Principles of Safe Autonomy: Lecture 10: SLAM

Sayan Mitra

Feb 25, 2019

Reference: Probabilistic Robotics
by Sebastian Thrun, Wolfram Burgard, and Dieter Fox
Slides adapted from the book's website

Announcements

- Midterm 1 next Monday 3/4 (Includes up to 2/20, Localization)
- 3/13: Project intermediate reports due (Template posted)
- MP3 will be release today/tomorrow
- Change of dates:
  - Midterm 2: 4/24
  - Poster/Demo: 5/1 (Prizes, Dinner!)

The SLAM Problem

- SLAM stands for simultaneous localization and mapping
- The task of building a map while estimating the pose of the robot relative to this map

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

The SLAM Problem

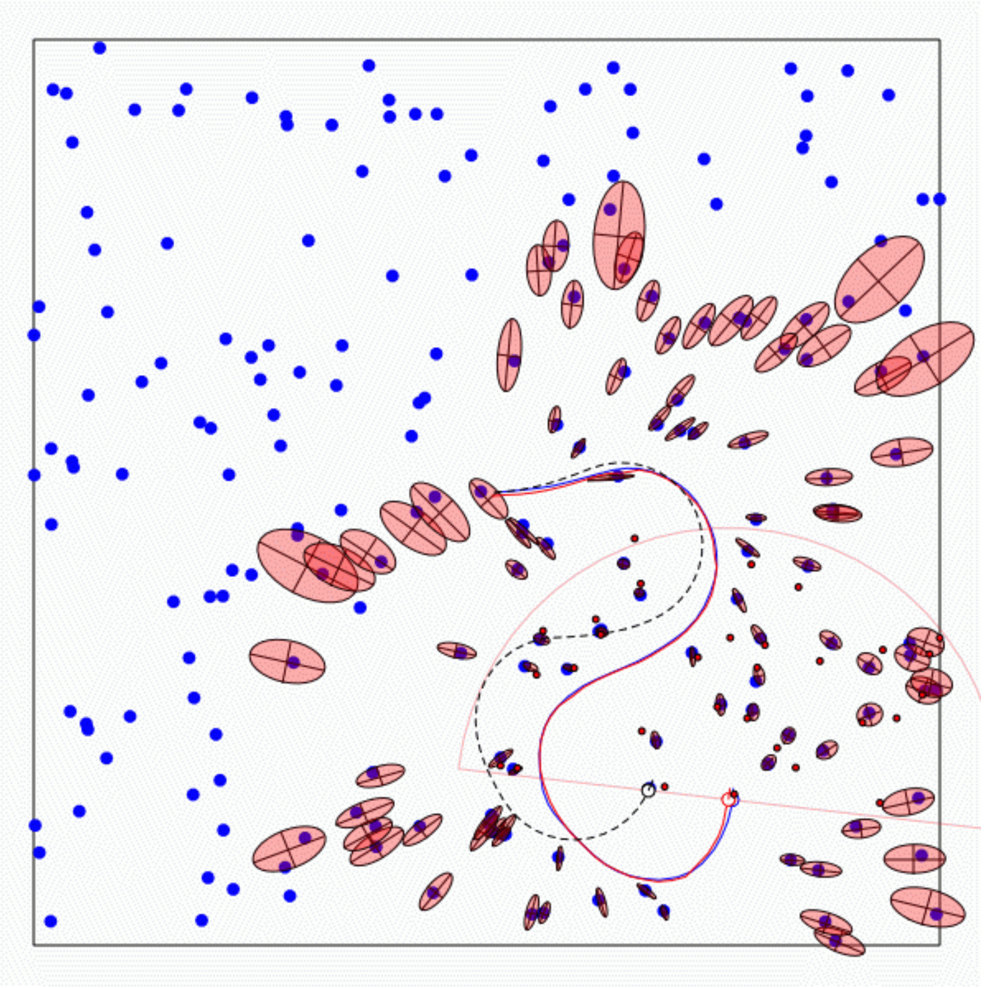
A robot moving though an unknown, static environment

Given:

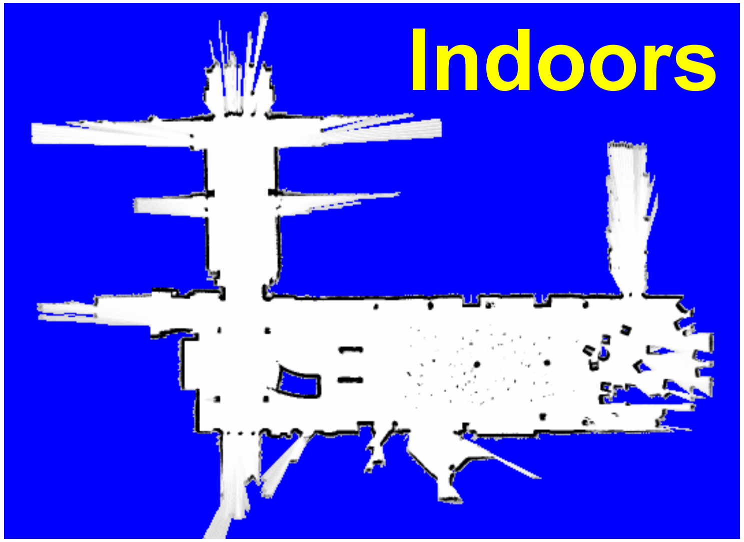
- The robot's controls
- Observations of nearby features

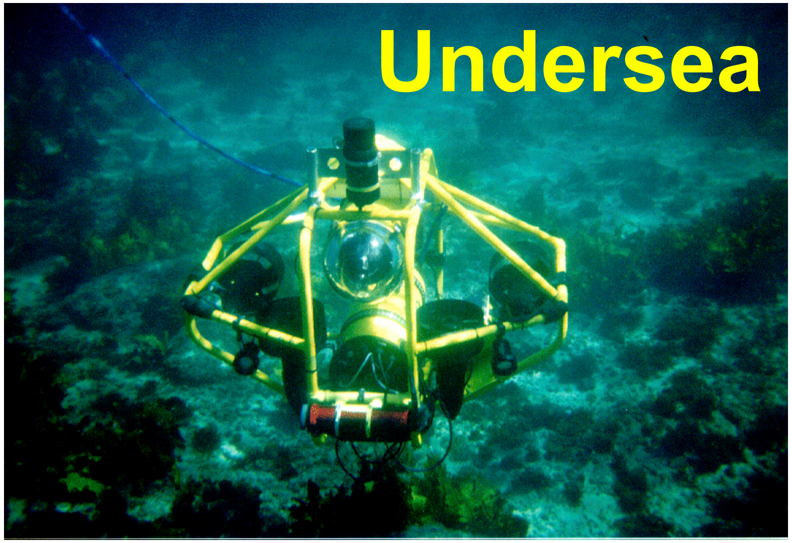
Estimate:

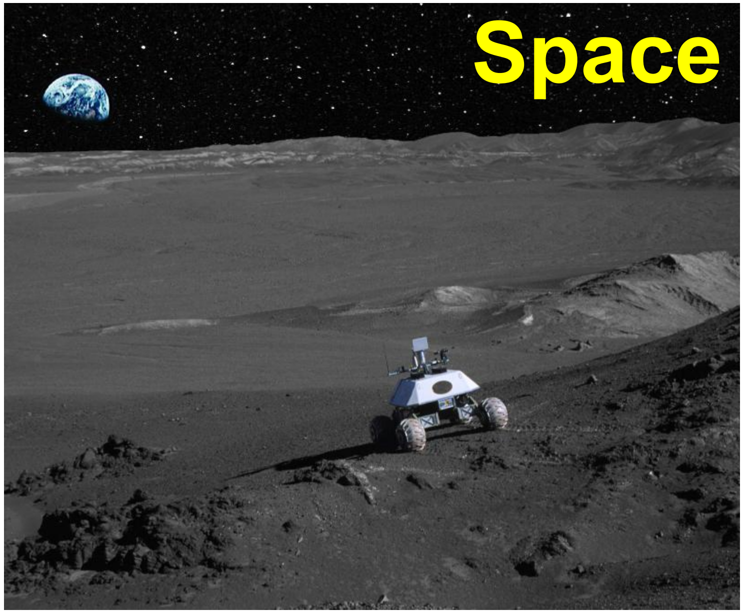
- Map of features
- Path of the robot



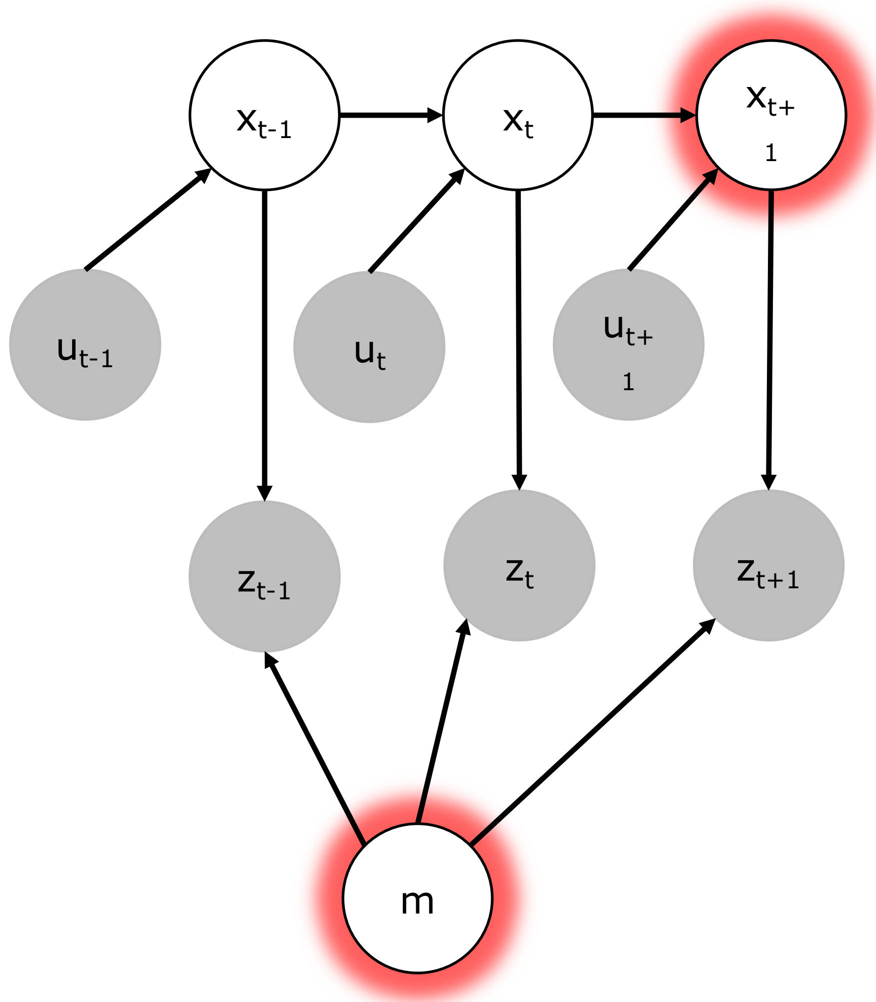
SLAM Applications







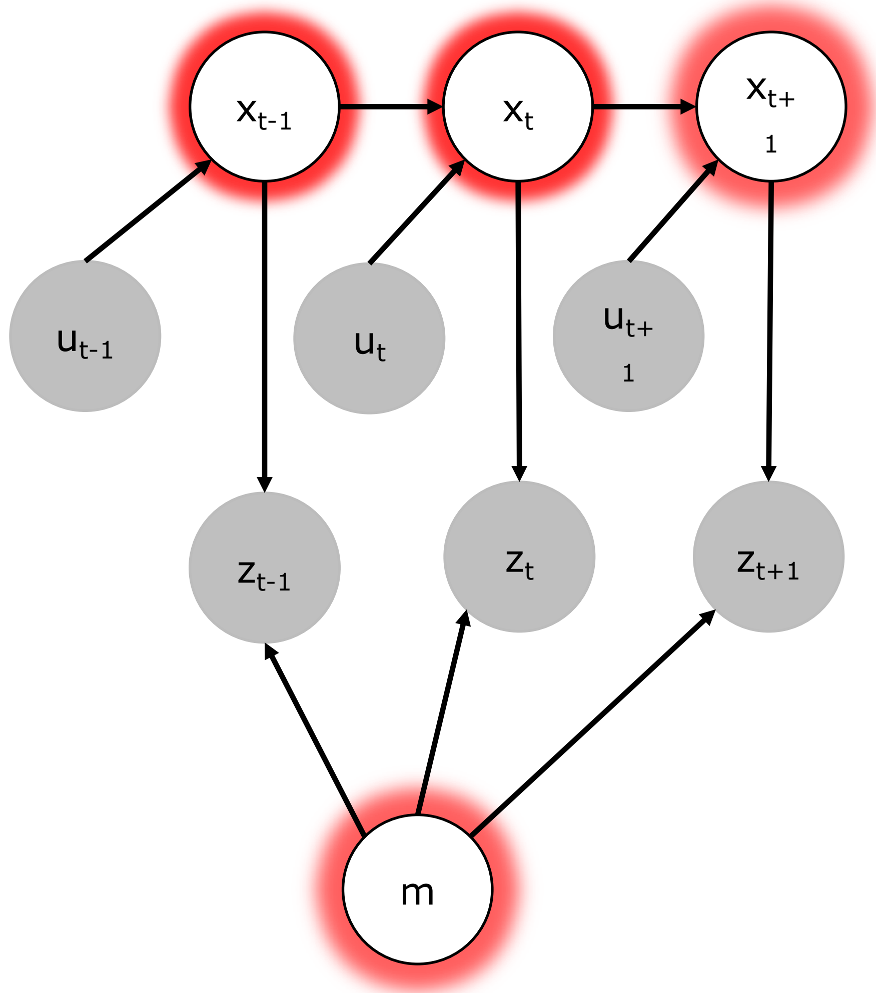
Online SLAM



Shaded known: control inputs (u), measurements(z). White nodes to be determined (x,m)

want to calculate
$p(x_t, m | z_{1:t}, u_{1:t})$

Full SLAM



Shaded known: control inputs (u), measurements(z). White nodes to be determined (x,m)

want to calculate $p(x_{1:t}, m | z_{1:t}, u_{1:t})$

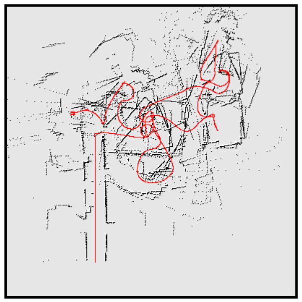
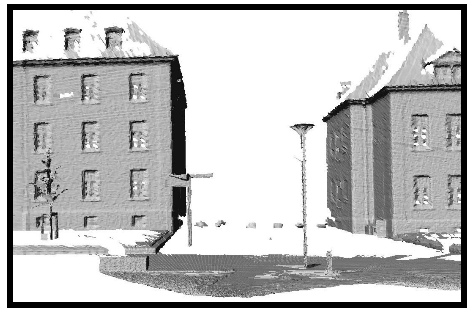
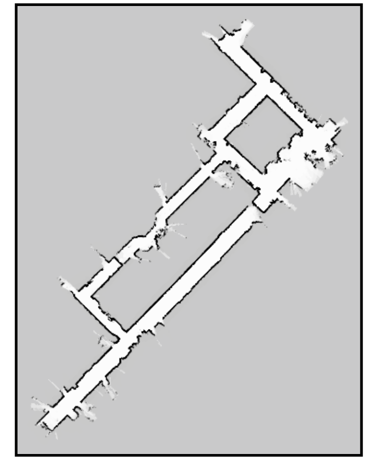
Continuous unknowns: $x_{1:t}, m$
Discrete unknowns: Relationship of detected objects to new objects

$$p(x_{1:t}, c_t, m | z_{1:t}, u_{1:t})$$

c_t: corrsnpondence variable

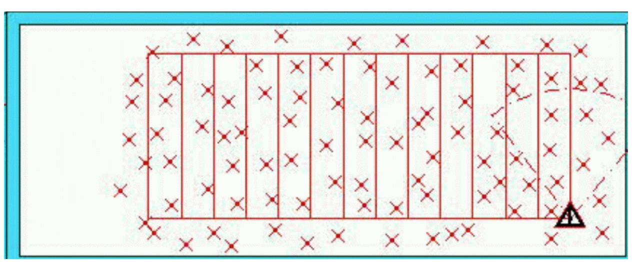
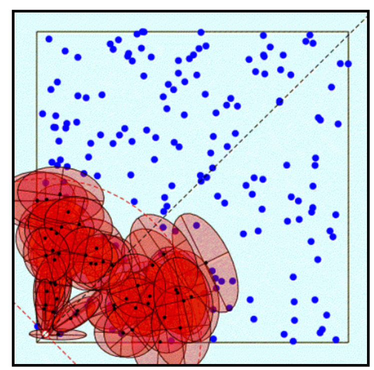
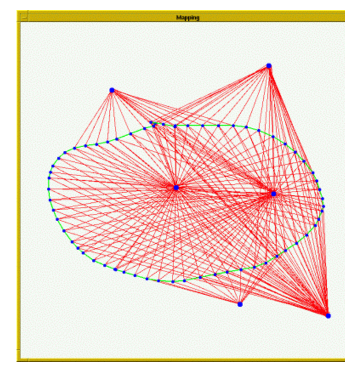
Representations

- Grid maps or scans

[Lu & Milios, 97; Gutmann, 98: Thrun 98; Burgard, 99; Konolige & Gutmann, 00; Thrun, 00; Arras, 99; Haehnel, 01;…]

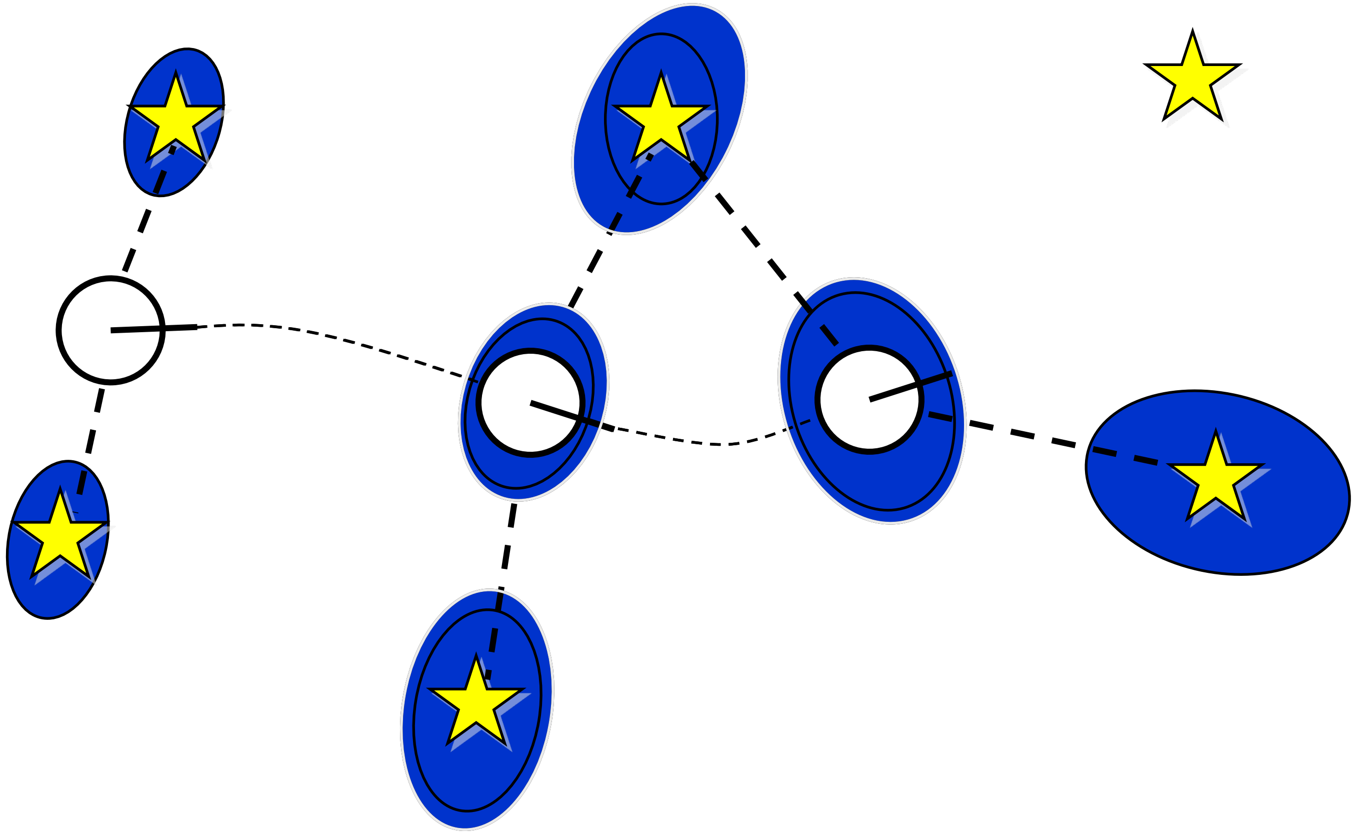
- Landmark-based

[Leonard et al., 98; Castelanos et al., 99: Dissanayake et al., 2001; Montemerlo et al., 2002;…

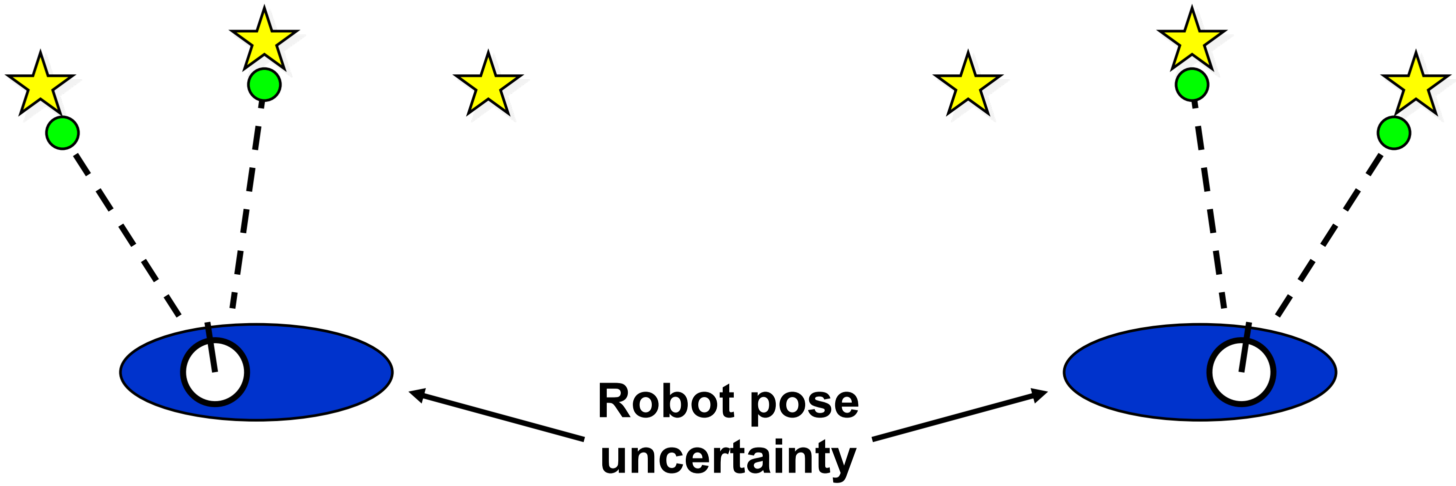
Why is SLAM a hard problem?

SLAM: robot path and map are both **unknown**



Robot path error correlates errors in the map

Why is SLAM a hard problem?



- In the real world, the mapping between observations and landmarks is unknown
- Picking wrong data associations can have catastrophic consequences
- Pose error correlates data associations

SLAM:
Simultaneous Localization and Mapping

- Full SLAM:

Estimates entire path and map!

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

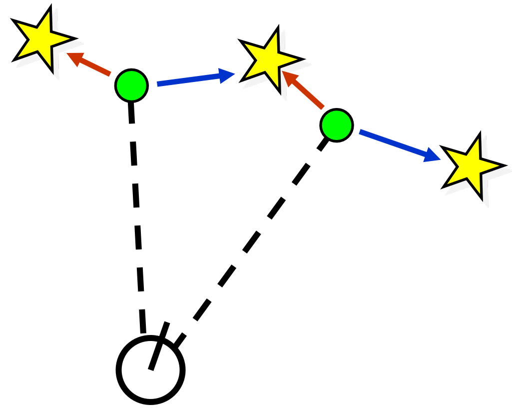
- Online SLAM:

$$p(x_t, m \mid z_{1:t}, u_{1:t}) = \int\int\ldots\int p(x_{1:t}, m \mid z_{1:t}, u_{1:t})\, dx_1 dx_2 \ldots dx_{t-1}$$

Integrations typically done one at a time

Estimates most recent pose and map!

Data Association Problem



- A data association is an assignment of observations to landmarks
- In general there are more than $\binom{n}{m}$ (n observations, m landmarks) possible associations
- Also called “assignment problem”

Particle Filters

- Represent belief by random samples
- Estimation of non-Gaussian, nonlinear processes

- Sampling Importance Resampling (SIR) principle
  - Draw the new generation of particles
  - Assign an importance weight to each particle
  - Resampling

- Typical application scenarios are tracking, localization, …

Localization vs. SLAM

- A particle filter can be used to solve both problems
- Localization: state space $<x, y, \theta>$
- SLAM: state space $<x, y, \theta, map>$
  - for landmark maps = $<l_1, l_2, \ldots, l_m>$
  - for grid maps = $<c_{11}, c_{12}, \ldots, c_{1n}, c_{21}, \ldots, c_{nm}>$
- **Problem:** The number of particles needed to represent a posterior grows exponentially with the dimension of the state space!

- Naïve implementation of particle filters to SLAM will be crushed by the curse of dimensionality

Dependencies

- Is there a dependency between the dimensions of the state space?
- If so, can we use the dependency to solve the problem more efficiently?

Dependencies

- Is there a dependency between the dimensions of the state space?
- If so, can we use the dependency to solve the problem more efficiently?

- In the SLAM context
  - The map depends on the poses of the robot.
  - We know how to build a map given the position of the sensor is known.

Conditional Independence

- A and B are conditionally independent given C if $P(A, B \mid C) = P(A|C)\, P(B|C)$

- Height and vocabulary are not independent
- But they are conditionally independent given age

Factored Posterior (Landmarks)

poses map observations & movements

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

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

SLAM posterior

Robot path posterior

landmark positions

Does this help to solve the problem?

Factorization first introduced by Murphy in 1999

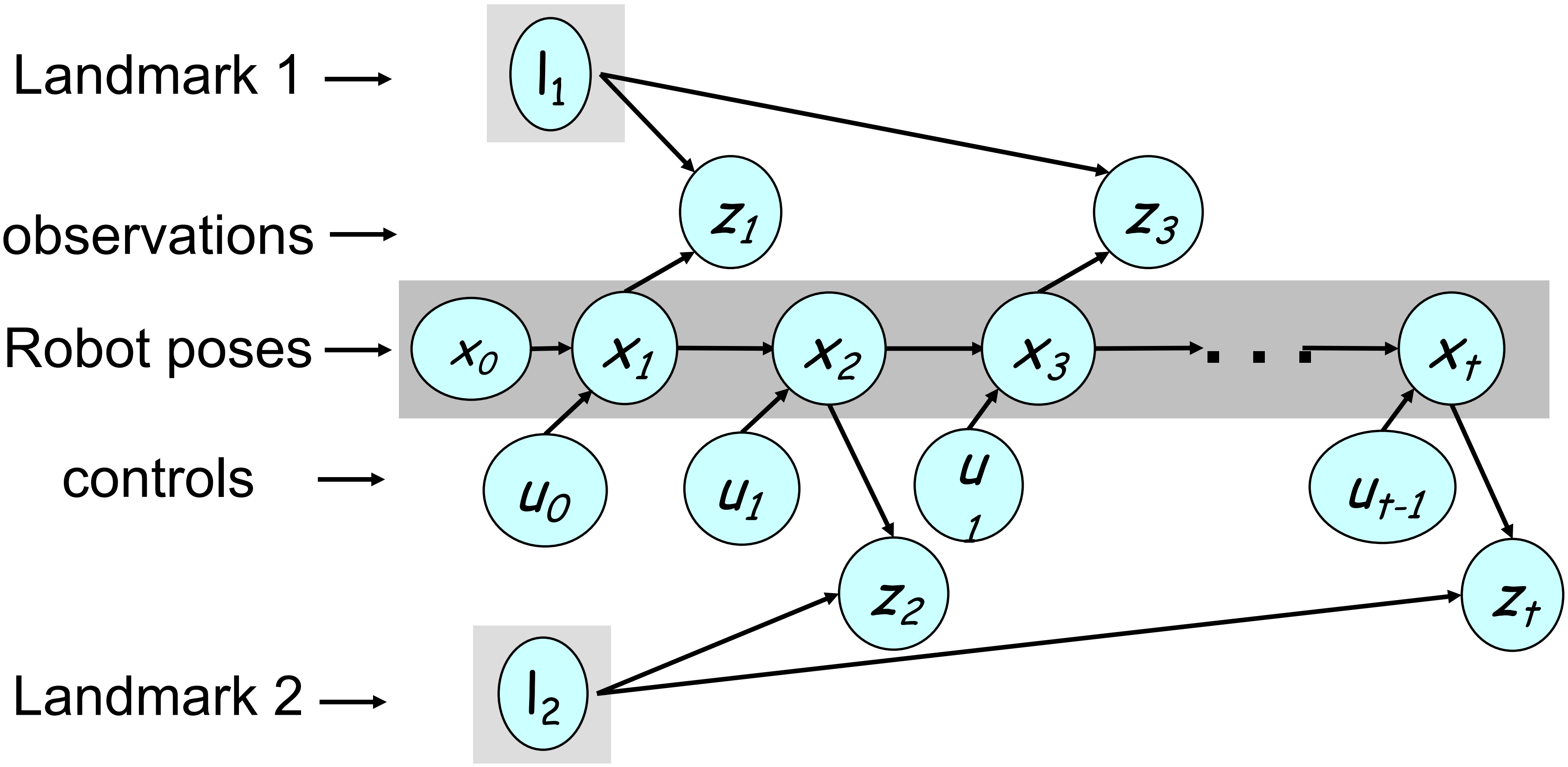
Factored Posterior (Landmarks)

poses map observations & movements

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

Factorization first introduced by Murphy in 1999

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

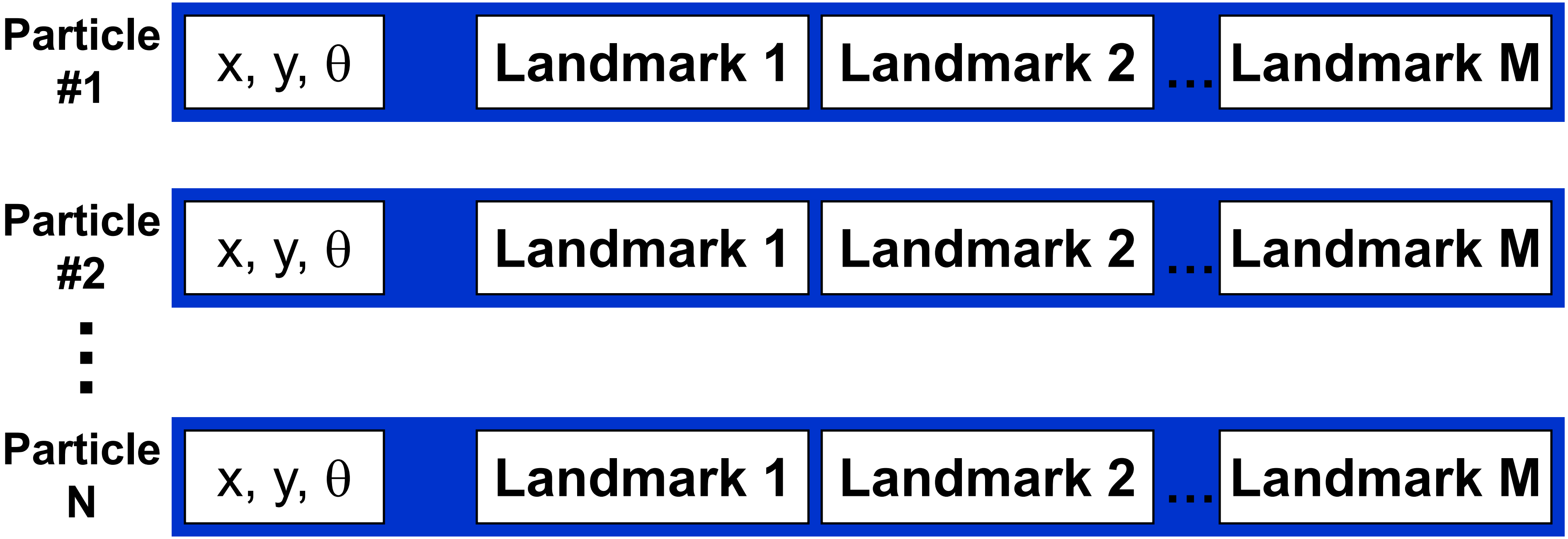
Rao-Blackwellization

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

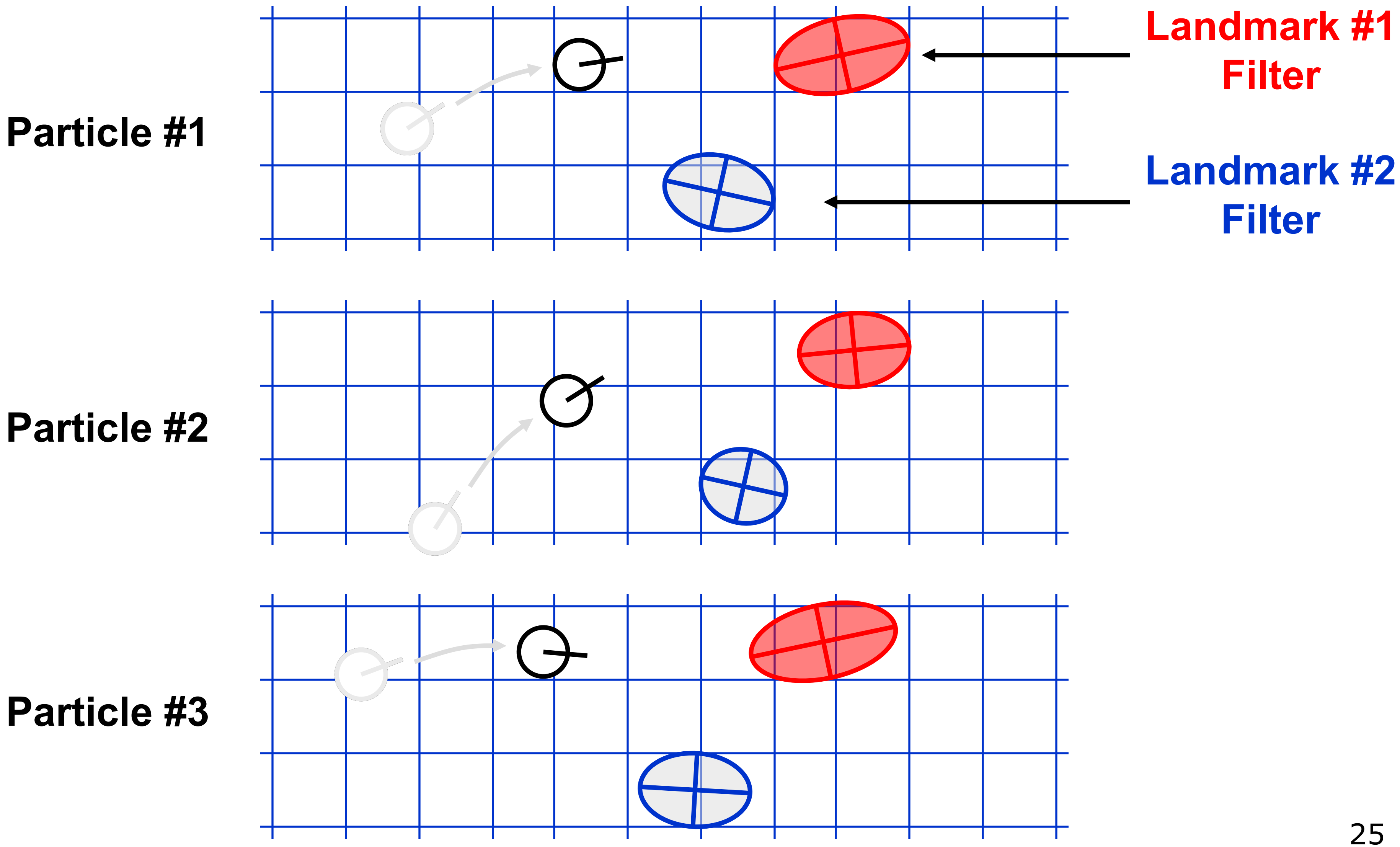
- This factorization is also called Rao-Blackwellization
- Given that the second term can be computed efficiently, particle filtering becomes possible!

FastSLAM

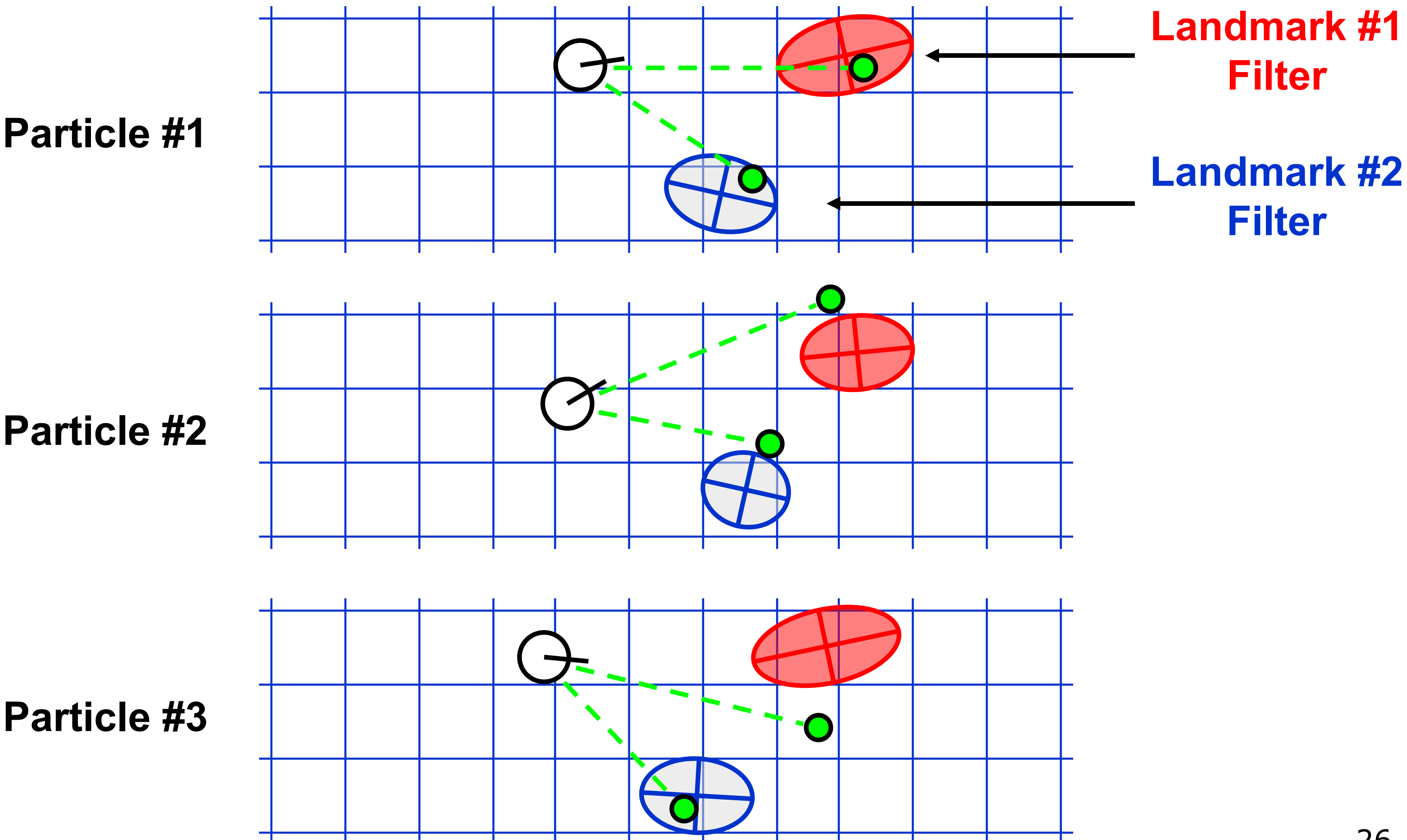
- Rao-Blackwellized particle filtering based on landmarks [Montemerlo et al., 2002]
- Each landmark is represented by a 2x2 Extended Kalman Filter (EKF)
- Each particle therefore has to maintain *M* EKFs



FastSLAM – Action Update

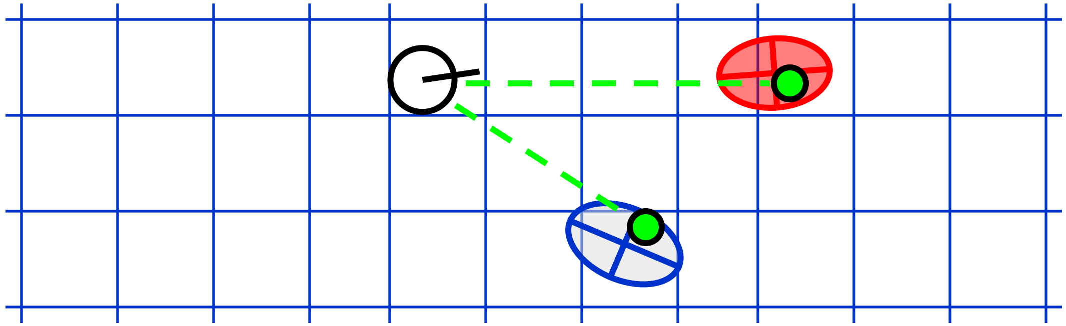


FastSLAM – Sensor Update



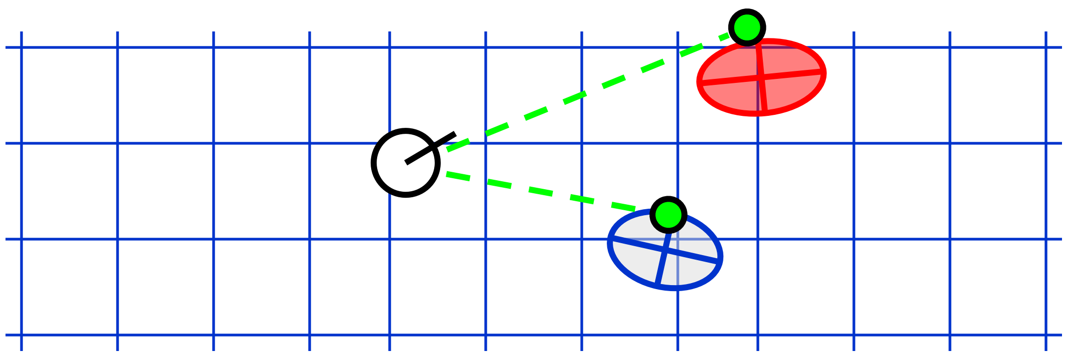
FastSLAM – Sensor Update

Particle #1



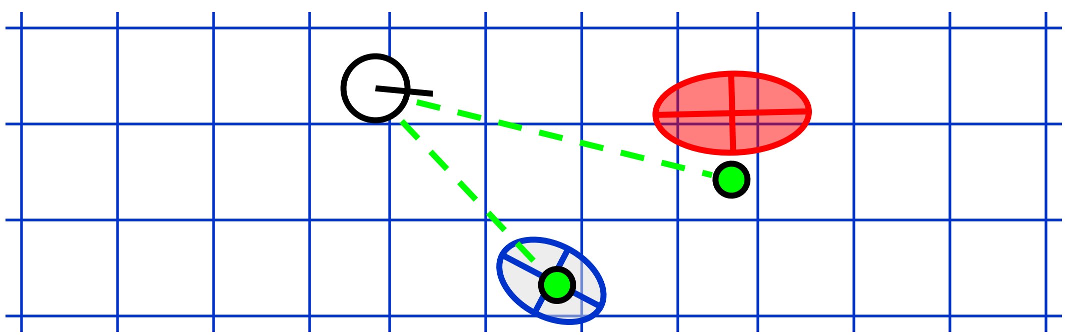
Weight = 0.8

Particle #2



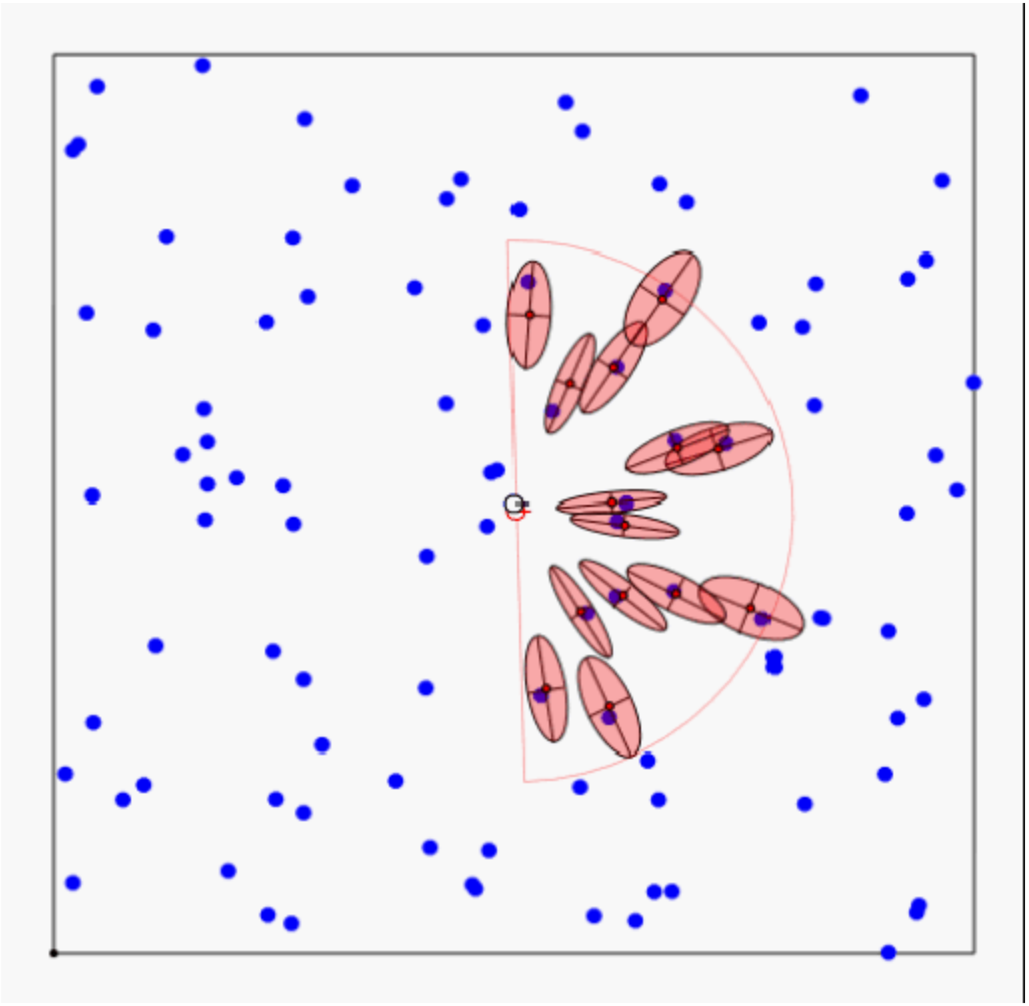
Weight = 0.4

Particle #3



Weight = 0.1

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

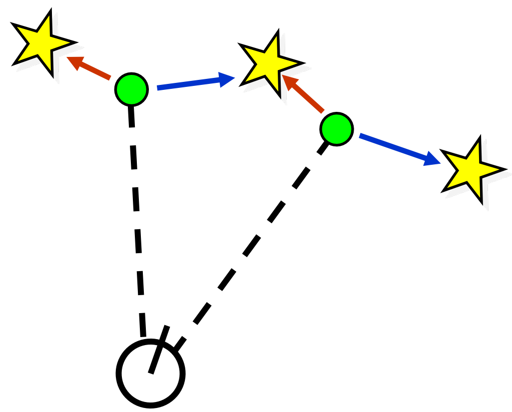
$O(N \cdot \log(M))$
Log time per particle

N = Number of particles
M = Number of map features

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

Data Association Problem

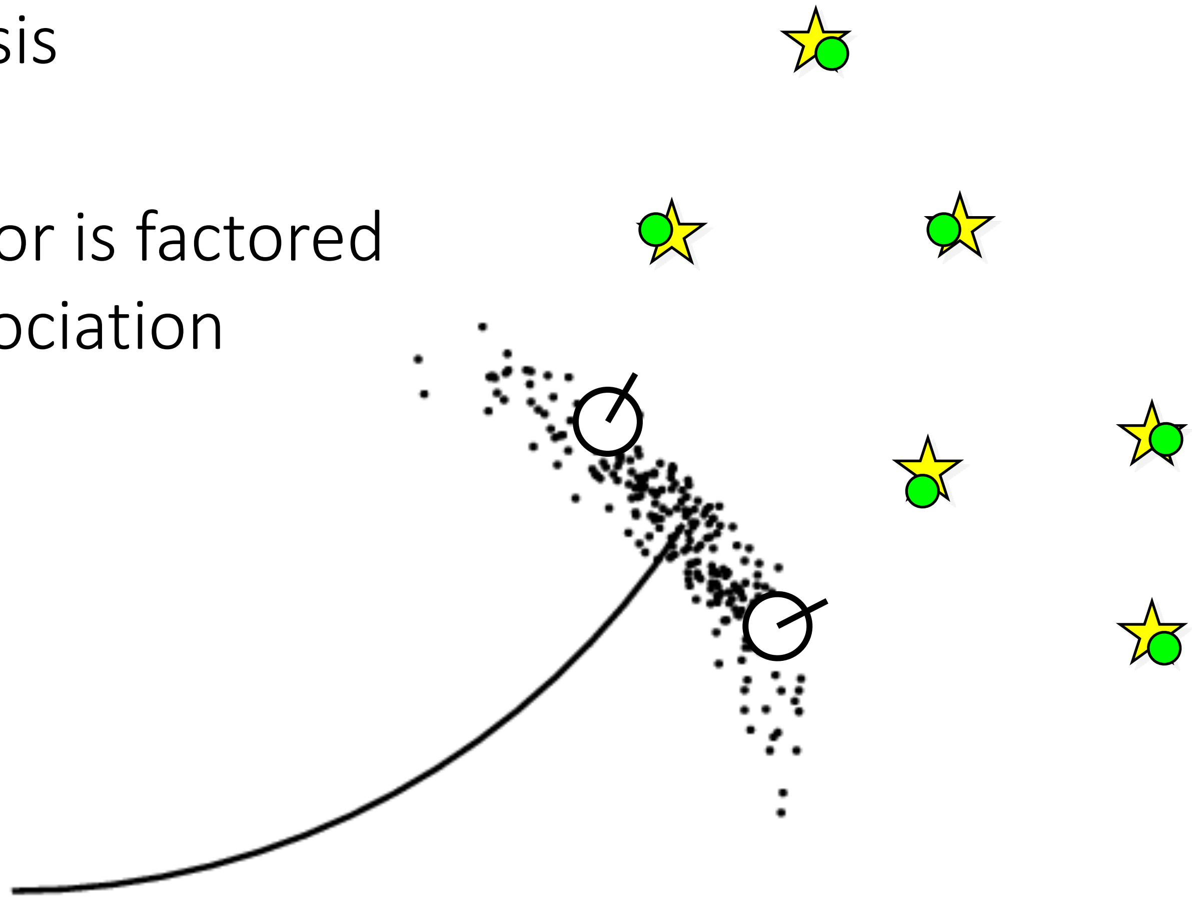
- Which observation belongs to which landmark?



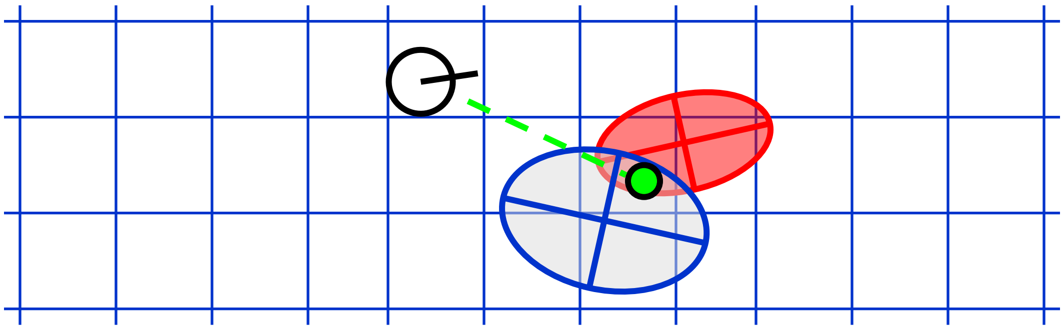
- A robust SLAM must consider possible data associations
- Potential data associations depend also on the pose of the robot

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



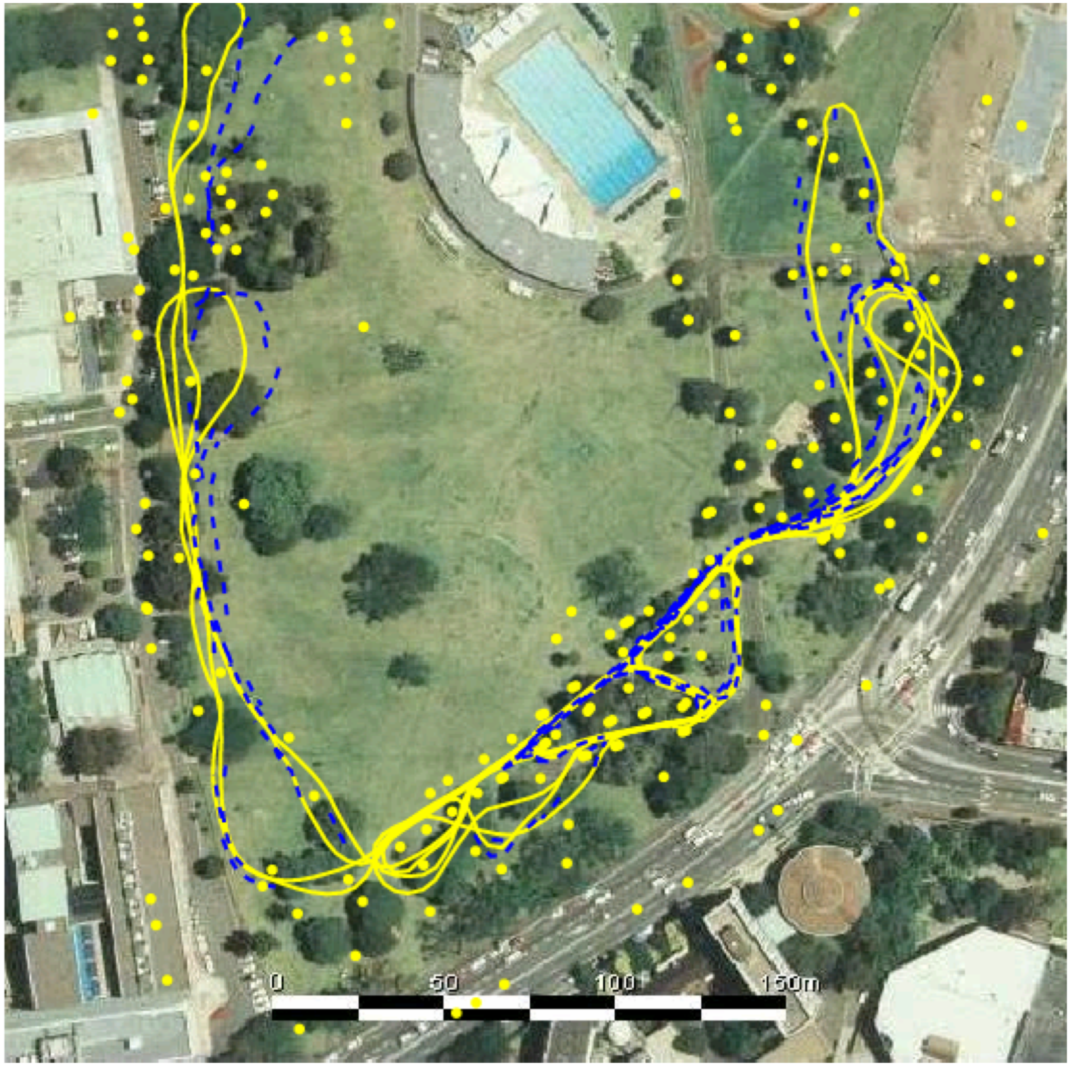
Was the observation generated by the red or the blue landmark?

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

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



Blue = GPS
Yellow = FastSLAM

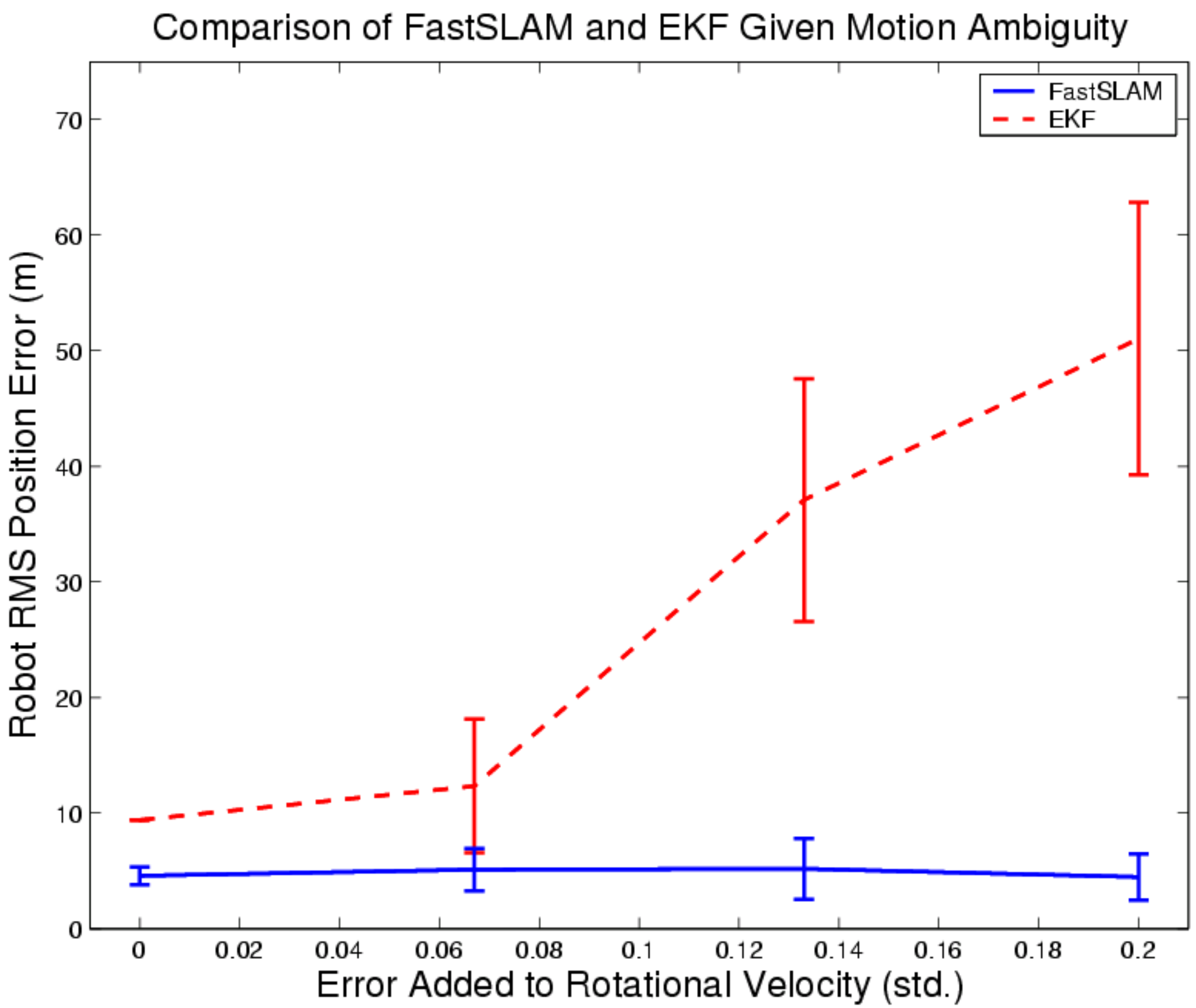
Dataset courtesy of University of Sydney

Results – Victoria Park

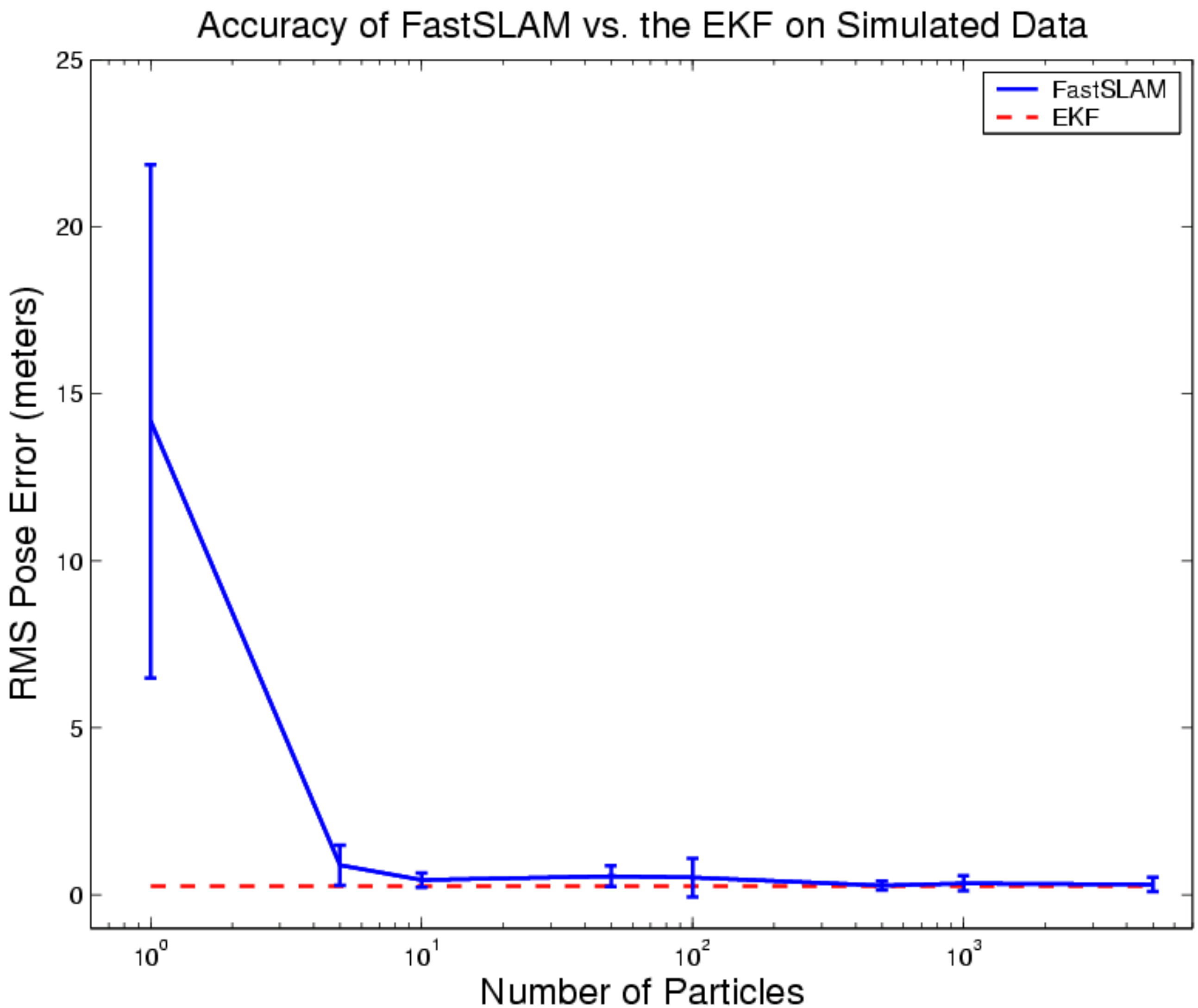
https://www.youtube.com/watch?v=BIOJSNHYSbc

Dataset courtesy of University of Sydney

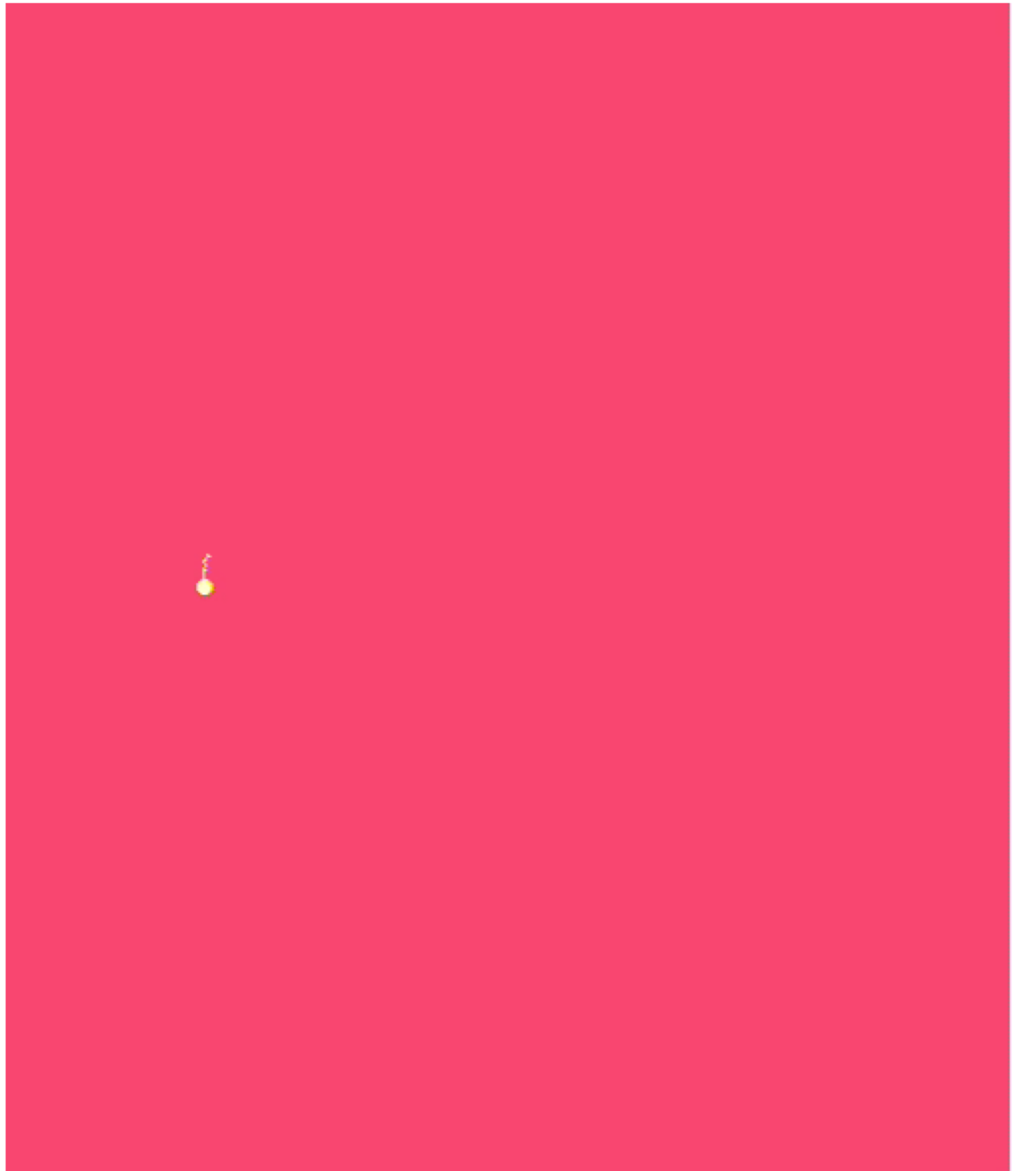
Results – Data Association



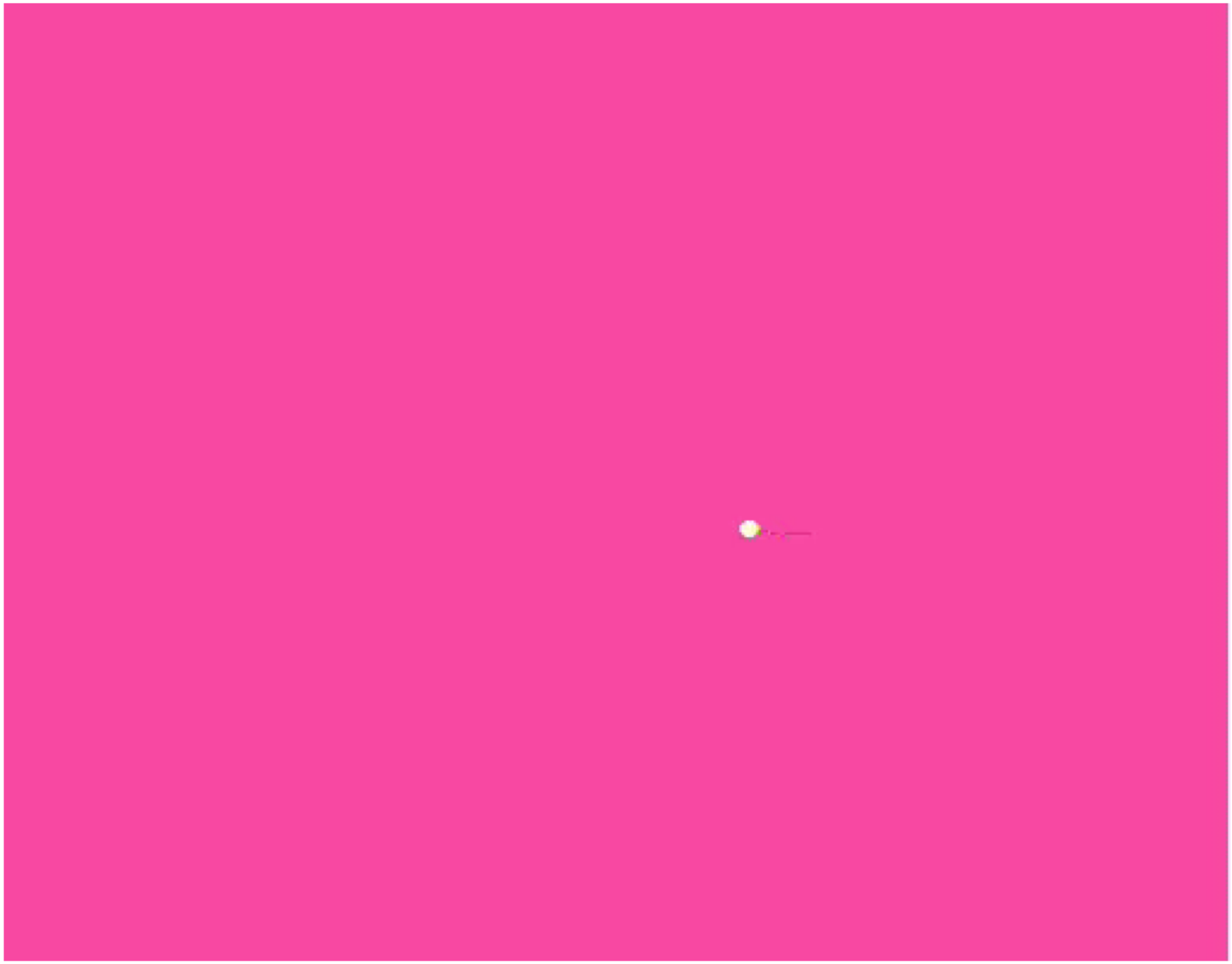
Results – Accuracy



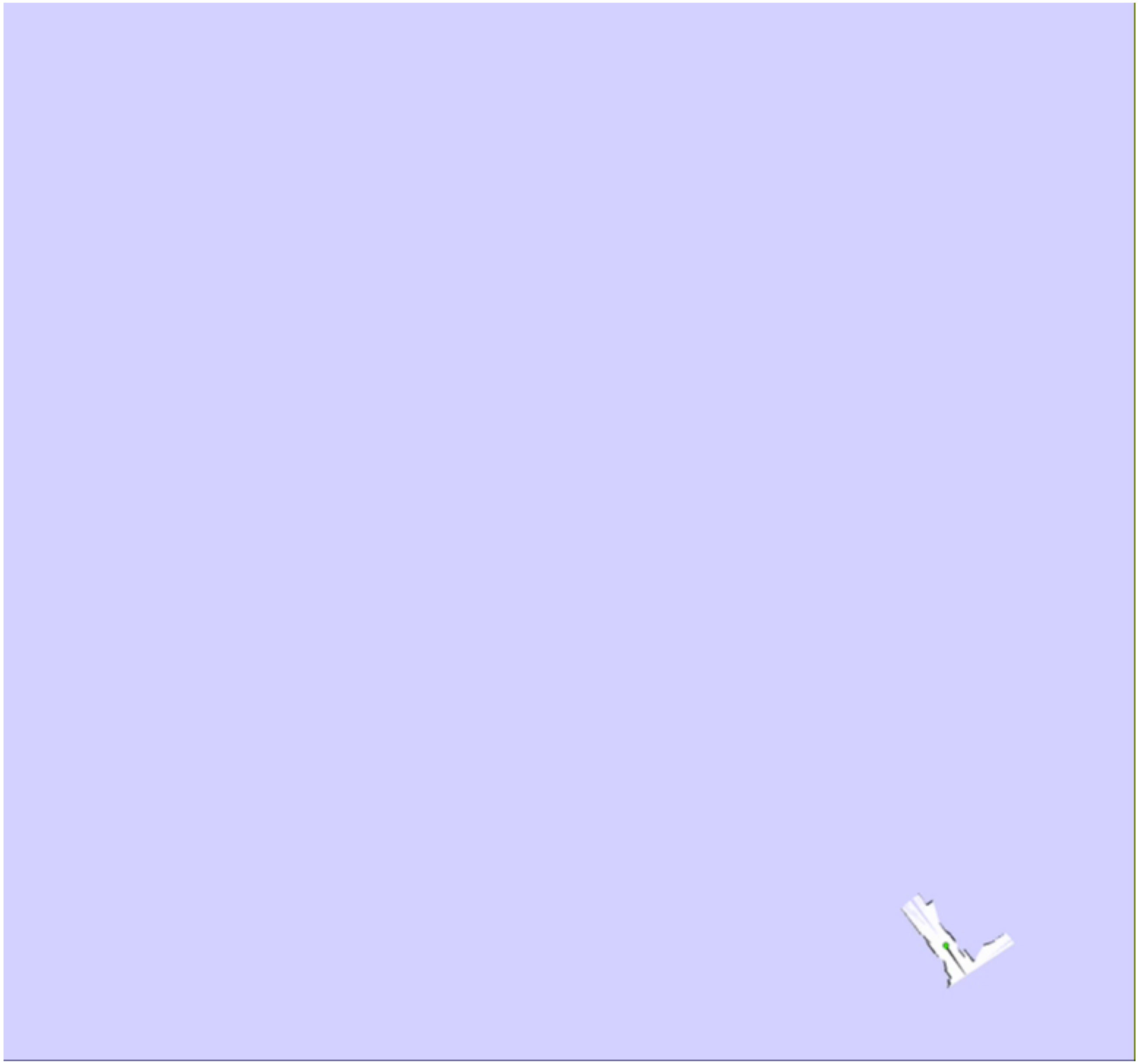
FastSLAM with Scan-Matching



FastSLAM with Scan-Matching



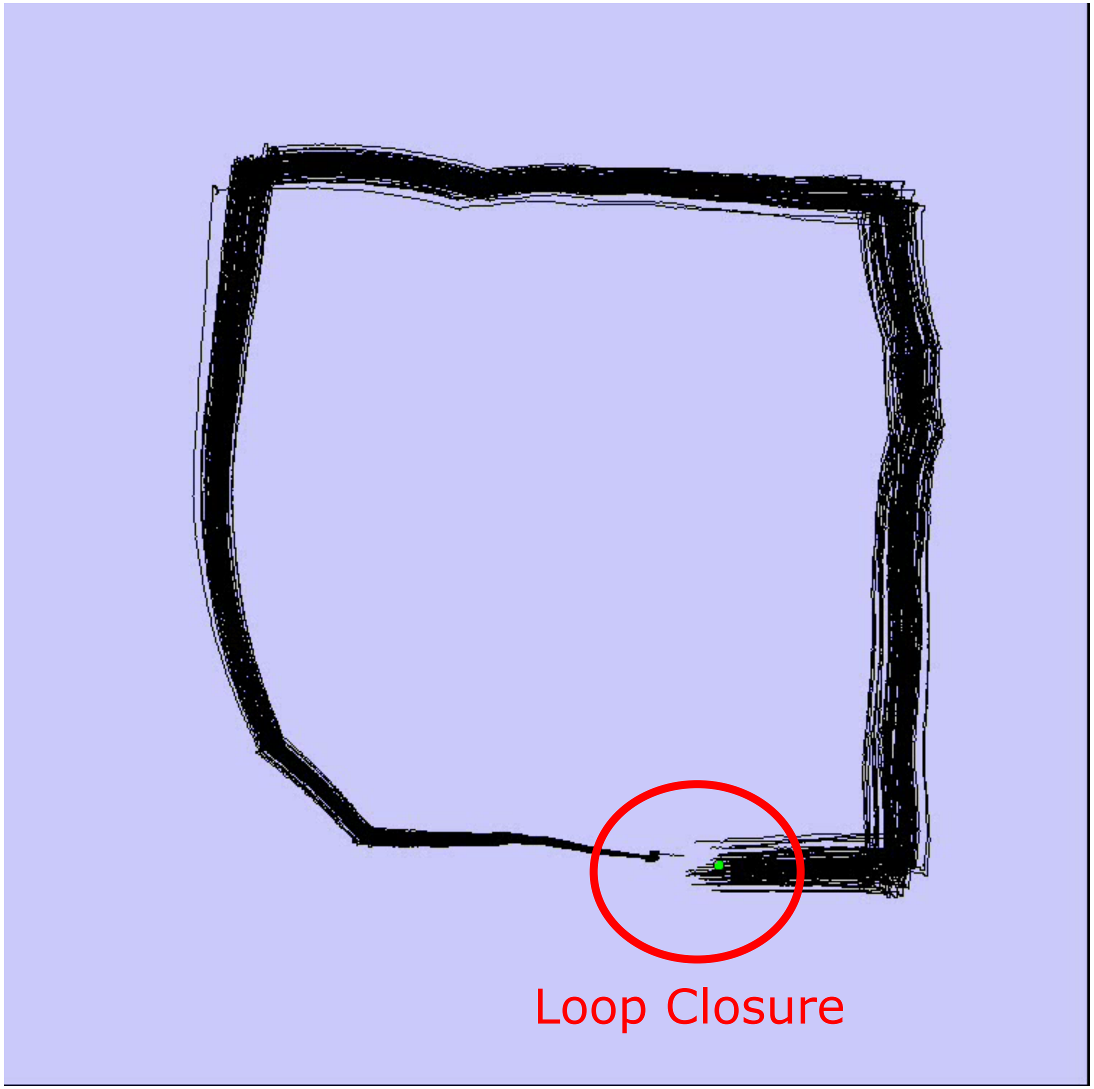
FastSLAM with Scan-Matching



Grid-based SLAM

- Can we solve the SLAM problem if no pre-defined landmarks are available?
- Can we use the ideas of FastSLAM to build grid maps?
- As with landmarks, the map depends on the poses of the robot during data acquisition
- If the poses are known, grid-based mapping is easy ("mapping with known poses")

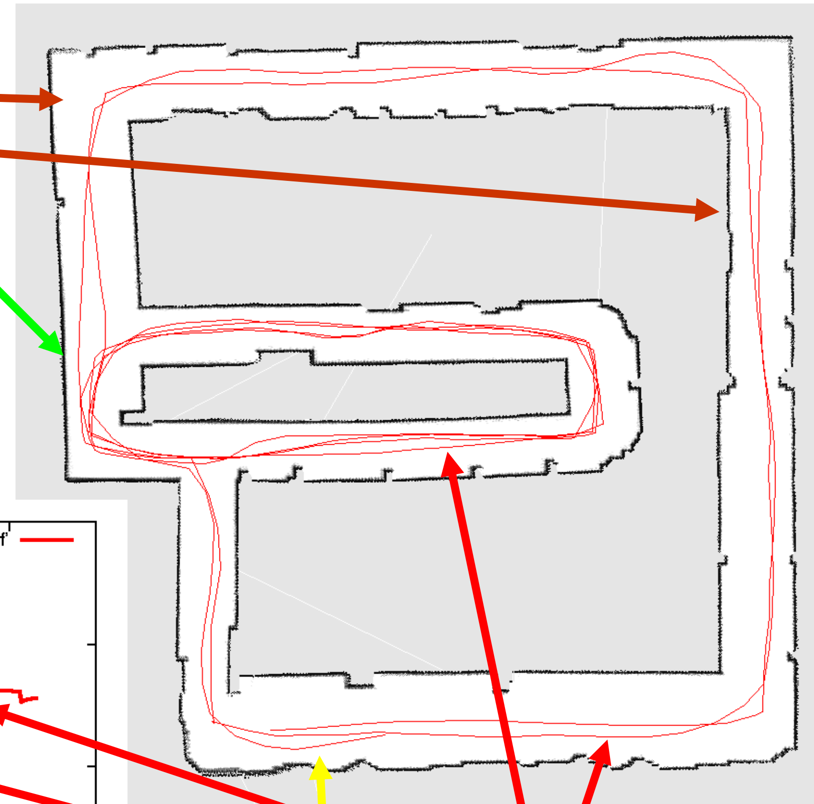
FastSLAM with Scan-Matching

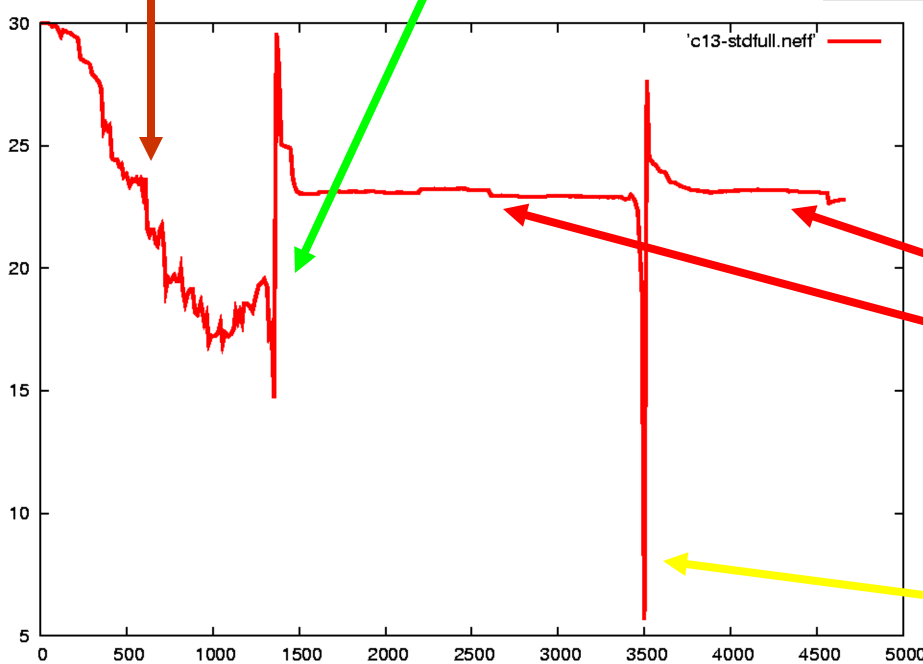


Typical Evolution of n_{eff}

visiting new areas

closing the first loop





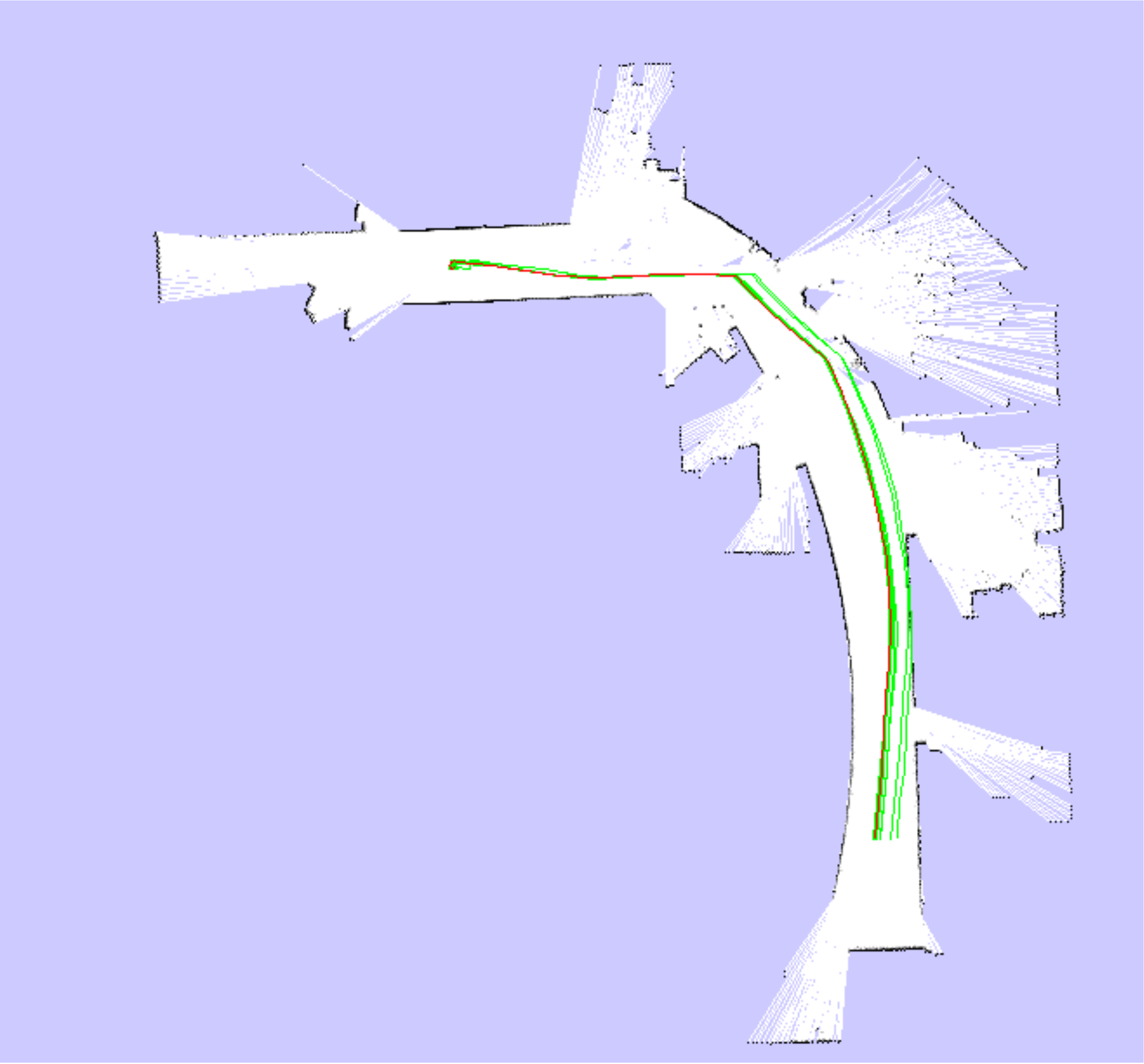
visiting known areas

second loop closure

Intel Lab

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

Intel Lab



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

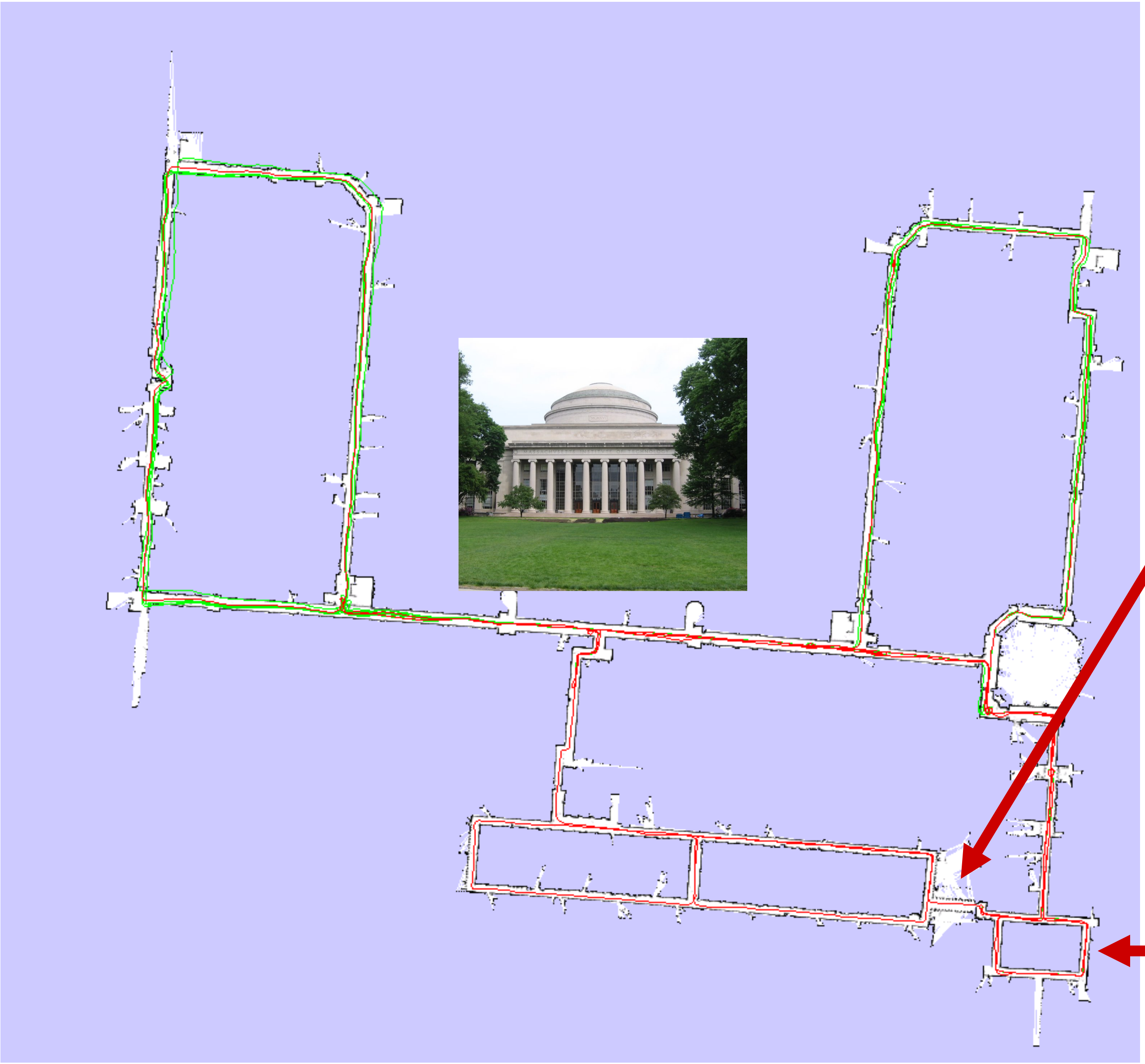
Outdoor Campus Map

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

MIT Killian Court

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

MIT Killian Court



More Details on FastSLAM

- M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM: A factored solution to simultaneous localization and mapping, *AAAI02*
- 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
- M. Montemerlo, S. Thrun, D. Koller, B. Wegbreit. FastSLAM 2.0: An Improved particle filtering algorithm for simultaneous localization and mapping that provably converges. IJCAI-2003
- G. Grisetti, C. Stachniss, and W. Burgard. Improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling, ICRA05
- A. Eliazar and R. Parr. DP-SLAM: Fast, robust simultanous localization and mapping without predetermined landmarks, IJCAI03