A new tuning approach of Single Input Fuzzy Logic Controller (SIFLC) for Remotely Operated Vehicle (ROV) Depth Control

Fauzal Naim Zohedi
Center for Robotics and Industrial Automation (CeRIA), Fakulti Kejuruteraan Elektrik, Universiti Teknikal Malaysia Melaka

Mohd Shahrieel Mohd Aras
Center for Robotics and Industrial Automation (CeRIA), Fakulti Kejuruteraan Elektrik, Universiti Teknikal Malaysia Melaka

Hyreil Anuar Kasdirin
Center for Robotics and Industrial Automation (CeRIA), Fakulti Kejuruteraan Elektrik, Universiti Teknikal Malaysia Melaka

Mohd Bazli Bahar
Center for Robotics and Industrial Automation (CeRIA), Fakulti Kejuruteraan Elektrik, Universiti Teknikal Malaysia Melaka

https://doi.org/10.5109/4491657
A new tuning approach of Single Input Fuzzy Logic Controller (SIFLC) for Remotely Operated Vehicle (ROV) Depth Control

Fauzal Naim Zohedi*1, Mohd Shahrieel Mohd Aras1, Hyreil Anuar Kasdirin1, Mohd Bazli Bahar1
1Center for Robotics and Industrial Automation (CeRlA), Fakulti Kejuruteraan Elektrik, Universiti Teknikal Malaysia Melaka, Malaysia.

*Author to whom correspondence should be addressed:
E-mail: fauzal@utem.edu.my
(Received January 7, 2021; Revised September 6, 2021; accepted September 6, 2021).

Abstract: Remotely Operated Vehicle (ROV) is an important vehicle to do underwater task. The uncertainty environment of underwater make it harder for ROV to maneuver and hold position at certain depth. Research on ROV controller for holding position had been conducted. Proportional, Integral and Derivative (PID), Fuzzy Logic Controller (FLC) and Single Input Fuzzy Logic Controller (SIFLC) was designed and compared. This paper discusses the modelling of developed ROV and tuning the SIFLC to get the best transient response. Steady state error (SSE), percent overshoot (%OS), time rise (Tr) and settling time (Ts) were analyzed to select the best controller. The result shows ROV depth can be controlled more precisely using SIFLC with 1.5 %OS, 11.5s Ts and 7.06s Tr.

Keywords: Remotely operated vehicle; depth control; PID controller; FLC controller and SIFLC controller

1. Introduction

Remotely Operated Vehicle (ROV) is an underwater robot that used to replace human in doing underwater task. It plays very important roles in underwater industries and marine activities. The ROV task can be underwater exploration, oil and gas pipeline monitoring, ship hull cleaning and many more3). This complicated task need a skilled ROV operator that have the ability to handle two (2) task simultaneously, maneuvering and manipulation of manipulator2,3). A holding position of ROV need to be achieved to ensure the ROV operator can do manipulation task or underwater observation accurately. Automatic control for holding position is essential to accomplish this3).

Model of the ROV plant need to be developed to implement automatic control. The model is very complex as it has 6 degrees of freedoms. Adding with uncertainty of the underwater environment (hydrodynamics), an accurate model of ROV is very difficult to develop4). In this paper, the model of ROV was generated using System Identification (SI) method. The SI method is based on experimental result that generate the relationship of input and output. Figure 1 shows the ROV that was used to in this research project 5).

Figure 1: ROV used in the research project

Since the invention of ROV, the depth controller had been designed and developed to hold position of the ROV at certain depth. The most common problem in depth control is high overshoot that can harm ROV or its inspected environment. Several basic controllers had been implemented to ROV depth control6)7). There were PID, FLC, Artificial Neural Network (ANN) and Sliding Mode Control (SMC).

S.M. Zanoli et al 8) implemented PID controller to control ROV depth. It can control the ROV but still produce high overshoot. The designed PID was then couple with Continuous Input Smoother (CIS) to cater the problem. It was able to eliminate overshoot but produce slow time response. Z. Tang et al 9,10) state that the simple PID may produce good result but has to tolerate between respond speed and overshoot. S.M. Zanoli et al and Z.
Tang, et al also implement FLC to tuned PID to eliminate overshoot. Result shows a bit oscillation produces before the system stable. A. Nag et al \(^1\) implement FLC to control depth for Autonomous Underwater Vehicle (AUV). The FLC shows adaptability in input changing and very good at tracking. R. Hernández-Alvarado et al \(^2\) use ANN as controller to auto tune PID value to adapt with input changing of ROV depth system. ANN can learn, adapt, and evolve similar as human brain.\(^{13,14}\) Mean square error (MSE) of conventional PID shows higher value compare to neural network auto tuning PID result. SMC was implemented to ROV depth control by B. Sun, et all\(^{15}\). A conventional SMC will keep the system as closely as possible to the sliding surface. It will play with switching surface with bang-bang manner. This manner may produce jitter or chattering effect.\(^{16}\) Table 1 shows the advantages and disadvantages of basic controller implemented.

<table>
<thead>
<tr>
<th>Type of controller</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>PID</td>
<td>Easy to execute and maintain</td>
<td>only for linear system</td>
</tr>
<tr>
<td>FLC</td>
<td>No mathematical modelling required and precise order</td>
<td>complicated in the tuning process</td>
</tr>
<tr>
<td>Neural network control</td>
<td>Convergence to a precise model</td>
<td>slower response</td>
</tr>
<tr>
<td>Sliding mode control</td>
<td>Non-linear system</td>
<td>Energy wastes occur</td>
</tr>
</tbody>
</table>

From the Table 1, there are advantages and disadvantages of all basic controllers. In this research project, Single Input Fuzzy Logic Controller (SIFLC) which based on FLC was selected. It was a simplified version of FLC where the tuning process was easier compared to tuning FLC itself. It also has normal FLC ability which is model free to cope with ROV uncertainty.

This paper is organized as follows. In Section 1, Introduction of the project is presented, and literature review of the project is discussed. Then, Section 2 establishes the model of ROV using System Identification (SI). Section 3 describes the methodology of the project where controllers designed are discussed. Section 4 illustrate the simulation and analysis of the results. Finally, the final remarks are elucidated in Section 5.

### 2. System Modelling of ROV

Modelling of ROV plant was generated using SI approach. This approach was based on experimental input output result.\(^{17,18}\) From the experimental input output data, a relationship between input output was generated in transfer function. Five (5) steps need to be followed to implement SI approach\(^{19}\). Figure 2 shows the flow chart of SI approach.

![Flow chart of SI approach](image)

Start with system observation and data gathering, ROV system is observed and the data is gathered. Two sets of input output data were gathered. One (1) set for training and another for validation. The input given to ROV system can be pulse, steps, Random Binary Sequence (RBS), Pseudo Random Binary (PRBS), m-level Pseudo Random (m-PRS) and multi-sine.\(^{19}\) In this project, multi-sine input was chosen. Then model structure was selected. As MATLAB was used in this project, the Instrument Variable (IV) approach was selected. Next, the selected model structure is implemented for model estimation and model validation to generate a ROV model. Lastly, the model generated is used to design ROV controller.

For this developed ROV, pressure sensor was used to measure depth of the ROV. This was based on the pressure depth equation as shown in Equation (1).

\[
P = \rho gh
\]

Pressure due to weight of liquid is given by Eq. (1), where \(P\) is pressure, \(\rho \) (103kg/m\(^3\) or g/mL) is the density of the liquid (water = 1 and sea water = 1.025), \(g\) is the acceleration due to gravity (g=9.80m/s\(^2\)) and \(h\) is the height of the liquid.

The deeper the ROV gone, the more pressure will be
A new tuning approach of Single Input Fuzzy Logic Controller (SIFLC) for Remotely Operated Vehicle (ROV) Depth Control

produced. From this relation, and input output experimental data was gathered where the offset is set to 1.5m. Three (3) sinusoid inputs with difference frequencies; 0.5, 2 and 5 rad/s were given to the ROV system. To gain an ideal result, the experiment was conducted in a controlled environment. Disturbance was not considered. The data was shown as Figure 3.

The input data was the depth set point while the output data is the the actual depth. From the data shown Figure 2, transfer function of the system was generated using MATLAB System Identification (SI) method was shown in Eq. (2). The setup selected was instrument variable approach; IV and 3 poles and 2 zeros transfer function. The best fitting match shown 96.43%. It was acceptable because within 80% to 99% best fits.

$$H(S) = \frac{0.02332s^2+0.04058s+0.01126}{s^3+0.7114s^2+0.1861s+0.01398}$$ (2)

The generated output transient response was shown in Figure 4.

The generated transient response was then analyzed in term of percent overshoot (%OS), steady state error (SSE), rise time (Tr) and settling time (Ts). The data was tabulated in Table 2.

<table>
<thead>
<tr>
<th>%OS</th>
<th>Tr(s)</th>
<th>Ts(s)</th>
<th>SSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original</td>
<td>0</td>
<td>17.178</td>
<td>40.16</td>
</tr>
</tbody>
</table>

3. Controller Design

Three approaches for Controller design were PID controller, Fuzzy Logic controller (FLC) and Single-Input FLC (SIFLC). To implement the controllers, the developed ROV system was studied. The developed ROV uses 12V DC motor thruster to maneuver from one place to another. It was an under actuated ROV as it contained only 4 thrusters. 2 thrusters for forward or reverse (surge) and turn right or left (yaw). Another 2 thrusters use for dive and emerge (heave). The depth set point given to the ROV system and maneuver the ROV to the specific depth. Feedback given by pressure sensor sense in voltage was converted to depth and compared with input setpoint. Figure 5 shows a closed loop transient response for the plan. Table 3 shows the closed loop ROV performances in terms of OS%, Tr, Ts and SSE.

Even the close loop response has better Ts and Tr compared to open loop, it has higher SSE compare the open loop system as shown in Table 2 and Table 3. It does not even reach half of the set point given. Based on the errors in the open loop and close loop result, PID, FLC and SIFLC was designed.

3.1. PID controller

Conventional PID controller is the most common or basic controller used to ROV system. The P, I, and D blocks were put in parallel in front of the plan to control the system. The P counter the direct error; the I indicate the total errors in the system while D shows how fast to the errors happen. The P controller will make the response faster but intend to produce overshoot. The I controller tend to eliminate SSE while the D controller decrease overshoot. The summarize effect of PID is shown in Table 4.
### Table 4: Effect of increasing parameters of PID

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Tr</th>
<th>%OS</th>
<th>Ts</th>
<th>SSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kp</td>
<td>Decrease</td>
<td>Increase</td>
<td>A bit change</td>
<td>Decrease</td>
</tr>
<tr>
<td>Ki</td>
<td>Decrease</td>
<td>Increase</td>
<td>Increase</td>
<td>Eliminate</td>
</tr>
<tr>
<td>Kd</td>
<td>A bit change</td>
<td>Decrease</td>
<td>Decrease</td>
<td>No effect</td>
</tr>
</tbody>
</table>

The PID controller block diagram is shown as in Figure 6. The P, I and D are connected in parallel between each other and placed in series within the depth block diagram. The PID was tuned using automatic tuning in MATLAB Simulink\(^{20}\).

![Figure 6: PID controller block diagram.](image)

### 3.2. Fuzzy Logic Controller (FLC)

FLC was introduced by Lotfi A Zadeh in 1965 \(^{21}\). This controller was based on human decision making and can implement linguistic decision. \(^{22}\) There are 4 basic components in FLC shown in Figure 7. \(^{23}\)\(^{24}\)

![Figure 7: Basic configuration of FLC components.](image)

Two (2) inputs were selected which are errors and differential errors or errors rate. The rules for decision making of FLC can be varies from 3 X 3 to 7 X 7 or even more. The less numbers of rules may lessen the computational time and faster the output result. In this project 7 X 7 fuzzy rules were selected. Table 5 shows the rules table for depth control of ROV.

![Figure 9: Derivation of d, distance between point Q and P\(^{25}\).](image)

\[
e' + (e)\lambda = 0 \quad (3)
\]
\[
d = \left( e(k) \times \frac{\lambda}{\sqrt{1+\lambda^2}} \right) + \left( e'(k) \times \frac{\lambda}{\sqrt{1+\lambda^2}} \right) \quad (4)
\]

### 3.3. Single Input FLC (SIFLC)

SIFLC is an improvisation of FLC to make it simplified and easier to tuned. It has only one (1) input compare to two (2) input that FLC used. Theoretically SIFLC output result should be identical to FLC. Figure 8 shows the basic block diagram of SIFLC \(^{25}\)\(^{26}\).

![Figure 8: Basic SIFLC block diagram.](image)

Two (2) input of FLC; error and differential of error were converted to d using signed distance method (SDM). ‘d’ is the distance between two (2) points; Q and P as shown in Figure 9. Line ‘a’ and ‘b’ are the line created based on pattern in the Table 5. ‘a’ is the main diagonal line while ‘b’ is the first 1st diagonal line to the right. Eq. (3) is the equation of the main diagonal line.

### Table 5: Rules table for depth control of ROV

<table>
<thead>
<tr>
<th>err vs du/dt</th>
<th>PL</th>
<th>PM</th>
<th>PS</th>
<th>Z</th>
<th>NS</th>
<th>NM</th>
<th>NL</th>
</tr>
</thead>
<tbody>
<tr>
<td>NL</td>
<td>Z</td>
<td>NS</td>
<td>NM</td>
<td>NL</td>
<td>NL</td>
<td>NL</td>
<td>NL</td>
</tr>
<tr>
<td>NM</td>
<td>PS</td>
<td>Z</td>
<td>NS</td>
<td>NM</td>
<td>NL</td>
<td>NL</td>
<td>NL</td>
</tr>
<tr>
<td>NS</td>
<td>PM</td>
<td>PS</td>
<td>Z</td>
<td>NS</td>
<td>NM</td>
<td>NL</td>
<td>NL</td>
</tr>
<tr>
<td>Z</td>
<td>PL</td>
<td>PM</td>
<td>PS</td>
<td>Z</td>
<td>NS</td>
<td>NM</td>
<td>NL</td>
</tr>
<tr>
<td>PS</td>
<td>PL</td>
<td>PL</td>
<td>PL</td>
<td>PL</td>
<td>PS</td>
<td>Z</td>
<td>NS</td>
</tr>
<tr>
<td>PM</td>
<td>PL</td>
<td>PL</td>
<td>PL</td>
<td>PL</td>
<td>PL</td>
<td>PL</td>
<td>PL</td>
</tr>
<tr>
<td>PL</td>
<td>PL</td>
<td>PL</td>
<td>PL</td>
<td>PL</td>
<td>PL</td>
<td>PL</td>
<td>PL</td>
</tr>
</tbody>
</table>
output transient result of SIFLC. The \((\lambda)\) linked to the FLC by the input of the FLC. The range of error and integral error was plotted in a graph shown in Figure 10.

![Figure 10: Plotted graph of input 2 versus input 1 FLC.](image)

The varying of \((\lambda)\) SIFLC result was then analyzed, and the best result was selected. Then the best gradient was used to redesign the input range of FLC to get better result of FLC.

Figure 11 shows the simulation block diagram for this research project. Six (6) signals were compared to analyze the best controller for the ROV. The signals were step input, open loop, closed loop, PID, FLC and SIFLC.

![Figure 11: Block diagram for step input, open loop, closed loop, PID, FLC and SIFLC.](image)

The highlighted block in Figure 11 was the block that affected by the retuned of \((\lambda)\) value. The retuning of \((\lambda)\) value affected the ‘d’ value as Eq. (4). Due to that, the truth table of ‘d’ versus output was also affected. Lookup table in the block diagram need to be change based on the retuned ‘d’ value. For the FLC, once SIFLC tuning was done, the range of error and integral error was adjusted as the \((\lambda)\) was basically the gradient of input 1 and input 2 in FLC. Figure 12 shows the flow diagram of the tuning process.

![Figure 12: Flow diagram for SIFLC and FLC tuning](image)

<table>
<thead>
<tr>
<th>Table 6: ‘d’ truth table before tuned</th>
</tr>
</thead>
<tbody>
<tr>
<td>In 1</td>
</tr>
<tr>
<td>In 2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 7: ‘d’ truth table after tuned</th>
</tr>
</thead>
<tbody>
<tr>
<td>In 1</td>
</tr>
<tr>
<td>In 2</td>
</tr>
</tbody>
</table>

4. Results and Discussion

Figure 13 shows the result of all 6 signals implemented to the system before the tuning of SIFLC implemented.

![Figure 13: 6 signals implemented to the system before the tuning of SIFLC implemented](image)
Directly from Figure 13 can be seen that the FLC and SIFLC have the highest %OS. It was led by SIFLC. Close loop and open loop show the highest SSE produced. PID shows the most relevant controller shown but slower in term of Tr. Then, the SIFLC was retuned by using heuristic method where the gradient ($\lambda$) was varies. Next based on SIFLC result, FLC was retuned. The result was shown as below in Figure 14.

![Figure 14: Retuned of SIFLC and FLC result](image)

### Table 6: Performance of PID and SIFLC controller

<table>
<thead>
<tr>
<th>Controller</th>
<th>%OS</th>
<th>Tr(s)</th>
<th>Ts(s)</th>
<th>SSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>PID</td>
<td>6.989</td>
<td>7.07</td>
<td>27.1</td>
<td>0.0017</td>
</tr>
<tr>
<td>FLC</td>
<td>24.375</td>
<td>3.03</td>
<td>15.7</td>
<td>0.0011</td>
</tr>
<tr>
<td>SIFLC</td>
<td>27.564</td>
<td>2.983</td>
<td>18.15</td>
<td>0.0109</td>
</tr>
<tr>
<td>FLC (new)</td>
<td>5.851</td>
<td>5.245</td>
<td>18.9</td>
<td>0.0131</td>
</tr>
<tr>
<td>SIFLC (new)</td>
<td>1.531</td>
<td>7.057</td>
<td>11.52</td>
<td>0.0027</td>
</tr>
</tbody>
</table>

The performance of all controllers was tabulated in Table 6. The new tuned SIFLC and FLC shows tremendous recovery in the %OS but compensated by Tr value. In terms of Ts value, SIFLC lead all other controllers. For SSE, all controllers produced acceptable result and compensate either the original system or close loop system.

From result, all controllers simulated can be used for ROV control, however %OS value produced may damage the ROV or its investigated environment. From all presented result, retuned SIFLC is the best as it has the best %OS. PID controller is stronger in linear systems, but FLC is suitable for linear and non-linear system. SIFLC is adapting the FLC ability but with single input. SIFLC also shows better stability and overshoot. In terms of tuning aspect, PID is easier compared to FLC but SIFLC seems to simplify complexity of tuning the FLC because of the single input.

### 5. Conclusion

Three controllers had been successfully implemented to ROV depth system. SIFLC shows the best performance as it eliminates %OS and have the best Ts value. It adapts FLC ability which are model free and adaptable to cater disturbance and non-linearity system. The single input simplifies the FLC tuning process. Tuning the $\lambda$ value is proven can improve the transient response of SIFLC and can be used to retuned FLC. Compared to PID, it was easy to tune but highly depends on linear model. For future research, it the controller will be tested experimentally, and optimization will be embedded in the system to cater the varies effect of set point.

### Acknowledgement

We wish to express our gratitude and appreciation the support granted by Universiti Teknikal Malaysia Melaka in pursuing this research specially to Underwater Technology Research Group, Faculty of Electrical Engineering, Centre of Research, and Innovation Management for supporting this research.

### References


A new tuning approach of Single Input Fuzzy Logic Controller (SIFLC) for Remotely Operated Vehicle (ROV) Depth Control


