Instructions for the laboratory work

Fuzzy Logic Control

1. Mamdani Control of Inverted Pendulum (50 minutes)

This task introduces you to Mamdani fuzzy control and gives you the basic understanding of Fuzzy Inference System (FIS) design in MATLAB.

1.1. Open simulation

Open folder invpen_mamdani, set the MATLAB path to the folder, open `invpen_mamdani.mdl` and explore the model.

The model contains the inverted pendulum model and a Mamdani FIS, which was discussed as an example during the lecture and is provided in the Passino book. Specify different set points and observe the behavior of the system.

In the Cart & Pole Dynamics block the initial value of pole angle is equal to 0.1 rad. You can change the initial angle and angular velocity values, but do not change the lower physical specifications. Also you may replace the Constant 0 with some other signal source, like the one provided with the model.

DO NOT change the target position function type in the animation window.

1.2. Open the FIS in the MATLAB FIS Editor by either:

Write the command `fuzzy` in the MATLAB command window, in the opened editor choose File->Import->From Workspace and enter FIS variable name (default `fismatrix`).

OR write `fuzzy invpen_mamdani.fis` in the command window.

Get acquainted with the editor through the menu options.

1.3. Open `mamdani_fis_code.m`

This script shows you how to create a simple FIS with MATLAB code.

1.4. Try to tune the FIS

As you can see, the controller at hand is not very efficient. The cart just rolls in the direction of angular displacement, instead of swinging the pendulum back and forth and balancing it. Try to tune the FIS to improve the situation as you see fit.

FIS internal tuning:

Choose a method to change FIS parameters from two options: FIS Editor or coding a script. (You also may try both.)

Try to intuitively locate the problem and make the corresponding changes to the FIS.

Save your FIS with a different file name, e.g. `invpen_mamdani1.fis`, so you will have the original FIS just in case. In the file `init_m.m` change the FIS file name to your chosen.

Make changes to FIS parameters and observe the reaction. You may also add additional inputs, e.g. cart position.

FIS external tuning:

Use the input and output gains of the FIS. External tuning may be performed in hand with internal tuning. Recommended gain value interval is (0,10].

2. Sugeno Control of Inverted Pendulum (15 minutes)

This task introduces you to Sugeno fuzzy control. The system is already tuned.

2.1. Open simulation

Open folder invpen_sugeno, set the MATLAB path to the folder, open `invpen_sugeno.mdl` and explore the model.

Observe the behavior of the control system. Here the initial state of the Pole & Cart are all zero, you may change those. Also in the animation window you can change the target position function type.

2.2. Open Sugeno FIS

Open `invpen_sugeno.fis` as described in 1.2.

Study the Sugeno FIS and establish the differences in the inference mechanism, reviewed during the lecture.

3. System Identification with ANFIS (25 minutes)

This task introduces you to the Adaptive Neuro-Fuzzy Inference System (ANFIS).

3.1. Open simulation

Open folder invpen_identify, set the MATLAB path to the folder, open `invpen_sugeno.mdl` and explore the model.

You will find it identical to the system from Task 2, except for the To Workspace block.

3.2. Open ANFIS Editor

Open the editor using the `anfisedit` command. Explore the editor GUI.

3.3. Gather training and testing data, train ANFIS

Change the To Workspace block variable name to `train_data`, choose a target position function type in the animation window and run the simulation for two signal cycles. Change the To Workspace block variable name to `test_data`, choose a different target position function type in the animation window and run the simulation for two signal cycles.

Load the training data from workspace into ANFIS Editor, specify FIS properties as you see fit and train. After training load testing data and perform a test.

Save the trained FIS in a file with a different name, e.g. `anfis.fis`, change the FIS name in `init_s.m` to this name and perform the simulation.

Open the FIS Editor and compare the trained FIS with the Sugeno FIS from Task 2.

3.4. Increase the quality of control

Generate more training and testing data using other target position function types, load your trained FIS into the ANFIS editor and train it using new data. Test with new test data.

Save the FIS and load it into the simulation. Observe the changes.

Try to generate a training dataset, that will train the FIS for all (Sinusoid, Square, Saw) kinds of target position function types.

Open the FIS Editor and compare the trained FIS with the Sugeno FIS from Task 2.