Fuzzy Logic Based Motion Controller For Underwater Remotely Operated Vehicle

P.M. Saranguhewa[1] S.D.S. Lakmal[2], R.M.T. Lakmali[3], N.D. Jayasundere, S.H.K.K. Gunawickrama

Department of Electrical & Information Engineering, Faculty of Engineering, University of Ruhuna
Hapugala, Galle, Sri Lanka
[1]shpmadushanka@gmail.com, [2]sdsameeralakmal@gmail.com, [3]rmtlakma@gmail.com

ABSTRACT

This work demonstrates the applicability of fuzzy controller to an Underwater Remote Operated Vehicle (ROV) for motion controlling with sensor feedbacks. Stabilizing Yaw, Pitch, Roll and depth against external disturbances are considered whilst responding to the remote manoeuvring commands for forward, reverse, lateral and vertical movements. All functionalities are handled by a unified fuzzy controller. Stability and responsivity of the ROV is fine-tuned by adjusting the control parameters. Performance is evaluated using field experiments. Results show the effectiveness of fuzzy controlling of the ROV motion against external disturbances.

Keywords – **fuzzy controller, ROV, control theory, stabilization against disturbances, unified motion control**

1.INTRODUCTION

Remotely operated underwater vehicle (termed ROV) is an underwater robot or an underwater drone. It is a safe and widely used type of underwater vehicle serving a range of military, commercial, and scientific applications. A prototype of underwater remotely operated vehicle was developed with envisaged application of inspecting subsea environment such as coral reefs [1]. As ocean has disturbances like tides, waves and currents, a suitable motion controlling mechanism is necessary for smooth operation of the ROV. The ROV has five propellers with each propeller having more than one type of movement imparted to the structure. Both stabilizing of the structure and maneuvering of the vehicle is done using the same set of propellers. For motion controlling, a unified controller is introduced which takes the feedbacks from the parameter checking sensors as well as the control commands from the operation and gives control signals to the actuators for both maneuvering and stabilization of the structure at the same time. Creating a suitable controller is made difficult by factors like non-linear dynamics of the ocean environment, presence of disturbances as well as observational noises.

There are several controlling strategies that has been developed for the purpose of motion controlling of ROVs. Among them are, supervisory control [2], neural network control [3], self-turning control [4], LQG/LTR (Linear Quadratic Gaussian with Loop Transfer Recovery) [5] and sliding mode control [6]. Recent interests have been in using fuzzy controllers which are known to be effective robust controllers for a variety of applications. For the fuzzy controller, the dynamics of the system is not needed to be fully known. Thus this becomes a suitable candidate for the purpose of ROV motion control. This paper describes the applicability of fuzzy controller for motion controlling, its implementation and results.

2.BACKGROUND

This ROV was developed in our Department in 2013 [1] as a low cost solution for underwater inspection. Integrating improved functionality to this prototype will result in an economical, compact and application oriented ROV. Developed ROV prototype is shown in FIGURE 1. Five motors used as the propellers are shown with labels.

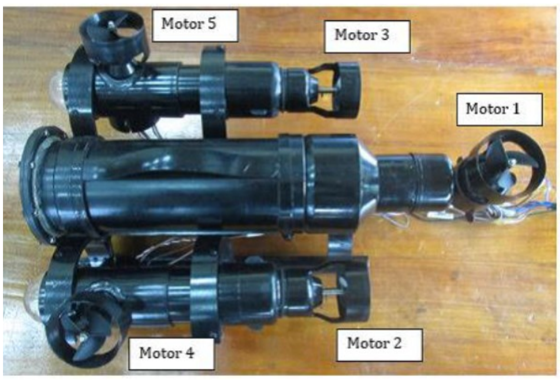


FIGURE 1: ROV prototype with the motors labelled

The main components of the ROV are:

- Five motors with propellers (two horizontal and three vertical)
- High Definition camera
- Accelerometer, Gyroscope, Magnetometer

- Depth sensor
- Ethernet interface (For remote controlling and receiving visual feed & other sensory data)
- Water detector, Temperature sensor
- LED front lights

These components are interconnected using Raspberry Pi, Arduino and network switch modules. Communication methods used are I^2C (Inter – Integrated Circuit), Serial and Ethernet. ROV is connected to a Remote PC using a tether via the Ethernet interface. A client application in the remote PC connects with the ROV. It displays real time data from the sensors of ROV including the live visual feed. Remote controlling of the vehicle is also done using the same application. Either the PC interface or a joystick connected to the PC can be used to control the ROV.

This ROV was designed to be used for subsea inspection. So the medium in which this ROV moves is a dynamic, disturbances prone environment. Ocean contains ripples, waves and currents of various strengths and direction leaving a highly dynamic medium for the ROV. For proper manoeuvring under this environment and keeping the stability of the ROV, good motion control mechanism has to be used. This control system has to reorient the structure against the disturbances and move the structure according to the command signals.

3. CORRECTION PARAMETERS

Considered correction parameters for the ROV are yaw, pitch, roll and depth. ROV has to have movements in forward, backward, lateral and vertical directions according to remote user's command. All these motions are achieved by varying the motor speeds using PWM (pulse width modulation) signals. Respective motors and their directions for each motion is tabulated in TABLE VI.

TABLE 1

MOTOR CONTROLLING DIRECTIONS FOR RESPECTIVE MOTIONS

Motion	Motors	Direction
Forward	2,3	2 and 3 Anticlockwise
Backward	2,3	2 and 3 Clockwise
Upward	1,4,5	1,4 and 5 Clockwise
Downward	1,4,5	1,4 and 5 Anticlockwise
Turn left	2,3	2-Clockwise and 3-Anticlockwise
Turn right	2,3	2-Anticlockwise and 3-Clockwise
Climb	1,4,5	1-Anticlockwise and 4,5-Clockwise
Dive	1,4,5	1-Clockwise and 4,5-Anticlockwise
Roll left	4,5	4-Anticlockwise and 5-Clockwise
Roll right	4,5	4-Clockwise and 5-Anticlockwise

The Table explains the combinations of motors and motor directions required for each motion. In the real scenario, several of these motions has to be applied simultaneously for both error correction and manoeuvring. That makes the number of motor state possibilities higher and each motor has to contribute partially for each simultaneous motion it is involved with. The controller has to respond to both the error amounts measured by the sensors as well as to the remote control commands. The controller must also decide the amounts for each motor that has to be utilized for each required motion and create a unified dynamic motion controlling mechanism for the ROV.

For error correction, errors in the orientation of the structure of the ROV is detected using the magnetometer, gyroscope and accelerometer in the inertial measurement unit. Sensor data contains noise limiting the accuracy of the measurements. These has to be filtered minimizing the degradation of responsively.

For this requirement a MIMO (Multi Input Multi Output) fuzzy controller was developed and implemented on the ROV as its motion controller. Parameters of the controller are adjusted by experimentation for better performance. Effectiveness of this controller on the ROV is evaluated by field testing.

4. FUZZY CONTROLLER DESIGN

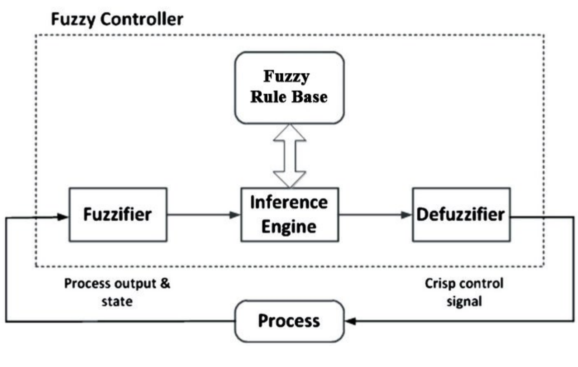


FIGURE 2: Components of a fuzzy controller

A Block diagram of a basic fuzzy controller is shown in FIGURE 2. As illustrated fuzzy controller accepts feedbacks from the process and takes them as inputs. Then it is fuzzified, which is allocating membership values for the inputs values according to predefined fuzzy sets. The fuzzified values are inferred in accordance with the fuzzy rule base, which is also predefined in the controller. The output fuzzy values are then defuzzified in order to get a crisp output value, and a control signal is provided to the process accordingly.

In ROV motion control, the process is the motion of the structure. It has to follow the remote commands of the user and any other undesired movements are taken as errors. Error values are calculated for yaw, pitch, roll and depth in this design and taken as feedbacks from the process to the fuzzy controller. Fuzzy sets, membership functions and rule base is defined according to the requirements. Defuzzified outputs of the fuzzy controller are mapped into the actuators of the design, which are the motors.

Mamdani fuzzy inference method [7] is adapted for this controller. Max-min operators are used for rule inferring. Outputs membership functions are clipped by the evaluated rules rather than scaled. Outputs for each control signal is aggregated and centroid of the resultant is taken as the crisp value for the output.

5. FILTERING SENSOR DATA

Four sensors are used to collect real time data for the fuzzy controller. Data from an accelerometer, magnetometer, gyroscope and pressure sensor is used to calculate yaw, pitch, roll and depth errors. All of this data is scaled to fit into the required range. Offset errors and scale errors were compensated by initial calibration and calculations. Readings from the pressure sensor are amplified for better resolution. All this data contains noise. To reduce the effect of the noise the data is filtered prior to further processing.

Weighted averaging method and Kalman filtering methods were considered for filtering. After testing all methods, the best results were given by one dimensional Kalman filtering for the particular sensor data. A nice feature of this Kalman filter is it can predict future states with the current data. It compensates larger time delays in sensor feedback by predicting intermediate levels.

Following equations are used for one dimensional Kalman filtering of the sensor outputs of ROV.

$$x = x$$

$$p = p + q;$$

$$k = p / (p + r);$$

$$x = x + k * (\text{measurement } x);$$

$$p = (1 - k) * p;$$

The first two formulas represent the prediction of the Kalman Filter. Latter three formulas calculate the measurement update. Variable x is for filtered value, q is for the process noise, r is for the sensor noise, p is for the estimated error and k is for the Kalman Gain. State of the filter is defined by the values of these variables. Those parameters were adjusted to give a clean output without a significant delay. Raw data and its filtered output is shown in FIGURE 3 below.

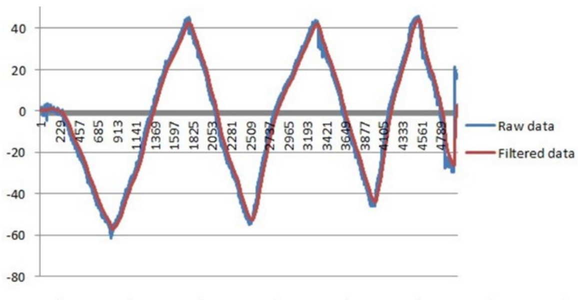


FIGURE 3: Row data and its filtered output

6. FUZZY SUB-CONTROLLERS

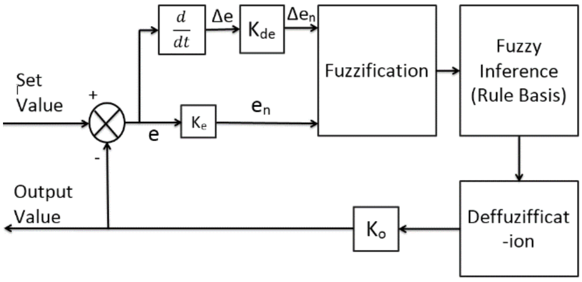
There are four main distinct fuzzy controllers required for error correction. They are,

- Yaw controller
- Pitch controller
- Roll controller
- Depth controller

Same abstract controller is adapted for each of these controllers. Controller takes inputs of error and rate of error change of the considered parameter. Set value of the controller is defined according to the initial sensor readings and the remote control values provided by the user thereafter. Membership functions for the fuzzy sets of error and error rate are defined. Also membership functions for the output are defined. A rule base is created with linguistic terms in accordance with input and output fuzzy sets in this form,

IF (error is negative) AND (error rate is positive) THEN (motor is clockwise); (weight = 0.75)

There are about 10 rules like this for each controller. After inferring the inputs with the rule base, aggregated, defuzzified output is given. Components of the abstract fuzzy controller are



shown in FIGURE 4.

FIGURE 4: Components of the abstract fuzzy controller

If the fuzzy controller for yaw correction is taken as an example, membership functions for the inputs were first selected as triangular functions. With field testing with different shapes, a combination of triangular and Gaussian curves are selected as the optimum membership functions. Selected input membership functions are shown in FIGURE 5 below.

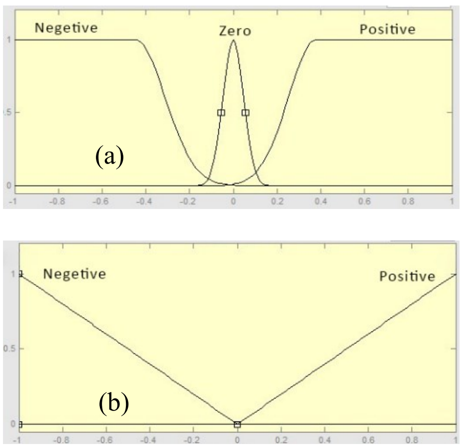


FIGURE 5: (a) Membership Function of Yaw Error, (b) Membership Functions of Yaw Error Rate

Similarly, membership functions for clockwise rotation and anti-clockwise rotation of the outputs are selected. Output of the yaw correction is mapped to motor 2 and motor 3. They are shown in FIGURE 6.

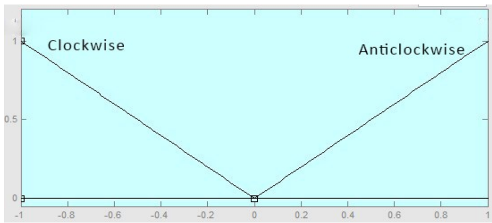


FIGURE 6: Output Membership Functions of Motors

Set value for this yaw correction function is dynamically changed according to the remote commands for left turning and right turning. For pitch, roll and depth similar controllers are constructed according to the respective parameters and motors acting as the actuators. For depth correction error and current vertical acceleration are taken as inputs whereas error and error rate are taken as inputs for pitch and roll.

7. OUTPUT AGGREGATION

In this ROV, several motions are associated with each motor. To achieve unified motion control with both error correction and remote manoeuvring, aggregation of remote commands and fuzzy controller outputs are required. Main output groups mapping to same set of actuators are discussed under following. Components of each group have to be combined in a meaningful manner to give desired motion to the structure.

1) *Forward, lateral movements and yaw correction:* These three movements are achieved using motor 2 and motor 3 of ROV. Lateral movements (left turns and right turns) are integrated to the yaw correction fuzzy controller as the set value. Combination of forward motion has to be done externally. A fuzzy controller is created to share the available motor power for the required functionality. It gives the amount of power which each function can utilize according to the current yaw of structure, commanded angle to turn and current forward speed.

FIGURE 7 shows how the motor power is distributed for each function. Fuzzy controller deciding the amount to utilize is created with input output fuzzy sets as shown in FIGURE 8 and a rule base written for fair allocation.

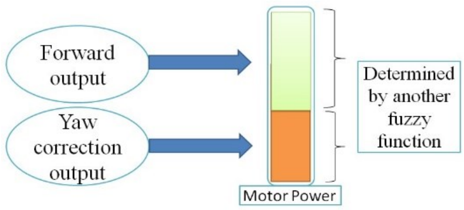


FIGURE 7: Motor power allocation for Forward and Yaw correction

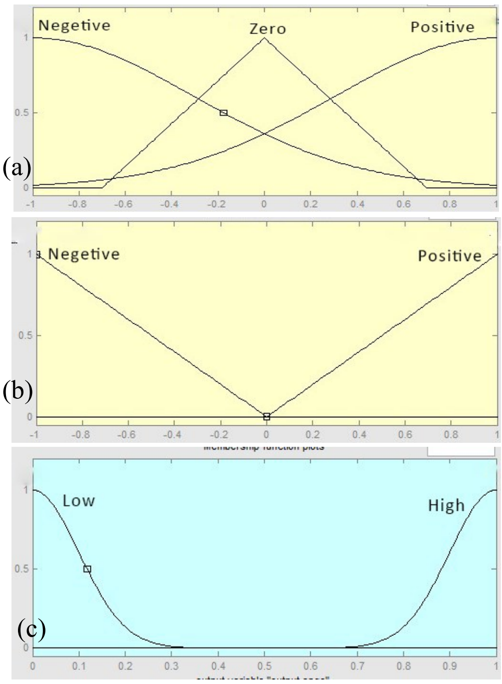


FIGURE 8: (a) Membership Functions for Yaw Error, (b) Membership Functions of Yaw Error Rate, (c) Membership functions for Yaw factor

2) Depth, pitch and roll correction: Motors 1,
And 5 are utilized for each of these movements. As depth is a critical parameter, depth correction is given priority. After allocating motor power to depth correction, rest is distributed for pitch correction and roll correction according to a distribution factor calculated by a separate fuzzy controller as illustrated in FIGURE 9. The Fuzzy controller takes pitch controller output and roll controller output as inputs and calculates the proportion for the two to be allocated for the available motor power. Rule base for this controller is written accordingly. FIGURE 10 shows the input and output fuzzy sets of this controller.

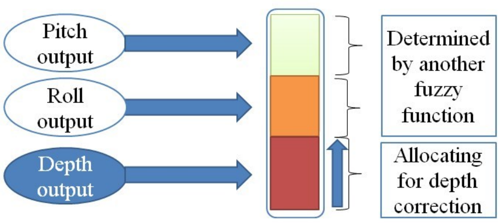


FIGURE 9: Motor power allocation for Pitch, Roll and Depth correction

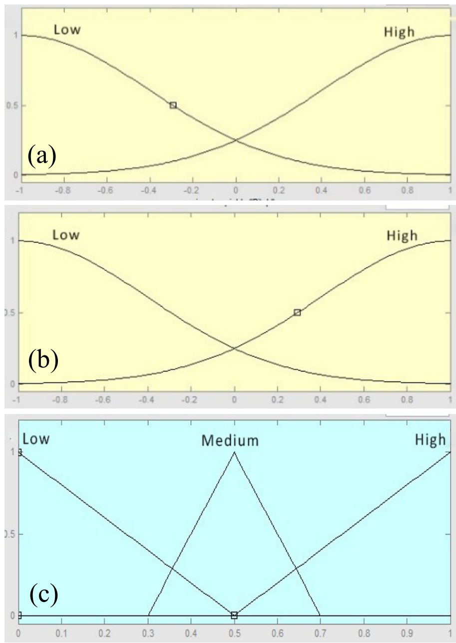


FIGURE 10: (a) Input membership functions for Pitch, (b) Input membership functions for Roll, (c) Output membership functions for the pitch factor

With the combination of all these a fully functional controller is built which takes remote maneuvering commands from user as well as feedbacks from the sensors, take necessary decisions and controls the actuators as a unified fuzzy controller unit.

8. PERFORMANCE

Developed motion controller was first simulated using MatLab. After adjustments it is implemented on ROV using python scripts hosted on the Raspberry Pi module. Computational tools of 'SciPy' python library is used for creating the controller.

It was run without giving disturbances to check if it responds correctly to user commands. Then it is checked for how it responds to disturbances while stationary. Finally, disturbances were given to the structure while it was moving. With field testing fuzzy controller parameters were adjusted for improvements.

Without external forces ROV was responding fast to the manoeuvring signals. Movements were smoother as control commands were also processed by the fuzzy controller in the same manner it responds to a disturbance. When the ROV is commanded to stay still and disturbances were given, it corrected its orientation to its initial state in yaw, pitch, roll and depth as desired. When the ROV is rotated externally at this state, fuzzy controller corrected the yaw as in the graph shown in FIGURE 11.

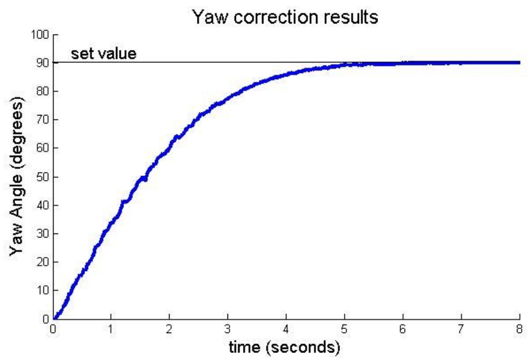


FIGURE 11: Yaw correction results of the fuzzy controller

When disturbances were applied while the ROV was on the move, it adjusts itself with little speed reduction of the original motion at some cases. Every movement including the self-correction ones were relatively smooth, resulting in smooth transitions between motion states of the ROV.

9. CONCLUSIONS

With our fuzzy motion controller, ROV was successfully able to accept and respond to user manoeuvring commands while adjusting it-self when external disturbances are present. This unified controller drove all actuators of the system to achieve the required functionality regardless of the overlapping nature of the control subsystems. Implementing this kind of fuzzy controller would seamlessly improve the performance of a ROV for underwater applications.

10. REFERENCES

[1] Chandimal MP, Chandrasiri HPA, Madusha TPC, Wickramasinghe PGGN, Jayasundere ND, " Underwater Remotely Operated Vehicle", Proceedings Annual Technical Conference 2013 of IET-YP Sri Lanka, 1st March 2013, pp.69-72.

[2] Yoerger, D.R., Newman, J.B., Slotine, J.-J.E., 1986. Supervisory control system for the JASON ROV. IEEE Journal of Oceanic Engineering 11 (3), 392–399

[3] Yuh, J., 1990. A neural net controller for underwater robotic vehicles. IEEE Journal of Oceanic Engineering 15 (3), 161–166

[4] Goheen, K.R., Jefferys, E.R., 1990. Multivariable self-turning autopilots for autonomous underwater vehicles. IEEE Journal of Oceanic Engineering 15 (3), 144–151

[5] Triantafyllou, M.S., Grosenbaugh, M.A., 1991. Robust control for underwater vehicle systems with time delays. IEEE Journal of Oceanic Engineering 16 (1), 146–151.

[6] Christi, R., Papoulias, F.A., Healey, A.J., 1990. Adaptive sliding mode control of autonomous underwater vehicles in the dive plane. IEEE Journal of Oceanic Engineering 15 (3), 152–160.

[7] Ranganath Kothamasu, Samuel H. Huang, 2007, Adaptive Mamdani fuzzy model for condition-based maintenance, Fuzzy Sets and Systems, Volume 158, Issue 24, 2715-27