SSY281 Model Predictive Control

Assignment 6

Yongzhao Chen

March 13, 2023

1 (

Lyapunov functions)

1.1 a

Calculate the eignvalues of Q and A, results are here:

$$\begin{aligned} eignQ &= \begin{bmatrix} -1.2621 \\ -0.0013 \\ 0.0727 \\ 0.5586 \end{bmatrix} \\ eigenA &= \begin{bmatrix} 1.0000 \\ 1.0949 \\ 0.9126 \\ 0.9636 \end{bmatrix} \end{aligned} \tag{1}$$

We can see, there are negative values in $eighQ$, and not all the eignvalues of A are in the unit circle.

The system is not stable.

1.2 b

From the question, calculate new A_k and Q_b to judge the stability:

```
Ak = A-B*K;
Qb = S-Ak'*S*Ak
```

The result is:

$$\begin{aligned} eigQak &= \begin{bmatrix} 0.1894 + 0.0000i \\ 0.9156 + 0.0101i \\ 0.9156 - 0.0101i \\ 0.9901 + 0.0000i \end{bmatrix} \\ eigQb &= \begin{bmatrix} -4.3889 \times 10^3 \\ -0.0000 \\ 0.0002 \\ 0.0010 \end{bmatrix} \end{aligned} \tag{2}$$

I made two assistant functions:

```
function eignQ = checkQ(Q)
eignQ = eig(Q);
if all((eignQ) > 0)
    disp(sprintf("All eig of %s is positive.", inputname(1)
       ));
else
    disp(sprintf("Not all eigenvalues of %s are positive.",
        inputname(1)));
end
end


function eigenA = checkA(A)
eigenA = eig(A);
if all(abs(eigenA) < 1)
   disp(sprintf("All eig of %s is smaller than 1.",
      inputname(1)));
else
    disp(sprintf("Not all eigenvalues of %s are smaller
       than 1",inputname(1)))
end
end
```

We can discern that all eigenvalues of matrix A are situated inside the unit circle. Additionally, matrix Q contains some negative eigenvalues. Based on the eigenvalues, we can ascertain that three out of four of them are in close proximity to the boundary of the unit circle. Hence, the system should be stable, but we need to verify it.

1.3 c

Calculate new S_c using `dlyap`:

```
Sc = dlyap(Ak',Q)
```

Check its eigenvalues, the results are:

$$eignSc = \begin{bmatrix} 0.0001 \times 10^3 \\ 0.0010 \times 10^3 \\ 0.0043 \times 10^3 \\ 2.7319 \times 10^3 \end{bmatrix} \tag{3}$$

Now the result verifies the system is stable.

2 Stability with receding horizon control

2.1 a

The cost function is:

$$V_N(x) = x(N)'P_f x(N) + \sum_{i=0}^{N-1} (x(i)'Qx(i) + u(i)'Ru(i))$$

where $Q = I_4$ and $R = 1$. When $N = 1$, the influence of the matrix Q on the system's stability can be examined.

With $N = 1$, the cost function can be simplified to:

$$V_1(x) = x(1)'Qx(1) + u(0)'Ru(0)$$

To find a control input sequence $u(k)$ that minimizes the cost function based on the optimization objective, we can use recursive equations to calculate $u(k)$. Specifically, we can represent the cost function as:

$$V_N(x(k), u(k), P) = x(k)'Qx(k) + u(k)'Ru(k) + V_{N-1}(x(k+1), u(k+1), P)$$

where $V_{N-1}(x(k+1), u(k+1), P)$ represents the cost function from time step $k+1$ to time step N, which depends on the state sequence $x(k+1 : N)$, control input sequence $u(k : N-1)$, and a matrix P. Then, we can use the recursive equation to calculate the control input sequence $u(k)$ that minimizes the cost function:

$$u(k) = -(R + B'P_{k+1}B)^{-1}B'P_{k+1}Ax(k)$$

Here, P_{k+1} represents the state-control gain matrix from time step $k+1$ to time step N, which can be computed recursively using the following equation:

$$\begin{aligned} P_N &= P_f \\ P_k &= Q + A'P_{k+1}A - A'P_{k+1}B(R + B'P_{k+1}B)^{-1}B'P_{k+1}A, \\ \text{where} k &= N-1, ..., 0 \end{aligned} \tag{4}$$

In this problem, $Q = I_4$, which means it contributes equally to the cost function for each state variable's square weight. A small value of Q emphasizes the controller's response speed to state variables, while a larger value emphasizes the stability of the state variables. Therefore, when $Q = I_4$, each state variable is given equal weight, and the system's stability is considered balanced.

2.2 b

Set $P_f = Q$, use loop to get result:

```
    for N = 1:100
    [P0,K] = DP(A,B,Pf,Q,R,N);
    Ak = A-B*K;
    if all(abs(eig(Ak))<1)
        disp("The shortest N is: ")
        disp(N);
        disp("The K is")
        disp(K);
        break;
    end
end
```

Results say, the shortest N is 38,and the P_0 and K correspond exactly to the solution of the Lyapunov equation and the gain matrix K for the discrete-time system of the RHC controller.

$$P0 = \begin{bmatrix} 3.0334\times10^4 & 0.3348\times10^4 & -0.0177\times10^4 & -0.0719\times10^4 \\ 0.3348\times10^4 & 0.0375\times10^4 & -0.0020\times10^4 & -0.0080\times10^4 \\ -0.0177\times10^4 & -0.0020\times10^4 & 0.0039\times10^4 & 0.0005\times10^4 \\ -0.0719\times10^4 & -0.0080\times10^4 & 0.0005\times10^4 & 0.0019\times10^4 \end{bmatrix} \tag{5}$$
$$K = \begin{bmatrix} 45.6628 & 5.1672 & -0.0068 & -0.3898 \end{bmatrix}$$

2.3 c

Calculating the new Qc from the P_0 and K above, result is:

$$Qc = \begin{bmatrix} -1.6635\times10^4 & -0.1870\times10^4 & 0.0020\times10^4 & 0.0169\times10^4 \\ -0.1870\times10^4 & -0.0209\times10^4 & 0.0002\times10^4 & 0.0019\times10^4 \\ 0.0020\times10^4 & 0.0002\times10^4 & -0.0000\times10^4 & -0.0000\times10^4 \\ 0.0169\times10^4 & 0.0019\times10^4 & -0.0000\times10^4 & -0.0001\times10^4 \end{bmatrix} \tag{6}$$

If we were dealing with the Lyapunov equation, then Q_c should be positive semi-definite.

In this part I find that using `issymmetric` and it eigenvalue to check Q_c is not the best way. Use `chol` function to do that is the fastest. Anyway, I still use eigenvalues to analyse.

The eigenvalues of Q_c is:

$$ans = \begin{bmatrix} -1.6847\times10^4 \\ 0.0001\times10^4 \\ 0.0001\times10^4 \\ -0.0000\times10^4 \end{bmatrix} \tag{7}$$

If we were dealing with the Lyapunov equation, then Q_c should be positive semi-definite. However, from the eigenvalues above, now it is apparent that it is not, so V is not a Lyapunov equation.

2.4 d

The key code for this part is listed:

```
[Pf, ,K] = dare(A,B,Q,R)
N = 1;
% [P0,K] = DP(A,B,Pf,Q,R,N); % same K as above
Ak = A-B*K;
Qc = Ak'*Pf*Ak-Pf+Q+(K'*B'*R*B*K)
[ ,flag] = chol(Qc)
checkQ(Qc)
```

The flag says Qc is not correct for Lyapunov equation(mentioned in c), Check the eigenvalues again:

$$ans = \begin{bmatrix} -3.0965 \times 10^3 \\ -0.0000 \times 10^3 \\ 0.0000 \times 10^3 \\ -0.0000 \times 10^3 \end{bmatrix} \tag{8}$$

so V is still not a Lyapunov equation

3 Receding horizon controller example

4 Stability for constrained systems