OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG FACULTY OF COMPUTER SCIENCE

Project Proposal for Digital Engineering Projects

Project Topic:	Extending a finite-element-implementation for further application cases
Project abbreviation:	
Institute/ Chair/ Research Group:	Chair of Computational Mechanics Institute of Mechanics Faculty of Mechanical Engineering
Advisors:	Prof. Dr.-Ing. Daniel Juhre Dr.-Ing. Fabian Duvigneau M.Sc. Lukas Maurer
Preferred group size:	1-4
Desired project period:	6 months
Required/Desired knowledge:	• Good programming skills • Basic understanding of the finite element method (FEM)

Is any external affiliation involved (e.g., industrial partner, affiliated institute)?

yes ☐ no ☒

Which one(s)?

<u>Project Description:</u>

General description:

We are solving many different engineering problems in different fields with the help of the finite-element-method (FEM). For some special tasks (beyond the scope of standard commercial FE-software) we use our own FE-implementations. In this project an existing tool (Matlab code) should be extended for further application cases. The individual tasks are diversified and are related to geometry approximation, boundary conditions, element types, large deformations and nonlinear material behavior. Our code is able to execute both finite-element- and finite-cell-simulations. The latter method is based on the FEM, but uses a non-geometry-conforming discretization as well as higher order shape functions.

We are hoping that we could arouse your interest. We cannot explain everything here in detail, so please just contact us, if you are interested to work on our topic.

Please note, that the overall project is designed in a modular fashion. Consequently, it is possible to work on this project with one, two, three or four persons. Naturally, the amount of work will be adapted/reduced, if the team is smaller.

Project goals:

- *The overall aim is one code (written in Matlab) including the developed parts of all group members. The combination of all separate codes to one code is the task of all the group members. A version control system (for instance, git) has to be used.*

- *1st Person: Implementing shell elements and non-linear material laws*
    - *Adding linear & quadratic shell elements to the existing FE-code (only important for the FEM part)*
    - *Implementing standard non-linear material laws, which then can be used in the FEM and FCM part in an identical way:*
        - *Neo-Hookean*
        - *Mooney-Rivlin*
        - *Arruda-Boyce*
        - *Ogden*
    - *Documentation*

- *2nd Person: Implementing large deformations and isoparametric geometry description*
    - *Implementing large deformations (finite strains), which then can be used in the FEM and FCM part in an identical way*
    - *Adding isoparametric geometry description for already implemented higher order elements (only important for the FEM part)*
    - *Documentation*

- *3rd Person: Improving the already implemented octree decomposition versions due to efficiency*
    - *Increasing the efficiency of the octree decomposition*
    - *Increasing the efficiency of the Boolean octree decomposition*
    - *Further code optimization*
    - *Documentation*

- *4th Person: Introducing Dirichlet boundary conditions for the FCM part of the code by the penalty and Nitsche method*
    - *Extending a penalty realization (already working for linear elements) for higher order elements*
    - *Implementing the Nitsche method for realizing boundary conditions in the FCM*
    - *Documentation*