Probabilistic Robotics

The Sparse Extended Information Filter

MSc course Artificial Intelligence 2018
https://staff.fnwi.uva.nl/a.visser/education/ProbabilisticRobotics/

Arnoud Visser
Intelligent Robotics Lab
Informatics Institute
Universiteit van Amsterdam
A.Visser@uva.nl

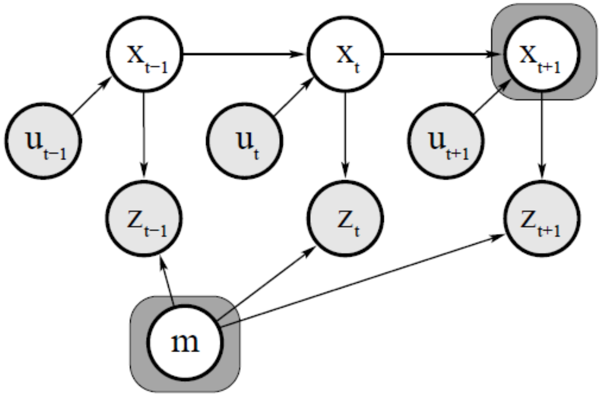
Images courtesy of Sebastian Thrun, Wolfram Burghard, Dieter Fox, Michael Montemerlo, Dick Hähnel, Pieter Abbeel and others.

Simultaneous Localization and Mapping

A robot acquires a map while localizing itself relative to this map.

Online SLAM problem

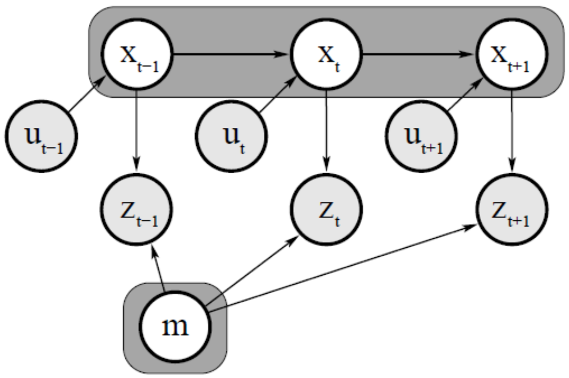
$$p(x_t, m \mid z_{1:t}, u_{1:t})$$



Estimate map *m* and current position x_t

Full SLAM problem

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



Estimate map *m* and driven path $x_{1:t}$

SEIF SLAM

SEIF SLAM reduces the state vector *y* again to the current position x_t

$$y_t = \left(x_t m_{1,x} m_{1,y} s_1 \cdots m_{N,x} m_{N,y} s_N\right)^T$$

This is the same state vector *y* as EKF SLAM

State estimate

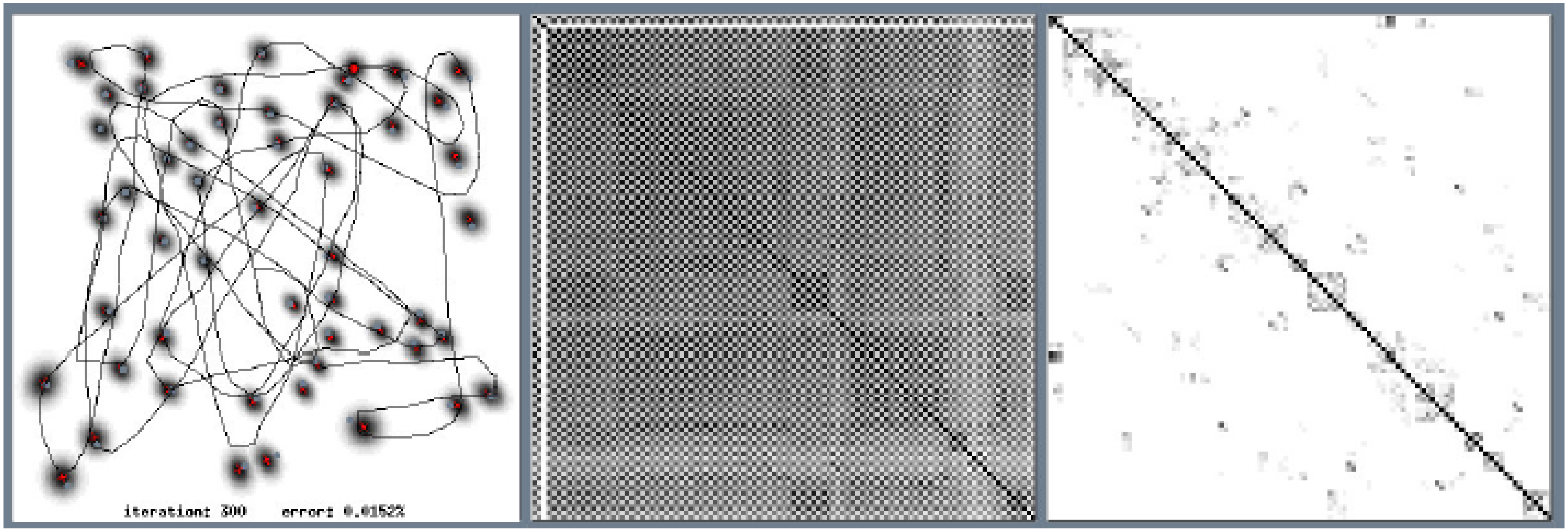
SEIF SLAM requires every timestep inference to estimate the state

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

The state estimated is also done by GraphSLAM,
as a post-processing step.

Sparseness of *Information Matrix*

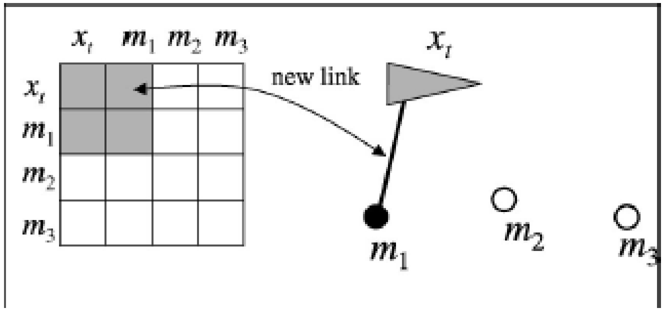
After a while, all landmarks are correlated in EKF's correlation matrix



The normalized information matrix is naturally sparse; most elements are close to zero (but none is zero).

Acquisition of the *information matrix*

The observation of a landmark m_1 introduces a constraint:



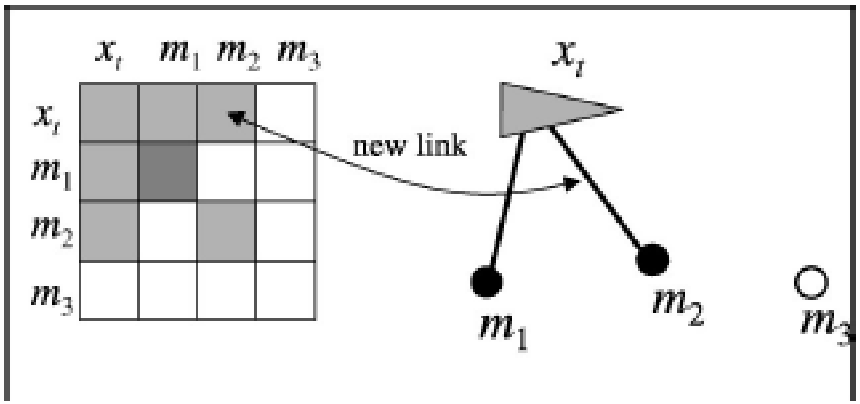
The constraint is of the type:

$$H_t^T Q_t^{-1} H_t$$

Where $h(x_t,m_j)$ is the measurement model and Q_t the covariance of the measurement noise.

Acquisition of the *information matrix*

The observation of a landmark m_2 introduces another constraint:

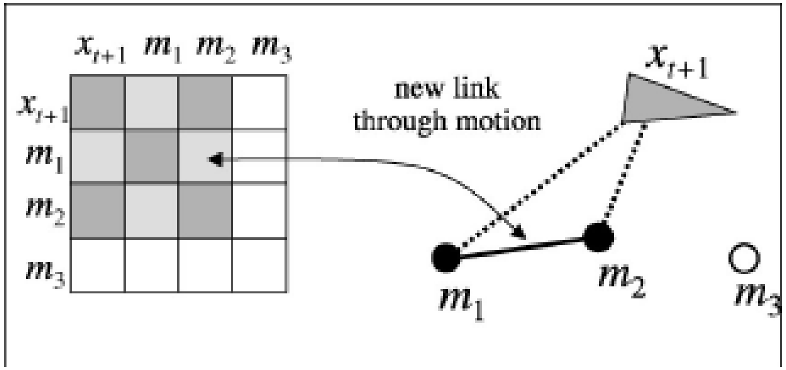


The *information vector* increases with the term:

$$H_t^T Q_t^{-1}(z_t^i - h(\bar{\mu}_t) + H_t \mu_t)$$

Acquisition of the *information matrix*

The movement of the robot from x_1 to x_2 also introduces an constraint:



The constraint is now between the landmarks m_1 and m_2 (and not between the path x_{t-1} to x_t):

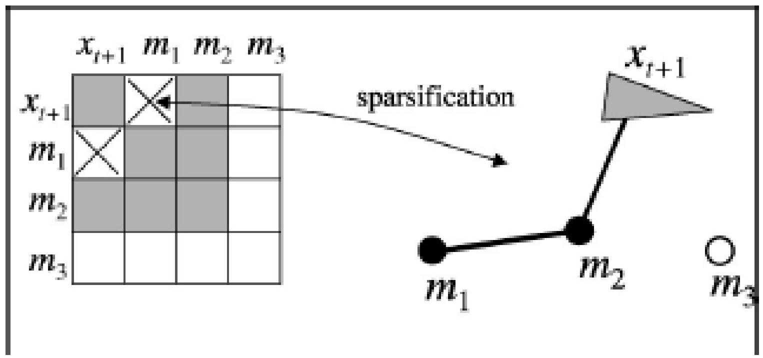
$$\overline{\Omega}_t = [G_t \Omega_{t-1}^{-1} G_t^T + F_x^T R_t F_x]^{-1}$$

Which can be simplified to

$$\overline{\Omega}_t = \Phi_t - \kappa_t$$

Acquisition of the *information matrix*

The *information matrix* can become really sparse by applying a *sparsification step*:



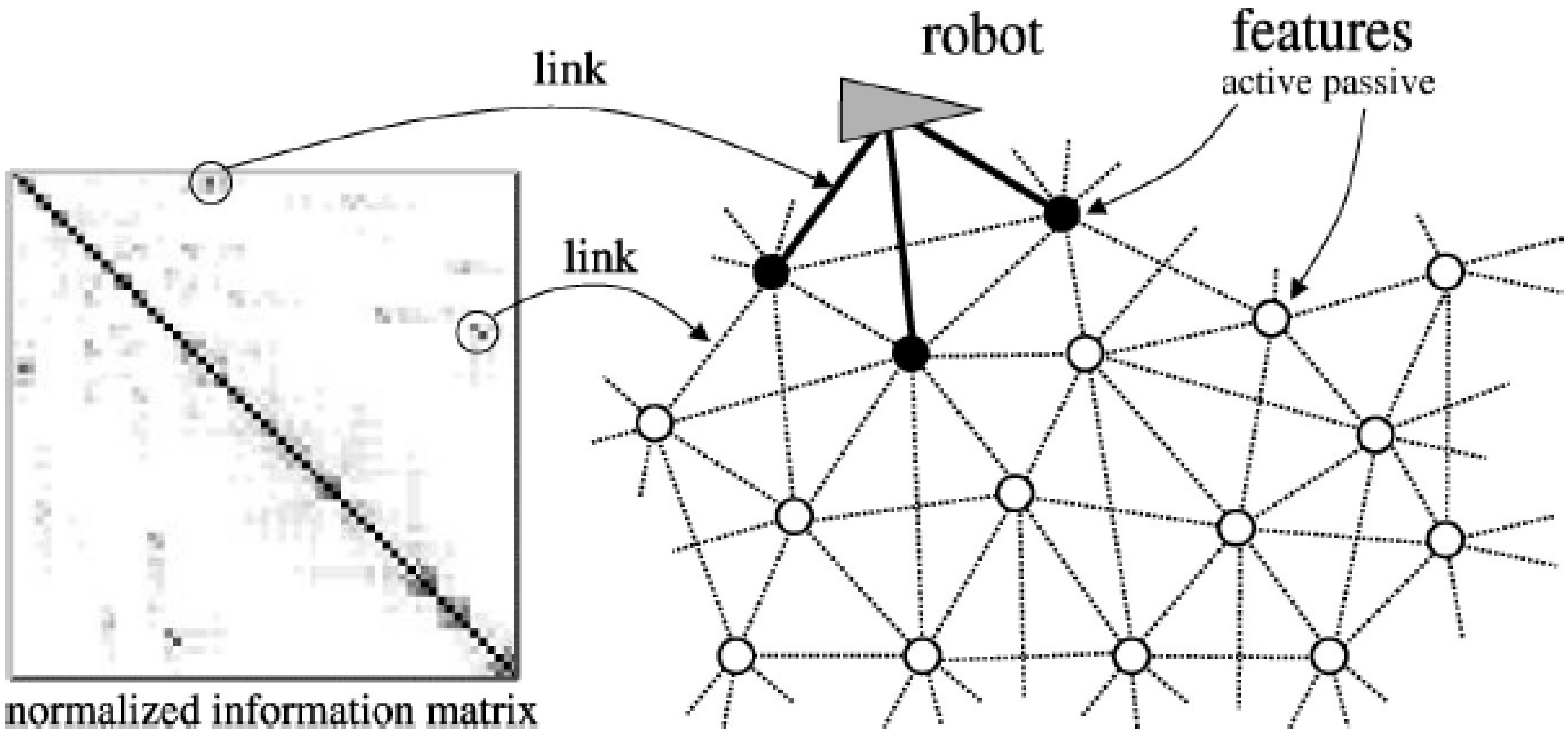
This is done by partition the set of features into three disjoint subsets:

$$m = m^{+} + m^{0} + m^{-}$$

Where m^- is the set of passive features and $m^+ \cap m^0$ is the set of active features. The number of features that are allowed to remain active (set m^+) is thresholded to guarantee efficiency.

Network of features

- Approximate the sparse information matrix with the argument that not all features are strongly connected:



Updating the current state estimate

The current state estimate $\hat{\mu}_t$ is needed every timestep:

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

Yet, from the current state estimate only subset is needed:

$$y_t = \left(x_t \cdots m^+_{1,x} \, m^+_{1,y} \, s^+_1 \cdots m^+_{2,x} \, m^+_{2,y} \, s^+_2 \cdots \right)^T$$

i.e. the robot position x_t and the locations of the active landmarks m^{+}.

This can be done with an iterative hill climbing algorithm:

$$\mu_i \leftarrow \left(F_i \Omega F_i^T \right)^{-1} F_i [\xi - \Omega \mu + \Omega F_i^T F_i \mu]$$

Where F_i is a projection matrix to extract element i from matrix Ω.

Full Algorithm

The algorithm combines the four steps;
two updates and two approximations:

$$\text{Algorithm SEIF\_SLAM\_known\_correspondences}(\xi_{t-1}, \Omega_{t-1}, \mu_{t-1}, u_t, z_t, c_t)$$

$$\bar{\xi}_t, \bar{\Omega}_t, \bar{\mu}_t = \text{SEIF\_motion\_update}(\xi_{t-1}, \Omega_{t-1}, \mu_{t-1}, u_t)$$

$$\tilde{\mu}_t = \text{SEIF\_update\_state\_estimate}(\bar{\xi}_t, \bar{\Omega}_t, \bar{\mu}_t)$$

$$\xi_t, \Omega_t = \text{SEIF\_measurement\_update}(\bar{\xi}_t, \bar{\Omega}_t, \tilde{\mu}_t, z_t, c_t)$$

$$\tilde{\xi}_t, \tilde{\Omega}_t = \text{SEIF\_sparsification}(\xi_t, \Omega_t)$$

$$\text{return } \tilde{\xi}_t, \tilde{\Omega}_t, \tilde{\mu}_t$$

SEIF_measurement_update

for all observed features $z_t^i = (r_t^i \; \phi_t^i \; s_t^i)$

Calculate $\hat{z}_t^i, H_t^i$

endfor

$$\xi_t = \bar{\xi}_t + \sum_i {H_t^i}^T \; {Q_t}^{-1} \left[z_t^i - \hat{z}_t^i + H_t^i \mu_t \right]$$

$$\Omega_t = \bar{\Omega}_t + \sum_i {H_t^i}^T \; {Q_t}^{-1} H_t^i$$

return ξ_t, Ω_t

SEIF_sparsification

Calculate $\breve{\Omega}_t = \Omega_t^1 - \Omega_t^2 + \Omega_t^3$ *with*

$\Omega_t^1 = p(x_t, m^+ | m^- = 0, z_{1:t}, u_{1:t}, c_{1:t})$

$\Omega_t^2 = p(m^+ | m^- = 0, z_{1:t}, u_{1:t}, c_{1:t})$

$\Omega_t^3 = p(m^+, m^0, m^- | z_{1:t}, u_{1:t}, c_{1:t})$

$$\breve{\xi}_t = \xi_t + \mu_t \left(\breve{\Omega}_t - \Omega_t \right)$$

return $\breve{\xi}_t, \breve{\Omega}_t$

SEIF_motion_update

$\bar{\Omega}_t = \Phi_t - \kappa_t$

$\bar{\mu}_t = \mu_{t-1} + F_x{}^T \delta$

$\bar{\xi}_t = \xi_{t-1} + (\lambda_t - \kappa_t)\mu_{t-1} + \bar{\Omega}_t F_x{}^T \delta$

return $\bar{\xi}_t, \bar{\Omega}_t, \bar{\mu}_t$

Canonical form of

$\bar{\Sigma}_t = G_t \Sigma_t G_t{}^T + F_x{}^T R_t F_x$

SEIF_update_state_estimate

For most map features m^{-}

$$\mu_{i,t} = \bar{\mu}_{i,t}$$

For a few features m^{+}

$$\mu_{i,t} = \left(F_i \Omega_t F_i{}^T\right)^{-1} F_i \left[\xi_t - \Omega_t \bar{\mu}_t + \Omega_t F_i{}^T F_i \bar{\mu}_t\right]$$

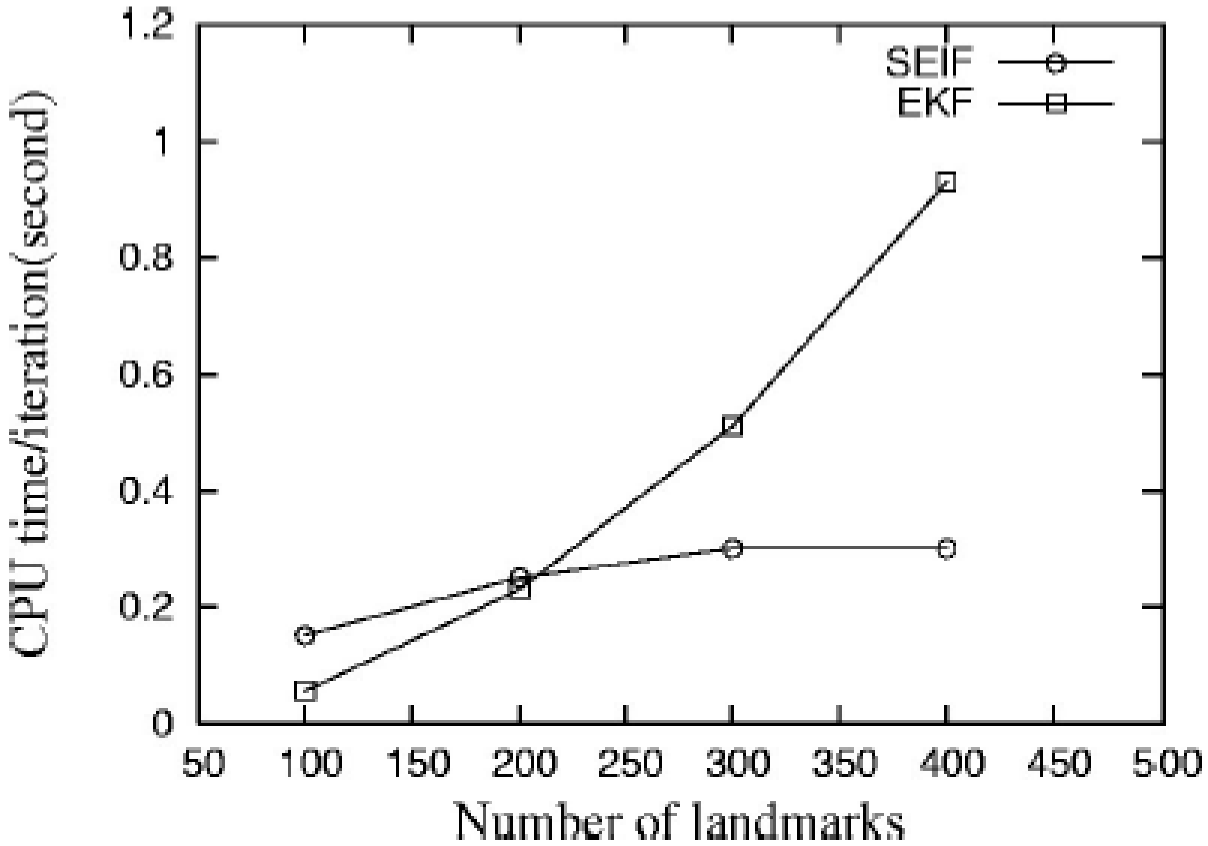
For the pose

$$\mu_{x,t} = \left(F_x \Omega_t F_x{}^T\right)^{-1} F_x \left[\xi_t - \Omega_t \bar{\mu}_t + \Omega_t F_x{}^T F_x \bar{\mu}_t\right]$$

return μ_t

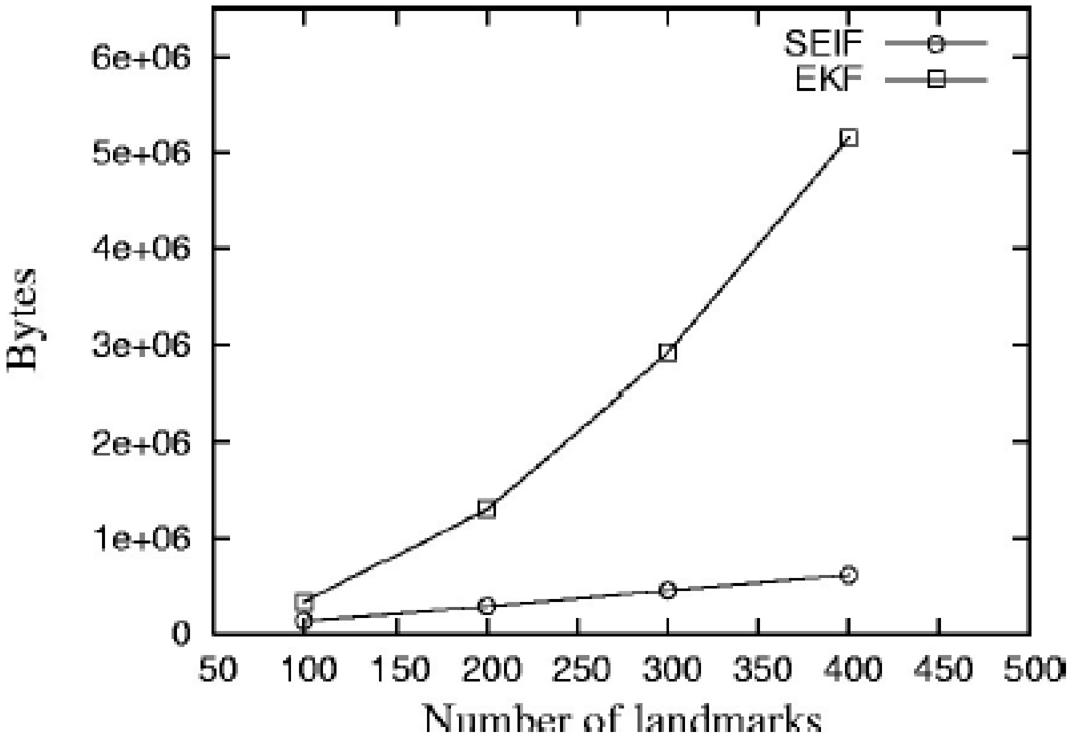
The effect of sparsification

The computation requires ‘constant’ time:



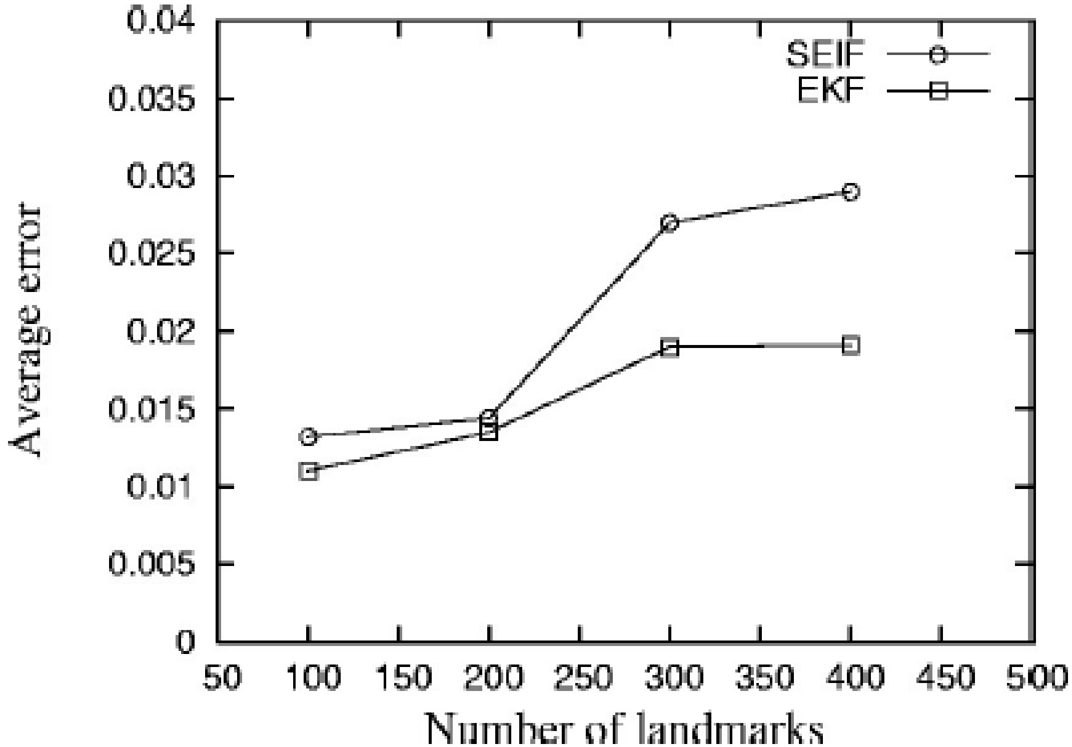
The effect of sparsification

The memory scales linearly:



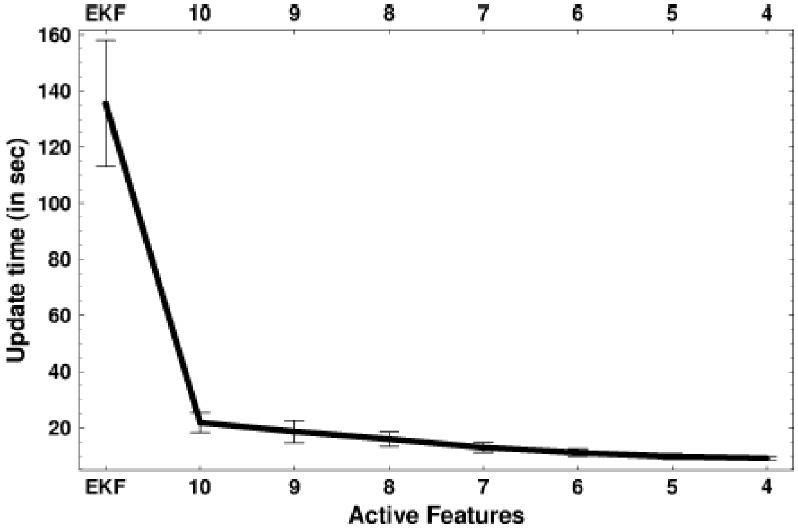
The effect of sparsification

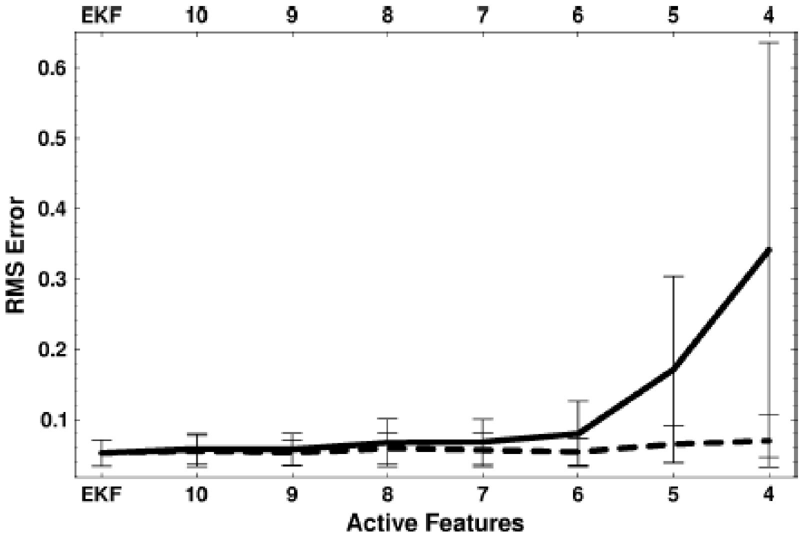
The prize is less accuracy, due to the approximation:



The degree of sparseness

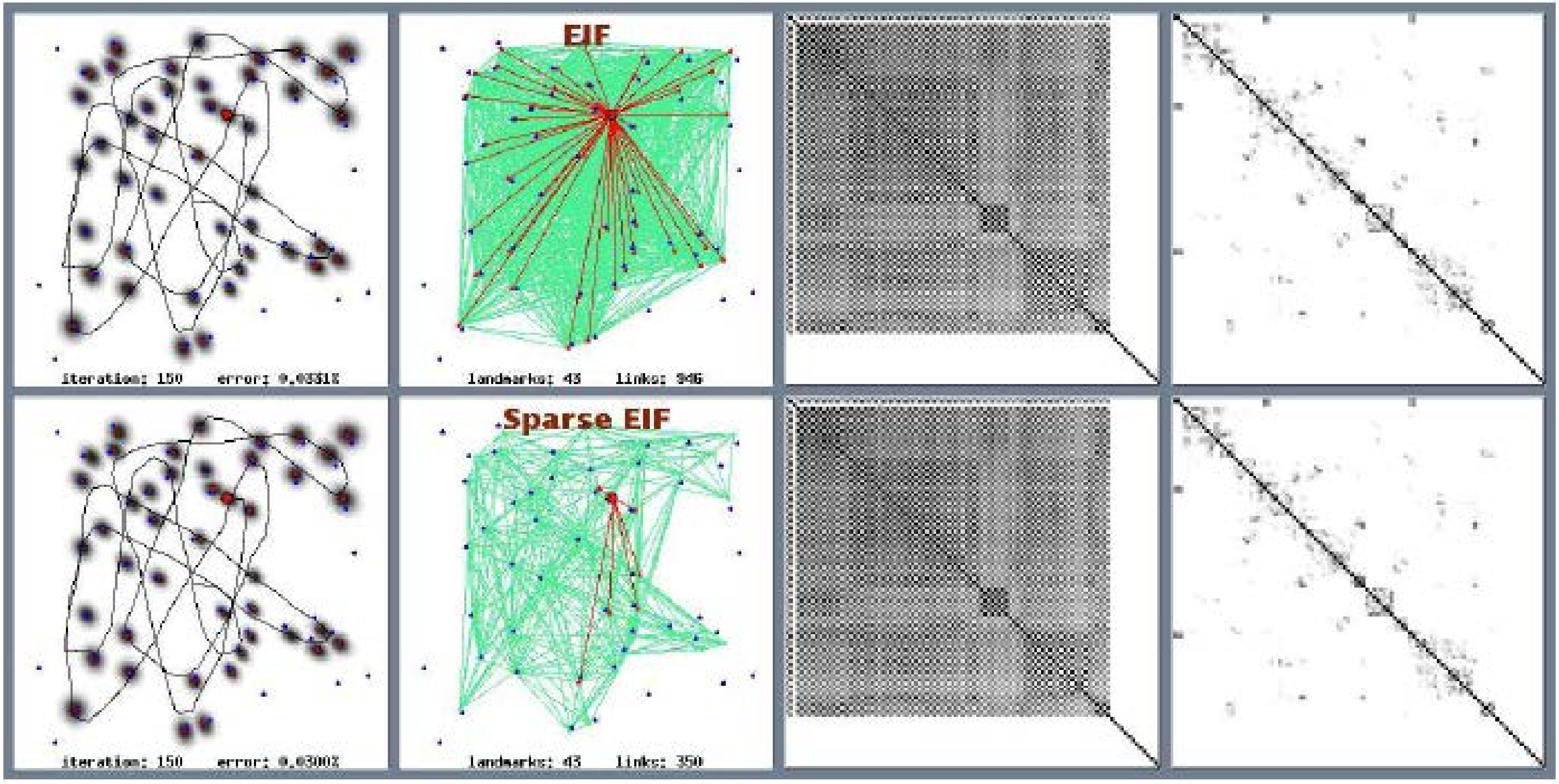
By choosing the number of active features,
accuracy can be traded against efficiency :





Effect of approximation

The effect of sparsification is less links between landmarks, more confidence, but nearly same information matrix:



P

Full Algorithm

To extend the algorithm for unknown correspondences, an estimate for the correspondence is needed:

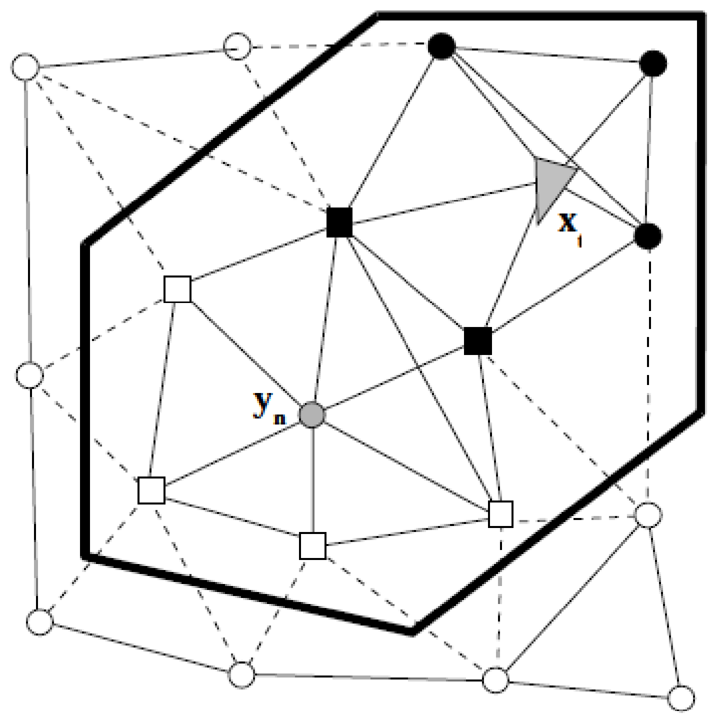
$$\hat{c}_t = \underset{c_t}{\operatorname{argmax}}\ p(z_t \mid z_{1:t-1}, u_{1:t}, \hat{c}_{1:t-1}, c_t)$$

$$\hat{c}_t = \underset{c_t}{\operatorname{argmax}} \int p(z_t \mid y_t, c_t)\, p(y_t \mid z_{1:t-1}, u_{1:t}, \hat{c}_{1:t-1}) dy_t$$

$$\hat{c}_t = \underset{c_t}{\operatorname{argmax}} \int\int p(z_t \mid x_t, y_{c_t}, c_t)\, p(x_t, y_{c_t} \mid z_{1:t-1}, u_{1:t}, \hat{c}_{1:t-1}) dx_t dy_{c_t}$$

Estimating the correspondence

To probability $p(x_t, y_{c_t} \mid z_{1:t-1}, u_{1:t}, \hat{c}_{1:t-1})$ can be approximated by the Markov blanket of all landmarks connected to robot pose x_t and landmark y_{c_t}



Correspondence test

Based on the probability that m_j corresponds to m_k:

$$\text{Algorithm SEIF\_correspondence\_test}(\Omega,\xi,\mu,m_j,c_k)$$

$$B = B(j) \cup B(k)$$

$$\Sigma_B = (F_B \Omega F_B{}^T)^{-1}$$

$$\mu_B = \Sigma_B F_B \xi$$

$$\Sigma_\Delta = (F_\Delta \Omega_B F_\Delta{}^T)^{-1}$$

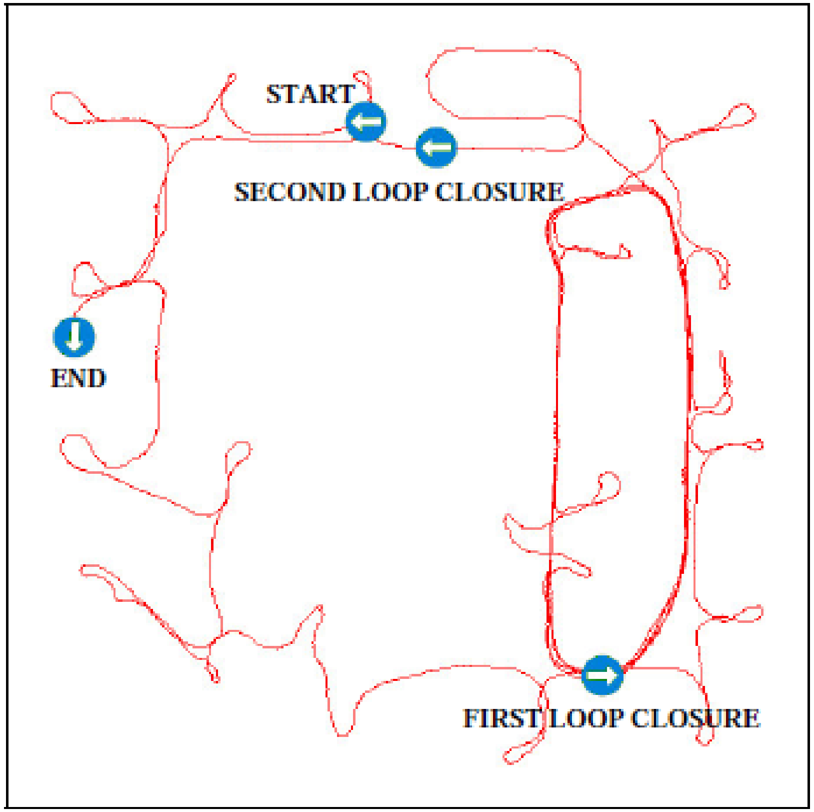
$$\mu_\Delta = \Sigma_\Delta F_\Delta \xi_B$$

$$\text{return} \quad \det(2\pi\Sigma_\Delta)^{-\frac{1}{2}} \exp\{-\tfrac{1}{2}\mu_\Delta^T \Sigma_\Delta^{-1} \mu_\Delta\}$$

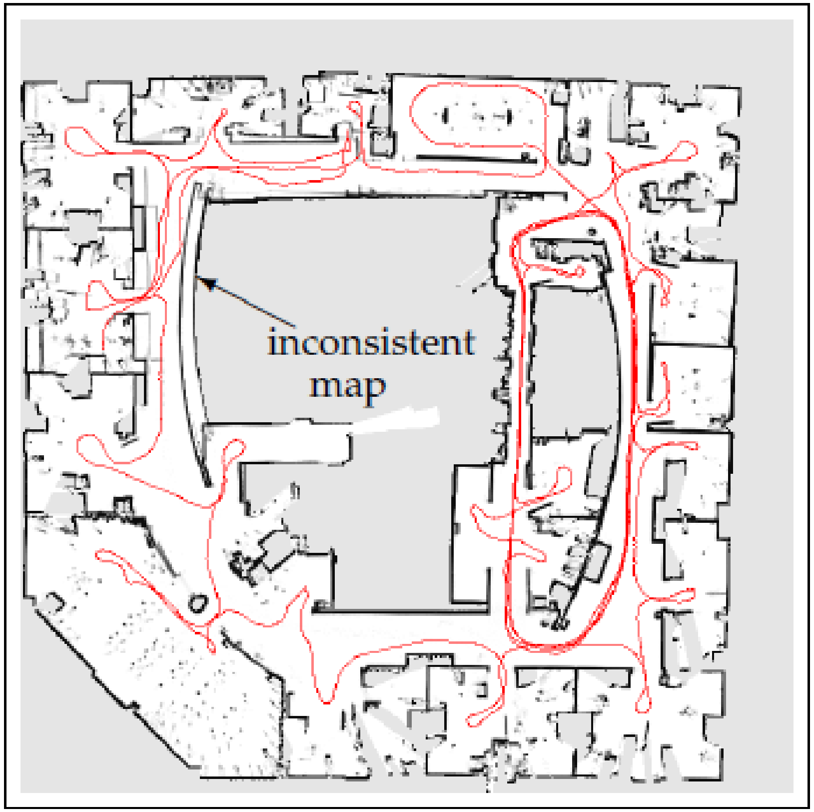
Results

MIT building (multiple loops):

(a) Robot path



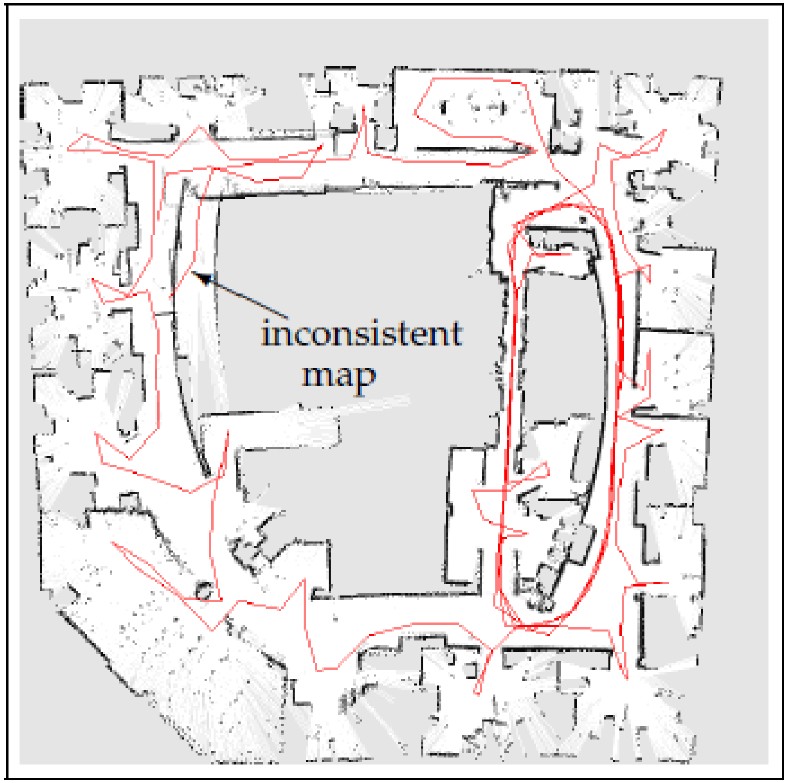
(b) Incremental ML (map inconsistent on left)



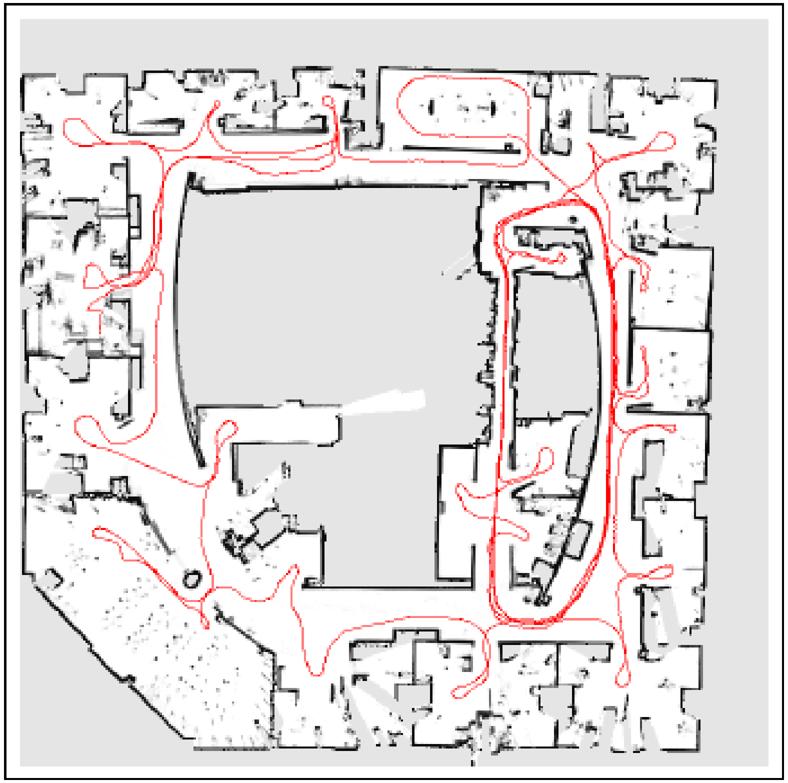
Results

MIT building (multiple loops):

(c) FastSLAM (see next Chapter)



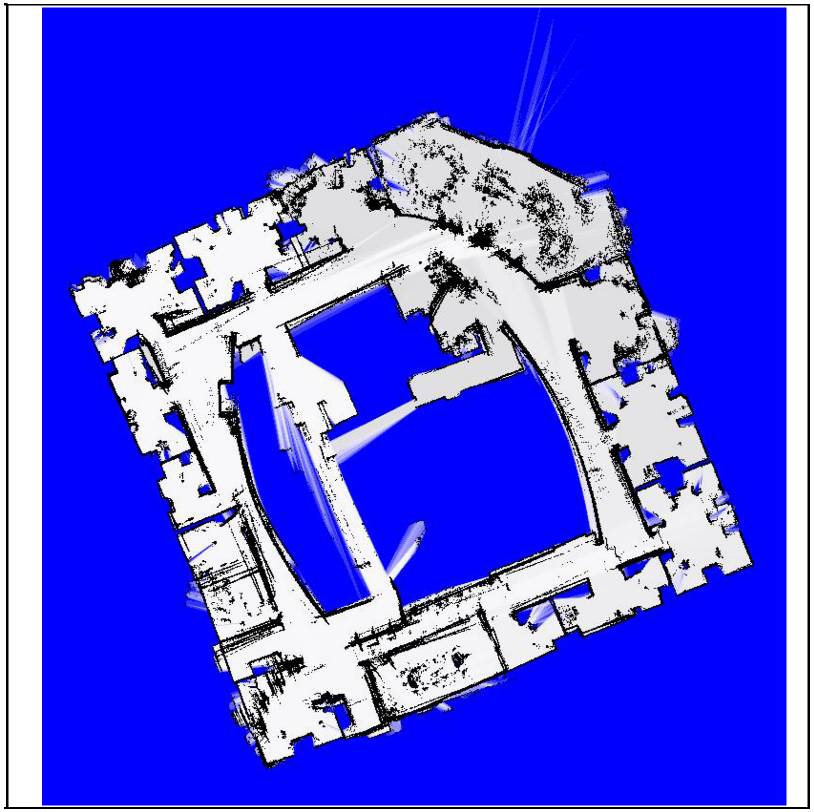
(d) SEIFs with branch-and-bound data association



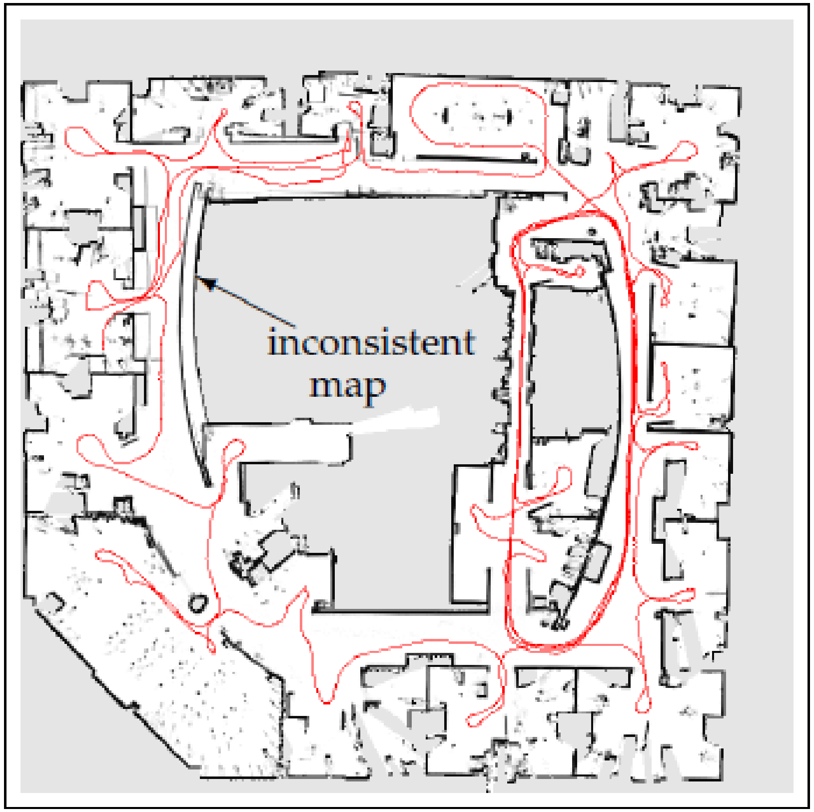
Results

MIT building (multiple loops):

UvA approach Q-WSM



(b) Incremental ML (map inconsistent on left)



Conclusion

The Sparse Extended Information Filter:

- Solves the Online SLAM problem efficiently.
- Where EKF spread the information of each measurement over the full map, SEIF limits the spread to ‘active features’.
- All information in the stored in the canonical parameterization. Yet, an estimate of the mean $\hat{\mu}_t$ is still needed. This estimate is found with a hill climbing algorithm (and not a inversion of the information matrix).
- The accuracy and efficiency can be balanced by selecting an appropriate number of ‘active features’.

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