Proceedings of the Fifth International Conference on Computing Methodologies and Communication (ICCMC 2021)
IEEE Xplore Part Number: CFP21K25-ART

Machine Learning approach for Inverse Kinematics in Trajectory Planning of Pioneer 2 Manipulator with Cubic Spline Interpolation

M Arunadevi
Professor, Mechanical Engineering
DSCE,Bangalore.
arunadevi.dsce@gmail.com

Praveen D Jadhav
Research Scholar, Mechanical Engineering,
DSCE,Bangalore.
Praveendjadhav1@gmail.com

Ruchik.M.A
Student, Mechanical Engineering
DSCE.Bangalore.
ruchik_m@yahoo.com

Kishore Gandhi.P
Student, Mechanical Engineering
DSCE,Bangalore.
p.kishoregandhi@gmail.com

Siddu.K.V
Student, Mechanical Engineering
DSCE,Bangalore.
siddukvmonster100@gmail.com

Vishnu Prasad.S
Student, Mechanical Engineering
DSCE,Bangalore.
vishnuinfinite5@gmail.com

2021 5th International Conference on Computing Methodologies and Communication (ICCMC) | 978-1-6654-0360-3/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICCMC51019.2021.9418318

***Abstract*— The primary objective of robot manipulators is to achieve the desired orientation and point of end effector in order to accomplish the pre-established task. Inverse kinematic analysis will be used in the pioneer 2 robot to obtain a successful solution to design and operate the arm. This paper considers a 5-dof revolute Pioneer2 manipulator which is compact, low cost and lightweight. When the DOF of the robot increases, the inverse kinematic problem becomes more and more complex and gives n number of joint configurations for the same position. This results in making the standard solution for this problem becomes trickier. To overcome the computational complexity of kinematic analysis of Pioneer 2 robot, the objective of this study is to perform intelligent computation of inverse kinematics with the use of machine learning techniques that consists of linear regression, K-Nearest Neighbor algorithm and Artificial Neural Network. By comparing three algorithms R -square values and RMSE values, it is observed that KNN algorithm is giving better results. Therefore, KNN can be used for better solution of inverse kinematics with fast results and high accuracy. Then the smooth trajectory is achieved using cubic spline interpolation.**

Keywords—Inverse Kinematics, KNN, ANN, Linear Regression, Pioneer2 Manipulator, Machine Learning, Cubic Spline Interpolation

I. INTRODUCTION

Industrial robots or robot manipulators are automated machines used in industries for process which require high precision, flexibility and repeatability that manual labor cannot provide. This manipulator control is the one of the main research area in robotics. Forward kinematics talks about the position of end effector in Cartesian space from particular values of the joint parameters using kinematic equations used for a robotic calculation. Inverse kinematics makes use of a given orientation and joint variable in Cartesian space to find the joint variables in joint space.

In this research paper [1], the author explained about the industrial robots which are used in the sector where the manpower is can't be worked. As many researchers have proposed to optimize the errors in different DOFs. So, the author used some technique to solve these errors that are ANFIS, PSO, DE. By optimizing the working function of robot then there will be accurate decision making by the robot. In [2], Due to its high accuracy and good precision of robots. These robots are used in major sector like medical, military and industrial application. This paper talks about the improvement of kinematics in robot and robots dynamics. The paper presented research and differentiate between joint space and the Cartesian space in robot kinematics. In [3] they presented about BQGA which is the optimization algorithm to represent population element. Hence, this paper proposes an improvement plan for these problems in BQGA. In [4] the author presented the planning approach which is an energy based trajectory using machine learning for IRs. They showed in-terms of method architecture, robotics movement digitalization, hybrid algorithm and optimization process. In this research paper [5], they presented the reviews and discussions about S-Curve. Trajectory with Sigmoid Jerk S-Curve is described, simulation on manipulator with three and six DOFs are addressed. In this paper [6], the prediction of optimal joint angels of the robot manipulator with the help of neural network predictors are used. The experiment implemented using KUKA, KR and details about industrial implementation, robot manipulator, and predictor for the neural network. In this paper [7], the main aim is to reduce the mean square error of the neural network-based solution of inverse kinematic problem using GSA. The solutions of both the forward and inverse kinematic problems for 6R PUMA robot are solved. In [8] the author presented about optimization of algorithm and scenarios in the robot with the help of inverse kinematics equations is described. The manipulated result is compared with the graph and tables. The working function of 5-DOF pioneer is explained in [9] which includes inverse kinematics with the help of forward kinematics. In [10] the authors prove that correct kinematic analysis helps in optimizing the design of the robot and guiding the movement of joints for effective functioning. An analytical solution for the inverse kinematics of a robotic arm is derived with five degrees of freedom. In [11], the difficulties in solving the Inverse kinematic equations of 2, 3, 5 DOF redundant robot manipulators arises due to uncertainty, varying time and non-linear equations with transcendental functions are explained. In [12], the modeling and control of Pioneer 2 robotic arms are discussed and explained about the ineffective solutions of robotic manipulators with high degrees of freedom. In [13], authors discussed inverse kinematics which can be solved by methods such as iterative, DH notation and transformation equation. In [14], the different machine learning techniques such as Linear regression, Artificial Neural network and K- Nearest Neighbor algorithm are used for the prediction of properties in manufacturing for the type of supervised regression problems. In [15, 16] the kinematic analysis of 5- DOF robot manipulator designed for serving various applications are explained. A pioneer 2 robot which is a 5DOF robot have a list of advantages which include adjustable action, suitable operation, compact volume etc. So, in the process of making Pioneer 2 robot more precise, different algorithms like KNN, ANN and linear regression are proposed for the kinematic analysis of the same robot manipulator in this paper. In [17,18], the different applications of robot manipulator like developing a robotic arm for playing chess and sensing control in manufacturing process of automobile industry is discussed.

II. Problem Description

This paper considers a 5-dof revolute Pioneer2 manipulator which is compact, low cost and lightweight. If the degrees of freedom of manipulator is high (3 and above) or redundant, than inverse kinematics solutions will become more complex and gives n number of joint configurations for the same position.

A. *Working Environment*

The structure of 5-dof Pioneer2 manipulator with revolute joint is shown in Figure 1.

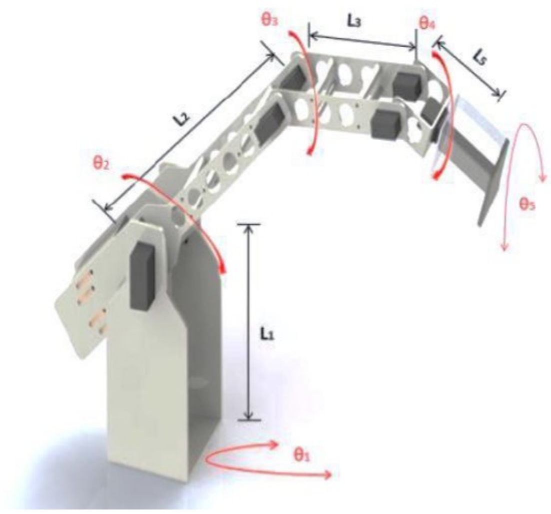

Fig. 1. Structure of 5-dof revolute Pioneer2 manipulator

This robot is mainly used for grasping and manipulation of light weight objects like soda cans with the weight limit of 150grams. Different joints and movements are

- Joints rotation
- Base rotation
- Shoulder rotation
- Elbow rotation
- Wrist rotation
- Gripper mount
- Gripper fingers

Servo motors are used to drive all the joints except gripper fingers.

B. *Objective*

To overcome the computational complexity of kinematic analysis of Pioneer 2 robot, the objective of this study is to perform intelligent computation of inverse kinematics with the help of machine learning algorithms such as linear regression, K-Nearest Neighbor algorithm and Artificial Neural Network.

III. Methodology

In this study, the data is generated using forward kinematics equations of a Pioneer 2 robot. Modelling of different machine learning algorithms is done using the generated data. Then the joint variables can be predicted using the machine learning models. Finally the accuracy of algorithm is compared and best algorithm will be selected for the future predictions. The same is explained using the flowchart which is given Figure 2.

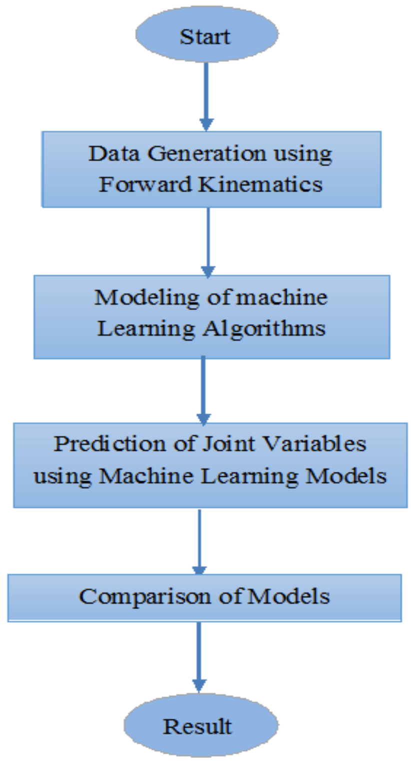

Fig. 2. Flow chart for methodology

A. *Mathematical Modeling Of Pioneer 2 Robot*

Different joint variable value ranges and constants are defined to find forward and inverse kinematics solutions of Pioneer 2 robot is shown in Table 2.

TABLE I.

DATA SET GENERATED USING FORWARD KINEMATICS

Sl.No	θ_1(deg)	θ_2(deg)	θ_3(deg)	θ_4(deg)	θ_5(deg)	X(m)	Y(m)	Z(m)
1	-180	-180	-180	-180	-180	-190	9.18861E-15	150
2	-179.64	-179.64	-179.64	-179.64	-179.64	-190.0011518	0.476383123	150.9110546
3	-179.28	-179.28	-179.28	-179.28	-179.28	-190.0044746	0.963099826	151.8220659
..	..	..	..	..	..	..	..	..
..	..	..	..	..	..	..	..	..
998	179.28	179.28	179.28	179.28	179.28	-190.005	-0.92205416	148.1779
999	179.64	179.64	179.64	179.64	179.64	-190.001	-0.46611948	149.0889
1000	180	180	180	180	180	-190	1.83428E-11	150

TABLE II.

JOINT VARIABLES OF PIONEER 2 ROBOT

Joints	θ_i (degree)	d_i(mm)	a_i(mm)	α_i(degree)
0	$\theta_1=\pm180$	$d_1=150$	$a_1=60$	-90
1	$\theta_2=\pm180$	0	$a_2=145$	0
2	$\theta_3=\pm180$	0	0	-90
3	$\theta_4=\pm180$	$d_2=125$	0	90
4	$\theta_5=\pm180$	0	0	-90
5	0	$d_3=130$	0	0

The data sets for training the algorithms were generated by using forward kinematic equations as follows

$$X = -d_3c_1s_{23}c_4s_5 - d_3s_1s_4s_5 + d_3c_1c_{23}c_5 + d_2c_2c_{23} + a_2c_1c_2 + a_1c_1$$

$$Y = -d_3s_1s_{23}c_4s_5 + d_3c_1s_4s_5 + d_3s_1c_{23}c_5 + d_2s_1c_{23} + a_2s_1c_2 + a_1s_1$$

$$Z = -d_3c_{23}c_4s_5 - d_3s_{23}c_5 - d_2s_{23}c_5 - d_2s_{23} - a_2s_2 + d_1$$

Where, a_i= Link Length
d_i = Joint Distance
$c_i = \cos\theta_i$, (i=1,2,3...n)
$s_i = \sin\theta_i$,, (i=1,2,3...n)

The training data set is generated using the above formula and the input values X, Y and Z coordinates are calculated by using the different values of θ1,θ2,θ3,θ4 and θ5 which is shown in table 2. The above data sets are utilized for training of algorithm, testing and validation of different algorithms like the Linear Regression, KNN and MLP neural network. This modeling, testing and predictions are performed using python. There are three steps are involved in any data science algorithm which follows

1. Data Processing
2. Modeling
3. Validation

A. Data Processing

Data processing is a very important step in data science because the accuracy and performance of any model depends on data only. Data processing will be performed in two steps:

a. Data selection
b. Feature engineering.

1) Data Selection: Method of data selection varies based on application, in this study, a set of 1000 data were generated using forward kinematics model, which includes joint variables and cartesian variables (X,Y,Z Coordinates) which is presented in table1.

2) Feature Engineering: It is the method of selecting the suitable input parameters for the model. In this paper, cartisian variables(X, Y and Z) are selected as features for the Joint variables prediction.

B. Modeling

Modeling is creating a network, equation or model which will be used for data analysis.The algorithm modeling is performed in three steps which are selecting the algorithm, training the algorithm and predicting the response using trained algorithm. In this Linear Regression, K-Nearest Neighbor and Artificial Neural Network are selected for modeling generated data. Python software is used for the modeling and predictions.

1) Linear regression

Linear regression is the techniques used to find the relationship between input variable and output variable by fitting a linear equation for the given data set. This is a type of supervised learning and which deals with continuous output only.

2) *K-Nearest Neighbor Algorithm (KNN)*

It is also a type of supervised learning but which can be used to solve the continuous as well as categorical data set. In this method data points have similar characteristics of test input are selected as a neighbors and its corresponding outputs are used for predicting the output for given test input. The following three steps are performed to obtain the results which are

1. Giving the input data and find the best value of K based on given data set.
2. Calculate the distance between every data point and test input data.
3. Sort the calculated distance in an ascending order.
4. Select the top K entries and find the corresponding output values
5. If output data is continuous, fine mean of outputs. If output data is categorical, find the repeated occurrence of K outputs.

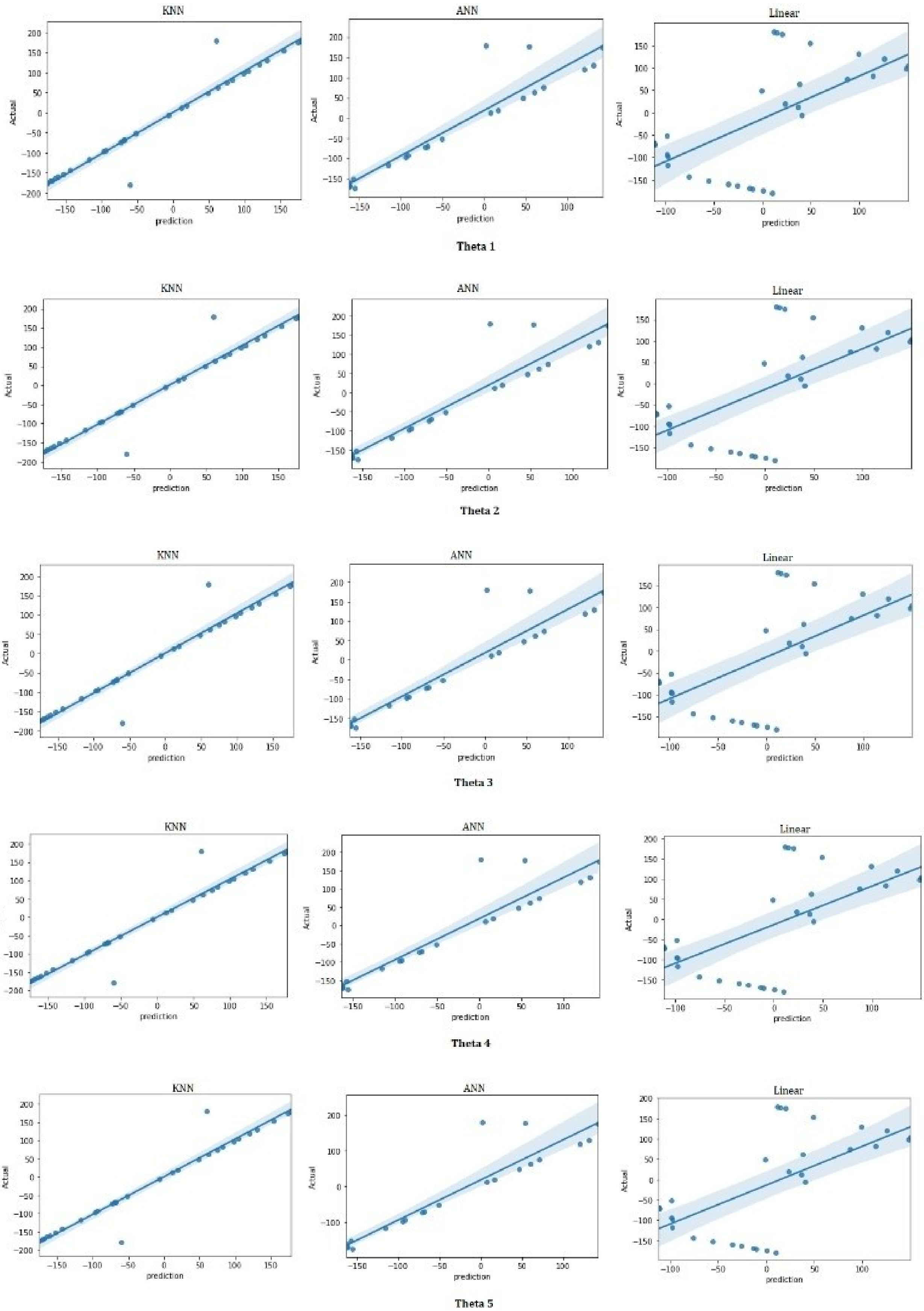

Fig. 3. Theta values predicted using KNN, ANN and Linear Regression versus actual values

3) Artificial Neural Network

There are three layers of neurons such as input layer, hidden layer and output layer are arranged in network. Each neuron get the input and give some weightage and gives an output.The activation function or output function is the sum of function of weightage and input. A suitable activation function gives the better output prediction. The accuracy of the model depends on the weightage values only.

C. Validation of model

Validation is the process of checking the accuracy of created model. In every data science techniques data will be divided in to test set which will be used for testing the created model and training set which will be used for training the model.

IV. Results and Discussions

The model building, training and testing of algorithms are done using python3.7. Linear regression, KNN and ANN algorithm tools in python are utilized for modeling and testing. In this work, data set is generated using forward kinematics models. A thousand data sets were first generated for the input variables X, Y and Z coordinates in mm and joint variables in degrees. The training, testing and validation of different machine learning techniques are done using above generated data set.
Actual theta value versus Predicted theta value plots for every joint variable using three algorithms are shown in Figure 3. From the above graph, observations made that the better fit is achieved using KNN algorithm by comparing with ANN & Linear Regression. For θ1-θ5 it clearly shows the points are fitting properly with zero line only in KNN algorithm. In KNN, points are uniformly speeded with Zero line and very less number of outliers. By comparing three algorithms R -square values, it is proved that KNN algorithm is giving better results.

TABLE III.

R^2 VALUE FOR JOINT VARIABLES

Technique	R Squared value
LR	91.3%
KNN	99.4%
ANN	60.5%

The highest R-squared value was obtained using the k-NN algorithm (99.4%). The R-squared values obtained using A-NN and Linear Regression are (91.3%) and (60.5%), respectively which is presented in Table.3.

RMSE values are calculated by varying the percentage of training size of dataset (50% to 90%) and the same is plotted in Figure 4, which again proves that RMSE values of the KNN algorithm for all the different percentage of training datasets.

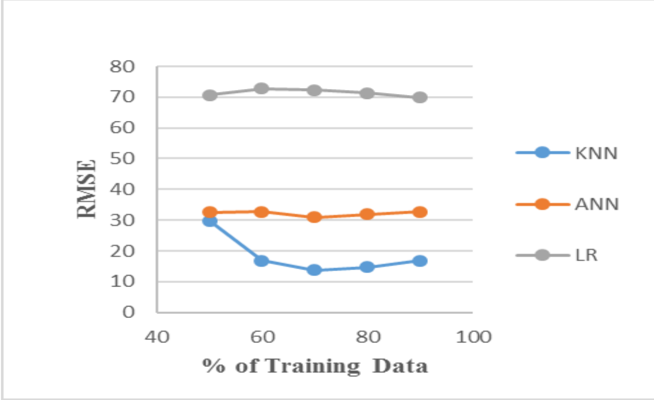

Fig. 4. Comparison of RMSE for LR, KNN and ANN

So KNN algorithm is chosen for prediction of joint variables. The initial point, final point and four intermediate control points in Cartesian space (X, Y) is shown in Table 4.

TABLE IV

CONTROL POINTS

Sl.No	X(mm)	Y(mm)	Z(mm)
1	-200	150	-400
2	-80	180	150
3	40	210	500
4	160	240	700
5	280	270	650
6	400	300	500

After validation, the three joint angles are predicted using the model by giving six points in Cartesian space. Control points defined in Cartesian space is given as an input and the path is obtained using cubic B –spline polynomials which is shown in figure.

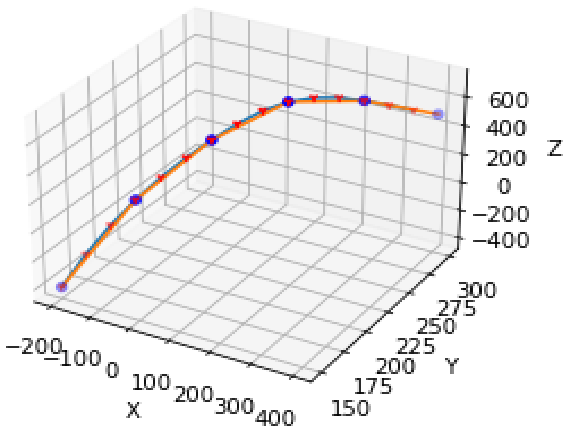

Fig. 5. Cubic Spline Interpolation

The predicted joint variables using KNN algorithm is given in table 5.

TABLE V

JOINT ANGLES OBTAINED

Sl.No	θ_1 (deg)	θ_2 (deg)	θ_3 (deg)	θ_4 (deg)	θ_5 (deg)
1	-70	130	105	27	10
2	40	90	45	-127	-10
3	40	40	-15	27	0
4	20	-90	-90	47	-55
5	3.3	-70	-120	7	-25
6	37	-40	-75	40	-60

Control points defined in joint space is given as an input and the path is obtained using cubic B –spline polynomials[19]. From the path the position, velocity, acceleration and jerk values are obtained by varying the parameter u from 0 to 3.

u = (k/4).

u values can be obtained by varying the k value from 0 to 12.

If $(0\leqslant u\leqslant 1)$

Angular Displacement Equation

$$Pi(k) = \theta i0(1-u)^3 + \frac{\theta i1}{4}(7u^3 - 18u^2 + 12u) + \frac{\theta i2}{12}(-11u^3 + 18u^2) + \frac{\theta i3}{6}(u^3)$$

Angular Velocity Equation

$$Vi(k) = \theta i0(-3(1-u)^2) + \frac{\theta i1}{4}(21u^2 - 36u + 12) + \frac{\theta i2}{12}(-33u^2 + 36u) + \frac{\theta i3}{2}(u^2)$$

Acceleration Equation

$$Ai(k) = \theta i0(6(1-u)) + \frac{\theta i1}{4}(42u - 36) + \frac{\theta i2}{12}(-66u + 36) + \theta i3 * u$$

Jerk Equation

$$Ji(k) = \theta i0(-6) + \theta i1(10.5) + \theta i2(-5.5) + \theta i3(1)$$

If $(1\leqslant u\leqslant 2)$

Angular Displacement Equation

$$\theta i1\frac{(2-u)^3}{4} + \frac{\theta i2(7u^3 - 18u^2 + 5u - 18)}{12} + \frac{\theta i3(-7u^3 + 27u^2 - 27u + 9)}{12} + \theta i4\frac{(u-1)^3}{4}$$

Angular Velocity Equation

$$\theta i1\frac{-3(2-u)^2}{4} + \frac{\theta i2(21u^2 - 36u + 5)}{12} + \frac{\theta i3(-21u^2 + 54u - 27)}{12} - \theta i4\frac{3(u-1)^2}{4}$$

Acceleration Equation

$$\theta i1\frac{6(2-u)^1}{4} + \frac{\theta i2(42u - 36)}{12} + \frac{\theta i3(-42u + 54)}{12} + \theta i4\frac{6(u-1)}{4}$$

Jerk Equation

$$\theta i1(-1.5) + \theta i2\,(3.5) + \theta i3\,(-3.5) + \theta i4\,(1.5)$$

If $(2\leqslant u\leqslant 3)$

Angular Displacement Equation

$$P_i(k) = \frac{\theta_{i2}(3-u)^3}{6} + \frac{\theta_{i3}(11u^3 - 81u^2 + 189u - 135)}{12} + \frac{\theta_{i4}(-7u^3 + 45u^2 - 93u + 63)}{4} + \theta_{i5}(u-2)^3$$

Angular Velocity Equation

$$Vi(k) = -\frac{\theta i2}{2}(3-u)^2 + \frac{\theta i3}{12}(33u^2 - 162u + 189) + \frac{\theta i4}{4}(-21u^2 + 90u - 93) - \theta i5(3(u-2)^2)$$

Acceleration Equation

$$Ai(k) = \theta i2(3-u) + \frac{\theta i3}{12}(66u - 162) + \frac{\theta i4}{4}(-42u + 90) + \theta i5(6(u-2))$$

Jerk Equation

$$Ji(k) = \theta i2(-1) + \theta i3(5.5) + \theta i4(-10.5) + \theta i5(6)$$

Total execution time and jerk are calculated using cubic spline equations.The generated cubic spline profile has minimum execution time and jerk as shown in Table 7.

TABLE VII

TIME AND JERK VALUES FOR CUBIC SPLINE TRAJECTORY

Cubic Spline Trajectory	
Time (sec)	4.09
Jerk(rad/s^3)	27.8

V. CONCLUSION

In this paper, the methods like Linear Regression, K-Nearest Neighbor and Artificial Neural Network are used to obtain the inverse kinematics solution of Pioneer 2 robot. Forward kinematic equations of Pioneer 2 robot is utilized to create the training data set for the machine learning algorithms. The difference in actual and predictions of Linear Regression, gives best results as compared to KNN and ANN. Therefore, KNN can be used better solution of inverse kinematics with fast results and high accuracy. Computational complexity of Inverse kinematics can be solved using forward kinematics with machine learning algorithms. Then the smooth trajectory is achieved using cubic spline interpolation. Limitations of using KNN algorithm is accuracy depends on quality of data, requires more space and sensitive to irrelevant feature. In future, the same methodology can be applied for different industrial robots with different machine learning algorithms.

REFERENCES

[1] Kumar, A., Banga, V.K., Kumar, D. and Yingthawornsuk, T., 2019, November. Kinematics Solution using Metaheuristic Algorithms. In 2019 15th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS) (pp. 505-510). IEEE.

[2] Mary, Ali H., Tolgay Kara, and Abbas H. Miry. "Inverse kinematics solution for robotic manipulators based on fuzzy logic and PD control." In 2016 Al-Sadeq International Conference on Multidisciplinary in IT

[3] Lv, X. and Zhao, M., 2019. Application of improved BQGA in robot kinematics inverse solution. Journal of Robotics, 2019.

[4] Yin, S., Ji, W. and Wang, L., 2019. A machine learning based energy efficient trajectory planning approach for industrial robots. Procedia CIRP, 81, pp.429-434.

[5] Fang, Y., Hu, J., Liu, W., Shao, Q., Qi, J. and Peng, Y., 2019. Smooth and time-optimal S-curve trajectory planning for automated robots and machines. Mechanism and Machine Theory, 137, pp.127-153.

[6] Yıldırım, Ş. and Ulu, B., 2017, May. Design of Artificial Neural Network Predictor for Trajectory Planning of an Experimental 6 DOF Robot Manipulator. In International Workshop on Computational Kinematics (pp. 153-160). Springer, Cham.

[7] Jha, P., Biswal, B.B. and Sahu, O.P., 2015. Inverse kinematic solution of robot manipulator using hybrid neural network. Int J Mater Sci Eng, 3(1), pp.31-38.

[8] Ayyıldız, M. and Çetinkaya, K., 2016. Comparison of four different heuristic optimization algorithms for the inverse kinematics solution of a real 4-DOF serial robot manipulator. Neural Computing and Applications, 27(4), pp.825-836.

[9] Jing Li, Fei Ye, NanYan Shen ∗, ZiRui Wang, Liang Geng- Dimensional synthesis of a 5-DOF hybrid robo- 20 February 2020- Mechanism and Machine Theory 150 (2020) 103865

[10] Ali Roshanianfarda, Noboru Noguchia, "Kinematics Analysis and Simulation of A 5DOF Articulated Robotic Arm Applied to Heavy Products Harvesting", Journal of Agr i cul tural Sc iences 24 (2018) Page 91-104, SEP 2017.

[11] Jinyan Lu, De Xu, Peng Wang- A Kinematics Analysis for a 5-DOF Manipulator", IEEE , 26th chinese control and Decision Conference(CCDC-2014)

[12] Layatitdev Das, Jajneswar Nanda, S. S. Mahapatra , "A Comparative Study of Prediction of Inverse Kinematics Solution of 2-DOF, 3-DOF and 5-DOF Redundant Manipulators by ANFIS", IJCSN International Journal of Computer Science and Network, Volume 3, Issue 5- Technical Report No CSM-414, October 2014.

[13] Guo-Shing Huang, Po-Hao Tseng , "Development and Analysis of 5-DOF Manipulator Kinematics", International Conference on System Science and Engineering (ICSSE) National Chi Nan University, Taiwan, July 7-9,2016

[14] M. A. Devi, C. P. S. Prakash, R. P. Chinnannavar, V. P. Joshi, R. S. Palada and R. Dixit, "An Informatic Approach to Predict the Mechanical Properties of Aluminum Alloys using Machine Learning Techniques," 2020 International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 2020, pp. 536-541, doi: 10.1109/ICOSEC49089.2020.9215277.

[15] Song Kang and Wusheng Chou, "Kinematic Analysis, Simulation and Manipulating of a 5-DOF Robotic Manipulator for Service Robot", Proceedings of 2019 IEEE ,International Conference on Mechatronics and Automation,August 4 - 7, Tianjin, China , 2019.

[16] V.N. Iliukhin, K.B. Mitkovskiib D.A. Bizyanova, A.A. Akopyan," The modeling of Inverse Kinematics for 5 DOF manipulator",Procedia Engineering 176 (2017) 498 – 505.

[17] Ranganathan, G. "An Economical Robotic ArmPlaying Chess Using Visual Servoing." Journal of Innovative Image Processing (JIIP) 2, no. 03 (2020): 141-146. 2.

[18] Smys, S., and G. Ranganathan. "Robot assisted sensing control and manufacture in automobile industry." Journal of ISMAC 1, no. 03 (2019): 180-187.

[19] Arunadevi M, C P S Prakash, "A Multi Performance Characteristics Optimization of Robotic Arm Trajectory using Machine learning Technique", Grenze International Journal of Engineering and Technology, Special Issue, (Vol. 4, Issue/ 3, pp. 32-36), 2018.