Supplementary Material to: Tightly Coupled 3D Lidar Inertial Odometry and Mapping

Haoyang Ye, Yuying Chen and Ming Liu

A IMU Pre-integration

Similar to [1] and [2], the raw IMU inputs $\hat{\mathbf{a}}_k$ and $\hat{\boldsymbol{\omega}}_k$ can be converted into pre-integration measurements as follows,

$$\begin{aligned}
\Delta\mathbf{p}_{ij} &= \mathbf{R}_i^T\left(\mathbf{p}_j - \mathbf{p}_i - \mathbf{v}_i\Delta t_{ij} - \frac{1}{2}\mathbf{g}^W\Delta t_{ij}^2\right) \\
&= \sum_{k=i}^{j-1}\left[\Delta\mathbf{v}_{ik}\Delta t + \frac{1}{2}\mathbf{R}(\Delta\mathbf{q}_{ik})(\hat{\mathbf{a}}_k - \mathbf{b}_{a_k})\Delta t^2\right] \\
\Delta\mathbf{v}_{ij} &= \mathbf{R}_i^T\left(\mathbf{v_j} - \mathbf{v_i} - \mathbf{g}^W\Delta t_{ij}\right) \\
&= \sum_{k=i}^{j-1}\mathbf{R}(\Delta\mathbf{q}_{ik})(\hat{\mathbf{a}}_k - \mathbf{b}_{a_k})\Delta t \\
\Delta\mathbf{q}_{ij} &= \mathbf{q}_i^{-1}\otimes\mathbf{q}_j = \prod_{k=i}^{j-1}\delta\mathbf{q}_k = \prod_{k=i}^{j-1}\begin{bmatrix}\frac{1}{2}\Delta t(\hat{\boldsymbol{\omega}}_k - \mathbf{b}_{g_k}) \\ 1\end{bmatrix}.
\end{aligned} \tag{A.1}$$

The body motion between the timestamps i and the j lidar sweeps can be represented via pre-integration measurements in Eq. (A.3), where the small angle approximation [3] is applied in Eq. (A.2).

$$\delta\mathbf{q} = \begin{bmatrix}\frac{1}{2}\delta\boldsymbol{\theta} \\ 1\end{bmatrix}, \tag{A.2}$$

$$\begin{aligned}
\mathbf{p}_j &= \mathbf{p}_i + \mathbf{v}_i\Delta t_{ij} + \frac{1}{2}\mathbf{g}^W\Delta t_{ij}^2 \\
&\quad + \mathbf{R}_i\left(\Delta\mathbf{p}_{ij} + \mathbf{J}_{\mathbf{b}_{g_i}}^{\Delta\mathbf{p_{ij}}}\delta\mathbf{b}_{g_i} + \mathbf{J}_{\mathbf{b}_{a_i}}^{\Delta\mathbf{p_{ij}}}\delta\mathbf{b}_{a_i}\right) \\
\mathbf{v}_j &= \mathbf{v}_i + \mathbf{g}^W\Delta t_{ij} \\
&\quad + \mathbf{R}_i\left(\Delta\mathbf{v}_{ij} + \mathbf{J}_{\mathbf{b}_{g_i}}^{\Delta\mathbf{v_{ij}}}\delta\mathbf{b}_{g_i} + \mathbf{J}_{\mathbf{b}_{a_i}}^{\Delta\mathbf{v_{ij}}}\delta\mathbf{b}_{a_i}\right) \\
\mathbf{q}_j &= \mathbf{q}_i\otimes\left(\Delta\mathbf{q}_{ij}\otimes\begin{bmatrix}\frac{1}{2}\mathbf{J}_{\mathbf{b}_{g_i}}^{\Delta\boldsymbol{\theta}_{ij}}\delta\mathbf{b}_{g_i} \\ 1\end{bmatrix}\right),
\end{aligned} \tag{A.3}$$

where $\Delta\mathbf{p}_{ij}$, $\Delta\mathbf{v}_{ij}$ and $\Delta\mathbf{q}_{ij}$ are the pre-integration measurements of the position, velocity and orientation, and $\mathbf{J}^{\Delta(\cdot)}_{\mathbf{b}_{g_i}}$ and $\mathbf{J}^{\Delta(\cdot)}_{\mathbf{b}_{a_i}}$ are the Jacobians of the pre-integration measurements w.r.t to the IMU biases. When the correction of the bias $\delta\mathbf{b}_{(\cdot)}$ is slight, the pre-integration measurements can be updated efficiently using Eq. (A.3). $\Delta\boldsymbol{\theta}_{ij}$ is the error state of the pre-integration quaternion. We denote the pre-integration measurement between i and j as $z^i_j = \{\Delta\mathbf{p}_{ij}, \Delta\mathbf{v}_{ij}, \Delta\mathbf{q}_{ij}\}$. With the continuous-time linearized propagation of the error states and the IMU noise parameters, we can estimate the covariances $\mathbf{C}^{B_i}_{B_j}$ of the pre-integration measurements and biases.

The IMU pre-integration measurements are unrelated to the positions, velocities and orientations of the IMU, and are only related to the raw IMU inputs and biases. When the biases change slightly, the measurements can be efficiently updated by Equation (A.3).

B Covariance of the Pre-integration Measurement

The covariance of the IMU states can be discretized from the continuous-time error-state model[3]. Similarly, the pre-integration measurement of the IMU can be written in such a way [1]. The noises from the raw IMU inputs are modeled as Gaussian white noises:

$$\mathbf{n}_a \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}_a^2), \mathbf{n}_g \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}_g^2). \tag{B.4}$$

And the corresponding biases are modeled as a random walk:

$$\begin{gathered}\dot{\mathbf{b}}_{a_t} = \mathbf{n}_{b_a}, \dot{\mathbf{b}}_{g_t} = \mathbf{n}_{b_g} \\ \mathbf{n}_{b_a} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}_{b_a}^2), \mathbf{n}_{b_g} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}_{b_g}^2)\end{gathered}. \tag{B.5}$$

Adopting the above assumptions, together with the small angle approximation, Equation (A.2), the error-state model of the pre-integration measurement can be written as

$$\begin{aligned}\begin{bmatrix}\delta\dot{\Delta\mathbf{p}}_{it} \\ \delta\dot{\Delta\mathbf{v}}_{it} \\ \delta\dot{\Delta\boldsymbol{\theta}}_{it} \\ \delta\dot{\mathbf{b}}_{a_t} \\ \delta\dot{\mathbf{b}}_{g_t}\end{bmatrix} &= \begin{bmatrix}0 & \mathbf{I} & 0 & 0 & 0 \\ 0 & 0 & -\mathbf{R}^i_t\lfloor\hat{\mathbf{a}}_t - \mathbf{b}_{a_t}\rfloor_\times & -\mathbf{R}^i_t & 0 \\ 0 & 0 & -\lfloor\hat{\boldsymbol{\omega}}_t - \mathbf{b}_{g_t}\rfloor_\times & 0 & -\mathbf{I} \\ & & \mathbf{0} & & \\ & & \mathbf{0} & & \end{bmatrix}\begin{bmatrix}\delta\Delta\mathbf{p}_{it} \\ \delta\Delta\mathbf{v}_{it} \\ \delta\Delta\boldsymbol{\theta}_{it} \\ \delta\mathbf{b}_{a_t} \\ \delta\mathbf{b}_{g_t}\end{bmatrix}, \\ &+ \begin{bmatrix}0 & 0 & 0 & 0 \\ -\mathbf{R}^i_t & 0 & 0 & 0 \\ 0 & -\mathbf{I} & 0 & 0 \\ 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & 0 & \mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{n}_{a_t} \\ \mathbf{n}_{g_t} \\ \mathbf{n}_{b_{a_t}} \\ \mathbf{n}_{b_{g_t}}\end{bmatrix} = \mathbf{F}_t\delta\mathbf{z}_{it} + \mathbf{G}_t\mathbf{n}_t\end{aligned} \tag{B.6}$$

where $\lfloor\cdot\rfloor_\times$ stands for the skew-symmetric matrix. i denotes the timestamp of the last IMU state and $t \in [i, j]$ denotes the continuous timestamp with i and j. The pre-integration items in the equation are continuously integrated, as introduced in [1].

Then, we can use the first-order update recursively to get the covariance of the pre-integration error state $\mathbf{C}_{B_j}^{B_i}$ and its Jacobian w.r.t the error states $\mathbf{J}^{ij}$,

$$\begin{aligned}
&\mathbf{C}_{B_i}^{B_i} = \mathbf{0} \\
&\mathbf{C}_{B_{t+\delta t}}^{B_i} = (\mathbf{I} + \mathbf{F}_t \delta t)\mathbf{C}_{B_t}^{B_i}(\mathbf{I} + \mathbf{F}_t \delta t)^T \\
&\qquad\qquad + (\mathbf{G}_t \delta t)\mathbf{Q}(\mathbf{G}_t \delta t)^T \qquad\qquad , \\
&\mathbf{J}^{i,i} = \mathbf{I} \\
&\mathbf{J}^{i,t+\delta t} = (\mathbf{I} + \mathbf{F}_t \delta t)\mathbf{J}^{i,t}
\end{aligned} \tag{B.7}$$

where δt is the time interval between time-consecutive raw IMU inputs, $\mathbf{Q}$ is $\text{diag}(\boldsymbol{\sigma}_a^2, \boldsymbol{\sigma}_g^2, \boldsymbol{\sigma}_{b_a}^2, \boldsymbol{\sigma}_{b_g}^2)$, and $\mathbf{C}_{B_t}^{B_i}$ and $\mathbf{J}^{i,t}$ are the corresponding covariance and Jacobian at timestap $t \in [i, j]$. The final $\mathbf{J}^{i,j}$ includes the $\mathbf{J}_{\mathbf{b}(\cdot)}^{\Delta(\cdot)}$ needed in Equation (A.3).

C IMU Residual

$$\begin{aligned}
&\mathbf{r}_\mathcal{B}(z_{\beta+1}^\beta, \mathbf{X}) \\
&= \begin{bmatrix}
\mathbf{R}_\beta^T \left(\mathbf{p}_{\beta+1} - \mathbf{p}_\beta - \mathbf{v}_\beta \Delta t - \frac{1}{2}\mathbf{g}^W \Delta t^2\right) - \Delta\mathbf{p} \\
\mathbf{R}_\beta^T \left(\mathbf{v}_{\beta+\mathbf{1}} - \mathbf{v}_\beta - \mathbf{g}^W \Delta t\right) - \Delta\mathbf{v} \\
2\left[\Delta\mathbf{q}^{-1} \otimes \mathbf{q}_\beta^{-1} \otimes \mathbf{q}_{\beta+1}\right]_{xyz} \\
\mathbf{b}_{a_{\beta+1}} - \mathbf{b}_{a_\beta} \\
\mathbf{b}_{g_{\beta+1}} - \mathbf{b}_{g_\beta}
\end{bmatrix},
\end{aligned} \tag{C.8}$$

where $\Delta t = \Delta t_{\beta\beta+1}$, $\Delta\mathbf{p} = \Delta\mathbf{p}_{\beta\beta+1}$, $\Delta\mathbf{v} = \Delta\mathbf{v}_{\beta\beta+1}$ and $\Delta\mathbf{q} = \Delta\mathbf{q}_{\beta\beta+1}$ for clarity, and $[\cdot]_{xyz}$ stands for the vector part of a quaternion.

D Marginalization

According to the local window, some of the states will be marginalized out after the whole optimization. We denote the marginalized states as $\mathbf{X}_m$, the remaining states related to $\mathbf{X}_m$ as $\mathbf{X}_r$ and the irrelevant states as $\mathbf{X}_n$. Since the marginalization will not influence $\mathbf{X}_m$, in the following, the states we mention will contain $\mathbf{X}_m$ and $\mathbf{X}_r$ only.

For the cost function in Equation (6), the Gaussian-Newton algorithm will utilize the form $\mathbf{H}\delta\mathbf{X} = -\mathbf{b}$, where $\mathbf{H} = \sum \mathbf{J}^T\mathbf{C}^{-1}\mathbf{J}$ and $\mathbf{b} = \sum \mathbf{J}^T\mathbf{C}^{-1}\mathbf{r}$ consist of $\mathbf{r}$, the residuals of the cost; $\mathbf{J}$, Jacobians of the residual w.r.t the states; and $\mathbf{C}$, the covariance matrices of the states. Here, $\delta\mathbf{X}$ is the error state we will actually estimate in the optimization, with which we can have the update as

$\tilde{\mathbf{X}} = \mathbf{X} \boxplus \delta\mathbf{X}$. $\boxplus$ denotes the small angle update for the quaternion, Equation (A.2), and the addition for other states in $\mathbf{X}$. $\mathbf{H}$ and $\mathbf{b}$ are from the sum of all terms in Equation (6). The superscripts and subscripts are omitted for simplicity.

Using block matrices, we can have

$$\begin{bmatrix} \mathbf{H}_{mm} & \mathbf{H}_{mr} \\ \mathbf{H}_{rm} & \mathbf{H}_{rr} \end{bmatrix} \begin{bmatrix} \delta\mathbf{X}_m \\ \delta\mathbf{X}_r \end{bmatrix} = - \begin{bmatrix} \delta\mathbf{b}_m \\ \delta\mathbf{b}_r \end{bmatrix}. \tag{D.9}$$

With the Schur complement, the remaining parts will be

$$\begin{aligned} \mathbf{H}_{\mathcal{P}} &= \mathbf{H}^*_{rr} = \mathbf{H}_{rr} - \mathbf{H}_{rm}\mathbf{H}^{-1}_{mm}\mathbf{H}_{mr} \\ \mathbf{b}_{\mathcal{P}_0} &= \mathbf{b}^*_r = \mathbf{b}_r - \mathbf{H}_{rm}\mathbf{H}^{-1}_{mm}\mathbf{b}_m \end{aligned}. \tag{D.10}$$

We denote $\check{\mathbf{X}}_r$ as the remaining states when marginalization takes place. In the next optimization, the new estimation of the remaining states will be in the form of $\tilde{\mathbf{X}}_r = \check{\mathbf{X}}_r \boxplus \delta\mathbf{X}_r$. It will return the new $\mathbf{b}_{\mathcal{P}} = \mathbf{b}^*_r$ as

$$\mathbf{b}_{\mathcal{P}} = \mathbf{b}^*_r = \mathbf{b}_{\mathcal{P}_0} - \mathbf{H}_{\mathcal{P}}\delta\mathbf{X}_r, \tag{D.11}$$

where $\mathbf{b}_{\mathcal{P}_0}$ is fixed when the marginalization takes place. This coincides with the "first-estimate Jacobians" [4, 5], which are applied to maintain the consistency of the estimation. With $\mathbf{H}_{\mathcal{P}}$ and $\mathbf{b}_{\mathcal{P}}$ provided, the error item from the marginalization can be written as $\|\mathbf{r}_{\mathcal{P}}(\mathbf{X})\|^2 = \mathbf{b}^T_{\mathcal{P}}\mathbf{H}^+_{\mathcal{P}}\mathbf{b}_{\mathcal{P}}$.

E Derivative of the Rotation-constrained Mapping

We formulate the rotation-constrained mapping by considering $\delta\bar{\mathbf{q}}_z$, an orientation correction on the z-axis only, first.

$$\begin{aligned} \tilde{\mathbf{q}}^W_L &= \delta\bar{\mathbf{q}}_z \otimes \check{\mathbf{q}}^W_L = \check{\mathbf{q}}^W_L \otimes \delta\mathbf{q}^W_L \\ \delta\mathbf{q}^W_L &= (\check{\mathbf{q}}^W_L)^* \otimes \delta\bar{\mathbf{q}}_z \otimes \check{\mathbf{q}}^W_L \end{aligned}. \tag{E.12}$$

The quaternion $\delta\mathbf{q}$ describing a small rotation can be converted to the corresponding rotation matrix $\mathbf{R}(\delta\mathbf{q})$ as

$$\mathbf{R}(\delta\mathbf{q}) = \mathbf{R}(\delta\boldsymbol{\theta}) \approx \mathbf{I}_{3\times3} + \lfloor\delta\boldsymbol{\theta}\rfloor_\times, \tag{E.13}$$

where small angle approximation (A.2) is applied. Substituting (E.12) and E.13 into the cost function in (7), the Jacobian w.r.t. the last estimated rotation can be derived in detail as follows:

$$\mathbf{J}^{\mathbf{C}}_{\boldsymbol{\theta}}\delta\boldsymbol{\theta}^W_L = -\boldsymbol{\omega}^T\check{\mathbf{R}}^W_L\lfloor\mathbf{x}\rfloor_\times\delta\boldsymbol{\theta}^W_L = \boldsymbol{\omega}^T\check{\mathbf{R}}^W_L\lfloor\delta\boldsymbol{\theta}^W_L\rfloor_\times\mathbf{x}, \tag{E.14}$$

$$\begin{aligned} \mathbf{R}(\delta\boldsymbol{\theta}^W_L) &= (\check{\mathbf{R}}^W_L)^T \cdot \mathbf{R}(\delta\bar{\boldsymbol{\theta}}_z) \cdot \check{\mathbf{R}}^W_L \\ &= \mathbf{I}_{3\times3} + (\check{\mathbf{R}}^W_L)^T \cdot \lfloor\delta\bar{\boldsymbol{\theta}}_z\rfloor_\times \cdot \check{\mathbf{R}}^W_L, \\ \Rightarrow \lfloor\delta\boldsymbol{\theta}^W_L\rfloor_\times &= (\check{\mathbf{R}}^W_L)^T \cdot \lfloor\delta\bar{\boldsymbol{\theta}}_z\rfloor_\times \cdot \check{\mathbf{R}}^W_L \end{aligned} \tag{E.15}$$

$$
\begin{aligned}
\mathbf{J}^{\mathbf{C}}_{\boldsymbol{\theta}} \delta\boldsymbol{\theta}^W_L &= \boldsymbol{\omega}^T \lfloor \delta\bar{\boldsymbol{\theta}}_z \rfloor_\times \breve{\mathbf{R}}^W_L \mathbf{x} = -\boldsymbol{\omega}^T \lfloor \breve{\mathbf{R}}^W_L \mathbf{x} \rfloor_\times \delta\bar{\boldsymbol{\theta}}_z \\
&= -\boldsymbol{\omega}^T \breve{\mathbf{R}}^W_L \lfloor \mathbf{x} \rfloor_\times (\breve{\mathbf{R}}^W_L)^T \delta\bar{\boldsymbol{\theta}}_z \\
&= \mathbf{J}^{\mathbf{C}}_{\boldsymbol{\theta}} \cdot (\breve{\mathbf{R}}^W_L)^T \delta\bar{\boldsymbol{\theta}}_z
\end{aligned} . \tag{E.16}
$$

With the ideal lidar-IMU odometry, the $\delta\boldsymbol{\theta}_z$ with 3-axes orientation is the same as $\delta\bar{\boldsymbol{\theta}}_z$:

$$
\delta\bar{\boldsymbol{\theta}}_z = \breve{\boldsymbol{\Omega}}_z \cdot \delta\boldsymbol{\theta}_z = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \delta\boldsymbol{\theta}_z = \delta \begin{bmatrix} 0 \\ 0 \\ \theta_z \end{bmatrix} . \tag{E.17}
$$

However, noise cannot be eliminated. Thus, to allow small perturbations on the other two axes, the matrix $\breve{\boldsymbol{\Omega}}_z$ is modified to

$$
\breve{\boldsymbol{\Omega}}_z = \begin{bmatrix} \epsilon_x & 0 & 0 \\ 0 & \epsilon_y & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{E.18}
$$

as an approximation of the information matrix of the orientation of the lidar w.r.t $\mathcal{F}_W$.

Substituting (E.17) and (E.18) into (E.16), the final Jacobian for the rotational components becomes

$$
\mathbf{J}^{\mathbf{C}}_{\boldsymbol{\theta}_z} = \mathbf{J}^{\mathbf{C}}_{\boldsymbol{\theta}} \cdot (\breve{\mathbf{R}}^W_L)^T \cdot \breve{\boldsymbol{\Omega}}_z. \tag{E.19}
$$

Then applying $\mathbf{J}^{\mathbf{C}}_{\boldsymbol{\theta}_z}$, instead of $\mathbf{J}^{\mathbf{C}}_{\boldsymbol{\theta}}$, in the optimization, we can obtain the optimized $\delta\boldsymbol{\theta}_z$. After each iteration, the $\delta\boldsymbol{\theta}_z$, with its corresponding $\delta\mathbf{q}_z$, can be used to update $\tilde{\mathbf{q}}^W_L$ by (E.12) as

$$
\begin{aligned}
\tilde{\mathbf{q}}^W_L &= \breve{\mathbf{q}}^W_L \otimes (\breve{\mathbf{q}}^W_L)^* \otimes \delta\mathbf{q}_z \otimes \breve{\mathbf{q}}^W_L \\
&= \begin{bmatrix} \frac{1}{2}\delta\boldsymbol{\theta}_z \\ 1 \end{bmatrix} \otimes \breve{\mathbf{q}}^W_L
\end{aligned} . \tag{E.20}
$$

References

[1] T. Qin, P. Li, and S. Shen, "Vins-mono: A robust and versatile monocular visual-inertial state estimator," *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 1004–1020, 2018.

[2] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "On-manifold preintegration for real-time visual–inertial odometry," *IEEE Transactions on Robotics*, vol. 33, no. 1, pp. 1–21, 2017.

[3] N. Trawny and S. I. Roumeliotis, "Indirect kalman filter for 3d attitude estimation," *University of Minnesota, Dept. of Comp. Sci. & Eng., Tech. Rep*, vol. 2, p. 2005, 2005.

[4] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, "A first-estimates jacobian ekf for improving slam consistency," in *Experimental Robotics*. Springer, 2009, pp. 373–382.

[5] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual–inertial odometry using nonlinear optimization," *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2015.