Supplementary Material: Rao-Blackwellized Particle Smoother on Lie Groups

In this supplementary material, we provide the mathematical derivations of the Rao-Blackwellized Particle Smoother on Lie Groups. Please read section 4 in the submitted paper before reading this note.

I. Definitions

We remind the adjoint representation $Ad_{SE(3)}\left(\cdot\right) \subset \mathbb{R}^{6\times 6}$ of $SE\left(3\right)$ on $\mathbb{R}^6$ that enables us to transport an increment $\epsilon_{ig}^{i} \in \mathbb{R}^6$, that acts onto an element C_{ig} through left multiplication, into an increment $\epsilon_{ig}^{g} \in \mathbb{R}^6$, that acts through right multiplication:

$$exp_{SE(3)}^{\wedge}\left(\epsilon_{ig}^{i}\right)C_{ig} = C_{ig}exp_{SE(3)}^{\wedge}\left(Ad_{SE(3)}\left(C_{ig}^{-1}\right)\epsilon_{ig}^{i}\right) = C_{ig}exp_{SE(3)}^{\wedge}\left(\epsilon_{ig}^{g}\right) \tag{1}$$

where $\epsilon_{ig}^{g} = Ad_{SE(3)}\left(C_{ig}^{-1}\right)\epsilon_{ig}^{i} = Ad_{SE(3)}\left(C_{gi}\right)\epsilon_{ig}^{i}$. We also remind the expression of the left Jacobian of $SE\left(3\right)$:

$$\Phi_{SE(3)}\left(\epsilon\right) = \sum_{m=0}^{\infty}\frac{1}{(m+1)!}ad_{SE(3)}\left(\epsilon\right)^{m} \tag{2}$$

II. Filtering

We wish to approximate the following probability distribution:

$$p\left(x_{0:t}, s_{0:t}|y_{1:t}\right) = p\left(x_{0:t}|y_{1:t}, s_{0:t}\right)p\left(s_{0:t}|y_{1:t}\right) \tag{3}$$

The term $p\left(s_{0:t}|y_{1:t}\right)$ will be sampled with N_p particles as follows:

$$p\left(s_{0:t}|y_{1:t}\right) \approx \sum_{l=1}^{N_p} w_t^{(l)}\delta\left(s_{0:t} - s_{0:t}^{(l)}\right) \tag{4}$$

The term $p\left(x_{0:t}|y_{1:t}, s_{0:t}\right)$ will be approximated by a concentrated Gaussian distribution on Lie groups using an Extended Kalman Filter on Lie Groups [1].

A. Recursive formula

Let us define:

$$s_{0:t}^{(l)} \sim q\left(s_{0:t}^{(l)}|y_{1:t}\right) = q\left(s_t^{(l)}|s_{0:t-1}^{(l)}, y_{1:t}\right)q\left(s_{0:t-1}^{(l)}|y_{1:t-1}\right) \tag{5}$$

Moreover, the target distribution we are interested in can be expressed as follows:

$$\begin{aligned} p\left(s_{0:t}^{(l)}|y_{1:t}\right) &= \frac{p\left(y_t|s_{0:t}^{(l)}, y_{1:t-1}\right)p\left(s_{0:t}^{(l)}|y_{1:t-1}\right)}{p\left(y_t|y_{1:t-1}\right)} &(6)\\ &= \frac{p\left(y_t|s_{0:t}^{(l)}, y_{1:t-1}\right)p\left(s_t^{(l)}|s_{0:t-1}^{(l)}, y_{1:t-1}\right)p\left(s_{0:t-1}^{(l)}|y_{1:t-1}\right)}{p\left(y_t|y_{1:t-1}\right)} &(7)\end{aligned}$$

Consequently, we obtain the following recursive formula for $w_t^{(l)}$:

$$\begin{aligned} w_t^{(l)} &= \frac{p\left(s_{0:t}^{(l)}|y_{1:t}\right)}{q\left(s_{0:t}^{(l)}|y_{1:t}\right)} = \frac{p\left(y_t|s_{0:t}^{(l)}, y_{1:t-1}\right)p\left(s_t^{(l)}|s_{0:t-1}^{(l)}, y_{1:t-1}\right)p\left(s_{0:t-1}^{(l)}|y_{1:t-1}\right)}{p\left(y_t|y_{1:t-1}\right)q\left(s_t^{(l)}|s_{0:t-1}^{(l)}, y_{1:t}\right)q\left(s_{0:t-1}^{(l)}|y_{1:t-1}\right)} &(8)\\ &= w_{t-1}^{(l)}\frac{p\left(y_t|s_{0:t}^{(l)}, y_{1:t-1}\right)p\left(s_t^{(l)}|s_{0:t-1}^{(l)}, y_{1:t-1}\right)}{p\left(y_t|y_{1:t-1}\right)q\left(s_t^{(l)}|s_{0:t-1}^{(l)}, y_{1:t}\right)} &(9)\end{aligned}$$

B. Optimal Sampling

Let us take:

$$q\left(s_t^{(l)}|s_{0:t-1}^{(l)},y_{1:t}\right)=p\left(s_t^{(l)}|s_{0:t-1}^{(l)},y_{1:t}\right)=\frac{p\left(y_t|s_{0:t}^{(l)},y_{1:t-1}\right)p\left(s_t^{(l)}|s_{0:t-1}^{(l)},y_{1:t-1}\right)}{p\left(y_t|s_{0:t-1}^{(l)},y_{1:t-1}\right)} \tag{10}$$

We obtain the following formula for $w_t^{(l)}$:

$$\begin{aligned} w_t^{(l)}&=w_{t-1}^{(l)}\frac{p\left(y_t|s_{0:t}^{(l)},y_{1:t-1}\right)p\left(s_t^{(l)}|s_{0:t-1}^{(l)},y_{1:t-1}\right)}{p\left(y_t|y_{1:t-1}\right)}.\frac{p\left(y_t|s_{0:t-1}^{(l)},y_{1:t-1}\right)}{p\left(y_t|s_{0:t}^{(l)},y_{1:t-1}\right)p\left(s_t^{(l)}|s_{0:t-1}^{(l)},y_{1:t-1}\right)}\\ &=w_{t-1}^{(l)}\frac{p\left(y_t|s_{0:t-1}^{(l)},y_{1:t-1}\right)}{p\left(y_t|y_{1:t-1}\right)}\propto w_{t-1}^{(l)}p\left(y_t|s_{0:t-1}^{(l)},y_{1:t-1}\right) \end{aligned} \tag{11}$$

The term $p\left(y_t|s_{0:t-1}^{(l)},y_{1:t-1}\right)$ can be computed as:

$$\begin{aligned} p\left(y_t|s_{0:t-1}^{(l)},y_{1:t-1}\right)&=\sum p\left(y_t,s_t|s_{0:t-1}^{(l)},y_{1:t-1}\right)\\ &=\sum p\left(y_t|s_t,s_{0:t-1}^{(l)},y_{1:t-1}\right)p\left(s_t|s_{0:t-1}^{(l)},y_{1:t-1}\right) \end{aligned} \tag{12}$$

In our application $p\left(s_t|s_{0:t-1}^{(l)},y_{1:t-1}\right)=p\left(s_t\right)$, moreover we can employ the following approximation:

$$\begin{aligned} p\left(y_t|s_t,s_{0:t-1}^{(l)},y_{1:t-1}\right)&=\int p\left(x_t,y_t|s_t,s_{0:t-1}^{(l)},y_{1:t-1}\right)dx_t\\ &=\int p\left(y_t|x_t,s_t\right)p\left(x_t|s_t,s_{0:t-1}^{(l)},y_{1:t-1}\right)dx_t\\ &\approx\mathcal{N}_{SE(3)}\left(y_t\left(s_t\right);\mu_{t|t-1}\left(C\right),P_{t|t-1}\left(C\right)+Q\right) \end{aligned} \tag{13}$$

where we approximated $p\left(x_t|s_t,s_{0:t-1}^{(l)},y_{1:t-1}\right)\approx\mathcal{N}_{SE(3)\times\mathbb{R}^6}\left(x_t;\mu_{t|t-1},P_{t|t-1}\right)$ using an Extended Kalman Filter on Lie Groups [1]. Moreover, the notations $\mu_{t|t-1}\left(C\right)$ and $P_{t|t-1}\left(C\right)$ correspond to extracting the sub-matrix corresponding to the camera pose C in $\mu_{t|t-1}$ and $P_{t|t-1}$ respectively.

C. Filtering Algorithm

Algorithm 1 LG-RBPF

Filtering

- For $i=1,2,...,N_p$
  - Initialize $\mu_{0|0}^{(i)}$ by choosing randomly a map coordinate in the database with a null speed
  - $P_{0|0}^{(i)}=P_0$ and $w_0^{(i)}=\frac{1}{N_p}$
- EndFor
- For $t=1,2,...,T$
  - For $i=1,2,...,N_p$
    - **LG-EKF propagation**: Propagate $\mu_{t-1|t-1}^{(i)}$ and $P_{t-1|t-1}^{(i)}$ to get $\mu_{t|t-1}^{(i)}$ and $P_{t|t-1}^{(i)}$ (Alg.2)
    - **Optimal Sampling**: Draw $s_t^{(i)}$ (Alg.4) and evaluate $p\left(y_t|s_{0:t-1}^{(i)},y_{1:t-1}\right)$
    - **LG-EKF update**: Update $\mu_{t|t-1}^{(i)}$ and $P_{t|t-1}^{(i)}$ to get $\mu_{t|t}^{(i)}$ and $P_{t|t}^{(i)}$ (Alg.3)
    - Update weights: $w_t^{(i)}=w_{t-1}^{(i)}p\left(y_t|s_{0:t-1}^{(i)},y_{1:t-1}\right)$
    - Normalize weights and Resample particles
  - EndFor
- EndFor

Algorithm 2 LG-EKF Propagation

- $v = \mu_{t-1|t-1}(v)$
- $C = \mu_{t-1|t-1}(C)$
- $\mu_{t|t-1}(C) = exp^{\wedge}_{SE(3)}(v\Delta t)\, C$
- $\mu_{t|t-1}(v) = v$
- $P_{t|t-1} = FP_{t-1|t-1}F^T + R$
  - where $F = \begin{bmatrix} Ad_{SE(3)}\left(exp^{\wedge}_{SE(3)}(v\Delta t)\right) & \Phi_{SE(3)}(v\Delta t)\,\Delta t \\ 0 & Id \end{bmatrix}$

Algorithm 3 LG-EKF Update

- $K = P_{t|t-1}J^T\left(JP_{t|t-1}J^T + Q\right)^{-1}$
- $\delta = Klog^{\vee}_{SE(3)}\left(y_t\mu_{t|t-1}(C)^{-1}\right)$
- $\mu_{t|t}(C) = exp^{\wedge}_{SE(3)}(\delta(C))\,\mu_{t|t-1}(C)$
- $\mu_{t|t}(v) = \delta(v) + \mu_{t|t-1}(v)$
- $P_{t|t} = \Phi_G(\delta)(Id - KJ)\,P_{t|t-1}\Phi_G^T(\delta)$
  - where $J = \begin{bmatrix} Id & 0 \end{bmatrix}$ and $\Phi_G(\delta) = \begin{bmatrix} \Phi_{SE(3)}(\delta(C)) & 0 \\ 0 & Id \end{bmatrix}$

Algorithm 4 Optimal Sampling

- For $l = 1, ..., N$
  - $S = P_{t|t-1}(C) + Q_{VM}$
  - $e = log^{\vee}_{SE(3)}\left(y_t(l)\,\mu_{t|t-1}(C)^{-1}\right)$
  - $w(l) = \left(\sqrt{(2\pi)^6|S|}\right)^{-1} exp\left(-0.5\left(e^TS^{-1}e\right)\right)p(s_t = l)$
- EndFor
- In practice, to reduce the computational cost of this for loop, we use a kdtree to retrieve the 30 nearest neighbor measurements of $\mu_{t|t-1}$ and perform the for loop only for these 30 measurements.
- For $l = N+1, ..., N+5$
  - $S = P_{t|t-1}(C) + Q_{CBIR}$
  - $e = log^{\vee}_{SE(3)}\left(y_t(l)\,\mu_{t|t-1}(C)^{-1}\right)$
  - $w(l) = \left(\sqrt{(2\pi)^6|S|}\right)^{-1} exp\left(-0.5\left(e^TS^{-1}e\right)\right)p(s_t = l)$
- EndFor
- $p\left(y_t|s^{(i)}_{0:t-1}, y_{1:t-1}\right) = \sum_l w(l)$
- Set $s_t = l$ with probability $w(l)$

III. Smoothing

A. Derivation

We wish to sample from the following distribution:

$$\begin{aligned} p(x_{1:T}, s_{1:T}|y_{1:T}) &= p(x_1, s_1|x_{2:T}, s_{2:T}, y_{1:T})\,p(x_{2:T}, s_{2:T}|y_{1:T}) \\ &= p(x_1, s_1|x_{2:T}, s_{2:T}, y_{1:T})\,p(x_2, s_2|x_{3:T}, s_{3:T}, y_{1:T})\,p(x_{3:T}, s_{3:T}|y_{1:T}) \\ &= p(x_T, s_T|y_{1:T})\prod_{t=1}^{T-1} p(x_t, s_t|x_{t+1:T}, s_{t+1:T}, y_{1:T}) \end{aligned} \tag{14}$$

However:

$$\begin{aligned} p(x_t, s_t|x_{t+1:T}, s_{t+1:T}, y_{1:T}) &= \sum_{s_{1:t}} p(x_t, s_{1:t}|x_{t+1:T}, s_{t+1:T}, y_{1:T}) \\ &= \sum_{s_{1:t}} p(x_t|x_{t+1:T}, s_{1:T}, y_{1:T})\,p(s_{1:t}|x_{t+1:T}, s_{t+1:T}, y_{1:T}) \end{aligned} \tag{15}$$

From the markov property of our model, we have:

$$\begin{aligned} p\left(s_{1:t}|x_{t+1:T},s_{t+1:T},y_{1:T}\right) &= p\left(s_{1:t}|x_{t+1},s_{t+1},y_{1:t}\right) \\ &= \frac{p\left(x_{t+1},s_{t+1}|s_{1:t},y_{1:t}\right)p\left(s_{1:t}|y_{1:t}\right)}{p\left(x_{t+1},s_{t+1}|y_{1:t}\right)} \\ &= \frac{p\left(x_{t+1},s_{t+1}|s_{1:t},y_{1:t}\right)p\left(s_{1:t}|y_{1:t}\right)}{\sum_{s_{1:t}}p\left(x_{t+1},s_{1:t+1}|y_{1:t}\right)} \\ &= \frac{p\left(x_{t+1},s_{t+1}|s_{1:t},y_{1:t}\right)p\left(s_{1:t}|y_{1:t}\right)}{\sum_{s_{1:t}}p\left(x_{t+1},s_{t+1}|s_{1:t},y_{1:t}\right)p\left(s_{1:t}|y_{1:t}\right)} \\ &= \frac{p\left(x_{t+1}|s_{1:t+1},y_{1:t}\right)p\left(s_{t+1}|s_{t}\right)p\left(s_{1:t}|y_{1:t}\right)}{\sum_{s_{1:t}}p\left(x_{t+1}|s_{1:t+1},y_{1:t}\right)p\left(s_{t+1}|s_{t}\right)p\left(s_{1:t}|y_{1:t}\right)} \end{aligned} \tag{16}$$

From the output of the filtering stage, we have:

$$p\left(s_{1:t}|y_{1:t}\right) \approx \sum_{l=1}^{N_p} w_t^{(l)}\delta\left(s_{1:t}-s_{1:t}^{(l)}\right) \tag{17}$$

Thus:

$$p\left(s_{1:t}|x_{t+1:T},s_{t+1:T},y_{1:T}\right) \approx \frac{p\left(x_{t+1}|s_{1:t+1},y_{1:t}\right)p\left(s_{t+1}|s_{t}\right)\left(\sum_{l=1}^{N_p} w_t^{(l)}\delta\left(s_{1:t}-s_{1:t}^{(l)}\right)\right)}{\sum_{s_{1:t}}p\left(x_{t+1}|s_{1:t+1},y_{1:t}\right)p\left(s_{t+1}|s_{t}\right)\left(\sum_{l=1}^{N_p} w_t^{(l)}\delta\left(s_{1:t}-s_{1:t}^{(l)}\right)\right)} \tag{18}$$

Denominator:

$$\sum_{s_{1:t}}p\left(x_{t+1}|s_{1:t+1},y_{1:t}\right)p\left(s_{t+1}|s_{t}\right)\left(\sum_{l=1}^{N} w_t^{(l)}\delta\left(s_{1:t}-s_{1:t}^{(l)}\right)\right)ds_{1:t} = \sum_{l=1}^{N} w_t^{(l)}p\left(x_{t+1}|s_{1:t}^{(l)},s_{t+1},y_{1:t}\right)p\left(s_{t+1}|s_t^{(l)}\right) \tag{19}$$

Numerator:

$$p\left(x_{t+1}|s_{1:t+1},y_{1:t}\right)p\left(s_{t+1}|s_{t}\right)\left(\sum_{l=1}^{N} w_t^{(l)}\delta\left(s_{1:t}-s_{1:t}^{(l)}\right)\right) = \sum_{l=1}^{N} w_t^{(l)}p\left(x_{t+1}|s_{1:t}^{(l)},s_{t+1},y_{1:t}\right)p\left(s_{t+1}|s_t^{(l)}\right)\delta\left(s_{1:t}-s_{1:t}^{(l)}\right) \tag{20}$$

Consequently:

$$p\left(s_{1:t}|x_{t+1:T},s_{t+1:T},y_{1:T}\right) \approx \sum_{l=1}^{N}\frac{w_t^{(l)}p\left(x_{t+1}|s_{1:t}^{(l)},s_{t+1},y_{1:t}\right)p\left(s_{t+1}|s_t^{(l)}\right)}{\sum_{k=1}^{N} w_t^{(k)}p\left(x_{t+1}|s_{1:t}^{(k)},s_{t+1},y_{1:t}\right)p\left(s_{t+1}|s_t^{(k)}\right)}\delta\left(s_{1:t}-s_{1:t}^{(l)}\right) \tag{21}$$

$$= \sum_{l=1}^{N} w_{t|t+1}^{(l)}\delta\left(s_{1:t}-s_{1:t}^{(l)}\right) \tag{22}$$

where

$$w_{t|t+1}^{(l)} = \frac{w_t^{(l)}p\left(x_{t+1}|s_{1:t}^{(l)},s_{t+1},y_{1:t}\right)p\left(s_{t+1}|s_t^{(l)}\right)}{\sum_{k=1}^{N} w_t^{(k)}p\left(x_{t+1}|s_{1:t}^{(k)},s_{t+1},y_{1:t}\right)p\left(s_{t+1}|s_t^{(k)}\right)} \tag{23}$$

$$\propto w_t^{(l)}p\left(x_{t+1}|s_{1:t}^{(l)},s_{t+1},y_{1:t}\right)p\left(s_{t+1}|s_t^{(l)}\right) \tag{24}$$

Thus, we obtain:

$$\begin{aligned} p\left(x_t,s_t|x_{t+1:T},s_{t+1:T},y_{1:T}\right) &= \sum_{s_{1:t}}p\left(x_t|x_{t+1:T},s_{1:T},y_{1:T}\right)p\left(s_{1:t}|x_{t+1:T},s_{t+1:T},y_{1:T}\right) \\ &\approx \sum_{s_{1:t}}p\left(x_t|x_{t+1:T},s_{1:T},y_{1:T}\right)\sum_{l=1}^{N} w_{t|t+1}^{(l)}\delta\left(s_{1:t}-s_{1:t}^{(l)}\right) \\ &\approx \sum_{l=1}^{N} w_{t|t+1}^{(l)}p\left(x_t|x_{t+1:T},s_{1:t}^{(l)},s_{t+1:T},y_{1:T}\right)\delta\left(s_t-s_t^{(l)}\right) \end{aligned} \tag{25}$$

Consequently, knowing $\tilde{x}_{t+1:T}$ and $\tilde{s}_{t+1:T}$, then $\tilde{s}_t$ can be sampled from:

$$\tilde{s}_{1:t} \sim \sum_{l=1}^{N} w_{t|t+1}^{(l)} \delta \left(s_{1:t} - s_{1:t}^{(l)} \right) \tag{26}$$

Then, $\tilde{x}_t$ is sampled from:

$$\tilde{x}_t \sim p \left(x_t | \tilde{x}_{t+1:T}, s_{1:t}^{(k)}, \tilde{s}_{t+1:T}, y_{1:T} \right) \approx \mathcal{N}_{SE(3)\times\mathbb{R}^6} \left(x_t; \tilde{\mu}_{t|T}, \tilde{P}_{t|T} \right) \tag{27}$$

where $\tilde{\mu}_{t|T}$ et $\tilde{P}_{t|T}$ are obtained using a Rauch-Tung-Striebel smoother on Lie groups (which is derived in the other pdf provided as supplementary material). The weights can be computed as:

$$w_{t|t+1}^{(l)} \propto w_t^{(l)} p \left(\tilde{s}_{t+1} | s_t^{(l)} \right) \mathcal{N}_{SE(3)\times\mathbb{R}^6} \left(\tilde{x}_{t+1}; \mu_{t+1|t}^{(l)}, P_{t+1|T}^{(l)} \right) \tag{28}$$

By iterating the previous 3 equations, a trajectory of $p\left(x_{1:T}, c_{1:T} | y_{1:T}\right)$ can be sampled.

B. Smoothing Algorithm

Algorithm 5 LG-RBPS

- For $i = 1, 2, ..., N_p$
  - Set $\tilde{s}_T = s_T^{(j)}$ with probability $w_T^{(j)}$
  - Set $\tilde{\mu}_{T|T} = \mu_{T|T}^{(j)}$, $\tilde{P}_{T|T} = P_{T|T}^{(j)}$
  - Draw $x_T^{(i)} \sim \mathcal{N}_{SE(3)\times\mathbb{R}^6} \left(x_T; \tilde{\mu}_{T|T}, \tilde{P}_{T|T} \right)$
  - For $t = T-1, T-2, ..., 1$
    * For $k = 1, 2, ..., N_p$
      · Set r with the value of $\mathcal{N}_{SE(3)\times\mathbb{R}^6} \left(X_{t+1}; \mu_{t+1|t}^{(k)}, P_{t+1|t}^{(k)} \right)$ evaluated in $X_{t+1} = x_{t+1}^{(i)}$
      · $w_{t|t+1}^{(k)} \propto w_t^{(k)} p\left(\tilde{s}_{t+1}\right) r$
    * EndFor
    * Set $j = k$ with probability $w_{t|t+1}^{(k)}$
    * $\tilde{s}_t = s_t^{(j)}$, $\tilde{\mu}_{t|t} = \mu_{t|t}^{(j)}$ and $\tilde{P}_{t|t} = P_{t|t}^{(j)}$
    * **LG-RTS Smoother**: Smooth $\tilde{\mu}_{t|t}$ and $\tilde{P}_{t|t}$ to get $\tilde{\mu}_{t|T}$ and $\tilde{P}_{t|T}$ using $\tilde{\mu}_{t+1|T}$ and $\tilde{P}_{t+1|T}$
    * Draw $x_t^{(i)} \sim \mathcal{N}_{SE(3)\times\mathbb{R}^6} \left(x_t; \tilde{\mu}_{t|T}, \tilde{P}_{t|T} \right)$
  - EndFor
- EndFor

Algorithm 6 LG-RTS smoother

- $L = P_{t|t} F^T \left(R + F P_{t|t} F^T \right)^{-1}$ where F has already been defined in Alg.2
- $r = \begin{bmatrix} log_{SE(3)}^{\vee} \left(\mu_{t+1|T}(C) \, \mu_{t+1|t}^{-1}(C) \right) \\ \mu_{t+1|T}(v) - \mu_{t+1|t}(v) \end{bmatrix}$
- $\delta = Lr$
- $\mu_{t|T}(C) = exp_{SE(3)}^{\wedge}\left(\delta(1:6)\right) \mu_{t|t}(C)$
- $\mu_{t|T}(v) = \delta(7:12) + \mu_{t|t}(v)$
- $P_{t|T} = \Phi_G(\delta) \left(P_{t|t} + L \left(\Phi_G^{-1}(r) \, P_{t+1|T} \Phi_G^{-T}(r) - P_{t+1|t} \right) L^T \right) \Phi_G^T(\delta)$ where Φ_G has already been defined in Alg.3

References

[1] Guillaume Bourmaud, Rémi Mégret, Audrey Giremus, and Yannick Berthoumieu. Discrete extended Kalman filter on Lie groups. In *Signal Processing Conference (EUSIPCO), 2013 Proceedings of the 21st European*, 2013.