SSY191 - Sensor Fusion and Nonlinear Filtering
Solution to analysis in Home Assignment 02

Lucas Rath

Analysis

In this report I will present my independent analysis of the questions related to home assignment 1. I have not discussed the solution with anyone and I swear that the analysis written here are my own.

1 Scenario 1 A first Kalman filter and its properties

A) As can be seen in Figure 1, the measurements behave fairly bad according to the model. Indeed, this depends on the covariance of the measurement noise, which is considerably high. According to the measurement model, the measured samples are normal distributed as $p(y_k|x_k) = \mathcal{N}(y_k; x_k, R)$, where R is the covariance noise. We are then able to visually confirm this in the figure, since the measurements seem to be normally distributed around the true state sequence. In addition, 99.7% of the measurements should be contained within $y_k = [x_k - 3\sqrt{R}, x_k + 3\sqrt{R}] = [x_k - 4.7, x_k + 4.7]$, which can be again visually confirmed in the figure.

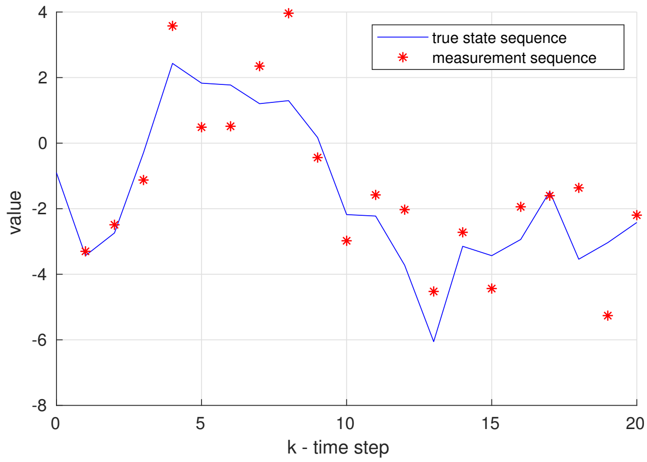


Figure 1: *State and measurement sequence for a linear and Gaussian state space model*

B) The estimates of the filter outputs approximate well the true states and can be said to be reasonable, given that the measurements are very noisy, see Figure 2. This is only possible because at each step, the Kalman filter uses information of the prior together with the motion model to be able to predict and then update the estimated output using also the measurement model. If the filter had only taken into account the measurements and not the motion model, the outputs would have been much noisier and far from the true estimated states.

In addition, the covariance of the posterior distribution $p(x_k|y_{1:k}) = \mathcal{N}(x_k; \hat{x}_{k|k}, P_{k|k})$ at each step represents very well the uncertainty of the estimates. As can be verified by looking at the 3-sigma level curves in Figure 2, all the measurements lie inside this region. Theoretically, this region should contain 99.7% of the measurements, which is the case for this filtering experiment.

To be able to better illustrate the output uncertainty, the posterior density and the true state are plotted at a few different time instances, as seen in Figure 3. As can be verified, these posterior distributions are fairly good, because the true value has a relevant likelihood in comparison to the maximum, i.e. these distribution represent fairly well the probability of the true state being in certain range.

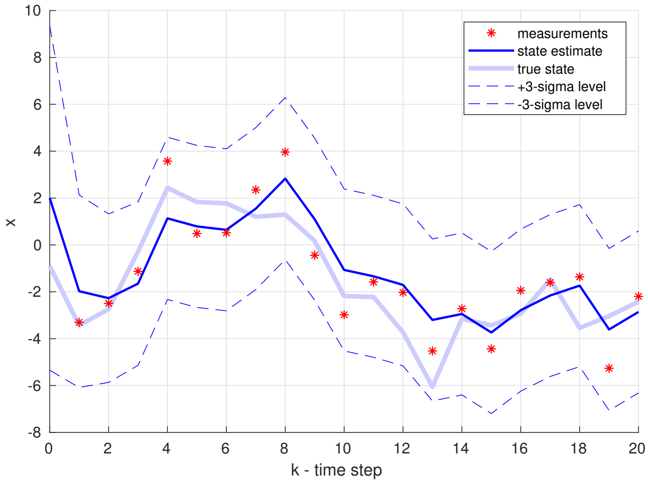
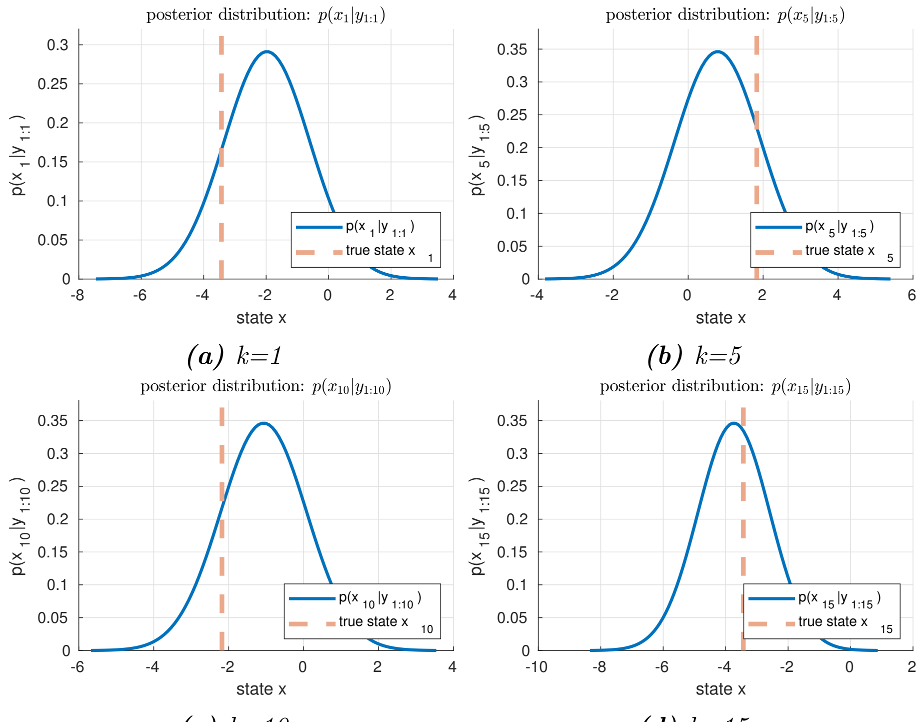


Figure 2: *State and measurement sequence for a linear and Gaussian state space model. In addition, the state estimate using linear Kalman Filter and the 3-sigma level of the estimated states are plotted.*



(a) *k=1* ***(b)*** *k=5*

(c) *k=10* ***(d)*** *k=15*

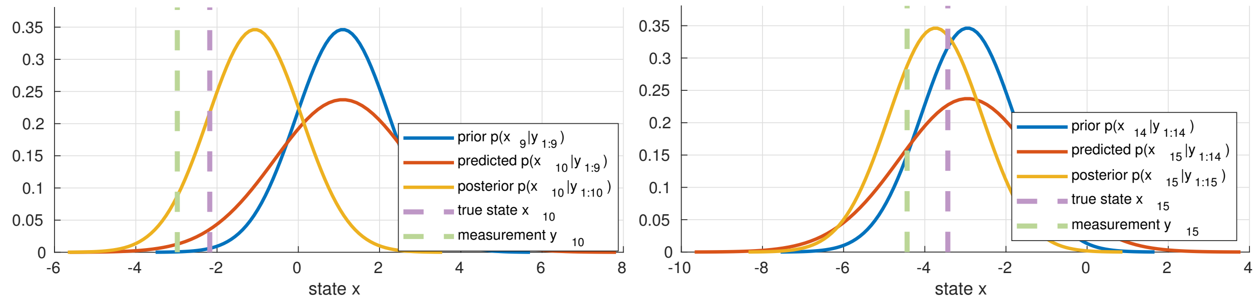
Figure 3: *Posterior density and true state for some different time instances k.*

C) Figure 4 shows now the prior, prediction and update distribution as well as the true state and measured samples for two time instances. In the prediction step, we can verify that the distribution will remain with the same mean as the prior but with increased covariance, which is very reasonably, according to the given motion model $p(x_k|x_{k-1})$. The posterior will then be built taking into account the predicted density $p(x_k|y_{1:k})$, the measurement model $p(y_k|x_k)$ and the value of the observed output. The result is an updated posterior distribution, which represents very well the probability of the true state given the past measurements up to the actual time step. As can be verified, the true state fits well the posterior distribution when we analyze in the course of filtering process.

As a last remark, we can see that the posterior mean is not equal the measurement y_k. Instead it takes into account the prior distribution and the last measurements to then give a better estimate of the true state. The mean of the posterior distribution is always located somewhere in between the actual measurement and the mean of the predicted distribution, which is a very reasonable behaviour. This can be easily proven by analyzing the formula of the posterior distribution, which according to the "lemma for the update step" leads to:

$$E[x_k|y_{1:k}] = \hat{x}_{k|k-1} + \frac{P_{k|k-1}}{(P_{k|k-1} + R_k)}(y - \hat{x}_{k|k-1}) \tag{1}$$

for the case when $H_k = 1$. Since $P_{k|k-1}$ and R_k are positive definite, the cited statement always holds for linear and Gaussian models. How close the mean of the posterior distribution will be in relation to the prior mean and the actual measurement depends however on the motion and measurement noise of the model.



(a) *k=1* ***(b)*** *k=5*

Figure 4: *Prior, predicted and posterior density for a given time instant of the Kalman filter. The true state and the measurement are represented by the vertical dashed lines.*

D) An important step when designing the Kalman filter consists in checking its consistency. Since we are dealing with simulations, we know the true state and we can study the distribution of estimation error $x_k - \hat{x}_{k|k}$, which should have mean close to zero, in case the filter is performing well, without a bias in the estimate. To check this, we generate a long a long true state sequence and a corresponding measurement sequence, which is filtered using Kalman filter. The normalized histogram of the estimation error can be seen in Figure 5, which is plotted together with the normal distribution $\mathcal{N}(x; 0, P_{k|k})$.

First we observe that the mean of the estimation error is -0.0364, which is a reasonable value and represents a good filtering performance. Second, the normal distribution $\mathcal{N}(x; 0, P_{N|N})$ fits very well the histogram. The reason for this behaviour is because, as said before, the estimation error mean is close to zero and more important, because over time $P_{N|N} = Cov(x_k|y_{1:k}) = Cov(x_k - \hat{x}_k|y_{1:k})$ will converge to a constant scalar/matrix, which is equal to the estimation error covariance.

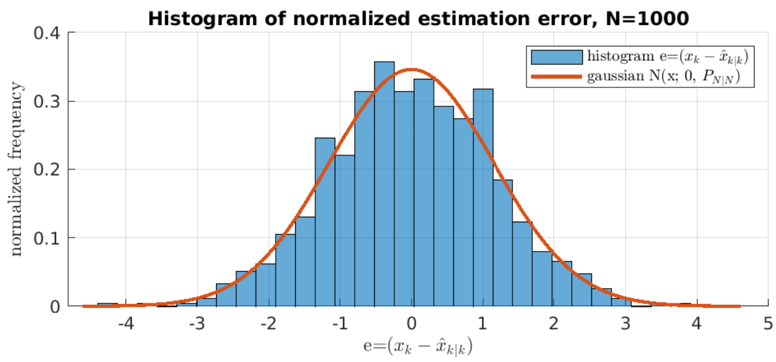


Figure 5: *Comparison of the histogram of the estimation error $x_k - \hat{x}_{k|k}$, generated by the Kalman filter and the probability density distribution $\mathcal{N}(x; 0, P_{N|N})$, where $P_{N|N}$ is the covariance of the posterior $p(x_N|y_{1:N})$ and N=1000 is the length of the generated state sequence.*

Another important consistency measure is the analysis of the innovation $v(k) = y_k - H_k.\hat{x}_{k|k-1}$. Ideally, the innovation should have mean equal zero and a covariance $Cov(v_k, v_{k-l})$ equal 0 for all $l \neq 0$. This two conditions can be proven to hold for linear and Gaussian state space models.

Analyzing the same results from the experiment of Figure 5, we get an expected value for the innovation equal to -0.0364, which is fairly close to zero. It then follows that $Cov(v_k, v_{k-l})$ is also called auto-correlation function and can be estimated numerically. The results of the normalized auto-correlation of the innovation for different lags are plotted in Figure 6. As can be verified, the auto-correlation at lag equal zero is much higher than at any other lags, which confirms the consistency of our Kalman filter.

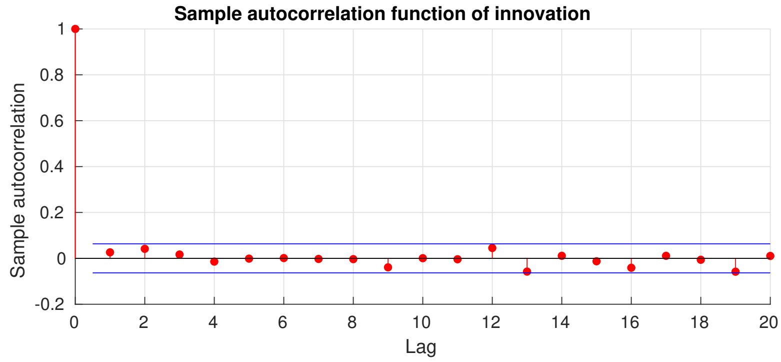


Figure 6: *Normalized auto-correlation function of the innovation process v_k of the Kalman filter for a state and measurement sequence of 1000 samples.*

F) We are going to study now what happens if we choose an incorrect prior distribution. Figure 7 shows measurements and the results of two Kalman filters; the blue line represents the Kalman with the correct prior distribution and the one in yellow with a wrong prior. As can be verified, the filter in yellow rapidly corrects its estimation mean and variance to reasonable values in only three steps. After five steps its mean and covariance converge to the exactly same value as the blue filter. This happens because in the course of time, the posterior $p(x_k|y_{1:k})$, will have almost no influence from the first prior $p(x_0)$ and the history of measurements $y_{1:k}$ will update the posterior accordingly to a better distribution.

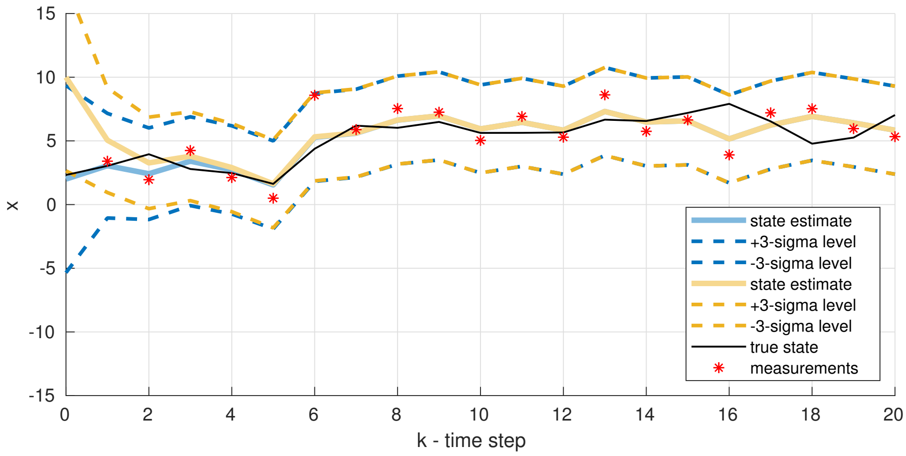
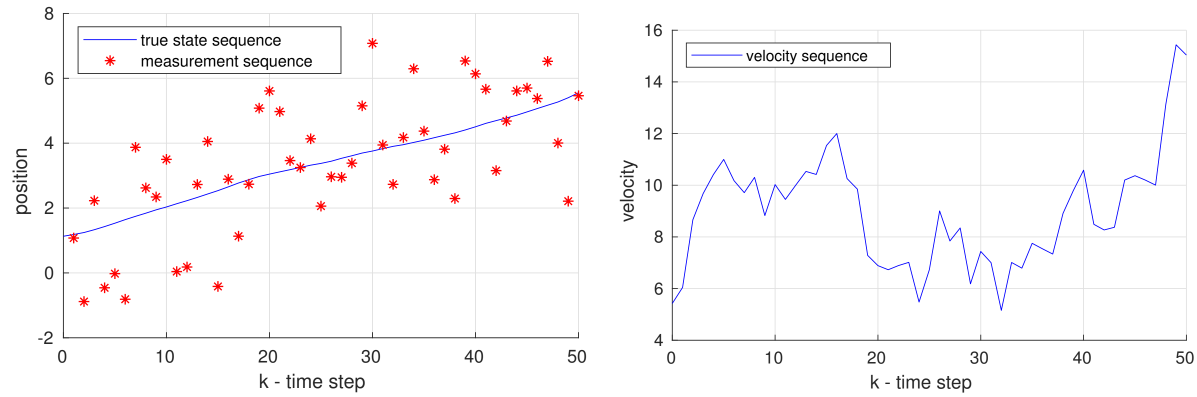


Figure 7: *Measurements and two Kalman filter state estimates using different prior distributions. The blue line is using the true prior distribution $p(x_0) = \mathcal{N}(x_0; 2, 6)$ while the yellow line is using a wrong distribution $p(x_0) = \mathcal{N}(x_0; 10, 6)$.*

2 Scenario 2 Kalman filter and its tuning

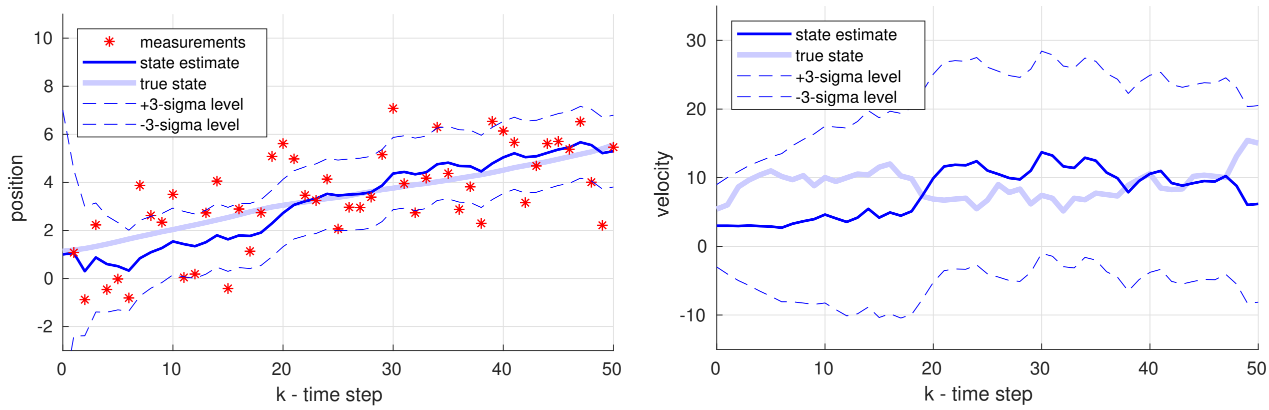
A) Figure 8 shows a true state and its measurement sequence based on a constant velocity (CV) model. As expected, the velocity is described as a random walk, i.e. the acceleration is described as a white and zero-mean noise process. This result seems reasonable because as can be seen in the position plot of the Figure 8a, the position derivative is kept piece-wise constant and does not change too much.



(a) Position *(b) Velocity*

Figure 8: *Generated state and measurement sequence using constant velocity model for sampling period equal 0.01 seconds.*

B) We then proceed by applying the Kalman filter to the state and measurement sequences presented above, as seen in Figure 9. As can be seen, even though the measurements are very noisy, the Kalman filter is able to output very reasonable estimates. This is only possible because we have used a correct model to represent the dynamics of our observed system. Since we assume a constant-velocity model, which is consistent with the reality, we make a good prediction, which helps the filtering process in determining a good posterior density. As expected, in Figure 9, the velocity estimate is kept fairly constant. The velocity is allowed to change according to a Gaussian distribution but is also driven by the measured positions.



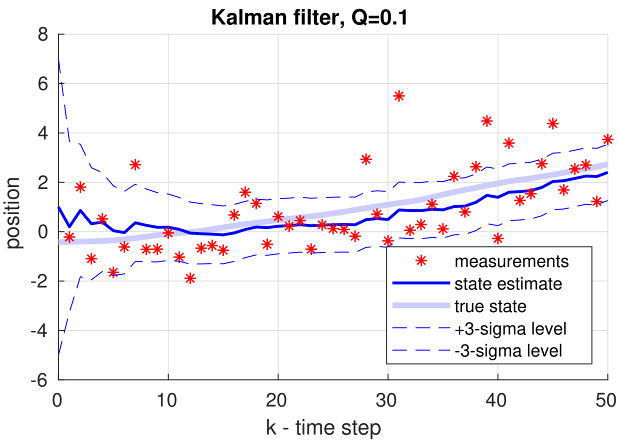
(a) Position *(b) Velocity*

Figure 9: *Generated state and measurement sequence using constant velocity model for sampling period equal 0.01 seconds. In addition, the Kalman filter estimates are shown together with its 3-sigma level curves.*

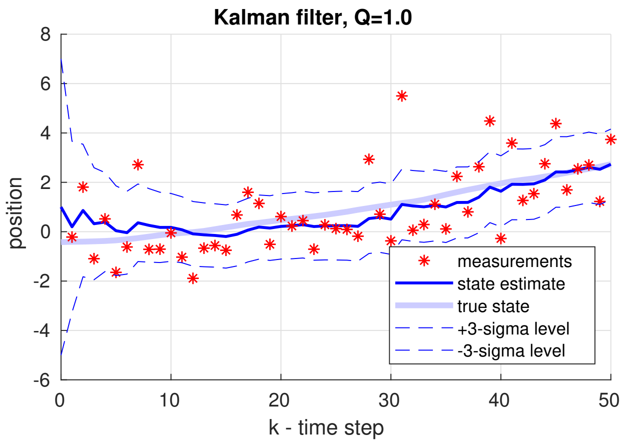
C) Next, we will study what happens when choosing different motion noise variances Q. Figure 10 shows the Kalman filter estimates for different values of Q. The state sequence was generated using Q=1.5.

We expect that the smaller Q we use in our Kalman filter, the slower we allow our model to change velocity. As an extreme case, if we choose Q=0, the speed would not be allowed to change at all. In addition, the smaller Q in comparison to R, the more we assume we trust more our motion model than our measurements. As a consequence, the estimated states are more stable, because the estimated position will be closer to the mean of the predicted distribution $E[x_k|y_{1:k-1}]$ generated by the motion model, and less susceptible to the measurement noise. However, we will not be able to achieve larger accelerations. This can be confirmed by analyzing Figure 10a. The true state sequence presented an acceleration inconsistent with our motion model distribution $p(x_k|k_{k-1})$, i.e. the likelihood that the velocity would change that fast is very low. As a result, the Kalman filter estimate outputs a position that is not able to follow properly the true position and this translates into a bias in the estimation error. However, the estimates are not so noisy, what can be also good depending on the application. If we had plotted a 3D 3-sigma ellipse of the joint distribution $p(\left[\begin{smallmatrix} x_k \\ y_k \end{smallmatrix}\right] | y_{1:k-1})$, we would see that the smaller Q in comparison to R, the more the principal component is aligned with the y_k axis.

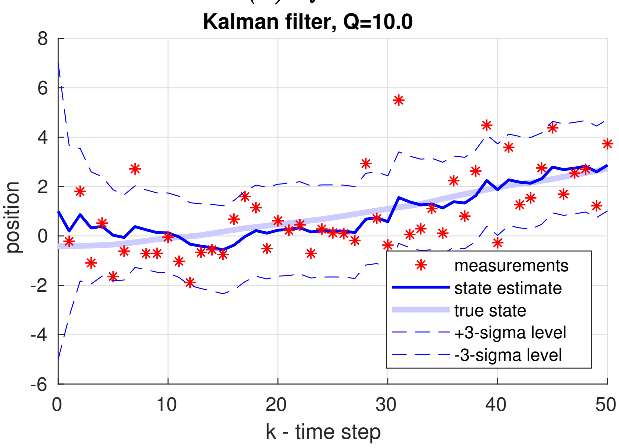
On the other hand, if we choose a very high Q, as in Figure 10c, we assume that is more likely that our velocity can have larger changes at each time step. The result is that our estimates are able to follow better the acceleration imposed by the true state sequence but we have more noise introduced by the measurements, i.e. the innovation has a higher variance. As a direct consequence, the posterior covariance and therefore the 3-sigma ellipse region is also larger than for small values of Q.



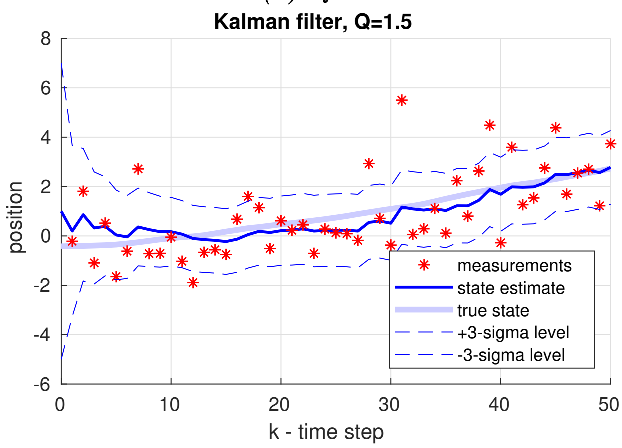
(a) Q=0.1



(b) Q=1



(c) Q=10



(d) Q=1.5

Figure 10: *Kalman filter of a constant velocity model for different motion noise variance Q. The value of Q used to generate the state sequence was equal to 1.5.*