Exercise 5.7. Resolution of the pole placement equation[1]

We will illustrate here the resolution of the pole placement equation when the system only has a single input. We consider the system:

$$\dot{\mathbf{x}} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \mathbf{x} + \begin{pmatrix} 1 \\ 2 \end{pmatrix} \mathbf{u}$$

that we are looking to stabilize by state feedback of the form: $u = w - \mathbf{K}\mathbf{x}$, with: $\mathbf{K} = \begin{pmatrix} k_1 & k_2 \end{pmatrix}$. Calculate $\mathbf{K}$ so that this characteristic polynomial $P_{\text{con}}(s)$ of the closed-loop system has the roots -1 and -1.

Solution of Exercise 5.7

Analytic solution

The characteristic polynomial $P_{\text{con}}(s)$ of the closed-loop system has the roots -1 and -1 if it has the following form

$$P_{\text{con}}(s) = (s+1)(s+1) = s^2 + 2s + 1.$$

The pole placement equation $\det(s\mathbf{I} - \mathbf{A} + \mathbf{B}\mathbf{K}) = P_{\text{con}}(s)$ is

$$\det\left(\begin{pmatrix} s & 0 \\ 0 & s \end{pmatrix} - \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} + \begin{pmatrix} 1 \\ 2 \end{pmatrix} \begin{pmatrix} k_1 & k_2 \end{pmatrix} \right) = s^2 + 2s + 1,$$

that is,

$$\det \begin{pmatrix} s + k_1 - 1 & k_2 - 2 \\ 2k_1 - 3 & s + 2k_2 - 4 \end{pmatrix} = s^2 + 2s + 1,$$

or

$$s^2 + s(k_1 + 2k_2 - 5) + k_2 - 2 = s^2 + 2s + 1.$$

We obtain the following linear system

$$\begin{aligned} k_1 + 2k_2 - 5 &= 2 \\ k_2 - 2 &= 1 \end{aligned}$$

whose solution is $k_1 = 1, k_2 = 3$.

Numerical solution using `Matlab`

Let's check the controllability of the system

[1]Adapted from https://www.ensta-bretagne.fr/jaulin/automooc.pdf

```
clear all
close all
clc

  A = [1  2;
       3  4];

B = [1;
     2];

Co = ctrb(A,B);

rank(Co)
```

We obtain

```
ans =
     2
```

The system is controllable. Let's check the stability of the system

```
clear all
close all
clc

  A = [1  2;
       3  4];

B = [1;
     2];

eig(A)
```

We get

```
ans =

   -0.3723
    5.3723
```

The system is unstable. Let's check this by plotting the step response of the system after choosing $y_1(t) = x_1(t)$ and $y_2(t) = x_2(t)$, which means

$$\mathbf{C} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \ \mathbf{D} = \mathbf{0}.$$

```
clear all
close all
clc

  A = [1  2;
       3  4];

B = [1;
     2];

C=[1 0;
      0 1];

D=0;

sysA=ss(A,B,C,D);
t = (0:0.1:1)';

yA=step(sysA,t);

figure

plot(t,yA),grid
xlabel('t'),
ylabel('y(t)')
legend('Response y(t) to u(t) step')
```

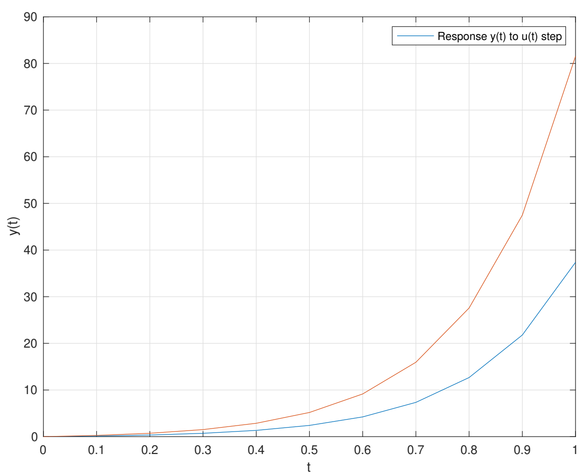

Figure 1: Step response of the system with original poles obtained with the `Matlab` predefined function `step`.

Pole assignment problems can be solved using the `Matlab` predefined function `place`. Notice that the function `place` does not admit poles with multiplicity greater than the rank of matrix **B**. This means that it is not able to solve this instance of the problem in which the multiplicity of the pole in -1 is 2 and the rank of matrix **B** is one.

Therefore, we change the value of one of the poles to be assigned which become -1 and -1.5.

The `Matlab` code to find the matrix $\mathbf{K}$ that places the poles of the system in $-1, -1.5$ is

```
clear all
close all
clc

  A = [1  2;
       3  4];

B = [1;
     2];


p = [-1, -1.5];

K = place(A,B,p)
```

The result is

```
K =
    0.5000    3.5000
```

Let's check that the new eigenvalues are -1 and -1.5

```
clear all
close all
clc

  A = [1  2;
       3  4];

B = [1;
     2];

C=[1 0;
      0 1];

D=0;

sysA=ss(A,B,C,D);

t = (0:0.1:1)';

yA=step(sysA,t);

figure

plot(t,yA),grid
xlabel('t'),
ylabel('y(t)')
legend('Response y(t) to u(t) step')

K= [0.5 3.5]

F= A - B*K

eig(F)
```

We get

```
ans =

   -1.0000
   -1.5000
```

Finally let's plot the new step response

```
clear all
close all
clc

  A = [1  2;
       3  4];

B = [1;
     2];

C=[1 0;
      0 1];

D=0;

sysA=ss(A,B,C,D);

t = (0:0.1:1)';

yA=step(sysA,t);

figure

plot(t,yA),grid
xlabel('t'),
ylabel('y(t)')
legend('Response y(t) to u(t) step')

K= [0.5 3.5]

F= A - B*K

eig(F)

sysB=ss(F,B,C,D);

t = (0:0.1:10)';

yB=step(sysB,t);

figure

plot(t,yB),grid
xlabel('t'),
ylabel('y(t)')
legend('Response y(t) to u(t) step')
```

We get

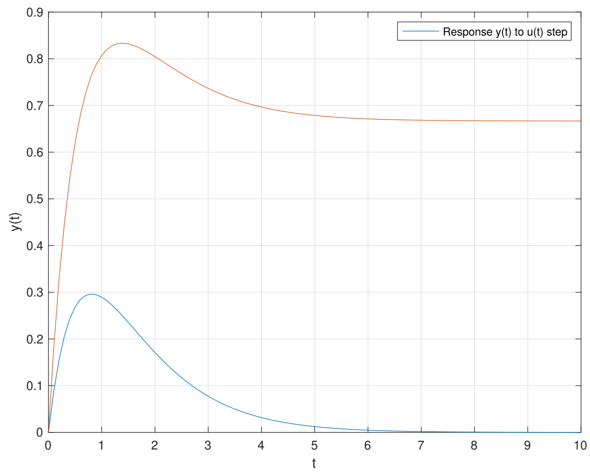

Figure 2: Step response of the system with assigned poles obtained with the `Matlab` predefined function `step`.

Simulate the step response of the system using Euler and Runge-Kutta methods