Control Theory.
Homework Three.
Polina Turishcheva. Group 3. Variant B.

Task 1
Name : Polina
Email : p.turischeva@innopolis.university
Task 2
A Source code. K_i and const should be set to zero for this part of task, they will be used in E part.
$\mu = 44, k = 1$

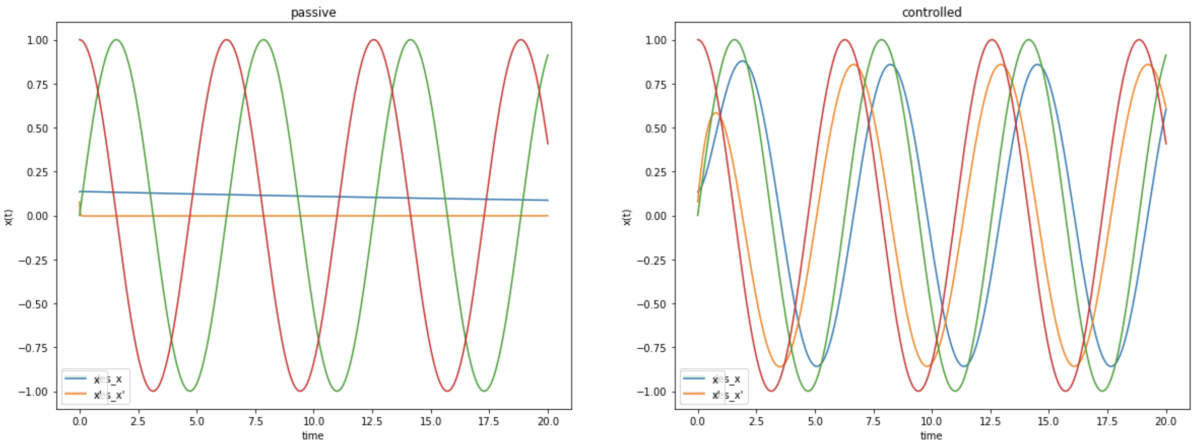

Figure 1: random input, $\mathrm{K}_p = 300, K_d = 4000$

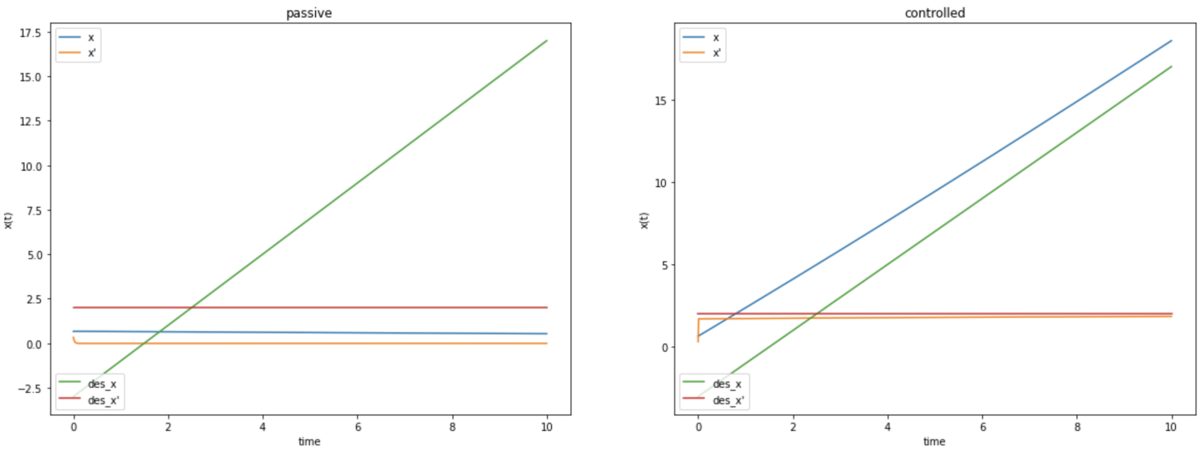

Figure 2: y=2x-3. Random input, $\mathrm{K}_p = 300, K_d = 4000$

B In all of the following examples $K_p = 200$, $K_d = -3$ and initial input is random.

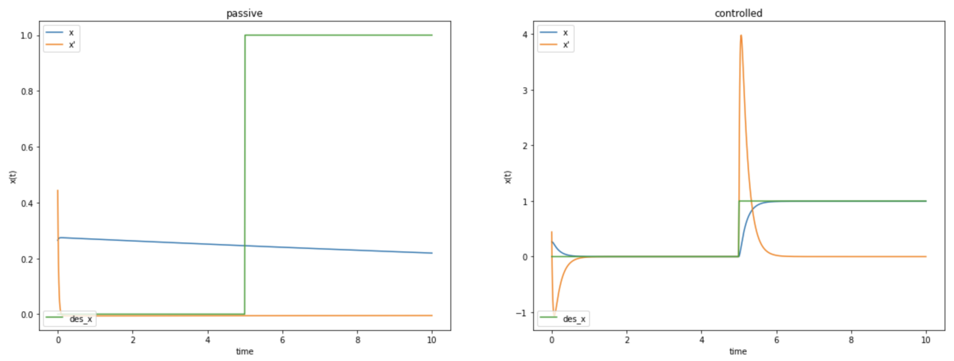

Figure 3: Step response. Rapid changes in the derivative are caused by random input and rapid change in step function.

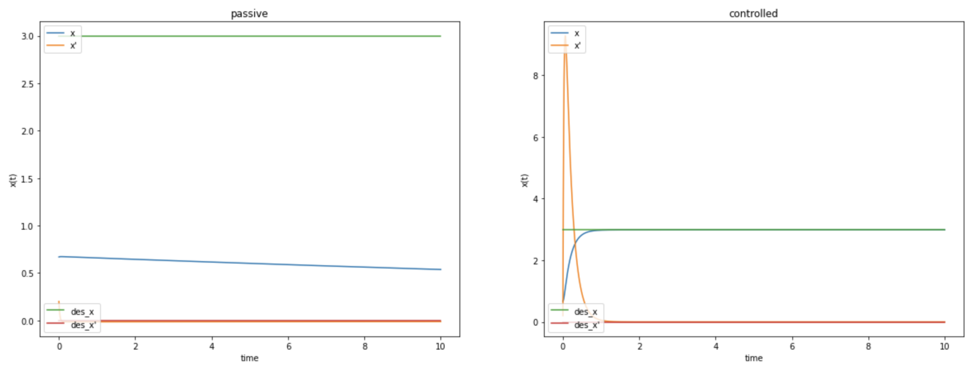

Figure 4: Immediate step response. PD for const.

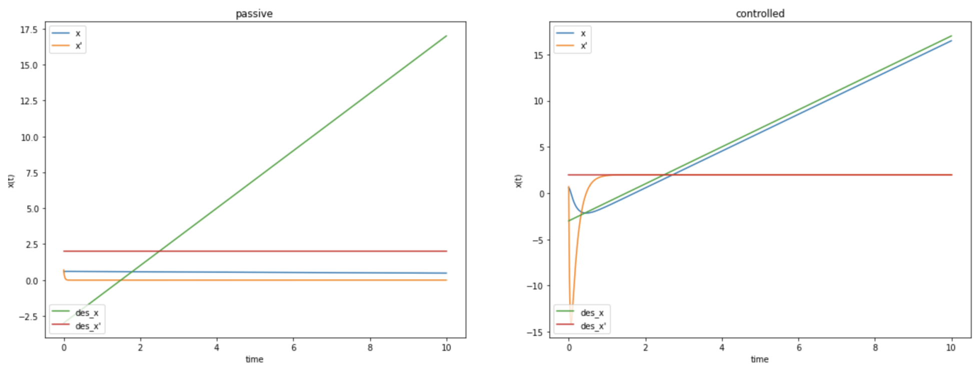

Figure 5: PD for line.

C

From here we take the formula:

$$\frac{K_d s + K_p}{ms^2 + (K_d + \mu)s + (K_p + k)} = \frac{-3s + 200}{s^2 + 41s + 201} = \frac{(-3s + 200)}{(x + 20.5 + \sqrt{877}/2)(x + 20.5 - \sqrt{877}/2)}$$

As both poles are negative, hence, the system is stable.

D

1. Find eigenvalues, if both are less than 0, we do not need any controller
2. Here one is positive. Eigenvalues are $2.5 \pm \sqrt{285}/2$.
3. Controller looks like $\begin{bmatrix} x_1' \\ x_2' \end{bmatrix} = \begin{bmatrix} 10 & 3 \\ 5 & -5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, we choose a, b,c,d such that both eigenvalues become negative. One of such matrixes is $\begin{bmatrix} -11 & -3 \\ 0 & 0 \end{bmatrix}$, after summing the equation will become $\begin{bmatrix} x_1' \\ x_2' \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 5 & -5 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, as matrix is diagonal its eigenvalues are its diagonal elements and both of them are already negative.

E Same code from the link A is workable. Just define K_i as non-zero value, and const=9.8.

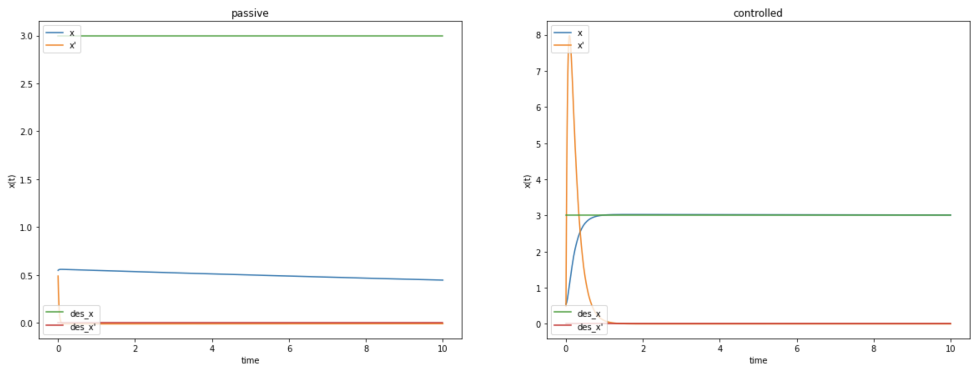

Figure 6: PID for const

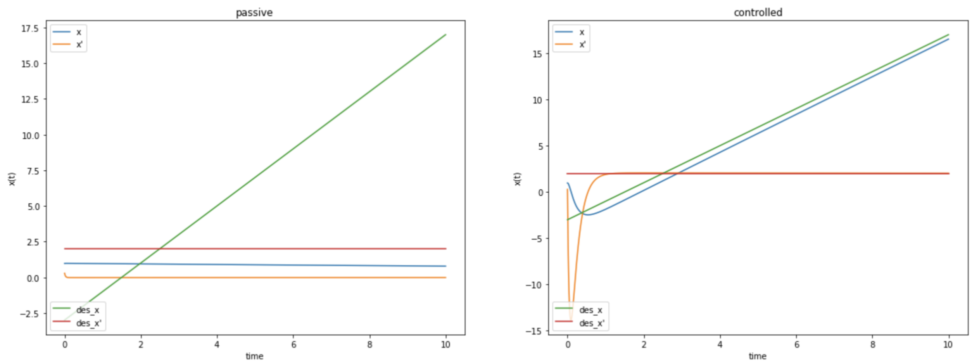

Figure 7: PID for line

On both pictures $K_p = 100$, $K_i = 10$, $K_d = -20$.

Task 3

My variant: $W = \dfrac{s^2 + 3s + 8}{s^4 + 2s^3 + 3s^2 + 13s + 7}$

1. First I used the instruction via the link and created the schema with my transfer function.

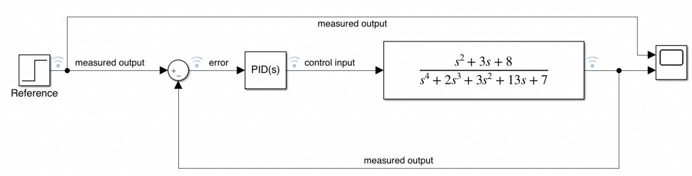

Figure 8: PID Schema

2. Named the PID block (block menu → State Attributes → State name). In block menu → main set random coefficients, applied changes, in *automated tune* chose *Transfer Function Based (PID Tuner App)*.

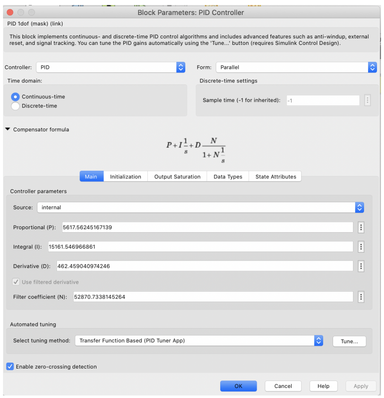

Figure 9: PID block menu

3. Pressed tune and in an opened window pressed show parameters and played a bit with Response Time and Transient behaviour. Changing them I could see the changes in overshoot and rise time. In order to check steady state error I should press the scope in the initial scheme and see the difference between 2 responses.

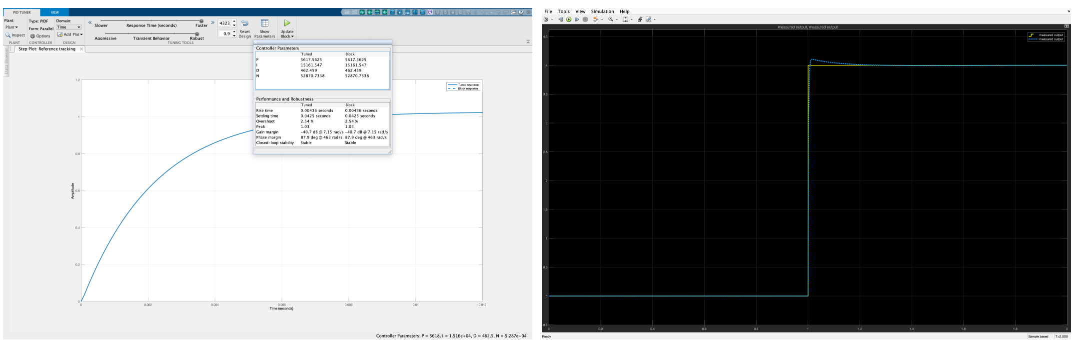

Figure 10: Tuner and Scope Graphs

Here are my final coefficients : P≈5617.5625, I≈15161.5470, D≈462.4590

Task 4

My variant: $W = \dfrac{s+3}{2s^3+4s^2+7s+13}$

Fisrt we draw a root locus map for a closed loop with only transfer function. We do it in Control System Designer. However, the interface to edit root locus map their is user-UNfriendly, that is why I did it in simulink and than redraw the map.

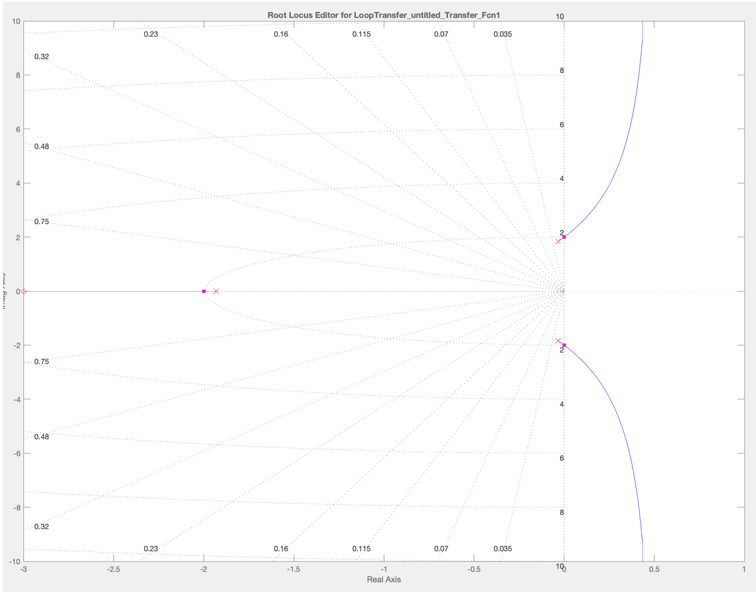

Figure 11: Root Locus for transfer function

We can see that vertical asymptotic line has positive real coordinate, which means that system is unstable. As far as we should shift the asymptote to the left we should add lead compensator. After small playing with numbers we get the following scheme and root locus for this scheme:

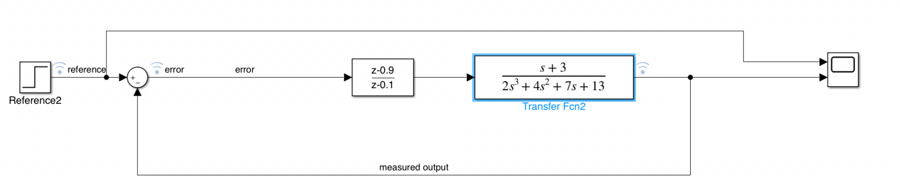

Figure 12: Scheme with compensator

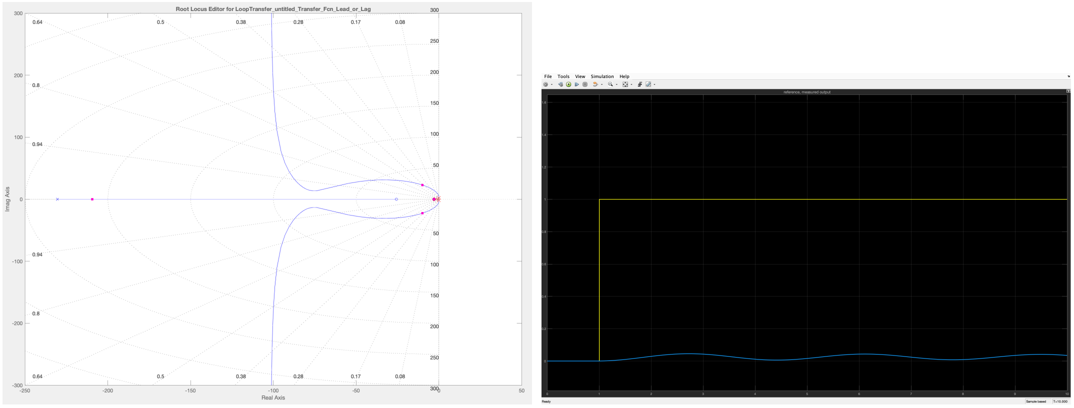

Figure 13: The are small oscillations but we will fix them after adding a gain. Gain is needed because we have very low constant, which is a baseline for oscillations. Note that adding a gain does not change the root locus map.

I have iteratively tried many gains and dicovered that 10000 is the optimal one. It give the least steady state error and does not cause oscillations (gains more that 10000 causes them). Unfortunately it gives relatively big overshoot 31.7% (the peak value is 1.317). If you want to decrease the overshoot you can decrease the pole of compensator- decreasing it to 0.01 gives peak of 1.264 and does not change the steady state error (not noticable at least). If you want to decrease overshoot even more you can increase a bit a zero of compensator, however, it will change the steady state error- not crucially but noticeable change.

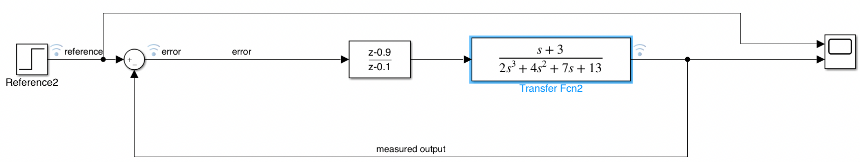

Figure 14: Scheme with compensator and gain

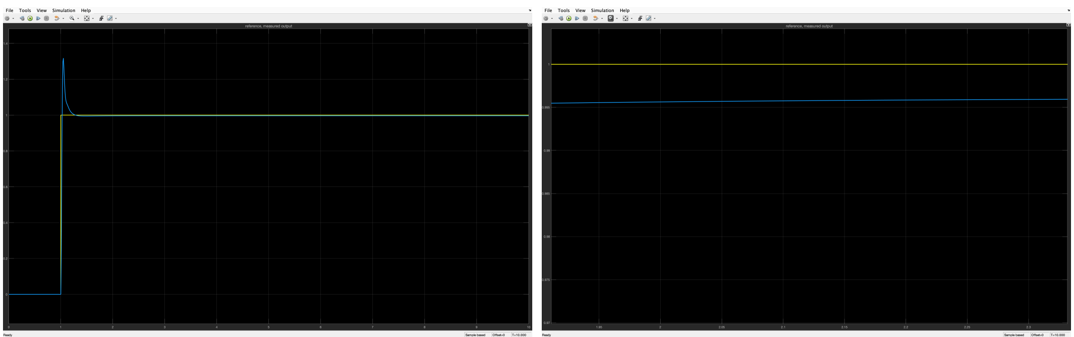

Figure 15: Response for scheme on Fig.11. Steady state error is less than 0.005%

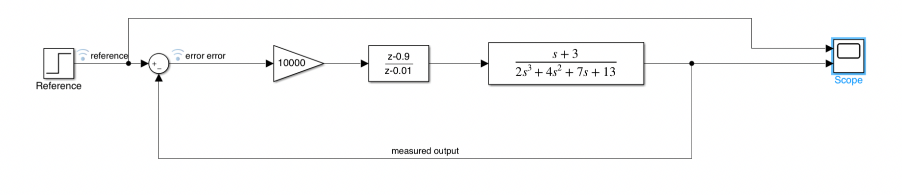

Figure 16: Scheme with compensator and gain

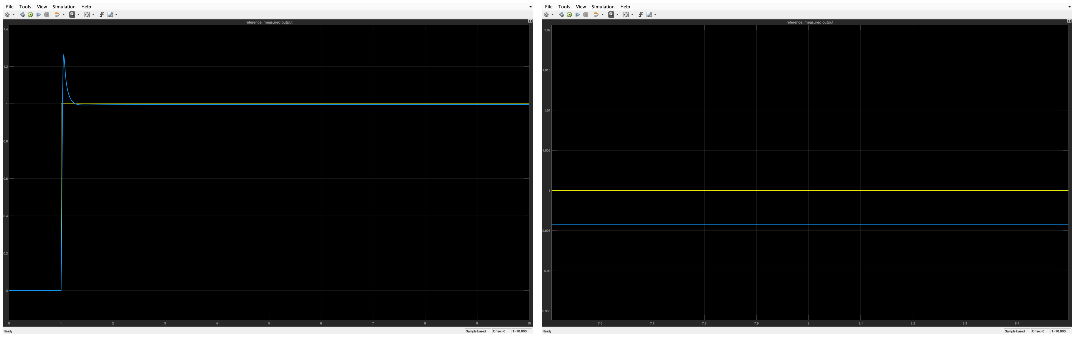

Figure 17: Response for scheme on Fig.13. Steady state error is still less than 0.005%

All thing considered, I choose gain=10000, zero of compensator=0.9, zero of compensator=0.01 as optimal parameters.