Training Data Requirement for a Neural Network to Predict Aerodynamic Coefficients

T. Rajkumar and Jorge Bardina
SAIC@ NASA Ames Research Center, Moffett Field, California, USA 94035
NASA Ames Research Center, Moffett Field, California, USA 94035

ABSTRACT

Basic aerodynamic coefficients are modeled as functions of angle of attack, speed brake deflection angle, Mach number, and side slip angle. Most of the aerodynamic parameters can be well-fitted using polynomial functions. We previously demonstrated that a neural network is a fast, reliable way of predicting aerodynamic coefficients. We encountered few under fitted and/or over fitted results during prediction. The training data for the neural network are derived from wind tunnel test measurements and numerical simulations. The basic questions that arise are: how many training data points are required to produce an efficient neural network prediction, and which type of transfer functions should be used between the input-hidden layer and hidden-output layer. In this paper, a comparative study of the efficiency of neural network prediction based on different transfer functions and training dataset sizes is presented. The results of the neural network prediction reflect the sensitivity of the architecture, transfer functions, and training dataset size.

Keywords: Neural network, aerodynamic coefficients, design of network architecture, training data requirements

1. INTRODUCTION

The Virtual Flight Rapid Integration Test Environment (VF-RITE)[5] project was initiated to develop a process and infrastructure to facilitate the use of computational fluid dynamics (CFD), wind tunnel, and/or flight data in a real-time, piloted flight simulation and quickly return performance knowledge to the design team. The goal of this project was to develop a design environment that merges these technologies and data in an effort to improve the current design methodologies and reduce the design cycle time of new vehicles. Although many sources of data are available, the aerodynamic database for different vehicles are captured and distributed via DARWIN[6,7,13,15], a NASA Ames Research Center developed collaborative distributed portal system. Preliminary simulation studies can be conducted to identify problems or deficiencies in aerodynamic performance and vehicle stability and control. These problems can be addressed and corrected during the initial phases of the design. This environment also allows for the opportunity to develop preliminary control systems early on in the vehicle development.

Mathematical aerodynamic models derived from CFD data for air/space vehicle configurations were generated and integrated into the NASA Ames Vertical Motion Simulator (VMS) facility. Selection and integration of different CFD datasets was initially done by hand to determine all the steps involved in the modeling process. The datasets contained different levels of physics fidelity which warranted careful integration and generation of the data. Information technologies were then developed to aid in the decision making (determine data and models) and automation of the process. Finally, the new math models were implemented and evaluated during a simulation experiment in the VMS facility. The handling qualities of each configuration were evaluated using Cooper-Harper ratings[12] during the approach and landing phases. The data and aerodynamic model were supplied to the VMS software using a dataset called the "function table". The function table is used by the VMS operator to program the desired motion. The information in the function table in conjunction with the aerodynamic model, which derives some of its information from the CAD model, provides the complete basis for the running of the full motion simulator. As more data becomes available, flight control system designs may need to be modified to allow for minor nonlinearities in control effects.

The governing equations behind a simple function table are primarily linear and generally do not include compressibility effects or other higher order terms. This level of modeling was considered adequate for approach and landing simulations of a vehicle. The vehicle and data considered in this paper is the Space Shuttle Vehicle (SSV). The function

table consists of data for all Mach number, angle of attack (Alpha), side slip angle (Beta), and speed brake deflection angle. The collection of data for the above mentioned parameters can be very expensive for any given vehicle. In order to rapidly turn around a vehicle design and to study the impacts of lateral and longitudinal coefficients[18,20], a neural network approach was adopted to predict aerodynamic coefficients. Artificial neural networks[9] (ANN) are employed, such that once properly trained can provide a means of rapidly evaluating the impact of design changes.

2. NEED FOR NEURAL NETWORK

Aerodynamic forces and moments coefficients can be generated using CFD codes of varying physics fidelity such as Vortex lattice method (e.g., Vorview[16]), Euler (e.g., AIRPLANE[23] and FlowCart[29]), and Navier-Stokes[1,10,11] (e.g., Overflow-D[2]). Navier-Stokes simulations are the most time consuming of these methods but have the greatest fidelity. Depending on the complexity of the geometry, the creation of a viscous mesh around a three dimensional vehicle may take a grid generation specialist several days or even months to complete. Any modification to the geometry (such as changing the deflection angle on a single control surface) may also result in several weeks of additional work. The computer time required to run a single steady state case on a parallel super computer such as the SGI Origin 2000 cluster is usually on the order of hours. Consequently, it is not currently feasible to use a Navier-Stokes solver for all aerodynamic data generation or for initial design calculations. During conceptual design, numerous changes to a vehicle configuration necessitates a fast and robust flow solver like Vorview. Once the design space is narrowed down to several configurations, an optimizer coupled with an Euler solver (AIRPLANE) can be used to maximize the performance characteristics of the vehicle. The optimized configuration is then modeled using a Navier-Stokes code such as Overflow-D to predict more realistic aerodynamic data.

Wind tunnel tests of scaled models have traditionally been used to characterize aerodynamic coefficients of an air vehicle. Wind tunnel data, in original form, are sometimes unsuitable for use in piloted simulations because data obtained from different wind tunnels for different scale models of the same vehicle are not always consistent. Traditionally, the approach considered to describe the aerodynamics of a vehicle is to fit, wherever possible, a polynomial function for each aerodynamic coefficient[14]. This ensures a smooth continuous function and removes some of the scatter from the wind tunnel measurements. Fitting a smooth function through the wind tunnel data results in smooth derivatives of the data which are important in performing stability analyses. Also, measurements of the same coefficient from two different wind tunnels are usually taken at dissimilar values of angle of attack and side slip angle, so some means of reconciling these dissimilar sets of raw data is needed. The curve fitting method used to generate the parameters for each polynomial description is an unweighted least squares algorithm.

The collection of aerodynamic data can be expensive from wind tunnel tests or numerical simulations. Wind tunnel testing can be costly due to complex model fabrication, high personnel overhead, and intensive power utilization. Numerical simulation of complex vehicles on the wide range of conditions required for flight simulation requires steady and unsteady data. Steady aerodynamic data at low Mach numbers and angles of attack may be obtained with simpler Euler codes. Steady data for a vehicle with zones of flow separation present, usually at higher angles of attack, require Navier-Stokes simulations which are costly due to the large processing time required to attain convergence. Preliminary unsteady data may be predicted with simpler methods based on correlations and vortex methods[17] however, accurate prediction of the dynamic coefficients requires complex and time consuming numerical simulations[19]. Using a neural network approach, data from experimental and numerical simulation can be fused together to generate an efficient archive database. The neural network could also be used to generate the function table for vehicles of similar geometry without conducting additional wind tunnel tests or CFD simulations. In general a neural network performs very well in a classification and prediction arena within the data range being utilized during the training phase. The dataset to train the neural network can come from either Navier-Stokes simulations, from wind tunnel test measurements, or a combination of both.

3. SELECTION OF DATASET AND NEURAL NETWORK

If a neural network is going to be effective[26], the training dataset must be complete enough to satisfy the following goals:

- Every group must be represented: The training dataset consists of several subgroups, each having its own central tendency toward a particular pattern. All of these patterns must be represented.
- Within each class, statistical variation must be adequately represented: The range of data presented to the neural network must represent the entire range of data with noise included. In certain cases, only lower and upper bound data with few scattered data points in between the range will be available. In that case, it is acceptable to have sparse dataset.

When a subclass lies near a decision boundary, it is important to use a large dataset in order to avoid learning patterns of noise, which are in common among a large fraction of the representatives of that subclass. In general, proper selection of the training dataset leads to the success of the neural network classification or prediction.

Larger networks require large training datasets. Over fitting is more likely when the model is large. When designing a neural network and selecting the training dataset for a given problem, the designer has to explore answers for the following questions:

- Are there any transformations required for the training dataset?
- Are there enough sample representations in all sub-classes?
- How many neural network architecture layers are needed? (three layer or four layer network)
- How many hidden neurons?
- What type of transfer functions?
- How long is training required for the network?

Real time problems can be solved using three layer neural networks with a few hidden neurons. In our previous paper[27,28], we concluded that a three layer neural network with a Levenberg-Marquardt training algorithm was sufficient to capture the relationship between angle of attack, and Mach number as inputs and coefficient of lift as output. The transfer functions used for predicting the coefficient of lift were pure linear, hyperbolic tangent and sigmoid. The training dataset used in the previous paper[27,28] had 233 data pairs. This study investigates how many data samples and which neural network architecture produces the best prediction of aerodynamic coefficients.

The following sections describe the basic definitions of aerodynamic coefficients[22] (Fig. 1) that will be predicted by the neural network. Aerodynamic coefficients are parameters in the aerodynamic model used to compute the forces and moments acting on an air vehicle. The aerodynamic coefficients are categorized into longitudinal and lateral groups.

Longitudinal Aerodynamic Coefficients:

Lift force coefficient (C_L): a measure of the aerodynamic force that is perpendicular to the relative wind and opposes the pull of gravity. It is function of the vehicle speed, angle of attack, and dependent on the aircraft geometry. C_L is usually determined from wind tunnel experiments, CFD codes as well as semi-empirical methods. It can be estimated fairly accurately for most aircraft.

Drag force coefficient (C_D): a measure of the force that acts in the opposite direction a vehicle travels. The drag coefficient contains not only the complex dependencies of object shape and inclination, but also the effects of air viscosity and compressibility. To correctly use the drag coefficient, it must be guaranteed that the viscosity and compressibility effects are the same between the measured case and the predicted case, otherwise the prediction will be inaccurate. For very low speeds (< 200 mph) the compressibility effects are negligible. At higher speeds, it becomes important to match Mach numbers between the two cases. (Mach number is the ratio of the vehicle velocity to the speed of sound.)

Pitching moment coefficient (C_P): a measure of the moment that rotates the aircraft nose up and down about the y-axis, a motion known as pitch. Pitch control is typically accomplished using an elevator on the horizontal tail. The pitching moment is usually negative when measured about the aerodynamic center, implying a nose-down moment.

Lateral Aerodynamic Coefficients:

Side force coefficient (C_S): is a measure of the aerodynamic force acting perpendicular the vehicle's body axis in the y-direction.

Rolling moment coefficient (C_R): a measure of the moment that rotates the wing tips up and down about the x-axis, a motion known as roll. Roll control is usually provided using ailerons located at each wingtip.

Yawing moment coefficient (C_Y): a measure of the moment that rotates the nose left and right about the z-axis, a motion known as yaw. Yaw control is most often accomplished using a rudder located on the vertical tail.

The general equations relating aerodynamic forces and moments to their corresponding aerodynamic coefficients are as follows:

$$\text{Force Coefficient} = \frac{\text{Force}}{(\text{Dynamic Pressure})\,(\text{Reference Area})}$$

$$\text{Moment Coefficient} = \frac{\text{Moment}}{(\text{Dynamic Pressure})\,(\text{Reference Area})\,(\text{Reference Length})}$$

An aerodynamic coefficient basically depends on angle of attack, side slip angle, and Mach number with minor increments due to other factors (e.g., viscosity). It can be expressed as a polynomial function of these parameters. Aerodynamic coefficients also vary depending on the geometry of the vehicle. The inputs for the neural network considered in this study are angle of attack (Alpha), side slip angle (Beta), speed brake deflection angle and Mach number. The general aerodynamic parameters and range of values for the Space Shuttle Vehicle configuration used in this study are tabulated in Table 1. The outputs from the neural network are the coefficients of the aerodynamic model according to the aerodynamic coefficient equations. A good training dataset for a particular vehicle type and geometry is usually selected from a wind tunnel test. Sometimes if such a dataset is not available from wind tunnels tests, a good training dataset can be derived from Euler, Navier Stokes, Vortex lattice method[16] numerical simulations. This dataset consists of a comprehensive input and output tupple for an entire parameter space. After a training dataset is selected, one must determine the type of neural network architecture and transfer functions that will be used to interpolate the data. The next section will discuss the selection procedure of the neural network architecture and transfer functions used in this work.

Parameters	Ranges of values
Angle of attack (degrees)	$-10 < \alpha < 60$
Side angle (degrees)	$0 < \beta < 10$
Mach number	$0.25 \leq M \leq 0.98$
Speed brake deflection angle (degrees)	$0 < \delta_{speedbrake} < 87.2$

Table 1 Range of Values Involved in Aerodynamic Coefficients

4. SELECTION OF TRANSFER FUNCTIONS

The transfer function when applied to the net input of a neuron, determines the output of that neuron. Its domain must generally be real numbers although there is no theoretical limit to what the net input can be. The range of the transfer function (i.e., values it can output) is usually limited. The most common limits are -1 to +1. The majority of neural network models use a sigmoid (S-shaped) transfer function[26]. A sigmoid function is loosely defined as a continuous, non linear function whose domain is the real number set, whose first derivative is always positive, and whose range is bounded. Other possible transfer functions are hyperbolic tangent and pure linear. [Kalman and Kwansy, 1992][3] make a very eloquent case for choosing the hyperbolic tangent function. They proposed and justified four criteria that an ideal transfer function should meet and showed that only the hyperbolic tangent function met all four. The sigmoid function never reaches its theoretical maximum and minimum. For example, neurons that use the sigmoid function should be

considered fully activated at around 0.9 and turned off at about 0.1 or so. Three types of transfer functions: linear, sigmoid, and hyperbolic tangent were compared in this study to determine the most efficient neural network prediction.

5. ANALYSIS OF NEURAL NETWORK ARCHITECTURE

The process of defining an appropriate neural network architecture[21,25] can be divided into the following categories: (i) determining the type of neural network (whether three layer or four layer, etc.); (ii) determining the number of hidden neurons; (iii) selecting the type of transfer functions[8]; (iv) devising a training algorithm; and (v) checking for over and/or under fitting of the results and validation of neural network output[24]. If a function consists of a finite number of points, a three layer neural network is capable of learning that function. For this study, the type of neural network considered is a three layer neural network with an input layer, a hidden layer and an output layer. For the case of the Space Shuttle Vehicle, the inputs for the longitudinal aerodynamic coefficients are angle of attack (Alpha), Mach number, and speed brake deflection angle. The inputs for the lateral aerodynamic coefficients are angle of attack (Alpha), side slip angle (Beta), Mach number, and speed brake deflection angle. The training data available for the longitudinal and lateral aerodynamic coefficients for the Space Shuttle Vehicle are given in Table 2.

Input Parameters	Available data	Total
Longitudinal Aerodynamic Coefficients (C_L, C_D, C_M)		
Angle of attack (degrees) (for 0.25-0.4 Mach number)	-10.000, -7.500, -5.000, -2.500, 0.000, 2.500, 5.000, 7.500, 10.000, 12.500, 15.000, 17.500, 20.000, 22.500, 25.000	15
Angle of attack (degrees) (for 0.6-0.98 Mach number)	-10.000, -7.500, -5.000, -2.500, 0.000, 2.500, 5.000, 7.500, 10.000, 12.500, 15.000, 17.500, 20.000, 22.500, 25.000, 27.500, 30.000, 32.500, 35.000, 37.500, 40.000, 42.500, 45.000, 47.500, 50.000, 52.500, 55.000, 57.500, 60.000	29
Mach number	0.250, 0.400, 0.600, 0.800, 0.850, 0.900, 0.920, 0.950, 0.980	9
Speed brake deflection angle (degrees)	0.000, 15.000, 25.000, 40.000, 55.000, 70.000, 87.200	7
Lateral Aerodynamic Coefficients (C_S, C_R, C_Y)		
Angle of attack (degrees) (for 0.25-0.4 Mach number)	Same as above	15
Angle of attack (degrees) (for 0.6-0.98 Mach number)	Same as above	29
Sideslip angle (degrees)	0.000, 1.000, 2.000, 3.000, 4.000, 6.000, 8.000, 10.000	8
Mach number	Same as above	9
Speed brake deflection angle (degrees)	0.000, 25.000, 55.000, 87.200	4

Table 2 Data Availability for Aerodynamic Coefficients

The prediction of aerodynamic coefficients using the neural network is classified into two categories: longitudinal and lateral aerodynamic coefficients. In longitudinal aerodynamic coefficient prediction, the neural network architecture design is based on neural network training experiments using the coefficient of lift and then that architecture is duplicated for the other two aerodynamic coefficients. For the category of lateral aerodynamic coefficients, the coefficient of rolling moment is used for the primary experiment. The objective of these experiments is to find the appropriate number of hidden neurons and the most suitable transfer functions between various layers of the neural network. Once the architecture is sorted out, neural network parameters can be tuned to bring about the necessary prediction accuracy.

The maximum number of iterations for training is limited to 1000 cycles. Beyond 1000 cycles, the convergence rate tends to be extremely slow. During the training phase, the error of tolerance to reach the goal is set to 0.001. The learning and momentum rates are selected appropriately to get faster convergence of the network. The input and output values are scaled to the range [0.1, 0.9] to ensure that the output will lie within the output region of the nonlinear

sigmoid transfer function. Presentable variable values lie between 0.1 and 0.9 inclusive. The scaling is performed using the following equation:

$$A = r(V - V_{min}) + A_{min}$$

$$r = \frac{A_{max} - A_{min}}{V_{max} - V_{min}}$$

v – Observed Variable   A – Presentable Variable

Once the scaled training dataset is prepared, it is ready for neural network training. The Levenberg-Marquardt method[21] for solving the optimization was selected for backpropagation[4] training. It was selected due to its guaranteed convergence to a local minima and its numerical robustness. The criteria to stop neural network training are:

- The maximum number of epochs (repetitions, i.e., 1000 iterations) is reached.
- Performance has been minimized to the goal.
- The performance gradient falls below the minimum gradient limit.

Once the neural network is trained with a suitable dataset, it is ready to predict the aerodynamic coefficients for a new dataset that it was not exposed to during the training phase.

6. RESULTS

As mentioned in the previous section, the coefficient of lift and rolling moment coefficient predictions were the pilot experiments which will be repeated for the other aerodynamic coefficients. The number of available data pairs was 1631 for longitudinal aerodynamic coefficients and 7456 for lateral aerodynamic coefficients. In our previous paper[27], the training dataset was classified into two datasets based on low and high Mach number ranges. In this current study, the training and prediction datasets were considered on the basis of speed brake deflection angle. Longitudinal aerodynamic coefficients datasets were classified based on 7 speed brake deflection angle datasets in which each dataset has 233 data pairs. The speed brake deflection angle was considered as the discriminator due to the low number of clusters present in longitudinal aerodynamic coefficients. For lateral aerodynamic coefficients, side slip angle (beta) was considered as the discriminator. The transfer functions adopted for the input-hidden layer were sigmoid (logsig) and hyper tangent (tanh) functions. For the hidden-output layer the transfer functions tested were sigmoid (logsig) and pure linear function (purelin).

The three inputs to the neural network for predicting the longitudinal aerodynamic coefficient were angle of attack, speed brake deflection angle and Mach number. The number of hidden neurons did not exceed beyond 15 (i.e., 5 times of input number of neurons) since the function which we were trying to fit was very simple. The dataset was divided into a training dataset and a prediction dataset. Around 66 different combinatorial experiments were conducted to determine the best neural network architecture. From the results of the 66 experiments, 4 architectures were selected, based on the accuracy of predicted results as presented in Table 3. The number of predicted results that produced more than 100 % over fitted values was the criteria used for selecting the best architecture. The four best architectures were adopted for predicting coefficient of drag and pitching moment coefficient.

Architecture (NNL)	Transfer function		Training data pairs	Prediction data pairs	Number of iterations for convergence			No. of data points exceeded 100 % over fitting		
	Input-Hidden	Hidden-Output			CL	CD	Cm	CL	CD	Cm
3-10-1 (1st)	logsig	purelin	816	816	5	**5**	45	51	**10**	48
3-10-1 (2nd)	tansig	purelin	326	1306	5	6	54	38	41	411
3-10-1 (3rd)	logsig	logsig	816	816	**4**	6	154	**28**	32	**29**
3-10-1 (4th)	logsig	logsig	544	1087	5	10	**23**	75	35	90

Table 3 Neural Network Configuration to Predict Coefficient of Lift (Bold numbers indicate fast convergence and best fit)

The training data pairs were always selected from the lower speed brake deflection angle clusters. The training dataset presented to the neural network consisted of speed brake deflection angles up to 25 degrees and a few data pairs at 40 degrees (816 data pairs). The higher speed brake deflection angle cluster was used as the prediction dataset. We provided a portion of the training data starting from 8.3% (136 data pairs) to 50% (816 data pairs) to test the neural network for prediction. In figure 2, "1st NN prediction" refers to the first row in the table 3 with specific neural network architecture and training data size. At lower angles of attack, the first three neural network architectures performance are reasonable in comparison to the 4th neural network architecture. The reason for the poor performance of the 4th neural network architecture was due to lower number of training data pairs. Similarly, the 2nd neural network architecture did not perform well at higher angles of attack due to lower number of training cases. The first and third neural network architectures performed well compared to the other neural network architectures for coefficient of lift prediction.

The same four architectures were repeated to predict the coefficient of drag. Except for the first neural network architecture, the others performed reasonably well (Figure 3). The third neural network architecture performed the best and it nearly replicated the original pattern. The coefficient of drag function behaves like an 'S' function, so the sigmoid function was the appropriate function to choose between layers. Once again, the four network architectures were used to predict pitching moment coefficient. The 2nd neural network architecture did not perform very well for the coefficient of pitching moment (Figure 4). The first and third neural network architectures produced a reasonable fit for the coefficient of pitching moment. It was also observed that at least 50% of the training cases were required to form a good training dataset in order to reasonably predict the longitudinal aerodynamic coefficients. Based on this analysis with the given data, a three layer neural network architecture with sigmoid transfer function produced the most reasonable prediction of longitudinal aerodynamic coefficients.

The four inputs used for the lateral aerodynamic coefficients prediction were angle of attack (alpha), side angle attack (beta), speed brake deflection angle, and Mach number. Initial analysis focused on predicting the coefficient of rolling moment. The four neural network architectures chosen for the longitudinal coefficients prediction were also applied to the lateral aerodynamic coefficients. The first neural network architecture considered for lateral aerodynamic coefficients was a three layer 4-12-1 network. The sigmoidal transfer function was used between layers and performed reasonably well. In order to reduce the under fitting and over fitting, the number of hidden neurons was varied between 12 to 40 neurons as shown in Figure 5. Although it was difficult to exactly match the highly non-linear nature of the coefficient of rolling moment data, the 4-24-1 neural network architecture produced a reasonably closer pattern than any other architectures.

The next focus was to determine how many data pairs were required to constitute a good training dataset. A similar approach to selecting a training dataset for longitudinal coefficients did not work as well for lateral aerodynamic coefficients. The interval of data in lateral aerodynamic coefficient is limited and the variation in the range is also low. Hence, the strategy for organizing the training dataset for lateral aerodynamic coefficient had to be totally different than for longitudinal coefficients. The available 7456 data pairs for lateral aerodynamic coefficients were divided into 8 clusters based on side slip angle (beta). All lateral coefficients are zero when beta is 0 degree, so only seven datasets had real values for the lateral coefficients.

The training data and prediction data combinations are provided in Table 4. The 'T' or 'Trg' refers to training dataset and 'P' refers to prediction dataset. In figure 6, the 2nd training dataset selection did not produce a reasonable result since only the beta=10 degree training dataset was selected while the remaining clusters were used for prediction datasets. In figure 7, the 2nd training dataset based plot is omitted, and the rest of the plot is expanded to better examine the accuracy of predicting the rolling moment coefficient. The 4th training dataset architecture produced the best fit. The number of over fitting exceedance and time to converge for training the neural network was as the lowest for the 4th training data architecture. Hence we decided to use 7 data clusters for the training dataset and 1 data cluster as the prediction dataset. Using this training data selection approach, we applied the neural network to predict coefficient of side force and yawing coefficient as shown in figures 8 and 9. Both neural networks produced very reasonable predictions with only a few over or under fitted results.

Side slip angle (Beta) degrees	0	1	2	3	4	6	8	10	Number of iterations for convergence	No. of data points exceeded 100 % over fitting
1st Trg	T	T	P	P	P	P	T	T	73	407
2nd Trg	T	P	P	P	P	P	P	T	98	4737
3rd Trg	T	T	T	P	P	T	T	T	59	203
4th Trg	T	T	T	T	P	T	T	T	54	119

Table 4 Selection of Training and Prediction Dataset for Coefficient of Rolling Moment

7.CONCLUSION

This study investigated the training data and transfer functions required to produce an efficient neural network architecture for predicting aerodynamic coefficients. The aerodynamic dataset considered for this analysis was for the Space Shuttle Vehicle in approach and landing. Once a neural network is exposed to a good training dataset, it can be used to predict complex aerodynamic coefficients. In this analysis, the neural network was used to predict aerodynamic coefficients independent of each other. The neural network architectures for longitudinal aerodynamic coefficient prediction and lateral aerodynamic coefficient were different. The best neural network architecture for predicting longitudinal aerodynamic coefficients was a three layer neural network architecture using a sigmoidal function between layers. For lateral coefficients, the requirement for the number of hidden neurons was higher than that for longitudinal coefficients. Future research will investigate creating a cluster of neural networks to handle all aerodynamic coefficients at once rather than independently. Overall the neural network prediction for aerodynamic force coefficients was relatively better as compared to predicting aerodynamic moment coefficients. Further research will focus on finding an optimized neural network architecture for both aerodynamic forces and moments.
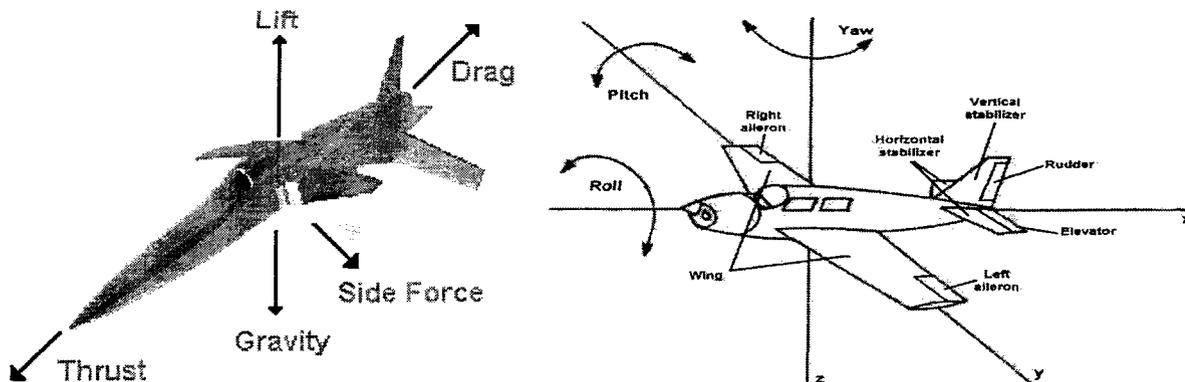
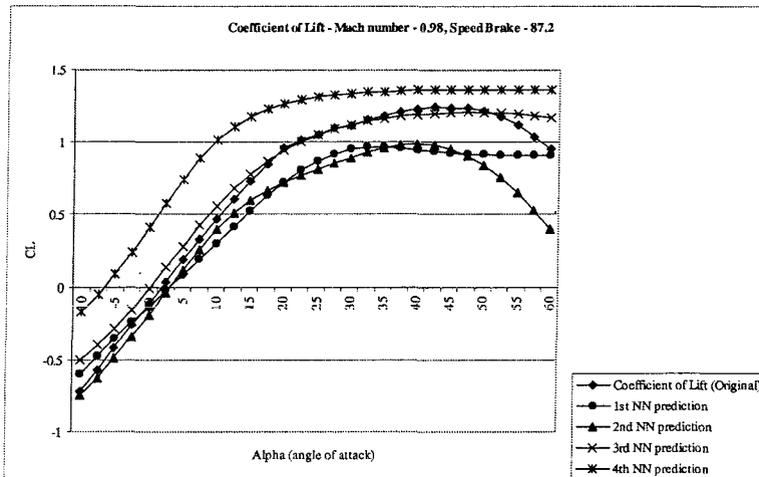## ACKNOWLEDGEMENT

Figure 1 Aerodynamic Forces and Moments

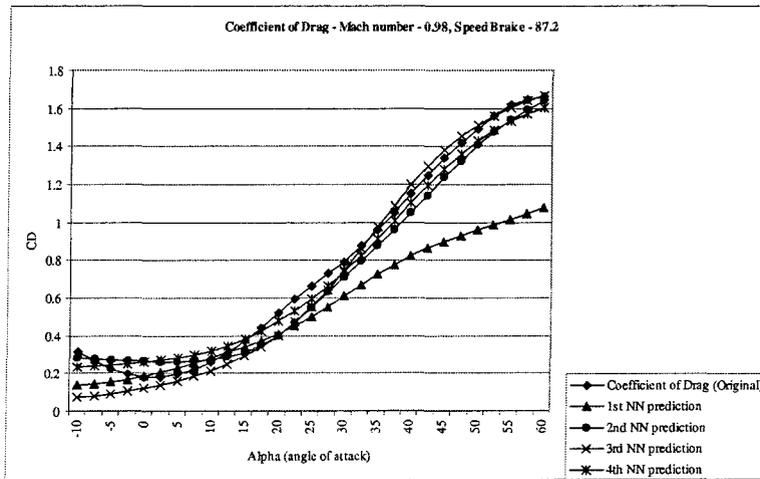Figure 2 Neural Network Predictions for Coefficient of Lift

Figure 3 Neural network Predictions for Coefficient of Drag
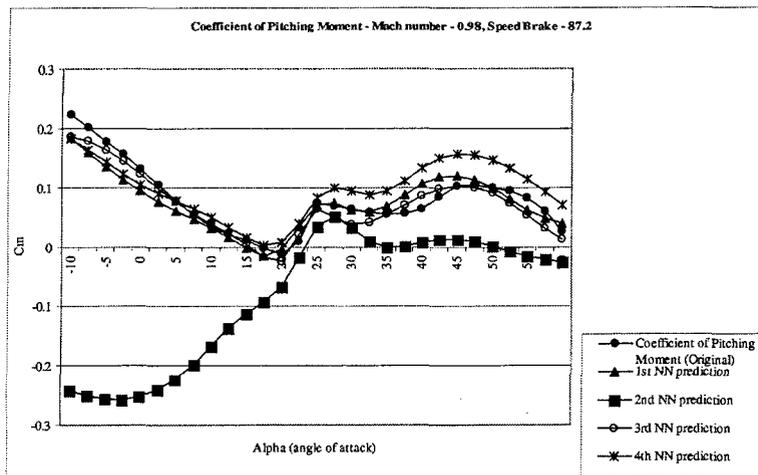

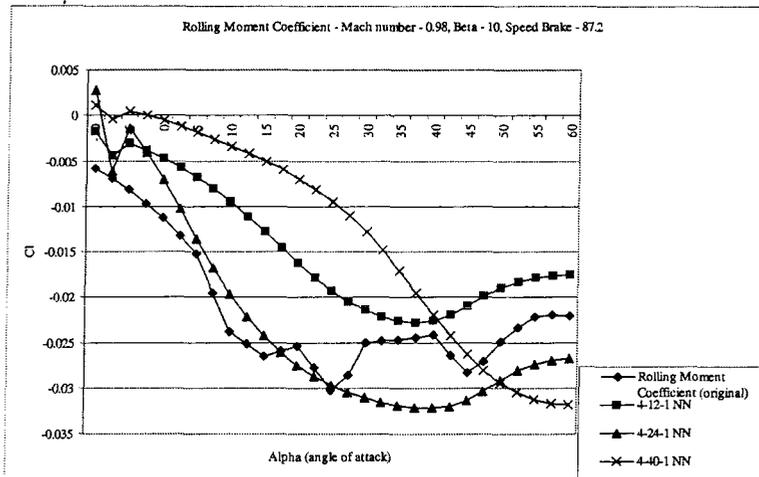Figure 4 Neural Network Predictions for Coefficient of Pitching Moment

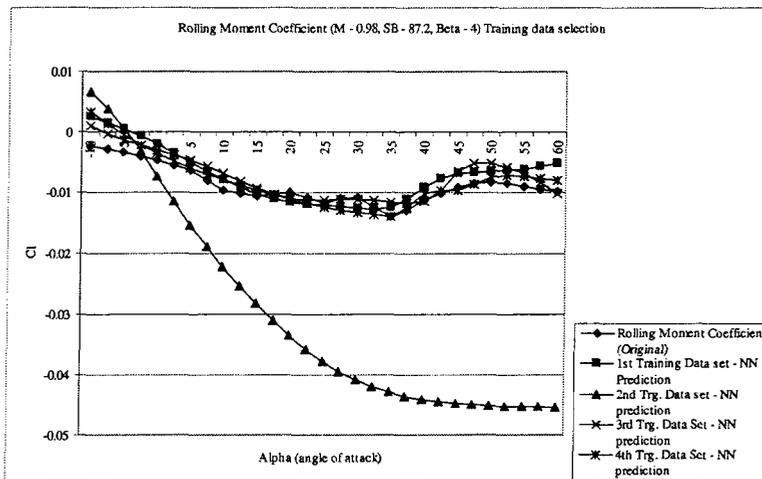Figure 5 Neural Network Predictions for Coefficient of Rolling Moment

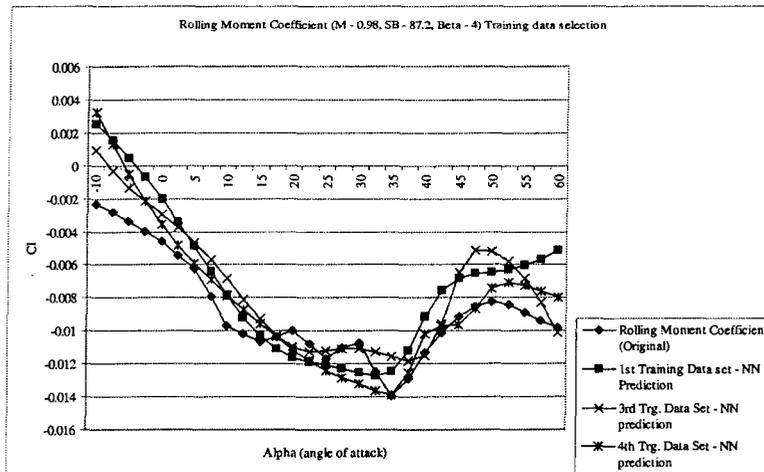Figure 6 Neural Network Predictions for Coefficient of Rolling Moment

Figure 7 Neural Network Predictions for Coefficient of Rolling Moment (2nd Trg. Omitted from fig. 6)
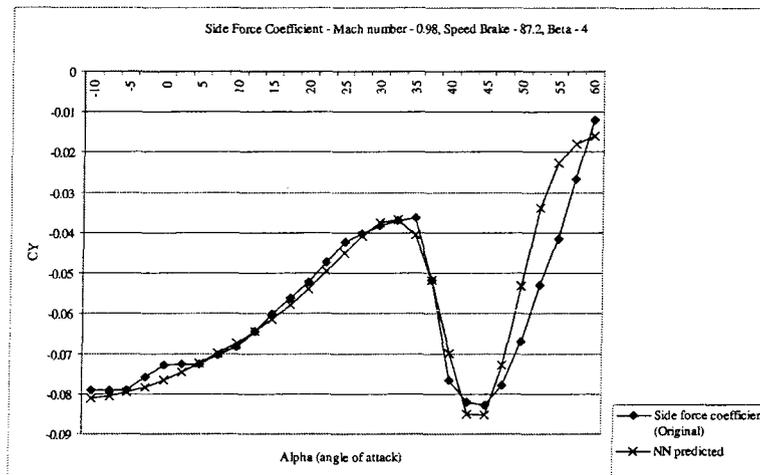
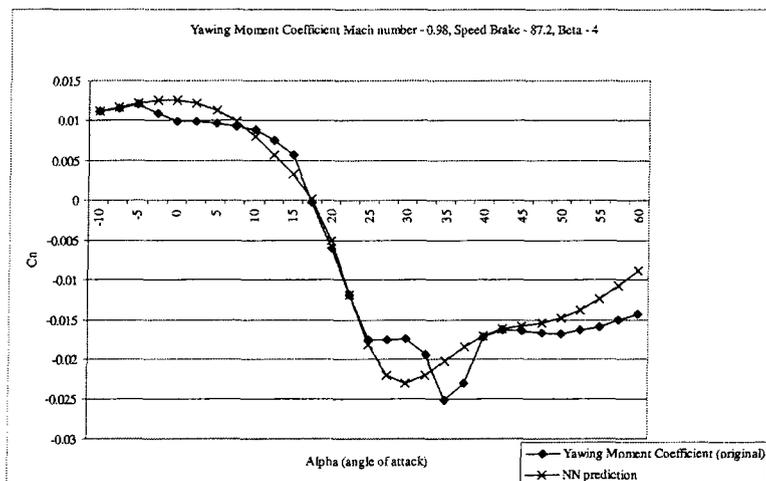Figure 8 Neural Network Predictions for Coefficient of Side Force

Figure 9 Neural Network Predictions for Coefficient of Yawing Moment

REFERENCES

1. A Miele, *Theory of Optimum Aerodynamic Shapes*, Academic Press, 1965.

2. B. Dwyer, "Dynamic Ground Effects Simulation Using OVERFLOW-D", NASA High-Speed Research Program Aerodynamic Performance Workshop, Volume 2, pp 2388-2469, NASA/CP-1999-209692/VOL2, 1998.

3. B. L. Kalman, and S.C. Kwansy, ""Why Tanh?" Choosing a Sigmoidal Function", International Joint Conference on Neural Networks, Baltimore, MD, 1992.

4. C. Jondarr, "Back propagation Family", Technical Report C/TR96-05, Macquarie University, Australia, 1996.

5. S.Cliff, S. Smith, S. Thomas and F. Zuniga, "Aerodynamic Database Creation for Flight Simulation", VMS complex (internal report), NASA Ames Research Center, 2002.

6. D. J. Korsmeyer, J. D. Walton and B. L. Gilbaugh, "DARWIN - Remote Access and Intelligent Data Visualization Elements", AIAA-96-2250, AIAA 19th Advanced Measurement and Ground Testing Technology Conference, New Orleans, LA, June 1996.

7. D. J. Korsmeyer, J. D. Walton, C. Aragon, A. Shaykevich and L. Chan, "Experimental and Computational Data Analysis in CHARLES and DARWIN", ISCA-97, 13th International Conference on Computers and Their Applications, Honolulu, HI, March 1998.

8. D. L. Elliott, "A Better Activation Function for Artificial Neural Networks", ISR technical report TR93-8, 1993.

9. D.E. Rumelhart., G.E.Hinton and R.J. Williams, "Learning Internal Representations by Error Propagation", in D.E. Rumelhart and J.L. McClelland, eds. *Parallel Data Processing*. Vol 1, Cambridge, MA: The MIT Press, pp 318-362, 1986.

10. D.P. Raymer, *Aircraft Design: A Conceptual Approach*, AIAA, Washington DC, 1999.

11. G. Sovran, T. Morel, W.T. Mason, *Aerodynamic Drag Mechanisms of Bluff Bodies and Road Vehicles*, Plenum Press, New York, 1978.

12. G.E. Cooper, and R.P. Harper, "The Use of Pilot Rating in the Evaluation of Aircraft Handling Qualities", NASA TN D-5153, April 1969.

13. J. A. Schreiner, P. J. Trosin, C. A. Pochel and D. J. Koga, "DARWIN—Integrated Instrumentation and Intelligent Database Elements", AIAA-96-2251, 19th AIAA Advanced Measurement and Ground Testing Technology Conference, New Orleans LA, June 1996.

14. B.Jackson, and C. Cruz, "Preliminary Subsonic Aerodynamic Model for Simulation Studies of the HL-20 Lifting Body", NASA TM 4302, 1992.

15. J. D. Walton, D. J. Korsmeyer, R. K. Batra and Y. Levy, "The DARWIN Workspace Environment for Remote Access to Aeronautics Data", AIAA-97-0667, 35th Aerospace Sciences Meeting, Reno NV, January 1997.

16. L. R. Miranda, R.D. Elliot, and W.M. Baker, "A Generalized Vortex Lattice Method for Subsonic and Supersonic Flow Applications", NASA CR 2865, December, 1977.

17. L.G. Birckelbaw, W.E. McNeill, and D. A. Wardwell, "Aerodynamics Model for a Generic STOVL Lift-Fan Aircraft", NASA TM 110347, 1995.

18. M. Norgaard, C. Jorgensen, and J. Ross, "Neural Network Prediction of New Aircraft Design Coefficients", NASA TM 112197, 1997.

19. M.A. Park, and L.L. Green, "Steady-State Computation of Constant Rotational Rate Dynamic Stability Derivatives", AIAA Paper 2000-4321, 2000.

20. M.M. Rai, and N.K. Madavan, "Aerodynamic Design Using Neural Network", AIAA Journal, Vol. 38, No. 1, pp 173-182, 2000.

21. M.T. Hagan, and M.B. Menhaj, "Training Feed Forward Networks With the Marquardt Algorithm", IEEE Transactions on Neural Networks, Vol. 5, No. 6, pp 989-993, 1994.

22. R. Eppler, *Airfoil Design and Data*, Springer Verlag, 1990.

23. S. D. Thomas, M.A. Won, S.E. Cliff, "Inlet Spillage Drag Predictions Using the AIRPLANE Code", 1997 NASA High-Speed Research Program Aerodynamic Performance, Vol. 1, Part 2, pp 1605-1648, NASA/CP- 1999-209691, December 1999.

24. S. Lawrence, G. Lee, and T. Ah Chung, "Lessons in Neural Network Training: Over fitting May be Harder than Expected", Proc. of the Fourteenth National Conference on Artificial Intelligence, AAAI-97, 1997.

25. S. Lawrence, G. Lee, and T. Ah Chung, "What Size Neural Network Gives Optimal Generalization? Convergence Properties of Back propagation", Technical Report UMIACS-TR-96-22 and CS-TR-3617, 1996.

26. T. Master, *Practical Neural Network Recipes in C++*, Academic Press Inc., 1993.

27. T. Rajkumar and J. Bardina, "Prediction of Aerodynamic Coefficients Using Neural Network for Sparse Data", Proc. of FLAIRS 2002, Florida, USA, 2002.

28. T. Rajkumar, C. Aragon, J. Bardina and R. Britten, "Prediction of Aerodynamic Coefficients for Wind Tunnel Data using a Genetic Algorithm Optimized Neural Network", Proc. Data Mining Conference, Bologna, Italy, 2002.

29. M.J. Aftonis, M.J. Berger, G. Adomauicious, "A Parallel Multilevel Method for Adaptively Refined Cartesian Grids with Embedded Boundaries" AIAA 2000-0808, 38th Aerospace Sciences Meeting, Reno, NV, January 2000.