ELSEVIER

PII: S1359-8368(97)00030-9

Composites Part B **29B** (1998) 277–285

1359-8368/98/$19.00

Stacking sequence optimization by a genetic algorithm with a new recessive gene like repair strategy

Akira Todoroki[a,*] and Raphael T. Haftka[b]
[a]*Department of Mechano-aerospace Engineering, Tokyo Institute of Technology, 2-12-1,Ohokayama, Meguro-ku, Tokyo 152, Japan*
[b]*Department of Aerospace Engineering, Mechanical and Engineering Science, University of Florida, Gainesville, FL 32611, USA*
(Received 1 July 1996; accepted 30 June 1997)

A genetic algorithm is used to obtain the stacking sequence of the laminate that has the set of lamination parameters that are the closest to a set of target lamination parameters. In addition, the laminate is required to be balanced and to have no more than 4 contiguous plies of the same orientation in order to satisfy practical considerations. This problem is a constrained combinatorial optimization problem which is usually difficult to solve. The difficulty of enforcing constraints in genetic optimization is handled by introducing a new repair strategy. The new repair strategy does not alter genes but only changes decoding rules, and is similar in this respect to the way recessive genes operate in biology. The relationship between the reliability of the genetic algorithm and the probability of repair was investigated, and it is shown that 100% probability of repair is optimal when the target laminates include 45° plies. Since practical composite laminates usually include 45° plies, it is concluded that the repair strategy discussed herein should always be used with the optimization. © 1998 Elsevier Science Limited. All rightss reserved

(Keywords: A. laminates; optimum design; genetic algorithm; stacking sequence; lamination parameters)

INTRODUCTION

Because of the high specific stiffness and strength, polymer matrix composites have greatly attracted the attention of structural designers. Since laminated composite plates provide design freedom through tailoring of the stacking sequence, optimum design of stacking sequences can help to obtain high performance laminates. Miki[1] has developed a graphical laminate optimization method that is based on lamination parameters. This method has the advantage that a designer can easily visualize the entire design space. However, most practical composite laminates are restricted to some discrete sets of ply orientation angles such as 0°, 90° and ± 45° because of the availability of test data for structural verification of the behavior. This practical consideration makes the stacking sequence design problem a combinatorial optimization problem, which is not easy to solve. Haftka and Walsh[2] have applied an integer programming approach for stacking sequence optimization problems of composite laminate buckling.

Genetic algorithms have recently attracted attention for solving these difficult combinatorial optimization problems and there were several papers on optimization of composite structures by the use of this method[3–8]. Genetic algorithms are probabilistic global optimization methods, in which design variables are coded into individual genes or chromosomes and for which an individual design has one or more chromosomes. This optimization method works with a population of designs rather than a single design. New generations of designs evolve from previous generations by applying crossover and mutation operations to the genetic strings in a manner similar to the evolution process of living creatures. Moreover, an individual design that has high fitness (as measured by the objectives of the optimization) also has a high probability of producing descendants.

Because genetic algorithms usually require a large number of analyses, the stacking sequence design problem was solved in the present study by a two level procedure that was proposed by Yamazaki[9]. At the top level, the laminate is designed by using lamination parameters as continuous design variables. At the lower level, the stacking sequence is designed to match the set of lamination parameters obtained at the upper level. The present paper addresses this lower-level problem of designing a laminate with a set of

* Author to whom correspondence should be addressed. Fax: +81-3-5734-3178

lamination parameters that are closest to a target set of lamination parameters.

To avoid some undesirable stiffness coupling effects in a laminated composite structure, angle plies must be balanced. Moreover, to prevent large-scale matrix cracking, which increases the risk of delamination, the same ply orientation should not be stacked over more than four contiguous plies. These practical considerations are difficult constraints for genetic algorithms, and implementation details can greatly affect computational cost. Le Riche and Haftka[4] and Harrison *et al.*[5] have addressed these problems by using a penalty function. In addition, they have developed a special decoding rule for angle plies (such as 45° plies) that helps satisfy the ply balance constraint. In the present study, however, a repair strategy is used to enforce the two practical constraints mentioned earlier.

The new repair process developed in the present paper does not alter genes but changes the decoding rules used to convert the genes into a laminate configuration. In this novel approach, the repair process is similar to the way recessive genes work in biological systems. This point is important because repair systems that alter genes often prevent the occurrence of beneficial evolution that requires two successive mutations (because the genes are repaired after the first one). Orvosh *et al.*[10] reported that the conventional repair mechanisms should be applied with less than 100% probability. Therefore, the relationship between probability of repair and design reliability is investigated in detail in the present study.

The program written to implement the genetic algorithm employs the object-oriented language C++. All of the data and algorithms are encapsulated in containers called classes. Moreover, all of the procedures are implemented by the process known as message passing. The genetic algorithm is encapsulated in a class GA. The class GA evaluates individual fitness through a derived class of an abstract class Model by using the process called message passing. Therefore, this program is highly adaptable for new problems by implementing only a special model class as a derived class of an abstract class Model.

OPTIMIZATION FORMULATION AND SOLUTION

The design objective of the present study is to obtain stacking sequences of symmetric laminates that have lamination parameters that are the closest to target lamination parameters. Ply angles are limited to 0°, ± 45° and 90° in the present study in accordance with the available experimental data. This limitation makes the design of stacking sequences a combinatorial optimization problem.

Two constraints are applied to the combinatorial stacking sequence optimization problem. The first constraint is a limit of 4 contiguous plies with the same fiber orientation, which reduces the chance of unwanted matrix cracking. This constraint is referred to '4-contiguous-ply-rule' in the following. The second constraint is a requirement of balanced laminate construction, which is intended to reduce or eliminate undesirable stiffness coupling.

In this section, we first address formulations for composite laminates and optimization problems. After that, we describe the genetic algorithm and the procedure of the implementation. The new repair strategy is also described in this section. Finally, the object-oriented programming approach is addressed.

Formulation

The dimensional and non-dimensional in-plane lamination parameters V_i, and V_i* (i = 1–4) of a laminate, respectively, are defined as follows.

$$V_1 = \int_{-h/2}^{h/2} Cos2\theta \, dz \tag{1}$$

$$V_2 = \int_{-h/2}^{h/2} Cos4\theta \, dz \tag{2}$$

$$V_3 = \int_{-h/2}^{h/2} Sin2\theta \, dz \tag{3}$$

$$V_4 = \int_{-h/2}^{h/2} Sin4\theta \, dz \tag{4}$$

$$V_i^* = V_i/h \tag{5}$$

where z is the distance from the laminate mid-plane, $\theta = \theta(z)$ is the fiber orientation angle, and h is the total laminate thickness. The in-plane stiffness matrix A_{ij}^* is calculated by using the in-plane lamination parameters as follows.

$$\begin{aligned} A_{11}^* &= U_1 + U_2V_1^* + U_3V_2^* \\ A_{12}^* &= U_4 - U_3V_2^* \\ A_{22}^* &= U_1 - U_2V_1^* + U_3V_2^* \\ A_{66}^* &= U_5 - U_3V_2^* \\ A_{16}^* &= U_2V_3^*/2 + U_3V_4^* \\ A_{26}^* &= U_2V_3^*/2 - U_3V_4^* \\ A_{ij} &= hA_{ij}^* \end{aligned} \tag{6}$$

where U_i (i = 1–5) are the material invariants as follows:

$$\begin{aligned} U_1 &= \frac{1}{8}(3Q_{11} + 3Q_{22} + 2Q_{12} + 4Q_{66}) \\ U_2 &= \frac{1}{2}(Q_{11} - Q_{22}) \\ U_3 &= \frac{1}{8}(Q_{12} + Q_{22} - 2Q_{12} - 4Q_{66}) \\ U_4 &= \frac{1}{8}(Q_{11} + Q_{22} + 6Q_{12} - 4Q_{66}) \\ U_5 &= \frac{1}{8}(Q_{11} + Q_{22} - 2Q_{12} + 4Q_{66}) \end{aligned} \tag{7}$$

where

$$Q_{11}=\frac{E_1}{1-\nu_{12}\nu_{21}}$$

$$Q_{12}=\frac{\nu_{12}E_2}{1-\nu_{12}\nu_{21}}=\frac{\nu_{21}E_2}{1-\nu_{12}\nu_{21}} \tag{8}$$

$$Q_{22}=\frac{E_2}{1-\nu_{12}\nu_{21}}$$

$$Q_{66}=G_{12}$$

The out-of-plane lamination parameters $W_i{}^*$ are defined as follows.

$$W_1=\int_{-h/2}^{h/2} Cos2\theta z^2 dz \tag{9}$$

$$W_2=\int_{-h/2}^{h/2} Cos4\theta z^2 dz \tag{10}$$

$$W_3=\int_{-h/2}^{h/2} Sin2\theta z^2 dz \tag{11}$$

$$W_4=\int_{-h/2}^{h/2} Sin4\theta z^2 dz \tag{12}$$

$$W_i^*=12W_i/h^3 \tag{13}$$

The out-of-plane stiffness matrix $D^*{}_{ij}$ is calculated by using the out-of-plane lamination parameters as follows.

$$D_{11}^*=U_1+U_2W_1^*+U_3W_2^*$$

$$D^{12^*}=U_4-U_3W_2^*$$

$$D_{22}^*=U_1-U_2W_1^*+U_3W_2^*$$

$$D_{66}^*=U_5-U_3W_2^* \tag{14}$$

$$D_{16}=U_2W_3^*/2-U_3W_4^*$$

$$D_{26}^*=U_2W_3^*/2-U_3W_4^*$$

$$D_{ij}=(h^3/12)D_{ij}^*$$

The objective function to be maximized that measures how close the actual lamination parameters are to the target lamination parameters is defined as follows.

$$f=\frac{1}{\left\{0.01+\sum_{i=1}^{2}\left(\left|V_I^*-Vi^{*0}\right|+\left|W_i^*-W_i^{*0}\right|\right)+\alpha+\beta\right\}} \tag{15}$$

where $V_i{}^{*0}$ and $W_i{}^{*0}$ are the target values of $V_i{}^*$and $W_i{}^*$, respectively. The symbol α in the equation (15) is a penalty that is applied for the presence of unbalanced stacking sequences. In the present study, $\alpha = 0.05$ for an unbalanced laminate, $\alpha = 0$ for a balanced one. A second penalty parameter, β, applies to designs with W_3^{*0} for which D_{16}^* and D_{26}^* bending–twisting coupling is to be minimized. The value 0.01 is added to prevent the denominator becoming 0 for the real optimum cases. How the genetic algorithm maximizes the objective function is shown later.

The parameters V_3^*, V_4^* and W_4^* are zero for symmetric laminates made from only 0°, ± 45° and 90° plies. However, W_3 is not always zero even if the laminate is symmetric and balanced. In some problems such as buckling of compression-loaded plates, the coupling stiffness D_{16}^* and D_{26}^* that are functions of only W_3^*, can be neglected when the non-dimensional values γ and δ defined as follows are small.

$$\gamma=\frac{D_{16}^*}{(D_{11}^{*3}D_{22}^*)^{\frac{1}{4}}},\ \delta=\frac{D_{26}^*}{(D_{11}^{*3}D_{22}^*)^{\frac{1}{4}}} \tag{16}$$

.For moderate aspect ratios, the limit on γ and δ is about 0.2[11], while the limit on γ and δ for long anisotropic plates, these values must be lower than 0.1[12]. In the present work, the limit on γ and δ was taken to be 0.2. However, the approach described here can be used with stricter or less strict limit, as the required for a particular application

After calculating D_{11} and D_{22}, the upper tolerance level for W_3^*, W_3^{*0}, can be calculated from equations (14) and (16), and W_3^* is set to be smaller than or equal to this value. The penalty value of β in equation (15) is applied when W_3^* is larger than W_3^{*0}, and is then equal to

$$\beta=\left|W_3^*-W_3^{*0}\right|\quad W_3^*\geq W_3^{*0} \tag{17}$$

$$\beta=0\quad W_3^*<W_3^{*0}$$

Solution by a genetic algorithm

In order to solve the combinatorial optimization problem described earlier, a genetic algorithm was used, in which each laminate stacking sequence is represented by one chromosome. Each gene in a chromosome corresponds basically to a ply angle. However, the value of the gene does not always equal the angle. Instead, in the present study, a new repair system that is based on the behavior of recessive genes is used to decode chromosome information so as to satisfy the constraints. This approach is illustrated in *Figure 1*.

In order to represent the ply angles as genes, trinary numbers are used with each gene having a value of 0,1 or 2. Basically, the number 0 corresponds to a 0° ply and the number 2 corresponds to a 90° ply. The first (outermost), third, fifth, etc. occurrences of the number 1 correspond to a 45° ply while even-number occurrences correspond to a −45° ply. For example, the chromosome [0/0/1/1/1/2/1] represents the stacking sequence [0/0/45/−45/45/90/−45]s. When the number of occurrences of the number 1 is odd, there is one unbalanced 45° ply in the stacking sequence. Because of laminate symmetry, only half of the plies are represented by the chromosome.

The flow of the genetic algorithm is illustrated in *Figure 2*. The first generation of laminates is selected at random. Each laminate is evaluated by using equation (13), and

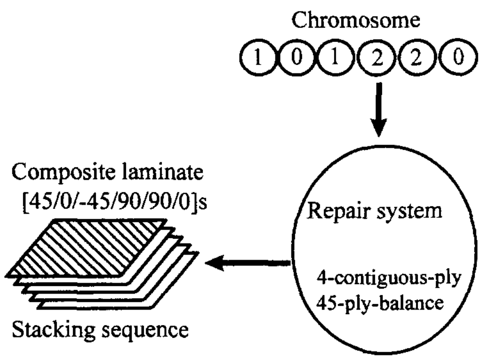

Figure 1 Chromosome decoding with the repair system used in the present study

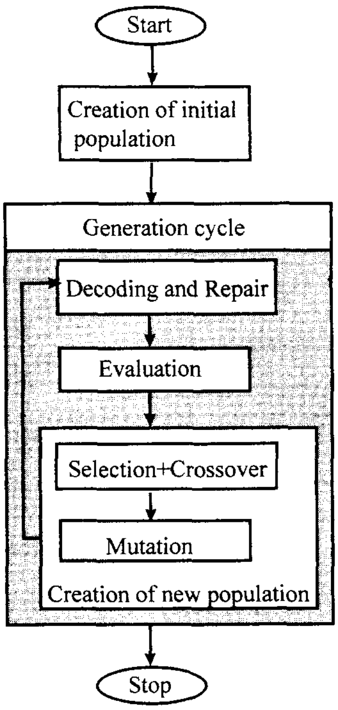

Figure 2 Flow chart of the genetic algorithm

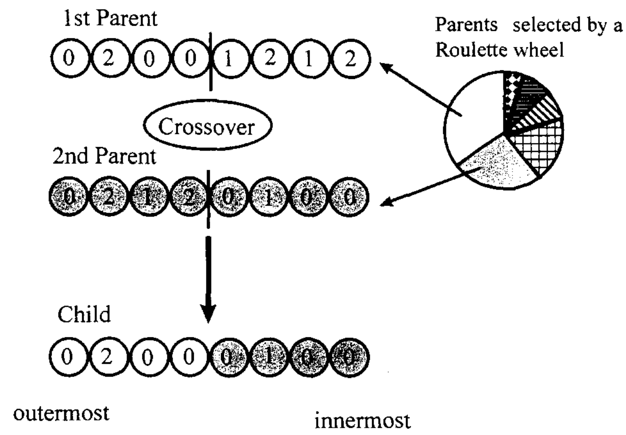

Figure 3 Illustration of the crossover process

elitist strategy is used in which the best laminate is copied into the next generation. A roulette wheel method is adopted for selection of two parents of laminates to create a new laminate for the next generation. Each laminate has area that is proportional to the ratio of its fitness to the sum of the fitness of all laminates on the roulette wheel.

Next, crossover is conducted with a probability value of P_c. When single-point-crossover is performed, the two parent chromosomes are cut at their center points, and the child chromosome is given the same combination of genes as the first parent in its outer half chromosome; it is given the same genes as the second parent in the inner half chromosome. This crossover process is illustrated in *Figure 3*. When crossover is not conducted, the first parent is copied into the next generation. This procedure continues until the number of laminates is equal to the population size. The population size is 10, and the probability of crossover is $P_c = 80\%$ in the present study.

After every individual (laminate) is created, mutation is applied to the laminates, except for the best laminate of the previous generation. In the present study, the probability of mutation P_m is defined for the entire chromosome. Therefore, a probability of mutation of 80% means that, on average, 80% of chromosomes will undergo mutation. The number of mutated genes in a chromosome is optional. In the present study, the number of mutated genes is 2, and the probability of mutation is 80%. For this combination, the probability of mutation of each gene is 18.75% for a chromosome with 8 genes (16 plies). After mutation is completed, the children replace the parents, and the process begins a new unless the number of generations specified was reached.

Repair system

As noted before, gene values of 0, 1 and 2 basically correspond to 0°, 45° or −45°, and 90° plies, respectively. At first, each chromosome is decoded by using the basic correspondence. However, if the laminate stacking sequence resulting from the decode by using basic correspondence violates the 45°-ply-balance rule and the 4-contiguous-ply rule discussed earlier, a repair system is used to enforce these constraints. These rules and how they are enforced by the repair system are described subsequently.

The 4-contiguous-ply-rule. As an example, consider a special case such as [2/2/2/2/2/2/2/2]s which without repair would correspond to a unidirectional 90° laminate. For the repair system used herein, the chromosome is directly translated into ply angles beginning with the outermost ply. When the translation reaches the 5th ply from the outermost ply, that 90°-ply violates the 4-contiguous-ply-rule. Then, repair is applied in which the value of gene is automatically incremented by a value of one before further translation. When the violating gene value is 2, its number is changed to 0. Therefore, the 5th occurrence of 2 is decoded as a 0°-ply. Note that the gene is not changed in the chromosome, but it is decoded as if its value was one higher

For the innermost plies, this repair procedure has to be modified, because when the two innermost plies have the same ply angle, laminate symmetry means that there are already 4 contiguous plies which have the same orientation in the middle of the laminate. Therefore, the repair process is not allowed to stack more than 2 plies in the innermost position within a laminate. When the innermost plies violate this rule, the gene value of the innermost ply is incremented in the same way as described earlier.

By using this decoding and repair procedure, the chromosome [2/2/2/2/2/2/2/2]s is translated into the stacking sequence [90/90/90/90/0/90/90/0]s. Since this repair system occurs only in rare instances, this repair system is implemented at a 100% probability in the present study.

The 45°-ply-balance-rule. When the laminates considered here are unbalanced, there is only one unbalanced 45° ply in the stacking sequence. In the present study, two repair procedures are used to enforce the balanced laminate constraint. The first procedure is a +45°-ply-repair procedure, which attempts to delete one +45° ply and replace it by a 90° or 0 ° ply. The 45° ply position to be replaced by a 90° or 0° ply is selected from the innermost 45° ply. If the 4-contiguous-ply-rule is violated by this replacement, the next 45° ply is selected for replacement. When there is only one 45° ply, or there is no appropriate 45° ply, a −45°-repair procedure is attempted.

In the −45°-repair procedure, a 90° or 0° ply is replaced by a −45° ply. The innermost −45° ply is located, and the adjacent inner or outer ply (inner ply first) is replaced if the 4-contiguous-ply-rule is not violated. If the innermost ply is not appropriate for replacement, the next innermost −45° ply is located. For the situation when there is only one 45° ply in the laminate, the adjacent inner or the outer ply to the 45° ply is selected (inner ply first) to be replaced.

This repair procedure was tested in 10 000 cases beginning with 10-ply laminates (5-gene chromosomes) and ending with 16-ply laminates (8-gene chromosomes). This repair procedure was found to be 100% effective.

If the repair procedure changed the genes instead of just the laminate it could prevent beneficial changes from occurring as a results of two or more consecutive mutations. For example, consider a case when it is beneficial to transform the chromosome [0101] corresponding to the [0/45/0/−45]s laminate into [0202] corresponding the [0/90/0/90]s laminate. When the chromosome [0101] mutates into the chromosome [0102], the repair process reverses the change and the chromosome still corresponds to the [0/45/0/−45]s laminate. However, the repair procedure does not change the chromosome. One additional mutation in a future generation can transform the gene to [0202], and the innermost gene '2' will now be developed into the 90° ply. The innermost gene '2' acts like a recessive gene. Since the repair system of the 45°-ply-balance-rule is applied frequently, it may be applied with a various probability of less than one in the present study to investigate that the repair procedure does not prevent the beneficial changes.

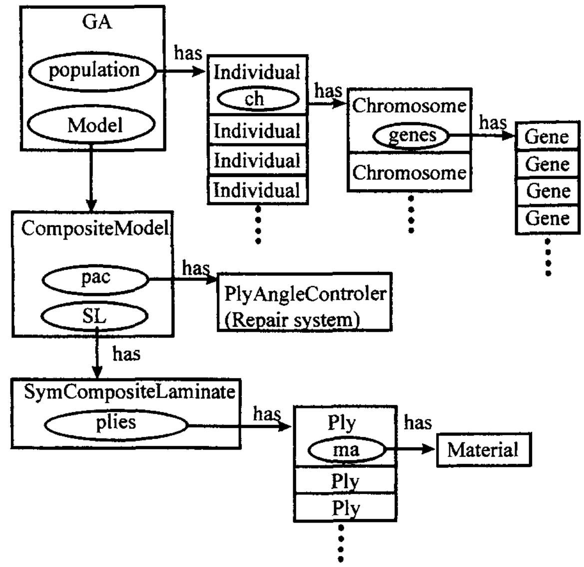

Figure 4 Structure and relations of objects in the computer implementation of the genetic algorithm

Object-oriented programming design

In the present study, the genetic algorithm is implemented in an object-oriented programming environment by using the C++ language. By using the object-oriented design approach, a programming package that has high flexibility to handle changes in design objects is obtained. Relations among all major classes for the present genetic algorithm are shown in *Figure 4*. In this figure, each rectangle shows an instance of each class. In addition, each ellipse inside of a rectangle represents an instance variable which contains an instance of another class or an array of instances of other classes.

A class labeled as GA in *Figure 4* has instances of a class labeled as Individual, that are contained within the instance variable labeled as population, and an instance of the class labeled as Model. The class GA controls all optimization procedures. Each 'individual' has chromosomes and each chromosome has genes. Since these chromosomes and genes operate in dynamically obtained memory, there are no upper bounds on the length of the chromosomes or the total number of chromosomes if there is enough memory. Each 'individual' is evaluated by message passing between an instance of class Individual and an instance of class Model. A user can easily change design objects by changing only the class Model. The class Model is an abstract class, and therefore, users only have to implement each model class as a derived class from Model. In the present study, CompositeModel is a derived class from the abstract class Model.

All procedures of composite laminate structural analyses are encapsulated in a class CompositeLaminate and a SymCompositeLaminate. The class SymCompositeLaminate is a class derived from the class CompositeLaminate and has a symmetric stacking sequence. Lamination parameters are calculated in the class SymCompositeLaminate.

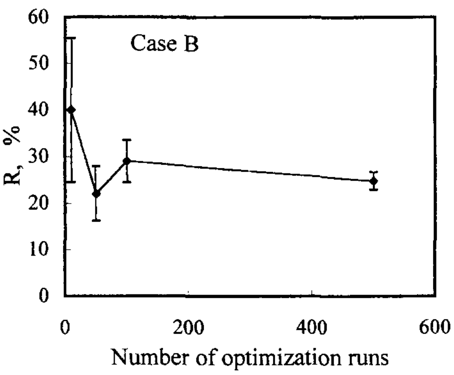

Figure 5 Reliability and its standard deviation for Case B

RESULTS AND DISCUSSION

Description of examples

In order to investigate the efficiency of the repair system, two cases of target lamination parameters for 16-ply laminates (8 genes) were used as examples in the present study. The stacking sequence that matches the target lamination parameters of the first case (Case A: $V_1^{*0} = 1$, $V_2^{*0} = 1$, $W_1^{*0} = 1$, $W_2^{*0} = 1$) is [0/0/0/0/0/0/0/0]s. However, because of the 4-contiguous-ply-rule, the best match is [0/0/0/0/90/0/0/90]s with $V_1^{*0} = 0.5$, $V_2^{*0} = 1$, $W_1^{*0} = 0.852$, $W_2^{*0} = 1$, which does not include angle plies. The stacking sequence that matches the target lamination parameters of the second case (Case B: $V_1^{*0} = 0$, $V_2^{*0} = 0.5$, $W_1^{*0} = 0.3398$, $W_2^{*0} = 0.828$) is [0/90/0/0/45/90/−45/90]s, which includes a pair of angle plies. For both cases, the target values of W_3^{*0} were set to 0. For case A, W_3^* are exactly 0, and for case B, $W_3^* = 0.058$, which is very small. Each target case was optimized with a population size of 10 and is run over 50 generations.

Since genetic algorithms use random numbers, the result of a single optimization can be a matter of chance. Therefore, in order to evaluate the probability of reaching the optimum, the reliability R is estimated by performing a large number of optimizations. Consider the case in which N optimization runs are conducted, and the number of runs reaching the global optimum is N_c. Then, the reliability R_N is estimated as follows.

$$R_N = N_c/N \tag{18}$$

The standard deviation of R_N is $\sigma(R_N)$ and is obtained as follows.

$$\sigma(R_N) = \sqrt{\frac{(1 - R_N)R_N}{N}} \tag{19}$$

In the present study, case B with 0% probability of repair was selected to estimate the number of optimizations needed. As previously described, the population size was 10, and the parameters of the genetic algorithm were set to $P_c = 0.8$ (probability of crossover) and $P_m = 0.8$ (probability of mutation). The standard deviations of the estimates of reliability versus the number of runs were calculated by using equation (17) and are shown in *Figure 5*. In the figure, the ordinate is the reliability of design, and the abscissa is the number of optimization of runs. For investigation of the standard deviation, three sets of runs were conducted at each parameter setting. The solid line in the figure is the average value of three sets of each run. The results show that in order to get a reliability with a 2% standard deviation, 500 runs are needed. Thus, reliability of design is defined herein by 500 optimization runs.

It is important to note that a very large number of runs are needed in order to accurately assess the reliability of the genetic algorithm. Good accuracy is needed in order to be able to assess the effects of various strategies such as repair schemes. Consider, for example, what happens if 50 runs are used instead of 500 runs to calculate the reliability. From equation (17), the standard deviation of the reliability estimate would then be about 5.6%. So, if with one repair scheme the reliability is 15% and with another it is 25%, the difference is within 5.6% for each value. Therefore, there is a significant probability that the difference is a matter of chance rather than a matter of strategy.

Entire design space calculation

In order to investigate the character of the design space, all possible combinations of stacking sequences were analyzed for both cases. The fitnesses of all possible

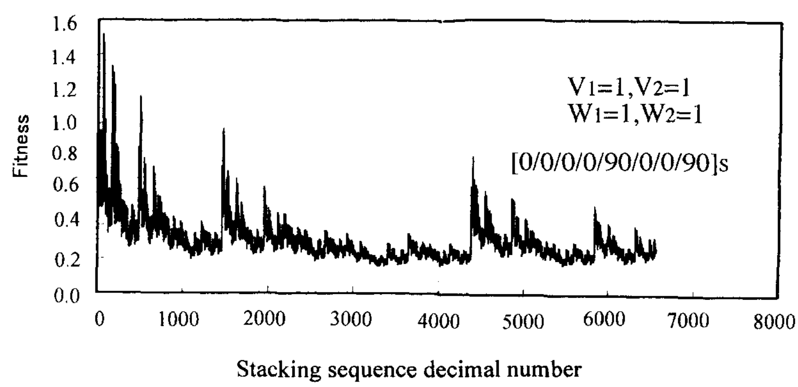

Figure 6 Entire design space of Case A

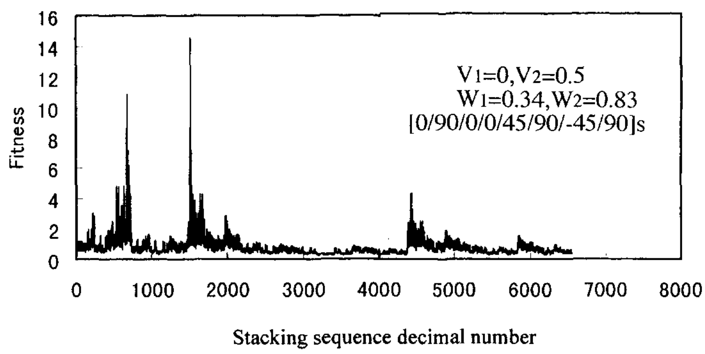

Figure 7 Entire design space of Case B

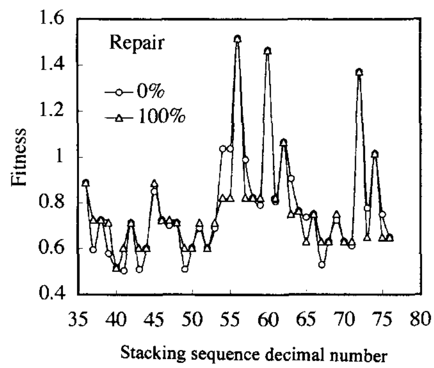

Figure 8 Design space near the optimum stacking sequence for Case A

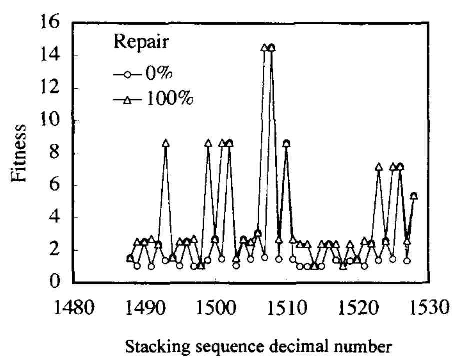

Figure 9 Design space near the optimum stacking sequence for Case B

chromosomes of cases A and B are shown in *Figures 6* and *7*, respectively. The chromosomes were decoded with a probability of repair (the 45-ply-balance-rule) of 0% (the 4-contiguous-ply repair is applied at 100% probability as mentioned previously). The ordinate shows the values of objective function, and the abscissa is the chromosome expressed as a decimal number.

As described previously, genes are represented by trinary numbers so that the entire chromosome can be represented by a trinary number. For convenience, these trinary numbers are expressed as decimal numbers in *Figures 6* and *7*. For example, the decimal number 1508 corresponds to the trinary number [02001212] which corresponds to the stacking sequence [0/90/0/0/45/90/−45/90]s. For 16-ply laminates (8-gene chromosome), all possible combinations of stacking sequences are described by decimal numbers between 0 and $3^8 - 1 = 6560$. Consecutive numbers usually have almost the same stacking sequences, with the difference being the innermost ply.

Figures 6 and *7* show that the design spaces for both problems have many local maxima. For case A, the decimal number 0 corresponds to the stacking sequence [0/0/0/0/45/0/0/−45]s, because of the 4-contiguous-ply-rule, with a fitness value of 0.950. The decimal number that corresponds to the maximum fitness design is 56. The number 56 corresponds to the stacking sequence [0/0/0/0/90/0/0/90]s, with a fitness equal to 1.519. For case B, the best laminate corresponds to the decimal number 1508, the stacking sequence [0/90/0/0/45/90/−45/90]s, and a fitness of 14.543.

In order to see the effect of repair, the fitness near the optima with the repair system (100% probability) and without repair system are shown in *Figures 8* and *9* for case A and case B, respectively. For case A (*Figure 8*), the differences between the fitnesses with the repair system and without the repair system are small. However, for case B (*Figure 9*), the repair substantially increases the fitness of many chromosomes. Now, two numbers correspond to the optimum. Without repair, the number 1508 correspond to the optimal stacking sequence. The number 1507 also corresponds to the optimal laminate because of the operation caused by the +45°-repair system. Therefore, the repair system increases the number of chromosomes corresponding to the optimal laminate for case B. However, as shown in *Figure 9*, many local maxima become much higher than their neighbors and may possibly leading to difficulties in moving from these local optimum. For this reason, the effect of probability of repair on the reliability of the algorithm was investigated.

Effect of probability of repair

The effect of probability of repair on the reliability for cases A and case B is shown in *Figures 10* and *11*, respectively. The ordinate is the reliability of the genetic algorithm obtained from 500 optimization runs. The abscissa is the probability of repair P_r (probability of repair of the 45°-ply-balance-rule).

For case A (*Figure 10*), the reliability is almost constant

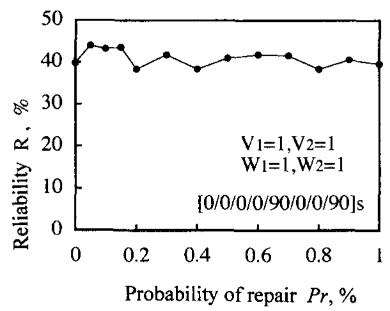

Figure 10 Variation of the reliability for Case A with the probability of repair

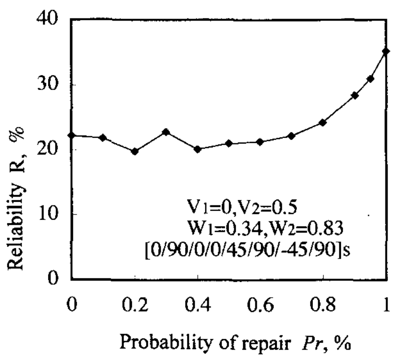

Figure 11 Variation of the reliability for Case B with the probability of repair

at about 41% except for the region around $P_r = 0.1$. In the region around $P_r = 0.1$, the reliability increases up to 43%, but the increased value is still within a standard deviation of 2%. At $P_r = 100\%$, the reliability still remains at the same value as that of $P_r = 0\%$. This negligible effect of repair for case A is consistent with the small effects on fitness revealed in *Figure 8*. Since case A does not have any 45° plies in its optimal stacking sequence, the 45°-balance repair has no effect on the reliability.

For case B (*Figure 11*), the repair system clearly increases the reliability of the genetic algorithm. For $P_r = 0\%$, the reliability is 22.2%, while for $P_r = 100\%$ the reliability is 36.2%. For case B, the optimal laminate is [0/90/0/0/45/90/−45/90]s and has a pair of angle plies so that the repair strategy matters. Usually, composite laminates include some angle plies. Therefore, a 100% probability of repair is expected to improve the reliability more than the low probability of repair for most laminates. Even if the optimum laminate does not include angle plies, using repair at a probability of 100% still gives the same reliability as no repair. The advantage shown in the present work for a 100% probability of repair contradicts the general recommendation to apply repair at a low probability[9]. This contradiction in probability in probably caused by the fact that the present repair system does not change the chromosome as most repair systems do.

Though the reliability is low even when the probability of repair is 100%, the reliability can be raised by several ways. For example, if the population size is doubled from the previous size of 10, and the maximum number of generations is also doubled to 100, the reliability increases from 36.2% to 61%. This approach increases the cost of a single run. However, performing 4 runs of the genetic algorithm under the previous conditions (population size 10, maximum number of generations 50) gives a reliability of 83.4%, which is higher than 61% and has the same calculation cost. Therefore, in this case one should use several runs instead of increasing population size or maximum number of generations. In order to increase the reliability earlier 90%, with a base reliability of 36.2%, at least 6 optimizations runs are needed.

CONCLUDING REMARKS

A genetic algorithm with a new repair system that incorporates benefits of recessive genes was applied to the problem of obtaining the stacking sequence with lamination parameters that are closest to a set of target lamination parameters. The repair system was used to enforce two constraints. In particular, a 4-contiguous-ply-rule repair was used at a probability of 100%, and a 45°-ply-balance-rule was used at various levels of probability. The repair system was tested by using ten thousands cases and shown to be 100% effective. The relations between the genetic algorithm reliability and the probability of repair were investigated for two target laminates. The probability of repair did not have much of an effect on the genetic algorithm reliability in the case that the optimum laminate did not include 45° plies. However, the GA reliability increased to 36.2% (100% of probability of repair) from 22.2% (0% of probability of repair) for the case which had 45° plies. The effectiveness of repair at a high level of probability contradicts the recommended use of low probability with general repair systems. The reason for this contradiction may be that this repair system alters the decoding rules of chromosomes but does not change genes. From these results, the repair system in the present study is concluded to be effective for repair of balance and excess of identical contiguous plies at a 100% probability of repair.

ACKNOWLEDGEMENTS

This work was supported in part by NASA grant NAG1-1669.

REFERENCES

1. Miki, M., Design of laminated fibrous composite plates with required flexural stiffness. *ASTM STP*, 1985, **864**, 387–400.
2. Haftka, R. T. and Walsh, J. L., Stacking-sequence optimization for buckling of laminated plates by integer programming. *AIAA Journal*, 1992, **30**, 814.
3. Le Riche, R. and Haftka, R. T., Optimization of laminate stacking sequence for buckling load maximization by genetic algorithm. *AIAA Journal*, 1993, **31**, 951.
4. Le Riche, R. and Haftka, R. T., Improved genetic algorithm for minimum thickness composite laminate design. *Composites Engineering*, 1995, **5**, 143.

5. Harrison, P. N., Le Riche, R. and Haftka, R. T., Design of stiffened composite panels by genetic algorithm and response surface approximations. AIAA-95-1163OP, 1995, **58**.
6. Nagendra, S., Jestin, D., Gürdal, Z., Haftka, R. T. and Watson, L. T., Improved genetic algorithm for the design of stiffened composite panels. *Composites Structure*, 1995, **58**, 543.
7. Kogiso, N., Watson, L. T., Gürdal, Z. and Haftka, R. T., Genetic algorithms with local improvement for composite laminate design. *Structural Optimization*, 1994, **7**, 207.
8. 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*, 1994, **1**, 95.
9. Yamazaki, K., Two-Level optimization technique of composite laminate panels by genetic algorithms. AIAA-96-1539-CP, 1996, 1882.
10. Orvosh, D. and Davis, L., Shall we repair? genetic algorithms, combinatorial optimization, and feasibility constraints, *Proceedings of the fifth International Conference on Genetic Algorithms*, Morgan Kaufmann Publishers, 17–21, 1993, 650.
11. Nemeth, M. P., Importance of anisotropy on buckling of compression-loaded symmetric composite plates. *AIAA Journal*, 1986, **24**, 1831.
12. Nemeth, M. P., Buckling behavior of long anisotropic plates subjected to combined loads. NASA, TP3568, 1995.