Your Paper

You

September 30, 2023

1 Q1

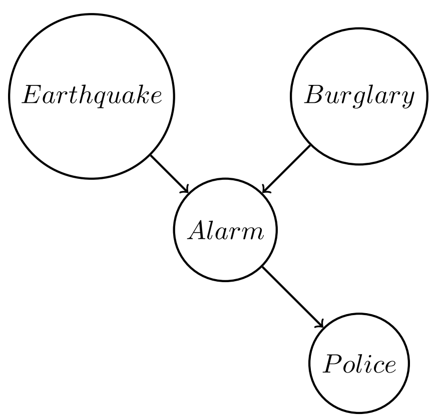
(a) Do robot actions always increase uncertainty? Explain your answer in 2-3 sentences.

No, there can be certain actions that are performed repeatedly to reduce uncertainty. For example, if in a room, you keep moving forward, after a certain point, the uncertainity that you are at the front wall keeps decreasing.

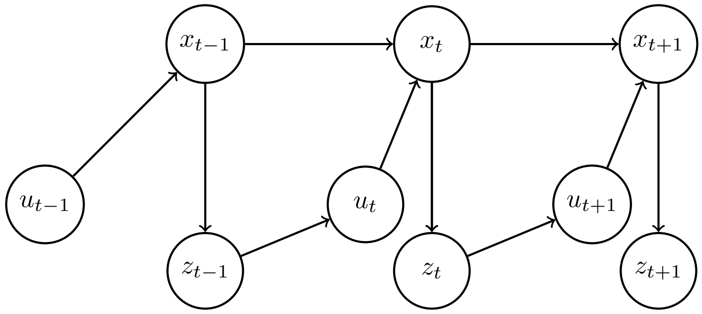
(b) What happens if at any point in Bayesian filtering the probability of a state assignment becomes 1? What are ways to avoid that? Explain your answer in 2-3 sentences.

If the probability of a state assignment becomes 1, this means that it is no longer considering any other options at that particular condition. This means that the filter no longer learns and updates for that state (it becomes stuck), as the update probability is multiplied with the previous steps, if it is 0, the whole thing becomes 0.
A way to avoid this is to maintain a small ϵ value as the minimum probability, so that it is non-zero, and can update.

(c) If an earthquake occurs, or there is a burglary, the alarm is likely to go off. If the alarm goes off, a police may arrive. Design a Bayesian network illustrating the causal relationships.



(d) In the recursive estimation case, what if the controls were dependent on observations? Visualize a Bayesian network showing this dependence.



(e) Why do Extended Kalman Filters (EKFs) fail in handling multiple hypotheses? Explain your answer in 2-3 sentences.

EKFs, like other Gaussian filters, make the assumption that the posterior probability is Gaussian, this assumption fails when there are multiple hypotheses, since Gaussians are unimodal and have a single maximum. Distributions with multiple hypotheses have multiple maximum points (multimodal), thus EKFs fail.

2 Q2

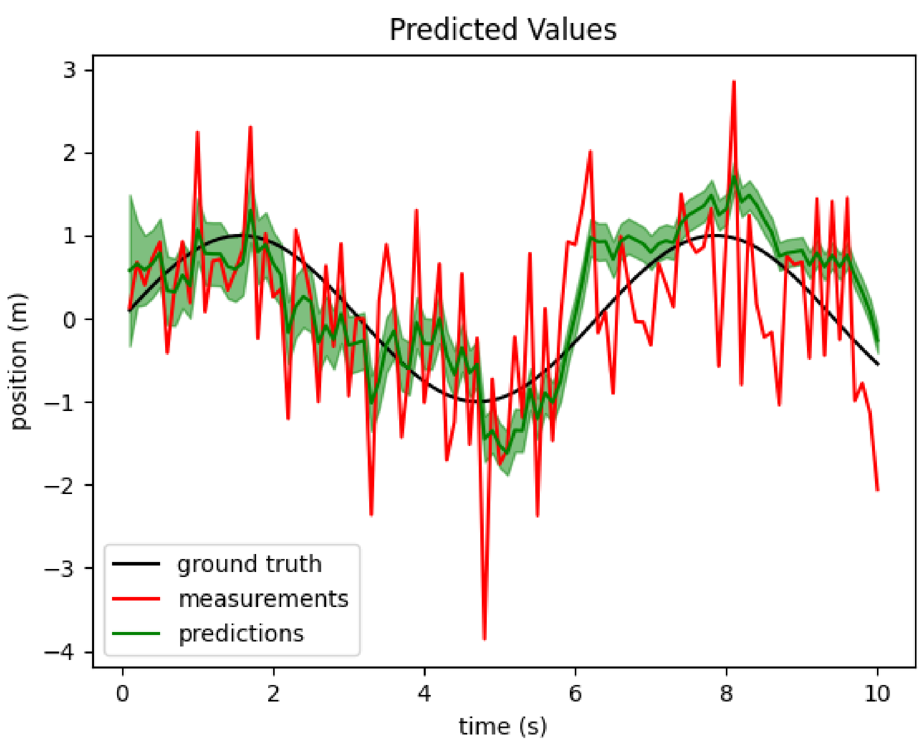


Figure 1: The KF estimates for 2a over 100 timesteps

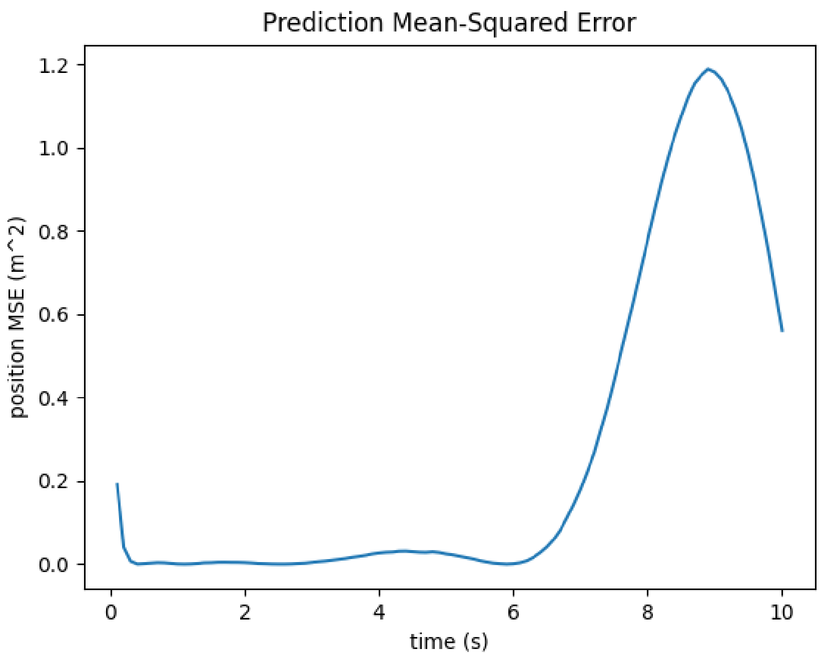


Figure 2: MSE for the position over 100 timesteps for 10000 samples

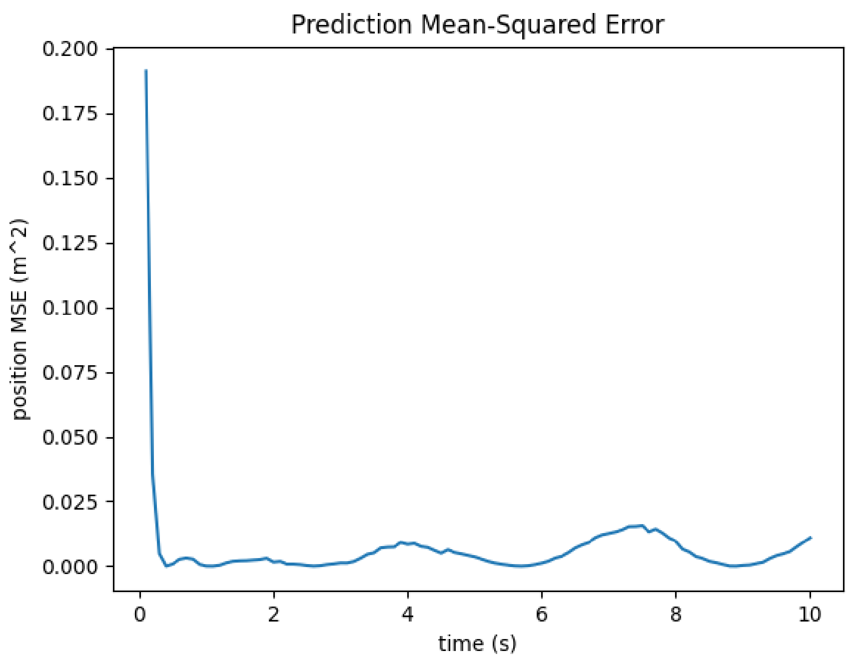


Figure 3: MSE for 2b, adding noise to the motion model

Comparing the two MSE values. You can see that due to the lack of noise in 2a, the filter seems to overcommit to a certain incorrect state estimation, and then propogates that further, increasing the noise as time goes on. In contrast, when noise is added in 2b, although there is some variance, the overall loss stays quite low.

3 Q3

To make the state estimate the position as well as α, we append α to the state.

$$x_{1,t} = x(t)$$

$$x_{2,t} = \alpha$$

$$x_{t+1} = \begin{bmatrix} x_{2,t} & 0 \end{bmatrix} \begin{bmatrix} x_{1,t} \\ x_{2,t} \end{bmatrix} + \begin{bmatrix} w_t \\ 0 \end{bmatrix}$$

$$z_t = \sqrt{x_{1,t}^2 + 1} + v_t$$

Let us first linearize the dynamics model:
Using the first order Talyor Approximation:

$$g(x_{t-1}, u_t) \approx g(u_t, \mu_{t-1}) + G_t(u_t, \mu_{t-1})(x_{t-1} - \mu_{t-1})$$

$$g(u_t, \mu_{t-1}) = \begin{bmatrix} \mu_{2,t-1} * \mu_{1,t-1} \\ \mu_{2,t-1} \end{bmatrix}$$

$$G_t(u_t, \mu_{t-1}) = \begin{bmatrix} \mu_{2,t-1} & \mu_{1,t-1} \\ 0 & 1 \end{bmatrix}$$

$$g(x_{t-1}, u_t) \approx \begin{bmatrix} \mu_{2,t-1} * \mu_{1,t-1} \\ \mu_{2,t-1} \end{bmatrix} + \begin{bmatrix} \mu_{2,t-1} & \mu_{1,t-1} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{1,t-1} - \mu_{1,t-1} \\ x_{2,t-1} - \mu_{2,t-1} \end{bmatrix}$$

Therefore, finally

$$x_{t+1} = \begin{bmatrix} \mu_{2,t-1} * \mu_{1,t-1} \\ \mu_{2,t-1} \end{bmatrix} + \begin{bmatrix} \mu_{2,t-1} & \mu_{1,t-1} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{1,t-1} - \mu_{1,t-1} \\ x_{2,t-1} - \mu_{2,t-1} \end{bmatrix} + \begin{bmatrix} w_t \\ 0 \end{bmatrix}$$

Now let us move to the sensor model,

$$z_t = \sqrt{x_{1,t}^2 + 1} + v_t$$

$$h(x_t) = h(\mu_t) + H_t(\mu_t)(x_t - \mu_t)$$

$$h(x_t) = \sqrt{\mu_{1,t}^2 + 1} + \begin{bmatrix} \frac{\mu_{1,t}}{\sqrt{\mu_{1,t}^2 + 1}} & 0 \end{bmatrix} \begin{bmatrix} x_{1,t} - \mu_{1,t} \\ x_{2,t} - \mu_{2,t} \end{bmatrix}$$

Finally, the sensor model is,

$$z_t = \sqrt{\mu_{1,t}^2 + 1} + \begin{bmatrix} \frac{\mu_{1,t}}{\sqrt{\mu_{1,t}^2 + 1}} & 0 \end{bmatrix} \begin{bmatrix} x_{1,t} - \mu_{1,t} \\ x_{2,t} - \mu_{2,t} \end{bmatrix} + v_t$$

Now that we know the equations, if we find a good initialization point, we can use EKF to estimate the values of α

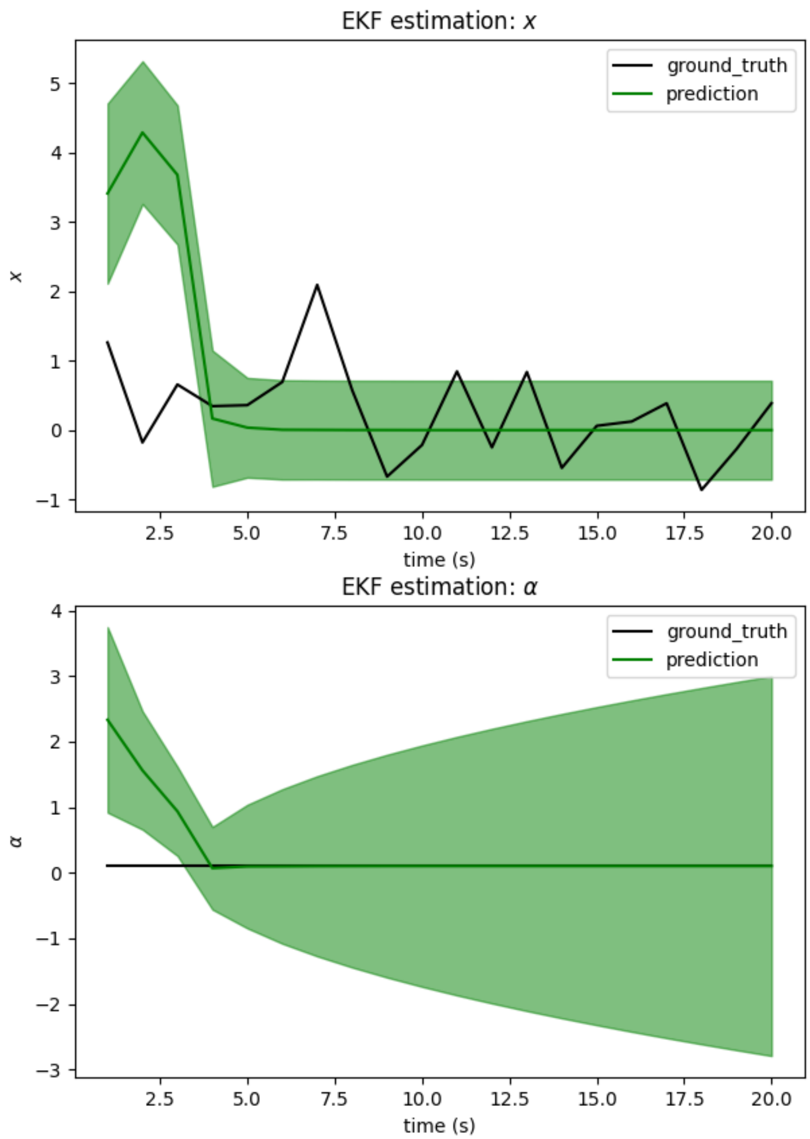


Figure 4: Running EKF given the R, Q and initial assumptions given, this is the estimation for T=20

As you can see here, the estimation of α and x seem to work reasonably well. Showing that the linearization is quite effective. However, due to the the noise and errors due to linearization and the noise due to Q, we can see the variance of α grow as time goes on, showing that there is room for improvement. Similarly, the noise makes the value of x move around, which means the variance cannot reduce beyond a certain point. Furthermore, since Q ¿ R, the variance of α is higher.