Near-optimal gait generations of a two-legged robot on rough terrains using soft computing

Pandu Ranga Vundavilli, Dilip Kumar Pratihar*

Department of Mechanical Engineering, Indian Institute of Technology, Kharagpur 721 302, India

Abstract

A two-legged robot will have to generate its near-optimal gaits after ensuring maximum dynamic balance margin and minimum power consumption, while moving on the rough terrains containing some staircases and sloping surfaces. Moreover, the changes of joint torques should lie below a pre-specified small value to ensure its smooth walking. The balance of the robot and its power consumption are also dependent on hip trajectory and position of the masses on various limbs. Both neural network- and fuzzy logic-based gait planners have been developed for the same, the training of which are provided using a genetic algorithm off-line. Once optimized, the planners are found to generate optimal gaits of the two-legged robot successfully for the test cases.

1. Introduction

A two-legged robot should consume minimum energy after maintaining its smooth walking and dynamic balance. Dynamic balance margin (DBM) is calculated using the concept of zero moment point (ZMP) [1]. The robot is said to be dynamically balanced, if the ZMP lies inside its support polygon. The issues related to modeling and control of biped robots were studied by various investigators. Some of those attempts are discussed below.

A semi-inverse method was proposed by Juricic and Vukobratovic [2] to determine trunk motion of a biped robot. Takanishi et al. [3] introduced a control method, which stabilized dynamic walking using the trunk and waist motions. Shih and Chiou [4] could develop a statically stable biped robot walking on sloping surfaces and its performance was experimentally validated. Shih [5] designed and built a 7-DOF biped robot with variable length legs and translatable balance weights, and carried out experiments to generate an efficient walking gait for ascending and descending the stairs. Dynamically balanced walking gait for a humanoid robot to climb sloping surface was studied by Zhou et al. [6]. Its gait synthesis was formulated as a constrained optimization problem.

Although the above attempts laid down the foundation to solve the problem related to walking control of the biped robots, optimization of some parameters on various terrain conditions was carried out using some gradient-based methods, whose solutions may get stuck at local minima. Mousavi and Bagheri [7] carried out mathematical simulations of a seven-link biped robot moving on various surfaces including stair-case for the fixed and moving ZMP considerations. Soft computing-based approaches (that is, neural network (NN), Fuzzy logic (FL), genetic algorithm (GA) and their various combinations) are proved to be the most successful tools for handling complex, uncertain, imprecise, real-world problems [8]. Attempts were also made to solve optimization problems of biped robots related to mechanical design, controller design and gait generation using the principle of soft computing.

Capi et al. proposed a gait synthesis method using a GA, which could generate joint angle trajectories based on minimum consumed energy and change in joint torques. This method was verified through computer simulations [9] and on real robot [10]. The minimum consumed energy gaits were used to teach a radial basis function neural network (RBFNN). After a supervised learning of the network, it was used to control the robot in real-time. Juang [11] developed intelligent locomotion control on various sloping surfaces. Three back-propagation neural networks (BPNs) were utilized, out of those, one was the controller and the remaining two were emulator and slope information indicator. The controller was pre-trained on a horizontal surface, and the emulator was used to identify robot dynamics. The compensation signal for various slope angles was provided by the slope information indicator NN. Goncalves and Zampieri [12] developed a recurrent neural network to determine the trunk motion of a biped walking machine based on ZMP criterion.
The lower limbs’ movement and planned ZMP were considered as inputs to the network, whereas the estimated ZMP was the output utilized for determining the coordinates of the trunk mass. Park and Choi [13] employed a real-coded GA to search for optimal locomotion pattern and locations of mass centers of the links. The coefficients of the polynomials and mass centers of the limbs were used as the design variables.

Park [14] proposed a method to reduce the trunk motion of a biped robot with the help of fuzzy logic. His algorithm was tested on a 7-DOF biped robot through computer simulations and the stability of locomotion was found to increase with the reduced motion on the trunk. A method for stable gait generation of a biped robot was developed by Jha et al. [15] utilizing a genetic-fuzzy system, in which a GA was used to optimize knowledge base (KB) of a fuzzy logic controller (FLC), offline. Vundavilli et al. [16] developed an appropriate planner for the dynamically balanced biped robot was developed by Jha et al. [15] utilizing a genetic-
stability of locomotion was found to increase with the reduced

biped robot with the help of fuzzy logic. His algorithm was tested
on a 7-DOF biped robot through computer simulations and the
indicated in Section 7.

In this paper, an attempt has been made to optimize the
positions of mass centers of the links, determine suitable hip
trajectory (assumed to be a cubic polynomial) and generate the
gaits of a biped robot negotiating various terrains (that is,
staircase, sloping surface). The GA-tuned NN- and FL-based gait
planners (obtained off-line) are utilized to generate its optimal
gaits. Moreover, the present work deals with locomotion of the
biped robot over various terrains, namely sloping surface and
staircase, whereas the above study carried out by the same
authors previously concentrated on one terrain only, say either
staircase or sloping surface.

The remaining part of the paper is organized as follows:
Section 2 deals with kinematic and dynamic models of the biped
robot. Mathematical formulation of the problem has been
included in Section 3. The proposed algorithm to solve the
said problem has been explained in Section 4. Results are
presented and discussed in Section 5. Some concluding remarks
are made in Section 6. The scope for future study has been
indicated in Section 7.

2. Kinematic and dynamic models of the biped robot

A 7-DOF biped robot will have to plan its dynamically balanced
gait after ensuring minimum power consumption and optimizing
positions of mass centers on the limbs and hip trajectory, while
moving through various terrains. Moreover, the rate of change of
torque for each joint must be below a pre-defined value to ensure
its smooth walking. The developed kinematic and dynamic
models of the said robot are explained below.

2.1. Ascending the staircase and sloping surface

The schematic views of a biped robot moving up through a
staircase and along a sloping surface are shown in Fig. 1. The
robot consists of a trunk, two upper legs, two lower legs and two
ankles. The joints of the robot are considered to be rotary in
nature. The masses of the limbs are assumed to be lumped at
some convenient points (to be decided by an optimizer) on the

\[
\begin{align*}
\text{ZMP} & = \text{FT} - \text{F} + \text{FN} \\
\text{ZMP} & = \frac{\text{FT}}{2} \\
\text{Supporting foot} & = \text{ZMP + ZMP} \\
\text{Foot trajectory} & = \text{ZMP + ZMP} \\
\text{ZMP} & = \frac{\text{FT}}{2} \\
\text{Foot trajectory} & = \text{ZMP + ZMP} \\
\end{align*}
\]

Fig. 1. Schematic views of a two-legged robot (7-DOF) ascending a (a) staircase, (b) sloping surface.
where \( f_s \) represents the length of the foot, \( s_w \) and \( s_h \) indicate the width and height of the staircase, respectively. Similarly, the boundary conditions for ascending the sloping surface can be expressed like the following (refer to Fig. 1(b)):

At \( x = x_1 \cos x, \ z = x_1 \sin x \),

\[
x = x_3 \cos x, \quad z = x_3 \sin x,
\]

\[
x = (x_1 \cos x + x_2 \cos z)/2, \quad z = x_2 \sin z + \frac{f_s}{2},
\]

\[
x = (x_1 \cos x + x_2 \cos z)/2, \quad z = x_3 \sin z + \frac{f_s}{2},
\]

where \( x \) indicates the angle of the slope. The maximum velocity and duration of each time step of the swing leg are set equal to 0.056 m/s and 5.0 s, respectively. Each time step consists of acceleration and deceleration parts for 1.0 s each and constant velocity of the swing leg for 3.0 s. The trajectory of the hip joint also is assumed to follow a cubic polynomial as follows:

\[
z = c_0 + c_1 x + c_2 x^2 + c_3 x^3.
\]

Among four coefficients of the polynomial, two (that is, \( c_2 \) and \( c_3 \)) are coded in the GA-string, and the remaining two (that is, \( c_1 \) and \( c_4 \)) are calculated from initial and final boundary conditions of the hip joint. The joint angles of lower limbs are calculated utilizing inverse kinematics. The following repeatability conditions are to be followed by the limbs to ensure a cyclic gait:

\[
\theta_{j,\text{initial}} = \theta_{j,\text{final}},
\]

\[
\theta_{k,\text{initial}} = \theta_{k,\text{final}},
\]

where \( (i \text{ and } j) \) take the values of \( 2 \) and \( 6 \) and \( (3 \text{ and } 5) \), and \( k = 1, 4 \).

Dynamic balance of the two-legged robot depends on the effects of inertia and acceleration of the masses. These effects are largely dependent on the distances of masses on the limbs from the respective joints (that is, \( r_1 \) through \( r_7 \)). Here, the distances of masses from the joints have been optimized using an GA. Once the placements of the masses are obtained, the distance of ZMP from the ankle joint of the supporting leg (measured in the direction of motion) can be determined as follows:

\[
x_{ZMP} = \frac{\sum_{i=1}^{7} (I_i \omega_i + m_i x_i (\ddot{z}_i - g) - m_i \dot{x}_i \dot{z}_i)}{\sum_{i=1}^{7} m_i (\ddot{z}_i - g)},
\]

where \( I_i \) represents the moment of inertia of \( i \)-th link (kg m²), \( \omega_i \) denotes the angular acceleration of link \( i \) (rad/s²), \( m_i \) denotes the mass of \( i \)-th link (kg), \( (x_i, y_i, z_i) \) is the coordinate of \( i \)-th lumped mass, \( g \) indicates the acceleration due to gravity (m/s²), \( \ddot{z}_i \) is the acceleration of link \( i \) in \( z \)-direction (m/s²), \( \dot{x}_i \) is the acceleration of link \( i \) in \( x \)-direction (m/s²). The robot is said to be dynamically balanced, only when the ZMP lies inside the foot support polygon. Otherwise, the hip joint, knee joint and ankle joint forces of the supporting leg must be updated to bring the ZMP back to a safe zone. The DBM is defined as the distance between the edge of foot supporting polygon and the point, where the ZMP acts, as follows:

\[
x_{DBM} = \left( \frac{L_f}{2} \cos x - |x_{ZMP}| \right),
\]

where \( L_f \) is the length of the supporting foot and \( x_{ZMP} \) is the distance of ZMP from the ankle joint of the supporting foot. The angle \( x \) is to be set equal to \( 0 \)° for the staircase. The dynamics of the bipedal robot is solved to obtain joint torques \( \tau_j \) utilizing the Lagrange–Euler formulation. The joint angles are assumed to follow a fifth-order polynomial as follows:

\[
q_i(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5,
\]

where \( a_0, a_1, a_2, a_3, a_4 \) and \( a_5 \) are the coefficients, whose values are determined using different values of \( q_i \) at different intervals of time in a cycle, and \( i = 1, 2, \ldots, n \) joints. The amount of power consumption \( P \) is then determined utilizing the following expression:

\[
P = \frac{1}{T} \sum_{t=0}^{T} |\tau_i q_i| dt,
\]

where \( T \) is the time of travel.

2.2. Descending the staircase and sloping surface

A two-legged robot moving down the staircase and sloping surface along with the trajectory to be followed by the swing foot is shown in Fig. 2. The problems related to descending the staircase and sloping surface have been modeled in the similar way of ascending the same. However, some necessary changes are to be incorporated in the boundary conditions for the descending problems. The boundary conditions for staircase descending are given below (refer to Fig. 2(a)).

At \( x = 2s_w + x_1 - x_3, \ z = 2s_h \)

\[
x = 2s_w - x_3 + \frac{f_s}{2}, \quad z = 2s_h + \frac{f_s}{2},
\]

\[
x = s_w - x_3 + \frac{f_s}{2}, \quad z = s_h + \frac{f_s}{2},
\]

\[
x = 0, \quad z = 0.
\]

Similarly, the boundary conditions for descending the sloping surface are shown below (refer to Fig. 2(b)).

At \( x = x_1 \cos x, \ z = x_1 \sin x \)

\[
x = (x_1 \cos x + x_2 \cos z)/2, \quad z = x_1 \sin x + \frac{f_s}{2},
\]

\[
x = (x_2 \cos x + x_3 \cos z)/2, \quad z = x_2 \sin x + \frac{f_s}{2},
\]

\[
x = x_3 \cos z, \quad z = x_3 \sin z.
\]

In this case, the ZMP can be calculated using Eq. (3), after replacing \( g \) by \( -g \).

3. Mathematical formulation of the problem

This problem may be posed as an optimization problem as stated below:

Maximize \( \left( \frac{L_f}{2} \cos x - |x_{ZMP}| \right) + \frac{1}{p} \)

subject to \( \Delta \tau_{ij} \leq \Delta \tau_{\text{specified}} \),

where \( i \) and \( j \) represent the number of joints and time instants, respectively; \( \Delta \tau \) denotes the change in joint torque value.

4. Proposed soft computing-based approaches

Both genetic-neural and genetic-fuzzy systems have been developed for the said purpose as explained below.
4.1. Genetic-neural system (GA-NN)

In GA-NN algorithm, the coefficients of hip trajectory, positions of the lumped masses and connecting weights of the fully connected feed-forward NNs are optimized offline, utilizing a GA.

4.1.1. Ascending the staircase and sloping surface

The methodology adopted in the GA-NN system for solving the problems related to ascending the staircase and sloping surface is the same, except the selection of inputs for the NN. The inputs are the positions of ankle joints from the end of corresponding stair in case of staircase ascending problem, whereas in case of negotiating the sloping surface, the positions of ankle joints measured along the sloping surface with respect to global coordinate system are considered as the inputs. The problem has been solved utilizing two GA-trained NNs (refer to Fig. 3). The inputs to the first NN are the positions of feet placements (that is, \( x_1 \) and \( x_2 \)). Two outputs of this NN are the hip height \( h_1 \) and projected distance of hip joint from the swing foot \( l_1 \). The changes in joint

Fig. 2. Schematic views of a two-legged robot (7-DOF) descending a (a) staircase, (b) sloping surface.

Fig. 3. Architecture of the proposed neural networks.
angles of the swing leg (that is, $\delta h_2$ and $\delta h_3$) are given as inputs to the second NN. The NN yields two outputs, such as in angles for the swing foot and trunk of the biped robot, that is, $\delta h_1$ and $\delta h_4$, respectively. The generated gait is verified for its dynamic balance. The torques and power consumption at various joints have then been calculated.

The input and output layers of two NNs are consisting of two neurons each. The GA-string carries information of NN-parameters, such as weight values: $[V, W]$, bias values, coefficients $c_0$ and $c_1$ of the hip trajectory and positions of the lumped masses (that is, $r_1$, $r_2$, $r_3$ and $r_4$). The remaining positions: $r_5$, $r_6$, and $r_7$ are taken to be equal to $r_3$, $r_2$, and $r_1$, respectively, to make the structure of the biped robot symmetric about the central axis. A batch mode of training has been employed to train the NNs for the staircase and sloping surface ascending cases separately, using 216 cases. The fitness $f$ of a GA-string is calculated, as follows:

$$f = \frac{\sum_{i=1}^{S} DBM_i + 1/P_i}{S},$$

where $S$ represents the number of training cases considered. A high penalty equals to $-100$ is added to the fitness value, if either the NNs represented by the GA-string are unable to generate the dynamically balanced gait or the change in torque for a joint exceeds the pre-defined value.

### 4.1.2. Descending the staircase and sloping surface

The problems of descending gait generation have been solved by following a procedure similar to that of ascending gait generation. The input and output layers are consisting of two neurons each in both the modules of NN. The fitness of a GA-string has been calculated using Eq. (8).

### 4.2. Genetic-fuzzy system (GA-FL)

The gait generation problems of the biped robot moving on the staircase and sloping surface have been modeled using Mamdani approach of fuzzy logic controller [18]. The working principle of a combined GA-FL system is similar to that of the GA-NN system, in which two NNs of the latter are replaced by two FL modules. In GA-FL system, KB (that is, database and rule base) of the FL-system is optimized offline, utilizing a GA.

#### 4.2.1. Ascending the staircase and sloping surface

Two modules of FL have been utilized in the combined GA-FL approach. The membership function distributions of the input and output variables for both the FL-based modules used to solve the staircase ascending problems are shown in Figs. 4 and 5. To tackle the problems related to ascending the sloping surface also, the membership function distributions of the variables are assumed to be the similar, except their starting points. In the proposed algorithm, the coefficients of hip trajectory, positions of the lumped masses and KB of the FL-modules are optimized using the GA. As two inputs (that is, $x_1$ and $x_2$) of the first FL are represented using four linguistic terms (refer to Fig. 4), the number of rules turns out to be equal to 16. A particular rule of this FL-module may look like the following: If $x_1$ is $L$ AND $x_2$ is $M$ THEN $h_1$ is $M$ and $l_1$ is $L$. Similarly, the second FL-module also contains 16 rules, as its two inputs (that is, $\delta h_2$ and $\delta h_3$) are represented using four linguistic terms (refer to Fig. 5) each. Two bits of binary number are assigned to represent each linguistic term of the output variables of the first and second modules of FL-system. For example, 00, 01, 10 and 11 are used to represent $L$, $M$, $H$, and $VH$, respectively, for the first FL-module. Similarly, for the second module, the angle variations like $NL$, $NS$, $PL$ and $PS$ are denoted with the help of 00, 01, 10 and 11, respectively. Therefore, four bits are used to represent two outputs for each rule of the first and second modules of the FL-system each. Moreover, there are 14 real variables (that is, $\alpha_1$, $\alpha_2$, $\alpha_{3}$, $\alpha_{4}$, $\alpha_{5}$, $\alpha_{6}$, $\alpha_{7}$, $\alpha_{8}$, $\alpha_{9}$, $c_0$, $c_1$, $r_1$, $r_2$, $r_3$, $r_4$) and 10 bits are assigned to represent each of them. Thus, the GA-string is $80 + 60 + 16 \times 4 + 16 + 16 \times 4 \times 4 = 300$-bits long, which will look as follows:

| 10111011 | 1001 | 0111001 | 110111 | 1101100 | 010 |
| 10111011 | 1001 | 0111001 | 110111 |
| 10111101 | 1001 | 0111001 | 110111 |
| 1101100 | 010 |

A batch mode of training has been adopted as explained above.

#### 4.2.2. Descending the staircase and sloping surface

A GA-FL system has been designed similarly to solve the problems related to descending the staircase and sloping surface.
5. Results and discussion

The following length (m) and mass (kg) parameters of the biped robot have been considered: \( L_1 = 0.06, L_2 = 0.34, L_3 = 0.30, L_4 = 0.60, L_5 = 0.30, L_6 = 0.34, L_7 = 0.06; m_1 = 0.5, m_2 = 2.0, m_3 = 5.0, m_4 = 25.0, m_5 = 5.0, m_6 = 2.0, m_7 = 0.5 \). The angle of the sloping surface has been kept fixed to 10° during the training.

5.1. Problems related to ascending the staircase and sloping surface

Results of the said approaches in obtaining optimal ascending gaits of the biped robot are explained below.

5.1.1. GA-NN approach

The appropriate number of hidden neurons for each NN is determined through a systematic parametric study, in which one parameter has been changed at a time keeping the others unaltered. The optimal number of hidden neurons for the first and second modules of NN are found to be equal to (5 and 3) and (4 and 6) for the problems related to ascending the staircase and sloping surface, respectively. Both the modules of NN are assumed to have linear, tan-sigmoid and log-sigmoid transfer functions for the input, hidden and output layers, respectively. The coefficients of linear, tan-sigmoid and log-sigmoid transfer functions are kept equal to 1.0, 1.0 and 2.0, respectively. A binary-coded GA is run with a uniform crossover of probability equals to 0.5. For the problems related to ascending the staircase, the following parameters are used: \( \alpha_1 = 2.00, \alpha_2 = 4.00, \alpha_3 = 0.02, \alpha_4 = 0.10, \alpha_5 = 0.28, \alpha_6 = 0.13 \) for the problems related to sloping surface. The optimal values of the parameters \( \alpha_0, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6, \alpha_7 \) and \( \alpha_8 \) are seen to be equal to 0.02, 0.02, 0.04, 0.02, 14.96, 14.93, 4.07 and 3.92 units, respectively, for descending the staircase and 0.01, 0.01, 0.03, 0.01, 6.67, 5.56, 2.08 and 3.19 units, respectively, for descending the sloping surface. The optimal values of the parameters \( \alpha_0, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6, \alpha_7 \) and \( \alpha_8 \) are seen to be equal to (3.54, 3.18, 0.02, 0.10, 0.27 and 0.11) and (2.13, 2.07, 0.014, 0.10, 0.28 and 0.32), for the problems related to descending the staircase and sloping surface, respectively.

5.1.2. GA-FL approach

A GA has been used to optimize the KBs of two modules of FL-system. The optimal values of \( \alpha_0, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6, \alpha_7 \) and \( \alpha_8 \) are found to be equal to (1.19, 2.52, 0.02, 0.10, 0.14 and 0.29) and (3.19, 1.91, 0.01, 0.10, 0.19 and 0.16), for the problems related to ascending the staircase and sloping surface, respectively. Moreover, for the above two problems, the optimal values of \( \alpha_0, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6, \alpha_7 \) and \( \alpha_8 \) are seen to be equal to 0.01, 0.01, 0.02, 0.01, 0.07, 0.02, 7.25, 8.11, 2.58, 2.21, respectively. Both the modules of FL-system tackling the staircase ascending problems are seen to contain 15 optimal rules each in their rule bases, whereas the first and second modules of FL used for solving the problems related to descending the sloping surface are found to have 16 and 15 rules, respectively.

5.2. Problems related to descending the staircase and sloping surface

Results of the above approaches for descending gait generation of the biped robot are explained below.

5.2.1. GA-NN approach

The appropriate number of hidden neurons for the first and second modules of NN are found to be equal to (6 and 3) and (3 and 5), for handling the problems related to descending the staircase and sloping surface, respectively. The GA has obtained the optimized values of \( \alpha_0, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6, \alpha_7 \) and \( \alpha_8 \) as (1.00, 4.00, 0.02, 0.10, 0.28 and 0.10) and (1.00, 4.00, 0.01, 0.10, 0.28 and 0.13) for tackling the problems related to descending the staircase and sloping surface, respectively.

5.2.2. GA-FL approach

In this approach, the optimized values of \( \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6, \alpha_7 \) and \( \alpha_8 \) are seen to be equal to 0.02, 0.02, 0.04, 0.02, 14.96, 14.93, 4.07 and 3.92 units, respectively, for descending the staircase and 0.01, 0.01, 0.03, 0.01, 6.67, 5.56, 2.08 and 3.19 units, respectively, for descending the sloping surface. The optimal values of the parameters \( \alpha_0, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6, \alpha_7 \) and \( \alpha_8 \) are seen to be equal to (3.54, 3.18, 0.02, 0.10, 0.27 and 0.11) and (2.13, 2.07, 0.014, 0.10, 0.28 and 0.32), for the problems related to descending the staircase and sloping surface, respectively.

5.3. Simulation results

The aim of the present study is to design suitable NN- and FL-based gait planners of a biped robot moving through rough terrains consisting of staircase and sloping surface. Once the optimal modules are obtained using the GA, those can be used for some test cases generated at random. It has been assumed that a test case will consist of two scenarios, such as either staircase–staircase or staircase-sloping surface or sloping surface-staircase or sloping surface-sloping surface. Fig. 6 shows the flow-chart explaining the way simulation has been performed. Two-bits binary numbers are generated at random to represent a test case. There are four different possibilities with the generated two bits, such as 00, 01, 10 and 11, correspond to staircase–staircase, staircase–sloping surface, sloping surface–staircase, and sloping surface–sloping surface, respectively. It is important to mention that ascending through a particular terrain (either staircase or sloping surface) has been followed by its descending. The results of one particular test scenario, say 01 (that is, staircase ascending and descending followed by slope surface ascending and descending) have been presented below to test the effectiveness of the proposed algorithms.

Fig. 6. Flowchart for simulation of test cases.
5.3.1. GA-NN approach

Let us consider the following inputs of the gait planner generated randomly for a test case: $x_1 = 0.101334\, \text{m}$, $x_2 = 0.058299\, \text{m}$, $x_3 = 0.088504\, \text{m}$. The values of average torque (N m) required at seven joints of the robot for ascending and descending the staircase are found to be equal to (37.11, 15.62, 22.78, 23.16, 5.57, 69.18, 0.07) and (28.21, 13.14, 3.15, 1.53, 3.12, 43.82, 0.09), respectively. The average torque of joint 6 (that is, knee joint of the supporting leg) has turned out to be the maximum. It is so, as it is the supporting joint for the rest of the mechanism, while moving the swing leg from an earlier support phase to the next one.

Let us also consider the following inputs of the gait planner for handling the sloping surface problem: $x_1 = 0.071153\, \text{m}$, $x_2 = 0.208495\, \text{m}$, $x_3 = 0.347058\, \text{m}$. The joint angles have followed the said repeatability conditions. The values of average torque (N m) for the joints 1–7 are found to be equal to (23.62, 5.17, 7.59, 6.36, 4.41, 60.61, 0.31) and (21.10, 11.27, 4.36, 4.78, 6.29, 92.71 and 0.06) for the problems related to ascending and descending the sloping surface, respectively. The torque at the knee joint of the supporting leg has turned out to be the maximum.

The DBM increases initially for the staircase ascending and descending, and sloping surface ascending cases and then decreases towards the end of the cycle. However, the DBM increases continuously throughout the cycle, while descending the sloping surface. It is so, as the ZMP moves from one side of the ankle joint to other side in the direction of motion, for the first three cases and it lies on one side only of the ankle joint and moves towards the ankle joint in the direction of motion, in the later case. It is interesting to observe that ascending the terrains needs more power than the descending does. It may be due to the fact that the robot moves against the gravity in case of ascending the terrain and moves in the direction of gravity, while descending the same.

5.3.2. GA-FL approach

The same set of inputs has been used to test the performance of this approach. The values of average torque (N m) for the joints

![Fig. 7. Simulations using GA-NN approach: staircase—(a) ascending, (b) descending; sloping surface—(c) ascending, (d) descending.](image-url)
1–7 are seen to be equal to (15.97, 5.81, 11.49, 10.30, 17.85, 69.01, 0.17) and (9.01, 5.82, 7.69, 7.91, 4.36, 81.77, 0.13) for the problems related to ascending and descending the staircase, respectively. Similarly, the values of average torque (N m) required at seven joints of the robot for ascending and descending the sloping surface are found to be equal to (19.37, 3.64, 4.59, 6.81, 11.05, 60.89, 0.06) and (5.99, 2.54, 11.87, 10.93, 7.73, 84.09, 0.08), respectively. The torque required at the knee of supporting leg is found to be the maximum. The obtained variations of DBM in a cycle follow the pattern of that yielded by the GA-NN approach. Once again, power consumption during ascending the terrain is seen to be more than that during descending, and the reason behind this fact has been explained earlier. The average DBM for the cycle obtained by the GA-FL approach is seen to be more than that yielded by the GA-NN approach. Moreover, the average power consumption of the robot determined by this approach is found to be less than that obtained by the GA-NN approach. Thus, the GA-FL approach is seen to perform slightly better than the GA-NN approach. It is so, as some problem information has been fed to the database of FL-based approach prior to its GA-based optimization, which is not done in NN-based approach. Simulation results of the biped robot, while negotiating the staircase and sloping surface using the GA-NN and GA-FL approaches are shown in Figs. 7 and 8, respectively. The biped robot is found to successfully tackle the staircase and sloping surface in both the approaches. Moreover, repeatability conditions have been maintained in the generated gait.

5.3.3. Robustness test

Robustness of the developed GA-NN and GA-FL gait planners of a two-legged robot moving on the staircase and sloping surface has been tested by allowing a little variation in the values of input variables (that is, $x_1$, $x_2$ and $x_3$). Results of the robustness test are shown in Table 1. It is interesting to note that ±15% variations in the input variables have led to a maximum of 0.431% and −2.153% variations in the DBM values for the GA-NN and GA-FL approaches, respectively. Thus, GA-NN approach is able to

---

**Fig. 8.** Simulations using GA-FL approach: staircase—(a) ascending, (b) descending; sloping surface—(c) ascending, (d) descending.
generate more robust gaits compared to the GA-FL approach does. It may be due to the differences in their design and training phases, although the training is provided to both of them using the same set of scenarios.

5.3.4. Adaptability test

The adaptability of the developed GA-NN- and GA-FL-based gait planners has been tested by changing the geometry of the terrains (that is, changing the width ($s_w$) and height ($s_h$) of the staircase; angle of the sloping surface; and location ($d_x$) and width ($d_w$) of the ditch). Results of the adaptability test are shown in Table 2. It is important to mention that both NN- and FL-based gait planners have generated the dynamically balanced gaits in all the scenarios. Thus, the developed gait planners are found to be adaptive in nature.

5.3.5. Comparisons with others’ studies

The solutions obtained by Mousavi and Bagheri [7] were not optimal in any sense, as no optimizer was utilized. However, in the present study, optimization of the morphology and gait planner has been carried out with the help of NN- and FL-based approaches, which uses a GA to evolve the optimal systems for navigating through various terrains. Park [14] utilized FL-based approach to solve gait generation problems of a biped robot. However, no attempt was made to optimize the mechanical structure, whereas in the present study, a simultaneous optimization of both gait planner and structure has been tried. In [10,11], the investigators had used a GA to optimize the gait and trajectory of a biped robot. However, in their work, no attempts were made to determine optimal locations of the link-masses, which has been done in the present study. The proposed algorithm has yielded better performance compared to [16,17] also.

6. Concluding remarks

Conclusions have been drawn from the above study as follows:

- Both the NN- and FL-based planners are able to generate dynamically balanced gaits of the biped robot.
- Ascending gait is found to be dynamically more balanced than the descending gait. However, ascending through the terrains requires more power compared to the descending does. The knee torque of the supporting leg has been found to be the maximum.
- GA-FL approach has performed better than the GA-NN approach in terms of DBM and power consumption. Some problem information has been injected to the database of FL-based approach prior to its GA-based optimization, which cannot be done in the NN-based approach. On the other hand, GA-NN approach is found to be more robust compared to the GA-FL approach.
- The CPU time values for solving 20 test cases by the GA-NN and GA-FL approaches are found to be equal to 0.01 and 0.02 s, respectively, on a P-IV machine. It is important to mention that time-delay issue in the actuators should be considered for real-time generation of the gaits, as it is as significant as CPU time of the algorithm. However, the following issues are also to be considered for conducting experiments with the real robot: on-line modeling and recognition of the environment; modeling of foot-ground interaction; movement of the robot in lateral direction; implementation of closed-loop control system; and others.

### Table 1

Results of robustness test for the developed gait planners.

<table>
<thead>
<tr>
<th>Percent changes in inputs</th>
<th>GA-NN approach</th>
<th>GA-FL approach</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Ascend</td>
<td>Descend</td>
</tr>
<tr>
<td>−15</td>
<td>0.088</td>
<td>0.029</td>
</tr>
<tr>
<td>−10</td>
<td>0.192</td>
<td>0.062</td>
</tr>
<tr>
<td>−5</td>
<td>0.192</td>
<td>0.099</td>
</tr>
<tr>
<td>5</td>
<td>−0.064</td>
<td>−0.103</td>
</tr>
<tr>
<td>10</td>
<td>−0.104</td>
<td>−0.243</td>
</tr>
<tr>
<td>15</td>
<td>−0.211</td>
<td>−0.409</td>
</tr>
</tbody>
</table>

### Table 2

Results of adaptability test showing the DBM values for the developed gait planners.

<table>
<thead>
<tr>
<th>Scenario no.</th>
<th>Staircase ($s_w$, $s_h$) (in m)</th>
<th>Sloping surface (in deg)</th>
<th>Ditch ($d_x$, $d_w$) (in m)</th>
<th>Ditch (in m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>NN-based gait planner</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>(0.13, 0.08)</td>
<td>0.02534</td>
<td>0.02403</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>(0.14, 0.09)</td>
<td>0.02539</td>
<td>0.02424</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>(0.15, 0.1)</td>
<td>0.02522</td>
<td>0.02475</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>(0.16, 0.11)</td>
<td>0.02505</td>
<td>0.02428</td>
<td>11</td>
</tr>
<tr>
<td>5</td>
<td>(0.17, 0.12)</td>
<td>0.02489</td>
<td>0.02429</td>
<td>12</td>
</tr>
<tr>
<td>FL-based gait planner</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>(0.13, 0.08)</td>
<td>0.02592</td>
<td>0.02587</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>(0.14, 0.09)</td>
<td>0.02581</td>
<td>0.02577</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>(0.15, 0.1)</td>
<td>0.02571</td>
<td>0.02562</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>(0.16, 0.11)</td>
<td>0.02567</td>
<td>0.02545</td>
<td>11</td>
</tr>
<tr>
<td>5</td>
<td>(0.17, 0.12)</td>
<td>0.02538</td>
<td>0.02518</td>
<td>12</td>
</tr>
</tbody>
</table>
7. **Scope for future work**

In a walking cycle, a biped robot will have both single as well as double support phases. The single support phase consumes a major part of the cycle time and the double support phase is followed during a small fraction of it. The present study concentrates on gait generation problems of the biped robot during its single support phase only, and the same during the double support phase has been kept in the scope for future work. The performances of the developed algorithms have been tested through computer simulations. However, it will be more interesting to test their performances through real experiments. The authors are working on these issues.

**References**


