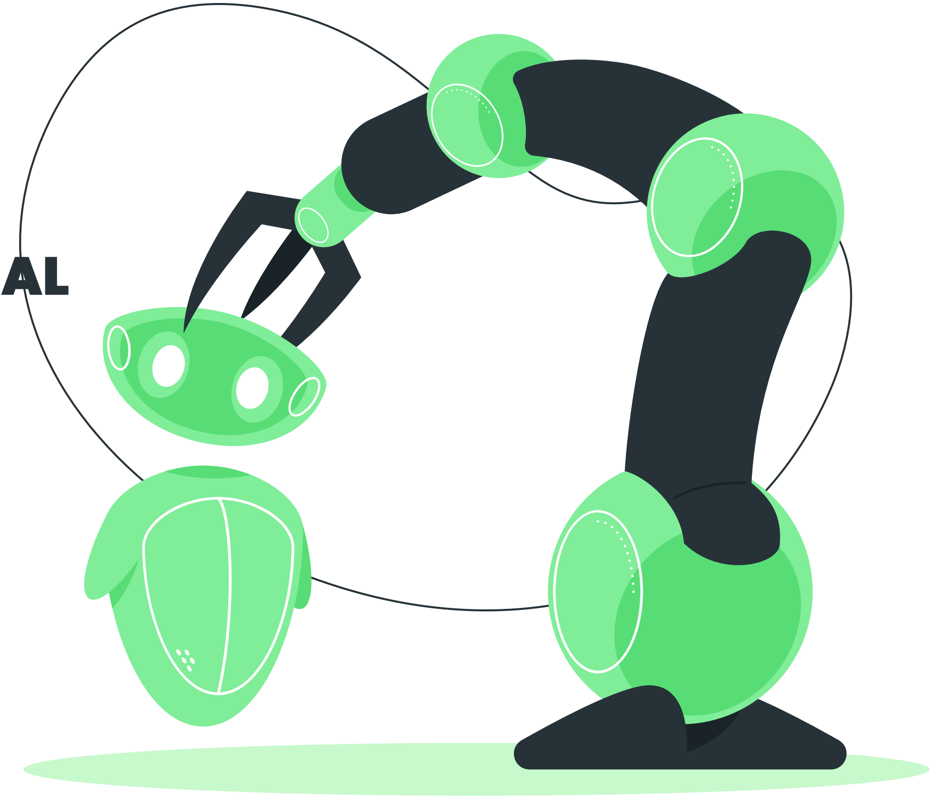
MODELLING THE INVERSE KINEMATICS OF A 2 LINK MANIPULATOR USING DEEP NEURAL NETWORK

PROJECT REPORT



TEAM MEMBERS

- ADITYA RAJ
- RAJIB DAS

MENTORS

- TUSAR SAHU
- APURBA PRASAD PADHY

Table of Contents

1 ABSTRACT

2 MOTIVATION

3 WORKFLOW

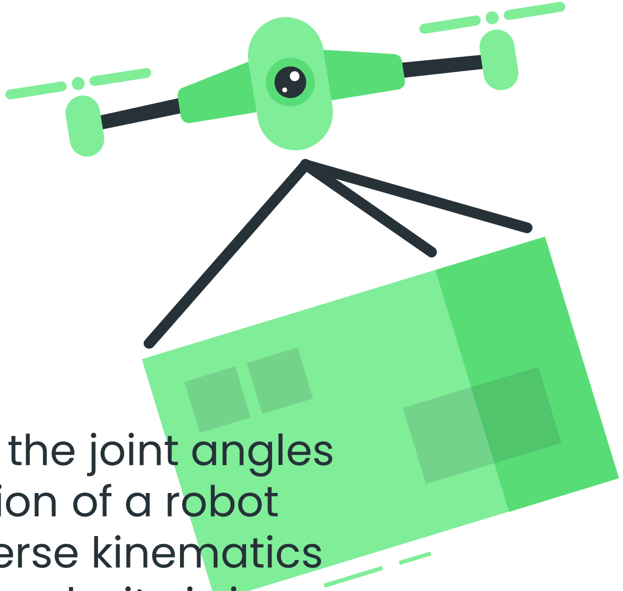
4 COMPONENTS

5 APPLICATION

6 FUTURE IMPROVEMENTS

7 ACKNOWLEDGEMENT

ABSTRACT



In this project our purpose is to calculate all of the joint angles that would result in a specific position/orientation of a robot arm ....This problem is famously called the inverse kinematics problem. In inverse kinematics learning, the complexity is in the geometric and non linear equations (trigonometric equations, which in our case is the cosine function) and in the matrix inversion, this in addition to some other difficulties faced in inverse kinematics like having multiple solutions.

MOTIVATION

- The true power and advantage of neural networks lies in their ability to represent both linear and non-linear relationships and in their ability to learn these relationships directly from the data being modelled.

- Traditional linear models are simply inadequate when it comes to modeling data that contains **non-linea**r **characteristics**. The most common neural network model is the multilayer perceptron (MLP). This type of neural network is known as **a supervised network** because it requires a desired output in order to learn. The goal of this type of network is to create a model that correctly maps the input to the output using historical data so that the model can then be used to produce the output when the desired output is unknown.

WORKFLOW

- CREATING REQUIRED DATASETS: ALMOST 1036800!! DATASETS
- WE DIVIDED THE WHOLE DATASET INTO 3 PART
    1. TRAINING DATASETS(8,36,800)
    2. VALIDATION DATSETS (1,00,000)
    3. TESTING (1,00,000)
- CREATING THE NEURAL NETWORK:

COMPONENTS OF DNN

1

HIDDEN LAYERS

A total of 5 hidden layers was used in the network in addition to the 1 input layer and one output layer.The input layer contains two nodes reading the x and y coordinate simultaneously.The output also contains 1 nodes in order to give the value of each of two thetas .

2

ACTIVATION FUNCTION

We rejected softmax ,tanh and sigmoid because they are basically used in classification sort of problems and in case of softmax it is not used because it basically scales them to probabilities in case of sigmoid and tanh we tried using them by mapping our input to their domains but the loss was very much hence at the end we ended up using relu as our activation function

APPLICATION

Using DNN in calculating the angles in 2 link manipulator optimizes the time taken which is required in fast movement of the robotic arm . it can be useful when there is a large number of datasets and lot more complexity.

FUTURE IMPROVEMENT

The DNN approach can further be used in much more complex quadruped or multiple link simulator and with some given constraints.

ACKNOWLEDGEMENT

We would like to thank MODELS AND ROBOTICS SECTION OF IIT ROORKEE for giving us this wonderful opportunity to be a part of this adventurous project and enhances our skills and reach. We would also like to thank our mentors for providing all the possible help and guiding us till the end.