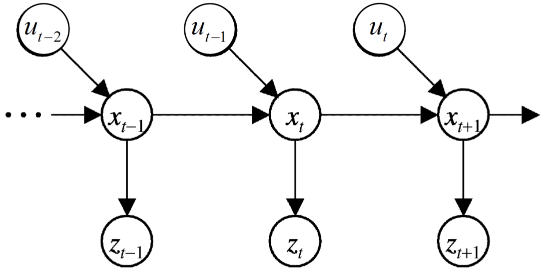
# ECE276A: Sensing & Estimation in Robotics
# Lecture 12: Graph SLAM

Nikolay Atanasov

natanasov@ucsd.edu

UC San Diego
JACOBS SCHOOL OF ENGINEERING
Electrical and Computer Engineering

# Markov Assumptions




- **Motion model**: given $\mathbf{x}_t$, $\mathbf{u}_t$, the state $\mathbf{x}_{t+1}$ is independent of the history $\mathbf{x}_{0:t-1}$, $\mathbf{z}_{0:t-1}$, $\mathbf{u}_{0:t-1}$:

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t, \mathbf{w}_t) \sim p_f(\cdot \mid \mathbf{x}_t, \mathbf{u}_t)$$

- **Observation model**: given $\mathbf{x}_t$, the observation $\mathbf{z}_t$ is independent of the history $\mathbf{x}_{0:t-1}$, $\mathbf{z}_{0:t-1}$, $\mathbf{u}_{0:t-1}$:

$$\mathbf{z}_t = h(\mathbf{x}_t, \mathbf{v}_t) \sim p_h(\cdot \mid \mathbf{x}_t)$$

# Joint Distribution Factorization

- The Markov assumptions induce a factorization of the joint probability density function of the states $\mathbf{x}_{0:T}$, observations $\mathbf{z}_{0:T}$, and inputs $\mathbf{u}_{0:T-1}$:

$$
\begin{aligned}
&p(\mathbf{x}_{0:T}, \mathbf{z}_{0:T}, \mathbf{u}_{0:T-1}) \\
&\overset{\text{Conditional}}{\underset{\text{probability}}{=\!=\!=}} p(\mathbf{z}_T | \mathbf{x}_{0:T}, \mathbf{z}_{0:T-1}, \mathbf{u}_{0:T-1}) p(\mathbf{x}_{0:T}, \mathbf{z}_{0:T-1}, \mathbf{u}_{0:T-1}) \\
&\overset{\text{Markov}}{\underset{\text{assumption}}{=\!=\!=}} \underbrace{p_h(\mathbf{z}_T | \mathbf{x}_T)}_{\text{observation model}} \; p(\mathbf{x}_{0:T}, \mathbf{z}_{0:T-1}, \mathbf{u}_{0:T-1}) \\
&\overset{\text{Conditional}}{\underset{\text{probability}}{=\!=\!=}} p_h(\mathbf{z}_T | \mathbf{x}_T) p(\mathbf{x}_T | \mathbf{x}_{0:T-1}, \mathbf{z}_{0:T-1}, \mathbf{u}_{0:T-1}) p(\mathbf{x}_{0:T-1}, \mathbf{z}_{0:T-1}, \mathbf{u}_{0:T-1}) \\
&\overset{\text{Markov}}{\underset{\text{assumption}}{=\!=\!=}} p_h(\mathbf{z}_T | \mathbf{x}_T) \underbrace{p_f(\mathbf{x}_T | \mathbf{x}_{T-1}, \mathbf{u}_{T-1})}_{\text{motion model}} \underbrace{p(\mathbf{u}_{T-1} | \mathbf{x}_{T-1})}_{\text{control policy}} p(\mathbf{x}_{0:T-1}, \mathbf{z}_{0:T-1}, \mathbf{u}_{0:T-2}) \\
&= \cdots \\
&= \underbrace{p(\mathbf{x}_0)}_{\text{prior}} \prod_{t=0}^{T} \underbrace{p_h(\mathbf{z}_t \mid \mathbf{x}_t)}_{\text{observation model}} \prod_{t=0}^{T-1} \underbrace{p_f(\mathbf{x}_{t+1} \mid \mathbf{x}_t, \mathbf{u}_t)}_{\text{motion model}} \prod_{t=0}^{T-1} \underbrace{p(\mathbf{u}_t \mid \mathbf{x}_t)}_{\text{control policy}}
\end{aligned}
$$

# Parameter Estimation

- Consider data $D$ generated by probabilistic model $p(D|\boldsymbol{\theta})$ with parameters $\boldsymbol{\theta}$
- Parameter estimation strategies:
  - **Maximum Likelihood Estimation (MLE)**: maximize the likelihood of the data $D$ given the parameters $\boldsymbol{\theta}$:

$$\boldsymbol{\theta}_* \in \arg\max_{\boldsymbol{\theta}} p(D|\boldsymbol{\theta})$$

  - **Maximum A Posteriori (MAP)**: maximize the likelihood of the parameters $\boldsymbol{\theta}$ given the data $D$:

$$\boldsymbol{\theta}_* \in \arg\max_{\boldsymbol{\theta}} p(\boldsymbol{\theta}|D) = \arg\max_{\boldsymbol{\theta}} p(D|\boldsymbol{\theta})p(\boldsymbol{\theta}) = \arg\max_{\boldsymbol{\theta}} p(D,\boldsymbol{\theta})$$

- SLAM as a MAP problem:
  - data: $D = \{\mathbf{z}_{0:T}, \mathbf{u}_{0:T-1}\}$
  - parameters: $\boldsymbol{\theta} = \mathbf{x}_{0:T}$
  - joint pdf: $p(D,\boldsymbol{\theta}) = p(\mathbf{x}_0)\prod_{t=0}^{T} p_h(\mathbf{z}_t \mid \mathbf{x}_t)\prod_{t=0}^{T-1} p_f(\mathbf{x}_{t+1} \mid \mathbf{x}_t, \mathbf{u}_t)\prod_{t=0}^{T-1} p(\mathbf{u}_t \mid \mathbf{x}_t)$

# MAP Formulation of SLAM

- SLAM as a MAP problem (usually $p(\mathbf{u}_t|\mathbf{x}_t)$ is not considered):

$$\max_{\mathbf{x}_{0:T}} \log p(\mathbf{x}_0) + \sum_{t=0}^{T} \log p_h(\mathbf{z}_t|\mathbf{x}_t) + \sum_{t=0}^{T-1} \log p_f(\mathbf{x}_{t+1} \mid \mathbf{x}_t, \mathbf{u}_t)$$

- Start with initial guess $\hat{\mathbf{x}}_{0:T}$, e.g., from odometry
- Linearize the motion model $f(\mathbf{x}, \mathbf{u}, \mathbf{w})$ and observation model $h(\mathbf{x}, \mathbf{v})$
- Solve the linearized MAP problem to obtain a descent direction $\tilde{\mathbf{x}}_{0:T}$
- Update the guess $\hat{\mathbf{x}}'_{0:T} = \hat{\mathbf{x}}_{0:T} + \tilde{\mathbf{x}}_{0:T}$
- Perform descent by re-linearizing around $\hat{\mathbf{x}}'_{0:T}$ and obtaining a new descent direction $\tilde{\mathbf{x}}'_{0:T}$

# Motion Model Linearization

- Motion model linearization around state $\hat{\mathbf{x}}_t$ and noise 0:

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t, \mathbf{w}_t) \approx f(\hat{\mathbf{x}}_t, \mathbf{u}_t, 0) + F_t(\mathbf{x}_t - \hat{\mathbf{x}}_t) + Q_t \mathbf{w}_t$$

- Motion model Jacobians:

$$F_t = \frac{df}{d\mathbf{x}}(\hat{\mathbf{x}}_t, \mathbf{u}_t, 0) \qquad Q_t = \frac{df}{d\mathbf{w}}(\hat{\mathbf{x}}_t, \mathbf{u}_t, 0)$$

- Let $\tilde{\mathbf{x}}_t := \mathbf{x}_t - \hat{\mathbf{x}}_t$ and $\boldsymbol{\eta}_{t+1} := \hat{\mathbf{x}}_{t+1} - f(\hat{\mathbf{x}}_t, \mathbf{u}_t, 0)$:

$$\begin{aligned} \tilde{\mathbf{x}}_{t+1} + \hat{\mathbf{x}}_{t+1} &\approx f(\hat{\mathbf{x}}_t, \mathbf{u}_t, 0) + F_t \tilde{\mathbf{x}}_t + Q_t \mathbf{w}_t \\ \tilde{\mathbf{x}}_{t+1} + \boldsymbol{\eta}_{t+1} &\approx F_t \tilde{\mathbf{x}}_t + Q_t \mathbf{w}_t \end{aligned}$$

- Motion model pdf with $\mathbf{w}_t \sim \mathcal{N}(0, W)$ and $W_t := Q_t W Q_t^\top$:

$$p_f(\mathbf{x}_{t+1} | \mathbf{x}_t, \mathbf{u}_t) \approx$$

$$\frac{1}{\sqrt{(2\pi)^{d_x} \det(W_t)}} \exp\left(-\frac{1}{2}\left(\tilde{\mathbf{x}}_{t+1} + \boldsymbol{\eta}_{t+1} - F_t \tilde{\mathbf{x}}_t\right)^\top W_t^{-1} \left(\tilde{\mathbf{x}}_{t+1} + \boldsymbol{\eta}_{t+1} - F_t \tilde{\mathbf{x}}_t\right)\right)$$

$$\log p_f(\mathbf{x}_{t+1} | \mathbf{x}_t, \mathbf{u}_t) \approx$$

$$-\frac{1}{2}\log((2\pi)^{d_x} \det(W_t)) - \frac{1}{2}\left(\tilde{\mathbf{x}}_{t+1} + \boldsymbol{\eta}_{t+1} - F_t \tilde{\mathbf{x}}_t\right)^\top W_t^{-1} \left(\tilde{\mathbf{x}}_{t+1} + \boldsymbol{\eta}_{t+1} - F_t \tilde{\mathbf{x}}_t\right)$$

# Observation Model Linearization

- Observation model linearization around state $\hat{\mathbf{x}}_t$ and noise 0:

$$\mathbf{z}_t = h(\mathbf{x}_t, \mathbf{v}_t) \approx h(\hat{\mathbf{x}}_t, 0) + H_t(\mathbf{x}_t - \hat{\mathbf{x}}_t) + R_t \mathbf{v}_t$$

- Observation model Jacobians:

$$H_t = \frac{dh}{d\mathbf{x}}(\hat{\mathbf{x}}_t, 0) \qquad\qquad R_t = \frac{dh}{d\mathbf{v}}(\hat{\mathbf{x}}_t, 0)$$

- Let $\tilde{\mathbf{x}}_t := \mathbf{x}_t - \hat{\mathbf{x}}_t$ and $\tilde{\mathbf{z}}_t := \mathbf{z}_t - h(\hat{\mathbf{x}}_t, 0)$:

$$\tilde{\mathbf{z}}_t = H_t \tilde{\mathbf{x}}_t + R_t \mathbf{v}_t$$

- Observation model pdf with $\mathbf{v}_t \sim \mathcal{N}(0, V)$ and $V_t := R_t V R_t^\top$:

$$p_h(\mathbf{z}_t | \mathbf{x}_t) \approx \frac{1}{\sqrt{(2\pi)^{d_z} \det(V_t)}} \exp\left(-\frac{1}{2} \left(\tilde{\mathbf{z}}_t - H_t \tilde{\mathbf{x}}_t\right)^\top V_t^{-1} \left(\tilde{\mathbf{z}}_t - H_t \tilde{\mathbf{x}}_t\right)\right)$$

$$\log p_h(\mathbf{z}_t | \mathbf{x}_t) \approx -\frac{1}{2} \log((2\pi)^{d_z} \det(V_t)) - \frac{1}{2} \left(\tilde{\mathbf{z}}_t - H_t \tilde{\mathbf{x}}_t\right)^\top V_t^{-1} \left(\tilde{\mathbf{z}}_t - H_t \tilde{\mathbf{x}}_t\right)$$

# Descent Direction from Linearized MAP Problem

- Linearized MAP problem is a least-squares problem:

$$\min_{\tilde{\mathbf{x}}_{0:T}} \left\{ \|\Sigma_0^{-1/2}\tilde{\mathbf{x}}_0\|_2^2 + \sum_{t=0}^{T} \|V_t^{-1/2}\left(\tilde{\mathbf{z}}_t - H_t\tilde{\mathbf{x}}_t\right)\|_2^2 + \sum_{t=0}^{T-1} \|W_t^{-1/2}\left(\tilde{\mathbf{x}}_{t+1} + \boldsymbol{\eta}_{t+1} - F_t\tilde{\mathbf{x}}_t\right)\|_2^2 \right\}$$

- Using that $\left\|\begin{pmatrix}\mathbf{x}_1\\ \mathbf{x}_2\end{pmatrix} - \begin{pmatrix}\mathbf{y}_1\\ \mathbf{y}_2\end{pmatrix}\right\|_2^2 = \|\mathbf{x}_1 - \mathbf{y}_1\|_2^2 + \|\mathbf{x}_2 - \mathbf{y}_2\|_2^2$ for $\mathbf{x}_1, \mathbf{y}_1 \in \mathbb{R}^{d_1}$, $\mathbf{x}_2, \mathbf{y}_2 \in \mathbb{R}^{d_2}$, rewrite the least-squares cost in matrix notation:

$$\|\Sigma_0^{-1/2}\tilde{\mathbf{x}}_0\|_2^2 + \sum_{t=0}^{T} \|V_t^{-1/2}\left(\tilde{\mathbf{z}}_t - H_t\tilde{\mathbf{x}}_t\right)\|_2^2 + \sum_{t=0}^{T-1} \|W_t^{-1/2}\left(\tilde{\mathbf{x}}_{t+1} + \boldsymbol{\eta}_{t+1} - F_t\tilde{\mathbf{x}}_t\right)\|_2^2$$

$$= \|\Sigma_0^{-1/2}\tilde{\mathbf{x}}_0\|^2 + \left\|\begin{bmatrix} V_0^{-1/2}\left(\tilde{\mathbf{z}}_0 - H_0\tilde{\mathbf{x}}_0\right) \\ \vdots \\ V_T^{-1/2}\left(\tilde{\mathbf{z}}_T - H_T\tilde{\mathbf{x}}_T\right)\end{bmatrix}\right\|_2^2 + \left\|\begin{bmatrix} W_0^{-1/2}\left(\boldsymbol{\eta}_1 + \tilde{\mathbf{x}}_1 - F_0\tilde{\mathbf{x}}_0\right) \\ \vdots \\ W_{T-1}^{-1/2}\left(\boldsymbol{\eta}_T + \tilde{\mathbf{x}}_T - F_{T-1}\tilde{\mathbf{x}}_{T-1}\right)\end{bmatrix}\right\|_2^2$$

# Descent Direction from Linearized MAP Problem

- Using that $\left\| \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{pmatrix} - \begin{pmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{pmatrix} \right\|_2^2 = \|\mathbf{x}_1 - \mathbf{y}_1\|_2^2 + \|\mathbf{x}_2 - \mathbf{y}_2\|_2^2$ for $\mathbf{x}_1, \mathbf{y}_1 \in \mathbb{R}^{d_1}$, $\mathbf{x}_2, \mathbf{y}_2 \in \mathbb{R}^{d_2}$, rewrite the least-squares cost in matrix notation:

$$
\begin{aligned}
&\|\Sigma_0^{-1/2}\tilde{\mathbf{x}}_0\|_2^2 + \sum_{t=0}^{T} \|V_t^{-1/2}\left(\tilde{\mathbf{z}}_t - H_t\tilde{\mathbf{x}}_t\right)\|_2^2 + \sum_{t=0}^{T-1} \|W_t^{-1/2}\left(\tilde{\mathbf{x}}_{t+1} + \boldsymbol{\eta}_{t+1} - F_t\tilde{\mathbf{x}}_t\right)\|_2^2 \\
&= \|\Sigma_0^{-1/2}\tilde{\mathbf{x}}_0\|^2 + \left\| \begin{bmatrix} V_0^{-1/2}H_0 & & \\ & \ddots & \\ & & V_T^{-1/2}H_T \end{bmatrix} \begin{pmatrix} \tilde{\mathbf{x}}_0 \\ \vdots \\ \tilde{\mathbf{x}}_T \end{pmatrix} - \begin{bmatrix} V_0^{-1/2}\tilde{\mathbf{z}}_0 \\ \vdots \\ V_T^{-1/2}\tilde{\mathbf{z}}_T \end{bmatrix} \right\|_2^2 \\
&+ \left\| \begin{bmatrix} W_0^{-1/2}F_0 & -W_0^{-1/2} & & \\ & W_1^{-1/2}F_1 & \ddots & \\ & & \ddots & -W_{T-1}^{-1/2} \\ & & & W_{T-1}^{-1/2}F_{T-1} \end{bmatrix} \begin{pmatrix} \tilde{\mathbf{x}}_0 \\ \vdots \\ \tilde{\mathbf{x}}_T \end{pmatrix} - \begin{bmatrix} W_0^{-1/2}\eta_1 \\ \vdots \\ W_{T-1}^{-1/2}\eta_T \end{bmatrix} \right\|_2^2
\end{aligned}
$$

# Descent Direction from Linearized MAP Problem

$$\left\| \underbrace{\begin{bmatrix} \Sigma_{0|0}^{-1/2} & & & \\ V_0^{-1/2} H_0 & & & \\ & \ddots & & \\ & & \ddots & \\ & & & V_T^{-1/2} H_T \\ W_0^{-1/2} F_0 & -W_0^{-1/2} & & \\ & W_1^{-1/2} F_1 & \ddots & \\ & & \ddots & -W_{T-1}^{-1/2} \\ & & & W_{T-1}^{-1/2} F_{T-1} \end{bmatrix}}_{J} \begin{pmatrix} \tilde{\mathbf{x}}_0 \\ \vdots \\ \tilde{\mathbf{x}}_T \end{pmatrix} - \underbrace{\begin{bmatrix} 0 \\ V_0^{-1/2} \tilde{\mathbf{z}}_0 \\ \vdots \\ \vdots \\ V_T^{-1/2} \tilde{\mathbf{z}}_T \\ W_0^{-1/2} \boldsymbol{\eta}_1 \\ W_1^{-1/2} \boldsymbol{\eta}_2 \\ \vdots \\ W_{T-1}^{-1/2} \boldsymbol{\eta}_T \end{bmatrix}}_{\mathbf{b}} \right\|_2^2$$

$$= \|J \tilde{\mathbf{x}}_{0:T} - \mathbf{b}\|_2^2$$

# Descent Direction from Linearized MAP Problem

- The linearized MAP problem to obtain a descent direction $\tilde{\mathbf{x}}_{0:T}$ leads to a least-squares problem:

$$\min_{\tilde{\mathbf{x}}_{0:T}} \|J\tilde{\mathbf{x}}_{0:T} - \mathbf{b}\|_2^2$$

- Setting the gradient to zero leads to the **normal equations**:

$$J^\top J \tilde{\mathbf{x}}_{0:T} = J^\top \mathbf{b}$$

- The matrix of model Jacobians $J$ is **sparse**
- $J^\top J$ is the **info matrix** of the Gaussian distribution of $\tilde{\mathbf{x}}_{0:T} \mid \mathbf{z}_{0:T}, \mathbf{u}_{0:T-1}$
- The normal equations can be solved via:
  - Cholesky factorization of $J^\top J$
  - QR factorization of $J$
  - QR factorization is a more efficient and robust way to solve the normal equations because it avoids computing $J^\top J$, which is slow and squares the condition number of $J$

# Descent Direction from QR Factorization

- Number of variables: $n$
- Number of measurement constraints: $m$
- QR factorization: $J = Q \begin{bmatrix} R \\ 0 \end{bmatrix} \in \mathbb{R}^{m \times n}$
- $R \in \mathbb{R}^{n \times n}$ is the **upper-triangular square root information matrix**

$$R^\top R = J^\top J$$

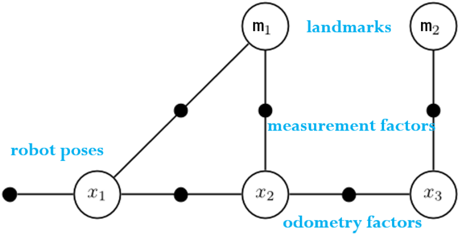
- $Q \in \mathbb{R}^{m \times m}$ is an orthogonal matrix: $Q^\top Q = I$
- Descent direction via QR factorization:

$$\begin{aligned}\|J\tilde{\mathbf{x}}_{0:T} - \mathbf{b}\|_2^2 &= \left\| Q \begin{bmatrix} R \\ 0 \end{bmatrix} \tilde{\mathbf{x}}_{0:T} - \mathbf{b} \right\|_2^2 = \left\| Q^\top Q \begin{bmatrix} R \\ 0 \end{bmatrix} \tilde{\mathbf{x}}_{0:T} - Q^\top \mathbf{b} \right\|_2^2 \\ &= \left\| \begin{bmatrix} R \\ 0 \end{bmatrix} \tilde{\mathbf{x}}_{0:T} - \begin{bmatrix} \mathbf{b}'_1 \\ \mathbf{b}'_2 \end{bmatrix} \right\|_2^2 = \|R\tilde{\mathbf{x}}_{0:T} - \mathbf{b}'_1\|_2^2 + \underbrace{\|\mathbf{b}'_2\|_2^2}_{\text{residual}}\end{aligned}$$

- Since $R$ is upper-triangular, back-substitution can be used to compute $\tilde{\mathbf{x}}_{0:T}$

# Factor Graph

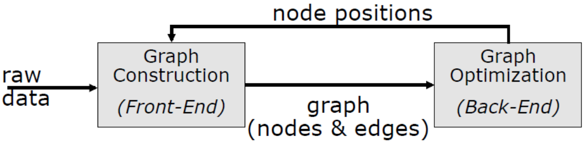
- **Factor graph**: bipartite graph describing data (observations $\mathbf{z}_t$, inputs $\mathbf{u}_t$) and variables (states $\mathbf{x}_t$, landmarks $\mathbf{m}_j$) in a SLAM problem




- **Nodes**: variables to be estimated: robot states $\mathbf{x}_t$ and landmark states $\mathbf{m}_j$
- **Factors**: relate two variables by input $\mathbf{u}_t$ or observation $\mathbf{z}_t$ data and associated motion or observation model:
  - Motion factor: $\log p_f(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{u}_t)$
  - Observation factor: $\log p_h(\mathbf{z}_t|\mathbf{x}_t, \mathbf{m}_j)$

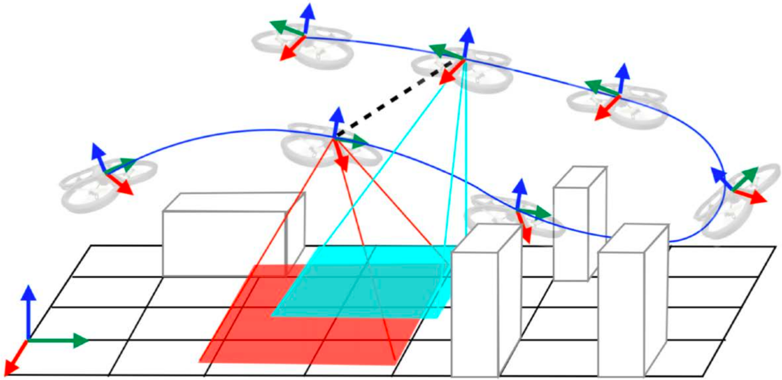
# Factor Graph SLAM

- **Front-end**: construction of factor graph using odometry, laser-scan matching, feature matching, etc.
- **Back-end**: graph optimization to estimate the variables $(\mathbf{x}_{0:T}, \{\mathbf{m}_j\})$




- The factor graph formulation of SLAM with Gaussian noise leads to a nonlinear least-squares problem
- Given an initial estimate of the robot trajectory and landmark poses (e.g., from odometry and triangulation of 2-D image features), we use the Gauss-Newton algorithm to solve the nonlinear least-squares problem
- Assuming a Gaussian distribution for the constraints is not always the best choice in the presence of outliers. A heavy-tailed distribution can be used for outlier rejection.
- Loop closure: observing previously seen landmarks generates graph factors between non-successive robot poses

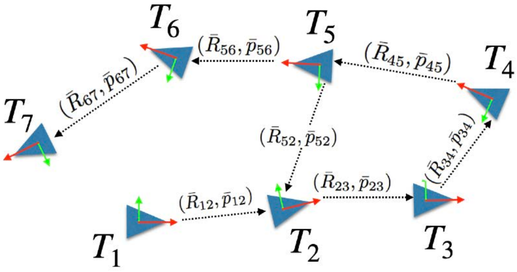
# Pose Graph



- **Variables**: robot poses $T_i$
- **Measurements**: relative poses from odometry and loop closures: $\bar{T}_{ij}$

# Pose Graph Optimization

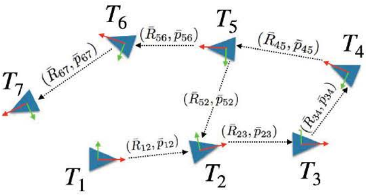
- **Pose graph**



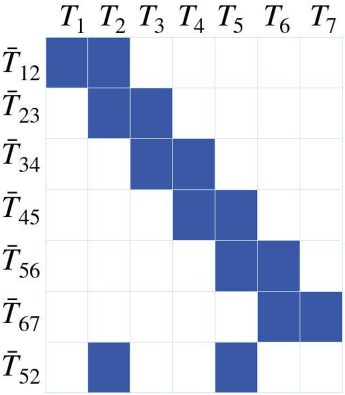

- **Pose graph optimization**:

$$\min_{T_1,\ldots,T_n} \sum_{(i,j)\in\mathcal{E}} \| W_{ij}^{-1/2} \log( \bar{T}_{ij}^{-1} T_i^{-1} T_j)^\vee \|_2^2$$

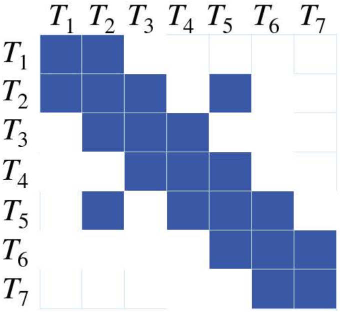
# Pose Graph Optimization: Sparsity




Jacobian **J**

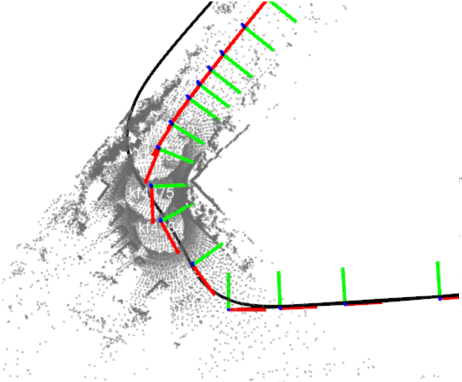



Hessian $\mathbf{J}^T \mathbf{J}$

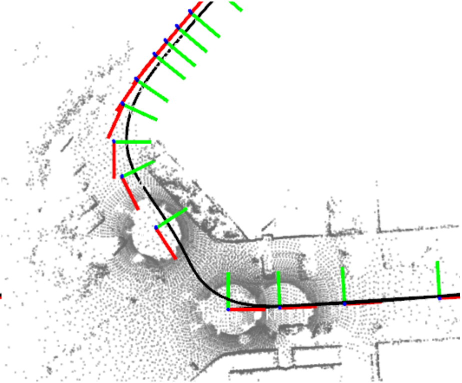



a.k.a. Information Matrix of the estimate

# Pose Graph Optimization: Example



(a) Before PGO

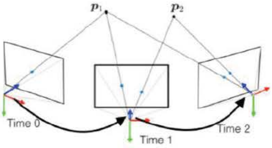


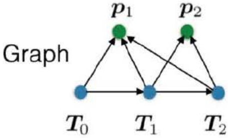
(b) After PGO

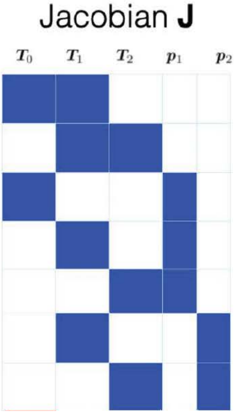
https://www.youtube.com/watch?v=KYvOqUB_odg

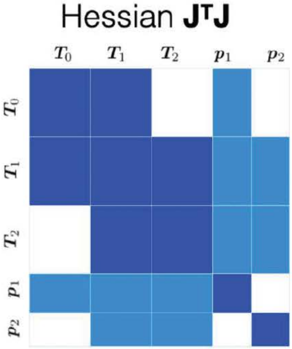
# Landmark-based SLAM

$$\min_{\{T_t\},\{\mathbf{m}_j\}} \sum_t \| W^{-1/2} \log(\bar{T}_{t,t+1}^{-1} T_t^{-1} T_{t+1})^\vee \|_2^2 + \sum_j \sum_{t \in \mathcal{V}_j} \| V^{-1/2} (\mathbf{z}_{t,j} - h(T_t, \mathbf{m}_j)) \|_2^2$$

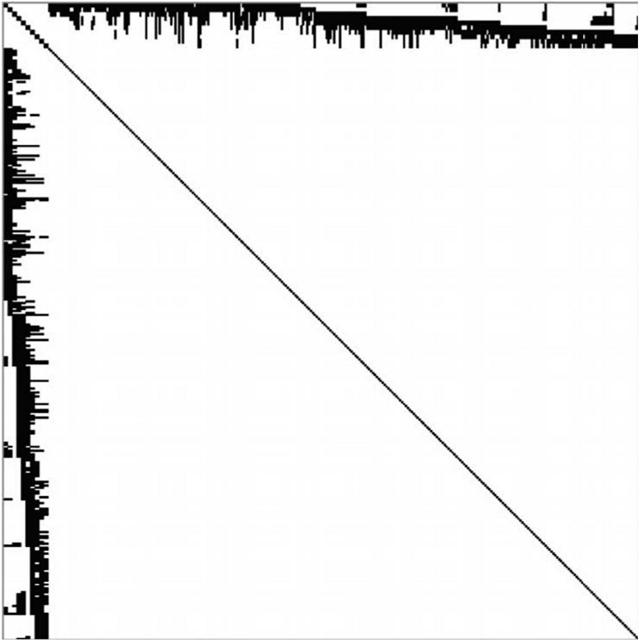


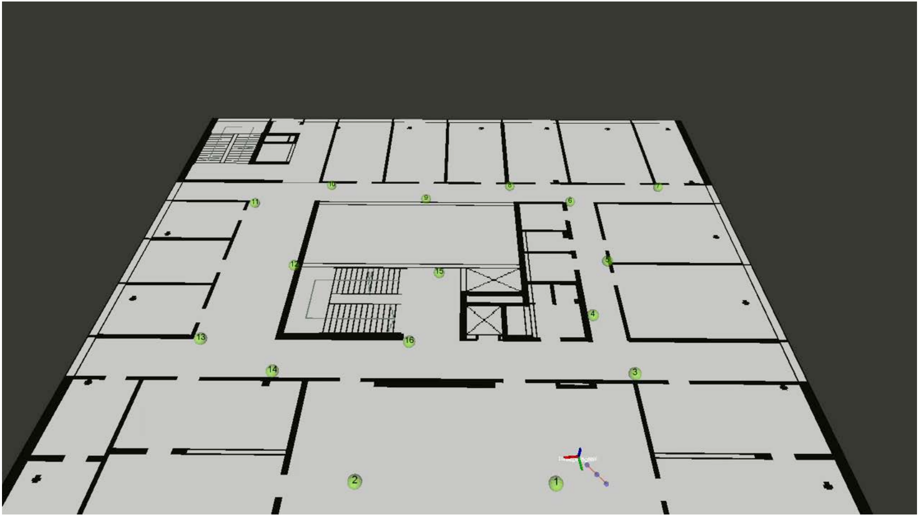



# Landmark-based SLAM Hessian Sparsity



# Landmark-based SLAM: Example



https://www.youtube.com/watch?v=OdJ042prg_M

# Landmark-based SLAM: Variable Marginalization

- What if we only need a subset of the variables?
- Normal equations: $J^\top J\tilde{\mathbf{x}} = J^\top \mathbf{b}$
- Information matrix blocks:

$$J^\top J\tilde{\mathbf{x}} = \begin{bmatrix} \Omega_{aa} & \Omega_{ab} \\ \Omega_{ab}^\top & \Omega_{bb} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{x}}_a \\ \tilde{\mathbf{x}}_b \end{bmatrix} = \begin{bmatrix} \mathbf{c}_a \\ \mathbf{c}_b \end{bmatrix}$$

- Pre-multiply by $\begin{bmatrix} I & -\Omega_{ab}\Omega_{bb}^{-1} \\ 0 & I \end{bmatrix}$ and subtract second from first equation:

$$\begin{bmatrix} \Omega_{aa} - \Omega_{ab}\Omega_{bb}^{-1}\Omega_{ab}^\top & 0 \\ \Omega_{ab}^\top & \Omega_{bb} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{x}}_a \\ \tilde{\mathbf{x}}_b \end{bmatrix} = \begin{bmatrix} \mathbf{c}_a - \Omega_{ab}\Omega_{bb}^{-1}\mathbf{c}_b \\ \mathbf{c}_b \end{bmatrix}$$

- We can obtain $\tilde{\mathbf{x}}_a$ by solving the smaller system determined by the Schur complement of $\Omega_{bb}$:

$$(\Omega_{aa} - \Omega_{ab}\Omega_{bb}^{-1}\Omega_{ab}^\top)\tilde{\mathbf{x}}_a = \mathbf{c}_a - \Omega_{ab}\Omega_{bb}^{-1}\mathbf{c}_b$$

# Landmark-based SLAM: Variable Marginalization

- Probabilistic perspective of Schur complement:

$$\begin{bmatrix} \tilde{\mathbf{x}}_a \\ \tilde{\mathbf{x}}_b \end{bmatrix} \sim \mathcal{N}\left( \begin{bmatrix} \Omega_{aa} & \Omega_{ab} \\ \Omega_{ab}^\top & \Omega_{bb} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{c}_a \\ \mathbf{c}_b \end{bmatrix}, \begin{bmatrix} \Omega_{aa} & \Omega_{ab} \\ \Omega_{ab}^\top & \Omega_{bb} \end{bmatrix}^{-1} \right)$$

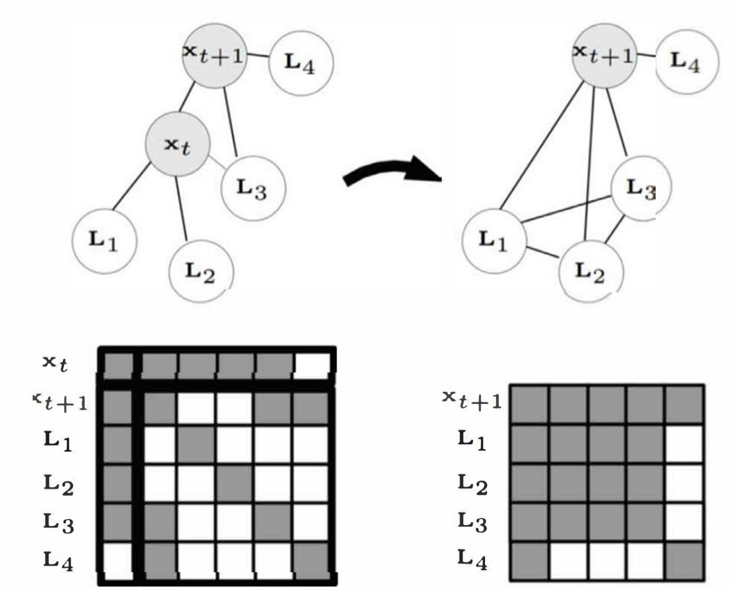
- Marginal of $\tilde{\mathbf{x}}_a$:

$$\begin{aligned} p(\tilde{\mathbf{x}}_a) &= \int p(\tilde{\mathbf{x}}_a, \tilde{\mathbf{x}}_b) d\tilde{\mathbf{x}}_b \\ &= \phi\left(\tilde{\mathbf{x}}_a; (\Omega_{aa} - \Omega_{ab}\Omega_{bb}^{-1}\Omega_{ab}^\top)^{-1}(\mathbf{c}_a - \Omega_{ab}\Omega_{bb}^{-1}\mathbf{c}_b), (\Omega_{aa} - \Omega_{ab}\Omega_{bb}^{-1}\Omega_{ab}^\top)^{-1}\right) \end{aligned}$$

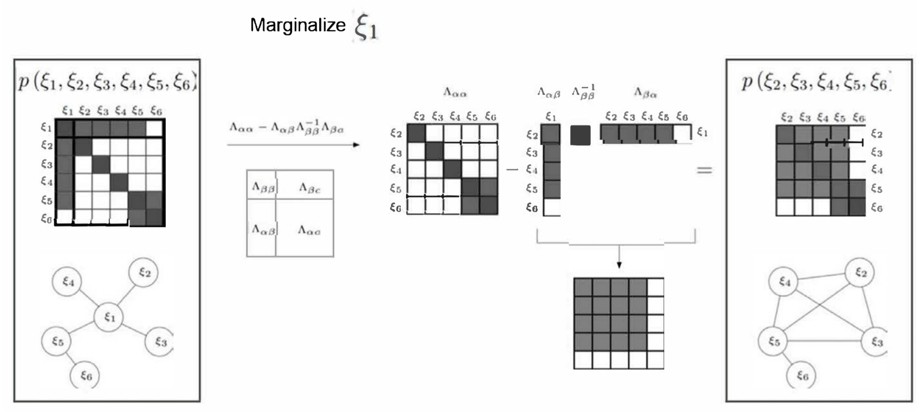
- Marginalizing a variable creates non-zero off-diagonals (called **fill-in**) in the information matrix for all variables that had a non-zero off-diagonal element with the marginalized variable $\Rightarrow$ **loss of sparsity**

- In graph terms, variable elimination creates a clique between the neighbors of the eliminated node

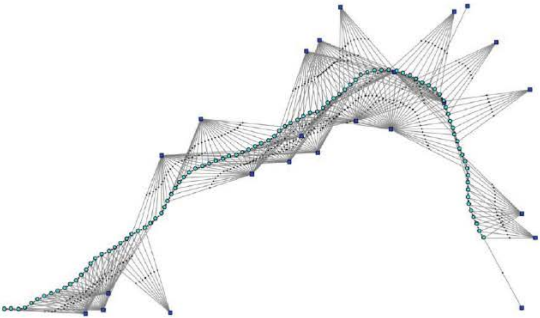
# Landmark-based SLAM: Variable Marginalization



# Landmark-based SLAM: Variable Marginalization



# Smoothing vs Filtering



- **Smoothing**: equivalent to MAP optimization
  - **many variables**: estimates entire robot trajectory and map
  - **sparse** info matrix $J^\top J$
- **Fixed-lag smoothing**:
  - **fewer variables**: estimate only variables in a time window
  - **denser** info matrix after Schur complement to marginalize old variables
- **Filtering**:
  - **fewest variables**: estimate only current pose and landmarks
  - **densest** info matrix after Schur complement to marginalize all old variables