TSRT14: Sensor Fusion
Lecture 9

— Simultaneous localization and mapping (SLAM)

Gustaf Hendeby

gustaf.hendeby@liu.se

LINKÖPING UNIVERSITY

Ad

PhD Thesis Defense: Anton Kullberg

Title: Dynamic rEvolution:
Adaptive state estimation via Gaussian processes and iterative filtering

When: June 14, 2024, at 9:30–12:00 (approx)

Where: Nobel, Physcis buildning, LiU

(Attending a PhD thesis defense counts as MSc thesis ascultation.)

Le 9: simultaneous localization and mapping (SLAM)

Whiteboard:

- SLAM problem formulation
- Framework for EKF-SLAM and FastSLAM (with PF and MPF)

Slides:

- Algorithms
- Properties
- Examples and illustrations

Lecture 8: summary

SIS PF Algorithm

Choose the number of particles N, a proposal density $q(x_k^{(i)}|x_{0:k-1}^{(i)}, y_{1:k})$, and a threshold N_{th} (for instance $N_{\text{th}} = \frac{2}{3}N$).

- *Initialization:* Generate $x_0^{(i)} \sim p_{x_0}, i = 1, \dots, N$.

Iterate for $k = 1, 2, \dots$:

1. *Measurement update:* For $i = 1, 2, \dots, N$:
   $w_{k|k}^{(i)} \propto w_{k|k}^{(i)} p(y_k|x_k^{(i)})$, and normalize $w_{k|k}^{(i)}$.
2. *Estimation:* MMSE $\hat{x}_{k|k} \approx \sum_{i=1}^N w_k^{(i)} x_{k|k}^{(i)}$.
3. *Resampling:* Resample with replacement when $N_{\text{eff}} = \frac{1}{\sum_i (w_{k|k}^{(i)})^2} < N_{\text{th}}$.
4. *Prediction:* Generate samples $x_{k+1}^{(i)} \sim q(x_k|x_{k-1}^{(i)}, y_k)$,
   update the weights $w_{k+1|k}^{(i)} \propto w_{k|k}^{(i)} \frac{p(x_k^{(i)}|x_{k-1}^{(i)})}{q(x_k^{(i)}|x_{k-1}^{(i)}, y_k)}$, and normalize $w_{k+1|k}^{(i)}$.

Simultaneous Localization and Mapping (SLAM)

LINKÖPING UNIVERSITY

SLAM: problem formulations

Localization concerns the estimation of pose from known landmarks.

Navigation concerns estimation of pose, velocities and other states from known landmarks.

Mapping concerns the estimation of landmark positions from known values of pose.

SLAM concerns the joint estimation of pose and landmark positions.

- Variations on the same theme: *Simultaneous navigation and mapping SNAM?!* and *Simultaneous tracking and mapping STAM?!*

EKF SLAM: model

- Assume a linear(-ized) model

$$\begin{aligned} x_{k+1} &= F x_k + G v_k & \text{cov}(v_k) &= Q \\ \mathbf{m}_{k+1} &= \mathbf{m}_k \\ y_k &= H_k^x x_k + H_k^{\mathbf{m}}(c_k^{1:I_k})\mathbf{m}_k + e_k, & \text{cov}(e_k) &= R. \end{aligned}$$

- The map is represented by $\mathbf{m}_k$.
- The index $c_k^{1:I_k}$ relate the observed landmark i to a map landmark j_i, which affects the measurement model.
- Association is crucial for some sensors (laser, radar, etc.), but less of a problem some applications (camera using image features, microphones using designed pings).
- The state and its covariance matrix

$$\hat{z}_{k|k} = \begin{pmatrix} \hat{x}_{k|k} \\ \hat{\mathbf{m}}_{k|k} \end{pmatrix}, \qquad P_{k|k} = \begin{pmatrix} P_{k|k}^{xx} & P_{k|k}^{x\mathbf{m}} \\ P_{k|k}^{\mathbf{m}x} & P_{k|k}^{\mathbf{m}\mathbf{m}} \end{pmatrix}.$$

Original SLAM Application

- Assume a ground robot with three states: $x = (X, Y, \psi)^T$.
- Robot measures speed and turn rate: $u = (v, \dot{\psi})^T$.
- Simple dynamics.
- Sensor:
  1. Ranging sensor (sonar, laser scanner, radar) measures distance to obstacles (walls, furniture); tens to hundreds of landmarks.
  2. Vision (camera, Kinect, stereo camera) provides detections (corners, markers, patterns) as potential landmarks; thousands or tens of thousands of landmarks.
- $P^{xx}_{k|k}$ small matrix, $P^{\mathbf{m}x}_{k|k}$ thin matrix and $P^{\mathbf{mm}}_{k|k}$ large matrix.

Approach?

Both EKF and PF apply to the problem, but how to handle the large dimensions in the best way? Start with studying the basic EKF.

EKF SLAM: basic KF steps

Time update:

$$\hat{z}_{k|k-1} = \begin{pmatrix} F & 0 \\ 0 & I \end{pmatrix} \hat{z}_{k-1|k-1},$$

$$P_{k|k-1} = \begin{pmatrix} F_k P^{xx}_{k-1|k-1} F_k^T + G_k Q_k G_k^T & F_k P^{x\mathbf{m}}_{k-1|k-1} \\ P^{\mathbf{m}x}_{k-1|k-1} F_k^T & P^{\mathbf{mm}}_{k-1|k-1} \end{pmatrix}$$

Measurement update:

$$\begin{aligned}
S_k &= H^x_k P^{xx}_{k|k-1} H^{xT}_k + H^{\mathbf{m}}_k P^{\mathbf{mm}}_{k|k-1} H^{\mathbf{m}T}_k + H^{\mathbf{m}}_k P^{\mathbf{m}x}_{k|k-1} H^{xT}_k + H^x_k P^{x\mathbf{m}}_{k|k-1} H^{\mathbf{m}T}_k + R_k \\
K^x_k &= \big(H^x_k P^{xx}_{k|k-1} + H^{\mathbf{m}}_k P^{\mathbf{m}x}_{k|k-1}\big) S_k^{-1} \\
K^{\mathbf{m}}_k &= \big(H^x_k P^{x\mathbf{m}}_{k|k-1} + H^{\mathbf{m}}_k P^{\mathbf{mm}}_{k|k-1}\big) S_k^{-1} \\
\varepsilon_k &= y_k - H^x_k \hat{x}_{k|k-1} - H^{\mathbf{m}}_k \hat{\mathbf{m}}_{k|k-1} \\
\hat{z}_{k|k} &= \hat{z}_{k|k-1} + \begin{pmatrix} K^x_k \\ K^{\mathbf{m}}_k \end{pmatrix} \varepsilon_k \\
P_{k|k} &= P_{k|k-1} - \begin{pmatrix} K^x_k \\ K^{\mathbf{m}}_k \end{pmatrix} S_k^{-1} \begin{pmatrix} K^x_k \\ K^{\mathbf{m}}_k \end{pmatrix}^T
\end{aligned}$$

EKF SLAM: KF problems

- All elements in $P^{\mathbf{mm}}_{k|k}$ are affected by the measurement update.
- It turns out that the cross correlations are essential for performance.
- No simple turn-around.

EKF SLAM: information form

- Focus on sufficient statistics and information matrix

$$\imath_{k|l} = \mathcal{I}_{k|l}\,\hat{z}_{k|l}$$

$$\mathcal{I}_{k|l} = P_{k|l}^{-1} = \begin{pmatrix} P_{k|l}^{xx} & P_{k|l}^{x\mathbf{m}} \\ P_{k|l}^{\mathbf{m}x} & P_{k|l}^{\mathbf{mm}} \end{pmatrix}^{-1} = \begin{pmatrix} \mathcal{I}_{k|l}^{xx} & \mathcal{I}_{k|l}^{x\mathbf{m}} \\ \mathcal{I}_{k|l}^{\mathbf{m}x} & \mathcal{I}_{k|l}^{\mathbf{mm}} \end{pmatrix}.$$

- Measurement update trivial

$$\imath_{k|k} = \imath_{k|k-1} + H_k^T R_k^{-1} y_k$$
$$\mathcal{I}_{k|k} = \mathcal{I}_{k|k-1} + H_k^T R_k^{-1} H_k.$$

Note:

The update is sparse!!!

EKF SLAM: information filter algorithm (1/4)

Initialization:

$$\imath^{x}_{1|0} = 0_{3\times 1}$$
$$\imath^{\mathbf{m}}_{1|0} = 0_{0\times 0}$$
$$\mathcal{I}^{xx}_{1|0} = 0_{3\times 3}$$
$$\mathcal{I}^{\mathbf{m}x}_{1|0} = 0_{0\times 3}$$
$$\mathcal{I}^{\mathbf{mm}}_{1|0} = 0_{0\times 0}$$

Note:

The information form allows for representing no prior knowledge with zero information (infinite covariance).

EKF SLAM: information filter algorithm (2/4)

1. **Associate** a map landmark $j = c_k^i$ to each observed landmark j, and construct the matrix $H_k^{\mathbf{m}}$. This step includes data gating for outlier rejection and track handling to start and end landmark tracks.
2. **Measurement update:**

$$
\begin{aligned}
\imath_{k|k}^{x} &= \imath_{k|k-1}^{x} + H_k^{xT} R_k^{-1} y_k \\
\imath_{k|k}^{\mathbf{m}} &= \imath_{k|k-1}^{\mathbf{m}} + H_k^{\mathbf{m}T} R_k^{-1} y_k \\
\mathcal{I}_{k|k}^{xx} &= \mathcal{I}_{k|k-1}^{xx} + H_k^{xT} R_k^{-1} H_k^{x} \\
\mathcal{I}_{k|k}^{x\mathbf{m}} &= \mathcal{I}_{k|k-1}^{x\mathbf{m}} + H_k^{xT} R_k^{-1} H_k^{\mathbf{m}} \\
\mathcal{I}_{k|k}^{\mathbf{mm}} &= \mathcal{I}_{k|k-1}^{\mathbf{mm}} + H_k^{\mathbf{m}T} R_k^{-1} H_k^{\mathbf{m}}
\end{aligned}
$$

Note:

$H_k^{\mathbf{m}}$ is very thick, but contains mostly zeros.
The low-rank sparse corrections influencing only a fraction of the matrix elements.

EKF SLAM: information filter algorithm (3/4)

3. Time update:

$$
\begin{aligned}
\bar{\mathcal{I}}^{xx}_{k|k-1} &= F_k^{-1}\, \mathcal{I}^{xx}_{k-1|k-1}\, F_k^{-T} \\
\bar{\mathcal{I}}^{x\mathbf{m}}_{k|k-1} &= F_k^{-1}\, \mathcal{I}^{x\mathbf{m}}_{k-1|k-1} \\
M_k &= G_k \big(G_k^T F_k^{-1}\, \mathcal{I}^{xx}_{k-1|k-1}\, F_k^{-T} + Q_k^{-1}\big)^{-1} G_k^T, \\
\mathcal{I}^{xx}_{k|k-1} &= \bar{\mathcal{I}}^{xx}_{k|k-1} - \bar{\mathcal{I}}^{xx}_{k|k-1} M_k \bar{\mathcal{I}}^{xx}_{k|k-1} \\
\mathcal{I}^{x\mathbf{m}}_{k|k-1} &= \bar{\mathcal{I}}^{x\mathbf{m}}_{k|k-1} - \bar{\mathcal{I}}^{xx}_{k|k-1} M_k \bar{\mathcal{I}}^{x\mathbf{m}}_{k|k-1}, \\
\mathcal{I}^{\mathbf{mm}}_{k|k-1} &= \bar{\mathcal{I}}^{\mathbf{mm}}_{k|k-1} - \bar{\mathcal{I}}^{\mathbf{m}x}_{k|k-1} M_k G_k^T \bar{\mathcal{I}}^{x\mathbf{m}}_{k|k-1} \\
\imath^{x}_{k|k-1} &= \big(I - \mathcal{I}^{xx}_{k|k-1}\, G_k Q_k G_k^T\, F_k^T\big)\, \imath^{x}_{k-1|k-1} \\
\imath^{\mathbf{m}}_{k|k-1} &= \imath^{\mathbf{m}}_{k-1|k-1} - \mathcal{I}^{\mathbf{m}x}_{k|k-1}\, G_k Q_k G_k^T\, F_k^T\, \imath^{x}_{k|k-1}
\end{aligned}
$$

Note:

Now, $\mathcal{I}^{\mathbf{mm}}_{k|k-1}$ is corrected with the inner product of $\bar{\mathcal{I}}^{x\mathbf{m}}_{k|k-1}$ which gives a full matrix. Many of the elements in $\mathcal{I}^{x\mathbf{m}}_{k|k-1}$ are close to zero and may be truncated!

EKF SLAM: information filter algorithm (4/4)

4. Estimation:

$$P_{k|k} = \mathcal{I}_{k|k}^{-1},$$
$$\hat{x}_{k|k} = P_{k|k}^{xx} \imath_{k|k}^{x} + P_{k|k}^{x\mathbf{m}} \imath_{k|k}^{\mathbf{m}},$$
$$\hat{\mathbf{m}}_{k|k} = P_{k|k}^{\mathbf{m}x} \imath_{k|k}^{x} + P_{k|k}^{\mathbf{m}\mathbf{m}} \imath_{k|k}^{\mathbf{m}}.$$

Here is another catch, the information matrix needs to be inverted! The pose is needed at all times for linearization and data gating. How to proceed?

Idea:

Solve

$$\imath_{k|l} = \mathcal{I}_{k|l}\, \hat{z}_{k|l},$$

directly using a gradient search algorithm initialized at previous estimate.

EKF SLAM: summary

- EKF SLAM scales well in state dimension.
- EKF SLAM scales badly in landmark dimension, though natural approximations exist for the information form.
- EKF SLAM is not robust to incorrect associations.

FastSLAM: idea

Basic factorization idea:

$$p(x_{1:k}, \mathbf{m}|y_{1:k}) = p(\mathbf{m}|x_{1:k}, y_{1:k})p(x_{1:k}|y_{1:k}).$$

- The first factor corresponds to a classical mapping problem, and is solved by the (E)KF.
- The second factor is approximated by the PF.
- Leads to a marginalized PF (MPF) where each particle is a pose trajectory with an attached map corresponding to mean and covariance of each landmark, but **no** cross-correlations.

FastSLAM: mapping solution (1/3)

Assume observation model linear(-ized) in landmark position

$$0 = h^0(y_k^n, x_k) + h^1(y_k^n, x_k) m_k^{l_n} + e_k^n, \qquad \text{cov}(e_k^n) = R_k^n.$$

This formulation covers:

- First order Taylor expansions.
- Bearing and range measurements, where $h^i(y_k^n, x_k)$ has two rows per landmark in 2D SLAM.
- Bearing-only measurements coming from a camera detection.

FastSLAM: mapping solution (2/3)

Linear estimation theory applies.

WLS estimate:

$$\hat{m}^l = \Big(\underbrace{\sum_{k=1}^N h^{1T}(y_k^{n_l}, x_k)(R_k^{n_l})^{-1} h^1(y_k^{n_l}, x_k)}_{\mathcal{I}_N^l}\Big)^{-1}$$

$$\underbrace{\sum_{k=1}^N -h^{1T}(y_k^{n_l}, x_k)(R_k^{n_l})^{-1} h^0(y_k^{n_l}, x_k)}_{\imath_N^l} = (\mathcal{I}_N^l)^{-1}\imath_N^l.$$

In this sum, h^i is an empty matrix if the map landmark n does not get an associated observation landmark at time k.

Under a Gaussian noise assumption, the posterior distribution is Gaussian

$$\big(m^l | y_{1:N}, x_{1:N}\big) \sim \mathcal{N}\big((\mathcal{I}_N^l)^{-1}\imath_N^l, (\mathcal{I}_N^l)^{-1}\big).$$

FastSLAM: mapping solution (3/3)

Kalman filter for mapping on information form

$$
\begin{aligned}
\imath_k^l &= \imath_{k-1}^l + h^{1T}(y_k^{n_l}, x_k) R_k^{-1} h^0(y_k^{n_l}, x_k),\\
\mathcal{I}_k^l &= \mathcal{I}_{k-1}^l + h^{1T}(y_k^{n_l}, x_k) R_k^{-1} h^1(y_k^{n_l}, x_k),\\
\hat{m}^l &= (\mathcal{I}_k^l)^{-1} \imath_k^l.
\end{aligned}
$$

Note the problem of inverting a large matrix to compute the landmark positions.

Likelihood in the Gaussian case:

$$
\begin{aligned}
&p(y_k^{n_l} | y_{1:k-1}, x_{1:k})\\
&\qquad = \mathcal{N}\big(h^0(y_k^{n_l}, x_k) + h^1(y_k^{n_l}, x_k)\hat{m}_{k-1}^l, R_k^{n_l} + h^1(y_k^{n_l}, x_k)(\mathcal{I}_k^l)^{-1} h^{1T}(y_k^{n_l}, x_k)\big).
\end{aligned}
$$

FastSLAM: the algorithm (1/2)

1. **Initialize** the particles
$$x_1^{(i)} \sim p_0(x),$$
where N denotes the number of particles.
2. **Data association** that assigns a map landmark n_l to each observed landmark l. Initialize new map landmarks if necessary.
3. **Importance weights**
$$w_k^{(i)} = \prod_l \mathcal{N}\big(h^0(y_k^l, x_k) + h^1(y_k^l, x_k)\hat{m}_{k-1}^{n_l}, R_k^l + h^1(y_k^l, x_k)(\mathcal{I}_k^{n_l})^{-1} h^{1T}(y_k^l, x_k)\big).$$
where the product is taken over all observed landmarks l, and normalize $\bar{w}_k^{(i)} = w_k^{(i)} / \sum_{j=1}^N w_k^{(j)}$.
4. **Resampling** a new set of particles with replacement
$$\Pr(x_k^{(i)} = x_k^{(j)}) = \bar{w}_k^{(j)}, \quad j = 1, \ldots, N.$$

FastSLAM: the algorithm (2/2)

5. Map measurement update:

$$p(\mathbf{m}^{(i)}|x_{1:k}^{(i)}, y_{1:k}) = \mathcal{N}\big((\mathcal{I}_k^{(i)})^{-1}\imath_k^{(i)}, (\mathcal{I}_k^{(i)})^{-1}\big),$$

$$\imath_k = \imath_{k-1} + h^{1T}(y_k, x_k)R_k^{-1}h^0(y_k, x_k),$$

$$\mathcal{I}_k = \mathcal{I}_{k-1} + h^{1T}(y_k, x_k)R_k^{-1}h^1(y_k, x_k).$$

6. Pose time update:

fastSLAM 1.0 (SIR PF)

$$x_{k+1}^{(i)} \sim p(x_{k+1}|x_{1:k}^{(i)}).$$

fastSLAM 2.0 (PF with optimal proposal)

$$x_{k+1}^{(i)} \sim p(x_{k+1}|x_{1:k}^{(i)}, y_{1:k+1})$$

$$\propto p(x_{k+1}|x_{1:k}^{(i)})p(y_{k+1}|x_{k+1}, x_{1:k}^{(i)}, y_{1:k}).$$

FastSLAM: summary

FastSLAM is ideal for a ground robot with three states and vision sensors providing thousands of landmarks.

- FastSLAM scales linearly in landmark dimension.
- As the standard PF, FastSLAM scales badly in the state dimension.
- FastSLAM is relatively robust to incorrect assocations, since associations are local for each particle and not global as in EKF-SLAM.

GraphSLAM: SLAM as a batch/smoothing problem

- EKF SLAM and FastSLAM both attempt to estimate the posterior distribution $p(x_k, \mathbf{m}|y_{1:k})$ recursively.
- The batch or smoothing problem is to find $p(x_{1:k}, \mathbf{m}|y_{1:k})$.
- In GraphSLAM, the batch problem is formulated as a factor graph, for which there are efficient solvers: GT-SAM, g2o, ceres, ...
- GraphSLAM has gained popularity as computing power has increased, and is today a go to state-of-the-art SLAM solution.

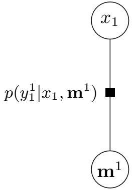
GraphSLAM: illustration

1. Initial state, $k = 1$

$$p(x_1) \quad \propto \quad p(x_1) \qquad = \quad p(x_1)$$

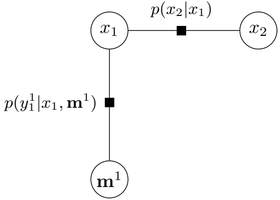
GraphSLAM: illustration

1. Initial state, $k = 1$
2. Observation of landmark $\mathbf{m}_1$



$$p(x_1, \mathbf{m}|y_1) \quad \propto \quad p(y_1^1|x_1, \mathbf{m}^1)\, p(x_1) \quad = \quad p(y_1^1|x_1, \mathbf{m}^1)\, p(x_1)$$

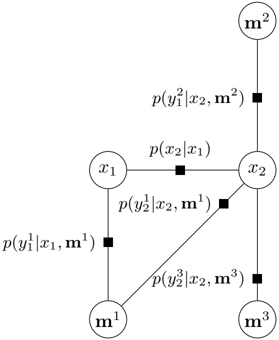
GraphSLAM: illustration



1. Initial state, $k = 1$
2. Observation of landmark $\mathbf{m}_1$
3. Propagate to $k = 2$.

$$p(x_{1:2}, \mathbf{m}|y_1) \quad \propto \quad p(x_2|x_1)\, p(x_1, \mathbf{m}^1|y_1) \quad = \quad p(y_1^1|x_1, \mathbf{m}^1)\, p(x_2|x_1) p(x_1)$$

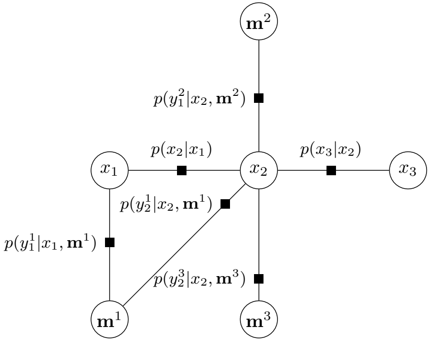
GraphSLAM: illustration



1. Initial state, $k = 1$
2. Observation of landmark $\mathbf{m}_1$
3. Propagate to $k = 2$.
4. Observe $\mathbf{m}^1$, $\mathbf{m}^2$, $\mathbf{m}^3$

$$p(x_{1:2}, \mathbf{m}|y_{1:2}) \quad \propto \quad \prod_{j=1}^{3} p(y_2^j|x_2, \mathbf{m}^j)\, p(x_{1:2}, \mathbf{m}^1|y_1) \quad = \quad \prod_{k=1}^{2} \prod_{j} p(y_k^j|x_k, \mathbf{m}^j)\, p(x_2|x_1) p(x_1)$$

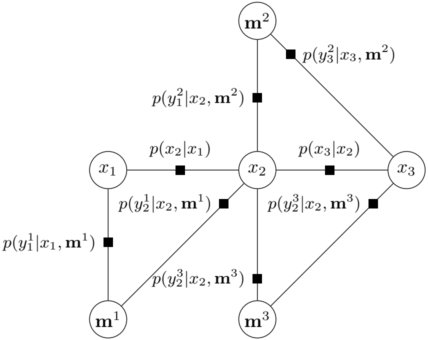
GraphSLAM: illustration



1. Initial state, $k = 1$
2. Observation of landmark $\mathbf{m}_1$
3. Propagate to $k = 2$.
4. Observe $\mathbf{m}^1$, $\mathbf{m}^2$, $\mathbf{m}^3$
5. Propagate to $k = 3$.

$$p(x_{1:3}, \mathbf{m}|y_{1:2}) \quad \propto \quad p(x_3|x_2)\, p(x_{1:2}, \mathbf{m}|y_{1:2}) \quad = \quad \prod_{k=1}^{2} \prod_{j} p(y_k^j|x_k, \mathbf{m}^j) \prod_{k=1}^{2} p(x_{k+1}|x_k) p(x_1)$$

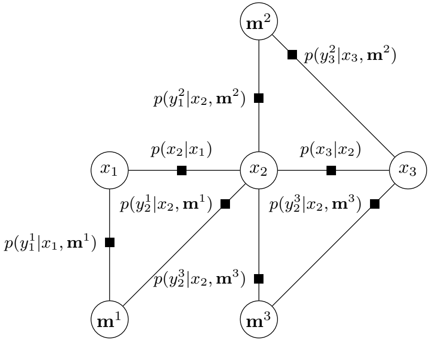
GraphSLAM: illustration



1. Initial state, $k=1$
2. Observation of landmark $\mathbf{m}_1$
3. Propagate to $k=2$.
4. Observe $\mathbf{m}^1$, $\mathbf{m}^2$, $\mathbf{m}^3$
5. Propagate to $k=3$.
6. Observe $\mathbf{m}^2$, $\mathbf{m}^3$

$$p(x_{1:3},\mathbf{m}|y_{1:3}) \quad \propto \quad \prod_{j=2}^{3} p(y_3^j|x_3,\mathbf{m}^j)\, p(x_{1:3},\mathbf{m}|y_{1:2}) \quad = \quad \prod_{k=1}^{3}\prod_{j} p(y_k^j|x_2,\mathbf{m}^j)\prod_{k=1}^{2} p(x_{k+1}|x_k)p(x_1)$$

GraphSLAM: illustration



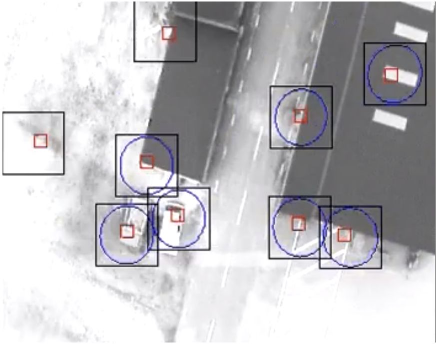
1. Initial state, $k = 1$
2. Observation of landmark $\mathbf{m}_1$
3. Propagate to $k = 2$.
4. Observe $\mathbf{m}^1$, $\mathbf{m}^2$, $\mathbf{m}^3$
5. Propagate to $k = 3$.
6. Observe $\mathbf{m}^2$, $\mathbf{m}^3$

Feed the graph representation to a solver to obtain an estimate.
Assuming Gaussian noise, the solvers internally solve a NWLS problem.

SLAM Illustration

- Airborne simultanous localization and mapping (SLAM) using UAV with camera.
- Research collaboration with IDA.
- **General idea:** augment state vector with parameters representing the map.

Comparison of EKF and FastSLAM on same dataset.



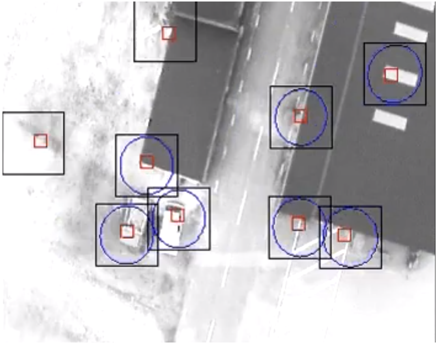
http://youtu.be/VQ1aCH13Yc4

http://youtu.be/hA_NZeuoy9Y

SLAM Illustration

- Airborne simultanous localization and mapping (SLAM) using UAV with camera.
- Research collaboration with IDA.
- **General idea:** augment state vector with parameters representing the map.

Comparison of EKF and FastSLAM on same dataset.

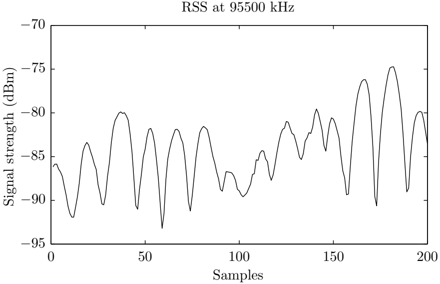


http://youtu.be/VQ1aCH13Yc4

http://youtu.be/hA_NZeuoy9Y

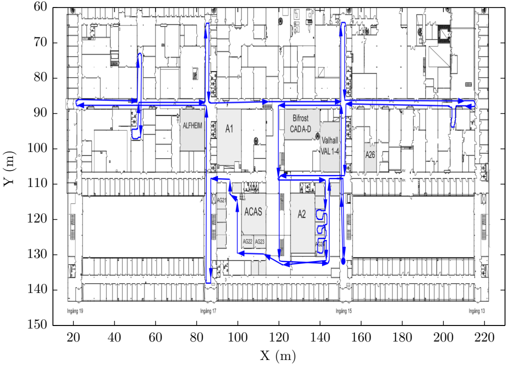
Ex: RSS SLAM (1/2)

- Prize winning MSc thesis at FOI 2014.
- Foot-mounted IMU for odometry.
- Opportunistic radio signals for fingerprinting.



Ex: RSS SLAM (2/2)

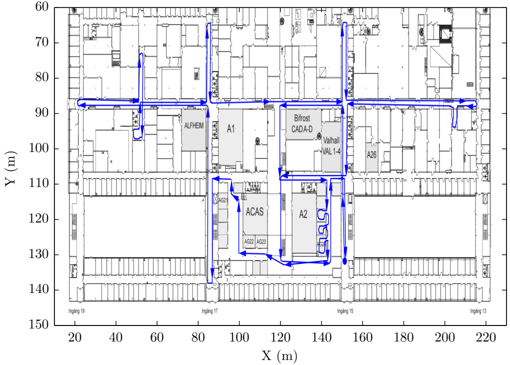
- Estimate the error in the odometry.
- Gaussian process to represent the map.
- Particle filter solution.



https://youtu.be/1KD2vwmfN0O

Ex: RSS SLAM (2/2)

- Estimate the error in the odometry.
- Gaussian process to represent the map.
- Particle filter solution.



https://youtu.be/1KD2vwmfNOO

SLAM Examples on Youtube

- Indoor mapping by UAV with laser scanner http://youtu.be/IMSozUpFFkU
- Indoor mapping using hand-held stereo camera with IMU at FOI http://youtu.be/7f-nqXmo1qE
- Intelligent vacuum cleaner using ceiling vision http://youtu.be/bq5HZzGF3vQ
- GraphSLAM https://youtu.be/d9ItdnJFOhU

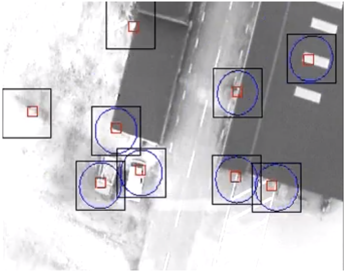
Summary

LINKÖPING UNIVERSITY

Lecture 9: summary

Simultaneous Localization And Mapping (SLAM)

- Joint estimation of trajectory $x_{1:k}$ and map parameters θ in sensor model $y_k = h(x_k; \theta) + e_k$.
- Algorithms:
  - EKF-SLAM: EKF (information form) on augmented state vector $z_k = (x_k^T, \theta^T)^T$.
  - FastSLAM: MPF on augmented state vector $z_k = (x_k^T, \theta^T)^T$.
  - GraphSLAM: Formulate the problem as a graph, and use dedicated solvers.



http://youtu.be/VQlaCHl3Yc4

http://youtu.be/hA_NZeuoy9Y

Gustaf Hendeby

gustaf.hendeby@liu.se

www.liu.se

LINKÖPING UNIVERSITY