CHAPTER 3

Manipulator kinematics

3.1 INTRODUCTION

Kinematics is the science of motion that treats the subject without regard to the forces that cause it. Within the science of kinematics, one studies the position, the velocity, the acceleration, and all higher order derivatives of the position variables (with respect to time or any other variable(s)). Hence, the study of the kinematics of manipulators refers to all the geometrical and time-based properties of the motion. The relationships between these motions and the forces and torques that cause them constitute the problem of dynamics, which is the subject of Chapter 6.

In this chapter, we consider position and orientation of the manipulator linkages in static situations. In Chapters 5 and 6, we will consider the kinematics when velocities and accelerations are involved.

In order to deal with the complex geometry of a manipulator, we will affix frames to the various parts of the mechanism and then describe the relationships between these frames. The study of manipulator kinematics involves, among other things, how the locations of these frames change as the mechanism articulates. The central topic of this chapter is a method to compute the position and orientation of the manipulator's end-effector relative to the base of the manipulator as a function of the joint variables.

3.2 LINK DESCRIPTION

A manipulator may be thought of as a set of bodies connected in a chain by joints. These bodies are called links. Joints form a connection between a neighboring pair of links. The term **lower pair** is used to describe the connection between a pair of

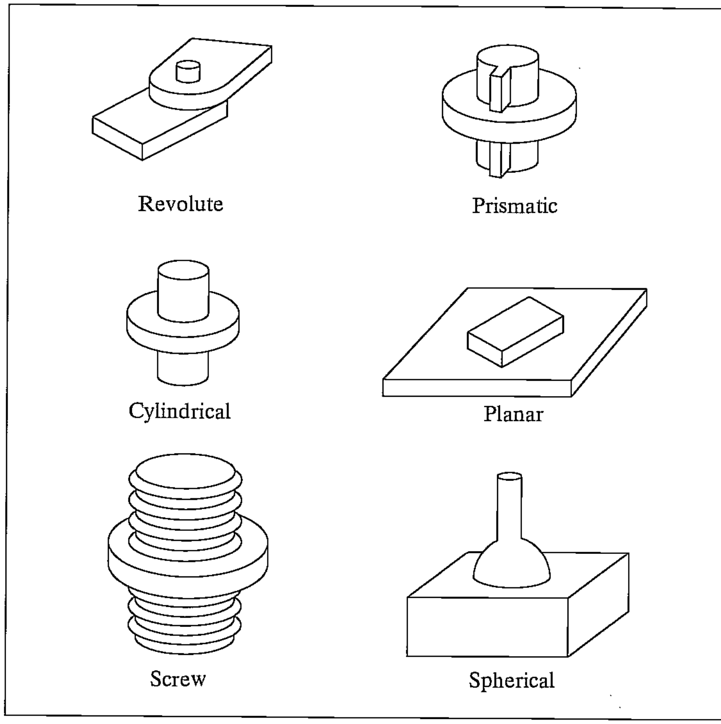


FIGURE 3.1: The six possible lower-pair joints.

bodies when the relative motion is characterized by two surfaces sliding over one another. Figure 3.1 shows the six possible lower pair joints.

Mechanical-design considerations favor manipulators' generally being constructed from joints that exhibit just one degree of freedom. Most manipulators have **revolute joints** or have sliding joints called **prismatic joints**. In the rare case that a mechanism is built with a joint having n degrees of freedom, it can be modeled as n joints of one degree of freedom connected with $n - 1$ links of zero length. Therefore, without loss of generality, we will consider only manipulators that have joints with a single degree of freedom.

The links are numbered starting from the immobile base of the arm, which might be called link 0. The first moving body is link 1, and so on, out to the free end of the arm, which is link n. In order to position an end-effector generally in 3-space, a minimum of six joints is required.[1] Typical manipulators have five or six joints. Some robots are not actually as simple as a single kinematic chain—these have parallelogram linkages or other closed kinematic structures. We will consider one such manipulator later in this chapter.

A single link of a typical robot has many attributes that a mechanical designer had to consider during its design: the type of material used, the strength and stiffness

[1]This makes good intuitive sense, because the description of an object in space requires six parameters—three for position and three for orientation.

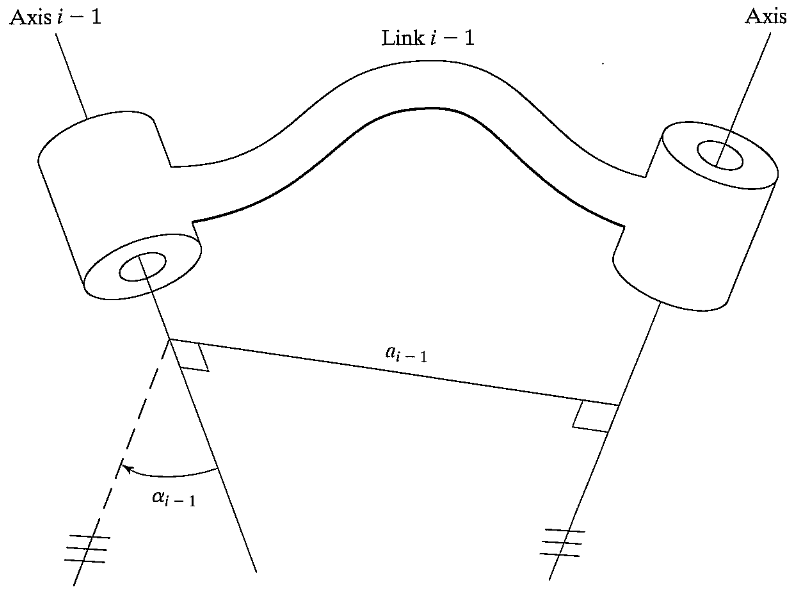


FIGURE 3.2: The kinematic function of a link is to maintain a fixed relationship between the two joint axes it supports. This relationship can be described with two parameters: the link length, a, and the link twist, α.

of the link, the location and type of the joint bearings, the external shape, the weight and inertia, and more. However, for the purposes of obtaining the kinematic equations of the mechanism, *a link is considered only as a rigid body that defines the relationship between two neighboring joint axes of a manipulator*. Joint axes are defined by lines in space. Joint axis i is defined by a line in space, or a vector direction, about which link i rotates relative to link $i-1$. It turns out that, for kinematic purposes, a link can be specified with two numbers, which define the relative location of the two axes in space.

For any two axes in 3-space, there exists a well-defined measure of distance between them. This distance is measured along a line that is mutually perpendicular to both axes. This mutual perpendicular always exists; it is unique except when both axes are parallel, in which case there are many mutual perpendiculars of equal length. Figure 3.2 shows link $i-1$ and the mutually perpendicular line along which the **link length**, a_{i-1}, is measured. Another way to visualize the link parameter a_{i-1} is to imagine an expanding cylinder whose axis is the joint $i-1$ axis—when it just touches joint axis i, the radius of the cylinder is equal to a_{i-1}.

The second parameter needed to define the relative location of the two axes is called the **link twist**. If we imagine a plane whose normal is the mutually perpendicular line just constructed, we can project the axes $i-1$ and i onto this plane and measure the angle between them. This angle is measured from axis $i-1$ to axis i in the right-hand sense about a_{i-1}.[2] We will use this definition of the twist

[2] In this case, a_{i-1} is given the direction pointing from axis $i-1$ to axis i.

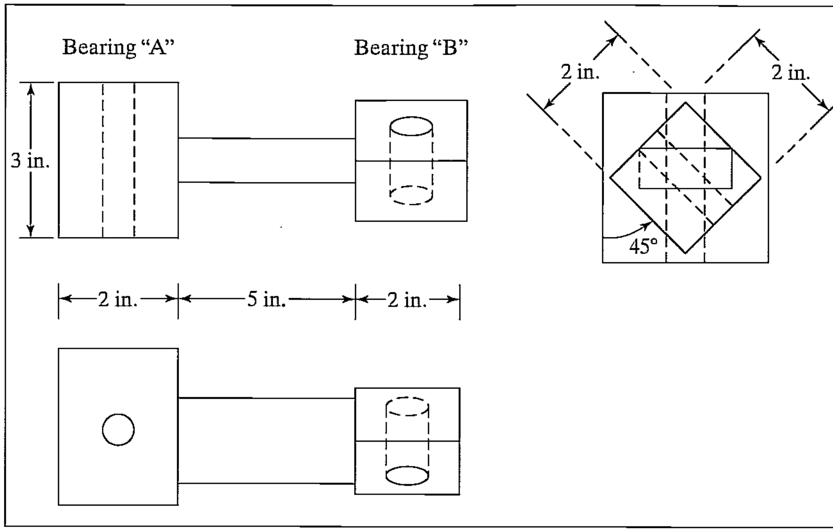


FIGURE 3.3: A simple link that supports two revolute axes.

of link $i-1$, α_{i-1}. In Fig. 3.2, α_{i-1} is indicated as the angle between axis $i-1$ and axis i. (The lines with the triple hash marks are parallel.) In the case of intersecting axes, twist is measured in the plane containing both axes, but the sense of α_{i-1} is lost. In this special case, one is free to assign the sign of α_{i-1} arbitrarily.

You should convince yourself that these two parameters, length and twist, as defined above, can be used to define the relationship between any two lines (in this case axes) in space.

EXAMPLE 3.1

Figure 3.3 shows the mechanical drawings of a robot link. If this link is used in a robot, with bearing "A" used for the lower-numbered joint, give the length and twist of this link. Assume that holes are centered in each bearing.

By inspection, the common perpendicular lies right down the middle of the metal bar connecting the bearings, so the link length is 7 inches. The end view actually shows a projection of the bearings onto the plane whose normal is the mutual perpendicular. Link twist is measured in the right-hand sense about the common perpendicular from axis $i-1$ to axis i, so, in this example, it is clearly $+45$ degrees.

3.3 LINK-CONNECTION DESCRIPTION

The problem of connecting the links of a robot together is again one filled with many questions for the mechanical designer to resolve. These include the strength of the joint, its lubrication, and the bearing and gearing mounting. However, for the investigation of kinematics, we need only worry about two quantities, which will completely specify the way in which links are connected together.

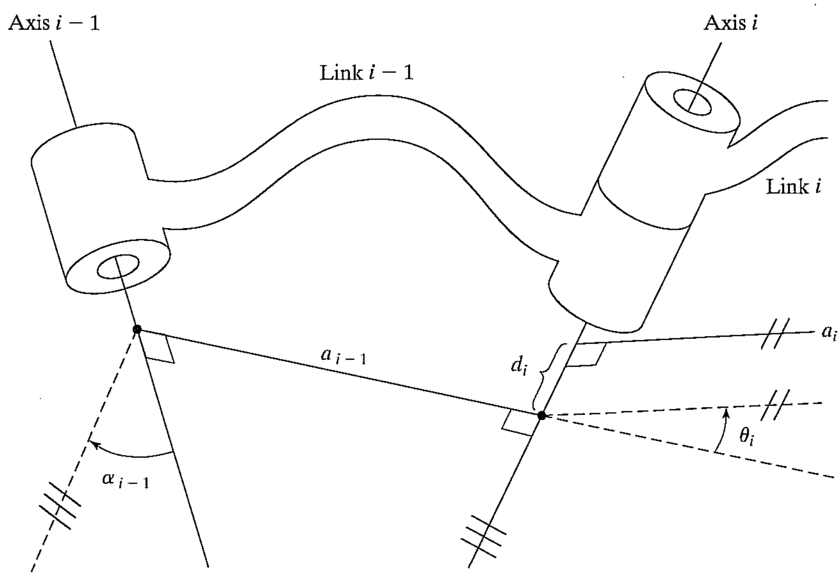


FIGURE 3.4: The link offset, d, and the joint angle, θ, are two parameters that may be used to describe the nature of the connection between neighboring links.

Intermediate links in the chain

Neighboring links have a common joint axis between them. One parameter of interconnection has to do with the distance along this common axis from one link to the next. This parameter is called the **link offset**. The offset at joint axis i is called d_i. The second parameter describes the amount of rotation about this common axis between one link and its neighbor. This is called the **joint angle**, θ_i.

Figure 3.4 shows the interconnection of link $i-1$ and link i. Recall that a_{i-1} is the mutual perpendicular between the two axes of link $i-1$. Likewise, a_i is the mutual perpendicular defined for link i. The first parameter of interconnection is the link offset, d_i, which is the signed distance measured along the axis of joint i from the point where a_{i-1} intersects the axis to the point where a_i intersects the axis. The offset d_i is indicated in Fig. 3.4. The link offset d_i is variable if joint i is prismatic. The second parameter of interconnection is the angle made between an extension of a_{i-1} and a_i measured about the axis of joint i. This is indicated in Fig. 3.4, where the lines with the double hash marks are parallel. This parameter is named θ_i and is variable for a revolute joint.

First and last links in the chain

Link length, a_i, and link twist, α_i, depend on joint axes i and $i+1$. Hence, a_1 through a_{n-1} and α_1 through α_{n-1} are defined as was discussed in this section. At the ends of the chain, it will be our convention to assign zero to these quantities. That is, $a_0 = a_n = 0.0$ and $\alpha_0 = \alpha_n = 0.0$.[3] Link offset, d_i, and joint angle, θ_i, are well defined

[3] In fact, a_n and α_n do not need to be defined at all.

for joints 2 through $n - 1$ according to the conventions discussed in this section. If joint 1 is revolute, the zero position for θ_1 may be chosen arbitrarily; $d_1 = 0.0$ will be our convention. Similarly, if joint 1 is prismatic, the zero position of d_1 may be chosen arbitrarily; $\theta_1 = 0.0$ will be our convention. Exactly the same statements apply to joint n.

These conventions have been chosen so that, in a case where a quantity could be assigned arbitrarily, a zero value is assigned so that later calculations will be as simple as possible.

Link parameters

Hence, any robot can be described kinematically by giving the values of four quantities for each link. Two describe the link itself, and two describe the link's connection to a neighboring link. In the usual case of a revolute joint, θ_i is called the **joint variable**, and the other three quantities would be fixed **link parameters**. For prismatic joints, d_i is the joint variable, and the other three quantities are fixed link parameters. The definition of mechanisms by means of these quantities is a convention usually called the **Denavit–Hartenberg notation** [1].[4] Other methods of describing mechanisms are available, but are not presented here.

At this point, we could inspect any mechanism and determine the Denavit–Hartenberg parameters that describe it. For a six-jointed robot, 18 numbers would be required to describe the fixed portion of its kinematics completely. In the case of a six-jointed robot with all revolute joints, the 18 numbers are in the form of six sets of $(\alpha_i, \alpha_i, d_i)$.

EXAMPLE 3.2

Two links, as described in Fig. 3.3, are connected as links 1 and 2 of a robot. Joint 2 is composed of a "B" bearing of link 1 and an "A" bearing of link 2, arranged so that the flat surfaces of the "A" and "B" bearings lie flush against each other. What is d_2?

The link offset d_2 is the offset at joint 2, which is the distance, measured along the joint 2 axis, between the mutual perpendicular of link 1 and that of link 2. From the drawings in Fig. 3.3, this is 2.5 inches.

Before introducing more examples, we will define a convention for attaching a frame to each link of the manipulator.

3.4 CONVENTION FOR AFFIXING FRAMES TO LINKS

In order to describe the location of each link relative to its neighbors, we define a frame attached to each link. The link frames are named by number according to the link to which they are attached. That is, frame $\{i\}$ is attached rigidly to link i.

[4] Note that many related conventions go by the name Denavit–Hartenberg, but differ in a few details. For example, the version used in this book differs from some of the robotic literature in the manner of frame numbering. Unlike some other conventions, in this book frame $\{i\}$ is attached to link i and has its origin lying on joint axis i.

Intermediate links in the chain

The convention we will use to locate frames on the links is as follows: The $\hat{Z}$-axis of frame $\{i\}$, called $\hat{Z}_i$, is coincident with the joint axis i. The origin of frame $\{i\}$ is located where the a_i perpendicular intersects the joint i axis. $\hat{X}_i$ points along a_i in the direction from joint i to joint $i+1$.

In the case of $a_i = 0$, $\hat{X}_i$ is normal to the plane of $\hat{Z}_i$ and $\hat{Z}_{i+1}$. We define α_i as being measured in the right-hand sense about $\hat{X}_i$, and so we see that the freedom of choosing the sign of α_i in this case corresponds to two choices for the direction of $\hat{X}_i$. $\hat{Y}_i$ is formed by the right-hand rule to complete the ith frame. Figure 3.5 shows the location of frames $\{i-1\}$ and $\{i\}$ for a general manipulator.

First and last links in the chain

We attach a frame to the base of the robot, or link 0, called frame $\{0\}$. This frame does not move; for the problem of arm kinematics, it can be considered the reference frame. We may describe the position of all other link frames in terms of this frame.

Frame $\{0\}$ is arbitrary, so it always simplifies matters to choose $\hat{Z}_0$ along axis 1 and to locate frame $\{0\}$ so that it coincides with frame $\{1\}$ when joint variable 1 is zero. Using this convention, we will always have $a_0 = 0.0$, $\alpha_0 = 0.0$. Additionally, this ensures that $d_1 = 0.0$ if joint 1 is revolute, or $\theta_1 = 0.0$ if joint 1 is prismatic.

For joint n revolute, the direction of $\hat{X}_N$ is chosen so that it aligns with $\hat{X}_{N-1}$ when $\theta_n = 0.0$, and the origin of frame $\{N\}$ is chosen so that $d_n = 0.0$. For joint n prismatic, the direction of $\hat{X}_N$ is chosen so that $\theta_n = 0.0$, and the origin of frame $\{N\}$ is chosen at the intersection of $\hat{X}_{N-1}$ and joint axis n when $d_n = 0.0$.

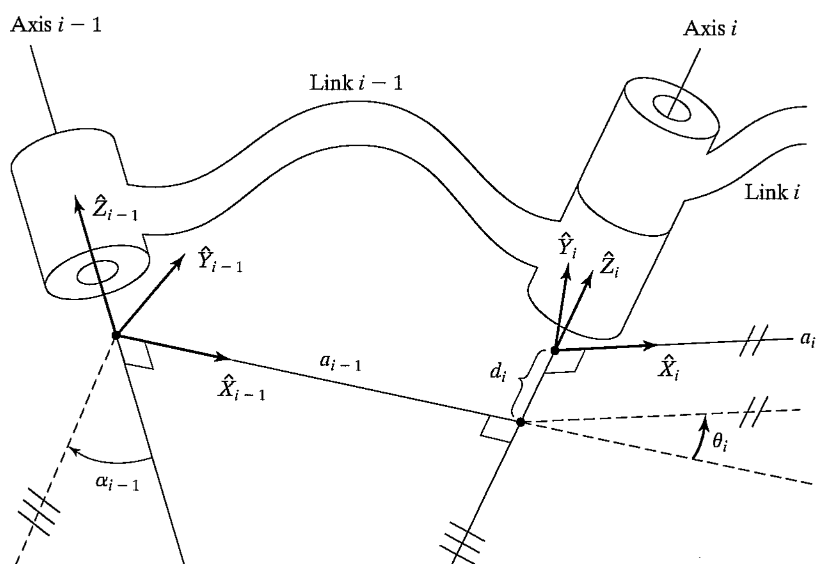


FIGURE 3.5: Link frames are attached so that frame $\{i\}$ is attached rigidly to link i.

Summary of the link parameters in terms of the link frames

If the link frames have been attached to the links according to our convention, the following definitions of the link parameters are valid:

$$a_i = \textit{the distance from } \hat{Z}_i \textit{ to } \hat{Z}_{i+1} \textit{ measured along } \hat{X}_i;$$

$$\alpha_i = \textit{the angle from } \hat{Z}_i \textit{ to } \hat{Z}_{i+1} \textit{ measured about } \hat{X}_i;$$

$$d_i = \textit{the distance from } \hat{X}_{i-1} \textit{ to } \hat{X}_i \textit{ measured along } \hat{Z}_i; \text{ and}$$

$$\theta_i = \textit{the angle from } \hat{X}_{i-1} \textit{ to } \hat{X}_i \textit{ measured about } \hat{Z}_i.$$

We usually choose $a_i > 0$, because it corresponds to a distance; however, α_i, d_i, and θ_i are signed quantities.

A final note on uniqueness is warranted. The convention outlined above does not result in a unique attachment of frames to links. First of all, when we first align the $\hat{Z}_i$ axis with joint axis i, there are two choices of direction in which to point $\hat{Z}_i$. Furthermore, in the case of intersecting joint axes (i.e., $a_i = 0$), there are two choices for the direction of $\hat{X}_i$, corresponding to the choice of signs for the normal to the plane containing $\hat{Z}_i$ and $\hat{Z}_{i+1}$. When axes i and $i+1$ are parallel, the choice of origin location for $\{i\}$ is arbitrary (though generally chosen in order to cause d_i to be zero). Also, when prismatic joints are present, there is quite a bit of freedom in frame assignment. (See also Example 3.5.)

Summary of link-frame attachment procedure

The following is a summary of the procedure to follow when faced with a new mechanism, in order to properly attach the link frames:

1. Identify the joint axes and imagine (or draw) infinite lines along them. For steps 2 through 5 below, consider two of these neighboring lines (at axes i and $i+1$).
2. Identify the common perpendicular between them, or point of intersection. At the point of intersection, or at the point where the common perpendicular meets the ith axis, assign the link-frame origin.
3. Assign the $\hat{Z}_i$ axis pointing along the ith joint axis.
4. Assign the $\hat{X}_i$ axis pointing along the common perpendicular, or, if the axes intersect, assign $\hat{X}_i$ to be normal to the plane containing the two axes.
5. Assign the $\hat{Y}_i$ axis to complete a right-hand coordinate system.
6. Assign $\{0\}$ to match $\{1\}$ when the first joint variable is zero. For $\{N\}$, choose an origin location and $\hat{X}_N$ direction freely, but generally so as to cause as many linkage parameters as possible to become zero.

EXAMPLE 3.3

Figure 3.6(a) shows a three-link planar arm. Because all three joints are revolute, this manipulator is sometimes called an ***RRR*** (or **3*R***) **mechanism**. Fig. 3.6(b) is a schematic representation of the same manipulator. Note the double hash marks

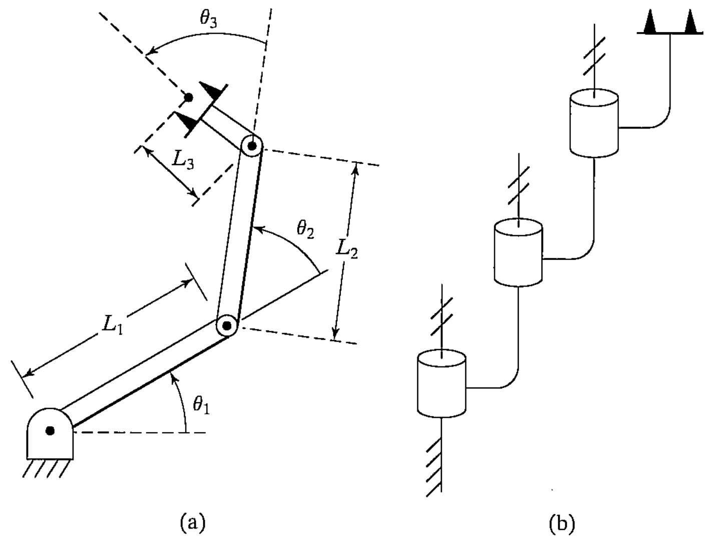


FIGURE 3.6: A three-link planar arm. On the right, we show the same manipulator by means of a simple schematic notation. Hash marks on the axes indicate that they are mutually parallel.

indicated on each of the three axes, which indicate that these axes are parallel. Assign link frames to the mechanism and give the Denavit–Hartenberg parameters.

We start by defining the reference frame, frame {0}. It is fixed to the base and aligns with frame {1} when the first joint variable (θ_1) is zero. Therefore, we position frame {0} as shown in Fig. 3.7 with $\hat{Z}_0$ aligned with the joint-1 axis. For this arm, all joint axes are oriented perpendicular to the plane of the arm. Because the arm

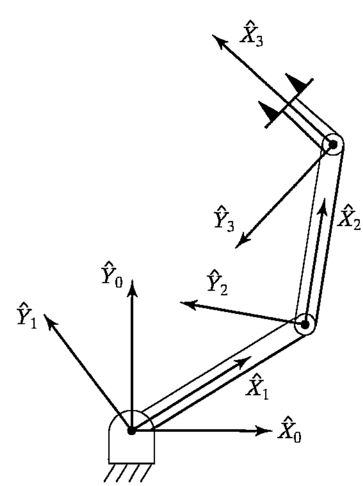


FIGURE 3.7: Link-frame assignments.

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	0	L_1	0	θ_2
3	0	L_2	0	θ_3

FIGURE 3.8: Link parameters of the three-link planar manipulator.

lies in a plane with all $\hat{Z}$ axes parallel, there are no link offsets—all d_i are zero. All joints are rotational, so when they are at zero degrees, all $\hat{X}$ axes must align.

With these comments in mind, it is easy to find the frame assignments shown in Fig. 3.7. The corresponding link parameters are shown in Fig. 3.8.

Note that, because the joint axes are all parallel and all the $\hat{Z}$ axes are taken as pointing out of the paper, all α_i are zero. This is obviously a very simple mechanism. Note also that our kinematic analysis always ends at a frame whose origin lies on the last joint axis; therefore, l_3 does not appear in the link parameters. Such final offsets to the end-effector are dealt with separately later.

EXAMPLE 3.4

Figure 3.9(a) shows a robot having three degrees of freedom and one prismatic joint. This manipulator can be called an "*RPR* mechanism," in a notation that specifies the type and order of the joints. It is a "cylindrical" robot whose first two joints are analogous to polar coordinates when viewed from above. The last joint (joint 3) provides "roll" for the hand. Figure 3.9(b) shows the same manipulator in schematic

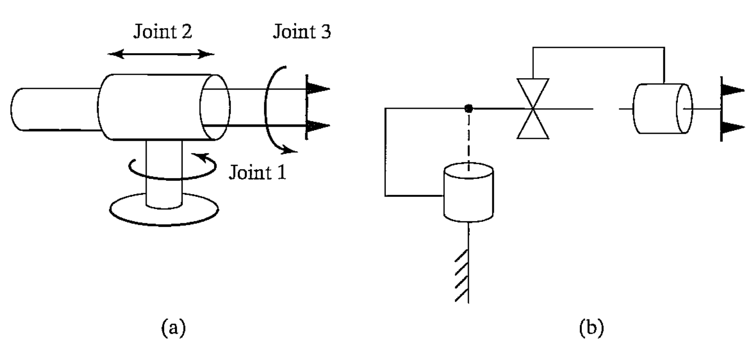


FIGURE 3.9: Manipulator having three degrees of freedom and one prismatic joint.

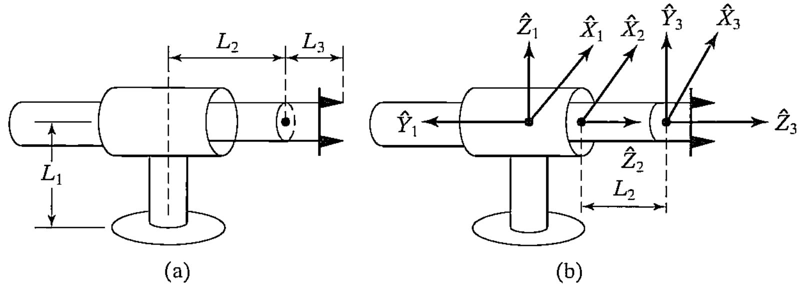


FIGURE 3.10: Link-frame assignments.

form. Note the symbol used to represent prismatic joints, and note that a "dot" is used to indicate the point at which two adjacent axes intersect. Also, the fact that axes 1 and 2 are orthogonal has been indicated.

Figure 3.10(a) shows the manipulator with the prismatic joint at minimum extension; the assignment of link frames is shown in Fig. 3.10(b).

Note that frame {0} and frame {1} are shown as exactly coincident in this figure, because the robot is drawn for the position $\theta_1 = 0$. Note that frame {0}, although not at the bottom of the flanged base of the robot, is nonetheless rigidly affixed to link 0, the nonmoving part of the robot. Just as our link frames are not used to describe the kinematics all the way out to the hand, they need not be attached all the way back to the lowest part of the base of the robot. It is sufficient that frame {0} be attached anywhere to the nonmoving link 0, and that frame $\{N\}$, the final frame, be attached anywhere to the last link of the manipulator. Other offsets can be handled later in a general way.

Note that rotational joints rotate about the $\hat{Z}$ axis of the associated frame, but prismatic joints slide along $\hat{Z}$. In the case where joint i is prismatic, θ_i is a fixed constant, and d_i is the variable. If d_i is zero at minimum extension of the link, then frame {2} should be attached where shown, so that d_2 will give the true offset. The link parameters are shown in Fig. 3.11.

Note that θ_2 is zero for this robot and that d_2 is a variable. Axes 1 and 2 intersect, so a_1 is zero. Angle α_1 must be 90 degrees in order to rotate $\hat{Z}_1$ so as to align with $\hat{Z}_2$ (about $\hat{X}_1$).

EXAMPLE 3.5

Figure 3.12(a) shows a three-link, $3R$ manipulator for which joint axes 1 and 2 intersect and axes 2 and 3 are parallel. Figure 3.12(b) shows the kinematic schematic of the manipulator. Note that the schematic includes annotations indicating that the first two axes are orthogonal and that the last two are parallel.

Demonstrate the nonuniqueness of frame assignments and of the Denavit–Hartenberg parameters by showing several possible correct assignments of frames {1} and {2}.

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_1
2	90°	0	d_2	0
3	0	0	L_2	θ_3

FIGURE 3.11: Link parameters for the *RPR* manipulator of Example 3.4.

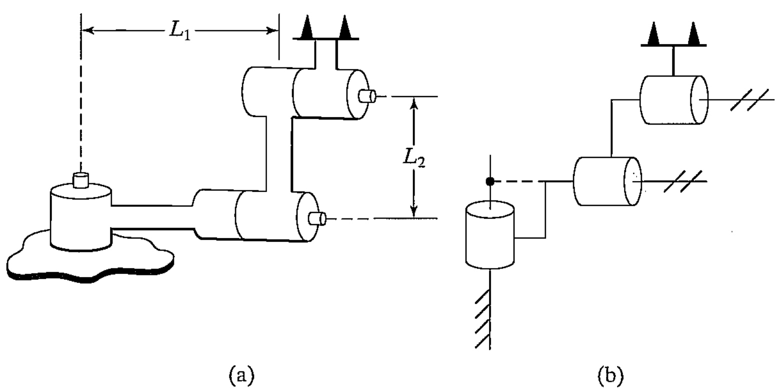


FIGURE 3.12: Three-link, nonplanar manipulator.

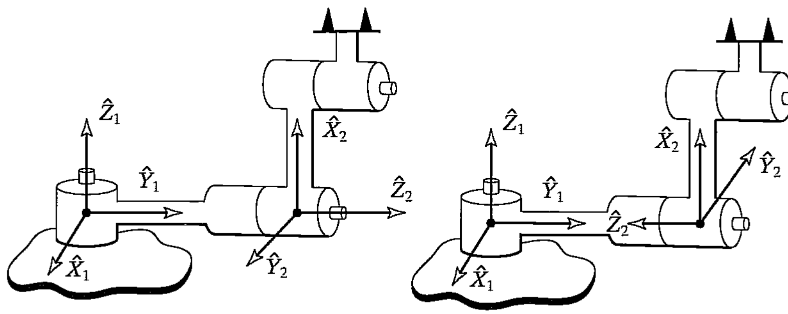
Figure 3.13 shows two possible frame assignments and corresponding parameters for the two possible choices of direction of $\hat{Z}_2$.

In general, when $\hat{Z}_i$ and $\hat{Z}_{i+1}$ intersect, there are two choices for $\hat{X}_i$. In this example, joint axes 1 and 2 intersect, so there are two choices for the direction of $\hat{X}_1$. Figure 3.14 shows two more possible frame assignments, corresponding to the second choice of $\hat{X}_1$.

In fact, there are four more possibilities, corresponding to the preceding four choices, but with $\hat{Z}_1$ pointing downward.

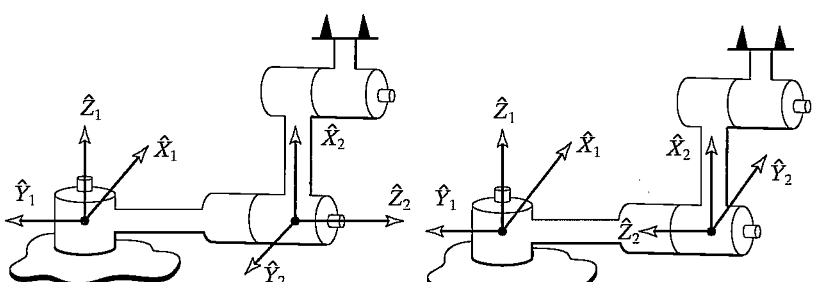
3.5 MANIPULATOR KINEMATICS

In this section, we derive the general form of the transformation that relates the frames attached to neighboring links. We then concatenate these individual transformations to solve for the position and orientation of link n relative to link 0.



$a_1 = 0$	$a_2 = L_2$		$a_1 = 0$	$a_2 = L_2$	
$\alpha_1 = -90°$	$\alpha_2 = 0$	$\theta_2 = -90°$	$\alpha_1 = 90°$	$\alpha_2 = 0$	$\theta_2 = 90°$
$d_1 = 0$	$d_2 = L_1$		$d_1 = 0$	$d_2 = -L_1$	

FIGURE 3.13: Two possible frame assignments.



$a_1 = 0$	$a_2 = L_2$		$a_1 = 0$	$a_2 = L_2$	
$\alpha_1 = 90°$	$\alpha_2 = 0$	$\theta_2 = 90°$	$\alpha_1 = -90°$	$\alpha_2 = 0$	$\theta_2 = -90°$
$d_1 = 0$	$d_2 = L_1$		$d_1 = 0$	$d_2 = -L_1$	

FIGURE 3.14: Two more possible frame assignments.

Derivation of link transformations

We wish to construct the transform that defines frame $\{i\}$ relative to the frame $\{i-1\}$. In general, this transformation will be a function of the four link parameters. For any *given* robot, this transformation will be a function of only one variable, the other three parameters being fixed by mechanical design. By defining a frame for each link, we have broken the kinematics problem into n subproblems. In order to solve each of these subproblems, namely ${}^{i-1}_{\ i}T$, we will further break each subproblem into four subsubproblems. *Each of these four transformations will be a function of one link parameter only and will be simple enough that we can write down its form by inspection.* We begin by defining three intermediate frames for each link—$\{P\}$, $\{Q\}$, and $\{R\}$.

Figure 3.15 shows the same pair of joints as before with frames $\{P\}$, $\{Q\}$, and $\{R\}$ defined. Note that only the $\hat{X}$ and $\hat{Z}$ axes are shown for each frame, to make the drawing clearer. Frame $\{R\}$ differs from frame $\{i-1\}$ only by a rotation of α_{i-1}.

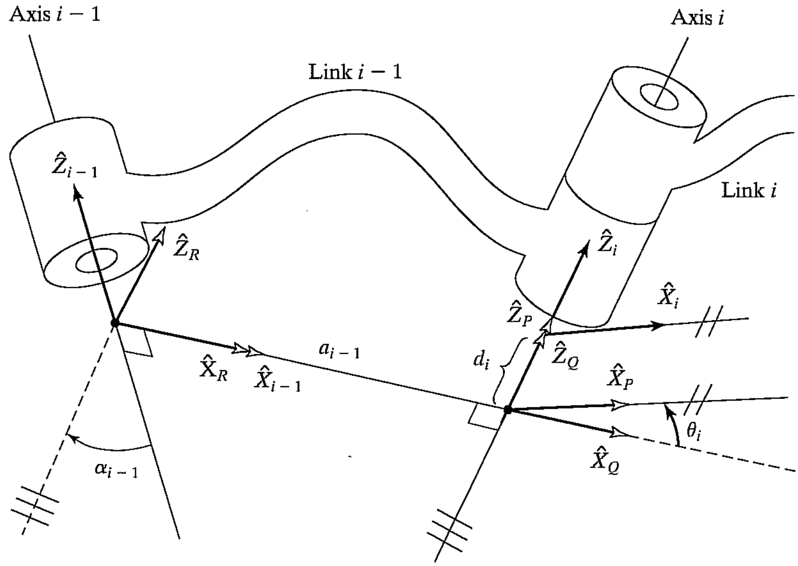


FIGURE 3.15: Location of intermediate frames $\{P\}$, $\{Q\}$, and $\{R\}$.

Frame $\{Q\}$ differs from $\{R\}$ by a translation a_{i-1}. Frame $\{P\}$ differs from $\{Q\}$ by a rotation θ_i, and frame $\{i\}$ differs from $\{P\}$ by a translation d_i. If we wish to write the transformation that transforms vectors defined in $\{i\}$ to their description in $\{i-1\}$, we may write

$$^{i-1}P = {}^{i-1}_{R}T \, {}^{R}_{Q}T \, {}^{Q}_{P}T \, {}^{P}_{i}T \, {}^{i}P, \tag{3.1}$$

or

$$^{i-1}P = {}^{i-1}_{i}T \, {}^{i}P, \tag{3.2}$$

where

$$^{i-1}_{i}T = {}^{i-1}_{R}T \, {}^{R}_{Q}T \, {}^{Q}_{P}T \, {}^{P}_{i}T. \tag{3.3}$$

Considering each of these transformations, we see that (3.3) may be written

$$^{i-1}_{i}T = R_X(\alpha_{i-1}) D_X(a_{i-1}) R_Z(\theta_i) D_Z(d_i), \tag{3.4}$$

or

$$^{i-1}_{i}T = \text{Screw}_X(a_{i-1}, \alpha_{i-1}) \, \text{Screw}_Z(d_i, \theta_i), \tag{3.5}$$

where the notation $\text{Screw}_Q(r, \phi)$ stands for the combination of a translation along an axis $\hat{Q}$ by a distance r and a rotation about the same axis by an angle ϕ. Multiplying out (3.4), we obtain the general form of $^{i-1}_{i}T$:

$$^{i-1}_{i}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{3.6}$$

EXAMPLE 3.6

Using the link parameters shown in Fig. 3.11 for the robot of Fig. 3.9, compute the individual transformations for each link.

Substituting the parameters into (3.6), we obtain

$$
\begin{aligned}
{}^0_1T &= \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\
{}^1_2T &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & -d_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\
{}^2_3T &= \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & 0 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 1 & l_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}.
\end{aligned}
\tag{3.7}
$$

Once having derived these link transformations, we will find it a good idea to check them against common sense. For example, the elements of the fourth column of each transform should give the coordinates of the origin of the next higher frame.

Concatenating link transformations

Once the link frames have been defined and the corresponding link parameters found, developing the kinematic equations is straightforward. From the values of the link parameters, the individual link-transformation matrices can be computed. Then, the link transformations can be multiplied together to find the single transformation that relates frame $\{N\}$ to frame $\{0\}$:

$$
{}^0_NT = {}^0_1T\,{}^1_2T\,{}^2_3T \ldots {}^{N-1}_{\ \ N}T. \tag{3.8}
$$

This transformation, ${}^0_NT$, will be a function of all n joint variables. If the robot's joint-position sensors are queried, the Cartesian position and orientation of the last link can be computed by ${}^0_NT$.

3.6 ACTUATOR SPACE, JOINT SPACE, AND CARTESIAN SPACE

The position of all the links of a manipulator of n degrees of freedom can be specified with a set of n joint variables. This set of variables is often referred to as the $n \times 1$ **joint vector**. The space of all such joint vectors is referred to as **joint space**. Thus far in this chapter, we have been concerned with computing the **Cartesian space** description from knowledge of the joint-space description. We use the term *Cartesian space* when position is measured along orthogonal axes and orientation is measured according to any of the conventions outlined in Chapter 2. Sometimes, the terms **task-oriented space** and **operational space** are used for what we will call Cartesian space.

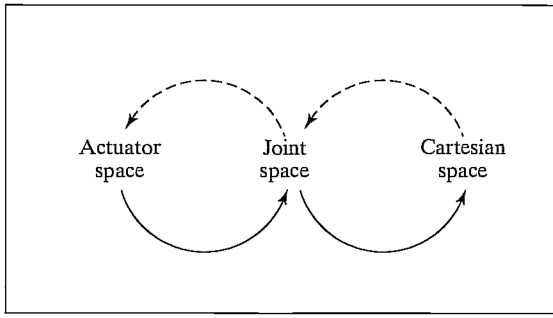


FIGURE 3.16: Mappings between kinematic descriptions.

So far, we have implicitly assumed that each kinematic joint is actuated directly by some sort of actuator. However, in the case of many industrial robots, this is not so. For example, sometimes two actuators work together in a differential pair to move a single joint, or sometimes a linear actuator is used to rotate a revolute joint, through the use of a four-bar linkage. In these cases, it is helpful to consider the notion of *actuator positions*. The sensors that measure the position of the manipulator are often located at the actuators, so some computations must be performed to realize the joint vector as a function of a set of actuator values, or **actuator vector**.

As is indicated in Fig. 3.16, there are three representations of a manipulator's position and orientation: descriptions in **actuator space**, in **joint space**, and in **Cartesian space**. In this chapter, we are concerned with the mappings between representations, as indicated by the solid arrows in Fig. 3.16. In Chapter 4, we will consider the inverse mappings, indicated by the dashed arrows.

The ways in which actuators might be connected to move a joint are quite varied; they might be catalogued, but we will not do so here. For each robot we design or seek to analyze, the correspondence between actuator positions and joint positions must be solved. In the next section, we will solve an example problem for an industrial robot.

3.7 EXAMPLES: KINEMATICS OF TWO INDUSTRIAL ROBOTS

Current industrial robots are available in many different kinematic configurations [2], [3]. In this section, we work out the kinematics of two typical industrial robots. First we consider the Unimation PUMA 560, a rotary-joint manipulator with six degrees of freedom. We will solve for the kinematic equations as functions of the joint angles. For this example, we will skip the additional problem of the relationship between actuator space and joint space. Second, we consider the Yasukawa Motoman L-3, a robot with five degrees of freedom and rotary joints. This example is done in detail, including the actuator-to-joint transformations. This example may be skipped on first reading of the book.

The PUMA 560

The Unimation PUMA 560 (Fig. 3.17) is a robot with six degrees of freedom and all rotational joints (i.e., it is a $6R$ mechanism). It is shown in Fig. 3.18, with

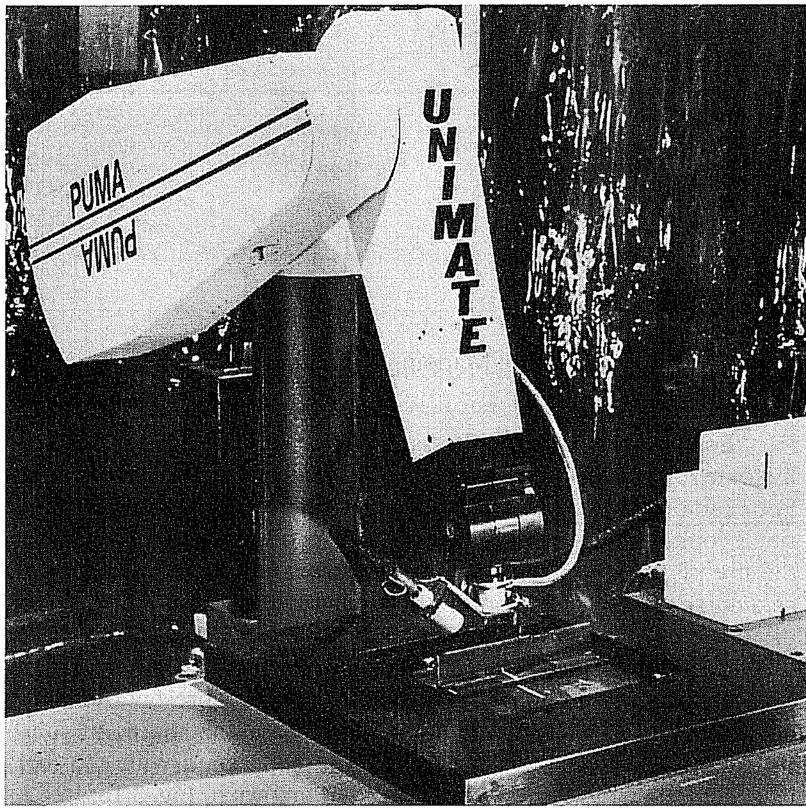


FIGURE 3.17: The Unimation PUMA 560. Courtesy of Unimation Incorporated, Shelter Rock Lane, Danbury, Conn.

link-frame assignments in the position corresponding to all joint angles equal to zero.[5] Figure 3.19 shows a detail of the forearm of the robot.

Note that the frame {0} (not shown) is coincident with frame {1} when θ_1 is zero. Note also that, for this robot, as for many industrial robots, the joint axes of joints 4, 5, and 6 all intersect at a common point, and this point of intersection coincides with the origin of frames {4}, {5}, and {6}. Furthermore, the joint axes 4, 5, and 6 are mutually orthogonal. This wrist mechanism is illustrated schematically in Fig. 3.20.

The link parameters corresponding to this placement of link frames are shown in Fig. 3.21. In the case of the PUMA 560, a gearing arrangement in the wrist of the manipulator couples together the motions of joints 4, 5, and 6. What this means is that, for these three joints, we must make a distinction between joint space and actuator space and solve the complete kinematics in two steps. However, in this example, we will consider only the kinematics from joint space to Cartesian space.

[5]Unimation has used a slightly different assignment of zero location of the joints, such that $\theta_3^* = \theta_3 - 180°$, where θ_3^* is the position of joint 3 in Unimation's convention.

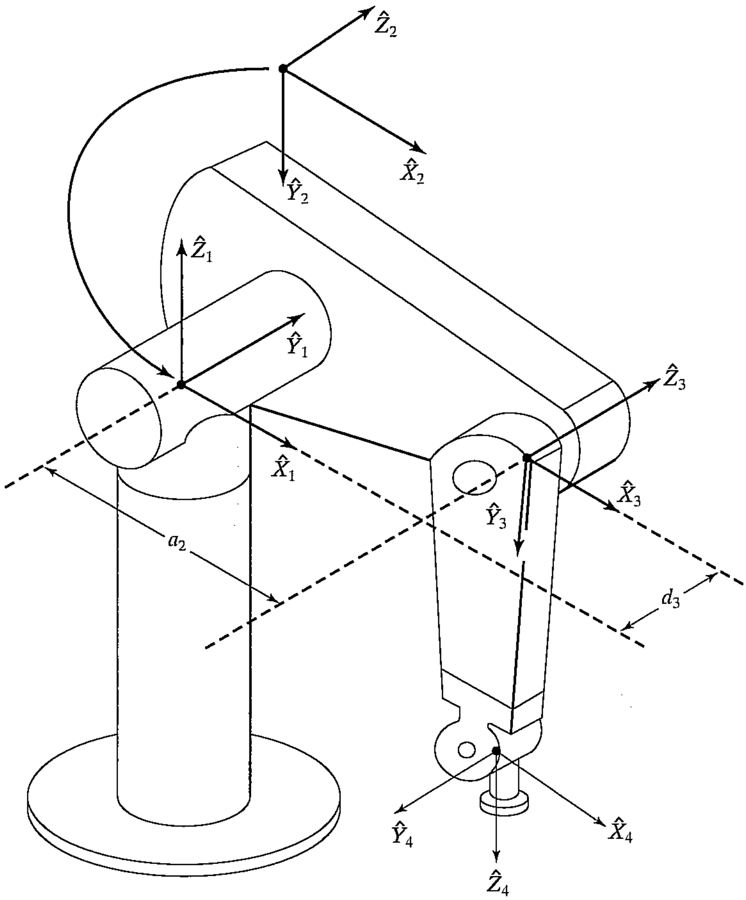


FIGURE 3.18: Some kinematic parameters and frame assignments for the PUMA 560 manipulator.

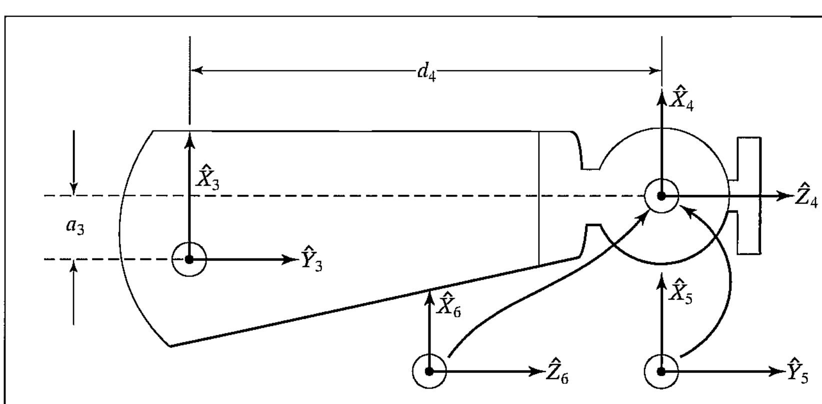


FIGURE 3.19: Kinematic parameters and frame assignments for the forearm of the PUMA 560 manipulator.

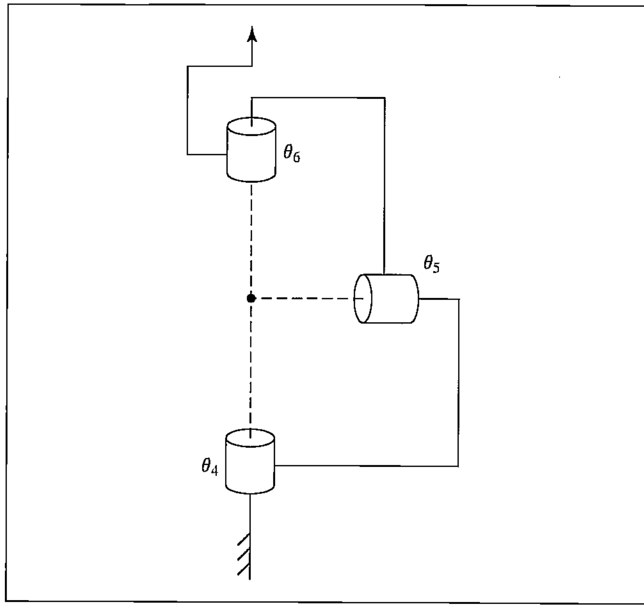


FIGURE 3.20: Schematic of a $3R$ wrist in which all three axes intersect at a point and are mutually orthogonal. This design is used in the PUMA 560 manipulator and many other industrial robots.

i	$\alpha_i - 1$	$a_i - 1$	d_i	θi
1	0	0	0	θ_1
2	$-90°$	0	0	θ_2
3	0	a_2	d_3	θ_3
4	$-90°$	a_3	d_4	θ_4
5	$90°$	0	0	θ_5
6	$-90°$	0	0	θ_6

FIGURE 3.21: Link parameters of the PUMA 560.

Using (3.6), we compute each of the link transformations:

$$
{}^{0}_{1}T = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

$$
{}^{1}_{2}T = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_2 & -c\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

$$
{}^{2}_{3}T = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & a_2 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

$$
{}^{3}_{4}T = \begin{bmatrix} c\theta_4 & -s\theta_4 & 0 & a_3 \\ 0 & 0 & 1 & d_4 \\ -s\theta_4 & -c\theta_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

$$
{}^{4}_{5}T = \begin{bmatrix} c\theta_5 & -s\theta_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s\theta_5 & c\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

$$
{}^{5}_{6}T = \begin{bmatrix} c\theta_6 & -s\theta_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_6 & -c\theta_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{3.9}
$$

We now form ${}^{0}_{6}T$ by matrix multiplication of the individual link matrices. While forming this product, we will derive some subresults that will be useful when solving the inverse kinematic problem in Chapter 4. We start by multiplying ${}^{4}_{5}T$ and ${}^{5}_{6}T$; that is,

$$
{}^{4}_{6}T = {}^{4}_{5}T\,{}^{5}_{6}T = \begin{bmatrix} c_5c_6 & -c_5s_6 & -s_5 & 0 \\ s_6 & c_6 & 0 & 0 \\ s_5c_6 & -s_5s_6 & c_5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3.10}
$$

where c_5 is shorthand for $\cos\theta_5$, s_5 for $\sin\theta_5$, and so on.[6] Then we have

$$
{}^{3}_{6}T = {}^{3}_{4}T\,{}^{4}_{6}T = \begin{bmatrix} c_4c_5c_6 - s_4s_6 & -c_4c_5s_6 - s_4c_6 & -c_4s_5 & a_3 \\ s_5c_6 & -s_5s_6 & c_5 & d_4 \\ -s_4c_5c_6 - c_4s_6 & s_4c_5s_6 - c_4c_6 & s_4s_5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{3.11}
$$

[6] Depending on the amount of space available to show expressions, we use any of the following three forms: $\cos\theta_5$, $c\theta_5$, or c_5.

Because joints 2 and 3 are always parallel, multiplying ${}^{1}_{2}T$ and ${}^{2}_{3}T$ first and then applying sum-of-angle formulas will yield a somewhat simpler final expression. This can be done whenever two rotational joints have parallel axes and we have

$$
{}^{1}_{3}T = {}^{1}_{2}T\,{}^{2}_{3}T = \begin{bmatrix} c_{23} & -s_{23} & 0 & a_2c_2 \\ 0 & 0 & 1 & d_3 \\ -s_{23} & -c_{23} & 0 & -a_2s_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3.12}
$$

where we have used the sum-of-angle formulas (from Appendix A):

$$
\begin{aligned}
c_{23} &= c_2c_3 - s_2s_3, \\
s_{23} &= c_2s_3 + s_2c_3.
\end{aligned}
$$

Then we have

$$
{}^{1}_{6}T = {}^{1}_{3}T\,{}^{3}_{6}T = \begin{bmatrix} {}^{1}r_{11} & {}^{1}r_{12} & {}^{1}r_{13} & {}^{1}p_x \\ {}^{1}r_{21} & {}^{1}r_{22} & {}^{1}r_{23} & {}^{1}p_y \\ {}^{1}r_{31} & {}^{1}r_{32} & {}^{1}r_{33} & {}^{1}p_z \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

where

$$
\begin{aligned}
{}^{1}r_{11} &= c_{23}[c_4c_5c_6 - s_4s_6] - s_{23}s_5s_6, \\
{}^{1}r_{21} &= -s_4c_5c_6 - c_4s_6, \\
{}^{1}r_{31} &= -s_{23}[c_4c_5c_6 - s_4s_6] - c_{23}s_5c_6, \\
{}^{1}r_{12} &= -c_{23}[c_4c_5s_6 + s_4c_6] + s_{23}s_5s_6, \\
{}^{1}r_{22} &= s_4c_5s_6 - c_4c_6, \\
{}^{1}r_{32} &= s_{23}[c_4c_5s_6 + s_4c_6] + c_{23}s_5s_6, \\
{}^{1}r_{13} &= -c_{23}c_4s_5 - s_{23}c_5, \\
{}^{1}r_{23} &= s_4s_5, \\
{}^{1}r_{33} &= s_{23}c_4s_5 - c_{23}c_5, \\
{}^{1}p_x &= a_2c_2 + a_3c_{23} - d_4s_{23}, \\
{}^{1}p_y &= d_3, \\
{}^{1}p_x &= -a_3s_{23} - a_2s_2 - d_4c_{23}.
\end{aligned} \tag{3.13}
$$

Finally, we obtain the product of all six link transforms:

$$
{}^{0}_{6}T = {}^{0}_{1}T\,{}^{1}_{6}T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}.
$$

Here,

$$
\begin{aligned}
r_{11} &= c_1[c_{23}(c_4c_5c_6 - s_4s_5) - s_{23}s_5c_5] + s_1(s_4c_5c_6 + c_4s_6), \\
r_{21} &= s_1[c_{23}(c_4c_5c_6 - s_4s_6) - s_{23}s_5c_6 - c_1(s_4c_5c_6 + c_4s_6), \\
r_{31} &= -s_{23}(c_4c_5c_6 - s_4s_6) - c_{23}s_5c_6, \\
\\
r_{12} &= c_1[c_{23}(-c_4c_5s_6 - s_4c_6) + s_{23}s_5s_6] + s_1(c_4c_6 - s_4c_5s_6), \\
r_{22} &= s_1[c_{23}(-c_4c_5s_6 - s_4c_6) + s_{23}s_5s_6] - c_1(c_4c_6 - s_4c_5s_6), \\
r_{32} &= -s_{23}(-c_4c_5s_6 - s_4c_6) + c_{23}s_5s_6, \\
\\
r_{13} &= -c_1(c_{23}c_4s_5 + s_{23}c_5) - s_1s_4s_5, \\
r_{23} &= -s_1(c_{23}c_4s_5 + s_{23}c_5) + c_1s_4s_5, \\
r_{33} &= s_{23}c_4s_5 - c_{23}c_5, \\
\\
p_x &= c_1[a_2c_2 + a_3c_{23} - d_4s_{23}] - d_3s_1, \\
p_y &= s_1[a_2c_2 + a_3c_{23} - d_4s_{23}] + d_3c_1, \\
p_z &= -a_3s_{23} - a_2s_2 - d_4c_{23}.
\end{aligned}
\tag{3.14}
$$

Equations (3.14) constitute the kinematics of the PUMA 560. They specify how to compute the position and orientation of frame {6} relative to frame {0} of the robot. These are the basic equations for all kinematic analysis of this manipulator.

The Yasukawa Motoman L-3

The Yasukawa Motoman L-3 is a popular industrial manipulator with five degrees of freedom (Fig. 3.22). Unlike the examples we have seen thus far, the Motoman is not a simple open kinematic chain, but rather makes use of two linear actuators coupled to links 2 and 3 with four-bar linkages. Also, through a chain drive, joints 4 and 5 are operated by two actuators in a differential arrangement.

In this example, we will solve the kinematics in two stages. First, we will solve for joint angles from actuator positions; second, we will solve for Cartesian position and orientation of the last link from joint angles. In this second stage, we can treat the system as if it were a simple open-kinematic-chain $5R$ device.

Figure 3.23 shows the linkage mechanism that connects actuator number 2 to links 2 and 3 of the robot. The actuator is a linear one that directly controls the length of the segment labeled DC. Triangle ABC is fixed, as is the length BD. Joint 2 pivots about point B, and the actuator pivots slightly about point C as the linkage moves. We give the following names to the constants (lengths and angles) associated with actuator 2:

$$\gamma_2 = AB, \phi_2 = AC, \alpha_2 = BC,$$

$$\beta_2 = BD, \Omega_2 = \angle JBD, l_2 = BJ,$$

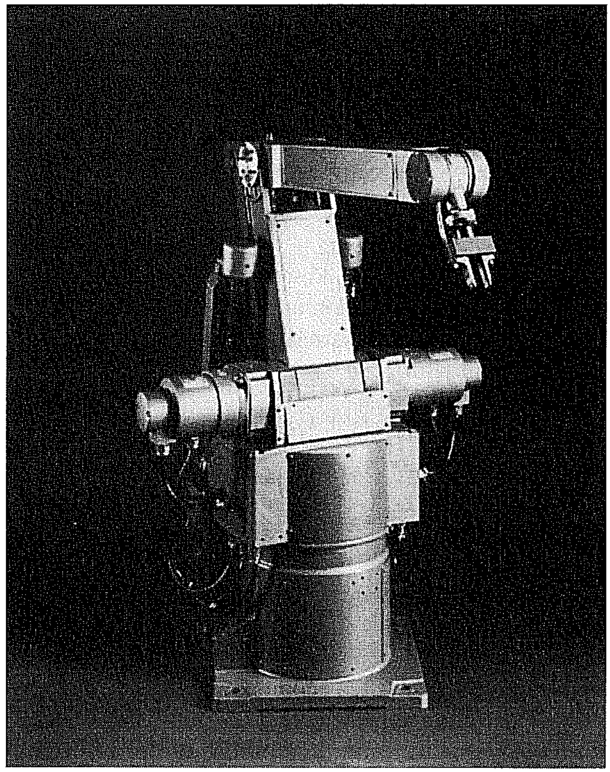


FIGURE 3.22: The Yasukawa Motoman L-3. Courtesy of Yasukawa.

we give the following names to the variables:

$$\theta_2 = -\angle JBQ, \psi_2 = \angle CBD, g_2 = DC.$$

Figure 3.24 shows the linkage mechanism that connects actuator number 3 to links 2 and 3 of the robot. The actuator is a linear one that directly controls the length of the segment labeled HG. Triangle EFG is fixed, as is the length FH. Joint 3 pivots about point J, and the actuator pivots slightly about point G as the linkage moves. We give the following names to the constants (lengths and angles) associated with actuator 3:

$$\gamma_3 = EF, \phi_3 = EG, \alpha_3 = GF,$$
$$\beta_3 = HF, l_3 = JK.$$

We give the following names to the variables:

$$\theta_3 = \angle PJK, \ \psi_3 = \angle GFH, \ g_3 = GH.$$

This arrangement of actuators and linkages has the following functional effect. Actuator 2 is used to position joint 2; while it is doing so, link 3 remains in the same orientation relative to the base of the robot. Actuator 3 is used to adjust

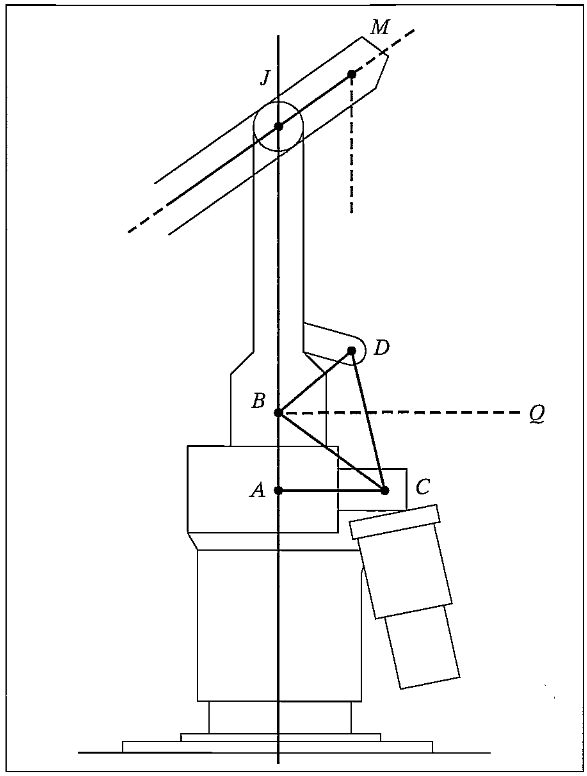


FIGURE 3.23: Kinematic details of the Yasukawa actuator-2 linkage.

the orientation of link 3 relative to the base of the robot (rather than relative to the preceding link as in a serial-kinematic-chain robot). One purpose of such a linkage arrangement is to increase the structural rigidity of the main linkages of the robot. This often pays off in terms of an increased ability to position the robot precisely.

The actuators for joints 4 and 5 are attached to link 1 of the robot with their axes aligned with that of joint 2 (points B and F in Figs. 3.23 and 3.24). They operate the wrist joints through two sets of chains—one set located interior to link 2, the second set interior to link 3. The effect of this transmission system, along with its interaction with the actuation of links 2 and 3, is described functionally as follows: Actuator 4 is used to position joint 4 relative to the base of the robot, rather than relative to the preceding link 3. This means that holding actuator 4 constant will keep link 4 at a constant orientation relative to the base of the robot, regardless of the positions of joints 2 and 3. Finally, actuator 5 behaves as if directly connected to joint 5.

We now state the equations that map a set of actuator values (A_i) to the equivalent set of joint values (θ_i). In this case, these equations were derived by straightforward plane geometry—mostly just application of the "law of cosines."[7]

[7]If a triangle's angles are labeled a, b, and c, where angle a is opposite side A, and so on, then $A^2 = B^2 + C^2 - 2BC \cos a$.

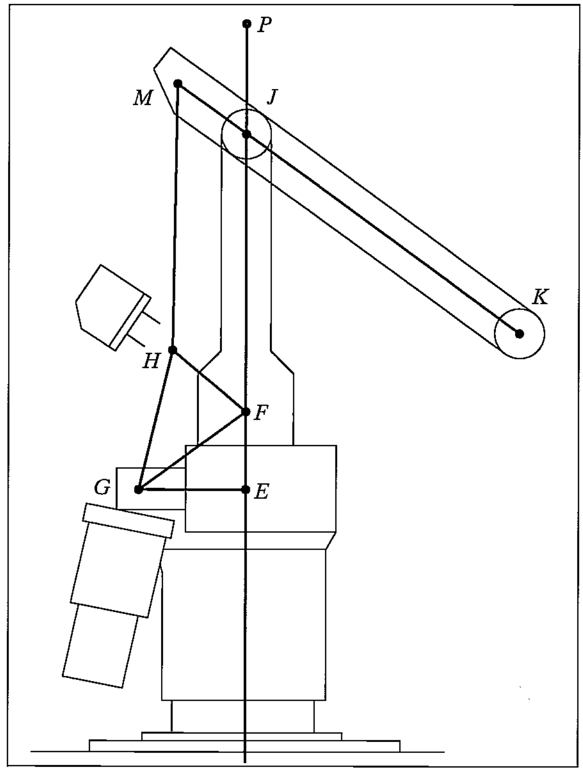


FIGURE 3.24: Kinematic details of the Yasukawa actuator-3 linkage.

Appearing in these equations are scale (k_i) and offset (λ_i) constants for each actuator. For example, actuator 1 is directly connected to joint axis 1, and so the conversion is simple; it is just a matter of a scale factor plus an offset. Thus,

$$\begin{aligned}
\theta_1 &= k_1 A_1 + \lambda_1, \\
\theta_2 &= \cos^{-1}\left(\frac{(k_2 A_2 + \lambda_2)^2 - \alpha_2^2 - \beta_2^2}{-2\alpha_2\beta_2}\right) + \tan^{-1}\left(\frac{\phi_2}{\gamma_2}\right) + \Omega_2 - 270^\circ, \\
\theta_3 &= \cos^{-1}\left(\frac{(k_3 A_3 + \lambda_3)^2 - \alpha_3^2 - \beta_3^2}{-2\alpha_3\beta_3}\right) - \theta_2 + \tan^{-1}\left(\frac{\phi_3}{\gamma_3}\right) - 90^\circ, \\
\theta_4 &= -k_4 A_4 - \theta_2 - \theta_3 + \lambda_4 + 180^\circ, \qquad (3.15) \\
\theta_5 &= -k_5 A_5 + \lambda_5.
\end{aligned}$$

Figure 3.25 shows the attachment of the link frames. In this figure, the manipulator is shown in a position corresponding to the joint vector $\Theta = (0, -90^\circ, 90^\circ, 90^\circ, 0)$. Figure 3.26 shows the link parameters for this manipulator. The resulting link-transformation matrices are

$$
{}^0_1T = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

$$
{}^1_2T = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_2 & -c\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

$$
{}^2_3T = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & l_2 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3.16}
$$

$$
{}^3_4T = \begin{bmatrix} c\theta_4 & -s\theta_4 & 0 & l_3 \\ s\theta_4 & c\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

$$
{}^4_5T = \begin{bmatrix} c\theta_5 & -s\theta_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s\theta_5 & c\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.
$$

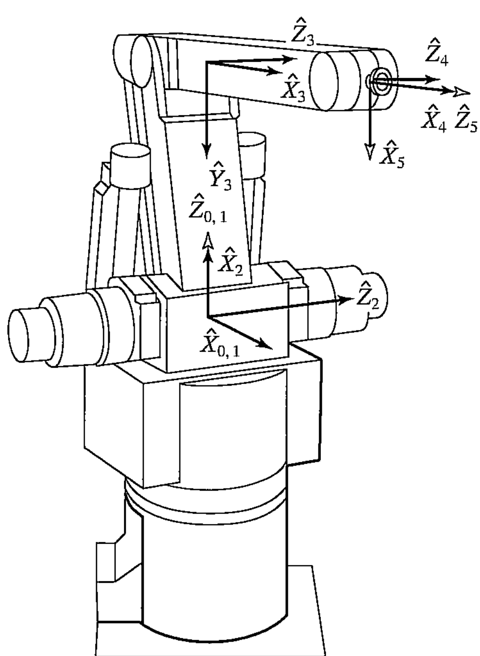


FIGURE 3.25: Assignment of link frames for the Yasukawa L-3.

i	$\alpha_i - 1$	$a_i - 1$	d_i	θ_i
1	0	0	0	θ_1
2	$-90°$	0	0	θ_2
3	0	L_2	0	θ_3
4	0	L_3	0	θ_4
5	$90°$	0	0	θ_5

FIGURE 3.26: Link parameters of the Yasukawa L-3 manipulator.

Forming the product to obtain ${}^0_5T$, we obtain

$$
{}^0_5T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & p_x \\ r_{21} & r_{22} & r_{23} & p_y \\ r_{31} & r_{32} & r_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix},
$$

where

$$
\begin{aligned}
r_{11} &= c_1 c_{234} c_5 - s_1 s_5, \\
r_{21} &= s_1 c_{234} c_5 + c_1 s_5, \\
r_{31} &= -s_{234} c_5, \\
r_{12} &= -c_1 c_{234} s_5 - s_1 c_5, \\
r_{22} &= -s_1 c_{234} s_5 + c_1 c_5, \\
r_{32} &= s_{234} s_5, \\
r_{13} &= c_1 s_{234}, \\
r_{23} &= s_1 s_{234}, \\
r_{33} &= c_{234},
\end{aligned}
$$

$$p_x = c_1(l_2c_2 + l_3c_{23}),$$
$$p_y = s_1(l_2c_2 + l_3c_{23}),$$
$$p_z = -l_2s_2 - l_3s_{23}. \tag{3.17}$$

We developed the kinematic equations for the Yasukawa Motoman in two steps. In the first step, we computed a joint vector from an actuator vector; in the second step, we computed a position and orientation of the wrist frame from the joint vector. If we wish to compute only Cartesian position and not joint angles, it is possible to derive equations that map directly from actuator space to Cartesian space. These equations are somewhat simpler computationally than the two-step approach. (See Exercise 3.10.)

3.8 FRAMES WITH STANDARD NAMES

As a matter of convention, it will be helpful if we assign specific names and locations to certain "standard" frames associated with a robot and its workspace. Figure 3.27 shows a typical situation in which a robot has grasped some sort of tool and is to position the tool tip to a user-defined location. The five frames indicated in Fig. 3.27 are so often referred to that we will define names for them. The naming and subsequent use of these five frames in a robot programming and control system facilitates providing general capabilities in an easily understandable way. All robot motions will be described in terms of these frames.

Brief definitions of the frames shown in Fig. 3.27 follow.

The base frame, $\{B\}$

$\{B\}$ is located at the base of the manipulator. It is merely another name for frame $\{0\}$. It is affixed to a nonmoving part of the robot, sometimes called link 0.

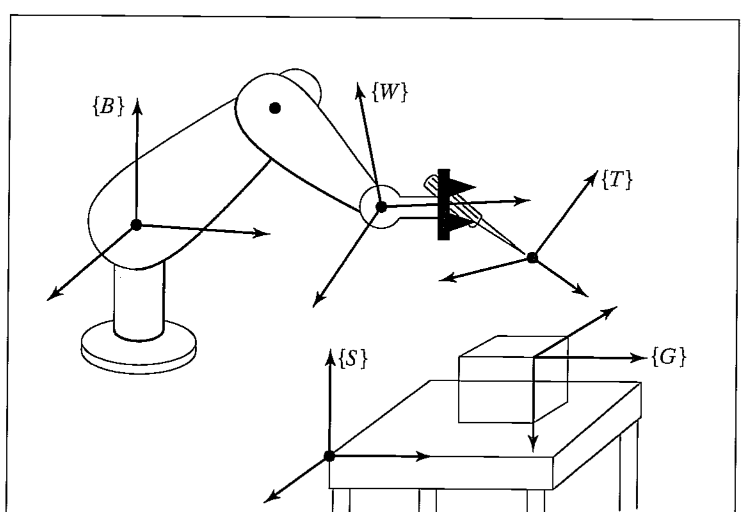


FIGURE 3.27: The standard frames.

The station frame, $\{S\}$

$\{S\}$ is located in a task-relevant location. In Fig. 3.28, it is at the corner of a table upon which the robot is to work. As far as the user of this robot system is concerned, $\{S\}$ is the universe frame, and all actions of the robot are performed relative to it. It is sometimes called the task frame, the world frame, or the universe frame. The station frame is always specified with respect to the base frame, that is, ${}^{B}_{S}T$.

The wrist frame, $\{W\}$

$\{W\}$ is affixed to the last link of the manipulator. It is another name for frame $\{N\}$, the link frame attached to the last link of the robot. Very often, $\{W\}$ has its origin fixed at a point called the wrist of the manipulator, and $\{W\}$ moves with the last link of the manipulator. It is defined relative to the base frame—that is, $\{W\} = {}^{B}_{W}T = {}^{0}_{N}T$.

The tool frame, $\{T\}$

$\{T\}$ is affixed to the end of any tool the robot happens to be holding. When the hand is empty, $\{T\}$ is usually located with its origin between the fingertips of the robot. The tool frame is always specified with respect to the wrist frame. In Fig. 3.28, the tool frame is defined with its origin at the tip of a pin that the robot is holding.

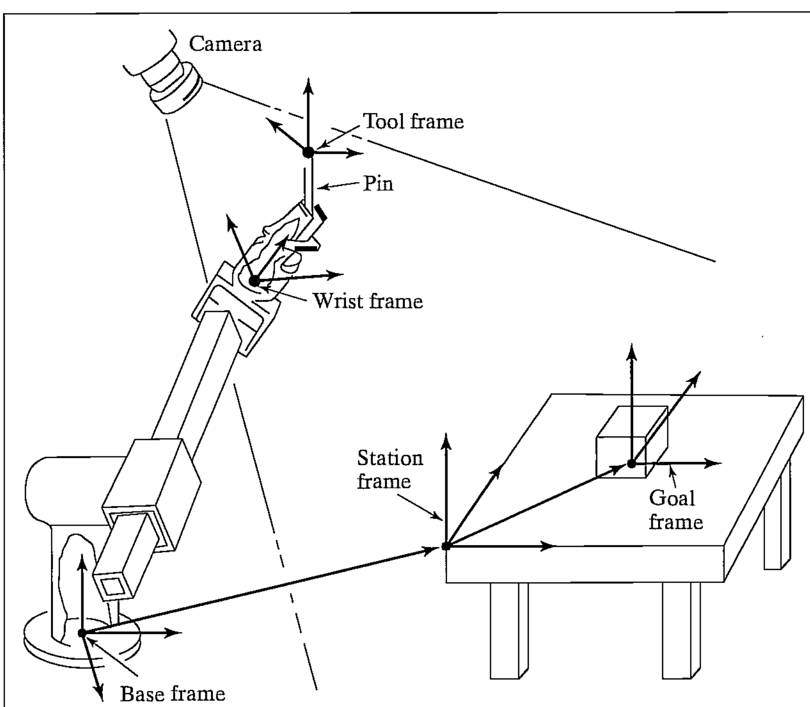


FIGURE 3.28: Example of the assignment of standard frames.

The goal frame, $\{G\}$

$\{G\}$ is a description of the location to which the robot is to move the tool. Specifically this means that, at the end of the motion, the tool frame should be brought to coincidence with the goal frame. $\{G\}$ is always specified relative to the station frame. In Fig. 3.28, the goal is located at a hole into which we want the pin to be inserted.

All robot motions may be described in terms of these frames without loss of generality. Their use helps to give us a standard language for talking about robot tasks.

3.9 WHERE IS THE TOOL?

One of the first capabilities a robot must have is to be able to calculate the position and orientation of the tool it is holding (or of its empty hand) with respect to a convenient coordinate system. That is, we wish to calculate the value of the tool frame, $\{T\}$, relative to the station frame, $\{S\}$. Once ${}^B_W T$ has been computed via the kinematic equations, we can use Cartesian transforms, as studied in Chapter 2, to calculate $\{T\}$ relative to $\{S\}$. Solving a simple transform equation leads to

$$ {}^S_T T = {}^B_S T^{-1}\, {}^B_W T\, {}^W_T T. \tag{3.18} $$

Equation (3.18) implements what is called the **WHERE** function in some robot systems. It computes "where" the arm is. For the situation in Fig. 3.28, the output of **WHERE** would be the position and orientation of the pin relative to the table top.

Equation (3.18) can be thought of as *generalizing* the kinematics. ${}^S_T T$ computes the kinematics due to the geometry of the linkages, along with a general transform (which might be considered a fixed link) at the base end (${}^B_S T$) and another at the end-effector (${}^W_T T$). These extra transforms allow us to include tools with offsets and twists and to operate with respect to an arbitrary station frame.

3.10 COMPUTATIONAL CONSIDERATIONS

In many practical manipulator systems, the time required to perform kinematic calculations is a consideration. In this section, we briefly discuss various issues involved in computing manipulator kinematics, as exemplified by (3.14), for the case of the PUMA 560.

One choice to be made is the use of fixed- or floating-point representation of the quantities involved. Many implementations use floating point for ease of software development, because the programmer does not have to be concerned with scaling operations capturing the relative magnitudes of the variables. However, when speed is crucial, fixed-point representation is quite possible, because the variables do not have a large dynamic range, and these ranges are fairly well known. Rough estimations of the number of bits needed in fixed-point representation seem to indicate that 24 are sufficient [4].

By factoring equations such as (3.14), it is possible to reduce the number of multiplications and additions—at the cost of creating local variables (usually a good trade-off). The point is to avoid computing common terms over and over throughout 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 and 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 ${}^{2}_{3}T$.

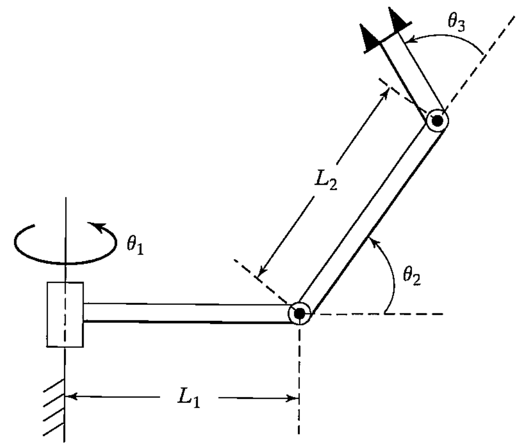


FIGURE 3.29: The $3R$ nonplanar arm (Exercise 3.3).

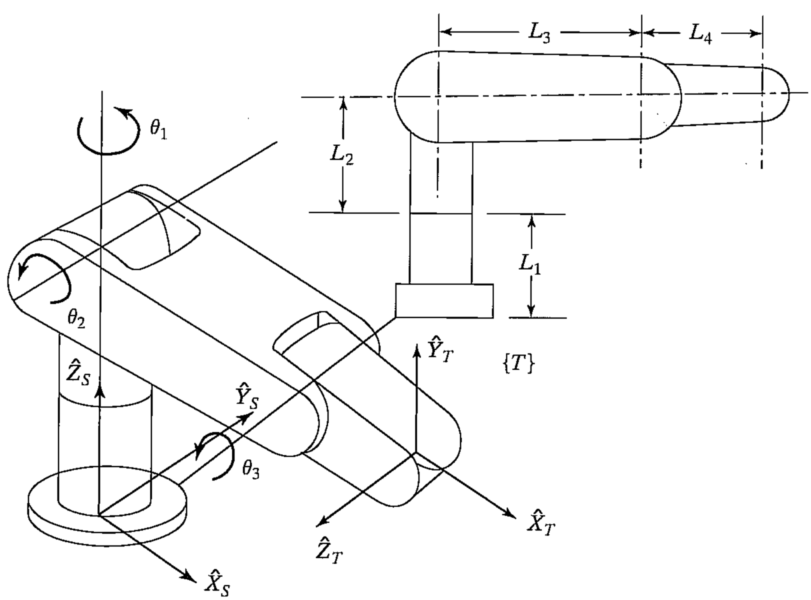


FIGURE 3.30: Two views of a $3R$ manipulator (Exercise 3.4).

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 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, ${}^{W}_{T}T$.

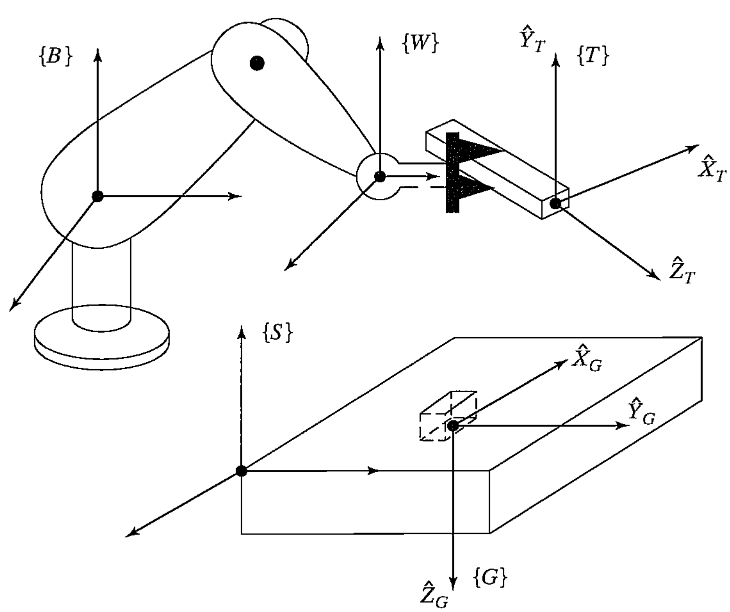


FIGURE 3.31: Determination of the tool frame (Exercise 3.8).

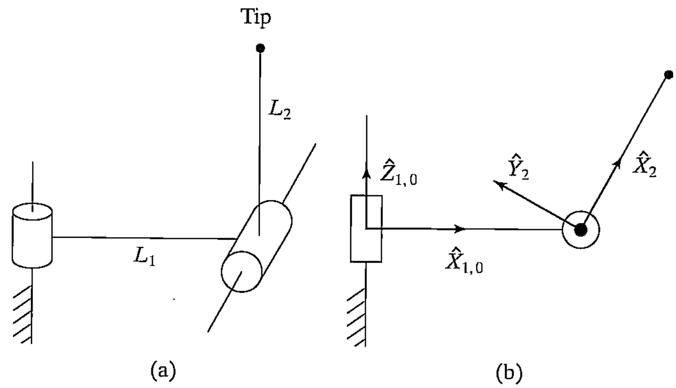


FIGURE 3.32: Two-link arm with frame assignments (Exercise 3.9).

3.9 [11] For the two-link manipulator shown in Fig. 3.32(a), the link-transformation matrices, ${}^0_1T$ and ${}^1_2T$, were constructed. Their product is

$$
{}^0_2T = \begin{bmatrix} c\theta_1 c\theta_2 & -c\theta_1 s\theta_2 & s\theta_1 & l_1 c\theta_1 \\ s\theta_1 c\theta_2 & -s\theta_1 s\theta_2 & -c\theta_1 & l_1 s\theta_1 \\ s\theta_2 & c\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.
$$

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 ${}^0P_{tip}$, 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, α, d, θ) 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 α, where a is the length of the common perpendicular joining the two and α is the angle made by the two axes when projected onto a plane normal to the common perpendicular. Given a line defined as passing 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 α.

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.

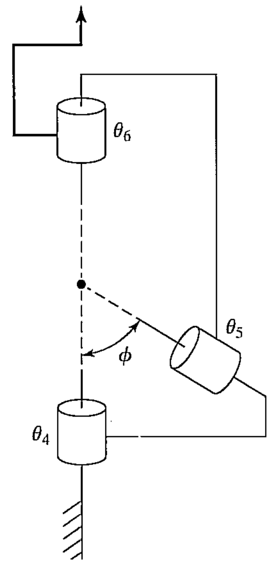


FIGURE 3.33: $3R$ nonorthogonal-axis robot (Exercise 3.11).

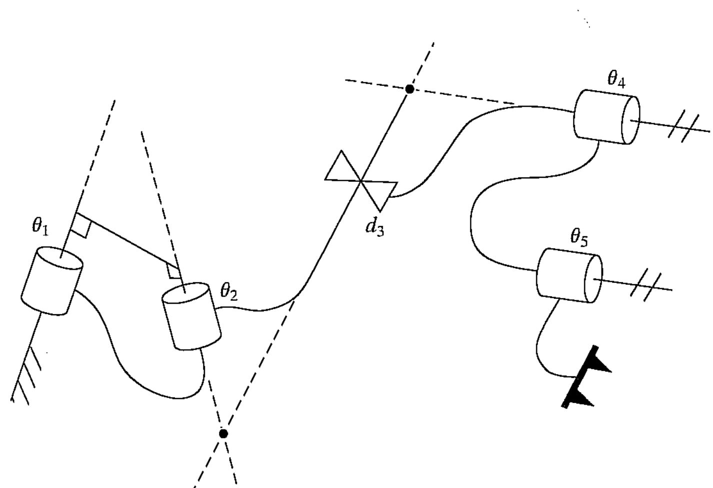


FIGURE 3.34: Schematic of a $2RP2R$ manipulator (Exercise 3.13).

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.
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?

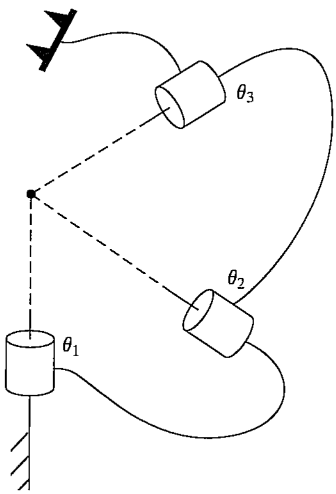


FIGURE 3.35: Schematic of a $3R$ manipulator (Exercise 3.15).

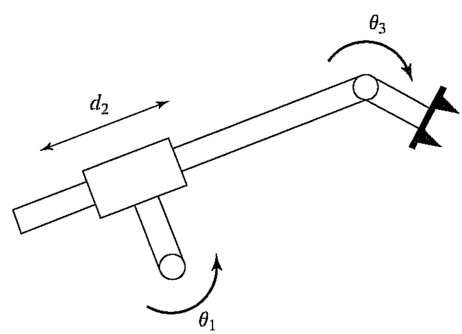


FIGURE 3.36: RPR planar robot (Exercise 3.16).

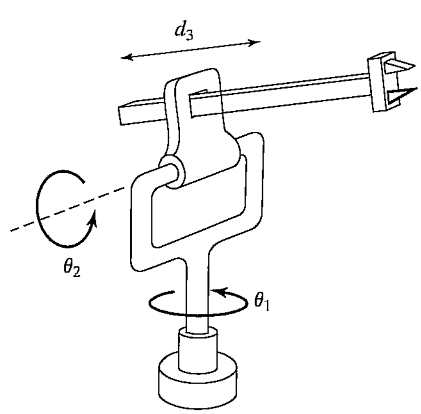


FIGURE 3.37: Three-link RRP manipulator (Exercise 3.17).

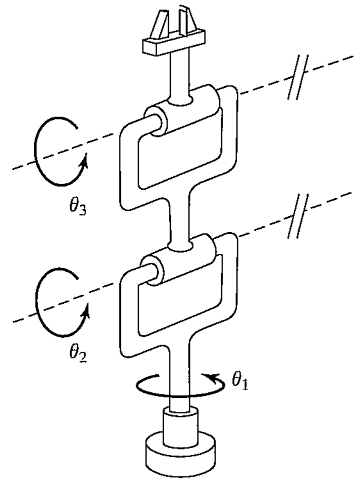


FIGURE 3.38: Three-link RRR manipulator (Exercise 3.18).

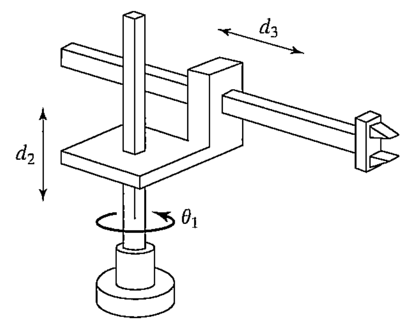


FIGURE 3.39: Three-link RPP manipulator (Exercise 3.19).

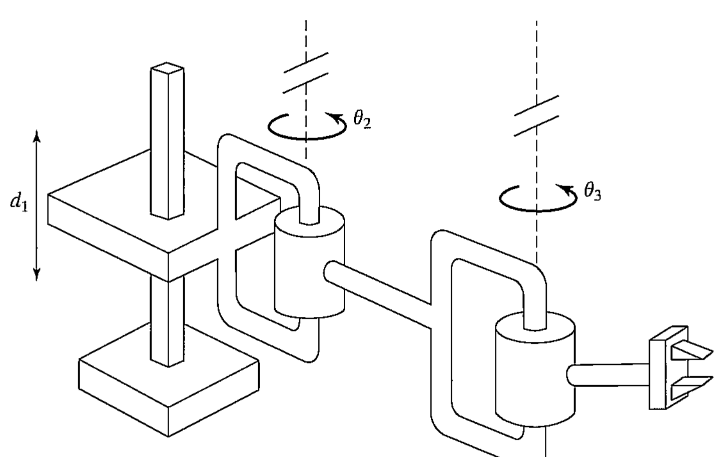


FIGURE 3.40: Three-link PRR manipulator (Exercise 3.20).

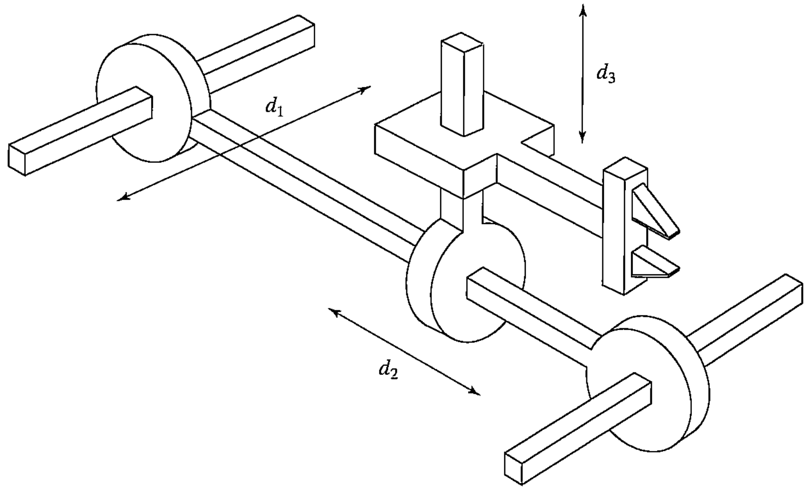


FIGURE 3.41: Three-link PPP manipulator (Exercise 3.21).

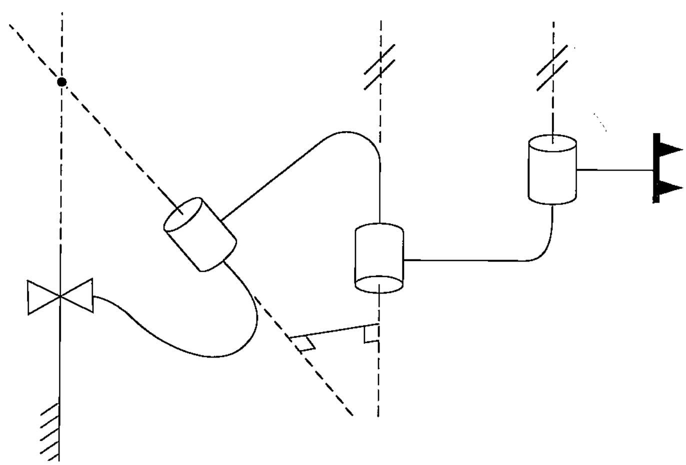


FIGURE 3.42: Schematic of a $P3R$ manipulator (Exercise 3.22).

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 (the wrist frame relative to the base frame) as output. Use the procedure heading (or equivalent in C)

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

where "wrelb" is the wrist frame relative to the base frame, ${}^B_W T$. The type "frame" consists of a 2×2 rotation matrix and a 2×1 position vector. If desired, you may represent the frame with a 3×3 homogeneous transform in which the third row is [0 0 1]. (The manipulator data are $l_1 = l_2 = 0.5$ meters.)

2. Write a routine that calculates where the tool is, relative to the station frame. The input to the routine is a vector of joint angles:

```
Procedure WHERE(VAR theta: vec3; VAR trels: frame);
```

Obviously, WHERE must make use of descriptions of the tool frame and the robot base frame in order to compute the location of the tool relative to the station frame. The values of ${}^{W}_{T}T$ and ${}^{S}_{B}T$ should be stored in global memory (or, as a second choice, you may pass them as arguments in WHERE).

3. A tool frame and a station frame for a certain task are defined as follows by the user:

$$ {}^{W}_{T}T = [x\ y\ \theta] = [0.1\ 0.2\ 30.0], $$

$$ {}^{B}_{S}T = [x\ y\ \theta] = [-0.1\ 0.3\ 0.0]. $$

Calculate the position and orientation of the tool relative to the station frame for the following three configurations (in units of degrees) of the arm:

$$ [\theta_1\ \theta_2\ \theta_3] = [0.0\ 90.0\ \ -90.0], $$

$$ [\theta_1\ \theta_2\ \theta_3] = [-23.6\ \ -30.3\ 48.0], $$

$$ [\theta_1\ \theta_2\ \theta_3] = [130.0\ 40.0\ 12.0]. $$

MATLAB EXERCISE 3

This exercise focuses on DH parameters and on the forward-pose (position and orientation) kinematics transformation for the planar 3-DOF, 3R robot (of Figures 3.6 and 3.7). The following fixed-length parameters are given: $L_1 = 4$, $L_2 = 3$, and $L_3 = 2$ (m).

a) Derive the DH parameters. You can check your results against Figure 3.8.

b) Derive the neighboring homogeneous transformation matrices ${}^{i-1}_{i}T$, $i = 1, 2, 3$. These are functions of the joint-angle variables θ_i, $i = 1, 2, 3$. Also, derive the constant ${}^{3}_{H}T$ by inspection: The origin of $\{H\}$ is in the center of the gripper fingers, and the orientation of $\{H\}$ is always the same as the orientation of $\{3\}$.

c) Use Symbolic MATLAB to derive the forward-pose kinematics solution ${}^{0}_{3}T$ and ${}^{0}_{H}T$ symbolically (as a function of θ_i). Abbreviate your answer, using $s_i = \sin(\theta_i)$, $c_i = \cos(\theta_i)$, and so on. Also, there is a $(\theta_1 + \theta_2 + \theta_3)$ simplification, by using sum-of-angle formulas, that is due to the parallel Z_i axes. Calculate the forward-pose kinematics results (both ${}^{0}_{3}T$ and ${}^{0}_{H}T$) via MATLAB for the following input cases:

i) $\Theta = \{\theta_1\ \ \theta_2\ \ \theta_3\}^T = \{0\ \ 0\ \ 0\}^T$.

ii) $\Theta = \{10^\circ\ \ 20^\circ\ \ 30^\circ\}^T$.

iii) $\Theta = \{90^\circ\ \ 90^\circ\ \ 90^\circ\}^T$.

For all three cases, check your results by sketching the manipulator configuration and deriving the forward-pose kinematics transformation by inspection. (Think of the definition of ${}^{0}_{H}T$ in terms of a rotation matrix and a position vector.) Include frames $\{H\}$, $\{3\}$, and $\{0\}$ in your sketches.

d) Check all your results by means of the Corke MATLAB Robotics Toolbox. Try functions *link*(), *robot*(), and *fkine()*.