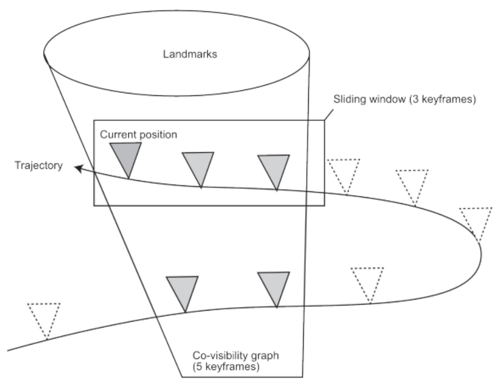
Notes on Visual SLAM 14 lectures – Chapter 10

1 Sliding window filter and optimization

BA: graph optimization of camera poses and spatial points, useful in SfM, but real-time SLAM problems require control of the problem's scale.
Sliding window: only keep N keyframes
Covisibility graph: features that are observed together with the current keyframe. Take key frames and landmarks in the covisibility graph.



1.1 Sliding window

Assume there are N keyframes in this window, and their poses are denoted as: $x_1, ..., x_N$. After marginalization, we get the conditional distribution of the poses:

$$[x_1, ..., x_N | y_1, ..., y_M] \sim N([\mu_1, ..., \mu_N]^T, \Sigma) \tag{1}$$

in which μ_k is the mean of k-th keyframe, and Σ is the covariance matrix of all keyframes. Mean part refers to the optimal result after BA, and Σ is the result of marginalization.

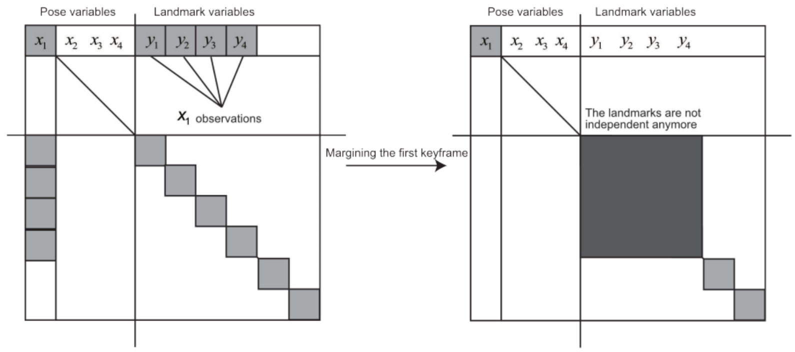
1.1.1 Adding new keyframes

delete the old keyframe x_1 but the whole problem will no longer be sparse. Fill-in: marginalization of x_1 will make the Hessian of the landmark-landmark part filed with information by the operation, and make it no longer a diagonal block.
Make modifications tot he marginalization process, we can maintain the sliding window BA's sparsity, for y_1 to y_4, they may fall into the three cases listed below:

A. landmark is only observed in x_1 and does not appear in the remaining keyframes, can throw away the landmark without any impact on the window, isolating the landmark

B. the landmark is seen in $x_2 - x_4$ but not seen in the future. When road signs are marginalized, a priori of the pose-pose part is generated, so it becomes the prior information of the future pose estimation

C. the landmark is seen in $x_2 - x_4$ and maybe seen in the future. Then this landmark should not be marginalized, because we will need to update its estimate later. We can pretend that the observation of this landmark by x_1 can be simply discarded (equivalent to thinking that x_1 did not see it), so we kept the diagonal structure of the landmark part at a small cost. In this case don't do anything



1.1.2 Intuitive explanation of the marginalization in SWF

Marginalization: decomposing a joint distribution into a conditional and marginal distribution. Marginalize the keyframe means keeping the current estimated value of this keyframe and find the conditional probability of other state variables conditioned on this keyframe.

- keyframe is marginalized: the landmark points it observes will generate a priori information of there these landmarks should be
- landmark points marginalized: a priori information of where the keyframe to observe them should be

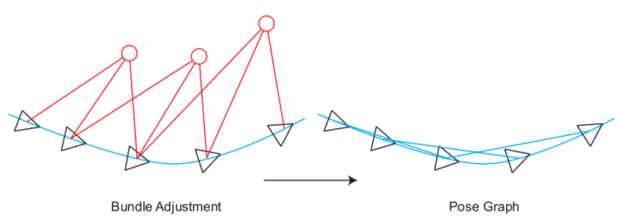
mathematically:

$$p(x_1, ..., x_4, y_1, ..., y_6) = p(x_2, ..., x_4, y_1, ..., y_6|x_1)p(x_1) \tag{2}$$

in which $p(x_1)$ is discarded in this case, after it is marginalized.

2 Pose graph optimization

Ignore the landmarks at all and only focus on the poses: edges between the pose vertices can be set with measurement by the ego-motion estimation obtained from feature matching; once initial estimation is completed, no longer optimize the positions of the landmarks points but only care about the connections between all camera poses. Node here represents camera pose $T_1, ..., T_n$, the edge is the estimation of the relative motion



between the two pose nodes.
Estimation may come from feature point method or the direct method, or GPS or IMU integration. A movement between T_i and T_i is ΔT_{ij}, and can be expressed as:

$$\Delta\xi_{ij} = \xi_i^{-1} \circ \xi_j = \ln(T_i^{-1} T_j)^\vee \tag{3}$$

or in SE(3):

$$T_{ij} = T_i^{-1} T_j \tag{4}$$

Move T_{ij} in the above equation to the right side:

$$e_{ij} = \ln(T_{ij}^{-1} T_i^{-1} T_j)^\vee \tag{5}$$

there are two optimization variables ξ_i and ξ_j, so we find the derivative of e_{ij} about these two variables. According to the derivation method of Lie algebra, give ξ_i and ξ_j a left disturbance $\delta\xi_i$ and $\delta\xi_j$, then the error becomes:

$$\ln(T_{ij}^{-1}T_i^{-1}\exp((-\delta\xi_i)^\wedge)\exp(\delta\xi_j^\wedge)T_j)^\vee \tag{6}$$

Recall the adjoint property:

$$T\exp(\xi^\wedge)T^{-1} = \exp((\mathrm{Ad}(T)\xi)^\wedge) \tag{7}$$

in which $\mathrm{Ad}(T) = \begin{bmatrix} R & t^\wedge R \\ 0 & R \end{bmatrix}$ rearrange it:

$$\exp(\xi^\wedge)T = T\exp((\mathrm{Ad}(T^{-1})\xi)^\wedge) \tag{8}$$

then we have:

$$\begin{aligned} \hat{e}_{ij} &= \ln(T_{ij}^{-1}T_i^{-1}\exp((-\delta\xi_i)^\wedge)\exp(\delta\xi_j^\wedge)T_j)^\vee \\ &= \ln(T_{ij}^{-1}T_i^{-1}T_j\exp((-\mathrm{Ad}(T_j^{-1}\delta\xi_i)^\wedge)\exp((\mathrm{Ad}(T_j^{-1}\delta\xi_j)^\wedge))^\vee \\ &\approx \ln(T_{ij}^{-1}T_i^{-1}T_j[I-(\mathrm{Ad}(T_j^{-1})\delta\xi_i)^\wedge+(\mathrm{Ad}(T_j^{-1})\delta\xi_j)^\wedge])^\vee \\ &\approx e_{ij}+\frac{\partial e_{ij}}{\partial\delta\xi_i}\delta\xi_i+\frac{e_{ij}}{\partial\delta\xi_j}\delta\xi_j \end{aligned}$$

in which the two Jacobians are:

$$\frac{\partial e_{ij}}{\partial\delta\xi_i} = -\mathcal{J}_r^{-1}(e_{ij})\mathrm{Ad}(T_j^{-1}) \tag{9}$$

and

$$\frac{\partial e_{ij}}{\partial\delta\xi_j} = \mathcal{J}_r^{-1}(e_{ij})\mathrm{Ad}(T_j^{-1}) \tag{10}$$

if the error is close to zero, we have:

$$\mathcal{J}_r^{-1}(e_{ij}) \approx I + \frac{1}{2}\begin{bmatrix} \phi_e^\wedge & \rho_e^\wedge \\ 0 & \phi_e^\wedge \end{bmatrix} \tag{11}$$

Let ε be the set of all edges, the overall objective function is:

$$\min \frac{1}{2}\sum_{i,j\in\varepsilon} e_{ij}^T\Sigma_{ij}^{-1}e_{ij} \tag{12}$$