AIAA Modeling and Simulation Technologies Conference and Exhibit
5-8 August 2002, Monterey, California

Investigation of Low Airspeed Estimation For Tiltrotor Aircraft Using Neural Networks

Naipei P. Bi David J. Haas
Naval Surface Warfare Center, Carderock Division
West Bethesda, Maryland

Abstract

The objective of this work is to investigate a neural network based approach to estimate the velocity and sideslip angle for a tiltrotor type aircraft in low airspeed flight. In order to evaluate neural network performance, a sensitivity study was conducted on various network design parameters that included the learning rate, the number of processing elements in the hidden layers, the size of training dataset and the number of inputs/outputs in the neural network. Flight test data from the V-22 aircraft was utilized for training and testing various neural network paradigms, including Back Propagation and Radial Basis Function networks. For the data analyzed, the minimum achieved Root Mean Square errors for the total, longitudinal and lateral velocities were 2.9, 2.0 and 4.57 knots respectively using a Back Propagation network and 2.1, 2.21 and 2.28 knots respectively using a Radial Basis Function. In addition, the sideslip angle error was less than 15 degrees for 90 percent of the total data utilized. Results of the sensitivity study also show that properly selecting the learning rate and the number of processing elements in the hidden layers can significantly improve the velocity estimation. Due to the limited amount of flight test data analyzed, further study is required to develop a comprehensive network that can provide accurate low airspeed indication over a full range of low airspeeds and azimuths.

Introduction

Tiltrotor aircraft offer the advantage of high speed flight over conventional helicopters. However, similar to conventional helicopters, tiltrotor aircraft also spend a significant portion of their flight time in the low airspeed regime such as during shipboard operations, take-off and landings. Low speed information is very useful in determining the fatigue life of aircraft components. The aircraft component fatigue life depends mainly upon the material strength, flight regime occurrence and the maximum loads in each flight regime.[1] Even though the flight regime occurrence can vary significantly from helicopter to helicopter, the low speed flight regime is often one of the most significant regimes due to the high number of occurrences and possible high flight loads in this regime. With an accurate low airspeed and sideslip angle indication, low airspeed flight can be more accurately classified, and therefore, the remaining aircraft component fatigue life can be better monitored and determined. In addition to flight regime classification, proper low airspeed information can also improve pilot situational awareness during low airspeed flight conditions.

Conventionally, a pressure device called a Pitot-static probe system measures helicopter airspeed. However, this system does not function well at airspeeds below approximately 40 knots. In this low airspeed flight regime the airflow environment around the fuselage is very complex due to the downwash from the main rotor as well as gusts blown up by the main rotor when flying near the ground**.** Additionally, the Pitot-static system is not very sensitive to pitch angle and/or yaw angle in low airspeed flight.

There has been interest for a long time in developing an alternative to the conventional Pitot-static system for indicating low airspeed. In the early 1970s, a true airspeed sensor for helicopters and other V/STOL aircrafts was developed using a system called J-Tec TAS system.[2] This system operated on the aerodynamic phenomenon of vortex shedding from a bluff body and was tested in the wind tunnel and on a CH-3E helicopter. According to [2], some attractive results were obtained. In the 1980s, some efforts were made on the estimation of low airspeed analytically based on measurements of various physical parameters

Presented at the AIAA Modeling and Simulation Technologies Conference & Exhibit,
Monterey, CA, 5-8 Aug 2002 (AIAA 2002-4867)

during flight. Faulkner and Buchner[3] developed an indirect method of airspeed estimation using the measurement of control inputs and helicopter response. A feasibility study was conducted and the difficulty noted in using their approach was to obtain the helicopter flapping angle in the rotating frame.

Recent efforts have been made in using neural networks to estimate the velocity and sideslip angle for conventional helicopters. Goff[4] used an artificial neural network based method to predict airspeed and sideslip angle in both steady state and transient conditions for the Lynx Mk9 helicopter. McCool *et al*[5-6] developed neural network based low airspeed sensors for the HH-60J and CH-46E helicopters that predicted airspeed and sideslip angle in the low airspeed environment. Their results have shown that airspeed can be predicted with an accuracy of approximately 5 knots for those helicopters and the sideslip angle can also be successfully predicted with reasonable accuracy in low airspeed flight regime. One of the reasons why the neural network based approach is attractive for the estimation of flight velocities is that neural networks are capable of providing a nonlinear mapping that represents the underlining relationships between measured physical parameters and desired targets when the correct network paradigm is selected and is properly trained.

To develop a neural network, extensive flight test data that contains the important physical traits of the aircraft are required for training, testing and validation of the neural network. The introduction of Health and Usage Monitoring System (HUMS) on helicopters makes a neural network approach for low airspeed estimation more practical because most, if not all of the network inputs are already being monitored. Therefore, with HUMS on board, the flight test data required for neural network development can be more easily collected.

While efforts have been made in using a neural network based approach to estimate velocity and sideslip angle in low airspeed flight for conventional helicopters, little effort has been made in investigating this approach for tiltrotor aircraft. The development of a neural network depends strongly on its specific application. A tiltrotor aircraft has unique characteristics that are quite different from a conventional helicopter and likewise the nonlinear mapping between control parameters and low airspeed will also be different. Therefore, additional investigation is required for developing a neural network specifically for tiltrotor aircraft.

The objective of this paper is to conduct an investigation of using a neural network based approach to develop a virtual sensor for estimating the total, longitudinal, and lateral velocities and sideslip angle for a tiltrotor type aircraft in low airspeed flight. In order to evaluate neural network performance, several neural network paradigms were evaluated and a sensitivity study was conducted on various network design parameters that included the learning rate, the number of processing elements in the hidden layers, the size of the training dataset and the number of inputs/outputs in the neural network.

Neural Network Development

There are several key aspects of developing a neural network that can successfully predict flight velocity and sideslip angle for a specific aircraft. These aspects include: (1) obtaining flight test data that contain the essential physical traits of the specific type of aircraft of interest and that covers all of the flight regimes of interest; (2) selecting proper datasets for training and testing the neural network; (3) determining the appropriate input and target (output) parameters; (4) selecting a proper neural network paradigm; (5) optimizing the network by varying network design parameters; and (6) testing and validation of the neural network.

Data Selection and Preparation

Flight test data used in this study are from the V-22 aircraft. A total of 15 different flights consisting of a total of 360 records were utilized. Most of the data was acquired at altitudes ranging between 19 and 38 feet from the ground, which is between one to two rotor radii in length, and at velocities less than 50 knots.

A key aspect of developing a successful neural network is to have representative data that is free of errors for training and testing the neural network. The training dataset must cover all flight regimes of interest and must contain all physical traits that are important to the specific application. This is because a neural network has

a strong capability for interpolation, but does not have the capability for extrapolating data.

The training dataset should be free of errors so that the neural network will not learn an improper relationship between the inputs and the targets. In order to be free of errors, a careful quality control assessment of the flight test data must be conducted, which includes: removing noise; discarding spikes; eliminating dead signals; and removal of other suspected bad data points. It is important to have a good understanding of the physical meaning of each of the input and output parameters on which data quality control is conducted. Flight test data quality control is critical in developing a proper neural network and can be very time consuming.

Figure 1 shows a polar plot of the flight data utilized in this study in terms of the azimuth and speed (0-360degrees, 0-40 knots) after quality control was conducted. It can be seen that there is a fairly good representation of the forward flight regime (0 degrees), but very little data for sideward or rearward flight regimes. In the current study, the training dataset was randomly selected with each record corresponding to 5 – 15 seconds of flight data. Table 1 lists the parameters used in current study.

Neural Network Paradigms

During this study, several neural network paradigms were explored. Results from two paradigms, the Back-Propagation Network (BPN) and the Radial Basis Function Network (RBFN), are discussed here. The BPN is one of the most commonly used networks. It consists of an input layer, an output layer and one or several hidden layers. The term back-propagation refers to two different things. First, it describes a method to calculate the derivatives of the network training error with respect to the weights through a clever application of the derivative chain-rule. Second, it describes a training algorithm, basically equivalent to gradient descent optimization, for using those derivatives to adjust the weights to minimize the error. [7]

The RBFN is another one of the most commonly used networks and is, in most general terms, any network that has an internal representation of hidden processing elements which are radially symmetric. The construction of a basic RBFN consists of three layers; an input layer, a hidden layer, and an output layer. These three layers play entirely different roles in the network. The input layer connects the network to the environment. The hidden layer applies a nonlinear transformation from the input space to the hidden space. The output layer is usually linear; supplying the network response to the activation pattern applied to the input layer.[8]

Results and Discussion

In the flight test data, there are three reference velocities, total velocity, V, Doppler longitudinal velocity, V_x and Doppler lateral or drift velocity, V_y, where V can be obtained by the vector addition of V_x and V_y. For the results, unless otherwise indicated, the training dataset includes all the input parameters listed in Table 1 and the training datasets used for the RBFN and the BPN are identical. All estimations of velocities and sideslip angles shown in this paper correspond to the test dataset; and the output parameters from networks include all three velocities, V, V_x and V_y.

Velocity Estimation

Figures 2 and 3 show the correlation between the measured total velocity and neural network estimation of the total V by the BPN and the RBFN respectively. A hidden layer with 200 processing elements (PEs) was used in both the BPN and the RBFN. The deviations from the 45-deg reference line represent the differences between measured flight test data and the values estimated by the neural networks. Data points on the 45-deg reference line represent exact matches. It can be seen from Figs. 2 & 3 that both networks captured the basic characteristics of total velocity. Figures 4-(a) to 5-(b) show the estimated longitudinal velocity V_x and lateral V_y for the BPN and the RBFN respectively. It can be seen from Fig 4-(a) to Fig. 5-(b) that the basic characteristics of V_x and V_y are also captured by both network paradigms. The Root Mean Square (RMS) error is used as a measurement of estimation accuracy and is defined as:

$$\text{RMS Error} = \sqrt{\frac{\sum_{i=1}^{n}(e_i - m_i)^2}{n}}$$

where e is the neural network estimated value, m is the measured flight test data, i is the record index, and n is the total number of the test dataset records used for testing the networks. Table 2 lists the corresponding RMS errors of V, V_x and V_y from the BPN and the RBFN.

For convenience of discussion in this paper, RMS_v, RMS_x and RMS_y are used respectively to represent the RMS error of V, V_x and V_y estimated by networks. It can be seen from Figs. 2 - 5 and Table 2 that in general, the RBFN gives a better estimation than the BPN does in this specific application. This is particularly true for V_y estimation where RMS_y from the RBFN is 2.28 knots while the RMS_y from the BPN is 4.57 knots. Recall that both the RBFN and the BPN use the same datasets for training and testing. Therefore, these results indicate that the performance of a network paradigm can strongly depend upon the specific application.

Sideslip Angle Estimations

The sideslip angle is defined as follows:

Sideslip angle in radians

$$= \sin^{-1}\left(V_y / V\right) \text{ for } V_x \geq 0$$

$$= -\pi - \sin^{-1}\left(V_y / V\right) \text{ for } V_x < 0 \text{ and } V_y < 0$$

$$= \pi - \sin^{-1}\left(V_y / V\right) \text{ for } V_x < 0 \text{ and } V_y \geq 0$$

The range of sideslip angle is from $-\pi$ radians (-180 degrees) to $+\pi$ radians (+180 degrees). It should be pointed out that the estimation of sideslip angle in the low airspeed regime is very challenging because small changes in one velocity component can result in a large change in sideslip angle near zero velocity. Also, the sideslip angle is ill-defined when airspeed magnitude goes to zero. Figure 6 shows the sideslip angle estimation when the measured total velocity magnitude $|V_m|$ is greater than 7.0 knots. The RBFN with 200 processing elements in the hidden layer was used. It can be seen from Fig. 6 that the basic characteristics of the sideslip angle was captured by the RBFN.

In order to have a better understanding of how the accuracy of the neural network for sideslip angle estimation varies with the velocity magnitude, Figure 7 shows the sideslip angle error distribution at different total velocities. As expected, at very low airspeeds, the sideslip angle error is large. However, as the total velocity increases from zero, the sideslip angle error decreases rapidly. When the velocity magnitude is greater than 15 knots, the sideslip angle error is less than 15 degrees for most data points. Figure 8 further shows the performance of the RBFN in the estimation of sideslip angle in terms of sideslip angle error against percent of total data points. It can be seen from Fig. 8 that about 38 percent of the total data have a sideslip angle error greater than 5 degrees and only 10 percent of total data have a sideslip angle error greater than 15 degrees. In other words, 90 percent of total data have an error less than 15 degrees, which indicates that sideslip angle estimation by the RBFN is quite good for the data analyzed.

Input Parameters

A sensitivity analysis was conducted to examine the effects of removing various input parameters on RMS error using the RBFN. Table 3 shows selected results. In this study, the baseline results were first obtained by using all of the input parameters listed in Table 1. Then, estimation of V_x and V_y was obtained by removing a specific parameter of interest to identify its effect. It can be seen from Table 3 that the Nacelle angle has a noticeable impact (5 to 8 percent) on RMS errors. The range of nacelle angle variation in the flight test data available was from 82.6 deg to 95.2 deg. Table 3 also shows the aircraft C.G. effect is significant (the range of C.G. travel in the data analyzed was less than 2 percent rotor radius). Radar altitude effect is also noticeable and indicates that without radar altitude, the RMS errors will be larger. Pedal was found to have the least effect of the parameters investigated.

Processing Elements

The number of processing elements (PEs) in the hidden layer of the RBFN determines the complexity of the neural network. The sensitivity of the neural network accuracy as a function of the number of PEs was investigated. The learning rates in the hidden layer and the output layer were not changed. Figure 9 shows that as the number of PEs in the hidden layer increases, RMS_v and RMS_y errors decrease monotonically. When the number of PEs was increased from 50

to 200, RMS_v and RMS_y errors reduced from 2.92 and 3.34 knots to 2.2 and 2.3 knots respectively. As the number of PEs continues to increase above 200, RMS_v and RMS_y reduce only slightly.

RMS_x, however, behaves differently from RMS_v and RMS_y. It fluctuates between 2.3 and 2.4 knots when the PE number changes from 50 to 200, then, it increases steadily as the number of PEs increases beyond 200. At a PE number of 500, the RMS_x reaches 2.9 knots, which is about 20 percent larger than the one with 200 PEs. These results suggest that if one network is used to estimate multiple outputs, some optimization is required for obtaining the best combination of target outputs with the minimum RMS errors. For the current study, 200 PEs in the hidden layer is a good comprise for having small RMS errors for V, V_x and V_y collectively. It should be noted that the time required for training the neural network may need to be considered in determining how many PEs to use in the hidden layer. The larger the PE number; the more time required for neural network training.

Number of Network Outputs

The number of network outputs was investigated to determine if more than one network should be used to estimate V, Vx and V_y. Figures 10 and 11 show RMS_v and RMS_y using single-output and three-output RBF networks. The training dataset and the testing dataset were identical for both cases. The number of PEs in the hidden layer was varied from 50 to 400 and the learning rates and other network design parameters, other than the number of outputs, were identical for both networks. The single output results shown in Figs. 10 and 11 correspond to a separate network to estimate each velocity and the three-output results correspond to one network used to estimate all three velocities at once. It is interesting to notice from Figs. 10 and 11 that for the current application, using separate networks to estimate individual velocity components has no advantage over using a single network to estimate all three velocities at once. RMS_x has similar characteristics as RMS_v and RMS_y (not shown).

Learning Rate

The learning rate corresponding to each layer in the neural network has a direct effect on the training process in terms of convergence and speed, and the optimum learning rate depends on the specific network and is related to the specific application. Therefore, a sensitivity study on learning rate was performed. Figure 12 shows the change in RMS_v RMS_x and RMS_y errors for different learning rates. The network used was a RBFN and the number of PEs in the hidden layer was 200. The corresponding learning rates for the output layer were always set at one half of the values used in the hidden layer, and all other network design parameters were not changed.

It is interesting to note that for the current application, all RMS_v RMS_x and RMS_y errors decrease when the learning rate increases from 0.05 to 1.5. However, when a learning rate of 2 is used, the RMS_x continues to decrease slightly, whereas RMS_v and RMS_y increase significantly and reach 3.2 and 3.9 knots respectively. These results suggest that with a large learning rate, the network goes through large oscillations during the training process. This type of oscillation should be avoided, and a learning rate less than 1.5 was selected for this application.

Training Data Size

The effect of the size of the training dataset is examined in Figure 13. It can be seen from Fig. 13 that for the current application, the size of training dataset should be greater than 4 percent of the total data available in order to minimize RMS errors for V, V_x and V_y. With the training data size equal to two percent of the total data, the RMS error increased approximately 40 to 60 percent compared to the results using 4 percent of the total data. It can also be seen from Fig. 13 that using a training dataset that is too large does not help in improving the estimation. However, the time required to train the neural network will increase significantly when the size of the training dataset increases. It should be pointed out that the selection of the training data size depends also on whether the training dataset includes all of the important physical traits of the specific aircraft.

Total Velocity Estimation

There are two ways to estimate the total velocity, V. One is to estimate it directly from the neural network by designating it as one of the outputs. The other way is to calculate V by the vector addition of the neural network estimated V_x and V_y. In additional to the RMS_v which represents the RMS error introduced by direct estimation as an output of the neural network, RMS_{vcal} is introduced to represent the RMS error using the total velocity calculated by the vector addition of V_x and V_y. Figure 14 shows clearly that for the whole range of the number of PEs used, the RMS_{vcal} error is larger than the RMS_v error. For example, at a PE number of 200, the RMS_{vcal} is about 4.7 knots while the RMS_v is only about 2.2 knots. The explanation for the larger RMS_{vcal} is that when V_x and V_y are used to calculate the total velocity, the errors in both V_x and V_y are compounded in the total V, which can be larger than the error when V is estimated directly by the neural network.

Time HistoryEstimation

An assessment was made on the comparison of the time history of longitudinal and lateral velocities between the flight test data and the estimation by the RBFN. Sample plots are shown in figures 15 and 16. The range of V_x is from 17 knots to 6 knots and the range of V_y is from -23 knots to -3 knots. Generally, the estimated values agree well with flight test data for this specific case. However, more evaluation is required to have a better understanding of the performance of the neural network in estimating the time history of V_x and V_y in different flight conditions.

Summary & Conclusions

An investigation has been conducted to examine a neural network based approach to develop a virtual sensor for estimating the longitudinal, lateral and total velocities for a tiltrotor type aircraft in low airspeed flight. Sensitivity studies were conducted on various network design parameters that included the learning rate, the number of processing elements in the hidden layers, the size of training dataset and the number of network inputs/outputs in order to evaluate neural network performance. Flight test data from the V-22 aircraft were utilized for training and testing various neural network paradigms, including Back Propagation and Radial Basis Function networks. The following conclusions are made from the current study:

(1) The RMS errors of V, V_x and V_y are 2.9, 2.0 and 4.57 knots respectively using the BPN and 2.1, 2.21 and 2.28 knots respectively using the RBFN, which are quite satisfactory. The differences between the RMS errors for these two paradigms indicate that the performance of a paradigm can strongly depend upon the specific application.

(2) Sideslip angle estimation is within a reasonable accuracy with sideslip angle errors less than 15 degrees for 90 percent of the data points analyzed.

(3) The results show that the learning rate used in the networks can significantly affect RMS errors.

(4) The number of processing elements in the hidden layer should be carefully selected to minimize RMS errors for all three velocities.

(5) For the current application, using separate networks to estimate individual velocity components had no advantage over using a single network to estimate all three velocities at once.

(6) Training data size can affect the RMS errors if it is too small, however, no effect was seen when its size is sufficiently large. In this study, four percent of the total data was used.

(7) Input parameters must include all major physical traits of the specific application to minimize the RMS errors.

Due to the limited amount of flight test data analyzed, additional study is recommended to examine a greater range of airspeed and azimuth.

Acknowledgements

The authors wish to thank Mr. Kenneth Reader of NSWCCD for constructive suggestions.

References:

1. W. Martin, G. Collingwood, and G. Barndt, "Structure Life Monitoring of the V-22," Presented at the American Helicopter Society 55th Annual Forum, Montreal, Quebec, Canada, May 25 – May 27, 1999.
2. K.W. McElreath, "True Airspeed Sensor For V/STOL Aircraft," AFFDL-TR-72-131, December 1972.
3. Faulker, A.J. and Buchner, F., "Flight Investigation of a Helicopter Low Airspeed Estimation System Based on Measurement of Control Parameters," Proceedings of the Sixth European Rotorcraft and Powered Lift Aircraft Forum, Sept. 1980.
4. Goff, D.A., Thomas, S., Jones P. and Massey, C., "Predicting Airspeed and Sideslip Angle using an Artificial Neural Network," Presented at the American Helicopter Society 55th Annual Forum, Montreal, Quebec, Canada, may 25 – May 27, 1999.
5. McCool, K.M., Morales, M. A. and Haas, D. J., "Neural Network Based Low Airspeed Sensor**,"** Proceedings of AHS 56th Annual Forum, Virginia Beach, VA, May, 2000
6. McCool, K.M. and Haas, D., "Prediction of helicopter Airspeed and Sideslip Angle in the Low Speed Environment**,"** Proceedings of AHS 53rd Annual Forum, Virginia Beach, VA, April-May, 1997
7. Russell D Reed and Robert J marks II, *Neural Smithing – Supervised Learning in Feedforward Artificial Neural Networks*. The MIT Press, 1999.
8. Simon Haykin, *Neural Networks – A Comprehensive Foundatiion*, Second Edition, Prentice Hall, 1999.

Table 1: Neural Network Input Parameters

No.	Parameter
1.	Engine torque
2.	Longitudinal stick
3.	Lateral stick
4.	Weight
5.	Radar altitude
6.	Normal acceleration
7.	Longitudinal acceleration
8.	Lateral acceleration
9.	Pedal
10.	Pitch
11.	Roll
12.	Aircraft C.G.
13.	Nacelle angle
14.	Pitch acceleration
15.	Roll acceleration
16.	Yaw acceleration
17.	Pitch rate
18.	Roll rate
19.	Yaw rate

Table 2: RMS Errors for BPN and RBFN

Parameter	Network	RMS error (kts)
V	BPN	2.90
V	RBFN	2.10
V_x	BPN	2.00
V_x	RBFN	2.21
V_y	BPN	4.57
V_y	RBFN	2.28

Table 3: Sensitivity to Network Input Parameters

Parameters	RMS_x	RMS_y
Baseline (all parameters)	2.28	2.21
Less Nacelle angle	2.39	2.39
Less C.G.	2.92	2.45
Less Pedal	2.35	2.37
Less Radar Altitude	2.35	2.47

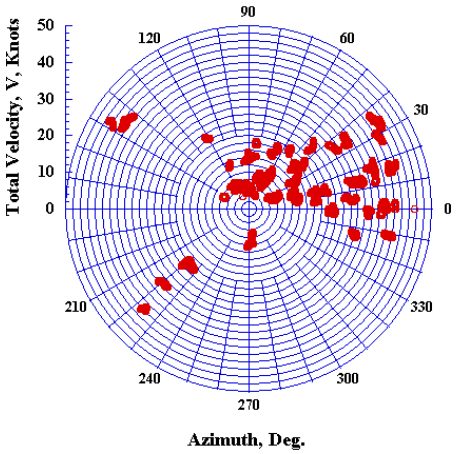

Figure 1: Polar plot of flight test data

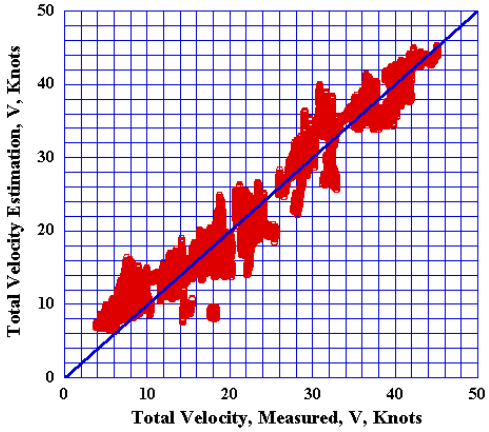

Figure 2: Total velocity V comparison between measured and estimation by BPN

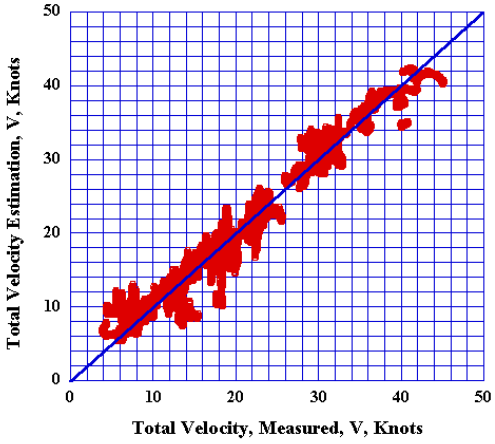

Figure 3: Total velocity V comparison between measured and estimation by RBFN

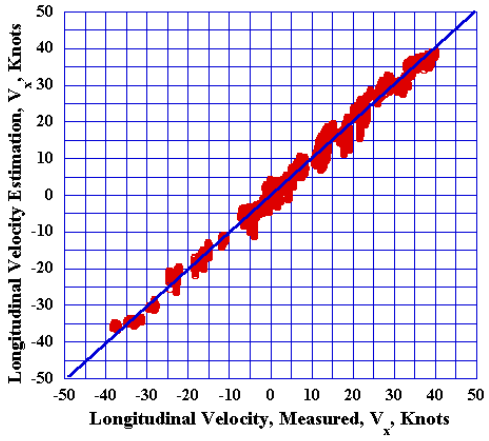

Figure 4-(a): Longitudinal velocity V_x comparison between measured and estimation by BPN

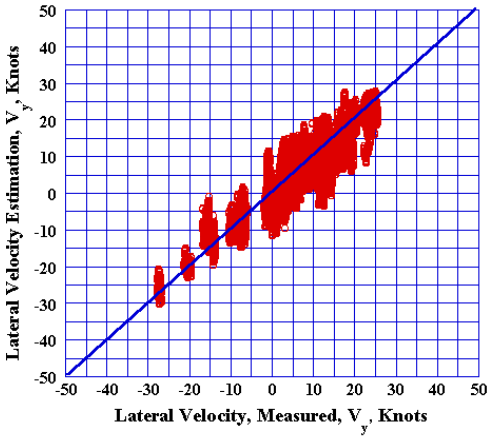

Figure 4-(b): Lateral velocity V_y comparison between measured and estimation by BPN

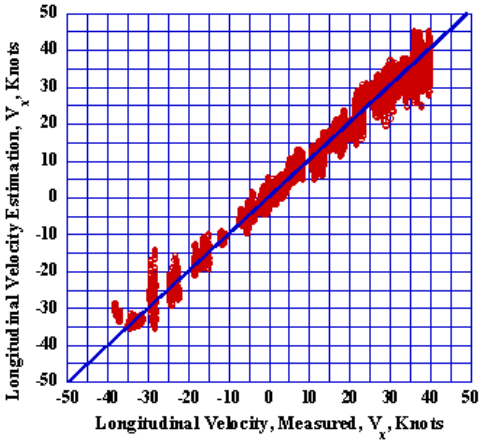

Figure 5-(a): Longitudinal velocity V_x comparison between measured and estimation by RBFN

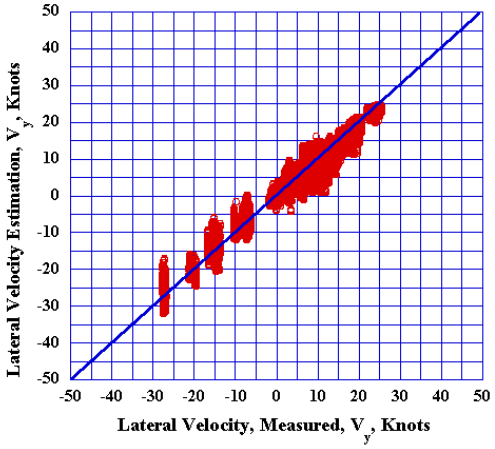

Figure 5-(b): Lateral velocity comparison between measured and estimation by RBFN

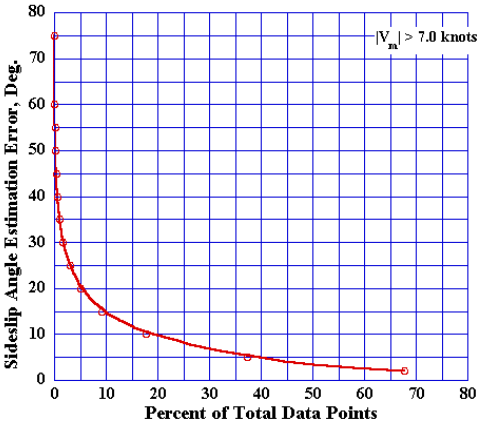

Figure 8: Sideslip angle estimation error distribution by RBFN

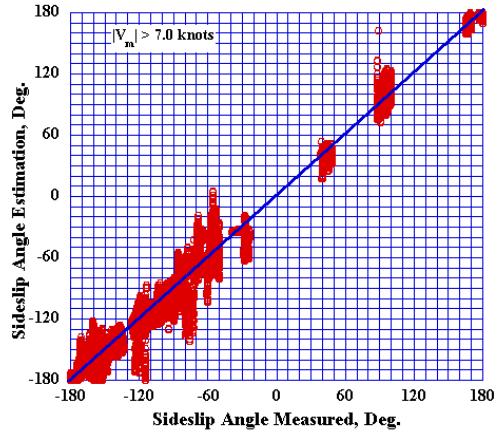

Figure 6: Sideslip angle comparison between measured and estimation by RBFN

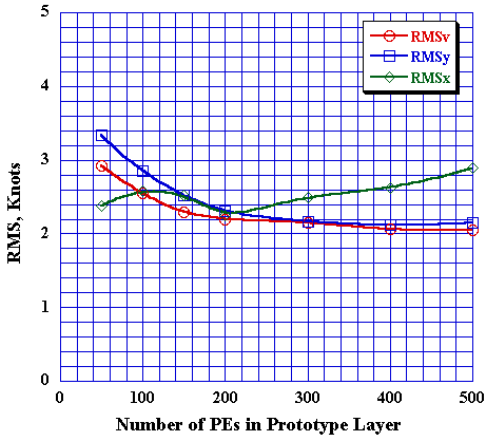

Figure 9: Effects of number of PEs in hidden layer of RBFN on RMS error

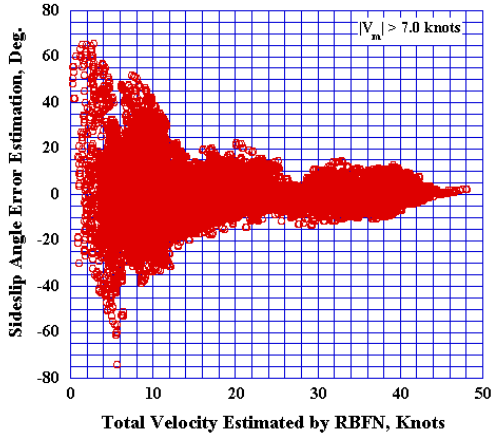

Figure 7: Sideslip angle error vs. total velocity estimation by RBFN

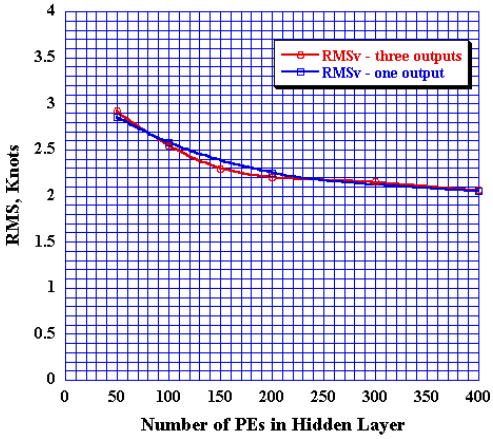

Figure 10: Comparison of RMS_v between single output and three output networks by RBFN

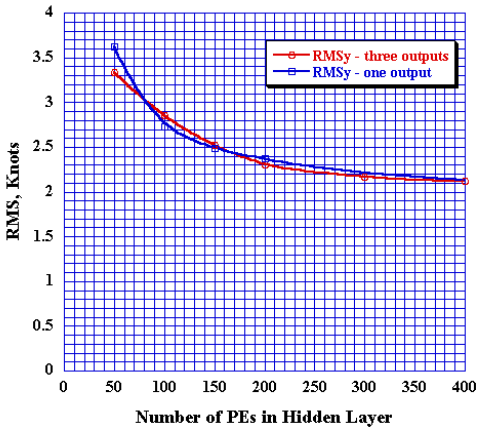

Figure 11: Comparison of RMS_y between single output and three output networks by RBFN

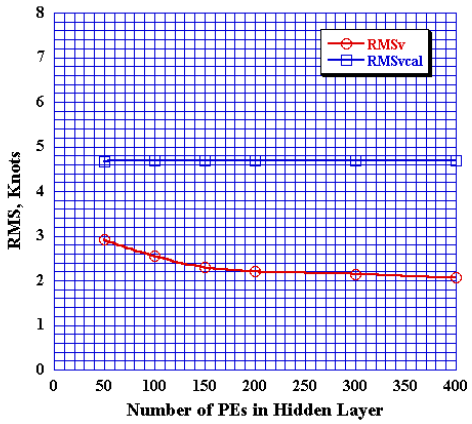

Figure 14: Comparison of RMS_v and RMS_{vcal} by RBFN

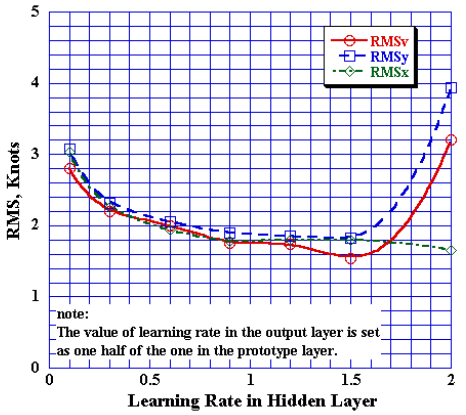

Figure 12: Effects of learning rate on RMS values by RBFN

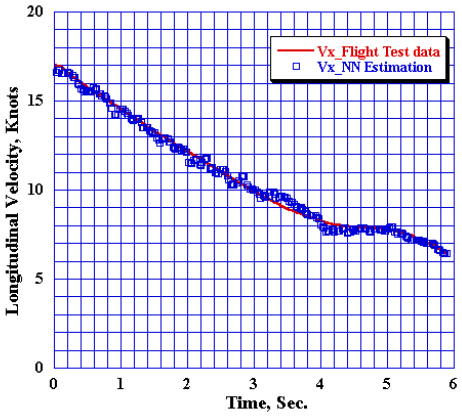

Figure 15: Longitudinal velocity time history estimation by RBFN

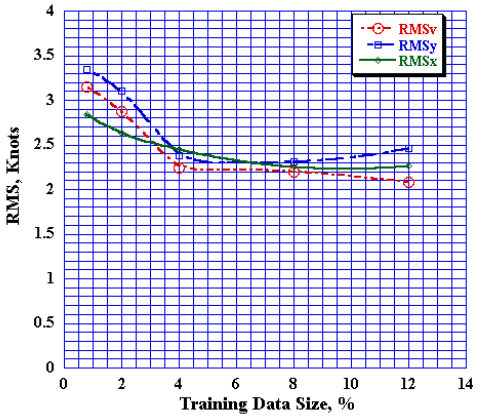

Figure 13: Effects of training data size on RMS by RBFN

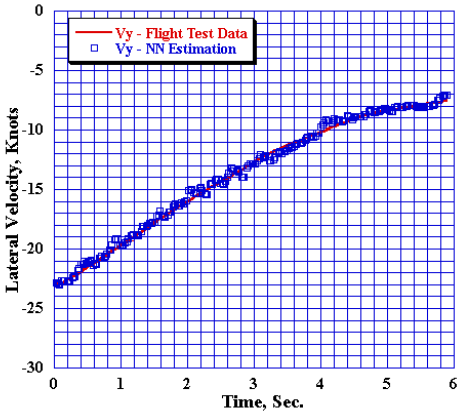

Figure 16: Lateral velocity time history estimation by RBFN