WASP Autonomous Systems Course – Laboratory Exercises

Daniel Axehill and colleagues

September 9, 2016

1 Recommended software

- Matlab, including Control System Toolbox
- ACADO (`http://acado.github.io/`)
- Lab files: `https://gitlab.ida.liu.se/wasp_cdm/labs.git`

2 Tasks

1. **PID control of the TurtleBot's DC motors** The TurtleBot is equipped with two electrical motors that can be used to give the robot different longitudinal velocities as well as rotational speeds. In order to be able to obtain good performance of the robot, we would like to control the angular velocities of the motors. The input to the motor is the applied voltage and the measured output is the motorshaft's angular velocity.

   A model of the DC motor is

$$J\ddot{\theta} + b\dot{\theta} = Ki \tag{1a}$$

$$L\frac{di}{dt} + Ri = V - K\dot{\theta} \tag{1b}$$

   (a) Derive the transfer function from the input voltage V and the angular velocity $\omega = \dot{\theta}$.

   (b) Assume $J = 0.01$, $b = 0.1$, $K = 0.01$, $R = 1$, and $L = 0.5$. Design a PID controller such that the rise time is $0.02\,\mathrm{s}$ and the overshoot not more than $2\,\%$. The static control error should be zero.

   (c) Plot the sensitivity function and the complementary sensitivity function. What are the interpretations of these two functions (in general)?

   (d) How much are the gain margin and the phase margin, respectively? What does that mean in terms of robustness?

2. **Path following control for a TurtleBot** Consider the TurtleBot robot used in the Summer School. A model of the TurtleBot can be expressed by a differential wheel drive model

$$\dot{x} = v\cos\theta \tag{2a}$$

$$\dot{y} = v\sin\theta \tag{2b}$$

$$\dot{\theta} = u \tag{2c}$$

where (x, y) is global position, θ is global orientation, v is longitudinal velocity and u is angular velocity (which depends on control signal to the right and left wheel) of the TurtleBot respectively. Assume a path has been generated by a path planner with constant longitudinal velocity $v = a$. The objective in this task is to design an LQ controller to be able to follow this path even with disturbances acting on the system. An error model for lateral and angular error dynamics of the TurtleBot around this path can be expressed in the Frenet frame as

$$\dot{z} = a\sin\tilde{\theta} \tag{3a}$$

$$\dot{\tilde{\theta}} = u - u_0(s) \tag{3b}$$

where z is the lateral distance from the projection of the TurtleBot to the desired path, $\tilde{\theta}$ is the orientation error with respect to the path and $u_0(s)$ is the nominal angular velocity at this specific projection. Denote the state vector $\mathbf{p} = \left[z, \tilde{\theta}\right]^T$ then (3) can be expressed as $\dot{\mathbf{p}} = f(\mathbf{p}, u)$.

   1. Show that $(\mathbf{p}, u) = (\bar{0}, u_0)$ is an equilibrium to $\dot{\mathbf{p}} = f(\mathbf{p}, u)$.
   2. Linearize the model (3) around this equilibrium to get a linear model on the form $\dot{\mathbf{p}} = A\mathbf{p} + B\tilde{u}$, where $\tilde{u} = u - u_0$.
   3. Use Matlab to design an LQ-controller, $\tilde{u} = -L\mathbf{p}$, based on the linearized model and implement this in the code that has been distributed. You only need to add some lines in the file `Runme.m`.
   4. Show why it is, without loss of generality, valid to put $Q_2 = 1$. This result is only valid for systems with a single input.
   5. Choose $Q_2 = 1$ and design a controller for $Q_1 = \text{diag}\begin{pmatrix}10 & 1\end{pmatrix}$, $Q_1 = \text{diag}\begin{pmatrix}1 & 10\end{pmatrix}$ and $Q_1 = \text{diag}\begin{pmatrix}10 & 10\end{pmatrix}$ and compare the closed-loop system behavior. Can you explain these different behaviors?
   6. **Optional** Choose an output y such that the relative degree becomes 2 and compute a feedback control law that feedback linearize (3). Apply LQ control to the obtained linear system.

3. **Optimal Control using ACADO** In this part we will again consider the model of the TurtleBot

$$\dot{x} = v\cos\theta$$

$$\dot{y} = v\sin\theta$$

$$\dot{\theta} = u$$

but this time the lateral velocity v is not fixed and instead considered as a control signal together with u.

We will consider two problems: energy optimal control and time optimal control of the TurtleBot. ACADO will be used to solve the problems, so make sure you have ACADO installed on your computer before continuing (see link in the beginning of the document).

(a) Energy Optimal Control of a TurtleBot using ACADO

The task here is a minimum energy problem of moving the TurtleBot from one state to another. The problem is stated as follows: at $t = 0$ the TurtleBot is at rest at $(x, y, \theta) = (0, 0, 0)$. At $t = 10$ it should be at $(x, y, \theta) = (-1, 0, \pi)$, and the goal is to minimize the control energy it takes to get there (i.e., $\int_0^{10} u^2(t) + kv^2(t)dt$). Restrictions apply on the lateral velocity ($|v| \leq 0.7$) and the angular velocity ($|\dot{\theta}| \leq \pi$).

1. Now we will start looking at the optimal control problem. What is the mathematical definition of the objective function?
2. What are the constraints for the problem?
3. Summarize once again the objective and the constraints and write them down in the optimal control problem standard form.
4. Next implement and solve the optimal control problem using ACADO. It might help to briefly browse through the ACADO online tutorials at www.acadotoolkit.org to look up the notation. A starting point is available in `AS_ACADO_skeleton.m`.
5. Play with the different parameters, for example k or the end time T and see how the solution changes character. You can also add more constraints, such as the available space.

(b) Time Optimal Control of a TurtleBot using ACADO

In this part we use the same model of the TurtleBot as in the previous exercise, but the aim here is to move the TurtleBot from $(x, y, \theta) = (0, 0, 0)$ to $(x, y, \theta) = (-1, 0, \pi)$ as fast as possible. The same restrictions on lateral and angular velocities apply.

1. As specified we would like to move the TurtleBot as fast as possible between the specified points. What is the mathematical definition of the objective function?
2. Summarize once again the objective and the constraints and write them down in the standard formulation of optimal control problems.
3. Solve the above problem optimal control problem in ACADO. You can use the code template `AS_ACADO_skeleton.m` to learn the syntax.
4. What is the result for the minimum time? Can you interpret the result for the optimal control input? Compare with the results from when optimizing for minimum energy.

(c) **Optional** Extend the model of the TurtleBot with an inverted pendulum on top that can fall in any direction. Apply ACADO to the problem and see if it is able to bring the pendulum to the upright position starting with small off-sets from the upright position. Play with different lengths of the pendulum and input constraints on the TurtleBot.

Literature

[1] Torkel Glad and Lennart Ljung. *Control Theory – Multivariable and Nonlinear Methods*. Taylor & Francis, 2000.

[2] Torkel Glad and Lennart Ljung. *Reglerteori – Flervariabla och olinjära metoder*. Studentlitteratur, 2003.

[3] Torkel Glad and Lennart Ljung. *Reglerteknik – Grundläggande teori*. Studentlitteratur, 2006.