

## OPTIMIZATION ALGORITHMS FOR LAMINATED COMPOSITE PANELS USING NONLINEAR MODAL FINITE ELEMENTS

KRZYSZTOF BOGDANSKI

Faculty of Power and Aeronautical Engineering, Warsaw University of Technology

Ul. Nowowiejska 24, 00-665 Warszawa Poland

kbogdanski@meil.pw.edu.pl

JEAN-MICHEL DHAINAUT

Mechanical Engineering Department, Embry-Riddle Aeronautical University,

600 S Clyde Morris Blvd, Daytona Beach, FL 32114, USA

dhain460@erau.edu

ANDRES JOSEPHSOHN

San Andres School, Buenos-Aires, Argentina

Roque Saenz Pena 601, Buenos Aires, B1636FHA, Argentina

andres.josephsohn@sanandres.esc.edu.ar

[Received: February 2, 2016, Accepted: May 12, 2016]

**Abstract.** A methodology for minimization of composite panels deflections and stresses that uses a time domain nonlinear modal finite element model with two different optimization algorithms (genetic and DB algorithms) is described. The nonlinear modal formulation is based on geometrical nonlinearities rather than material nonlinearities, which does not require updating of the stiffness matrix at each time step, making it extremely time efficient when compared to commercial finite element softwares. Optimization algorithms are implemented in Matlab and can be used either with the finite element code itself or as a post-processing option. The method is applied to rectangular 10-ply symmetrically laminated plates under uniform pressure loads, with simply supported and clamped boundary conditions. The design constraints are based on the Tsai-Wu failure criterion. Results of the optimization using genetic algorithm include the influence of the initial size of population and number of generations. The DB algorithm proposed by the authors is shown to be more effective for the presented examples than the genetic algorithm.

*Keywords:* Composite optimization, lay-ups, stacking sequences, genetic algorithm, DB algorithm

### 1. NOMENCLATURE

#### Latin Symbols

- $a, b$  panel dimensions,
- $b$  indicates that a vector (or matrix) depends on bending
- $mb$  indicates that a vector (or matrix) depends on couples bending membrane ( $mb=bm$ )
- $E$  modulus of elasticity
- $G$  shear modulus of elasticity

---

$h$	panel thickness
$m$	indicates that a vector (or matrix) depends on membrane
$z$	distance from neutral axis
$[K]$	linear stiffness matrices
$[K1]$	first order stiffness matrices (they depend on $\{W_b\}$ )
$[K2]$	second order stiffness matrices (they depend on $\{W_b\}$ )
$[M]$	mass matrices
$P(t)$	time dependent load vector
$u, v$	in-plane displacements
$W$	transverse panel deflection.

### Greek Symbols

$\rho$	panel density
$\nu$	Poisson's ratio
$\zeta$	damping coefficient

## 2. INTRODUCTION

The latest commercial aircraft from Boeing and Airbus, the 787 and the A350, first flew in 2009 and 2013, respectively. Both represented a technological step change by having structures composed of up to 50% of composite materials. They are only two examples of a trend that indicates that the industry will rely more and more on the beneficial properties of composites. The best mechanical properties of composite materials are obtained from laminates composed of many layers of unidirectional fibers embedded in a polymer matrix. These plies are distributed with different orientations, allowing the laminates properties to be tailored to the specific structural application. Hence, it is clear that to extract the maximum performance out of these materials, efficient stacking sequence optimization techniques are needed. Different optimization techniques based on Lagrangian minimization principle exist, however, in this work, the technique that will be explored is that of genetic algorithms (GA), which relies on evolutionary principles developed by Holland [1]. GA solve the problem of optimization by following a process analogous to what is observed in nature, where only the mutations that give an advantage to the individual are propagated to the new generations. The mechanics of GA is based on operations that result in structured, yet randomized exchange of genetic information between individual strings (chromosome strings) of the parents and consists of crossover, mutation and inversion [2]. For the sake of brevity, only the genetic representations for optimization of structural design will be discussed here. Goldberg and Samtani [3] were the first to use GA in structural design, solving the weight optimization problem of a 10 bar truss. Different variants of GA widely used in various structural problems have already been discussed in detail [4]. More specifically, multiple authors [5–7] combined commercially available finite element software with GA in the design of composite structures. However, it has been remarked that the major limitation in this application stems from the generality of the finite element package, i.e., those programs can be adapted to analyze composites but that is not their main objective. Hence, the novelty of the present work is that the authors would be using their own modal finite element program that is able

to accurately predict the linear and nonlinear response of composite beams, plates, and shells subjected to different loading conditions at a reasonable computational cost. Accurate prediction of behavior of aircraft panels to different loading conditions is essential for design. Traditional design and analysis methods for commercial and military aircraft [8] [9] are based on linearization and simplified loading conditions. However, experimental data have shown that, in many instances, the loads induce large nonlinear deflection responses. Table 1 shows the non-dimensional displacement ( $W_{\max}/h$ ) and the maximum normal stress for linear and nonlinear analysis of composite panels with varying stacking sequences when subjected to a uniform load of 1,500 Pa. It can be seen that the linear analysis overpredicts the responses, something that will result in a heavier, overbuilt structure.

Table 1. Comparison of linear and nonlinear deformations and normal stresses

Stacking	Deformation $W_{\max}/h$		Normal stress $\sigma$ [MPa]	
	Linear	Non-linear	Linear	Non-linear
0 0 0 0 0 0 0 0 0	1.067	0.534	33.58	12.63
-45 0 0 0 0 0 0 0 -45	1.283	0.616	140.19	40.54
-45 45 -45 45 -45 -45 45 -45 45 -45	1.302	0.765	153.43	59.30
45 90 90 90 45 45 90 90 90 45	2.375	0.989	265.95	61.10

$h$  is the panel thickness

To the authors' best knowledge, all commercial finite element packages study geometrical nonlinear deformations at the structural degree of freedom, something that is extremely computationally expensive because all nonlinear matrices need to be re-assembled at each time of the solution by considering deformations from the previous step. This is a severe obstacle in optimization that is intended to be overcome with the approach presented in this paper.

### 3. FORMULATION

This section introduces the fundamentals of the nonlinear modal formulation and genetic algorithms.

**3.1. Nonlinear modal finite element.** The governing equations of motion (EOM) for a two-dimensional panel subject to thermal and time dependent loading are developed. First, the EOM are expressed in terms of structural degrees of freedom (DOF) or in physical coordinates. Then the system level EOM are transformed into modal coordinates based on the expansion theorem. The plate is subjected to load intensities driving the response into the nonlinear range. The von Karman large deflection plate theory is employed here to describe the nonlinear strain and displacement relationships.

**3.1.1. Equations of motion in structural degrees of freedom.** The 24-DOF Bogner-Fox-Schmit C conforming rectangular plate element is used for the finite element model. Considering large amplitude deflections, i.e., that the transverse displacement of the

panel is of the same order of magnitude as the panel thickness, the in-plane membrane response becomes coupled with the transverse bending. As the plate bends, the middle surface stretches and significant membrane forces develop. The load-transverse deflection response becomes nonlinear. The von-Karman plate theory addresses the above in-plane extension effects by introducing additional quadratic terms to the strains developed in a vibrating plate. The von-Karman nonlinear strain-displacement relationship is given by

$$\{\varepsilon\} = \begin{Bmatrix} u_{,x} \\ v_{,y} \\ u_{,y} + v_{,x} \end{Bmatrix} + \frac{1}{2} \begin{Bmatrix} w_{,x}^2 \\ w_{,y}^2 \\ 2w_{,x}w_{,y} \end{Bmatrix} + z \begin{Bmatrix} -w_{,xx} \\ -w_{,yy} \\ -2w_{,xy} \end{Bmatrix} = \{\varepsilon_m^0\} + \{\varepsilon_\theta^0\} + z\{\kappa\} \quad (1)$$

where  $\{\varepsilon_m^0\}$  is the linear membrane strain vector,  $\{\varepsilon_\theta^0\}$  is the von Karman nonlinear membrane strain vector, and  $z\{\kappa\}$  is the bending strain vector.

When the principle of virtual work and the finite element expressions are combined, the assembled governing EOM for the panel subjected to aerodynamic, thermal, and acoustic excitations are derived and expressed as

$$\begin{aligned} & \begin{bmatrix} M_b & 0 \\ 0 & M_m \end{bmatrix} \begin{Bmatrix} \ddot{W}_b \\ W_m \end{Bmatrix} + \\ & \left( \begin{bmatrix} K_b & K_{bm} \\ K_{mb} & K_m \end{bmatrix} - \begin{bmatrix} K_{N\Delta T} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} K1_{Nm} + K1_{Nb} & K1_{bm} \\ K1_{mb} & 0 \end{bmatrix} + \begin{bmatrix} K2 & 0 \\ 0 & 0 \end{bmatrix} \right) \begin{Bmatrix} W_b \\ W_m \end{Bmatrix} = \\ & = \begin{Bmatrix} P_{b\Delta T} \\ P_{m\Delta T} \end{Bmatrix} + \begin{Bmatrix} P_b(T) \\ 0 \end{Bmatrix} \quad (2) \end{aligned}$$

or

$$[M] \{\ddot{W}\} + ([K_o] - [K_{N\Delta T}] + [K1] + [K2]) \{W\} = \{P\} \quad (3)$$

where  $\{P\} = \{P_{b\Delta T} + P_b(t)\}$ . If the membrane inertia term  $\{\ddot{W}_m\}$  is neglected, the in-plane displacement vector can be expressed in terms of the bending displacement as

$$\begin{aligned} \{W_m\} &= [K_m]^{-1} \{P_{m\Delta T}\} - [K_m]^{-1} [K_{mb}] \{W_b\} - \\ & - [K_m]^{-1} [K1_{mb}] \{W_b\} = \{W_m\}_0 + \{W_m\}_1 + \{W_m\}_2 \quad (4) \end{aligned}$$

where  $\{W_m\}_0 = [K_m]^{-1} \{P_{m\Delta T}\}$  is a constant matrix in function of the linear membrane matrix  $[K_m]$  and the constant membrane in-plane load  $\{P_{m\Delta T}\}$ ;  $\{W_m\}_1 = -[K_m]^{-1} [K_{mb}] \{W_b\}$  is in function of the constant membrane matrix  $[K_m]$ , the constant coupling bending-membrane matrix  $[K_{mb}]$  and the linear vertical bending deformation  $\{W_b\}$ ;  $\{W_m\}_2 = -[K_m]^{-1} [K1_{mb}(W_b)] \{W_b\}$  is in function of the constant matrix  $[K_m]$ , the first order nonlinear matrix  $[K1_{mb}]$  which depends on  $\{W_b\}$  and the bending deformation  $\{W_b\}$ . Consequently, the term  $\{W_m\}_2$  quadratically ( $\{W_b\}^2$ ) depends on  $\{W_b\}$ . Thus  $[K1_{Nm}]$  which depends on  $\{W_m\} = \{W_m\}_0 + \{W_m\}_1 + \{W_m\}_2$  has three components  $[K1_{Nm}^0]$  evaluated with  $\{W_m\}_0$ ,  $[K1_{Nm}^1]$  evaluated with  $\{W_m\}_1$  and  $[K1_{Nm}^2]$  evaluated with  $\{W_m\}_2$ . Recalling that  $\{W_m\} =$

$\{W_m\}_0 + \{W_m\}_1 + \{W_m\}_2$  it should be clear that  $\{W_m\}$  is the sum of a constant term  $\{W_m\}_0$ , and two terms that depends on the bending deformation  $\{W_b\}$  only. The term  $\{W_m\}_1$  depend linearly on  $\{W_b\}$  while the term  $\{W_m\}_2$  depends quadratically on  $\{W_b\}$ . Next, by substituting  $\{W_m\}$  which is now in function of  $\{W_b\}$  into equation (2), the system EOM can be written in terms of the bending deformation  $\{W_b\}$  only

$$[M_b] \left\{ \ddot{W}_b \right\} + ([K_L] + [K1(W_b)] + [K2(W_b^2)]) \{W_b\} = \{F\} \quad (5)$$

where

$$[K_L] = [K_b] - [K_{N\Delta T}] + [K1_{bm}] [K_m]^{-1} [P_{m\Delta T}] - [K1_{bm}] [K_m]^{-1} [K_{mb}] + [K1_{Nm}^0], \quad (6a)$$

$$[K1] = [K1_{Nb}] + [K1_{Nm}^1] - [K_{bm}] [K_m]^{-1} [K1_{mb}] - [K1_{bm}] [K_m]^{-1} [K_{mb}], \quad (6b)$$

$$[K2] = [K2_b] + [K1_{Nm}^2] - [K1_{bm}] [K_m]^{-1} [K1_{mb}] \quad (6c)$$

and

$$\{F\} = \{P_{b\Delta T}\} + \{P_b(t)\} - [K_{bm}] [K_m]^{-1} \{P_{m\Delta T}\}. \quad (6d)$$

For more details on the derivation of the nonlinear modal formulation readers can consult [10]

**3.1.2. Equations of motion in modal degree of freedom.** Equation (5) is expressed in terms of structure DOF, which means a costly computational burden since the governing equations increase proportionally with the number of elements. This obstacle is resolved by transforming the system EOM in the structural DOF into a set of truncated modal coordinates by expressing the response as a linear combination of some base functions

$$\{W_b\} = \sum_{r=1}^n q_r(t) \{\phi_r\} = [\Phi_b] \{q\} \quad (7)$$

where  $q_r(t)$  denotes the modal coordinates of the  $r$ -th mode, which reflects the contribution to the total deflection from the  $r$ -th mode;  $\Phi_b = [\phi_1, \phi_2, \dots, \phi_r, \dots, \phi_n]$  is the modal matrix, in which  $\phi_r$  is the mode shape due to the transverse bending of the  $r$ -th mode obtained from the linear vibration analysis.

The transformation of system governing EOM into modal coordinates is achieved by substituting equation (6) into equation (5) and pre-multiplying the whole equation by  $[\Phi_b]^T$ . If structural modal damping in the form of  $\zeta_r$  ( $r = 1, 2, \dots, n$ ) is assumed the transformed EOM in modal coordinates can be written in a brief form as follows:

$$[\bar{M}_b] \{\ddot{q}\} + 2\zeta_r \frac{\omega_r}{\omega_0^2} [\bar{M}_b] \{\dot{q}\} + ([\bar{K}_L] + [\bar{K}_q] + [\bar{K}_{qq}]) \{q\} = \{\bar{F}\} \quad (8)$$

Here the diagonal mass matrix is of the form

$$[\bar{M}_b] = [\Phi_b]^T [M_b] [\Phi_b] \quad (9)$$

The linear, quadratic, and cubic stiffness matrices are as follows:

$$\begin{aligned} [\bar{K}_L] = & [\Phi_b]^T ([K_b] - [K_{N\Delta T}]) [\Phi_b] + \\ & + [\Phi_b]^T \left( [K1_{bm}] [K_m]^{-1} [P_{m\Delta T}] \right) [\Phi_b] - [\Phi_b]^T \left( [K1_{bm}] [K_m]^{-1} [K_{mb}] \right) [\Phi_b] + \\ & + [\Phi_b]^T ([K1_{Nm}^0]) [\Phi_b] , \quad (10) \end{aligned}$$

$$\begin{aligned} [\bar{K}_q] = & [\Phi_b]^T \left( \sum_{r=1}^n q_r \left( [K1_{Nb}]^{(r)} + [K1_{Nm}^1]^{(r)} \right) \right) [\Phi_b] - \\ & - [\Phi_b]^T \left( \sum_{r=1}^n q_r \left( [K_{bm}] [K_m]^{-1} [K1_{mb}]^{(r)} \right) \right) [\Phi_b] - \\ & - [\Phi_b]^T \left( \sum_{r=1}^n q_r \left( [K1_{bm}]^{(r)} [K_m]^{-1} [K_{mb}] \right) \right) [\Phi_b] , \quad (11) \end{aligned}$$

$$\begin{aligned} [\bar{K}_{qq}] = & [\Phi_b]^T \left( \sum_{r=1}^n \sum_{s=1}^n q_r q_s \left( [K2]^{(rs)} + [K1_{Nm}^2]^{(rs)} \right) \right) [\Phi_b] - \\ & - [\Phi_b]^T \left( \sum_{r=1}^n \sum_{s=1}^n q_r q_s \left( -[K1_{bm}]^{(r)} [K_m]^{-1} [K1_{mb}]^{(s)} \right) \right) [\Phi_b] \quad (12) \end{aligned}$$

The modal load vector is given by:

$$\begin{aligned} \{\bar{F}\} = & [\Phi_b]^T \{P_{b\Delta T}\} + [\Phi_b]^T \{P_b(t)\} - \\ & - [\Phi_b]^T [K_{bm}] [K_m]^{-1} \{P_{m\Delta T}\} . \quad (13) \end{aligned}$$

**3.2. Genetic algorithms.** The application of GA operators to a problem first requires the representation of the possible combinations of the variables in terms of bit strings, analogous to chromosomes in biological genetics [2]. In order to increase the speed of the genetic search the following assumptions are considered: i) the stacking sequences are limited to using  $0^\circ$ ,  $\pm 45^\circ$  and  $90^\circ$  plies; ii) the laminate thickness can only be an integer multiple of the standard ply thickness; iii) the laminate is symmetric. The initial population set is generated by a function that generates pseudo-random numbers uniformly and, to generate future generations, parents are selected from the initial population based on their fitness. Next, genetic operators are used to create new stacking sequences: the children population. Good features from the initial population set propagate to the children population by using a biased roulette wheel where better parents are assigned a larger area. A description of the genetic operators for the construction of new generations is given next [7].

*Crossover operator.* Once a pair of parents is selected, the children are generated by combining information from both parents, for example, by splicing the left part

of the string of one of the parents with the right part of the string from the other parent.

Parent 1:  $[90/0/45]_s$

Parent 2:  $[45/90/0]_s$  possible child designs are:

Child:  $[90/90/0]_s$ ,  $[90/0/0]_s$ ,  $[45/0/45]_s$ ,  $[45/90/45]_s$

It is important to note that the present work utilizes a modified crossover algorithm given the constraints that all laminates must be symmetric.

*Mutation operator.* Mutation performs the valuable task of preventing premature loss of important genetic information by introducing random alteration in the child string obtained by the crossover operator. Inferior designs may have some good traits that would get lost in the gene pool (roulette wheel) when the parents are not selected. For example if you only consider the first child from the crossover  $[90/90/45]_s$  a possible mutation is  $[90/0/45]_s$ , which may lead to a good design but that would not be accounted for by the crossover operator.

The implementation of the GA was done in Matlab®. Due to the assumptions mentioned in Section 2.2, the number of possible solutions (stacking sequences) is limited and is generated as a matrix.

To fully test the program, ensuring that it fulfills all of the assumptions, the results for each layup are calculated and stored in a separate matrix. In the final version of the program, this approach can be easily replaced by real time calculations.

In the first generation, a specified number (size of population) of indices (corresponding to their chromosomes) are selected using random permutations. Selected chromosomes form the population and the random permutations used ensure the selection of unique (non-repeating) sets. The results for every chromosome are then sorted by increasing values of the Tsai-Wu failure criterion on the outer (first) layer of the laminate. That forms the fitness function. The best two from the list are then selected as parents and will be used in crossover and mutation operations. Afterwards, parents are crossed over with the rest of the population forming offspring. Subsequently, these two parents are mutated and form new sets of chromosomes. From parents, offspring and mutated chromosomes, a list of unique stacking sequences is formed and the corresponding results are extracted from the result matrix. The fitness function (value of Tsai-Wu for the outer layer) is evaluated and the ordered results constitute the first generation. Subsequent generations are generated in a similar way but, instead of randomly choosing chromosomes at the start, the best two from the previous generation become parents and the rest are used for crossover operation.

**3.3. DB algorithm.** The authors propose a DB algorithm that reduces the optimization computational cost. It does not depend on either the size of the populations nor on the number of generations. The advantage of such an approach will be explained in the results section.

At first only one stacking sequence is randomly chosen. Since the ply orientations are limited to only four angles, i.e.,  $0^\circ$ ,  $\pm 45^\circ$  and  $90^\circ$ , the chosen layup is replicated three times. In the next step, a randomly chosen layer is replaced in the replicated

layups with all the other possible angle ply orientations.

Example:

Randomly chosen stacking: [90/0/45]

Newly formed stacking sequences with the third layer replaced: [90/0/0], [90/0/-45], [90/0/90].

According to the fitness function (the same as the one used in genetic algorithm: value of Tsai-Wu for the outer layer), the best layup is chosen from the four stacking sequences. This best stacking is then replicated again three times and a different layer is now replaced with the other ply orientations. That process is repeated until all of the layers have been replaced and the very best out of all stacking sequences is chosen. The number of calculations is equal to:  $N_{o\text{calc}} = L(A - 1) + 1$  where  $L$  is the number of layers and  $A$  is the number of possible angle ply orientations. In the present case,  $L = 5$  and  $A = 4$ , giving  $N_{o\text{calc}} = 16$ . Using the same matrices with stacking sequences and results a program using the DB algorithm has been written using Matlab®.

#### 4. RESULTS

**4.1. Material properties.** Composite and isotropic panels are used to validate the finite element code and the DB algorithm. The material properties of isotropic aluminum panels are  $E_1 = 73$  GPa,  $\nu = 0.30$ ,  $\rho = 2750$  kg/m<sup>3</sup> and of composite Graphite/Epoxy T300/5208  $E_1 = 181.0$  GPa,  $E_2 = 10.3$  GPa,  $G_{12} = 7.17$  GPa,  $G_{23} = 6.21$  GPa,  $\nu_{12} = 0.28$  and  $\rho = 1550$  kg/m<sup>3</sup>.

A proportional damping ratio of  $\zeta_r \omega_r = \zeta_s \omega_s$  with  $\zeta_1 = 0.02$  is used for transient responses. For validation purposes, simply-supported (S-S) boundary conditions are considered and for generalization clamped (C) boundary conditions are considered as well. For all support cases the maximum deformation occurs at the center of the panel for the uniform loading but the maximum stress depends on the support conditions. For instance, maximum stress occurs at center of the plate for SS but at the midpoint of the longest edge for C conditions. Preliminary mesh convergence and modal convergence was verified for all composite panels. It was found that eight modes were sufficient for converged stress responses on a  $24 \times 24$  mesh. The validation of the nonlinear modal finite element code and of the GA can be done separately since they are completely independent of each other. The former is only used to evaluate the responses, and the GA is simply a numerical procedure used for finding the optimum stacking sequence but does not depend on the linearity of the responses.

**4.2. Validation.** The validation of the present nonlinear modal FE formulation, equation (7), is performed by comparing results for isotropic SS panels obtained with classical formulation [11]. The linear and nonlinear stiffness matrices will be verified by solving the single mode linear and nonlinear free vibration problems, respectively. Table 2 shows the non-dimensional frequency parameter  $\omega a \sqrt{\rho(1 - \nu^2)}/E_1$  which neglects the in-plane inertia and characterizes the linear free vibration behavior by the fundamental linear bending mode (1,1) only. Results are obtained for different shell geometries ranging from spherical shells ( $R_x/R_y = 1$ ) to hyperbolic paraboloids



( $R_x/R_y = -1$ ). The curved panels ( $b/a = 1$ ) are modeled by an  $8 \times 8$  mesh on a quarter shell, more than adequate for modeling the fundamental mode (1,1). The validation was performed on curved and cylindrical panels because no data for flat panels was available.

Table 2. Non-dimensional frequency parameter  $\omega a \sqrt{\rho(1 - \nu^2)/E_1}$

	$\omega a \sqrt{\rho(1 - \nu^2)/E_1}$	
$R_x/R_y$	Ref. [11]	F.E.
-1.0	0.05695	0.05831
-0.5	0.06174	0.06251
0.0	0.07429	0.07471
0.5	0.09144	0.09178
1.0	0.1111	0.1115
$R_x = R_y$		
3.048	0.1111	0.1115
5.08	0.07429	0.08096
12.7	0.06007	0.006145
25.4	0.05776	0.005813
$H = h/a$		
0.005	0.09955	0.09901
0.01	0.1111	0.1115
0.02	0.1485	0.1489
0.1	0.5622	0.5778
0.2	1.033	1.1437

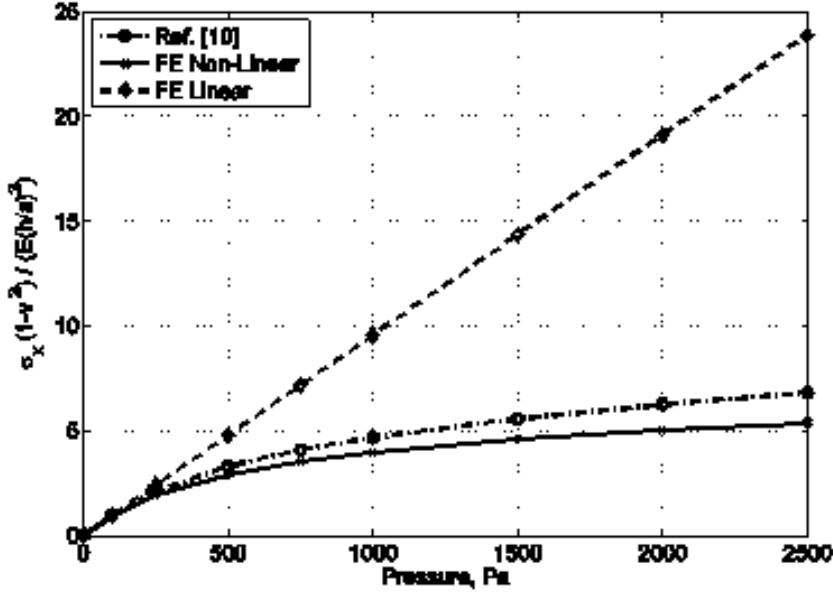


Figure 1. Non-dimensional stress vs. uniform pressure

It can be observed that results compare very well, except for the very thick shell case ( $h/a = 0.2$ ) where the difference is 10.7%.

The stress convergence was based on stresses rather than displacements because the Tsai-Wu failure criteria used for the optimization is based on stresses. The same mesh size and number of modes will be employed in subsequent sections.

This difference is due to the fact that the present modal FE formulation neglects shear effects, negligible for thin-shells but significant in thick ones. The uniform pressure loading is verified on a  $0.254 \times 0.381 \times 0.00127$  m ( $12 \times 15 \times 0.05$  in.) flat panel modeled with a  $20 \times 20$  mesh on a full panel. Isotropic material properties and simply-supported boundary conditions are assumed. Figure 1 shows the non-dimensional stress at different uniform load intensities.

**4.3. Results. Genetic algorithm.** The GA algorithm was tested on several different population sizes with the following assumptions: i) stacking sequences consisted of 10 layers (5 symmetrical), ii) there are 1024 possible (unique) stacking sequences. Calculations have been performed for both simply supported and clamped boundary conditions

Since all the stacking sequences were calculated, it was found that for the applied load and simply supported boundary conditions, the highest value of Tsai-Wu for one layup was more than twenty times greater than the lowest. The distribution of the results in comparison to the lowest value is shown in Figure 2. The solution is treated as converged if the result of the Tsai-Wu for the outer layer does not differ more than 5% from the best result of the 1024 stacking sequences. This results in a list consisting of 24 solutions (2.34% of the total population).

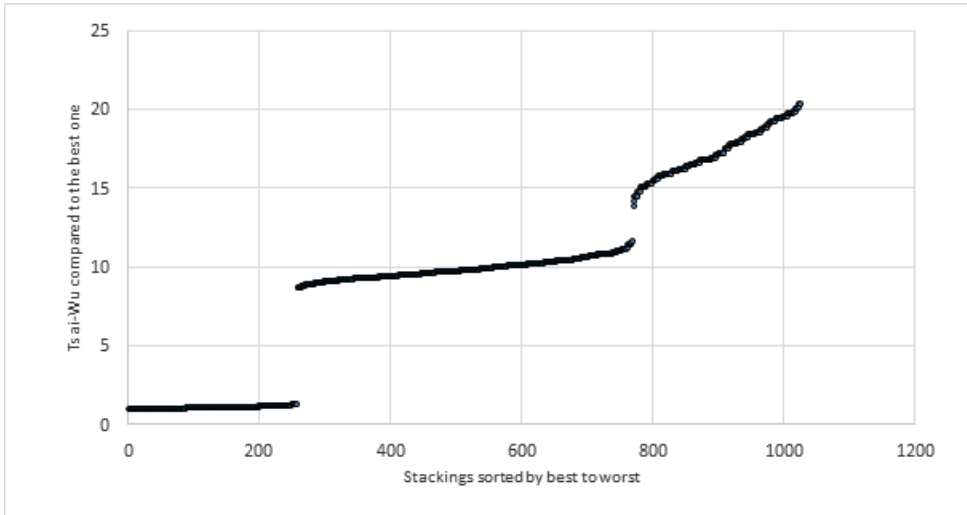


Figure 2. Distribution of results for simply supported boundary conditions in comparison to the best result

Table 2 presents the results for simply supported boundary conditions of the stacking sequences generated using the GA sorted by each generation and population size. Each of the results has been calculated using statistics based on 40 samples. Number of solutions means how many unique calculations have to be performed after every generation. The adjacent column contains the percentage of the results obtained after every generation that belong to the set of the 24 best results. The least number of calculations performed in order to find the best results was 46 with the population of 15 after the fourth generation. A 90% chance of achieving the best solution is possible after 30 calculations with the population size of 10 after the fifth generation. The same efficiency can be obtained after the second generation with a 50% larger population. However, the computational cost is 23% higher.

Table 3. Percentage of 24 best results (2.3% of all possible) in each generation sorted by population size

Population size	5		10		15	
Generation	Number of solutions	%	Number of solutions	%	Number of solutions	%
1	11	25	13	45	27	85
2	15	35	20	50	37	90
3	18	65	23	70	47	90
4	20	75	27	70	46	100
5	22	85	30	90	47	100
6	24	85	33	95	49	100
7	25	85	35	95	50	100

Population size	20		25		30	
Generation	Number of solutions	%	Number of solutions	%	Number of solutions	%
1	26	75	40	85	37	95
2	34	85	53	100	52	100
3	40	95	62	100	58	100
4	46	95	68	100	63	100
5	50	100	70	100	67	100
6	53	100	73	100	70	100
7	56	100	75	100	73	100

The results in Table 3 clearly show that the larger the population size, the higher the probability of getting the best answer in that generation. However, due to the larger size, the number of solutions needed to be calculated grows rapidly. To get a 90% chance of getting the solution, the lowest number of solutions is 30 with a population size of 10 after the fifth generation. The same efficiency can be obtained after the second generation with a 50% larger population but the computational cost is 23% higher.

Table 4 shows how the probability of finding the best solution decreases when for convergence the best 1% of the results is selected as opposed to 2.3%. It is worth

noting that with a population size of 30, all of the results were in the best 1% of the results after the second generation.

Table 4. Comparison of percentage of 10 and 24 best results in each generation sorted by population size

Population size	10		20		30	
Generation	Best 10 [%]	Best 24 [%]	Best 10 [%]	Best 24 [%]	Best 10 [%]	Best 24 [%]
1	25	45	55	75	80	95
2	30	50	70	85	100	100
3	35	70	80	95	100	100
4	45	70	85	95	100	100
5	65	90	90	100	100	100
6	80	95	95	100	100	100
7	90	95	95	100	100	100

Table 5. Percentage of 9 best results (0.88% of all possible) for fixed boundary conditions in each generation sorted by population size

Population size	5		10		15	
Generation	Number of solutions	%	Number of solutions	%	Number of solutions	%
1	12	15	16	15	34	30
2	17	20	23	25	46	45
3	20	20	28	30	52	60
4	22	35	33	45	55	60
5	23	40	35	50	59	70
6	24	45	37	55	62	70
7	26	60	38	55	63	75

Population size	20		25		30	
Generation	Number of solutions	%	Number of solutions	%	Number of solutions	%
1	29	40	48	60	41	65
2	42	60	64	65	55	80
3	50	65	71	65	63	90
4	56	70	74	70	70	95
5	59	70	76	80	74	95
6	62	70	82	85	79	100
7	65	75	84	85	82	100

The second tested example differed by changing the boundary conditions from simply supported to clamped. Using the same criterion that the result of the Tsai-Wu for the outer layer does not differ more than 5% from the best result of the 1024 stacking sequences, the results list consists of only 9 solutions (0.9% of the total

population). Table 5 presents the results for fixed boundary conditions using GA. In comparison to the simply supported boundary conditions the results are worse due to the lesser number of results considered for convergence. A 90% chance of achieving the best solution is possible with the population size of 30 after 63 calculations and three generations. In order to have a 100% chance of getting the solution converged, the number of calculations has to increase up to 79 (by 25.4%).

**4.3.1. DB Algorithm.** Since the DB algorithm is simpler because it is not influenced by the number of generations or population size, the analysis of the results is significantly faster.

The algorithm was run 10,000 times and in 80.6% of the calculations the best result, out of 1,024, was obtained. The two best results for simply supported boundary conditions were obtained in 100% of the calculations. Compared to the presented GA, 16 unique calculations would result in a maximum 45% chance of finding a solution in the best 24 possible results, whereas there is a 100% chance of getting one of the best two results with the DB algorithm. For fixed boundary conditions, the DB algorithm would give a 72.3% chance of converging (one of the best 9 results). Using the GA, 16 unique calculations would only give a 15% chance of achieving the converging solution. It is worth mentioning that the worst result obtained for the DB algorithm was the 72nd best answer and only appeared in 0.18% of all of the performed calculations.

## 5. CONCLUSIONS

An efficient nonlinear finite element modal formulation was combined with two different optimization algorithms. It was shown that the newly developed DB algorithm is more effective than the most popular GA for the studied cases.

For the first case (simply supported boundary conditions) and given assumptions, the GA allows finding a result in the best 1% of all of the results, with about 90% efficiency, by calculating about 3% of all of the possible solutions. By reducing the efficiency to 85%, the computational costs only drops to requiring the calculation of 2.2% of the solutions. Since the GA is basically a semi-controlled random process, it does not guarantee finding the best answer in the process. On the other hand, the DB algorithm gave a 100% chance of finding one of the best two solutions after only 16 calculations (1.6% of all possible stacking sequences) and in 80% of cases it gave the best possible solution.

The second tested case (fixed boundary conditions) also showed better results gained from the DB algorithm compared to the GA. It has given a 72.3% chance of convergence compared to only 15% for the GA after 16 unique calculations. A 100% chance of finding a result in the best 1% of all of the results is possible using GA, but it takes on average 79 unique calculations. In future work, the authors will keep on testing and developing the DB algorithm, including increasing the number of layers and applying more complex loads.

## REFERENCES

1. HOLLAND, J. H.: *Adaptation of Natural and Artificial Systems*. MIT Press, 1992. ISBN: 978-0-262-08213-6.
2. GÜRDAL, H. R. T., Z. and HAJELA, P.: *Design Optimization of Laminated Composite Materials*. Wiley Inter-Science, 1999. ISBN: 0-471-25276.
3. GOLDBERG, D. E. and SAMTANI, M. P.: Engineering optimization via genetic algorithm. In *Proceedings of the 9th Conference on Electronics Computation, February 23-26*, American Society of Civil Engineers, 1986, pp. 471–482. ISBN-13: 978-0872625129.
4. HAFTKA, R. T.: *Genetic Algorithms for Optimization of Composite Laminates*, pp. 431–442. In *Mechanics of Composite materials and Structures*, NATO Sc. Series, Vol. 361., Editor: C. A. M. Soares, Kluwer Academic Publisher, 1999.
5. MUC, A. and GURBA, W.: Genetic algorithms and finite element analysis in optimization of composite structures. *Composite Structures*, **54**, (2001), 275–281.
6. WALKER, M. and SMITH, R. E.: A technique for multi-objective optimization of laminated composite structures using genetic algorithms and finite element analysis. *Composite Structures*, **62**, (2003), 123–128.
7. ALMEIDA, F. S. and AWRUCH, A. M.: Design optimization of composite laminated structures using genetic algorithms and finite element analysis. *Composite Structures*, **88**, (2009), 443–454.
8. HOLEHOUSE, I.: *Sonic Fatigue Design Techniques for Advanced Composites Aircraft Structures*. Technical report: AFWL-TR-80-3019, Rohr Industries, P.O. Box 878, Chula Vista, California, 92012, 1980. 1-190.
9. RUDDER, F. F. and PLUMBEE, H. E.: Sonic Fatigue Design Guide for Military Aircraft. Technical report: AFFDL-TR-74-112, AF Flight Dynamics Laboratory, Wright Petterson Air Force Base, Ohio 45433, 1975. 1-571.
10. DHAINAUT, J.-M.: Nonlinear Response and Fatigue Estimate of Surface Panels to White and Non-White Random Exitations. Ph.D. thesis, Old Dominion, Norfolk, VA, 2001.
11. LOCKE, J. E.: A Finite Element Formulation for the Large Deflection Random Response of Thermally Buckled Structures. Ph.D. thesis, Old Dominion University, Norfolk, VA, 1988.