IMPROVING GENETIC ALGORITHM EFFICIENCY AND RELIABILITY IN THE DESIGN AND OPTIMIZATION OF COMPOSITE STRUCTURES

G. Soremekun*, Z. Gürdal†, R. T. Haftka‡, and L. T. Watson§

Virginia Polytechnic Institute and State University

Blacksburg, Virginia

October 1996

*Graduate Research Assistant, Department of Engineering Science and Mechanics, VPI&SU.

†Professor, Department of Engineering Science and Mechanics, VPI&SU

‡Professor, Department of Engineering Mechanics and Aerospace Engineering, University of Florida

§Professor, Departments of Computer Science and Mathematics, VPI&SU.

Abstract

Genetic algorithms (GAs) often require many iterations for convergence. If the cost for the analysis of each laminate is high, then GA optimization becomes infeasible due to the large amount of CPU time required. A genetic algorithm's ability to find optimal laminate designs that have complicated stacking sequence patterns in an efficient manner may be improved if the GA takes greater advantage of all the information generated throughout the search scheme. In a standard GA procedure, an elitist method is typically implemented where the worst laminate from the child population is replaced with the best laminate from the parent population. Valuable information that may exist in other laminates of the parent population is no longer utilized once the child population has been created. The present paper suggests new multiple elitist and variable elitist schemes where more than just the best laminate from the old population may be preserved in successive generations providing the GA with additional laminate designs with good performance. These additional designs may contain pieces of the stacking sequence pattern that are vital for achieving the optimal laminate and help the GA converge more rapidly. Results generated by utilizing the multiple elitist and variable elitist methods have shown to yield richer final populations and minor improvements in the computational cost, while maintaining a high level of reliability.

List of Figures

List of Tables

Introduction

The utilization of composite materials in structural applications has grown considerably in recent years. The major advantage in using composite materials is their high strength-to-weight ratio. The design of composite laminates is typically formulated as a continuous optimization problem with ply thicknesses and ply orientation angles as design variables.[1] However, composite ply thicknesses are discrete numbers and the design of the stacking sequence requires discrete programming. Genetic algorithms (GAs) are well suited for the problem of laminate stacking sequence design, which involves finding the correct ply orientation angles typically arranged in discrete patterns. GAs are also excellent all-purpose discrete optimization algorithms because they can handle linear and nonlinear problems or noisy search spaces by using payoff (objective function) information only. Traditional derivative based methods are illsuited to discontinuous or discrete functions.[2] GAs also work with a population of laminates instead of a single point, helping them to find the optimal design and other near optimal designs as well.

Many recent studies have concentrated on improving the genetic algorithm's reliability and efficiency in composite structural optimization. Le Riche and Haftka[3] studied the problem of composite panel weight minimization subject to buckling and strength constraints. By using a combination of penalty parameters and tuning the various genetic operators, convergence to feasible designs was guaranteed while simultaneously increasing the efficiency of the genetic search. With these modifications, a 56% reduction in the price of the search was obtained.[3]

Kogiso et al.[4] implemented two different methods for reducing the cost of the GA for maximization of the buckling load of composite plates. In the first method, a binary tree was set up to store pertinent information about laminate designs that had already been analyzed.[4] After the creation of a new population of designs, the tree is searched for laminate designs with either an identical stacking sequence or similar performance, such as laminates with identical in-plane strains. Depending on what information can be retrieved from the tree, the analysis for a given laminate may not be required or significantly reduced. While such a method can reduce computational cost, a binary tree requires a large amount of computer memory and, depending on the problem being investigated, may not be cost effective.

A second method, local improvement, involves searching in small discrete neighborhoods around nominal designs by using approximations.[5] The neighborhood area is determined by randomly interchanging any two plies in a nominal design. The performance of neighborhood designs is effectively and efficiently determined by use of a binary tree (described above) and numerical approximations based on lamination parameters. Local improvement was found to substantially reduce the cost of the algorithm.

In other approaches, such as the so called injection island technique, parallel processing is used where multiple subpopulations, with identical GA representations, are used to generate solutions. In injection island parallelism, information is both given and received by all subpopulations to obtain optimal designs. Malott and Averill[4] modified this idea by using a GA which simultaneously uses different search strategies. Strategies with increasing levels of resolution simultaneously search small abstract spaces for areas of possible interest by using child subpopulations. The level of resolution depends on the complexity the gene string of each subpopulation.[6] Information obtained from the lower resolution searches is continuoulsy passed to parent subpopulations where the most refined search is implelmented. Convergence of the low resolution searches occurs quickly and is then discontinued, saving valuable CPU time. Improvements in both the quality of designs found by the GA and the cost of the search were realized.

Large computation costs can also be avoided by reducing the complexity of the analysis required when using the GA. Time consuming analysis techniques such as the finite element method (FEM) can be eliminated when using GAs by utilizing a combination of optimization methods. Yamazaki[7] used a two-level optimization technique in maximizing the critical buckling load of composite plates and stiffened panels.[7] In the first level of optimization, structural and sensitivity analyses are carried out, taking a significant portion of the total CPU time required. Lamination parameters and the dimensions of the structure are used as design variables and optimized using a linear programming algorithm. Once the optimal lamination parameters have been determined, a second level of optimization implements a GA to find the stacking sequence that best matches the optimal lamination parameters. The second level of optimization does not require expensive structural analysis, allowing the GA to be more efficient in its search for the optimum stacking sequence.[7]

The present paper introduces the concepts of multiple elitist and variable elitist selection in which a prescribed number of designs among the top performers of the parent generation are passed on to the child generation to produce new populations in the GA optimization process. The number of top performers passed to each successive generation remains constant throughout the genetic search in multiple elitist selection and is varied in variable elitist selection. Three different implementations of the multiple elitist scheme and one implementation of the variable elitist scheme are explored using a genetic algorithm for minimum weight design of a cantilever laminated composite plate. The primary objective of the new selection module is to increase the GA's reliability and final population richness (to be defined) while decreasing the computational cost of the algorithm.

Problem description

The design problem under consideration consists of a cantilevered plate 14.0in long by 5.5in wide and loaded under a pure bending moment (M_x^a = 1200.0 lbs) applied along the right end of the panel. The symmetrically laminated composite plate is made of Glass-Epoxy; see Table 1 for material properties. Each ply in the stacking sequence may be oriented at any angle between 0 and 90 degrees in increments of 10 degrees as shown in Figure 1.

The bend-twist response of the composite plate based on Classical Lamination Theory (CLT) is the analysis used in the GA. Such composite plate problems are used as simplified models of wing or hydrofoil structures. Aerodynamic loading caused by a wind gust or certain aircraft maneuver are simulated by the applied moment at the tip of the plate, causing the wing to bend and/or twist. As mentioned earlier, ply orientation angles are restricted to positive values only. The result is an unbalanced laminate stacking sequence which will allow the plate to twist as it bends under the influence of the moment applied at the plate tip. By designing the plate for maximum opposite twist at the plate tip, the effect of the aerodynamic loading or maneuver can be reduced or eliminated, as was done in Mallot and Averill.[6]

First, to simplify the analysis, the out-of-plane deflection of the plate, $w(x, y)$, is assumed to vary linearly along the y-axis. This assumption eliminates the κ_y curvature and decouples κ_x from κ_{xy} , which are defined by

$$\begin{aligned} \kappa_x &= -\frac{\partial^2 w}{\partial x^2}, \\ \kappa_y &= -\frac{\partial^2 w}{\partial y^2} = 0, \\ \kappa_{xy} &= -\frac{\partial^2 w}{\partial x \partial y}. \end{aligned} \tag{1}$$

By integrating the expressions for the curvatures twice and imposing the boundary conditions, an expression for the out-of-plane displacement can be derived as

$$w(x, y) = -\frac{x}{2}\left[x\ \kappa_x + y\ \kappa_{xy}\right]. \tag{2}$$

A relationship between the applied moment and curvatures of the laminate is given using classical lamination theory. The symmetry of the stacking sequence eliminates coupling between bending

Table 1: Material properties for Glass-Epoxy.

Property	**Value**
Young's modulus, E_1	5.6×10^6 psi
Young's modulus, E_2	2.0×10^6 psi
Shear modulus, G_{12}	0.7×10^6 psi
Poisson's ratio, ν_{12}	0.2
Ply thickness, t	0.011 in
X_t, tensile strength	3.0×10^4 psi
Y_t, tensile strength	4.0×10^3 psi
X_c, compressive strength	3.5×10^4 psi
Y_c, compressive strength	9.0×10^3 psi
S, shear strength	1.5×10^4 psi

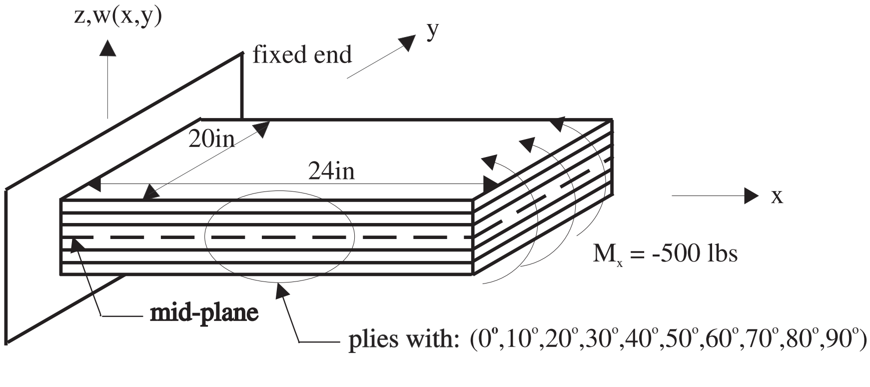

Figure 1: Plate configuration and loading conditions.

and extension, yielding the expression

$$\begin{Bmatrix} M_x \\ M_y \\ M_{xy} \end{Bmatrix} = \begin{bmatrix} D_{11} & D_{12} & D_{16} \\ D_{12} & D_{22} & D_{26} \\ D_{16} & D_{26} & D_{66} \end{bmatrix} \begin{Bmatrix} \kappa_x \\ \kappa_y \\ \kappa_{xy} \end{Bmatrix}. \tag{3}$$

By taking the partial derivative of Eq. (2) with respect to y and using Eq. (3) for the case of pure bending (i.e., $M_y = M_{xy} = 0$), the twist angle Θ_y (see Figure 1) of the plate can be determined by

$$\Theta_y = \frac{\partial w}{\partial y} = -\frac{x}{2}\, M_x d_{16}, \tag{4}$$

where $[d] = [D]^{-1}$. The twist angle is a function of x only and has a maximum value at the tip of the plate.

The Tsai-Hill failure criterion is used to ensure that the material does not fail. Using classical lamination theory once again, the stresses in the kth layer of a laminate under pure bending can be calculated using the relationship

$$\begin{Bmatrix} \sigma_x^{(k)} \\ \sigma_y^{(k)} \\ \tau_{xy}^{(k)} \end{Bmatrix} = z^{(k)} \begin{bmatrix} \overline{Q}_{11}^{(k)} & \overline{Q}_{12}^{(k)} & \overline{Q}_{16}^{(k)} \\ \overline{Q}_{12}^{(k)} & \overline{Q}_{22}^{(k)} & \overline{Q}_{26}^{(k)} \\ \overline{Q}_{16}^{(k)} & \overline{Q}_{26}^{(k)} & \overline{Q}_{66}^{(k)} \end{bmatrix} \begin{Bmatrix} \kappa_x \\ \kappa_y \\ \kappa_{xy} \end{Bmatrix}. \tag{5}$$

To determine the stresses along the principal material directions $(\sigma_1, \sigma_2, \tau_{12})$, the off-axis stresses from Eq. (5) are transformed through the ply orientation angle using the transformation equations provided by Jones.[8] The Tsai-Hill criterion for the kth ply in the laminate, P_{th}^k, is determined by

$$P_{th}^k = \frac{(\sigma_1^k)^2}{X^2} - \frac{\sigma_1^k\,\sigma_2^k}{X^2} + \frac{(\sigma_2^k)^2}{Y^2} + \frac{(\tau_{12}^k)^2}{S^2}, \tag{6}$$

where X, Y, and S are the longitudinal, transverse, and shear strengths for Glass-Epoxy, respectively. Values for the material strength constants used in Eq. (6) depend on whether the ply is in tension or compression, see Table 1. Material failure is assumed to occur if Eq. (6) yields values greater than one. Note that for a more detailed analysis, the Tsai-Hill criterion would be enforced at the plate root where stresses would be critical.

Optimization formulation

The goal of the optimization is to find the thinnest stacking sequence of the plate that will yield the greatest twist angle at the end of the plate, due to the applied bending moment, without laminate failure due to excessive stress. The stacking sequence is also constrained to have no more than three contiguous plies with the same orientation to avoid problems with matrix cracking. To limit the number of design variables, plies will be stacked in groups of three (i.e., $0_3, 10_3, 20_3$, etc.). Thus, a laminate will be penalized if any two contiguous ply stacks have the same orientaion. The contiguity constraint will be relaxed at the mid-plane of the plate where the symmetry of the stacking sequence forces the two ply stacks adjacent to the laminate mid-plane to have identical orientations. The GA optimization code works with a string that corresponds to one half of the laminate stacking sequence only, thereby automatically satisfying the symmetry constraint.

The optimization problem can be formulated as:

$$\text{maximize } \Theta_y \quad \text{such that} \quad \max_k \; P_{th}^k \leq 1, \tag{7}$$

where P_{th}^k is the value of the Tsai-Hill strength constraint for the kth ply in a laminate. Due to the nature of the problem (maximization of the twisting displacement) the weight of the laminate does not have to be incorporated into the optimization formulation. The minimum weight of a laminate will be governed by the Tsai-Hill strength constraint. If a laminate is too thin, then the material will fail, violating the strength constraint, and make the laminate design undesirable. The thinest laminates that do not violate any constraint will yield the largest twist angles at the end of the plate and the best performance.

To apply a genetic algorithm the degree of constraint violation must be transformed into penalty parameters which augment the unconstrained objective function.[1] For the problem studied here, the numerical values of the penalty parameters are used to the scale the value of the twist angle at the plate tip.

The augmented function is defined as

$$\Phi = \begin{cases} \dfrac{\Theta_y}{P_c}, & \max\limits_k \ P_{th}^k \leq 1, \\ \\ \dfrac{\Theta_y}{P_c\ (P_{th}^k)^\lambda}, & \max\limits_k \ P_{th}^k > 1, \end{cases} \tag{8}$$

where P_c is the ply contiguity constraint and is defined as

$$P_c = 1 + n_c. \tag{9}$$

The variable n_c used in Eq. (9) is the number of contiguous ply stacks with the same orientation in half the laminate, excluding the two stacks adjacent to the laminate mid-plane. If the strength constraint is not violated, then the first expression in Eq. (8) is used to determine the value of the objective function, otherwise, the second expression is used. A laminate is not penalized if the only contiguous ply stacks with the same orientation are the two found at the mid-plane of the laminate (i.e., $n_c = 0$). Laminates that do not violate any constraints will have an objective function value equal to the magnitude of the twist angle at the end of the plate.

For laminates that fail under the given loading condition (i.e., $P_{th}^k > 1$) the twist angle of the laminate is divided by P_{th}^k (k indicating the ply in the laminate which most violates the Tsai-Hill strength constraint) raised to the power λ. Thus, the greater the degree of constraint violation the greater the laminate is penalized. During testing of the algorithm it was found that very thin laminates appeared desirable even when the strength constraint was being violated. This problem was handled by adding the exponent λ to the scale factor. The value of λ may vary depending on the thickness of a ply stack. For the value of t given in Table 1, λ is set to a value of 2.

GA code

One string of genes is used to represent one half of the symmetrically laminated composite plate. Each gene in the string is represented by an integer value between 0 and 10. and determines whether the ply stack location is empty or occupied with a ply stack of prescribed orientation, see Figure 2. Empty ply stacks are used because the number of genes is fixed while the number of ply stacks in a laminate is not. An example of a decoded stacking sequence is seen in Figure 3, where E represents an empty 3-ply stack. Note that empty stacks are pushed to the outer edge (left end) of the laminate stacking sequence.

Coding the ply orientation angles as consecutive integers is not really necessary, since all the genetic operators could apply directly to the angles. However, the implementation of random choices with given probabilities is both easier to describe and program for integer intervals than for arbitrary sets of objects.

GA procedure

An initial population of genetic strings with randomly chosen genes is created first. The size of the population used in the present work remains constant throughout the genetic optimization. Various genetic operators are applied at given probabilities to generate new laminates. In order to form successive generations, parents are chosen from the current population based on their fitnesses, as described in the following subsection. Next, the crossover, mutation, and permutation operators are applied to create children, who replace their parents in the next generation. One generation after another is created until some stopping criterion is met.

0 = 1 three-ply empty stack
1 = 1 three-ply stack oriented at 0^o
2 = 1 three-ply stack oriented at 10^o
3 = 1 three-ply stack oriented at 20^o
4 = 1 three-ply stack oriented at 30^o
5 = 1 three-ply stack oriented at 40^o
6 = 1 three-ply stack oriented at 50^o
7 = 1 three-ply stack oriented at 70^o
9 = 1 three-ply stack oriented at 80^o
10 = 1 three-ply stack oriented at 90^o

Figure 2: Code key for laminate stacking sequence.

Coded Orientation: $[2/0/9/5/4/2/1]_s$

Decoded Orientation: $[E/10_3^o/80_3^o/40_3^3/30_3^o/10_3^o/0_3^o]_s$

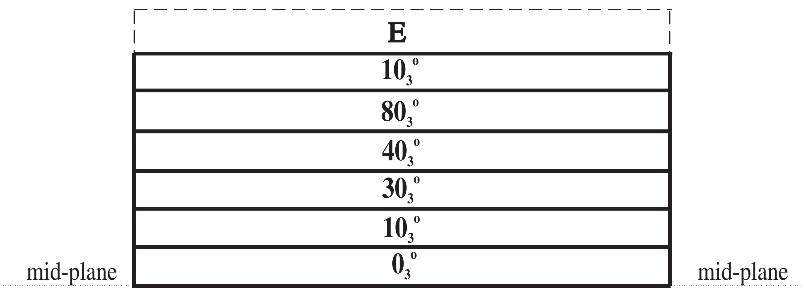

Figure 3: Sample stacking sequence arrangement.

Parent selection

Before parent selection can begin, all laminates must be ranked from best to worst according to the value of each laminate's objective function. A roulette wheel is implemented where the i^{th} ranked laminate in the population is given an interval $[\phi_{i-1}, \phi_i)$, whose size depends on the population size, P, and its rank, i, in the population:

$$\phi_i = \phi_{i-1} + \frac{2(P - i + 1)}{P(P+1)}, \tag{10}$$

where $\phi_o = 0$, and $i = 1, ..., P$. For example, if there are three laminates in a population, the roulette wheel is divided into three pieces with the best laminate taking 50% of the wheel, the second best taking 33%, and the poorest taking 17%.

A uniform random number is generated between 0 and 1; laminate i is selected as a parent if the number lies in the interval $[\phi_{i-1}, \phi_i)$. Continuing with the above example, if random numbers $0.3 \in [0, .5)$ and $0.7 \in [.5, .83)$ are drawn, then laminate 1 and laminate 2 will become parents of the first child. Parents of a child are required to be distinct laminates from the population.

Crossover

Children are created by combining a portion of each parent's genetic string in an operation called one-point crossover. The operation is applied by first drawing a random number to determine the crossover point, see Figure 4. The gene string is then split at the same point in both parents. The left piece from parent 1 and the right piece from parent 2 are combined to form a child laminate. To ensure that empty plies are not swapped, all empty plies are pushed to the left side of the coded string (this corresponds to the outer edge of the laminate). The random crossover point is restricted to fall in the non-empty region of both parent laminates, see Figure 4. If, during the creation of the child population, crossover is not applied then one of the parent laminates is cloned into the child string. Child laminates are also forced to be distinct from each other and from laminates in the parent population. If a distinct child cannot be found after a prescribed number of iterations, then one of the parents is cloned into the child population also. The crossover process is repeated as many times as necessary to create a new population of laminates.

Mutation

After a child is created, the operations of adding, deleting, or mutating genes occur with small probabilities. These operators make up genetic mutation, illustrated in Figure 5. When adding a ply stack, a uniform random number is chosen to determine the orientation. Outer plies in the laminate will get set up faster because they have a greater influence on the objective function. Thus, added ply stacks are always introduced at the mid-plane of the laminate, see Figure 5-a.

To delete a ply stack, a random number is chosen and the corresponding stack is removed from the stacking sequence by replacing it with a 0 gene. The laminate is then re-stacked so that all empty plies are pushed to the outer edge of the laminate, see Figure 5-b. Gene alteration is shown in Figure 5-c. Each gene in the string switches with a small probability to any other permissible integer value (as defined in Figure 2) except 0 and the value of the gene before ply alteration occurs. Ply alteration does not operate on empty genes either.

Ply swap

The final genetic operator is ply swap. Ply swap is implemented by randomly selecting two genes and switching their positions, see Figure 6.

Parent 1 - [**0/0/0/3/7/4/2/9/1** 10/8/1]$_s$
Parent 2 - [0/0/3/4/2/9/6/10/1 **4/7/7**]$_s$

- crossover point

Child - [**0/0/0/3/7/4/2/9/1/4/7/7**]$_s$

Figure 4: One-point crossover.

Before Ply Addition - [0/0/**0**/1/4/9/2/10/3/6/1/2]$_s$
After Ply Addition - [0/0/1/4/9/2/10/3/6/1/2/**7**]$_s$

a) Ply Addition

Before Ply Deletion - [0/0/0/1/4/9/**2**/10/3/6/1/2]$_s$
After Ply Deletion - [0/0/0/1/4/9/**0**/10/3/6/1/2]$_s$
After Re-stacking - [0/0/0/**0**/1/4/9/10/3/6/1/2]$_s$

b) Ply Deletion

Before Ply Alteration - [0/0/0/1/**4**/9/2/10/3/**6**/1/2]$_s$
After Ply Alteration - [0/0/0/1/**7**/9/2/10/3/**9**/1/2]$_s$

c) Ply Alteration

Figure 5: Mutation.

Before Ply swap - [0/0/9/**1**/3/5/3/7/7/**2**/8/4]$_s$
After Ply swap - [0/0/9/**2**/3/5/3/7/7/**1**/8/4]$_s$

Figure 6: Ply swap.

Creating the next generation

The fundamental difference in the GA proposed here from the typical GA is in the determination of the next generation. The purpose of the various genetic operators is to produce successively improved populations of laminates. There is, however, no guarantee that there will be a design in the child population that is better than the best design of the parent population. In order to preserve the largest value of the objective function an elitist method is typically used.

Elitist selection

An elitist method (designated EL) ranks the child population and parent population separately. The best laminate from the parent population and the worst laminate from the child population are identified. To create the new population, the best laminate from the parent population replaces the worst laminate from the child population. The EL method is shown in Table 2. The potential problem with EL selection is that important genetic information that may exist in other desirable laminates of the parent population is lost, such as laminate B in Table 2. If such information can be identified and preserved, it may help the GA converge in less time, find better laminates, and generate additional laminates in the final population with very good fitnesses.

Multiple elitist selection

The multiple elitist scheme is designed to preserve more information about good laminate stacking sequences from the parent population than the elitist method. Three different methods of multiple elitist selection are explored.

In the first multiple elitist selection method, ME_1, the parent and child populations of size P are each separately ranked from best to worst. Both populations are then combined and ranked together

Table 2: Elitist method (P = 5).

Generation	Laminate	Fitness
Parent Population	A	61
	B	53
	C	40
	D	26
	E	15
Child Population	K	60
	L	37
	M	25
	N	23
	O	12
New Population	A	61
	K	60
	L	37
	M	25
	N	23

(forming $2P$ laminates). The number of top laminates from the combined population which will be carried over to the next generation is designated N_k. If N_k is equal to P (the maximum possible value of N_k), then the next generation is filled with the first half of the combined population and the procedure is complete. If N_k is less than P, then N_k laminates from the combined population comprise the first part of the new generation.

To fill the remainder of the new generation the ranked child population is searched, starting from the best laminate, for laminates that have not already been passed on to the new generation. The first child found which is not located in the new generation fills the first empty location in the new generation, and successive children fill out the rest of the new generation. The ME_1 selection method is demonstrated in Table 3 for a population size of 5 and $N_k = 3$.

Although the ME_1 method may appear identical to the EL method when $N_k = 1$, this is not the case. When the top laminate comes from the child population the next generation will consist entirely of child laminates. However, if the top laminate comes from the parent population, then the ME_1 and EL methods are identical.

The second multiple elitist scheme, ME_2, is very similar to the ME_1 scheme. The difference between the two methods is in the way child laminates are added to the new population after N_k laminates from the combined population have been selected. In ME_2 selection, laminates from the portion of the child population not already in the new population will be randomly chosen to fill the remainder of the new population. This method will allow for the possibility of some of the poorer laminates, which may contain important genetic information, from the child population to migrate to the new population. Both ME_1 and ME_2 selection are identical if N_k is set equal to the population size. The ME_2 scheme is shown in Table 4.

The final multiple elitist selection scheme, ME_3, is an extension of the EL method. The top N_k laminates from the population are selected and placed into the new population. The number

Table 3: ME_1 selection method (P = 5, N_k = 3).

Generation	Laminate	Fitness
Parent Population	A	61
	B	53
	C	40
	D	26
	E	15
Child Population	K	60
	L	37
	M	25
	N	23
	O	12
Combined Population	A	61
	K	60
	B	53
	C	40
	L	37
	D	26
	M	25
	N	23
	E	15
	O	12
New Population	A	61
	K	60
	B	53
	L	37
	M	25

Table 4: ME_2 selection method (P = 5, N_k = 3).

Generation	Laminate	Fitness
Parent Population	A	61
	B	53
	C	40
	D	26
	E	15
Child Population	K	60
	L	37
	M*	25*
* - chosen randomly	N	23
	O*	12*
Combined Population	A	61
	K	60
	B	53
	C	40
	L	37
	D	26
	M	25
	N	23
	E	15
	O	12
New Population	A	61
	K	60
	B	53
	M	25
	O	12

of child laminates required to fill the remainder of the new population are created from the parent population and placed into the new population. For ME_3 selection, the value of N_k cannot be set equal to the population size since this would code all the parent laminates into the new generation and prevent the GA from exploring the design space. Note that setting $N_k = 1$ in ME_3 selection is a variation of the EL selection method. The ME_3 selection scheme is depicted in Table 5.

Variable elitist selection

The purpose of variable elitist selection (VE) is to determine what the effects of using multiple values of N_k during an optimization run will have on the reliability, richness, and cost of the algorithm. The VE scheme used in the present study is a modified version of ME_1 selection. To implement this method, a value of N_k is selected and the optimization process is started using the ME_1 selection scheme. After a certain number of generations without improvement in the fitness of the best laminate, the value of N_k is increased to the value of the population size (the maximum possible value) and is held constant until the GA converges.

Comparing selection methods

To compare the different selection methods, three criteria will be used. The first criterion is apparent reliability which is determined by taking the number of runs that the GA found the best known design and dividing it by the total number of runs conducted. Given n optimization runs with apparent reliability R, the standard deviation of R is given as

$$\sigma_r = \sqrt{\frac{R(1-R)}{n}}. \tag{11}$$

In the present paper, a reliability of 90% or greater will be considered acceptable, corresponding to $\sigma_r = 4.2\%$.

Table 5: ME_3 selection method (P = 5, N_k = 3).

Generation	Laminate	Fitness
Parent Population	A	61
	B	53
	C	40
	D	26
	E	15
Child Population	K	60
	L	37
New Population	A	61
	B	53
	K	60
	C	40
	L	37

The richness percentage of the final population will be the second criterion. Population richness is calculated by taking the total number of laminates in the final population of each run with objective values within 0.5% of the best known design and dividing it by the product of the total number of runs conducted and the population size. Richness percentages will only be considered if the reliability of the algorithm is 90% or greater.

A final criterion will be the cost of the genetic search. The cost of the algorithm is measured by the average number of analyses per run required for convergence. For the EL, ME_1, ME_2, and VE selection schemes, the average number of analyses conducted over n optimization runs is determined by

$$A_n = \frac{X_g P}{n}, \tag{12}$$

where X_g is the number of generations analyzed, and P is the size of the population. The cost calculation for the ME_3 selection scheme is determined by

$$A_n = \frac{X_g N_c}{n}, \tag{13}$$

where N_c is the number of child laminates created in each generation.

The convergence criterion for the GA is 300 generations without improvement of the fitness of the best design in the parent population. Again, computation cost of the algorithm will only be considered if reliability percentages are greater than 90%.

Fifty runs will be conducted for the EL method and for each value of N_k in the ME_1, ME_2, ME_3, and VE methods to accurately determine the reliability, richness, and computation cost of the algorithm. Mean values for the computational cost from the 50 runs are also reported.

Results

Results will be presented and compared in this section to determine the positive and negative aspects of each selection method. For all runs conducted, the maximum number of ply stacks used in the coded stacking sequence is 15 (30 for the entire laminate). Thus, with 11 choices for each design variable, there are $11^{15} \cong 4 \times 10^{15}$ possible laminate designs.

For this problem, two optimum design points were found for all runs conducted, see Table 6.

These designs exhibit significant differences in stacking sequence with only a difference of approximately 1.7% in fitness.

Results also showed that as N_k approached the size of the population in the multiple elitist and variable elitist selection methods, the GA's ability to find the optimal design was significantly reduced (i.e., there was premature convergence to the local optimum design). This can be explained by looking at the method of reproduction in the multiple elitist selection. If N_k is set to a large value, then once the GA converges to a particular stacking sequence pattern the majority of the remaining laminates in the population will be saturated with similar stacking sequences also. This is an undesirable feature of multiple selection since the ability of the genetic operators to explore other regions of the design space is disabled.

For example, if the ME_1 selection scheme is used and N_k is set equal to the population size of 10, a final population of laminates similar to the one shown in Table 7 will result if the GA converges to the local optimum design. Consider what happens if the top two laminates, which are most likely to be chosen, are selected as parents. After crossover is applied, the resulting child laminate will have the exact same stacking sequence as the second parent, regardless of the crossover point that is selected. In fact, this phenomenon will occur if any pair of the top six, eighth or ninth laminates are selected as parents. Furthermore, the ply alteration and permutation operators can not alter the stacking sequence enough to guide the GA away from the local optimum. Thus, once the outermost plies of the stacking sequence have been fixed, they can never be significantly altered again. The severity of this problem is exacerbated because the outermost plies have the most influence on the fitness of the laminate, due to the nature of the problem.

The GA is less likely to get trapped at the local optimum design point when EL selection is used, or when N_k is kept small when using multiple elitist or variable elitist selection. The reason for this is that while the top few laminates in the population are preserved the remaining laminates are free to search the design space for the optimal design (i.e., the opportunity exists for the top laminates to get crossed with other dissimilar designs in the population).

The advantage of the VE selection method is that a low value of N_k at the beginning of the search allows the GA to converge to a near optimal stacking sequence. When the GA is close to converging, the value of N_k is maximized to rigorously search the design space around the optimum.

It also became apparent that the richness criterion is not of particular importance for the problem being studied here. Laminates that qualified for the richness criterion were simple permutations of the optimal stacking sequence with the best fitness, where the orientation of the inner most ply (which has little affect on the bend-twist performance of the plate) is switched to other permissible angles. An example of this is shown in Table 7. Thus the final population of an optimization run which found the optimal stacking sequence and had a high richness percentage did not possess other laminates with significantly different design characteristics.

Other problems, where many designs with good performance have different design properties, may benefit from high richness percentages. Thus, richness percentages will be given for all runs conducted to give the reader an idea of how alternate selection schemes can increase the richness of the final population.

Table 6: Properties of optimum and local optimum laminate designs.

Optimum Design	P_{th}^{k}
$[E_7/\mathbf{20_3/10_3/20_3/10_3/20_3/30_3/20_3}]_s$	0.9995
Local Optimum Design	P_{th}^{k}
$[E_7/\mathbf{10_3/20_3/30_3/20_3/30_3/90_3/20_3}]_s$	0.9968

Table 7: Convergence to local optimum design, ME_1 selection, $P = N_k = 10$.

Laminate	**Rank**
$E_7/10_3/20_3/30_3/20_3/30_3/90_3/20_3]_s$	1
$E_7/10_3/20_3/30_3/20_3/30_3/90_3/30_3]_s$	2
$E_7/10_3/20_3/30_3/20_3/30_3/90_3/80_3]_s$	3
$E_7/10_3/20_3/30_3/20_3/30_3/90_3/10_3]_s$	4
$E_7/10_3/20_3/30_3/20_3/30_3/90_3/40_3]_s$	5
$E_7/10_3/20_3/30_3/20_3/30_3/90_3/70_3]_s$	6
$E_7/10_3/20_3/30_3/20_3/30_3/20_3/90_3]_s$	7
$E_7/10_3/20_3/30_3/20_3/30_3/90_3/50_3]_s$	8
$E_7/10_3/20_3/30_3/20_3/30_3/90_3/60_3]_s$	9
$E_7/10_3/20_3/30_3/20_3/30_3/20_3/30_3]_s$	10

The remainder of this section will be split into two parts, results for a population size of 20 and those for a population size of 40. Large population sizes were required to ensure that acceptable levels of reliability could be obtained and to monitor the effect that different population sizes had on the various selection methods. Furthermore, since the performance of the GA using multiple elitist selection was degraded for large values of N_k, its value was varied from one to half the population size only for all results presented. Using the VE selection method and a population size of 20, N_k is held fixed at a preset value between 1 and half the population size until the GA records its 100th iteration without improvement in the best laminate, and is then increased to the size of the population. When VE selection is used in conjunction with a population size of 40, the value of N_k is held fixed until the GA records its 50th iteration without improvement. GA operator probabilities for all runs conducted are given in Table 8.

Population size: 20

Reliability and richness percentages are shown in Figure 7 for all selection methods. Data at the top and bottom of the plot represent reliability and richness respectively. Acceptable levels of reliability could only be achieved for values of $N_k \leq 5$ when using multiple elitist selection methods. For these values of N_k richness percentages were increased up to 15% compared to 5.5% for elitist selection.

Results obtained with VE selection show that acceptable levels of reliability could only be obtained for values of $N_k \leq 2$. However, variable elitist selection produced richness percentages 35 percentage points higher than ME selection and 50 percentage points higher than EL selection while maintaining acceptable levels of reliability.

A comparison of the cost of the elitist method with each of the multiple elitist and variable elitist selection schemes is depicted in Figures 8, 9, 10, and 11. The errors bars in each figure represent ± one standard deviation for the 50 runs conducted using each multiple and variable

Table 8: GA operator implementation probabilities.

Operator	**Probability**
Crossover	1.00
Single ply alteration	0.02
Ply addition	0.05
Ply deletion	0.10
Permutation	0.80

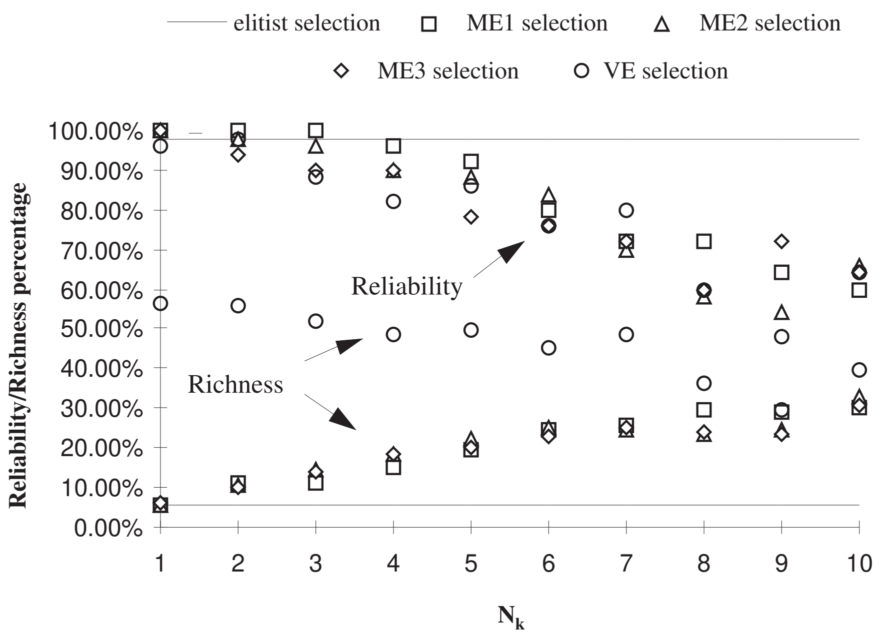

Figure 7: Reliability/Richness % comparison – EL vs. ME_1, ME_2, ME_3, and VE ($P = 20$).

elitist selection method. Results using multiple elitist selection show that for acceptable levels of reliability ($N_k \leq 5$ for ME_1, $N_k \leq 4$ for ME_2, $N_k \leq 3$ for ME_3), lower computational costs were achieved for al ME selection methods when looking at the average number of analyses per run. Lower computational costs were also achieved for VE selection while maintaining acceptable levels of reliability ($N_k \leq 2$).

The ME_3 selection scheme gave the best cost results which makes sense since fewer analyses are required when creating each new generation of laminates. For a value of $N_k = 3$, ME_3 selection will take less analyses to converge for any given run, than on average for EL selection. For all other runs conducted, there was essentially no difference between selection methods when looking at the computational cost of the algorithm. To show a clear improvement in computational cost, the average number of analyses plus one standard deviation for any alternate selection method should be less than the average number of analyses minus one standard deviation for the elitist method. It is also important to note that by relaxing the convergence criterion to 120 iterations without improvement of the best laminate for the EL selection method, the minimum level of acceptable reliability, 90%, can be achieved on average in approximately 4700 analyses with a standard deviation of ± 1200 analyses.

Population size: 40

Further improvement in reliability and richness percentages is realized when using a larger population size, see Figure 12. Acceptable reliability percentages can be achieved for all multiple elitist selection methods for values of N_k up to 9, and up to 10 for variable elitist selection. Richness percentages obtained using ME selection are improved up to 25% (when $N_k = 9$) compared to 5.5% for the elitist selection scheme. Results obtained using VE selection show that richness percentages are improved up to 25% for all values of N_k up to 10 when compared to EL selection also. VE

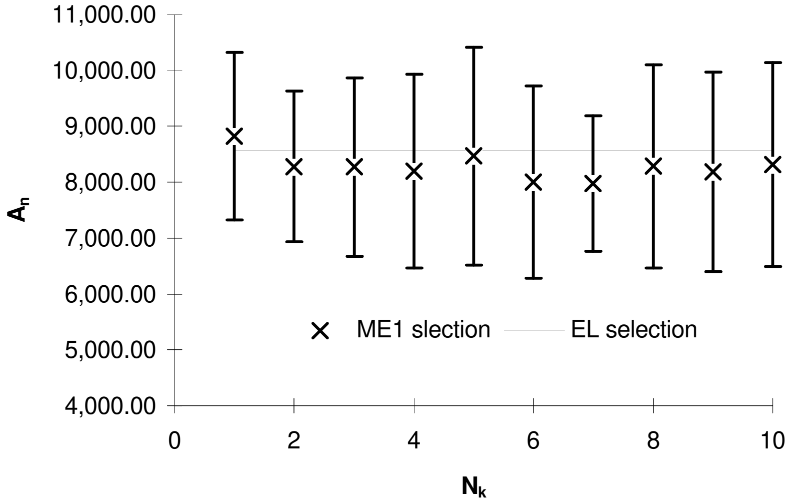

Figure 8: Cost comparison - EL vs. ME_1 ($P = 20$).

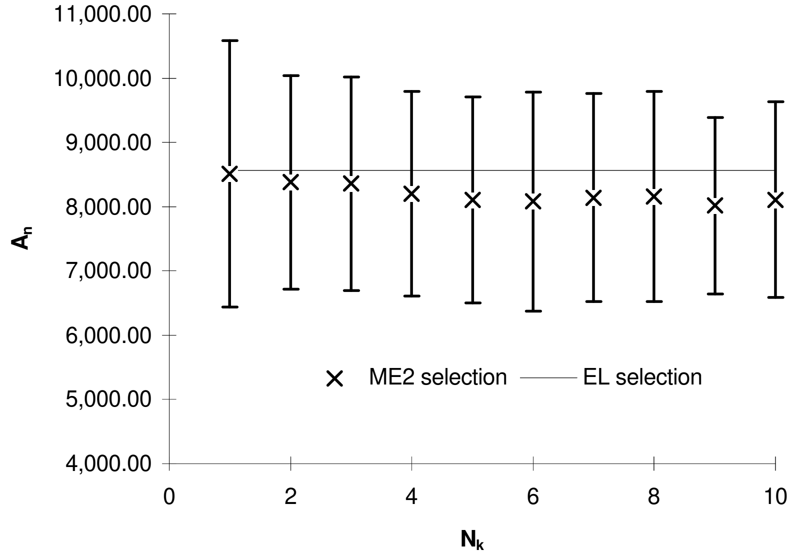

Figure 9: Cost comparison - EL vs. ME_2 ($P = 20$).

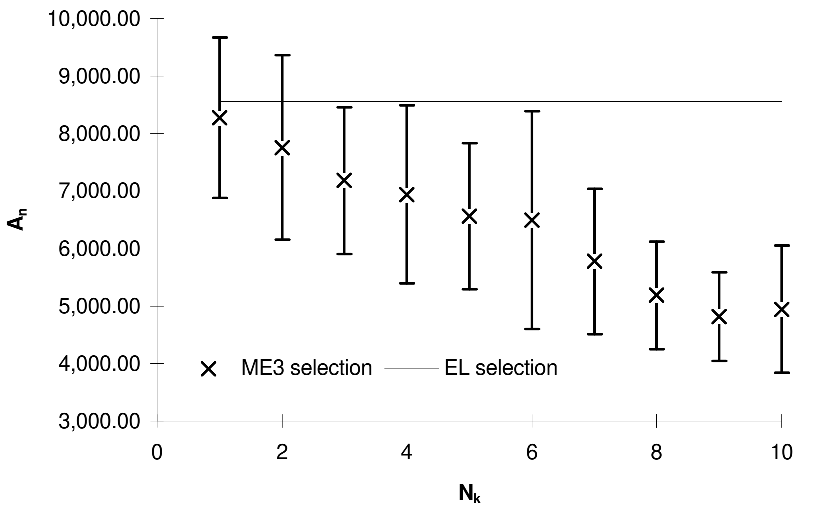

Figure 10: Cost comparison - EL vs. ME_3 ($P = 20$).

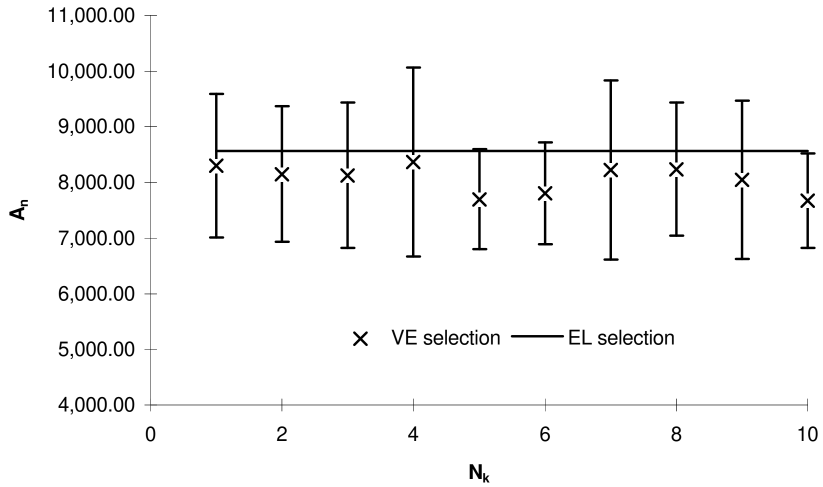

Figure 11: Cost comparison - EL vs. VE ($P = 20$).

selection also generated better richness percentages than ME selection, especially for lower values of N_k. Note also that richness percentages obtained for a population size of 40 are lower when compared to results for a population of 20. This is because the total number of laminates used to calculate the richness percentage increases with an increase in population size while the number of laminates within 0.5% of the fitness of the optimal laminate remains constant.

Improvements are also achieved using multiple and variable elitist selection when comparing the computational cost of the algorithm for the larger population size, see Figures 13, 14, 15, and 16. Once again, ME_3 selection showed best results for multiple elitist selection. For any value of N_k between 3 and 10 in ME_3 selection, the algorithm can take approximately 2000 fewer analyses on average to converge when compared to EL selection, and this difference is statistically significant. Similar results were obtained for the other selection methods also.

Increased Design Space:

For the design problem described earlier, the design space was artificially reduced by creating stacks of 3 plies. In thi section we remove that limitation and allow each ply to assume one of the ply orientations listed in Figure 2. With this modification, the maximum number of plies in half the laminate stacking sequence that any laminate can have is increased to 30 (60 for the entire laminate) giving $11^{30} \cong 1.74 \times 10^{31}$ possible laminate designs. The ply contiguity constraint, on the other hand, was kept the same requiring no more than 3 plies of the same orientation to appear adjacent to one another. Therefore, the GA implementation of this constraint was modified.

The optimal design sequence for the modified design problem is given in Table 9. By using single plies in the stacking sequence, the thickness of the optimal laminate for the modified problem is two plies thinner than the optimal design found when using 3-ply stacks. This is because laminate thickness can decrease in increments of 0.011 inches (by dropping a single ply) instead of 0.033

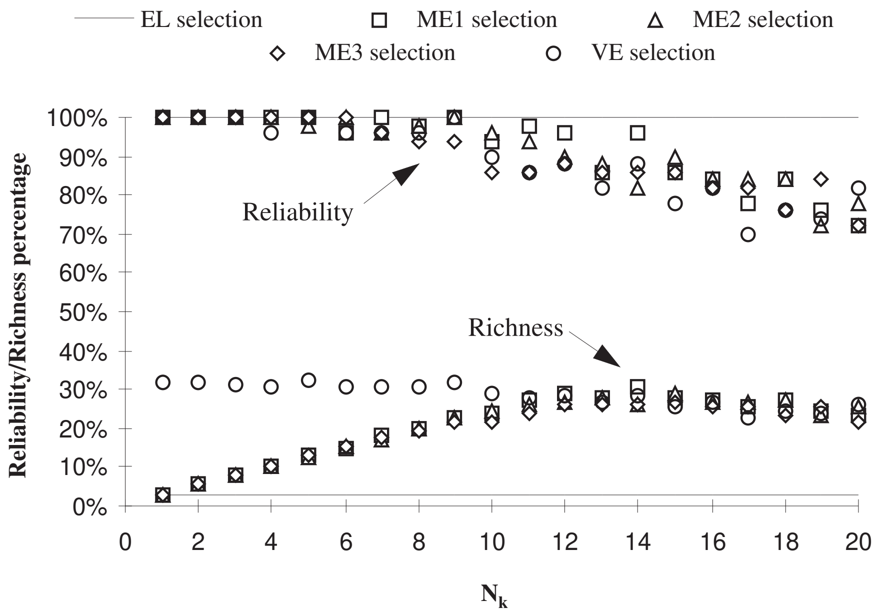

Figure 12: Reliability/Richness % comparison – EL vs. ME_1, ME_2, ME_3, and VE ($P = 40$).

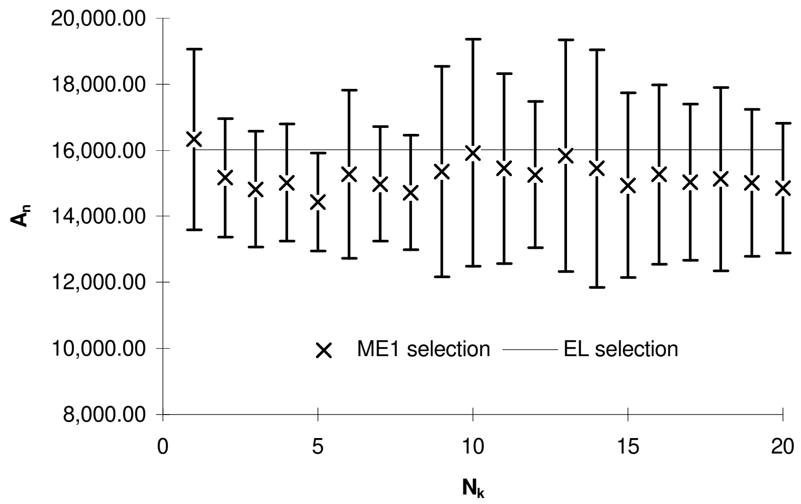

Figure 13: Cost comparison - EL vs. ME_1 ($P = 40$).

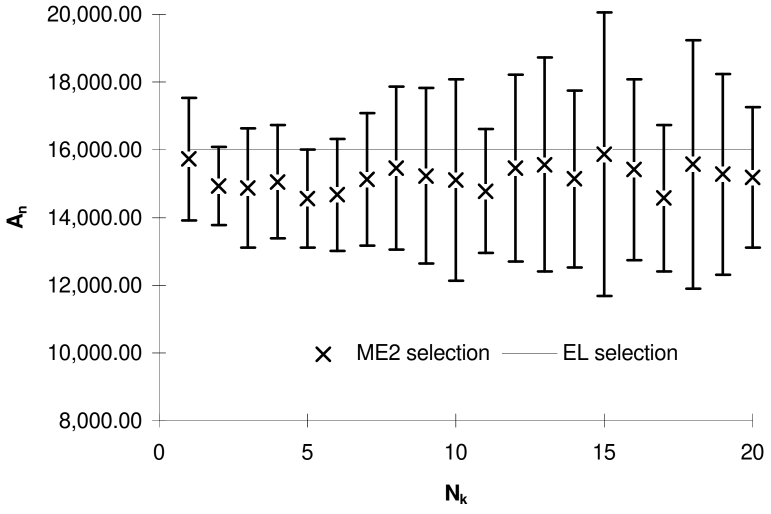

Figure 14: Cost comparison - EL vs. ME_2 ($P = 40$).

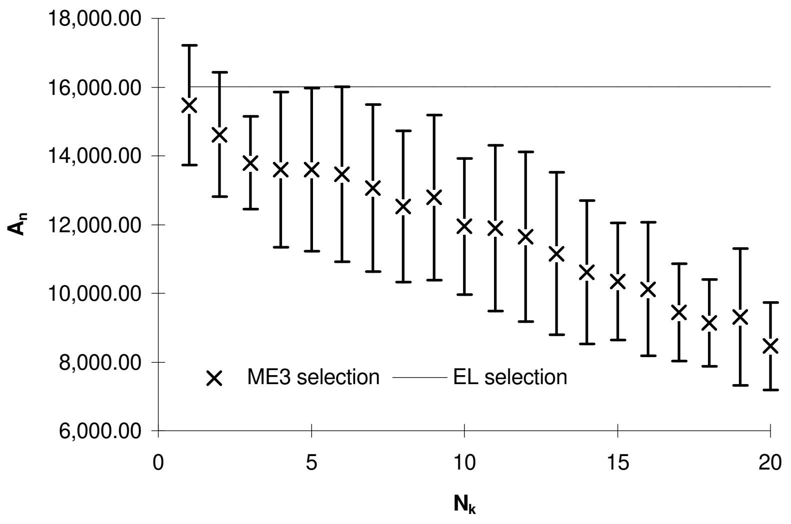

Figure 15: Cost comparison - EL vs. ME_3 ($P = 40$).

inches (by dropping a 3-ply stack).

Results with the larger design space have been obtained for the EL and ME_1 selection schemes. In order to produce acceptable levels of reliability the size of the population was increased to 200. Fifty optimization runs were conducted once again to compare both selection schemes. Figure 17 shows a comparison of richness and reliability percentages for both selection methods. The value of N_k was varied from 1 to 100 for ME_1 selection. The EL scheme found the optimal stacking sequence only once in 50 runs. However, for small values of N_k (between 15 and 30) ME_1 selection produced reliaiblity percentages as high as 84%. Richness percentages were substantially higher for ME_1 selection also.

Conclusions

Acceptable levels of reliability could not be achieved for large values of N_k when using multiple or variable elitist selection.

For small values of N_k, the reliability of the EL selection was so high that little or no improvement could be obtained with alternate selection schemes. Thus, it was of interest to try to improve final population richness and decrease computational cost of the algorithm while maintaining a high level of reliability. Population richness (although not significant for this problem) was increased substantially using multiple elitist and variable elitist selection while maintaining high levels of reliability. Small reductions in the computational cost were also realized using all alternate selection methods, with ME_3 selection showing the most improvement.

In summary, a multiple elitist selection scheme is reliable only for $N_k \ll P$, and for such N_k the multiple elitist and variable elitist selection schemes significantly increased richness at cost and reliability comparable to the elitist strategy. It is important to note also that better results were

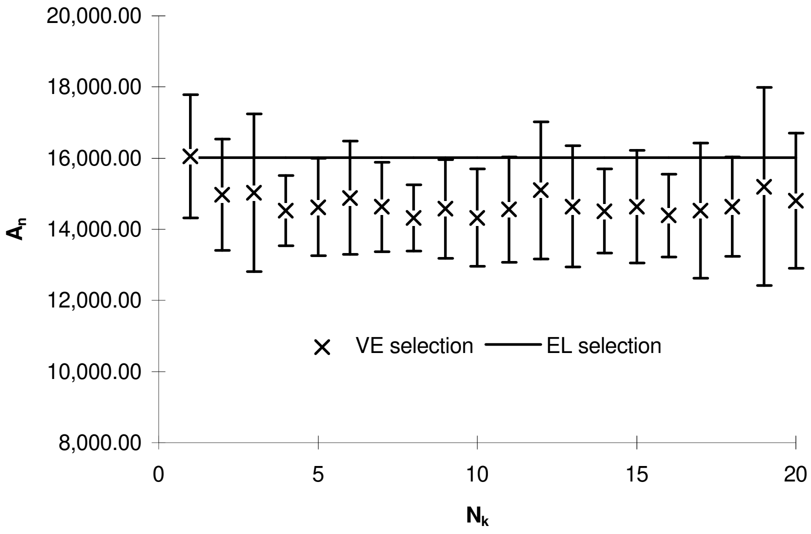

Figure 16: Cost comparison - EL vs. VE ($P = 40$).

Table 9: Properties of optimum laminate design using single-ply stacks.

Optimum Design	P_{th}^{k}
$[E_{10}/(\mathbf{10/20})_\mathbf{2}/\mathbf{10_3/20/10_2}$ $\mathbf{/20_2/10_3/20/10_2/20/10}]_\mathbf{s}$	1.0000

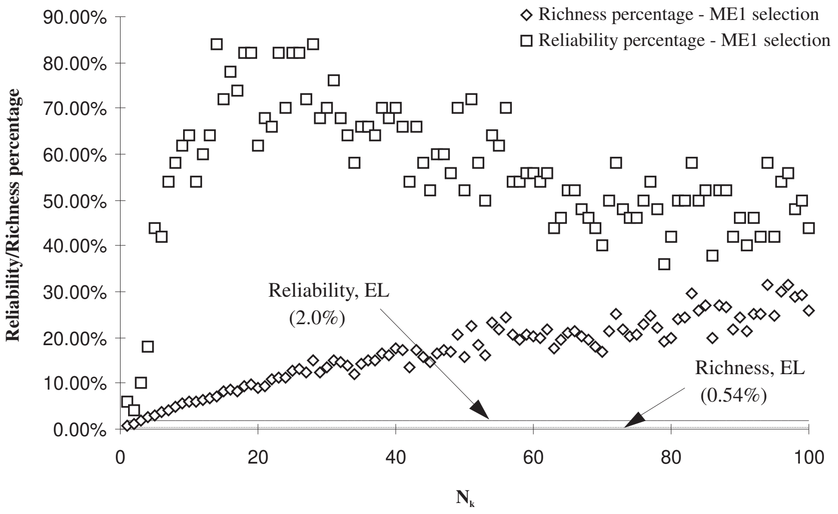

Figure 17: Reliability/Richness % comparison – EL vs. ME_1 ($P = 200$).

achieved when the population size was increased for this problem and when the size of the design space was increased. Thus, optimization problems with large design spaces that are well suited to GAs and require large population sizes may benefit from multiple or variable elitist selection schemes.

Acknowledgement

This research was supported in part by the Air Force Office of Scientific Research Grant F49620–96–1–0104.

References

1 Gürdal, Z., Haftka, R. T., and Nagendra, S., *Genetic Algorithm for the Design of Laminated Composite Panels.* SAMPE Journal, Vol. 30, No. 3, May/June 1994, pp. 29–35.

2 Goldberg. D. E., *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison - Wesley, Reading, MA., 1989.

3 Le Riche, R., and Haftka, R. T., *Improved Genetic Algorithm for Minimum Thickness Composite Laminate Design.* Proceedings of International Conference on Composite Engineering, Aug. 28–31, 1994, New Orleans, LA.

4 Kogiso, N., Watson, L. T., Gürdal Z., Haftka, R. T., and Nagendra, S., *Design of Composite Laminates by a Genetic Algorithm with Memory.* Mechanics of Composite Materials and Structures, Vol. 1, 1994, pp. 95–117.

5 Kogiso, N., Watson, L. T., Gürdal Z., and Haftka, R. T., *Genetic Algorithms with Local Improvement for Composite Laminate Design.* Structural Optimization, Vol. 7, No. 4, 1994, pp. 207–218.

6 Malott, B., and Averill, R. C., *Use of Genetic Algorithms for Optimal Design of Laminated Sandwich Panels with Bend Twisting Coupling.* AIAA Paper 96-1538-CP, 1996.

7 Yamazaki, K., *Two-Level Optimization Technique of Composite Laminate Panels by Genetic Algorithms.* AIAA Paper 96-1539-CP, 1996.

8 Jones, R.M., *Mechanics of Composite Materials.* Hemisphere, New York, 1975.