

MCE/EEC 647/747: Robot Dynamics and Control  
Term Project - Fall 2010  
Kinematics, Dynamics, Control and Simulation of Articulated Arm  
REVISED VERSION

Part I

Consider the articulated manipulator shown in Figure 1. This represents the first two links of a PUMA robot. The positions of the centers of mass of links 1 and 2 are as follows:

$$CM_1^0 = [0 \ 0 \ -a_1]^T \quad CM_2^2 = [a_2 \ 0 \ b_2]^T \quad (1)$$

Due to symmetry, the inertia tensors at the center of mass have the forms:

$$I_1 = \begin{bmatrix} I_{1x} & 0 & 0 \\ 0 & I_{1x} & 0 \\ 0 & 0 & I_{1z} \end{bmatrix} \quad I_2 = \begin{bmatrix} I_{2x} & 0 & 0 \\ 0 & I_{2y} & 0 \\ 0 & 0 & I_{2z} \end{bmatrix} \quad (2)$$

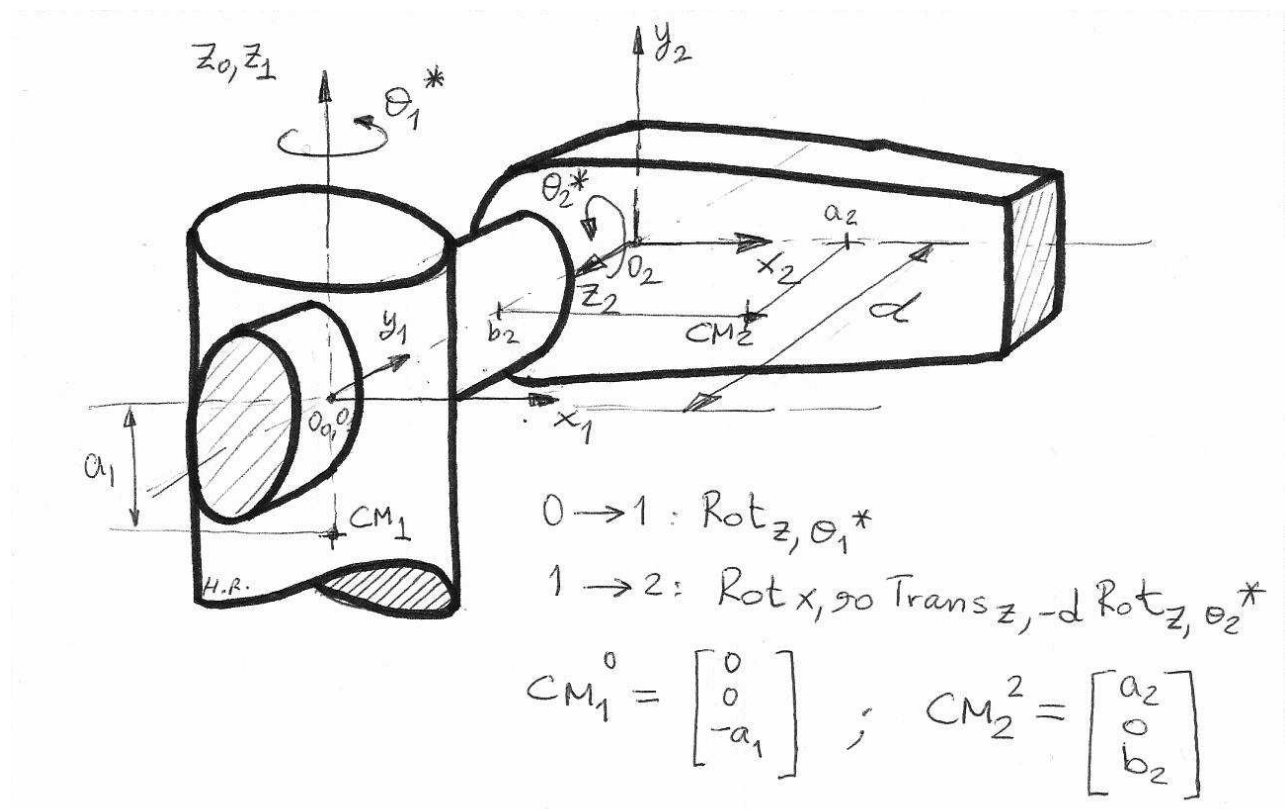


Figure 1:

## Kinematics and Dynamics

1. For uniformity in the solutions, use the coordinate systems of the figure to find the necessary transformation matrices.
2. Find the two velocity Jacobians and the two angular velocity Jacobians relative to each center of mass.
3. Find the  $D(q)$ ,  $C(q, \dot{q})$  and  $g(q)$  matrices.
4. Set up a Matlab/Simulink simulation implementing the dynamic equation  $D(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau$ . Please set up a parametric simulation system, that is, all parameters (masses, lengths, etc.) should be symbolic. Set up a separate m-file to load values for the parameters.
5. Try out your simulation by setting all parameters to 1 and applying  $\tau = [1 \ 1]^T$ . Simulate for 1 second and plot all joint positions and velocities against time.

## Linear Parameterization

1. Let the masses of the links be  $m_1$  and  $m_2$ , and let the joint angles be  $q_1$  and  $q_2$ . Show that the following regressor and parameter vector satisfy the dynamic equations derived above:

$$Y(q, \dot{q}, \ddot{q}) = \begin{bmatrix} \ddot{q}_1 & \cos^2(q_2)(\ddot{q}_1 + \ddot{q}_2) - 2\cos(q_2)\sin(q_2)(\dot{q}_1\dot{q}_2 + \dot{q}_2^2) & \ddot{q}_1 + \ddot{q}_2 & 0 \\ 0 & \cos^2(q_2)(\ddot{q}_1 + \ddot{q}_2) + \cos(q_2)\sin(q_2)(\dot{q}_1^2 - \dot{q}_2^2) & \ddot{q}_1 + \ddot{q}_2 & \cos(q_2) \end{bmatrix} \quad (3)$$

$$\Theta = \begin{bmatrix} I_{1z} \\ m_2 a_2^2 + I_{2y} - I_{2x} \\ I_{2x} + m_2(b_2 - d)^2 \\ m_2 g a_2 \end{bmatrix} \quad (4)$$

2. Find the regressor  $Y(a, v, q, \dot{q})$  corresponding to the term  $\hat{M}a + \hat{C}v + \hat{g}$
3. Consult with the instructor to see whether you're on the right track and ready for the next stage (control analysis/design).

## Part II

Design an adaptive-passivity based controller so that the following trajectory is tracked:

$$\begin{aligned} q_1^d(t) &= \sin t \\ q_2^d(t) &= -\sin t \end{aligned}$$

Take  $g = 9.8 \text{ m/s}^2$  and follow these instructions for finding the nominal values for the parameters:

- Refer to the 2 research papers appearing at the end of this document: Armstrong [1] and Corke [2].
- The coordinate system used in [1] can be used to figure out how the listed parameters correspond to the our coordinate system. For example, our  $a_2$  is  $r_x$  for Link 2. However, our  $b_2$  is  $-r_y$  in [1]. You need to figure out these correspondences for all parameters appearing in the model.
- You will not use the numerical values from [1], because of more accurate measurements carried out by Tarn. The parameters found by various people have been compiled in [2]. This paper uses the same coordinate system as in [1]. Extract the values of the parameters corresponding to Tarn's findings in [2]. The y-coordinate of the CG of link 2 should be taken as zero instead of the small value reported.
- Re-run the open-loop simulation with unit torque input and the real parameters, just to verify that the model can run with these numerical values.

You must pre-compute the desired velocities and accelerations to be used directly in the control algorithms (avoid online differentiation). Build a simulation using Matlab/Simulink to test each one of the following cases:

1. **With velocity sensing:** Choose suitable gain matrices  $\Lambda$ ,  $K$  and  $\Gamma$ . Use the true parameters in the plant and half their values as initial parameter guesses in the adaptation law. Run the adaptive system and tune the gain matrices to obtain good parameter and error convergence. Plot each adapted parameter vs. time. Plot  $\tilde{q}_1(t)$  and  $\tilde{q}_2(t)$ . Then plot the  $y$  position vs. the  $x$  position of the end-effector corresponding to  $q_1^d(t)$  and  $q_2^d(t)$  with a red dashed line. On the same plot, plot the actual  $y$  vs.  $x$  (using  $q_1(t)$  and  $q_2(t)$ ) with a solid line of another color. Finally plot the two components of the control input against time, along with the corresponding currents (use the torque constants from [2], Armstrong values). Did the parameters converge to their true values?
2. **With high-gain observer:** Repeat the above replacing the actual velocity sensing with the estimated velocity from a high-gain observer. As done in class, include a switch block allowing the user to toggle between sensed and estimated velocity. The sample file has been uploaded to the web site.

### Part III: Laboratory

1. Implement and tune a soft real-time PI controller for the waist of the PUMA 560 manipulator, as demonstrated in class. Use trapezoidal integration to discretize the I component.
2. Implement and tune a high-gain observer to estimate the waist velocity. Verify the accuracy of the estimation by external means (videocamera).
3. Use the velocity estimate to add D action to the controller.

**This is a guided part of the exam worth 20 points. These points will be awarded on the basis of attendance, participation and reporting**

### How to turn in the exam:

All hand calculations must be explained. Please scan any handwritten sheets. Email the instructor, attaching the hand calculations and m-file(s) with instructions for running the simulations. Students are encouraged to use symbolic engines (Matlab, Python, Mathematica, Maple, etc.) to assist with symbolic matrix multiplications and simplification, although it is not necessary to use these packages to carry the operations.

**Reports and Matlab files due on May 14, 2010 by 5PM**

### References

- [1] B. Armstrong, O. Khatib, and J. Burdick. The explicit dynamic model and inertial parameters of the PUMA 560 arm. In *1986 IEEE International Conference on Robotics*, volume 3, pages 510–518, 1986.
- [2] P. Corke and B. Armstrong. A search for consensus among model parameters reported for the PUMA 560 robot. pages 1608–1613, 1994.