Prelab 5: LQR controller design for inverted pendulum robot

The goals of this prelab are (1) to become familiar with MATLAB commands such as "ss" (to create a state space system), "lqr" (to calculate LQR gain matrix K), and "initial" (to simulate an initial condition response, given some initial state, for your closed-loop system), and (2) to write initial code in MATLAB that calculates the A and B matrices for your model of the balancing robot, based of various parameters.You should also download two m-files, available next to the link for this assignment on the course "Laboratory" web page, and the class notes from Lectures 13 and 14, (which are also on the "Lectures" page). These files are called:

segway_eom.m – This file calculates the equations of motion of the robot, symbolically.

sample_lqr.m – This file demonstrates how to find a gain matrix, K, that satisfies the LQR optimization problem, as discussed in Lecture 15.

Lecture13and14_withMATLAB.pdf – See especially pages 6 and 7 of the pdf. (These pages have hand labels "4" and "5" at the bottom, due to extra pages at the start of the pdf…)

Problem 1) Using the provided code to help, write an m-files that does all of the following tasks below, in an automated way. (i.e., if you change parameter values in lab, your code must be able to recalculate everything automatically.) For all tasks below, you states must be defined in the following order: $X = [\phi_w \quad \theta_b \quad \dot{\phi}_w \quad \dot{\theta}_b]^T$

(Note the "transpose" at upper right of expression above. X is 4x1 column vector.)

a. Calculate the A and B matrices for a state space representation of the (open-loop) inverted pendulum robot dynamics. This simply requires transforming the two equations of motion output from "segway_eom.m" into state space format. As discussed in class, you may wish to use an intermediate matrix "M", which will be invertible. Note, the bottom of page 7 of "Lecture13and14_withMATLAB.pdf" gives approximate values for the A and B matrices, but your values should be <u>somewhat different</u>, based on the parameters we are giving you in this prelab! The sign of terms in B depends on the arbitrary wiring of the Lego motor and should match the signs shown on page 7.
b. In MATLAB, start with Q=diag([1 0 0 0]) and R=[1]. Following the procedure in "sample_lqr.m", produce plots of an initial condition response for the initial state below:
   X0 = [0, 5*pi/180, 0, 0]'
   In other words, the body is initially tilted at 5 degrees, but all other states are zero. As in "sample_lqr.m", produce plots for all 4 states and ALSO for the control effort, u. u should be calculate after the output has been calculated. Recall that our control law requires that "u=-K*x". (In "sample_lqr.m", we assume output y equals the state vector x; y and x are just the same thing, because matrix C is the identity matrix…)
c. Now, Repeat this for the same Q, but with R=[0.1];
d. Again repeat, but now use R=[10.0];
e. Calculate open-loop poles and closed-loop poles for each case (a-d) above.
f. Comment on the effect of changing R both on the states over time and on the magnitude of the control output, u.