Intelligent adaptive systems coursework 1

# Relevant examples of IK learning

Various different approaches have been used in inverse kinematics learning such as using Anfis, neural networks and genetic algorithms. By far the most popular are artificial neural networks, these can be broken down into various types such as multilayer perceptron’s, radial basis functions and recurrent neural networks.

One approach taken by ANfis RRR paper used Anfis to learn the inverse kinematics of both a 2 joint and 3 joint planar manipulators. The number of Anfis system was the same as the number of joints, the inputs into the 2 joint system were the X and Y end effector locations. While the input into the 3 joint system was the X, Y and Phi. The paper states that they achieved a suitable level of error. The approach used here will be very similar to the one used for this coursework.

Another study by Second anfis paper also used a similar approach, this study solely focused on a 3-joint manipulator. The inputs were the X Y and Phi (EE pose) of the end effector position corresponding to the thetas. It was trained with 100 samples and 200 epochs; it matched the chosen trajectories quite well. Photo?

One study by perceptron1 looks at using a multi-layer perceptron for inverse kinematics, the inputs to the network are the X, Y and phi. The chosen structure had one hidden layer of 100 sigmoid activated neurons with 3 outputs neurons corresponding to the joint angles. The network can match the desired output arbitrary well. This study notes that further improvements to the network could be made, such as looking at the size and number of hidden layers. This paper is used as a reference for the MLP developed in this assignment. Photo, number of epochs?

A study by RBF and ANN looked at both MLPs and RBFs for inverse kinematics of a 3R planar manipulator and compared them. The inputs to both types of network were the X, Y and Phi. The paper describes details of both of the systems developed such as the learning used.

Some studies such as the one by MIXED have looked at combining approaches to reduce error. In this approach the inverse kinematics is calculated with 3 separate systems (2 MLP’s and one RBF) and the error of each is compared and the best one is chosen.

Small comparison summary?

* Genetic algorithms

# Inverse kinematics with Anfis

## Data/workspace generation

In order to generate a dataset that Anfis could be trained on a workspace was generated using the forward kinematics equations for a 3R planar manipulator.

* FK equations

These positions were calculated across a range of joint angles. The size of the dataset can be adjusted by changing the interval in the for loop (this can be done for a specific angle or for all). The dataset for the pose (Phi) is also created at this time, as it is the summation of the joint angles.

The end result is a dataset with a variety of end effector poses and positions and their associated joint angles which will be the training output. This is similar to the approach used in ANFIS RRR PAPER, the workspace is shown in workspace figure.

* Workspace photo here

## Anfis structure & training

For learning inverse kinematics, a set of 3 Anfis systems? were used, the inputs to each were the X & Y of the generated end effector locations as well as the corresponding Phi(pose), the training outputs of each system were the corresponding joint angle (theta).

* Number of epochs
* Number of membership functions
* How did I get to the right number?
* Relate this to the relevant paper?

## Validation

* Error metric
* How close to target?
* Comparison to actual IK calculations? Error graph
* Show IK path on workspace?

## Discussion

* Discuss general accuracy and ease of this method?
* Benefits of using Anfis?

# Inverse kinematics with MLP

## Data/workspace generation

* Larger data set than anfis, how many points
* Same method as anfis
* Picture of workspace

## MLP structure & training

* Paper had 1 hidden of 100, but we’ve gone for 12 \*12? (any ann with 2 hidden layers can map any function ref)

## Validation

* Talk about the actual MLP performance as well as your own validation set
* Comparison of IK’s, error graph
* Same error metric
* Show IK path on workspace?

## Discussion

* Compare to Anfis, can use a much larger data set, more accurate

# Problems with IK learning

* Problem with elbow up or down
* How was this solved?
* Implementation?

# Search algorithms

* Genetic algorithms
* Implementation?