J Glob Optim (2009) 45:427–449
DOI 10.1007/s10898-008-9383-x

Optimization of aircraft structural components by using nature-inspired algorithms and multi-fidelity approximations

Felipe A. C. Viana · Valder Steffen Jr. ·
Sergio Butkewitsch · Marcus de Freitas Leal

Received: 21 May 2007 / Accepted: 24 November 2008 / Published online: 7 December 2008

Abstract In this work, a flat pressure bulkhead reinforced by an array of beams is designed using a suite of heuristic optimization methods (Ant Colony Optimization, Genetic Algorithms, Particle Swarm Optimization and LifeCycle Optimization), and the Nelder-Mead simplex direct search method. The compromise between numerical performance and computational cost is addressed, calling for inexpensive, yet accurate analysis procedures. At this point, variable fidelity is proposed as a tradeoff solution. The difference between the low-fidelity and high-fidelity models at several points is used to fit a surrogate that corrects the low-fidelity model at other points. This allows faster linear analyses during the optimization; whilst a reduced set of expensive non-linear analyses are run "off-line," enhancing the linear results according to the physics of the structure. Numerical results report the success of the proposed methodology when applied to aircraft structural components. The main conclusions of the work are (i) the variable fidelity approach enabled the use of intensive computing heuristic optimization techniques; and (ii) this framework succeeded in exploring the design space, providing good initial designs for classical optimization techniques. The final design is obtained when validating the candidate solutions issued from both heuristic and classical optimization. Then, the best design can be chosen by direct comparison of the high-fidelity responses.

Keywords Ant Colony Optimization · Genetic Algorithms · Particle Swarm Optimization · Variable fidelity design · Pressure bulkhead optimization

F. A. C. Viana · V. Steffen Jr. (✉)
School of Mechanical Engineering, Federal University of Uberlandia, Av. João Naves de Ávila 2121, Campus Santa Mônica, Uberlandia, MG 38400-902, Brazil
e-mail: vsteffen@mecanica.ufu.br

S. Butkewitsch · M. de Freitas Leal
Embraer – Empresa Brasileira de Aeronautica S.A , Av. Brigadeiro Faria Lima 2170, São José dos Campos, São Paulo 12227-901, Brazil

1 Introduction

Cabin pressurization has made it possible for aircraft to fly under different weather conditions and landscape formations, thus giving a significant contribution to air travel safety. From the structural viewpoint, the pressurized cabin of a modern aircraft is a system of sealed pressure vessels containing an atmosphere near to that of sea level. Its functional requirements include [1]:

- the transmission of internal and external flight loads;
- the necessity for non-structural cutouts such as doors and windows;
- an efficient shape for both aerodynamics and space allocation; and
- minimum structural weight.

More specifically, devices called pressure bulkheads close the extremities of pressurized cabins. These bulkheads are designed to blow first should the cabin pressure become too high as opposed to that of the main fuselage of the aircraft. At this point, additional complexities in systems and structures have to be taken into account, such as those involved in bearing the pressure loads and performing and controlling the air flow associated with pressurization. According to Niu and Niu in [2], pressure bulkheads should have a dome-like structure in preference to a flat one. It is also known, however, that some requirements, mostly related to space availability, impose the adoption of the flat geometry. The importance of this distinction is due to the completely different mechanical behaviors of each of these configurations.

A curved dome ideally tends to support the lateral pressures developed at its curved walls by tensile stresses alone. These stresses, called membrane stresses, occur in tangential directions at each point and are called membrane stresses. In reality, some amount of bending stiffness also occurs, leading to the development of flexural (tension + compression) stress fields. Such behavior becomes prevalent as the surface of the bulkhead goes flat. The true bulkheads will lie somewhere in-between the curved and flat types, and will possibly include these two extremes. Maximum efficiency could be obtained by minimizing the departures from the total membrane stress system.

It is easy to see that the enormous challenge involved in performing a design optimization of pressure bulkheads calls for advanced techniques of numerical optimization. Venter and Sobieszczanski-Sobieski in [3] discussed that in recent years non-gradient nature based, probabilistic search algorithms (which generally mimic some natural phenomena) have attracted much attention from the scientific community due to features such as easiness to code, ability of handle non-continuous functions, capability of using parallel architectures, and the trend of finding the global, or near global, solution. Even though the large number of function evaluations required by these methods often presents a drawback when compared with classical gradient-based algorithms, advances in computational throughput have helped to use this class of algorithm in real world problems, as can be seen in Venter and Sobieszczanski-Sobieski [4] and Viana et al. [5]. At this point, the concept of multi-fidelity approximations [6–9] presents an interesting alternative to reduce the computational effort required during the evaluation of the objective function, while preserving an acceptable fidelity level. In this technique, the advantages of high-fidelity and low-fidelity models are used in an optimization process. The high-fidelity model provides solution accuracy while the low-fidelity model reduces the computational cost. Giunta et al. in [6] used variable complexity (or fidelity) response surface modeling in the wing design of a high-speed civil transport aircraft to overcome computational expense. They used low, medium and high fidelity aircraft analysis methods in that low fidelity methods include algebraic expressions for estimating lift and drag, medium fidelity methods include linear theory aerodynamic analyses, and high fidelity methods include Navier-Stokes equations. Vitali et al. in [9] used this concept to combine a high fidelity analysis model with a low fidelity model to calculate the crack propagation constraint in the design optimization process. Correction response surfaces were employed to relate the high fidelity models to the low fidelity models. The authors concluded that the multi fidelity approach was found to be more accurate than the single fidelity response surface method at the same computational cost.

The present work presents two main contributions. The first one is the optimal design of a totally flat pressure bulkhead. The second contribution is the use of multi-fidelity approximations combined with heuristic optimization methods in the solution of a real world problem. The authors aimed at testing heuristic methods in real world applications, and to take advantage of the robustness found in these techniques to handle the design optimization of the flat pressure bulkhead. The remaining of the paper is organized as follows. Section 2 reviews the theory behind and presents the numerical modeling of the pressure bulkhead. Section 3 discusses the variable-fidelity approach. Section 4 presents the optimization algorithms. Section 5 gives details about the case study. Section 6 exposes the results and discussions. Finally, the paper is closed by recapitulating salient points and concluding remarks.

2 Using plates for the design and analysis of aircraft pressure bulkheads

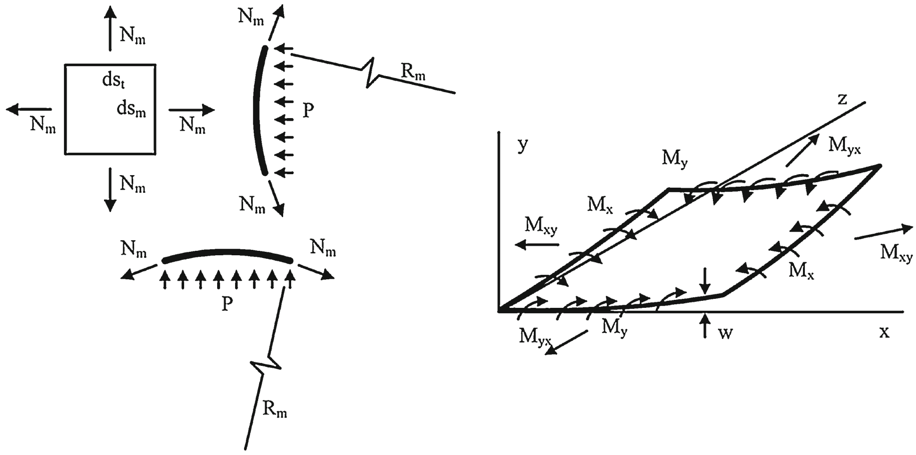
Although numerical models are actually used to account for specific geometries and boundary conditions, this section will pursue an analytical development that describes the mechanical behavior of plates in bending, aiming at achieving the insight necessary to implement and interpret the results of an optimization procedure devoted to the design a flat pressure bulkhead.

As discussed by Przemieniecki in [10], ideal plates can be understood as the two-dimensional counterparts of the beams, since they resist transversal loads by bending and shear stresses, without mechanical action at the neutral surface. Referring to Fig. 1, consider the initially flat plate being actuated by bending moments (per unit length) M_x and M_y, and also by twisting couples ($M_{xy} = -M_{yx}$). As in the case of beams, the curvature (second derivative of the displacement w with respect to the spatial coordinates x and y) and the applied moments are related to each other by the bending stiffness (product of the axial elasticity modulus $E$by the area moment of inertia I). Thus, considering that for a unit width plate $I = t^3/12$ and that the Poisson (υ) effect induces a negative curvature at the plane perpendicular to the bending axis:

$$\begin{cases} \dfrac{\partial^2 w}{\partial x^2} = \dfrac{12}{Et^3} \cdot \left(M_x - \upsilon \cdot M_y\right) \\ \dfrac{\partial^2 w}{\partial y^2} = \dfrac{12}{Et^3} \cdot \left(M_y - \upsilon \cdot M_x\right) \end{cases} \tag{1}$$

Equation (1) is usually rearranged such that the moments are expressed as functions of the curvature:

$$\begin{cases} M_x = D \cdot \left(\dfrac{\partial^2 w}{\partial x^2} + \upsilon \cdot \dfrac{\partial^2 w}{\partial y^2}\right) \\ M_y = D \cdot \left(\dfrac{\partial^2 w}{\partial y^2} + \upsilon \cdot \dfrac{\partial^2 w}{\partial x^2}\right) \end{cases} \tag{2}$$



(a) membrane stresses

(b) plate in bending

Fig. 1 Plate under stress and in bending

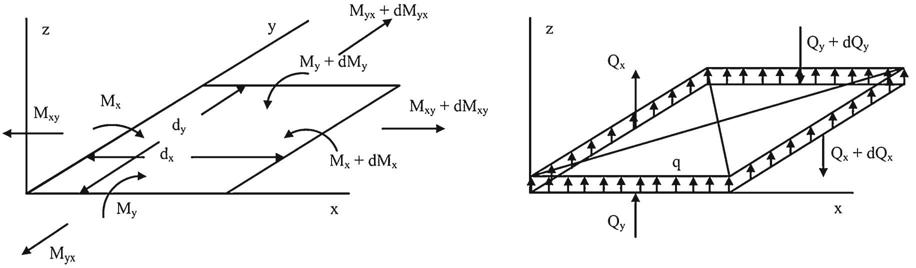


Fig. 2 Plate in bending under the action of transversal loads

where the generalized stiffness D is given by:

$$D = \frac{E \cdot t^3}{12} \cdot \left(1 - \upsilon^2\right) \tag{3}$$

Analogously, the twist of the element (change in x-direction slope per y-direction unit distance) can be expressed as in Eq. (4), as demonstrated by Den Hartog in [11]:

$$M_{xy} = D \cdot (1 - \upsilon) \cdot \frac{\partial^2 w}{\partial x \cdot \partial y} \tag{4}$$

Besides relating moments to curvatures in pure bending, consider the superposition of shear forces (Q) and transverse pressure (q) as indicated in Fig. 2.

Due to these additional lateral loads, the bending and twisting moments vary along the plate, giving rise to:

$$\begin{cases} Q_x = \dfrac{\partial M_x}{\partial x} + \dfrac{\partial M_{xy}}{\partial y} \\ Q_y = \dfrac{\partial M_y}{\partial y} + \dfrac{\partial M_{xy}}{\partial x} \end{cases} \tag{5}$$

By summing the moments about x and y axis, and similarly, by adding up the forces along the axis perpendicular to the plate, one obtains:

$$q = \frac{\partial Q_x}{\partial x} + \frac{\partial Q_y}{\partial y} \tag{6}$$

Finally, by eliminating all internal forces (M_x, M_y, M_{xy}, Q_x and Q_y) in the above equations, it is possible to find the relation between the transverse pressure q and the transverse deflection field (w), accounting for the stiffness D:

$$\frac{\partial^4 w}{\partial x^4} + \frac{\partial^4 w}{\partial y^4} + 2 \cdot \frac{\partial^4 w}{\partial x^2 \cdot \partial y^2} = \frac{q}{D} \tag{7}$$

In a typical plate bending analysis, the highly non-linear Eq. (7) is integrated for a specified set of boundary conditions. The resulting displacement field can then be fed back into Eqs. (2)–(5) to recover the stresses and internal forces.

Equation (7) highlights the dominance of the bending stiffness in resisting the transversal pressure load. Using a thicker web for the bulkhead increases the bending stiffness, although this may not be the most effective choice with respect to structural weight. The choice of higher reinforcement beams is potentially better, albeit subject to space availability constraints in the direction perpendicular to the bulkhead web. Less efficient, but still capable of contributing somehow, the width and thickness of the reinforced beams could also be manipulated to increase the overall bending stiffness. A suitable compromise combining all of these potential design variables could lead to an optimum design.

For a more realistic analysis, it should be considered, however, that when the carrying of transverse loads entails large (more than a few tenths of the plate thickness) deflections in the same direction (see Fig. 3), the plates deform into curved surfaces, giving rise to mid-surface stresses, not considered in the derivations from Eqs. (1)–(7). Still, the edge supports resist the in-plane movements of the plates, leading to further membrane (mid-surface) action.

All in all, the already highly non-linear relation expressed in Eq. (7) needs to be further complicated by the inclusion of mid-surface stresses for an accurate consideration of the effects involved in the mechanical analysis of a flat bulkhead. Besides, the exact boundary conditions needed to integrate it are unclear, since the plate restraints are elastic.

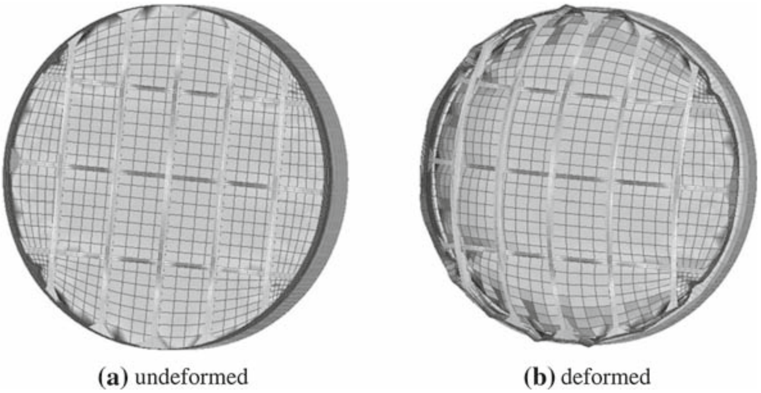

(a) undeformed **(b)** deformed

Fig. 3 Post-processing displacements indicating large deflections at a pressurized flat bulkhead reinforced by beams

From the analysis point of view, all previous discussion justifies the use of non-linear finite element modeling to deal with such complex calculations. On the other hand, within a design optimization framework, several analyses are required iteratively and, due to their usually high computational cost, a feasibility obstacle can arise. In this scenario, the next sections are devoted to the discussion of aspects relative to the use of optimization techniques in the context outlined so far.

3 Variable fidelity models applied to optimization in engineering

For the purposes of the present work, the optimal design problem is stated as the nonlinear, constrained problem of finding the point in the design space, $\mathbf{x}^*$, that will minimize the objective function (or vector of objective functions) for a given set of system parameters [12]:

$$\mathbf{x}^* = \arg\left(\min \mathbf{f}(\mathbf{x})\right) \tag{8}$$

subject to:

$$\begin{array}{ll} x_i^l \leq x_i \leq x_i^u, & i = 1, 2, \ldots, n_{dv} \\ g_j(\mathbf{x}) \leq 0 & j = 1, 2, \ldots, n_{ineqcnstrt} \\ h_j(\mathbf{x}) = 0 & j = m+1, m+2, \ldots, n_{cnstrt} \end{array} \tag{9}$$

where:

- $\mathbf{f}(\mathbf{x}) = \left[f_1(\mathbf{x}) \quad f_2(\mathbf{x}) \ldots f_{n_{obj}}(\mathbf{x})\right]^T$ is the vector of objective functions. This vector is composed by objective functions that can eventually support but more commonly conflict with each other.
- $x_i^l \leq x_i \leq x_i^u$ imposes the side constraints to the design space, and
- $g_j(\mathbf{x})$ and $h_j(\mathbf{x})$ are the inequality and equality constraints, respectively.

In engineering problems, high-fidelity models are typically time consuming and computationally expensive. As a result, the computational cost of complex high-fidelity engineering simulations often makes it impractical to rely exclusively on simulation for design optimization [13]. For this reason, the use of surrogates models (also known as meta-models or approximations) to represent the functions involved in an optimization problem has become an established approach [14,15]. The statistical procedure used to generate them can be summarized as follows:

1. Formulate the problem: which includes establishing the goals of the investigation, identifying the key independent variables and responses and if possible, postulating of a model that relates these variables.
2. Design of experiments: the design space is sampled in order to reveal its contents and tendencies [16]. At this step, the gain of as much information as possible must be balanced with the cost of simulation/experimentation.
3. Choice of surrogate model: the nature of the surrogate itself is determined, tacking into account that the relations contained in the data gathered in the previous step have to be represented, with the highest possible accuracy. Besides the traditional Polynomial Response Surface (PRS) [17,18], more sophisticated and more expensive surrogates have become popular. Surrogates such as Radial Basis Neural Networks (RBNN) [19,20], Kriging models (KRG) [21], [22], and Support Vector Regression (SVR) [23,24] that require optimization in the fitting process, increasingly replace PRS, which only requires the solution of a system of linear equations.

4. Model fitting: the model whose shape is defined in “3” is fitted to the data collected in “2”.
5. Assess quality of fit: the precedent steps are sufficient to build a first tentative model, which overall quality and usefulness has to be evaluated by adequate sets of metrics [17,21], and [25].

Even though surrogate models alleviate the use of optimization techniques, constructing them may still require that a significant number of analyses be performed. When high fidelity analysis results are used in constructing surrogates, the computational cost associated with generating them becomes huge as the number of design variables increases. On the other hand, if low fidelity analysis results are used in constructing surrogates, then accuracy becomes a problem. Under these conditions, a multi-fidelity approach can be used to reduce the computational cost and to produce acceptable accuracy [6–9]. The framework used to vary the fidelity is graphically represented in Fig. 8.

The following rationale describes how different fidelity levels are used in optimization; it should further clarify the purposes of the variable fidelity approach:

1. Direct coupling of the optimizer and high fidelity analysis: this might be the most accurate approach due to the high fidelity. Moreover, it does not have statistical fitting limitations, since the optimizer and the analysis module are linked directly, and not by means of a surrogate. However, the computational overhead and the possibility of numerical ill-conditioning often prevent this approach from being applied.
2. Coupling of the optimizer and a statistical surrogate of the high fidelity analysis: in view of the drawbacks of the previous approach, a very common alternative is coupling the optimizer to a statistical surrogate of the high fidelity analyses. This alternative has the potential to reduce computational cost and improve the numerical conditioning. There is no fidelity problem, since the surrogate is constructed through the high fidelity data. However, statistical fitting problems may appear, mainly with global regression approximations.
3. Coupling of the optimizer and the low fidelity analysis: in another scenario, if the optimization is performed directly over the linear models, lack of accuracy due to their low fidelity may occur. This approach would be advantageous from the viewpoint of computational cost and absence of statistical inaccuracies, since surrogates are not used.
4. Coupling the optimizer and variable fidelity analyses: weighting the strengths (low computational cost, fidelity accuracy and statistical accuracy) and the weaknesses (long runtimes, low fidelity and statistical limitations) that characterize one or another of the previous approaches, variable fidelity can be the most suitable strategy:
    (a) Low fidelity analysis is coupled directly to the optimizer, ensuring faster computations and lack of statistical inaccuracies.
    (b) The surrogate that is constructed is not intended to model the design space, but to model the error between two homologous spaces, for the same quantity evaluated at different fidelity levels. Statistical inaccuracies may occur with surrogates of a given response because all the points used to construct it have to be, in principle, uncorrelated. For modeling the error, the situation is much more advantageous, since a natural correlation arises from the fact that the error results from different computations held at the same point. As an effect, surrogates for the error tend to be more accurate for a given number of design variables and sampled points.

Table 1 summarizes the previous discussion.

Table 1 Comparison of different approaches to use analysis in optimization. It can be observed that although high fidelity analysis presents the best accuracy, it may imply in numerical obstacles for the optimization, such non-linearities and difficult gradient calculation. On the other hand, the use of low fidelity may hurt the accuracy level. Thus, the direct use of surrogates and variable fidelity appear as the most suitable strategies in optimization

Approach		Computational cost	Fidelity related accuracy	Numerical conditioning
1	High fidelity analysis	Highest	Highest	Worst
2	Surrogate of the high fidelity analysis	Lowest	In general, problem dependent	Best
3	Low fidelity analysis	Intermediate	Low	Acceptable
4	Variable fidelity	Intermediate	In general, acceptable	Acceptable

Correction surrogates couple high-fidelity and low-fidelity methods of analysis at a number of points in the design domain. In this work, the surrogate for the difference between the analyses is used. The process starts with high and low-fidelity computation of the response of interest, $f(\mathbf{x})$, at different p points. At these points, the difference, $\delta(\mathbf{x})$:

$$\delta(\mathbf{x}) = f_{HF}(\mathbf{x}) - f_{LF}(\mathbf{x}) \tag{10}$$

is computed. The subscripts HF and LF indicate the value of $f(\mathbf{x})$ obtained from high-fidelity and low-fidelity models, respectively.

In the context of this paper, low fidelity analysis means the use of relatively coarse meshed linear static finite element models, such as those usually intended for global structural analysis (for determination of load paths accounting for the stiffness of the structural components). In this case, the bulkhead web is represented by plate elements and the reinforcements by beams. In opposition, high fidelity calculations are performed with much more refined models, suitable for non-linear static analyses. In this case, all details are represented by plate elements, including the reinforcing beams.

The next step is to build the surrogate model $\hat{\delta}(\mathbf{x})$ for $\delta(\mathbf{x})$. Here, there is no surrogate model to be preferably used. The variable-fidelity approximation to $f(\mathbf{x})$ at any other point, $\mathbf{x}$, is obtained from a low-fidelity analysis as:

$$\hat{f}(\mathbf{x}) \cong \hat{\delta}(\mathbf{x}) + f_{LF}(\mathbf{x}) \tag{11}$$

In the variable fidelity scenario, the use of low fidelity models has to be maximized, so that they are used only at the core of the calculations needed for the optimization procedure. On the other hand, the use of high fidelity analyses is kept to a minimum, i.e. just the necessary to fit $\hat{\delta}(\mathbf{x})$.

Once the correction surrogates are built, the optimization problem can be solved by using an adequate algorithm. The common practice is to use classical gradient-based algorithms [6–9]. It means that the optimizer receives the values of the functions at the point plus those used for obtaining gradient information from the lower-fidelity model to build internal local models used during the optimization task (e.g. Taylor series). On top of that, in order to avoid issues with local minima, starting the optimizer with different initial design is also a common practice. In terms of low-fidelity simulations, the total cost of this approach is a function of the number of variables (related to the cost of the gradient computation) and the number of iterations of the optimization task.

In this work, the use of heuristic nature-inspired optimization algorithms is proposed. The main reasons are the following:

- They do not require gradient information: which implies that the resources can be directly used for the search, and that there is no propagation of the errors due to the computation of the gradients based on corrected responses (problem reported in [26]).
- They have the trend to find the global or near global solution: which reduces the need of running multiple times the optimization task.

In addition, in order to complete the heuristic optimization, a Nelder-Mead simplex direct search (NMSDS) [27] is employed with initial designs given by the candidate solutions of the heuristic optimizers. NMSDS also does not use numerical or analytic gradients. A simplex in the n_{dv}-dimensional space is characterized by the $n_{dv} + 1$ distinct points in its vertices (e.g., if $n_{dv} = 2$, a simplex is a triangle). At each step of the search, a new point in the neighborhood of the current simplex is generated. One of the vertices is replaced by this new point if the new function value is smaller than the values at the vertices of the simplex (which generates a new simplex). This step is repeated until the diameter of the simplex is less than a specified tolerance. The MATLAB *fminsearch* function (set with the default options) is used as implementation of the NMSDS algorithm. This cascade-type scheme using a heuristic algorithm for global search and a classical algorithm for local search is believed to reduce the changes of fail because of local minima.

4 Nature-inspired optimization methods

In this work, a suite of heuristic methods is employed to solve the design optimization problem of aircraft structural components. Ant Colony Optimization, Genetic Algorithms, Particle Swarm Optimization (PSO) and LifeCycle Model (LC) are the approaches used to perform this case study. These methods are briefly reviewed below.

4.1 Ant colony optimization (ACO)

ACO was introduced by Marco Dorigo in his doctoral thesis in 1992 [28]. ACO is inspired in the behavior of real ants and their communication scheme by using pheromone trail. When searching for food, real ants start moving randomly, and upon finding food they return to their colony while laying down pheromone trails [29]. This means that if other ants find such a path, they return and reinforce it. However, over time the pheromone trail starts to evaporate, thus reducing its attractive strength. When a short and a long path are compared, it is easy to see that a short path gets marched faster and thus the pheromone density remains high. Consequently, if one ant finds a short path (from the optimization point of view, it means a good solution) when marching from the colony to a food source, other ants are more likely to follow that path, and positive feedback eventually encourages all the ants in following the same single path. ACO follows some basic concepts, as presented below [28,29]:

- A search performed by a population of ants, i.e., by simple independent agents.
- Incremental construction of solutions.

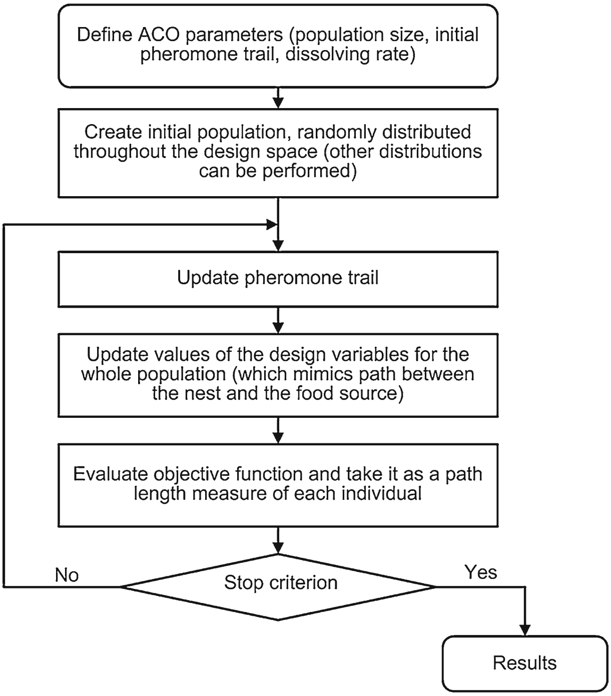

Define ACO parameters (population size, initial pheromone trail, dissolving rate)
Create initial population, randomly distributed throughout the design space (other distributions can be performed)
Update pheromone trail
Update values of the design variables for the whole population (which mimics path between the nest and the food source)
Evaluate objective function and take it as a path length measure of each individual
No
Stop criterion
Yes
Results

Fig. 4 ACO basic algorithm

- Probabilistic choice of solution components based on stigmergic information of pheromone. A stigmergic process is the process through which the results of a worker insect's activity act as a stimulus for further activities.
- No direct communication between ants.

ACO has attracted much attention from the research community due to its efficiency in solving combinatorial optimization problems, such as the routing problem in a computer network [30] and more recently continuous problems, such as in the design of vibration damping devices [5].

The outline of a basic ACO is as shown in Fig. 4.

More details about ACO are provided by [5], and [28–30].

4.2 Genetic algorithm (GA)

GA is an optimization algorithm used to find approximate solutions to difficult-to-solve problems through the application of the principles of evolutionary biology to computer science. GA is based on Darwin's theory of survival and evolution of species [31–33]. GA uses biologically-derived concepts such as inheritance, mutation, natural selection, and recombination (or crossover). Due to the vast literature on GA, for the sake of simplicity, just an overview is provided in this text.

The algorithm starts from a population of random individuals, viewed as candidate solutions to a problem. During the evolutionary process, each individual of the population is

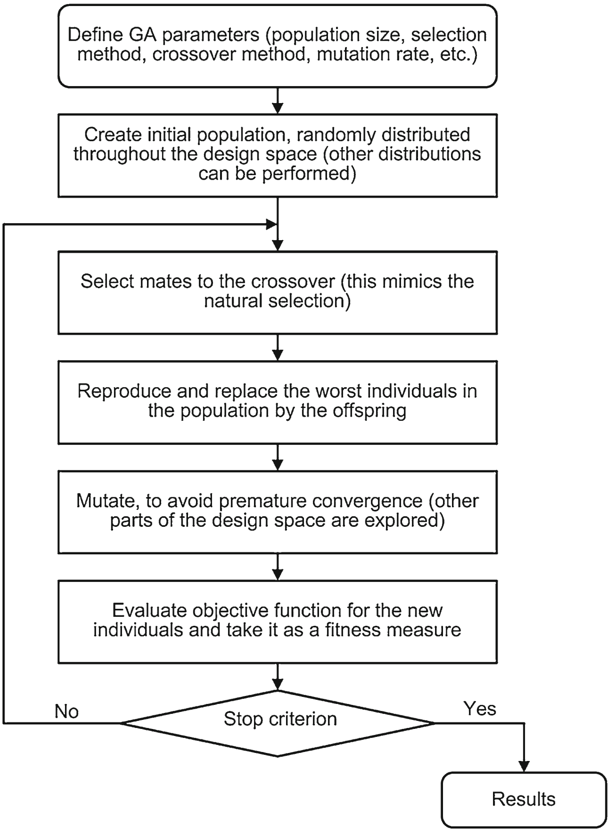

Define GA parameters (population size, selection method, crossover method, mutation rate, etc.)
Create initial population, randomly distributed throughout the design space (other distributions can be performed)
Select mates to the crossover (this mimics the natural selection)
Reproduce and replace the worst individuals in the population by the offspring
Mutate, to avoid premature convergence (other parts of the design space are explored)
Evaluate objective function for the new individuals and take it as a fitness measure
Stop criterion
No
Yes
Results

Fig. 5 GA basic algorithm

evaluated, reflecting its adaptation capability to the environment. Some of the individuals of the population are preserved while others are discarded; this process mimics the natural selection in Darwinism. The members of the remaining group of individuals are paired in order to generate new individuals to replace those that are discarded in the selection process. Finally, some of them can be submitted to mutation, and as a consequence, the chromosomes of these individuals are altered. The entire process is repeated until a satisfactory solution is found. The outline of a basic GA is as shown in Fig. 5.

Although the initially proposed GA algorithm was dedicated to discrete variables only, nowadays improvements are available to deal with discrete and continuous variables. See [31–33] for further details.

4.3 Particle swarm optimization (PSO)

PSO was introduced by Kennedy and Eberhart in [34], as emerged from experiences with algorithms inspired in the social behavior of some bird species. Consider the following situation: a swarm of birds searching for food around a delimited area. Suppose there is just one place with food and the birds do not know where it is. The success of one bird is shared by the whole swarm (learning from the experience of other individuals). In this sense, the

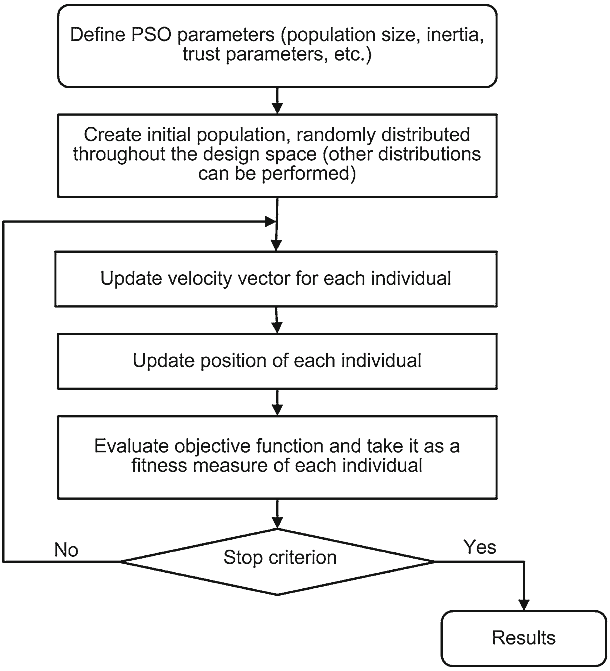

Define PSO parameters (population size, inertia, trust parameters, etc.)
Create initial population, randomly distributed throughout the design space (other distributions can be performed)
Update velocity vector for each individual
Update position of each individual
Evaluate objective function and take it as a fitness measure of each individual
Stop criterion
No
Yes
Results

Fig. 6 PSO basic algorithm

adjustment between exploration (the capacity of individual search) and exploitation (taking advantage of someone else's success) is required. If there is little exploration, the birds will all converge on the first good place encountered. On the other hand, if there is little exploitation, the birds will slowly converge or they will try to find food alone. It is clear that the best policy is a trade-off between both policies.

In PSO, the flight of each bird (individual of the population) is modeled by using a velocity vector, which considers the contribution of the current velocity, as well as two other parts accounting for the self-knowledge of the individual and the knowledge of the swarm (herein referred to as population) about the search space. This way, the velocity vector is used to update the position of each individual in the population [3,4], and [34]. An outline of a basic PSO algorithm is shown in Fig. 6.

PSO is comprehensively presented in [3,4], and [34].

4.4 Lifecycle model

LC is a hybrid nature-inspired optimization method inspired by the idea of life cycle stages, initially proposed by Krink and Løvberg in [35]. From the mathematical point of view, natural algorithms such as GA, PSO and ACO are heuristic search methods of proven efficiency as optimization tools. LC is intended to put together the positive characteristics found in each method and creates a self-adaptive optimization approach. Each individual, as a candidate solution, decides based on its success if it would prefer to belong to a population of a GA, to a swarm of a PSO, or to a colony of ACO. This means that various heuristic techniques contribute to form a robust high performance optimization tool. The idea is that complex problems can be conveniently considered from the optimization viewpoint. As can be seen,

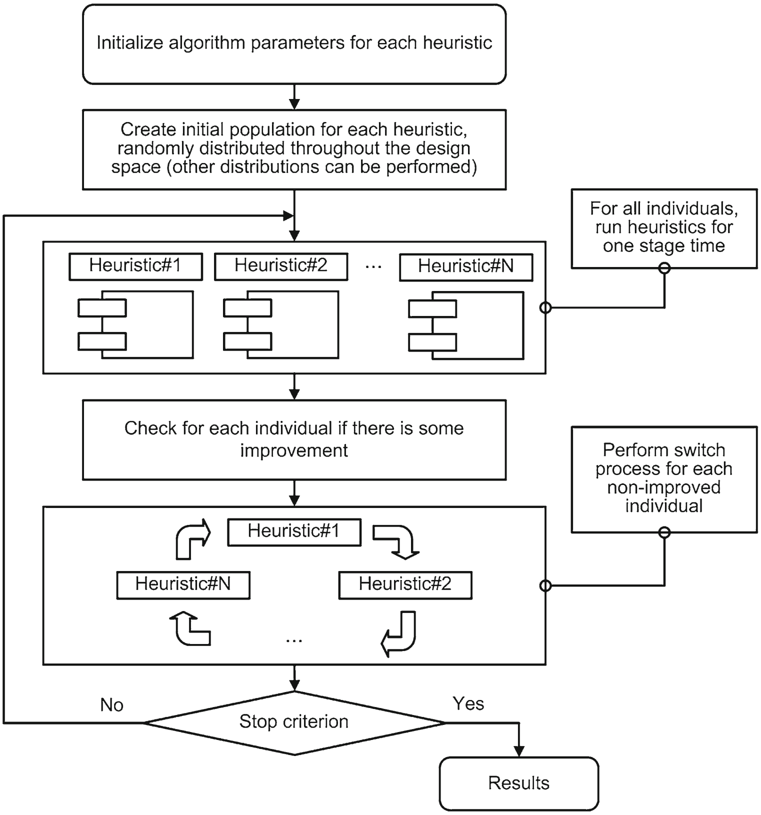

Initialize algorithm parameters for each heuristic
Create initial population for each heuristic, randomly distributed throughout the design space (other distributions can be performed)
For all individuals, run heuristics for one stage time
Heuristic#1
Heuristic#2
...
Heuristic#N
Check for each individual if there is some improvement
Perform switch process for each non-improved individual
Heuristic#1
Heuristic#N
Heuristic#2
...
No
Stop criterion
Yes
Results

Fig. 7 LC basic algorithm

the less successful individuals must change their status in order to improve their fitness. This means that the optimization approach does not follow a rigid scheme, in which various techniques are used sequentially in a cascade-type structure. In other words, it is the mechanism of self-adaptation to the optimization problem that rules the procedure. A LC outline follows in Fig. 7.

Since the algorithm is composed by various heuristics, it is necessary to set the parameters of every heuristic used in the LC. Nevertheless, there is a parameter inherent to the LC, namely the number of iterations that represent a stage of the LC, known as stage interval. At the end of each stage interval, the less successful individuals must change their stage in order to improve their fitness. To close the definition, LC stages must be presented. In the present work, two heuristics are used as stages, namely the GA and the PSO. Other versions of the LC can be proposed by considering other heuristics and a mix of them, as shown in [35].

To learn more about LifeCycle see [35] and [36].

In terms of implementation, all the nature-inspired approaches tested in the work have a common feature, namely, they are highly flexible and robust general frameworks. The SIMPLE Optimization Toolbox [37], a toolbox developed at Federal University of Uberlandia as an add-on to MATLAB, provides implementation for the algorithms used to handle the optimization problem.

Table 2 Pressure bulkhead description

Web material	Al 2024
Number of horizontal reinforcement beams	3
Number of vertical reinforcement beams	5
Reinforcement beams material	Al 7050
Pressure differential	Equivalent to 30,000 ft

Table 3 Design variables values normalized by the bulkhead diameter. Normalization makes the design variables non-dimensional. This is done to preserve interests in the technological content of this work

Design variable	Lower bound	Baseline	Upper bound
x_1 : width of reinforcement beams	0.01	0.02	0.03
x_2 : height of reinforcement beams	0.040	0.05	0.06
x_3 : thickness of reinforcement beams	0.0008	0.0012	0.0016
x_4 : web thickness	0.0008	0.0012	0.0016

Table 4 Baseline design response values selected to participate of the optimization problem statement

Response	Baseline value	Target value
$f_1(\mathbf{x})$: ratio of maximum tension stress at the reinforcement beams with respect to allowable	1.01	0.8
$f_2(\mathbf{x})$: ratio of minimum compression stress at the reinforcement beams with respect to allowable	0.85	0.68
$f_3(\mathbf{x})$: web maximum misplacement, normalized with respect to the bulkhead diameter	0.116	0.928
$f_4(\mathbf{x})$: maximum limit tension stress with respect to yield limit	0.19	0.15
$f_5(\mathbf{x})$: pessure bulkhead Mass	22.5635	18.0508

5 Case study: Numerical optimization of a pressure bulkhead

The optimization framework based on heuristic methods and on variable fidelity analyses, outlined in Sects. 4 and 3, is tested through a case study focused on a flat reinforced pressure bulkhead, as described in Sects. 1 and 2.

Tables 2, 3 and 4 contain useful design data, such as the general information about the pressure bulkhead; design variable descriptions and their respective design space specifications; and finally, response descriptions and respective values for the baseline design and target design as well. All this information is important for the good understanding and definition of the optimization problem.

In order to keep the computational cost low (in terms of low-fidelity simulations), instead of a multi-objective approach for generating the Pareto front, the responses are combined into a functional whose minimization implies that all responses tend to a target value and depart from a non-desirable one. This simplifies the solution of the problem and can be considered acceptable given the general scope of this work. The formulation is shown in Eq. (12). The target and avoidable values are not necessarily design goals, but play an important role at

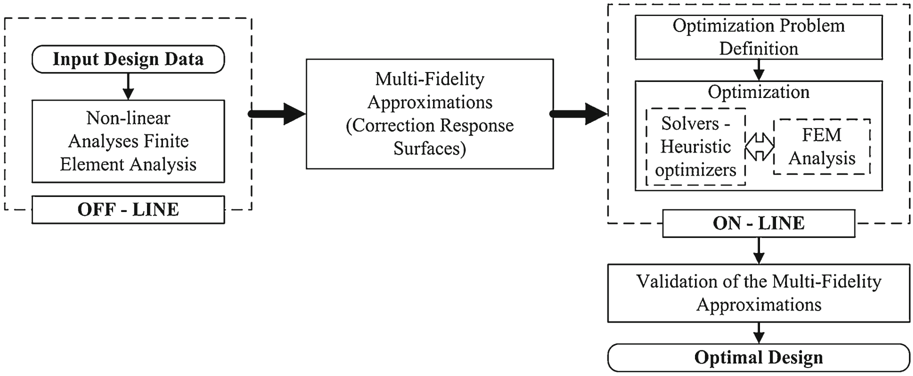

Input Design Data
Non-linear Analyses Finite Element Analysis
OFF - LINE
Multi-Fidelity Approximations (Correction Response Surfaces)
Optimization Problem Definition
Optimization
Solvers - Heuristic optimizers
FEM Analysis
ON - LINE
Validation of the Multi-Fidelity Approximations
Optimal Design

Fig. 8 Variable-fidelity optimization framework

the optimization problem by defining the tendency of the desired optima with respect to the baseline design. This scheme is known as *Compromise Programming*, better described by Vanderplaats in [12]. In the context of the present work, it is a useful approach since the nature-inspired optimization techniques are defined for unconstrained and single-objective problems only. The final problem is expressed as the minimization of the following functional:

$$J(\mathbf{x}) = \sqrt{\sum_{i=1}^{n_{obj}} \left(w_i \frac{\left(f_i(\mathbf{x}) - f_i^*(\mathbf{x})\right)}{f_i^{worst}(\mathbf{x}) - f_i^*(\mathbf{x})} \right)^2} \tag{12}$$

where:

- $J(\mathbf{x})$ is a compromise objective function.
- $f_i(\mathbf{x})$ is the i-th response of interest, in a total of n_{obj}.
- $f_i^*(\mathbf{x})$ is the target value of the i-th response.
- $f_i^{worst}(\mathbf{x})$ is the worst value accepted for the i-th response.
- w_i is the weighting factor applied for the i-th response of interest.

During the optimization task, each of the responses present in Eq. (12) is calculated through a linear static finite element analysis (low-fidelity) and corrected by the PRS that correlate the result with the one obtained by means of non-linear finite element analysis (high-fidelity).

The sets of four design variables and five responses that describe the pressure bulkhead modeling are listed in Tables 3 and 4, respectively. The target values were chosen to be 80% of those at the baseline design (since all of them are quantities to be minimized). The bulkhead mass is set to be 2.5 times more important than the other responses (i.e. the weight coefficients are $w_i = 1$, for $i = 1 - 4$ and $w_5 = 2.5$). The baseline responses are chosen as the worst accepted values, thus forcing the optimizer to move away from them in a clear attempt to improve the design.

For the sake of simplicity, the variable fidelity is implemented with a second order PRS model for the difference between the high and low fidelity analyses of responses $f_1(\mathbf{x})$ to $f_4(\mathbf{x})$. Since $f_5(\mathbf{x})$ is the pressure bulkhead mass, it does not require any correction (its fidelity does not depend on whether the calculations are linear or non-linear; eventually, small differences may appear due to either rounding or truncation).

The samples for the regression are generated through a Central Composite Design (CCD), which demands 25 finite element simulations (both linear and non-linear). Following the idea

Table 5 Comparison of the adjusted correlation coefficients, R_a^2, of the PRS models for the responses, $\hat{f}(\mathbf{x})$, and for the correction surrogates, $\hat{\delta}(\mathbf{x})$ (the closer R_a^2 is to 1, the better). It can be observed the success in using variable-fidelity over the direct meta-modeling of the responses

Surrogate of the response	$\hat{f}_1(\mathbf{x})$ 0.65	$\hat{f}_2(\mathbf{x})$ 0.34	$\hat{f}_3(\mathbf{x})$ 0.93	$\hat{f}_4(\mathbf{x})$ 0.87
Correction surrogate	$\hat{\delta}_1(\mathbf{x})$ 0.99	$\hat{\delta}_2(\mathbf{x})$ 0.96	$\hat{\delta}_3(\mathbf{x})$ 0.982	$\hat{\delta}_4(\mathbf{x})$ 0.952

presented in Fig. 8, most of the off-line computational effort is due to these runs, apart from isolated validations. According to this procedure, the obtained PRS are shown in Eqs. (13)–(16):

$$\begin{aligned}\hat{\delta}_1(x) = {} & -3.07513 + 1.57573x_1 + 0.75861x_2 + 0.97382x_3 + 0.23060x_4 \\ & -0.41144x_1^2 - 0.11233x_2^2 - 0.23423x_3^2 \\ & -0.18688x_1x_2 - 0.32223x_1x_3 - 0.23890x_1x_4 - 0.14728x_2x_3 \end{aligned} \tag{13}$$

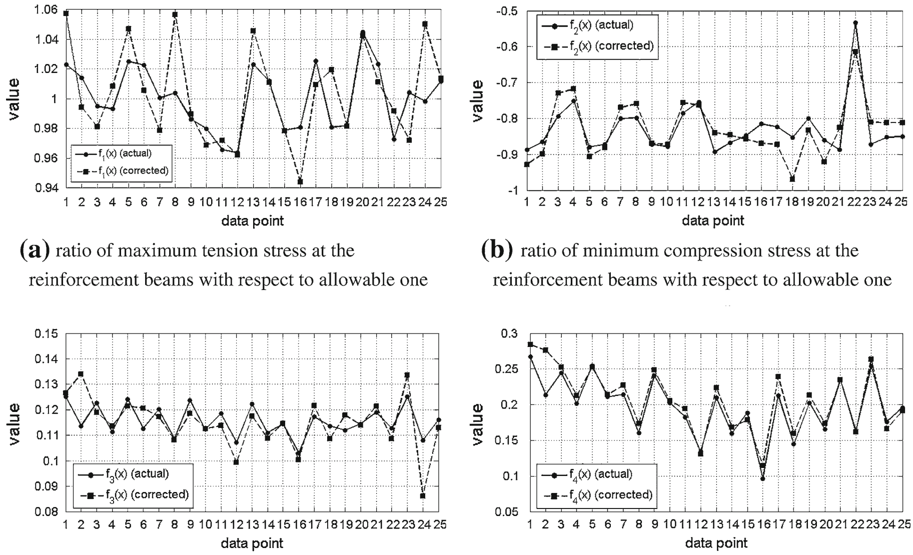
$$\begin{aligned}\hat{\delta}_2(x) = {} & -5.09962 + 1.68555x_1 + 1.33653x_2 + 0.96027x_3 \\ & -0.63840x_3^2 - 0.29912x_1x_2 \end{aligned} \tag{14}$$

$$\hat{\delta}_3(x) = -7.77607 + 0.41315x_2 + 5.39596x_4 - 2.36998x_4^2 \tag{15}$$

$$\begin{aligned}\hat{\delta}_4(x) = {} & -18.5759 - 2.6548x_1 - 1.2855x_2 - 2.4576x_3 + 6.3808x_4 \\ & -1.2053x_4^2 - 1.0776x_1x_3 \end{aligned} \tag{16}$$

Table 5 gives the comparison of the adjusted correlation coefficients, R_a^2, of the PRS models for the responses, $\hat{f}(\mathbf{x})$, and for the correction surrogates, $\hat{\delta}(\mathbf{x}).R_a^2$ primarily measures the statistical accuracy of the PRS surrogates (the closer R_a^2 is to 1, the better). As expected from the theoretical analysis, it can be observed that (i) due to the non-linearities, it is worth using the variable-fidelity framework over the PRS models of the responses; and (ii) the correction PRS models adjust very well to the data. As a consequence, the estimated response values (linear analysis + error correction) and the actual non-linear results have very close variation in the case of all four responses of interest, as shown in Fig. 9. As a final verification, the data corresponding to the 25 runs of the CCD was used to calculate two sets of correlation coefficients, as shown in Table 6. All sets of data move in the same direction (positive correlations) and, in general, the correlation is greater when the correction surrogates are used with the linear results. Given that the correction surrogates have demonstrated good predicting capabilities, it is now safe to perform the intended optimization procedures.

One of the aims of this work is to show the performance of the heuristic algorithms when combined with variable fidelity in a real world problem. Given the random nature of the algorithms used, results may vary from one run to another (or at least present slight differences). This way, in the first part of the study, each algorithm is tested 10 times (each time with different initial populations). The success of the algorithms in performing the initial global search is observed if the results of these repetitions are close to each other. Table 7 shows the setup used for each of the heuristic optimization algorithms. Secondly, a set of initial designs given by the heuristic methods fed the NMSDS in order to finish the local search. Finally, one of the final designs is used to validate the results of the optimization performed with variable fidelity analysis. This is done by checking the differences between the uncorrected and corrected linear (low-fidelity) simulations with respect to the non-linear one (high-fidelity).



(a) ratio of maximum tension stress at the reinforcement beams with respect to allowable one

(b) ratio of minimum compression stress at the reinforcement beams with respect to allowable one

(c) web maximum misplacement, normalized with respect to the bulkhead diameter

(d) maximum limit tension stress with respect to yield limit

Fig. 9 Variation of actual and corrected values of responses f_1 (**x**) to f_4 (**x**). Actual responses are those obtained through high-fidelity analyses and corrected response refers to those obtained through low-fidelity analyses and correction PRS surrogates

Table 6 Correlation coefficients of non-corrected and corrected linear results with respect to actual non-linear results. These values show the improvement capabilities of the correction surrogates adopted on the variable fidelity framework

Response	Uncorrected linear to non-linear correlation coefficients (%)	Corrected linear to non-coefficients linear correlation coefficients (%)
$f_1(\mathbf{x})$	51,32	66,64
$f_2(\mathbf{x})$	28,94	79,28
$f_3(\mathbf{x})$	70,09	87,86
$f_4(\mathbf{x})$	61,26	94,62

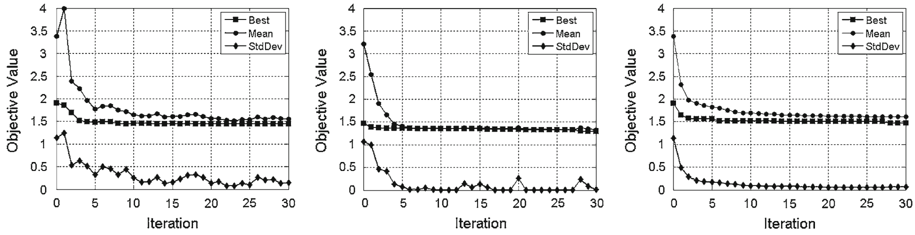
6 Results and interpretation

In the first part of the study, the performance and convergence of the heuristic algorithms is tested Fig. 10 and Figure 11 shows an arbitrary run executed with each of them. They help as an initial comparison basis to analyze the behavior of the used techniques. Figure 10 shows how the best, the mean and the standard deviation of the objective function values for the population of ACO, GA and PSO evolve during the optimization procedure. Figure 11 illustrates the evolution of the LC along the iterations. Figure 11-(a) shows which heuristic is conducting the optimization process at a given iteration. Fig. 11-(b) shows the transitions due to its self-adaptation skills.

Given the random nature of the algorithms used, the profiles of Figs. 10 and 11 may vary from one run to another (final results may also vary slightly). This way, for the sake of the statistical analysis, each run was repeated 10 times and the best, worst, average, standard deviation and coefficient of variation of the results from each of the 10 repetitions were

Table 7 Setup for heuristic optimization methods

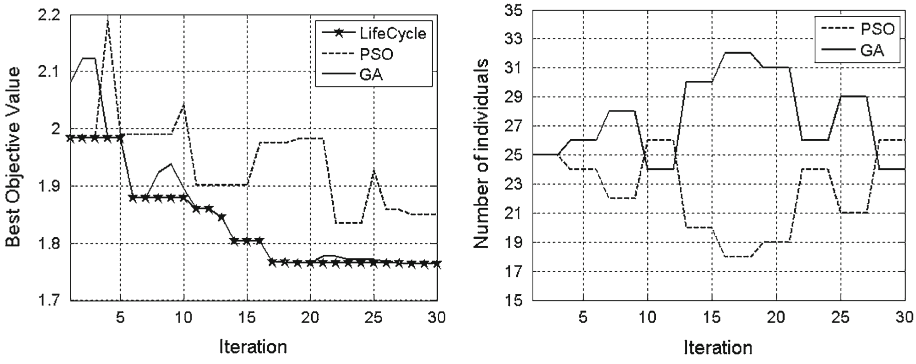
General	Population size	50
	Iterations	30
	Stop criterion	Maximum. number of iterations
ACO	Dissolving rate	1.25
GA	Number of individuals for elitism	2
	Selection function	simpleGASelectionRoulette
	Crossover function	simpleGACrossoverHeuristic
	Crossover fraction	0.8
	Mutation function	simpleGAMutationUniform
	Migration direction	‘forward’
	Migration interval	20
	Migration fraction	0.2
PSO	Inertia (w)	1.4
	Self trust (c_1)	1.5
	Swarm trust (c_2)	2.5
	DT	1
	Vmax	0.2
	Mass extinction factor	0.975
	Coefficient of Variation	1
	Swarm Subset	0.2



(a) ACO **(b)** GA **(c)** PSO

Fig. 10 Sample of one run of the optimization for each basic algorithm. In all cases, it can be observed good convergence both in terms of results (seen by the best value of the objective function) and in terms of the dispersion of the population (seen by the mean and standard deviation of the objective function)

recorded. The average indicates the central tendency which the heuristic results tend to. The standard deviation indicates the stability of the optimal solutions, measuring the degree of convergence. In order to express this convergence degree regardless of the magnitude of each of the responses, the coefficient of variation is calculated through the division of the standard variation by the mean value of the optima. The average value for $J(\mathbf{x})$ is the average of values of the functional over the 10 runs; it does not correspond to the value obtained when running the analysis with the average design. Table 8 reports this information for each of the heuristic methods. The best and average results are shown in bold face. This table also shows



(a) best objective trace **(b)** number of individuals trace.

Fig. 11 Sample of one run of the optimization for the LC algorithm. Figure 11-(a) illustrates how GA and PSO alternate in providing to LC the optimal solution. During the first 4 iterations, this is done by PSO, then the solution is few times updated by GA. Figure 11-(b) shows how this process is reflects the split of the population between GA and PSO. Since most of the time GA performs better than PSO, it can be observed that the population trends to perform GA rather than PSO

that all algorithms converged fairly well. While GA achieved the best result, LC presents better dispersion (smaller differences among the best, average and worst designs). This is an evidence of the LC robustness, somehow expected, since LC is intended to take advantage of multiple heuristics simultaneously and to reduce the bias of a poorly performed algorithm. Altogether, Table 8 also indicates the success of all algorithms in pointing to a close region of the design space. Table 9 shows what happens when running the objective function with the average result found by each algorithm. The response values are all obtained by linear analysis (low fidelity) and further on properly corrected. Indeed, despite the apparent differences among the average values (which suggests some advantage for GA), the four averages are statistically equivalent at a 95% significance level, since all of them belong to the corresponding confidence interval that goes from 1.667 to 1.828.

In the second part of the study, the efficiency of heuristics in providing an initial guess for a classical algorithm is tested. Table 10 shows the results when running another step of optimization using NMSDS with initial designs as given by Table 9. All final designs present even closer values for the design variables, individual responses and respective functional, $J(\mathbf{x})$. It means that indeed the initial designs were in a close region of the design space.

As seen in Fig. 8, the final step of the variable-fidelity approach is the validation of the optimization outcomes. This is done by running the high-fidelity simulations using the optimal design. Table 11 shows the results of an arbitrary run of the LC algorithm as performed lonely (simulation #10) and the results of LC + NMSDS (as in Table 10). In general, the non-linear and linear corrected simulations present just small differences for responses $f_1(\mathbf{x})$ to $f_4(\mathbf{x})$ ($f_5(\mathbf{x})$ is the pressure bulkhead mass and it does not depend on the level of fidelity). The similar results (bold face in Table 11) show once again that the candidate solutions found by the heuristic methods lay in a close region of the design space, where the optimal may be. From the design viewpoint, the changes introduced by the optimizers seem to be sound choices. Material has been removed from the web to the reinforcement beams, in a clear effort to minimize the mass, which was given highest priority by means of the 2.5 weighting factor in Eq. (12). Most important, the optimizers were capable of finding a design that overall improved the involved quantities, despite their inherent conflicting nature. The small

Table 8 Optimization results for all heuristic algorithms (performed lonely). All algorithms presented good convergence. While GA presents the best "best solution," LC presents the best dispersion measures, and as a consequence, LC was shown to be the most robust. This is due to the LC ability in taking advantage of multiple heuristics simultaneously

Algorithm	Statistics	Design variable				Objective function $J(\mathbf{x})$
		$x_1 \times 10^{-3}$	$x_2 \times 10^{-3}$	$x_3 \times 10^{-3}$	$x_4 \times 10^{-3}$	
ACO	Best	25	41.94	1.5	1.08	**1.44**
	Average	24.96	40.19	1.5	1.03	**1.73**
	Worst	24.96	39.4	1. 5	1.02	1.76
	StdDev	0.035	0.73	0.0001	0.02	0.10
	COV	0.001	0.02	0.0001	0.02	0.06
GA	Best	29.08	42.87	1.65	0.98	**1.3**
	Average	26.70	38.37	1.63	1.00	**1.67**
	Worst	22.5	47.4	1.47	1.06	1.9
	StdDev	2.55	5.65	0.09	0.04	0.16
	COV	0.001	0.02	0.0001	0.02	0.06
PSO	Best	24.89	41.35	1.49	1.07	**1.47**
	Average	24.61	40.72	1.49	1.04	**1.76**
	Worst	24.3	38.87	1.48	1.06	1.81
	StdDev	0.42	1.59	0.007	0.03	0.10
	COV	0.02	0.04	0.005	0.03	0.06
LC	Best	25.00	40.28	1.50	1.03	**1.764**
	Average	24.62	40.47	1.5	1.04	**1.774**
	Worst	24.38	41.05	1.5	1.05	1.785
	StdDev	0.4	0.64	0.002	0.01	0.009
	COV	0.02	0.02	0.001	0.01	0.005

Table 9 Comparison of the average design and the corresponding responses for each algorithm. It can be observed that the performance of the different algorithm is very close, with a small advantage for GA

Design		Algorithm			
		ACO	GA	PSO	LC
Design variables	$x_1 \times 10^{-3}$	24.96	26.703	24.614	24.624
	$x_2 \times 10^{-3}$	40.191	38.367	40.717	40.473
	$x_3 \times 10^{-3}$	1.5	1.625	1.49	1.499
	$x_4 \times 10^{-3}$	1.03	1.003	1.038	1.036
Responses (linear corrected)	$f_1(\mathbf{x})$	0.96	0.92	0.96	0.96
	$f_2(\mathbf{x})$	−0.72	−0.66	−0.72	−0.72
	$f_3(\mathbf{x}) \times 10^{-3}$	109.42	109.44	109.23	109.28
	$f_4(\mathbf{x})$	0.18	0.18	0.18	0.18
	$f_5(\mathbf{x})$	21.56	21.82	21.60	21.6
Functional	$J(\mathbf{x})$	1.77	**1.67**	1.78	1.77

Table 10 Results obtained by NMSDS when using initial designs given by different heuristics. The similarities of the final results confirm that the initial results were in a close region of the design space

Design		Algorithm			
		ACO + NMSDS	GA + NMSDS	PSO + NMSDS	LC + NMSDS
Design variables	$x_1 \times 10^{-3}$	29.93	29.96	29.93	29.94
	$x_2 \times 10^{-3}$	33.06	33.04	33.04	33.04
	$x_3 \times 10^{-3}$	1.73	1.73	1.73	1.73
	$x_4 \times 10^{-3}$	0.96	0.96	0.96	0.96
Responses (linear corrected)	$f_1(\mathbf{x})$	0.89	0.89	0.89	0.89
	$f_2(\mathbf{x})$	−0.67	−0.67	−0.67	−0.67
	$f_3(\mathbf{x}) \times 10^{-3}$	112.34	112.35	112.34	112.35
	$f_4(\mathbf{x})$	0.17	0.17	0.17	0.17
	$f_5(\mathbf{x})$	21.79	21.79	21.79	21.79
Functional	$J(\mathbf{x})$	1.64	1.64	1.64	1.64

Table 11 Validation of the optimal designs given by LC and LC + NMSDS. The non-linear responses reveal that the solutions are within a close region of the design space. In general, an improved in the quantities is observed (the small increase in the displacement is secondary in view of the improvements observed in the other responses)

Design		Response				
		$f_1(\mathbf{x})$	$f_2(\mathbf{x})$	$f_3(\mathbf{x}) \times 10^{-3}$	$f_4(\mathbf{x})$	$f_5(\mathbf{x})$
Baseline (Non-linear)		1.01	−0.85	116.0	0.2	22.56
LC (simulation #10,	Linear uncorrected	3.42	−5.35	1.71	0.09	20.95
$J(\mathbf{x}) = 1.76$)	Linear corrected	0.92	−0.66	109.44	0.17	20.95
	Non-linear	**0.91**	**−0.65**	**117.5**	**0.17**	**20.95**
	Optimization effect (%)	−9.90	−23.53	+1.29	−15.00	−7.14
LC + NMSDS	Linear uncorrected	0.06	−0.0709	1.85	0.0016	21.79
($J(\mathbf{x}) = 1.64$)	Linear corrected	0.89	−0.67	112.35	0.17	21.79
	Non-linear	**0.88**	**−0.62**	**118.20**	**0.18**	**21.79**
	Optimization effect (%)	−12.87	−27.06	1.90	−10.00	−3.41

increase in the displacement is secondary in view of the improvements observed in the other responses. Following this idea, if different designs were validated, the choice of a single one would depend on the analysis of the individual responses. Therefore, the designer can choose which solution is the most convenient for the current application.

7 Summary and conclusion

In this paper, the use of intensive computing heuristic techniques for the optimal design of aircraft structural components has been explored. Then (i) the coupling of non-linear high-fidelity and linear low-fidelity analyses through the variable fidelity approach; and (ii) the performance of different heuristic optimization methods have been investigated.

The study allows the following conclusions:

- Variable fidelity approach enabled performing intensive computing heuristic techniques through the rational use of expensive non-linear analyses and on-line inexpensive linear analyses.
- The four applied nature-inspired methods converged in the sense of pointing out the same region of the design space as a candidate to contain the most suitable design.

Finally, it should be pointed out that this contribution successfully demonstrated that heuristic techniques can be used for an initial exploration of the design space with the purpose of identifying the improvement trends and ruling out the inadequate design choices. In the sequence, a point-based classical optimization algorithm can refine the design. However, the final design is only obtained when the validation step of the variable-fidelity approach is performed. Then, the best design can be chosen by direct comparison of the high-fidelity responses.

Acknowledgments Dr. Viana is thankful to CNPq Brazilian Research Agency for the partial support of this work during his PhD. Dr. Steffen acknowledges CNPq for the partial financing of this research work (Proc. No. 470346/2006-0). Dr. Leal and Dr. Butkewitsch, they are both grateful to EMBRAER.

References

1. Bruhn, E.F.: Analysis and Design of Flight Vehicle Structures. Jacobs Pub (1973)
2. Niu, M.C., Niu, M.: Airframe Structural Design—Practical Design Information and Data on Aircraft Structures, 2nd edn, pp. 398. Adaso Adastra Engineering Center, USA (1999)
3. Venter, G., Sobieszczanski-Sobieski, J.: Particle Swarm Optimization. In: Proceedings of the 43rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Denver, CO, USA, AIAA-2002-1235, 22–25 Apr (2002)
4. Venter, G., Sobieszczanski-Sobieski, J.: Multidisciplinary optimization of a transport aircraft wing using particle swarm optimization. In: Proceedings of the 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Atlanta, GA (2002)
5. Viana, F.A.C., Kotinda, G.I., Rade, D.A., Steffen, V., Jr.: Tuning Dynamic Vibration Absorbers by Using Ant Colony Optimization. Comput. Struc (2007). doi:10.1016/j.compstruc.2007.05.009
6. Giunta, A.A., Balabanov, V., Haim, D., Grossman, B., Mason, W.H., Watson, L. T., Haftka, R. T.: Wing design for a high-speed civil transport using a design of experiments methodology. In: Proceedings of the 6th AIAA/NASA/USAF Multidisciplinary Analysis and Optimization Symposium, Bellevue, pp. 96–4001. WA, USA, AIAA, 4–6 Sept (1996)
7. Alexandrov, N.M., Lewis, R.M., Gumbert, C.R., Green, L.L., Newman, P.A.: Optimization with Variable-fidelity Models applied to Wing Design. Institute for Computer Applications in Science and Engineering (ICASE), NASA Langley Research Center, Hampton, USA, ICASE Technical Report N. 99–49 (1999)
8. Marduel, X., Tribes, C., Trépanier, J.Y.: Variable-Fidelity Optimization—Efficiency and Robustness. Optim. Eng. **7**, 479–500 (2006). doi:10.1007/s11081-006-0351-3
9. Vitali, R., Haftka, R.T., Sankar, B.V.: Multi-fidelity design of stiffened composite panel with a crack. Struct. Optim. **23**(5), 347–356 (2002)
10. Przemieniecki, J.S.: Theory of Matrix Structural, Analysis, pp. 468. Dover Publications Inc., New York (1985)
11. Den Hartog, J.P.: Advanced Strength of Materials, pp. 70–99. Dover Publications Inc., New York (1952)
12. Vanderplaats, G.N.: Numerical Optimization Techniques for Engineering Design, 4th edn. Vanderplaats Research and Development, Inc., Colorado Springs (2005)
13. Jin, R., Chen, W., Simpson, T.W.: Comparative Studies of Metamodelling Techniques under Multiple Modeling Criteria. Struct. Multidisciplinary Optim. **23**, 1–13 (2001). doi:10.1007/s00158-001-0160-4
14. Simpson, T.W., Poplinski, J.D., Koch, P.N., Allen, J.K.: Metamodels for Computer-based Engineering Design: Survey and Recommendations. Eng. Comput. **17**(2), 129–150 (2001). doi:10.1007/PL00007198

15. Queipo, N.V., Haftka, R.T., Shyy, W., Goel, T., Vaidyanathan, R., Tucker, P.K.: Surrogate-Based Analysis and Optimization. Prog. Aerosp. Sci. **41**(1), 1–28 (2005). doi:10.1016/j.paerosci.2005.02.001
16. Montgomery, D.C.: Design and Analysis of Experiments, 6th edn. Wiley, New York (2004)
17. Box, G.E.P., Hunter, W.G., Hunter, J.S.: Statistics for experimenters. Wiley, New York (1978)
18. Myers, R.H., Montgomery, D.C.: Response Surface Methodology—Process and Product Optimization using Designed Experiments, 2nd edn. Wiley-Interscience, USA (2002)
19. Smith, M.: Neural Networks for Statistical Modeling. Von Nostrand Reinhold, New York (1993)
20. Cheng, B., Titterington, D.M.: Neural Networks—a Review from a Statistical Perspective. Stat. Sci. **9**, 2–54 (1994). doi:10.1214/ss/1177010638
21. Sacks, J., Welch, W.J., Mitchell, T.J., Wynn, H.P.: Design and Analysis of Computer Experiments. Stat. Sci. **4**(4), 409–435 (1989). doi:10.1214/ss/1177012413
22. Lophaven, S.N., Nielsen, H.B., Søndergaard, J.: DACE—A MATLAB Kriging Toolbox. Technical Report IMM-TR-2002-12, Informatics and Mathematical Modelling, Technical University of Denmark (2002)
23. Smola, A.J., Scholkopf, B.: A tutorial on support vector regression. Stat. Comput. **14**, 199–222 (2004). doi:10.1023/B:STCO.0000035301.49549.88
24. Clarke, S.M., Griebsch, J.H., Simpson, T.W.: Analysis of Support Vector Regression for Approximation of Complex Engineering Analyses. J. Mech. Des. **127**, 1077–1087 (2005). doi:10.1115/1.1897403
25. Meckesheimer, M., Booker, A.J., Barton, R.R., Simpson, T.W.: Computationally Inexpensive Metamodel Assessment Strategies. AIAA J. **40**(10), 2053–2060 (2002). doi:10.2514/2.1538
26. Balabanov, V., Venter, G.: Multi-fidelity optimization with high-fidelity analysis and low-fidelity gradients. In Proceedings of the 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Albany, USA (2004)
27. Lagarias, J.C., Reeds, J.A., Wright, M.H., Wright, P.E.: Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions. SIAM J. Optim. **9**(1), 112–147 (1998). doi:10.1137/S1052623496303470
28. Dorigo, M.: Optimization, Learning and Natural Algorithms. PhD thesis, Politecnico di Milano, Italy (1992)
29. Socha, K.: ACO for continuous and mixed-variable optimization. In: Proceedings in ANTS 2004 - Fourth International Workshop on Ant Colony Optimization and Swarm Intelligence, Brussels, Belgium (2004)
30. Di Caro, G., Dorigo, M.: AntNet—Distributed Strigmergic Control for Communication Networks. J. Artif. Intell. Res **9**, 317–365 (1998)
31. Gen M., Cheng R. (1999) Genetic Algorithms and Engineering Optimization. Wiley-Interscience, New York, USA
32. Michalewicz, Z., Fogel, D.B.: How to Solve it—Modern Heuristics, 1st edn. Springer-Verlag, New York, USA (2000)
33. Haupt, R.L., Haupt, S.E.: Pratical Genetic Algorithms, 2nd edn. Wiley-Interscience Publication, New York, USA (2004)
34. Kennedy, J., Eberhart, R.C.: Particle Swarm Optimization. In: Proceedings of the 1995 IEEE International Conference on Neural Networks, pp. 1942–1948. Perth, Australia (1995)
35. Krink, T., Løvberg, M.: The lifecycle model—combining particle swarm optimisation, genetic algorithms and hill-climbers. In Proceedings of the 7th International Conference on Parallel Problem Solving from Nature, 621–630 (2002)
36. Rojas, J.E., Viana, F.A.C., Rade, D.A., Steffen, V. Jr.: Identification of External Forces in Mechanical Systems by using Lifecycle Model and Stress-Stiffening Effect. Mech Syst Signal Process **21**(7), 2900–2917 (1998)
37. Viana, F.A.C., Steffen, V., Jr.: SIMPLE Optimization ToolBox—Users Guide, 2th edn. http://fchegury.110mb.com/, [Cited 5 Nov 2006] (2006)