Pose Graph SLAM

$$e_{ij}(x) = z_{ij} - \hat{z}(x_i, x_j)$$

$e_{ij}(x)$ adalah fungsi error, z berdasarkan measurement model, $\hat{z}$ berdasarkan model matematika, dan x merupakan state. Diasumsikan error pada data sensor memiliki distribusi Gaussian,

$$p(x) = \frac{1}{\sqrt{2\pi^n|\Sigma|}} \exp\left(-\frac{1}{2}(x-\mu)^T\Omega(x-\mu)\right) \sim \mathcal{N}(\mu, \Sigma)$$

$\Omega = \Sigma^{-1}$ merupakan information matrix, sehingga error merupakan multivariate normal distribution dengan $\mu = \hat{z}$ dan varians Σ. Dengan menerapkan log-likelihood pada $p(x)$, state optimal x^* dapat diperoleh dengan Maximum Likelihood Estimization, x optimal apabila $\ln(p(x))$ mencapai nilai maksimum.

$$x^* = \operatorname{argmax} p(x) = \underset{x}{\operatorname{argmin}}\ e_{ij}^T \Omega_{ij} e_{ij}$$

Dikarenakan adanya kumulatif error pada trajectory, yang merupakan error bukan hanya pada 1 pose melainkan terdapat kemungkinan error terjadi lebih dari 1 pose. Sehingga fungsi error merupakan jumlah dari setiap error pada setiap pose,

$$F(x) = e_{ij}^T \Omega_i e_{ij}$$
$$x^* = \underset{x}{\operatorname{argmin}} \sum_{i,j} F(x_{ij})$$

Persamaan tersebut tidak memiliki solusi pasti dikarenakan terdapat matriks rotasi yang mengindikasikan persamaan non-linear. Permasalahan dapat diselesaikan dengan menggunakan optimisasi dan dilinearkan menggunakan Taylor-Expansion.

$$F_{ij}(x+\Delta x) = e_{ij}(x+\Delta x)^T \Omega_{ij} e_{ij}(x+\Delta x)^T$$

Dengan menggunakan **First-Order Taylor Expansion**,

$$e_{ij}(x+\Delta x) = e_{ij} + J_{ij}\Delta x$$

Subtitusikan ke $F_{ij}(x+\Delta x)$,

$$\begin{aligned}
F_{ij}(x+\Delta x) &= (e_{ij} + J_{ij}\Delta x)^T \Omega_{ij}(e_{ij} + J_{ij}\Delta x) \\
&= (e_{ij}^T + (J_{ij}\Delta x)^T) + (\Omega_{ij} e_{ij} + \Omega_{ij} J_{ij}\Delta x)(\\
&= (e_{ij}^T + \Delta x^T J_{ij}^T)(\Omega_{ij} e_{ij} + \Omega_{ij} J_{ij}\Delta x) \\
&= e_{ij}^T \Omega_{ij} e_{ij} + e_{ij}^T \Omega_{ij} J_{ij}\Delta x + \Delta x^T J_{ij}^T \Omega_{ij} e_{ij} + \Delta x^T J_{ij}^T \Omega_{ij} J_{ij}\Delta x \\
&= e_{ij}^T \Omega_{ij} e_{ij} + e_{ij}^T \Omega_{ij} J_{ij}\Delta x + \left(\Delta x^T J_{ij}^T \Omega_{ij} e_{ij}\right)^T + \Delta x^T J_{ij}^T \Omega_{ij} J_{ij}\Delta x \\
&= e_{ij}^T \Omega_{ij} e_{ij} + e_{ij}^T \Omega_{ij} J_{ij}\Delta x + e_{ij}^T \Omega_{ij}^T J_{ij}\Delta x + \Delta x^T J_{ij}^T \Omega_{ij} J_{ij}\Delta x
\end{aligned}$$

Dikarenakan Ω_{ij} adalah metriks simetrik, maka $\Omega_{ij} = \Omega_{ij}^T$

$$F_{ij}(x+\Delta x) = \underbrace{e_{ij}^T \Omega_{ij} e_{ij}}_{c} + 2\underbrace{e_{ij}^T \Omega_{ij} J_{ij}}_{b} \Delta x + \Delta x^T \underbrace{J_{ij}^T \Omega_{ij} J_{ij}}_{H} \Delta x$$

Dapat diperoleh turunan pertama dari $F_{ij}(x+\Delta x)$,

$$\frac{\partial F_{ij}(x+\Delta x)}{\partial \Delta x} \approx 2b + 2H\Delta x = 0$$

$$\Delta x = -H^{-1} b$$

$$x \leftarrow x + \Delta x$$

$\hat{z}_{ij}$ dihitung dengan *inverse pose composition*,

$$T \in SE(n)$$

$$R \in SO(n)$$

$$T_i^{-1} = \begin{bmatrix} R_i^T & -R_i^T t_i \\ 0 & 1 \end{bmatrix}$$

$$T_j = \begin{bmatrix} R_j & t_j \\ 0 & 1 \end{bmatrix}$$

$$\hat{z}_{ij} = T_i^{-1} \cdot T_j$$

$$\hat{z}_{ij} = \begin{bmatrix} R_i^T R_j & R_i^T (t_j - t_i) \\ 0 & 1 \end{bmatrix} \triangleq \begin{bmatrix} R_i^T (t_j - t_i) \\ \theta_j - \theta_i \end{bmatrix} \rightarrow \text{vector space}$$

$$\text{vector space} = \begin{bmatrix} \Delta t \\ \Delta \theta \end{bmatrix}$$

Nilai e_{ij} diperoleh dengan cara yang sama seperti $\hat{z}_{ij}$, Sehingga $e_{ij} = z_{ij}^{-1} T_i^{-1} T_j$

$$e_{ij} = \begin{bmatrix} R_{ij}^T (R_i^T (t_j - t_i) - t_{ij}) \\ \theta_j - \theta_i - \theta_{ij} \end{bmatrix} \rightarrow \text{vector space}$$

Tentukan turunan parsial e_{ij} terhadap x_i dan x_j,

$$x_i = \begin{bmatrix} t_i \\ \theta_i \end{bmatrix};\ x_j = \begin{bmatrix} t_j \\ \theta_j \end{bmatrix}$$

$$\frac{\partial e_{ij}}{\partial x_i} = \begin{bmatrix} \frac{\partial \Delta t}{\partial t_i} & \frac{\partial \Delta t}{\partial \theta_i} \\ \frac{\partial \Delta \theta}{\partial t_i} & \frac{\partial \Delta \theta}{\partial \theta_i} \end{bmatrix},\ \frac{\partial e_{ij}}{\partial x_i} = \begin{bmatrix} \frac{\partial \Delta t}{\partial t_j} & \frac{\partial \Delta t}{\partial \theta_j} \\ \frac{\partial \Delta \theta}{\partial t_j} & \frac{\partial \Delta \theta}{\partial \theta_j} \end{bmatrix}$$

$$\frac{\partial e_{ij}}{\partial x_i} = \begin{bmatrix} -R_{ij}^T R_i^T & R_{ij}^T \frac{\partial R_i^T}{\partial \theta_i}(t_j - t_i) \\ 0 & -1 \end{bmatrix}$$

$$\frac{\partial e_{ij}}{\partial x_j} = \begin{bmatrix} R_{ij}^T R_i^T & 0 \\ 0 & 1 \end{bmatrix}$$

Jacobian matriks J_{ij} hanya berisikan nilai $\frac{\partial e_{ij}}{\partial x_i}$ pada baris ke-i, $\frac{\partial e_{ij}}{\partial x_j}$ pada baris ke-j, dan yang lainnya bernilai 0

$$J_{ij} = \begin{bmatrix} 0 & 0 & 0 & \dots & \frac{\partial e_{ij}}{\partial x_i} & \dots & \frac{\partial e_{ij}}{\partial x_j} & \dots & 0 & 0 & 0 \end{bmatrix}$$

$$\nabla_{ij} = J_{ij}^T \Omega_{ij} e_{ij}$$
$$H_{ij} = J_{ij}^T \Omega_{ij} J_{ij}$$

$$\nabla = \sum_{i,j} J_{ij}^T \Omega_{ij} e_{ij}$$
$$H = \sum_{i,j} J_{ij}^T \Omega_{ij} J_{ij}$$

Setelah ∇ dan H dapat diperoleh, permasalahan Non-Linear Least Square dapat diselesaikan dengan iterasi proses optimisasi menggunakan Gauss-Newton atau Levenberg-Marquardt hingga $\Delta F(x) < \text{tolerance}$.

$$\text{(GN) } \Delta x = -H^{-1} \nabla$$
$$\text{(LM) } \Delta x = -(H + \lambda \, \text{diag}(H))^{-1} \nabla$$
$$x \leftarrow x + \Delta x$$

References :

- [SLAM][En] Errors and Jacobian Derivations for SLAM Part 1
- A Technical Walkthrough of the SLAM Back-end
- Graph-based SLAM using Pose Graphs (Cyrill Stachniss)