Design of Composite Laminates by a Genetic Algorithm with Memory

N. Kogiso[1] , L. T. Watson[2] , Z. Gürdal[3] , R. T. Haftka[4] and S. Nagendra[5]
Virginia Polytechnic Institute and State University
Blacksburg, VA 24061, USA

ABSTRACT

This paper describes the use of a *genetic algorithm with memory* for the design of minimum thickness composite laminates subject to strength, buckling and ply contiguity constraints. A *binary tree* is used to efficiently store and retrieve information about past designs. This information is used to construct a set of linear approximations to the buckling load in the neighborhood of each member of the population of designs. The approximations are then used to seek nearby improved designs in a procedure called *local improvement*. The paper demonstrates that this procedure substantially reduces the number of analyses required for the genetic search. The paper also demonstrates that the use of genetic algorithms helps find several alternate designs with similar performance, thus giving the designer a choice of alternatives.

INTRODUCTION

The design of composite laminates is often formulated as a continuous optimization problem with ply thicknesses and ply orientation angles used as design variables (e.g., Schmit and Farshi [1]). However, for many practical problems, ply thicknesses are fixed and ply orientation angles are limited to a small set of angles such as 0˚, 90˚, and ±45˚. Designing the laminate then becomes a stacking sequence optimization which can be formulated as an integer programming problem.

The laminate stacking sequence design problem with frequency constraints has been formulated by Mesquita and Kamat [2] with the numbers of plies as the design variables, leading to a nonlinear integer programming problem which was solved by using the branch and bound algorithm. Haftka and Walsh [3] showed that the use of ply identity design variables linearizes the integer programming formulation of the stacking sequence buckling maximization design problem.

---

[1] Visiting Student, Dept. of Aerospace Engineering.
[2] Professor, Depts. of Computer Science and Mathematics.
[3] Associate Professor, Dept. of Engineering Science & Mechanics.
[4] Christopher Kraft Professor of Aerospace and Ocean Engineering.
[5] Research Associate, Dept. of Engineering Science & Mechanics.
[6] A version of the paper was published as part of the proceedings of the AIAA/ASME/ASCE/AHS/ASC 35'th Structures, Structural Dynamics, and Materials Conference, April 18-20, 1994, Hilton Head, S. C.

However, when strength constraints are included, the problem becomes nonlinear again. The buckling maximization of laminates with strain constraints has been solved by Nagendra et al. [4] using a sequential linear integer programming technique based on the branch and bound algorithm.

More recently, Le Riche and Haftka [5] solved the laminate stacking sequence design problem by genetic algorithms (GA) [6, 7] which can handle nonlinear integer problems without the need for linearization. Nagendra et al. [8] applied the same approach to the design of stiffened composite panels. In both works it was found that the genetic algorithm can find a variety of alternative designs with similar performance, thus giving the designer a choice of alternatives. However, the cost of the genetic search is high requiring thousands of analyses.

Kogiso et al. [9] considered the use of information about past designs (memory) for the purpose of reducing the number of analyses required by the genetic algorithm. The information was stored in a binary tree format and was used to construct a set of linear approximations to the buckling load in the neighbourhood of each member of the population of designs. The approximations were then used to seek nearby improved designs in a procedure called local improvement. The procedure was applied to the problem of buckling load maximization for given laminate thickness. Substantial reductions in the number of required analyses were achieved by this use of memory of past designs.

The present work applies the same strategies to the dual problem of the minimum thickness design of laminates subject to buckling and strength constraints. Because the number of plies in the laminate varies, coding the designs as genetic strings is more complex. Also, the objective function is more complex, involving the laminate thickness, constraints margins and constraint violations, dealt with by penalty parameters. The paper investigates the effect of the variable string length and the use of penalty parameters on the performance of the algorithm.

ANALYSIS

The simply supported plate, shown in Figure 1, has longitudinal and lateral dimensions of *a* and *b*, respectively, and is loaded in the x and y directions by λN_x, λN_y, respectively, where λ is a load amplitude parameter. The laminate is assumed to be balanced, symmetric, and made up of 0, 90, and ±45 plies of thickness *t* each. The maximum number of plies that may be encountered during the optimization process is denoted by N. To reduce the number of design variables and enforce the balanced condition, the laminate is constrained to be made up of stacks of two 0 degree plies, two 90 degree plies, or a +45 and -45 pair of plies. These stacks are denoted by 0_2, 90_2, and ±45.

Taking into account the symmetry, at most N/4 ply orientations are required to define the entire laminate.

For a simply supported plate under biaxial compression loading, the plate buckles when the load amplitude parameter λ reaches the critical value λ_{cb} given by

$$\frac{\lambda_{cb}(m,n)}{\pi^2} = \frac{D_{11}\left(\frac{m}{a}\right)^4 + 2\left(D_{12} + 2D_{66}\right)\left(\frac{m}{a}\right)^2\left(\frac{n}{b}\right)^2 + D_{22}\left(\frac{n}{b}\right)^4}{\left(\frac{m}{a}\right)^2 N_x + \left(\frac{n}{b}\right)^2 N_y} \quad , \tag{1}$$

where m and n are the number of half waves in the x and y directions, respectively, that minimize λ_{cb}. The $D_{ij}s$ are the flexural stiffnesses, which depend on the lamination sequence. For a laminate made out of a single fibrous material, the flexural stiffnesses can be expressed in terms of only two lamination sequence parameters and material constants (e.g., Miki and Sugiyama [10]),

$$\begin{aligned}
D_{11} &= \frac{h^3}{12}\left(U_1 + U_2 W_1^* + U_3 W_2^*\right) \quad , \\
D_{22} &= \frac{h^3}{12}\left(U_1 - U_2 W_1^* + U_3 W_2^*\right) \quad , \\
D_{12} &= \frac{h^3}{12}\left(U_4 - U_3 W_2^*\right) \quad , \\
D_{66} &= \frac{h^3}{12}\left(U_5 - U_3 W_2^*\right) \quad ,
\end{aligned} \tag{2}$$

where U_i (i=1,...,5) are the material constants, h is the total plate thickness, and $W_1{}^*$ and $W_2{}^*$ are the bending lamination parameters

$$\begin{aligned}
W_1^* &= \frac{12}{h^3}\int_{-h/2}^{h/2} z^2 \cos 2\theta \, dz = \int_{-1/2}^{1/2} \xi^2 \cos 2\theta \, d\xi \quad , \\
W_2^* &= \frac{12}{h^3}\int_{-h/2}^{h/2} z^2 \cos 4\theta \, dz = \int_{-1/2}^{1/2} \xi^2 \cos 4\theta \, d\xi \quad ,
\end{aligned} \tag{3}$$

defined in terms of the ply orientation angle θ. The flexural stiffnesses D_{16} and D_{26} are assumed to be negligible. Here, $\xi = z/h$ is introduced to normalize the thickness, h, of the laminate. Equation (3) shows that, once normalized, the bending lamination parameters $W_1{}^*$ and $W_2{}^*$ are independent of the laminate thickness.

It is clear from (1) - (3) that the buckling load divided by the cube of the thickness of the plate, λ_{cb}/h^3, depends only on the bending lamination parameters, which are functions of the stacking sequence. Here, λ_{cb}/h^3 is called the *normalized buckling load* and is used for local improvement discussed later.

The strain failure constraint requires that all strains remain below their allowable limits. In our case γ_{xy} is zero, and the laminate strains are related to the loads on the plate by the relations

$$\lambda N_x = A_{11}\varepsilon_x + A_{12}\varepsilon_y \ , \qquad \lambda N_y = A_{12}\varepsilon_x + A_{22}\varepsilon_y \ . \tag{4}$$

The strains in the ith layer are obtained from the laminate strains by the transformation

$$\begin{aligned} \varepsilon_{1_i} &= \cos^2\theta_i\varepsilon_x + \sin^2\theta_i\varepsilon_y \ , \\ \varepsilon_{2_i} &= \sin^2\theta_i\varepsilon_x + \cos^2\theta_i\varepsilon_y \ , \\ \gamma_{12_i} &= \sin 2\theta_i\left(\varepsilon_y - \varepsilon_x\right) \ , \end{aligned} \tag{5}$$

where the A_{ij}s are the in-plane stiffnesses, and θ_i is the ply orientation angle of the ith ply. These stiffnesses can also be expressed in terms of in-plane lamination parameters and the material constants as

$$\begin{aligned} A_{11} &= h\left(U_1 + U_2V_1^* + U_3V_2^*\right) \ , \\ A_{22} &= h\left(U_1 - U_2V_1^* + U_3V_2^*\right) \ , \\ A_{12} &= h\left(U_4 - U_3V_2^*\right) \ , \end{aligned} \tag{6}$$

where the in-plane lamination parameters are defined as

$$\begin{aligned} V_1^* &= \frac{1}{h}\int_{-h/2}^{h/2}\cos 2\theta \, dz = \int_{-1/2}^{1/2}\cos 2\theta \, d\xi \ , \\ V_2^* &= \frac{1}{h}\int_{-h/2}^{h/2}\cos 4\theta \, dz = \int_{-1/2}^{1/2}\cos 4\theta \, d\xi \ . \end{aligned} \tag{7}$$

The nondimensional variable ξ is introduced to demonstrate that the in-plane lamination parameters are also independent of the thickness, as are the bending lamination parameters. The strain failure load λ_{cs} is the largest load factor λ such that all the principal strains in every layer are less than or equal to the allowable strain values.

It is clear from (4) - (7) that the strain failure load divided by the thickness, λ_{cs}/h, is only related to the in-plane lamination parameters and is called the *normalized strain failure load*. The normalized strain failure load is used for the binary tree discussed later.

The critical load λ_{cr} is defined as the smaller of the buckling load and the strain failure load:

$$\lambda_{cr} = \min\left(\lambda_{cs}, \lambda_{cb}\right) \ . \tag{8}$$

OPTIMIZATION PROBLEM

Genetic Algorithms

For a genetic algorithm, each design must be coded as a finite string of digits (the genetic code). The present work uses the stack coding defined by Nagendra et al. [8] which uses two plies of a given orientation angle together. A 0_2 stack is assigned the digit 1, a ±45 stack the digit 2, and a 90_2 stack the digit 3. Because the total laminate thickness is not fixed but the length of the string used to represent laminates is fixed, we choose a string length corresponding to an upper bound on the thickness. Therefore, to represent laminates with thicknesses smaller than the upper bound a code of 0 is introduced to identify empty stacks. For example, using a string length of 10 (*N=40*), the 32-ply symmetric laminate $[\ast/\ast/90_2/\pm45_2/90_2/0_2/\pm45_2/0_2]_S$ is encoded as 1 2 2 1 3 2 2 3 0 0, where $\ast$ denotes an empty stack. The leftmost 1 corresponds to the 0_2 stack closest to the laminate plane of symmetry. The rightmost 3 describes the outermost 90_2 stack. The zeros on the right correspond to empty stacks.

During the optimization process (initialization of the population, crossover, mutation and permutation operations), zeros can be generated in the interior of a laminate design code. This means the laminate has empty stacks inside. This problem is resolved by packing the string to move all zeros toward the outside of the laminate after the genetic reproduction. For example, the string 1 2 0 2 3 2 0 3 0 2 is packed to 1 2 2 3 2 3 2 0 0 0 which represents $[\pm45/90_2/\pm45/90_2/\pm45_2/0_2]_S$ laminate with a total thickness of 28 plies.

Pseudocode for the genetic algorithm is shown in Figure 2. The genetic search begins with the random generation of a population of strings representing design alternatives. Each individual string is analyzed for buckling and strength and its characteristics are stored for future use. Each design is assigned a fitness value based on its objective function that determines its probability for selection as a parent. Parents exchange parts of their genes (strings) in a process called crossover to create offspring (new design strings). Additional genetic operators, namely mutation and permutation (see Le Riche and Haftka, [5]), are applied to the child designs which then replace the parent generation. Before the evaluation of each design, interior zero genes in the design code are packed to the right.

Here, the best design is always carried to the next generation, which is an "elitist plan" version of the genetic algorithm. The optimization process is repeated until some specified number of generations provide no improvement in the best design. Following the generation of a new population, a local improvement procedure is implemented. The local improvement procedure replaces each design generated by the genetic operators by the estimated best of the neighboring designs. The estimation procedure is based on the characteristics of neighboring previously encountered designs and is described in detail later in the subsection "Local Improvement".

Commonly, genetic algorithms choose parents for mating according to their fitness. Here the fitness was chosen to be the relative rank in terms of the value of the objective function in the population (with the lowest value ranked first). The fitness assigned to the *i*th best individual of a population of m designs is defined as

$$\text{Fitness} = \frac{m+1-i}{\sum_{k=1}^{m} k} = \frac{2(m+1-i)}{m(m+1)} \quad . \tag{9}$$

The value of the fitness is equal to the fraction of a simulated roulette wheel assigned to the design in the process of selecting parents. Thus the best design ($i=1$) covers an area of $2/(m+1)$ on a roulette wheel with total area of 1, or slightly less than twice the average portion.

Objective Function for Thickness Minimization

The objective of this paper is to find the minimum thickness composite plate design subject to the constraints that the plate should not buckle nor fail under a given applied load factor λ_{ap}. The thickness of the plate must be an integer multiple of the ply thickness. We use the total number of stacks N_S as a surrogate variable for the thickness, where N_S can take only integer values. Additionally, the maximum number of contiguous plies of the same orientation is limited to four plies to alleviate matrix cracking problems (Le Riche and Haftka [5]). The total number of plies in a given laminate is $N_p = 4\, N_S$.

The optimization problem can be formulated as

$$\begin{aligned}
&\text{minimize} \qquad N_p \\
&\quad \text{subject to} \\
&\qquad\qquad g_1 = \lambda_{ap} - \lambda_{cr} \le 0\,, \\
&\qquad\qquad g_2 = \text{maximum number of contiguous plies of the same orientation} \le 4\ .
\end{aligned} \tag{10}$$

The stacking sequence affects the performance of the laminated plate, even if the thickness of the plate is fixed. Consequently, it is often possible to obtain many minimum thickness designs that satisfy all the constraints. Though these optima satisfy all of the constraints and have the same thickness, they have different constraint margins. Ideally, the preferred design is the one which has the highest constraint margin of all minimum thickness designs. This is accomplished by reducing the objective function slightly based on the constraint margin.

In addition to the constraint margins, the strength and the buckling constraints must be factored into the objective function, because genetic algorithms cannot easily directly deal with constraints. The constrained optimization problem was, therefore, transformed into an unconstrained one by combining the number of plies and the constraint margins and violations into an objective function ϕ defined as

$$\phi = \begin{cases} p_c{}^n \left\{ N_p + \varepsilon\left(\lambda_{ap} - \lambda_{cr}\right)\right\} & \text{if } g_1 \le 0\ , \\ \dfrac{p_c{}^n N_p}{\left(\lambda_{cr}/\lambda_{ap}\right)^{p_\lambda}} & \text{if } g_1 > 0\ , \end{cases} \tag{11}$$

where p_c (here *10/9*) is a penalty parameter for the ply contiguity constraints, and n is the number of violations of the contiguous ply constraint . When the critical load constraint is satisfied ($\lambda_{cr} \ge \lambda_{ap}$), a small fraction, ε, of the constraint margin is subtracted from the number of plies. When the critical load constraint is violated ($\lambda_{cr} < \lambda_{ap}$), the objective function is penalized by dividing by the critical load ratio raised to the power $p_\lambda > 0$.

The contiguous ply penalty is computed as follows. For example, when a laminate has three contiguous 0_2 or 90_2 stacks, the objective function is penalized by (10/9) for each occurrence of three contiguous stacks. Four contiguous 0_2 or 90_2 stacks in the laminate are recognized as two successive violations of the contiguous ply constraint and penalized by $(10/9)^2$ for each occurrence of four stacks. Moreover, when the laminate has two successive 0_2 or 90_2 stacks adjacent to the mid plane, it is recognized to have four successive 0_2 or 90_2 stacks through the mid plane and penalized by $(10/9)^2$.

The values of the parameters p_λ and ε have to be tuned for best performance. The value of ε must be small enough so that a thicker design with a large margin will not be favored over a thinner feasible design. To estimate an upper limit for ε, we note that when the plate thickness is multiplied by a factor of $f > 1$, the buckling load is approximately multiplied by f^3 and the strength failure load by f. Assuming that at the optimum $\lambda_{cr} \approx \lambda_{ap}$, a conservative requirement is therefore

$$N_p \leq fN_p - \varepsilon\left(f^3 - 1\right)\lambda_{ap} \qquad \text{or} \qquad \varepsilon \leq \frac{N_p}{\lambda_{ap}\left(f^2 + f + 1\right)} \ . \tag{12}$$

In the present work, $\lambda_{ap}=1$ and N_p is around 50 and N_s around 12. Since comparable designs have N_s+1 or N_s+2 stacks, f is around 1.1 or 1.2 in this problem. To be on the safe side, here ε is chosen to be one to satisfy the bound of Eq. (12) with $f=2$.

Similar considerations apply to the choice of p_λ (except that now $f < 1$). This power should be large enough so that thinner designs which violate the critical load constraint will have higher values of the merit function ϕ than thicker designs with less constraint violation. The ratio $\lambda_{cr}/\lambda_{ap}$ is proportional to N_p (for strength) or proportional to N_p^3 (for buckling). When the plate thickness is multiplied by a factor of $f < 1$, and again assuming that at the optimum $\lambda_{cr} \approx \lambda_{ap}$, the requirement is therefore

$$N_p \leq \frac{\lfloor fN_p \rfloor}{f^{3p_\lambda}} \qquad \text{for buckling} \ , \tag{13}$$
$$N_p \leq \frac{\lfloor fN_p \rfloor}{f^{p_\lambda}} \qquad \text{for strength} \ ,$$

where $\lfloor x \rfloor$ is the integer floor function. Typically, $f = (N_s\text{-}1) / N_s$, so any $p_\lambda > 1$ would suffice for either load being critical. If fN_p is not integral, $p_\lambda = 2$ is a safe choice for the sizes N_p considered here. It should be noted, however, that Eqs. (12) and (13) apply only when we compare two feasible designs (Eq. 12) or two infeasible designs (Eq.13). When we compare a feasible design to an infeasible design, these do not apply and the values of p_λ and ε have to be tuned to the application.

Binary Tree Memory

During the evolution process, populations often contain designs that have also appeared in previous generations. If the calculation of the objective function is expensive, it is worthwhile to keep track of designs to avoid duplicate calculations. Even more important, the information about previous designs is used to construct approximations for the local improvement procedure described below. The binary tree data structure (Kernighan and Ritchie [11]) provides a way to store previous designs which permits efficient search for duplicate designs.

In previous work (Kogiso et al. [9]), the binary tree was shown to provide the objective function values for 20 to 50 percent of the analyses which would otherwise be calculated unnecessarily. We use this same binary tree structure to investigate the effectiveness of the procedure for the minimum thickness design problem.

The binary tree structure is used to store all data pertinent to the design: the encoded design string, thickness, in-plane lamination parameters, bending lamination parameters, objective function, normalized strain failure load, and normalized buckling load. Except for normalization, these data are the same as those stored by Kogiso et al. [9]. Figure 3 shows the pseudocode for the calculation of the objective function using the binary tree. After a new generation of design strings is created by the genetic operations, the binary tree is searched for each new design. If the design is found, the objective function value is obtained from the tree without analysis. Otherwise, the tree is searched for designs with identical in-plane lamination parameters. If a design with identical in-plane lamination parameters is found, the normalized failure load is obtained from the tree. The failure load is then calculated by multiplying by the thickness of the new design. Here, since the normalized failure load is independent of the thickness, the thickness of the design found from the tree is irrelevant. Otherwise, the failure load is calculated by in-plane analysis. Then the buckling load is calculated. Finally, the objective function value is obtained by considering the value of the critical load and adjusting the penalty for the ply contiguity constraint. This new design and its concomitant data are then inserted into the binary tree.

Local Improvement

To take advantage of the information about previous designs stored in the binary tree memory we use a local improvement procedure [12]. The procedure searches for improved designs in the neighborhoods of each of the designs generated by the genetic operators. As in Kogiso et al. [9], the present work considers as neighbors all the designs obtained by interchanging two stacks in the laminate. An example of a stack interchange is

$$1\ 2\ \underline{3}\ 1\ 3\ 2\ \underline{2}\ 1\ 2\ 2\ 0\ 0\ 0$$
$$1\ 2\ \underline{2}\ 1\ 3\ 2\ \underline{3}\ 1\ 2\ 2\ 0\ 0\ 0 \qquad (14)$$

Since the original designs are already packed, zeros are located at the right end of the strings, and there is no need to consider interchange with zeros, so, the two point interchange is limited to these three types ($1 \leftrightarrow 2$, $2 \leftrightarrow 3$, $3 \leftrightarrow 1$). The number of all possible laminates obtained by interchange is less than or equal to $N_S(N_S-1) / 2$.

The laminates obtained by the interchanges are not analyzed, but instead their fitness is estimated from the information on neighbouring designs stored in the binary tree memory. Since the interchange operation does not change the in-plane stiffnesses and the strain failure load, only the buckling load and ply contiguity constraints are influenced by this operation, therefore, we need to use only the bending parameters to estimate the fitness.

To define neighbourhoods and construct approximations based on information at neighbouring designs we need a measure of distance between the designs in the bending parameter space. An example of the distribution of the perturbed designs in the bending lamination parameter space is shown in Figure 4. The central point corresponds to the nominal design, and the three branches correspond to the three possible exchanges. The distance from the nominal point depends on the locations of the exchanged stacks.

The buckling load at interchanged designs is estimated by a linear approximation in the bending lamination parameters based on the buckling loads of neighbouring designs. The buckling load λ of the interchanged designs is estimated by

$$\frac{\lambda}{h^3} \approx \frac{\lambda_0}{h^3} + A\Delta W_1^* + B\Delta W_2^* \quad , \qquad (15)$$

where λ_0 is the buckling load of the nominal design, and h is the plate thickness of the nominal and interchanged designs. ΔW_1^* and ΔW_2^* are the change in the bending lamination parameters (W_1^*, W_2^*) from the nominal design. The coefficients A and B are determined as follows. First, the binary tree is searched for the five nearest neighbors of the nominal design in the Euclidean (W_1^*, W_2^*) plane. Since the normalized buckling load is used here, the thicknesses of the nearest neighbors do not matter. The coefficients A and B are determined by the best least squares approximation of the form Eq. (15) to the five nearest neighbors.

The approximate buckling load is then used to evaluate the objective function for all the perturbed designs, and the best one is used to replace the nominal one. An example of the effectiveness of this local improvement is shown in Figure 5. This data was obtained from the 10th generation of a single genetic optimization run when the population size was set to eight. Figure 5 (a) shows the buckling load of nominal and best interchanged designs for the entire population. The first column of each group is the buckling load factor of the nominal design. The second column represents the exact value of the buckling load factor for the best design which replaces that nominal design. The third column represents the approximate value for the best design used by the local improvement procedure to select the best design. Figure 5 (b) shows a similar plot for the total objective function. The Designs 1, 5, and 6 in Figure 5 (b) are not replaced, because these nominal designs are better than all of the interchanged designs. Despite the fact that the buckling load of interchanged design was lower than the nominal design, the 3rd design was replaced, because the objective function was greatly improved. The nominal Design 3 violated the contiguous ply constraint, while the local improvement selected the design which had a smaller buckling load factor but satisfied the contiguous ply constraint.

The accuracy of the approximations for all the perturbed designs of Figure 4 is shown in Figure 6. The horizontal axis in Figure 6 is the distance from the nominal design in the bending lamination plane. The vertical axis is the buckling load normalized by the buckling load of the nominal design $[90_2/(\pm 45/0_2)_4/\pm 45_2/90_2]_S$. When the circles (approximate analysis) overlap the diamonds (exact analysis), the approximation works well. The approximation fails completely when it predicts an increase in buckling load while the true buckling load is actually less than the nominal value. Figure 6 indicates such failures are likely to occur only when the change in the buckling load from the nominal design is small (in fact no such failure is observed in Figure 6). Thus, it is unlikely that the design selected as the best among the perturbed designs on the basis of the approximation is worse than the nominal design. Note that the design which has the highest buckling load is not always selected as the best perturbed design, because the selection also considers the contiguous ply constraint.

The distribution of accuracy of all points from one genetic optimization run is shown in Figure 7. The horizontal axis is the normalized improved value by approximation $(\lambda_{approx} - \lambda_{nom}) / \lambda_{nom}$, and the vertical axis is that by the exact analysis $(\lambda_{ex} - \lambda_{nom}) / \lambda_{nom}$. When the approximation works well, points lie close to the 45˚ line where λ_{approx} is equal to λ_{ex}. When the point lies above the line, the approximation underestimates the buckling load, otherwise, it overestimates the load. If the point is in the first or the third quadrants, the approximation predicts correctly whether the interchange increases or decreases the buckling load. When the point is in the second or the fourth quadrants, the approximation does not even capture the sense of the effect of the interchange. As can be seen from the figure, very few points lie in the bad quadrants, and these have small normalized values. So, these points are not likely to be selected as the best among the perturbed designs.

RESULTS

Results were obtained for a graphite-epoxy plate with the following material properties: $E_1=18.50x10^6$ psi (127.55 GPa); $E_2=1.89x10^6$ psi (13.03 GPa); $G_{12}=0.93x10^6$ psi (6.41 GPa); $\nu_{12}=0.3$; $t = 0.005$ in. (0.127 mm). The ultimate allowable strains are $\varepsilon_1^{ua}=0.008$, $\varepsilon_2^{ua}=0.029$, $\gamma_{12}^{ua}=0.015$. These allowable strains were reduced by a safety factor of 1.5. The plate has longitudinal and lateral dimensions of $a=20$ in. (0.508 m) and $b=5$ in. (0.127 m), respectively.

The genetic algorithm was applied for three load cases. Load case 1 has $N_x=13000$ lb/in (2.277x10^6 N/m) and $N_y=1625$ lb/in (2.846x10^5 N/m). Load case 2 has $N_x=12500$ lb/in (2.189x10^6 N/m) and $N_y=3125$ lb/in (5.473x10^5 N/m). Load case 3 has $N_x=9800$ lb/in (1.716x10^6 N/m) and $N_y=4900$ lb/in (8.581x10^5 N/m). These loads correspond to load ratios of $N_y/N_x = 0.125$, 0.25, and 0.5, respectively. These load magnitudes are chosen for consistency with Kogiso et al. [9] so that the optimum design for each load case is a 48-ply plate.

The (*best known*) global optimal design has the largest constraint margin out of the multiple minimum thickness designs. In this problem, there are many near optimal designs. Since the value of the objective function is dominated by the penalty values, ε and p_λ, it is not sufficient to discuss the near optimal designs only in terms of the values of the objective function itself. To evaluate the near optimal designs correctly, the designs whose critical loads are within a tenth of a percent of the critical load of the global optimum, which have the minimum thickness, and which satisfy the contiguous ply constraint, are called *practical optima* here. To evaluate the efficiency of the algorithm we define a *normalized price*, which is the average number of evaluations (also called *price*) of the objective function divided by the probability of reaching a practical optimum (called *practical reliability*). In the present work such properties were obtained by averaging 100 runs.

The convergence criterion is defined as the number of generations without improvement in the objective function, and called the *stopping criterion*. If the objective function has not improved during this number of generations, the genetic algorithm is considered to have converged.

Average prices and practical reliabilities were calculated by performing *one hundred* genetic optimizations for each of the three load cases. The algorithm was considered satisfactory only if the practical reliability was at least 0.8. This means that a single genetic optimization run has at least an 80 percent chance of finding practical optimum designs. The requirement of 0.8 practical reliability was used to determine the stopping criterion. The following parameters of the genetic algorithm were considered as default values (based on Le Riche and Haftka, [5]). The population size is 8; the crossover probability is 1.0; the permutation probability is 1.0; and the mutation probability is 0.01.

The Effect of String Length

For the plate geometry and loading considered, the global optimal design has a 48 ply laminate. The corresponding number of nonzero digits in the chosen string length is 12 because of the assumed plate symmetry and the encoding of two-ply stacks. First, we investigated the effect of the string length on the performance of genetic algorithms.

Table 1 shows the effect of the string length on the average performance of the three load cases without local improvement. The performance was averaged over 100 optimizations for each load case for a total of 300 optimizations per line in Table 1. The performance is measured both in terms of the number of analyses (price) and number of different designs, and hence the number of analyses required (nodes in the binary tree). The difference between the two numbers represents the contribution of the binary tree to avoiding repeated analyses of the same designs. The price and number of nodes in the binary tree are reported as a mean and standard deviation. The table indicates that the shorter string length has higher performance because the design space is smaller. However, if the string length is set to smaller than 12, no design can satisfy the critical load constraint. Since, in general, we do not know the solution ahead of time, we selected a string length of 16 for subsequent studies.

Performance of Genetic Algorithms without Local Improvement

The optimal designs are presented in Table 2, obtained with the parameter settings $\varepsilon = 1.0$ and $p_\lambda = 1.0$. Le Riche and Haftka [5] investigated the performance of genetic algorithms without local improvement for the critical load maximization problem and reported the optimal designs for a 48 ply laminate plate. The optimal designs obtained from the thickness minimization problem are the same as their optimal designs.

Table 3 shows the performance of genetic algorithms without local improvement for the individual load cases for both the minimum thickness and the maximum critical load designs. For the minimum thickness design problem, load case 2 had the highest price for 0.80 practical reliability. The number of practical optima influences the performance of genetic algorithms as observed earlier by Kogiso et al. [9]. Thus load case 1, with the largest number of practical optima, requires the smallest number of analyses for convergence to a practical optimum.

For all load cases, the prices are higher for the minimum thickness designs than those for the critical load maximization problem. For the minimum thickness design problem, designs must converge to the minimum thickness first before reaching the practical optima. As some generations are necessary to reach the minimum thickness before reaching the practical optima, more calculations are required to reach practical optima with 0.80 reliability. Figure 8 shows the effect of the stopping criterion on average performance from 100 optimization runs for load case 2 without local improvement. The probability of attaining minimum thickness is the proportion of runs where the best design has the minimum thickness (48 plies), when the stopping criterion is satisfied. The probability of constraint satisfaction is the proportion of runs where the best design is at the minimum thickness and also satisfies the critical load constraint ($\lambda_{cr} \geq 1$) but not necessarily the contiguous ply constraint. No design which satisfies the load constraint but does not have the minimum thickness is included in the probability of constraint satisfaction.

Figure 8 shows that more than 15 percent of the runs converged to thicker designs when the stopping criterion was set to less than 30 generations. This means that several generations were required to reach the minimum thickness, before reaching practical optima. For the buckling load maximization problem, the practical reliability was almost 0.8 (0.78 to be exact) when the stopping criterion was 63 generations without improvement. For the thickness minimization problem, a stopping criterion of about 90 generations was required to reach a practical reliability of 0.8, see Figure 8.

Performance of Genetic Algorithms with Local Improvement

Table 4 shows the performance of genetic algorithms with local improvement. Compared with Table 3, the performance improves by about 30 percent for load case 1, and improves by more than a factor of three for load case 2. For load case 3, however, the performance becomes worse and does not reach 0.80 practical reliability even if the stopping criterion is increased to 88 generations without improvement in the objective function. The average performance of these three load cases is worse than that without local improvement due to the poor performance of load case 3 which is explained next.

Performance of NZP Seeding

Kogiso et al. [9] showed that the reason for the poor performance of the local improvement was that optimal designs for load case 3 were singular optima . The optimum design has no zero plies, but designs with few zero plies are improved by adding more zero plies rather than by removing them. Seeding the initial population with "no-zero-ply" (NZP) designs was a remedy for the singularity problem. With the population sized fixed at 8, we tried two, four, or six initial NZP designs. This corresponds to 25%, 50%, or 75% of the population. The NZP designs are created randomly from the genetic codes 0, 2, and 3. The appearance probability of zero is set to 25% , and that of the others is set to 37.5% to keep the ratio of the empty stacks as 25%.

Table 5 shows the performance of NZP seeding for load case 3 with and without local improvement. NZP seeding reduces the number of analyses both with and without local improvement. However, with local improvement the practical reliability does not reach 0.8 even if the stopping criterion is increased to around 100 generations. The performance is also worse than without local improvement. This is a substantially different trend compared to the critical load maximization problem (Kogiso et al. [9]), where seeding and local improvement were much more effective and improved the performance by more than a factor of ten.

Performance of the NZP seeding for the average of the three load cases with and without local improvement is shown in Table 6. Combined with local improvement, the NZP seeding reduces the normalized price and number of nodes in the binary tree by 30 - 40 percent. Nevertheless, the effect of NZP seeding on the performance is not so strong as in the case of critical load maximization problem (Kogiso et al.[9]).

Kogiso et al. [9] discussed the effect of singularity for the fixed thickness case. The singularity there was due to the fact that as the 0˚ plies are removed from near optimum laminates, the critical load deteriorated until the last 0˚ ply is removed. Once the last 0˚ plies are removed, the critical load increases suddenly. However, the singularity does not seem to be an issue for the next thicker designs which have 13 stacks (totally 52 plies). The best 13-stack designs for load case 3 are shown in Table 7 (these designs are far from practical optima). It is clear that the best 13-stack designs have a lot of zeros, and good stacking sequences are completely different for the 12 stack and 13 stack cases. Therefore, for this load case, the transition from 13-stack designs to 12-stack designs is difficult, and many runs get stuck with 13-stack designs.

The results also demonstrate that local improvement can reduce the probability of finding the global optimum if measures to maintain population diversity are not taken. Here, seeding with NZP designs was such a measure.

Simultaneous Load Cases

We also checked the case where all three loads are applied at once. To compare with the buckling maximization case, the load is set to N_x=*12000* lb/in (2.1015x10^6 N/m), N_y=*1500* lb/in (2.6269x10^5 N/m) for load case 1, N_x=*10800* lb/in (1.8914x10^6 N/m), N_y=*2700* lb/in (4.7284x10^5 N/m) for load case 2, and N_x=*9000* lb/in (1.5761x10^6 N/m), N_y=*4500* lb/in (7.8807x10^5 N/m) for load case 3, which represents a constant N_x+N_y of 13500.

The application of all three loads simultaneously is handled by choosing the failure load to be the minimum of the three individual failure load factors (each of which is the minimum of the buckling load factor and the strain failure load factor). For local improvement, the approximation was applied to each load case independently, but the same five nearest neighbors were used to construct an approximation. This procedure leads to a compromise optimal design where several failure loads are critical. Table 8 shows the optimal and near optimal designs for this case. It is seen that while the buckling load for load case 3 is the critical load, the other two buckling loads as well as the strength failure load for load case 1 are near critical and constrain the design. Only four designs are within the region of the practical optima. These are the same designs as Kogiso et al. [9] found for the buckling load maximization problem.

For this combined load case, the multiple constraint and the small set of practical optima have a dramatic effect on the performance of the algorithm without local improvement as shown in Table 9. Even when the stopping criterion is increased to 625 generations, the practical reliabilities do not reach 0.80. With local improvement, on the other hand, as seen from Table 10, good reliability can be achieved at a cost which is not much higher than the cost of a single load case (compare to Table 4). Comparing Tables 9 and 10, local improvement reduces the number of analyses an order of magnitude.

Tuning of Penalty Values

Tuning of the penalty values (p_λ and ε) is important for accuracy and good performance. As discussed after Equation (13), $p_\lambda > 1$ is necessary to avoid having infeasible designs with better performance than the true optimum. However, large values of p_λ ,which intensify the penalty for constraint violations, have a detrimental effect on the search.

During the optimization, the optimal thickness design can be created from the next thicker or the next thinner design by the mutation operator, but the probability of mutation was set to a low value (0.01) in the present work. For the load cases considered in the present study, the optimal design has a small constraint margin (0.03 -- 0.05). In this case, most of the minimum thickness designs violate the critical load constraint. When the design violates the constraint, but the effect on the objective function is not too large, that minimum thickness design can survive. If the effect of constraint violation is too large, this minimum thickness design will probably not survive.

Figure 9 shows the effect of p_λ on the performance for load case 2 without local improvement. The best performance was obtained when p_λ was set to around 1.0. When it is set to a larger value, the genetic algorithm experiences difficulty finding minimum thickness designs. While if it is set to a smaller value (greater than 0.5), the genetic algorithm tends to converge to designs which violate the constraints and have small objective function values.

Figure 10 shows the effect of ε on the performance for load case 2 without local improvement. When e is set to a large value, the difference in the objective function values between 12 stacks and 13 stacks becomes small, and the genetic algorithm tends to converge to thicker designs.

These parameter values depend on the magnitude of the load or the constraint margin of the optimal design. We also investigated another load case, $N_x=14000$ lb/in (2.4518x10^6 N/m), and $N_y=1750$ lb/in (3.0647x10^5 N/m), where $N_y/N_x = 0.125$. The load ratio is the same as for load case 1. This load magnitude is larger than the critical load of the optimal design for load case 1. Therefore, the optimal designs become 52 ply (13 stacks). Table 11 shows the optimal design and sample designs with the highest critical load for 12-stack and 14-stack designs. The value of the objective function under several critical load penalty values is shown in Table 12. The combinations of p_λ and ε were chosen to keep the optimum design at 13 stacks. Figure 11 shows the probability of reaching the optimal thickness under several combinations of the penalty parameters. A 13-stack design is optimum for all the combinations. For $\varepsilon=15$ and $p_\lambda=1$, the probability is getting worse even though the stopping criterion is increasing. For this combination of ε and p_λ, the genetic algorithm tends to converge to 12-stack designs which have lower values of the objective function (see Table 12). Note, however, that for ε and p_λ in the theoretically predicted range, this type of failure rarely occurs.

The results indicate that the present form of the penalty function may lead to sensitivity of the reliability to choice of penalty parameters. A modified form of the penalty function which reduces this sensitivity is discussed in [13].

CONCLUDING REMARKS

A genetic algorithm with memory was applied to the problem of minimum thickness design of composite laminates subject to buckling, strength and ply contiguity constraints. Information about designs encountered during the genetic search is stored in an efficient binary tree structure. The use of memory avoids duplicating the analyses of previously encountered designs. Furthermore, the memory is used to construct approximations to the buckling loads that are then used to find improved designs in the neigbourhood of the designs generated by the genetic search. The local improvement scheme proved to be effective for reducing the number of analyses needed for the optimization, particularly under multiple load cases, where it reduced the number of required analyses by an order of magnitude. Additionally, the present case of minimum thickness design was compared to previous results for the dual problem of maximizing buckling loads for a given laminate thickness. It was found that the minimization of thickness is more difficult and requires substantially more analyses for the genetic search. Thus, to recapitulate :

- *The use of memory avoids duplicating the analyses of previously encountered composite laminate designs.*
- *Local improvement works well when the population of designs is diverse, but for small populations, it may reduce the probability of finding a global optimum in the presence of local optima.*
- *Local improvement reduced the number of analyses by an order of magnitude for design under multiple load cases.*
- *Minimization of thickness requires substantially more analyses for the genetic search as compared to the dual problem of load maximization for fixed thicness.*

ACKNOWLEDGMENTS

This work was supported in part by NASA grants NAG-1-168 and NAG-1-643, Department of Energy Grant DE-FG05-88ER25068, Air Force Office of Scientific Research Grant F49620-92-J-0236, and a scholarship from the Osaka Foundation of International Exchange.

References

1. Schmit, L. A., and Farshi, B., "Optimum Design of Laminated Fiber Composite Plates", *International Journal of Numerical Methods in Engineering*, Vol. 11, pp. 623 - 640, 1977.

2. Mesquita, L., and Kamat, M. P., "Optimization of Stiffened Laminated Composite Plates with Frequency Constraints", *Engineering Optimization*, Vol. 11, pp. 77 - 88, 1987.

3. Haftka, R. T., and Walsh, J. L., "Stacking-Sequence Optimization for Buckling of Laminated Plates by Integer Programming", *AIAA J.*, Vol. 30 (3), pp. 814 - 819, 1992.

4. Nagendra, S., Haftka, R. T., and Gürdal, Z., "Stacking Sequence Optimization of Simply Supported Laminates with Stability and Strain Constraints", *AIAA J.*, Vol. 30 (8), pp. 2132 - 2137, 1992.

5. Le Riche, R., and Haftka, R. T., "Optimization of Laminate Stacking Sequence for Buckling

Load Maximization by Genetic Algorithm", *AIAA J.*, Vol. 31 (5), pp. 951 - 956, 1993.

6. Holland, J. H., *Adaptation in Natural and Artificial Systems*, The University of Michigan Press, Ann Arbor, MI, 1975.

7. Goldberg, D.E., *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison- Wesley publishing Company, Inc., Reading, MA.

8. Nagendra, S., Haftka, R. T., and Gürdal, Z., "Design of Blade Stiffened Composite Panels by a Genetic Algorithm Approach", *Proc.of the 34'th AIAA/ASME/AHS Structures, Structural Dynamics and Materials Conference* , held at La Jolla, CA, April 19-22, pp. 2418-2436, 1993.

9. Kogiso, N., Watson, L. T., Gürdal, Z., and Haftka, R. T., "Genetic Algorithms with Local Improvement for Composite Laminate Design", *Structural Optimization*, to appear.

10. Miki, M., and Sugiyama, Y., "Optimum Design of Laminated Composite Plates Using Lamination Parameters", *Proc. of the AIAA/ASME/ASCE/AHS/ASC 32nd Structures, Structural Dynamics and Materials Conference*, Baltimore MD, Part 1, pp. 275 - 283, 1991.

11. Kernighan, B. W., and Ritchie, D. M., *The C Programming Language* , Second Edition, Prentice Hall, Englewood Cliffs, N.J., pp. 139-143, 1988.

12. Tovey, C. A., "Low Order Polynomial Bounds on the Expected Performance of Local Improvement Algorithms", *Mathematical Programming*, Vol. 35, pp. 193 -224, 1986.

13. Le Riche, R., and Haftka, R.T., "Genetic Algorithms for Minimum Thickness Design of Composite Laminates", First International Congress on Composite Engineering, New Orleans, LA, August 28-31, 1994.

FIGURE TITLES :

Figure 1. Laminated plate geometry and loading.

Figure 2. Pseudocode for genetic algorithm with local improvement.

Figure 3. Calculation of objective function with the aid of a bianary tree.

Figure 4. Example of the distribution of perturbed designs obtained by stack interchange, nominal design $[90_2 / (\pm 45 / 0_2)_4 / \pm 45_2 / 90_2]_s$, $W_1^* = 0.09838$, $W_2^* = 0.11806$.

Figure 5. Example of the effect of local improvement during the 10th generation.

Figure 6. Exact versus approximate normalized buckling loads.
Nominal design and perturbed designs showed in Figure 4.

Figure 7. Distribution of accuracy of all designs from a single run.

Figure 8. Effect of stopping criterion on practical reliability and probability that best design in the final population satisfies constraints and attains the minimum thickness. Population size = 8, permutation probability = 1.0, mutation probabilility = 0.01, $\varepsilon = 1.0$, $p_\lambda = 1.0$.

Figure 9. The effect of the penalty value (p_λ) on the performance for load case 2 without local improvement. Population size = 8, permutation probability = 1.0, mutation probabilility = 0.01, $p_\lambda = 1.0$.

Figure 10. The effect of the penalty value (ε) on the performance for load case 2 without local improvement. Population size = 8, permutation probability = 1.0, mutation probabilility = 0.01, $p_\lambda = 1.0$.

Figure 11. Probability of reaching the optimum thickness for several combinations of penalty parameters.

TABLES :