Application of Neural Network to Identify Black box Model of Twin Rotor MIMO System Based on Mean Squared Error Method

Huong T.M. Nguyen[1]
[1] Thai Nguyen University of Technology,
Thai Nguyen city, Viet Nam

Mai Trung Thai[1]
[1] Thai Nguyen University of Technology,
Thai Nguyen city, Viet Nam

Abstract:- In model predictive control, building the correct model and solving the optimal problem are two jobs that always require lots of time. These are also two issues that many scientists are interested in studying when applying model-driven reporting control to certain objects. With a TRMS object, we can build a white box model, a gray box model or a black box model.

Some authors have built TRMS model published in [2], [3], [4], [5]. We have studied the solving methods of optimal problem in model predictive control in [6], [7], [8].

In [9], we built a white box model of TRMS object according to Newton method. Studying the effects of the interchannel effects of the white box model TRMS. In [10], we used Gradient descent back-propagation, and some of its conjugate algorithms to identify the TRMS model. In this paper, we applicate the neural network in order to identify black box model of twin rotor MIMO system based on mean squared error method, using these results compare to the real model so as to choose a suitable algorithm and provide the ability to apply that model in simulation and object control.

Keywords *- Black Box Model, Neural Network, Yaw Angle, Pitch Angle, Identify, Mean Squared Error.*

I. INTRODUCTION

Artificial Neural Networks (ANNs) are intended to mimic the behavior of biological neural networks (NNs). In fact, there are many types of neural networks and their applications are also different. With a particular application, these networks will be trained by using the suitable algorithms. We select MLP network (Multi Layer Perceptron) to model the TRMS object with many different training methods. Hence, MLP network and related algorithms have also been briefly presented in this paper.

The TRMS system is a equipment in the laboratory [1] and it is a suitable test equipment to evaluate and perform the advanced control techniques. The TRMS system is connected to a PC through an interface to transfer control signals to the actuators and to receive the corresponding feedback signals from the sensors.

The system includes two propellers perpendicular to each other with two movements, in which one moves vertically and the other moves horizontally. However, both these movements affect each other.

Therefore, the more accurate the TRMS model building, the higher the quality of control in general and predictive control in particular.

II.THE MODEL OF TRMS

The physical model of TRMS system is shown in Fig 1.

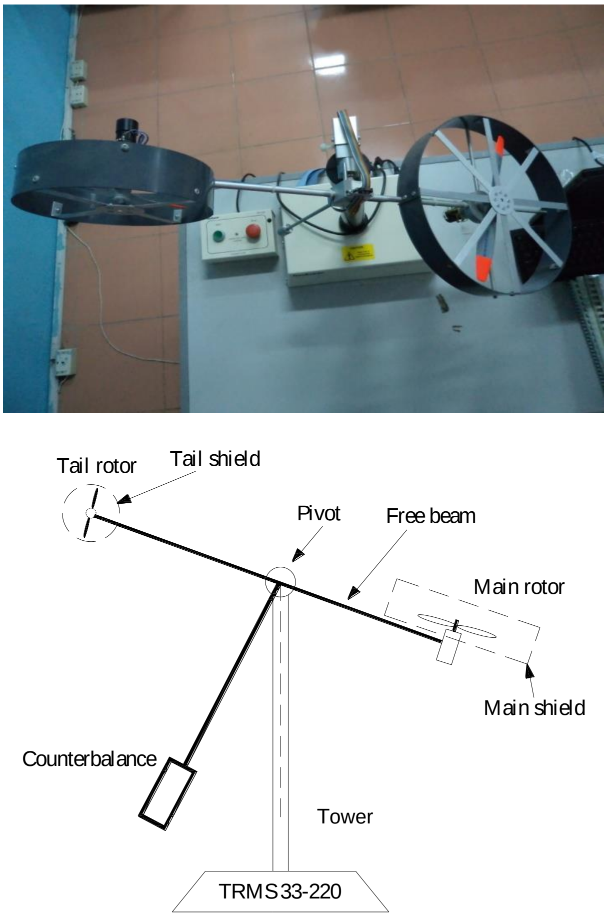

Fig 1. The physical model of TRMS

A. General Principles of MLP Network

The structure of the network, its parameters include weights, biases and the type of its processing elements determine the function of a network. The number of layers and the number of neurons in each layer determine the structure of a typical MLP network. An iterative process decides the structure of an MLP network. Some algorithms such as Levenberg-Marquardt back-propagation, gradient descent back-propagation, quasi-Newton back-propagation, Bayesian regularisation back-propagation, conjugate gradient back-propagation, one step secantback-propagation, resilient back-propagation, and scaled conjugate gradient back-propagation can be used to train the MLP network.

A.1. Designing the stimulative Signal

We can use various different input signals, such as sine and square waves with different amplitudes and frequencies to stimulate the TRMS object. The frequency from 0.01Hz to 1Hz and the voltage within -2.5V and 2.5V will be used to train the MLP network.

A.2. Model Structure Selection and Training algorithms

We select the Neural Network Auto Regressive External input Model Structure (NNARX) approach to model the system, because the input and output datasets of the real system is available.

In an NNARX model architecture, the inputs are the past control inputs of the real system:

$$\left[u(t-d), u(t-d-1), \ldots, u(t-d-m)\right]$$

and also the past observed outputs:

$$\left[y(t-1), y(t-2), \ldots, y(t-n)\right]$$

where:

- d is multiple of the sampling period.
- m is input delay space
- n is output delay space

The NNARX model architecture of a SISO system is shown in Fig 2.

with $\hat{y}(t)$ is written by

$$\hat{y}(t) = f_{NN}\,(u(t-d), u(t-d-1), \ldots,\ u(t-d-m), y(t-1), y(t-12), \ldots, y(t-n)) \qquad (1)$$

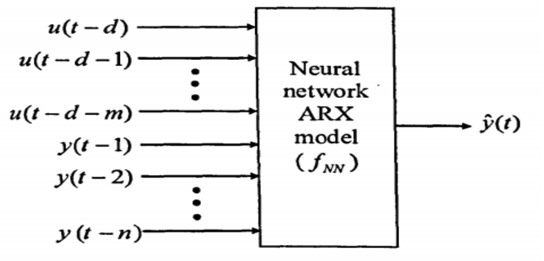

Fig 2. The NNARX model architecture of a SISO system

Some algorithms are used in the artcle consitting of:

+ Levenberg-Marquardt (LM) back-propagation;
+ BFGS Quasi-Newton back-propagation;
+ Bayesian regularisation back-propagation;
+ One step secant back - propagation;
+ Resilient;

These algorithms can be called Mean Squared Error (MSE) method. The target of MSE model is to determine the weights and the biases of an NN by minimizing the criterion as follow:

$$V = \frac{1}{2Q}\sum_{q=1}^{Q} e_q^T e_q \qquad (2)$$

with

$$e_q = t_q - y_q \qquad q = 1, 2, ..Q \qquad (3)$$

Q : Number of samples for training

t_q : Vector of target output of sample *q*.

y_q : Vector of prediction output of sample *q*.

e_q : Error of sample

V : Mean squared error

A.3. Model Test

Model test is a important work in which a model is tested to clarify whether the model adequately represents the characteristics of the real system corresponding. Each tested model will be suitable for its application in the future.

Following [5], we have

$$r_{ee}(\tau) = \frac{\sum_{i=1}^{N-\tau}(e_i - \bar{e})(e_{i+\tau} - \bar{e})}{\sum_{i=1}^{N}(e_i - \bar{e})^2} = \begin{cases} 1 & \text{if } \tau = 0 \\ 0 & \text{if } \tau \neq 0 \end{cases} \qquad (4)$$

$$r_{ue}(\tau) = \frac{\sum_{i=1}^{N-\tau}(u_i - \bar{u})(e_{i+\tau} - \bar{e})}{\sqrt{\sum_{i=1}^{N}(u_i - \bar{u})^2 \sum_{i=1}^{N}(e_i - \bar{e})^2}} = 0 \quad \forall\ \tau \qquad (5)$$

$$r_{u^2e^2}(\tau) = \frac{\sum_{i=1}^{N-\tau}(u_i^2 - \bar{u}^2)(e_{i+\tau}^2 - \bar{e}^2)}{\sqrt{\sum_{i=1}^{N}(u_i^2 - \bar{u}^2)^2 \sum_{i=1}^{N}(e_i^2 - \bar{e}^2)^2}} = 0 \quad \forall\ \tau \qquad (6)$$

$$r_{u^2e}(\tau) = \frac{\sum_{i=1}^{N-\tau}(u_i^2 - \bar{u}^2)(e_{i+\tau} - \bar{e})}{\sqrt{\sum_{i=1}^{N}(u_i^2 - \bar{u}^2)^2 \sum_{i=1}^{N}(e_{i+\tau} - \bar{e})^2}} = 0 \quad \forall\ \tau \qquad (7)$$

$$r_{e\beta}(\tau) = \frac{\sum_{i=1}^{N-\tau}(e_i - \bar{e})(\beta_{i+\tau} - \bar{\beta})}{\sqrt{\sum_{i=1}^{N}(e_i - \bar{e})^2 \sum_{i=1}^{N}(\beta_i - \bar{\beta})^2}} = 0 \quad \tau \geq 0 \qquad (8)$$

In which
Eq.(4) is the auto-correlation of prediction errors.
Eq. (5) is the cross-correlation between the residuals of a model and the inputs
Eq.(6) is the cross-correlation of the squared errors and the squared inputs
Eq.(7) is the cross-correlation between the residuals of the model and the squared inputs.
Eq.(8) is the cross-correlation between the inputs and the residuals as well as its multiplication.
In the equations from Eq. (4) to Eq.(8):

- $\beta_i = u_i e_i$ (9)
- The bar above each symbol represents the average value of a signal as

$$\bar{x} = \frac{1}{N}\sum_{q=1}^{Q} x_i \qquad (10)$$

with N as the total number of data.

It is common to test whether the functions for delays in the interval $\tau \in [-20;20]$ are zero within an asymptotic 95% confident interval, i.e. if the following condition is held:

$$-1.96/\sqrt{N} < r < 1.96/\sqrt{N} \qquad (11)$$

III.SIMULATION RESULTS

The TRMS object's black box model with 2 degrees of freedom on Matlab/Simulink and the real model when using the Levenberg-Marquardt algorithm to train and test the network is shown from Fig 3 to Fig 6.

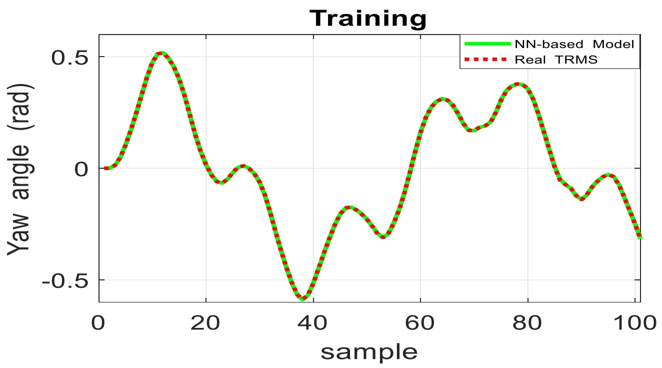

Fig 3. The yaw angle of the TRMS model when using the Levenberg-Marquardt algorithm to train the network

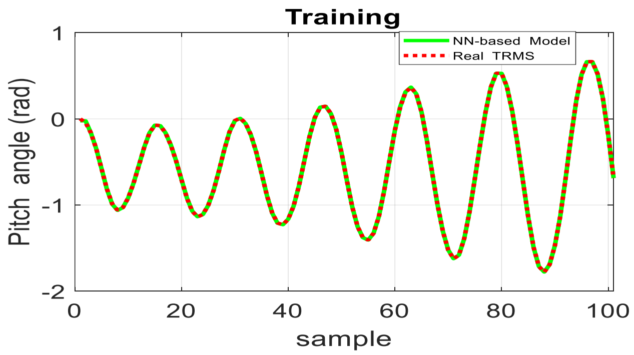

Fig 4. The pitch angle of the TRMS model when using the Levenberg-Marquardt algorithm to train the network

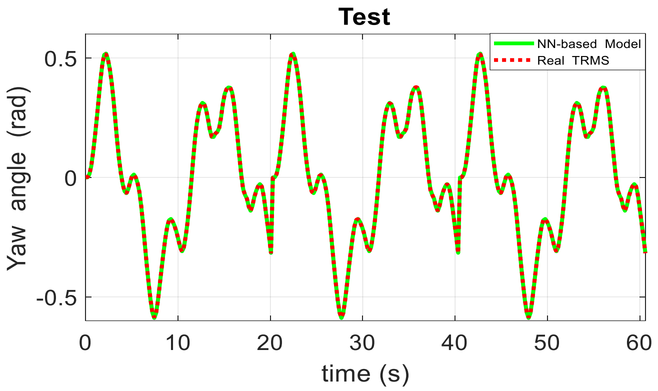

Fig 5. The yaw angle of the TRMS model when using the Levenberg-Marquardt algorithm to test the network

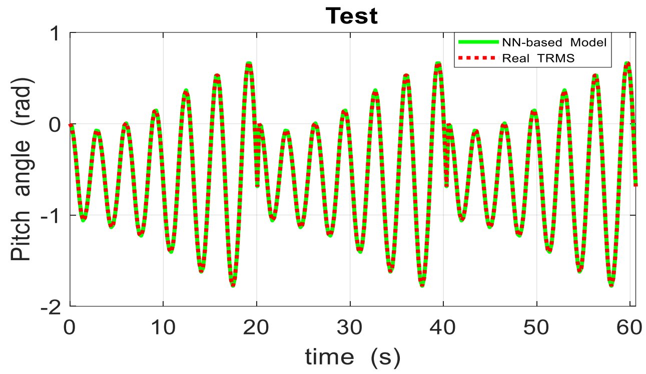

Fig 6. The pitch angle of the TRMS model when using the Levenberg-Marquardt algorithm to test the network

Remaks

The mean squared error of the pitch angle in training process is 3.167360. 10^{-6}

The mean squared error of the pitch angle in testing process is 2.100938. 10^{-4}

The mean squared error of the yaw angle in training process is 8.355767. 10^{-6}

The mean squared error of the yaw angle in testing process is 1.367820. 10^{-4}

The TRMS object's black box model with 2 degrees of freedom on Matlab/Simulink and the real model when using the BFGS quasi-Newton algorithm to train and test the network is shown from Fig 7 to Fig 10.

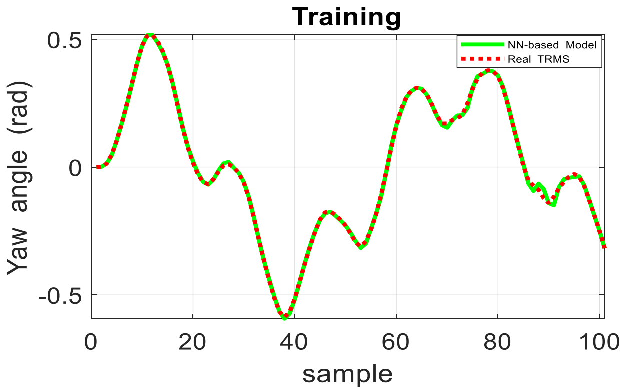

Fig 7. The Yaw angle of the TRMS model when using the BFGS quasi-Newton algorithm to train the network

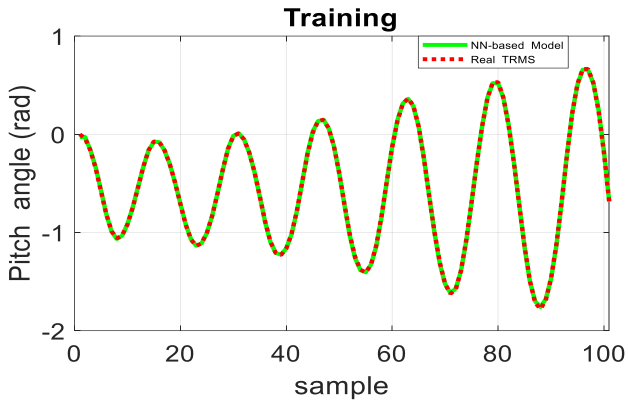

Fig 8. The Yaw angle of the TRMS model when using the BFGS quasi-Newton algorithm to train the network

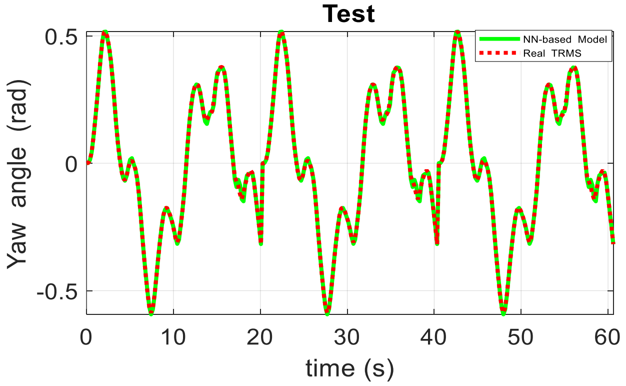

Fig 9. The Yaw angle of the TRMS model when using the BFGS quasi-Newton algorithm to test the network

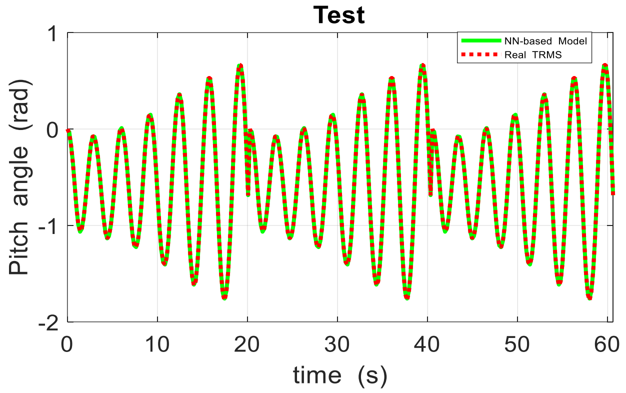

Fig 10. The pitch angle of the TRMS model when using the BFGS quasi-Newton algorithm to test the network

Remaks

The mean squared error of the pitch angle in training process is 4.665457. 10^{-5}

The mean squared error of the pitch angle in testing process is 4.665457. 10^{-5}

The mean squared error of the yaw angle in training process is 5.769073. 10^{-5}

The mean squared error of the yaw angle in testing process is 5.769073. 10^{-5}

The TRMS object's black box model with 2 degrees of freedom on Matlab/Simulink and the real model when using the Bayesian regularisation algorithm to train and test the network is shown from Fig 11 to Fig 14.

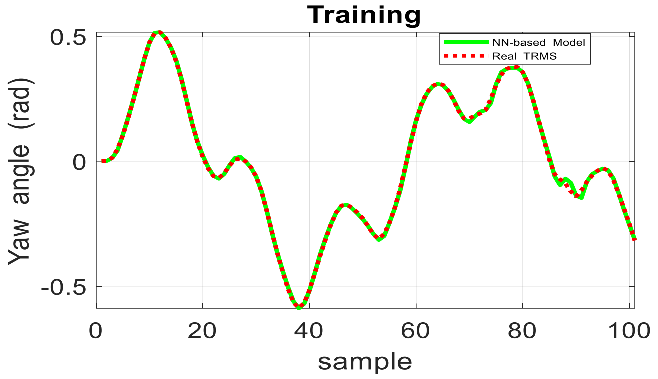

Fig 11. The yaw angle of the TRMS model when using the Bayesian regularisation algorithm to train the network

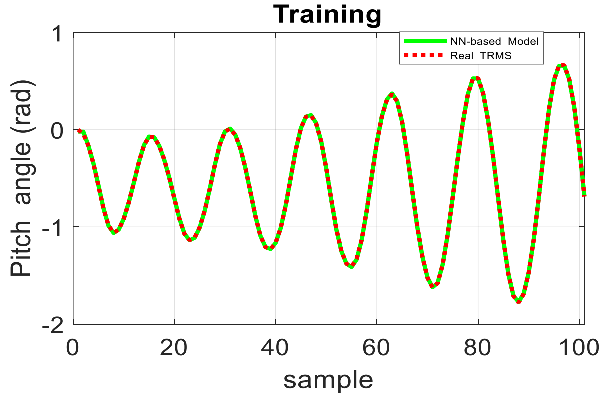

Fig 12. The pitch angle of the TRMS model when using the Bayesian regularisation algorithm to train the network

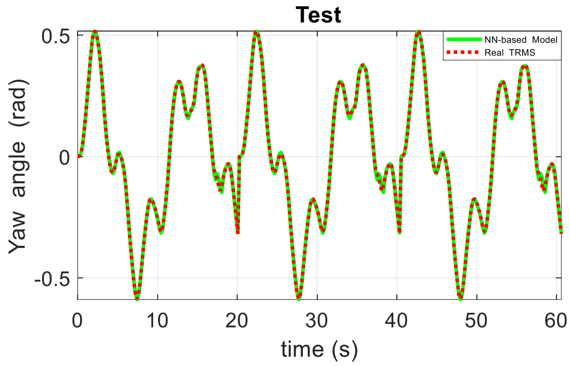

Fig 13. The yaw angle of the TRMS model when using the Bayesian regularisation algorithm to test the network

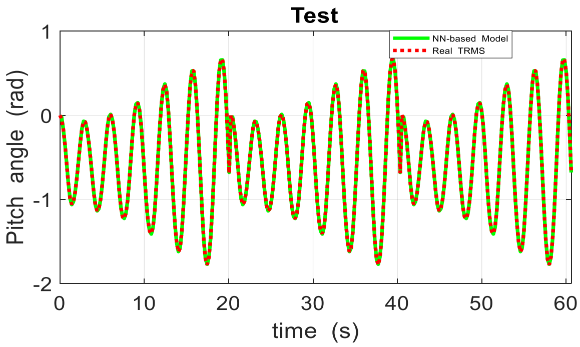

Fig 14. The pitch angle of the TRMS model when using the Bayesian regularisation algorithm to test the network

Remaks

The mean squared error of the pitch angle in training process is 1.942704. 10^{-5}

The mean squared error of the pitch angle in testing process is 1.942704. 10^{-5}

The mean squared error of the yaw angle in training process is 4.993197. 10^{-5}

The mean squared error of the yaw angle in testing process is 4.993197. 10^{-5}

The TRMS object's black box model with 2 degrees of freedom on Matlab/Simulink and the real model when using the One step secant algorithm to train and test the network is shown from Fig 15 to Fig 18.

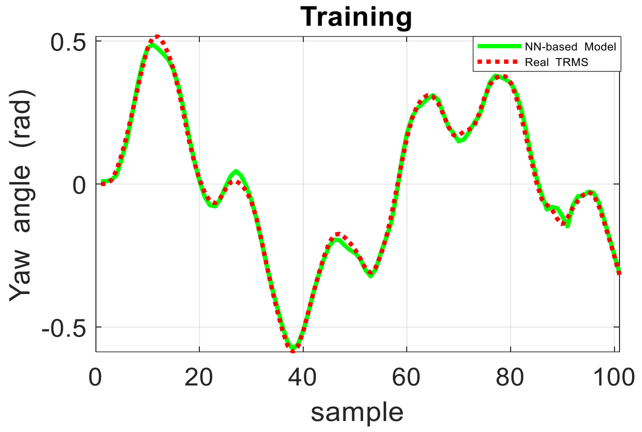

Fig 15. The yaw angle of the TRMS model when using the One step secant algorithm to train the network

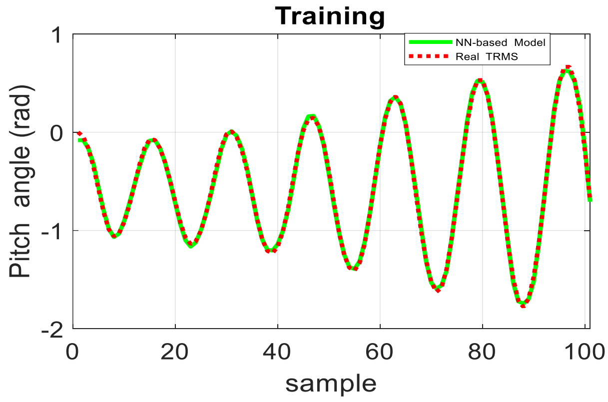

Fig 16. The pitch angle of the TRMS model when using the One step secant algorithm to train the network

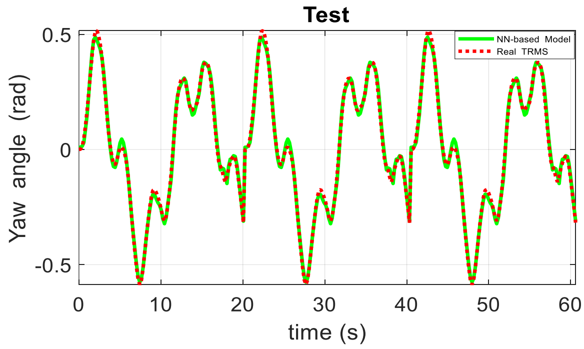

Fig 17. The yaw angle of the TRMS model when using the One step secant algorithm to test the network

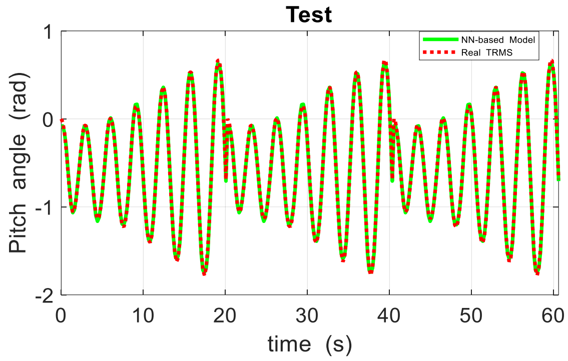

Fig 18. The pitch angle of the TRMS model when using the One step secant algorithm to test the network

Remaks:

The mean squared error of the pitch angle in training process is 4.610790. 10^{-4}

The mean squared error of the pitch angle in testing process is 4.610790.10^{-4}

The mean squared error of the yaw angle in training process is 2.402541.10^{-4}

The mean squared error of the yaw angle in testing process is 2.402541.10^{-4}

The TRMS object's black box model with 2 degrees of freedom on Matlab/Simulink and the real model when using the Resilient algorithm to train and test the network is shown from Fig 19 to Fig 22.

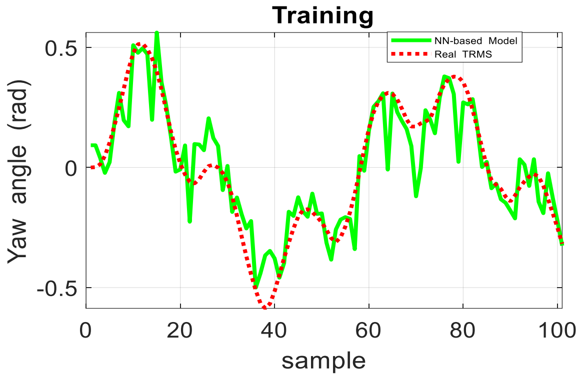

Fig 19. The yaw angle of the TRMS model when using the Resilient algorithm to train the network

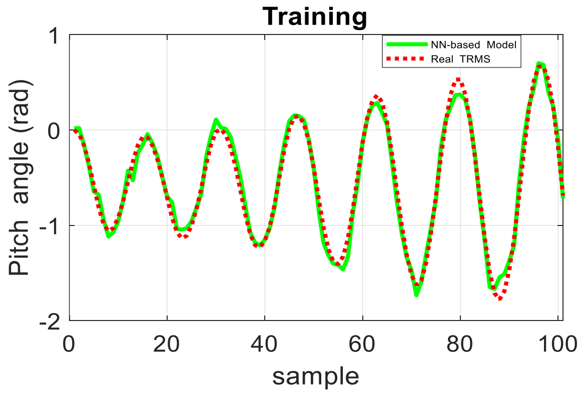

Fig 20. The pitch angle of the TRMS model when using the Resilient algorithm to train the network

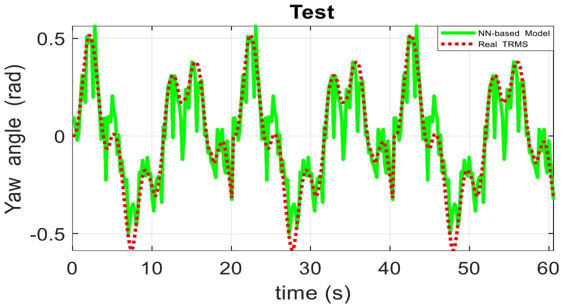

Fig 21. The yaw angle of the TRMS model when using the Resilient algorithm to test the network

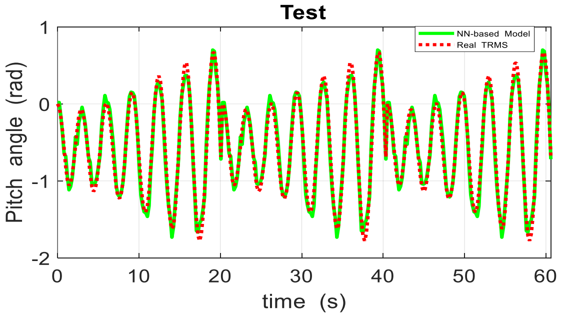

Fig 22. The pitch angle of the TRMS model when using the Resilient algorithm to test the network

Remaks

The mean squared error of the pitch angle in training process is 7.218312. 10^{-3}

The mean squared error of the pitch angle in testing process is 7.218312. 10^{-3}

The mean squared error of the yaw angle in training process is 1.184010. 10^{-2}

The mean squared error of the yaw angle in testing process is 1.184010. 10^{-2}

Remarks

The simulation results show that when using the same datasets with different training algorithms to recognize the TRMS object model will give the different mean squared error results. The real model and the recogniseable model of the training model and the test model decrease from 10^{-2} to 10^{-6}.

With these datasets, if using the training algorithms are Levenberg-Marquardt, BFGS Quasi-Newton, Bayesian Regularisation and One step Secant, the squared deviation about 10^{-4} or 10^{-6} are both acceptable.

IV.CONCLUSION

Building a model of the TRMS according to the black box model using Neural network through the simulations on Matlab and comparing with the real model show that the mean squared error between the training and the testing model in both yaw and pitch angles when using the same network training structure and datasets.

We can use the models using the training and testing algorithms as Levenberg-Marquardt, BFGS Quasi-Newton, Bayesian Regularisation and One step Secant because of small model bias, and the model trained and tested by the experts should not use Resilient algorithm because of large model error. In the next research, we will build a black box model or a gray box model thanks to GA identification in order to exploit and test the modern control methods to control TRMS objects.

ACKNOWLEDGEMENTS

This research was supported by the Thai Nguyen University of Technology, Viet Nam (http://www.tnut.edu.vn/), under grant number T2020-B72.

REFERENCES

[1]. Twin Rotor MIMO System 33-220 User Manual, 1998 (Feedback Instruments Limited, Crowborough, UK).

[2]. A. Rahideh, M.H. Shaheed, Mathematical dynamic modelling of a twin rotor multiple input–multiple output system, Proceedings of the IMechE, Part I. Journal of Systems and Control Engineering 221 (2007) 89–101.

[3]. Ahmad, S. M., Shaheed, M. H., Chipperfield, A. J., and Tokhi, M. O. Nonlinear modelling of a twin rotor MIMO system using radial basis function networks. IEEE National Aerospace and Electronics Conference, 2000, pp. 313–320.

[4]. Ahmad, S. M., Chipperfield, A. J., and Tokhi, M. O. Dynamic modelling and optimal control of a twin rotor MIMO system. IEEE National Aerospace and Electronics Conference, 2000, pp. 391–398.

[5]. Shaheed, M. H. Performance analysis of 4 types of conjugate gradient algorithm in the nonlinear dynamic modelling of a TRMS using feedforward neural networks. IEEE International Conference on Systems, man and cybernetics, 2004, pp. 5985–5990.

[6]. Huong T.M. Nguyen, Thai. Mai.T, Anh. Do.T.T, Lai Lai K. (2014), "Stabilization for Twin Rotor MIMO System based on BellMan's Dynamic Programming Method", *Journal of science and Technology of Thai Nguyen University*, pp. 161-165, issue. 14, vol. 128, 2014.

[7]. Huong. Nguyen.T.M, Thai. Mai.T, Chinh. Nguyen. H, Dung. Tran.T and Lai. Lai.K, *"Model Predictive Control for Twin Rotor MIMO system",* The University of Da Nang Journal of science and Technology, 12[85], pp. 39 – 42, 2014 .

[8]. Huong T.M. Nguyen, Thai Mai T., Lai Lai K. (2015), *"Model Predictive Control to get Desired Output with Infinite Predictive Horizon for Bilinear Continuous Systems*", International Journal of Mechanical Engineeringand Robotics Research, Vol. 4, No. 4, pp. 299 - 303.

[9]. Huong T.M. Nguyen, "*Interchannel Interference in While box Model of Twin Rotor MIMO System".* SSRG International Journal of Electrical and Electronics Engineering (SSRG-IJEEE), Volume 8 Issue 4, 36-40, April 2021.

[10]. Huong T.M. Nguyen, Mai Trung Thai[1], "*Black box Modeling of Twin Rotor MIMO System by Using Neural Network*". SSRG International Journal of Electrical and Electronics Engineering (SSRG-IJEEE), Volume 8 Issue 6, 15-22, June 2021.