Computational Intelligence
Fuzzy Systems

Ioannis Nektarios Sotiropoulos, Angel Poc Lopez

In this lab exercise we want to develop Mamdani FIS which will act as a controller of a compound pendulum. We want to stabilize it at a specific angle applying a torque.

1 Description of membership functions & rules

Before defining the rules we need to take as a given that we have the error of the angle and derivative of the error as input that allows us to measure that control through the output variable which is the thrust.

$error = \theta_{ref} - \theta_{output}$

$\frac{d(error)}{dt} = error_{new} - error_{old}$

1.1 Membership functions

The membership functions which were used for input and output are triangle functions.

Input

- *Error*: We used 3 levels for the error *Negative - Zero - Positive.* We get the Negative result when we are at an angle bigger than the reference one, Zero when we reach at the reference point and positive when the angle is smaller.

- *Derivative of error*: We used 3 levels here as well *Decreasing - Stationary - Increasing.* The Decreasing has been set to the negative values of the derivative, the Stationary for the ones that the error don't change (around the zero value) and Increasing for the positive values of the metric.

Output

- *Thrust*: We used 3 levels here for the thrust *Negative - Neutral - Positive.* The thrust can be interpreted as pushing and pulling or as applying more and less force (torque) to move the pendulum. The positive values represent more force and the negative values less force, while the neutral has a range between -12.5 and 12.5.

All membership functions are spread uniformly around zero with the zero value to have bigger grade of membership for the zero or neutral membership function.

1.2 Rules.

It's important to analyze the possible input values for all cases of θ in order to have a better overview of the pendulum problem and infer the possible output.

When $\theta > \theta_{ref} => error < 0$ and the pendulum:

- is getting further than the desired angle then the $Derivative of error < 0$ and thus, it's decreasing. At this case the we should apply $thrust < 0$ in order to move it back.

- is getting towards to the desired angle then the $Derivative of error > 0$ and thus, it's increasing. At this case we should apply $thrust < 0$ or a neutral value(because of its tendency to move towards θ_{ref}) in order to move it back faster or slower. However, when it reaches $\theta = \theta_{ref} => error = 0$ then we want to apply the opposite $thrust > 0$ (or a neutral value) in order to stop its tendency to move to the area where $\theta < \theta_{ref}$.

When $\theta < \theta_{ref} => error > 0$ and the pendulum:

- is getting further than the desired angle then the $Derivative of error > 0$ and thus, it's increasing. At this case we should apply $thrust > 0$ in order to move it back.

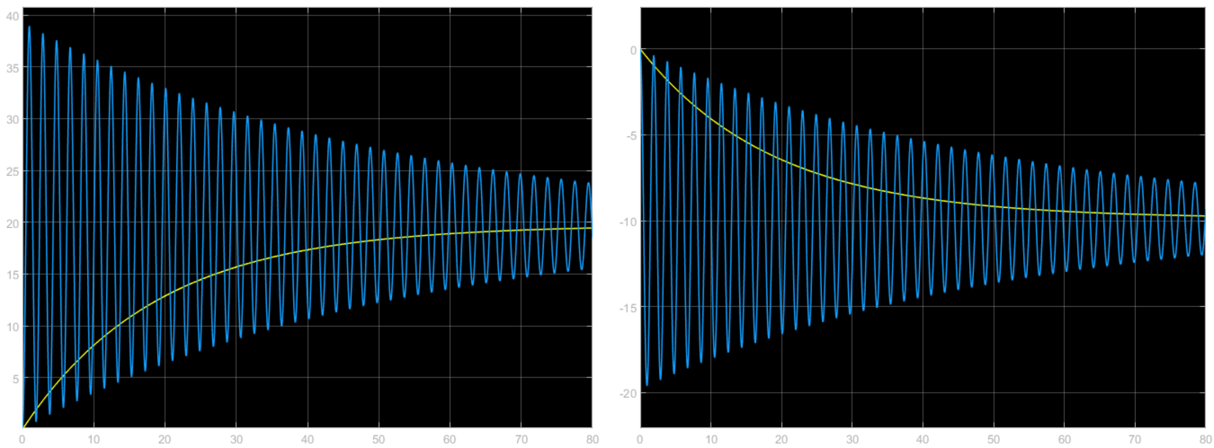
- is getting towards to the desired angle then the $Derivative of error < 0$ and thus, it's decreasing.

At this case we should apply $thrust > 0$ or a neutral value in order to move it back faster or slower. However, when it reaches $\theta = \theta_{ref} => error = 0$ then we want to apply the opposite $thrust < 0$ (or a neutral value) in order to stop its tendency to move to the area where $\theta > \theta_{ref}$.

For the Stationary Derivative the pendulum may be at the same position, so, we assume the same thrust level (or a more neutral value) as the other levels in the same area of θ. Given that, we need to clarify that in the case of $error = 0$ we assume $thrust =< neutralvalue >$.

2 Plots for $\theta_{ref} = 20$ & $\theta_{ref} = -10$

As we mentioned in the rules section, the rules were chosen for all the values of θ. From the *Figure1a* and *Figure1b* we can see that with the fuzzy controller we have a smooth convergence and we don't surpass the desired angles since we are near the neutral area and we try to keep the pendulum at this exact position applying small positive and negative thrust continuously. The thrust values come from the defuzzification process which uses the centroids in this case. Because of that after convergence we don't use just zero thrust in the neutral area, but a thrust which has a membership degree in the positive or negative area as well.



(a) $\theta = 20$

(b) $\theta = -10$

3 Behavior increasing the number of membership functions.

We think there are two approaches trying to increase the number of membership functions: having an odd or an even total number of them. In the case of the even number of membership functions, you would either end up having an asymmetric model (from the point of view of the distribution of the membership functions along the domain) or a symmetric model in which there would not be a membership function centered in the value 0 thrust, which would lead to an unstable model that would never stop at the desired angle. On the other hand, we deal with an odd number of membership functions. We decided to test it and simulate it for different function definitions.

By trying different kind of rules we came to the conclusion that with 5 and more membership functions in the output we can make rules which favor more abrupt changes, so, we can get faster convergences.

An example of such a system is one with equally distributed membership functions along the domain or a system similar to the the initial case with a membership function of the same width inserted between every pair of previously existing functions (the former converges faster).

A new set of rules has also been defined. The intuition here is to have a bigger thrust when there is a big amount of error and assign an intermediate negative or positive membership function, as we approach to 0 error. This system converges notably faster than the initial one. It reaches faster a point near the reference theta and once it is there, it approaches the goal taking smaller steps.