Fuzzy logic control system for a Underwater unmanned vehicle

Phaneendra Nalla
Graduate student of electrical and computer engineering department, Michigan Tech
Subrat S. Kamat
Graduate student of electrical and computer engineering department, Michigan Tech
Zicheng Sun
Graduate student of mechanical engineering department, Michigan Tech
Du Zhaoxu
Graduate student of electrical and computer engineering department, Michigan Tech

***Abstract*- In some scenario of underwater activity, it is too dangerous for human participating in underwater vehicle. At this circumstance, underwater unmanned vehicle (UUV) is needed. To maintain the stable movement of the UUV, a robust control system is specifically needed. One of the most mature control system for UUV uses PID control as feedback control. PID control is an autonomous control method, it provide robust control against disturbance. However, PID control is not optimal for UUV, since it's a passive control method. Thus, fuzzy logic inference system (FIS) will be introduced in the feedback system for UUV as controller. Fuzzy logic control system is an expert system. Based on preset rules, it can provide well planned solution according to real time input as parameters. For FIS system, input data is derived from available state space models while output data is given. These data is used in Adaptive Network Fuzzy Inference System (ANFIS) to train the FIS system. After training, FIS performed a stable controller behavior.**

.

I. INTRODUCTION

To set up the simulation model for a simple UUV, state space model is introduced according to paper from Javier etc. This model uses rudder angle, stern angle and bow angle as input and gives yaw angle, yaw rate, pitch angle, pitch rate, depth and heave velocity etc. as output. Part of the out of this model is introduced in simulation model as the feedback data to create input data for FIS. These data is fuzzificated, and is coded into membership functions thus to create FIS system. Sugeno type FIS is created for ANFIS training. Program is written to give optimal output according to the understanding of the FIS input data. Given both input and output. ANFIS system can train the FIS by setting up its rules and repeated learning from given data and adjusting rules. After training, simulation model is created. Trained FIS is tested in this model, and data is adjusted to give optimal output.

This project deals with a system which experiences very high nonlinear changes in its Environment. Also, these Nonlinear Changes cannot be interpreted as a mathematical function and hence designing a controller to control the movement of UUV in its desired way is very tough. An ANFIS makes it simple to design the controller by exploiting the concept of neural networks. ANFIS initially trains the FIS system with the existing data so that rules and membership function were created automatically accounting for all the nonlinear changes in the environment.

II. MODELS

The block of system is shown in Figure X.

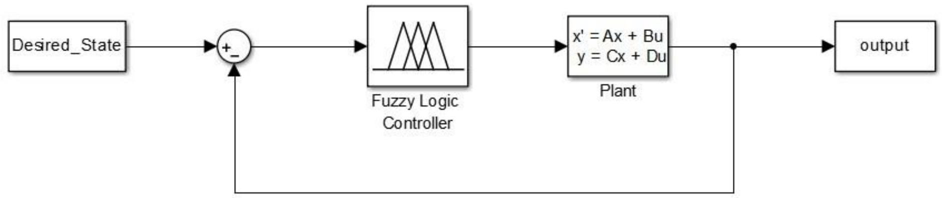


Figure X. System block

A. Plant model

Under assumptions of constant surge velocity and deeply submerging etc. Linear model is obtained in paper by Javier etc.

(linear model equations)

where A is a 8 × 8 matrix representing: , B is an 8 × 3 matrix representing hydrodynamics coefficients, and u(t) is the input matrix for UUV plant.x(t) is the time variable state matrix.

(x(t) equation)

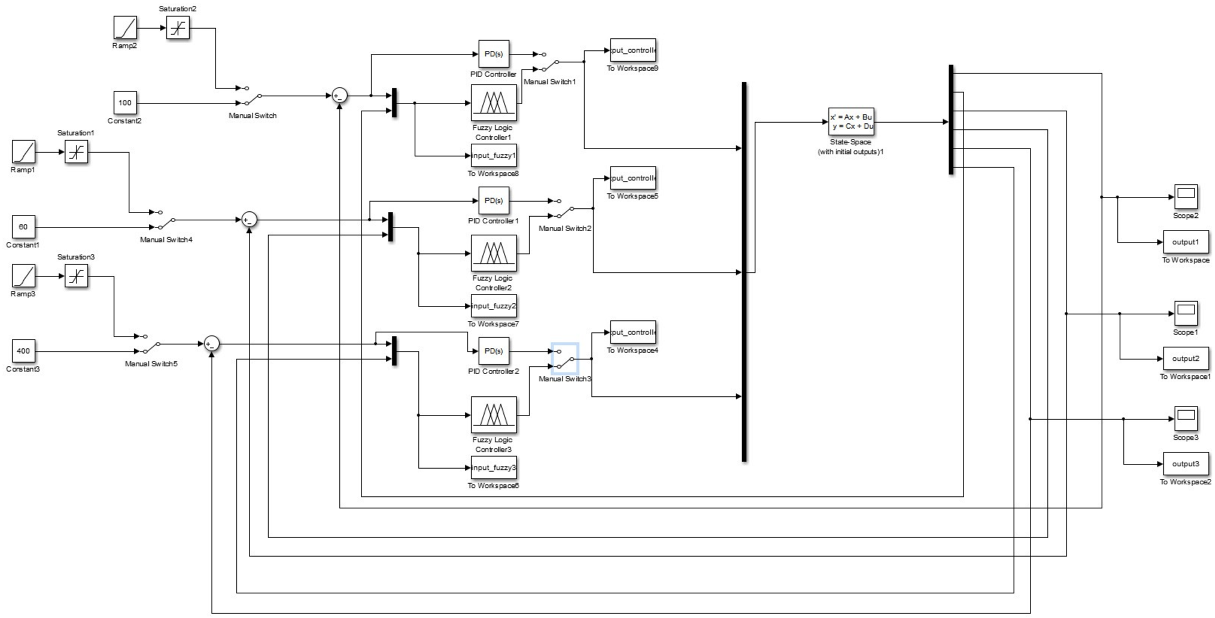
where r(t) is yaw rate, phi(t) is yaw Euler angle, w(t) is heave velocity, q(t) is pitch rate, theta(t) is pitch Euler angle and z(t) is depth.

B is an 8 × 3 matrix representing^.

B. Simulation model

In this paper, a simulink model is designed in order to simulate the motion of the submarine linear model and its fuzzy logic controller.

Simulink model of the UUV for MATLAB simulation and its controller is shown in Fig.X. It include 3 independent feedback control loop, which are designed to control the yaw angle, pitch angle and depth of the submarine, respectively.



Yaw angle, pitch angle and depth, these three variables are functions of time, will be passed to the sum block and compare with desired yaw angle, desired pitch angle and desired depth and generate the difference, respectively. The other three variables, the yaw rate, pitch rate and depth rate will maintain the same as they obtained from the plant. These difference and rate will feedback to three different fuzzy logic controllers. Fuzzy logic controller1 takes the difference of actual and desired yaw angle and yaw rate as the input and generate the rudder angle as the output. Fuzzy logic controller2 takes the difference of actual and desired pitch angle and pitch rate as inputs and generate the stern angle as output. Fuzzy logic controller3 takes the difference of actual and desired depth and depth rate as input and generate the bow angle as output.

Beside the plant and controller, there are some ports in the simulink model used to connect with the matlab workspace. Three "to workspace¡± blocks were connected to output of the plant to record the yaw angle, pitch angle and depth. Also, the input ports and output ports of each controller were connected to some "to workspace¡± blocks. Therefore, the input and output history of the controller are recorded. These information can be used to debug and optimize the controller.

III. FIS CONTROLLER

A. Fuzzy inference systems

MATLAB ANFIS toolbox accept one output for each FIS only. To train the data, three FISes separately control rudder angle, stern angle and bow angle are created, providing one output each. First FIS is used to control rudder angle. It requires two inputs, one is difference between desired and actual yaw angle, and the other is current yaw rate. The second FIS is used to control stern angle. It requires two inputs, one is difference between desired and actual pitch angle, and the other is pitch rate. The third FIS is used to control the bow angle. It requires two inputs, one the difference between desired and actual depth, and the other is heave velocity. The third FIS is applied with Mamdani type, and is not trained in ANFIS. Three FISes are shown in Fig.X

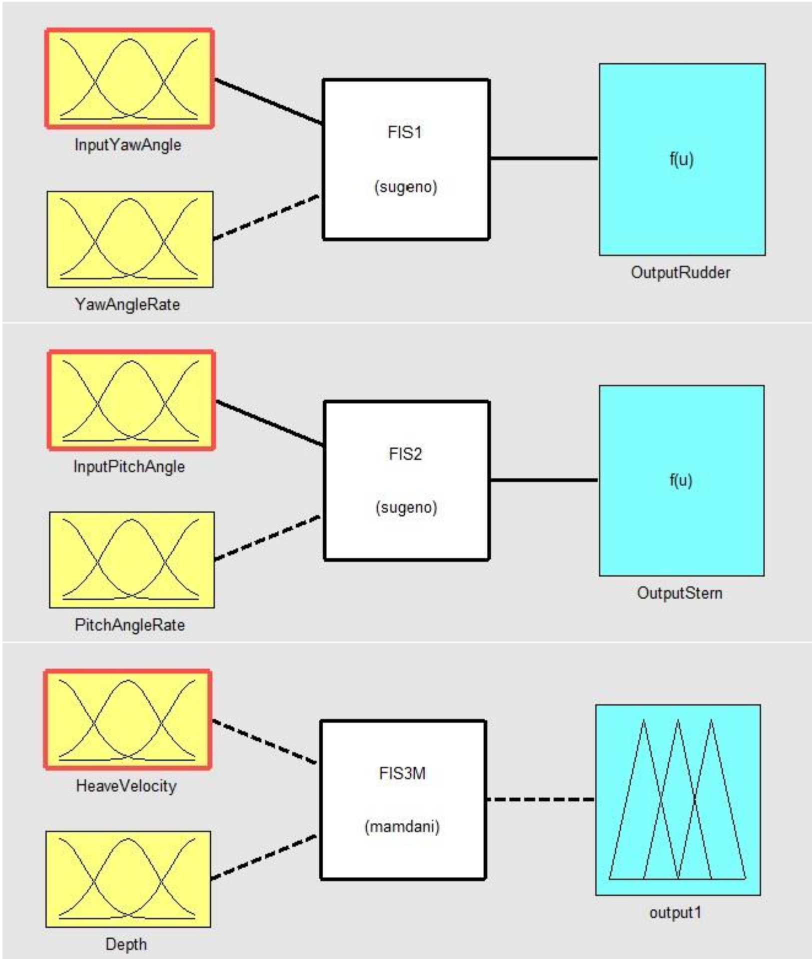


Figure X. Three FIS controllers

According to the state space model, the range of plant output is defined. Since state space model is confined within certain maximum rudder, stern and bow output shown in Table.X, any desired plant output more than a maximum value will be given an input of these maximum values.

TABLE.X PLANT OUTPUT RANGE

OUTPUTS OF PLANT	***Range***
Rudder Angle	-75° ~ +75°
Stern Angle	-35° ~ +35°
Bow Angle	-35° ~ +35°

To initial the input and output data, membership functions for FIS1 and FIS2 input are defined. Initial output for these two FISes are assigned.

A. Control data

Three fuzzy interface systems are required to be used as controllers to control the Yaw, Pitch and Bow angle of the UUV. The three fuzzy interface systems would control rudder angle, stern angle and bow angle given to the state space model of the UUV.

• Yaw angle control data

Now for the control of the yaw angle the fuzzy controller should have its inputs as the error of desired angle i.e. desired UUV yaw angle−actual UUV yaw angle and the second input is the yaw rate , which would decide the direction of rotation of the UUV in real time thus using these two inputs the corresponding output or the rudder angle could be decided by a logical relation. This could be done by using an ANFIS system to train a fuzzy interface system so as to get an optimal controller to control the rudder angle to get the desired yaw angle.

Now the ANFIS system requires a training data which would be used for training in the ANFIS so as to create a new FIS system based on the sugeno model which would train the FIS system according to the data fed to it thus the ANFIS acts as a black box. The data needed to be given for the Yaw angle control of the UUV are two inputs that are error in yaw angle and the yaw rate with an output i.e. the rudder angle. This data is obtained by randomly giving rudder angles to the state space model of the UUV and then correspondingly getting the yaw angle and yaw rate values at various time intervals. About 33000 lines of data was collected each line consisting of two columns , the first column being the error in yaw angle and the second being the yaw rate. The error in yaw angle is taken by adding the corresponding input rudder angle to the corresponding Yaw angle. The reason of adding the current yaw angle to the input rudder angle instead of subtracting the yaw angle from the rudder angle is because the yaw angle and rudder angle are inverse signed to each other ,for example say for a positive yaw angle we would be required to give a negative rudder angle input. So the yaw angle error would be the addition of current rudder angle and current yaw angle.

Now as the rudder angle has a limitation on its angle rotation, which we have given as -75 degrees to +75 degrees so the output of the fuzzy controller which is the rudder angle is needed to be fed to the state space model should also have a limitation of -75 degrees to 75 degrees. For this a test.m, MATLAB code was created which would create a data of outputs depending on the two input values to the fuzzy controller.

The code works by first initializing all the outputs (i.e. the rudder), where for the absolute value of the input yaw error angle is larger or equal to 75o , the output of the fuzzy controller i.e. the rudder angle would be the negative sign of the current yaw error angle multiplied by 75, for example if yaw angle error is 335 degrees , then the output would be -75 degrees. Now the code to create the remaining of the output data that doesn't match the error yaw angle of above 75 degrees limitation is divided into four parts. The first part being when both the yaw angle error and yaw rate are negative, the second part being when both are negative and the third and fourth part being when one of the yaw angle error and yaw rate are either positive or negative. Then for each condition at different ranges of yaw rate the output rudder angle which needs to be fed to the state space model is specified by taking the negative current yaw error angle and then adding or subtracting that value with logically decided values so as to control the rudder accordingly. The ranges of yaw angle rate taken are 2 to 1.5, 1.5 to 1 ,1 to 0.5, 0.5 to 0 for positive yaw rates and when negative yaw rates are the inputs then the range of yaw angle rates taken to decide the rudder angle are -2.5 to-1.5, -1.5 to -1, -1 to -0.5 and -0.5 to 0. At last all these conditions create a set of output data which is then stored in a matrix , with the first two columns containing the inputs, the first input being the yaw angle error and the second column being the yaw rate and the third column being the corresponding computed output found from test.m code.

This data matrix containing the inputs and outputs are then fed to the ANFIS as the training data so as to create an ANFIS model.

- Pitch angle control data

The Pitch angle control data would decide the pitch angle of the UUV so the second fuzzy controller would be used to control the pitch angle of the UUV. The fuzzy controller's inputs would be the Pitch angle error (current stern angle−(−current pitch angle))and the pitch rate with the output being the stern angle which needs to be given to the state space model of the UUV. Thus as the stern angle has a limitation of ±35 degrees, the fuzzy controller has to control the pitch angle by controlling the stern angle between the range of +35 degrees and -35 degrees, so similarly a code is created which is sterntest.m which would create a set of data containing the input and output data which needs to be fed to an ANFIS to create an FIS system to control the pitch angle of UUV. The same method was used for creation of data as mentioned for the creation of data in which the inputs being the Pitch angle error and the pitch rate in the first two columns of the data and the third column being the stern angle which is the output of the fuzzy controller and input of the state space model of the UUV. The code was created in the same way as that of which was used to create the rudder / yaw angle control data, with the same values for the pitch rate ranges as the ones used for the yaw rate with the only exception being that the limitation of the stern angle would be ±35 degrees. So the obtained data would then be used to train the ANFIS to create a FIS for the controlling of the pitch angle of the UUV. About 77000 lines of data was used to train the FIS using ANFIS

- Depth control

The depth control would be done by using the error in the depth, i.e. the desired depth − actual depth and the heave velocity to get the desired bow angle to be fed to the state space model of the UUV. For this as the desired depth cannot be given as an input to the UUS model so an ANFIS cannot be used to train a FIS as the required data of the change in depth would not be available from the state space model of the UUV. So for this we create a normal FIS using a Mamdani model using a fuzzy tool box as explained earlier.

A. *ANFIS*

- Initializing ANFIS and Generating FIS Structure:

Training data to the ANFIS contains data input in all but the last column. The last column contains a single vector of output data. In this project, we have trained ANFIS using the data obtained from the state space model of the Underwater Unmanned Vehicle. ANFIS then creates a set of rules, makes adjustments to the membership function parameters so that the Fuzzy Inference System can control the UUV with minimal errors.

The following command is used to train the FIS in MATLAB.

[fis,error,stepsize] = anfis(trnData, initFis) where fis is the name of Fuzzy Inference System created,

trnData is the training data set,

error is an array of root mean squared errors,

stepsize is an array of step sizes. The step size is decreased (by multiplying it with the component of the training option corresponding to the step size decrease rate) if the error measure undergoes two consecutive combinations of an increase followed by a decrease. The step size is increased (by multiplying it with the increase rate) if the error measure undergoes four consecutive decreases,

initFIS is the name of FIS used to provide ANFIS with an initial set of membership functions for training.

This command initializes the Fuzzy Inference System with all the Membership Functions and rules set to minimize the error in the heading direction of UUV.

In this project, ANFIS was used to create two FIS structures (FIS1 and FIS2) to output the Rudder Angle and Stern Angle which in turn have impact on the Yaw angle and the Pitch angle of UUV.

FIS1 uses the difference in desired and actual Yaw angle as one of its input and rate of change in Yaw angle as it’s another input and produces the output rudder angle using the rules and membership functions established. This Rudder angle produced by the FIS1 acts on UUV and is expected to nullify the error.

For example, ANFIS outputs a Large Negative Value, when it finds the Error as large Positive value and rate of change as Large Positive, so that the Large Negative value would try to minimize the error in the Yaw Angle.

FIS2 uses the difference in desired and actual Pitch angle as one of its input and rate of change in Pitch angle as it’s another input and produces the output Stern angle using the rules and membership functions established. This Stern angle produced by the FIS2 acts on UUV and is expected to nullify the error.

- Surface of FIS system

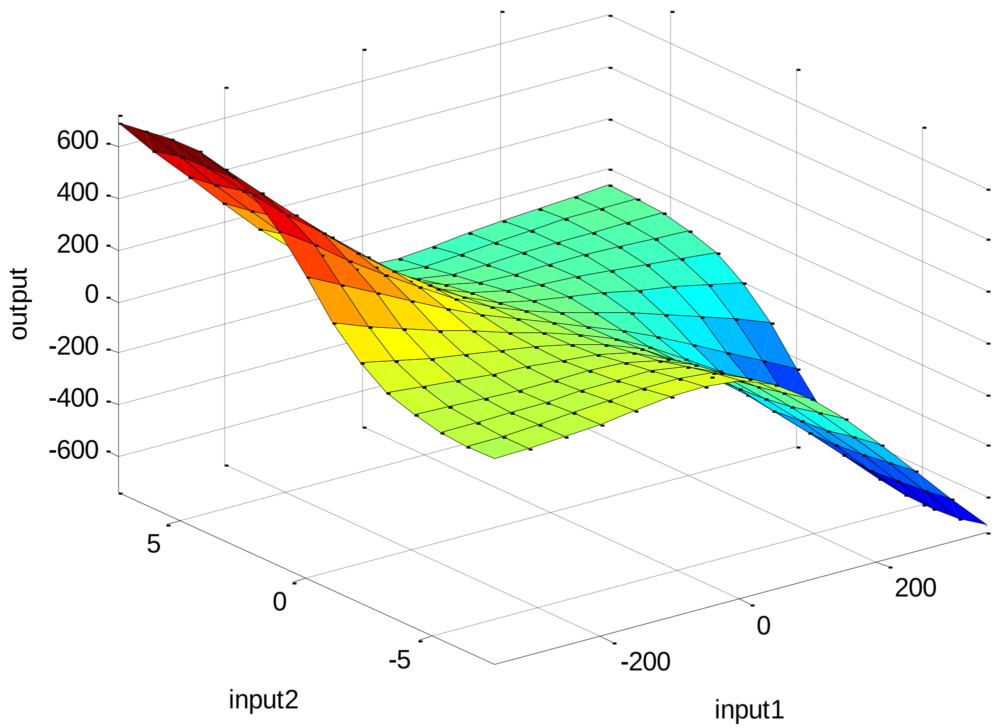


Figure X. Surface of FIS1

Figure X. is the surface of the FIS1 structure generated by the ANFIS corresponding to the inputs and outputs of the training data. A very high value of error in yaw angle and high rate of change implies that the UUV is moving very much away from its desired direction and hence FIS corresponds to a high value of rudder angle so that it commands the UUV to make a large change in its direction. This makes the error to gradually decrease and then the system becomes stable. Also, when the error is low and the rate of change of Yaw angle is low, the output is almost zero which commands the UUV to do nothing as it is exactly moving in the desired direction. This surface gives a very good description of how the fuzzy logic controller is working.

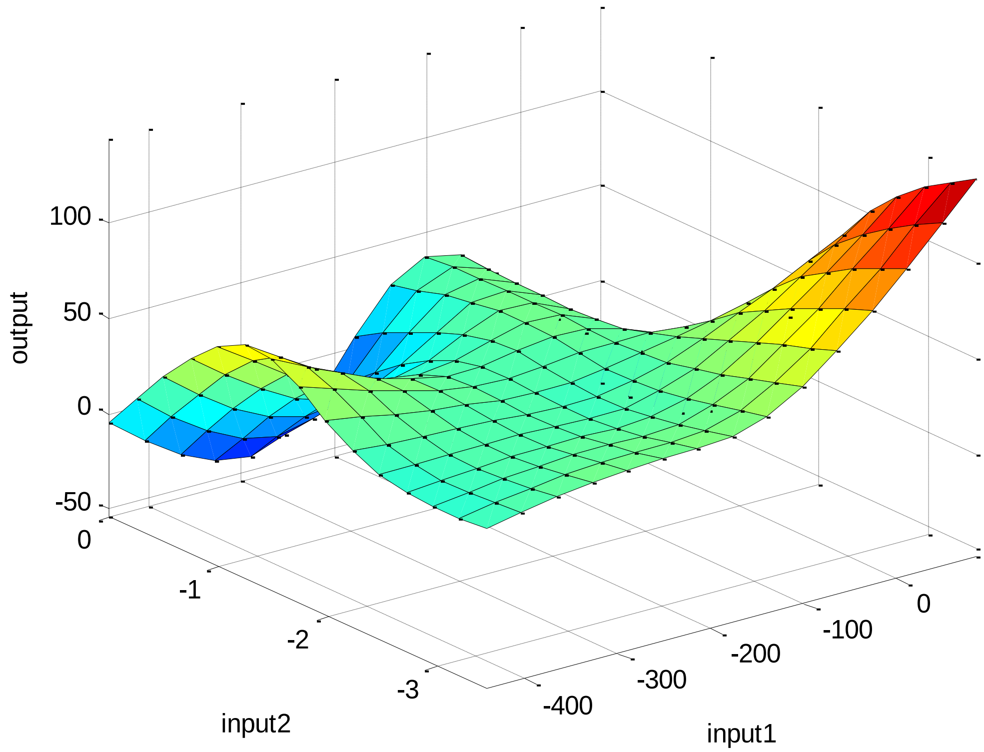


Figure X. Surface of FIS2

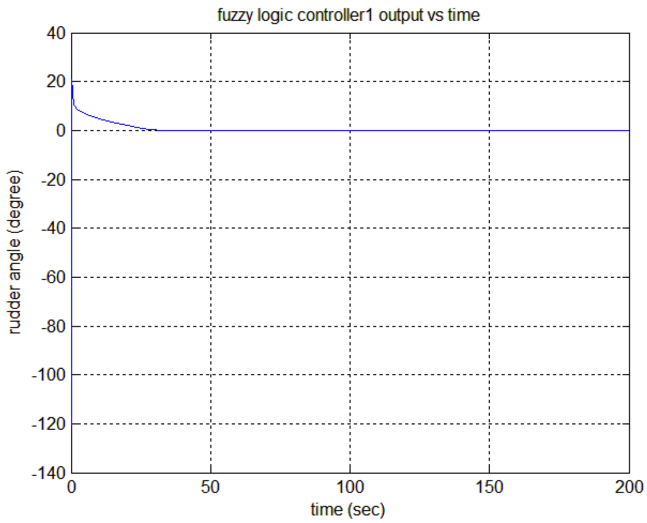
Figure X. is the surface of the FIS2 Structure created by the ANFIS. When both the difference in actual and desired pitch angle and the rate of change in pitch angle are high, the surface corresponds to a very low value on one extreme and a high value on the other extreme. This might result in an unstable performance of the system and it happened for this case. This instability in the system might be because of the lack of data for stern rate and we are in search of a solution for this.

• Using of trained FISes:

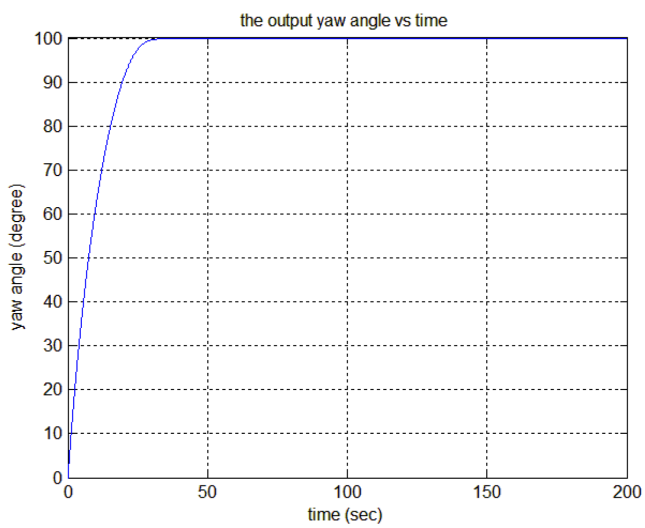
FIS1 and FIS2 structures generated by using ANFIS can now be used in the Simulink model to get the output Rudder and Stern Angle corresponding to the different inputs that it receives.

IV. RESULT

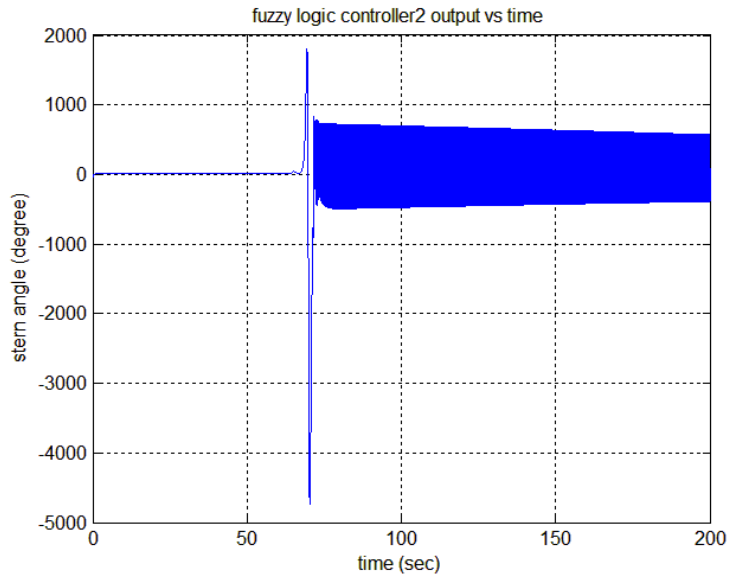
The fuzzy logic controller1 output which is used to control the Yaw angle of the UUV by controlling the rudder angle input to the state space model of UUV :

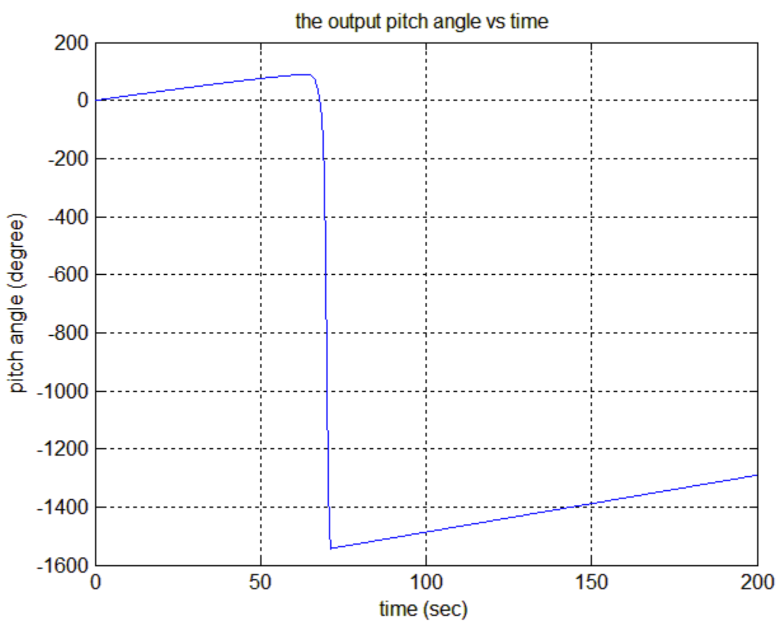


It is seen that the rudder angle which is the output of the fuzzy logic controller at first has a high angle and then it approaches to zero thus proving that the rudder angle is being controlled , below shown figure is the yaw angle output of the state space model of the UUV , which shows that the yaw angle settles to the desired yaw angle which in the shown case is 100

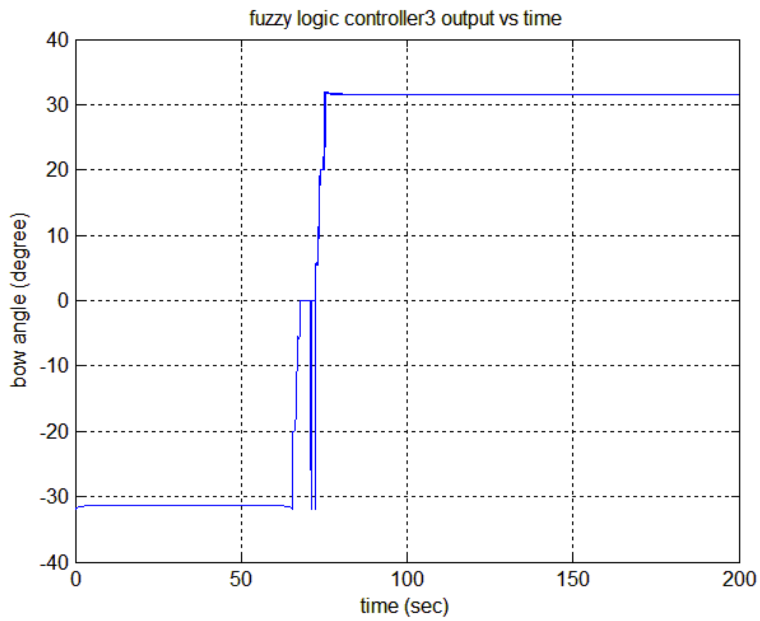


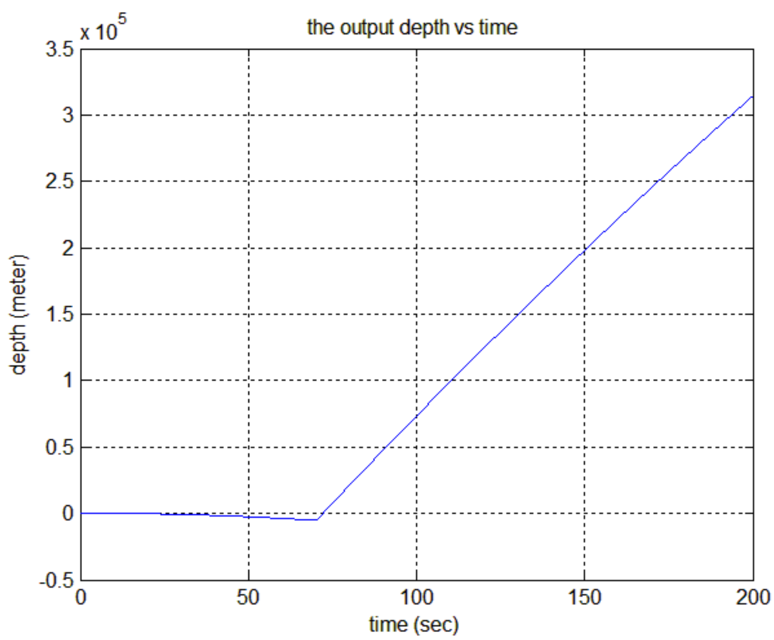
The fuzzy logic controller 2 which is used to control the pitch angle by controlling the stern angle input is found to be approaching to an unstable thus the pitch angle output was found to be unstable , it at first was able to approach the desired angle and then would become unstable as shown.





The fuzzy logic controller 3 which was designed manually using a fuzzy tool box to decide the corresponding bow angle so as to reach to a desired depth. The output of the fuzzy controller was found to be stable at first but it then approaches a particular angle and remains stable at that position only , thus the depth would increase or decrease continuously so this shows some instability in the fuzzy logic controller 3 :





IV. FUTURE WORK

A. The fuzzy logic controller 2 that is the FIS that was created to control the pitch angle of the UUV needs to further modified , by using a better and refined data to train the FIS using ANFIS so that the surface of the FIS system would be better by being symmetrical about 0 pitch angle and pitch angle rate. So further work will be done in improving the fuzzy logic controller 2.

B. The fuzzy logic controller 3 which uses a Mamdani FIS model has to be checked and verified for as the FIS model is not able to give a good output in which the bow angle would not approach zero and the pitch angle would not approach the required angle, so further changes need to be made in the FIS3 and the model if needed. So the solution to this problem needs to be found.

C. A GUI is needed to be created so as to check for the real time response of the entire model.

D. If the system works well on a real time basis then a disturbance could be induced to the model and we can see how the model would work

REFERENCES

[1] Javier Garcia, Diana M. Ovalle, Francisco Periago,. "Optimal control design for the nonlinear manoeuvrability of a submarine." http://www.researchgate.net/publication/228909707_Optimal_control_design_for_the_nonlinear_manoeuvrability_of_a_submarine