Solution to analysis in Home Assignment 3

Nicholas Granlund, nicgra@student.chalmers.se

Analysis

In this report I will present my independent analysis of the questions related to home assignment 1. I have discussed the solution with Martin Lamm, Louise Olsson and NONE and I swear that the analysis written here are my own.

1 Approximation of mean and covariance

Task a)

To approximate the mean and covariance of $\mathbf{y}$ we simply generate a non-linear measurements sequence using $\mathbf{h}(\mathbf{x})$ and $\mathbf{r}$, then we sample $\mathbf{y}$ to get an approximation of the mean and covariance. We use $N = 10000$ samples to get an accurate-enough approximation. By doing so we get the samples plotted in Figure 1.1 for the different state-densities.

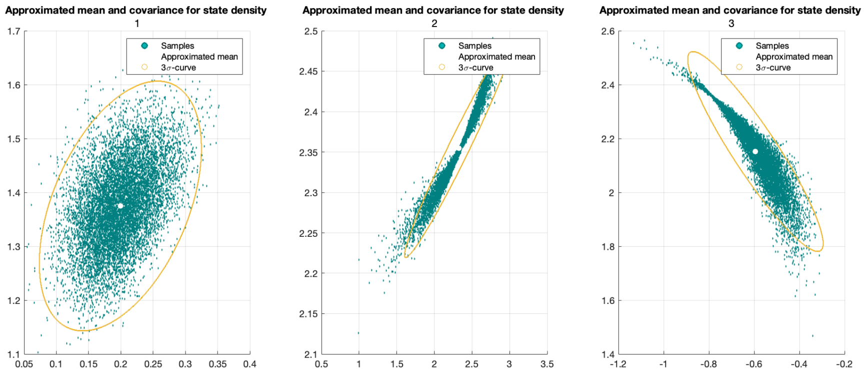

Figure 1.1: Samples used to approximate mean and covariance for the three different state-densities.

The approximated densities was as follows:

$$p_1(\mathbf{y}) \approx \mathcal{N}\left(\mathbf{y}; \begin{bmatrix} 0.1974 \\ 1.3734 \end{bmatrix}, \begin{bmatrix} 0.0017 & 0.0015 \\ 0.0015 & 0.0060 \end{bmatrix}\right) \tag{1}$$

$$p_2(\mathbf{y}) \approx \mathcal{N}\left(\mathbf{y}; \begin{bmatrix} 2.3264 \\ 2.3549 \end{bmatrix}, \begin{bmatrix} 0.0562 & 0.0104 \\ 0.0104 & 0.0020 \end{bmatrix}\right) \tag{2}$$

$$p_3(\mathbf{y}) \approx \mathcal{N}\left(\mathbf{y}; \begin{bmatrix} -0.5947 \\ 2.1520 \end{bmatrix}, \begin{bmatrix} 0.0099 & -0.0113 \\ -0.0113 & 0.0152 \end{bmatrix}\right) \tag{3}$$

Task b)

Now by instead computing the approximated mean and covariance analytically with EKF, UKF or CKF, we do not need to sample the distribution. We will use EKF and CKF for this analysis.

The computations are very similar to the prediction step in the non-linear Kalman filter. But instead of predicting what the next state will be ($x_k = f(x_{k-1}) + Bu_{k-1}$), we predict what the measurement will be ($y_k = h(x_k) + r$). By performing the computations used in the prediction step from lecture 6, we obtain the following approximations for the mean and covariance:

EKF:

$$p_{1,EKF}(\mathbf{y}) \approx \mathcal{N}\left(\mathbf{y}; \begin{bmatrix} 0.1974 \\ 1.3734 \end{bmatrix}, \begin{bmatrix} 0.0017 & 0.0015 \\ 0.0015 & 0.0060 \end{bmatrix}\right)$$

$$p_{2,EKF}(\mathbf{y}) \approx \mathcal{N}\left(\mathbf{y}; \begin{bmatrix} 2.3562 \\ 2.3562 \end{bmatrix}, \begin{bmatrix} 0.0500 & 0.0100 \\ 0.0100 & 0.0020 \end{bmatrix}\right)$$

$$p_{3,EKF}(\mathbf{y}) \approx \mathcal{N}\left(\mathbf{y}; \begin{bmatrix} -0.5880 \\ 2.1588 \end{bmatrix}, \begin{bmatrix} 0.0092 & -0.0111 \\ -0.0111 & 0.0148 \end{bmatrix}\right)$$

UKF:

$$p_{1,UKF}(\mathbf{y}) \approx \mathcal{N}\left(\mathbf{y}; \begin{bmatrix} 0.1983 \\ 1.3743 \end{bmatrix}, \begin{bmatrix} 0.0017 & 0.0015 \\ 0.0015 & 0.0059 \end{bmatrix}\right)$$

$$p_{2,UKF}(\mathbf{y}) \approx \mathcal{N}\left(\mathbf{y}; \begin{bmatrix} 2.3269 \\ 2.3550 \end{bmatrix}, \begin{bmatrix} 0.0600 & 0.0108 \\ 0.0108 & 0.0020 \end{bmatrix}\right)$$

$$p_{3,UKF}(\mathbf{y}) \approx \mathcal{N}\left(\mathbf{y}; \begin{bmatrix} -0.5949 \\ 2.1524 \end{bmatrix}, \begin{bmatrix} 0.0099 & -0.0112 \\ -0.0112 & 0.0151 \end{bmatrix}\right)$$

CKF:

$$p_{1,CKF}(\mathbf{y}) \approx \mathcal{N}\left(\mathbf{y}; \begin{bmatrix} 0.1983 \\ 1.3743 \end{bmatrix}, \begin{bmatrix} 0.0017 & 0.0015 \\ 0.0015 & 0.0059 \end{bmatrix}\right)$$

$$p_{2,CKF}(\mathbf{y}) \approx \mathcal{N}\left(\mathbf{y}; \begin{bmatrix} 2.3265 \\ 2.3550 \end{bmatrix}, \begin{bmatrix} 0.0556 & 0.0105 \\ 0.0105 & 0.0020 \end{bmatrix}\right)$$

$$p_{3,CKF}(\mathbf{y}) \approx \mathcal{N}\left(\mathbf{y}; \begin{bmatrix} -0.5948 \\ 2.1523 \end{bmatrix}, \begin{bmatrix} 0.0097 & -0.0112 \\ -0.0112 & 0.0150 \end{bmatrix}\right)$$

Task c)

The plots for the approximated mean/covariance for the different state-densities can be seen below in Figures 1.2-1.4.

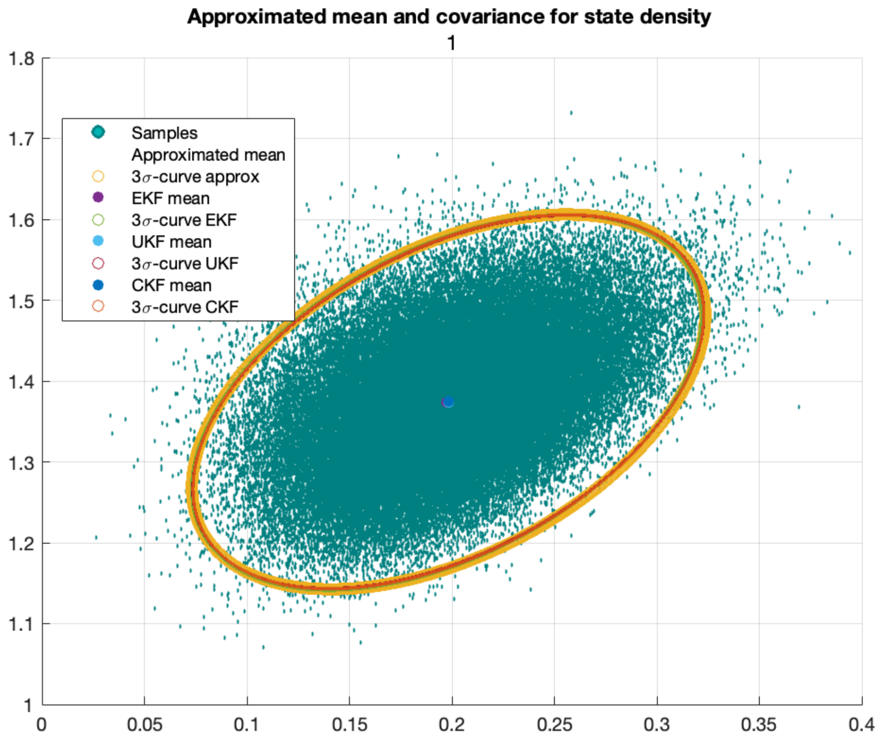

Figure 1.2: Samples used to approximate mean and covariance for the state-density 1. Plotted alongside the analytically computed mean and covariance for EKF, UKF and CKF

The analytically computed mean and covariance for EKF, UKF and CKF are more ore less identical to the sampled mean and covariance. The 3σ-levels coincide rather similarly. Since the distribution of $\mathbf{y}$ is somewhat linear given the first state-density, the approximations are similar and describe the distribution well.

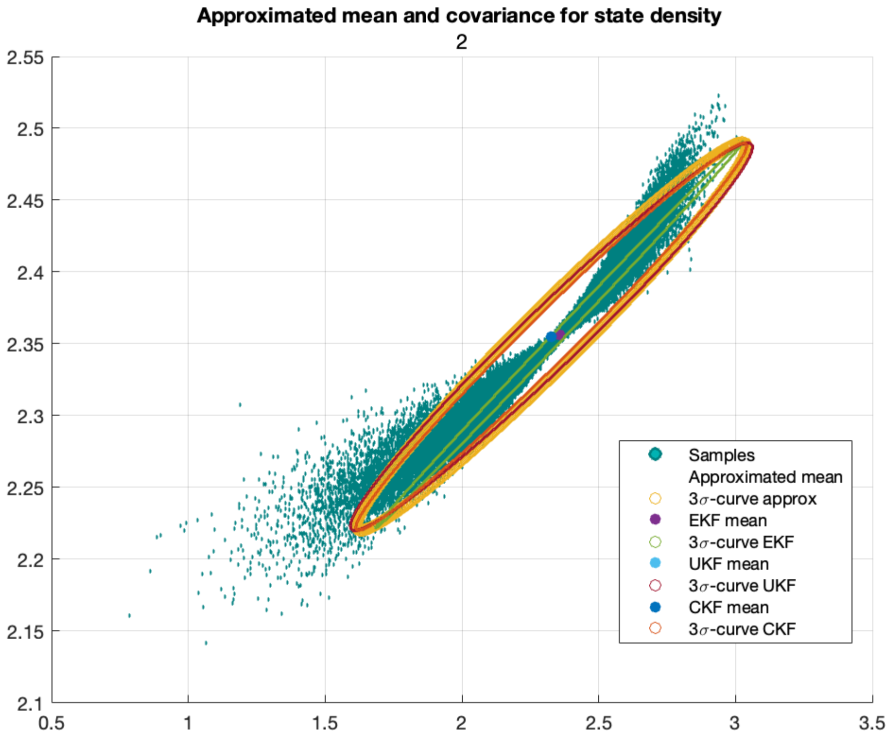

Figure 1.3: Samples used to approximate mean and covariance for the state-density 2. Plotted alongside the analytically computed mean and covariance for EKF, UKF and CKF

As seen by the plot, the distribution of $\mathbf{y}$ is less linear compared to the previous distribution. Given this state-density, the distribution follows a somewhat increasing exponential graph/form.

The EKF approximation is the one with the narrowest covariance where it falls somewhat short of the actual covariance. The accuracy difference of UKF and CKF is negligable. It should be noted however that CKF did a worse estimation of the mean, but got a better estimate for the covariance compared to UKF.

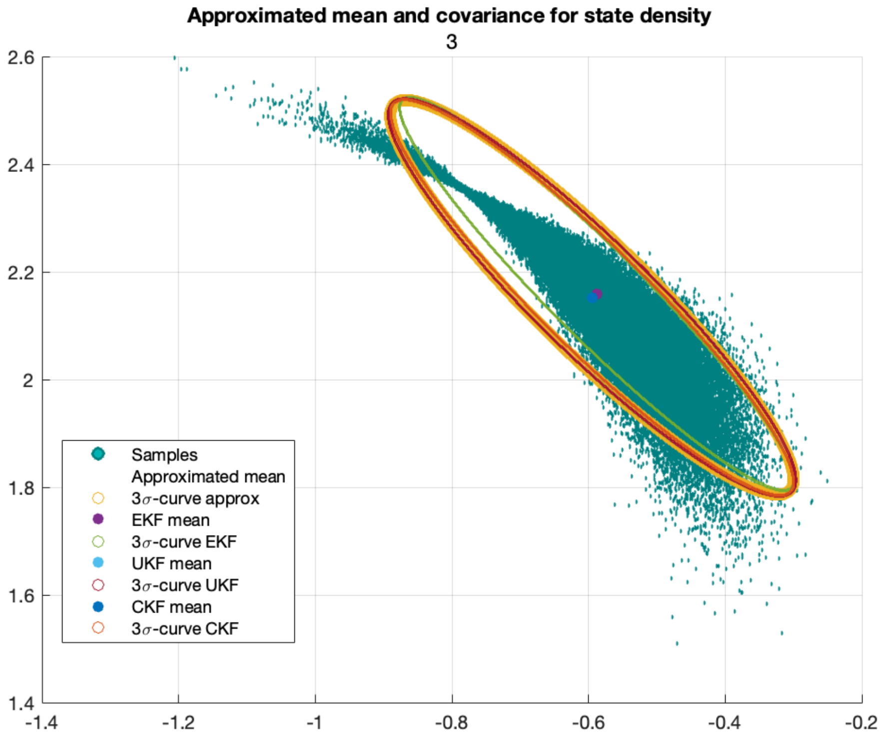

Figure 1.4: Samples used to approximate mean and covariance for the state-density 3. Plotted alongside the analytically computed mean and covariance for EKF, UKF and CKF

The third distribution is the one that is the most non-linear of them all which follows a decaying exponential graph/form. For this distribution, the EKF approximated the worst mean of them all where the first component x_2 was estimated as too small. EKF, UKF and CKF did a comparable estimation of the covariance where neither was better than the other.

Task d)

The differences worth mentioning has been brought up in **task c)**. The cases where the EKF performs worst, is in the cases where the distribution of $\mathbf{y}$ is highly non-linear. This is due to the fact that the EKF works on linearization whereas UKF and CKF does not. This is prominent in the first distribution where $p(\mathbf{y})$ can be described accurately with a gaussian. The EKF performs worse for the two other distributions where they are more non-linear.

The trade-off here is between accuracy and computational complexity. the EKF performs les computations than UKF or CKF and may therefor be more desireable for real-time applications, however as previously mentioned EKF may be innacurate if the system is highly non-linear and/or the process consists of large noise. In the cases where the system dynamics is highly-non linear, UKF and CKF would be preferable, even though they are more computationally demanding.

Since this system is non-linear and is corrupted by a somewhat large noise, I would choose to use the CKF as my filter.

2 Non-linear Kalman filtering

Task a)

By generating a state sequence $\mathbf{x_0}, \mathbf{x_1}, \mathbf{x_2} ... \mathbf{x_{100}}$, and a corresponding measurement sequence $\mathbf{y_1}, \mathbf{y_2} ... \mathbf{y_{100}}$ we are able to filter the position using the different non-linear Kalman filters. We use the first disturbance model where the noise is determined by Case 1 from Table 1 in the assignment description. This is the case which is corrupted by the largest noise. By doing this we obtain the result seen below in Figure 2.1.

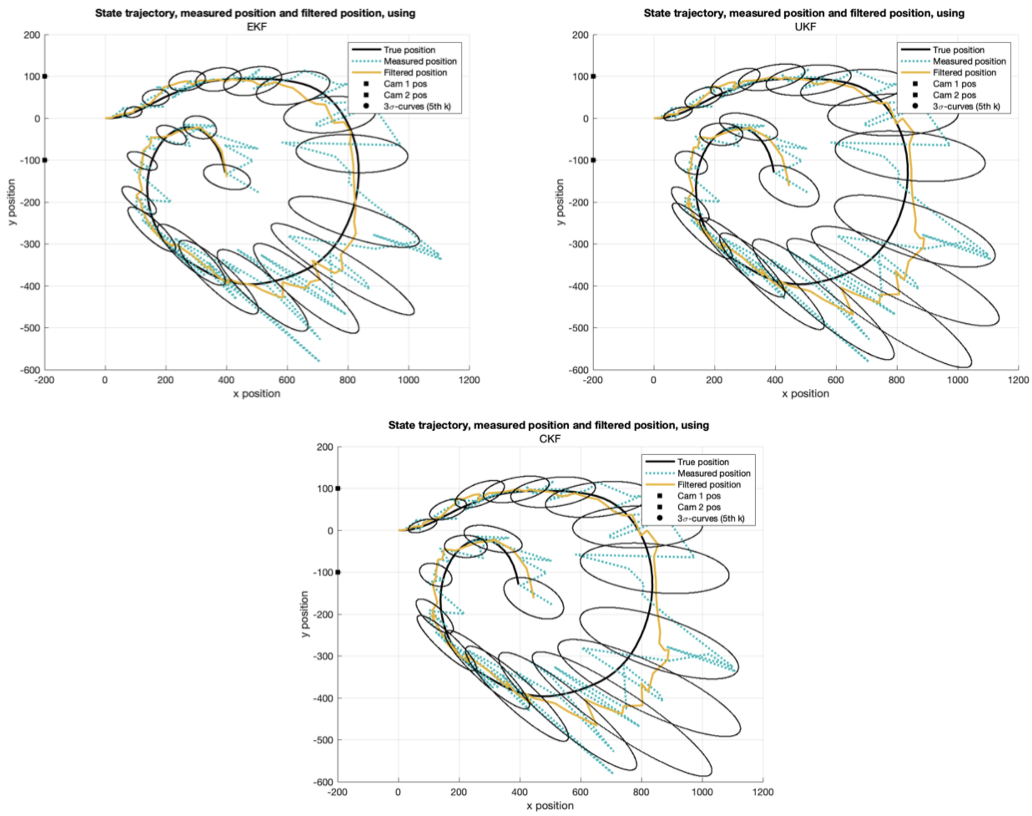

Figure 2.1: Generated state sequence, measurements, the filtered position, the sensor positions and 3σ-contours every fifth time instant. This is done for EKF, UKF and CKF respectively. The filters have been working on the exact same states $\mathbf{x}$ and the same measurements $\mathbf{y}$

A short summary of the plots above is that the EKF manages to track the true trajectory more accurately than the sigma-points based methods. This is rather prominent in the bottom right part of the plots where the estimated position deviates greatly from the actual position (yellow graph compared to black graph).

Task b)

Now we repeat the steps done in **Task a)** for the other disturbance cases. Same as before, the generated states and measurements are the same for all the different filters. For disturbance Case 2 we get te following result:

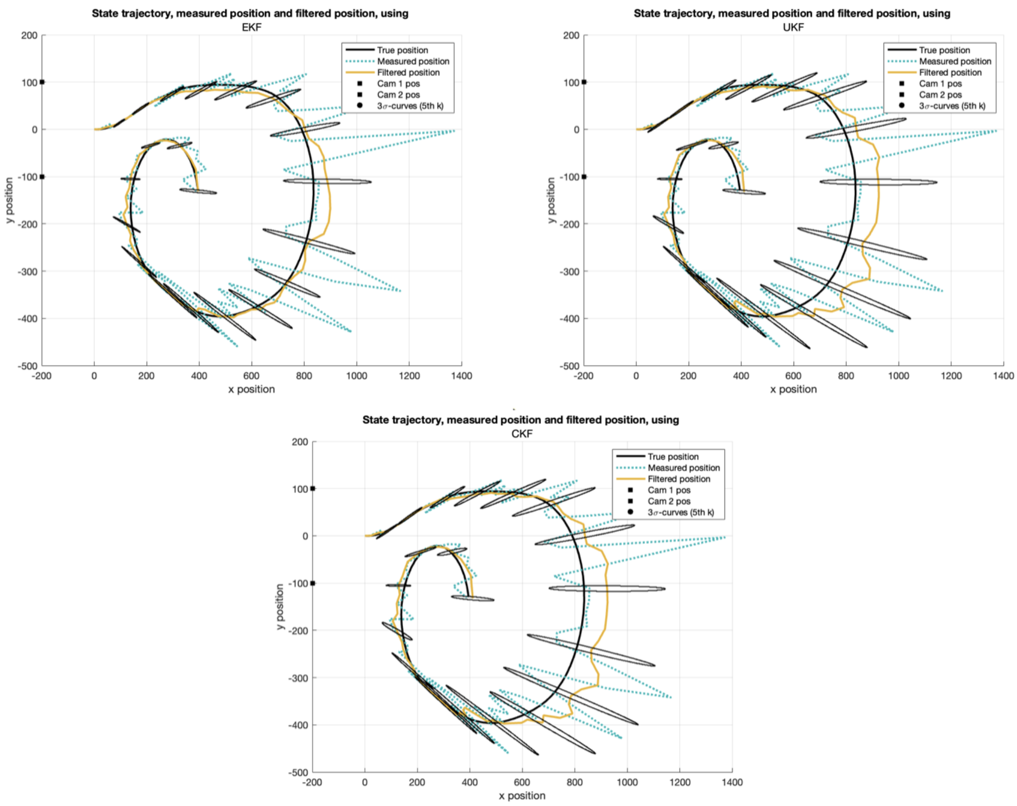

Figure 2.2: Generated state sequence, measurements, the filtered position, the sensor positions and 3σ-contours every fifth time instant. This is done for EKF, UKF and CKF respectively. The filters have been working on the exact same states $\mathbf{x}$ and the same measurements $\mathbf{y}$

And for disturbance case 3 we get:

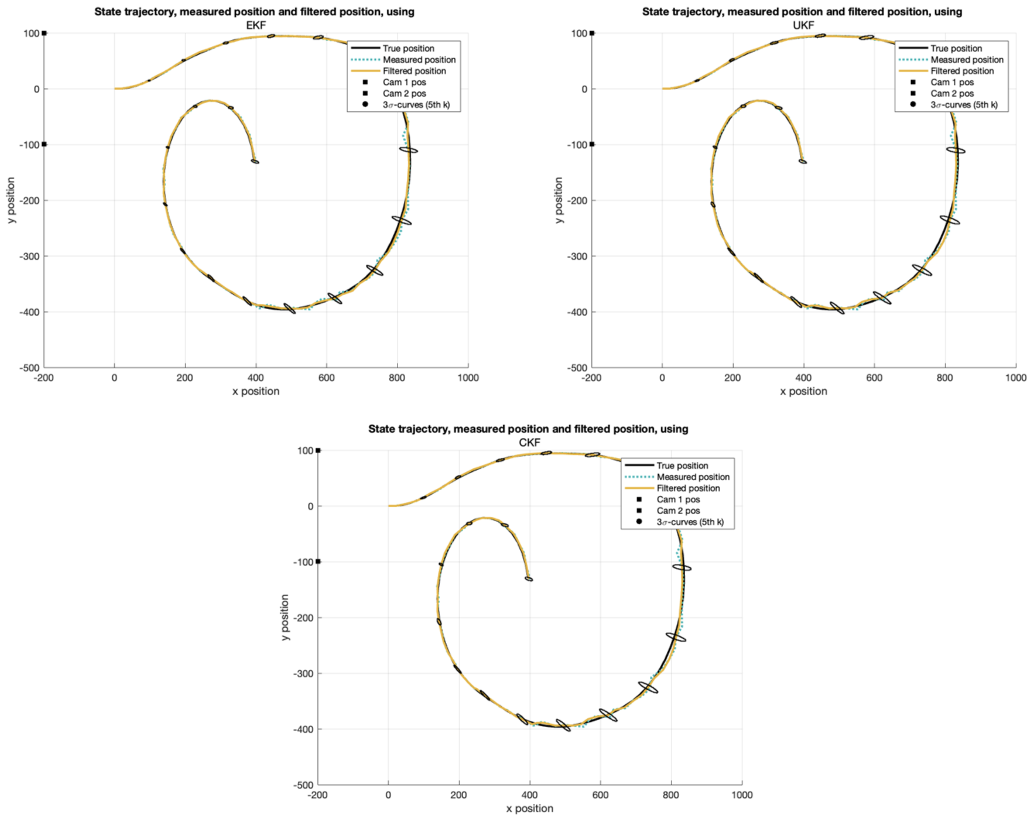

Figure 2.3: Generated state sequence, measurements, the filtered position, the sensor positions and 3σ-contours every fifth time instant. This is done for EKF, UKF and CKF respectively. The filters have been working on the exact same states $\mathbf{x}$ and the same measurements $\mathbf{y}$

One can see that when the noise levels are low (Case 3), then the performance of the filters are comparable. Also when the noise levels decrease both in process and measurements, the certainty in the estimation is higher, which is an expected result. However, in Case 2 the two sensors are affected by different levels of noise. This is rather evident in the estimation error covariance where it is highly skewed along the x position. The intuitive explanation for this is that at this noise level, the sensors are having trouble accurately track the depth of the position and thus introducing a higher uncertainty. (the 3σ-curves are 'pointed' towards the sensors). This explains why the error in estimated position is the largest when the trajectory is perpendicular to the sensors. Moreover, it is also visible how the covariance is skewed towards the sensor with the least noise. This is also expected since the filter will trust that sensor more than the more noise corrupted one. (the lowest placed sensor has the lowest noise levels for case 2)

For case 1 and case 3, where the sensors are corrupted by the same amount of noise, the estimation error covariance is more even for both positional variables. And the skewedness is due to the process disturbances.

Task c)

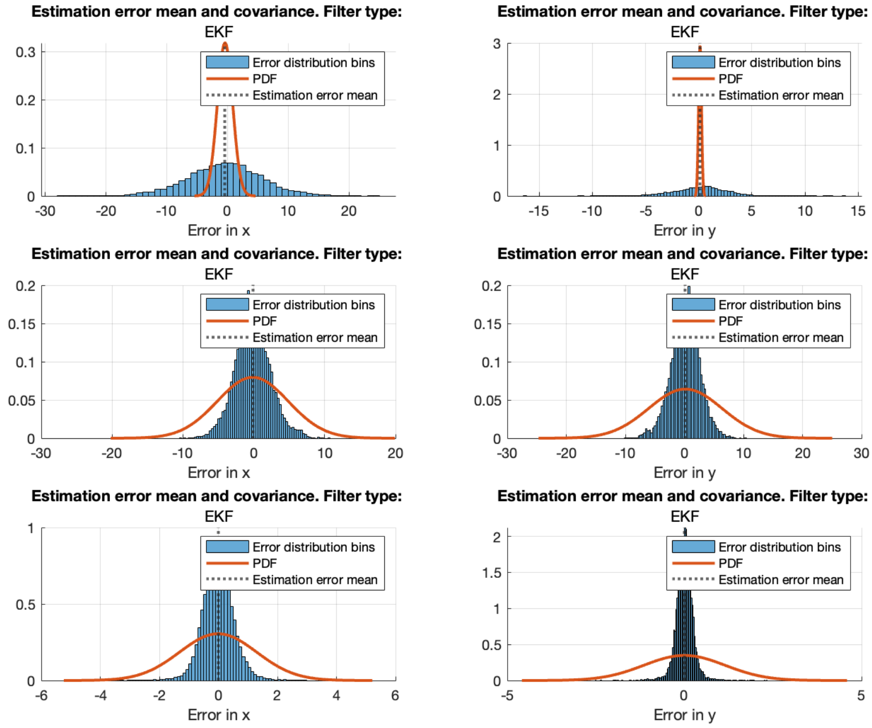

Figure 2.4: EKF Estimation error for disturbance case 1,2 and 3

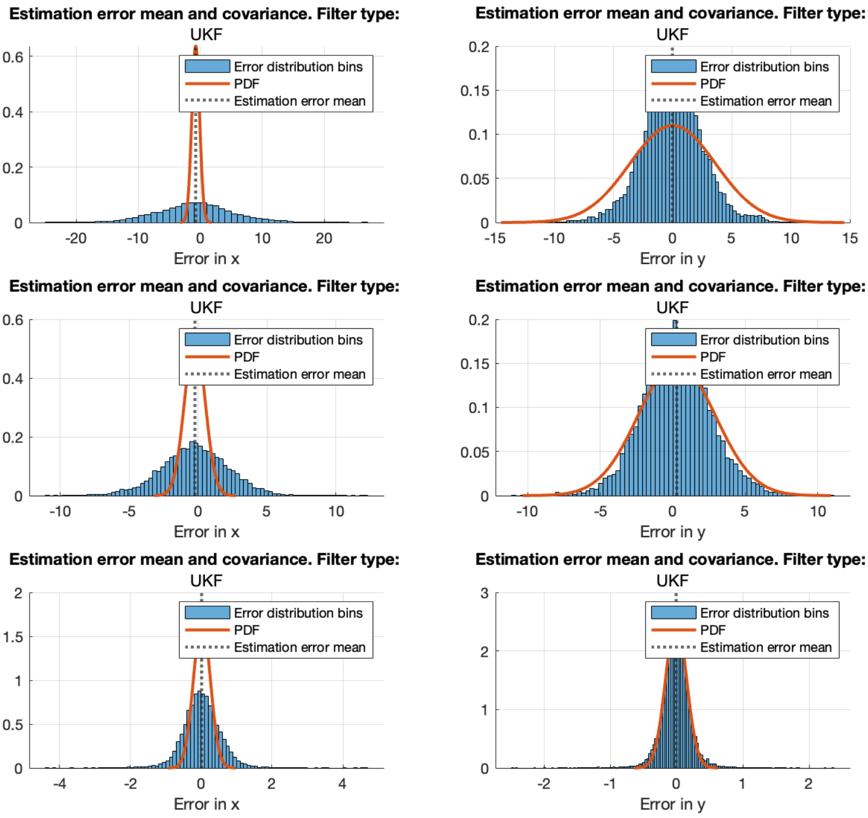

Figure 2.5: UKF Estimation error for disturbance case 1,2 and 3

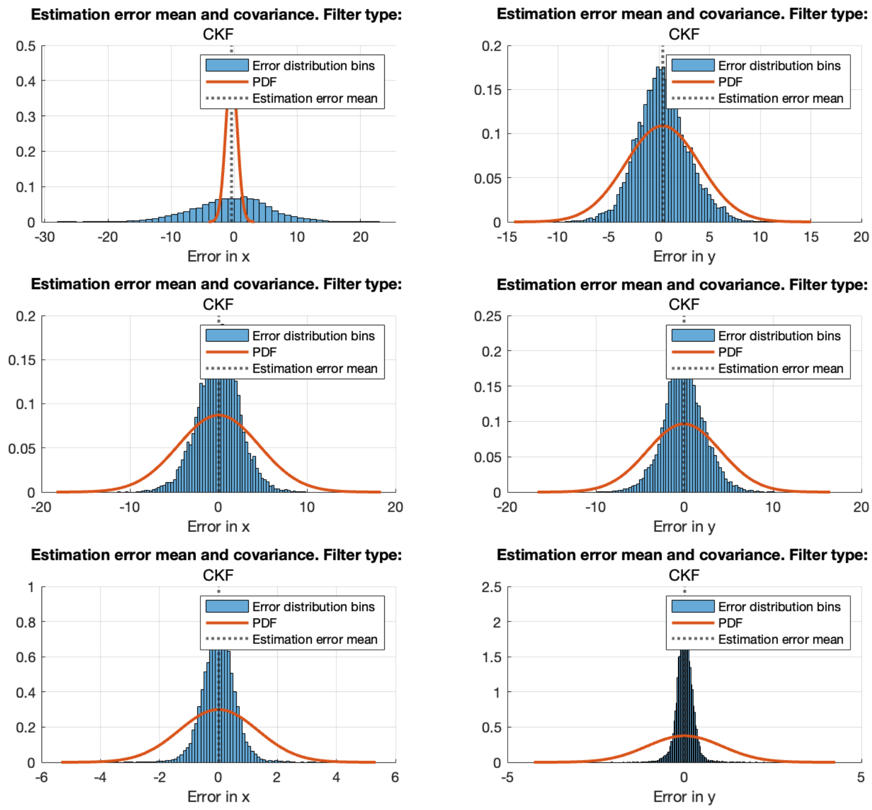

Figure 2.6: CKF Estimation error for disturbance case 1,2 and 3

3 Tuning non-linear filters

Task a)

By implementing the model given in the assignment description and setting the base noise levels as $\sigma_v = 1$ and $\sigma_\omega = \pi/180$ we can get an estimation of the position and filter the measurements with my favourite non-linear Kalman filter, EKF. The results can be seen in Figure 3.1 below.

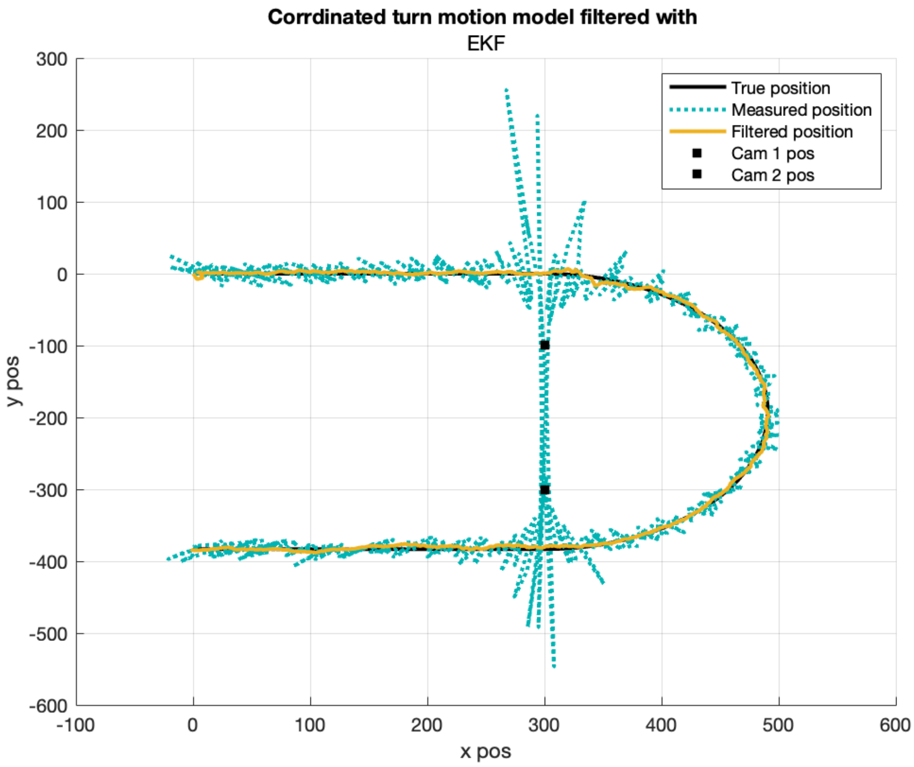

Figure 3.1: Coordinated turn model filtered with EKF with the given initial covariances σ_v and σ_ω

Similar to question 2, when the position is measured close to tangential to the sensors, the noise in the measurement is more prominent. Using the given covariances for the process disturbances, the EKF can filter the measurements rather well and the filtered position follows the true position to good extent. The estimated position deviates after passing the tangent to the sensor positions.

What happens when you make the process noise very large?

When the process noise becomes very large, the Kalman filter will listen more to the measurements. The Kalman filter essentially works on the quotient between Q/R and decides if it should value the prediction or the measurements greater than the other. In the case where the predicion is expected to be corrupted by alot of noise, then the measurement would be more accurate. In that case we will get an estimated position which is more noisy.

If we increase the noise for σ_v, then the filter will better track straight trajectories, and if we increase the noise for σ_ω, the filter will be worse att tracking straight trajectories, but better in tracking the position when it performs the coordinated turn.

What happens when you make the process noise very small?

The oposite will occur. The filter will expect the prediction to be less corrupted by noise, and will therefor weigh the prediction more than the measurement. This will in return impact the turn-rate tracking since the prediction will imply a straight trajectory.

The filter will not expect the turn to occur and will thus lag behind in tracking it. This can be show in Figure 3.2 below where the noises have been decreased by a factor of 1000 from their initial values.

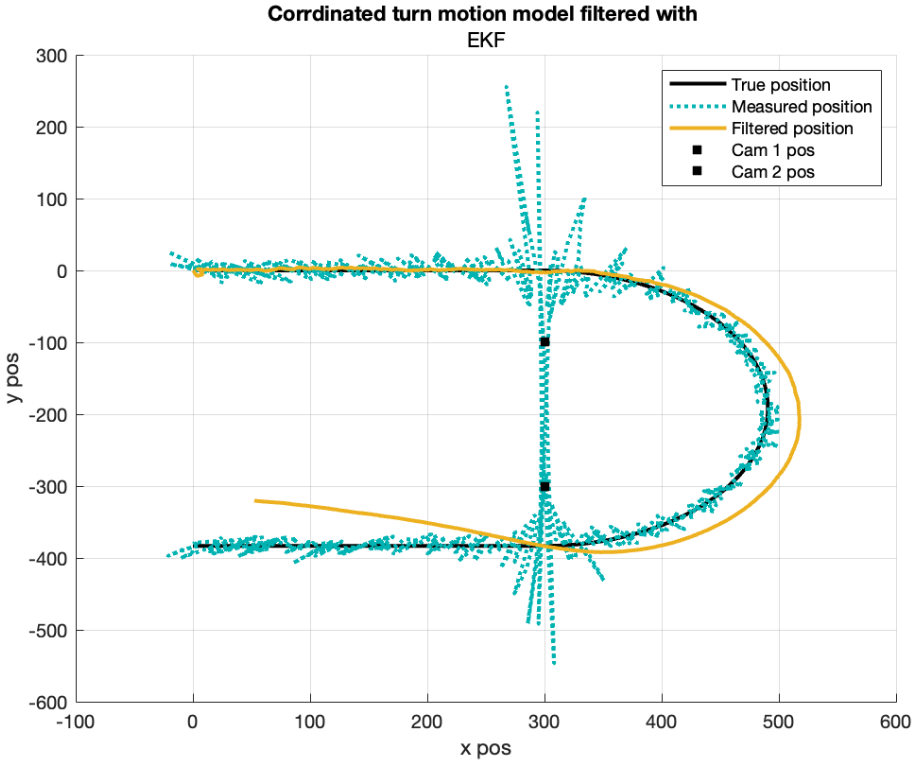

Figure 3.2: Coordinated turn model filtered with EKF with very small covariances σ_v and σ_ω

Task b)

I believe that the initial tuning performs rather well.

Task c)

Properly tuned filter:

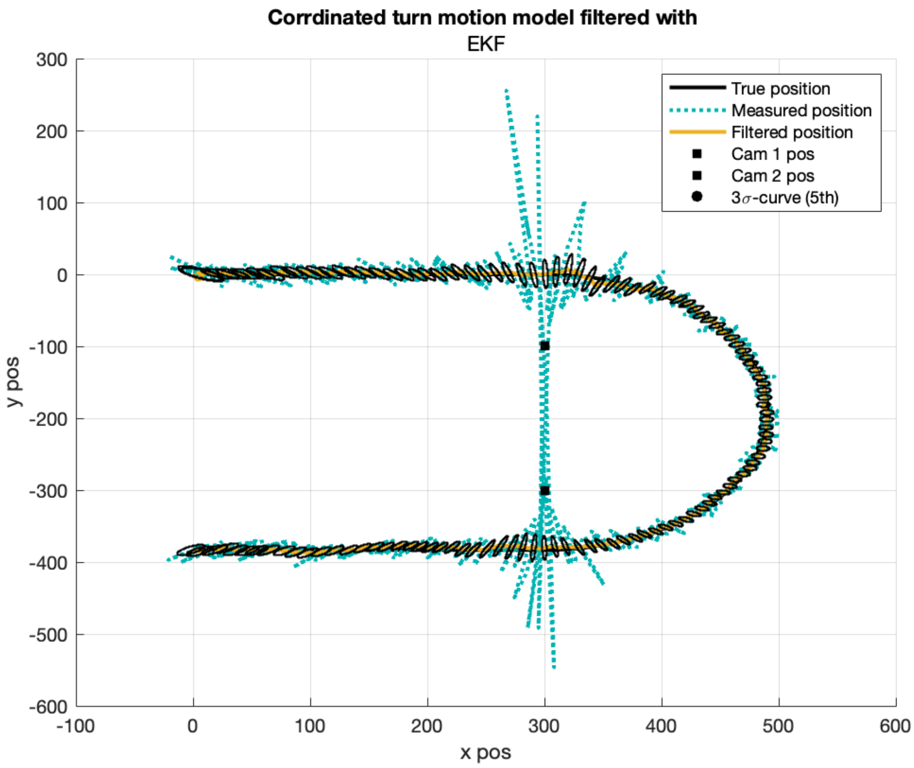

Figure 3.3: Coordinated turn motion model and filtered position with EKF with process noise properly tuned.

Small process noise:

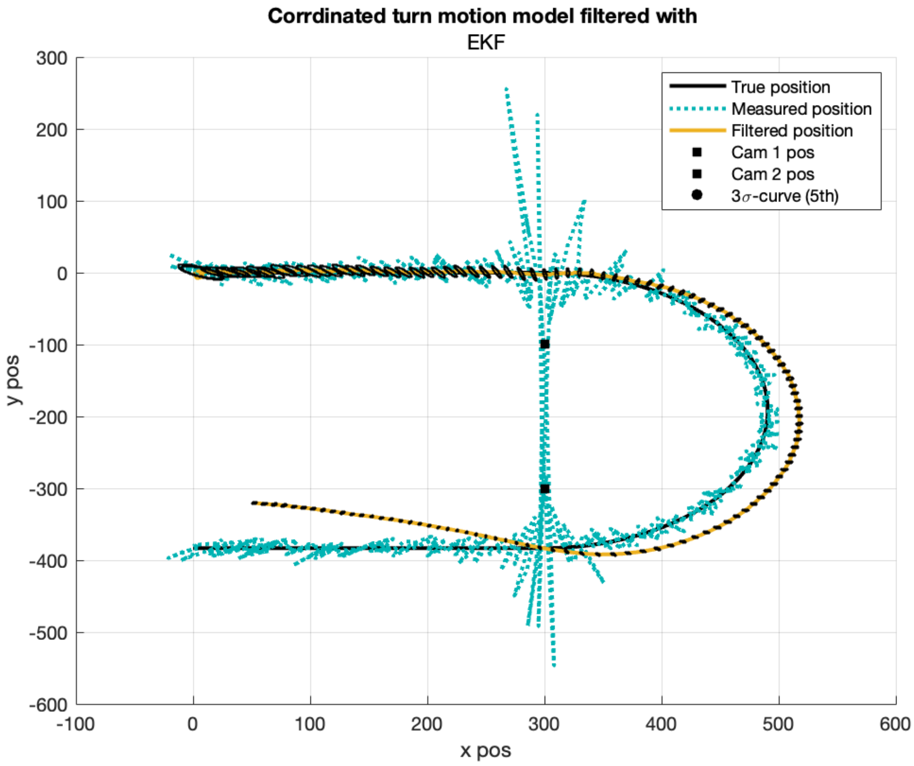

Figure 3.4: Coordinated turn motion model and filtered position with EKF with small process noise.

Large process noise:

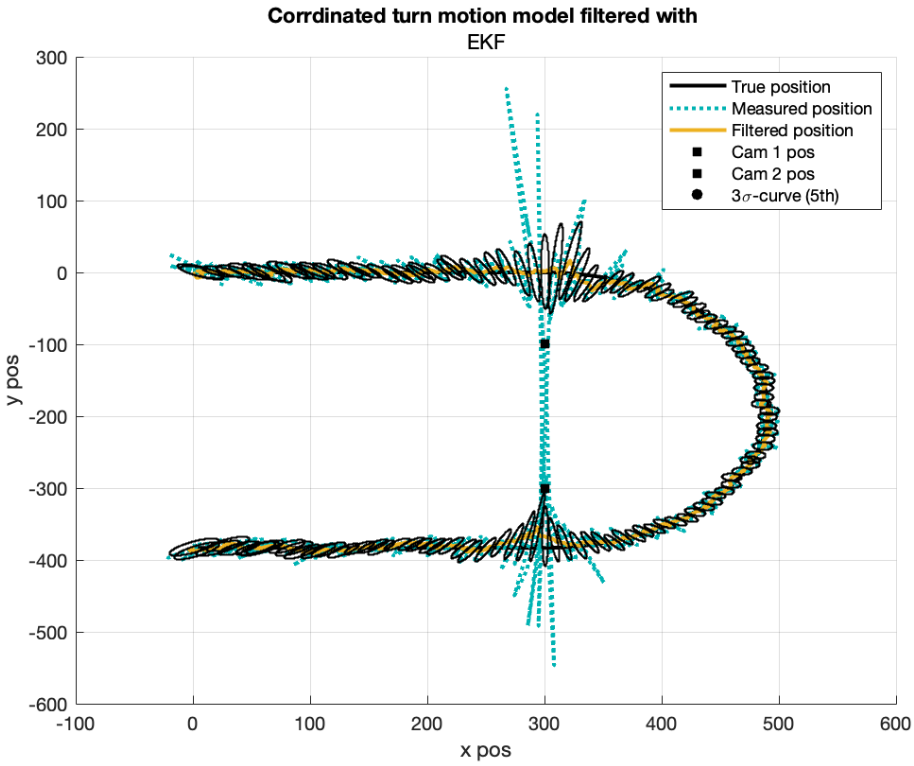

Figure 3.5: Coordinated turn motion model and filtered position with EKF with large process noise.

We use the initial tuning parameters σ_v and σ_ω as given by the assignment description as the good tuned filter. This is scaled up by a factor of 100 for the filter with large process noise, and scaled down with a factor for 1000 for the filter with small process noise. The positional estimation error can be seen in Figure 3.6 below.

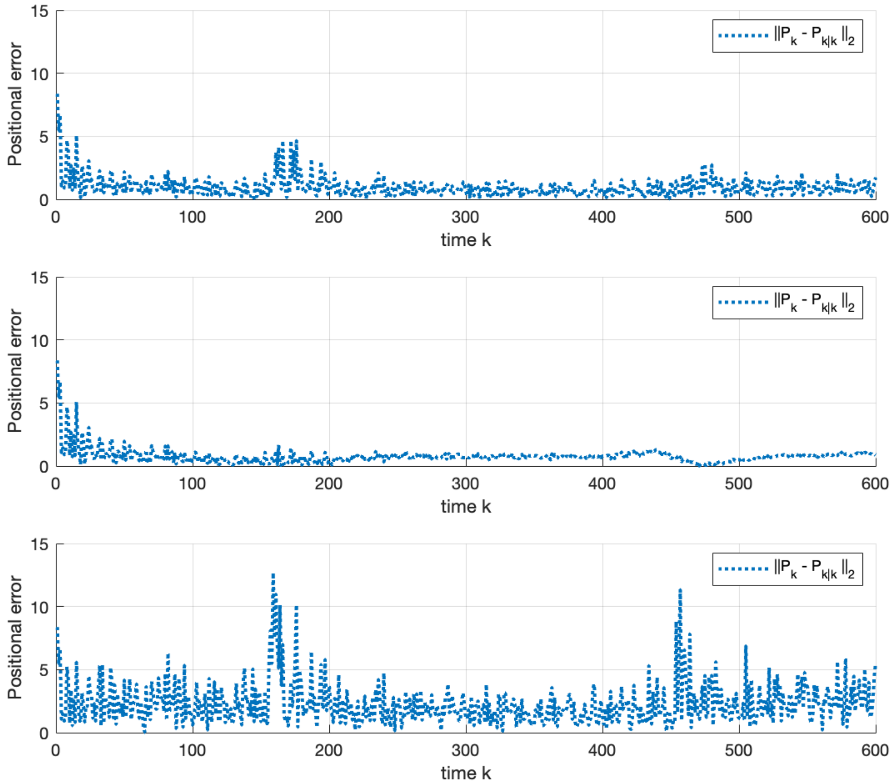

Figure 3.6: Positional errors for a good tuned filter (upper plot), filter for small process noise (middle plot) and filter for large process noise (lower).