A real-coded GA algorithm for optimizing the damping response of composite laminates

M.V. Pathan[a], S. Patsias[b] and V.L. Tagarielli[a,*]

[a] *Department of Aeronautics, Imperial College London, SW7 2AZ London, UK*

[b] *Rolls-Royce plc, PO Box 31, DE24 8BJ Derby, UK*

Abstract

We develop a real-coded constrained genetic algorithm (GA) and assess its performance in selected classical optimisation problems. The proposed GA uses a roulette selection method, BLX-α crossover operation, non-uniform mutation along with single elitist selection at every generation. The GA is then applied, in conjunction with the finite element (FE) method, to optimise the damping response of a laminate comprising unidirectional composite laminae and viscoelastic damping layers. Modal loss factors are maximised against the constraints of given structural stiffness and mass.

Keywords: Real-coded genetic algorithm; Composite laminates; Optimization; Damping

Submitted to *Computers & Structures*, February 2017

1. Introduction

* Corresponding author. E-mail v.tagarielli@imperial.ac.uk

Fibre reinforced polymers (FRPs) are being increasingly used in automotive and aerospace sector due to their superior specific stiffness, strength and damping. These materials can be tailored by adjusting volume fractions of the constituents, layer thickness and stacking sequence. Recently, the issue of accurate numerical predictions of the stiffness and damping properties of the FRPs has been addressed at both ply [1–4] and laminate [5] level. In certain applications however, the inherent damping of composites plies is not sufficient and viscoelastic damping layers or inserts are used to increase structural damping; this is typically done placing a compliant damping layer between relatively stiffer composite plies, to induce shear deformation in the soft material, thereby dissipating energy. The poor stiffness of damping layers, with density comparable to that of composite plies, may degrade the specific structural stiffness of the laminate.

The effectiveness of damping layers also depends on their thickness and location in a given laminate. Several authors have studied the optimal location of viscoelastic layers for maximum damping in composite laminates, using analytical or numerical techniques [6–15]. The task can be formulated as a constrained optimization problem, where the objective is to maximize the damping capacity of a laminate, having certain constraints on mass, stiffness and load-carrying capacity [7,15]; single- and multi-objective algorithms have been published (e.g. [11,16]). The effectiveness of optimisation algorithms to maximise the damping of structures made from laminates also depends on the type of objective function; different objective functions have been considered in the literature: maximum modal loss factors or their sums [6,8], minimum deflection at resonant frequencies [12], minimum vibrational energy [13], among others. Several studies have also explored the use of discontinuous damping surface patches to increase the modal damping capacity of the laminate [8,17].

The design space of laminated composite structures has a high number of dimensions of both discrete and continuous nature. Classical non-linear programming techniques are unsuited for such non-convex search spaces given the fact they are local search methods that have a tendency to get stuck in the local extrema. Typically, such problems are better handled by techniques belonging to the class of evolutionary algorithms, most popular of which are genetic algorithms (GAs). GAs are inspired by the natural evolutionary principles of selection, crossover, mutation and evolution; for a comprehensive discussion of GAs the readers are referred to [18,19]. The idea is to select the best performing candidate solutions in a certain population and then combine their genomic information to possibly create children with better performance; GAs are stochastic search-based approaches which makes them efficient over

other methods. Several authors have applied GAs to optimize damping in FRP laminates. Zheng et al.[13,20] optimized thickness and location of a viscous patch for the case of a simply-supported beam; Xie et al.[12] minimized structural displacements at resonant frequencies by tailoring the thicknesses of the constituent plies. Montemurro et al. [7] performed damping optimization of a hybrid laminate consisting of transversely isotropic FRP plies and isotropic viscoelastic damping plies. The design variables considered in this study were ply number, laminate sequence and ply thickness. More recently, Xu et al.[8] performed multi-objective design optimization of damping in a laminate using FE analysis with a modified NSGA-II algorithm [21]. Most of the works dealing with optimization of FRP laminates are based on binary-coded GAs, which are not efficient in dealing with real-valued design variables [22]. Moreover, the inherent material damping due to the fibre composites has been largely ignored in all studies.

In this paper we develop a real-coded GA (i.e. a GA using real number representation for the candidate solutions) and test its effectiveness in dealing with non-convex benchmark optimization problems, comparing to selected state-of-the-art evolutionary algorithms. The algorithm is then employed to maximise the damping of a cantilever beam, with constraints on the structural mass and stiffness; the design variables are ply thickness and stacking sequence. This is done in conjunction with FE simulations, in which the response of a cantilever beam is simulated in detail, including all non-linearities as well as the anisotropic, viscoelastic response of all constituent materials.

The outline of the paper is as follows: in Section 2 we define the optimization problem and give details of the proposed real-coded GA. In Section 3 we test the algorithm in selected benchmark problems and in Section 4 we apply the proposed GA to the case of layered composites with damping layers.

2. Optimisation Algorithm

2.1 Constrained optimization problem

A single-objective constrained optimization problem can be formulated as the minimization problem

$$\min f(\vec{x}), \text{ subject to}$$
$$\begin{cases} g_j(\vec{x}) \le 0, & j = 1,...,l \\ h_j(\vec{x}) = 0, & j = l+1,...,m \\ p_j \le x_j \le q_j, & j = 1,...,n \end{cases} \tag{1}$$

where $\vec{x} = [x_1, x_2, ..., x_n]^T$ is the *n*-dimensional solution vector; the function is subjected to *l* inequality constraints (g_j) and $m - l$ equality constraints, represented by h_j. In practice the equality constraints are difficult to handle and are usually converted to inequality constraints by allowing a small tolerance ε, i.e.

$$\left|h_j(\vec{x})\right| - \varepsilon \le 0 \ . \tag{2}$$

Penalty based methods [18] are often used in optimization algorithms to handle constraints, which can be linear or non-linear. This approach transforms a constrained optimisation problem into an unconstrained one, by suitable modification of the original objective function $f(\vec{x})$. The 'penalised' objective function $f_p(\vec{x})$ is constructed as

$$f_p(\vec{x}) = \begin{cases} f(\vec{x}) & \text{if } g_i(\vec{x}) \le 0 \text{ and } h_j(\vec{x}) = 0 \\ f(\vec{x}) + \sum_{i=1}^{l} c_i G_i(\vec{x}) & \text{if } g_i(\vec{x}) > 0 \text{ and } h_j(\vec{x}) = 0 \\ f(\vec{x}) + \sum_{j=l+1}^{m} d_j H_j(\vec{x}) & \text{if } g_i(\vec{x}) \le 0 \text{ and } h_j(\vec{x}) \ne 0 \\ f(\vec{x}) + \sum_{i=1}^{l} c_i G_i(\vec{x}) + \sum_{j=l+1}^{m} d_j H_j(\vec{x}) & \text{if } g_i(\vec{x}) > 0 \text{ and } h_j(\vec{x}) \ne 0 \end{cases} \tag{3}$$

$$\begin{aligned} G_i(\vec{x}) &= \max\{0, g_i(\vec{x})\}; & i &= 1,...,l \\ H_j(\vec{x}) &= \max\{0, h_j(\vec{x})\}; & j &= l+1,...,m \end{aligned} \tag{4}$$

The selection of the penalization coefficients c_i and d_j is difficult, as high penalty coefficients limit the accuracy in proximity of the constraints while low coefficients results in large number of iterations.

2.2 Genetic Algorithm

Encoding of the candidate solutions is essential for an efficient GA search process. Traditionally the candidate solutions (chromosomes) are coded using binary representations, due to simplicity of implementation. The binary representation has limitations when dealing with continuous search spaces, where the size of binary strings can grow in length, resulting in storage and manipulation problems [23]. Binary coding also suffers from the "Hamming Cliffs" problem [22]. The use of real-coding is more natural for real-valued design variables, i.e. continuous search spaces, as it substantially simplifies the algorithm, resulting in higher efficiency.

Genetic algorithms are unconstrained optimization techniques, and constraints are imposed by penalty-based methods [7,24] or via multi-objective optimization [11,16]. In this study we use a recently proposed penalty-based method, referred to as automatic dynamic penalization strategy [25].

2.3 Description of proposed GA

In this study we develop a real-coded GA with a 'roulette' selection method, BLX-0.5 crossover operator [26] and non-uniform mutation (NUM) operator proposed by Michalewicz [19]. The choice of GA operators are based on findings of Herrera et al. [22], where several types of real-coded GA operators were compared, concluding that the BLX-0.5 and NUM gave the best performance. We briefly describe these operators below.

2.3.1 BLX-α crossover operator

For two parent candidate solutions with *n* design variables, $\vec{x}_i = [x_{i,1}, x_{i,2}, ..., x_{i,n}]$ and $\vec{x}_j = [x_{j,1}, x_{j,2}, ..., x_{j,n}]$ selected from a population of size P_N at generation *t*, $X^t = [\vec{x}_1, \vec{x}_2, ..., \vec{x}_{P_N}]$, the BLX-$\alpha$ operator generates the *k*-th component of a new offspring $\vec{x}_z$ belonging to the next generation, i.e. to the population at time $t+1$, X^{t+1}. The *k*-th component of $\vec{x}_z$ is a uniform random scalar in the range $[\min(x_{i,k}, x_{j,k}) - \alpha I, \max(x_{i,k}, x_{j,k}) + \alpha I]$, where I defines the distance between parent candidates given by $I = \max(x_{i,k}, x_{j,k}) - \min(x_{i,k}, x_{j,k})$ and α is a user defined parameter.

The effectiveness of the BLX-α is in its ability to search in a space domain not necessarily bounded by that of the parents; in addition, the GA is self-adaptive, since the search space depends on the distance between the parents. The choice of the parameter α is crucial as quantitatively defines the search domain. In this study we use $\alpha = 0.5$, based on the findings in Herrera et al. [22].

For the case of child solutions violating the design variable bounds, the values of the design variables are forced to the value of the nearest bound, to ensure that the search stays within the desired space.

2.3.2 Non-Uniform Mutation Operator

The NUM operator, as the BLX-α operator, possesses self-adaptive capabilities; this algorithm reduces the range of the allowable mutations with increasing generation number *t*, allowing for larger mutations at small *t* and fine-tuning towards the end of the optimisation problem. This allows for efficient search throughout the allowable search space but it ensures that good solutions are not lost at later generations. The operator mutates the *k*-th component of a certain parent $\vec{x}_i$ as

$$x'_{i,k} = \begin{cases} x_{i,k} + (1 - r^{\left(1-\frac{t}{G_N}\right)^b})(q_{i,k} - x_{i,k}), & \text{if } \tau \le 0.5 \\ x_{i,k} - (1 - r^{\left(1-\frac{t}{G_N}\right)^b})(x_{i,k} - p_{i,k}), & \text{if } \tau > 0.5 \end{cases} \quad (5)$$

Here r and τ are random variables with values between 0 and 1, G_N is the maximum number of generations. $p_{i,k}$ and $q_{i,k}$ are the lower and upper bounds of the design variable $x_{i,k}$ as defined in Eq.(1). The parameter *b* determines the degree of non-uniformity, i.e. it controls the contraction in the allowable mutation range with increasing generation number, with a higher value of b causing a faster reduction of the mutation range. This parameter has been set to 5 for all cases in this study.

2.3.3 Handling of constraints

For the case when both feasible and infeasible solutions exist, we use the recently proposed automatic dynamic penalty (ADP) method [25] for handling constraints, due to its ability to automatically select and update the values of variables c_i and d_i featuring in Eq. (3). These are obtained, for a generation *t*, as

$$\begin{aligned} c_i(t) &= \frac{\left|\phi_{best}^{F} - \phi_{best}^{IF}\right|}{(G_i)_{best}^{IF}} \quad i = 1,\ldots,l \\ d_i(t) &= \frac{\left|\phi_{best}^{F} - \phi_{best}^{IF}\right|}{(H_i)_{best}^{IF}} \quad i = l+1,\ldots,m \end{aligned} \tag{6}$$

Here the superscripts *F* and *IF* represent feasible and infeasible solutions at a given generation, respectively. ϕ_{best}^{F} and ϕ_{best}^{IF} represent the minimum value of the objective function for the feasible (x_i^F) and infeasible (x_i^{IF}) solutions, respectively. $(G_i)_{best}^{IF}$ and $(H_i)_{best}^{IF}$ are magnitudes of the constraint violations for inequality and equality constraints, respectively, evaluated for the best-performing infeasible solution (x_i^{IF}) using Eq.(4). The method chooses the variables c_i and d_i such that the penalized objective function of the best-performing infeasible solution is equal to that of the best-performing feasible solution. This increases the likelihood of some good (but infeasible) solutions to be selected as parents for the next generation [25].

For the case where all candidate solutions are infeasible, we ignore the values of objective function and solely use the information regarding the amount of constraint violation to drive the search towards a feasible region. We first normalize and sum the amount of constraint violation according to

$$G_{norm}(\vec{x}_i) = -\sum_{j=1}^{l+m} \bar{G}_j(\vec{x}_i) \tag{7}$$

where

$$\bar{G}_j(\vec{x}_i) = \frac{G_j(\vec{x}_i) - \min(G_j(\vec{x}_i))}{\max(G_j(\vec{x}_i)) - \min(G_j(\vec{x}_i))}$$

$$G_j(\vec{x}_i) = \begin{cases} \max\{0, g_j(\vec{x}_i)\}, & j = 1,..,l \\ \max\{0, \left|h_j(\vec{x}_i)\right| - \varepsilon\}, & j = l+1,..,m \end{cases}.$$

The function $G_{norm}(\vec{x}_i)$ is a measure of the constraint violation of an infeasible candidate solution, with higher values reflecting lower constraint violation. The value of function $G_{norm}(\vec{x}_i)$ instead of penalised objective function $f_p(\vec{x}_i)$, is now used to select potential candidate solutions for crossover operation using 'roulette selection' method.

A major problem with GAs is that of premature convergence, where the entire population consists of duplicates of sub-optimal solutions. To maintain genetic diversity we employ an additional step to remove duplicate candidates after every generation; these are substituted by randomly generated candidates. We carry forward the best performing feasible solution after every generation, to ensure it is not lost during this phase.

3. Solution of benchmark problems

In this Section we examine the effectiveness of the proposed algorithm in solving classical non-convex constrained optimisation problems, comparing its performance to that of established algorithms.

3.1 Non-convex constrained optimization problems

We study the performance of the proposed algorithm in the case of a non-convex [25] objective function, namely

$$f(x_1, x_2) = -\exp\left(ka\sqrt{x_1^2 + x_2^2}\right)\sin(ax_1)\cos(ax_2) \tag{8}$$

We minimize the objective function subjected to 3 different non-linear constraints:

Case 1:

$$g_1(x_1, x_2) = \exp(cx_1^2) - 1 - x_2 \le 0 \tag{9}$$

Case 2:

$$g_2(x_1, x_2) = \alpha\left[\left(\frac{\beta}{4\pi}x_1 + \gamma\right)\sin\left(\frac{\pi}{2}x_1\right)\sin\left(\frac{\pi}{2}x_2\right) - \delta\right] \le 0 \tag{10}$$

Case 3: constraints g_1 and g_2 simultaneously active.

Bounds on the design variables are $x_1 \in [0, 4\pi]$ and $x_2 \in [0, 2\pi]$. The values of the constants are taken as $a = 1$, $b = 0.6$, $k = 0.2$, $c = 0.012$, $\alpha = 0.1$, $\beta = 0.4$ and $\delta = 0.7$.

When comparing different algorithms we fix the total number of function evaluations (NFEs) to 20,000 ($P_N = 200,\ G_N = 100$), similar to [25,27]. A crossover probability of 0.85 and a mutation rate of 0.05 is used in all cases; 30 independent runs are performed using the proposed algorithm, and the statistical metrics of the objective function are as shown in Table 1; the proposed algorithm is able to find the optimal solutions in every run. As GAs are stochastic in nature, they do not necessarily converge around the optimal solution; however our algorithm does, as shown by the fact that the best and worst solutions found are very close to each other.

The rate of convergence of the objective function is plotted against the number of function evaluations (NFE) in Figure 1, for the 3 types of constraints. We see that the proposed algorithm converges at around 5000 NFEs for all cases. The best solution obtained using the proposed algorithm is compared to the solutions provided by 2 different algorithms, namely the BIANCA GA [25] and the rank-iMDDE [27], a differential evolution algorithm. The comparison is presented in Tables 2, 3 and 4 below, for the 3 different types of constraints considered.

Table 1: Statistical metrics of the objective function obtained using proposed algorithm over 30 independent runs for cases 1-3.

	Best	Worst	Mean	Std dev
Case 1	-8.11646195	-8.09050748	-8.1125394	0.00670
Case 2	-9.50126786	-9.50012748	-9.5011532	0.00024
Case 3	-7.48572844	-6.85586924	-7.4550397	0.12104

Table 2: Comparison of the design variables (x_1, x_2) , constraint violation (g_1) and objective function (f) of the best solution obtained using different algorithms for case 1.

	BIANCA GA [25]	Rank-iMDDE [27]	Proposed Algorithm
x_1	10.71119	10.71252	10.71252
x_2	2.96646	2.96338	2.96338
g_1	-4.43×10^{-3}	-1.128×10^{-11}	-5.14974×10^{-9}
f	-8.09933	-8.11646	-8.11646

Table 3: Comparison of the design variables (x_1, x_2) , constraint violation (g_2) and objective function (f) of the best solution obtained using different algorithms for case 2.

	BIANCA GA [25]	Rank-iMDDE [27]	Proposed Algorithm
x_1	11.27620	11.29429	11.29434
x_2	2.46284	2.47128	2.4713031
g_2	-1.0985×10^{-4}	-1.7688×10^{-12}	-6.467341×10^{-9}
f	-9.49783	-9.50127	-9.5012678

Table 4: Comparison of the design variables (x_1, x_2) , constraint violation (g_1, g_2) and objective function (f) of the best solution obtained using different algorithms for case 3.

	BIANCA GA [25]	Rank-iMDDE [27]	Proposed Algorithm
x_1	10.45320	10.47395	10.47395
x_2	2.71465	2.73013	2.730131
g_1	-3.8653×10^{-3}	-4.38089×10^{-12}	-1.12470×10^{-8}
g_2	-3.30729×10^{-3}	-1.16759×10^{-12}	-6.139986×10^{-9}
f	-7.37696	-7.48573	-7.485728

Clearly the proposed algorithm performs better than BIANCA and similarly to the more complex rank-iMDDE. Moreover, it is to be noted that the optimal results obtained using the proposed algorithm in Tables 2-4 are acquired using only 30 independent runs compared to the 100 independent runs used for the case of rank-iMDDE algorithm. Since both the proposed method and BIANCA use identical ADP techniques, the difference in results must be explained by the benefits of real-valued coding and the differences in GA operators and diversity mechanisms.

3.2 Benchmark engineering problems

The proposed GA is now applied to the case of widely studied engineering design problems: i) welded beam, ii) spring and iii) pressure vessel; its performance is compared to selected evolutionary genetic algorithms. One of these is GAFAT [28], a hybrid GA which uses centre-based differential crossover, non-uniform mutation (NUM) and Levenberg-Marquardt mutation (LMM). We also consider two algorithms for particle swarm optimization: HPSO, a hybrid algorithm proposed by He and Wang [29], and CVI-PSO, proposed by Mazhoud et al. [30]. For all cases, the maximum NFE is fixed to 100,000 ($P_N = 200,\ G_N = 100$), with crossover and mutation probabilities of 0.85 and 0.05, respectively. 30 independent runs are performed using the proposed algorithm for each engineering problem; the performance of other algorithms is obtained from the literature [27,28]. Remarkably, the proposed algorithm gave consistent results irrespective of the population size or number of generations used.

3.2.1 Design of a welded beam

Details of this problem can be found in [29]. Briefly, the objective is to minimize the overall cost of the beam subject to constraints on shear and bending stresses, buckling load, end deflection, side constraints as well as box constraints on the design variables. Table 5 summarises the performance of several algorithms in this optimisation problem. Most algorithms agree on the best solution; the proposed algorithm performs better than BIANCA and HPSO; GAFAT and rank-iMDDE are significantly better, in terms of standard deviation, than all other algorithms, displaying also the fastest convergence rates.

Table 5: Comparison of statistical metrics of results obtained by different algorithms for the welded beam problem.

Algorithm	Best	Worst	Mean	Std dev	Max_NFEs
BIANCA [25]	1.725436	1.793233	1.752201	2.30E-02	80,000
GAFAT [28]	1.724852	1.724852	1.724852	5.80E-16	20,000
HPSO [29]	1.724852	1.814295	1.74904	4.00E-02	81,000
CVI-PSO [30]	1.724852	1.727665	1.725124	6.12E-04	25,000
rank-iMDDE [27]	1.724852	1.724852	1.724852	9.06E-16	19,830
Proposed	**1.724885**	**1.756508**	**1.739539**	**1.68E-02**	**40,000**

3.2.2 Design of a tension-compression spring

The problem is as described in [26, 27] and the aim is to minimize the weight of a tension/compression spring subject to constraints on the minimum deflection, shear stress, surge frequency, outer diameter and given bounds on the design variables. Table 6 shows that the proposed algorithm has the fastest convergence rate and provides a best solution similar to those obtained by other hybrid algorithms and better than that found with the 'pure' GA BIANCA.

Table 6: Comparison of statistical metrics of results obtained by different algorithms for the tension-compression spring problem.

Algorithm	Best	Worst	Mean	Std dev	Max_NFEs
BIANCA [25]	0.012671	0.012913	0.012681	5.12E-05	80,000
GAFAT [28]	0.012665	0.012665	0.012665	3.20E-17	20,000
HPSO [29]	0.012665	0.012719	0.012707	1.60E-05	81,000
CVI-PSO [30]	0.012666	0.012843	0.012731	5.58E-05	25,000
rank-iMDDE [27]	0.012665	0.012668	0.012665	2.45E-07	19,565
Proposed	**0.012666**	**0.012815**	**0.012705**	**2.74E-05**	**20,000**

3.2.3 Design of a pressure vessel

The design of a cylindrical pressure vessel with hemispherical heads has to be optimised, to minimize the overall cost of production including the cost of materials, forming and welding; full details of this problem can be found in [29]. The statistical metrics of the results for this problem are as given in Table 7. For this problem all algorithms except BIANCA are able to find the best solution. In terms of other statistical metrics, the proposed algorithm performs better than BIANCA, HPSO and CVI-PSO.

Table 7: Comparison of statistical metrics of results obtained by different algorithms for the pressure vessel problem.

Algorithm	Best	Worst	Mean	Std dev	Max_NFEs
BIANCA [25]	6059.94	6447.32	6182.00	122.32	80,000
GAFAT [28]	6059.71	6059.71	6059.71	2.80×10^{-12}	30,000
HPSO [29]	6059.71	6288.68	6099.93	86.20	81,000
CVI-PSO [30]	6059.71	6820.41	6292.12	288	25,000
rank-iMDDE [27]	6059.71	6059.71	6059.71	1.95×10^{-12}	23,465
Proposed	**6059.71**	**6097.7752**	**6075.16**	**16.4132**	**40,000**

The rate of convergence of the objective function normalized by the value of the best solution obtained is plotted against the number of function evaluations (NFE) in Figure 2, for the 3 engineering benchmark problem discussed above. We see that the proposed algorithm converges at around 20000 NFEs for all problems. In summary, the proposed algorithm is capable of dealing effectively with complex non-convex constrained optimisation problems, as well as in classical engineering design optimisation problems. The algorithm performs better than BIANCA and provides results similar to the other algorithms. Overall, GAFAT is the best performing algorithm, while rank-iMDDE on average requires less NFEs to converge. The good performance of GAFAT can be explained by the use of LMM operator, which is a gradient-based repair operator. The use of such operator is, however, numerically expensive and requires knowledge of gradients of constraints, which limits its realm of applicability.

In summary, compared to "pure" (non-hybrid) GAs, the proposed algorithm performs better than other state-of-the-art algorithms such as BIANCA, giving results that are on par with

hybrid algorithms; the proposed algorithm provides a good combination of programming simplicity, computational efficiency and quality of results.

4. Laminate Analysis

The proposed algorithm is now applied to the problem of damping optimization in fibre composite laminates with viscoelastic inserts. The use of viscoelastic inserts significantly increases the damping of composite laminates but also degrades their stiffness and increases mass. The aim of the constrained optimisation is to find an optimal laminate configuration which maximizes the sum of the first *N* modal loss factors of a composite structure, subject to constraints on the maximum in-plane and bending stiffness degradation and increase in mass. The design variables considered are the ply thickness, ply orientation and the location of viscoelastic inserts within the laminate's stack. Such a problem is highly discrete and non-convex in nature.

The structure considered is a slender cantilever, with length and width of 100mm and 25mm respectively. We perform the optimization study for two different anisotropic viscoelastic fibre composite plies, i.e. HMS carbon/DX210 epoxy and E-glass/SR1500 epoxy; the mechanical properties of these composite plies are taken from literature [33,34] and given in Table 8, where $\eta_{11}, \eta_{22}, \eta_{12}, \eta_{23}$ are the material loss factors in longitudinal, transverse, axial shear and transverse shear mode of vibration, respectively. For both cases, the properties of the damping material inserts considered are taken as those of a widely used viscoelastic material (ISD 112, [35]). This is modelled in this study as an isotropic, linearly viscoelastic solid with $E_v = 70\text{MPa}$, $\nu_v = 0.3$ and the material loss factor $\eta_v = 0.5$. The densities of laminae and viscoelastic plies are taken as 1500kg/m^3 and 968kg/m^3, respectively.

Table 8: Elastic and damping properties of carbon-epoxy and glass-epoxy lamina.

	GFRP	CFRP
$E_{11}(\mathrm{GPa})$	29.9	172
$E_{22}(\mathrm{GPa})$	5.85	7.17
$G_{12}(\mathrm{GPa})$	2.45	3.76
$G_{23}(\mathrm{GPa})$	2.45	3.76
ν_{12}	0.26	0.29
$\eta_{11}\%$	0.35	0.071
$\eta_{22}\%$	1.30	0.67
$\eta_{12}\%$	1.80	1.12
$\eta_{23}\%$	1.80	1.12

4.1 Loading Conditions

The primary damping mechanism in a hybrid laminate is via shearing or extensional strains of the viscoelastic insert, while a minor contribution is due to the inherent material damping of the composite laminae. The damping properties of such hybrid laminate structures can be quantified using the concept of modal loss factors [36,37], defined as

$$\eta_s^i = \frac{D_c^i + D_v^i}{U_c^i + U_v^i} \tag{11}$$

In Eq. (11), η_s^i is the modal loss factor corresponding to the *i-th* mode of vibration of a certain structure. *D* and *U* represent the dissipated and strain energies, respectively; the subscripts *c* and *v* refer to composite or viscoelastic plies. The dissipated energies in the case of composite and viscoelastic plies are given as

$$D_c^i = \frac{1}{2}\int_V (\eta_{c11}\sigma_{c11}^i\varepsilon_{c11}^i + \eta_{c22}\sigma_{c22}^i\varepsilon_{c22}^i + \ldots + \eta_{c12}\sigma_{c12}^i\varepsilon_{c12}^i)dV \tag{12}$$

$$D_v^i = \frac{1}{2}\int_V \eta_v(\sigma^i : \varepsilon^i)dV \tag{13}$$

where η_{cij} and η_v represent, respectively, the anisotropic loss factors in the case of fibre composites and the isotropic loss factor of viscoelastic inserts. To save computational cost, we assume that the damping properties of both composite and viscoelastic plies are independent

of frequency; however this dependence can be easily included, if needed, by using an iterative modal strain energy method scheme [38] within a FE framework.

The commercial FE software ABAQUS standard is used to evaluate the modal loss factors associated with the first three vibration modes of the cantilever, conducting a natural frequency extraction step; we also calculate the stiffness of the laminated cantilever by performing additional quasi-static FE simulations. We consider both in-plane and bending stiffness and at every iteration we evaluate the in-plane stiffness in directions *x* and *y* (*x* and *y* define the plane of the cantilever, while *z* is the through-thickness direction), as well as the bending stiffness of the cantilever loaded by a transverse end-force.

In order to accurately capture, in the simulations, the shearing of the compliant viscoelastic layers, the laminate is meshed by 3D solid elements with quadratic shape functions (C3D20). A mesh convergence study showed that an element size of 5 x 5 mm in the *x*,*y* plane and a single element per layer in the through-thickness direction, was sufficiently small to give mesh-independent results.

4.2. Formulation of the optimisation framework

We now proceed to define the optimization problem. The objective function of this study is to maximize the sum of first *N* modal loss factors, or to minimise the negative of their sum:

$$
\begin{aligned}
&\phi = -\sum_{i=1}^{N} \eta_i \\
&\min \ \phi(x_i) \\
&\text{s.t.} \\
&\begin{cases}
g_X(x) = \dfrac{R_X^{ref} - R_X(x_i)}{R_X^{ref}} - Tol_X \le 0 \\
g_Y(x) = \dfrac{R_Y^{ref} - R_Y(x_i)}{R_Y^{ref}} - Tol_Y \le 0 \\
g_{Flex}(x) = \dfrac{R_{Flex}^{ref} - R_{Flex}(x_i)}{R_{Flex}^{ref}} - Tol_{Flex} \le 0 \\
g_{mass}(x) = \dfrac{M(x_i) - M^{ref}}{M^{ref}} - Tol_{Mass} \le 0
\end{cases}
\end{aligned}
\tag{14}
$$

The constraints g_X, g_Y and g_{Flex} limit the variation of the values of in-plane and bending laminate stiffness, R_X, R_Y, R_{Flex}, respect to a reference value; similarly, g_{mass} expresses a

constraint on the maximum allowable variation in mass. The *ref* subscript refers to a reference laminate, which for this study is chosen to be a standard quasi-isotropic composite laminate with no viscoelastic interlayers. Tol_X, Tol_Y, Tol_{Flex} and Tol_{mass} quantify the maximum allowable change in the stiffness parameters or structural mass, with respect to the reference laminate. In this study all tolerances are fixed to 0.05.

The proposed GA was implemented in the FE environment using the Python 2.7 scripting language, as this allows a programming interface with ABAQUS standard. A candidate solution laminate x_i with n plies was encoded using real variables as follows:

$$x_i = \left[[ID_1, t_1, \theta_1], [ID_2, t_2, \theta_2], ..., [ID_n, t_n, \theta_n]\right] \tag{15}$$

where each ply contains three design variables: a ply ID, which identifies if a given ply is composite or viscoelastic ply, a ply thickness *t* and a laminate orientation θ (this is the angle formed by the fibre direction and direction *x*). The ply ID is an integer with value either 0 or 1 corresponding to a viscoelastic or composite ply. The ply thickness was allowed to vary between 0.125 mm and 1.25 mm with discretization steps of 0.125 mm, which correspond to a thickness of a typical prepreg composite ply. The ply orientation was allowed to vary between -75 to 90 with increments of 15 degrees. A schematic diagram of the composite laminate showing the reference coordinate system and design variables are shown in Figure 3.

When using the crossover and mutation operators, appropriate rounding is carried out for the thickness and orientation variables, so they adhere to the prescribed discretization scheme. In total, the optimization problem includes *3n* design variables, with no assumption on the symmetry of the laminate. We apply the proposed algorithm to the two optimisation problems described below; in both cases we use a population size of 30 and crossover and mutation rates of 0.85 and 0.05, respectively. For the composite beam problem, the maximum number of generations is arbitrary. The optimization was conducted until the value of the objective function was observed to remain consistent for at least 50 generations, after which convergence was assumed.

4.3 Results and discussion

Case I. We consider a symmetric laminate with predefined locations of the viscoelastic inserts, such that the *ID* variable of each ply is fixed and only the thickness *t* and orientation θ of the laminae are considered as design variables. Due to the symmetry conditions, the size of the optimization problem reduces to finding thicknesses of three plies and the orientation of two FRP plies, resulting in a total of 5 design variables. We choose a configuration [C/V/C/V/C], where C represents a composite ply (GFRP or CFRP) and V a viscoelastic insert. The rate of convergence of the optimal solution for the case of both GFRP and CFRP is as shown in Figure 4; we find that the algorithm starts with infeasible solutions and is able to converge at around 2500 NFEs for both GFRP and CFRP hybrid laminates.

The results for the optimal solution and reference laminate for the case of GFRP hybrid laminate are shown in

Table 9; the optimal laminate configuration consists of viscoelastic layers of similar thickness as the 0° composite plies and a 90° ply of maximum thickness. The proposed algorithm efficiently explores regions close to unfeasible domains, given that the constraint on the mass is closely met; the stiffness parameters increase, rather than decreasing, by introduction of the viscoelastic layers. The three loss factors, corresponding to the first two bending modes and a torsional mode, increase by an order of magnitude compared to the reference laminate. For comparison, we also include the results obtained for the optimal configuration but without the viscoelastic plies. We find that this configuration has on average 5.5% higher in-plane load carrying capacity as well as 17.3% higher overall damping compared to the reference configuration, with a 16.6 % reduction in mass. However, when compared to the optimal solution with viscoelastic inserts, we observe that the flexural load carrying and damping performance of the structure are greatly reduced. The reduction in the flexure stiffness can be explained by the fact that in absence of the viscoelastic inserts the stiffer composite plies are closer to the neutral axis of the beam. Similar results are obtained for the case of CFRP as given in Table 10. We observe that for similar vibration modes (two bending and torsional modes), the modal loss factors are higher for the case of CFRP compared to those of GFRP. This is due to the higher stiffness mismatch between 0° and 90° CFRP plies which induce higher shear deformation in the viscoelastic layer.

Table 9: Details of the optimal result found using proposed algorithm for GFRP laminate case 1.

	Reference Configuration	Optimal Solution (GFRP)	Optimal Solution without viscoelastic plies
Orientation (°)	[0/45/-45/90]s	[0/V/90/V/0]	[0/90/0]
Thickness (mm)	[0.375/0.375/0.375/0.375]	[0.625/0.5/1.25/0.5/0.625]	[0.625/1.25/0.625]
Mass (g)	11.25	11.795 (4.8%)	9.375 (-16.6%)
Force X direction (N)	10413	11225 (8.1%)	11240.66 (7.9%)
Force Y direction (N)	174912	180693 (3%)	180456.53 (3.1%)
Force flexure (N)	3.58	5.436 (52.7%)	2.637 (-26.3%)
η_1	0.0051	0.06704	0.0038
η_2	0.0094	0.18970	0.0159
η_3	0.0057	0.22362	0.0041
Objective function	- 0.02028	-0.48034	- 0.0238

Table 10: Details of the optimal result found using proposed algorithm for CFRP laminate case 1.

	Reference Configuration	Optimal Solution (CFRP)	Optimal Solution without viscoelastic plies
Orientation (°)	[0/45/-45/90]s	[0/V/90/V/0]	[0/90/0]
Thickness (mm)	[0.375/0.375/0.375/0.375]	[0. 5/0.75/1.125/0.75/0.5]	[0. 5/1.125/0.5]
Mass (g)	11.25	11.6 (3.1%)	7.96 (-29.2%)
Force X direction (N)	46147	45093.45 (-2.3%)	45149.60 (-2.3%)
Force Y direction (N)	789204	805436.85 (2.1%)	805340.93 (2.0%)
Force flexure (N)	18.45	18.95 (2.7%)	8.83 (-52.1%)
η_1	0.00128	0.24371	0.00084
η_2	0.00381	0.22349	0.0085
η_3	0.00191	0.33424	0.0013

Objective function	-0.00700	-0.80145	-0.01064

Case II: We now consider a more general problem: we optimise the damping performance of a symmetric laminate with a total of 8 plies, with no predefined location of the viscoelastic inserts, giving a total of 12 design variables (4 for each ply id, 4 for each ply thickness and 4 for each ply orientation). Doubling the number of design variables results in a longer convergence time, of around 6000 NFEs for both GFRP and CFRP plies, as shown in Figure 5.

The optimal and reference laminate configurations for the GFRP laminate are shown in Table 11. The optimal configuration consists of 0° and 90° plies of maximum thickness to satisfy the in-plane stiffness requirements in X and Y directions, respectively. The viscoelastic insert is located off centre within the laminate to provide maximum dissipation for the three modes analysed here. The constraints g_X, g_Y and g_{mass} are met closely, with the mass slightly increasing, but most stiffness parameters showing a minor improvement. The optimal solution without the viscoelastic layers exhibits similar in-plane stiffness performance with a 15% mass reduction, albeit with a 23% reduction in the flexural load carrying capacity compared to the reference configuration, similar trends as observed for case 1 in Table 9. However, again the modal loss factors are an order of magnitude lower than the optimal configuration with the viscoelastic layers.

Table 12 presents the results of the case 2 for CFRP-viscoelastic plies. We notice that the layup orientation of the optimal solution is identical to that obtained for the case of GFRP laminate in Table 11. However in this case, the optimal solution consists of viscoelastic layer of minimal thickness sandwiched between composite plies of near maximum thickness. This is due to the higher axial stiffness mismatch between CFRP and viscoelastic plies, where any addition of viscoelastic plies strongly affects the stiffness of the laminate. Again, the capability of the algorithm to search very close to infeasible regions is clearly demonstrated. The optimal solution shows an order of magnitude increase in modal loss factors, along with minor increase in mass and decrease of the in-plane stiffness in direction *y*. For the case of optimal solution without the viscoelastic plies, it is to be noted that for the case of CFRP, unlike GFRP, there is a marked increase in both longitudinal in-plane (15.8%) and flexural load carrying capacity (20%) compared to the reference configuration, whilst having same mass.

Table 11: Details of the optimal result found using proposed algorithm for GFRP laminate case 2.

	Reference Configuration	Optimal Solution (GFRP)	Optimal Solution without viscoelastic plies
Orientation (°)	[0/45/-45/90]s	[0/-45/V/90]s	[0/-45/90]s
Thickness (mm)	[0.875/0.875/0.875/0.875]s	[1.25/0.5/0.875/1.25]s	[1.25/0.5/1.25]s
Mass (g)	26.25	26.73(1.8%)	22.25 (-15.2%)
Force X direction (N)	24035.15	24408(1.4%)	24769.14 (-3.1%)
Force Y direction (N)	406049.75	399503(-1.8%)	400361.15 (-1.4%)
Force flexure (N)	44.73	46.44(3.9%)	34.35 (-23.2%)
η_1	0.00536	0.17038	0.00427
η_2	0.00757	0.23750	0.00662
η_3	0.0113	0.01033	0.01383
Objective function	-0.02012	-0.418209	-0.02472

Table 12: Details of the optimal result found using proposed algorithm for CFRP laminate case 2.

	Reference Configuration	Optimal Solution (CFRP)	Optimal Solution without viscoelastic plies
Orientation (°)	[0/45/-45/90]s	[0/45/V/90]s	[0/45/90]s
Thickness (mm)	[0.875/0.875/0.875/0.875]s	[1.25/1.125/0.125/1.125]s	[1.25/1.125/1.125]s
Mass (g)	26.25	26.86(2.32%)	26.25 (0%)
Force X direction (N)	103855	118928.8 (14.5%)	120363.25 (15.8%)
Force Y direction (N)	1813552	1766393 (-2.6%)	1770846.2 (-2.35%)
Force flexure (N)	225	226.51 (0.70%)	270.8 (20.0%)
η_1	0.00176	0.141292	0.00165
η_2	0.00223	0.217772	0.00187
η_3	0.00628	0.013886	0.00711
Objective function	-0.01027	-0.37295	-0.01063

5. Conclusions

In this paper we explored the application of a new real-coded GA to the optimisation of damping in a hybrid composite-viscoelastic laminate. The GA uses the BLX-α crossover function with non-uniform mutation, with single individual elitist selection and removal of duplicate candidates at every generation. The performance of the proposed GA was found similar to that of other state-of-the-art evolutionary algorithms that utilize more complex selection and crossover operators, for the case of various non-convex mathematical and engineering benchmark problems.

The laminate problem is formulated as a constrained optimization problem, with the objective to maximize modal damping corresponding to the first three vibration modes, with constraints on the maximum mass increase and stiffness degradation. The design variables included the location of viscoelastic plies, the ply thickness and the stacking sequence. For the two cantilever optimisation problems considered in this study, the optimal solutions suggest that to maximise damping the laminate will tend to assume a 'sandwich' configuration, with stiff, thick plies at the upper and lower faces of the cantilever, and a 'core' comprising less stiff plies (with respect to the main loading direction) sandwiching viscoelastic layers. By this strategy the laminate damping is increased by an order of magnitude, at the expenses of minor decreases in stiffness and weight-effectiveness.

It should be noted that it was not possible to compare the damping response of the proposed GA to other state-of-the-art metaheuristic algorithms, due to lack of availability of the source code of the other algorithms; the code would need rewriting in Python language to allow for communication with the ABAQUS FE package employed in this study. The FE simulations require substantial computational time and only a single run of the GA was performed for this reason, hence the lack of statistical information. It is therefore impossible, presently, to conclude on which algorithm performs best in the optimisation of laminate damping; we leave the pursuit of the answer to this question to future studies.

Acknowledgements

This work is part of a collaborative R&T Project "SILOET II Project 10" which is co-funded by Innovate UK and Rolls-Royce plc and carried out by Rolls-Royce plc and Imperial College. The authors are grateful to Rolls-Royce plc for giving permission to publish it.

REFERENCES

[1] M.V. Pathan, V.L. Tagarielli, S. Patsias, P.M. Baiz-Villafranca, A new algorithm to generate representative volume elements of composites with cylindrical or spherical fillers, Compos. Part B Eng. 110 (2017) 267–278. doi:10.1016/j.compositesb.2016.10.078.

[2] M.V. Pathan, V.L. Tagarielli, S. Patsias, Numerical predictions of the anisotropic viscoelastic response of uni-directional fibre composites, Compos. Part Appl. Sci. Manuf. 93 (2017) 18–32. doi:10.1016/j.compositesa.2016.10.029.

[3] M.V. Pathan, V.L. Tagarielli, S. Patsias, Effect of fibre shape and interphase on the anisotropic viscoelastic response of fibre composites, Compos. Struct. 162 (2017) 156–163. doi:10.1016/j.compstruct.2016.11.046.

[4] M.V. Pathan, V.L. Tagarielli, J.A. Rongong, S. Patsias, Measurements and predictions of the anisotropic viscoelastic response of unidirectional carbon fibre composites, Compos. Sci. Technol. Rev. (n.d.).

[5] A.E. Mahi, M. Assarar, Y. Sefrani, J.-M. Berthelot, Damping analysis of orthotropic composite materials and laminates, Compos. Part B Eng. 39 (2008) 1069–1076. doi:10.1016/j.compositesb.2008.05.003.

[6] N. Le Maoût, E. Verron, J. Bégué, Simultaneous geometrical and material optimal design of hybrid elastomer/composite sandwich plates, Compos. Struct. 93 (2011) 1153–1157. doi:10.1016/j.compstruct.2010.10.008.

[7] M. Montemurro, Y. Koutsawa, S. Belouettar, A. Vincenti, P. Vannucci, Design of damping properties of hybrid laminates through a global optimisation strategy, Compos. Struct. 94 (2012) 3309–3320. doi:10.1016/j.compstruct.2012.05.003.

[8] C. Xu, S. Lin, Y. Yang, Optimal design of viscoelastic damping structures using layerwise finite element analysis and multi-objective genetic algorithm, Comput. Struct. 157 (2015) 1–8. doi:10.1016/j.compstruc.2015.05.005.

[9] A.L. Araújo, C.M. Mota Soares, C.A. Mota Soares, J. Herskovits, Optimal design and parameter estimation of frequency dependent viscoelastic laminated sandwich composite plates, Compos. Struct. 92 (2010) 2321–2327. doi:10.1016/j.compstruct.2009.07.006.

[10] D.A. Saravanos, C. Chamis, Multiobjective shape and material optimization of composite structures including damping, AIAA J. 30 (1992) 805–813. doi:10.2514/3.10988.

[11] J.F.A. Madeira, A.L. Araújo, C.M. Mota Soares, C.A. Mota Soares, Multiobjective optimization of viscoelastic laminated sandwich structures using the Direct MultiSearch method, Comput. Struct. 147 (2015) 229–235. doi:10.1016/j.compstruc.2014.09.009.

[12] Z. Xie, W. Steve Shepard Jr, K.A. Woodbury, Design optimization for vibration reduction of viscoelastic damped structures using genetic algorithms, Shock Vib. 16 (2009) 455–466.

[13] H. Zheng, G.S.H. Pau, Y.Y. Wang, A comparative study on optimization of constrained layer damping treatment for structural vibration control, Thin-Walled Struct. 44 (2006) 886–896. doi:10.1016/j.tws.2006.08.005.

[14] J.S. Grewal, R. Sedaghati, E. Esmailzadeh, Vibration analysis and design optimization of sandwich beams with constrained viscoelastic core layer, J. Sandw. Struct. Mater. 15 (2013) 203–228. doi:10.1177/1099636213476510.

[15] A. El-Sabbagh, A. Baz, Topology optimization of unconstrained damping treatments for plates, Eng. Optim. 46 (2014) 1153–1168. doi:10.1080/0305215X.2013.832235.

[16] A.L. Araújo, P. Martins, C.M. Mota Soares, C.A. Mota Soares, J. Herskovits, Damping optimization of viscoelastic laminated sandwich composite structures, Struct. Multidiscip. Optim. 39 (2009) 569–579. doi:10.1007/s00158-009-0390-4.
[17] C.M. Chia, J.A. Rongong, K. Worden, Strategies for using cellular automata to locate constrained layer damping on vibrating structures, J. Sound Vib. 319 (2009) 119–139. doi:10.1016/j.jsv.2008.06.023.
[18] T. Baeck, D.B. Fogel, Z. Michalewicz, Handbook of Evolutionary Computation, CRC Press, 1997.
[19] Z. Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs, Third, Springer-Verlag, New York, n.d.
[20] H. Zheng, C. Cai, X.M. Tan, Optimization of partial constrained layer damping treatment for vibrational energy minimization of vibrating beams, Comput. Struct. 82 (2004) 2493–2507. doi:10.1016/j.compstruc.2004.07.002.
[21] K. Deb, J. Sundar, Reference point based multi-objective optimization using evolutionary algorithms, in: Proc. 8th Annu. Conf. Genet. Evol. Comput., New York, 2006: pp. 635–642.
[22] F. Herrera, M. Lozano, J.L. Verdegay, Tackling real-coded genetic algorithms: Operators and tools for behavioural analysis, Artif. Intell. Rev. 12 (1998) 265–319.
[23] R.M. Krzanowski, J. Raper, Spatial Evolutionary Modeling, Oxford University Press, 2001.
[24] C. Si, J. Shen, X. Zou, Y. Duo, L. Wang, Q. Wu, A dynamic penalty function for constrained optimization, in: Y. Tan, Y. Shi, F. Buarque, A. Gelbukh, S. Das, A. Engelbrecht (Eds.), Adv. Swarm Comput. Intell., Springer International Publishing, Cham, 2015: pp. 261–272. http://link.springer.com/10.1007/978-3-319-20472-7_28 (accessed May 28, 2016).
[25] M. Montemurro, A. Vincenti, P. Vannucci, The Automatic Dynamic Penalisation method (ADP) for handling constraints with genetic algorithms, Comput. Methods Appl. Mech. Eng. 256 (2013) 70–87. doi:10.1016/j.cma.2012.12.009.
[26] L.J. Eshelman, J.D. Schaffer, Real-coded genetic algorithms and interval-schema, in: Proc Third Int Conf Genet. Algorithms, Morgan Kaufmann Publishers, San Mateo, n.d.: pp. 86–91.
[27] W. Gong, Z. Cai, D. Liang, Engineering optimization by means of an improved constrained differential evolution, Comput. Methods Appl. Mech. Eng. 268 (2014) 884–904. doi:10.1016/j.cma.2013.10.019.
[28] J. Zhao, L. Wang, P. Zeng, W. Fan, An effective hybrid genetic algorithm with flexible allowance technique for constrained engineering design optimization, Expert Syst. Appl. 39 (2012) 6041–6051. doi:10.1016/j.eswa.2011.12.012.
[29] Q. He, L. Wang, A hybrid particle swarm optimization with a feasibility-based rule for constrained optimization, Appl. Math. Comput. 186 (2007) 1407–1422. doi:10.1016/j.amc.2006.07.134.
[30] I. Mazhoud, K. Hadj-Hamou, J. Bigeon, P. Joyeux, Particle swarm optimization for solving engineering problems: A new constraint-handling mechanism, Eng. Appl. Artif. Intell. 26 (2013) 1263–1273. doi:10.1016/j.engappai.2013.02.002.
[31] J.S. Arora, Introduction to Optimum Design, McGraw-Hill, New York, 1989.
[32] A.D. Belegundu, J.S. Arora, A study of mathematical programming methods for structural optimization. Part I: Theory, Int. J. Numer. Methods Eng. 21 (1985) 1583–1599.
[33] R.D. Adams, Ni, R.G., Damping and dynamic moduli of symmetric laminated composite beams - theoretical and experimental results, J. Compos. Mater. 18 (1984) 104–121.

[34] J.-M. Berthelot, Y. Sefrani, Damping analysis of unidirectional glass and Kevlar fibre composites, Compos. Sci. Technol. 64 (2004) 1261–1278. doi:10.1016/j.compscitech.2003.10.003.
[35] 3M, 3M Viscoelastic Damping Polymers 112-130 Technical Data, 2012.
[36] Conor D. Johnson; David A. Kienholz, Finite element prediction of damping in structures with constrained viscoelastic layers, AIAA J. 20 (1982) 1284–1290. doi:10.2514/3.51190.
[37] E. Kerwin, Damping of flexural waves by a constrained viscoelastic layer, J. Acoust. Soc. Am. 31 (1959) 952–962.
[38] S.H. Zhang, H.L. Chen, A study on the damping characteristics of laminated composites with integral viscoelastic layers, Compos. Struct. 74 (2006) 63–69. doi:10.1016/j.compstruct.2005.03.008.

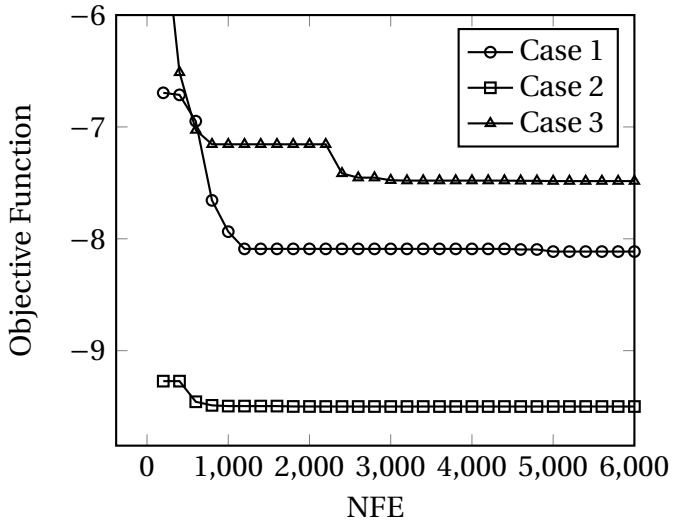


Figure 1: Convergence of the objective function of the best candidate solution with number of function evaluations for case 1-3.

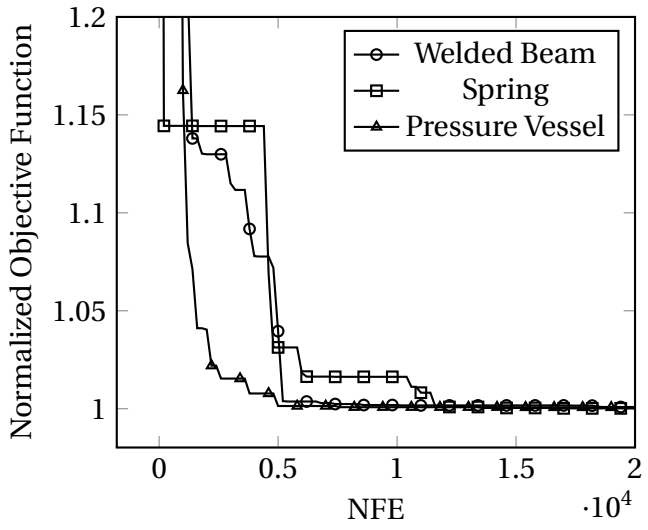


Figure 2: Convergence of the objective function corresponding to the best candidate solution for the case of engineering benchmark problems.

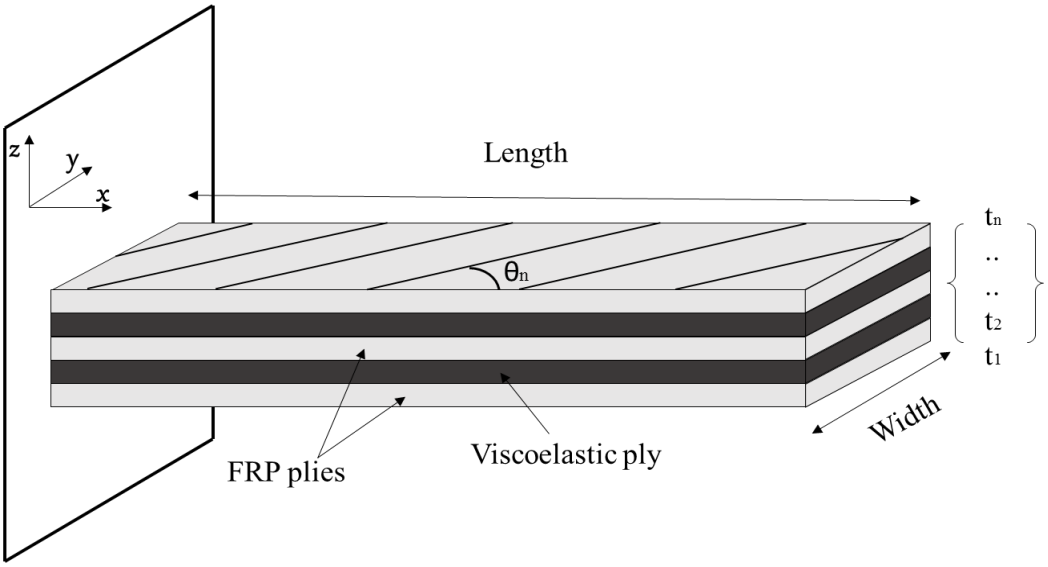
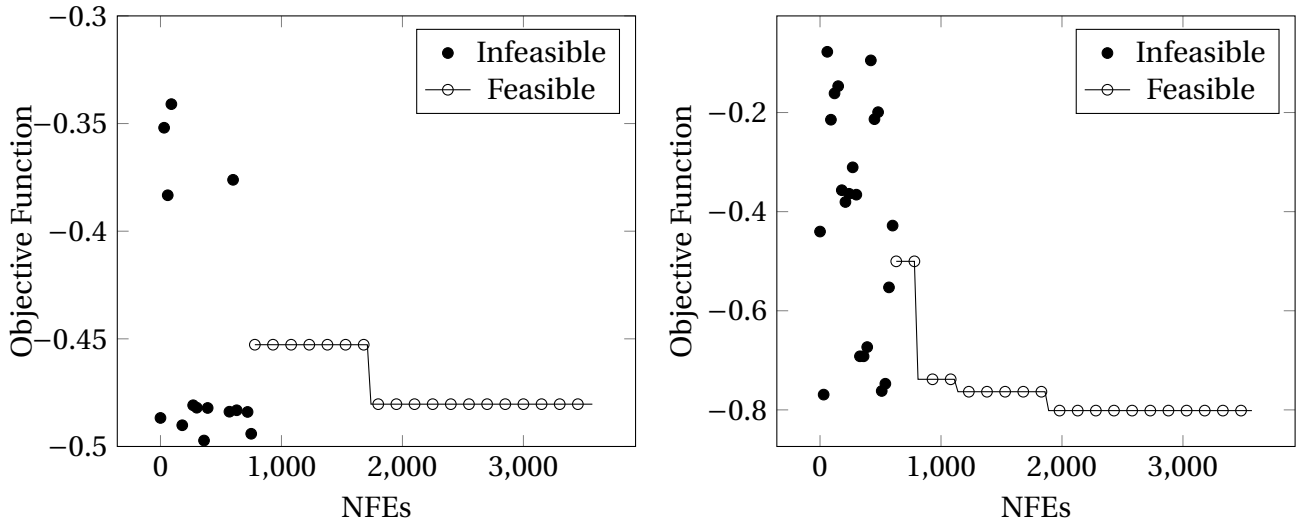
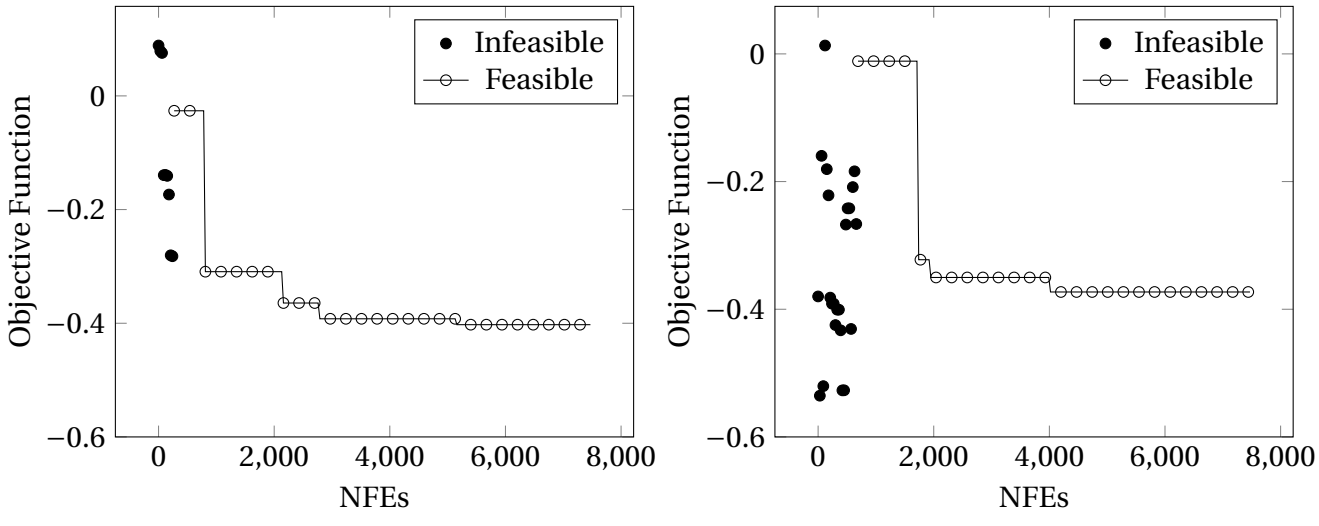


Figure 3: Schematic diagram of the fibre composite-viscoelastic sandwich beam optimization problem.



(a) (b)

Figure 4: Convergence of the objective function corresponding to the best candidate solution for laminate case 1; (a) GFRP; (b) CFRP.



(a) (b)

Figure 5: Convergence of the objective function corresponding to the best candidate solution for laminate case 2; (a) GFRP; (b) CFRP.