Final Assignment

Gilian Breysens (4575466), Akshat Srivastava (5007259),
Shuhul Razdan (5049768)

December 2019

Assignment Part 1

Task 1.1 : Fault Detection

The nominal dynamics of the plant are given as follows

$$x_1(k+1) = x_1(k) + \frac{T_s}{A_1}\left(c_1 A_1^p \operatorname{sign}\left(x_3(k) - x_1(k)\right) \times \sqrt{2g\left|x_3(k) - x_1(k)\right|} + u_1(k)\right) \tag{1}$$

$$x_2(k+1) = x_2(k) + \frac{T_s}{A_2}\left(c_2 A_2^p \operatorname{sign}\left(x_3(k) - x_2(k)\right) \times \sqrt{2g\left|x_3(k) - x_2(k)\right|} - c_3 A_3^p \sqrt{2g x_2(k)} + u_2(k)\right) \tag{2}$$

$$\begin{aligned} x_3(k+1) = x_3(k) + \frac{T_s}{A_3} c_1 A_1^p \operatorname{sign}\left(x_1(k) - x_3(k)\right) \times \\ \sqrt{2g\left|x_1(k) - x_3(k)\right|} - c_2 A_2^p \operatorname{sign}\left(x_3(k) - x_2(k)\right) \times \sqrt{2g\left|x_3(k) - x_2(k)\right|} \end{aligned} \tag{3}$$

The actual system runs with certain values of the parameters that are different from the assumed values due to manufacturing tolerance and other uncertainties. But these values do have an upper bound which could be deduced from the manufacturing information of the concerned unit.

It is convenient here to bullet what is the information at hand. We are aware of the dynamics of the system. Additionally, we also have knowledge about the range of uncertainties which the involved parameters can take, as well as the statistical properties of measurement noise. The Simulink model also contains an observer of the actual system. We re-create the system using the parameters with maximum or minimum uncertainty taken into account, wherever applicable. For e.g if uncertainty in tank area, A_1 is $\pm\alpha$, then we take $\bar{A}_1 = A_1 + \alpha$ such that the observed effect of uncertainty is largest on the dynamics of the plant. The resulting dynamics are as follows.

$$\bar{x}_1(k+1) = \bar{x}_1(k) + \frac{T_s}{\bar{A}_1}\left(\bar{c}_1 \bar{A}_1^p \operatorname{sign}\left(\bar{x}_3(k) - \bar{x}_1(k)\right) \times \sqrt{2g\left|\bar{x}_3(k) - \bar{x}_1(k)\right|} + u_1(k)\right) \tag{4}$$

$$\bar{x}_2(k+1) = \bar{x}_2(k) + \frac{T_s}{\bar{A}_2}\left(\bar{c}_2 \bar{A}_2^p \operatorname{sign}\left(\bar{x}_3(k) - \bar{x}_2(k)\right) \times \sqrt{2g\left|\bar{x}_3(k) - \bar{x}_2(k)\right|} - \bar{c}_3 \bar{A}_3^p \sqrt{2g \bar{x}_2(k)} + u_2(k)\right) \tag{5}$$

$$\begin{aligned} \bar{x}_3(k+1) = \bar{x}_3(k) + \frac{T_s}{\bar{A}_3} \bar{c}_1 \bar{A}_1^p \operatorname{sign}\left(\bar{x}_1(k) - \bar{x}_3(k)\right) \times \\ \sqrt{2g\left|\bar{x}_1(k) - \bar{x}_3(k)\right|} - \bar{c}_2 \bar{A}_2^p \operatorname{sign}\left(\bar{x}_3(k) - \bar{x}_2(k)\right) \times \sqrt{2g\left|\bar{x}_3(k) - \bar{x}_2(k)\right|} \end{aligned} \tag{6}$$

with the nominated parameters shown below along with their respective manufacturing uncertainty as $\pm\alpha$,$\pm\beta$ and $\pm\gamma$.

$$\begin{aligned} \bar{A}_i &= A_i + \alpha_i \\ \bar{A}_i^p &= A_i^p - \beta_i \\ \bar{c}_i &= c_i - \gamma_i \end{aligned} \tag{7}$$

Additionally, the output captured by the sensor is contaminated with measurement noise. The noise is modelled using a random distribution with maximum and minimum value specified. For the purpose of producing the residual, we take the value of noise such as to maximize the affect.

$$\bar{\xi} = \max(\xi)$$

The output could be stated as follows

$$\bar{y}(k) = \bar{x}(k) + \bar{\xi}$$

Using this data, we can produce an upper bound residual dynamics by invoking the use of observer similar to the one implemented for the actual system. The residual produced from the modified system could be stated as

$$\bar{r}(k+1) = \lambda\bar{r}(k) + \bar{\delta}(k)$$

where,

$$\bar{\delta}(k) = \max_{\eta}\max_{\xi}|\delta(k)|$$

To make the analysis easier, let us take all the uncertainty as 10% of the nominal values ($A_i = 0.156\ m^2$, $A_p = 5\times10^{-5}\ m^2$ and $c_i = 1$). Thus the respective values for evaluating $\bar{r}(k)$, given by (7), are $\bar{A}_i = 0.1716$, $\bar{A}_i^p = 4.5\times10^{-5}$ and $\bar{c}_i = 0.9$. The expression for $\delta(k)$ is given by the following equation.

$$\delta(k) = f(y(k) - \xi(k), u(k)) - f(y(k), u(k)) + \eta(k) + \xi(k+1) \tag{8}$$

To implement the discussed theory, numerous Simulink blocks were used. The expression for $\eta(k)$ was evaluated analytically and fed as a short algorithm, the value of which is determined at each time step.

$$\eta_1 = u_1(k)\left(\frac{T_s}{A_1} - \frac{T_s}{\tilde{A}_1}\right) + \sqrt{2g\,|x_3(k) - x_1(k)|}(sign(x_3(k) - x_1(k))) \times \left(\frac{T_s}{A_1}c_1A_1^p - \frac{T_s}{\tilde{A}_1}\tilde{c}_1\tilde{A}_1^p\right) \tag{9}$$

$$\begin{aligned} \eta_2 = u_2(k)\left(\frac{T_s}{A_2} - \frac{T_s}{\tilde{A}_2}\right) + \sqrt{2g\,|x_3(k) - x_1(k)|}(sign(x_3(k) - x_2(k)))\times \\ \left(\frac{T_s}{A_2}c_2A_2^p - \frac{T_s}{\tilde{A}_2}\tilde{c}_2\tilde{A}_2^p\right) - \sqrt{2gx_2(k)} \times \left(\frac{T_s}{A_2}c_2A_2^p - \frac{T_s}{\tilde{A}_2}\tilde{c}_2\tilde{A}_2^p\right) \end{aligned} \tag{10}$$

$$\begin{aligned} \eta_3 = \sqrt{2g\,|x_1(k) - x_3(k)|}(sign(x_1(k) - x_3(k))) \times \left(\frac{T_s}{A_3}c_1A_2^p - \frac{T_s}{\tilde{A}_2}\tilde{c}_2\tilde{A}_2^p\right) - \sqrt{2gx_2(k)} \\ \times sign(x_3 - x_2) \times \left(\frac{T_s}{A_2}c_2A_2^p - \frac{T_s}{\tilde{A}_2}\tilde{c}_2\tilde{A}_2^p\right) \end{aligned} \tag{11}$$

Figure 1 shows the diagram of the Simulink implementation. The blue, green and violet area show individual component with their respective label. These component are built upon the given observer model in *centralized_FD_basic.slx*.

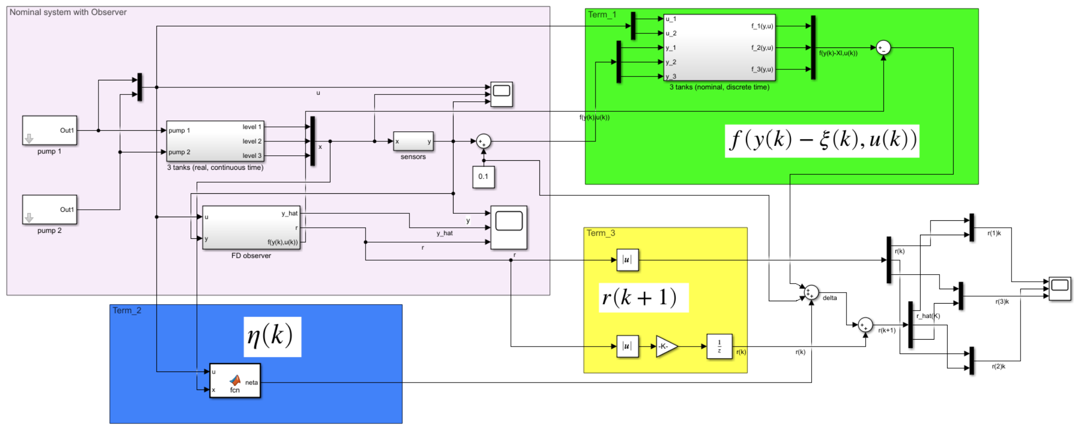

Figure 1: Simulink implementation

To properly assess each type of fault, we enable one fault at a time. At first, pump fault 1 and 2 are tested and the residual for each is displayed in Figure 2 and Figure 3 respectively. Furthermore, the residual for the tank leakage fault is shown in Figure 4. All the faults are set to be triggered at $T = 400$s. Additionally, the former pump fault is given a magnitude of 0.7 (i.e. the amount of blockage represented as a factor). The area of leakage on the other hand is set to 0.01m^2.

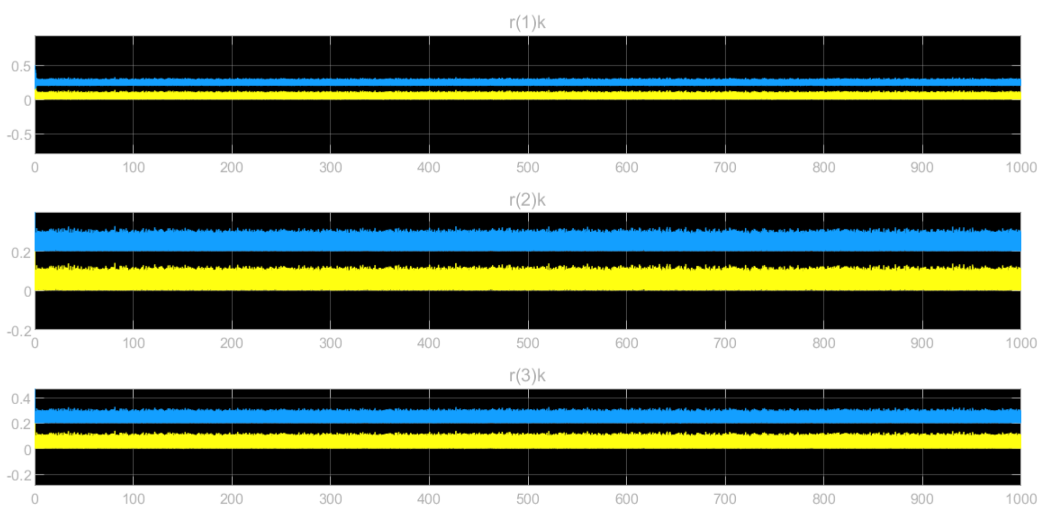

Figure 2: Residual and Threshold for Pump#1 Faliure

As per the graph, it is evident that pump faults are undetectable. This could be due to quick observer poles. To investigate this, the poles of the observer were slowed. The result was an increase in steady state error of the observer output, as compared to the output of the plant. Although the fault was detectable now, but the resulting deviation was sought to be impractical. Thus we only focused on tank leakage fault as shown in Figure 4. The stated fault was detectable since the threshold 1 was intersected around 400 s. Threshold 3 was intersected a little later as well, but no significant change was noticed in residual 2.

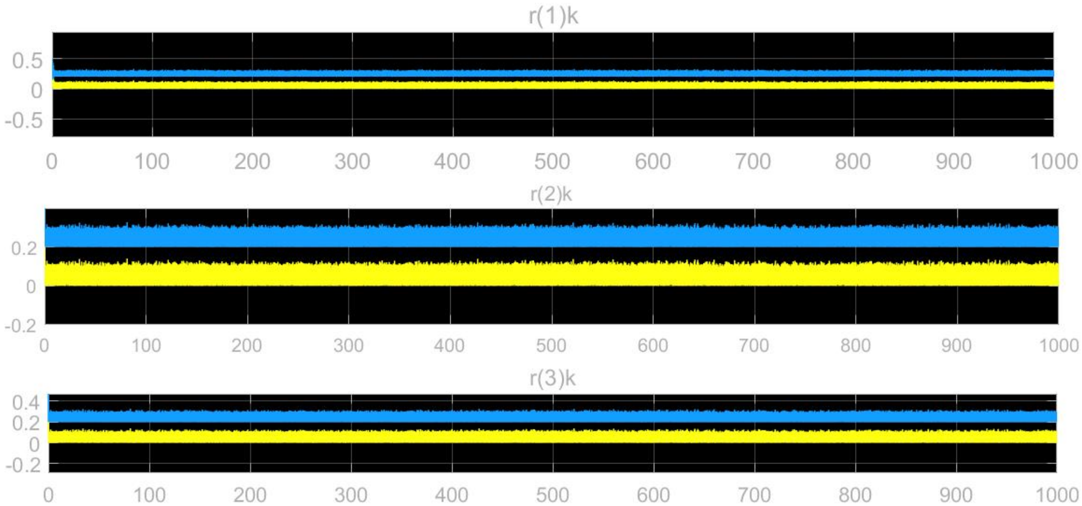

Figure 3: Residual and Threshold for Pump#2 Faliure

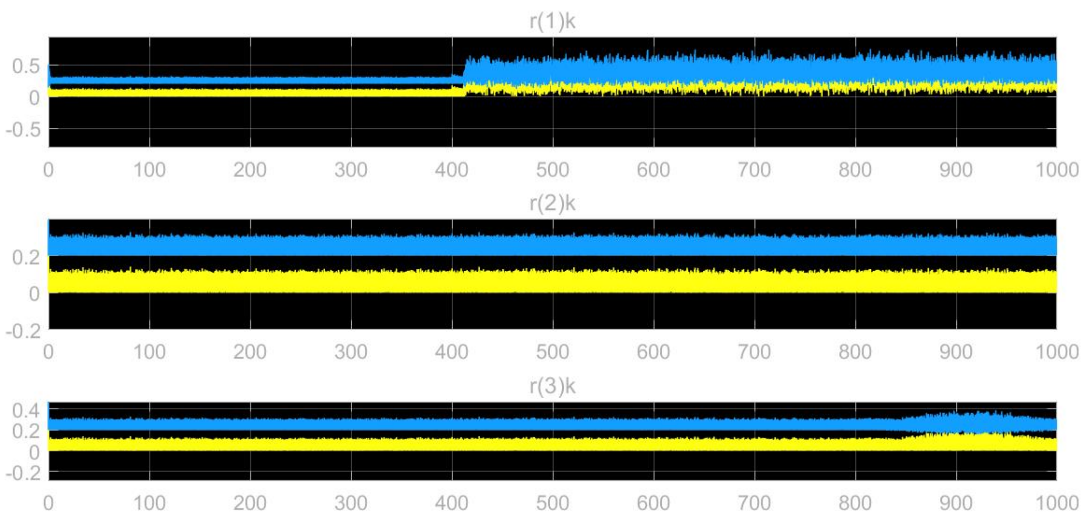

Figure 4: Residual and Threshold for Tank leakage

Task 1.2: Fault Isolation

In the previous section, we realized that the pump fault was not being detected. Formally stating, the residual produced by the fault did not satisfy the theorem of detectability. Attempting to isolate a fault which cannot be detected by FDAE is trivial thus we only focus on isolating the leakage fault. We can even predict that the FEI (Fault estimation and isolation) model for fault 1 and 3 would intersect the threshold only. We start by designing three different observers each embedding an individual fault. The fault profile for each fault could be stated as follows:

$$
\begin{array}{ll}
\phi^1 = \begin{bmatrix} \vartheta_1^1 g_1^1(k) \\ 0 \\ 0 \end{bmatrix} & \begin{array}{l} \vartheta_1^1 = a_1 \\ g_1^1(k) = -\frac{T_s}{A_1} u_1(k) \end{array} \\
\phi^2 = \begin{bmatrix} \vartheta_1^2 g_1^2(k) \\ 0 \\ 0 \end{bmatrix} & \begin{array}{l} \vartheta_1^2 = \pi \left(\rho_1\right)^2 \\ g_1^2(k) = -\frac{T_s}{A_1} \sqrt{2 g x_1(k)} \end{array} \\
\phi^3 = \begin{bmatrix} 0 \\ \vartheta_2^3 g_2^3(k) \\ 0 \end{bmatrix} & \begin{array}{l} \vartheta_2^3 = a_2 \\ g_2^3(k) = -\frac{T_s}{A_2} u_2(k) \end{array}
\end{array}
\tag{12}
$$

These observer profiles are activated when the FDAE designed in the previous section gives a positive for detection. From a practical side of view, the leakage fault required the radius as a parameter, but to make things easier, we took the area itself as a parameter rather than the radius. The basic observer with fault is designed as follows.

$$
\hat{x}^l(k+1) = f(y(k), u(k)) + \hat{\phi}^l \left(y(k), u(k), \hat{\vartheta}^l(k) \right) + \Lambda \left[\hat{y}^l(k) - y(k) \right]
$$

$$
\hat{y}^l(k) = \hat{x}^l(k)
$$

To implement this in Simulink, we modified the previously used observer block such as to accommodate the fault. Since we do not know the exact value of the parameters in the fault profile, for example, the dimension of the hole of leakage or the ratio of water flowing from damaged pump, we have to utilize a learning law such that our observer traces the actual fault and thereby adjusts the parameter involved. For this, we implement the gradient based learning law with a projection operator to avoid saturation problems.

$$
\hat{\vartheta}_i^l(k+1) = \mathcal{P}_{\Theta_i^l} \left(\hat{\vartheta}_i^l(k) + \gamma_i^l(k) g_i^l(k) \Delta r_i^l(k+1) \right) \qquad \text{,where}
$$

$$
\mathcal{P}_{\hat{\Theta}}(\hat{\vartheta}) \begin{cases} \hat{\vartheta} & \text{if } |\hat{\vartheta}| \leq M_{\hat{\Theta}} \\ \frac{M_{\hat{\theta}}}{|\hat{\theta}|} \hat{\vartheta} & \text{if } |\hat{\vartheta}| > M_{\hat{\Theta}} \end{cases}
$$

$$
\Delta r_i^l(k+1) = r_i^l(k+1) - \lambda r_i^l(k) \qquad \text{and}
$$

$$
\gamma_i^l(k) \frac{\mu_i^l}{\varepsilon_i^l + \left\| g_i^l(k) \right\|^2}
$$

with $\varepsilon_i^l > 0$ and $0 < \mu_i^l < 2$. The values were chosen as 8 and 1 respectively. Furthermore, the Radius limit $M_{\hat{\Theta}}$ is taken the same as the actual area of the tank. The reason why only certain residue are being utilized to evaluate the learning of parameter is since, dimension mis-match would occur while implementing the stated learning method, i.e. the $\Delta r_i^l(k+1)$ is a vector, while $g_i^l(k)$, $\gamma_i^l(k)$ and $\hat{\vartheta}_i^l(k)$ are all scalar values. The Simulink implementation is shown in Figure 5. To gain an insight about whether the learning law is working properly or not, we actuated only the second fault, and observed the learning curve of ϑ_1^2. It was found that without noise, the learning law worked as predicted with a slight offset, as shown in Figure 6. But unaccountable results were observed when noise was enabled. This is probably since the gradient law, which relies on the slope. Due to noise, the lower level signal could get contaminated, rendering the learning law useless.

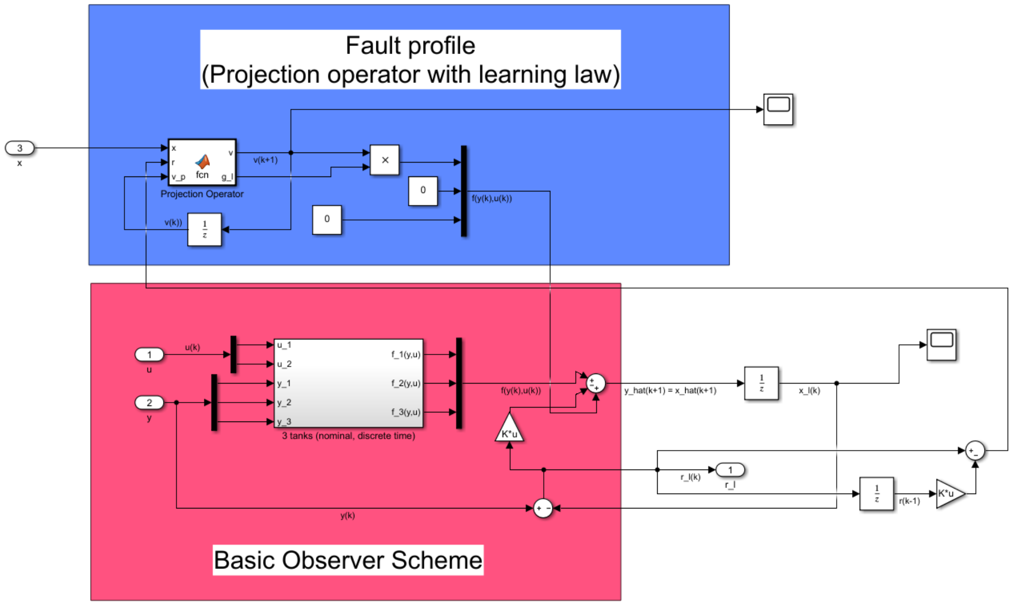

Figure 5: Fault profile for tank leakage with learning law

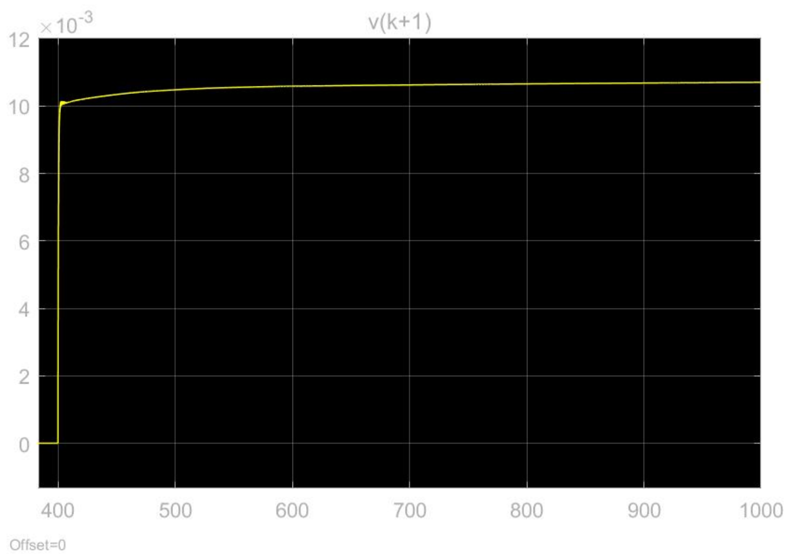

Figure 6: Learning curve of tank leakage area

We proceeded towards constructing the other two FEIs. The following equation is used to construct the threshold.

$$\begin{aligned}\delta^l(k) =& f(y(k)-\xi(k),u(k)) - f(y(k),u(k)) + \eta(k) + \xi(k+1) + \\ & \phi^l\left(y(k)-\xi(k),u(k),\vartheta^l(k)\right) - \hat{\phi}^l\left(y(k),u(k),\hat{\vartheta}^l(k)\right)\end{aligned}$$

The resulting Simulink model is shown in Figure 7 with the FEI unit highlighted in yellow.

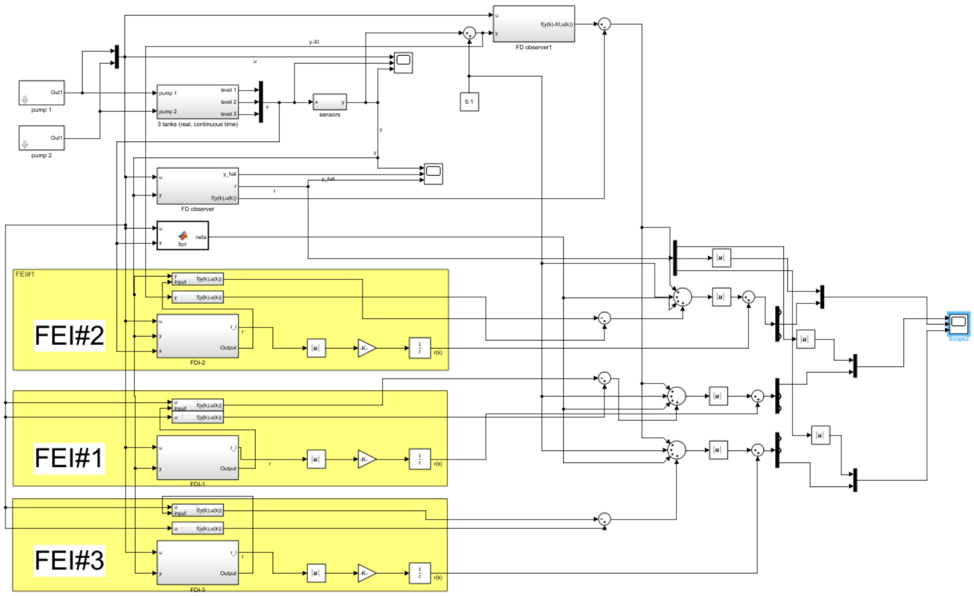

Figure 7: Fault estimation and isolation unit

The resulting residual and respective threshold is shown below in Figure 8

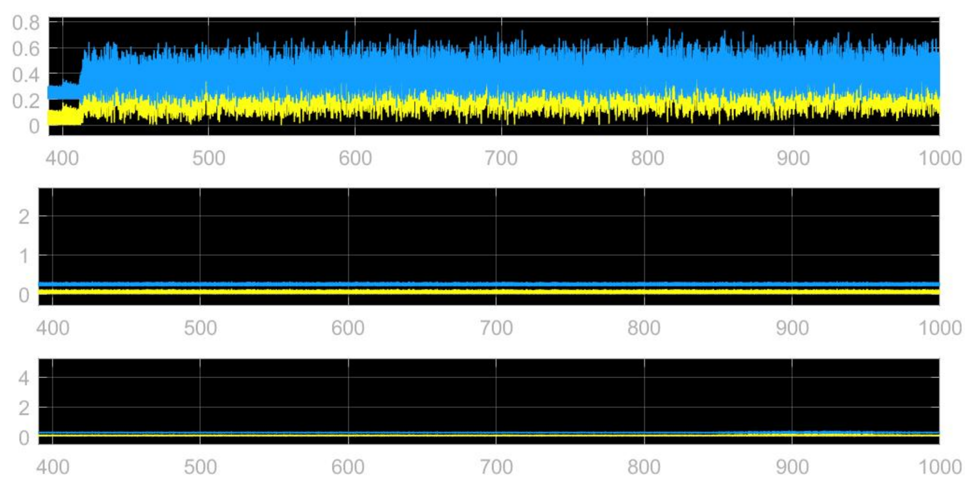

Figure 8: Residual with respective threshold

The preceding figure shows that the threshold intersection only takes place for the 1st and the 3rd residual, while the 2nd fault remains distant from the threshold. Therefore, we can conclude that the it is indeed the leakage fault that is presently causing the FDAE to react. It must be noticed that not all the residual of each FEI are being checked. Only those residuals are being checked which were involved in learning curve of the fault parameter.

Assignment Part 2

Task 2.1: FTC using virtual actuator and linearised model

The nonlinear plant equations are first linearized around an equilibrium point. The equilibrium point is chosen such that $X_{2,0} < X_{3,0}$. The equilibrium point we used was $X_{1,0} = 3, X_{2,0} = 2, X_{3,0} = 2.5$. Since the designed controller works with linearized dynamics, it uses $\delta x(k)$ and $\delta u(k)$. Hence, it is necessary to add the equilibrium point to the output. The poles of the closed loop system were placed at $-0.1, -0.2, -0.3$. Figure 9 shows the block diagram of the nominal dynamics.

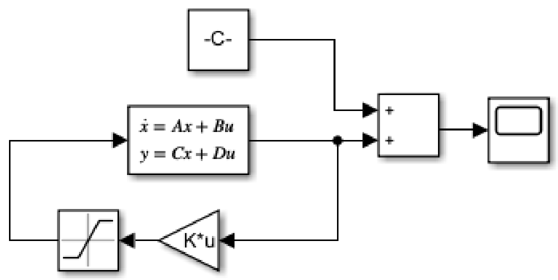

Figure 9: Block diagram of nominal system

Figure 10 shows the response of the nominal system.

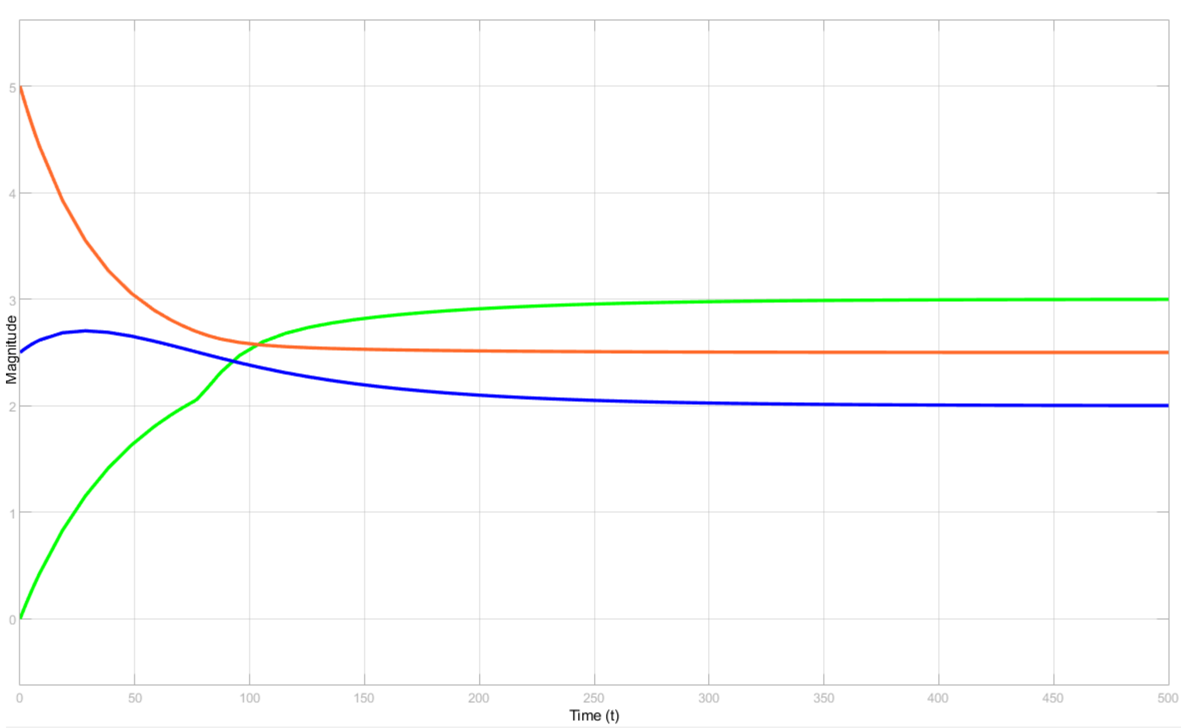

Figure 10: Response of nominal system

Since an actuator fault will occur, we design a virtual actuator so that this is accommodated for. We simulated a fault on the second pump, hence the second column of the B matrix will become zero. The scheme for the virtual actuator was implemented for the linearized system as shown in Figure 11. M was chosen such that the $A - B_f M$ is stable by placing poles at $-5, -6, -7$. N is computed such that $B_{Delta} = B - B_f N = 0$. This is possible because (A, Bf) is controllable.

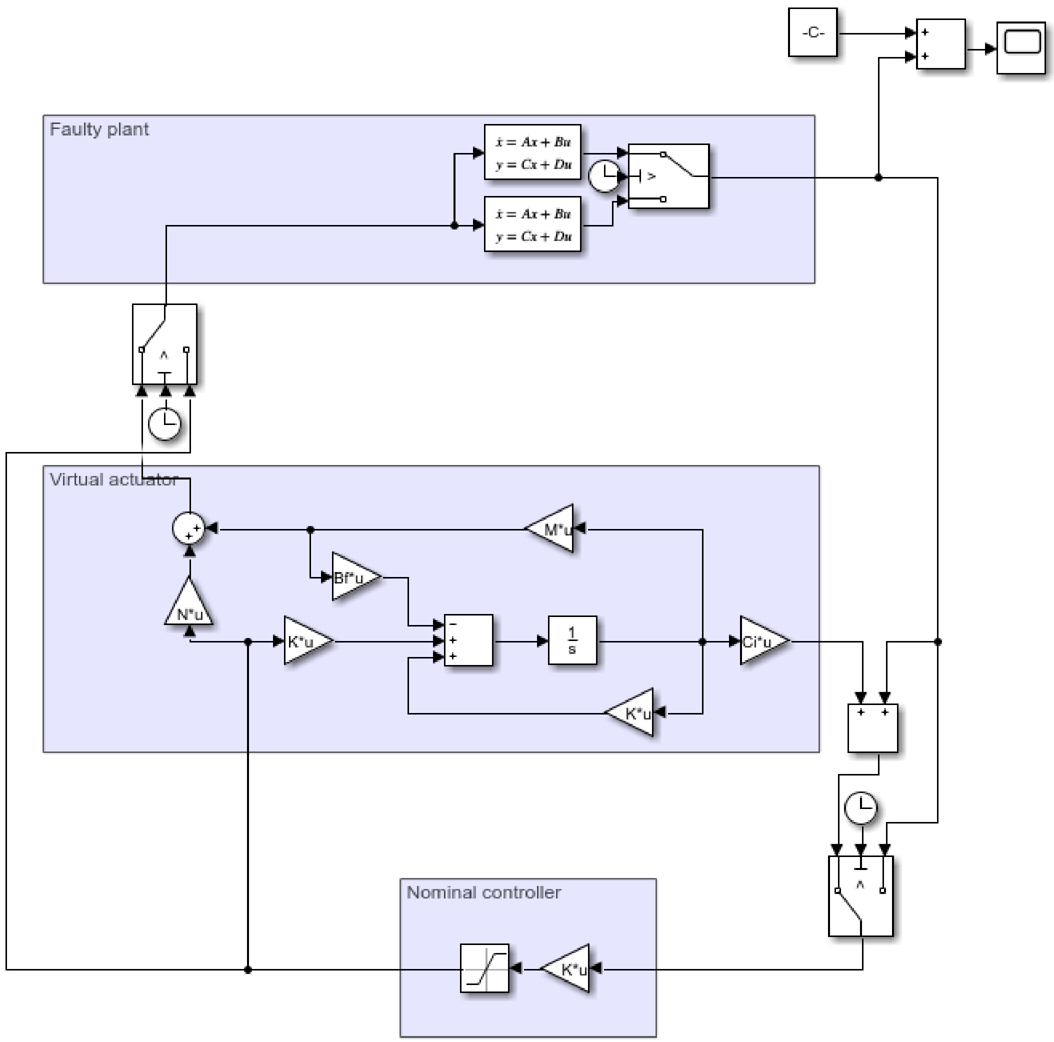

Figure 11: Virtual actuator for the linearized system

At $t = 200\,\mathrm{s}$, a fault occurs and the virtual actuator is activated. Figure 12 shows the dynamics of this system.

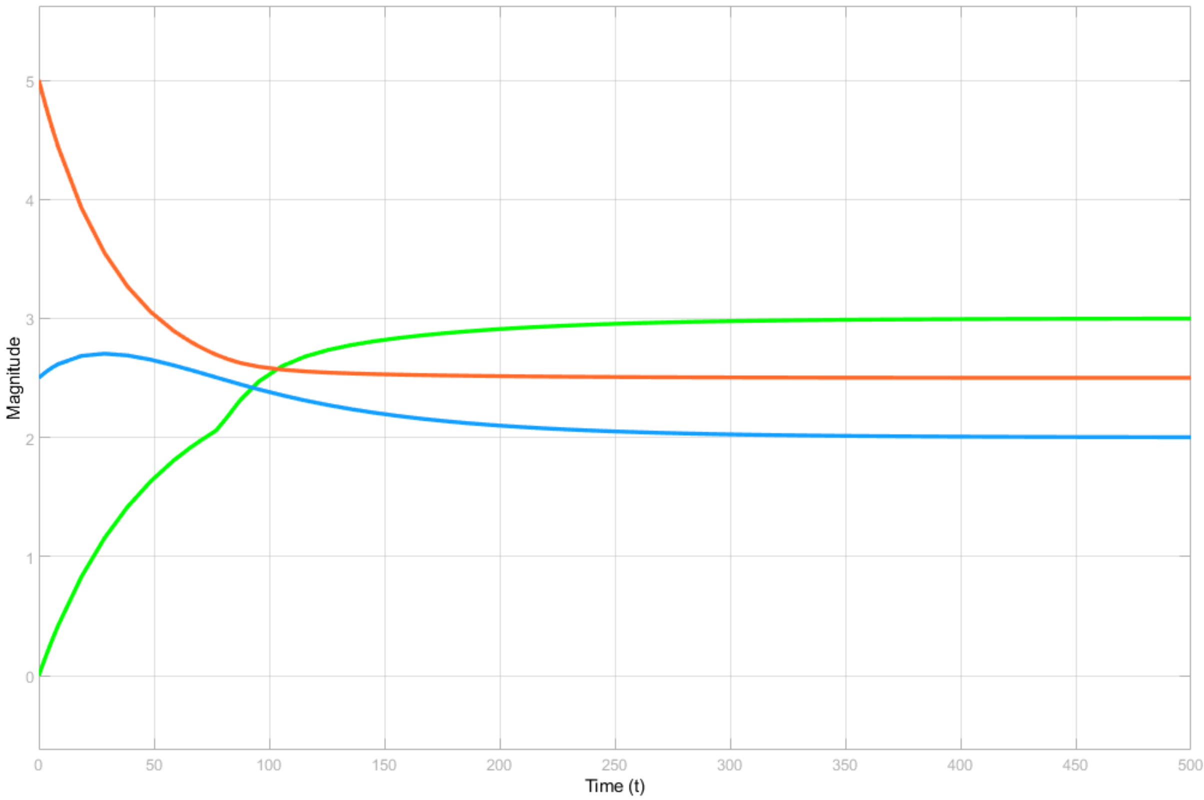

Figure 12: Virtual actuator for the linearized system

Task 2.2: FTC using virtual actuator and nonlinear model

The approach in this task is similar to the previous task. We have to add the equilibrium inputs and subtract the equilibrium points from the output. Figure 13 shows the block diagram for the nonlinear system.

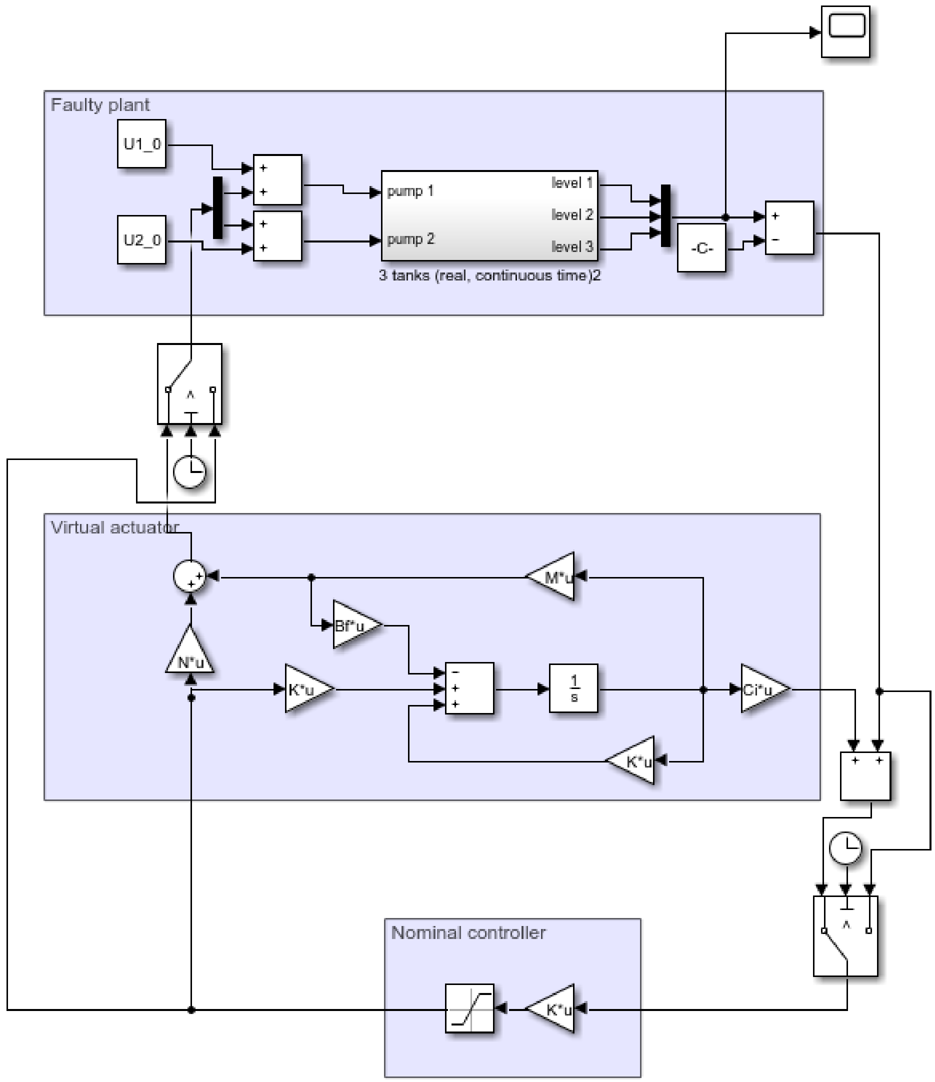

Figure 13: Virtual actuator diagram for the linearized system

Figure 14 shows the response of the virtual actuator for the nonlinear system.

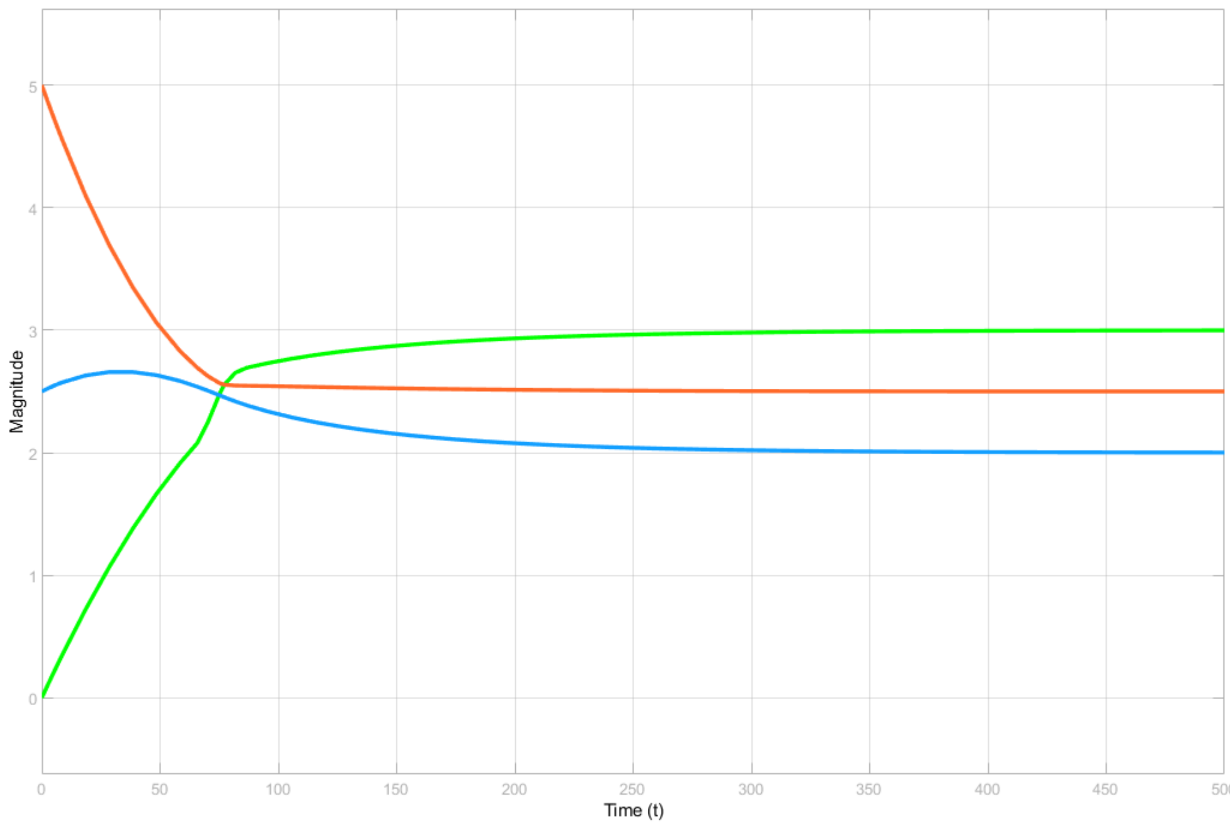

Figure 14: Virtual actuator dynamics for the nonlinear system

Figure 15 shows a comparison between the dynamics of the previous linear and nonlinear models. It can be seen that there is some difference but they converge to the same results.

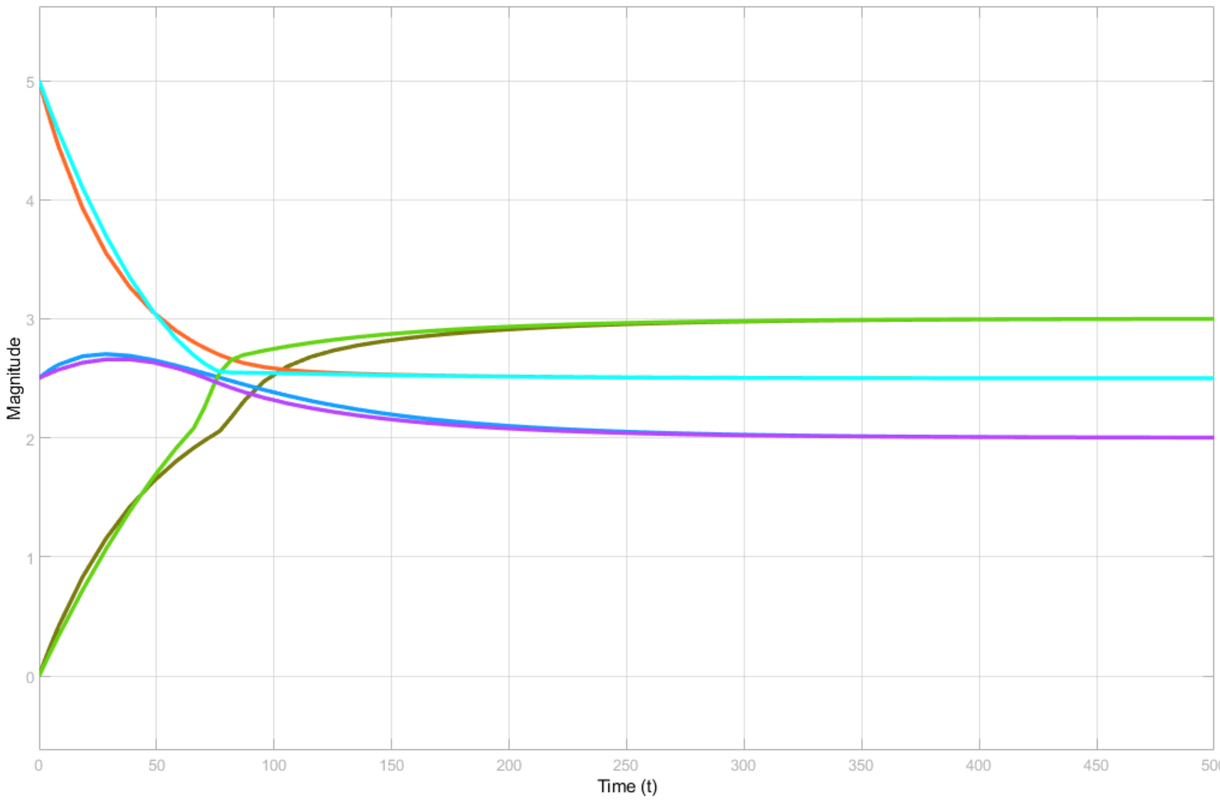

Figure 15: Comparison between the linear and nonlinear systems with virtual actuator.