JOURNAL OF AIRCRAFT
Vol. 36, No. 5, September–October 1999

Minimizing Computational Data Requirements for Multi-Element Airfoils Using Neural Networks

Roxana M. Greenman* and Karlin R. Roth†
NASA Ames Research Center, Moffett Field, California 94035

Artificial neural networks were used to fill in a design space of computational data to optimize the flap position for maximum lift for a multi-element airfoil. Multiple-input, single-output networks were trained using NASA Ames Research Center's variation of the Levenberg–Marquardt algorithm. The computational data set was generated using an incompressible Navier–Stokes algorithm with the Spalart–Allmaras turbulence model. An empirically-based criteria, designated the "pressure difference rule," was applied to the training set because numerical inaccuracies for the computational method were identified near maximum lift. The neural networks were trained with only three values of the inputs: flap deflection, gap, and overlap at various angles of attack. The entire computational data set was thus sparse, and yet by using only 52–70% of the computed data, the trained neural networks predicted the aerodynamic coefficients within 1.7% of the maximum lift coefficient. In addition, a genetic algorithm and a gradient-based optimizer were integrated with the neural networks to optimize the high-lift rigging. This new optimization process had a higher fidelity and a reduction in CPU time when compared with an optimization procedure that excluded the genetic algorithm.

Nomenclature

C_d = drag coefficient, $D/(q_\infty S)$
C_l = lift coefficient, $L/(q_\infty S)$
C_m = moment coefficient, $M/(q_\infty Sc)$
C_p = pressure coefficient, $p - p_\infty/q_\infty$
c = chord, 30 in.
D = drag force
L = lift force
L/D = lift-to-drag ratio
M = pitching moment
ol = overlap
q_∞ = freestream dynamic pressure, $\frac{1}{2}\rho_\infty V_\infty^2$
Re_c = Reynolds number, $\rho_\infty V_\infty c/\mu_\infty$
S = wing area
V = freestream velocity
α = angle of attack
ΔC_p = pressure difference, $\Delta C_{p_{\text{diff}}} = C_{p_{\text{peak}}} - C_{p_{\text{te}}}$
δ = deflection angle
ρ = density

Subscripts

f = flap
max = maximum
nn = neural network computation
s = slat
te = trailing edge
∞ = freestream value

Introduction

THE design of an aircraft's high-lift system is a crucial part of the design phase of commercial and military airplanes because this system controls the takeoff and landing performance. A well-designed high-lift system can lead to increased payloads and increase the operational flexibility by extending ranges and by decreasing takeoff and landing distances. Traditionally, high-lift designs have been accomplished through extensive wind-tunnel and flight-test programs that are expensive and difficult due to the extremely complex flow interactions. Recently, computational fluid dynamics (CFD) has been incorporated in high-lift design.[1–3] For high-lift applications, CFD can also be expensive because the entire design space is large, grids must be generated around geometrically complex high-lift devices, and complex flow phenomena must be resolved. To achieve optimum rapid designs, new computational tools for speedy and efficient analysis of high-lift configurations are required.

Artificial neural networks are a collection (or network) of simple computational devices used to calculate nonlinear data.[4–6] Recently, neural networks have been applied to a wide range of problems in the aerospace industry. One study demonstrated that application of neural networks to rotor blade design reduced the time required to optimize the blades.[7] Faller and Schreck[8] used neural networks to predict real-time, three-dimensional, unsteady separated flowfields and aerodynamic coefficients of a pitching wing. It has also been shown that neural networks trained with measured data predict with sufficient accuracy to enable identification of instrumentation system degradation.[9] Steck and Rokhsaz[10] have trained neural networks to predict aerodynamic forces with sufficient accuracy for design and modeling. As a final example, Rai and Madavan[11] demonstrated the feasibility of applying neural networks to aerodynamic design of turbomachinery airfoils.

Neural networks have been used at NASA Ames Research Center to minimize the amount the data required to define the aerodynamic performance characteristics of a wind-tunnel model.[12,13] It was shown that when only 50% of the data acquired from the wind-tunnel test was used to train neural nets, the results had the same predictive accuracy as the experimental measurements. The success of NASA Ames Research Center's neural-network application for wind-tunnel data prompted this current study [14,15] to use neural networks to minimize the amount of CFD data required to accurately train neural networks to predict high-lift aerodynamics of a particular multi-element airfoil as a function of angle of attack and a specific range of slat and flap positions.

In this study, neural networks are used to minimize the number of geometries needed to be computed to accurately define the high-lift performance of a multi-element airfoil within a given design

Received 27 May 1998; presented as Paper 99-0258 at the AIAA 37th Aerospace Sciences Meeting, Reno, NV, 11–14 January 1999; revision received 22 January 1999; accepted for publication 25 January 1999.

*Aerospace Engineer, Integrated Systems Technologies Branch. Member AIAA.
†Chief, Chief Technology Office. Senior Member AIAA.

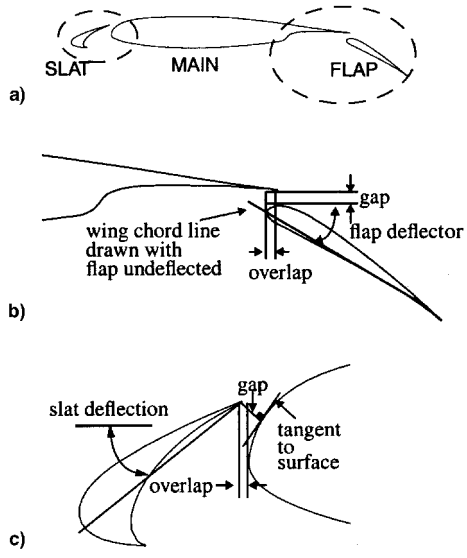

Fig. 1 a) Three-element airfoil, b) definition of flap rigging parameters, and c) definition of slat rigging parameters.

space. The three-element airfoil is sketched in Fig. 1a. The computational training set for the neural networks is generated using a two-dimensional, incompressible Navier–Stokes solver. Brief descriptions of the geometry, grid generation, and the numerical method are provided in the next section. The neural networks are trained by using the flap riggings and angles of attack as the inputs and the aerodynamic forces as the outputs; they are described in the Results section. This paper examines the fidelity of the computational database near maximum lift conditions, evaluates the impact of selective reductions within the training data set on the neural-net predictions, and tests an optimization process in which the neural networks represent the design space.

Computational Training Set Generation

Geometry Definition

Extensive wind-tunnel investigations[‡] have been carried out for the geometry shown in Fig. 1 in the 7- by 10-ft Wind Tunnel No. 1 at the NASA Ames Research Center. The model is a three-element wing whose sections consist of a 12%c LB-546 slat,[16] NACA 63_2-215 Mod B main element, and a 30%c Fowler flap, where c is the chord of the cruise wing. There are two different slat-deflection angles: 6 and 26 deg. This paper will only present the results for the 6-deg slat (see Ref. 14 for the 26-deg-deflected slat results). Each slat has a $\text{gap}_s = 2.0\%c$ and an $ol_s = -0.05\%c$. For the computational database, 27 different flap riggings were created for each slat deflection. The flap riggings are combinations of the flap deflection, gap, and overlap that are defined in Fig. 1b. The flap deflection angles are $\delta_f = 25.0$, 29.0, and 38.5 deg. The three gap settings are $\text{gap}_f = 1.50$, 2.10, and 2.70%c; whereas the overlap settings are $ol_f = 0.40$, 1.00 and 1.50%c. All gap and overlap values in this paper are expressed in terms of percent chord. In this study, the neural networks are defined to be successfully trained to predict the aerodynamic coefficients when given a set of inputs that are not in the training set, the outputs are predicted within the experimental error of the 7- by 10-ft Wind Tunnel results. The experimental error of the total lift coefficient C_l is ± 0.02 for $C_l < 0.95 C_{l_{\max}}$ and ± 0.06 for $C_l \geq 0.95 C_{l_{\max}}$, where $C_{l_{\max}}$ is the maximum lift coefficient. There were no experimental tolerances reported for the drag and moment coefficients.

Grid Generation

The grids around the three-element airfoil are generated by using the overset multiple airfoil grid generator (OVERMAGG[17]), which is an automated script system used to perform overset multi-element airfoil grid generation. The rule-based high-lift grid-generation techniques contained within OVERMAGG have evolved from the high-lift CFD applications by Rogers.[18] OVERMAGG takes as input the surface definition of the individual elements of the airfoil. It then creates a surface grid for each individual element by generating and redistributing points from the given surface definition. It calls the hyperbolic generator (HYPGEN) code[19] to generate volume grids about each element. OVERMAGG also automatically calls the PEGSUS code[20] to unite the individual meshes into an overset grid system, which is the final output of OVERMAGG.

Figure 2 shows the grid system. A total of 121,154 grid points are used, consisting of a 242×81 C-grid around the slat, a 451×131 C-grid around the main element, and a 351×121 embedded grid around the flap, which is used to help resolve the merging wake in this region. The normal wall spacing for all grids is 5×10^{-6} chords and $Re_c = 3.7 \times 10^6$. Grid density studies, documented in Ref. 14, show that this system adequately captures the flow.

Numerical Method

To generate the necessary computational training data, each high-lift configuration is analyzed at 10 different angles of attack. The incompressible Navier–Stokes equations in two-dimensional (INS2D) generalized coordinates are solved using the INS2D[21,22] flow solver. This code has been used extensively to predict multi-element airfoil flows.[1,18] INS2D uses an artificial compressibility approach to couple the mass and momentum equations. The convective terms are differenced using a third-order-accurate upwind-biased flux splitting. The equations are solved using a generalized minimum residual (GMRES) implicit scheme. Because the flow is turbulent, the Spalart–Allmaras[23] turbulence model is used. This turbulence model has been successfully used to compute high-lift flowfields.[18,24] The flow was treated as fully turbulent for all elements in the computations.

The maximum residual in the solution is reduced by seven orders of magnitude, and the maximum divergence in the converged solution is on the order of 10^{-4} or less. A typical solution on a Cray C90 computer converged in 200 iterations for a total of 268 s (10.6 μs/iteration/point). As the angle of attack increases and approaches the angle for maximum lift, the flow around the airfoil tends to separate from the upper surface and creates a large wake behind the airfoil. Inside this separated region, the flow is recirculating. The flowfield near maximum lift is more complex, and, thus, more time is required for convergence. The solutions near maximum lift converged in 800 iterations for a total of 1072 s.

Neural Networks

The architecture of the neural networks in this study is a multilayer network with tangent hyperbolic activation functions in hidden layer units, and a linear transfer function in the output unit. Individual four-input, one-output networks are used to model each of the desired aerodynamic coefficients. A NASA Ames Research Center's variation of the Levenberg–Marquardt training scheme[12,25] is used because it provides better accuracy than all other schemes tested, including the back-propagation training method. The single-output networks for each of the aerodynamic coefficients yield more precise modeling than multiple-output networks.[9,12] The neural network contains 15 nodes in the hidden layer and Fig. 3 shows a sketch of the architecture. The neural networks are trained using 250 training iterations. Reference 14 outlines the process that was used to determine the correct number of iterations used for training.

The four independent input variables are flap deflection, gap, overlap, and angle of attack α, as illustrated in Fig. 3. The outputs are lift, drag, and moment coefficients C_l, C_d, and C_m, respectively, and the lift-to-drag ratio L/D. Thus, four different networks are trained to completely define the aerodynamics. $(L/D)_{\text{nn}}$ can also be computed by the ratio of the lift and drag coefficients, $C_{l_{\text{nn}}}/C_{d_{\text{nn}}}$; however, because this is an important parameter in aerodynamics, it is computed independently to see if there are any advantages in accuracy.

The objective of this study is to determine the least amount of data required to accurately train the networks. The approach is to create

[‡]Storms, B., private communication, NASA Ames Research Center, Moffett Field, CA, July 1997.

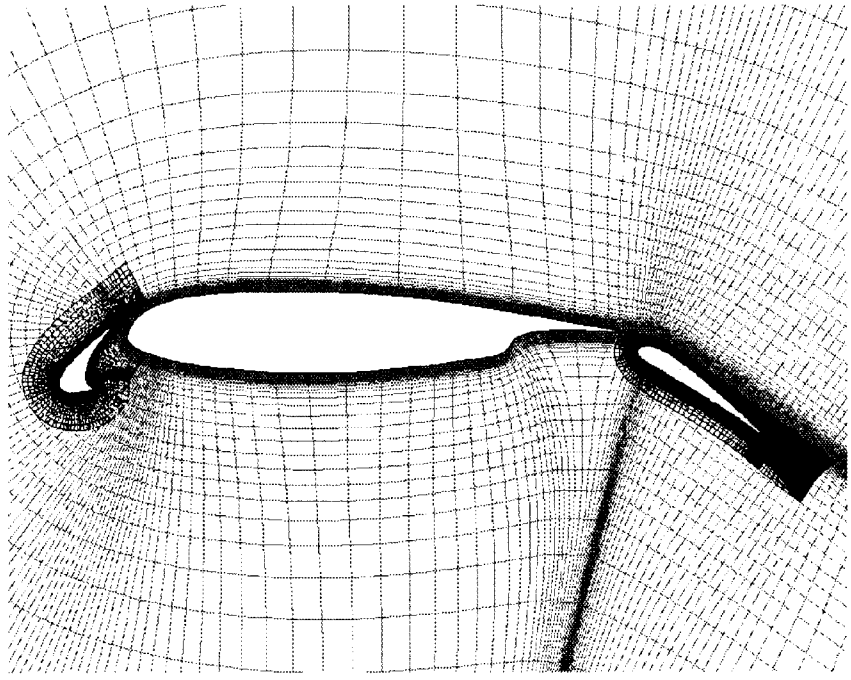

Fig. 2 Grid around three-element airfoil (every other point shown for clarity).

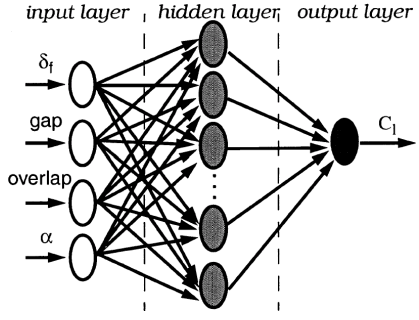

Fig. 3 Neural network architecture with 15 nodes in the hidden layer.

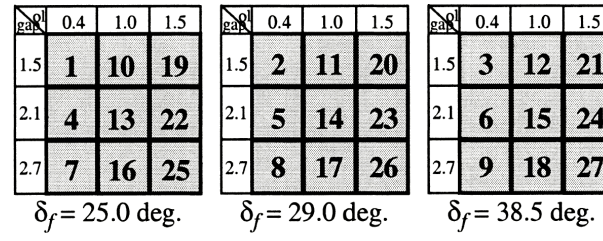

Fig. 4 Computational cases used to train neural networks.

different subsets of the computational data that include a portion of the flap riggings to train the neural networks. Once the neural networks are trained with each subset, they are tested for all of the different flap riggings for the entire range of angles of attack. The number of angles of attack in the training set and the testing set may be different for each high-lift configuration. The accuracy of the neural networks are evaluated by computing the rms error for each aerodynamic coefficient. For the computational training data set, the deviations from the neural-net predicted data and the INS2D-computed value for each angle of attack and for each flap rigging is calculated. Then the rms error is calculated for each configuration.

Results

Computational Training Set

The neural networks are trained with computational data consisting of different flap riggings as shown in Fig. 4. The large numbers in the shaded boxes in Fig. 4 are the case numbers for particular flap riggings. There are three flap-gap settings of $\text{gap}_f = 1.5$, 2.1, and 2.7%c, and three flap-overlap settings of $ol_f = 0.4$, 1.0, and 1.5%c. The flap deflection angles are $\delta_f = 25.0$, 29.0, and 38.5 deg. In this study, 10 angles of attack were computed for each case from $0.0 \leq \alpha \leq 22.0$ deg.

Physics of Computational Training Sets

The flowfield about a high-lift multi-element airfoil is very complex and difficult to compute by any available method.[26,27] Some of the difficult features to capture are trailing viscous wakes, whose strength and location vary with angle of attack, merging wakes and boundary layers, different transition locations on each of the airfoils elements, boundary-layer separation, and reversed flows in the main element wake. In addition, the airfoil performance varies with Reynolds and Mach number.[28,29]

Pressure distributions are plotted and compared for two cases with different slat and flap deflections in Fig. 5. The high-lift setting for the first case is $\delta_s = 6.0$ deg, $\text{gap}_s = 2.0\%c$, $ol_s = -0.05\%c$, $\delta_f = 38.5$ deg, $\text{gap}_f = 2.1\%c$, $ol_f = 1.0\%c$; the second case is $\delta_s = 26.0$ deg, $\text{gap}_s = 2.0\%c$, $ol_s = -0.05\%c$, $\delta_f = 53.0$ deg, $\text{gap}_f = 2.1\%c$, $ol_f = 1.0\%c$. Figure 5a shows the slat, main, and flap elements for both configurations.

The 6-deg-deflected slat configuration has attached flow on all three elements, as seen in Fig. 5b. High-suction pressures are observed on the slat for this case. Sharp spikes are evident in the pressures near the element trailing edges, particularly the flap. This is caused by the numerical grid that comes to a sharp point at the trailing edge. There are multiple spikes in the pressure distribution near the leading edge of the flap that are attributed to the geometry of the flap. The flap geometry at this region is faceted in the original model. The pressure spikes indicate that the flow is accelerating around the corners of these facets. In contrast, the $\delta_s = 26.0$-deg configuration has a very different pressure distribution. Examining the pressure distribution and particle traces (not shown) of the airfoil show that the slat and the flap have separated flow. Numerical methods, including the one used to generate the computational training set, have difficulty predicting severely separated flows and can result in extraneous nonlinearities. These nonlinearities will lead to higher errors in the neural-net characterization of the data.

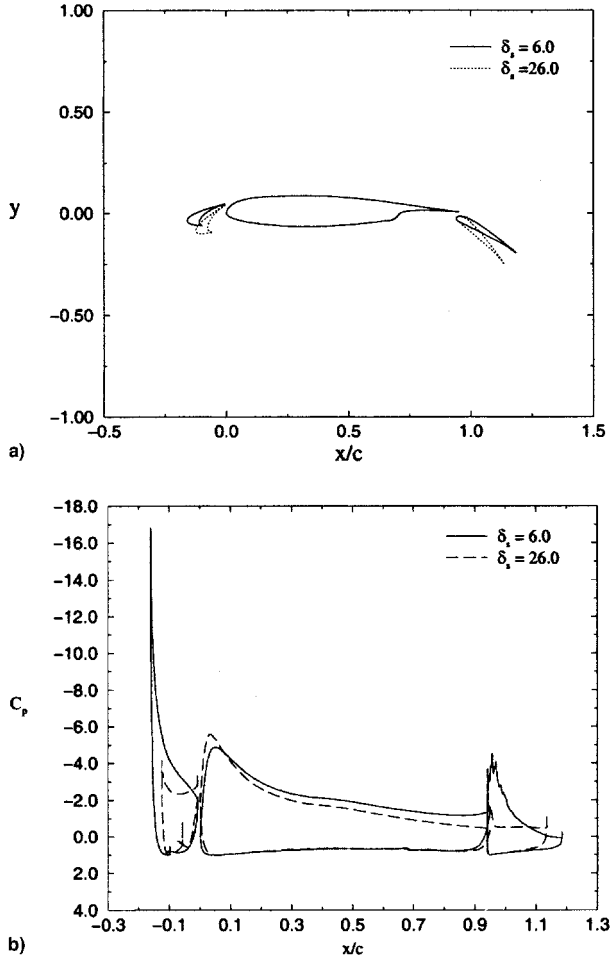

Fig. 5 Comparison of slat-deflection aerodynamics: $\delta_s = 6.0$ deg, $\text{gap}_s = 2.0\%c$, $ol_s = -0.05\%c$, $\delta_f = 38.5$ deg, $\text{gap}_f = 2.1\%c$, $ol_f = 1.0\%c$, $\alpha = 10$ deg; and $\delta_s = 26.0$ deg, $\text{gap}_s = 2.0\%c$, $ol_s = -0.05\%c$, $\delta_f = 53.0$ deg, $\text{gap}_f = 2.1\%c$, $ol_f = 1.0\%c$, $\alpha = 10$ deg: a) high-lift riggings and b) pressure distribution.

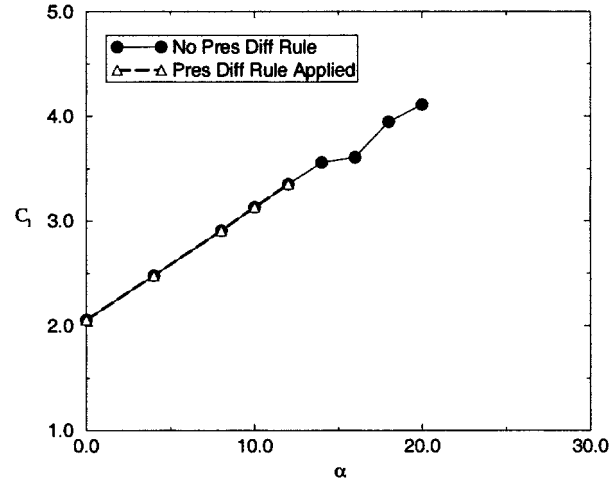

Fig. 6 Computational lift coefficient vs angle of attack for $\delta_s = 6.0$ deg, $\text{gap}_s = 2.0\%c$, $ol_s = -0.05\%c$, $\delta_f = 38.5$ deg, $\text{gap}_f = 2.7\%c$, $ol_f = 0.4\%c$.

Maximum Lift Criteria

Figure 6 shows a plot for the computed lift curve for the configuration $\delta_s = 6.0$ deg, $\text{gap}_s = 2.0\%c$, $ol_s = -0.05\%c$, $\delta_f = 38.5$ deg, $\text{gap}_f = 2.7\%c$, $ol_f = 0.4\%c$. It is shown that the computational curve (solid line) continues to increase and thus does not accurately predict the maximum lift. Previous studies[27] have also shown that state-of-the-art two-dimensional computations cannot accurately predict maximum lift. Further, Rogers has demonstrated that a two-dimensional CFD prediction using INS2D of a NACA 4412 airfoil predicts $C_{l_{\max}}$ to be too high and at too high an angle of attack.[30] No current computational method has been shown to resolve all of these features for high lift and accurately predict maximum lift.

Valarezo and Chin[27] reported a hybrid method that couples cost-effective CFD technology with empirically observed phenomenon to predict maximum lift for complex multi-element wing geometries. Their semi-empirical $C_{l_{\max}}$ criteria for multi-element airfoils or wings, designated the "pressure difference rule" (PDR), states that for a given Reynolds and Mach number combination, there exists a certain difference between the peak suction pressure coefficient and the trailing-edge pressure coefficient, $\Delta C_{p_{\text{diff}}} = C_{p_{\text{peak}}} - C_{p_{\text{te}}}$, at the maximum lift condition. For the flow conditions of this study, the pressure difference value is −13.0. The rule is applied to each element of the airfoil. The slat is the element that has pressure differences greater than the acceptable value, thus, it is the critical element in determining the maximum lift.

To see if the neural networks would predict the aerodynamic data better once the pressure difference rule is applied, the neural networks are trained with the entire data set for the 6-deg-deflected slat and then also with the data set that satisfies the pressure difference rule. As Fig. 6 illustrates, the PDR predicts that maximum lift occurs at an angle of attack of $\alpha = 12.0$ deg; thus, only the data up to and including $\alpha = 12.0$ deg (dashed line) is used in the training set for this particular high-lift rigging.

The comparison of the rms error for each training set is shown in Fig. 7 for cases 1–27. For all four outputs, C_l, C_d, C_m, and L/D, the rms error is much lower when the PDR is applied. As expected, the training data are better behaved and the neural network is able to predict the aerodynamic characteristics more accurately.

Minimizing Training Data Samples

Even though the computational database that is used for training is sparse, a study was conducted to see how much further the training set could be reduced and still allow the neural networks to predict within the acceptable accuracy. By reducing the training data set further, the required computational resources can be decreased. Several subsets of the computational data are created to train the neural networks and to test the accuracy of the prediction. Each configuration that is generated has its flowfield computed at several angles of attack, but not necessarily at the same angles. The number of angles of attack also varies for each configuration. In general, once the grid is generated, it does not acquire extra engineering effort to compute solutions at different angles of attack. The neural networks are therefore trained using data sets (denoted by method numbers in this paper), which contain various numbers of configurations but all of the angles of attack for each configuration. Nine methods of training were manually selected and evaluated for both the $\delta_s = 6.0$- and 26.0-deg configurations, as discussed in Ref. 14. Similar results were reported for each slat configuration.[14] Only a representative subset of the training methods is reported here because of space limitation.

Method 1 designates the training set that includes all of the configurations in the training set. As expected, this has the lowest rms error for all of the training methods tested because all of the configurations tested are included in the training set. The rms errors for C_l, C_d, and C_m are shown in Fig. 8. Method 1 represents the training data in a nonlinear fashion for all four inputs (δ_f, gap_f, ol_f, and α). Method 1 predicts C_l, C_d, and C_m with almost no error for all 27 cases. Case 27, which has a flap setting of $\delta_f = 38.5$ deg, $\text{gap}_f = 2.7\%c$, and $ol_f = 1.5\%c$, has the highest prediction error. This case is at the boundary of the design space, and, hence, is not as well defined as the points on the interior of the design space.

The next method that is used to train the neural networks is method 2, which has a nonlinear representation (three values used to represent data) of the flap deflection and angle of attack. The other two inputs, flap gap and overlap, are represented in a linear (two values used to represent data) manner. This method contains 56% of the configurations in the entire training set. In Fig. 9, the boxes that are shaded represent the cases that are in the training set, whereas the numbers in the white boxes and in the parentheses are the cases that are omitted from the training set. The rms errors are low for the cases that are in the training set and high for the

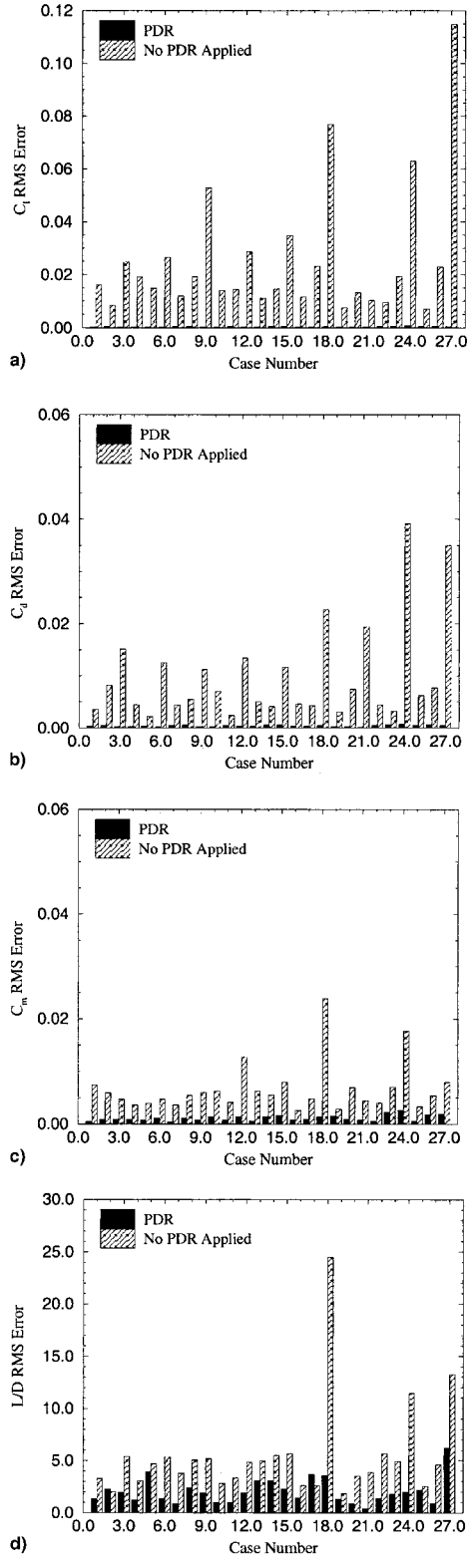

Fig. 7 Comparison of rms error for maximum lift criteria: a) C_l, b) C_d, c) C_m, and d) L/D rms errors.

cases that are not included in the training set. The C_l rms error is greater than the acceptable limit. This method was expected to do poorly because it represents the aerodynamic data as linear when it is known to be nonlinear.

Training method 5 is a checkerboard method alternating between flap configurations as shown in Fig. 10. For $\delta_f = 25$ and 38.5 deg, the corner and middle high-lift riggings are used in the training set. In contrast, the interior ones are kept in the training set for $\delta_f = 29.0$ deg. Thus, there is representation in the training set for

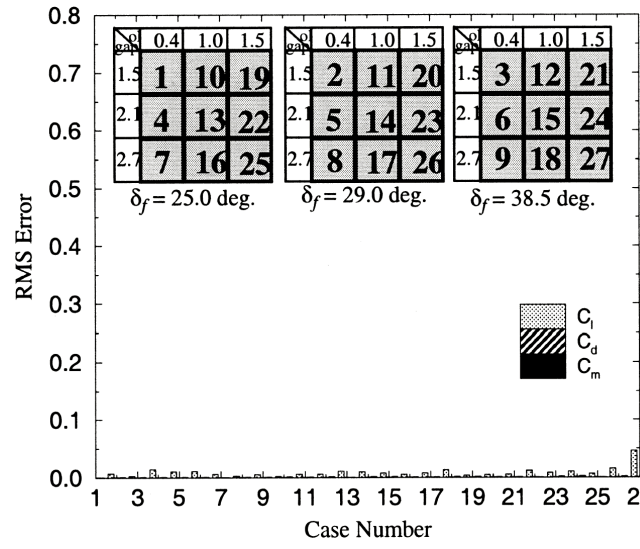

Fig. 8 Summary of rms error from neural-network prediction of aerodynamic coefficients for method 1.

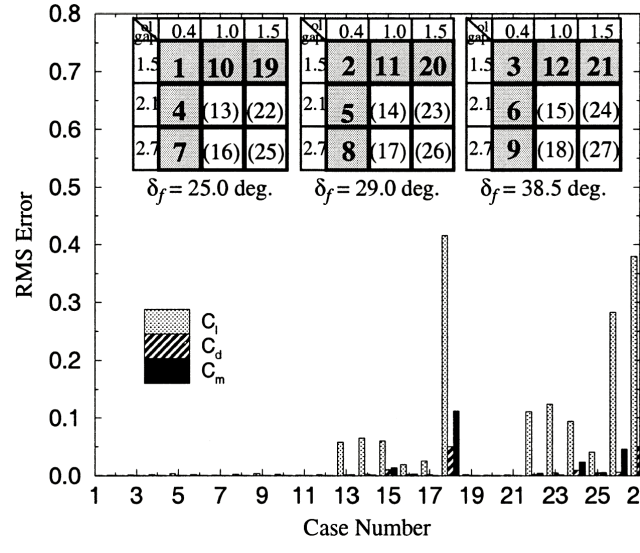

Fig. 9 Summary of rms error from neural-network prediction of aerodynamic coefficients for method 2.

every high-lift rigging combination, even though they are for various flap deflections. The rms errors (Fig. 10) show that the prediction for C_l, C_d, and C_m are very good for the cases that are in the training set. Even though there are a few cases that were not included in the training where the C_l rms error is high, the prediction for most of the cases is quite good. The C_d prediction for most of the cases (included or not in the training set) is good. The moment coefficient prediction is accurate for most of the configurations in the training set. Also, the C_m prediction is good for more than half of the cases omitted in the training set.

To further improve the accuracy of the prediction while still reducing the number of configurations relative to the full training set, careful selections of the configurations contained in the training set were created. An example of a subset that was very successful in training the neural networks to predict the aerodynamics of the flap-edge geometry is method 8. Method 8 contains 70% of the data of the full training set. As is apparent by the error bars in Fig. 11, the error is low for most cases and is well within the acceptable error, even for cases that are not in the training set. The C_l rms errors are exceptionally high for cases 6, 9, and 22, but the drag and moment coefficient predictions are very good. Overall, the prediction characteristics are good for this method.

Several other approaches of choosing subsets for the training set were used. These methods range from 48 to 74% of the entire

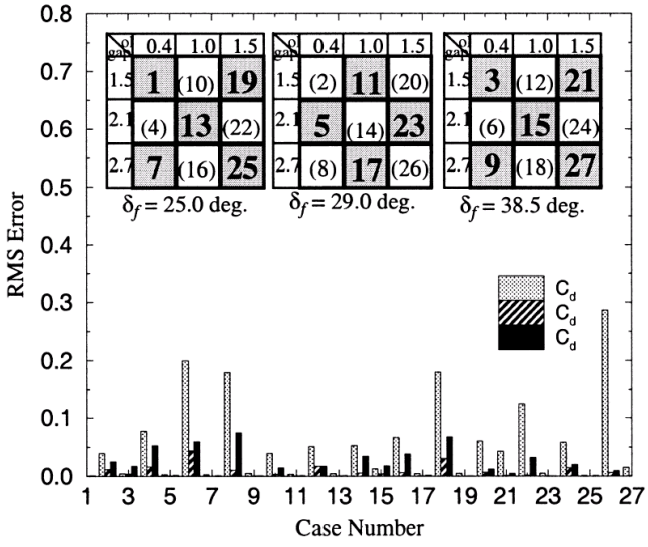

Fig. 10 Summary of rms error from neural-network prediction of aerodynamic coefficients for method 5.

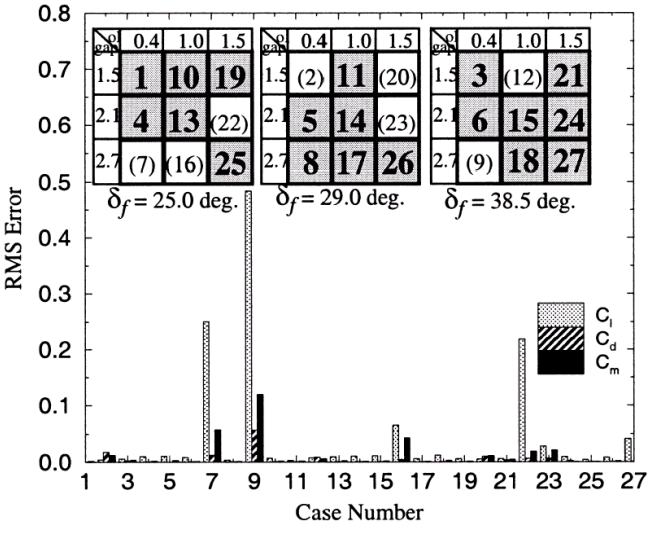

Fig. 11 Summary of rms error from neural-network prediction of aerodynamic coefficients for method 8.

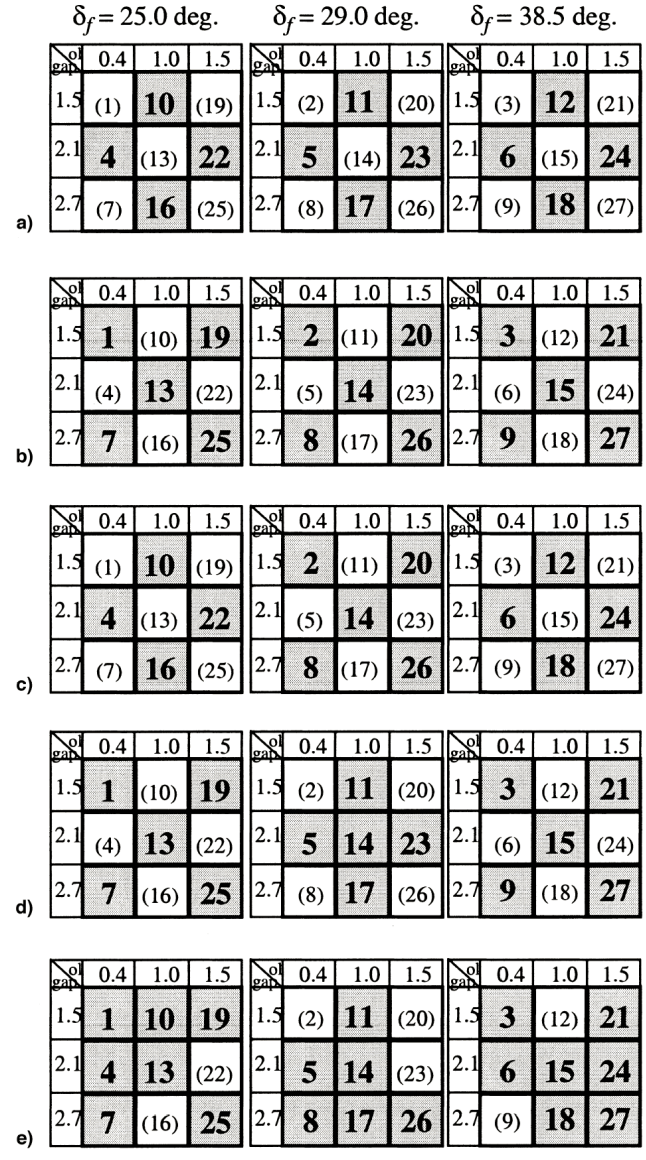

Fig. 12 Additional methods to create training sets. Methods: a) 3, b) 4, c) 6, d) 7, and e) 9.

training set. The various methods are shown in Fig. 12. The mean and standard deviation of each method's rms error of the lift, drag, and moment coefficients were calculated to evaluate the accuracy of the predictions obtained using the various training sets. As expected, method 1 (refer to Fig. 8) has the lowest mean and standard deviations of rms errors for all three aerodynamic coefficients because the training set contains 100% of the data, as shown in Fig. 13 for the 6-deg-deflection slat. The highest rms error is obtained by method 4, which contains 55% of the data. This method is generated by choosing the corners and the center of each deflection matrix as shown in Fig. 12. By examining this method, one sees that there is not enough representation of the $ol_f = 1.0\%c$ or $\text{gap}_f = 2.1\%c$ data to train the neural networks. This led to the more carefully chosen methods of 5, 8, and 9.

Method 5 (Fig. 10) is the most accurate method of training if only ~50% of the data is available. Method 5 consists of 52% of the data. If more computer resource are available, methods 8 and 9 can be used to train, because they have the lowest mean rms errors besides method 1. Methods 8 and 9 contain 70 and 74% of the entire training set, respectively, and their mean rms error is within the acceptable error. Thus, with only 70% of the sparse training set, the neural networks can accurately predict the computational database for this multi-element airfoil.

Example of Neural Network Prediction

The neural network's ability to predict the aerodynamics of a high-lift rigging that is not included in the training set is tested by training the neural network with method 8, which contains 70% of the full training set and comparing the predicted values denoted by an open square with the INS2D calculated values denoted by a filled diamond in Fig. 14. The specific configuration studied consisted of a flap high-lift setting of $\delta_f = 27.0$ deg, $\text{gap}_f = 2.4\%c$, $ol_f = 1.1\%c$, which is not used to train the neural networks as shown in Fig. 14. The lift coefficient vs angle of attack for both the INS2D and neural network calculations are shown in Fig. 14a. The neural network accurately predicts C_l for all angles of attack tested. In this case, the pressure difference rule predicted $\alpha = 10.0$ deg to be the location of maximum lift. Thus, the neural network was only tested from $\alpha = 0.0$ to 10.0 deg. The neural network did not predict the drag coefficient exactly as seen in Fig. 14b; however, the neural network did predict the same trend as the INS2D results. The neural network underpredicted drag for all angles of attack except $\alpha = 2.0$ deg, where it predicted the value very close to the numerical value. The pitching moment, except for a small C_{m_0} shift, was accurately predicted by the neural network, as illustrated in Fig. 14c. The lift-to-drag ratio prediction shows the same trends as the INS2D calculated values, except for the value at $C_l = 2.46$. $(L/D)_{\text{nn}}$ is also calculated from $C_{l_{\text{nn}}}/C_{d_{\text{nn}}}$) and is also shown in Fig. 14d (denoted with an open

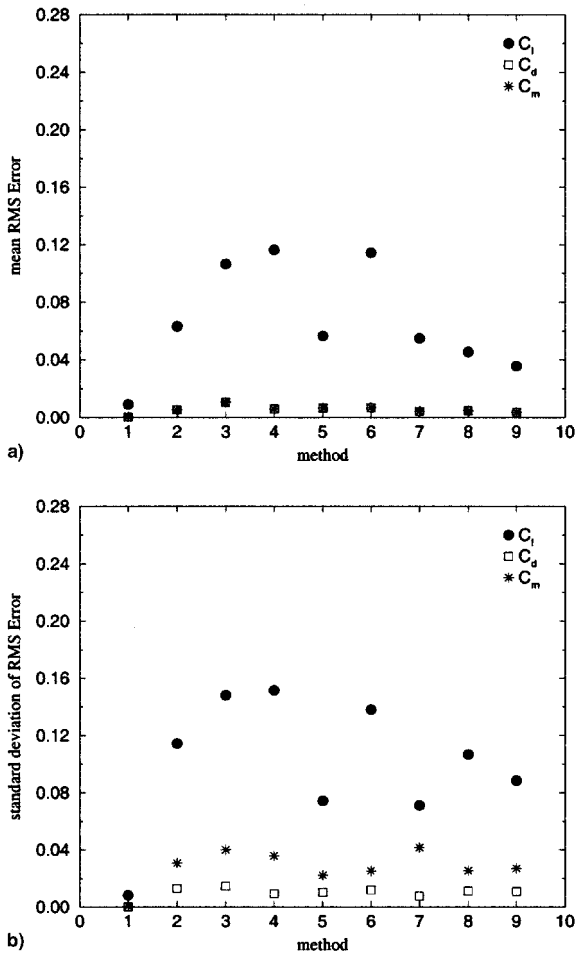

Fig. 13 a) Mean and b) standard deviations of the rms error for various subsets of the training data for slat deflection of 6 deg.

circle) to test if the neural network to predict L/D is necessary. These predicted values are inaccurate. This occurs because the drag coefficient was predicted inaccurately and the errors are amplified in calculating $C_{l_{\text{nn}}}/C_{d_{\text{nn}}}$. Overall, however, the neural networks accurately predicted the aerodynamics of a high-lift rigging that is not included in the training set that contains only 70% of the data.

The neural networks can also be used to find the angle of attack where $C_{l_{\max}}$ occurs by finding the angle of attack where $\Delta C_p = -13.0$ with nearly no additional computer resources. An additional neural network is trained to predict ΔC_p. For this high-lift configuration, the neural networks predict $C_{l_{\max}} = 3.61$ at $\alpha = 10.56$ deg. To check the accuracy of the neural networks, INS2D calculations are performed. The INS2D results show that $C_{l_{\max}} = 3.57$ and occurs at $\alpha = 11.04$ deg. The neural networks predict $C_{l_{\max}}$ within the acceptable error of ± 0.06 for $C_l \geq 0.95 C_{l_{\max}}$, but underpredicted the angle of attack for $C_{l_{\max}}$.

Optimization with Neural Networks

Neural networks are also used in this study to develop an optimization process that will reduce the amount of computational data required to find the highest value of the objective function.[14,15] The neural networks are used to represent the design space and then a gradient-based optimizer, non-linear programming solver (NPSOL)[31] and/or a simple genetic algorithm[§] is deployed to search the design space for an optimal solution. For these optimizers to determine the optimal solution, many function evaluations need to be calculated. This can be expensive when using CFD because a new grid needs to be generated and the flow solution must be solved for each evaluation. However, by integrating a trained neural network

[§] Cheung, S., private communication, Oct. 1998.

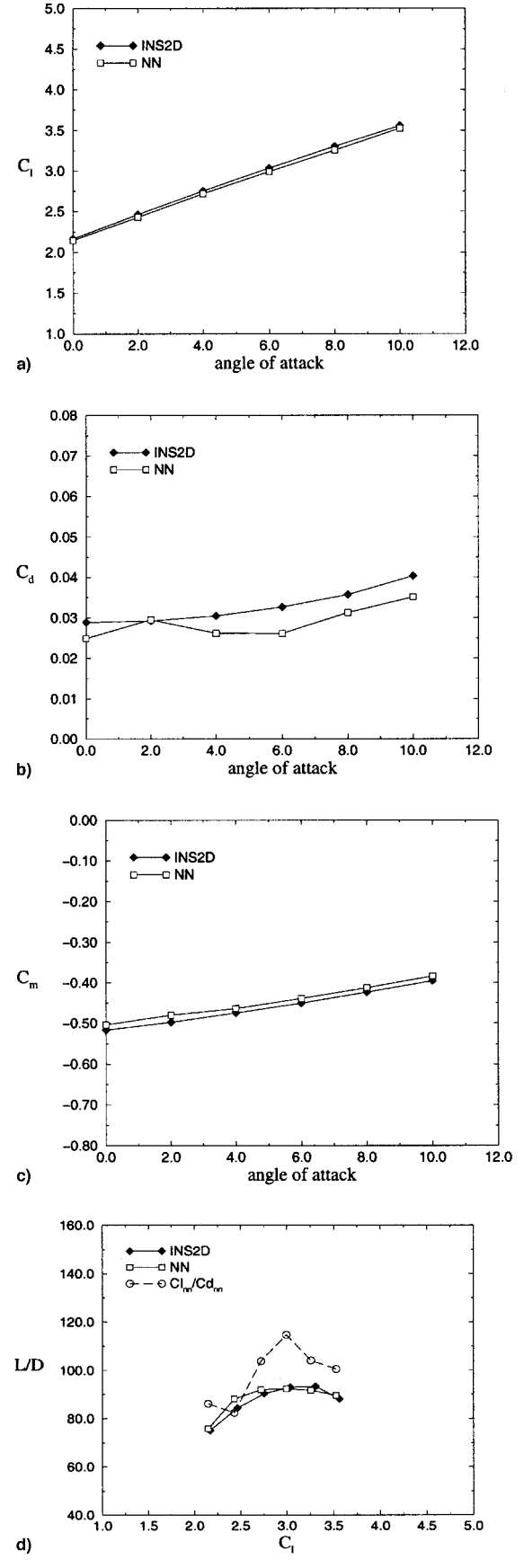

Fig. 14 Comparison of aerodynamic characteristics computed using INS2D and neural nets for a configuration not included in the training set, δ_s = 6.0 deg, gap_s = 2.0%c, ol_s = −0.05%c, δ_f = 25.0 deg, gap_f = 2.1%c, ol_f = 1.0%c: a) Lift vs angle of attack, b) drag vs angle of attack, c) pitching moment vs angle of attack, and d) lift-to-drag ratio.

to the optimizers, the information required can now be quickly determined from the neural network.

This current study integrates the genetic algorithm to the neural networks and then NPSOL is used. The genetic algorithm is used to approximate the global maximum. The solution obtained from the genetic algorithm is then used as a starting point for NPSOL. NPSOL is then expected to find a more accurate maximum. The gradient-based optimizers do not guarantee a global maximum, but only an improvement; whereas simple genetic algorithms find an approximation of the global. Executing the gradient-based optimizer after the genetic algorithm is a relatively quick and efficient way to find a more accurate maximum.

The current results are compared with a previous optimization study that integrated only a gradient-based optimizer with the neural networks.[14,15] In both studies the design variables are the flap deflection, gap, overlap, and angle of attack. The objective function is to maximize the lift coefficient with a constraint on the pressure difference, $\Delta C_p \geq -13.0$. Method 9 was used as the training set for the neural networks. The genetic algorithm determined that the modified configuration was $\delta_f = 35.8$ deg, $\text{gap}_f = 2.10\%c$, $ol_f = 1.14\%c$, $\alpha = 9.73$, and $C_l = 3.96$. These were used as the starting values for NPSOL and only one optimization run was executed. The modified configuration was found to be $\delta_f = 36.0$ deg, $\text{gap}_f = 2.14\%c$, $ol_f = 1.20\%c$, $\alpha = 9.82$, and $C_l = 3.97$. The previous study (without genetic algorithms) performed five NPSOL iterations and determined that the best improvement was $\delta_f = 37.5$ deg, $\text{gap}_f = 2.08\%c$, $ol_f = 0.40\%c$, $\alpha = 9.00$, and $C_l = 3.94$. By including the genetic algorithm into the optimization procedure, the modified lift coefficient was increased by 0.76%.

In addition, the new optimization process that includes the genetic algorithms is higher in fidelity and is also more efficient. The total CPU time to execute the optimization procedure with only NPSOL integrated to the neural networks is 377.6 s,[14,15] and the time to execute the optimization process that included the genetic algorithm is only 115.2 s, which is a 69.5% decrease in CPU time.

Conclusions

Neural networks were used to fill in a design space of computational data to optimize the flap position for $C_{l_{\max}}$ for a multi-element airfoil. Because numerical inaccuracies were identified for the computational method near maximum lift, a semi-empirical maximum lift criteria, designated PDR, was applied to the computational data. Although the PDR was applied to all three elements, the slat was found to be the critical element in locating maximum lift for this airfoil. The neural networks were initially trained with only three values for the flap deflection, gap, and overlap at 10 angles of attack. These trained networks predicted the aerodynamic lift, drag, and moment coefficients to within the experimental error (roughly 1.7% of $C_{l_{\max}}$) for the three-element airfoil. Moreover, through selective reductions of the database, accurate neural-net predictions were obtained when using only 52–70% of the computed data in the basic configuration matrix. Finally, the neural networks were integrated with a genetic algorithm and a gradient-based optimizer to find the optimal high-lift rigging. In comparison with a previous optimization process that did not include the genetic algorithm, the optimization produced a configuration with a 0.76% increase in lift while decreasing the CPU time for the optimization by 69.5%.

Acknowledgments

This work would not have been possible without the helpful discussions and suggestions from James C. Ross, Stuart E. Rogers, and Charles C. Jorgensen at NASA Ames Research Center.

References

[1]Ying, S. X., "High Lift Challenges and Directions for CFD," *Proceedings of AIAA/NPU Atmospheric Flight Mechanics Conference*, Northwestern Polytechnical Univ., China, 1996, pp. 164–182.

[2]Eyi, S., Lee, K. D., Rogers, S. E., and Kwak, D., "High-Lift Design Optimization Using Navier–Stokes Equations, *Journal of Aircraft*, Vol. 33, No. 3, 1996, pp. 499–504.

[3]Besnard, E., Schmitz, A., Boscher, E., Garcia, N., and Cebeci, T., "Two-Dimensional Aircraft High Lift System Design and Optimization," AIAA Paper 98-0123, Jan. 1998.

[4]Kung, S. Y., *Digital Neural Networks*, Prentice–Hall, Englewood Cliffs, NJ, 1993.

[5]Mehrotra, K., Mohan, C. K., and Ranka, S., *Elements of Artificial Neural Networks*, MIT Press, Cambridge, MA, 1997.

[6]Zeidenberg, M., *Neural Networks in Artificial Intelligence*, Ellis Horwood, New York, 1990.

[7]LaMarsh, W. J., Walsh, J. L., and Rogers, J. L., "Aerodynamic Performance Optimization of a Rotor Blade Using a Neural Network as the Analysis," AIAA Paper 92-4837, Sept. 1992.

[8]Faller, W. E., and Schreck, S. J., "Real-Time Prediction of Unsteady Aerodynamics: Application for Aircraft Control and Maneuverability Enhancement," *IEEE Transactions on Neural Networks*, Vol. 6, No. 6, 1995, pp. 1461–1468.

[9]McMillen, R. L., Steck, J. E., and Rokhsaz, K., "Application of an Artificial Neural Network as a Flight Test Data Estimator," *Journal of Aircraft*, Vol. 32, No. 5, 1995, pp. 1088–1094.

[10]Steck, J. E., and Rokhsaz, K., "Some Applications of Artificial Neural Networks in Modeling of Nonlinear Aerodynamics and Flight Dynamics," AIAA Paper 97-0338, Jan. 1997.

[11]Rai, M. M., and Madavan, N. K., "Application of Artificial Neural Networks to the Design of Turbomachinery Airfoils," AIAA Paper 98-1003, Jan. 1998.

[12]Ross, J. C., Jorgenson, C. C., and Norgaard, M., "Reducing Wind Tunnel Data Requirements Using Neural Networks," NASA TM 112193, May 1997.

[13]Jorgensen, C. C., and Ross, J. C., "System and Method for Modeling the Flow Performance Features of an Object," U.S. Patent 5,649,064, filed July 1997.

[14]Greenman, R. M., "Two-Dimensional High-Lift Aerodynamic Optimization Using Neural Networks," Ph.D. Dissertation, Aeronautical and Astronautical Engineering Department, Stanford Univ., Stanford, CA, May 1998; also NASA TM 112233, June 1998.

[15]Greenman, R. M., and Roth, K. R., "High-Lift Optimization Design Using Neural Networks on a Multi-Element Airfoil," *Proceedings of the Computers in Engineering Conference*, American Society of Mechanical Engineers, DETC98/CIE-6006, Sept. 1998.

[16]Rogers, S. E., Menter, F., Durbin, P. A., and Mansour, N. N., "A Comparison of Turbulence Models in Computing Multi-Element Airfoil Flows," AIAA Paper 94-0291, Jan. 1994.

[17]Rogers, S., *Manual for the OVERMAGG Script System*, NASA Ames Research Center, Moffett Field, CA, July 1997.

[18]Rogers, S., "Progress in High-Lift Aerodynamic Calculations," AIAA Paper 93-0194, Jan. 1993.

[19]Chan, W. M., Chui, I. T., and Buning, P. G., "User's Manual for the HYPGEN Hyperbolic Grid Generator and the HGUI Graphical User Interface," NASA TM 108791, Oct. 1993.

[20]Suhs, N. E., and Tramel, R. W., "PEGSUS 4.0 User's Manual," Arnold Engineering and Development Center, TR-91-8, Arnold AFB, TN, Nov. 1991.

[21]Rogers, S. E., and Kwak, D., "An Upwind Differencing Scheme for the Steady State Incompressible Navier–Stokes Equations," *Journal of Applied Numerical Mathematics*, Vol. 8, Aug. 1991, pp. 43–64.

[22]Rogers, S. E., and Kwak, D., "An Upwind Differencing Scheme for the Time Accurate Incompressible Navier–Stokes Equations," *AIAA Journal*, Vol. 28, No. 2, 1990, pp. 253–262.

[23]Spalart, P. R., and Allmaras, S. R., "A One-Equation Turbulence Model for Aerodynamic Flows," AIAA Paper 92-0439, Jan. 1992.

[24]Dominik, C., "Application of the Incompressible Navier–Stokes Equations to High-Lift Flows," AIAA Paper 94-1872, June 1994.

[25]Norgaard, M., Jorgensen, C. G., and Ross, J. C., "Neural-Network Prediction of New Aircraft Design Coefficients," NASA TM 112197, May 1997.

[26]Rumsey, C. L., Gatski, T. B., Ying, S. X., and Bertelrud, A., "Prediction of High-Lift Flows Using Turbulent Closure Models," AIAA Paper 97-2260, June 1997.

[27]Valarezo, W. O., and Chin, V. D., "Method of Prediction of Wing Maximum Lift," *Journal of Aircraft*, Vol. 31, No. 1, 1994, pp. 103–109.

[28]Fiddes, S. P., Kirby, D. A., Woodward, D. S., and Peckham, D. H., "Investigation into the Effects of Scale and Compressibility on Lift and Drag in the RAE 5m Pressurized Low-Speed Wind Tunnel," *Aeronautical Journal*, Vol. 89, March 1985, pp. 93–108.

[29]Valarezo, W. O., Dominik, C. J., McGhee, R. J., Goodman, W. L., and Paschal, K. B., "Multi-Element Airfoil Optimization for Maximum Lift at High Reynolds Numbers," AIAA Paper 91-3332, Sept. 1991.

[30]Rogers, S. E., "Progress in Computational Fluid Dynamics for High-Lift Aerodynamics," NASA TM 110-419, Dec. 1996, pp. 73, 74.

[31]Gill, P. E., Murray, W., Saunders, M. A., and Wright, M. H., "User's Guide for NPSOL 5.0: A Fortran Package for Nonlinear Programming," Dept. of Operations Research, TR SOL 94, Stanford Univ., Stanford, CA, 1994.