We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

154
Countries delivered to

Our authors are among the

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
**BOOK
CITATION
INDEX**
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Chapter

Kinematics for Spacecraft-Type Robotic Manipulators

Ijar Milagre da Fonseca, Maurício Nacib Pontuschka and Glaydson Luiz Bertoze Lima

Abstract

The scope of this chapter is the study of the forward and inverse kinematics for a space robot. The main focus is to compute the position and orientation of manipulators' end-effectors relative to their platform. Such platform plays the role of workstations referred in the literature approaching ground manipulators. In this study, the method is to write the manipulator kinematics' equations as functions of the joint variables by following the Denavit-Hartenberg convention. The homogeneous transform technique is used to study the kinematics. The set of coordinate frames defined in this chapter follows the convention for frames that appears in the literature for ground robot manipulators. The kinematics related to the spacecraft attitude is added in the formulation because the manipulator studied in this chapter is type spacecraft. The objective is to provide an overview and clear understanding of the kinematics' equations for spacecraft-type manipulators. To be consistent with orbital dynamics area, the inertial, orbital, and body-fixed coordinate frames are included in this kinematics study. The forward and inverse kinematics formulations are derived. The MATLAB®/Simulink tools are presented for the computer simulations of the forward and inverse kinematics.

Keywords: space robot manipulators, forward kinematics, inverse kinematics, attitude, orbit

1. Introduction

Kinematics is mainly concerned with the geometry of motion. The subject of kinematics is to some extent mathematical and does not consider any forces associated with the motion. A comprehensive study of kinematics is a prerequisite to the successful formulation of the equations of motion and the dynamic analysis. Consistent with the geometry of motion, kinematics involves the definition of systems of reference, methods of establishing relationships between frames, and algebraic manipulations of matrices and vectors. Kinematics is primarily concerned with describing the orientation of a body with respect to a known reference or coordinate frame. When dealing with robotic manipulators, we have to include also translational kinematics. In this chapter, the calculation goes beyond the algebraic manipulations of rotation matrices connecting frames. This includes the derivation of the equations for the rotational and the translational kinematics that characterize robot manipulators' kinematics. Space robotic manipulators are designed to implement orbital operations. There is a considerable difference regarding Earth-based manipulators, as they increase the type and number of reference systems for proper

problem formulation [1–4]. The main differences with regard to Earth-based robotic manipulators are the inclusion of an Earth-centered inertial coordinate frame (ECI), besides the local vertical, local horizontal (LVLH) reference frame, and the spacecraft fixed reference frame. These frames are used to describe the kinematics' differential equations related to orbit and attitude motions. The other reference frames are typical of those defined for ground manipulators. For space manipulators, we need to consider the relationships between all these reference systems.

2. Forward kinematics of manipulators and D-H convention—homogeneous transforms

Forward kinematics is the problem of computing the position and orientation of the robot tool with respect to a fixed reference frame defined on the base of the space robot. Such manipulators can be seen as a set of links connected in a chain of joints. The joints are points of connection between a pair of links. The joints of a manipulator may be revolute (rotatory), prismatic (sliding), or a combination of both. According to that link to joint configuration, any robot manipulator kinematics can be described in terms of four quantities associated with each link. Two of these quantities describe the link itself. The other two describe the link's connection to the neighboring link. For revolute joints, the angle θ_i is the joint variable, and the other three quantities are fixed link parameters. These parameters are illustrated in **Figure 1** as the twist angle α_{i-1}, the distances a_{i-1}, and d_i. For prismatic joints, the joint variable is d_i, while the angle θ_i is zero. In this case, α_{i-1}, and a_{i-1} are fixed link parameters (see **Figure 1**). The description of manipulators by these quantities is referred as the Denavit-Hartenberg convention, denoted here as D-H [5, 6]. The understanding of those four quantities requires the frame convention shown in **Figure 2**. The frames are denoted by the symbol {}, for example {i-1} denotes a frame whose origin is in joint *i-1*. A summary of the link parameters considering frames convention shown in **Figure 2** can be written as:

- a_i is the distance measured along x-axis from z_i to z_{i+1}

- α_i is the angle rotation measured about x_i from z_i to z_{i+1}

- d_i is the distance measured along z_i from x_{i-1} to x_i

- θ_i is a joint angle measured about z_i from x_{i-1} to x_i.
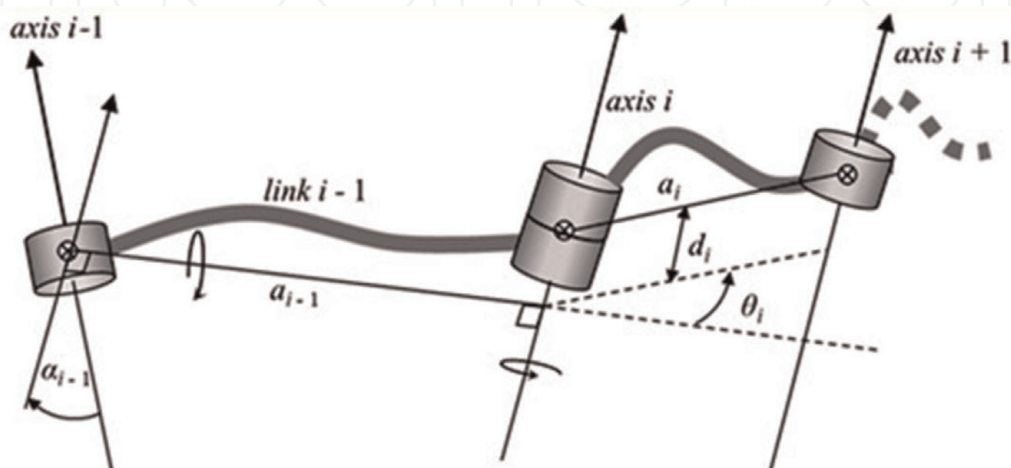
Figure 1.
D-H convention showing the notation, joint variable, and parameters.
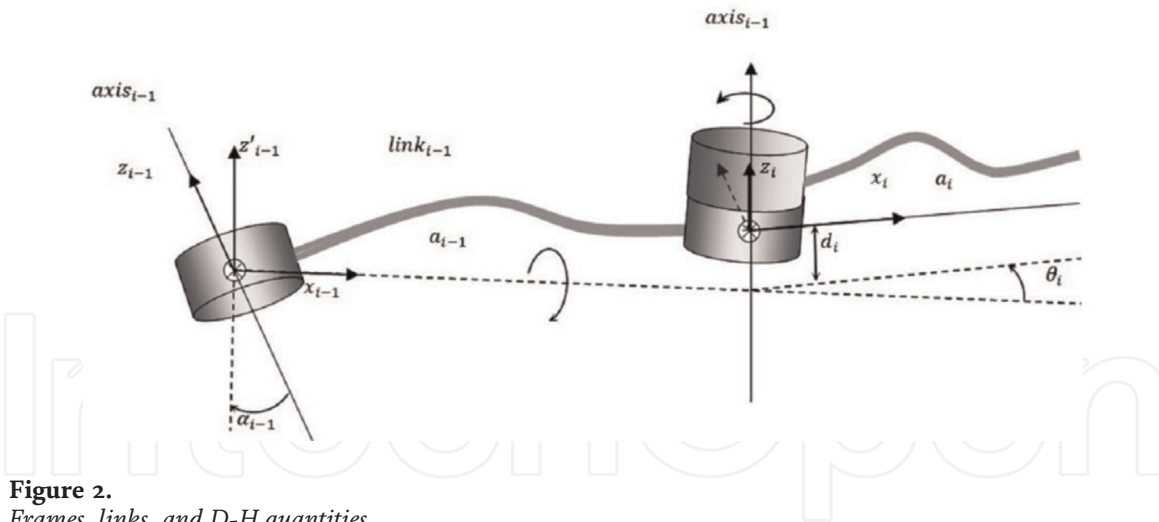
Figure 2.
Frames, links, and D-H quantities.

The D-H convention describes the manipulator rotational plus translational kinematics by using homogeneous transform. This transform is a 4 × 4 matrix in the form:

$$
\left\{ {}^{i-1}_{}p \atop 1 \right\} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & {}^{i-1}p_x \\ \sin\theta_i & \cos\theta_i & 0 & {}^{i-1}p_y \\ 0 & 0 & 1 & {}^{i-1}p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \left\{ {}^{i}p_o \atop 1 \right\} = \begin{bmatrix} {}^{1-1}_{i}R & {}^{i-1}p_o \\ 0 & 1 \end{bmatrix} \left\{ {}^{i}p \atop 1 \right\} \quad (1)
$$

The result is ${}^{i-1}_{i}p = {}^{i-1}_{i}R^i p + {}^{i-1}p_o$ and **1 = 1.** The subscript o is just to say that the position of the end-effector is given with respect to the origin of the frame ${}^{1-1}_{i}R$.

The 3 × 3 ${}^{1-1}_{i}R$ block matrix is related to the rotation between frames $\{i - 1\}$ and $\{i\}$ by the joint angle θ_i. The 3 × 1 vector ${}^{i-1}p_o$ is related to the end-effector position with respect o the frame $\{i - 1\}$. The advantage of this approach is the use of matrix notation including the position in addition to the orientation.

Such matrix notation facilitates the mathematical formulation in a compact and comprehensive format. Also, the D-H formulation is appropriate to implement problems in the MATLAB® computational environment.

Let us discuss now the manipulation of the frames as defined in **Figure 2** and their parameters. The D-H convention considers that all the joint angle rotations take place along z-axis. The rule is the same for the joint variables d_i (prismatic joints). In the case of prismatic joints, the joint angle is zero. This convention regarding joint variables along z-axis means that when z-axis is not parallel to the next one, a twist angle rotation α must be done so as to generate a new z-axis parallel to the next one. Angle α can be positive or negative. Another convention applies to the distance a_i. Once we have the new z-axis parallel to the next one, the distance a_i is determined by drawing a mutual perpendicular line between both z-axes. Such line will determine the distance between the joints and is also the place of a new x-axis. A y-axis completes the right-handed frame (**Figure 2**). If the line containing a_i crosses the z-axis offset of the joint, it defines a distance d_i as shown in **Figure 2**. Finally, we rotate the z-axis by the joint variable θ_i and a new frame is generated, taking into account the offset d_i.

On the base of the space robot attachment, the $Ox_0y_0z_0$ frame or $\{0\}$ is defined. Note that the frame is $\{i - 1\}$, and for $i = 1$, this frame is $\{0\}$. In the literature of robot manipulators, $\{0\}$ frame does not move with respect to the space robot platform. Accordingly, the joint associated to $\{0\}$ is also named joint 0 from which the link 0

connects to the joint 1. It is common to define joint 0 coincident with joint 1. In this case, a_0 is zero. For ground robots, it is common to define $Ox_0y_0z_0$ as an inertial frame. However, such definition is not consistent with the concept of inertial frames, mainly when we have space robot orbiting the Earth. For a space robotic manipulator, the $Ox_0y_0z_0$ frame is defined with its origin somewhere on the platform and locates the placement of joint zero. **Figures 2** and **3** illustrate the frames, joint axes, joint variables, and the other fixed quantities according to the D-H convention.

To include the translation in the same matrix notation, we use the homogeneous transform as shown in Eq. (1).

Before going to the other frames definition for the spacecraft, let us implement the matrix algebra, considering **Figure 2**. To clarify the procedure to obtain the general expression for the rotational-translational kinematics, let us redefine some frames in **Figure 2**.

- Let the frame formed by $x'_{i-1}, y'_{i}, z'_{i-1}$, obtained be a rotation α_{i-1} about x_{i-1}, be defined as $\{R\}$;

- Consider frame formed by $x'_{i-1}, y'_{i-1}, z'_{i-1}$ translated by a_{i-1} as $\{Q\}$;

- Consider the frame obtained by a rotation θ_i and parallel to frame $\{i\}$ as $\{P\}$;

- Consider frame $\{i\}$ obtained by a translation d_i.

Consider now **Figure 2** that shows the various frames in addition to the D-H parameters.

To write the kinematics' equations in the frame $\{i-1\}$, we need to compute the rotation matrices using the homogeneous transform as below
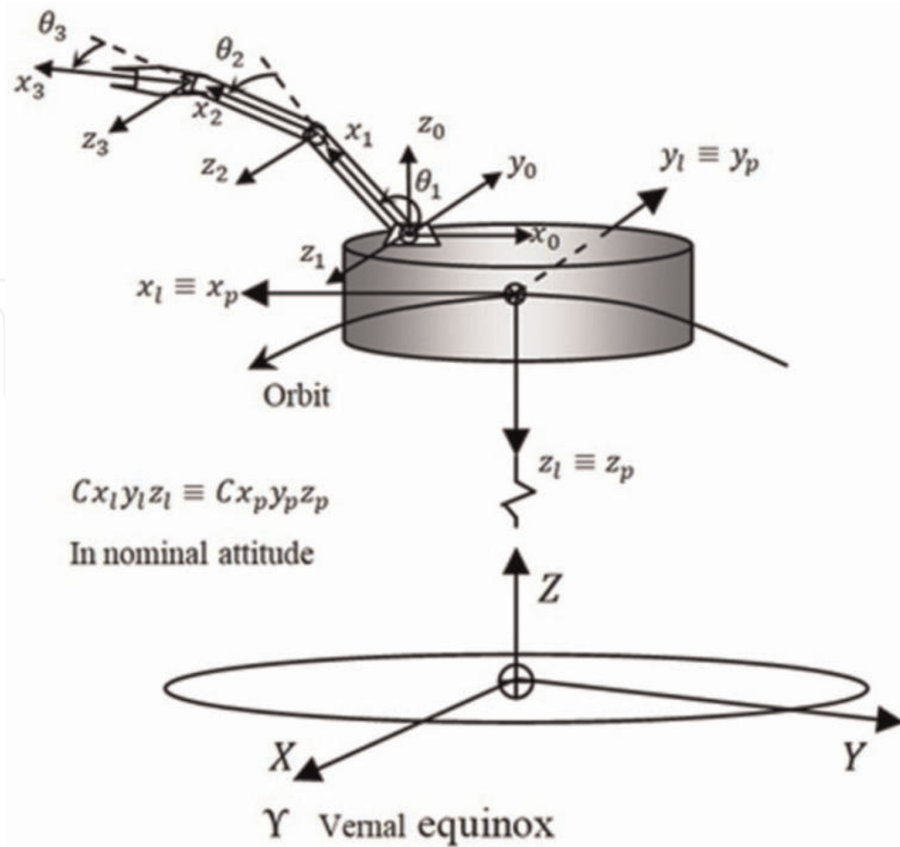
Figure 3.
Space robot on-orbit configuration and frames.

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

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

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

$$
=
\begin{bmatrix}
c\theta_i & -s\theta_i & 0 & a_{i-1} \\
s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1}d_i \\
s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1}d_i \\
0 & 0 & 0 & 1
\end{bmatrix}
$$

$$(2)$$

where ${}^{i-1}_{i}T = {}^{i-1}_{R}T^{R}_{Q}T^{Q}_{P}T^{P}_{i}T$ and the subscript to superscript means transform from the previous to the next coordinate frame.

These abbreviations will be used throughout this chapter.

$$s_i = s\theta_i = \sin\theta_i \tag{3}$$

$$c_i = c\theta_i = \cos\theta_i \tag{4}$$

$$c_{12} = c(\theta_1 + \theta_2) = c_1c_2 - s_1s_2 = c\theta_1c\theta_2 - s\theta_1s\theta_2 \tag{5}$$

$$s_{12} = s(\theta_1 + \theta_2) = c_1s_2 + s_1c_2 = c\theta_1s\theta_2 - s\theta_1c\theta_2 \tag{6}$$

$$c_{123} = c\left(\sum_{i}^{3}\theta_i\right) \text{ and } s_{123} = s\left(\sum_{i}^{3}\theta_i\right) \text{ for } i = 1, 2, 3 \tag{7}$$

In Eq. (2), the first matrix,

$$
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_{i-1} & -s\alpha_{i-1} & 0 \\ 0 & s\alpha_{i-1} & c\alpha_{i-1} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{8}
$$

is given by a rotation of α_{i-1} about x_{i-1}. Note that no translation appears in the matrix. That matrix makes the z_{i-1} parallel to the next, z_i-axis. The next matrix,

$$
\begin{bmatrix} 1 & 0 & 0 & a_{i-1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{9}
$$

refers to the translation a_{i-1}. This translation is along the x_{i-1}-axis, defining the distance a_{i-1}. As z_{i-1} is parallel to z_i, the distance a_{i-1} is a mutual perpendicular to both the previous and the next z-axis. The rotation matrix,

$$\begin{bmatrix} c\theta_i & -s\theta_i & 0 & 0 \\ s\theta_i & c\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad (10)$$

results of rotation about z_i by the joint angle θ_i . This rotation generates a frame parallel to $\{i\}$ at a distance d_i (offset) from the joint i. The 4th matrix,

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad (11)$$

translates the frame by d_i completing the transform from frame $\{i\}$ to $\{i-1\}$. Sometimes, it is more convenient to write the kinematic differential equations in the frame $\{i\}$. In this case, we have to transpose the rotation matrices so as to have the equations written in frame $\{i\}$ but in coordinates of $\{i-1\}$.

For space robots, we must compute the kinematics not only of the manipulator but also for the whole spacecraft (rotational and translational kinematics) [3, 4]. To do this, it is necessary to define three more frames:

- the Earth-centered inertial (ECI) frame $\bigoplus XYZ$, defined at the center of the Earth (the symbol $\bigoplus$ means Earth)

- the local vertical, local horizontal frame (LVLH) denoted as $Cx_l y_l z_l$. C is the center of mass of the space robot. The subscript l is a short for LVLH;

- and a spacecraft fixed system of reference, defined in the center of mass of the space robots [7] and denoted by $Cx_p y_p z_p$.

Figure 3 illustrates these new frames. The $Cx_l y_l z_l$ is fixed in the orbit. So, it follows with the orbit synchronized with the orbital velocity and with z_l pointing toward the local vertical (Nadir direction). In the literature, this frame is also named *RPY* (roll, pitch, and yaw). Roll is the rotation angle about the orbital velocity direction (*x*-direction), pitch is the rotation angle about the negative orbit normal (*y*-axis), and yaw is the rotation angle about the local vertical (z-axis). The roll, pitch, and yaw angles yield the space robot attitude with respect to the $Cx_l y_l z_l$ frame. Both the $Cx_l y_l z_l$ and the $Cx_p y_p z_p$ frames coincide with each other in a nominal attitude specification, that is, the space robot is pointing toward the local vertical, and the spacecraft attitude is zero. As the $Cx_p y_p z_p$ frame is fixed in the spacecraft, the attitude is given by the angles between $Cx_l y_l z_l$ and $Cx_p y_p z_p$. Other conventions may be used for the definition of these frames. The reference systems $\bigoplus XYZ$, $Cx_l y_l z_l$, and $Cx_p y_p z_p$ are included in the formulation to allow for describing the rotational and translational kinematics in frame $Cx_p y_p z_p$ or in the frame $Cx_l y_l z_l$. In general, the equations of dynamics are written in the body fixed frame, $Cx_p y_p z_p$. We can write the kinematics and/or the kinematics' differential equations in any of those three frames by using transformation matrices. The {ECI} allows writing the translational kinematic expressions in terms of the orbit parameters [7]. If the coupling between the orbit and attitude motion is negligible (in general, it is), we can write the rotational kinematics in the $Cx_p y_p z_p$ or $Cx_l y_l z_l$ frames, including the

orbital velocity only. The practice is to use the Euler angles to describe the angular orientation between both frames (the attitude of the spacecraft with respect the $Cx_l y_l z_l$ frame). According to the Euler Theorem, we can do this through three sequences of rotations from one frame to another frame. **Figure 3** shows both frames in a nominal configuration in which the attitude angles are zero and both frames are coincident. As the spacecraft moves, the fixed frames also move. **Figure 4** illustrates the rotation between those frames. So, we can derive the kinematic expressions by using the spacecraft fixed frame, $x_p y_p z_p$.

Let us consider that the spacecraft fixed frame is misaligned from the $Cx_l y_l z_l$ by the angles as shown in **Figure 4**. **Figure 4a** illustrates the space robot in-orbit configuration with attitude different from zero. Note that in this figure, both $Cx_l y_l z_l$ and $Cx_p y_p z_p$ are misaligned. **Figure 4b** illustrates the sequence of rotations from $Cx_l y_l z_l$ to $Cx_p y_p z_p$.

The sequence of three rotations is $\varphi \longleftarrow \theta \longleftarrow \psi$ [7]. The arrows indicate the direction of the sequence as: first, a rotation about z_l by the angle ψ generating an auxiliary frame $Cx'_p y'_p z'_p$. Then, a second rotation is done about y'_l by the angle θ generating the auxiliary frame $Cx''_p y''_p z''_p$. Finally, the third rotation is done about x'' to coincide with $Cx_p y_p z_p$. There are 12 sets of Euler angle sequences of rotations to describe the rotational kinematics. However, the sequence of rotations $\varphi \longleftarrow \theta \longleftarrow \psi$ is one the most used transformations for satellites' rotational kinematics description. This transform provides the relationship between both $Cx_l y_l z_l$ and $Cx_p y_p z_p$ frames. The notation of the sequence used here is the same as $\varphi \longrightarrow \theta \longrightarrow \psi$. However, the first notation emphasizes the resulting structure of matrix multiplication when implementing the transformation from $Cx_l y_l z_l$ to $Cx_p y_p z_p$ frames as we are going to show next.

First rotation, about z_l

$$\begin{Bmatrix} x'_p \\ y'_p \\ z'_p \end{Bmatrix} = \begin{bmatrix} c\psi & s\psi & 0 \\ -s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x_l \\ y_l \\ z_l \end{Bmatrix} \tag{12}$$
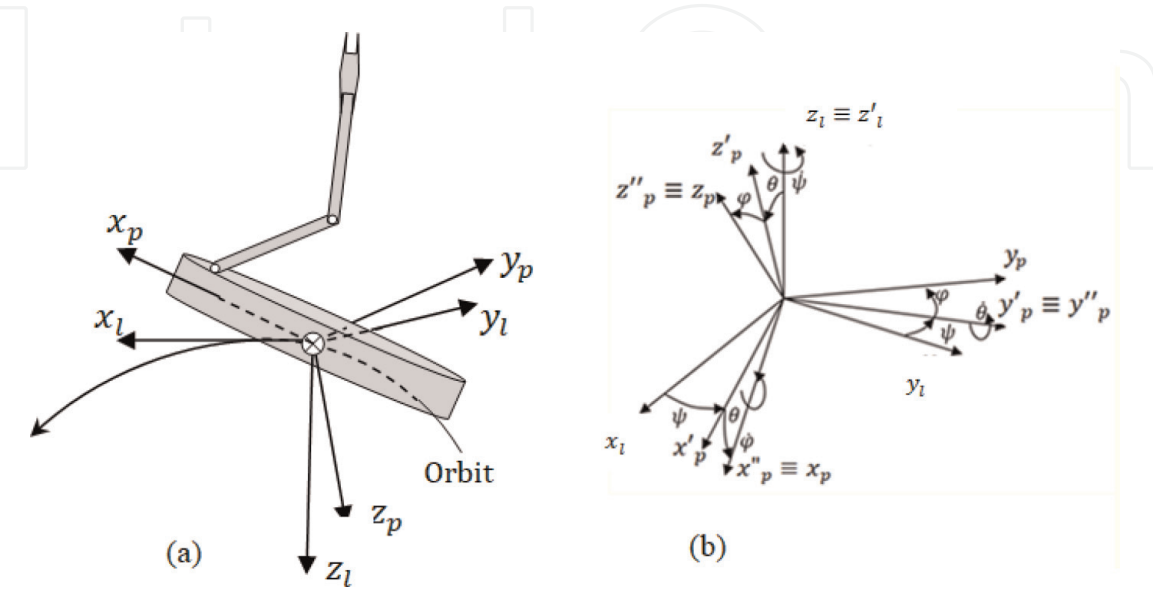
Figure 4.
(a) Satellite on-orbit configuration with $Cx_p y_p z_p$ misaligned from $Cx_l y_l z_l$ and (b) the Euler sequence of rotations $\varphi \longrightarrow \theta \longrightarrow \psi$.

Second rotation, about $y'_p \equiv y''_p$

$$\begin{Bmatrix} x''_p \\ y''_p \\ z''_p \end{Bmatrix} = \begin{bmatrix} c\theta & 0 & -s\theta \\ 0 & 1 & 0 \\ s\theta & 0 & c\theta \end{bmatrix} \begin{Bmatrix} x'_p \\ y'_p \\ z'_p \end{Bmatrix} \tag{13}$$

Third rotation, about $x''_p \equiv x_p$

$$\begin{Bmatrix} x_p \\ y_p \\ z_p \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\varphi & s\varphi \\ 0 & -s\varphi & c\varphi \end{bmatrix} \begin{Bmatrix} x''_p \\ y''_p \\ z''_p \end{Bmatrix} \tag{14}$$

By substituting Eq. (13) into Eq. (14) and substituting Eq. (12) into the result, we have:

$$\begin{Bmatrix} x_p \\ y_p \\ z_p \end{Bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\varphi & s\varphi \\ 0 & -s\varphi & c\varphi \end{bmatrix} \begin{bmatrix} c\theta & 0 & -s\theta \\ 0 & 1 & 0 \\ s\theta & 0 & c\theta \end{bmatrix} \begin{bmatrix} c\psi & s\psi & 0 \\ -s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x_l \\ y_l \\ z_l \end{Bmatrix} \tag{15}$$

So, the structure is really like $\varphi \longleftarrow \theta \longleftarrow \psi$.

The final matrix allows writing the equations in the spacecraft fixed frame $Cx_p y_p z_p$ in coordinates of $Cx_l y_l z_l$ as:

$$\begin{Bmatrix} x_p \\ y_p \\ z_p \end{Bmatrix} = \begin{bmatrix} c\theta c\psi & c\theta s\psi & -s\theta \\ s\varphi s\theta c\psi - c\varphi s\psi & s\varphi s\theta s\psi + c\varphi c\psi & s\varphi c\theta \\ c\varphi s\theta c\psi + s\varphi s\psi & c\varphi s\theta s\psi - s\varphi c\psi & c\varphi c\theta \end{bmatrix} \begin{Bmatrix} x_l \\ y_l \\ z_l \end{Bmatrix} = {}^p_l \boldsymbol{R} \tag{16}$$

By transposing this matrix, we can write the kinematics' equation in the LVLH frame:

$$\begin{Bmatrix} x_l \\ y_l \\ z_l \end{Bmatrix} = \begin{bmatrix} c\theta c\psi & c\theta s\psi & -s\theta \\ s\varphi s\theta c\psi - c\varphi s\psi & s\varphi s\theta s\psi + c\varphi c\psi & s\varphi c\theta \\ c\varphi s\theta c\psi + s\varphi s\psi & c\varphi s\theta s\psi - s\varphi c\psi & c\varphi c\theta \end{bmatrix}^{\mathrm{T}} \begin{Bmatrix} x_p \\ y_p \\ z_p \end{Bmatrix} = {}^l_p \mathrm{R}\{p\} \tag{17}$$

The relationships from $\{n\}$ to $\{0\}$ can be written as:

$$ {}^0_n T = {}^0_1 T {}^1_2 T {}^2_3 T ... {}^{n-1}_n T \tag{18}$$

To write the kinematics of the spacecraft-like robot manipulator in the LVLH system of reference, we have to concatenate all the matrices. By using Eqs. (17) and (18) and Eq. (2), we can establish the relationships between i the coordinate frames as:

$$ {}^l_i T = {}^l_P T {}^p_0 T {}^{i-1}_i T \tag{19}$$

where

$$ {}^l_p T = \begin{bmatrix} {}^l_p \boldsymbol{R}_{3\times 1} & \boldsymbol{0}_{3\times 1} \\ \boldsymbol{0}_{1\times 3} & 1 \end{bmatrix} = \begin{bmatrix} c\theta c\psi & s\varphi s\theta c\psi - c\varphi s\psi & c\varphi s\theta c\psi + s\varphi s\psi & r_{cx} \\ c\theta s\psi & s\varphi s\theta s\psi + c\varphi c\psi & s\varphi c\theta & r_{cy} \\ -s\theta & s\varphi c\theta & c\varphi c\theta & r_{cz} \\ 0 & 0 & 0 & 1 \end{bmatrix}; $$

$$
{}_0^p T = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \text{—this transform relates } Ox_O y_O z_O \text{ with } Cx_p y_p z_p \text{ (see } \textbf{Figure 3})
$$

${}_i^{i-1}T$ is given by Eq. (2)

$$
i = 1, 2, ..., n
$$

The vector $\{r_c\} = \{r_{cx} \; r_{cy} \; r_{cz}\}^T$ represents the translation from the center of mass (CM) to the robot base or the $Ox_O y_O z_O$ frame. This is a constant vector in this formulation. However, when the robot manipulator is in orbit operation, the links move, changing the location of the system CM, in other words, the CM location becomes a variable of the problem. By the time the space shuttle was in operation, this type of problem appeared. When it grasped significant massive target, the system CM motion affected the efficiency of the robot arm to put the target where it was planned. Another problem experienced by the space shuttle manipulator was the interaction between the attitude control system and the robot arm structural flexibility. The attitude control was in action to keep the shuttle stabilized while the manipulator was commanded to grasp a target. The structural flexibility of the long arms was excited by the attitude actuators and entered in elastic vibration mode causing problems to accomplish a safety grasping of the target. But this is another story and may be subject of research on another occasion.

Example 1. Consider the robot manipulator shown in **Figure 3** where the space robot is stabilized with $Cx_l y_l z_l$ and $Cy_p z_p$ frames. Find the kinematics relationship written in the $Ox_O y_O z_O$ frame. First, we built the D-H as shown in **Table 1** for the problem.

By inspection in **Figure 3**, we can see that all the z-axes are parallel and the joints are revolute. So, the parameters α and d are zeros. Following the D-H table:

$$
{}_1^0 T = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 1 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{20}
$$

$$
{}_2^1 T = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & l_1 \\ s\theta_2 & c\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{21}
$$

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

Table 1.
D-H parameters for Example 1.

$$
{}_3^2T = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & l_2 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\tag{22}
$$

so

$$
{}_3^0T = {}_1^0T\,{}_2^1T\,{}_3^2T = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 1 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & l_1 \\ s\theta_2 & c\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & l_2 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\tag{23}
$$

Compacting the result through the definition of $c = \cos$, $s = \sin$, and $i = \theta_i, i = 1, 2, 3$ we have

$$
{}_3^0T = \begin{bmatrix} c_3c_{12} - s_3s_{12} & -c_3s_{12} - s_3c_{12} & 0 & l_1c_1 + l_2c_{12} \\ c_3s_{12} + s_3c_{12} & c_3c_{12} - s_3s_{12} & 0 & l_1s_1 + l_2s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c_{123} & -s_{123} & 0 & l_1c_1 + l_2c_{12} \\ s_{123} & c_{123} & 0 & l_1s_1 + l_2s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\tag{24}
$$

where

$$
c_{123} = \cos\left(\sum_{i=1}^{3} \theta_i\right) = c_3c_{12} - s_3s_{12}
\tag{25}
$$

$$
s_{123} = \sin\left(\sum_{i=1}^{3} \theta_i\right) = c_3s_{12} + s_3c_{12}
\tag{26}
$$

$$
i = 1, 2, 3
$$

Note that we considered the spacecraft stable in attitude with the $Cx_p y_p z_p$ aligned to the $Cx_l y_l z_l$ frame. But be aware that if the manipulator moves in attitude, the kinematics' differential equations of the spacecraft shall account for the manipulator links of kinematics' differential equations. In particular, if the manipulator is to grasp an on-orbit target, the attitude of the manipulator-like spacecraft must be synchronized to that of the target so that the relative attitude of both spacecraft and target is zero. As the dynamic analysis is out of the scope of this chapter, we will assume that the robot arm kinematics' equations differ from ground robots only by $Cx_l y_l z_l$ and $x_p y_p z_p$ frames. For orbital robots these frames are used to describe the kinematics differential equations related to orbit and attitude motions.

Example 2. Consider **Figure 5** which illustrates a revolute-prismatic-revolute (RPR) manipulator. In this case, we have a rotation (θ_1), a translational variable (d_2), and another rotation (θ_2) as illustrated in **Table 2**. The convention is that those degrees-of-freedom occur along z-axis.

For this case, we have one rotation of θ_1, translation of d_2, and a rotation of θ_3.

Following the D-H convention above, we should make a few comments. First, we have a rotation on z-axis. Then, we use the twist angle rotation to make a
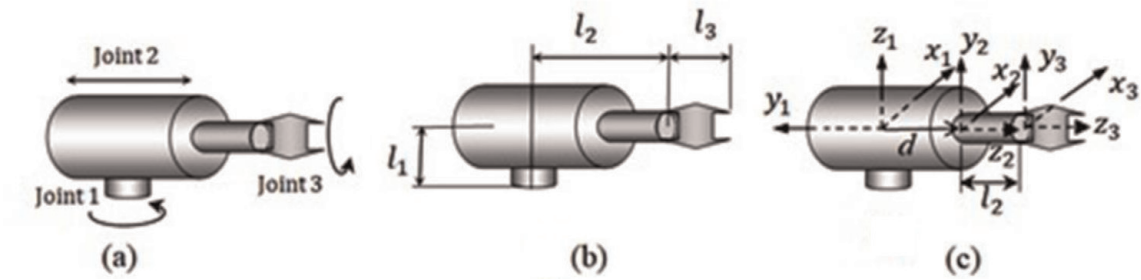
Figure 5.
(a) RPR manipulator showing the joints, (b) D-H parameters, and (c) frames.

new z-axis for the joint variable d_2. The sequence of rotations is shown in Eqs. (18)–(20).

$$
{}_1^0T = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 \\ s\theta_1 & c\theta_1 & 0 \\ 0 & 0 & a1 \end{bmatrix} \tag{27}
$$

$$
{}_2^1T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\alpha_1 & -s\alpha_1 \\ 0 & c\alpha_1 & c\alpha_1 \end{bmatrix} \tag{28}
$$

$$
{}_3^2T = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 \\ s\theta_3 & c\theta_3 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{29}
$$

To account for the translation d_2, we can use the homogeneous transformation as:

$$
\begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{30}
$$

$$
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_1 & -s\alpha_1 & 0 \\ 0 & s\alpha_1 & c\alpha_1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{31}
$$

$$
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 0 \end{bmatrix} (\alpha_1 = 90^o) \tag{32}
$$

$$
\begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & 0 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 1 & l_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{33}
$$

Note that this notation of homogeneous matrix allows including the two translations of this problem. Also, note that the offset translation is along z. The transform matrix ($\alpha_2 = 90^o$) yields:

$$
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_1 & -s\alpha_1 & 0 \\ 0 & s\alpha_1 & c\alpha_1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 0 \end{bmatrix}
$$

$$
= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & -d_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$(34)$$

This result is a transformation, $_2^1 T$, that is, the transform from frame {2} to frame {1}. Combining all the transformations, we can write from frame {3} to frame {0} as:

$$
_3^0 T = {}_1^0 T {}_2^1 T {}_3^2 T = \begin{bmatrix} c\theta_1 & -s\theta_2 & 0 & 0 \\ s\theta_2 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & -d_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & 0 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
= \begin{bmatrix} c\theta_1 c\theta_3 & -c\theta_1 s\theta_3 & s\theta_3 & s\theta_1(l_2 + d_2) \\ s\theta_1 c\theta_3 & -s\theta_1 s\theta_3 & -c\theta_1 & -c\theta_1(l_2 + d_2) \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$(35)$$

We can also find the transpose of this transformation to have the transformation:

$$
_0^3 T = {}_2^3 T {}_1^2 T {}_0^1 T \begin{bmatrix} c\theta_3 & s\theta_3 & 0 & 0 \\ -s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -d_2 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\theta_1 & s\theta_1 & 0 & 0 \\ -s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
= \begin{bmatrix} c\theta_1 c\theta_3 & s\theta_1 c\theta_3 & s\theta_3 & 0 \\ -c\theta_1 s\theta_3 & -s\theta_1 s\theta_3 & c\theta_3 & 0 \\ s\theta_1 & -c\theta_1 & 0 & -(l_2 + d_2) \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$(36)$$

3. Inverse kinematics

Inverse kinematics consists of finding the joint variables given the robot manipulator end-effector position and orientation with respect to the user workstation. The workstation means a known location and reference system. It could be the platform of the robotic system. In this chapter, the platform reference system is the frame $Cx_p y_p z_p$ represented by {P}. In general, such frame is defined in the center of mass of

the space robotic system. We know the location of that system of reference. Also, we know the location of the $Cx_ly_lz_l$ frame because we know the space robot orbit. In the previous section, we considered the relationship between frames to write the equations in the {0}. In the inverse problem, we want to find joint variables given the specified position and orientation of the tool relative to {0} or {p}. We assume that the robot attitude and orbit control subsystem keeps the space robot stabilized while it works. This is to simplify the formulation. This assumption is reasonable since this is exactly what must be done when the space robot is in operation. We have already presented and discussed the several frames defined to study the robot kinematics. We will refer to just one more, the frame defined on the tool, {T}.

There are some aspects of inverse kinematics that make it more complex than forward kinematics [5, 6]. We will briefly report them, but we will not go into too many details. In this chapter, our focus is on the solution methods for inverse kinematics problem.

One of the main aspects of the inverse kinematics is solvability because we have to face the problem of existence of solutions and of multiple solutions. Also, this problem brings to light the concept of manipulator workspace which is the volume of space that the end-effector of the manipulator can reach. For space robot, this problem is critical because if the object to be manipulated is beyond space robot workspace, the space mission may not accomplish goals like grasping a target, docking to another spacecraft, put a piece of structure in right location during assembling space structures, and so on. For a solution to exist, the point specified to be reached by the end-effector must be inside the manipulator workspace. Another problem related to the workspace of robots is to plan a trajectory [9] preventing collisions with objects, another robot, or astronaut inside the workspace. There are two useful definitions related to workspace. One is the dextrous workspace [5] and the other is the reachable workspace. The first refers to the space volume that the end-effector reaches with all orientation. The second is that volume of space that the robot can reach in at least one orientation. If the position and orientation specified for the robot task is inside the workspace, then at least one solution exists. Workspace also depends on the tool-frame transformation, since it is frequently the tool-tip that is discussed when we discuss reachable points in space. Details about the solvability problem for inverse kinematics may be seen in [5].

3.1 The inverse kinematics solution methods

In solving inverse problems, the first thing we must consider is that there are no general algorithms [5, 8, 9] that may be employed to solve manipulator kinematics. We state that a manipulator is solvable when the joint variables can be determined by an algorithm for a given position and orientation of the tool frame, {T}.

The solutions may be classified as close solution and numerical solutions. Numerical solutions are in general slower than close solutions and would require another chapter for discussion. We will concentrate in the close solutions methods. These methods can be subdivided into algebraic and geometric solutions. This classification is somehow hazy since each solution method uses both algebraic and geometric kinematic formulations. In presenting and discussing the methods, we point out that according to the concept of solution we stated here, the revolute and prismatic manipulators having six degrees of freedom (DOF) in series chain are solvable.

3.2 Algebraic solution

Consider the robot manipulator mounted on the spacecraft platform and the on-orbit configurations in the nominal attitude as shown in **Figure 3**. Let us apply the

algebraic solution. Note that the frame defined in the last link is the frame for n = 3. So, the transform from frame {0} to frame {N} is given by Eq. (24) repeated here for didactic reasons.

$$
{}_{N}^{O}T = \begin{bmatrix} c_{123} & -s_{123} & 0 & l_1c_1 + l_2c_{12} \\ s_{123} & c_{123} & 0 & l_1s_1 + l_2s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\tag{37}
$$

Now consider that for inverse kinematics, the specified position of the end-effector is given:

$$
{}_{N}^{O}T = \begin{bmatrix} c\varphi & -s\varphi & 0 & x \\ s\varphi & c\varphi & 0 & y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\tag{38}
$$

Note that this specified orientation and position are the angle φ and the Cartesian position (x, y, 0) of the tool. As Eq. (37) must be equal to Eq. (38), the problem can be solved by comparing the elements of the matrices. The joint angles and the parameters l_i, $i = 1, 2$ are unknown. The specified orientation given by φ and the components of the vector $\{x \quad y \quad z\}^T$, in the planar case, x and y are known and represent the tool position with respect to {0}. By equating a_{11} element of both matrices we have:

$$
c\varphi = c_{123} = \cos\left(\sum_{i=1}^{3} \theta_i\right)
\tag{39}
$$

$$
s\varphi = s_{123} = \cos\left(\sum_{i=1}^{3} \theta_i\right)
\tag{40}
$$

$$
x = l_1c_1 + l_2c_{12}
\tag{41}
$$

$$
y = l_1s_1 + l_2s_{12}
\tag{42}
$$

where

$$
c_{12} = c_1c_2 - s_1s_2
$$

$$
s_{12} = c_1s_2 + s_1c_2
$$

The algebraic solution requires the solution of Eq. (39) to Eq. (42). We can compute the square of Eqs. (41) and (42) to obtain:

$$
x^2 + y^2 = l_1^2 + l_2^2 + 2l_1l_2c_2
\tag{43}
$$

Solving Eq. (43) by c_2, we compute c_2 as:

$$
c_2 = \frac{x^2 + y^2 - (l_1^2 + l_2^2)}{2l_1l_2}
\tag{44}
$$

Important information can be extracted from Eq. (44). We know that the cosine function must be between −1 and + 1. By checking such constraint, we obtain information whether or not the solution exists. In case a solution does not exist, the meaning is that the point is not inside the robot manipulator workspace.

Let us assume that the orientation and position specified in inside the workspace. In this case, we can compute s_2 as:

$$s_2 = \pm\sqrt{1 - c_2^2} \tag{45}$$

The joint angle can then be computed by:

$$\theta_2 = \text{Atan2}\left(\frac{s_2}{c_2}\right) \tag{46}$$

$\text{Atan2}\left(\frac{s_2}{c_2}\right)$ computes $\tan^{-1}\left(\frac{s_2}{c_2}\right)$, but uses the signs of both s_2 and c_2 to identify the quadrant in which the resulting angle lies.

Note that there are signs to be considered in Eq. (45). The signs are related to multiple solutions.

Once we have obtained θ_2, we can obtain θ_1 by manipulating Eqs. (41) and (42). To solve the problem, we should define:

$$k_1 = l_1 + l_2 c_2 \tag{47}$$
$$k_2 = s_2 \tag{48}$$

and use the trigonometric identities

$$c_{12} = c_1 c_2 - s_1 s_2 \tag{49}$$
$$s_{12} = c_1 s_2 + s_1 c_2 \tag{50}$$

to rewrite Eqs. (41) and (42) as:

$$x = k_1 c_1 - k_2 s_1 \tag{51}$$
$$x = k_1 s_1 + k_2 c_1 \tag{52}$$

Next, we make a change of variable by writing k_1 and k_2 in the form:

$$r = +\sqrt{k_1^2 + k_2^2} \tag{53}$$

and

$$\gamma = \text{Atan2}(k_1, k_2) \tag{54}$$

Using these new variables, we can write Eqs. (51) and (52) as:

$$\frac{x}{r} = c\gamma c\theta_1 - s\gamma cs\theta_1 = c(\gamma + \theta_1) \tag{55}$$

$$\frac{y}{r} = c\gamma s\theta_1 + s\gamma\theta_1 = s(\gamma + \theta_1) \tag{56}$$

from which

$$\gamma + \theta_1 = \text{Atan2}\left(\frac{y}{r}, \frac{x}{r}\right) = \text{Atan2}(y, x) \tag{57}$$

Then, using the definition of γ, we obtain the expression for θ_1 as:

$$\theta_1 = \text{Atan2}(y, x) - \text{Atan2}(k_1, k_2) \tag{58}$$

As we have θ_1 and θ_2, we can find θ_3 by using Eqs. (39) and (40)

$$\sum_{i=1}^{3} \theta_i = \text{Atan2}(s\varphi, c\varphi) = \varphi = \theta_1 + \theta_2 + \theta_3 \qquad (59)$$

Once have θ_1 and θ_2, we obtain θ_3 from Eq. (59).

Transcendental equations commonly arise when applying the algebraic approach. The greater the number of degrees of freedom of the robotic manipulator, the more laborious is the solution of the problem.

3.3 Geometric solution

By using this approach, we simply work the geometry of the arm. For the same problem, we may have the position of the joint associated to the frame {N} with respect to the origin of frame {O} by using geometry. Frame {N} refers to the frame attached to the last link of the manipulator. Consider **Figure 6** showing the triangle formed by the geometrical configuration of the sides of the links and the line from the origin of the frame {O} to the origin of frame {N}. Applying the law of cosines, we can write:

$$x^2 + y^2 = l_1^2 + l_2^2 - 2l_1l_2 \cos(180 - \theta_2) = l_1^2 + l_2^2 + 2l_1l_2c_2 \qquad (60)$$
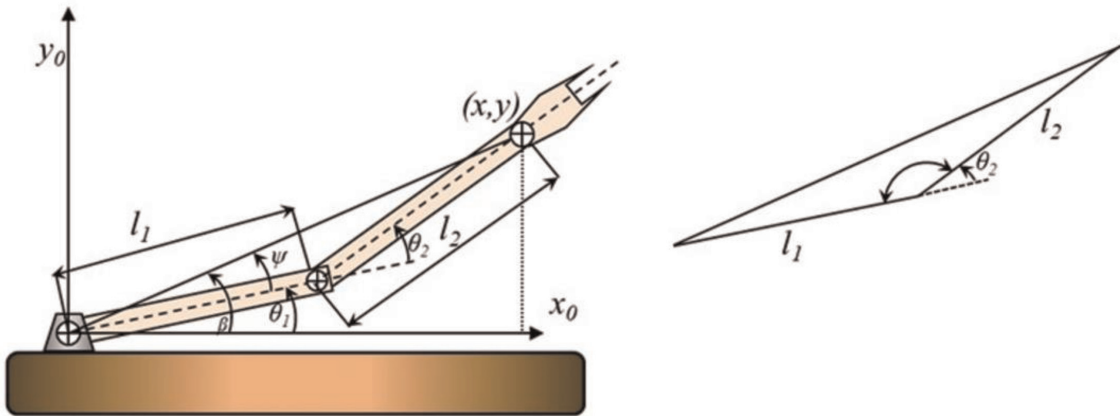
$$c_2 = \frac{x^2 + y^2 - (l_1^2 + l_2^2)}{2l_1l_2} \qquad (61)$$

Again, this solution must be between 1 and -1 in order to have a valid solution. Assuming that the solution exists, then:

$$s_2 = \sqrt{1 - \left[\frac{x^2 + y^2 - (l_1^2 + l_2^2)}{2l_1l_2} \right]^2} \qquad (62)$$

Finally, we can write another solution for θ_2 in terms of the tangent since we have both sin and cosine of the angle θ_2

$$\theta_2 = \text{Atan2}\left\{ \pm\sqrt{1 - \left[\frac{x^2 + y^2 - (l_1^2 + \ell_2^2)}{2l_1l_2} \right]^2}, \frac{x^2 + y^2 - (l_1^2 + l_2^2)}{2l_1l_2} \right\} \qquad (63)$$

Figure 6.
Space robot manipulator—geometric solutions for the inverse kinematics.

Finally, we can write other solution for θ_2 in terms of the tangent since we have both sin of and cosine of the angle θ_2

Now Let us compute a solution for θ_1

$$\beta = \text{Atan2}(y, x) \tag{64}$$

For ψ, we use the law of cosines to obtain:

$$c\psi = \frac{x^2 + y^2 + l_1{}^2 - l_2{}^2}{2l_1 l_2} \tag{65}$$

Geometry yields then

$$\theta_1 = \beta \pm \psi \tag{66}$$

where $\beta = \text{Atan2}(y, x)$.

ψ is obtained by using the law of cosines as $\ell_2{}^2 = x^2 + y^2 + l_1{}^2 - 2\sqrt{x^2 + y^2}l_1\cos\psi$

By solving this equation by $\cos\psi$, we obtain Eq. (65). Finally, we use Eqs. (64) and (65), to obtain θ_1 as in Eq. (66).

Now, we can compute θ_3 from:

$$\sum_{i=1}^{3} \theta_i = \theta_1 + \theta_2 + \theta_3 = \varphi \tag{67}$$

Solving Eq. (67) for θ_3, we complete the solution. Note that φ is the specified orientation and we had already computed θ_1 and θ_2. Thus, the problem is solved.

3.4 Algebraic solution method involving working matrix algebra

Let us consider a six-DOF revolute manipulator. The transformation from frames {O} to the {N} (N = 6) is

$$
{}^0_6T = \begin{bmatrix} a_{11} & a_{12} & a_{13} & p_x \\ a_{21} & a_{22} & a_{23} & p_y \\ a_{31} & a_{32} & a_{33} & p_z \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{68}
$$

We know that this final matrix represents the following successive transforms

$$
{}^0_6T = {}^0_1T\,{}^1_2T\,{}^2_3T\,{}^3_4T\,{}^4_5T\,{}^5_6T \tag{69}
$$

Where, for the revolute manipulator

$$
{}^0_1T = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{70}
$$

By premultiplying the left side of Eq. (68) by the inverse of Eq. (70), we obtain:

$$\begin{bmatrix} c_1 & s_1 & 0 & 0 \\ -s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} & p_x \\ a_{21} & a_{22} & a_{23} & p_y \\ a_{31} & a_{32} & a_{33} & p_z \\ 0 & 0 & 0 & 0 \end{bmatrix} = {}_2^1T\,{}_3^2T\,{}_4^3T\,{}_5^4T\,{}_6^5T\,T \tag{71}$$

By equating both sides of Eq. (71), we start identifying solutions by comparing the elements of matrices from both sides. Then, we continue by multiplying both sides, by the inverse of the ${}_2^1T$ and repeat the process to identify more solutions to obtain:

$$\begin{bmatrix} c_2 & s_2 & 0 & -l_1c_2 \\ -s_2 & c_2 & 0 & l_1s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} & p_x \\ a_{21} & a_{22} & a_{23} & p_y \\ a_{31} & a_{32} & a_{33} & p_z \\ 0 & 0 & 0 & 0 \end{bmatrix} = {}_3^2T\,{}_4^3T\,{}_5^4T\,{}_6^5T\,T \tag{72}$$

By implementing this process, we can solve the inverse kinematics for the six-DOF revolute joints with all the z-axis parallels.

Consider **Table 2** and the associated transformation matrix Eq. (36) for the RPR manipulator of **Figure 5**. Let us solve the inverse kinematics using the above method of matrix algebra. Al the matrix manipulations were computed by using the MATLAB® symbolic manipulator. The specified orientation and position is given by Eq. (68), by premultiplying the right side of this equation by inverse of Eq. (30), we obtain:

$$\begin{bmatrix} c_1 & s_1 & 0 & 0 \\ -s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} & p_x \\ a_{21} & a_{22} & a_{23} & p_y \\ a_{31} & a_{32} & a_{33} & p_z \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & 0 \\ 0 & 0 & -1 & -(l_2 + d_2) \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} a_{11}c\theta_1 + a_{21}s\theta_1 & a_{12}c\theta_1 + a_{22}cs & a_{13}s\theta_1 & p_xc\theta_1 + p_ys\theta_1 \\ a_{21}c\theta_1 - a_{11}s\theta_1 & a_{22}c\theta_1 - a_{12}c\theta_1 & a_{23}c\theta_1 - a_{13}s\theta_1 & p_yc\theta_1 - p_xs\theta_1 \\ a_{31} & a_{32} & a_{33} & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}_3^1T \tag{73}$$

Comparing the resulting matrices, on the left and right, for the elements (1,4) and (2,4), respectively, we obtain:

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

Table 2.
The D-H parameters and joint variables.

$$\theta_1 = -\text{Atan2}\left(p_x, p_y\right) \tag{74}$$

$$d_2 = p_x s\theta_1 - p_y c\theta_1 - l_2 \tag{75}$$

Comparing elements from both sides yields

$$p_z = 0 \tag{76}$$

Premultiplying the result of left side again by the inverse of $^1_2 T$, we obtain:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & -d_3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_1 & s_1 & 0 & 0 \\ -s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & a_{13} & p_x \\ a_{21} & a_{22} & a_{23} & p_y \\ a_{31} & a_{32} & a_{33} & p_z \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & 0 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ s\theta_3 & c\theta_3 & 1 & l_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{77}$$

The left side yields

$$\begin{bmatrix} a_{11}c\theta_1 + a_{21}s\theta_1 & a_{12}c\theta_1 + a_{22}s\theta_1 & a_{13}c\theta_1 + a_{23}s\theta_1 & p_x c\theta_1 + p_y s\theta_1 \\ a_{31} & a_{32} & a_{33} & p_z \\ a_{11}s\theta_1 - a_{21}c\theta_1 & a_{12}s\theta_1 - a_{22}c\theta_1 & a_{13}s\theta_1 - a_{23}c\theta_1 & p_x s\theta_1 - p_y c\theta_1 - d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{78}$$

By comparing the matrix elements (2,1) and (2,2) for both sides, we can compute θ_3 as:

$$a_{31} = s\theta_3 \text{ and } a_{32} = c\theta_3 \tag{79}$$

then, we obtain from $\tan\theta_3 = \frac{a_{31}}{a_{32}} \Longrightarrow \theta_3 = \text{Atan2}(a_{31}, a_{32})$

Of course, this example is simple. When we have a six-DOF manipulator, the algebraic manipulation to solve the inverse kinematics is hard, and the solution may require using transformation of variables. The MATLAB® software package has applications that easier the hard algebraic manipulation such as the symbolic manipulator and some specific functions for forward and inverse kinematics computation. MATLAB® functions include kinematics, trajectory generation, dynamics, and control. The toolbox also includes Simulink® models to describe the evolution of arm or mobile robot state over time for the sake of control. Regarding this chapter, functions like manipulating and converting between data types such as vectors, rotation matrices, homogeneous transformations, and twists are very important. Some important functions related to this chapter are:

- **angvec2r** that converts angle and vector orientation to a 3x3 rotation matrix;

- **angvec2tr** that converts angle and vector orientation to a 4x4 homogeneous transform;

- **syms** command that allows the definition of symbolic variables for rotation and homogeneous algebra. The symbolic manipulation allows include several commands to simplify results of matrix algebraic manipulation.

- **eul2tr (phi, theta, psi, options)** converts the Euler angles to a (4x4) homogeneous transformation matrix.

- **rpy2tr (roll, pitch, yaw, options)** command refers to the LVLH frame referred in this chapter and converts the roll-pitch-yaw angles to homogeneous transform.

There are several other commands/functions that are very useful for applications approached in this chapter and the reader is advised to see [10].

4. Conclusion

In this chapter, we have approached the forward and inverse kinematics for space robot manipulators via the D-H convention. In all cases presented here, the spacecraft nominal attitude was considered stabilized. Otherwise, it would be necessary to consider kinematics in terms of absolute position described in the LVLH or in the spacecraft fixed frame with variable attitude angles. Some examples are solved, and a related reference is provided. Al the algebraic matrix manipulations presented here were obtained by using the MATLAB® Symbolic tool box. Also, the chapter presented other features of the MATLAB® and Simulink® software related to the subject of the chapter. Despite its importance, differential kinematics is out of scope of this chapter and maybe the subject of future work.

Acknowledgements

Author details

Ijar Milagre da Fonseca[1,2*], Maurício Nacib Pontuschka[3]
and Glaydson Luiz Bertoze Lima[1]

1 Technological Institute of Aeronautics – DCTA/ITA, Sao Jose dos Campos, Brazil

2 National Institute for Space Research – DMC/INPE, Sao Jose dos Campos, Brazil

3 Pontifical Catholic University of Sao Paulo PUC-SP, Sao Paulo, Brazil

*Address all correspondence to: ijar@ita.br

IntechOpen

References

[1] Fonseca I, Mint PM. The state-of-the-art in space robotics. Journal of Physics: Conference Series. 2015;**641**:7. DOI: 10.1088/1742-6596/641/1/012025

[2] Fonseca I, Pontuschka M, Saotome O, Bainum P, Lima G. The impact of the non-inertial base motion in the operations of robotic manipulators. In: 65th International Astronautical Congress. 2014

[3] Fonseca I, Goes L, Seito N, Duarte M, Mint OE. Attitude dynamics and control of a spacecraft like a robotic manipulator when implementing on-orbit servicing. Acta Astronautica. 2017; **137**:490-497. DOI: 10.1016/j. actaastro.2016.12.020

[4] Skaar SB, Ruoff CF. Teleoperation and Robotics in Space, Progress in Astronautics and Aeronautics. Vol. 61. Washington DC: American Institute of Aeronautics and Astronautics, Inc; 1994. pp. 257-258

[5] Craig J. Introduction to Robotics Mechanics and Control. 3rd ed. New York: Pearson; 2005. pp. 62-123. DOI: 10.13140/2.1.4486.1446

[6] Spong MW, Hutchinson S, Vidyasagar M. Robot Dynamics and Control. 2nd ed. ISBN: 978-0-471-64990-8. New York: John Wiley & Sons, Inc; 2004. pp. 57-89

[7] Wie B. Space Vehicle Dynamics and Control. 2nd ed. Reston, VA: American Institute of Aeronautics and Astronautics, Inc; 2012. pp. 397-381. DOI: 10.2514/4.860119

[8] Zohar I, Ailon A, Mint RR. Mobile robot characterized by dynamic and kinematic equations and actuator dynamics: Trajectory tracking and related application. Robotics and Autonomous Systems. 2011;**59**:343-353. DOI: 10.1016/j.robot.2010.12.001

[9] Song W, Hu GM. A fast inverse kinematics algorithm for joint animation. Procedia Engineering. 2011; **24**:350-354. DOI: 10.1016/j. proeng.2011.11.2655

[10] Corke P. Robotics, Vision & Control: Fundamental Algorithms in MATLAB. 2nd ed. Springer. 2011