

# Design of Bang-bang Controller based on A Fuzzy-Neuro Approach

Grantham K.H. Pang and Samy A. Mesbah

Department of Electrical & Computer Engineering

University of Waterloo

Waterloo, Ontario, Canada N2L 3G1.

email: pang@kingcong.uwaterloo.ca

## Abstract

A fuzzy-neuro approach for the design of bang-bang controller is presented in this paper. The approach has been used with success for the time optimal bang-bang control of a heating system. The improved bang-bang controller suppresses the oscillations often observed at the output of an on-off controller. A fuzzy system is used for the implementation of the on-off control. An extension of the fuzzy control is provided by an equivalent neural network of the fuzzy system. A test application, that of a house heating with a two-state furnace, is developed and evaluated with standard hysteresis switching, fuzzy control, and fuzzy-neuro control.

## 1. Introduction

There are many control applications which require a controller to output one of two states: "on" or "off". Examples include attitude control using an on/off thruster [3] and flow control valve operated in a bang-bang mode [11]. These applications are not well suited to standard linear control design methodologies. A switching technique involving hysteresis is commonly used for these types of problems and this is adequate for many applications. The major drawback of this method is a lack of robustness, that is the dead zone is chosen to optimize performance for one nominal plant under one set of conditions.

One of the recent control philosophies is fuzzy control. A fuzzy controller can be intuitively designed with relative simplicity and it has the potential to be very robust. In addition, the controller can be easily designed so that its output can be restricted to any range desired. Fuzzy control has the drawback that proof of stability is generally difficult. Also, it is sometimes not so intuitive how to improve the performance of a fuzzy controller.

Artificial neural networks (ANN) have also found use in control systems. One of the major drawbacks in using

an ANN for control is that there is so much freedom in structural implementation choice that it is often difficult to decide how complex a structure is actually necessary for the desired control. Furthermore, the implementation is not at all intuitive and the inner workings of the network are very much invisible to the designer.

An improvement can be made by integrating fuzzy control with neural networks, which gives a fuzzy neural network (FNN). A fuzzy controller is first obtained which is the "starting point" of the FNN. An ANN is then developed and its structure is completely defined by the pre-designed fuzzy controller. Performance can then be improved with real or simulated data. The structure is such that the internal workings are known to the designer as they relate to fuzzy controller components.

The paper is outlined as follows. Section 2 focuses on fuzzy-neuro control. Section 3 deals with the development of a specific on/off control application: house temperature control. This will be used to test the fuzzy and fuzzy-neuro control methodologies. Section 4 deals with the development of a fuzzy controller for the test application. The implementation details of the corresponding FNN are then detailed in Section 5 along with the methods used to improve controller performance.

## 2. Fuzzy-Neuro Control

There have been many papers on the modelling of fuzzy systems with ANN approaches. All techniques are based on the equivalence of specific classes of rule-based fuzzy systems and the type of neural network proposed [7]. Most models involve using a structured ANN which is functionally equivalent to the original fuzzy system and using some sort of gradient descent (generally back propagation or some very similar variant) method to allow the network to learn with real data. Some approaches, like Jang [6], Berenji [2], Horikawa [4], and Ishibuchi *et al.* [5] provide generalised methodologies which can be applied to a wide range of

fuzzy systems. Other approaches, like the one proposed by Yamakawa [13], are more efficient but work for a more specific class of systems — in this case systems which contain only triangular complementary membership functions. These approaches have been successful in producing workable fuzzy-neuro controllers but share a lack of intuitiveness in the resulting neural network.

This section presents a method which allows the benefits of a neural network approach to be realized without sacrificing the intuitive nature of fuzzy control. The proposed model allows full access to the components of the fuzzy controller even after learning. This allows this type of setup to simultaneously share the benefits of both fuzzy control and neural network capabilities.

### 2.1. Fuzzy Neural Network (FNN)

The approach which will be described and used was first proposed by Wang and Kim [12]. The system is subdivided into three main functional layers connected in sequence: fuzzification, rule reasoning, and defuzzification. Each of the three layers can be constructed and trained independently and the connections between the networks made afterward. This allows the different components of the fuzzy controller to be trained into the corresponding components of the FNN to produce a network which is practically identical to the original controller.

The advantage of this method is that it is very intuitive. The structure of the network represents the structure of a fuzzy controller and separate sub-networks correspond to the individual components of the original fuzzy controller. This means the membership functions (input and output) and rules can be retrieved from the FNN after training.

## 3. Application to Temperature Regulation

In this section, a model of the thermal characteristics of a typical house is developed. As with most system modelling, the model for a furnace/thermostat system in a house can be chosen with almost arbitrary complexity [10]. The key is to choose a model that is adequate for the problem at hand. The plant model includes the furnace itself (which can only have two states: “on” and “off”), along with a simple model of the temperature change when the furnace is at either of these two “steady-state” operating points. Note that the model must also take into account heat loss, which depends on *outside* temperature and insulation.

### 3.1. Furnace

The “on” state furnace in a house can be modelled for control purposes as a constant gain. The input, which will be denoted as  $m(t)$ , can be “0” or “1” and the gain simply converts the “1” value to the heat output of the

furnace in a desired set of units (which will be energy per unit time). This gain, which will be denoted as  $Q_f$ , will be a parameter which should be varied for different simulations to account for different furnace types.

### 3.2. Heating Phenomena

The model of adding heat to a fluid (air) is fairly common and detailed in Roots [10]. To make simulation manageable, the heating phenomenon will be modelled as a *simple* process, which can be represented by an ordinary differential equation, which is of first or second order, linear and with constant coefficients.

As a precursor to the development of a mathematical expression for the heating process, some parameters needed by the model must be defined:

- $\theta(t)$  — inside temperature
- $\theta_a(t)$  — ambient (outside) temperature
- $\tau$  — time delay from controller action to effect seen at the controller temperature sensor
- $R_e$  — equivalent thermal resistance between temperature measurement point ( $\theta$ ) and the outside ( $\theta_a$ )
- $C_e$  — equivalent thermal capacitance (as above)
- $T = R_e C_e$  — time constant related to insulating properties of house
- $\Gamma = Q_f R_e$  — constant describing achievable equilibrium temperature difference between temperature measurement point ( $\theta$ ) and the outside ( $\theta_a$ )
- $m$  — this is the furnace input, that is “1” or “0”

The differential equation which describes the heating process can be derived from the treatment given by Roots [10]:

$$T\dot{\theta}(t) + \theta(t) = \Gamma m(t - \tau) + \theta_a(t). \quad (1)$$

Note that this is the same as a standard electrical  $RC$  circuit where  $\theta_a$  represents the ground potential and  $\Gamma$  is the potential difference across the  $RC$  network. Taking the Laplace transform, the plant model can be written as:

$$\theta(s) = m(s) \frac{\Gamma e^{-s\tau}}{1 + sT} + \theta_a(s) \frac{1}{1 + sT} \quad (2)$$

This model contains a pure time delay which cannot be represented in state-space form. Due to the implementation methodology for the fuzzy controller it is necessary to represent the system in state-space form. For this reason, the pure time delay ( $e^{-s\tau}$ ) term is approximated linearly by  $\frac{1}{1+s\tau}$ , so the plant model now becomes:

$$\theta(s) = m(s) \frac{\Gamma}{(1 + sT)(1 + s\tau)} + \theta_a(s) \frac{1}{1 + sT} \quad (3)$$

Parameter	Units
$\theta$	$^{\circ}C$
$\theta_a$	$^{\circ}C$
$Q_f$	Watts
$R_e$	$^{\circ}C/Watt$
$C_e$	$Joule/^{\circ}C$
$\tau$	seconds
$\Gamma$	$^{\circ}C$
$T$	seconds

Table 1: Units for model parameters

This approximation may not actually be too bad since a real furnace will actually heat up gradually over a period of time, that is  $\tau$  is actually a combination of pure delay and the time for the furnace to reach the steady-state output value. This approximated time delay is modeled as an exponential rise over the delay time.

### 3.3. Model Uncertainties

It is clear that there are uncertainties inherent in the model. First, heat loss will depend on the actual house to be heated. The outside temperature is bound to fluctuate within a range of values for different seasons (and even times of day). As well, there will be a constant time delay based on the furnace start-up time and location of the heat source with respect to the temperature sensor. These are simple uncertainties. One could pursue even more complex uncertainties such as temperature gradients with rooms and the house as a whole. This controller design will attempt to minimize the effect of the simple uncertainties on temperature regulation.

**3.3.1 Parameter Ranges:** The previous section elucidates parameters which can be different for every environment in which the proposed controller may be used. The thermal resistance and capacitance ( $R_e$  and  $C_e$ ), furnace heat output ( $Q_f$ ), and time delay ( $\tau$ ) are all uncertain model parameters. In addition, the ambient temperature ( $\theta_a$ ) is considered a disturbance so its curve will almost always be uncertain. It is clear at this point that the identified uncertainties are all structured sets, that is upper and lower bounds can be specified for each parameter.

It is necessary to determine the ranges for each of the model parameters which would be considered a reasonable representations of all situations the controller will face. Table 1 shows the units of model parameters and variables used for this example. With a well-defined set of units, it is now possible to investigate the ranges of the given parameters. The source of this information is the American Society of Heating, Refrigerating, and Air Conditioning Engineers (ASHRAE) 1985 Fun-

Parameter	Lower Bound	Upper Bound
$Q_f$	13000	44000
$R_e$	$6.08 \times 10^{-4}$	$7.30 \times 10^{-3}$
$C_e$	9075	27225
$\tau$	120	900
$\Gamma$	7.90	321.16
$T$	5.51	198.72

Table 2: Summary of parameter bounds

damentals Handbook [1]. It should be noted that most parameter limits are assumed worst cases and these are not directly stated in any source. Table 2 shows a summary of the calculated parameter bounds.

**3.3.2 Nominal and Perturbed Plant Transfer Functions:** The nominal plant is chosen to be as "average" as possible, that is an average sized house with a mid-size furnace and fairly good insulation. The extremes are chosen as just that: the fastest and slowest heating houses from the calculated parameter ranges. The nominal plant transfer function is chosen as:

$$P_{nom} = \frac{66.15}{(41.4s + 1)(300s + 1)} \quad (4)$$

### 3.4. On/Off Switching Control Law

Most thermostats today use a simple on/off switching technique based directly on the error signal, that is when the temperature is at the desired temperature (or even slightly below) switch the furnace off. When the temperature difference of the house and the reference is beyond a specified threshold turn the furnace on. This can be written as a conditional statement:

```

If (furnace off AND error > on threshold)
  Turn furnace on
Elseif (furnace on AND error < off threshold)
  Turn furnace off
End

```

This is probably the simplest control law to state but it is nonlinear so analysis is not as simple a task as with linear controllers.

## 4. Fuzzy Controller Design

This section deals with the design of a fuzzy controller for the test application of temperature control. A number of different schemes for fuzzy control were tried but for the sake of brevity, only the results of the best fuzzy scheme will be given. It should be mentioned that all

results using a fuzzy controller were considerably better than with on/off switching. The choice of rules and membership functions was based mostly on intuition and rules of thumb, along with "tweaking" once acceptable performance was achieved.

The simulation of the fuzzy temperature control system was accomplished with MEDAL [9], a MATLAB like software package which includes built-in fuzzy logic and expert system constructs.

#### 4.1. Membership Functions

All attempted schemes involved defining the temperature error and the controller output as linguistic variables. The most sophisticated scheme also defined the change in temperature from the previous time step as a linguistic variable to act as an indicator of the heating rate.

The temperature error linguistic variable is associated with fuzzy numbers called *small*, *medium*, and *large*. The temperature change linguistic variable is associated with fuzzy numbers called *negative*, *small*, *medium*, and *large*. The output linguistic variable is associated with fuzzy numbers *on*, *mid*, and *off*.

The *mid* membership function represents a case where it is not clear whether the furnace should be on or off. This was chosen to represent the decision a human operator would face when he or she is not sure of what to do. The defuzzified value will lie in  $[0, 1]$  and is used to determine the value of  $m(t)$  at each time step.

#### 4.2. Rules

The fuzzy rules used were as follows:

```

IF (TempErr is small AND DeltaTemp is negative)
    THEN Output is off
IF (TempErr is small AND DeltaTemp is small)
    THEN Output is mid
IF (TempErr is small AND DeltaTemp is medium)
    THEN Output is off
IF (TempErr is small AND DeltaTemp is large)
    THEN Output is off
IF (TempErr is medium AND DeltaTemp is negative)
    THEN Output is mid
IF (TempErr is medium AND DeltaTemp is small)
    THEN Output is on
IF (TempErr is medium AND DeltaTemp is medium)
    THEN Output is mid
IF (TempErr is medium AND DeltaTemp is large)
    THEN Output is off
IF (TempErr is large AND DeltaTemp is negative)
    THEN Output is on
IF (TempErr is large AND DeltaTemp is small)
    THEN Output is on
IF (TempErr is large AND DeltaTemp is medium)

```

```

    THEN Output is on
IF (TempErr is large AND DeltaTemp is large)
    THEN Output is mid

```

#### 4.3. Observations

In essence, the control law looks like hysteresis loops which have separate thresholds for abrupt changes and regulation, and the loop thresholds expand or contract based on the relative speed of the plant in terms of temperature change. This allows overshoot to be minimized *and* good regulation to be achieved. These two objectives conflicted with each other in the on/off controller design attempt.

#### 4.4. Fuzzy Control Simulation

The operation of the fuzzy control system is as follows:

1. execute rules with current temperature error and temperature change from last time step
2. defuzzify output linguistic variable
3. if output  $\geq 0.5$ , then the furnace will be on for the next time step, otherwise it will be off.
4. simulate system for next time step
5. return to step 1

A MEDAL m-file was written to perform the simulation.

### 5. Fuzzy-Neuro Controller Design and Results

This section deals with the implementation of a fuzzy-neuro controller for the test application of temperature control. The fuzzy controller described in the previous section is used as the basis for this FNN.

#### 5.1. Architectural Implementation

The section details the specific implementation of the fuzzy controller described in the previous section as an FNN. The implementation was written in the C programming language. A MATLAB m-file was written to perform the simulations. The fuzzy-neuro thermostat is structured using the guidelines detailed in section 2. This involves two sub-networks in the fuzzification layer, 12 sub-networks in the rule reasoning layer, and the two sub-layers in the defuzzification layer.

**5.1.1 Fuzzification Layer:** The fuzzification layer contains two sub-networks which were trained with the membership functions for the input variables. Each sub-network consists of a single neuron input layer, contains two hidden layers with seven neurons each, and an output layer with three (temperature error) or four (temperature change) neurons in the neuron output layer. Three of the output neurons in each output layer correspond to the *small*, *medium*, and *large* membership values. The fourth output neuron in the temperature change sub-network output layer corresponds to the *negative* membership value. The use of

two hidden layers is an arbitrary design choice which allows a reasonably complex mapping to be trained.

**5.1.2 Rule Reasoning:** The 12 rules used in the original fuzzy controller are implemented with 12 rule reasoning sub-networks. In fact only one sub-network was actually initially trained with a product AND function and its weights and thresholds copied to the other 11 sub-networks.

These sub-networks are composed of two input layer neurons, two hidden layers each with seven neurons, and one output layer neuron. The choice of hidden layers is again arbitrary and was used to allow relatively complex functions to be represented by the sub-network. The sub-networks are initially trained to return the product of the two inputs (which are in the range  $[0, 1]$ ). Hence, the output is also in the  $[0, 1]$  interval.

**5.1.3 Defuzzification Layer:** The  $[0, 1]$  range of the universe of discourse of the output variable is divided into 21 discrete points. The weights are the membership values ( $\mu_i$ ) at each of the 21 discrete output values. It is interesting to note that with the 12 rules there are 12 sets of 21 weights in the fuzzy controls sub-layer and each of these sets can change independently. When the initial weight entry is made, there is much redundant information in the weights but this will change when the entire FNN is trained.

## 5.2. Network Training

The purpose of converting the fuzzy controller to an FNN was to tune the controller with numerical data. The FNN was trained to reduce the amount of overshoot and improvements have been made. Figure 1 shows the input membership functions of the FNN after training. Robustness test was carried out as follows. Since the outside temperature can have an effect on the model behaviour, simulations with  $P_{nom}$  were performed where the ambient temperature rises during the simulation time (figure 2). Figure 3 shows the temperature change when there is a drop in the reference temperature.

## 6. Conclusions

In summary, the objective was to show that fuzzy-neuro control is a viable control strategy for special control problems. It was found that a standard approach (on/off switching) for two-state systems is acceptable but improvements were noticed with a very simple fuzzy controller. The FNN allowed some improvement in an already excellent fuzzy controller performance and could be tuned to any situation with case study data. This indicates that fuzzy and FNN control

are powerful design techniques in certain control problems and, coupled with the ease of implementation, very practical methodologies.

## References

- [1] ASHRAE handbook 1985 fundamentals. American Society of Heating, Refrigerating, and Air Conditioning Engineers, Atlanta, 1985.
- [2] H. R. Berenji. Learning and tuning fuzzy logic controllers through reinforcements. *IEEE Trans. on Neural Net.*, 3(5):724-740, 1992.
- [3] Richard Y. Chiang and Roger J. Jang. Fuzzy logic attitude control for cassini spacecraft. *Proc. 3rd IEEE Conference on Fuzzy Systems*, pages 1532-1537, 1994.
- [4] S. Horikawa, T. Furuhashi, and Y. Uchikawa. On fuzzy modelling using fuzzy neural networks with the back-propagation algorithm. *IEEE Trans. on Neural Net.*, 3(5):801-806, 1992.
- [5] H. Ishibuchi, R. Fujioka, and H. Tanaka. Neural networks that learn from fuzzy if-then rules. *IEEE Trans. on Fuzzy Systems*, 1(2):85-97, 1993.
- [6] J. S. R. Jang. Self-learning fuzzy controllers based on temporal back propagation. *IEEE Trans. on Neural Net.*, 3(5):714-723, 1992.
- [7] Grantham K. H. Pang. Fuzzy-neuro method in control design. NATO Advanced Workshop on Emerging Computing Techniques for Engineering Design at Natplio, Greece. Proceedings to be published by Springer-Verlag, 1994.
- [8] Grantham K. H. Pang, K. Takahashi, T. Yokota, and H. Takenaga. Adaptive route selection for dynamic route guidance system based on fuzzy-neural approaches. submitted to *IEEE Trans. on Vehicular Tech.*, 1995.
- [9] Grantham K.H. Pang. Medal: Matrix & expert system development aid language. *Proc. IEEE Symp. on Computer-Aided Control System Design*, pages 148-155, 1992.
- [10] W. Roots. *Fundamentals of Temperature Control*. Academic Press, New York, 1969.
- [11] R. Ushiyama and H. Walden. Coldplate temperature regulation using fuzzy logic control. *SAE(Society of Automotive Engineers) Trans.*, 100(1):1790-1801, 1991.
- [12] F. Y. Wang and H. M. Kim. Implementing adaptive fuzzy logic controllers with neural networks: a design paradigm. to be published in *Journal of Intelligent and Fuzzy Systems*.
- [13] T. Yamakawa. A neo fuzzy neuron and its applications to system identification and prediction of chaotic behaviour. In J. M. Zurada, R. J. Marks II, and C. J. Robinson, editors, *Computational Intelligence: Imitating Life*, pages 383-395. IEEE Press, 1994.

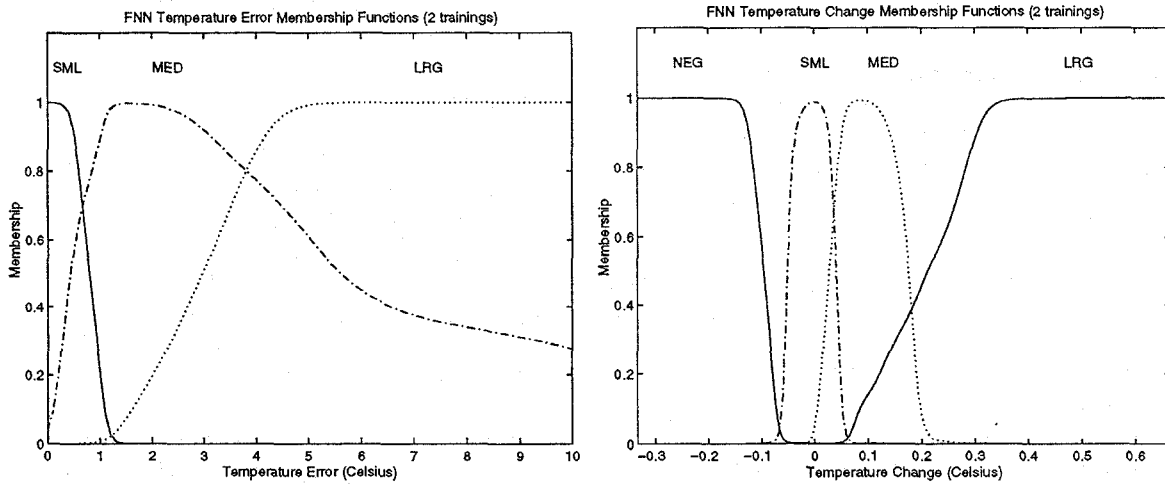


Figure 1. Input membership functions after training

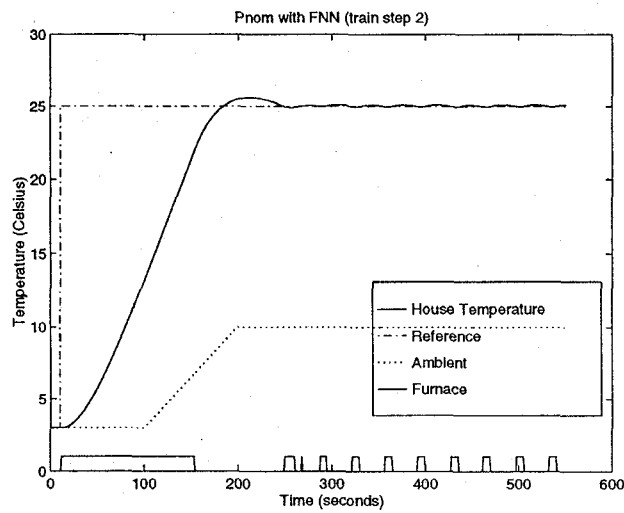


Figure 2.  $P_{nom}$  with ambient temperature change

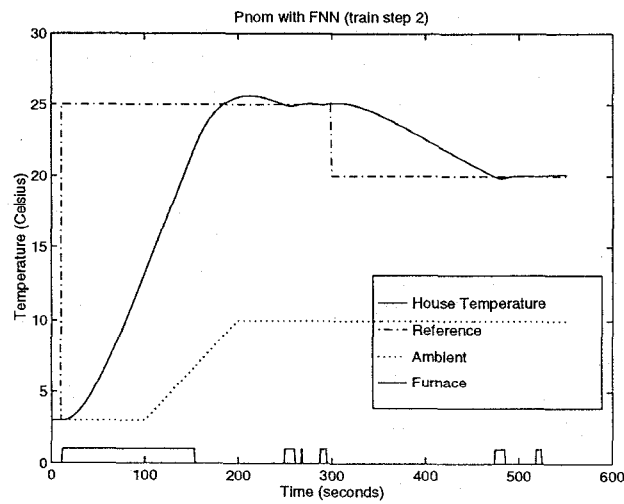


Figure 3.  $P_{nom}$  with ambient and reference temperature change