the computation. There has been some application of computer-assisted automatic factorization of such equations [5].

The major expense in calculating kinematics is often the calculation of the transcendental functions (sine and cosine). When these functions are available as part of a standard library, they are often computed from a series expansion at the cost of many *multiply times*. At the expense of some required memory, many manipulation systems employ table-lookup implementations of the transcendental functions. Depending on the scheme, this reduces the amount of time required to calculate a sine or cosine to two or three *multiply times* or less [6].

The computation of the kinematics as in (3.14) is redundant, in that nine quantities are calculated to represent orientation. One means that usually reduces computation is to calculate only two columns of the rotation matrix, then to compute a cross product (requiring only six multiplications and three additions) to compute the third column. Obviously, one chooses the two least complicated columns to compute.

## BIBLIOGRAPHY

**[1]** J. Denavit and R.S. Hartenberg, "A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices," *Journal of Applied Mechanics*, pp. 215–221, June 1955.

**[2]** J. Lenarčič, "Kinematics," in *The International Encyclopedia of Robotics*, R. Dorf and S. Nof, Editors, John C. Wiley and Sons, New York, 1988.

**[3]** J. Colson and N.D. Perreira, "Kinematic Arrangements Used in Industrial Robots," *13th Industrial Robots Conference Proceedings*, April 1983.

**[4]** T. Turner, J. Craig, and W. Gruver, "A Microprocessor Architecture for Advanced Robot Control," 14th ISIR, Stockholm, Sweden, October 1984.

**[5]** 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.

**[6]** C. Ruoff, "Fast Trigonometric Functions for Robot Control," *Robotics Age*, November 1981.

## EXERCISES

**3.1** [15] Compute the kinematics of the planar arm from Example 3.3.

**3.2** [37] Imagine an arm like the PUMA 560, except that joint 3 is replaced with a prismatic joint. Assume the prismatic joint slides along the direction of $\hat{X}_1$ in Fig. 3.18; however, there is still an offset equivalent to $d_3$ to be accounted for. Make any additional assumptions needed. Derive the kinematic equations.

**3.3** [25] The arm with three degrees of freedom shown in Fig. 3.29 is like the one in Example 3.3, except that joint 1's axis is not parallel to the other two. Instead, there is a twist of 90 degrees in magnitude between axes 1 and 2. Derive link parameters and the kinematic equations for ${}^B_W T$. Note that no $l_3$ need be defined.

**3.4** [22] The arm with three degrees of freedom shown in Fig. 3.30 has joints 1 and 2 perpendicular, and joints 2 and 3 parallel. As pictured, all joints are at their zero location. Note that the positive sense of the joint angle is indicated. Assign link frames {0} through {3} for this arm—that is, sketch the arm, showing the attachment of the frames. Then derive the transformation matrices ${}^0_1 T$, ${}^1_2 T$, and ${}^2_3 T$.

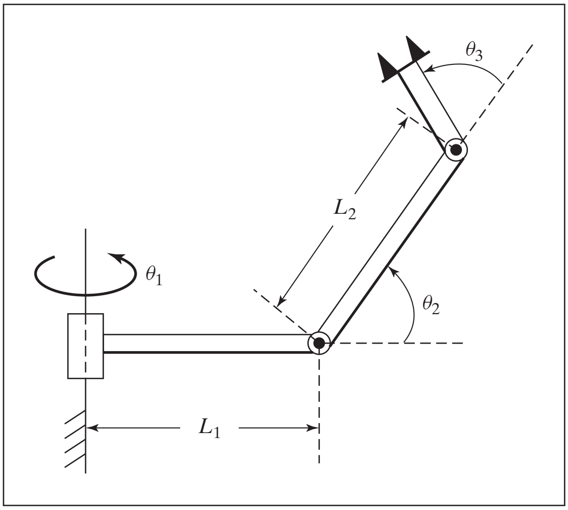


FIGURE 3.29: The $3R$ nonplanar arm (Exercise 3.3).

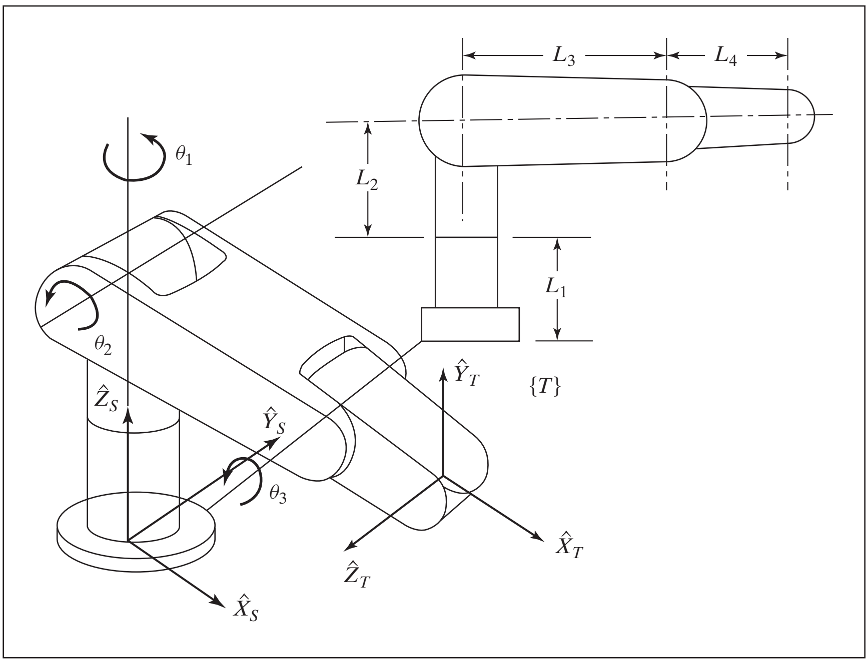


FIGURE 3.30: Two views of a $3R$ manipulator (Exercise 3.4).

**3.5** [26] Write a subroutine to compute the kinematics of a PUMA 560. Code for speed, trying to minimize the number of multiplications as much as possible. Use the procedure heading (or equivalent in C)

```
Procedure KIN(VAR theta: vec6; VAR wrelb: frame);
```

Count a sine or cosine evaluation as costing 5 *multiply times*. Count additions as costing 0.333 *multiply times* and assignment statements as 0.2 *multiply times*. Count a square-root computation as costing 4 *multiply times*. How many *multiply times* do you need?

**3.6** [20] Write a subroutine to compute the kinematics of the cylindrical arm in Example 3.4. Use the procedure heading (or equivalent in C)

```
Procedure KIN(VAR jointvar: vec3; VAR wrelb: frames);
```

Count a sine or cosine evaluation as costing 5 *multiply times*. Count additions as costing 0.333 *multiply times* and assignment statements as 0.2 *multiply times*. Count a square-root computation as costing 4 *multiply times*. How many *multiply times* do you need?

**3.7** [22] Write a subroutine to compute the kinematics of the arm in Exercise 3.3. Use the procedure heading (or equivalent in C)

```
Procedure KIN(VAR theta: vec3; VAR wrelb: frame);
```

Count a sine or cosine evaluation as costing 5 *multiply times*. Count additions as costing 0.333 *multiply times* and assignment statements as 0.2 *multiply times*. Count a square-root computation as costing 4 *multiply times*. How many *multiply times* do you need?

**3.8** [13] In Fig. 3.31, the location of the tool, ${}^{W}_{T}T$, is not accurately known. Using force control, the robot feels around with the tool tip until it inserts it into the

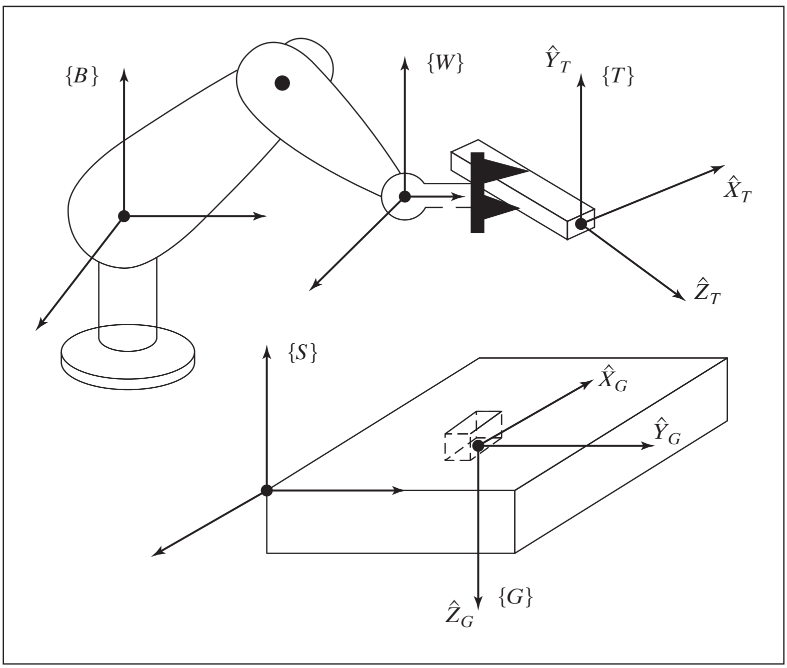


FIGURE 3.31: Determination of the tool frame (Exercise 3.8).

socket (or Goal) at location ${}^S_G T$. Once in this "calibration" configuration (in which $\{G\}$ and $\{T\}$ are coincident), the position of the robot, ${}^B_W T$, is figured out by reading the joint angle sensors and computing the kinematics. Assuming ${}^B_S T$ and ${}^S_G T$ are known, give the transform equation to compute the unknown tool frame, ${}^W_T T$.

**3.9** [11] For the two-link manipulator shown in Fig. 3.32(a), the link-transformation matrices, ${}^0_1 T$ and ${}^1_2 T$, were constructed. Their product is

$$
{}^0_2 T = \begin{bmatrix} c\theta_1 c\theta_2 & -c\theta_1 s\theta_2 & s\theta_1 & l_1 c\theta_1 \\ s\theta_1 c\theta_2 & -s\theta_1 s\theta_2 & -c\theta_1 & l_1 s\theta_1 \\ s\theta_2 & c\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.
$$

The link-frame assignments used are indicated in Fig. 3.32(b). Note that frame $\{0\}$ is coincident with frame $\{1\}$ when $\theta_1 = 0$. The length of the second link is $l_2$. Find an expression for the vector ${}^0 P_{tip}$, which locates the tip of the arm relative to the $\{0\}$ frame.

**3.10** [39] Derive kinematic equations for the Yasukawa Motoman robot (see Section 3.7) that compute the position and orientation of the wrist frame directly from actuator values, rather than by first computing the joint angles. A solution is possible that requires only 33 multiplications, two square roots, and six sine or cosine evaluations.

**3.11** [17] Figure 3.33 shows the schematic of a wrist which has three intersecting axes that are not orthogonal. Assign link frames to this wrist (as if it were a 3-DOF manipulator), and give the link parameters.

**3.12** [08] Can an arbitrary rigid-body transformation always be expressed with four parameters $(a, \alpha, d, \theta)$ in the form of equation (3.6)?

**3.13** [15] Show the attachment of link frames for the 5-DOF manipulator shown schematically in Fig. 3.34.

**3.14** [20] As was stated, the relative position of any two lines in space can be given with two parameters, $a$ and $\alpha$, where $a$ is the length of the common perpendicular joining the two, and $\alpha$ is the angle made by the two axes when projected onto a plane normal to the common perpendicular. Given a line defined as passing

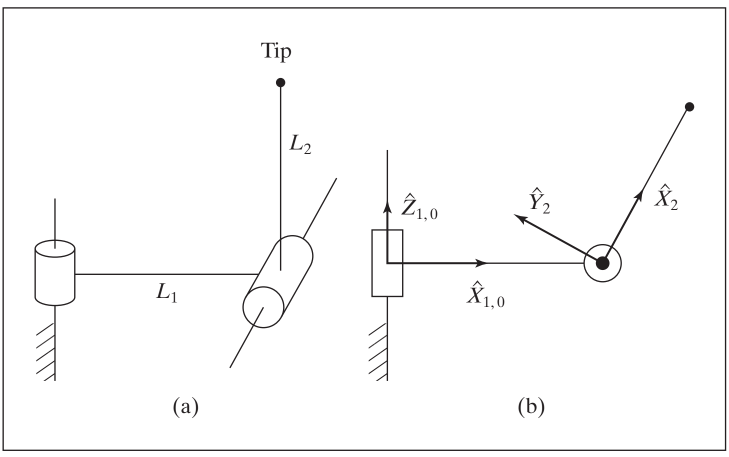


FIGURE 3.32: Two-link arm with frame assignments (Exercise 3.9).

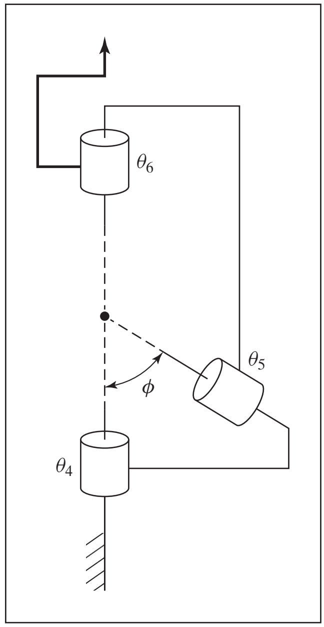


FIGURE 3.33: $3R$ nonorthogonal-axis robot (Exercise 3.11).

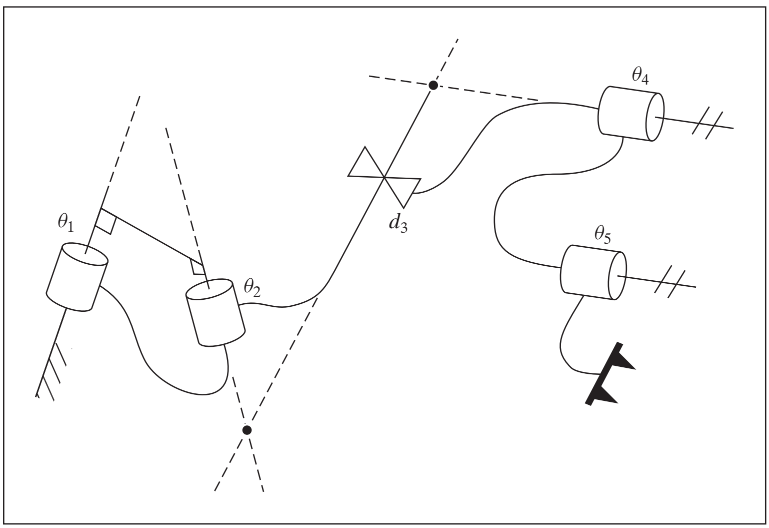


FIGURE 3.34: Schematic of a $2RP2R$ manipulator (Exercise 3.13).

through point $p$ with unit-vector direction $\hat{m}$, and a second passing through point $q$ with unit-vector direction $\hat{n}$, write expressions for $a$ and $\alpha$.

**3.15** [15] Show the attachment of link frames for the 3-DOF manipulator shown schematically in Fig. 3.35.

**3.16** [15] Assign link frames to the $RPR$ planar robot shown in Fig. 3.36, and give the linkage parameters.

**3.17** [15] Show the attachment of link frames on the three-link robot shown in Fig. 3.37.

**3.18** [15] Show the attachment of link frames on the three-link robot shown in Fig. 3.38.

**3.19** [15] Show the attachment of link frames on the three-link robot shown in Fig. 3.39.

**3.20** [15] Show the attachment of link frames on the three-link robot shown in Fig. 3.40.

**3.21** [15] Show the attachment of link frames on the three-link robot shown in Fig. 3.41.

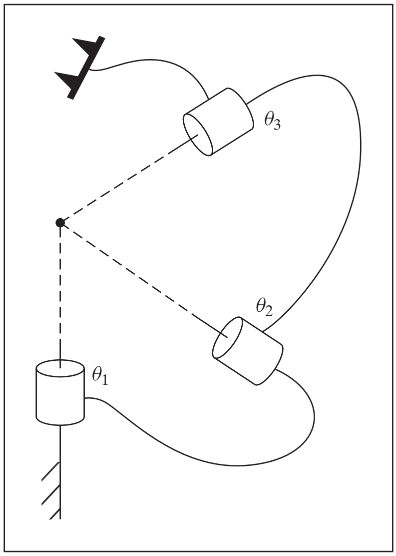


FIGURE 3.35: Schematic of a $3R$ manipulator (Exercise 3.15).

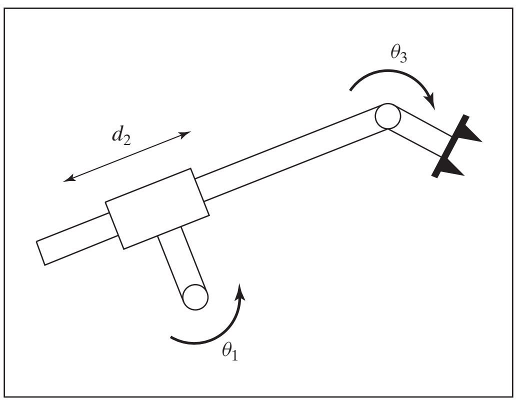


FIGURE 3.36: $RPR$ planar robot (Exercise 3.16).

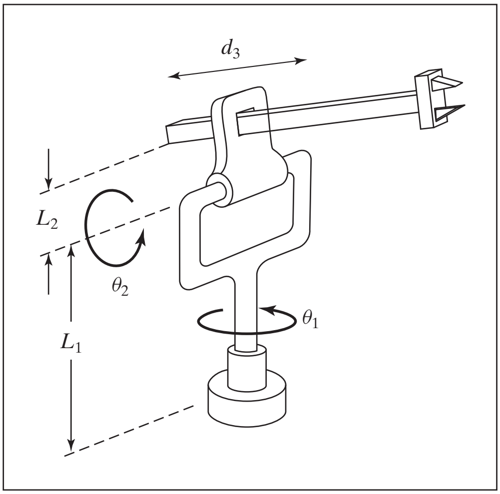


FIGURE 3.37: Three-link *RRP* manipulator (Exercise 3.17).

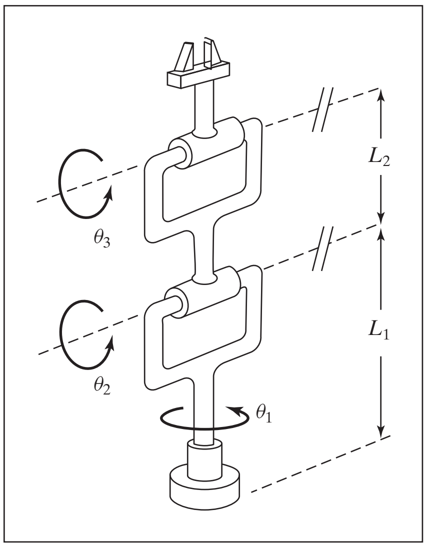


FIGURE 3.38: Three-link *RRR* manipulator (Exercise 3.18).

**3.22** [18] Show the attachment of link frames on the $P3R$ robot shown in Fig. 3.42. Given your frame assignments, what are the signs of $d_2$, $d_3$, and $a_2$?

**3.23** [15] Show the attachment of link frames on the gantry crane of Fig. 3.43, which has four degrees of freedom (one of them is redundant).

**3.24** [18] For the PUMA 560 kinematics obtained in Section 3.7, compare the computation time required to individually calculate the $r_{ij}$ terms of (3.14) versus obtaining column one via the cross product, $[r_{12} \quad r_{22} \quad r_{32}]^T \times [r_{13} \quad r_{23} \quad r_{33}]^T$. Count a sine or cosine evaluation as costing 5 *multiply times*, additions as costing 0.333 *multiply times*, and assignment statements as 0.2 *multiply times*.

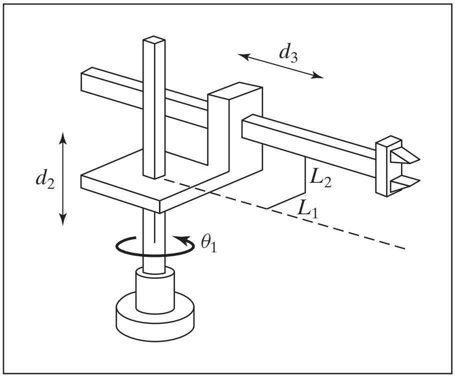


FIGURE 3.39: Three-link *RPP* manipulator (Exercise 3.19).

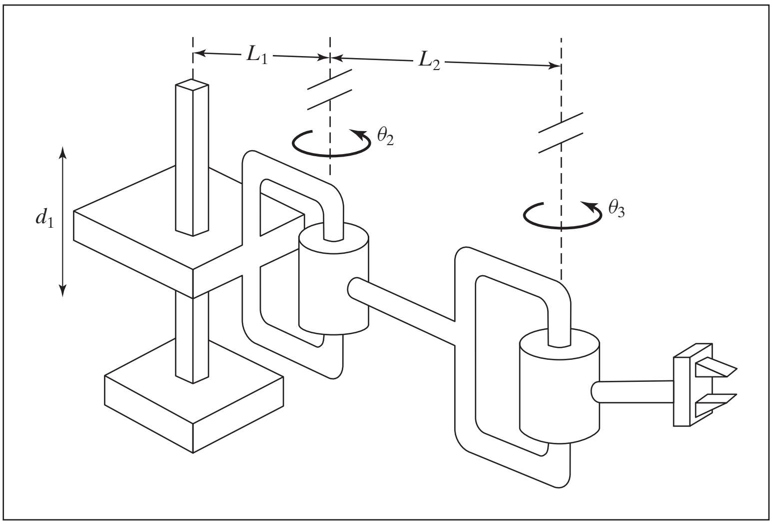


FIGURE 3.40: Three-link *PRR* manipulator (Exercise 3.20).

**3.25** [32] A certain human leg has the following dimensions (in cm): femur length = 500, shank length = 400, ankle-to-heel distance = 50, ankle-to-toe distance = 150. For the leg shown in Fig. 3.44, the three joint angles, $\Theta = (\theta_{\text{hip}}, \theta_{\text{knee}}, \theta_{\text{ankle}})$, are zero when the leg is vertical.

- **a)** Show the attachment of link frames on the leg.
- **b)** Give values for the link parameters, assuming the frame origins are coplanar.
- **c)** Compute the stride length, that is the distance along the ground from toe contact-point to heel contact-point, using the joint vectors $\Theta_{\text{toe-contant}} = (-4.15°, -38.3°, -2.57°)$ and $\Theta_{\text{heel-contact}} = (9.64°, -19.9°, 31.8°)$ and assuming that the heel contact-point, the ankle joint, and the toe contact-point are collinear.

**3.26** [35] Some 6R painting robots route tubing through a hollow upper arm to provide a slim profile for easier access to complex part shapes. One such manipulator, the

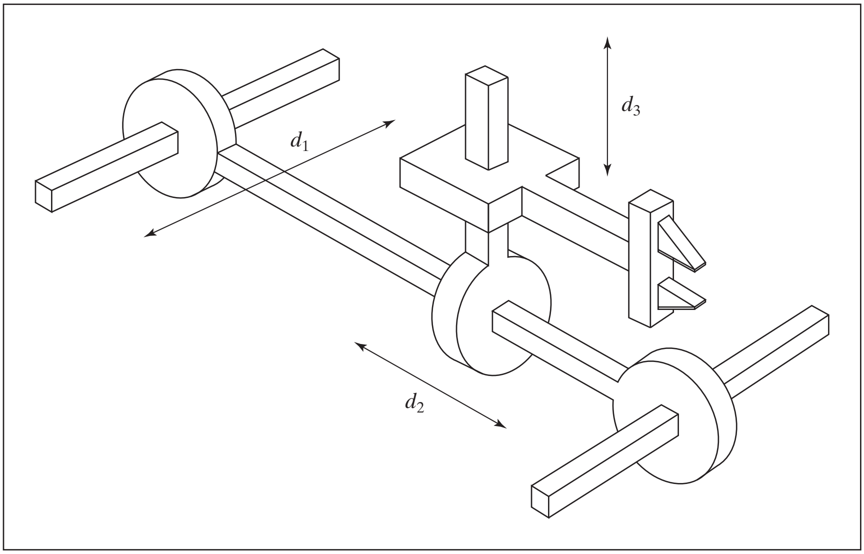


FIGURE 3.41: Three-link *PPP* manipulator (Exercise 3.21).

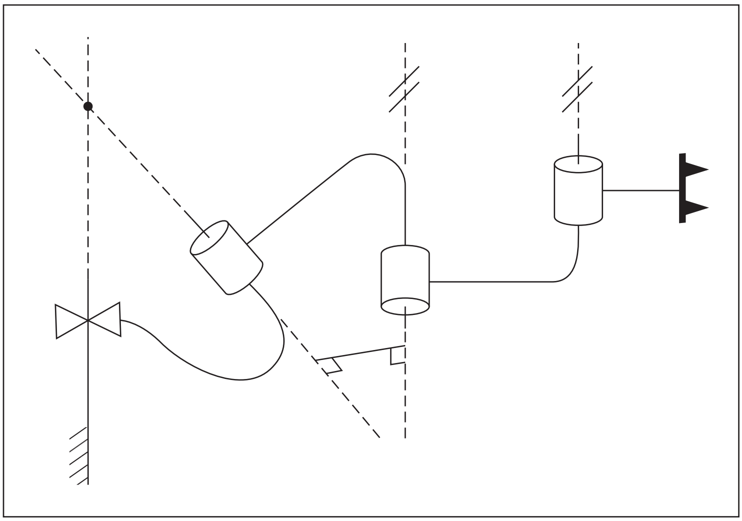

FIGURE 3.42: Schematic of a *P3R* manipulator (Exercise 3.22).

Motoman EPX2800, is depicted in Fig. 3.45. In contrast to the PUMA 560 (see Figs. 3.18 and 3.19) its axes of joints 4, 5, and 6 do not all intersect at a common point. The angle between joint axes 4 and 5 is 45 degrees, as is the angle between joint axes 5 and 6. Show the attachment of link frames on the EPX2800 and give values for the link parameters, including $\theta_{1\ldots6}$ for the configuration shown, in which all frame origins are coplanar. Let the origin of frame {6} be on the flange at the manipulator's end.

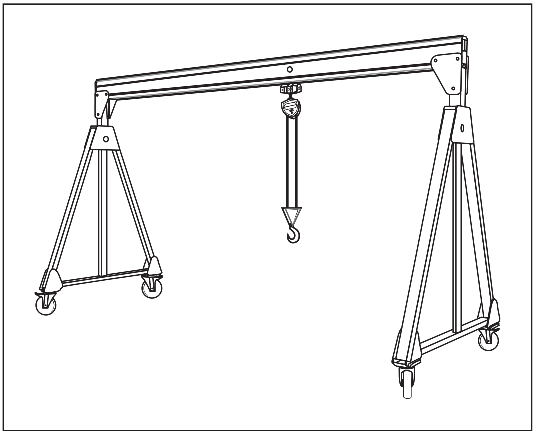

FIGURE 3.43: Gantry crane (Exercise 3.23).

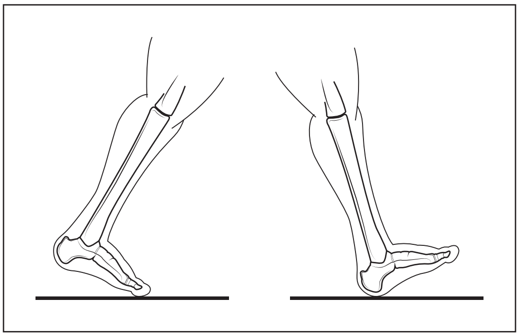

FIGURE 3.44: Toe contact and heel contact for a striding leg (Exercise 3.25).

**3.27** [40] A PUMA 560 is grasping a pin-like tool as in Fig. 3.28, but the location of the tool, ${}^W_T T$, is not known. One can determine the tool offset (that is, ${}^6p_x$, ${}^6p_y$, and ${}^6p_z$) without having access to a known fixed point, such as $\{S\}$, if there is at least a fixed point of unknown coordinates. This is achieved by visually making the tool point coincident with the fixed point twice, using different wrist orientations each time. Write an equation for computing the tool offset, ${}^6P$, given two such configurations.

**3.28** [15] For the manipulator shown in Fig. 3.37, give the linkage parameters.

**3.29** [15] For the manipulator shown in Fig. 3.38, give the linkage parameters.

**3.30** [15] For the manipulator shown in Fig. 3.39, give the linkage parameters.

**3.31** [15] For the manipulator shown in Fig. 3.40, give the linkage parameters.

## PROGRAMMING EXERCISE (PART 3)

1. Write a subroutine to compute the kinematics of the planar $3R$ robot in Example 3.3—that is, a routine with the joint angles' values as input, and a frame