Kalman filter: Lab report

Jonathan Lys - IMTA/RENNES 1
jonathan.lys@imt-atlantique.net

Abstract

In this lab, we implement a Kalman Filter to retrieve information on a state space from noisy observations, with both zero-mean and non-zero mean noise on the state and observations. We then implement a recursive linear least mean squared error (LLMSE) estimator. Note that the code will be written in Matlab/Simulink.

I. Introduction

In general, the random process is defined by:

$$\mathbf{X}_{k+1} = F_k\mathbf{X}_k + G_k\mathbf{U}_k$$
$$\mathbf{Y}_k = H_k\mathbf{X}_k + \mathbf{B}_k$$

II. Kalman filtering of a 4D-state 2D-observation system

We implement the filter on Simulink (the code can be found in the `kalmanfilter.mdl` file). Here is an overview of the filter:

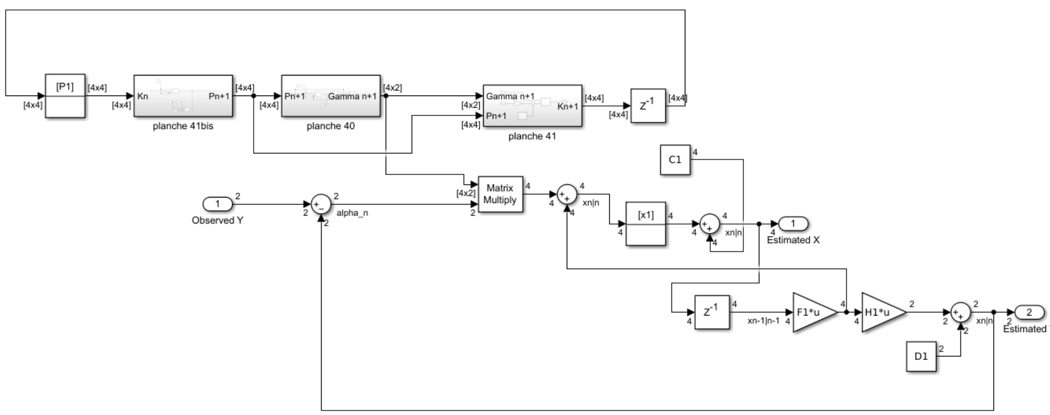
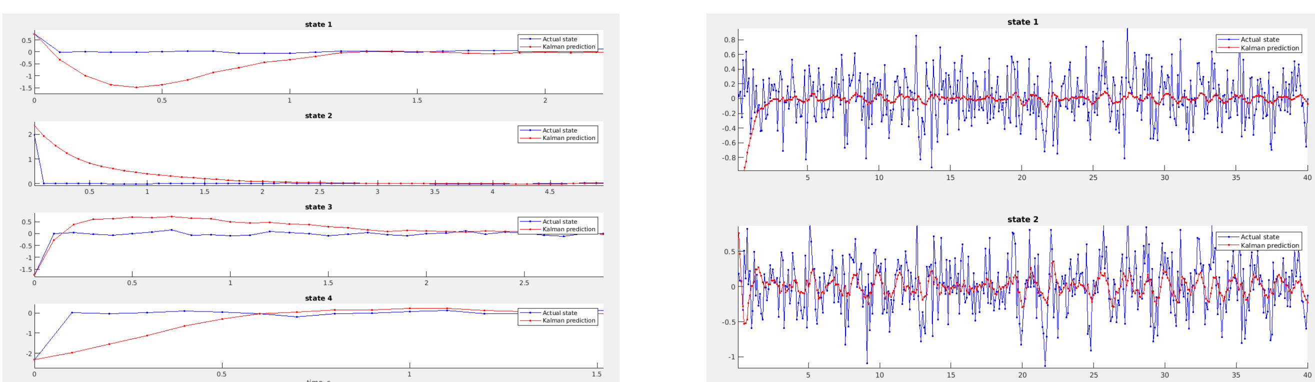


Fig. 1: Overview of the final Kalman filter

A. Changing the initial state

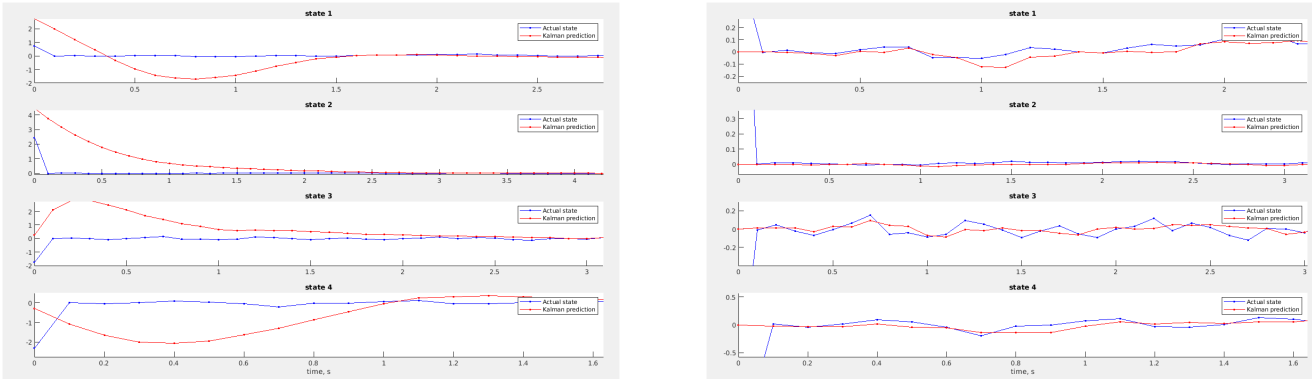
We compare 3 situations, with $x_1 = 0_{4,1}$, $x_1 = x_0$ and $x_1 = x_0 + 2$. We only plot the first values because the rest is similar through the experiments. The observation space will be similar throughout all experiments, with the noise of the signal being high enough to prevent the model from making useful prediction on the observation.



(a) State space

(b) Observation space

Fig. 2: Baseline with $P_1 = 0$, $x_1 = x_0$



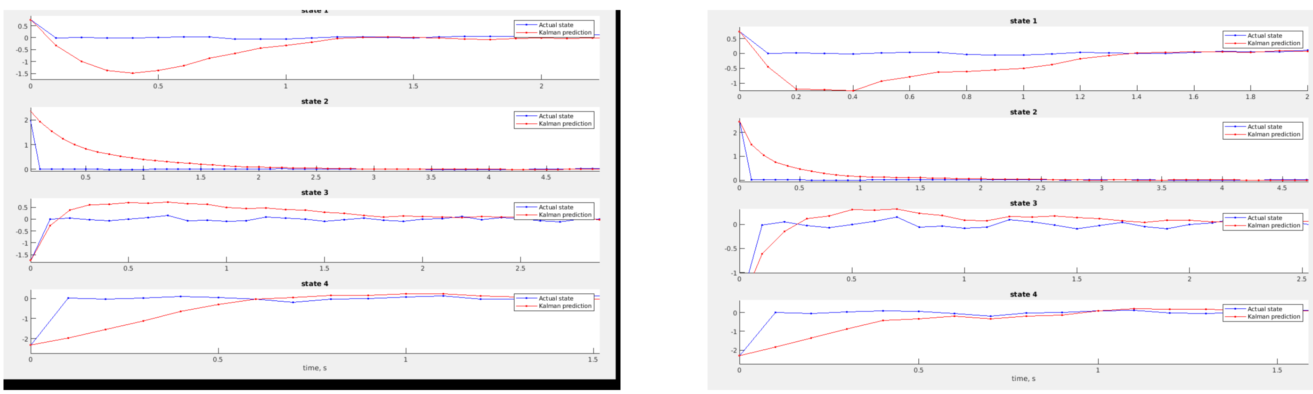
(a) $P_1 = 0$, $x_1 = 0$ (b) $P_1 = 0$, $x_1 = x_0 + 2$

Fig. 3: Different perturbed initial state

For all the coordinates, we have a faster convergence when the initial state is correctly instantiated. However, the model still converges if we initialize incorrectly.

B. Changing the initial state covariance

We tested several values for the initial state covariance. However, it seems that a too large initial value leads to instability in the matrix inversion (even if it is chosen to be a Cholesky inversion). We only test $P_1 \in \{0.3I_4, 0.5I_4\}$, and compare it to the baseline with $P_1 = 0_4$. We also only plot the first values.



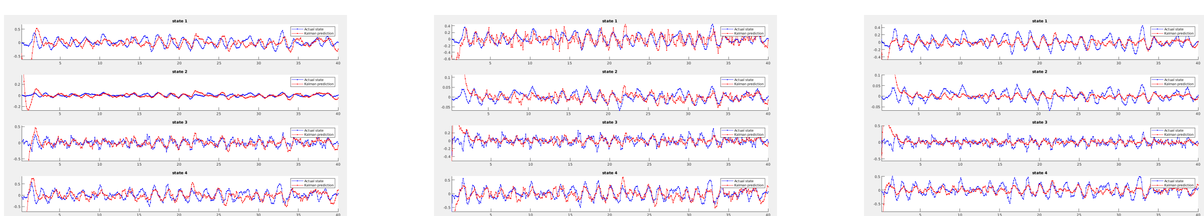
(a) $P_1 = 0.3I_4$ (b) $P_1 = 0.5I_4$

Fig. 4: Different values for the initial state covariance

We observe that a high value for the initial covariance leads to less bias (curve closer to the GT) but more variance (the curve is less smooth). I see this value of initial covariance as a confidence rate towards the data points. When the confidence is low, the filter relies less on the data points and more on the state model itself, which is why the estimation is smoother.

C. Uncertainties on the model

When adding a small noise ($\sigma^2 = 0.1$) to the values of the model matrix, we obtain the following curve:
If we increase the variance again, the model diverges, because the norm of F is already greater than 1.



(a) Adding noise to F (b) Adding noise to G (c) Adding noise to H

Fig. 5: Perturbation of the model matrices

For reference, we also plot the original behavior for the entire time span.

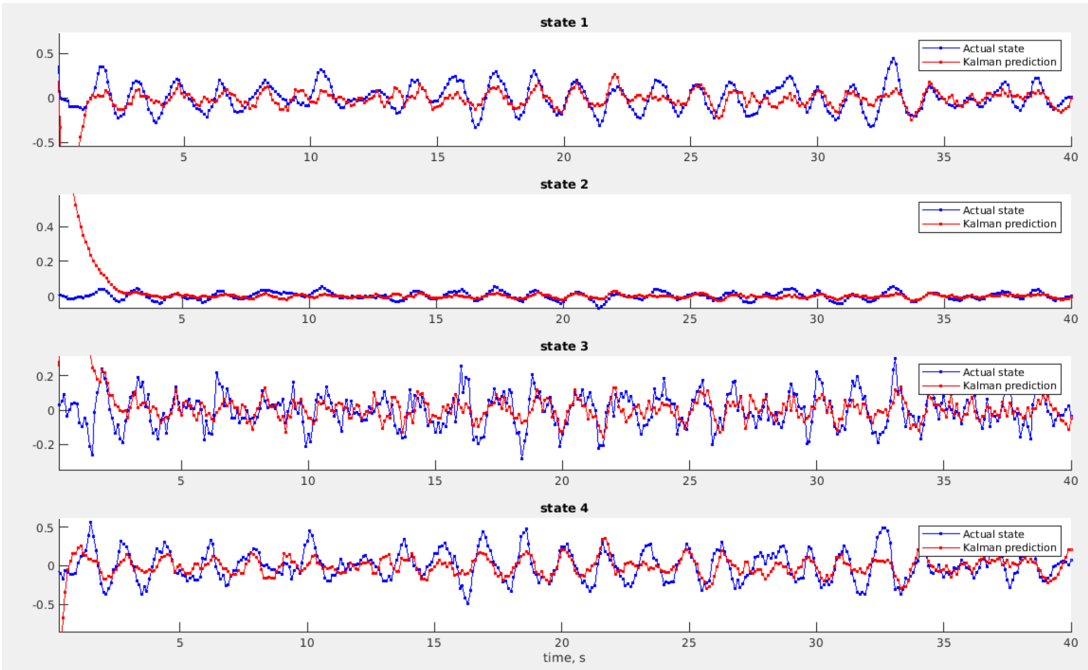


Fig. 6: Full view of the baseline - state space

III. Non-zero-mean Kalman model

Without changing the rest of the hypothesis, we now assume that $\mathbb{E}[\mathbf{U}_k] \neq 0$ and $\mathbb{E}[\mathbf{B}_k] \neq 0$.

A. Equivalent formulation with zero-mean noise

We then decompose the noise terms to obtained zero-mean noise. By setting $C_k = \mathbb{E}\left[G_k \mathbf{U}_k\right] = G_k \mathbb{E}\left[\mathbf{U}_k\right]$ and $D_k = \mathbb{E}\left[\mathbf{B}_k\right]$, U_k and B_k are now centered noise in the following equations:

$$\mathbf{X}_{k+1} = F_k \mathbf{X}_k + G_k \mathbf{U}_k + C_k$$
$$\mathbf{Y}_k = H_k \mathbf{X}_k + \mathbf{B}_k + D_k$$

Also note that in this equation, the new noise processes are the original processes minus their means.

B. Known expectation

We have $\mathbb{E}\left[\mathbf{X}_k\right] = F_k \mathbb{E}\left[\mathbf{X}_{k-1}\right] + C_k$. Also, because $\mathbb{E}\left[\mathbf{Y}_k\right] = H_k \mathbb{E}\left[\mathbf{X}_k\right] + D_k$, if we know $\mathbb{E}\left[\mathbf{X}_k\right]$, we know $\mathbb{E}\left[\mathbf{Y}_k\right]$. If we assume that we know $\mathbb{E}\left[\mathbf{X}_0\right]$, by induction, we can compute $\mathbb{E}\left[\mathbf{X}_k\right]$ and thus $\mathbb{E}\left[\mathbf{Y}_k\right]$ for all k.

C. Centered innovation process

We first recall the formula of the LLMSE estimator in our case:

$$\hat{\mathbf{X}}_{k|k} = \mathbb{E}\left[\mathbf{X}_k\right] + \mathbf{P}_{k|k} H_k^T \left(H_k \mathbf{P}_{k|k} H_k^T + R_k\right)^{-1} \left(\mathbf{Y}_k - \mathbb{E}\left[\mathbf{Y}_k\right]\right)$$

Taking the expectation of this expression, the term depending on the observation $\mathbf{Y}_k$ is zero. Then, we have: $\mathbb{E}\left[\mathbf{X}_k\right] = \mathbb{E}\left[\hat{\mathbf{X}}_{k|k}\right]$. This is also true if we replace the observation $\mathbf{Y}_k$ by the observation at step $k-1$.

By linearity of the expectation, we have $\mathbb{E}\left[\mathbf{Y}_k\right] = H_k \mathbb{E}\left[\mathbf{X}_k\right] + D_k = H_k \mathbb{E}\left[\hat{\mathbf{X}}_{k|k-1}\right] + D_k = \mathbb{E}\left[\hat{\mathbf{Y}}_{k|k-1}\right]$.

D. Equivalent space of the innovation errors

We again use induction to show that $\mathbf{Y}_k \in \text{ Vect}(\alpha_1, \dots, \alpha_n)$. We first recall the expression of the innovation process:

$$\alpha_k = \mathbf{Y}_k - \hat{\mathbf{Y}}_{k|k-1} \Leftrightarrow \mathbf{Y}_k = \alpha_k + \hat{\mathbf{Y}}_{k|k-1}$$

Thus, $\mathbf{Y}_0 = \alpha_0 + \hat{\mathbf{Y}}_{0|-1}$. If we choose $\hat{\mathbf{Y}}_{0|-1} = 0$, the property is true.

Then, $\mathbf{Y}_{k+1} = \alpha_{k+1} + \hat{\mathbf{Y}}_{k+1|k}$. Because $\hat{\mathbf{Y}}_{k+1|k}$ is itself a projection (in the sense of the LLMSE) on the observation space, times a matrix (H_k in our case), it is already in $\text{ Vect}(\alpha_1, \dots, \alpha_k) \subset \text{ Vect}(\alpha_1, \dots, \alpha_k, \alpha k + 1)$, the proof by induction is completed.

E. Expression of the estimators

To show the expression of $\hat{\mathbf{Y}}_{k|k-1}$, we project (6) on $\mathcal{Y}_{k-1}$, we get

- $H_k \hat{\mathbf{X}}_{k|k-1}$ for $H_k \mathbf{X}_k$ (by linearity)
- D_k is unchanged because it is a constant
- 0 for $\mathbf{B}_k$ because the noise at step k is independent from everything in $\mathcal{Y}_{k-1}$.

Similarly, because $\mathbf{U}_k$ is orthogonal to the observation space $\mathcal{Y}_k$, we get 0 when projecting equation (5) on the observation space, and the rest of the terms are obtained with the exact same arguments as previously. Thus,

$$\begin{aligned}\hat{\mathbf{Y}}_{k|k-1} &= H_k \hat{\mathbf{X}}_{k|k-1} + D_k \\ \hat{\mathbf{X}}_{k+1|k} &= F_k \hat{\mathbf{X}}_{k|k} + C_k\end{aligned}$$

F. Recursive Kalman Filter expressions

We write:

$$\begin{aligned}&\hat{\mathbf{X}}_{k|k} = \mathbb{E}\left[\mathbf{X}_k|\mathcal{Y}_{k-1}, \alpha_k\right] = \mathbb{E}\left[\mathbf{X}_k|\mathcal{Y}_{k-1}\right] + \mathbb{E}\left[\mathbf{X}_k|\alpha_k\right] \\ &= \hat{\mathbf{X}}_{k|k-1} + \Gamma_k \alpha_k = F_{k-1}\hat{\mathbf{X}}_{k-1|k-1} + C_{k-1} + \Gamma_k(\mathbf{Y}_k - \hat{\mathbf{Y}}_{k|k-1}) \\ &= \hat{\mathbf{X}}_{k|k-1} + \Gamma_k \alpha_k = F_{k-1}\hat{\mathbf{X}}_{k-1|k-1} + C_{k-1} + \Gamma_k(\mathbf{Y}_k - H_k\hat{\mathbf{X}}_{k|k-1} + D_k)\end{aligned}$$

Thus,

$$\hat{\mathbf{X}}_{k|k} = F_k \hat{\mathbf{X}}_{k|k} + C_k + \Gamma_k(\mathbf{Y}_k - H_k(F_{k-1}\hat{\mathbf{X}}_{k-1|k-1} + C_{k-1}) + D_k)$$

For the implementation, we can keep the fo llowing form (the difference from the classic formula being the addition of the mean of the noise):

$$\hat{\mathbf{X}}_{k|k} = F_{k-1}\hat{\mathbf{X}}_{k-1|k-1} + C_{k-1} + \Gamma_k \alpha_k$$

G. Generating the Kalman/Riccati Gains

To compute the Riccati gains, we have to compute $\text{Cov}(\mathbf{X}_k, \alpha_k)$ and the variance of the error $\text{Cov}(\alpha_k)^{-1}$:

For the first one, we can introduce the $\hat{\mathbf{X}}_{k|k-1}$ term since it is orthogonal to α_k (as an element of $\mathcal{Y}_{k-1}$), and won't change the value. Also, we use the orthogonality of B_k to the residual of the projection on the state space at step $k-1$.

$$\begin{aligned}&\text{Cov}(\mathbf{X}_k, \alpha_k) = \text{Cov}(\mathbf{X}_k - \hat{\mathbf{X}}_{k|k-1}, \alpha_k) = \text{Cov}(\mathbf{X}_k - \hat{\mathbf{X}}_{k|k-1}, H_k(\mathbf{X}_k - \hat{\mathbf{X}}_{k|k-1}) + B_k) \\ &\text{Cov}(\mathbf{X}_k - \hat{\mathbf{X}}_{k|k-1}, H_k(\mathbf{X}_k - \hat{\mathbf{X}}_{k|k-1})) = H_k\text{Cov}(\mathbf{X}_k - \hat{\mathbf{X}}_{k|k-1}, \mathbf{X}_k - \hat{\mathbf{X}}_{k|k-1}) = H_k \mathbf{K}_{k|k-1}\end{aligned}$$

For the second one, we develop α_k, still using B_k independence:

$$\begin{aligned}&\text{Cov}(\alpha_k) = \text{Cov}(H_k(\mathbf{X}_k - \hat{\mathbf{X}}_{k|k-1}) + B_k) = \text{Cov}(H_k(\mathbf{X}_k - \hat{\mathbf{X}}_{k|k-1})) + \text{Cov}(B_k) \\ &H_k\text{Cov}((\mathbf{X}_k - \hat{\mathbf{X}}_{k|k-1}))H_k^T + \text{Cov}(B_k) = H_k \mathbf{K}_{k|k-1} H_k^T + \mathbf{R}_k\end{aligned}$$

Finally, we derive the expression of $\mathbf{K}_{n+1|n}$, using U_k's independence:

$$\begin{aligned}&\mathbf{K}_{n+1|n} = \text{Cov}(\mathbf{X}_{n+1} - \hat{\mathbf{X}}_{n+1|n}) = \text{Cov}(F_n\mathbf{X}_n + G_n\mathbf{U}_n + C_n - (F_n\hat{\mathbf{X}}_{n|n} + C_n)) \\ &= \text{Cov}(F_n(\mathbf{X}_n - \hat{\mathbf{X}}_{n|n})) + \text{Cov}(G_n\mathbf{U}_n) = F_n\mathbf{K}_n F_n^T + G_n\mathbf{Q}_n G_n^T\end{aligned}$$

H. Implementation

To add the noise, we simply add a bias to the noises on the state space and the observation model. This bias is also added in the expression of the estimators. For consistency, we declare C and C_1 for the state space and D and D_1 for the observation space. Note that we will also test the case where $C \neq C_1$ and $D \neq D_1$

For the values of the biases, they are drawn from a normal distribution (but will always be the same due to the seed).

We again show the baseline results, and compare it with perturbed versions of the filter:

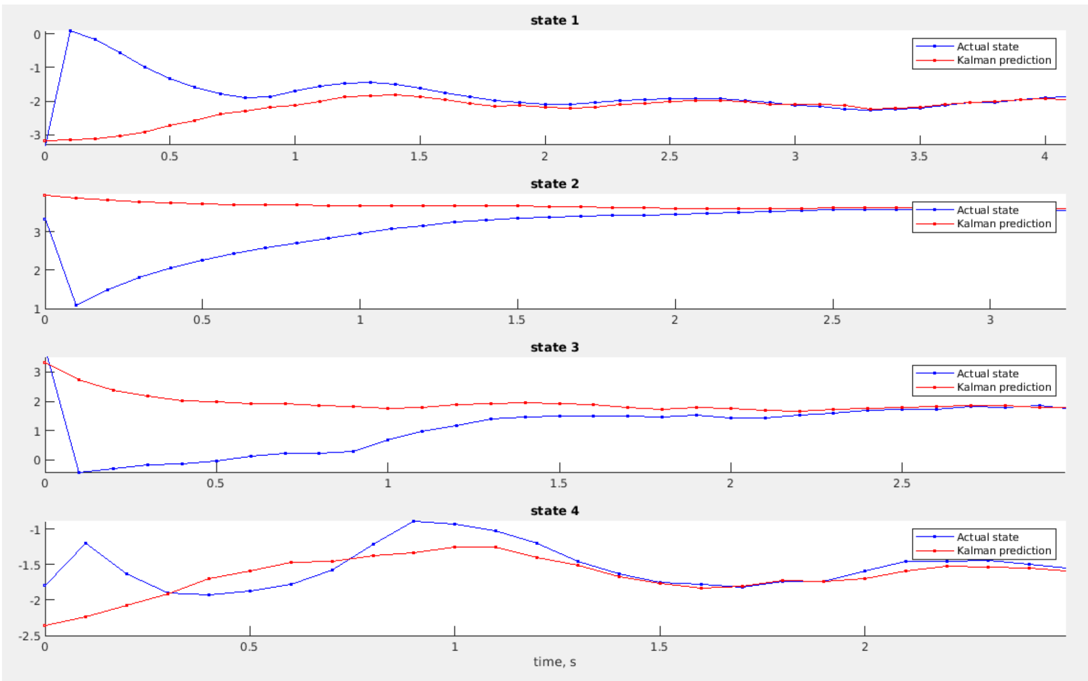
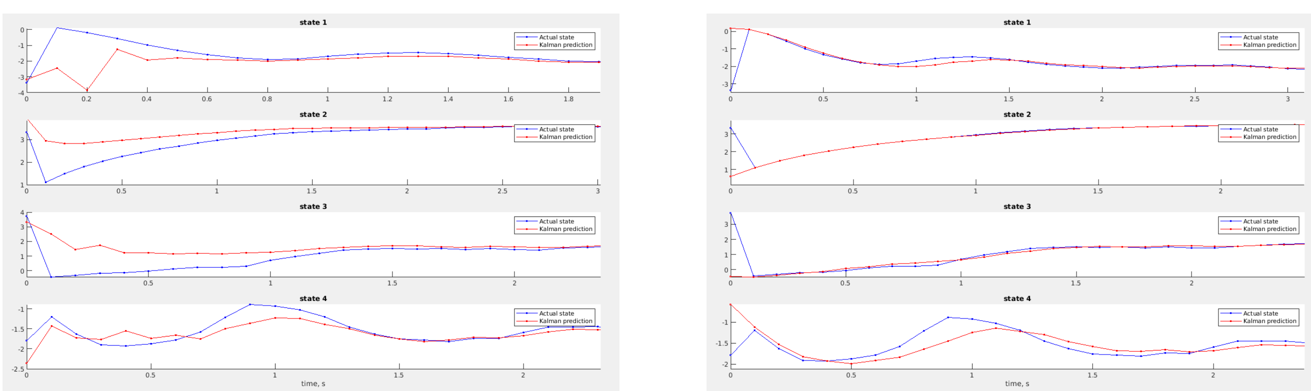


Fig. 7: Baseline for non-zero mean noise

We then change the initial conditions (initial state covariance and initial state.)

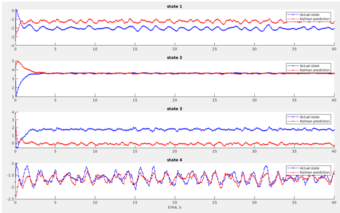


(a) $P_1 = 0.5I_4$, $x_1 = x_0$

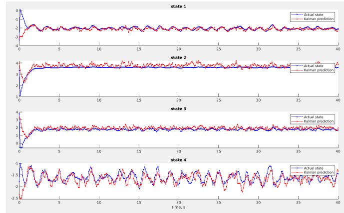
(b) $P_1 = 0_4$, $x_1 = 0$

Fig. 8: Changed initial conditions

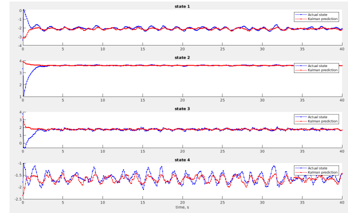
Similarly to the zero-mean case, a higher initial covariance produces a less smooth estimation, and a different initial state delays convergence. We then modify the evolution matrices F, G, H.



(a) Adding noise to F



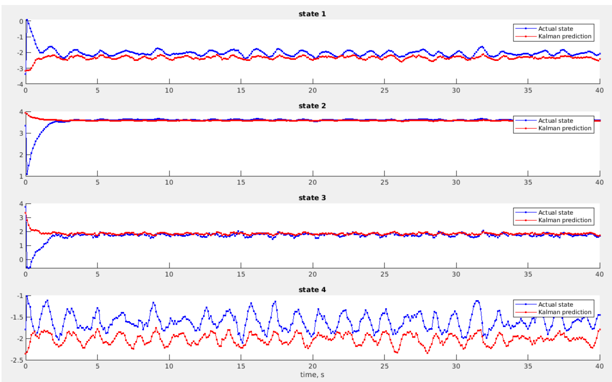
(b) Adding noise to G



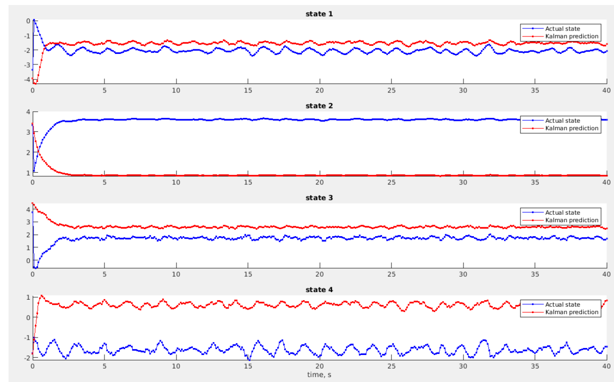
(c) Adding noise to H

Fig. 9: Perturbation of the model matrices

Here, the model is sensitive to this change in the sense that even the mean of the state can be misevaluated by the model, especially when changing F. Note that the perturbation is random and of small variance to avoid divergence of the model. We then change the means of the noise. We introduce a normal perturbation (of variance 1) to the means to test robustness.



(a) $C_1 \neq C$



(b) $D_1 \neq D$

Fig. 10: Changed mean of the noise

Here too, the mean of the state can be wrongly evaluated, but given that the perturbation is important (the same order of magnitude as the value of the mean itself), the fact that some coordinates still manage to converge is notable.

IV. LLMSE

In this final section, we implement the recursive linear least mean squared error estimator to solve the estimation of θ in:

$$\mathbf{x} = \mathbf{H}\theta + \mathbf{w}$$

To solve this problem, we use the following equations:

$$\begin{aligned}
\hat{\theta}_n &= \hat{\theta}_{n-1} + \mathbf{K}_n \left(x_n - \mathbf{h}_n^T \hat{\theta}_{n-1} \right) \\
\mathbf{M}_n &= (I - \mathbf{K}_n \mathbf{h}_n^T)\mathbf{M}_{n-1} \\
\mathbf{K}_n &= \frac{\mathbf{M}_{n-1}\mathbf{h}_n}{\sigma_w^2 + \mathbf{h}_n^T \mathbf{M}_{n-1} \mathbf{h}_n}
\end{aligned}$$

The corresponding Simulink model is:

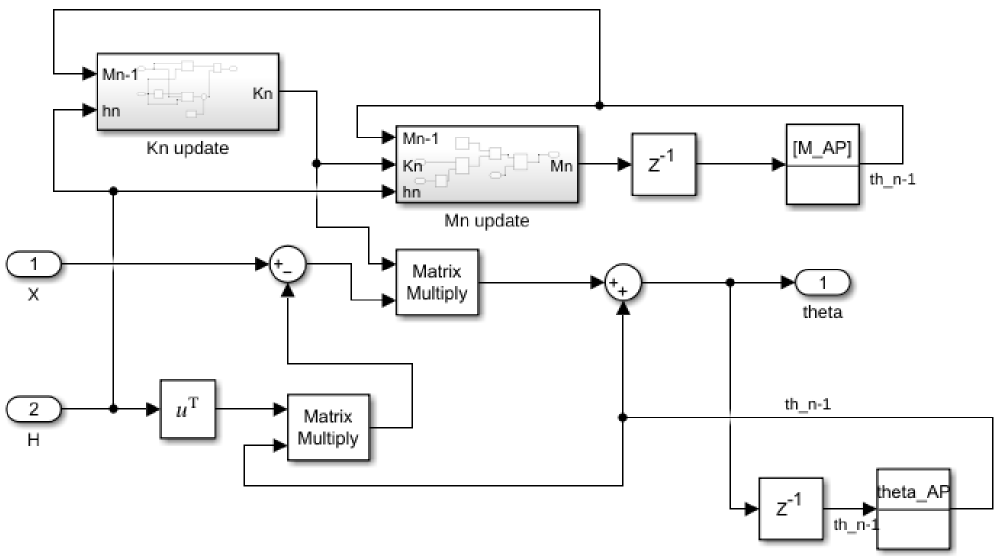
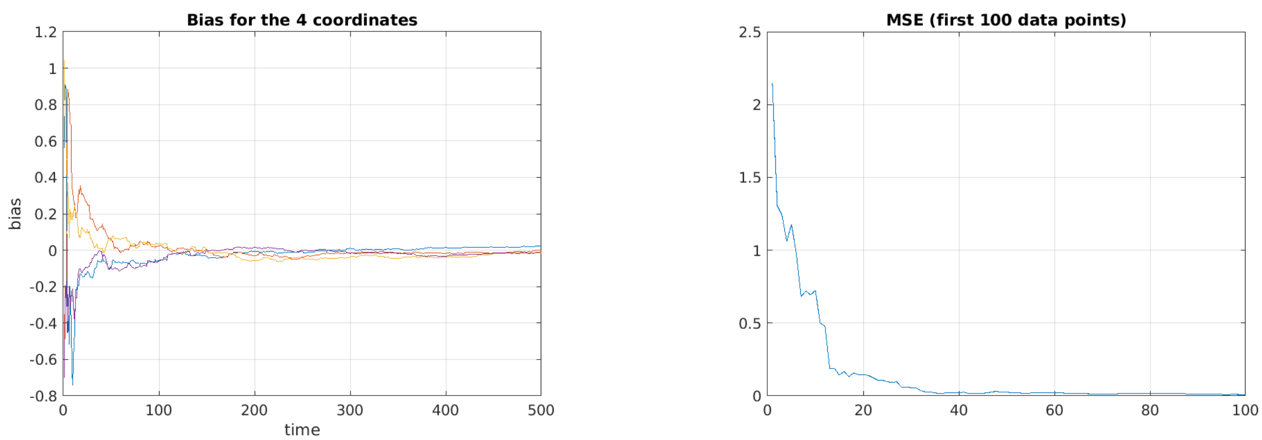


Fig. 11: Simulink implementation of the recursive LLMSE

After simulation, we get the following per-coordinate bias and mean squared error for the estimation of θ.



(a) Per-coordinate bias

(b) MSE

Fig. 12: Errors of the RLLMSE

We note that the mean squared error converges rapidly towards 0, and consequently only plot the first 100 data points.

*   *   *