Optimization and Blending of Composite Laminates Using Guide Based Genetic Algorithms *

David B. Adams† Layne T. Watson‡
Zafer Gürdal§ Christine M. Anderson-Cook¶

Abstract

Composite panel structure optimization is commonly decomposed into panel optimization subproblems, with specified local loads, resulting in manufacturing incompatibilities between adjacent panel designs. A new method proposed here for constructing globally blended panel designs uses a parallel decomposition antithetical to that of earlier work. Rather than performing concurrent panel genetic optimizations, a single genetic optimization is conducted for the entire structure with the parallelism solely within the fitness evaluations. A guide based genetic algorithm approach is introduced to exclusively generate and evaluate valid globally blended designs, utilizing a simple master-slave parallel implementation, implicitly reducing the size of the problem design space and increasing the quality of discovered local optima.

1 Introduction

The design of fiber-reinforced composite laminates is a discrete optimization process involving the specification of material type and orientation of ply layers in the stacking sequence. For large structures, such as the design of a wing or fuselage, the optimization is divided into smaller, tractable, subproblems using predefined local loads to constrain the optimization [1][2][3][4]. Serious manufacturing difficulties occur during isolated local optimization as resulting stacking sequence orientations vary widely between adjacent panels generating the need for a globally blended solution.

* This work was supported in part by AFOSR Grant F496320-99-1-0128 and NSF Grant DMI-9979711.

† Department of Computer Science, Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061-0106.

‡ Departments of Computer Science and Mathematics, Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061-0106.

§ Departments of Aerospace and Ocean Engineering, and Engineering Science and Mechanics, Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061-0203.

¶ Department of Statistics, Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061-0439.

Earlier work on genetic algorithm (GA) based blending includes the use of the edit distance metric to allow a set of independently evolving panel populations to evolve to a blended global solution using reference migration [1], the use of sublaminate definitions and design variable zones by Sormekun et al. [2], and the addition of continuity constraints proposed by Liu and Haftka [5]. The edit distance method utilizes a multiple-deme parallel approach that obtains blended designs through evolutionary pressures from neighboring populations. It is a nonstandard parallel decomposition that works asynchronously in real time to emulate semi-isolated populations with random migration to produce blended global designs from a pool of globally unconstrained local stacking sequence design possibilities. Evolutionary pressures are controlled through a user defined scaling factor that modifies the severity of penalties imposed for blending mismatches. These penalties, however, were found to hinder convergence to a global optimum by creating local optima in the search space that are artifacts of the algorithm itself [1]. The approach used in [2] is a two step procedure that relies on first optimizing the individual panels followed by identifying common thickness zones across multiple panels that are redefined and reoptimized using blended stacking sequences. Because of the heuristic nature of the approach, however, it is possible that suboptimal designs are generated, although blended designs are obtained with little weight penalty compared to unblended minimum weight structural design.

The particular focus of this work is to introduce a guide based genetic algorithm that implicitly reduces the search space of possible designs by forcing the generation of global (defining the entire structure) individuals that are always completely blended. Multiple demes are replaced by a single population of interbreeding individuals, concentrating the parallelism in master-slave load distribution analysis. This method is a fundamentally different parallel decomposition of the problem from that in Adams et al. [1]. Instead of discovering a globally blended design at the end of the design process, all designs considered are always blended throughout the genetic algorithm.

Section 2 provides some background on genetic algorithms. Section 3 provides background on the blending problem and GA approaches to it, specifically the asynchronous parallel approach based on edit distance and reference migration. Section 4 defines the guide based genetic algorithm approach and outlines the parallel distribution of work. Section 5 presents results for an an 18 panel design problem with comparisons to other work in the literature. The intent is to compare and contrast qualitatively different blending methods applied to the 18 panel problem.

2 Genetic Alogorithms

A genetic algorithm (GA) is a nondeterministic directed search algorithm using ideas based on natural selection to guide the exploration of the search space toward a global optimum. Common elements that occur in most genetic algorithms are those of population initialization, parent selection, crossover, mutation, and the selection of successive generations. Each element of the algorithm has many variations, modified to suit the needs of the problem at hand, including but not limited to attempts to mimic natural genetics in every phase. The original work on genetic algorithms is attributed to Holland [6] in

1975, with application work following soon after in static function optimization by DeJong [7]. Goldberg [8] popularized the idea with his book in 1989, and is cited extensively in the literature defining genetic algorithms as search procedures based on the mechanics of natural selection and natural genetics. Bäck [9] is an excellent recent reference. Much of the work today using genetic algorithms can still fit this definition, though the concepts of natural selection and genetics are expanded to encompass some unnatural elements beyond the pristine translations from biology. In practice, genetic operators are tailored to specific problems in ways that have no analog in nature [3][10][11].

3 The Blending Problem

The blending problem here is defined as a laminate stacking sequence optimization on multiple interconnected panels with given local loading constraints. Each laminate is assumed to be balanced, symmetric, and constructed with a finite number of possible fixed orientations for each ply. The goal is to minimize the weight of the structure, determining the material type and orientation of each ply layer of every panel such that the loading constraints are satisfied locally and the panels form a blended global design. A blending measure for this work is defined in Section 4, but in general can be any measure of the manufacturability of a global design. Maintaining the continuity of material type and orientation across panel design boundaries, in part or in whole, is the primary concern in determining the blendedness of a global design. Ply orientation mismatches across panel design boundaries can cause manufacturing costs to rise in addition to the structural integrity issues associated with continuity breaks in the material.

3.1 Laminate Encoding

Each laminate must be encoded for use in the genetic algorithm. Following the coding scheme used by McMahon [12], integer values from zero to seven represent the orientation of each ply. The positive integers map to orientation angles 0, 15, 30, 45, 60, 75, and 90 degrees from one to seven, respectively, with the zero encoding representing an empty ply. Successive occurrences of a gene map to the orientation with alternating plus and minus signs, starting from the center of the (symmetric) laminate. For example, the first occurrence of a 2 encoding maps to $15°$, the second 2 maps to $-15°$, and so on. This is the meaning of balanced. Symmetric means that the stacking sequence is symmetric with respect to the mid-plane of the laminate, and therefore only half of the orientations actually need to be encoded. Similarly, the material type for each ply is represented by an integer, but here the material type is restricted to a single value meaning the material type for each ply is the same. The current work dissallows empty ply layers to be present during genetic operations. However, the zero encoding is still utilized by the analysis portion of the algorithm through a ply layer stripping process as described in Section 4 to allow laminate thickness variation to minimize its weight.

3.2 Implementation Details

An important aspect of the functioning of a GA concerns how individuals are selected to become parents of a child in the next generation. Since the goal of the algorithm is to simulate a survival of the fittest evolution strategy, more fit individuals should have a higher probability of producing children. This is accomplished using a roulette wheel selection process common in the literature [3][8]. The roulette wheel assigns to each individual of a population a probability of being selected according to its fitness ranking. The probability is biased such that the chance of selecting a particular individual is greater than choosing another individual with lower fitness. The precise method for associating the probabilities is given by

$$\text{fraction of roulette wheel} = \frac{2(N+1-i)}{N^2+N}$$

where i represents the ith best individual in a local population of interbreeding individuals or deme of size N. The standard Fortran 95 random number generator is used to provide approximate uniform distributions where random numbers are required to direct evolution.

A similarly important component of the GA is the ranking of the parents and ways in which they are allowed to combine to create a child design. This is referred to as the crossover type. For all the runs presented in this paper a simple one-point crossover is implemented. A single crossover point falls within selected parent strings. The two parent strings are then divided at that crossover point and the left piece from parent one is combined with the right piece from parent two. Similarly a second child is created with the right piece from parent one and the left piece from parent two. Although the crossover type can have a large impact on the variation of designs obtained from the initial population, the point of this paper is simply to review the feasibility of the guide based design blending algorithm. No further crossover types are examined. However, to generate new designs throughout the generations, a small probability of mutation is added allowing random genes in a child to change their orientation once crossover is complete. Population initialization is accomplished through random generation of encodings, requiring that no duplicates exist. Child populations are required to be unique with respect to the current filling child population and the generating parent population. Once the child population is created a selection scheme is invoked to choose the next generation from the union of the child population and the parent population. Results are presented using standard elitist selection in Section 5 [13].

3.3 Edit Distance and Reference Migration

The edit distance blending method defined by Adams et al. [1] is briefly described here to highlight the identifying characteristics of the algorithm and to provide a stage to present guide based design. The method is built upon the idea of a measure of blendedness. A metric, the edit distance, is used to determine the degree of blendedness when two panel stacking sequences are compared with each other. This measure can then be used to alter the fitness function of a GA, rewarding designs that have a higher degree of blendedness with a reference design.

The algorithm utilizes a GA optimization on each panel using predetermined local loading constraints. Each panel optimization is allocated to a single processing node in a parallel processing environment and is run asynchronously with respect to other panels. Migrant individuals are sent out at random to neighboring nodes and stored for reference. The migration is referred to as reference migration since the migrants do not enter the target population. Instead, migrants are used to modify the fitness function of the target panel optimization through the measure of blendedness. Those designs that are evolving closely with neighboring panel designs are rewarded with better fitness values and are more likely to create children. This creates the possibility for a localized optimization to converge to a globally blended design. The method met with limited success. Though it was used to discover the best known global design for the 18 panel problem presented in Section 5, convergence to local optima prevented the algorithm from increasing the quality of the solution with added iterations/generations. Additionally, designs that were not perfectly blended were in general locally optimal designs that had no practical utility.

Some of the major features of this edit distance based blending approach can be summarized as:

- asynchronous, real time algorithm,
- multiple demes each running an independent genetic algorithm,
- measure of blendedness (edit distance) allows for possibly nearly perfectly blended designs to result from a run that are manufacturable,
- blending accomplished through evolutionary pressures of reference migration so that unblended designs evolve to global blending,
- convergence is dictated by the scaling factor [1] and does not benefit from increased iterations of the algorithm.

4 Guide Based Design

A simplified definition of blending is used in this work. Two adjacent panel designs are outwardly (inwardly) blended if one design is obtained by deleting a contiguous series of outermost (innermost) plies from the other. A global design is perfectly outwardly (inwardly) blended if and only if blending holds for every pair of adjacent panels starting from the innermost layer near the mid-plane (starting from the outermost layer and moving inward toward the mid-plane). This simplified definition serves to reduce the search space of the problem and provide equal footing for comparisons with previous work on the same test case [1][2]. The creation of a guide based GA design process grew out of the need to limit the optimization to consider only those designs that are globally blended.

Using the simplified definition of blending, a perfectly outwardly (inwardly) blended global design can be generated by designing each local panel as an initial (final) sequence ($a_1, \ldots, a_k$ is an *initial sequence*, and $a_{n-k+1}, \ldots, a_n$ is a *final* sequence, of the sequence $a_1, \ldots, a_k, \ldots, a_n$) of a single larger guide design. Figure 1 displays the distinction between outward and inward blending rules. A GA can be created to operate on and generate a

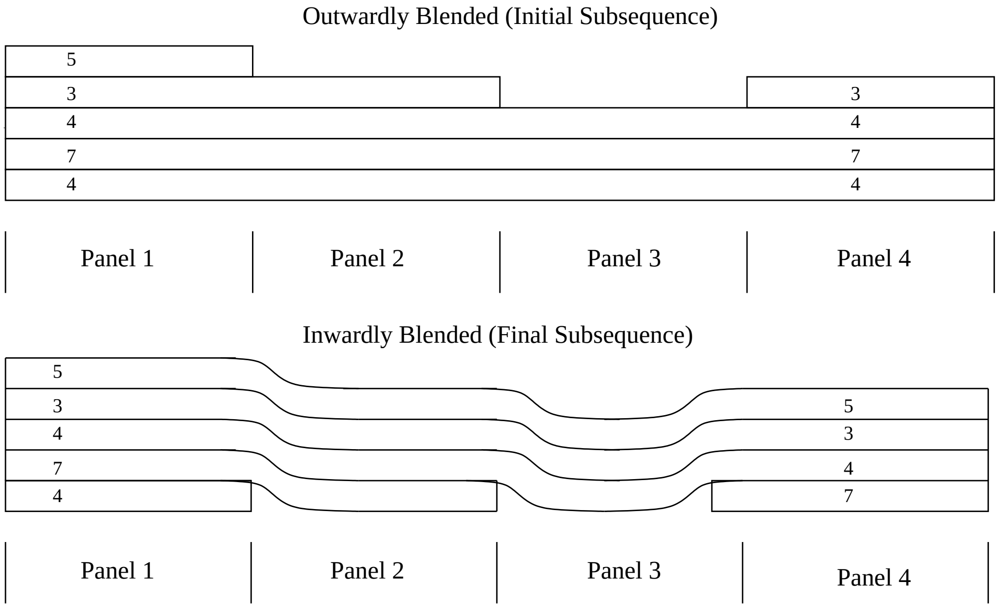

FIG. 1. *Outwardly/inwardly blended distinction example.*

population of individuals that guide the global design process so that only globally blended designs are in the problem domain. During the analysis phase of the GA a guide design (a single individual from the population) is evaluated to determine the optimal initial sequence for each local panel satisfying loading constraints and minimizing weight. This is accomplished by stripping ply layers from the guide design, starting from the outermost (for outward blending) layers, one layer at a time and analyzing the resulting designs according to the constraints for each local panel. (More efficient variations of this are possible, e.g., starting with the initial sequence length equal to that of the last optimal design analyzed for this panel, and then adding/subtracting layers from this length.) The guide designs in the population acquire fitness values through the combination of optimal local designs generated by the guide itself so that each guide design has a single best series of initial sequences corresponding to each panel that can be used to construct a usable fitness function value. An optimal initial sequence is discovered for each local panel and the fitness of the global design becomes the sum of the individual fitness values for every optimized local panel. This modified analysis method provides the means to assign fitness values to guide designs themselves and allows a GA to operate on populations of guide designs to drive a global optimization process to an optimal blended design. An illustrative fictitious example of guide based design is presented in Figure 2.

As each guide design is assigned a fitness value, an accompanying array of integers is generated to indicate how many ply layers are applied to each panel. This array is used to reconstruct the optimal design discovered for an analyzed guide for each panel and becomes a table of information associated with each generation of the population.

A simple master-slave parallel code is implemented for this work. The master process generates and runs the GA code while distributing guide designs to slave processes for analysis in a lock step message-passing phase. The load distribution is calculated statically

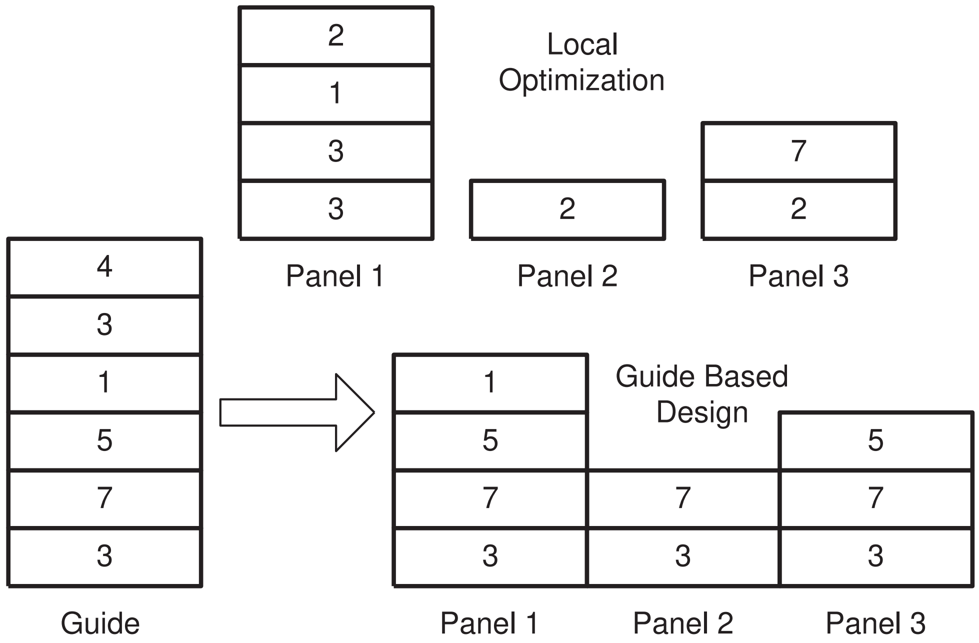

FIG. 2. *Guide based optimization example. The figure illustrates how isolated optimization creates designs that are locally optimal but are not easily manufacturable due to the ply orientation mismatches at through-the-thickness layers of the laminate. A side view of three sample panels, listing ply orientation encodings for each layer, is presented for both local optimization and a possible guide based design. Isolated local optimization generally produces light weight designs as less ply layers are required for Panels 2 and 3. The guide based design is optimized and constructed from the 6-ply thick guide design given on the left. The optimal initial sequence for the given guide produces a global design that is completely (outwardly) blended and therefore manufacturable. By operating on populations of guide designs a GA can discover an optimal blended global design minimizing weight and maximizing constraint satisfaction margins.*

as an equal division of slave processes into population size with remaining work given arbitrarily to selected nodes with lower rank. For a population size of 100, 3 slave processes were used distributing 34 guides to slave process 1 and 33 guides to slave processes 2 and 3.

Some of the major features of this new guide design based approach can be summarized as:

- synchronous, lock step algorithm,
- single deme running a distributed genetic algorithm,
- nearly blended designs are not possible,
- blending accomplished directly through generation from a guide design to create globally blended designs,
- convergence is dictated by the GA parameters and allows additional generations to increase the probability that the top ranked solution is indeed optimal.

Observe that these characteristics are all notably different from the characteristics itemized for the edit distance approach in Section 3.3.

4.1 Master-Slave Parallelism

Master-slave implementations are the simplest and most common form of parallel genetic algorithms. The classification of master-slave encompasses a wide variety of design decisions that affect the performance and scalability of the algorithm. A nice overview of parameter evaluation for parallel genetic algorithms can be found in the collection of work by Erick Cantú-Paz [14], in addition to the classical parallel algorithm design overview found in Quinn [15]. For the application developer, work load and communication time balancing are the primary concerns for applying a master-slave paradigm. General practice assigns the master process all the duties associated with the generic GA while distributing analysis work (fitness evaluation) to slave nodes.

Though a static distribution of work can perform well using deterministic non-iterative analysis routines, a more robust and scalable distribution method is required for complex analyses. Since the time required for a single analysis can vary greatly in iterative methods, a dynamic producer-consumer model is more appropriate. That is, the master generates a queue of available work at each generation that is then distributed in small subsets to those slave processes requesting work. The slave processes consume work units and request more when they have completed their subset. For the current problem the smallest subdivision of labor, without parallelizing the analysis itself, is a single analysis of a local panel design. The work queue then at each generation becomes population size multiplied by the number of panels and the number of plies. For the 18 panel design problem presented in Section 5 there are 43,200 ($100 \times 18 \times 24$, see Table 1) work units per generation.

The determination of how many jobs are distributed per request to those processes requesting work is part of the balance of communication and computation time. Similarly the boundaries on which work is distributed can relieve the master process from a large portion of data gathering. Dividing the work of a guide design analysis along panel division boundaries allows slave processes to only report the optimal local design for a given panel. Otherwise, the master process would be required to compute a min/max for each panel and choose the one most fit. Requiring this work from the master process increases the communication load by a factor of N, where N is the number of ply layers in a guide. The work load chunk then becomes the number W of panel analyses distributed to each processor such that the maximum tolerance for wasted compute time is bounded by the time T it takes to evaluate W designs. The worst case scenario is that all slave processes request work simultaneously for a remaining single unit of size W. In this case, all but one process remains idle for T units of time.

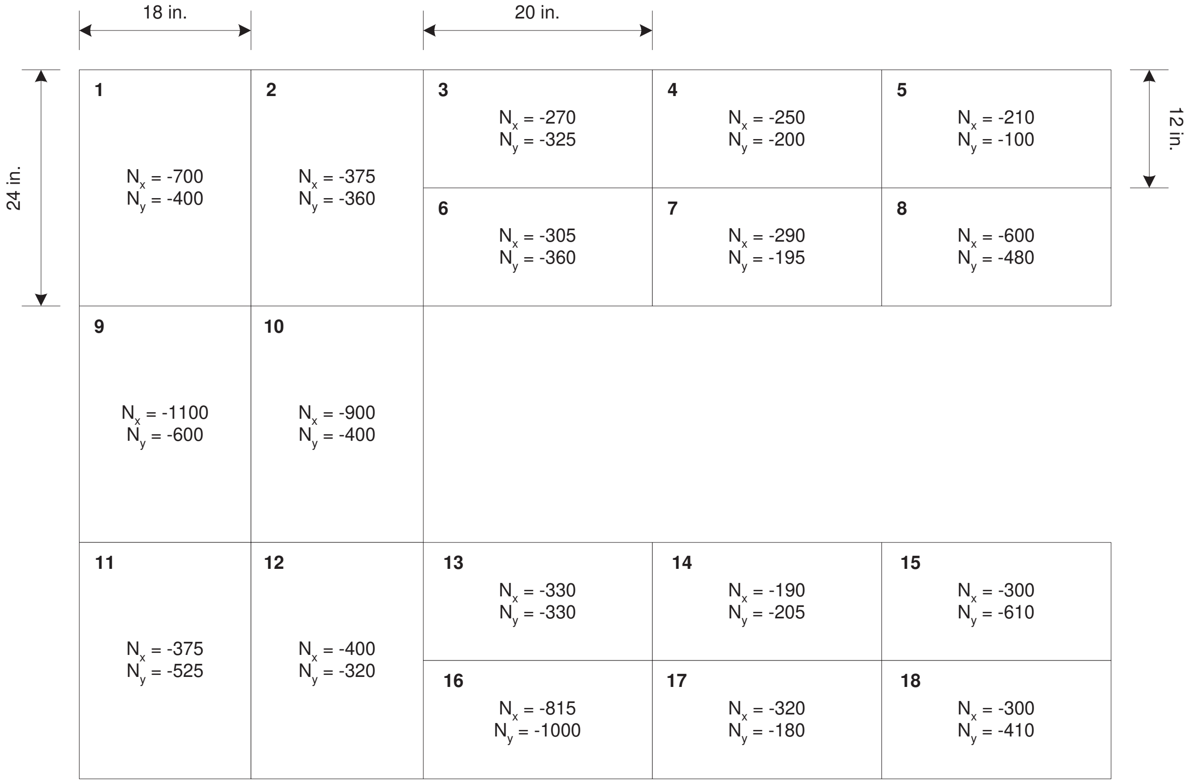

FIG. 3. *Eighteen panel irregular grid test problem.*

TABLE 1. *Static GA parameters.*

Probability of crossover	1.00
Probability of mutation (orientation)	0.05
Population size	100
Laminate ply genes	24
Number of elite retained	1
Termination criterion (generations)	2000

5 Results

Results are given here for a test problem consisting of eighteen panels configured and loaded as shown (N_x, N_y are the loads in the horizontal, vertical directions given in lb/in) in Figure 3 [2]. The panels are composed of 0, ±15, ±30, ±45, ±60, ±75, and 90 degree ply orientations. One material type, graphite-epoxy (IM7/8552), is used for construction of each ply. Table 1 summarizes the parameters of the simple GA operating on guide designs.

For comparison, each of the 18 panels is isolated and optimized locally without respect to a blending measure. The addition of blending constraints on the problem restricts possible global designs and allows individual panel optimization to serve as a lower bound for global weight. Though the global design obtained from isolated local panel optimization is not blended, the reference point given by the weight is useful in determining the quality of a blended global design. The resulting global design is presented in Table 2 with a total structural weight of 27.92 kg [1].

TABLE 2. *Isolated local panel optimization, total structural weight 27.92.*

Panel	Weight (kg)	Design
1	2.68	$[\pm 30_8]_s$
2	2.35	$[\pm 30_7]_s$
3	0.93	$[\pm 75_2/-75/-60/75/60/\pm 75]_s$
4	0.84	$[-75/60/75/-60/\pm 75/-75/60/75]_s$
5	0.74	$[60/\pm 60/75/-75/-60/75/60]_s$
6	1.03	$[\pm 75_2/-75/60/75/-75/\pm 60/75]_s$
7	0.84	$[60/\pm 75/\pm 60/\pm 75_2]_s$
8	1.12	$[60/\mp 75/-60/75/\pm 75_2/60/\pm 75]_s$
9	3.18	$[30/\pm 30_9]_s$
10	2.85	$[-45/\pm 30_5/-30/15/30/-30/45/30]_s$
11	2.51	$[30/\pm 30_7]_s$
12	2.35	$[\pm 30_7]_s$
13	1.03	$[\pm 75/-75/60/75/-75/\pm 60/75/\pm 75]_s$
14	0.84	$[75/\pm 75/-60/\pm 75/60/\pm 75]_s$
15	1.11	$[\pm 75_6]_s$
16	1.67	$[\pm 75_9]_s$
17	0.84	$[60/-60/-75/60/75/\pm 75_2]_s$
18	1.03	$[75/\pm 75_5]_s$

Sormekun et al. [2] discovered a blended design, using a less restrictive blending measure, with a total structural weight of 29.21 kg. The method, as briefly described earlier, used an iterative process to fix layers of the global design based on local designs with the fewest ply layers. Though the method was less restrictive with respect to blending than the methods in [1] and here, it serves as a measuring stick to determine the quality of other blended designs.

A slightly better design was discovered using the edit distance blending metric with umbra migration in previous work by Adams et al. [1]. The edit distance approach creates multiple demes, each running an independent local optimization for an individual panel, using migration to transfer evolutionary data between neighboring panels. That is, each panel is optimized locally in real time by an independent genetic algorithm while modifying the objection function values of individuals based on a blending metric comparison with migrants from adjacent evolving populations. A modified edit distance metric serves to measure the blendedness of adjacent panels by counting the number of edit operations required to turn the encoding of one individual into another while considering empty plies

TABLE 3. *Guide based design with total structural weight 28.63 kg.*

Panel	Ply Encodings	Design								
1	17	[		± 45	30	∓ 30	45	∓ 45	60	$-60/\pm 60_2/-75/60/75]_s$
2	14	[				∓ 30	45	∓ 45	60	$-60/\pm 60_2/-75/60/75]_s$
3	11	[						∓ 45	60	$-60/\pm 60_2/-75/60/75]_s$
4	9	[							60	$-60/\pm 60_2/-75/60/75]_s$
5	8	[								$-60/\pm 60_2/-75/60/75]_s$
6	11	[						∓ 45	60	$-60/\pm 60_2/-75/60/75]_s$
7	9	[							60	$-60/\pm 60_2/-75/60/75]_s$
8	13	[				30	45	∓ 45	60	$-60/\pm 60_2/-75/60/75]_s$
9	19	[	± 45	± 45	30	∓ 30	45	∓ 45	60	$-60/\pm 60_2/-75/60/75]_s$
10	18	[	-45	± 45	30	∓ 30	45	∓ 45	60	$-60/\pm 60_2/-75/60/75]_s$
11	15	[			30	∓ 30	45	∓ 45	60	$-60/\pm 60_2/-75/60/75]_s$
12	14	[				∓ 30	45	∓ 45	60	$-60/\pm 60_2/-75/60/75]_s$
13	11	[						∓ 45	60	$-60/\pm 60_2/-75/60/75]_s$
14	9	[							60	$-60/\pm 60_2/-75/60/75]_s$
15	13	[				30	45	∓ 45	60	$-60/\pm 60_2/-75/60/75]_s$
16	19	[	± 45	± 45	30	∓ 30	45	∓ 45	60	$-60/\pm 60_2/-75/60/75]_s$
17	9	[							60	$-60/\pm 60_2/-75/60/75]_s$
18	11	[						∓ 45	60	$-60/\pm 60_2/-75/60/75]_s$

to blend with any orientation. Forced convergence of the umbra migration method creates global designs that are perfectly blended using the same measure of blendedness given in Section 4. The best discovered design had a total structural weight of 29.19 kg and falls within the domain of guide designs explored by the current work. In fact, all converged designs using umbra migration fall within this domain. It is interesting to note though that once convergence is achieved it becomes impossible for other blended designs to be evaluated by this method. The user has little confidence in the quality of the answer until many runs have been made and, unlike other GA based methods, additional generations do not provide the possibility of finding a better solution on a single run.

Using guide based design even for a small number of generations, given the size of the problem design space, produced (outwardly) blended designs that were consistently better than the best known umbra migration results. Table 3 displays one of many optimal (best known) designs discovered with a weight of only 28.63 kg. In fact, approximately 90% of the runs conducted for 2000 iterations had a final design weight of 28.63 kg, differing only sightly in their fitness values, caused by different constraint margins. Not only does this provide the practicing engineer with a wealth of good designs to choose from, the algorithm itself is more robust (than those in [1] and [2]) and can continue to find possibly

TABLE 4. *Other guide based designs found. Each design is described by three lines. Line 1: guide design. Line 2: constructor defining the number of plies for panels 1–18, left to right. Line 3: weight (kg)/fitness value (reflects constraint margins, larger is better).*

$[60/90_2/75/0/\pm 45_2/30/\mp 30/45/\mp 45/\pm 60_3/-75/60/75]_s$	17 14 11 9 8 11 9 13 19 18 15 14 11 9 13 19 9 11	$28.63/-81.103688$
$[15/90/15/0/90/-45/\pm 45_2/30/\mp 30/\pm 45/\pm 60/75/\pm 60/45/60/\mp 45]_s$	17 14 11 9 8 11 9 13 19 18 15 14 11 9 13 19 9 11	$28.63/-81.173725$
$[-60/90/-30/90/15/\pm 45_2/30/\mp 30/45/\mp 45/60/\mp 60_3/\mp 75]_s$	17 14 11 9 8 11 9 13 19 18 15 14 11 9 13 19 9 11	$28.63/-81.106705$
$[75/15/-30/90/-60/45/\mp 45_2/30/\mp 30/\mp 45/60/\mp 60_2/\mp 75_2]_s$	17 14 11 9 8 11 9 13 19 18 15 14 11 9 13 19 9 11	$28.63/-81.093024$
$[90/30/-45/-30/90/\pm 45_2/\pm 30/45/30/\mp 45/60/\mp 60_3/\mp 75]_s$	17 14 11 9 8 11 9 13 19 18 15 14 11 9 13 19 9 11	$28.63/-81.106640$
$[-75/45/60/0/90/45/\mp 45_2/30/\mp 30/\mp 45/\mp 60_2/75/\mp 60/\mp 75]_s$	17 14 11 9 8 11 9 13 19 18 15 14 11 9 13 19 9 11	$28.63/-81.091774$

better solutions given more generations to explore the design space. Table 4 lists some of these other (outwardly blended) designs found, showing the variability possible among good designs, though same weight designs seemed to converge to the same number of plies for each location. Perhaps the strongest argument for using GAs for composite structure design is the (typical) production of many dissimilar yet quite good composite designs.

Depending on the thickness differences between adjacent panels, inwardly blended designs may have lower local stress concentrations than outwardly blended designs. For comparison with the outwardly blended optimal design in Table 3, Table 5 shows an optimal inwardly blended design. While there are many significantly different minimum weight outwardly blended designs, numberous GA runs produced essentially the same minimum weight inwardly blended design in Table 5. Note also that the optimal inwardly blended design (29.00 kg) is heavier than the optimal outwardly blended design (28.63 kg).

6 Conclusions

A new guide based parallel GA has been proposed for composite panel structure optimization. For an 18-panel test problem, the present algorithm produced the lightest known design, as well as a variety of minimal weight near optimal designs. This production of several substantially different near optimal designs has practical importance, and is perhaps

Table 5. *Guide based design (inwardly blended) with total structural weight 29.00 kg.*

Panel	Ply Encodings	Design					
1	17	$[60/\pm 45_3/\pm 45$	± 45	± 45	± 45	± 45	$]_s$
2	14	$[60/\pm 45_3/\pm 45$	± 45	± 45	45		$]_s$
3	11	$[60/\pm 45_3/\pm 45$	± 45				$]_s$
4	10	$[60/\pm 45_3/\pm 45$	45				$]_s$
5	8	$[60/\pm 45_3/45$					$]_s$
6	11	$[60/\pm 45_3/\pm 45$	± 45				$]_s$
7	10	$[60/\pm 45_3/\pm 45$	45				$]_s$
8	13	$[60/\pm 45_3/\pm 45$	± 45	± 45			$]_s$
9	19	$[60/\pm 45_3/\pm 45$	± 45	± 45	± 45	± 45	$\pm 45]_s$
10	18	$[60/\pm 45_3/\pm 45$	± 45	± 45	± 45	± 45	$45]_s$
11	15	$[60/\pm 45_3/\pm 45$	± 45	± 45	± 45		$]_s$
12	14	$[60/\pm 45_3/\pm 45$	± 45	± 45	45		$]_s$
13	11	$[60/\pm 45_3/\pm 45$	± 45				$]_s$
14	9	$[60/\pm 45_3/\pm 45$					$]_s$
15	13	$[60/\pm 45_3/\pm 45$	± 45	± 45			$]_s$
16	19	$[60/\pm 45_3/\pm 45$	± 45	± 45	± 45	± 45	$\pm 45]_s$
17	10	$[60/\pm 45_3/\pm 45$	45				$]_s$
18	12	$[60/\pm 45_3/\pm 45$	± 45	45			$]_s$

the strongest justification for the use of GAs in composite structure design. The definition of blending here amounts to perfect blending, although a less restrictive blending definition could still result in designs manufacturable with the minimal number of tow machine passes. Also, in general the local loads depend on the panel designs, so the algorithm here must be iterated to a local load fixed point. Future work includes precise mathematical definitions of blending, manufacturable, and a fixed point algorithm for the local panel optima dependent on the local loads.

References

[1] D. B. Adams, L. T. Watson, and Z. Gürdal, *Optimization and blending of composite laminates using genetic algorithms with migration*, Mechanics of Advanced Materials and Structures, to appear.

[2] G. Sormekun, Z. Gürdal, C. Kassapoglou, and D. Toni, *Stacking sequence blending of multiple composite laminates using genetic algorithms*, Composite Structures, 56 (2002), pp. 53–62.

[3] M. T. McMahon and L. T. Watson, *A distributed genetic algorithm with migration for the design of composite laminate structures*, Parallel Algorithms and Applications, 14 (2000), pp. 329–362.

[4] B. P. Kristinsdottir, Z. B. Zabinsky, M. E. Tuttle, and S. Neogi, *Optimal design of large composite panels with varying loads*, Composite Structures, 51 (2001), pp. 93–102.

[5] B. Liu and R. T. Haftka, *Composite wing structural design optimization with continuity constraints*, in Proceedings of 42nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA Paper 2001-1205, Seattle, WA, 2001.

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

[7] K. A. De Jong, *An analysis of the behavior of a class of genetic adaptive systems*, Ph.D. dissertation, Univ. of Michigan, Ann Arbor, MI, 1975.

[8] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley, Reading, MA, 1989

[9] T. Bäck, *Evolutionary Algorithms in Theory and Practice*, Oxford University Press, NY, 1996

[10] N. Kogiso, L. T. Watson, Z. Gürdal, R. T. Haftka, and S. Nagendra, *Design of composite laminates by a genetic algorithm with memory*, Mechanics of Composite Materials and Structures, 1 (1994), pp. 95–117.

[11] R. Le Riche, *Optimization of composite structures by genetic algorithms*, Ph.D. dissertation, Department of Aerospace Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA 1994.

[12] M. T. McMahon and L. T. Watson, *A distributed genetic algorithm with migration for the design of composite laminate structures*, Technical Report TR-98-20, Dept. of Computer Science, Virginia Polytechnic Institute and State University, Blacksburg, VA, 1998.

[13] G. Soremekun, Z. Gürdal, R. T. Haftka, and L. T. Watson, *Composite laminate design optimization by genetic algorithm with generalized elitist selection*, Computers and Structures, 79 (2001), pp. 131–144.

[14] E. Cantú-Paz, *Efficient and Accurate Parallel Genetic Algorithms*, Kluwer Academic Publishers, Boston, MA, 2001

[15] M. J. Quinn, *Designing Efficient Algorithms for Parallel Computers*, McGraw-Hill Book Company, New York, NY, 1987