L12: Smoothing and Mapping (SAM)

* Summary of L11 (Data Association)

$c^i = \underset{j}{\mathrm{argmin}} \| m_j - z^i \|_2$ Euclidean Nearest Neighbour

$c^i = \underset{j}{\mathrm{argmin}} \| m_j - z^i \|_{\Sigma_j}$ Mahalanobis N.N.

$c_t^* = \underset{c_t}{\mathrm{argmax}} \{ p(z_t | c_t, y_t) \}$ Maximum Likelihood

$\max(\Pi p) = \Pi \underbrace{\max(p)}_{\text{equivalent to Mahalanobis N.N.}} \longrightarrow \min(-\log p)$ ($\|\cdot\|_{\Sigma}$)

JCBB: Joint Compatibility Branch and Bound

Evaluates hypothesis recursively and eliminates branches

$d^2_{\mathcal{H}_t^i} < \chi^2_{d,\alpha}$ \ α confidence level $d = \dim(f_{\mathcal{H}_t^i})$

chi squared table for different α, d

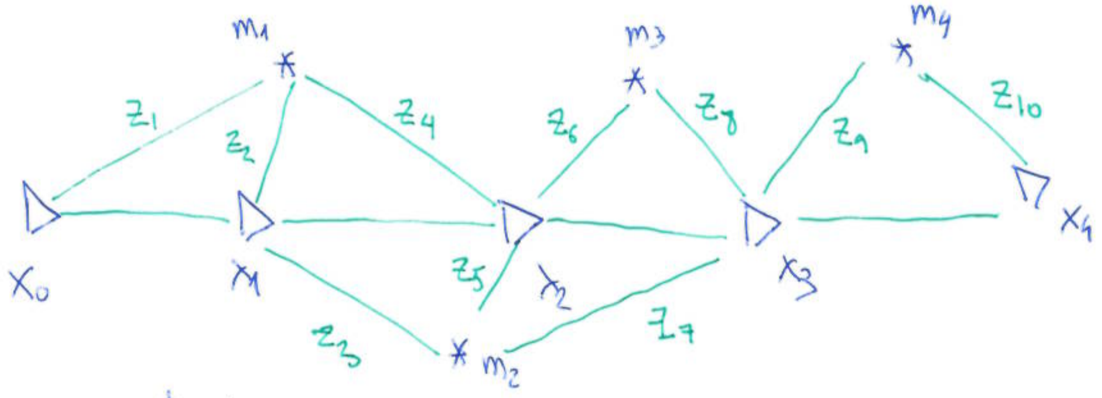
* SAM as a full SLAM problem (almost)

Smoothing → the robot trajectory $x_{1:t}$

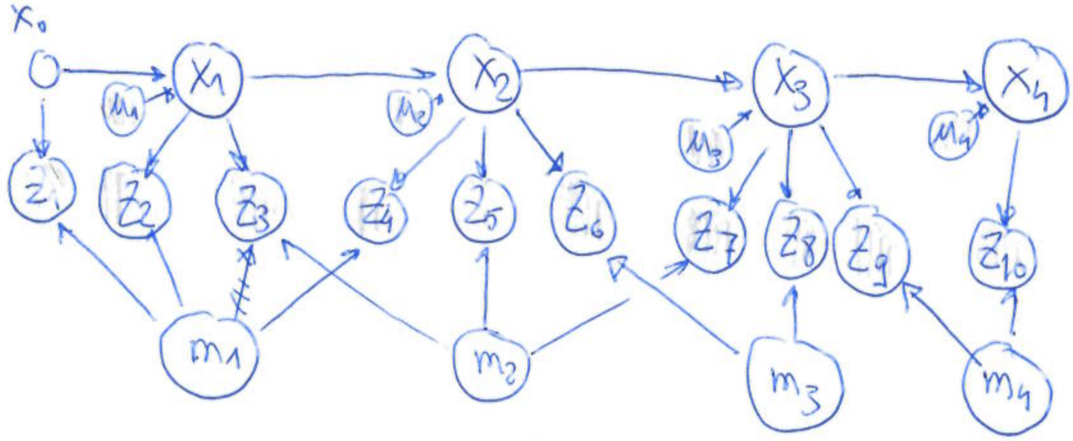
mapping → Set of landmarks (today) or any other representation

- Alternative to filter-based (L11) SLAM and still efficient
- Keeping the full trajectory is beneficial rather than just x_t.

*SAM as a Bayes Network



Graphical models are a powerful tool to describe SLAM



$$P(X, M, Z, U) = p(x_0) \prod_{i=1}^{M} p(x_i \mid x_{i-1}, u_i) \cdot \prod_{k=1}^{K} p(z_k \mid x_{i_k}, m_{j_k})$$

$\{x_{0:t}\}$ $\{m_j\}$

transitions

landmark observations

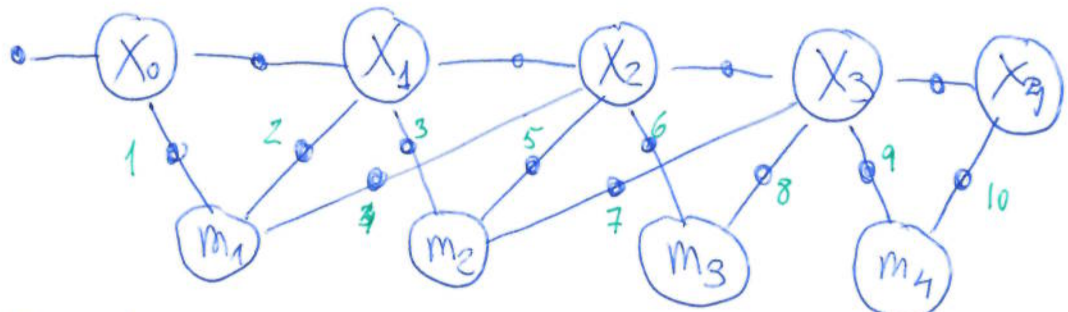
Objective: maximize the joint probability

Optimizing the graphical models dives us into:

- Graph theory
- Linear Algebra.

* SAM as a Factor Graph : (Bipartite graph)

- Bayesian networks are a natural way to express relations
- FG's have a tighter connection to optimization.



Eliminate Z, U as variables, now they are factors expressing the distributions between variables (X, M)

$$\Theta = \{X, M, Z, U\}, \qquad p(\Theta) = \prod \phi_i(\theta_i) \cdot \prod \psi_{ij}(\theta_i, \theta_j)$$

$$\phi_0(x_0) \propto p(x_0)$$

$$\psi_{(i-1)i}(x_{i-1}, x_i) \propto p(x_i \mid x_{i-1}, u_t)$$

$$\psi_{i_k, j_k}(x_{i_k}, m_{j_k}) \propto p(z_k \mid x_{i_k}, m_{j_k})$$

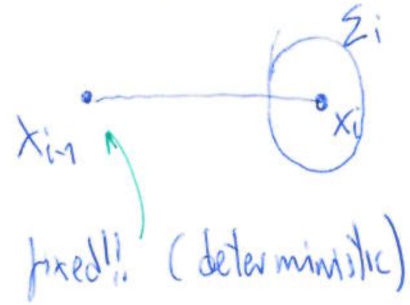
Same identical representation as Bayes network if factors are defined this way.

→ Transition model: (R_i)

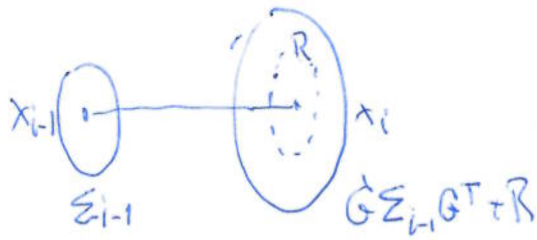
$$p(x_i \mid x_{i-1}, u_i) = N(x_i \,;\, g_i(x_{i-1}, u_i), \Sigma_{u_i})$$

$$= \eta \exp\left\{ -\frac{1}{2} \| g_i(x_{i-1}, u_i) - x_i \|^2_{\Sigma_i} \right\}$$

Ex: $\psi_{(i-1)i}$

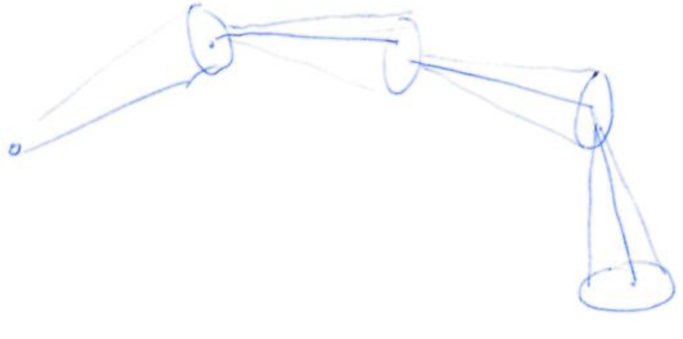


EKF.



propagation model
- Odometry
- unicycle kinematic
- car kinematic, etc.

Smoothing: optimization of the chain trajectory



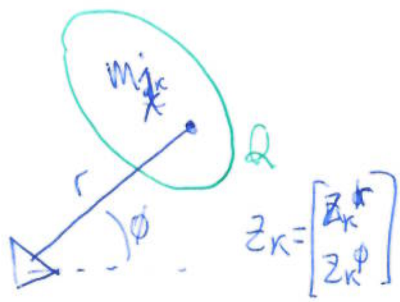
Relative relations $(i-1, i)$

Covariances do not get propagated over the state variables.

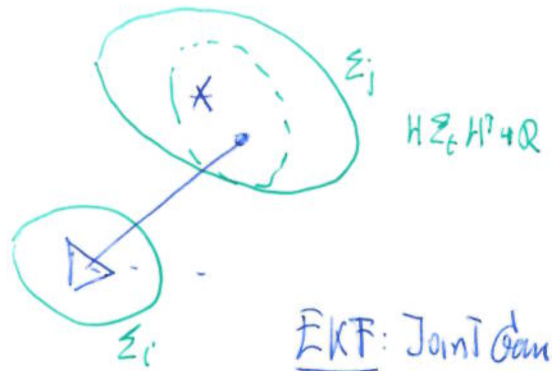
→ Observation model

$$p(z_k \mid x_{i_k}, m_{j_k}) = \mathcal{N}\left(z_k \mid h_k(x_{i_k}, m_{j_k}), \underset{(Q)}{\Sigma_k}\right)$$

$$= \eta \exp\left\{ -\frac{1}{2} \| h_k(x_{i_k}, m_{j_k}) - z_k \|^2_{\Sigma_k} \right\}$$



fixed: x_{i_k}, m_{j_k}



EKF: Joint Gaussi

* Solving SAM

$$\theta^* = \underset{\theta}{\operatorname{argmax}}\, p(X, M \mid Z, U) = \underset{\theta}{\operatorname{argmax}}\, p(X, M, Z, U)$$

$$= \underset{\theta}{\operatorname{argmin}} \left\{ -\log p(X, M, Z, U) \right\}$$

$$= \underset{\theta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{M} \| g_i(x_{i-1}, u_i) - x_i \|^2_{\Sigma_i} + \sum_{k=1}^{K} \| h_k(x_{i_k}, m_{j_k}) - z_k \|^2_{\Sigma_k} \right\}$$

Non-linear least squares problem (NLLSQ)

Equivalent to EKF of the augmented state $x_{0:t}$.

Linearize the NLLSQ

fixed / error perturbation

$$g_i(x_{i-1}, u_i) - x_i \simeq \Big[\underbrace{g_i(x^0_{i-1}, u_i)} + \underbrace{G_i^{i-1}\, \delta x_{i-1}} \Big] - \Big[\underbrace{x^0_i} + \underbrace{\delta x_i} \Big]$$

residual   Variables

$$= \left(G_i^{i-1}\, \delta x_{i-1} - \delta x_i \right) - a_i$$

Jacobian
$$G_i^{i-1} = \left. \frac{\partial g_i(x_{i-1}, u_i)}{\partial x_{i-1}} \right|_{x^0_{i-1}}$$

$$h_k(x_{i_k}, m_{j_k}) - z_k \simeq h_k(x^0_{i_k}, m^0_{j_k}) + H_k^{i_k}\, \delta x_{i_k} + J_k^{j_k}\, \delta m_{j_k} - z_k$$

$$= \left(H_k^{i_k}\, \delta x_{i_k} + J_k^{j_k}\, \delta m_{j_k} \right) - c_k$$

Jacobian:
$$H_k^{i_k} = \left. \frac{\partial h}{\partial x_{i_k}} \right|_{(x^0_{i_k}, m^0_{j_k})}$$

$$J_k^{j_k} = \left. \frac{\partial h}{\partial m_{j_k}} \right|_{(x^0_{i_k}, m^0_{j_k})}$$

* Linearized LSQ

$$\delta^* = \underset{\delta}{\arg\min} \left\{ \sum_{i=1}^{M} \| G_i^{i-1} \delta x_{i-1} - I \delta x_i - a_i \|_{\Sigma_i}^2 + \right.$$

$$\left. + \sum_{k=1}^{K} \| H_k^{i_k} \delta x_{i_k} + J_k^{j_k} \delta m_{j_k} - c_k \|_{\Sigma_k}^2 \right\}$$

How to simplify this expression?

$$\| e \|_{\Sigma}^2 = e^T \Sigma^{-1} e = e^T \left(\sqrt{\Sigma} \sqrt{\Sigma}^T \right)^{-1} e$$

$$= e^T \Sigma^{-\frac{1}{2}} \Sigma^{-\frac{T}{2}} e = \| \Sigma^{-T/2} e \|_2^2$$

(transposed, squared root and inverse)

Mahalanobis becomes Euclidean distance by premultiplying

$$\delta^* = \underset{\delta}{\arg\min} \left\{ \sum_{i=1}^{M} \| \underbrace{\Sigma_i^{-T/2}}_{W_i} (G_i^{i-1} \delta x_{i-1} - I \delta x_i) - (\Sigma_i^{-T/2} a_i) \|_2^2 + \right.$$

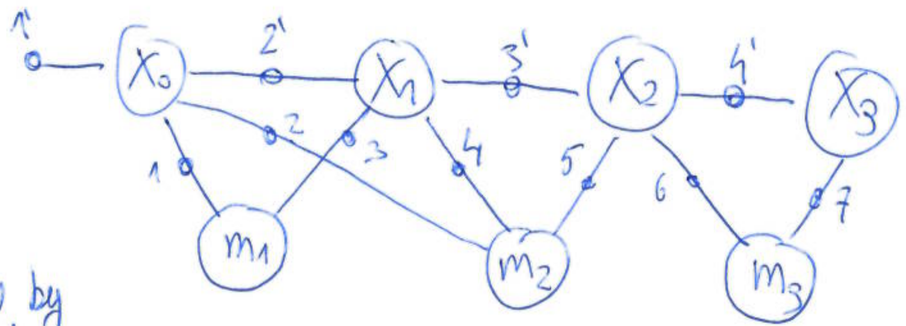
$$\left. + \sum_{k} \| \underbrace{\Sigma_k^{-T/2}}_{W_k} (H_k^{i_k} \delta x_{i_k} + J_k^{j_k} \delta m_{j_k}) - (\Sigma_k^{-T/2} c_k) \|_2^2 \right\}$$

$$\boxed{\delta^* = \underset{\delta}{\arg\min} \| A \delta - b \|_2^2}$$

Sumatory as a matrix product.

LLSQ

Ex 3



Premultiply row by information factor W.

$$
A = \begin{array}{c} \\ (W_1' x) \\ (W_2' x) \\ (W_3') \\ (W_4') \\ \vdots \\ \\ \\ W_3 \\ W_4 \\ W_5 \\ W_6 \\ W_7 \end{array}
\begin{array}{c}
\begin{array}{ccccccc} (x_0) & (x_1) & (x_2) & (x_3) & (m_1) & (m_2) & (m_3) \end{array} \\
\left[\begin{array}{ccccccc}
-I & & & & & & \\
G_1^0 & -I & & & & & \\
 & G_2^1 & -I & & & & \\
 & & G_3^2 & -I & & & \\
H_1^0 & & & & J_1^1 & & \\
H_2^0 & & & & & J_2^2 & \\
 & H_3^1 & & & J_3^1 & & \\
 & H_4^1 & & & & J_4^2 & \\
 & & H_5^2 & & & J_5^2 & \\
 & & H_6^2 & & & & J_6^3 \\
 & & & H_7^3 & & & J_7^3
\end{array}\right]
\end{array}
$$

Odometry factors

Observation factors

* Solve the LLSQ

$$\min_{\delta} \| A\delta - b \|_2^2$$

$$\frac{\partial}{\partial \delta} \;\; \frac{1}{2}(A\delta - b)\cdot A = 0 \quad \Longleftrightarrow \quad A\delta = b$$

$$\begin{array}{cc} & N \\ M+n & \boxed{\;A\;} \end{array} \; \boxed{\delta} = \boxed{\;} \; b$$

A is not square. It is an overconstrained problem.

$$A\delta = b$$
$$A^T A \delta = A^T b$$
$$\delta = (A^T A)^{-1} A^T b$$

Pseudo-inverse (next lecture more on this)

Algorithm raw SAM:

- calculate A, b, around $\{X^\circ, M^\circ\} = \theta^\circ$
- $\delta^* = \arg\min \| A\delta - b \|_2^2$
- Update X°, M°, $\theta^\circ := \theta^\circ + \delta^*$
- if convergence : return θ