Physical Human Robot Interaction

Filippo Grotto VR460638

February 25, 2022

Contents

1 Introduction

In bilateral telemanipulation the aim is to allow to interact with a remote environment by using a joystic. The user manipulates the environment and perceives the reaction force through the haptic device (force feedback). The role of the control is to guarantee and/or enhance the coupling characteristics between the user at the master side and the environment at the slave side. Several components plays a role in a bilateral telemanipulation architecture: human operator, haptic device, master controller, communication channel (using a proper protocol), slave controller, slave robot and environment. A crucial part is the communication channel that has to be robust to delays and packet losses.

The hybrid two-port representation that we are going to use are the following

$$\begin{bmatrix} f_m \\ \dot{x}_m \end{bmatrix} = \begin{bmatrix} H_{11}(s) & H_{12}(s) \\ H_{21}(s) & H_{22}(s) \end{bmatrix} = \begin{bmatrix} \dot{x}_s \\ -f_s \end{bmatrix} \qquad \text{Hannaford hybrid matrix}$$

$$\begin{bmatrix} f_m \\ -\dot{x}_s \end{bmatrix} = \begin{bmatrix} \bar{H}_{11}(s) & \bar{H}_{12}(s) \\ \bar{H}_{21}(s) & \bar{H}_{22}(s) \end{bmatrix} = \begin{bmatrix} \dot{x}_m \\ f_s \end{bmatrix} \qquad \text{Lawrence hybrid matrix}$$

where

- $\bar{H}_{11}(s)$ is the uncontrained movement impedance the equivalent inertia and damping that the operator feels moving the master robot if the slave is in free motion. It should be as low as possible.

- $\bar{H}_{21}(s)$ position tracking during unconstrained motion: ability of the slave robot to follow the position of the master robot. It should tend to unity, if no position scaling is desired, with infinite bandwidth.
- $\bar{H}_{12}(s)$ tracking of forces in contact tasks when the operator keeps the master steady against the forces that the slave encounters. It should tend to unity, if no force scaling is desired, with infinite bandwidth.
- $\bar{H}_{22}(s)$ contact admittance: position tracking during contact tasks.

The key concepts of an architecture are:

- **Stability**: meaning that all the variales within the teleop system are bounded
- **Transparency**: meaning that the operator at the master side has the feeling to interact directly with the remote environment at the slave side.
- **Telepresence** denotes a dynamic behavior in which the environmental effects experienced by the slave are transferred through the master to the human without alteration

In order to have **perfect transparency** we want to have:

$$\begin{bmatrix} H_{11}(s) & H_{12}(s) \\ H_{21}(s) & H_{22}(s) \end{bmatrix} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \qquad \text{Hannaford hybrid matrix}$$

$$\begin{bmatrix} \bar{H}_{11}(s) & \bar{H}_{12}(s) \\ \bar{H}_{21}(s) & \bar{H}_{22}(s) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \qquad \text{Lawrence hybrid matrix}$$

In general, it would be enough to obtain good transparency at low frequencies, so the operator can accurately determine stiffness while in contact with an environment, or determine payload inertia while in free motion. It is also good to define some performance indicies like **position tracking**, **force rendering** and **impedance coupling**. The higher the bandwidth of the transparency, the larger the degree of telepresence but in this case stability becomes a limiting factor in achievable bandwidths

2 Four channel bilateral teleoperation architecture

This architecture was described by Dale A. Lawrence in *Stability and Transparency in Bilateral Teleoperation*. There are 4 channeels meaning two signals from master to save and two signals from slave to master. However this architecturee caan be specialized to obtainother teleoperationschemees according to available sensors. It is based on some assumptions:

- No communication delays between slave side and master side
- Perfect knowledge of the master and slave robot dynamics
- Force and velocity (position) measurements at the master and slave side are available

Considering the conditions for perfect transparency of Hannaford hybrid matrix (or Lawrence) we can derive the following conditions:

$$\begin{cases} C_3C_2 = I \\ C_4 = -(Z_m + C_m) \\ C_1 = Z_s + C_s \\ C_2 = I \end{cases}$$

Moreover it is possible to set $C_2 < 1$ to reduce operator's fatigue or $C_2 > 1$ to increase operator's level of sensitivity.

Moreover if we consider inner force controller the perfect transparency requires:

$$\begin{cases} C_3 = 1 + C_{sf} \\ C_4 = -(Z_m + C_m) \\ C_1 = Z_s + C_s \\ C_2 = 1 + C_{mf} \end{cases}$$

Force and position feedback act in oppositve ways, in the sense that one softens and the other stiffens the seder device. When the slave is in contact with hard enviroment the contact force is the dominant signal for transmission, in from motion or soft environment the position/velocity is the dominant signal.

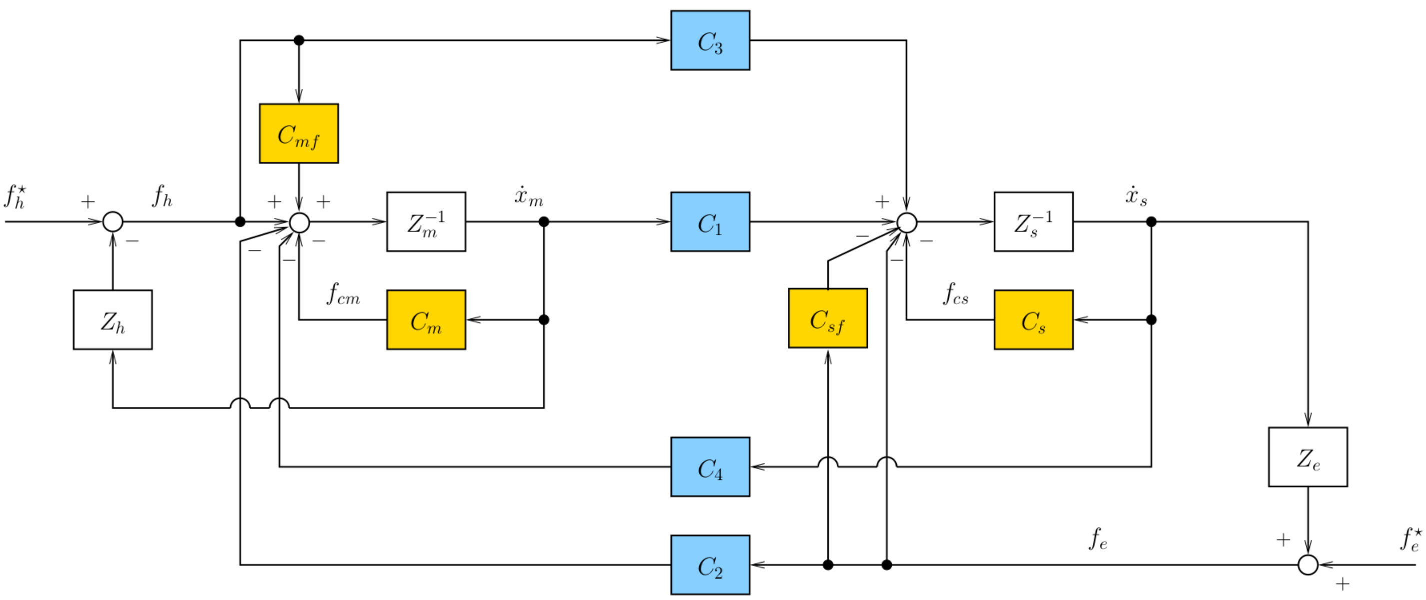


Figure 1: 4 Channel Bilateral Telemanipulation architecture with inner force loops C_{mf} and C_{sf}.

2.1 HW 1: Continuous and discretized implementation

Implement the SISO Four-channel bilateral teleoperation architecture with

$$C_m = B_m + \frac{K_m}{s} \quad C_s = B_s + \frac{K_s}{s}$$

$$Z_m^{-1} = \frac{1}{M_m s + D_m} \quad Z_s^{-1} = \frac{1}{M_s s + D_s}$$

where $Mm = 0.5$, $M_s = 2$. Moreover $D_s = 10$ and $D_m = 5$ or both zero in the initial case. In Fig 2 a simple plot of the positions and velocities (slave and master) are reported for proper selected tuning parameters of the related master and slave controller. In order to properly tune the controllers the following closed-loop systems were considered:

$$G_m = \frac{1}{M_m s^2 + B_m s + K_m} \quad G_s = \frac{1}{M_s s^2 + B_s s + K_s}$$

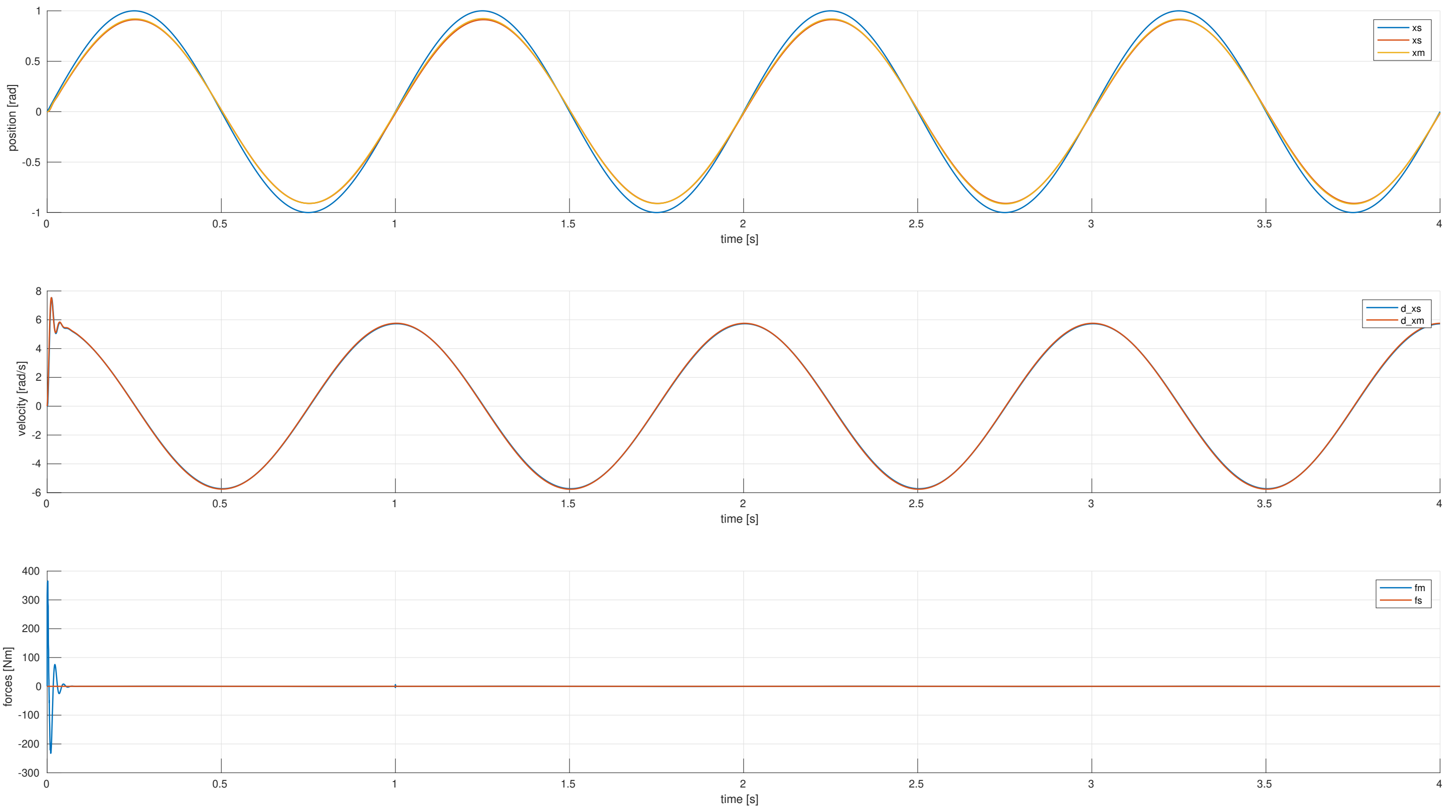


Figure 2: 4 channel architecture in free motion with a sinusoidal reference

The proper parameters were selected considering the step reponse of the two second order systems. For the human intention controller the parameters were selected by comparing the reference position with the master/slave position and perform proper tuning (an analytical closed-loop system might also be considered for this analysis).

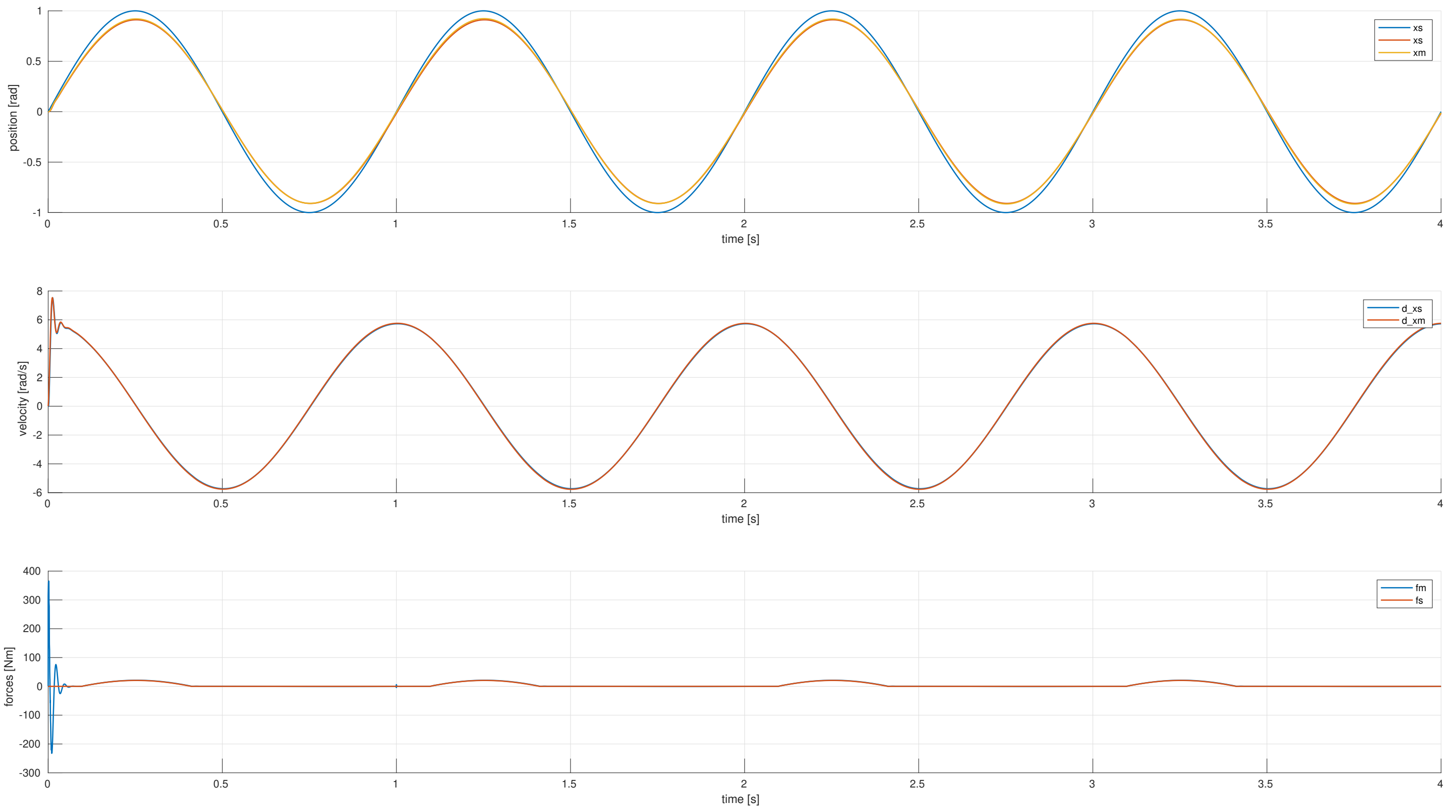


Figure 3: 4 channel architecture in contact at 0.5 with a sinusoidal reference

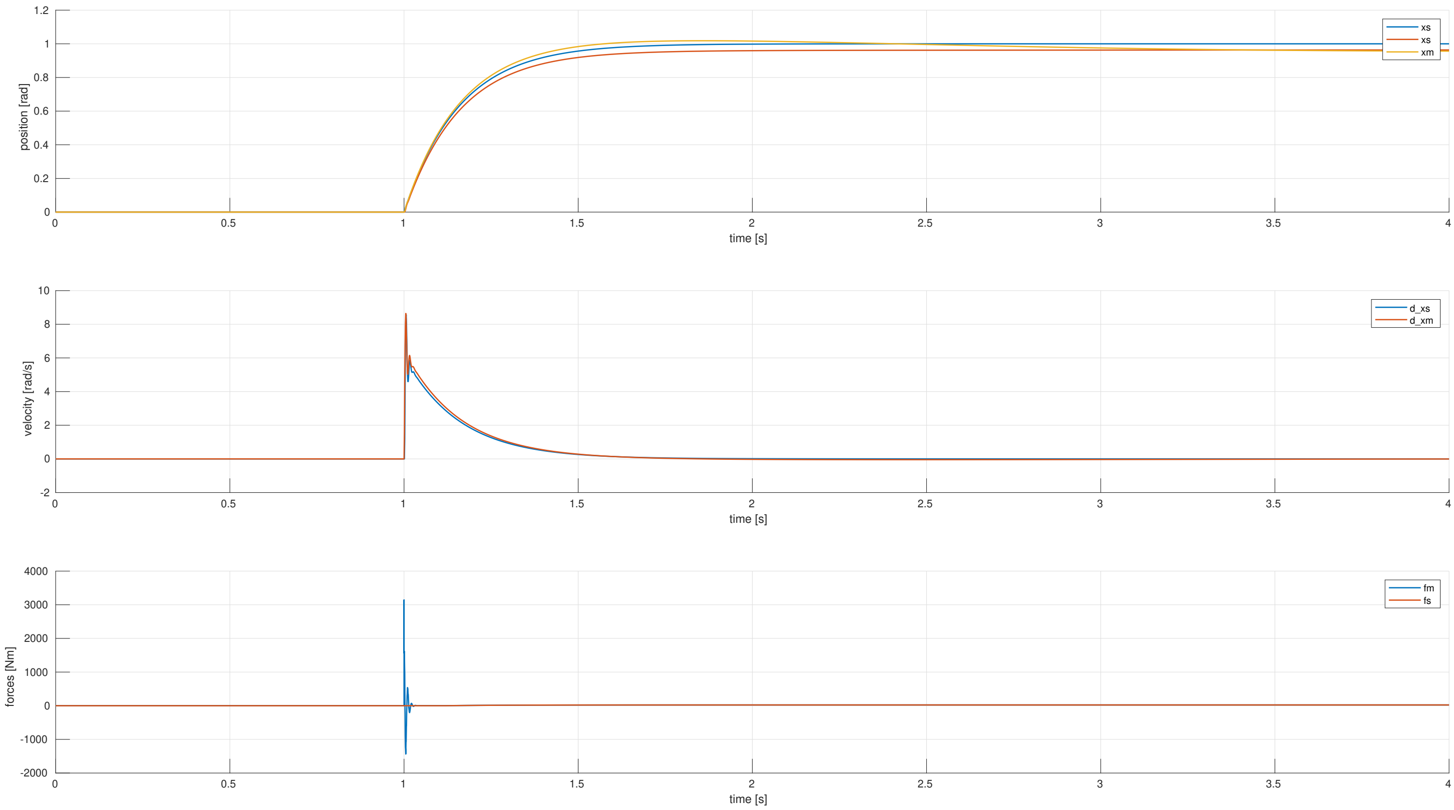


Figure 4: 4 channel architecture in contact at 0.5 with a step reference

As it is visible in Fig 2 the architecture is correctly following the reference provided. There is however a large effect due to the initial motion in all the presented figures due to high gain. In Fig 3 the contact interaction is clearly visible in the forces plot when the position reaches 0.5.

Finally the Fig 4 gives an idea of the reponse of the system to a step function and Fig 5 emphasises the problem related to estimation of velocity and accelerations with noisy measurements. Moreover the inner force feedback can be introduced as constants and be properly tuned as desired.

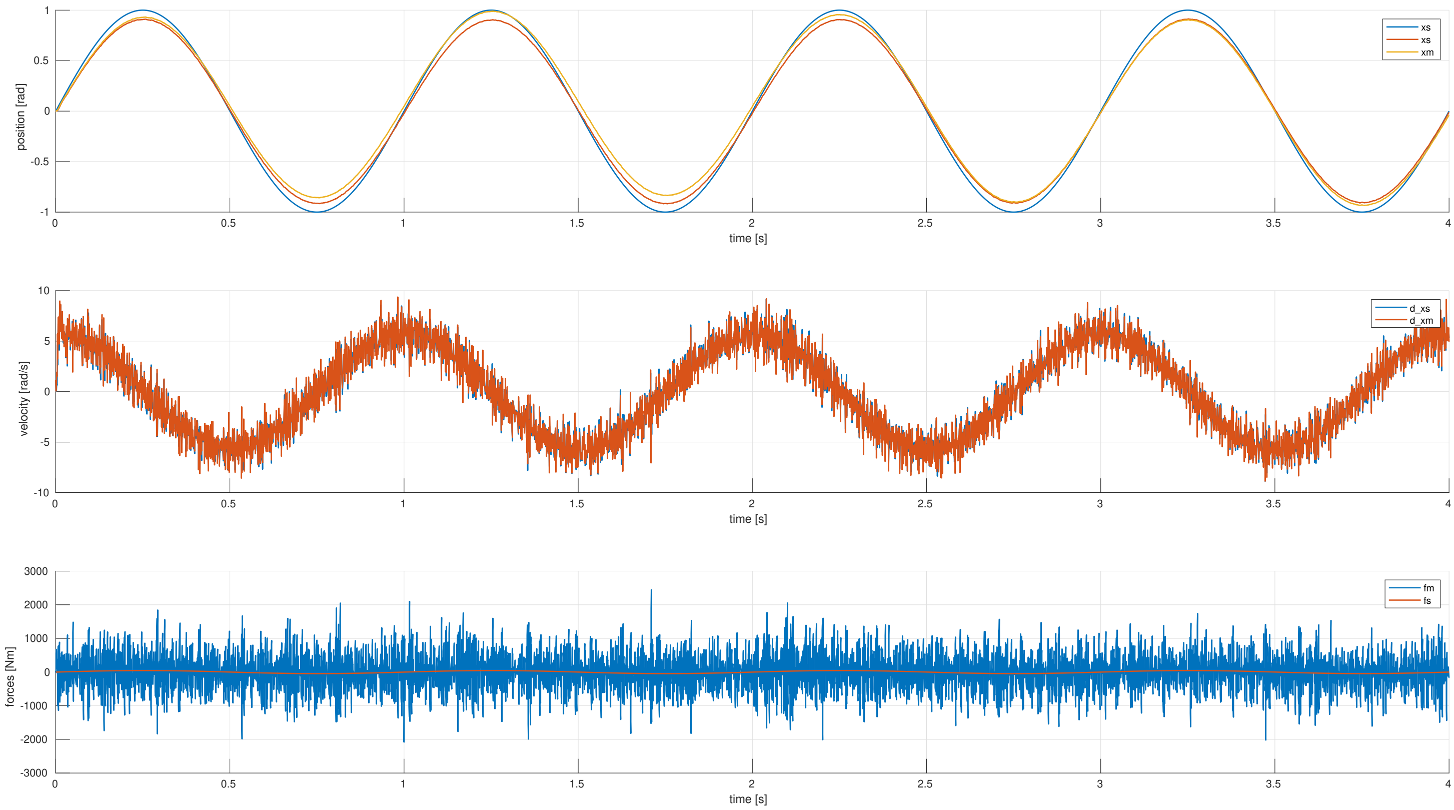


Figure 5: Four channel architecture with white noise

The entire architecture was translated in the related discretized version according to our specification in terms of encoders and the related derivative were performed using the simulink block. In the following section kalman filters will be introduced and they will be used to filter noisy estimations of velocities and accelerations.

2.2 HW 2: Derive hybrid matrix

Derive the hybrid matrix for the four channel bilateral teleoperation considering the inner force loop at the master and slave side. We can directly consider Fig 1.

$$\begin{bmatrix} f_m \\ -v_s \end{bmatrix} = \begin{bmatrix} \overline{H_{11}} & \overline{H_{12}} \\ \overline{H_{21}} & \overline{H_{22}} \end{bmatrix} = \begin{bmatrix} v_m \\ f_s \end{bmatrix}$$

Let's start by defining:

$$v_m = Z_{cm}^{-1}(f_m - C_2 f_s - C_4 v_s + C_{mf} f_m) \tag{1}$$

$$v_s = Z_{cs}^{-1}(-f_s - C_{sf} f_s + C_1 v_m + C_3 f_m) \tag{2}$$

Finally let's compute the 4 components of the hybrid matrix considering one component to zero at each step.

$$\overline{H_{11}} : f_m \to v_m \qquad f_s = 0$$

$$Z_{cm} v_m = f_m - C_4 Z_{cs}^{-1} C_1 v_m - C_4 Z_{cs}^{-1} C_3 f_m + C_{mf} f_m$$

$$v_m (Z_{cm} + C_4 Z_{cs}^{-1} C_1) = f_m (1 - C_4 Z_{cs}^{-1} C_3 + C_{mf})$$

$$\overline{H_{11}} = \frac{Z_{cm} Z_{cs} + C_1 C_4}{(1 + C_{mf}) Z_{cs} - C_3 C_4}$$

$$\overline{H_{12}} : f_m \to f_s \qquad v_m = 0$$

$$0 = f_m - C_2 f_s + C_4 Z_{cs}^{-1} C_{sf} f_s + C_4 Z_{cs}^{-1} f_s - C_{mf} f_m$$

$$f_m (1 - C_{mf} - C_4 Z_{cs}^{-1} C_3) = f_s (C_2 - C_4 Z_{cs}^{-1} C_{sf} C_{sf} - C_4 Z_{cs}^{-1})$$

$$\overline{H_{12}} = \frac{C_2 Z_{cs} - C_4 (1 + C_{sf})}{(1 + C_{mf}) Z_{cs} - C_3 C_4}$$

$$\overline{H_{21}} : -v_s \to v_m \qquad f_s = 0$$

$$v_s Z_{cs} = -(C_1 v_m + C_3 f_m)$$

$$v_s (Z_{cs} - \frac{C_3 C_4}{1 + C_{mf}} = -v_m (C_1 + \frac{C_3 Z_{cm}}{1 + C_{mf}})$$

$$\overline{H_{21}} = -\frac{C_1 (1 + C_{mf}) + C_3 Z_{cm}}{(1 + C_{mf}) Z_{cs} - C_3 C_4}$$

$$\overline{H_{22}} : -v_s \to f_s \qquad v_m = 0$$

$$v_s Z_{cs} = C_{sf} f_s + f_s + C_{sf} f_s$$

$$v_s (Z_{cs} - \frac{C_3 C_4}{1 + C_{mf}}) = f_s (C_{sf} + 1 - \frac{C_3 C_2}{1 + C_{mf}})$$

$$\overline{H_{22}} = \frac{(1 + C_{sf})(1 + C_{mf}) - C_2 C_3}{(1 + C_{mf}) Z_{cs} - C_3 C_4}$$

3 HW 3-4: Kalman Filter, Predictor and Smoother

In this section we are briefy going to consider the following estimation problems:

- **Filtering** estimate $\hat{s}(t)$ using measurements till time t
- **Prediction**: estimate $\hat{s}(t+h)$ using measurements till time t
- **Smoothing**: estimate $\hat{s}(t-h)$ using measurements till time t

Our aim is to estimate the velocities and accelerations from noisy position measurements. Some samples were provided for the task and the kalman filter and predictor were implemented as well as the steady state versions. In Fig **??** and in Fig **??** the estimations of velocities and accelerations are reported. The theory behind the kalman filter and predictor is not fully reported. Let's derive the discretized model for the position, velocities and accelerations to use in the kalman filter:

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} w(t) \qquad y(t) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} x(t) + v(t) \tag{3}$$

Let's discretize the continuous-time model:

$$A_d = e^{AT_s} = e^{\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} T_s} = \begin{bmatrix} 1 & T_s & \frac{T_s^2}{2} \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix}$$

$$C_d = C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

$$B_d = \left(\int_0^{T_s} e^{\begin{bmatrix} 1 & \tau & \tau^2/2 \\ 0 & 1 & \tau \\ 0 & 0 & 1 \end{bmatrix}} d\tau \right) = \begin{bmatrix} T_s^3/6 \\ T_s^2/2 \\ T_s \end{bmatrix}$$

It is important to notice the presence of two variance matrixes R related to v_k and Q related to w_k. The variance matrix R also called noise variance depends on the sensors whereas the variance matrix Q also caled model variance is chosen in order to explain the measurements as well as possible. They are crucial for proper tuning of the kalman filter.

Briefly the Kalman filter can be defined using a recursive formulation composed of a prediction and an estimation step with proper initial conditions:

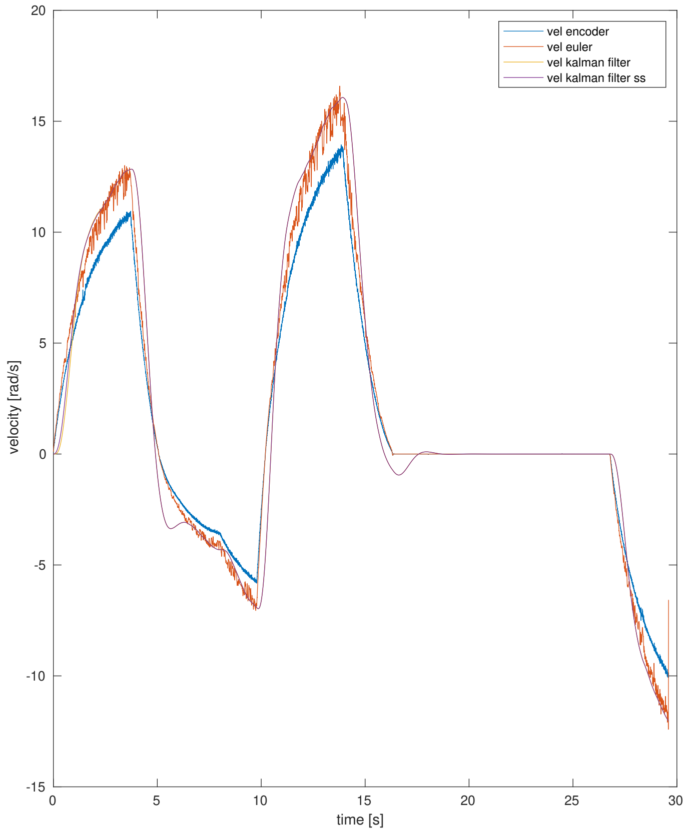
$$\hat{x}_{k+1|k+1} = A\hat{x}_{k|K} + K_{k+1}(y_{k-1} - CA\hat{x}_{k|k})$$

$$P_{k+1|k} = AP_{k|k-1}A^T - AP_{k|k-1}C^T(CP_{k|k-1}C^T + R)^{-1}CP_{k|k-1}A^T + Q \qquad \textbf{Ricati equation}$$

where the kalman gain maps the output estimation error into the correction of the prediction state

$$K_{k+1} = P_{k+1|k}C^T(CP_{k+1|k}C^T + R)^{-1} \qquad \textbf{Kalman gain}$$

when k goes to infinity we can consider the steady-state kalman filter which uses the Algebraic Ricati Equation (ARE) for getting P_∞. Informally the steady-state kalman filter is not optimal at the beginning of the experiments but we can reach reasonable results.



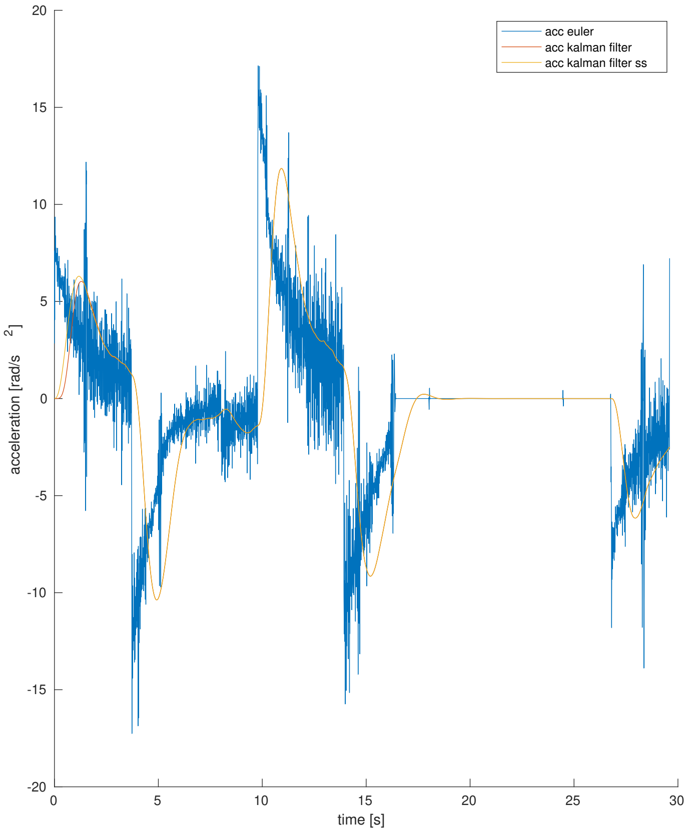


Figure 6: Velocities and accelerations estimations using Kalman filter, Kalman filter in steady state conditions and euler approximations with 5Hz low pass filter. As it is visible at the beginning the Kalman filter is different from the steady state version.

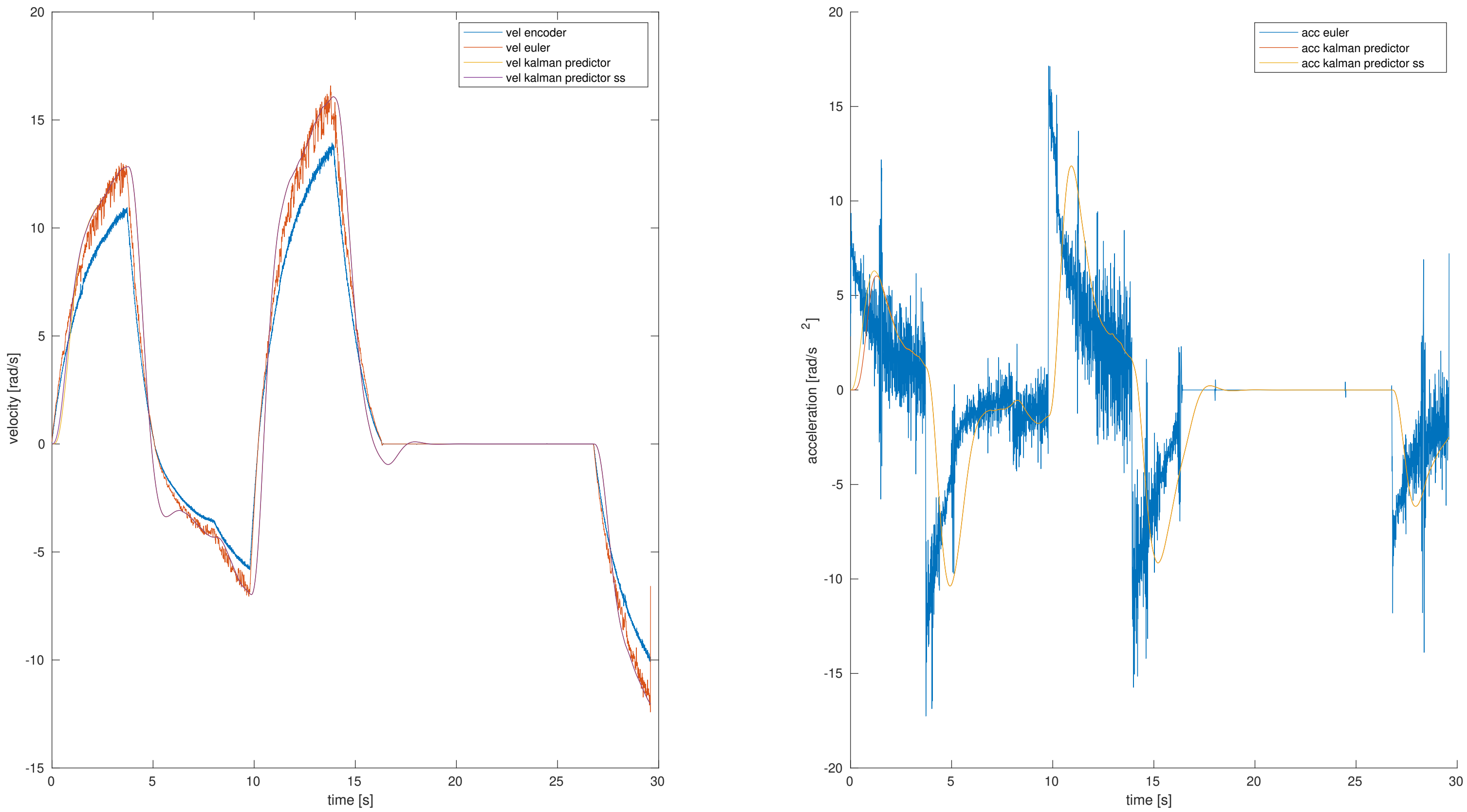


Figure 7: Velocities and accelerations estimations using Kalman predictor, Kalman predictor in steady state conditions and euler approximations with 5Hz low pass filter. As it is visible at the beginning the Kalman predictor is different from the steady state version.

In Fig 8 the use of the kalman smoother is reported. Informally, thanks to both forward (Kalman filter) and backward (smoothing) steps the Kalman smoother in general performs a better estimation for the given dataset.

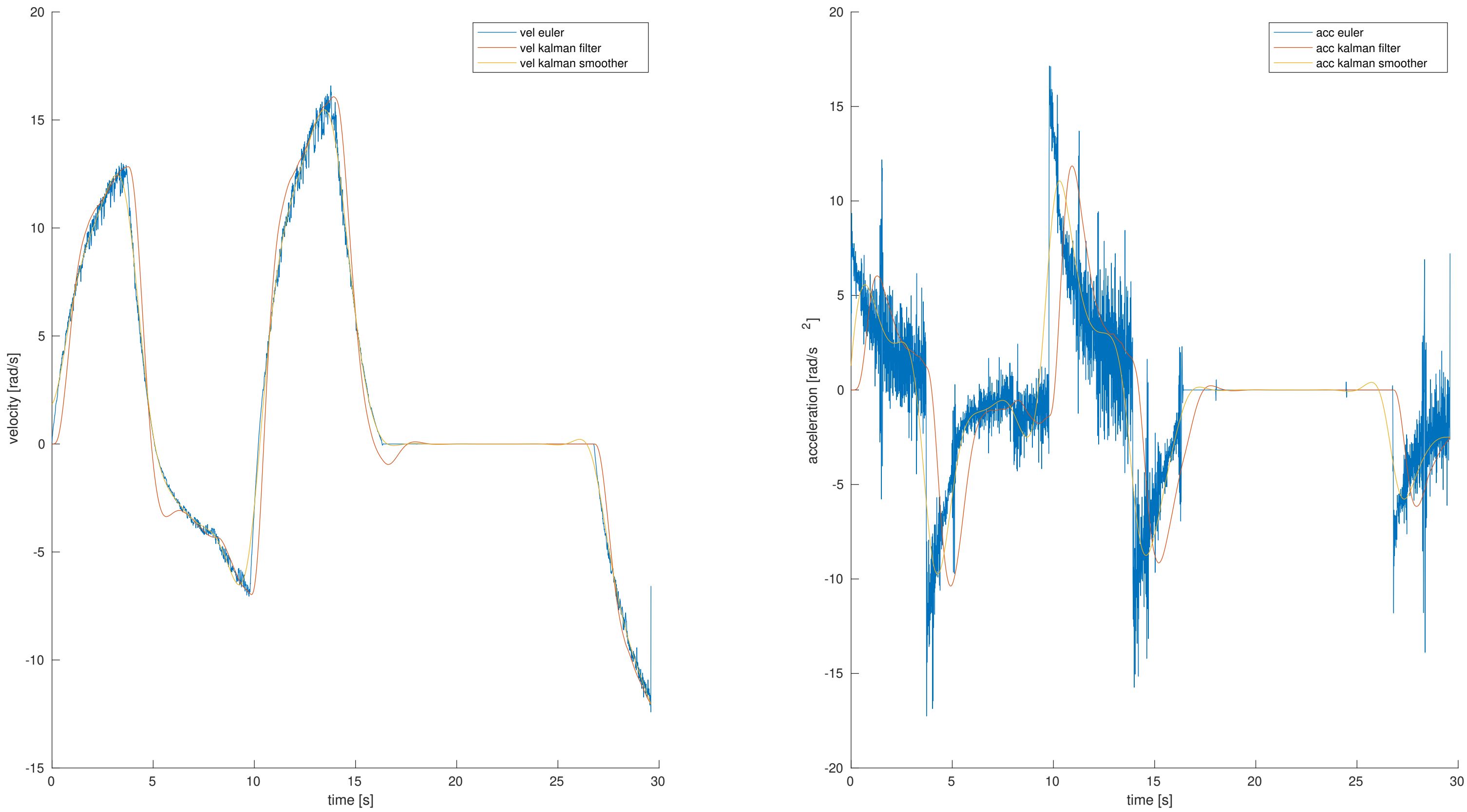


Figure 8: Velocities and accelerations estimations using Kalman filter and Kalman Smoother

4 HW 5: Least Square, RLS and Adaptive Estimation

The aim of this section is to present three techniques to solve a gray-box indentification problem meaning that the class of the model is known but the specific parameters are unknown. If we consider a linear model:

$$y = x\beta \qquad \textit{for example a DC motor} \qquad V(t) = \begin{bmatrix} \dot{w}(t) & w(t) \end{bmatrix} \begin{bmatrix} \tau/k \\ 1/k \end{bmatrix}$$

the aim is to find the prediction:

$$\hat{y} = x\hat{\beta}$$

where $\hat{\beta} \in \mathbb{R}^m$ is the matrix of coefficients that we have to determine.

The following methods are reported:

- **Least square solution** $\hat{\beta} = (X^T X)^{-1} X^T Y$ if $X^T X$ is nonsingular
- **Adaptive algorithm** $\hat{\beta}(k) = \hat{\beta}(k-1) + T_s g x^T(k) e(k)$ with $g > 0$. Basically we are moving in the opposite direction of the gradient of the minimized square error.

- **Recursive Least square** by solving the causal formulation:

$$\begin{cases} e(k) = y_k - x_k\hat{\beta}(k-1) \\ P(k) = \frac{1}{\lambda}\left(P(k-1) - \frac{P(k-1)x_k^T x_k P(k-1)}{\lambda + x_k P(k-1)x_k^T}\right) \\ K(k) = P(k)x_k^T \\ \hat{\beta}(k) = \hat{\beta}(k-1) + K(k)e(k) \end{cases}$$

where $0 < \lambda \leq 1$ is called forgetting factor

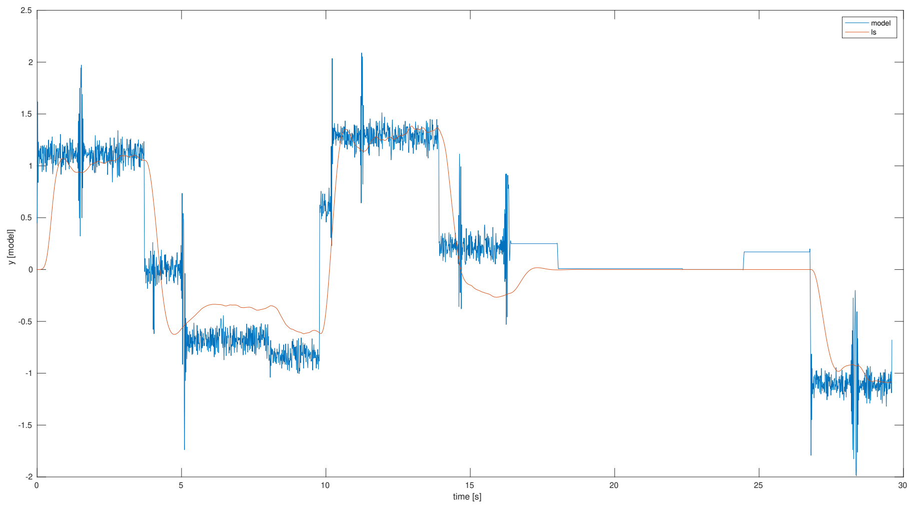


Figure 9: Least square estimation of the DC model of the motor

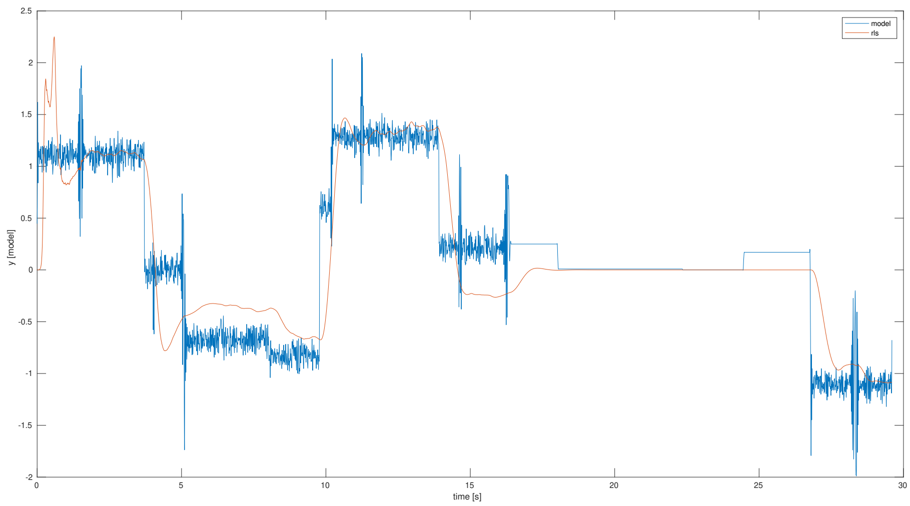


Figure 10: Recursive Least square estimation of the DC model of the motor

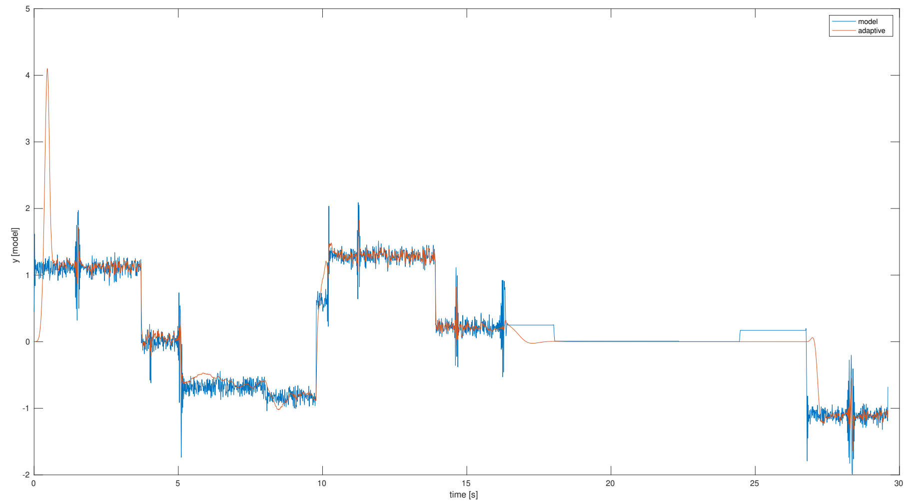


Figure 11: Adaptive estimation of the DC model of the motor

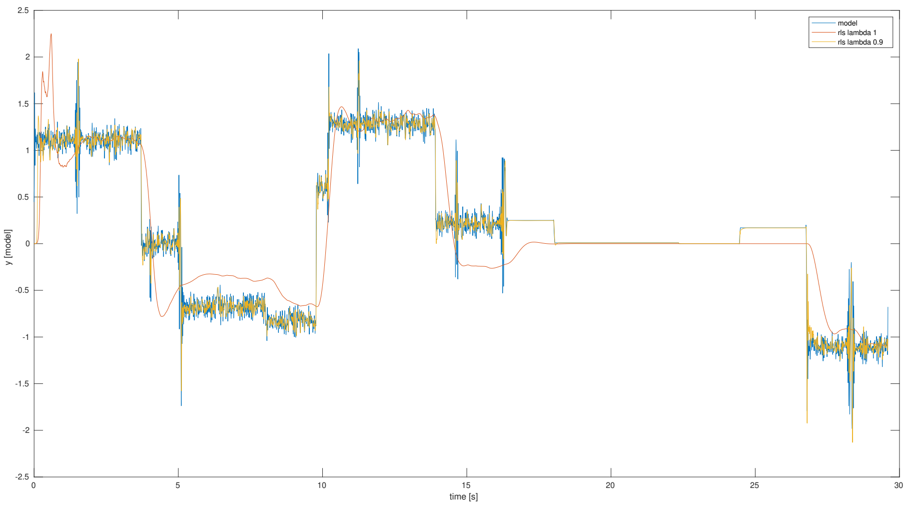


Figure 12: Recursive Least square with forgetting factor of the DC model of the motor