Increasing Flexibility and Reusability of Finite Element Simulations With ViennaX

Josef Weinbub
Institute for Microelectronics, Technische Universität Wien
weinbub@iue.tuwien.ac.at

Karl Rupp
MCS Division, Argonne National Laboratory
rupp@mcs.anl.gov

Siegfried Selberherr
Institute for Microelectronics, Technische Universität Wien
selberherr@iue.tuwien.ac.at

Abstract

The ever-increasing availability of finite element (FE)-related software tools puts pressure on the software development of simulation applications. Albeit the fact that feature-rich packages such as the deal.II library are available, solely utilizing a static set of tools for an application limits the capabilities of the implementation with respect to flexibility and reusability [1]. For instance, exchanging the linear solver typically requires actual coding and thus implies in-depth knowledge of the code base. A usual way to tackle these challenges are component approaches, such as the Cactus framework [2]. However, the available frameworks either offer an almost prohibitively high entry-level for users or focus on domain-specific applications, like computational fluid dynamics. We introduce our approach, being the plugin execution framework ViennaX [3], which aims for general applicability in scientific computing. Albeit the framework's genericity, in this work we focus on FE-based applications. Engineering simulations such as stress simulations are decoupled into reusable ViennaX components, for instance, assembly and linear solver components. These components are connected via the data communication layer to setup a modular simulation. Due to the ViennaX's run-time system, components can be exchanged without programming efforts, thus increasing flexibility significantly. For instance, different error estimators or mesh adaptation components can be utilized. Additional aspects essential for FE applications like adaptive FE approaches are discussed as well. We depict the utilization of GPUs via the ViennaCL [4] library as well as clusters to underline the applicability for high-performance computing. It is shown that utilizing ViennaX not only results in high flexibility and reusability but also in high performance simulations.

References

1. D.E. BERNHOLDT ET AL.. A Component Architecture for High-Performance Scientific Computing. Intl J High Perform C 20(2) (2006).
2. T. GOODALE ET AL.. The Cactus Framework and Toolkit. In Proc. of VECPAR (2002).
3. J. WEINBUB ET AL.. ViennaX: A Parallel Plugin Execution Framework for Scientific Computing. Eng. Comput. (2013), in press. Web page: http://viennax.sourceforge.net/.
4. K. RUPP ET AL.. ViennaCL. Web Page: http://viennacl.sourceforge.net/.