Hybrid Neuro-Predictive-Fuzzy Algorithm for a Model Helicopter's Yaw Angle Control

Morteza Mohammadzaheri and Lei Chen *

Abstract—**In this paper, a Neuro-Predictive (NP) controller was first designed and implemented on a highly non-linear system, a model helicopter in a constrained situation. It was observed that the closed loop system with the NP controller had a significant overshoot and a long settling time in comparison to the same system with an fuzzy controller. In order to improve the system's performance, a Sugeno-type fuzzy compensator, having only two rules, is added to the control loop to adjust control input. The newly designed Neuro-Predictive control with the Fuzzy Compensator (NPFC) improves the system's performance. Furthermore, it is shown that the NPFC controlled system is robust to disturbance and system parameter changes.**

Keywords: ***Neuro-Predictive, Fuzzy Control, Model Helicopter, Overshoot***

1 Introduction

Predictive control, as a method of using current and predicted output to determine control input, was initially introduced by Camacho in the classical Model Predictive Controll (MPC) [1]. It is known that for *prediction*, a linear *model* is needed in the classical MPC. Quite often linear state-space models are used. Such models can be used to predict the behaviour of many systems [2]. When a general linear model is not available, Artificial Neural Network (ANN) method can be used to obtain piecewise linear models through recorded data under certain conditions. Such ANN models can be used in the classical predictive control [3] as well. But naturally, nonlinear models are more suitable for nonlinear systems in order to predict a wide range behaviour of the nonlinear systems. For example, Soloway and Haley used a nonlinear ANN as a model for predictive control purposes [4]. However using the classical MPC method to derive the control input is not applicable any more when nonlinear models are implemented. In order to derive the control input in the presence of nonlinear ANN models, nonlinear optimisation methods are often used [5, 6, 7], or an additional ANN should be implemented [8]. Neuro-predictive controllers have been implemented in a variety of applications such as the control of food or chemical processes and the control of air/fuel ratio of engines [9, 10, 11]. This technique has also been used to control a hybrid water and power supply [12] and a 6-DOF robot [13]. In biomedical engineering, neuro-predictive controllers are used to control insulin pump of diabetic patients [14]. In this research, a neuro-predictive approach is used to control a model helicopters yaw movement. A fuzzy inference system is also designed as a compensator or brake to improve the effectiveness of the neuro-predictive controller.

*Both authors are with the School of Mechanical Engineering, The University of Adelaide, South Australia, Tel/Fax: +61 8 8303 5469/3476 Email: ley@mecheng.adelaide.edu.au, morteza@mecheng.adelaide.edu.au

2 Control Method

The designed hybrid controller includes three main parts: an ANN to predict the behaviour of system, a *nonlinear optimisation method* to minimise the performance function, and a *fuzzy inference system* to improve the effectiveness.

The model helicopter under consideration is a dynamic inertial system. In such a system, the current value of the system's output is determined by not only the inputs of the system, but also the output at previous instants. For example, for a Single Input and Single Output (SISO) system as shown in Fig. 1, the nonlinear recurrent model can be described by following equation:

$$y_{s(\text{k+1})} = f[u_{(\text{k-nu+1})}, \cdots, u_{(\text{k})}, y_{(\text{k-ny+1})}, \cdots, y_{(\text{k})}], \quad (1)$$

where u is the input, y is the output, y_s is the estimated output and f is the nonlinear mathematical model.

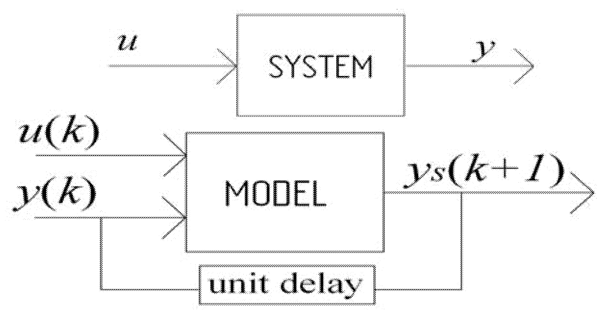

Figure 1: Scheme of a first order SISO dynamic system and its recurrent model, u and y are the input and output of the system, respectively.

(Advance online publication: 19 February 2008)

In order to model the system's dynamics, dynamic or recurrent models can be obtained from the recorded data of the input/output signals which are available in the form of numeric arrays. These data should be arranged properly to be used in the training/modelling. To obtain the model function f as shown in Eq. (1) through training, the recorded raw data of this SISO system can be organised as:

$$\text{raw data} = \left[\begin{array}{c|c} \text{input} & \text{output} \\ u_1 & u_2 \\ \vdots & \vdots \\ u_m & u_m \end{array}\right], \tag{2}$$

prepared data =

$$\left[\begin{array}{ccc|ccc|c} & \text{input} & & & & & \text{output} \\ u_1 & \cdots & u_{\text{m}} & y_1 & \cdots & y_{\text{m}} & y_{\text{ny+1}} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ u_{\text{m-r}} & \cdots & u_{\text{m+nu-r-1}} & y_{\text{m-r}} & \cdots & y_{\text{m+ny-r-1}} & y_{\text{m+ny-r}} \end{array}\right], \tag{3}$$

where m is the rank of the raw data vector, r is the order number of the system model, i.e., $r = \max(nu, ny)$. Usually, $ny \geq nu$, therefore, the order of the model is often considered as the number of delayed outputs that are used for the estimation as shown in Fig. 1.

A neural network was trained off-line using recorded data before operation; besides, it was trained on-line during operation as well. A perceptron structure with two layers of connections is used in this study. After training, for the first estimation, the input of the ANN includes the tentative control input of system (u'), previous control inputs of system ($u(k-i)$ when $i \geq 1$), current and previous actual outputs of systems ($y(k-i)$ when $i \geq 1$). The output of the ANN includes the first predicted value of the output $y_s(k+1)$. To estimate $y_s(k+i)$, when $i > 1$, the previously estimated values of y_s are used as previous output values of the system which were originally estimated based upon the actual outputs of the system, the actual inputs and the tentative control inputs of the system.

Predicted outputs of the ANN can be used to calculate the performance function. In the discrete domain, the performance function is defined as:

$$J(k) = \sum_{i=1}^{N} [y_s(k+i) y_d]^2 + \rho[u'(k) - u(k-1)]^2, \tag{4}$$

where y_s and y_d are the estimated and desired outputs of the system, respectively, u' and u are the tentative and actual control inputs, respectively. Additionally, ρ is a factor defining the importance of constancy of the control input. Among the right-hand side terms of Eq. (4) (the arguments of J function), u' is the only independent variable that is not influenced by the current and previous situation of the system. This variable can be selected arbitrarily and can affect other variables and the performance, whereas other terms of the right-hand side of the equation (arguments of J) are thoroughly dependent on the current situation of the system, therefore, their values can not be adjustable. In other words, for control purpose, it can be assumed that:

$$J = J(u'). \tag{5}$$

Now, u' should be so determined that J has its minimal value. To do this, the first-order approximation of the Taylors series for the performance function can be written as:

$$J(u' + \Delta u') \cong J(u') + \frac{\partial J(u')}{\partial u' \Delta u'}. \tag{6}$$

Differentiate Eq. (6), it becomes:

$$\frac{\partial J(u' + \Delta u')}{\partial u'} \cong \frac{\partial J(u')}{\partial u'} + \frac{\partial^2 J(u')}{\partial u'^2 \Delta u'}. \tag{7}$$

In order to minimise $J(u' + \Delta u')$, its derivative is set to zero. Consequently, it is obtained:

$$\Delta u' \cong -\frac{\partial J(u')}{\partial u'} \left[\frac{\partial^2 J(u')}{\partial u'^2 \Delta u'}\right]^{-1}. \tag{8}$$

The right-hand side of Eq. (8) is called Newtons direction [15]. Especially, a performance function gradient, g_k is defined as:

$$g_k \triangleq \frac{\partial J(u')}{\partial u'} = \frac{J(k) - J(k-1)}{u'(k) - u(k-1)}. \tag{9}$$

Furthermore, another performance function gradient, G_k, can be defined as:

$$G_k \triangleq \left[\frac{\partial^2 J(u')}{\partial u'^2 \Delta u'}\right]^{-1} = \frac{u'(k) - u(k-1)}{g(k) - g(k-1)}. \tag{10}$$

To modify the control input, the following relation can be used:

$$\Delta u' = u'_{new} - u'_{old} = -G_k g_k. \tag{11}$$

In practice, an adjustable coefficient is used for $G_k g_k$ to obtain a quicker convergence, i.e.,

$$\Delta u' = u'_{new} - u'_{old} = -\eta G_k g_k. \tag{12}$$

Using Eq. (12), it is obtained that:

$$J(u'_{new}) = J(u'_{old} - \eta G_k g_k). \tag{13}$$

Equation 13 can be rewritten as:

$$\text{Argument of } J = u'_{old} - \eta G_k g_k. \tag{14}$$

Both u'_{old} and $G_k g_k$ are known at this stage. While changing η, the *Argument of* J moves along a line. There is an optimum point on this line that minimises J. Such an optimisation problem is classified as a linear search. The backtracking method, introduced by Dennis and

Schnabel [16], is selected for the linear search. The modified $u'(u'_{new})$ is used as the new control input.

Beside neural modelling and selecting optimisation (predictive) algorithm, a simple Sugeno-type fuzzy inference system is designed to make the response of neuro-predictive controller decay quickly when the error is sufficiently small.

3 System Model

The model helicopter used in this research (as shown in Fig. 2) is a highly nonlinear two input-two output system. The helicopter has two degrees of freedom, the first possible motion is the rotation of the helicopter body with respect to the horizontal axis (which changes the pitch angle) and the second is rotation around the vertical axis (which change the yaw angle). The helicopter can rotate from $-170°$ to $170°$ in yaw angle, and from $-60°$ to $60°$ in pitch angle. System inputs are voltages of main and rear motors, and the yaw and pitch angles are considered as its outputs.

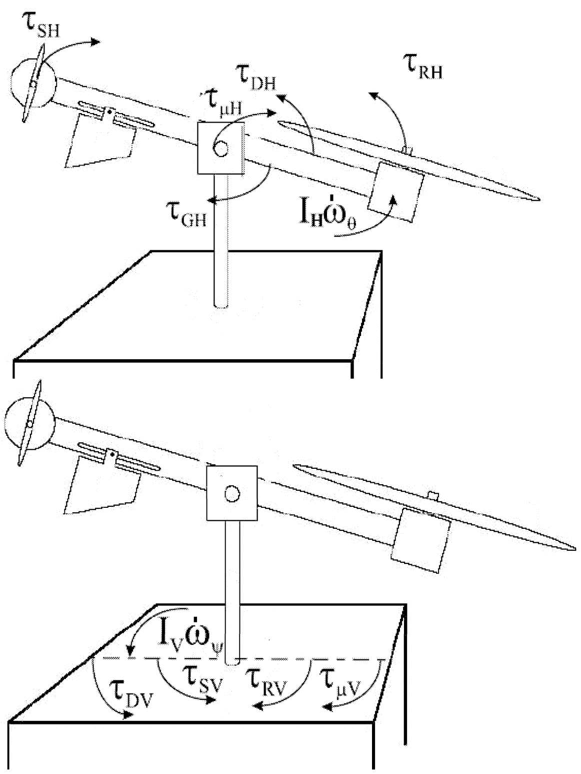

Figure 2: The torques exerted on the helicopter around the horizontal and vertical axes.

An analytical model is obtained using Newton and Euler's laws. After modelling, the following differential equations are obtained [17]:

$$\frac{d\omega_R}{dt} = \frac{\tau_{\rm RR} - \tau_{\rm AR} - \tau_{\mu\rm R}}{I_{\rm R}}, \quad (15)$$

$$\frac{d\omega_{\rm S}}{dt} = \frac{\tau_{\rm SS} - \tau_{\rm AS} - \tau_{\rm RRMF}}{I_{\mu\rm S}}, \quad (16)$$

$$\frac{d\theta}{dt} = \omega_\theta, \quad (17)$$

$$\frac{\omega_\theta}{dt} = \frac{\tau_{\rm RH} - \tau_{\rm SH} - \tau_{\mu\rm H} - \tau_{\rm GH} - \tau_{\rm DH}}{I_{\rm H}}, \quad (18)$$

$$\frac{d\Psi}{dt} = \omega_\Psi, \quad (19)$$

$$\frac{\omega_\Psi}{dt} = \frac{\tau_{\rm SV} - \tau_{\rm RV} - \tau_{\mu\rm V} + \tau_{\rm DV}}{I_{\rm V}}. \quad (20)$$

The variables and indices are listed at Nomenclature. In Eqs. (15 ∼ 20) the torques can be substituted by their equal expressions obtained from kinetics of the system.

In this research, a special situation is studied. The motion of the helicopter is so constrained that the vertical motion is negligible. Moreover, the input voltage of the main motor is set to *zero*. As a result, the only input of the system is the input voltage of the rear motor. Also, the yaw angle (the angle in the horizontal plane) is considered as the unique output. In this situation, the behaviour of the system can be represented only by Eqs. (19) and (20). Since there is no change in the pitch angle, gyroscopic torque does not exist. Furthermore, the main motor does not generate any torque as there is no excitation applied on it. Consequently, Eq. (20) is simplified as below:

$$\frac{\omega_\Psi}{dt} = \frac{\tau_{\rm SV} - \tau_{\mu\rm V}}{I_{\rm V}}, \quad (21)$$

where:

$$\tau_{\rm SV} = r_{\rm S}k_{\mu\rm V}\text{sign}(\omega_{\rm S})(\omega_{\rm S})^2, \quad (22)$$

$$\tau_{\mu\rm V} = C_{\mu\rm V}\omega_\Psi. \quad (23)$$

The equations defining the behaviour of this first order system can be written as:

$$\dot{\Psi} = \omega_\Psi,$$
$$\dot{\omega}_\Psi = \frac{r_{\rm S}k_{\mu\rm V}\text{sign}(\omega_{\rm S})(\omega_S)^2 - C_{\mu\rm V}\omega_\Psi}{I_{\rm V}}, \quad (24)$$

where ω_Ψ is the angular velocity of helicopter body in the yaw direction and ω_S is the angular velocity of the rear motor blades that is a nonlinear function of the input voltage of rear motor.

4 Design of Hybrid Controller

4.1 Neural Network Model

A three layer recurrent perceptron is used to model the system. The numbers of neurons in the input and hidden layers are 6 and 7 regardless of biases. The value of each bias is 1. The input and output layers have linear activation functions with slope of one, whereas, the neurons in the hidden layer have sigmoid activation functions, i.e., the output of the i^{th} neuron of the hidden layer is:

$$O_i = tanh(\sum_{j=1}^{7} w_{ij} y_j), \tag{25}$$

where w_{ij} is the weight of connections between i^{th} neuron of the hidden layer and j^{th} neuron of the input layer whose output is y_j, and 7 is the number of neurons in the input layer in addition to the bias. In this research, Levenberg-Marquardt algorithm [15] is applied for batch back-propagation training. A scheme of neural network together with the input and output data during training is shown in Fig. 3.

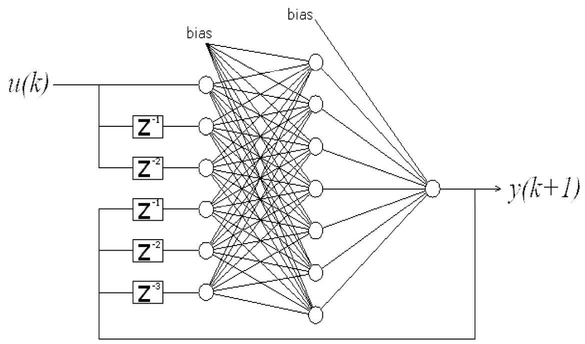

Figure 3: Neural network structure and input-output data in training stage.

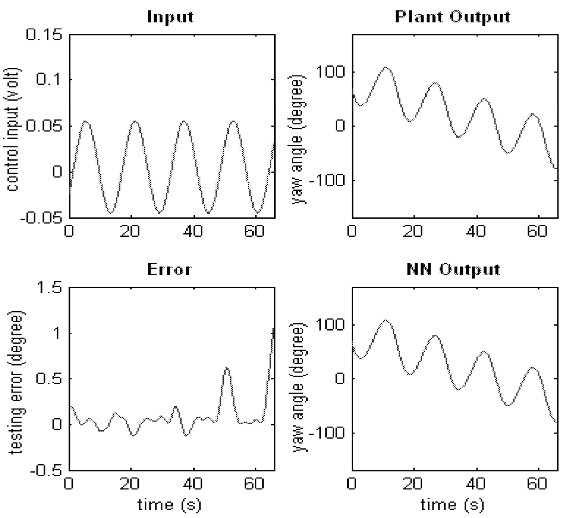

Figure 4: Verification information of ANN regarding testing area.

A set of 1300 input-output recorded data of system was used for training. In order to obtain such a data set, pulse signals were sent to the system with a time interval of 0.1 second for 130 seconds and the output values were recorded at any time. Testing data were obtained by sending a sinusoidal voltage signal to the rear motor of the helicopter. The training was completed with only 8 iterations.

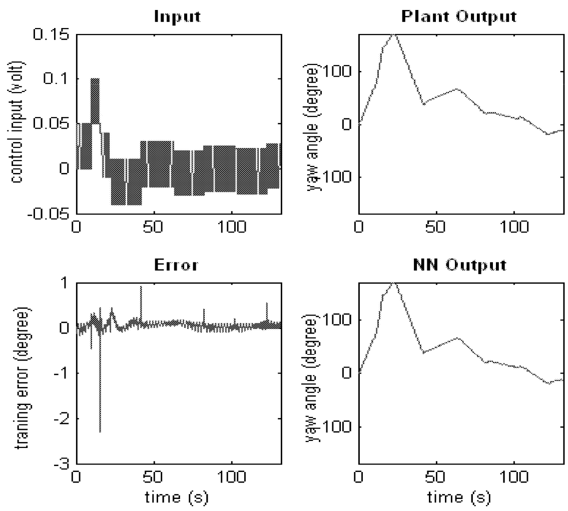

Figure 5: Verification information of ANN regarding training area.

The performance function of training is the sum of squared errors, and the data were normalised before training. Figures 4 and 5 both illustrate the success of the training.

After the successful training, the neural network was used to predict the future outputs of the system. Unlike the training stage, the inputs and outputs used here were not recorded data. For the first estimation, the neural network's inputs and outputs are shown in Fig. 6.

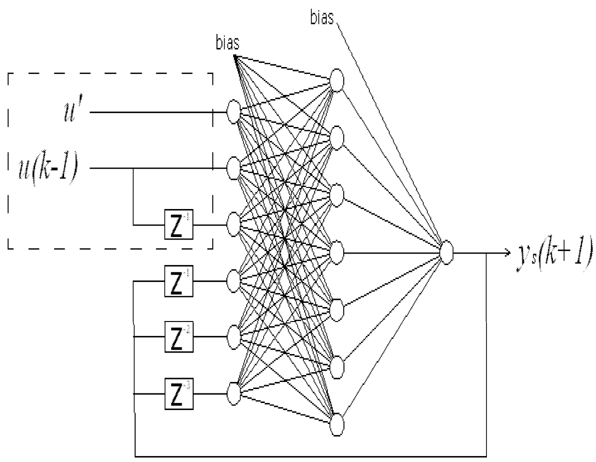

Figure 6: Neural network structure and input-output data in the first estimating (predicting) stage.

The tentative input u' and the estimated output y_s both appear in the estimation stage. For next steps of estimation, $y_s(k+1)$ is replaced by $y_s(k+i)$ $(i > 1)$. It is shown in Fig. 6 with a dashed rectangle that some of the inputs of the ANN change their values when $y_s(k+i)$ $(i > 1)$ is estimated. For example, when $i = 2$ or $y_s(k+2)$ is estimated, the inputs of the ANN located in the dashed rectangle of Fig. 6, are shown in Fig. 7. The case of $i > 2$, is shown in Fig. 8.

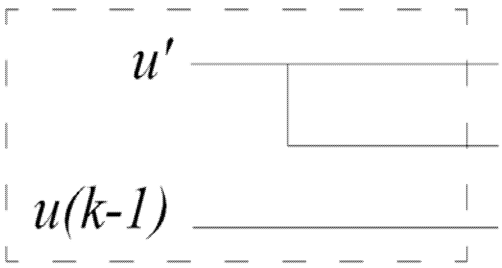

Figure 7: Dashed area of Fig. 6 for i=2.

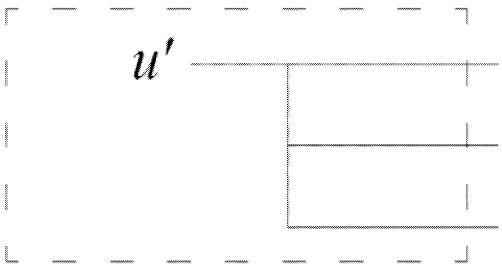

Figure 8: Dashed area of Fig. 6 for i > 2.

4.2 Predictive Control

In order to explain the details of predictive control, Eq. (4) is re-written here:

$$J(k) = \sum_{i=1}^{N} [y_s(k+i) y_d]^2 + \rho[u'(k) - u(k-1)]^2.$$

The purpose of predictive control is to define tentative control input u' so that J is minimised. Using the designed and trained ANN in the previous section, the predicted output values can be calculated. To obtain the predicted output values, at each instant, the ANN should be used N times as shown in Eq. (4). The estimated (predicted) output value of any stage of prediction is applied as one of the inputs for the next prediction stage. In this study, $N = 7$ is used. The whole neural predictive model can be assumed as having seven sequential identical neural networks, any one output (except the last one) provides one of the inputs for the next ANN. The *Neural Predictive Model* obtains the predicted output values of the system $(y(k+i), i = 1 \sim 7)$, using the previous and current values of the output of the system y, the previous values of the control input u and the tentative control input u'. Consequently, the values of the performance function J can be calculated as shown in Fig. 9.

In order to explain the total process of predictive control, a general model is considered as the sum of the neural predictive model and J function. The output of this general model is the value of $J(k)$. Additionally, as previously stated, the tentative control input is derived from the nonlinear optimisation function which is a combination of Eqs. (9),(10), (12), and the linear search. The neuro-predictive control algorithm is shown in Fig. 10. For the first step, $J(-1)$ and $J(-2)$ should be determined using previous recorded data.

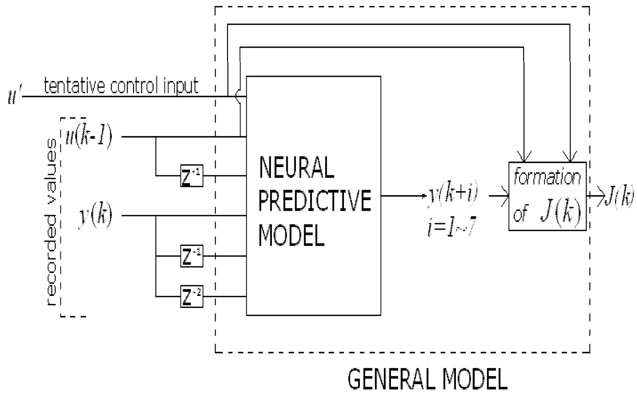

Figure 9: The process of calculation of the performance function.

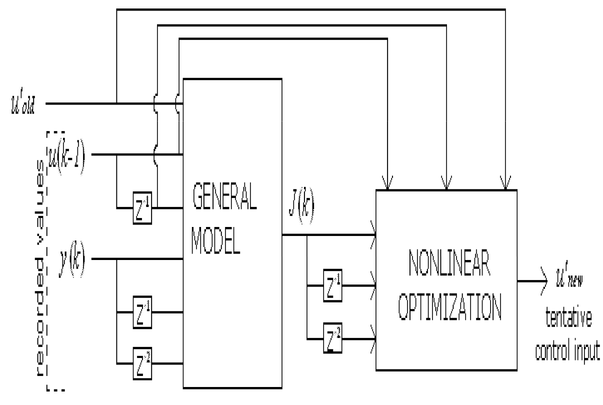

Figure 10: Neuro-predictive controller.

4.3 Fuzzy compensator

After implementation of the neuro-predictive controller, it was observed that it reached the desired setpoint more quickly than the existing fuzzy controller, but a serious problem was also observed. The system under neuro-predictive control has a large percent overshoot and a long settling time. In order to solve this problem, a fuzzy compensator is added to the controller. The input of this fuzzy inference system is the absolute value of the error and the output is a coefficient multiplying by the tentative control input (derived from the neuro-predictive algorithm) to achieve a modified control input. This fuzzy compensator is a Sugeno-type FIS with only two rules, i.e.,

- if the *absolute error* is A, then the correction coefficient = 2;
- if the *absolute error* is any other value, then the correction coefficient = 0.

Here A is a Gaussian membership function whose membership grade can be calculated as:

$$mg = exp[-\frac{1}{2}(\frac{\text{absolute error} - 20}{5})^2]. \quad (26)$$

A scheme of this fuzzy inference system is shown in Fig.11.

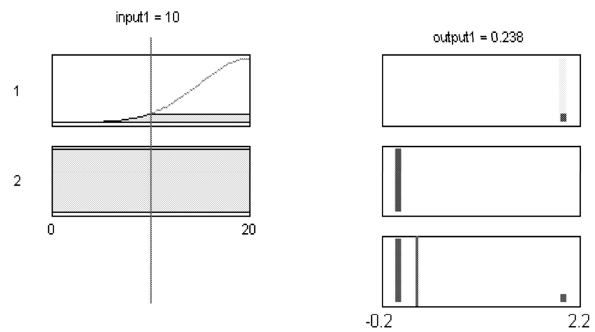

Figure 11: Fuzzy compensator scheme.

The role of fuzzy compensator is to reduce the control input (the rear motor voltage) when the error is small. The effect of this simple corrector is discussed in the next section.

5 Simulation Results

The responses of the closed loop system for the three different controllers are shown in Fig.12 for four different set-points. In those cases, the helicopter was rotated form a stationary situation to the desired values of yaw angle by the rear motor, while the rear motor was controlled by an existing *fuzzy controller*, *neuro-predictive controller* or *NPFC*, separately.

An energy consumption criterion (*ECC*) is defined to represent the total energy consumption of the closed loop system during operation, and it can be calculated as:

$$ECC = \int_0^T | u(t) | dt, \quad (27)$$

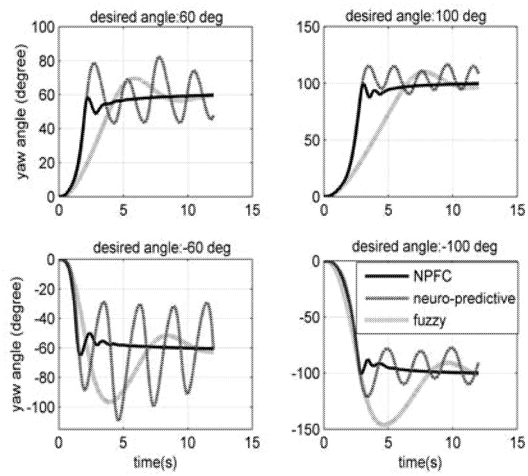

Figure 12: Responses of different controllers and set-points.

where T is the final time for calculation and $u(t)$ is the input voltage of the helicopter's rear motor or control input. Since the neuro-predictive controllers are essentially designed to reduce the deviation of the inputs rather than the absolute value of inputs, another criterion namely, the input deviation criterion (*IDC*) is needed and defined as:

$$IDC = \int_0^T | u(t) - u(t-\tau) | dt, \quad (28)$$

where τ is the sampling time of the system.

Table 1 shows the simulation results for the four cases. The table also shows information about the maximum overshoot and the settling time for the yaw angle to be settled within 5 degrees of the desired value.

Table 1: Operational information of different controllers.

Set point (°)	Controller type	ECC $(V \cdot s)$	IDC $(V \cdot s)$ (°)	Maximum overshoot %	Settling time (s)
-60	Fuzzy	5.434	0.32	36.367	7.4
	NP	11.26	2.83	49.238	-
	NPFC	9.784	0.79	9.124	2.7
-100	Fuzzy	6.758	0.33	45.222	10.8
	NP	11.80	0.97	24.781	-
	NPFC	12.51	0.69	10.041	5
60	Fuzzy	4.101	0.16	10.239	6.9
	NP	21.14	2.75	22.023	-
	NPFC	22.64	1.32	7.931	4.1
100	Fuzzy	5.951	0.16	16.734	9.0
	NP	27.92	1.95	18.913	-
	NPFC	28.38	1.41	6.247	5.3

It is clearly shown from this study that the proposed NPFC controller performs better in terms of the overshoot and settling time in comparison to the existing fuzzy controller which is considered as a satisfactory controller for this type of high inertia systems. From the simulations, it can be seen that the control input generated by NPFC controller does not exceed the permitted range for the input voltage although it consumes more energy.

6 Robustness Analysis

In this section, the designed controller (NPFC) is evaluated regarding the parameter changes and disturbance rejection. In the first case, a NPFC controller with $\rho = 2$ was designed. In order to test the robustness of the controller, the helicopter was exposed to a sudden impact causing a dramatic yaw angle change over 25°. The disturbances were exerted around the sixth second time mark during systems operation as shown in Fig. 13. At the same moment the error was about 2° and converged to zero. The desired yaw angles were 80° and −80°. The assumed impacts were considerably severer than those impacts may be encountered in reality. The controlled system shows very good robustness to the disturbance. Fig. 13 shows the response of the NPFC controlled system under the mentioned disturbances.

In order to analyse the performance under parameter changes of the system, Eq. (19) and Eq. (24) defining systems dynamic can be re-written as:

$$\begin{aligned} \dot{\psi} &= \omega_{\psi}, \\ \dot{\omega}_{\psi} &= \frac{1}{I_{\mathrm{V}}}(r_{\mathrm{S}} k_{\mathrm{S}} \mathrm{sign}(\omega_{\mathrm{S}}) \omega_S^2 - c_{\mu\mathrm{V}} \omega_{\psi}). \end{aligned}$$

There are four parameters in these equations: the moment of inertia I_{V} for the helicopter body around its vertical axis, the distance of the rear motor r_{S} from the joint of the helicopter body with its basis (shown in Fig. 3), the rear motor blade constant $k_{\mu\mathrm{S}}$ and the friction coefficient for rotation around vertical axis $c_{\mu V}$. Among these parameters, I_{V} and r_{S} are geometrical constants. As to the operation environment of the model helicopter, k_{FS} is also assumed to be a constant. Therefore, the only variant parameter of system is $c_{\mu\mathrm{V}}$ whose original value for properly lubricated joint is 0.0095.

Dust or lack of fabrication may cause this parameter increased. Fig. 14 shows the effect of a sudden increase of this parameter by 300% during operation. The increase of $c_{\mu V}$ occurred at the fourth and the sixth second time marks. The NPFC controlled system performs well under the test.

Two main reasons can be attributed to the robustness of the NPFC controller, i.e.,

(i) The output of fuzzy compensator (and consequently the controller's input) increases considerably as soon as the absolute error increases.

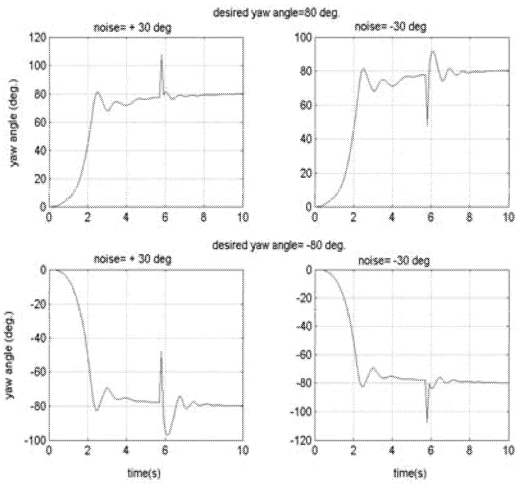

Figure 13: Responses of the NPFC controlled system with disturbance.

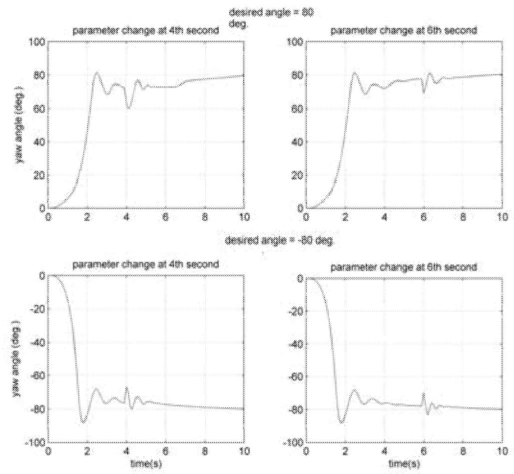

Figure 14: Responses of the NPFC controlled system with sudden parameter change.

(ii) On-line training makes the neural network model adaptive to changes in the model parameters.

7 Conclusion

In this paper, neuro-predictive controllers are studied and implemented on a system with non-linear and non-symmetric dynamics. Although pure neuro-predictive controllers do not work well for this system, but NP controllers, combined with a fuzzy compensator, shows a satisfactory response in comparison to the existing fuzzy controllers. In the NPFC the control input is adjusted through multiplying by a small positive number generated by fuzzy inference system. The fuzzy compensator is designed so that as the error is small, the output converges to zero. It is indicated that the designed controller improves the performance of the closed loop system. Moreover, it is shown that the NPFC controlled system is very robust to disturbance and parameter changing.

References

[1] Camacho, E. Fernndez, Bordons, C., *Model Predictive Control*, Springer, 2004.

[2] Ghafari, A., Nikkhah Bahrami M., Mohammadzaheri, M., "Linear Modeling of Nonlinear Systems Using Artificial Neural Networks Based on I/O Data and Its Application in Power Plant Boiler Modeling (in Persian)", *Journal of Engineering Faculty of University of Tehran, Mechanics and Metallurgy*, V39, N1, pp. 53-60,1/05.

[3] Seyab,R.K. Al, Cao, Y.,"Nonlinear model predictive control for the ALSTOM gasifier" *Journal of Process Control* V16,pp. 795-808, 2006

[4] Soloway, D. Haley, P.J., "Neural Generalized Predictive Control," *Proceedings of the 1996 IEEE International Symposium on Intelligent Control*, pp. 277-281, 1996

[5] Razi, M., Farrokhi, M., Saeidi, M.H., Khorasani, A.R.F., "Neuro-Predictive Control for Automotive Air Conditioning System, *IEEE, International Conference on Engineering of Intelligent Systems* pp. 1 - 6, 4/06.

[6] Zhang, Yun, "The research on the GA-based neuro-predictive control strategy for electric discharge machining process", *International Conference on Machine Learning and Cybernetics*, V2, pp. 1065 - 1069, 8/04

[7] Parlos, Alexander G., Parthasarathy, Sanjay, Atiya, Amir F., "Neuro-Predictive Process Control Using On-Line Controller Adaptation", *IEEE Transactions on Control system Technology*, V9, No.5, pp.741 - 755, 9/01

[8] Vila, Jean-Pierre, Wagner, V'erene, "Predictive neurocontrol of uncertain systems: design and use of a neuro-optimizer", *Automatica*, V39, pp. 767 - 777, 2003.

[9] M. Morari, C.E. Garcia, J.H. Lee, D.M. Prett; *Model Predictive Control*; Prentice Hall, 1994.

[10] Akesson, Bernt M., Toivonen, Hannu T., "A neural network model predictive controller", *Journal of Process Control*, V16, N9,pp.937-946,10/06

[11] S.W. Wang, D.L. Yu, J.B. Gomm, G.F. Page, S.S. Douglas, "Adaptive neural network model based predictive control for air-fuel ratio of SI engines", *Engineering Applications of Artificial Intelligence*, V19, pp.189-200, 2006

[12] Al-Alawi, Ali, Al-Alawi, Saleh M., Islam, Syed M., "Predictive control of an integrated PV-diesel water and power supply system using an artificial neural network" , *Renewable Energy.* V32, N8, pp. 1426-1439, 07/07

[13] Koker, Rasit " Design and performance of an intelligent predictive controller for a six-degree-of-freedom robot using the Elman network", *Information Sciences* V176, pp. 1781-1799,2006

[14] Schlotthauer, Gaston, Gamero, Lucas G., Torres, Maria E., Nicolini, Guido A., "Modeling, identification and nonlinear model predictive control of type I diabetic patient" *Medical Engineering & Physics* V28, pp.240-250, 2005

[15] J. R. Jang, C. Sun, E. Mizutani. *Neuro-Fuzzy and Soft Computing*, Prentice-Hall Inc., 1997.

[16] Jr., J.E. Dennis, Schnabel, Robert B., *Handbooks in Operations Research and Management Science*, V1, optimisation, Chapter I "A view of unconstrained optimisation", pp. 1-72, 1989.

[17] Unneland, Kari, *Application of Model Predictive Control to a Helicopter Model* , Master's thesis , Swiss Federal Institute of Technology, Zurich, 2003

Nomenclature

ω	angular velocity
θ	pitch angle
ψ	yaw angle
I	moment of inertia
τ_{RR}	Torque generated by the main motor
τ_{AR}	Torque generated by the air friction in the main motor
$\tau_{\mu R}$	Torque generated by the main motor's mechanical friction
τ_{SS}	Torque generated by the rear motor
τ_{AS}	Torque generated by by the air friction in the rear motor
$\tau_{\mu S}$	Torque generated by the rear motor's mechanical friction
τ_{RH}	Torque generated by the main motor around the horizontal axis
τ_{SH}	Torque generated by the rear motor around the horizontal axis
$\tau_{\mu H}$	Torque generated by the mechanical friction around the horizontal axis
τ_{DH}	Gyroscopic torque around the horizontal axis
τ_{GH}	Gravitational torque around the horizontal axis
τ_{RV}	Torque generated by the main motor around the vertical axis
τ_{SV}	Torque generated by the rear motor around the vertical axis
$\tau_{\mu V}$	Torque generated by the mechanical friction around the vertical axis
τ_{DV}	Gyroscopic torque around the vertical axis
S	relevant to rear rotor
R	relevant to main rotor
H	around horizontal axis
V	around vertical axis
D	relevant to gyroscopic effect
G	relevant to gravitational force
A	relevant to air friction
μ	relevant to mechanical friction
η	convergence coefficient