Norwegian University of Science and Technology
Department of Engineering Cybernetics

TTK4195
Modeling and Control of Robots
Spring 2015

Exercise set 8

Hand-in date: March 17th, 23:00

The following problem are found in the physical book: "Robot Modeling and Control" (2006) by Spong et al.

1. Problem 8-12

Consider the coupled nonlinear system

$$\ddot{y}_1 + 3y_1y_2 + y_2^2 = u_1 + y_2u_2$$
$$\ddot{y}_2 + \cos(y_1)\dot{y}_2 + 3(y_1 - y_2) = u_2 - 3(\cos y_1)^2 y_2 u_1$$

where u_1, u_2 are the inputs and y_1, y_2 are the outputs.

(a) What is the dimension of the state space?

(b) Choose state variables and write the system as a system of first order differential equations in state space.

(c) Find an inverse dynamics control so that the closed-loop system is linear and decoupled, with each subsystem having natural frequency 10 radians anddamping ratio 1/2.

(2 Points)

Additionally, consider the following:

d) Derive the linearization (first-order approximation of Taylor expansion) of the nonlinear state space system $\dot{x} = f(x) + g(x)u$ about the desired operating point $\{x_d, u_d\}$ so that a linear system $\dot{z} = Az + Bw$ is obtained with $z = x(t) - x_d, w = u(t) - u_d$.
(2 points)

Simulation Problem

Consider again the planar RPR manipulator from Exercise 07, Problem 7-9. Create a simulation model in MATLAB from the computed equations of motion in Matrix form. Use the following model parameters:

- Masses $m_1 = m_2 = m_3 = 5$kg
- Link lengths $l_1 = l_2 = l_3 = 0.5$m
- Centers of mass l_{ci} lie in the middle of the links
- Inertia about the axis of rotation in x_0, y_0 - plane is calculated in the body attached frames by the following equation: $I_i = \frac{m_i l_i^2}{12}$ for a thin rigid rod.

The parameters for the provided MATLAB matrices $M(q), C(q, \dot{q}), G(q)$ in the file Matrix M, C, F for RPR are as follows

```
% Global variables in MATLAB
global m1 m2 m3;              % Masses
global l1 l2 l3;              % Link lengths
global lc1 lc2 lc3;           % Centers of mass
global In1yy In2zz In3zz;     % Moment of inertia

m1 = 5; m2 = 5; m3 = 5;
l1 = 0.5; l2 = 0.5; l3 = 0.5;
lc1 = 0.5/2; lc2 = -0.5/2; lc3 = 0.5/2;
In1yy = 1/12*(m1*(l1)^2);
In2zz = 1/12*(m2*(l2)^2);
In3zz = 1/12*(m3*(l3)^2);
```

Note the difference between "1" (one) and "l" (L) in the MATLAB code, as they look similar.

Choose appropriate set points and smooth trajectories (e.g. sinusoid, polynomials), respectively, and implement the following motion control methods:

1. Set-point regulation with PD control + gravity compensation. **(2 Points)**
2. Trajectory tracking with joint space inverse dynamics control. **(2 Points)**
3. Trajectory tracking with task space inverse dynamics control using the analytical Jacobian defined by
$$\dot{X} = \begin{bmatrix} \dot{x}_e \\ \dot{y}_e \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} * & * & * \\ * & * & * \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} \tag{4.102}$$
with x_e and y_e the Cartesian end-effector position and its orientation $\alpha = q_1 + q_3$, where first joint angle q_1 and third joint angle q_3 are relative angles. **(1 Point)**
4. Evaluate the closed-loop performance with measurement noise and possible parameter uncertainties such as unknown load. **(1 Point)**

Submit your executable simulation files for each step.