



Sharif University of Technology
Scientia Iranica
Transactions A: Civil Engineering
<http://scientiairanica.sharif.edu>



Invited Paper

Employing a novel gait pattern generator on a social humanoid robot

A. Meghdari^{a,*}, S. Behzadipour^b, and M. Abedi^b

a. *Social & Cognitive Robotics Laboratory, Center of Excellence in Design, Robotics, and Automation (CEDRA), School of Mechanical Engineering, Sharif University of Technology, Tehran, Iran.*

b. *School of Mechanical Engineering, Sharif University of Technology, Tehran, Iran.*

Received 11 February 2019; accepted 11 March 2019

KEYWORDS

Social robots;
 Bipedal robots;
 Gait pattern
 generating;
 Inverse kinematics;
 Static stability
 condition.

Abstract. This paper presents a novel Gait Pattern Generator (GPG) developed for the “Alice” social humanoid robot, which up to now lacked an appropriate walking pattern. Due to the limitations of this robot, the proposed gate pattern generator was formulated based on a nine-mass model to decrease the modeling errors and the inverse kinematics of the whole lower-body was solved in such a way that the robot remained statically stable during the movements. The main challenge of this work was to solve the inverse kinematics of a 7-link chain with 12 degrees of freedom. For this purpose, a new graphical-numerical technique has been provided using the definition of the kinematic equations of the robot joints’ Cartesian coordinates. This method resulted in a significant increase in the solution rate of calculations. Finally, a novel algorithm was developed for step-by-step displacement of the robot towards a desired destination in a two-dimensional space. Performance of the proposed gate pattern generator was evaluated both with a model of the robot in a MATLAB Simulink environment and in real experiments with the Alice humanoid robot.

© 2019 Sharif University of Technology. All rights reserved.

1. Introduction

In recent years, social humanoid robots have emerged with the goal of improving interaction between humans and environment [1-3]. Despite the complexities of manufacturing and controlling bipedal and humanoid robots, research on this class of robots is ongoing and growing [4-6]. Gait pattern generation is one of the challenges of working with bipedal robots. High degrees of freedom, robot stability, and unique characteristics of each robot are some of the difficulties

of GPG planning. Some important characteristics of the robot, which affect the GPG, are size and weight of the robot, number of Degrees of Freedom (DoF), sensors and actuators, data processing performance, and goals expected for the robot. However, gait pattern generating generally includes modelling, kinematic and dynamic analysis, stability condition, path planning, and controller design [7]. The final objective of a GPG is to generate robot joint trajectories that result in stable robot walking.

Stability is an important problem in GPG planning since ground contact forces must be compressive during the robot movement. In general, there are two types of stability conditions: static and dynamic [8-11]. In static stability, the Ground Projection of the Center of Mass (GCoM) of the robot is kept within the stability area (the convex hull of all contact points between the feet and the ground). Indeed, static

*. *Corresponding author.*
E-mail addresses: meghdari@sharif.edu (A. Meghdari);
Behzadipour@sharif.ir (S. Behzadipour);
majid.abedi92@yahoo.com (M. Abedi)

criterion is the oldest stability condition that was first suggested for a walking robot with massless legs [12]. However, it has been proven that it is necessary to consider the mass of the legs to determine the robot stability when the leg mass is great enough [13]. This condition is easier to implement, but the walking produced is usually too slow and not human-like because of excessive movement in the frontal plane. On the other hand, keeping the Zero-Moment-Point (ZMP) in the stability area is one of the dynamic conditions. Although ZMP is a normal stability criterion, it is not comprehensive. Thus, other criteria, such as Foot Rotation Indicator (FRI) and Centroidal Moment Pivot (CMP), have been derived to improve the ZMP method [14]. Also, due to the complex dynamics of a humanoid robot, it is impossible to generate a stable walking pattern in real time using these conditions. These conditions are usually used on a simple model of the robot. For this purpose, a one-mass model (linear inverted pendulum mode [15]), a two-mass model (gravity-compensated inverted pendulum mode [16]), and a three-mass model have been suggested [17,18]. Although some analytic gait planning is achieved using these simple models, it is necessary to compensate for the modelling error. To this end, force sensors applied to the soles of the robot are necessary to calculate the real ZMP position. However, if a more precise model (i.e., the multiple masses inverted pendulum mode [19]) is used, the ZMP equation becomes highly nonlinear and complex. In this case, the robot joint trajectories are calculated by a time-consuming iterative procedure, which makes it inappropriate for generating a real-time walking pattern.

Solving the Inverse Kinematics (IK) of humanoid robots is also challenging because of their high number of DoF. Most commercial bipedal robots have 12 DoF in their lower-body (6 DoF in each leg) [20,21]. There is no analytic IK solution for a whole lower-body with such a structure. However, by using the simpler models (one-, two-, or three-mass model), it is possible to solve the IK of each leg (2 six-DoF chains) separately. Using these simplifications, the trajectory of the body of the robot can be calculated analytically. While it is possible to provide a closed-form IK solution for each leg for most commercial biped robots (with 6 DoF in each leg) [22], this solution exists only if the 3 axes of the hip intersect each other at one point (that is, the hip is assumed as a spherical joint).

One should note that it is possible to generate a gait pattern analytically fast enough for use in real-time applications only if there is no offset at the hip of the robot and the mass of the legs is light enough to use in simple models. Otherwise, iterative methods are probably the only solutions. However, these methods are not generally effective because of the expensive computation and large amount of time

required to achieve the desired accuracy. Therefore, it is necessary to find a solution to reduce the iterations. For example, Limón et al. [23] provided a two-step approach for a six-DoF leg with a non-negligible offset at the hip. At First, an approximate IK solution was derived analytically regardless of the offset at the hip. Then, this result was used as the initial condition for a numerical refining procedure.

In this research, a GPG is developed for the humanoid robot “Alice” with the given Iranian name “Mina” [24], as shown in Figure 1. This robot has been used in autism treatment as a therapy assistant and in English as a Second Language (ESL) education [25–28]. Although this robot has the capability of walking, no walking pattern has been designed for it thus far. To resolve this limitation, we found a walking pattern for Alice and implemented it.

Considering the unique characteristics of this robot (especially the non-negligible offset at the hip, sensors, and actuators as well as the mass of the legs) discussed in Section 2, a static stability with high accuracy has been used in the proposed GPG. To this end, the GPG has been designed based on a nine-mass model. Using this highly accurate model makes it necessary to solve the IK of the whole lower-body. Hence, the main objective of this work was to solve the IK of a 12-DoF chain. This was not common in the previously proposed GPG, because there was no analytical method for this problem. Previous iterative methods for this case would be very time consuming, because forward kinematics (common in the numerical process of joint angles calculation) are very complicated for this 12-DoF chain. Also, it is not possible to find an approximate analytical solution as the initial condition for this chain by ignoring the offset at the hip (similar to the approach in [23]). Accordingly, a new graphical-numerical method is provided in this paper to reduce the calculation time. A set of kinematic equations based on the robot geometry are derived for this purpose. Finally, the GPG is completed with an algorithm for a step by step displacement of the robot towards a desired destination in a two-dimensional space [29].

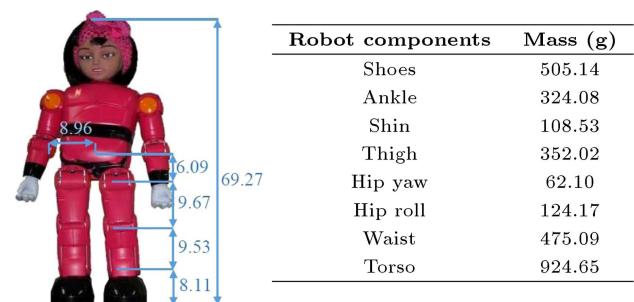


Figure 1. Alice “Mina” robot and its components (dimensions are in centimetres) [24].

2. The Alice “Mina” humanoid robot

Alice (renamed Mina for Persian dialogues) is an R50 model poorly designed and constructed by the Robokind Company [24,30]. It has a height of 69 cm; the masses of its components are presented in Figure 1. Also, it has 32 DoF. This robot, similar to most of the commercial bipedal robots, has 6 DoF in each leg (3 in the hip, one in the knee, and 2 in the ankle). Therefore, it is capable of walking on straight and curved paths, up and down stairs, and on sloped ground. These joints are actuated by Dynamixel RX-64. These servos are controlled by the internal computer of the robot. JavaScript is utilized to send the goal position of each joint to the internal computer. This is the only way of joints position control that is suggested for this robot. In this method, joint acceleration cannot be controlled and slow movements ensure that all joints will follow the desired trajectories, simultaneously. Also, for the mass of the robot legs (which is great enough to be considered) and due to the lack of force sensors in the robot soles (to compensate for the modelling error), a nine-mass model is used in the proposed GPG. These factors make it difficult to use a dynamic stability criterion. Fortunately, considering the goals expected for this robot [25–29], it does not need to walk too fast; thus, static stability has been used in this paper. Review of the construction of this robot (see Figure 1) shows a 6.09 cm offset at the hip. Regarding the length of the robot's leg, approximate gait planning for spherical hip joints, which neglects the offset, results in a significant error in the robot walk. Since there are no force sensors in its soles, there is no analytic IK solution for each leg and it is also impossible to compensate for the modelling error using a feedback control. In this paper, the proposed GPG is designed with regard to the mentioned important characteristics of this robot.

3. Gait pattern generating for the Alice robot

3.1. Robot modelling

Kinematic and dynamic modelling is the first step in GPG planning. A model of the robot, with 7 links and 8 joint, is shown in Figure 2(a). The dimensions corresponding to Figure 3 can be seen in Table 1. In this model, it is assumed that the body and shoes are located in vertical and horizontal directions, respectively. The first assumption allows the robot to walk like a human. Also, based on this assumption, the upper body is modelled by a single link (link 4). The second assumption ignores the height of the shoes. In this model, q_2 , q_3 , q_6 , and q_7 each have one degree of freedom and the other joints have 2 DoF. Also, the dynamic of the robot is modelled by 9 lumped masses. The weight and location of each lumped mass (see Figure 3) are presented in Table 1. In the next sections,

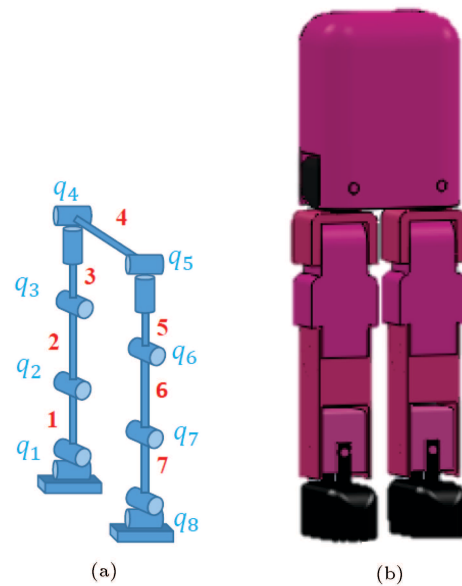


Figure 2. (a) A 7-link model of the robot with 9 concentrated masses. (b) Model of the robot designed in SolidWorks.

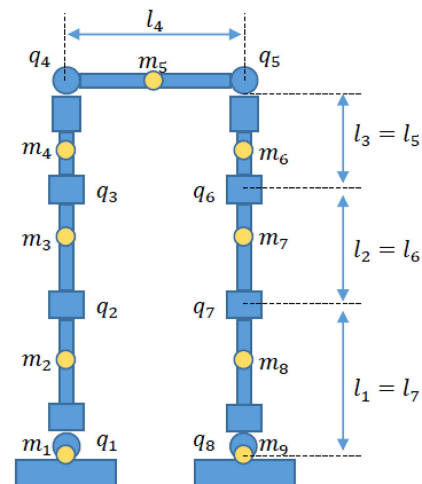


Figure 3. Masses and lengths of the model presented in Table 1.

the kinematics and dynamics of the robot are analyzed according to this model.

A simulation environment is needed to evaluate the proposed GPG before implementing it on the robot. For this purpose, a more accurate model of the robot is designed in SolidWorks (see Figure 2(b)). Then, it is imported into MATLAB Simulink to run the GPG [29].

3.2. Inverse kinematics

The purpose of this section is to calculate the robot joint angles in a desired posture. The desired posture is determined by conditions such as coordinates of the shoes and the GCoM. Due to the high number of DoF, an infinite number of solutions will be achieved.

In this study, inverse kinematics are solved by a

Table 1. Kinematic and dynamic characteristics of the 7-link model.

Model components	Length (cm)	Mass (kg)	Location of the center of mass (cm)
			$\vec{com}_i = [Xcom_i \ Ycom_i \ Zcom_i]^T$
Right shoe	0	$m_1 = 0.82922$	$\vec{com}_1 = \vec{q}_1$
Link 1	$l_1 = 9.57$	$m_2 = 0.10853$	$\vec{com}_2 = \vec{q}_2 + \frac{4.5}{l_1} + (\vec{q}_1 - \vec{q}_2)$
Link 2	$l_2 = 9.67$	$m_3 = 0.35202$	$\vec{com}_3 = \vec{q}_3 + \frac{4.63}{l_1} + (\vec{q}_2 - \vec{q}_3)$
Link 3	$l_3 = 6.09$	$m_4 = 0.0621$	$\vec{com}_4 = \frac{(q_3+q_4)}{2}$
Link 4	$l_4 = 8.98$	$m_5 = 1.6481$	$\vec{com}_5 = \frac{(q_4+q_5)}{2}$
Link 5	$l_5 = l_3$	$m_6 = m_4$	$\vec{com}_6 = \frac{(q_5+q_6)}{2}$
Link 6	$l_6 = l_4$	$m_7 = m_3$	$\vec{com}_7 = \vec{q}_6 + \frac{4.63}{l_1} + (\vec{q}_7 - \vec{q}_6)$
Link 7	$l_7 = l_1$	$m_8 = m_2$	$\vec{com}_8 = \vec{q}_7 + \frac{4.5}{l_1} + (\vec{q}_8 - \vec{q}_7)$
Left shoe	0	$m_9 = m_1$	$\vec{com}_8 = \vec{q}_8$

numerical method. Because the 3 axes of the hip do not intersect each other in one point in this robot, it is not possible to provide a closed-form solution for the inverse kinematics of each leg. Alternatively, the stability condition is an important criterion to define the desired postures of the robot. In this study, the static method is used as the stability condition. Thus, the inverse kinematics of the lower-body, i.e., a 12-DoF kinematic chain, is solved to keep the GCoM of the 9 lumped masses in the stability area.

3.2.1. Derivation of the kinematic equations

Derivation of the kinematic equations of the robot is the first step in the numerical solution for inverse kinematics. To this end, forward kinematic equations can be used. First, the coordinate frames (see Figure 4) are selected based on the commonly used convention of Denavit-Hartenberg (DH) [31]. Thus, the base frame ($X_0Y_0Z_0$) and the final frame ($X_{12}Y_{12}Z_{12}$) are attached

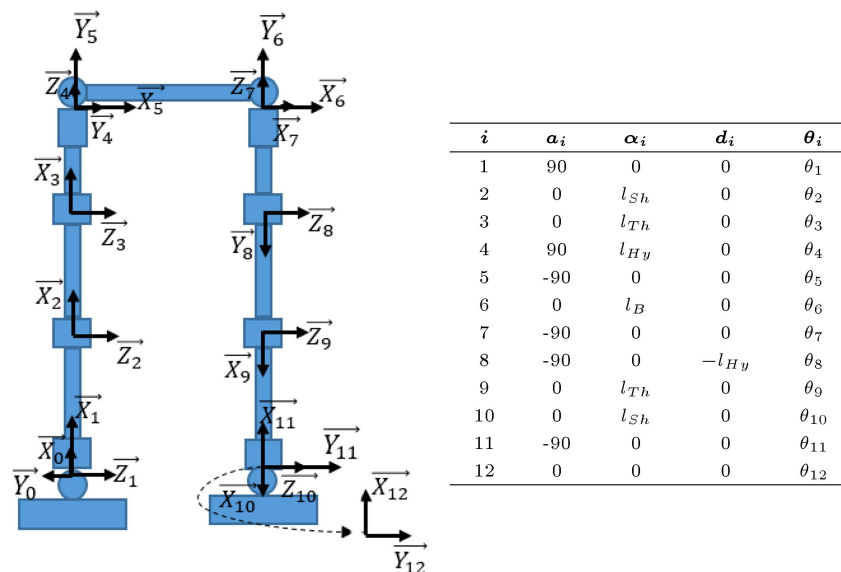
to the right and left shoes, respectively. Then, H_{12}^0 , the homogeneous transformation matrix that expresses the position and orientation of the left shoe with respect to the right shoe is obtained as follows:

$$H_{12}^0 = \prod_{i=1}^{12} H_i^{i-1}, \quad (1)$$

where, the homogeneous transformation, H_i^{i-1} , is represented as a product of four basic transformations based on the DH parameters in Figure 4:

$$H_i^{i-1} = Rot_{z,\theta_i} Trans_{z,d_i} Trans_{x,a_i} Rot_{x,\alpha_i}. \quad (2)$$

Now, it is possible to solve Eq. (1) numerically to obtain the joint angles. However, because these equations are very complicated, motion generation will be very time-consuming. In this paper, the kinematic equations are derived with the Cartesian coordination of joints ($\vec{q}_i =$

**Figure 4.** DH coordinate frame assignment and DH parameters for the lower body of the Alice robot.

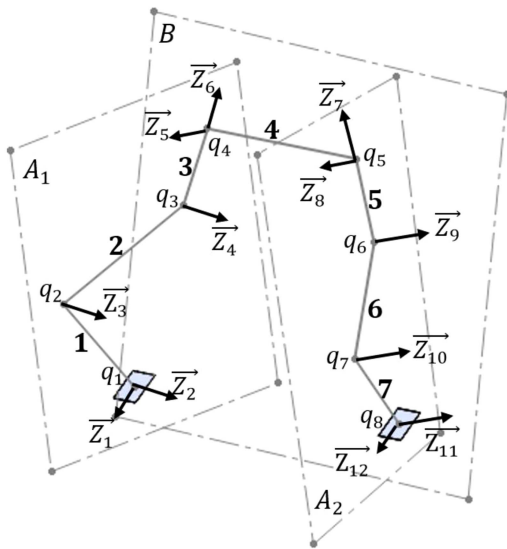


Figure 5. Connection of the model links.

$[x_i \ y_i \ z_i]^T$, $i = 1 : 8$). These equations are much easier to solve numerically, because they are shorter without trigonometric functions. First, the distance between the two joints is defined as follows:

$$\vec{P}_{ij} = [X_{ij} \ Y_{ij} \ Z_{ij}]^T = \vec{q}_j - \vec{q}_i. \quad (3)$$

Then, the following 7 equations are easily obtained:

$$|\vec{P}_{ij}| = l_i^2, \quad j = i + 1, \quad i = 1 : 7. \quad (4)$$

However, it is difficult to derive equations in Cartesian coordinates that indicate how the links are connected to each other by the joints. In the robot structure (see Figure 5), links 1, 2, and 3 always remain coplanar (plane A_1). Also, links 3, 4, and 5 and links 5, 6, and 7 constitute planes B and A_2 , respectively. The following equations indicate these conditions:

$$(\vec{P}_{12} \times \vec{P}_{23}) \cdot \vec{P}_{34} = 0, \quad (5)$$

$$(\vec{P}_{34} \times \vec{P}_{45}) \cdot \vec{P}_{56} = 0, \quad (6)$$

$$(\vec{P}_{56} \times \vec{P}_{67}) \cdot \vec{P}_{78} = 0. \quad (7)$$

Robot kinematics is fully expressed by Eqs. (4) to (7). But, the system still has 8 degrees of redundancy. Therefore, an infinite number of solutions will be achieved for a desired posture. However, only some of the solutions are acceptable, since an acceptable solution must satisfy the modelling assumptions and stability condition. According to the modelling assumptions, the body and shoes must be in vertical and horizontal directions, respectively. Also, the GCoM of the 9 lumped masses should remain in the stability area

to keep the robot stable during the movement. In this study, an acceptable solution is selected by defining 8 auxiliary equations.

At First, the solutions associated with robot stability should be selected. For this purpose, the static method is used as the stability condition, regardless of the effects of velocity and acceleration in a slow walk. Suppose that $\vec{comd} = [Xcomd \ Ycomd \ 0]^T$ is the desired location of the GCoM. Eqs. (8) and (9) lead to robot stability:

$$\sum_{i=1}^9 m_i Xcom_i - Xcomd \sum_{i=1}^9 m_i = 0, \quad (8)$$

$$\sum_{i=1}^9 m_i Ycom_i - Ycomd \sum_{i=1}^9 m_i = 0. \quad (9)$$

Also, the body of the robot should be in the vertical direction. Thus, first, plane B should be perpendicular to the ground and then, joints q_4 and q_5 must be at the same height. Eqs. (10) and (11) keep the robot body upright:

$$Nb_z = 0, \quad (10)$$

$$z_4 = z_5 = H. \quad (11)$$

Nb_z is the Z -component of the vector orthogonal to the plane B (\vec{n}_B), which is obtained as follows:

$$\vec{n}_B = [Nb_x \ Nb_y \ Nb_z]^T = \vec{P}_{43} \times \vec{P}_{45}. \quad (12)$$

H is the desired height of the body. This has a major impact on the torque, temperature, and energy consumption of the motors.

The orientation of the planes A_1 , B , and A_2 is another important point. Planes A_1 and A_2 are perpendicular to plane B when the robot walks in a straight line. However, when the robot simultaneously walks and rotates around the Z -axis, the orientation of the planes must be controlled.

The orientation of the body is determined by vector \vec{P}_{45} . To this end, vector $\vec{n}'_B = [Nb'_x \ Nb'_y \ 0]^T$ is defined in the X - Y plane. The angle θ_B between \vec{n}'_B and the Y -axis is set as the rotation of the body around the Z -axis. Eq. (13) is used to make \vec{P}_{45} parallel to \vec{n}'_B :

$$Nb'_x(y_5 - y_4) - Nb'_y(x_5 - x_4) = 0. \quad (13)$$

Similarly, orientation of planes A_1 and A_2 is determined by the projection of vectors \vec{n}_{A_1} and \vec{n}_{A_2} in the X - Y plane. \vec{n}_{A_1} and \vec{n}_{A_2} are the vectors orthogonal to planes A_1 and A_2 , respectively. Thus, θ_{A_1} and θ_{A_2} (the angles between the Y -axis and the projection of the vectors \vec{n}_{A_1} and \vec{n}_{A_2} in the X - Y plane) will be adjusted.

Defining the 8 auxiliary conditions decreased the degrees of redundancy of the system to zero, significantly reducing the number of solutions while allowing the desired posture for the robot to be achieved.

3.2.2. Calculation of the joints' coordinates

At this stage, the kinematics and auxiliary equations are utilized to achieve the desired posture for the robot. To this end, the unknown variables (coordinates of joints) must be calculated. The specified variables are \vec{q}_1 and \vec{q}_8 , because they are two inputs to the inverse kinematics process. Also, through Eq. (9), z_4 and z_5 are specified by the body height. Therefore, 16 other equations are utilized to calculate 16 other unknown variables. In this study, the Newton-Raphson method is applied to solving this system of equations. Thus, if the parameters \vec{q}_1 , \vec{q}_8 , \vec{comd} , θ_{A_1} , θ_{A_2} , and θ_B (input parameters to the IK process) are determined, starting from an initial condition, coordinates of the joints will be achieved.

3.2.3. Calculation of the joint angles

At the next stage, the joint angles which correspond to the coordinates of the joints must be determined. The following relations are derived based on the model geometry (see Figure 6):

$$\theta_3 = \cos^{-1} \left(\frac{\vec{P}_{21} \cdot \vec{P}_{32}}{|\vec{P}_{21}| |\vec{P}_{32}|} \right), \quad (14)$$

$$\theta_4 = w_1 \cos^{-1} \left(\frac{\vec{P}_{43} \cdot \vec{P}_{32}}{|\vec{P}_{43}| |\vec{P}_{32}|} \right), \quad (15)$$

$$\theta_5 = \frac{\pi}{2} - \cos^{-1} \left(\frac{\vec{n}_{A_1} \cdot \vec{n}_B}{|\vec{n}_{A_1}| |\vec{n}_B|} \right), \quad (16)$$

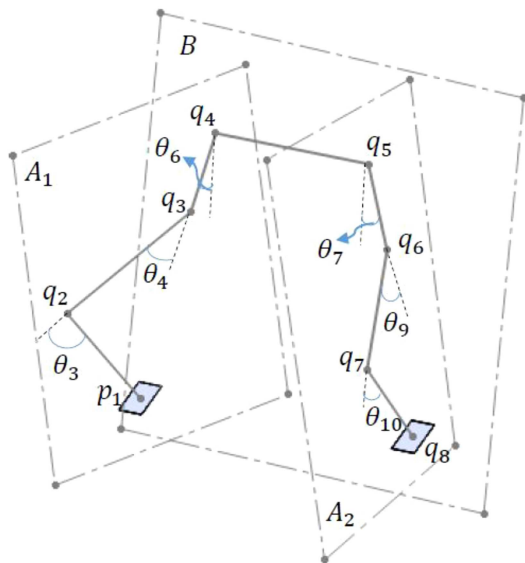


Figure 6. The angles θ_3 , θ_3 , θ_6 , θ_7 , θ_9 , and θ_{10} .

$$\theta_6 = \cos^{-1} \left(\frac{\vec{P}_{43} \cdot \vec{P}_{45}}{|\vec{P}_{43}| |\vec{P}_{45}|} \right) - \frac{\pi}{2}, \quad (17)$$

$$\theta_7 = \frac{\pi}{2} - \cos^{-1} \left(\frac{\vec{P}_{54} \cdot \vec{P}_{56}}{|\vec{P}_{54}| |\vec{P}_{56}|} \right), \quad (18)$$

$$\theta_8 = \frac{\pi}{2} - \cos^{-1} \left(\frac{\vec{n}_{A_2} \cdot \vec{n}_B}{|\vec{n}_{A_2}| |\vec{n}_B|} \right), \quad (19)$$

$$\theta_4 = w_2 \cos^{-1} \left(\frac{\vec{P}_{56} \cdot \vec{P}_{67}}{|\vec{P}_{56}| |\vec{P}_{67}|} \right), \quad (20)$$

$$\theta_{10} = \cos^{-1} \left(\frac{\vec{P}_{78} \cdot \vec{P}_{67}}{|\vec{P}_{78}| |\vec{P}_{67}|} \right). \quad (21)$$

In the above relations, considering the range of motion of the robot's joints, the \cos^{-1} function is used to calculate the angles. The signs of θ_4 and θ_9 are corrected by w_1 and w_2 as follows:

$$w_1 = \text{sgn}(Nb_x \cdot x_{42} + Nb_y \cdot y_{42}), \quad (22)$$

$$w_2 = \text{sgn}(Nb_x \cdot x_{47} + Nb_y \cdot y_{47}). \quad (23)$$

To calculate the angles θ_1 and θ_2 , three vectors \vec{f}_1 , \vec{t}_1 , and \vec{n}_1 are defined as shown in Figure 7; \vec{f}_1 is perpendicular to \vec{n}_{A_1} and \vec{P}_{12} ; \vec{t}_1 is the vector along the common line of A_1 and the ground; and \vec{n}_1 is perpendicular to \vec{Z} (the vector perpendicular to the ground) and \vec{t}_1 . Thus, these three vectors are calculated through the following equations:

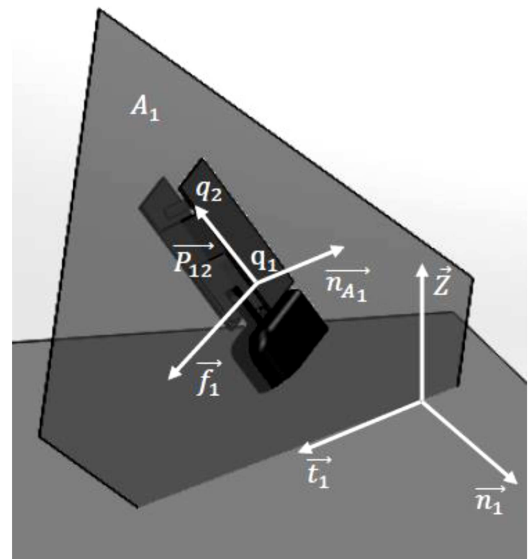


Figure 7. Definition of the vectors \vec{f}_1 , \vec{t}_1 , and \vec{n}_1 .

$$\vec{f}_1 = [fx_1 \quad fy_1 \quad fz_1]^T = \vec{P}_{12} \times \vec{n}_{A_1}, \quad (24)$$

$$\vec{t}_1 = \vec{Z} \times \vec{n}_{A_1}, \quad (25)$$

$$\vec{n}_1 = [nx_1 \quad ny_1 \quad nz_1]^T = \vec{t}_1 \times \vec{Z}. \quad (26)$$

When θ_1 and θ_2 are equal to zero, \vec{Z}_1 and \vec{Z}_2 are parallel to \vec{f}_1 and \vec{n}_{A_1} , respectively. Therefore, the right shoe is moved to a horizontal position by two successive rotations with the angles θ_1 and θ_2 as follows:

$$\theta_1 = \text{sgn}(nz_1) \cos^{-1} \left(\frac{\vec{n}_{A_1} \cdot \vec{n}_1}{|\vec{n}_{A_1}| |\vec{n}_1|} \right), \quad (27)$$

$$\theta_2 = \text{sgn}(fz_1) \cos^{-1} \left(\frac{\vec{f}_1 \cdot \vec{t}_1}{|\vec{f}_1| |\vec{t}_1|} \right). \quad (28)$$

In the same way, θ_{11} and θ_{12} are calculated to move the left shoe to a horizontal position. Thus, it is found that if the parameters \vec{q}_1 , \vec{q}_8 , \vec{comd} , θ_{A_1} , θ_{A_2} , and θ_B (the inputs to inverse kinematics) are determined, joint angles (the outputs of inverse kinematics) will be achieved. Using this ability, the GPG is designed in the next section.

4. GPG planning

The goal of the GPG is to move the Alice robot to a desired position in the horizontal plane. Thus, the robot displacements consist of moving along the X - and Y -axis ($\vec{g} = X\vec{I} + Y\vec{J}$) and rotating around the Z -axis

(θ). To achieve this goal, the GPG sends appropriate commands to the Alice robot at any moment.

Since the outputs of the GPG constitute an appropriate control method for the Alice robot, the desired joint angles should be sent to the robot (see Figure 8). To this aim, the GPG first discretizes the whole displacement to several points. Then, the joint angles are calculated for all points and sent to the robot successively. This means that after positioning the servos in the desired angles at a point, the joint angles of the next point will be sent to the robot.

According to Figure 8, planning a path between the initial and the desired positions is the first step to generate the points. In this study, the shortest path to the desired position is selected (see Figure 9(a)). It consists of two rotations in the starting and ending positions (α_1 and α_2), and a straight-line motion between them (\vec{L}). For this purpose, according to Figure 9(b), displacement vectors of the right and left shoes (\vec{g}_1 and \vec{g}_2) are derived:

$$\begin{aligned} \vec{g}_1 &= x_{g_1} \hat{I} + y_{g_1} \hat{J} = \vec{u}_1 + \vec{g} + \vec{s}_1 = \frac{l_4}{2} \hat{J} \\ &+ \left(X\hat{I} + Y\hat{J} \right) + \frac{l_4}{2} \sin(\alpha) \hat{I} - \frac{l_4}{2} \cos(\alpha) \hat{J}, \end{aligned} \quad (29)$$

$$\begin{aligned} \vec{g}_2 &= x_{g_2} \hat{I} + y_{g_2} \hat{J} = \vec{u}_2 + \vec{g} + \vec{s}_2 = -\frac{l_4}{2} \hat{J} \\ &+ \left(X\hat{I} + Y\hat{J} \right) - \frac{l_4}{2} \sin(\alpha) \hat{I} + \frac{l_4}{2} \cos(\alpha) \hat{J}. \end{aligned} \quad (30)$$

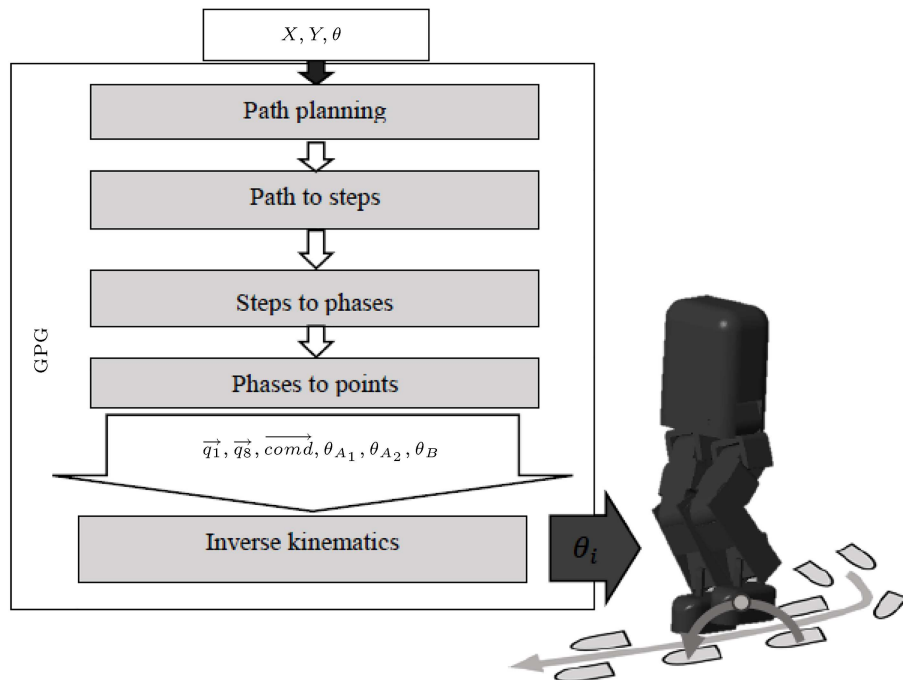


Figure 8. The Gait Pattern Generator (GPG).

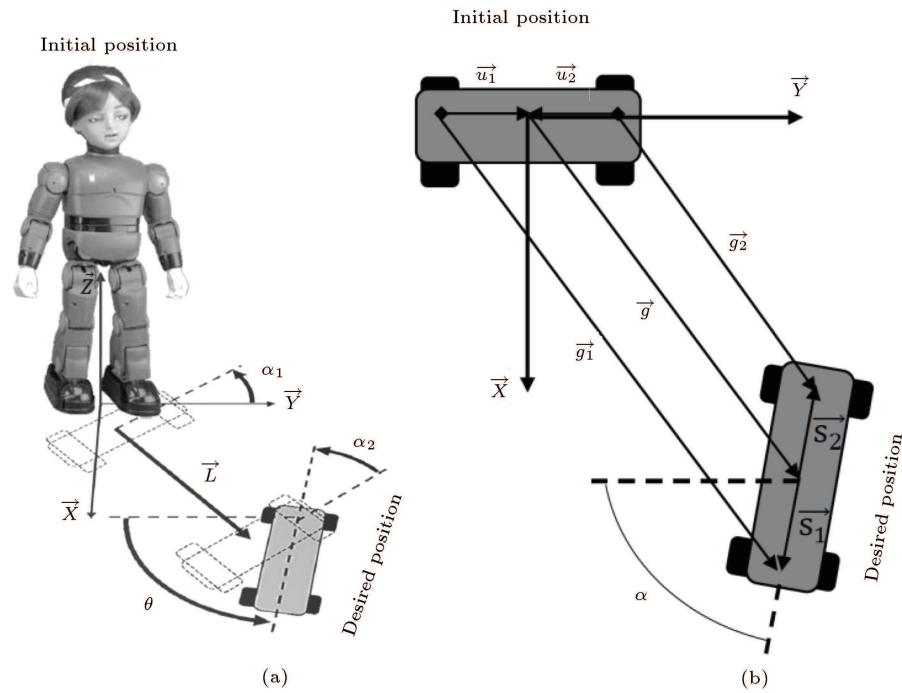


Figure 9. (a) Path planning for Alice robot. (b) The shoes displacement vectors (\vec{g}_1 and \vec{g}_2).

Then, \vec{L} , α_1 , and α_2 are calculated based on Y as follows:

If $Y > 0$:

$$\vec{L} = \vec{g}_2, \quad \alpha_1 = \tan^{-1} \left(\frac{y_{g2}}{x_{g2}} \right), \quad \alpha_2 = \alpha - \alpha_1, \quad (31)$$

If $Y < 0$:

$$\vec{L} = \vec{g}_1, \quad \alpha_1 = \tan^{-1} \left(\frac{y_{g1}}{x_{g1}} \right), \quad \alpha_2 = \alpha - \alpha_1. \quad (32)$$

Because traveling along a path is not a continuous procedure for a bipedal robot, it is difficult to provide a single algorithm for discretizing the path to these positions. Bipedal robots move toward a destination by taking several steps with a maximum length limit. Each step includes several phases, such as the swing of each leg, switching a leg to swing mode, rotation of the robot body, and rotation of each leg. As can be seen in Figure 8, the path is discretized to points in three stages. First, locations of feet placements on the ground are determined. This is done in the “path to steps” block so that the robot reaches the final position with minimum steps. Then, some phases are combined to create the steps in the “steps to phases” block. Finally, each phase can be defined by a variation of one or more input parameters of the IK process (e.g., coordinates of swing foot) as a continuous function of time. This definition is achieved in the “phases to points” block to determine the points as robot positions

at specific time intervals. For this purpose, polynomial functions are utilized. For example, in the straight-line motion, coordinates of the swing foot (\vec{q}_1 or \vec{q}_8) are defined as follows:

$$x = \frac{6L}{T^5}t^5 - \frac{15L}{T^4}t^4 + \frac{10L}{T^3}t^3, \quad (33)$$

$$z = -\frac{64h}{T^6}t^6 + \frac{192h}{T^5}t^5 - \frac{192h}{T^4}t^4 + \frac{64L}{T^3}t^3. \quad (34)$$

i is 1 for swing of the right foot and 8 for swing of the left foot. l , h , and T are respectively the length, height, and duration of a step. For smooth take-off and landing, coefficients are calculated in such a way that the velocity and acceleration of the swing foot are equal to zero at the beginning and end of the step.

Also, the coordinates of the GCoM ($[X_{com} \ Y_{com} \ 0]^T$) that lead to static stability of the robot during the movement are defined as depicted in Figure 10(a). GCoM is moved with a polynomial function of time to switch a leg to swing mode (Figure 10(b)). Similarly, defining θ_{A1} , θ_{A2} , and θ_B results in the starting and ending rotations.

In the next stage, the robot joint angles are calculated for all points. To this end, the inverse kinematics mentioned in the last section is utilized. In this process, each point is used as the initial condition to calculate the next point. The calculated angles are saved in a matrix; then, they are sent to the robot successively. The process mentioned in this section is evaluated in the next sections.

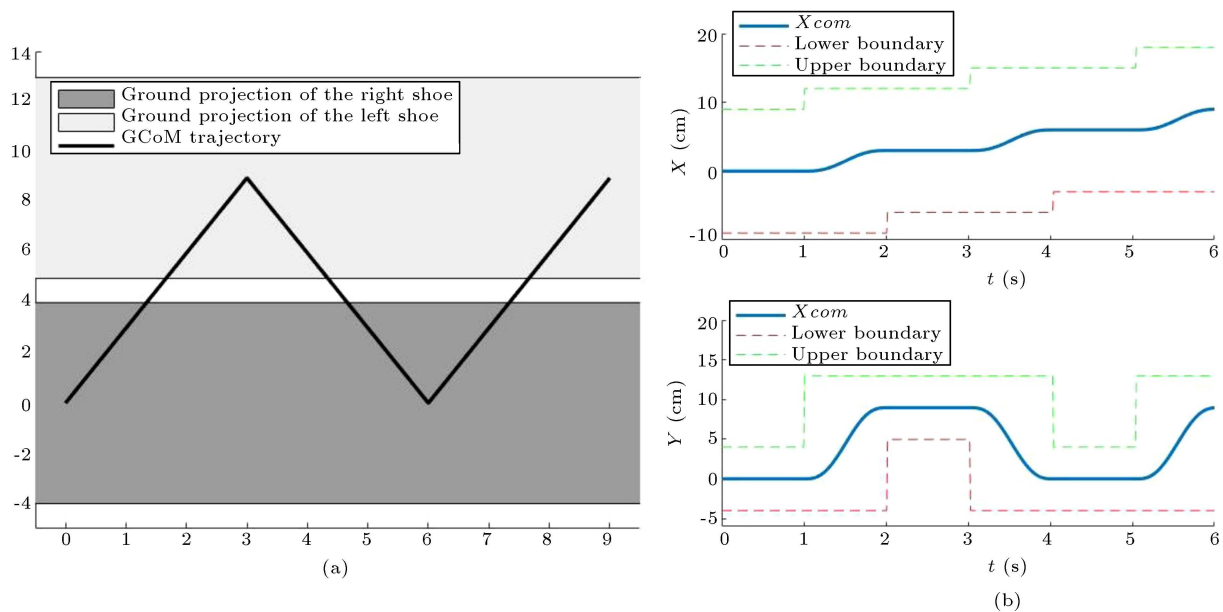


Figure 10. (a) GCoM trajectory in X-Y plane. (b) GCoM displacement with a polynomial function of time.

5. Results and discussion

5.1. Simulation results

The proposed GPG was run in the simulation environment for evaluation and correction before implementing it on the robot. For this purpose, the GPG was coded in MATLAB. An example of the joint trajectories generated by the GPG is depicted in Figure 11. According to Figure 12, all the robot movements were correctly performed and the robot reached the desired position. Consequently, correctness of the algorithms of path planning, point generating, and inverse kinematics was proven. Also, robot stability during the movement indicates that the stability condition was applied properly.

Then, the model was run in different average gait velocities (V_{ave}) to determine the maximum V_{ave} at

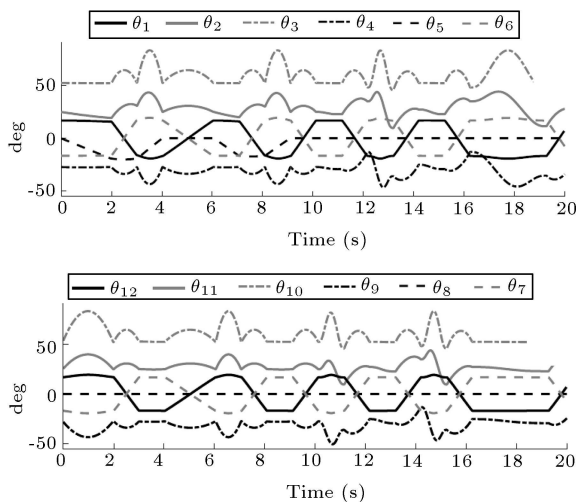


Figure 11. Joint trajectories generated by the GPG.

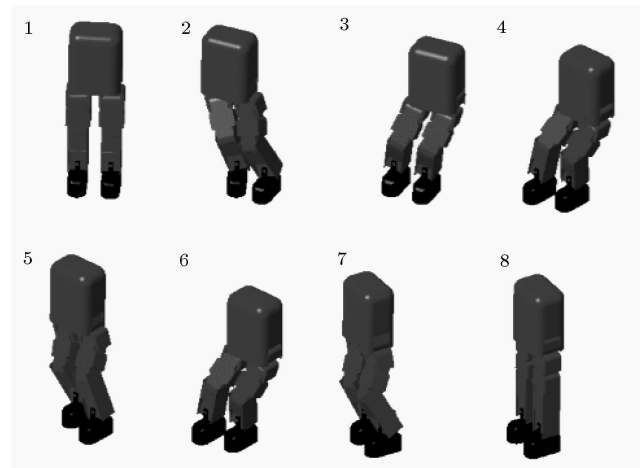


Figure 12. Snapshots of the robot walking in the simulation environment.

which the robot remained stable. To this end, the Centre of Pressure (CoP) of the ground contact forces was determined during the movements. Also, the stability margin was defined as the smallest distance of the CoP from the edges of the stability area. The stability margin of the robot for walking with three different values of V_{ave} is depicted in Figure 13. According to the evaluation, the stability margin reaches zero and the robot becomes unstable for V_{ave} more than 1 cm/s. Indeed, walking of the robot was very slow by using the static stability condition.

5.2. Implementation of the GPG on the robot

Finally, the GPG was coded in Java in NetBeans IDE and implemented on the robot. The robot movements indicated appropriate performance of the

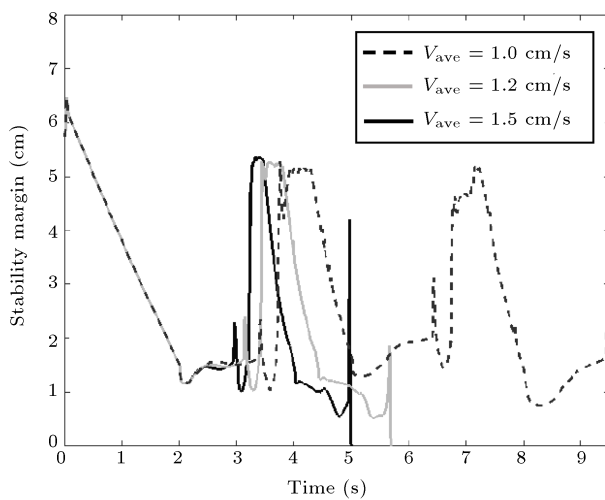


Figure 13. Robot stability margins for walking with AGV of 1 cm/s, 1.2 cm/s, and 1.5 cm/s.

IK and acceptable accuracy of the robot kinematics modelling (see Figure 14). Unfortunately, the robot frequently fell down while walking. Factors like the robot dynamics modelling and performance of the servos were investigated as possible reasons for the falls of the robot.

Weighing the robot indicated 55% error in the mass of the model. It seemed that the previously reported masses in [29,30], which were used in Section 3.1, were wrong or incomplete. Therefore, the model was modified by trial and error. The mass of the components of the model was changed to improve stability of the robot. Although falls of the robot decreased, the stability problem was not thoroughly solved.

Inappropriate contact between the shoes and the ground at the end of the swing phases was another reason for instability of the robot. The investigation revealed that the shoes were not in the horizontal position while the robot was walking because of improper joint position tracking. Figure 15 shows the position tracking error of the 12 lower-body joints. It seems that

the torques generated in the servos were not proper for tracking the desired positions. Thus, default PID coefficients should be tuned to achieve a better tracking performance. However, there was no way to resolve this problem, because it was not possible to access the controllers of the servos.

6. Conclusion

The purpose of this research was to generate a Gait pattern for the Alice humanoid robot. This pattern was specified according to the characteristics of the robot. First, a kinematic and dynamic model of the robot was developed. Then, inverse kinematics of the lower-body was solved using the Newton-Raphson numerical method. To this end, the kinematic equations of the robot were derived with the Cartesian coordination of joints and the static method was used as the stability condition. Finally, the GPG was developed for a step by step displacement of the robot towards a desired destination in a two-dimensional space.

One of the strengths of this method was using a multi-mass model that led to fewer errors than the one-, two-, and three-mass models [15–18]. This method is more appropriate for robots in which the mass of the leg parts is not negligible in comparison with the body mass. In addition, the static method and the kinematic equations that were used in this paper resulted in a significant increase in the solution rate of the calculations compared to the algorithm suggested in [19]. This was true even for a model with more complex kinematics (a three-dimensional model with non-spherical hip joints), although the produced walking was too slow. However, this problem did not have a serious conflict with the goals expected for the robot [25–28].

Simulations showed good performance of the proposed displacement algorithms, inverse kinematics, and stability condition. Finally, the GPG was implemented on the Alice robot. Acceptable movements of the robot indicated suitability of the kinematic modelling.

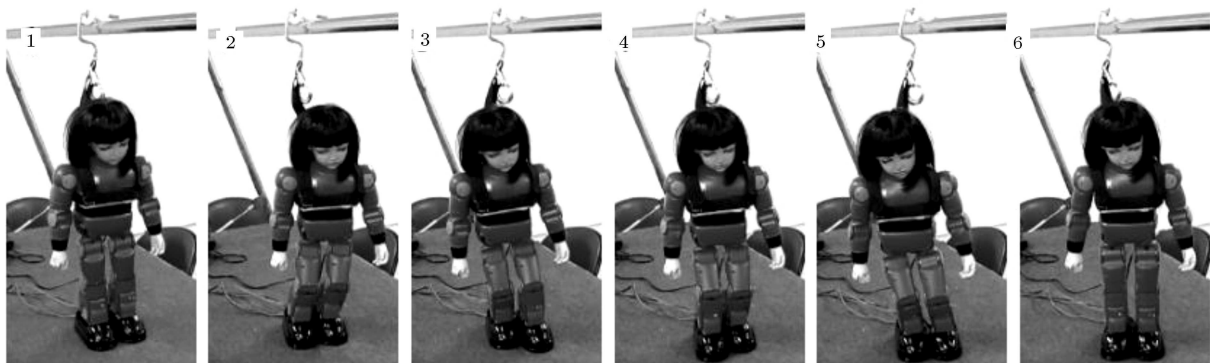


Figure 14. Snapshots of the robot walking during the experiment.

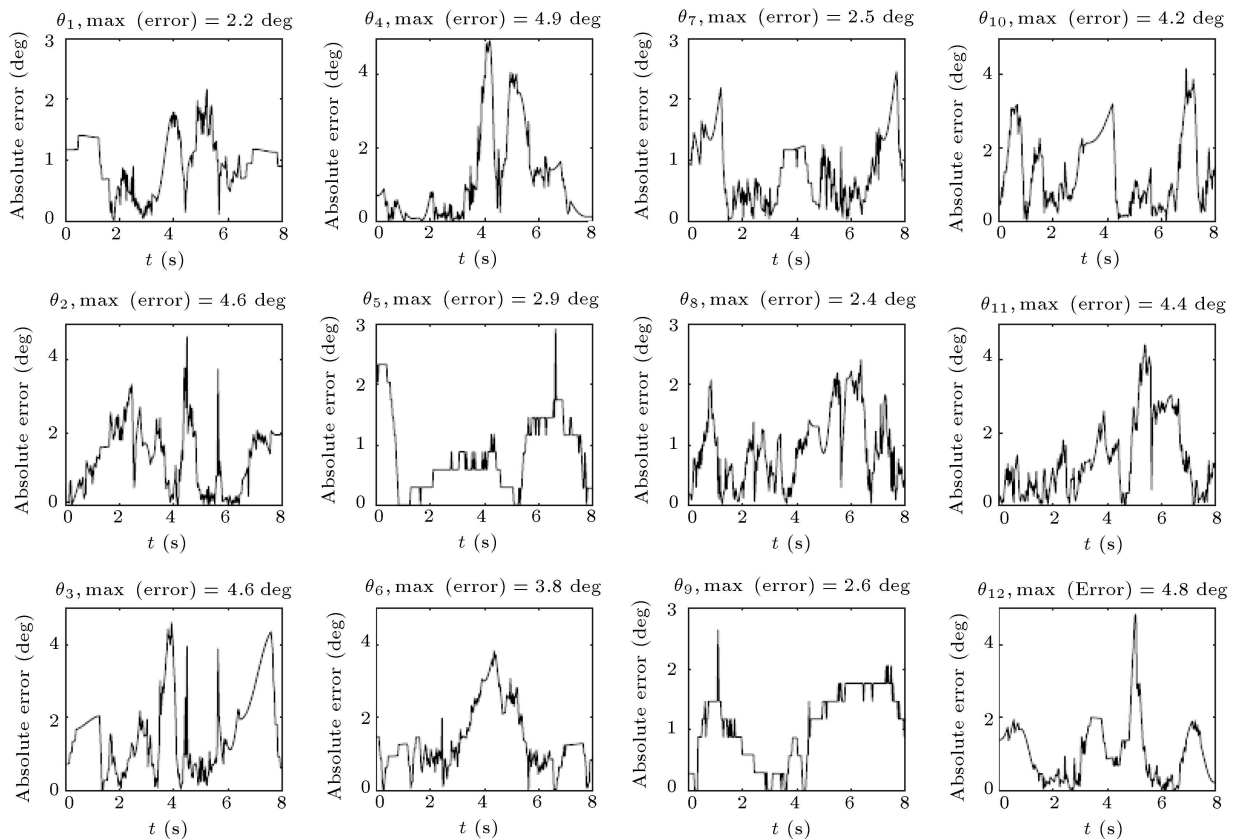


Figure 15. Position tracking error of the 12 lower-body joints.

However, the stability of the robot was not acceptable. Factors disrupting stability of the robot were identified and the walking pattern was reformed as much as possible. From this research, one can also conclude that the Alice humanoid robot manufactured by RoboKind Company [24,30] is not recommendable for education and research due to its poor design, performance, and customer support.

Acknowledgement

This research was supported in part by a grant from the National Science Foundation of Iran-INSF (<http://en.insf.org/>) under contract #FY1395-1397 received by the corresponding author, Ali Meghdari.

References

1. Taheri, A., Meghdari, A., Alemi, M., et al. "Teaching music to children with autism: a social robotics challenge", *Scientia Iranica*, **26**(1), pp. 40-58 (2019).
2. Alemi, M., Meghdari, A., and Ghazisaedy, M. "The impact of social robotics on L2 learners' anxiety and attitude in English vocabulary acquisition", *International Journal of Social Robotics*, **7**(4), pp. 523-535 (2015).
3. Alemi, M., Ghanbarzadeh, A., Meghdari, A., et al. "Clinical application of a humanoid robot in pediatric cancer interventions", *International Journal of Social Robotics*, **8**(5), pp. 743-759 (2016).
4. Meghdari, A., Alemi, M., Zakipour, M., et al. "Design and realization of a sign language educational humanoid robot", *Journal of Intelligent & Robotic Systems*, pp. 1-15 (2018).
5. Meghdari, A., Shariati, A., Alemi, M., et al. "Design performance characteristics of a social robot companion "Arash" for pediatric hospitals", *International Journal of Humanoid Robotics*, **15**(05), p. 1850019 (2018).
6. Meghdari, A., Shariati, A., Alemi, M., et al. "Arash: A social robot buddy to support children with cancer in a hospital environment", *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, **232**(6), pp. 605-618 (2018).
7. Al-Shuka, H.F., Allmendinger, F., Corves, B., et al. "Modeling, stability and walking pattern generators of biped robots: a review", *Robotica*, **32**(6), pp.907-934 (2014).
8. Jianghai, Z., Xiaodong, Y., Feng, H., et al. "Walking pattern generation of biped robot using trajectory planning of gravity center", In *2014 IEEE International Conference on Mechatronics and Automation*, pp. 890-895 (2014).
9. Vukobratovic, M., Frank, A.A., and Juricic, D. "On the stability of biped locomotion", *IEEE Transactions*

- on *Biomedical Engineering*, **BME-17**(1), pp. 25-36 (1970).
10. Goswami, A. "Postural stability of biped robots and the foot-rotation indicator (FRI) point", *The International Journal of Robotics Research*, **18**(6), pp. 523-533 (1999).
 11. Goswami, A. and Kallem, V. "Rate of change of angular momentum and balance maintenance of biped robots", In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, **4**, pp. 3785-3790 (2004).
 12. Garcia, E., Estremera, J., and De Santos, P.G. "A comparative study of stability margins for walking machines", *Robotica*, **20**(6), pp. 595-606 (2002).
 13. De Santos, P.G., Jimenez, M.A., and Armada, M.A. "Dynamic effects in statically stable walking machines", *Journal of Intelligent and Robotic Systems*, **23**(1), pp. 71-85 (1998).
 14. Popovic, M.B., Goswami, A., and Herr, H. "Ground reference points in legged locomotion: Definitions, biological trajectories and control implications", *The International Journal of Robotics Research*, **24**(12), pp. 1013-1032 (2005).
 15. Kajita, S., Kanehiro, F., Kaneko, K., et al. "The 3D linear inverted pendulum mode: A simple modeling for a biped walking pattern generation", In *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the Next Millennium (Cat. No. 01CH37180)*, **1**, pp. 239-246 (2001).
 16. Park, J.H. and Kim, K.D. "May. Biped robot walking using gravity-compensated inverted pendulum mode and computed torque control", In *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No. 98CH36146)*, **4**, pp. 3528-3533 (1998).
 17. Shimmyo, S., Sato, T., and Ohnishi, K. "Biped walking pattern generation by using preview control based on three-mass model", *IEEE Transactions on Industrial Electronics*, **60**(11), pp. 5137-5147 (2013).
 18. Meghdari, A., Sohrabpour, S., Naderi, D., et al. "A novel method of gait synthesis for bipedal fast locomotion", *Journal of Intelligent and Robotic Systems*, **53**(2), pp. 101-118 (2008).
 19. Albert, A. and Gerth, W. "Analytic path planning algorithms for bipedal robots without a trunk", *Journal of Intelligent and Robotic Systems*, **36**(2), pp. 109-127 (2003).
 20. Sakagami, Y., Watanabe, R., Aoyama, C., et al. "The intelligent ASIMO: System overview and integration", In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, **3**, pp. 2478-2483 (2002).
 21. Gouaillier, D., Hugel, V., Blazevic, P., et al. "Mechatronic design of NAO humanoid", In *2009 IEEE International Conference on Robotics and Automation*, pp. 769-774 (2009).
 22. Ali, M.A., Park, H.A., and Lee, C.G. "Closed-form inverse kinematic joint solution for humanoid robots", In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 704-709 (2010).
 23. Limón, R.C., Ibarra, Z.J.M., and Armada, R.M.Á. "Inverse kinematics of a humanoid robot with non-spherical hip: A hybrid algorithm approach", *International Journal of Advanced Robotic Systems*, **10**(4), p. 213 (2013).
 24. <https://lego-discounter.com/hanson-robokind-r50-humanoid-alice>
 25. Taheri, A.R., Alemi, M., Meghdari, A., et al. "Social robots as assistants for autism therapy in Iran: Research in progress", In *2014 Second RSI/ISM International Conference on Robotics and Mechatronics (ICRoM)*, pp. 760-766 (2014).
 26. Taheri, A., Meghdari, A., Alemi, M., et al. "Clinical interventions of social humanoid robots in the treatment of a pair of high-and low-functioning autistic Iranian twins", *Scientia Iranica, Transactions B, Mechanical Engineering*, **25**(3), pp. 1197-1214 (2018).
 27. Taheri, A.R., Alemi, M., Meghdari, A., et al. "Clinical application of humanoid robots in playing imitation games for autistic children in Iran", *Procedia-Social and Behavioral Sciences*, **176**, pp. 898-906 (2015).
 28. Taheri, A., Alemi, M., Meghdari, A., et al. "Impact of humanoid social robots on treatment of a pair of Iranian autistic twins", In *Social Robotics: 7th Int. Conf.*, ICSR 2015, Paris, France, pp. 623-632 (2015).
 29. Abedi, M. "On the design of a gait pattern for the Alice "Mina" social robot", M.Sc. Thesis, Sharif University of Technology, Tehran, Iran (January, 2016).
 30. <https://www.robokind.com/>
 31. Spong, M.W., Hutchinson, S., and Vidyasagar, M. "Forward and inverse kinematics", In *Robot Modeling and Control*, First Edn., pp. 65-103 (2006).

Biographies

Ali Meghdari received his PhD in Mechanical Engineering from the University of New Mexico (UNM) in 1987. He then joined the robotics group of Los Alamos National Laboratory (LANL) as a Post-Doctoral research collaborator. In 1988, he accepted the position of Assistant Professor of Mechanical Engineering at Sharif University of Technology (SUT) in Tehran. During 1993-94, he was a visiting researcher at the AHMCT center of the University of California-Davis and during 1999-2000, he served the IBDMS research center at the Colorado School of Mines, and the Rocky Mountain Musculoskeletal Research Laboratory (RMMRL) as a visiting research professor. Professor Meghdari has performed extensive research in the areas of robotics, social and cognitive robotics, mechatronics, bio-robotics, nano-manipulation, and modelling of biomechanical systems. He has been the recipient

of various scholarships and awards, the latest being: the 2012 Allameh Tabatabaei Distinguished Professorship Award by the National Elites Foundation of Iran (BMN); the 2001 Mechanical Engineering Distinguished Professorship Award from the Ministry of Science, Research & Technology (MSRT) in Iran; and the 1997 ISESCO Award in Technology from Morocco. He was nominated and elected as a fellow of the American Society of Mechanical Engineers (ASME) in 2001. He is currently director of the Centre of Excellence in Design, Robotics and Automation (CEDRA). Since 2005, he has been an affiliate member of the Iranian Academy of Sciences (IAS).

Saeed Behzadipour received his BSc and MSc in Mechanical Engineering from Sharif University of

Technology, Iran, in 1998 and 2000, respectively. He received his PhD in Mechanical Engineering from University of Waterloo, Canada, in 2005. He was an Assistant Professor of Mechanical Engineering at the University of Alberta, Canada, from 2005 to 2011. He is currently an Associate Professor of Mechanical Engineering at Sharif University of Technology. His research interests include medical robotics and rehabilitation technologies.

Majid Abedi received his MSc in Mechanical Engineering from Sharif University of Technology, Tehran, Iran, in 2016. He is currently a PhD candidate in the School of Mechanical Engineering at Sharif University of Technology. His main research interests are robotics and rehabilitation.