Computational Intelligence

Project 1

Vellios Georgios Serafeim
velliosgeo@gmail.com

Speed Control of a Workbench Mechanism with Fuzzy Controllers

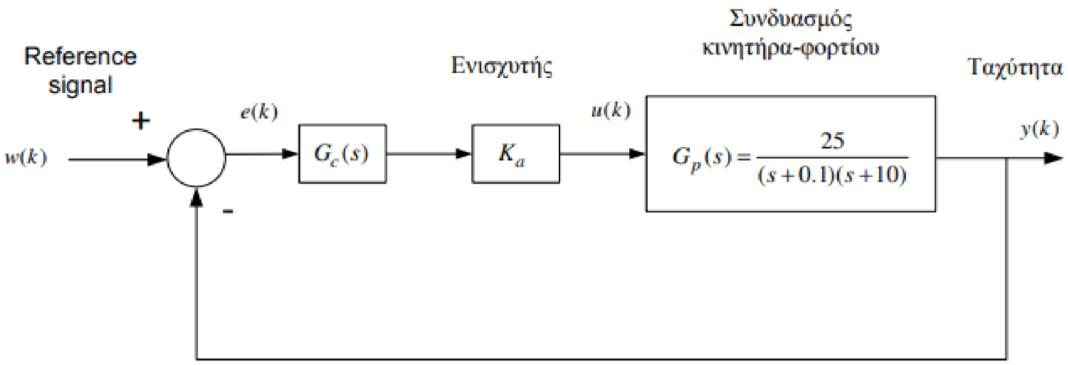
The purpose of this project is the design of controllers to control the speed of a precision dc motor that moves a work-table. The maximum speed of the work table is 50 rad / sec . Two controllers will be designed, a linear PI controller and a fuzzy controller.

Linear Controller design

The PI linear controller to be designed is of the form:

$$G_c(s) = K_p + \frac{K_I}{s} = \frac{K_p(s+c)}{s}, \quad c = \frac{K_I}{K_p}$$

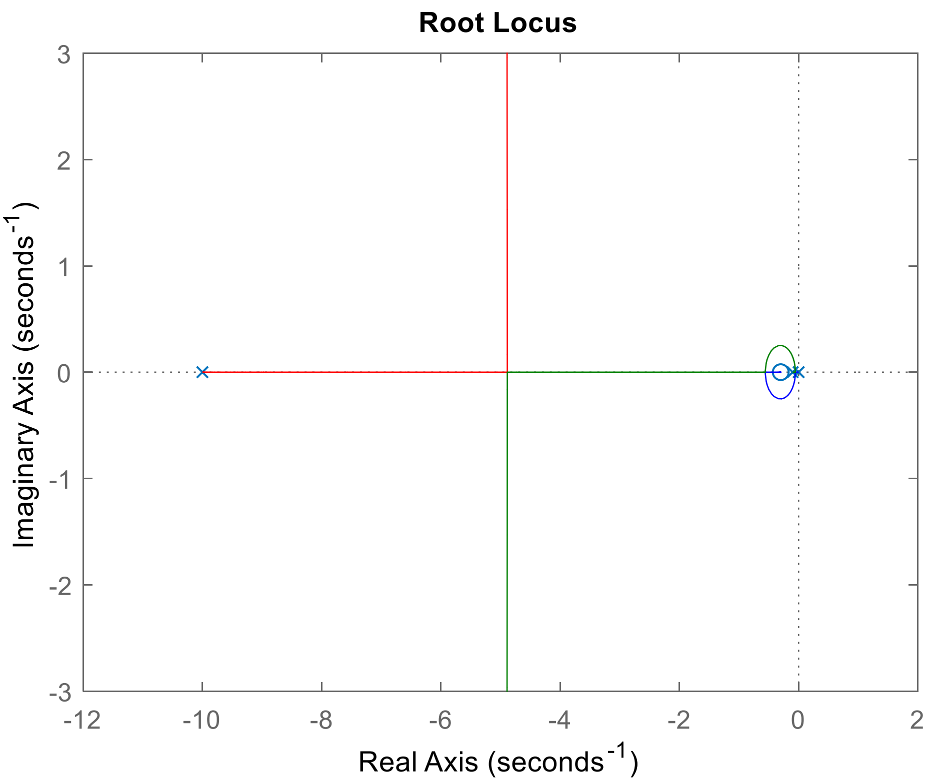
while the control system is as follows:



The specifications that the linear controller must meet are:

- Overshoot for step input less than 8 %
- Rise time less than 0.6 seconds

The poles of the system are -0.1 and -10 so the zero of the controller is placed between the two poles and close to the dominant (-0.1) so we choose to place it at the value -0.3. We then create the locus of roots for the open-loop function, $\frac{25K_p(s+c)}{(s+0.1)(s+10)}$ choose an initial K_p , and then investigate by trial and error values for the controller gain K_p via the stepinfo command for the closed-loop system until both of our specifications are satisfied to a satisfying degree. The locus of the roots for the open-loop function, the values of K_p as well as the overshoot and rise time for them are shown in the table below.

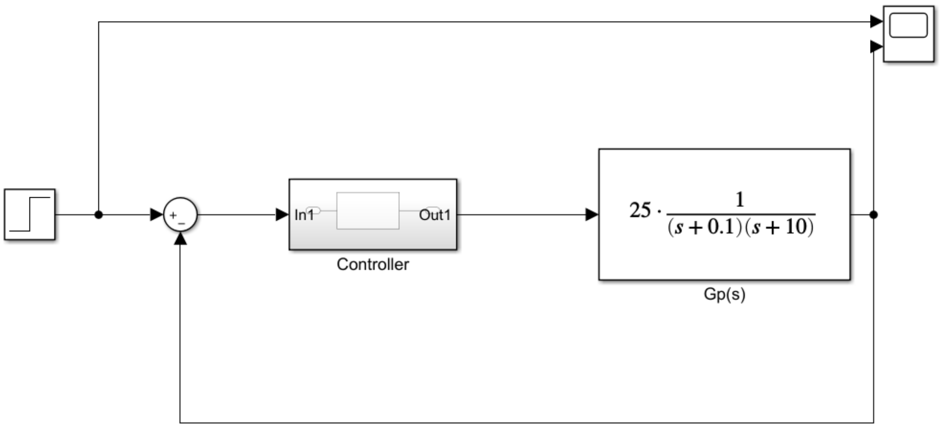


Locus of roots for the open-loop function (K_p =1.3)

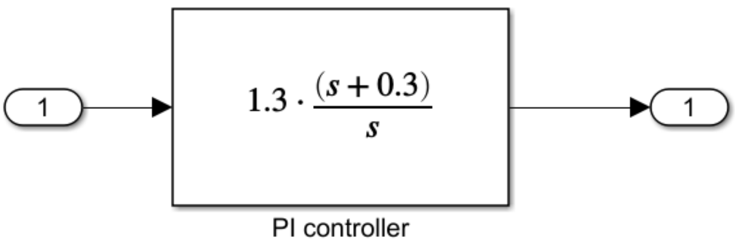
K_p	**Rise time (s)**	**Overshoot (%)**
2	0.0751	40.33
1.5	0.3822	6.0182
1.4	0.4098	5.7741
1.1	0.526	5.7744
1.2	0.4804	5.6499
1.3	**0.4422**	**5.6499**

The final selections for the controller are **K_p=1.3** , K_I = c * K_p ⇒ $\boldsymbol{K_I = 0.39}$ which achieve an elevation of 5.65% and a rise time of 0.44 seconds.

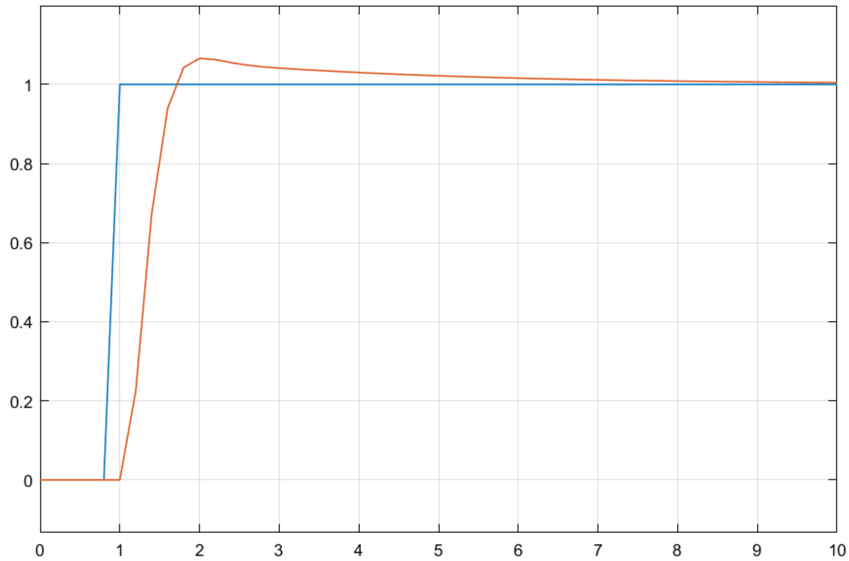
The system was simulated in Matlab Simulink in the file PIController.slx as presented below.



using the controller

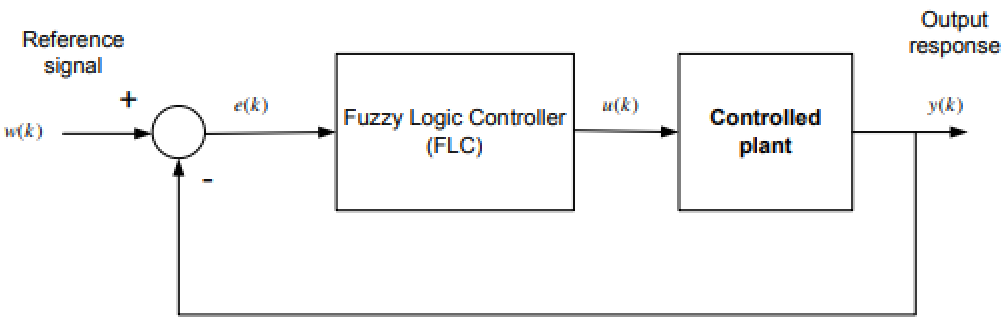


The response of the closed-loop system for a step function is presented below.



Fuzzy Controller (FLC) Design

a) The FZ - PI type controller will be used for the fuzzy controller. The control system is shown in the figure below.



The implementation of the closed loop system will be done in discrete time with sampling rate 0.01 sec . The error verbal variables $E, \dot{E}$ are described by 7 verbal values and the control signal verbal variable $\dot{U}$ is described by 9 verbal values. The rule base of the fuzzy controller we are creating is described in the table below. (Underlined are the special rules created by the control signal word variable being described by 9 values rather than 7).

DE / E	**NL**	**NM**	**NS**	**ZR**	**PS**	**PM**	**PL**
PL	**ZR**	PS	PM	PL	PL	<u>PV</u>	<u>PV</u>
PM	NS	**ZR**	PS	PM	PL	PL	<u>PV</u>
PS	NM	NS	**ZR**	PS	PM	PL	PL
ZR	NL	NM	NS	**ZR**	PS	PM	PL
NS	NL	NL	NM	NS	**ZR**	PS	PM
NM	<u>NV</u>	NL	NL	NM	NS	**ZR**	PS
NL	<u>NV</u>	<u>NV</u>	NL	NL	NM	NS	**ZR**

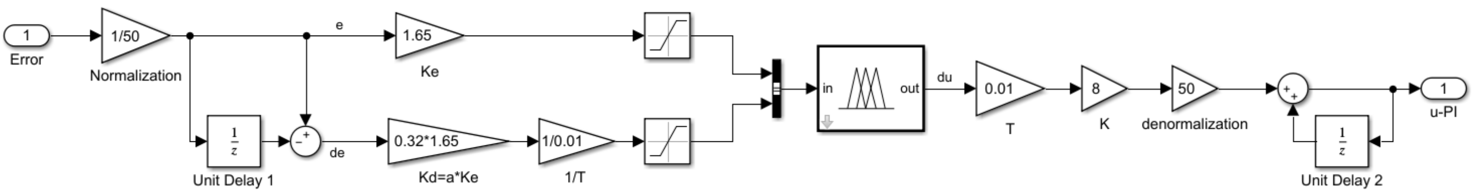
Rulebase for Fuzzy PI controller

The initial values for the gains of the fuzzy controller were set based on the values for the PI controller. So:

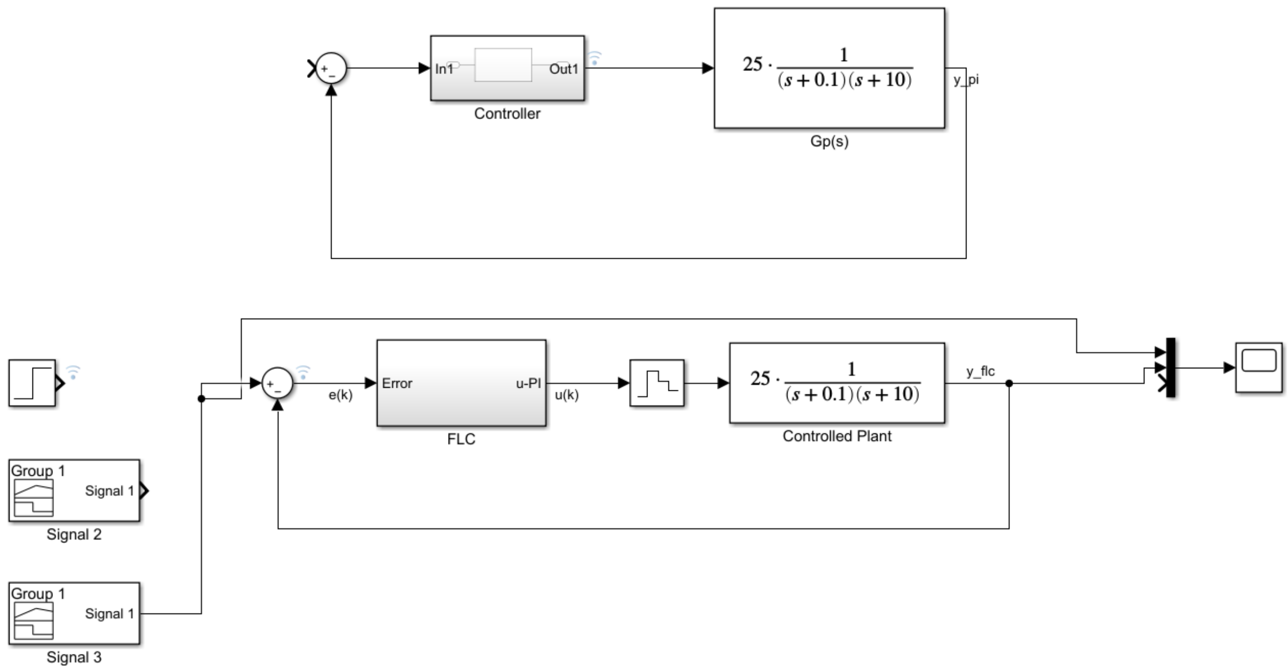
$$\boldsymbol{K_e} = \mathbf{1}\,, T_i = a = \frac{K_p}{K_I} = \frac{1.3}{0.39} \Rightarrow \boldsymbol{a} = \mathbf{3.334},$$

$$K = \frac{K_P}{F\{T_I * K_e\}} = \frac{1.3}{F\{3.334 * 1\}} = \frac{1.3}{1} \Rightarrow \boldsymbol{K} = \mathbf{1.3}$$

The fuzzy controller was created in Matlab simulink and is presented below.

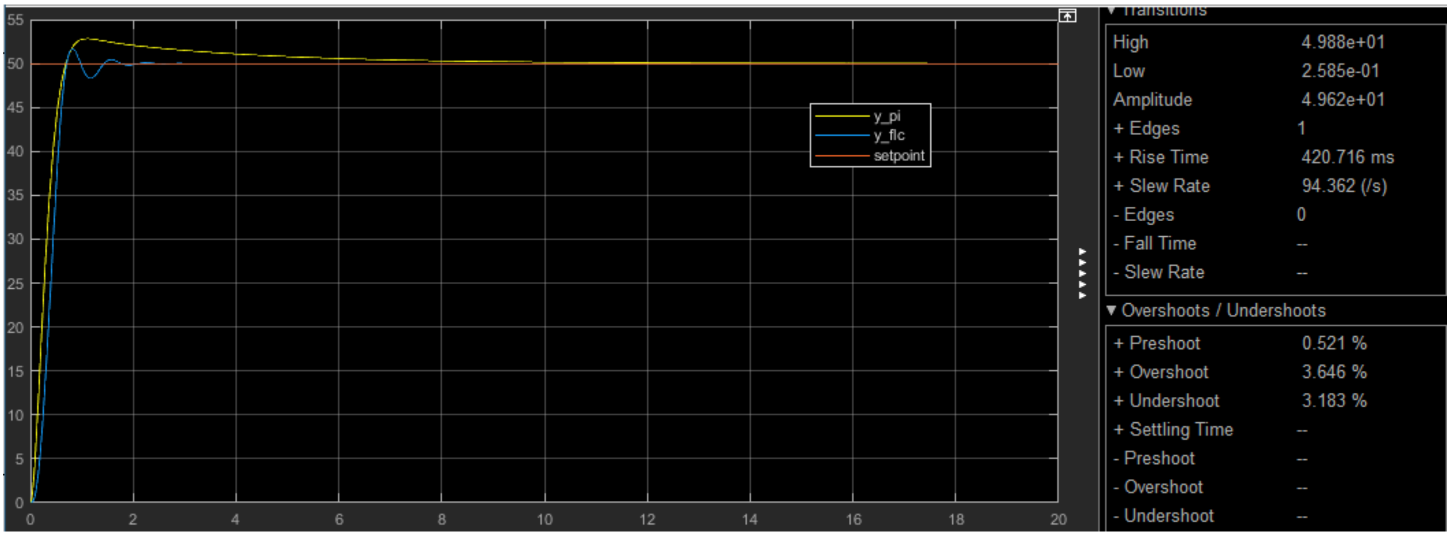


The closed-loop systems for the two controllers (PI and fuzzy) are presented in the below figure.



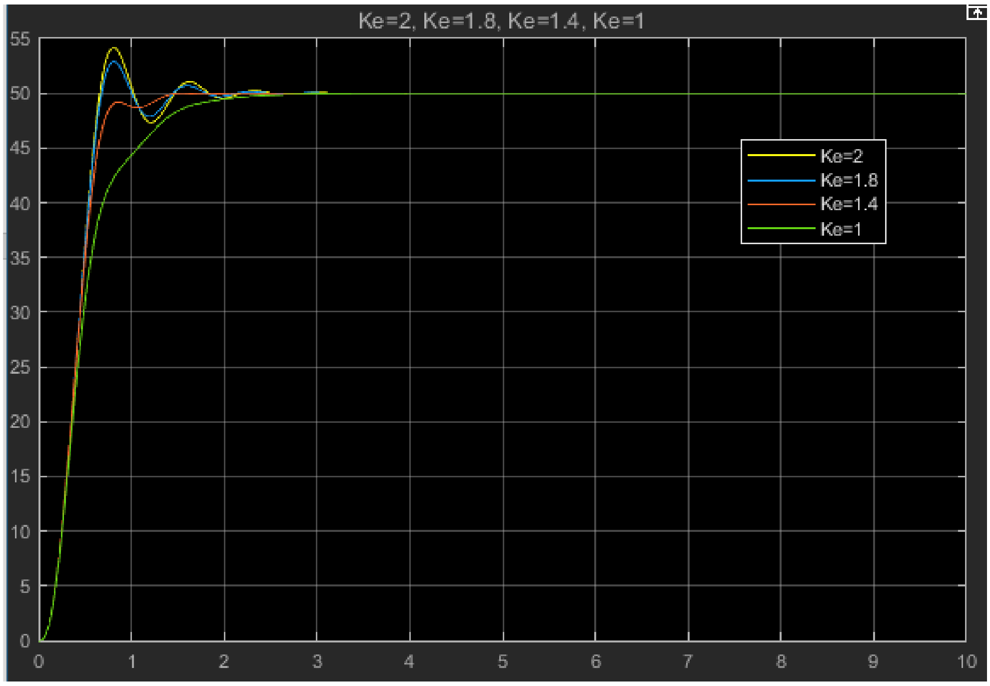
Using the initial values for the controller parameters, the system presents a Rise time of 6.945 sec and an Overshoot of 0.501%. To achieve the goal for the controller for a step input with a width of 50 we adjust the scaling gains and observe that the specifications are met for the values $\boldsymbol{K = 8, K_e = 1.65, a = 0.32}$.

With the specific values we achieve Rising Time 0.42 seconds and Overshooting 3.646% as shown in the diagram below comparing the output of the closed loop system for the two controllers for a step input with a width of 50.

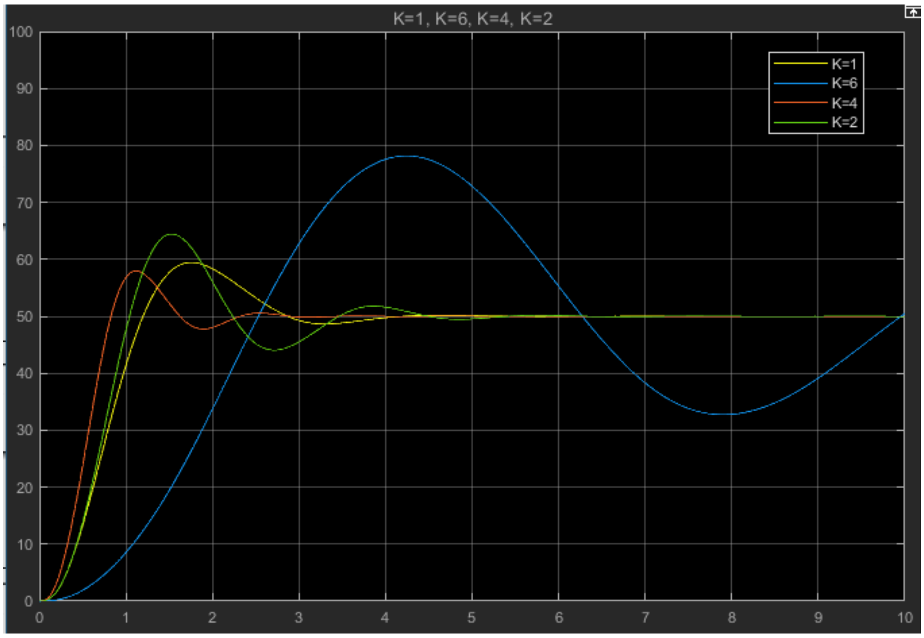


We observe that the system with the fuzzy controller has a much better response than the one with the PI controller so the fuzzy controller is better than the PI.

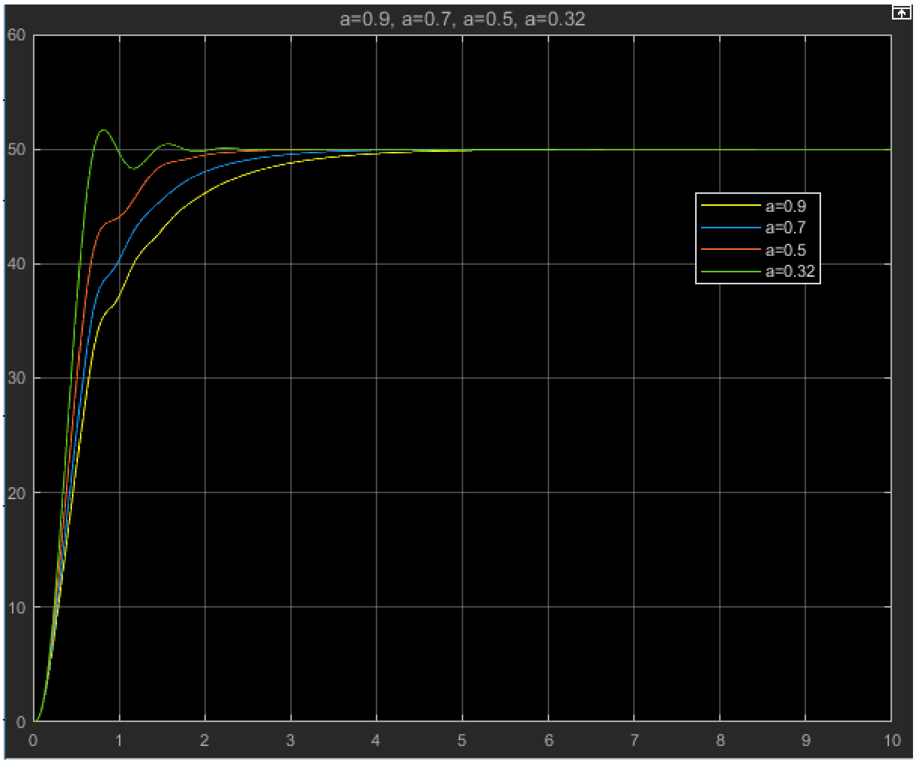
Presented below is the response of the closed-loop system with the fuzzy controller for different values of K, K_e and α.



Response of the system for different values of K_e



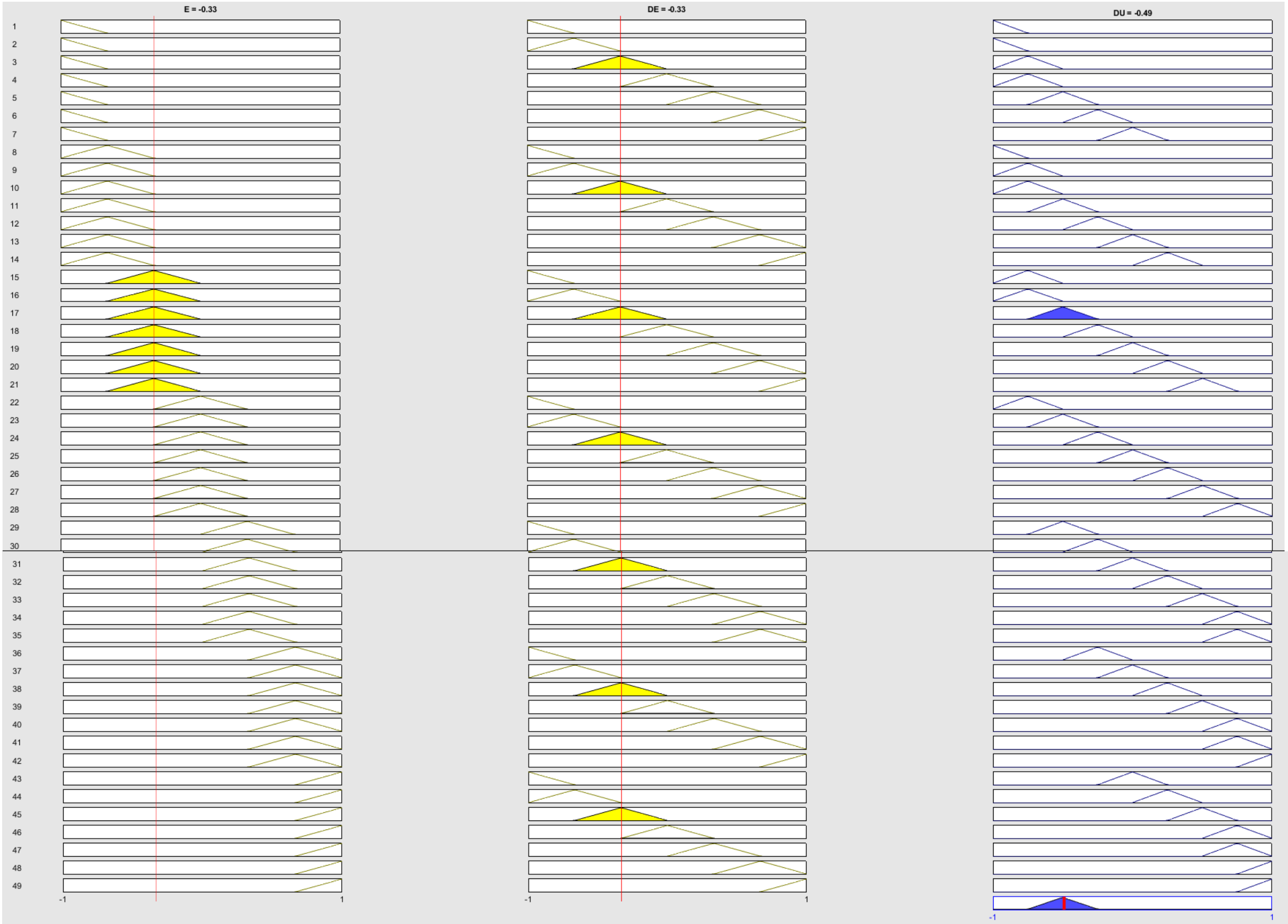
System response for different K values



System response for different values of a

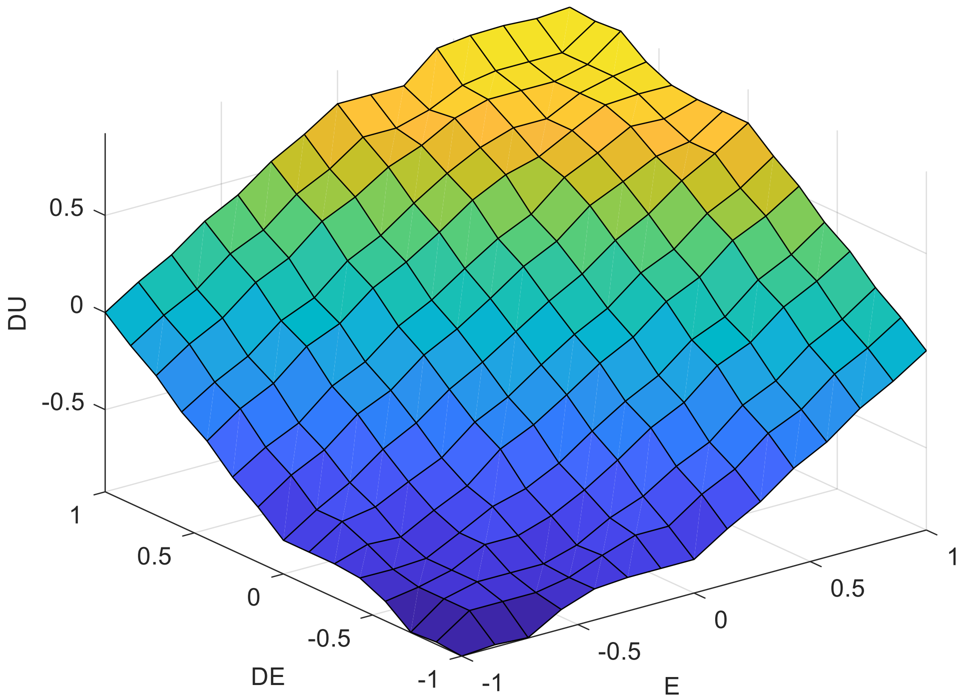
We notice that as K_e increases, so does overshooting . Also, as K increases, the response of the system becomes faster, but up to a certain value for K. Finally, as α decreases, the response becomes faster, but, at the same time, overshooting also increases .

b) We consider an excitation where $e \; is \; NS$, $\Delta e \; is \; NS$. The rules that are triggered are presented below.



We notice that only one rule is excited, the $\boldsymbol{If\ e\ is\ NS\ AND\ \Delta e\ is\ NS\ THEN\ \dot{U}\ is\ NM}$ that is also verified by the value returned by evalfis , namely -0.49, which in the range [-1, 1] and is in fact Negative Medium. The figure above also confirms the disambiguation technique. In the region where e < 0 and Δe < 0 it is correct that y(k) > y(k -1) (rising path), the error is not self-correcting and the response moves away from the reference point so the controller must operate to bring the response back to the desired levels.

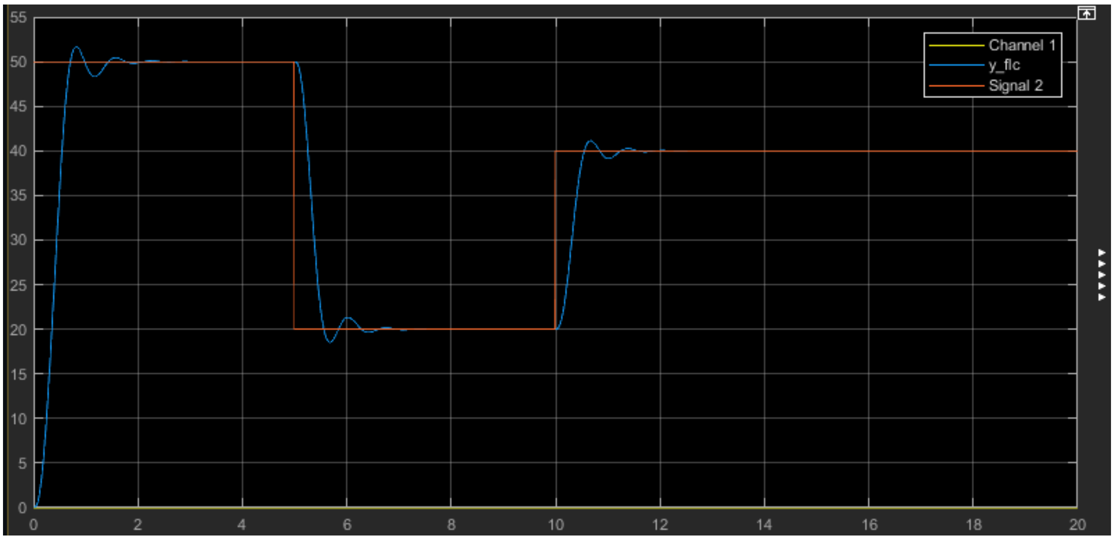
c) The output of the controller in relation to the inputs is presented in the figure below.

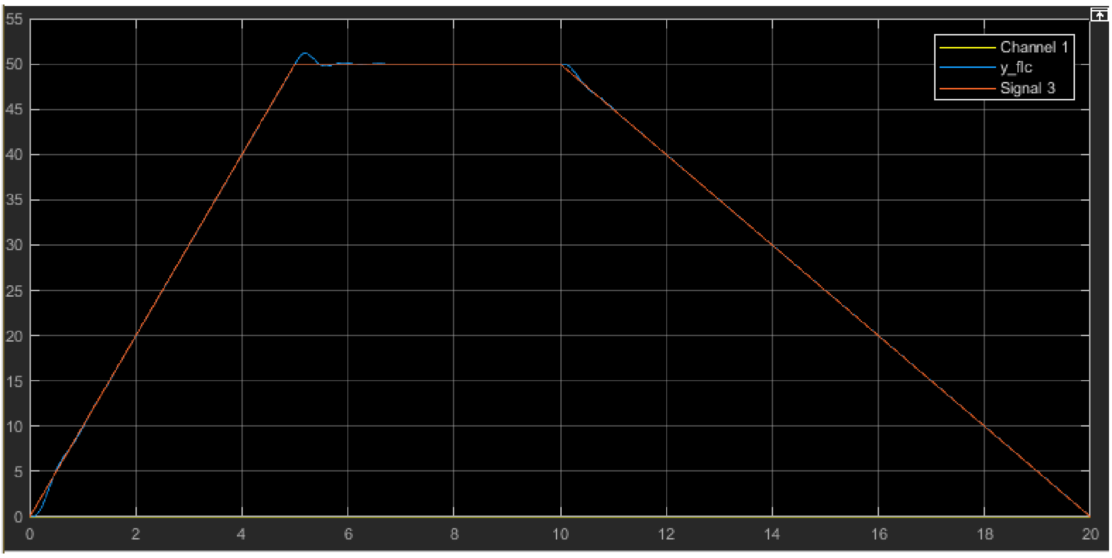


In the center of the diagram where E and ΔE have zero or small positive and negative values, the output is quite close to the reference point and, as a result, ΔU is a zero or small value. When E is <-0.5 and ΔE is positive or E>0.5 and ΔE is negative, ΔU is used to speed up the response or slow down the rate of approach of r . The extreme values of the controller are observed when the two inputs are identical and their absolute value is greater than 0.5.

Scenario 2

By inputting two different profiles of the reference signal, we observe the response of the closed loop system in the 2 diagrams below.





We observe that in the first case, due to the sudden changes in the input signal, the controller is slightly delayed in reaching the reference point. Conversely for the ramp input we observe that the controller follows the reference input in a large margin. So, it is concluded that the controller is really effective for ramp inputs.