ISTANBUL TECHNICAL UNIVERSITY
ELECTRIC - ELECTRONICS FACULTY

Intellingent Control Systems
2021 Spring
(CRN: 22921)

Assoc. Prof. Gülay Öke Günel
Res. Asst. Gökçen Devlet Şen

Homework 2

İbrahim Halil Bayzan – 040170403

01/06/2022

Contents

List of Figures

Neural Network Controller

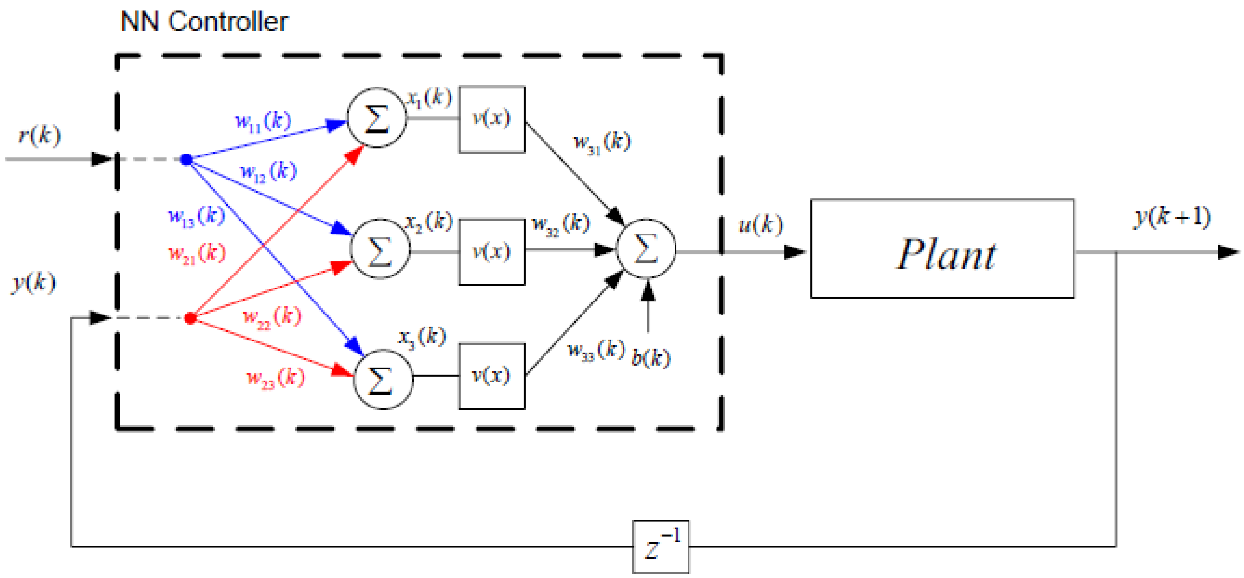

Figure 1: Block diagram with neural network structure

It is desired to design a neural network controller for a nonlinear time varying plant, see Figure 1, that has the following input-output relationship and some prespecified initial weights and hyperparameters.

$$y(k+1) = \frac{1.2(1-0.8e^{-0.1k})y(k)}{1+y^2(k)} + u(k)$$

Jacobian of the system is approximated with

$$\frac{\partial y(k+1)}{\partial u(k)} = \begin{cases} 1, & u(k) - u(k-1) \cong 0 \\ \dfrac{y(k+1) - y(k)}{u(k) - u(k-1)}, & else \end{cases}$$

Furthermore, tan-sigmoid function is chosen as the activation function in the hidden layer and is represented with σ.

a) Obtaining the update rules

In order to obtain the update rule for weights, it is necessary to compute each partial derivative from output to the weight to be updated. Update rules can be expressed as below

$$w_{31}(k+1) = w_{31}(k) - \eta \frac{\partial J}{\partial e}\frac{\partial e}{\partial y}\frac{\partial y}{\partial u}\frac{\partial u}{\partial w_{31}}$$

$$w_{32}(k+1) = w_{32}(k) - \eta \frac{\partial J}{\partial e}\frac{\partial e}{\partial y}\frac{\partial y}{\partial u}\frac{\partial u}{\partial w_{32}}$$

$$w_{33}(k+1) = w_{33}(k) - \eta \frac{\partial J}{\partial e} \frac{\partial e}{\partial y} \frac{\partial y}{\partial u} \frac{\partial u}{\partial w_{33}}$$

$$w_{11}(k+1) = w_{11}(k) - \eta \frac{\partial J}{\partial e} \frac{\partial e}{\partial y} \frac{\partial y}{\partial u} \frac{\partial u}{\partial v_1} \frac{\partial v_1}{\partial x_1} \frac{\partial x_1}{\partial w_{11}}$$

$$w_{21}(k+1) = w_{21}(k) - \eta \frac{\partial J}{\partial e} \frac{\partial e}{\partial y} \frac{\partial y}{\partial u} \frac{\partial u}{\partial v_1} \frac{\partial v_1}{\partial x_1} \frac{\partial x_1}{\partial w_{21}}$$

$$w_{12}(k+1) = w_{12}(k) - \eta \frac{\partial J}{\partial e} \frac{\partial e}{\partial y} \frac{\partial y}{\partial u} \frac{\partial u}{\partial v_2} \frac{\partial v_2}{\partial x_2} \frac{\partial x_2}{\partial w_{12}}$$

$$w_{22}(k+1) = w_{22}(k) - \eta \frac{\partial J}{\partial e} \frac{\partial e}{\partial y} \frac{\partial y}{\partial u} \frac{\partial u}{\partial v_2} \frac{\partial v_2}{\partial x_2} \frac{\partial x_2}{\partial w_{22}}$$

$$w_{13}(k+1) = w_{13}(k) - \eta \frac{\partial J}{\partial e} \frac{\partial e}{\partial y} \frac{\partial y}{\partial u} \frac{\partial u}{\partial v_3} \frac{\partial v_3}{\partial x_3} \frac{\partial x_3}{\partial w_{13}}$$

$$w_{23}(k+1) = w_{23}(k) - \eta \frac{\partial J}{\partial e} \frac{\partial e}{\partial y} \frac{\partial y}{\partial u} \frac{\partial u}{\partial v_3} \frac{\partial v_3}{\partial x_3} \frac{\partial x_3}{\partial w_{23}}$$

where,

$$\frac{\partial J}{\partial e} = e_{tr}, \qquad \frac{\partial e}{\partial y} = -1, \qquad \frac{\partial u}{\partial v_i} = w_{3i}, \qquad \frac{\partial v_1}{\partial x_i} = \sigma'(x_i), \qquad \frac{\partial x_i}{\partial w_{1i}} = r, \qquad \frac{\partial x_i}{\partial w_{2i}} = y,$$

$$\frac{\partial u}{\partial w_{3i}} = v_i; \quad \text{where i} = 1, 2, 3$$

since

$$J\big(e_{tr}(k)\big) = \frac{1}{2} e_{tr}^2(k)$$

and

$$e_{tr}(k) = r(k) - y(k)$$

b) Plotting signals and weights

Using these update rules, it is possible to obtain the following signals.

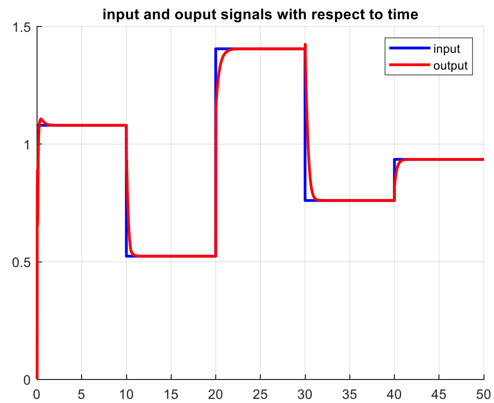

Figure 2: Input and output signals with respect to time

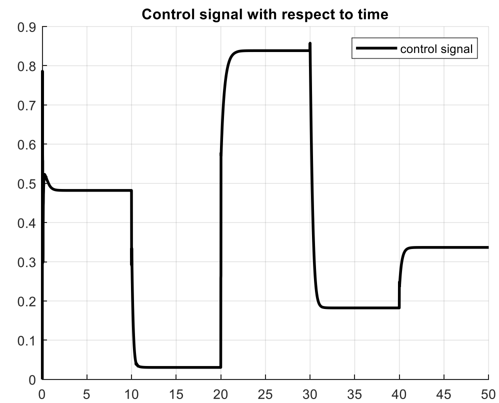

Figure 3: Control signal with respect to time

It can be seen that the system is able to track the input signal. Furthermore, the change of weights over time is found as below.

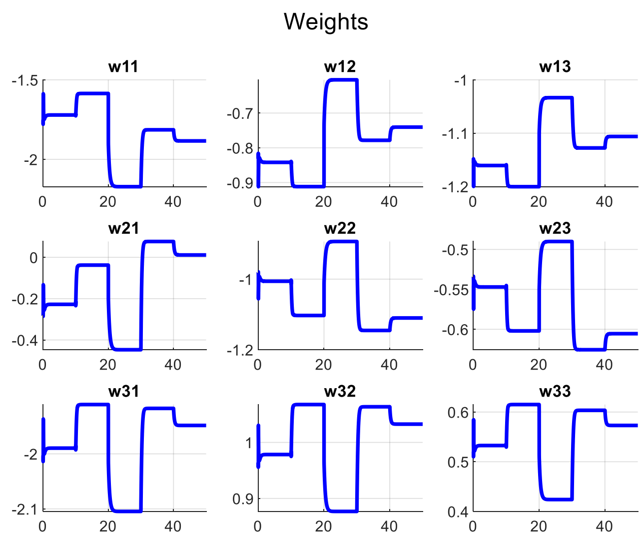

Figure 4: Weights with respect to time

PID Type Fuzzy Controller

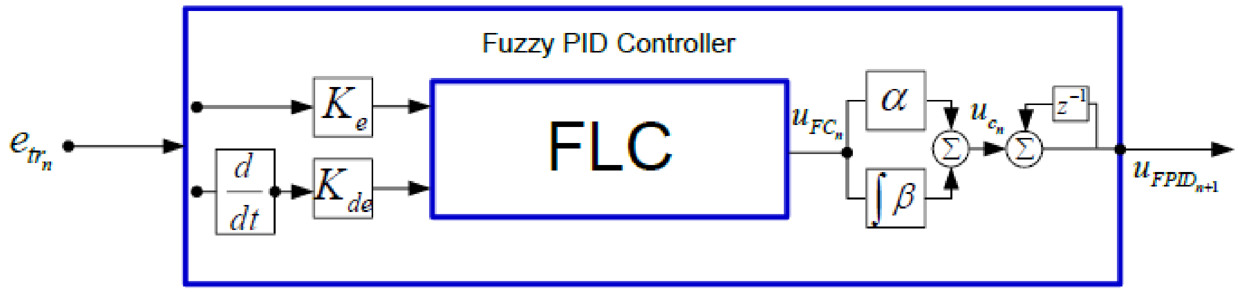

Figure 5: Fuzzy PID controller structure

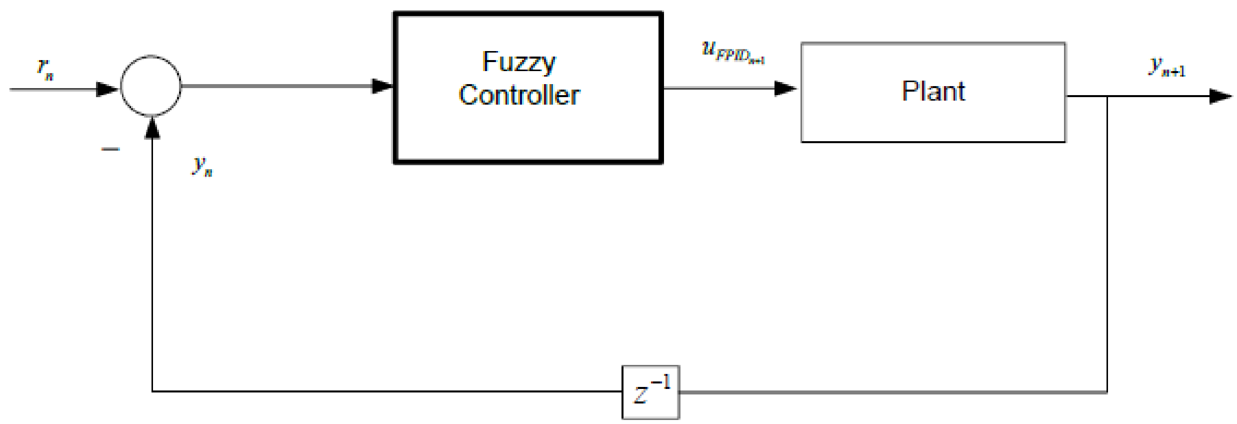

Figure 6: Block diagram with fuzzy controller

It is desired to design a PID type fuzzy controller for a nonlinear time-varying continuously stirred tank reactor (CSTR). There are two inputs to the fuzzy controller which are dependent on the value and the derivative of the tracking error. These inputs are fuzzified using triangular membership functions with $\{-1, -0.4, 0, 0.4, 1\}$. These fuzzified error signals will then be used to calculate to control signal. Moreover, the center of gravity method has been chosen as the defuzzification method.

All these relationships can be listed as below.

$$e_n = K_{e_n} e_{tr_n} \quad \dot{e}_n = K_{de_n} \dot{e}_{tr}$$

$$u_{FC_n} = A_i(e_n)B_j(\dot{e}_n)u_{ij} + A_{i+1}(e_n)B_j(\dot{e}_n)u_{i+1j} + A_i(e_n)B_{j+1}(\dot{e}_n)u_{ij+1} + A_{i+1}(e_n)B_{j+1}(\dot{e}_n)u_{i+1j+1}$$

$$u_{FPID_{n+1}} = U_{FPID_n} + \alpha_n u_{FC_n} + \beta_n[u_{FC_n} + u_{FC_{n-1}}]$$

where u_{ij} is to be determined using Table 1.

	$\dot{e}_{-2}$	$\dot{e}_{-1}$	$\dot{e}_{0}$	$\dot{e}_{1}$	$\dot{e}_{2}$
e_{-2}	-1	-0.7	-0.5	-0.3	0
e_{-1}	-0.7	-0.4	-0.2	0	0.3
e_{0}	-0.5	-0.2	0	0.2	0.5
e_{1}	-0.3	0	0.2	0.4	0.7
e_{2}	0	0.3	0.5	0.7	1

Table 1: Fuzzy control rules

a) Obtaining the control procedure

The control procedure of CSTR using a fuzzy controller can be put in the following order.

1. Calculating tracking error using current states of the system

2. Obtaining the inputs of the controller using K_e, K_{de} and tracking error

3. Determining the output of the fuzzy controller with fuzzified error and derivative of error values and correct control rule

4. Calculating the current control signal with a PI controller outside of the fuzzy controller (and feeding itself back)

5. Applying the control signal to the system and recalculating its new states

b) Applying the fuzzy controller without optimization

Without optimizing hyperparameters following signals are obtained.

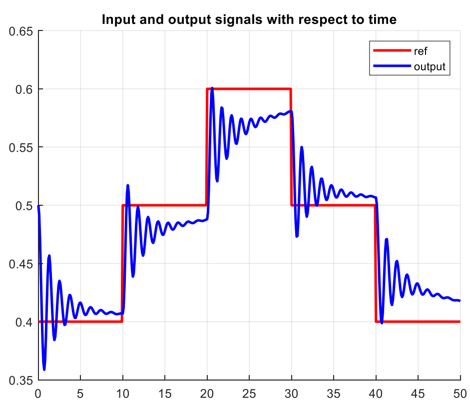

Figure 7: Input and output signals with respect to time

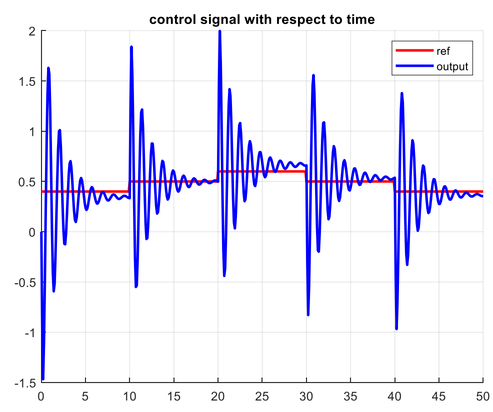

Figure 8: Control signal with respect to time

It is obvious that these signals are not acceptable because not only do they oscillate too much but they also are slow and they have a big steady state error.

c) Applying the fuzzy controller with optimization

As a solution to problems that were encountered in b), genetic algorithm will be used to optimize the hyperparameters which are $K_e, K_{de}, \alpha, \beta$. Choosing objective function

$$J = \frac{1}{2}\sum_{n=1}^{\infty}[r_n - y_n]^2 + \frac{1}{2}\lambda\sum_{n=1}^{\infty}[u_n - u_{n-1}]^2$$

for $\lambda \in \{0.1, 0.5, 1, 10, 100\}$ results in the following signals.

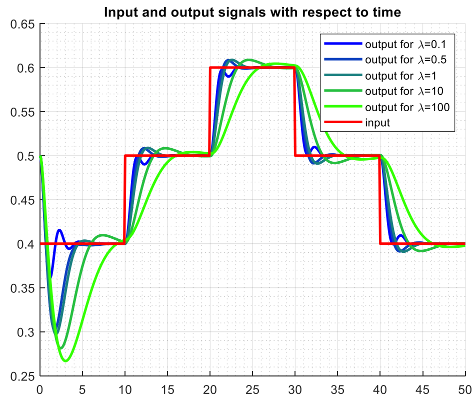

Figure 9: Input and output signals with respect to time

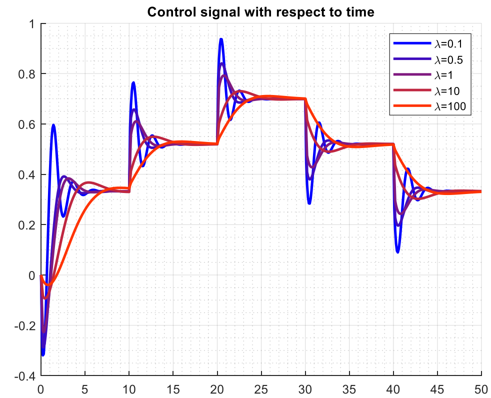

Figure 10: Control signal with respect to time

d) ANFIS

Introduction

The fuzzy controller designed in c) is not adaptive, meaning in case of disturbance or uncertainty it will not work as desired. There are many possible adaptation mechanisms but this work will be focusing on Adaptive Neuro-Fuzzy Inference System (ANFIS), which uses neural networks to adjust the controller.

Architecture and control procedure

The architecture of an ANFIS can be generalized as in Figure 11.

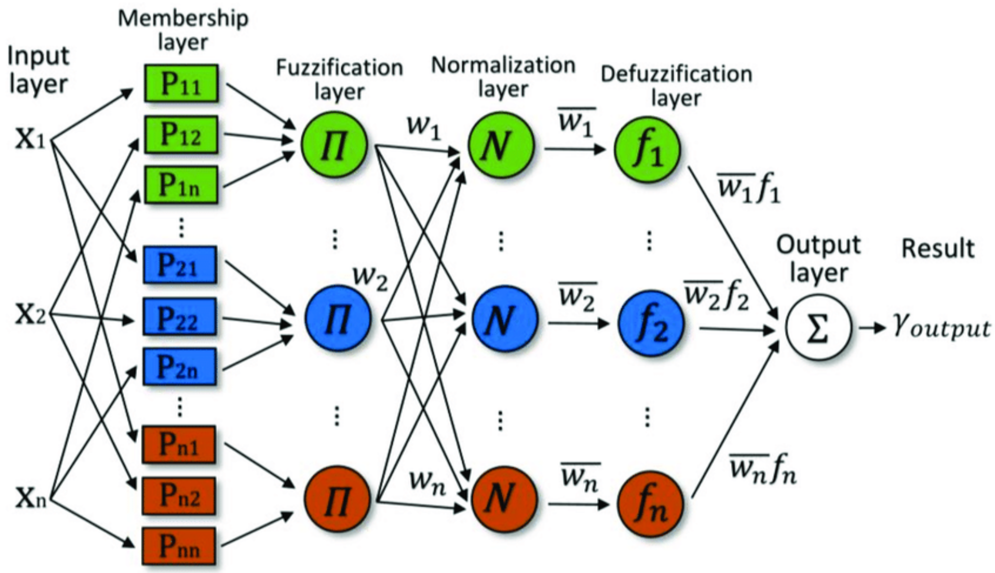

Figure 11: ANFIS Architecture

The control procedure can be expressed as below.

1. Using the input layer and membership functions, the signals for the fuzzification layer are created, and outputs are called the membership grade of a fuzzy set. In this layer, there can be some updatable parameters in the membership functions.
2. Signals are fuzzified in the fuzzification layer with a T-norm operator, outputs are called as the firing strength of a rule.
3. Firing strengths are normalized by dividing each firing strength by the sum of all firing strengths, outputs are called the normalized firing strength of a rule.
4. Normalized firing strength is multiplied with a term that is an affine linear combination of inputs. In this layer, there can be some updatable parameters in the combination term.
5. Overall output is computed is attained by summing the outputs of the defuzzification layer.
6. The backpropagation algorithm starts from the output to updatable layers.
7. The procedure is repeated until achieving desired performance.

Some notes

- Using this controller will decrease the interpretability of the system but it will result in a more successful mapping precision due to the neural networks. Increasing the number of epochs might lead to meaningless or non-orthogonal membership functions
- ANFIS is a universal approximator, meaning it has a mathematical background that it will learn any function under certain circumstances.

- The control procedure that is mentioned above is based on Sugeno fuzzy model, however, a more complicated procedure using Mamdani fuzzy model can be achieved. Unfortunately using this is not worth it because it brings too much complexity along with an indefinite approximation power.