Aerodynamic and Aeroacoustic Optimization of Rotorcraft Airfoils via a Parallel Genetic Algorithm

Brian R. Jones,1 William A. Cready,1 and Anastasios S. Lyrintzis2
Purdue University, West Lafayette, Indiana 47907-1322

A parallel genetic algorithm (GA) methodology was developed to present a family of two-dimensional aircraft designs that address rotorcraft aerodynamic and aeroacoustic concerns. The GA operated on 20 design variables, which constituted the control points for a spline representing the airfoil surface. The GA took advantage of available computer resources by operating in either serial mode, where the GA and function evaluations were run on the same processor or “master/worker” parallel mode, where the GA runs on the master processor and function evaluations are conducted independently on separate worker processors. The multiple objectives of this work were to minimize the drag and overall noise of the airfoil. Constraints were placed on lift coefficient, moment coeff cient, and boundary-layer convergence. The aerodynamic analysis code XFOIL provided pressure and shear distributions in addition to lift and drag predictions. The aeroacoustic analysis code WAFDOP provided thickness and leading-edge note predictions. The airfoils comprising the resulting Pareto-optimal set exhibited favorable performance when compared with typical rotorcraft airfoils under identical design conditions using the same analysis engine. The relationship between the quality of results and the analyses used in the optimization is also discussed. The new airfoil shapes could provide starting points for further investigation.

Nomenclature

$\alpha$ = airfoil chord length
$c_d$ = sectional drag coefficient
$c_l$ = skin-friction coefficient
$c_{sl}$ = sectional lift coefficient
$c_{se}$ = section moment coefficient about $\frac{1}{2}$ chord
$c_p$ = pressure coefficient
$c_f$ = function coefficient
$E_c$ = constraint function
$M$ = chord Mach number
$m$ = number of CPUs used during a parallel run
$m_{max}$ = number of constraints
$m_{d}$ = number of design flow conditions
$m_{obs}$ = number of observer locations
$P_e$ = scaled penalty factor
$P_0$ = penalty function
$R_e$ = Reynolds number based on chord
$v_f$ = fully drawndown coefficients
$v_a$ = boundary-layer-edge velocity
$v_{mean}$ = normalized airfoil station
$v_{sur}$ = mean value of surface separation, projected along streamline axis
$v_{cor}$ = normalized airfoil ordinate
$a$ = geometric angle of attack
$\Psi$ = azimuth angle
$\gamma$ = for all instances in the set
$L$ = there exists in the set

Subscripts:

[lower case] = lower surface of airfoil

1Presented at Page 99-001 at the AIAA/NASA/SSMM 70th Symposium on Multidisciplinary Analysis and Optimization, St. Louis, MO, 3-4 September 1999; received 21 March 1999; reviewed received 6 April 2000, accepted for publication 1 April 2000. Copyright © 2000 by the authors. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission.
2Graduate Student, School of Aeronautics and Astronautics, currently Analysis, Aeroacoustics and CFD, Los Alamos National Aeronautics Company, Fort Worth, TX 76101, Student Member AIAA.
3Assistant Professor, School of Aeronautics and Astronautics, 1202 Griscom Hall, Senior Member AIAA.
4Associate Professor, School of Aeronautics and Astronautics, 1202 Griscom Hall, Associate Fellow AIAA.

Introduction

This paper discusses the application of a parallel genetic algorithm (GA) to a multiobjective airfoil design problem. Airfoil shapes were designed to minimize the drag coefficient and the overall averaged sound-pressure level (OASPL) for three flow conditions representative of those experienced by a helicopter rotor airfoil, when the helicopter is in forward flight. An angle of attack, a Mach number, and a Reynolds number describe these flow conditions.

Direct airfoil design presents an initial airfoil shape to improve performance of the airfoil. Although successful, this approach generally produces airfoils deviating only slightly from the initial design. Calculus-based search methods can only encounter this limitation because they follow the nearest local optimum to the original design. Airfoil features, the trailing-edge tabs, drop-stubs, and complex camber would be difficult to discover using a traditional method that perturbs a known shape.

In contrast, inverse-airfoil design produces an airfoil whose pressure distribution matches a desired distribution. The inverse approach may reassemble that no physical shape can produce. The inverse approach suffers further disadvantages when applied to multiobjective design. To obtain a family of airfoil shapes, the designer uses inverse methods must specify multiple pressure profiles that represent the many possible compromises between objectives.

This research uses a direct problem formulation by defining an airfoil’s surface using the location of spline control points. These control points constitute the design variables. Employing the GA to solve the direct problem allows the discovery of shapes unlikely to be found using other methods.

Genetic Algorithms

Since its first description, the genetic algorithm has been applied to many engineering optimization problems.1 Based on Darwin’s survival of the fittest concept, the GA performs optimization tasks by “evolving” a population of highly fit designs over many generations. A GA has the ability to search highly multimodal, discontinuous design spaces. The GA also locates designs at or near the global optimum without requiring an initial design point.
The GA represents design variables as strings of binary numbers, which serve as chromosomes. Initially, the GA randomly generates a population of individuals. After decoding the chromosome of each individual into the corresponding design variables, each design is analyzed to determine a fitness value. Individuals with better fitness values are considered more optimal so that this fitness value must reflect both the objective of the design and any constraints imposed upon the design.

This employs selection, crossover, and mutation operators to perform a selection. The selection process prioritizes the survival-of-the-fittest function that allows better individuals to survive and reproduce. Crossover combines portions of chromosomes from the surviving parent designs to form the next generation of designs combining desirable features of good designs on average. Not always, results in better designs. This gives the GA its stochastical capability. The mutation operator is used quite infrequently, as in nature, but this operator can mutate a binary bit in a chromosome to its opposite value (e.g., 0 to 1), which can introduce beneficial design traits that did not exist in the current population. If the mutated trait is poor, the design with this mutation will be unlikely to survive. This process transforms an initial population of randomly selected designs into a population of individuals that have adapted to their environment by becoming more optimal. Additional details of the genetic algorithm can be found in several texts, such as Ref. 1.

Related Research

Applications of the GA to single-objective aerospace design problems include aircraft aerodynamics for fixed-wing aircraft,14 rotor applications,15 and rotor system design for aerocones.16 GA applications to the multiojective and multidisciplinary design of aircraft have also been investigated. Aircraft design has been a popular application for the GA, although most of these efforts attempted to solve more intrinsic problem17,18 frequently for transonic aircraft. Of those that attempt the direct problem,19 many present as existing shape, thus limiting the chances of locating nontraditional aircraft shapes. Further GA applications have started to explore multiojective and multidisciplinary problems, including aircraft aerodynamic/structural,19 propulsion, cascade design,18 and wing design.20 Rotor blades have also been optimized to reduce vibration loads using a panel GA.20 Additional discussion about GA applications for rotorcraft can be found in Ref. 17.

There is GA and its operators. The GA used in this work utilized an uniform crossover, tournament selection, and elitism.21 Parallel evolution provides efficiency benefits to the GA, and many implementations have been studied. These included distributed coarse grid parallelization schemes,22,23登上,23,24 and island methods. The coarse-grain scheme was adapted for this application. Finally, the use of the GA to solve multiojective design problems has been investigated, the n-brach tournament, a generalization of the two-brach tournament approach, has been used here.

Problem Formulation

For a set of three flow conditions, a family of low-drag, low-noise helicopter rotor airfoils representing the Pareto-optimal set was generated. The airfoil design problem addresses a two-dimensional shape, but the flow conditions used in this work correspond to different aspect angles of a helicopter rotor in forward flight. Aerodynamically, problems are difficult solving, a two-dimensional airfoil, and to specialize a three-dimensional four-airfoil model, described next, was developed to calculate a new measure for each airfoil. This measure included information from two observer locations and made use of the aerodynamic conditions associated with each of the three flow conditions.

Aerodynamic Analysis Methods

Because the quality of any optimization method's results depends on the analyses used, well-established codes were selected. The airfoil analysis code, XFoil,25 formed the core of the analysis. For physically valid airfoils (e.g., no crossing of the upper and lower surfaces, XFoil predicts the aerodynamic coefficients $C_{L}$, $C_{D}$, and $C_{M}$, which allows the Karman-—Tien compressibility correction with a solution generated from closely coupled inviscid and viscous methods. The code utilizes, streaming flight data, using panel methods, the inviscid solution, and an integral boundary-layer method provides the viscous solution. Additionally, XFoil provides low-Reynolds-number transitional separation bubbles and provides compressible analysis, up to sonic conditions. Despite problems analyzing large regions of separated flow, the viscous capabilities of XFoil allow it to predict stall-like conditions for reasonable aerodynamic shapes with rounded leading edges and gradual changes in surface curvature.

Aerodynamic Analysis Methods

The airfoil's loading and thickness noise are evaluated using the aerodynamic code, WOPWOP. This program evaluates rotor thickness and leading edge undesirable observer locations. The noise calculations depend upon the airfoil thickness, pressure, and shear distributions as functions of distance from leading edge, which are provided by XFoil for this work. Because this work addresses the design of two-dimensional airfoil sections and WOPWOP natively handles a three-dimensional rotor system, the rotor model was simplified. The primary simplification reduces the rotor system to a single rectangular platform, omitting blade bending in the rigid direction from the 75% radius station to the tip. Top effects are of a three-dimensional phenomenon and are ignored here. The airfoil loading computed by XFoil for a flow condition is assumed empanned for all rotor radial stations a...
described by the B-spline is divided into 200 panels for aerodynamic and aeroelastic predictions. This relatively high number of panels allows traditional cosine spacing (with many points near the leading and trailing edges) for the aerodynamic and aeroelastic analyses and provided acceptable resolution for less regular shapes.

To keep the problem tractable, panels must be imposed on the position of each spline control point. Excessively limiting these parameters removes potentially beneficial designs from the search space, whereas excessive freedom wastes computing effort on very irregular designs. Figure 2 displays the limits used in this problem description. The figure also shows the small, limit gap at the trailing edge. The bold line represents the upper and lower limits of the 10 upper surface control points, and the thin line represents the limits for the lower surface. This search space still allows features like trailing-edge radii and large amounts of camber, while eliminating pointed leading edges and other unreasonable sightings.

**Problem Formulation**

Building upon the ideas in Ref. 5, the two-dimensional airflow conditions used here represent the ones experienced at the 75% radius position on a color blade of the example helicopter from Prouty17 in forward flight with and advance ratio of 0.3. Figure 3 illustrates these flow conditions, and Table 1 presents the corresponding constraint limits. These constraints ensure that aircrafts generated by the GA maintain a lift coefficient equal to, or greater than, that of the example aircraft and maintain a maximum coefficient smaller than or equal to that of the example.

Addressing an aerodynamic objective and an aerodynamic objective neglected two fitness functions. The aerodynamic objective sought to minimize the aircraft's drag coefficient at all three flow conditions. Because the GA performs its search using the fitness value to represent a design, this value must reflect the objective function value and any constraint violations. The aerodynamic fitness function was

\[ f = \sum_{i=1}^{3} f_i = \sum_{i=1}^{3} \left( \frac{C_{L,\text{opt}}}{C_{D,\text{opt}}} \right) + \sum_{i=1}^{3} f_i \]

(1)

In this form drag was minimized, subject to constraints imposed via an exterior penalty method, as a sum including all their \((f_i = 1, 2, 3)\) flow conditions. Because \(C_{D,\text{opt}}\) is larger at higher angles of attack, the drag coefficient at each flow condition was scaled by the NASA 0012's drag coefficient at the same flow condition. This scaling provided nearly equal consideration of all three flow conditions.

XFOIL determines boundary-layer properties using an inviscid solution involving both an inviscid and viscous analysis. If the boundary-layer solution failed to converge within a specified number of iterations for a given flow condition, then any resulting aerodynamic data were deemed unreliable for this application. Because of this, two sets of constraint functions were used based on the convergence tolerance of XFOIL's viscous-inviscid boundary-layer iteration scheme. Constraints were enforced via linear penalty functions of the form

\[ f_i = \max(0, g_i) \]

(2)

where \(g_i\) is positive valued when violated. The first set of constraints was used when the boundary-layer solution converged. In this case constraints were enforced for the lift and moment coefficients using two constraint functions, and the third function was set to zero:

\[ R = 1 - \left( g_i / g_i^{\text{converged}} \right) \]

(3)

\[ g_i = \left( g_i / g_i^{\text{converged}} \right) - 1 \]

(4)

\[ g_i = 0 \]

(5)

When the boundary-layer solution failed, the ratio \((C_{L,\text{opt}} / C_{D,\text{opt}})\) was assigned a value of 100 in order to be larger than the nominal value for aircrafts with converged boundary-layer solutions by at least an order of magnitude. The fitness value will then have a large contribution from the \((C_{L,\text{opt}} / C_{D,\text{opt}})\) value, and the airflow design will not be selected at favor of an individual with a converged boundary-layer solution. In this situation the constraint took the form

\[ g_i = 1 \]

(6)

\[ g_i = 0 \]

(7)

\[ g_i = \frac{1}{2} \left[ \left( C_{L,\text{opt}} / C_{D,\text{opt}} \right) - 1 \right] \]

(8)

The third constraint function \(g_3\) reflects how poor the solution is; this allows an airflow with a nearly converged boundary-layer to have a better fitness value than an airflow whose iterative boundary-layer solution failed near the leading edge of the aircraft. This constraint uses the chordwise position of the first boundary-layer velocity prediction during initialization of the injection scheme to measure how far the solution had proceeded for the upper and lower surfaces. The constraint is satisfied when the upper and lower surfaces have a boundary-layer velocity prediction during initialization. Although this does not directly indicate the quality of the boundary-layer solution, this measurement provided the GA with a consistent means of comparing two individuals whose boundary-layer solution has failed.

Drawdown coefficients \(e_i\) in Eq. (1) scaled the penalty in the magnitude of the objective. The penalty associated with Eq. (8) multiplied a large value of \(e_i\) to encourage designs that would allow solution of the boundary-layer calculations.

The aeroelastic fitness function incorporated the multiple flow conditions as well as multiple observer locations. The objective sought to minimize the aircraft's OASPL values, as summed over multiple flow conditions and as seen by multiple observers. Note that OASPL includes thickness and loading noise, but duct noise include
the effects of high-speed impulsive (HSI) noise (i.e., shock-wave noise) and blade-vortex interaction (BVI).

\[
\text{minimize } P^* = \sum_{i=1}^{n} \left( c_i \left( \frac{s_i}{c_{\text{max}}(i)} \right) \right) + \sum_{i=1}^{n} (\text{DASPL} + \epsilon_i \text{max}(0, \text{Eq1}))
\]  

Two observer locations were placed 50 m from the hub of the simplified WOPROW rotor model, one in the rotor plane and one below the rotor along a ray angular at 45 deg to the rotor plane. Because DASPL combines loading and thickness noise, using multiple observer locations presented a single source from dominating the fitness function. For example, observations in the rotor plane are dominated by thickness noise, whereas loading noise dominates observations below the rotor. The directivity of DASPL varies smoothly with the changing angle of an observer located below the rotor plane. Figure 4 illustrates the smooth variation of the aerocoustic fitness function for two of the aircrafts eventually generated by the GA. The simplified WOPROW rotor model for the two-dimensional aircraf t has no azimuthal directivity of the noise because of the constant loading aspects just described. Thus, two observer positions were deemed adequate for this work, whereas two observers might not be enough for more complicated problems (i.e., including BVI or instering a three-dimensional rotor blade design).

A scaled penalty factor \( P^* \) enforces aerodynamic constraints for the aerocoustic fitness function so that penalties imposed on the aerodynamic fitness have the same scale as those imposed on the aerodynamic fitness:

\[
P^* = f_j \left( \sum_{i=1}^{n} c_i \left( \frac{s_i}{c_{\text{max}}(i)} \right) \right)
\]  

The aerocoustic fitness function in Eq. (9) contains one additional constraint not addressed in the aerodynamic fitness function. A linear penalty function enforces a boundary-layer separation constraint \( \epsilon_s \), when more than 30% of the total aircraf t surface (both upper and lower) projected along the ordinate axis experiences separated flow. This reduces the artificial benefit to thickness noise predictions obtained when 2D codes acts the separation coefficient to zero at areas of separated flow. This penalty is scaled to the order of the objective function using the drawdown coefficient \( c_s \):

\[
\epsilon_s = (s/c_s) \times 0.3 - 1
\]  

Implementation

To solve this multiobjective aircraf t problem, a parallel GA was developed that incorporated some necessary additional features. A unique variable encoding for the binary chromosomes followed a Gray-coding scheme to avoid Hamming distance issues between adjacent variable values. \( \epsilon \) Empirical derived relationships for the population size and mutation rate were used. Features were developed to deal with the close coupling of the two objectives and with the difficulty in obtaining an initially viable population. These include an adaptive single objective to multiobjective fitness evaluation and the "reparenting" of infeasible individuals. Reference 29 includes additional details of these special features.

Genetic Operators

The GA employed an \( n \)-branch tournament selection, where \( n \) corresponds to the number of objectives considered. This method is an extension of the two-branch tournament selection.\(^{27}\) After placing the entire current population in a "pot," a user-defined number of individuals is randomly selected without replacement from this pot. These individuals compete, with the most fit individual surviving as a parent. Every member of the population competes on one of the \( n \)-fitness functions. Completing the selection process for a given branch leaves the pot empty. After the pot is refilled using the current population, selection continues using the next branch. Figure 5 illustrates the process when minimizing two fitness functions.

After selection two members of the parent pool are randomly selected, without replacement, and "mated" in order to pass on their traits to two children. This is implemented using uniform crossover.\(^{29}\) Mutation occurs with a very low probability. This operation changes a bit in the chromosome to its opposite value (i.e., 1 to 0, or 0 to 1), which helps the evolution proceed toward a global solution by introducing binary patterns that may not exist in the present population. This process repeats until a new generation of individuals is created. The implemented GA also included an elitism operator. This operator replaces an individual from each new population with the best individual of the previous generation. When performing a multiobjective optimization, multiple replacements occur as the best individual in each fitness function survives as an "elite." The addition of elitism to the GA was observed to benefit the solution of multiobjective problems.

Parent Set Determination

To be considered Parent optimal, individuals must be both feasible and non-dominated.\(^{27}\) A design is considered non-dominated if no
other design exists that is better at all objectives. Mathematically, a design with a vector of objective values $\mathbf{v}$ is dominated by a design with a vector of objective values $\mathbf{v}'$ if the following condition is satisfied:

$$ v_{i} \leq v'_{i}, \quad 3 \leq i \leq m, \quad i = 1, 2, \ldots, m $$

(12)

The GA evaluates thousands of designs during a run, and any that are dominated design solutions during the run is of interest. Consequently, the GA stores the set of feasible nondominated individuals as a non-dominated set. This data structure best handles the fluctuating length of this set, as members are inserted and removed according to the conditions of Eq. (12). Evolution halts when either the maximum number of generations is reached or a stopping criteria is satisfied. Here, the stopping criterion is satisfied when the approximate Pareto-optimal set ceases to change.

Feasibility Handling Features

Although the direct airflow design problem proved challenging because of the limits of the analytic tools, the design space size, and the degree of coupling between objectives. Consequently, a system of feasibility classes was developed. The range of possible airflow shapes effectively required the GA to solve two separate problems. First, the GA needed to evolve a population of airflow of which XFOIL could generate sensible solutions. Only then could the GA solve the multidisciplinary problem because WINGSPAN calculations received information from XFOIL. An airflow shape upper and lower surfaces of cross presents a physically impossible shape, whereas the shape in Fig. 1 is typical of on surface prompting numerical instabilities within XFOIL. Shapes of both kinds prevent successful aerodynamic analysis.

Six classes were defined to allow constraint handling: 1) unknown—initialization class, no analysis performed; 2) infeasible—physically impossible shape; 3) unrelatable—solution aborted as a result of numerical instability in XFOIL; 4) unstable—computational solution attempted, but boundary-layer solution did not converge; 5) infeasible—converged solution returned with violated constraints, and 6) feasible—converged solution returned without violated constraints.

If the analysis routines deemed a design impossible or unrelatable in the generation, the design is replaced with another randomly generated individual. During successive generations, the GA requires the crossover and mutation operators to the parent designs to form a different design. Because each parent design was at least soluble, their design chromosones contain some valuable genetic material. That property prevents the loss of this information. Random replacement after the initial generation would reduce the selection pressure that drives the evolution process.

Complementing reparation, an adaptive fitness function evaluation scheme was developed. The GA first sought to minimize only the aerodynamic forces until at least 60% of the population maintain convergent boundary layers, and then the GA begins multidisciplinary optimization at the first generation. This proved advantageous here because the aerodynamic analysis requires pressure and shear data from the aerodynamic analysis. By comparing on a single objective first, the GA was able to reach a viable design space. After obtaining a usable population, both fitness functions can be evaluated, and multidisciplinary optimization can ensue.

Parallelization

Increased computational cost accompanies using a GA to solve the direct airflow design problem. Adequate resolution of the spline control points that determine the airflow surface requires a large chromosome and correspondingly large population. When evolved over enough generations to satisfactory genotype the Pareto-optimal set, solving the problem requires tens of thousands of fitness evaluations. Because the fitness evaluation of each individual in the population is independent of other individuals, parallel processing is well suited to ease the computational load. The Message Passing Interface (MPI) was selected to parallelize the code. A modified worker model was employed. In this approach, the manager node assigns all GA operations and distributes individual airflow designs to the worker nodes for analysis. The sole purpose of a worker node is to calculate the fitness value of its current individual. The manager and workers communicate via blocking communication, which dictates that neither the sending nor receiving processor can begin another task until the communication is complete. Because the GA begins with a random population, large differences in evaluation time can exist among different designs. Dynamic load balancing allows idle workers to begin new tasks without waiting for slower workers to finish.

Results

Because of limited access to computational resources, two separate computer systems were used to generate results. Employing the adaptive single-objective optimization, generating a population in which 60% of the individuals maintained converged boundary-layer solutions was completed on the JSR/PERS machine developed by the Panafra University Department of Electrical Engineering. Several Pentium II (350 MHz) PCs processed 450 generations over approximately 10 days before generating a population of designs meeting the viability condition. This population was then transplanted to the 256-node Sun Ultra Supercomputer Center IBM SP2 to generate multidisciplinary results. An additional 5 and 20 min of computation across 37 available nodes created an approximate Pareto-optimal set consisting of 26 feasible nondominated individuals. Figure 6 displays the Pareto front of this set with three representative airflow. The function space plot compares fitness values, which
account for any penalties. The three highlighted airfoils exhibit the best predicted aerodynamic performance (based on the fitness function $J$); the best predicted aeroacoustic performance (based on $J_a$); and a compromise between aerodynamic and aeroacoustic performance.

Because the GA requires no initial starting point and does not perturb known shapes, the discovery of nontraditional airfoils is possible. This research intended to discover airfoil shapes that would not have been found via traditional optimization approaches, and the resulting airfoils are unique. As a measure of validation, the pressure coefficient distribution at the various flow conditions of the compromise airfoil are displayed in Fig. 7. These profiles behave as expected. Dips in the pressure profile correspond to dips in the airfoil surface. Similar investigations indicated that the flow would separate from the upper surface near the leading edge of the best aerodynamic airfoil in Fig. 8. A slight pressure recovery exists along the upper surface of the aerodynamic best airfoil. Additional validation was made by using the Envia-adaptive integral boundary-layer code MIES9 to predict pressure distributions and aerodynamic coefficients for unconventional airfoils. The predictions of XFOIL and MIES showed good agreement, varying by 2% or less. Reference 29 presents details of these comparisons.

Figure 8 shows the individuals comprising the estimated Pareto-optimal set. Scanning from left to right across each row relates the set as traversed from the best aerodynamic, to worst aerodynamic, to best aeroacoustic, and to worst aeroacoustic. Many of the features evident on the airfoil appear reasonable when examined independently. Most airfoils have a camber toward the leading edge. Increased camber thickness $c_t$ while postulating the camber toward the leading edge reduces the aerodynamic pitching moment. Reduced trailing edge produces a restructuring moment, further reducing $c_t$. Larger radius leading edges help prevent flow separation at higher angles of attack, whereas thinner airfoils have less profile drag and reduced thickness noise. Individuals near the aerodynamic edge of the Pareto-front maintain laminar flow over 80% of the lower surface and 20% of the upper surface for the multiple flow conditions.

Perhaps the most unusual feature are the "waves" in the upper and lower surfaces that grow more pronounced in airfoils toward the aeroacoustic side of the Pareto front. This feature appears responsible for lower estimates of OASPL, even though thickness and loading noise predictions remain essentially constant or increase for airfoils closer to the aeroacoustic best airfoil. To illustrate this, Fig. 9 presents WOPWOP calculated noise values for the aeroacoustic best airfoil. The aeroacoustic fitness function (Eq. 9) is the sum of the OASPL values for both observer locations for each flow condition. The OASPL value for this airfoil is lower than either the thickness or loading measures for most of the observers and flow conditions, suggesting that a cancellation occurs between the thickness and loading noise. This is true for airfoils with the wavy surface feature. Although this cancellation may be nonphysical, it is an interesting, appealing feature that was discovered with the GA. The noise cancellation of these waves needs further study. If the analysis technique, rather than a physical phenomenon, creates this cancellation, using the thickness and loading noise separately in the evaluation of the aeroacoustic fitness instead of OASPL can remedy this.

Comparisons with existing airfoils additionally increase the validity of these nontraditional true airfoil shapes. Figure 10 compares the GA-generated compromise airfoil against the NACA 0012 (Ref. 33) and the Boeing vertical tail (Ref. 10). The NACA 0012 is the airfoil of Pride's sample helicopter. The VR-7 tail is an airfoil shape developed specifically for a helicopter rotor. In the plot, the published results are taken from Refs. 33 and 34 for Reynolds numbers closest to those of the design flow conditions; this provides an idea of how closely the predictions agree with experimental data. No published data for the VR-7 matched the Reynolds number of flow condition 3. The combination of XFOIL and WOPWOP used to provide fitness values for the GA produced the remaining values shown in these plots. The GA compromise airfoil compares favorably in light of the specified design constraints. Before making detailed comparisons, one should note that XFOIL was unable to converge on a boundary-layer solution for the NACA 0012 at the second flow condition (corresponding to $\alpha=180$ deg). This may suggest drag predictions of less than desirable accuracy. Observations taken over several runs indicate that the second flow condition was the most difficult of the three to obtain sufficient boundary-layer solutions.

Although the NACA 0012 performs well in pitching moment, it displays shortcomings in all other categories. The VR-7 performs
well in all categories, with the exception of moment. The GA compromise airfoil is predicted to be quieter than the VR-7, while maintaining good aerodynamic performance. Relaxing the $c_{L_{max}}$ constraint would allow a more direct comparison to the VR-7. Consid-
ering that the GA had no incentive to improve $c_{L_{max}}$, the NACA 0012 $c_{L_{max}}$ constraint, the GA compromise airfoil outperforms favorably. Given the advantage of the $c_{L_{max}}$ constraint, the GA is a designer to consider tradeoffs between the different objectives and select an airfoil suitable to stated needs. For exam-
ple, the best feasible aerodynamic airfoil generated by the GA has a better predicted aerodynamic force than that of the NACA 0012 or the VR-7. Given the level of analysis tools employed, the GA-generated airfoils appear valid. However, these shapes should not be used as final airfoil designs but as starting points for further refinement.

**Discussion**

Even with feasibility classes and heuristic single-objective/multi-objective fitness evaluation, the GA expanded most of its effort evolving a solvable population from a vast design space. At the end of the run, only 75% of the possible designs were feasible designs (visible design with all con-
straints satisfied). These low percentages result because the search ability of the GA that allows for the discovery of nontraditional shapes does not require XFOIL and WOPWOP to evaluate designs that may be beyond their intended scope. The most likely solution to this problem is the development of more robust analysis tools capable of evaluating unusual shapes, although this would add additional computational expense.

**Concerns for Application**

The GA can generate nonconventional shapes because of its population-based search and global optimization behavior. How-
ever, like most optimization techniques, the GA will exploit lim-
itations of the analysis methods and shortcomings in the problem formulation. For example, airfoils can encounter transonic effects at some of the flow conditions, yet wave drag is not addressed with the current analysis. The drag calculations for conventional airfoil shapes may be suspect; higher-order computational fluid dynamics code including turbulence modeling may be needed. Similarly, the current transonic analysis neglects HSL, BVL, vortical/flow noise associated with supersonic-induced unsteady flow. The unsteady noise is only one of the many possible reasons to predict the airfoils' performance. Because of the aforementioned concerns, addi-
tional analysis of the two-dimensional airfoils using higher-order tools and modeling and analysis of a three-dimensional rotor using these shapes are required before the GA-generated airfoils should be considered for application.

**Parallel Execution**

The additional capability obtained through parallel operation cannot be underestimated. The engineering manager gauged GA's achieved a speedup close to the ideal 1/(n – 1) value despite solving the air-
foil problem on an IBM SP2. The test documented in Fig. 11 plots the solution time for five generations of the most critical problems against an increasing number of single processor nodes. Although restricted to the number of processors in current computing allo-
cations, this study shows a strong similarity between the actual and ideal speed-up values.

The performance displayed in Fig. 11 was expected for several reasons. The problem has a significantly high ratio of computation time to communication time. The communication times are on the order of microseconds, but airfoil evaluations averaged approxi-
mately 16 s each on this system. On average, a XFOIL analysis

![Fig. 10 Comparison of GA-generated compromise airfoil to existing airfoils.](image)

**Fig. 11 Distribution of system scalability study, actual and ideal.**

![Fig. 11 Distribution of system scalability study, actual and ideal.](image)
times 3.1 times longer than a WDFOP analysis. For some cases the actual and ideal wall clock times differ slightly. Towards the beginning of the trend line, the difference results from the lack of a two-node (one manager-one worker) computation. For this case the total wall clock time would actually increase above the set of (one node) computation because of the addition of communication time.

The remaining discrepancies result from varying numbers of the fitness evaluations needed during the different runs. Although the same random seed was used in all runs to provide the same initial generation, the differing number of nodes used allowed the GA to process the individuals in different orders. The worker nodes returned in-

dividuals to the manager node as the fitness evaluations were com-
pleted, and the manager placed these in the population in the order they were returned. With different numbers of worker nodes, the re-
turn order varied so that the pairs of individuals selected as parents varied with the number of workers. For example, this reordering re-
quired the repertory of 1724 individuals during the 32-node run, whereas the 16-node run replaced 822 individuals. This negated much of the expected speed up between the 16- and 32-node runs.

The discrepancy resulting from workers idled between generations while the manager performs the genetic and statistical operations also remains. Despite these details, the dynamically load balanced parallel GA scales quite well as the number of CPUs increases.

Conclusions

Given a problem representing a helicopter rotor airfoil and using the codes XFOIL, and WDFOP, the GA generated a set of rotor airfoil shapes representing compromises between aerodynamic ef-

ficiency and minimum noise. Based upon predictions from the flow-

order analysis tools employed, the resulting nontraditional shapes appear to offer good aerodynamic and acoustic performance.

The n-brach test床位 selection allowed a range of designs to be
generated in a single run of the GA, which lets designers choose an airfoil best suited to their needs. Also, the inclusion of advanced fles-

sibility handling and genetically adapted fitness function eval-

uation allowed the GA to successfully evolve a population in a design space containing large ranges of sensible airfoil shapes. Among the genes, the most sensitive to changes in the airfoil was the twist of the airfoil surfaces, a result that is not surprising. A detailed analysis of the airfoil shapes is needed to determine how much of the fitness function is related to each gene. Depending upon how much of the fitness function is related to each gene, the problem's scalability closely matched the ideal value of 1 (0.1-0.5).

The parallel GA generated solutions that would not have been obtained using other optimization methods and did so in reasonable times.

Acknowledgments

Some calculations were performed on the 12-node [Para] SP2 at the San Diego Supercomputer Center, a member of the National Initiative for Advanced Supercomputing Infrastructure. Additional calculations were performed on the PARAGON cluster in the School of Electrical Engineering at Purdue University with the assistance of Hank Dettl and his students. Mark Tweet of the Massachusetts Institute of Technology-supplied the code [XFOIL]. Kenนอน of NASA's Langley Research Center provided the code [WDFOP] along with support and advice for this endeavor.

References

[3] O'yoshi, S. and Takahashi, S., "Genetic Optimization of Target Prec-


ings of the 1997 IEEE International Conference on Evolutionary Compu-

neering, vol. 5, No. 4-6, 1994, pp. 415-423.
This new volume, edited by the Director of the University of Virginia's Center for Advanced Computational Technology at NASA Langley Research Center, focuses on the component technologies that will play a major role in structures technology for future aerospace systems. Contributors use case histories to demonstrate the technology's development and carry it through to the current state of the art. Each chapter describes current capabilities, deficiencies and barriers; current research activities; future directions of development; and applicability of the technology in the future—both near- and far-term.

Contents:
Perspectives on Structures Technology • Smart Air and Space Structures • Affordable Composite Structures • Extreme Environments Structures • Flexible Load-Beating Structures (including Inflatable) • Aging Systems and Sustainment Technology • Computational Structures Technology

Progress in Astronautics and Aeronautics
ISBN: 1-56347-166-4
List Price: $114.95
AIAA Member Price: $74.95
Source Code: 945