An Integrated Design Tool for Tow-Steered Laminates of Composites in Abaqus and MSC.Patran/Nastran

Xin Liu* and Bangde Liu† and Twinkle Kothari‡
University of Texas at Arlington, Arlington, TX, 76019

Su Tian§ and Yufei Long¶
Purdue University, West Lafayette, IN 47906

Frank Leone‖
NASA Langley Research Center, Hampton, VA 23681

Wenbin Yu**
Purdue University, West Lafayette, IN 47906

Tow-steered composites can be tailored for optimal mechanical performance of lightweight structures. However, there are no commercial-grade design tools for tow-steered composite structures, which hinders the design innovation of tow-steered composites in realistic structures. The novelty of this paper is to develop an integrated design framework along with the development of graphical user interface (GUI) plug-ins in commercial finite element (FE) software Abaqus†† and MSC.Patran/Nastran††. The GUI plug-ins take all the design setups and communicate with external codes for the material modeling and optimization, and hence provide a unified design environment within the FE codes. The mechanics of structure genome (MSG) plate model computes shell element properties based on user-defined fiber paths and layup, which are defined via the GUI plug-ins. The optimization is performed by an open-source code, Dakota††, from Sandia National Laboratories (Sandia), which also coordinates the structural analysis, material modeling, and optimization in design iterations. Two examples are presented to demonstrate the user-friendliness and versatility of the developed GUI plug-ins. The developed tools will ease the design process and facilitate the application of tow-steered composites in realistic aerospace structures.

I. Introduction

THERE is an urgent need for innovative material concepts for exploration vehicles, space habitats, and other space hardware [1–3]. Tow-steered composites are an innovative lightweight concept, and preliminary studies have shown significant potential for further mass reduction and performance improvement compared to unidirectional fiber-reinforced composites (UDFRCs) [4, 5]. Such improvements are mainly contributed from tailoring fiber orientations for optimal load paths. However, for tow-steered composites, the capabilities of existing design tools are lagging behind evolving manufacturing techniques.

Currently, there are no commercially available design tools for tow-steered composites. Existing design of tow-steered composites often uses a glue language (e.g., Python) to connect the design variables, finite element (FE)-based structural analysis, and optimizer, but there are several shortcomings. First, it requires extra programming skills which require additional efforts by engineers, especially when adding new design variables. Second, engineers are more comfortable carrying out analysis within commercial FE tools (e.g., MSC.Nastran†† and Abaqus††), but there are no built-in graphical user interfaces (GUI) in these FE tools for the design of tow-steered composites. In addition, shell elements are often

---

[*] Assistant Professor, Department of Industrial, Manufacturing, and Systems Engineering, AIAA member.

[†] Graduate Assistant, Department of Industrial, Manufacturing, and Systems Engineering

[‡] Graduate Assistant, Department of Aerospace Engineering

[§] Graduate Assistant, School of Aeronautics and Astronautics.

[¶] Graduate Assistant, School of Aeronautics and Astronautics.

[‖] Research Aerospace Engineer, Durability, Damage Tolerance, and Reliability Branch.

[**] Professor, School of Aeronautics and Astronautics, AIAA Associate Fellow.

[††] This is not an endorsement by the National Aeronautics and Space Administration (NASA)

employed in the structural analysis to reduce the computational cost. The efficiency benefit of using shell elements becomes significant when performing design optimization, which requires to keep iterating structural analysis to find the optimized design variables. The properties of shell elements are often defined using ABD matrix from the classical lamination theory (CLT) and its refinements [6]. Some new theories have also been developed with assumptions of more complex displacement/stress/strain distributions through the thickness [7, 8]. Such assumptions inevitably lose accuracy for predicting the structural behavior and also limit the further development of the theory. Furthermore, some newly developed theories do not have the effective properties in terms of the ABD matrix. To include the non-standard shell element properties, special-purpose codes must be developed for these theories [8–10]. Recently, mechanics of structure genome (MSG)-based plate modeling has been employed to analyze various plate-like composite structures, showing significantly improved efficiency while keeping the same accuracy as direct numerical simulations [11–14]. The MSG plate model provides different functionalities depending on the quantities of interest. The MSG-based Kirchhoff-Love plate model computes shell element properties in terms of ABD matrix but can recover three-dimensional (3D) stress fields in each layer. Note that the ABD matrix computed from MSG-based Kirchhoff-Love plate model and CLT is identical. This plate model will be used in this paper, but other functionalities in the MSG plate model (e.g., capturing transverse shear stress and initial curvatures of shell structures) can be easily included for more complex problems.

In this paper, GUI plug-ins in Abaqus and MSC.Patran/Nastran are developed for the design optimization of tow-steered composites. The plug-ins are written in application programming interface (API) languages (i.e., Patran Command Language (PCL) in MSC.Patran/Nastran and Python in Abaqus), providing open architectures so that advanced users can easily modify the existing modules and add new functionalities. The GUI plug-ins ensure that the design optimization, including design setups, structural analysis, multiscale plate modeling, and optimization, can be carried out in an integrated framework within MSC.Patran/Nastran and Abaqus. The major contribution of this paper is to develop a unified design environment and plug-in tools in commercial FE tools to ease the design process of tow-steered composites for realistic aerospace structures.

II. Methodology

A. An integrated design framework

The proposed design framework is shown in Fig. 1. Users create a new or use an existing FE model and then define all the design variables (e.g., fiber paths, layup sequence, optimization method, objective function, and constraints) via the GUI plug-ins. In addition, users need to generate a FE input file that contains necessary parameters for a standard FE analysis. For MSC.Patran/Nastran, the input file is called a BDF file. For Abaqus, the input file is called an INP file. After completing the design setups, GUI plug-ins generate a master design file, which contains all the design setups and necessary parameters for multiscale plate modeling and optimization.

The GUI plug-ins call a Python script to run Dakota[††], an open-source design optimization code developed by Sandia National Laboratories [15]. Various optimization methods (e.g., gradient-based and derivative-free) are developed in Dakota to handle different types of variables (e.g., continuous, discrete, and mixed). In the proposed design framework, Dakota controls the data flow between MSG plate modeling, structural analysis, and optimization. Specifically, Dakota first calls the MSG plate model to compute shell element stiffness for each element based on the corresponding laminate layup. The element stiffness is location dependent because the fiber path varies in the in-plane directions. The computed shell element stiffness will be employed to update the the BDF/INP files. Then, Dakota calls an FE solver (e.g., MSC.Nastran or Abaqus) to perform structural analysis. For a parametric study, the GUI plug-in automatically iterates the above steps based on the user-defined design space.

For an optimization analysis, the structural responses are read by Dakota and form a predefined objective function. The objective function along with other setups (e.g., constraints) are sent to a built-in optimizer in Dakota to update the design variables. Users can select any optimizer in Dakota, depending on the requirements of the specific problem.

B. MSG-based plate modeling

MSG is a unified approach to the multiscale constitutive modeling of three-dimensional (3D) solids, plates/shells, and beams [16]. In MSG, a structure gene (SG) is defined as the smallest mathematical building block of a structure. It contains all material and geometrical information of a microstructure. For the MSG-based plate model, field measures are represented as functions of x_1 and x_2 defined over the reference surface, while x_3 is eliminated. We also use micro-coordinates y_i to describe the SG with $y_i = x_i/\varepsilon$ with ε being a small parameter. In multiscale structural modeling,

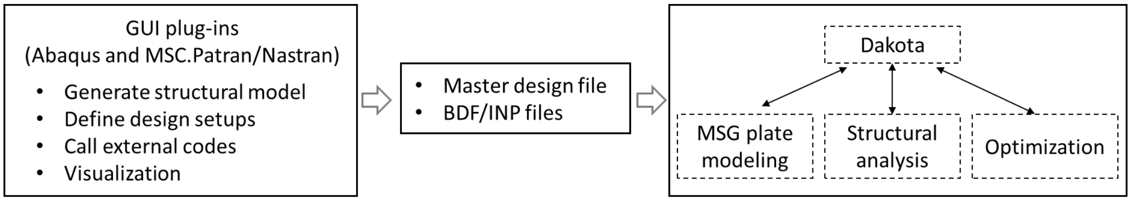


Fig. 1 Design framework of the proposed tool

a field function of the original heterogeneous structure can be generally written as a function of the macro-coordinates x_k, which remains in the macroscopic structural model, and the micro-coordinates y_j. The partial derivative of a function $f(x_k, y_j)$ can be expressed as

$$\frac{\partial f(x_k, y_j)}{\partial x_i} = \frac{\partial f(x_k, y_j)}{\partial x_i}|_{y_j=\text{const}} + \frac{1}{\varepsilon}\frac{\partial f(x_k, y_j)}{\partial y_i}|_{x_k=\text{const}} \equiv f_{,i} + \frac{1}{\varepsilon} f_{|i} \tag{1}$$

The 3D displacement field can be expressed in terms of the two-dimensional (2D) displacement variables as [13]:

$$\begin{aligned}
u_1(x_1, x_2, y_1, y_2, y_3) &= \overline{u}_1(x_1, x_2) - \varepsilon y_3 \overline{u}_{3,1}(x_1, x_2) + \varepsilon w_1(x_1, x_2, y_1, y_2, y_3) \\
u_2(x_1, x_2, y_1, y_2, y_3) &= \overline{u}_2(x_1, x_2) - \varepsilon y_3 \overline{u}_{3,2}(x_1, x_2) + \varepsilon w_2(x_1, x_2, y_1, y_2, y_3) \\
u_3(x_1, x_2, y_1, y_2, y_3) &= \overline{u}_3(x_1, x_2) + \varepsilon w_3(x_1, x_2, y_1, y_2, y_3)
\end{aligned} \tag{2}$$

where u_i and $\overline{u}_i$ denote the displacements of the original 3D heterogeneous structure and the 2D plate model respectively. w_1, w_2, and w_3 are the unknown fluctuating functions, which are used to describe the displacement field that cannot be captured by the traditional Kirchhoff-Love plate model. When solving the fluctuating functions by a numerical approach (e.g., FE), standard shape functions will be employed to approximate the functional forms. As a result, the fluctuating functions can be a linear or second-order functions within a element depending on the element order selected by users. The infinitesimal strain field in the 3D linear elasticity theory can be defined as:

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \tag{3}$$

Plug Eq. (2) into Eq. (3), and the 3D strain field can be expressed using plate strain strains $\epsilon_{\alpha\beta}$, curvatures $\kappa_{\alpha\beta}$, and fluctuating functions as:

$$\begin{aligned}
\varepsilon_{11} &= \epsilon_{11} + \varepsilon y_3 \kappa_{11} + w_{1|1} + \varepsilon w_{1,1} \\
\varepsilon_{22} &= \epsilon_{22} + \varepsilon y_3 \kappa_{22} + w_{2|2} + \varepsilon w_{2,2} \\
\varepsilon_{33} &= w_{3|3} \\
2\varepsilon_{12} &= 2\epsilon_{12} + 2\varepsilon y_3 \kappa_{12} + w_{1|2} + w_{2|1} + \varepsilon w_{1,2} + \varepsilon w_{2,1} \\
2\varepsilon_{13} &= w_{1|3} + w_{3|1} + \varepsilon w_{3,1} \\
2\varepsilon_{23} &= w_{2|3} + w_{3|2} + \varepsilon w_{3,2}
\end{aligned} \tag{4}$$

where the plate strains and curvatures are defined as:

$$\epsilon_{\alpha\beta}(x_1, x_2) = \frac{1}{2}(\overline{u}_{\alpha,\beta} + \overline{u}_{\beta,\alpha});\ \kappa_{\alpha\beta}(x_1, x_2) = -\overline{u}_{3,\alpha\beta} \tag{5}$$

The total potential energy of the 3D structure can be defined as:

$$\Pi = \frac{1}{2}\int_s U_{2D}\, ds - W \tag{6}$$

where W is the work done by external sources. U_{2D} is the 2D strain energy density defined as:

$$U_{2D} = \frac{1}{2\omega}\langle \sigma_{ij}\varepsilon_{ij} \rangle \tag{7}$$

where ω denotes the area spanning the $y_1 - y_2$ plane of the SG. The "$\langle\rangle$" denotes the integration over the SG. The fluctuating functions follow the constraints:

$$\langle w_i \rangle = 0 \tag{8}$$

By minimizing the total potential energy, the fluctuating functions w_i are solved [16]. The 2D kinetic variables called the plate stress resultants are defined as:

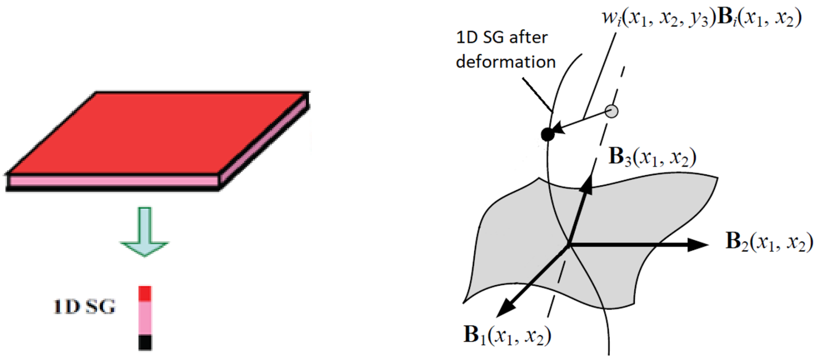
$$\begin{gathered}
\frac{\partial U_{2D}}{\partial \epsilon_{11}} = N_{11};\ \frac{\partial U_{2D}}{\partial 2\epsilon_{12}} = N_{12};\ \frac{\partial U_{2D}}{\partial \epsilon_{22}} = N_{22} \\
\frac{\partial U_{2D}}{\partial \kappa_{11}} = M_{11};\ \frac{\partial U_{2D}}{\partial 2\kappa_{12}} = M_{12};\ \frac{\partial U_{2D}}{\partial \kappa_{22}} = M_{22}
\end{gathered} \tag{9}$$

The plate constitutive equation that relates the plate stress resultants and strain measures can be obtained as:

$$\begin{Bmatrix} N_{11} \\ N_{22} \\ N_{12} \\ M_{11} \\ M_{22} \\ M_{12} \end{Bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{16} & B_{11} & B_{12} & B_{16} \\ A_{12} & A_{22} & A_{26} & B_{12} & B_{22} & B_{26} \\ A_{16} & A_{26} & A_{66} & B_{16} & B_{26} & B_{66} \\ B_{11} & B_{12} & B_{16} & D_{11} & D_{12} & D_{16} \\ B_{12} & B_{22} & B_{26} & D_{12} & D_{22} & D_{26} \\ B_{16} & B_{26} & B_{66} & D_{16} & D_{26} & D_{66} \end{bmatrix} \begin{Bmatrix} \epsilon_{11} \\ \epsilon_{22} \\ 2\epsilon_{12} \\ \kappa_{11} \\ \kappa_{22} \\ 2\kappa_{12} \end{Bmatrix} \tag{10}$$

Here, the 6×6 plate stiffness matrix is composed of the $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{D}$ matrices. Although we used the same notation of $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{D}$ matrices from CLT, the way to obtain these stiffness matrices has no relation to that which has been used to derive CLT. This stiffness matrix can be used as input to conduct the macroscopic plate analysis.

For tow-steered laminates, a one-dimensional (1D) SG is defined as shown in Fig. (2a). The SG is a line segment described by 1D coordinates along the thickness direction of the plate, consisting of several connecting sub-line segments, where each sub-line segment represents a lamina and can have its respective thickness and properties. Therefore, lamina properties and thickness are required as the input for the MSG-plate model to compute properties of shell elements (e.g., ABD matrix). The deformed shape of the initially straight SG can be described by the fluctuating functions, w_i (see Fig. (2b)).

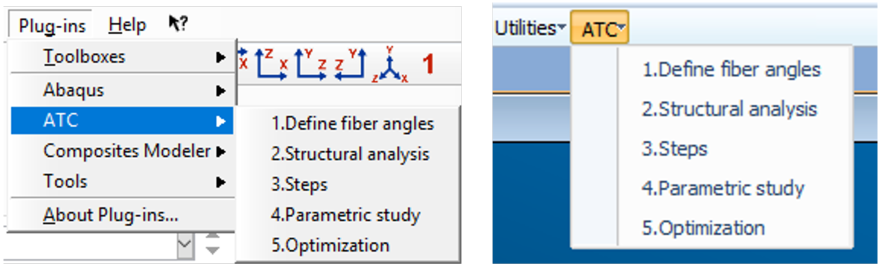


(a) 1D SG of a laminate **(b) Deformed shell**

Fig. 2 MSG plate model for tow-steered laminates

III. Development of GUI plug-ins

The basic layout of the proposed GUI plug-ins in Abaqus and MSC.Patran is shown in Fig. 3. "ATC" stands for Advanced Tailorable Composites. The "Define fiber angles" module offers a flexible way to input any fiber angle expressions with arbitrary design variables. The "Structural analysis" module allows users to perform a quick evaluation of a FE model with design variables. The "Step" module is used to define the data transformation between the MSG plate modeling, FE structural analysis, and external optimizer. The "Parametric study" module takes the user-defined design space and the partitions of the space to iterate the structural analysis with different design variables. The "Optimization" module specifies the design variables, objective function, constraints, and all the other needed parameters for a complete design optimization analysis.



(a) Abaqus (b) MSC.Patran

Fig. 3 The layout of the proposed GUI

A. Define fiber angles

Users first need to define composite layers with customized fiber angles. The angles can be defined by constants, one-line expressions, and external Python scripts. The constant angles are designed for traditional UDFRCs. The one-line expressions are designed for the fiber paths with simple mathematical functions. The external scripts can be used to define a complex fiber path that cannot be expressed with a simple function. Users can define an arbitrary fiber path with different complexities. In addition, material and layer thickness are also defined in this step. Further, users can define a failure criterion for failure analysis of the laminate, which can be used as a constraint for the design optimization.

After defining all the layers, users select the "Layup" function to define the laminate. The layup can be defined using indices or names of previous defined layers. Then, users can assign the defined laminate to a region or multiple regions in a FE model. This functionality brings two benefits. First, many complex FE models have already been created in Abaqus or MSC.Patran/Nastran. Users may only want to replace the materials with tow-steered composites in some regions to investigate the potential benefits (e.g., weight savings). In this case, users just need to use the plug-ins to define the tow-steered laminate and assign it to those regions. It only requires some minor modifications of the original FE model such as defining an element set for the regions and define a new material property for the lamina if needed. Second, FE models with multiple types of elements (e.g., solid, shell, and beam elements) can also be analyzed with the developed GUI plug-ins, because only the properties of shell elements need to be re-defined while the rest of FE model remains the same.

The data structure of defining a laminate can be summarized in Fig. 4. A laminate contains several attributes such as regions and layup. Regions are defined using the built-in functions in Abaqus and MSC.Patran/Nastran. Layup is constructed from the pre-defined lamina using the "Define fiber angles" module. A lamina has the attributes "Name", "Orientation", "Material", and "Thickness", where the "Orientation" is also an object and has its own attributes for defining fiber orientations. This data structure can be easily extended to include other design variables in the future development. For example, the material and thickness can be re-defined as an object and/or design variables which only requires a minor modification of the current code.

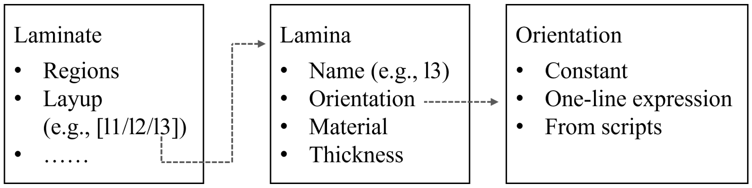


Fig. 4 The data structure for defining a tow-steered composite laminate

B. Step

The “Steps” module is designed for advanced users to design complex structures with a user-defined objective function. The “Steps” module requires users to input necessary parameters for the GUI plug-ins to communicate with external scripts and transfer data between multiscale plate modeling, structural analysis, and optimization. For example, users need to extract the data from Abaqus results file (.odb) or Nastran results file (.f06), and then use the extracted results to construct the objective function or constraints. This module provides the flexibility for users to define customized optimization problems.

C. Structural analysis

The “Structural analysis” module provides a one-time analysis. Users can use this function to have a quick evaluation of the design setups before other computationally expensive analyses. This quick evaluation will help users avoid unnecessary computational costs due to inappropriate design setups. In addition, a single analysis can be used to estimate the computing time of the FE model, and therefore provides guidance in defining the parameters (e.g., number of partitions in the design space, maximum iterations, and stopping criterion) for parametric study and optimization.

D. Parametric study

The “Parametric study” module can be used to understand the design space with a controlled computational cost. Users can perform parameter study to understand the distribution of the structural responses (e.g., displacements in a nodal set) with a set of predefined design variables. For more advanced finite element analysis (FEA), such as buckling or failure analysis, users need to develop external scripts and connect the scripts with additional parameters in the “Steps” module. In addition to a better understanding of design space, the distribution of structural responses can help users tailor the optimization parameters (e.g., reduce the range of design variables) to reduce the computational costs.

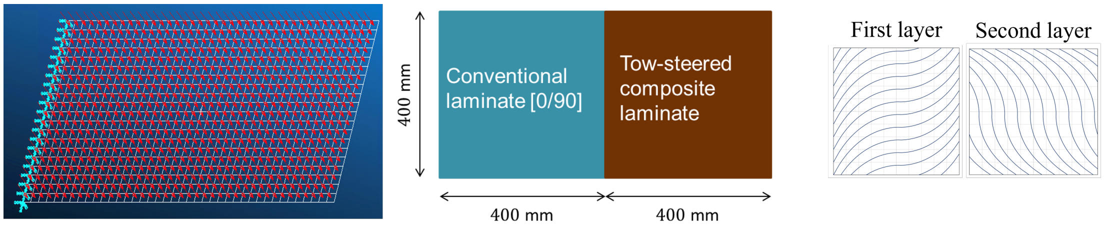
E. Optimization

Users need to specify the design variables and parameters for an external optimizer. Dakota takes all the design variables and parameters needed for an optimization analysis. All the functionalities in Dakota can be called from the GUI plug-ins. For simple optimization problems, the objective function and constraints can be defined in the optimization GUI windows. For complex optimization problems, users need to develop external scripts and connect the scripts to the optimization framework via the “Steps” module.

IV. Examples

A. Example 1: Structural analysis of a clamped plate with two regions

In this example, a clamped plate composed of two regions is used to demonstrate the GUI plug-ins in defining a simple fiber path and assigning tow-steered composites to a part of the FE model. One region is a conventional UDFRCs laminate and the other is a tow-steered composite laminate. The example is demonstrated using MSC.Patran/Nastran as shown in Fig. 5(a). Note that the same example can also be carried out using the Abaqus GUI plug-in. The plate contains two layers and is subjected to a uniform pressure 0.0001 N/mm^2 on the top surface. The dimensions of the plate are shown in Fig. 5(b). The lamina properties are given as: $E_1 = 37000$ MPa, $E_2 = E_3 = 9000$ MPa, $G_{12} = G_{13} = G_{23} = 4000$ MPa, and $\nu_{12} = \nu_{13} = \nu_{23} = 0.28$.



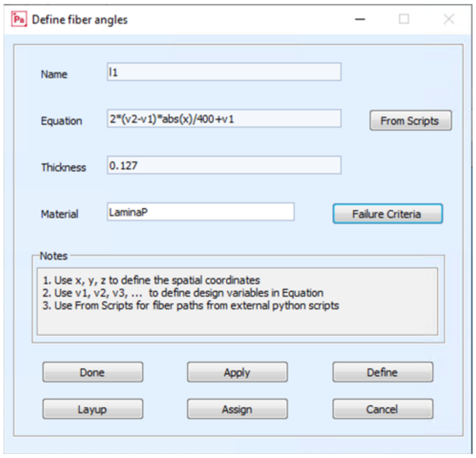
(a) Boundary conditions (b) Dimensions of the plate (c) Fiber paths in the laminate

Fig. 5 Problem statement

The fiber angle expressions of the tow-steered composite laminate are given in the following equations:

- First layer: $\theta(x) = 2(\theta_1 - \theta_0)|\frac{x}{400}| + \theta_0$
- Second layer: $\theta(y) = 2(\theta_1 - \theta_0)|\frac{y}{400}| + \theta_0$

The θ_1 and θ_0 are the design parameters. For structural analysis, we assume that $\theta_1 = 60$ and $\theta_0 = 0$. The fiber paths in the laminate are shown in Fig. 5(c). In the "Define fiber angles" module, the fiber angles are input via the one-line expression function as shown in Fig. 6. The corresponding layer thickness and material are also defined. By default, the symbols v$_i$ are recognized as the design variables by the plug-in and can be assigned with a value from the “Define” function. With the defined layers, users can construct the layup using the layer index, i.e., [1/2], as shown in Fig. 7.



Name: l1

Equation: 2*(v2-v1)*abs(x)/400+v1

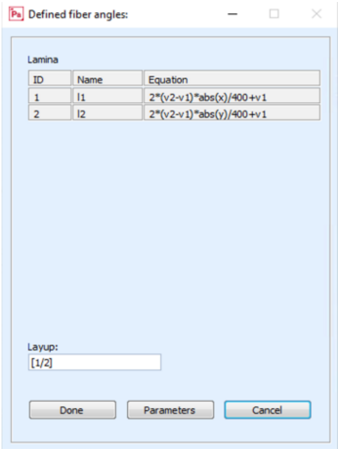
Thickness: 0.127

Material: LaminaP

Notes

1. Use x, y, z to define the spatial coordinates
2. Use v1, v2, v3, ... to define design variables in Equation
3. Use From Scripts for fiber paths from external python scripts

Fig. 6 Define fiber angle in the first layer



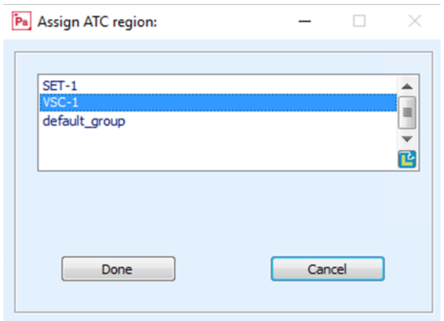
Lamina

ID	Name	Equation
1	l1	2*(v2-v1)*abs(x)/400+v1
2	l2	2*(v2-v1)*abs(y)/400+v1

Layup: [1/2]

Fig. 7 Define the layup

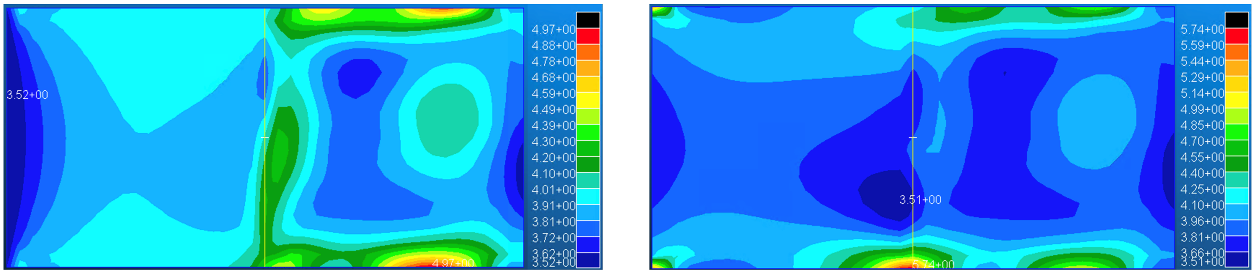
The defined laminate is then assigned to a region of the FE model. In MSC.Patran/Nastran, the region can be defined using "Group" as shown in Fig. 8. In Abaqus, the region can be defined using "Set". The plug-in code will only replace the properties of the shell elements in the selected region. If no region is specified, the code will assign the tow-steered composites laminate to the entire FE model.



SET-1
VSC-1
default_group

Fig. 8 Assign the tow-steered laminate

After defining the laminates, users can directly use the “Structural analysis” module to perform FE analysis. All other model definitions (e.g., boundary and loading conditions) are completed using the same steps as a normal FEA. In this example, the "Steps" module is skipped because the structural analysis does not need to transfer data to the external optimizer. The “Structural analysis” module will automatically call the FE solver (i.e., MSC.Nastran) for the analysis. After the analysis is complete, users can read the result file back to MSC.Patran. The von-Mises stress contour plots of this example are given in Fig. 9. It is clear that the stress distributions are very different in the two regions. Note that this example is just for a demonstration of the GUI plug-ins. For a realistic structure, the connection between conventional laminate and tow-steered composites laminate should also be considered.



Layer 1 Layer 2

Fig. 9 von-Mises stress contour plots (MPa)

B. Example 2: Optimization of tow-steered composites in a cylindrical structure subjected to buckling load

A pressurized cylindrical structure is employed which contains 24 layers and 15 design variables. This structure is taken from the work of Blom et al. [17]. The goal is to maximize the buckling moment by optimizing fiber paths while respecting a maximum fiber curvature constraint. The structural configuration can be found in [17]. The layer thickness is 0.0072 in. AS4/8773 material is used with the properties $E_1 = 18.83 \times 10^6$ psi, $E_2 = 1.317 \times 10^6$ psi, $G_{12} = 7.672 \times 10^6$ psi, and $\nu_{12} = 0.32$. In our FE model, two reference points are defined at two ends respectively. The nodes at two ends of the cylinder are coupled to the corresponding reference points. All degrees of freedoms of the reference points are constrained except the rotation along x-axis. The bending moment along x-axis is applied at the both reference points. A buckling analysis is performed in this example. The mathematical definition of this optimization problem can be summarized as:

Maximize: M_{cr}

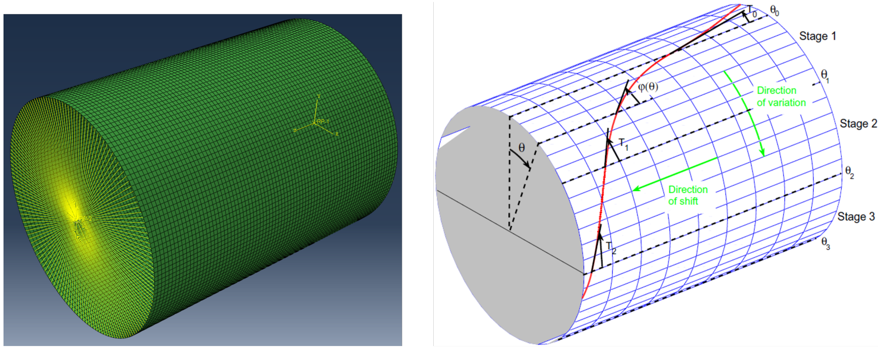
Subject to: $|\kappa| - 0.05 \leq 0$

This optimization problem is a simplified version from the original problem in [17]. In this example, the critical buckling moment M_{cr} is maximized. κ is the in-plane curvature for a steered fiber path within a given ply, which is constrained to be smaller than 0.05 in^{-1}. The κ can be computed as

$$\kappa = |\frac{\cos\phi_{i+1} - \cos\phi_i}{R\,(\theta_{i+1} - \theta_i)}| \tag{11}$$

The FE model and the fiber path are shown in Fig. 10. The fiber angle ϕ is defined as a function of θ:

$$\cos\phi(\theta) = \cos\phi_i + (\cos\phi_{i+1} - \cos\phi_i)\frac{\theta - \theta_i}{\theta_{i+1} - \theta_i} \tag{12}$$

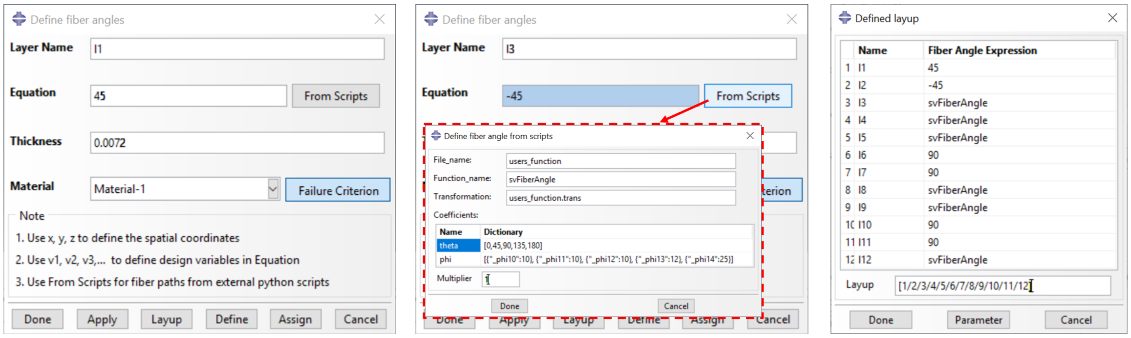


(a) Abaqus model (b) Fiber path

Fig. 10 FE model and the fiber path of the cylindrical structure [17]

Before the optimization analysis, a baseline model is constructed to verify the FE results using the developed GUI plug-in. The design layup of the baseline model is $[\pm45/0_2/\pm45/0_2/90/\pm45/90]_s$ from [17]. The buckling load is

5369 in-kips and 5851 in-kips from the GUI plug-in and the reference, respectively. The difference may come from the boundary conditions, which are not detailed in the reference. For the tow-steered composite laminate, the layup of the structure is $[\pm 45/\pm\phi_1(\theta)/0/90/\pm\phi_3(\theta)/0/90/\pm\phi_5(\theta)]_s$. The angles ϕ_i are the design variables and θ contains the coefficients determined by users. The angles ϕ can be considered as an interpolation of coefficients θ, and therefore cannot be defined using a one-line expression. In this example, there are 5 design variables in the expression of ϕ_1, ϕ_3, anf ϕ_5, respectively. Therefore, there are 15 design variables. An external script is used to define the complex fiber angle expression. In the "Define fiber angles" module as shown in Fig. 11(a), the lamina with UDFRCs is defined by a constant value. The complex fiber expression is defined by a Python script (see Fig. 11(b)). The "From script" function enables users to define arbitrary fiber angles. The GUI plug-in will compute the corresponding location-dependent shell properties. After all the layers are defined, users use the "Layup" function to define the laminate (see Fig. 11(c)).



Layer Name: l1
Equation: 45
Thickness: 0.0072
Material: Material-1

Note
1. Use x, y, z to define the spatial coordinates
2. Use v1, v2, v3,... to define design variables in Equation
3. Use From Scripts for fiber paths from external python scripts

Layer Name: l3
Equation: -45

File_name: users_function
Function_name: svFiberAngle
Transformation: users_function.trans
Coefficients:

Name	Dictionary
theta	[0,45,90,135,180]
phi	[{"_phi10":10}, {"_phi11":10}, {"_phi12":10}, {"_phi13":12}, {"_phi14":25}]

Multiplier: 1

	Name	Fiber Angle Expression
1	l1	45
2	l2	-45
3	l3	svFiberAngle
4	l4	svFiberAngle
5	l5	svFiberAngle
6	l6	90
7	l7	90
8	l8	svFiberAngle
9	l9	svFiberAngle
10	l10	90
11	l11	90
12	l12	svFiberAngle

Layup: [1/2/3/4/5/6/7/8/9/10/11/12]

(a) (b) (c)

Fig. 11 Define different fiber angles from the GUI window: (a) Define a constant angle, (b) Define an angle from a script, and (c) Define the layup.

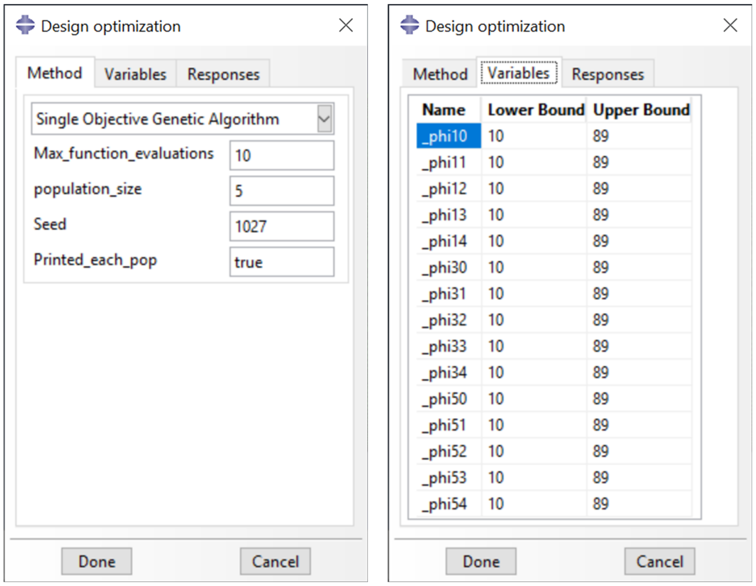
Since no regions are specified, the GUI plug-in automatically assigns the laminate to the entire FE model. After that, users need to define parameters in the "Steps" module. Since this optimization problem is to maximize the buckling load, a Python script is needed to extract the buckling load from the .odb file. In addition, the fiber curvatures will also be computed using Eq. (11). After defining all the necessary steps, users select the "Optimization" module to complete the design setup. Users need to select the optimization method and the corresponding parameters as shown in Fig. 12. The meanings of these parameters can be found from Dakota user manuals [15]. In addition, users need to specify the design variables and the bounds of each variable.

The optimization is carried out after defining all the design setups. The detailed results from each iteration are stored in the working directory. The final summary is written in a text file which contains the design variables and the response for each iteration. The screenshot of results from iteration 1 and the final summary are shown in Fig. 13. Although the example is performed using the Abaqus GUI, the same example can be performed in the MSC.Patran/Nastran GUI.

V. Conclusion

The GUI plug-ins of Abaqus and MSC.Patran/Nastan were developed for the design optimization of tow-steered composites. All the design setups can be defined via the plug-ins, which significantly reduces the additional programming efforts required in other approaches. The "Define fiber angles" module provides different ways for users to input arbitrary fiber angle expressions with different complexities. In addition, users can assign the defined laminate to any region(s) in a FE model. The plug-ins provide different analysis modules such as "Structural analysis", "Parametric study", and "Optimization". Each analysis module, with a different computational cost, serves for different purposes during the design and analysis of tow-steered composite structures. Two examples were presented to demonstrate the user-friendliness and versatility of the developed design tools.

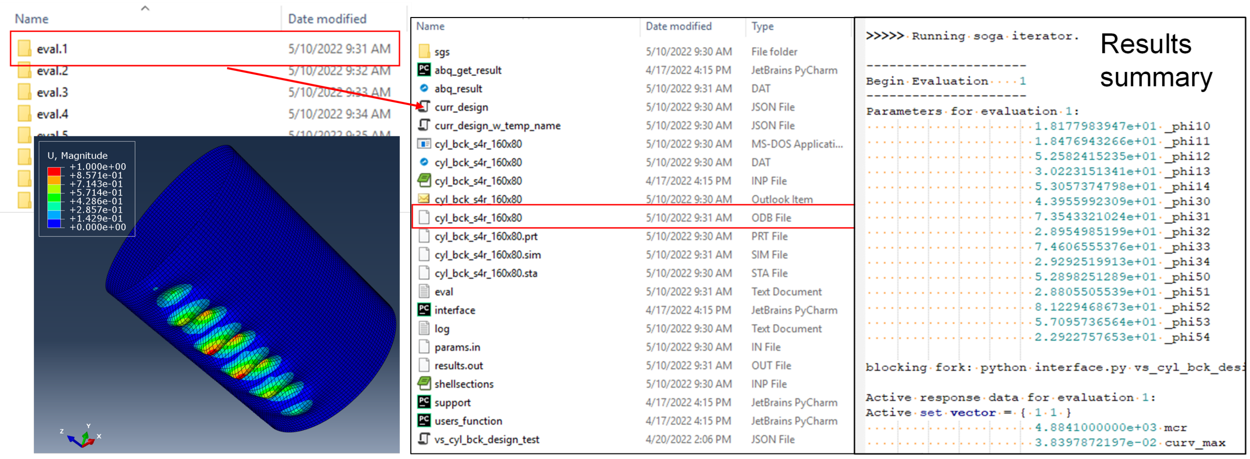
For the future development, the authors will mature the developed GUI plug-ins for the design optimization of more realistic aerospace structures. In addition, the GUI plug-ins will be extended for more design variables such as layer thickness and material selection. Further, a machine learning module will be developed to provide surrogate models, which will provide a more feasible computational cost for realistic engineering structures.



Single Objective Genetic Algorithm	
Max_function_evaluations	10
population_size	5
Seed	1027
Printed_each_pop	true

Name	Lower Bound	Upper Bound
_phi10	10	89
_phi11	10	89
_phi12	10	89
_phi13	10	89
_phi14	10	89
_phi30	10	89
_phi31	10	89
_phi32	10	89
_phi33	10	89
_phi34	10	89
_phi50	10	89
_phi51	10	89
_phi52	10	89
_phi53	10	89
_phi54	10	89

Fig. 12 Define optimization setups



Name	Date modified
eval.1	5/10/2022 9:31 AM
eval.2	5/10/2022 9:32 AM
eval.3	5/10/2022 9:33 AM
eval.4	5/10/2022 9:34 AM
eval.5	5/10/2022 9:35 AM

Name	Date modified	Type
sgs	5/10/2022 9:30 AM	File folder
abq_get_result	4/17/2022 4:15 PM	JetBrains PyCharm
abq_result	5/10/2022 9:31 AM	DAT
curr_design	5/10/2022 9:30 AM	JSON File
curr_design_w_temp_name	5/10/2022 9:30 AM	JSON File
cyl_bck_s4r_160x80	5/10/2022 9:30 AM	MS-DOS Applicati...
cyl_bck_s4r_160x80	5/10/2022 9:30 AM	DAT
cyl_bck_s4r_160x80	4/17/2022 4:15 PM	INP File
cyl_bck_s4r_160x80	5/10/2022 9:30 AM	Outlook Item
cyl_bck_s4r_160x80	5/10/2022 9:31 AM	ODB File
cyl_bck_s4r_160x80.prt	5/10/2022 9:30 AM	PRT File
cyl_bck_s4r_160x80.sim	5/10/2022 9:31 AM	SIM File
cyl_bck_s4r_160x80.sta	5/10/2022 9:30 AM	STA File
eval	5/10/2022 9:31 AM	Text Document
interface	4/17/2022 4:15 PM	JetBrains PyCharm
log	5/10/2022 9:30 AM	Text Document
params.in	5/10/2022 9:30 AM	IN File
results.out	5/10/2022 9:31 AM	OUT File
shellsections	5/10/2022 9:30 AM	INP File
support	4/17/2022 4:15 PM	JetBrains PyCharm
users_function	4/17/2022 4:15 PM	JetBrains PyCharm
vs_cyl_bck_design_test	4/20/2022 2:06 PM	JSON File

```
>>>>> Running soga iterator.

---------------------
Begin Evaluation    1
---------------------
Parameters for evaluation 1:
                      1.8177983947e+01 _phi10
                      1.8476943266e+01 _phi11
                      5.2582415235e+01 _phi12
                      3.0223151341e+01 _phi13
                      5.3057374798e+01 _phi14
                      4.3955992309e+01 _phi30
                      7.3543321024e+01 _phi31
                      2.8954985199e+01 _phi32
                      7.4606555376e+01 _phi33
                      2.9292519913e+01 _phi34
                      5.2898251289e+01 _phi50
                      2.8805505539e+01 _phi51
                      8.1229468673e+01 _phi52
                      5.7095736564e+01 _phi53
                      2.2922757653e+01 _phi54

blocking fork: python interface.py vs_cyl_bck_des

Active response data for evaluation 1:
Active set vector = { 1 1 }
                      4.8841000000e+03 mcr
                      3.8397872197e-02 curv_max
```

Fig. 13 Screenshot of the optimization results

VI. Acknowledgments

The authors would like to acknowledge the financial support provided by NASA Small Business Technology Transfer (STTR) program through grant number 80NSSC21C0405, An Efficient High-fidelity Design Tool for Advanced Tailorable Composites.

References

[1] Sleight, D. W., Segal, K. N., Guin, W., Miller, S., and McDougal, M., "Development of Composite Sandwich Bonded Longitudinal Joints for Space Launch Vehicle Structures," *AIAA Scitech 2019 Forum*, 2019, p. 0236.

[2] Wilkie, W. K., "Overview of the NASA Advanced Composite Solar Sail System (ACS3) Technology Demonstration Project," *AIAA Scitech 2021 Forum*, 2021, p. 1260.

[3] Liu, X., Furrer, D., Kosters, J., and Holmes, J., "Vision 2040: A Roadmap for Integrated, Multiscale Modeling and Simulation of Materials and Systems," Tech. rep., 2018.

[4] Guimarães, T. A., Castro, S. G., Cesnik, C. E., and Rade, D. A., "Supersonic Flutter and Buckling Optimization of Tow-steered Composite Plates," *AIAA Journal*, Vol. 57, No. 1, 2019, pp. 397–407.

[5] Singh, K., and Kapania, R. K., "Optimal Design of Tow-steered Composite Laminates with Curvilinear Stiffeners," *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, 2018, p. 2243.

[6] Barbero, E. J., *Introduction to composite materials design*, CRC press, 2010.

[7] Groh, R., and Weaver, P. M., "A Computationally Efficient 2D Model for Inherently Equilibrated 3D Stress Predictions in Heterogeneous Laminated Plates. Part II: Model Validation," *Composite Structures*, Vol. 156, 2016, pp. 186–217.

[8] Trinh, L. C., Ojo, S. O., Groh, R. M., and Weaver, P. M., "A Mixed Inverse Differential Quadrature Method for Static Analysis of Constant-and Variable-Stiffness Laminated Beams Based on Hellinger-Reissner Mixed Variational Formulation," *International Journal of Solids and Structures*, Vol. 210, 2021, pp. 66–87.

[9] Yazdani, S., and Ribeiro, P., "Geometrically Non-Linear Static Analysis of Unsymmetric Composite Plates with Curvilinear Fibres: P-version Layerwise Approach," *Composite Structures*, Vol. 118, 2014, pp. 74–85.

[10] Liguori, F. S., Zucco, G., Madeo, A., Garcea, G., Leonetti, L., and Weaver, P. M., "An Isogeometric Framework for the Optimal Design of Variable Stiffness Shells Undergoing Large Deformations," *International Journal of Solids and Structures*, Vol. 210, 2021, pp. 18–34.

[11] Rouf, K., Liu, X., and Yu, W., "Multiscale Structural Analysis of Textile Composites Using Mechanics of Structure Genome," *International Journal of Solids and Structures*, Vol. 136, 2018, pp. 89–102.

[12] Liu, X., Yu, W., Gasco, F., and Goodsell, J., "A Unified Approach for Thermoelastic Constitutive Modeling of Composite Structures," *Composites Part B: Engineering*, Vol. 172, 2019, pp. 649–659.

[13] Liu, X., Gasco, F., Yu, W., Goodsell, J., and Rouf, K., "Multiscale Analysis of Woven Composite Structures in MSC.Nastran," *Advances in Engineering Software*, Vol. 135, 2019, p. 102677.

[14] Long, Y., Yu, W., Fernandez, J. M., and Bergan, A., "Mechanics of Structure Genome-Based Nonlinear Shell Analysis," *AIAA SCITECH 2022 Forum*, 2022, p. 1119.

[15] Adams, B., Bohnhoff, W., Dalbey, K., Ebeida, M., Eddy, J., Eldred, M., Hooper, R., Hough, P., Hu, K., Jakeman, J., et al., "Dakota, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 6.13 User's Manual." Tech. rep., Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 2020.

[16] Yu, W., "A Unified Theory for Constitutive Modeling of Composites," *Journal of Mechanics of Materials and Structures*, Vol. 11, No. 4, 2016, pp. 379–411.

[17] Blom, A. W., Stickler, P. B., and Gürdal, Z., "Optimization of a composite cylinder under bending by tailoring stiffness properties in circumferential direction," *Composites Part B: Engineering*, Vol. 41, No. 2, 2010, pp. 157–165.