FUZZY MODELING USING GENETIC ALGORITHM AND RECURSIVE LEAST SQUARES FOR PROCESS CONTROL APPLICATION

RIBHAN ZAFIRA ABDUL RAHMAN 1, RUBIYAH YUSOF 2, MARZUKI KHALID 3, and MOHD FAISAL IBRAHIM 4

1 Department of Electrical and Electronics, Faculty of Engineering, Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia
2, 3 Center for Artificial Intelligence and Robotics (CAIRO), Universiti Teknologi Malaysia, Jalan Semarak 54100 Kuala Lumpur, Malaysia
4 Department of Electrical, Electronics and Systems Engineering, Faculty of Engineering, Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor, Malaysia
1 ribhan@eng.upm.edu.my, 2 rubiyah@utmkl.utm.my, 3 marzuki.khalid@gmail.com, 4 faisal@eng.ukm.my

Abstract: In process identification, traditional mathematical techniques are rather insufficient due to difficulty in modeling of highly nonlinear components in the plants. New methods of modeling based on artificial intelligence techniques such as neural networks and fuzzy logic have shown promising results for modeling of nonlinear plants. In this paper, the modeling of nonlinear processes based on the Takagi-Sugeno fuzzy model with genetic algorithm and the traditional recursive least squares estimates is proposed. The genetic algorithm is used to tune the antecedent fuzzy membership functions automatically to improve the modeling performance and the recursive least squares estimation approach is used to obtain the parameters at the consequent part of the fuzzy model. The proposed approach is applied to a process control rig with three sub-systems: a heating element, a heat exchanger and a compartment tank. The key issues of finding the best models for the plant are described. A comparison between the proposed fuzzy modeling approach with genetic algorithm and recursive least squares and that of a conventional linear modeling approach is discussed. The results show that the proposed technique has better convergence and better performance in modeling of nonlinear processes.

Keywords: Takagi Sugeno fuzzy model, genetic algorithm, recursive least squares and process control

1. INTRODUCTION

Process control is a branch of control engineering that deals with the operation of plants in industries such as petrochemicals, food, steel, glass, paper and energy. It is inherently nonlinear and cannot be effectively controlled using conventional control techniques which are developed on the basis of linearized process models. In many process control systems, time delay, nonlinearity, and multi-interacting loops are among the major difficulties for good control. To have a good control it is very important to get the accurate model.

In this research, we consider a plant consisting of three sub-processes such as a heating element, a heat-exchanger, and a tank with a highly nonlinear and complex model. Due to these complexities, conventional linear model techniques are insufficient to be used for such complex processes. For example, the nature of conventional linear model itself will linearized the system model even though it is the nonlinear model.

There is a range of modeling techniques that attempt to mimic the behavior of nonlinear system. One of them is fuzzy modeling. Fuzzy modeling can be considered as a black box model with has the ability to provide not only heuristic knowledge but also accurate representation of complex non-linear system. The need for fuzzy modeling in nonlinear systems arises due to the inability of conventional modeling techniques in nonlinear system identification. This limitation arises proportionally with the degree of nonlinearity of the system.

Fuzzy modeling refers to a process whereby a dynamical system is modeled in the form of a set of
fuzzy rules and corresponding membership functions [Yi, 1998]. Takagi T. and Sugeno M. [Takagi and Sugeno, 1985] introduced a new type of fuzzy model, referred to as the Takagi Sugeno (TS) fuzzy model, by combining linguistic terms with analytical mathematical functions. This TS fuzzy model is an enhancement of the standard fuzzy model as introduced by Zadeh L.A. [Zadeh, 1968]. Recently, there has been a number of research works where the TS fuzzy model is used to approximate plants. The significant of this fuzzy model relies on the fact that it can facilitate system modeling with the intelligent linguistic fuzzy structure in order to obtain the system model with higher precision and accuracy. This ability of the TS fuzzy model provides a better alternative way in solving the critical weakness in conventional techniques when facing with nonlinear system identification.

In fuzzy systems, the best configuration of the fuzzy system’s parameters in the antecedent and consequent parts are needed to get a more accurate model of the system [Babuska et al, 1998]. The choice of these parameters will affect the accuracy of the model. Takagi T. and Sugeno M. [Takagi and Sugeno, 1985] used least squares estimator to construct a linear system model that can be used for tuning the TS fuzzy model. Passino K.M. and Yurkovich S. [Passino and Yurkovich, 1998] had described the potential of using the least squares method to solve an array of linear equations at the consequent part of fuzzy model as a whole. Gradient descent and backpropagation neural networks are some of the methods used to tune the parameters [Yu and Li, 2004; Nie et al, 1994]. However, these techniques still have some drawbacks such as slow convergence and local minima. In a further effort to improve the previous solutions, a natural evolution based learning method namely genetic algorithm (GA) is proposed to optimize the fuzzy model’s parameters. Historically, the theory and applicability of GA were developed by Holland J.H. [Holland, 1975] and popularized by Goldberg D.E. [Goldberg, 1989]. A GA uses the principle of evolution, natural selection and genetics from natural biological systems in a computer algorithm to simulate evolution. Essentially, GA is an optimization technique that performs a parallel, but directed search to evolve the most fitted solution. The idea of searching a collection of candidate solutions for a desired solution with systematic approach has exploited the ability of GA in solving many complex optimization problems. Setnes M. and Roubos H. [Setnes and Roubus, 2000] used real-coded GA to optimize the fuzzy clustering model. They tested the technique on four examples: a synthetic nonlinear dynamic systems model, the iris data classification problem, the wine data classification problem and dynamic modeling of a diesel engine turbocharger.

For TS type Fuzzy model, some authors implemented a hybrid concept of GA and least square estimation technique to determine the parameter of fuzzy models [Wong and Chen, 2000; Teng and Wang, 2002]. Wong C.C and Chen C.C. [Wong and Chen, 2000] applied the proposed method of binary GA on the models of fifth-order polynomial, two inputs sinc function and inverted pendulum cart. In paper written by Teng Y.W and Wang W.J. [Teng and Wang, 2002], a real-coded GA is used to determine the antecedent part meanwhile batch least square method is used to determine singleton consequent part. They also tested the proposed method on the fifth-order polynomial model.

This paper is the continuation based on Rubiyah et al [Rubiyah et al, 2004; Rubiyah et al, 2005] where the proposed technique is use to estimated the model from the real system. In the paper, we combine the techniques of fuzzy modeling, GA and recursive least square (RLS) to improve system identification of a complex process. We also concentrate on the way of how the fuzzy membership functions can be coded into chromosomes. Our proposed technique provides the GA a wider searching space which can search any possible point in the universe of discourse of the membership functions.

The paper has been organized as follows. In the next section, a description of the process is given followed by the linear model and TS fuzzy model. The RLS approach is the next discussed. The next two sections discuss the GA-RLS-tuned TS fuzzy model approach, the results and discussions. This is then followed by the conclusion.

2. THE PROCESS CONTROL RIG

The specific process plant that is used is the pilot plant process control rig located at Centre of Artificial Intelligent and Robotic (CAIRO), Universiti Teknologi Malaysia (UTM), Kuala Lumpur, Malaysia. The plant is consists of two process; Basic Process Rig (BPR) and Temperature Process Rig (TPR). Figure 1 shows the schematic diagram of process control rig. The primary flow circuit is integral to the TPR, while the secondary flow is supplied from sump tank in BPR through heat exchanger in TPR.
2.1. Overview of the process

In the primary flow circuit, the heating element which is used to heat up the fluid and supplied to the heat exchanger and circulate back to heat element by turn on the pump. Servo valve is used to control the flow from the heating element. This in turn controls the flow rate through the heat exchanger. Besides on the secondary flow circuit, the centrifugal pump will pump out the fluid from sump tank through the heat exchanger to the compartment tank. If the flow from the primary flow circuit increases, more heating fluid flows through the heat exchanger, hence transferring more energy across into the fluid from secondary flow circuit. The overall effect is heating the fluid from the secondary flow circuit. Alternatively, if the flow rate from the primary flow circuit reduced, the overall effect would be less heat energy transferred to the fluid from the secondary flow circuit.

2.2. Nonlinear dynamic model

The dynamic system of the process control rig system can be classified as a multi input multi output (MIMO) system. The system has three manipulated variables and four controlled variables which influence each other as follow:-

**Manipulated variables:**
1. heat supplied from electrical heater, \( Q \)
2. heating element flowrate, \( q_h \)
3. sump tank flowrate, \( q_c \)

**Controlled variables:**
1. temperature of heating element, \( T_{he} \)
2. temperature of hot water, \( T_h \)
3. temperature of cold water, \( T_c \)
4. level of compartment tank, \( h \)

In deriving the principal equations for the whole process control rig system, the modeling of the process may have complex equations and difficult to handle. For that reason, the process model is divided into smaller component or sub-models, where each sub-model is considered to process one output variable. Figure 2 below shows the block diagram of the whole system where it is divided into three sub-systems.

The derivations of time dependent equations are done on every sub-systems of heating element, heat exchanger, and compartment tank. By using the principle of energy balance equation, the mathematical model of heating element and heat exchanger is derived [Lurben, 1990; Geankoplis, 1995]. For compartment tank, the concept of resistance and capacitance is used to describe the dynamic system. The mathematical model of the systems is described as equation 1, 2, 3 and 4.

2.2.1. Heating Element Model

\[
\frac{dT_{he}}{dt} = \frac{q_h}{V_{he}} (T_{he} - T_h) + \frac{Q}{c_p \rho h V_{he}} + \frac{UA}{c_p \rho h V_{he}} (T_{he} - 300)
\]

(1)

where \( T_{he} \) is the temperature of heating element, \( T_h \) is the temperature of hot water, \( q_h \) is the flowrate, \( V_{he} \) the volume, \( A \) the heat transfer surface area, \( c_p \) the specific heat, \( \rho \) the density, \( U \) the heat transfer coefficient, and \( Q \) is the power of heater.

2.2.2. Heat Exchanger Models

\[
\frac{dT_h}{dt} = \frac{q_h}{V_h} (T_{he} - T_h) - \frac{U h A_h}{c_p h V_h} (T_h - T_c)
\]

(2)

\[
\frac{dT_c}{dt} = \frac{q_c}{V_c} (T_{c} - T_c) - \frac{U c A_c}{c_p c V_c} (T_c - T_h)
\]

(3)

where subscripts \( c \) and \( h \) denote the cold and hot side, respectively. \( T_h \) and \( T_c \) are the outlet temperatures of heat exchanger, \( T_{he} \) is the temperature of heating element, \( q_h \) and \( q_c \) are the flowrates, \( V_h \) and \( V_c \) the volumes, \( A \) the heat transfer surface area, \( c_p h \) and \( c_p c \) the specific heats, \( \rho h \) and \( \rho c \) the densities, and \( U \) the heat transfer coefficient.

2.2.3. Compartment Tank Model

\[
\frac{dh}{dt} = \frac{q_c - q_{ot}}{A}
\]

(4)

where \( h \) is the height of fluid in the compartment tank, \( q_c \) is the input flowrate and \( q_{ot} \) is the output flowrate.
flowrate from the compartment tank. The \( q_{ot} \) is considered to be constant value to the process.

The dynamic input-output of the whole model is shown in Figure 3.

3. THE MODEL

In this paper, we have developed linear model and TS fuzzy model of the same process control application to show comparison between both models in term of output’s prediction performance.

3.1 Linear Model

Linear model is perhaps the most basic relationship between the input and the output of the system and usually used in conventional method. For discrete time variant form, the equation for single input single output linear model output can be written as:

\[
y(s+1) = a_1y(s) + \ldots + a_ny(s-n) + a_1u(s)\ldots + a_mu(s-m) + b
\]

where,
- \( s \): one time unit for every sampling interval
- \( u \): input
- \( y \): output
- \( a_1, \ldots, a_n, a_1, \ldots, a_m \): coefficients
- \( b \): noise parameter,
- \( n \): n-th order of output,
- \( m \): m-th order of input.

Thus, the form for linear equation’s output is determined based on the number of inputs. Base on the data acquired from experiments, RLS method is applied in order to automatically identify the coefficients/parameters of linear model.

3.2 TS Fuzzy Model

In this paper, we deal with Takagi and Sugeno’s fuzzy model. The structure of TS fuzzy model consists of three main components: antecedent part, rule base and consequent part. Input variables are represented by membership functions as in standard fuzzy system. In the consequent part, mathematical functions are used instead of membership functions. The structure can be seen as a combination of linguistic and mathematical regression modeling. Figure 4 shows a general block diagram of TS Fuzzy model.
This fuzzy model has rule in the following form:

\[ R_i : \text{if } x_1 \text{ is } A_1 \ \text{and} \ x_2 \text{ is } A_2 \ \text{and} \ldots \ \text{and} \ x_n \text{ is } A_n \ \text{then} \ y_i = f(x) \]  

(6)

where, \( x \) are the vector of input variables \( (x_1,x_2,...,x_n) \), \( A_1,A_2,...,A_n \) are the linguistic terms of input membership function, \( y_i \) is the output of \( i \)-th rule, and \( f(x) \) is the mathematical function of \( i \)-th rule.

For defuzzification method, it is possible to use weighted average method to get the crisp output of the model.

\[ y = \sum_{i=1}^{R} \left( y_i \mu_i \right) / \sum_{i=1}^{R} \mu_i \]  

(9)

where, \( R \) is total rules and \( \mu_i \) is membership value that each \( y_i \) holds for the given input.

In the dynamic modeling, TS fuzzy model can be described as the discrete time variant dynamic system. Typically, ARX model structure is applied to the function \( f(x) \).

\[ y_i(t + 1) = \sum_{k=0}^{q} \theta_{a_k} y_i(t - k) + \sum_{k=0}^{p} \theta_{b_k} x_i(t - k) \]  

(10)

where \( x(t-k) \) and \( y(t-k) \) are the system input and output regressor. \( q \) and \( p \) are integers related to the model order.

### 4. RECURSIVE LEAST SQUARE

In this paper, RLS is used in two forms. The first form is to train linear model’s parameters. Meanwhile, the second form is to train consequent part’s parameters of TS fuzzy model. In this section, we discuss on the standard RLS for linear model development. In similar, with some modification on this standard RLS, it can also be used in the development of TS fuzzy model.

RLS is one member of a family of prediction error identification that is based on the minimization of prediction error functions. Let linear equation

\[ y_i = a_i^T x + b_i \]  

(11)

where, \( a_i \) is a vector of parameter, \( x \) is a vector of inputs variables, \( b_i \) is a scalar offset, \( y_i \) is a output.

RLS method is used to obtain the parameters \( a_i \) and \( b_i \) in equation 11. The non-weighted recursive least square equations is used in this paper are given by

\[ P(s) = P(s-1) - P(s-1) \xi \xi^T P(s-1) \]  

(12)

\[ \theta(s) = \theta(s-1) + P(s) \xi^T (y_i - \hat{y}(s)) \]  

(13)
where $s$ is time index (usually equal to number of data pairs) and $I$ is matrix identity. Initialization of $\theta(s)$ and $P(s)$ are needed in RLS algorithm. The simple approach to do this that often used in practice is to use $\theta(s) = 0$ and $P(s) = \alpha I$ for some large $\alpha > 0$.

Finally, a number of data pairs are trained using RLS algorithm until the parameters, $\theta$ converges to the best values for the output.

Recursive method has an advantage compared to batch method. Taken computer processing limitation, batch least square method cannot operate with a large number of data, meanwhile RLS is able to handle the computation with the same computer power. The recursive version of least square method updates all the coefficients, $\theta$ each time it get a new data pair, without using all the old data in the computation and without having to compute the inverse matrix as well. Hence, processing time using RLS is considered as practical and reliable without compromising on the result performance.

5. GA-RLS-TUNED TS FUZZY MODEL

In this paper, the identification techniques will emphasis in tuning both the parameters of antecedent and consequent parts of fuzzy model. For that purpose, GA is used to fine tune the parameters of membership functions of input variables at the antecedent part. Subsequently, the parameters of linear equations as the mathematical functions in consequent part can be defined using RLS method. This identification technique that utilizes the capability of GA and RLS method can be considered as the technique that optimizes the parameters of TS fuzzy model.

The configuration of membership functions at the antecedent part plays as an important factor to achieve good nonlinear interpolation between linear functions in the consequent part. Using GA, the parameters of membership functions can be automatically optimized. To apply GA in the optimization of membership functions, an appropriate coding and a fitness measure are needed. Figure 5 shows the flowchart of the GA operation.

In particular, the binary GA that we use in this work begins by defining the candidate parameters, the objective function and the fitness function. For instance, the parameters are the values that represent every membership function. The performance of system’s model is usually evaluated based on error calculation between the model’s output and the plant’s output. Thus, the quadratic error or mean squared error is selected to be the objective function in GA operation. After defining the objective function, another function called as fitness function is needed to enable the GA to use the objective function for selecting candidate parameters to be mated. For example, the fuzzy model with the lowest error can be said as the best model.

The next step is the parameter representation. Take note that the GA works with a finite parameter space. Therefore parameter representation is needed to ensure the GA will search the best solution of the problem only in the range set by the designer. Typically, the actual parameters are encoded to the binary representation parameters before performing any GA elements. At the end of every generation, the binary parameters need to be converted back to the actual parameters so that the evaluation process can be performed. In this stage, RLS is used to define parameters of consequent part before evaluation process takes place.

In brief, the GA starts with a community of chromosomes known as the initial population. Taking into account time and computer power factors, population size is heuristically selected not more than 100 chromosomes. Then, the chromosomes of parameters are passed to the objective function and fitness function for evaluation. Among the chromosomes in the population, some of them will be arbitrarily selected. This selection component in the GA guides the algorithm to the solution by preferring individuals with high fitness over low-fitness. In other words, the chromosome with higher fitness function value will have higher chances of being selected. One approach to guide the selection procedure which is used in this work is Roulette wheel scheme.

This reproduction population will then be mated through crossover component. Crossover is the process of creating one or more offspring from the parents. The crossover allows the parents to exchange their information by swapping the genes between the chromosomes. Simple one point crossover method is used in this work.

The last component of the GA is mutation. Mutation can be defined as the random deformation of the chromosome by randomly changing its bit string value. This process is performed to avoid the algorithm from stuck at local minimum by introducing traits not in the original population. Bit by bit mutation method that inverts the randomly selected bit from the population of new offspring, is used in this work.
Define: parameters
   objective function
   fitness function

Represent parameter

Create population

Evaluate fitness for every chromosome

Selection / Reproduction

Crossover

Mutation

New population

Figure 5. Flowchart of a binary GA operation.

Figure 6. An arrangement of membership functions of inputs in a chromosome for the TS fuzzy model.

\[ c_{ij} \]: center for input i and membership function j

\[ w_{ij} \]: width for input i and membership function j

i: 1, 2, ..., N
j: 1, 2, 3

Figure 7. Chromosome contains information of membership functions of all input variables.
It is a fact that the crossover and mutation processes depend on it probability rate. A simple practice is to heuristically choose these values. On the other hand, Seng et al. used a systematic approach by applying dynamic probability of crossover and mutation as they provide faster convergence when compared to constant probability [Seng et al. 1999]. Thus, we implemented the latter approach in our design. Equation 14 and 15 show the calculation of dynamic probability of crossover and mutation, respectively.

$$\text{Crossover rate} = \exp \left( -\frac{N}{M} \right)$$  \hspace{1cm} (14)

$$\text{Mutation rate} = \exp \left( 0.05 \times \frac{N}{M} - 1 \right)$$  \hspace{1cm} (15)

where, $N$ = instant generation
$M$ = maximum generation

A new population is generated at every GA iteration. The iteration can be stopped once the best chromosomes have been found as can be selected as the final solution.

### 5.1. Tuning antecedent part with Genetic Algorithm

Firstly, a fix number of inputs and a fix number of membership functions at antecedent part have to be determined. In this paper, symmetrical Gaussian membership function is chosen which is described by equation 7, where it can be seen that there are two parameters that can be manipulated: the center, $c$ and the width, $w$ of a Gaussian membership function.

Consider $N$ input variables with $F$ membership functions at the antecedent part of TS fuzzy model, the configuration of all membership functions of input variables can be arranged in a chromosome as in Figure 6. From the figure, every sub-chromosome contains information that characterizes the membership functions of an input variable. For each sub-chromosome, the information can be a group of values of the center, $c$ and the width, $w$ of a Gaussian membership functions. A general method to encode all those parameters is directly encoded by the values of $c$ and $w$ for every membership function as they are in the universe of discourse. The advantage of using this independent coding is that the GA will have a larger searching space to find the best possible arrangement of membership functions.

This is true if comparison is made with ‘distance between points’ approach. While ‘distance between points’ approach allows only one overlapping of membership function, our method does not limit it. Through our experimental results, it shown that the best configuration of membership functions is always has more than one overlapping.

If there are $N$ input variables which have three symmetrical Gaussian membership functions for each input, the whole chromosome can be built by combining $N$ sub-chromosomes as in Figure 7.

Another issue that needs to be concerned is the range of values of the center, $c$ and the width, $w$ of membership function. The boundary of the values will ensure the GA to limit the searching process only in the range initially set by the designer. This can be done by heuristically determining the maximum and the minimum values of every $c$ and $w$. For the center, $c$, the values from the minimum and the maximum values of universe of discourse of membership function can be selected. Meanwhile, for the width, $w$, it can arbitrarily choose the width’s range let say from 5% to 30% of the range of universe of discourse. In binary representation, the total length of chromosome can be calculated using equation 16.

$$L^{bin} = \sum_{i=1}^{N} \sum_{j=1}^{F} \left[ \frac{\log_{10} \left( \left( c_{\text{max}} - c_{\text{min}} \right) \times 10^{\text{FP}+1} \right) + \log_{10} \left( w_{\text{max}} - w_{\text{min}} \right) \times 10^{\text{FP}+1} }{ \log_{10} 2 } \right]$$  \hspace{1cm} (16)

$N$ : number of input variables
$F$ : number of membership function
FP: number of floating point

After determining the parameters, specific formula to guide the GA for searching the best fuzzy model has to be assigned. In the case where a set of experimental input-output data pairs is given, it can be utilized to compare the performance of the fuzzy model. The objective function can be developed based on this performance comparison.

As the aim is to minimize the error between the output of fuzzy model and output data, in this paper, the mean of squared error (MSE) is used as a proper evaluation function. MSE is given by

$$MSE = \frac{1}{M} \sum_{i=1}^{M} \left[ y_i - y_{\text{i}} \right]^2$$  \hspace{1cm} (17)

where, $y_i$ is output from fuzzy model, $y_{\text{i}}$ is output data and $M$ is number of data pairs.
Since the objective is to minimize MSE value, the fitness function is defined as follows:

$$J_k = -MSE_k + \max(MSE)$$

(18)

where, k: chromosome k-th

If the normalized fitness function, $f_k(i)$ is used, it can be calculated by the formula:

$$f_k(i) = \frac{J_k(i)}{\sum_{i=1}^{n} J_k(i)}$$

(19)

where, $J_k(i)$ is the fitness function for chromosome i.

The common GA operation can be executed to find the best fuzzy model with combination of RLS method to obtain parameters of consequent part.

5.2. Tuning consequent part with RLS

RLS method is used to obtain the parameters of linear equations in the consequent part and is dependent on the values of the membership functions in antecedent part. In other words, the membership functions really play an important factor to achieve good nonlinear interpolation between linear functions that are constructed with RLS. This is quite different from linear model identification where the parameters of the model can be directly calculated from input variable values, whereas the parameters of the linear equations of fuzzy model can only be identified indirectly from the values of input variables and the membership for each rule.

Thus, before implementing the RLS method in fuzzy model identification, the equation applied need to be rearranged to find the output so that the equation can comply with the RLS equation. Recall TS Fuzzy model output in equation 8 and 9.

Define

$$\bar{\xi}_i = (\mu_i) / (\sum_{i=1}^{R} \mu_i)$$

(20)

where $\mu_i$ is the membership value for every i-th rule. Then equation 8 will be

$$y = \sum_{i=1}^{R} y_i \bar{\xi}_i$$

(21)

Hence, combining equation 7 and 21,

$$y = \sum_{i=1}^{R} (a_i^T x + b_i) \bar{\xi}_i$$

(22)

Let

$$\xi(x) = [x, \xi_1, ..., x_t, \xi_R, x_{h}, \xi_1, ..., x_{c}, \xi_R, \xi_1, ..., \xi_R]^T$$

(23)

and the coefficients of all equations is combined

$$\theta = [a_{1,0}, ..., a_{R,0}, a_{1,n}, ..., a_{R,n}, b_1, ..., b_R]^T$$

(24)

then, the output of Fuzzy model can be rearranged as follow

$$y = \theta^T \xi(x)$$

(25)

Equation 25 is the final arrangement of output’s calculation originally derived from equation 7. This equation is in exactly the right form for use with RLS method since $\xi(x)$ is known as regression vector. Roughly, the input data are mapped into $\xi(x)$ using inference mechanism and the RLS algorithm produces an estimate of the best coefficients, $\theta$.

6. RESULTS AND DISCUSSIONS

In order to obtain a model for each process, the input and output data of the system are needed for training and evaluation purpose. The data is obtained from the experiments done on the real process plant. For that purpose, we have developed the graphic user interface (GUI) software for data acquisition and modeling. We can control all the actuators and acquire all the data from the sensors using the GUI software from the process plant.

Before we start the experiments, we set the input of the systems, Q, qh and qc as a random value. The heat, Q is controlled by the heater where the heater is control by using pulse width modulation (PWM) signal. Meanwhile, the flows, qh and qc are controlled by the servo valve where the servo valve is control by using pseudorandom binary sequence signal (PRBS).

554 input-output data pairs are taken for 2 sets of experiments. The first set of experiment is used for training purpose and another set is used for evaluation purpose. The sampling time is set at 10 seconds. The mathematical model of the systems as described in equation 1, 2, 3 and 4, is used as a reference to represent the actual plant.

Then, the data obtained is used as input to fuzzy model. For fuzzy logic structure, number of fuzzy sets is 5 for each input, and the number of rules is 5.
Meanwhile, for GA structure, number of generation is 200, and population size is 20.

Based on the dynamic input-output model in Figure 3, the inputs of the model consist of 2 manipulated variables: heat supply (Q), heating element flowrate \( q_h \) and 2 regressor: temperature of hot water from heat exchanger \( T_h \), temperature of heating element \( T_{he} \). The selection of input variables is based on physical knowledge of the process plant.

Every input variable is represented by 5 Gaussian fuzzy sets: VERY LOW (VL), LOW (L), MEDIUM (M), HIGH (H) and VERY HIGH (VH). A small number of fuzzy set in Takagi-Sugeno type is adequate to appropriate the model.

### 6.1 Heating Element Model, \( T_{he} \)

The RLS model of \( T_{he} \) is obtained as below:

\[
T_{he}'(t+1) = -0.0001Q(t) + 0.0071q_h(t) \cdot -0.0249T_h(t) + 1.0082T_{he}'(t) + 0.8244
\]

Figure 8, 10, 12 and 14 shows the membership function of every input of each sub-system. The parameters of membership function are automatically tuned by GA. Meanwhile, the best configuration of fuzzy models with GA had shown in Figure 9, 11, 13, and 15.

### 6.2 Heat Exchanger Model – Hot Water, \( T_h \)

The RLS model of \( T_h \) is obtained below:

\[
T_h'(t+1) = 0.0363q_h(t) + 0.1278T_{he}'(t) - 0.08127T_h(t) + 0.9125T_h(t) + 0.4611
\]
6.3 Heat Exchanger Model – Cold Water, $T_c$

The RLS model of $T_c$ is obtained as below:

$$T_c(t+1) = 0.0091q_c(t) + 0.0856T_c(t) + 0.8974T_h(t) + 0.0972$$

6.4 Compartment Tank Model, $h$

The RLS model of $h$ is obtained as below:

$$h(t+1) = 0.0764q_c(t) + 0.9828h(t) - 0.2339$$

The effectiveness of the fuzzy model with GA is tested using evaluation data. The fuzzy model with GA is compared to the RLS model with the value of mean squared error (MSE) as performance index for each model as shown in Table 1.
The results have shown that fuzzy model with GA give better mean squared error when compared with RLS model.

Fuzzy with GA system not only have powerful approximation abilities for modeling unknown dynamic non-linear system, but the parameters of membership function of the model can also be obtained. Therefore, we can say that fuzzy model can solve the nonlinear, complex and uncertain process. The model is more similar with the real process from the value of mean squared error obtained.

7. CONCLUSION

Although fuzzy control has been proven to be effective in control applications, the same is not true in fuzzy modeling. Fuzzy modeling in process control applications lack many other techniques in terms of applications and innovations. In this paper, modeling of a real process control rig using fuzzy modeling tuned by GA with a combination of the RLS has been proposed. A TS fuzzy model with GA as an automatic tuner for the antecedent fuzzy parameters with the conventional RLS approach for the consequent part shows better modeling results when compared with just the conventional RLS modeling approach applied on a process control rig. The results have shown that TS fuzzy model with GA and RLS can effectively minimize the error between fuzzy output and real data output by optimizing the parameters of fuzzy model.

8. REFERENCES


AUTHOR BIOGRAPHIES:

Ribhan Zafira Abdul Rahman received the B.E. degree in Electrical and Electronics Engineering from the Liverpool John Moores University, United Kingdom in 1998, and the M.E. degree in Electrical Engineering from Universiti Teknologi Malaysia, Malaysia in 2001. Her research interest includes artificial intelligence, process control, control systems and fault detection and diagnosis. Presently she is a lecturer at Universiti Putra Malaysia, Malaysia. She is currently pursuing her PhD degree in fault detection and diagnosis at Universiti Teknologi Malaysia.

Rubiyah Yusof is a Professor at the Faculty of Electrical Engineering, Universiti Teknologi Malaysia, Malaysia (UTM). She is currently the Deputy Director for the Business and Advanced Training Centre (BATC), UTM. Her interests include adaptive control, system identification, biometric systems and ICT. She also is a co-author of the book entitled “Neuro Control and its Applications” published by Springer, United Kingdom.

Marzuki Khalid is a Professor in Intelligent Control at the Faculty of Electrical Engineering, Universiti Teknologi Malaysia, Malaysia. He is currently the Director for the Centre of Artificial Intelligence and Robotics (CAIRO). He is also the Director of the Malaysia-Japan University Center by the Ministry of Higher Education. His research interest is in the field of artificial intelligence, control systems and image processing. He is currently on the editorial boards of several international journals.

Mohd Faisal Ibrahim received the B.Eng and M.Sc degrees in Electrical and Electronics Engineering from the Universiti Teknologi Malaysia, Malaysia in 2002 and 2005, respectively. Currently, he is a lecturer at Universiti Kebangsaan Malaysia (UKM), Malaysia. His research interest is in artificial intelligence and network security. Prior to joint UKM, he has 2 years industrial experience working with commercial-based R&D Company, MIMOS.