OPEN DATA FORMATS AND SCRIPTING IN INTEGRATED MESO-LEVEL TEXTILE COMPOSITE SIMULATIONS

S.V. Lomov*, I. Verpoest1, J. Cichosz2, Ch. Hahn2, D.S. Ivanov3, B. Verleye4

1 Department MTM, KU Leuven, Belgium, 2 Lehrstuhl für Carbon Composites, TU Muenchen, Germany, 3 ACCIS, Faculty of Engineering, University of Bristol, UK 4 Department of Computer Sciences, KU Leuven, Belgium

* Corresponding author (stepan.lomov@mtm.kuleuven.be)

Keywords: textile composites, modeling, mechanical properties, internal geometry, permeability

1 Introduction

Meso-level (unit cell) modelling of textile composite is an established field, with a plethora of works published and in-house, commercial and open source software tools available, for example WiseTex [1] and TexGen [2]. A meso-level textile processor (MLTP) can be defined as a numerical tool, which:

− accepts information of a textile reinforcement parameters (such as weave structure, yarns spacing, yarn dimensions etc.), parameters of the composite (overall fibre volume fraction, ply thickness, nesting of the plies etc.) and local (in relation to a scale of a composite part) overall deformation of the reinforcement (shear, compression etc.), and

− creates a geometrical model of the reinforcement in the given textile composite.

The geometrical model can be further processed by different models to produce such parameters as permeability of the textile reinforcement, stiffness matrix, thermal conductivity or dumping parameters of the consolidated composites etc. Moreover, the unit cell geometrical model can be transformed into a “general purpose” meso-level finite element (FE) model of the unit cell, allowing further in-depth simulation of the composite/reinforcement properties and behaviour.

Given a MLTP, the user gains access to its functionality via its “native” data formats and user interface. Plugging the MLTP in an integrated simulation chain, however, most often requires interaction and active collaboration with the author(s) of the MLTP, and is not a trivial task. The aim of the present paper is to demonstrate possibilities, given by the open data exchange format (XML) and command line scripting possibilities, on an example of WiseTex, an MLTP developed in KU Leuven [1]. These features allow integration of the MLTP into custom modelling systems, addressing such tasks as, among others: (1) parametric studies of composite properties, depending on the parameters of the textile reinforcement, (2) integration with meso-level finite element (FE) modelling; (3) upstream integration with textile process models; (4) upstream integration with simulations of composite processing; (5) downstream integration with structural and impregnation analysis of composite parts. The reader is referred to [3] for detailed discussion of MLTP and the data exchange.

2 Data exchange and data formats

The general data flow in a MLTP is shown in Fig.1. A MLTP can be seen as a processor of TEX (textile data) and DEF (deformation data) into GEO (geometrical model). If the DEF is absent (no deformation, relaxed state), then the deformation of the dry textile can be modelled in post-processing of GEO, for example, via meso-FE models. TEX and DEF input are either user-specified, or are a result of (pre)-processing in textile and composite process models. Examples of the former are models of braiding, which deliver such parameters as braiding angle and the braiding density depending on the machine settings. For weaving, software packages as ScotWeave and its likes are widely used in apparel and technical fabrics industry. The latter, composite
processing models, are represented by draping simulations, which computes local in-plane deformations and the thickness of a reinforcement during forming.

TEX data is organised in a hierarchy of data levels for fibres, yarns and the fabric. The easy and open way of implementing this hierarchy is use of XML (Extensible Markup Language), that defines a set of rules for encoding documents in a format that is both human- and machine-readable.

3 Scripting

MLTP and the micromechanical, permeability etc modelling software calculations can be controlled by direct user input via GUI or by commands in a script, which allows programming of complex calculation tasks and integration into wider modelling environment, with script commands generated by other modules of the software. Scripts in WiseTex and in micromechanical (TexComp) and permeability (FlowTex) software in the WiseTex package use the syntax of DOS command line.

There are different possibilities for use of the scripts. The simplest are serial calculations with TEX input data sets with varying parameters (for example, changing ends/picks spacing of a woven fabric or braiding angle for braids) – the result of such calculations will be a set of GEO models and the corresponding reinforcement permeability and/or stiffness of the impregnated composite. Look-up tables can be built with this technique for the following use in macro-level flow modelling or structural analysis of composite parts:

```
for p = 0.001, 0.002, 0.05, 0.01, 0.02 // MPa
  WiseTexCL "woven.XML" "compr.XML" compress:p
  extract VF value from "compr.XML"
end
```

Scripting allows also integration of the meso-level simulations in macro-analysis. For example, consider the problem of structural analysis of a 3D part, with shear angles of the textile reinforcement different from point to point because of draping of the reinforcement over the mould. Assume that a draping simulation software generates values of the reinforcement shear angle GAMMA for all finite elements (i is the element number) in the part model. Then the script for calculation of the stiffness of the composite in all the element will be as follows:

```
for all elements
  WiseTexCL "unsheared.xml" "sheared.xml" in-plane:element.GAMMA
  TexCompCL sheared.xml epoxy.xml C.csv
  transform C matrix into the local coordinate system of the element
  store C in element
end
```

4 Examples

The integration approach is illustrated on the following examples: (1) processing of digital images and calculations of the fabric permeability tensor, (2) micromechanics models and (3) meso-level finite element ABAQUS model

4.1 Permeability: Image analysis – MLTP – Stokes solver – post-processing

Resin injection such as resin transfer moulding (RTM) is commonly applied in industry for serial production of composite components. RTM process simulation enables prediction of process parameters that are essential for component and process development [4, 5]. Material properties are indispensable for realistic simulations. These are mostly determined experimentally, which is costly and time consuming. Furthermore, methods for permeability testing are not yet standardized, which leads to wide scatter of the measurement results [6]. Meso-level calculations of homogenised textile permeability [7, 8], which provide estimations of permeability tensor for macro-level impregnation
OPEN DATA FORMATS AND SCRIPTING IN INTEGRATED MESO-LEVEL TEXTILE COMPOSITE SIMULATIONS

simulations (Darcy solvers), can serve as (a) a “first guess” of the permeability when experimental data is not readily available; (b) as guidelines for experiments, eliminating obvious outsiders and reducing the number of tests and (c) as a source for establishing dependencies of permeability on fibre volume fraction and shear angle of the preform, reducing experiment to a limited number of reference sets of these variables.

The open data exchange and scripting of the MLTP (WiseTex) are used to carry on permeability predictions for stitched carbon fibre multiaxial fabrics (so-called non-crimp fabrics, NCF [9]) based on a unit cell data extracted from image analysis of fabric images. The integration data flow, presented in detail in [10] is shown in Fig. 2. The main source of TEX information for permeability prediction are digital images of scanned carbon fiber fabrics. TEX data is extracted from image analysis and forwarded to the MLTP via an open XML user interface. GEO is employed to determine effective permeabilities using the Stokes solver implemented in the FlowTex software [8, 11]. Results of permeability modeling are post-processed to obtain a principal permeability tensor. For the entire path, a MatLab software tool has been developed that conducts the following steps: (1) image analysis, (2) data transfer to the MLTP, (3) model export, discretization (voxel mesh), permeability calculation and (4) post-processing of results and export to a material card for RTM solver. For all steps where the MLTP and permeability model are involved, the tool is employed to control them via command line scripting.

4.2 Micromechanics: beam model for braided composites

The possibilities of integrating GEO modeling of an MLTP with micromechanical calculation are not limited to a link with an inclusion model (WiseTex – TexComp). The generic nature of the GEO data allows easy programming an interface to a specific model. This statement is illustrated here with an example of a simplified FE model of a braided composite, which uses 1D beam elements.

The braiding preforming process offers the possibility to produce near-net-shaped preforms, which are subsequently impregnated to a braided composite. Most braided composites are produced to high fiber volume fractions resulting in tight yarn architectures. This means that there is no free space between two adjacent yarns and that the yarn cross sectional shape and dimension (TEX data) show strong variation inside the braided composite.

As these variations lead to locally changing material properties, constitutive modeling of braided composites has to allow calculations with various sets of TEX input parameters. The prediction of the homogenized macroscopic constitutive behavior of a biaxially braided composite can be conducted using a MLTP to obtain the unit cell GEO description of the textile yarn architecture. While analytical models can be used for the prediction of elastic properties [12, 13], stress analysis and failure prediction are commonly conducted using finite element (FE) models. To simplify the meshing procedure, an alternative approach for the generation of a FE unit cell is used here: the main idea is to represent the unit cell geometry by using beam elements for the yarns and thus reduce effort for meshing and handling interpenetrations.

The idea of a FE unit cell with yarns represented by one-dimensional elements was first published by Cox [14], who used truss elements to construct a “binary model”. The approach presented here splits up the unit cell consisting of yarns and matrix pockets into beam elements and so-called effective medium continuum elements. The beam elements are used to represent the longitudinal and bending properties of the yarns, while transversal and shear properties of the yarns as well as the properties of the matrix pockets are smeared in the isotropic effective medium continuum elements. The properties both the beam elements and the effective medium continuum elements, are calculated using Hashin’s [15] circular cylinder model. For the beam
elements, representing the yarn longitudinal stiffness, the packing density (fiber volume fraction inside the yarn) is used for the calculation, while the smeared properties of the effective medium elements are obtained by using the fiber volume fraction of the braided composite. Finally, to account for the overlapping volumes of the beam and the effective medium elements, the beam elements stiffness is obtained by subtracting the effective medium stiffness from the yarn stiffness. More details on the implementation can be found in [16].

**Fig 3 Beam model generation**

The FE model of the beam unit cell is generated using a MatLab framework, which enables using the MLTP XML data format and command line functions. Furthermore, Python scripts for the transfer of the model to the third-party FE-software can be called using the framework. An overview of the model generation is given in Fig. 3. In the MatLab framework, a standard WiseTex XML-file (extension .B2DX) for the 2D braid is used to define nominal TEX parameters, including the model specifications that do not change in the analyses (e.g. fiber name or textile parameters of the yarn). Results from photomicrographs of the braided composite are used to assess the variations of the TEX data (e.g. yarn width, spacing etc.). This data serves as an input for the MatLab framework, which uses XML-functions to edit the TEX values in the standard Braid.B2DX file and writes it to a new NewBraid.B2DX file.

The GEO data of the braid is calculated by using the command line option of the MLTP, which can be easily invoked from MatLab. The GEO data is also stored in the NewBraid.B2DX file and is furthermore used to build up the meso-FE beam unit cell model in Abaqus. The MatLab framework calls the Python scripting interface of Abaqus/CAE to import the coordinates of the yarn path, which are connected using spline functions. Furthermore, the dimensions of the yarn cross section are used to define the beam cross section and the material coordinate system is used to assign the cross section orientation to every beam element. The surrounding box of the effective medium is built and the mesh procedure as well as the coupling of beams and continuum elements are conducted by the script. After the model generation, the FE analysis is started from the framework. After the analysis has finished, further post-processing steps, for calculating elastic constants and for the stress analysis in the unit cell, are conducted. Fig 4 illustrates exemplarily the result of these calculations: stresses inside the yarn obtained using the beam unit cell in comparison with full FE modelling.

**Fig 4 Beam modelling of 2/2 carbon braided composite: stresses along a braiding yarn, in comparison with full FE modeling**

### 4.2 Meso-FE: FE models of textile unit cells

The local behaviour of fabrics and textile composites can be simulated with various scale resolutions. An in-depth understanding of deformation mechanisms is often impossible without the 3D finite element models of textile reinforcements at the unit cell level. The integration of a MLTP model to standard finite element packages provides a direct input for these simulations. The integration does not only require a reconstruction of solid volumes from the geometric specification. It also implies that the adjustment of input geometry has to be done. The most important deficiency of GEO models generated by most of MLTPs is interpenetration of the yarn volumes. GEO data can be used as a starting point for other yarn interpenetration processing
algorithms, like contact resolution [17], separation-and-compression [18, 19], “consistent models” [20], yarn inflation [21], or for other approaches for building a meso-FE model, like voxel models [22, 23].

The downstream integration of MLTP models requires a tool, which (a) automatically translates the geometry into finite element software, (b) creates solid models and performs the automatic corrections of yarn volumes, and (c) assigns properties and sets contact surfaces. The here we sketch such a tool for the meso-scale modelling of textile preforms in Abaqus. It is written in Python code which is used in Abaqus as one of the internal languages.

The tool reads the WiseTex model (GEO data) and organises the geometric primitives in accordance to the data structures of Abaqus. Every yarn is treated as a separate part. Each part represents an object, which contains a set of attributes and sub-objects: Features, Section Assignments, Orientations, etc. In the context of textile modelling, the features are (a) the section sketches, which contain the planar geometry of yarn cross-sections, (b) the coordinate planes in 3D space where the sketches are projected to.

The sketch sections are discretized. They come up from WiseTex as ellipses and they are then transformed to polygons connecting a user-defined number of points on the ellipse contour. Each point on the upper side of the contour contains a symmetric twin on the lower side. Each point can be moved along the line connecting twins. The displacement of twins is coupled to preserve the constant distance between them. Once the cross-sections are projected, the solid volumes are constructed by connecting these sections through an Abaqus loft operation. The result is the solid model of a yarn built as a set of segments connecting the cross sections. Every segment has its own fibre coordinate systems and, if necessary, a set of material properties defined by intra-segment fibre volume fraction. The part instances are then combined in the Assembly object representing all the yarns in a unit cell model.

Such a data organisation is convenient for improving the cross-section shapes. The major challenge of this “surgery” is to identify the interpenetration and remove them locally without lossoing the yarn volume and hence, without increasing the intra-yarn fibre volume fraction. The excessive fibre volume fraction is a common problem for the majority of known meso-scale models [24]. The yarn sections modification is done in two stages: (1) handling lateral yarn intersections between parallel yarns (e.g warp-warp), (2) removal of intersections between interlacing yarns (e.g warp-weft).

An example of dense twill carbon fabric (areal density 370g/m², 12K yarn tows) is shown in Fig 5. The analysis of the fabric clearly shows that the spacing between parallel yarns is smaller than the inter-yarn distance at some locations. This is a rather complex situation, which is hardly discussed in literature. It can be seen that the simple operations described above substantially improve the quality of the model and reduce the yarn intersections. The waviness arising from handling the lateral intersections closely resembles the yarn waviness in reality. This indicates that the purely geometrical algorithm may reflect the physics of yarn interaction. The intersections are not completely eliminated as far as the point-discretization is rather rough. However they are minimised and resolved to an acceptable degree when contact problems are easily solved. Thus, the tool, integrated with MLTP models, has shown to be effective for modelling the complex features of real textile architectures.

Conclusions
The open data exchange and scripting in meso-level textile composite modelling software allow an easy integration of meso-modelling in custom composite simulation systems and in the full modelling chain textile process – composite processing – composite material – composite part.

The upstream part of this chain links manufacturing/processing parameters (TEX and
DEF data) with the internal architecture of the composite reinforcement (GEO data), treated locally in the composite part. The downstream part refers to models transforming the structural, geometrical description of the unit cell into properties of the composite: mechanical, transport (fluid, thermal) or physical (e.g., electrical conductivity). Being local, these results can be further transferred to macro-level structural, thermal, impregnation... analysis of a composite part as properties of discretisation elements (most commonly, finite elements) of these models.

Acknowledgements

The work in KU Leuven (SL, IV, BV) was performed in the framework of SIMPOSIUM project, funded by European Commission (FP7). The work in Bristol (DI) has been supported by InterCom project, funded by Marie Curie Career Integration Grant of European Commission (FP7). In Munich (JC) the work was performed in cooperation with the Polymer Competence Center Leoben GmbH (PCCL, Austria) within the framework of the COMET-program of the Federal Ministry for Transport, Innovation and Technology and the Federal Ministry of Economy, Family and Youth.

References