

- [28] W. Schiehlen, "Computer Generation of Equations of Motion," in *Computer Aided Analysis and Optimization of Mechanical System Dynamics*, E.J. Haug, Editor, Springer-Verlag, Berlin & New York, 1984.
- [29] G. Cesaro, F. Nicolo, and S. Nicosia, "DYMIR: A Code for Generating Dynamic Model of Robots," in *Advanced Software in Robotics*, Elsevier Science Publishers, North-Holland, 1984.
- [30] J. Murray, and C. Neuman, "ARM: An Algebraic Robot Dynamic Modelling Program," IEEE International Conference on Robotics, Atlanta, March 1984.
- [31] M. Walker and D. Orin, "Efficient Dynamic Computer Simulation of Robotic Mechanisms," *ASME Journal of Dynamic Systems, Measurement, and Control*, Vol. 104, 1982.

## EXERCISES

- 6.1 [12] Find the inertia tensor of a right cylinder of homogeneous density with respect to a frame with origin at the center of mass of the body.
- 6.2 [32] Construct the dynamic equations for the two-link manipulator in Section 6.7 when each link is modeled as a rectangular solid of homogeneous density. Each link has dimensions  $l_i$ ,  $\omega_i$ , and  $h_i$  and total mass  $m_i$ .
- 6.3 [43] Construct the dynamic equations for the three-link manipulator of Chapter 3, Exercise 3.3. Consider each link to be a rectangular solid of homogeneous density with dimensions  $l_i$ ,  $\omega_i$ , and  $h_i$  and total mass  $m_i$ .
- 6.4 [13] Write the set of equations that correspond to (6.46)–(6.53) for the case where the mechanism could have sliding joints.
- 6.5 [30] Construct the dynamic equations for the two-link nonplanar manipulator shown in Fig. 6.8. Assume that all the mass of the links can be considered as a point mass located at the distal (outermost) end of the link. The mass values are  $m_1$  and  $m_2$ , and the link lengths are  $l_1$  and  $l_2$ . This manipulator is like the first two links of the arm in Exercise 3.3. Assume further that viscous friction is acting at each joint, with coefficients  $v_1$  and  $v_2$ .
- 6.6 [32] Derive the Cartesian space form of the dynamics for the two-link planar manipulator of Section 6.7 in terms of the base frame. *Hint:* See Example 6.5, but use the Jacobian written in the base frame.

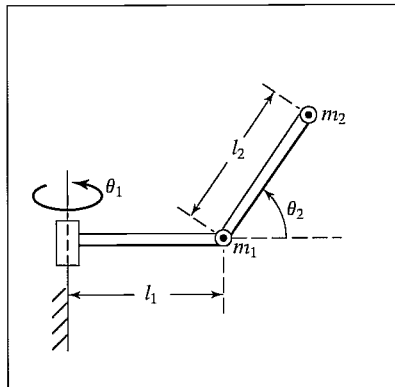


FIGURE 6.8: Two-link nonplanar manipulator with point masses at distal ends of links.

- 6.7 [18] How many memory locations would be required to store the dynamic equations of a general three-link manipulator in a table? Quantize each joint's position, velocity, and acceleration into 16 ranges. Make any assumptions needed.
- 6.8 [32] Derive the dynamic equations for the two-link manipulator shown in Fig. 4.6. Link 1 has an inertia tensor given by

$$C_1 I = \begin{bmatrix} I_{xx1} & 0 & 0 \\ 0 & I_{yy1} & 0 \\ 0 & 0 & I_{zz1} \end{bmatrix}.$$

Assume that link 2 has all its mass,  $m_2$ , located at a point at the end-effector. Assume that gravity is directed downward (opposite  $\hat{Z}_1$ ).

- 6.9 [37] Derive the dynamic equations for the three-link manipulator with one prismatic joint shown in Fig. 3.9. Link 1 has an inertia tensor given by

$$C_1 I = \begin{bmatrix} I_{xx1} & 0 & 0 \\ 0 & I_{yy1} & 0 \\ 0 & 0 & I_{zz1} \end{bmatrix}.$$

Link 2 has point mass  $m_2$  located at the origin of its link frame. Link 3 has an inertia tensor given by

$$C_3 I = \begin{bmatrix} I_{xx3} & 0 & 0 \\ 0 & I_{yy3} & 0 \\ 0 & 0 & I_{zz3} \end{bmatrix}.$$

Assume that gravity is directed opposite  $\hat{Z}_1$  and that viscous friction of magnitude  $v_i$  is active at each joint.

- 6.10 [35] Derive the dynamic equations in Cartesian space for the manipulator of Exercise 6.8. Write the equations in frame {2}.
- 6.11 [20] A certain one-link manipulator has

$$C_1 I = \begin{bmatrix} I_{xx1} & 0 & 0 \\ 0 & I_{yy1} & 0 \\ 0 & 0 & I_{zz1} \end{bmatrix}.$$

Assume that this is just the inertia of the link itself. If the motor armature has a moment of inertia  $I_m$  and the gear ratio is 100, what is the total inertia as seen from the motor shaft [1]?

- 6.12 [20] The single-degree-of-freedom “manipulator” in Fig. 6.9 has total mass  $m = 1$ , with the center of mass at

$${}^1P_C = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix},$$

and has inertia tensor

$${}^C I_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}.$$

From rest at  $t = 0$ , the joint angle  $\theta_1$  moves in accordance with the time function

$$\theta_1(t) = bt + ct^2$$

in radians. Give the angular acceleration of the link and the linear acceleration of the center of mass in terms of frame {1} as a function of  $t$ .

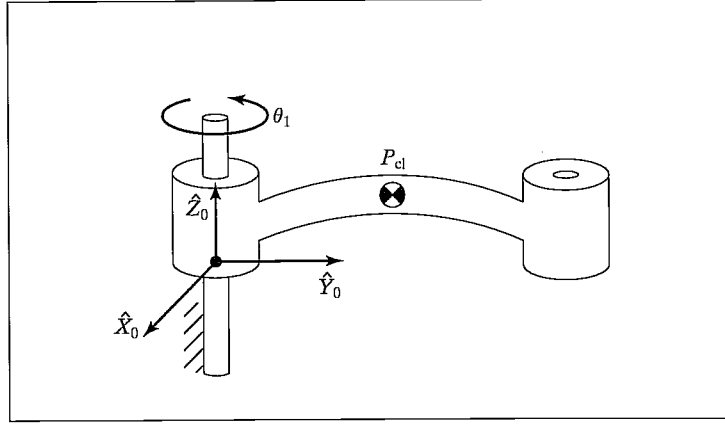


FIGURE 6.9: One-link “manipulator” of Exercise 6.12.

- 6.13** [40] Construct the Cartesian dynamic equations for the two-link nonplanar manipulator shown in Fig. 6.8. Assume that all the mass of the links can be considered as a point mass located at the distal (outermost) end of the link. The mass values are  $m_1$  and  $m_2$ , and the link lengths are  $l_1$  and  $l_2$ . This manipulator is like the first two links of the arm in Exercise 3.3. Also assume that viscous friction is acting at each joint with coefficients  $v_1$  and  $v_2$ . Write the Cartesian dynamics in frame  $\{3\}$ , which is located at the tip of the manipulator and has the same orientation as link frame  $\{2\}$ .
- 6.14** [18] The following equations were derived for a 2-DOF RP manipulator:

$$\begin{aligned}\tau_1 &= m_1(d_1^2 + d_2)\ddot{\theta}_1 + m_2 d_2^2 \ddot{\theta}_1 + 2m_2 d_2 \dot{d}_2 \dot{\theta}_1 \\ &\quad + g \cos(\theta_1)[m_1(d_1 + d_2 \dot{\theta}_1) + m_2(d_2 + \dot{d}_2)] \\ \tau_2 &= m_1 \dot{d}_2 \ddot{\theta}_1 + m_2 \ddot{d}_2 - m_1 d_1 \dot{d}_2 - m_2 d_2 \dot{\theta}^2 + m_2(d_2 + 1)g \sin(\theta_1).\end{aligned}$$

Some of the terms are obviously incorrect. Indicate the incorrect terms.

- 6.15** [28] Derive the dynamic equations for the RP manipulator of Example 6.5, using the Newton–Euler procedure instead of the Lagrangian technique.
- 6.16** [25] Derive the equations of motion for the PR manipulator shown in Fig. 6.10. Neglect friction, but include gravity. (Here,  $\hat{X}_0$  is upward.) The inertia tensors of the links are diagonal, with moments  $I_{xx1}$ ,  $I_{yy1}$ ,  $I_{zz1}$  and  $I_{xx2}$ ,  $I_{yy2}$ ,  $I_{zz2}$ . The centers of mass for the links are given by

$$\begin{aligned}{}^1P_{C_1} &= \begin{bmatrix} 0 \\ 0 \\ -l_1 \end{bmatrix}, \\ {}^2P_{C_2} &= \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.\end{aligned}$$

- 6.17** [40] The velocity-related terms appearing in the manipulator dynamic equation can be written as a matrix-vector product—that is,

$$V(\Theta, \dot{\Theta}) = V_m(\Theta, \dot{\Theta})\dot{\Theta},$$

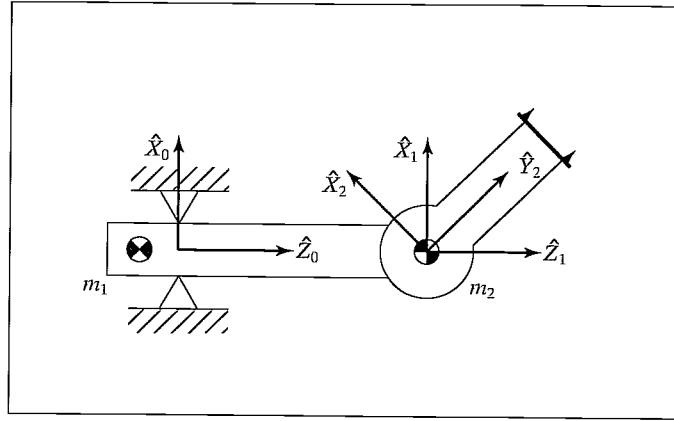


FIGURE 6.10: PR manipulator of Exercise 6.16.

where the  $m$  subscript stands for “matrix form.” Show that an interesting relationship exists between the time derivative of the manipulator mass matrix and  $V_m(\cdot)$ , namely,

$$\dot{M}(\Theta) = 2V_m(\Theta, \dot{\Theta}) - S,$$

where  $S$  is some skew-symmetric matrix.

- 6.18** [15] Give two properties that any reasonable friction model (i.e., the term  $F(\Theta, \dot{\Theta})$  in (6.114)) would possess.
- 6.19** [28] Do Exercise 6.5, using Lagrange’s equations.
- 6.20** [28] Derive the dynamic equations of the 2-DOF manipulator of Section 6.7, using a Lagrangian formulation.

### PROGRAMMING EXERCISE (PART 6)

- 1.** Derive the dynamic equations of motion for the three-link manipulator (from Example 3.3). That is, expand Section 6.7 for the three-link case. The following numerical values describe the manipulator:

$$l_1 = l_2 = 0.5\text{m},$$

$$m_1 = 4.6\text{Kg},$$

$$m_2 = 2.3\text{Kg},$$

$$m_3 = 1.0\text{Kg},$$

$$g = 9.8\text{m/s}^2.$$

For the first two links, we assume that the mass is all concentrated at the distal end of the link. For link 3, we assume that the center of mass is located at the origin of frame {3}—that is, at the proximal end of the link. The inertia tensor for link 3 is

$$c_s I = \begin{bmatrix} 0.05 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.1 \end{bmatrix} \text{Kg-m}^2.$$

The vectors that locate each center of mass relative to the respective link frame are

$${}^1P_{C_1} = l_1 \hat{X}_1,$$

$${}^2P_{C_2} = l_2 \hat{X}_2,$$

$${}^3P_{C_3} = 0.$$

2. Write a simulator for the three-link manipulator. A simple Euler-integration routine is sufficient for performing the numerical integration (as in Section 6.12). To keep your code modular, it might be helpful to define the routine

```
Procedure UPDATE(VAR tau: vec3; VAR period: real; VAR
theta, thetadot: vec3);
```

where “tau” is the torque command to the manipulator (always zero for this assignment), “period” is the length of time you wish to advance time (in seconds), and “theta” and “thetadot” are the *state* of the manipulator. Theta and thetadot are updated by “period” seconds each time you call UPDATE. Note that “period” would typically be longer than the integration step size,  $\Delta t$ , used in the numerical integration. For example, although the step size for numerical integration might be 0.001 second, you might wish to print out the manipulator position and velocity only each 0.1 seconds.

To test your simulation, set the joint-torque commands to zero (for all time) and perform these tests:

- (a) Set the initial position of the manipulator to

$$[\theta_1 \ \theta_2 \ \theta_3] = [-90 \ 0 \ 0].$$

Simulate for a few seconds. Is the motion of the manipulator what you would expect?

- (b) Set the initial position of the manipulator to

$$[\theta_1 \ \theta_2 \ \theta_3] = [30 \ 30 \ 10].$$

Simulate for a few seconds. Is the motion of the manipulator what you would expect?

- (c) Introduce some viscous friction at each joint of the simulated manipulator—that is, add a term to the dynamics of each joint in the form  $\tau_f = v\dot{\theta}$ , where  $v = 5.0$  newton-meter-seconds for each joint. Repeat test (b) above. Is the motion what you would expect?

## MATLAB EXERCISE 6A

This exercise focuses on the inverse-dynamics analysis (in a resolved-rate control framework—see MATLAB Exercise 5) for the planar 2-DOF 2R robot. This robot is the first two R-joints and first two moving links of the planar 3-DOF 3R robot. (See Figures 3.6 and 3.7; the DH parameters are given in the first two rows of Figure 3.8.)

For the planar 2R robot, calculate the required joint torques (i.e., solve the inverse-dynamics problem) to provide the commanded motion at every time step in a resolved-rate control scheme. You can use either numerical Newton–Euler recursion or the analytical equations from the results of Exercise 6.2, or both.

Given:  $L_1 = 1.0$  m,  $L_2 = 0.5$  m; Both links are solid steel with mass density  $\rho = 7806$  kg/m<sup>3</sup>; both have the width and thickness dimensions  $w = t = 5$  cm. The revolute joints are assumed to be perfect, connecting the links at their very edges (not physically possible).

The initial angles are  $\Theta = \begin{Bmatrix} \theta_1 \\ \theta_2 \end{Bmatrix} = \begin{Bmatrix} 10^\circ \\ 90^\circ \end{Bmatrix}$ .

The (constant) commanded Cartesian velocity is  ${}^0\dot{X} = {}^0\begin{Bmatrix} \dot{x} \\ \dot{y} \end{Bmatrix} = {}^0\begin{Bmatrix} 0 \\ 0.5 \end{Bmatrix}$  (m/s).

Simulate motion for 1 sec, with a control time step of 0.01 sec.

Present five plots (each set on a separate graph, please):

1. the two joint angles (degrees)  $\Theta = \{\theta_1 \ \theta_2\}^T$  vs. time;
2. the two joint rates (rad/s)  $\dot{\Theta} = \{\dot{\theta}_1 \ \dot{\theta}_2\}^T$  vs. time;
3. the two joint accelerations (rad/s<sup>2</sup>)  $\ddot{\Theta} = \{\ddot{\theta}_1 \ \ddot{\theta}_2\}^T$  vs. time;
4. the three Cartesian components of  ${}^0_H T, X = \{x \ y \ \phi\}^T$  (rad is fine for  $\phi$  so it will fit) vs. time;
5. the two inverse dynamics joint torques (Nm)  $T = \{\tau_1 \ \tau_2\}^T$  vs. time.

Carefully label (by hand is fine!) each component on each plot. Also, label the axis names and units.

Perform this simulation twice. The first time, ignore gravity (the motion plane is normal to the effect of gravity); the second time, consider gravity  $g$  in the negative  $Y$  direction.

## MATLAB EXERCISE 6B

This exercise focuses on the inverse-dynamics solution for the planar 3-DOF, 3R robot (of Figures 3.6 and 3.7; the DH parameters are given in Figure 3.8) for a motion snapshot in time only. The following fixed-length parameters are given:  $L_1 = 4$ ,  $L_2 = 3$ , and  $L_3 = 2$  (m). For dynamics, we must also be given mass and moment-of-inertia information:  $m_1 = 20$ ,  $m_2 = 15$ ,  $m_3 = 10$  (kg),  ${}^C I_{ZZ1} = 0.5$ ,  ${}^C I_{ZZ2} = 0.2$ , and  ${}^C I_{ZZ3} = 0.1$  (kgm<sup>2</sup>). Assume that the CG of each link is in its geometric center. Also, assume that gravity acts in the  $-Y$  direction in the plane of motion. For this exercise, ignore actuator dynamics and the joint gearing.

- a) Write a MATLAB program to implement the recursive Newton–Euler inverse-dynamics solution (i.e., given the commanded motion, calculate the required driving joint torques) for the following motion snapshot in time:

$$\Theta = \begin{Bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{Bmatrix} = \begin{Bmatrix} 10^\circ \\ 20^\circ \\ 30^\circ \end{Bmatrix} \quad \dot{\Theta} = \begin{Bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{Bmatrix} = \begin{Bmatrix} 1 \\ 2 \\ 3 \end{Bmatrix} \text{ (rad/s)} \quad \ddot{\Theta} = \begin{Bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{Bmatrix} = \begin{Bmatrix} 0.5 \\ 1 \\ 1.5 \end{Bmatrix} \text{ (rad/s}^2\text{)}$$

- b) Check your results in (a) by means of the Corke MATLAB Robotics Toolbox. Try functions *rne()* and *gravload()*.

## MATLAB EXERCISE 6C

This exercise focuses on the forward-dynamics solution for the planar 3-DOF, 3R robot (parameters from MATLAB Exercise 6B) for motion over time. In this case, ignore gravity (i.e., assume that gravity acts in a direction normal to the plane of motion). Use the Corke MATLAB Robotics Toolbox to solve the forward-dynamics problem (i.e.,

given the commanded driving joint torques, calculate the resulting robot motion) for the following constant joint torques and the given initial joint angles and initial joint rates:

$$\mathbf{T} = \begin{Bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{Bmatrix} = \begin{Bmatrix} 20 \\ 5 \\ 1 \end{Bmatrix} \text{ (Nm, constant)} \quad \Theta_0 = \begin{Bmatrix} \theta_{10} \\ \theta_{20} \\ \theta_{30} \end{Bmatrix} = \begin{Bmatrix} -60^\circ \\ 90^\circ \\ 30^\circ \end{Bmatrix}$$

$$\dot{\Theta}_0 = \begin{Bmatrix} \dot{\theta}_{10} \\ \dot{\theta}_{20} \\ \dot{\theta}_{30} \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix} \text{ (rad/s)}$$

Perform this simulation for 4 seconds. Try function *fdyn()*.

Present two plots for the resulting robot motion (each set on a separate graph, please):

1. the three joint angles (degrees)  $\Theta = \{\theta_1 \ \theta_2 \ \theta_3\}^T$  vs. time;
2. the three joint rates (rad/s)  $\dot{\Theta} = \{\dot{\theta}_1 \ \dot{\theta}_2 \ \dot{\theta}_3\}^T$  vs. time.

Carefully label (by hand is fine!) each component on each plot. Also, label the axis names and units.