Development and implementation of a deterministic model in Matlab for control of an elevator and Fuzzy Logic Using Block Diagram

First Author: I. Aldair Morales C. Second Author: F. Alfredo Forero. G
Students Systems Engineering University Amazonia

Florencia - Caqueta
I i.morales@udla.edu.co | forerogomez4@gmail.com

SUMMARY

The following work is based on the concepts of fuzzy logic and operation of block diagrams, where a control problem occurs, the task is to simulate through the development and implementation of a mathematical model and the use of fuzzy logic by a fuzzy controller controlling an elevator, wherein variables include deterministic modeling.

Said control problem by applying fuzzy logic allows us to introduce new concepts concerning the control schemes applied in the field of engineering, management tools plus mathematical programming and modeling of this as it is Matlab.

Keywords: Fuzzy logic controllers, mathematical model, Matlab, programming, block diagrams, Laplace transform.

I. INTRODUCTION

This article provides fundamental basis to develop and implement using the integrated development environment Matlab a deterministic model for controlling an elevator and it based on fuzzy logic and use of block diagrams, where we have a enrichment of knowledge regarding control mechanisms.

The development will be doing is a basic problem of deterministic models, also based on fuzzy logic to create a fuzzy controller and implement it on a block diagram well structured with the control system of an elevator.

The problem posed in this article is based on the subject of Artificial Intelligence, guided by the teacher Milher Fabian Tovar Rubiano program Systems Engineering, more precisely in the course of Artificial Intelligence.

The subject as such, presents the advancement of solid foundations for what is introduced to Computational Intelligence, where students put into practice the knowledge acquired based on neural networks, evolutionary, genetic algorithms and case specific basis fundamentals of fuzzy logic.

In this article, several sections where we will see a focus on what we want to accomplish by using fuzzy logic, then the identification of mathematical factors (development of the mathematical model) are presented and refers to all programming that performed directly in Matlab.

Furthermore we rely primarily on mathematical theorems implemented in the world of electronics, specifically in the field of electrical circuits to have a concrete structure of the control system using the Laplace transform.

II. FIRST APPROACH TO DEVELOPMENT AND IMPLEMENTATION MATHEMATICAL MODEL (LAYOUT)

What we want to do in this work, as previously we had mentioned, is to develop the control system of an elevator by using fuzzy logic, moreover able to have a control signal to stabilize and we can determine that the system function properly, and to do so by implementing a mathematical model of deterministic type, moreover this done in Matlab using block diagrams, more broadly using Simulink.

The first approach we can do in this case is taken into account as is the operation of the lift.

.

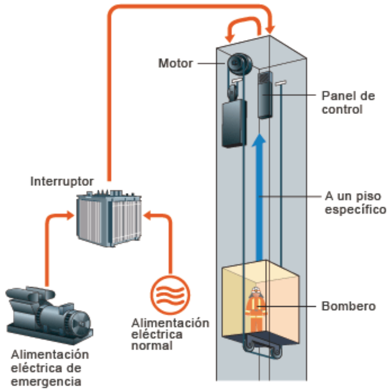

Figure 1: Structure of a Elevator[1]

In this case we will focus on applying our knowledge to the development of elevator control system, where we will consider the portion of power, motor, control panel.

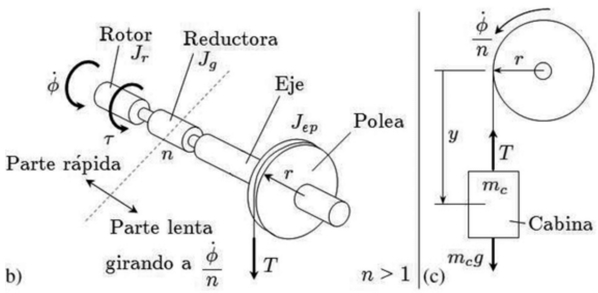

Figure 2 (b) rotating parts Elevator system(c) Forces acting on the elevator[2].

Given the above, our system will have an input, process and output, and a feedback via a sensor.

- ✓ Input: Voltage.
- ✓ Output: Angular Velocity.
- ✓ Process: Angular Velocity / voltage.
- ✓ Sensor: The sensor is an additional control that we can implement it as optional.

Diagram 1: Control system with its input and output process[3].

So what you want to get with this is a transfer function, which is defined as:

[1] Figure 1: Structure of a Elevator, image taken from http://www.mitsubishielectric.com/elevator/es/products/basic/elevators/nexiez_mrl/index_safety.html taking into account security elevators and escalators.
[2] I Figure 2: (b) rotating parts Elevator system c) Forces acting on the elevator, taken from https://www.researchgate.net/figure/Partes-del-modelo-dinamico-del-ascensor-a-Circuito-equivalente-del-motor-DC-b_fig8_324114705
[3] Process diagram of the control system with its input and output process.

$$Función\ de\ Transferencia = \frac{Salida\ \dot{\theta}(t)}{Entrada\ e} \qquad (1)^4$$

III. APPLYING MATHEMATICAL MODEL DEVELOPMENT OF LAPLACE TRANSFORMATION

In order to implement the mathematical model and then make a Laplace transform, must consolidate several previous concepts such as the note about the rotor of an engine and an electric motor operation.

Now, develop the mathematical model implies taking into account the block diagram, then we refer to the circuit diagram provided by the teacher Milher Fabian and work with the development of the mathematical model.

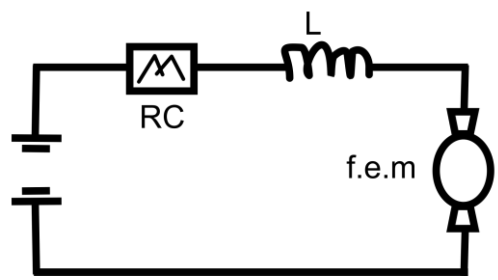

Figure 3: Circuit control system Elevator[5].

Based on the above circuit, we can raise the deterministic mathematical model as follows, taking into account the relevant mathematical procedures.

$$\dot{\theta} = velocidad\ angular\ (\frac{\partial\theta}{\partial t})$$

$$-e + v_r + v_l + v_{f.e.m} = 0 \qquad (2)$$

Regarding the above equations are:

$$v_r = iR$$

$$v_i = \frac{\partial i}{\partial t}L$$

$$v_{f.e.m} = k_c\dot{\theta}(t)$$

Solving (2)

$$-e(t) + i(t)R + \frac{\partial i}{\partial t}L + k_c\frac{\partial\theta}{\partial t} = 0 \qquad (3)$$

With respect to the above equation, we can think about how to implement or use a transfer function that had previously described.

$$F.T = \frac{(\dot{\theta}(t))}{(e)}$$

Thus we can implement a function of torque generated from the correspondence between the gear and the pulley raising and lowering the elevator.

[4] Transfer function.
[5] Circuit elevator control system.

$\sum T = J\ddot{\theta}$ Developing the equation would

$$\sum T = J\frac{\partial^2 \theta(t)}{\partial t^2}$$

$$Tm = J\ddot{\theta}(t)$$

$$Tm = K_{c1} i(t)$$

Then you can say that the role of Torque () we can solve and get:T

$$K_{c1} i(t) = J\frac{\partial^2 \theta}{\partial (t)^2} \qquad (4)$$

And this equation (4) we apply the Laplace Transform to obtain a transfer function with complex values.

$$L\{f(t)\} = F(s) \qquad (5)^6$$

$$L^{-1}\{F(s)\} = f(t) \qquad (6)^7$$

To solve and to find the values of the transfer function, the initial conditions = 0 will have.

solving (5)

$$L\left\{\frac{Kdf(t)}{dt}\right\} = Ks + F(s) + F(s_0)$$

$$L\left\{\frac{Kd^2 f(t)}{dt^2}\right\} = Ks^2 + (s) + SF(s_0) - F(s_0)$$

So

$$L\left\{-e(t) + i(t)R + \frac{\partial i(t)}{\partial (t)} L + K_c \frac{\partial (\theta)}{\partial (t)}\right\} = L\{0\}$$

$$L\{-e(t)\} + L\{i(t)R\} + L\left\{\frac{\partial i(t)}{\partial (t)} L\right\} + L\left\{K_c \frac{\partial (\theta)}{\partial (t)}\right\} = L\{0\}$$

$$-e(s) + i(s)R + Lsi(s) + Kcs\theta(s) = 0 \qquad (7)$$

Then we obtain a function of the form:

$$-e(s) + i(s)[R + Ls] + Kcs\theta(s) = 0 \qquad (8)$$

$$L\{Kc1 * i(t)\} = L\left\{J\frac{\partial^2 \theta(t)}{\partial t^2}\right\}$$

$$Kc1 * i(s) = Js^2\theta(s) \qquad (9)$$

Now clearing and solving (9)

$$i(s) = \frac{Js^2\theta(s)}{Kc1}$$

[6] Function Laplace Transform. Function taken from https://es.wikipedia.org/wiki/Transformada_de_Laplace Free Encyclopedia.

[7] inverse function of the Laplace Transform

Given the equation (8)

$$-e(s) + i(s)[R + Ls] + Kcs\theta(s) = 0$$

$$-e(s) + s\theta(s)\left[\frac{J}{Kc1} s[R + Ls] + Kc\right] = 0$$

$$e(s) = w(s)\left[\frac{J}{Kc1} s[R + Ls] + Kc\right] = 0$$

Where

- ✓ $e(s)$ It is the entrance
- ✓ $w(s)\, es\ la\ salida$

So

$$1 = \frac{w(s)}{e(s)}\left[\frac{J}{Kc1} s[R + Ls] + Kc\right]$$

With the above equation, we can get an idea of the transfer function, which is as follows:

$$\frac{w(s)}{e(s)} = \frac{1}{\frac{J}{Kc1} s[R + Ls] + Kc} \qquad (10)^8$$

IV. MATLAB BLOCK DIAGRAM

To calculate our transfer function we have three default values, which are:

- ✓ $J = 1$
- ✓ $Kc1 = 0.8$
- ✓ $Kc = 1.2$

But the latter differs each group working in our table corresponds to the values of $Kcf.e.m$

- ✓ $f.e.m\ (fuerza\ electromotriz) = 1.5$

We also have values and:RL

- ✓ $R = 2$
- ✓ $L = 0.2$

Doing the math we get:

$$\frac{w(s)}{e(s)} = \frac{1}{\frac{(1)(0.2)}{0.8} s^2 + \frac{(1)(2)}{0.8} s + 1.5}$$

$$\frac{w(s)}{e(s)} = \frac{1}{0.25s^2 + 2.5s + 1.5}$$

Our transfer function obtained what we do to implement it in Matlab, for this case will work under the Matlab R2015a version.

[8] Transfer function found from the transformation was by Laplace.

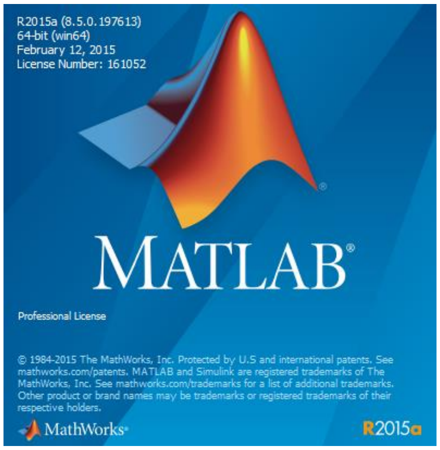

Figure 4: Initialization screen in Windows 8.1 Matlab[9]

By Simulink to design our block diagram, what we do is to select the work location (current folder) and then we click on the tool Simulink or simply line functions we write the word Matlab Simulink. $fx \gg$

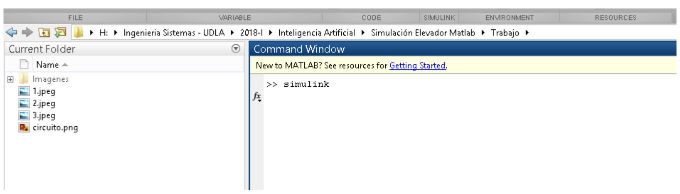

Figure 5: Current Folder (current folder) where we will keep all our files created in Matlab.

The block diagram implemented in the Integrated Development Environment in its mathematical Matlab R2015a version is as follows:

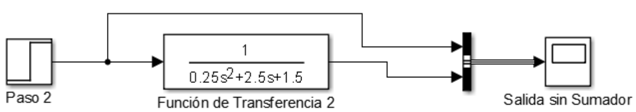

Figure 6: Transfer Function[10]

Wherein the transfer function calculated using the values obtained above, with the graph as follows:

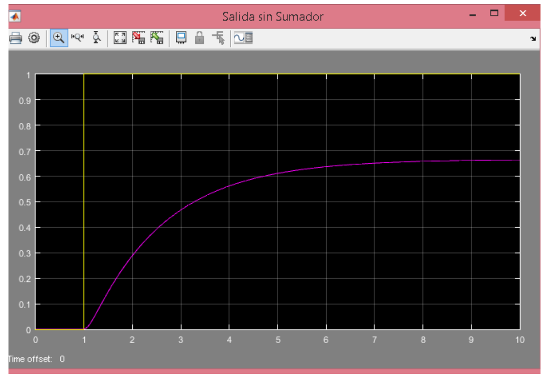

Figure 7: Graph output control system held in Simulink Block diagram.

V. Development and Implementation of Fuzzy Controller in Matlab

To implement the fuzzy controller, what we should do in Matlab, it is go to the main window and look inside your Apps designer fuzzy logic (Fuzzy Logic Designer) and clicking on the we will set our fuzzy controller, furthermore we must bear in mind that going to be deterministic Sugeno.

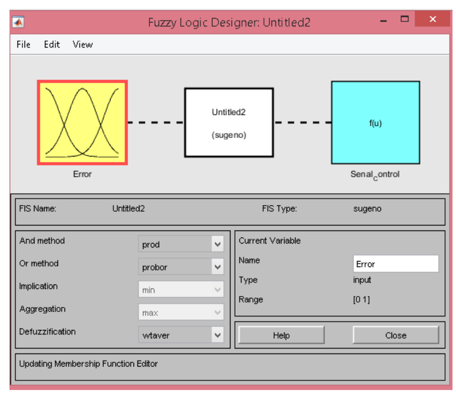

Figure 7: First approach to the design of our fuzzy controller.

One can see that we have modified corresponding to the input and the output, ie, assign name Error (input) and control signal (output).

Then to set up my fuzzy universe and consider the inference rules for the development of my fuzzy controller to be implemented in the block diagram of the elevator, we have to double click on the error and set the controller there.

[9] Initialization screen Matlab R2015a - Pantallazo taken from own computer.

[10] Figure 6: Transfer function and is known as the first function of the system fed back, ie, the system is unchanged.

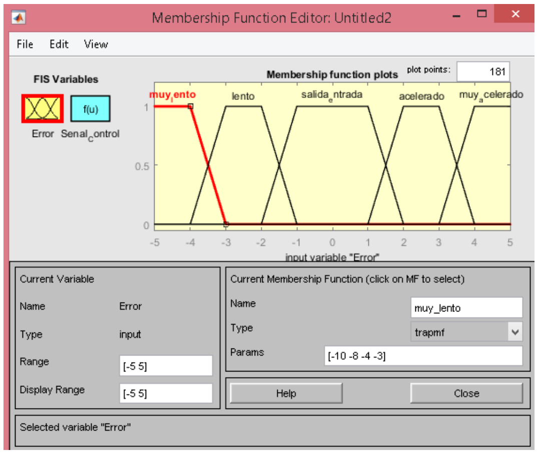

Figure 8: Setting the features that you will have our fuzzy controller

Given what is known as a horizontal data validation, we have taken ranges from -4, -3, -2, -1, 0, 1, 2, 3, 4, so that we can implement our fuzzy sets in the controller.

Furthermore we must consider that we are working on a linear function, therefore we must manage our system values to control when you are in any of the five states we have.

Configuration values that we will take into account, we do so by the rules of inference we can handle our fuzzy controller.

In order to assign control values, what should we do in the control signal that we as output, moreover to take into account that these values are belonging to what we stated in the same fuzzy set, ie, when in area very slow, this should accelerate more, but this we will see in greater detail in the rules of inference.

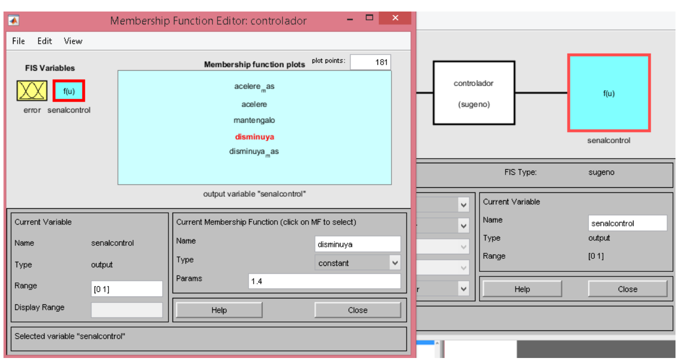

Figure 9: Fuzzy sets values with respect to the input.

Once we have these values defined in order of 1 to 5, then what we do is go to the inference rules to apply the respective conditions.

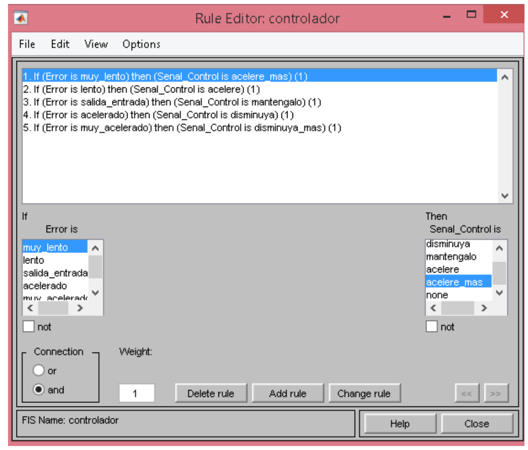

Figure 10: Inference rules

Once this is done, you can simulate using View> Rules, the rules of inference we just declare.

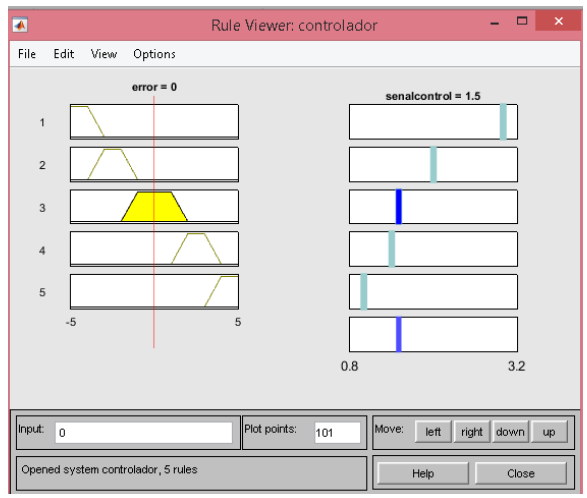

Figure 11: Simulation of inference rules

What we have to do next is to mount our fuzzy controller constructed by App Fuzzy Logic Designer in our block diagram, and for this we added thereto a fuzzy controller from the library of Matlab in Simulink, looking into options Fuzzy Logic Toolbox, and select the name "Fuzzy Logic Controller".

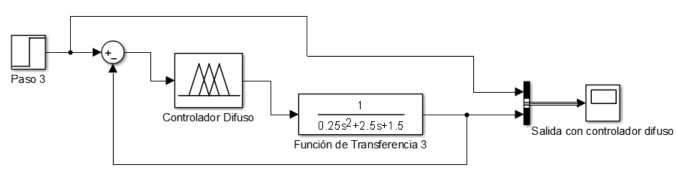

Figure 11: Block diagram of the fuzzy controller.

In order to run our control system, we must export to Workspace our controller with .fis extension, ie, we need to incorporate into our work environment, for this we click on File> Export> to Workspace

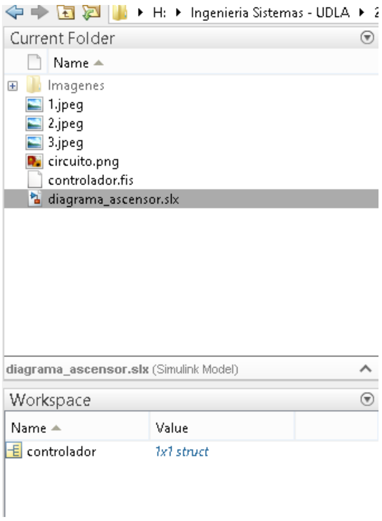

Figure 12: Export to Workspace our fuzzy controller with .fis extension.

Once you run the simulation of our control system, it is observed that the demarcation line error is stabilized, which is why so a proper functioning of the system is evident.

In addition, the result of the function is generally positive because we have almost no margin for error as to the stabilization of the system.

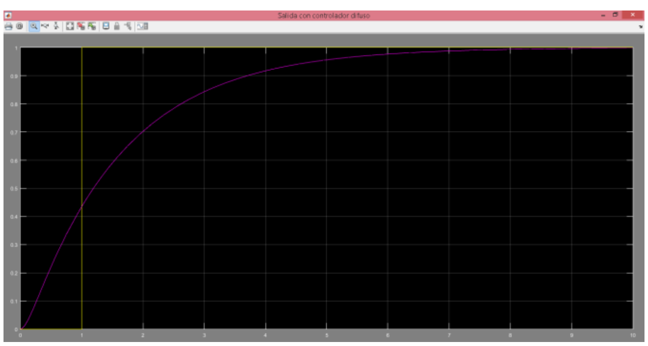

Figure 13: Graph end of our control system implementing the fuzzy controller.

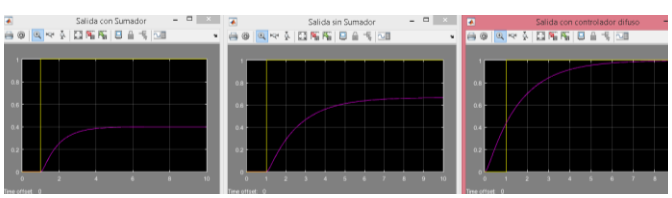

Figure 14: Comparison of the three diagrams implemented blocks, where the latter represents the implementation of the fuzzy controller.

VI. CONCLUSIONS

According to the statement of the problem and its respective solution by a control system of an elevator which was designed to stabilize the control signal I was shown that by using block diagram and implementation of fuzzy logic in terms of terms computer, could solve the problem of controlling the system, in this case the elevator.

Likewise, it should clarify that all system very robust it has errors, which can be done before any case is optimized by using tools, for our problem it had already mentioned the use of fuzzy logic applying in the block diagrams.