Chapter 3

Optimum Design of Stiffened Composite Panels Using Genetic Algorithms

3.1 Introduction

In the previous chapter, a nonlinear finite element for the postbuckling analysis of thin composite panels was introduced. A finite element computer code for analyses of composite panels, called FEPA, was developed. The objective of this chapter is to combine the finite element code FEPA with a genetic algorithm optimization code that was developed by Soremekun and Gurdal [70] in order to obtain a postbuckling design optimization code for stiffened composite panels, FEPAD (Finite Element Postbuckling Analysis and Design).

The design of composite laminates is often formulated as a continuous optimization problem with ply thickness and ply orientation angles used as design variables (e.g. Schmit and Farshi [92]). However, for many practical problems, ply thicknesses are fixed and ply orientation angles are limited to a small set of angles such as 0, 90 and 45 degrees. Designing the laminate then becomes a stacking sequence optimization which can be formulated as an integer programming problem. In this study, genetic algorithms are applied to this integer programming problem.

Applications of genetic algorithms to the optimum design of nonconvex engineering problems can be found in several recent studies (e.g. Refs [93-97]). In the area of composite structural design, recent applications of genetic algorithms include Callahan and Weeks [98], and Ball et al [99]. Le Riche and Haftka [100] solved the laminate stacking sequence design problem subject to buckling and strength constraints. Nagendra et al. [101-102] applied the same approach to the design of stiffened composite panels. In both works [101,102] it was found that the genetic algorithm can find a variety of alternative designs with similar performance, thus giving the designer a choice of alternatives.

3.2 Problem Description

The panel under consideration is 1168 *mm* long and 178 *mm* wide and has one blade stiffener of 37 *mm* height. The panel is designed for an axial compressive load of 65000 *N*. The panel's loaded edges are clamped while the longitudinal edges are simply supported. A total of 74 finite elements are used to model the skin, while 12 elements are used to model the blade segment. Thus, a total of 104 nodal points were used in the finite element model (624 degrees of freedom). The panel's geometry and finite element model are shown in Figure (3.1). The skin and stiffener blade are constrained to be balanced and symmetric laminates made up of 0^o, $\pm 45^o$ and 90^o plies. To account for geometric imperfections, it is assumed that the panel has an initial profile whose shape is the same as its first buckling mode shape. The modal amplitude is assumed to be 0.1% of the panel length, thus making it constant during the optimization process. Notice that assuming the modal amplitude proportional to the laminate thickness would result in different imperfections throughout the optimization. It is important to recall here that the main objective of this study is to incorporate both manufacturing and uncertainty information in the determination of the imperfection profile to be used in the design. This new approach will be presented in detail in Chapter 6.

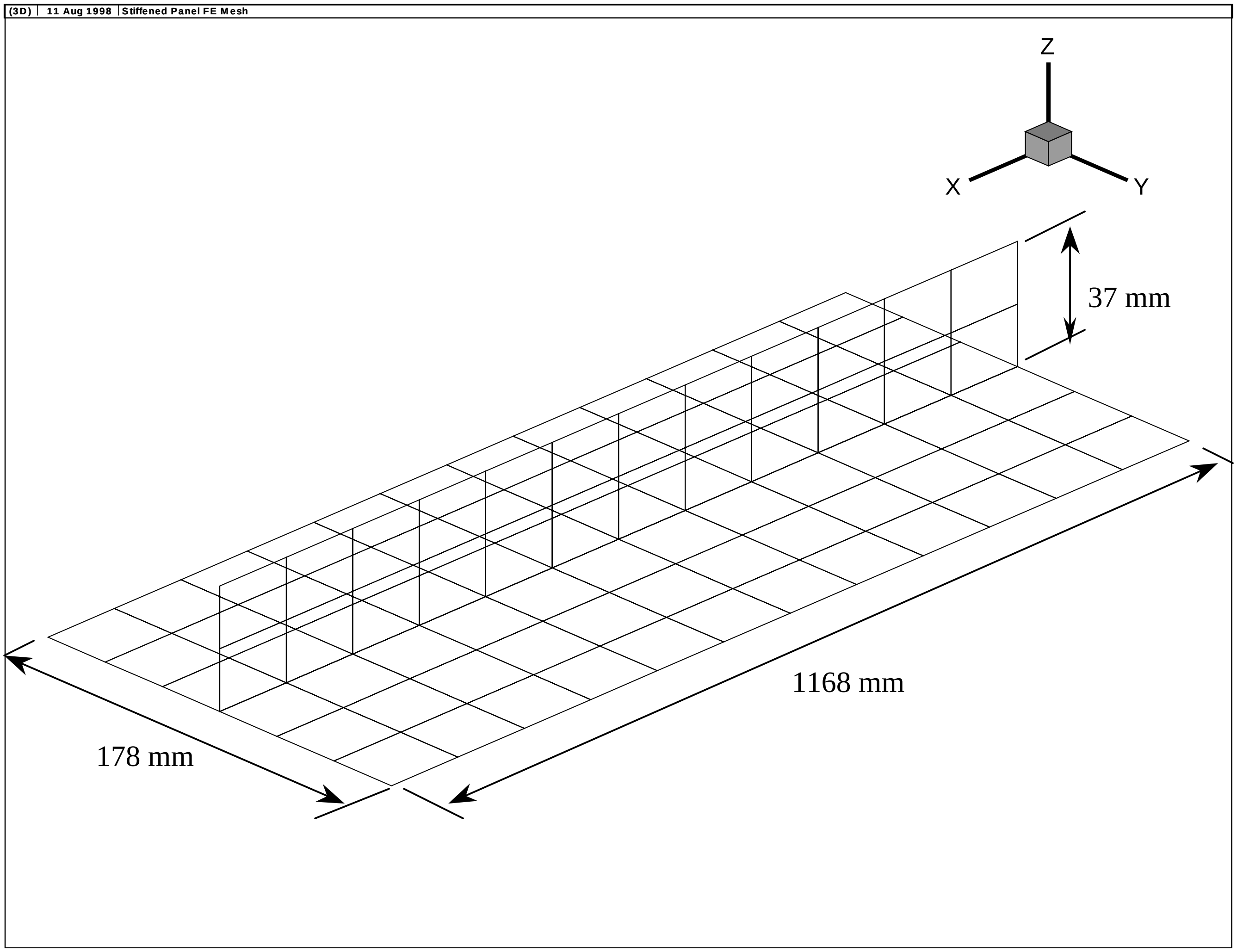

Figure (3.1) Panel geometry and finite element model

3.3 Optimization Formulation

The goal of the optimization is to find the stacking sequence of minimum weight panels that will not fail due to excessive stresses in the postbuckling range for the previously defined design load and boundary conditions. In addition the optimum design must satisfy the balanced stacking sequence constraint.

The panel weight W is a function of the number of layers in the skin n_s, the number of layers in the blade n_b, and the planform areas A_s and A_b of the skin and blade respectively. Thus, the panel weight can be calculated as

$$W = \rho\, t\,[n_s A_s + n_b A_b] \tag{3.1}$$

where t and ρ are the ply thickness and density given in Table (3.1) along with the material properties for Hercules AS4/3502 graphite/epoxy used in this study. Also given in Table (3.1) are the allowable stresses used in determining the panel failure load.

Table (3.1) Hercules AS4/3502 graphite/epoxy lamina material properties

Young's Modulus (longitudinal)	$E_{11} = 131.0 \times 10^6\ N/m^2$
Young's Modulus (transverse)	$E_{22} = 13.0 \times 10^6\ N/m^2$
Shear Modulus	$G_{12} = 6.4 \times 10^6\ N/m^2$
Poisson's Ratio	$\nu_{12} = 0.38$
Density	$\rho = 1577.8\ kg/m^3$
Ply Thickness	$t = 0.14\ mm$
Allowable Stresses	
$\sigma_{1\max}(tension) = 1400\ N/mm^2$	
$\sigma_{1\max}(compression) = -1138\ N/mm^2$	
$\sigma_{2\max}(tension) = 80.9\ N/mm^2$	

$$\sigma_{2\max}(compression) = -189\, N/mm^2$$

$$\tau_{12\max} = 69\, N/mm^2$$

The failure criterion used in this work is the maximum stress criterion. This criterion is a single-point phenomenological theory used to predict first ply failure. Any other closed-form criteria could have been used, including strain-based criteria. The maximum stress failure criteria can be described as follows:

Failure occurs if any of the following conditions are satisfied :

$$X_c > \sigma_1 > X_t$$

$$Y_c > \sigma_2 > Y_t \qquad (3.2)$$

$$\sigma_6 > T$$

where σ_1, σ_2 are normal stresses along the fiber and normal to the fiber, respectively; and σ_6 is the shear stress in the 12-plane; X and Y correspond to the strengths in the 1 and 2 directions, and subscripts *t* and *c* denote tension and compression respectively. The optimization problem can be formulated as finding the stacking sequences of the panel (i.e. ply orientation θ_i of the i^{th} ply) of the skin, and of the stiffener laminates in order to minimize the weight *W* of the panel.

$$\text{Minimize} \qquad W(n_s, n_b)$$

$$\text{Subject to} \qquad \frac{P_f}{P_D} \geq 1 \;\text{ or }\; \lambda_f \geq 1 \qquad (3.3)$$

$$\text{And } g(\hat{\theta}) \leq 0$$

Where P_f is the panel's failure load and P_D is the panel's design load. $g(\hat{\theta})$ represents the balanced condition. The vector $\hat{\theta}$ of ply orientations has $n_s + n_b$ components.

The constrained optimization problem must be transformed into an unconstrained problem to be able to use genetic algorithms. This is done by using penalty parameters and defining a fitness function *F* as

$$F = \begin{cases} 100 - (Weight + 2.5 \times Abs(g) + P_{unbalance}) & g \leq 0 \\ 100 - (Weight - \varepsilon \times g + P_{unbalance}) & g > 0 \end{cases}$$

where

$$g = \frac{P_{fail}}{P_{Design}} - 1$$

where P_{fail} is the panel final failure load and P_{Design} is the required minimum failure load. The penalty parameter $P_{unbalance}$ is used to enforce the balanced laminate constraint. When the failure constraint is satisfied ($g > 0$) a small fraction ($\varepsilon = 0.05$) of the constraint value g is subtracted from the weight, to discriminate between multiple designs with the same weight all of which satisfy the failure constraint. The best design has the largest possible margin of all the panels of the same weight.

3.4 Implementation of the Genetic Algorithm

The implementation of the genetic algorithm is shown schematically in Figure (3.2). The process starts with the generation of a random initial population of n_d designs.

Next the decoder translates the genetic string of each design into input for FEPA. FEPA starts by incrementing the applied load and evaluates displacements, stresses and strains at each nodal point in the finite element mesh, at each load step. After these calculations are performed at a given load step, FEPA checks for failure by calculating the principal stresses at every nodal point and comparing them to the failure stresses. If failure occurred the analysis is stopped and the value of the load at a given load step is returned as P_{fail} for the specific design. If no failure occurred, the load is increased in steps until the design load P_{Design} is reached. Any panel failing at loads less than P_{Design} is considered nonfeasible and is penalized for violating the failure constraint. Next the panel mass and failure load are assigned to individual parent strings in the population. The fitness processor evaluates the objective function for each design and ranks the designs. The evaluated population is processed then, by means of the genetic operators, to create a new population which combines the most desirable characteristics of the old population. According to the "elitest" version of the genetic algorithm [93,103], the old population is replaced by the new one except for the best design which is always kept unchanged. This scheme guarantees that the best design will occur in the final population. The process is repeated until convergence, which is defined to occur when the maximum number of generations without improvement in the best design becomes equal to 20.

The genetic algorithm begins with the random generation of a population of design alternatives. The following selection process is biased so that high performance designs have a higher probability of transmitting their features to the next generation. This is implemented by allocating each individual a portion of a roulette wheel in proportion to their fitness. This process is described in more detail in the next section. Once parents are selected various genetic operators are applied to create child designs. The genetic algorithm of Ref. [102] includes the operators of crossover, mutation. These are described in more detail in the following sections.

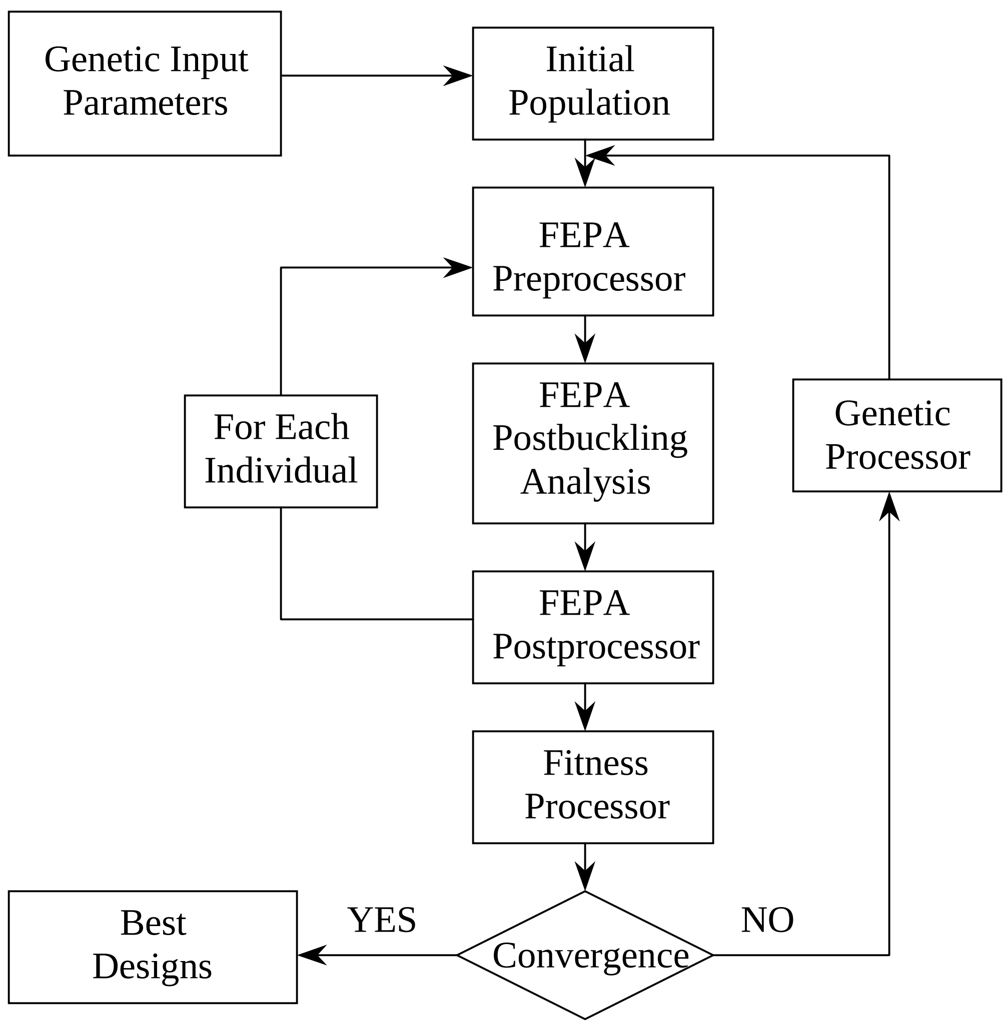

Figure (3.2) The employed genetic algorithm

3.5 Parent Selection

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

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

where $\phi_o = 0$, and $i = 1,\cdots,P$. For example, if there are three panels in a population, the roulette wheel is divided into three pieces with the best panel taking 50% of the wheel, the second best taking 33%, and the poorest taking 17%, see Figure (3.3). A uniformly distributed random number is generated between 0 and 1; panel *i* is selected as a parent if the number lies in the interval $[\phi_{i-1},\phi_i)$. Continuing with the above example, if random numbers $r_1 = 0.3 \in [0,0.5)$ and $r_2 = 0.7 \in [0.5,0.83)$ are drawn, then panel 1 and panel 2 will become parents of the first child, see Figure (3.3). Parents of a child are required to be distinct designs from the population.

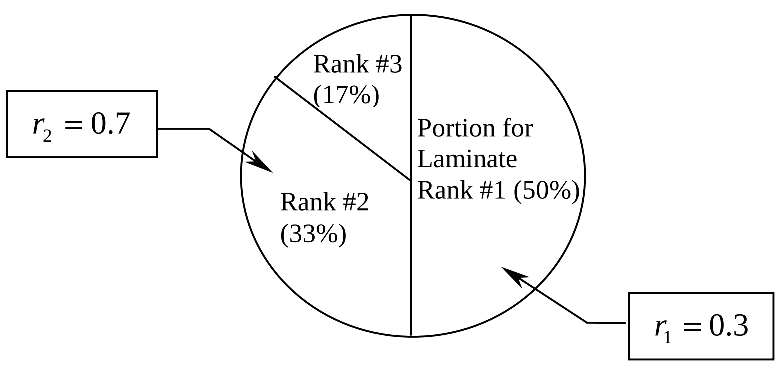

Figure (3.3) Roulette wheel distribution for 3 laminates and parent selection

3.6 Crossover

Children are created by combining a portion of each parent's genetic string by an operation called one-point crossover. To determine the crossover point, a uniformly distributed random number between 0 and 1 is chosen and then multiplied by the maximum number of non-empty genes in the two parents minus one. The integer ceiling value of this product determines the crossover point, as shown in Figure (3.4) . The gene string is then split at the same point in both parents. The left piece from parent 1 and the right piece from parent 2 are combined to form a child panel. To ensure that empty plies are not swapped, all empty plies are pushed to the left side of the coded string (this corresponds to the outer edge of the laminate). The random crossover point is restricted to fall in the non-empty region of both parent laminates to ensure that the child laminate is unique, see Figure (3.4). If during the creation of the child population, crossover is not applied then one of the parent laminates is cloned into the child string. Child laminates are also forced to be distinct from each other and from laminates in the parent population. If a distinct child cannot be found after a prescribed number of iterations, then one of the parents is cloned into the child population also. The crossover process is repeated as many times as necessary to create a new population of panels.

Stacking Sequence Representation

Parent 1: $[E_3 / 20^o_3 \| / 40^o_3 / 10^o_3 / 60^o_3]_s$

Child generated : $[E_3 / 20^o_3 / 50^o_3 / 30^o_3 / 10^o_3]_s$

Parent 2: $[E_3 / E_3 \| / 50^o_3 / 30^o_3 / 10^o_3]_s$

Figure (3.4) One-point crossover

3.7 Mutation

Mutation introduces small changes in children produced by crossover. One of the two children is randomly picked and endures mutation with a low probability P_m. The mutation procedure can change ply orientation as well as delete or add a stack. These operations are illustrated in Figure (3.5). To delete a ply stack, for example, a random number is chosen and the corresponding stack is removed from the stacking sequence by replacing it with an empty ply. The laminate is then re-stacked so that all empty plies are pushed to the outer edge of the laminate, see Figure (3.5b). For more details on the genetic algorithm operators the reader is referred to Soremekun and Gurdal [70].

Before ply addition : $[E_3 / 20^o_3 / 50^o_3 / 30^o_3 / 10^o_3]_s$

After ply addition : $[20^o_3 / 50^o_3 / 30^o_3 / 10^o_3 / 90^o_3]_s$

Figure (3.5a) Ply addition

Before ply deletion: $[90^o_3 / 20^o_3 / 50^o_3 / 30^o_3 / 10^o_3]_s$

After ply deletion: $[90^o_3 / 20^o_3 / E_3 / 30^o_3 / 10^o_3]_s$

After restacking: $[E_3 / 90^o_3 / 20^o_3 / 30^o_3 / 10^o_3]_s$

Figure (3.5b) Ply deletion

Before ply alteration: $[E_3 / 90^o_3 / 20^o_3 / 30^o_3 / 10^o_3]_s$

After ply alteration: $[E_3 / 90^o_3 / 20^o_3 / 50^o_3 / 10^o_3]_s$

Figure (3.5c) Single ply-stack alteration

Figure (3.5) Mutation operator

3.8 Optimization Results

In this section the results obtained for the case study introduced in section 3.2 are presented. Due to the large amount of time required to perform the nonlinear finite element analysis, the average optimization run time was about 15 hours when performed on a PC with a 333 MHz processor. Due to this large optimization time, neither reliability studies nor optimizer tuning runs could be afforded. Instead, values for the different probabilistic operators where obtained from previously published studies dealing with smaller composite optimization problems [70]. It is important to recall here that the main objective of this study is the incorporation of manufacturing and uncertainty information about the geometric imperfections in the design optimization process. Thus an in-depth study of the genetic algorithm optimization method and its application in the case of the nonlinear design of stiffened composite panels is beyond the scope of this study.

In this specific example the imperfections used in the optimization were assumed to have the same profile as the first buckling mode of the corresponding unstiffened panel (calculated using the plate linear buckling module of FEPA) with a maximum amplitude equal to 0.1% of the panel length. Relating the imperfection amplitude to the panel length rather than its thickness fixes its magnitude through out the optimization since the panel length is not a design variable. This is the common way for incorporating the geometric imperfections in the design optimization process (see Chapter 1), it is applied here to allow us to compare the obtained optimum designs with those resulting from the new scheme suggested in this study.

Table (3.2) shows multiple practical optimum and near optimum designs obtained using GA. This table also demonstrates the wealth of design alternatives that can be generated by the GAs. Notice that due to the symmetry of the panel geometry, the applied boundary conditions and loading, and the assumed geometric imperfection shape about the stiffener plane, the optimizer drives the skin stacking sequence to an all zero lay up. This kind of lay up is not practical since such a panel has no shear stress carrying capability. To avoid this problem in the future, a small inplane transverse load is applied along with the longitudinal inplane compression. Finally, it is important to notice that the optimum panel obtained in this example, failed at a much lower load than the required design load when analyzed with a realistic imperfection profile obtained from the manufacturing model presented in the next chapter. Obtaining optimum designs that survive when analyzed with realistic imperfections is one of the main goals of this study. This and other design reliability related issues will be discussed with more details later in this manuscript. The objective of this chapter is only to introduce the reader to Genetic Algorithms and present the traditional way for designing stiffened panels for geometric imperfections.

Table(3.2) Near optimum feasible designs along with their failure load

Panel Mass (Kg)	$P_{failure}$ (Newton)	Plies	Laminate S: skin B: Blade
0.523744	59500	(S)-[6]	$[0_3]_s$
		(B)-[26]	$[-45/0_2/90/0/90/0_6/45]_s$
0.542836	52500	(S)-[6]	$[0_3]_s$
		(B)-[28]	$[45/0_3/\mp 45_2/0_5/90]_s$
0.577408	59500	(S)-[8]	$[0_4]_s$
		(B)-[22]	$[45/0/\mp 45/0/-45/0_4/45]_s$
0.615592	70000	(S)-[8]	$[0_2/90/0]_s$
		(B)-[26]	$[90/0/45/90_2/0/-45/0_5/45]_s$
0.615592	66500	(S)-[8]	$[0_4]_s$
		(B)-[26]	$[-45/90/0/90/0/\pm 45/0_2/45/0_3]_s$
0.65016	59500	(S)-[10]	$[90/0_3/90]_s$
		(B)-[20]	$[90_2/0_3/90/\mp 45/90/0]_s$
0.66925	56000	(S)-[10]	$[-45/0_3/45]_s$
		(B)-[22]	

	$[90/45/90/0_2/90/\mp 45/90/0/90]_s$