

MULTI-LEVEL hp -FEM: HIGH-ORDER MESH ADAPTIVITY WITHOUT THE DIFFICULTIES OF HANGING NODES

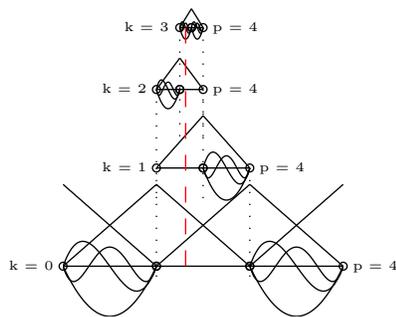
Nils Zander^{1*}, Tino Bog¹, Stefan Kollmannsberger¹, Dominik Schillinger²
and Ernst Rank¹

¹ Chair for Computation in Engineering, Technische Universität München, Arcisstr. 21,
80333 München, Germany

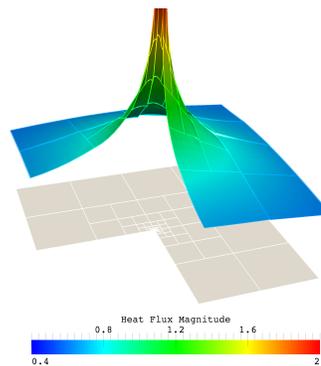
² Department of Civil Engineering, University of Minnesota, 500 Pillsbury Drive S.E.,
Minneapolis, 55455 MN, USA

Key words: *high-order Finite Elements, hp-adaptivity, arbitrary level hanging nodes*

Within the framework of the Finite Element Method, a major challenge of mesh adaptivity is to guarantee the continuity requirements of the approximate solution. In particular, mesh *irregularities* caused by *hanging nodes* should be avoided. Since the early 1980's, various refinement strategies have been introduced for this purpose. One common idea is to ensure continuity by appropriately constraining hanging nodes (see e.g [1, 6]). Although this approach has proven to work well for various applications, the implementation bears some challenges. These become even more severe in the context of hp -refinements, as here also edge- and face-modes have to be handled accordingly. For this reason, hp -algorithms are typically restricted to *1-irregular* meshes to reduce the implementation complexity (see e.g. [1]). Only recently, advanced numerical schemes have been implemented that allow for arbitrary-level hanging nodes (see e.g. [5]). However, the idea to post-constrain hanging nodes remains unchanged, which requires a sophisticated algorithmic treatment.



(a) Multi-level hp adaptivity in 1D



(b) L-shaped domain example

To overcome these challenges, the authors suggest an alternative refinement strategy that avoids the difficulties associated with arbitrary-level, high-order, hanging nodes. The approach is based on the idea of *hp-d* refinements, in which a high-order base mesh is superposed with a finer *h*-overlay mesh in the domain of interest (see [2]). Recently, the approach has been extended using hierarchical *h*- and NURBS-overlay meshes, which allows for adaptive *h*-refinement (see [3, 4]). In the present work, the idea is taken further to accommodate for adaptive high-order-refinement by employing a *high-order* hierarchical overlay mesh (see Figure a). This new approach offers the capabilities of *hp*-adaptive-methods while allowing for arbitrary-level irregular meshes. It will be demonstrated that the suggested refinement scheme leads to exponential rates of convergence in the pre-asymptotic range for non-smooth problems (see Figure b). Furthermore, the work will demonstrate the robustness of the method by showing transient simulations in which both refinement and coarsening steps are applied adaptively over time.

E-mail: *nils.zander@tum.de*

REFERENCES

- [1] L. Demkowicz. *Computing with hp-adaptive finite elements, Vol. 1: One and Two Dimensional Elliptic and Maxwell Problems*. Applied mathematics and nonlinear science series. Chapman & Hall/CRC, Boca Raton, 2007.
- [2] E. Rank. Adaptive remeshing and h-p domain decomposition. *Computer Methods in Applied Mechanics and Engineering*, 101(1-3):299–313, 1992.
- [3] D. Schillinger, L. Dedè, M. A. Scott, J. A. Evans, M. J. Borden, E. Rank, and T. J. Hughes. An isogeometric design-through-analysis methodology based on adaptive hierarchical refinement of NURBS, immersed boundary methods, and t-spline CAD surfaces. *Computer Methods in Applied Mechanics and Engineering*, 249-252:116–150, 2012.
- [4] D. Schillinger, A. Düster, and E. Rank. The hp-d-adaptive finite cell method for geometrically nonlinear problems of solid mechanics. *International Journal for Numerical Methods in Engineering*, 89(9):1171–1202, 2012.
- [5] P. Solin, J. Cerveny, and I. Dolezel. Arbitrary-level hanging nodes and automatic adaptivity in the hp-FEM. *Mathematics and Computers in Simulation*, 77(1):117–132, 2008.
- [6] P. Solin, K. Segeth, and I. Dolezel. *Higher-order finite element methods*. Studies in advanced mathematics. Chapman & Hall/CRC, Boca Raton, FL, 2004.