An automated unit-cell modelling tool UnitCells© on Abaqus platform drawing functionalities from multiple external codes

Q. Pan, T. Yu, L. Jeanmeure, S. Li
Faculty of Engineering, University of Nottingham, University Park, Nottingham NG7 2RD, UK

Abstract

This paper introduced an automatic unit cell modelling tool, UnitCells©, based on Abaqus as a platform, drawing facilitating functions from various codes, including Texgen for generating textile configurations and Hypermesh for meshing. It is to create unit cells for various types of composite materials at micro and meso scales. Using Abaqus as a platform and its Python Script programming facility as a vehicle, a fully automated composites characterization tool has been achieved. Various effective material properties of composites can be readily obtained by using this tool. These properties are elastic properties, thermal expansion, thermal and electric conductivity. The characterisation process is fully controllable using embedded visual interfaces to define relevant geometric and material parameters. It can also be used for multi-scale modelling in the sense that different phases can be characterised in one length scale and fed into the analysis of another unit cell at a scale above. As a demonstration, some cases of unit cell model were analysed using this tool and presented in this paper.

Keywords: multi-scale modelling, characterisation, unit cell, Python script, TCL script

1. Introduction

Finite element method (FEM) has been widely used for predicting the effective properties of composite materials, especially for modern textile composites [1]. Analyses are often based on micromechanics of composites where properties of the constituents and the architectures of composite are assumed to be known. The accuracy of overall effective material properties obtained from FEM relies on how well the model captures the real material. Therefore, sometimes modelling detailed architectures within composite material is necessary.

On micro scale, yarns in a textile composite can be regarded as UD composites and the matrix as composite of particulate inclusions such as performance enhancing additives or voids. The architectures of these two types of model are relatively simple and easy to create.

The meso scale models of composite materials, especially those of 3D braided composites, can have very complicated yarn architectures if a detailed representation of real yarn architectures is desired. Those models are notoriously difficult to generate. As a result, many approximated models were adopted by investigators.

Bigaud and Hamelin [2] introduced a simplified model. In their model, unit cell was discretized into a number of cubic elements, and the allocation of material properties for each integral point were determined by the position of each integration point, e.g. whether it is on the location of matrix or fibre. This kind of model is convenient to create, but it is unable to depict the local stress at the interfaces between yarns and matrix. Thus, the accuracy and applicability of this kind of model was limited, e.g. it cannot be used to predict strength.

Chen et al. [3] described a micro structure of 3D braided preform. In their model, the preform was divided to three different ranges: interior, surface and corner. Yarns in the interior range were regarded as constant cross section cylinder with straight path. This type of straight yarn model reflects better details than the model in [2] described above, which may obtain more accurate effective properties.

However, for real composite materials, the cross sections of yarns are not constant due to the interaction between adjacent yarns [4]. Fang et al. [5] proposed that the yarns within all representative volume cells should have octagon cross-sections and they should have
surface contact to the neighbouring yarns. The octagon cross sections were then divided into seven ranges and each range was assigned with a local coordinate system.

Jiang et al. [6] suggested a more detailed model for 3D braided composite by defining the paths of yarns as curves governed by mathematic functions. This model might be able to get more accurate results, but creating curved yarns are not always straightforward.

Depending on the architecture of the material, we can either create a unit cell [7] or a representative volume element (RVE) [8] to represent the whole model. The effective material properties obtained for the unit cell and RVE should be equivalent to that of the real material. In order to obtain the effective material properties of a unit cell or RVE, appropriate boundary conditions must be assigned. The periodic boundary conditions for a unit cell require that the coordinates of nodes on any pair of opposite faces be precisely related [9-10]. This dictates that the surface tessellation of the mesh for the unit cell has to be created accordingly to satisfy those boundary conditions. For 2D and 3D textile composite models which have complex fibre architectures that make it almost impossible to obtain a suitable mesh within a single piece of software, UnitCells© [11] as a piece of software was created to achieve this objective relatively easily.

Using Abaqus [12] as a platform and its Python Script programming facility as a vehicle, UnitCells© has been established as a fully automated composites characterization tool. Moreover, it is capable of drawing relevant functions from TexGen [13], an open source code for generating textile composites architectures, to generate the desirable textile preform configurations and from Hypermesh [14], a commercial FE pre-processor, to generate appropriate meshes. In an automated manner, correct periodic boundary conditions are imposed and precise loads are applied before a complete material characterisation simulation. The effective material properties of the composite are readily obtained from simulation using this tool. The characterisation process is also fully controllable using the visual interfaces to define relevant geometric and material parameters. It can also be used for multi-scale modelling in the sense that different phases can be characterised in a length scale below, e.g. yarns in a textile composite as UD composite and the matrix as composite of particulate inclusions such as performance enhancing additives or voids.

This paper intends to describe the main features of the newly developed UnitCells© tool. For demonstration, several multi-scale unit cell models for UD, particle-reinforced, woven and 3D braided composites were created and analysed using this tool for obtaining effective material properties like elastic properties, thermal expansion, thermal conductivity and electric conductivity.

As 3D braided composites were deemed to be a considerably challenging architecture for modelling, a detailed modelling investigation for this kind of material is described in section 6 to illustrate the difficulty associated with the problem and the capability of UnitCells© in dealing with this.

2. Overview of UnitCells©

Since UnitCells© is based on Abaqus, all the original functions of Abaqus can still be accessed when using this tool. The main interface window (Fig.1) of UnitCells© is almost the same as that of Abaqus, but with added new module named "UnitCells". A user's manual is also provided and embedded within this tool.

![Fig.1 Main window of UnitCells©](image)

As mentioned, UnitCells© would operate automatically for carrying out modelling and analysis tasks once all necessary input parameters are defined by the user in corresponding input panels. The analysis results will be displayed once the analysis is completed and written into a report file in plain text.

For micro-scale unit cell modelling...
involving the single fibre and surrounding matrix, the geometries are relatively simple and the unit cells included are entirely created and meshed in Abaqus without calling for other resources.

However, for meso-scale unit cell modelling, typically for textile composites, the creation of an appropriate unit cell for FE analysis is significantly more challenging than the micro-scale cases. Taking plain weave woven composite for example, because of its weaving pattern, it is difficult to generate the geometry and mesh using Abaqus alone. In this case, UnitCells© would call up TexGen [13] to build the geometry, and then employ Hypermesh to mesh the model. All these processes were controlled by Python scripts and executed automatically.

3. Unit cells for UD fiber-reinforced composites

On micro-scale level, a lamina or a yarn within fibre-reinforced composites can be regarded as unidirectional fibre-reinforced composites. Typical idealised packing scheme of UD composites are hexagonal and square arrangement [9] as shown in Fig.2. Both of these have relative simple geometry, and can be readily meshed using Python Script executable in Abaqus. For this type of analysis in UnitCells©, users only need to input the fibre volume fraction and material properties of fibre and matrix, all the following analysis process will be executed automatically. It is user’s choice which of the two should be used. Considerations can be reflection of reality as close as possible or as idealisations of the reality. Users are nevertheless reminded that UD composites in which fibres are distributed at random over their cross sections usually exhibits transversely isotropic characteristics in a statistic sense and the hexagonal packing captures this feature while the square packing does not.

Internally, half-sized unit cells are used inside UnitCells© to reduce computational effort. The boundary condition imposed for this is based on central reflection symmetry condition [15].

![Hexagonal and Square Unit Cells](image)

Fig.2 Unit cells for UD composites

For the demonstration of this tool, a set of hexagonal and square unit cell models have been analysed. Fibre and matrix were defined as isotropic materials, and their properties (elastic constants, thermal expansion coefficients and thermal conductivities) were as follows:

\[
\begin{align*}
E_f &= 76 \times 10^6 \text{Pa} \\
E_m &= 3.01 \times 10^9 \text{Pa} \\
\nu_f &= 0.2 \\
\nu_m &= 0.3 \\
\alpha_f &= 4.9 \times 10^{-6} / \degree C \\
\alpha_m &= 60 \times 10^{-6} / \degree C \\
\kappa_f &= 4.1 \text{ W/mK} \\
\kappa_m &= 0.4 \text{ W/mK}
\end{align*}
\]

All the unit cells were set to have the same fibre volume fraction (45%). Their effective properties obtained by UnitCells© are shown in Table.1.

<table>
<thead>
<tr>
<th>Unit Cell</th>
<th>( E_1 ) (GPa)</th>
<th>( E_2 ) (GPa)</th>
<th>( G_{12} ) (GPa)</th>
<th>( G_{13} ) (GPa)</th>
<th>( \nu_{12} )</th>
<th>( \alpha_1 ) (( 10^{-6} / \degree C ))</th>
<th>( \alpha_2 ) (( 10^{-6} / \degree C ))</th>
<th>( \kappa_1 ) (W/mK)</th>
<th>( \kappa_2 ) (W/mK)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Square</td>
<td>35.49</td>
<td>7.976</td>
<td>2.206</td>
<td>2.850</td>
<td>0.247</td>
<td>7.699</td>
<td>38.00</td>
<td>2.046</td>
<td>0.871</td>
</tr>
<tr>
<td>Hex</td>
<td>35.70</td>
<td>7.004</td>
<td>2.557</td>
<td>2.824</td>
<td>0.247</td>
<td>7.661</td>
<td>38.08</td>
<td>2.057</td>
<td>0.869</td>
</tr>
</tbody>
</table>

From Table.1 we can see that the results for hexagonal and square unit cell are different to some extent, especially those in the transverse direction. The difference will be more pronounced if the Young’s in 45° direction in the plane transverse to the fibre is evaluated [9].

4. Unit cells for particulate-reinforced composites

For particulate-reinforced composites, its effective material property is isotropic in a statistic sense. There are a range of idealised packing systems for particle-reinforced composites [10]. As a compromise between complexity of model construction and
reasonable capture of key features in reality through idealisation, two packing schemes have been adopted in UnitCells©, simple cubic (SCP) and face centred cubic (FCC). Users can choose either to create unit cells automatically using appropriate geometric dimensions/volume fraction and constituents properties. In general, unit cells for particle-reinforced composites have more complex geometry than that of UD composites, making them more difficult to build and mesh automatically. Fortunately, after some rational partitions, particle-reinforced unit cells for SCP and FCC packing can be created and meshed properly using Python Script in Abaqus. For instance, the FCC unit cell has the shape of a dodecahedron, which could be divided into four parallelepipeds, and every parallelepiped was constructed by three quadrangular pyramids as sketched in Fig.3.

![Fig.3 A dodecahedron unit cell for particulate-reinforced composites](image)

Within each parallelepiped, a further partition is introduced to allow the definition of the reinforcing particulate and its surrounding matrix. The reinforcing particulate can be idealised as a sphere as illustrate in Fig.4, a dodecahedron or rendered between these two shapes to allow the shape of the reinforcing particulate to deviate from a perfect sphere to an extent. The reinforcing particulate can also be replaced by a void to simulate a porous material.

![Fig.4 The mesh of FCC unit cell](image)

The same scheme can be easily adapted for the generating of the unit cell for SCP, where the parallelepiped is replaced by a right cube and there are eight of them to construct a full unit cell. Again, central reflection symmetry [15] has been employed to reduce the computation demand in this case.

These unit cells with user-defined particulate volume fractions can be created and analysed for their effective elastic constants, thermal expansion coefficients, thermal conductivity coefficients and electric conductivities. Similar to the analysis process for UD composites mentioned, the user only needs to input basic parameters.

For demonstration, a set of SCP and FCC unit cell models have been analysed. In these models, inclusions and matrix were defined as isotropic materials. Properties used for modelling inclusion and matrix are assumed to be the same as those for the UD composite as in the previous section.

All unit cells modelled had the same inclusion volume fraction (45%) and their equivalent properties obtained by UnitCells© is shown in Table.2. Although the predicted Young’s modules in all three coordinate axes are identical due to the symmetries present in the material, the materials represented by these unit cells are not isotropic. Evaluation of their counterparts in an off-axis direction, e.g. along a diagonal, would reveal significant discrepancies [10]. In this respect, that for FCC leads to an overall behaviour much closer to isotropy than that for SCP.

<table>
<thead>
<tr>
<th>Unit Cell</th>
<th>$E$ (GPa)</th>
<th>$\nu$</th>
<th>$G$ (GPa)</th>
<th>$\alpha$ (10^{-6}/° C)</th>
<th>$\kappa$ (W/mK)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCP</td>
<td>10.16</td>
<td>0.193</td>
<td>2.798</td>
<td>28.87</td>
<td>1.062</td>
</tr>
<tr>
<td>FCC</td>
<td>7.033</td>
<td>0.274</td>
<td>3.107</td>
<td>29.77</td>
<td>0.9997</td>
</tr>
</tbody>
</table>

5. Unit cells for woven composites

Yarns in woven fabric of textile composites are difficult to model using Abaqus alone. This issue is worsened when different architectures are involved. This has been tackled in UnitCells© by virtue of cross-linking Texgen [13] and Hypermesh[14] during the modelling process.

In UnitCells©, geometry of yarns was established by Texgen (open source software...
which is specially developed to generate textile preform pattern geometries) [13].

Due to the complicated yarn architectures of textile composites, it is often difficult to ensure identical tessellations on corresponding faces of the unit cell before periodic boundary conditions can be imposed [10]. However, thanks to advanced meshing capability in Hypermesh[14], a commercial FE pre-processor, yarn geometry generated by Texgen can be imported and meshed properly using TCL script within Hypermesh. The mesh so generated can be imported into Abaqus then for further modelling processes.

In reality, the yarns in woven composites are curved. To capture this geometrical feature, also bearing in mind that yarns can be reasonably approximated as transversely isotropic, there is a need to define local coordinate systems for the yarns along their paths when they are modelled in UnitCells©. Taking advantage of TexGen which is capable of defining local coordinate systems for yarns based on the geometry created, UnitCells© calls TexGen to define the local coordinate system after the mesh is generated by Hypermesh. The yarn’s local orientation can then be imported into Abaqus for analysis. As mentioned, all above procedures are automated in UnitCells© via execution of Python script in Abaqus, human intervention is only required for manual input of basic parameters. An example of a plain-weave woven composite was created and meshed using UnitCells© (Fig.5). It has been proven that the mesh generated from this tool satisfied node correspondence requirements such that periodic boundary conditions can be imposed to the unit cell.

It cannot be overemphasised the significance of basic ‘sanity checks’ on the unit cell generated. Users of UnitCells© are also advised to conduct such checks before committing to serious applications. Although the software have been extensively verified in this respect for all the unit cells involved, one will be more confident with the results produced, especially when sophisticated internal architectures are involved. To facilitate such ‘sanity checks’, one can modify the material properties of all phases to a known set, preferably, that corresponding to an isotropic material, such that the unit cell becomes a uniform block of material. To pass the ‘sanity checks’ uniform stress fields should be obtained under all loading conditions, as illustrate in Fig. 6 for one of the loading cases and the predicted effective properties reproduce the input data identically, except for numerical noises due to rounding errors. Although it is a little boring to read the contour plot as well as the legend, it is crucial to be able to achieve this, as one is assured that something is bound to be wrong if one obtains a fancy and colour contour plot instead in this case. Any prediction using a model which fails to pass such ‘sanity checks’ is deemed to be meaningless.
Other weave patterns such as twill (Fig. 5(c)) and satin are also available in UnitCells© for users to choose in order to construct appropriate unit cells and carry out full analysis for the characterisation of composites made of such textile preforms.

A set of trials for textile composite unit cells have been carried out where weft and warp yarns were defined to have the same properties based on hexagonal unit cell result as shown in Table 1, and the properties for matrix were defined based on the results from FCC unit cell as shown in Table 2. The predicted effective properties from the trials are shown in Table 3 below with all textile composite unit cells kept at the same yarn volume fraction of 52%.

<table>
<thead>
<tr>
<th>Unit cells</th>
<th>$E_1$ (GPa)</th>
<th>$E_2$ (GPa)</th>
<th>$E_3$ (GPa)</th>
<th>$G_{23}$ (GPa)</th>
<th>$G_{31}$ (GPa)</th>
<th>$G_{12}$ (GPa)</th>
<th>$\nu_{12}$</th>
<th>$\nu_{31}$</th>
<th>$\nu_{13}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plain weave</td>
<td>13.27</td>
<td>13.27</td>
<td>7.409</td>
<td>2.767</td>
<td>2.767</td>
<td>2.795</td>
<td>0.15</td>
<td>0.15</td>
<td>0.18</td>
</tr>
<tr>
<td>Twill 2/2</td>
<td>13.78</td>
<td>13.79</td>
<td>7.455</td>
<td>2.750</td>
<td>2.750</td>
<td>2.796</td>
<td>0.142</td>
<td>0.142</td>
<td>0.175</td>
</tr>
<tr>
<td>Satin 4 H</td>
<td>13.89</td>
<td>13.89</td>
<td>7.459</td>
<td>2.748</td>
<td>2.748</td>
<td>2.797</td>
<td>0.143</td>
<td>0.145</td>
<td>0.171</td>
</tr>
</tbody>
</table>

6. Unit cells for 3D braided composites

The yarns of 3D braided composites are normally of very complex meso-scale architectures as sketched in Fig. 7 [15] which can hardly be modelled using Abaqus alone. In UnitCells©, the modelling of 3D braided composites is achieved with relative ease. For simplicity, yarns assumed to be straight. Choices of a range of yarn cross sections are provided in UnitCells© in the main input panel.

To demonstrate this, a four-directional 3D braided composite model with yarns of circular cross section was created using UnitCells© and illustrated in Fig.8.

6.1 Yarns volume fraction

The most restrictive aspect of the circular idealisation of the yarn cross section is the fiber volume fraction which can be achieved as a reasonable representation of 3D braided composites in reality which can be normally be as much as 50% [16]. The maximum yarn volume fraction achievable with this idealisation is around 42%. Which varies slightly with the braiding angle. Bearing in mind of a realistic fibre volume fraction, for instance, at 70%, the fibre volume fraction in the composite is far too low to cover a practically useful range. As a result, alternative will be sought for in rest parts of this section.

6.2 Elliptical yarn cross section

It is clear that there are gaps inaccessible for yarns circular cross section to occupy (Fig.8). To fill some of the gaps, elliptical yarn cross sections could be used instead of circular ones to increase the yarn volume fraction. Fig.9 shows a unit cell with such a modification.
It can then be found that the maximum yarn volume fraction obtainable by adjusting the aspect ratio of the ellipse at around 70%. This is then possible to deliver an overall fibre volume fraction of 50%. This option is therefore provided in UnitCells© for users to choose.

6.3 Rectangular cross section

As a geometrically simple choice, rectangular cross section can be chosen for the yarns as shown in Fig. 10. However, the simplicity is at a price. The maximum yarn volume fraction obtainable is only 50%, which is independent of the braid angle. This option is also made available in UnitCells©.

6.4 Hexagonal cross section

Another attempt to maximize the yarn volume fraction with relatively simple yarn cross section shapes is presented below. The cross section is a sideway squashed hexagon as sketched in Fig. 11. The highest yarn volume fraction of 75% is obtained when \(2r=b\), i.e. yarns are in touch with each other, as shown in Fig.12 for one of the end surfaces of the unit cell, and \(l_m=2l_a\). A unit cell produced in UnitCells© is illustrated in Figure 13.

Similar to unit cells with rectangular yarn cross section, the max yarn volume fraction for hexagonal yarn cross section is also independent of braiding angle.

Using the results from hexagonal unit cell as shown in Table 1 for yarns, and the results from FCC unit cell in Table 2 for matrix, the effective properties for a set of 3D braided composites with yarns of different cross sections are shown in Table 4. All 3D braided composite unit cells used were set to the same yarn volume fraction of 40%.
Another useful feature of UnitCells© is laminate characterisations as an analysis at macroscopic scale. Designers of laminates are often interested in their effective properties, such as Young’s module and flexure modulus. Conventional laminated analysis produces the stiffness matrices. Working out the effective properties is not difficult but often subject to confusions, e.g. mixing up with the stiffness and modulus.

The laminate characterisation offered in UnitCells© can start from constituent lamina with known properties for the lamina involved as a straightforward application. However, UnitCells© can accommodate more sophisticated multi scale characterisations as described below.

The lamina for the laminate can be obtained from the micro or meso-scale characterisations previously, e.g. for a UD composite and any textile preforms available in UnitCells©. In fact, users are offered a choice after micro or meso-scale characterisations if they would like to proceed with a laminate characterisation. Also, the matrix in the micro or meso-scale characterisations can be moderated microscopically. For instance, performance enhancing additives can be introduced in the matrix before it is used to impregnate fibres or fabrics before forming the composite [1]. In this case, the unit cell for particulate-reinforced composites can be employed to fulfil the role. Using the results from the hexagonal unit cell as shown in Table 1 for a symmetric quasi-isotropic laminate $[0^\circ/90^\circ/45^\circ/-45^\circ]_s$, as a continuation of the UD composite characterisation, the laminate effective properties are shown in Table 5 below.

### Table 4 Effective properties of 3D braid composite with different yarn cross sections with a braiding angle of 21.8° (yarn volume fraction at 40% except for the final row as indicated differently at 75%)

<table>
<thead>
<tr>
<th>Yarn cross section</th>
<th>$E_1$ (GPa)</th>
<th>$E_2$ (GPa)</th>
<th>$E_3$ (GPa)</th>
<th>$G_{23}$ (GPa)</th>
<th>$G_{31}$ (GPa)</th>
<th>$G_{12}$ (GPa)</th>
<th>$v_{32}$</th>
<th>$v_{31}$</th>
<th>$v_{12}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circle</td>
<td>7.175</td>
<td>7.175</td>
<td>14.65</td>
<td>3.649</td>
<td>3.651</td>
<td>2.894</td>
<td>0.315</td>
<td>0.316</td>
<td>0.306</td>
</tr>
<tr>
<td>Rectangular</td>
<td>7.176</td>
<td>7.176</td>
<td>14.81</td>
<td>3.647</td>
<td>3.647</td>
<td>2.894</td>
<td>0.317</td>
<td>0.317</td>
<td>0.304</td>
</tr>
<tr>
<td>Ellipse</td>
<td>7.175</td>
<td>7.175</td>
<td>14.61</td>
<td>3.647</td>
<td>3.647</td>
<td>2.899</td>
<td>0.317</td>
<td>0.317</td>
<td>0.304</td>
</tr>
<tr>
<td>Hexagon</td>
<td>7.176</td>
<td>7.176</td>
<td>14.81</td>
<td>3.647</td>
<td>3.647</td>
<td>2.899</td>
<td>0.317</td>
<td>0.317</td>
<td>0.304</td>
</tr>
<tr>
<td>Hexagon (75%)</td>
<td>7.132</td>
<td>7.130</td>
<td>20.81</td>
<td>4.085</td>
<td>4.088</td>
<td>2.717</td>
<td>0.354</td>
<td>0.354</td>
<td>0.309</td>
</tr>
</tbody>
</table>

### Table 5 Effective properties of symmetric quasi-isotropic laminate

<table>
<thead>
<tr>
<th>Unit Cell</th>
<th>$E_x$ (GPa)</th>
<th>$E_y$ (GPa)</th>
<th>$V_{xy}$</th>
<th>$V_{yx}$</th>
<th>$G_{xy}$ (GPa)</th>
</tr>
</thead>
<tbody>
<tr>
<td>In-plane</td>
<td>16.55</td>
<td>16.55</td>
<td>0.295</td>
<td>0.295</td>
<td>6.389</td>
</tr>
<tr>
<td>Flexural</td>
<td>24.47</td>
<td>16.40</td>
<td>0.107</td>
<td>0.160</td>
<td>3.728</td>
</tr>
</tbody>
</table>

### 8. Conclusions

The present paper introduced a newly developed piece of software UnitCells© for micro/meso/macro level analysis of unit cells representing composites of various micro/meso/macro-architectures. This tool is built on the platform of Abaqus, drawing a range of facilitating functionalities from other established codes including Texgen and Hypermesh for mesoscopic architectures of textile composites and mesh generations. A high level of automated operations has been achieved. The effective properties of the composites concerned can be obtained as the direct outcomes of the highly automated analysis. The capabilities of UnitCells© have been illustrated through a range of typical examples presented in this paper, including models for UD, particle-reinforced, woven and 3D braided composites and laminates constructed from the composites characterised within UnitCells© or defined separately from any known laminate.

### Acknowledgement

The first, second and fourth authors wish to acknowledge the financial support from ACAE, Shanghai, China to the publication of this paper at ICCM19.

### References

[1] H Li, S Li, Yongchang Wang, Prediction of effective thermal conductivities of woven fabric composites using unit cells at multiple


