EE106/206 HW7: Modelling, Control, Parameter Identification, and State Estimation

Robert Matthew

Due: Tuesday, November 6, 2018 @ 11:59 PM

Problem-sets are due as a PDF on Gradescope. Feel free to use a computer to help you with this problem set. If you do write any code, to help you solve the problem, attach the code at the end of your problem set. If you use any pre-made code (such as MATLAB's pseudo-inverse function pinv()), state that you use it as a step in your solution.

For this problem set, consider the single pendulum shown in Figure 1.

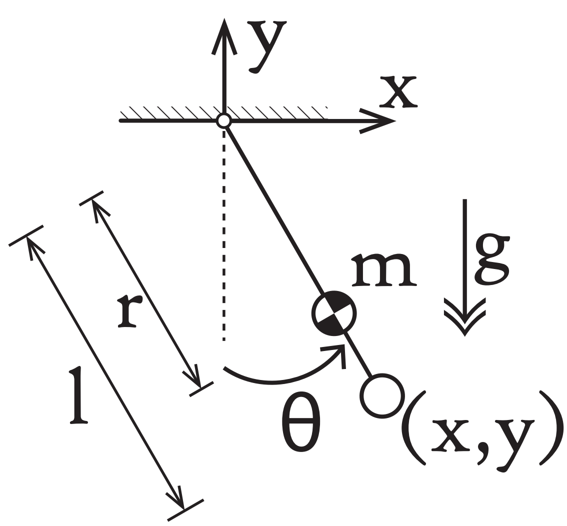

Figure 1: Single pendulum under motion capture. Pendulum consists of a pivot at position T_x, T_y, and a mass m at distance r from the pivot. A motion capture dot is located along the same line, at a distance l from the pivot.

Question 1: Modelling & Control .... 5 points

(a) Derive observation equations for this system. This equation should relate the observed values (x, y) to the system state $(\theta \dots)$.

(b) Derive the continuous time dynamic equations of motion for this system. This equation should be of the form $\ddot{\theta} = h(t, \theta, \tau)$, where τ is your input torque.

(c) Write the continuous time dynamic equations in the form:

$$\dot{\boldsymbol{X}} = f(t, \boldsymbol{X}) + g(t, \boldsymbol{X}, \tau)$$

where $\boldsymbol{X}$ is a vector of your system states $(\theta \dots)$.

(d) For the free case $(\tau = 0)$, simulate the evolution of this system and recover the angular positions and velocities overtime. (Use *ode45* in MATLAB or *scipy.intergrate odeint* in Python).

(e) Using these angular positions and velocities, recover the angular acceleration.

(f) From the angular state, find the corresponding (x, y) coordinates of the motion capture marker.

(g) Plot this (x, y) points on a graph.

(h) In reality, motion capture has noise. Simulate this by adding gaussian noise to each of these points. Plot this new synthetic motion capture data.

(i) Add the following controller to your system:

$$\tau = -mr^2 \left(\alpha (\theta - \theta_d) + \beta \dot{\theta}\right)$$

Simulate this system for different values of α, β, and θ_d. Plot the state trajectories over time for your controller.

You may complete some or all of the following problems:

Question 2: OPTIONAL: Control .... 5 points

(a) The controller implemented in question 1 is quite poor. Research and implement a better controller for the single pendulum. This pendulum should settle to the position $\theta = \frac{7}{8}\pi$ with a steady state error of less than 0.01 radians as fast as possible. Plot the state trajectories over time for your controller.

Question 3: OPTIONAL: Parameter Identification .... 5 points

(a) As seen in the slides, Parameter Identification can be used to recover system parameters. Use a non-linear least square solver to recover the recoverable system parameters from your noisy synthetic motion capture data. (Use *lsqnonlin* in MATLAB or *lmfit* in Python). Compute the error in your recovered parameters.

Question 4: OPTIONAL: State Estimation .... 5 points

(a) As seen in the slides, State Estimation can be used to recover both the system states and parameters. Use an unscented Kalman filter to recover the state and any recoverable parameters from the noisy motion capture data in the case where a torque controller is added. (Use *Learning the Unscented Kalman Filter* (from file exchange) in MATLAB or *pykalman* in Python). Assume that the parameters that can be found though parameter identification (T_x, T_y, l) are known. Tune your covariance matrices and plot the error between your recovered and true states.