ECE276A: Sensing & Estimation in Robotics
Lecture 16: Batch Estimation, Factor Graphs

Instructor:

Nikolay Atanasov: natanasov@ucsd.edu

Teaching Assistants:

Qiaojun Feng: qif007@eng.ucsd.edu

Tianyu Wang: tiw161@eng.ucsd.edu

Ibrahim Akbar: iakbar@eng.ucsd.edu

You-Yi Jau: yjau@eng.ucsd.edu

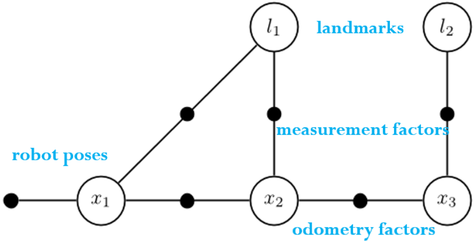
Harshini Rajachander: hrajacha@eng.ucsd.edu

UC San Diego

JACOBS SCHOOL OF ENGINEERING

Electrical and Computer Engineering

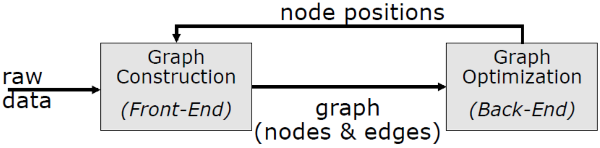
Factor Graph



- A graphical model capturing the first-order Markov assumptions

- **Front-end**: constructs the graph using laser-scan-matching or feature matching
  1. **Nodes**: variables to be estimated: robot poses and landmark positions
  2. **Edges** (called **factors**): have associated measurement error functions and information matrices, defining a Mahalonobis norm on the error
     - **Odometry**: $e_{ij}(x_i, x_j) = \log\left(z_{ij}^{-1} x_i^{-1} x_j\right)^{\vee}$ and $\Omega_{ij} = W^{-1}$
     - **Camera**: $e_{ij}(x_i, x_j) = z_{ij} - h(x_i, x_j)$ and $\Omega_{ij} = V^{-1}$

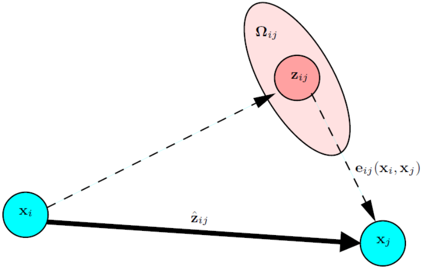
- **Back-end**: performs inference over the graph



Inference over Factor Graphs

- Inference over the graph: a nonlinear least-squares problem:

$$\arg\max_{x} \sum_{(i,j)\in E} \underbrace{e_{ij}(x)^T \Omega_{ij} e_{ij}(x)}_{F_{ij}(x)}$$



- Linearization of the factors $F_{ij}(x)$ leads to a **sparse linear system**
- Assumptions:
  - A "good" initial guess is available
  - The error functions are smooth in the neighborhood of the minima
- Iterative linearization:
  1. linearize the error functions $ildex_{ij}(x)$ around the current guess
  2. compute the gradient of the quadratic objective $\sum_{(i,j)\in E} F_{ij}(x)$, set it equal to zero, and solve the resulting linear system
  3. update the current guess and repeat
- The linearization points can be corrected iteratively via the **Gauss-Newton** or **Levenberg-Marquardt** algorithms

Extended Kalman Smoothing via Least Squares

- **Linearization trajectory**: initial estimate $\hat{x}_{0:T}$, e.g., from odometry
- **Motion model linearization**:

$$x_{t+1} = f(x_t, u_t, w_t) \approx f(\hat{x}_t, u_t, 0) + F_t(x_t - \hat{x}_t) + Q_t w_t \qquad w_t \sim \mathcal{N}(0, W)$$

- **Observation model linearization**:

$$z_t = h(x_t, v_t) \approx h(\hat{x}_t, 0) + H_t(x_t - \hat{x}_t) + R_t v_t \qquad v_t \sim \mathcal{N}(0, V)$$

- **Jacobians**:

$$F_t := \frac{df}{dx}(\hat{x}_t, u_t, 0) \qquad Q_t := \frac{df}{dw}(\hat{x}_t, u_t, 0)$$
$$H_t := \frac{dh}{dx}(\hat{x}_t, 0) \qquad R_t := \frac{dh}{dv}(\hat{x}_t, 0)$$

- **Error model**: $\tilde{x}_t := x_t - \hat{x}_t$ and $\eta_{t+1} := \hat{x}_{t+1} - f(\hat{x}_t, u_t, 0)$ and $\tilde{z}_t := z_t - h(\hat{x}_t, 0)$

$$\begin{aligned} \tilde{x}_{t+1} + \eta_{t+1} &= F_t \tilde{x}_t + w'_t, & w'_t &\sim \mathcal{N}(0, \overbrace{Q_t W Q_t^T}^{W_t}) \\ \tilde{z}_{t+1} &= H_{t+1}\tilde{x}_{t+1} + v'_{t+1}, & v'_{t+1} &\sim \mathcal{N}(0, \underbrace{R_{t+1} V R_{t+1}^T}_{V_{t+1}}) \end{aligned}$$

Extended Kalman Smoothing via Least Squares

- **Joint distribution**:

$$p(x_{0:T}, z_{0:T}, u_{0:T-1}) = \underbrace{p_{0|0}(x_0)}_{\text{prior}} \prod_{t=1}^{T} \underbrace{p_h(z_t \mid x_t)}_{\text{observation model}} \prod_{t=1}^{T} \underbrace{p_f(x_t \mid x_{t-1}, u_{t-1})}_{\text{motion model}}$$

- SLAM via MAP leads to nonlinear least squares:

$$\arg\max_{x_{0:T}} \log p(x_{0:T}, z_{0:T}, u_{0:T-1}) \xlongequal[\text{initial guess } \hat{x}_{0:T}]{\text{linearize around}}$$

$$\approx \hat{x}_{0:T} + \arg\min_{\tilde{x}_{0:T}} \left\{ \|\tilde{x}_0\|^2_{\Sigma_{0|0}} + \sum_{t=0}^{T} \|\tilde{z}_t - H_t \tilde{x}_t\|^2_{V_t} + \sum_{t=1}^{T} \|\eta_t + \tilde{x}_t - F_t \tilde{x}_{t-1}\|^2_{W_t} \right\} \xlongequal[\|x\|_\Sigma := \sqrt{x^T \Sigma^{-1} x} = \|\Sigma^{-1/2} x\|_2]{\text{Mahalonobis Distance}}$$

$$= \hat{x}_{0:T} + \arg\min_{\tilde{x}_{0:T}} \left\{ \|\Sigma_{0|0}^{-1/2} \tilde{x}_0\|_2^2 + \sum_{t=0}^{T} \|V_t^{-1/2} (\tilde{z}_t - H_t \tilde{x}_t)\|_2^2 + \sum_{t=1}^{T} \|W_{t-1}^{-1/2} (\eta_t + \tilde{x}_t - F_t \tilde{x}_{t-1})\|_2^2 \right\}$$

- Solve the linear least squares problem to obtain $\tilde{x}_{0:T}$
- Update the linearization points: $\hat{x}'_{0:T} = \hat{x}_{0:T} + \tilde{x}_{0:T}$
- Repeat by linearizing around $\hat{x}'_{0:T}$

Sparse Least Squares

- $\left\| \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \right\|_2^2 = \|x_1 - y_1\|_2^2 + \|x_2 - y_2\|_2^2$ for $x_1, y_1 \in \mathbb{R}^{d_1}$, $x_2, y_2 \in \mathbb{R}^{d_2}$
- Using this we can write the least-squares problem in matrix notation:

$$\|\Sigma_{0|0}^{-1/2}\tilde{x}_0\|^2 + \sum_{t=0}^{T} \| V_t^{-1/2} (\tilde{z}_t - H_t \tilde{x}_t) \|^2 + \sum_{t=1}^{T} \| W_{t-1}^{-1/2} (\eta_t + \tilde{x}_t - F_{t-1}\tilde{x}_{t-1}) \|^2$$

$$= \|\Sigma_{0|0}^{-1/2}\tilde{x}_0\|^2 + \left\| \begin{bmatrix} V_0^{-1/2} (\tilde{z}_0 - H_0 \tilde{x}_0) \\ \vdots \\ V_T^{-1/2} (\tilde{z}_T - H_T \tilde{x}_T) \end{bmatrix} \right\|_2^2 + \left\| \begin{bmatrix} W_0^{-1/2} (\eta_1 + \tilde{x}_1 - F_0 \tilde{x}_0) \\ \vdots \\ W_{T-1}^{-1/2} (\eta_T + \tilde{x}_T - F_{T-1} \tilde{x}_{T-1}) \end{bmatrix} \right\|_2^2$$

$$= \|\Sigma_{0|0}^{-1/2}\tilde{x}_0\|^2 + \left\| \begin{bmatrix} V_0^{-1/2} H_0 & & \\ & \ddots & \\ & & V_T^{-1/2} H_T \end{bmatrix} \begin{pmatrix} \tilde{x}_0 \\ \vdots \\ \tilde{x}_T \end{pmatrix} - \begin{bmatrix} V_0^{-1/2} \tilde{z}_0 \\ \vdots \\ V_T^{-1/2} \tilde{z}_T \end{bmatrix} \right\|_2^2$$

$$+ \left\| \begin{bmatrix} W_0^{-1/2} F_0 & -W_0^{-1/2} & & \\ & W_1^{-1/2} F_1 & \ddots & \\ & & \ddots & -W_{T-1}^{-1/2} \\ & & & W_{T-1}^{-1/2} F_{T-1} \end{bmatrix} \begin{pmatrix} \tilde{x}_0 \\ \vdots \\ \tilde{x}_T \end{pmatrix} - \begin{bmatrix} W_0^{-1/2} \eta_1 \\ \vdots \\ W_{T-1}^{-1/2} \eta_T \end{bmatrix} \right\|_2^2$$

Sparse Least Squares

$$\left\| \underbrace{\begin{bmatrix} \Sigma_{0|0}^{-1/2} & & & \\ V_0^{-1/2} H_0 & & & \\ & \ddots & & \\ & & \ddots & \\ & & & V_T^{-1/2} H_T \\ W_0^{-1/2} F_0 & -W_0^{-1/2} & & \\ & W_1^{-1/2} F_1 & \ddots & \\ & & \ddots & -W_{T-1}^{-1/2} \\ & & & W_{T-1}^{-1/2} F_{T-1} \end{bmatrix}}_{J} \begin{pmatrix} \tilde{x}_0 \\ \vdots \\ \tilde{x}_T \end{pmatrix} - \underbrace{\begin{bmatrix} 0 \\ V_0^{-1/2} \tilde{z}_0 \\ \vdots \\ \vdots \\ V_T^{-1/2} \tilde{z}_T \\ W_0^{-1/2} \eta_1 \\ W_1^{-1/2} \eta_2 \\ \vdots \\ W_{T-1}^{-1/2} \eta_T \end{bmatrix}}_{b} \right\|_2^2$$

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

Sparse Least Squares

- Via linearization, we managed to reduce the SLAM problem to:

$$\arg\max_{x_{0:T}} \log p(x_{0:T}, z_{0:T}, u_{0:T-1}) \xrightarrow[\text{initial guess } \hat{x}_{0:T}]{\text{linearize around}} \hat{x}_{0:T} + \arg\min_{\tilde{x}_{0:T}} \|J\tilde{x}_{0:T} - b\|_2^2$$

- The matrix of Jacobians J is **sparse**
- $J^T J$ is the **information matrix** of the joint Gaussian distribution of $x_{0:T} \mid z_{0:T}, u_{0:T-1}$
- Setting the gradient to zero leads to the **Normal equations**:

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

- Can be solved via **Cholesky decomposition** of $J^T J$
- A more efficient and robust way, which avoids having to compute the information matrix $J^T J$ (which also squares the condition number), is **QR factorization**

Solution via QR Factorization

- QR factorization: $J = Q \begin{bmatrix} R \\ 0 \end{bmatrix} \in \mathbb{R}^{m \times n}$
- The number of variables (nodes) is n
- The number of constraints (factors) is m
- $R \in \mathbb{R}^{n \times n}$ is the **upper triangular square root information matrix** since $R^T R = J^T J$
- $Q \in \mathbb{R}^{m \times m}$ is an orthogonal matrix
- Solution via QR factorization:

$$
\begin{aligned}
\|J\tilde{x}_{0:T} - b\|_2^2 &= \left\| Q \begin{bmatrix} R \\ 0 \end{bmatrix} \tilde{x}_{0:T} - b \right\|_2^2 = \left\| Q^T Q \begin{bmatrix} R \\ 0 \end{bmatrix} ildex_{0:T} - Q^T b \right\|_2^2 \\
&= \left\| \begin{bmatrix} R \\ 0 \end{bmatrix} \tilde{x}_{0:T} - \begin{bmatrix} b_1' \\ b_2' \end{bmatrix} \right\|_2^2 = \|R\tilde{x}_{0:T} - b_1'\|_2^2 + \underbrace{\|b_2'\|_2^2}_{\text{residual}}
\end{aligned}
$$

- Since R is upper triangular, simple back-substitution can be used to compute $\tilde{x}^*_{0:T}$ — leading to a least squares estimate for the complete robot trajectory as well as all landmarks $x_{0:T}$ conditioned on all measurements $z_{0:T}$, $u_{0:T-1}$

Factor Graph SLAM Summary

- The factor graph view of SLAM leads to a nonlinear least squares problem
- Assuming an initial estimate of the robot trajectory and landmark poses is available (e.g., from odometry and triangulation of 2-D image features), we can use the Gauss-Newton algorithm to solve the nonlinear least squares problem
- Gauss-Newton iterates between linearizing the system and solving the resulting linear equation to update the pose-landmark estimates
- 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 constraints between non-successive robot poses