COMPARATIVE ANALYSIS OF CONTROL PROCESS WITH FUZZY LOGIC AND REINFORCEMENT LEARNING

Martínez Montoto, Rodrigo
tinrmm02@estudiantes.unileon.es
University of León

Abstract— *This paper is about a comparison between two different types of control. The process that we are going to control is the inverted pendulum and the two types of controls are fuzzy logic and reinforcement learning. How to build the controls is argued, as well as the results obtained.*

Keywords— inverted pendulum, Simulink, Matlab, control process, fuzzy logic, reinforcement learning.

I. INTRODUCTION

The purpose of the present research is to do a comparison between two different methods of control. The process that is going to be controlled is the inverted pendulum and the methods of control are fuzzy logic and reinforcement learning.

The work was made under simulation, in the Simulink and Matlab environments.

To achieve our goals with the fuzzy logic was necessary to implement the mathematical model of the inverted pendulum in Simulink, and after that the fuzzy control was implemented, achieving excellent results.

To achieve our goals with the reinforcement learning was necessary to implement the dynamics of the inverted pendulum in a Matlab algorithm, to make discrete its states (because it is too hard to work with continuous processes), and to implement an algorithm for the control of the process.

After that a comparison is made.

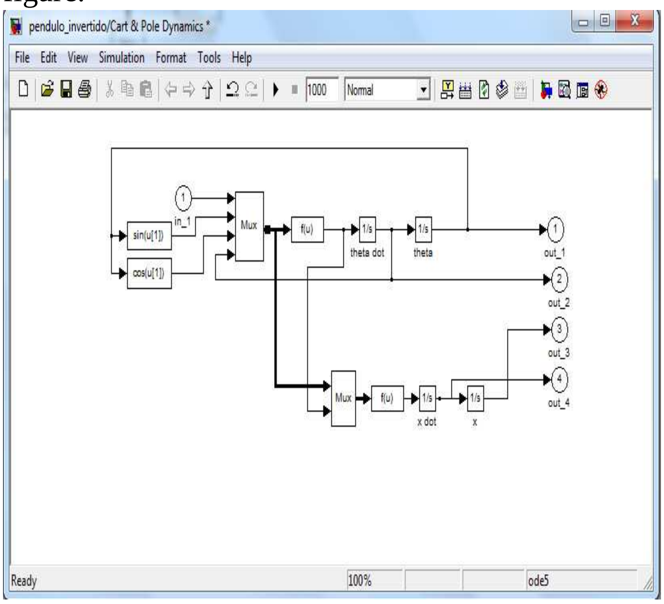
II. FUZZY CONTROL FOR THE INVERTED PENDULUM

This control is grounded in the fuzzy logic that is a tool to model the expert knowledge of the human being through fuzzy sets. These sets are characterized by the grade of membership of the universe elements to the sets. This grade of membership can vary between 0 and 1. On the other hand, the membership of the elements of the crisp sets can only be or 0 or 1, it means either the element is part of the set or is not part of the set.

An advantage of the fuzzy logic designing controls is that the common language can be used. It permits very robust controls and makes intuitive the designing of the control. There is not an area of human activity in what the fuzzy logic cannot be used.

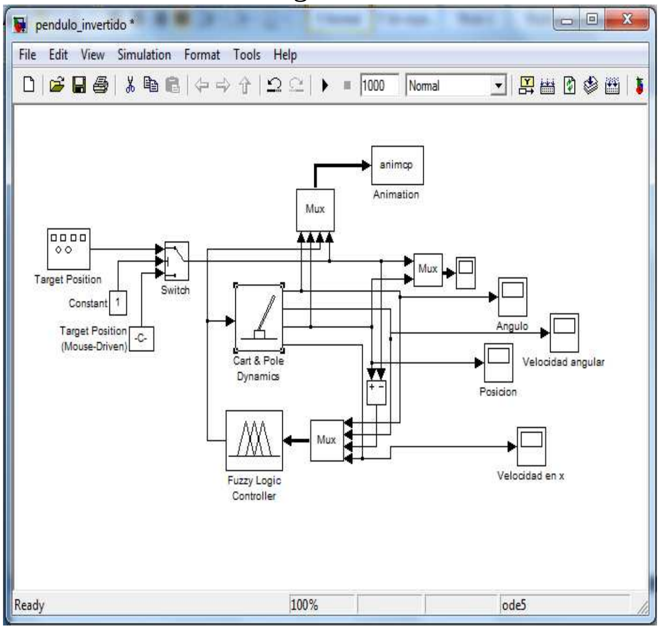
Talking about our process, first of all we need the mathematical model of the inverted pendulum. In this paper this model is considered as known because is typical example of the modeled.

Once we have the model of the inverted pendulum, the next step is to implement it in the Simulink environment as it is showed in the next figure.



The figure shows the subsystem of the inverted pendulum. It is seen one input and four outputs. The input is the force with the cart and pole system is pushed or pulled. The outputs are the angle respect the vertical axis, the angular velocity of the fall of the inverted pendulum, the position in the horizontal axis of the cart and the linear velocity of the cart.

This subsystem is part of a bigger system that it is showed in the next figure.



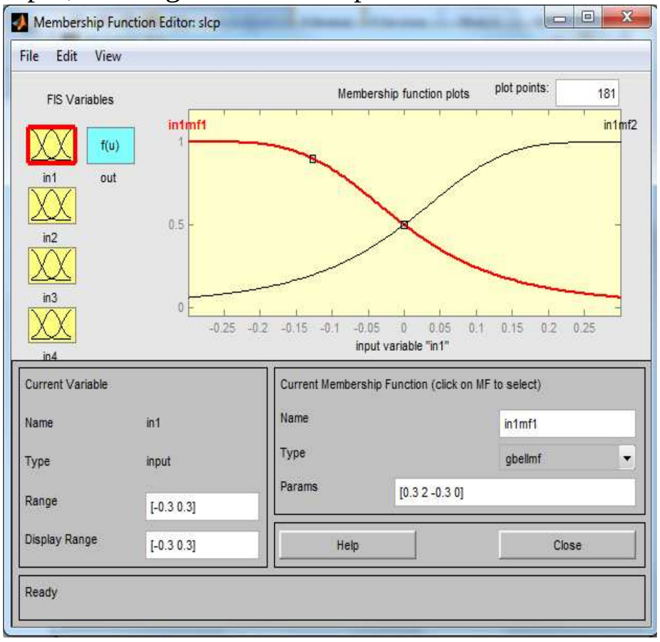
This system represents the control process with fuzzy logic of the inverted pendulum.

The block Cart and Pole Dynamics is the model of the inverted pendulum that it was seen before. The block Fuzzy Logic Controller is fuzzy control. The other blocks are for plot the outputs, make a simulation of the process and make a goal position for the inverted pendulum.

The fuzzy inference system chosen is the Sugeno system. The difference between Sugeno and Mamdani is the output of the controller. In the Sugeno system the output is a constant or at least linear, so it is not necessary to convert the out value in an understandable value.

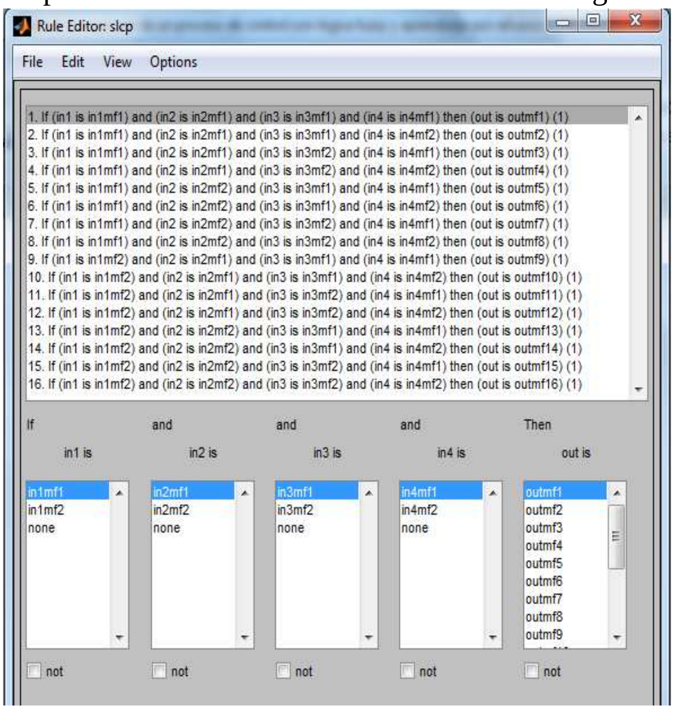
Talking about the fuzzy control itself, I have to mention that it is necessary to make the inputs fuzzy. The function chosen for this purpose is the bell function. The next figure shows the input 1, the angle input, in which we can see how it is represented as a fuzzy input.

All the inputs are represented the same; the meaning is the only thing that is different for the inputs. We have two membership functions for each input, one negative and one positive.



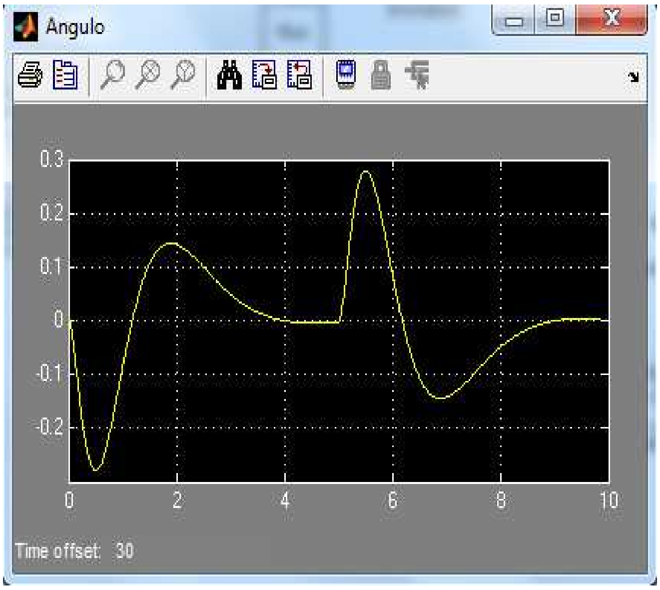
The bigger the membership value, the more negative or positive is the input.

The rules of our controller are if-then. So if we have an input1 value and an input2 value and an input3 value and an input4 value, then we have an output value. The rules are shown in the next figure.



The output in our controller is linear; it depends in a list of parameters that are going to be chosen in regard of the inputs.

With the fuzzy controller explained, it is the turn of the response of the system. The next figure shows the response of the angle. The process responds very quick and very good regarding our goals. So it can be said that the fuzzy control is a success.



III. REINFORCEMENT LEARNING FOR THE INVERTED PENDULUM

The purpose of this part of the research is to implement control for the inverted pendulum by an algorithm of reinforcement learning.

The reinforcement learning is part of the automatic learning, of which purpose is to develop techniques for the computers with they can learn to do a job automatically.

The most important part of the reinforcement learning is the Markov decision processes (MDP). MDP is formed by S, A, Psa and R.

S is the set of states in that the process can be.

A is the set of actions that the process can take.

Psa is the probability of going to a state taking an action.

R is the rewards function.

The dynamic of a MDP is as follow: It begins in the state s_0, and then we take the action a_0. According with the probability of achieving a state taking an action we are going to achieve the state s_1, and after this we have to repeat the process.

To make this valuable, we have to give rewards when the process is in some states. The reward can be positive or negative, depending if we want to go to this state or not. If we want to go to that stare the reward must be positive, and if we want to avoid that the reward must be negative.

The reward of the states is discounted with a factor ϒ. The immediate state is not discounted, and the following states are discounted by Υ^t. Because of this we are interested in achieving the good states as soon as possible.

The decision of taking an action or another is made by a policy π. It exists a value function V that says how good the policy is and what is the best way to achieve our goal.

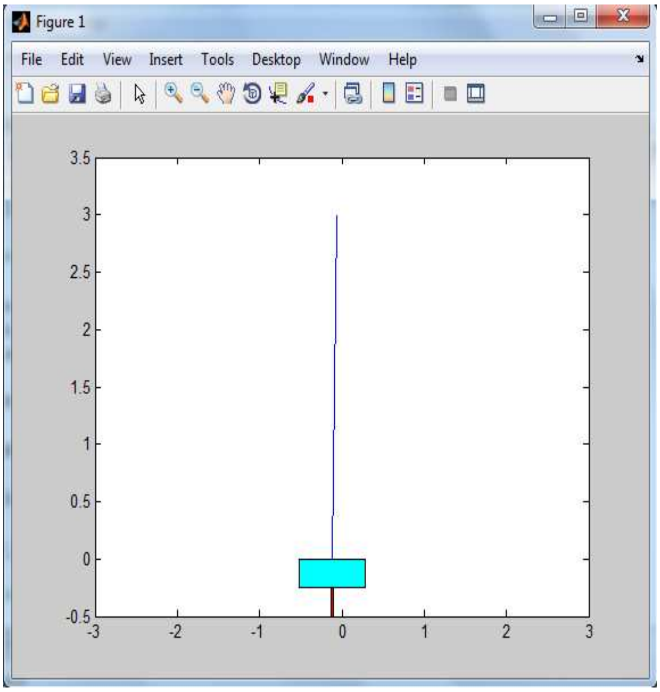
The Bellman's equations are satisfied. It means that it doesn't matter in which state we start; the way to achieve our goal is the same.

Talking about our process, we are going to need some algorithms to make it works.

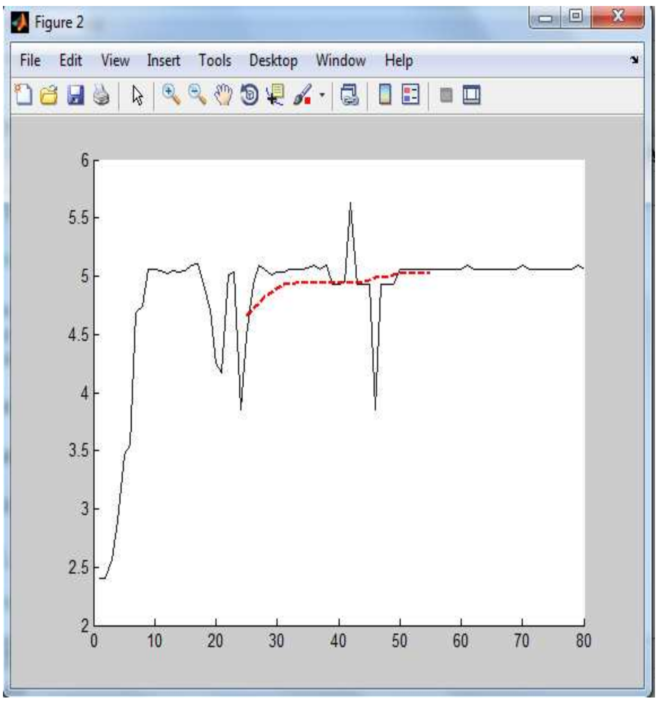
The main algorithm is called "control.m", this performs the control, but it needs help of other algorithms. The algorithm "cart_pole.m" simulates the dynamics of the inverted pendulum, and gives us the variables of the process. The algorithm "get_state.m" gives us the discrete state of the inverted pendulum regarding of the variables of the process, the angle, the position, the angular velocity and the linear velocity. The algorithm "show_cart.m" shows us a representation of the process. The next figure is the representation.

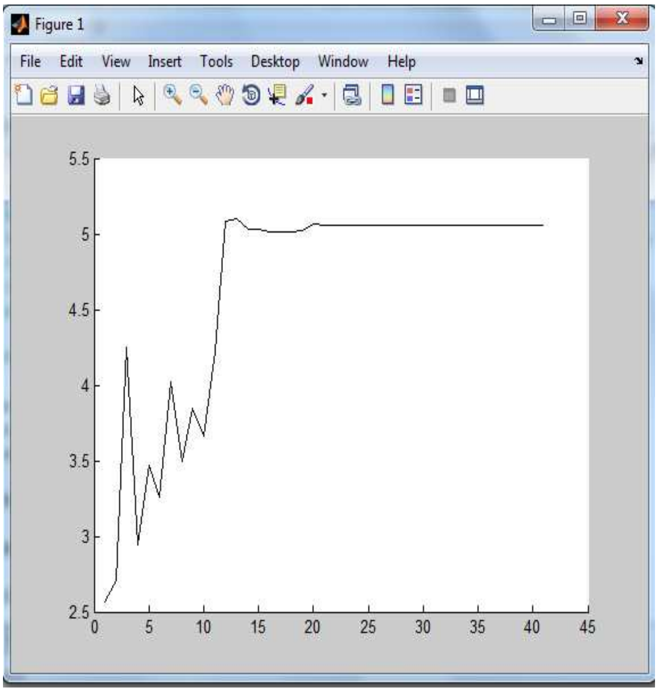
Talking about the control itself, in our case, the only reward is -1 in the failure state. This state is when the pole is down more than 12º, or when the cart is more left or more right than -2.4 and 2.4, so our goal is to keep the cart and pole in the limits. The actions than can take the process are 1 or 2, and they mean to move the cart and pole right or left.

In our case the probabilities matrix and reward matrix are unknown, so we have to start a policy randomly and watch what happens. After a failure, the process updates these matrixes and begins again. After doing this once and once, the algorithm will converge at some point which is when the value matrix do not learn anything.



After this, the control algorithm calls the algorithm "plot_learning_curve.m" that plot the learning curve of the process. Because some matrixes are initially random, the results vary a little. But it does not mean that the results are bad, they are actually very good. The next two figures show two different learning curves.





The first figure shows some irregularities, but as it was explained before, the reason is the matrixes that are random in the beginning.

IV. COMPARATIVE ANALYSIS AND CONCLUSIONS

Once the controllers are analyzed, it is time to analyze them, making a comparison.

Talking about the response, the fuzzy controller is better, because it is quicker and more precise. Furthermore, it allows you to make a more concrete control of the process. It is possible because the fuzzy logic implements the expert knowledge of the human being into the control in rules form.

The performance of the fuzzy control is better than the performance of the reinforcement learning control because the value function is initially random, among other things. It is important take in mind that the need of having the states discrete in the reinforcement learning adds complexity to the process, making more difficult to build a more concrete controller.

Then the main difference between the two controllers is that the fuzzy control is more concrete and quick because of the possibility of implementing the expert knowledge, the reinforcement learning does not need to have any knowledge about the process, it is based in the probability and mathematics, trying the control once and once in order to improve the result until the process stops to learn, and it achieves good results.

REFERENCES

[1] CASTAÑOS LUNA, F. *Levantamiento y estabilización del péndulo invertido.* 2003
[2] AKOLE, M. y TYAGI, B. *Design of fuzzy logic controller for nonlinear model of inverted pendulum-cart system.* XXXII National systems conference, NSC. 2008
[3] CORNELL UNIVERSITY. *CS-4758/6758 Robot learning. Reinforcement learning.*
[4] BERGASA PASCUAL, L.M. *Diseño y simulación de un controlador borroso de un péndulo invertido en entorno MATLAB/SIMULINK.*
[5] YI, J. y YUBAZAKI, N. *Stabilization fuzzy control of inverted pendulum systems.* Artificial intelligence in engineering. 2000
[6] STANFORD UNIVERSITY. *CS229 Reinforcement learning and control.* Lecture notes. 2011
[7] STANFORD UNIVERSITY. *CS229 Unsupervised learning and Reinforcement learning.* Problem set. 2011
[8] MATHWORKS. *Matlab user manual. 2007*

Author

Industrial engineering degree, specialty on industrial electronics and automatic by the University of León.