

# Control of Non-linear Equation of Submarine Using PI-like Fuzzy Controller

Matheus F. Mollon<sup>1</sup>, Eduardo H. Kaneko<sup>2</sup>, Wagner de S. Chaves<sup>3</sup>, Lucas Niro<sup>4</sup>, Marcio A. F. Montezuma<sup>5</sup>

<sup>1,2,3</sup>Graduate Student, University of Technology - Paraná, CornélioProcópio, Paraná, Brazil

<sup>4,5</sup>Department of Mechanical Engineering, University of Technology - Paraná, CornélioProcópio, Paraná, Brazil

**Abstract**—This paper presents a methodology to facilitate the development of a Fuzzy Logic Controller (FLC). The process is done through the use of model rule base, in addition to the adjustment of the membership functions and the universe of discourse related to the system variables. This adjustment is accomplished by refinement rules similar to those used in a classical control technique. In this way, PID control system designers can use their knowledge of parameter tuning to obtain the desired performance in the Fuzzy system. This knowledge allied to the model rule base contributes to decrease the controller development time. Therefore, as an application of the method, a non-linear model of a submarine is used to evaluate the performance of the FLC. A model rule base developed by MacVicar-Whelan and the basic rules of refinement elaborated by Procyk and Mandani are applied for the development of the FLC. The simulations are performed through MATLAB software and the FLC is developed with the Fuzzy Logic Toolbox.

**Keywords**—Control Systems, Fuzzy Logic Control, PI-like Fuzzy Controller.

## I. INTRODUCTION

Artificial Neural Networks (ANN) and Fuzzy Logic have been successfully used in several types of applications [1], such as the control of pollutant emissions by internal combustion engines [2], temperature control in incubators [3], speed control of an induction motor [4], petrochemical cracking [5], among others.

Fuzzy logic controllers (FLCs) are rules-based systems that use fuzzy linguistic variables to model human approaches, in order to solve problems without a mathematical modeling [6]. However, for its construction, the rule base is crucial, being the most difficult aspect of the FLC project, since there is no systematic tool for the creation of the rule base [7]. In this context, there are two main methods for building the rule base. The first one is based on the intuition and experience of the designer, where the FLC is designed as a specialist system, allowing the addition of custom rules by experience in the control strategy. Usually, it is difficult to extract the

operator control skills that can be used to build the FLC rule base. Besides, there is no reason to believe that these rules are the best control strategy. Then, most FLCs combine an approach based on operator experience with a good understanding of the system and control theory [7].

The second way to build the FLC rule base is based on the use of a model rule base, which is considered as a basic tool that combines common sense engineering and experience in FLC controllers. Thus, with this method, the implementation of a Fuzzy Logic Controller could be similar to a traditional PID Controller, in addition to maintaining the simplicity of PID controller implementation with the performance of a non-linear controller [7].

The dynamics of a submarine is non-linear and complex, whose operating conditions change according to sea currents, depth variation, among other factors [8]. Some works as in [8, 9] provide intelligent control techniques applied to submarine control, such as the fuzzy controller, which is used in [8] for submarine steering control, and in [9] for control of the submarine depth.

Based on this context, the objective of this project is to implement a FLC that is effective and robust, through the use of a model rule base and techniques for controller improvement that resemble those used in PID tuning, with the advantage of perform a non-linear controller. The FLC is developed to control the speed of a submarine represented by a non-linear equation. The methodology for implementing the controller is detailed to allow its use in other types of models and systems as a guide, since each type of application requires specific adjustments. In general, the developed FLC presented satisfactory results in relation to speed control and disturbance tolerance in the submarine case.

The work is organized as follows. Section II shows the structure of FLC controller. Section III describes the FLC adjustments and simulation of the submarine's non-linear equation. In section IV it is presented the conclusions about this work, as well as, its limitations and future works.

## II. STRUCTURE OF FUZZY LOGIC CONTROLLER

The control law of FLC is described as a knowledge base composed of IF...THEN rules with vague characteristics and an inference mechanism based on Fuzzy logic. The rule base is the most important part of the controller, since it is formed by a family of logic rules that describe the relations between input  $e(k)$  and output  $u(k)$  of the controller. The rule base suggested by MacVicar-Whelan [10] is a good example of a model rule base, being developed in order to complete the traditional rule base focused on the designer's experience. In addition, with the MacVicar-Whelan rule base it is possible to add special situations. This expansion of the rule base is governed by the three fundamentals described below [7, 11]:

1. If the error  $e(k)$  and the change of error  $\Delta e(k)$  have their values at zero, then the present control is maintained;
2. If the error  $e(k)$  tends to zero at a satisfactory rate, then the present control is maintained;
3. If the error  $e(k)$  is not self correcting, then the change of control action  $\Delta u(k)$  is not zero and depends on the signal and magnitude of  $e(k)$  and  $\Delta e(k)$ .

The error  $e(k)$  is defined as the difference between the desired value  $r(k)$  and the output value  $y(k)$  of the control plant, as shown in (1).

$$e(k) = r(k) - y(k) \dots \dots \dots (1)$$

The change of error  $\Delta e(k)$  is presented in (2).

$$\Delta e(k) = e(k) - e(k-1) \dots \dots \dots (2)$$

and the change of the control action  $\Delta u(k)$  is presented in (3).

$$\Delta u(k) = u(k) - u(k-1) \dots \dots \dots (3)$$

where notation  $(k-1)$  indicates the previous value or sample.

The input variables of the FLC are error  $e(k)$  and error change  $\Delta e(k)$ . The output variable is the change in the control action  $\Delta u(k)$ . This controller is similar to the PI controller, since an integrator is added to the output of the controller to obtain the control action  $u(k)$  through  $\Delta u(k)$ . Fig.1 shows the structure of the Mandani FLC controller used in this project.

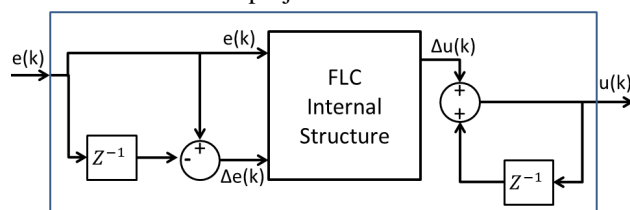


Fig.1: Structure of the FLC controller.

The MacVicar-Whelan rule base extracted from [10] and used in this work is presented in Table 1. The first column and row are the linguistic variables for the system error

$e(k)$  and error change  $\Delta e(k)$ , respectively. The relationship between  $e(k)$  and  $\Delta e(k)$  with the control action change  $\Delta u(k)$  is done by the following logic:

*RI:* If error  $e(k)$  is **A** and change of error  $\Delta e(k)$  is **B**, then change of control action  $\Delta u(k)$  is **C**.

In *RI*, **A**, **B** and **C** are the linguistic variables present in Table 1. The intensity **A** is a value of the first row, **B** is a value of the first column and **C** is the intersection between the row and column equivalent to values **A** and **B**, that results in the intensity of the control action  $\Delta u(k)$ . The relationship between the notation and the meaning of each linguistic variable is shown in Table 2.

Table 1: Model rule base proposed by MacVicar-Whelan.

$\Delta e(k) \backslash e(k)$	NL	NM	NS	Z	PS	PM	PL
NL	NL	NL	NL	NL	NM	NS	Z
NM	NL	NL	NM	NM	NS	Z	PS
NS	NL	NM	NS	NS	Z	PS	PM
Z	NL	NM	NS	Z	PS	PM	PL
PS	NM	NS	Z	PS	PS	PM	PL
PM	NS	Z	PS	PM	PM	PL	PL
PL	Z	PS	PM	PL	PL	PL	PL

Table 2: Definition of the linguistic variables.

Term	Definition
NL	Negative Large
NM	Negative Medium
NS	Negative Small
Z	Zero
PS	Positive Small
PM	Positive Medium
PL	Positive Large

## III. SIMULATION AND CONTROLLER ADJUSTMENT

The non-linear model for the velocity response of the submarine in relation to the applied acceleration or thrust is illustrated in (4), where  $\dot{v}$  is the velocity time derivative,  $v$  is the submarine velocity in m/s and  $T$  is the applied thrust or acceleration in  $\text{m/s}^2$ .

$$\dot{v} + \|v\|v = T \dots \dots \dots (4)$$

With the submarine model in (4) and the controller structure defined in Section II, it is necessary to select the operating range for each system variable. The variable  $e(k)$  must have an operating range of  $[-50, +50]$ , because the adopted velocity is 50 m/s and the initial velocity could be zero. Thus, it is necessary that the variable  $e(k)$  is at least in this operating range, according to (1).

For the error change  $\Delta e(k)$  the same range of error  $e(k)$  could be considered, but as we use a sample rate of 0.1 seconds, there is no need for such a large operating range.

Therefore, a smaller range of  $[-10, +10]$  is adopted. For the change of acceleration  $\Delta u(k)$ , an operating range of  $[-10, +10]$  has been defined so that there is no high rise time in the system response. Normalizing the operation ranges for  $[-1, 1]$ , we obtain the following normalization coefficients:  $k_e = 1/50$  for  $e(k)$ ,  $k_{de} = 1/10$  for  $\Delta e(k)$  and  $k_{du} = 10$  for  $\Delta u(k)$ . However, these coefficients are used only for the appliance of the refinement rules based on [12]. The changes in the variables operating ranges are performed based on these coefficients, but without the normalization in the FLC variables. The control system developed in MATLAB for the submarine is illustrated in Fig.2, where  $r(t)$  is the setpoint defined by the operator,  $y(t)$  is the speed of the submarine and  $d(t)$  is the disturbance applied to the system.

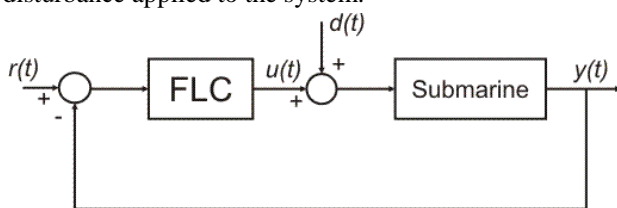


Fig. 2: Fuzzy logic control system.

With the control system developed in MATLAB software, it is simulated and its performance is refined until an acceptable solution is found, but it probably is not the best one. In addition, the initial condition considered is initial velocity and acceleration equal to zero or  $v(0) = 0$  and  $T(0) = 0$ , respectively.

The parameters used to evaluate the FLC controller are the same as those used in the classical PID control. In this project, the FLC refinement method in relation to the speed response is applied by analyzing the following parameters: rise time ( $t_r$ ), settling time ( $t_s$ ), overshoot ( $M_o$ ), maximum peak over setpoint ( $M_{pk}$ ), Integral of Time multiplied by Absolute of the Error (ITAE) and Integral Absolute Error (IAE). For the settling time, a tolerance band was defined around  $\pm 2\%$  of the setpoint. In addition, the calculation of  $M_{pk}$  is performed by subtracting the largest value from the response (maximum peak) in relation to the setpoint. Therefore, this value represents the excess speed corresponding to the overshoot percentage.

For the first simulation scenario (C1), the membership functions for each input and output variables were equally distributed within the operating range, according to the linguistic variables defined in Table 1. Fig. 3, 4 and 5, present the membership functions for  $e(k)$ ,  $\Delta e(k)$  and  $\Delta u(k)$ , respectively. Then, simulating the system for this scenario, we obtained the answers presented in the charts of Fig. 6a.

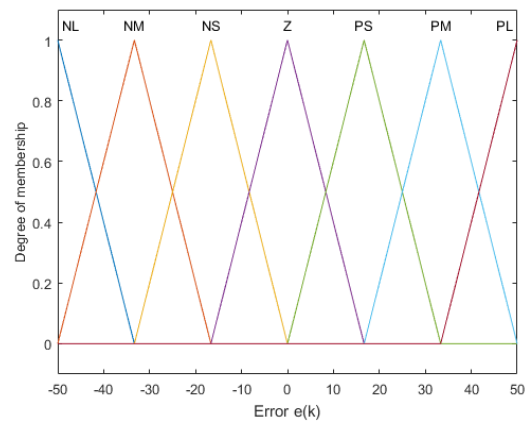


Fig. 3: Membership functions for  $e(k)$ .

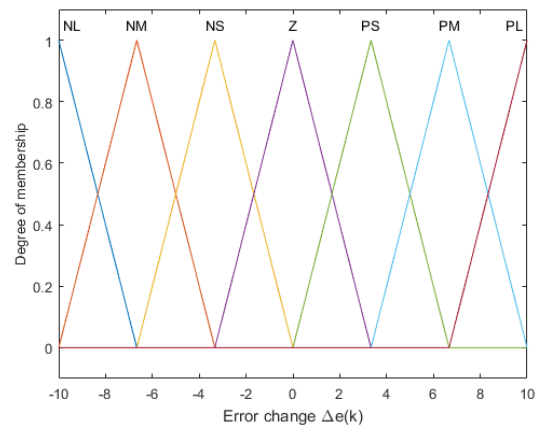


Fig. 4: Membership functions for  $\Delta e(k)$ .

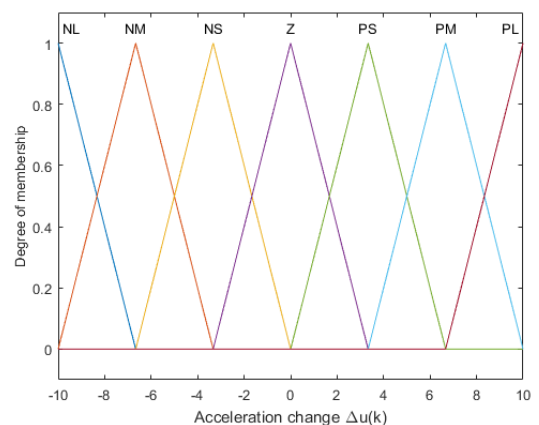


Fig. 5: Membership functions for  $\Delta u(k)$ .

Based on the charts shown in Fig. 6a, it can be verified that the speed response of the system presents great overshoot and high settling time, according to the calculated parameters in the second column of Table 3. Then, as a way of improving it, some basic rules of refinement developed by Procyk and Mandani are used [7, 12, 13]:

1. High  $k_e$  values result in good system response (low steady-state error and low rise time), but produce low stability (large overshoot). On the other hand, low values of  $k_e$  result in poor response;

2. A fast convergence is limited by high values of  $k_e$  and  $k_{de}$ , in addition to relatively low values of  $k_{de}$ ;
3. Low values of  $k_{du}$  increase the rise time and the integral of the quadratic error;
4. Small values of  $k_{de}$  results in overshoots and oscillations.

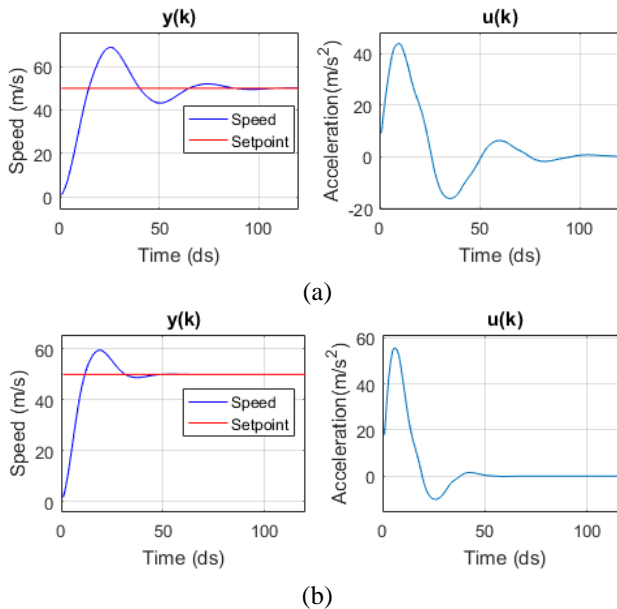


Fig. 6: System responses for speed and acceleration: (a) Scenario 1; (b) Scenario 2.

Based on the refinement rules, we can perform the increase of  $k_{du}$ , as in scenario 2. In this way, the value of  $k_{du}$  is changed from 10 to 20, resulting in a universe of discourse of  $[-20, 20]$  for  $\Delta u(k)$ . Simulating the response of the system again (Scenario C2), we obtained the responses of the submarine behavior shown in Fig. 6b, where the values of the refinement parameters are in the third column of Table 3. Based on these values, there are smaller values for overshoot, settling time, ITAE and IAE, which indicates that the refinement action was effective in improving the system response. In order to improve this response again, however, without changing the obtained characteristics and increasing its stability, it is necessary to change the membership functions of the error variable  $e(k)$  as shown in Fig. 7a.

Table 3: Evaluation parameters for Scenarios 1 and 2.

Parameters	Scenario 1	Scenario 2
$t_r(s)$	0.99	0.77
$t_s(s)$	8.12	4.07
$M_o(\%)$	37.31	19.21
$M_{pk}(m/s)$	18.85	9.61
ITAE	1790.22	387.13
IAE	817.05	399.90

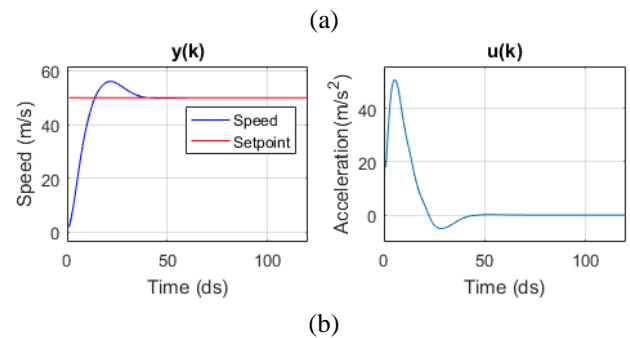
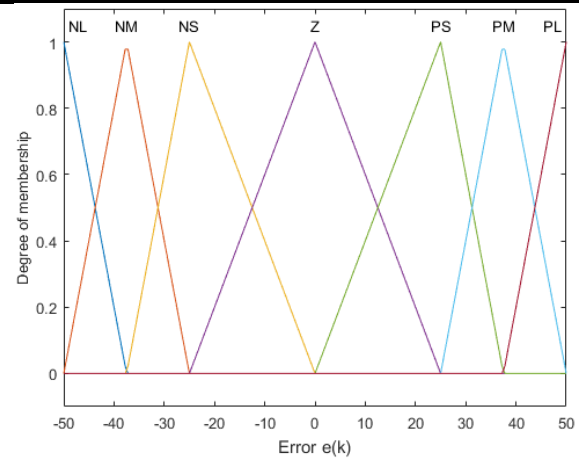


Fig. 7: Scenario 3: (a) membership functions of the error variable  $e(k)$ ; (b) System responses.

After the changes in the error variable, the system was simulated again (Scenario 3), in which the answers obtained are in Fig. 7b and the results for the evaluation parameters are in the second column of Table 4. In comparison with the responses of the scenario 2, there was a lower overshoot, shorter settling time, lower ITAE and IAE, but the rise time was higher. So, as a refinement action to improve the simulation response in Scenario 4, the peaks of the membership functions  $\Delta u(k)$  are displaced, as shown in Fig. 8a. The results of the simulation are presented in Fig. 8b and the evaluation parameters are in Table 4 (third column).

Table 4: Evaluation parameters for Scenarios 3 and 4.

Parameters	Scenario 3	Scenario 4
$t_r(s)$	0.93	0.90
$t_s(s)$	3.47	3.29
$M_o(\%)$	12.15	9.30
$M_{pk}(m/s)$	6.07	4.65
ITAE	342.34	278.20
IAE	385.59	354.29



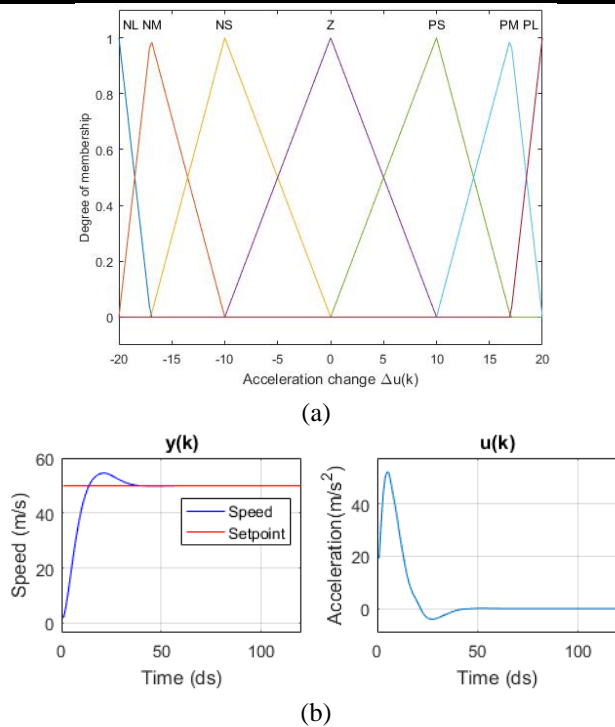
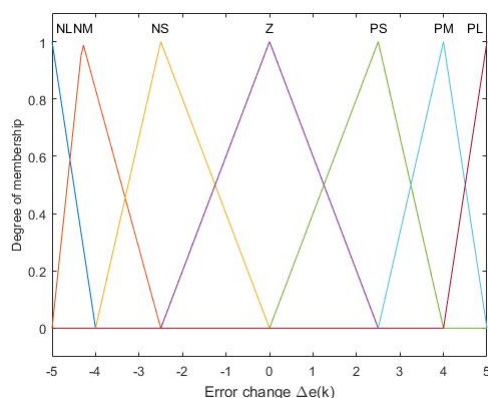


Fig. 8: Scenario 4: (a) membership functions of the acceleration change  $\Delta u(k)$ ; (b) System responses.

In comparison with Scenario 3, the system response in relation to speed presented a small improvement in all the calculated parameters, with a reduction of 2.85% in overshoot. As a refinement action to improve the response to Scenario 5 and eliminate overshoot, it is necessary to increase the value of  $k_e$  from 1/10 to 1/5, in other words, reduce the interval from  $\Delta e(k)$  to  $[-5, 5]$ , as shown in Fig. 9a. The simulation result is shown in Fig. 9b and the calculated parameters are in the second column of Table 5.

Table 5: Evaluation parameters for Scenarios 5 and 6.

Parameters	Scenario 5	Scenario 6
$t_r(s)$	2.59	1,54
$t_s(s)$	4.96	2,49
$M_o(\%)$	0.00	0,00
$M_{pk}(m/s)$	0.00	0,00



ITAE	763.76	290,48
IAE	595.98	411,35

According to the parameters obtained and Fig. 9b, the response for velocity has a high rise time and presents an overdamped characteristic. An interesting factor is that  $\Delta e(k)$  influences the acceleration in this project. The change of the error  $\Delta e(k)$  with sample time of 0.1 s produces a triangle where the opposite leg is  $\Delta e(k)$  and the adjacent leg is the sample time. The tangent of this angle is the speed change rate that results in the acceleration  $u(k)$ . Thus, it is possible to avoid overshoot by decreasing the discourse universe of  $\Delta e(k)$ , which limits the acceleration rise. However, with this limitation the system presents a slow response. Therefore, to increase the system speed response, the peaks of the membership functions of the  $\Delta e(k)$  variable must be displaced, as shown in Fig. 10a. After performing this change, the system is simulated in Scenario 6.

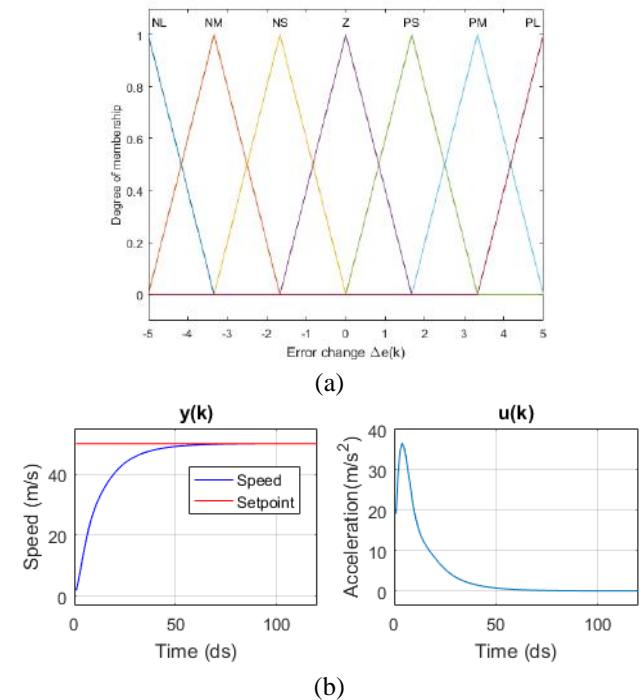


Fig. 9: Scenario 5: (a) membership functions of the error change  $\Delta e(k)$ ; (b) System responses.

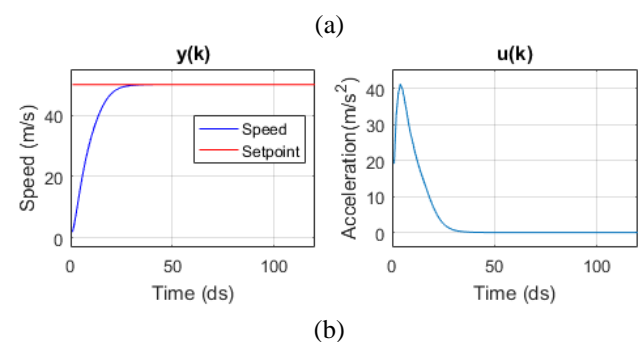


Fig. 10: Scenario 6: (a) membership functions of the error change  $\Delta e(k)$ ; (b) System responses.

The evaluation parameters for the scenario 6 are in the third column of Table 5. Therefore, there was a significant improvement in settling time, and the characteristics desired by the designer were reached. A smaller rise time would result in a higher power value, which is not acceptable for the motors in our hypothetical scenario. Finally, the system is simulated with a disturbance  $d(t)$  which may be a sea current, or something that collides against the submarine's hull. For this, a random disturbance with a maximum amplitude of  $\pm 1 \text{ m/s}^2$  was inserted and the result is shown in Fig. 11. In general, the system tends to correct the disturbances and to maintain the submarine speed in 50 m/s.

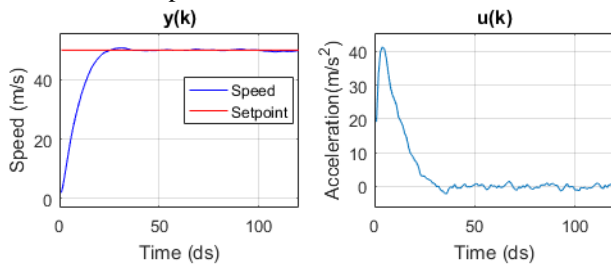


Fig. 11: System response with random noise.

In Table 6 are all the simulation scenarios performed, as well as the parameters obtained and the indication of the refinement action applied for the subsequent scenario. Fig. 12 shows the evolution of the speed response for each scenario.

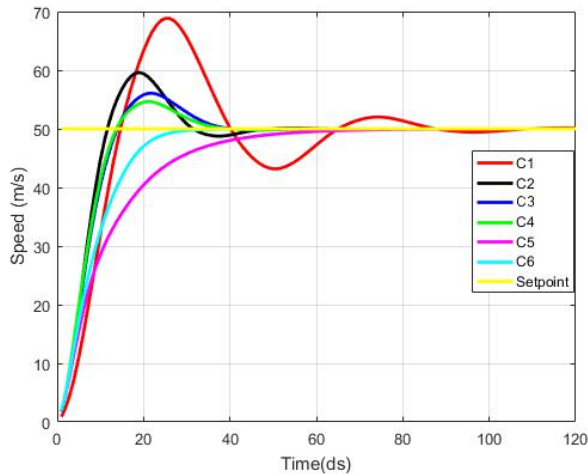


Table 6: Simulation scenarios, evaluation parameters and refinement action.

Scenario	$t_r(s)$	$t_s(s)$	$M_o(\%)$	$M_{pk}(m/s)$	ITAE	IAE	Refinement
C1	0,99	8,12	37,31	18,85	1790,22	817,05	Increase in the range of $\Delta u(k)$ .
C2	0,77	4,07	19,21	9,61	387,13	399,90	Change the membership functions of $e(k)$
C3	0,93	3,47	12,15	6,07	342,34	385,59	Change the membership functions of $\Delta u(k)$
C4	0,90	3,29	9,30	4,65	278,20	354,29	Reduction of the $\Delta e(k)$ interval.
C5	2,59	4,96	0,00	0,00	763,76	595,98	Change the membership functions of $\Delta e(k)$
C6	1,54	2,49	0,00	0,00	290,48	411,35	End

Fig. 12: Evolution of the speed response in the scenarios. Finally, Fig. 13 shows the Fuzzy surface obtained through the controller rules base, where the x and y axes represent the operating range of the input variables  $e(k)$  and  $\Delta e(k)$ . In the z-axis, the operating range for  $\Delta u(k)$  is displayed.

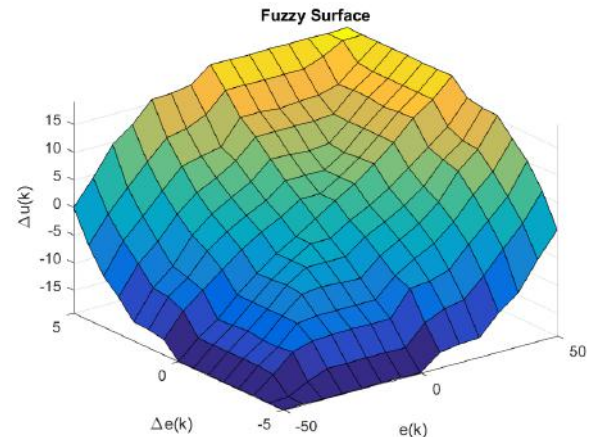


Fig.13: Fuzzy Surface.

#### IV. CONCLUSION

The FLC development using the model rules base suggested by MacVicar-Whelan proved to be effective for the control of the system, besides being a practical approach, since it is not necessary to create the rule base. The refinement techniques proposed by Procyk and Mandani were also effective in refining the submarine velocity response due to ease of understanding and application, making the implementation of the FLC similar to the application of a classical PID system. The FLC satisfactorily controlled the non-linear model of the submarine and proved to be robust in the presence of noise.

The main difficulties and limitations in the development of a FLC methodology are related to the kind of the analyzed system. Although the use of a model rule base facilitates the process, each system presents different characteristics related to its behavior. Therefore, it is not possible to develop a generalist method that will work with all types of systems. But in this work, with the submarine example, a notion is presented in how the changes in the membership functions and operating ranges of the controller variables can influence the system responses, according to the refinement rules. In this way, the development of this work can be used as a guide in the construction of FLCs for other types of applications.

As future work, it is intended to apply this methodology to build a PI-like Fuzzy controller to control a DC motor in simulation and in the real system, in addition to comparing the control with a classical control technique, such as the Proportional Integral (PI) control.

### REFERENCES

- [1] B.A. de Felippes, J. G. Aguiar and A. C. G. C. Diniz, "Sistema da qualidade em laboratórios universitários: incentivo ao ensino", pesquisa e extensão. Revista de Ensino de Engenharia, v. 30, n. 2, pp. 14-23, 2012.
- [2] T. Sardarmehni, N. Safarizade, M. B. Menhaj and H. Rahmani, "Robust neural predictive control of normalized air-to-fuel ratio in internal combustion engines," 2013 13th Iranian Conference on Fuzzy Systems (IFSC), Qazvin, 2013, pp. 1-4.
- [3] C. E. B. Romanini, "Development of a fuzzy control system for chicken egg setter," M.S. thesis, Faculty of Agricultural Engineering, State University of Campinas, Campinas, 2009.
- [4] H. Ren and Y. Zhang, "Nonlinear control based on an improved neural predictive control scheme," Proceedings of 2013 3rd International Conference on Computer Science and Network Technology, Dalian, pp. 1222-1225, 2013.
- [5] C. Vieira, "Forecasting Financial Markets with Artificial Neural Networks," M.S. thesis, School of Economics & Management, Higher Institute of Economics and Management, Lisboa, 2013.
- [6] F. Cheong, R. Lai, "On simplifying the automatic design of a fuzzy logic controller," Annual Meeting of the North American Fuzzy Information Processing Society Proceedings, 2002, pp. 481-487.
- [7] R. R. Yager and D. P. Filev, "Essentials of Fuzzy Modeling and Control," New York, USA: Wiley-Interscience, 1994.
- [8] D. M. Pathan, M. A. Unar and M. u. D. Memon, "Fuzzy Logic Controller for Submarine," 2005 Pakistan Section Multitopic Conference, Karachi, 2005, pp. 1-6.
- [9] M. Ekici, H. Kahveci and A. S. Akpınar, "A LabVIEW based submarine depth control simulator with PID and fuzzy controller," 2013 IEEE INISTA, Albena, 2013, pp. 1-6.
- [10] P. J. MacVicar-Whelan, "Fuzzy sets for man-machine interaction," In International Journal of Man-Machine Studies, v. 8, Issue 6, 1976, pp. 687-697.
- [11] K. L. Tang and R. J. Mulholland, "Comparing Fuzzy Logic with Classical Controller Designs," IEEE Transactions on Systems, Man and Cybernetics, v.17, 1987, pp. 1085-1087.
- [12] T.J. Procyk, E.H. Mamdani, "A linguistic self-organizing process controller," Automatica, vol. 15, Issue 1, 1979, pp. 15-30.
- [13] J. T. Procyk, "A self-organizing controller for dynamic processes", Doctoral dissertation, University of London, 1977.