Robot Mapping

Assignment: Sheet 5 The Unscented Transform

Authors:

Elias M Maharmeh

Robot Mapping lecture
University of Freiburg
Winter term 2021/22

Prof. Dr. Wolfram Burgard
Dr. Daniel Büscher
Dr. Lukas Luft
Kshitij Sirohi
Shengchao Yan

Sheet 5

Topic: The Unscented Transform

Due: December 6, 2021

Exercise: The Unscented Transform

Implement the Unscented Transform in Octave. The implementation should consist of two parts: computing the sigma points and recovering the transformed Gaussian.

(a) Implement the function in `compute_sigma_points.m`, which samples the $2n+1$ sigma points given the mean vector and covariance matrix. You should also compute the corresponding point weights $w_m^{[i]}$ and $w_c^{[i]}$ for $i = 0, \dots, 2n$.

(b) Implement the function in `recover_gaussian.m` to compute the mean and covariance of the resulting distribution given the transformed sigma points and their weights.

To support this task, we provide a small Octave framework on the course website. The tasks described above should be implemented inside the framework in the directory `octave` by completing the stubs. After implementing the missing parts, you can test your solution by running the main script. The program will produce a plot containing both the original and transformed distributions and save it in the `plots` directory.

The code provides three different functions describing transformations applied to the distribution. Test your implementation on each of them by uncommenting the corresponding parts in `transform.m`.

After completing the exercise, try other transformations by implementing them in `transform.m`. Moreover, you can change the parameters α and κ in `main.m` for computing λ and inspect how this affects the sampled sigma points.

Hint: To compute the square root of the covariance matrix in *Octave*, you can use the function `sqrtm`. Alternatively, you can compute the Cholesky decomposition using `chol`.

1 Computing the sigma points

The sigma points are calculated according to the following equation :

$$\mathbf{x}^1 = \mu \tag{1}$$

$$\mathbf{x}^i = \mu + L_{(i)} \tag{2}$$

$$\mathbf{x}^k = \mu - L_{(k)} \tag{3}$$

where:

- $\mathbf{x}^t$ is the t^{th} sigma point
- $i \in [2, 3, ..., n+1]$
- $k \in [n+2, n+3, ..., 2n+1]$
- $L_{(t)}$ is the t^{th} column of the square root of the $\sqrt{(n+\lambda)\Sigma}$
- $\mathbf{x}^t \in \mathcal{R}^n$, where n is the state dimension

In this step of sigma point generation, the weights also are generated according to the following equations.

$$\omega_m^1 = \frac{\lambda}{n+\lambda} \tag{4}$$

$$\omega_c^1 = \omega_m^1 + (1 - \alpha^2 + \beta) \tag{5}$$

$$\omega_m^i = \omega_c^i = \frac{1}{2(n+\lambda)} \tag{6}$$

$$\omega_m = [\omega_m^1 \omega_m^2 \dots \omega_m^{2n+1}] \tag{7}$$

$$\omega_c = [\omega_c^1 \omega_c^2 \dots \omega_c^{2n+1}] \tag{8}$$

where :

- ω_m, ω_c are the mean and covarience weights respectively.
- λ, α, β are parameters for controlling the locations of the sigma points.

In order to compute the sigma points, the following manipulation is performed.

Let SP be the augmented matrix of sigma points.

$$SP = \begin{bmatrix} \mathbf{x}^1 & \mathbf{x}^2 & \dots & \mathbf{x}^{2n+1} \end{bmatrix} \tag{9}$$

The dimension of SP belongs to the space of $\mathcal{R}^{n*2n+1}$, where each column represents a sigma point with the same dimension of the state.

Using the *Octave* convention, the sigma points are generated according to the follwoing steps :

- Compute the square root of the scaled covarience matrix $L = \sqrt{(n+\lambda)\Sigma}$, which is a lower triangle matrix of the same dimension of the covarience matrix, i.e $\mathcal{R}^{n,n}$. Thus each column of the n columns of the L matrix is of a dimension consistant with the state dimension
- Initialize the matrix SP

$$SP = \begin{bmatrix} \mu & [\mu \;\; \mu \dots \mu]_{n,n} & [\mu \;\; \mu \dots \mu]_{n,n} \end{bmatrix}_{n,2n+1} \tag{10}$$

- Then, perform the following block matrix operation, in order to get the sigma points:

$$SP = \begin{bmatrix} \mu & [\mu\ \ \mu \ldots \mu]_{n,n} + L_{n,n} & [\mu\ \ \mu \ldots \mu]_{n,n} - L_{n,n} \end{bmatrix}_{n,2n+1} \tag{11}$$

  This could be done in *Octave* using the slicing operation according to the following expression.

$$SP(:, 2 : n+1) = SP(:, 2 : n+1) + L; \tag{12}$$

$$SP(:, n+2 : 2n+1) = SP(:, n+2 : 2n+1) - L; \tag{13}$$

- Calculate the weight according to the equations from 4 to 8. This gives us two weighting vectors ω_m, ω_c of dimension $2n+1$

2 Recovering the Gaussian pdf

Using the following two relations between the mean and covarience to estimate the resulting Gaussian distribution from the sigma points.

$$\tilde{\mu} = \sum_{i=1}^{2n+1} \omega_m^i g(\mathbf{x}^i) \tag{14}$$

$$\tilde{\Sigma} = \sum_{i=1}^{2n+1} \omega_c^i (g(\mathbf{x}^i) - \tilde{\mu})(g(\mathbf{x}^i) - \tilde{\mu})^T \tag{15}$$

Where $g(.)$ is the nonlinear mapping function.

The implementation of equations 14,15 using the SP matrix and the two weighting vectors ω_m, ω_c is described below using the *Octave* notation for matrix multiplication and the dot element-wise operation.

$$\tilde{\Sigma} = sum(\omega_m . * SP, 2); \tag{16}$$

For the calculation of the covarience matrix, the following steps are performed:

$$\tilde{SP} = [SP - \tilde{\mu}] \tag{17}$$

$$\tilde{SP}_w = [\omega_c . * \tilde{SP}] \tag{18}$$

$$\tilde{\Sigma} = [\tilde{SP}_w][\tilde{SP}]^T \tag{19}$$