Robot Mapping

Summary on the Kalman Filter & Friends: KF, EKF, UKF, EIF, SEIF

Gian Diego Tipaldi, Wolfram Burgard

Three Main SLAM Paradigms

Kalman filter	Particle filter	Graph-based

Kalman Filter & Its Friends

Kalman filter

Particle filter

Graph-based

- Kalman filter
- **Extended Kalman Filter**
- **Unscented Kalman Filter**
- **Extended Information Filter**
- **Sparse Extended Information Filter**

Kalman Filter Algorithm

1: $\textbf{Kalman\_filter}(\mu_{t-1}, \Sigma_{t-1}, u_t, z_t)\textbf{:}$

2: $\bar{\mu}_t = A_t \, \mu_{t-1} + B_t \, u_t$

3: $\bar{\Sigma}_t = A_t \, \Sigma_{t-1} \, A_t^T + R_t$

prediction

4: $K_t = \bar{\Sigma}_t \, C_t^T (C_t \, \bar{\Sigma}_t \, C_t^T + Q_t)^{-1}$

5: $\mu_t = \bar{\mu}_t + K_t(z_t - C_t \, \bar{\mu}_t)$

6: $\Sigma_t = (I - K_t \, C_t) \, \bar{\Sigma}_t$

7: $return \; \mu_t, \Sigma_t$

correction

Non-linear Dynamic Systems

- Most realistic problems in robotics involve nonlinear functions

$$\cancel{x_t = A_t x_{t-1} + B_t u_t + \epsilon_t} \qquad \cancel{z_t = C_t x_t + \delta_t}$$

$$\Downarrow \qquad\qquad \Downarrow$$

$$x_t = g(u_t, x_{t-1}) + \epsilon_t \qquad z_t = h(x_t) + \delta_t$$

requires linearization

$\Rightarrow$ **EKF**

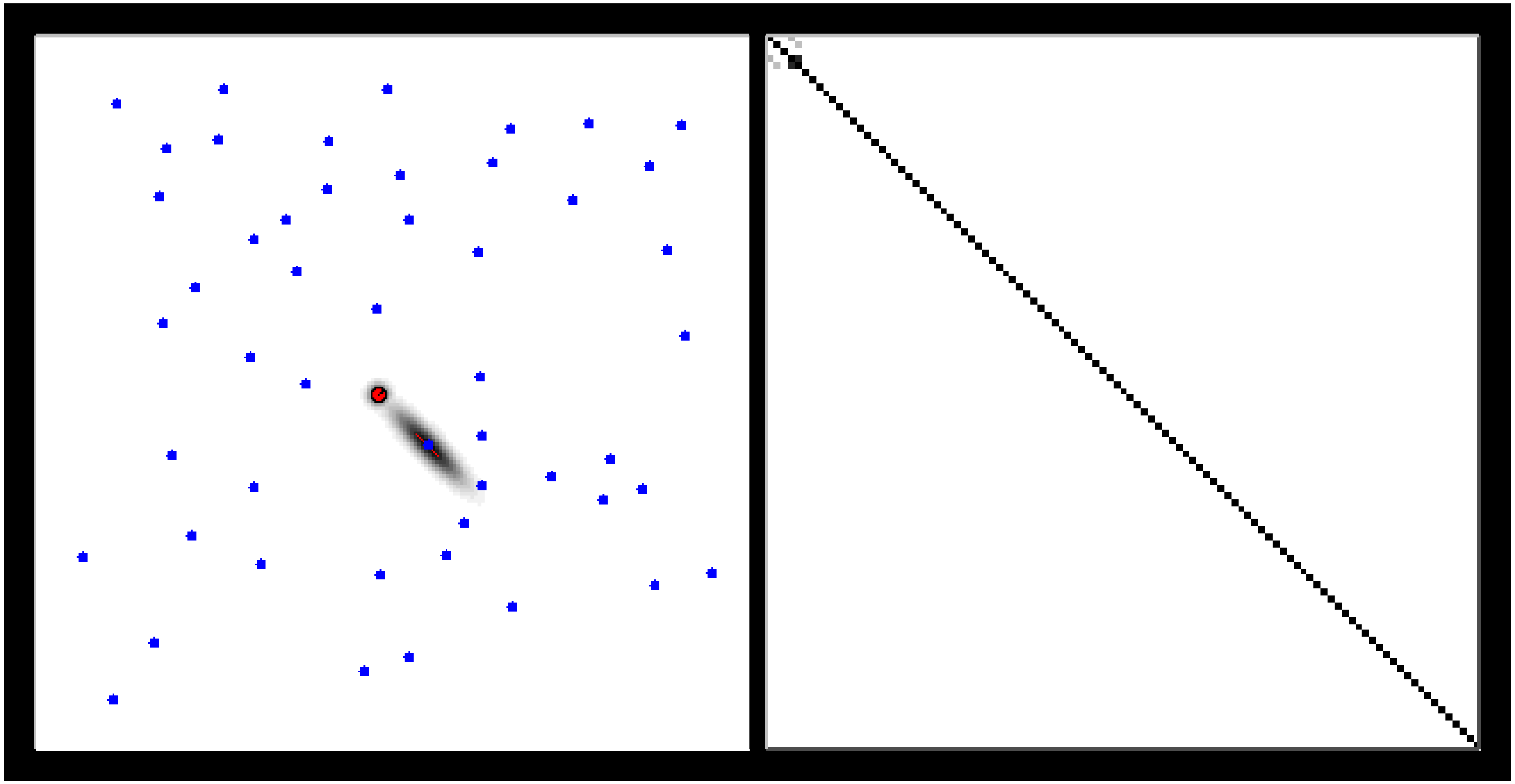
KF vs. EKF

- EKF is an extension of the KF
- Approach to handle the non-linearities
- Performs local linearizations
- Works well in practice for moderate non-linearities and uncertainty

EKF for SLAM

$$\underbrace{\begin{pmatrix} x_R \\ m_1 \\ \vdots \\ m_n \end{pmatrix}}_{\mu} \quad \underbrace{\begin{pmatrix} \Sigma_{x_R x_R} & \Sigma_{x_R m_1} & \dots & \Sigma_{x_R m_n} \\ \Sigma_{m_1 x_R} & \Sigma_{m_1 m_1} & \dots & \Sigma_{m_1 m_n} \\ \vdots & \vdots & \ddots & \vdots \\ \Sigma_{m_n x_R} & \Sigma_{m_n m_1} & \dots & \Sigma_{m_n m_n} \end{pmatrix}}_{\Sigma}$$

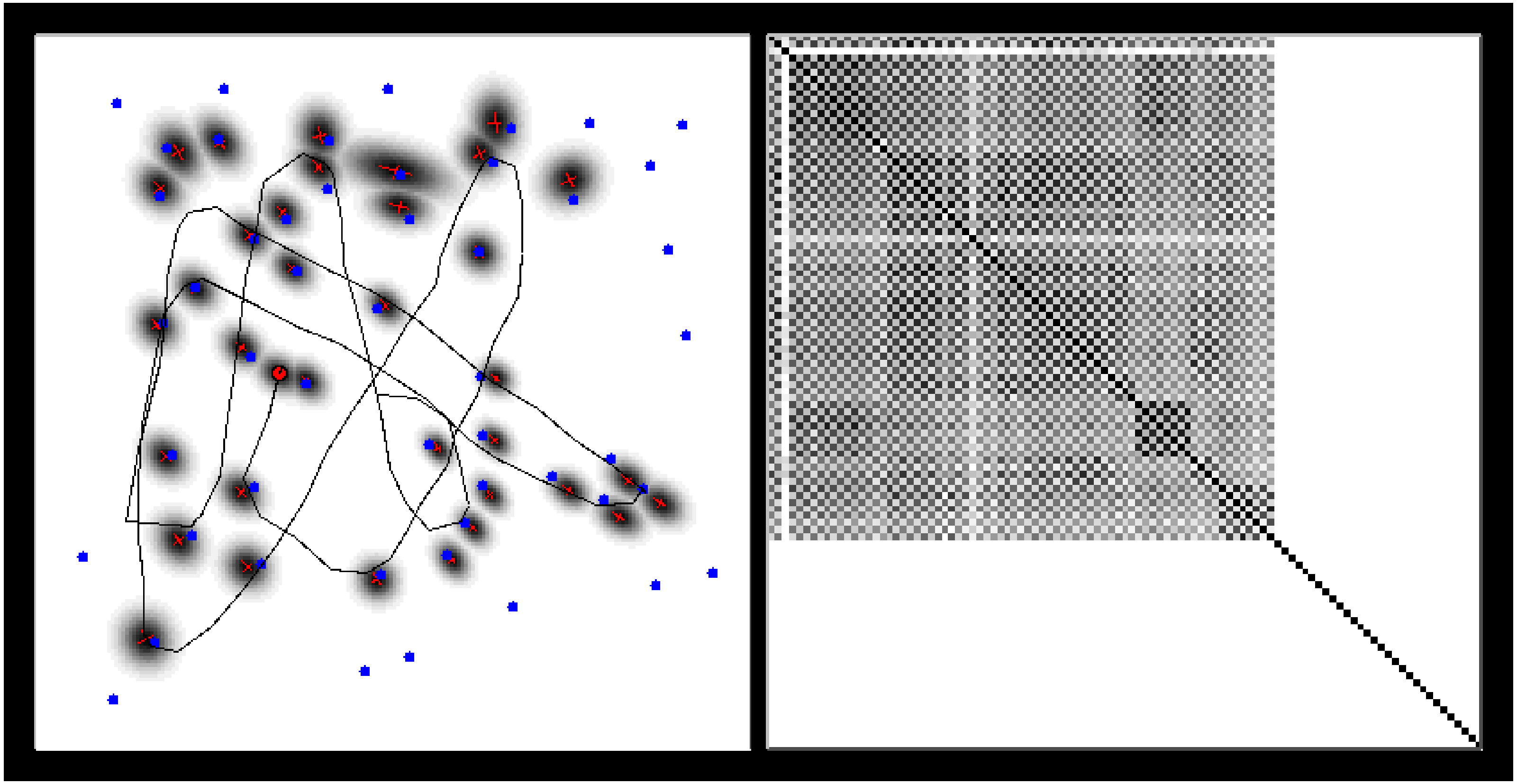
EKF SLAM



Map Correlation matrix

Courtesy: M. Montemerlo

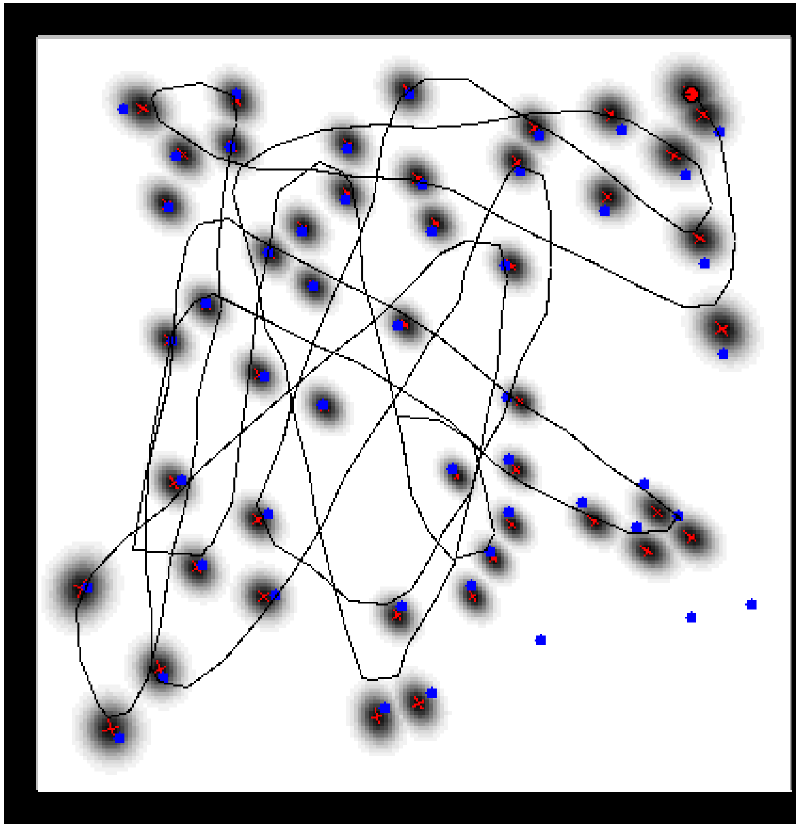
EKF SLAM



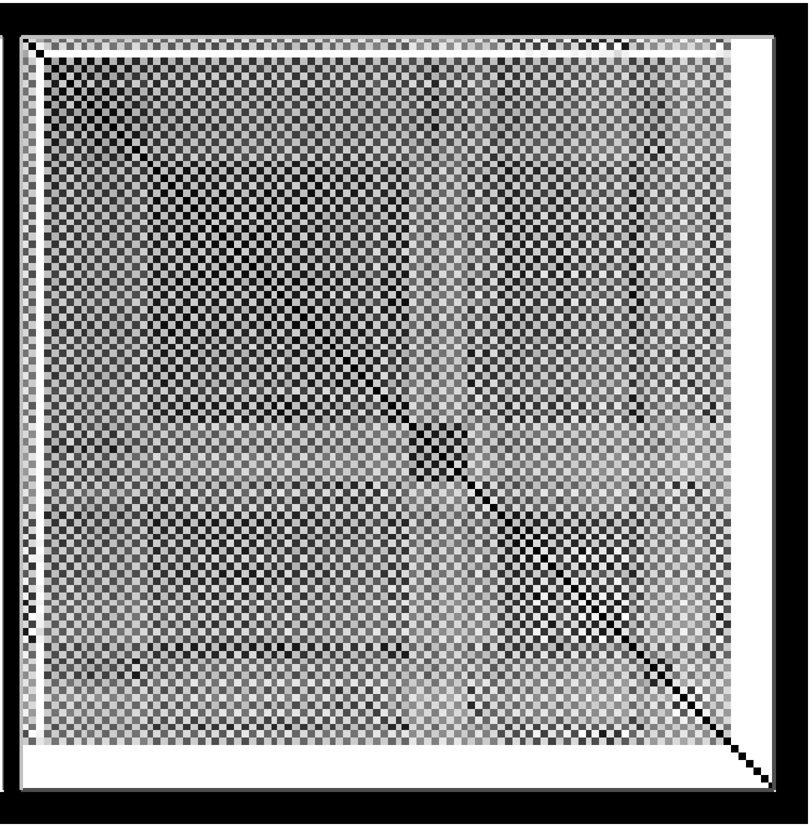
Map Correlation matrix

Courtesy: M. Montemerlo

EKF SLAM



Map

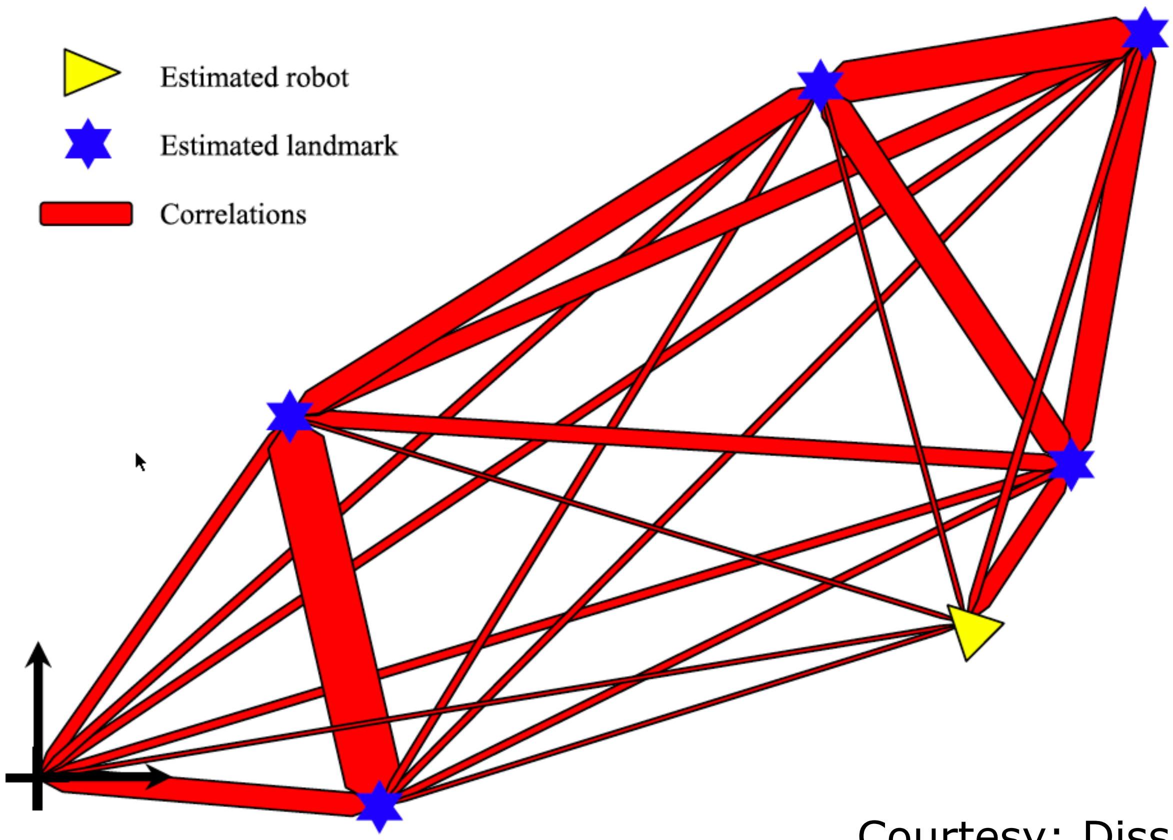


Correlation matrix

Courtesy: M. Montemerlo

EKF-SLAM Properties

- In the limit, the landmark estimates become **fully correlated**



Courtesy: Dissanayake

EKF-SLAM Complexity

- Cubic complexity only on the measurement dimensionality
- Cost per step: dominated by the number of landmarks: $O(n^2)$
- Memory consumption: $O(n^2)$
- The EKF becomes computationally intractable for large maps!

Unscented Kalman Filter (UKF)

UKF Motivation

- Kalman filter requires linear models
- EKF linearizes via Taylor expansion

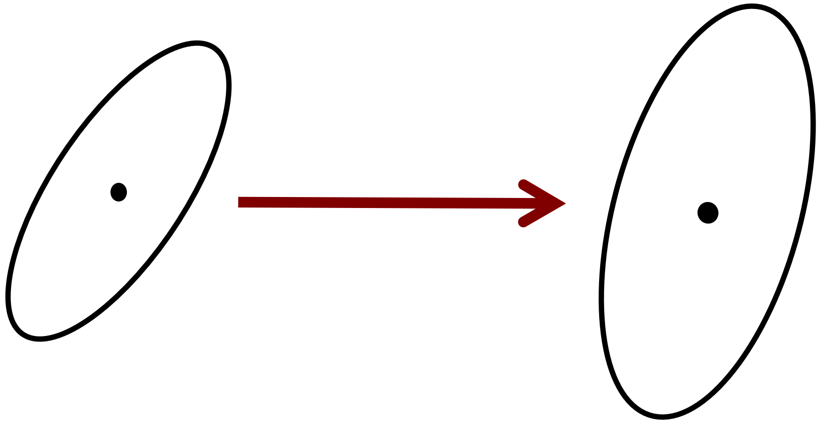
Is there a better way to linearize?

Unscented Transform

⬇

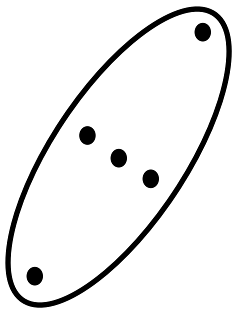
Unscented Kalman Filter (UKF)

Taylor Approximation (EKF)



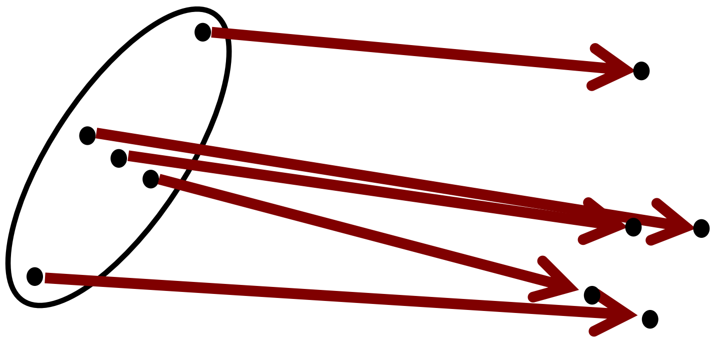
Linearization of the non-linear
function through Taylor expansion

Unscented Transform



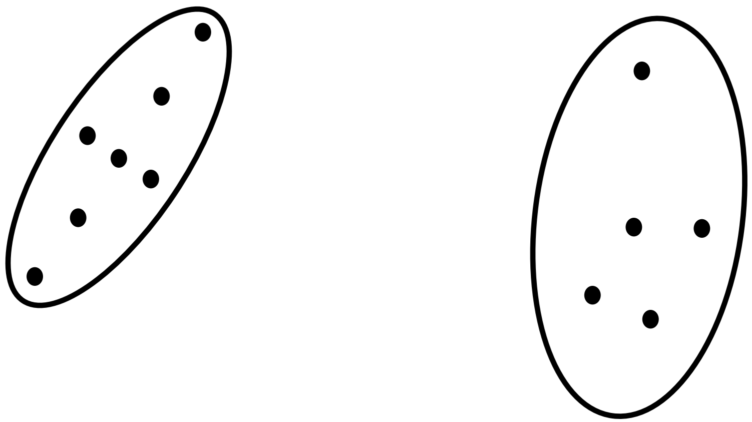
Compute a set of (so-called) sigma points

Unscented Transform



Transform each sigma point through the non-linear motion and measurement functions

Unscented Transform



Reconstruct a Gaussian from the transformed and weighted points

UKF vs. EKF

- Same results as EKF for linear models
- Better approximation than EKF for non-linear models
- Differences often “somewhat small”
- No Jacobians needed for the UKF
- Same complexity class
- Slightly slower than the EKF

EIF: Two Parameterizations for a Gaussian Distribution

moments

$$\Sigma = \Omega^{-1}$$

$$\mu = \Omega^{-1}\xi$$

covariance matrix
mean vector

canonical

$$\Omega = \Sigma^{-1}$$

$$\xi = \Sigma^{-1}\mu$$

information matrix
information vector

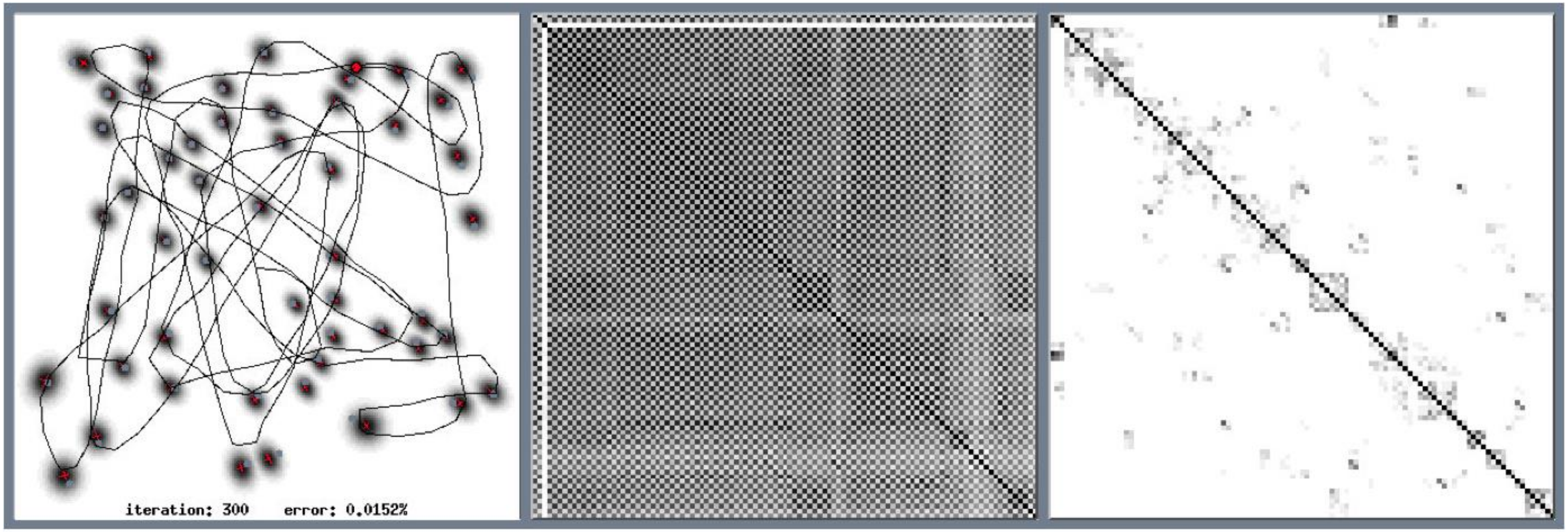
Extended Information Filter

- The EIF is the EKF in information form
- Instead of the moments Σ, μ the canonical form is maintained using Ω, ξ
- Conversion between information for and canonical form is expensive
- EIF has the same expressiveness than the EKF

EIF vs. EKF

- Complexity of the prediction and corrections steps differs
- KF: efficient prediction, slow correction
- IF: slow prediction, efficient correction
- “The application determines the filter”
- In practice, the EKF is more popular than the EIF

Motivation for SEIF SLAM



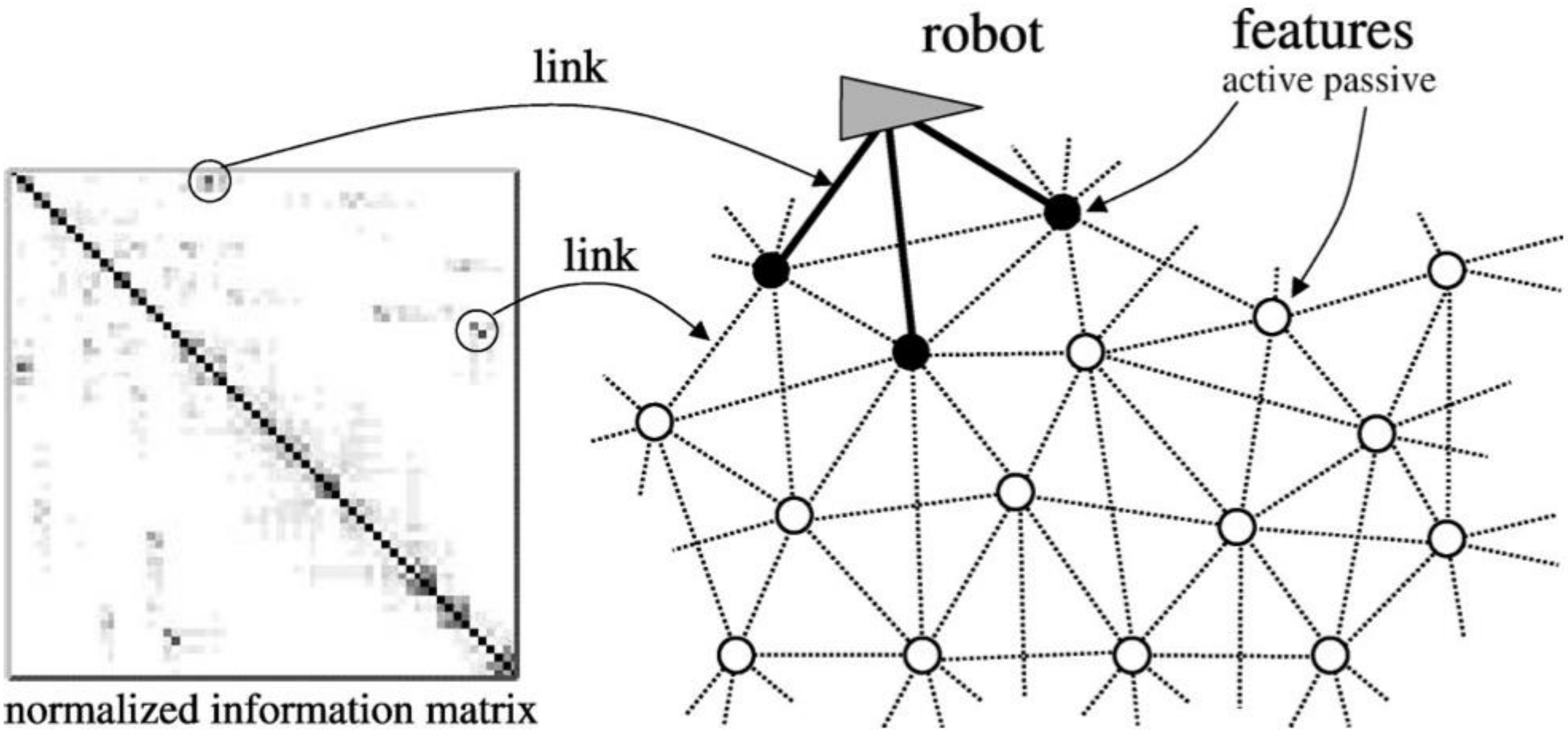
Gaussian estimate (map & pose)

normalized covariance matrix

normalized information matrix

Courtesy: Thrun, Burgard, Fox

Keep the Links Between in the Information Matrix Bounded



Courtesy: Thrun, Burgard, Fox

Four Steps of SEIF SLAM

1. Motion update
2. Measurement update
3. Update of the state estimate
4. Sparsification

Efficiency of SEIF SLAM

- Maintains the robot-landmark links only for a small set of landmarks at a time
- Removes robot-landmark links by sparsification (equal to assuming conditional independence)
- This also bounds the number of landmark-landmark links
- Exploits the sparsity of the information matrix in all computations

SEIF SLAM vs. EKF SLAM

- SEIFs are an efficient **approximation** of the EIF for the SLAM problem
- Neglects links by sparsification
- **Constant time** updates of the filter (for known correspondences)
- **Linear memory** complexity
- **Inferior quality** compared to EKF SLAM

Summary

- KFs deal differently with non-linear motion and measurement functions
- KF, EKF, UKF, EIF suffer from complexity issues for large maps
- SEIF approximations lead to sub-quadratic memory and runtime complexity
- All filters presented so far, **require Gaussian distributions**

Slide Information

- These slides have been created by Cyrill Stachniss as part of the robot mapping course taught in 2012/13 and 2013/14. I created this set of slides partially extending existing material of Edwin Olson, Pratik Agarwal, and myself.
- I tried to acknowledge all people that contributed image or video material. In case I missed something, please let me know. If you adapt this course material, please make sure you keep the acknowledgements.
- Feel free to use and change the slides. If you use them, I would appreciate an acknowledgement as well. To satisfy my own curiosity, I appreciate a short email notice in case you use the material in your course.
- My video recordings are available through YouTube: http://www.youtube.com/playlist?list=PLgnQpQtFTOGQrZ4O5QzbIHgl3b1JHimN_&feature=g-list

Cyrill Stachniss, 2014
cyrill.stachniss@igg.uni-bonn.de