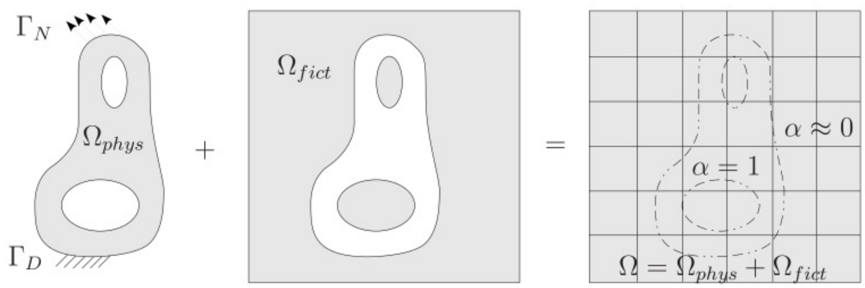
# FCM Toolbox for MATLAB

Since the early years of computational engineering, the classical Finite Element Method (FEM) has become the state-of-the-art approach to solve boundary value problems numerically. Although major enhancements allowed for highly sophisticated simulations, the idea to geometrically resolve the physical domain on the discretization level remained unchanged. In cases of complex geometries, this intrinsic need for a conform mesh is still a burden in today's engineering practice.

The recently introduced Finite Cell Method (FCM) overcomes this problem by combining the benefits of high-order Finite Elements (p-FEM) with the idea of fictitious domains [1]. The approach embeds the possibly complex physical domain $\Omega_{phys}$ in a fictitious domain $\Omega_{fict}$ and solves the problem on their simply shaped union. The localization factor $\alpha$ allows to recover the original geometry on the integration level.




The FCM shall now be enhanced by novel numerical approaches. The immediate verification of these new ideas demands for a slim and efficient framework that is easy to extend. The development of such a FCM toolbox in MATLAB is the task of this softwarelab.

By choosing this task, you have the chance to gain inside into cutting-edge numerical methods and to prove your expertise in programming. Your task will be to design, implement and test an object-oriented FCM framework that solves boundary value problems on non-conforming Cartesian grids. Your work will involve a literature research on high-order Finite Elements, the Finite Cell Method and the essentials of software engineering. Based on this knowledge, you, as a group, will design the structure of the toolbox. With this outline at hand, you will implement the framework so that, in the end, you can simulate linear-elastic deformation of structures.

Supervisor

Tino Bog and Nils Zander

References

[1] J. Parvizian, A. Düster, and E. Rank. Finite cell method – h- and p-extension for embedded domain problems in solid mechanics. Computational Mechanics, 41:121–133, 2007.