A floating connector element formulation for multi-level modelling of composite structures

E.S. Kocaman[a,1,*], B.Y. Chen[b], S.T. Pinho[a]

[a]*Department of Aeronautics, South Kensington Campus, Imperial College London. London SW7 2AZ, United Kingdom*
[b]*Faculty of Aerospace Engineering, Delft University of Technology. Kluyverweg 1, 2629 HS Delft, Netherlands*

Abstract

Design and optimisation of large structures, including the positioning of lower-level components, typically require extensive user involvement and sequential mechanical analysis/optimisation iterations. This paper presents an original method that enables adaptive positioning of lower-level models (such as components) within higher level-models (such as large structures), and that achieves a combined mechanical/optimisation problem for the design of structures with various hierarchical levels (such as the positioning of stiffeners within a wingbox). As the position of the lower-level model evolves, our proposed method does not require re-generating of the geometry, remeshing or modifying the stiffness matrix of the elements corresponding to the various hierarchical levels. Instead, we achieve the adaptive positioning via an original concept that we propose here: Floating Connector (FC) elements. In this paper, we validate the FC elements against reference purely-mechanical solutions, show that they can be combined with gradient-descent method and genetic algorithms, and that they can be applied to optimise the positioning of a stiffener runnout taking into account a debonding manufacturing defect.

Keywords:
Multi-level modelling, Structural design, Optimization, Composites

[*] Corresponding author
Email address: e.kocaman15@imperial.ac.uk (E.S. Kocaman)

[1] Tel: +44 7887816544. Address: Department of Aeronautics, South Kensington Campus, Imperial College London, London SW7 2AZ, UK.

1. Introduction

1.1. Background

In the design of large engineering structures, such as wing-boxes in aircraft, it is often the case that structural details, such as stiffeners, need to be designed and positioned during the design of the overall structure.

The current practice for the numerical design of these large structures with structural details involves creating large (yet typically coarse) finite element (FE) models of the large structure, as well as detailed models of the structural details, assuming a certain configuration. Eventually, these two models may be used in a multi-level framework.

Within the multi-level framework, there are various iterative [1–7] and concurrent [8–16] methods in the literature to include a structural detail into a larger model. In iterative (sub-modelling) approaches, a global and a local model are simulated separately within an iterative procedure. The results from one model are used to calculate the appropriate boundary conditions for the other model, until convergence is achieved. In concurrent approaches, structural details are directly integrated to the large model and the overall system is simulated simultaneously.

For optimising the positioning (or other characteristics) of the components, various approaches can be followed [15–22]. For instance, the software used for the mechanical analysis can then be used iteratively with an optimisation software. Alternatively, various mechanical FE models corresponding to various configurations that span the entire design space can be created and ran, and the optimisation can then be done using surrogate models [17–20] that use the sampled simulation data.

When proceeding as described above for the design of large engineering structures, there is typically a substantial investment required for creating each FE model for each new configuration of the large structure: in terms of the effort required of the modeller, the time it takes to create each new new (CAD and) FE model, and the associated financial cost. Alternatively, a significant investment can instead go towards creating a parametric mesh that can generate various configurations automatically. In either case, defining an optimal configuration then requires iterating between independent FE and optimisation software codes (some of this iteration can be automated via dedicated via dedicated software packages such as Isight [23]).

As a consequence of the above, detailed FE models of large structures are typically built only very late in the design stage, once many design decisions have already been frozen. Therefore, effective simulation methodologies that enable combined mechanical FE and optimisation analyses of large structures, allowing for the positioning (or other characteristics) of structural details to be defined during the analysis, would be key to enable an early use of numerical simulation during the design process.

Ideally, from the designer's point of view, a full numerical model of a large structure would not rigidly represent just one single configuration, but would instead be equally compatible with any number of configurations (Figure 1), with the parameters defining each configuration being themselves variables to be determined in a single combined mechanical / optimisation analysis.

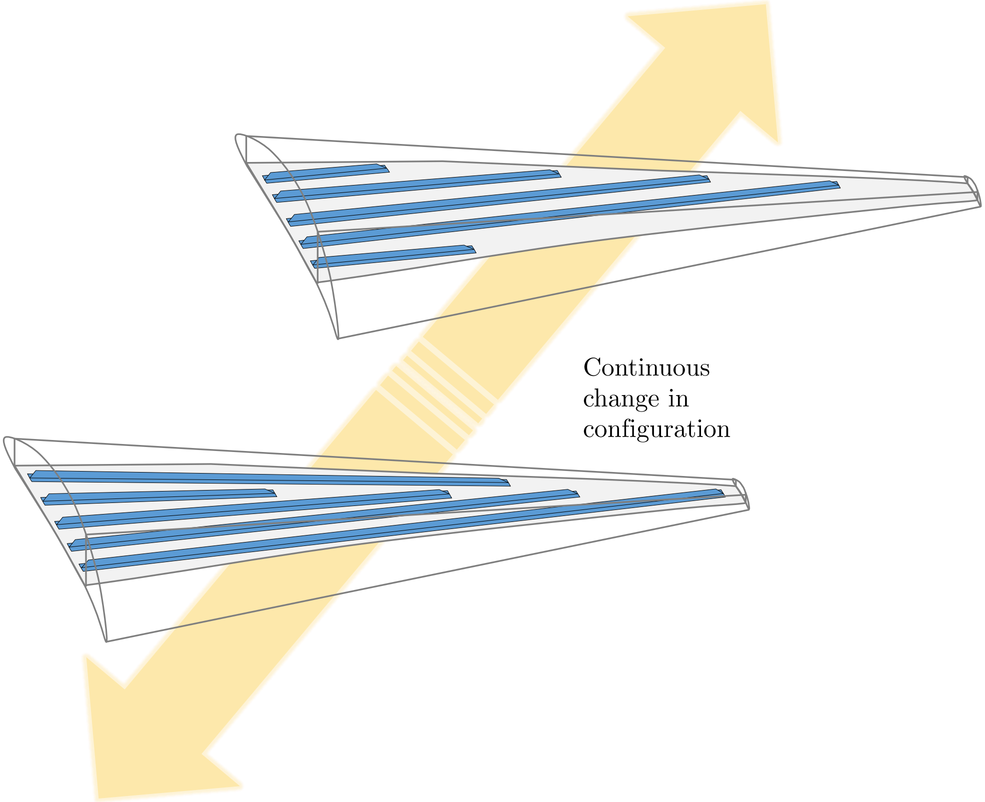


Figure 1: The full FE model of a large structure allows for the precise configuration to be defined during (and by) the solution process

1.2. Objective, novelty and outline

In this paper, we derive a new numerical framework for multi-level design of large structures, where design variables associated with sub-structures (e.g. the positioning of a component) are solved for in a numerical optimisation problem concurrently with the variables associated with the mechanical problem (displacements).

This numerical framework is underpinned by an original element type, which we present in this paper and call ‘Floating Connector (FC) element’, that can be used to connect sub-structures (i.e. components such as stiffeners) to an underlying large structure (e.g. a wing) without locking their relative position. Instead, this relative position is defined during the analysis itself via an optimisation code ‘inside’ the element.

This new element type constitutes a step change in the literature, in that it allows us to integrate different hierarchical levels of a structural model adaptively, without going through costly geometry and mesh generation as the configuration evolves, and in that it allows us to solve the optimisation problem concurrently with the mechanical problem. Additionally, we will show that FC elements also provide an improved platform for investigating the effect of various manufacturing defects (such as debonding between a stiffener and a skin).

The element description and formulation is presented in Section 2. Section 3 contains verification of the floating connection, while Section 4 contains verification of the concurrent optimisation capabilities. The application of the element for skin-stiffener debonding investigation is demonstrated in Section 5 followed by Section 6 where the implications of the results are discussed and the overall conclusions are summarized in Section 7.

2. Development of a *floating connector* methodology

2.1. Element description

The proposed element topology incorporates all the nodes along the interacting surfaces of the two bodies (that need to be connected) into one floating connector element (see Figure 2). This element contains floating connectors which introduce multi-point constraints (MPCs) between suitable Degrees of Freedoms (DoFs) that can be selected adaptively using a suitable optimization algorithm. The implementation of the element concept for a skin stiffener system is presented in Figure 2. The element topology for this case (see Figure 3) is composed of three different entities; these are the nodes at the top surface of the skin, at the bottom surface of the stiffener and the floating connectors that ensure the connection for a given stiffener position. The FC element also incorporates in its formulation an optimization algorithm to adaptively re-position the stiffener along the skin based on a chosen objective function (presented in Section 4).

Although we will explore in this paper the use of the floating connector element for design optimisation problems, the concept itself can also be used to investigate the effect of manufacturing defects. For instance, in the skin-stifener problem, the element can readily introduce debonding between stiffener and skin by removing the appropriate floating connectors in order to simulate a kissing bond, or to investigate the most critical areas for debonding location.

2.2. Mathematical formulation

To represent the connection between the respective n_{skn} skin and n_{stf} stiffener nodes, we can start by defining suitable MPC equations. Consider the stiffener DoFs $\mathbf{q}^{\text{stf}}$ and skin DoFs $\mathbf{q}^{\text{skn}}$ involved in these equations and arranged in a vector $\mathbf{q}^{\text{T}} = \begin{bmatrix} (\mathbf{q}^{\text{skn}})^{\text{T}} & (\mathbf{q}^{\text{stf}})^{\text{T}} \end{bmatrix}$. The overall MPC equation system

$$\mathbf{L}\mathbf{q} = \mathbf{0}, \tag{1}$$

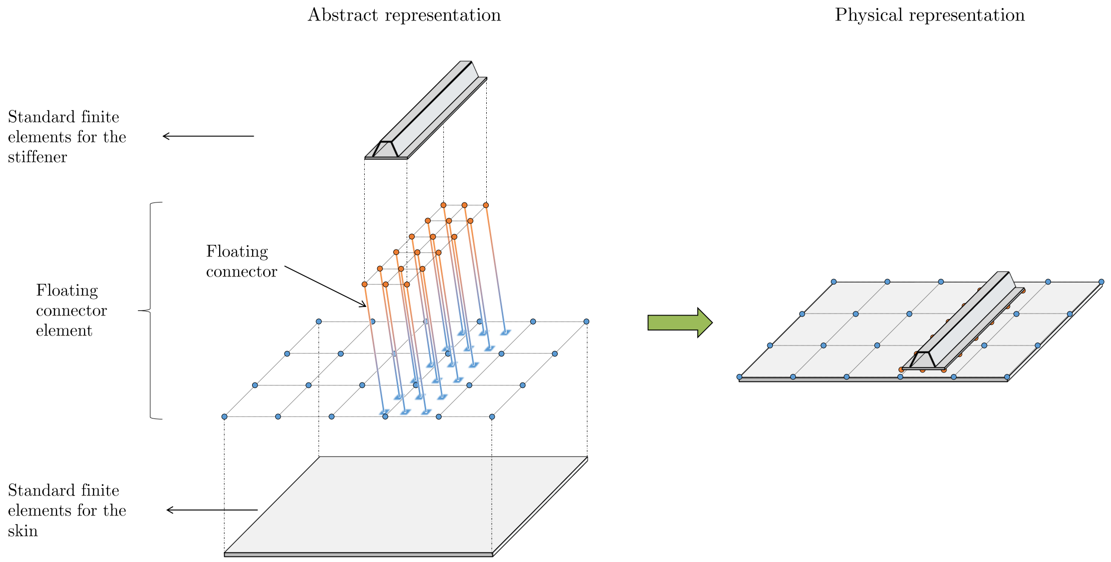

Abstract representation
Physical representation
Standard finite elements for the stiffener
Floating connector
Floating connector element
Standard finite elements for the skin

Figure 2: The floating connector element creates an evolving physical connection between two separate components, even if they are spatially 'distant' in the FE model

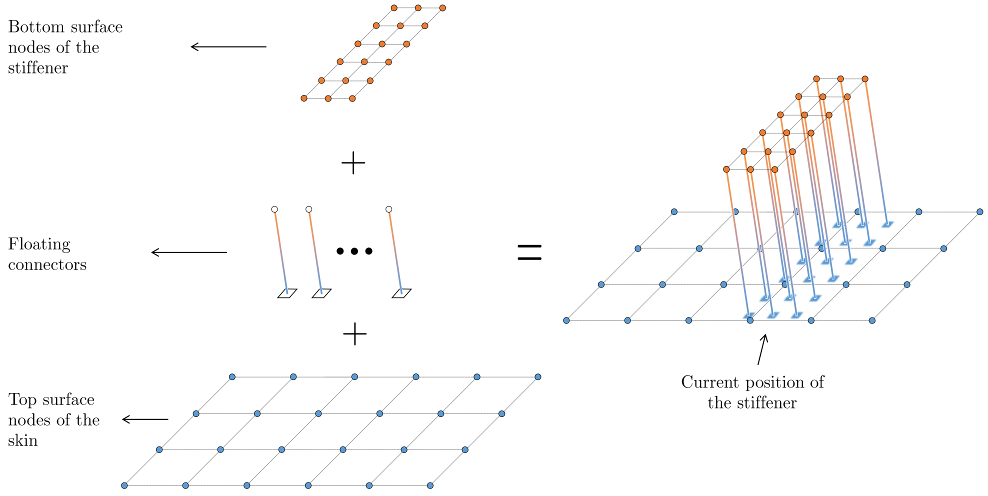

Bottom surface nodes of the stiffener
+
Floating connectors
+
=
Top surface nodes of the skin
Current position of the stiffener

Figure 3: Floating connector element topology

contains a coefficient matrix $\mathbf{L}$ with sub-matrices $\mathbf{L}^{\text{stf}}$ and $\mathbf{L}^{\text{skn}}$ relative to the stiffener and skin DoFs respectively:

$$\mathbf{L} = \begin{bmatrix} \mathbf{L}^{\text{skn}} & \mathbf{L}^{\text{stf}} \end{bmatrix}, \tag{2}$$

To define the coefficient matrix $\mathbf{L}$, and as shown in Figure 4, we consider a skin element with domain Ω^e, n^e nodes and shape function N_i^e $(i = 1, 2, ..., n^e)$, as well as a node k from the stiffener with position $\mathbf{x}_k$ that should be connected to the skin and overlaps this element domain Ω^e. The MPC equation (which can be used to define the DoFs of node k as a function of the DoFs of the skin element) can then be written as

$$\sum_{i=1}^{n^e} \mathbf{N}_i^e \mathbf{q}_i - \mathbf{q}_k = \mathbf{0}, \tag{3}$$

where the DoFs, expressed in a coordinate system x, y and z, as

$$\mathbf{q}_i^{\mathrm{T}} = \begin{bmatrix} q_i^x & q_i^y & q_i^z \end{bmatrix} \text{ and} \tag{4}$$

$$\mathbf{q}_k^{\mathrm{T}} = \begin{bmatrix} q_k^x & q_k^y & q_k^z \end{bmatrix}, \tag{5}$$

and $\mathbf{N}_i^e = N_i^e \mathbf{I}_3$ where $\mathbf{I}_3$ is the identity matrix with size 3. Noting Equation 3 can be expanded over all nodes n_{skn} of the skin:

$$\sum_{i=1}^{n_{\text{skn}}} \mathbf{N}_i^e \mathbf{q}_i - \mathbf{q}_k = \mathbf{0}. \tag{6}$$

Noting Equations 1, 2 and 6, it is clear that the matrices $\mathbf{L}^{\text{skn}}$ and $\mathbf{L}^{\text{stf}}$ are defined as

$$\mathbf{L}^{\text{skn}} = \begin{bmatrix} \mathbf{a}_{11} & \cdots & \mathbf{a}_{1i} & \cdots & \mathbf{a}_{1n_{\text{skn}}} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{a}_{k1} & \cdots & \mathbf{a}_{ki} & \cdots & \mathbf{a}_{kn_{\text{skn}}} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{a}_{n_{\text{stf}}1} & \cdots & \mathbf{a}_{n_{\text{stf}}i} & \cdots & \mathbf{a}_{n_{\text{stf}}n_{\text{skn}}} \end{bmatrix} \text{ and } \mathbf{L}^{\text{stf}} = -\mathbf{I}_{n_{\text{stf}}} \tag{7}$$

with

$$\mathbf{a}_{ki} = \begin{cases} \mathbf{N}_i^e(\mathbf{x}_k) & \Leftarrow \quad \mathbf{x}_k \subset \Omega^e \\ \mathbf{0} & \Leftarrow \quad \mathbf{x}_k \not\subset \Omega^e \end{cases}, \text{ and} \tag{8}$$

where $\mathbf{I}_{n_{\text{stf}}}$ is the identity matrix of size n_{stf}. Having defined the MPC equations, we then integrate them into the FC element using a penalty stiffness formulation. Considering the constraints $\mathbf{g}_{\text{c}}$ given by

$$\mathbf{g}_{\text{c}} = \mathbf{L}\mathbf{q} = \mathbf{0}, \tag{9}$$

the Courant quadratic penalty energy can be expressed as

$$U_{\text{p}} = k_{\text{p}} |\mathbf{g}_{\text{c}}|^2 = k_{\text{p}} \mathbf{q}^{\mathrm{T}} \mathbf{L}^{\mathrm{T}} \mathbf{L} \mathbf{q}, \tag{10}$$

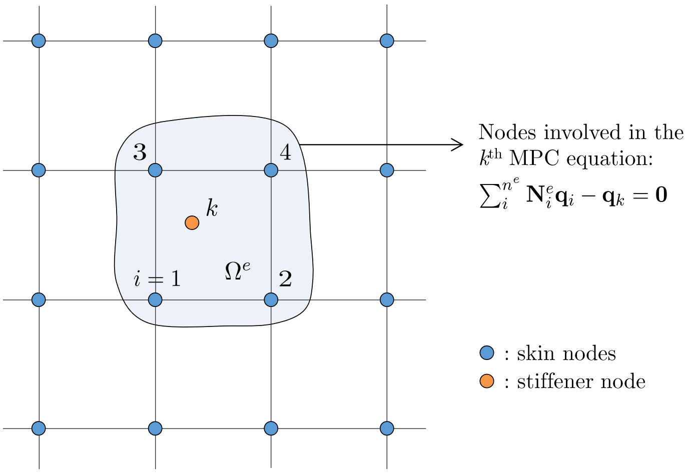

3
4
k
$i = 1$
Ω^e
2
Nodes involved in the
k^{th} MPC equation:
$\sum_i^{n^e} \mathbf{N}_i^e \mathbf{q}_i - \mathbf{q}_k = \mathbf{0}$
● : skin nodes
● : stiffener node

Figure 4: MPC implementation

where k_{p} is the penalty stiffness. Then, minimization with respect to the DoFs yields the element stiffness matrix $\mathbf{K}_{\text{p}}$ as

$$\mathbf{K}_{\text{p}} = k_{\text{p}} \mathbf{L}^{\text{T}} \mathbf{L}, \tag{11}$$

with the internal force vector $\mathbf{f}_{\text{p}}$ given by

$$\mathbf{f}_{\text{p}} = \mathbf{K}_{\text{p}} \mathbf{q}. \tag{12}$$

It is worth noting that the concept of floating connector element is not restricted to the penalty stiffness method. Other methods (e.g. such as Augmented Lagrangian or cohesive zone modelling) can also be used and the element stiffness can be derived accordingly.

This formulation can be readily implemented in a user-defined element subroutine in a typical finite element code and integrated into a numerical model composed of standard finite elements accounting for the skin and stiffener parts. This point is worth emphasizing: the floating connector element is defined independently of the two components it connects; hence these two components can be modelled directly using any element types available to the modeller in a particular finite element software, with the floating connector elements being simply added afterwards.

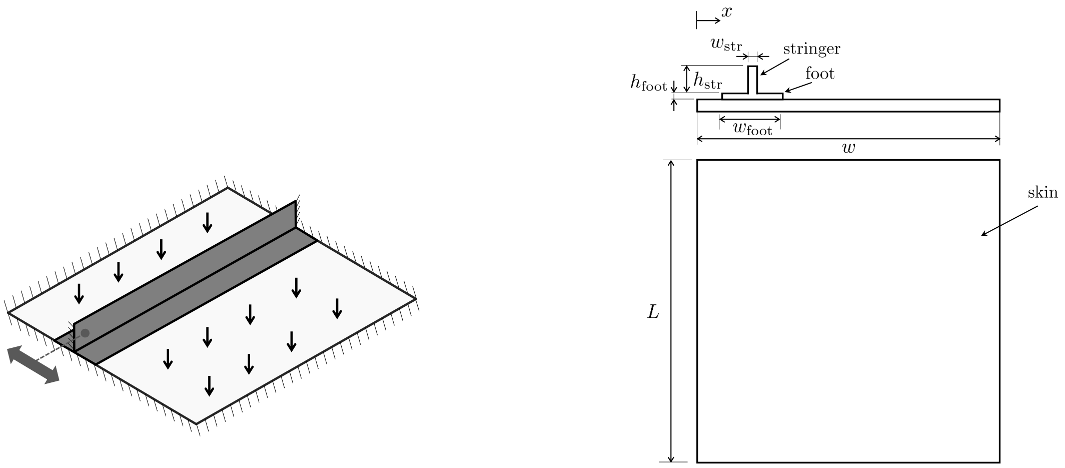
The implementation of the element in a commercial FE software starts with the creation of an FE model for each different level (the skin and stiffener in our case). The individual skin and stiffener models are introduced into the same simulation platform where they can be positioned in any location. An input file that contains all the information for nodes, elements, and connectivities is generated via the commercial FE software. Then, using a pre-processing code, all the nodes of the skin and stiffener that can potentially interact are extracted from the input file. In our case, these nodes include the nodes on the bottom surface of the stiffener and the top surface of the skin. Using these nodes (in particular their nodal number), a user-defined element corresponding to the floating connector element is constructed (element nodes are specified) and added into the input file. The modified input file is used to start the simulation where the user-element code performs the FE calculations for the floating connector element, whereas the commercial software performs calculations for the standard elements. The use of a floating connector element is therefore relatively straight-forward. More importantly, it can also potentially be incorporated in standard FE pre-processors, making its use as trivial as using as contact algorithm.

3. Verification of the floating connection

3.1. Introduction

In order to confirm that the proposed floating connector element correctly represents the connection between two components, we implemented it in Abaqus [24] and defined a verification test case where we compared the elastic response of models using the floating connector to that of models using standard built-in tie constraints. To this end, a skin-stiffener system as shown in Figure 5a was simulated with the geometrical dimensions given in Figure 5b and Table 1. In this verification case, the skin was exposed to 10 MPa pressure loading at the top surface and the design variable is the position x of the stiffener (Figure 5b). The edges of the skin and stiffener are clamped as illustrated in Figure 5a. The skin and stiffener parts are composed of uni-directional plies with quasi-isotropic stacking sequence and 0.125 mm. The skin and the stringer part of the stiffener (see Figure 5b) has a stacking sequence of $[0/-45/45/90]_{2\mathrm{s}}$ whereas the stiffener foot has $[0/-45/45/90]_{\mathrm{s}}$. The material properties of the plies are provided in Tables 2 and 3. The Courant penalty stiffness was chosen as $10^8\,\mathrm{N/mm}^3$.

We created two different models using either solid or shell meshes (the solid mesh is shown in Figure 6), and in both cases verified that the results were mesh converged. We ran one simulation using the floating connector element so that the stiffener sweeps different positions x (see Figure 5b), and recorded the corresponding maximum deflection of the skin. We then ran several simulations (one for each stiffener position) using instead the standard built-in tie constraints available in Abaqus [24], and repeated the whole process using solid (Section 3.2) and shell (Section 3.3) elements. The results for these case are provided Sections 3.2 and 3.3.



(a) Test schematic for the skin-stiffener system

(b) The geometry of the skin-stiffener system

Figure 5: Skin-stiffener system schematic and geometry

Table 1: Geometry of the skin-stiffener system

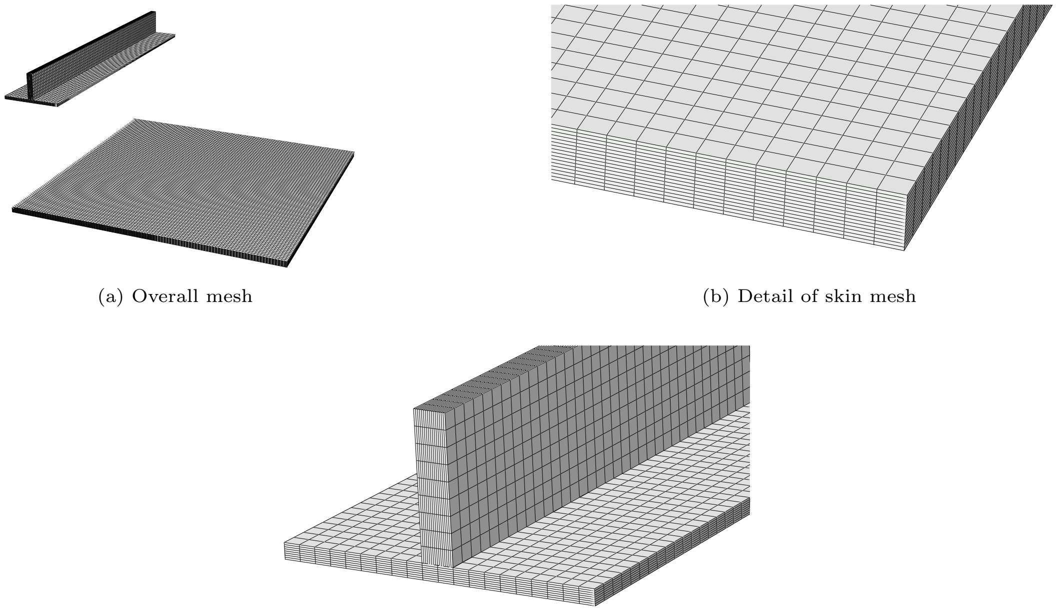
Description	Symbol	Value (mm)
Panel length	l	100
Panel width	w	100
Stringer width	w_{stringer}	20
Stringer height	h	9
Panel thickness	t	2
Stringer thickness	t_{stringer}	2
Foot thickness	t_{stringer}	1

Table 2: Elasticity related material properties for IM7-8552 [25]

E_{11} (GPa)	$E_{22} = E_{33}$ (GPa)	$\nu_{12} = \nu_{13}$	ν_{23}	$G_{12} = G_{13}$ (GPa)	G_{23} (GPa)
161	11.38	0.32	0.44	5.17	3.98

Table 3: Fracture and strength related material properties for IM7-8552 [25]

G_{Ic} (kJ/m^2)	G_{IIc} (kJ/m^2)	η
0.21	0.77	2.1



(a) Overall mesh

(b) Detail of skin mesh

(c) Detail of stiffener mesh

Figure 6: The mesh composed of solid elements for the verification simulation

3.2. Solid to solid connection

For the model with solid elements, we used solid hexahedral elements with linear shape functions (C3D8). All of the elements in the model have dimensions of 1 x 1 x 0.125 mm such that each ply is represented with one element in the thickness. We verified that this mesh size leads to mesh-converged results. The maximum skin deflection for each stiffener position is shown in Figure 7.

For the simulation with the floating connector element, Figure 8 illustrates the evolution of the mesh as the status of the floating connector element represents a connection between the stiffener and the skin at different positions x. Together, Figures 7 and 8 demonstrate that the floating connector element correctly represents the connection between solid components.

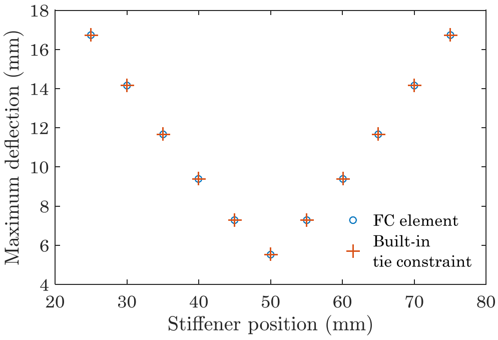
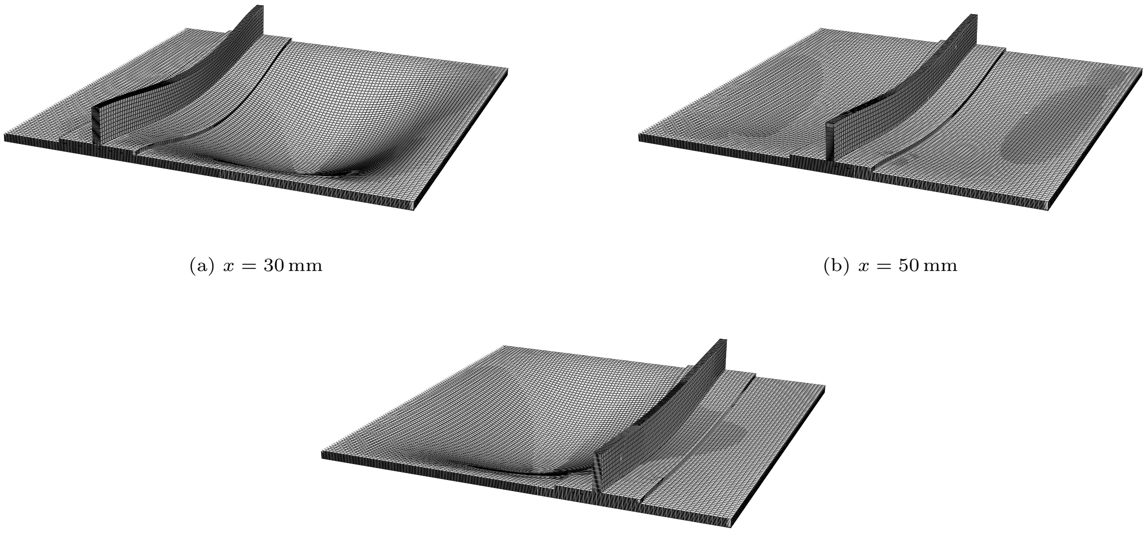


Figure 7: Maximum skin deflection vs. stiffener position predictions using solid elements in the component parts



(a) $x = 30\,\mathrm{mm}$

(b) $x = 50\,\mathrm{mm}$

(c) $x = 70\,\mathrm{mm}$

Figure 8: The status of the floating connector element allows us to represent a stiffener at any desired position for the model composed of solid elements. All sub-figures were obtained from one single simulation. (Note that, for visualisation purposes, the stiffener in these images is a dummy stiffener with negligible stiffness and with MPC equations used to deform it as the real stiffener and to shift it to the correct value of x.)

3.3. Shell to shell connection

For the model with shell elements, we used 4-noded shell elements (S4) with the dimensions of 1 x 1 mm. Again we verified that this mesh size produces mesh-converged results. The maximum skin deflection for each stiffener position is shown in Figure 9.

For the simulation with the floating connector element, Figure 10 shows the evolution of the mesh as the status of the floating connector element represents a connection between the stiffener and the skin at different positions x. From Figures 9 and 10, we observe that the floating connector element correctly applies the connection between components composed of shell elements as well.

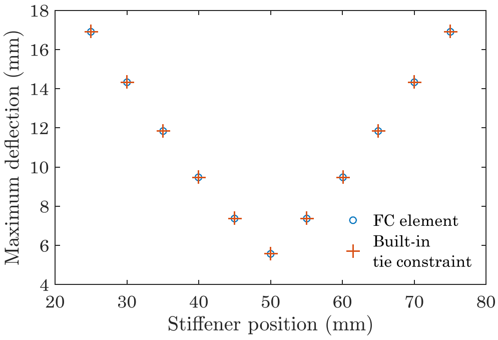
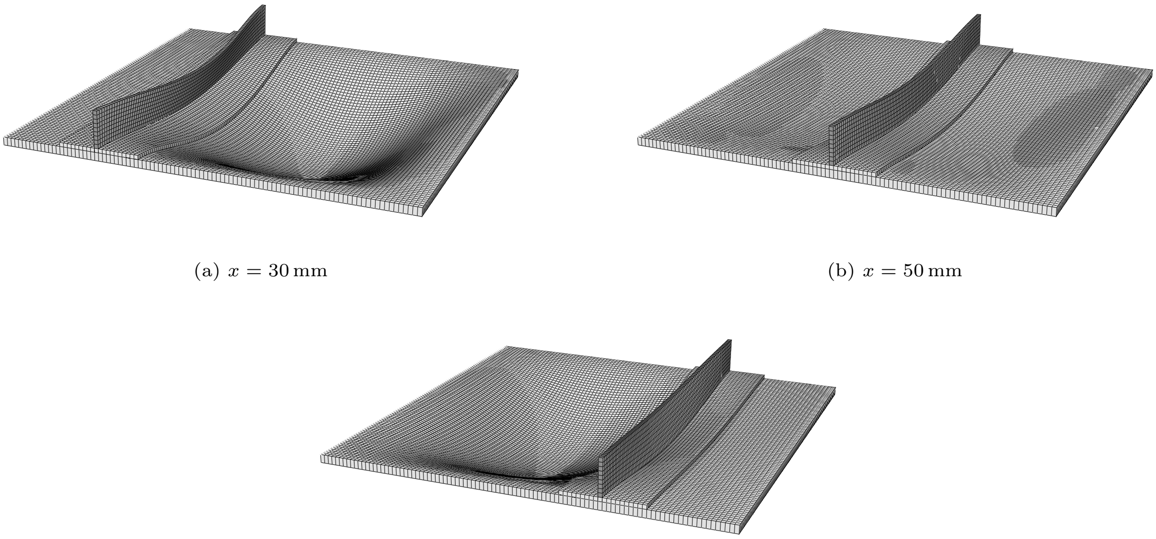


Figure 9: Maximum skin deflection vs. stiffener position predictions using shell elements in the component parts



(a) $x = 30\,\mathrm{mm}$

(b) $x = 50\,\mathrm{mm}$

(c) $x = 70\,\mathrm{mm}$

Figure 10: The status of the floating connector element allows us to represent a stiffener at any desired position for the model composed of shell elements. All sub-figures were obtained from one single simulation. (Note that, shell thickness's were rendered to better show the contact). For visualisation purposes, the stiffener in these images is a dummy stiffener with negligible stiffness and with MPC equations used to deform it as the real stiffener and to shift it to the correct value of x, and the shell thicknesses were rendered to better show the contact.)

4. Optimization

4.1. Introduction

In order to demonstrate the adaptive connection capability of the FC element, different optimizations algorithms can be integrated to the element formulation to attain the optimum stiffener location along the skin based on a chosen objective function (minimum skin deflection in this case). We implemented two of these algorithms, namely gradient descent and genetic algorithm, and the respective descriptions are provided in Sections 4.2 and 4.3.

4.2. Gradient descent

In gradient descent optimization, the gradient of the function intended to be optimized (cost function J) with respect to a system parameter θ is used to direct the solution to a local minimum point. The generic formulation can be written as

$$\theta_k = \theta_{k-1} - \eta \cdot \nabla_\theta J(\theta_{k-1}), \tag{13}$$

where k refers to the step number, η is the learning rate and $\nabla_\theta J(\theta_{k-1})$ is the gradient of J with respect to θ evaluated at step $k-1$. For the specific verification case we choose, the equation becomes

$$x_k^{\text{stf}} = x_{k-1}^{\text{stf}} - \eta \nabla_{x^{\text{stf}}} J(x_{k-1}^{\text{stf}}), \text{ with} \tag{14}$$

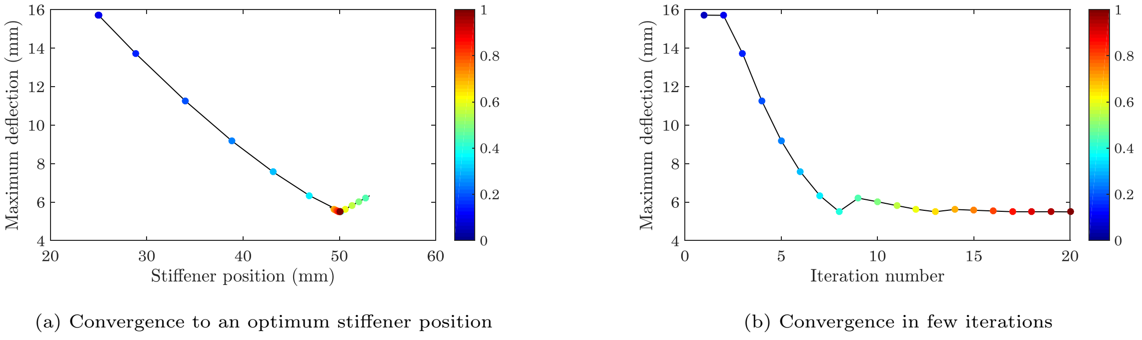
$$J(x^{\text{stf}}) = \delta^{\text{skn}}(x^{\text{stf}}), \tag{15}$$

where x^{stf} is the stiffener position and $\delta^{\text{skn}}(x^{\text{stf}})$ is the absolute value of the maximum deflection of the skin surface for a stiffener at position x^{stf}. In this algorithm, the gradient part determines the direction whereas the learning rate determines the step size for the parameter update. Considering the chosen skin-stiffener system, in order to calculate the gradient, a finite difference scheme can be used:

$$\nabla_{x^{\text{stf}}} J(x_{k-1}^{\text{stf}}) \approx \frac{J(x_{k-1}^{\text{stf}}) - J(x_{k-2}^{\text{stf}})}{x_{k-1}^{\text{stf}} - x_{k-2}^{\text{stf}}}. \tag{16}$$

At each iteration, the integrated gradient descent algorithm of the element uses the current deformed state and determines the stiffener position for the next iteration of the numerical analysis.

Using this algorithm, the maximum deflection vs. stiffener position during the various iterations of the analysis are provided in Figure 11a. In the current implementation, the learning rate is set as 10 mm and divided by 4 each time the algorithm overshoots the extrema point. This effectively means modifying the learning rate when the gradient changes its sign resulting in excellent convergence (see Figure 11b). An animation of the combined mechanical-optimisation study with this gradient descent method is given as extra material with this paper.



(a) Convergence to an optimum stiffener position

(b) Convergence in few iterations

Figure 11: The gradient descent method implemented within a FC element leads to a combined mechanical-optimisation problem which converges in few iterations (the colour of the circles represents the normalized iteration number)

4.3. Genetic algorithm

4.3.1. Introduction

The genetic algorithm implemented within the FC element (see Figure 12) is briefly explained in this section. For more details on genetic algorithms, the reader is referred to the literature [26, 27]. The genetic algorithm (see Figure 12) involves several steps: population initialization; fitness calculation; parent selection; crossover; mutation; survivor selection; and termination. Before the explaining these, some key terminology is described first in Section 4.3.2.

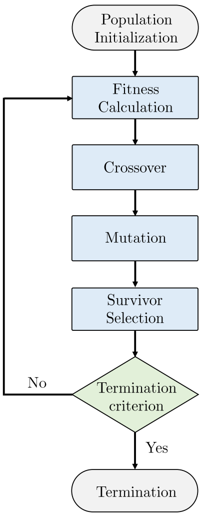


Figure 12: Genetic algorithm structure

4.3.2. Key terminology

A genetic algorithm enables finding the optimum solution, starting from a given set of individual solutions, using operations such as crossover and mutation inspired by biological evolution. The set of such possible individual solutions of the problem is referred as the population (see Figure 13).

The actual representation of the solutions of a system that corresponds to a real-life instance (e.g. stiffener positions given as real numbers) are referred to as 'phenotype' representation. In order to apply the genetic algorithm operations, the solutions however need to be represented in an encoded form which is referred to as 'genotype' representation. The encoded representation of each solution is called a chromosome, and this is composed of genes. The genes represent one element of the encoded sequence of a chromosome and the value it takes is called an allele (see Figure 13).

In order to assess the optimality of each solution, we need a fitness function which assigns a certain numerical value to the solution based on a measure of its performance. Thus, the fitness function operates on the phenotype representation to assess the optimality of the solution.

As mentioned, the main genetic operations are 'crossover' and 'mutation'. 'Crossover' generates a new generation of solutions by blending the genetic information of selected chromosomes in the previous generation whereas 'mutation' modifies the allele of the genes based on a low mutation probability.

During the genetic algorithm operations, the fitness function operates on the phenotype representation

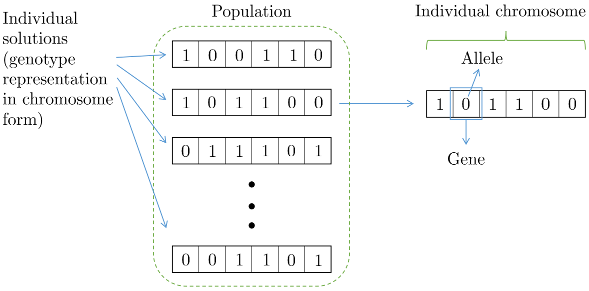


Figure 13: Genetic algorithm terminology

to assess the optimality of the solutions. Then, the phenotype representation is converted to a genotype representation to run the crossover and mutation operations. The new chromosomes that are generated in each generation then need to be decoded back to phenotype representation to calculate their fitness value. Thus, in order to convert between the various representations, suitable encoding and decoding operations are essential.

In the current implementation of the algorithm, a binary representation is used for the encoding. Considering a given interval for the stiffener position $x^{\text{stf}} \subset [x_{\text{in}}, x_{\text{fn}}]$, each of the real-valued solutions x^{stf} can be represented as a binary number with $n_{\text{bt}} \in \mathbb{N}$ digits. Thus, the overall solution space can contain up to $2^{n_{\text{bt}}}$ distinct solutions. Consider a binary representation of a solution as

$$(x^{\text{stf}})_2^{\text{bn}} = (b_{n_{\text{bt}}-1}\; b_{n_{\text{bt}}-2}\; \cdots\; b_i\; \cdots\; b_2\; b_1\; b_0), \tag{17}$$

where b_i refers to the individual digit numbers in binary form with $b_i \in \{0, 1\}$. The genetic algorithm operates on the sequences (chromosomes) $(x^{\text{stf}})_2^{\text{bn}}$ which correspond to each individual solution x^{stf}.

In order to convert real valued representation (phenotype) into chromosome form (genotype), first, each of the n_{ppl} admissible stiffener position x^{stf} can be represented as an integer in the interval $[0, 2^{n_{\text{bt}}}]$ as

$$(x^{\text{stf}})^{\text{bn}} = (2^{n_{\text{bt}}} - 1)\frac{x^{\text{stf}} - x_{\text{in}}}{x_{\text{fn}} - x_{\text{in}}}, \tag{18}$$

which can be written in the binary form $(x^{\text{stf}})_2^{\text{bn}}$ shown in Equation 3, hence defining the bits b_i.

Equally, using the binary representation $(x^{\text{stf}})_2^{\text{bn}}$, the real-valued solutions x^{stf} can be recovered using:

$$(x^{\text{stf}})^{\text{bn}} = \sum_{i=1}^{n_{\text{bt}}} b_i 2^{i-1}, \tag{19}$$

$$x^{\text{stf}} = x_{\text{in}} + (x^{\text{stf}})^{\text{bn}} \cdot \frac{x_{\text{fn}} - x_{\text{in}}}{2^{n_{\text{bt}}} - 1}. \tag{20}$$

The resolution Δx that can be achieved with n_{bt} digits for the binary number is therefore:

$$\Delta x = \frac{x_{\mathrm{fn}} - x_{\mathrm{in}}}{2^{n_{\mathrm{bt}}} - 1}. \tag{21}$$

4.3.3. Population initialization

Population initialization refers to the initial seeding that corresponds to various solutions x_j^{stf} in the search space

$$X_0^{\mathrm{stf}} = \left\{ x_1^{\mathrm{stf}},\, x_2^{\mathrm{stf}},\, \cdots, x_j,\, \cdots,\, x_{n_{\mathrm{ppl}}}^{\mathrm{stf}} \right\}, \tag{22}$$

where $n_{\mathrm{ppl}} \leq 2^{n_{\mathrm{bt}}}$ refers to the number of solutions in the population. The seeding can be realized in various ways, e.g. uniformly or randomly. In the uniform case, the initial population is distributed over the solution space uniformly while, in the random case, a random bit $b_{ij} \in 0, 1$ is assigned to the individual bit i of each individual solution j of the population with $1 \leq i \leq n_{\mathrm{bt}}$ and $1 \leq j \leq n_{\mathrm{ppl}}$. In this work, we used $n_{\mathrm{ppl}} = 10$ and $n_{\mathrm{bt}} = 8$ for both the verification in Section 4 and the application in Section 5.

4.3.4. Fitness calculation

To assess the fitness of each solution, the respective output is calculated and a fitness value is assigned to determine the probability that the solution can contribute its genetic information to the next generation of solutions. The fitness value needs to be higher for fitter individuals in the population.

Additionally, it is convenient for the forthcoming calculation of probabilities that the fitness values are all non-negative. With this in mind, let δ_j^{skn} be the absolute value of the maximum deflection of the skin for solution j (corresponding to a stiffener position x_j^{stf}). Thus, a suitable fitness function can be:

$$F_j = -\delta_j^{\mathrm{skn}} + c, \tag{23}$$

with

$$c = \max_{j \in \{1, n_{\mathrm{ppl}}\}} \left\{ \delta_j^{\mathrm{skn}} \right\}. \tag{24}$$

Alternatively, we can use Equation 23 to define the rank of each solution x_j^{stf} and then define the fitness of x_j^{stf} as

$$F_j = n_{\mathrm{ppl}} + 1 - \mathrm{rank}(x_j^{\mathrm{stf}}). \tag{25}$$

In this work, we used this latter approach throughout.

4.3.5. Parent selection

After the fitness calculation, the probability of a solution contributing to the next generation is calculated. With the fitness function being higher for fitter solutions, and always non-negative, the probability of a solution j being selected to contribute to the next generation is

$$P_j = F_j / F_{\mathrm{T}}, \tag{26}$$

with

$$F_{\mathrm{T}} = \sum_{j=1}^{n_{\mathrm{ppl}}} F_j. \tag{27}$$

This probability is then used repeatedly to select sets of two parents. The process for selecting each parent consists of the following:

(i) generating a random number $r \subset [0, F_{\mathrm{T}}]$;

(ii) starting from the first individual ($j = 1$) summing the fitness values until the summation exceeds the number r; and

(iii) the last individual solution included in the summation is chosen as a parent.

Graphically, this process can be visualised as spinning a Roulette wheel divided in n_{ppl} segments corresponding to each individual solutions, and with the length of each segment being proportional to the respective fitness (Figure 14). Finally, the parents are arranged in sets of 2, with each set being used to generate two children.

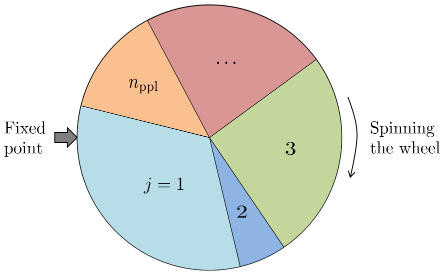
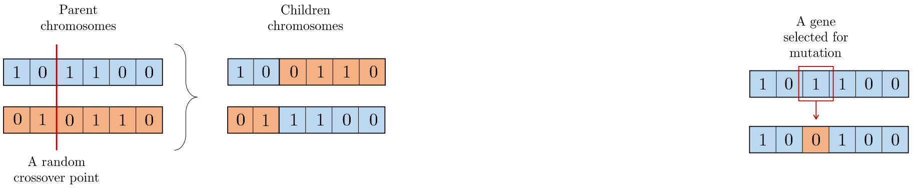


Figure 14: Roulette wheel selection

4.3.6. Crossover

'Crossover' refers to the operation whereby parent chromosomes contribute certain portion of their genetic information to the children population. In the current implementation (see Figure 15a), a one point crossover was used. In this approach, two parent chromosomes are selected and divided at a random section. Then, the divided parts up to and after the random section from the two parents are switched to preliminarily define two children chromosomes.



(a) One point crossover process

(b) Chromosome mutation

Figure 15: Genetic algorithm operations

4.3.7. Mutation

Once the children chromosomes have been preliminary defined, a mutation operation is applied to some genes, according to a pre-defined small probability P_{mt}, so as to diversify the population and explore the solution space. In the current implementation, we use bit flip mutation for the mutation operation. In this approach, if a certain gene is selected for mutation based on a low probability, its bit number (allele) is flipped (see Figure 15b). In this work, we used $P_{\mathrm{mt}} = 10\,\%$.

Additionally, for each generation, if we observe multiple numbers of exactly the same chromosome, we apply a mutation with 100 % probability on the first half of the digits of the chromosomes (i.e. for b_0 to b_3 in our case in which we have $n_{\mathrm{bt}} = 8$) to introduce diversity to the population and reduce crowding.

4.3.8. Survivor selection

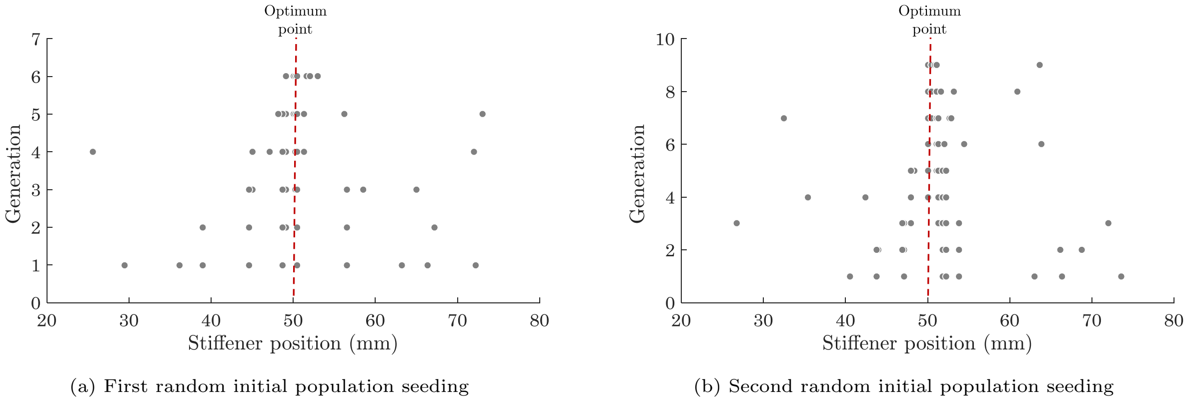
To define the next generation of solutions, we retain the top $n_{\mathrm{rm}} < n_{\mathrm{ppl}}$ solutions of the previous generation, and replace the remaining $n_{\mathrm{ch}} = n_{\mathrm{ppl}} - n_{\mathrm{rm}}$ solutions with the children solutions previously generated. In this work, we used $n_{\mathrm{ch}} = 5$.

4.3.9. Termination

The algorithm can be terminated either using a suitable clustering criterion, or when a certain number of generations n_{gen} has been created. Regarding the former, a suitable convergence criterion is to require that a certain fraction f of the solutions are closer to each other than a certain tolerance ϵ. In our case, we used $n_{\mathrm{gen}} = 100$. Additionally, for the verification in Section 4 and the application in Section 5, we used $f = 80\,\%$ and $\epsilon = 2\,\mathrm{mm}$.

4.3.10. Verification

We applied genetic algorithm optimisation to the stiffener problem defined in Section 3.1. The change of the stiffener position for each generation is provided in Figure 16. As it can be expected from a genetic algorithm, the solutions corresponding to the different members of the population start to cluster around the optimum point. An animation of the concurrent mechanical-optimization study with this genetic algorithm is given as extra material with this paper.



(a) First random initial population seeding

(b) Second random initial population seeding

Figure 16: Combined mechanical-optimisation simulation of the position of a stiffener, using FC elements and a genetic algorithm, showing clustering at the optimal position for two different random seedings of the initial population

5. Application

5.1. Introduction and problem definition

This paper introduced a novel FC element which enables coupled mechanical-optimisation finite element simulations. We now apply this novel methodology to a meaningful engineering problem (Figure 17): the positioning a runout stiffener on a plate, so as to minimise the deflection of the plate while not exceeding the critical energy release for debonding of the runout. We consider that the runout contains a manufacturing defect, i.e. an initial debonding representative of the minimum debond size that cannot be identified via inspection.

The runnout problem shown in Figure 17 is similar to the problem introduced in Section 3.1 (i.e. same material and same geometry where applicable). The only differences are that there is a runout, with the stiffener being $\ell_{\text{stf}} = 90\,\text{mm}$ long, with a runout region $\ell_{\text{ro}} = 10\,\text{mm}$ long and with a debond length $\ell_{\text{db}} = 10\,\text{mm}$. The applied pressure is $p = 210\,\text{kPa}$. To be conservative, we require that the maximum total energy release rate at any point along the debond must be lower than the mode I critical energy release rate.

5.2. Numerical model and formulation

To model the skin-stiffener system, hexahedral solid elements were used with linear shape functions, except at the runout section where solid tetrahedral elements with linear shape function are assigned.

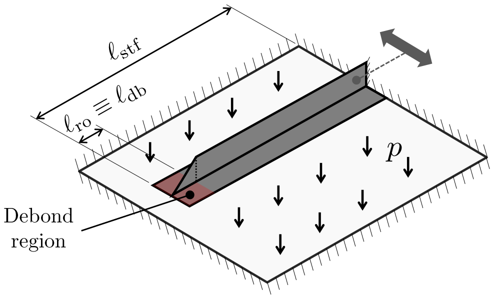


Figure 17: Application test schematic and geometry

Using the FC element formulation, the manufacturing defect (initial debonding of the runnout) can be introduced readily. This is achieved by de-activating the floating connectors (see Section 2.2) which operate on the stiffener nodes along the debonded region.

Debonds can be modelled efficiently using various techniques, including cohesive elements [28–39] and VCCT [9, 10, 25, 40, 41]. In this work, we will use the latter. In order to implement the constraint of no debond growth, the energy release rates at the debond front need to be calculated. Using the virtual crack closure technique [40] (VCCT), the energy release rates in mode I, II and III (G_{I}, G_{II}, and G_{III}, respectively) at a certain node along the debond front can be given as

$$\begin{aligned} G_{\text{I}} &= -\frac{1}{2\ell_a\ell_b} F_n [\![q_n^{\text{W}}]\!], && (28)\\ G_{\text{II}} &= -\frac{1}{2\ell_a\ell_b} F_s [\![q_s^{\text{W}}]\!], \text{ and} && (29)\\ G_{\text{III}} &= -\frac{1}{2\ell_a\ell_b} F_t [\![q_t^{\text{W}}]\!], && (30) \end{aligned}$$

where ℓ_a and ℓ_b refer to the element in-plane dimensions, $F_\bullet$ are the components of the nodal forces at the node on the debond front and $[\![q_\bullet^{\text{W}}]\!]$ represent the components of the separation in the wake of the debond front. The subscripts n, s and, t refer (using standard notation) to the local coordinate of the debond for which the VCCT forces and separations are calculated [40].

Within the FC element subroutine, the separations at the wake of the debond and the forces at the debond front can be readily calculated. From Equation 3, the separation vector $[\![\mathbf{q}_k]\!]$ for a node k can be written as

$$[\![\mathbf{q}_k]\!] = \mathbf{q}_k - \sum_i^{n^e} \mathbf{N}_i^e \mathbf{q}_i . \tag{31}$$

Equation 31 can be used directly to calculate the separations in the wake of the debond front $[\![q_\bullet^{\text{W}}]\!]$ in Equations 28, 29 and 30. The nodal forces for the corresponding node at the debond front can be calculated from the relevant components of the separation vector at the debond front ($[\![q_\bullet^{\text{DF}}]\!]$, using again Equation 31) and the Courant quadratic penalty stiffness k_{p} of the element (see Equation 10) as:

$$F_n = k_{\mathrm{p}} \, [\![q_n^{\mathrm{DF}}]\!], \tag{32}$$

$$F_s = k_{\mathrm{p}} \, [\![q_s^{\mathrm{DF}}]\!], \text{ and} \tag{33}$$

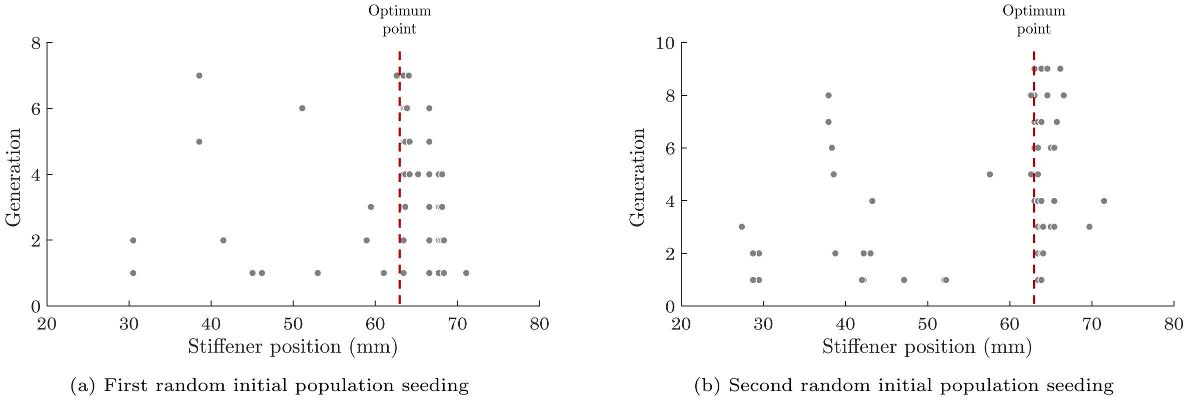
$$F_t = k_{\mathrm{p}} \, [\![q_t^{\mathrm{DF}}]\!]. \tag{34}$$

According to the problem definition in Section 5.1, the (conservative) constraint for the validation case is then $G_{\mathrm{T}} < G_{\mathrm{Ic}}$ where G_{T} is the total energy release rate along the debond front given as $G_{\mathrm{T}} = G_{\mathrm{I}} + G_{\mathrm{II}} + G_{\mathrm{III}}$. This constraint needs to be evaluated at each node along the debond front.

We show the solution to the constrained optimization problem using the genetic algorithm route only, as the solution using gradient descent is equivalent. In the genetic algorithm, the solutions which do not verify the constraint are given zero fitness.

5.3. Results

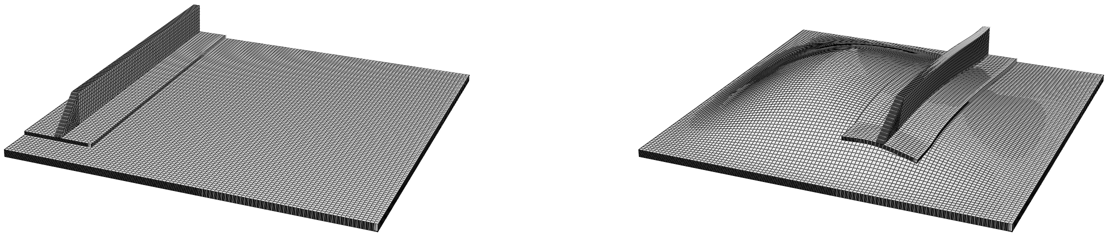
The evolution of the stiffener position over various generations is provided in Figure 18. The initial geometry and final deformation of the skin-stiffener system is also given in Figure 19. An animation of the combined mechanical-optimization study for this runout problem is given as extra material with this paper. The results show that the optimum position for the stiffener is at $x^{\mathrm{stf}} \approx 63\,\mathrm{mm}$. Note that the problem is not exactly symmetric due to the laminated nature of the components (e.g. the 45° plies are not at the same hight in the laminate thickness as the $-45°$ plies); as a result, while there is a local optimum on the left of Figure 18, the absolute optimum is on the right. As the system is not isotropic, equilibrium occurs at a non-symmetric stiffener position where the clustering happens.



(a) First random initial population seeding

(b) Second random initial population seeding

Figure 18: Combined mechanical-optimisation simulation of the position of a stiffener runout, using FC elements and a genetic algorithm, showing clustering at the optimal position for two different random seedings of the initial population. The simulation solves the mechanical problem and optimises for minimum deflection, with a constraint on maximum energy release rate.



(a) Initial status of the skin-stiffener system

(b) Final status for the skin-stiffener system where the stiffener is positioned at the optimal position $x^{\text{stf}} \approx 63\,\text{mm}$

Figure 19: The stiffener position evolves naturally to the optimal position during the constrained combined mechanical-optimisation analysis

6. Discussion

The verification results (Figures 7 and 9) show the newly proposed FC element accurately captures the connection between two generic structural components. This provides a new route for adaptively connecting different components within a finite element simulation.

When used to sweep the design domain, as shown in Section 3, the FC simulation route can simulate numerous configurations without needing to create separate finite element analyses, without modifying the overall geometry of the model, without re-meshing the components, and in fact without even needing to recompute the individual stiffness matrices of the individual components. In comparison, sweeping the design domain using manual or scripted model generation would have none of these advantages.

The potential reduction in the overall simulation time depends on the number of configurations n_{cf} that need to be simulated. The methodology requires only one model generation step at the beginning of the simulation.

Moreover, it is important to emphazise that the model provides a versatile simulation platform where different levels can be configured and integrated on-the-fly inside one finite element analysis. This enables the user to investigate the levels in a more effective manner without switching between different FE models and software.

The optimization results (Sections 4 and 5 and additional material) show that the FC element is able to integrate different optimization algorithms into its formulation. When a gradient descent algorithm is activated within the FC element, the simulation successfully reaches the desired optimum (Figure 11 and extra material). When a genetic algorithm is activated within the FC element, the simulation converges to a cluster around the desired optimal (Figure 16 and extra material). When the optimization problem includes constraints, the simulation converges to a cluster on one side of the desired optimal (as expected because of the constraint).

The application in Section 5 demonstrates three further points. Firstly, it demonstrates that the FC element approach can also deal with constrained optimisation problems in relevant engineering applications. Secondly, it demonstrates that the FC element can readily be used to perform fracture mechanics calculations, in this case using VCCT (although modelling decohesion using a cohesive element approach would be possible as well). Thirdly, it demonstrates that, in addition to being used for configurational design problems, FC elements can also readily be used to analyse the effect of defects in engineering structures. In this case, we analysed the effect of a debond of a fixed size; in general, other manufacturing defects could be considered and the design variables could instead search for maximum allowable defects.

7. Conclusions

We formulated an original Floating Connector (FC) element that enables adaptive positioning of lower-level models (such as components) within higher level-models (such as large structures) in a single FE model. The conclusions from this paper are:

- this new FC element formulation can lead to a combined mechanical/optimisation problem for the design of multi-level structures (such as the positioning of stiffeners within a wingbox);
- the combined mechanical/optimisation approach does not require re-generating the geometry, remeshing or modifying the stiffness matrix of the elements in the FE model enabling significant time reduction particularly for complex models;
- FC elements can be implemented as user-elements in standard FE software packages with a user-element interface;
- FC elements can integrate a variety of optimisation algorithms, including gradient descent and genetic algorithms;
- FC elements allow naturally the incorporation of manufacturing defects (or other pre-existing damage) in the connection between the different hierarchical levels (e.g. debondings) and optimisation taking their existence into account; and
- FC element can also potentially be used to determine maximum allowable manufacturing defects given a specific performance requirement.

The conclusions above imply that this new FC element provides a versatile simulation platform with on-the-fly multi-level configuration capability. It enables significant user time reduction, particularly for large problems and where a large number of configurations need to be considered. The methodology therefore opens new avenues for more effective structural design with significantly less user input.

Acknowledgement

The first author greatly acknowledges the scholarship from The Scientific and Technological Research Council of Turkey (TUBITAK) and British Council Turkey in the framework of the programmes BIDEB-2213 and Newton-Katip Celebi Fund. The third author is grateful for the funding form EPSRC under grant EP/M002500/1.

Data availability

The raw/processed data required to reproduce these findings cannot be shared at this time as the data also forms part of an ongoing study.

[1] J. Reinoso, A. Blázquez, A. Estefani, F. París, J. Cañas, E. Arévalo, F. Cruz, Experimental and three-dimensional global-local finite element analysis of a composite component including degradation process at the interfaces, Composites Part B 43 (2012) 1929–1942.

[2] J. Reinoso, A. Blázquez, A. Estefani, F. París, J. Canas, A composite runout specimen subjected to tension–compression loading conditions: Experimental and global–local finite element analysis, Compos. Struct. 101 (2013) 274–289.

[3] Y. Sato, T. Okabe, R. Higuchi, K. Yoshioka, Multiscale approach to predict crack initiation in unidirectional off-axis laminates, Adv. Compos. Mater 23 (2014) 461–475.

[4] S. Hühne, J. Reinoso, E. Jansen, R. Rolfes, A two-way loose coupling procedure for investigating the buckling and damage behaviour of stiffened composite panels, Compos. Struct. 136 (2016) 513–525.

[5] M. Akterskaia, E. Jansen, S. Hühne, R. Rolfes, Efficient progressive failure analysis of multi-stringer stiffened composite panels through a two-way loose coupling global-local approach, Compos. Struct. 183 (2018) 137–145.

[6] M. Akterskaia, E. Jansen, S. R. Hallett, P. Weaver, R. Rolfes, Analysis of skin-stringer debonding in composite panels through a two-way global-local method, Compos. Struct. 202 (2018) 1280–1294.

[7] S. Guinard, R. Bouclier, M. Toniolli, J.-C. Passieux, Multiscale analysis of complex aeronautical structures using robust non-intrusive coupling, Advanced Modeling and Simulation in Engineering Sciences 5 (2018) 1.

[8] R. McCune, C. Armstrong, D. Robinson, Mixed-dimensional coupling in finite element models, Int. J. Numer. Methods Eng. 49 (2000) 725–750.

[9] R. Krueger, J. Ratcliffe, P. Minguet, Panel stiffener debonding analysis using a shell/3d modeling technique, Compos. Sci. Technol. 69 (2009) 2352–2362.

[10] R. Krueger, T. O'Brien, A shell/3d modeling technique for the analysis of delaminated composite laminates, Composites Part A 32 (2001) 25–44.

[11] H. B. Dhia, G. Rateau, The arlequin method as a flexible engineering design tool, Int. J. Numer. Methods Eng. 62 (2005) 1442–1462.

[12] L. Gigliotti, S. Pinho, Multiple length/time-scale simulation of localized damage in composite structures using a mesh superposition technique, Compos. Struct. 121 (2015) 395–405.

[13] P. R. Budarapu, J. Reinoso, M. Paggi, Concurrently coupled solid shell-based adaptive multiscale method for fracture, Comput. Methods Appl. Mech. Eng. 319 (2017) 338–365.

[14] E. Kocaman, B. Chen, S. Pinho, A polymorphic element formulation towards multiscale modelling of composite structures, Comput. Methods Appl. Mech. Eng. 346 (2019) 359–387.

[15] R. Degenhardt, A. Kling, K. Rohwer, A. Orifici, R. Thomson, Design and analysis of stiffened composite panels including post-buckling and collapse, Computers & Structures 86 (2008) 919–929.

[16] R. Zimmermann, H. Klein, A. Kling, Buckling and postbuckling of stringer stiffened fibre composite curved panels–tests and computations, Compos. Struct. 73 (2006) 150–161.

[17] R. Yondo, E. Andrés, E. Valero, A review on design of experiments and surrogate models in aircraft real-time and many-query aerodynamic analyses, Prog. Aerosp. Sci. 96 (2018) 23–61.

[18] R. Rikards, H. Abramovich, K. Kalnins, J. Auzins, Surrogate modeling in design optimization of stiffened composite shells, Compos. Struct. 73 (2006) 244–251.

[19] P. Badalló, D. Trias, E. Lindgaard, Damage tolerance optimization of composite stringer run-out under tensile load, Compos. Struct. 133 (2015) 98–104.

[20] N. V. Queipo, R. T. Haftka, W. Shyy, T. Goel, R. Vaidyanathan, P. K. Tucker, Surrogate-based analysis and optimization, Prog. Aerosp. Sci. 41 (2005) 1–28.

[21] L. Lamberti, S. Venkataraman, R. T. Haftka, T. F. Johnson, Preliminary design optimization of stiffened panels using approximate analysis models, Int. J. Numer. Methods Eng. 57 (2003) 1351–1380.

[22] V. M. F. Correia, M. A. A. Gomes, A. Suleman, C. M. M. Soares, C. A. M. Soares, Modelling and design of adaptive composite structures, Comput. Methods Appl. Mech. Eng. 185 (2000) 325–346.

[23] D. Systèmes, 5.7 documentation (isight), Providence, RI: Dassault Systèmes (2012).

[24] D. Systèmes, Abaqus 6.14 documentation, Providence, RI: Dassault Systèmes (2014).

[25] R. Krueger, Development and application of benchmark examples for mixed-mode i/ii quasi-static delamination propagation predictions, NASA/CR-2012-217562, NASA, 2012 (2012).

[26] M. Mitchell, An introduction to genetic algorithms, MIT press, 1998.

[27] K. Deb, An introduction to genetic algorithms, Sadhana 24 (1999) 293–315.

[28] L. F. Kawashita, S. R. Hallett, A crack tip tracking algorithm for cohesive interface element analysis of fatigue delamination propagation in composite materials, Int. J. Solids Struct. 49 (2012) 2898–2913.

[29] L. F. Kawashita, A. Bedos, S. R. Hallett, Modelling mesh independent transverse cracks in laminated composites with a simplified cohesive segment method, Computers, Materials, & Continua 32 (2012) 133–158.

[30] F. Van der Meer, L. Sluys, A phantom node formulation with mixed mode cohesive law for splitting in laminates, Int. J. Fract. 158 (2009) 107.

[31] F. Van Der Meer, L. Sluys, The thick level set method: sliding deformations and damage initiation, Comput. Methods Appl. Mech. Eng. 285 (2015) 64–82.

[32] A. Ahmed, F. Van der Meer, L. Sluys, A geometrically nonlinear discontinuous solid-like shell element (dsls) for thin shell structures, Comput. Methods Appl. Mech. Eng. 201 (2012) 191–207.

[33] S. Pinho, L. Iannucci, P. Robinson, Formulation and implementation of decohesion elements in an explicit finite element code, Composites Part A 37 (2006) 778–789.

[34] P. P. Camanho, C. Davila, S. Pinho, Fracture analysis of composite co-cured structural joints using decohesion elements, Fatigue & fracture of engineering materials & structures 27 (2004) 745–757.

[35] P. Areias, J. Song, T. Belytschko, Analysis of fracture in thin shells by overlapping paired elements, Comput. Methods Appl. Mech. Eng. 195 (2006) 5343–5360.

[36] B. Chen, T. Tay, S. Pinho, V. Tan, Modelling the tensile failure of composites with the floating node method, Comput. Methods Appl. Mech. Eng. 308 (2016) 414–442.

[37] E. Barbieri, M. Meo, A meshfree penalty-based approach to delamination in composites, Composites Science and Technology 69 (2009) 2169–2177.

[38] E. V. Iarve, M. R. Gurvich, D. H. Mollenhauer, C. A. Rose, C. G. Dávila, Mesh-independent matrix cracking and delamination modeling in laminated composites, International journal for numerical methods in engineering 88 (2011) 749–773.

[39] K. Hoos, E. V. Iarve, M. Braginsky, E. Zhou, D. H. Mollenhauer, Static strength prediction in laminated composites by using discrete damage modeling, Journal of Composite Materials 51 (2017) 1473–1492.

[40] R. Krueger, Virtual crack closure technique: history, approach, and applications, Appl. Mech. Rev. 57 (2004) 109–143.

[41] A. Riccio, M. Gigliotti, A novel numerical delamination growth initiation approach for the preliminary design of damage tolerant composite structures, Journal of composite materials 41 (2007) 1939–1960.