Open Source Software for Coupled Field Simulation

M. Kaltenbacher[1], D. Mayrhofer[1], E. Mušeljić[1], K. Roppert[1], F. Toth[2], F. Wein[3]
[1]Institute of Fundamentals and Theory in Electrical Engineering, TU Graz, Austria
[2]Institute of Mechanics and Mechatronics, TU Wien, Austria
[3]Department of Mathematics, Friedrich-Alexander-Universität Nürnberg (FAU), Germany

Keywords: finite element method, multiphysics, open source software

Abstract The open source software *openCFS* is a finite element-based multi-physics modeling and simulation tool available for Linux, Windows and macOS. With about 20 years of research-driven development, the core of *openCFS* is used in scientific research and industrial application. The modeling strategy focuses on physical fields and their respective couplings.

1 Introduction

In most cases the fabrication of prototypes within the design process is a lengthy and costly task, and reliable computer tools capable of precisely simulating the multi-field interactions are of utmost importance. Arbitrary modifications of geometry and selective variation of material parameters are easily performed, and the influence on the behavior can be studied immediately. In addition, the simulation provides access to physical quantities that cannot be measured, e.g. the magnetic field in a solid body, and simulations strongly support the insight into physical phenomena [1].

The modeling of complex technical as well as medical systems leads to so called multi-field problems, which are described by a system of nonlinear partial differential equations (PDEs). The complexity consists of the simultaneous computation of the involved single fields as well as in the coupling terms, which in most cases introduce additional nonlinearities, e.g. moving/deforming conductive bodies within an electromagnetic field. For the efficient solution of these multi-field problems, we have developed an enhanced simulation environment based on the finite element (FE) method, which is continuously improved by new numerical schemes, advanced material models and coupling strategies.

2 Capabilities and Workflow

With a special focus on electromagnetics, structural mechanics, acoustics, and heat transfer, openCFS allows high-end computations of many coupled fields: electromagnetics-mechanics-acoustics; piezoelectrics-acoustics; electro-thermo-mechanics; electrostatics-mechanics-acoustics; aeroacoustics. The following features, which set *openCFS* apart from commercial simulation programs, provide an overview of methodological capabilities of *openCFS*:

- **Flexible discretization:** Non-conforming grid techniques can handle computational grids being considerably different in adjacent subdomains. Thereby, not only the numerical error can be strongly reduced, but also the pre-processing of complex geometries is significantly simplified.
- **Coupling strategies:** *openCFS* allows for both volume as well as surface coupling between different physical fields and performs a simultaneous solution of the coupled fields.
- **Higher order finite elements:** In addition to standard FE methods (iso-parametric approximation), *openCFS* allows for higher order elements, which guarantee optimal convergence rates and therefore computational efficiency.
- **Moving domains** Based on the Arbitrary-Lagrangian-Eulerian frame of reference, *openCFS* allows to incorporate domain movement for the computation of e.g. displacement dependent forces.
- **Optimization framework:** *openCFS* includes state-of-the-art density-based topology optimization / material optimization with a choice of solvers for different physics. The optimization can be easily extended using Python functions, e.g. to perform shape and topology optimization over geometric primitives (feature-mapping).
- **Extensive documentation:** As the software is used in courses at several universities, there is extensive documentation available that is suitable for quickly getting started with learning the software alongside the physical and numerical background. Material is publicly available in the form of the user documentation (https://opencfs.gitlab.io/userdocu/) and in videos (https://www.youtube.com/@openCFS).

The computational core of the software *openCFS* is written in C++, and provides functionality for solving PDEs resulting from single and coupled physical fields. The geometry information is read from external mesh files, where various formats from gmsh to Ansys-CDB are supported. The custom XML-based input file format allows for a user-friendly and easily scriptable definition of the physical problem, and field results can be visualized in the widely used software ParaView. Additionally, there is a wealth of tools available that facilitate advanced pre- and post-processing or automatization workflows.

Concurrent development of a large software project requires a suitable collaboration platform. The central place to connect users and developers are the git repositories on https://gitlab.com/openCFS. Here, you find *source code* with *README files* with essential information (e.g. for building), as well as a *Developer wiki* with further in-depth resources supporting new and advanced developers. Additionally, the GitLAb platform offers an issue tracker and merge-request workflow, including code review and testing via automatized CI/CD pipelines for various build configurations. The *Testsuite* repository containing more than 500 self-contained test cases of various features of openCFS is used for this purpose. The tests can also serves as examples/templates for own simulations and optimizations.

In addition, the association *Verein zur Förderung der Forschung im Zusammenhang mit der Software openCFS* has been founded in 2020. The main objectives of the association are (1) promotion of the use of *openCFS* for scientific problems which are solved with the help of numerical simulation, thus leading to new, innovative products for the benefit of mankind (Technology for People); (2) promotion of the scientific documentation of *openCFS* and the use of *openCFS* in higher education; (3) promotion of young scientists, especially students at universities in Austria and Germany and worldwide, in the field of numerical simulation of complex technical systems. [2]

4 Demonstrative Example

Induction heating is a pollution-free, fast and secure technology with high energy efficiency, and it facilitates targeted heating for applications including surface hardening, melting, brazing, soldering and heating for shrink fitting, to name just a few. Figure 1 schematically displays a transverse flux induction heating system including the inductor and the metal sheet moving with velocity $\boldsymbol{u}$. As shown the magnetic field $\boldsymbol{B}$ indicated by the dashed flux lines is perpendicular to the sheet, resulting in eddy current loops parallel to the inductor with three turns. The

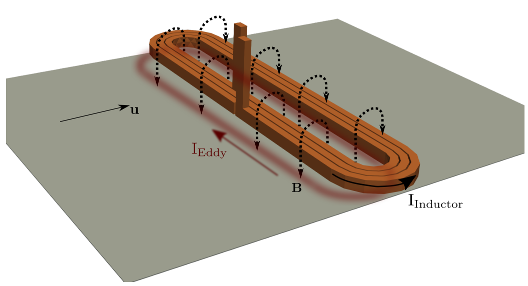

Fig. 1. Principle of the transverse flux induction heating system.

challenge for the numerical simulation of the induction heating process is the large time-scale disparity between the electromagnetic field changing in microseconds and the thermal field changing in seconds. To achieve a computationally efficient scheme, we have developed a coupling scheme in which the nonlinear electromagnetic field is computed in the frequency domain via the harmonic balancing method and the thermal field in the time domain. In doing so, within each time step of the thermal field we solve the electromagnetic field in the frequency domain with the current temperature distribution, evaluate Joule's losses (due to the induced eddy currents in the metal sheet) and use them for computing the new temperature distribution. Figure 2 displays the computational domain and the generated non-conforming computational grid. Due to such a non-conforming grid, we can strongly reduce the size of the algebraic system of equations obtained by applying the FE method, and in addition, the high-quality grid strongly reduces the numerical error. Figure 3 shows the comparison between measured and

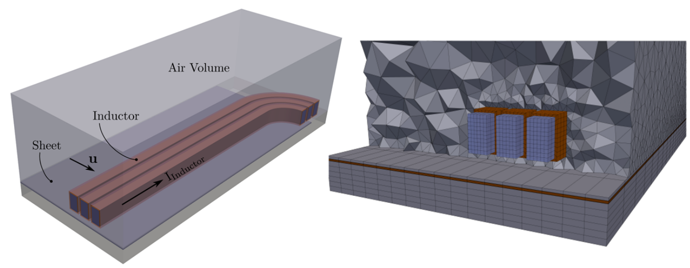

Fig. 2. Computational domain (quarter setup) and details of the non-conforming grid.

simulated temperature distribution along the steel sheet.

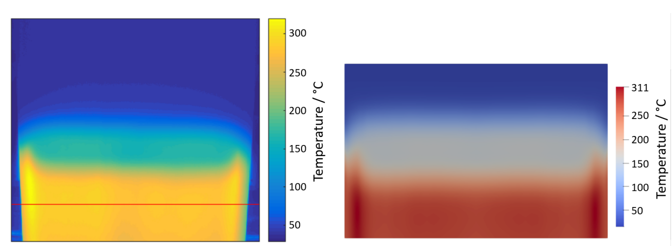

Fig. 3. Comparison of temperature fields from experiment (left) and simulation (right).

References

[1] Manfred Kaltenbacher. *Numerical Simulation of Mechatronic Sensors and Actuators: Finite Elements for Computational Multiphysics*. Springer, 3. edition, 2015. doi:10.1007/978-3-642-40170-1.

[2] Verein zur Förderung der Forschung im Zusammenhang mit der Software openCFS. https://opencfs.org. URL: https://opencfs.org.