International Journal of Engineering Research and Technology.
ISSN 0974-3154 Volume 11, Number 11 (2018), pp. 1759-1779

http://www.irphouse.com

Inverse Kinematics Software Design and Trajectory Control Programming of SCARA Manipulator robot

Mohamed T. Eraky[1*], Dmitry V. Zubov[2]

[1]*Ph.D. postgraduate student, Moscow Polytechnic University, 38, Bolshaya Semenovskaya Street, Moscow, 107023, Russia*

[2]*Ph.D. Associate Professor, Moscow Polytechnic University, 38, Bolshaya Semenovskaya Street, Moscow, 107023, Russia*

**Corresponding Author*

Abstract

The objective of this research is to compute the inverse kinematics and control of a trajectory for your New Machine design. This study was done by using a developed geometrical approach method to compute inverse kinematics problems with a highly accurate result, saving a long time in mathematical calculations, easier and comfort in using for mechatronics and robotics designers and engineers. The advantages of this research, first is to design and study an inverse kinematics of three Degree of Freedom as Revolute-Revolute-Prismatic joints 3-DOF (RRP) Manipulator robot (such as exist in SCARA Robot), second is to control of end-effector trajectory, finding of all possible solutions with selection the optimal trajectory.

Keywords: LabVIEW; SolidWorks; Trajectory Control; SCARA Robot; Optimal design; Inverse kinematics; Motion; Mechatronics.

INTRODUCTION

The control of robot manipulators has been a research area for years and has developed various control strategies [1, 2, 3]. Due to the robot manipulators being composed of several joints bonded together, the joints have highly nonlinear dynamics with a strong link between them. This complicates the control task, especially with model uncertainties or external disturbances. Some techniques have been proposed with control systems that take the model into account, as a computed torque control [4, 5].

The work provides satisfactory results in terms of tracking errors and robustness. The uncertainties in the model due to bad estimates or model parameters are difficult to design an efficient algorithm based on a precise mathematical model. The work of Jiang and Ishida [6] proposes a dynamic trajectory tracking control of industrial robotmanipulators using a PD controller and a neural network controller. The neural network is a three-layer feed-forward network. The learning law of neural network weights was derived using a simplified dynamic model of the robot and a back-propagation theory. In [7], the problem of designing robust variable structure control and sliding mode planes was considered for the robot manipulator SCARA RP41. The simulation results show the robustness of the extension variable structure control, a sliding mode opposite the transported load, parametric variations, an imprecise model, and external perturbation signals. The work of [8] presents a discrete-time sliding mode neuro-adaptive control (DTSMNAC) method for robot manipulators. This control structure is a practical design that combines a discrete-time neuro-adaptation technique with sliding mode control to compensate the dynamics variations in the robot. Using an online adaptation technique, a DTSMNAC controller is used to approximate the equivalent control in the neighborhood of the sliding surface.

In this study, the design of an adaptive trajectory-tracking controller based on an inverse kinematics program and LabVIEW using NI-SoftMotion module as a controller is developed on a SCARA robot. This controller is designed based on connection between LabVIEW and Solidwork programs, where NI-SoftMotion module works to send and receive the data and information of inverse kinematic problem.

INVERSE KINEMATICS OF SCARA MANIPULATOR ROBOT

A. Kinematics Models

Let consider this work is to create CAD models of various mechanisms and robots and then to control their motion using LabVIEW programming. This would help in performing the actual kinematic analysis, through their CAD models. In general, there are two types of motion studies related to mechanism analysis – kinematic and dynamic [9]. Dynamics is the study of motion in response to externally applied loads. Kinematics studies the motion of bodies without consideration of the forces or moments that cause the motion. Robot kinematics refers the analytical study of the motion of a robot manipulator. Formulating the suitable kinematics models for a robot mechanism is very crucial for analyzing the behavior of industrial manipulators. There are mainly two different spaces used in kinematics modelling of manipulators namely, Cartesian space and Quaternion space. The transformation between two Cartesian coordinate systems can be decomposed into a rotation and a translation [10]. Denavit-Hartenberg (1955) method that uses four parameters is the most common method for describing the robot kinematics.Denavit & Hartenberg showed that a general transformation between two joints requires four parameters. These parameters known as the Denavit-Hartenberg (DH) parameters have become the standard for describing robot kinematics [9].

B. The main advantages of proposed control program

The proposal of an inverse kinematics program and LabVIEW using NI-SoftMotion module as a controller has the following advantages:

1. This control technique can be applied to an inverse kinematic problem, which is the case of the SCARA robot manipulator and can be applied for other manipulators.
2. The complete analysis was done in a New Geometrical Developed Approach method.
3. It can control most of the robot manipulator systems with knowing their possible solution, which can be found.
4. The main advantage of this method is that it does not require previous knowledge of the robot dynamics.
5. The proposed control program gives to mechatronics and mechanical design engineers all possible trajectory and can select the suitable trace and can be saved a power of manipulator actuators by select a short path.

C. The parts of the block diagram are mentioned as follows

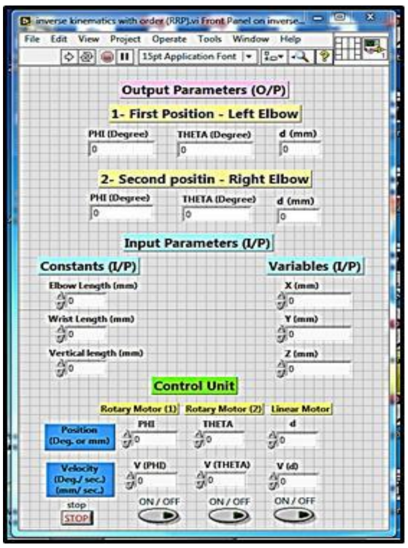
Input Parameters – includes two types: the first is constant as manipulator dimensions (L, El and Wr), and the second is variables as Cartesian Coordinates of end-effector/ tool (x, y and z).

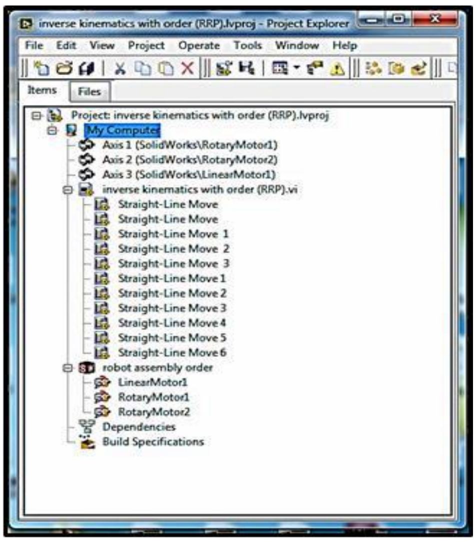
Output Parameters – includes the a rotational joint angles and a linear displacement (ϕ,θ and d).

ON/OFF – Controls for the Line Move Function input of motors.

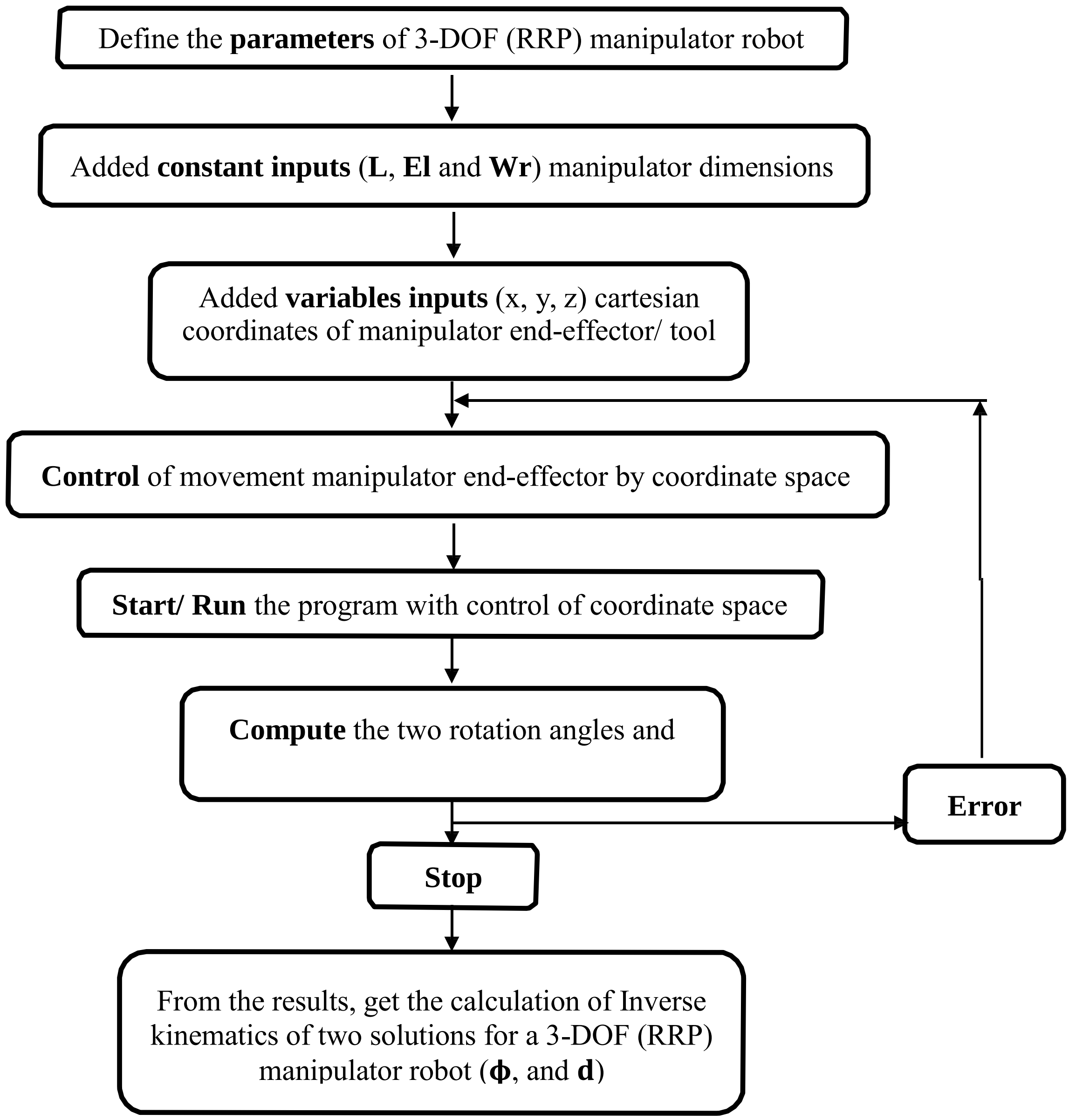
Stop – To exit the loop and stop motion control.

Interface programs LabVIEW and SolidWorks of inverse kinematics and trajectory control of SCARA robot is as shown below.



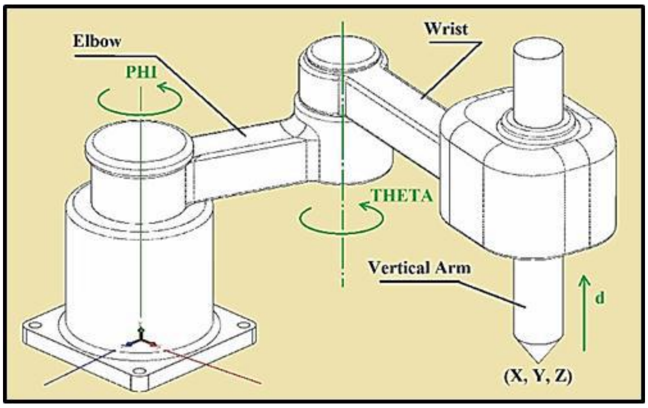


Algorithm of Inverse Kinematics Programming Design and Control of 3-DOF (RRP) Manipulator robot Using LabVIEW and SolidWorks is as shown below.



SCARA Manipulator robot

The isometric of 3-DOF (RRP) manipulator robot with three moving axes (such as exist in SCARA Robot) is as shown below.



In this study, we defined the 3-DOF (RRP) SCARA manipulator robot parameters as the following:

x: a displacement of end-effector manipulator along x-axis from the rotation point.

y: a displacement of end-effector manipulator along y-axis from the rotation point.

z: a displacement of end-effector manipulator along z-axis from the rotation point.

L: a vertical arm manipulator length.

El: an elbow manipulator length.

Wr: a wrist manipulator length.

L_1: a distance between the reference point (0, 0, 0) and end-effector point (x, y, z), which measured from top plane of a manipulator.

ϕ_2: an angle between L1 and z-axis ,which measured from top plane of a manipulator.

ϕ_3: an angle between L1 and El ,which measured from top plane of a manipulator.

ϕ: an angle between z-axis and El (rotary motor angle of an elbow) ,which measured from top plane of a manipulator.

θ: an angle between Wr and El (rotary motor angle of a wrist) ,which measured from front plane of a manipulator.

d: a linear vertical displacement of L (linear motor displacement of a vertical arm L), which measured from front plane of a manipulator.

Manipulator Robot Parameters Constraints:

1- $L > 0$

2- $El > 0$

3- $Wr > 0$

4- $Wr > El$

Manipulator Robot WorkSpace Constraints:

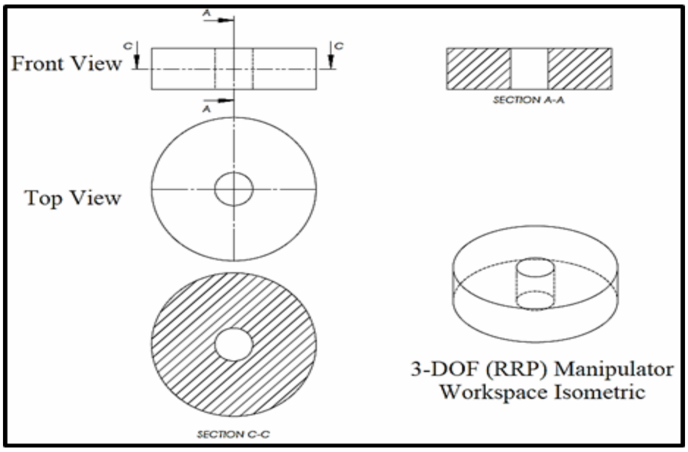
1- $\sqrt{x^2 + z^2} \geq (Wr - El)$

2- $\sqrt{x^2 + z^2} \leq (Wr + El)$

3- $0 \leq d \leq L$

The 3-DOF (RRP) robotic arm's manipulator Workspace

In performing tasks, a manipulator has to reach a number of workpieces or fixtures. In some cases, these can be positioned as needed to suit the workspace of the manipulator. In other cases, a robot can be installed in a fixed environment with rigid workspace requirements [1]. Workspace is also sometimes called work volume or work envelope is as shown below.



What are the limits for the 3-DOF (RRP) robotic arm's manipulator?

The achievable working space looks like a vertical hollow cylinder, where:

1- The maximum outer diameter will be:

$2(Wr + El)$

2- The minimum inner diameter will be:

$2(Wr - El)$

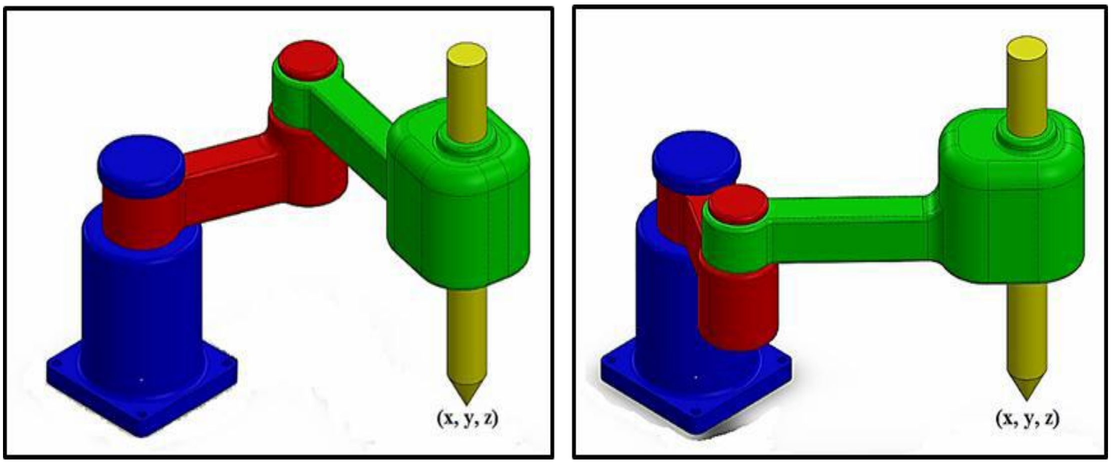
3- The maximum height of vertical hollow cylinder will be:

L

Possible Solutions for an Inverse Kinematics of 3-DOF (RRP) Manipulator Robot

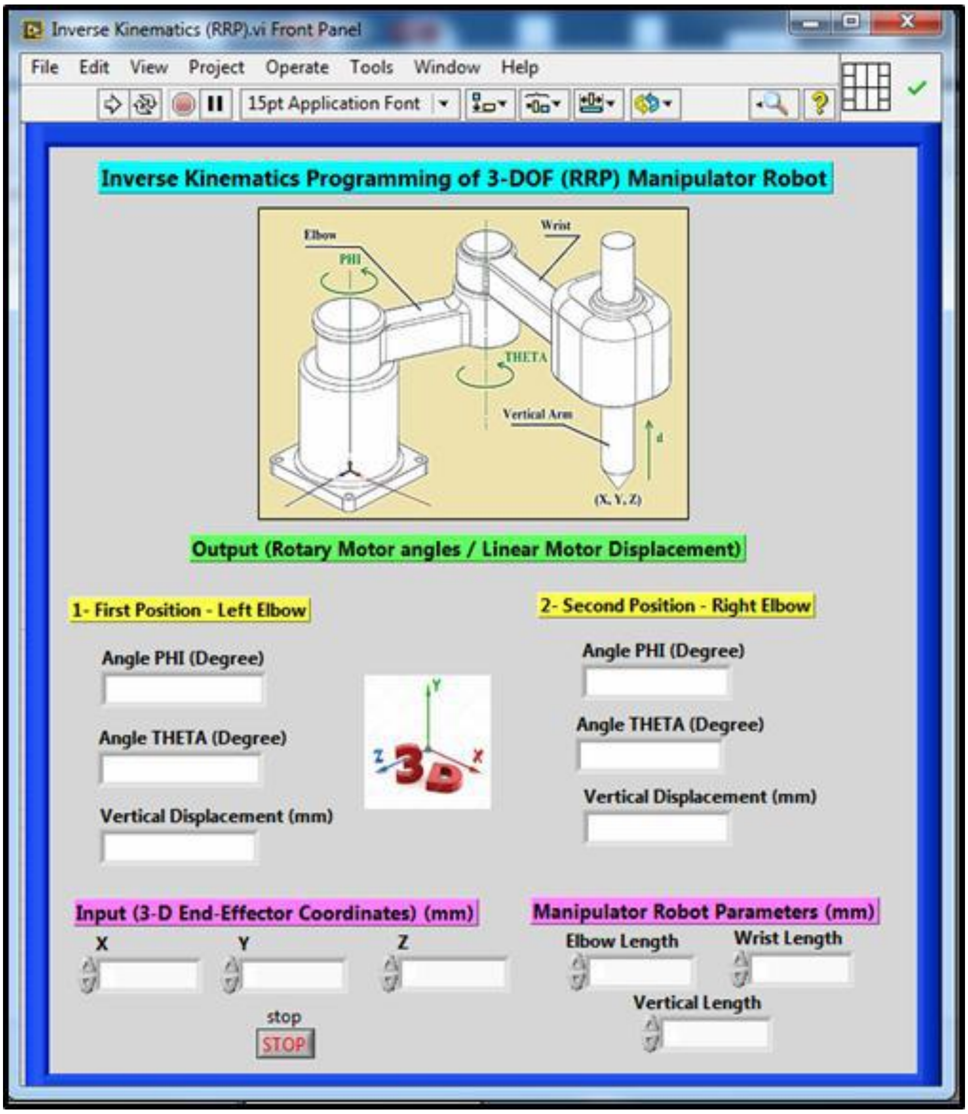
The two possible position solutions of 3-DOF (RRP) manipulator robot is as shown below:

1- First Position – Left Elbow. 2- Second Position – Right Elbow.



- There are two possible solutions to reach the manipulator end-effector/ tool to the purposed point in (x, y, z), if this point was located inside the manipulator workspace (inside the vertical hollow cylinder).
- There are one possible solution to reach the manipulator end-effector/ tool to the purposed point in (x, y, z), if this point was located in the outlines profile of workspace the manipulator workspace frame (in the outlines of a vertical hollow cylinder).
- There are no possible solution, if a purposed point in (x, y, z) was located outside of the manipulator workspace (outside the vertical hollow cylinder).
- In this study, both forward and inverse kinematics models are derived for the SCARA robot with a new geometrical approach method is realized in LabVIEW program. In figure 5, the designed graphical user interface of the SCARA robot is given. In the designed interface, required parameters are determined and calculated for each joint.

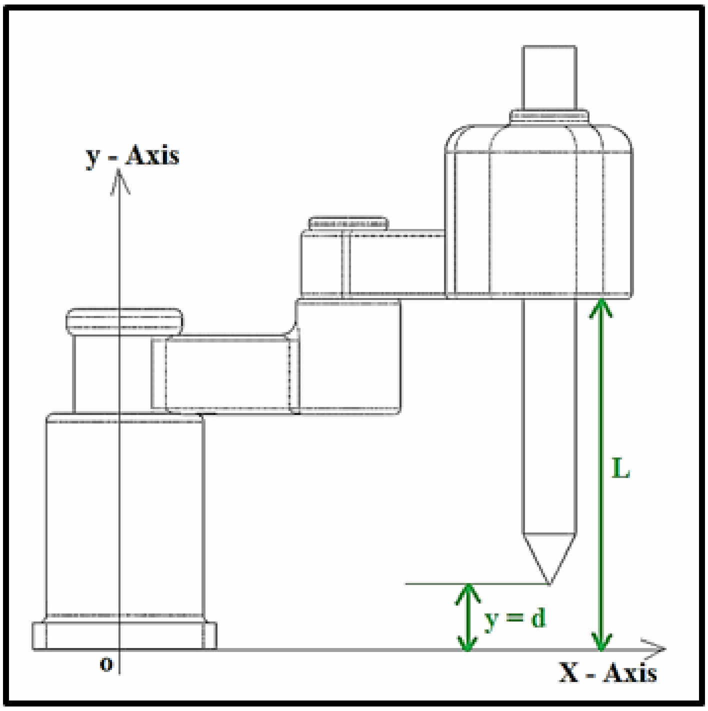
Interface program of forward and inverse kinematics SCARA robot is as shown below.



Mathematical Model Programming by a New Geometrical Developed Approach of an Inverse Kinematics Study (IK)

$$L_1 = \sqrt{x^2 + z^2}\ ,\ \phi_2 = \cos^{-1}\left(\frac{z}{L_1}\right),\ \phi_3 = \cos^{-1}\left(\frac{L_1{}^2 + El^2 - Wr^2}{2 \times L_1 \times El}\right),\ \theta = \cos^{-1}\left(\frac{Wr^2 + El^2 - L_1{}^2}{2 \times Wr \times El}\right),\ \ d = y$$

Front view of left elbow for a 3-DOF (RRP) Manipulator robot is as shown below.



i- First Position: Left Elbow (LE)

1- When (x > 0) and (z ≥ 0), then:

$$\phi = (\phi_2 + \phi_3)$$

2- When (x ≥0) and (z < 0), then:

$$\phi_2 = \cos^{-1}\left(\frac{z}{L_1}\right)$$

$$\phi = (\phi_2 + \phi_3)$$

3- When (x < 0) and (z ≤ 0), then:

$$\phi_2 = \sin^{-1}\left(\frac{x}{L_1}\right)$$

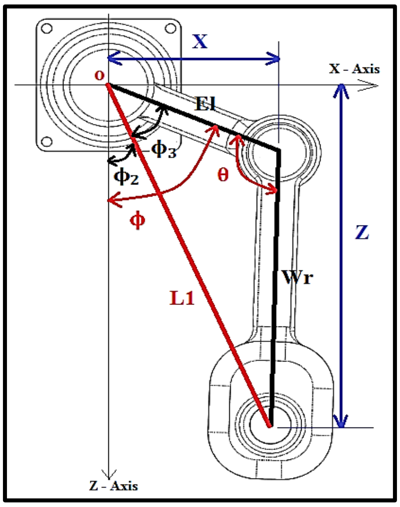
$$\phi = 180 + (\phi_2 + \phi_3)$$

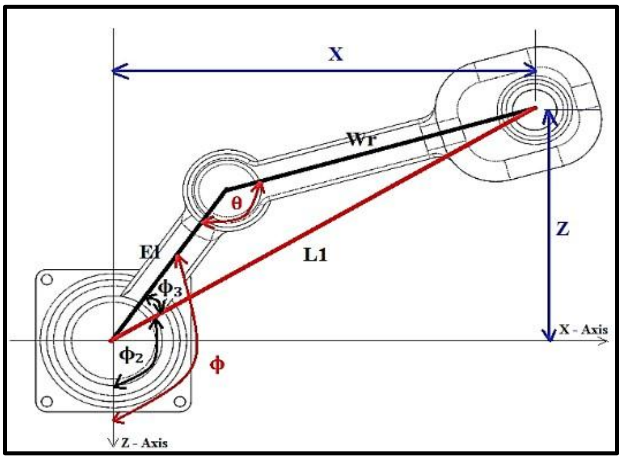
4- When (x ≤ 0) and (z > 0), then:

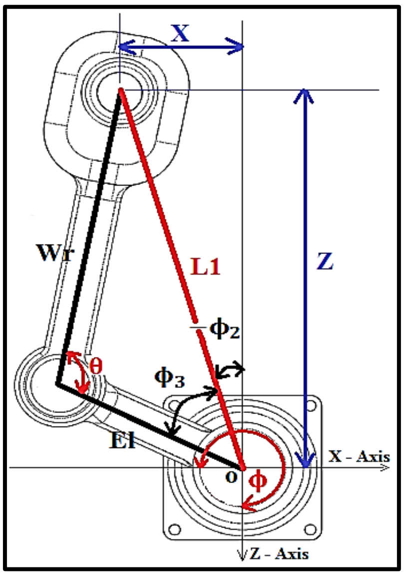
$$\phi_2 = \sin^{-1}\left(\frac{x}{L_1}\right)$$

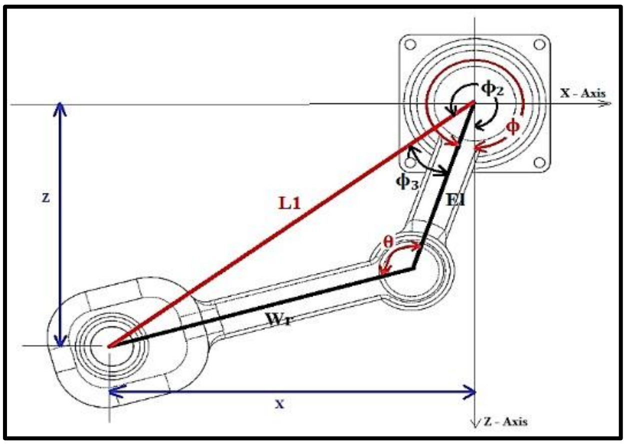
$$\phi = (\phi_2 + \phi_3)$$

Top view of left elbow when (x> 0), (z ≥ 0) && (x ≥ 0), (z <0) && (x <0), (z ≤ 0) and (x ≤ 0), (z> 0) is as shown below.





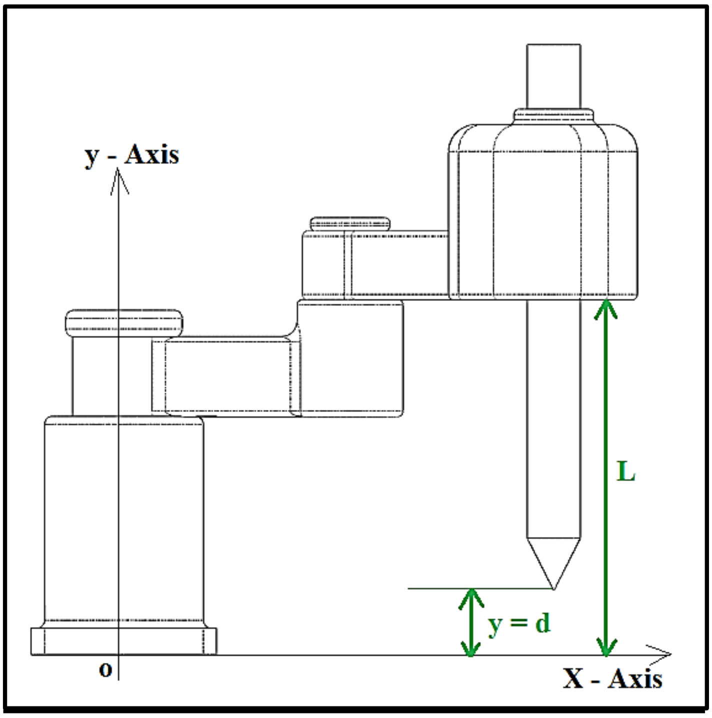




ii- Second Position: Right Elbow (RE)

$$\theta = 360 - \cos^{-1}\left(\frac{Wr^2 + El^2 - L_1^2}{2 \times Wr \times El}\right)$$

Front view of right elbow for a 3-DOF (RRP) is as shown below.



1- When (x > 0) and (z ≥ 0), then:

$$\phi = 360 - (\phi_3 - \phi_2)$$

2- When (x ≥0) and (z < 0), then:

$$\phi_2 = \cos^{-1}\left(\frac{z}{L_1}\right)$$

$$\phi = (\phi_2 - \phi_3)$$

3- When (x < 0) and (z ≤ 0), then:

$$\phi_2 = \sin^{-1}\left(\frac{x}{L_1}\right)$$

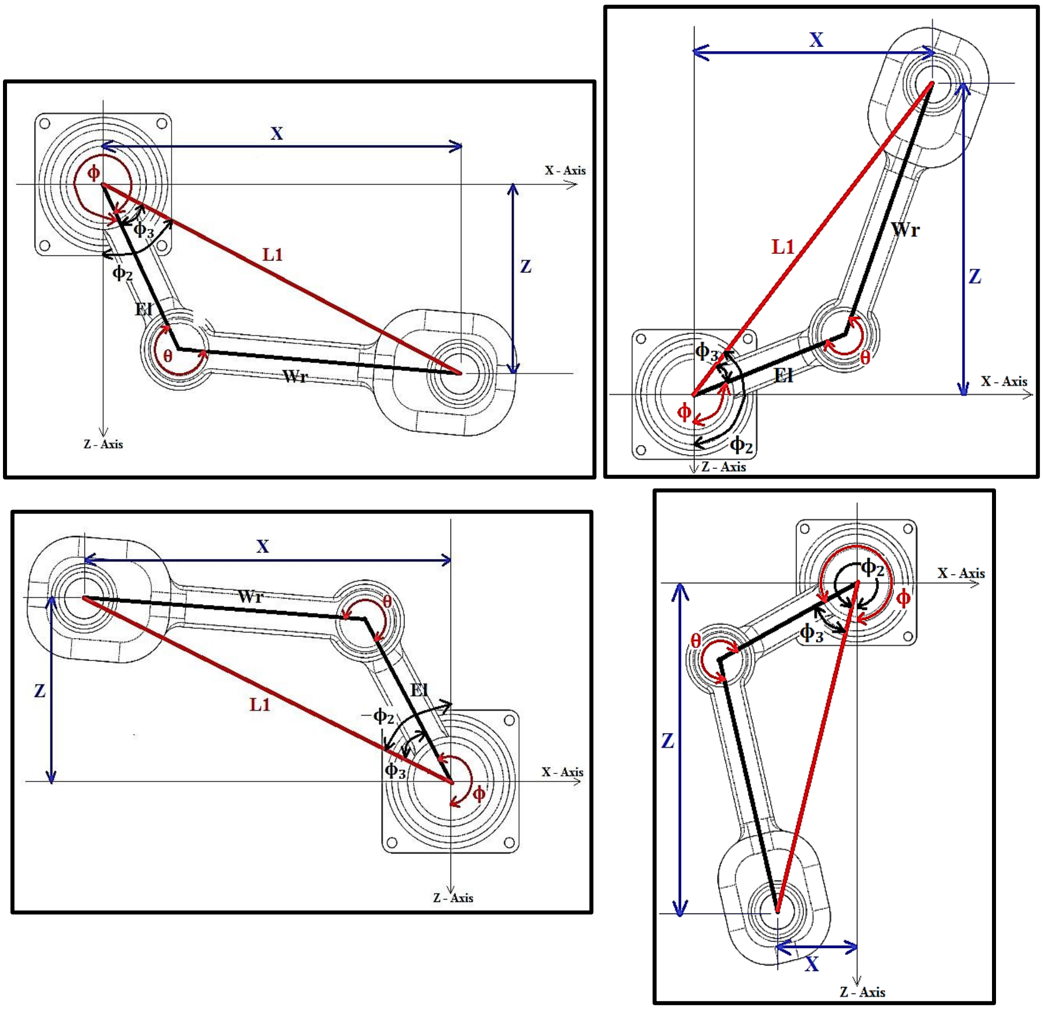
$$\phi = 360 - (180 + (\phi_2 + \phi_3))$$

4- When (x ≤ 0) and (z > 0), then:

$$\phi_2 = \sin^{-1}\left(\frac{x}{L_1}\right)$$

$$\phi = 360 + (\phi_2 - \phi_3)$$

Top view of right elbow when (x> 0), (z ≥ 0) && (x ≥ 0), (z <0) && (x <0), (z ≤ 0) and (x ≤ 0), (z > 0) is as shown below.



Trajectory Control of an End-Effector SCARA robot

We can get different trajectories for reaching to the same point, which is demanded, by control with order arrangement of motors rotation first, second, etc. and this has been done according to which a trajectory is suitable with different specific applications to avoid a contact and stick with anybody exist in a manipulator workspace. Therefore, this makes a designer has alternative and accessible solutions to reach the best possible motion path (trajectory). All possible trajectory control of SCARA end-effector robot is presented in Table 1.

Table 1: All possible trajectory control of SCARA end-effector robot.

O/P position of end-effector	Order of rotary motor (ϕ)	Order of rotary motor (θ)
First Position – Left Elbow	(1) + ϕ	(2) + θ
	(2) + ϕ	(1) + θ
	(1) + ϕ -360	(2) + θ-360
	(2) + ϕ -360	(1) + θ-360
	(1) + ϕ	(2) + θ-360
	(2) + ϕ	(1) + θ-360
	(1) + ϕ -360	(2) + θ
	(2) + ϕ -360	(1) + θ
Second Position – Right Elbow	(1) + ϕ	(2) + θ
	(2) + ϕ	(1) + θ
	(1) + ϕ -360	(2) + θ-360
	(2) + ϕ -360	(1) + θ-360
	(1) + ϕ	(2) + θ-360
	(2) + ϕ	(1) + θ-360
	(1) + ϕ -360	(2) + θ
	(2) + ϕ -360	(1) + θ

Case Study

In this study, we used the 3DOF rotational manipulator robot input parameters as in the following table:

Table 2: Input parameters of SCARA Robot inverse kinematics and control program.

Manipulator Dimensions (mm)	**X**	**Y**	**Z**
Elbow = 450 Wrist = 720 Vertical Arm = 670	$X_0 = 0$ $X_f = 500$	$Y_0 = 0$ $Y_f = 300$	$Z_0 = -270$ $Z_f = 600$

After make a run of a program with the last input data, we get the output results as in the following table:

Table 3: Output results of SCARA Robot inverse kinematics and control program.

O/P position of end-effector	Trajectory Graph Number	Order of (ϕ) degree	Order of (θ) degree	Trajectory Length (mm)
First Position – Left Elbow	6	(1) +105.1	(2) +80.1	1502.3
	9	(2) +105.1	(1) +80.1	2439.4
	2	(1) -254.9	(2) -279.9	4718.1
	13	(2) -254.9	(1) -279.9	6991.8
	5	(1) +105.1	(2) -279.9	4011.9
	15	(2) +105.1	(1) -279.9	4949.0
	7	(1) -254.9	(2) +80.1	2208.5
	10	(2) -254.9	(1) +80.1	4482.2
Second Position – Right Elbow	1	(1) +334.5	(2) +279.9	5093.2
	12	(2) +334.5	(1) +279.9	8077.0
	3	(1) -25.5	(2) -80.1	1127.1
	11	(2) -25.5	(1) -80.1	1354.2
	8	(1) +334.5	(2) -80.1	2583.6
	16	(2) +334.5	(1) -80.1	5567.4
	4	(1) -25.5	(2) +279.9	3636.7
	14	(2) -25.5	(1) +279.9	3863.8

RESULTS AND DISCUSSION

From the inverse kinematics and control program of SCARA robot, we can control of the order and arrangement of rotary motors 1, 2 (ϕ, θ) in both of O/P out-put positions: First Position – Left Elbow and Second Position – Right Elbow. Therefore, we get 16- possible trajectory path of the end effector when moving from point to another as the following drawings in the last pages.

The possible trajectories of output results program we can choose the optimal path of them by select the minimum value of trajectory length (shorter path), which achieve the saving in working time and energy using so, the optimal trajectories are:

1- The first optimal trajectory is number 3 with minimum path length of 1127.1 mm.

2- The second optimal trajectory is number 11 with path length of 1354.2 mm.

FUNDING

The work is executed at support of the project 15.8792.2017/8.9 "Development of optical methods for monitoring the status of biotech facilities and the creation of the basis of their automation systems".

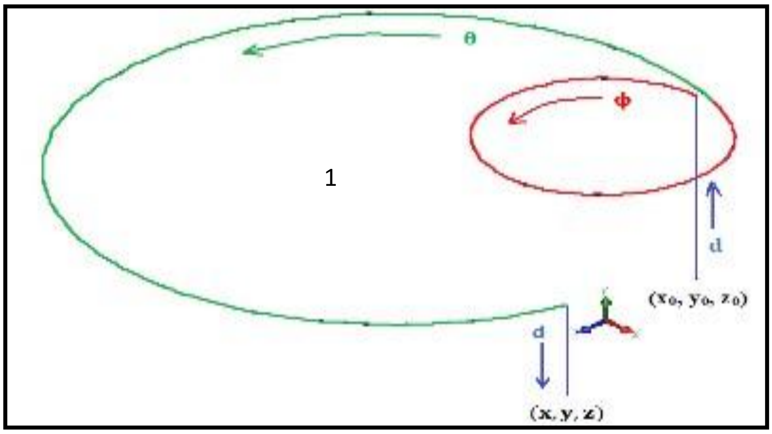
CONCLUSION

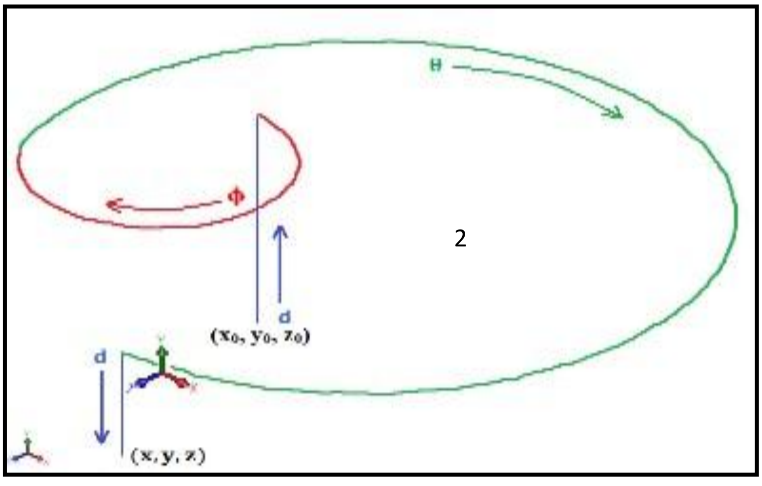
This program interface of inverse kinematics can be used for any three Degree of Freedom as Revolute-Revolute-Prismatic joints 3-DOF (RRP) Manipulator robot (such as exist in SCARA Robot), with any dimension lengths.

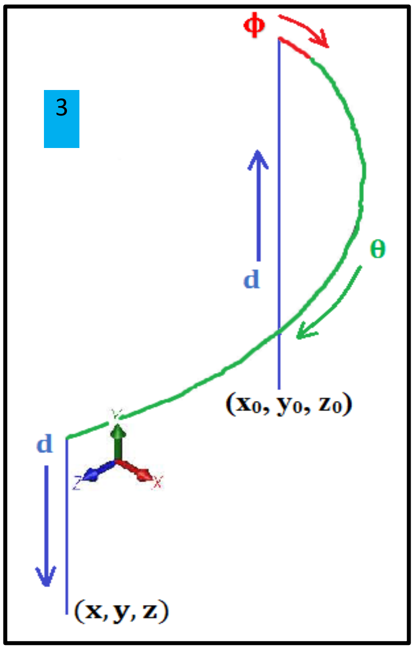
Manipulator trajectory can be controlled by order arrangement of motors rotation to avoid a contact and stick with anybody exist in a manipulator workspace.

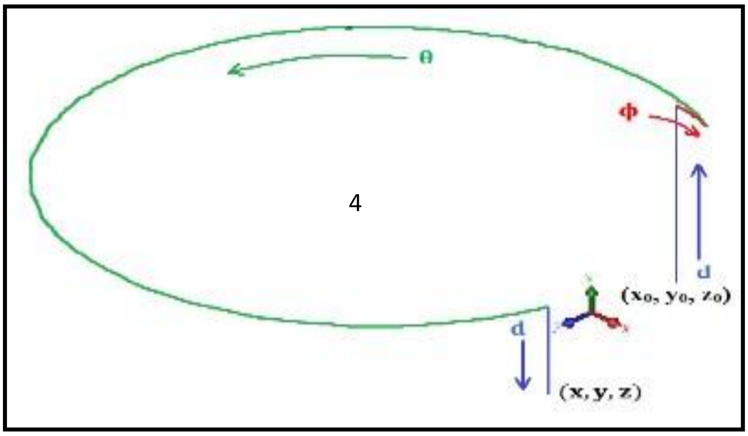
Choosing the optimal path from all possible trajectories of output results program by select the minimum value of trajectory length (shorter path), which achieve the saving in working time and energy using.

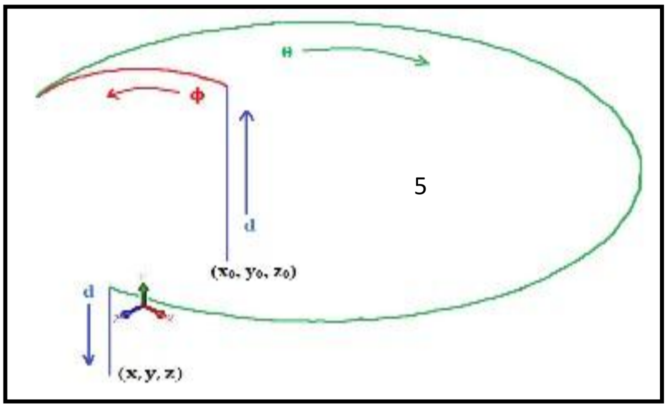
Finally, this program saves a lot of time in calculation with respect to another calculations method such as numerical and algebraic methods and gives us an accurate result.

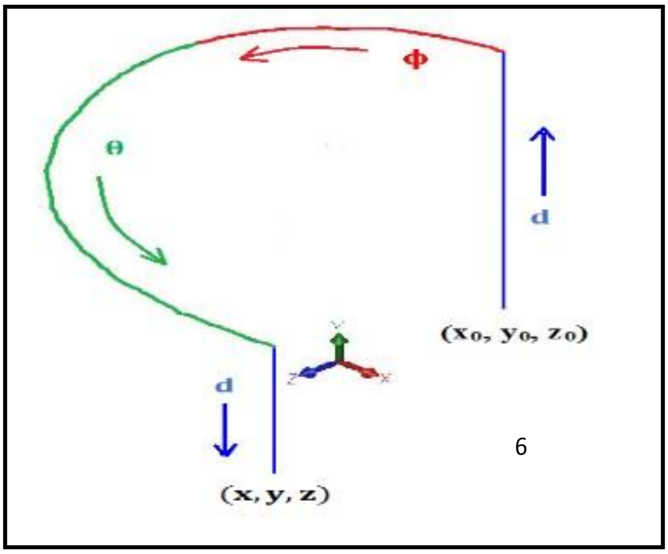


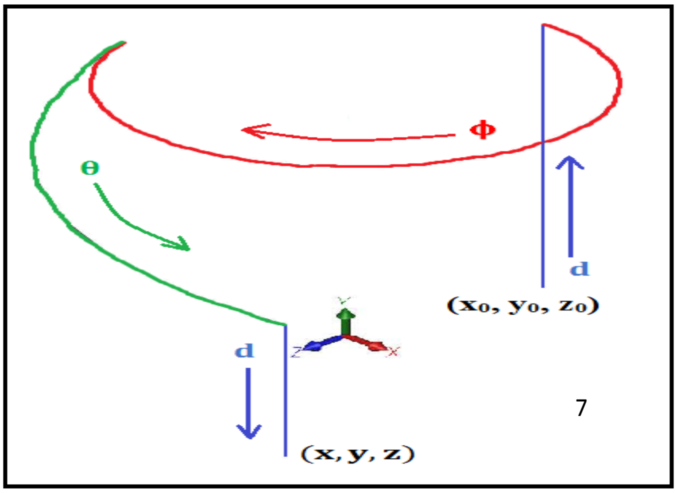


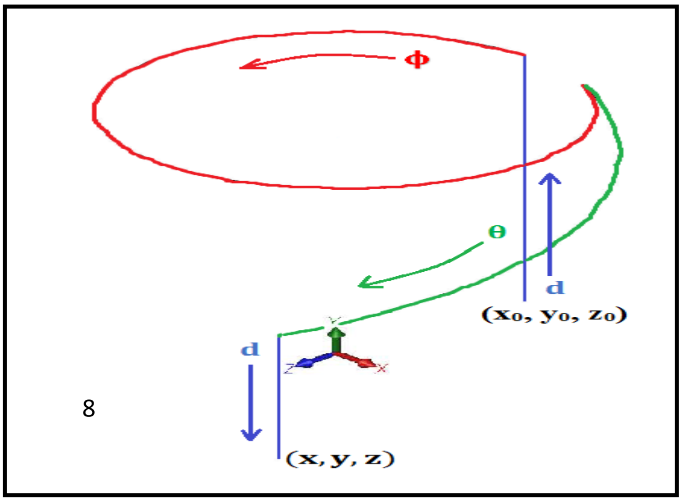

3
ϕ
θ
d
(x_0, y_0, z_0)
d
(x, y, z)

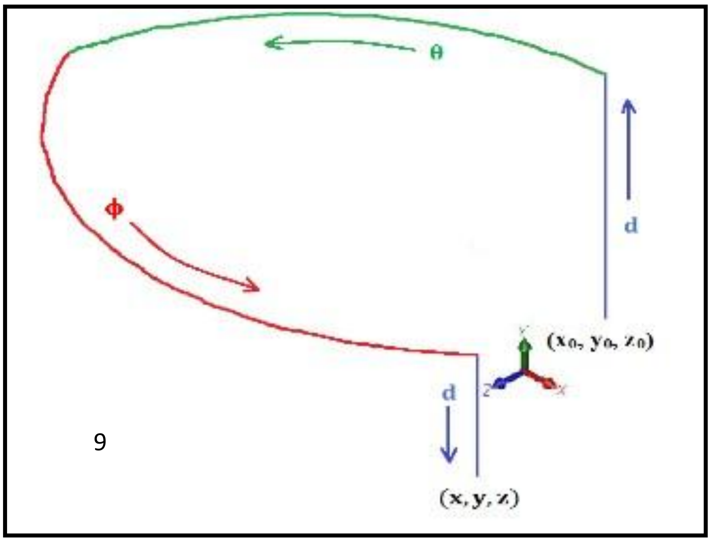

θ
ϕ
4
d
(x_0, y_0, z_0)
d
(x, y, z)

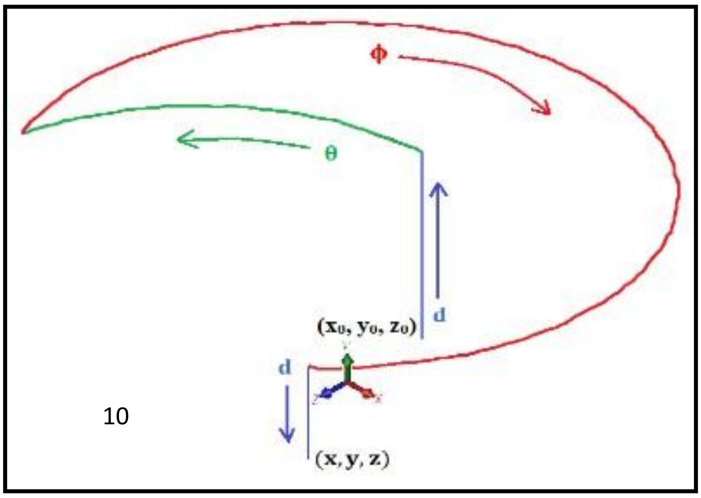

θ
ϕ
5
d
(x_0, y_0, z_0)
d
(x, y, z)

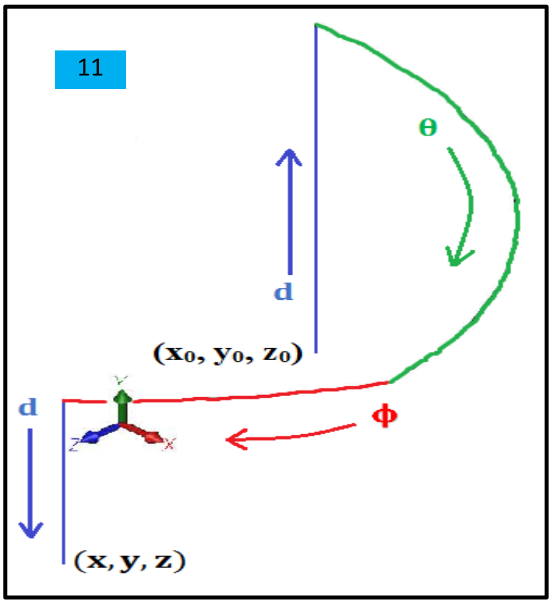

Φ
θ
d
(x0, y0, z0)
d
(x, y, z)
6

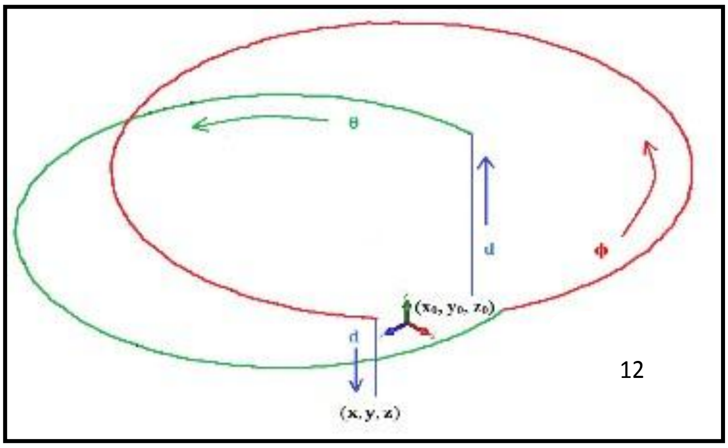

Φ
θ
d
(x0, y0, z0)
d
(x, y, z)
7

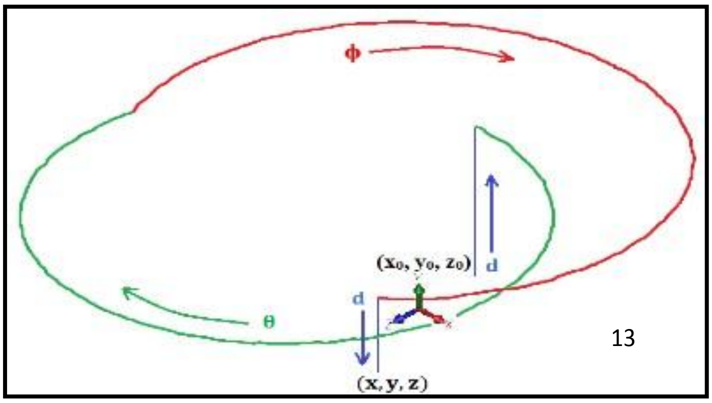

Φ
θ
d
(x0, y0, z0)
d
8
(x, y, z)

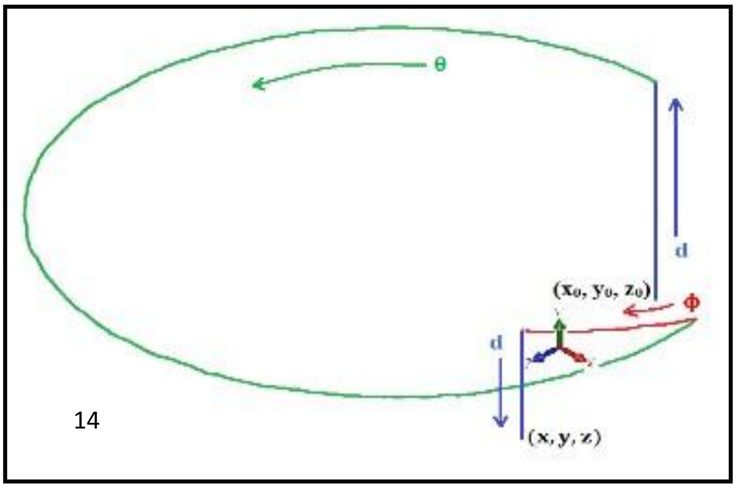

θ
d
ϕ
(x0, y0, z0)
d
9
(x, y, z)

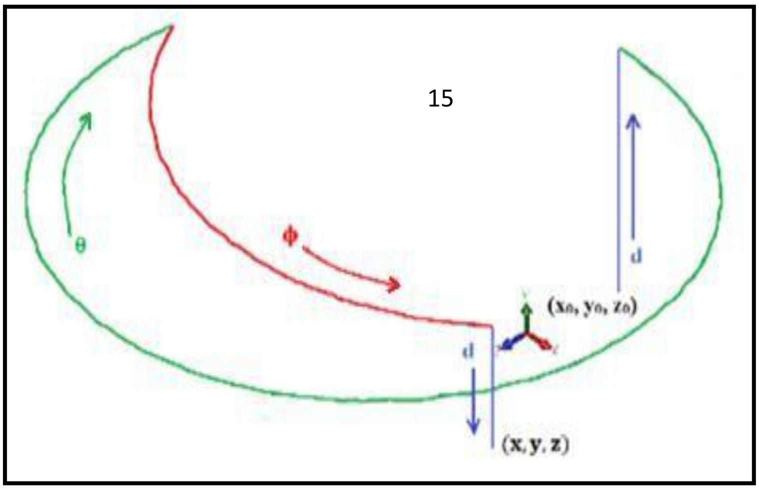

ϕ
θ
d
(x0, y0, z0)
d
10
(x, y, z)

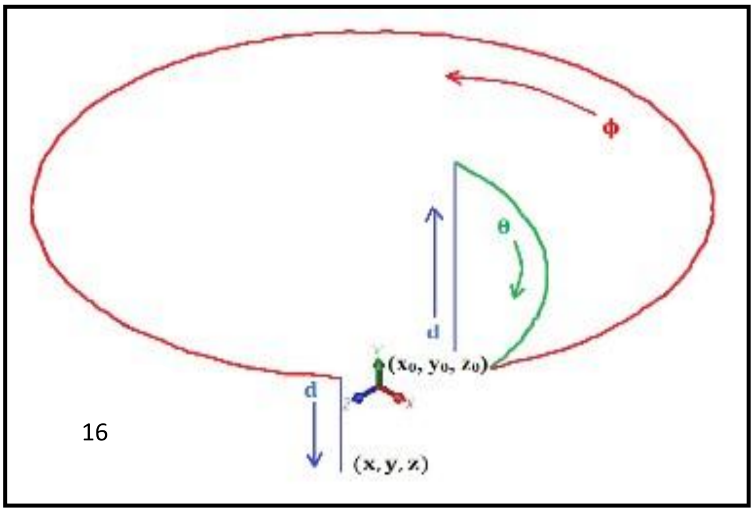

11
θ
d
(x_0, y_0, z_0)
d
Φ
(x, y, z)


θ
d
φ
(x_0, y_0, z_0)
d
(x, y, z)
12


φ
(x_0, y_0, z_0)
d
d
θ
(x, y, z)
13


θ
d
(x_0, y_0, z_0)
ϕ
d
(x, y, z)
14


15
θ
ϕ
d
(x_0, y_0, z_0)
d
(x, y, z)


ϕ
θ
d
(x_0, y_0, z_0)
d
(x, y, z)
16

REFERENCES

[1] de Wit, CC., Siciliano, B., and Bastin, G., 1996, "Theory of robot control Hand Book," Springer, New York.

[2] Lewis, F., Abdallah, C., and Dawson, D., 1993, "Control of robot manipulators," MacMillan Publishing Co., New York.

[3] Samson, C., and Espiau, B., 1990, "Application of the task-function approach to sensor-based control of robot manipulators," I FAC 11th Triennial World, pp. 269-274.

[4] Spong, M., 1996, "Motion control of robot manipulator," Levine Handbook control, CRC Press., pp. 1339–1350.

[5] Khalil, W., Dombre, E., 2002, "Modeling identification and control of robots," Hermes Penton Science, London.

[6] Jiang, Z-H., Ishida, T., 2007, "Trajectory tracking control of industrial robot manipulators using a neural network controller," IEEE international conference on systems, man and cybernetics ISIC. pp. 2390–2395.

[7] Khalal, O., Mellit, A., Rahim, M., Salhi, H., Guessoum, A., 2007, "Robust control of manipulator robot by using the variable structure control with sliding mode," Mediterranean conference on control and automation, 07, pp.1–6.

[8] Rossomando, F. G., Soria, C. M., 2017, "Discrete-time sliding mode neuro-adaptive controller for SCARA robot arm," Neural Comput & Applic., 28, pp.3837–3850.

[9] Craig, J., 2005, "Introduction to Robotics Mechanics and Control,"Edition Pearson Education.

[10] Tingjun, W., Shenshun, HU., Jun XU., Dewei, YAN, and Jiawen, BI., 2009, "Simulation design and application of music playing robot based On SolidWorks," IEEE Measuring Technology and Mechatronics Automation.