



Kinematics, Actuation, and Sensing Architectures for Rehabilitation and Assistive Robotics

2

Sergio D. Sierra M. , Luis Arciniegas-Mayag, Orion Ramos, Juan Maldonado, Marcela Múnera, and Carlos A. Cifuentes 

2.1 Introduction

In the process of analyzing and designing robotic technologies for rehabilitation and gait assistance, it is necessary to understand the different elements that describe and compose such a device. In this sense, this chapter presents the necessary tools to mathematically model rehabilitation devices such as lower-limb exoskeletons, social robots, and robotic walkers. Likewise, the concepts necessary to understand the actuation systems that allow the movement of these devices and the safe interaction with the users are presented. Finally, the most common sensing architectures reported in the literature are described, which allow these devices to acquire information from their internal systems, their environment, and the user.

2.2 Robotic Geometric and Kinematic Modeling

In robotics, kinematics studies the motion of a robot part concerning a reference system. In a manipulator robot, this system is usually chosen according to the task to be performed by the robot with respect to the base and the end effector. Geometric kinematic models essentially involve the relationship between the robot and its workspace, usually a Cartesian space. In this chapter the position of robot links will be considered in static situations only. To understand the complex geometry of a robot, one must add frames to the various parts of the mechanism and then describe

S. D. Sierra M. · L. Arciniegas-Mayag · O. Ramos · J. Maldonado · M. Múnera · C. A. Cifuentes (✉)

Biomedical Engineering Department of the Colombian School of Engineering Julio Garavito, Bogotá D.C., Colombia

e-mail: sergio.sierra@escuelaing.edu.co; luis.arciniegas@mail.escuelaing.edu.co; orion.ramos@mail.escuelaing.edu.co; juan.maldonado-me@mail.escuelaing.edu.co; marcela.munera@escuelaing.edu.co; carlos.cifuentes@escuelaing.edu.co

the relationships between these frames. The study of the kinematics of manipulator robots relates, among other things, to how the locations of these frames change as the mechanism articulates. In other words, either forward or inverse kinematics relates the end position of the end effector to the angles of the robot's joints.

This section will explain the difference between forward and inverse kinematics applied in robotics, develop the Denavit–Hartenberg methodology for the solution of forward kinematics, and present some examples with exoskeletons and humanoid robots that have different kinematic chains with different degrees of freedom.

2.2.1 Forward vs. Inverse Kinematics

The study of kinematics in robotics is based on calculating the parameters that define the positions of a specific part of the robot. These parameters can be joint angles or equations that depend only on the robot's dimensions. In manipulator robotics, there are two types of kinematics, forward kinematics and inverse kinematics. Each has a specific purpose and is calculated in different ways. Forward kinematics relates the final position of the kinematic chain concerning the angles of each joint. In other words, it finds the (X, Y, Z) position of the hand, head, or leg if the angles of each joint are known. On the other hand, inverse kinematics relates the angles of each joint to a desired (X_f, Y_f, Z_f) position. For example, in robots with arms, this type of kinematics is used when the robot is to point to something specific, so the inverse kinematics solution will find the necessary angles that must be placed in each motor or joint for the hand to point precisely to a defined position. Figure 2.1 shows the differences between the two types of kinematics graphically.

To solve the forward kinematics, the lengths of the kinematic chain's links must be known to use them with mathematical tools that finally model the robot's position. The most commonly used and standardized method to solve this kinematic problem is called Denavit–Hartenberg (DH) [1, 2]. This method is based on finding 4 parameters $(\theta_i, d_i, \alpha_i, a_i)$ per robot segment that relate the robot base to the final part of the kinematic chain (P_f) . Usually this type of kinematics can be expressed with the Denavit–Hartenberg Table [2].

The most commonly used methods to find the inverse kinematics solution are the geometric method and the screw method. However, the second depends on

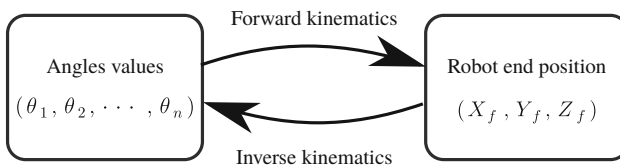


Fig. 2.1 Scheme that relates the angles of the joints and the positions of the links with the forward and inverse kinematics. If the angles of the joints are known, the final position can be calculated. If the final position is known, the angles required at each joint can be calculated

the previous calculation of the Denavit matrices [3]. The geometric method uses trigonometric rules to find angles of the triangles formed by the kinematic chain's links in a specific plane. Usually, functions such as arctan and decompositions of the laws of cosines or sines are used. The geometric method's solution will be different for each robot configuration, so it is more complex to calculate than forward kinematics.

2.2.2 Denavit–Hartenberg Convention

The relationship between the joints and the robot workspace is realized via geometric modeling. The most convenient way to perform this process is denoted as the Denavit–Hartenberg (DH) method. This approach focuses on forward kinematic computation only. The goal is to calculate the position and orientation of the end effector relative to the base as a function of the joint variables. This method consists of representing the robot's kinematic chain using a set of bodies connected by joints. These bodies are called links or segments. The joints or degrees of freedom (DoFs) are those that connect two links. The DH convention expresses the rotation and translation using a single homogeneous matrix of one DoF of the kinematic chain in mathematical terms.

The structure of the DH matrix is generally composed of the rotation and translation of the axes involved in the degree of freedom and the distance between them. The matrix will always have dimensions of 4×4 as follows:

$$H_n = \begin{bmatrix} R_{n(3 \times 3)} & t_{n(3 \times 1)} \\ 0_{1 \times 2} & 1 \end{bmatrix}, \quad (2.1)$$

where $R_{n(3 \times 3)}$ is the rotation matrix (3×3) that considers the rotation in x -, y -, and z -axis and $t_{n(3 \times 1)}$ is the translation vector with size (3×1), where each position of the vector is a translation on the x -, y - and z -axis.

The DH method lists the links starting from the arm's fixed base, called link 0. The first moving body is the link 1 and successively until reaching the free end of the arm (end effector), which is link n . The purpose of this is to locate the frames in each joint in the most appropriate way to obtain the DH parameters. The location of the frames according to the method can be summarized in the following six steps:

1. Identify the joint axes and mark the lines on them. The following steps consider two of these adjacent lines (on the i and $i + 1$ axes).
2. Identify the common perpendicular line between them, or the point of intersection. This point will be the origin of the frame of the i_n link.
3. Assign the Z_i -axis in a manner that it points to the joint axis.
4. Assign the X_i -axis to be normal to the plane of Z_i at its intersection point.
5. Assign the Y_i -axis to complete the coordinate system of this frame according to the right-hand rule.
6. Perform these steps for the next joint until the joint n is reached.

Parameter	Name	Description
θ_i	Joint Angle	$X_i \xrightarrow[\text{@}Z_i]{\text{rotation,ccw}} X_{i+1}$
d_i	Joint Offset	$X_i \xrightarrow[\text{@}Z_i]{\text{distance}} X_{i+1}$
α_i	Twist Angle	$Z_i \xrightarrow[\text{@}X_i]{\text{rotation,ccw}} Z_{i+1}$
a_i	Link Length	$Z_i \xrightarrow[\text{@}X_i]{\text{distance}} Z_{i+1}$

Fig. 2.2 Description and calculation of the four DH parameters according to the axes and frames located in the system

After performing the location of all the frames in the kinematic chain, the four DH parameters (θ_i , d_i , α_i , a_i) are obtained for each joint. The way to obtain these parameters can be seen in Fig. 2.2. This way of obtaining the parameters is only functional if the frames were correctly placed according to the previous steps.

The angle θ_i will be the rotation in the Z_i -axis between the X axes of the links of the degree of freedom. In the particular case in which this angle depends on an actuator (motor), it is expressed as a variable angle on which the end effector's final position will depend. The parameter d_i is the distance between the two links of the degree of freedom around the Z_i -axis. If the actuator (linear motor) modifies this distance, it will be a variable parameter and not a constant value. The twist angle α_i is the angle formed between the Z axes of the degree of freedom around the X_i -axis. This value is usually constant, but as in the previous cases, if any actuator modifies it, it should be taken as a variable angle. Finally, the distance a_i is defined by the space generated between the Z_i and Z_{i+1} axes through the X_{i+1} -axis.

By identifying these parameters for each degree of freedom, it is possible to relate the base to the end effector depending on the variable angles and distances (which are modified by actuators). In other words, it will be possible to know the position and orientation of the end of the kinematic chain if the actual actuator's position that modifies angles or distances is known.

Based on this standard procedure to obtain the DH parameters, the following sections of the chapter will calculate the forward and inverse kinematics of the AGoRA lower-limb exoskeleton [4, 5] that has only two degrees of freedom in the same plane, and then calculate the kinematics of social robots where the motions are in the 3D Cartesian space. Finally, an approximation of the forward and inverse kinematics in mobile robots will be made.

2.2.3 Modeling Lower-Limb Exoskeletons

Each lower-limb exoskeletons presented in the literature show various kinematic models to mimic the human movements, where this parameter is estimated by applying the manipulator robot’s concepts. The first step to define the kinematic parameter is presented through the DH convention explained in Sect. 2.2.2 where the joint angle θ_i , joint offset d_i , twist angle α_i , and the link length a_i are established. As an example, the *AGoRA* lower-limb exoskeleton is designed using these manipulator concepts. In this case, the *AGoRA* exoskeleton can express showing the DH convention of one limb. In this sense, the *AGoRA* exoskeleton is a 2 DoF robot where the joint angle is expressed by q_1 and q_2 , l_1 , and l_2 express the link lengths. Figure 2.3 shows the *AGoRA* exoskeleton identifying some DH parameters. Taking into account the parameters shown is defined the DH convention expressed in Table 2.1.

The DH convention shown in the table presents the *AGoRA* exoskeleton as a manipulator robot with 2 degrees of freedom (DoFs) focused on the movement generation in the sagittal plane. This exoskeleton is focused on the hip and knee joints of the patient’s right limb.

In this sense, the *AGoRA* exoskeleton is presented in two homogeneous matrices shown in Eqs. 2.2 and 2.3

$$H_1 = \begin{bmatrix} \sin(q_1) & \cos(q_1) & 0 & l_1 \sin(q_1) \\ -\cos(q_1) & \sin(q_1) & 0 & -l_1 \cos(q_1) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.2}$$

Fig. 2.3 *AGoRA* exoskeleton DH convention; (a) *AGoRA* exoskeleton’s sagittal plane; (b) *AGoRA* exoskeleton’s frontal plane

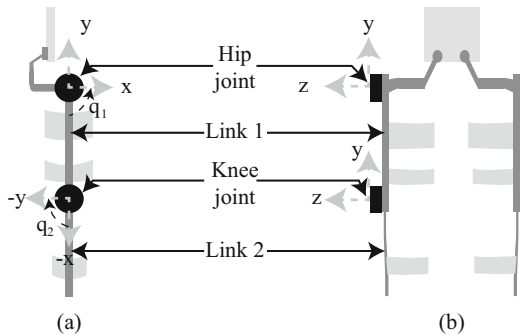


Table 2.1 *AGoRA* exoskeleton kinematic definition using the DH convention; θ_i is the revolute joint variable; d_i is the distance from the origin exoskeleton frame to the x -axis; a_i presents the link’s joint; α_i represents the x -axis rotation

Joint	θ_i	d_i	a_i	α
1	q_1	0	l_1	0
2	q_2	0	l_2	0

$$H_2 = \begin{bmatrix} \sin(q_2) & \cos(q_2) & 0 & l_2 \sin(q_2) \\ -\cos(q_2) & \sin(q_2) & 0 & -l_2 \cos(q_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.3)$$

The *AGoRA* exoskeleton rotation and translation of each axis are described in three expressions. The hip and knee joint's rotation is presented in the z -axis. Consequently, each rotation has a translation value related to the link's length operated depending on the joint's rotation angle (q_1 and q_2). Equation 2.2 shows the rotation and translation of the shaft from joint 1 to joint 2. Equation 2.3 shows the rotation and translation of the shaft from joint 2 to the end effector. Finally, the multiplication of the matrix H_1 and H_2 (Eq. 2.4) provides information about the rotation and translation movements from the reference coordinate axis to the end-effector reference axis (Eq. 2.5). The values obtained in each of the equations are called homogeneous matrix transformations.

$$H_0^2 = H_1 H_2 \quad (2.4)$$

$$H_0^2 = \begin{bmatrix} -\cos(q_1 + q_2) & \sin(q_1 + q_2) & 0 & l_1 \sin(q_1) - l_2 \cos(q_1 + q_2) \\ -\sin(q_1 + q_2) & -\cos(q_1 + q_2) & 0 & -l_1 \cos(q_1) - l_2 \sin(q_1 + q_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.5)$$

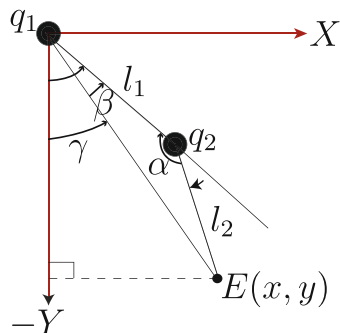
Finally, the movement generation of an exoskeleton may be defined for each link geometrically. In that order, the forward kinematics of a manipulator robot provides end-effector coordinates information depending on the rotational information of each of the robot's links. In lower-limb exoskeletons, the end effector equals the distal location of the exoskeleton structure from the reference axis. For the *AGoRA* lower-limb exoskeleton, the forward kinematics of each link is expressed as follows:

$$Link_{1(x,y)} = \begin{bmatrix} l_{c1} \sin q_1 \\ -l_{c1} \cos q_1 \end{bmatrix} \quad (2.6)$$

$$Link_{2(x,y)} = \begin{bmatrix} l_1 \sin q_1 & l_{c2} \sin(q_1 + q_2) \\ -l_1 \cos q_1 & -l_{c2} \cos(q_1 + q_2) \end{bmatrix}. \quad (2.7)$$

Equation 2.6 shows the hip joint forward kinematic, where l_{c1} equals to the link 1 center of mass, and q_1 is the hip joint rotation angle. Equation 2.7 shows the knee joint forward kinematic, where l_1 is the link 1 length, and l_{c2} is the link 2 center of mass. Finally, q_2 is the knee joint rotation angle.

Fig. 2.4 Lower-limb *AGoRA* exoskeleton right limb, where l_1 and l_2 equal to the link 1 length and link 2 lengths, respectively; $E(x, y)$ is the end-effector position; q_1 and q_2 equal to the hip angle rotation and knee angle rotation, respectively



On the other hand, using the end-effector coordinates of the robot, it is possible to estimate the angles of each joint, which is named the inverse kinematics model. The estimation of q_1 and q_2 can be performed in two ways. The first one consists of solving these variables from Eqs. 2.6 and 2.7. The second method is performed geometrically. Figure 2.4 shows the development of this second method by positioning the *AGoRA* exoskeleton in a Cartesian plane and a point (x, y) where the end effector of the knee joint is located.

In the estimation of the q_1 and l_2 values, q_2 is calculated as follows:

$$r^2 = x^2 + y^2 \quad (2.8)$$

$$r^2 = l_1^2 + l_2^2 - 2l_1l_2 \cos \alpha \quad (2.9)$$

$$q_2 = \pi - \alpha. \quad (2.10)$$

Using Eq. 2.8 in Eq. 2.9 obtains $\cos \alpha$ value:

$$\cos \alpha = \frac{l_1^2 + l_2^2 - x^2 - y^2}{2l_1l_2}. \quad (2.11)$$

Applying Eq. 2.11 in Eq. 2.10 estimates the q_2 value:

$$q_2 = \arccos \frac{x^2 + y^2 - l_1 - l_2}{2l_1l_2}. \quad (2.12)$$

The estimation of q_1 value is performed as follows:

$$\gamma = q_1 - \beta, \quad (2.13)$$

where the $\tan \beta$ equals to

$$\tan \beta = \frac{-l_2 \sin q_2}{l_1 + l_2 \cos q_2} \quad (2.14)$$

$$\tan \gamma = \frac{x}{y}. \quad (2.15)$$

Applying tan in Eq. 2.13 replaces Eqs. 2.14 and 2.15 obtaining the following expression:

$$q_1 = \arctan\left(\frac{x}{y}\right) - \arctan\left(\frac{l_2 \sin q_2}{l_1 + l_2 \cos q_2}\right). \quad (2.16)$$

2.2.4 Modeling Social Robots

Once the kinematics of a single plane of motion device such as the *AGoRA* lower-limb exoskeleton is known, the kinematics of each of the moving parts of two social robots used for human–robot interaction will now be analyzed. The moving parts or kinematic chains to be analyzed of these robots in this section are defined as head, upper limb, and lower limb. The kinematic chain of the head only has one degree of freedom in the *CASTOR* robot [6, 7], the upper limb of the *CASTOR* robot has 3 degrees of freedom in different planes, and finally the lower limb of the *NAO* robot has 6 degrees of freedom. The following section will describe each of the degrees of freedom and elements needed to calculate the forward and inverse kinematics of the mentioned kinematic chains. The general architecture of the open-source *CASTOR* robot used for autism therapies and the *NAO* commercial robot from SoftBank Robotics [8] will be defined. It should be noted that a DoF is not necessarily located where the actuators or motors are placed, but is the point of motion as are the joints in the human body.

2.2.4.1 Modeling the Head and Upper Limb of the *CASTOR* Robot

In the case of the *CASTOR* robot, it has 14 degrees of freedom distributed into 12 active and 2 passive degrees of freedom. This means that this robot can change the position of the links of its system by using 12 actuators and has 2 joints to receive impacts or reject external disturbances. This system uses 7 servomotors (AX12, Dynamixel, Seoul Korea) to the kinematic chains of the head and upper limbs. The other 5 active joints deal with the gestures of the face. The definition of each degree of freedom of the robot is described as follows: 1 DOF for each elbow (i.e., flexo-extension movement), 2 DOFs per shoulder (i.e., flexo-extension and abduction movements), 1 DOF for head rotation movement, and 5 DOFs for facial expressions. Besides allowing deformation in the huggable structure and rejection of external stimulations, the robot incorporates 2 passive DOFs. The relevant joints for the kinematic analysis can be seen in Fig. 2.5.

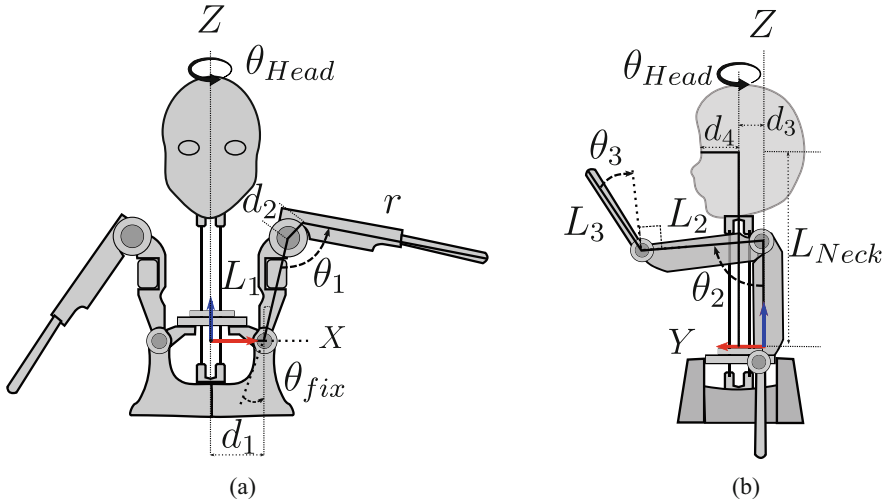


Fig. 2.5 Description of the angular and longitudinal variables involved in the *CASTOR* robot used to solve the forward and inverse kinematics. **(a)** Front view of the robot. **(b)** Side view of the robot

Table 2.2 DH parameters for the kinematic chain of the *CASTOR* robot head

Matrix _{<i>i</i>}	θ_i	d_i	a_i	α_i
1	0	0	d_3	0
2	θ_{Head}	L_{Neck}	0	0
3	0	0	d_4	0

By dividing this robot into two kinematic chains, head and upper limbs, it can be identified that the head part has one degree of freedom (θ_{head}) and the upper limbs have 3 degrees of freedom for each arm ($\theta_1, \theta_2, \theta_3$). To complement the general architecture of this robot, the system’s overall dimensions must be defined. The *CASTOR* robot is 50 cm from base to head and 35.4 cm wide. Each arm of the robot has a total length of 35.6 cm [7].

2.2.4.2 The Head Kinematic Chain

As shown in Fig. 2.5 the kinematic chain of the head of the *CASTOR* robot only depends on one joint θ_{head} and the distances that are necessary to transport the base point (X_0, Y_0, Z_0) to the endpoint of the kinematic chain (X_f, Y_f, X_f), which in this case would be the eyes of the robot. Following the standard steps to complete the DH table [2], coordinate axes are established for each section of the robot neck kinematic chain, as seen in Fig. 2.6a. As can be seen, only the dimensions of the neck (L_{Neck}), the distance from the center to the neck (d_3), the joint rotation angle (θ_{Head}), and the distance from the neck to the eyes (d_4) are needed. From these coordinate axes, the DH steps are applied to complete Table 2.2.

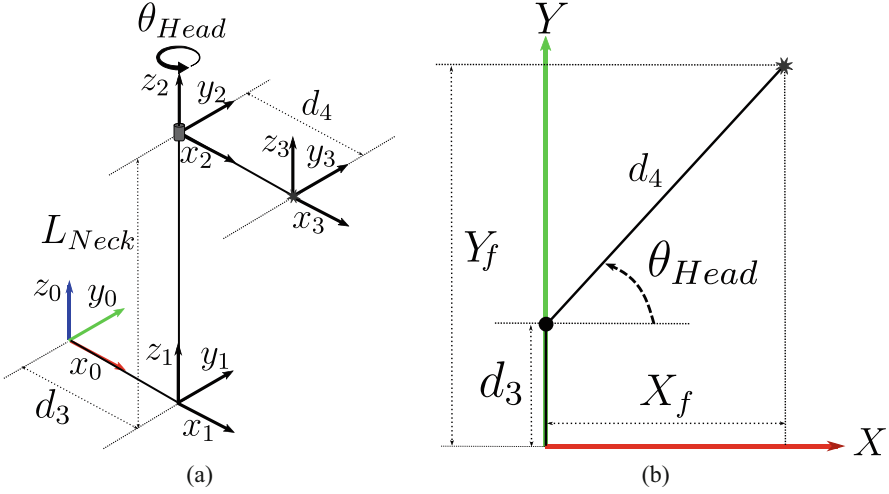


Fig. 2.6 (a) Kinematic chain of the *CASTOR* robot head with the coordinate axes located according to the Denavit–Hartenberg methodology. (b) View of the X – Y plane where the angle θ_{Head} is observed and calculated based on the triangle formed by the dimensions of the final position (X_f, Y_f) where the robot head should point. This calculation considers the separation distance d_3 between the base of the robot and the rotation point of the neck, since it can be different from zero

Once the parameters of Denavit have been obtained, the matrices of each row of the table must be calculated to relate the base of the robot with its endpoint. For each of the rows of Table 2.2, the matrix DH_i must be expressed according to Eq. 2.17, and multiply all these matrices. This last resulting matrix will be the one that relates all the spatial transformations that are necessary to apply to the origin to reach the final point.

$$DH_i = \begin{bmatrix} c_{\theta_i} & -s_{\theta_i}c_{\alpha_i} & s_{\theta_i}s_{\alpha_i} & a_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i}c_{\alpha_i} & -c_{\theta_i}s_{\alpha_i} & a_i s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.17)$$

The three necessary matrices to perform the forward kinematics calculation are shown below in the *CASTOR* robot head, where Matrix 1 is defined as DH_1 , Matrix 2 as DH_2 , and Matrix 3 as DH_3 .

$$DH_1 = \begin{bmatrix} 1 & 0 & 0 & d_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.18)$$

$$DH_2 = \begin{bmatrix} \cos(\theta_{Head}) & -\sin(\theta_{Head}) & 0 & 0 \\ \sin(\theta_{Head}) & \cos(\theta_{Head}) & 0 & 0 \\ 0 & 0 & 1 & L_{Neck} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.19)$$

$$DH_3 = \begin{bmatrix} 1 & 0 & 0 & d_4 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.20)$$

Therefore, the forward kinematics of the *CASTOR* robot head will be defined by the result of the matrix multiplication of R_t and P_0 (see Eq. 2.21), where R_t is the multiplication of the 3 matrices (Eqs. 2.18, 2.19, and 2.20) found in Table 2.2 and P_0 is the (X, Y, Z) coordinate vector of the robot base. The R_t matrix obtained for the *CASTOR* robot is shown in Eq. 2.22

$$P_f = R_t P_0 = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}, \quad (2.21)$$

where

$$R_t = DH_1 \cdot DH_2 \cdot DH_3, \quad P_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$$R_t = \begin{bmatrix} \cos(\theta_{Head}) & -\sin(\theta_{Head}) & 0 & d_3 + d_4 \cos(\theta_{Head}) \\ \sin(\theta_{Head}) & \cos(\theta_{Head}) & 0 & d_4 \sin(\theta_{Head}) \\ 0 & 0 & 1 & L_{Neck} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.22)$$

Solving Eq. 2.21 with the matrix already calculated allows obtaining the three equations that relate the final coordinates x_f, y_f, z_f with the angles of the kinematic chain, which means that the forward kinematics has been solved satisfactorily. In Eq. 2.23 the mathematical expressions of the solution of the forward kinematics of the *CASTOR* social robot neck can be seen.

$$\begin{bmatrix} x_f \\ y_f \\ z_f \\ 1 \end{bmatrix} = \begin{bmatrix} d_3 + d_4 \cos(\theta_{Head}) \\ d_4 \sin(\theta_{Head}) \\ L_{Neck} \\ 1 \end{bmatrix}. \quad (2.23)$$

The inverse kinematic calculation of the *CASTOR* robot head is performed by the geometrical method from the known robot dimensions. For the solution to this problem, it is necessary to define an endpoint for the kinematic chain. In this case, it would be the robots' view at the point (X_f, Y_f, Z_f) . As this kinematic chain only consists of 1 DOF, it will only be necessary to find the equation that establishes the relation between the final point (X_f, Y_f, Z_f) and the angle that must be positioned in the neck θ_{Head} .

The top view is shown in Fig. 2.6b. The angle θ_{Head} can be identified in the triangle formed by dimensions X_f, Y_f , and d_3 . The way to describe the angle by the geometric method, in this case, is to employ the tangent function since the dimensions of the opposite cathetus to the angle θ_{Head} are known. The mathematical expression that solves the inverse kinematics of the neck based on the known robot parameters is seen in Eq. 2.24.

$$\theta_{Head} = \tan^{-1} \left(\frac{X_f - d_3}{Y_f} \right). \quad (2.24)$$

2.2.4.3 The Upper-Limb Kinematic Chain

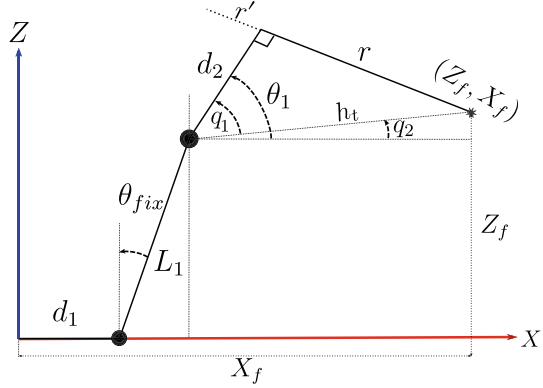
For the *CASTOR* robot arm, the kinematic calculation is more complex due to the number of degrees of freedom and their forms of motion. In Fig. 2.5 it can be seen how this kinematic chain depends on the three angles $\theta_1, \theta_2, \theta_3$, the distances L_1, L_2, L_3, d_2, d_4 , and the fixed angle of the passive degree of freedom of the robot θ_{fix} . Using the procedure for the DH parameter finding, Table 2.3 is made, which summarizes the forward kinematics of the left arm of the *CASTOR* robot.

Using these values the relationship between the angles $(\theta_1, \theta_2, \theta_3)$ and the robot hand, which is the endpoint of the arm, can be found. The final equation is given by Eq. 2.21, where R_f is the multiplication of the 7 DH matrices found in Table 2.3.

Table 2.3 DH parameters of the kinematic chain of the *CASTOR* robot arm obtained only by the known dimensions and angles of the robot

Matrix _{<i>i</i>}	θ_i	d_i	a_i	α_i
1	0	0	d_1	0
2	90	0	0	θ_{fix}
3	0	L_1	0	$180 - (\theta_{fix} + \theta_1)$
4	-90	0	$-d_2$	0
5	0	0	0	$-\theta_2$
6	0	L_2	0	$-(90 + \theta_3)$
7	0	L_3	0	0

Fig. 2.7 $X - Z$ plane where the angle θ_1 and the actual dimensions needed to determine inverse kinematics equation of this first angle is seen



For the inverse kinematic calculation of this part of the robot it used the geometrical method starting from the known dimensions of the arm and the defined final position (X_f, Y_f, Z_f) to which the robot hand is to be pointed (see Fig. 2.5).

To calculate the first angle of the arm θ_1 the robot must be viewed from the $X - Z$ plane as seen in Fig. 2.7, since in this perspective all dimensions are known or can be calculated. As can be seen, the angle θ_1 is the sum of the two supplementary angles q_1 and q_2 . So the mathematical part will be based on determining these angles based on the known dimensions.

The equation that solves the supplementary angle q_2 is seen in Eq. 2.25 and is based on the solution of the small right triangle that is formed with one leg on the X -axis and the other on the Z -axis. Simultaneously, the solution of the supplementary angle q_1 is calculated with the hypotenuse h_t of the small right triangle and the known leg dimension d_2 (see Eq. 2.26).

$$q_2 = \tan^{-1} \left(\frac{Z_f - L_1 \cos(\theta_{fix})}{X_f - d_1 - L_1 \sin(\theta_{fix})} \right) \quad (2.25)$$

$$q_1 = \cos^{-1} \left(\frac{d_2}{h_t} \right), \quad (2.26)$$

where

$$h_t = \sqrt{(X_f - d_1 - L_1 \sin(\theta_{fix}))^2 + (Z_f - L_1 \cos(\theta_{fix}))^2}.$$

As defined, the angle θ_1 is the sum of the two expressions of Eqs. 2.25 and 2.26. This expression, as can be seen, only depends on the known parameters based on the dimensions of the robot and the distances of the endpoint coordinates (X_f, Y_f, Z_f) . Equation 2.27 shows the solution of angle θ_1 without mathematical reductions.

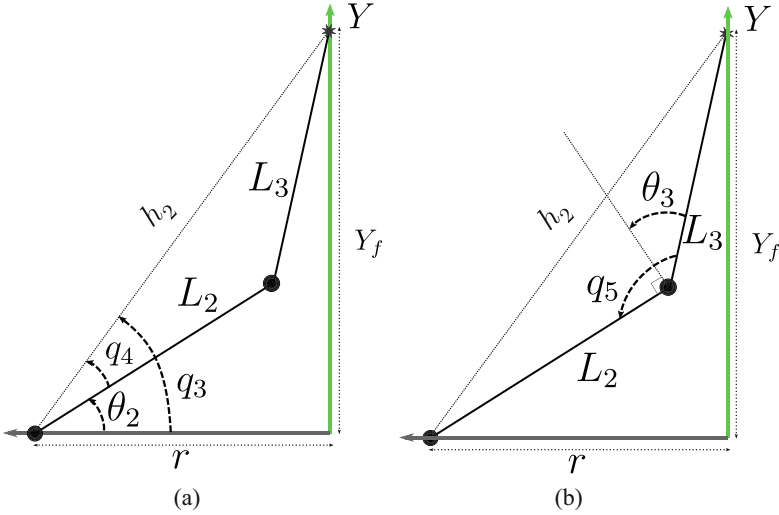


Fig. 2.8 The auxiliary $Y-r$ plane where the angles θ_2 θ_3 and the links L_2 and L_3 can be observed in their actual dimension necessary to determine the inverse kinematics equations of the angles of the joints 2 and 3. Part (a) shows the auxiliary variables for calculating the angle θ_2 , and part (b) shows the auxiliary variables necessary to perform the calculation of the angle θ_3

$$\theta_1 = \cos^{-1} \left(\frac{d_2}{h_t} \right) + \tan^{-1} \left(\frac{Z_f - L_1 \cos(\theta_{fix})}{X_f - d_1 - L_1 \sin(\theta_{fix})} \right). \quad (2.27)$$

The angles θ_2 and θ_3 are calculated in the same way. In this case, the perspective of the $Y-r'$ plane as shown in Fig. 2.8 is used. This plane is the only one that allows seeing the dimensions of the links L_2 and L_3 in their actual size and where it is easier to find the geometric expressions to determine the angles of the remaining joints. The angle θ_2 , as seen in Fig. 2.8a, is defined as the difference between the auxiliary angles q_3 and q_4 , so it is necessary to find the geometric expressions from the triangles formed in the figure that describe these angles with the known dimensions.

The expression that defined the auxiliary angle q_3 (see Eq. 2.28) is calculated from the largest right triangle in Fig. 2.8a formed by the legs Y_f and r . Since the outer triangle is not a right triangle, the sine or cosine rule must be used to find the expression for the auxiliary angle q_4 .

$$q_3 = \tan^{-1} \left(\frac{Y_f}{r} \right), \quad (2.28)$$

where

$$r = h_t \sin(q_1).$$

Since L_1 and L_2 are known from the dimensions of the robot and h_2 can be calculated since it is the hypotenuse of the right triangle formed by Y_f and r (see Fig. 2.8a), the cosine rule is used to relate the angle q_4 to the 3 dimensions of the sides of the triangle. In Eq. 2.29, the angle q_4 is shown cleared from the cosine rule.

$$q_4 = \cos^{-1} \left(\frac{L_3^2 - L_2^2 - h_2^2}{-2L_2h_2} \right), \quad (2.29)$$

where

$$h_2 = \sqrt{Y_f^2 + r^2}$$

Knowing the expressions of the supplementary angles of Eqs. 2.28 and 2.29, it is possible to express the angle θ_2 with only terms of known dimensions or final point coordinates. In Eq. 2.30, the inverse kinematics solution for the angle of joint 2 of the *CASTOR* robot arm is seen.

$$\theta_2 = \tan^{-1} \left(\frac{Y_f}{r} \right) - \cos^{-1} \left(\frac{L_3^2 - L_2^2 - h_2^2}{-2L_2h_2} \right). \quad (2.30)$$

The angle θ_3 is calculated in the same way, and the same plane as angle θ_2 (see Fig. 2.8b), but in this case, as the initial position of the robot forearm is at 90 degrees from the link L_2 (defined by the initial position), the general expression for this angle is seen in Eq. 2.31.

$$\theta_3 = 90 - q_5. \quad (2.31)$$

Computing the auxiliary angle q_5 employing the cosine rule (See Eq. 2.32) it is easy to complete the solution of the inverse kinematics of angle θ_3 of the kinematic chain of the *CASTOR* robot arm. In Eq. 2.33, the general solution of the last angle is presented.

$$q_5 = \cos^{-1} \left(\frac{h_2^2 - L_2^2 - L_3^2}{-2L_2L_3} \right) \quad (2.32)$$

$$\theta_3 = 90 - \cos^{-1} \left(\frac{h_2^2 - L_2^2 - L_3^2}{-2L_2L_3} \right). \quad (2.33)$$

Therefore, the equations modeling the inverse kinematics of the robot arm *CASTOR* can be summarized in Eq. 2.34. These inverse kinematics expressions and the forward kinematics calculations were used to develop a case study in which this

robot helps children with autism through pre-configured gestures such as waving or pointing to their body parts to improve interaction with the environment [7].

$$\begin{aligned}
 \theta_1 &= \cos^{-1} \left(\frac{d_2}{h_t} \right) + \tan^{-1} \left(\frac{Z_f - L_1 \cos(\theta_{fix})}{X_f - d_1 - L_1 \sin(\theta_{fix})} \right) \\
 \theta_2 &= \tan^{-1} \left(\frac{Y_f}{r} \right) - \cos^{-1} \left(\frac{L_3^2 - L_2^2 - h_2^2}{-2L_2h_2} \right) \\
 \theta_3 &= 90 - \cos^{-1} \left(\frac{h_2^2 - L_2^2 - L_3^2}{-2L_2L_3} \right).
 \end{aligned} \tag{2.34}$$

Knowing the methodology of the kinematic calculation of the *CASTOR* robot, it is possible to perform analysis of more complex robots kinematically speaking. In the following section the kinematics calculation of the lower limb of the social robot *NAO* will be developed.

2.2.4.4 Modeling the Lower Limb of the *NAO* Robot

This social robot has 25 degrees of freedom to perform different tasks and movements divided as follows: in the head it has 2 DOFs, in each upper limb, this robot has 6 DOFs, and in the lower part it has 11 DOFs in total, distributed in each leg and the pelvis [9]. In this case, the 25 degrees of freedom of this robot are active, which means an independent motor modifies the system position for each joint. The movements of each degree of freedom are defined as follows: each arm has 2 DOFs at the shoulder, 2 DOFs at the elbow, 1 DOF at the wrist, and 1 additional DOF for the handgrip. The 2 DOFs on the head allow it to rotate about the yaw and pitch axes. For the lower kinematic chain, each leg has 2 DOFs at the ankle, 1 DOF at the knee, and 2 DOFs at the hip. The pelvis has a unique mechanism composed of two joints attached to each part of the hip. These joints are rotated at 45, facilitating control and reducing the system 1 degree of freedom less than other commercial robots [10]. Figure 2.9 shows the degrees of freedom (Fig. 2.9a) and the kinematic leg chain of the *NAO* robot (Fig. 2.9b).

As shown in Fig. 2.9b, the kinematic chain of the lower limbs consists of 6 DOFs that can be modified to adjust the position of each leg ($\theta_1 \dots \theta_6$). Overall this robot has a height of 57.3 cm, a width of 27.5 cm, and each leg has a total length of 24.8 cm [8]. To finalize the calculations of the selected kinematic chains, it is necessary to develop the forward and inverse kinematics of the *NAO* robot leg. As already mentioned, this robot has 6 degrees of freedom that must be solved by the methods presented above. The solution of the kinematics is more complex and requires more advanced mathematics than the previous cases.

Based on the kinematic chain of the leg of this robot (see Fig. 2.9b), the DH table is generated, which collects the parameters that solve the forward kinematics of the lower limb. Table 2.4 presents 9 DH matrices that depend only on the dimensions of the leg and the angles of the motors.

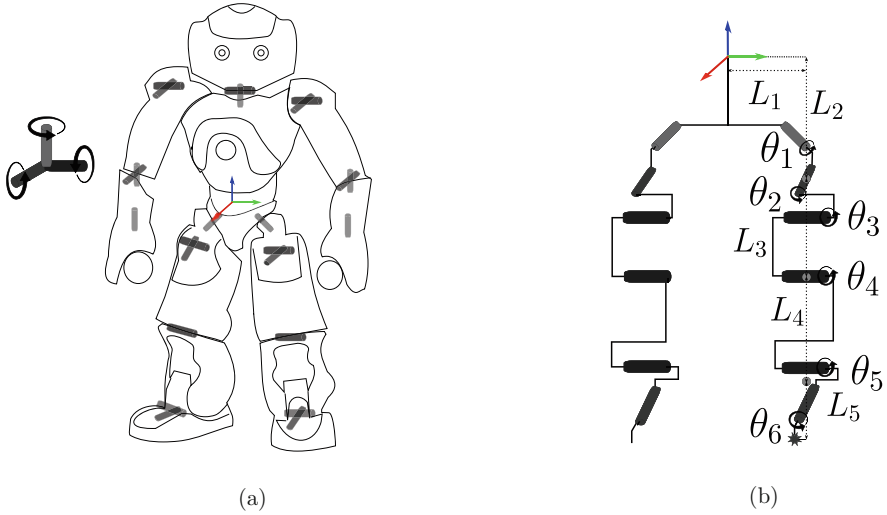


Fig. 2.9 The location of each degree of freedom of the *NAO* robot is shown in part (a). Part (b) shows the dimensions and angles involved in the kinematic chain of the robot leg

Table 2.4 DH parameters of the *NAO* robot leg kinematic chain obtained with only the known robot dimensions and known angles

Matrix _{<i>i</i>}	θ_i	d_i	a_i	α_i
1	90	0	0	45
2	90	L_1	0	180
3	θ_1	0	0	-90
4	$\theta_2 + 135$	0	0	-90
5	θ_3	0	L_2	0
6	θ_4	0	L_3	0
7	θ_5	0	0	90
8	θ_6	0	L_4	90
9	0	L_5	0	90

The solution of the inverse kinematics of the *NAO* robot leg is considered a high-level mathematical problem given that this 6 DOF kinematic chain generates a non-linear system that is quite complex to solve. To facilitate the calculations of this problem initially matrix transformations based on the forward kinematics are performed to reduce the complexity of the math. The complete calculation and solution of this kinematic chain can be found in [9, 11].

These same concepts of forward and inverse kinematics can be applied in mobile robotics, which aims to find the speeds of each wheel to reach a specific point (inverse kinematics) or according to the configuration of the moving wheels and their speeds to find out where the robot will reach in an estimated time (forward kinematics). The following section will explain how the modeling is performed in mobile robots.

2.2.5 Modeling Mobile Robots

As presented in previous sections, one of the main characteristics of robotic arms or manipulators, and thus crucial difference with mobile robots, is that they are fixed to a specific point and usually comprise of a single chain of actuated links [12]. In this sense, unlike the robotic arms in social robots or exoskeletons, which can move only in a specific workspace, mobile robots are capable of moving around freely and autonomously within a predefined environment [13]. This capability makes the mobile robots suitable for several applications, including gait assistance and rehabilitation.

Before delving into the modeling of these robots, it should be mentioned that the term "*mobile robots*" covers a wide variety of robots, including: (i) ground robots, where *wheeled mobile robots* and *legged mobile robots* are distinguished, (ii) aerial robots or *unmanned aerial vehicles*, (iii) aquatic robots or *autonomous underwater vehicles*, and (iv) hybrid robots, where mobile robots are equipped with one or more manipulators [13].

For the purpose of this book, the focus will be on wheeled mobile robots, since rehabilitation robots such as robotic wheelchairs, smart canes, and smart walkers can be labeled with this category. In general, these types of rehabilitation devices exhibit a locomotion configuration that is based on wheeled mechanical structures. The wheeled mobile robots are prevalent in both the industrial and rehabilitation concepts, given their low mechanical complexity and efficient energy consumption. In this regard, the following sections briefly present the basic concepts of kinematics and locomotion, focusing on smart walkers.

2.2.5.1 Wheeled Locomotion

The wheel is the most popular locomotion mechanism in mobile robotics, and it is also in robotic walkers. It has been demonstrated that implementing wheels provides outstanding efficiencies and simple mechanical structures [12]. Like most wheeled robots, the smart walkers are designed so that all wheels are always in ground contact. Thus, the smart walkers commonly include from three to four wheels to guarantee sufficient stable balance. Considering that patients frequently use smart walkers in clinical or indoor scenarios, no suspension systems are demanded. However, when interacting in uneven terrains, a suspension system might be required to maintain safe ground contact [12].

Considering that balance is rarely an issue in wheeled mobile robots, the focus is often on traction and stability, maneuverability, and control [12]. In this way, the design problem is commonly tackled by proposing an appropriate wheel design. Mainly, Page et al. reported that six types of wheels are often found in smart walkers. These wheel classes include: (i) fixed wheels, (ii) centered orientable wheels, (iii) off-centered wheels, (iv) Swedish wheels, (v) spherical wheels, and (vi) active split offset casters (ASOC) [14]. This report also stated that most of the smart walkers implement fixed wheels and off-centered wheels (i.e., caster wheels). The fixed wheels are commonly linked to the propulsion system, and the caster wheels provide

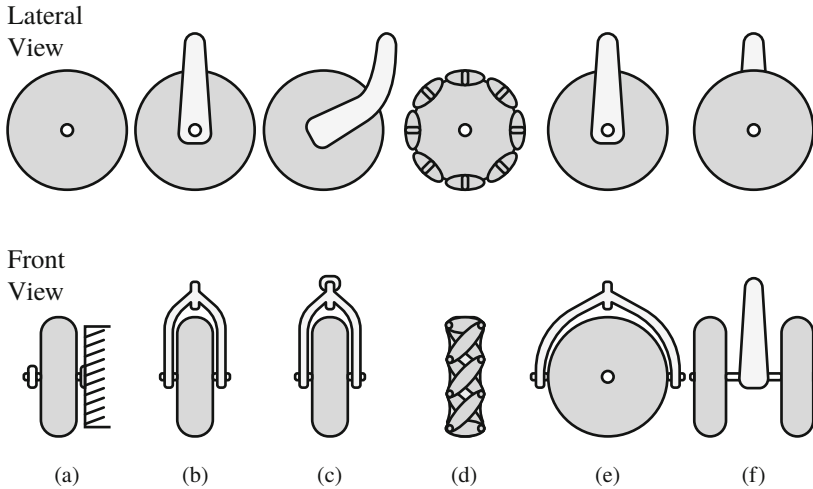


Fig. 2.10 Illustration of the primary wheel classes implemented in smart walkers. (a) Fixed wheel. (b) Centered orientable wheel or standard wheel. (c) Off-centered wheel or caster wheel. (d) Swedish wheel. (e) Spherical wheel. (f) Active split offset caster (ASOC) wheel

stability. Figure 2.10 shows these types of wheels, and a more detailed description of each wheel class can be found in [12, 14].

2.2.5.2 Wheel Configurations in Smart Walkers

Although the selection of the wheel's design is an essential issue, the geometrical arrangement and number of wheels attached to the robot's structure are directly related to the maneuverability, controllability, and stability of the platform. In general, any wheel configuration can be defined by the following elements: (i) the number and type of wheels, (ii) the wheels arrangement, and (iii) the locomotion type or actuated wheels. For instance, a particular robot might have two motorized fixed wheels in the rear and two caster wheels in front, whereas another robot might have two free wheels in the rear and one steered traction wheel in front.

As presented in [12, 13], rolling vehicles in industrial robotic applications can exhibit a wide range of wheel configurations, varying the elements mentioned above (i.e., type, number, and arrangement of wheels). Similarly, in [14], the authors reported the most common configurations in smart walkers. In general, the smart walkers can exhibit the following configurations:

- Two motorized fixed wheels in the rear and one free caster wheel in front [15]
- Two fixed wheels in the rear and one motorized caster wheel in front [16]
- Three motorized omnidirectional wheels [17]
- Two fixed wheels in the rear and two motorized and steered wheels in front [18, 19]
- Two motorized wheels in the rear and two free caster wheels in front [20]

- Two fixed wheels in the rear and two free caster wheels in front (i.e., passive device) [21]
- Two free caster wheels in the rear, two motorized wheels in the middle, and two free caster wheels in front [22]

2.2.5.3 Wheel Drive Types

The configurations mentioned above can also be classified in terms of the drive system that powers the wheels' locomotion. In general, wheeled mobile robots can be configured in six different drive types: (i) differential drive, (ii) tricycle, (iii) omnidirectional, (iv) synchro drive, (v) Ackerman steering, and (vi) skid steering [13]. For simplicity purposes, the focus in this book will be on the drive types that are used in the majority of smart walkers, which are: (i) differential drive, (ii) tricycle, and (iii) Ackerman steering.

- **Differential Drive:** This type of drive system consists of two fixed motorized wheels mounted on the left and right sides of the robot platform. The two wheels can be placed whether in front or in rear and are independently driven. To guarantee balance and stability, one, two, or more free castor wheels are used. Smart walkers such as *AGoRA Walker*, *UFES Smart Walker*, and the *ABSGo++* employ this drive type [15, 20, 22]. The motion possibilities in this configuration depend on the rotation speed and direction of the motorized wheels:
 - *Forward or backward motion:* The wheels rotate at the same speed, and depending on the direction, the structure moves straight forward or backward.
 - *Curved motion:* One wheel rotates faster than the other, making the structure follow a curved path. The slower wheel dictates the turning side.
 - *On-site turning motion:* The wheels rotate at the same speed in opposite directions, making the structure turn about the midpoint of the motorized wheels.
- **Tricycle:** This type of drive system has a single wheel that can be motorized, steered, or both [13]. Smart walkers such as *MARC* employ this drive type [16]. To guarantee balance and stability, two free fixed wheels are placed in the back. In this case, the motion possibilities work as follows:
 - *Forward motion:* The motorized wheel is in the middle position and driven at the desired speed.
 - *Curved motion:* The motorized wheel is positioned at a specific angle and driven at the desired speed.
 - *Circular motion:* The motorized wheel is at 90° , making the structure rotate in a circular path, whose center is the middle point between the rear wheels.
- **Ackerman steering:** This configuration describes the standard steering used in automobiles. Two linked motorized rear wheels and two linked steered front wheels characterize it. Smart walkers such as *GUIDO* and *c-Walker* use this drive type [18, 19]. There are only two motion possibilities:
 - *Forward or backward motion:* The rear wheels rotate at the same speed because they are linked, and the front wheels are not steered.

- *Curved motion*: The rear wheels rotated at the same speed and the front wheels are steered at the desired angle. The structure follows a curved motion with a minimum turning radius.

2.2.5.4 Mobile Robot Kinematics

Mobile robot kinematics describes the behavior of mobile robots considering their physical configuration in a defined workspace, the relations between their geometric parameters, and their mechanical constraints [12,23]. In this sense, this section seeks to describe such considerations when modeling robots like the smart walkers.

At this point, a key difference between mobile robots and manipulators or robotic arms arises, the position estimation. As explained in previous sections, the robot arms in social robots and the exoskeletons have one end that is treated as the fixed point or ground [12]. In this sense, the end effector's position can be instantaneously estimated by following the kinematic equations and measuring the position of each joint. However, with mobile robots the movement is not fixed, and the robot's motion must be integrated over time to estimate its position [12]. Thus, position estimation is a challenging task that cannot be achieved instantaneously.

With this in mind, understanding mobile robot kinematics addresses the problem of describing how each wheel contributes to the overall motion, as well as imposes motion constraints. Moreover, it also addresses the formulation of forward kinematic models that describe the robots' movement in terms of their geometry and wheels' behavior.

Representing Robot Position

The first step in deriving a kinematic model for a smart walker (or a mobile wheeled robot) is representing its position in a particular environment. To this end, the following assumptions are considered:

1. The smart walker is modeled as a rigid body on wheels.
2. The smart walker is only allowed to move in the horizontal plane.
3. The degrees of freedom of the wheels and internal joints of the smart walker are ignored.
4. The translational friction between the wheels and the ground at the point of contact is large enough so that the wheels do not experience translational slippage.
5. The rotational friction between the wheels and the ground at the point of contact is small so that the wheels can rotate.
6. The center of mass is located at the point of interest.

To set proper coordinate systems, two reference frames are defined, the global reference frame and the local reference frame. The global or inertial reference frame is located at the origin $O: \{X_I, Y_I\}$ of the horizontal plane. The local reference frame $\{X_R, Y_R\}$ is located at the robot chassis and defines the point of interest. According to this, only three elements are required to define the position or local frame of a smart walker, the X_R coordinate, the Y_R coordinate, and the orientation φ along with the Z_R or vertical axis. Thus, the pose (i.e., position and orientation)

and velocities of a robot at the global reference frame are defined by Eqs. 2.35 and 2.36.

$$\xi_I = [x \ y \ \varphi]^T \quad (2.35)$$

$$\dot{\xi}_I = [\dot{x} \ \dot{y} \ \dot{\varphi}]^T. \quad (2.36)$$

Note that in Eqs. 2.35 and 2.36, the robot's position is defined by x and y , which represent any position in the global frame. Similarly, the angular difference between the global and local reference frames is represented by φ . As an illustration, Fig. 2.11 shows a representation of a smart walker in a given environment and the relationships between the global and local reference frames. In this case, the point of interest (i.e., local reference frame) is located at the user's estimated position.

This representation can be generalized employing the concept of *homogeneous transformations*. A *homogeneous transformation* describes the position and orientation of a solid body with respect to the global reference frame using a 4×4 transformation matrix \mathbf{A} . This transformation matrix is described by Eq. 2.37.

$$\mathbf{A} = \begin{bmatrix} \mathbf{R} & \mathbf{p} \\ \mathbf{0} & 1 \end{bmatrix}, \quad (2.37)$$

where \mathbf{p} is the position vector and \mathbf{R} represents the rotation of the local frame with respect to the global frame. Rotations might occur along any axis. However, in mobile robots such as the smart walkers, the motion only occurs on a horizontal

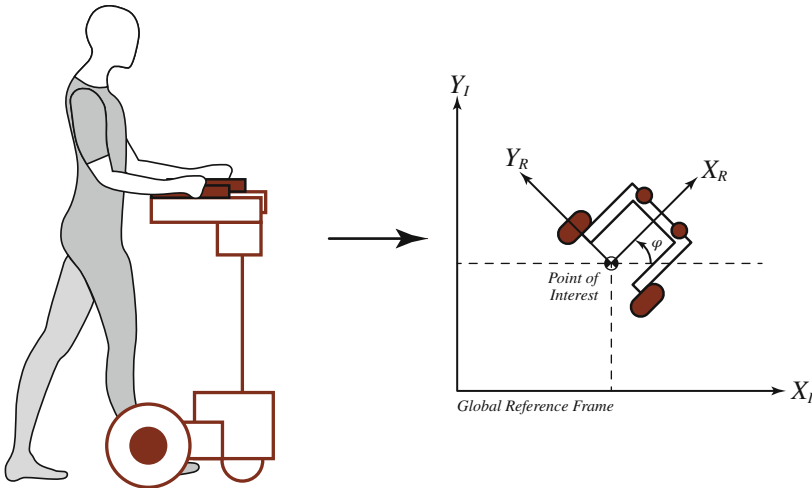


Fig. 2.11 Position representation of a smart walker in a given environment

plane. This also means that the rotations are only with respect to the vertical axis z . In this sense, the pose of a robot in the global reference frame, which is rotated about the vertical axis z , can be described as

$$\begin{aligned}\xi_I &= \mathbf{R}_z \xi_R \\ \xi_I &= \begin{bmatrix} \cos(\varphi) & -\sin(\varphi) & 0 \\ \sin(\varphi) & \cos(\varphi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_R \\ y_R \\ \varphi \end{bmatrix} \\ \xi_I &= \begin{bmatrix} x_R \cos(\varphi) - y_R \sin(\varphi) & 0 \\ x_R \sin(\varphi) + y_R \cos(\varphi) & 0 \\ 0 & 0 & \varphi \end{bmatrix}.\end{aligned}\tag{2.38}$$

The orthogonal rotation matrix \mathbf{R}_z is also useful to map motion along the axes of the global reference frame to motion along the axes of the local reference frame [12]. In particular, this mapping is a function of the pose of the robot as described in Eq. 2.39

$$\begin{aligned}\dot{\xi}_R &= \mathbf{R}_z^{-1} \dot{\xi}_I \\ \dot{\xi}_R &= \begin{bmatrix} \cos(\varphi) & \sin(\varphi) & 0 \\ -\sin(\varphi) & \cos(\varphi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\varphi} \end{bmatrix}.\end{aligned}\tag{2.39}$$

Nonholonomic Constraints

As described in [23], and without delving into the strict definition of nonholonomic constraints, mobile robots are systems that are subject to these constraints as they are under-actuated robots. When referring to *nonholomicity*, it is stated that if there is a difference between the number of degrees of freedom (n) of a robot and the independent motions (k) that the robot can produce, the *nonholomicity exists*. For instance, in smart walkers with differential drive configurations, only two wheels are actuated, that is, $k = 2$, whereas three degrees of freedom exist, that is, $n = 3$. Thus, there is one nonholonomic constraint (i.e., $n - k = 1$) [23].

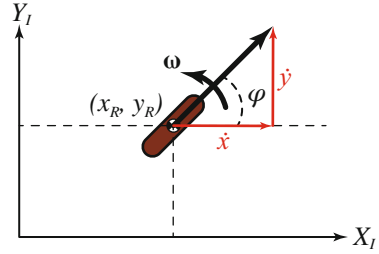
In general, this nonholonomic constraint in mobile robotics often refers to: (1) the motion constraint of a disk that rolls on a plane without slipping or (2) the speed restriction in the robot's traverse direction [23]. This also means that the velocities \dot{x} , \dot{y} , and $\dot{\varphi}$ cannot take arbitrary values [23], and thus they are constrained as shown in Eq. 2.40.

$$\dot{x} \sin(\varphi) - \dot{y} \cos(\varphi) = 0.\tag{2.40}$$

Unicycle Kinematic Model

One of the most standard and straightforward kinematic models for wheeled mobile robots refers to the unicycle model. With this model, the robots are analyzed as

Fig. 2.12 Illustration of the unicycle kinematic model



if they were a simple conventional wheel rolling on a horizontal plane [23]. This configuration is illustrated in the global reference frame in Fig. 2.12.

The pose of the unicycle configuration can be described using the representation of Eq. 2.35, which is $\xi_I = [x_R, y_R, \varphi]^T$. This representation uses the position coordinates of the point of interest located at the ground contact of the wheel, and the orientation with respect to the x -axis [23]. Figure 2.12 also shows the linear velocity of the wheel μ and the angular velocity about the vertical axis ω . Moreover, from Fig. 2.12 the motions along x -axis and y -axis can be used to describe the unicycle model as follows:

$$\dot{\xi}_I = \begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \mu \cos(\varphi) \\ \mu \sin(\varphi) \\ \omega \end{bmatrix} = \begin{bmatrix} \cos(\varphi) \\ \sin(\varphi) \\ 0 \end{bmatrix} \mu + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega. \quad (2.41)$$

Note that the nonholonomic constraint can be derived from \dot{x}_R and \dot{y}_R by eliminating μ . Moreover, considering the unicycle model presented in Eq. 2.41, and assuming that the linear and angular velocities are the joints of the system, the Jacobian matrix of the system is [23]:

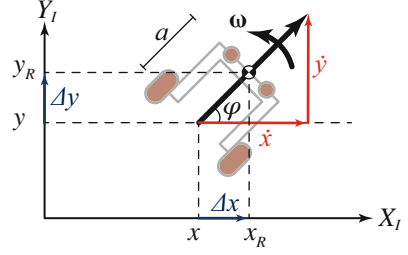
$$\mathbf{J} = \begin{bmatrix} \cos(\varphi) & 0 \\ \sin(\varphi) & 0 \\ 0 & 1 \end{bmatrix}. \quad (2.42)$$

Displaced Kinematic Model

Another interesting kinematic model arises when the point of interest is displaced to a new point in the robot's front. This model adds little complexity to the previous unicycle model, while functional, for instance, in path following tasks. Figure 2.13 illustrates the new location of the point of interest. As described in [24], although the displaced point of interest can have velocities in any direction, the robot is still considered as a nonholonomic robot.

Moreover, from Fig. 2.13 it can be observed that the x and y coordinates of the previous unicycle model are displaced by Δx and Δy , whereas the orientation of the robot is still about to vertical axis located in the middle of the wheels. In this sense, the pose of the robot is now defined by Eq. 2.43.

Fig. 2.13 Illustration of the displaced kinematic model



$$\xi_I = \begin{bmatrix} x + \Delta x \\ y + \Delta y \\ \varphi \end{bmatrix}. \quad (2.43)$$

Considering that the distance between the middle of the wheels and the new point of interest is a , the displacements Δx and Δy can be described as follows:

$$\xi_I = \begin{bmatrix} x + a \cos(\varphi) \\ y + a \sin(\varphi) \\ \varphi \end{bmatrix}. \quad (2.44)$$

To obtain the final displaced kinematic model, the motions along x_R and y_R are required. Thus, taking derivatives at both sides of Eq. 2.44, the following model is obtained:

$$\begin{aligned} \dot{\xi}_I &= \begin{bmatrix} \dot{x} - a \dot{\varphi} \sin(\varphi) \\ \dot{y} + a \dot{\varphi} \cos(\varphi) \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \mu \cos(\varphi) - a \omega \sin(\varphi) \\ \mu \sin(\varphi) + a \omega \cos(\varphi) \\ \omega \end{bmatrix} \\ \dot{\xi}_I &= \begin{bmatrix} \cos(\varphi) \\ \sin(\varphi) \\ 0 \end{bmatrix} \mu + \begin{bmatrix} -a \sin(\varphi) \\ a \cos(\varphi) \\ 1 \end{bmatrix} \omega. \end{aligned} \quad (2.45)$$

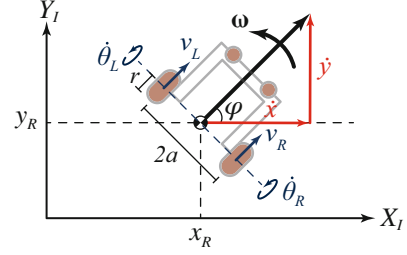
In this case, assuming the linear and angular velocities as the system's action variables, the Jacobian matrix of the system is

$$\mathbf{J} = \begin{bmatrix} \cos(\varphi) & -a \sin(\varphi) \\ \sin(\varphi) & a \cos(\varphi) \\ 0 & 1 \end{bmatrix}. \quad (2.46)$$

Differential Drive Kinematic Model

Considering that the most of smart walkers in the literature employ a differential drive locomotion, it is of great relevance to obtain a kinematic based on the independent speed of the actuated wheels. Figure 2.14 illustrates the geometry and

Fig. 2.14 Illustration of kinematic and geometrical parameters for the differential drive configuration



kinematic parameters of a differential smart walker with two actuated wheels in the rear.

As shown in Fig. 2.14, the angular speeds of the left and right wheels are $\dot{\theta}_L$ and $\dot{\theta}_R$, respectively. Similarly, the linear velocities of the left and right wheels are v_L and v_R , respectively. Moreover, the point of interest in this formulation is the same as in the unicycle, i.e., in the midpoint between the wheels, so that the linear and angular velocities of the smart walker are also μ and ω , respectively. In this sense, the velocities of each wheel can be described by Eq. 2.47 [23].

$$\begin{aligned} v_L &= \mu - a\dot{\omega}, \\ v_R &= \mu + a\dot{\omega}. \end{aligned} \quad (2.47)$$

From Eq. 2.47, v_R and v_L are added and subtracted to obtain

$$\begin{aligned} \mu &= \frac{1}{2}(v_R + v_L), \\ 2a\omega &= v_R - v_L. \end{aligned} \quad (2.48)$$

Considering the non-slippage condition previously defined, the linear velocities of the wheels can also be defined as $v_R = r\dot{\theta}_R$ and $v_L = r\dot{\theta}_L$, respectively. In this way, using the definition of \dot{x}_R and \dot{y}_R described by Eq. 2.41, the kinematic model for the differential drive configuration can be derived from Eq. 2.48 as follows:

$$\begin{aligned} \dot{\xi}_I &= \begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \frac{r}{2}(\dot{\theta}_R \cos(\varphi) + \dot{\theta}_L \cos(\varphi)) \\ \frac{r}{2}(\dot{\theta}_R \sin(\varphi) + \dot{\theta}_L \sin(\varphi)) \\ \frac{r}{2a}(\dot{\theta}_R - \dot{\theta}_L) \end{bmatrix} \\ \dot{\xi}_I &= \begin{bmatrix} \frac{r}{2} \cos(\varphi) \\ \frac{r}{2} \sin(\varphi) \\ \frac{r}{2a} \end{bmatrix} \dot{\theta}_R + \begin{bmatrix} \frac{r}{2} \cos(\varphi) \\ \frac{r}{2} \sin(\varphi) \\ \frac{-r}{2a} \end{bmatrix} \dot{\theta}_L. \end{aligned} \quad (2.49)$$

In this case, assuming the linear velocities of the wheels as the system's joints, the Jacobian matrix is

$$\mathbf{J} = \begin{bmatrix} \frac{r}{2} \cos(\varphi) & \frac{r}{2} \cos(\varphi) \\ \frac{r}{2} \sin(\varphi) & \frac{r}{2} \sin(\varphi) \\ \frac{r}{2a} & \frac{-r}{2a} \end{bmatrix}. \quad (2.50)$$

2.3 Robotic Actuation Systems

An actuation system means a system that uses a type of energy at its input. As a result, it is generated mechanical energy, and the systems that provide this function are named transducers. These actuation systems are used for various workspaces and different applications. In this particular case, the rehabilitation and assistance applied robotics mentioned in Chap. 1 contemplates using actuation systems in robots whose working environment is shared with humans. Therefore, the implementation of different actuation systems is conditioned by the collaboration level that each of these robotic devices provides to the human. For this purpose, factors such as the interaction of forces presented between the human and the robotic device, guaranteeing the user's safety during the interaction with the robotic device, protecting the robotic device during the development of various tasks, and evaluating the similarity of the robotic device's movements compared to the human movements. In this sense, the purpose of this section is to show the different actuation systems used in devices designed for clinical environments. This section will focus on this topic in the following order: (i) actuation systems in lower-limb exoskeletons; (ii) actuation systems in social robotics; and (iii) actuation systems in robotic walkers.

2.3.1 Actuation Systems in Lower-Limb Exoskeletons

An essential factor in developing lower-limb exoskeletons is the implementation of an actuation system to transmit a force to the user's joints using an energy source. Currently, several proposals can be found that fulfill this purpose. Some of these include electric actuation systems, hydraulic actuation systems, and pneumatic actuation systems. Besides, various power transmission mechanisms complement these actuation systems. The main purpose of this section is to provide an actuation system general description mentioned. Additionally, some of the transmission mechanisms are usually mentioned in the literature and applied in the design and development of lower-limb exoskeletons.

2.3.1.1 Electric Actuation System

The electric actuation system is one of the most widely used applications related to the development of lower-limb exoskeletons [25]. This system uses electrical signals as a power source. In this case, electrical signals are converted into mechanical energy using DC electric motors. This actuation system can generate the torque profiles required to generate movement in the user's joints. However, the use of

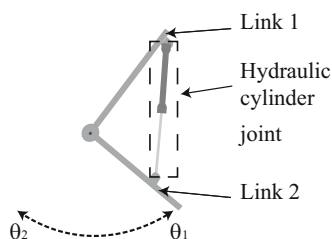
the electric actuation system has some disadvantages. For example, the actuation system's size and weight depend on the torque profiles that it can generate [26], the cost of this system is higher compared to other actuation systems (hydraulic and pneumatic) [27, 28], and it does not present a low impedance in comparison to the pneumatic actuators (back drivable device) [29, 30]. Some of these disadvantages can be compensated by using a sensory interface and various types of mechanical energy transmission. However, this increases the cost of the system. Finally, this system is used to implement position and velocity control in the user's joints [26]. Some examples of the use of this system will be shown in Sect. 2.3.1.4.

2.3.1.2 Hydraulic Actuation System

Other actuation systems have been used to develop active joints in lower-limb exoskeletons, such as hydraulic actuation systems. These systems are composed of a fluid or liquid (oil, fuel) with high system viscosity [31, 32]. The primary purpose of this system is focused on the generation of a force/torque that would generate the joint's movement [32]. One of the advantages in its implementation is that it has a high power-to-weight ratio and the ability to move objects at low speeds and operate at a constant pressure without requiring large amounts of additional energy [33]. Another property of its use is the zero impedance to the joint. This means that this actuation system will allow the movement of the joint without increasing the user's metabolic cost [34]. Finally, this system does not require that the complete structure of the actuation system be located in the joint of the device. Therefore, much of the actuation system can be located in a section where it will not suffer damage caused by the joint movement [34]. On the other hand, this actuation system is not efficient for implementing a position controller [32]. As mentioned, this system is implemented to perform force/torque based control strategies. As an example, Fig. 2.15 shows a schematic of the location of part of the hydraulic actuation system in a lower-limb exoskeleton joint.

The implementation of this actuation system in lower-limb exoskeletons has been relevant in some commercially distributed devices and others that have demonstrated their effectiveness in various competitions. Exoskeletons such as *BLEEX* (University of California Berkeley, USA) [35] apply this principle to complement hip, knee, and ankle movements in the sagittal plane, this design contributed to the *BLEEX* exoskeleton to operate at a speed of 1.3 m/s [36]. *CPWalker*, a system comprised of a body weight support (BWS), robotic walker,

Fig. 2.15 Part of the hydraulic actuation system (hydraulic cylinder) is coupled between the two links. Subsequently, a stiffness variation applied in the hydraulic cylinder could allow various movements in the user's limbs



and lower-limb exoskeleton, uses a hydraulic pump in the system used for the BWS to lift the user's body [37]. Other exoskeletons such as those mentioned in [38] apply this actuation system at the hip and knee; thus, thrust force values from 5 N to 2 KN are obtained. An *ElectroHydraulic AFO* (EHO) (Université Laval, Canada) is considered a back drivable system given its implementation of hydraulic actuation systems.

2.3.1.3 Pneumatic System Actuator

The electrical actuation system has been applied for various applications. However, the acquisition of these actuation systems requires a high cost. For this reason, other alternatives are investigated to fulfill the electrical actuation system function. In this sense, the pneumatic actuation system has been adapted to these applications related to the exoskeleton design. Its performance is based on compressed air as an energy source to generate calculated pressures in the system presented in [39]. As a result, calculated force or torque is applied to the exoskeleton's joints. This actuation system shares some characteristics of hydraulic actuation systems, such as low effectiveness in developing a velocity or position control [39, 40]. Additionally, it performs a non-linear response due to the element that it uses to generate energy. On the other hand, this system provides higher efficiency in implementing control strategies with force/torque [41]. Some of the advantages of this actuation system are the low cost, the facility for coupling in the mechanical structure, a lightweight structure [26, 42], and low difficulty for the system maintenance [42]. Some disadvantages of the system are the low effectiveness of implementing position or velocity control [42] and the actuation system not efficient in power transmission that is not used to generate fast responses [40]. Finally, the actuator behaves like a damping system that varies the stiffness of the joint and avoids abrupt movements of the system [40].

Previously, this actuation system was not considered for lower-limb exoskeleton actuation systems due to limited precision and accuracy in some cases. Therefore, they were implemented on stationary platforms. On the other hand, the performance of these systems could depend on the type of control applied on the platforms. In this sense, several lower-limb exoskeletons can be found that use this type of actuation system. For example, *ALEX* (University of Delaware, USA) uses a Pneumatic Muscle Actuator (PMA) to actuate the sagittal plane's hip and knee joints. As a result, an assist-as-needed (AAN) control system was developed, which adjusts the assist level of the device by evaluating the angular position of the joint [43]. Lower-limb exoskeletons for activities of daily living assistance such as *WalkOn* use pneumatic cylinders for knee joint actuation in the sagittal plane, obtaining 18 m/min of gait speed [44]. Likewise, these actuators are applied in the development of soft robots, where pneumatic muscles are placed to complement the muscle contraction for the generation of dorsi-plantarflexion and inversion–eversion movements in [45]. On the other hand, using these devices can also reduce the metabolic cost of a healthy user using an ankle exoskeleton during gait with a pneumatic actuation system [46].

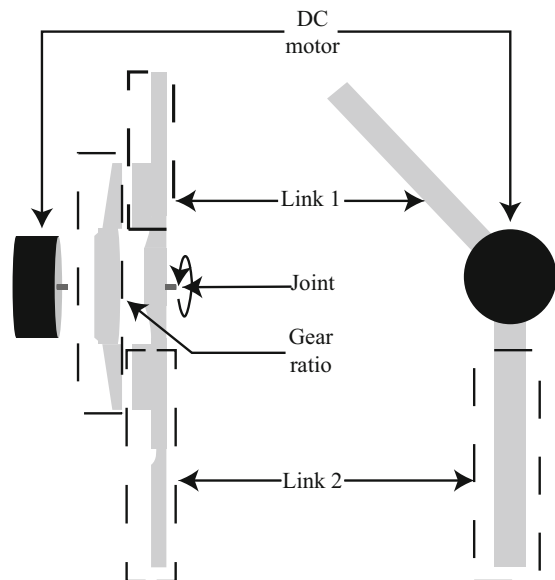
2.3.1.4 Transmission Mechanisms

Lower-limb exoskeleton joints have been actuated by various transmission mechanisms; these mechanisms fulfill the objective of applying the assistive torques for the hip, knee, and ankle joints of the lower limb. Several actuation systems have been developed to support the user's movements and provide assistance in the primary movements of the lower limbs of human body. The following are the main categories:

- **Stiff Actuators:**

The implementation of electric motors in the center of rotation (CR) of the joints of a lower-limb exoskeleton has been applied in the development of exoskeletons focused on activity of daily living assistance. This type of system has been used to execute predefined trajectories [47] applying a position controller. For instance, *ReWalk* (ReWalk Robotics, United Kingdom) [48], *Ekso* (University of California Berkeley, USA) [49], and *Indego* (Vanderbilt University, USA) [50] use this type of methodology for the execution of various activities of daily living with predefined trajectories. Usually, this actuation system focuses on the assistance of the hip, knee, and ankle joints in the sagittal plane, to provide primary movements of the human body in activities of daily living and walking activity. Figure 2.16 shows an example of a stiff actuator implemented in the lower-limb exoskeleton joint.

Fig. 2.16 Stiff actuator system schematic, the DC motor is coupled to the gearbox ratio. Subsequently, this actuation system is located directly in the exoskeleton's joint



This transmission mechanism uses a DC electric motor, and the gearbox magnifies the torque generated by the motor as follows:

$$\tau_j = \tau_n \times n_{gr}, \quad (2.51)$$

where τ_j is the joint torque, τ_n is equal to the motor nominal torque, and N_{gr} is the gearbox ratio. In general, exoskeletons with stiff actuators are developed for spinal cord injury users. Therefore, these are designed as portable and wearable devices, for prolonged use. These devices are complemented with a Graphical User Interface that provides the user with the capability to control the activities developed using the lower-limb exoskeleton [44, 48, 51]. On the other hand, this stiff actuator can be used in control strategies aimed in the physical Human–Robot Interaction (pHRI) using sensors that estimate the user’s kinetic parameters and simulate a back drivable device. This actuation system will allow the generation of required torques to the user’s lower-limb movements [4, 52, 53]. The disadvantage of using gearboxes is based on the reduction of the angular velocity of the actuation system by dividing the value of the motor nominal speed by the gearbox ratio. The implementation of DC motors with gearboxes is adapted to the joints to generate the desired angular positions in the walking activity, assisting the lower extremities [44, 54, 55]. On the other hand, the reduction ratio of the gearboxes limits the movements of the joint performed by the user, so this actuation system is used in lower-limb exoskeletons for people with spinal cord injury.

- **Compliant Actuators**

The series elastic actuators (SEAs), for example, are presented to a shock tolerance actuation system [56]; this main characteristic focuses on uncoupling the joint to the DC motor and the gearbox ratio using an elastic element [57]. As a consequence, the SEA properties include a low impedance and low friction. Generally, it is implemented in the development of force controller; impedance–admittance controller is applied in devices to take advantage of the physical Human–Robot Interaction (pHRI) [41]. The SEAs respond to the replacement of high precision position controllers in manipulator robots that interact with users according to the workspace [58]. In the last years, various compliant actuators have been presented in the literature. This section is defined as a mechanically adjustable compliance and controllable equilibrium position actuator (MACCEPA) [59]. This actuator was used to develop some lower-limb exoskeletons [51, 60, 61] aimed at rehabilitating activities of daily living applying the pHRI definition.

The MACCEPA is an actuation system focused on including the user’s motion intention in the control strategies. Its development is based on a system mass–spring system, which exercises a pretension on the spring varying the stiffness of the joint for the development of various activities. Figure 2.17 shows the concept of the MACCEPA system presented in [59]. With radius (r), the disc rotates to

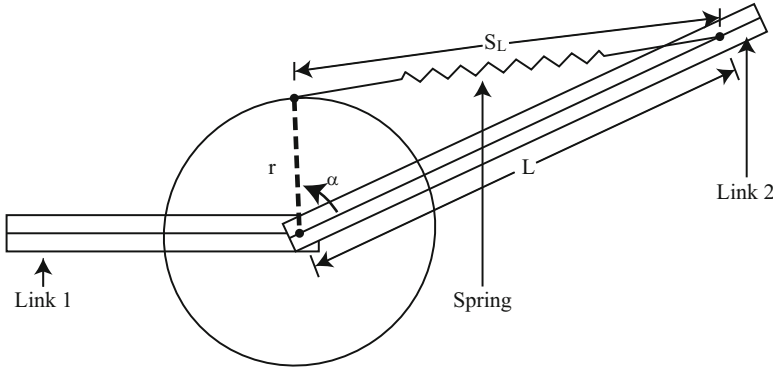


Fig. 2.17 MACCEPA definition, where L is the length of the starting point of the joint to the point where the spring is attached; S_L is equal to the length of the pretensioned spring; r is equal to the disc radius that provides the pretension to the system

vary the spring pretension, which provides stiffness to the system according to the activity to be performed.

The torque generated by this system is expressed by the equation shown below:

$$\tau = krL \times \sin \alpha \times \left(1 + \frac{P - |L - r|}{\sqrt{r^2 + L^2 - 2rL \times \cos \alpha}} \right), \quad (2.52)$$

where k equals to the system's elasticity constant, r is the disc radius that changes the pretension of the spring, L is the length from the disc location to the point where the end of the spring is attached, and P corresponds to the extension of the spring caused by the pretension [59]. So the MACCEPA actuator adjusts the rotation of the disc to generate pretension in the joint; this allows generating different torque profiles depending on the value of α and the elasticity constant of the spring. The actuation system has been characterized as a back drivable device. Likewise, the variation of the spring pretension can turn it into a non-back drivable system, capable of generating 70 Nm for the hip and knee joints [60].

Exoskeletons such as *ALTACRO* (Vrije University, Belgium) [60] implement the MACCEPA for gait rehabilitation by testing different levels of joint stiffness and the effects that this configuration has on gait [59]. Wearable lower-limb exoskeletons such as *Biomot* (Future and Emerging Technologies (FET), Spain) [61] apply this actuation system to the hip and knee joints. Proposing an exoskeleton focused on the assistance of these joints in the sagittal plane.

- **Cable-Driven Actuators:**

The actuation systems presented in the literature for lower-limb exoskeletons were focused on the generation of the necessary torques for hip, knee, and in some cases ankle joints. On the other hand, some factors such as the actuation

system location, dependence on a rigid structure, weight, among others generate limitations for the movement of these joints. For this reason, some systems were developed, such as the driven cables, which present an actuation mechanism focused on the implementation of a mass–spring–damper system.

These actuation systems offer the capability to generate movements in the hip, knee, and ankle joints, without presenting constraints associated with the exoskeleton mechanical structure [62]. The implementation of this actuation system contributes to the pHRI, where the main objective is the user's participation in the development of activities of daily living. Their location is distributed in a way that they are not located directly in the lower-limb joints. This characteristic decreases the weight perceived by the user's joints [26, 63]. Additionally, the weight of these joints is considered lower than the implementation of electric motors adapted with gearboxes. Finally, the implementation of these systems does not require a rigid structure [64].

As an example, L^Ower extremity P^Owered ExoSkeleton (*LOPES*) (University of Twente, Netherlands) [62] makes use of this system, proposing a series elastic actuator system using Bowden cables. The *LOPES* actuation system includes the use of Bowden cables, complemented with the implementation of springs for the lower-limb joints. This system was focused on the variation of the pretension of the springs, for the generation of assistive torques for the activity of the march. Factors such as wear and friction of the cables with the system structure are relevant for developing a closed-loop control [62]. As an example, Fig. 2.18a shows a schematic of this system. An Exosuit *Myosuit* (ETH Zurich, Switzerland) [65] is a wearable device that does not have a rigid structure, and implements driven cable actuators to assist the sitting-to-standing activity and walking activity. The *Myosuit* actuation system allows the user to generate voluntary movements monitored through loads cells and IMU sensors to support the motion intention generated by the user who suffers muscle weakness. Figure 2.18b shows a schematic of this exosuit in the user's lower limbs.

In conclusion, the actuation systems shown in this section are some of the relevant actuation mechanisms developed for the lower-limb exoskeletons most mentioned in the literature. Each of these actuation systems was aimed to use the pHRI in the control strategies developed according to the robot workspaces. In this case, the lower-limb exoskeletons mentioned are designed for the rehabilitation and assistance of the relevant activities of daily living performed by a user who suffers from a neurological disease. Each of the mentioned actuation systems presents some advantages and disadvantages depending on the requirements in maximum torque, maximum angular velocity, actuation system's weight, usability, and the required control strategies in developing various activities. Finally, these factors will affect the support and assist the hip, the knee, and the ankle joint in various primary movements of the human body.

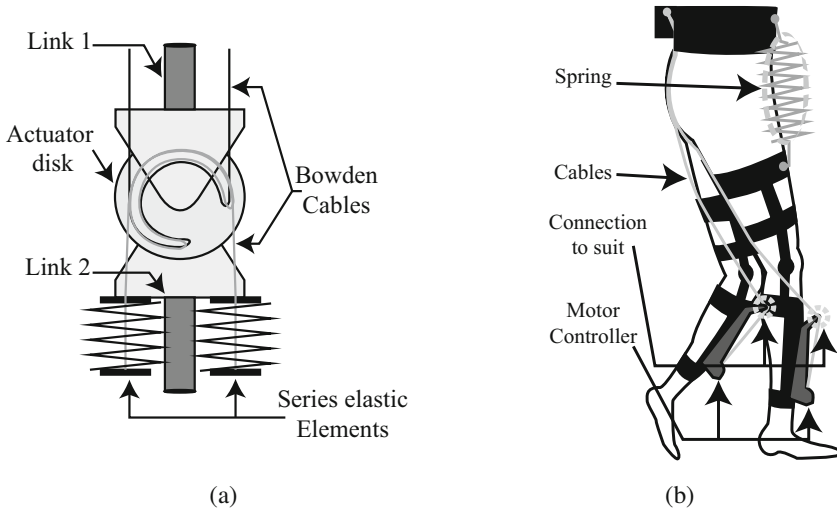


Fig. 2.18 Cable-driven actuators schematic of lower-limb exoskeletons developing activities of daily living: (a) actuation system schematic of the LOPES exoskeleton developed by Bowden cables located in lower-limb joints; (b) exosuit named *Myosuit* is considered a wearable exoskeleton that applies soft robotics to support various activities of daily living using driven cables actuation system in the lower-limb joints

2.3.2 Actuation Systems in Social Robots

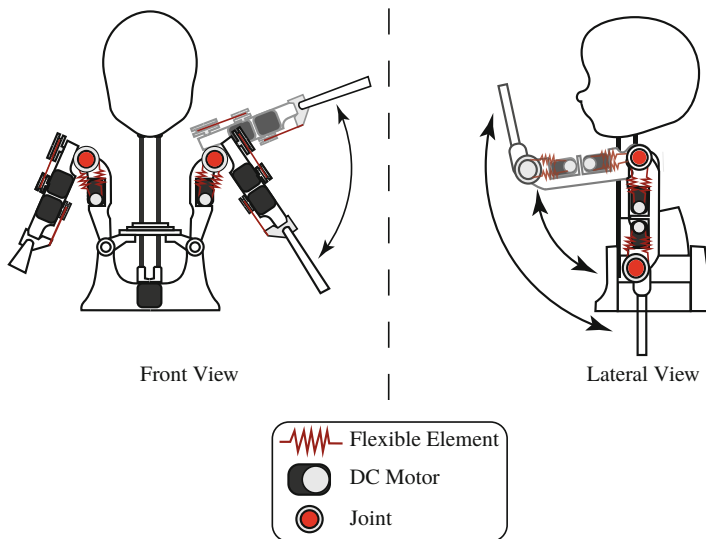
As mentioned in the previous section (Sect. 2.3.1) there are different types of actuation for lower-limb exoskeletons, within which are pneumatic, hydraulic, or electric actuation systems. In Social Assistive Robots (SAR), different actuation systems based on electric actuation systems have been implemented, as shown in Table 2.5.

The table shows the most common types of actuation systems on social robots. In this case, most of the robots are actuated by stiff actuation systems and a few by an actuation system based on series elastic actuators (SEAs). The stiff actuation systems are composed of DC motors, and a gearbox that increases the torque generated by the motor. On the other hand, actuation systems based on SEA consist of an elastic element between the DC motor and the gearbox. This elastic element provides benefits such as shock tolerance, lower reflected inertia, and minor damage to the environment [82].

Most of the SAR contain many degrees of freedom because they attempt to generate realistic biological movements. The *NAO* robot is one of the robots with the most degrees of freedom allowing it to generate movements such as walking and dancing. Other robots, such as *CASTOR*, *KASPAR*, *Huggable Bear*, and *Paro*, have less complex actuation systems but perform limb movements or generate facial expressions.

Table 2.5 Actuation systems in Social Assistive Robots

Robot	Purpose	Actuation system	Degrees of freedom
<i>CASTOR</i> [7]	Autism spectrum disorder	SEA	14
<i>Keepon</i> [66]	Autism spectrum disorder	Stiff	4
<i>Probo</i> [67, 68]	Autism spectrum disorder	SEA	20
<i>KASPAR</i> [69, 70]	Autism spectrum disorder	Stiff	17
<i>Pleo</i> [71, 72]	Support for well-being	Stiff	14
<i>TIAGo</i> [73]	Elderly care	Stiff	12
<i>Pepper</i> [74]	Elderly care	Stiff	19
<i>HOBBIT</i> [75]	Elderly care	Stiff	5
<i>Huggable Bear</i> [76]	Autism spectrum disorder	Stiff	8
<i>NAO</i> [77–79]	Autism spectrum disorder	Stiff	25
<i>Paro</i> [80, 81]	Autism spectrum disorder	Stiff	7

**Fig. 2.19** Series elastic actuators (SEAs) in *CASTOR* robot improve the interaction and safeguard the structure to provide the ability to interact physically and socially with CwASD during therapy

The robots focused on therapies with children with autism spectrum disorder (CwASD) such as *CASTOR* and *Probo* are the only ones based on a SEA actuation system. This is for the type of actuation, and SEA allows the execution of therapies focused on physical interaction, necessary for the development of CwASD, without representing a risk for both the child and the robot [83]. Figure 2.19 shows the actuation system in *CASTOR* Robot, an example of SEA in SAR.

2.3.3 Actuation Systems in Smart Walkers

In case of mobile wheeled robots such as the smart walkers, the actuation systems are commonly more straightforward than the actuators employed by social robots and exoskeletons. In general, the actuation systems in smart walkers can be categorized according to their purpose: (1) actuation systems for motion and (2) actuation systems to interact with the user. As shown in Fig. 2.20, there are several actuation interfaces that can be equipped in smart walkers. Depending on the design requirements or rehabilitation purpose, these, more than one of these, actuation systems can be used.

2.3.3.1 Actuation Systems for Motion

Most of the robotic walkers reported in the literature implement differential drive systems, in which only two motorized wheels are used [15, 20, 22, 84]. Usually, each of the actuated wheels is equipped with a brushed DC electric motor coupled to a gearbox to increase the delivered torque by the motor. However, the most current developments in robotic walkers migrate to brushless DC motors that are directly integrated into the wheels (i.e., BLDC hub), such as the wheel motors equipped in electric scooters.

Although the literature lacks in reporting the torque characteristics required for a walker-assisted gait application, the generated torque should be sufficient to move the robot chassis, while a person partially supports its weight on it. In this way, when selecting or designing these types of actuation system, it is helpful to consider that:

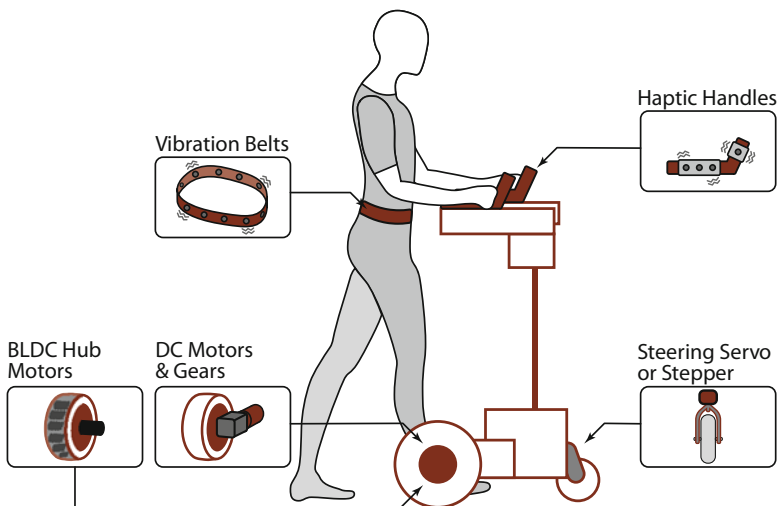


Fig. 2.20 Illustration of common actuation interfaces in smart walkers

1. The smart walker should support an average payload from 60 kg to 90 kg, based on the average weight of an adult male.¹
2. The smart walker should be able to provide an average linear speed of 1.5 m/s, based on the average walking speed of a healthy adult.²

In the tricycle drive and Ackerman steering configurations, the actuation systems are frequently located in front to provide the steering of the device [16, 18, 19]. In such configurations, the actuation is often accomplished by electrical stepper motors or servomotors. Once again, even though the electrical characteristics have not been appropriately benchmarked in the literature, the actuation system should be able to steer the device considering the friction of the wheel and the weight of the smart walker while supporting a user.

Finally, electromechanical brakes or braking systems are often implemented in smart walkers. Although these devices do not provide motion during gait assistance, they are generally used to limit movement in hazardous situations or guidance and cognitive support tasks.

2.3.3.2 Actuation Systems to Interact with the User

There are several ways to communicate information with the user in rehabilitation and assistance tasks in smart walkers. One of these communication channels is achieved using actuators, whether worn by the user or mounted on the device, to provide haptic feedback to the user [21]. In these scenarios, small DC motors or vibration motors are often equipped, through which information such as the presence of obstacles, a dangerous situation, or the path to be followed can be indicated to the user. For instance, vibration belts and haptic handlebars have been reported in the literature [21, 85].

2.4 Robotic Sensory Architectures

One of the most important tasks in assistive and rehabilitation robots is the ability to acquire information about the users and the surrounding environment. This is commonly accomplished by sensing devices that are capable of measuring physical magnitudes and converting them into electrical signals. In this section, the most common sensors used in lower-limb exoskeletons, social robots, and smart walkers are presented. The particular selection of sensors and input devices in these devices is referred to as sensory architectures.

¹<https://www.cdc.gov/nchs/fastats/body-measurements.htm>.

²<https://www.sciencedirect.com/topics/medicine-and-dentistry/walking-speed>.

2.4.1 Sensory Architecture of Lower-Limb Exoskeletons

The implementation of various sensory interfaces is developed with many objectives. First, the assessment of kinematic parameters for the gait pattern provides information to evaluate the development of this task by the health-care professional. On the other hand, the acquisition of this data is also processed to apply different control strategies implemented for the rehabilitation or assistance methods for the walking activity. These methods contribute to the therapy sessions an increase in patient compliance in developing therapy sessions promoting neural plasticity. This section mentions various sensors aimed at the acquisition of kinematic and kinetic parameters; these are relevant for the control strategies implemented in the lower-limb exoskeletons.

2.4.1.1 Kinematic Parameters

Different parameters are required in developing activities of daily living. As an example, the displacement, velocities, and acceleration of the exoskeleton joint are estimated. In this sense, this section shows some sensors that are used in the estimation of these parameters:

- **Magnetic/incremental encoders:**

This sensor is used to estimate the number of motor shaft turns. In this case, the encoders are coupled in the center of rotation of each joint of the exoskeleton. As a result, the angular position of the joints in various planes of the human body is obtained. Therefore, this sensor is implemented in rehabilitation and assistive exoskeletons.

Some examples of their application are presented in lower-limb rehabilitation exoskeletons such as *BioMot* (Future and Emerging Technologies (FET), Spain) [61], *ALEX* (University of Delaware, USA) [86], and *E_ROWA* [87] that monitor the joint angular position in the sagittal plane using it as an input to control strategies based on activities of daily living rehabilitation. On the other hand, robotic orthoses are instrumented to observe the behavior of a specific joint in various planes of the human body. Such is the case of *Ankleboot* (Massachusetts Institute of Technology, USA), which monitors ankle plantarflexion–dorsiflexion and inversion–eversion movements [88]. Finally, lower-limb exoskeletons for activities of daily living assistance such as *ReWalk* (ReWalk Robotics, United Kingdom) [48], *REX* (REX Bionics, New Zealand) [89], *VariLeg* (ETH Zurich, Switzerland) [51], *Indego* (Vanderbilt University, USA) [50], *HAL* (Tsukuba University/Cyberdyne, Japan) exoskeleton [90] use these sensors to measure the angular position of the hip, knee, and ankle joints in the sagittal plane in generating control strategies that provide 100% assistance to the user’s joints.

- **Inertial Measurement Unit (IMU):**

The acquisition of different human body parameters requires the user’s instrumentation with non-invasive sensors, where the sensor placement time is a relevant factor to use with a lower-limb exoskeleton. For these requirements,

MEMS sensors are used. For instance, the Inertial Measurement Unit (IMU) sensors are commonly mainly implemented for these applications because of the size, the patient instrumentation time, and the parameters provided by the IMU sensor [91]. Generally, the IMU sensor is composed of a gyroscope, an accelerometer, and a magnetometer that provide angular velocity, linear acceleration, and magnetic field strength in different axes [92, 93]. The IMU sensor is located in various parts of the user's human body for the activity of daily living recognition using machine learning methods to process the IMU data [92, 94–96]. Into the IMU implementation advantages is taken into account the sensor placement time, the sensor size, and the parameters that can be estimated.

2.4.1.2 Kinetic Parameters

The interaction forces between the user and the lower-limb exoskeleton are estimated for the development control strategies focused on the pHRI. On the other hand, some methodologies to use the human body electrical signal are implemented to control the voluntary movements of the user's lower limb using a lower-limb exoskeleton. The main goal of the acquisition and processing of these parameters is to increase the patient's participation in the therapy sessions, where each signal is rendered in terms of position, velocity, or acceleration values. Subsequently, the lower-limb actuation system is commanded through these values.

- **Force sensor:** Currently, several sensors are implemented. Strain gauge sensors are located in the lower-limb exoskeleton mechanical structure [52, 97]; this allows to estimate if the interaction is in terms of force/torque. As an example, *AGoRA* exoskeleton (Colombian School of Engineering Julio Garavito, Colombia) [4, 53], *ALLOR* (Federal University of Espirito Santo, Brazil) [97], and *CPWalker* [98] apply this method in the estimation of the exerted forces and the development of different control strategies using this parameter as an input system.

The acquisition of kinetic and kinematic parameters was aimed to monitor and use an input signal in various control strategies. Generally, the sensor modules showed in this section are integrated into lower-limb exoskeletons for rehabilitation and assistance workspaces. Remarkably, the mentioned sensors are non-invasive sensors and the acquired outcomes provide various parameters that allow the healthcare professional to assess the development of activities of daily living.

2.4.2 Sensory Architecture of Social Robots

Sensory systems provide robots with the ability to sense and receive information from the surrounding environment. Navigation sensors such as lasers or ultrasonic sensors give the possibility to plan a trajectory or prevent the robot from colliding. Other robots include pressure sensors, tactile sensors, and microphones that allow better interaction with the patient/user. Also, robots include cameras that serve as

Table 2.6 Sensory systems in Social Assistive Robots

Robot	Sensors
<i>CASTOR</i> [7]	Tactile sensors
<i>Keepon</i> [66]	Tactile sensors, cameras, and microphone
<i>Probo</i> [67,68]	Tactile sensors, camera, and microphone
<i>KASPAR</i> [69,70]	Cameras and force sensors
<i>Pleo</i> [71,72]	Tactile sensors, camera, microphones, infrared sensors, temperature sensor, ground foot sensors, and orientation and motion sensors
<i>TIAGo</i> [73]	Microphones, force/torque sensor, laser, RGB-D camera, sonar sensor, and Inertial Measurement Unit (IMU)
<i>Pepper</i> [74]	Tactile sensors, cameras, microphones, 3D sensor, gyroscope, sonar sensors, lasers, and Bumper sensors
<i>HOBBIT</i> [75]	Lasers, depth camera, and RGB-D camera
<i>Huggable Bear</i> [76]	Cameras, force sensors, temperature sensors, electric field sensors, and temperature sensors
<i>NAO</i> [77–79]	Tactile sensors, microphones, OmniVision cameras, pressure sensors, inertial sensor, sonar rangefinder, and infrared sensors
<i>Paro</i> [80,81]	Tactile sensors, microphones, temperature sensors, and light sensors

the robot’s eyes. This provides the ability to detect facial expressions, gestures, and distances. Table 2.6 shows different sensory systems implemented in social robots.

Most robots feature touch sensors, microphones, and cameras. On the other hand, more complex robots such as *Pleo*, *TIAGo*, *Pepper*, and *NAO* implement a variety of sensors that, as mentioned above, give the robot a better perception of the environment.

2.4.3 Sensory Architectures of Smart Walkers

As in mobile robots, the sensing devices equipped in smart walkers cover a wide range of sensors, ranging from devices to acquire information from the environment to devices to obtain information from the user. In this sense, this section briefly describes the most common sensors employed to extract information about the user’s state and intentions. Moreover, considering that the smart walkers move around indoor environments, they require sensing devices to estimate their motion and global position, as well as sensors to overcome unforeseen environmental characteristics [5,22].

2.4.3.1 Sensing Loops in Smart Walkers

The sensors mounted in the smart walkers and the sensors worn by the users can be classified in terms of the source of information delivered by the sensing devices.

- **Proprioceptive Sensors:** These devices include those sensors that estimate information related to the internal systems of the smart walker. Commonly,

proprioceptive sensors in smart walkers are used to estimate the velocity, position, and orientation in a given environment, as well as, to estimate the battery level, internal temperature, brake's state, among others.

- **Exteroceptive Sensors:** These sensors are used to obtain information related to the environment and the user, and thus they are further classified into two categories or sensing interfaces:
 1. *Human–Robot Interface (HRi):* This interface refers to the communication channel between the user and the robot, and therefore to the sensors that acquire information from the user. With the proper implementation of this interface, the smart walker is capable of: (1) the recognition of the interaction forces between the user and the platform, (2) the estimation of the user's navigation commands or intentions of movement, (3) the detection of the user's presence and proper support on the walker, (4) the estimation of the user's gait parameters and biomechanical indicators, and (5) the monitoring of the user's overall health state [22].
 2. *Robot–Environment Interface (REi):* This interface refers to the sensing devices that provide information from the environment and surrounding objects. The sensors for environment sensing allow: (1) the building and autonomous update of the environment map, (2) the autonomous localization of the device in the environment, (3) the detection of surrounding obstacles and people, and (4) the detection of hazardous situations.

2.4.3.2 Common Sensors in Smart Walkers

The smart walkers can be equipped with a wide range of sensors for multiple purposes. For instance, Table 2.7 describes some of the sensors found in smart walkers, classifying them by typical use and classification. Regarding the classification of these sensors, they can be whether proprioceptive or exteroceptive. Those sensors classified only as exteroceptive are devices that can be used in both HRi or REi. An extensive description of some of the sensors presented in Table 2.7 can be found in [12].

In addition to the above, to analyze the sensory interfaces of smart walkers reported in the literature, Table 2.8 summarizes some of the most notable smart walkers, describing their main functionalities and sensory interfaces. For instance, the CO-Operative Locomotion Aide (COOL Aide) is a three-wheeled passive SW intended to assist the elderly with routine walking tasks [103]. It includes mapping and obstacle detection systems, as well as navigation and guidance algorithms. Additionally, it is equipped with force sensors on its handlebars and a laser rangefinder (LRF) to estimate the user's desired direction to turn [5]. Other passive walkers, such as those presented in [16,21], include navigation and guidance algorithms based on haptic feedback systems and laser-based mapping.

Regarding active smart walkers, multiple sensory interfaces have been implemented [15,17,18,20,99–101,104]. These interfaces are equipped with sensors such

Table 2.7 Classification of common sensors used in mobile robotics applications, focusing on walker-assisted gait

Typical use	Sensor	Classification
Tactile or physical interaction sensors	Bumpers	Exteroceptive—REi
	Contact switches	Exteroceptive—REi
User estimation	Joysticks	Exteroceptive—HRi
	Potentiometers	Exteroceptive—HRi
	Microphones	Exteroceptive—HRi
	Temperature sensors	Exteroceptive—HRi
	Hearth rate sensors	Exteroceptive—HRi
	Laser rangefinders	Exteroceptive—HRi
	Ultrasonic sensors	Exteroceptive—HRi
	Force sensors or load cells	Exteroceptive—HRi
	Plastic optical fiber (POF) sensors	Exteroceptive—HRi
	Inertial Measurement Units (IMUs)	Exteroceptive—HRi
	Foot and hand pressure sensors	Exteroceptive—HRi
Wheels and motors sensors	Encoders (all classes)	Proprioceptive
	Potentiometers	Proprioceptive
Orientation or localization sensors	IMUs	Proprioceptive
	Gyroscopes	Proprioceptive
	RFID readers	Exteroceptive—REi
Motion sensors	Accelerometers	Proprioceptive
	Encoders (all classes)	Proprioceptive
Ranging and obstacle sensing	Laser rangefinders	Exteroceptive—REi
	2D Light Detection and Ranging (LiDAR)	Exteroceptive—REi
	3D LiDAR	Exteroceptive—REi
	Ultrasonic sensors	Exteroceptive—REi
	Reflective sensors	Exteroceptive—REi
Vision sensors	CCD/CMOS cameras	Exteroceptive
	Depth cameras	Exteroceptive

Table 2.8 Related works involving smart walkers that integrate interfaces for Human–Robot–Environment Interaction

Walker	Sensory interface	Main functionalities
GUIDO [18]	<ul style="list-style-type: none"> – Force sensors – LRF – Sonars – Encoders 	<ul style="list-style-type: none"> – Autonomous navigation – Detection of user’s intentions – Sound feedback
XR4000 [99]	<ul style="list-style-type: none"> – Force sensors – LRF – Sonars – Infrared sensors – Encoders 	<ul style="list-style-type: none"> – Autonomous navigation – Detection of user’s intentions
ASBGo++ [20]	<ul style="list-style-type: none"> – Force sensors – LRF – Sonar – Infrared sensors – Camera – Encoders 	<ul style="list-style-type: none"> – Autonomous navigation – Detection of user’s intentions – Gait monitoring – User position feedback
JARoW [17]	<ul style="list-style-type: none"> – Infrared sensors – Encoders – LRFs 	<ul style="list-style-type: none"> – User’s position estimation and prediction – Obstacle avoidance
UFES [15]	<ul style="list-style-type: none"> – Force sensors – LRF – IMUs – Encoders 	<ul style="list-style-type: none"> – Path following – Obstacle avoidance – Detection of user’s intentions – Gait monitoring
PAMM [100]	<ul style="list-style-type: none"> – Force sensors – Sonars – Camera – Encoders 	<ul style="list-style-type: none"> – Autonomous navigation – Health monitoring
MOBOT	<ul style="list-style-type: none"> – Force sensors – LRFs – Cameras – Kinect sensors – Microphones 	<ul style="list-style-type: none"> – Autonomous navigation – Detection of user’s intentions – Speech and gesture recognition – Body pose estimation – Gait Analyzer
CAIROW [101]	<ul style="list-style-type: none"> – Force sensors – LRFs 	<ul style="list-style-type: none"> – Environment analyzer – Force analyzer – Gait analyzer
ISR-AIWALKER [102]	<ul style="list-style-type: none"> – Force sensors – Kinect sensor – Encoders – RGB-D Camera 	<ul style="list-style-type: none"> – Detection of user’s intention – Gripping recognition – Gait analyzer – Autonomous navigation

(continued)

Table 2.8 (continued)

Walker	Sensory interface	Main functionalities
COOL Aide [103]	<ul style="list-style-type: none"> – Force sensors – LRF – Encoders 	<ul style="list-style-type: none"> – Autonomous navigation – Detection of user’s intentions
Wachaja et al. [21]	<ul style="list-style-type: none"> – LRF – Tilting LRF 	<ul style="list-style-type: none"> – 3D Mapping and localization – Obstacle avoidance – Vibrotactile feedback
MARC [16]	<ul style="list-style-type: none"> – Sonars – Infrared sensors – Encoders 	<ul style="list-style-type: none"> – Path following – Obstacle avoidance
c-Walker [19]	<ul style="list-style-type: none"> – Kinect like sensor – RFID reader – IMU – Camera – Encoders 	<ul style="list-style-type: none"> – Autonomous navigation – People detection and tracking – Guidance

as encoders, IMUs, and LRFs to provide navigation, guidance, and user interaction. Moreover, these smart walkers are also equipped with sensing technologies to estimate the user’s intentions to move, based on gait analysis systems and rule-based algorithms.

2.5 Conclusions

Several aspects define a gait rehabilitation and assistance device. In this sense, this chapter introduced concepts about the kinematic modeling, actuation systems, and sensing architectures of lower-limb exoskeletons, social robots, and smart walkers. Different methodologies, such as forward and inverse kinematics, DH convention, and homogeneous transformations were explained, regarding the kinematic modeling.

In terms of the actuation mechanisms of these rehabilitation robots, a concise description of the actuators employed was presented. In particular, several insights were given to make an optimal selection of an appropriate actuation system depending on the task to be performed with the devices. Additionally, factors such as the actuation system cost, sensory interface for the implementation of control strategies, weight, and mechatronic integration must be considered.

Finally, the third aspect corresponds to the sensory interface that estimates the kinematic and kinetic parameters of the devices, and allows them to acquire internal and external information. Those presented in this chapter provide the input parameters for various control strategies and the required information to guarantee a safe and natural interaction in clinical and everyday scenarios.

References

1. M.W. Spong, S. Hutchinson, M. Vidyasagar, Forward kinematics: the Denavit-Hartenberg convention. *Robot Dynamics and Control*, 2nd edn. (Wiley, New York, NY, USA, 2004), pp. 57–82
2. A.A. Hayat, R.G. Chittawadigi, A.D. Udai, S.K. Saha, Identification of Denavit-Hartenberg parameters of an industrial robot, in *Proceedings of Conference on Advances in Robotics*, pp. 1–6 (2013)
3. Q. Chen, S. Zhu, X. Zhang, Improved inverse kinematics algorithm using screw theory for a six-DOF robot manipulator. *Int. J. Adv. Robot. Syst.* **12**(10), 140 (2015)
4. M. Sánchez-Manchola, D. Gomez-Vargas, D. Casas-Bocanegra, M. Munera, C.A. Cifuentes, Development of a robotic lower-limb exoskeleton for gait rehabilitation: AGoRA exoskeleton, in *2018 IEEE ANDESCON, ANDESCON 2018 - Conference Proceedings*, pp. 1–6 (2018)
5. S.D. Sierra M., L. Arciniegas, F. Ballen-Moreno, D. Gomez-Vargas, M. Munera, C.A. Cifuentes, Adaptable robotic platform for gait rehabilitation and assistance: design concepts and applications, in *Exoskeleton Robots for Rehabilitation and Healthcare Devices* (Springer, 2020), pp. 67–93
6. A.A. Ramírez-Duque, L.F. Aycardi, A. Villa, M. Munera, T. Bastos, T. Belpaeme, A. Frizera-Neto, C.A. Cifuentes, Collaborative and inclusive process with the autism community: a case study in Colombia about social robot design. *Int. J. Soc. Robot.*, 1–15 (2020)
7. D. Casas-Bocanegra, D. Gomez-Vargas, M.J. Pinto-Bernal, J. Maldonado, M. Munera, A. Villa-Moreno, M.F. Stoelen, T. Belpaeme, C.A. Cifuentes, An open-source social robot based on compliant soft robotics for therapy with children with ASD. *Actuators* **9**(3), 1–22 (2020)
8. NAO Technical overview—NAO Software 1.14.5 documentation (2020)
9. N. Kofinas, E. Orfanoudakis, M.G. Lagoudakis, Complete analytical forward and inverse kinematics for the NAO humanoid robot. *J. Intell. Robot. Syst.* **77**(2), 251–264 (2015)
10. D. Gouaillier, V. Hugel, P. Blazevic, C. Kilner, J. Monceaux, P. Lafourcade, B. Marnier, J. Serre, B. Maisonnier, Mechatronic design of NAO humanoid, in *2009 IEEE International Conference on Robotics and Automation* (IEEE, 2009), pp. 769–774
11. N. Kofinas, E. Orfanoudakis, M.G. Lagoudakis, Complete analytical inverse kinematics for NAO, in *2013 13th International Conference on Autonomous Robot Systems* (IEEE, 2013), pp. 1–6
12. R. Siegwart, I.R. Nourbakhsh, D. Scaramuzza, *Introduction to Autonomous Mobile Robots*, 2nd edn. (MIT Press, 2011)
13. S.G. Tzafestas, Mobile robots, in *Introduction to Mobile Robot Control* (Elsevier, 2014), pp. 1–29
14. S. Page, L. Saint-Bauzel, P. Rumeau, V. Pasqui, Smart walkers: an application-oriented review. *Robotica* **35**, 1243–1262 (2017)
15. M.F. Jiménez, M. Monllor, A. Frizera, T. Bastos, F. Roberti, R. Carelli, Admittance controller with spatial modulation for assisted locomotion using a smart walker. *J. Intell. Robot. Syst.*, 1 (2018)
16. G. Wasson, J. Gunderson, S. Graves, R. Felder, An assistive robotic agent for pedestrian mobility, in *Proceedings of the Fifth International Conference on Autonomous Agents - AGENTS '01* (ACM Press, New York, USA, 2001), pp. 169–173
17. G. Lee, E.J. Jung, T. Ohnuma, N.Y. Chong, B.J. Yi, JAIST Robotic Walker control based on a two-layered Kalman filter, in *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 3682–3687 (2011)
18. G.J. Lacey, D. Rodriguez-Losada, The evolution of Guido. *IEEE Robot. Autom. Mag.* **15**(4), 75–83 (2008)
19. L. Palopoli, A. Argyros, J. Birchbauer, A. Colombo, D. Fontanelli, A. Legay, A. Garulli, A. Giannitrapani, D. Macii, F. Moro, P. Nazemzadeh, P. Padeleris, R. Passerone, G. Poier, D. Prattichizzo, T. Rizano, L. Rizzon, S. Scheggi, S. Sedwards, Navigation assistance and guidance of older adults across complex public spaces: the DALi approach. *Intell. Serv. Robot.* **8**, 77–92 (2015)

20. J. Alves, E. Seabra, I. Caetano, C.P. Santos, Overview of the ASBGo++ smart walker, in *2017 IEEE 5th Portuguese Meeting on Bioengineering (ENBENG)* (IEEE, 2017), pp. 1–4
21. A. Wachaja, P. Agarwal, M. Zink, M.R. Adame, K. Möller, W. Burgard, Navigating blind people with walking impairments using a smart walker. *Autonomous Robots* **41**, 555–573 (2017)
22. S.D. Sierra M., M. Garzón, M. Múnera, C.A. Cifuentes, Human–Robot–environment interaction interface for smart walker assisted gait: AGoRA walker. *Sensors* **19**, 2897 (2019)
23. S.G. Tzafestas, Mobile robot kinematics, in *Introduction to Mobile Robot Control* (Elsevier, 2014), pp. 31–67
24. C.Z. Resende, R. Carelli, T.F. Bastos-Filho, M. Sarcinelli-Filho, A new positioning and path following controller for unicycle mobile robots, in *2013 16th International Conference on Advanced Robotics (ICAR)* (IEEE, 2013), pp. 1–6
25. M.R. Islam, B. Brahmi, T. Ahmed, M. Assad-Uz-Zaman, M.H. Rahman, *Exoskeletons in Upper Limb Rehabilitation: A Review to Find Key Challenges to Improve Functionality* (Elsevier Inc., 2020)
26. A.J. Young, D.P. Ferris, State of the art and future directions for lower limb robotic exoskeletons. *IEEE Trans. Neural Syst. Rehab. Eng.* **25**(2), 171–182 (2017)
27. P. Zhang, *Transducers and Valves*, 1st edn. (Peng Zhang, 2010)
28. J. Varghese, V.M. Akhil, P.K. Rajendrakumar, K.S. Sivanandan, A rotary pneumatic actuator for the actuation of the exoskeleton knee joint. *Theor. Appl. Mech. Lett.* **7**(4), 222–230 (2017)
29. R.J. Varghese, D. Freer, F. Deligianni, J. Liu, G.-Z. Yang, Wearable robotics for upper-limb rehabilitation and assistance, in *Wearable Technology in Medicine and Health Care* (Elsevier, 2018), pp. 23–69
30. T. Sénac, A. Lelevé, R. Moreau, C. Navales, L. Nouaille, M.T. Pham, P. Vieyres, A review of pneumatic actuators used for the design of medical simulators and medical tools. *Multimodal Technol. Interact.* **3**(3), 1–22 (2019)
31. R. Bishop, *The Mechatronics Handbook, Second Edition - 2 Volume Set*. Mechatronics Handbook 2e (Taylor & Francis, 2002)
32. J. Segil, *Handbook of Biomechanics* (Academic Press, 2019)
33. I. Sardellitti, E. Cattin, S. Roccella, F. Vecchi, M.C. Carrozza, P. Dario, P.K. Artemiadis, K.J. Kyriakopoulos, Description, characterization and assessment of a bio-inspired shoulder joint-first link robot for neuro-robotic applications, in *Proceedings of the First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechanics, 2006, BioRob 2006*, vol. 2006, no. February 2001, pp. 112–117 (2006)
34. J. Zhu, Y. Wang, J. Jiang, B. Sun, H. Cao, Unidirectional variable stiffness hydraulic actuator for load-carrying knee exoskeleton. *Int. J. Adv. Robot. Syst.* **14**(1), 1–12 (2017)
35. H. Kazerooni, R. Steger, L. Huang, Hybrid control of the Berkeley Lower Extremity Exoskeleton (BLEEX). *Int. J. Robot. Res.* **25**(5-6), 561–573 (2006)
36. A.B. Zoss, H. Kazerooni, A. Chu, Biomechanical design of the Berkeley Lower Extremity Exoskeleton (BLEEX). *IEEE/ASME Trans. Mechatron.* **11**(2), 128–138 (2006)
37. C. Bayón, O. Ramírez, J.I. Serrano, M.D. Castillo, A. Pérez-Somarriba, J.M. Belda-Lois, I. Martínez-Caballero, S. Lerma-Lara, C. Cifuentes, A. Frizzera, E. Rocon, Development and evaluation of a novel robotic platform for gait rehabilitation in patients with Cerebral Palsy: CPWalker. *Robot. Autonom. Syst.* **91**, 101–114 (2017)
38. H. Kim, Y.J. Shin, J. Kim, Kinematic-based locomotion mode recognition for power augmentation exoskeleton. *Int. J. Adv. Robot. Syst.* **14**(5), 1–14 (2017)
39. E.F. Kececi, Actuators, in *Mechatronic Components* (Elsevier, 2019), pp. 145–154
40. D. GRAY, Chapter 7 - actuators, in *Centralized and Automatic Controls in Ships*, ed. by D. GRAY (Pergamon, 1966), pp. 44–54
41. S. Arumugom S. Muthuraman, V. Ponselvan, Modeling and application of series elastic actuators for force control multi legged robots. *J. Comput.* **1**(1), 26–33 (2009)
42. M.A. Gull, S. Bai, T. Bak, A review on design of upper limb exoskeletons. *Robotics* **9**(1), 1–35 (2020)

43. S. Hussain, P.K. Jamwal, M.H. Ghayesh, S.Q. Xie, Assist-as-needed control of an intrinsically compliant robotic gait training orthosis. *IEEE Trans. Ind. Electron.* **64**(2), 1675–1685 (2017)
44. J. Choi, B. Na, P.-G. Jung, D.-w. Rha, K. Kong, WalkON suit. *IEEE Robot. Autom. Mag.* no. November, 75–86 (2017)
45. Y.-L. Park, B.-r. Chen, N. O. Pérez-Arancibia, D. Young, L. Stirling, R.J. Wood, E.C. Goldfield, R. Nagpal, Design and control of a bio-inspired soft wearable robotic device for ankle–foot rehabilitation. *Bioinspiration Biomimetics* **9**, 016007 (2014)
46. S. Galle, P. Malcolm, S.H. Collins, D. De Clercq, Reducing the metabolic cost of walking with an ankle exoskeleton: interaction between actuation timing and power. *J. NeuroEng. Rehab.* **14**(1), 1–16 (2017)
47. H. Lee, P.W. Ferguson, J. Rosen, *Chapter 11. Lower Limb Exoskeleton Systems—Overview* (INC, 2020)
48. A. Esquenazi, M. Talaty, A. Packel, M. Saulino, The ReWalk powered exoskeleton to restore ambulatory function to individuals with thoracic-level. *Am. J. Phys. Med. Rehab.*, 911–921 (2012)
49. E. Bionics, Ekso GT (2020)
50. Indego, Indego Exoskeleton (2020)
51. S.O. Schrade, K. Dätwyler, M. Stücheli, K. Studer, D.A. Türk, M. Meboldt, R. Gassert, O. Lamberg, Development of VariLeg, an exoskeleton with variable stiffness actuation: First results and user evaluation from the CYBATHLON 2016 Olivier Lamberg; Roger Gassert. *J. NeuroEng. Rehab.* **15**(1), 1–18 (2018)
52. C. Bayón, O. Ramírez, F. Mollà, J. Serrano, M. Del Castillo, J. Belda-Lois, R. Poveda, R. Raya, T. Martín Lorenzo, I. Martínez Caballero, S. Lerma Lara, C. Cifuentes, A. Frizera, E. Rocon, CPWalker, robotic platform for gait rehabilitation and training in patients with cerebral palsy. *IEEE Trans. Neural Syst. Rehab. Eng.* (Under Review), 3736–3741 (2015)
53. M.D. Sánchez-Manchola, L.J. Arciniegas Mayag, M. Munera, C.A. Garcia, Impedance-based backdrivability recovery of a lower-limb exoskeleton for knee rehabilitation, in *4th IEEE Colombian Conference on Automatic Control: Automatic Control as Key Support of Industrial Productivity, CCAC 2019 - Proceedings* (Medellin, Colombia, 2019), pp. 1–6
54. M. Talaty, A. Esquenazi, J.E. Briceno, Differentiating ability in users of the ReWalk™ powered exoskeleton: An analysis of walking kinematics. *IEEE International Conference on Rehabilitation Robotics* (2013)
55. A. Ramanujam, C.M. Ciriigliaro, E. Garbarini, P. Asselin, R. Pilkar, G.F. Forrest, Neuromechanical adaptations during a robotic powered exoskeleton assisted walking session. *J. Spinal Cord Med.* **41**(5), 518–528 (2018)
56. W. Zou, N. Yu, Modeling and control of a cable-driven series elastic actuator (2018)
57. W.M. Dos Santos, A.A.G. Siqueira, *Impedance Control of a Rotary Series Elastic Actuator for Knee Rehabilitation*, vol. 19 (IFAC, 2014)
58. K. Isik, S. He, J. Ho, L. Sentis, Re-engineering a high performance electrical series elastic actuator for low-cost industrial applications. *Actuators*, 1–16 (2017)
59. R. Van Ham, B. Vanderborght, M. Van Damme, B. Verrelst, D. Lefeber, MACCEPA, the mechanically adjustable compliance and controllable equilibrium position actuator: Design and implementation in a biped robot. *Robot. Autonom. Syst.* **55**, 761–768 (2007)
60. P. Chelle, V. Grosu, P. Beyl, A. Mathys, R. Van Ham, M. Van Damme, B. Vanderborght, D. Lefeber, The MACCEPA actuation system as torque actuator in the gait rehabilitation robot ALTACRO, in *2010 3rd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics* (IEEE, 2010), pp. 27–32
61. T. Bacek, M. Moltedo, K. Langlois, G.A. Prieto, M.C. Sanchez-Villamañan, J. Gonzalez-Vargas, B. Vanderborght, D. Lefeber, J.C. Moreno, BioMot exoskeleton - Towards a smart wearable robot for symbiotic human-robot interaction, in *IEEE International Conference on Rehabilitation Robotics*, pp. 1666–1671 (2017)
62. J.F. Veneman, R. Ekkelenkamp, R. Kruidhof, F.C. Van Der Helm, H. Van Der Kooij, A series elastic- and Bowden-cable-based actuation system for use as torque actuator in exoskeleton-type robots. *Int. J. Robot. Res.* **25**(3), 261–281 (2006)

63. J.C. Perry, J. Rosen, S. Burns, Upper-limb powered exoskeleton design. *IEEE/ASME Trans. Mechatron.* **12**(4), 408–417 (2007)
64. L.M. Mooney, H.M. Herr, Biomechanical walking mechanisms underlying the metabolic reduction caused by an autonomous exoskeleton. *J. NeuroEng. Rehab.* **13**(1), 1–12 (2016)
65. J.E. Duarte, K. Schmidt, R. Riener, J.E. Duarte, K. Schmidt, R. Riener, J.E. Duarte, K. Schmidt, R. Riener, The Myosuit : textile-powered mobility The Myosuit : textile-powered mobility. *IFAC-PapersOnLine* **51**(34), 242–243 (2019)
66. H. Kozima, M.P. Michalowski, C. Nakagawa, Keepon: A playful robot for research, therapy, entertainment. *Int. J. Soc. Robot.* **1**(1), 3–18 (2009)
67. J. Saldien, K. Goris, S. Yilmazyildiz, W. Verhelst, D. Lefeber, On the design of the huggable robot Probo. *J. Phys. Agents* **2**(2), 3–11 (2008)
68. K. Goris, J. Saldien, B. Vanderborgh, D. Lefeber, Mechanical design of the huggable robot Probo. *Int. J. Humanoid Robot.* **8**(3), 481–511 (2011)
69. K. Dautenhahn, C.L. Nehaniv, M.L. Walters, B. Robins, H. Kose-Bagci, N.A. Mirza, M. Blow, KASPAR - a minimally expressive humanoid robot for human-robot interaction research. *Appl. Bionics Biomech.* **6**(3-4), 369–397 (2009)
70. S. Costa, H. Lehmann, K. Dautenhahn, B. Robins, F. Soares, Using a humanoid robot to elicit body awareness and appropriate physical interaction in children with autism. *Int. J. Soc. Robot.* **7**(2), 265–278 (2015)
71. Y. Fernaeus, M. Håkansson, M. Jacobsson, S. Ljungblad, How do you play with a robotic toy animal? A long-term study of Pleo, in *Proceedings of IDC2010: The 9th International Conference on Interaction Design and Children*, pp. 39–48 (2010)
72. J. Dimas, I. Leite, A. Pereira, P. Cuba, R. Prada, A. Paiva, Pervasive Pleo: Long-term attachment with artificial pets, in *Workshop on Playful Experiences in Mobile HCI* (2010)
73. S. Coşar, M. Fernandez-Carmona, R. Agrigoroaie, J. Pages, F. Ferland, F. Zhao, S. Yue, N. Bellotto, A. Tapus, ENRICHME: Perception and interaction of an assistive robot for the elderly at home. *Int. J. Soc. Robot.* **12**(3), 779–805 (2020)
74. M. Eskenazi, L. Devillers, J. Mariani, (eds.), *Advanced Social Interaction with Agents*, vol. 510 of *Lecture Notes in Electrical Engineering* (Springer International Publishing, Cham, 2019)
75. D. Fischinger, P. Einramhof, K. Papoutsakis, W. Wohlking, P. Mayer, P. Panek, S. Hofmann, T. Koertner, A. Weiss, A. Argyros, M. Vincze, Hobbit, a care robot supporting independent living at home: First prototype and lessons learned. *Robot. Autonom. Syst.* **75**, 60–78 (2016)
76. W.D. Stiehl, J. Lieberman, C. Breazeal, L. Basel, R. Cooper, H. Knight, L. Lalla, A. Maymin, S. Purchase, The Huggable: A therapeutic robotic companion for relational, affective touch, in *2006 3rd IEEE Consumer Communications and Networking Conference, CCNC 2006*, vol. 2, pp. 1290–1291 (2006)
77. A. Di Nuovo, D. Conti, G. Trubia, S. Buono, S. Di Nuovo, Deep learning systems for estimating visual attention in robot-assisted therapy of children with autism and intellectual disability. *Robotics* **7**(2), 25 (2018)
78. O.A. Henkemans, B.P. Bierman, J. Janssen, R. Looije, M.A. Neerinx, M.M. van Dooren, J.L. de Vries, G.J. van der Burg, S.D. Huisman, Design and evaluation of a personal robot playing a self-management education game with children with diabetes type 1. *Int. J. Human Comput. Stud.* **106**, no. January, 63–76 (2017)
79. Z. Shen, Y. Wu, Investigation of practical use of humanoid robots in elderly care centres, in *HAI 2016 - Proceedings of the 4th International Conference on Human Agent Interaction*, pp. 63–66 (2016)
80. T. Washio, A. Sakurai, K. Nakajima, H. Takeda, S. Tojo, M. Yokoo, (eds.), *New Frontiers in Artificial Intelligence*, vol. 4012 of *Lecture Notes in Computer Science* (Springer, Berlin, Heidelberg, 2006)

81. T. Shibata, T. Mitsui, K. Wada, A. Touda, T. Kumasaka, K. Tagami, K. Tanie, Mental commit robot and its application to therapy of children, in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM*, vol. 2, no. July, pp. 1053–1058 (2001)
82. G.A. Pratt, M.M. Williamson, Series elastic actuators, in *IEEE International Conference on Intelligent Robots and Systems*, vol. 1, pp. 399–406 (1995)
83. J.J. Cabibihan, H. Javed, M. Ang, S.M. Aljunied, Why robots? A survey on the roles and benefits of social robots in the therapy of children with autism. *Int. J. Soc. Robot.* **5**(4), 593–618 (2013)
84. T. Mikolajczyk, I. Ciobanu, D.I. Badea, A. Iliescu, S. Pizzamiglio, T. Schauer, T. Seel, P.L. Seiciu, D.L. Turner, M. Berceanu, Advanced technology for gait rehabilitation: An overview. *Adv. Mech. Eng.* **10**, 1–19 (2018)
85. A. Trujillo-León, W. Bachtá, J. Castellanos-Ramos, F. Vidal-Verdú, Assistive handlebar based on tactile sensors: Control inputs and human factors. *Sensors* **18**, 2471 (2018)
86. S.S. Banala, S.K. Agrawal, Active leg exoskeleton (ALEX) for gait rehabilitation of motor-impaired patients, in *IEEE Int Conf Rehabil Robot 2007*, 401–7, *Rehabilitation Robotics*, vol. 00, no. c (2007)
87. W. Huo, S. Mohammed, Y. Amirat, K. Kong, Fast gait mode detection and assistive torque control of an exoskeletal robotic orthosis for walking assistance. *IEEE Trans. Robot.* **34**(4), 1035–1052 (2018)
88. A. Roy, H.I. Krebs, S.L. Patterson, T.N. Judkins, I. Khanna, L.W. Forrester, R.M. Macko, N. Hogan, Measurement of human ankle stiffness using the Anklebot, in *2007 IEEE 10th International Conference on Rehabilitation Robotics, ICORR'07*, vol. 00, no. c, pp. 356–363 (2007)
89. R. Bionics, REX (2019)
90. H. Kawamoto, S. Lee, S. Kanbe, Y. Sankai, Power assist method for HAL-3 using EMG-based feedback controller, in *SMC'03 Conference Proceedings. 2003 IEEE International Conference on Systems, Man and Cybernetics. Conference Theme - System Security and Assurance (Cat. No.03CH37483)*, vol. 2 (IEEE, 2003), pp. 1648–1653
91. M.N. Victorino, X. Jiang, C. Menon, *Wearable Technologies and Force Myography for Healthcare* (Elsevier Inc., 2018)
92. H. Nguyen, K. Lebel, S. Bogard, E. Goubault, P. Boissy, C. Duval, Using inertial sensors to automatically detect and segment activities of daily living in people with Parkinson's disease. *IEEE Trans. Neural Syst. Rehab. Eng.* **26**(1), 197–204 (2018)
93. J.C. Moreno, F. Brunetti, E. Navarro, A. Forner-Cordero, J.L. Pons, Analysis of the human interaction with a wearable lower-limb exoskeleton. *Appl. Bion. Biomech.* **6**(2), 245–256 (2009)
94. M. Sánchez-Manchola, M.J. Bernal, M. Munera, C.A. Cifuentes, Gait phase detection for lower-limb exoskeletons using foot motion data from a single inertial measurement unit in hemiparetic individuals. *Sensors (Switzerland)* **19**(13), 2988 (2019)
95. M. Tariq, H. Majeed, M.O. Beg, F.A. Khan, A. Derhab, Accurate detection of sitting posture activities in a secure IoT based assisted living environment. *Future Gener. Comput. Syst.* **92**, 745–757 (2019)
96. B. Chen, C.H. Zhong, X. Zhao, H. Ma, X. Guan, X. Li, F.Y. Liang, J.C.Y. Cheng, L. Qin, S.W. Law, W.H. Liao, A wearable exoskeleton suit for motion assistance to paralysed patients. *J. Orthopaedic Transl.* **11**, no. March, 7–18 (2017)
97. A.C. Villa-Parra, D. Delisle-Rodriguez, J.S. Lima, A. Frizzera-Neto, T. Bastos, Knee impedance modulation to control an active orthosis using insole sensors. *Sensors (Switzerland)* **17**(12), 2751 (2017)
98. C. Bayon, O. Ramirez, M. Del Castillo, J. Serrano, R. Raya, J. Belda-Lois, R. Poveda, F. Molla, T. Martin, I. Martinez, S. Lerma Lara, E. Rocon, CPWalker: Robotic platform for gait rehabilitation in patients with Cerebral Palsy, in *2016 IEEE International Conference on Robotics and Automation (ICRA)* (IEEE, 2016), pp. 3736–3741

99. A. Morris, R. Donamukkala, A. Kapuria, A. Steinfeld, J. Matthews, J. Dunbar-Jacob, S. Thrun, A robotic walker that provides guidance, in *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, vol. 1, pp. 25–30 (2003)
100. M. Spenko, H. Yu, S. Dubowsky, Robotic personal aids for mobility and monitoring for the elderly. *IEEE Trans. Neural Syst. Rehab. Eng.* **14**(3), 344–351 (2006)
101. W.-H. Mou, M.-F. Chang, C.-K. Liao, Y.-H. Hsu, S.-H. Tseng, L.-C. Fu, Context-aware assisted interactive robotic walker for Parkinson’s disease patients, in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IEEE, 2012)*, pp. 329–334
102. J. Paulo, P. Peixoto, U.J. Nunes, ISR-AIWALKER: Robotic walker for intuitive and safe mobility assistance and gait analysis. *IEEE Trans. Human Mach. Syst.* **47**(6), 1110–1122 (2017)
103. C. Huang, G. Wasson, M. Alwan, P. Sheth, Shared navigational control and user intent detection in an intelligent walker, in *AAAI Fall 2005*, no. April (2005)
104. E. Efthimiou, S.E. Fotinea, T. Goulas, M. Koutsombogera, P. Karioris, A. Vacalopoulou, I. Rodomagoulakis, P. Maragos, C. Tzafestas, V. Pitsikalis, Y. Koumpouros, A. Karavasili, P. Siavelis, F. Koureta, D. Alexopoulou, The MOBOT rollator human-robot interaction model and user evaluation process, in *2016 IEEE Symposium Series on Computational Intelligence, SSCI 2016* (2017)