JEAS

JOURNAL OF ENGINEERING
AND
APPLIED SCIENCES

Journal of Engineering and Applied Sciences, Volume 19, Number 1, December 2021, 597 - 614

Position Modelling of a Palm Tree Climbing Robot (Pamtreebot)

I. K Okonkwo Jr. and S. C. Nwigbo
[1]Department of Mechanical Engineering, Nnamdi Azikiwe University, Awka

*CorrespondingAuthor's E-mail: 27eleven1@gmail.com

Abstract
The end-effecter positions of a palm tree climbing robot have been modelled and tracked in this paper. In the introduction section, the significance of robots in specialized tasks, the necessity of the proposed pamtreebot for safer palm tree farming, the concept of position modelling and a review of previous researches were outlined. Next, the mechanical framework of the pamtreebot was presented and illustrated with a CAD drawing. To model the various positions of its limbs as it climbs, the forward kinematics equations for each limb posture were derived using the Denavit-Hartenberg convention and the required joint angles for each posture were also derived analytically from the forward kinematics equations hence the inverse kinematics. Finally, a path trajectory function is developed using 'point-to-point motion technique' and substituted into the inverse kinematics models to trace the time histories of the joint angles. The inverse kinematics results were compared to results from RoboAnalyzer software and they were similar.

Keywords: Treebot, kinematics, link, joint angle, degree of freedom, planar kinematic chain, revolute joint, end-effecter.

1. Introduction

Unsafe, hazardous, highly repetitive and unpleasant tasks which humans prefer not to do have been reserved for robots. They are increasingly being used to operate in environments which are unhealthy or impractical for workers and even in extreme environments where humans cannot survive such as outer space or at the bottom of the sea. Robots can be programmed and expected to obey and perform tasks with great speed and accuracy without getting tired or facing the commands emotionally. In palm tree farming, climbing the tree is mostly prerequisite for tree maintenance, palm fruit harvesting and palm wine tapping. Palm trees stand 32-50ft tall; to reach the upper part of a palm tree, the farmer relies on a tough raffia belt to secure and support him as he climbs. Palm produce play a huge role in the country's economy, however, climbing a palm tree is too risky and not worth the life of a human being. Statistical figures have shown significant level of fatalities. The methods and tools mostly used are crude, non-ergonomic, inefficient and yet very unsafe. The proposed palm tree climbing robot rather known as 'pamtreebot' is the resolution in the quest to relieve and secure the farmer. This research is a fundamental step towards prevention of further casualties that could result from climbing of palm trees.

Our focus in this paper is to describe or represent mathematically, the proposed end-effecter/joint angles positions of a palm tree climbing robot as it climbs, hence the position modelling. Two kinematic problems are associated with robot arm motion; the forward kinematics problem and the inverse kinematics problem. The forward kinematics problem is to determine the position and orientation of the end-effecter, from given values of joint parameters. The forward kinematics equations can give the location of the tip of the arm using the angles at the joints because the overall motion of robot arm is achieved by combined motions at its joints causing the motion of each link in respect to the previous as one body in space. Inverse kinematics refers to the reverse process of forward kinematics. In contrast to forward kinematics, it is concerned with determining possible values of joint parameters which fit a desired location (position and orientation) of the end-effecter. That is, given a desired location for the tip of the arm's end-effecter, what should the angles of the joints be so as to place the tip of the arm at the desired location? There is usually more than one solution. Inverse kinematics can be solved either analytically from the forward kinematics equations or derived geometrically from the configuration of the arm. Once the joint angles are known, the desired end-effecter motion can be practically achieved by the moving each joint through the required angular displacement. Asides forward and inverse kinematics, trajectory tracking is one of the main features of robot arm position modelling. Trajectory is a path on which a timing law is specified. Trajectory planning algorithm generates a time sequence of variables that describe end-effector/joint position over time in respect of the imposed constraints. Robot manipulators move along pre-specified trajectories which are sequence of points where end effector position, and orientations are known. Trajectories may be joint space or Cartesian spaces that are a function of time (Panchanand, 2015).

Robot arm modelling and trajectory tracking have received much attention from researchers in the past decades and in recent times. Sajjad et al (2018) designed and implemented a SCARA PRR manipulator. The forward and inverse kinematic equations of the robot were derived using D-H parameters and analytical methods respectively. The trajectories were planned using the calculated kinematic equations and the simulation was done in MATLAB VRML environment. Khin et al (2016) designed a robot arm comprising two joints, three links and servo motors to drive. The position control of robot arm was designed by using kinematic control methods. Subedi et al (2021) presented a closed-form dynamic model a planar multi-link flexible manipulator is presented using the assumed modes method with the Lagrangian formulation to obtain the dynamic equations of motion. Hossein and Hassan (2014) modeled, simulated and controlled a 3-DOF (degrees of freedom) articulated robot manipulator by extracting the kinematic and dynamic equations using Lagrange method and compared the derived analytical model with a simulated model using SimMechanics toolbox. The model is further linearized with feedback and a PID controller is implemented to track a reference trajectory. Gaoyuan et al (2020) focused on the design of badminton robots, and the designs high-precision binocular stereo vision hardware. The three-dimensional trajectory points are extended by Kalman filter and the kinematics equation of badminton is established. The parameters of the kinematics equation of badminton are solved by the method of least squares. Farhan (2014) presented for educational purpose, a robot arm model based on Simulink. Mathematical, Simulink models and MATLAB program were developed to return maximum numerical visual and graphical data to select, design, control and analyze arm system. Dubey et al (2016) designed the kinematics of the robot by referring to the motion of coconut harvester. In (Okubanjo et al, 2017), the control algorithm is expanded on the derived mathematical equations to control the robot arm in joint angle position. Mahil and Al-durra, (2016) presented a linearized mathematical model of 2-DOF robotic manipulator and derived a mathematical model based on kinematic and dynamic equations using the combination of Denavit Hartenberg and Lagrangian methods.

Sane Subhash (2015) considered a two link planar manipulator revolving in a horizontal plane. Its kinematics was modeled and it was proposed to follow a desired trajectory by using an effective control method. Amarendra et al (2018) developed a mathematical model to determine the position of mobile robot by sensing the angular velocity and heading angle of the caster wheel. Using the established equations, simulations were carried out using MATLAB version 8.6 to observe and verify the position coordinates of mobile robot and in turn obtain its trajectory. Chaitanyaa and Sreenivasulua (2016) optimized a two link revolute robotic arm for maximization of work space area covered by its end effecter. A mathematical model for optimization was built considering singularities which influence the variation of design variables. Ali et al (2019) presented the development and evaluation of a 4-DOF articulated robotic arm as an actuating unit of a robot tractor for the agricultural outdoor environment. The controlling algorithm of the system was developed by consideration of kinematic and dynamic aspects of the real-world condition. Mohammed and Sunar (2015) derived the kinematics model of a 4-DOF robot arm using both Denavit-Hartenberg (DH) method and product of exponential formula, which is based on the screw theory. Panchanand (2015) introduced inverse kinematics solutions through conventional methods with an efficient study of the existing tools and techniques focusing on various industrial manipulators at different configurations. In (Virgala 2014), the paper deals with analyzing, modeling and simulation of motion of a humanoid robot hand with 24-DOF. A new method for the inverse kinematic model was introduced using Matlab functions.Shekhar and Jayswa (2016) tracked the point to point trajectory by cubic polynomial from initial to intermediate point as well as from intermediate to final point is used. To avoid the singular configuration, direct kinematics was used. Mahmoud (2014) investigated several control strategies to handle the trajectory tracking problem for a 2-DOF robotic arm using artificial neural networks (ANNs). Mathematical models for the 2-DOF robot arm and its joints driving motors were developed and simulation experiments were carried out under the Dynamic Modeling Laboratory (Dymola) environment which uses the Modelica object-oriented multi-domain system modeling language.

Robots can perform arbitrary sequences of pre-stored motions computed as functions of sensory input. The position models developed in this paper are significant in the development of computer control programmes for the proposed pamtreebot –a discrete model with an optimal non-slip climbing strategy.

2.0 Material and Methods

2.1 Design Presentation of the Pamtreebot

The design of the pamtreebot (see fig. 1) comprises a trunk, two upper limbs – the Upper Right Limb (URL) and the Upper Left Limb (ULL) and two lower limbs – the Lower Right Limb (LRL) and the Lower Left Limb (LLL). The upper limb configurations are three revolute joint (RRR) planar arms while the lower limb configuration are two revolute joint (RR) planar arm. Thus the upper limbs allow 3 independent movements while the lower limbs allow 2 independent displacements or aspects of motion. This means that the upper arm have 3-DOF while the lower limbs have 2-DOF. The links connected by revolute joints – A, B, C, D and E form planar kinematic open chains. The end-effecter links are the last links on the chain with pointed tips for hooking on to the tree trunk as it climbs i.e. the end-effecter links 3 (EL3) are the end-effecters of the upper limbs while the end-effecter links 2 (EL2) are the end-effecters on the lower limbs.

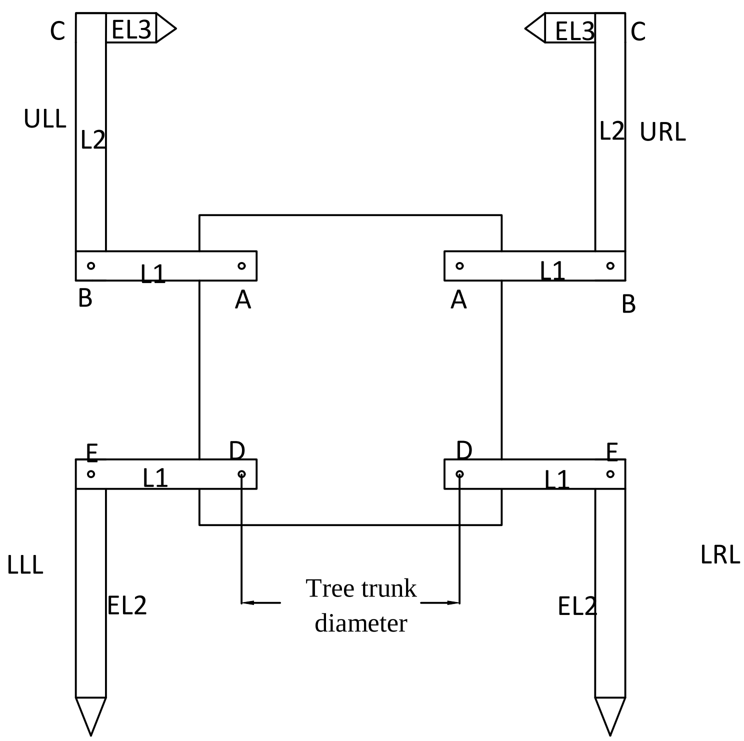

Fig. 1: AutoCAD Drawing of the Pamtreebot

2.2 Modelling the Kinematics

We shall consider the limbs on the right; their results can be mirrored for the left limbs since the motions are identical. To climb, the limbs take up three major postures -P1, P2 and P3.

2.2.1 The Lower Right Limb at P1

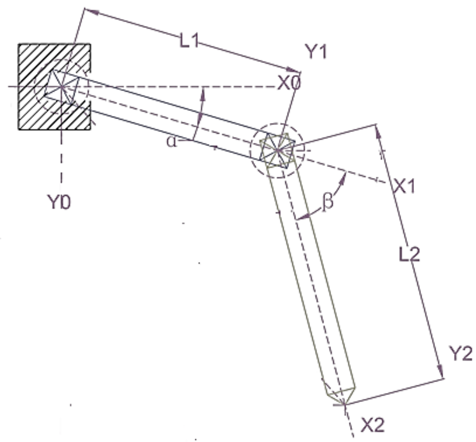

Fig 2: Lower Right Limb at Posture 1

The hip joint is taken as the origin (0,0) for the lower limbs.

α is the hip joint angle. For the LRL, it is the angular displacement of the link 1 from the positive horizontal axis while for the LLL, it is the angular displacement of the link 1 from the negative horizontal axis.

β is the knee joint angle. It is the angular displacement of the end-effecter (link 2) from the axis of the link 1.

Forward Kinematics:
To derive (or compute) the end-effecter position, we can simply resolve the link lengths with respect to the joint angles but rather we utilize a lengthy but accurate common systematic convention for selecting reference frames known as the Denavit-Hartenberg convention where a homogenous transformation matrix from the base frame to the uppermost frame is derived so as to express the position and orientation of one frame with respect to another. The frame setup is as follows:
Joint axes - z_0 and z_1 are normal to the page
Base frame - $o_0x_0y_0z_0$, where the origin is chosen at the point of intersection of the z_0 axis with the page and the direction of the x_0 axis is completely arbitrary.
First frame - $o_1x_1y_1z_1$, where the origin o_1 has been located at the intersection of z_1 and the page.
Second frame (end-effecter frame) - $o_2x_2y_2z_2$, where the origin o_2 has been located at the intersection of z_2 and the page.

Table 1: Link Parameters for the Lower Right Limb at Posture 1

Link	**Link length** $\boldsymbol{l_i}$	**Link Twist** $\boldsymbol{t_i}$	**Link Offset** $\boldsymbol{h_i}$	**Joint Angle** $\boldsymbol{\theta_i}$
1	l_1	0	0	$-\alpha$
2	l_2	0	0	$-\beta$

The transformation matrix T_2^0, a product of the homogenous matrices will give us the position and orientation of the end-effecter expressed in base coordinates.

Each homogenous matrix A_i^0 is given by

$$A_i^0 = \begin{bmatrix} c_{\theta_i} & -s_{\theta_i}c_{t_i} & s_{\theta_i}s_{t_i} & l_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i}c_{t_i} & -c_{\theta_i}s_{t_i} & l_i s_{\theta_i} \\ 0 & s_{t_i} & c_{t_i} & h_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

For the joints 1 and 2,

$$A_1^0 = \begin{bmatrix} c_\alpha & s_\alpha & 0 & l_1 c_\alpha \\ -s_\alpha & c_\alpha & 0 & -l_1 s_\alpha \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad A_2^0 = \begin{bmatrix} c_\beta & s_\beta & 0 & l_2 c_\beta \\ -s_\beta & c_\beta & 0 & -l_2 s_\beta \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Now the transformation matrix T_2^0,

$$T_2^0 = A_1^0 A_2^0$$

$$= \begin{bmatrix} c_{\alpha+\beta} & s_{\alpha+\beta} & 0 & l_1 c_\alpha + l_2 c_{\alpha+\beta} \\ -s_{\alpha+\beta} & c_{\alpha+\beta} & 0 & -l_1 s_\alpha - l_2 s_{\alpha+\beta} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

The first three elements of the fourth column of the transformation matrix T_2^0 represent the position coordinates (x_e, y_e, z_e) of the tip of the end-effecter with origin o_3 of frame 3 with respect to the base frame 0 while the rotational part of T_2^0 (upper left 3×3 rotation matrix) gives the orientation of the frame $o_2x_2y_2z_2$ with respect to the base frame.

Inverse Kinematics:
To determine the joint angles required for this posture, we can solve the forward kinematics equations for α and β. The forward kinematics equations are:

$$x_e = l_1 c_\alpha + l_2 c_{\alpha+\beta} \tag{2}$$

$$y_e = -l_1 s_\alpha - l_2 s_{\alpha+\beta} \tag{3}$$

x_e and y_e are respectively the abscissa and the ordinate of the end-effecter's tip from the origin (hip joint).

Squaring and summing the simultaneous equations (2) and (3)

$$x_e{}^2 + y_e{}^2 = l_1{}^2 + 2l_1 l_2 c_\beta + l_2{}^2 \tag{4}$$

Solving for c_β,

$$c_\beta = \frac{x_e{}^2 + y_e{}^2 - l_1{}^2 - l_2{}^2}{2l_1 l_2} \tag{5}$$

For more accuracy in our β results, let us avoid arc cosine or arc sine. We can substitute c_β into the half-angle identity $\tan(\beta/2) = \sqrt{\frac{1-c_\beta}{1+c_\beta}}$ to obtain β with arc tangent as

$$\beta = 2\, atan\left[\frac{(l_1 + l_2)^2 - (x_e{}^2 + y_e{}^2)}{x_e{}^2 + y_e{}^2 - (l_1 - l_2)^2}\right]^{1/2} \tag{6}$$

The arc tangent of two functions may be written as '$atan2$', thus we may write our '$2\ atan$' of functions x_e and y_e as '$2\ atan2$'.

Now to solve for α, we can rewrite the forward kinematics equations as

$$x_e = l_1 c_\alpha + l_2 c_\alpha c_\beta - l_2 s_\alpha s_\beta \tag{7}$$

$$y_e = -l_1 s_\alpha - l_2 s_\alpha c_\beta - l_2 c_\alpha s_\beta \tag{8}$$

Solving eq.s (7) and (8) for c_α,

$$c_\alpha = \frac{x_e\left(l_1 + l_2 c_\beta\right) - y_e l_2 s_\beta}{x_e{}^2 + y_e{}^2} \tag{9}$$

or solving eq.s (7) and (8) for s_α,

$$s_\alpha = -\left[\frac{y_e\left(l_1 + l_2 c_\beta\right) + x_e l_2 s_\beta}{y_e{}^2 + x_e{}^2}\right] \tag{10}$$

Still avoiding arc cosine and arc sine, we can substitute c_α and s_α – eq.s (9) and (10) into the identity $\tan\alpha = s_\alpha / c_\alpha$ to obtain α with arc tangent as

$$\boldsymbol{\alpha} = atan\left[\frac{-y_e\left(l_1 + l_2 c_\beta\right) - x_e l_2 s_\beta}{x_e\left(l_1 + l_2 c_\beta\right) - y_e l_2 s_\beta}\right] \tag{11}$$

2.2.2 The Lower Right Limb at P2

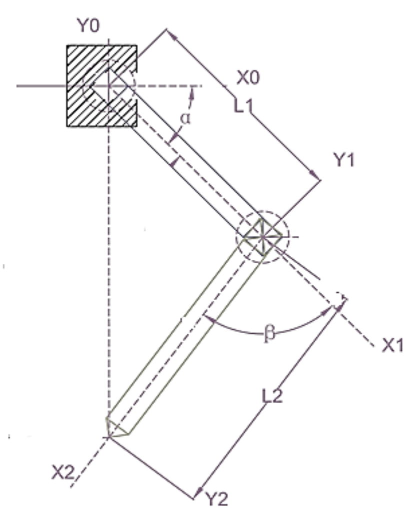

Fig. 3: Lower Right Limb at Posture 2

Forward and Inverse Kinematics:
The link parameters for this posture are same as that of P1 so all its forward and inverse kinematics equations will also be same (but clearly, in this configuration, $x_e = 0$).

2.2.3 The Lower Right Limb at P3

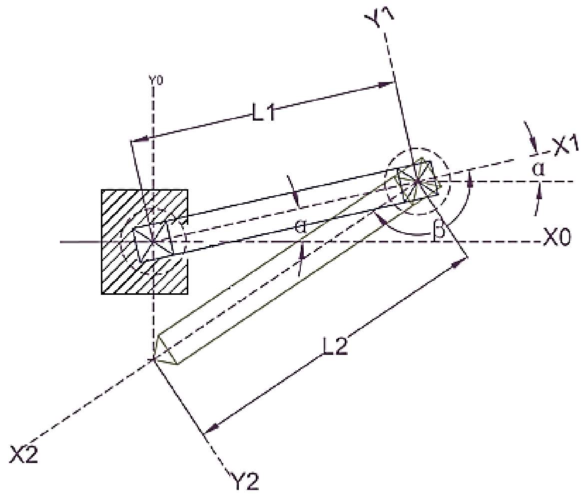

Fig. 4: Lower Right Limb at Posture 3

Forward Kinematics:

Table 2: Link Parameters for the Lower Right Limb at Posture 3

Link	Link length l_i	Link Twist t_i	Link Offset h_i	Joint Angle θ_i
1	l_1	0	0	A
2	l_2	0	0	−β

$$A_1^0 = \begin{bmatrix} c_\alpha & -s_\alpha & 0 & l_1 c_\alpha \\ s_\alpha & c_\alpha & 0 & l_1 s_\alpha \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad , \quad A_2^0 = \begin{bmatrix} c_\beta & s_\beta & 0 & l_2 c_\beta \\ -s_\beta & c_\beta & 0 & -l_2 s_\beta \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_2^0 = \begin{bmatrix} c_{\alpha-\beta} & -s_{\alpha-\beta} & 0 & l_1 c_\alpha + l_2 c_{\alpha-\beta} \\ s_{\alpha-\beta} & c_{\alpha-\beta} & 0 & l_1 s_\alpha + l_2 s_{\alpha-\beta} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(12)

Inverse Kinematics:
Solving for the joint angles α and β as before, we obtain the same inverse kinematics models as in P1. Also in this configuration, $x_e = 0$.

2.2.4 The Upper Right Limb at P1

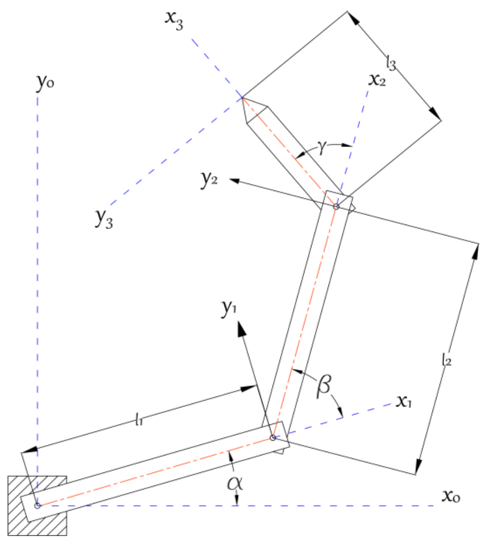

Fig. 5: Upper Right Limb at Posture 1

α is the shoulder joint angle. For the URL, it is the angular displacement of the link 1 from the positive horizontal axis while for the ULL, it is the angular displacement of the link 1 from the negative horizontal axis.

β is the elbow joint angle (angular displacement of the link 2 from the axis of the link 1)

γ is the wrist joint angle (angular displacement of the end-effecter from the axis of link 2)

Forward Kinematics:
Frame Setup:
Joint axes - z_0, z_1 and z_2 are normal to the page
Base frame - $o_0x_0y_0z_0$, where the origin is chosen at the point of intersection of the z_0 axis with the page and the direction of the x_0 axis is completely arbitrary.
First frame - $o_1x_1y_1z_1$, where the origin o_1 has been located at the intersection of z_1 and the page.
Second frame - $o_2x_2y_2z_2$, where the origin o_2 has been located at the intersection of z_2 and the page.
The end-effecter frame $o_3x_3y_3z_3$ is fixed by choosing the origin o_3 at the end of link 3 as shown.

Table 3: Link Parameters for the Upper Right Limb at Posture 1

Link	Link length l_i	Link Twist t_i	Link Offset h_i	Joint Angle θ_i
1	l_1	0	0	α
2	l_2	0	0	β
3	l_3	0	0	γ

For the joints 1, 2 and 3,

$$A_1^0 = \begin{bmatrix} c_\alpha & -s_\alpha & 0 & l_1 c_\alpha \\ s_\alpha & c_\alpha & 0 & l_1 s_\alpha \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad A_2^0 = \begin{bmatrix} c_\beta & -s_\beta & 0 & l_2 c_\beta \\ s_\beta & c_\beta & 0 & l_2 s_\beta \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad A_3^0 = \begin{bmatrix} c_\gamma & -s_\gamma & 0 & l_3 c_\gamma \\ s_\gamma & c_\gamma & 0 & l_3 s_\gamma \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The transformation matrix T_3^0 is given by

$$T_3^0 = A_1^0 A_2^0 A_3^0$$

$$= \begin{bmatrix} c_{\alpha+\beta+\gamma} & -s_{\alpha+\beta+\gamma} & 0 & l_1 c_\alpha + l_2 c_{\alpha+\beta} + l_3 c_{\alpha+\beta+\gamma} \\ s_{\alpha+\beta+\gamma} & c_{\alpha+\beta+\gamma} & 0 & l_1 s_\alpha + l_2 s_{\alpha+\beta} + l_3 s_{\alpha+\beta+\gamma} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{13}$$

Inverse Kinematics:

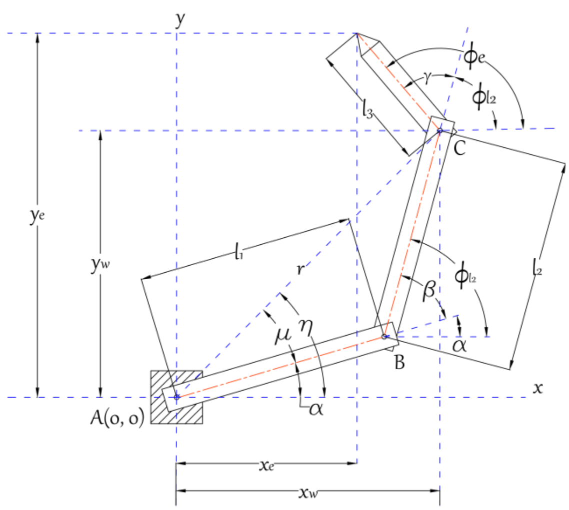

Fig. 6: Upper Right Limb at Posture 1

ϕ_e is the orientation of the end-effecter ($\phi_e = \alpha + \beta + \gamma$). For the URL, it is measured from the positive x-axis anticlockwise up to the center line of the end-effecter while for the ULL,it is measured from the negative x-axis clockwise up to the center-line of the end-effecter. x_w and y_w are respectively the abscissa and the ordinate of the wrist from the origin (shoulder joint).

The forward kinematics equations are:

$$x_e = l_1 c_\alpha + l_2 c_{\alpha+\beta} + l_3 c_{\phi_e} \tag{14}$$

$$y_e = l_1 s_\alpha + l_2 s_{\alpha+\beta} + l_3 s_{\phi_e} \tag{15}$$

In terms of the wrist positions,

$$x_e = x_w + l_3 c_{\phi_e} \tag{16}$$

$$y_e = y_w + l_3 s_{\phi_e} \tag{17}$$

i.e.

$$x_w = l_1 c_\alpha + l_2 c_{\alpha+\beta} \tag{18}$$

$$y_w = l_1 s_\alpha + l_2 s_{\alpha+\beta} \tag{19}$$

Solving the wrist eq.s (18) and (19) by squaring and summing, we have

$$x_w{}^2 + y_w{}^2 = l_1{}^2 + 2 l_1 l_2 c_\beta + l_2{}^2 \tag{20}$$

$$c_\beta = \frac{x_w{}^2 + y_w{}^2 - l_1{}^2 - l_2{}^2}{2 l_1 l_2} \tag{21}$$

Substituting c_β into the half-angle identity $\tan(\beta/2) = \sqrt{\frac{1-c_\beta}{1+c_\beta}}$ to obtain β with arc tangent, we have

$$\beta = 2\ atan\left[\frac{(l_1 + l_2)^2 - (x_w{}^2 + y_w{}^2)}{x_w{}^2 + y_w{}^2 - (l_1 - l_2)^2}\right]^{1/2} \tag{22}$$

In terms of the desired end-effecter position (x_e, y_e) and orientation ϕ_e, put $x_w{}^2 = \left(x_e - l_3 c_{\phi_e}\right)^2$ and $y_w{}^2 = \left(y_e - l_3 s_{\phi_e}\right)^2$ (see eq.s (16) and (17))

$$\boldsymbol{\beta = 2\ atan\left\{\frac{(l_1 + l_2)^2 - \left[\left(x_e - l_3 c_{\phi_e}\right)^2 + \left(y_e - l_3 s_{\phi_e}\right)^2\right]}{\left(x_e - l_3 c_{\phi_e}\right)^2 + \left(y_e - l_3 s_{\phi_e}\right)^2 - (l_1 - l_2)^2}\right\}^{1/2}} \tag{23}$$

To solve for α, we can rewrite the wrist equations as

$$x_w = l_1 c_\alpha + l_2 c_\alpha c_\beta - l_2 s_\alpha s_\beta \tag{24}$$

$$y_w = l_1 s_\alpha + l_2 s_\alpha c_\beta + l_2 c_\alpha s_\beta \tag{25}$$

Solving eq.s (24) and (25), for c_α,

$$c_\alpha = \frac{x_w\left(l_1 + l_2 c_\beta\right) + y_w l_2 s_\beta}{x_w{}^2 + y_w{}^2} \tag{26}$$

or solving for s_α,

$$s_\alpha = \frac{y_w\left(l_1 + l_2 c_\beta\right) - x_w l_2 s_\beta}{x_w{}^2 + y_w{}^2} \tag{27}$$

Substituting c_α and s_α into the identity $\tan\alpha = s_\alpha / c_\alpha$ to obtain α with arc tangent, we have

$$\alpha = atan\left[\frac{y_w(l_1 + l_2 c_\beta) - x_w l_2 s_\beta}{x_w(l_1 + l_2 c_\beta) + y_w l_2 s_\beta}\right] \tag{28}$$

$$\boldsymbol{\alpha = atan\left[\frac{(y_e - l_3 s_{\phi_e})(l_1 + l_2 c_\beta) - (x_e - l_3 c_{\phi_e}) l_2 s_\beta}{(x_e - l_3 c_{\phi_e})(l_1 + l_2 c_\beta) + (y_e - l_3 s_{\phi_e}) l_2 s_\beta}\right]} \tag{29}$$

Solving geometrically for the wrist joint angle, $\gamma = \phi_e - (\alpha + \beta)$ (30)

2.2.5 The Upper Right Limb at P2

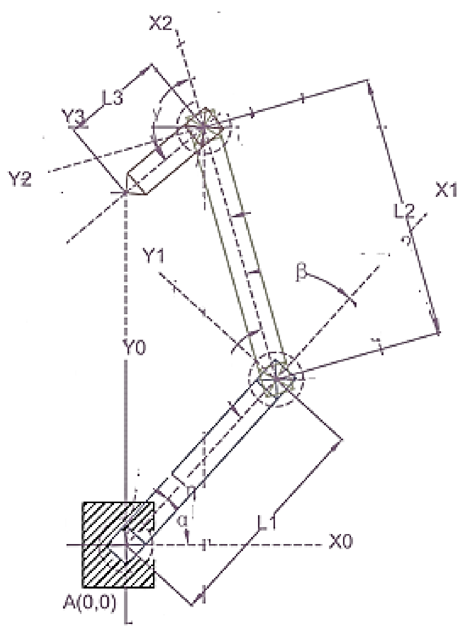

Fig 7: The Upper Right Limb at Posture 2

The link parameters for this configuration (URL P2) are same as that of URL P1 so all its forward and inverse kinematics equations will also be same (but in this arm posture, $x_e = 0$).

2.2.6 The Upper Right Limb at P3

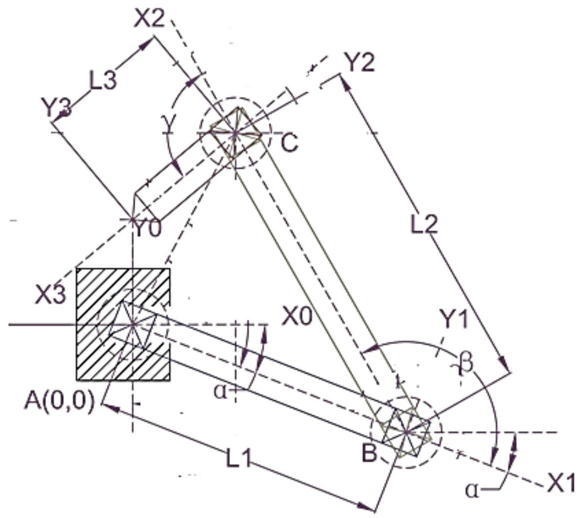

Fig 8: The Upper Right Limb at Posture 3

Forward Kinematics:

Table 4: Link Parameters for the URL at Posture 3

Link	Link length l_i	Link Twist t_i	Link Offset h_i	Joint Angle θ_i
1	l_1	0	0	$-\alpha$
2	l_2	0	0	β
3	l_3	0	0	γ

$$A_1^0 = \begin{bmatrix} c_\alpha & s_\alpha & 0 & l_1 c_\alpha \\ -s_\alpha & c_\alpha & 0 & -l_1 s_\alpha \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} A_2^0 = \begin{bmatrix} c_\beta & -s_\beta & 0 & l_2 c_\beta \\ s_\beta & c_\beta & 0 & l_2 s_\beta \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},\ A_3^0 = \begin{bmatrix} c_\gamma & -s_\gamma & 0 & l_3 c_\gamma \\ s_\gamma & c_\gamma & 0 & l_3 s_\gamma \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_3^0 = \begin{bmatrix} c_{\alpha-\beta-\gamma} & s_{\alpha-\beta-\gamma} & 0 & l_1 c_\alpha + l_2 c_{\alpha-\beta} + l_3 c_{\alpha-\beta-\gamma} \\ -s_{\alpha-\beta-\gamma} & c_{\alpha-\beta-\gamma} & 0 & -l_1 s_\alpha - l_2 s_{\alpha-\beta} - l_3 s_{\alpha-\beta-\gamma} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{30}$$

Inverse Kinematics:

Solving for the joint angles, α and β as we did in URL P1 we obtain the same α and β models. (Also in this posture, $x_e = 0$).

The γ model here is different

$$\gamma = \phi_e - (\beta - \alpha) \tag{31}$$

2.3 Trajectory Tracking Models

The limbs must swing through specified paths in order to achieve the programmed postures. We shall develop (using point-to-point motion technique) motion timing laws from which we can generate paths to track the limb positions within certain constraints of particular interest. Third-order polynomial functions provide valid solutions to generate trajectories. Let us express the end-effecter position p as third-order polynomial time function

$$p(t) = k_3t^3 + k_2t^2 + k_1t + k_0 \tag{32}$$

where $k_3,\ k_2,\ k_1$ and k_0 are arbitrary constants. Differentiating $p(t)$ will fetch us the velocity function.

$$v(t) = 3k_3t^2 + 2k_2t + k_1 \tag{33}$$

To solve for the arbitrary constants let us set boundary conditions as the end-effecter moves from an initial position p_i to a final position p_f in time t_f.

Table 5: Generalized Boundary Conditions for Each Swing

	Initial pose	Goal pose
End-effecter position, $p(t)$	p_i	p_f
End-effecter velocity, $v(t)$	0	0
Time taken, t	0	t_f

Applying the initial conditions to eq.s (32) and (33) we obtain

$$k_0 = p_i \quad \text{and} k_1 = 0$$

Applying the goal conditions to the eq.s (32) and (33) we obtain

$$p_f - p_i = k_3{t_f}^3 + k_2{t_f}^2 \tag{34}$$

$$0 = 3k_3{t_f}^2 + 2k_2t_f \tag{35}$$

Solving eq.s (34) and (35) we have $k_2 = \frac{3(p_f - p_i)}{{t_f}^2}$ and $k_3 = \frac{-2(p_f - p_i)}{{t_f}^3}$

With the constants $k_0,\ k_1,\ k_2$ and k_3 established, we can thus rewrite the original functions –eq.s (32) and (33) as

$$p(t) = \frac{-2(p_f - p_i)}{{t_f}^3}t^3 + \frac{3(p_f - p_i)}{{t_f}^2}t^2 + p_i \tag{36}$$

$$v(t) = \frac{-6(p_f - p_i)}{{t_f}^3}t^2 + \frac{6(p_f - p_i)}{{t_f}^2}t \tag{37}$$

3.0 Results and Discussions

3.1 Inverse Kinematics Results

Table 6: Specified End-Effecter Tip Coordinates and End-Effecter Orientations (for Upper Limbs) with their Corresponding Joint Angles Generated from the Inverse Kinematic Models.
(l_1 = 0.315m, l_2 = 0.369m and l_3 = 0.144m).

Reference Limb	Set Posture	Specified End-Effecter Tip Coordinates from the Origin (x_e, y_e) (m,m)	Specified End-Effecter Orientation ϕ_e (rad) (for the upper limbs)	Required Shoulder/ Hip Joint Angle (Inner Links) α (rad)	Required Elbow/Knee Joint Angle (Outer Link for Upper Limbs/End-Effecter Link for Lower Limbs) β (rad)	Required Wrist Joint Angle (End-Effecter Link for Upper Limbs) γ (rad)
URL	P1	(0.305,0.556)	2.269	0.288	1.022	0.959
URL	P2	(0,0.495)	3.840	0.832	1.018	1.990
URL	P3	(0,0.113)	3.840	0.375	2.464	1.751

Table 7: Specified End-Effecter Tip Coordinates (for Lower Limbs) with their Corresponding Joint Angles Generated from the Inverse Kinematic Models.
(l_1 = 0.315m and l_2 = 0.369m).

Reference Limb	Set Posture	Specified End-Effecter Tip Coordinates from the Origin (x_e, y_e) (m,m)	Required Shoulder/Hip Joint Angle (Inner Links) α (rad)	Required Elbow/Knee Joint Angle (Outer Link for Upper Limbs/End-Effecter Link for Lower Limbs) β (rad)
LRL	P1	(0.397,-0.446)	0.288	1.022
LRL	P2	(0,-0.518)	0.787	1.431
LRL	P3	(0,-0.135)	0.222	2.777

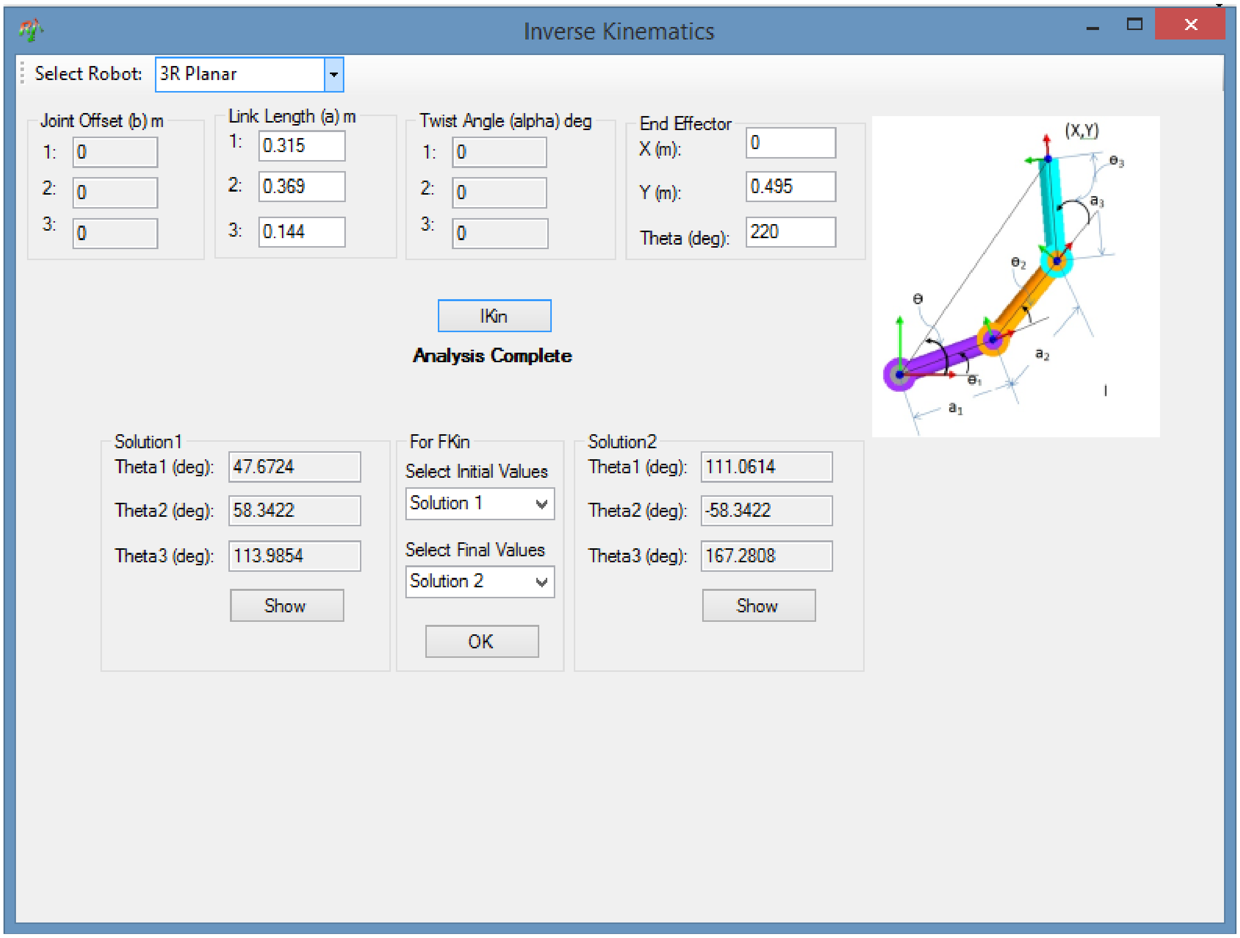

Plate 1: Screen Snip of the RoboAnalyzer Inverse Kinematics Window Showing Results of the URL at P2

On tables 6 and 7, the target positions for the treebot's end-effecters, as it climbs, have been specified and the required joint angles were generated using the inverse kinematics models developed in this paper. The same α and β models are valid for the three postures of the LRL. For the URL, the same α and β models are valid for the three postures but its γ models differ (one γ model is applicable in P1 and P2 while the other γ model in P3). The results are comparable to the ones obtained with RoboAnalyzer robotics software (note that the software gives its results in degrees). Plate 1 shows a screen snip of results from the software for posture P2 of the URL.

3.2 Trajectory Tracking Results

The trajectories can be used to trace the positions (with respect to time) of the joint angles and the end-effecter coordinates per motion. To demonstrate this, let us consider the LRL configuration as it changes its posture from say from P2 to P1 (swinging out from tree trunk) with the end-effecter moving from a point $(0, -0.518)$ to another point $(0.397, -0.446)$ in 1 second i.e.

$$(x_{ei}, y_{ei}) = (0, -0.518) \quad ; \quad (x_{ef}, y_{ef}) = (0.397, -0.446)$$

Using the position function eq. (36) to generate the ordinate and abscissa functions,

$$y_e(t) = -0.144t^3 + 0.216t^2 - 0.518 \tag{38}$$

$$x_e(t) = -0.794t^3 + 1.191t^2 \tag{39}$$

We can substitute these functions –eq.s (38) and (39) into the LRL's inverse kinematics models –eq.s (6) and (11) to trace the time histories of the joint angles i.e. to plot the joint angles w.r.t. time. The LRL is configured with $l_1 = 0.315m$ and $l_2 = 0.369m$ and the calculated joint angle β at target P1 is $1.022rad$, thus $c_\beta = 0.5216$, $s_\beta = 0.8532$. Substituting we have

$$\beta = 2\,atan\left\{\frac{0.4679 - [(-0.794t^3 + 1.191t^2)^2 + (-0.144t^3 + 0.216t^2 - 0.518)^2]}{(-0.794t^3 + 1.191t^2)^2 + (-0.144t^3 + 0.216t^2 - 0.518)^2 - 0.0029}\right\}^{1/2} \quad (40)$$

and

$$\alpha = atan\left[\frac{0.3231t^3 - 0.4845t^2 + 0.2629}{-0.3577t^3 + 0.5364t^2 + 0.1631}\right] \quad (41)$$

Plotting α and β for values of $t = 0s, \dots, 1s$ we have

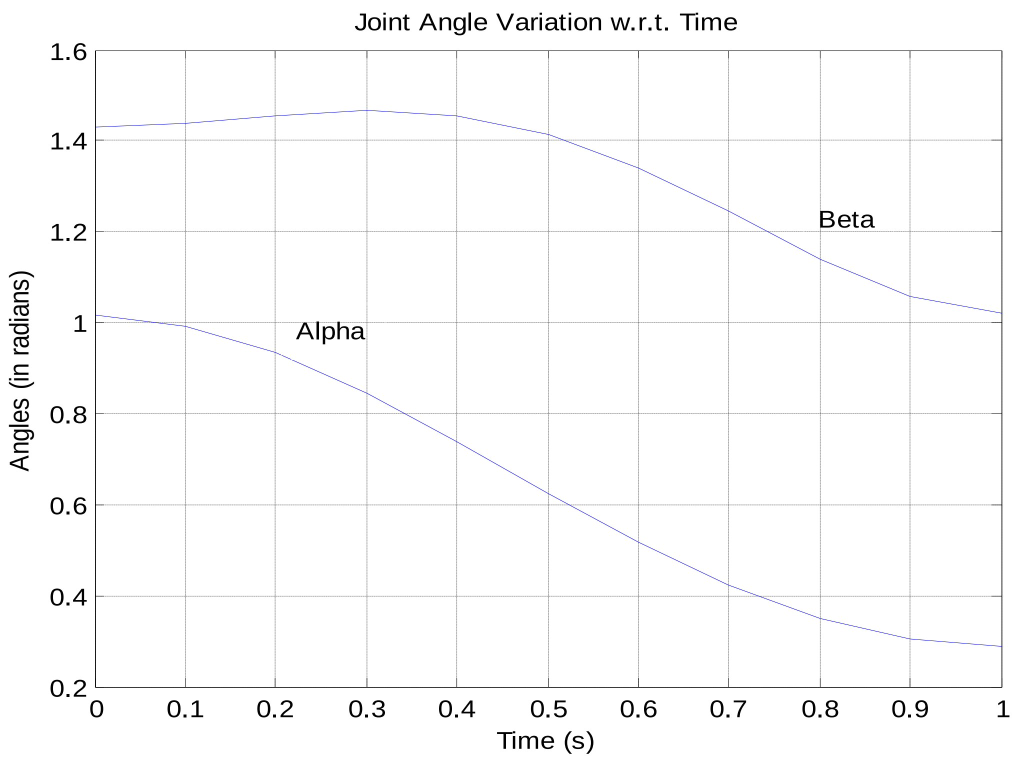

Fig. 9: Joint Angles Variations of the LRL from P2 to P1

Now plotting x_e and y_e for values of $t = 0s, \dots, 1s$ we have

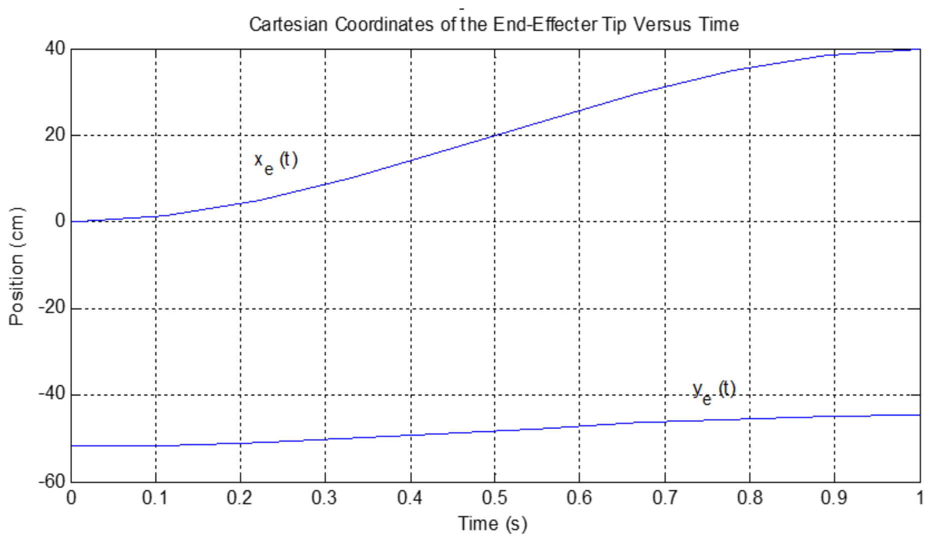

Fig. 10: Position-Time History of the LRL's End-Effecter Coordinates from P2 to P1

In fig. 9, as the LRL moves from P2 to P1, the joint angles vary simultaneously through the motion. The initial and final joint angles conform with the results on table 7 and are within the time constraint. Fig. 10 shows the trajectories of the coordinates of the tip of the end-effecter as the LRL moves from P2 to P1 in $1s$. The initial and final coordinates also conform with the respective coordinates on table 7.

4.0. Conclusion

The target positions of the end-effecters and the joint parameters of the pamtreebot as it climbs, have been represented mathematically in this paper.The information presented in this paper also provides a basis for further research into 3 dimensional limb motions for the pamtreebot and if further exploited will launch an industry for large scale production of agro robots in Nigeria aimed at safer palm tree farming and economic boost in the country.

5.0 Recommendations

The following areas should be considered in future research

- Developing a software to be installed on a microcontroller chip (embedded on the pamtreebot) which can use the mathematical models developed to quickly compute the required kinematic and dynamic properties for every desired end-effecter target and send corresponding signals to the servos for implementation. A version of the software of the software could be used for animating the pamtreebot motions on a computer.
- Developing position models for 3 dimensional motion of the pamtreebot

Nomenclature

l_i – length of link i (m)
θ_i– generalized joint angle (rad)
α – shoulder/hip joint angle (rad)
β – elbow/knee joint angle (rad)
γ – wrist joint angle (rad)
s_α sine of α
c_α cosine of α
$s_{\alpha+\beta} = \sin(\alpha+\beta)$
$c_{\alpha-\beta} = \cos(\alpha-\beta)$
ϕ_{l_2} – orientation of link 2 (rad)
ϕ_e – orientation of the end-effecter (rad)
(x_e, y_e, z_e) – position of end-effecter's tip from the origin (m)
(x_w, y_w, z_w) – position of wrist from the origin (m)

URL – Upper Right Limb
ULL – Upper Left Limb
LRL– Lower Right Limb
LLL – Lower Left Limb
L1– Link 1
L2 – Link 2
EL2 – End-Effecter Link 2
EL3 – End-Effecter Link 3
RRR – three revolute joints
RR – two revolute joints
PRR – first joint prismatic and two revolute joints
DOF – degree of freedom
A_i^0 – homogenous matrix
T_i^0 – transformation matrix
$p(t)$ – end-effecter position function
$v(t)$ – end-effecter velocity function

References

Ali R., Noboru N. and Tatsuki K., 2019. International Journal of Agricultural and BiologicalEngineering, 12(1):146-158.

Amarendra J.H., Robins M. and Somashekhar S.H., 2018.Mathematical Model to Estimate the Position of Mobile Robot by Sensing Caster Wheel Motion, MATEC Web of Conferences, 144(24).

Chaitanyaa G. and Reddy S., 2016. Genetic Algorithm Based Optimization of a Two Link Planar Robot Manipulator. International Journal of Lean Thinking 7(2), 1 – 13.

Dubey A., Santosh P. and Arunava B., 2016. Autonomous Control and Implementation of Coconut Tree Climbing and Harvesting Robot. Procedia Computer Science 85, 755 – 766.

Farhan A.S., 2014. Modeling, Simulation and Control Issues for a Robot ARM. Education and Research (III), International Journal of Intelligent Systems and Applications, 6(4): 26-39,DOI:10.5815/ijisa.2014.04.03.

Gaoyuan C., Bin Z. and Rodrigues M., 2020. Trajectory Simulation of Badminton Robot Based on Fractal Brown Motion. Fractals 28(8).

Hossein S. and Hassan Z., 2014. Modeling, Simulation and Position Control of 3DOF Articulated Manipulator. Indonesian Journal of Electrical Engineering and Informatics (IJEEI), 2(3),132-140.

Khin M.M., Zaw M.M.H. and Hla M.T., 2016. Position Control Method for Pick and Place Robot Arm for Object Sorting System. International Journal of Scientific & Technology Research, 5(6), 57 – 61.

Mahil S. and Al-durra A.2016.Modelling, Analysis and Simulation of 2 DOF Robotic Manipulator. 2016 IEEE 59th International Midwest Symposium on Circuits and Systems (MWSCAS),DOI:10.1109/MWSCAS.2016.7870099

Mahmoud M. Al A., 2014. Trajectory Tracking Control of a 2-DOF Robot Arm Using Neural Networks, Islamic University of Gaza, Thesis, 1 – 86.

Mohammed A.A. and Mehmet S., 2015.Kinematics Modeling of a 4-DOF Robotic Arm. 2015 International Conference on Control, Automation and Robotics, Singapore, DOI:10.1109/ICCAR.2015.7166008.

Okubanjo A.A., Oyetola O.K., Osifeko M.O., Olaluwoye O.O. and Alao P.O., 2017. Modeling of 2-DOF Robot Arm and Control. FUTO Journal Series, 3(2), 80 – 92.

Panchanand J., 2015. Inverse Kinematic Analysis of Robot Manipulators. National Institute of Technology Rourkela, Thesis, 96 – 137.

Sajjad A.S.F.B, Uk M.E., Soyaslan M. and Eldogan O., 2018.Modeling, Control, and Simulation of a SCARA PRR-Type Robot Manipulator. Scientia Iranica, B (2020) 27(1), 330 – 340.

Sane S., 2015. Studies On Trajectory Tracking of Two Link Planar Manipulator. National Institute of Technology Rourkela, Thesis, 5 – 7.

Shekhar C. and Jayswal S.C., 2016. Task Time Optimization of Two D.O.F. Robot Arm Manipulator Using Genetic Algorithm. International Journal of Mechanical and Production Engineering, 4(2), 75 – 80.

Subedi D., Ilya T. and Geir H., 2021. Dynamic Modeling of Planar Multi-Link Flexible Manipulators. MDPI, 10(70), https://doi.org/10.3390/robotics10020070.

Virgala I., Michal K., Martin V. and Piotr K., 2014.Analyzing, Modeling and Simulation of Humanoid Robot Hand Motion. Procedia Engineering, Vol. 96, 489 – 499. https://doi.org/10.1016/j.proeng.2014.12.121.