Guidance and Control of Vehicles
TTK4190

Aircraft Autopilot Design with State Estimation Part 2

Computer exercise 2

Authors:
Henning Ward & Andreas Thyholt Henriksen

October 13, 2017

NTNU – Trondheim
Norwegian University of Science and Technology

1 State Estimation using a Kalman-filter

The goal is to implement a Kalman Filter that shall estimate the sideslip β, roll rate p and yaw rate r, based on limited information about the states of the system through the measurements, that is measurements of ϕ, p and ψ contaminated by Gaussian white noise. In addition the aileron input is known, but the dynamic is considered unknown and therefore neglected. The bias d affecting the coordinate-turn equation is still considered to be a part of the system.

3.1

The system equations given by the aircraft model in the assignment can be changed to fit the system equations for the Kalman Filter, which can be described by a continuous form Kalman-filter. A discrete form could have been calculated, but since it contains the same information in a continuous form and is done easily in `MATLAB` using `c2d`{...}, it is not necessary in this task.

First the aileron dynamic is neglected since it is considered unknown, which gives $\boldsymbol{x} = [\beta, \phi, p, r]$ and input to the system dircetly from $\boldsymbol{u} = \delta_c$. Thus the fifth row of the state matrix $\mathbf{A}$ must be removed. The fifth column on the other hand except the last element, should be considered to be the input matrix $\mathbf{B}$. Second the output matrix $\mathbf{C}$ must be reduced such that β no longer is available, thus the third row must be removed. Doing so the system equations of the Kalman Filter can be represented through the standard form state-space model with $\mathbf{E} = \boldsymbol{I}$ given by

$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}\boldsymbol{u} + \mathbf{E}\boldsymbol{w} \tag{1a}$$

$$\boldsymbol{y} = \mathbf{H}\boldsymbol{x} + \mathbf{D}\boldsymbol{u} + \boldsymbol{v} \tag{1b}$$

with state, input, -and output-matrices

$$\mathbf{A} = \begin{bmatrix} -0.322 & 0.052 & 0.028 & -1.12 \\ 0 & 0 & 1 & -0.001 \\ -10.6 & 0 & -2.87 & 0.46 \\ 6.87 & 0 & -0.04 & -0.32 \end{bmatrix}, \mathbf{B} = \begin{bmatrix} 0.002 \\ 0 \\ -0.65 \\ -0.02 \end{bmatrix}, \mathbf{C} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \mathbf{D} = \mathbf{0}$$

It is a requirement that the system is observable in order to reconstruct recursively the state vector $\boldsymbol{x} \in \mathbb{R}^{\mathrm{n}}$ from the measurement vector $\boldsymbol{y} \in \mathbb{R}^{\mathrm{m}}$ and input $u = \delta_c$. Using the `MATLAB`-functions `rank(obsv(`$\mathbf{A}$,$\mathbf{C}$`))` $= 5 = \mathrm{n}$ ensures that this is the case.

The process -and measurment-noise $\boldsymbol{w} \sim \mathcal{N}(0, \mathbf{Q})$ and $\boldsymbol{v} \sim \mathcal{N}(0, \mathbf{R})$ is assumed to be zero-mean Gaussian white noise processes with process-noise and measurement-noise covariance matrices

- $\mathbf{Q} = \mathbf{Q}^\top, \quad \mathbf{Q} \succ 0, \quad \mathbf{Q} \in \mathbb{R}^{\mathrm{n}\times\mathrm{n}}$
- $\mathbf{R} = \mathbf{R}^\top, \quad \mathbf{R} \succeq 0, \quad \mathbf{R} \in \mathbb{R}^{\mathrm{m}\times\mathrm{m}}$

The matrix $\mathbf{Q}$ is a measure of the accuracy of the model, and $\mathbf{R}$ is a measure of the uncertainty of the centroids of the measurements. Low values of the entries in $\mathbf{Q}$ implies an accurate model, while low values of the entries in $\mathbf{R}$ implies greater certainty in the measurements. These values can be estimated, and in practice one often start with some reasonable initial guess/estimate for the diagonal entries in $\mathbf{Q}$ and $\mathbf{R}$, and then tune them. $\mathbf{R}$ is usually known pretty well,

The state estimate error covariance matrix

- $\mathbf{P} = \mathbf{P}^\top, \quad \mathbf{P} \succ 0, \quad \mathbf{P} \in \mathbb{R}^{\mathrm{n}\times\mathrm{n}}$

is computed online, is a measure of the estimated accuracy of the state estimates, in which the diagonal entries is the variance, and of-diagonal the covariance. In short, $\mathbf{P}$ works as a correction factor for how much the filter trusts the model conversely weighted by $\mathbf{Q}$ or the measurements conversely weighed by $\mathbf{R}$, when calculating the optimal Kalman gain $\mathbf{K}$ (see [1], chapter 11.3, for discrete Kalman equations). From this equations one can see that high values for $\mathbf{Q}$ compared to $\mathbf{R}$ makes the model trust the measurements more compared to the model, and the opposite if $\mathbf{R}$ is large compared to $\mathbf{Q}$.

3.2

In the case of measuring roll (ϕ), roll rate p and yaw rate r, a solution could be to use some sort of inertial measurements unit (IMU) which can measure the angular velocities, and further use some integration technique to find the angle for roll. An IMU uses the combination of accelerometers, gyroscopes and sometimes magnetometers to do the measurements. However, since we are only interested in the measurements for ϕ, p, and r, all of the sensors integrated in the IMU are not needed.

Three accelerometers are used to measure linear acceleration in each of the three axis respectively. Since there are no demands of linear measurements in this assignment, such accelerometers are not needed.

Gyros are used to measure angular rate among one given axis. Since measurement of yaw rate and roll rate are needed, angular rate among the x-axis and z-axis must be measured. This is realised by using two gyros respectively.

When it comes to measuring ϕ, the roll rate measurement may be integrated. When this is done, it is important to use some kind of integration technique or by using a reference position provided by e.g. GNSS to prevent drift.

Gyros may be vulnerable to noise caused by vibrations and other equipment on board of the aircraft. These errors will be random, and may be approximated using white noise. They may also be sensitive to linear acceleration and fierce movements of the aircraft. White noise will probably be a poor way to estimate this, since these movements are not a random process. Here it may be beneficial to add some kind of colour to the noise, giving a better approximation to the movement.

Gyros may also be exposed to some kind of drift. This should not be estimated by using white noise, but rather by using some kind of 'random walk in the park' model, for example the Wiener Process.

3.3

The Kalman filter is the optimal linear state estimator when it is unbiased and minimizes the variance. If the noise is assumed white, Gaussian, and uncorrelated with the initial states, the filter gives a minimum-variance estimate of the states. Then no other filter can do a better job in estimating the states, not even a non-linear one. (see `https://www.cds.caltech.edu/~murray/wiki/images/b/b3/Stateestim.pdf`) [2]

In this case the noise is assumed to be white and Gaussian, and therefore the filter is optimal. But as mentioned in section 3.2 this is in pratice most often not the case.

3.4

The Kalman filter was implemented in simulink using a `MATLAB`-function, see the attached autopilot.slx file. Worth mentioning is that it is an implementation of a discrete time variant Kalman filter based on the equations in table 11.1 in [1]. The `MATLAB`-function `c2d`{...} is used to discretize the continuous system matrices. Joseph's form is used to ensure symmetric state estimate error convariance matrix **P**. Since the sensors for measuring the states is discrete, a "zero-hold"-block with frequency set to 100 HZ is used. This correspond to a sampling rate of 100 HZ. All the sampling rates are shown in Figure 1

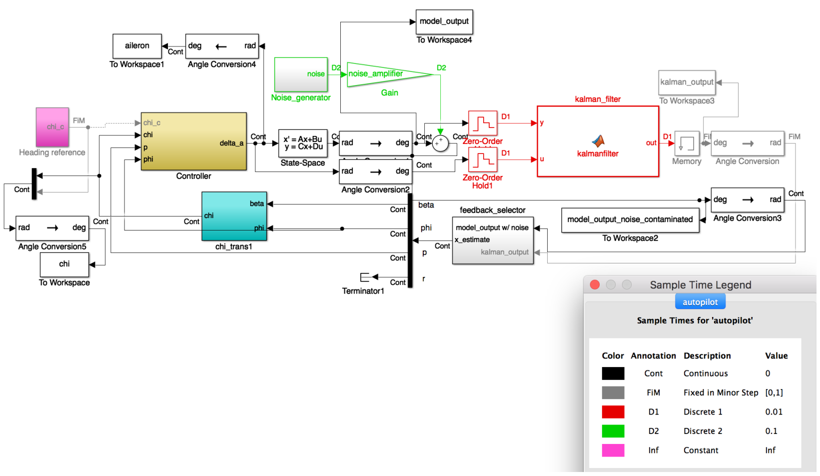

Figure 1: Simulink w. sampling rates

3.5

Since it is assumed that the sideslip β is no longer measureable, the sideslip will now be provided by feedback from the Kalman Filter. This is implemented in the autopilot.slx file. In the feedback_selector subsystem, there is a switch, making it simple to choose between the noise-contaminated measurement for feedback, and the Kalman Filter estimate feedback as asked for in the next exercise.

The noise is introduced in the noise_generator subsystem by using random number generators, with variances as provided in the assignment text.

In the model we chose the $\mathbf{E}$ matrix to be equal to the identity matrix, as the assumption of no correlation between the states are made. Further on, the $\mathbf{R}$ matrix is chosen with the variances given in the assignment text on the diagonal, while the $\mathbf{Q}$ matrix is chosen with elements equal to 10^{-6}, as the process-noise is assumed to be almost negligible.

The actual course angle follows the desired course in the same manner as in problem 2 as seen in figure Figure 2, that is with some overshoot of $\simeq 5°$. It is seen that the state feedback that is contaminated by noise affect the course by causing small and fast oscillations.

These oscillation is again fed into the controller, causing a high frequency oscillations high amplitude aileron reference input. The aileron dynamics acts as a low-pass filter which will lower the effect of the high frequencies, but this attitude is still not desirable because it creates a risk for causing wear and tear to the actuators controlling the aileron angle.

When the feedback signal is contaminated by noise, the true value of the output signal will also contain oscillations as seen in Figure 4, Figure 5 and Figure 6. However, as seen in the zoomed portions, the Kalman Filter estimations are able to persevere the true value with great accuracy, even though there are lots of oscillations present in the system. This result shows that the Kalman Filter provides an excellent estimation of the true value, thus proving that the assumption for the Kalman Filter to be optimal given white noise holds. However, since we are feeding the noisy signal as feedback, the advantage of the filters ability to omit the noise is lost.

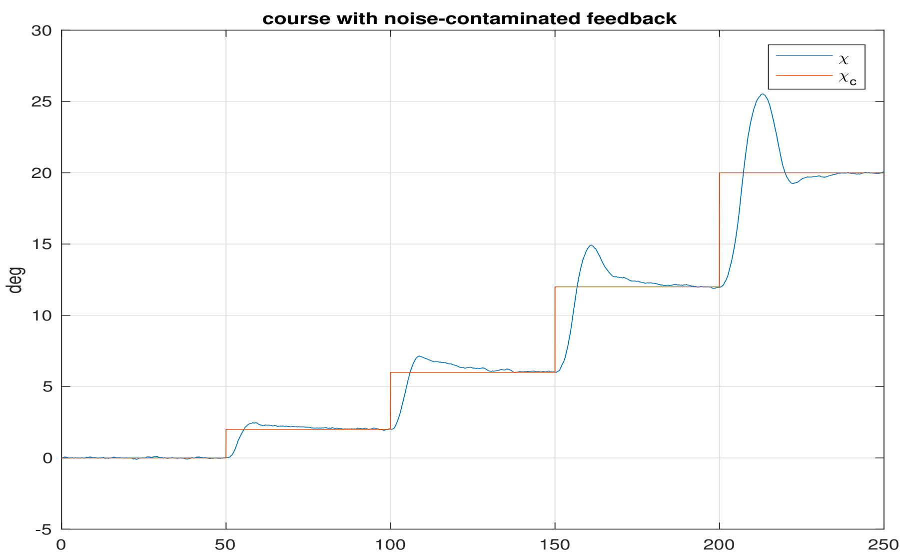

Figure 2: Course and aileron angle of aircraft, reduced model

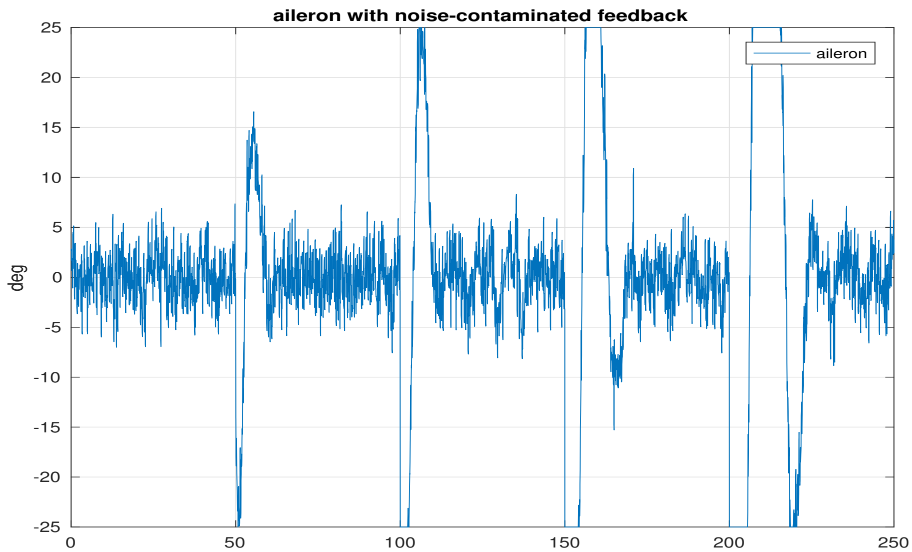

Figure 3: Course and aileron angle of aircraft, reduced model

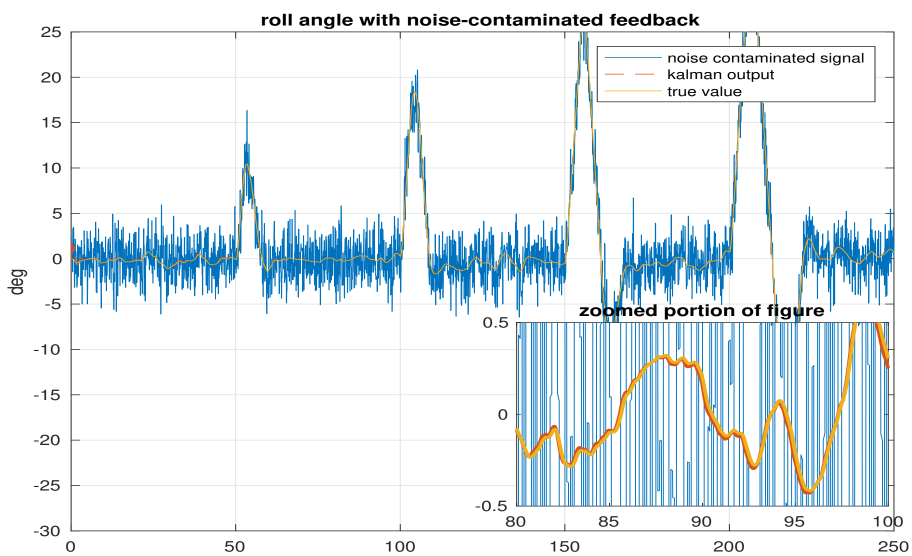

Figure 4: Course and aileron angle of aircraft, reduced model

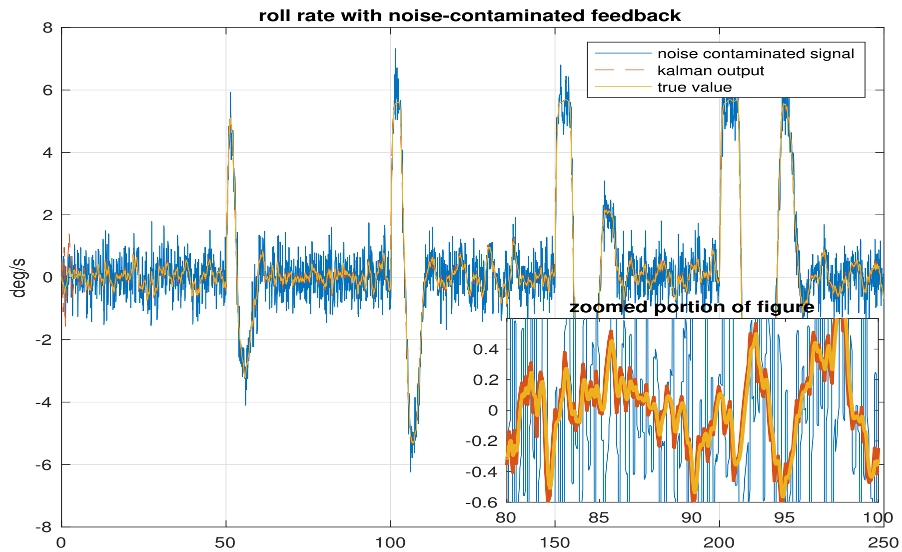

Figure 5: Course and aileron angle of aircraft, reduced model

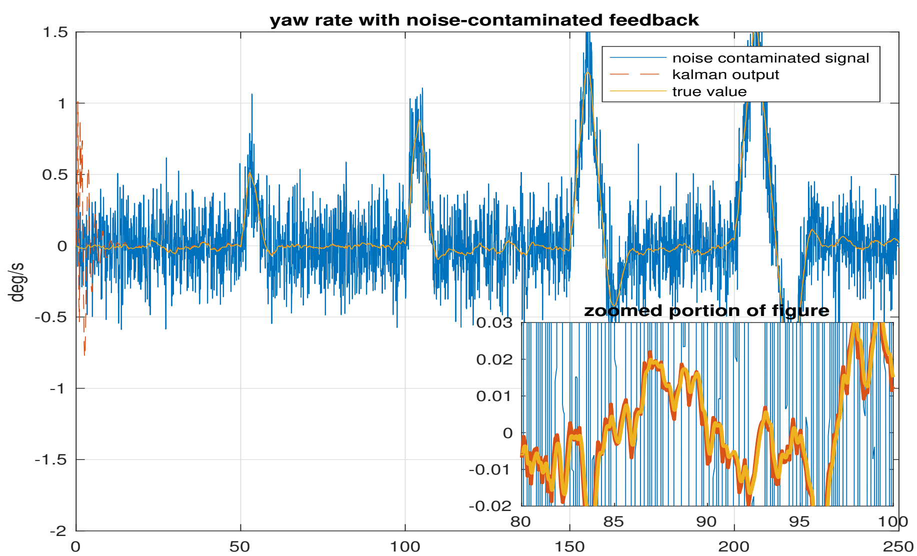

Figure 6: Course and aileron angle of aircraft, reduced model

3.6

Now the state-feedback is replaced with an output-feedback from the Kalman Filter, and therefore it is expected that all the high frequency attitudes from section 3.5 will have a much less contribution to the attitude of the system.

In all the figures below it is seen that almost all the high frequencies which dominated the attitude in section 3.5 is gone. The exception is the first 20 seconds. This is due to the filter which needs some start-up time for the error covariance matrix **P** and the kalman-gain **K** to converge. Another possibility is to instead implement the Discrete Kalman Filter as a steady-state implementation with constant Kalman matrices. It should remove these oscillation, but it could also cause some inaccuracies if the calculations are inaccurate. In practice the system should be given this start-up time before it gets the ability to affect the behaviour of the system, to protect the system from dangerously attitude.

In Figure 7 it is seen that this time the course angle has a nice and smooth trajectory compared to the one in Figure 2, which is as expected. All the high frequencies in the aileron reference input is also gone, which was desirable.

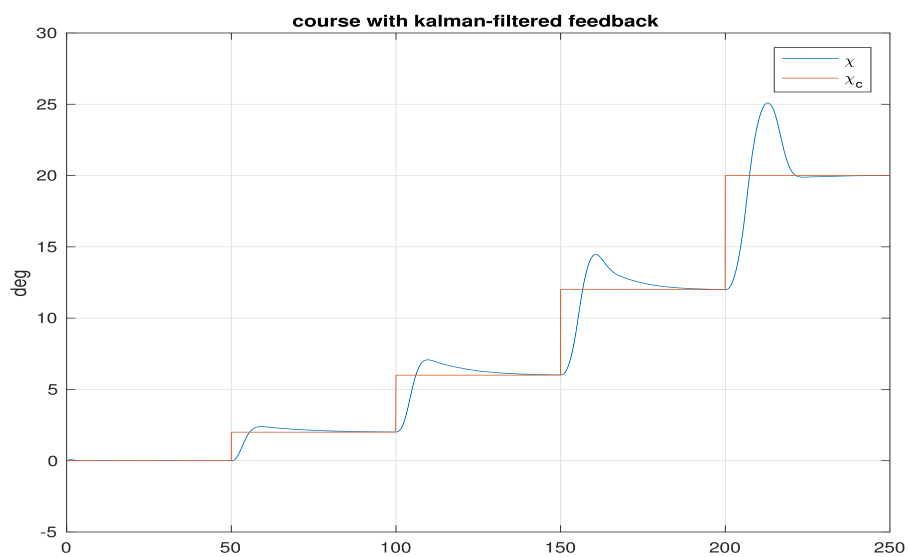

Figure 7: Course and aileron angle of aircraft, reduced model

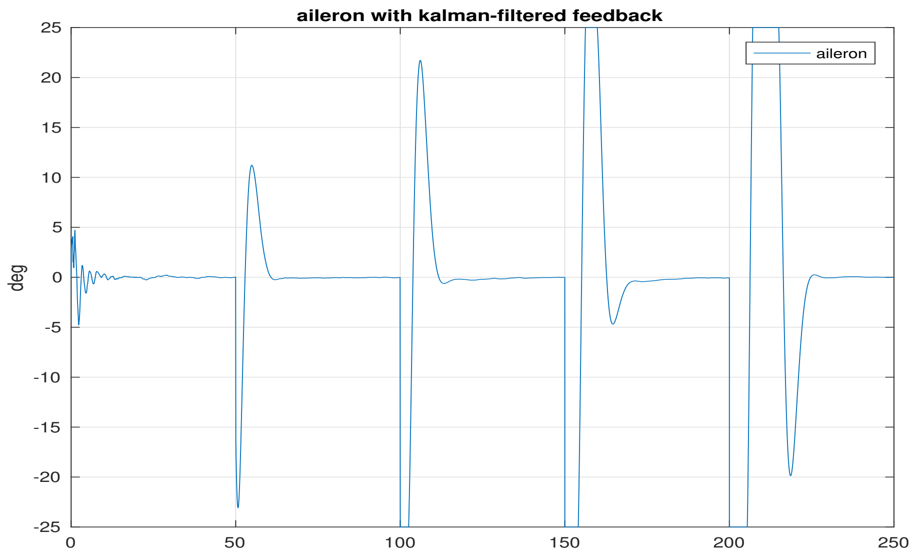

Figure 8: Course and aileron angle of aircraft, reduced model

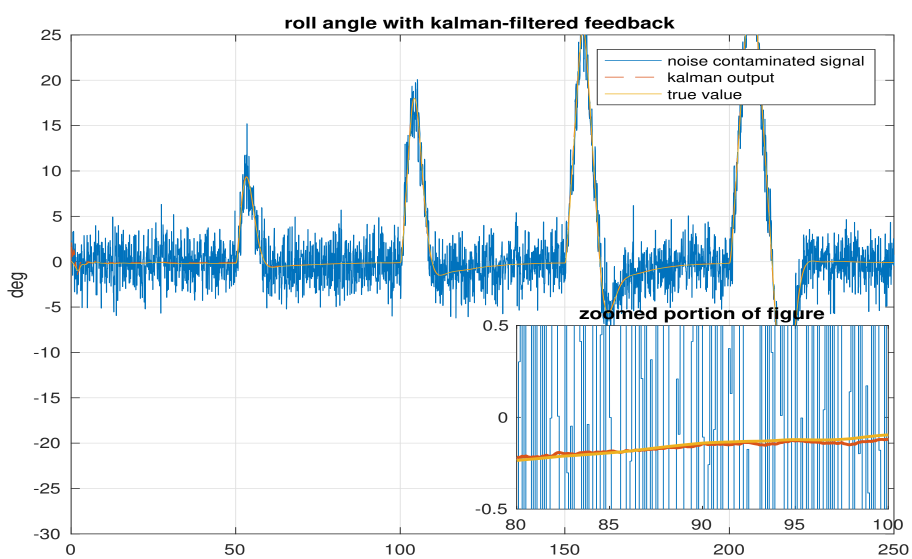

Figure 9: Course and aileron angle of aircraft, reduced model

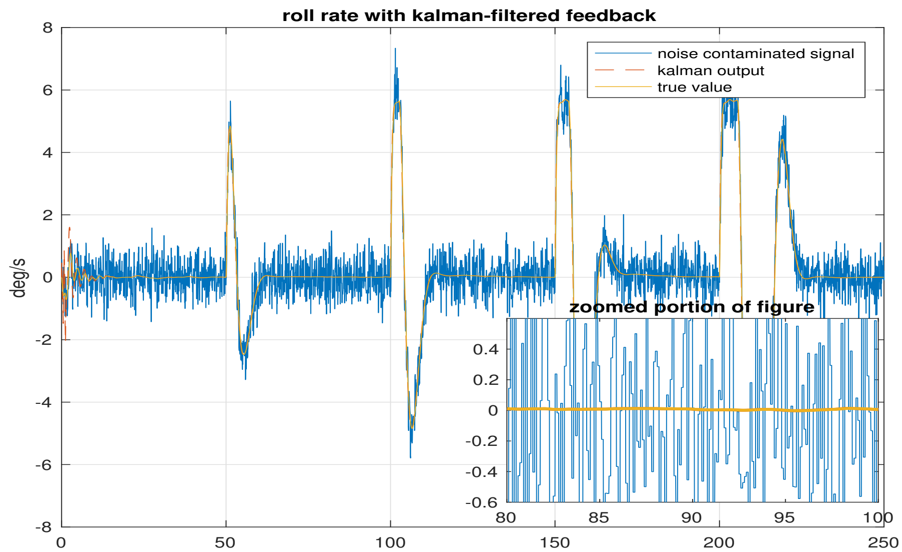

Figure 10: Course and aileron angle of aircraft, reduced model

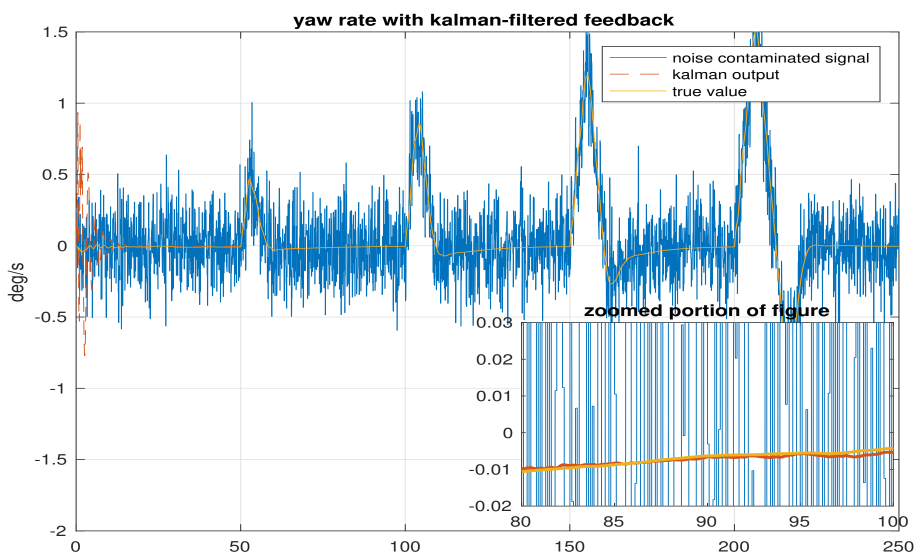

Figure 11: Course and aileron angle of aircraft, reduced model

References

[1] T. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control.* John Wiley & Sons, 2011.

[2] unknown, "State Estimator," https://www.cds.caltech.edu/~murray/wiki/images/b/b3/Stateestim.pdf, [Online; accessed 13-October 2017].

[3] O.-E. Fjellstad and T. I. Fossen, "Quaternion feedback regulation of underwater vehicles," *Proceedings of the IEEE Conference on Control Applications*, vol. 2, pp. 857–862, 1994.

[4] R. Beard and T. McLain, "Small unmanned aircraft, theory and practice."