FACTA UNIVERSITATIS
Series: **Mechanical Engineering** Vol. 11, N° 1, 2013, pp. 1 - 12

MODELING OF PIEZOELECTRIC SMART STRUCTURES BY IMPLEMENTATION OF A USER DEFINED SHELL FINITE ELEMENT

UDC 624.01+531;519.6

Tamara Nestorović[1], Somu Shabadi[1], Dragan Marinković[2,3], Miroslav Trajkov[1]

[1]Ruhr Univerisität Bochum, Germany
[2]Institut für Mechanik, TU Berlin, Germany
[3]Faculty of Mechanical Engineering, University of Niš, Serbia

Abstract. *In this paper, the nine node piezoelectric shell element developed in [1] has been implemented in the commercial FE software ABAQUS. In comparison with our previous work [2], the application has been extended to modeling of arbitrary thin walled structures also with complex geometries, whereby the automated mesh generation has been achieved by developing a Python based interface for meshing procedure. In order to be able to perform the post processing, special adaptation of the user element has to be performed for visualization purposes. The implemented element regards the piezoelectric thin layers polarized in the thickness direction and it is based on the e31 piezoelectric effect. It is also shown that this biquadratic nine node element based on degenerated shell approach is less prone to locking effects and more suitable for implementation with curved structures.*
Through several examples the accuracy of the implemented user defined shell element as well as of the Python mesh generator has been demonstrated, along with the possibilities for post processing. Meshing the structures with the nine node user element is not third party software dependent.

Key Words: *User Element, Piezoelectric Shell Element, ABAQUS, Piezoelectric Active Structures*

1. INTRODUCTION

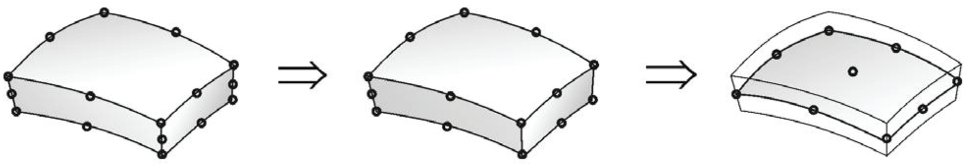
Piezoelectric composite structures modeling plays an important role in the overall design procedure for smart structures and systems. A reliable model enables optimization in the early development phases as well as simulation of the structural behavior and prediction in

Received February 06, 2013
Corresponding author: Tamara Nestorović
Ruhr Univerisität Bochum, Universitätsstraße 150, D-44801 Bochum, Germany
E-mail: tamara.nestorovic@rub.de

different 'what-if" scenario cases. The available tools improvement and the fulfillment of special user requirements are the main motivations for this work. This is also the underlying idea behind the user element implementation carried out in [2], which has motivated the authors to contribute to the field of modeling of piezoelectric structures and systems. For this purpose, a new shell type finite element has been developed by one of the authors. There has been an increase in the use of piezoelectric materials in smart composite structures. In order to improve the performance and for optimization purposes, which play an important role in design process and modeling, a finite element (FE) based approach has been considered. It is also necessary to model these kinds of structures with appropriate finite elements which promise a low computational effort along with an accurate numerical prediction of the structure's behavior [4]. Although some commercial FE software packages include finite elements which can be used for modeling piezoelectric properties, an appropriate solution for thin walled shell structures has not been completely implemented yet. Piezoelectric finite elements available in commercial FE software like ANSYS and ABAQUS are solid elements and application of these elements can often be computationally very costly [2]. Moreover, coupling solid piezoelectric finite elements with shell type basic structures can cause meshing and numeric problems and it often leads to a high computational effort. Therefore, numerous researchers have worked on developing a 2D finite element which offers satisfactory accuracy and less computational effort. In this regard, ACShell9 – *a nine node biquadratic active composite shell element* is developed by one of the authors in [1]. The lack of commercially available piezoelectric shell elements and the drawbacks of combining commercial solid piezoelectric elements with the shell-type structures have triggered off our motivation to implement this user-defined piezoelectric shell element into the commercial FE software package in order to perform the behavior analysis of thin-walled piezoelectric composite structures under different loading and boundary conditions. Since the ABAQUS provides an interface where the user can define a new finite element, it is taken into consideration for user implementation [2]. In the present work, the application has been extended to perform meshing using the new shell element [1] and post-processing of user-defined elements for visualization. Comparison between the performance of the degenerated shell element ACShell9 and that of the ABAQUS standard piezoelectric solid element has also been carried out.

2. A SHORT OVERVIEW OF THE ACSHELL9 ELEMENT

The ACShell9 element developed by one of the authors of this paper [1] will be briefly presented here. For more details about the element description, interested readers are referred to literature, e.g. [1, 3]. ACShell9 belongs to the degenerated shell element family, which was first developed by Ahmed et al. [3] and it is based on Mindlin-Reissner theory. This element involves a less computational effort than in the case of 3D solid elements and it can be used to model shells of various thickness. Fig. 1 shows the degeneration process from 20 node 3D solid element, via 8-node element to 9-node degenerated shell element [1]. Ahmad [3] has actually developed the 8-node element. It has been argued in [1] that the absence of 9th node results in shear and membrane locking and, hence, the addition of the 9th node has been considered. The ACShell9 element includes both mechanical and electrical degrees of freedom – 54 mechanical degrees of freedom, 3 displacements and 3 rotations at each of the 9 nodes and additionally N_{pe} electrical degrees of freedom depending on the number of piezoelectric layers [1]. The electrical degree of freedom assigned to each piezoelectric layer is the difference of electric potentials between electrodes of the upper and the lower surface of piezoelectric layer.



20 – Node 3D Solid Element | 16-Node 3D Solid Element (Nodes on Mid-Surface Eliminated) | 9-Node Degenerated Shell Element

Fig. 1 Degeneration process

There are three different coordinate systems in the formulation of the element as shown in Fig 2. Those are as follows:

1. The global Cartesian coordinate system (x, y, z)
2. The natural coordinate system (r, s, t) with $-1< r, s, t < +1$
3. The local-running Cartesian system (x', y', z')

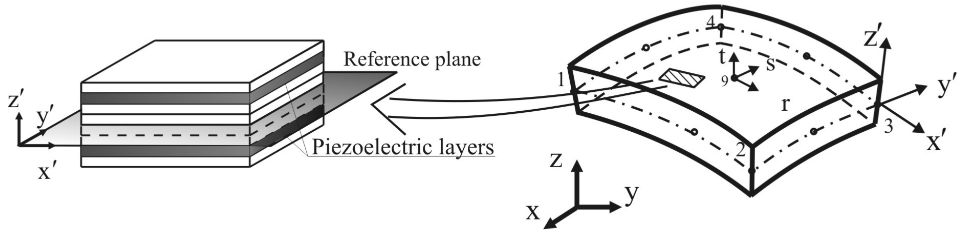


Fig. 2 Equivalent layer approach for multilayer material and coordinate systems

The element formulation utilizes the equivalent layer approach as shown in Fig. 2, where some of the layers are piezoelectric layers embedded between the passive ones. The layer-wise analytical integration is applied in the thickness direction while the numerical integration is performed for the in-plane directions. Each of these layers could be made of orthotropic materials, thus the need for a local coordinate system (x',y',z') becomes prominent. One of the basic requirement for the local coordinate system is that one of its axes (z-axis) should be perpendicular to the reference surface of the structure [1]. Numerical integration and its order play an important role in handling the shear and membrane locking among the shell elements. Since the ACShell9 element is based on the first-order shear deformation theory, it would not cause such severe locking but it would suffer sub-optimal convergence when the thickness of the element tends to zero [1]. The sub-optimal convergence is handled by reduced integration. The formulation of the stiffness involves 5th order polynomials. Thus, the exact evaluation can be achieved by applying at least 3 integration (Gauss) points in each direction. Prathap [4] has elaborated the properties of the reduced integration technique when used in combination with the 9-node degenerated shell element. The technique also offers a high numerical efficiency. Ahmed et al. [3] have found that the 2x2 rule is the optimal one for a general double-curved geometry of the element but it might also give rise to an hour-glass mode, which however cannot propagate through the FE assemblage in practical applications. It has been proposed in [1] that both the full (3×3) and the reduced (2×2) technique should be given as a choice to the user [1, 4, 7].

The rising need for modeling the piezoelectric structures with the finite element procedure imposes the implementation of the developed element in the commercial finite element software package. It has been demonstrated in [2] that the ACShell9 element can be used for modeling flexible structures when incorporated in the ABAQUS as a user-defined element subroutine. Since the ACShell9 is developed for analysis of any arbitrary shaped shell structures, it becomes necessary to extend the functionality of the application for meshing arbitrary shaped thin-walled structures with the 9-node quadrilateral element. The next section explains the implementation of the Python mesh generator.

3. PYTHON MESH GENERATOR IMPLEMENTATION

The ABAQUS scripting interface, which is an extension of the Python object-oriented scripting language, provides a good platform for automation of part generation, material assignment, meshing, applying boundary conditions and submission of analysis jobs. The motivation behind the development of the Python mesh generator is that the Mesh module of the ABAQUS graphical user interface does not support the ABAQUS standard shell element – S9R5. S4R5 and S8R5 are the only two quadrilateral elements which are supported by the Mesh module of the ABAQUS. Therefore, the structure is modeled and meshed with S8R5 from the ABAQUS/CAE. Upon application of appropriate boundary conditions, the temporary input file is exported. Fig 3 illustrates how the Python Generator takes a temporary input file generated from ABAQUS/CAE as its input. Python mesh generator calculates the 9th node by taking the average of the *x*, *y* and *z* coordinates of the 4 mid-side nodes of the S8R Quad Element from the temporary input file:

$$x_9 = \frac{1}{4}\sum_{i=5}^{8} x_i,\ \ y_9 = \frac{1}{4}\sum_{i=5}^{8} y_i,\ \ z_9 = \frac{1}{4}\sum_{i=5}^{8} z_i, \tag{1}$$

where (x_9, y_9, z_9) are the coordinates of the 9th node of the S9R5 element and (x_i, y_i, z_i), i=5÷8 are the coordinates of the mid-side nodes of the S8R element. The Python mesh generator also extracts nodes, elements and boundary conditions from the temporary input file and stores them into respective comma separated files. These are included in the user element input file for further analysis.

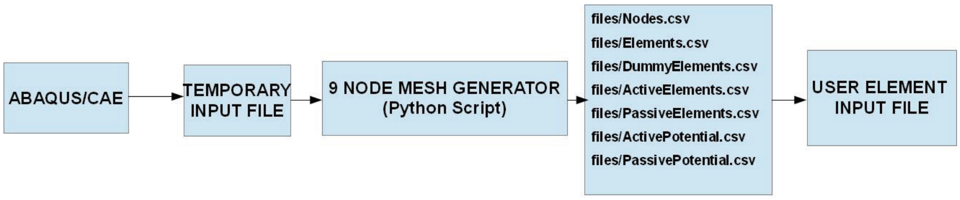


Fig. 3 Input and output of the 9-node mesh generator

4. Numerical Examples

In [2], the accuracy of the ACShell9 element has been demonstrated in various cases of inverse (actuator case) and direct (sensor case) piezoelectric effect in analysis. The implementation is only limited to a bimorph piezoelectric beam. In the present work, the numerical examples are carefully chosen with two main aspects: 1) to demonstrate the accuracy of the implemented ACShell9 user defined element in the case of static analysis of thin-walled structures with more than two layers in ABAQUS; and 2) to demonstrate the accuracy of the developed 9-node Python mesh generator as well as the post-processing results of the user-defined element.

4.1. Active beam structure

The following example is proposed in [1]. In this example a pair of piezopatches is bonded to the surfaces of a clamped beam and the voltage of 100V is applied over the electrodes of the patches. The patches are oppositely polarized. Hence their activation produces bending moments uniformly distributed over the edges of the patches. The piezoceramic patches have the thickness of 0.2 mm and they are made of PIC151 material, while the beam is made of aluminum and has the thickness of 0.5mm. The geometry of the beam and the material properties are given in Fig. 4.

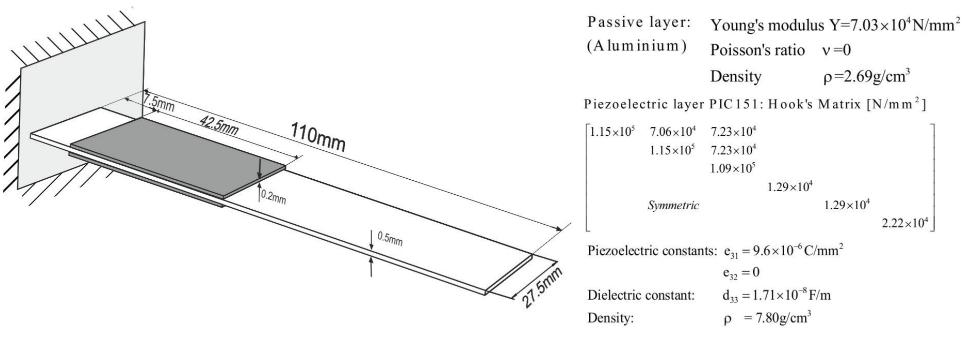


Fig. 4 Geometry and material properties of the active beam [1]

One of the objectives of the given example is to prove that the currently implemented user-defined subroutine can perform a linear static analysis on the cantilever beam where a part of the surface is covered with piezoelectric patches. Fig. 5 shows that the user defined active element – UEL 1 consists of 3 layers, while the UEL2 consists only of a single passive layer. For all the user-defined active elements, the material properties, thickness of both passive and active layers are passed on to the subroutine through UEL properties and, similarly, for the UEL passive element, only the properties of the passive layer are passed on to the subroutine in the input file.

The main objective of the example is to compare the deflections obtained with the ACShell9 user element to those obtained with the ABAQUS standard piezoelectric element – C3D20E.

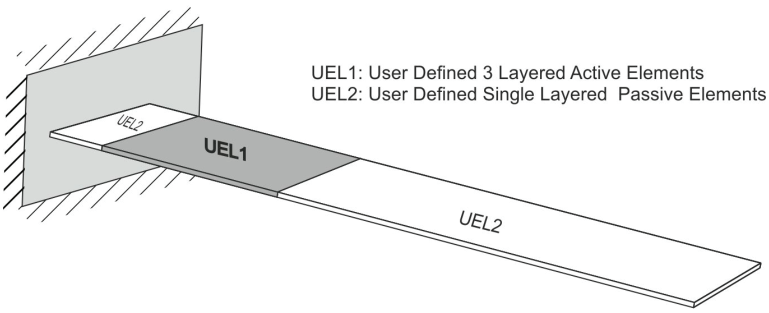


Fig. 5 Active beam user defined elements

Fig. 6 validates the implemented subroutine with the ABAQUS standard piezoelectric element – C3D20E and also the results published by Marinković [1] for the ACShell9 element. The obtained results are visualized by means of the ABAQUS standard Element (S9R5) with negligible stiffness and the same nodes and degrees of freedom as the implemented element, Fig. 3. The transverse displacements of the cantilever beam are caused by the bending moments generated along the edges due to the electric excitation of the oppositely polarized piezoelectric patches.

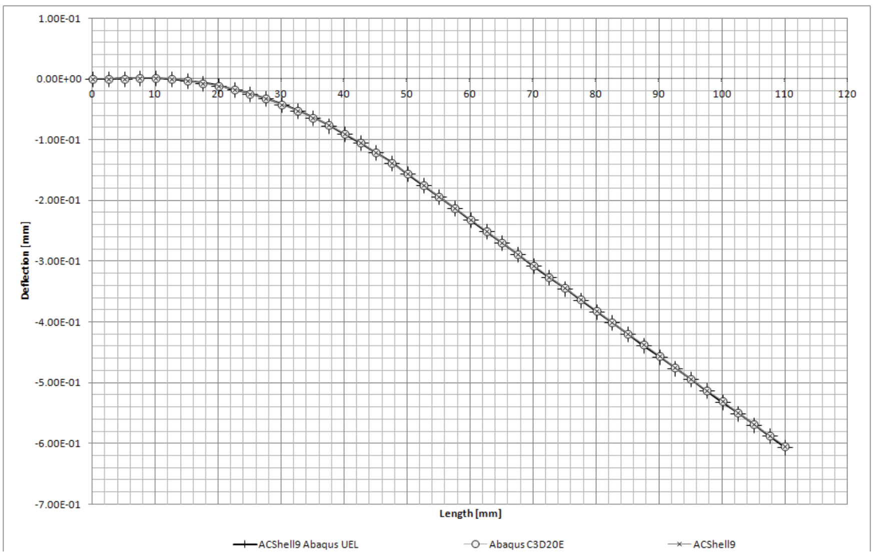
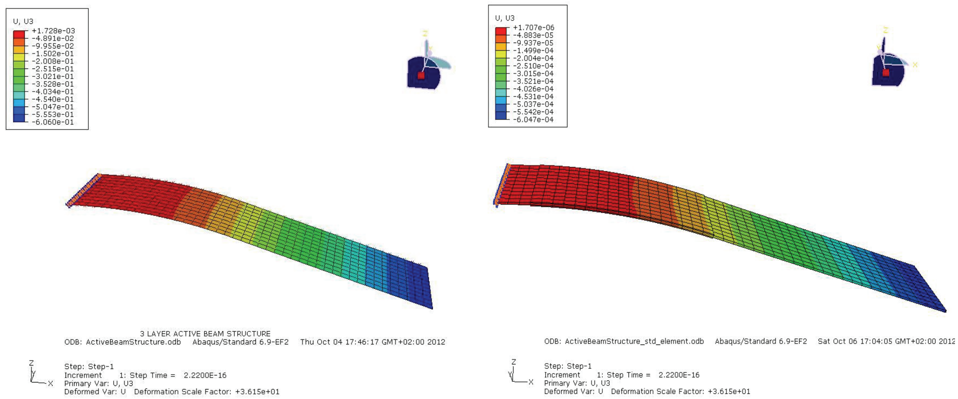


Fig. 6 Static deflection of the beam



a) using UEL SUBROUTINE

b) using ABAQUS Standard Element – C3D20E

Fig. 7 FE model of active beam structure

4.2. Cantilevered cylindered shell

This example aims at demonstrating the Python mesh generator accuracy by comparing it with the GiD preprocessor for curved shaped structures. The structure represented in Fig. 8 is made of aluminum and has the thickness of 0.5mm. Both the top and the bottom surfaces of the structure are completely covered by piezoelectric layers made of PIC151 material. The thickness of the piezoelectric layers is 0.2 mm. The actuation is achieved by applying 100V to each of the piezoelectric layers. The material properties are those shown in Fig. 4.

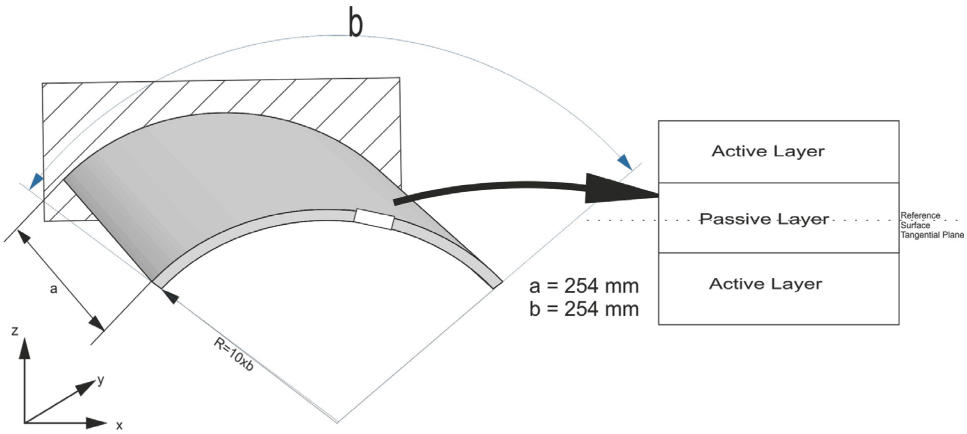


Fig. 8 Cantilevered composite shell

Fig. 9 shows that the results obtained using the Python 9-node mesh generator are in a good agreement with those from the GiD Preprocessor. This shows that, even for a coarse mesh, the 9-node mesh generator yields good results in modeling shallow shells. There would be little discrepancy when compared to the mesh generated using the standard mesh generating software like GiD, which uses much more sophisticated algorithms and the generated 9th node is in that case exactly inside the actual domain of the geometry. But in the implemented Python mesh generator, the interpolated 9th node may not lie on the surface of the geometry, especially when the surface has a sharp curvature. In such cases, the element size should be made very small. In the following example, it is shown that this drawback can be overcome by using a sufficiently fine mesh for arbitrarily shaped structures with sharp curvatures. It can also be deduced from the graph that the deflections obtained from the user-defined subroutine are comparable and rather close to the deflections obtained using the Abaqus-C3D20E elements. This actually proves that the implemented "ACShell9" element in ABAQUS can be used for performing static analysis on arbitrary shaped multilayered active structures. It also proves that the fine mesh is needed for more accurate results.

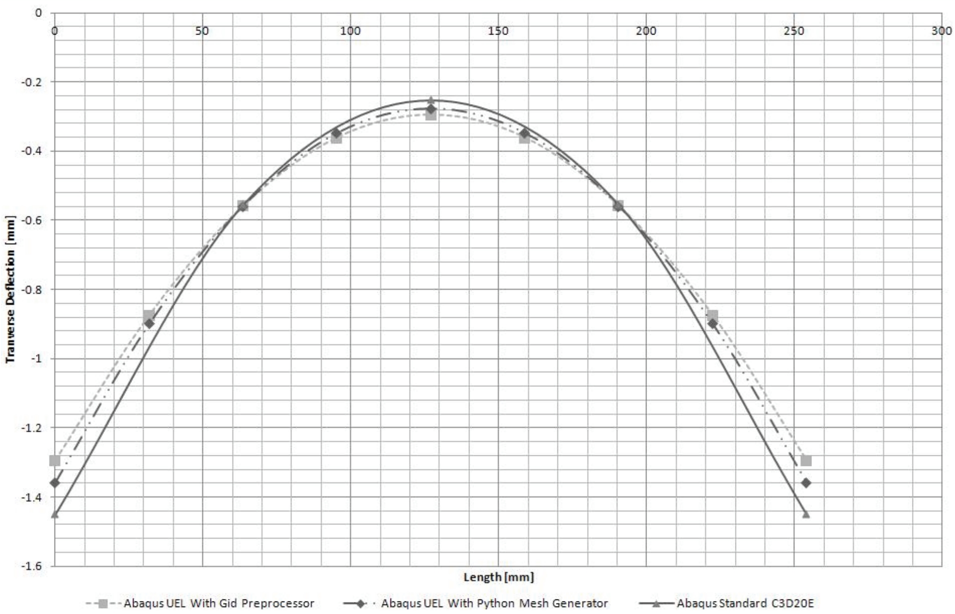


Fig. 9 Transverse deflection of points lying on the shell's curved free edge

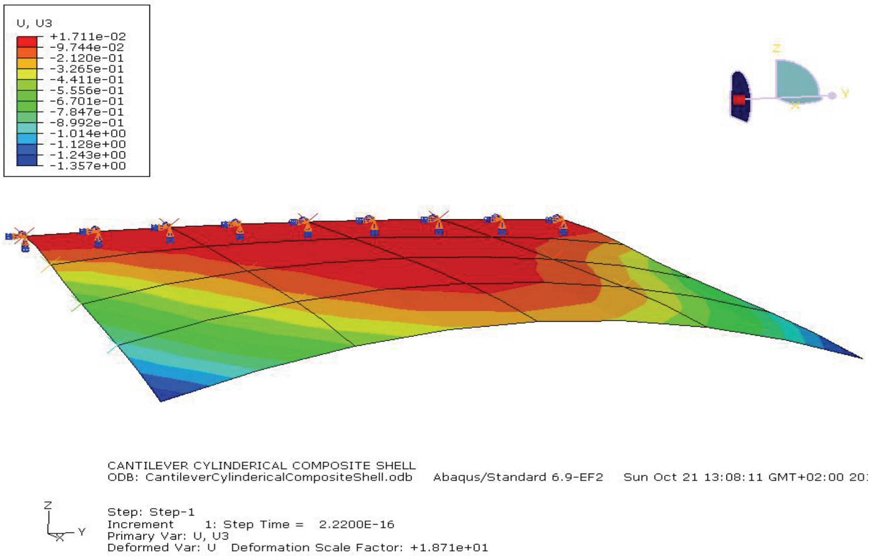


Fig. 10 Post-processing of cantilevered composite shell

4.3. Funnel structure

This particular numerical example demonstrates the accuracy of the ABAQUS Python 9-node mesh generator when implemented to arbitrary shaped curved structures. In this example, the funnel like structure is made of aluminum and has the thickness of 0.5mm. It is covered on the top and the bottom surfaces by piezoelectric layers made of PIC151 material. The thickness of the piezoelectric layers is 0.2mm. The material properties are the same as shown in Fig. 4. The geometry is represented in Fig. 11. The lower base is entirely fixed and the upper surface is free. The electrical potential of 100V is applied to each of the top and of the bottom piezoelectric layers.

It can be clearly noticed from Fig. 13 that the results obtained from the Python 9-node mesh generator are in a good agreement with the mesh generated using GiD pre-processor even for the structures with sharp curvatures. This validates the usability of the Python mesh generator shell-type curved structures.

Following are the limitations of the Python 9-node mesh generator:

1. The ABAQUS Python is very slow comparing to the ABAQUS implicit mesh generator.

2. The 9-node mesh generator does not directly mesh the model. The model has to be first meshed using ABAQUS standard 8-noded shell element (S8R) from ABAQUS Graphical User Interface (GUI).

3. The 9-node mesh generator might not yield a sufficiently accurate position of the 9^{th} node when arbitrarily shaped structures are discretized by a coarse mesh. The quality of such a mesh might be deficient for both the analysis and the post-processing.

Advantages of the Python 9-node mesh generator are as follows:

1. The accuracy of the 9-node Python mesh generator improves significantly with the mesh refinement.
2. This 9-node Python mesh generator along with ABAQUS macros to generate model could be used for automation and optimization by re-modeling and re-meshing by checking some of the parameters after analysis.
3. There is no third party software dependency to mesh the structure with the 9-node shell element.

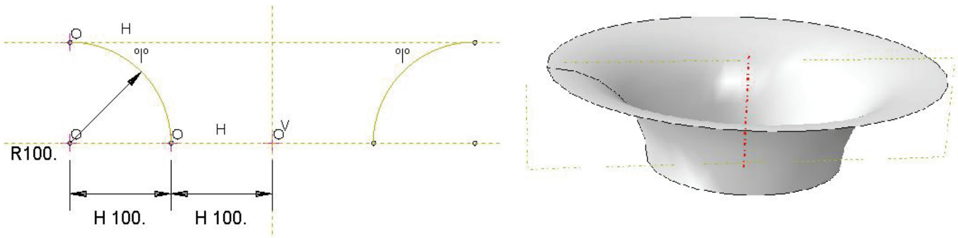
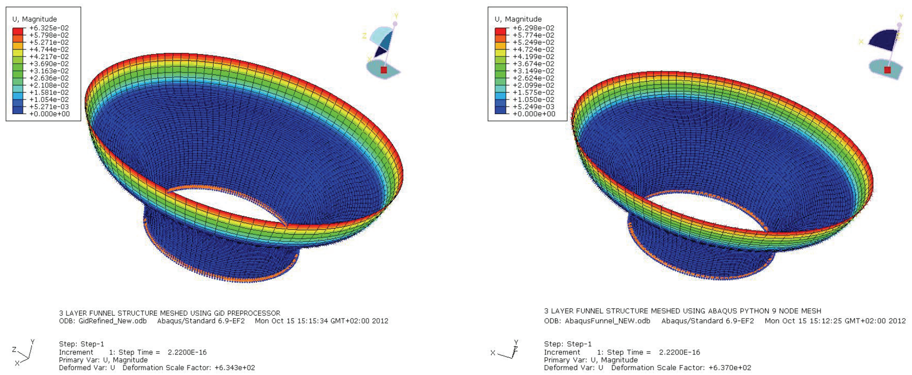


Fig. 11 Geometry of funnel structure



a) meshed using GiD Preprocessor   b) meshed using Python mesh generator

Fig. 12 Displacement magnitude plot of the funnel structure

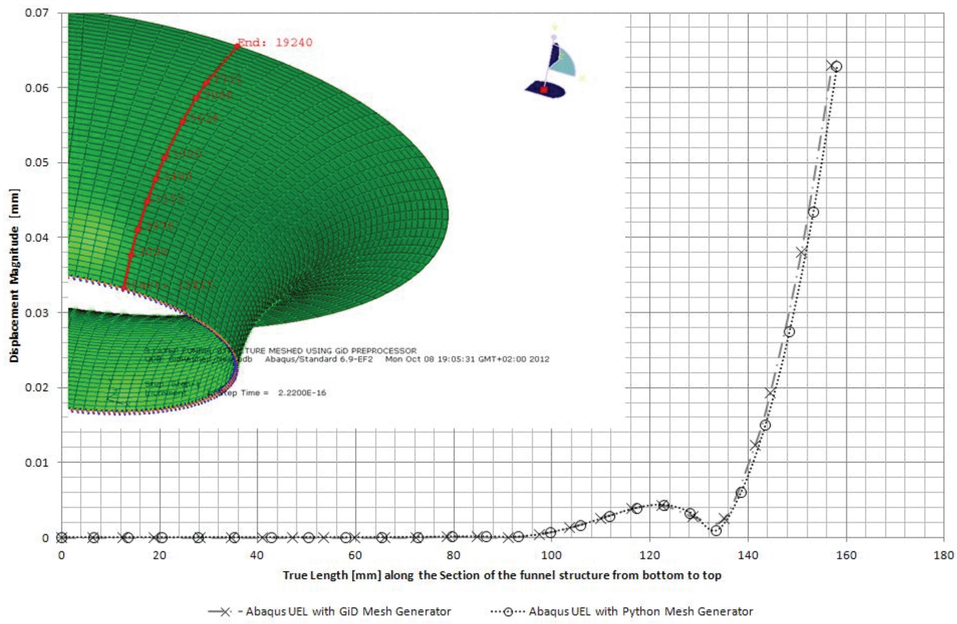


Fig. 13 Displacement [mm] along true path of section

5. Conclusions and Outlook

The active beam structure and active plate structure examples highlight the usability of the user-defined active elements in conjunction with the user-defined passive elements in the ABAQUS. The obtained results from the ACShell9 are validated by meshing the equivalent model with the ABAQUS standard piezoelectric element - C3D20E. The cantilevered cylindrical shell example and funnel structure example highlight the usability of the Python mesh generator for the arbitrarily shaped curved structures. The 9-node Python mesh generator has been validated using the GiD pre-processor. Finally, it can be concluded that the present work can be used for performing a static analysis on the arbitrary shaped thin walled composite structures. The Python mesh generator script can be used for meshing any arbitrary thin-walled structure by using the 9-node quadrilateral shell element.

References

1. Dragan Marinković, *A new finite composite shell element for piezoelectric active structures*, Fortschritt-Berichte VDI, Reihe 20, Nr. 406, VDI Verlag, Düsseldorf 2007
2. Tamara Nestorović, Dragan Marinković, Gurudutt Chandrashekar, Zoran Marinković and Miroslav Trajkov, *Implementation of a user defined piezoelectric shell element of analysis of active structures*, Finite Elements in Analysis and Design, 52, 2012, 11-22
3. Sohrabuddin Ahmad, Bruce M. Irons, O. C Ziekieqicz, *Analysis of thick and thin shell structures by curved finite elements*, Int. Journal for Numerical Methods in Engineering, Vol. 2, 1970, 419-451
4. Gangan Prathap, *The finite element method in structural mechanics*, 1993, Kluwer Academic Publishers, Netherlands

5. Gabbert Ulrich, A. Görnandt, Heinz Köppe*, Benchmark Problems for the analysis of piezothermoelastic smart structures,* NAFEMS seminar, Finite element applications for adaptive structural elements, Magdeburg, 1998
6. V. Balamurugan, S. Narayanan, *Active vibration control of smart shells using distributed piezoelectric sensors and actuator*s Smart Mater. Struct., 10, 2001, 173-180
7. Dragan Marinković, Heinz Köppe, Ulrich Gabbert, *Numerically efficient finite element formulation for modeling active composite laminates*, Mechanics of Advanced Materials and Structurs, 13(5), 2006, 379-392.

MODELIRANJE PIEZOELEKTRIČNIH AKTIVNIH STRUKTURA POMOĆU KORISNIČKOG KONAČNOG ELEMENTA TIPA LJUSKE

U ovom radu piezoelektrični element tipa ljuske sa devet čvorova razvijen u [1] implementiran je u okviru komercijalnog softvera konačnih elemenata ABAQUS. U poređenju sa našim prethodnim radom [2] proširili smo primenu na modeliranje tankozidnih struktura proizvoljno kompleksnih geometrija, pri čemu je automatsko generisanje mreže realizovano putem interfejsa zasnovanog na Pythton kodu. U cilju sprovođenja postprocesiranja izvršena je posebna adaptacija korisničkog elementa radi vizualizacije rezultata. Implementirani korisnički element tretira tanke piezoelektrične slojeve koji su polarizovani u pravcu debljine i zasniva se na e31 piezoelektričnom efektu. Takođe je pokazano da je ovaj bikvadratni element sa devet čvorova zasnovan na pristupu degenerisane ljuske manje podložan loking efektu i da je prikladniji za primene na zakrivljenim strukturama.

Na nekoliko primera potvrđena je tačnost implementiranog korisničkog elementa tipa ljuske, i generatora mreže zasnovanog na Python kodu, kao i mogućnost postprocesiranja. Generisanje mreže pomoću ovog korisničkog elementa sa devet čvorova nije uslovljeno primenom nekog eksternog softvera.

Ključne reči: *korisnički element, piezoelektrični element tipa ljuske, ABAQUS, piezoelektrične aktivne strukture*