A Two-Level Procedure for the Global Optimum Design of Composite Modular Structures-Application to the Design of an Aircraft Wing: Part 2: Numerical Aspects and Examples

Marco Montemurro, Angela Vincenti, Paolo Vannucci

To cite this version:

Marco Montemurro, Angela Vincenti, Paolo Vannucci. A Two-Level Procedure for the Global Optimum Design of Composite Modular Structures-Application to the Design of an Aircraft Wing: Part 2: Numerical Aspects and Examples. Journal of Optimization Theory and Applications, 2012, 155 (1), pp.24–53. 10.1007/s10957-012-0070-1 . hal-01666669

HAL Id: hal-01666669
https://hal.science/hal-01666669v1

Submitted on 13 Mar 2024

A Two-Level Procedure for the Global Optimum Design of Composite Modular Structures—Application to the Design of an Aircraft Wing

Part 2: Numerical Aspects and Examples

Marco Montemurro, Angela Vincenti, Paolo Vannucci

Abstract This work concerns a two-level procedure for the global optimum design of composite modular structures. The case-study considered is the least weight design of a stiffened wing-box for an aircraft structure. The method is based on the use of the polar formalism and on a genetic algorithm. In the first level of the procedure, the optimal structure is designed as composed by a single equivalent layer, while a laminate realizing the optimal structure is found in the second level. The method is able to automatically find the optimal number of modules; no simplifying assumptions are used and it can be easily generalized to other problems. The work is divided into two parts: the theoretical formulation in the first part, the genetic procedure and some numerical examples in this second one.

Keywords Laminates · Composite materials · Polar method · Genetic algorithms

1 Introduction

In this second part of the research, we consider the numerical aspects of the procedure for the optimization of modular composite structures presented in the first part. Namely, the algorithm used for the solution of the two-level optimization problems

---

M. Montemurro · A. Vincenti
Institut d'Alembert UMR7190 CNRS, Université Pierre et Marie Curie, 4, Place Jussieu, 75252
Paris, France

M. Montemurro
Centre de Recherche Public Henri Tudor, Luxembourg, Luxembourg

P. Vannucci (🖂)
Université de Versailles et Saint Quentin en Yvelines, 45, Avenue des Etats-Unis, 78035 Versailles,
France
e-mail: paolo.vannucci@uvsq.fr

is presented here, as well as the numerical finite element model used to simulate the structure, object of the examples considered in this study; finally, a large part of this paper is occupied by these numerical examples.

As already said in Part 1, we have used a special genetic algorithm as numerical tool for the solution of the two-level problems mentioned above. Genetic algorithms (GAs) have some peculiarities that render them very adapted to the solution of problems like those considered here. Namely, GAs are zero-order algorithms, so they do not need the computation of derivatives. This is very important when some of the design variables are discrete valued, as in our case. GAs can handle with the same easiness variables of all the types and most important, they are very indicated for highly non-convex optimum problems, which is our case. There is a huge literature on GAs; we cite only the fundamental texts of Goldberg [1], and of Michalewicz, [2]. They are today the most used tool for the numerical solution of laminates optimum design problems; to this purpose, we recall only the already cited works by Nagendra et al. [3], Kaletta and Wolf [4], or the papers from Muc and Gurba [5], and from Tabakov [6], just as some examples among many others.

However, the use of GAs for the solution of laminate optimum design problems has been, by several authors, too much simplified and reduced, in some cases, more to a guided search on a discrete set of integer values than to a true GA. Such transformations of GAs have been in several cases the consequence of using drastic simplifying assumptions in the formulation of the laminate optimum design problem.

We could not adopt GAs as normally done for the solution of laminates design problems for two reasons: because our point of view is to avoid all simplifying assumptions, *conditio sine qua non* for obtaining a true global optimum, and because we face in our strategy of optimization two major difficulties: the first one is the need for optimizing also the number of modules, i.e., of stiffeners and of layers, and this implies, as already said in the first part and exposed in detail in the following section that the algorithm must be able to find by itself the optimal number of design variables. The second difficulty is the fact that we need to handle constraints. GAs, like many other metaheuristics, are numerical procedures born to handle unconstrained optimum problems. They can be adapted to solve also problems with constraints, by using some classical procedures, like penalization or other approaches; see, for instance, [7]. For all these reasons, we needed to use a GA with a special structure and a good strategy for handling constraints. The numerical tool that we have used for the solution of the two optimum problems is the code BIANCA (Biologically Inspired ANalysis of Composite Assemblages), [8, 9] that has been coupled to a finite element code for the structural simulations concerning the first problem. The details of these two aspects are given below.

2 The Genetic Algorithm BIANCA

Genetic algorithms (GAs) have gained increasing popularity in design optimization, operation research and for general combinatorial search problems [1, 2]. The point that essentially distinguish GAs from other evolutionary metaheuristics is the *coding phase*: It translates the design variables from the *phenotypic* space to the *genotypic* one resulting, for example, in a binary *chromosome*, composed by *genes* that code the value of the design variables. The length of the chromosome, i.e., the number of genes, represents in the genotype space the amount of the information that is restrained in the individual, the phenotype. The chromosome's length L can be expressed as

$$L = \sum_{i=1}^{n_{\mathrm{var}}} \left\lceil \ln\left(\frac{x_{i\,\mathrm{UB}} - x_{i\,\mathrm{LB}}}{\Delta x_i} \right) / \ln d \right\rceil \tag{1}$$

where n_{var} is the number of decision variables, x_i and Δx_i are the ith design variable, and its resolution level, respectively, while d is the dimensionality of the encoding base. $x_{i\,\mathrm{LB}}$ and $x_{i\,\mathrm{UB}}$ are the lower and upper bounds for the ith design variable, while the $\lceil\ \rceil$ operator is the ceiling function (rounding to the next integer).

In traditional GAs, the chromosome length is fixed a priori by the total number of variables along with their resolutions and cannot change during the whole genetic process. As described in other existing works on this subject (see, for example, [10] and [11]) the traditional approach has substantially two drawbacks:

- the best achievable fitness is inherently limited by the chromosome length, and hence by the total number of variables. Therefore, the genetic asymptote, which is typical of the genetic process, is a direct consequence of the constraints of the problem as well as the number of design variables and their level of resolution;
- we do not a priori know how many design variables are required, and consequently how long the chromosome should be, for a given problem, in order to obtain a real global optimum.

In these last years, different research studies have been developed in the field of improved GAs with variable chromosome length. Among them, Kim and Weck [10] developed a GA that can change the chromosome length using a "progressive refinement" technique. They assumed that significant fitness improvements can be obtained by gradually increasing the chromosome length. They achieved the increase in the chromosome length mainly in two ways: either by increasing the resolution level of the existing design variables or by adding new design variables during the process. In every case, these operations were realized by a particular mutation operator acting on the chromosome structure: the concept consists in seeding the design space of finer resolution with mutated best designs from the domain of coarser resolution. They applied their strategy to two structural topology optimization problems: the design of a short cantilever and that of a bridge. Despite Kim and Weck developed an effective GA technique in their context, they did not develop genetic operators for crossover and mutation on individuals with different chromosome lengths. The variation of the number of variables is obtained through a mutation process which is linked to the concept "from coarse to fine" that is doubtless very effective when dealing with topology optimization problems, but could appear not as effective for other types of problems.

Ryoo and Hajela [11] developed a GA for topology optimization that also handles variable chromosome lengths. This GA allows only crossover between chromosomes of different lengths. Even though they implemented an interspecies crossover operator, they did not obtain an effective evolution of the species along the generations. In other words, the number of chromosomes with different length remains the same from the beginning until the end of the process, i.e., from the initial population until the final one.

Park et al. [12] developed an improved GA able to cross chromosomes with different lengths. Within their GA, the individual is characterized by a single chromosome: At each generation, the change in the length of the chromosome was realized by means of a special mutation operator acting directly on the number of genes composing the chromosome. Their strategy was applied to the weight minimization of laminated plates manufactured by the Resin Transfer Moulding (RTM) process, considering the technological requirements as constraints of the optimization problem.

In the following subsections, we show the structure of the individuals in BIANCA, which can also take into account for individuals belonging to different *species*. We present also the special genetic operators in BIANCA, for crossover and mutation on species. Some of these operators are inspired by the work of Park et al. and could be considered as a generalization of those presented in [12]. These operators show, however, some original features because we intended to preserve a deep genetic recombination strategy, which proved to be effective in other problems previously solved with our code.

2.1 General Features of BIANCA

According to the metaphor used by GAs, each point in the design space corresponds to an *individual*, algebraically represented by its *genotype*. The convergence toward the best individual is based upon a scheme of Darwinian selection, carried on by some *genetic operators*. Though the general structure of the code BIANCA is rather classical (see Fig. 1), particular attention has been devoted to the organization of the information, which is particularly rich and detailed, though non-redundant.

The GA BIANCA shows the following original features; see [8] for more details:

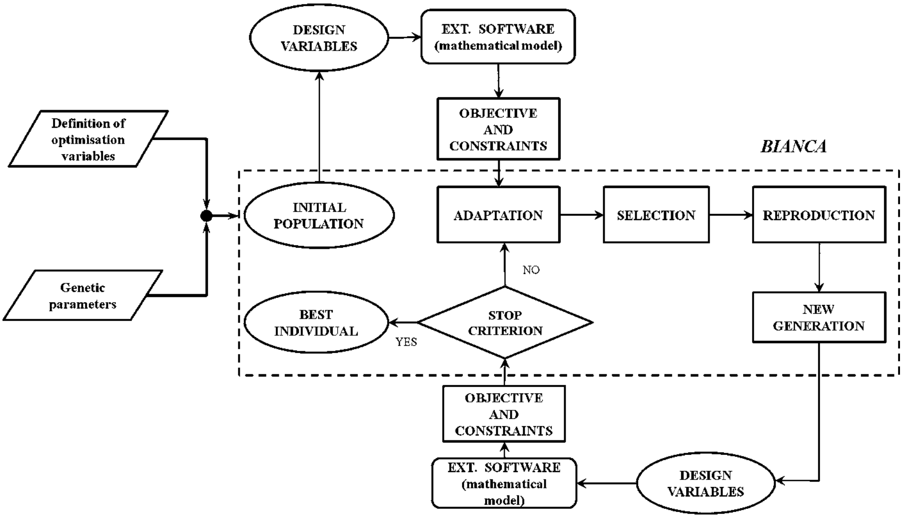


Fig. 1 Structure of the GA BIANCA and of the interface with external software

- *fitness evaluation*: the choice of the fitness determines the kind of problem, i.e., if it is a minimization or maximization one, and the selection pressure that the user decides to introduce. The fitness is evaluated in such a way that it can assume all the possible values in the range [0, 1], with 0 corresponding the least fitted individual and 1 to the best fitted one;
- *selection*: two known techniques of selection are included, i.e., roulette wheel and tournament;
- *standard genetic operators*: the main genetic operators are crossover and mutation, used with a certain probability on each gene of the individual's genotype, i.e., they act independently on each design variable;
- *additional genetic operators*: the *elitism* operator can be used to preserve the best individual of each generation;
- *handling constraints*: the *Automatic Dynamic Penalization* (ADP) method is implemented;
- *stop criterion*: maximum number of generations reached or test of convergence, i.e., no improvements of the mean fitness of the population after a given number of cycles.

The main feature of BIANCA is, however, its way of representing the information; actually, it is thanks to the special representation of the information implemented in BIANCA that we could effectively handle and solve some optimization problems, like the one considered in this research, where a sort of hierarchical structure of the parameters exist. Some of the main features of BIANCA are briefly described below.

2.2 Representation of Individuals and Species in BIANCA

Unlike what is done in the most part of GAs used for laminates design, that have a mono-chromosome algebraic structure, in BIANCA the information is organized in a *genome* composed of *chromosomes* which in turn are composed by *genes*; finally, each gene is a binary representation of a design variable. When the object of the optimization problem is a modular system, each constitutive module is represented by a chromosome, while each gene composing a chromosome codes a design variable related to the module.

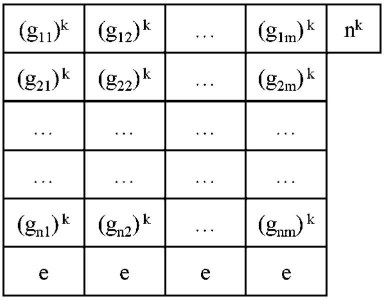
In agreement with the paradigms of natural sciences, individuals characterized by different number of chromosomes, i.e., modular structures composed of a different number of modules, belong to *different species*. BIANCA has been conceived for crossing also different species, and is able to make in parallel and without distinction the optimization of the species and of the individuals.

Moreover, the information restrained in the population is treated in BIANCA in such a way to allow for a deep mixing of the individual genotype. In fact, the reproduction operators, i.e., crossover and mutation, act on every single gene of the individuals, so allowing for a true independent evolution of each design variable. In particular, the typical reproduction operators of *crossover* and *mutation* have been specially conceived for crossing species and individuals during the same iterations.

Considering what said above, under the practical point of view in BIANCA an individual is represented by an array of dimensions $n_{\text{chrom}} \times n_{\text{gene}}$. The number of rows, n_{chrom}, is the number of chromosomes, while the number of columns, n_{gene}, is the number of genes. Basically, each design variable is coded in the form of a gene, and its meaning is linked both to the position and to the value of the gene within the chromosome. In principle, there are no limits on the number of genes and chromosomes for an individual. A number n_{ind} of individuals compose a population, and in BIANCA it is possible to work, at the same time, with several populations whose number, n_{pop}, can be defined by the user.

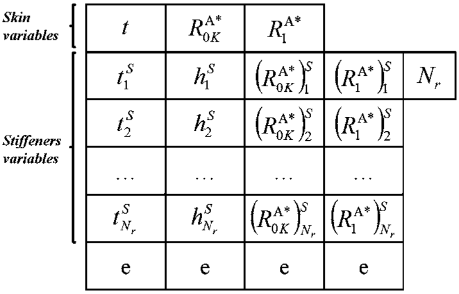
In order to include the number of chromosomes (i.e., of modules, and hence of design variables) among the design variables, and then to allow the reproduction among individuals belonging to different species, some modifications of the individual genotype were necessary. The genotype of each individual in BIANCA is represented by a binary array, shown in Fig. 2. In this picture, the quantity $(g_{ij})^k$ represents the jth gene of the ith chromosome of the kth individual. Letter e stands for empty location, i.e., there is no gene in this location while n^k is the kth individual chromosomes number. It appears clearly that every individual can have a different number of chromosomes, i.e., each individual can belong to a different species.

Figure 3 shows the genotype of the generic rth individual for the optimization problem of the wing-box structure. This individual has $N_r + 1$ chromosomes. The first chromosome is composed by 3 genes representing the design variables for the skin, i.e., thickness and polar parameters. Chromosomes from 2 to $N_r + 1$ contain 4 genes which are the design variables for each stiffener: thickness, height and the two anisotropy polar parameters. An exception is chromosome 2 that has 5 genes: the fifth additional gene codes the number of modules, i.e., for our problem the number of stiffeners (we remind the use of the superscript S for all quantities related to the stiffeners).



$(g_{11})^k$	$(g_{12})^k$	…	$(g_{1m})^k$	n^k
$(g_{21})^k$	$(g_{22})^k$	…	$(g_{2m})^k$	
…	…	…	…	
…	…	…	…	
$(g_{n1})^k$	$(g_{n2})^k$	…	$(g_{nm})^k$	
e	e	e	e	

Fig. 2 Structure of the individual's genotype with variable number of chromosome in BIANCA



t	R_{0K}^{A*}	R_1^{A*}		
t_1^S	h_1^S	$\left(R_{0K}^{A*}\right)_1^S$	$\left(R_1^{A*}\right)_1^S$	N_r
t_2^S	h_2^S	$\left(R_{0K}^{A*}\right)_2^S$	$\left(R_1^{A*}\right)_2^S$	
…	…	…	…	
$t_{N_r}^S$	$h_{N_r}^S$	$\left(R_{0K}^{A*}\right)_{N_r}^S$	$\left(R_1^{A*}\right)_{N_r}^S$	
e	e	e	e	

Fig. 3 Structure of the individual genotype for the first optimization problem

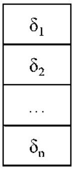


Fig. 4 Structure of the individual genotype for the second optimization problem

The code BIANCA has also been used in the search for the laminates, that is made in the second step of the procedure. The structure of the genotype of the individual-laminate is shown in Fig. 4. The genotype is made of n chromosomes which correspond to the number of layers, determined in the first step, and each chromosome is composed, on its turn, by a single gene which represents the ply orientation.

2.3 Evolution of Species and Individuals in BIANCA: New Operators of Crossover and Mutation

In a standard GA, the classical reproduction phase takes place by means of crossover and mutation operators, which act on the genotype of individuals.

In particular, in BIANCA crossover and mutation operate on a pair of homologous genes, with a given probability, p_{cross} and p_{mut} respectively, whose values are fixed by the user. Crossover and mutation are performed by means of Boolean operators, based on the computer-embedded binary representation of numbers. In this way, any coding-decoding process, from binary to integer representation and vice-versa, is no longer needed and the genetic operations of crossover and mutation are much faster. The choice of operating crossover and mutation separately on each gene, ensures a deep mixing of the genotype of the whole population at each generation, i.e., an extensively exploration and exploitation of the information on the whole design space. The reader is addressed to [8] for more details on these aspects.

With the introduction of the concept of reproduction between different species, new genetic operators are required in executing the reproduction phase.

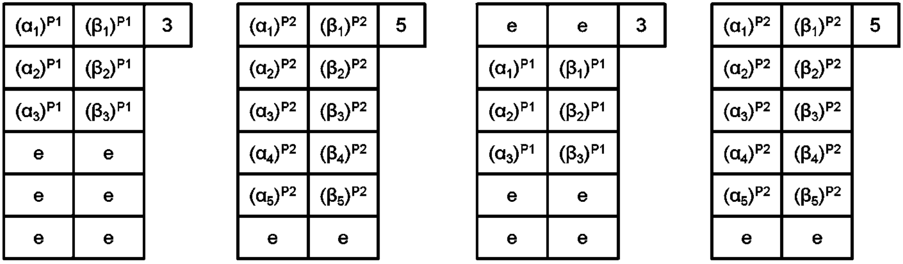
As mentioned above, the number of design variables determines the number of chromosomes, i.e., the *biological species*. So, if the number of variables has to be included among the optimization variables, and has to evolve during generations, a reproduction among the species has to be performed. In particular, the classical reproduction phase has been changed introducing new genetic operators called *chromosome shift operator*, *chromosome reorder*, *chromosome number mutation*, and *chromosome addition-deletion*. A brief description of these new operators and their use in the reproduction phase is given below.

2.3.1 The New Crossover Phase and the Role of Chromosomes Shift and Reorder Operators

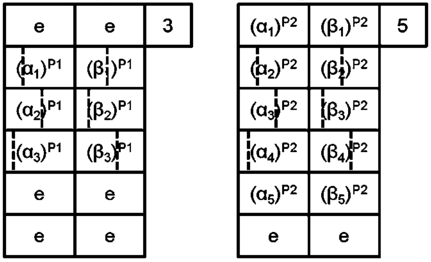
To explain the way whereby the reproduction phase takes place, one can consider the following case. There are two *parents*, $P1$ and $P2$, with 3 and 5 chromosomes respectively; see Fig. 5(a). In this example, the maximum number of chromosomes is assumed equal to 6, while the minimum number can be chosen arbitrarily between 1 and 6: Therefore, in Fig. 5(a), parent $P1$ shows 3 empty locations, while parent $P2$

(a) Parents couple

(b) Shift operator



(c) Crossover on each gene



(d) Children couple

(e) Reorder operator

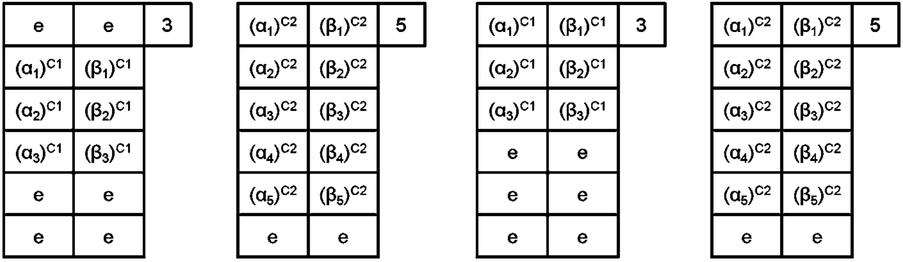


Fig. 5 Crossover among species: (**a**) parents couple, (**b**) effect of the *shift* operator, (**c**) crossover on homologous genes, (**d**) children couple and (**e**) effect of the *chromosome reorder* operator

only one. Moreover, there are two different variables for each individual, i.e., each chromosome has two different genes α and β. Before realizing the crossover on these two individuals, it can be noticed that there are different ways to pass the information restrained in the parents' genotype to the next generation, i.e., to their *children*. Here, at the next generation, two new individuals will be produced from this couple, one with 3 chromosomes and another one with 5 chromosomes. To improve the efficiency of the GA in terms of exploration and exploitation of the information on the design space, the concept of *shift factor* is introduced. The shift factor is randomly sorted, with a given probability p_{shift}, in the range $[0, |n^{P1} - n^{P2}|]$, where $|n^{P1} - n^{P2}|$ is the absolute value of the difference of the parents' chromosomes number. Using the shift factor, various combinations of crossover are possible and the shift operator acts on the individual with the smaller number of chromosomes. In the example mentioned before, the minimum shift factor is 0 and the maximum is 2. For example, if the sorted value of the shift factor is 1, all the genes of $P1$, which has the smaller number of chromosomes, are shifted by a quantity equal to 1 up-to-down as shown in Fig. 5(b).

After the shift operation, the crossover phase takes place. The crossover operator acts separately and independently on every single gene. The position of crossover is randomly chosen for each gene of both individuals. Naturally, this operator involves all the chromosomes of the parent with the smaller number of them, i.e., in the case shown in Fig. 5(c) all the genes of $P1$, while only the homologous genes of $P2$ undergo the action of crossover operator. At this point, two new individuals are created, $C1$ and $C2$ that have 3 and 5 chromosomes, respectively; see Fig. 5(d). It can be noticed that the 1st and 5th chromosome of $P2$ have not undergone the crossover phase, so according to the notation of Fig. 5(c) and (d) it is possible to write the following equalities, $(\alpha_1)^{P2} = (\alpha_1)^{C2}$, $(\alpha_5)^{P2} = (\alpha_5)^{C2}$ and $(\beta_1)^{P2} = (\beta_1)^{C2}$, $(\beta_5)^{P2} = (\beta_5)^{C2}$.

Before the mutation phase, a readjustment of the chromosomes position is required. The *chromosome reorder* operator achieves this phase by a translation of all chromosomes down-to-up in the structure of the individual with the smaller number of them; see Fig. 5(e).

2.3.2 The New Mutation Phase and the Role of Chromosomes Number Mutation and Addition-Deletion Operators

Mutation is a random process applied to the genotype to better explore the feasibility domain. Mutation is articulated in two phases: in a first stage, it acts on the number of chromosomes and then on the genes values.

In the first phase, the chromosomes number is arbitrarily changed by one at time for each individual, with a given probability $(p_{\mathrm{mut}})_{\mathrm{chrom}}$, then the *chromosome addition-deletion* operator acts on the genotype of both individuals, by adding or deleting a chromosome. The location of chromosome addition-deletion is also randomly selected. Naturally, if the chromosomes' number is equal to the maximum one, only deletion can occur. Similarly, if the chromosomes' number is equal to the minimum one, only addition can be applied. In the case shown in Fig. 6(a), the number of chromosomes of $C1$ is increased by one and a new one, $\{(\alpha_a)^{C1}, (\beta_a)^{C1}\}$, is randomly sorted and randomly added; in this example in correspondence of position 3, while the number of chromosomes of $C2$ is decreased by one and the chromosome deletion is randomly done at position 2.

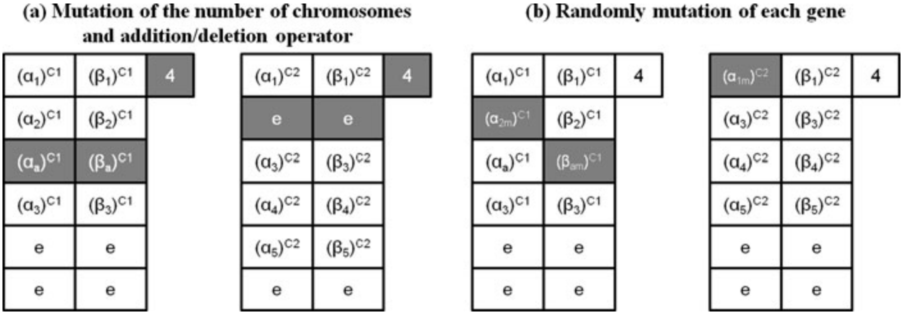


Fig. 6 Mutation of species: (**a**) mutation of the number of chromosomes and effect of the *chromosome addition-deletion* operator, (**b**) effect of the mutation operator on every gene

In the second phase, the mutation of the genes value takes place, for instance, one-bit change, with a probability p_{mut}, after a rearrangement of chromosomes position. In the example of Fig. 6(b), the mutation is undergone by the genes $(\alpha_2)^{C1}$ and $(\beta_a)^{C1}$ of the individual $C1$ and by the gene $(\alpha_1)^{C2}$ of the individual $C2$.

2.4 Handling Constraints in BIANCA

In BIANCA, we adopt a penalization strategy for handling the constraints. The constrained minimization problem, formulated in Part 1 by (24), is transformed into an unconstrained one defining the penalized objective function as

$$W_p(\mathbf{x}) = W(\mathbf{x}) + \sum_{k=1}^{q} c_k G_k(\mathbf{x}) \ , \tag{2}$$

where W_p is the penalized value of the objective function while W is the unpenalized one and $\mathbf{x}$ is the vector of design variables. G_k is the kth violated constraint, q is the total number of violated constraints, while c_k is the kth penalization coefficient.

The ADP method is an original method that we have developed to automatically choosing and updating the penalization coefficients. The basic idea is that some unfeasible individuals can be anyway important to drive exploration toward interesting zones of the feasible domain, namely when the optimum point lies on its boundary, i.e., on an active constraint. For this reason, in the ADP some unfeasible points are not automatically excluded from the population, especially at the early stages of the search, and are used to dynamically update the penalization coefficients in an automatic way. For the sake of shortness, this is not the place to enter in the details of the ADP, the reader is addressed to [8] or [9] for a deeper insight in the matter. We specify, however, that box constraints, like those specified in Part 1 by the first and second of (9) are quite natural to be handled in a GA, simply fixing the bounds of their domain, and do not need a special treatment by a penalization method or by another approach.

2.5 The Interface of BIANCA with External Software

In several problems, the value of the objective function and/or constraints, cannot be computed analytically, but it has to be evaluated using special numerical codes. Typically, this is the case of structural optimization, where the most part of times the structural response is numerically assessed using finite-element (FE) codes. For these cases, a very general interface has been developed, which renders BIANCA able to exchange input/output information with mathematical models supported by an external software.

Figure 1 shows the structure of the data-exchange between BIANCA and a generic external software. For each individual, BIANCA performs the genetic operations, such as selection, crossover, mutation, and so on, and then passes the design variables to the mathematical model written in a different environment. At this point, the external software evaluates the objective and the existing constraint functions values, and then passes them back to BIANCA. The data-exchange between BIANCA and the external software is simply done by means of two text files.

The first one is the text file written from BIANCA and passed to the external software, i.e., the *input file*, which contains the information related to the current individual at the current generation, i.e., the number and the values of the design variables restrained in that individual's genotype. This *input file* contains also additional information such as the number of objective functions, inequality constraints, and equality constraints.

The second one is the text file written from the external software and passed to BIANCA, i.e., the *output file*, wherein are written the values of the objective functions and of the equality and inequality constraints.

The writing operations of these files are made for each individual in the current generation, so the external software, during the whole optimization process, is called from BIANCA $N_{\text{ind}} \times N_{\text{gen}}$ times, where N_{ind} is the number of individuals while N_{gen} is the number of generations. Up to now, some current and well-known software packages have been interfaced with BIANCA in this way; like, for instance, MATLAB, ABAQUS and ANSYS, which is the FE code used for this research.

3 The Finite Element Model of the Structure

The finite-element analysis is used for evaluating the objective and constraint functions for each individual at each iteration of the first problem. The need to analyze, within the same generation, different geometrical configurations, each one corresponding to an individual, requires the creation of an ad-hoc input file for the FE code that has to be interfaced with BIANCA, according to the scheme shown in Fig. 1. Since the number of modules is included among the decision variables, the FE model must be conceived in order to take into account for variable geometry and mesh. Indeed, for each individual at the current generation, depending on the number of chromosomes, and hence on the number of stiffeners the FE code has to be able to vary in a correct way the number of elements wherein the structure is discretized, thus a correct parameterization of the model has to be done.

The geometry, mesh, loads, and boundary conditions of the wing-box FE model are shown in Fig. 7. The structure is modeled using shell elements with 8 nodes having 6 degrees of freedom per node. These shell elements have 3 integration points along the thickness. The mechanical properties of the material are defined specifying the Cartesian components of tensor $\mathbf{A}^*$, that are functions of the mechanical unknowns, the polar parameters; see Part 1.

The wing-box is considered simply supported at its terminal sections on two-wing ribs. The upper panel is subjected to a uniform compression unit force per unit length and the lower one to a uniform tensile unit force per unit length. Under such kind of loads, which are representative of the loads that the structural elements of the wing-box undergo in normal flight conditions, only the upper panel and the corresponding stiffeners can undergo compression instability phenomena. Both upper and lower panels have N Z-shaped stiffeners.

After a preliminary mesh sensitivity study, the average dimensions of the shell elements have been chosen equal to 14×14 mm^2. The number of shell elements in the whole wing-box structure can vary from 14080 to 17680, depending on the

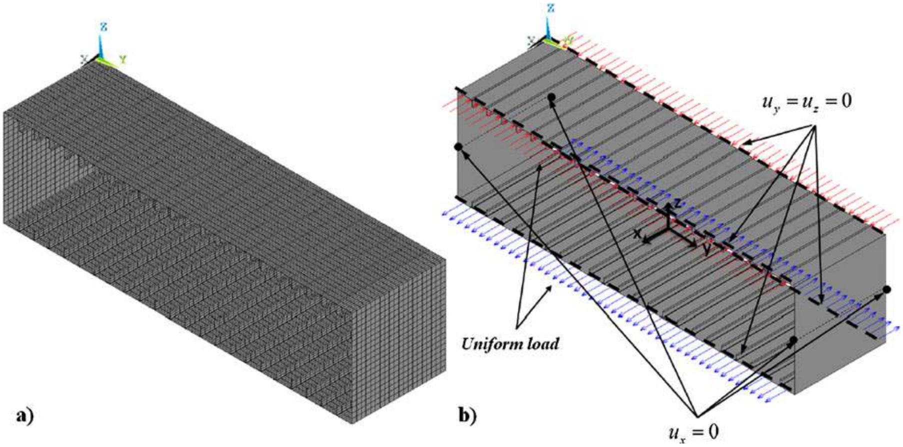


Fig. 7 (**a**) Mesh and (**b**) loads and boundary conditions for the wing-box FE model

number of stiffeners N, while the number of degrees of freedom of the whole model can vary from 270744 to 340164. For each individual, a linear buckling analysis is performed for assessing the first buckling load of the structure.

In order to fix a correct reference value of the first buckling load, i.e., of the quantity p_{ref} appearing in the formulation of the first level problem (see Part 1 of this work), a preliminary buckling analysis on a reference wing-box section model has been performed for the type of wing-box section considered here and represented in Fig. 7. The reference wing-box, which represents a standard structure, is made by Al-7075-T6 alloy with Young's modulus $E = 72395$ MPa, Poisson's ratio $\nu = 0.33$, yield stress $\sigma_y = 475$ MPa and density $\rho = 2760$ kg/m^3. Concerning the geometrical properties, the whole wing-box has the global dimensions shown in Fig. 1 of Part 1, while the upper and lower panels are made by 20 identical stiffeners having the following dimensions: $t^S = 2.96$ mm and $h^S = 62.33$ mm for the stiffeners thickness and height respectively, and $t = 4.93$ mm for the skin thickness. The reference values p_{ref} of the first buckling load and W_{ref} of the wing-box weight are the outcomes of this preliminary simulation: $p_{\text{ref}} = 1928$ N/mm and $W_{\text{ref}} = 1222.62$ N.

This structure is also compared in the following section with the solutions found for the considered examples. Actually, our goal is to compare the optimal solutions that we find with a case that carries the same buckling load and that can be considered as a usual, typical situation, briefly a standard wing-box girder for a standard aircraft structure. Of course, other comparisons could be done, nevertheless it is really significant to compare what can be done as an optimum with what is usually done as a standard: That was our goal.

4 Examples

For our optimization problem, we have considered three different examples. The design variables, their nature, and bounds for the optimization problem at hand are

Table 1 Design variables for the first optimization problem

Design variable	Type	Lower bound	Upper bound	Step
N	discrete	18	23	1
t_i^S [mm]	discrete	2.0	5.0	0.125
h_i^S [mm]	discrete	40.0	90.0	0.5
$(R_{0K}^{A^*})_i^S$ [MPa]	continuous	-19710	19710	–
$(R_1^{A^*})_i^S$ [MPa]	continuous	0.0	21433	–
t [mm]	discrete	2.0	5.0	0.125
$(R_{0K}^{A^*})$ [MPa]	continuous	-19710	19710	–
$(R_1^{A^*})$ [MPa]	continuous	0.0	21433	–

detailed in Table 1, already presented in Part 1 (we have preferred to show all the tables in an addendum at the end of the paper because some of them are very long).

- **Example 1**: The stiffeners are identical, i.e., they have the same value of thickness t^S, height h^S and polar parameters, i.e., $(R_{0K}^{A^*})^S$ and $(R_1^{A^*})^S$. Therefore, in this case, we have 8 design variables: the number of stiffeners N, the geometrical and polar parameters for the stiffeners, i.e., t^S, h^S, $(R_{0K}^{A^*})^S$, $(R_1^{A^*})^S$, and the geometrical and polar parameters for the skin, i.e., t, $R_{0K}^{A^*}$, $R_1^{A^*}$. In addition, for this first case, the total number of constraints is 11: 1 constraint on the buckling load, 5 geometric constraints for the skin and 5 for the stiffener polar parameters (see Part 1, Sect. 4.3). However, 8 constraints on the whole of 11 are box-constraints, so they are not treated by the ADP method, but simply used to specify the variation range of the anisotropy design variables $R_{0K}^{A^*}$ and $R_1^{A^*}$.
- **Example 2**: The problem of the minimum weight for the wing-box stiffened panel is now formulated in the most general case, i.e., with non-identical stiffeners. The total number of design variables depends on the number of stiffeners N, as explained in part 1, Sect. 4, and can vary between 76 and 96. Moreover, also the number of constraint equations is variable with the number of stiffeners: The minimum number of constraints is 96 and the maximum 121, respectively, 19 and 24 without the box-constraints.
- **Example 3**: We still consider the problem of the minimum weight for the wing-box stiffened panel with non-identical stiffeners. Nevertheless, in this last case, for obvious mechanical reasons, we assume that the whole wing-box section has a symmetric distribution of the geometrical and polar parameters for the stiffeners with respect to the x–z plane of the global reference system; see Fig. 7. With this assumption, the total number of design variables and constraints is considerably reduced and can vary between 40 and 52 for the design variables, and between 51 and 66 for the constraints, respectively, 10 and 13 without the box-constraints.

The three different cases are detailed hereafter separately, both for the first and second step. We precise that in all the three cases, the design is guided by the buckling constraint, regardless if it appears as a global or local phenomenon, which means that the best solution should belong to the boundary between the feasible and unfeasible domain. This is a precise choice for the mechanical design of the structure, though

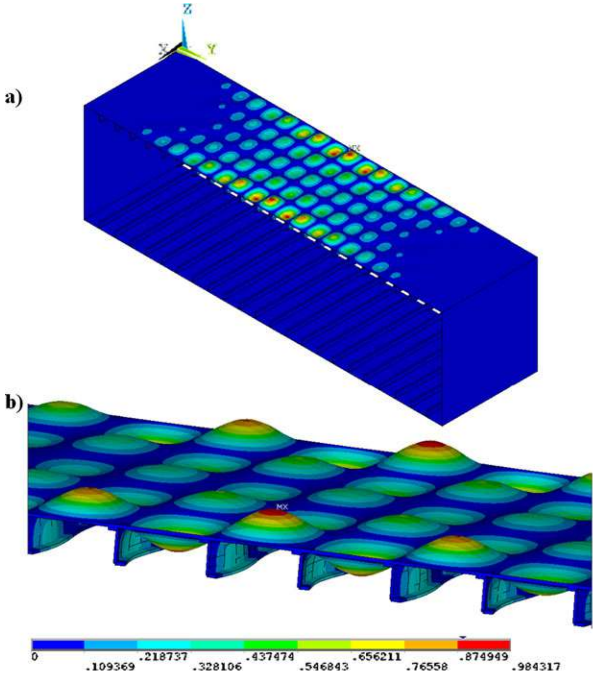


Fig. 8 Example 1: deformed shape of the (**a**) whole wing-box section and (**b**) upper-panel stiffeners

other situations can be considered, for instance concerning the possibility of a post-buckling design.

For what concerns the second problem, in all the three examples (see (29) of Part 1), the design variables are the layers orientations, which can vary between $-90°$ and $90°$ with a step of $1°$. In all the cases, the population size has been set to $N_{\text{ind}} = 500$ and the maximum number of generations to $N_{\text{gen}} = 500$. The crossover and mutation probability are $p_{\text{cross}} = 0.85$ and $p_{\text{mut}} = 1/N_{\text{ind}}$, respectively. Selection is performed by the roulette-wheel operator and the elitism is active.

As in each numerical technique, the quality of solutions found by BIANCA can be estimated on the basis of a numerical tolerance, that is the residual. For a discussion on the importance of the numerical residual in problems of this type, see [13] or [14]. It is worth noting that, being $F(\boldsymbol{\delta})$ a non-dimensional function; the residual of the solution is a non-dimensional quantity, also. It is worth to recall that the second problem must be solved for the skin and for each stiffener separately, when they are not identical.

4.1 Example 1: Wing-Box Panel with Identical Stiffeners

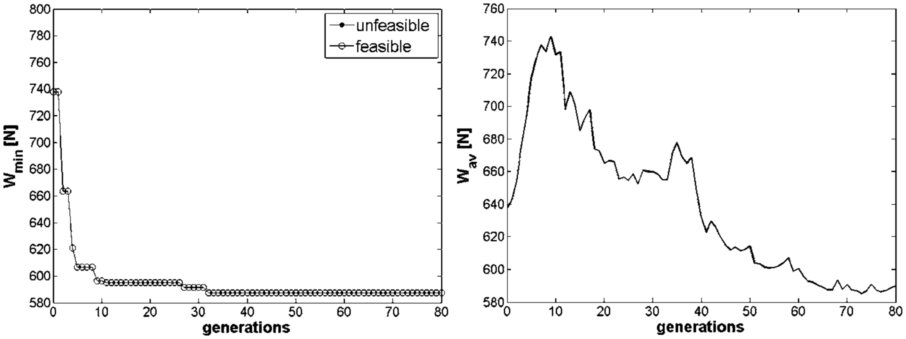
In this first case, since the stiffeners are identical, the genetic operators that perform the crossover between different species are no longer required: the genotype of each individual is composed by only one chromosome with 8 genes: The first gene represents the number of stiffeners N, the genes from 2 to 4 represent the skin design variables, i.e., t, $R_{0K}^{A^*}$ and $R_1^{A^*}$, while the last 4 genes represent the stiffeners design variables, i.e., t^S, h^S, $(R_{0K}^{A^*})^S$ and $(R_1^{A^*})^S$.

Concerning the genetic parameters that regulate the iterations of the GA BIANCA, the population size is set to $N_{\mathrm{ind}} = 50$ and the maximum number of generations is assumed equal to $N_{\mathrm{gen}} = 80$. The crossover and mutation probability are $p_{\mathrm{cross}} = 0.85$ and $p_{\mathrm{mut}} = 1/N_{\mathrm{ind}}$, respectively. Selection is performed by roulette-wheel operator, the elitism is active and the ADP method is used for handling constraints.

The best solution found by BIANCA is shown in Table 2. The optimal number of the stiffeners for the weight minimization is 22. The buckling load and the wing-box weight are $p_{\mathrm{cr}} = 1943$ N/mm and $W = 587.28$ N, respectively. Figure 8 shows the deformed shape of the structure when the applied load is equal to p_{cr}: we can see that, during the buckling phenomenon, the wing-box section is characterized by a local skin buckling around the stiffeners.

Considering that the value of the ply's thickness is 0.125 mm, we can notice that the laminate of each stiffener is made of 29 plies and has the orthotropy with $\widehat{K}^{A^*} = 1$, because the value of the polar quantity $(R_{0K}^{A^*})^S$ is negative. Instead, the skin laminate is made of 32 layers and has $\widehat{K}^{A^*} = 0$, because $R_{0K}^{A^*}$ is positive.

The global constrained minimum has been found after 32 generations; see Fig. 9(a). This solution gives a reduction of the weight of the whole structure of about 52 %, when compared to the reference solution, and the solution found is practically on the boundary of the feasible domain, $p_{\mathrm{cr}} \simeq p_{\mathrm{ref}}$. From Fig. 9(b), it can also be no-



a) b)

Fig. 9 Example 1: (**a**) best and (**b**) average values of the objective function along generations

Table 2 Example 1: best values of the design variables

Design variable	Value
N	22
t^S [mm]	3.625
h^S [mm]	40.0
$(R_{0K}^{A^*})^S$ [MPa]	−984.36
$(R_1^{A^*})^S$ [MPa]	6425.22
t [mm]	4.0
$R_{0K}^{A^*}$ [MPa]	16399.8
$R_1^{A^*}$ [MPa]	1293.26

ticed that the variation of the average value of the weight over the whole population along generations firstly increases (for about 9 iterations) and then decreases. This trend is due to the presence of a large amount of unfeasible individuals within the population in the initial generations: Unfeasible points represent solutions which are lighter than the feasible ones but that violate the constraint on the buckling load, thus these individuals are penalized by the ADP strategy and this results in an increase of the average of the objective function. However, though these individuals belong to the unfeasible region, they are preserved within the population because they can be useful for driving the search for the optimum point towards regions close to the boundary of the feasible domain; see Sect. 2.4.

Table 3 shows the best stacking sequences found using BIANCA for the second level problem, cfr. (29) of Part 1. The residual in the last column is the value of the global objective function $F(\boldsymbol{\delta})$ for the solution indicated aside; we remind that exact solutions correspond to the zeroes of the objective function. Figure 10 shows the first component of the homogenized stiffness tensors of the laminate, i.e., $\mathbf{A}^*$, $\mathbf{B}^*$ and $\mathbf{D}^*$, for stiffeners and skin: the solid line refers to the extension tensor, the dashed one to the bending tensor, while the dash-dotted one is linked to the coupling stiffness

Table 3 Example 1: best stacking sequences for the optimal solution

N of plies	Stacking sequence (°)	Residual
STIFFENERS		
29	[−8/28/26/ − 45/ − 58/ − 3/55/ − 31/76/34/ − 39/ − 87/ − 7/6/30/ − 12/ − 21/ − 51/18/ − 55/49/ − 8/18/12/57/44/ − 27/ − 79/ − 18]	2.9×10^{-4}
SKIN		
32	[−81/7/ − 3/ − 12/82/86/ − 87/20/ − 6/76/ − 7/85/ − 6/90/ − 7/87/10/ − 82/ − 4/ − 7/ − 82/18/ − 11/ − 84/ − 83/7/70/85/1/ − 12/1/89]	8.4×10^{-5}

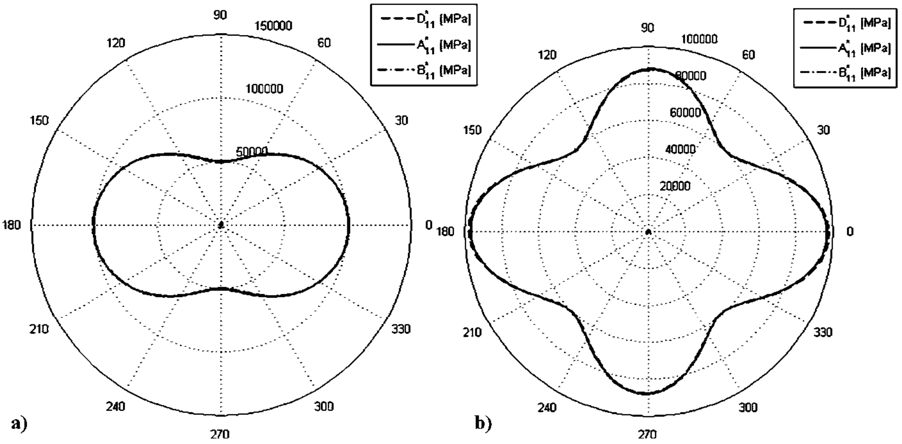


Fig. 10 Example 1: first component of the homogenized stiffness tensors of the laminate for (**a**) stiffeners and (**b**) skin

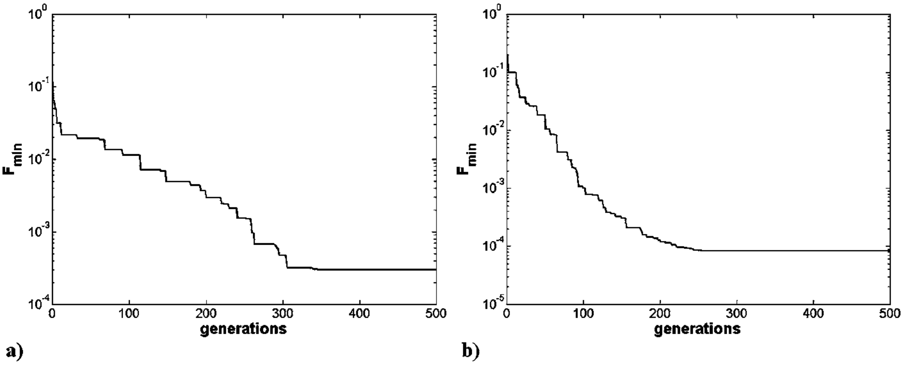


Fig. 11 Example 1: best values of the objective function during iterations for (**a**) stiffeners and (**b**) skin laminates

tensor. We can see that both laminates are uncoupled (the dash-dotted curve is almost invisible, reduced to a small black point in the center of the plot, because B^*_{11} is practically null), homogeneous (the solid and dashed curves are practically coincident) and orthotropic (they show two axes of symmetry in the plane). Moreover, the main orthotropy axis is aligned with the x axis of the structure, in fact, it is oriented at 0°. Similar considerations can be done for the other components of these tensors, not shown in Fig. 10 for the sake of brevity.

Figure 11 shows the variation of the best solution during iterations, for stiffener and skin laminates, respectively. The best solution is found after 340 generations for the stiffener laminate, while for the laminate of the skin it is found after 250 generations.

4.2 Example 2: Wing-Box Panel with Non-identical Stiffeners

Since the number of stiffeners is variable and they are not identical, a crossover between species is required and the optimal value of N is an outcome of the optimal search: the most adapted species automatically issues as a natural result of the Darwinian selection. The genotype of an individual has been detailed in Sect. 2.2 and shown in Fig. 3.

Concerning the genetic parameters, the population size is $N_{\text{ind}} = 70$ and the maximum number of generations is $N_{\text{gen}} = 80$. The crossover and mutation probability are still $p_{\text{cross}} = 0.85$ and $p_{\text{mut}} = 1/N_{\text{ind}}$, while the shift operator and chromosomes number mutation probability are $p_{\text{shift}} = 0.5$ and $(p_{\text{mut}})_{\text{chrom}} = (n_{\text{chrom}\max} - n_{\text{chrom}\min})/N_{\text{ind}}$, see [9] for details about these parameters. Once again, selection is performed by roulette-wheel operator, the elitism is active and the ADP method has been used for handling constraints.

The best solution found by BIANCA is shown in Table 4. The optimal number of stiffeners for the weight minimization is 23. The buckling load and the wing-box weight are $p_{\text{cr}} = 1931$ N/mm and $W = 620.19$ N, respectively. Figure 12 shows the deformed shape of the structure when the applied load is equal to p_{cr}. From Table 4, it can be noticed that the orthotropy type can be different for the stiffener laminates:

Table 4 Example 2: best values of the design variables

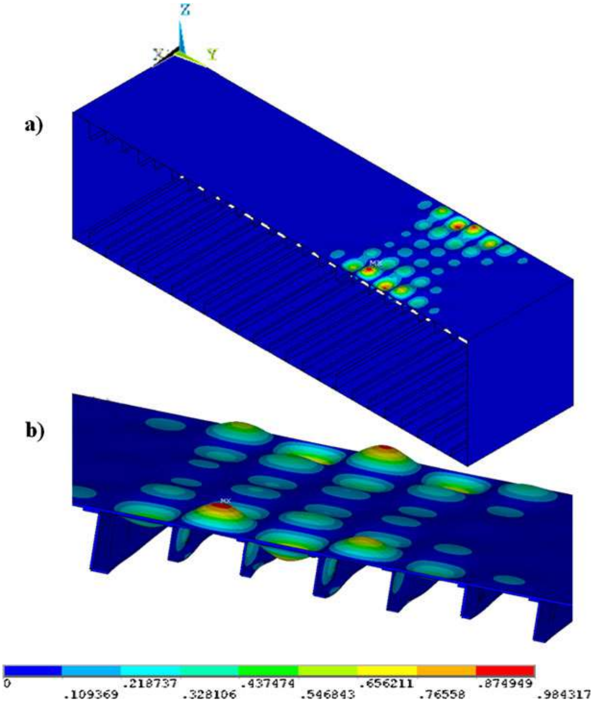
ID	t^S [mm]	h^S [mm]	$(R_{0K}^{A^*})^S$ [MPa]	$(R_1^{A^*})^S$ [MPa]
STIFFENERS				
01	2.75	86.5	−7052.03	11899.10
02	4.75	55.5	−9565.0	1970.67
03	2.625	55.0	−1129.82	14037.45
04	2.125	73.5	18888.6	14574.8
05	3.625	46.0	−5404.69	2750.73
06	4.625	49.0	5701.86	13261.0
07	2.125	58.0	5924.73	11249.3
08	2.0	65.0	−8450.64	8847.51
09	4.0	48.0	14876.8	4495.6
10	4.0	43.0	−1578.69	739.0
11	3.0	43.5	1801.56	7574.78
12	3.75	41.5	8042.03	4290.32
13	3.0	59.0	−1095.8	11495.6
14	4.25	52.0	17811.3	1149.56
15	4.375	54.0	10865.1	2832.84
16	4.0	84.0	12536.7	13178.9
17	2.125	48.5	3993.16	10633.4
18	3.125	48.5	12276.6	14349.0
19	3.0	56.0	12610.9	11536.7
20	2.125	56.5	−6333.33	7615.84
21	4.375	43.0	15322.6	8950.15
22	3.375	56.0	17551.3	5994.13
23	3.625	41.0	13242.4	7020.53

	t [mm]	$R_{0K}^{A^*}$ [MPa]	$R_1^{A^*}$ [MPa]
SKIN			
	4.0	12945.3	882.70

Despite every laminate is quasi-homogeneous and orthotropic, there are some orthotropic laminates with $(K^{A^*})^S = 1$ and others with $(K^{A^*})^S = 0$. The global constrained minimum has been found after 57 generations; see Fig. 13(a). This solution gives a reduction of the weight of the whole structure of about 49 %, when compared to the reference solution and it is very close to the boundary of the feasible domain, $p_{\text{cr}} \simeq p_{\text{ref}}$.

Figure 14(a) shows the evolution of species restrained in the whole population from the initial until the final generation. Since the individuals are randomly sorted in the first generation, also the species are uniformly distributed over the population, i.e., the number of individuals belonging to different species is equiprobable. We can notice that all the species can be found in the initial generations, while some of them *extinguish* during the generations and cannot be found within the final population:

Fig. 12 Example 2: deformed shape of the (**a**) whole wing-box section and (**b**) upper-panel stiffeners



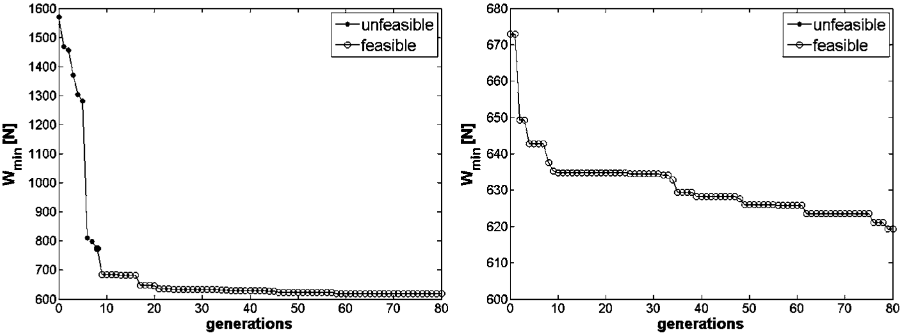


Fig. 13 Best values of the objective function along generations; (**a**) Example 2, (**b**) Example 3

Only solutions with 20 to 23 stiffeners can be found within the population in the last generation. In addition, thanks to the genetic operators detailed in Sect. 2.3, the number of individuals belonging to the fittest species is increased when compared to the initial population. Figure 14(b) shows the variation of the optimal number of stiffeners along the generations: It can be seen that for the first 15 generations the best species is the one showing 22 stiffeners, while from the 16th iteration until the end of the optimization process the best number of stiffeners is 23.

Another point deserves attention; comparing the plots in Figs. 13(a) and 14(b), one can notice something that happens systematically: convergence, toward the best

Table 5 Example 2: best stacking sequences for the optimal solution

ID	*N* of plies	Stacking sequence (°)	Residual
STIFFENERS			
01	22	[28/ − 29/ − 30/28/25/ − 29/ − 29/28/ − 26/42/− 26/26/24/27/ − 29/ − 26/ − 27/25/30/ − 31/ − 22/29]	1.5×10^{-3}
02	38	[40/ − 36/ − 72/ − 38/59/ − 45/24/42/85/ − 20/12/56/43/13/− 41/ − 35/ − 41/45/ − 61/ − 39/56/39/ − 65/7/51/− 70/ − 50/45/ − 28/ − 20/ − 24/45/ − 32/45/41/ − 74/32/ − 31]	3.3×10^{-4}
03	21	[20/20/ − 17/ − 35/ − 16/38/ − 23/ − 24/ − 10/17/29/22/19/− 23/26/ − 27/24/ − 42/ − 18/ − 18/30]	1.9×10^{-3}
04	17	[0/ − 7/0/86/5/2/2/89/2/0/ − 24/1/0/2/ − 87/6/ − 3]	2.1×10^{-3}
05	29	[37/ − 57/ − 3/40/58/ − 44/ − 11/ − 87/50/ − 22/ − 65/ − 38/− 41/23/ − 33/ − 70/18/29/63/41/42/2/61/ − 14/ − 50/87/− 19/29/ − 48]	2.9×10^{-4}
06	37	[6/8/1/61/ − 45/5/ − 24/22/ − 50/6/ − 7/8/7/51/ − 21/− 34/12/ − 21/ − 11/6/33/ − 6/16/78/5/7/ − 6/ − 51/25/− 4/6/18/ − 42/54/ − 12/ − 20/9]	2.4×10^{-5}
07	17	[5/38/−27/−56/−9/7/60/16/−8/7/−29/27/9/−40/81/−8/6]	1.5×10^{-3}
08	16	[−25/ − 47/28/45/ − 16/23/35/1/ − 42/48/ − 41/ − 36/0/− 33/45/24]	5.6×10^{-4}
09	32	[6/ − 1/3/6/77/ − 74/ − 85/ − 21/71/ − 17/ − 86/ − 1/90/ − 1/67/− 85/83/1/1/3/3/1/3/ − 66/ − 6/23/8/ − 77/ − 11/ − 12/85/83]	1.4×10^{-4}
10	32	[−37/79/36/ − 77/ − 1/ − 43/46/52/ − 35/ − 13/90/30/ − 31/− 45/ − 3/36/69/ − 76/87/89/ − 28/ − 57/51/21/− 34/35/14/17/9/ − 73/ − 37/71]	1.3×10^{-3}
11	24	[7/65/15/ − 66/ − 10/ − 9/ − 49/ − 45/ − 22/50/13/5/70/− 22/49/17/19/89/6/ − 33/ − 5/ − 5/ − 45/43]	6.8×10^{-4}
12	30	[2/ − 89/81/ − 15/ − 83/ − 50/ − 19/29/6/34/10/4/ − 10/19/75/− 10/86/3/90/ − 69/ − 39/3/ − 32/87/84/51/0/ − 33/4/21]	5.0×10^{-4}
13	24	[34/ − 45/5/5/ − 30/40/28/3/ − 29/ − 9/ − 25/25/9/ − 21/− 24/ − 56/29/20/55/48/4/ − 39/ − 5/ − 15]	1.0×10^{-3}
14	34	[85/ − 4/ − 75/4/ − 2/84/ − 2/83/88/ − 16/6/2/− 83/0/87/4/3/84/ − 7/84/ − 85/ − 74/86/ − 9/15/ − 86/10/1/− 6/88/ − 3/ − 89/88/5]	1.9×10^{-4}
15	35	[4/ − 79/82/37/ − 2/ − 10/ − 38/ − 70/76/90/10/ − 9/2/− 2/18/ − 77/ − 5/90/ − 3/78/81/ − 21/ − 87/28/9/ − 66/− 13/45/8/ − 14/82/ − 71/78/ − 2/ − 2]	1.6×10^{-4}
16	32	[12/11/ − 7/5/ − 85/ − 9/ − 14/ − 66/50/ − 5/ − 5/− 16/2/87/4/3/3/ − 8/ − 6/19/17/ − 15/17/ − 9/17/14/ − 67/− 9/67/ − 12/4/ − 6]	2.8×10^{-4}
17	17	[−19/20/ − 29/20/ − 71/ − 4/35/62/ − 7/ − 8/3/7/ − 45/− 35/7/70/12]	9.2×10^{-4}
18	25	[13/0/4/ − 23/ − 31/82/18/22/ − 1/2/ − 3/ − 3/ − 30/2/3/14/− 7/18/3/70/ − 77/ − 2/ − 14/ − 6/12]	4.4×10^{-4}
19	24	[2/ − 79/ − 17/27/ − 5/ − 2/ − 5/ − 4/68/ − 5/24/ − 2/ − 83/− 1/4/ − 69/ − 9/8/ − 18/ − 10/25/81/ − 11/7]	3.7×10^{-5}
20	17	[31/46/ − 14/ − 61/ − 38/ − 22/ − 17/25/ − 50/30/47/6/62/28/− 29/3/ − 44]	7.4×10^{-4}

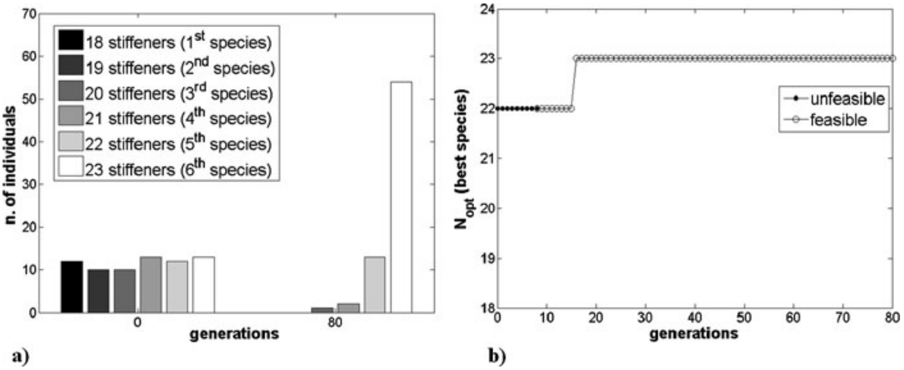


Fig. 14 Evolution of species, Example 2; (**a**) Number of individuals belonging to the same species in initial and final populations, (**b**) Best number of stiffeners vs. generations

Table 5 (*Continued*)

ID	N of plies	Stacking sequence (°)	Residual
21	35	[89/ − 10/9/5/5/ − 6/ − 75/58/ − 4/ − 3/ − 3/ − 2/73/ − 74/ − 2/ − 86/ − 2/ − 3/ − 1/ − 3/22/22/ − 5/ − 11/ − 6/ − 86/ − 82/ − 6/71/ − 9/16/4/ − 82/5/ − 10]	2.2×10^{-4}
22	27	[−5/7/ − 83/5/10/ − 15/ − 85/75/90/ − 4/ − 4/83/0/3/3/3/2/ − 85/ − 85/ − 85/ − 6/1/3/83/7/ − 8/89]	9.3×10^{-4}
23	29	[89/0/12/ − 25/62/10/ − 6/ − 10/ − 1/ − 74/ − 3/ − 1/ − 82/73/ − 86/22/ − 1/26/ − 2/86/10/ − 23/ − 8/ − 10/ − 18/ − 81/13/ − 89/9]	3.3×10^{-4}

N of plies	Stacking sequence (°)	Residual
SKIN		
32	[3/ − 79/9/ − 12/85/47/78/ − 5/ − 8/ − 78/81/ − 73/85/ − 37/ − 86/ − 4/ − 2/ − 90/6/6/ − 90/5/ − 5/25/86/40/ − 8/ − 5/ − 69/90/ − 10/88]	9.6×10^{-5}

value, of the number of modules (here, the number of stiffeners) and of the objective function are independent. They never occur at the same time, and the optimization of the number of modules happens always before that of the objective function. In other words, the strategy used in BIANCA for evolving at the same time species and individuals, normally let attain first the best species, and then continues to evolve individuals within the best species toward the best individual.

Concerning the second problem for this case, since the stiffeners are not identical, we have solved the second level problem for each laminate of the stiffeners and for the skin. Table 5 shows the best stacking sequences found by BIANCA. The number of plies for each laminate, indicated in column 3, is computed considering that the value of the thickness of the elementary layer is 0.125 mm. For the sake of brevity we do not show the polar diagrams and the variation of the best solution for the stiffener and skin laminates. Nevertheless, they are quite similar to those obtained for the wing-box section of the previous example and the same considerations can be done for this second case, also.

4.3 Example 3: Wing-Box Panel with Non-identical Stiffeners, Symmetric Distribution

In this last case, we consider a wing-box section with symmetric distribution of the geometrical and polar parameters for the stiffeners with respect to the x–z plane of the global reference system; see Fig. 7. The genotype of an individual and the genetic parameters are precisely the same of the previous case.

The best solution found by BIANCA is shown in Table 6. The optimal number of stiffeners for the weight minimization is 22, but in Table 6 only one half of the stiffeners, 11 on a whole of 22, is detailed, the other half being symmetrically placed with respect to the x–z plane. The buckling load and the wing-box weight are $p_{\mathrm{cr}} = 1933$ N/mm and $W = 619.26$ N respectively. Figure 15 shows the deformed shape of the structure when the applied load is equal to p_{cr}. Despite this solution has a symmetric distribution of the geometrical and polar parameters of the stiffeners, the buckling is not symmetric. This happens because the structure is not perfectly geometrically symmetric, because the Z-shaped stiffeners are not symmetrically disposed, for practical reasons; consequently the buckling deformed shape is not symmetric too.

Just like in the previous case, the stiffeners do not all belong to the same orthotropy type, some of them having $(K^{A^*})^S = 1$ while others $(K^{A^*})^S = 0$. The global constrained minimum has been found after 78 generations; see Fig. 13(b). This solution gives a reduction of the weight of the whole structure of about 49 %, when compared to the reference solution, and the solution is still very close to the boundary of the feasible domain, $p_{\mathrm{cr}} \simeq p_{\mathrm{ref}}$.

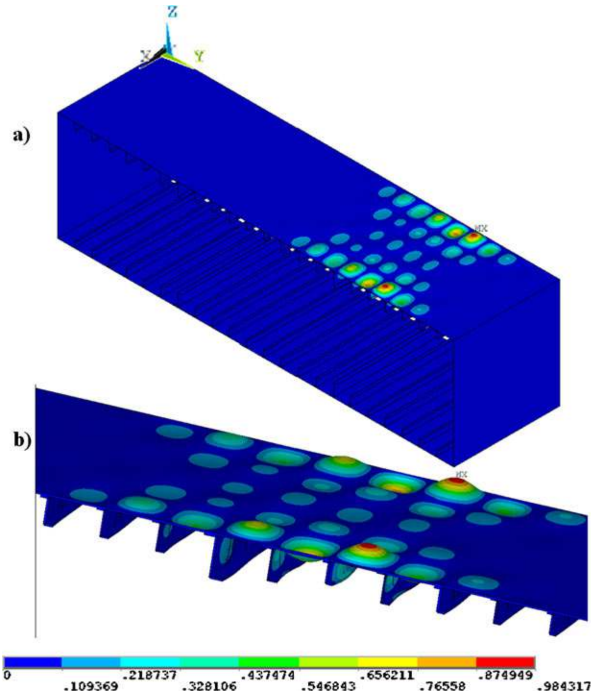
Table 7 shows the best stacking sequences found by BIANCA when the second step is solved for this example. Furthermore, in Table 7, only one half of the stiffeners

Table 6 Example 3: best values of the design variables

ID	t^S [mm]	h^S [mm]	$(R_{0K}^{A^*})^S$ [MPa]	$(R_1^{A^*})^S$ [MPa]
STIFFENERS				
01	4.0	40.0	−10642.20	5850.44
02	2.375	45.0	−5999.02	9648.09
03	4.875	46.5	−3101.66	4844.57
04	2.0	57.0	650.049	14266.9
05	4.25	42.0	16102.6	8211.14
06	2.5	63.0	−1578.69	11557.2
07	4.125	82.5	17477.0	4741.94
08	2.625	55.5	−14545.20	6656.71
09	3.25	52.5	−2990.22	13363.6
10	4.75	53.5	−2581.62	6343.11
11	4.625	43.5	15619.7	11228.7

t [mm]	$R_{0K}^{A^*}$ [MPa]	$R_1^{A^*}$ [MPa]
SKIN		
4.0	9527.86	205.279

Fig. 15 Example 3: deformed shape of the (**a**) whole wing-box section and (**b**) upper-panel stiffeners



is detailed. The remarks done for the case of Example 2 can be rephrased *verbatim* also for this case.

4.4 Verification of the Optimal Stacking Sequences

For the sake of completeness, we have verified the best solution of each one of the three examples above by a finite element analysis, though this is not strictly necessary. The FE model is exactly the same as in the first optimization phase, namely for what concerns boundary and loading conditions. For both the cases of the wing-box section with identical and non-identical stiffeners, the quality of the solution of the second level problem is evaluated. This is done in the following way: The buckling load and the deformed shape of the FE model created directly with the stacking sequences of the laminates are compared to those obtained with the FE model used in the first optimization phase, where the mechanical characteristics were entered through the Cartesian components of tensor $\mathbf{A}^{*}$. So, entering directly the stacking sequences we can assess the effect of the small imperfections of the final stacks with respect to the optimal solution found in the first step problem, imperfections that give rise to the residuals shown in Tables 3, 5, and 7.

For the case of Example 1, the deformed shape is identical to that of Fig. 8, while the value of the buckling load now is $p_{\mathrm{cr}_{\mathrm{ver}}} = 1949$ N/mm, which is slightly greater than $p_{\mathrm{cr}} = 1943$ N/mm found at the end of the first optimization step.

Furthermore, for the case of Example 2, the deformed shape practically does not change with respect to that presented in Fig. 12, and the buckling load is now $p_{\mathrm{cr}_{\mathrm{ver}}} =$

Table 7 Example 3: best stacking sequences for the optimal solution

ID	N of plies	Stacking sequence (°)	Residual
STIFFENERS			
01	32	[31/ − 39/35/ − 54/ − 38/ − 12/ − 30/41/20/ − 56/ − 26/55/55/38/37/31/ − 30/ − 38/39/ − 9/ − 44/ − 43/ − 44/ − 50/19/8/23/35/50/ − 45/54/ − 23]	3.1×10^{-4}
02	19	[28/ − 33/ − 40/42/ − 23/0/ − 20/26/33/17/34/ − 32/ − 39/72/ − 52/8/6/ − 26/36]	1.4×10^{-3}
03	39	[−3/ − 37/39/50/88/29/48/ − 59/9/ − 60/ − 38/3/ − 15/25/ − 29/ − 26/ − 7/33/ − 27/ − 36/ − 71/34/57/ −55/34/81/53/50/12/8/8/−61/−17/76/−26/−35/−1/−55/38]	2.7×10^{-4}
04	16	[30/ − 22/ − 28/ − 27/23/5/ − 10/27/ − 14/27/ − 52/ − 13/14/32/ − 22/7]	2.8×10^{-3}
05	34	[10/ − 12/ − 1/6/ − 82/2/ − 11/77/90/ − 86/ − 88/ − 88/ − 3/1/6/ − 19/10/46/5/6/ − 65/3/ − 6/1/2/ − 3/ − 16/90/89/ − 18/ − 10/9/87/13]	1.1×10^{-3}
06	20	[−7/26/17/ − 42/69/ − 38/22/ − 19/10/ − 6/ − 33/3/36/ − 40/28/1/46/ − 33/17/ − 29]	8.1×10^{-4}
07	33	[5/1/85/ − 4/ − 4/90/ − 86/ − 18/ − 77/8/82/6/ − 3/85/1/7/0/81/1/2/79/ − 3/ − 69/ − 87/0/0/ − 90/ − 21/4/2/ − 88/83/5]	2.3×10^{-4}
08	21	[35/33/ − 38/ − 37/35/ − 36/35/ − 38/ − 38/51/10/ − 42/34/35/34/34/ − 38/ − 38/ − 36/ − 36/34]	5.4×10^{-3}
09	26	[−39/34/17/ − 21/34/ − 7/ − 23/ − 14/ − 21/24/1/33/39/10/ − 46/ − 35/ − 26/26/ − 27/15/27/34/ − 34/ − 24/ − 10/23]	8.9×10^{-4}
10	38	[13/39/ − 57/1/ − 19/ − 55/77/24/10/ − 43/ − 25/55/48/ − 78/ − 15/ − 15/ − 23/35/ − 42/33/8/28/ − 55/ − 12/ − 29/ − 80/18/61/ − 16/7/37/21/ − 55/64/ − 31/47/ − 4/ − 35]	4.3×10^{-4}
11	37	[16/ − 16/7/ − 1/ − 7/ − 18/85/9/ − 4/ − 81/3/88/ − 88/2/0/9/9/ − 4/ − 24/1/ − 8/ − 2/41/ − 2/ − 2/ − 2/9/73/ − 4/ − 72/ − 4/83/ − 4/ − 3/ − 89/ − 5/9]	5.9×10^{-4}

N of plies	Stacking sequence (°)	Residual
SKIN		
32	[80/ − 9/85/ − 1/ − 63/19/7/14/ − 34/54/ − 59/85/89/ − 83/87/ − 13/78/ − 4/ − 90/4/6/71/ − 11/16/ − 70/36/ − 15/ − 70/ − 26/79/13/ − 90]	3.9×10^{-5}

2124 N/mm, about 10 % greater than $p_{\text{cr}} = 1931$ N/mm, found at the end of the first step problem.

Finally, in the case of Example 3, the buckling load is now $p_{\text{cr}_{\text{ver}}} = 2115$ N/mm, which is 9.4 % greater than $p_{\text{cr}} = 1933$ N/mm, the value found at the end of the first optimization step; also in this case, the deformed shape does not change with respect to that shown in Fig. 15.

4.5 Some Remarks on the Type of Laminate's Stacking Sequence

In this section, we want to highlight the importance of the use of non-standard stacking sequences for composite laminated panels. To this purpose, we consider again the Example 1, comparing the results already obtained with the ones that can be found using standard stacking sequences often employed in the aeronautical field, i.e. symmetric sequences with only the values 0°, ±45° and 90° for the ply orientations. The aim is to obtain the outcomes of the first optimization step for Example 1, shown in Table 2. It can be noticed that, for symmetric stacking sequences the coupling stiffness tensor of the laminate is null, so the first term in (29) of Part 1 is identically zero. Concerning the genetic parameters, they are strictly those already used in the previous calculations.

To obtain a standard sequence solution to the second level problem with a sufficiently small residual, we have found that 30 layers are necessary for the stiffeners and 40 for the skin. Table 8 shows the best stacking sequences found by BIANCA when using symmetric stacking sequences with standard orientations. What is apparent is that we need now a higher number of layers than the solution shown in Table 3. This means that the solution found using symmetric stacking sequences with standard orientations is not a global minimum. Indeed, in this case, the weight of the whole wing-box section is 710.23 N, with an increase of about 20 % of the weight of the structure when compared to the non-standard stacking sequence solution shown in Table 3, whose weight is 587.28 N.

Figure 16 is the equivalent of Fig. 10 for the present case of symmetric sequences; the remarks made in Sect. 4.1 can be rephrased *verbatim*, with the only difference that now coupling is exactly null.

Figure 17 shows the variation of the best solution during the iterations, for stiffeners and skin laminates. In particular, we can notice that the best solution is found after 40 generations for the stiffener laminate, while for the laminate of the skin it is found after 150 generations.

To remark that the solutions so found are not balanced, as often used to obtain in-plane orthotropy, but not bending orthotropy, as already recalled in Part 1. Nevertheless, they are orthotropic, and not only in extension, but also in bending. The assumption of balanced stacks will obviously lead to a further increase of the final weight of the structure. So, the use of unconventional stacking sequences, as done in this paper, is really more convenient for the reduction of the weight, as proved by the results shown above.

Table 8 Example 1: best symmetric stacking sequences with standard orientations

N of layers	Stacking sequence	Residual
STIFFENERS		
30	$[-45/45/0_3/90/45/-45/45/0/-45/0/-45/0/45/]_s$	6.6×10^{-4}
SKIN		
40	$[0/90/0/90/0/90_2/0/45/-45/90/0/90/0/90_2/0_2/90/0]_s$	9.6×10^{-5}

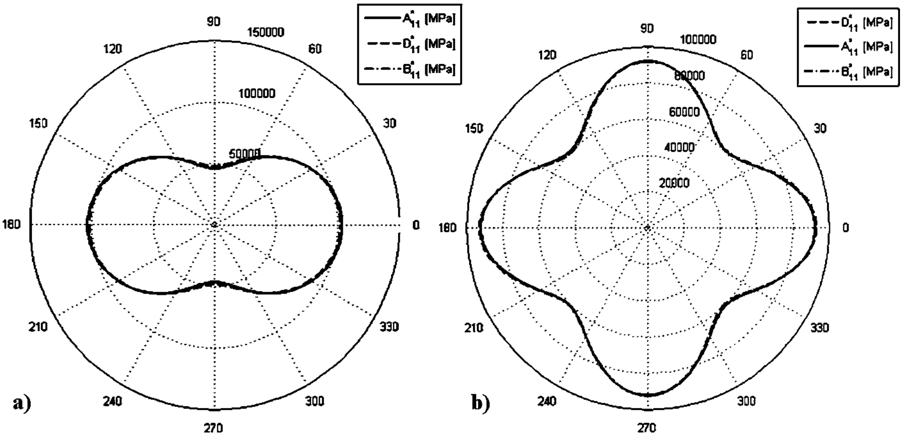


Fig. 16 Example 1 with symmetric stacking sequence and standard orientations: first component of the homogenized stiffness tensors of the laminate for (**a**) stiffeners and (**b**) skin

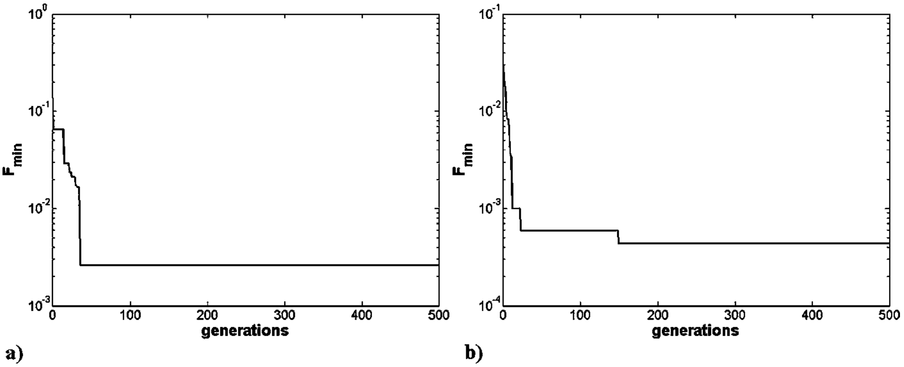


Fig. 17 Example 1 with symmetric stacking sequence and standard orientations: best values of the objective function during iterations for (**a**) stiffeners and (**b**) skin laminates

5 Concluding Remarks

The optimization procedure proposed in this research is characterized by several points that make it an innovative, effective, general method for the design of composite stiffened panels and more generally of modular systems. To the best knowledge of the authors, a similar approach does not exist in the literature, and authors did have not been able to find examples of wing-box optimal designs as general as that presented in the paper. We briefly recall the features of the procedure:

- no simplifying assumptions nor standard rules are used to design the composite structure; this allows for looking for a *true global minimum*, impossible to be obtained otherwise;
- the procedure is composed by two distinct but linked nonlinear minimum problems: The first one is a constrained problem that uses a *free material approach*

to the design of the geometric and material properties of the anisotropic structure, considered as composed by a single-layer fictitious anisotropic material; the second step is an unconstrained problem formulated to design a laminate able to realize the overall optimal mechanical properties designed in the first step; the link between the two problems is twofold: the second step makes use of the mechanical parameters found in the first step for determining a laminate and in the first step the geometrical constraints for the search of a suitable laminate are directly used in the formulation of the minimum problem, in order to have mechanical parameters that can really be obtained with a laminate in the second step;
- quasi-homogeneous sequences are used; this allows for writing exact geometric bounds, valid for both the extension and bending behavior and for reducing the number of mechanical design variables in the first step;
- bending orthotropy is really obtained, its type specified, and the orthotropy direction directly managed, without using special sequences or orientations;
- the number of the modules, i.e., of the layers and of the stiffeners, is directly optimized by the procedure, and this is entirely done by a genetic approach able to select not only individuals, but also species; in practice, the algorithm determines automatically the optimal number of design variables;
- the mechanical characteristics are represented by the polar formalism that gives several advantages, namely to explicit elastic symmetries, elastic and geometric bounds, and to eliminate from the procedure redundant mechanical properties;
- the numerical computations are carried on by a special GA, the code BIANCA, able to cross simultaneously species and individuals, to handle continuous and discrete valued variables during the same iterations and to effectively handle the constraints imposed to the problem;
- for the solution of the first step problem, the code BIANCA has been interfaced with a FE code, in order to numerically compute some mechanical quantities, namely the buckling load;
- the mathematical formulation of the second step problem allows for taking into account for all the possible combinations of elastic requirements and properties; it is stated as an unconstrained minimum problem of a positive semi-definite function, whose absolute minimum is equal to 0, which renders possible to know if a true global minimum has been attained;
- the non-bijective correspondence between the stacking sequences and the elastic properties of a laminate allows for taking into account, in the passage from the first to the second step, several different requirements and actually renders it possible the approach based on the two aforementioned steps.

The results presented in the paper show that when standard rules for the stacks of laminates are abandoned and the design of the optimal number of the modules composing the structure included in the design procedure, significant savings of the weight of the structure can be obtained: up to 50 %, when compared with a classical solution using un aluminum alloy, and up to 20 % when compared with standard aeronautical stacking sequences.

As already said at the end of Part 1, research is going on to include in the design process also other requirements, concerning namely the strength of the structure. Another interesting point on which we are working is to introduce, in the second level problem, the use of the so-called *quasi-trivial solutions* for the quasi-homogeneous laminates; see [15] or [16]. Such solutions automatically give exact quasi-homogeneity, regardless of the orientation angles. So, when they are used, the terms $f_1(\delta)$ and $f_2(\delta)$ in (29) of Part 1 can be discarded and it is easier to obtain a solution for the second level problem.

Other modular problems can be tackled with the use of the code BIANCA with evolution of species; namely, the authors have already worked on the design of minimum weight laminates having some given properties, see [17], while a research is in progress on the optimal design of buildings and towns with nonnegative energy.

A further commentary on the choice of a GA for the solution of the first and second level problem; today, GAs are with no doubt the most widely used numerical techniques to solve design problems of laminates. Their success is strictly linked to two circumstances: first, all the design problems of laminates are, mathematically speaking, characterized by a strong non-convexity. In gradient-like strategies, the solution is very sensitive to the choice of the initial point and for a given problem there are no means to have some indications about a good starting point: the search is completely blind, since its beginning. GAs, working simultaneously, at least in principle, on the whole design space, do not suffer from this dependence upon the initial values.

The second reason for the great success of GAs in solving laminate design problems, is due to the great robustness of GAs, also comparing to other metaheuristics. Especially when constrained problems are to be solved, GAs show a better robustness with respect to other heuristics; like, for instance, PSO (*Particle Swarm Optimization*).

In our case, the use of BIANCA for the first level problem was also justified by another fact: the design of the modular system. With the strategy described in the paper, it was almost natural to reduce the problem of designing the lightest structure with an unknown number of modules to the parallel evolution of species and individuals, as represented in BIANCA by an ordered array of variables. In some sense, we have been guided in this choice by a double natural paradigm: the evolution of individuals *and* of species. This consideration has conducted us to a simultaneous two-level Darwinian strategy. Also, the need of using discrete valued variables, namely for the thicknesses, has been decisive for the choice of a zero order algorithm.

Apart these points, strictly linked to the strategy of the solution, where the use of BIANCA plays a fundamental role for its special possibilities, the use of GAs or of other metaheuristics for solving the first level problem is still well suited because almost nothing is known about its mathematical structure; see concluding remarks of Part 1. Of course, a better knowledge of the mathematical properties of such a problem involving, we recall, the determination of the optimal number of modules, would better motivate the choice of a numerical strategy for its solution.

The use of heuristics has greatly changed the way of approaching and solving complicate optimum problems. Some heuristics can be viewed as the transformation of a sort of *natural intelligence* into an *artificial intelligence* by a numerical procedure. The natural intelligence simulated in a GA, is that of the biological systems, where the objective is the perpetuation of life, based on the survival of the fittest and the transformation of species, for a better adaptation to the natural environment. PSOs are numerical metaphors of the dynamics of a flock of birds, moving all together without the guide of a leader. Ant colony algorithms mimick the strategy used by ants for finding the route to food and for coming back to the colony, and so on.

Regardless the basic idea inspiring a heuristic, all of them are characterized by a major common circumstance: they work on a set of candidates to be the solution of a problem and they need a great quantity of steps, i.e., of objective function evaluations. So, metaheuristics are, in some sense, intrinsically *informatics methods*: their use and success is strictly linked to and dependent from the use of computers. Roughly speaking, mimicking natural intelligence forms needs a huge computing effort: a great quantity of simple computing steps to trace the route to a solution.

In some sense, this is opposite to what, historically, mathematics has always done: to find a solution searching the most efficient way. Classical approaches like descent methods works on only one solution candidate, and transform it using all the informations: the objective function, its derivatives and so on. Normally, these approaches are much more effective than heuristics, i.e., they need a smaller computing effort to find the solution. In addition, and this can be of the greatest importance in some cases, their convergence can be proved, while in the case of heuristics the convergence of the algorithm is not guaranteed, also when the true solution is found.

Unfortunately, in some special cases, classical methods are difficult to be adapted to a given problem, and as already said, they suffer from the sensibility on the initial point, which renders them delicate to be used with non convex problems. In such circumstances, the use of heuristics can become interesting and effective. To our opinion, their use should be restricted to some cases: highly non-convex problems when no indications on suitable initial points are available; mixed type variables, some other special cases, like for instance the first level problem of this work, when the number of modules is to be designed, also.

In all the cases, the choice of a heuristic should never be automatic; too often, researchers use today sort of *black-box heuristics*, because they are simple to be used. This is never a good reason; a mathematical analysis of the problem is, to our opinion, to be done in all the cases. This can considerably help in conceiving more direct, mathematically rigorous methods to solve the problem at hand. In the case of laminates optimum design, we are now going on with a problem concerning the optimal design of anisotropy fields for a laminate with respect to strength criteria. Also, in this case, the problem is formulated by a two-level approach, similar to the one used in this paper. Nevertheless, while the second step is practically identical to that presented here, and BIANCA is still used, the first level problem is quite different and we are using a different approach, which is based upon an algorithm composed by three steps, whose convergence can be proved. Research is going on in this direction, but it is important to notice that we could not do the same for the problem of the present work, because of the variable number of modules, to be designed, also. For solving this problem, our inspiration has been a simulation of the Darwinian selection at two levels, species and individuals, which has been decisive in the choice of the GA BIANCA as a numerical tool.

Acknowledgements Authors wish to thank Professor Franco Giannessi. His helpful suggestions have considerably contributed to improving the quality of the paper. FNR of Luxembourg, supporting the first author through *Aides à la Formation Recherche* Grant PHD-09-139, is gratefully acknowledged.

References

1. Goldberg, D.E.: Genetic Algorithms. Addison-Wesley, New York (1994)
2. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolutionary Programming. Springer, Berlin (1994)
3. Nagendra, S., Jestin, D., Gürdal, Z., Hatfka, R.T., Watson, L.T.: Improved genetic algorithm for the design of stiffened composite panels. Comput. Struct. **58**, 543–555 (1996)
4. Kaletta, P., Wolf, K.: Optimisation of composite aircraft panels using evolutionary computation methods. In: Proceedings of ICAS 2000 Congress, Harrogate, UK, 27 August–1 September 2000
5. Muc, A., Gurba, W.: Genetic algorithms and finite element analysis in optimization of composite structures. Comput. Struct. **54**, 275–281 (2001)
6. Tabakov, P.Y.: Multi-dimensional optimisation of laminated structures using an improved genetic algorithm. Comput. Struct. **54**, 349–354 (2001)
7. Coello Coello, C.A.: Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. Comput. Methods Appl. Mech. Eng. **191**, 1245–1287 (2002)
8. Vincenti, A., Ahmadian, M.R., Vannucci, P.: BIANCA: a genetic algorithm to solve hard combinatorial optimisation problems in engineering. J. Glob. Optim. **48**, 399–421 (2010)
9. Montemurro, M., Vincenti, A., Vannucci, P., Makradi, A.: Optimisation en poids de structures composites stratifiées. In: Proc. of JNC17—17 Journées Nationales sur les Composites, Poitiers (2011). http://hal.archives-ouvertes.fr/docs/00/60/30/56/PDF/143.pdf
10. Kim, I.Y., de Weck, O.I.: Variable chromosome length genetic algorithm for progressive refinement in topology optimization. Struct. Multidiscip. Optim. **29**, 445–456 (2005)
11. Ryoo, J., Hajela, P.: Handling variable string lengths in ga-based structural topology optimization. Struct. Multidiscip. Optim. **26**, 318–325 (2004)
12. Park, C.H., Lee, W.I., Han, W.S., Vautrin, A.: Multiconstraint optimisation of composite structures manufactured by resin transfer moulding process. J. Compos. Mater. **39**, 347–374 (2005)
13. Vincenti, A., Vannucci, P., Ahmadian, M.R.: Optimization of laminated composites by using genetic algorithm and the polar description of plane anisotropy. Mech. Adv. Mater. Struct. doi:10.1080/15376494.2011.563415 (2012, in press)
14. Vannucci, P.: Designing the elastic properties of laminates as an optimisation problem: a unified approach based on polar tensor invariants. Struct. Multidiscip. Optim. **31**, 378–387 (2006)
15. Vannucci, P., Verchery, G.: A special class of uncoupled and quasi-homogeneous laminates. Compos. Sci. Technol. **61**, 1465–1473 (2001)
16. Vannucci, P., Verchery, G.: Stiffness design of laminates using the polar method. Int. J. Solids Struct. **38**, 9281–9294 (2001)
17. Montemurro, M., Vincenti, A., Vannucci, P.: A completely genetic approach to the design of laminates with minimum number of plies. HAL Archives Ouvertes, hal-00673132, http://hal.archives-ouvertes.fr/docs/00/67/31/32/PDF/paper.pdf (2012)