2. FORWARD KINEMATICS

2.1 Introduction

In order to control the motion of a robot arm its particular kinematic equations must be known. In this chapter we are interested in determining the set of equations for calculating the position and orientation of the end-effector given values for the "joint variables" (the joint variable for a revolute joint is its angle, while the joint variable for a prismatic joint is its displacement). This is known as the robot's "forward kinematics solution" (also called its "direct kinematics solution"). The other required kinematic equations will be described in chapters 3 and 4.

The right-hand rule is a common method used to help visualize and understand vectors and coordinate systems in 3 dimensions. The right-hand rule determines the orientation of axes and the direction of rotations. The figure below shows the right-hand rule.

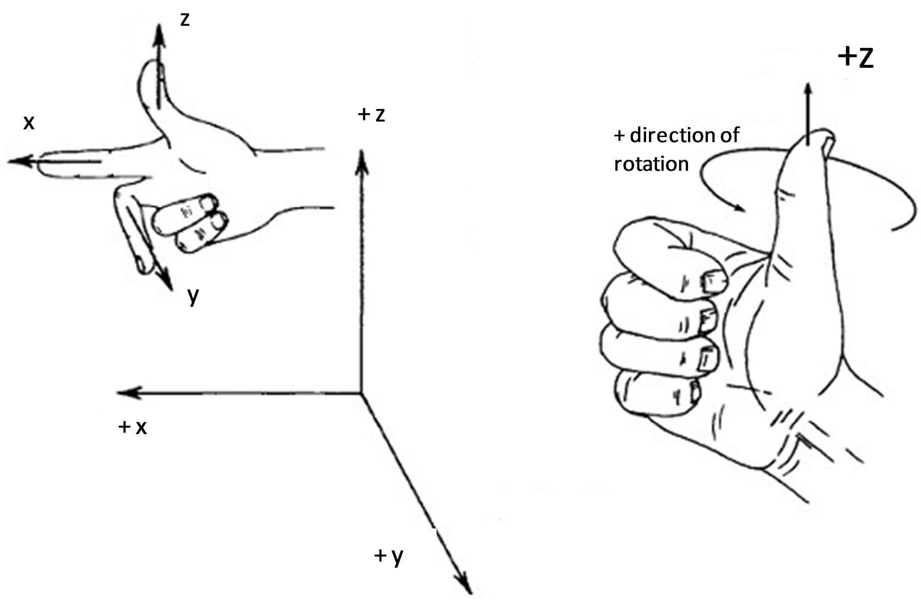


Figure 2.1 Right-hand rule for determining coordinates and determining rotation.

The thumb indicates the direction of the positive z-direction, fingers should point to the corresponding x-axis, and the palm (or fingers bent 90 deg) corresponds to positive y-direction.

For rotations, grasp the rotating axis with right hand with thumb oriented in the positive direction. Fingers will curl in the direction of positive rotation for that axis (for example, if you grasp the z-axis then your fingers should curl from x-axis to y-axis to indicate positive z rotation).

If you grasp the rotation axis and your fingers do not curl in the appropriate direction this is an indication that the rotation is in the negative direction of how you grasped.

2.2 Matrix Representations of Vectors, Frames and Rigid Bodies in 3D Space

Before deriving the forward kinematics solution for a particular robot we must cover some basic matrix representations. A vector in space can be written as follows:

$$\vec{V} = v_x\vec{i} + v_y\vec{j} + v_z\vec{k} \tag{2.1}$$

where $\vec{V}$ is the vector; $\vec{i}$, $\vec{j}$ and $\vec{k}$ are unit vectors in the directions of the X, Y, and Z axes of the fixed reference frame; and v_x, v_y and v_z are the components of the vector in the reference frame. This is illustrated in Figure 2.2.

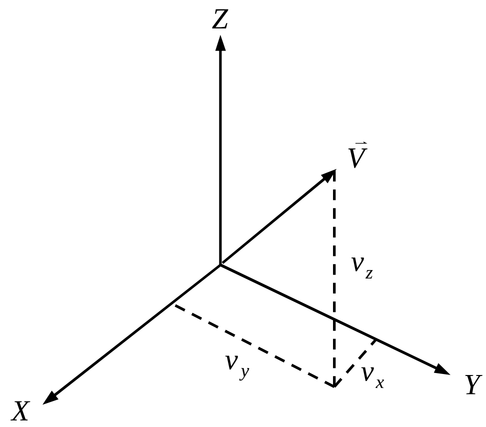


Figure 2.2 A vector in 3D space.

The three components of the vector can also be written in the matrix form:

$$\vec{V} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \tag{2.2}$$

For use in robotics (and other applications such as computer graphics) a fourth value is added to this representation. For example:

$$\vec{V} = \begin{bmatrix} e \\ f \\ g \\ h \end{bmatrix} \tag{2.3}$$

In this course, h will equal either 0 or 1.

To represent a new frame we must define its orientation relative to the reference frame and the position of its origin relative to the origin of the reference frame. We will look at these two problems separately.

To represent a new frame whose origin is coincident with the origin of the reference frame we need to define three unit vectors in the directions of the coordinate axes of the new frame. We will call these unit vectors the normal, $\vec{n}$, orientation, $\vec{o}$, and approach, $\vec{a}$, vectors. (These in fact define the directions of the X, Y and Z axes of the new frame). If we call the new frame {F}, we can write it in the form:

$$F = \begin{bmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{bmatrix} \tag{2.4}$$

An example is shown in Figure 2.3.

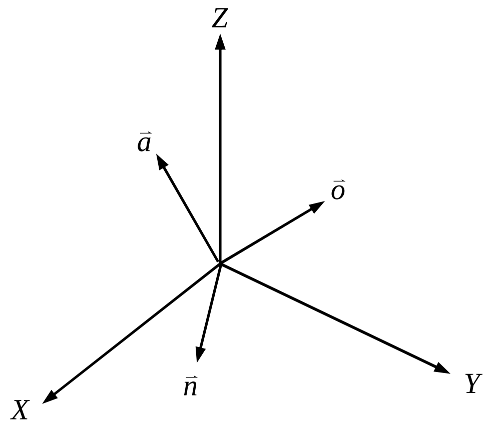


Figure 2.3 New frame with an origin coincident with the reference frame.

Normally the origin of the new frame would not be coincident with the reference frame. For this case we need to define a vector $\vec{P} = [P_x \; P_y \; P_z \; 1]^T$ describing the position of the origin of the new frame relative to the origin of the reference frame. An example is shown in Figure 2.4.

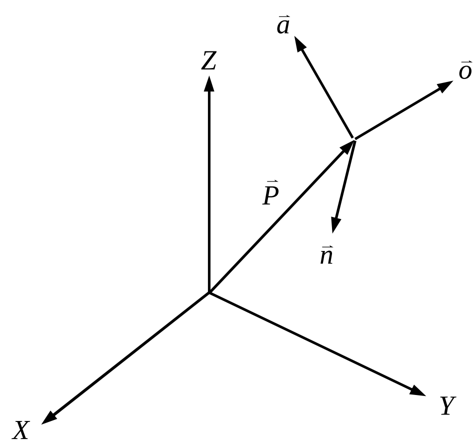


Figure 2.4 General representation of a frame.

We then write the matrix definition of the frame {F} as:

$$F = \begin{bmatrix} n_x & o_x & a_x & P_x \\ n_y & o_y & a_y & P_y \\ n_z & o_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.5}$$

Note that when a vector defines only a direction an *h* value of 0 is used. These are called "directional vectors". When a vector defines a position it is called a "positional vector" and an *h* value of 1 is used.

Example 2.1 (from Niku's book [1]): The origin of the frame {F} is located at $\vec{P} = [3 \;\; 5 \;\; 7 \;\; 1]^T$. Its n-axis is parallel to the X-axis, its o-axis is at 45º relative to the Y-axis, and its a-axis is 45º relative to the Z-axis. This is shown in Figure 2.5. The frame can be described by:

$$F = \begin{bmatrix} 1 & 0 & 0 & 3 \\ 0 & 0.707 & -0.707 & 5 \\ 0 & 0.707 & 0.707 & 7 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.6}$$

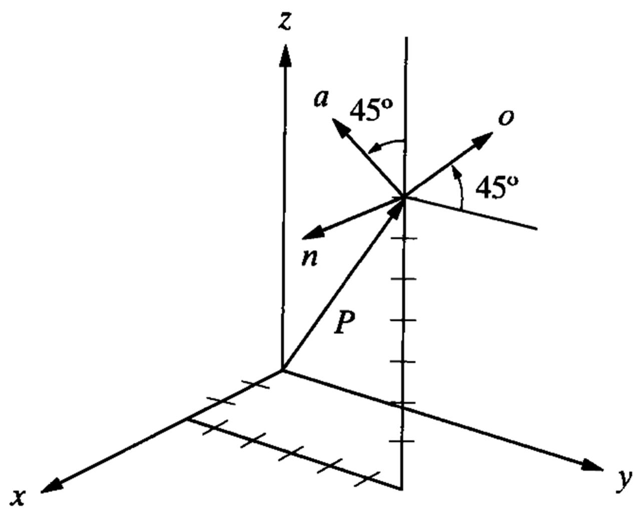


Figure 2.5 Example of a frame [1].

The position and orientation of a rigid body in space can be represented by attaching a frame to it. Since the relative position and orientation of the frame and the body are fixed, specifying the frame in space also specifies the body in space. This idea is illustrated in Figure 2.6. Note that, as discussed in Section 1.3, a rigid body in space has six DOF (three describing its position and three describing its orientation). This means that the nine pieces of information describing the orientation of the frame (*i.e.* the components $\vec{n}$, $\vec{o}$, and $\vec{a}$) are not independent from each other. In fact, the nine values must satisfy the following six constraints:

$$\vec{n} \bullet \vec{o} = 0 \tag{2.7}$$
$$\vec{n} \bullet \vec{a} = 0 \tag{2.8}$$
$$\vec{a} \bullet \vec{o} = 0 \tag{2.9}$$
$$\|\vec{n}\| = 1 \tag{2.10}$$
$$\|\vec{o}\| = 1 \tag{2.11}$$
$$\|\vec{a}\| = 1 \tag{2.12}$$

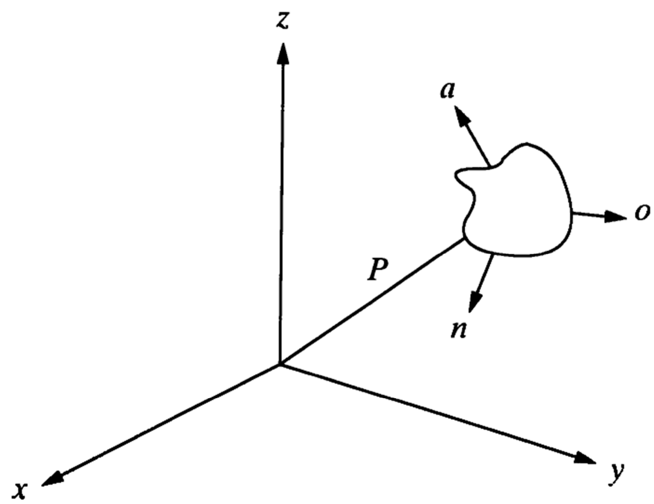


Figure 2.6 Representation of a rigid body in space [1].

Niku's [1] example 2.3 illustrates a method for completing the description of a frame when only partial information is given.

2.3 Homogeneous Transformation Matrices

2.3.1 Pure Translations and Pure Rotations

The 4×4 matrix defined in equation (2.5) is an example of what is termed a "homogeneous transformation matrix". This type of matrix is used to define a "transformation" which here refers to a movement in space. A transformation may consist of: a translation alone; a rotation about an axis; or a combination of one or more translations and rotations.

If the transformation is a translation alone (or a "pure translation") then the transformation matrix is simply:

$$T = \begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.13}$$

where $d_x, d_y,$ and d_z are the three components of the translation along the X, Y and Z axes of the reference frame. This is illustrated in Figure 2.7. We apply this transformation by pre-multiplying the frame by this transformation matrix. For example, to translate frame {F} to obtain another frame {G} we have:

$$G = \begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} n_x & o_x & a_x & P_x \\ n_y & o_y & a_y & P_y \\ n_z & o_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n_x & o_x & a_x & P_x + d_x \\ n_y & o_y & a_y & P_y + d_y \\ n_z & o_z & a_z & P_z + d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.14}$$

Note that only the positional vector for the frame {G} was changed (its $\vec{n}$, $\vec{o}$, and $\vec{a}$ were unaltered). This equation can also be written symbolically as:

$$G = \text{Trans}(d_x, d_y, d_z) * F \tag{2.15}$$

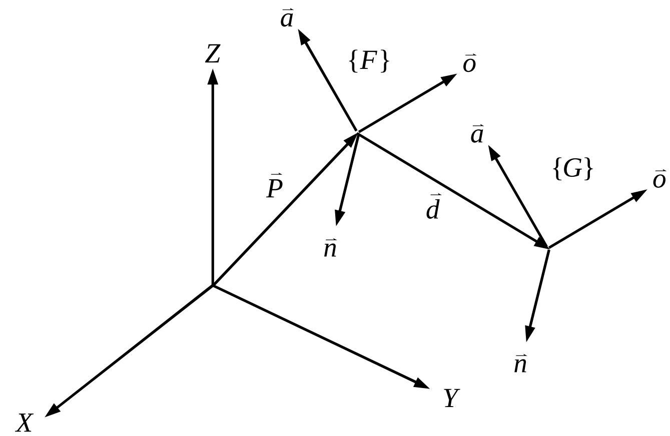


Figure 2.7 A transformation consisting of a pure translation.

Another basic type of transformation is a rotation about one of the axes of the reference frame. We'll start by looking rotating frame {F} about the X-axis by an angle θ to obtain the frame {G}. (As usual the sign of the rotation angle is given by the right-hand rule). Since we are only rotating about X this can be shown using only the YZ plane, as in Figure 2.8.

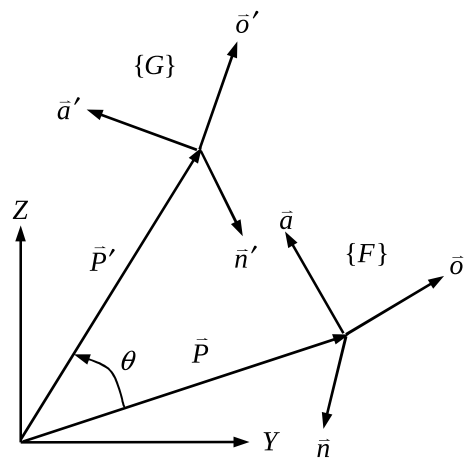


Figure 2.8 A transformation consisting of a pure rotation about the X-axis of the reference frame.

Note that both the directional vectors ($\vec{n}$, $\vec{o}$, and $\vec{a}$) and the positional vector ($\vec{P}$) are changed by this rotation. The transformation matrix representing this rotation is:

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\theta & -S\theta & 0 \\ 0 & S\theta & C\theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.16}$$

where $C\theta$ denotes $\cos\theta$, and $S\theta$ denotes $\sin\theta$. (For details on the derivation of this matrix please see section 2.5.2 of Niku's text [1]). As before we apply the transformation by pre-multiplying by this transformation matrix. For the frame {G} we now have:

$$G=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & C\theta & -S\theta & 0\\ 0 & S\theta & C\theta & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}n_x & o_x & a_x & P_x\\ n_y & o_y & a_y & P_y\\ n_z & o_z & a_z & P_z\\ 0 & 0 & 0 & 1\end{bmatrix}=\begin{bmatrix}n_x & o_x & a_x & P_x\\ n_yC\theta-n_zS\theta & o_yC\theta-o_zS\theta & a_yC\theta-a_zS\theta & P_yC\theta-P_zS\theta\\ n_yS\theta+n_zC\theta & o_yS\theta+o_zC\theta & a_yS\theta+a_zC\theta & P_yS\theta+P_zC\theta\\ 0 & 0 & 0 & 1\end{bmatrix}=\begin{bmatrix}n'_x & o'_x & a'_x & P'_x\\ n'_y & o'_y & a'_y & P'_y\\ n'_z & o'_z & a'_z & P'_z\\ 0 & 0 & 0 & 1\end{bmatrix} \quad (2.17)$$

As you would expect the X components are not affected by this rotation. This can be also be written in the symbolic form:

$$G = \text{Rot}(X,\theta) * F \quad (2.18)$$

For rotations about the Y and Z axes the matrices are:

$$\text{Rot}(Y,\theta)=\begin{bmatrix}C\theta & 0 & S\theta & 0\\ 0 & 1 & 0 & 0\\ -S\theta & 0 & C\theta & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \quad (2.19)$$

and

$$\text{Rot}(Z,\theta)=\begin{bmatrix}C\theta & -S\theta & 0 & 0\\ S\theta & C\theta & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}. \quad (2.20)$$

2.3.2 Combined Transformations and Order of Multiplications

To obtain the forward kinematics equations we must be able to combine the series of transformations resulting from the links and joints of the robot. Combined transformations are also used to program robots in some situations. The translations/rotations can be along/about the axes of the initial reference frame or along/about the axes of the current frame.

To make a transformation relative to the reference frame we pre-multiply by the transformation matrix.

To make a transformation relative to the current frame we post-multiply by the transformation matrix.

Example 2.2:

We want the following series of transformations relative to the reference frame:

1) Rotate about the Y-axis by 90°.
2) Translate by $[1 \;\; 3 \;\; -2]^T$.
3) Rotate about the Z-axis by –90°.

We can start by using a 4 x 4 identity matrix as a place keeper. So we have:

$$T = I \quad (2.21)$$

After step 1 the result is:

$$T = \text{Rot}(Y, 90°) * I \quad (2.22)$$

After step 2 the result is:

$$T = \text{Trans}(1,3,-2) * \text{Rot}(Y, 90°) * I \quad (2.23)$$

After step 3 the equation for the desired combined transformation is:

$$T = \text{Rot}(Z,-90°) * \text{Trans}(1,3,-2) * \text{Rot}(Y,90°) * I \tag{2.24}$$

Note that as a result of pre-multiplying the matrices they are written in an order opposite to the order of the transformations themselves. The matrix resulting from equation (2.24) equals:

$$T = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & -2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * I = \begin{bmatrix} 0 & 1 & 0 & 3 \\ 0 & 0 & -1 & -1 \\ -1 & 0 & 0 & -2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.25}$$

End of example.

We'll now define some standard notation. In robotics the reference frame is often called the "world frame" which we will denote by {W}. If a given T matrix defines the transformation of a new frame {A} relative to {W} then the equation relating {A} and {W} is written:

$$A = {}^{W}T_{A} W \tag{2.26}$$

Example 2.3:

The current frame is {A} and we'll denote the axes of {A} as X_A, Y_A and Z_A. We now want the following series of transformations:

1) Rotate about Y_A by 90°.
2) Translate by $[1\ \ 3\ \ -2]^T$ relative to {A}.
3) Rotate about Z_A by –90°.

Again we may start with:

$$T = I \tag{2.28}$$

After step 1 we have:

$$T = I * \text{Rot}(Y_A, 90°) \tag{2.29}$$

After step 2 we have:

$$T = I * \text{Rot}(Y_A, 90°) * \text{Trans}(1,3,-2) \tag{2.30}$$

After step 3 the equation for the desired combined transformation is:

$$T = I * \text{Rot}(Y_A, 90°) * \text{Trans}(1,3,-2) * \text{Rot}(Z_A, -90°) \tag{2.31}$$

Note that the order of the matrices is now opposite to that of equation (2.24). The matrix resulting from equation (2.31) equals:

$$T = I * \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & -2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & -2 \\ -1 & 0 & 0 & 3 \\ 0 & -1 & 0 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.32}$$

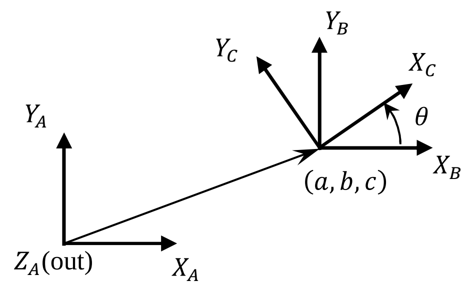
End of example.

Example 2.4.1

(a) Find the symbolic transformation matrix equation for: translation by $[a\ \ b\ \ c]^T$ relative to the reference frame, followed by a rotation of θ about Z of the current frame.
(b) What is the value of the final matrix if $[a\ \ b\ \ c] = [6\ \ 2\ \ 0.1]$ and $\theta = 40^o$?

Answer

(a) Let's call the reference frame "A". We must apply the translation first:



$$^AT_B = Trans(a,b,c) \cdot I$$

Frame B is the current frame. Next, to rotate about the Z axis of frame B we postmultiply as follows:

$$^AT_C = Trans(a,b,c) \cdot I \cdot Rot(Z,\theta)$$

Now the current frame is C. The reference frame is still frame A.

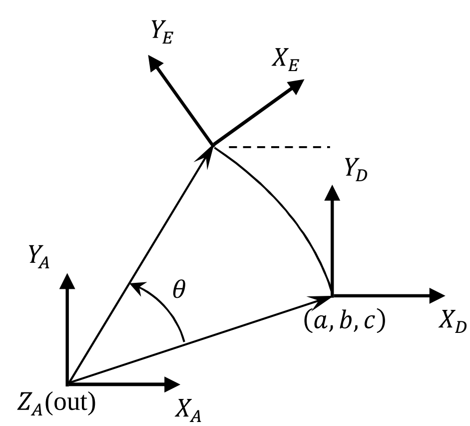
(b) $$^AT_C = \begin{bmatrix} 0.77 & -0.64 & 0 & 6 \\ 0.64 & 0.77 & 0 & 2 \\ 0 & 0 & 1 & 0.1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Example 2.4.2

(a) Find the symbolic transformation equation for: translation by $[a\ \ b\ \ c]^T$ relative to the reference frame, followed by a rotation of θ about Z of the reference frame.
(b) What is the value of the final matrix if $[a\ \ b\ \ c] = [6\ \ 2\ \ 0.1]$ and $\theta = 40^o$?

Answer

(a) Let's call the reference frame "A". We must apply the translation first:



$$^AT_D = Trans(a,b,c) \cdot I$$

Frame D is the current frame. To rotate about the Z axis of frame A we premultiply as follows:

$$^AT_E = Rot(Z,\theta) \cdot Trans(a,b,c) \cdot I$$

Now the current frame is E. The reference frame is still frame A.

(b) $$^AT_E = \begin{bmatrix} 0.77 & -0.64 & 0 & 3.3 \\ 0.64 & 0.77 & 0 & 5.4 \\ 0 & 0 & 1 & 0.1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Note how frame C (from the previous example) and frame E are not identical, as shown by both the figures and matrices (check the 4th column).

Example 2.4.3:

We are interested in the combined transformation resulting from a rotation of 180° about the Z-axis of the reference frame, followed by a translation of -4 units along the Y-axis of the current frame, followed by rotation of -90° about the X-axis of the current frame.

Again, we'll begin with:

$$T = I \tag{2.33}$$

Since the first rotation is relative to the reference frame we should pre-multiply by the matrix, giving:

$$T = \text{Rot}(Z, 180°) * I \tag{2.34}$$

Since the translation of -4 units is relative to this current frame we should post-multiply. This gives:

$$T = \text{Rot}(Z, 180°) * I * \text{Trans}(0, -4, 0) \tag{2.35}$$

Since the rotation of -90° is relative to this current frame we should post-multiply. This gives the equation for the desired combined transformation matrix:

$$T = \text{Rot}(Z, 180°) * I * \text{Trans}(0, -4, 0) * \text{Rot}(X, -90°) \tag{2.36}$$

Calculating the actual matrix is left as an exercise. Note that we would have obtained the same answer if the first rotation was relative to the current frame since $\text{Rot}(Z, 180°) * I = I * \text{Rot}(Z, 180°)$.

End of example.

Example 2.5:

In Figure 2.9 a robot and a part are illustrated. A frame {P} is attached to the part, and the world frame is attached to the base of the robot. We wish to be able to move the robot such that frame {E} at its end-effector touches the part at the origin of {P} with its axes aligned with {P}. In other words, we want to command the robot to move until:

$${}^{W}T_{E} = {}^{W}T_{P} \tag{2.37}$$

To determine ${}^{W}T_{P}$ the position and orientation of the part must be known. In practice, tooling known as a fixture may be used to fix the part's position and orientation, or this information could be measured using a sensor such as a vision system. For this example, let's say the origin of {P} is located at $[0.1 \;\; 1.2 \;\; 0.2]^T$ in world coordinates. Comparing {P} and {W} in Figure 2.9 it can be observed that {P} has been rotated by –90° about X_W, and then translated. In equation form, the

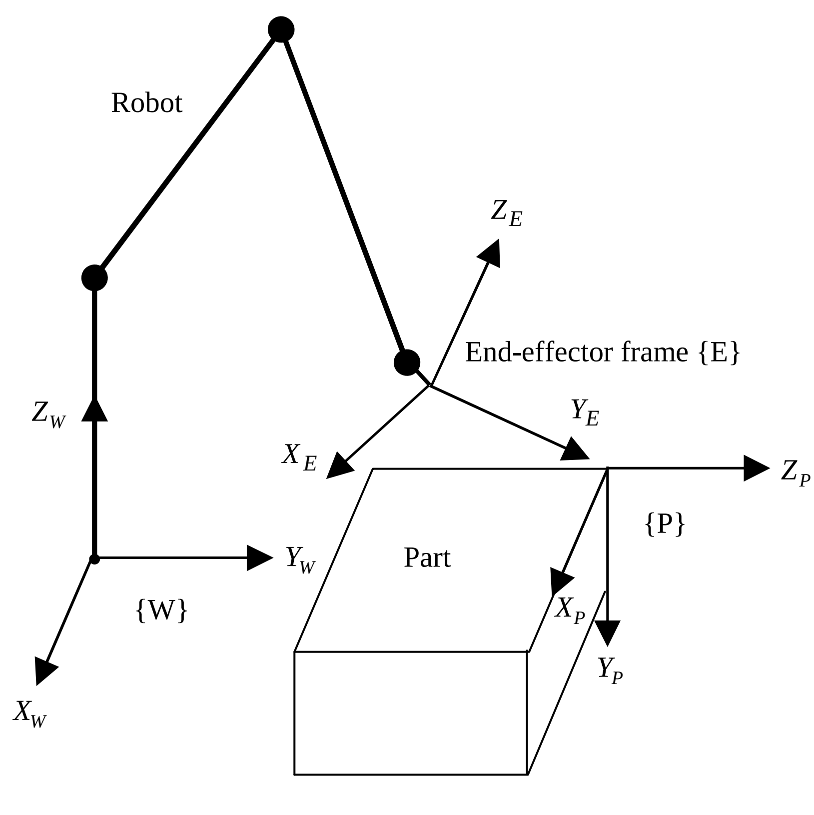


Figure 2.9 Transformation example.

combined transformation is:

$$
\begin{aligned}
{}^{W}T_{P} &= \mathrm{Trans}(0.1, 1.2, 0.2) * \mathrm{Rot}(X_W, -90°) \\
&= \begin{bmatrix} 1 & 0 & 0 & 0.1 \\ 0 & 1 & 0 & 1.2 \\ 0 & 0 & 1 & 0.2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0.1 \\ 0 & 0 & 1 & 1.2 \\ 0 & -1 & 0 & 0.2 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned} \tag{2.38}
$$

Note that we pre-multiplied by the translation in equation (2.38) since it is relative to the axes of the reference (world) frame.

2.3.3 Inverse of a Transformation Matrix

The ability to invert a transformation matrix is useful in solving certain robotics problems. Luckily a transformation matrix has special properties that make inverting it relatively simple. (These are described further in section 2.6 of Niku's text). For the homogeneous transformation matrix

$$
T = \begin{bmatrix} n_x & o_x & a_x & P_x \\ n_y & o_y & a_y & P_y \\ n_z & o_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.40}
$$

its inverse is:

$$
T^{-1} = \begin{bmatrix} n_x & n_y & n_z & -\vec{P} \bullet \vec{n} \\ o_x & o_y & o_z & -\vec{P} \bullet \vec{o} \\ a_x & a_y & a_z & -\vec{P} \bullet \vec{a} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.41}
$$

Comparing (2.40) and (2.41), the inverse is obtained by transposing the 3×3 rotation portion of the matrix and then calculating the dot products of the rows with $-\vec{P}$. Also note that:

$$
{}^{A}T_{B} = {}^{B}T_{A}^{-1} \tag{2.42}
$$

Example 2.6:

Several coordinate frames are illustrated in Figure 2.10. The arrows between the origins indicate the way the transformations have been defined. For example frame {A} has been defined relative to frame {U}. It may be observed that there are two ways to describe {D}, that is:

$$ {}^{U}T_{D} = {}^{U}T_{A} * {}^{A}T_{D} \tag{2.43}$$

and

$$ {}^{U}T_{D} = {}^{U}T_{B} * {}^{B}T_{C} * {}^{C}T_{D} \tag{2.44}$$

We can combine these descriptions to form the transform equation:

$$ {}^{U}T_{A} * {}^{A}T_{D} = {}^{U}T_{B} * {}^{B}T_{C} * {}^{C}T_{D} \tag{2.45}$$

With *n* such transform equations we can solve for *n* unknown transformations. This requires use of inverse transformations. For example if we wish to solve for ${}^{A}T_{D}$ then we must pre-multiply both sides of (2.43) to eliminate ${}^{U}T_{A}$ from the left side as follows:

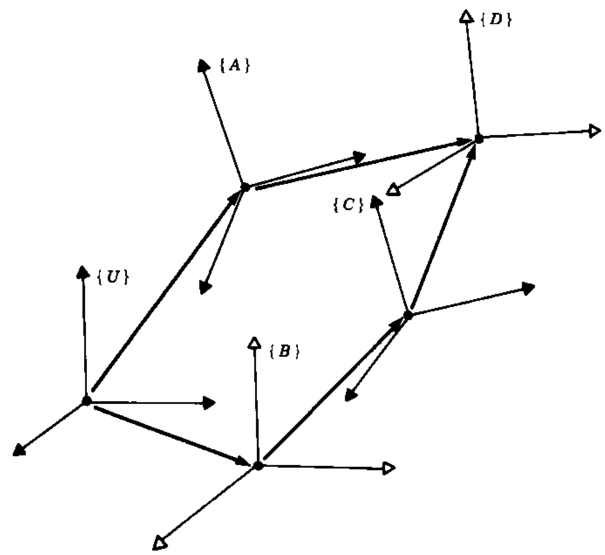


Figure 2.10 Inverse transformation example [2].

$$ {}^{U}T_{A}^{-1} * {}^{U}T_{A} * {}^{A}T_{D} = {}^{U}T_{A}^{-1} * {}^{U}T_{B} * {}^{B}T_{C} * {}^{C}T_{D} \tag{2.46}$$

$$ {}^{A}T_{D} = {}^{U}T_{A}^{-1} * {}^{U}T_{B} * {}^{B}T_{C} * {}^{C}T_{D} \tag{2.47}$$

Similarly, if we wish to solve for ${}^{B}T_{C}$ then we must pre- and post-multiply in the following way:

$$ {}^{U}T_{B}^{-1} * {}^{U}T_{A} * {}^{A}T_{D} * {}^{C}T_{D}^{-1} = {}^{U}T_{B}^{-1} * {}^{U}T_{B} * {}^{B}T_{C} * {}^{C}T_{D} * {}^{C}T_{D}^{-1} \tag{2.48}$$

$$ {}^{B}T_{C} = {}^{U}T_{B}^{-1} * {}^{U}T_{A} * {}^{A}T_{D} * {}^{C}T_{D}^{-1} \tag{2.49}$$

Example 2.7:

We wish to use a robot to pickup a bolt from a table. The robot is equipped with a gripper as an end-effector for this purpose. A frame {B} is attached to the base of the robot. Frame {T} is attached to the tip of the gripper's fingers. Frame {G} is attached to the bolt and frame {S} to the table. For programming the robot, we wish to solve for the transformation of the bolt relative to the gripper.

The transform equation for this situation is:

$$ {}^{B}T_{T} * {}^{T}T_{G} = {}^{B}T_{S} * {}^{S}T_{G} \tag{2.50}$$

So the desired solution is:

$$ {}^{B}T_{T}^{-1} * {}^{B}T_{T} * {}^{T}T_{G} = {}^{B}T_{T}^{-1} * {}^{B}T_{S} * {}^{S}T_{G} \tag{2.51}$$

$$ {}^{T}T_{G} = {}^{B}T_{T}^{-1} * {}^{B}T_{S} * {}^{S}T_{G} \tag{2.52}$$

For other examples please see section 2.6 of Niku's text [1].

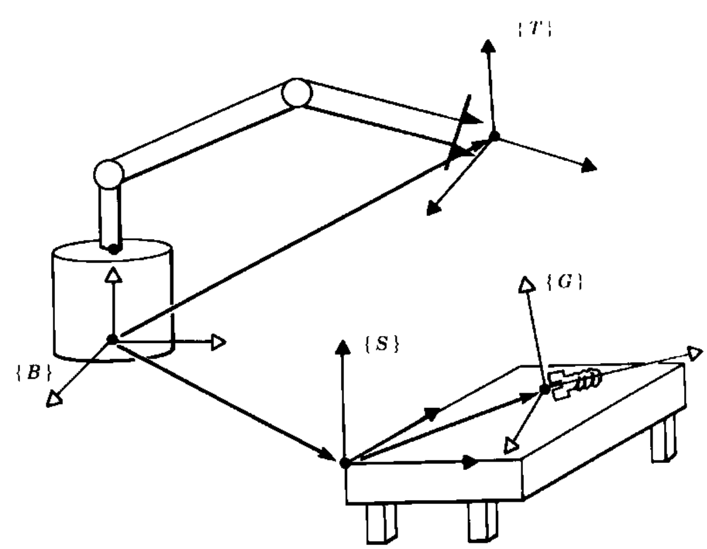


Figure 2.11 Another inverse transformation example [2].

2.4 Denavit-Hartenburg Notation

The Denavit-Hartenburg (D-H) method has been used to model serial robots and derive their equations of motion since the 1960s. The D-H parameters are a concise way to describe a robot's configuration and their use is standard in the robotics field. With the D-H method frames are attached to the each of the robot's links in a systematic way. The robot is modelled as a series of links connected by revolute or prismatic joints. The joints may be in any plane. The links may be of any length (including zero), may be twisted or bent, and may be in any plane [1]. The idea is illustrated for two links of a robot in Figure 2.12. Link *n* is connected to link *n*+1 by joint *n*+1. Note that, as shown, each joint can produce rotation or translation. The task of modelling a robot using the D-H convention begins with assigning the frames. The steps of the "frame assignment procedure" are:

1) Represent each joint with a Z-axis. If joint *n* is revolute the rotation angle is also the joint variable. This is denoted θ_n and the right-hand rule is used to determine the direction of the Z-axis given the positive direction for θ_n. If the joint is prismatic the direction of the Z-axis equals the positive direction of motion for the joint. For a prismatic joint *n* the displacement d_n is the joint variable.
2) Choose the origin of frame 0 and the direction of axis X_0. The direction of axis Y_0 is given by the right-hand rule. Frame 0 is fixed and is known as the "base frame". Steps 3-7 apply to the remaining frames.
3) If the two joint Z-axes are not parallel or intersecting (known as "skew lines") then there will always be a unique mutually perpendicular line called the common normal. Its length will be the shortest distance between the two skew lines. The common normal between Z_{n-1} and Z_n is denoted a_n (please see Figure 2.12). We assign axis X_n to be along a_n.
4) If the two joint Z-axes are parallel then an infinite number of common normals exist. We select the common normal that is collinear with the common normal of the previous joint. We assign axis X_n to be along a_n as before.
5) If the two joint Z-axes are intersecting then no common normal exists. We assign axis X_n to be normal to the plane formed by axis Z_{n-1} and axis Z_n.
6) With Z and X axes assigned, the Y axes are given by the right-hand rule.
7) After completing these steps for all joints of the robot, attach a frame to the end-effector.

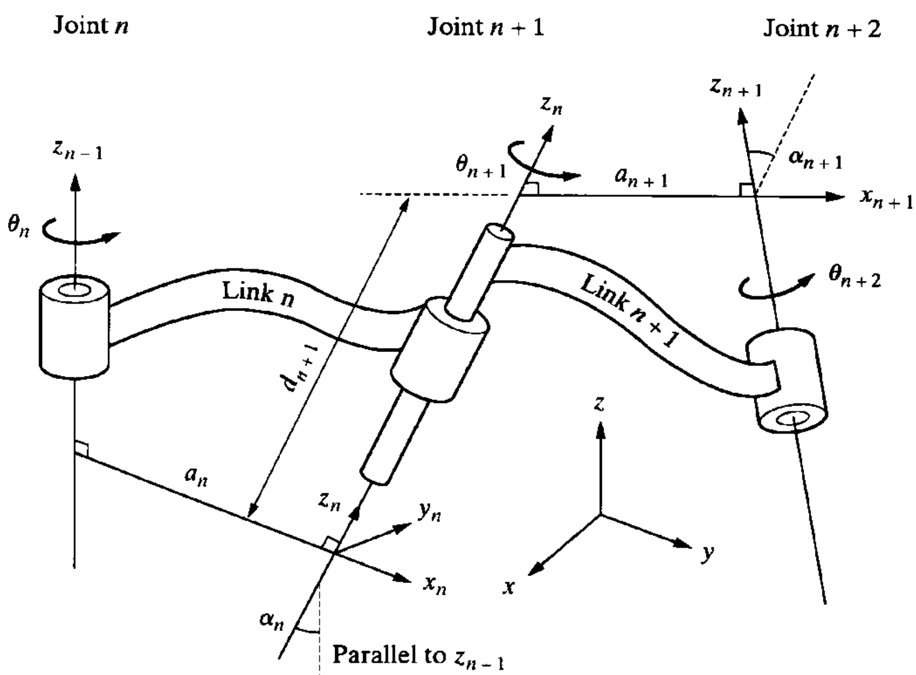


Figure 2.12 Frame assignment and D-H parameters for a pair of links [1].

Next we must determine the values of the D-H parameters: θ, d, a, and α. These values can either be constants or joint variables. The procedure for finding the D-H parameters is as follows:

1) First, imagine rotating frame n about the Z_n axis until X_n and X_{n+1} are parallel. If joint n+1 is a prismatic joint then θ_{n+1} equals the required signed rotation. If joint n+1 is a revolute joint then θ_{n+1} is a joint variable.
2) Second, imagine translating frame n along the Z_n axis until X_n and X_{n+1} are collinear. If joint n+1 is a prismatic joint then d_{n+1} is a joint variable. If joint n+1 is a revolute joint set d_{n+1} equal to the signed distance moved.
3) Third, imagine translating frame n along the X_n axis until the origins of frame n and frame n+1 are coincident. Set a_{n+1} equal to the signed distance moved.
4) Fourth, imagine rotating frame n about the X_n axis until Z_n and Z_{n+1} are collinear, and frames n and n+1 become equivalent. Set α_{n+1} equal to the signed rotation required.
5) Complete these steps for all of the frames.

With the frames assigned and the D-H parameters known the D-H representation of the robot is complete. The remaining issue is how to obtain the forward kinematics solution from the D-H representation. We are looking for the position and orientation of the end-effector given values for the joint variables. This is equivalent to finding the transformation of the frame at the last link relative to the base frame. We can find this transformation by transforming from frame 0 to frame 1, from frame 1 to frame 2, and so on. The transformation of frame n+1 relative to frame n is a product of the four transformations we imagined in steps 1-4 above. In equation form:

$$\begin{aligned}
{}^{n}T_{n+1} &= A_{n+1} \\
&= \mathrm{Rot}(Z,\theta_{n+1}) * \mathrm{Trans}(0,0,d_{n+1}) * \mathrm{Trans}(a_{n+1},0,0) * \mathrm{Rot}(X,\alpha_{n+1}) \\
&= \begin{bmatrix} C\theta_{n+1} & -S\theta_{n+1} & 0 & 0 \\ S\theta_{n+1} & C\theta_{n+1} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_{n+1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_{n+1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha_{n+1} & -S\alpha_{n+1} & 0 \\ 0 & S\alpha_{n+1} & C\alpha_{n+1} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} C\theta_{n+1} & -S\theta_{n+1}C\alpha_{n+1} & S\theta_{n+1}S\alpha_{n+1} & a_{n+1}C\theta_{n+1} \\ S\theta_{n+1} & C\theta_{n+1}C\alpha_{n+1} & -C\theta_{n+1}S\alpha_{n+1} & a_{n+1}S\theta_{n+1} \\ 0 & S\alpha_{n+1} & C\alpha_{n+1} & d_{n+1} \\ 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned} \tag{2.53}$$

Note that the notation A_{n+1} is often used in place of ${}^{n}T_{n+1}$ in robot analysis. Also note that the matrices have been post-multiplied because the transformations are relative to the current frame. For a six DOF robot the total transformation representing the forward kinematics equals:

$${}^{0}T_{6} = {}^{0}T_{1} * {}^{1}T_{2} * {}^{2}T_{3} * {}^{3}T_{4} * {}^{4}T_{5} * {}^{5}T_{6} = A_1 * A_2 * A_3 * A_4 * A_5 * A_6 \tag{2.54}$$

Example 2.8

The kinematic diagram for a two DOF planar robot is shown in Figure 2.13. The positive directions of the joints are indicated by the arrows. Applying the frame assignment procedure produces the frames shown in Figure 2.14. The D-H parameters are found using our five step procedure with n = 0, 1. The parameters are given in the table below:

n+1	θ	d	a	α
1	θ_1	0	a_1	0
2	θ_2	0	a_2	0

The joint variables are θ_1 and θ_2. Substituting the D-H parameters into equation (2.42) gives the A matrices:

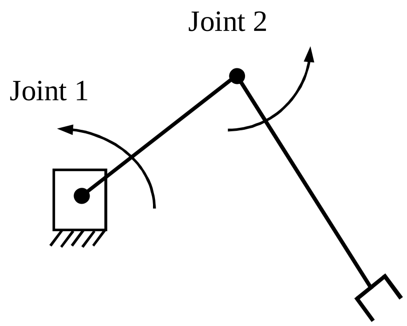


Figure 2.13 Planar 2R manipulator.

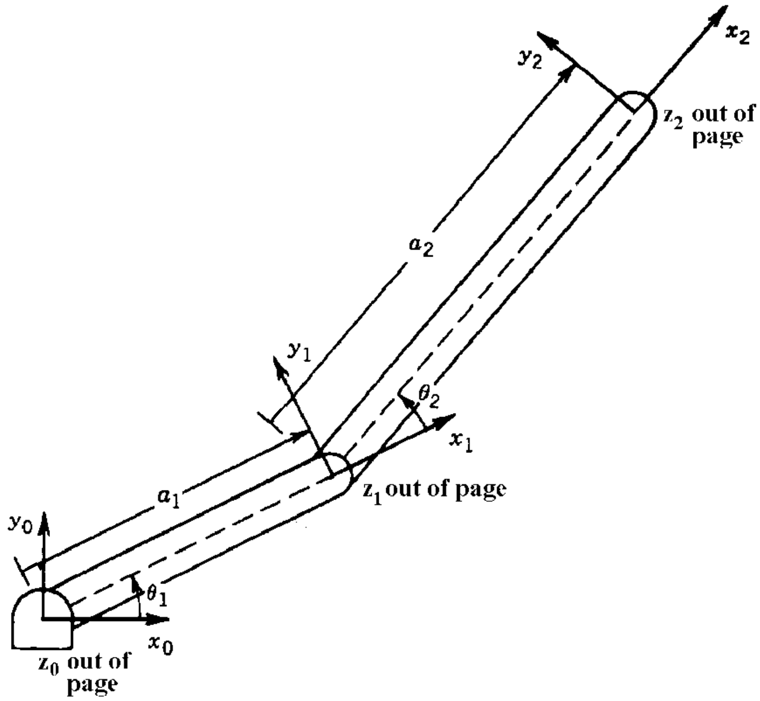


Figure 2.14 The D-H frames for the planar 2R manipulator [4].

$$A_1 = \begin{bmatrix} C\theta_1 & -S\theta_1 & 0 & a_1 C\theta_1 \\ S\theta_1 & C\theta_1 & 0 & a_1 S\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ and } A_2 = \begin{bmatrix} C\theta_2 & -S\theta_2 & 0 & a_2 C\theta_2 \\ S\theta_2 & C\theta_2 & 0 & a_2 S\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.55)$$

The forward kinematics solution is

$$^0T_2 = A_1 * A_2 = \begin{bmatrix} C\theta_{12} & -S\theta_{12} & 0 & a_1 C\theta_1 + a_2 C\theta_{12} \\ S\theta_{12} & C\theta_{12} & 0 & a_1 S\theta_1 + a_2 S\theta_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.56)$$

where the trigonometric identities

$$S\theta_1 C\theta_2 + C\theta_1 S\theta_2 = S(\theta_1 + \theta_2) = S\theta_{12} \text{ and} \quad (2.57)$$

$$C\theta_1 C\theta_2 - S\theta_1 S\theta_2 = C(\theta_1 + \theta_2) = C\theta_{12} \quad (2.58)$$

have been used to simplify the result. Equation (2.58) may be confirmed using basic trigonometry.

2.4.1 Forward Kinematics of Planar Two DOF Robots

Because visualising spatial robots can be challenging we will start with something simpler. The simplest robot is a planar two DOF robot. Each robot has two joint variables and only the X-Y position of its end-effector is controllable. Since the joints can only be either revolute (R) or prismatic (P) the number of possible joint combinations equals $2^2=4$, namely RR, RP, PR and PP. The kinematic diagrams for these planar robots are shown in Figures 2.15-2.18. Note that the joints are bi-directional. The arrows indicate the positive directions for each joint.

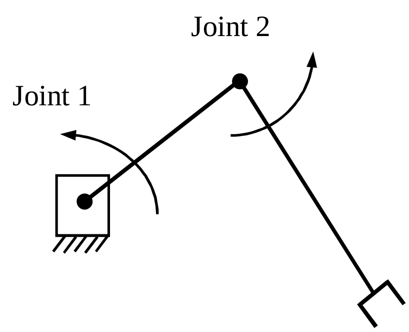


Figure 2.15 Planar RR manipulator.

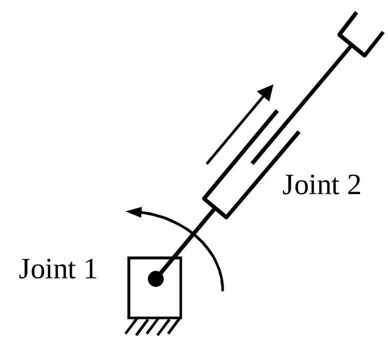


Figure 2.16 Planar RP manipulator.

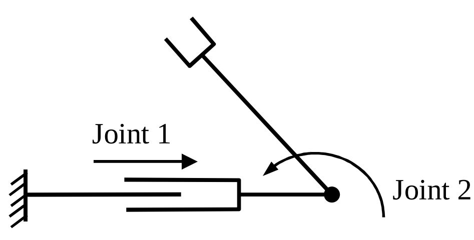


Figure 2.17 Planar PR manipulator.

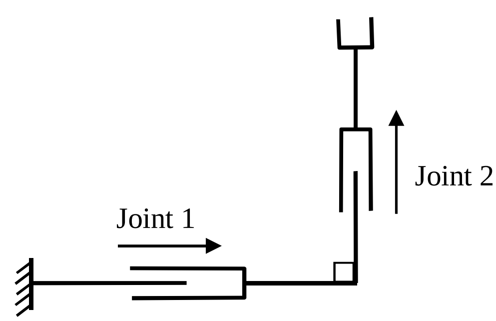


Figure 2.18 Planar PP manipulator.

The only other possible variation with these robots is the presence of a bend or an offset in one or both links. An example of a planar RR robot with an offset is shown in Figure 2.19.

The forward kinematics solutions for some of these robots will be completed as in-class exercises.

Note: The angle between the joints of the planar PP robot does not have to equal 90°. However, what angles cannot be used?

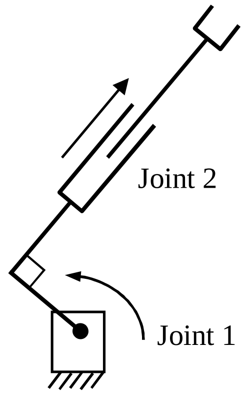


Figure 2.19 Planar RP manipulator with an offset.

2.4.2 Forward Kinematics of Planar Three DOF Robots

The next simplest type of robot is a planar three DOF robot. Each robot has three joint variables; and the X-Y position and the orientation angle of its end-effector are controllable. The number of possible joint combinations equals 2^3=8. These are: RRR, RRP, RPR, RPP, PRR, PRP, PPR, and PPP. However the PPP combination is an invalid choice since we cannot control the orientation of the end-effector. The kinematic diagrams for the remaining seven planar three DOF robots are shown in Figures 2.20-2.26. Note that it is possible for the links to contain bends or offsets as with the two DOF robots.

The forward kinematics solutions for some of these robots will be completed as in-class exercises.

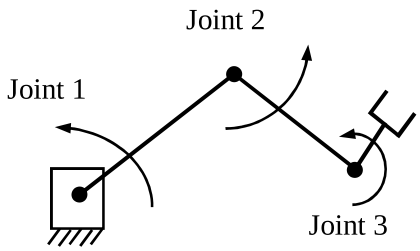


Figure 2.20 Planar RRR manipulator.

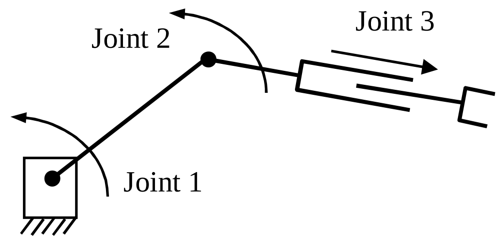


Figure 2.21 Planar RRP manipulator.

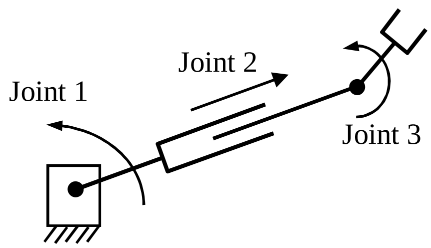


Figure 2.22 Planar RPR manipulator.

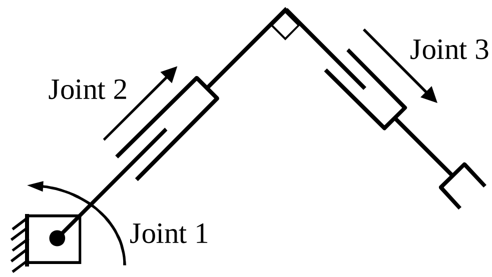


Figure 2.23 Planar RPP manipulator.

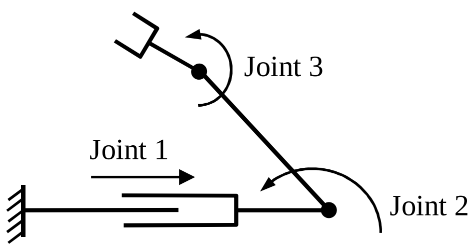


Figure 2.24 Planar PRR manipulator.

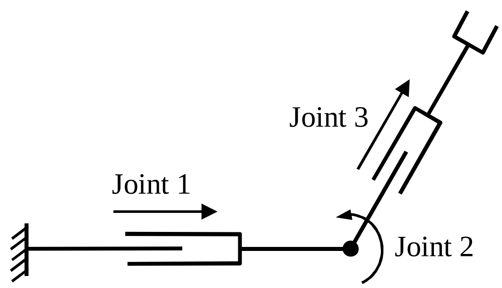


Figure 2.25 Planar PRP manipulator.

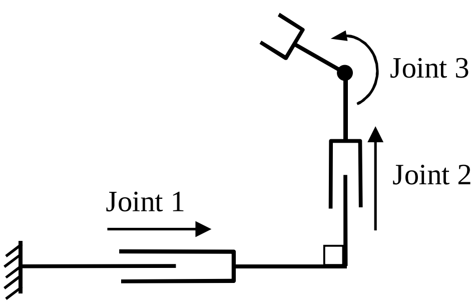


Figure 2.26 Planar PPR manipulator.

2.4.3 Forward Kinematics for Some Spatial Robots

Example 2.9

A three DOF Cartesian robot is shown in Figure 2.27. The positive directions of the joints are indicated by the arrows. Applying the frame assignment procedure produces the frames shown in Figure 2.28. Note that we have chosen to put the Z-axis for the tool frame in the direction of tool approach, and the X-axis of the tool frame in the direction of tool opening/closing.

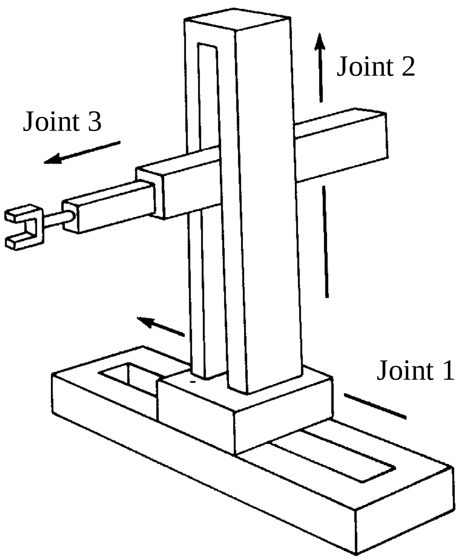


Figure 2.27 Three DOF Cartesian robot [3].

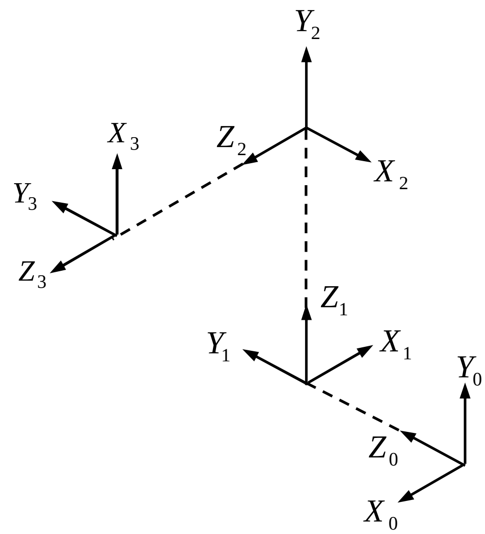


Figure 2.28 The D-H frames for the three DOF Cartesian robot.

The D-H parameters are found using our five step procedure with $n = 0, 1, 2$. The resulting parameters are given in the table below:

n+1	θ	d	a	α
1	180°	d_1	0	90°
2	-90°	d_2	0	90°
3	90°	d_3	0	0

Note that the joint variables are d_1, d_2 and d_3. Substituting the D-H parameters into equation (2.53) gives the A matrices:

$$A_1 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \; A_2 = \begin{bmatrix} 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ and } A_3 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.59}$$

The forward kinematics solution is:

$$^0T_3 = A_1 * A_2 * A_3 = \begin{bmatrix} 0 & 0 & 1 & d_3 \\ 1 & 0 & 0 & d_2 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.60}$$

It is useful to examine the meaning of this result. Recalling equation (2.5), the first three elements of the last column forms the position vector $[P_x \ P_y \ P_z]^T$ so we have $P_x = d_3$, $P_y = d_2$ and $P_z = d_1$ which agrees with Figures 2.27 and 2.28. The directional vectors are $[0\ 1\ 0]^T$, $[0\ 0\ 1]^T$ and $[1\ 0\ 0]^T$. This means X_3 should be pointing in the Y_0 direction, Y_3 should be pointing in the Z_0 direction and Z_3 should be pointing in the X_0 direction. A look at Figure 2.28 confirms this is correct.

Example 2.10

As covered in Chapter 1, the primary purpose of the major axes (the first three joints for a spatial robot) is to position the end-effector while the primary purpose of the minor axes (the joints after the major axes) is to orient the end-effector. If the minor axes consist of three revolute joints whose rotation axes intersect at a point this is termed a "spherical wrist". A spherical wrist with the configuration and positive motion directions shown in Figures 2.29 and 2.30 is known as an "Euler wrist". Note that joint 6 is orthogonal to joint 5, and joint 5 is orthogonal to joint 4. The point where the axes intersect is called the "wrist centre". Spherical wrists are employed with most robots.

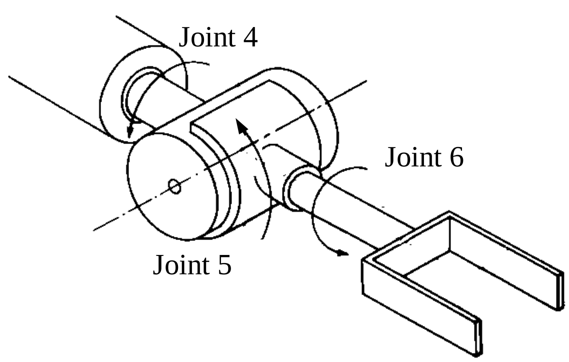


Figure 2.29 An Euler wrist [5].

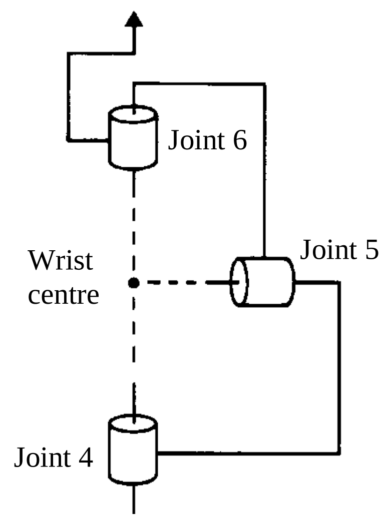


Figure 2.30 The wrist centre for a spherical wrist [2].

The D-H frames for an Euler wrist are shown in Figure 2.31. Note that frame 6 is attached to the tool and is offset from frame 5 in the Z_5 direction. The origins of frames 3-5 are equal to the wrist centre. The corresponding D-H parameters are listed in the table below. The joint variables are θ_4, θ_5 and θ_6.

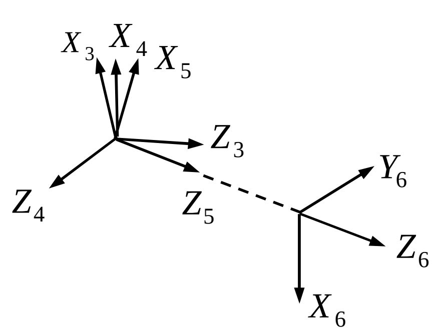


Figure 2.31 The D-H frames for the Euler wrist (Y axes have been omitted to improve clarity).

n+1	θ	d	a	α
4	θ_4	0	0	-90°
5	θ_5	0	0	90°
6	θ_6	d_6	0	0

The forward kinematics solution is:

$$^3T_6 = A_4 * A_5 * A_6 = \begin{bmatrix} C\theta_4 C\theta_5 C\theta_6 - S\theta_4 S\theta_6 & -C\theta_4 C\theta_5 S\theta_6 - S\theta_4 C\theta_6 & C\theta_4 S\theta_5 & d_6 C\theta_4 S\theta_5 \\ S\theta_4 C\theta_5 C\theta_6 + C\theta_4 S\theta_6 & -S\theta_4 C\theta_5 S\theta_6 + C\theta_4 C\theta_6 & S\theta_4 S\theta_5 & d_6 S\theta_4 S\theta_5 \\ -S\theta_5 C\theta_6 & S\theta_5 S\theta_6 & C\theta_5 & d_6 C\theta_5 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.61)$$

End of example 2.10.

Example 2.11

The PUMA-560 robot is an articulated robot whose design has been copied by several companies (recall Figure 1.17). It is a six DOF spatial robot that has all revolute joints. It also employs a wrist similar to an Euler wrist. Its D-H frames are shown in Figure 2.32. Due to the limitation of the D-H method that motion is only allowed along or about the X and Z axes, the origin of frame 3 is not coincident with frame 4. Its D-H parameters are given in the table below.

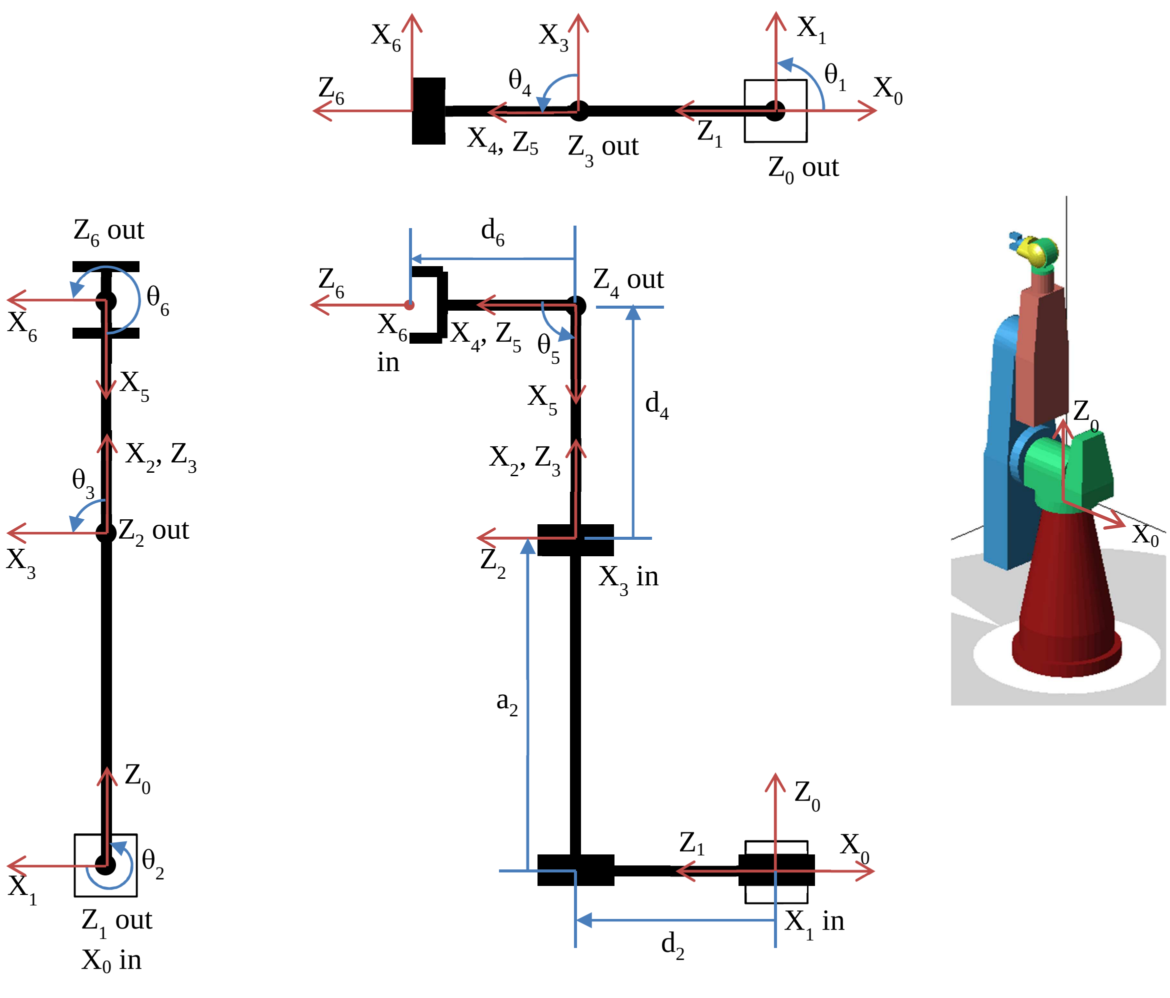


Figure 2.32 Left: Orthogonal views of the D-H frames for the PUMA-560 robot.
Right: Isometric view of a PUMA robot in the same pose (PUMA-762 model).

n+1	θ	d	a	α
1	θ_1	0	0	-90°
2	θ_2	d_2=149.5 mm	a_2=432 mm	0
3	θ_3	0	0	90°
4	θ_4	d_4=432 mm	0	-90°
5	θ_5	0	0	90°
6	θ_6	d_6=56.5 mm	0	0

End of example 2.11

Cylindrical Robot In-Class Exercise

A three DOF cylindrical robot is shown in Figure 2.33. The positive directions of the joints are indicated by the arrows.

a) Draw the D-H frames for this robot. Put the Z-axis for the tool frame in the direction of tool approach.
b) Determine the D-H parameters and put them in a table.
c) What are the joint variables?

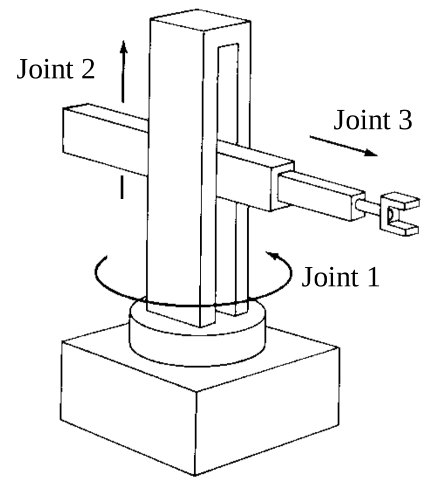


Figure 2.33 A three DOF cylindrical robot [1].

2.5 More Comments on Six DOF Industrial Robots

A typical six DOF industrial robot consists of three major axes followed by a spherical wrist. In this case the forward kinematics solution equals:

$$^0T_6 = (\text{Transformation for major axes}) * (\text{Transformation for wrist}) = {}^0T_3 * {}^3T_6 \tag{2.51}$$

Sometimes the base and tool frames defined using the D-H method are not appropriate. If this is the case then a new reference frame {R} and a new tool frame {H} can be defined, and the forward kinematics solution becomes

$$^RT_H = {}^RT_0 * {}^0T_6 * {}^6T_H \text{, where } {}^RT_H \text{ and } {}^6T_H \text{ are constant matrices.} \tag{2.52}$$

For example, with the Cartesian robot in example 2.9 we may want to have the Z-axis for the reference frame point vertically upwards instead of horizontally.

For more forward kinematics examples please see Chapter 2 of Niku's textbook [1].

Warning: Craig uses a different version of the D-H method in his textbook!

References

1. S.B. Niku, "Introduction to Robotics", Pearson Education, 2001.
2. J.J. Craig, "Introduction to Robotics", Addison Wesley Longman, 1989.
3. A.J. Koivo, "Fundamentals for Control of Robotic Manipulators", John Wiley & Sons, 1989.
4. M.W. Spong and M. Vidyasagar, "Robot Dynamics and Control", John Wiley & Sons, 1989.
5. P.G. Ranky and C.Y. Ho, "Robot Modelling", Springer-Verlag, 1985.