FCM Toolbox for MATLAB

Students: Mohamed Elhaddad, Raul Espinoza, Haoyue Hu, Aurélien Florian Joly, Chenshen Wu, Patrick Zerbe

Supervisors: Tino Bog, Nils Zander

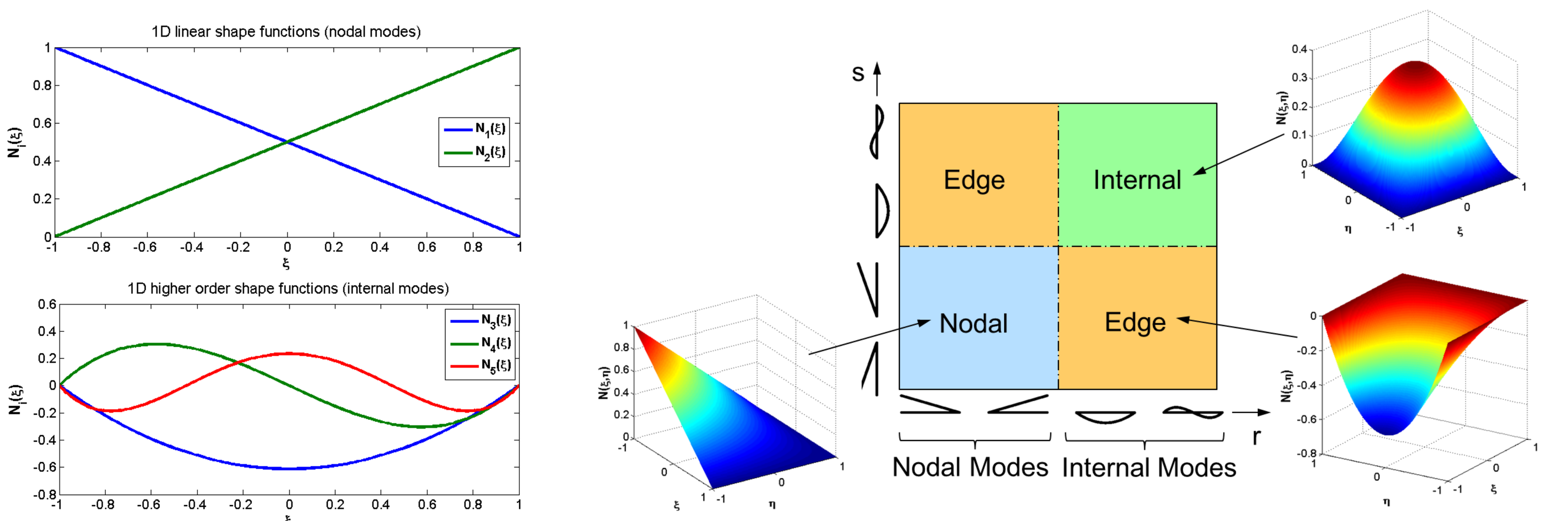
Abstract

An object-oriented Finite Cell Method (FCM) toolbox for MATLAB was designed and implemented to solve linear elasticity problems from solid mechanics. In classical Finite Element Method (FEM) the geometry is discretized directly, which is a problem in case of complex geometries. The FCM overcomes this problem by embedding the physical domain into a fictitious domain. The implementation is based on high-order FEM (p-FEM) with hierarchic shape functions. For the FCM an adaptive integration scheme and a weak treatment of boundary conditions is used.

p-FEM Theory

High-order FEM (p-FEM) is based on the idea to choose polynomials of higher order as ansatz functions. In order to achieve convergence, the polynomial degree is increased, while the mesh is kept the same. The advantage of p-FEM compared to h-FEM lies in its convergence properties. For numerical problems with smooth solutions, an exponential rate of convergence is obtained. In fact, the asymptotic rate of convergence of a uniform p-extension for linear elliptic problems is always faster than or equal to the convergence rate of a uniform h-extension [1].

The ansatz space is built with hierarchic shape functions, i.e. all the lower order shape functions are contained in the higher order basis. The 1D ansatz space is given by the linear shape functions (nodal modes) and the higher order shape functions (internal modes) using the Legendre polynomials. The shape functions for 2D and 3D ansatz spaces are constructed using the tensor product of 1D hierarchic shape functions. In 2D we distinguish between nodal modes, edge modes and internal modes (see figure below). In 3D there are additionally face modes, which are defined for each face and take the value zero over all other faces.

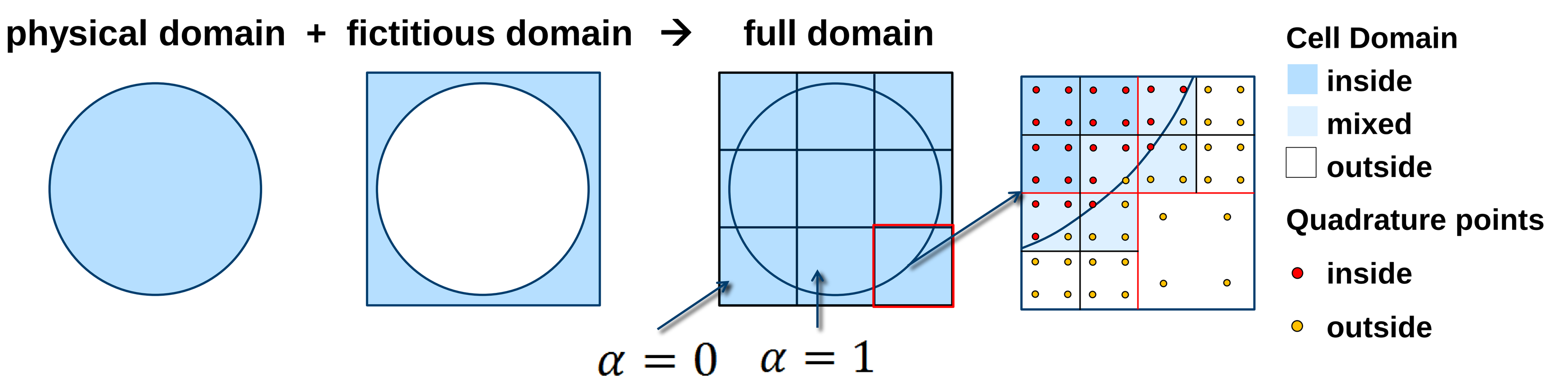


FCM Theory

Domain Description

In the Finite Cell Method (FCM) the physical domain is embedded in a fictitious domain to create a domain with simpler shape. This domain is then discretized with a Cartesian grid. A parameter α is introduced to indicate, if a point (e.g. integration point) belongs to the physical domain ($\alpha = 1$) or the fictitious domain ($\alpha = 0$). The material parameters (Young's modulus and density) are scaled with α such that in the physical domain the original values are kept, whereas in the fictitious domain both values become zero. Due to numerical reasons α is chosen to be very small (e.g. 1e-10) to avoid solver problems.

The original boundary value problem is recovered on the integration level. To this end, elements are recursively subdivided into cells, if they are cut by a domain boundary. This leads to a fine cell mesh in the area of domain boundaries and a coarse cell mesh in areas with one domain only. The cells are just used to numerically integrate their specific area. So the integration error due to material discontinuities is localized to the cells where the domain changes. The adaptive cell refinement of the elements is realized with octrees in 3D and quadtrees in 2D.

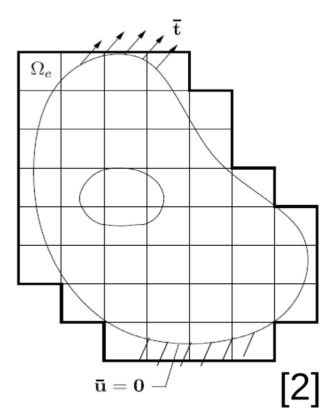


Boundary Conditions

Since the Finite-Cell mesh does not resolve the physical domain, the boundary conditions cannot be imposed in a classical manner. Instead, they are enforced in a weak sense. Considering the weak form of equilibrium for linear elasticity:

$$\int_\Omega \nabla v \cdot \underline{\sigma}\, d\Omega - \int_{\Gamma_D} (v \cdot \underline{\sigma}) \cdot n\, d\Gamma = \int_\Omega b \cdot v\, d\Omega + \int_{\Gamma_N} (v \cdot \underline{\sigma}) \cdot n\, d\Gamma$$

with Neumann boundary conditions $\underline{\sigma} \cdot \underline{n} = \hat{\underline{t}}$ on Γ_N
and Dirichlet boundary conditions $\underline{u} = \hat{\underline{u}}$ on Γ_D



[2]

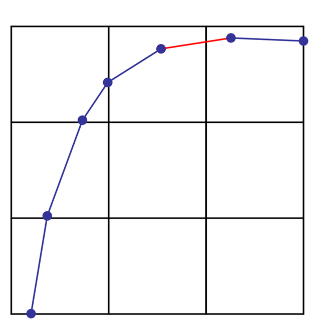
Neumann boundary conditions are satisfied in a weak sense through the integral term over Γ_N. Dirichlet boundary conditions can only be enforced strongly if Γ_D conforms to the mesh. Therefore, in FCM they are satisfied in a weak sense, using Nitsche's method, by adding the integral $-\int_{\Gamma_D} (\underline{u} - \hat{\underline{u}}) \cdot \underline{t}\, d\Gamma = 0$ to the weak form of the equilibrium equation.

To maintain positive definiteness of the stifness matrix, a penalty term is added to the weak form:

$$\int_\Omega \nabla v \cdot \underline{\sigma}\, d\Omega - \int_{\Gamma_D} v \cdot \underline{\sigma} \cdot n\, d\Gamma - \int_{\Gamma_D} u \cdot \underline{\sigma}(v) \cdot n\, d\Gamma + \beta \int_{\Gamma_D} u \cdot v\, d\Gamma$$

$$= \int_\Omega b \cdot v\, d\Omega + \int_{\Gamma_N} v \cdot \hat{\underline{t}}\, d\Gamma - \int_{\Gamma_D} \hat{\underline{u}} \cdot \underline{\sigma}(v) \cdot n\, d\Gamma + \beta \int_{\Gamma_D} \hat{\underline{u}} \cdot v\, d\Gamma$$

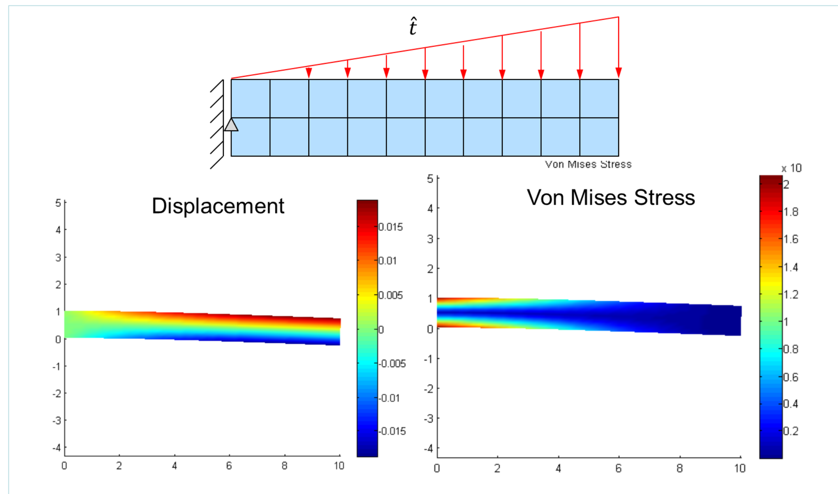
As implemented in the FCM Toolbox, some issues arose in dealing with weak boundary conditions for embedded geometries. For simple geometries, the boundary needs only a coarse description. To carry out numerical integration that is sufficiently accurate for higher order shape functions, a Gaussian quadrature is applied for each boundary element. Also, to account for boundary description elements intersecting cell boundaries, each quadrature point is evaluated in its own cell. Determining which cell contains a quadrature point is done efficiently due to the structured Cartesian grid.



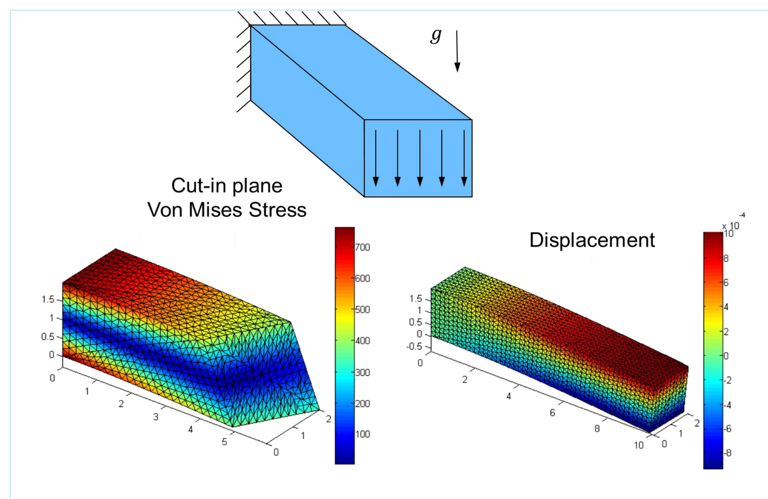
Results

p-FEM – 2D and 3D

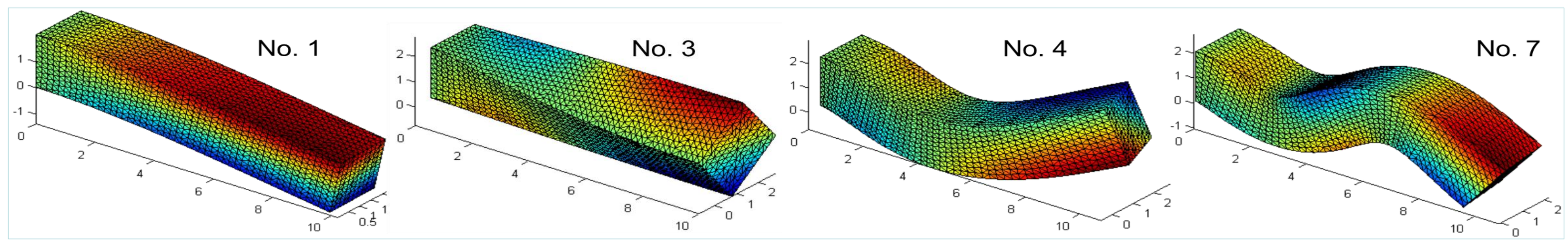
A cantilever beam was modeled in 2D and 3D. For the 2D case a distributed load was applied on the top. For the 3D case a body load (gravity) was added and a traction load was defined at the free end. Using modal analysis, that is also available in the toolbox, the eigenmodes of the beam can also be obtained in both 2D and 3D.



p-FEM 2D



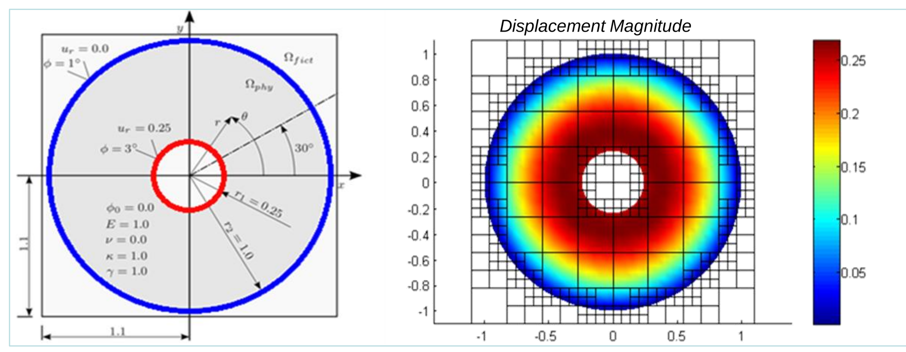
p-FEM 3D



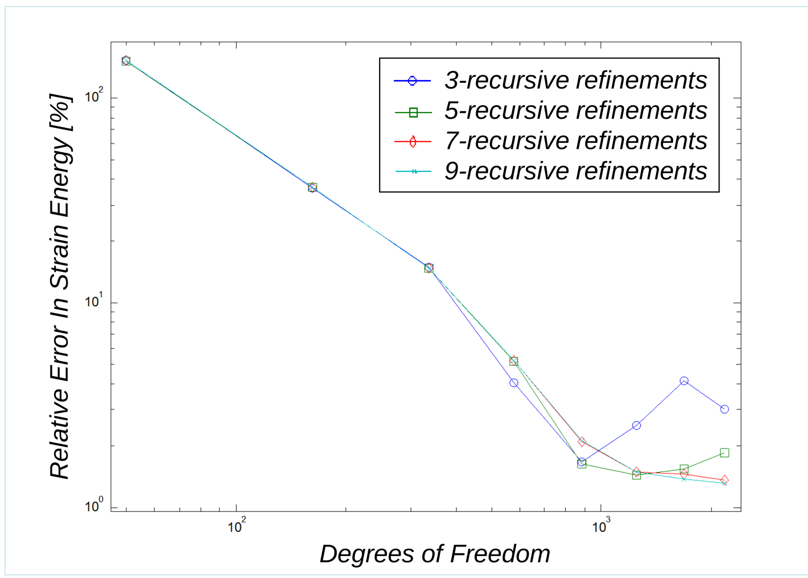
Modal Analysis

FCM – 2D

An explicitly defined annular disc is embedded in a square-shaped domain. The disc is fixed at the outer edge using weak boundary conditions, and has a radial displacement on the inner edge. A radial body load is also applied.



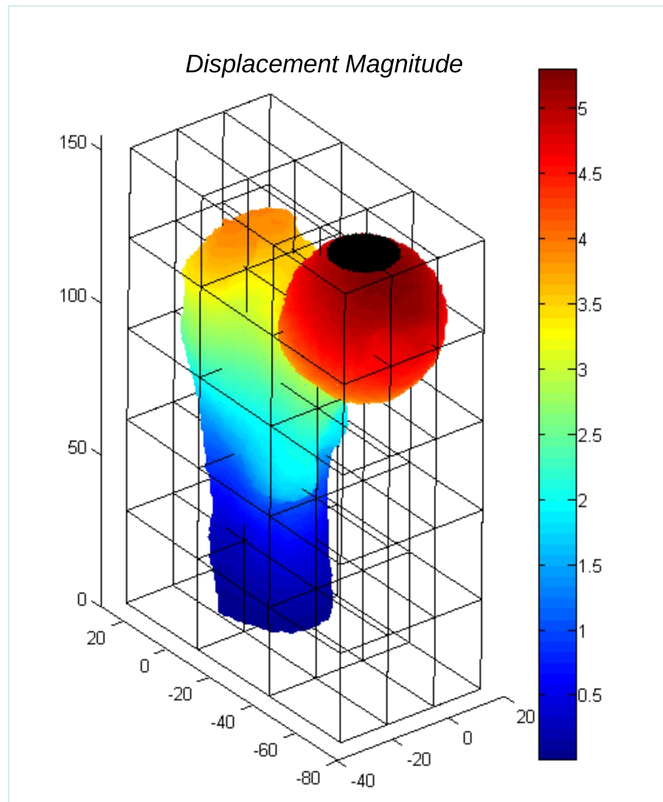
A 2D problem of an annular disc solved using FCM



Convergence rate of the annular disc problem

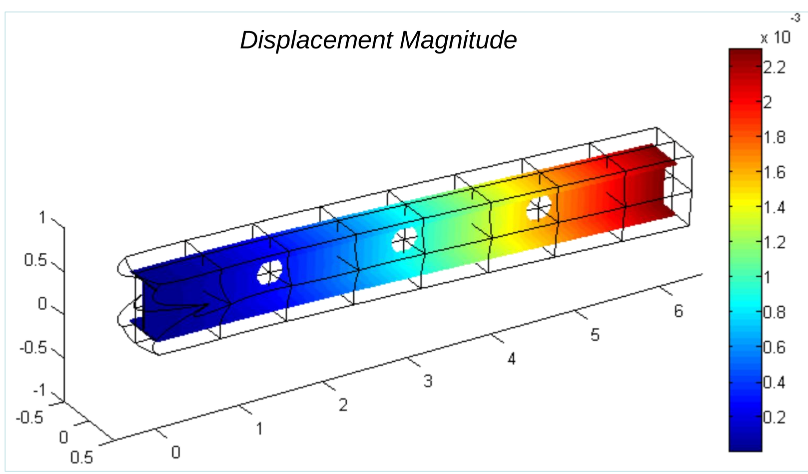
FCM – 3D

A human femur bone is defined via a voxel model. The bone is subject to a traction force at the top. The load surface is described by an STL file. The bottom surface of the whole mesh is fixed using strong boundary conditions.



A 3D bone defined via voxel model solved using FCM

A cantilever I-beam with circular holes was explicitly defined using the toolbox. It is subject to a body load (self weight) and is clamped using weak boundary conditions.



An explicitly defined 3D I-beam solved using FCM

Features

Geometry	• Defined explicitly or via a voxel model
Material	• Multiple materials in one problem
B.C.	• Defined anywhere explicitly or via an STL file
Analysis	• Static and modal analysis
Post-processing	• Evaluate various quantities and plot on deformed geometry/cutting plane/path

Conclusions / Outlook

As the result of the intensive work over 3 months of a team of 6 highly motivated students, this toolbox fulfills the initial requirements of the project. Its 15000 lines of MATLAB object-oriented code allow the simulation of 2D and 3D linear mechanical problems via the Finite Cell Method.

Using state-of-the-art methods for clean programming, thoroughly commented, documented and tested, the code was also designed to allow further extensions. The first steps of the further development of the program will consist of optimizing the computation time. New functionality may then be the subject of future master theses or software lab projects.

We would like to express our deep gratitude to our supervisors Tino Bog and Nils Zander for having taken a lot of their time to guarantee the success of the project.

References

[1] A. Düster, High-Order FEM, Lecture Notes, CiE, 2011

[2] A. Düster, J. Parivizian, Z. Yang, E. Rank, The finite cell method for three-dimensional problems of solid mechanics, Comput. Methods Appl. Mech. Engrg. 197 (2008) 3768–3782

[3] J. Parvizian, A. Düster, E. Rank, Finite cell method – h- and p-extension for embedded domain problems in solid mechanics, Comput. Mech. 41 (2007) 121–133