FINITE ELEMENT MAPPING FOR A STRUCTURAL ANALYSIS OF COMPOSITES

N. Mayer[a*], R. Hinterhoelzl[a], J. Prowe[b]

[a] *Technische Universitaet Muenchen,, Faculty of Mechanical Engineering, Institute for Carbon Composites, Boltzmannstr. 15, 85748 Garching b. Munich, Germany*
[b] *AIRBUS Group Innovations, Department TX3DL - Structures Engineering, Production and Aeromechanics, 81663 Munich, Germany*
**mayer@lcc.mw.tum.de*

Keywords: structural composites, mechanical properties, finite element analysis.

Abstract

A structural analysis for carbon composite is using a finite element analysis based on nominal definition of composite parts. Nominal material data sets can result unrealistic structural behavior. An improvement of simulation results can be realized with data sets from process simulations. Different finite element meshes augment a complexity of data modification and transformation between various applications. A main challenge is to provide a mesh-independent interface for finite element data conversion and to consider converted information in structural analysis. In this work, an approach for finite element data conversion for structural analysis is presented. All relevant material parameter and model data can be exchanged with developed mapping algorithms between process simulation tools and a mechanical analysis using structural simulation.

1. Introduction

Simulative investigations of manufacturing processes and a subsequent structural analysis for composite components operate with different software packages. Simulation setups distinguish in finite elements, process and material parameter. The software packages from ESI Group, SIMULIA Abaqus and MSC.Nastran are mostly used in the analysis steps such as draping, infiltration, curing and structural simulation. The existing interactions between commercial simulation packages are limited. Therefore, it is difficult to provide calculated information for the next simulation step [1,2] Most interfaces are based on internal solutions or additional self-made scripts. For example, an interface between draping simulation and structural analysis in Laminate Modeler tool [3] is based on CATIA data format. The software Beta-CAE ANSA [4] used an internal approach for data transfer into a crash simulation. The e-Xtream Engineering tools [5] convert Moldflow data onto an imported mesh from ANSYS, ABAQUS or PAM-Crash. ANSYS [6] software converts data sets from FiberSIM tool, however, only onto an internal mesh.

A significant challenge is to provide a mesh-independent interface for data transport and to respect transported data in mechanical and material laws. Furthermore, transferred approximated data can also result unrealistic structure behavior of composite components. Therefore, a mapping algorithm is required to transfer available data sets from the source finite element mesh into the destination mesh. These algorithms are based on two or three dimensional search algorithms with interpolation and extrapolation methods. Research fields with a crash analysis, an image analysis and computer graphic presented different search algorithms for detection reference elements in a source mesh for a target mesh. For example, contact algorithms [7] are used for detection, tracking and calculation of contact points between two parts. They operate with analytical or discrete methods**.** Contact algorithms can be classified in Lgrangian and Eulerian algorithms. Another search algorithm “Approximate Nearest Neigbor Searching” was implemented by David M. Mount and Sunil Arya [8]. Transfer of nodal, element and integration point data is usually realized by interpolation and extrapolation methods. A radial basis interpolation [9] and inverse distance methods [10] are mostly used for interpolation of data in grid points.

This paper describe in details a concept of mapping algorithms for a data transfer between process and structural simulation of composite parts. The mapping algorithm is based on Bucket Sort search algorithm [11] and shape functions of finite elements. This interface approach for composites used a common data format definition and wrapping tools for a data set translation in the defined format. In combination with wrapper tools, the developed common data format and mapping algorithms defines a simulation platform for more realistic finite element analysis. The simulation platform is verified with well-known composite structures such as L profile and industry relevant parts.

2. Description of analysis

2.1. Simulation platform for composites

Composite structural behavior can be described numerically using finite element methods. The goal of finite element analysis is to find an approximated solution for a system of differential equations which describes physical processes. An approximated model is formulated for a complex geometry with assigned boundary conditions and loads. Different composite manufacturing process simulation steps require the same geometry. Otherwise, some simulations require often a different discretization and often also different element types, for example shells or solids, to perform the analysis. For material and process data conversions from one mesh to another one, a mapping algorithm is required. A common data format of a simulation model with mapping algorithms can increase a performance of data conversions into a structural analysis. The developed simulation platform provides open data format and mapping algorithms for improving structural simulations. Figure 1 presents an integration concept of the developed composite simulation platform.

The simulation platform concept intended to eliminate known limitations, for example only two-dimensional mapping and the limited parameter transfer. It requires developments in four different fields [12,13,14,15]: wrapper, common data format, mapping tool and material database. The wrappers are small scripts which translate software-specific output data into a common data format and generate input files for new simulations. The common data format is intended to easily exchange information between different analysis tools. The mapping tool translates information which is available on a finite element mesh onto a second finite element mesh and calculates additional properties based on provided data in a material database. The material database is intended to store simulation parameters required for the description of material behavior.

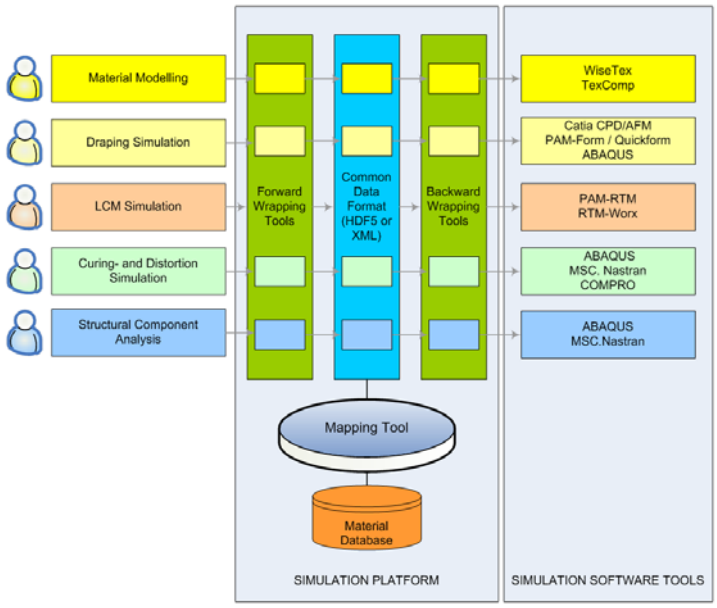

Figure 1. Simulation platform

The functionality of the mapping tool can be divided into the two operations: finite element mesh mapping and transfer of available data. The finite element mesh mapping: This step comprehends the preparation of source and target mesh for a search algorithm and the adjacent mapping by the search algorithm. The preparation of data consists of an adaption of the element type and the mesh layup. If one of the two meshes contains 3D elements, these elements are simplified to 2D elements. For that nodal information of the 3D element are projected to the middle plane and stored in nodes there. If the number of layup of source and target mesh differs, the number of layers of the source mesh is changed. So layers of a composite are summarized to the middle layer or in the other direction the middle layer gets duplicated and the created layers are moved into position. As a result every existing target layer is assigned to a source layer. For each of these groups a search algorithm will be applied. It locates the corresponding source element for each node in the target layer and thus builds a mapping between both meshes. Field Transfer: In a second step the information of the source mesh are transferred to the target mesh. This contains the following operations: Weighted calculation and transfer of nodal values from source elements to target nodes; Calculation of target element values from target node values; Interpolation and extrapolation of the values at integration points; Calculation of nodal or element values from additional properties based on newly obtained results or the linked material database. In the next sections we will discuss search algorithms and field transfer in details.

2.2. Search Algorithm

The method that transfers data from a source mesh *S* to a destination mesh *D* requires a search algorithm that identifies for each point in the destination mesh, the corresponding element in the source mesh and its closest location within this element. The point of the destination mesh can either be a node or an integration point within an element of mesh *D*. Such search algorithm can also be found in contact formulations for finite element solvers. Within the mapper application, the adopted search algorithm can be split in two steps: global and local search.

For a global search can be used following algorithms: master-slave [16], three-dimensional bucket sort [17, 18], algorithms based on a k-d-tree definition [8, 19] or on the Space Filling

Curve [20]. A first step of mapping search algorithm is a global search with the Bucket Sort algorithm [17, 18]. This algorithm contains the following 6 steps:

1. Find the maximum coordinates as well as the minimum coordinates in both source and destination meshes.

2. Calculate the characteristic length as 70% of the longest element edge length of all elements in the source mesh.

3. Calculate the number of buckets in each direction.

4. Calculate for all nodes in both meshes the bucket numbers.

5. Create for both meshes a list of nodes, which are inside the bucket. A node is assumed to be inside the bucket if the bucket numbers of the node are equal to the indices of the bucket.

6. Create for the source mesh a list of elements which are inside the bucket.

For a local search can be used algorithms based on a point-element testing [16], gap functions [21] or pinball [22] principles. The second step of mapping search algorithm is based on an algorithm proposed by Zhong and Nilsson [23] with a subsequent point-element testing.

2.3. Field Transfer

The data available in the source finite element mesh *S* is either stored at the nodes or is given for an integration point p in the elements of the mesh. The transfer of this data is handled in a different way: transfer of nodal or integration points data and relocation of data between node and integration point.

In case of so called nodal information the mapping algorithm is stated as

$$u^S(V) \rightarrow u^D(V)\,, \tag{1}$$

where u^S is a nodal field given in the source mesh and u^D the same field but given for the destination mesh. Both meshes cover the same domain V. The value of the destination field u_j^D at a given node n_j can be calculated using the shape functions N_i of the corresponding source element e^S according to

$$u_j^D = \sum_{i=1}^{N^e} N_i(\xi_j^S, \eta_j^S) u_i^S\,, \tag{1}$$

where the natural coordinates (ξ_j^S, η_j^S) indicate the position of the node n_j with respect to the source element e^S.

In case a value ϑ_i is available at the nodes n_i and must be transferred to the integration points g_j they can be simply calculated by using the element interpolation functions N_i and the natural coordinates $(\xi_{g_j}, \eta_{g_j}, \mu_{g_j})$ of the integration points g_j . For 2D elements this relation is written as

$$\vartheta_j = \sum_{i=1}^{N} N_i(\xi_{g_j}, \eta_{g_j})\vartheta_i \,, \tag{3}$$

whereas for 3D elements the third natural coordinate is taken into account:

$$\vartheta_j = \sum_{i=1}^{N} N_i(\xi_{g_j}, \eta_{g_j}, \mu_{g_j})\vartheta_i \,, \tag{4}$$

Figure 2 demonstrates that the mapping of values at integration points used also extrapolation functions.

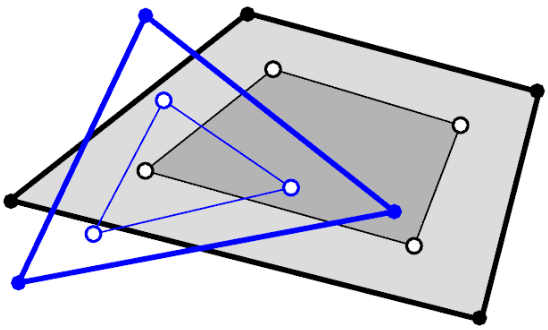

Figure 2. Transfer of data at integration points requires interpolation or extrapolation methods. Source mesh (blue) and target mesh (black).

Following data can be converted from a mesh to another mesh: fiber orientation, thickness, temperature and curing grad fields. The transfer of displacement and stress fields can cause unrealistic material behavior [24, 25]. In case of elasto-plastic material models a field transfer can violate constitutive relations of the target mesh. Mostly, only the displacement field is continuous for finite element approaches. Due to different material models we consider also an influence of thickness and fiber orientation distributions onto material characteristics.

3. Results

Due to information from engineering offices at aerospace and automotive companies (Airbus Group Innovations, Airbus Group Helicopters, DLR, Premium Airtec, Voith and BMW) in the year 2014 it was discovered that a most current needed interface is between draping simulation with ESI Group software and a structural simulation in SIMULIA Abaqus or MSC.Nastran. The considering results illustrate data transfer into a structural analysis with MSC.Nastran. The verification process was considered on an L profile and complex industry relevant parts. Hier, we present an L profile component manufactured by a laminate stacking sequence of eight plies oriented from top to bottom: [0/90/22.5/-22.5/45/-45/90/0]. We definite the manufacturing chain through draping and curing processes. The process simulations were implemented by ESI Group PAM-Form and ABAQUS tools. A generation of different meshes for each process was based on defined geometry in CATIA. Initially, the simulations were performed with nominal material and process parameters. Subsequently, resulting data were translated into the defined common data format STRUCTML. Then, the mapping tool was applied in order to perform a finite element data conversion. Finally, new input files with process parameter distributions were generated for improved simulations. The mapped fiber angle distribution induced stress distributions which are important for a tooling design. Figure 3 demonstrates the data transfer from draping mesh onto a structural mesh.

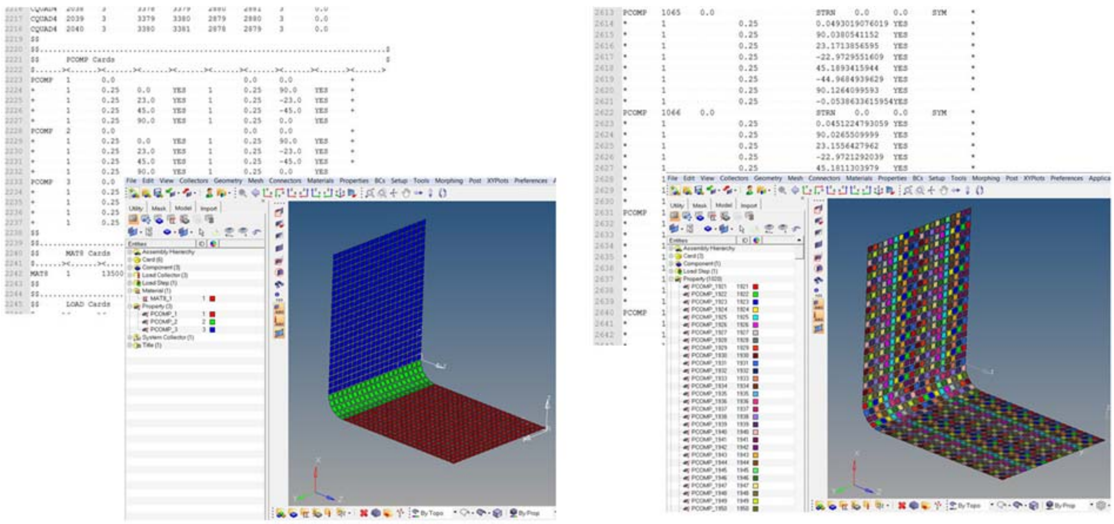

Figure 3. L profile 2D2D: To the left. MSC.Nastran nominal model. To the right. MSC.Nastran model with fiber orientation distribution from a draping simulation.

4. Conclusion

The data interface concept for composite materials with mapping algorithms has been presented. It enables a combination and transfer of results from mostly software packages for manufacturing simulations into a structural analysis. The developed simulation platform uses mapping algorithms for a data modification. These algorithms based on a mesh-independent approach. An adopted as-build structural analysis follows process and material parameter, and affects criteria for a tooling design. Such as-build analysis for composites increases an interaction between manufacturing and design, and minimizes manufacturing cycles. The defined common data format allows storing manufacturing and simulation results for an improved simulation or validation process. However, the transferred data can violate constitutive relations in finite element models. In the next step we consider an approach for an integration of transferring data into material models.

References

[1] M. E. Williams, G. R. Consolazio and M.I. Hoit. Data storage and extraction in engineering software using xml. *Advances in Engineering Software*, 36(11-12):709-719, 2005

[2] G. Moita, P. Almeida, D. Oliveira and M. Pinheiro. A data exchange neutral format for finite element codes using an intelligent element classification. *Revista CIATEC – UPF*, 3(1):48-64, 2011

[3] Anaglyph Ltd. MSC.Laminate Modeler. www.anaglyph.co.uk, 2013.

[4] BETA CAE Systems USA Inc.. *ANSA v13.1.3 Edition*. http://www.beta-cae.gr, 2013.

[5] e-Xtream Engineering. *DIGIMAP Release Notes*. 4th Edition 2012.

[6] ANSYS. ANSYS Documentation. 14th Edition 2013.

[7] N. G Bourago and V. N. Kukudzhanov. A review of contact algorithms. Izv. RAN, MTT, 1:45-87, 2005.
[8] D. M. Mount and S. Arya. http://www.cs.umd.edu/~mount/ANN/, 2014
[9] G. B. Wright. Radial Basis Function Interpolation: Numerical and Analytical Developments. *PhD thesis*, University of Colorado, 2003.
[10] D. Shepard (1968). A two-dimensional interpolation function for irregularly-spaced data. In *1968 ACM National Conference*, pages 517–524, 1968
[11] T. Belytschko, J. M. Kennedy and J. I. Lin. Three-dimensional penetration computation. In *Struct. Mech. React. Technol*., volume v1, pages 83-87, 1987.
[12] B.V.D. Broucke. Implementation ideas. *Technical report*, Airbus Group Innovations, 2011.
[13] Wille, Hein, Horn, A. Knote, M. Opitz, N. Mayer, J. Prowe, M. Calomfirescue, J. Balvers and H. Apmann. Development, Implemenetation, and Demonstration of a Composite Tool Chain for Concurrent Engineering. In *NAFEMS Seminar*, volume v1, 2012.
[14] T. Wille, R. Hein, N. Mayer and J. Prowe. Composite Process Chain "As-Build". In *CCeV: Faser-Verbund-Wekstoff*, volume v1, 2012.
[15] N. Mayer and J. Prowe. Data Transfer in the Integrated Tool Chain for the Simulation of Composites. In *AIRTEC, 7th International Conference "Supply on the wings"*, volume v1, 2012.
[16] J. O. Hallquist, G. L. Goudreau and D. J. Benson. Sliding interfaces with contact-impact in large-scale Lagrangian computation. *Comp. Meth. Appl. Mech. Engng.*, 51(1-3):107-137, 1985.
[17] T. Belytschko and J. I. Lin. A three-dimensional impact-penetration algorithm with erosion. *Computers and Structures*, 25(1):95–104, 1987.
[18] D. J. Benson and J. O. Hallquist. A single surface contact algorithm for the post-buckling analysis of shell structures. *Computer Methods in Applied Mechanics and Engineering*, 78(2):141–163, 1990
[19] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman and A. Y. Wu. An Optimal Algorithm for Approximate Nearest Neighbor Searching in Fixed Dimensions. *Journal of the ACM*, 45(6): 891–923.
[20] R. Diekmann, J. Hungershofery, M. Lux, L. Taenzer and J.-M. Wierumy. Using Space Filling Curves for Encient Contact Searching. In *16th IMACS World Congress*, volume v1, 2000.
[21] G. Hirota, S. Fisher, A. State, C. Lee and H. Fuchs. An Implicit Finite Element Method for Elastic Solids in Contact. In *SIGGRAPH*, volume v1, 2001.
[22] T. Belytschko and M. O. Neal. Contact-Impact by the Pinball Algorithm with Penalty and Lagrangian Methods. *Numerical Methods in Engineering*, 31(3):547-572, 1991.
[23] Z.-H. Zhong and L. Nilsson. Contact-impact algorithms on parallel computers. *Nuclear Engineering and Design*, 150(2-3):253–263, 1994.
[24] A. Bucher, A. Meyer, U.-J. Goerke and R. Kreissig. A comparison of mapping algorithms for hierarchical adaptive FEM in finite elasto-plasticity. *Computational Mechanics*, Springer-Verlag, 2006
[25] A. Bucher, U.-J. Goerke and R.Kreissig. Development of a generalized material interface for the simulation of finite elasto-plastic deformations. *Solid Structures*, 38(52):9423-9436.