ANFIS modelling of a twin rotor system using particle swarm optimisation and RLS

S. F. Toha, *Student member, IEEE* and M. O. Tokhi, *Senior member, IEEE*
Department of Automatic Control and Systems Engineering,
The University of Sheffield,
Mappin Street, Sheffield, S1 3JD, United Kingdom.
Email: cop06sft@sheffield.ac.uk

***Abstract*— Artificial intelligence techniques, such as neural networks and fuzzy logic have shown promising results for modelling of nonlinear systems whilst traditional approaches are rather insufficient due to difficulty in modelling of highly nonlinear components in the system. A laboratory set-up that resembles the behaviour of a helicopter, namely twin rotor multi-input multi-output system (TRMS) is used as an experimental rig in this research. An adaptive neuro-fuzzy inference system (ANFIS) tuned by particle swarm optimization (PSO) algorithm is developed in search for non-parametric model for the TRMS. The antecedent parameters of the ANFIS are optimized by a PSO algorithm and the consequent parameters are updated using recursive least squares (RLS). The results show that the proposed technique has better convergence and better performance in modeling of a nonlinear process. The identified model is justified and validated in both time domain and frequency domain**

Keywords; Twin rotor system, adaptive neuro-fuzzy inference system , particle swarm optimisation, recursive least squares.

I. INTRODUCTION

A great deal of progress has been made in the modelling of aerodynamic rotors over the past decade [1], [2], [3], [4]. Although the modelling effort has focused on aircraft rotors, the theory is generally valid for a wide range of rotor configurations. Unlike conventional fixed-wing aircraft, helicopter portrays distinct advantages in surveillance and inspection tasks as they can take off and land vertically in limited spaces and easily hover in places above the target. However, helicopters are much more complex in terms of system dynamics and control because the inputs are not directly applied torques or forces, but rather aerodynamic torques or forces by the main and tail rotors albeit all the aforementioned advantages.

A scaled and simplified version of practical helicopter namely a twin rotor multi-input multi-output system (TRMS) is used in this work. Although the dynamics of the TRMS are simpler than those of a real helicopter, they retain the most important helicopter features such as couplings and strong nonlinearities. It can be perceived as an unconventional and complex “air vehicle”. Therefore, it is crucial to deduce a good model of the TRMS as attempted in this work.

System identification is concerned with choosing mathematical models to characterize the input–output behaviour of an unknown system using observed data. In this work, a hybrid neuro-fuzzy system called adaptive neuro fuzzy inference system (ANFIS) is used. Fuzzy logic and neural networks are natural complementary tools in building intelligent systems. While neural networks are low-level computational structures that perform well when dealing with raw data, fuzzy logic deals with reasoning on a higher level, using linguistic information acquired from domain experts. However, fuzzy systems lack the ability to learn and cannot adjust themselves to a new environment. On the other hand, although neural networks can learn, they are opaque to the user. The merger of a neural network with a fuzzy system into one integrated system therefore offers a promising approach to building intelligent systems. Integrated neuro-fuzzy systems can combine the parallel computation and learning abilities of neural networks with the human-like knowledge representation and explanation abilities of fuzzy systems. As a result, neural networks become more transparent, while fuzzy systems become capable of learning [5].

The usual way of modelling a system using ANFIS is by identifying the number of inputs and the number of membership function using trial and error as well as expert knowledge [6]. There are some drawbacks by doing so [7]: first, the computational efficiency associated with fuzzy logic is lost using a high number of rules and second, the robustness decreases with increasing the number of rules. This is especially true when the dimension of the inputs and the number of fuzzy sets for each input variable become great since the number of possible rules exponentially increases with these numbers. Therefore, the ANFIS system can be optimised by adapting the antecedent parameters and consequent parameters so that a specified objective function is minimised. Few efforts have been devised using genetic algorithm (GA) and particle swarm optimisation (PSO). Lin [8] addressed a genetic algorithm-based neural fuzzy system (GA-NFS) based on Takagi–Sugeno–Kang (TSK) type model possessing a neural network’s learning ability for temperature control. A hybrid learning algorithm is proposed for parameters learning. The proposed algorithm combines the genetic algorithm (GA) and the least-squares estimate (LSE) method to construct the GA-NFS. Abdul Rahman et al. [9] on the other hand proposed the use of an automatic modified GA to tune the adaptive neuro-fuzzy modelling technique of a plant process control. The consequent part is then tuned using RLS. Juang and Wang [10] used a combination of ant and particle swarm cooperative optimisation to generate fuzzy system using on-line clustering method.

Although GA can provide good solutions in modelling applications, it however requires huge memory and faster processing units with large word lengths to execute huge number of repeated computations. Moreover, for highly multi-modal problems, the solutions may lose diversity and get trapped in local minima at some points unless special method is adopted to avoid premature convergence to suboptimal region of the search space [11]. Therefore, in this work, particle swarm optimisation PSO is employ to train the antecedent parameters and RLS is used to optimise the consequent parameters of a fuzzy inference system. The proposed methodology is used to identify the non-parametric model of the twin rotor system.

II. Twin Rotor MIMO System

The twin-rotor multiple-input multiple-output (MIMO) system (TRMS) is a laboratory set-up developed by Feedback Instruments Limited [12] for control experiments. Its behaviour in certain aspects resembles that of a helicopter. For example, it possesses a strong cross-coupling between the collective (main rotor) and the tail rotor, like a helicopter. A schematic diagram of the TRMS used in this work is shown in Figure 1. It is driven by two DC motors. Its two propellers are perpendicular to each other and joined by a beam pivoted on its base that can rotate freely in the horizontal and vertical planes. The beam can thus be moved by changing the input voltage in order to control the rotational speed of the propellers. The system is equipped with a pendulum counterweight hanging from the beam, which is used for balancing the angular momentum.

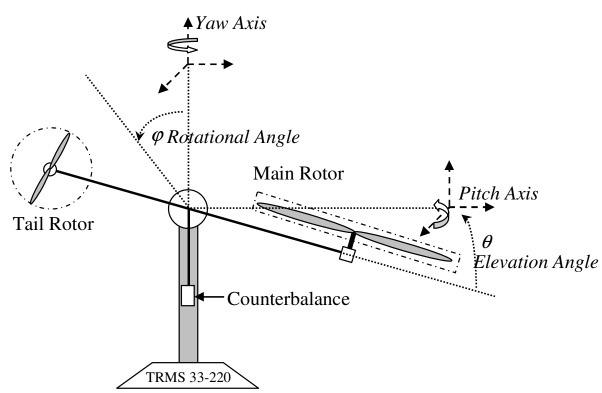

Figure 1. Twin rotor multi input multi output system

The system is balanced in such a way that when the motors are switched off, the main rotor end of the beam is lowered. The controls of the system are the supply voltages of the motors. It is important to note that the geometrical shapes of the propellers are not symmetric. Accordingly, the system behaviour in one direction is different from that in the other direction. Rotation of a propeller produces an angular momentum which, according to the law of conservation of angular momentum, is compensated by the remaining body of the TRMS beam. This results in interaction between the moment of inertia of the motors with propellers. This interaction directly influences the velocities of the beam in both planes. The measured signals are: position of the beam, which constitutes two position angles, and the angular velocities of the rotors.

III. Adaptive neuro-fuzzy inference system (ANFIS)

In this work, a Takagi-Sugeno-Kang type fuzzy model is used. This compares two main components, the antecedent and consequent parts. For simplicity, the ANFIS structure is set to have 2 inputs x_1 and x_2 as shown in Figure 2 [13].

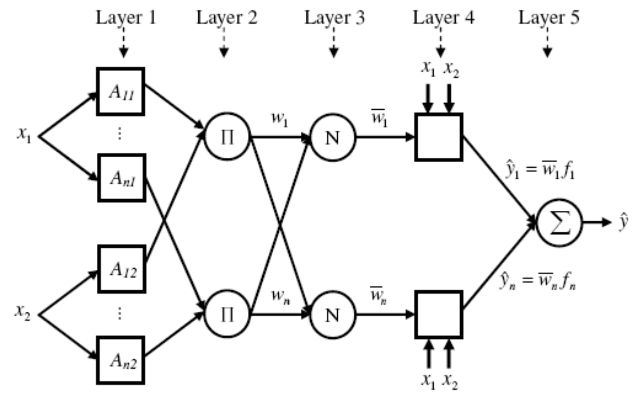

Figure 2. Adaptive neuro-fuzzy inference system

This fuzzy model has rules in the following form:

R_i : **If** x_1 is A_{i1} **And** x_2 is A_{i2} **Then** y_i is $f_i(x)$ (1)

where x_1 and x_2 are the input variables to the ANFIS. $A_{i1},\ldots, A_{in}$ are the linguistic terms of input membership function for the ith rule ($i = 1, 2, \ldots, n$) and y_i is the consequent part which is the mathematical function of ith rule. Fuzzy set A_{ij} at layer 1 uses a Gaussian membership function for each input variable and it has the form

$$A_{ij}(x) = \exp\left\{-\left(\frac{x_j - m_{ij}}{\sigma_{ij}}\right)^2\right\} \quad (2)$$

where m_{ij} and σ_{ij} represent the centre and width of the fuzzy set A_{ij} respectively. Parameters in this layer are referred to as *antecedent parameters*. The output of a fuzzy inference system with n rules is obtained by weighting the real values of consequent parts of all rules with the corresponding membership grade

$$\hat{y} = \sum_{i=1}^{n} \overline{w}_i f_i = \frac{w_i}{\sum_{i=1}^{n} w_i} \quad (3)$$

where

$$w_i = \prod_{j=1}^{n} A_{ij}(x_i) \text{ where input are } x_1 \text{ and } x_2 \quad (4)$$

and

$$y_i = f_i(x) = (a_i x_1 + b_i x_2 + c_i) \quad (5)$$

where $\{a_i, b_i, c_i\}$ is the parameter set and it is referred to as *consequent parameters*.

IV. PARTICLE SWARM OPTIMISATION

Particle swarm optimization is a stochastic, population based swarm intelligence algorithm developed by Kennedy and Eberhart [14]. The PSO algorithm is similar to evolutionary computation in producing a random population initially and generating the next population based on current cost, but it does not need reproduction or mutation to produce the next generation. Thus, PSO is faster in finding solutions compared to other evolutionary computation techniques.

Dynamic spread factor PSO (SFPSO) is employed in this paper [15]. The algorithm is found highly effective in improving major issues in basic PSO that are premature convergence and preservation of diversity. As originally developed, the inertia weight, w is decreased linearly from 0.9 to 0.4 during a run. Suitable selection of the inertia weight provides a balance between global and local exploration and exploitation and results in less iteration on average to find a sufficiently optimal solution. The mathematical representation of SFPSO is given as

$$\begin{aligned} x_{id_{new}} &= x_{id} + v_{id_{new}} \\ v_{id_{new}} &= (w * v_{id}) + c_1(rand_1(p_{id} - x_{id})) \\ &\quad + c_2(rand_2(p_{gd} - x_{id})) \end{aligned} \quad (6)$$

where w, c_1 and c_2 are given by

$$\begin{aligned} w &= \exp(-iter/(spread\_factor \times \max\_iteration)) \\ spread\_factor &= 0.5(spread + deviation) \\ c_1 &= 2(1 - iter/\max\_iteration) \\ c_2 &= 2 \end{aligned} \quad (7)$$

where x_{id} and v_{id} represent the position vector and velocity vector of the ith particle in the d-dimensional search space respectively. The first part of velocity vector in equation (6) represents the previous velocity, which provides the necessary momentum for particles to roam across the search space. The second part, known as 'cognitive' component, represents the personal thinking of each particle. The cognitive component encourages the particles to move towards their own best positions found so far. The third part is known as the 'social' component, which represents the collaborative effect of the particles, in finding the global optimal solution. The social component always pulls the particles towards the global best particle found so far. In order for particles to keep exploring the search space, it is imperative that they must know their whereabouts and relative distances from each other. The spread factor in SFPSO algorithm measures the distribution of particles in the search space as well as the precision and accuracy of the particles with respect to global optimum. Therefore, when all the particles move within the vicinity of global optimum, both the dynamic SF and hence the inertia weight will drop in value drastically. This will not only force all the particles to converge, but also allow the algorithm to achieve extremely high precision.

Throughout this work, SFPSO is used in search for parameter estimation of centre and width in (2). The SFPSO process is best explained in Figure 3.

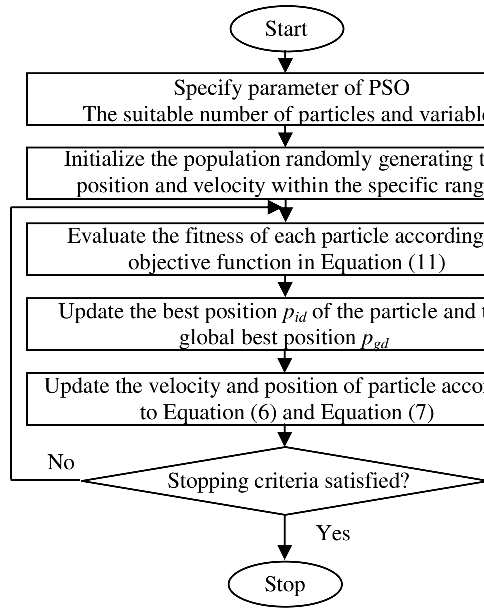

Figure 3. The flowchart of SFPSO algorithm

V. TRMS MODELLING USING ANFIS

An ANFIS consisting of a set of Takagi-Sugeno-Kang (TSK)-type fuzzy IF-THEN rules is used to map the system inputs to outputs. This hybrid combination enables to deal with both the verbal and the numeric power of intelligent systems. As is known from the theory of fuzzy systems, different fuzzification and defuzzification mechanisms with different rule base structures can lead to various solutions to a given task. The fuzzy regions are parameterized and each region is associated with a linear subsystem. Owing to the fuzzily defined antecedents, the nonlinear system forms a collection of loosely coupled multiple linear models. The degree of firing of each rule is proportional to the level of contribution of the corresponding linear model to the overall output of the model.

In this work, ANFIS is used to obtain non-parametric model of the system in hovering position, based on input-output data of the TRMS. The input data structure comprises the voltage of main rotor at previous time, $V_v(t-1)$ and the pitch angle of the beam at previous time, $\alpha_v(t-1)$. The output from the ANFIS non-parametric model is the predicted pitch angle of the beam $\hat{\alpha}_v(t)$ at the hovering motion. The membership function for both the inputs is set to be six. The ANFIS structure containing 36 rules is shown in Figure 4 [6].

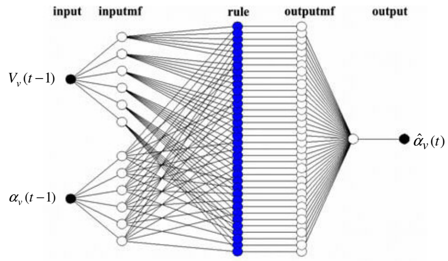

Figure 4. ANFIS structure for modelling the TRMS

VI. LEARNING ALGORITHM FOR ANFIS MODEL

The learning algorithms used in this work are PSO and RLS where the parameters of both antecedent part m_{ij} and σ_{ij}, and the consequent part $\{a_i, b_i, c_i\}$ are optimised. The adaptation of fuzzy inference system usually will rely on the back propagation method. This conventional technique is based on gradient descent algorithm is susceptible to get stuck at local minima. Therefore, SFPSO is used to overcome this problem and hence provide a faster convergence rate. Furthermore, RLS is used to tune the consequent part since it has advantages compared to least square [9]. Taking computer processing limitation into account, least square method cannot operate with a large number of data, while RLS is able to handle the computation with the same computer power. The recursive version of least square method updates all the coefficients, θ each time it gets 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 resulting performance.

A. *RLS for forward pass tuning*

In this section, RLS is used to solve the consequent parameters in (5). From (3) it is known that

$$\overline{w}_1 f_1 + \overline{w}_2 f_2 + \ldots + \overline{w}_n f_n = \hat{y}_1 + \hat{y}_2 + \ldots + \hat{y}_n$$
$$\begin{bmatrix} \overline{w}_1 x_1 & \overline{w}_1 x_2 & \overline{w}_1 \\ \overline{w}_2 x_1 & \overline{w}_2 x_2 & \overline{w}_2 \\ \vdots & \vdots & \vdots \\ \overline{w}_n x_1 & \overline{w}_n x_1 & \overline{w}_n x_1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_n \end{bmatrix} \tag{8}$$

where

$$\varepsilon(t) = y(t) - \varphi^T(t)\theta(t-1) \tag{9}$$

thus, the non-weighted RLS is used in this work to estimate the parameters $\{a_i, b_i, c_i\}$ are given by

$$P(t) = P(t-1) - P(t-1)\varphi(t)(I + \varphi^T(t)P(t-1)\varphi(t))^{-1}$$
$$\varphi^T(t)P(t-1) \tag{10}$$
$$\theta(t) = \theta(t-1) + P(t)\varphi(t)(y(t) - \varphi^T(t)\theta(t-1))$$

During the forward pass learning, the consequent parameters are estimated using RLS until the parameters θ converge to produce the best values for the output.

B. *SFPSO for backward pass tuning*

In this training procedure, the SFPSO algorithm is used to optimise the antecedent parameters while the consequent parameters remain fixed. A Gaussian membership function is used with variables m_{ij} and σ_{ij}, representing the centre and width of the membership function according to the fuzzy rule in (1). A swarm with 40 particles are chosen for this work, whereby, the number of variables will depends on the number of membership function for each input. 2 input x_1 and x_2 are used with *i*th membership functions ($i = 1,2,\ldots,6$) are used to obtain the number of rules that best suit to model the TRMS in hovering position. The coding of antecedent parameters in each particle is shown in Figure 5.

c^{i1}	w^{i1}	c^{i2}	w^{i2}	....	c^{n1}	w^{n1}	c^{n2}	w^{n2}

c^{ij} = centre for membership function, *i* and input, *j*

w^{ij} = width for membership function, *i* and input, *j*

i=1,2,…,n *j=1,2*

Figure 5. Particles contains information of membership function for all input variables

Therefore, the number of variables in each particle will represent a real-value of possible antecedent parameters. Each particle will be decoded into an objective function value. A fitness value is assigned to each particle in the swarm, where small values mean good fit. The objective function used in this work is defined by a mean squared error function (MSE)

$$f(x) = \frac{1}{s}\sum_{t=1}^{s}(y(t) - \hat{y}(t))^2 \tag{11}$$

where *y(t)* is the output from the system, $\hat{y}(t)$ is the predicted output from fuzzy model, and s is the number of training data pairs which is 700.

VII. RESULTS AND DISCUSSION

This section presents the results of both ANFIS and ANFIS-PSO modelling. To study variations in the detected vibration modes, modelling was carried out with the twin rotor system response to a pseudorandom binary sequence (PRBS) input. This type of input is chosen to ensure that the system is well-excited over the dynamic range of interest.

The adaptive network-based fuzzy inference system (ANFIS) is used for the non-parametric modelling of the twin rotor system (Figure 2). The input data structure comprises the voltage of main rotor at previous time, $V_v(t-1)$ and the pitch angle of the beam at previous time, $\alpha_v(t-1)$. The output from the ANFIS non-parametric model is the predicted pitch angle of the beam $\hat{\alpha}_v(t)$ at the hovering motion. 1000 pairs of input-output data is extracted from the real system where 700 data points are used as training data, and the remaining 300 data as testing data. The results of using a hybrid combination of SFPSO and RLS for the ANFIS modelling as compared with using 36 rules of ANFIS model [6] are shown in Table 1.

TABLE I. COMPARISON RESULTS OF ANFIS MODELLING

Method	Dataset	Mean-squared Error	Execution Time (s)
ANFIS (36 rules)	Training (700 data)	1.220×10^{-4}	31.943
	Checking (300 data)	4.211×10^{-4}	31.851
ANFIS PSO + RLS (16 rules)	Training (700 data)	2.407×10^{-5}	12.325
	Checking (300 data)	3.598×10^{-5}	12.007

Figure 6 shows the initial ANFIS structure before optimisation using 6 membership functions at each input with 36 number of rules. The final ANFIS structure after PSO and RLS optimisation is shown in Figure 7. The training algorithm changed shape and shifted the location of the Gaussian membership functions to reduce the ANFIS estimation error. Figure 8 shows the result of modeling the TRMS with an ANFIS after using PSO and RLS optimization. In this case only 16 fuzzy rules were sufficient to effectively model the 700 training data set from the two inputs of $V_v(t-1)$ and $\alpha_v(t-1)$ of the TRMS. The corresponding MSE (Figure 9) obtained by using the PSO optimization was 2.407×10^{-5} which is considerably smaller than the value obtained previously with a larger number of rules. The frequency domain plot (Figure 10) of the predicted and actual outputs indicates that the model has successfully captured the system dynamics surrounding the main resonance mode which is at 0.3497 Hz.

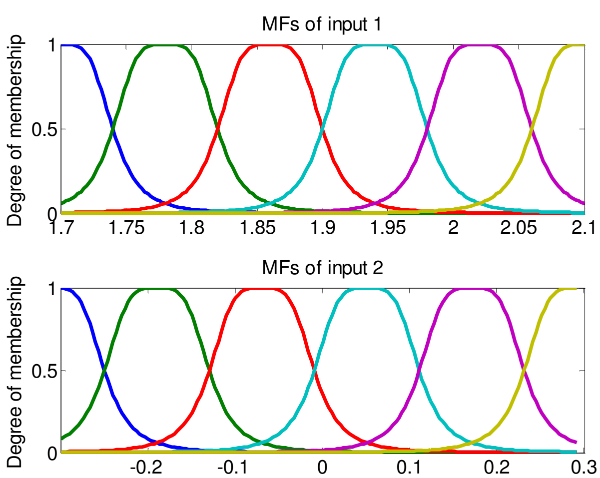

Figure 6. ANFIS membership function using 36 rules

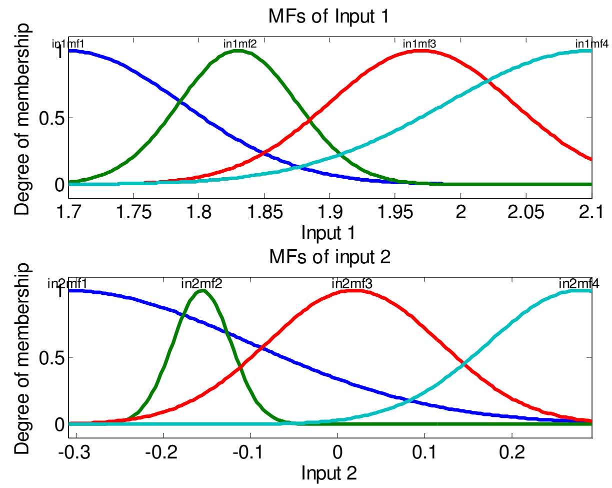

Figure 7. ANFIS membership function using 16 rules

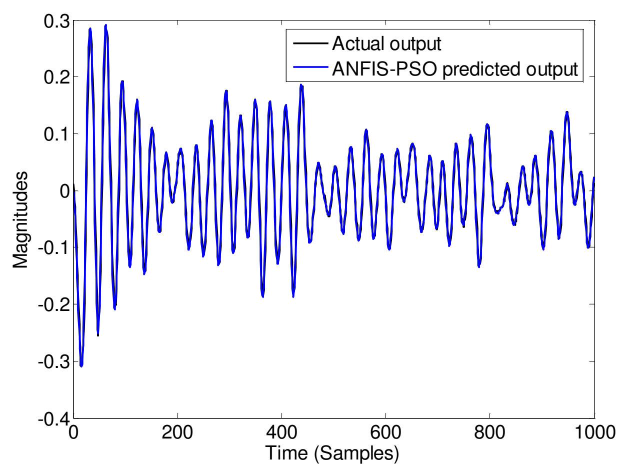

Figure 8. Actual and ANFIS-PSO predicted output of the system

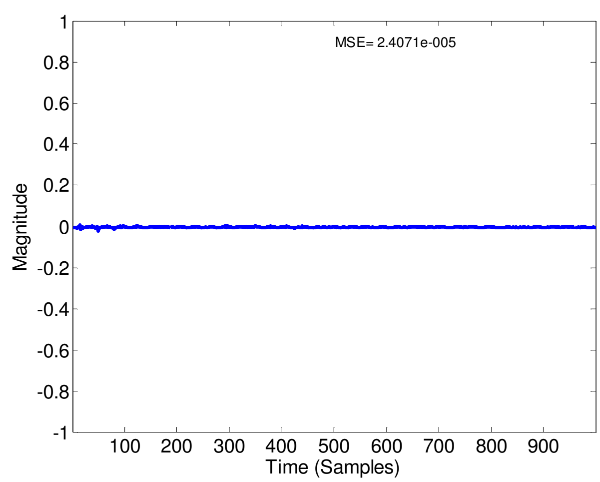

Figure 9. Error between actual and ANFIS-PSO predicted output

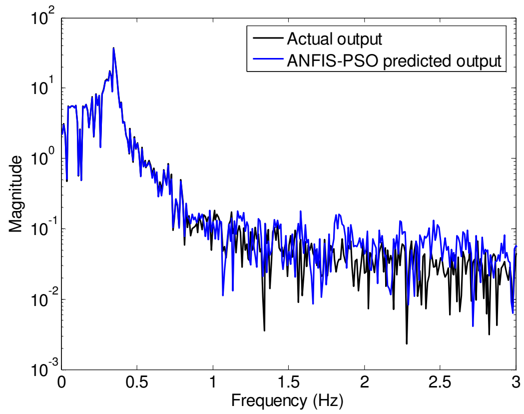

Figure 10. Power spectral density of actual and ANFIS-PSO predicted output

Correlation validation of vertical plane motion model is shown in Figures 11 and 12. It is noted the correlation functions are within the 95% confidence bands indicating that the model behaviour was unbiased and close to that of the real system. As a conclusion, it has been demonstrated that the methodology of using PSO and RLS for optimization in of ANFIS modelling is remarkably effective in terms of both prediction error and reduced number of fuzzy rules.

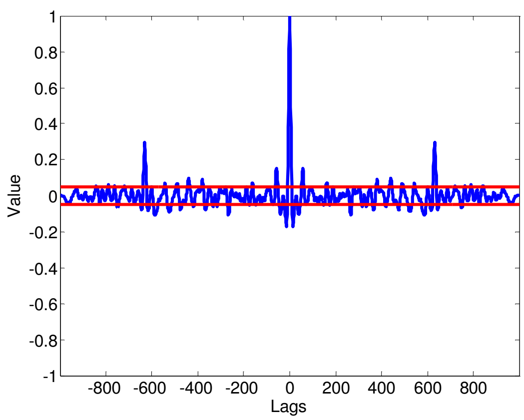

Figure 11. Auto-correlation of residual

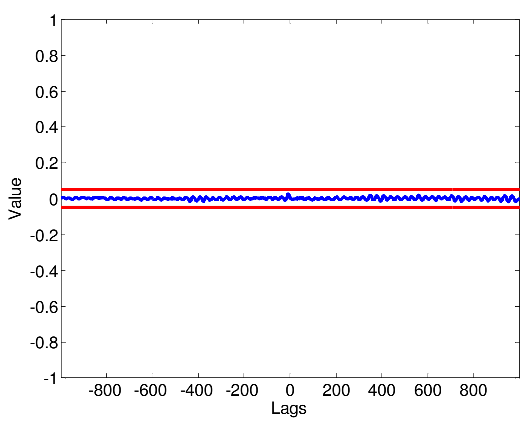

Figure 12. Cross-correlation of input and residual

VIII. CONCLUSION

In this work, an approach of non-parametric modelling has been developed and applied to a TRMS in simulation environment in 1 DOF hovering motion. A TRMS has been modelled with ANFIS-PSO modelling technique with a reduced number of rules in the membership function. Simulations demonstrated that the proposed ANFIS-PSO model has good generalization capability and robustness. A novel method for resolving the problem between number of rules and number of consequent terms has been constructed. Hence, it can optimize the number of generated rules and membership functions for modelling a sophisticated system.

REFERENCES

[1] Aleyaasin, M. and Ebrahimi, M., Hybrid modelling for analysis and identification of rotors. *Computer Methods in Applied Mechanics and Engineering*, **182**(1-2): p. 163-176 (2000).

[2] Pesheck, E., Pierre, C., and Shaw, S.W., Accurate reduced-order models for a simple rotor blade model using nonlinear normal modes. *Mathematical and Computer Modelling*, **33**(10-11): p. 1085-1097 (2001).

[3] Yu, C., Zhu, J., Hu, J., and Sun, Z., Experimental modeling using modified cascade correlation RBF networks for a four DOF tilt rotor aircraft platform. *Neurocomputing*, **69**(13-15): p. 1802-1805 (2006).

[4] Xiong, J.J. and Yu, X., Helicopter rotor-fuselage aeroelasticity modeling and solution using the partition-iteration method. *Journal of Sound and Vibration*, **302**(4-5): p. 821-840 (2007).

[5] Negnevitsky, M., *Artificial intelligence: a guide to intelligent systems*. Second ed: Addison Wesley, Pearson education (2005).

[6] Toha, S.F., Tokhi, M.O., and Hussain, Z. ANFIS modeling of a twin rotor system. *UKACC International Conference on Control 2008 (UKACC08)*. Manchester, UK. p. P125.pdf (2008).

[7] Yuhui, S., Eberhart, R., and Yaobin, C., Implementation of evolutionary fuzzy systems. *IEEE Transactions on Fuzzy Systems*, **7**(2): p. 109-119 (1999).

[8] Lin, C.-J., A GA-based neural fuzzy system for temperature control. *Fuzzy Sets and Systems*, **143**(2): p. 311-333 (2004).

[9] Abdul Rahman, R.Z., Yusof, R., Khalid, M., and Ibrahim, M.F., Fuzzy modelling using genetic algorithm and recursive least squares for process control application. *International Journal of Simulation Systems, Science and Technology*, **9**(3): p. 34-46 (2008).

[10] Juang, C.-F. and Wang, C.-Y., A self-generating fuzzy system with ant and particle swarm cooperative optimization. *Expert Systems with Applications*, **36**(3, Part 1): p. 5362-5370 (2009).

[11] Alam, M. and Tokhi, M., Modelling of a twin rotor system: a particle swarm optimization approach. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, **221**(3): p. 353-374 (2007).

[12] Feedback Instrument Ltd, *Twin Rotor MIMO System manual 33-007-0*, Sussex, UK (1996).

[13] Jang, J.S.R., ANFIS: adaptive-network-based fuzzy inference system. *Systems, Man and Cybernetics, IEEE Transactions on*, **23**(3): p. 665-685 (1993).

[14] Kennedy, J. and Eberhart, R. Particle swarm optimization. *IEEE International Conference on Neural Networks*. p. 1942-1948 vol.4 (1995).

[15] Abd Latiff, I. and Tokhi, M.O. Fast convergence strategy for particle swarm optimization using spreading factor. *IEEE Congress on Evolutionary Computation*. Trondheim, Norway. p. 2693-2700 (2009).