Engineering and Applied Sciences
2023; 8(2): 16-20
http://www.sciencepublishinggroup.com/j/eas
doi: 10.11648/j.eas.20230802.11
ISSN: 2575-2022 (Print); ISSN: 2575-1468 (Online)

Analysis of Forward Kinematics of 2R Robotic Arm

Fatima Mohammad Amin

School of Electronics Engineering, Vellore Institute Technology (VIT), Vellore, India

Email address:
fatimamohammad.amin2020@vitstudent.ac.in

To cite this article:
Fatima Mohammad Amin. Analysis of Forward Kinematics of 2R Robotic Arm. *Engineering and Applied Sciences*. Vol. 8, No. 2, 2023, pp. 16-20. doi: 10.11648/j.eas.20230802.11

Received: September 27, 2022; **Accepted**: October 26, 2022; **Published**: May 18, 2023

Abstract: In Kinematics, the analysis of acceleration and velocity is usually done whereas the forces are neglected. The machine's smallest unit is the link, which is akin to a cell in the human body. Robotics is built on the foundation of machines. A pair is established by the combination of links which leads to the construction of linkage. A mechanical linkage is a collection of bodies that are connected to control forces and movement. The combination of linkage (also known as Mechanism) generates Machine. The robot's kinematics equations are used in robotics, video games and animation. Inverse kinematics is the mathematical process of determining variable joint parameters necessary to place an object at the end of a kinematic chain in a specific position and orientation with respect to the start of the chain, such as robot manipulator or skeleton of an animated character. The role of Forward and Inverse Kinematics in a 2R Robotic Arm are studied with successful demonstrations. Forward Kinematics of a 2R Robotic Arm using Matlab and Python confirming similar results are exhibited (Figure 8 and Figure 9). Matlab illustrated 2D Path Tracing for 2R Robotic Arm with Inverse Kinematics. Finally, inverse kinematics of 2R Robotic Arm in Matlab using fuzzy logic is modelled successfully. Thus, it helps in the understanding of kinematics and can be used in the simulation of machines.

Keywords: Kinematics, Robotic Arms, Machines, Linkage, Matlab, Python, Fuzzy Logic

1. Introduction

A 2R Robotic Arm constitutes of a base, two arms or kinematic links and two rotating joints. Motion is the result of programming of these links. With regards to the base, link one will produce an angle theta 1 and link two an angle of theta 2, respectively [4]. Due to the robot's joints and degrees of freedom, the challenges with its kinematics are exceedingly complicated [12].

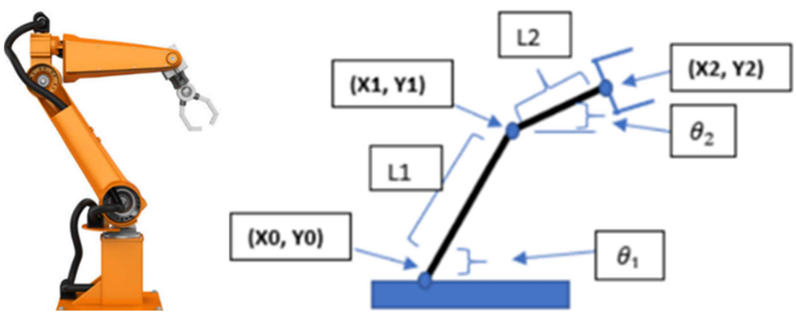


Figure 1. *The above figures represent the 2R Robotic Arm.*

Forward kinematics is the technique of calculating position and velocity of an end effector from known joint angles and angular velocities. For instance, wrist / cartesian fist's coordinates can be determined if arm's shoulder and elbow joint angles are known in sagittal plane. The application of a robot's kinematic equations is to determine position of end-effector, provided joint parameter values is known as forward kinematics. Robotics, video games and animation all make use of the robot's kinematics equations [1].

The robot's fundamental arms are employed here. The linkage known as Forward kinematics of a 2R Robotic Arm (where 2R means 2-rotating Link i.e., 2 is several link and R is rotation) is mainly focused upon. Two links are used to represent Robotic Arm with Link-1 hinged at (X0, Y0) and forming an angle 1, and Link-2 hinged towards end of Link-1 and forming an angle 2 with respect to datum. (Figure 7)

It is possible to position end of a kinematic chain such as a robot manipulator or skeleton of animated character at a precise location and orientation in reference to the start of the chain using mathematical technique known as inverse kinematics [5, 6].

The focus is to use MATLAB to evaluate forward and inverse kinematics and be able to demonstrate for a 2R Robotic Arm. It helps in understanding of kinematics and demonstrate that it can be used in simulation of machines. We need to explore role of Forward and Inverse Kinematics in a 2R Robotic Arm. The key idea is to first identify an admissible trajectory that complies with boundary value restrictions as well as system dynamics constraints [10]. Denavit-Hartenberg notation can also be used to generate the forward kinematics. [11]

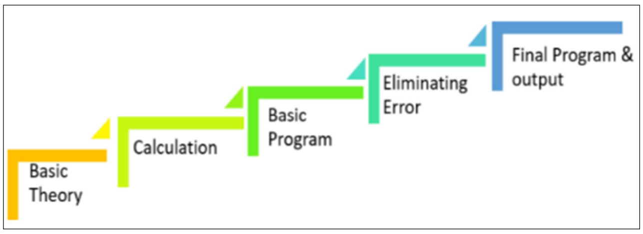
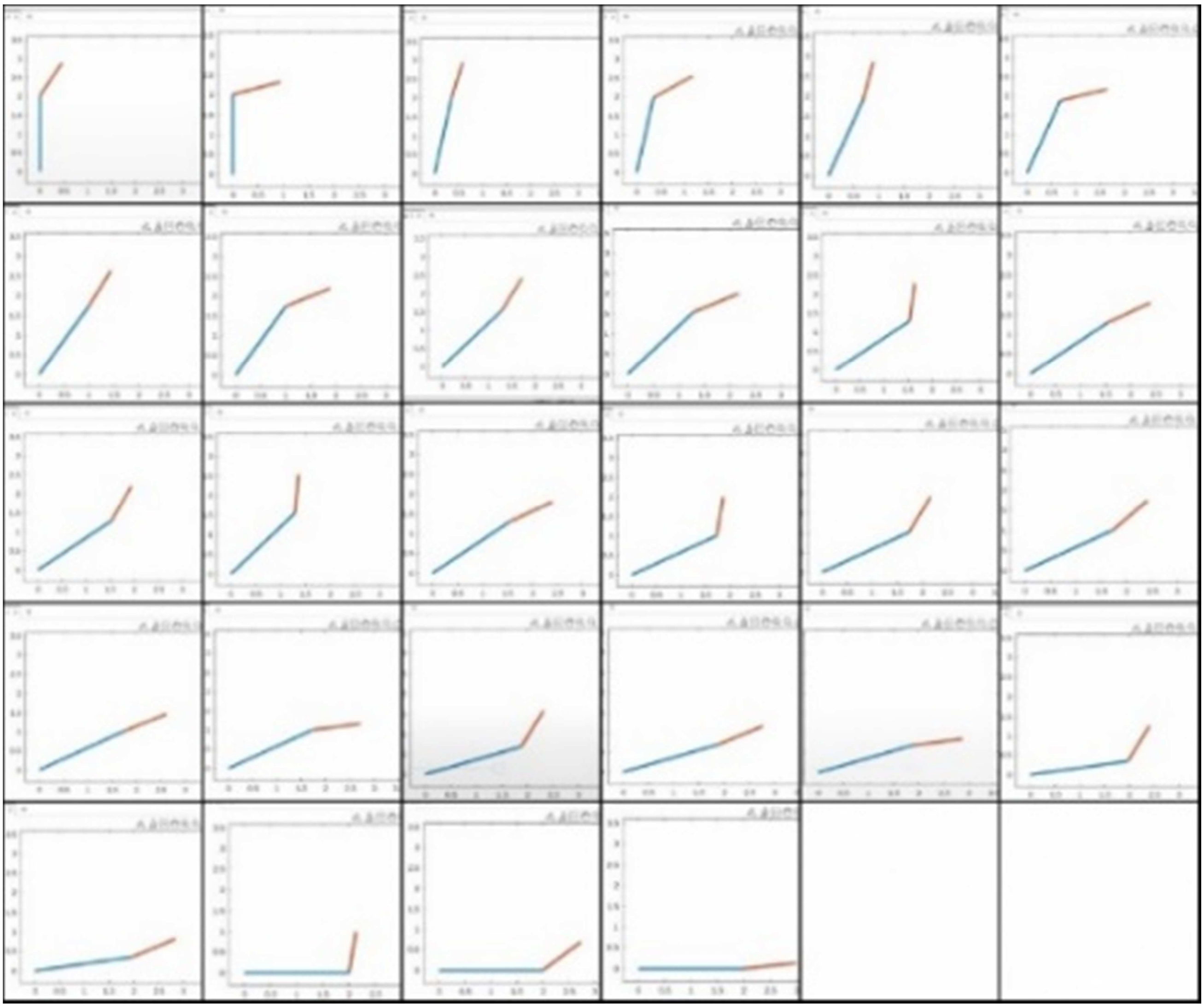


Figure 2. *Flowchart / Steps followed.*

2. Forward Kinematics of 2R Robotic Arm

In Matlab, we obtained below images in video like format, wherein cycle repeats itself.



Click below link for video presentation: https://drive.google.com/file/d/1NzmSkpgr-XBH13Fylf-rPurO_wrAjymj/view

Figure 3. *2R Robotic Arm's Forward Kinematics (on Matlab).*

We get expected result (same as Figure 9).

In Python, animation is made by mixing png files of various arm movements at different positions. Some of output images have been shown below:

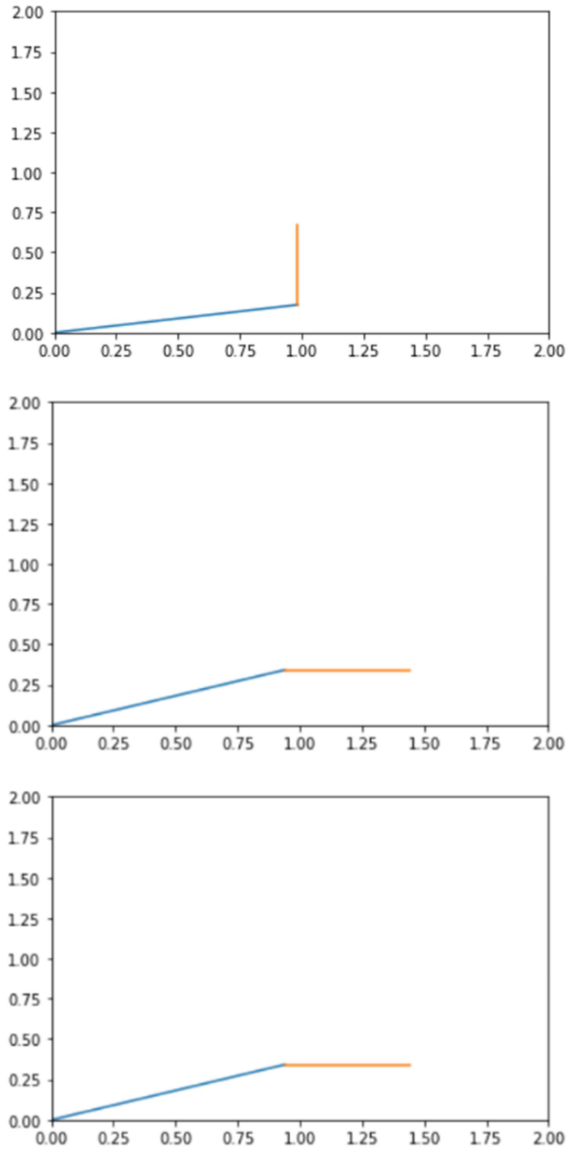


Figure 4. *Output images using Python.*

The results are again as expected (replication of Figure 9).

Click below link for video presentation:

https://drive.google.com/file/d/1S07853V8_1bTgXoHEVjHAVdyj2RDV8Rc/view?usp=sharing

3. 2-D Path Tracing with Inverse Kinematics

The code shows how to utilize inverse Kinematics class to compute inverse kinematics for a simple 2D manipulator. A simple rigid body tree object-based two-degree-of-freedom planar manipulator with revolute joints makes up the manipulator robot. A circular trajectory is built in a 2-D plane and delivered as points in inverse kinematics solver. The solver chooses right joint locations to produce this trajectory. The robot is then animated to show how its settings enable it to follow a circular trajectory. The crucial and challenging component of the manipulator's real-time control is its inverse kinematics [14]. By using inverse kinematics, the location and orientation of the end-effector are used to determine the joint variables. Because of the nonlinearity equations, solving inverse kinematics is regarded as a challenging issue [15].

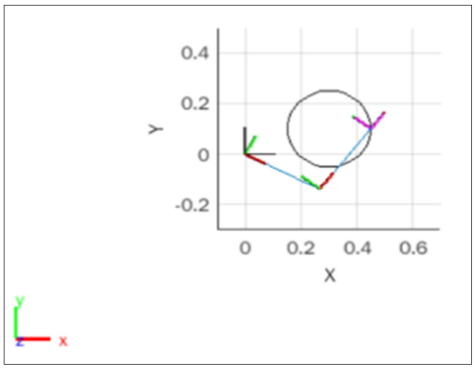


Figure 5. *Demonstrates the 2-D Path Tracing with Inverse Kinematics.*

4. Modelling Inverse Kinematics in a Robotic Arm

Given desired location of arm's tip, angles at joints can be numerically determined for straightforward designs like two-jointed Robotic Arm. The inverse kinematics of more complicated systems might be tricky to solve mathematically (such as n-jointed Robotic Arms operating in a three-dimensional input space). If forward kinematics of the tissue is known, we can construct a fuzzy inference system utilizing fuzzy logic that determines inverse kinematics without need for analytical solution. Furthermore, there is no need for extra background information to understand or evaluate the fuzzy answer [3].

Since two-joint Robotic Arm's forward kinematics formulas are known, it can be used to predict x and y coordinates of arm's tip for entire two-joint rotational angles. The angles and coordinates are recorded as training data for ANFIS (adaptive neuro-fuzzy inference system) network. ANFIS network learns how to convert coordinates (x, y) to angles during training (theta1, theta2). ANFIS network is then trained to control the Robotic Arm as part of a broader control system [7].

The control system moves Robotic Arm to designated place by applying force to its joints in line with angular orientations of those joints as determined by trained ANFIS network [2].

ANFIS is a hybrid neuro-fuzzy technique that gives fuzzy inference systems access to neural networks' learning capabilities. The learning algorithm modifies membership functions of a Sugeno-type fuzzy inference system using training input/output data [2]. The input/output data in this case refers to "coordinates/angles" dataset. The coordinates are fed into ANFIS, which then produces angles. The learning algorithm instructs ANFIS to map the coordinates to angles through a procedure known as training. By the time training is over, ANFIS network have mastered input-output map and is ready for deployment into the larger control system solution.

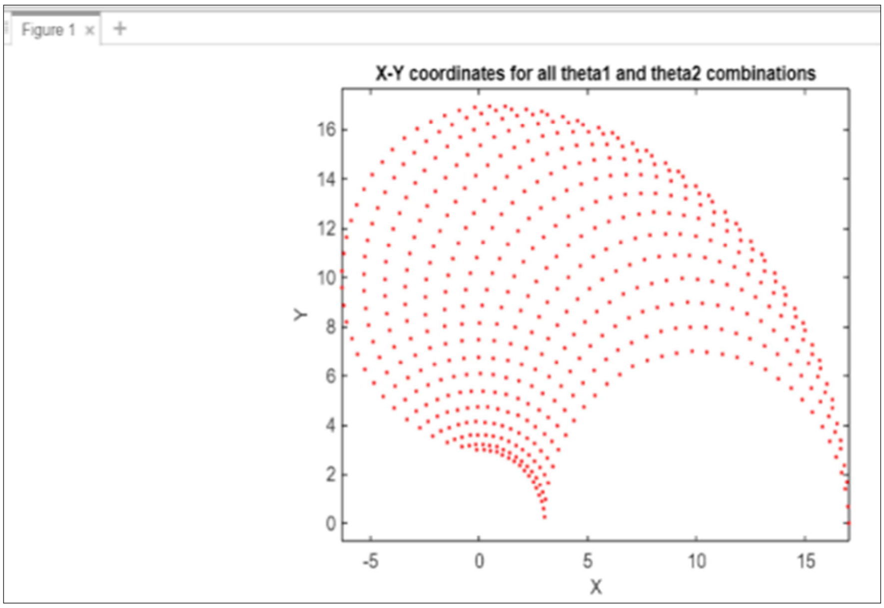


Figure 6. *Demonstrates the X and Y coordinates for all the theta1 and theta2 combinations.*

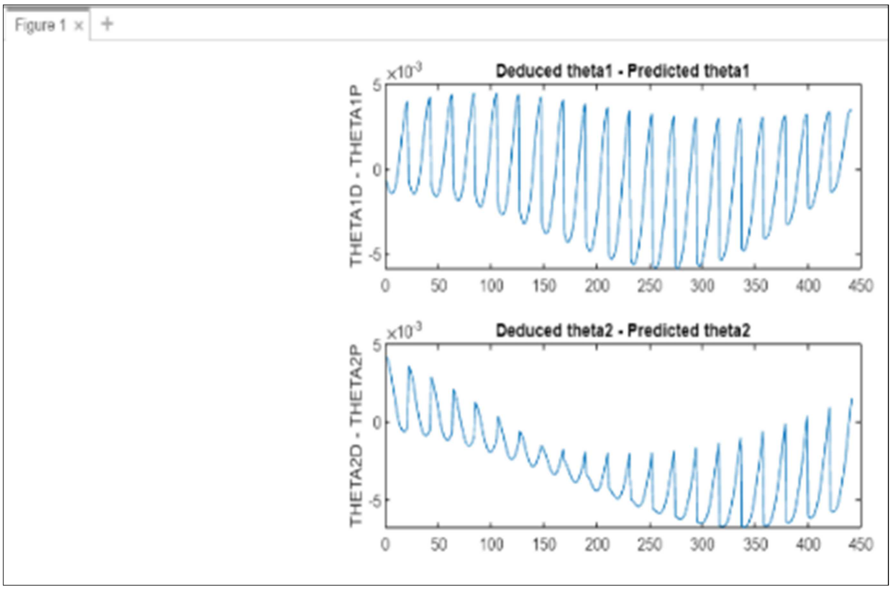


Figure 7. *The difference between the deduced and predicted theta values are compared in this simulation.*

For all theta1 and theta2 combinations, X-Y coordinates were generated using forward kinematics formulas. The X-Y data points are obtained by iteratively combining various values of theta1 and theta2, then determining x and y coordinates. (Figure 6)

To determine how successfully ANFIS networks will function inside larger control system, it is required to validate networks after they have been trained. It is possible to compare outcomes of ANFIS networks to outcomes of generated formulae because this project involves a two-joint Robotic Arm with inverse kinematics equations [8].

The next code does the same, and provides the difference between the deduced and predicted theta value (Refer Figure 9).

5. Conclusion

Kinematics can be utilized further to simulate machines i.e., robots (this concept can be widely applied in biomedical industry). Using Robotic Arms' inverse kinematics, it is possible to determine joint angles that must be adjusted to achieve a specific end-effector position. Robot kinematics investigates relationship between location, velocity, and acceleration of each link. The size and connectivity of kinematic chains are also considered in coordinating movement to compute actuator forces and torques. Forward and Inverse Kinematics is an essential component of robotics and is examined prior to creation of robot.

We may utilize programme in a variety of applications by minor alteration. In Matlab codes, we can generate various outputs by modifying data and generating small tweaks (e.g., moving Robotic Arm in multiple directions with multiple speeds can be achieved). Many specific outputs are also possible (such as movement of arm in same direction at different speeds or demonstrating Forward and Inverse Kinematics of a 3R Robotic Arm). The dynamics of the arm during motion are exceedingly complex due to the unpredictability of the load, gravity, and velocity contact forces between its segments, hence research in this area is extremely crucial [9]. Robot handwriting is challenging to implement due to specific problems like degrees of freedom and kinematic redundancy. This idea can be applied there to address those problems [13].

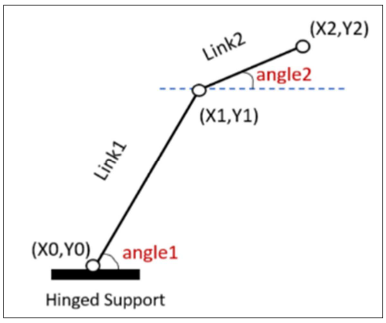


Figure 8. *The figure represents the 2R Robotic Arm's forward motion.*

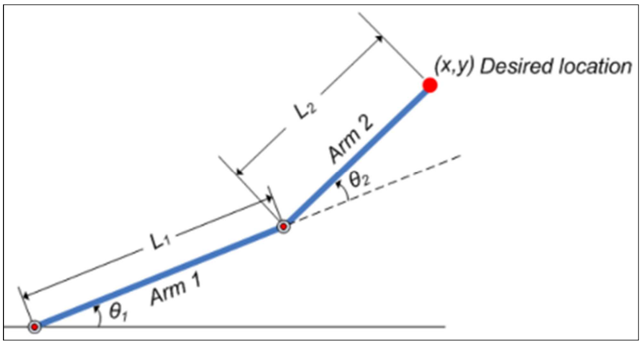


Figure 9. *The figure represents the robot arm's inverse kinematics, type 2R.*

References

[1] Y. Pan, J. Tang, Q. Zhao and S. Zhu, (2020), Forward and Inverse Kinematics Modelling and Simulation of Six-axis Joint Robot Arm Based on Exponential Product Method, IEEE 3rd International Conference on Automation, Electronics and Electrical Engineering.

[2] Pannawit Srisuk, Adna sento, Yuttana Kitjaidure, (2017), Inverse Kinematics Solution using Neural Networks from Forward Kinematics Equations, 9th International Conference on Knowledge and Smart Technology (KST).

[3] S. Fuenzalida, K. Toapanta, J. Paillacho and D. Paillacho, (2019), Forward and Inverse Kinematics of a Humanoid Robot Head for Social Human Robot-Interaction, IEEE Fourth Ecuador Technical Chapters Meeting (ETCM).

[4] Prayook Jatesiktat, Wei Tech Ang, (2017), Recovery of forearm occluded trajectory in Kinect using a wrist-mounted Inertial Measurement Unit, 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC).

[5] M. H. Rahman, Member, IEEE, M. Saad, Senior Member, IEEE, J. P. Kenné, and P. S. Archambault, Exoskeleton robot for rehabilitation of elbow and forearm movements, 18th Mediterranean Conference on Control and Automation, MED'10.

[6] Lincong Luo, Liang Peng, Zengguang Hou, Weiqun Wang, (2017), Design and Control of a 3-DOF Rehabilitation Robot for Forearm and Wrist, 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC).

[7] Sebastián Fuenzalida, Keyla Toapanta, Jonathan Paillacho, Dennys Paillacho, (2017), Forward and Inverse Kinematics of a Humanoid Robot Head for Social Human Robot-Interaction, 9th International Conference on Knowledge and Smart Technology (KST).

[8] Yunhao Pan, China Jingdi Tang, China Qingyuan Zhao, China Shengyi Zhu, (2020), Forward and Inverse Kinematics Modeling and Simulation of Six-axis Joint Robot Arm Based on Exponential Product Method, IEEE 3rd International Conference on Automation, Electronics and Electrical Engineering (AUTEEE).

[9] Mahmood Nahvi (1991), Coordination of human upper arm and forearm motion in rapid extensions, NASA. Ames Research Center, Human Machine Interfaces for Teleoperators and Virtual Environments.

[10] Tan Chen, Bill Goodwine (2020), Control of A 2R Planar Horizontal Underactuated Manipulator, 16th International Conference on Control, Automation, Robotics and Vision (ICARCV).

[11] Kuang-Chyi Lee; Ping-Yen Chiang; Ha-Anh Truong, (2018), The forward kinematics for the HR1-NFU humanoid robot, IEEE International Conference on Advanced Manufacturing (ICAM).

[12] Wei Shen (2021), Kinematics Analysis and Trajectory Planning of Robot Based on MATLAB, 2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA).

[13] Summit Sayem; Feroz Ahmed (2017), A Forward Kinematics approach towards Humanoid Robot handwriting, 2nd International Conference on Electrical & Electronic Engineering (ICEEE).

[14] Peng Chen; Jingchang Long; Wen Yang; Jianxing Leng, (2021), Inverse Kinematics Solution of Underwater Manipulator Based on Jacobi Matrix, OCEANS 2021: San Diego – Porto.

[15] Mohamad Reda A. Refaai, (2021), Using Multiple Adaptive Neuro-Fuzzy Inference System to Solve Inverse Kinematics of SCARA Robot, 18th International Multi-Conference on Systems, Signals & Devices (SSD).