Lab 1: Random Variables & Kalman Filter

Autonomous Vehicles - Ecole Centrale Nantes

Justin Lee
Student ID: 200406B
justin.lee@eleves.ec-nantes.fr

Steven Palma Morera
Student ID: 200443K
steven.palma-morera@eleves.ec-nantes.fr

***Abstract*—This document details the simulation of a DC Motor model and the design and testing of a normal Kalman Filter and a stationary Kalman Filter when the model parameters change and its variance varies.**

***Index Terms*—DC motor, Kalman Filter, MATLAB**

I. Introduction

In the second part of the lab, we simulate a DC motor model and used a normal and stationary kalman filter in order to predict the state of the motor based off the measurements being fed into them. All the images shown are of own elaboration and were obtained by testing the lab using Windows 10, MATLAB 2020 on December 6th, 2020. The full repository of the lab can be found here.

II. Simulation Design usign MATLAB

A. Kalman Filter

1) System Modeling: The objective of this section of the lab is to study the effects that different parameters of the Kalman Filter algorithm can have. Therefore, it is necessary to first create a simulated system where we will test the filters designed. From [1], the angular position of a rotor and its angular velocity of a DC motor is modeled as a state-space representation, given by equation 1 and 2. Where $x(t) = \begin{bmatrix} \theta(t) \\ \omega(t) \end{bmatrix}$ and $u(t)$ is a zero-mean square wave of duration D with period Δ, a peak-to-peak amplitude A and sampled at a rate of T_s. The angular position of the rotor (θ) is measured by an incremental encoder with a precision L angles per lap.

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & -1/T \end{bmatrix} x + \begin{bmatrix} 0 \\ G/T \end{bmatrix} u \tag{1}$$

$$\theta = \begin{bmatrix} 1 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \end{bmatrix} u \tag{2}$$

The previous two equations and the generation of the input signal can be found in the MATLAB scripts called "inputvoltage" and "simulate", where they are modeled as a discrete system. The values used for each variable can be found in table I. We will treat the x output of the "simulate" script as the ground truth for the states of the system and the output y as the measurements to be used in the Kalman filter simulation.

Parameter	Value
A	0.1 V
Δ	100 ms
D	500 ms
T_s	1 ms
L	512
G	50 $\frac{rad}{sV}$
T	20ms

TABLE I
Parameter values

2) Kalman Filtering: Two types of Kalman filter are developed, the standard kalman filter and the static kalman filter. A Kalman Filter is just a Bayesion Filter in which a the model is considered as linear and the noise is assumed to be normal, so it follows equations 3 and 4.

$$Y[n] = H_n x[n] + h_n + w[n] \tag{3}$$

$$X[n+1] = F_n x[n] + f_n + v[n] \tag{4}$$

We get then an equivalent Markov model of the system with input u_n and output y_n as shown in 5 and 6. Where, for this scenario, A_d,B_d,C_d and D_d are the discrete matrices of the state-space representation shown in equations 1 and 2. Also, w_n is assumed as a quantization noise for the measurement of the incremental encoder with variance R, and V_n is a white noise with variance Q that takes into account modeling errors.

$$Y[n] = C_d x[n] + D_d u[n] + w[n] \tag{5}$$

$$X[n+1] = A_d x[n] + B_d u[n] + v[n] \tag{6}$$

From this, the two Kalman Filter algorithms from [2] are designed in the MATLAB scripts called "kat" and "stat_kat". The objective then is to estimate the state variables of the system (x) from the measurements of the encoder (y) and input signal (u). The Kalman Filter algorithms will try to get this estimation $\hat{x}$ as close as possible to the ground truth x; however, their performance and behaviour will depend on different parameters.

The variance R, since it is a quantization error, it is modeled as an uniform random variable of value $\frac{2\pi}{12L}^2$, since the encoder goes from 0 to 2π. While the variance Q is changed constantly between 0.0001 (referred as small q) and 0.005 (referred as big q). Also, the initial condition of the Kalman filter is a design decision where it is assumed that the motor is initially stopped ($\omega_0 = 0$) but there is no information of the initial angular position ($\theta_0 = 0$). This leads to equation 7 for the initialization values.

$$\hat{X}_{1/0} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad P_{1/0} = \begin{bmatrix} \frac{2\pi}{12L}^2 & 0 \\ 0 & 0 \end{bmatrix} \tag{7}$$

III. Results & Analysis

A. Kalman Filter

1) System Modeling: For the system modeling part, Fig.1, Fig.2, Fig.3 and Fig.4 were obtained. Fig.1 shows the input of the system generated. It can be seen that it is a square wave that follows the specifications from table I. Fig.2 and Fig.3 shows the θ and ω signal, respectively, resulted from the system modeled in equations 1 and 2 when the input generated is fed into it. Finally, Fig.4 shows the simulated measurements of the state variable θ that are going to work as the input for the Kalman Filter algorithms.

Fig. 1. Input U generated

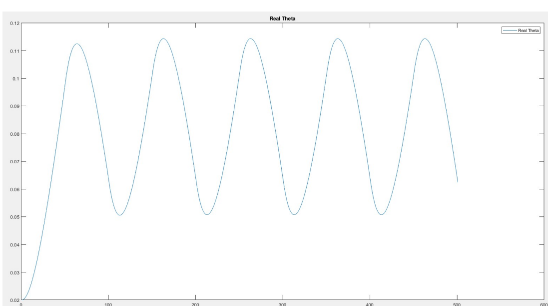

Fig. 2. Ground truth for θ

The results obtained follow precisely their expected behaviour; therefore, the implementation of the MATLAB scripts "inputvoltage" and "simulate" is correct.

2) Kalman Filtering: For testing the implementations and behaviours of the Kalman Filters the following methodology is applied: First, the initial $\hat{\theta}_{1/0}$ is around -0.05 and +0.05. Second, the variance Q is toggled between two values, 0.0001 and 0.005. Third, the filters are first tested considering that the modeling of the system in the filter is perfect and then they are also tested considering there exists a discrepancy in a variable value of the model T_s between the model simulated

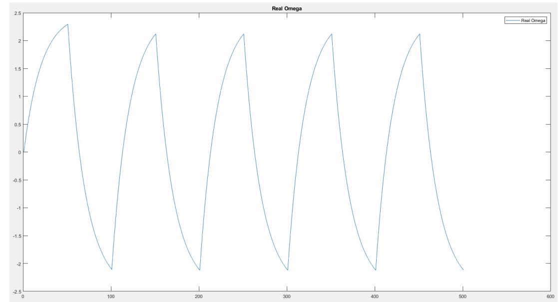

Fig. 3. Ground truth for Ω

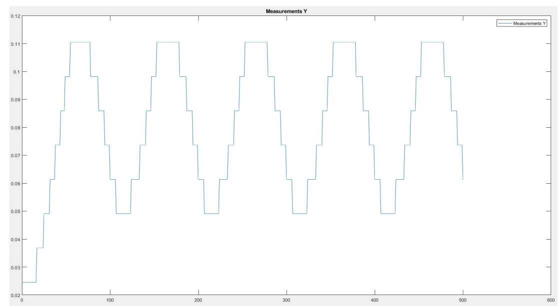

Fig. 4. Simulated measurements of θ

and the model used for the filtering. This methodology can be found in the MATLAB script called "Question2". Notice that along the experiment, the value of R - that is, the variance of the measurement - remains constant. Therefore, each kalman filter (standard and static) is tested four times: a perfect system with a small q, a perfect system with a big q, a rough system with a small q and a rough system with a big q.

The figures for the different experiments are shown from Fig. 5 to Fig.12:

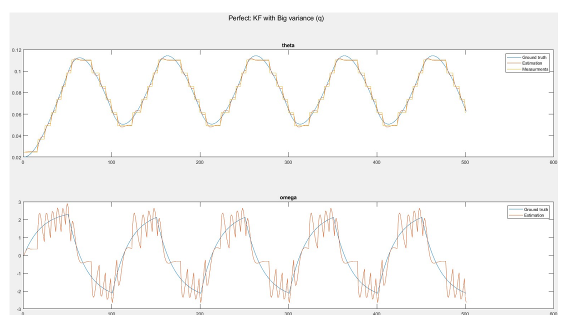

Fig. 5. KF: Perfect system with big variance (q)

When the normal Kalman filter is used on the perfect system with a big variance (Fig. 5), the filter tends to trust the measurement more so we get more flat lines in estimated theta and more erratic lines in omega. With a smaller variance (Fig. 6), we see that it trusts the model used for the filter more so the output of the filter looks like the model.

Also, notice that the stationary kalman filter implementation takes longer at the very beginning to match either the

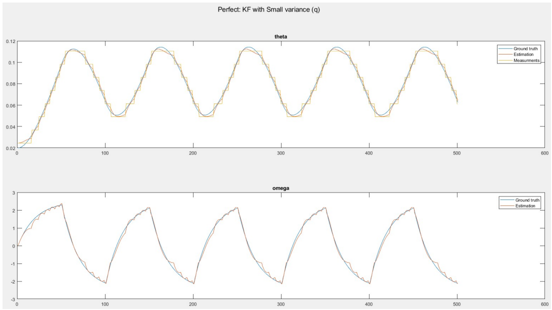

Fig. 6. KF: Perfect system with small variance (q)

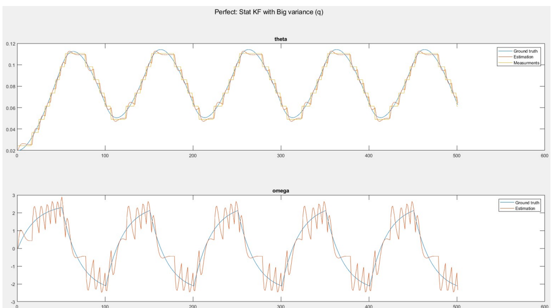

Fig. 7. Stat KF: Perfect system with big variance (q)

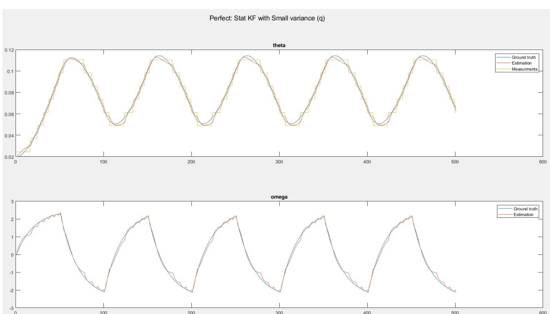

Fig. 8. Stat KF: Perfect system with small variance (q)

measurement or the model (depending on the value of Q), this is because the stationary kalman filter gain K is an approximation when the time tends to infinity so at the initial time instants this approximation will hold well, while the normal kalman filter is faster to converge since it evaluates the parameters in each iteration even in the starting instants.

In addition, notice that even with a rough system and a big variance (Fig. 9), the Kalman Filter is capable of getting a good estimation of the state variable θ. In all the figures shown, the estimation of the state variable θ is usually closer to the ground truth than the state variable ω. This is because there is a direct measure for the first so the filter can do the estimation considering the model and the measurement. This is not the case for the second, since its estimation is always a mathematical result from the estimation of the first, without a measurement to compare with.

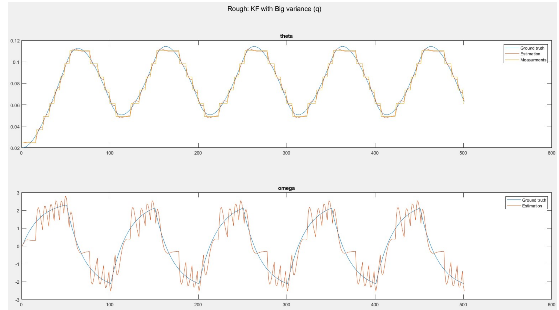

Fig. 9. KF: Rough system with big variance (q)

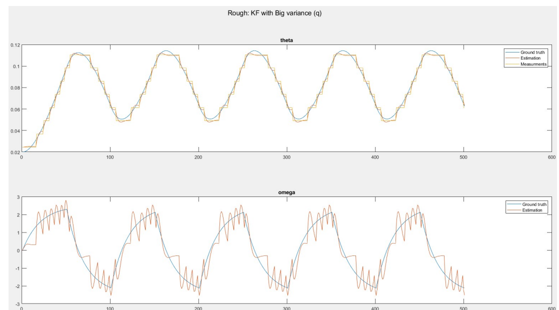

Fig. 10. KF: Rough system with small variance (q)

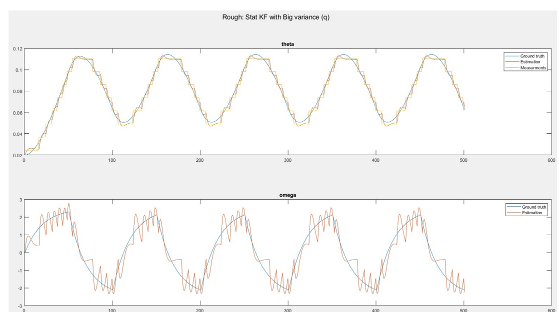

Fig. 11. Stat KF: Rough system with big variance (q)

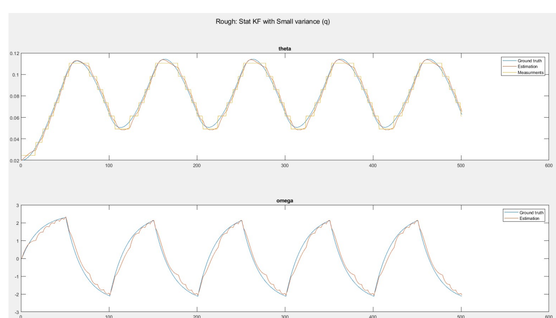

Fig. 12. Stat KF: Rough system with small variance (q)

IV. CONCLUSIONS

- The Kalman Filter is an effective method of estimating the state of a model when there are certain variances with the model and the measurement
- The estimation from the normal Kalman Filter is a balance between the prediction value from the model and the update value from the measurement. The estimation will be closer to the one which has a higher confidence level in each iteration, while for the stationary Kalman Filter this balance won't change.
- The normal kalman filter has better performance in the beginning, but the stationary kalman filter has better performance as time goes on, as its kalman gain is calculated at t = ∞

REFERENCES

[1] Le Carpentier, E. (2020). Lab work. Retrieved from: https://hippocampus.ec-nantes.fr/mod/folder/view.php?id=12398

[2] Le Carpentier, E. (2020). Statistical Signal Processing and Estimation Theory.