Handed out: October 30th, 2015 **Due: November 6th, Friday**

16.30/16.31 Problem Set 3

1 LQR Control Design

This problem is designed to give you insight into the LQR control design process. Consider the following one-state system.

$$\dot{x}(t) = x(t) + u(t)$$

where $x(t) \in \mathbb{R}$. We would like to design a controller of the form: $u(t) = -Kx(t)$. Please answer the following questions. Please be careful with your analytical calculations as you integrate and differentiate functions. Please feel free to use Matlab symbolic toolbox, Mathematica, or Maple.

1. For what values of K is the closed-loop system stable? *Please write down the dynamics governing the closed-loop system. Then, use the eigenvalue stability criterion to assess stability.*

2. What is the response of the closed loop system $x(t)$ when we start the system from initial condition $x(0) = 1$? *Please use the matrix exponential to write down the system response.*

3. Let us define the state cost as follows:
$$J_x = \int_0^\infty \alpha\, x^2(t)dt.$$
where α is a scalar constant. What is J_x as a function of K and α? *Please use the analytical form of $x(t)$ from part (2) to compute J_x. You will need to integrate an exponential function.*

4. Let us define the input cost as follows:
$$J_u = \int_0^\infty u^2(t)dt.$$
What is J_u as a function of K? *Please use the fact that $u(t) = -Kx(t)$ and use the analytical form of $x(t)$ from the part (2) to compute J_x. You will need integrate an exponential function.*

5. Now, let the net cost function be:
$$J = J_x + J_u.$$
What is the optimal gain K that minimizes J? Please directly work with the results you obtained in Parts (3) and (4) to find the minimizing K as a function of α. Please do not use the Riccati equation. *Please use the "optimality condition for one variable." That is, directly solve K from the equation: $\partial J/\partial K = 0$. You will need to differentiate K, and then you will need to solve for the roots of an equation that is quadratic in K. The variable α will be a parameter in this quadratic equation. Choose the root that guarantees stability.*

6. Now consider the following cost function:
$$J_{\text{LQR}} = \int_0^\infty (\alpha x^2(t) + u^2(t))dt$$
Please use the Riccati equation to find the optimal K as a function of α.

7. How does K change as a function of α? Consider the two cases: (i) as $\alpha \to \infty$ and (ii) as $\alpha \to 0$. Did you expect K to change as you found out? Please explain in a few sentences.

2 LQR Design for a Drone Hover Controller

Recall that in PSET 2, we represented the drone dynamics in a way that the dynamics became separable. This allowed us to design a separate controller for each independent loop using pole placement.
In this PSET, we choose the input the our drone system to be the thrust of each of the four propellers (compare this to the last PSET where we used $(\mathbf{T}, \tau_y, \tau_p, \tau_r)'$). We will be able to setup the control design problem as an optimization problem of special form so that we can apply the LQR framework.

2.1 Drone Dynamics

Let us define some notation first:

- $\mathbf{P}$ are inertial XYZ-world coordinates

$$\mathbf{P} = (X, Y, Z)'$$

  of the drone's center of mass.

- $\dot{\mathbf{p}}$ is the quadroter's velocity wrt. the world coordinate frame, expressed in the body frame

$$\dot{\mathbf{p}} = (\dot{x}, \dot{y}, \dot{z})'$$

- $\mathbf{O}$ is the euler-angles $(yaw, pitch, roll)$

$$\mathbf{O} = (\psi, \theta, \phi)'$$

  relating to a rotation R_{B2W} to translate a vector expressed in the drone's bodyframe to world coordinates by subsequent rotations about euler-roll about x, pitch about y, yaw about z.

- W^{-1} transforms the body-angular rates

$$\dot{\mathbf{o}} = (p, q, r)'$$

  about local x-y-z-axes to euler-rates

$$\dot{\mathbf{O}} = (\dot{\psi}, \dot{\theta}, \dot{\phi})'$$

- Let $\mathbf{G}$ be the gravitational vector expressed in world coordinates and $\mathbf{J}$ the quadrotor's inertia expressed in the body frame.

Now, let us define the state variables and the input variables:

- The state variables vector, $\mathbf{x}$, is defined as follows:

$$\mathbf{x} = \begin{bmatrix} \mathbf{P} \\ \mathbf{O} \\ \dot{\mathbf{p}} \\ \dot{\mathbf{o}} \end{bmatrix}$$

- The input variables are chosen to be the commands sent to the motors. These commands are proportional to thrust of each of the motors, so let's call these commands T_i.

$$\mathbf{u} = \mathbf{u}_{PSET3} = \mathbf{u}_{\text{commands}} = [T_1, T_2, T_3, T_4]'$$

Note that, assuming that the total thrust is fully aligned with the body-z-axis, there is a linear transformation that transforms the system input in PSET 2 $\mathbf{u}_{PSET2} = [\mathbf{T}, \tau_y, \tau_p, \tau_r]'$ to the system input in this PSET $\mathbf{u}_{\text{commands}} = [T_1, T_2, T_3, T_4]'$ via a matrix $\mathbf{M}$.
So we have

$$\begin{aligned} \mathbf{u}_{PSET2} &= \mathbf{M}\,\mathbf{u}_{commands} \\ \text{or, in terms of the individual inputs,} \\ T &= \mathbf{M}_{(1,:)}\ \mathbf{u}_{\text{commands}} \\ \tau_y &= \mathbf{M}_{(2,:)}\ \mathbf{u}_{\text{commands}} \\ \tau_p &= \mathbf{M}_{(3,:)}\ \mathbf{u}_{\text{commands}} \\ \tau_r &= \mathbf{M}_{(4,:)}\ \mathbf{u}_{\text{commands}} \end{aligned}$$

We can substitute these equations into the dynamics equations from PSET 2 and we will get drone dynamics of the form

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}_{\text{commands}}, t).$$

Now, recall that a linearized plant is of the form

$$\triangle\dot{\mathbf{x}} = \mathbf{A}\triangle\mathbf{x} + \mathbf{B}\triangle\mathbf{u}$$

This form allows us to set up the design of a full-state feedback controller K as an optimization problem that can be solved using the LQR framework.
We define the cost functional as

$$J = \int_0^\infty \mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u}\,dt$$

We aim to minimize this cost functional subject to the dynamics $\triangle\dot{\mathbf{x}} = \mathbf{A}\triangle\mathbf{x} + \mathbf{B}\triangle\mathbf{u}$. With slight abuse of notation we will refer to these dynamics as $\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$ in the remainder of this document.

2.2 Linearization, Dynamic Analysis, and Feedback Control

1. Please update your toolbox by downloading the toolbox from https://github.com/Parrot-Developers/RollingSpiderEdu (or pulling the current version if you use git).

2. Conceptually, explain the relation between optimizing the chosen cost functional and the actual goal of designing a hover controller for the drone system.

3. Now let us find the linear plant that we will use to compute the LQR controller. To this end, we will linearize the complex drone model from the Simulink simulation with the inputs being the motor commands T_i.

   Please use the linearizeDrone_motorcmdTostate.slx and Simulink's ControlDesign/Linear Analysis tool to find the A,B,C,D matrices around hover condition for this complex model.
   What are the A an B matrices?

4. Please look at the following Matlab file in the toolbox:
   /trunk/matlab/Simulation/controllers/controller_fullstate/LQRControl.m.
   This file provides a template for designing the LQR hover controller, with preset values for the bounds in Bryson's rule and cost weights for state error and control effort.
   Comment on the choice of the preset bounds and weights. Are they intuitive? Are they more intuitive than choosing eigenvalues for pole placement?

5. Use the LQRControl.m-script and your results from the linearization in 2.2.3 to compute K_{lqr}. Then use Simulink to simulate the LQR-controller with the linear plant model that you found in 2.2.3.
   An easy way to implement the effect of having modeled the system around our equilibrium point is to simply set the initial condition of the system to $-\mathbf{x}_{eq}$, i.e.
   $\mathbf{x_0} = 0, 0, 1.5, 0, 0, 0, 0, 0, 0, 0, 0, 0$ and then have the system being controlled into the origin.
   Please plot positions and orientations.

   Note that this simulation will not involve the complex nonlinear drone model of our Matlab toolbox, which we use for the labs. This is a simple simulation, where the plant model is a simple linear system with A and B matrices found in 2.2.3.

6. Now, try to tweak the bounds and cost weights to achieve a faster altitude response of the drone. Keep in mind that the control inputs into the plant are the four motor commands, which are proportional to the thrust provided by the motors. The LQR-optimization tries to minimize costs caused by these commands. Gaining altitude (hence changing the z-state) requires all four motors to provide considerably more thrust while changing e.g. the roll angle can be achieved by little differences in thrusts. To considerably change the altitude response you should therefore mainly tweak the overall control effort costs ρ and the weight on the state z.
   Use Matlab/Simulink to simulate this second controller with the linear model that you found in 2.2.3.
   Please plot positions and orientations, comment on your experience in finding new suitable cost weights and describe the bounds and weights you chose.
   Note that this simulation will not involve the complex nonlinear drone model of our Matlab toolbox, which we used for LAB 1. This is a simple simulation, where the plant model is a simple linear system with A and B matrices found in 2.2.3 of this problem.

3 Time spent

Please indicate the approximate amount of time you spent on this assignment.