USE OF GENETIC ALGORITHMS FOR OPTIMAL DESIGN OF LAMINATED COMPOSITE SANDWICH PANELS WITH BENDING-TWISTING COUPLING

B. Malott and R.C. Averill
Department of Materials Science and Mechanics
and
E.D. Goodman, Y. Ding, and W.F. Punch
Department of Computer Science
Michigan State University
East Lansing, MI 48824-1226

USE OF GENETIC ALGORITHMS FOR OPTIMAL DESIGN OF LAMINATED COMPOSITE SANDWICH PANELS WITH BENDING-TWISTING COUPLING

B. Malott and R.C. Averill
Department of Materials Science and Mechanics
and
E.D. Goodman, Y. Ding, and W.F. Punch
Department of Computer Science
Michigan State University
East Lansing, MI 48824-1226

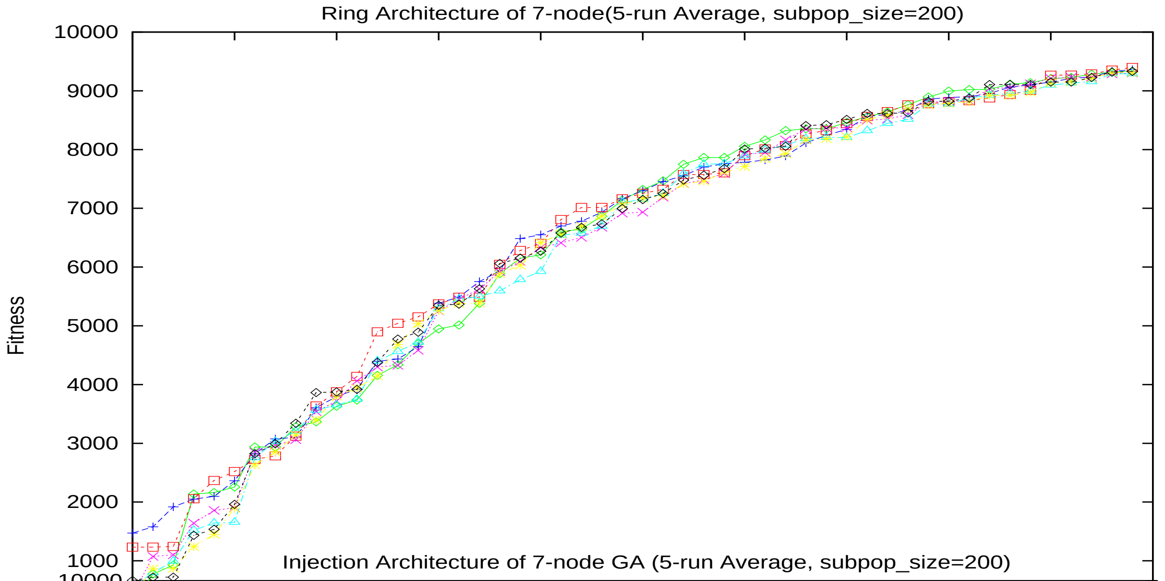


Figure 6. Results from seven-node ring architecture.

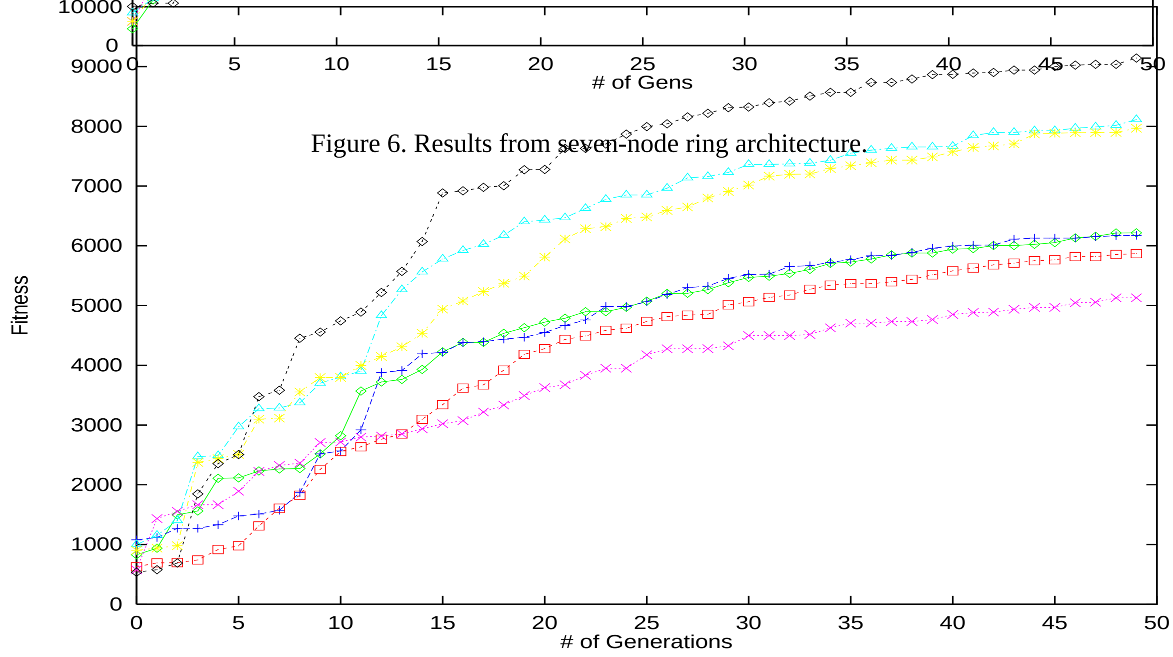


Figure 7. Results from seven-node island injection architecture.

38, pp. 13-28.

[7] Punch, W.F., Averill, R.C., Goodman, E.D., Lin, S.-C., Ding, Y., and Yip, Y.C., (1994) Optimal Design of Laminated Composite Structures Using Course-Grain Parallel Genetic Algorithms, *Computing Systems in Engineering*, Vol. 5, pp. 415-423.

[8] Nagendra, S., Haftka, R.T., Gurdal, Z., (1993) Design of a Blade Stiffened Composite Panel by a Genetic Algorithm, *Proceedings of the AIAA/ASME Structures, Structural Dynamics, and Materials Conference*, Part 4, pp. 2418-2436.

[9] Johnson, E.A., (1994) The Implementation of Continuously Updated Sharing in The Simple Genetic Algorithm Code, and it's Application to the Optimal Placement of Elastic Supports on a Simply Supported Plate, *Proceedings of the AIAA/ASME/ASCE/AHS Structures, Structural Dynamics, and Materials Conference*, Part 4, pp. 2276-2286.

[10] Leung, M., Nevill, G.E., (1994) Genetic Algorithms for Preliminary 2-D Structural Design, *Proceedings of the AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference* Part 4, pp. 2287-2291.

[11] Sandgren, E., Jensen, E., Welton, J.W., (1990) Topological Design of Structural Components Using Genetic Optimization Methods, presented at The Winter Annual Meeting of The American Society of Mechanical Engineers, Dallas TX.

[12] Furura, H., Haftka, R.T., (1993) Locating Actuators for Vibration Suppression on Space Trusses by Genetic Algorithms, *Structures & Controls Optimization*, ASME, Aerospace Division AD Vol. 38, pp. 1-11.

[13] Yang, P.C., Norris, C.H., and Stavsky, Y., (1966), Elastic Wave Propagation in Heterogeneous Plates, *International Journal of Solids*, Vol. 2, pp 665-684.

[14] Raymer, D. P., (1992) *Aircraft Design: A Conceptual Approach*", AIAA Education Series, Published by AIAA Inc., Washington D.C

[15] Gibson, L. J., and Ashby, M. F., (1988) Cellular Solids, Structure and Properties, Pergamon Press, New York.

[16] Whitney, J.M., (1973) Shear Correction Factors for Orthotropic Laminates Under Static Load, *Journal of Applied Mechanics*, Vol. 40, pp. 302-304, 1973.

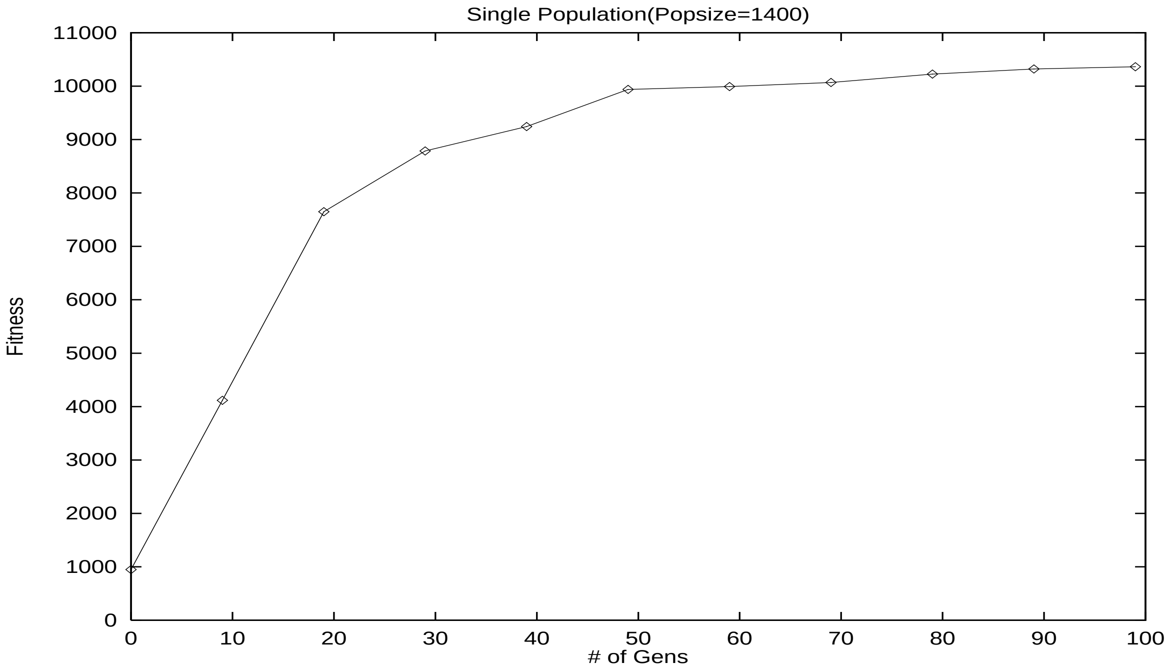


Figure 5. Results from single node architecture.

mance due to a more refined search space. Finally, the most refined subpopulation(s) should receive good building blocks from the moderate-resolution level, and should achieve the best solutions. However, in the small search space represented here this behavior is compressed into about the first 10 generations. By generation 3, after one generation of “processing” the first migrants from the lowest level, the middle level subpopulations (triangles and asterisks on the graph) have exceeded their performance. Then at about 6 generations, the highest refinement subpopulations (represented by diamonds and dotted line) dominates and continues to outperform the others for the remainder of the run. This run, while outperforming a set of 7 subpopulations at the highest level of refinement using identical parameters and population sizes in a ring topology, reached the same optimum, if allowed to continue to around generation 100. One characteristic of the injection architecture is an increased rate of convergence, especially early in the design process. For example, note in Figures 5-7 that the iiGA produces a design with fitness of 7000 within about 15 generations, while the single node and ring approaches require at least 20 generations to attain a design with the same level of fitness. Another advantage of the injection architecture is also evident: once the low-resolution subpopulations have converged (which happens relatively quickly), they can be either “frozen”, avoiding additional function evaluations, or perhaps partially reinitialized, broadening the search.

Table 1 compares the weight, twist (Wb - Wa), and minimum in-plane stiffness (Amin) values of the baseline case and two typical optimum result cases determined by the GA. Case 1 has a lamination sequence of [-39.375 / -39.375 / -84.375 / -45 /CORE / -16.875 / 87.188 / -2.813 / -75.938 / -5.625 / -87.188 / -22.5 / -81.563 / -28.125 / -14.063 / 87.188 / -78.75 / -30.938 / -61.875 / -19.688 / -28.125]. Case 2 layup is [0 / -36.563 / -81.563 / 75.938 / -25.313 / -22.5 / -87.188 / -8.438 / -19.688 / -16.875 / 84.375 / 2.813 / -22.5 / -87.188 / -84.375 / CORE / -36.563 / -36.563 / -36.563 / -87.188]. The baseline layup was [0 / 90 / 45 / -45]$_s$ for both top and bottom face sheets and the core orientation was zero degrees.

As Table 1 indicates, the designs of case 1 and case 2 are a great improvement over the baseline case. The twist changes sign meaning “opposite” twist is indeed achievable, and the GA approach is a viable means of finding designs to produce this type of response. In approximately 50 generations (35,000 evaluations), the GA is able to achieve results which typically reduce the weight to 66% of the baseline value, increase the minimum in-plane stiffness value by 33%, change the direction of the twist, Wb-Wa, and increase its magnitude 77 times (from -7.3e-6 (baseline) to +5.6e-4 (GA)). These results are from preliminary analyses. In the future, strength constraints, as well as other constraints, will be included in the fitness function.

Table 1:

	weight	(Wb - Wa)	Amin
Baseline	11.051	-7.313e-6	0.122e+9
Case 1	7.330	+5.588e-4	0.1592e+9
Case 2	7.330	+5.570e-4	0.1653e+9

References

[1] Goldberg, D.E., (1989) *Genetic Algorithms in Search, Optimization, and Machine Learning,* Addison-Wesley Publishing Company Inc., New York.

[2] Nagendra, S., Haftka, R.T., and Gurdal, Z., (1992) Stacking Sequence Optimization of Simply Supported Laminates With Stability and Strain Constraints, *AIAA Journal*, Vol. 30, pp 2132-2137.

[3] Le Riche, T., and Haftka, R.T., (1993) Optimization of Laminate Stacking Sequence for Buckling Load Maximization by Genetic Algorithm, *AIAA Journal*, Vol. 31.

[4] Kogiso, N., Watson, L.T., Gurdal, Z.,Haftka, R.T., and Nagendra, S., (1994) Minimum Thickness Design of Composite Laminates Subject to Buckling and Strength Constraints By Genetic ALgorithms, *Proceedings of the AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference,* Vol. 4, pp. 2257-2275.

[5] Le Riche, T., and Haftka, R.T., (1994) Improved Genetic Algorithms for Minimum Thickness Composite Laminate Design, *Mechanics of Composite Materials and Structures*.

[6] Kogiso, N., Watson, L.T., Gurdal, Z., and Haftka, R.T., (1993) Genetic Algorithms with Local Improvement for Composite Laminate Design, *Structures & Controls Optimization,* ASME, Aerospace Division, AD Vol.

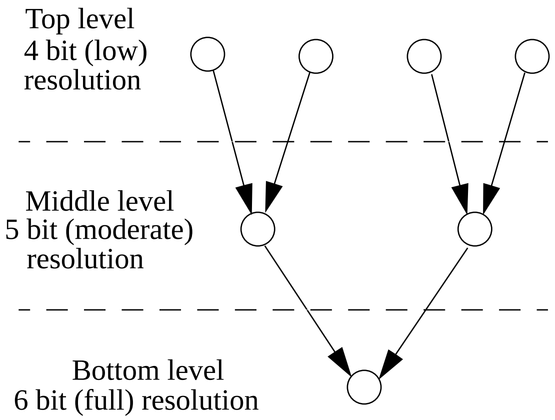


Figure 4. Island Injection Topology with 7 subpopulations.

$$fitness = cc\left(\frac{Amin}{aa}\right)\left(-\frac{Wb - Wa}{tw} - 10\frac{weight}{ww}\right) \quad (1)$$

Where *Wb* and *Wa* are transverse corner deflections, *tw* is the twist *(Wb-Wa)* for the reference case, *ww* is the reference case weight, *weight* is the weight of the plate design being evaluated, *Amin* is the minimum in-plane stiffness of the design being evaluated, *aa is* the reference minimum in-plane stiffness, and *cc* is a ply clustering penalty parameter.

5. Results

The results presented compare three different GA topologies. These are a single node with a population of 1,400, a ring topology with 7 subpopulations having population size of 200 each, and an island injection topology containing 7 subpopulations, each with a population size of 200. In all three cases, a total population size of 1,400 is used.

Figure 5 contains results obtained from use of a single population of size 1400. These results indicate the fitness begins to converge after about 50 generations. Figure 6 shows typical results using the ring architecture of Figure 2a. Results from this architecture show each node achieving approximately the same performance after relatively few generations and convergence around 50 generations.

Figure 4 shows the island injection architecture used in this study, and typical results are shown in Figure 7. GALOPPS used a two point crossover rate of 0.5, mutation rate of 0.001 per bit, a crowding factor of 3, incest reduction (a form of mate selection) of 3, and tournament selection. The crowding factor and incest reduction were used to help maintain population diversity. Core thickness and angular resolution for the plies in the four top level (lowest resolution) subpopulations is 4 bits, 5 bits for the two middle subpopulations, and 6 bits for the final subpopulation (only one subpopulation at this level was used).

While Figure 2 shows a typical iiGA configuration, a simpler 7-subpopulation tree was used for the preliminary testing (see Figure 4) because of computational resource limitations. In this configuration, the bottom level (highest resolution) contains only one subpopulation into which both moderate level subpopulations inject migrants. Due to the dimensionality of the problem being relatively small, both the iiGA and the ring architecture approaches were able to find a near-optimal solution within approximately 100 generations. Thus additional complexity must be added to the problem to allow conclusive comparison of the different GA architectures. However, based on the results to date, certain observations can still be made. Figure 7 depicts typical results of the iiGA, in which migration occurred after each set of 3 generations. The seven curves represent each subpopulation. The behavior sought is that the lower-refinement subpopulations, which are working in a smaller search space, should initially make more rapid progress, providing good "building blocks" to the moderate-refinement subpopulations, which should exceed their perfor-

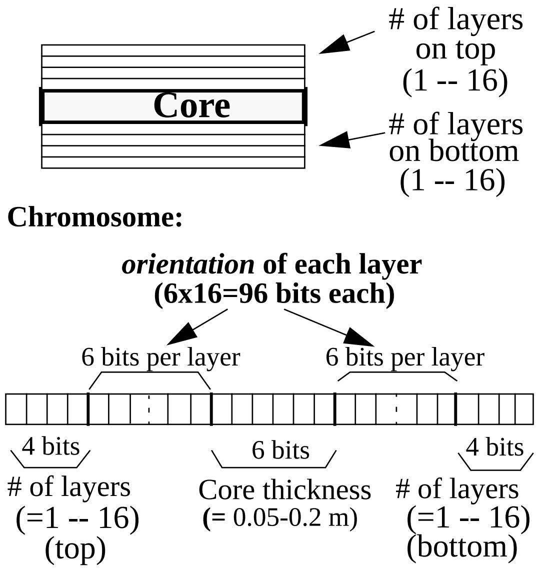


Figure 3. Chromosome representing a design of total length 206 bits.

more focused search. The leaf nodes are autonomous, so their search is not "colored" by the influence of various local optima discovered in other nodes on the way to the global optimum.

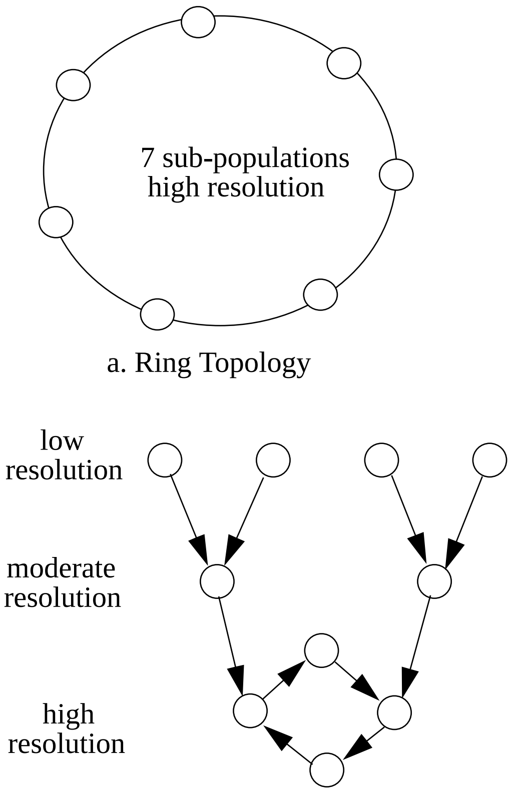
In previous work, we applied the iiGA approach to the composite material beam problem as follows. We created "chunkier" representations by representing multiple design elements of the original 480 elements as a single GA entry. Thus we can speak of a 1x2 chunk, which represents two elements of a row of the design as a single GA element, or a 4x4 chunk which represent a 4-row, 4-column set of elements as a single GA entry. The analysis time for the less refined representations is smaller, reducing execution time. The 1x1 chunks comprise the original 480-bit representation, and it is into this representation that each of the other representations ultimately injects its best solution.

The result of the iiGA approach to the composite beam problem in [7] showed dramatic improvement, both in the energy absorbing characteristics of the beams and the computational time required to find these solutions.

Extending the iiGA Approach to Composite Plates

The lessons learned from using iiGAs for the beam design in [7] are extended to the more complicated domain of composite sandwich panel design in the current study.

Binary coding of the design variables is used (see Figure 3). Each design is represented by a 206 bit string of binary numbers called a chromosome. This gives a design space of 2^{206}. The coded design string contains information about the number of plies in the top and bottom face sheets, as represented by the first and last 4 bits in the string, respectively. The next 6 bits inward contain information about the orientation of each ply, and the middle 6 bits give the core thickness. The fitness function determines the "goodness" of a given design based, in part, on a weighted combination of the amount of twist at the plate tip and overall weight:



a. Ring Topology

b. Typical Island Injection Topology

Figure 2. Comparison of levels of refinement and paths for migration among subpopulations in a parallel genetic algorithm.

engine. GALOPPS stands for “Genetic Algorithm Optimized for Portability and Parallelism System”, and was developed by E. D. Goodman. In the current study a new exchange topology called injection island parallelism is utilized.

Island Injection Architecture

In previous work with parallel genetic algorithms for design, a number of exchange topologies were examined, attempting to discover which topologies would be most appropriate for the laminated composite beam example [7]. In so doing, a new topology we call injection island parallelism, or iiGAs, was created. The two most interesting aspects of an iiGA are its migration rules and the heterogeneous nature of its populations.

In most GA parallel processing approaches, multiple subpopulations are used to generate solutions, and these solutions are exchanged between subpopulations in hopes of “bootstrapping” all the subpopulations to better solutions. We can note two properties in most such approaches. First, all of the subpopulations use the same GA representation of the problem. Second, the exchanges are typically two-way; that is, each subpopulation gives and receives solutions from other subpopulations.

Our iiGA follows neither of these practices. First, the subpopulations are organized in a hierarchical exchange which is one-way and treelike; that is, solutions are passed from children to parent subpopulations, but no exchanges occur between subpopulations on the same level, except perhaps at the lowest level; and the “leaf” subpopulations receive no solutions at all (See Figure 2). Second, we allow various subpopulations to work with different representations of the same problem. The form of the representation typically follows the hierarchical exchange topology, with “child” subpopulations using a “chunkier” or more abstract representation, which is then passed to a more detailed representation in the parent for refinement. This allows subpopulations to search smaller, but more abstract, search spaces for areas of possible interest, which can then be refined by parent subpopulations using a more detailed and

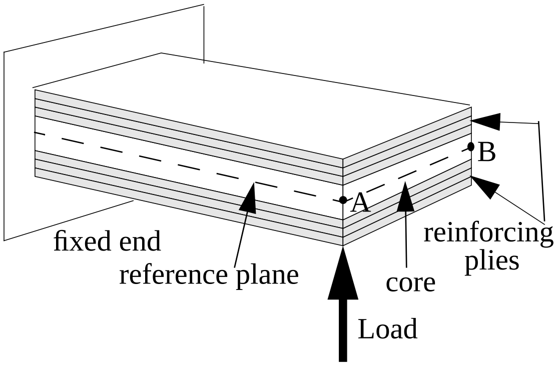


Figure 1. Schematic of the cantilever sandwich panel and loading conditions.

goal will be to achieve a specified amount of opposite wing twist for a given loading condition, rather than to maximize it.

3. Model Description

The model used is a rectangular cantilever sandwich panel, intended to represent an idealized aircraft wing (see Figure 1). For this demonstration problem, no attempt was made to model exactly actual aircraft wing geometries or loading. The sandwich panel is comprised of graphite/epoxy reinforcing plies with a core exhibiting properties consistent with the theory of Gibson and Ashby [15] for honeycomb materials, and having a thickness range from 0.05 to 0.2 meters. All reinforcement plies are of fixed thickness, 0.127mm. The number of plies in the top and bottom skins is left as a design variable, with the maximum allowable number being 16 and the minimum number being 4 (later to be replaced by a strength constraint). The plate aspect ratio (length to width ratio) was chosen to be 8, with the plate being 8 meters long by 1 meter wide.

The objective of the current work is to determine the optimum lamination scheme for the sandwich wing structure, illustrated in Figure 1. This plate represents a rectangular wing with zero sweep back or taper. The plate is subjected to a unit point load at the free end of the leading edge to simulate actual aerodynamic loading situations which cause the wing to bend and twist. By tailoring the stacking sequence of both the top and bottom skins, the twisting caused by such a loading can be reduced and perhaps reversed, yielding a more advantageous aerodynamic position for certain maneuvers. As previously stated, the goal of this study is to determine the stacking sequence (ply orientations and number of plies) and core thickness which minimizes overall weight while maximizing twist in the direction opposite that caused by the loading. The latter objective is achieved by maximizing the algebraic difference between the transverse deflection at the two free corners, i.e. W_B-W_A. A penalty parameter is used to ensure that the in-plane stiffness of the laminate is not compromised while achieving the other design goals. In the fitness function, overall weight, inplane stiffness, and twist response are normalized by the corresponding values for a reference, or baseline, design. The reference design is a sandwich panel with core thickness 0.125m and having face sheets with $[0/90/+45/-45]_s$ layups. This layup provides a quasi-isotropic material in the plane of the wing. In addition to stiffness constraints, a ply clustering constraint is included to help alleviate matrix cracking which occurs for designs having contiguous plies with the same orientation. This constraint penalizes the fitness of designs having two or more adjacent plies with the same fiber orientation. Strength constraints will be added in later investigations.

4. Analysis Technique

Displacements and stresses in the plate are determined by a finite element model based on first order shear deformation plate theory [13]. The finite element code internally calculates shear correction factors for each layup using the approach of Whitney [16]. An “optimal” finite element mesh which yields results converged to within 5% while allowing maximum computational efficiency is used. Computational efficiency is very important in GA design due to the large number of evaluations required. The finite element code is used as a fitness function evaluator in conjunction with the new general-purpose Genetic Algorithm GALOPPS v2.37 which serves as the function optimization use of well-established goals, decompositions, constraints and structures. Routine design typically involves the setting of design parameters in well-known design prototypes to meet a specific situation. Innovative design is more difficult, using known prototypes or structures and constraints, but in innovative ways to achieve new goals. Popular examples of this kind of design are “MacGuyver” or Rube Goldberg devices, where common objects are used in innovative ways to achieve design goals. Creative design is yet more difficult, requiring new prototypes, constraints and goals to achieve the design. GA design is particularly well-suited to innovative or creative design. GAs can be viewed as a design search technique that is influenced only by the representation of the design and the performance of the design. A GA designer can thus search large search spaces, typically unavailable to heuristic search techniques, discovering new designs that were previously unknown, without depending on linearization, existence of derivatives of the objective function, etc.

Haftka et al. [2-6] have used GAs for determining an optimum laminate stacking sequence which minimizes laminate thickness or weight subject to strength, buckling, and ply contiguity constraints. In these studies, balanced, symmetric, simply supported laminates comprised of 0, +/-45, and 90 degree plies of equal thickness were subject to in-plane biaxial loading. To improve efficiency of the algorithm Haftka et al. [3], [4], [6] utilized a genetic algorithm with memory, storing useful information about past designs in a *binary tree*. Use of binary tree storage was shown to reduce the number of analyses required by 30-40%. Another attempt at improved efficiency was a procedure called local improvement which utilized information stored in the binary tree [4,6]. In reference [5], the basic genetic operators were modified to improve performance.

GAs are gaining popularity and becoming more frequently used in a variety of optimization problems. Applications include minimizing the weight of a stiffened composite laminate with a hole [8], optimizing elastic support locations on a square plate to maximize the first system eigenvalues [9], optimizing a simple truss structure capable of supporting given forces [10], optimizing the cross-section and minimizing the weight of an automobile bumper and minimizing the weight of an automobile body panel with stress and deflection constraints [11], and optimizing the placement of actuators on large space structures [12].

Previously work has been done by the authors on design of laminated composite beams [7]. The objective was to design the beam for maximum energy absorption, such as is required for tank armor or automobile bumpers. This study reported work designing a 24-layer beam made of graphite-epoxy composite layers with clamped-clamped end conditions and an applied point load at midspan. A thin layer (about 5.5% of the nominal ply thickness) is placed at the top of each composite layer, so there are actually 48 layers in the model. Each thin layer may be assigned the same material properties as the layer immediately below it, or it may be assumed that the thin layer is compliant, with stiffness properties three orders of magnitude less than the composite layers. The length-to-thickness ratio of the beam is 50. The length of the beam is divided into 20 sublengths (finite elements), so there are 20x48 = 960 design elements. The GA had to decide whether to place a 0 degree ply or a 90 degree ply in each of the 480 structural ply design elements, and whether or not to place a compliant material in each of the 480 thin layer design elements.

In that application, we used a number of representations but focused on a 480-bit-string version which fully represents one half of the beam, and which was then mirrored across the vertical midline to create a full beam. Thus the search space examined by the GA was 2^{480}, representing all possible combinations of material and compliant material. The results of this work were quite promising, and the GA generated a number of unique designs.

In aerospace structures, the use of laminated composite materials offers many advantages over conventional materials such as aluminum. One particular advantage is the ability to tailor the lamination scheme (or ply stacking sequence) to achieve a desired structural response to a given loading situation. For example, during certain maneuvers, an aircraft wing may be subjected to aerodynamic loadings which cause the wing to twist in an aerodynamically undesirable manner. By taking advantage of the stretching-bending-twisting coupling of laminated composite materials, more desirable responses to such loads are possible. In the current study genetic algorithms are utilized for optimization of a cantilever sandwich plate (an idealization of an airfoil) layup.

2. Problem Description

During flight, aircraft wings are subjected to aerodynamic loading which causes bending and twisting to occur. The twisting load is due to a pressure differential across the airfoil in which pressure is greater at the leading edge than the trailing edge. Many aircraft are designed and constructed with zero to five degrees of initial wing twist. Wing twist is built into an aircraft to reshape the spanwise lift distribution to approximate an ellipse and to prevent tip stall [14]. At a given lift coefficient the lift distribution can be optimized by correct choice of initial wing twist. In this study opposite wing twist is maximized, where opposite wing twist refers to twist in the direction opposite that caused by the aerodynamic loads. This can be achieved using composite sandwich structures in which inherent bending-twisting coupling allows tailoring of the twisting response. For a given loading the object of this research was to determine the optimum layup (orientation and number of plies in the top and bottom face sheets) which maximizes opposite wing twist while minimizing weight, subject to stiffness and ply clustering constraints. Ultimately, a revised

USE OF GENETIC ALGORITHMS FOR OPTIMAL DESIGN OF LAMINATED COMPOSITE SANDWICH PANELS WITH BENDING-TWISTING COUPLING

B. Malott and R.C. Averill
Department of Materials Science and Mechanics
and
E.D. Goodman, Y. Ding, and W.F. Punch
Department of Computer Science
Michigan State University
East Lansing, MI 48824-1226

Abstract

A genetic algorithm approach is used for optimization of a cantilever sandwich plate (an idealization of an airfoil) with a vertical force applied to one free corner. The number of face sheet plies and the ply orientations are designed so as to minimize the weight of the structure while maximizing the twist in the direction opposite that caused by the loading, subject to stiffness, strength, and ply clustering penalties. A new type of GA topology called island injection, developed in a previous study for design of composite beams, is extended to the present case of sandwich panel design. Island injection GAs showed slightly better performance over ring and single-node topologies with identical parameters and population sizes. In all cases, the GA successfully identified designs with the desired structural response.

1. Introduction

Laminated composite structure design lends itself quite openly to optimization techniques. Ply thickness and stacking sequence are two often optimized design variables when considering a laminate design. Numerous optimization techniques are available, one category of which is calculus-based or gradient-based methods. Gradient-based methods can be classified as either direct or indirect [1]. Both approaches rely on the gradient of the objective function providing necessary information to locate a local optimum. One obvious drawback of this method is the gradient of the function must exist and be obtainable. Also, these methods seek local optima, which may or may not be the global optimum. Gradient based methods are clearly not well suited for finding singular optima common in optimization of laminated composite structures.

Various algorithms exist which search finite design spaces. Evaluating an objective function at all points in a given domain is obviously inefficient for large domains. Random search techniques perform a random search of the design space for optima, but are also inefficient. Directed random search techniques are a more effective means of searching a design space. Such methods may, for example, mimic nature in their search for an optimum state. Genetic algorithms (GA) are guided random searches which utilize information obtained during the search for direction. This technique is relatively effective and robust by comparison to other methods.

Nature's means of dealing with existence and reproduction of all species is described as "survival of the fittest". Genetic algorithms draw from this phenomenon, employing a similar philosophy when searching a design space for global optima. GAs maintain one or more populations containing a number of possible designs, and perform operations of selection, reproduction, crossover, mutation, and permutation of the coded design variables.

Genetic algorithms differ intrinsically from traditional search and optimization techniques. As previously stated, traditional methods utilize derivatives whereas GAs make use of objective function data. Also, GAs work with a coded set of variables and not the actual variables as do traditional methods. Another important distinction is that GAs simultaneously search from numerous points in the design space while traditional methods search from a single point. Thus it is apparent traditional methods may locate a local maximum and not the global maximum, depending on the initiation point of the search. GAs can provide many "near optimal" designs which is advantageous to the designer. Finally, GAs are governed by stochastic rules, and traditional methods are governed by deterministic rules.

The design of composite laminates with design variables of ply thickness and fiber orientation is a common continuous optimization problem. With only these two design variables present the optimization is an integer problem which can be linearized. Branch and bound algorithms have been used to solve this type of problem once it has been linearized as done by Haftka, et al. [2]. In their study, results for optimization of stacking sequence obtained via branch and bound methods were found to be within 0.5% of those found using GAs. Genetic algorithms can optimize nonlinear problems without the need for linearization. Inclusion of frequency and strength constraints, such as in buckling load optimization problems, leads to a nonlinear problem.

The advantage of a GA design approach can be found primarily in the *kind* of design it can enable. A categorization often used in the literature is that design is either *routine*, *innovative* or *creative*. Routine design is based on the