**Fuzzy control**

Overview:

The fuzzy logic control for the inverted pendulum is considered as the alternative to the classic PID control method. This method is based on Gray code, a binary number system in digital logic (Maxfield, 2011) i.e. to attain a finer resolution by deriving an intermediate logic state without having to obtain a higher computational bit processor.

Fuzzification:

In FLC(Fuzzy logic controller) the input values are converted into these intermediate logic state known as fuzzy value sets. There the numerical data get partitioned into set of linguistic rules. (Mcneil, 1994)

Defuzzification:

The signal coming out of the controller gets mapped as solid output crisp sets. The output can sometimes be left as fuzzy values in some application where there are no finite ranges a pure fuzzy system.

In PID control even after the system is well tuned, the dynamic nature of the parameters

makes it harder to attain the desired performance. This is where fuzzy controller acts as a redeemer to make the system less susceptible to disturbances. Two well known issue that FLC resolves are less overshoot and quicker settling time. (Abdelhameed et al., 2019)

The advantages of using fuzzy logic lies in the fact that the system can be viewed as of adaptive nature incorporating the knowledge of a human operator adjusting accordingly to the varying environment. The knowledge base of the controller is built upon by asserting behavioural rules, which boots the system response and have more control to any change in the system.

The further complex a system becomes the more bulky the computational tasks of the PID control which in turn could hinder its performance. Whereas utilising fuzzy logic the complexity can be decomposed into more manageable system. (Gouda, Danaher and Underwood, 2000)

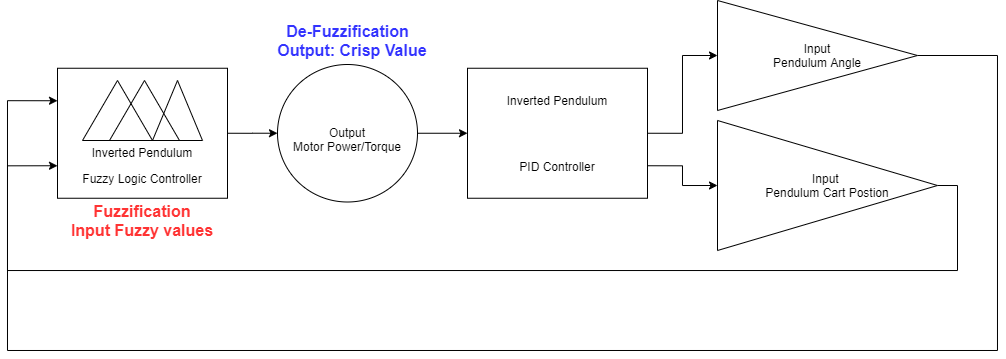


Fig 1: Block Diagram: 2 inputs and 1 Output based FLC(Fuzzy Logic control) System

Design brief

The implementation of the controller was carried out in MATLAB via graphical fuzzy logic toolbox editor.

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Type** | **Range** |
| Pendulum Angle\_θ | Input | -42 to +42 |
| Angular Velocity\_dps (Degree per second) | Input | -9 to +9 |
| Force\_N (to be applied to the motor) | Output | -28 to +28 |

This system as can be seen will be a Multiple Input Single Ouput (MISO).

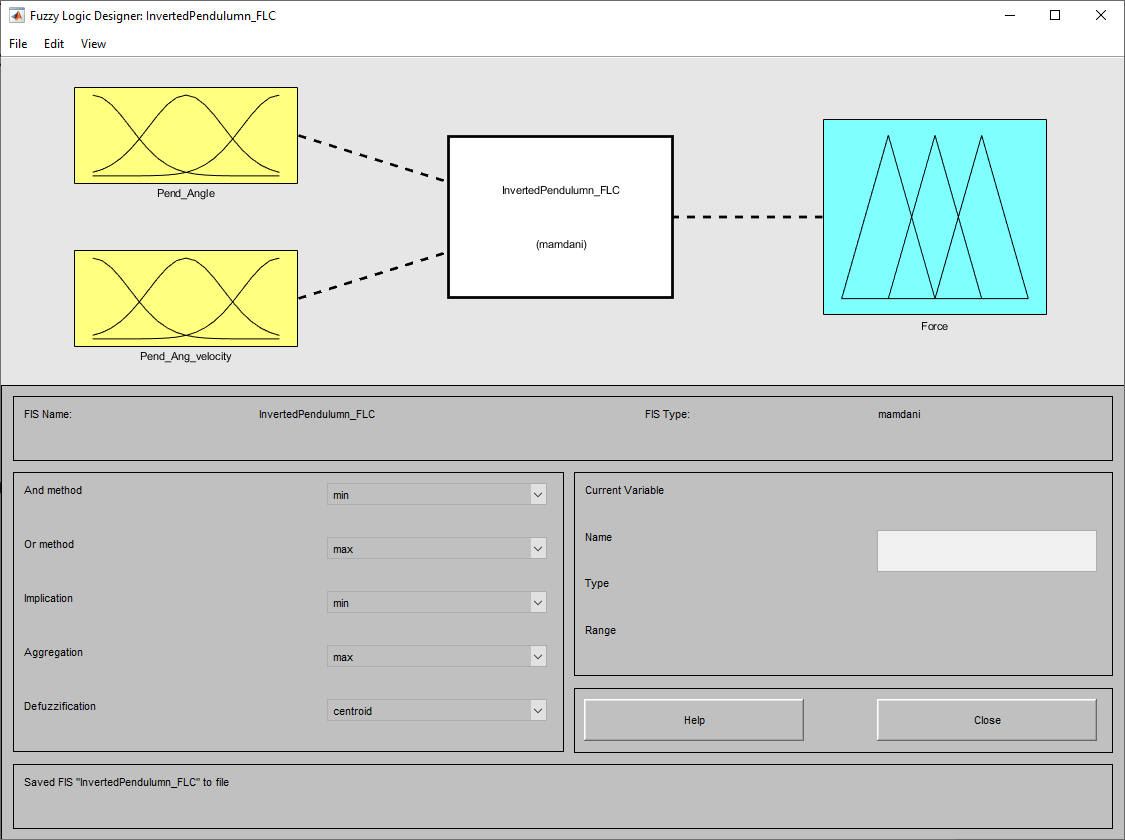


Fig 2: MATLAB Fuzzy logic toolbox editor

Membership functions:

Triangle – for all the parameters the triangle function is utilised which uses an Minimum and Maximum datum in X-axis, followed by a median datum point in Y-axis.

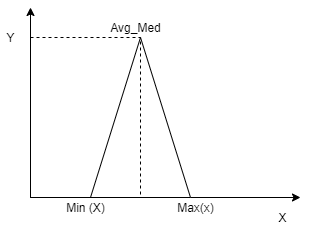


Fig 3: Triangle Function data points.

This function is commonly used in at least one part of a control system. For our application this function seemed suffice to extract the output accordingly.

The tuning of the system initially can be tested with fewer membership function then upon the performance review. If the system still pertains to be unstable additional functions can be added to the particular parameter to fine tune and gain more control.

Inference engine:

Grouped membership functions for each parameter with specific operating region. This determine which rule to used according to the mapped input. The relationship between input and output across the pre-theorised rules is derived here. These seven membership functions are defined in the FIS (Fuzzy Inference System) editor: Negative Big (NB), Negative Medium (NM), Negative Small,

Zero error (Z), Positive small (PS), Positive Medium (PM) and Positive Big (PB), can be seen below in figure 3.5 and 3.6.

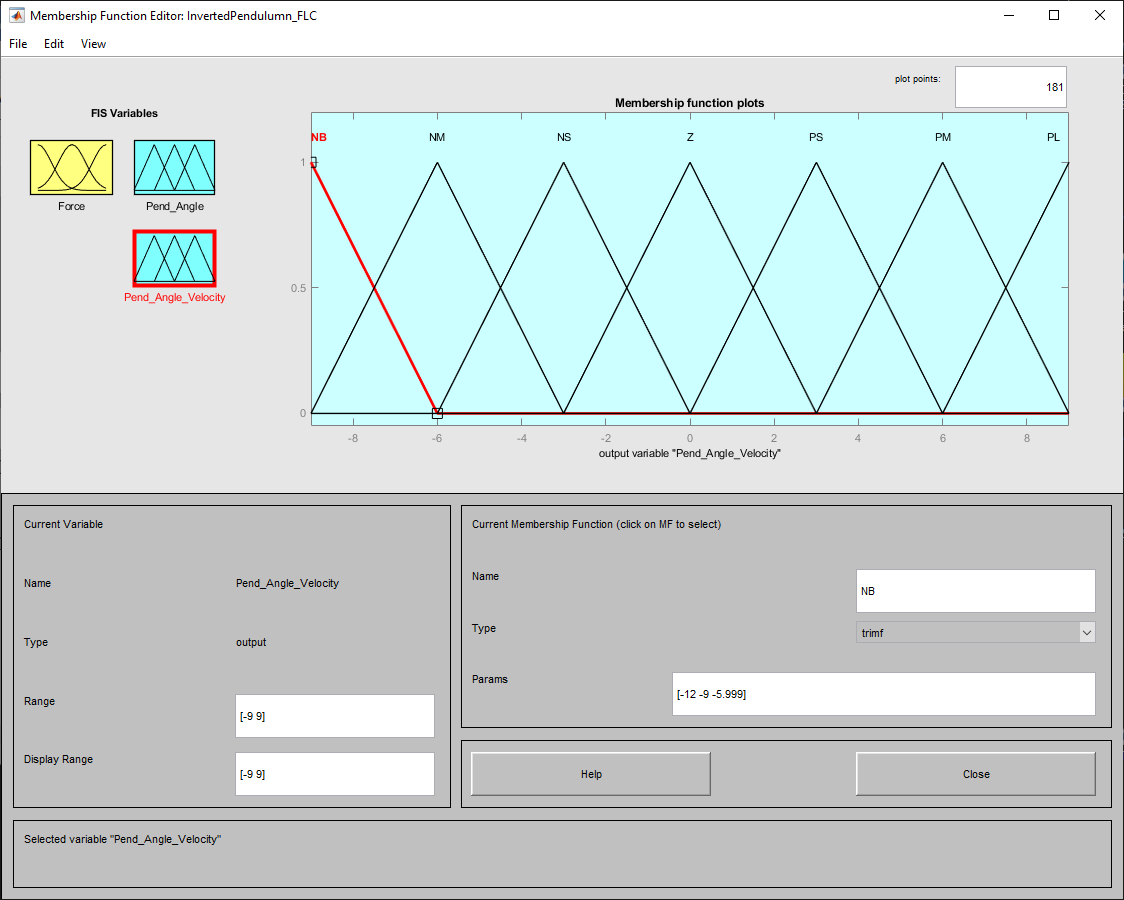


Fig 3.5 FIS editor: Angular velocity

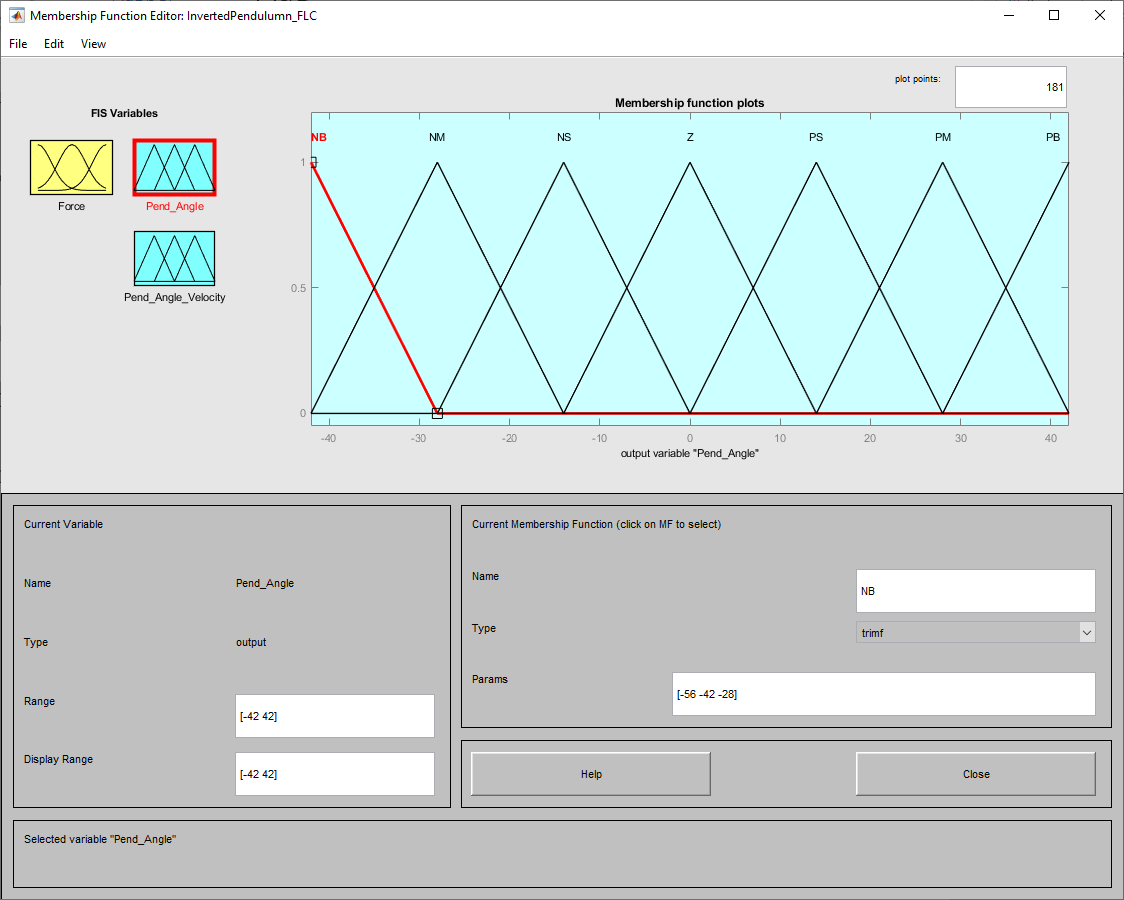


Fig 3.6 FIS editor: Pendulum\_Angle

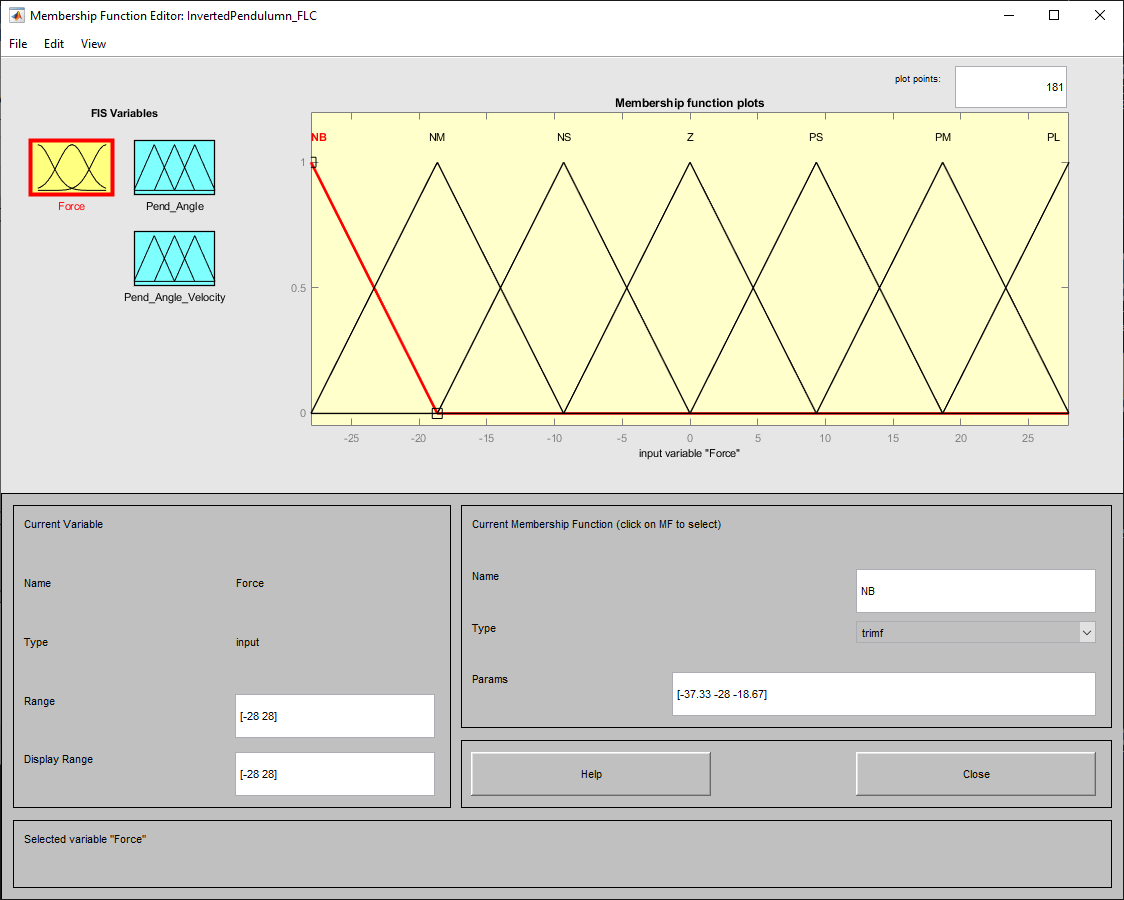


Fig 3.7 FIS editor: Output\_Force

Knowledge Base:

This is where the system’s behavioural rules are mapped from numerical to segregated linguistic sets.

The system uses ‘AND’ logic for comparison between two inputs and gives a decision with ‘then’ as output. Based on the 7 membership functions, 49 behaviour rules have been expressed in the rule table below:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Inverted Pendulum Fuzzy logic Rule Mapping:** Negative Big (NB), Negative Medium(NM), Negative Small,  Zero error (Z), Positive small (PS), Positive Medium (PM) and Positive Big (PB). | | | | | | | |
| Ang\_Vel  θ | NBAng\_Vel | NMAng\_Vel | NSAng\_Vel | ZAng\_Vel | PSAng\_Vel | PMAng\_Vel | PBAng\_Vel |
| NBθ | NBForce | NBForce | NMForce | NMForce | NSForce | NSForce | NSForce |
| NMθ | NBForce | NMForce | NMForce | NSForce | NSForce | NMForce | NMForce |
| NSθ | NMForce | NSForce | NSForce | NSForce | PSForce | PSForce | PSForce |
| Zθ | NBForce | NMForce | NSForce | NSForce | PSForce | PMForce | PBForce |
| PSθ | PMForce | PSForce | PMForce | PSForce | ZForce | NSForce | NSForce |
| PMθ | PBForce | PMForce | PSForce | PMForce | PSForce | ZForce | ZForce |
| PBθ | PBForce | PMForce | PMForce | PSForce | PSForce | PMForce | PSForce |

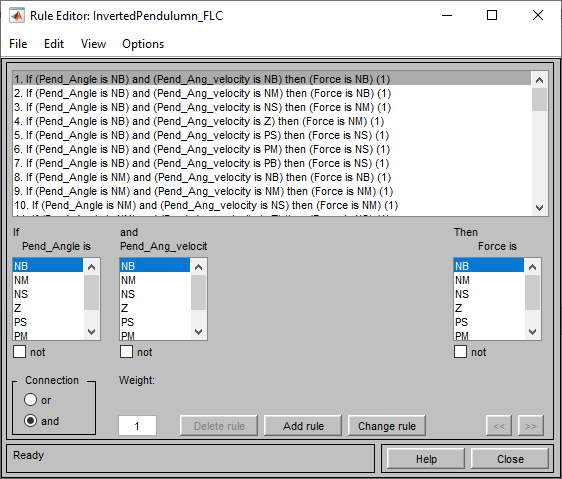


Fig 3.8 Behavioural Rule Editor

For defuzzification of the input membership function sets, the centroid method is used which is based on the following formula:

xCentroid =

where *μ*(*xi*) is the membership value for point *xi* in the universe of discourse. (uk.mathworks.com, n.d.) In brief this method samples data on the centre point of the fuzzy set’s x-axis and returns an average value. This method is used quite commonly in most application as it gives a uniform data distribution.

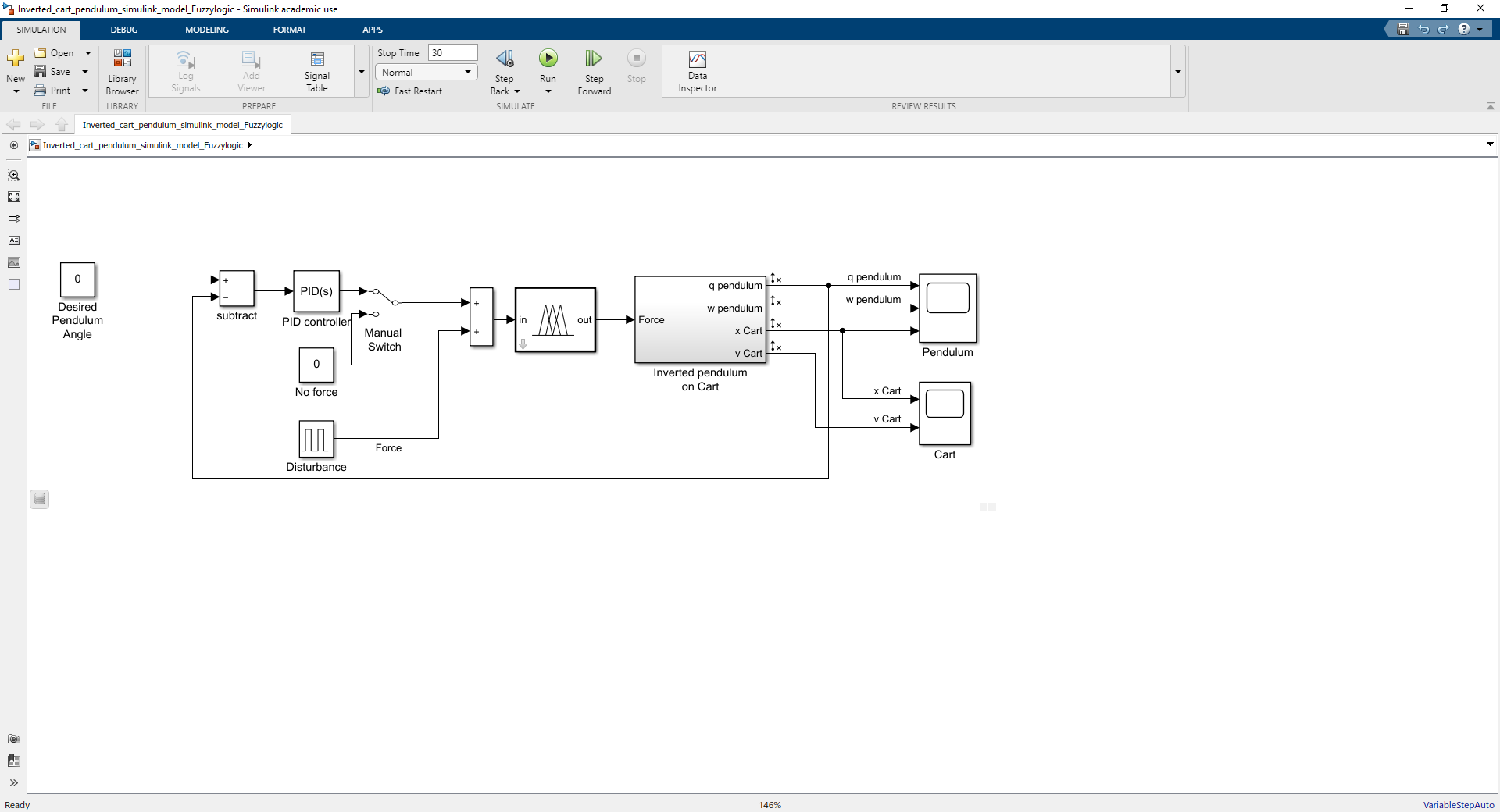


Fig 3.9 Simulink block diagram with Fuzzy controller integrated.

Analysis of Results:



Fig 4: Rule Viewer

This red line Illustrates the weightage of the input with respect to consequential output.

The Rule viewer shows the weightage placed on each parameter and by moving the red line placed on the input, change in output can be observed accordingly. This will verify the asserted behavioural rules defined for the system.

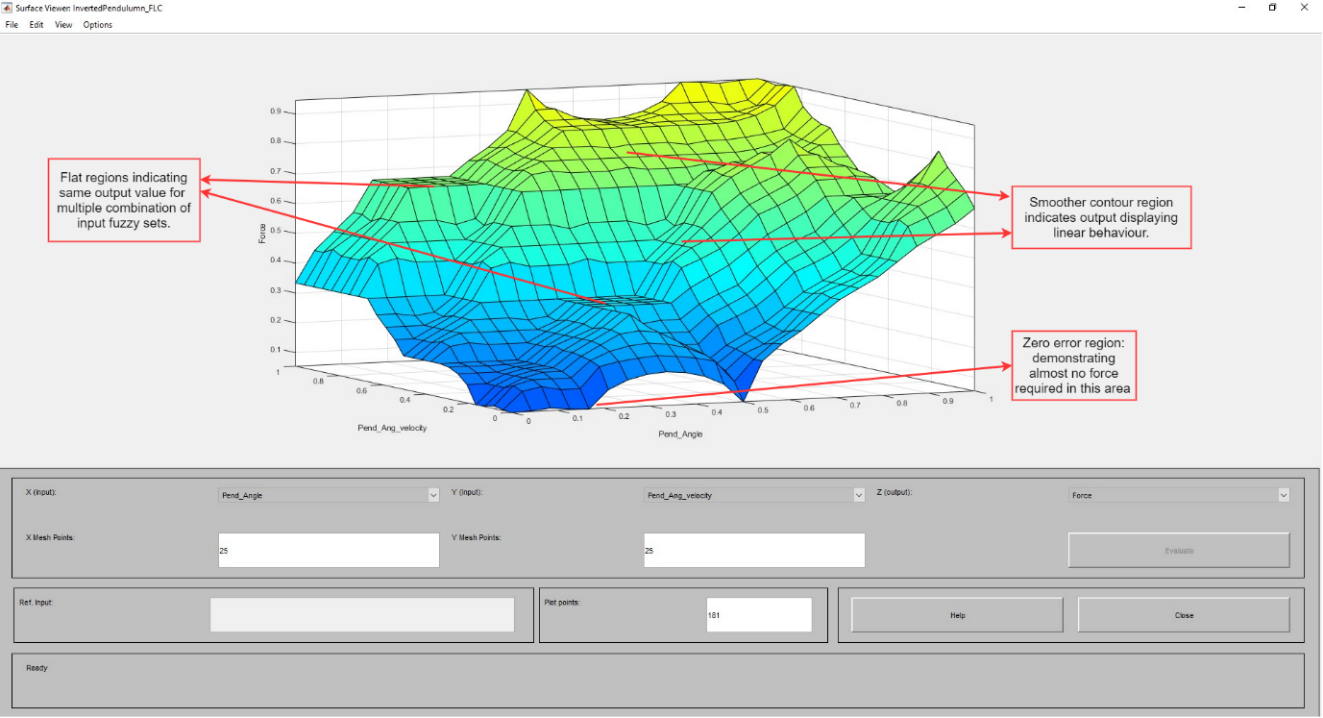


Fig 4.1: 3-D Surface Viewer

The surface viewer illustrates the three-dimensional behaviour model of the system based on the implemented fuzzy rules. This illustration mainly comes in handy when you want to display more than to parameters in one plot.

The transition of the system’s operation from bottom to top contains a mix of flat region and smooth contour. As annotated the flat region seems to be collating multiple input combination of fuzzy sets for one output value and the smoother contours exhibits progressive linearity in output.

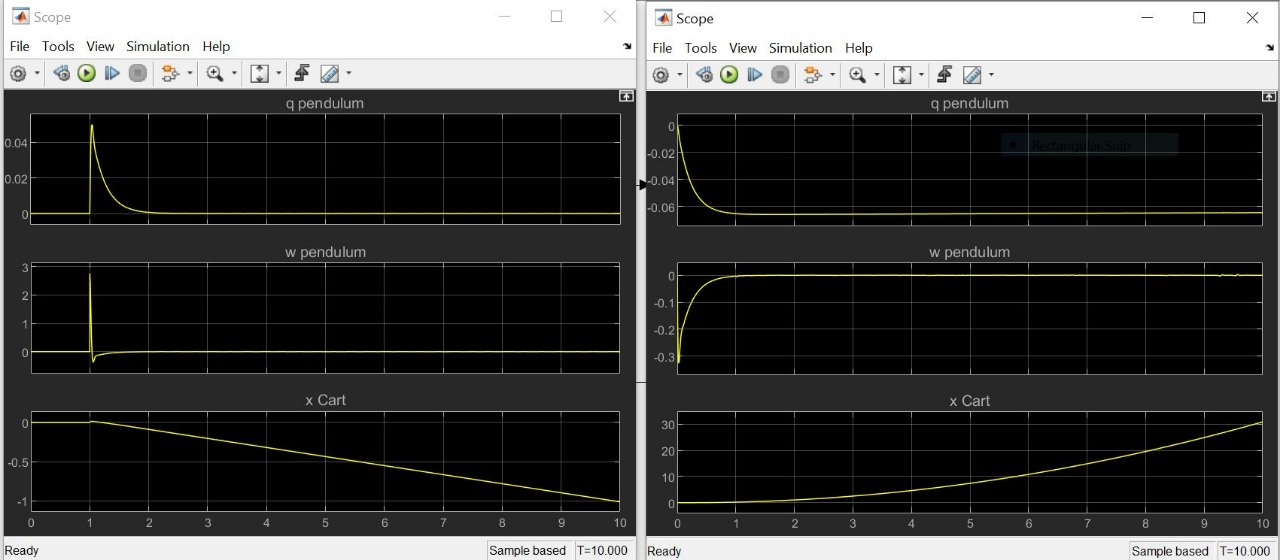


Fig 4.2 System Response PID(left), Fuzzy(Right).

The FLC in addition sums the PID controller’s effort to attain a better system response with quicker settling time and reduction of the signal overshoot. In overall using both (PID and FLC) increases robustness and minimises the disturbance effect on the system.

References and bibliography:

* Maxfield, M. (2011). EETimes - Gray Code Fundamentals - Part 1 -. [online] EETimes. Available at: https://www.eetimes.com/gray-code-fundamentals-part-1/# [Accessed 3 May 2020].
* Gouda, M.M., Danaher, S. and Underwood, C.P. (2000). Fuzzy Logic Control Versus Conventional PID Control for Controlling Indoor Temperature of a Building Space. IFAC Proceedings Volumes, 33(24), pp.249–254.
* Mcneil, Martin (1994). Fuzzy Logic A Practical Approach. Morgan Kaufmann Pub.
* Abdelhameed, E., Hassan, T., Marwa, M., Hamed, M. and El-Saady Ahmed, G. (2019). Design of Hybrid Fuzzy and Position-Velocity Controller for Precise Positioning of a Servo System.
* uk.mathworks.com. (n.d.). Defuzzification Methods - MATLAB & Simulink - MathWorks United Kingdom. [online] Available at: https://uk.mathworks.com/help/fuzzy/defuzzification-methods.html.