

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

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

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

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

Numbers displayed above are based on latest data collected.

For more information visit www.intechopen.com



Virtual Reality to Simulate Adaptive Walking in Unstructured Terrains for Multi-Legged Robots

Umar Asif

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/46404>

1. Introduction

Walking robots possess capabilities to perform adaptive locomotion and motion planning over unstructured terrains and therefore are in the focus of high interest to the researchers all over the world. However, true adaptive locomotion in difficult terrains involves various problems such as foot-planning, path-planning, gait dynamic stability and autonomous navigation in the presence of external disturbances [1]. There is a large volume of research work that is dedicated to the development of walking robots that exhibit biologically inspired locomotion. In this context, an insect-sized flying robot [2], an ape-like robot [3], an amphibious snake-like robot ACM-R5 [4], for-legged robots like Big Dog and Little Dog, an ant-inspired robot ANTON [5] and LAURON [6], are some well-known examples of bio-mimetic robots. A related study [7] describes an implementation of a neurobiological leg control system on a hexapod robot for improving the navigation of the robot over uneven terrains.

However, the crucial problem that the general approaches suffer is the challenging task of achieving accurate transition between different gaits while maintaining stable locomotion in the presence of external disturbances [1]. In natural rough terrains, prior knowledge of the terrain information is not always available therefore, this chapter aims to make use of virtual reality technology to inspect adaptive locomotion over unknown irregular terrains for possible improvement in gait mobility and stability in the presence of unexpected disturbances.

In order to evaluate and inspect the dynamic characteristics of an adaptive locomotion over an unstructured terrain, it is difficult and time consuming to conduct real-world experiments often. Therefore, virtual prototyping technology serves a prime role that results in better design optimization and allows inspection and evaluation of the mathematical models specifically those which are complicated or impractical to estimate using real

prototypes in real-world [16-18]. Virtual Reality Technology provides a rapid method of validating several characteristics of a mathematical model using dynamic simulations for the identification of potential errors in early design stages [8]. Possible defects can be identified and removed by modifying the simulation model itself which is less time consuming and practical if to be performed often. Thus, the complete control algorithm can be analyzed using the simulation model [8].

The chapter is structured as follows: second section describes the mathematical modelling of a hexapod robot using kinematics and dynamics equations. In the third section, gait generation is described for planning cyclic wave and ripple gaits. Fourth section is dedicated to the gait analysis over flat and uneven terrains. Finally, the chapter draws conclusion on the significance of using a simulation-based approach for rapid-prototyping of gait generation methods in multi-legged robots.

2. Mathematical modeling and design

The studies conducted in [8] & [9] use SimMechanics software to construct a simulation-based model of a six-legged robot which comprises of several subsystems interconnected to each other as described in Fig. 1. The physical robot is represented using a subsystem known as “plant”. Another subsystem which represents the kinematic engine contains the forward and reverse kinematics of the robot. Other subsystems include a gait generation block and an environment sensing block as shown in Fig. 1. The aforesaid subsystems are further described in the following subsections.

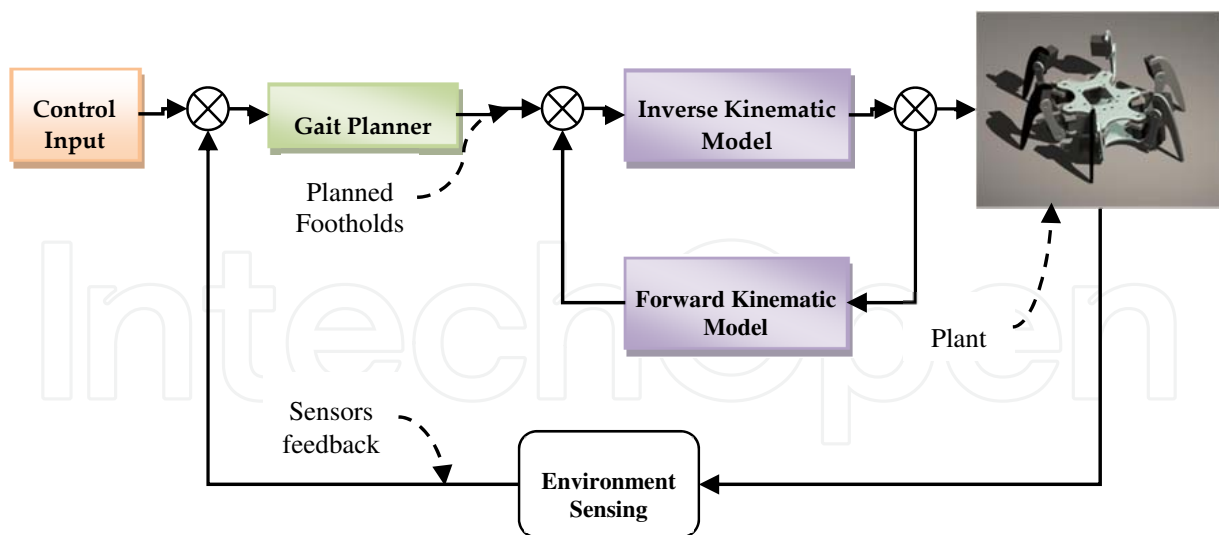


Figure 1. Overall control loop of the simulation model.

2.1. Plant subsystem

The “plant” is shown in Fig. 2 in the form of a block diagram that provides complete information about the mass, motion axes and moments of inertia of the bodies and their

subassemblies. Each leg attached to the main body is considered as a 3-degree of freedom serial manipulator which acts as the actuating element in the overall simulation model. Furthermore, the block diagram shows joints to describe appropriate connections between the rigid bodies which provide an opportunity for actuation and sensing. Fig. 3 & Fig. 4 show the respective subsystems of a hexapod robot as investigated in [8, 9].

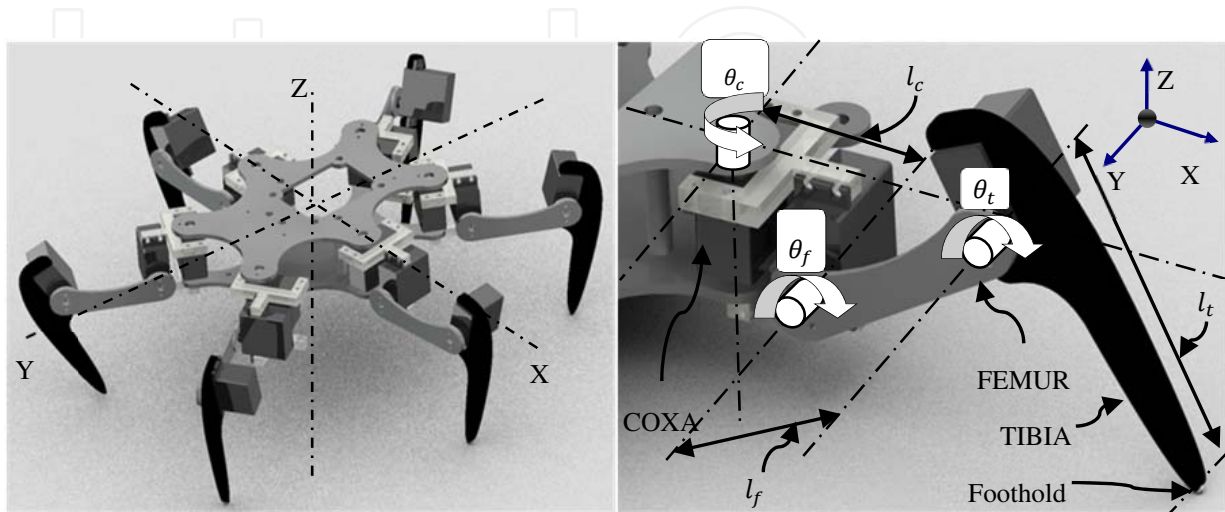


Figure 2. A six-legged hexapod robot and its leg kinematic representation.

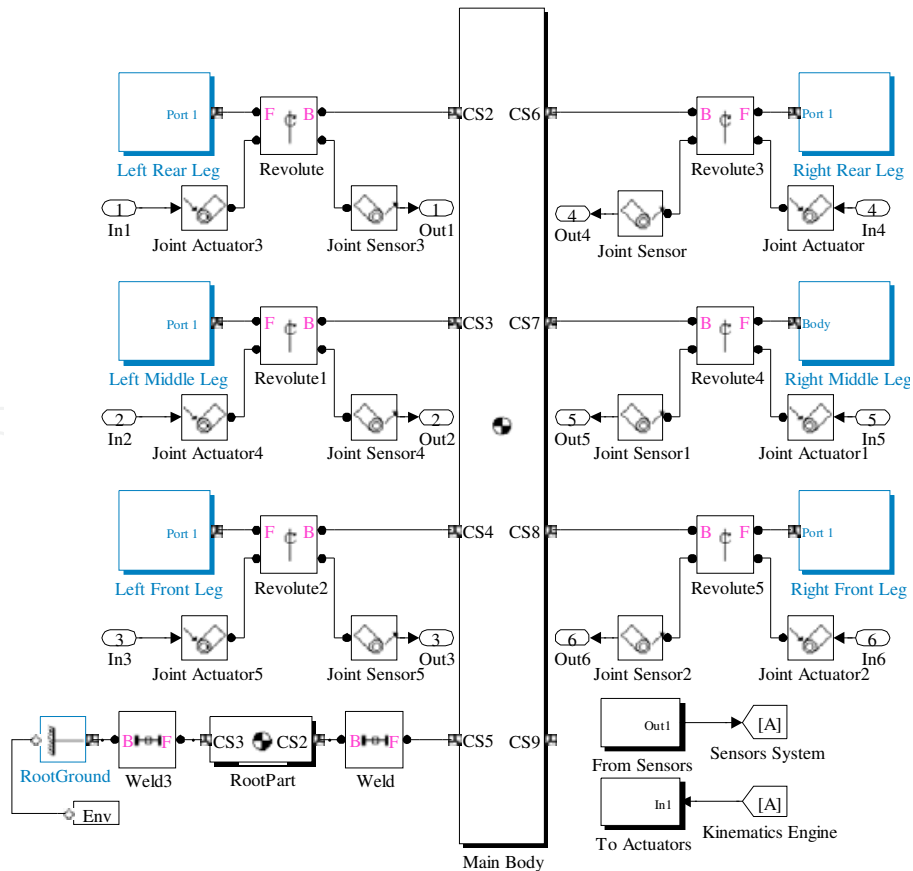


Figure 3. SimMechanics model representing the plant subsystem of a hexapod robot.

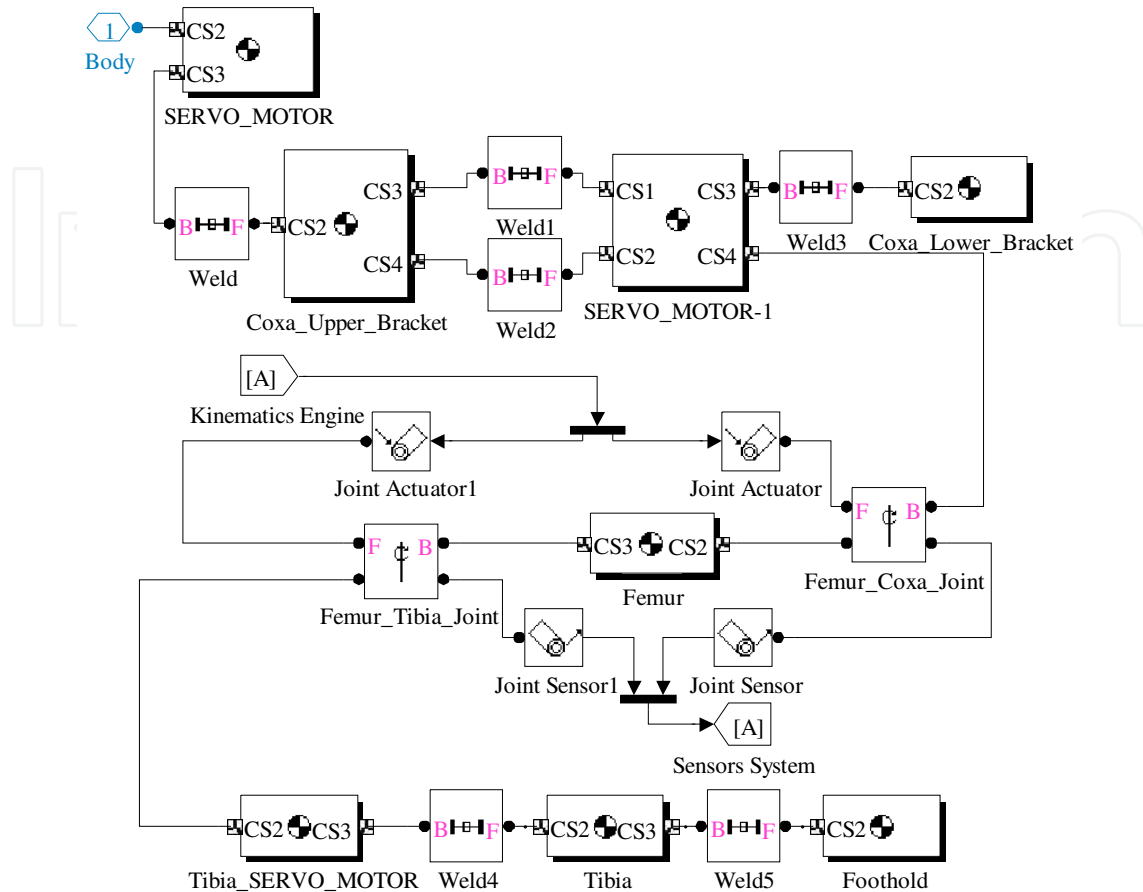


Figure 4. SimMechanics model of an individual leg as a 3-DOF serial manipulator.

2.2. Kinematics engine

The kinematic model of the robot reported here uses forward and inverse kinematic equations of a 3-DOF serial manipulator to control the foothold in 3D-space. The research studies [8 - 14] describe the mathematical modelling of a 3-DOF leg in 3D space in terms of a forward kinematic model. The details are given in the following subsections.

2.2.1. Forward kinematics

Each leg is modelled as a 3-DOF serial manipulator comprised of three members as: coxa, femur and tibia inter-related using pin joints. The joint connecting the body with the coxa (BodyCoxa-joint) is represented by θ_c , the joint connecting coxa with the femur (CoxaFemur-joint) is represented by θ_f , and the joint connecting the femur with the tibia (FemurTibia-joint) is represented by θ_t . The position of the foothold $[X_t \ Y_t \ Z_t]^T$ with respect to BodyCoxa-joint motion axis is obtained using Denavit Hartenberg convention and expressed as a general homogenous transformation expression as given by (1). Table1 enlists the D-H parameters of the robot leg shown in Fig.5.

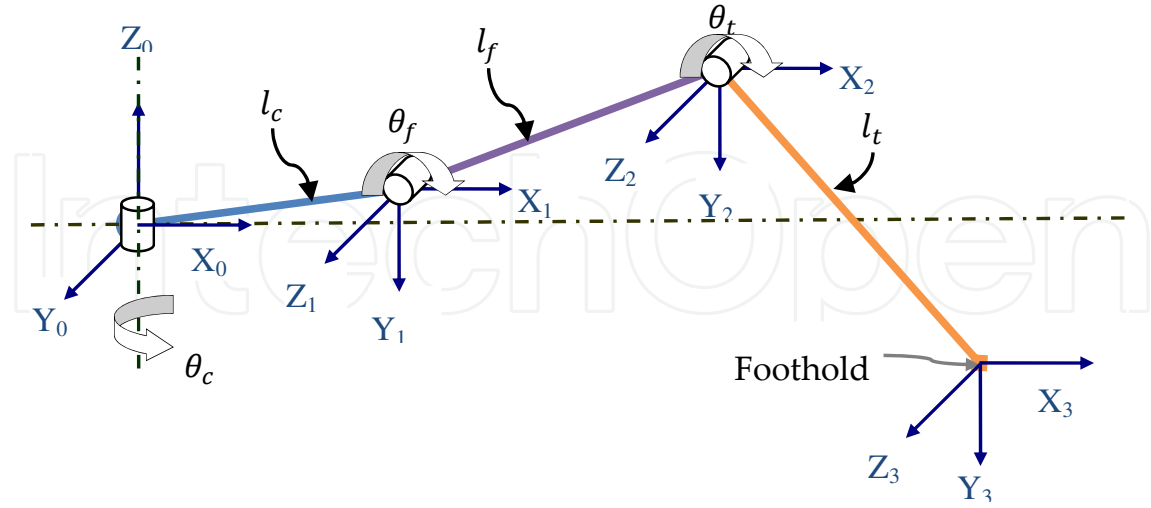


Figure 5. Kinematic Configuration of an individual leg.

i	α_i	θ_i	a_i	d_i
0	0	0	0	0
1	90	θ_c	l_c	0
2	0	θ_f	l_f	0
3	0	θ_t	l_t	0

Table 1. D-T parameters of leg shown in Fig. 5

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \cos\theta_c & 0 & \sin\theta_c & l_c \cos\theta_c \\ \sin\theta_c & 0 & -\cos\theta_c & l_c \sin\theta_c \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times$$

$$\begin{bmatrix} \cos\theta_f & -\sin\theta_f & 0 & l_f \cos\theta_f \\ \sin\theta_f & \cos\theta_f & 0 & l_f \sin\theta_f \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \cos\theta_t & -\sin\theta_t & 0 & l_t \cos\theta_t \\ \sin\theta_t & \cos\theta_t & 0 & l_t \sin\theta_t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

Solving (1) yields (2) which can be further simplified to (3).

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} l_c \cos\theta_c + l_f \cos\theta_c \cos\theta_f + l_t \cos\theta_c \cos\theta_f \cos\theta_t - l_t \cos\theta_c \sin\theta_f \sin\theta_t \\ l_c \sin\theta_c + l_f \cos\theta_f \sin\theta_c + l_t \cos\theta_f \sin\theta_c \cos\theta_t - l_t \sin\theta_c \sin\theta_f \sin\theta_t \\ l_f \sin\theta_f + l_t \cos\theta_f \sin\theta_t + l_t \cos\theta_t \sin\theta_f \end{bmatrix} \quad (2)$$

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \cos\theta_c (l_c + l_t \cos\theta_{f+t} + l_f \cos\theta_f) \\ \sin\theta_c (l_c + l_t \cos\theta_{f+t} + l_f \cos\theta_f) \\ l_t \sin\theta_{f+t} + l_f \sin\theta_f \end{bmatrix} \quad (3)$$

2.2.2. X-Y-Z (roll-pitch-yaw) body rotations

In order to obtain the position of foothold with reference to robot motion centroid, the body rotations in 3D-space must be brought into consideration. Body rotations (Roll-Pitch-Yaw) representation for orientation is given by (4).

$$\begin{cases} \text{Roll} = \text{psi } (\psi) = \text{Body Rotation about X – axis} \\ \text{Yaw} = \text{gamma } (\gamma) = \text{Body Rotation about Z – axis} \\ \text{Pitch} = \text{phi } (\varphi) = \text{Body Rotation about Y – axis} \end{cases}$$

$$Ft = R_{x,\psi} \times R_{y,\varphi} \times R_{z,\gamma} \quad (4)$$

Using general homogenous transformation matrices, the foot location with respect to body coordinate system can be obtained using (4) as given by (5)

$$\begin{bmatrix} Ft_x \\ Ft_y \\ Ft_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\psi & -\sin\psi & 0 \\ 0 & \sin\psi & \cos\psi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \cos\varphi & 0 & \sin\varphi & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\varphi & 0 & \cos\varphi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 & 0 \\ \sin\gamma & \cos\gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \quad (5)$$

$$\begin{bmatrix} Ft_x \\ Ft_y \\ Ft_z \end{bmatrix} =$$

$$= \begin{bmatrix} T_z \sin\gamma + T_x \cos\gamma \cos\varphi - T_y \cos\gamma \sin\varphi \\ T_x (\cos\psi \sin\varphi + \cos\varphi \sin\psi \sin\psi) + T_y (\cos\varphi \cos\psi - \sin\psi \sin\varphi \sin\psi) - T_z \cos\gamma \sin\psi \\ T_x (\sin\varphi \sin\psi - \cos\varphi \cos\psi \sin\gamma) + T_y (\cos\varphi \sin\psi + \cos\psi \sin\gamma \sin\varphi) + T_z \cos\gamma \cos\psi \end{bmatrix} \quad (6)$$

Where,

$$\begin{cases} T_x = \text{LegOffsetX} + Xt \\ T_y = \text{LegOffsetY} + Yt \\ T_z = \text{LegOffsetZ} + Zt \end{cases}$$

Thus, the forward kinematic model of the leg can be expressed by (6), also investigated in [13].

2.2.3. Inverse kinematics

In order to obtain the inverse kinematics solution, we solve the general homogenous transformation expression for a 3-DOF serial manipulator in 3D-space given by (7) as (8) & (9).

$$R = T_1 \times T_2 \times T_3 \quad (7)$$

$$\text{inv}(T_1) \times R = T_2 \times T_3 \quad (8)$$

$$\text{inv}(T_2) \times \text{inv}(T_1) \times R = T_3 \quad (9)$$

Where,

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} & P_x \\ r_{21} & r_{22} & r_{23} & P_y \\ r_{31} & r_{32} & r_{33} & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}, T_1 = \begin{bmatrix} \cos\theta_c & 0 & \sin\theta_c & l\cos\theta_c \\ \sin\theta_c & 0 & -\cos\theta_c & l\sin\theta_c \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_2 = \begin{bmatrix} \cos\theta_f & -\sin\theta_f & 0 & lf\cos\theta_f \\ \sin\theta_f & \cos\theta_f & 0 & lf\sin\theta_f \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, T_3 = \begin{bmatrix} \cos\theta_t & -\sin\theta_t & 0 & lt\cos\theta_t \\ \sin\theta_t & \cos\theta_t & 0 & lt\sin\theta_t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Solving (8) yields (10)

$$\begin{bmatrix} \cos\theta_c & \sin\theta_c & 0 & -lc \\ 0 & 0 & 1 & 0 \\ \sin\theta_c & -\cos\theta_c & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} r_{11} & r_{12} & r_{13} & P_x \\ r_{21} & r_{22} & r_{23} & P_y \\ r_{31} & r_{32} & r_{33} & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \cos\theta_f & -\sin\theta_f & 0 & lf\cos\theta_f \\ \sin\theta_f & \cos\theta_f & 0 & lf\sin\theta_f \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \cos\theta_t & -\sin\theta_t & 0 & lt\cos\theta_t \\ \sin\theta_t & \cos\theta_t & 0 & lt\sin\theta_t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10)$$

Comparing r_{34} elements of both sides of the (10) yields (11).

$$P_x \sin\theta_c - P_y \cos\theta_c = 0 \quad (11)$$

Thus, BodyCoxa joint angle θ_c can be determined from (11) as given by (12).

$$\theta_c = ATAN2(P_y, P_x) \quad (12)$$

Solving (9) yields (13)

$$\begin{bmatrix} \cos\theta_f & \sin\theta_f & 0 & -lf \\ -\sin\theta_f & \cos\theta_f & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} \cos\theta_c & \sin\theta_c & 0 & -lc \\ 0 & 0 & 1 & 0 \\ \sin\theta_c & -\cos\theta_c & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times$$

$$\times \begin{bmatrix} r_{11} & r_{12} & r_{13} & P_x \\ r_{21} & r_{22} & r_{23} & P_y \\ r_{31} & r_{32} & r_{33} & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta_t & -\sin\theta_t & 0 & lt \times \cos\theta_t \\ \sin\theta_t & \cos\theta_t & 0 & lt \times \sin\theta_t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (13)$$

Comparing r_{14}, r_{24} elements of both sides of (13) we get (14) & (15)

$$\cos\theta_f(P_x \cos\theta_c - lc + P_y \sin\theta_c) - lf + P_z \sin\theta_f = lt \cos\theta_t \quad (14)$$

$$P_z \cos\theta_f - \sin\theta_f(P_x \cos\theta_c - lc + P_y \sin\theta_c) = lt \sin\theta_t \quad (15)$$

Taking positive squares of both sides of (14) & (15) and summing them-up yields (16).

$$[\cos\theta_f \times A - lf + P_z \times \sin\theta_f]^2 + [P_z \times \cos\theta_f - \sin\theta_f \times A]^2 = lt^2 \quad (16)$$

Where,

$$A = (P_x \cos \theta_c - lc + P_y \sin \theta_c)$$

Further solving (16) yields:

$$\frac{(A^2 + lf^2 + P_z^2 - lt^2)}{2lf} = \cos \theta_f \times A + \sin \theta_f P_z$$

$$G = \cos \theta_f \times A + \sin \theta_f P_z \quad (17)$$

Where,

$$G = \frac{(A^2 + lf^2 + P_z^2 - lt^2)}{2lf}$$

Using the trigonometry identities given by (18) & (19), (17) can be further solved to determine CoxaFemur angle as given by (20)

$$\cos \theta_x \times a + \sin \theta_x \times b = c \quad (18)$$

$$\theta_x = \text{ATAN2}(b, a) + \text{ATAN2}(\sqrt{a^2 + b^2 - c^2}, c) \quad (19)$$

$$\theta_f = \text{ATAN2}(P_z, A) + \text{ATAN2}(\sqrt{A^2 + P_z^2 - G^2}, G) \quad (20)$$

Where,

$$a = A, b = P_z, c = G$$

Dividing (14) by (15) yields (21) which can be further simplified to obtain FemurTibia angle as given by (22)

$$\frac{lt \sin \theta_t}{lc \cos \theta_t} = \frac{P_z \cos \theta_f - \sin \theta_f (P_x \cos \theta_c - lc + P_y \sin \theta_c)}{\cos \theta_f (P_x \cos \theta_c - lc + P_y \sin \theta_c) - lf + P_z \sin \theta_f} \quad (21)$$

$$\theta_t = \text{ATAN2}((P_z \cos \theta_f - \sin \theta_f A), (A \times \cos \theta_f - lf + \sin \theta_f P_z)) \quad (22)$$

2.3. Robot dynamics

Lagrange-Euler formulation is selected here to derive the dynamics of the robot leg. The Lagrange-Euler equations yield a dynamic expression that can be expressed as given by (23).

$$\tau_e - J^T F = M(q)\ddot{q} + H(q, \dot{q}) + G(q) \quad (23)$$

In (23), $M(q)$ represents the mass matrix, H incorporates the centrifugal and Coriolis terms, and $G(q)$ is the gravity matrix. τ_e contains active joint torques and F is a matrix representing ground contact forces estimated using force sensors embedded in each foot.

2.3.1. Foothold dynamics

The position of the foothold with respect to BodyCoxa joint motion axis is given by (3). Time derivate of (3) yields velocity of the foothold as given by (24) which can be expressed in form of Jacobean as given by (25)

$$\begin{bmatrix} \dot{X}_t \\ \dot{Y}_t \\ \dot{Z}_t \end{bmatrix} = \begin{bmatrix} -(lc \times \sin\theta_c + lf \cos\theta_f \sin\theta_c + lt \sin\theta_c \cos\theta_{f+t}) & -(lf \cos\theta_c \sin\theta_f + lt \cos\theta_c \sin\theta_{f+t}) \\ lc \times \cos\theta_c + lf \cos\theta_c \cos\theta_f + lt \cos\theta_c \cos\theta_{f+t} & -(lf \sin\theta_c \sin\theta_f + lt \sin\theta_c \sin\theta_{f+t}) \\ 0 & lf \cos\theta_f + lt \cos\theta_{f+t} \\ -(lt \times \cos\theta_c \times \sin\theta_{f+t}) \\ -(lt \times \sin\theta_c \times \sin\theta_{f+t}) \\ lt \times \cos\theta_{f+t} \end{bmatrix} \times \begin{bmatrix} \dot{\theta}_c \\ \dot{\theta}_f \\ \dot{\theta}_t \end{bmatrix} \quad (24)$$

$$\dot{X}_t = J_D \dot{\theta}_t \quad (25)$$

Time derivate of (25) yields acceleration of the foothold as given by (26)

$$\ddot{X}_t = \dot{J}_D \dot{\theta}_t + J_D \ddot{\theta}_t \quad (26)$$

2.3.2. Femurtibia joint dynamics

The position of the FemurTibia joint with respect to BodyCoxa joint motion axis is given by (27).

$$\begin{bmatrix} X_f \\ Y_f \\ Z_f \end{bmatrix} = \begin{bmatrix} lc \cos\theta_c + lf \cos\theta_c \cos\theta_f \\ (lc \sin\theta_c + lf \cos\theta_f \sin\theta_c) \\ 0 \end{bmatrix} \quad (27)$$

Time derivate of (27) yields velocity matrix as given by (28) which can be expressed in terms of Jacobean as (29)

$$\begin{bmatrix} \dot{X}_f \\ \dot{Y}_f \\ \dot{Z}_f \end{bmatrix} = \begin{bmatrix} -(lc \sin\theta_c + lf \sin\theta_c \cos\theta_f) & -(lf \cos\theta_c \sin\theta_f) & 0 \\ lc \cos\theta_c + lf \cos\theta_c \cos\theta_f & -(lf \sin\theta_c \sin\theta_f) & 0 \\ 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} \dot{\theta}_c \\ \dot{\theta}_f \\ \dot{\theta}_t \end{bmatrix} \quad (28)$$

$$\dot{X}_f = J_D \dot{\theta}_f \quad (29)$$

Time derivate of (29) yields acceleration matrix as given by (30) which can be further expressed in form of Jacobean as given by (31).

$$\begin{bmatrix} \ddot{X}_f \\ \ddot{Y}_f \\ \ddot{Z}_f \end{bmatrix} = \begin{bmatrix} -lc \cos\theta_c \dot{\theta}_c^2 - lf \cos\theta_c \cos\theta_f \dot{\theta}_c^2 - lf \cos\theta_c \cos\theta_f \dot{\theta}_f^2 - \ddot{\theta}_c lf \cos\theta_f \sin\theta_c - \\ -lc \sin\theta_c \dot{\theta}_c^2 - lf \cos\theta_f \sin\theta_c \dot{\theta}_c^2 - lf \sin\theta_c \cos\theta_f \dot{\theta}_f^2 + \ddot{\theta}_c lf \cos\theta_f \cos\theta_c - \\ 0 \\ \ddot{\theta}_f lf \cos\theta_c \sin\theta_f - \ddot{\theta}_c lc \sin\theta_c + 2\dot{\theta}_c \dot{\theta}_f lf \sin\theta_c \sin\theta_f \\ \ddot{\theta}_f lf \sin\theta_c \sin\theta_f + \ddot{\theta}_c lc \cos\theta_c - 2\dot{\theta}_c \dot{\theta}_f lf \cos\theta_c \sin\theta_f \\ 0 \end{bmatrix} \quad (30)$$

$$\ddot{X}_f = J_D \dot{\theta}_f + J_D \ddot{\theta}_f \quad (31)$$

2.3.3. Coxafemur joint dynamics

The position of the CoxaFemur joint with respect to BodyCoxa joint motion axis is given by (32).

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \begin{bmatrix} lccos\theta_c \\ lcsin\theta_c \\ 0 \end{bmatrix} \quad (32)$$

Time derivate of (32) yields velocity matrix as given by (33) which can be expressed in terms of Jacobean as (34).

$$\begin{bmatrix} \dot{X}_c \\ \dot{Y}_c \\ \dot{Z}_c \end{bmatrix} = \begin{bmatrix} -lc \sin\theta_c & 0 & 0 \\ lc \cos\theta_c & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \times \begin{bmatrix} \dot{\theta}_c \\ \dot{\theta}_f \\ \dot{\theta}_t \end{bmatrix} \quad (33)$$

$$\dot{X}_c = J_D \dot{\theta}_c \quad (34)$$

Time derivate of (34) yields acceleration matrix (35) which can be expressed in terms of Jacobean as (36).

$$\begin{bmatrix} \ddot{X}_c \\ \ddot{Y}_c \\ \ddot{Z}_c \end{bmatrix} = \begin{bmatrix} -lc \cos\theta_c \dot{\theta}_c^2 - \ddot{\theta}_c lcsin\theta_c \\ -lc \sin\theta_c \dot{\theta}_c^2 + \ddot{\theta}_c lccos\theta_c \\ 0 \end{bmatrix} \quad (35)$$

$$\ddot{X}_c = J_D \dot{\theta}_c + J_D \ddot{\theta}_c \quad (36)$$

2.4. Control algorithm

Dynamics of a robot as given by (23) can be further described using Lagrange-Euler formulations as given by (37).

$$Q = M(q)\ddot{q} + H(q, \dot{q}) + G(q) \quad (37)$$

Where,

$$M = \sum_{i=1}^n (J_{Mi}^T m_i J_{Mi} + \frac{1}{2} J_{Ri}^T I_i^0 J_{Ri})$$

The kinetic energy and potential energy of a link can be expressed as given by (38) & (39).

$$K = \frac{1}{2} \dot{q}^T M(q) \dot{q} \quad (38)$$

$$P = \sum_{i=1}^n m_i g^T r_{ci} \quad (39)$$

The mass matrix M of the robot leg can be written by assuming the leg as a serial manipulator consisting of three links as given by (40).

$$M = \sum_{i=1}^3 (J_{Mi}^T m_i J_{Mi} + \frac{1}{2} J_{Ri}^T I_i^0 J_{Ri}) \quad (40)$$

The moments of inertia for each link (coxa, femur & tibia) of the leg can be determined from (41), (42) & (43).

$$I_c^0 = T_c^0 I_c^c T_c^T \quad (41)$$

$$I_f^0 = T_f^0 I_f^f T_f^T \quad (42)$$

$$I_t^0 = T_t^0 I_t^t T_t^T \quad (43)$$

The Jacobean terms of (40) can be determined from expressions (44), (45) & (46).

$$J_{Mc} = \begin{bmatrix} -lc \sin\theta_c & 0 & 0 \\ lc \cos\theta_c & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, J_{Rc} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad (44)$$

$$J_{Mf} = \begin{bmatrix} -(lc \sin\theta_c + lf \sin\theta_c \cos\theta_f) & -(lf \cos\theta_c \sin\theta_f) & 0 \\ lc \cos\theta_c + lf \cos\theta_c \cos\theta_f & -(lf \sin\theta_c \sin\theta_f) & 0 \\ 0 & 0 & 0 \end{bmatrix}, J_{Rf} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad (45)$$

$$J_{Mt} = \begin{bmatrix} -(lcsin\theta_c + lfcos\theta_f \sin\theta_c + ltsin\theta_c \cos\theta_{f+t}) & -(lfcos\theta_c \sin\theta_f + ltcos\theta_c \sin\theta_{f+t}) \\ lccos\theta_c + lfcos\theta_c \cos\theta_f + ltcos\theta_c \cos\theta_{f+t} & -(lfsin\theta_c \sin\theta_f + ltsin\theta_c \sin\theta_{f+t}) \\ 0 & lfcos\theta_f + ltcos\theta_{f+t} \\ -(lcos\theta_c \sin\theta_{f+t}) \\ -(ltsin\theta_c \sin\theta_{f+t}) \\ lcos\theta_{f+t} \end{bmatrix}, J_{Rt} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad (46)$$

The velocity coupling vector H and the gravitational matrices are given by (47) & (48)

$$H_{ijk} = \sum_{j=1}^n \sum_{k=1}^n \left(\frac{\partial M_{ij}}{\partial q_k} - \frac{1}{2} \frac{\partial M_{jk}}{\partial q_i} \right) \quad (47)$$

$$G_i = \sum_{j=1}^n m_j g^T J_{Mj}^i \quad (48)$$

Using desired time-based trajectories of each joint, the robot can be commanded to follow a desired path with the help of a computed torque control law as given by (49)

$$D(q)(\ddot{q} - k_D \dot{e} - k_p e + H(q, \dot{q}) + G(q)) = Q \quad (49)$$

The error vector “ e ” is given by (50)

$$e = q - q_d \quad (50)$$

Thus, the computed control law can be described by (51).

$$D(q)\ddot{q}_d + H(q, \dot{q}) + G(q) + D(q)(-k_D\dot{e} - k_P e) = Q \quad (51)$$

3. Gait generation

Gait generation is the main topic of investigation in the field of legged locomotion. Mobility and stability are two important factors which require specific attention while planning a gait for locomotion in natural environments. Generally, the gaits found in nature feature sequential

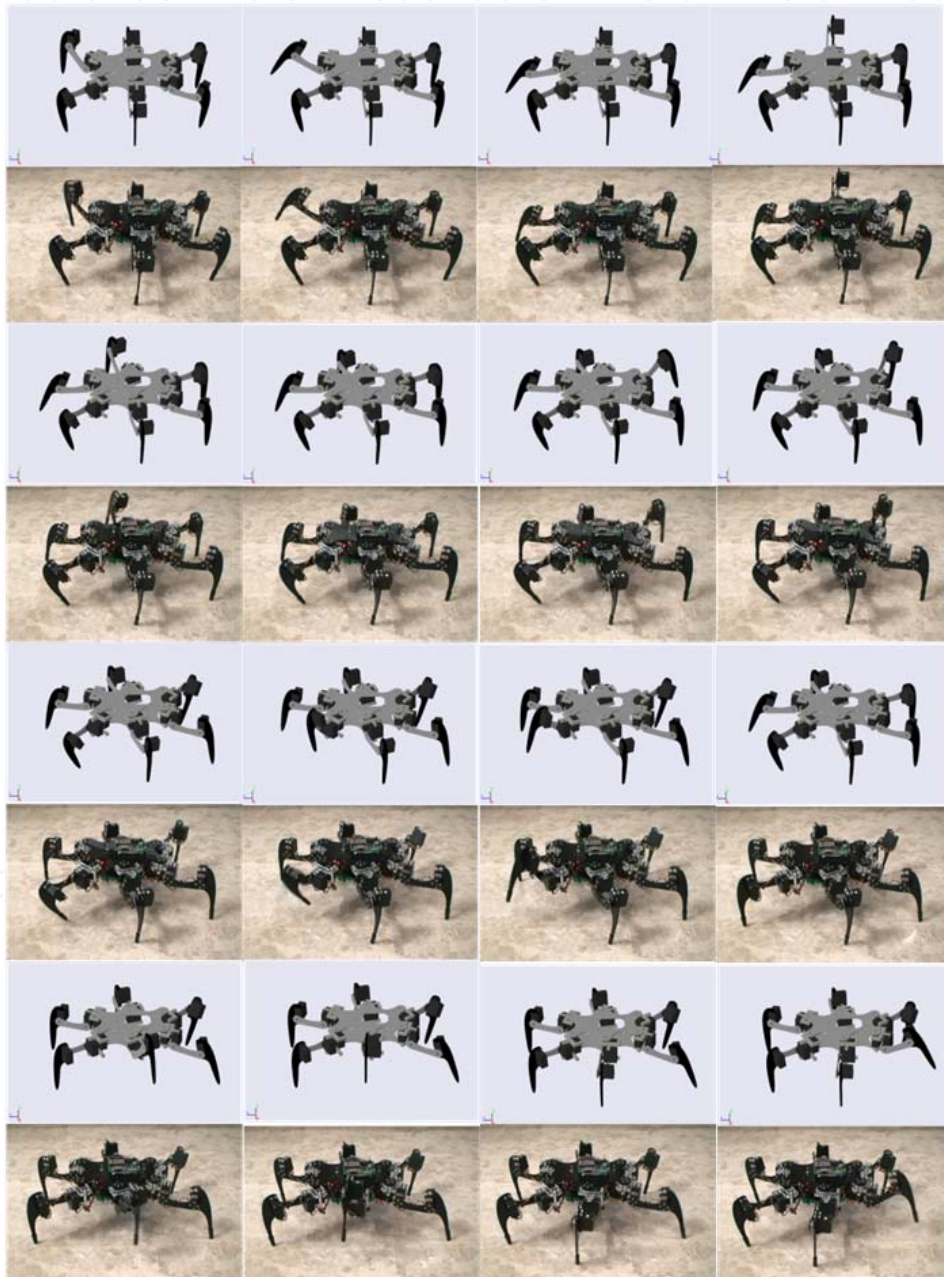


Figure 6. A comparative representation of wave gait generation in simulation model and actual prototype robot over a smooth flat surface.

motion of legs and the body. A large volume of work [5-14] deals with biologically-inspired gaits to replicate this sequential motion pattern found in insects into real robots. The gaits investigated here are cyclic periodic continuous gaits (wave gait & ripple gait) as investigated in [10 & 12]. Gait generation occurs in the gait planner shown in Fig. 1, where commanded foothold locations and orientations are combined with gait sequences to generate a sequential pattern of footholds in 3D-space. The computed leg-sequential pattern is submitted to the inverse kinematic model to determine appropriate joint angles to achieve the desired location. The motion of each leg is determined by the mathematical model. The sequential patterns of leg-lifts for a 12-step ripple gait and a 19-step wave gait are shown in Fig. 16 while, Fig. 6 shows a pictorial demonstration of the gait generation in simulation model and the actual prototype robot over a smooth flat surface.

SimMechanics provides a diverse library of software sensors to obtain information from the environment. These sensors mainly include rigid-body sensors and joint sensors. As shown in the figure, the sensors subsystem outlines a network of sensors employed at each leg and their respective joints. Sensors attached to the legs are used to obtain the absolute location and orientation of each foothold in 3D-space while the joint sensors provide information about each joint's linear and angular rates and accelerations. This sensory network acquires the information from each sensor, filters the data to reject noise. Motion planner, uses the feedback to compute corrections between the commanded data and the recorded data. A Butterworth high-pass 6th order digital filter is used to filter the sensor noise.

4. Dynamic simulations

The objective of carrying out dynamic simulations is to test the validity of the mathematical model and perform gait analysis over uneven terrains. In order to realize the first objective, the mathematical model is simulated first in forward dynamics mode and then in inverse dynamics mode. The robot is commanded to walk between two fixed points 2 meters apart. Straight line walking is achieved with an attitude control using GPS and Compass sensors feedback. Fig. 7 show respective views of robot straight-line walking achieved over flat surfaces using Microsoft Robotics Studio [15].

4.1. Test 1 – Forward dynamics mode

In forward dynamics mode, the objective is to analyze the torques produced at the joints provided each joint is actuated with a motion profile consisting of joint rotation, velocity and acceleration. The robot is steered with a periodic cyclic gait. The footholds planned by the gait generator are passed to the kinematics engine and appropriate joint angles are computed in return. The obtained joint angle profile of each joint is differentiated to compute joint rates and accelerations. Finally, the computed joint motion profiles are fed to the joint actuators to simulate the motion. Joint sensors are used to record the actual

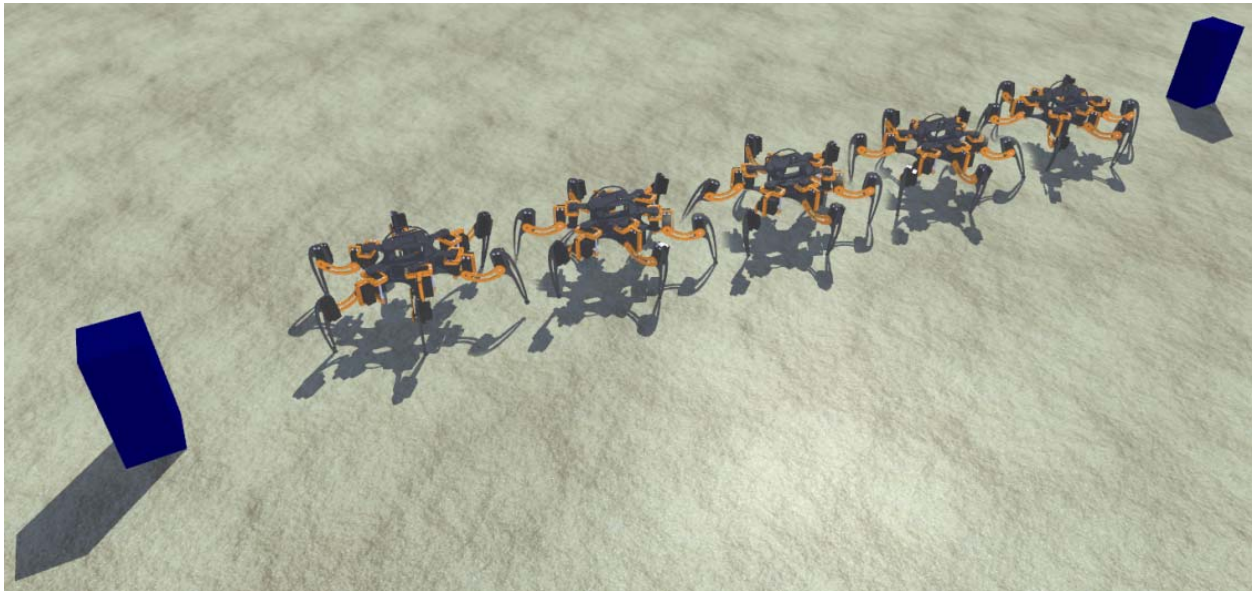
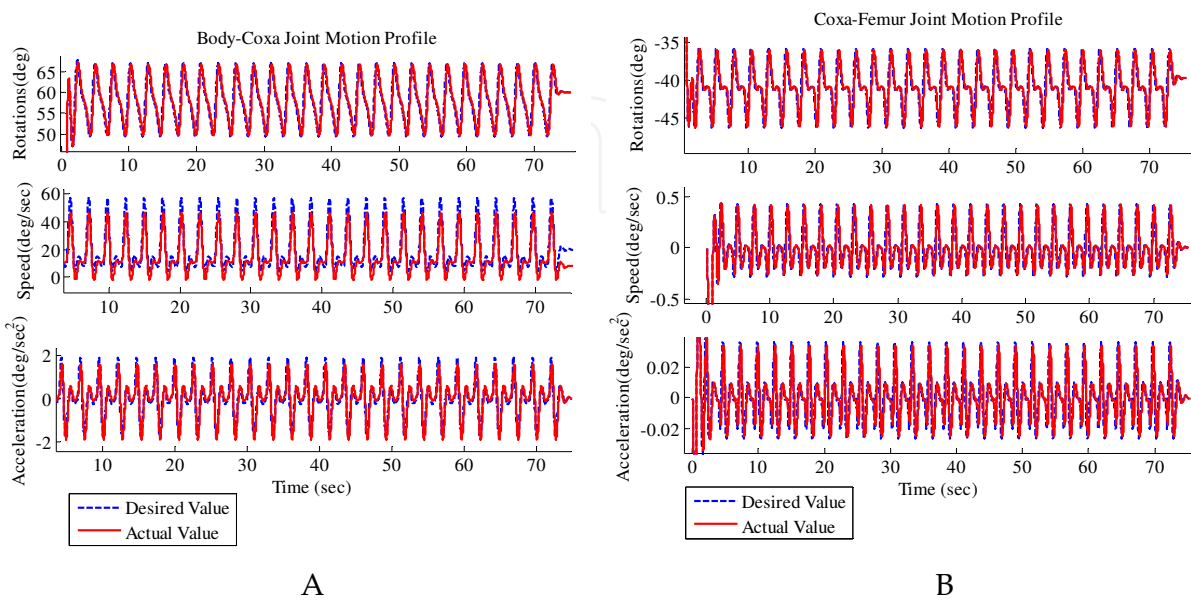


Figure 7. A view of a hexapod robot being steered between two fixed points (2-meter apart) over a flat surface.

torque produced during the entire sequential motion. Actual torques data obtained from the simulation and those determined by the dynamic model are plotted as shown by Fig. 8 (D).

4.2. Test 2 – Inverse dynamics mode

In inverse dynamics mode, the objective is to analyze the joint motion profiles provided the joints are actuated using torques. Joint sensors are used to record joint rotations, velocities



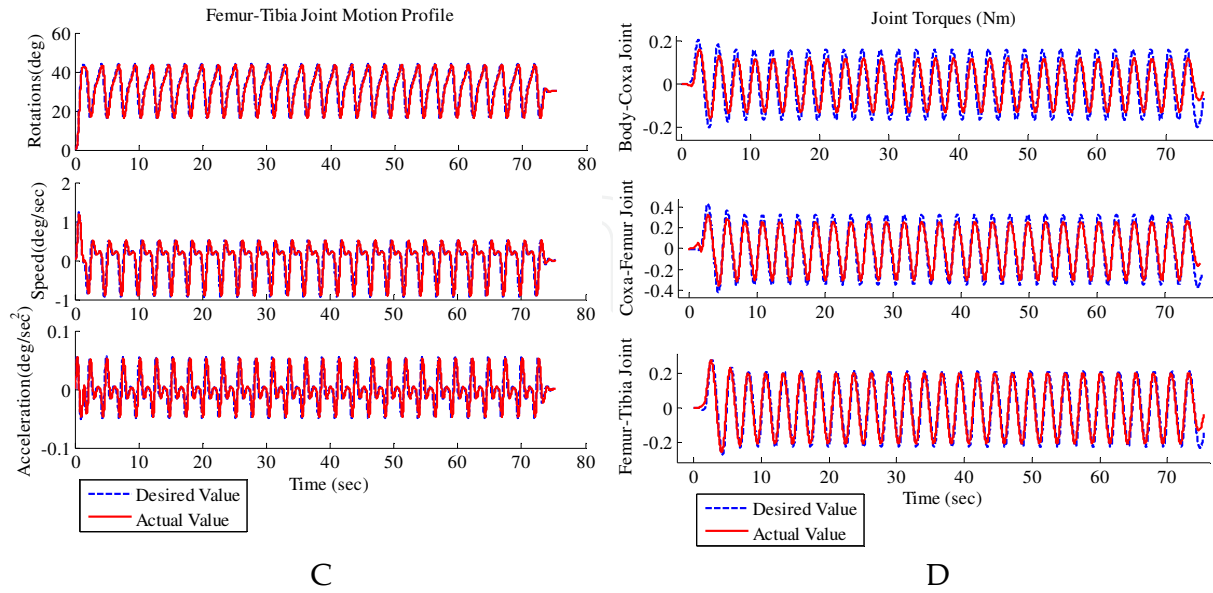


Figure 8. Comparative representation of sensors feedback and mathematical model results. (A) Motion profile of BodyCoxa joint. (B) Motion profile of CoxaFemur joint. (C) Motion profile of FemurTibia joint. (D) Joint torques.

and accelerations which are then matched with those determined by the dynamic model to study the errors. The comparative representation of actual motion data and desired data is shown by Fig. 8(A), Fig. 8(B) & Fig. 8(C). The close matching of the actual and desired data confirms the validity of inverse model for walking over flat surfaces.

4.3. Test 3 – Gait analysis over an uneven terrain

The objective of this test is to conduct a gait analysis in terms of robot speed, joint reaction torques and reaction forces at the foothold during robot-environment interactions.

4.3.1. Reaction forces at the foothold and reaction torques at the joints

In order to study an adaptive gait over an uneven terrain, we must first describe the joint reaction torques due to the reaction forces which arise at the footholds during the leg-terrain interactions.

4.3.1.1. Reaction torque at bodycoxa joint

Considering Fig. 9, the reaction torque about the BodyCoxa joint motion axis can be expressed by (52), further explained by (53) & (54).

$$\vec{\tau}_{BodyCoxa} = \{\vec{r}_1 \times \vec{F}_N + \vec{r}_1 \times \vec{F}_R + \vec{r}_2 \times \vec{W}_{Coxa}\}Nm \quad (52)$$

$$\left\{ \begin{array}{l} \vec{r}_1 = \{(Ft_{x_i}^G - X_c)\hat{i} + (Ft_{y_i}^G - Y_c)\hat{j} + (Ft_{z_i}^G - Z_c)\hat{k}\}m \\ \vec{r}_2 = \{(COM_{x_i}^G - X_c)\hat{i} + (COM_{y_i}^G - Y_c)\hat{j} + (COM_{z_i}^G - Z_c)\hat{k}\}m \\ \vec{F}_N = \text{Normal reaction force vector at the foothold} \\ \vec{F}_R = \text{Friction force vector at the foothold} \\ \vec{W}_{coxa} = \text{Weight of the coxa link} \\ \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \begin{bmatrix} lccos\theta_c \\ lcsin\theta_c \\ 0 \end{bmatrix} = \text{Position of Bodycoxa joint axis} \end{array} \right. \quad (53)$$

$$\begin{aligned} \vec{\tau}_{BodyCoxa} = & \{(Ft_{x_i}^G - X_c)\hat{i} + (Ft_{y_i}^G - Y_c)\hat{j} + (Ft_{z_i}^G - Z_c)\hat{k}\} \times \{F_{Nx}\hat{i} + F_{Ny}\hat{j} + F_{Nz}\hat{k}\} + \\ & \{(Ft_{x_i}^G - X_c)\hat{i} + (Ft_{y_i}^G - Y_c)\hat{j} + (Ft_{z_i}^G - Z_c)\hat{k}\} \times \{F_{Rx}\hat{i} + F_{Ry}\hat{j} + F_{Rz}\hat{k}\} + \\ & \{(COM_{x_i}^G - X_c)\hat{i} + (COM_{y_i}^G - Y_c)\hat{j} + (COM_{z_i}^G - Z_c)\hat{k}\} \times \{W_{coxa}\hat{j}\} \end{aligned} \quad (54)$$

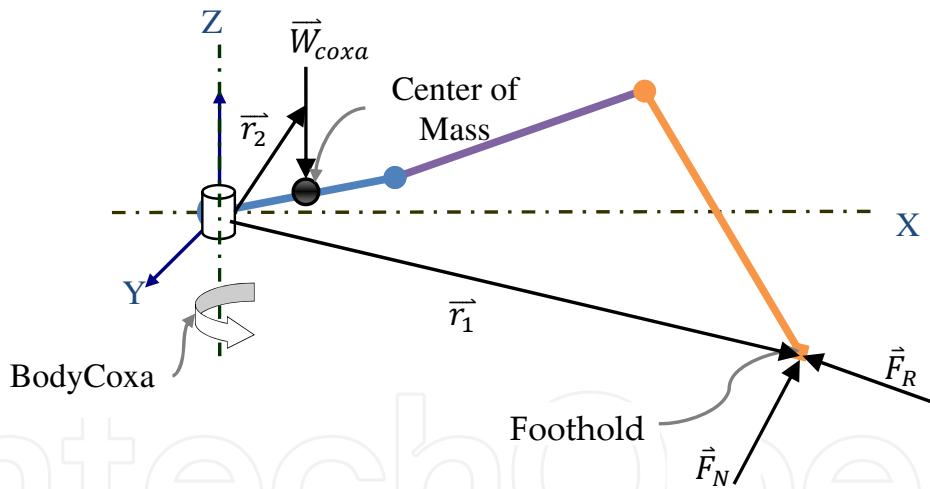


Figure 9. The total torque about BodyCoxa joint axis is equal to the sum of moments due to weight of the coxa link and moments due to friction force and normal reaction forces acting at the foothold.

4.3.1.2. Reaction torque at coxafemur joint

Considering Fig. 10, the reaction torque about the CoxaFemur joint motion axis can be expressed by (55), further explained in (56) & (57).

$$\vec{\tau}_{CoxaFemur} = \{\vec{r}_3 \times \vec{F}_N + \vec{r}_3 \times \vec{F}_R + \vec{r}_4 \times \vec{W}_{femur}\}Nm \quad (55)$$

$$\left\{ \begin{array}{l} \vec{r}_3 = \{(Ft_{x_i}^G - X_f)\hat{i} + (Ft_{y_i}^G - Y_f)\hat{j} + (Ft_{z_i}^G - Z_f)\hat{k}\}m \\ \vec{r}_4 = \{(COM_{x_i}^G - X_f)\hat{i} + (COM_{y_i}^G - Y_f)\hat{j} + (COM_{z_i}^G - Z_f)\hat{k}\}m \\ \vec{F}_N = \text{Normal reaction force vector at the foothold} \\ \vec{F}_R = \text{Friction force vector at the foothold} \\ \vec{W}_{femur} = \text{Weight of the femur link} \end{array} \right. \quad (56)$$

$$\begin{bmatrix} X_f \\ Y_f \\ Z_f \end{bmatrix} = \begin{bmatrix} lc \cos\theta_c + lf \cos\theta_c \cos\theta_f \\ (lc \sin\theta_c + lf \cos\theta_f \sin\theta_c) \\ 0 \end{bmatrix} = \text{Position of CoxaFemur joint axis}$$

$$\begin{aligned} \vec{\tau}_{CoxaFemur} = & \{(Ft_{x_i}^G - X_f)\hat{i} + (Ft_{y_i}^G - Y_f)\hat{j} + (Ft_{z_i}^G - Z_f)\hat{k}\} \times \{F_{Nx}\hat{i} + F_{Ny}\hat{j} + F_{Nz}\hat{k}\} + \\ & \{(Ft_{x_i}^G - X_f)\hat{i} + (Ft_{y_i}^G - Y_f)\hat{j} + (Ft_{z_i}^G - Z_f)\hat{k}\} \times \{F_{Rx}\hat{i} + F_{Ry}\hat{j} + F_{Rz}\hat{k}\} + \\ & \{(COM_{x_i}^G - X_f)\hat{i} + (COM_{y_i}^G - Y_f)\hat{j} + (COM_{z_i}^G - Z_f)\hat{k}\} \times \{W_{femur}\hat{j}\} \end{aligned} \quad (57)$$

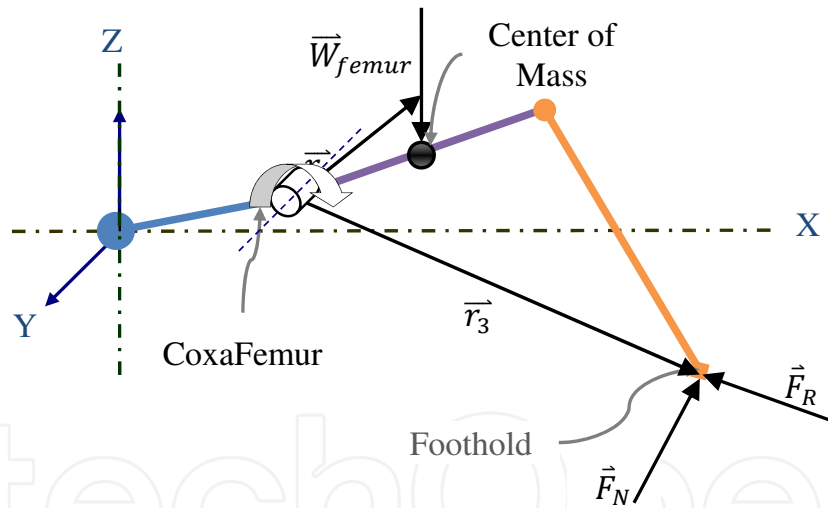


Figure 10. The total torque about CoxaFemur joint axis is equal to the sum of moments due to weight of the femur link and moments due to friction force and normal reaction forces acting at the foothold.

4.3.1.3. Reaction torque at femur tibia joint

Considering Fig. 11, the reaction torque about the FemurTibia joint motion axis can be expressed by (58), further explained in (59) & (60).

$$\vec{\tau}_{FemurTibia} = \{\vec{r}_5 \times \vec{F}_N + \vec{r}_5 \times \vec{F}_R + \vec{r}_6 \times \vec{W}_{tibia}\}Nm \quad (58)$$

$$\left\{ \begin{array}{l}
 \vec{r}_5 = \{(Ft_{x_i}^G - X_t)\hat{i} + (Ft_{y_i}^G - Y_t)\hat{j} + (Ft_{z_i}^G - Z_t)\hat{k}\} m \\
 \vec{r}_6 = \{(COM_{x_i}^G - X_t)\hat{i} + (COM_{y_i}^G - Y_t)\hat{j} + (COM_{z_i}^G - Z_t)\hat{k}\} m \\
 \vec{F}_N = \text{Normal reaction force vector at the foothold} \\
 \vec{F}_R = \text{Friction force vector at the foothold} \\
 \vec{W}_{tibia} = \text{Weight of the tibia link} \\
 \begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} \cos\theta_c (lc + lt \cos\theta_{f+t} + lf \cos\theta_f) \\ \sin\theta_c (lc + lt \cos\theta_{f+t} + lf \cos\theta_f) \\ lt \sin\theta_{f+t} + lf \sin\theta_f \end{bmatrix} = \text{Position of FemurTibia joint axis} \\
 \vec{\tau}_{FemurTibia} = \{(Ft_{x_i}^G - X_t)\hat{i} + (Ft_{y_i}^G - Y_t)\hat{j} + (Ft_{z_i}^G - Z_t)\hat{k}\} \times \{F_{Nx}\hat{i} + F_{Ny}\hat{j} + F_{Nz}\hat{k}\} + \\
 \{(Ft_{x_i}^G - X_t)\hat{i} + (Ft_{y_i}^G - Y_t)\hat{j} + (Ft_{z_i}^G - Z_t)\hat{k}\} \times \{F_{Rx}\hat{i} + F_{Ry}\hat{j} + F_{Rz}\hat{k}\} + \\
 \{(COM_{x_i}^G - X_t)\hat{i} + (COM_{y_i}^G - Y_t)\hat{j} + (COM_{z_i}^G - Z_t)\hat{k}\} \times \{W_{tibia}\hat{j}\}
 \end{array} \right. \quad (59)$$

$$\quad (60)$$

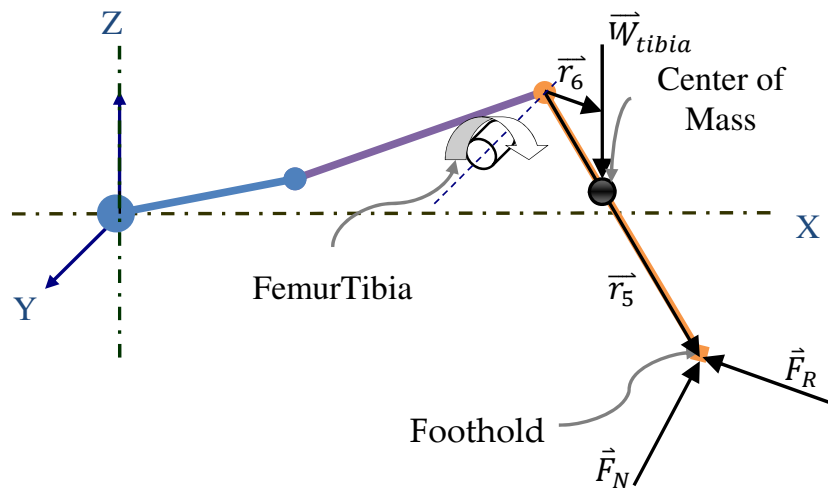


Figure 11. The total torque about FemurTibia joint axis is equal to the sum of moments due to weight of the tibia link and moments due to friction force and normal reaction forces acting at the foothold.

4.3.2. Gait analysis over an uneven terrain

The uneven terrain as shown in Fig. 13 is modelled in Microsoft Robotics Developer Studio. A terrain is modelled with random regions of depression and elevation so that the gait can be analyzed under the influence of unexpected disturbances. The robot is steered between defined waypoints and the joints torques are investigated with different gait patterns. Fig. 12(A) shows a comparative representation of torques computed by our mathematical model and those obtained through joint sensors provided by the simulation engine. The results shown in Fig. 12 are those obtained when the robot was steered with a fast-speed ripple gait. As evident from the forces plot shown in Fig. 12(B), friction force is close to the normal reaction point at some points and exceeds the normal reaction force at some other points which signifies that the robot experienced significant degrees of slip/drift which walking

through those regions of the terrain. As a consequence, the mathematical model failed to determine adequate torques at the joints to counter the drift each foothold is experiencing while the locomotion. Therefore, the fast-speed ripple gait was found inefficient over an unexpected terrain due to the non-uniform distribution of payload about the two sides of the robot body in its support pattern.

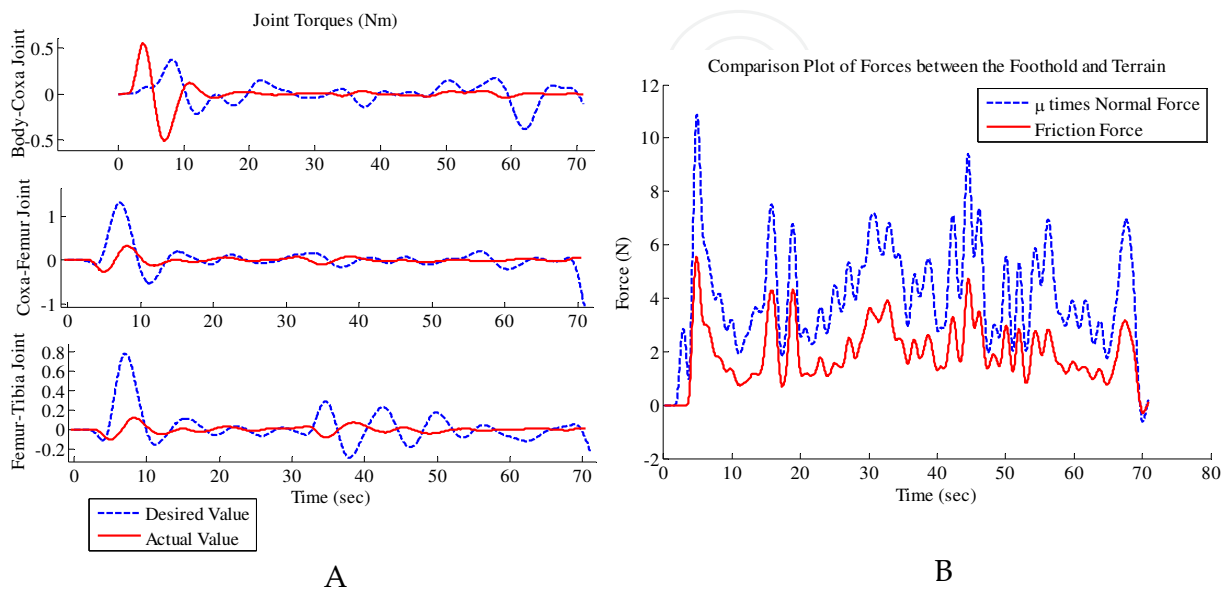


Figure 12. Results obtained when the robot was steered using a ripple gait. (A) Comparative representation of actual torques sensed at the joint and the torques data determined by the mathematical model. (B) Plot of normal reaction forces and friction forces at the footholds.

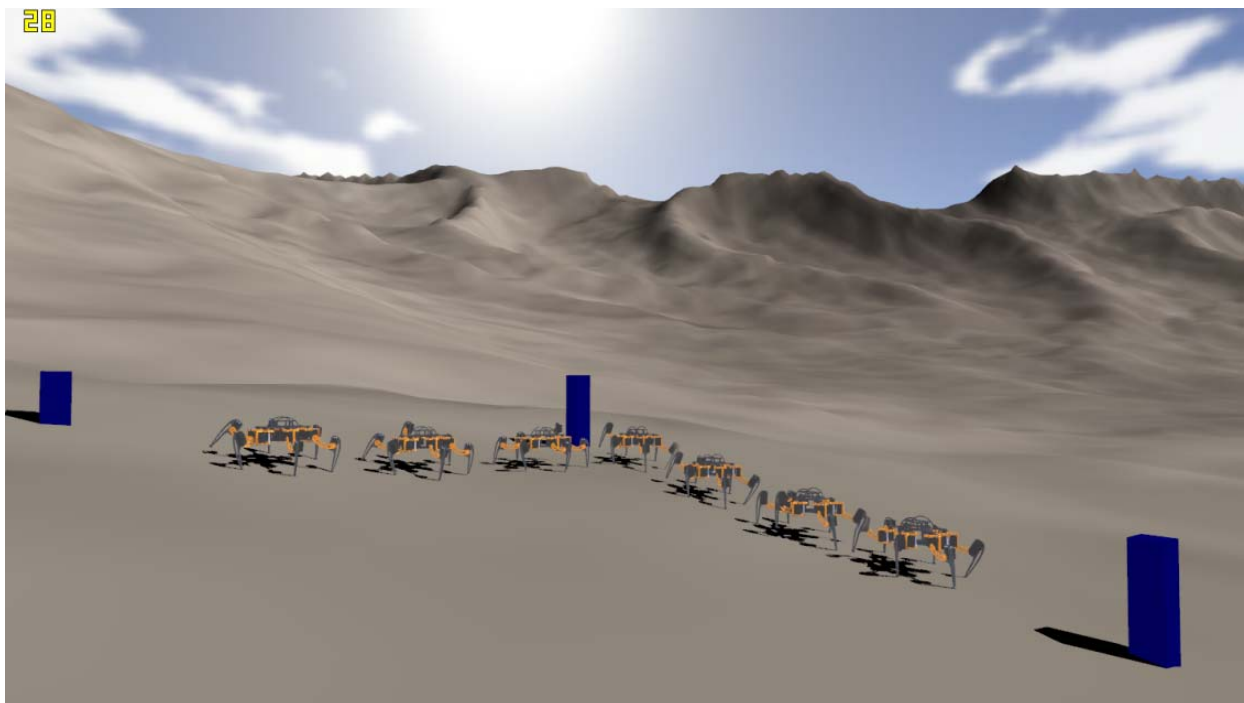


Figure 13. A view of the robot straight-line walking between desired waypoints over an unknown irregular terrain.

In the second run, the robot was steered with a slow-speed wave gait in which each leg is lifted whereas the remaining supports the body. This confirms an even sharing of payload on either side of the robot. The results of this simulation run are shown in Fig. 14 and Fig. 15. As evident from the Fig. 15(B) that the friction force data never exceeds the normal reaction force which confirms that the robot never experienced a slip/drift over the same track when using the wave gait. As a consequence, close matching of computed torques and the sensed torques during this simulation run as shown by Fig. 15(A) signifies the validity of our proposed mathematical model to realize adaptive walking over uneven terrains at slow speeds. Additional simulation results are exhibited in Fig. 14 which portrays the track of the centre of gravity of the robot and its speed profile. Fig. 15(C) & Fig. 15(D) show the profiles of body rotation angles and rates about x, y & z-axes. As evident from the Fig. 15(C), the roll and pitch rotations of the body are minimal which confirms the validity of our attitude control for maintaining the body flat over an unknown irregular terrain.

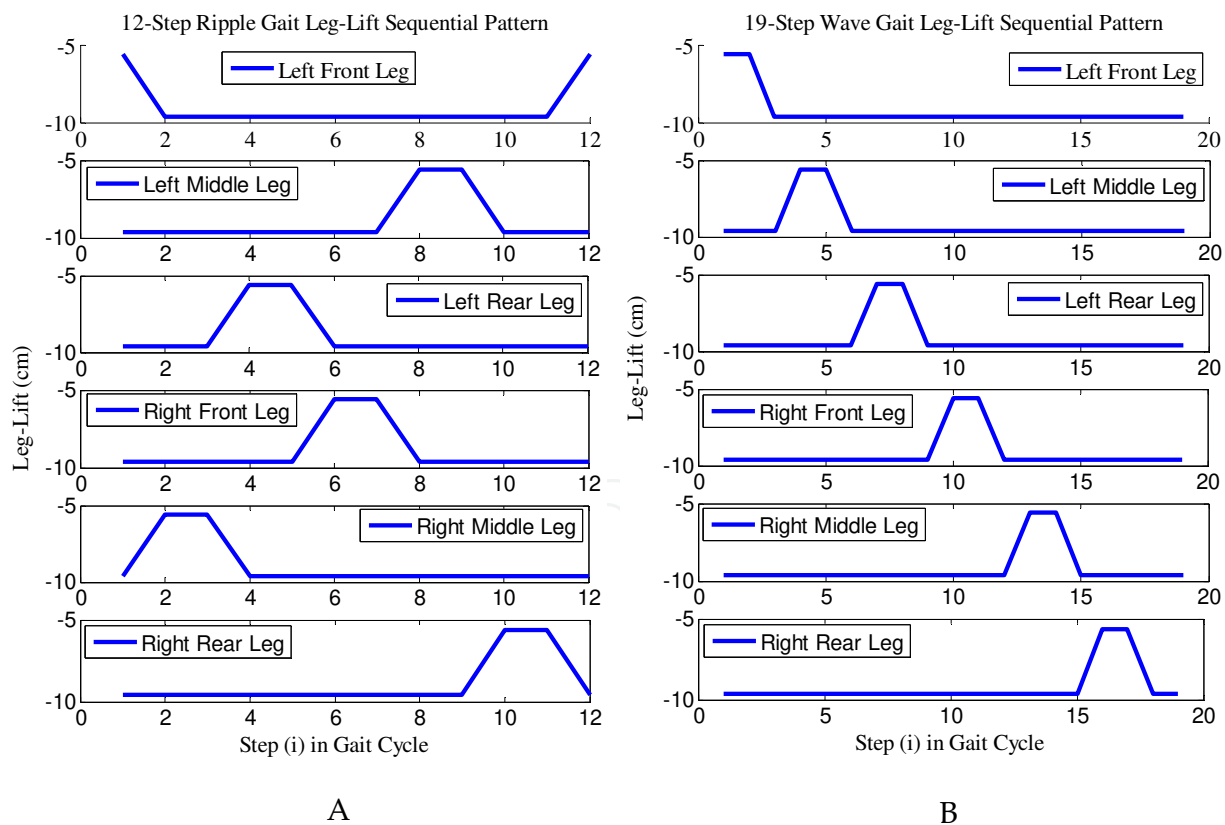


Figure 14. (A) 12-step ripple gait leg-lift sequential pattern. (B) 19-step wave gait leg-lift sequential pattern.

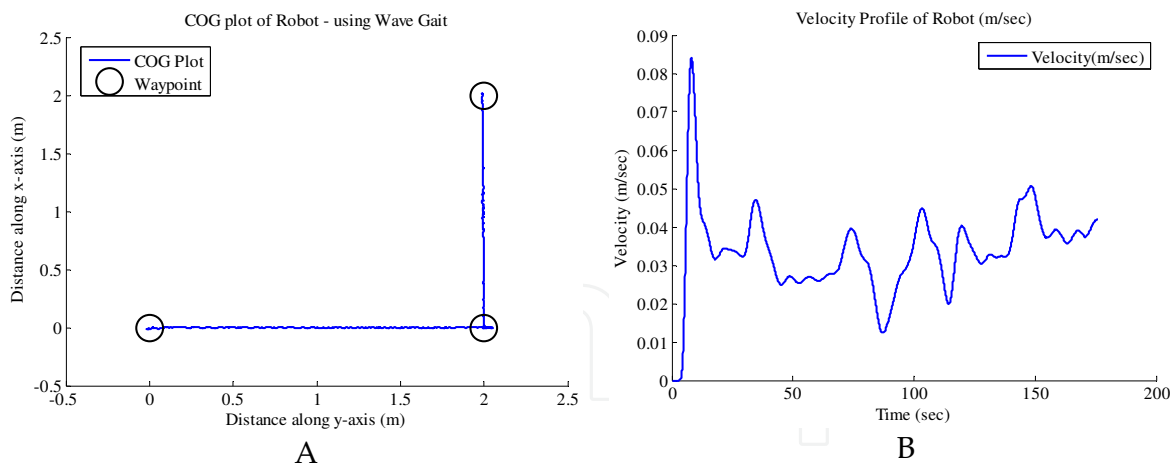


Figure 15. (A) Path outlined by the robot when using a wave gait. (B) Velocity profile of the robot during its locomotion.

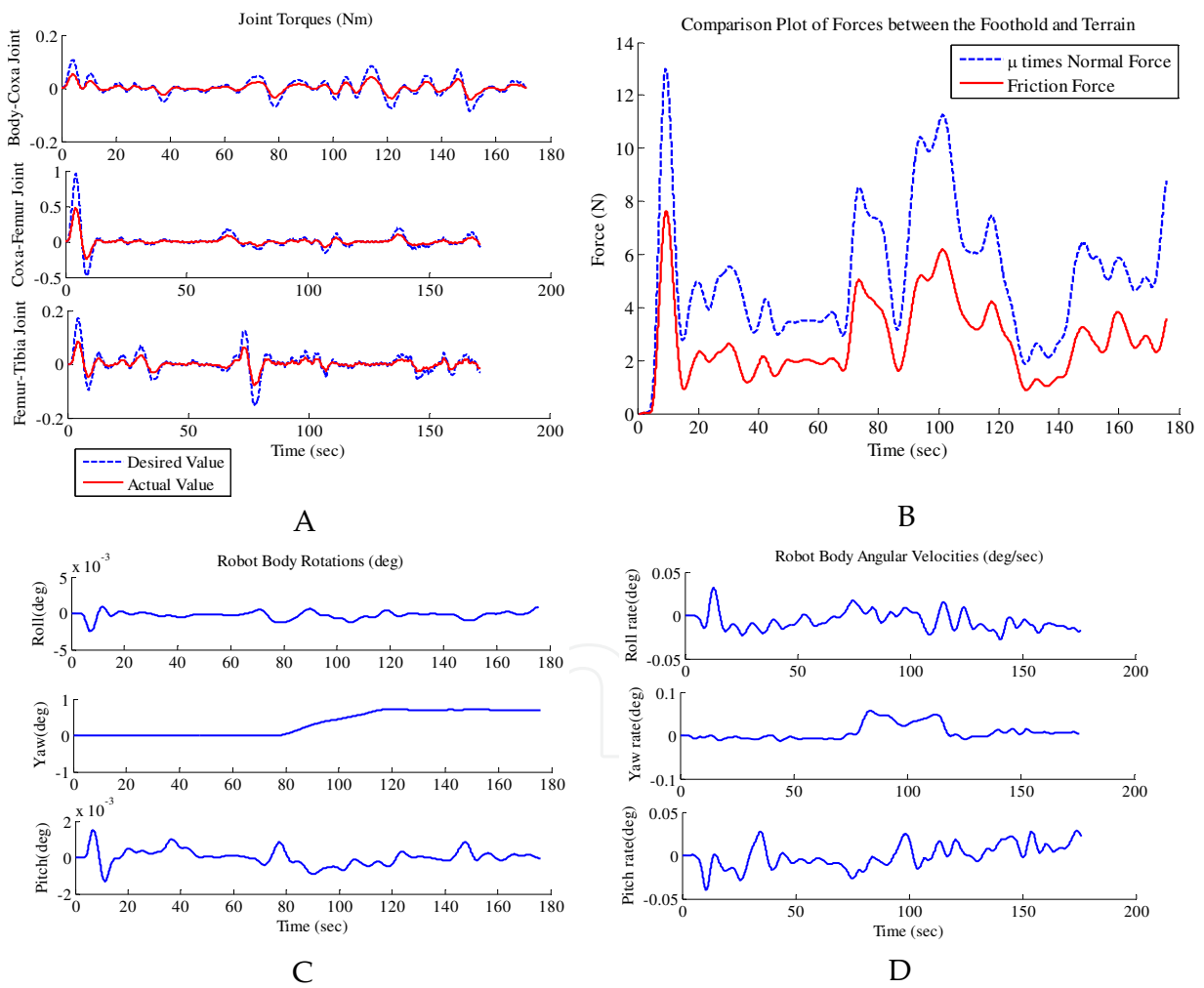


Figure 16. Results obtained when the robot was steered using a wave gait. (A) Comparative representation of actual torques sensed at the joint and the torques data determined by the mathematical model. (B) Plot of normal reaction forces and friction forces at the footholds. (C) Robot attitude tracking. (D) Angular velocity profiles of the robot's body.

5. Conclusions

This chapter outlines a study on signifying the importance of employing virtual reality technology in realizing legged locomotion over unknown irregular terrains. The objective is to emphasize that how virtual reality technology can serve a prime role in the improvement of legged locomotion by performing gait analysis over unstructured terrains which is often time intense and complex to conduct in real-worlds. The objective has been achieved by analyzing an adaptive gait of a six-legged hexapod robot with two unique support patterns namely: ripple and wave. A complete mathematical model of the robot is presented at the start of the chapter to equip the reader with the basic knowledge of forward and reverse kinematic equations used in robot kinematics. Robot dynamics is further described with a computed-torque control technique to incorporate the forces and moments at the feet. To validate the proposed mathematical model, simulation tests are first carried out in a constraint simulation environment using SimMechanics and results of mathematical model are compared with those obtained from on-board sensors. For gait analysis, the robot is steered over an unknown irregular terrain first using a fast-speed ripple gait and then using a slow-speed wave gait. Reaction forces at the footholds and the reaction torques about the joint motion axes in a leg are investigated which reveal that the wave gait is slow but more secure for adaptive locomotion over an unknown irregular terrain. This even sharing of active payload on either side of the robot body in wave gait results in greater stability especially in the regions of possible slip/drift. However, when using a fast speed ripple gait in which the force-distribution on either side of the robot body is not uniform, the support pattern fails to provide enough friction force against the normal reaction force that may prevent the slip/drift. As a consequence, ripple gaits provide high mobility and are effective over flat terrains however, wave gaits are more suited to walking over unknown irregular terrains.

Author details

Umar Asif

School of Mechanical & Manufacturing Engineering, National University of Sciences & Technology (NUST), Pakistan

Acknowledgement

The author would like to thank the editor for his detailed and pertinent comments.

6. References

- [1] U. Asif, J. Iqbal, "Motion Planning using an Impact-based Hybrid Control for Trajectory Generation in Adaptive Walking", *International Journal of Advanced Robotic Systems*, Vol. 8, No. 4, pp. 212-226, 2011.

- [2] Wood, R.J., "The First Take-off of a Biologically Inspired At-Scale Robotic Insect", in *IEEE Transactions on Robotics*, vol.24, no.2, pp. 341-347, April 2008.
- [3] Kuhn, D. and Rommermann, M. and Sauthoff, N. and Grimminger, F. and Kirchner, F., "Concept evaluation of a new biologically inspired robot LittleApe", in *IEEE/RSJ International Conference on Intelligent Robots and Systems 2009 (IROS 09)*, pp. 589-594, 10-15 October 2009.
- [4] Shumei Yu and Shugen Ma and Bin Li and Yuechao Wang, "An amphibious snake-like robot: Design and motion experiments on ground and in water", in *International Conference on Information and Automation 2009 (ICIA 09)*, pp. 500-505, 22-24 June 2009.
- [5] M. Konyev and F.Palis and Y. Zavgorodniy and A. Melnikov and A. Rudiskiy and A. Telesh and U. Sschmucker and V. Rusin, "Walking Robot Anton: Design, Simulation, Experiments", in *International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines 2008 (CLAWAR 2008)*, pp. 922-929, 8-10 September 2008.
- [6] A. Roennau, T. Kerscher and R. Dillmann, "Design and Kinematics of a Biologically-Inspired Leg for a Six-Legged Walking Machine", in *the Proc. of the 2010 3rd IEEE RAS & EMBS Int. Conf. on Biomedical Robotics and Biomechatronics*, The University of Tokyo, Tokyo, Japan, September 26-29, 2010.
- [7] Willian A. Lewinger and Roger D. Quinn, "A Hexapod Walks Over Irregular Terrain Using a Controller Adapted from an Insect's Nervous System", *The 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, October 18-22, 2010, Taipei, Taiwan.
- [8] U. Asif and J. Iqbal, Rapid Prototyping of a Gait Generation Method using Real Time Hardware in Loop Simulation, "*International Journal of Modeling, Simulation and Scientific Computing (IJMSSC)*", vol. 2, no. 4, pp. 393-411, 2011.
- [9] U. Asif and J. Iqbal, Design and Simulation of a Biologically Inspired Hexapod Robot using SimMechanics, "*Proceedings of the IASTED International Conference, Robotics (Robo 2010)*", pp. 128-135, Phuket, Thailand, November 24 - 26, 2010.
- [10] U. Asif and J. Iqbal, A Comparative Study of Biologically Inspired Walking Gaits through Waypoint Navigation, "*Advances in Mechanical Engineering*", vol. 2011 (2011), Article ID 737403, 9 pages, DOI: 10.1155/2011/737403.
- [11] U. Asif and J. Iqbal, "An Approach to Stable Walking over Uneven Terrain using a Reflex Based Adaptive Gait", *Journal of Control Science and Engineering*, Volume 2011 (2011), Article ID 783741, 12 pages, DOI:10.1155/2011/783741.
- [12] U. Asif, J. Iqbal and M. Ajmal Khan, "Kinematic Analysis of Periodic Continuous Gaits for a Bio-Mimetic Walking Robot", *Proceedings of the 2011 IEEE International Symposium on Safety, Security and Rescue Robotics*, pp. 80-85, November 1-5, 2011, Kyoto, Japan.
- [13] U. Asif and J. Iqbal, "Motion Planning of a Walking Robot using Attitude Guidance", *International Journal of Robotics and Automation*, pp. 41-48, vol. 27, no. 1, 2012. DOI: 10.2316/Journal.206.2012.1.206-3589.
- [14] U. Asif and J. Iqbal, "On the Improvement of Legged Locomotion in Difficult Terrains using a Balance Stabilization Method", *International Journal of Advanced Robotic Systems*, Vol. 9, No. 1, pp. 1-13, 2012.
- [15] Kyle Johns, Trevor Taylor, *Professional Microsoft Robotics Developer Studio* (Wiley publishing, Indianapolis, Indiana, 2008).

- [16] U. Asif and J. Iqbal, Modeling, Control and Simulation of a Six-DOF motion platform for a Flight Simulator, "International Journal of Modeling and Simulation", vol. 31, no. 4, pp. 307-321, 2011. DOI:10.2316/Journal.20.2011.4.205-5587.
- [17] U. Asif and J. Iqbal, Modeling, Simulation and Motion Cues Visualization of a Six-DOF Motion Platform for Micro-Manipulations, "International Journal of Intelligent Mechatronics and Robotics (IJIMR)", vol. 1, no. 3, pp. 1-17, 2011.
- [18] U. Asif and J. Iqbal, "A Robotic System with a Hybrid Motion Cueing Controller for Inertia Tensor Approximation in Micro-Manipulations", International Journal of Advanced Robotic Systems, Vol. 8, No. 4, pp. 235-247, 2011.

IntechOpen