JOURNAL OF AIRCRAFT
Vol. 33, No. 5, September–October 1996

Parallel Genetic Algorithm Implementation in Multidisciplinary Rotor Blade Design

Jongsoo Lee* and Prabhat Hajela†
Rensselaer Polytechnic Institute, Troy, New York 12180

An adaptation of genetic algorithms as an optimization tool for large-scale multidisciplinary design problems is described. The design of a hingeless composite rotor blade is used as a test bed for this class of problems, where the formulation of the objective and constraint functions requires the consideration of disciplines of aerodynamics, performance, dynamics, and structures. A rational decomposition approach based on the use of neural networks is proposed for partitioning the large-scale multidisciplinary design problem into smaller, more tractable subproblems. A design method based on a parallel implementation of genetic algorithms is shown to be an effective strategy, providing increased computational efficiency, and a natural approach to account for the coupling between temporarily decoupled subproblems.

Nomenclature

C_T = thrust coefficient
HP_a = horsepower available
HP_{HOGE} = horsepower in hover out-of-ground-effect
HP_{req} = horsepower required
I = rotor blade mass moment of inertia
L, U = lower and upper bounds on design variable, respectively
$[T]$ = transition matrix of interconnection weights
θ_0 = collective pitch angle
θ_{1c}, θ_{1s} = cyclic pitch components
σ = rotor solidity
ψ = azimuthal angle

Introduction

AN important area of research that has received considerable recent attention is the development of optimization methodology applicable to large-scale multidisciplinary systems.[1,2] The initiative has been largely motivated by a recognition that the design and development of a complex engineering system can no longer be conducted by handling its different subsystems in isolation. Design synthesis in multidisciplinary systems is typically characterized by a large number of design variables and constraints; additionally, there are complex interactions between the participating subsystems, which must be both identified and then suitably represented in the design process. In a number of design problems, the design space may be multimodal, thereby introducing the need for a global search strategy that offers an increased probability of locating the global optimum in addition to relative local optima. The latter may contribute to demands on computational resource requirements, and the need exists for further development of function approximation methods that alleviate these requirements.

The design space for the multidisciplinary rotor blade design problem is generally nonconvex, is characterized by a mix of continuous, discrete, and integer design variables, and has an underlying analysis that is inherently nonlinear and computationally demanding.[3] A number of previous studies have indicated that the use of traditional mathematical programming-based optimization techniques may result in a suboptimal design. Furthermore, the ability to include discrete variables in these traditional methods requires the use of specialized techniques such as the branch-and-bound,[4] which is only effective for moderate-sized problems. Genetic algorithms (GAs) are stochastic search techniques that use random sampling over the entire design domain to conduct a highly exploitative search. The distributed nature of this search makes it suitable for search in a nonconvex or disjoint design space[5]; furthermore, it is naturally amenable to handling discrete and integer variables in the search process. GAs have been applied to the design of rotorcraft blades for reduced vibration,[6] where specialized strategies have been proposed for extending the use of GAs[7] in large dimensionality problems. The present work seeks to extend their applicability to even larger problems through the use of problem decomposition.

Decomposition methods[8,9] have emerged as an efficient solution strategy to large-scale design problems. Here the optimal solution to the design problem is obtained as a number of coordinated solutions of smaller subproblems; solution coordination is necessary to account for any interactions among the decomposed subproblems. In this approach, the number of design variables in each subproblem can be kept small, and, furthermore, decompositions along the lines of disciplines may be possible in some situations. Given the multimodal nature of the design space for the rotor blade design problem, GAs represent a logical choice for a solution strategy, and their adaptation in the decomposition-based approach is the subject of this article. In using GAs in a decomposition-based design environment, the approach would be to assign subsets of design variables to different subproblems; additionally, constraints most critically affected by the variables of a particular subproblem would also be assigned for satisfaction within that subproblem. These smaller-sized subproblems can be handled by the genetic algorithm without any specialized treatment, if the interactions between the temporarily decoupled systems are appropriately considered. The challenge in the approach resides in developing a rational procedure for determining how the problem must be partitioned (a topology of decomposition), and in a procedure that naturally accounts for the interactions among the decomposed subproblems. This article outlines an approach whereby causality relations developed through the use of neural networks[10] are used to facilitate the task of problem decomposition. Once the subproblems are generated, GA-based searches are conducted in parallel in each of

Received Nov. 28, 1995; revision received June 2, 1996; accepted for publication June 23, 1996.

*Research Associate, Mechanical Engineering, Aeronautical Engineering and Mechanics Department. Member AIAA.

†Professor, Mechanical Engineering, Aeronautical Engineering and Mechanics Department. Associate Fellow AIAA.

the subproblems. This article describes strategies by which changes in a subproblem are communicated to other subproblems through interpopulation migration of designs. Note that such an approach allows for parallel processing of multiple populations, adding to the computational efficiency of the genetic search process.

As noted earlier, the computational costs involved in computing vibratory loads and power components for the rotor blade problem are significant, as these computations entail a nonlinear and time-dependent analysis. A neural-network-based function approximation approach was adopted wherein both the multilayer back-propagation (BP) network[11] and a variant of the counter-propagation (CP) network[12] were used. Subsequent sections of this article describe a typical multidisciplinary rotor blade design problem, encompassing various facets of problem complexity-coupling, presence of discrete/integer variables, and high cost of analysis. The approach for parallel GA implementation in a decomposition-based design is discussed and applied to the rotor blade design problem. A comparison of optimization results obtained from the proposed approach, and an all-in-one (nondecomposition-based) approach is also presented.

Multidisciplinary Design of a Helicopter Rotor Blade

The design of helicopters with acceptable levels of vibration has been an important consideration in the rotorcraft industry. Researchers and design engineers have sought solutions to this problem through designing the rotor system with adequate frequency separation from the harmonic of aerodynamic loads, tuning the fuselage to provide adequate frequency placement, and designing vibration isolation systems. Integral to the design of the rotor system is the tailoring of the individual blades. This is a complex, coupled, multidisciplinary problem, where improved blade performance can only be realized by a proper consideration of the interactions among many disciplines, including aerodynamics, acoustics, structures, and multibody dynamics.[13]

Formal optimization techniques have been explored extensively in the context of multidisciplinary design of rotorcraft systems.[14–17] Various passive methods for reducing rotorcraft vibrations through attenuation of the vibratory hub loads have been investigated at both analytical and experimental levels. These methods have considered a modification of the spanwise distribution of mass and stiffness, and the blade geometry to satisfy multidisciplinary design requirements. More recently, research has focused on the application of optimization methods to tailor the rotor blade properties so that structural response reduces the fixed system hub loads, and enhances aerodynamic performance indicated by the powers required in different flight regimes. The complexity of the design space, large problem dimensionality, and significant computational resource requirements imposed by the repetitive analysis environment of mathematical optimization, mandate the need for examining new computational paradigms for this class of problems. In particular, the focus should reside in the use of global search strategies coupled with global function approximation tools that would make this design problem more amenable to present-day and emerging computational hardware.

In the present study, the multidisciplinary design of a hingeless composite blade was selected to constitute the test bed for the proposed design strategy. A finite element-based multibody formulation for nonlinear dynamic analysis[18] was used to model the rotor blade. This formulation uses a multibody representation of flexible structures undergoing large displacements and finite rotations, and allows for a treatment of arbitrary topologies and all geometric and material nonlinearities. Elastic components are connected using various types of joint elements by which the rotor system can be represented. This approach involves more unknown coordinates than are strictly necessary to define the configuration of the system. Hence, equations for kinematic constraints linking the redundant coordinates are an integral part of the formulation. In this method, finite rotations are represented using components of the conformal rotation vector, and kinematic constraints among the various bodies are enforced via a Lagrange multiplier technique. For an airloads model, unsteady aerodynamics are included to obtain the induced flow and to calculate the aerodynamic forces and moments in hover and forward flight.

Multidisciplinary Design Requirements

The use of composites in the manufacture of rotorcraft blades has allowed for the possibilities to fabricate nonrectangular blades with variation in twist distribution and airfoil sections along the blade span. One specific objective of aerodynamic design is to reduce the horsepower required for different flight conditions, with a specified design gross weight operating at a specified altitude and temperature. Satisfactory aerodynamic performance is defined by three requirements[15]:

1) The horsepower required for flight conditions must not exceed the horsepower available.

2) An airfoil section stall along the rotor blade must be avoided for any forward flight operating condition, i.e., the rotor disk must retain a certain lift performance.

3) The vehicle must be in trim.

These aerodynamic design requirements can be summarized as follows:

$$HP_{\text{req}} \leq HP_a \tag{1}$$

$$(C_T/\sigma)^L \leq C_T/\sigma \leq (C_T/\sigma)^U \tag{2}$$

$$\theta(\psi) = \theta_0 + \theta_{1s} \sin \psi + \theta_{1c} \cos \psi \tag{3}$$

The rotor dynamic design considerations focus on the vibratory response of the blades that directly determines the extent of excitation of the fuselage by forces and moments transmitted through the hub. Design requirements include limits on blade frequencies, vertical and in-plane hub shears, rolling and pitching moments, and aeroelastic stability margins in hover and forward flight. The blade natural frequencies are required to be separated from multiples of the rotor speed, and, the transmitted vertical and in-plane hub shears and moments should be minimized.

Important factors in the structural design of rotorcraft blades are material strength for static structural loads and mass moment of inertia for autorotation.[19] Structural design requirements also include limits on blade deformations. Flapwise and in-plane bending deformations are usually satisfied because of the inherently high bending stiffness of composite blades. Additionally, material strength allowables also contribute to the definition of the feasible regions in the design space. Stresses in the blade structure for all load cases are required to be less than the uniaxial allowable values for the material. Furthermore, strength constraints that account for an interaction among the stress components must also be considered. In the present work, this was facilitated by the inclusion of a Tsai–Wu failure criterion for composite laminates. The autorotation requirement pertains to maintaining the mass moment of inertia of the rotor in the rotational plane at an acceptable level. An autorotation index (AI), given by Eq. (4), is required to be no less than 1.7 for a single rotor helicopter:

$$\text{AI} = \frac{I\Omega^2}{1100HP_{\text{HOGE}}} \tag{4}$$

A flight load that induces buckling is unlikely to occur because of the high tensile loads caused by centrifugal forces in a rotating system. However, buckling may be critical in ground handling or during low rotational speeds in a strong wind.

Design Model and Problem Statement

A simplified rotor blade design problem that involves an integration of the previously described multidisciplinary requirements was considered in the present work. The objective of the design problem is to design the blade geometry and internal structure to minimize a weighted sum of the rotor-fixed-system hub shear force and bending moments for a hingeless rotor blade in forward flight; aerodynamic, performance and structural design requirements are considered as constraints, and dynamic requirements (vibratory loads) constitute a multicriterion objective function. The design variables used in the rotor blade design problem are shown in Fig. 1. The blade is divided into 10 segments along the spanwise direction. As shown in Fig. 1b, the blade is modeled by a thin-walled graphite/epoxy composite box–beam. The flange sections of this beam are symmetric, balanced laminates with fiber orientation of ±45 deg and thickness t_1; note that t_1 is a discrete variable obtained by selecting an integer number of plies of thickness 0.1 mm. The vertical webs of the box–beam are also composite laminates. The layup of these laminates is of the type $[\pm 45/\pm\theta_1/\pm\theta_2]_T$, where the outer half of each vertical web is of orientation ±45 deg; the thicknesses of these vertical webs are t_2 and t_3, respectively; both of these variables are also discrete for the same reason as that cited for the horizontal flange. In addition to the ply thicknesses and fiber orientations, the placement of five nonstructural tuning masses along the blade span was also considered as design variables for the problem. The aerodynamic shroud around the blade is assumed to be a NACA 0012 geometry. Finally, the geometry of the blade was defined by a blade twist distribution parameter, a chord ratio, and a spanwise position of blade taper inception. The rotor angular velocity was also considered as a design variable, resulting in a total of 42 design variables. The lower and upper bounds on these design variables are shown in Table 1. The last column in this table indicates the finite precision with which each variable is allowed to change in the design space; in essence, all variables are considered to be discrete in nature. Design constraints in the problem include power required in hover and forward flight, and are denoted as HP_h and HP_f, respectively, the figure-of-merit η, which reflects the power performance ratio in hover out-of-ground effect, AI, lift performance indicated by C_T/σ, blade weight W_b, local buckling stresses in the structural box sections σ_{buck}, and a failure measure for composite structures $\bar{R}$. This resulted in a total of 20 design constraints.

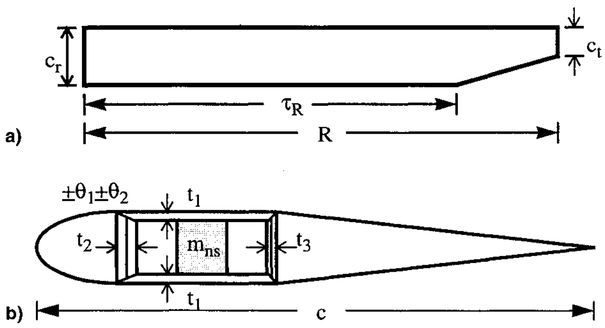

Fig. 1 a) Rotor blade planform geometry and b) cross section of blade airfoil.

Decomposition-Based Design Approach

A solution to the design problem formulated in the previous section may be obtained by linking the analysis of the blade to a traditional mathematical programming algorithm. In this approach, all variables and constraints would be considered simultaneously. Even if all of the design variables were allowed to vary in a continuous manner, the problem dimensionality, nonconvexities in the design space, and computational costs would severely limit the effectiveness of such an approach. As an example of the latter, the simplified blade analysis model described in the previous section requires about 3.3 CPU minutes on a SPARC 10 workstation, and a finite difference-based gradient computation at one design point would require about 141.9 CPU minutes on this machine. These requirements would clearly place severe demands on the available resources (both CPU time and wall-clock time). Two major requirements of a solution strategy for this problem are a global search strategy that minimizes the effects of nonconvexity in the design space, and a global function approximation strategy that would render the process computationally realizable. Genetic algorithms have been shown to overcome effects of nonconvexity; however, their use in high-dimensionality, computationally cumbersome design problems poses challenges that can be addressed through an innovative integration into a decomposition-based strategy. Subsequent sections of this article describe one such approach, and its application to the present design problem.

Genetic Algorithms in Decomposition-Based Design

Genetic algorithms are based on representing possible solutions to a given problem by a population of bit strings of finite lengths, and to use transformations analogous to the biological reproduction and evolution to improve and vary the coded solutions. A commonly used approach is to represent each design variable by a fixed-length binary string, and append the strings for each design variable head-to-tail to create a chromosome-like representation of the design. Several such strings are defined to constitute a population of designs, and this population is then subjected to transformations that result in an improvement of the objective function and in the satisfaction of the constraints.[7] In a large-scale design problem, the string representations of designs can get quite long, and the number of design alternatives in the search space that must be examined also increases dramatically. As an example, for the rotor blade design problem with 42 design variables represented with a relatively coarse precision, the binary bit string length is 179. This implies that the number of design alternatives represented by the string is 2^{179}, which is of order 10^{54}. It is easy to visualize dramatic increases in the number of design variables with an increasing number of variables and with an increased precision of representation. A logical solution to this problem would be to break the string-like repre-

Table 1 Range of design variables used in neural network training

Design variable	Symbol	X_i	Minimum	Maximum	Precision
Tuning mass, kg	m^i	X_1–X_5	0.0	4.0	0.1
Horizontal flange thickness ratio	t_1^i	X_6–X_{15}	0.25	0.35	0.01
Left vertical web thickness ratio	t_2^i	X_{16}–X_{25}	0.20	0.30	0.01
Right vertical web thickness ratio	t_3^i	X_{26}–X_{35}	0.20	0.30	0.01
Blade twist, deg	θ_t	X_{36}	4.0	8.0	0.1
Twist shape parameter	δ	X_{37}	0.3	1.0	0.1
Taper inception point	τ_R	X_{38}	0.50	0.80	0.01
Chord ratio	λ_c	X_{39}	1.0	2.0	0.1
Rotational speed, rad/s	Ω	X_{40}	24.0	30.0	1.0
Layup angle of vertical web, deg	θ_i	X_{41}, X_{42}	30	90	5

sentation of the design into a number of smaller segments, each containing a subset of the design variables (local variables). These smaller-length strings can then be evolved in a parallel manner, with each evolution working on the objective function for the problem and a subset of design constraints most critically affected by the local variables. This approach requires that a rational approach for partitioning the design problem and to account for any cross coupling between subproblems be available. Previous work has shown that neural networks trained on I/O data from the design domain can be used to extract causality,[12] and this approach was used here as a method for problem partitioning. Such trained networks can also be used to provide a global function approximation capability,[11] much like response surfaces. In addition to reducing computational costs, trained networks also provide an approach to account for coupling among temporarily decoupled subproblems.

Topology of Problem Decomposition

A BP neural network (see Fig. 2) may be used to develop a mapping between given input and output quantities, provided that a number of samples of I/O data are available to train the network. The training of the network requires an adjustment of network parameters, including the interconnection weights w_{ij}^{kl}, so that for the given training samples, the error between the network predicted and actual values of the output is minimized. The interconnection weights of such a trained network can be analyzed to determine the importance of any input component on an output quantity of interest. In the present study, the network represents a mapping between all design variables (inputs) and the objective and constraint values (outputs). In this situation, the weight analysis would identify the paths of influence between design variables and objective/constraint values. As shown in Ref. 20, the matrix of such dependencies, referred to as the transition matrix, can be developed by performing a matrix product of the interconnection weight matrices between different layers of the network as indicated in Eq. (5), and normalizing the elements of the resulting matrix as shown in Eq. (6):

$$[T] = \prod_{k=1}^{N-1} W^k \tag{5}$$

$$\bar{T}_{ij} = \frac{T_{ij}}{\max_j |T_{ij}|} \tag{6}$$

W^k is the kth weight matrix, the coefficients w_{ij}^{kl} of which represents the interconnection weight between the ith neuron of the kth layer and the jth neuron of the lth layer; N denotes the total number of layers of neurons in the network architecture. This normalized matrix $\bar{T}_{ij}$ indicates both the magnitude and sign of the influence between input and output quantities, and may be used for problem decomposition as described in the following section.

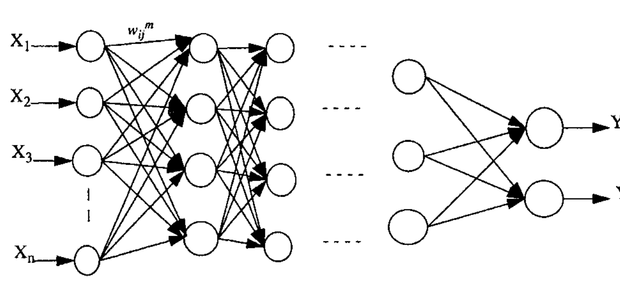

Fig. 2 BP network with h-hidden layers.

Optimal System Decomposition

To implement a decomposition-based design strategy in a large-scale system, the system must be partitioned into an appropriate number of subproblems, depending on available computing machines or parallel processors. If the transition matrix has components that clearly indicate the patterns of influence, the decomposition topology becomes obvious. However, in most problems there would be some arbitrariness as to which variable falls within a particular subproblem. A reasonable and logical approach for partitioning is one where balanced subsets of design variables would be assigned to different subproblems, and where each subproblem would be responsible for meeting the system level design objectives and for satisfying constraints most critically affected by the design variables of that subproblem. In mathematical terms, this translates into partitioning the transition matrix into K (where $2 \le K \le \text{NCON}$, and NCON is the total number of constraints in the design problem) different groups denoted as G_k; each group contains design variables x_i that have the strongest influence on constraints belonging to the group G_k. To formalize the partitioning procedure, another optimization procedure was formulated in terms of grouping identity variables V_{ij}, defined as follows:

$$\begin{aligned} V_{ij} &= 1 \quad \text{if} \quad x_i \in G_k \\ V_{ij} &= 0 \quad \text{if} \quad x_i \notin G_k \end{aligned} \tag{7}$$

The subscript j refers to the jth constraint. To obtain an optimal partitioning, the following problem can be solved:

Minimize

$$f = \frac{1}{PI} + \frac{1}{2} \sum_{i=1}^{K} \sum_{j=1}^{K} (1 - \delta_{ij}) |Nx_i - Nx_j| \tag{8}$$

Subject to

$$1 \le N(g_j)_{G_k} \le N(g_j)^U \tag{9}$$

where δ_{ij} is a Kronecker delta, Nx_k is the number of design variables assigned to group G_k, and $N(g_j)_{G_k}$ is the number of constraints in group G_k. The performance index PI is defined as

$$PI = \left(1 \Big/ \sum_{i=1}^{\text{NDV}} \sum_{j=1}^{\text{NCON}} |\bar{T}_{ij}| \right) \sum_{i=1}^{\text{NDV}} \sum_{j=1}^{\text{NCON}} (V_{ij} |\bar{T}_{ij}| / z_i) \tag{10}$$

where NDV is the number of design variables. Note that the product $V_{ij}|\bar{T}_{ij}|$ in Eq. (10) has a value of either zero (if the ith variable does not belong to the constraint group G_k) or the absolute value of the coefficient $\bar{T}_{ij}$ of the transition matrix. In the calculation of the performance index, this product is divided by the quantity z_i, which is simply the number of constraints affected by the ith design variable. The objective function of Eq. (8) has two components; the first term leads to a maximization of PI, whereas the second term ensures a minimal difference between the number of design variables in each group. The constraint in Eq. (9) is necessary to limit the number of design constraints allocated to a specific group to a number $N(g_j)^U$. This is an integer programming optimization problem with variables V_{ij}, and which can be conveniently solved using the GA approach.

Coordination Strategies in Parallel GA Implementation

Once the problem is decomposed into subproblems, each subproblem may be solved in parallel provided that there is some mechanism to coordinate the solutions in the different subproblems. Such coordination is necessary as the decoupling is seldom complete. Consider the design problem of minimizing or maximizing a function $F(X)$ in terms of a design variable vector X. Also, let the design constraints $g_j(X)$ belong to the global constraint set G. The vector X and constraint set G are said to define a system level problem. Using an optimal partitioning scheme allows for the creation of K subproblems, the objective function for each of which remains as the system level objective function. The Pth subproblem may be represented mathematically as follows:

$$\text{Min or Max} \quad F(X_P)$$
$$\text{Subject to} \quad g_P(X_p) \leq 0 \qquad (11)$$
$$X_1, X_2 \ldots, X_{P-1}, X_{P+1}, \ldots X_K \text{ are const}$$

Here, X_P and g_P are the design variable vector and constraints allocated to the Pth subproblem, respectively. The GA optimization strategy can be implemented for each of the subproblems; shorter string lengths, and hence, smaller population sizes are required in each subproblem. The genetic evolution process can be carried out in parallel. As stated earlier, the principal difficulty in this approach is that the constraint sets identified for a particular subproblem are not completely independent of the design variables that may have been assigned to another subproblem. Such coupling must be accommodated in the parallel optimization scheme, and the use of CP networks provides one approach to account for subproblem interactions.

A discussion of the CP network is beyond the scope of this article. Details on the network architecture and its functioning are available.[12] Like the BP network, this network also provides a function mapping capability between some input and output quantities. An important difference with the BP network, however, is an important property of this network referred to as a pattern completion capability; if an incomplete input pattern is presented to the network, the network estimates the most likely makeup of the missing components.

In the present work, the GA-based optimizer in each subproblem was linked to a trained CP network. The inputs to the CP network in each subproblem were the design variables for that subproblem and approximations of variables in other subproblems. While approximations of variables from other subproblems can be excluded, numerical testing demonstrated[21] that the quality of function approximations was significantly enhanced with the inclusion of such approximations. An effective strategy of providing these approximations is to migrate the best designs from each subproblem to all other subproblems after a prescribed number of cycles of GA search. A schematic of this setup is shown in Fig. 3, and a stepwise description of the numerical process is as follows.

1) Develop a trained CP network to map the relation; $X \rightarrow \{G, F\}$, where F is the system level objective function.

2) Develop a trained BP network to map the relation; $X \rightarrow \{G\}$. Analyze the weights of this trained network to establish a topology for decomposition.

3) For each subset of design variables in a subproblem, initialize a starting population of designs. Denote design variables of other subproblems as problem parameters for the sub-problem under consideration.

4) Evolve each subproblem in parallel for a fixed number of generations. Function analyses in each subproblem are obtained by presenting subproblem design variables and problem parameters to trained CP network of step 1.

5) Conduct interpopulation migration of problem parameters. Two strategies of this problem coordination were implemented: S1, for each subproblem, use as problem parameters the current best design variable values of other subproblems and S2, for each subproblem, evaluate all possible combinations of problem parameters. These would include current problem parameters, and those available as the new best designs in other subproblems. Select a combination so that the current objective function either improves or, at worst, stays the same.

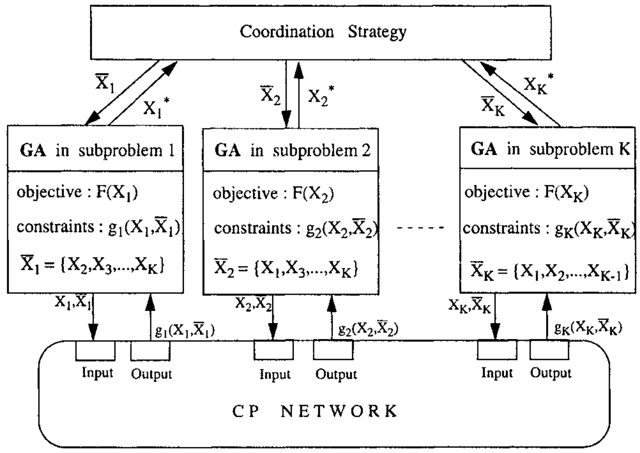

Fig. 3 Parallel design process in GA-based decomposition approach.

6) Repeat step 4 until no further improvement in the objective function value is obtained or the allowed number of function evaluations has been performed.

Results and Discussion

A number of numerical experiments were conducted to determine the validity of the proposed approach. At the very outset, the BP and CP networks were trained to develop causal relations and to generate function approximations. With a total of 1450 training patterns, the BP network yielded maximum errors of 4% when presented with 100 input patterns that were not part of the training process. Average errors in this case were only about 3%. The CP network required a larger number of training patterns. For 6800 training patterns, the average error for 240 testing patterns not part of the training set was about 9.27%. When the number of training patterns was increased to 12,800, the average error over all of the testing patterns was reduced to 5.23%. In the latter case, the maximum error over all of the testing patterns was about 7%. More detailed discussions on the use of neural networks to model rotor blade responses are available.[21]

In relation to the size of the training set, two observations must be made.

1) Although the CP network requires a large initial investment of computational resource, the network can be used to run many simulations of the GA approach, each with different random number seeds, and in a computationally efficient manner. It is important to note that each GA simulation itself may require function evaluations well in excess of the function evaluations necessary for establishing the training set. Hence, GA search would not be a practical alternative in the absence of the neural network-based approximation scheme.

2) The CP network used in this application was adopted primarily for the pattern completion capabilities of the network. As explained earlier, such capabilities provide a better mechanism for coordinating the problem parameter updates in each subproblem. It is entirely possible to use a BP network, trained to much higher predictive accuracy with fewer training patterns, as the function approximation tool. As described later in this section, such an approach was pursued in this work and produced some degradation in the final result.

An analysis of the interconnection weights of the trained BP network was performed to determine the topology for problem decomposition. Problem decomposition by constraints requires a transition matrix generated from the weights of a neural network, trained to develop the mapping between design variables as inputs and the corresponding constraint values as output. A total of only seven constraints was considered in the optimal problem partitioning, as constraints for local buckling were arbitrarily assigned to the same set of design variables that

Table 2 Topology of system decomposition

Problem definition	Subproblem 1	Subproblem 2	Subproblem 3
Objective		$F(X) = c_1F_z + c_2M_y + c_3M_z$	
Constraints	$HP_h \leq HP_a$	$HP_f \leq HP_a$	$W_b \leq W_b^U$
	$\eta^L \leq \eta \leq \eta^U$	——	$\bar{R} \leq 1$
	$AI \geq AI^L$	$C_T/\sigma^L \leq C_T/\sigma \leq C_T/\sigma^U$	$\sigma_{buck} \leq \sigma_{all}$
Design variables	$m^1, m^2, m^3, t_1^4, t_1^5, t_1^6, t_2^3, t_2^6, t_2^9, t_3^6, t_3^8, t_3^9, \tau, \pm\theta_1$	$t_1^3, t_1^{10}, t_2^4, t_2^5, t_2^7, t_2^8, t_3^2, t_3^3, t_3^5, t_3^7, t_3^{10}, \theta_t, \delta, \pm\theta_2$	$m^4, m^5, t_1^1, t_1^2, t_1^7, t_1^8, t_1^9, t_2^1, t_2^2, t_2^{10}, t_3^1, t_3^4, \lambda_c, \Omega$

	HP_h	F M	A I	HP_f	C_T/σ	W_b	$\bar{R}$
x 1	-0.31122	-0.20702	-0.10739	0.01432	-0.02469	-0.31651	-0.04328
x 2	-0.31767	-0.22166	-0.09736	0.02230	0.00049	-0.32416	0.15529
x 3	-0.33205	-0.27008	-0.05503	0.03852	0.06996	-0.34035	0.24476
x 9	-0.06477	-0.02082	-0.03332	-0.02082	-0.01283	-0.06346	0.02599
x10	-0.08221	-0.04840	-0.02593	-0.00124	0.05655	-0.08428	0.02478
x11	-0.10063	-0.06934	-0.01222	0.01204	-0.00474	-0.10248	0.11756
x18	0.01140	0.01592	-0.00378	-0.02108	-0.06511	0.01001	-0.03155
x21	-0.01210	-0.03661	-0.00413	-0.00026	0.02378	-0.01415	0.03511
x24	-0.00805	-0.00719	0.00293	0.01389	-0.06470	-0.00944	-0.08279
x31	0.02001	0.04601	-0.00441	-0.00455	0.00017	0.01652	0.03500
x33	-0.02104	-0.01701	0.00217	0.01078	0.01453	-0.02130	0.04315
x34	-0.01504	-0.03227	-0.00062	-0.01021	0.01551	-0.01440	-0.01239
x38	-0.20777	-0.17080	1.00000	0.15572	0.10275	-0.22482	0.37426
x41	0.03738	0.03629	-0.00408	-0.02273	-0.03160	0.00321	0.06405
x 8	-0.07918	-0.00643	-0.03324	-0.02226	-0.03423	-0.07776	0.01929
x15	-0.05017	-0.04212	0.01513	-0.01349	-0.09035	-0.04858	0.03682
x19	0.00161	-0.00197	-0.00065	0.00359	-0.02500	-0.00144	-0.06336
x20	-0.01344	-0.01270	-0.00095	-0.01467	-0.04590	-0.01014	0.00908
x22	0.01889	0.03187	0.00276	-0.01515	-0.04468	0.01751	0.01327
x23	0.00462	0.02569	-0.00061	-0.02368	-0.10400	0.00149	-0.01317
x27	0.00739	0.02657	-0.00967	-0.00698	-0.09053	0.00595	-0.13789
x28	0.02278	0.02255	0.00411	0.03776	-0.08852	0.02164	-0.04513
x30	-0.02402	-0.05278	-0.00852	0.02067	-0.04556	-0.02560	0.08914
x32	0.00829	0.01154	0.00103	-0.00334	-0.06217	0.00558	-0.05512
x35	-0.02667	-0.05944	-0.00075	0.00296	0.00125	-0.03032	-0.03318
x36	0.03934	0.03059	-0.00346	1.00000	1.00000	0.00344	0.14198
x37	0.01964	0.01195	-0.00840	-0.44560	-0.45198	0.00162	-0.02298
x42	0.01018	-0.00509	0.00175	0.04040	-0.07911	0.00112	-0.06664
x 4	-0.29512	-0.17174	0.00144	0.03140	0.07893	-0.30238	0.22224
x 5	-0.30075	-0.17085	0.12103	-0.01062	-0.04009	-0.30571	0.40037
x 6	-0.05802	-0.02470	-0.02052	0.01774	-0.01167	-0.05586	-0.30023
x 7	-0.13262	-0.08277	-0.03034	0.05067	0.04350	-0.12982	-0.37191
x12	-0.13337	-0.13914	-0.00130	-0.00175	0.02686	-0.13284	0.13375
x13	-0.09454	-0.04700	0.01197	0.00952	0.13411	-0.09813	0.22234
x14	-0.10299	-0.05171	0.02424	-0.01474	-0.01785	-0.10358	0.14926
x16	-0.02061	-0.02528	-0.00536	0.00044	0.07328	-0.01802	0.00577
x17	-0.05248	-0.06647	-0.00443	0.01262	-0.06872	-0.00707	-0.09438
x25	-0.00733	-0.02811	0.00606	0.01262	-0.06872	-0.00707	-0.09438
x26	-0.05582	-0.04970	-0.00349	0.01341	0.10149	-0.05465	0.02636
x29	-0.00498	-0.02420	0.00076	0.00700	-0.02614	-0.00261	-0.05519
x39	1.00000	0.21680	-0.14504	-0.33351	-0.21798	1.00000	-0.82832
x40	0.01542	1.00000	0.40144	0.77140	-0.07304	0.00535	1.00000

Fig. 4 Optimal system partitioning.

were driving the structural weight constraints. On the basis of this transition matrix, the problem decomposition was obtained using the optimal partitioning scheme described in Eqs. (8–10). By specifying the number of subproblems $K = 3$, the optimal partitioning of the transition matrix is as shown in Fig. 4. Note that in any row of this matrix, the magnitude of coefficients inside each solid box defining a subproblem are, in general, higher than those outside of this box. The second term in Eq. (8) also has a strong influence on the partitioning, resulting in each group being assigned the same number of design variables. Interestingly, the optimization results in a grouping of constraints that have similar properties; HP_h, η, and AI are related to the hover condition, and HP_f and C_T/σ are calculated for forward flight conditions. Table 2 shows the

Table 3 GA control parameters used in decomposition approach

GA parameters	Subproblem 1	2	3	All-in-one approach
$P_{\text{crossover}}$	0.80	0.80	0.80	0.80
P_{mutation}	0.02	0.02	0.02	0.01
NPOP	100	100	100	300
NSTRING	63	57	59	179

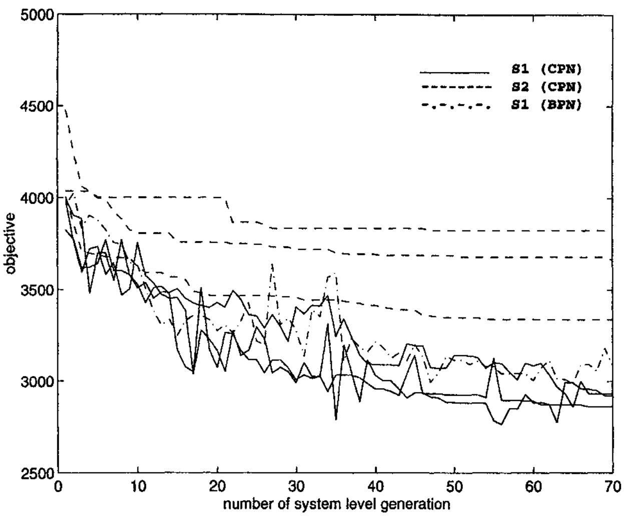

Fig. 5 Convergence history of design strategies in decomposition approach.

suggested decomposition topology, where the system level X and G are broken down into subsets with respect to design constraints. In Table 2, m^i is the tuning mass; t_j^i is the jth flange of the thin-walled beam in the ith segment; θ_t and δ are the blade twist angle at the tip and the shape parameter defining the nonlinear distribution of blade twist, respectively; λ_c is a chord ratio; τ_R is a nondimensionalized value of taper inception point; Ω is a rotational speed; and $\pm\theta_i$ is a composite ply angle in the box–beam.

The decomposition-based design was implemented for a number of test cases, using the parallel GA approach described in previous sections. Recognizing the random nature of the GA search, the search process was repeated for a number of different settings for the pseudorandom number generator. GA parameters such as probabilities of crossover and mutation, population size, and string lengths for the design, are summarized in Table 3.

The convergence histories of the system level objective function for two different strategies of the coordination are shown in Fig. 5. In strategy S1, the solution exhibits significant oscillations resulting from the introduction of infeasibilities with each update of problem parameters. However, the degree of oscillation tends to decrease as the solution converges. Numerical experiments with different random numbers demonstrate that this approach yields similar best objective function values after a few executions of the search process. The results for strategy S2 show a monotonically decreasing value of the system level objective function (this was a requirement of the updating scheme); however, the best objective function value appeared to be heavily dependent upon the initialization of the random number generator. Strategy S1 consistently resulted in better values of the best objective function value than strategy S2. However, strategy S2 does guarantee that once a feasible design is identified, an abrupt termination of the search process will at least produce a feasible design. Also shown in this plot are the results of using the BP network as the function approximation tool in place of the CP network used in previous numerical experiments. The convergence trend in using the BP network is similar to that recorded earlier for the CP networks. However, the lack of pattern completion capability in using this network produces somewhat higher values of the objective function value for a comparable number of function evaluations. The results of the decomposition-based strategy were compared against those obtained from treating all design variables and constraints in a single group by an all-in-one approach. Figure 6 shows this comparison in terms of the best objective function value and the required CPU time. Here, six different sets of results are presented, including four different implementations of the all-in-one approach and two sets of results corresponding to decomposition-based strategies S1 and S2. The all-in-one strategies include a straightforward GA implementation, or the plain GA, and three different advanced GA strategies.

Figure 6a shows the results for three different initializations of the random number generator. The best objective function values show the expected improvement in performance as advanced GA strategies are used. The decomposition-based de-

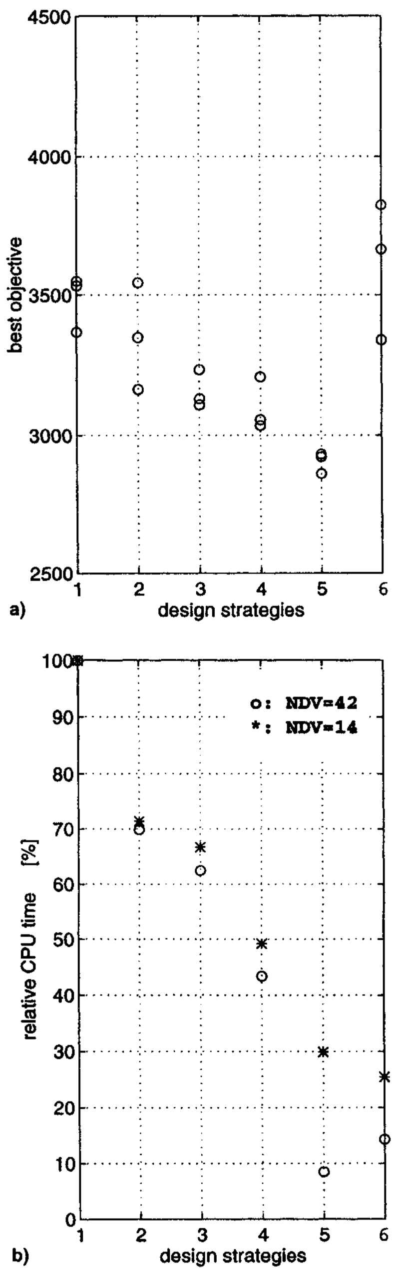

Fig. 6 Performance of large-scale design process; (1) plain GA, (2–4) advanced GA strategies,[7] (5) strategy S1, (6) strategy S2: a) best objective function and b) computational efficiency.

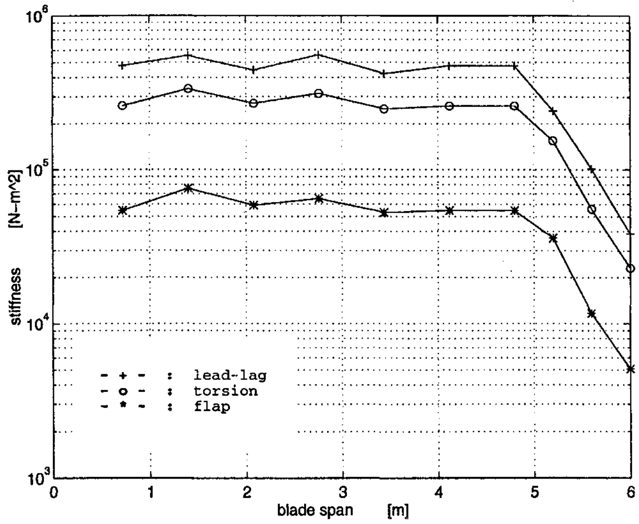

Fig. 7 Optimal stiffness distribution along blade span.

sign strategy S1 gives the best overall result. Strategy S2 yields results that were similar to the plain GA implementation. Clearly, this updating scheme severely limits the exploration of the design space by requiring that problem parameter updates must maintain a monotonic convergence trend in the objective function value. In Fig. 6b, the relative CPU time required to reach the best objective function value obtained from the plain GA is presented for the different design strategies. This figure clearly demonstrates that the decomposition approach is much more efficient from a computational resource standpoint, because of the parallel implementation in optimization. This figure also reinforces the fact that returns from a decomposition-based approach are much more significant as the size of the problem increases; the results for the 42-design variable problem are compared against similar results obtained for a 14-design variable problem defined in Ref. 22. For the best design obtained in the decomposition-based approach, the optimal stiffness distribution of rotor blade box–beam is shown in Fig. 7.

Concluding Remarks

This article describes an approach for adapting genetic algorithms in the decomposition-based design of large-scale multidisciplinary systems. A primary focus of the research was in the development and implementation of a rational approach by which the multidisciplinary design problem could be partitioned into a number of balanced subproblems. This task of partitioning was formulated and solved as an optimization problem. This method used quantitative information about the dependence of system response to design variables extracted from trained neural networks in carrying out the optimal problem partitioning. Once the problem was decomposed, the GA-based search was implemented in parallel in each of the subproblems. Strategies to account for the interactions among these decomposed subproblems was the other focus of the present study. The proposed methods were implemented in a multidisciplinary test problem, the design of a helicopter rotor blade to minimize vibrations at the rotor hub. Numerical results show the effectiveness of the decomposition-based approach over traditional all-in-one strategies, as indicated by both better performance and lower computational resource requirements.

Acknowledgment

Support for this work received under Army Research Office Contract DAAH-04-93-G-0003 to the Rensselaer Rotorcraft Technology Center is gratefully acknowledged.

References

[1]Sobieszczanski-Sobieski, J., "Multidisciplinary Design Optimization: An Emerging New Engineering Discipline," World Congress on Optimal Design of Structural Systems, Rio de Janeiro, Brazil, Aug. 1993.

[2]Tolson, R. H., and Sobieszczanski-Sobieski, J., "Multidisciplinary Analysis and Synthesis: Needs and Opportunities," *Proceedings of the AIAA/ASME/ASCE/AHS 26th Structures, Structural Dynamics, and Materials Conference* (Orlando, FL), AIAA, New York, 1985, pp. 1–12.

[3]Lee, J., "Genetic Algorithms in Multidisciplinary Design of Low Vibration Rotors," Ph.D. Dissertation in Mechanical Engineering, Rensselaer Polytechnic Inst., Troy, NY, 1996.

[4]Land, A. H., and Doig, A. G., "An Automatic Method for Solving Discrete Programming Problems," *Econometrica*, Vol. 28, 1960, pp. 497–520.

[5]Hajela, P., "Genetic Search—An Approach to the Nonconvex Optimization Problem," *AIAA Journal*, Vol. 26, No. 7, 1990, pp. 1205–1210.

[6]Hajela, P., and Lee, J., "Genetic Algorithm Based Sizing of Low Vibration Rotors," *Proceedings of the 5th ARO International Workshop on Dynamics and Aeroelastic Stability Modeling of Rotorcraft Systems*, Rensselaer Rotorcraft Technology Center, Troy, NY, 1993.

[7]Lin, C.-Y., and Hajela, P., "Genetic Search Strategies in Large Scale Optimization," *Proceedings of the AIAA 34th Structures, Structural Dynamics, and Materials Conference* (La Jolla, CA), AIAA, Washington, DC, 1993, pp. 2437–2447.

[8]Sobieszczanski-Sobieski, J., "Sensitivity of Complex, Internally Coupled Systems," *AIAA Journal*, Vol. 28, No. 1, 1990, pp. 153–160.

[9]Bloebaum, C. L., and Hajela, P., "Heuristic Decomposition for Non-Hierarchic Systems," *Proceedings of the AIAA 32nd Structures, Structural Dynamics, and Materials Conference* (Baltimore, MD), AIAA, Washington, DC, 1991, pp. 344–353.

[10]Hajela, P., and Szewczyk, Z., "On the Use of ANN Interconnection Weights in Optimal Structural Design," *Proceedings of the AIAA/NASA/Air Force 4th Symposium on Multidisciplinary Analysis and Optimization* (Cleveland, OH), AIAA, Washington, DC, 1992, pp. 924–931.

[11]Hajela, P., and Berke, L., "Neurobiological Computational Models in Structural Analysis and Design," *Computers and Structures*, Vol. 41, No. 4, 1991, pp. 657–667.

[12]Szewczyk, Z., and Hajela, P., "Neurocomputing Strategies in Decomposition Based Structural Design," *Proceedings of the AIAA 34th Structures, Structural Dynamics, and Materials Conference* (La Jolla, CA), AIAA, Washington, DC, 1993.

[13]Adelman, H., and Mantay, W. A. (eds.), "Integrated Multidisciplinary Optimization of Rotorcraft: A Plan for Development," NASA TM-101617, May 1989.

[14]Young, D. K., and Tarzanin, F. J., Jr., "Structural Optimization and Mach Scale Test Validation of a Low Vibration Rotor," *Journal of the American Helicopter Society*, Vol. 38, No. 3, 1993, pp. 83–92.

[15]Walsh, J. L., Bingham, G. L., and Riley, M. F., "Optimization Methods Applied to the Aerodynamic Design of Helicopter Rotor Blades," *Journal of the American Helicopter Society*, Vol. 32, No. 4, 1987, pp. 39–44.

[16]Friedmann, P. P., and Celi, R., "Structural Optimization with Aeroelastic Constraints of Rotor Blades with Straight and Swept Tips," *AIAA Journal*, Vol. 28, No. 5, 1990, pp. 928–936.

[17]Friedmann, P. P., "Helicopter Vibration Reduction Using Structural Optimization with Aeroelastic/Multidisciplinary Constraints," *Journal of Aircraft*, Vol. 28, No. 1, 1991, pp. 8–21.

[18]Bauchau, O. A., and Kang, N. K., "A Multibody Formulation for Helicopter Structural Dynamic Analysis," *Journal of the American Helicopter Society*, Vol. 38, No. 2, 1993, pp. 3–14.

[19]Nixon, M. W., "Preliminary Structural Design of Composite Main Rotor Blades for Minimum Weight," NASA TP-2730, July 1987.

[20]Goel, S., and Hajela, P., "Identification of Parameter Couplings in Turbine Design Using Neural Networks," *Proceedings of the AIAA/NASA/Air Force 5th Symposium on Multidisciplinary Analysis and Optimization* (Panama City, FL), AIAA, Washington, DC, 1994, pp. 590–597.

[21]Hajela, P., and Lee, J., "Role of Emergent Computing Techniques in Multidisciplinary Rotor Blade Design," *Emergent Computing Methods in Engineering Design; Applications of Genetic Algorithms and Neural Networks*, edited by P. E. Grierson and P. Hajela, Vol. 149, Springer–Verlag, Berlin, 1996, pp. 162–187.

[22]Hajela, P., and Lee, J., "Genetic Algorithms in Multidisciplinary Rotor Blade Design," *Proceedings of the AIAA 36th Structures, Structural Dynamics, and Materials Conference* (New Orleans, LA), AIAA, Washington, DC, 1995, pp. 2187–2197.