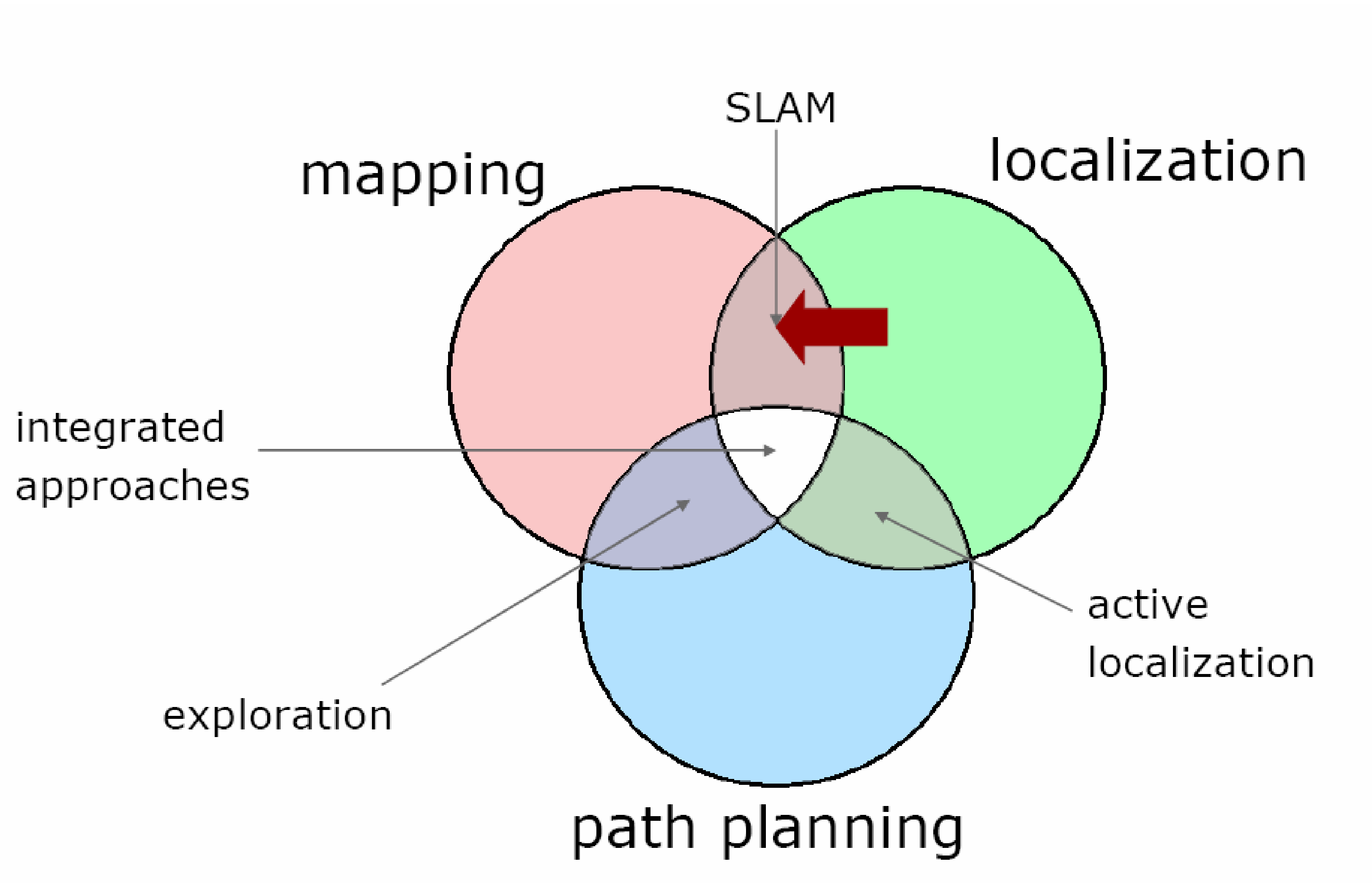
Simultaneous localization and mapping

Václav Hlaváč
Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Cybernetics
Czech Republic
http://cmp.felk.cvut.cz/~hlavac

Courtesy to several authors of presentations on the web.

Where does SLAM fit?



SLAM – task formulation

- **Inputs**:
  - Time sequence of proprioceptive and exteroceptive measurements made as robot through an initially unknown environment.
  - No external coordinate reference.
- **Outputs**:
  - A map of the robot environment.
  - A robot pose estimate associated with each measurement in the coordinate system of the map.

SLAM is an incremental task

- **State/Output**:
  - Map of the environment, which has been observed so far.
  - Robot pose estimate w.r.t. map.
- **Action/Input**:
  - Move to a new position/orientation.
  - Acquire additional observations.
- **Update state**:
  - Re-estimate robot's pose.
  - Revise the map appropriately.

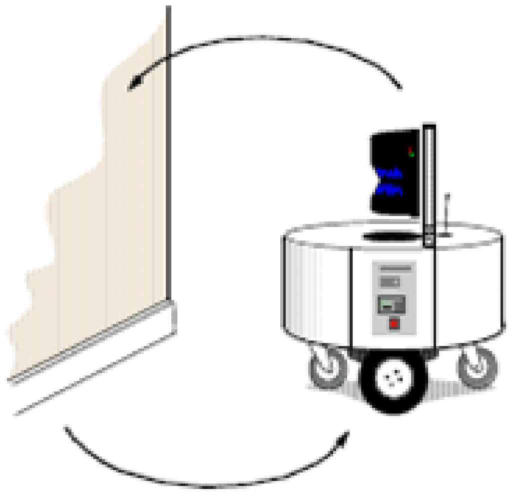
SLAM problem 1

- **Localization**: inferring location given a map.
- **Mapping**: inferring a map given a location.
- **SLAM**: learning a map and locating the robot simultaneously.

- **SLAM** is the process by which a robot builds a map of the environment and, at the same time, uses this map to compute its location.

SLAM Problem 2

- SLAM is a chicken or egg problem.
  - A map is needed for localizing a robot.
  - A good robot position estimate is needed to create/update the map.
- Consequently, SLAM is regarded as hard problem in robotics.



SLAM problem 3

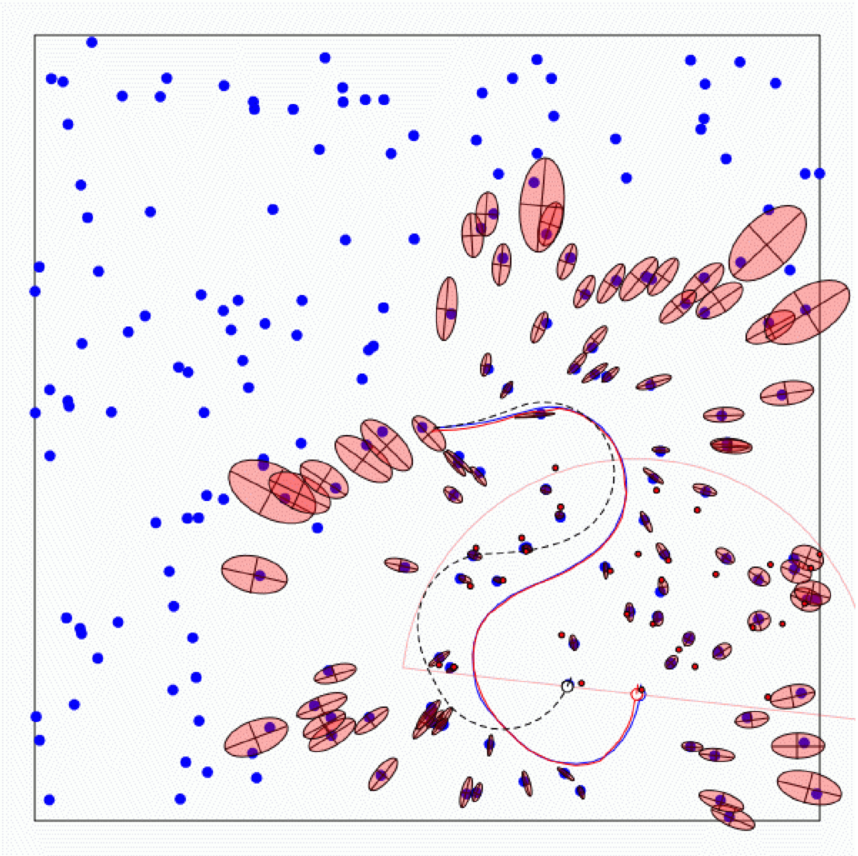
- SLAM is considered one of the most fundamental problems for (mobile) robots to be truly autonomous.
- Variety of approaches have been tried to approach SLAM problem.
- Probabilistic methods rule!
- History of SLAM dates to mid-1980s.

Why is SLAM hard?

- Chicken or egg problem.
- Many ingredients:
  - Autonomous, persistent, collaborative robots.
  - Mapping is multi-scale in generic environments.
- Map-making ~ learning:
  - Difficult also for humans.
  - Humans make mapping mistakes.
- Scaling issues:
  - Large spatial extent $\Rightarrow$ combinatorial expansion.
  - Persistent autonomous operations.
- Uncertainty at every level of the problem.

The SLAM Problem

A robot is exploring an unknown, static environment.



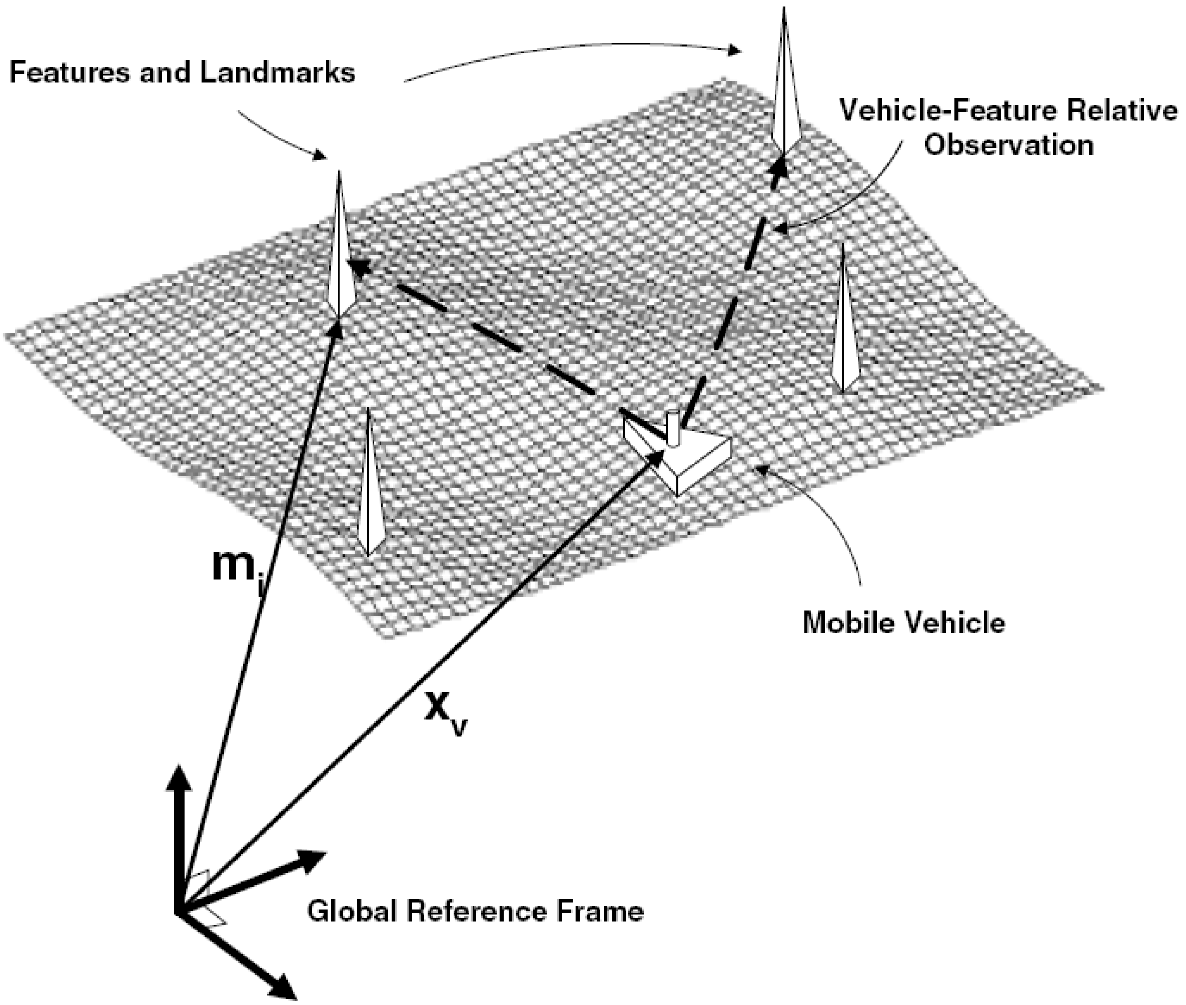
Given:

- The robot's controls
- Observations of nearby features

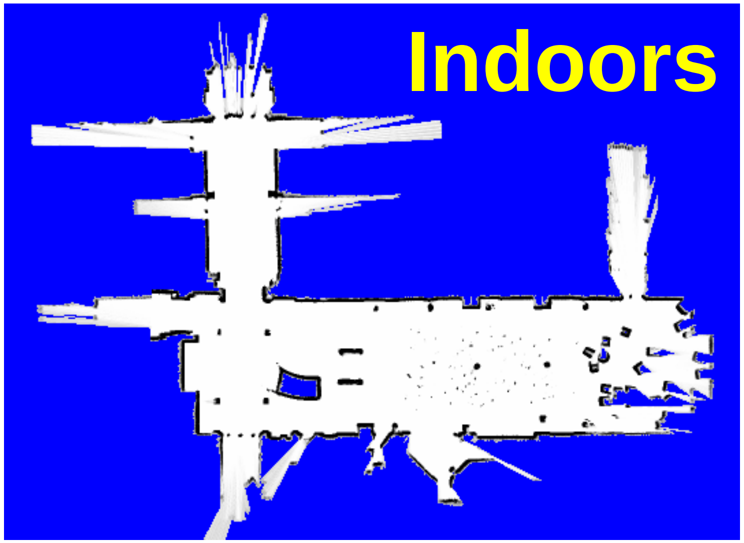
Estimate:

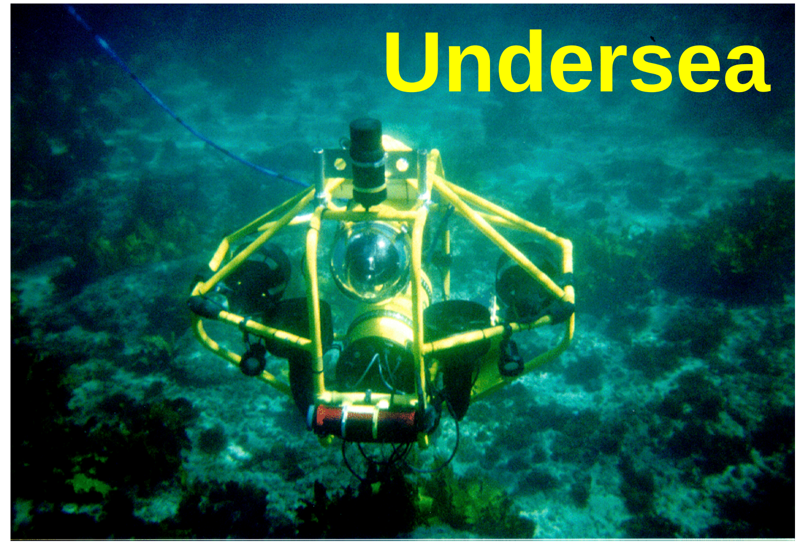
- Map of features
- Path of the robot

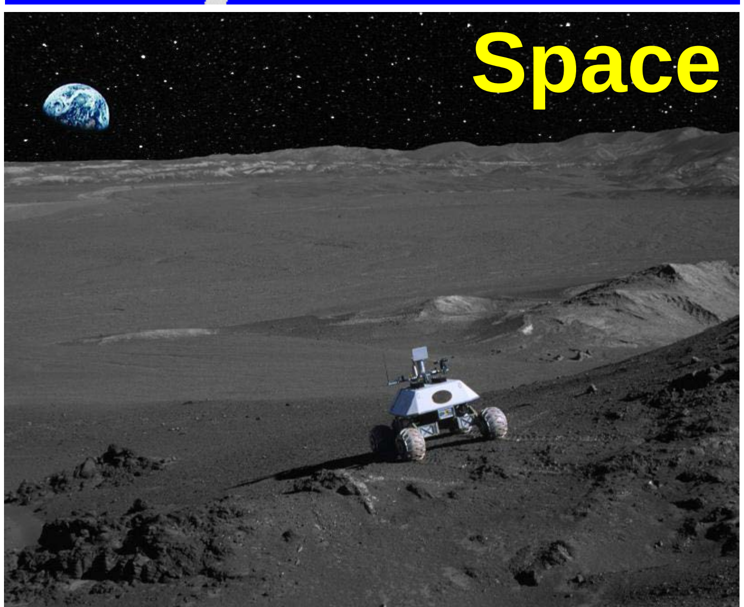
Structure of the Landmark-based SLAM-Problem



SLAM Applications

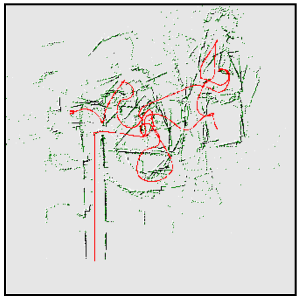
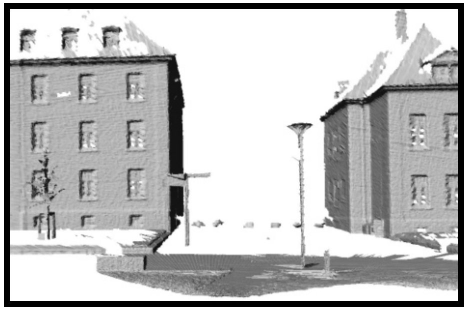
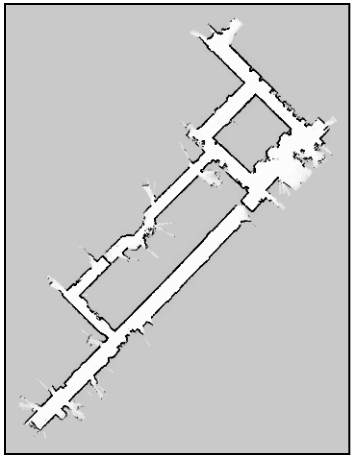






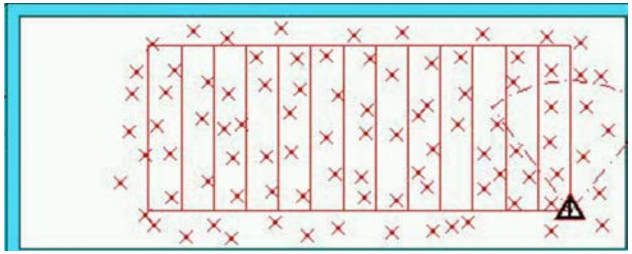
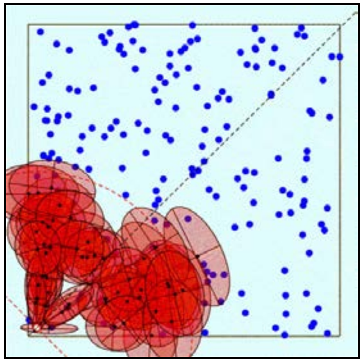
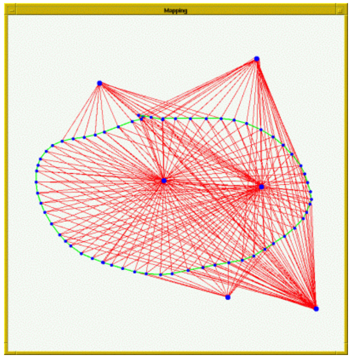
Representations

- Grid maps or scans

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

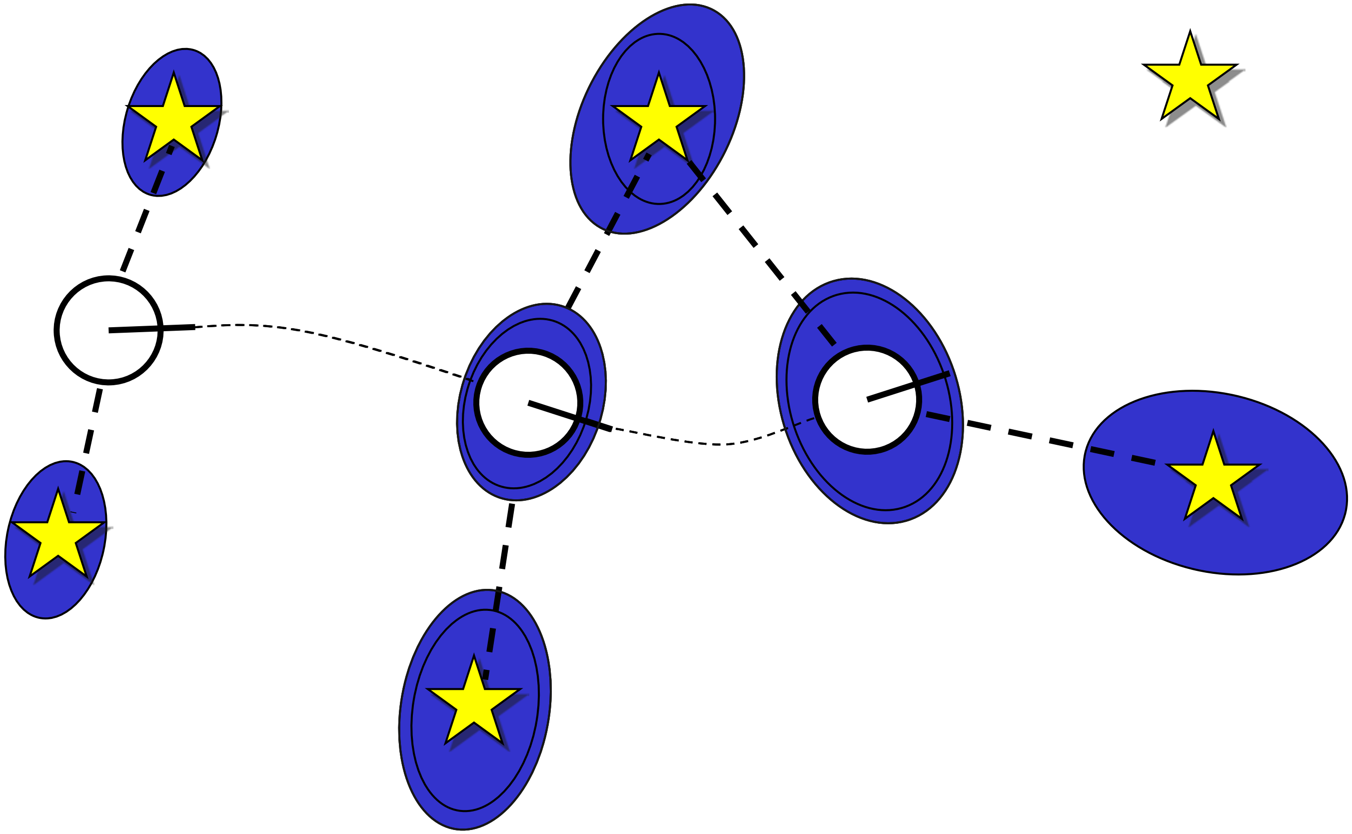
- Landmark-based

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

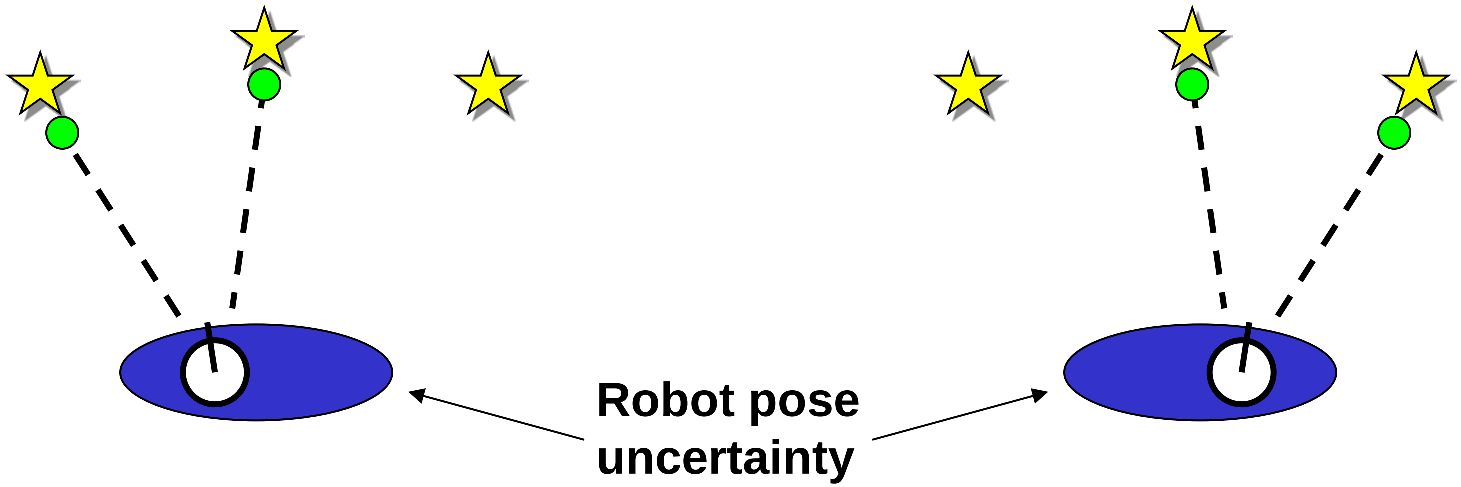
Why is SLAM a hard problem?

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!

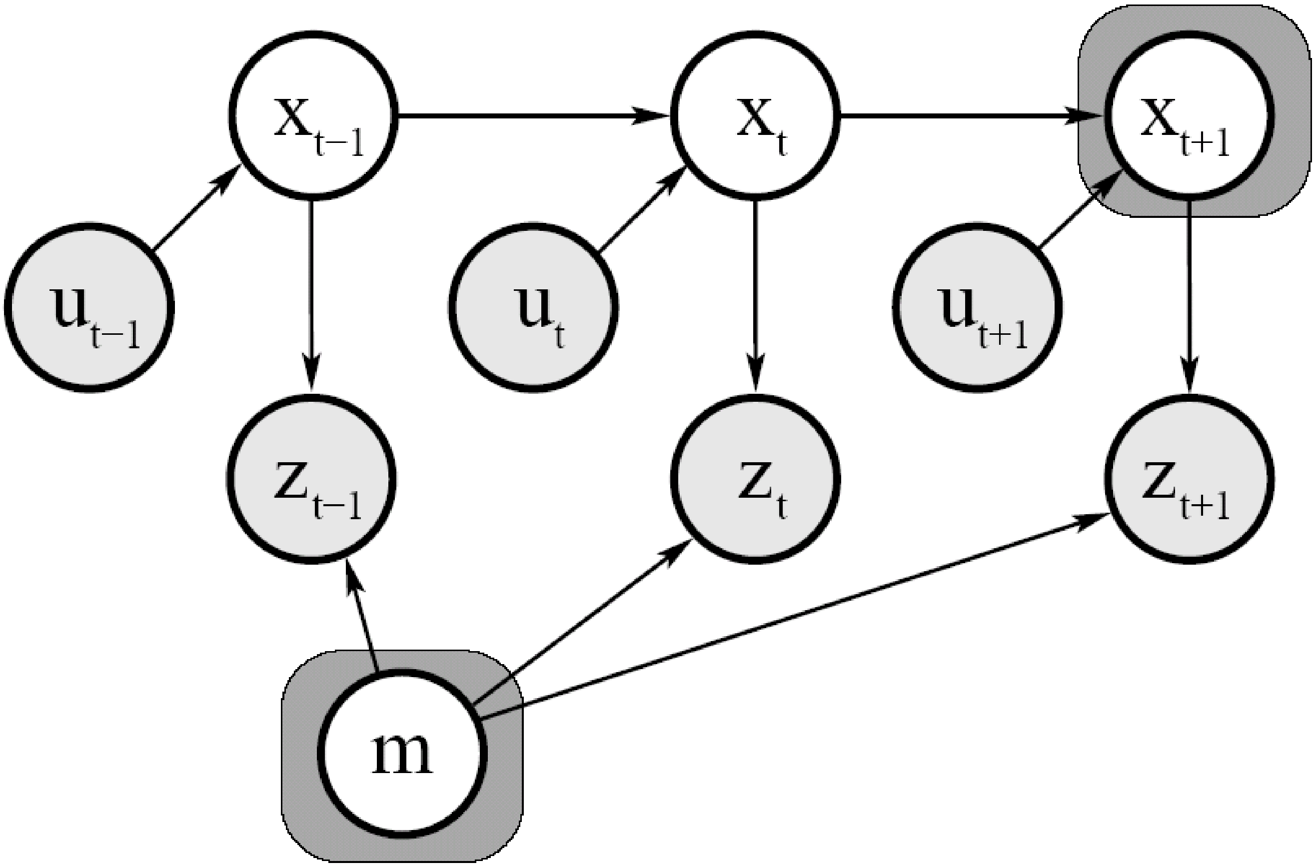
$$p(x_{1:t}, m \mid z_{1:t}, u_{1:t})$$

- Online SLAM:

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

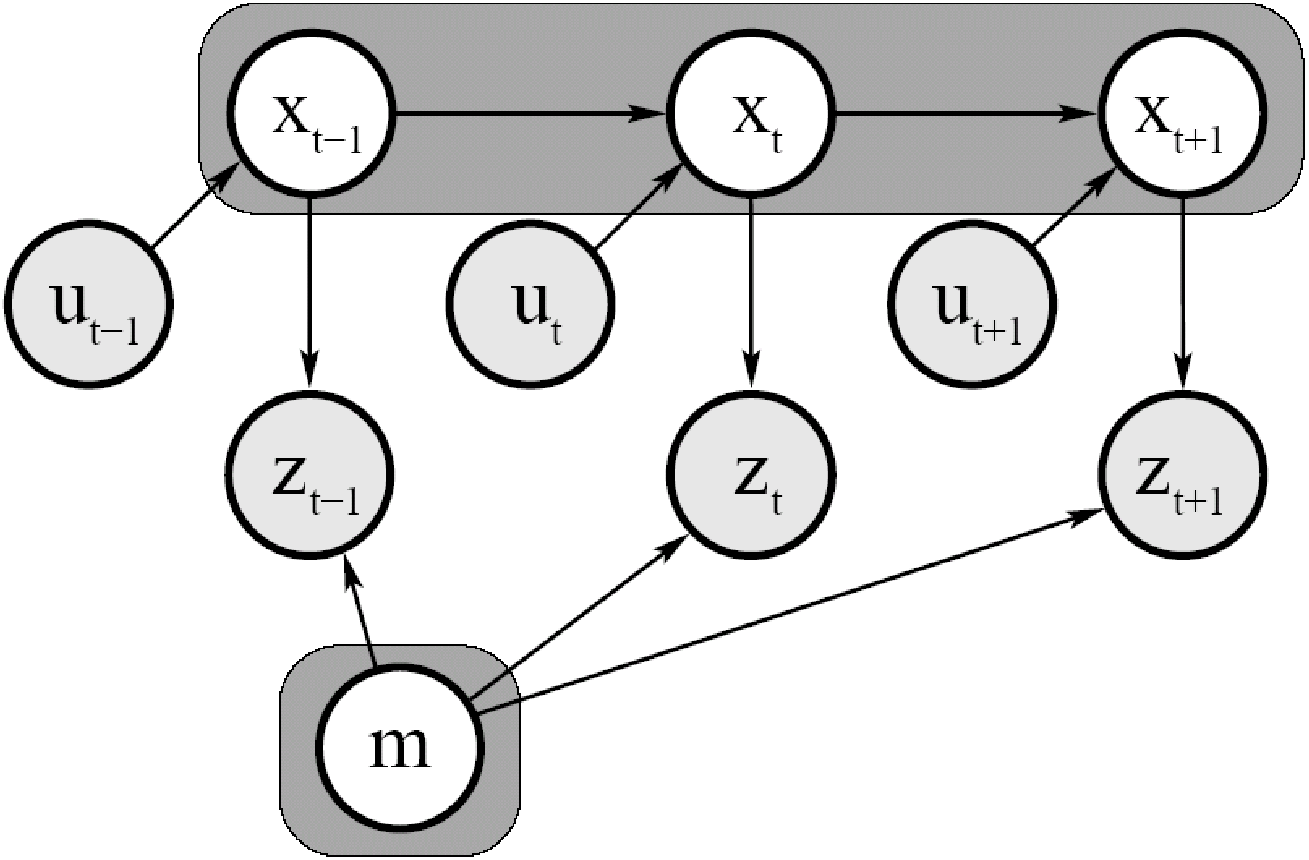
In Estimates most recent pose and map!

Graphical Model of Online SLAM:



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

Graphical Model of Full SLAM:



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

Techniques for Generating Consistent Maps

- Scan matching
- EKF SLAM
- Fast-SLAM
- Probabilistic mapping with a single map and a posterior about poses
  Mapping + Localization
- Graph-SLAM, SEIFs

Scan Matching

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

$$\hat{x}_t = \underset{x_t}{\arg\max} \left\{ p(z_t \mid x_t, \hat{m}^{[t-1]}) \cdot p(x_t \mid u_{t-1}, \hat{x}_{t-1}) \right\}$$

current measurement

robot motion

map constructed so far

Calculate the map $\hat{m}^{[t]}$ according to “mapping with known poses” based on the poses and observations.

Kalman Filter Algorithm

1. Algorithm **Kalman_filter**($\mu_{t-1}, \Sigma_{t-1}, u_t, z_t$):

2. Prediction:
3. $\bar{\mu}_t = A_t \mu_{t-1} + B_t u_t$
4. $\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + R_t$

5. Correction:
6. $K_t = \bar{\Sigma}_t C_t^T (C_t \bar{\Sigma}_t C_t^T + Q_t)^{-1}$
7. $\mu_t = \bar{\mu}_t + K_t (z_t - C_t \bar{\mu}_t)$
8. $\Sigma_t = (I - K_t C_t) \bar{\Sigma}_t$

9. Return μ_t, Σ_t

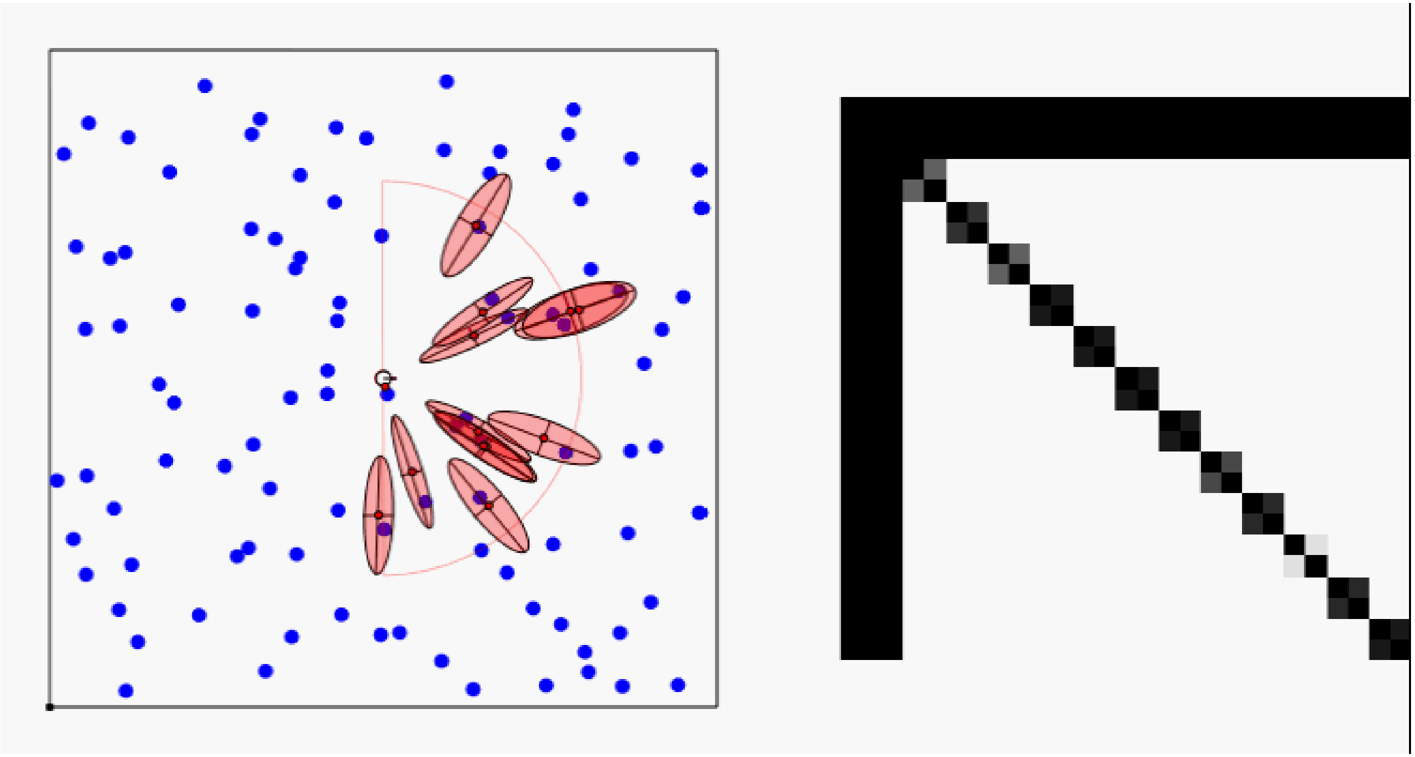
(E)KF-SLAM

- Map with N landmarks:(3+2N)-dimensional Gaussian

$$Bel(x_t, m_t) = \left\langle \begin{pmatrix} x \\ y \\ \theta \\ l_1 \\ l_2 \\ \vdots \\ l_N \end{pmatrix}, \begin{pmatrix} \sigma_x^2 & \sigma_{xy} & \sigma_{x\theta} & \sigma_{xl_1} & \sigma_{xl_2} & \cdots & \sigma_{xl_N} \\ \sigma_{xy} & \sigma_y^2 & \sigma_{y\theta} & \sigma_{yl_1} & \sigma_{yl_2} & \cdots & \sigma_{yl_N} \\ \sigma_{x\theta} & \sigma_{y\theta} & \sigma_\theta^2 & \sigma_{\theta l_1} & \sigma_{\theta l_2} & \cdots & \sigma_{\theta l_N} \\ \sigma_{xl_1} & \sigma_{yl_1} & \sigma_{\theta l_1} & \sigma_{l_1}^2 & \sigma_{l_1 l_2} & \cdots & \sigma_{l_1 l_N} \\ \sigma_{xl_2} & \sigma_{yl_2} & \sigma_{\theta l_2} & \sigma_{l_1 l_2} & \sigma_{l_2}^2 & \cdots & \sigma_{l_2 l_N} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \sigma_{xl_N} & \sigma_{yl_N} & \sigma_{\theta l_N} & \sigma_{l_1 l_N} & \sigma_{l_2 l_N} & \cdots & \sigma_{l_N}^2 \end{pmatrix} \right\rangle$$

- Can handle hundreds of dimensions

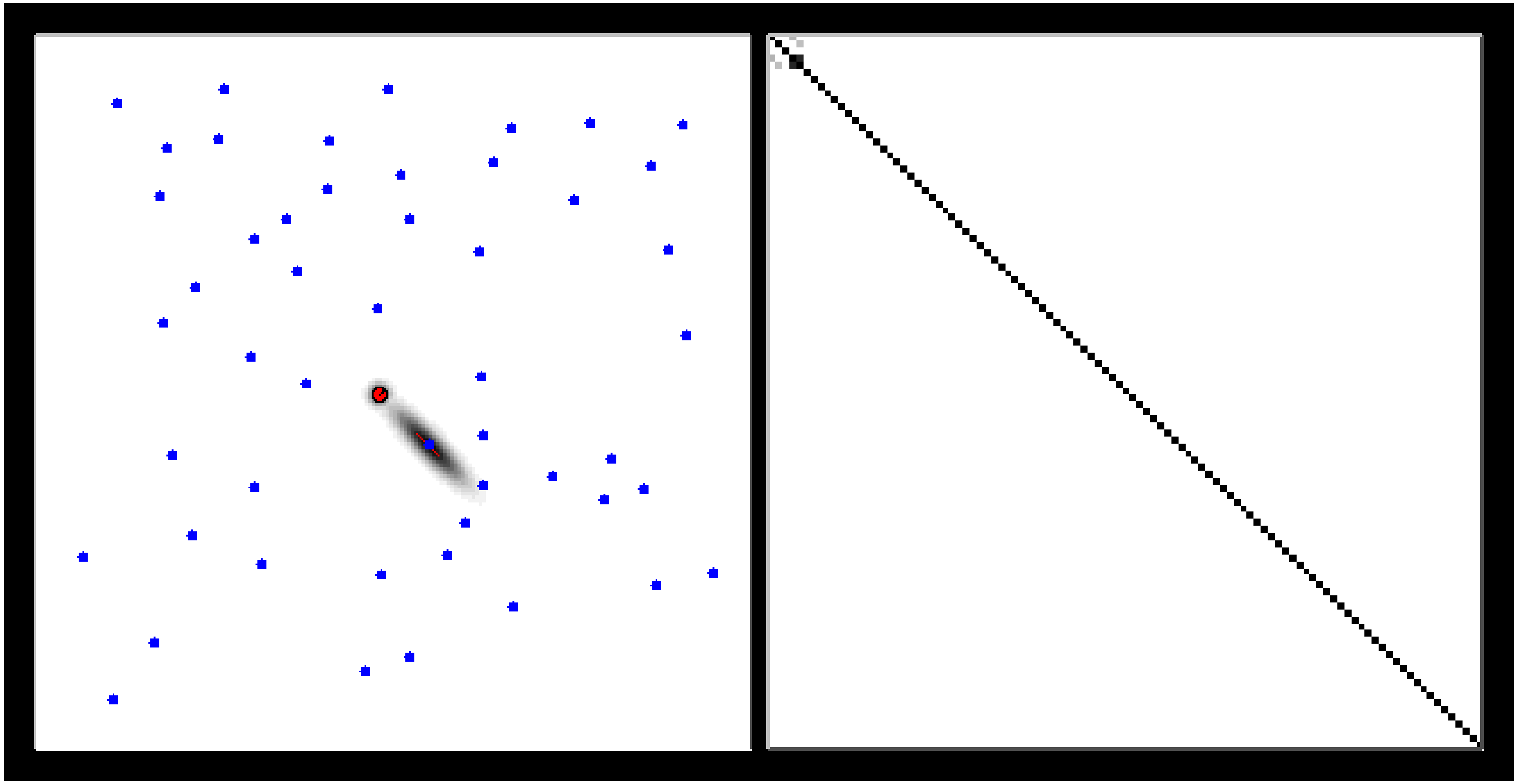
Classical Solution – The EKF



Blue path = true path **Red path** = estimated path **Black path** = odometry

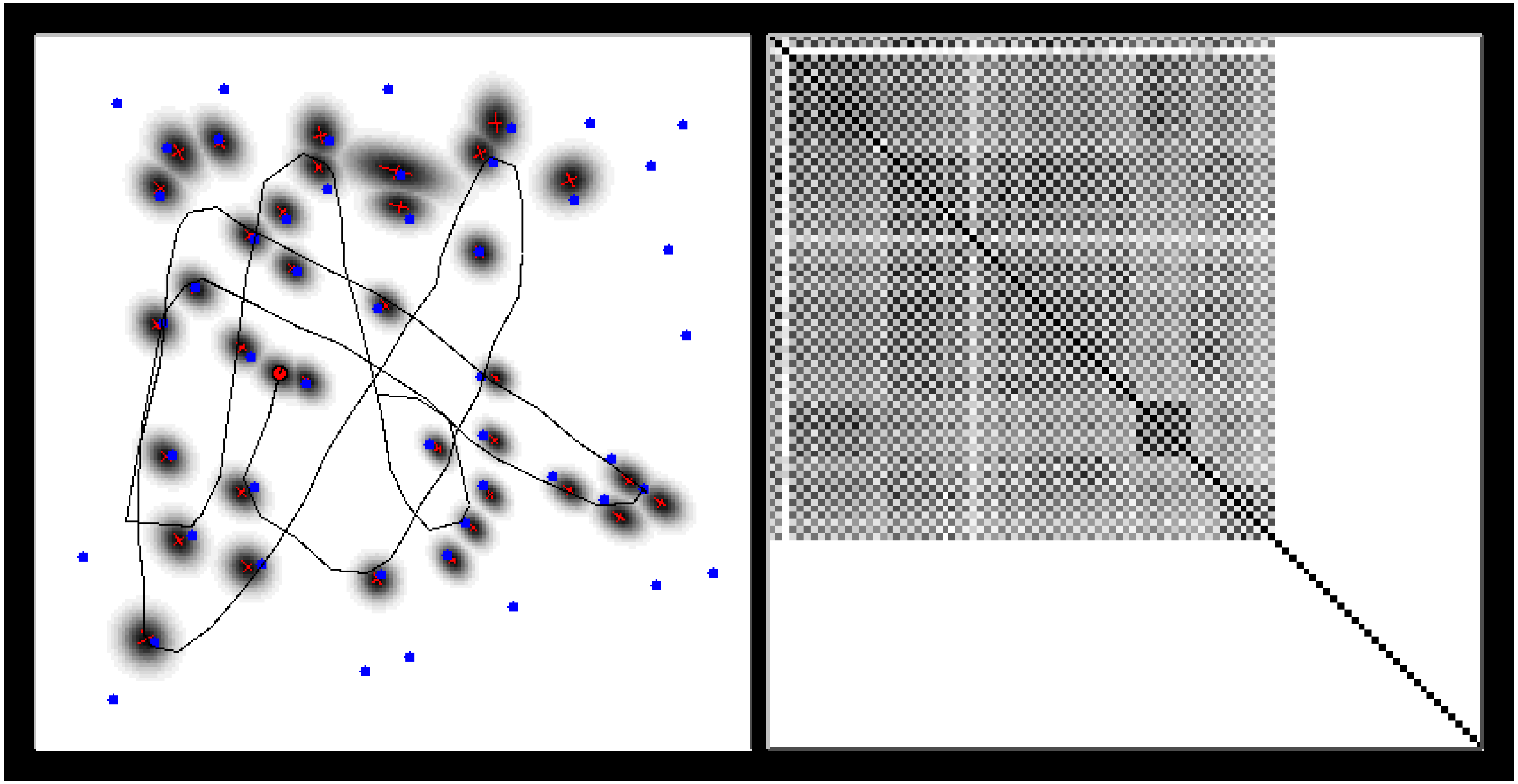
- Approximate the SLAM posterior with a high-dimensional Gaussian [Smith & Cheesman, 1986] …
- Single hypothesis data association

EKF-SLAM



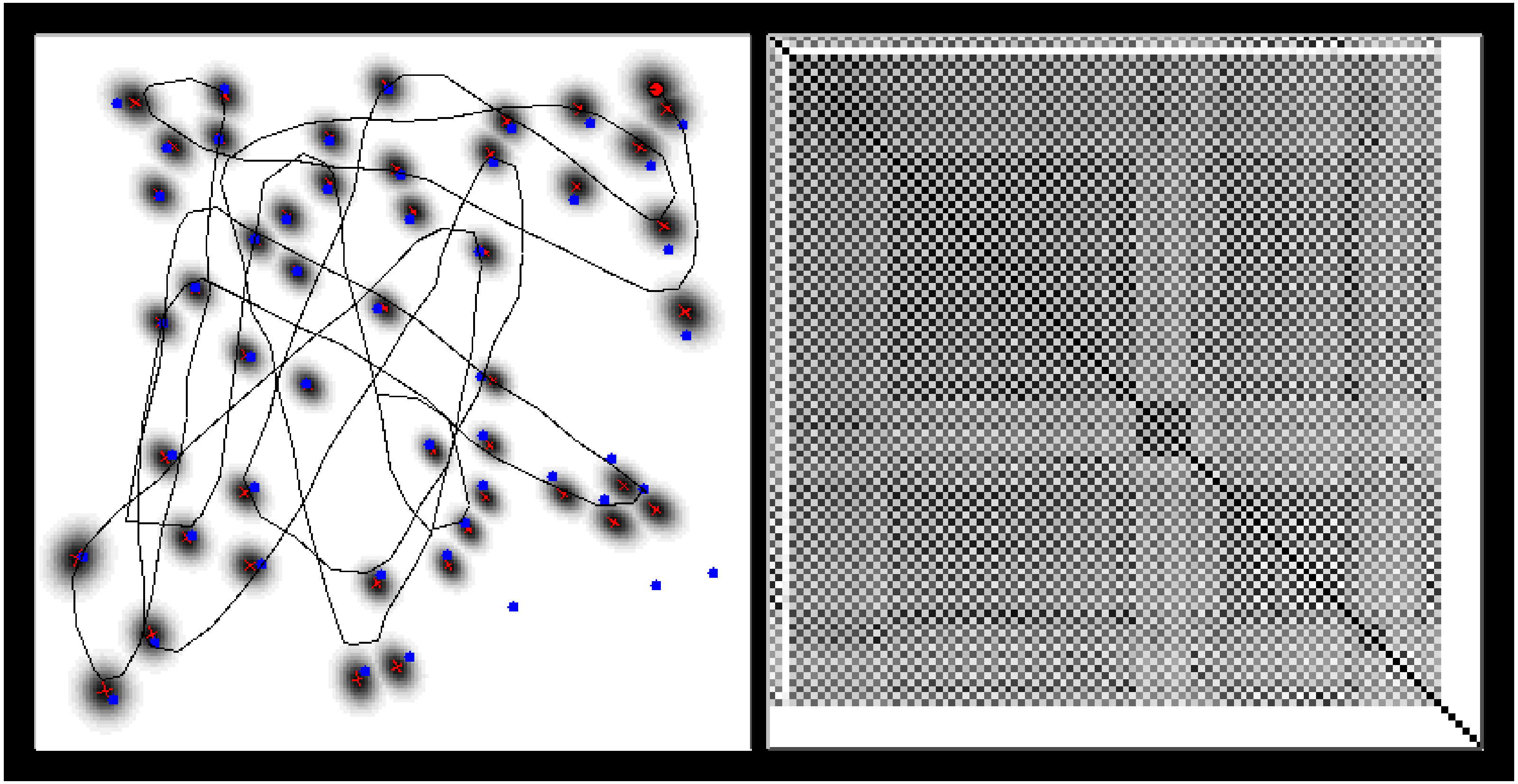
Map Correlation matrix

EKF-SLAM



Map Correlation matrix

EKF-SLAM



Map Correlation matrix

Properties of KF-SLAM (Linear Case)

[Dissanayake et al., 2001]

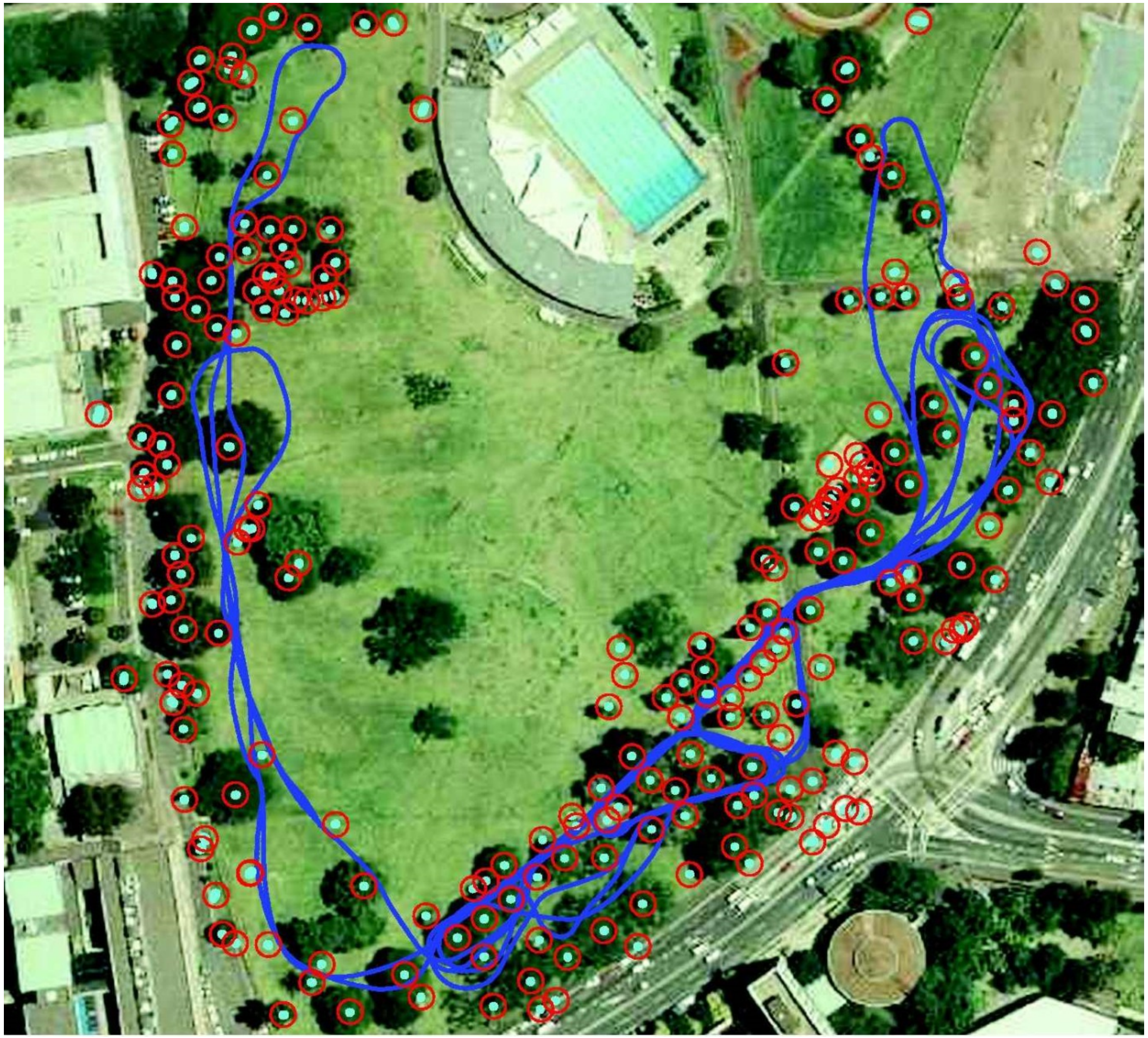
Theorem:

The determinant of any sub-matrix of the map covariance matrix decreases monotonically as successive observations are made.

Theorem:

In the limit the landmark estimates become fully correlated

Victoria Park Data Set



[courtesy by E. Nebot]

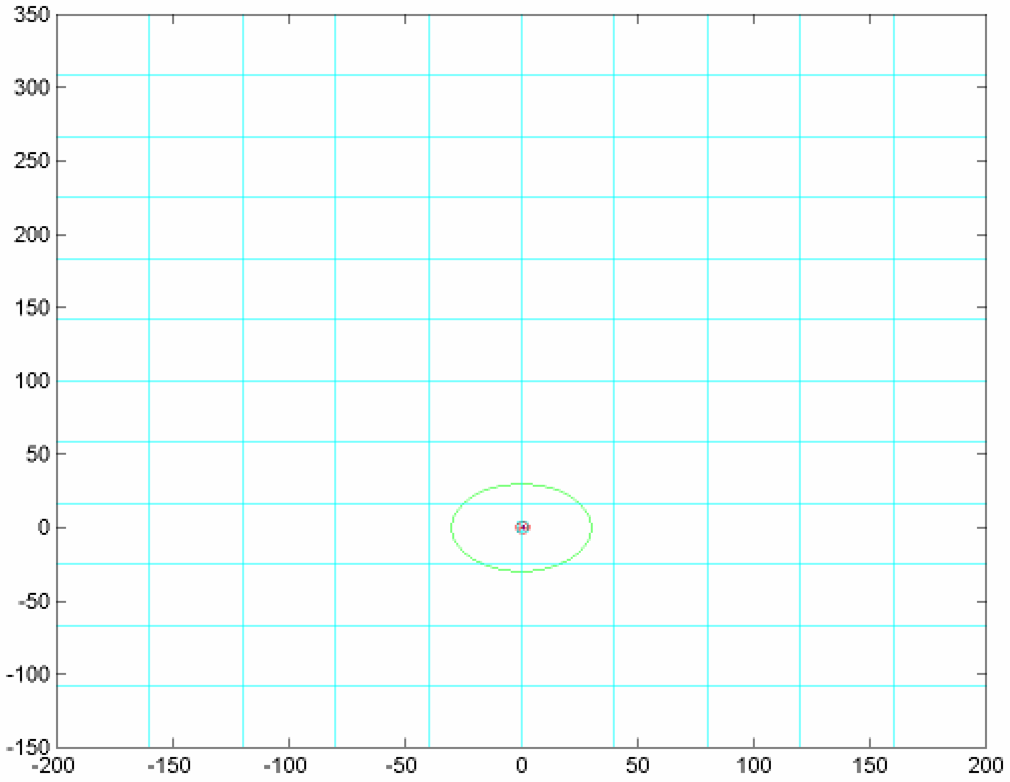
Victoria Park Data Set Vehicle

[courtesy by E. Nebot]

Data Acquisition

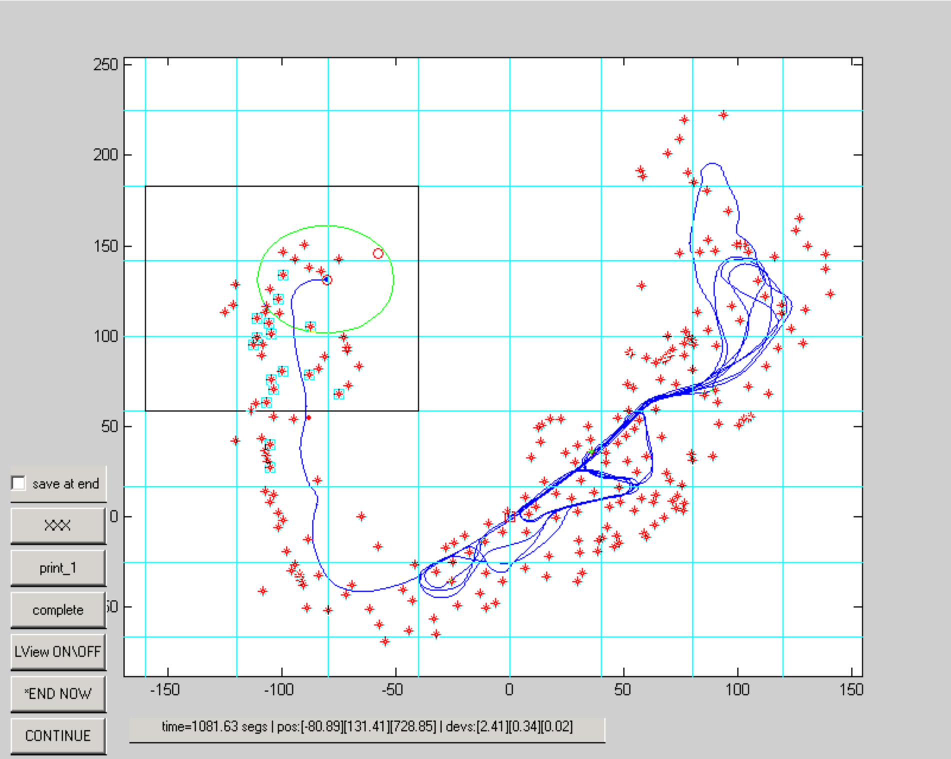
[courtesy by E. Nebot]

SLAM



[courtesy by E. Nebot]

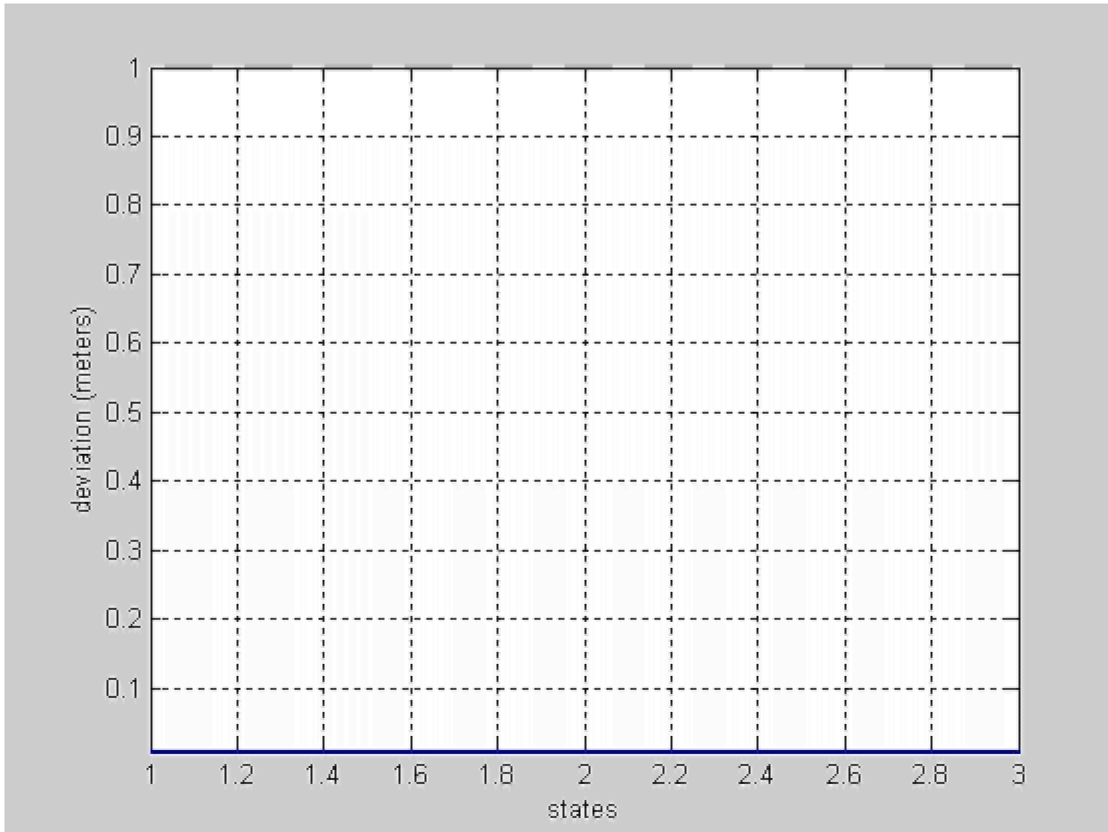
Map and Trajectory



Landmarks
Covariance

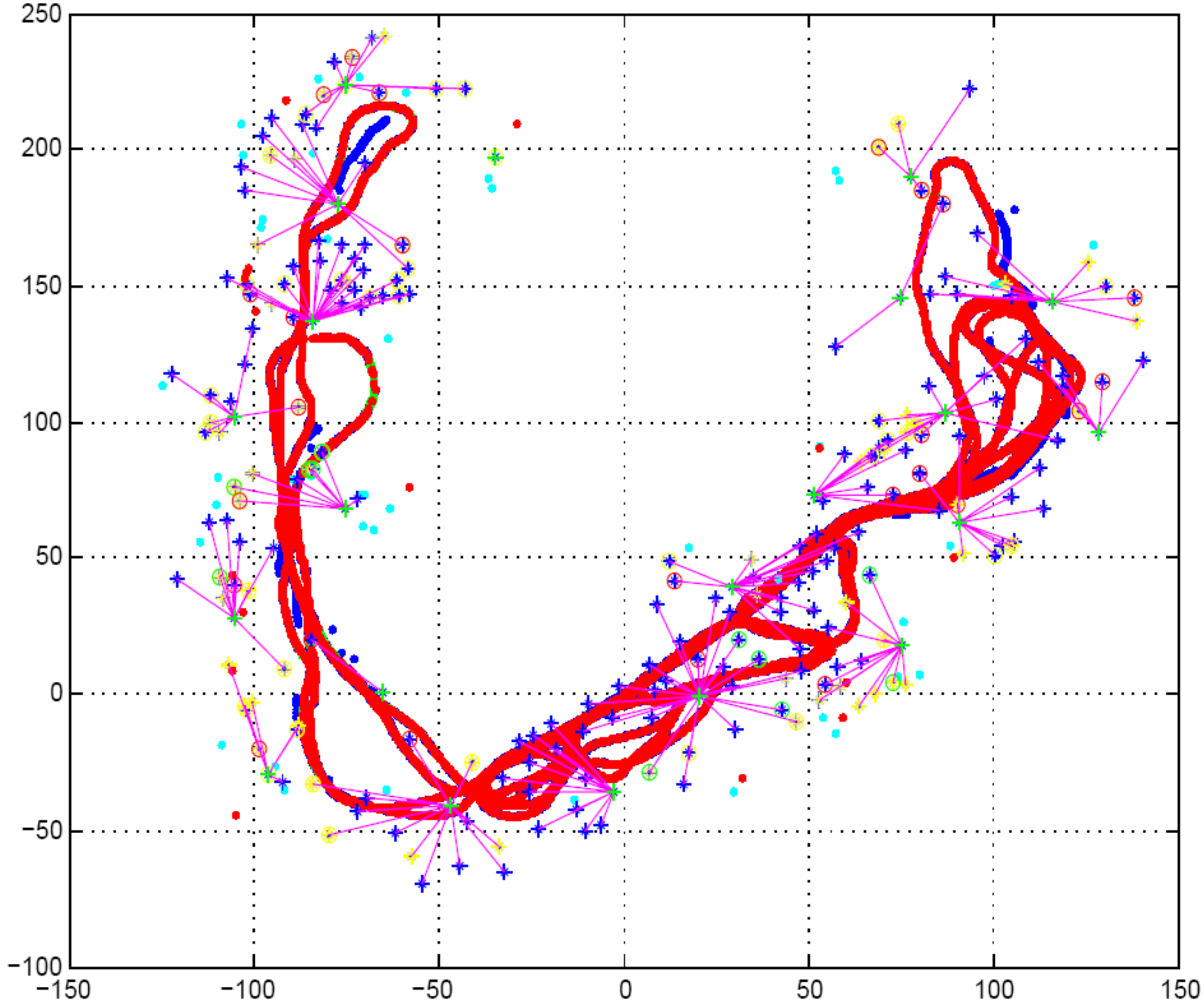
[courtesy by E. Nebot]

Landmark Covariance



[courtesy by E. Nebot]

Estimated Trajectory

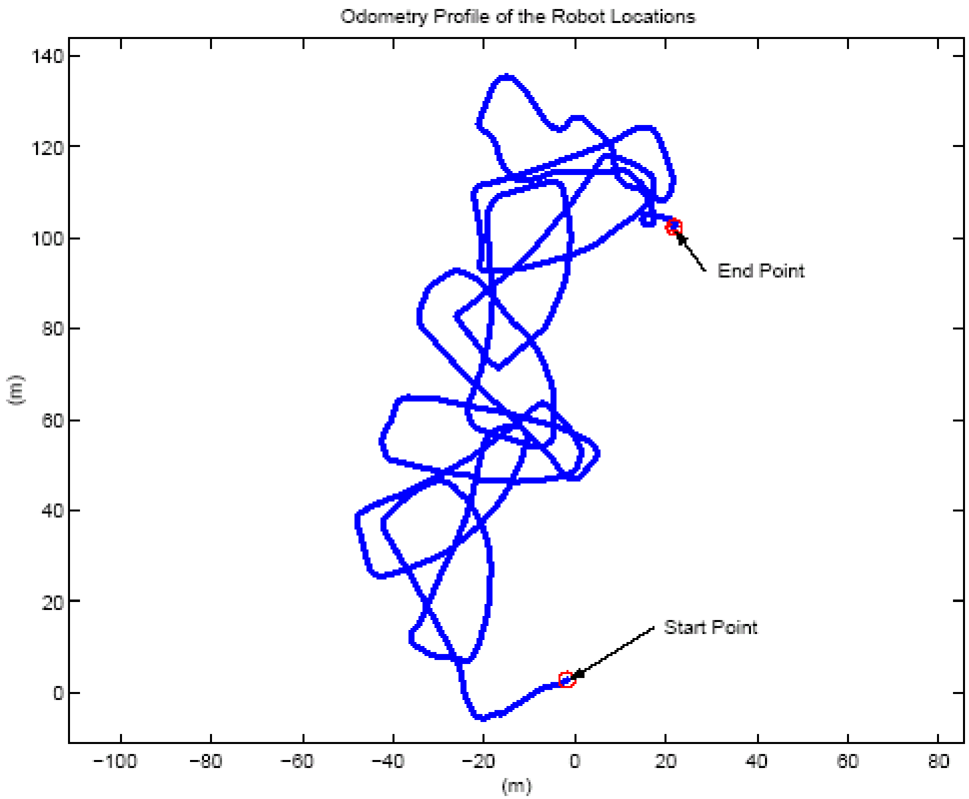


[courtesy by E. Nebot]

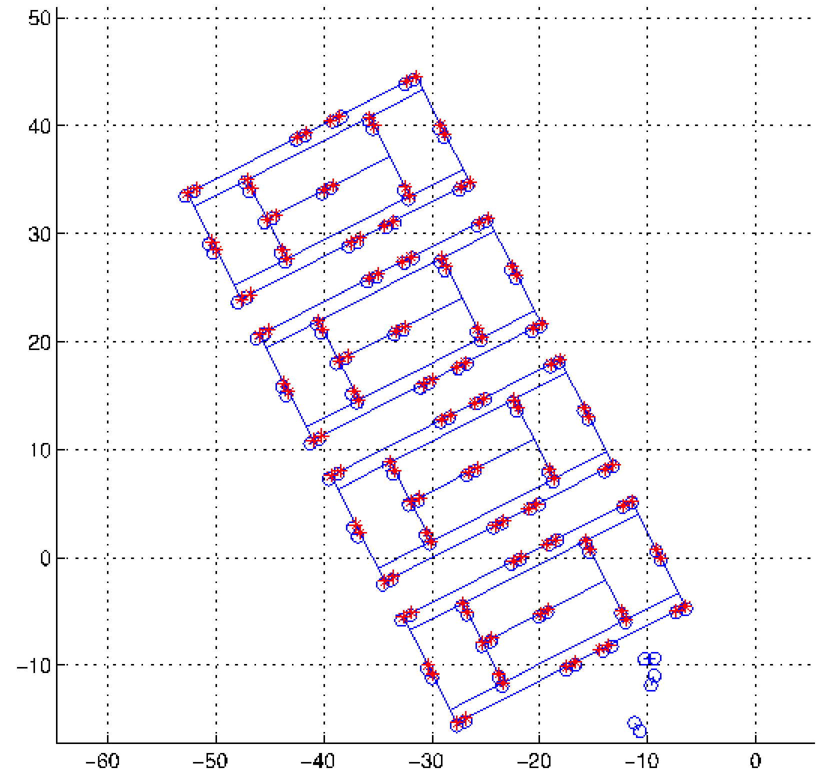
EKF SLAM Application

[courtesy by John Leonard]

EKF SLAM Application



odometry



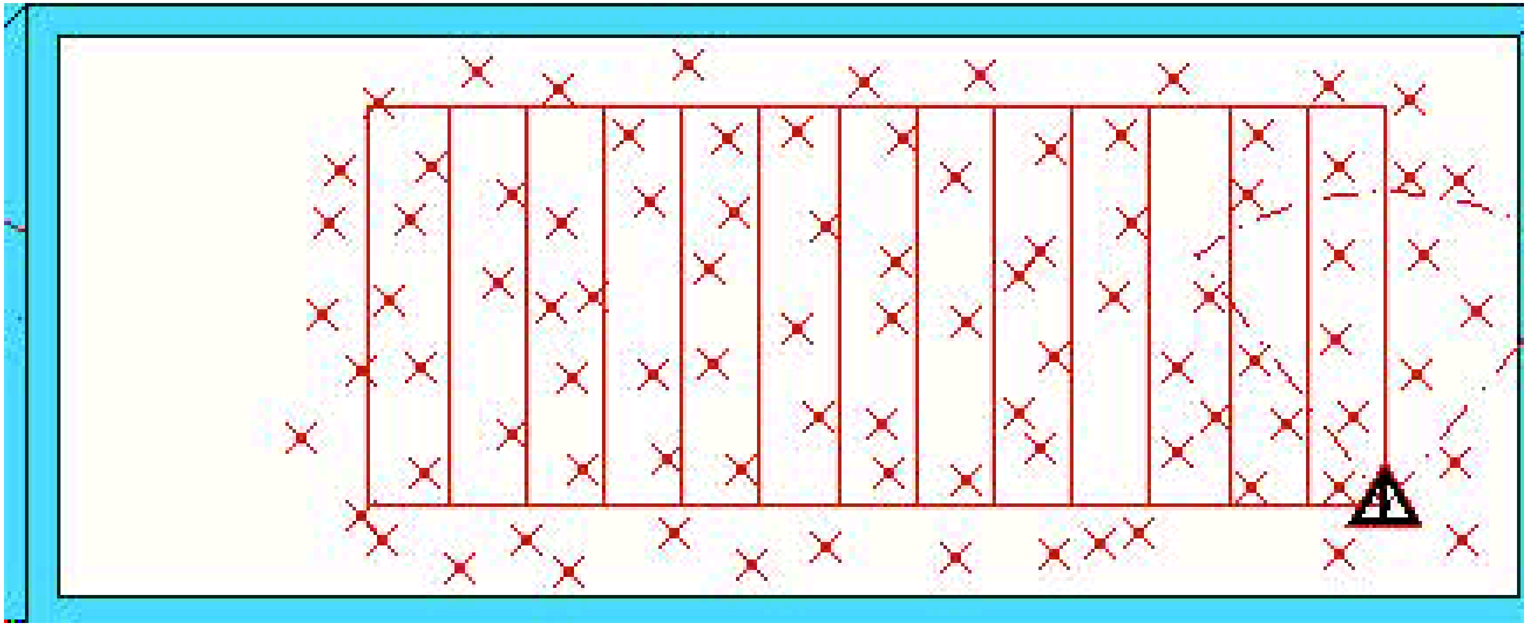
estimated trajectory

[courtesy by John Leonard]

Approximations for SLAM

- Local submaps
  [Leonard et al.99, Bosse et al. 02, Newman et al. 03]
- Sparse links (correlations)
  [Lu & Milios 97, Guivant & Nebot 01]
- Sparse extended information filters
  [Frese et al. 01, Thrun et al. 02]
- Thin junction tree filters
  [Paskin 03]
- Rao-Blackwellisation (FastSLAM)
  [Murphy 99, Montemerlo et al. 02, Eliazar et al. 03, Haehnel et al. 03]

Sub-maps for EKF SLAM



[Leonard et al, 1998]

EKF-SLAM Summary

- Quadratic in the number of landmarks: $O(n^2)$
- Convergence results for the linear case.
- Can diverge if nonlinearities are large!
- Have been applied successfully in large-scale environments.
- Approximations reduce the computational complexity.