Robot Navigation and Localization Project 3

Jeremy Lu

N13492085 (xl2810)

2020 April

Introduction

The goal is to estimate pose of robot using given data using Unscented Kalman Filter. In first part, we are using pose and position of angle of the robot. Second part, we use linear and angular velocity from project 2 to estimate.

Part 1 - Approach

Prediction

Step 1

Convert states and covariance into Augmented states and covariance

$$\boldsymbol{\mu}_{aug,t-1} = \begin{pmatrix} \boldsymbol{\mu}_{t-1} \\ \mathbf{0} \end{pmatrix} \qquad \boldsymbol{P}_{aug} = \begin{pmatrix} \boldsymbol{\Sigma}_{t-1} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{Q}_t \end{pmatrix}$$

Using the augmented states to computer sigma points,

$$\mathcal{X}^{(0)} = \boldsymbol{\mu}_{aug}, \qquad \mathcal{X}^{(i)} = \boldsymbol{\mu}_{aug} \pm \sqrt{n' + \lambda'}\left[\sqrt{\boldsymbol{\Sigma}_{aug}}\right]_i$$

Note that the size of augmented state is n = 15+12 = 27, there are 2 * n + 1 sigma points with size of each one is 27 * 1.

Step 2

Propagate Sigma points through linear function f(x, u, n):

$$\begin{bmatrix} \mathbf{x}_3 \\ G(\mathbf{x}_2)^{-1}(\omega_m - \mathbf{x}_4 - \mathbf{n}_g) \\ \mathbf{g} + R(\mathbf{x}_2)(\mathbf{a}_m - \mathbf{x}_5 - \mathbf{n}_a) \\ \mathbf{n}_{bg} \\ \mathbf{n}_{ba} \end{bmatrix} = f(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{n})$$

Specifically in MatLab, f = (X_sigma(1:15,i), angVel, acc,   X_sigma(16:27,i)),

The ouput from this step is to get X matrix which has size of 15 * 55.

Step 3

Then we computer mean and covariance:

$$\bar{\boldsymbol{\mu}}_t = \textstyle\sum_{i=0}^{2n'} W_i^{(m)} \chi_t^{(i)} \qquad \bar{\boldsymbol{\Sigma}}_t = \textstyle\sum_{i=0}^{2n'} W_i^{(c)\prime} \left(\chi_t^{(i)} - \bar{\boldsymbol{\mu}}_t\right)\left(\chi_t^{(i)} - \bar{\boldsymbol{\mu}}_t\right)^T$$

Update

Since the for this part the linear model applies, we calculate update as below:

$$\boldsymbol{z}_t = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{\eta}$$

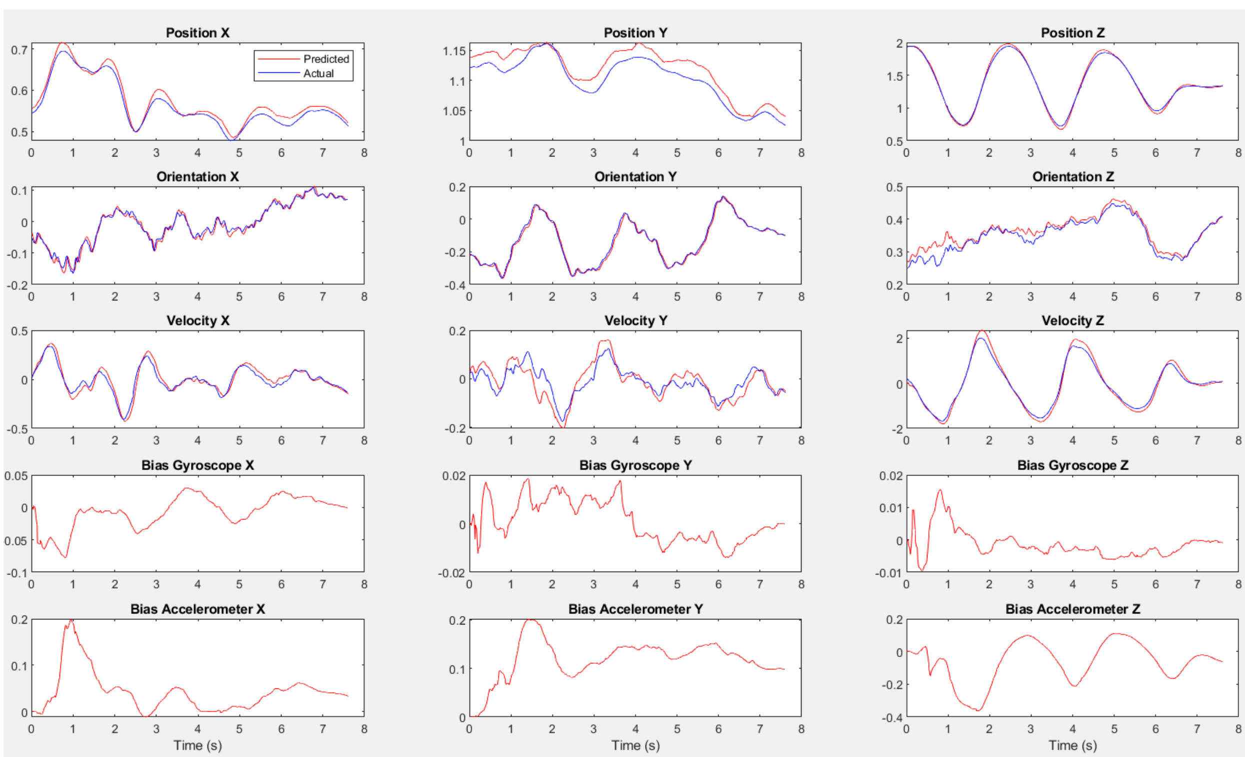
$$\boldsymbol{\mu}_t = \bar{\boldsymbol{\mu}}_t + \boldsymbol{K}_t(\boldsymbol{z}_t - C\,\bar{\boldsymbol{\mu}}_t)$$

$$\boldsymbol{\Sigma}_t = \bar{\boldsymbol{\Sigma}}_t - \boldsymbol{K}_t\,\boldsymbol{C}\,\bar{\boldsymbol{\Sigma}}_t$$

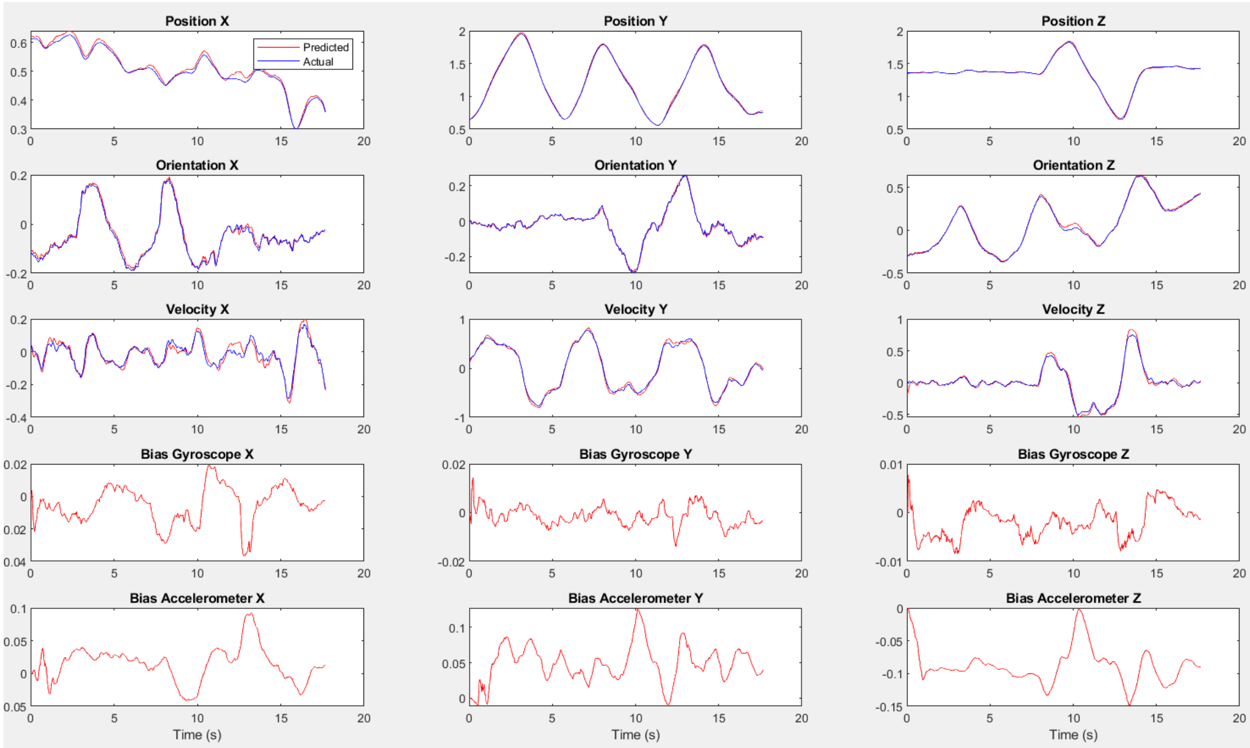
$$\boldsymbol{K}_t = \bar{\boldsymbol{\Sigma}}_t\,\boldsymbol{C}^T\,(\boldsymbol{C}\,\bar{\boldsymbol{\Sigma}}_t\,\boldsymbol{C}^T + \boldsymbol{R})^{-1}$$

Note that Cx is simply takes the states that we use to compare with measurement.

Part 1 – Result



Part 1 Dataset 4



Part 1 dataset 1

Comment

Both dataset result are close to vicon measurement, dataset 1 fits better than dataset 4 but both result are close enough.

Part 2 Approach

Prediction

Here we use the same prediction method as the part 1.

Update

This part we use additive noise for our estimation, we also use angular and linear velocity calculated from project 2 for estimation.

Step 1

We want to calculate sigma points , note that we DO NOT use augment state here, so:

$$\chi_t^{(0)} = \bar{\boldsymbol{\mu}}_t \qquad \chi_t^{(i)} = \bar{\boldsymbol{\mu}}_t \pm \sqrt{n+\lambda}\left[\sqrt{\bar{\boldsymbol{\Sigma}}_t}\right]_i$$

This time n is just the size of covariance matrix which is 15. And we have 31 sigma points.

Step 2

Then we propagate sigma point through function g(x)

$$\boldsymbol{z}_t = {}^{\boldsymbol{C}}\boldsymbol{v}_{\boldsymbol{C}}^{\boldsymbol{W}} = \boldsymbol{g}\left(\boldsymbol{x}_{2,}, \boldsymbol{x}_3, {}^{\boldsymbol{B}}\boldsymbol{\omega}_{\boldsymbol{B}}^{\boldsymbol{W}}, \boldsymbol{\eta}\right)$$

X2 is the angel information calculated from prediction stage

X3 is the linear velocity of body to world in world frame (bww)

B_omega_WB is the angular velocity from body to world in body frame which is the project 2 information.

Using above, we calculate predicted linear velocity:

$${}^{C}\dot{P}_{C}^{W} = R_{B}^{C}\,{}^{B}\dot{P}_{B}^{W} - R_{B}^{C} S(r_{BC}^{B})\,{}^{B}\dot{w}_{B}^{W}$$

We do not know angular velocity of body to world in body frame, the project 2 data only gives us angular velocity of body to world in world frame, but we can transform that using a rotation matrix calculated using angle information:

$${}^{B}\dot{w}_{B}^{W} = R_{W}^{B}\,{}^{W}\dot{w}_{B}^{W}$$

Output is sigma points.

Step 3

Then we calculate mean and covariance:

$$\boldsymbol{z}_{\mu,t} = \sum_{i=0}^{2n} W_i^{(m)} Z_t^{(i)}$$

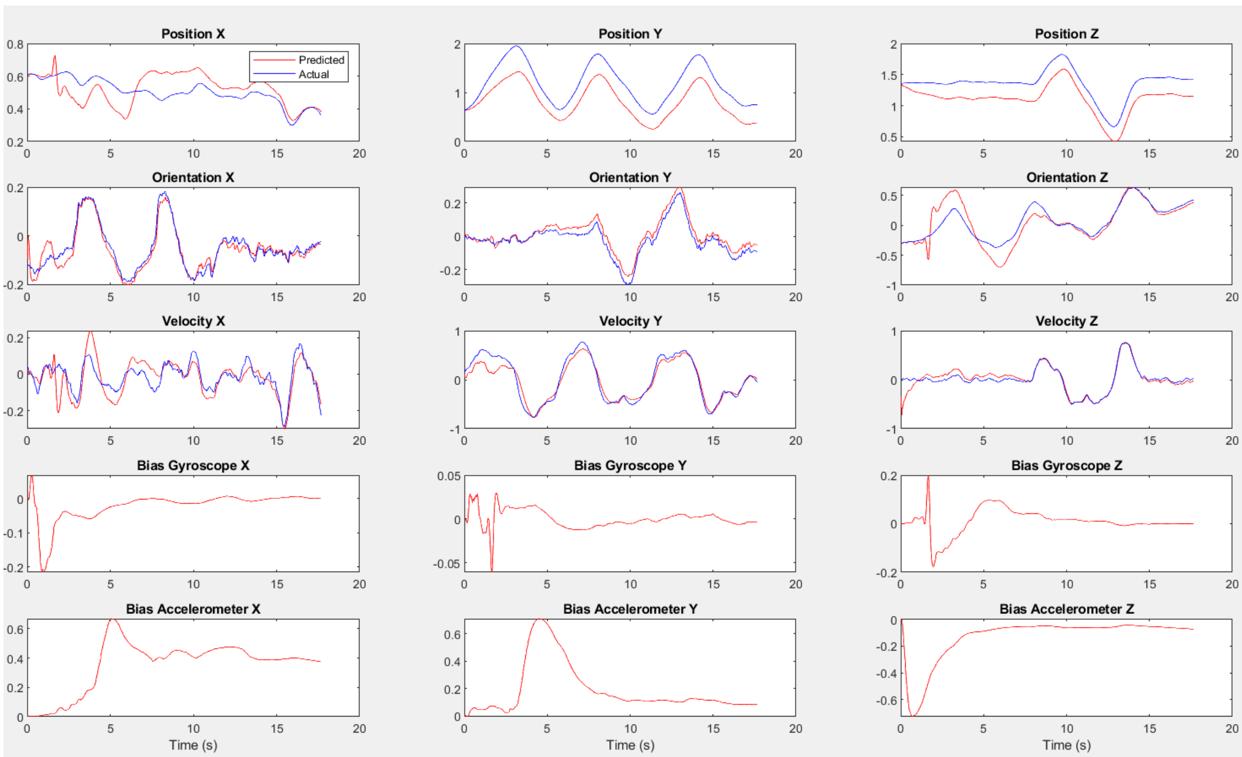
Update

$$\boldsymbol{C}_t = \sum_{i=0}^{2n} W_i^{(c)} \left(\chi_t^{(i)} - \bar{\boldsymbol{\mu}}_t\right)\left(Z_t^{(i)} - \boldsymbol{z}_{\mu k}\right)^T \qquad \boldsymbol{S}_t = \sum_{i=0}^{2n} W_i^{(c)} \left(Z_t^{(i)} - \boldsymbol{z}_{\mu k}\right)\left(Z_t^{(i)} - \boldsymbol{z}_{\mu,t}\right)^T + \boldsymbol{R}_t$$

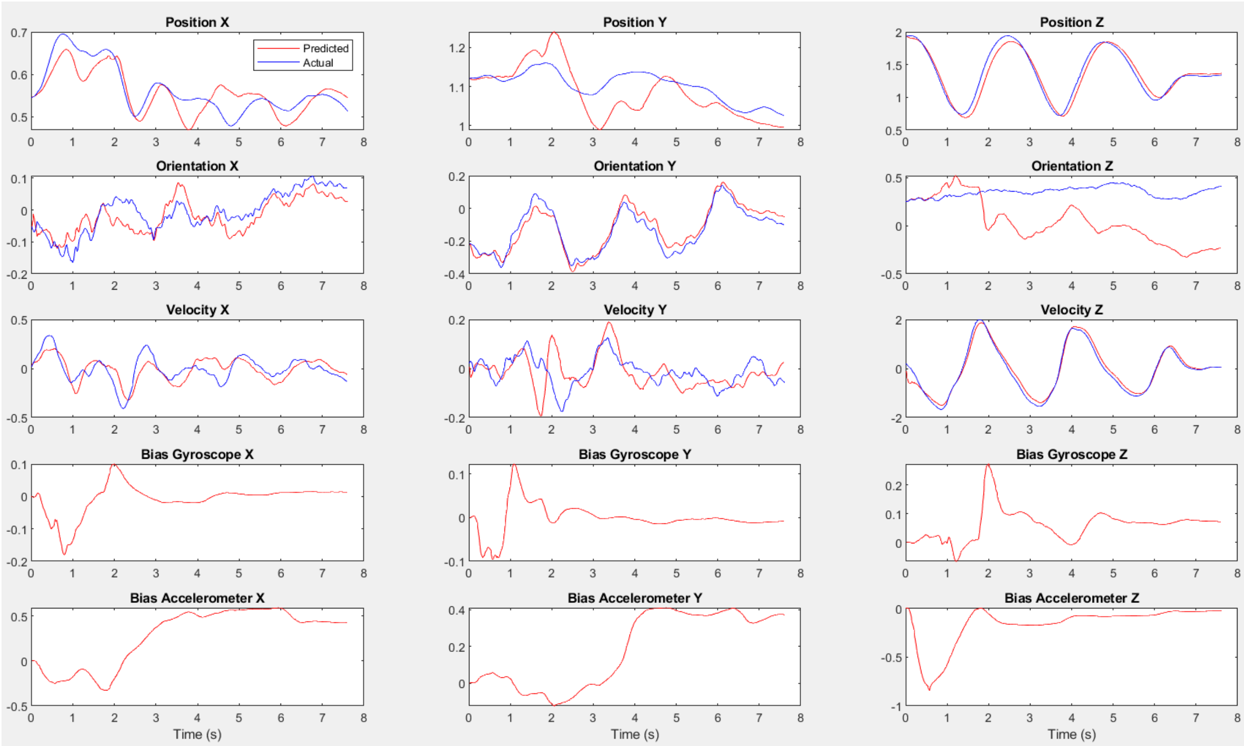
And finally, we compute kalman gain and updated states and covariance:

- $\boldsymbol{\mu}_t = \overline{\boldsymbol{\mu}}_t + \boldsymbol{K}_t\,(\boldsymbol{z}_t - \boldsymbol{z}_{\mu,t})$
- $\boldsymbol{\Sigma}_t = \overline{\boldsymbol{\Sigma}}_t - \boldsymbol{K}_t\,\boldsymbol{S}_t\boldsymbol{K}_t{}^T$
- $\boldsymbol{K}_t = \boldsymbol{C}_t\,\boldsymbol{S}_t{}^{-1}$

Part 2 result



Part2 Dataset1



Part2 Dataset4

Comment

From above result we can see that dataset 1 orientation and velocity fits are better than position fits. For dataset 4, the orientation Z are not fitting well. Generally the result of part 2 are less precise than part1, the reason might be the project 2 data are not very accurate which affect the update section.