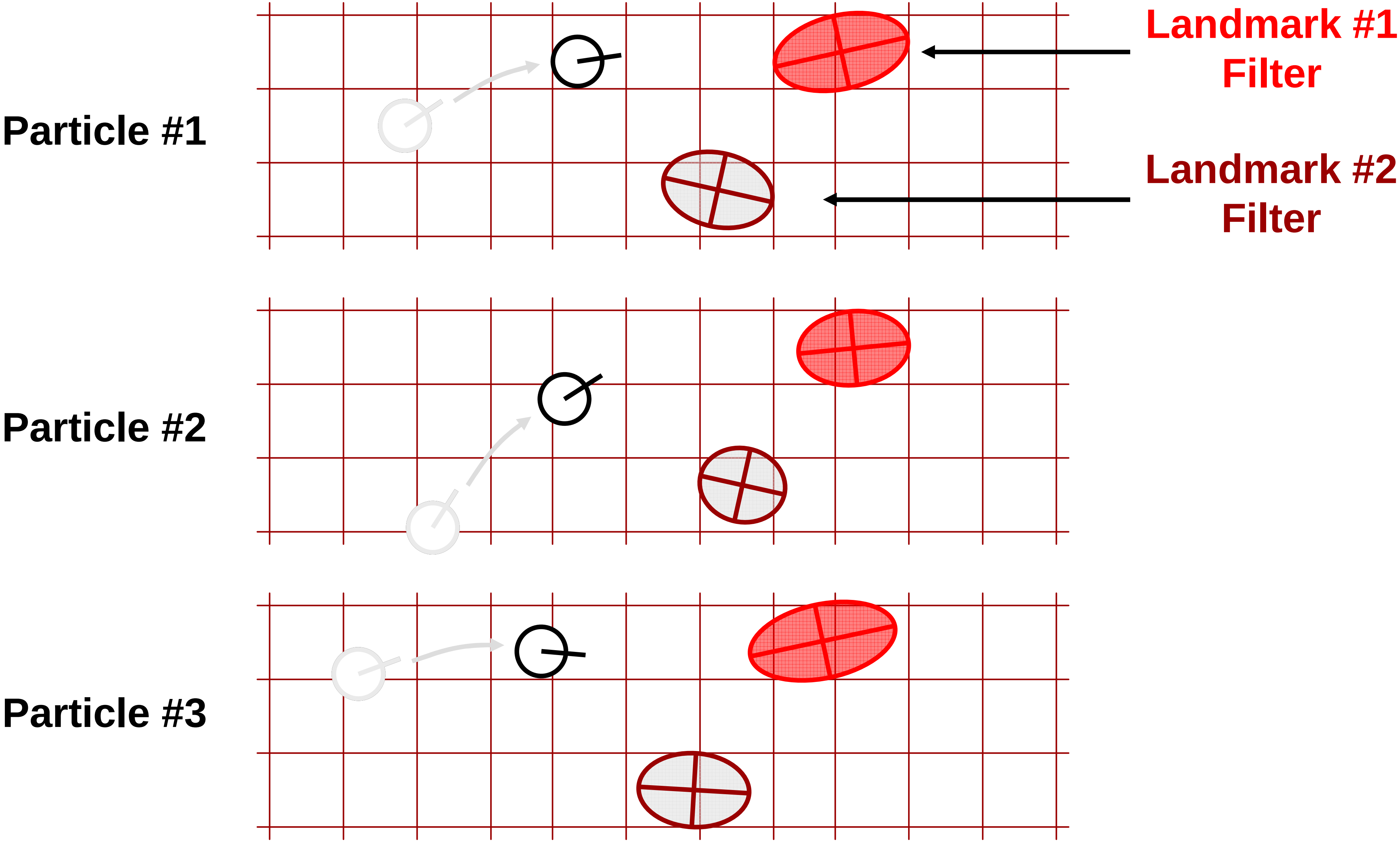
Conditionally independent landmark positions

FastSLAM

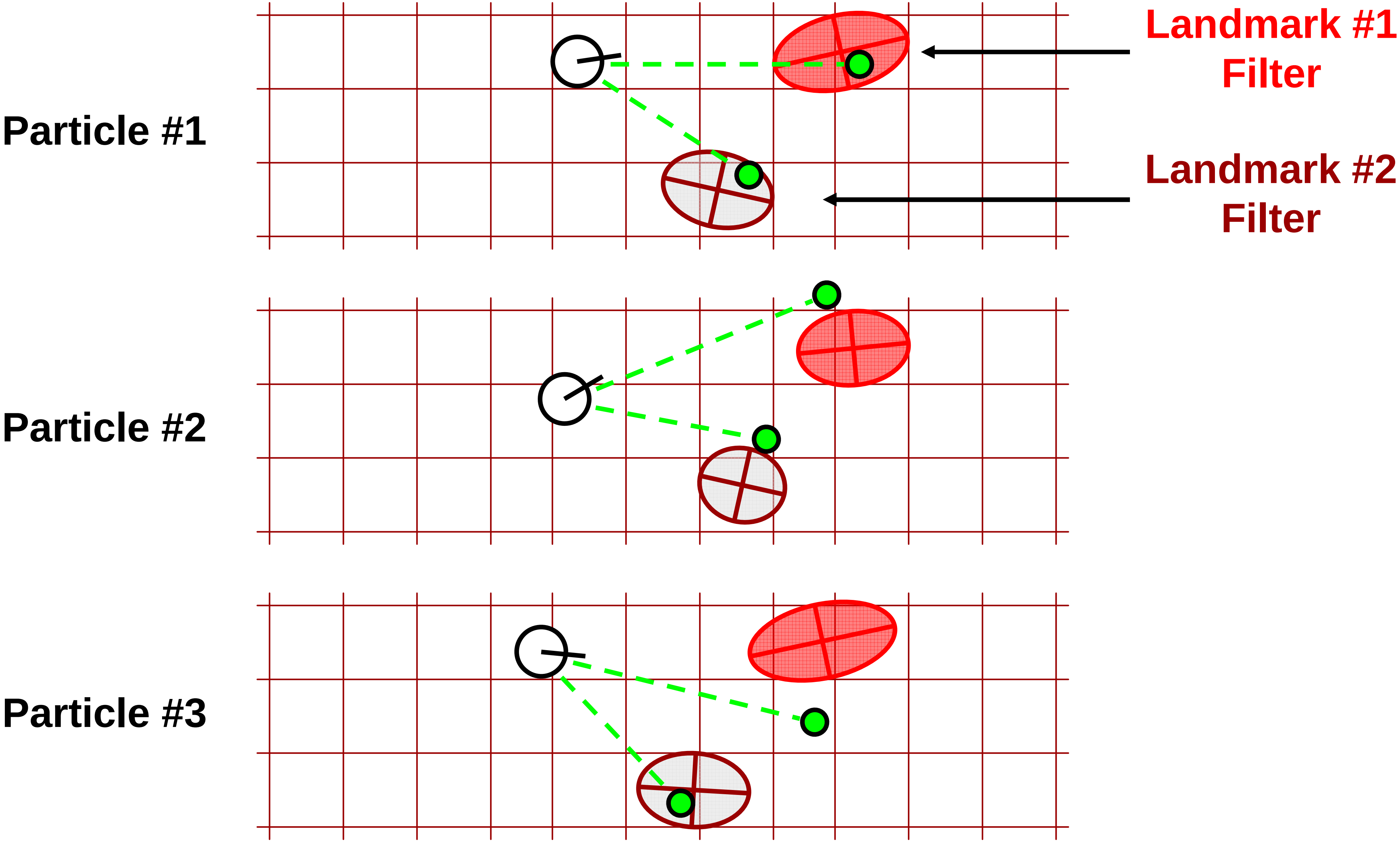
- Rao-Blackwellized particle filtering based on landmarks [Montemerlo et al., 2002]
- Each landmark is represented by a 2x2 Extended Kalman Filter (EKF)
- Each particle therefore has to maintain *M* EKFs



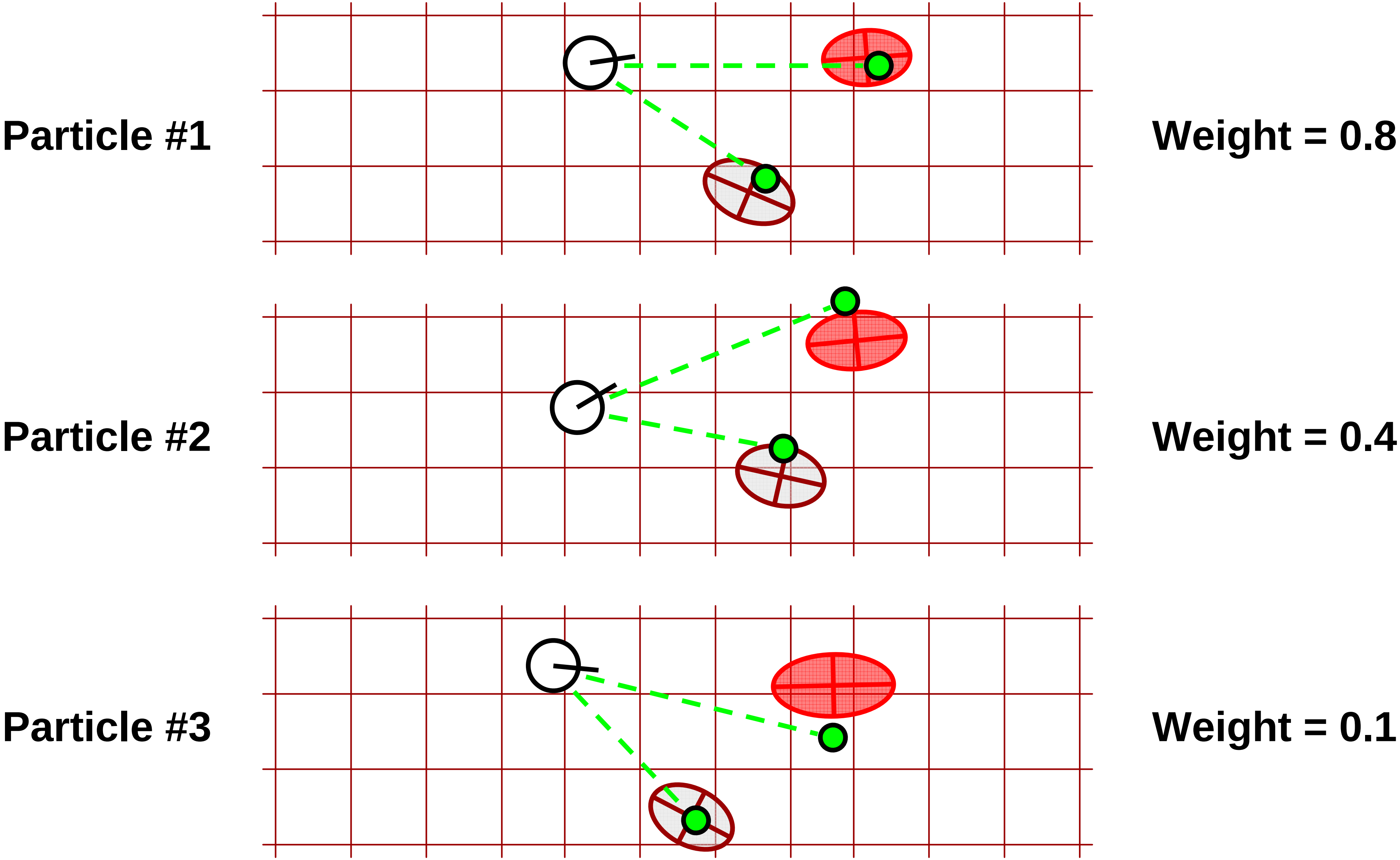
FastSLAM – Action Update



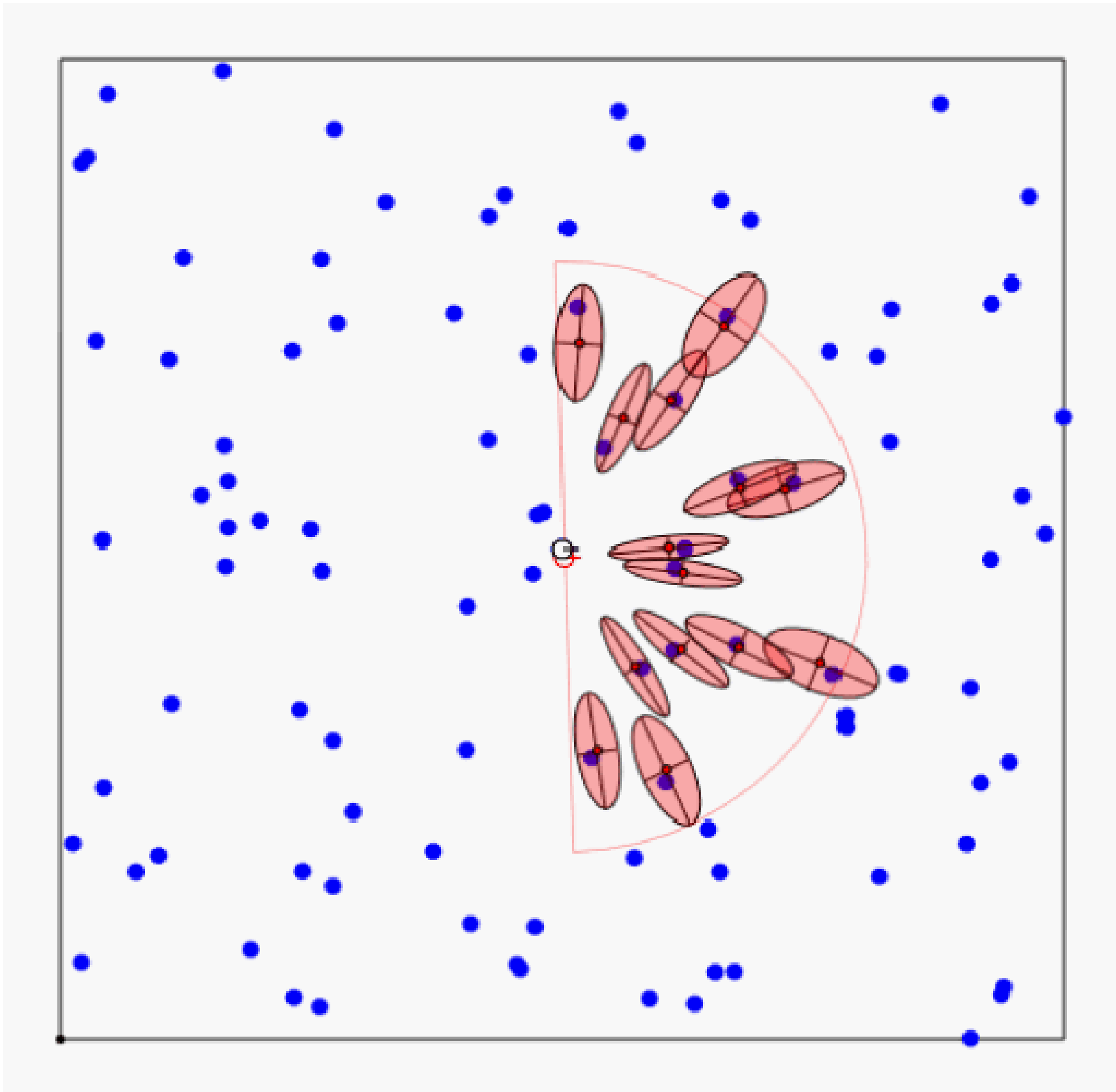
FastSLAM – Sensor Update



FastSLAM – Sensor Update



FastSLAM - Video



FastSLAM Complexity

- Update robot particles based on control u_{t-1}

$O(N)$
Constant time per particle

- Incorporate observation z_t into Kalman filters

$O(N \cdot \log(M))$
Log time per particle

- Resample particle set

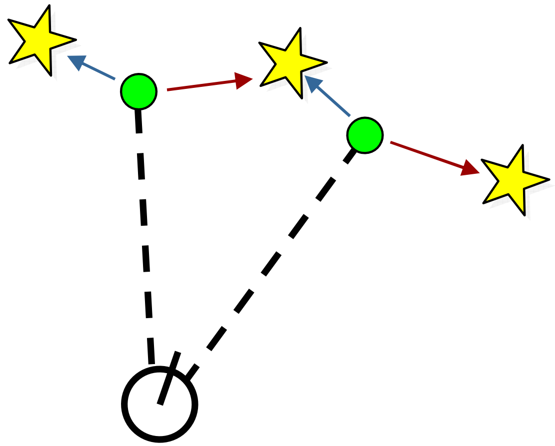
$O(N \cdot \log(M))$
Log time per particle

N = Number of particles
M = Number of map features

$\mathbf{O(N \cdot \log(M))}$
Log time per particle

Data Association Problem

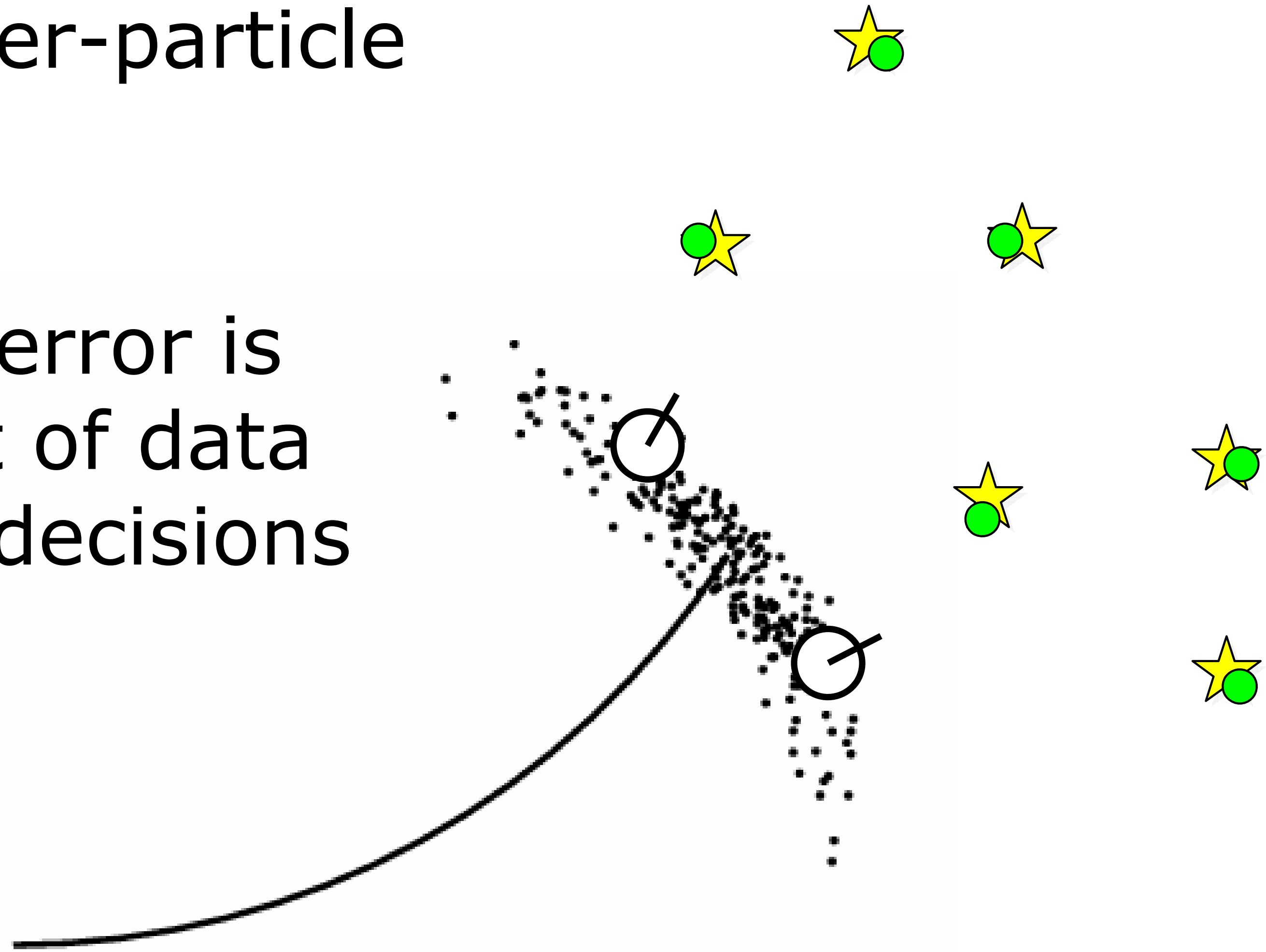
- Which observation belongs to which landmark?



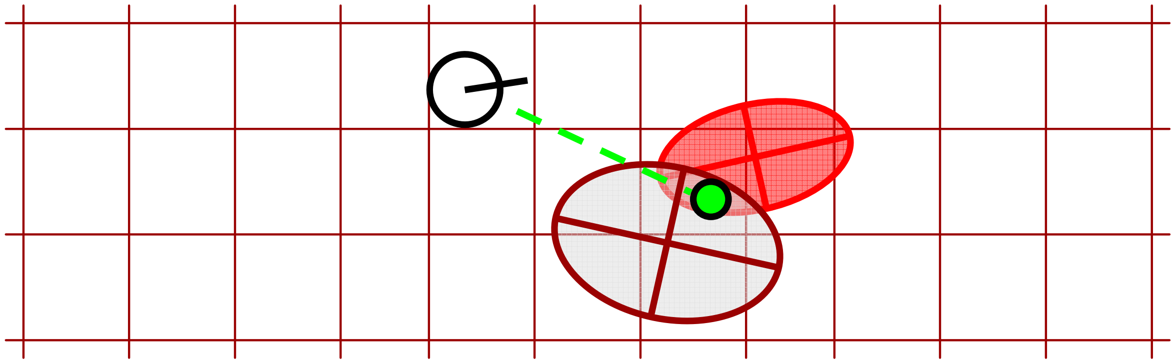
- A robust SLAM must consider possible data associations
- Potential data associations depend also on the pose of the robot

Multi-Hypothesis Data Association

- Data association is done on a per-particle basis
- Robot pose error is factored out of data association decisions



Per-Particle Data Association



Was the observation generated by the red or the blue landmark?

P(observation|red) = 0.3 P(observation|blue) = 0.7

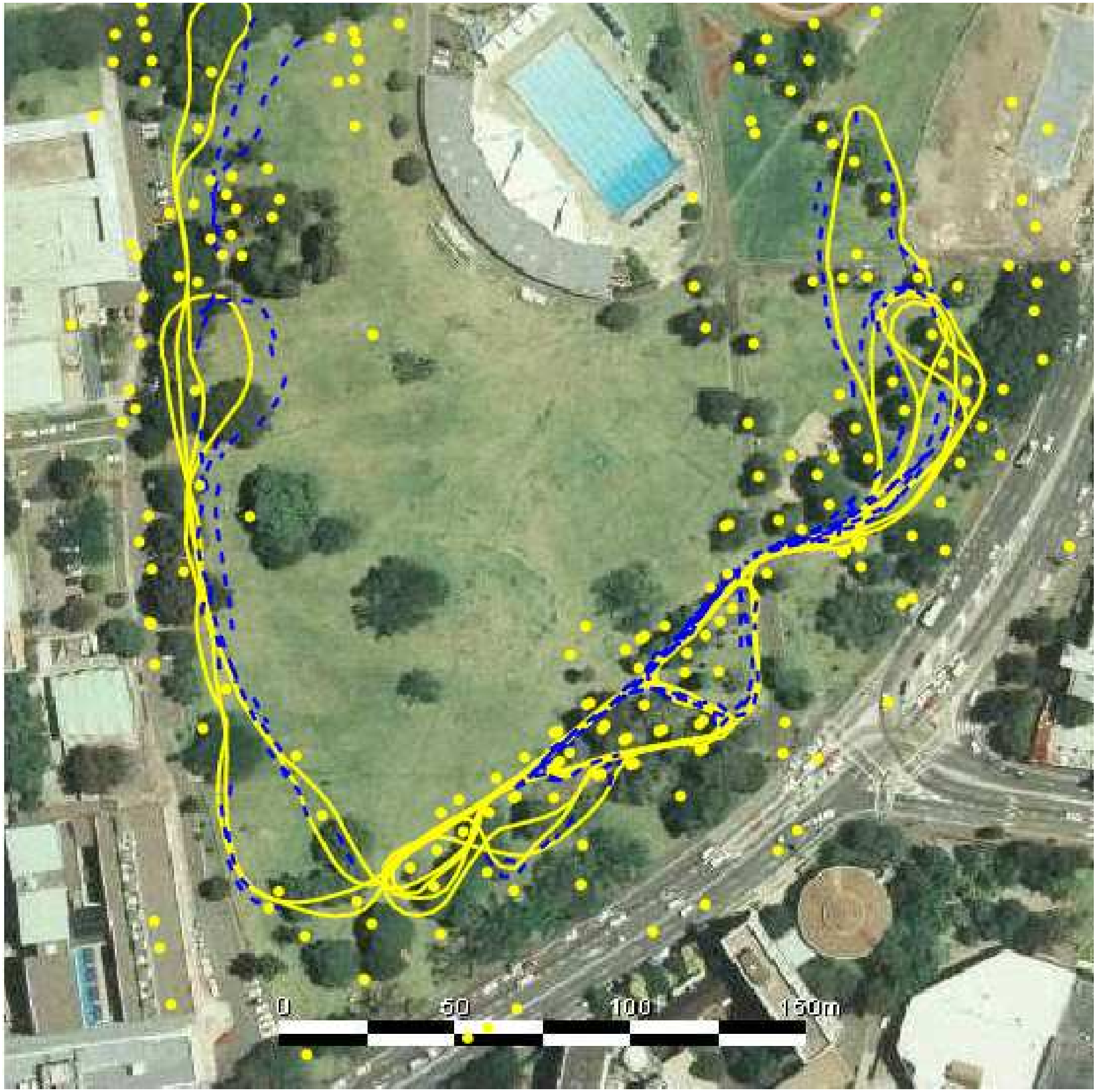
- Two options for per-particle data association
  - Pick the most probable match
  - Pick an random association weighted by the observation likelihoods
- If the probability is too low, generate a new landmark

Results – Victoria Park (Video)

Dataset courtesy of University of Sydney

Results – Victoria Park

- 4 km traverse
- < 5 m RMS position error
- 100 particles



Dataset courtesy of University of Sydney

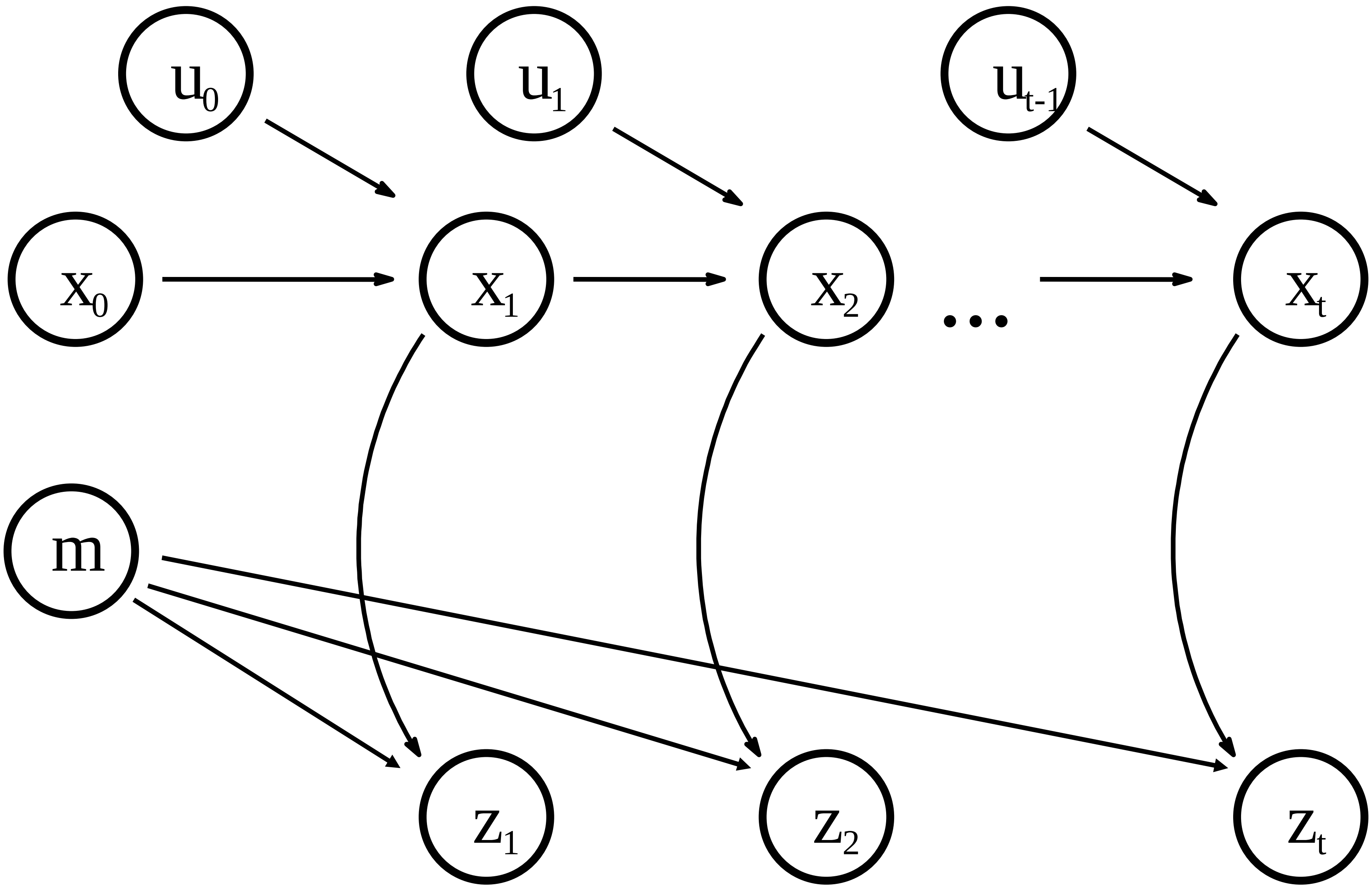
Blue = GPS
Yellow = FastSLAM

Part 2: Grid-based FastSLAM

- Operates on grid maps and typically laser range data

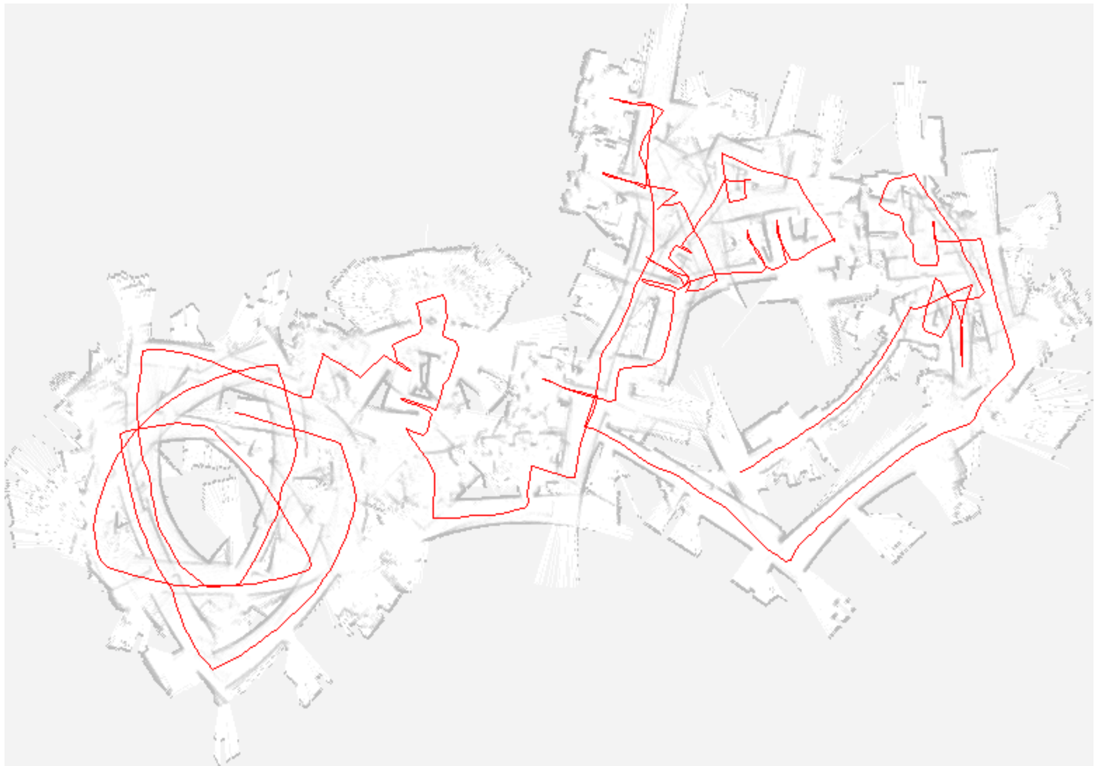
- Dirk Haehnel et al., 2003
- Eliazar and Parr, 2003
- Grisetti et al., 2005-2007
- several others…

The Graphical Model



Applying a Standard RBPF for Learning Grid Maps Does not Work…

- Standard laser end-point model
- Odometry motion model
- Number of particles varying from 500 to 2.000
- Typical result:



Problems in Practice

- Each (grid) map is rather big
- Each particle maintains its own map
- Therefore, one needs to keep the number of particles small

- **Solution**:
  Compute better proposal distributions

- **Idea**:
  Improve the pose estimate **before** applying the particle filter

Pose Correction Using Scan Matching

Maximize the likelihood of the i-th pose relative to the (i-1)-th pose

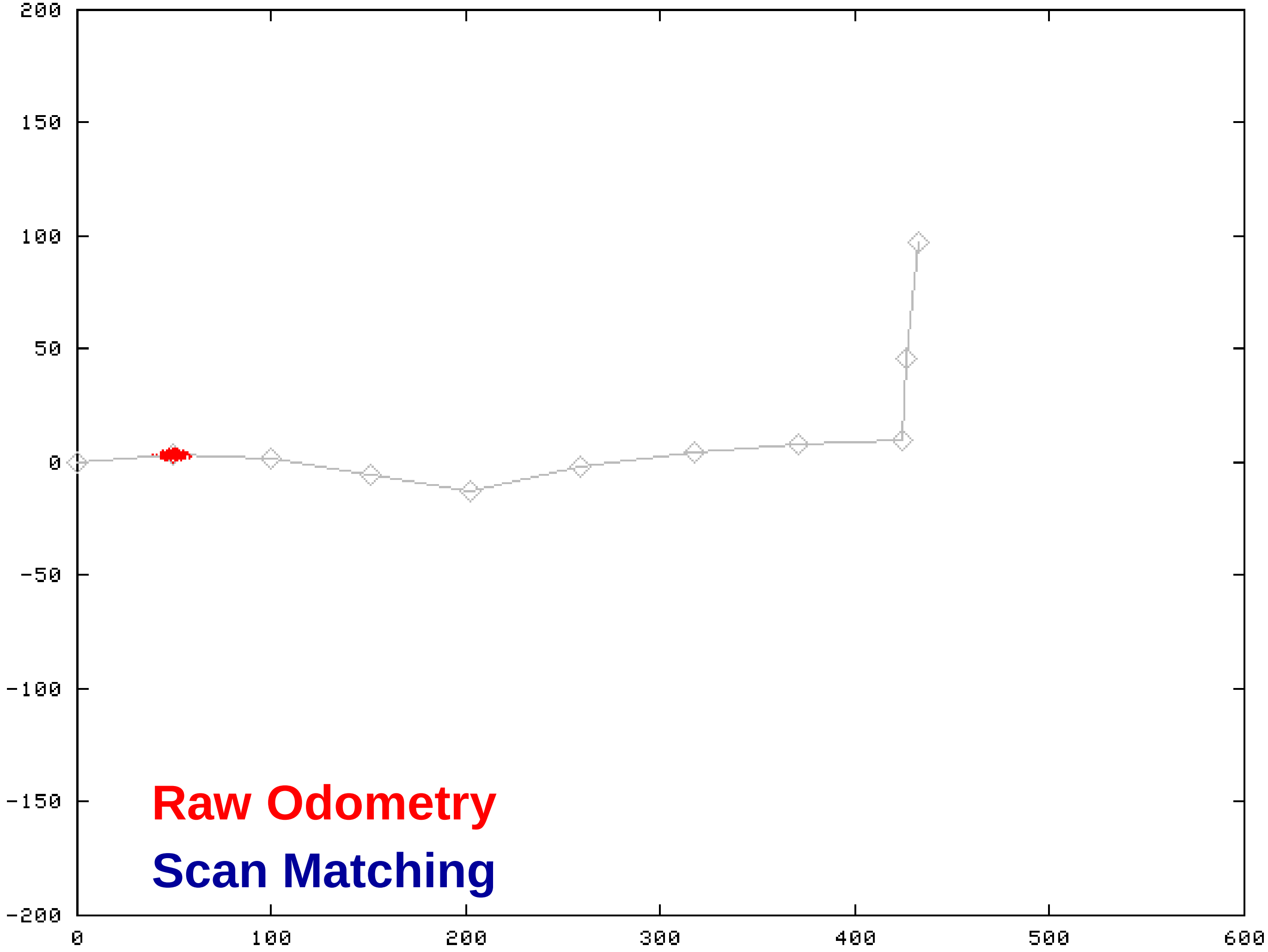
$$x_t^* = \underset{x_t}{\operatorname{argmax}}\, p(z_t \mid x_t, m_{t-1}) \cdot p(x_t \mid x_{t-1}^*, u_{t-1})$$

current measurement

robot motion

map constructed so far

Motion Model for Scan Matching



Mapping using Scan Matching

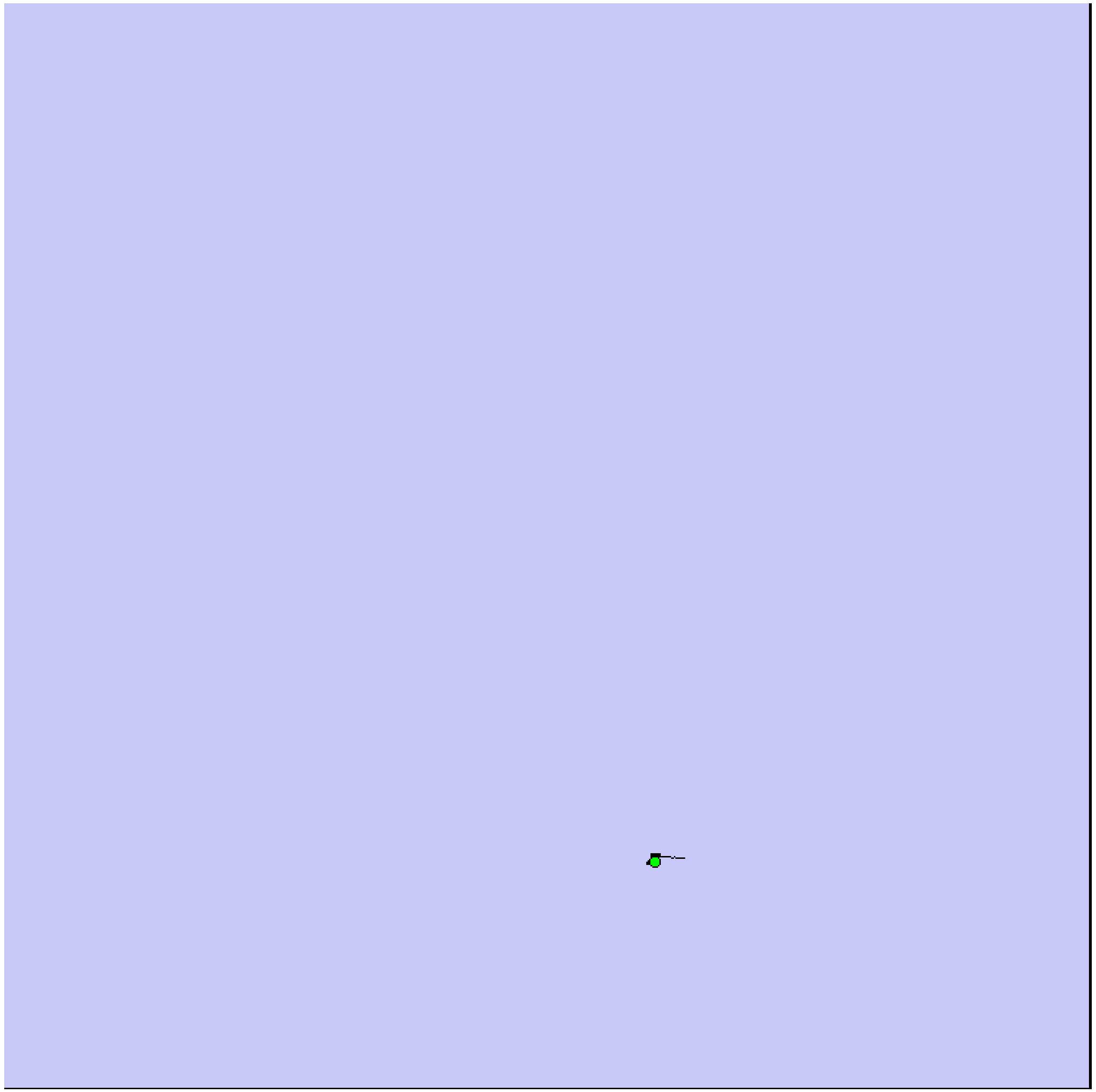
RBPF-SLAM with Improved Odometry

- Scan-matching provides a **locally consistent** pose correction
- Pre-correct short odometry sequences using scan-matching and use them as input to the Rao-Blackwellized PF
- Fewer particles are needed, since the error in the input in smaller

[Haehnel et al., 2003]

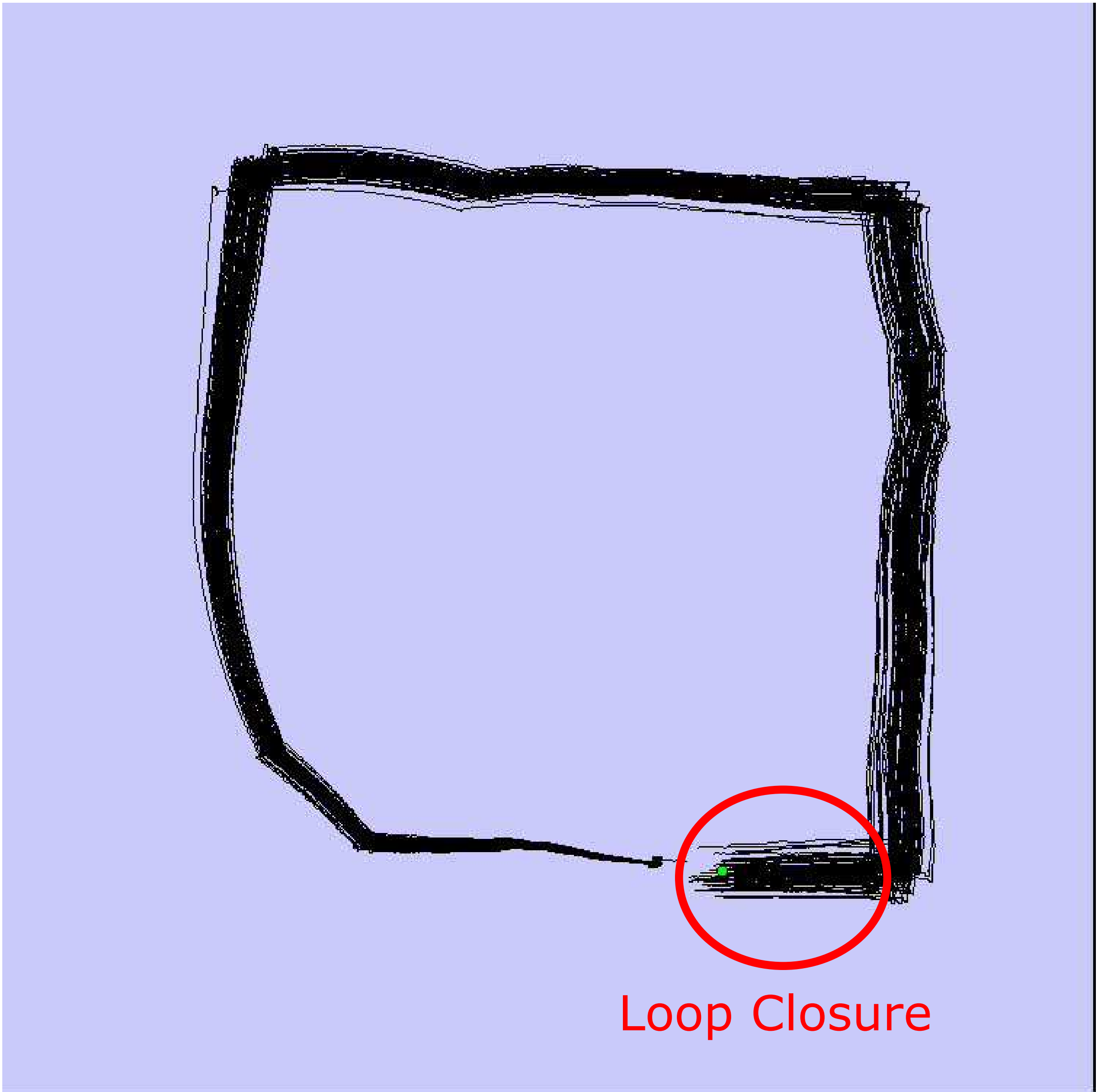
RBPF-SLAM with Scan-Matching

Map: Intel Research Lab Seattle



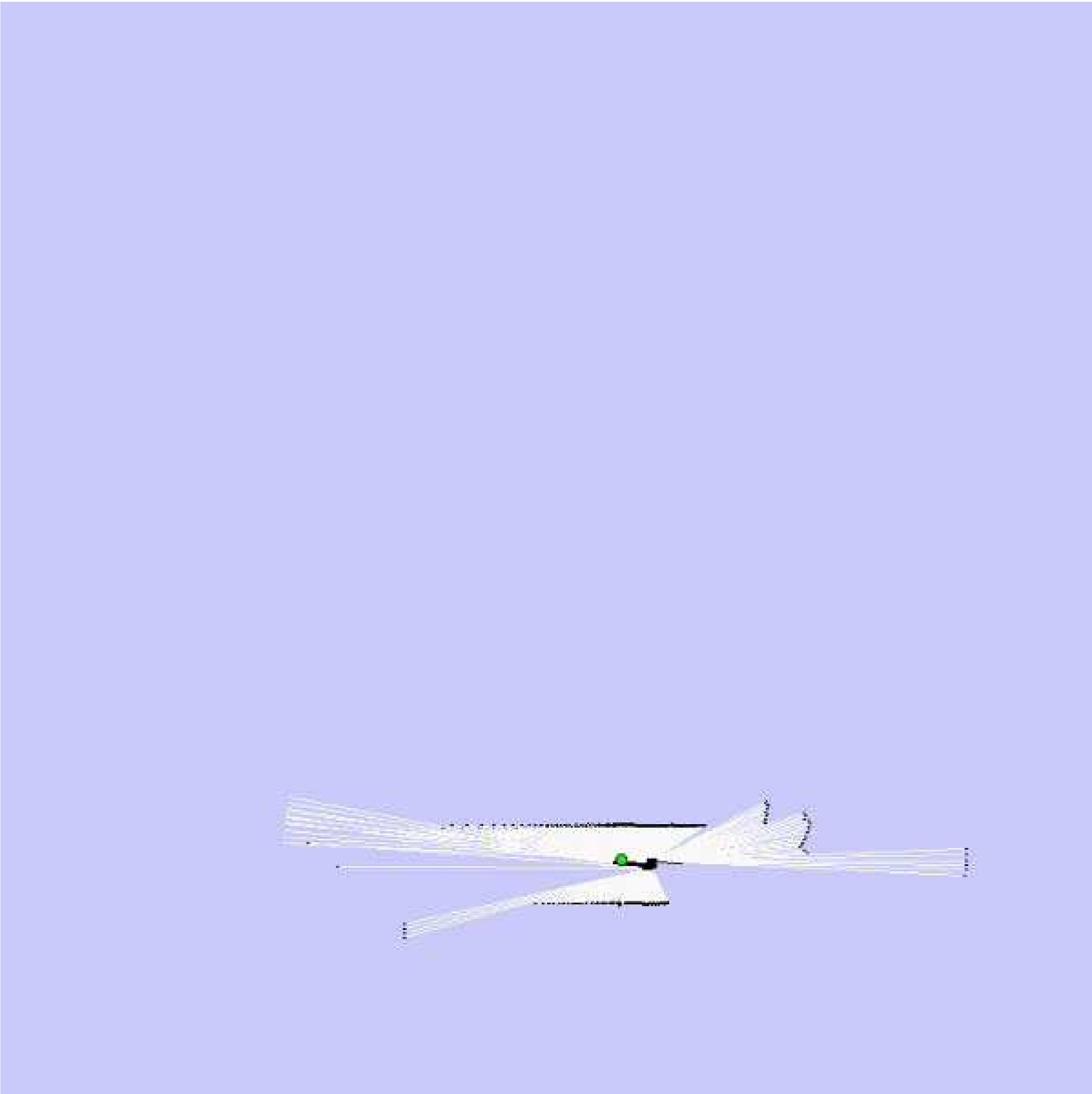
RBPF-SLAM with Scan-Matching

Map: Intel Research Lab Seattle

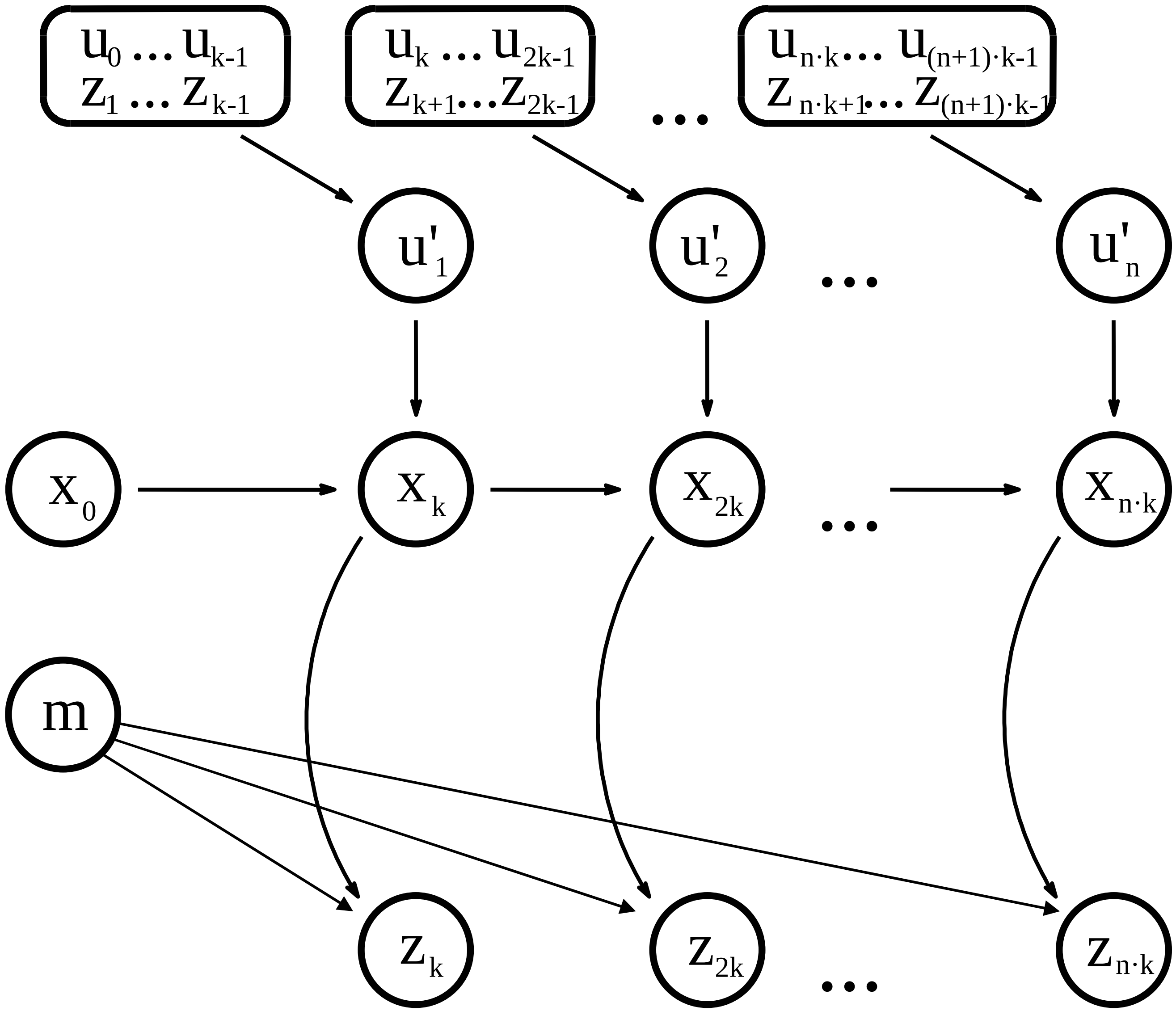


RBPF-SLAM with Scan-Matching

Map: Intel Research Lab Seattle



Graphical Model for Mapping with Improved Odometry



Conclusion (so far…)

- The presented approach is efficient
- It is easy to implement
- Scan matching is used to transform sequences of laser measurements into odometry measurements
- Provides good results for most medium-size datasets

What's Next?

- Further reduce the number of particles
- Improved proposals will lead to more accurate maps
- Use the properties of our sensor when drawing the next generation of particles

Proposal Distribution

- A particle filter uses a proposal distribution to sample the next generation of samples
- Most efficient SLAM methods use Gaussian approximation of the robot's motion model or of the observation likelihood function
- The optimal proposal is [Doucet, 98]:

observation likelihood | motion model

$$p(x_t | x_{t-1}^{(i)}, m^{(i)}, z_t, u_t) = \frac{p(z_t | x_t, m^{(i)}) p(x_t | x_{t-1}^{(i)}, u_t)}{p(z_t | x_{t-1}^{(i)}, m^{(i)})}$$

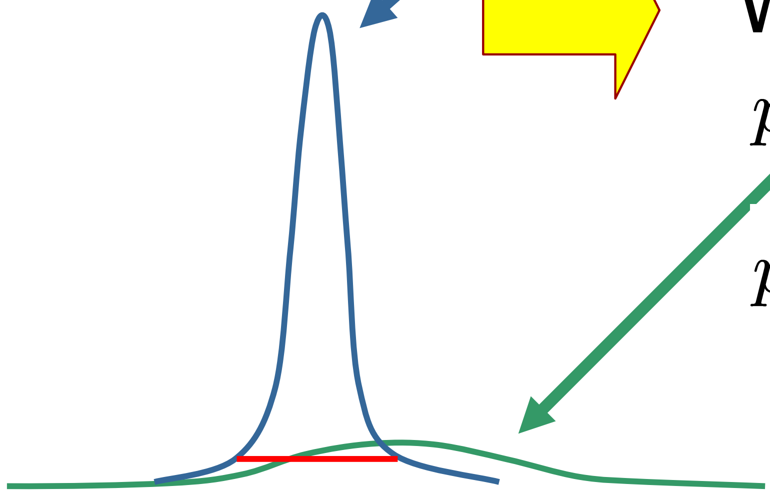
normalizer

The Optimal Proposal Distribution

$$p(x_t|x_{t-1}^{(i)}, m^{(i)}, z_t, u_t) = \frac{p(z_t|x_t, m^{(i)})p(x_t|x_{t-1}^{(i)}, u_t)}{\int p(z_t|x_t, m^{(i)})p(x_t|x_{t-1}^{(i)}, u_t)dx_t}$$

[Doucet, 98]

For lasers $p(z_t|x_t, m^{(i)})$ is extremely peaked and dominates the product.



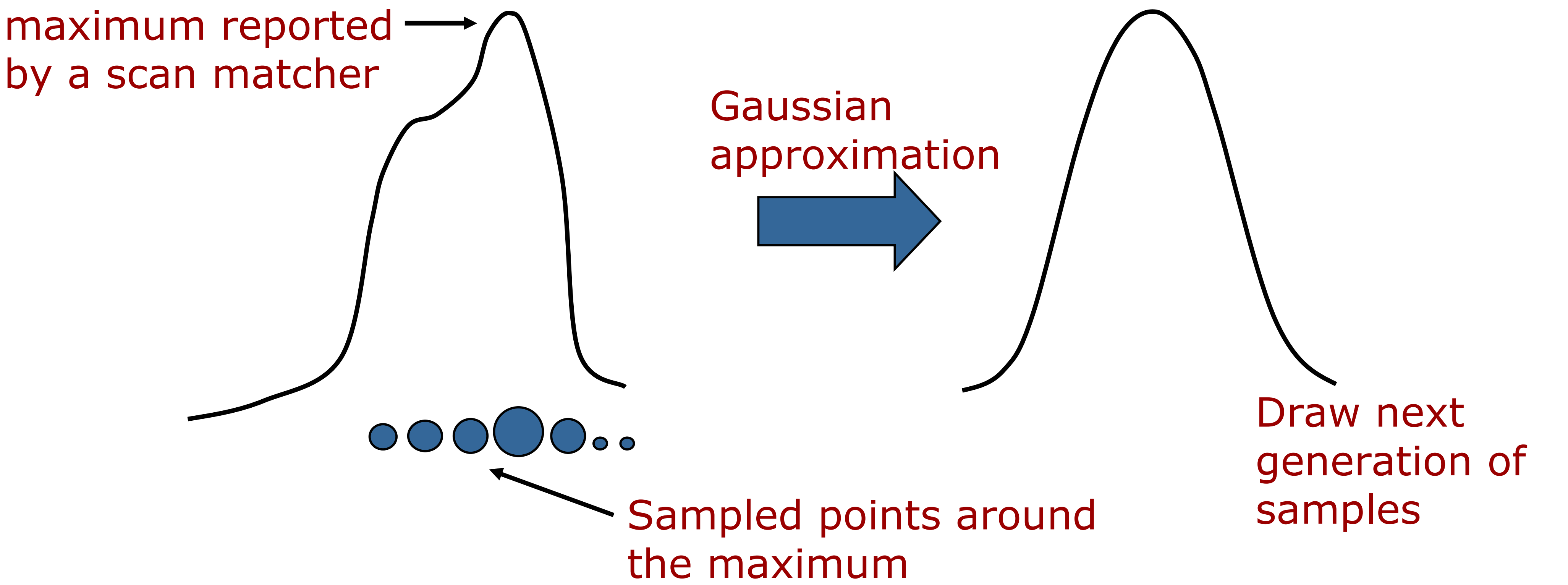
We can safely approximate $p(x_t|x_{t-1}^{(i)}, u_t)$ by a constant:

$$p(x_t|x_{t-1}^{(i)}, u_t)\ |_{x_t:p(z_t|x_t,m^{(i)})>\epsilon} = c$$

Resulting Proposal Distribution

$$p(x_t|x_{t-1}^{(i)}, m^{(i)}, z_t, u_t) \simeq \frac{p(z_t|x_t, m^{(i)})}{\int_{x_t \in \{x|p(z_t|x, m^{(i)}) > \epsilon\}} p(z_t|x_t, m^{(i)}) dx_t}$$

Approximate this equation by a Gaussian:



Resulting Proposal Distribution

$$p(x_t|x_{t-1}^{(i)}, m^{(i)}, z_t, u_t) \simeq \frac{p(z_t|x_t, m^{(i)})}{\int_{x_t \in \{x|p(z_t|x,m^{(i)})>\epsilon\}} p(z_t|x_t, m^{(i)}) dx_t}$$

Approximate this equation by a Gaussian:

$$p(x_t|x_{t-1}^{(i)}, m^{(i)}, z_t, u_t) \simeq \mathcal{N}(\mu^{(i)}, \Sigma^{(i)})$$

$$\mu^{(i)} = \frac{1}{\eta} \sum_{j=1}^{K} x_j \, p(z_t|x_j, m^{(i)})$$

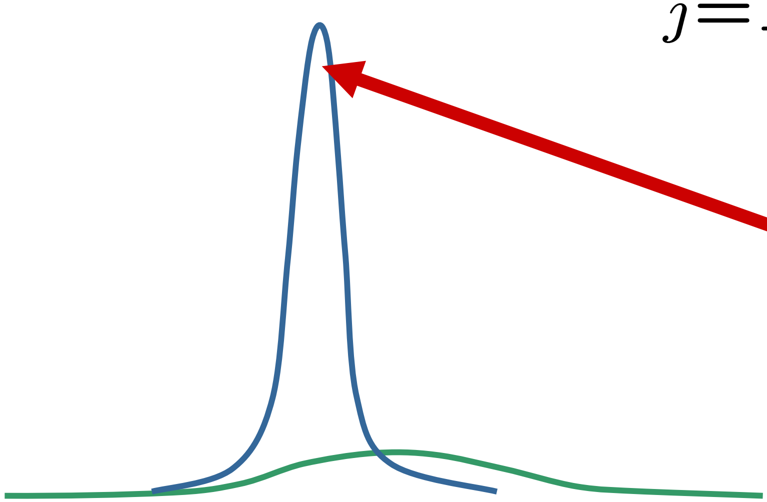
$$\Sigma^{(i)} = \frac{1}{\eta} \sum_{j=1}^{K} (x_j - \mu^{(i)})(x_j - \mu^{(i)})^T p(z_t|x_j, m^{(i)})$$

η is a normalizer

Sampled around the scan-match maxima

Computing the Importance Weight

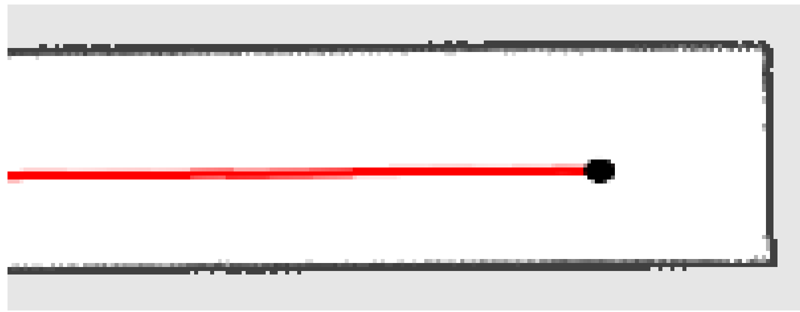
$$
\begin{aligned}
w_t^{(i)} &= w_{t-1}^{(i)} p(z_t | x_{t-1}^{(i)}, m^{(i)}) \\
&\simeq w_{t-1}^{(i)} \int p(z_t | x_t, m^{(i)}) p(x_t | x_{t-1}^{(i)}, u_t) d\, x_t \\
&\simeq w_{t-1}^{(i)} c \int_{x_t \in \{x | p(z_t | x, m^{(i)}) > \epsilon\}} p(z_t | x_t, m^{(i)}) dx_t \\
&\simeq w_{t-1}^{(i)} c \sum_{j=1}^{K} p(z_t | x_j, m^{(i)})
\end{aligned}
$$



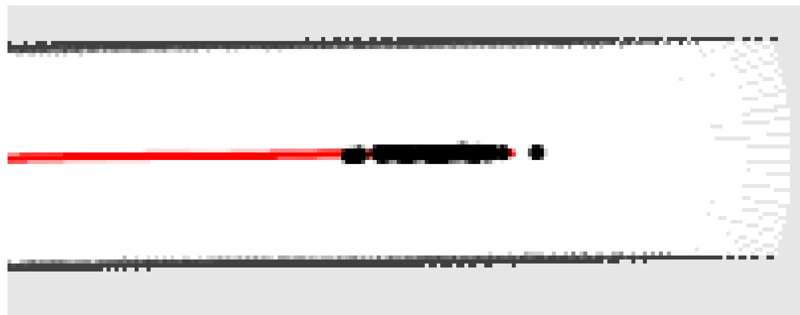
Sampled points around the maximum of the observation likelihood

Improved Proposal

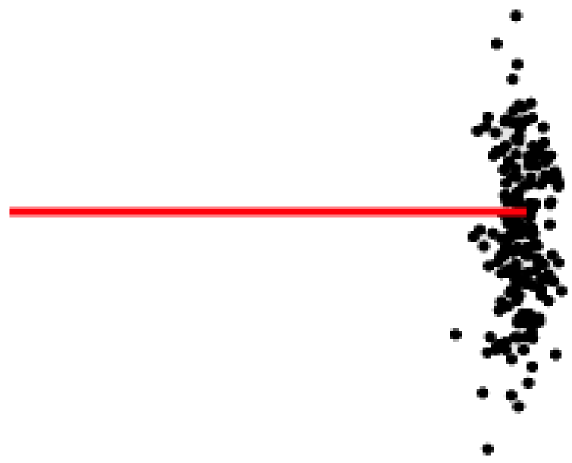
End of a corridor:



Corridor:

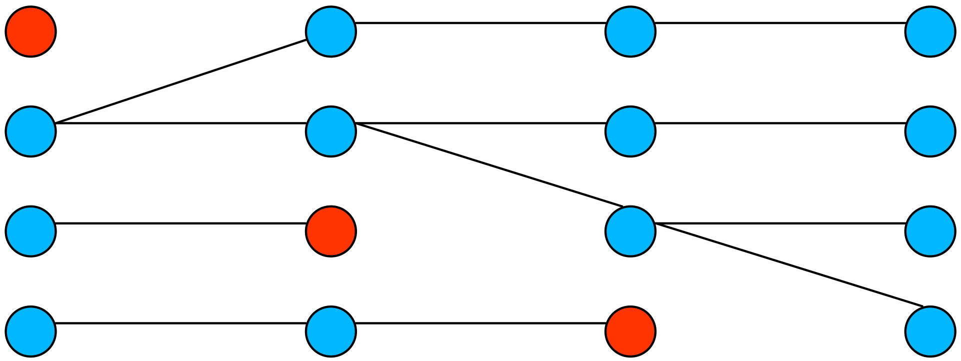


Free space:



Is Resampling Needed?

- If all particles have the save weight, resampling is useless.
- Using an improved proposal reduces the need of resampling.
- Particle depletion problem



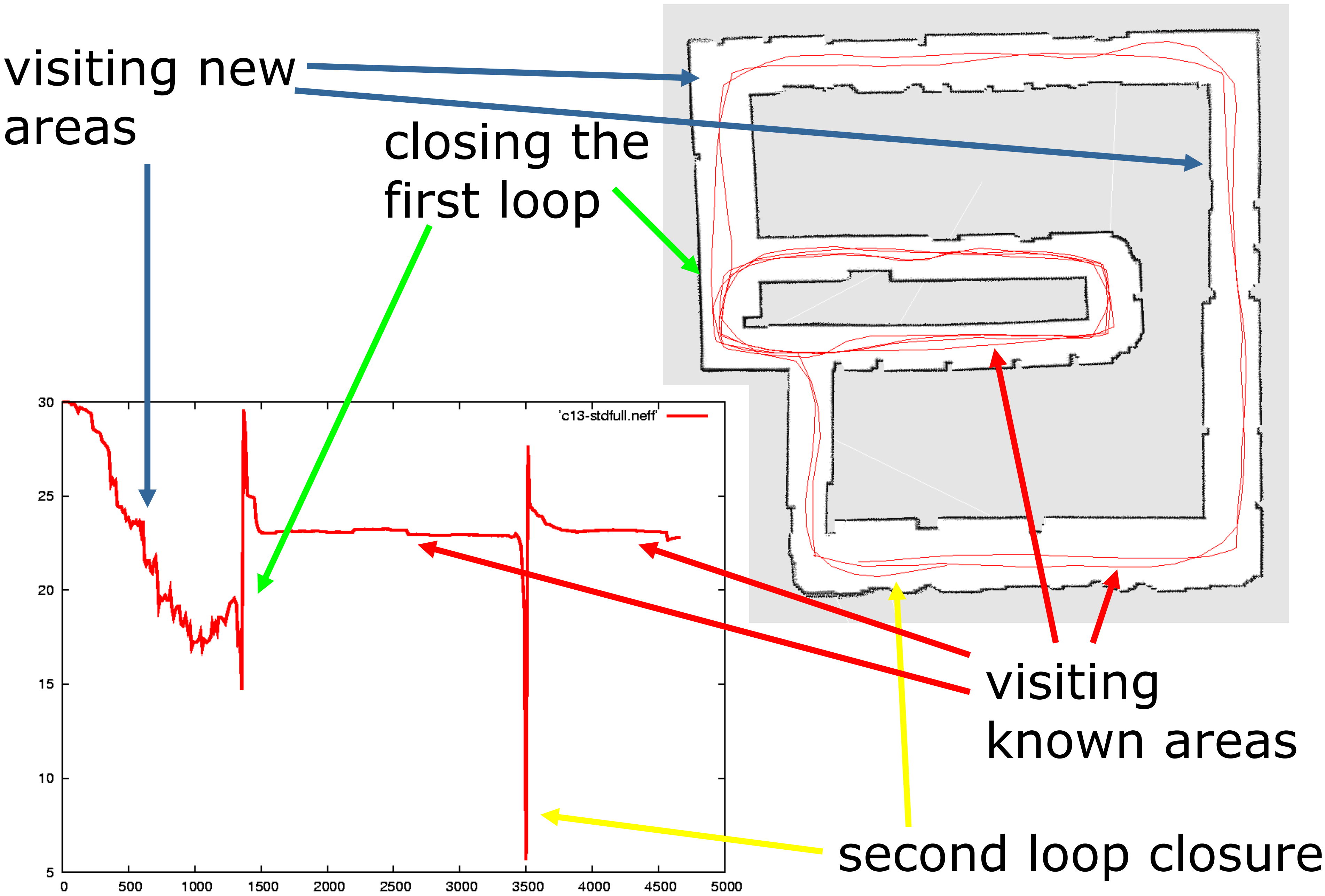
Goal: resample only if needed!

Effective Number of Particles

$$n_{eff} = \frac{1}{\sum_i \left(w_t^{(i)}\right)^2}$$

- Empirical measure of how well the goal distribution is approximated by samples drawn from the proposal
- We only resample when n_{eff} drops below a given threshold.
- See [Doucet, '98; Arulampalam, '01]

Typical Evolution of n_{eff}

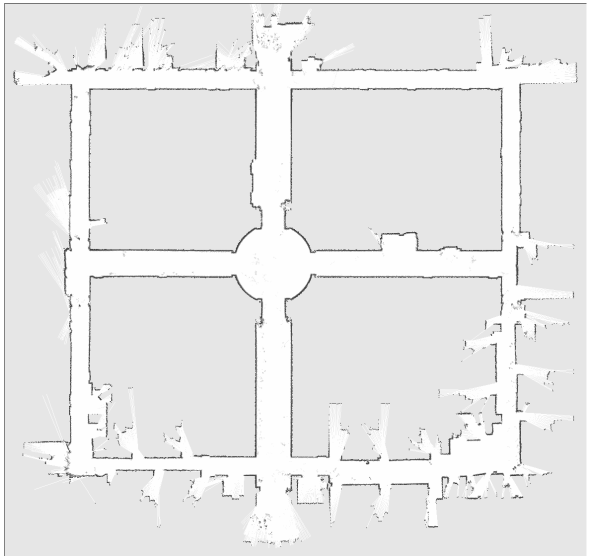


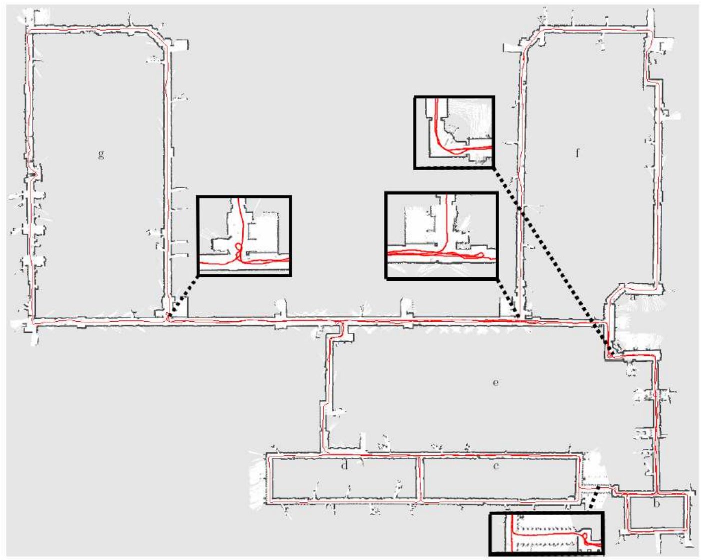
Intel Research Lab

- **15 particles**
- four times faster than real-time P4, 2.8GHz
- 5cm resolution during scan matching
- 1cm resolution in final map

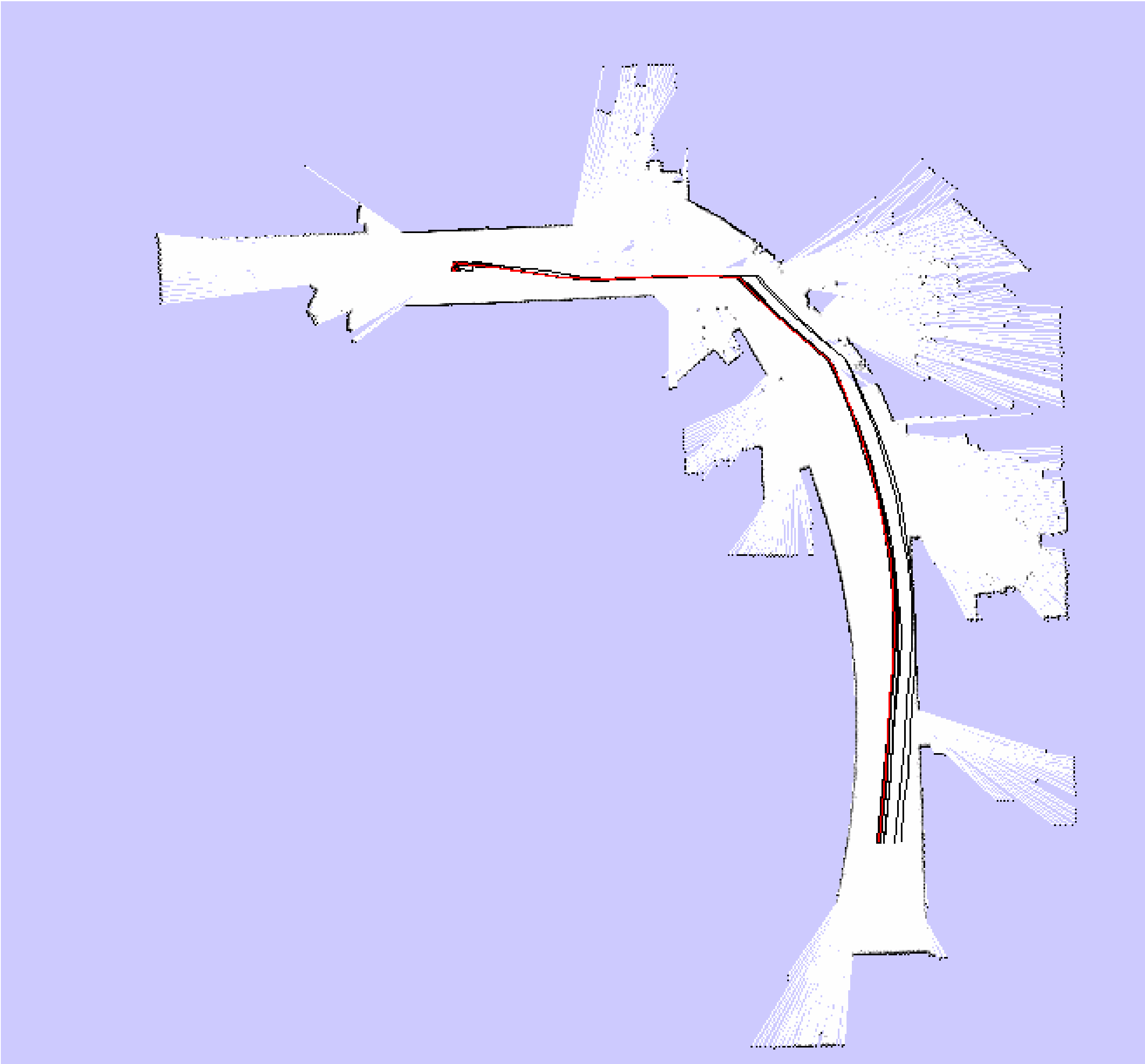
Experiments

- Real world datasets





Intel Research Lab

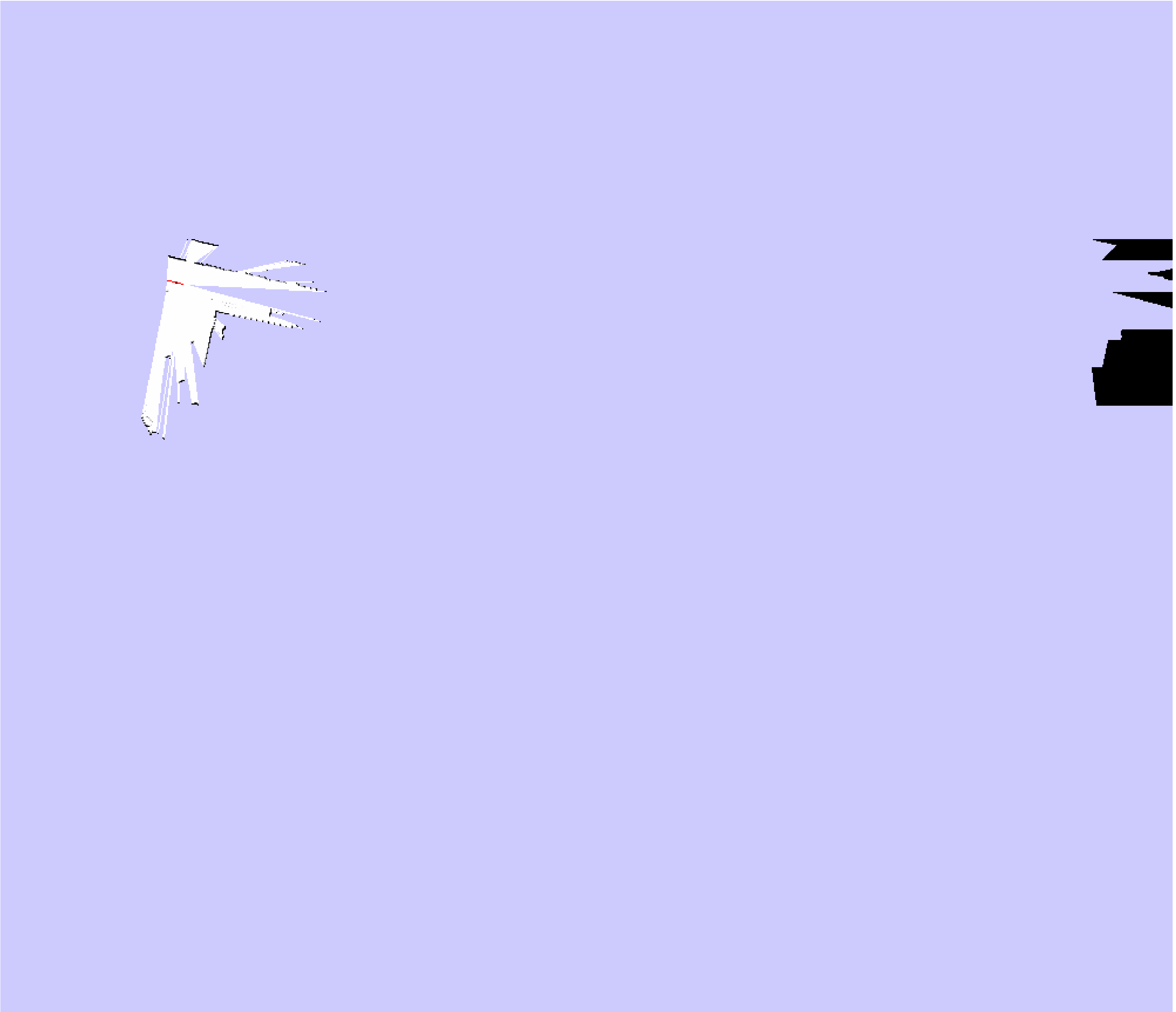


- **15 particles**
- Compared to RBPF-SLAM with Scan-Matching, the particles are propagated closer to the true distribution

Outdoor Campus Map

- **30 particles**
- 250x250m^2
- 1.75 km (odometry)
- 20cm resolution during scan matching
- 30cm resolution in final map

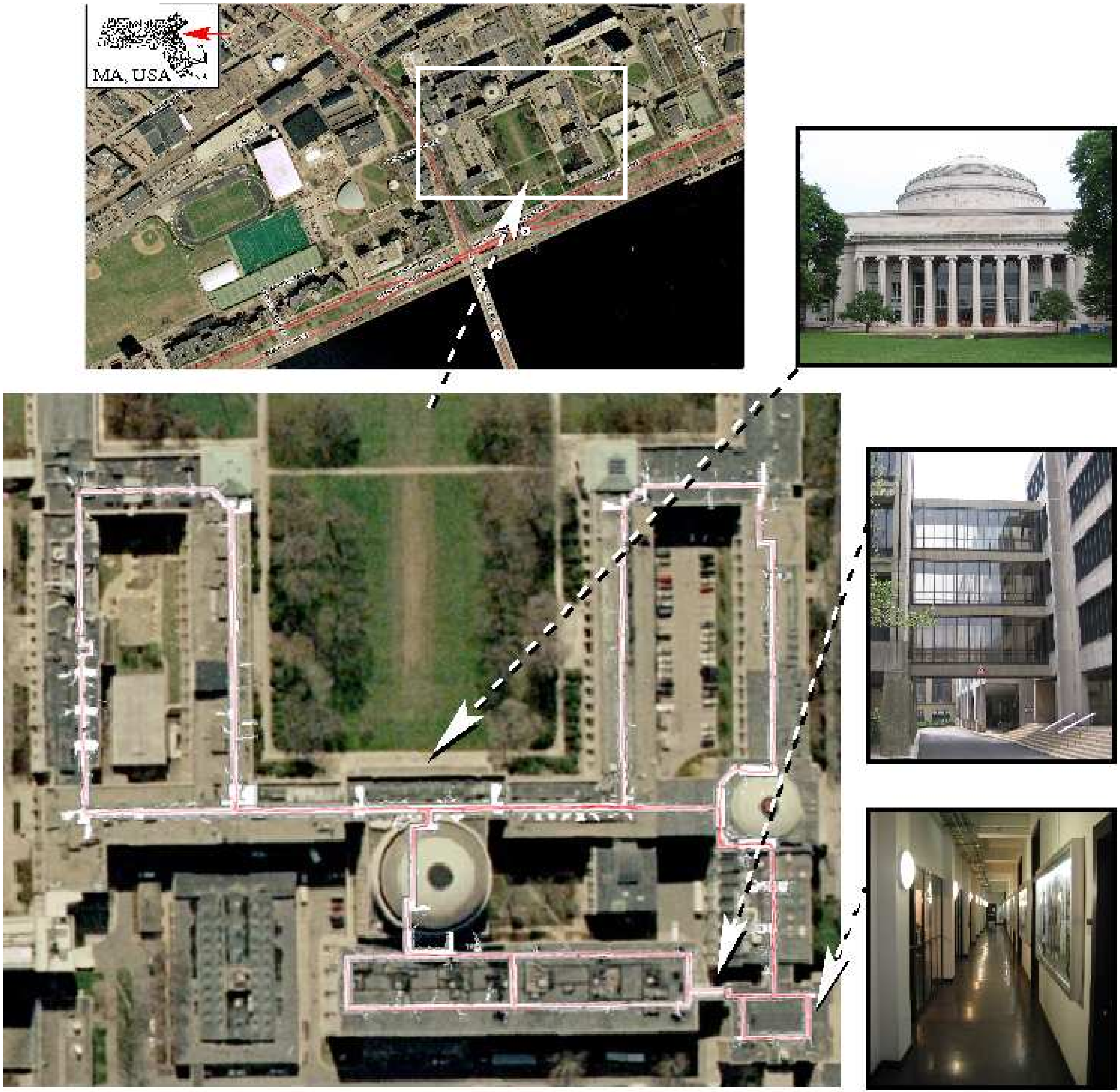
Outdoor Campus Map - Video



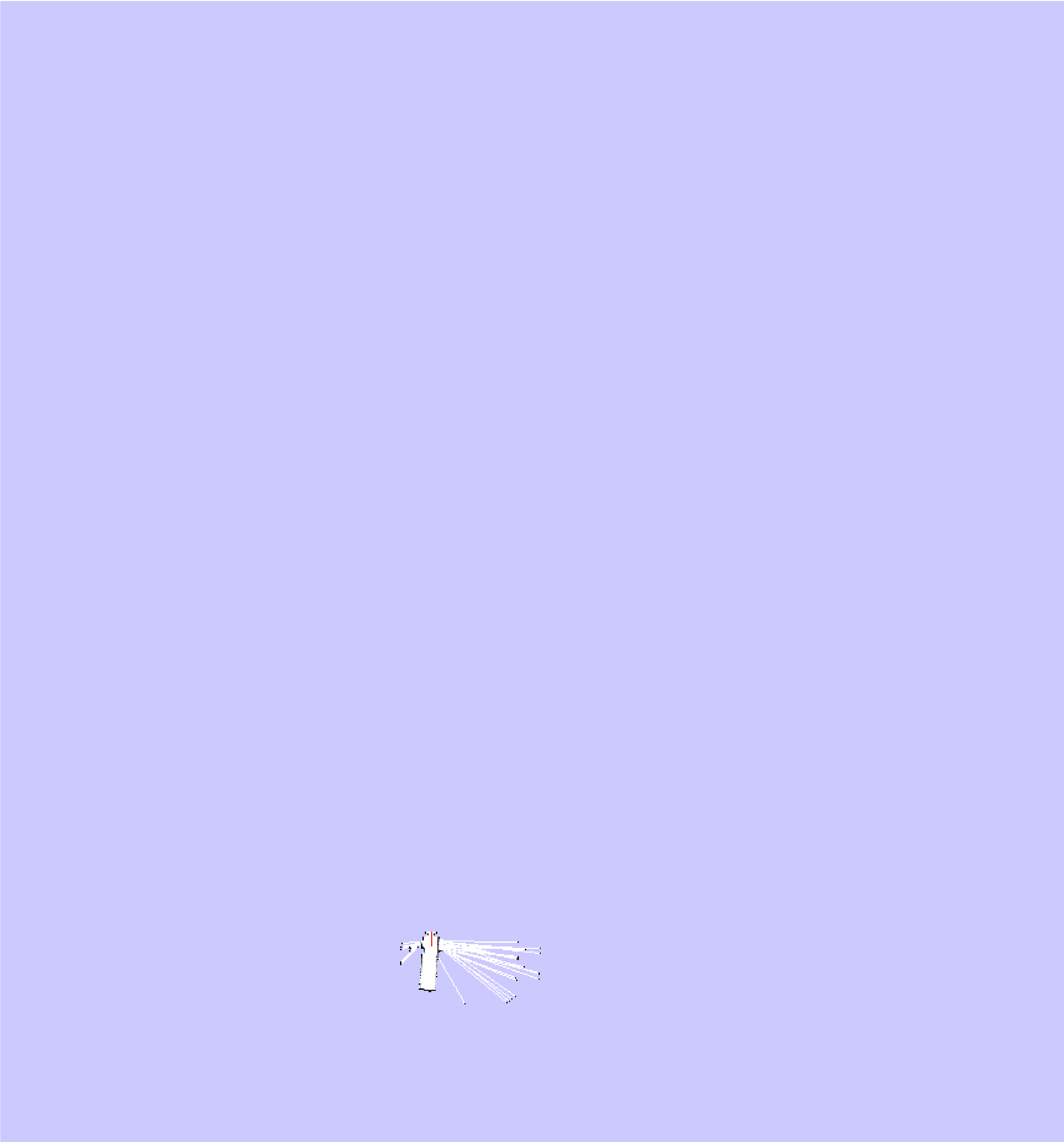
MIT Killian Court

- The **"infinite-corridor-dataset"** at MIT

MIT Killian Court

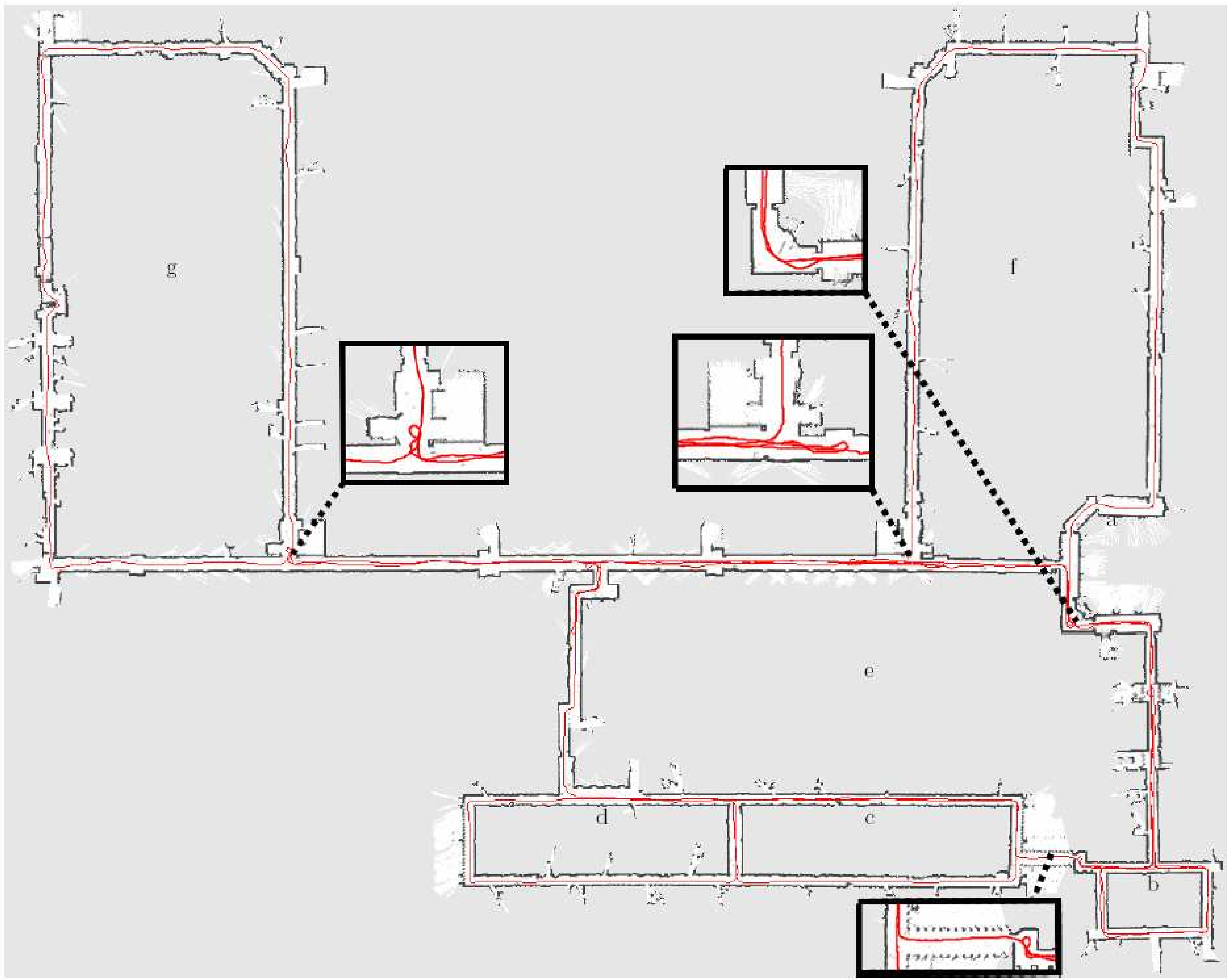


MIT Killian Court

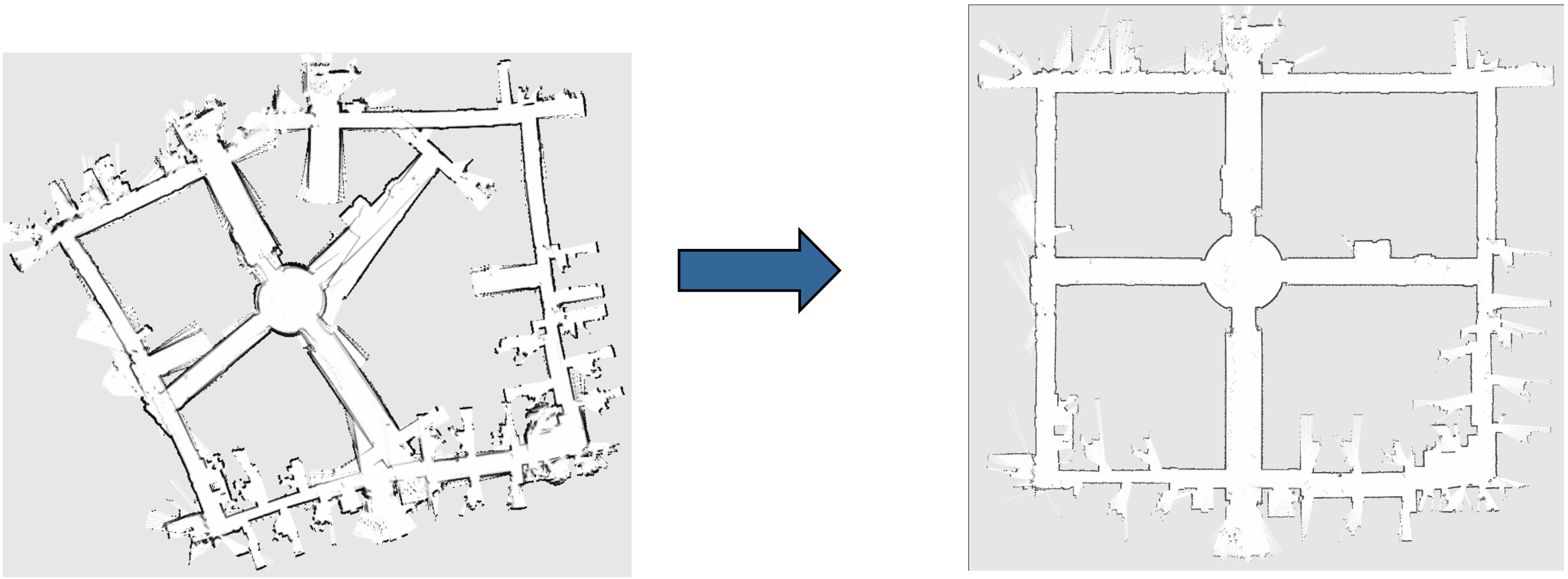


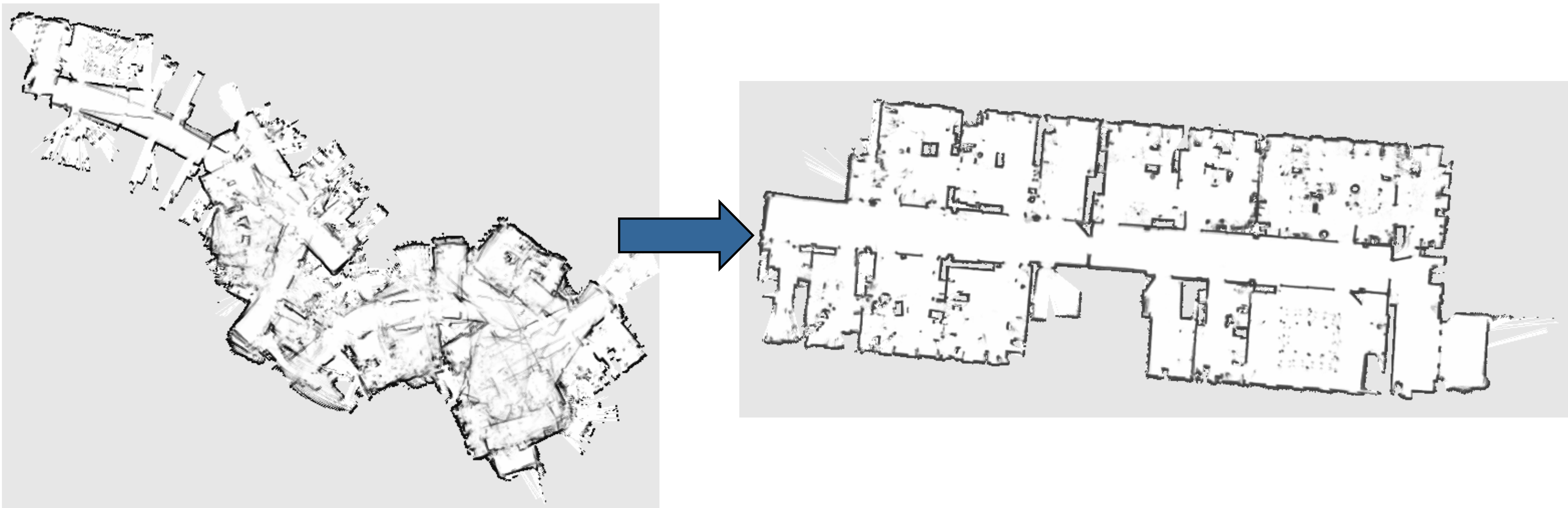
Dataset courtesy of Mike Bosse and John Leonard

MIT Killian Court



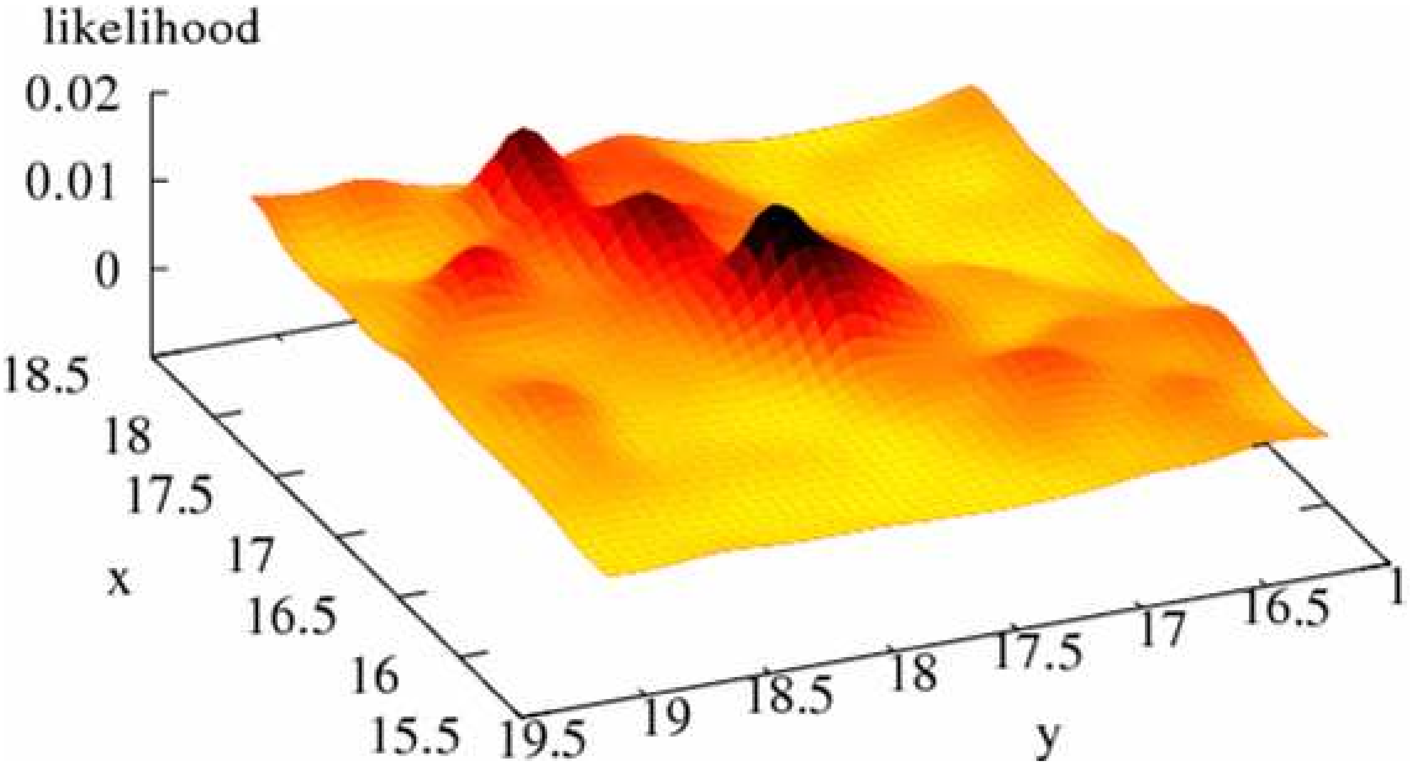
Other Example Datasets





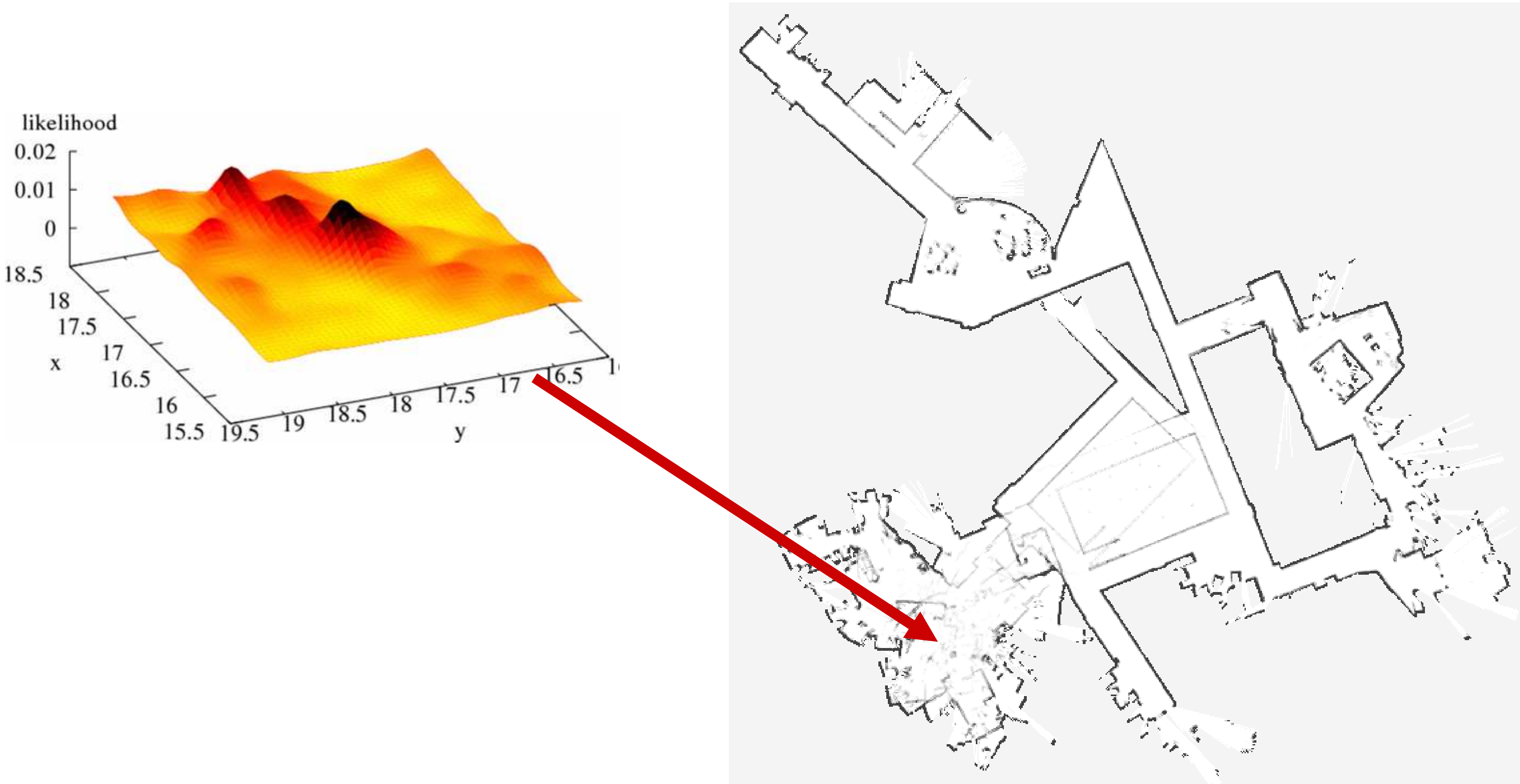
Problems of the Gaussian Proposal

- Gaussians are uni-model distributions
- In case of loop-closures, the likelihood function might be multi-modal



Problems of the Gaussian Proposal

- Multi-modal likelihood function can cause filter divergence



Use the Optimal Proposal?

$$p(x_t|x_{t-1}^{(i)}, m^{(i)}, z_t, u_t) = \frac{p(z_t|x_t, m^{(i)})p(x_t|x_{t-1}^{(i)}, u_t)}{p(z_t|x_{t-1}^{(i)}, m^{(i)})}$$

- To sample from the optimal proposal requires
  1. to point-wise evaluate it,
  2. to build a non-parametric representation (here a 3d histogram), and
  3. to sample from this
- To reduce the discretization effects of the histogram one can use a kernel smoother
- This is **possible** but very **inefficient**

The Cost of Sampling

Dataset	N	Execution time	
		optimal	Gausian proposal
MIT Killian Court	80	155 h	112 min
Freiburg Bldg. 79	30	84 h	62 min
Intel Research Lab	30	40 h	29 min
FHW Museum	30	38 h	27 min
Belgioioso	30	18 h	13 min
MIT CSAIL	30	10 h	7 min

Sampling from the optimal distribution is too expansive for real applications but it can be used for evaluation a proposal approximation.

Gaussian or Non-Gaussian?

- There exists statistical test to check whether or not sample a generated from a Gaussian:
  - Anderson-Darling test
    (based on the cumulative density function)

- How to determine the difference between the Gaussian and the optimal proposal
  - KLD
  - Cramer-von-Misses criterion

Anderson-Darling Test

AD test works for 1d Gaussians, but

$$
\begin{aligned}
p(x, y, \theta) &= p(x)p(y, \theta \mid x) \\
&= p(x)p(y \mid x)p(\theta \mid x, y)
\end{aligned}
$$

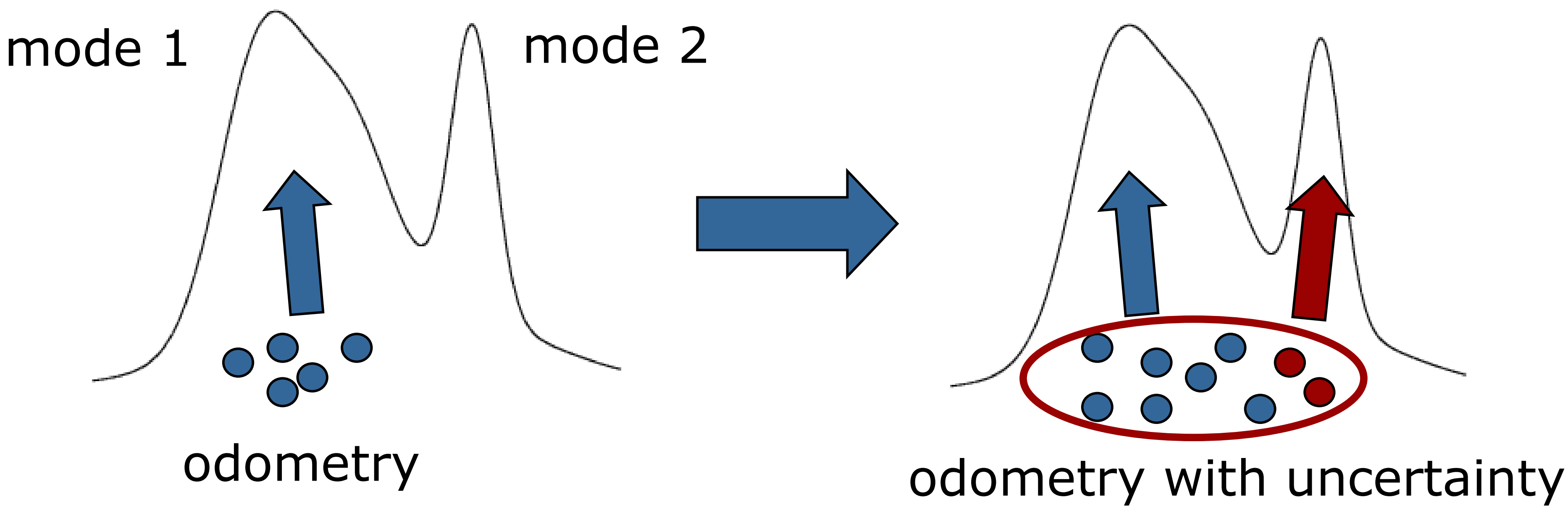
1. Sample the three terms sequentially,
2. Apply the AD test 3 times,
3. If one fails, considers it as non-Gaussian

Is a Gaussian an Accurate Representation for the Proposal?

Dataset	Gauss	Non-Gauss 1 mode	Multi-modal
Intel Research Lab	89.2%	7.2%	3.6%
FHW Museum	84.5%	10.4%	5.1%
Belgioioso	84.0%	10.4%	5.6%
MIT CSAIL	78.1%	15.9%	6.0%
MIT Killian Court	75.1%	19.1%	5.8%
Freiburg Bldg. 79	74.0%	19.4%	6.6%

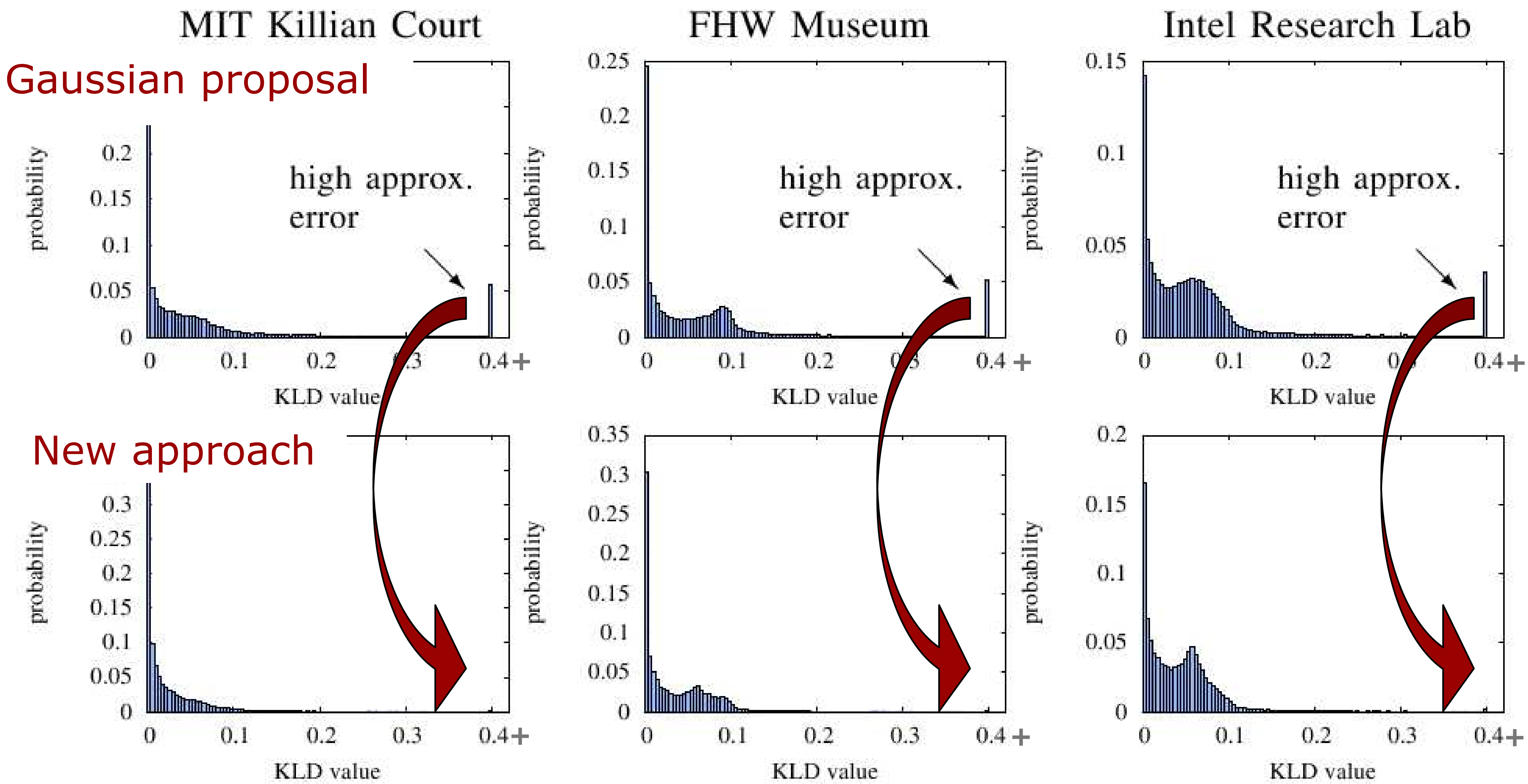
How to Overcome this Limitation?

- Approximate the likelihood in a better way!



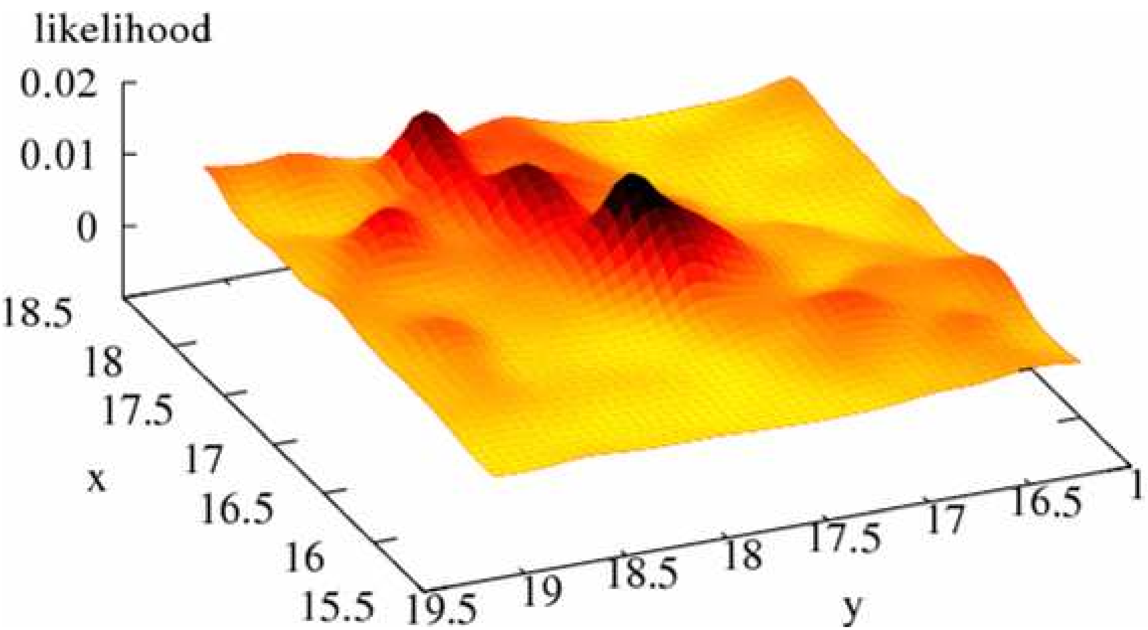
- Sample from odometry first and the use this as the start point for scan matching

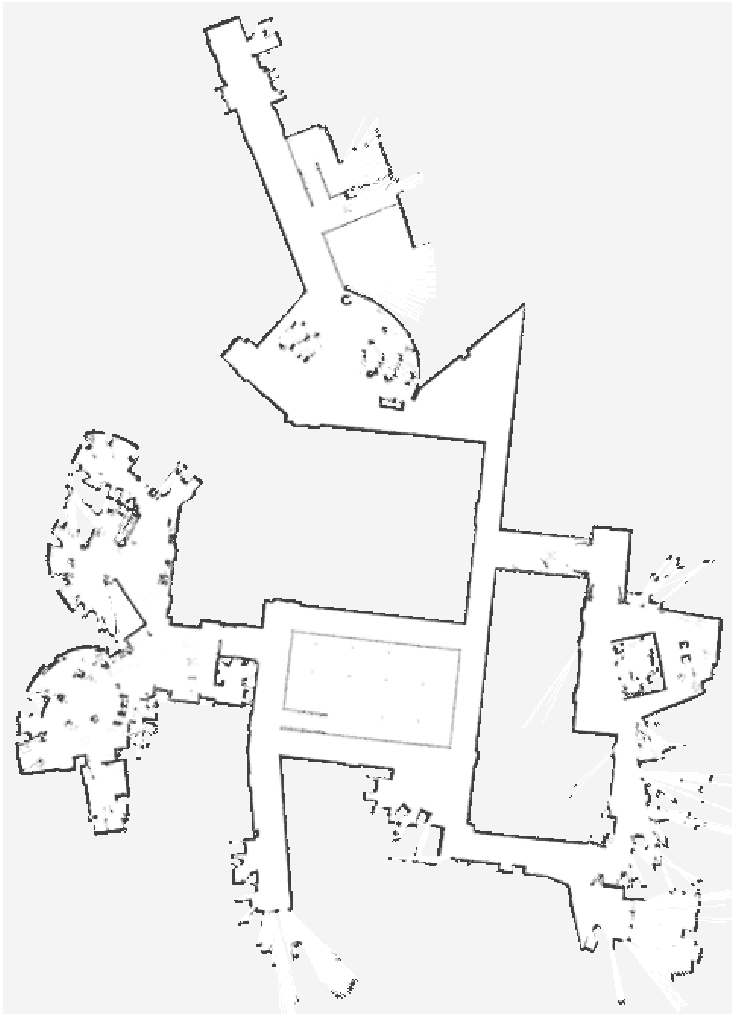
Experimental Evaluation



Final Approach

- It work’s with nearly zero overhead
- No large approximation errors (small KLD value)





Conclusion

- Rao-Blackwellized Particle Filters are means to represent a joint posterior about the poses of the robot and the map
- Utilizing accurate sensor observation leads to good proposals and highly efficient filters
- It is similar to scan-matching on a per-particle base with extra noise
- The number of necessary particles and re-sampling steps can seriously be reduced
- Possibility to deal with non-Gaussian observation likelihood functions
- Highly accurate and large scale map

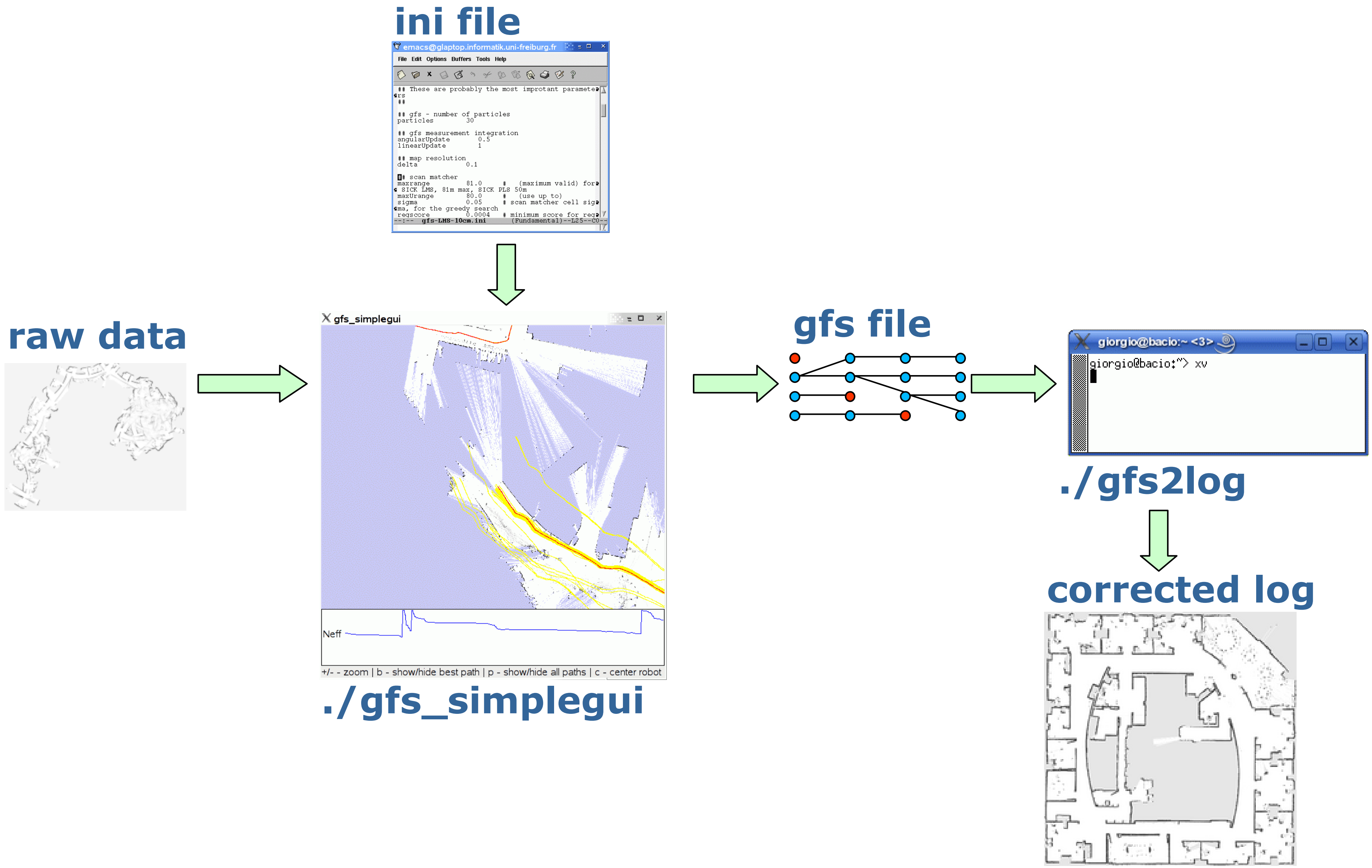
Open Source Implementation

- Open source implementation "**GMapping**" available at

  www.OpenSLAM.org

- Free for research and non-commercial applications
- Commercial licenses available

GMapping Overview



Running GMapping

- Command Line
```
gfs_simplegui
  -filename <logfile>
  -outfilename <gfsfile>
  -cfg <ini file>
  [additional parameters]
```

  Example:
```
gfs_simplegui -filename intel.clf -outfilename intel.gfs
  -cfg $GMAPPING_HOME/ini/gfs-LMS-10cm.ini
```

- The ini file specifies parameters for
  - Motion model
  - Range Finder
  - Scan Matcher
  - Likelihood
  - Particle Filter

More Details

- M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM: A factored solution to simultaneous localization and mapping, AAAI02 (The classic FastSLAM paper with landmarks)

- M. Montemerlo, S. Thrun D. Koller, and B. Wegbreit. FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges, IJCAI03. (FastSLAM 2.0 – improved proposal for FastSLAM)

- D. Haehnel, W. Burgard, D. Fox, and S. Thrun. An efficient FastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements, IROS03 (FastSLAM on grid-maps using scan-matched input)

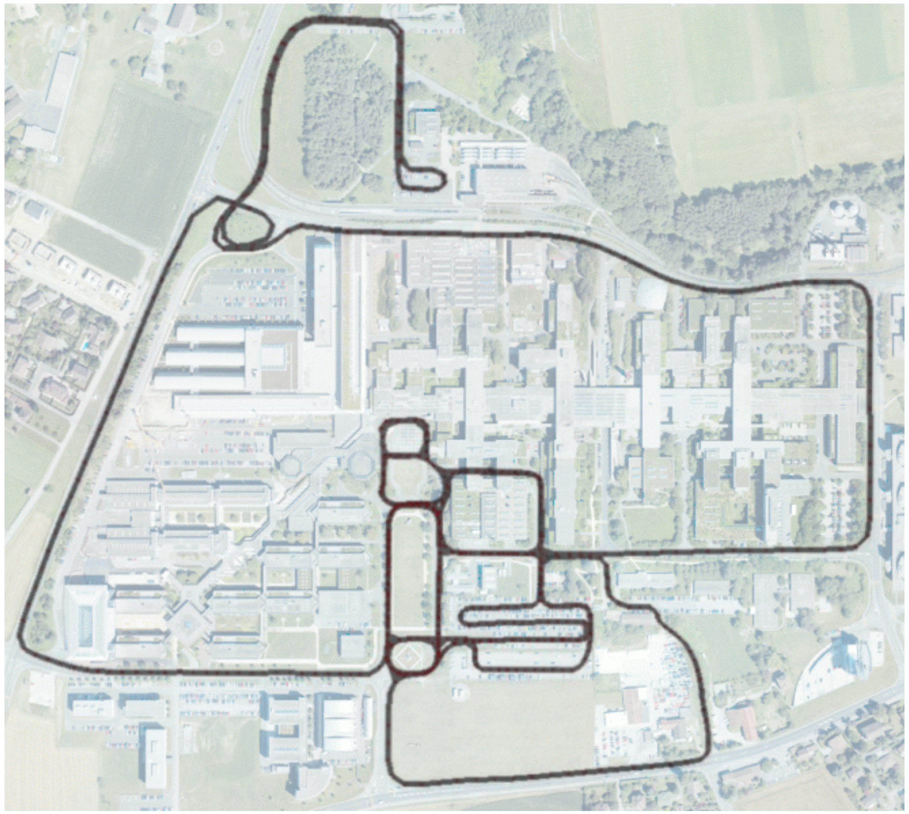
- A. Eliazar and R. Parr. DP-SLAM: Fast, robust simultaneous localization and mapping without predetermined landmarks, IJCAI03 (A representation to handle big particle sets)

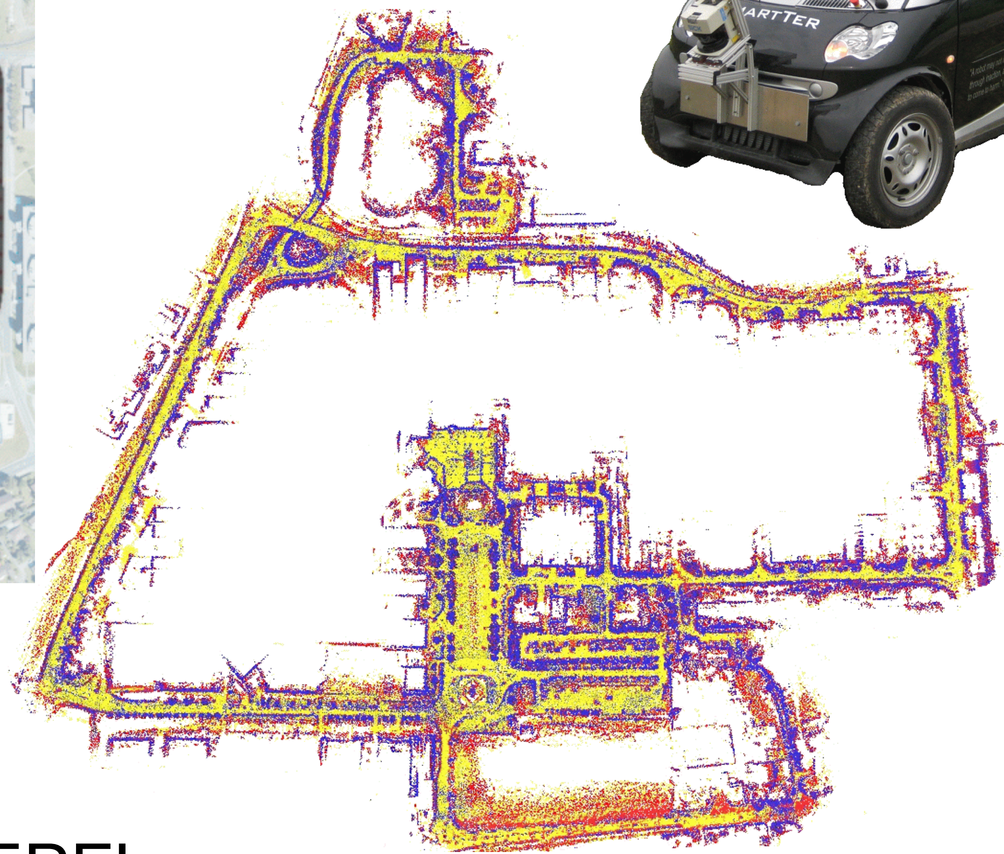
More Details (Own Work)

- Giorgio Grisetti, Cyrill Stachniss, and Wolfram Burgard. Improved Techniques for Grid Mapping with Rao-Blackwellized Particle Filters, Transactions on Robotics, Volume 23, pages 34-46, 2007
(Informed proposal using laser observation, adaptive resampling)

- Cyrill Stachniss, Grisetti Giorgio, Wolfram Burgard, and Nicholas Roy. Analyzing Gaussian Proposal Distributions for Mapping with Rao-Blackwellized Particle Filters, IROS07
(Gaussian assumption for computing the improved proposal)

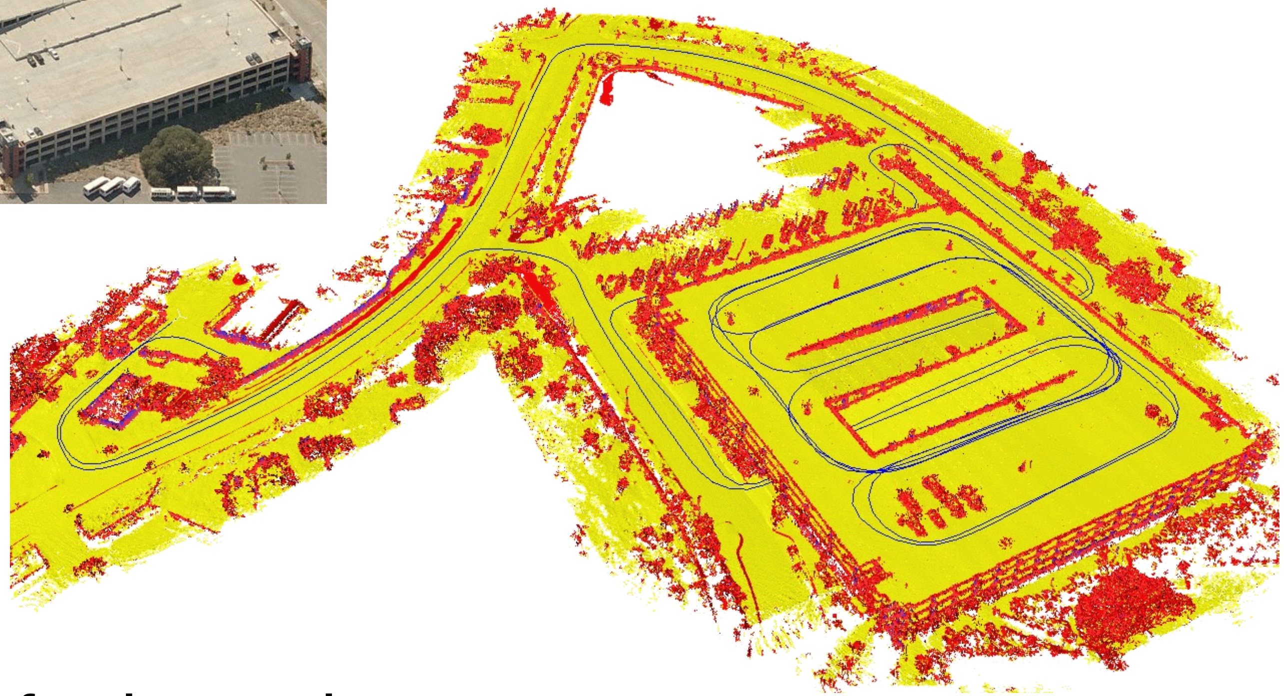
What’s Next? 3D Mapping





Joint work with ETH Zurich / EPFL

What’s Next? 3D Mapping



Joint work with the Stanford AI Lab