The multi-level hp-method for three-dimensional problems: dynamically changing high-order mesh refinement with arbitrary hanging nodes

Nils Zander*[1], Tino Bog[1], Mohamed Elhaddad[1], Felix Frischmann[1], Stefan Kollmannsberger[1], and Ernst Rank[1,2]

[1]Chair for Computation in Engineering, Technische Universität München, Arcisstr. 21, 80333 München, Germany
[2]Institute for Advanced Study, Technische Universität München, Lichtenbergstr. 2a, 85748 Garching, Germany

Abstract

One main challenge of the *hp*-version of the finite element method is the high implementational complexity of the method resulting from the added need of handling hanging nodes appropriately. The multi-level *hp*-formulation—recently introduced for two-dimensional applications—aims at alleviating these difficulties without compromising the approximation quality. This is achieved by changing from the conventional refine-by-replacement approach to a refine-by-superposition idea. The current work shows that the multi-level *hp*-approach can be extended naturally to three-dimensional refinement without increasing the complexity of the rule set ensuring linear independence and compatibility of the shape functions. In this way, a three-dimensional *hp*-refinement scheme is formulated, in which hanging nodes are avoided by definition. This ease of complexity allows for a highly flexible discretization kernel featuring arbitrary irregular meshes and a continuous refinement and coarsening throughout the simulation runtime. Different numerical examples demonstrate that—even in the presence of singularities—this novel refinement scheme yields exponential convergence with respect to both, the number of unknowns and the computational time. It is further shown that the refinement scheme is able to capture complex solution features that demand for three-dimensional refinement patterns. The dynamic discretization properties of the approach are demonstrated by continuously refining and coarsening the mesh during the simulation runtime to keep the refinement zone local to a moving singularity. Finally, it is shown that the high approximation power of the multi-level *hp*-scheme also carries over to curved geometries common in engineering practice without a significant detrimental effect on the conditioning of the stiffness matrix.

Keywords: high-order FEM, 3D *hp*-refinement, arbitrary hanging nodes, arbitrary irregular meshes, dynamically changing meshes

1 Introduction

The efficient numerical simulation of complex physical problems often requires a local refinement of the spatial discretization in domains of interest. The *hp*-version of the finite element method (*hp*-FEM) meets this demand as it features both, a variable element size h and a changing polynomial degree p. This allows to *replace* elements with a high error contribution near singularities with a set

*nils.zander@tum.de, Corresponding Author

Preprint submitted to *Computer Methods in Applied Mechanics and Engineering* August 11, 2016
The final publication is available at http://dx.doi.org/10.1016/j.cma.2016.07.007

of smaller elements. On the other hand, coarse elements with a high polynomial degree can capture the solution in regions with high regularity. This combination yields exponential convergence even in the presence of singularities as shown e.g. in [1–8]. The applicability of the method has been demonstrated in the context of various applications and was combined with other discretization techniques such as the discontinuous Galerkin (DGM) or generalized finite element method (GFEM) in e.g. [9–15].

However, the *hp*-method has not become the standard in engineering simulations. This may be attributed to the high implementational effort of the discretization kernel, which partially originates from the aforementioned "refine-by-replacement" idea [16, 17]. The difficulty of the approach is that shape functions of fine-scale elements have no corresponding counterpart in the coarse, unrefined neighbors as indicated in Figure 1. The resulting *hanging nodes* render the mesh *irregular* [18]. To still ensure the *compatibility* of the shape functions and thus the convergence of the method, interelement continuity has to be restored [19–21].

One possible solution is to also sub-divide adjacent elements. These elements then form the transition between the refined and coarse part of the mesh. As outlined in [22, 23], classical approaches in this context are the longest-edge-bisection [24–26], newest-vertex-bisection [27, 28], and the regular/red-green refinement [29, 30]. Although mesh regularity can be restored with these transition elements, the refinement spreads out and thus loses its local character [17]. Moreover, special care has to be taken to prevent distorted elements, which would decrease the approximation quality [23, 31]. Furthermore, the automatic creation of transition elements is challenging—in particular for three-dimensional problems where it typically demands for template solutions [32, 33].

An alternative approach is the *constrained approximation*. Here, the hanging fine-scale shape functions are expressed in terms of their corresponding large-scale parent. This allows to derive additional constraints that recover the interelement continuity. These constraints are incorporated in the assembly process to ensure compatibility. Different implementations of this process are described e.g. in [7, 8, 16, 18, 34, 35].

Although the constrained approximation approach is prevalent and has proven to yield excellent

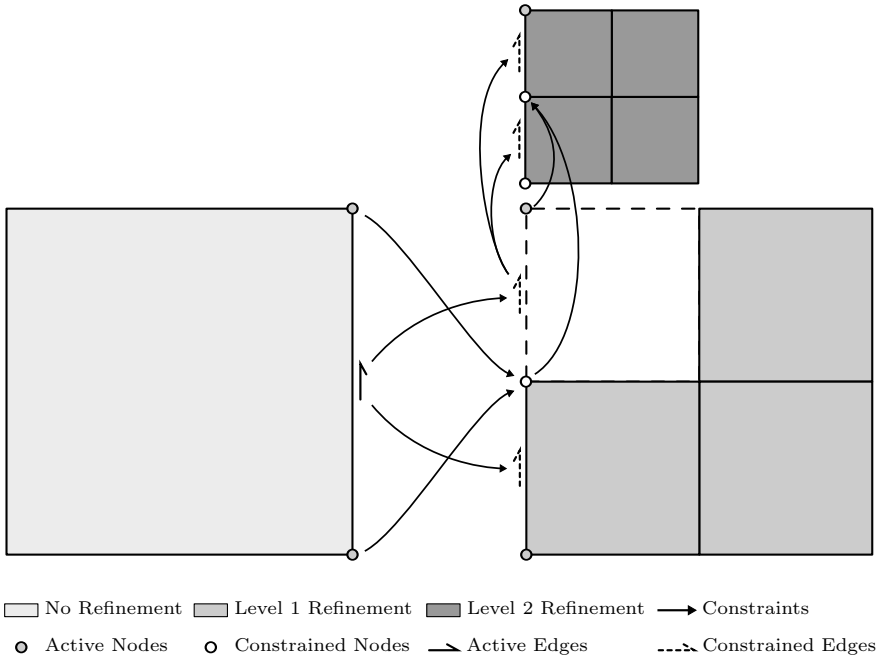


Figure 1: Multiply constrained modes occurring in classical *hp*-refinement strategies.

results in various applications, it comes with a high algorithmic complexity [16]. The implementation becomes even more involved if the constraints have to be derived in terms of already constrained nodes [7]. For this reason, most *hp*-frameworks do not allow for these multiply hanging nodes. Instead, only 1-irregular meshes are considered [7, 8, 18, 36–42]. Although this reduces the implementational complexity, the refinement again loses its local character which introduces unnecessary degrees of freedom [17, 43]. Although different formulations have been derived that allow for arbitrary hanging nodes [16, 17, 35, 43–45], the initial refine-by-replacement idea remained unchanged such that these approaches still demand for a sophisticated discretization kernel.

An alternative to complex refine-by-replacement strategies is to refine by *superposition*. The essential idea is to superpose a coarse *base* mesh by a finer *overlay* mesh. While the coarse base mesh can capture the global solution characteristics, the fine overlay mesh increases the approximation quality in the domains of interest. The continuity of the solution can be ensured easily by applying homogeneous Dirichlet boundary conditions on the overlay mesh. Consequently, hanging nodes are avoided *by construction*, which simplifies the refinement algorithm.

To the authors' knowledge, the superposition approach was first introduced in the pioneering work of Mote in 1971 [46]. A comprehensive review of the method is given in e.g. [47]. The essential idea was to increase the approximation accuracy of global functions known from the Ritz method by overlaying conventional finite elements in domains of interest. This concept has been extended and applied successfully in the context of e.g. the hierarchic finite element method [48, 49], embedded localization zones [50–52], the spectral overlay method [53], the *hp*-*d*-refinement method [54], the *s*-version of the finite element method [55–58], and the adaptive local overlapping grid method [59]. Partition of Unity methods (PUM, [60]), like the eXtended- and the generalized FEM [61, 62], also apply a similar concept by introducing additional shape functions to capture special solution characteristics.

The multi-level *hp*-method—recently introduced in [63]—utilizes the refine-by-superposition concept to formulate an *hp*-version of the finite element method, which a) circumvents the implementational difficulties caused by hanging nodes and b) yields the same approximation accuracy as conventional *hp*-methods. As shown in [63] for two-dimensional applications, the multi-level *hp*-formulation yields exponential convergence also in the presence of singularities. At the same time, the discretization kernel can be kept flexible allowing for a dynamic change of the mesh structure during the simulation. This renders the multi-level *hp*-approach as a suitable alternative for conventional *hp*-approaches if the desired accuracy lies in the target range of *hp*-FEM.

In the present work, the multi-level *hp*-method is extended to three-dimensional problems. For this purpose, Section 2 describes the idea of the formulation and its extension to higher dimensions in more detail. In Section 3, implementational aspects are addressed, showing that the rule-set ensuring compatibility and linear-independence of the three-dimensional shape functions is as simple as in two dimensions. The numerical examples presented in Section 4 demonstrate that exponential convergence can be obtained for three-dimensional examples containing singular points or steep gradients. The method is also applied to a transient problem, in which a moving singularity traverses through the domain. Keeping the refinement zone local to the moving singularity requires to continuously refine and coarsen the discretization throughout the runtime of the simulation. A fourth example demonstrates that the advantageous approximation properties of the proposed method also carry over to curved geometries common in engineering practice. The work closes with a concluding outlook in Section 5.

2 Multi-level *hp*-refinement

As mentioned in the previous section, the multi-level *hp*-approach improves the approximation quality by *superposing* coarse base elements with finer overlay elements. Hence, the final approximation is

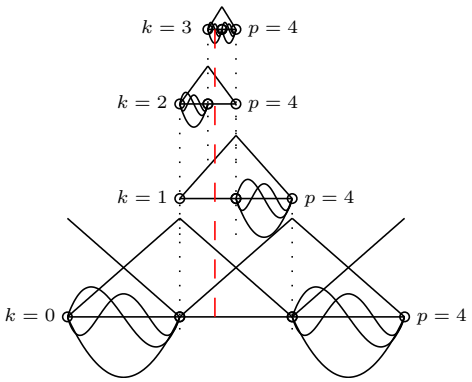
composed of a large-scale *base* mesh solution $\boldsymbol{u_b}$ and the fine-scale *overlay* solution $\boldsymbol{u_o}$:

$$\boldsymbol{u} = \boldsymbol{u_b} + \boldsymbol{u_o}. \tag{1}$$

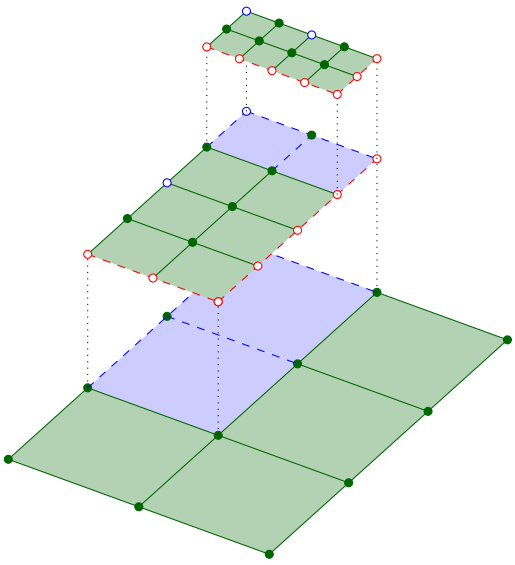
In the context of the multi-level *hp*-approach introduced in [63], this idea is used to avoid hanging nodes occurring in classical *hp*-formulations. To this end, the large-scale solution $\boldsymbol{u_b}$ is discretized using a coarse, high-order base-mesh, while the fine-scale solution $\boldsymbol{u_o}$ is discretized by smaller overlay elements in the domain(s) of interest. These finer elements themselves can be refined again by hierarchically superposing *multiple* levels of overlay meshes recursively, giving the method its name. This conceptual idea of the multi-level *hp*-method is depicted in Figure 2.

When following this hierarchical decomposition approach, it is essential to ensure a) *linear independence* and b) *compatibility* of the basis functions. Both of these requirements can be met without difficulty:

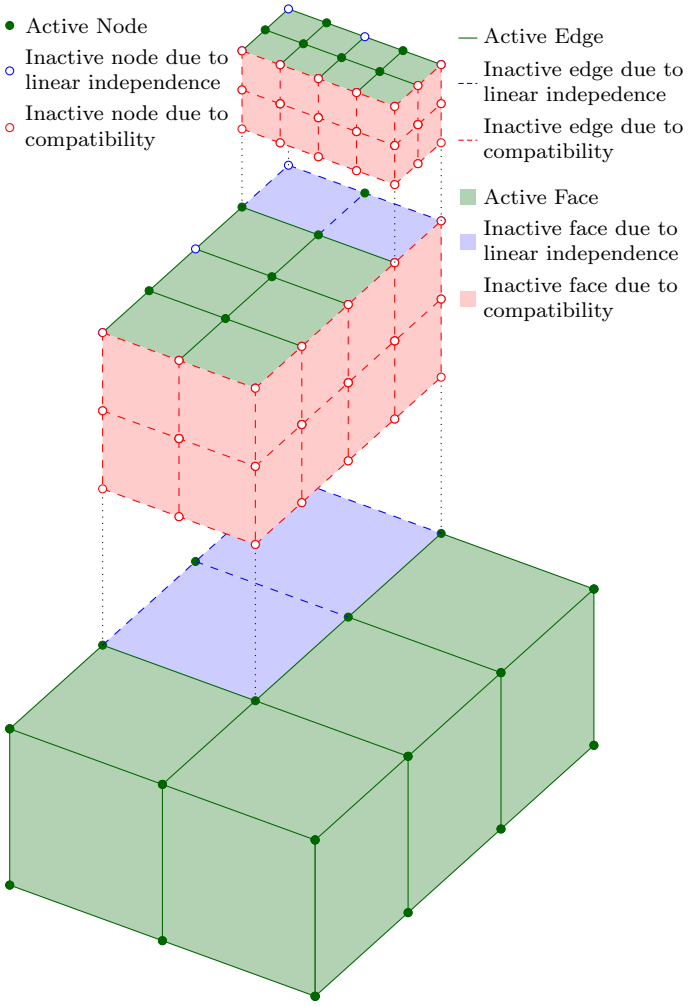
Linear independence To ensure linear independence, coarse-scale basis functions must not be representable by a linear combination of fine-scale overlay functions. This can be achieved easily by



(a) One-dimensional case

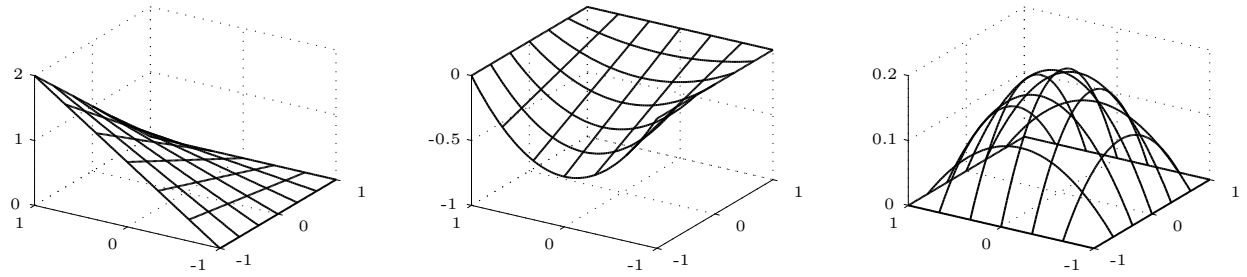


(b) Two-dimensional case



(c) Three-dimensional case

Figure 2: Conceptual idea of the multi-level *hp*-method following [63].



(a) Nodal-mode (b) Edge-mode (c) Face-mode

Figure 3: Two-dimensional mode types following e.g. [64, 65].

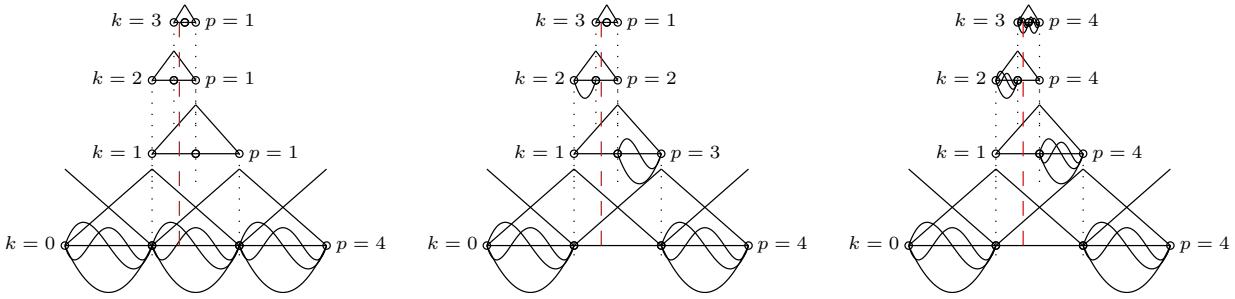
first arranging the overlay elements such that they do not overlap multiple parent elements. Further, using integrated Legendre shape functions, the basis functions can be associated directly to the nodes, edges, faces and solids of the mesh in the sense that the functions are non-zero on only one of these components and zero on all others (see Figure 3). This allows to easily remove selected shape functions from the discrete functions space by *deactivating* the respective topological component. Within this structured setting, linear independence can then be guaranteed by deactivating all topological components with active sub-components.

In the one-dimensional case depicted in Figure 2a, this translates to shifting all high-order shape functions from the refined base elements to the leaf-elements of the refinement tree. On the respective parent-level, only nodal modes remain active to ensure the global continuity of the solution. The same concept also carries over to the two- and three-dimensional settings. In these cases, the edge-, face-, and internal-modes of overlayed elements have to be deactivated as shown in Figures 2b and c.

Compatibility Given a variational index m, the compatibility of the discretization demands C^m-continuity of the shape functions within each element and C^{m-1}-continuity across the element boundaries [19–21]. For the Laplace-based problems considered in this work ($m = 1$), this translates in C^1-continuity within elements and C^0-continuity between elements. As piece-wise polynomial functions are used for the approximation, the basis functions are C^∞-continuous within each element by construction. The inter-element continuity can be ensured easily by applying homogeneous Dirichlet conditions on the boundary of the overlay meshes. This constrains the overlay solution $\boldsymbol{u_o}$ to zero on the boundary of the refinement zone, which ensures C^0-continuity between elements and thus the compatibility of the basis functions.

In the one-dimensional case depicted in Figure 2a, this translates to deactivating all overlay nodes on the boundary of each refinement level. As all remaining shape functions of the respective level are zero on the boundary, the C^0-continuity of the solution is guaranteed. This approach *naturally* extends also to higher dimensions. In the two-dimensional case depicted in Figure 2b, the compatibility is ensured by deactivating the respective nodes and edges on the boundary of the refinement level. Only when this boundary coincides with the boundary of the physical domain, the respective components stay active, as no continuity has to be ensured in this case. As shown in Figure 2c, the same idea also applies in the three-dimensional setting. Here, all nodes, edges, and faces on the boundary of the refinement zone have to be deactivated. In this way, the problem of hanging nodes is avoided *by construction*, as no degrees of freedom are present on the boundary of the refinement zone.

In addition to the element size h, the polynomial degree p of the shape functions can be varied. To obtain an optimal convergence, fully automated *hp*-frameworks make use of a regularity estimator to automatically adapt the polynomial degree. A detailed comparison of different approaches is given in



(a) Linear overlay, *hp-d*-approach [54, 66]

(b) Graded multi-level *hp*-approach

(c) Uniform multi-level *hp*-approach

Figure 4: Comparison of different approaches for hierarchical, high-order refinement.

e.g. [67]. The application of these methods in the context of the multi-level *hp*-approach is a subject of current research. For this reason, the examples presented in this work use one of the *a priori* setups depicted in Figure 4. On one end of the spectrum, only elements with linear shape functions are used for the overlay as shown in Figure 4a. In this case, the high-order shape functions on the refined base elements can remain active without introducing linear dependencies between the levels. With this distribution, the multi-level *hp*-approach is identical to the *hp-d*-refinement strategy introduced in [54] and extended in [66, 68–74]. The other end of the spectrum—depicted in Figure 4c—is formed by the aforementioned possibility of shifting all higher-order modes to the leaves. In this case, the distribution is *uniform* on the finest elements. One intermediate possibility illustrated in Figure 4b is a *graded* approach. Here, the polynomial degree is decreased when increasing the refinement level. This strategy mimics the graded mesh design, in which small, low-order elements are used near singular points, whereas coarse, high-order elements capture the solution in regions where it is smooth [64].

Besides allowing for a simplified handling of hanging nodes, the idea of performing an overlay *hp*-refinement also has a positive effect on the numerical properties of the resulting finite element equation system. This shall be exemplified using the following singular benchmark problem introduced in [1]:

$$\text{Find } u \text{ such that } u'' = -\alpha(\alpha-1)x^{\alpha-2} \quad \forall\, x \in (0,1) \tag{2a}$$

$$\text{with} \quad u(0) = 0, \quad u'(1) = 0 \quad \text{and} \quad \alpha = 0.65. \tag{2b}$$

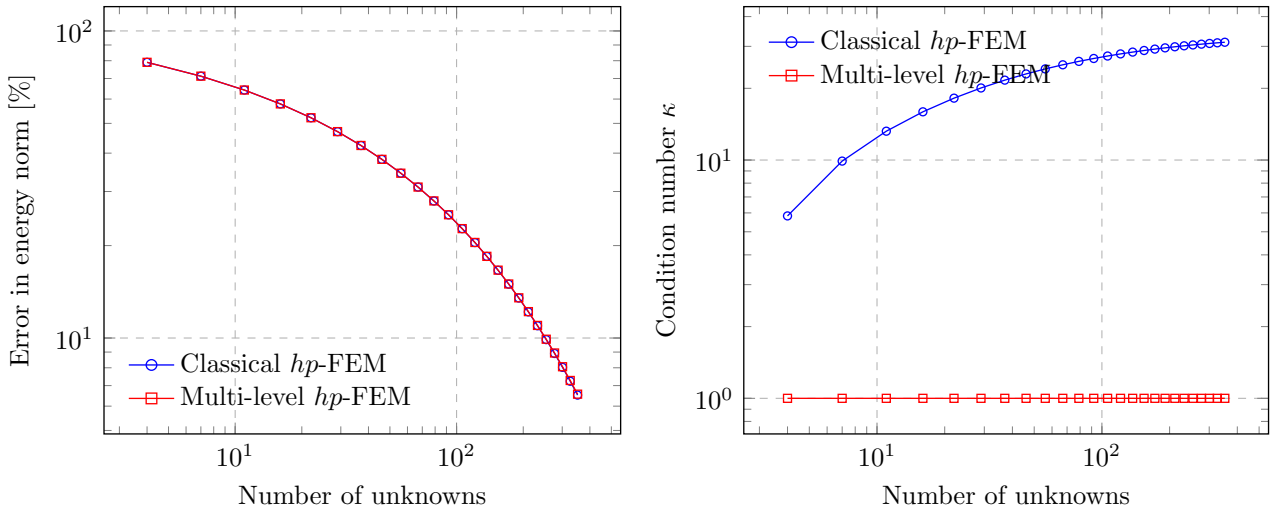
The analytical solution of this boundary value problem is given by

$$u(x) = -x^{\alpha} + \alpha x \quad \forall\, x \in [0,1]. \tag{3}$$

The singular nature of the analytical solution demands for the use of an *hp*-refinement to achieve an exponential convergence of the approximation error. To this end, the element size is graded geometrically towards the origin in a ratio of $q = 0.5$, and the polynomial degree of the ansatz functions is elevated linearly away from the singularity [2].

As illustrated in Figure 5a, this mesh design yields an exponential decay of the approximation error, which is identical for both, the classical and the multi-level *hp*-strategy. This is not surprising as both approaches construct the same ansatz space—at least for one-dimensional problems.

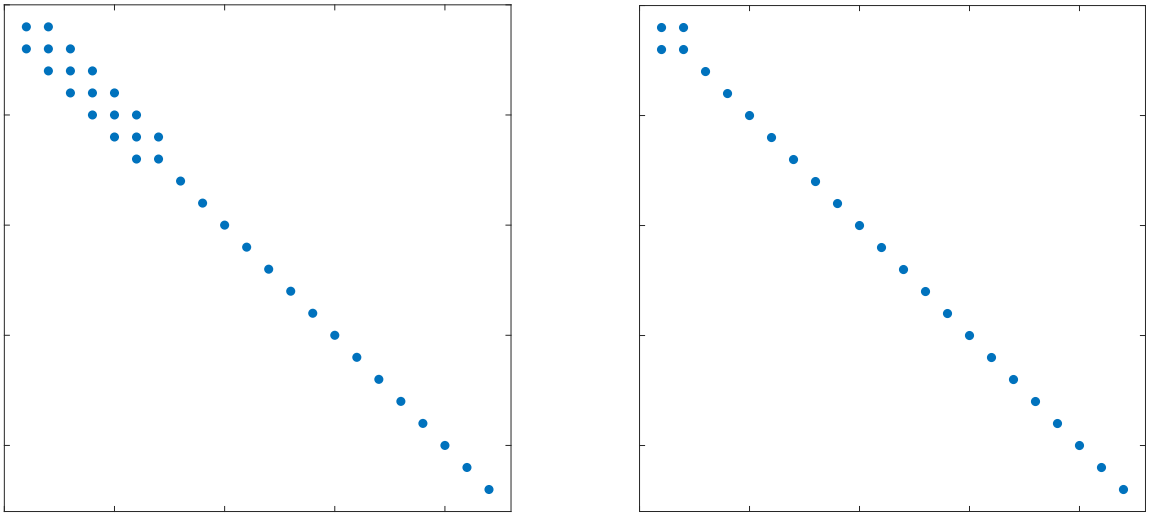
However, the different basis functions used by the two approaches lead to a significant difference in the condition number of the resulting stiffness matrix as illustrated in Figure 5b. While the condition



(a) Error in energy norm

(b) Condition number κ after Jacobi pre-conditioning

Figure 5: Comparison of classical and multi-level hp-FEM.



(a) Classical hp-FEM

(b) Multi-level hp-FEM

Figure 6: Comparison of sparsity patterns obtained using the classical and the multi-level hp-version of the finite element method (Number of unknowns: 22).

number of the classical *hp*-approach is monotonically increasing, the multi-level *hp*-approach yields a constant condition number $\kappa = 1$ for all discretizations. This improved characteristic partially originates from the orthogonality property of the used integrated Legendre shape functions. Interestingly, the same property applies to the superposed, fine-scale hat functions as these are orthogonal to all shape functions on all other refinement levels. Therefore, the multi-level *hp*-approach yields an almost fully orthogonal *hp*-basis when starting the analysis with one base element. Only the coarse linear shape functions on the base mesh remain non-orthogonal. Hence, the resulting stiffness matrix is almost diagonal with only two off-diagonal entries resulting from the two linear shape functions the base mesh, *cf.* Figure 6b. After applying the Dirichlet boundary condition and performing a Jacobi pre-conditioning, the basis is fully orthonormal and the condition number is independent of the number of elements.

The situation is fundamentally different when using the classical refine-by-replacement approach. Here, all linear shape functions are non-orthogonal, resulting in more off-diagonal entries in the stiffness matrix as illustrated in Figure 6a. Accordingly, the simple Jacobi pre-conditioning is not sufficient to fully orthonormalize the stiffness matrix. This leads to a constant growth of the condition number when refining the discretization.

Just as for the integrated Legendre shape functions, the orthogonality properties of the superposed hat functions do not carry over directly to cases involving more complex geometric domains in higher dimensions. For this reason, the solvability of the final equation system for such applications is studied for three-dimensional problems with curved geometries in Section 4.

Before addressing these aspects, the following section outlines the implementation of the discussed rule-set for a compatible and linear independent *hp*-basis.

3 Implementational aspects

As previously discussed, the main challenge in the multi-level *hp*-approach is to ensure linear independence and compatibility of the basis functions. This is achieved by first associating all shape functions spanning the approximation space to topological components of the mesh and then by activating and deactivating the "right" topological components. The following section describes the simple rule-set for this purpose in the context of an abstract object-oriented FEM-code. The outline follows the description given in [63].

3.1 Data structure

It is assumed that the connectivity of the mesh is represented by the different topological components: nodes, edges, faces, and solids. In an object-oriented framework, this corresponds to four separate classes, which inherit from a common base-class as depicted in Figure 7. This allows to treat the different types in a common way, since most of the implementation can be placed in this base-class. The main responsibility of the topology is the management of the degrees of freedom. This is reasonable in the context of higher-order FEM, as the shape functions can be identified as nodal-, edge-, face-, or internal-modes (see e.g. [65] and Figure 3). Hence, the degrees of freedom are

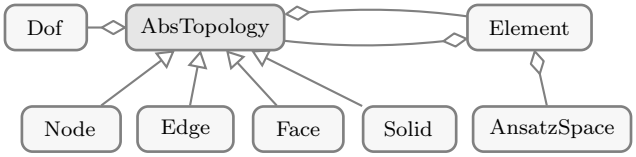


Figure 7: UML representation of the multi-level *hp*-data structure following [63].

associated to the topological components of their respective shape functions. The individual finite elements can then be defined *on* the topological components of the highest dimension—solids in 3D. Using the information provided by this *topological support*, the element constructs its ansatz-space.

Within this setup, individual components can be deactivated easily by setting the associated polynomial degree to zero. Accordingly, no degrees of freedom will be allocated by the respective component, and the finite element class will not compute the corresponding shape functions. This allows to easily remove individual shape functions from the ansatz space.

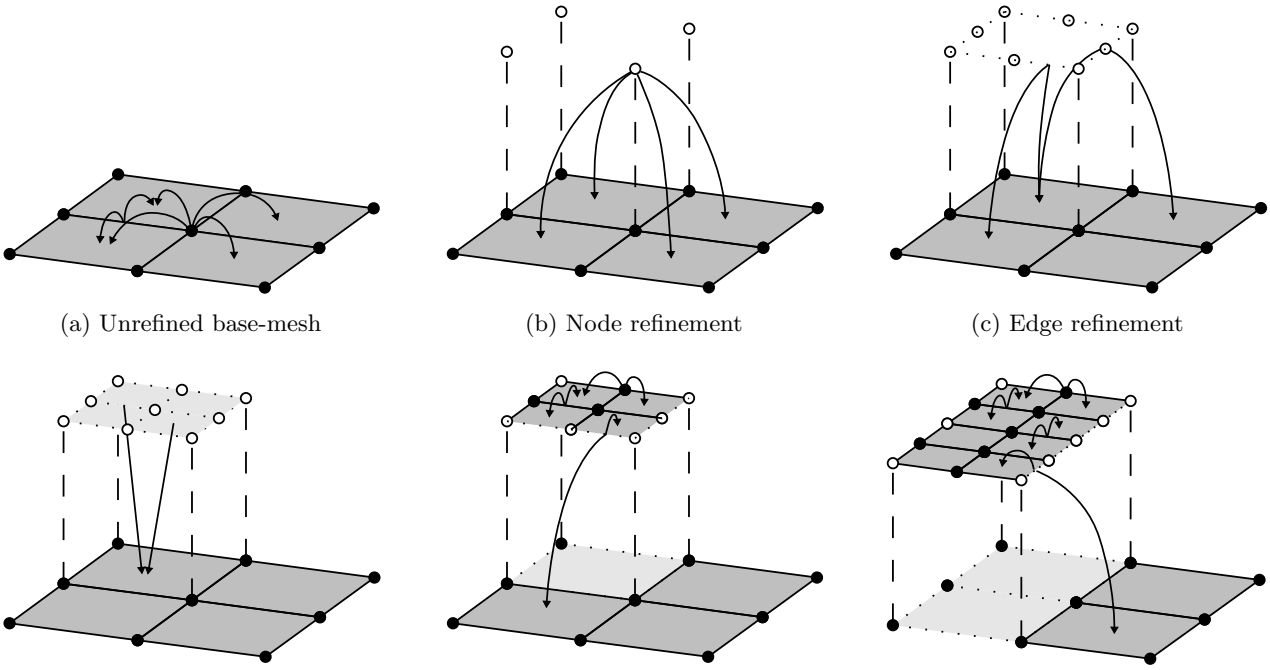
To select the correct components for deactivation, the element has to know the topological support it is defined on. Conversely, the individual components hold a list of adjacent elements. In Figure 8a, this is depicted exemplarily for one node and one edge by arrows pointing to the respective neighboring elements.

The data structure is completed by extending it in a tree-like manner. In this way, every component holds a reference to its sub-components. Similarly, the element knows its sub-elements. The generation of the refined objects is described in the next section.

3.2 Refinement procedure

Based on the data-structure introduced in the previous section, refining the discretization demands for subdividing the topology and the elements. The individual steps are shown in Figure 8. To reduce the complexity of the figure, the depiction is limited to the refinement of two-dimensional quadrilateral elements. However, this is no limitation of the algorithm itself.

To refine the topology, first the nodes and then the edges, faces and—in case of three-dimensional applications—also the solids are subdivided. For this purpose, *new* topological components are created, which *overlay* their respective parent in the next refinement level as illustrated in the



(a) Unrefined base-mesh (b) Node refinement (c) Edge refinement

(d) Face refinement (e) Element creation (f) Refinement of adjacent element

Figure 8: Implementation steps of multi-level *hp*-FEM following [63]. The arrows depict the adjacency relation between the created topological sub-components and the elements.

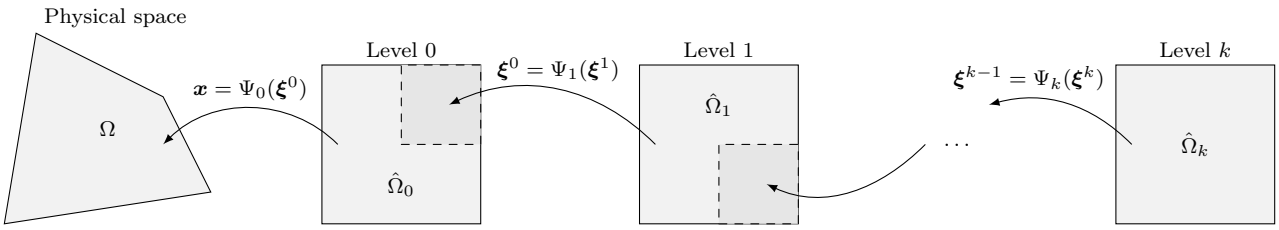


Figure 9: Concatenated geometry mapping: the sub-components are defined with respect to the index space of their parent topology.

Figures 8b-d. The list of adjacent elements is inherited from the respective parent. At this point, the sub-topologies are, therefore, still regarded as being connected to the original base elements. This is indicated by arrows pointing from the new components to the base elements in the Figures 8b-d.

An important decision at this point is whether to define the sub-topologies geometrically in the physical space or in the reference/index space of their parent. Defining the sub-topologies in the physical space offers the advantage that, in addition to the refinement of the actual solution space, also the details of the geometric model can be enhanced. However, if a refined geometric model is not available, the definition of the sub-components in the index space of the respective parent offers the advantage that the refined topology naturally follows the geometry of the possibly curved original component.

Without loss of generality, the latter approach is chosen in this work. As illustrated in Figure 9, the mapping between the local space $\hat{\Omega}_k$ of a sub-component on refinement level k and the global space Ω is a concatenated sequence of mappings:

$$\Psi = \Psi_0 \circ \Psi_1 \circ \ldots \circ \Psi_{k-1} \circ \Psi_k. \tag{4}$$

This sequence of mappings has to be taken into account when computing the derivatives of a function ϕ defined in $\hat{\Omega}_k$ with respect to the global coordinates $\boldsymbol{x}$. In case of the first derivative, the chain rule has to be applied as follows:

$$\frac{\partial \phi}{\partial x_i} = \frac{\partial \phi}{\partial \xi_j^k} \cdot \frac{\partial \xi_j^k}{\partial \xi_l^{k-1}} \cdot \ldots \cdot \frac{\partial \xi_n^1}{\partial \xi_m^0} \cdot \frac{\partial \xi_m^0}{\partial x_i}. \tag{5}$$

Here, $\boldsymbol{\xi^k}$ denote the index coordinates of the k^{th} refinement level. Using the inverse of the Jacobian matrix $\boldsymbol{J}_k$ of the mapping between the levels $k-1$ and k:

$$J_{ij}^k = \frac{\partial \xi_j^{k-1}}{\partial \xi_i^k} \quad \text{with} \quad J_{ij}^0 = \frac{\partial x_j}{\partial \xi_i^0}, \tag{6}$$

the above expression (5) can be re-written in matrix form as follows:

$$\nabla_{\boldsymbol{x}} \phi = \boldsymbol{J}_0^{-1} \cdot \boldsymbol{J}_1^{-1} \cdot \ldots \cdot \boldsymbol{J}_{k-1}^{-1} \cdot \boldsymbol{J}_k^{-1} \cdot \nabla_{\boldsymbol{\xi}^k} \phi. \tag{7}$$

After the topology has been refined, new sub-elements are created. In one-dimensional applications, the new elements are defined on sub-edges, whereas sub-faces and sub-solids serve as the support in two- and three-dimensions, respectively. As noted in Section 2, these new sub-elements do not *replace* the base element but *overlay* it instead. Therefore, the coarse elements are not removed from the mesh but still exist in the lower levels. However, they are de-registered from the adjacency lists of the newly created sub-topology, and the new sub-elements are registered instead. Hence, topological

components on the boundary of the refinement zone are connected to elements on different levels as depicted in Figure 8e. This information is essential for the rule-set of the deactivation process, which will be described in the next section.

3.3 Rule-set for a linear-independent and compatible basis

As outlined in Section 2, the compatibility and linear independence is ensured by deactivating the "right" topological components. Using the setup described above, these components can be selected based on the following simple rule-set:

Compatibility is ensured by deactivating all topological components whose adjacency list contains finite elements of different levels. In this simple way, all overlay components on the boundary of the refinement zone have no degrees of freedom. Hence, the overlay solution $\boldsymbol{u_o}$ is zero at the transition between the coarse and fine discretization, which ensures the inter-element C^0-continuity.

Linear independence of the shape functions is ensured by deactivating all topological components that have active sub-components. Furthermore, all direct successors of base nodes are deactivated.

These two rules automatically also activate the correct topological components on the boundary of the physical domain. As here the components are only connected to refined elements, they are activated without additional adjustment.

Furthermore, the adjacency information automatically updates when the adjacent element is refined as shown in Figure 8f. In this way, the sub-components can be re- or deactivated without difficulty by following the aforementioned simple rule-set. Multiple refinement levels can be handled easily by applying the algorithm recursively on the newly created sub-elements.

3.4 Numerical integration and evaluation of shape functions

Another aspect to be considered in the context of the multi-level *hp*-approach is the correct numerical integration of the stiffness matrix. Due to the previously discussed hierarchical superposition, the final set of shape functions is only C^0-continuous within the base elements. Therefore, applying conventional Gaussian quadrature on the support of the base element is not sufficient. Instead, the integral has to be evaluated separately on the leaf domains of the refinement tree as shown in Figure 10. For this purpose, the hierarchical structure of the overlays can be exploited: as each sub-element is fully contained within *one* parent element, no complex subdivision of the integration

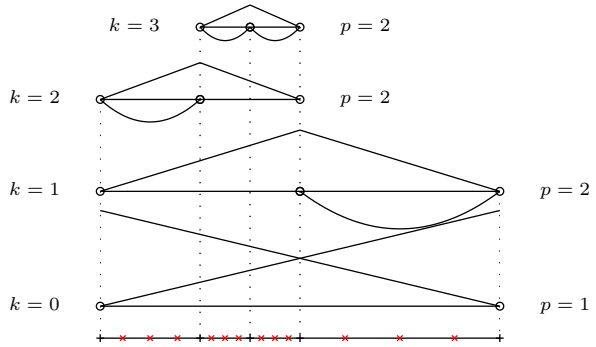


Figure 10: Integration of a multi-level element following [63].

domain is necessary. This is different in case of an unstructured overlay approach as discussed in e.g. [58]. Within the integration domains, the shape functions are polynomial. Thus, conventional Gaussian quadrature can be employed. On each of the resulting quadrature points, the values of all non-zero shape functions can be concatenated into one vector. With this active shape function set, the assembly of the finite elements can proceed as usual.

4 Numerical examples

Having introduced the essential idea and outlined the implementational aspects in the previous sections, the following section analyzes the approximation quality of the proposed method. To this end, the convergence characteristics of the refinement scheme in the presence of a singularity are evaluated in a first example. The second example studies the applicability of the method in the context of solutions demanding for a complex, three-dimensional refinement pattern. In the third example, the dynamic mesh refinement capabilities of the approach are assessed by considering a traveling singularity, whose local resolution demands for continuous change of discretization throughout the simulation. In the last example, the applicability of the refinement scheme to non-Cartesian geometries is demonstrated.

4.1 Fichera corner singularity

The aim of this first example is to analyze the convergence properties of the multi-level *hp*-approach in the presence of a singularity. For this purpose, the Fichera corner problem depicted in Figure 11a is considered. Following e.g. [45, 75–77], the Laplace problem is defined as

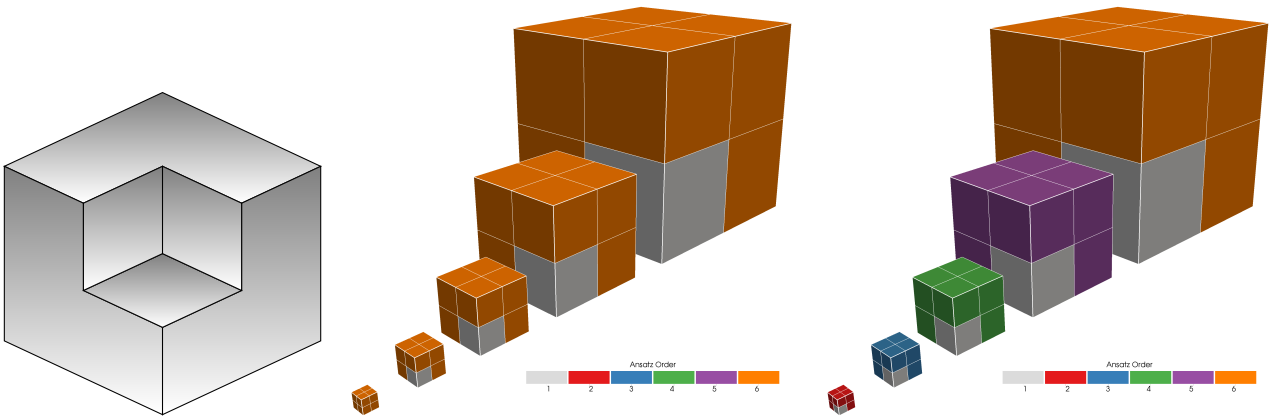
$$\Delta u = -\lambda \cdot (\lambda + 1) \cdot r^{\lambda - 2} \quad \forall \boldsymbol{x} \in \Omega \tag{8a}$$

$$\nabla u \cdot \boldsymbol{n} = 0 \quad \forall \boldsymbol{x} \in \Gamma_N \tag{8b}$$

$$u = r^{\lambda} \quad \forall \boldsymbol{x} \in \Gamma_D, \tag{8c}$$

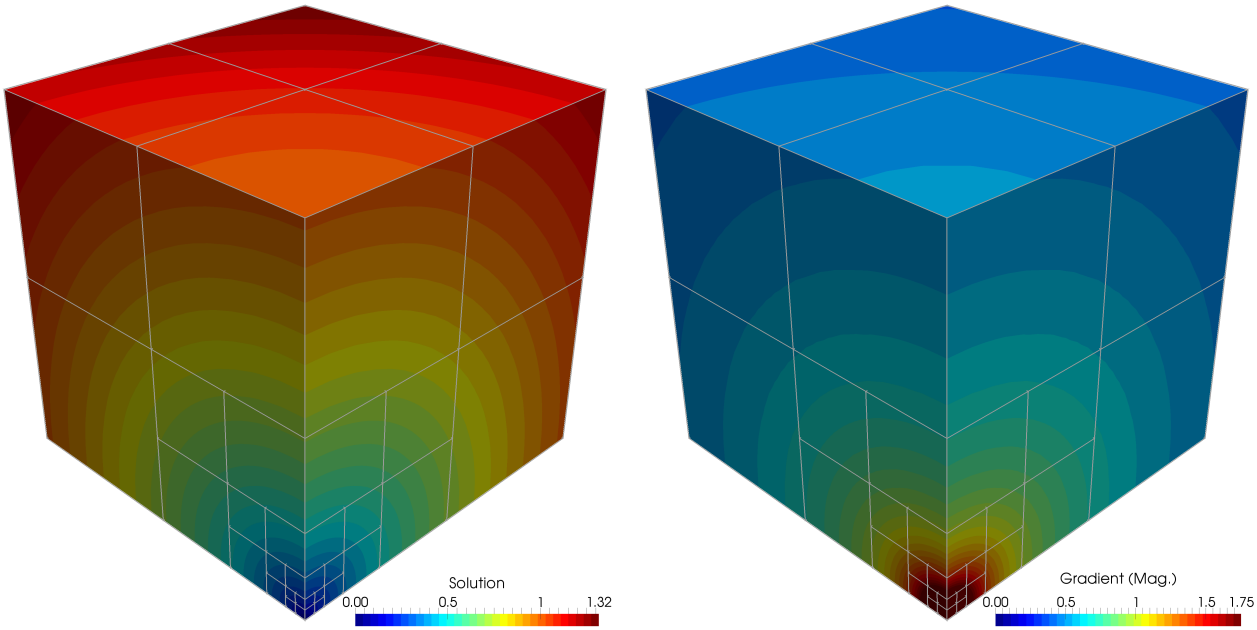
with r being the radial coordinate

$$r = \sqrt{x^2 + y^2 + z^2}, \tag{9}$$



(a) Fichera corner problem (b) Uniform multi-level *hp*-discretization on one octant (c) Graded multi-level *hp*-discretization on one octant

Figure 11: Setup and discretization of Fichera corner problem.



(a) Solution on one octant

(b) Gradient on one octant

Figure 12: Numerical approximation of the solution and its gradient using four levels of multi-level *hp*-refinement and $p = 6$.

$\boldsymbol{n}$ denoting the outward-pointing normal vector on the boundary and choosing $\lambda = \frac{1}{2}$.

The analytical solution to this partial differential equation reads

$$u = r^{\lambda}. \tag{10}$$

Due to the symmetry of the solution, only one octant of the Fichera corner domain is considered:

$$\Omega = (0,1)^3. \tag{11}$$

The six faces of this unit-cube are split into a Dirichlet and Neumann part as follows

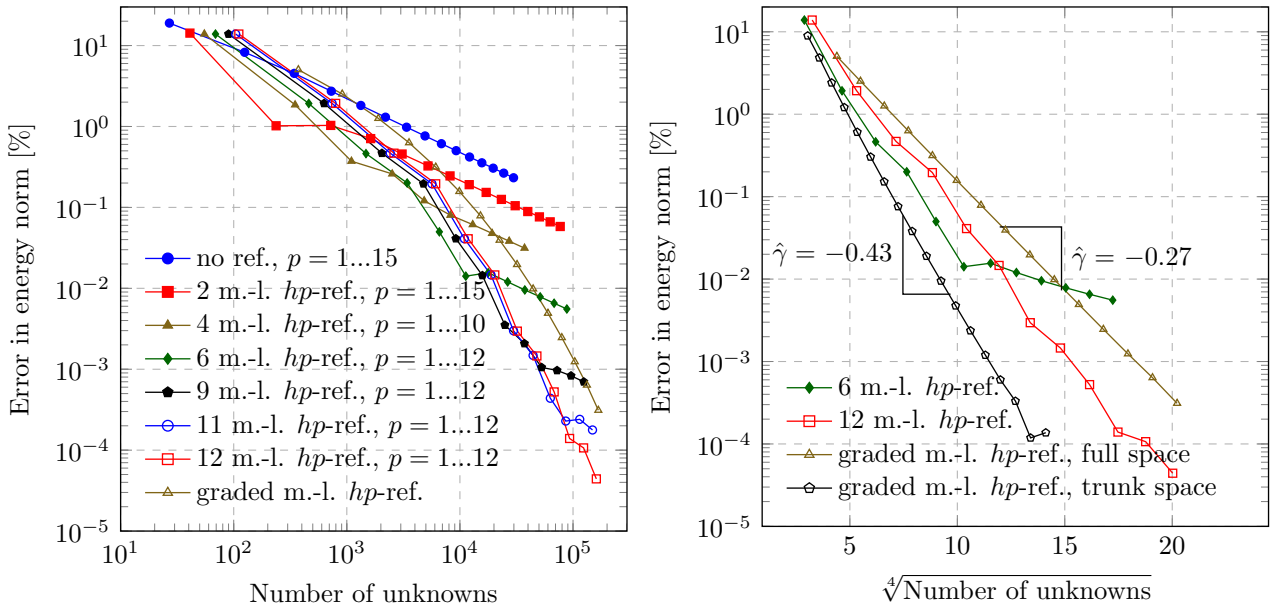
$$\Gamma_D = ([0,1] \times [0,1] \times \{1\}) \cup ([0,1] \times \{1\} \times [0,1]) \cup (\{1\} \times [0,1] \times [0,1]) \tag{12a}$$

$$\Gamma_N = ([0,1] \times [0,1] \times \{0\}) \cup ([0,1] \times \{0\} \times [0,1]) \cup (\{0\} \times [0,1] \times [0,1])\,. \tag{12b}$$

As λ is chosen smaller than one, the solution is singular at the origin $r = 0$. To assess the approximation quality of the proposed refinement scheme for this vertex-singularity, the error in the energy norm (H^1-semi norm) is considered:

$$||e||_E = \sqrt{\frac{|\,\Pi_{\text{ex}} - \Pi_{\text{fe}}\,|}{\Pi_{\text{ex}}}} \cdot 100\%. \tag{13}$$

Thereby, Π_{ex} and Π_{fe} denote the exact and approximated energy. In the present example, the exact



(a) Convergence in log-log scale

(b) Convergence in semi-log scale with $\hat{\gamma} = \frac{\gamma}{\ln(10)}$

Figure 13: Convergence of singular cube example for different refinement configurations using a $2 \times 2 \times 2$ base mesh (used abbreviations: ref. = refinement(s), m.-l. hp = multi-level hp).

reference value reads

$$\Pi_{\text{ex}} = \lambda^2 \int_{\Omega} r^{2\lambda - 2} \, \mathrm{d}\Omega \approx 1.48754835248779 \cdot 10^{-1}. \tag{14}$$

For the numerical approximation of the solution (10), the domain Ω is discretized using a grid of $2 \times 2 \times 2$ elements. This base mesh is refined using the multi-level hp-approach to better capture the singularity. For this purpose, the refinement is steered geometrically towards the origin as depicted in Figure 11. The right hand side of the resulting discrete weak form is strongly over-integrated to accurately capture the singular, non-polynomial right hand side in (8a). The solution and the gradient computed with this setup are depicted exemplarily in Figure 12. In the following, the discretization is adapted using two different refinement strategies.

In the first approach, the solution is approximated using a mesh with a fixed number of refinements. The accuracy is increased by elevating the ansatz order of the leaf-elements *uniformly* as shown in Figures 4c and 11b. As depicted in Figure 13a, pure p-enrichment on the coarse base mesh results in the expected algebraic rate of convergence [65]. This characteristic changes when elevating the ansatz order on a pre-refined mesh. Although the asymptotic convergence is still algebraic, the rates continuously increase over the first refinement steps until the polynomial degree exceeds an optimal value for the given mesh configuration. This S-shaped behavior corresponds to the characteristics of pure p-enrichment on a mesh graded towards a singularity [64]. It is important to note that the turning-point can be shifted to higher accuracy by choosing more refinement levels. This allows to achieve a desired accuracy with a high convergence-rate by choosing the number of refinements accordingly.

By changing the representation into a $\log ||e||$-$\sqrt[4]{N}$ scaling in Figure 13b, the decay of the error appears linear until the asymptotic range is entered. Hence, the pre-asymptotic convergence can be

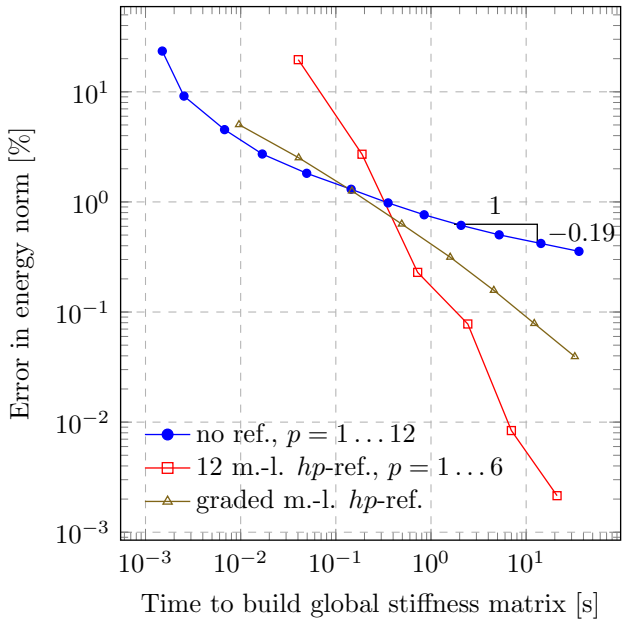
identified as being exponential in the following form:

$$||e||_E \leq C \exp\left(\gamma N^{\theta}\right) \quad \text{with } \theta = 1/4, \tag{15}$$

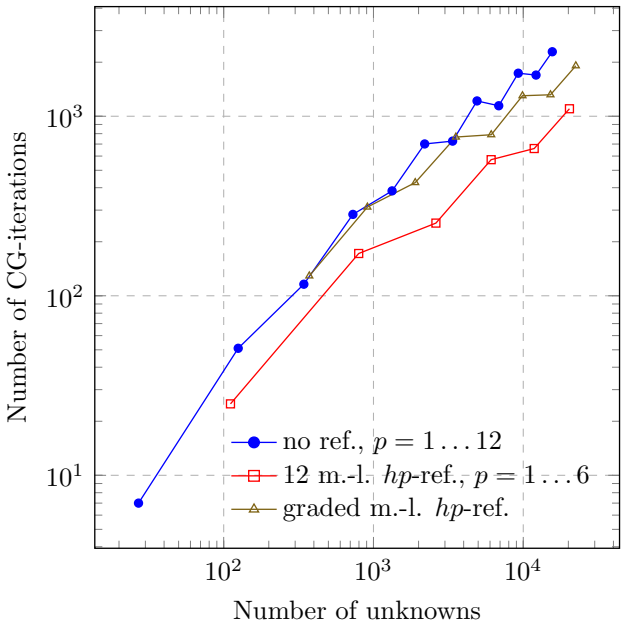
which matches the expectations of an *hp*-approximation in the presence of a vertex-singularity [78, 79]. In this estimate, C and γ are constants independent of the number of degrees of freedom N. These results confirm the findings presented in [63] for two-dimensional applications and show that the multi-level *hp*-approach can also achieve exponential convergence for singular problems in three dimensions.

In the second refinement approach, both h and p are adjusted simultaneously to obtain exponential convergence also in the asymptotic limit. For this purpose, a graded p-distribution is chosen, where the ansatz order decreases linearly over the refinement levels as shown in Figure 11c. Thus, the fine elements near the singular point are of low-order, while the coarse elements use high-order shape functions to approximate the solution where it is smooth. For the convergence graph depicted in Figure 13, the accuracy is increased by introducing an extra level of refinement at the singularity, while the ansatz order of all other leaf elements is elevated. In this way, exponential convergence is obtained also asymptotically. Interestingly, the pre-asymptotic convergence obtained with the uniform multi-level *hp*-refinement is slightly better than the results of the graded approach. This indicates that the linear distribution of the ansatz order chosen for the graded approach is not the best choice possible. Therefore, the authors expect that the results can be improved by using more sophisticated error- and smoothness-indicator schemes. It is further noteworthy that the refinement scheme can be combined with the trunk space, described e.g. in [64, 65], which yields a further increase of the convergence rates as depicted in Figure 13b.

The final aspect addressed in this study is the computational efficiency of the different refinement schemes. In particular, the cost of the additional overhead of the hierarchical meshes is of interest. To



(a) Runtime comparison using a single thread on an Intel(R) Core(TM) i7-4790 CPU @ 3.60GHz.



(b) Number of CG-iterations necessary for solving the FE-equation system up to an accuracy of 10^{-15} using a Jacobi pre-conditioning.

Figure 14: Comparison of refinement schemes in terms of computational performance.

assess this point, Figure 14a depicts the decay of the error in the energy norm versus the time required to integrate and assemble the global stiffness matrix. The time needed for building the right-hand-side vector is not considered as the singular load function demands for a significant over-integration, which would distort the analysis. All measurements are performed using the same in-house code and a single thread on an Intel(R) Core(TM) i7-4790 CPU running at a clock speed of 3.60GHz.

The comparison presented in Figure 14a shows that the algebraic convergence character of a uniform order elevation carries over to the computational performance if p is sufficiently high. In contrast, both *hp*-schemes show an exponential convergence. Although this exponential character is only slightly pronounced, the obtained convergence rate is significantly higher than the rate resulting from a pure p-refinement. For the present example, the three refinement schemes break even at an error level of about 1%. Beyond this point, the hierarchical *hp*-refinement achieves a higher accuracy than a pure p-elevation for the same computational time.

The second aspect to be considered is the solution of the finite element equation system. To this end, Figure 14b compares the number of conjugate gradient iterations needed to solve the respective equation system up to an accuracy of 10^{-15} using a Jacobi pre-conditioning scheme, *cf.* [80]. The depicted results demonstrate that, when using an *hp*-refinement, the number of necessary CG-iterations is not more than in case of a pure order elevation. This indicates that the use of high-order shape functions is dominating the overall convergence characteristic of the CG-scheme, and the use of finer overlay meshes has no negative effect.

4.2 Shock problem

The aim of the following example is to assess the applicability of the suggested *hp*-scheme to resolve sharp solution features. To this end, the well-known shock problem is considered by following e.g. [81].

The domain is defined as

$$\Omega = (0,1)^3, \tag{16}$$

and the radial coordinate is considered with respect to a shifted origin:

$$r = \sqrt{(x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2} \quad \text{with } x_0 = y_0 = z_0 = -1/4. \tag{17}$$

The manufactured solution is then defined as a shock-like function:

$$u = \tan^{-1}(\alpha(r - r_0)) \quad \text{with } r_0 = \sqrt{3} \quad \text{and } \alpha \in \{40, 80, 160\}. \tag{18}$$

Here, the scalar factor α determines the sharpness of the shock. This solution is approximated by solving the following partial differential equation numerically

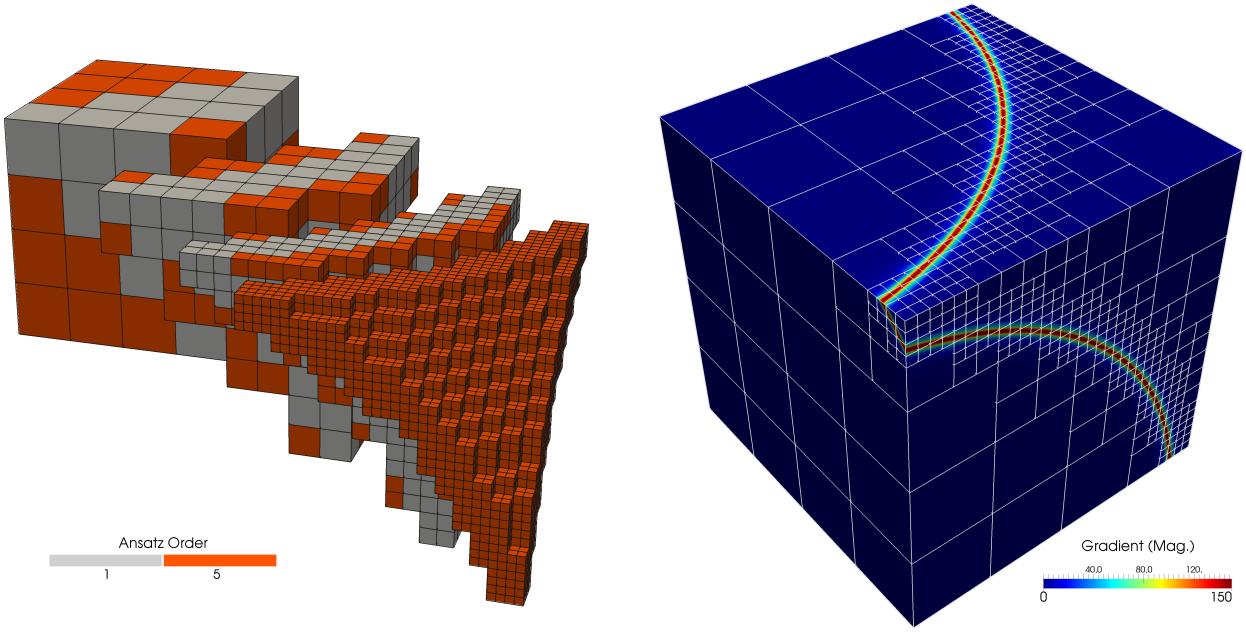
$$\Delta u = s \quad \forall \boldsymbol{x} \in \Omega \tag{19a}$$

$$\nabla u \cdot \boldsymbol{n} = g \quad \forall \boldsymbol{x} \in \partial\Omega, \tag{19b}$$

in which the source term s and the surface flux g follow from the previously defined solution:

$$s = 2\frac{\alpha}{rf}\left(1 - \alpha^2 \frac{r}{f}(r - r_0)\right) \quad \text{and} \quad g = \frac{\alpha}{rf}\begin{bmatrix} x - x_0 \\ y - y_0 \\ z - z_0 \end{bmatrix} \cdot \boldsymbol{n} \quad \text{with } f = 1 + \alpha^2 (r - r_0)^2. \tag{20}$$

To render the solution unique, an additional Dirichlet boundary condition is applied at the point $(1,1,1)$. As in the previous example, the error is considered in the energy norm using the following



(a) Overlay mesh construction for the multi-level *hp* refinement.

(b) Numerical approximation of the gradient ($\alpha = 160$) using three levels of multi-level *hp*-refinement and $p = 5$

Figure 15: Local *hp*-mesh refinement for the shock problem.

reference values:

$$\Pi_{\text{ex}} = \int_{\Omega} \left(\frac{\alpha}{f} \right)^2 \mathrm{d}\Omega \tag{21}$$

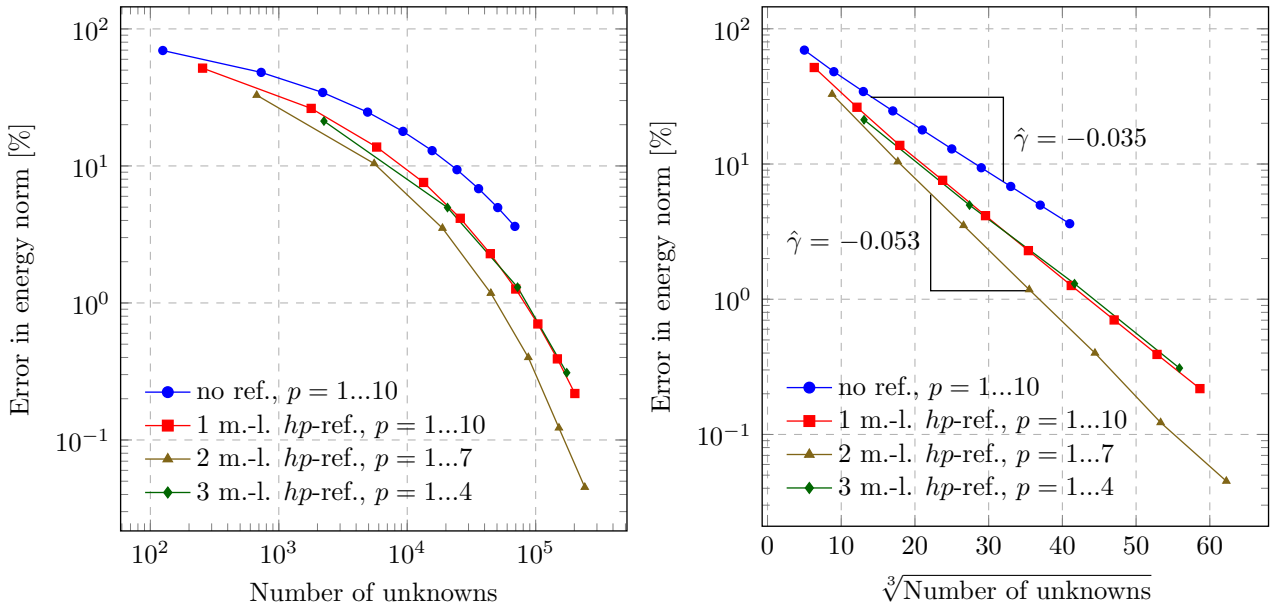
$$\Pi_{\text{ex}}(\alpha = 40) \approx 2.23858386360232 \cdot 10^1$$
$$\Pi_{\text{ex}}(\alpha = 80) \approx 4.48424691179860 \cdot 10^1$$
$$\Pi_{\text{ex}}(\alpha = 160) \approx 8.97291532487928 \cdot 10^1$$

For the numerical approximation of the solution, the domain is first discretized using $4 \times 4 \times 4$ elements. This base mesh is refined in the vicinity of the shock by superposing multiple levels of finer overlay elements. Similar to the previous example, the refinement is steered geometrically towards the surface of the sphere described by the shock ($r = r_0$). This results in the nested, three-dimensional mesh structure presented in Figure 15a. Figure 15b exemplarily depicts one numerical approximation computed on the locally refined discretization.

In the following, the convergence of the error in the energy norm is studied. To this end, the approximation accuracy is increased by elevating the polynomial degree of the shape functions uniformly while keeping the number of refinements fixed. In Figure 16, the convergence results are shown for a mild shock with $\alpha = 40$. In this setting, the pure p-extension on the non-refined base mesh already yields exponential convergence in the form

$$||e||_E \leq C \exp\left(\gamma N^{\theta}\right) \quad \text{with } \theta = 1/3, \tag{22}$$

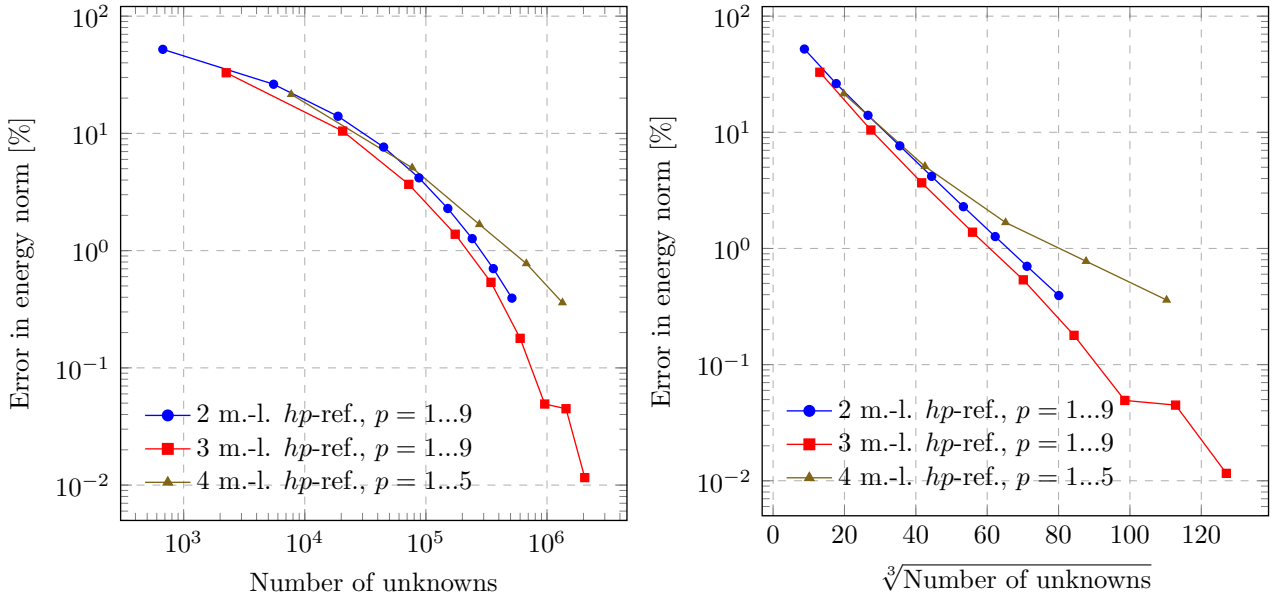
which—according to e.g. [79]—is to be expected for the regular solution under consideration. However, the convergence rate can be further improved by using one- or two-mesh refinements. As shown



(a) Convergence in log-log scale

(b) Convergence in semi-log scale with $\hat{\gamma} = \frac{\gamma}{\ln(10)}$

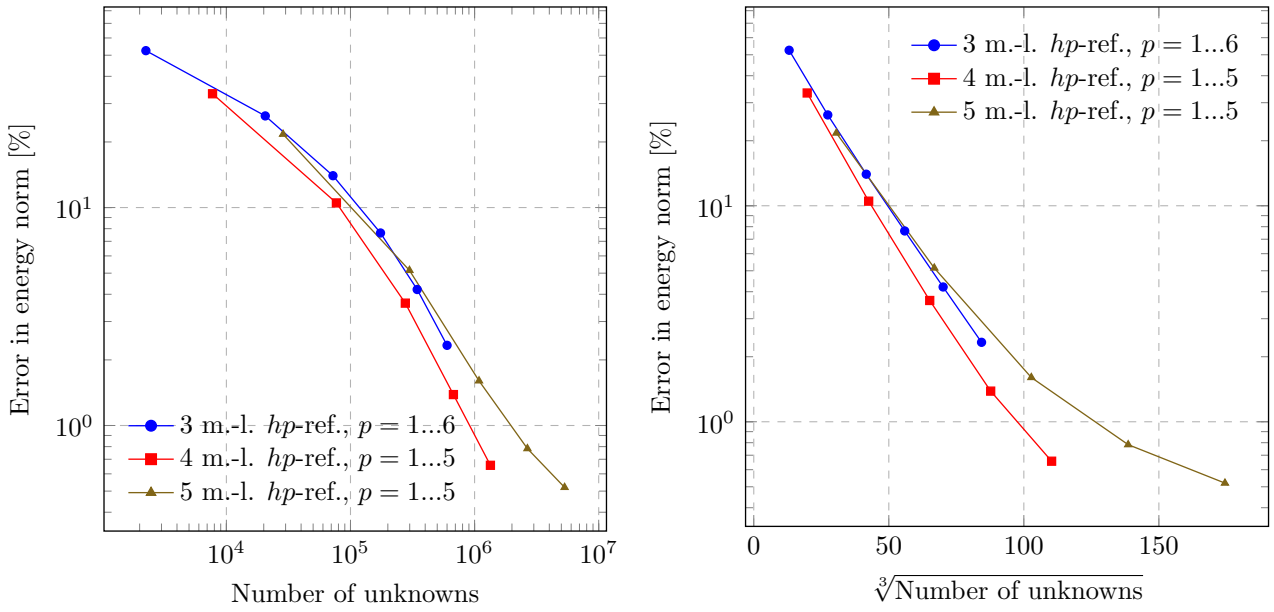
Figure 16: Convergence of the shock problem ($\alpha = 40$) for different refinement configurations (used abbreviations: ref. = refinement(s), m.-l. hp = multi-level hp).



(a) Convergence in log-log scale

(b) Convergence in semi-log scale

Figure 17: Convergence of the shock problem ($\alpha = 80$) for different refinement configurations (used abbreviations: ref. = refinement(s), m.-l. hp = multi-level hp).



(a) Convergence in log-log scale

(b) Convergence in semi-log scale

Figure 18: Convergence of the shock problem ($\alpha = 160$) for different refinement configurations (used abbreviations: ref. = refinement(s), m.-l. *hp* = multi-level *hp*).

in Figure 16b, the finer elements in the vicinity of the shock increase the exponential convergence rate γ by about 50%. When using three or more levels of refinement, the convergence rates decrease again because the finer elements introduce unnecessary unknowns, which over-resolve the mild shock.

This observation is confirmed when increasing the sharpness of the transition to $\alpha = 80$ and 160, respectively, as shown in Figures 17 and 18. In these configurations, the width of the shock is significantly thinner. Thus, an accurate resolution demands for a small element size. Accordingly, the optimal number of refinement levels increases to three and four, respectively. Further h-refinement decreases the rates of convergence again.

4.3 Dynamic multi-level *hp*-refinement and -coarsening

The data structure outlined in Section 3 allows for a dynamic change of the discretization during the simulation runtime. The applicability of this feature is analyzed in the following third example. For this purpose, a linear elastodynamic problem is considered:

$$\rho \ddot{\boldsymbol{u}} = \nabla \cdot \boldsymbol{\sigma} \qquad \forall (\boldsymbol{x}, t) \in \Omega \times (0, T] \tag{23a}$$

$$\boldsymbol{\sigma} = \boldsymbol{C} : \boldsymbol{\varepsilon} \qquad \forall (\boldsymbol{x}, t) \in \Omega \times (0, T] \tag{23b}$$

$$\boldsymbol{\varepsilon} = \frac{1}{2} \left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^\top \right) \qquad \forall (\boldsymbol{x}, t) \in \Omega \times (0, T] \tag{23c}$$

$$\text{with } \Omega = (-2, 2) \times (-1, 1) \times (0, 1).$$

Here, ρ denotes the density, $\boldsymbol{\sigma}$ the stress tensor, $\boldsymbol{C}$ the material tensor, $\boldsymbol{\varepsilon}$ the engineering strain, and $\boldsymbol{u}$ the displacement vector. For the present example, the density is chosen as $\rho = 7850\,\mathrm{kg/m^3}$, the Young's modulus as $E = 210 \cdot 10^9\,\mathrm{N/m^2}$ and Poisson's ratio as $\nu = 0.3$. Furthermore, the end time T is set to 10 s.

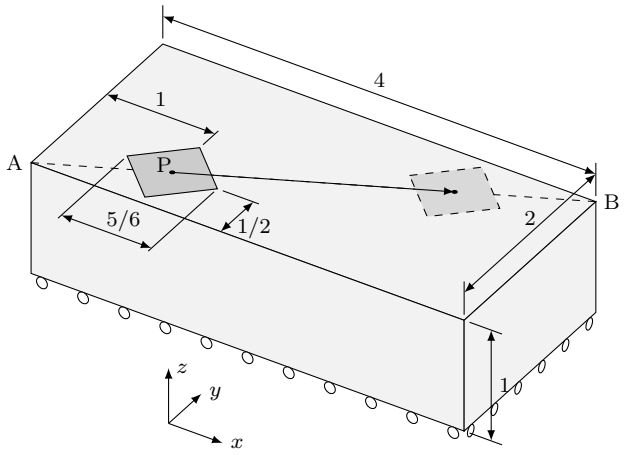


Figure 19: Setup of traveling load example.

As depicted in Figure 19, the domain is simply supported on the lower face:

$$u_z = 0 \quad \forall(\boldsymbol{x}, t) \in \Gamma_D \times (0, T] \quad \text{with } \Gamma_D = \{(x, y, z) \in \Omega \,:\, z = 0\}. \tag{24}$$

To prevent rigid body motions, the displacement is further fixed at two additional points as follows:

$$u_x = u_y = 0 \quad \text{on } \boldsymbol{x} = (0, 0, 0), t \in (0, T] \tag{25}$$
$$u_y = 0 \quad \text{on } \boldsymbol{x} = (2, 0, 0), t \in (0, T]. \tag{26}$$

A vertical traction

$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = - \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \forall(\boldsymbol{x}, t) \in \Gamma_q(t) \times (0, T] \tag{27}$$

is applied onto the squared load surface depicted in dark gray in Figure 19:

$$\Gamma_q(t) = \{(x, y, z) \in \Omega : |x - P_x(t)| + |y - P_y(t)| \leq w/2, z = 1\} \quad \text{with } w = \frac{5}{6}. \tag{28}$$

During the simulation runtime, the center point P—and with it the load surface—moves across the top surface of the domain as follows:

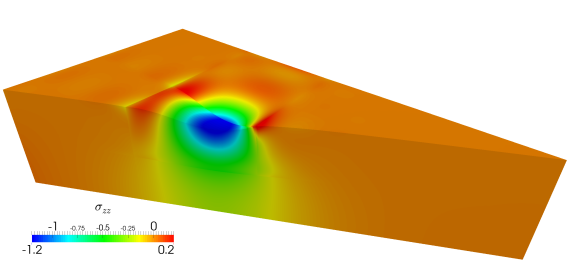
$$\begin{bmatrix} P_x(t) \\ P_y(t) \end{bmatrix} = - \begin{bmatrix} 1.0 \\ 0.5 \end{bmatrix} + \frac{t}{T} \begin{bmatrix} 2 \\ 1 \end{bmatrix}. \tag{29}$$

On the remaining part of the boundary

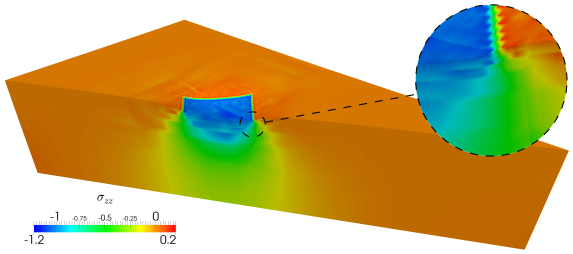
$$\Gamma_0 = (\partial\Omega \setminus \Gamma_D) \setminus \Gamma_q, \tag{30}$$

homogeneous Neumann boundary conditions are applied

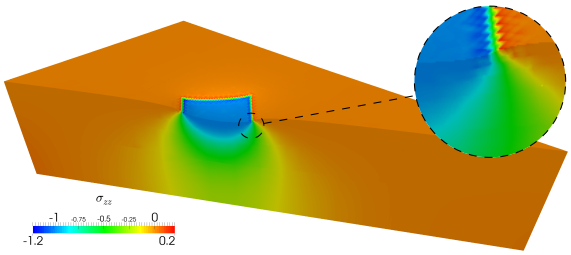
$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \boldsymbol{0} \quad \forall(\boldsymbol{x}, t) \in \Gamma_0(t) \times (0, T]. \tag{31}$$



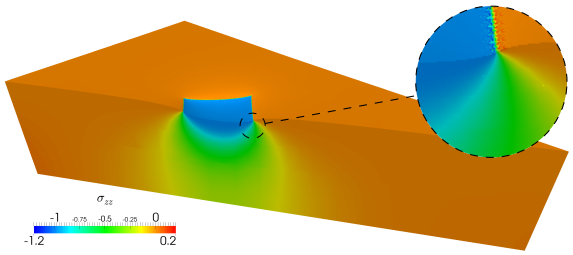
(a) No refinement ($p = 4$, 4,131 unknowns)



(b) *hp-d*-refinement [54, 66] ($p = 4$, $k = 6$, 10,257 unknowns)



(c) Graded multi-level *hp*-refinement (present approach, $p = 4$, $k = 6$, 51,165 unknowns)



(d) Uniform multi-level *hp*-refinement (present approach, $p = 4$, $k = 6$, 472,215 unknowns)

Figure 20: Comparison of vertical stress distribution σ_{zz} at $t = 38\%$ using different refinement strategies. For better visualization, the deformation is magnified by a factor of $5 \cdot 10^{10}$. Used abbreviations: p: polynomial degree of shape functions; k: number of refinement levels.

To complete the problem definition, the initial displacement and velocity are set to zero:

$$\boldsymbol{u} = \dot{\boldsymbol{u}} = \boldsymbol{0} \quad \forall \boldsymbol{x} \in \Omega,\, t = 0. \tag{32}$$

For the numerical approximation of the solution, the multi-level *hp*-approach is combined with the Newmark time-stepping scheme using 48 time steps [20, 21, 82]. In a first approach, the spatial domain Ω is discretized by $4 \times 2 \times 2$ elements of order four. To avoid any side-effects resulting from the numerical quadrature of the discontinuous traction (27), a high-resolution integration mesh of 64×64 integration cells per element is used to accurately capture the load on the top surface. This integration mesh is used for all following mesh refinement configurations.

From the boundary condition (27) applied on the top surface, it is evident that the vertical stress σ_{zz} on the top surface should be -1 within the load surface Γ_q and zero outside. Any deviation from this expectation can thus be identified as a discretization error.

As depicted in Figure 20a, the non-smooth solution cannot be represented with the coarse basemesh. Although the high-order elements are able to capture the global deformation, the numerical stresses suffer from severe oscillations. In particular, the rectangular load surface cannot be resolved sharply. In Figure 21, this is visualized by depicting the vertical stress σ_{zz} along the diagonal path $\overline{AB}$ across the top surface. Here, the four segments along the abscissa correspond to the boundaries of the four elements cut by the path. The figure demonstrates that the oscillations are not limited to the element incorporating the singular points. Instead, the error only decays after crossing multiple elements edges and thus spreads over large parts of the domain.

The accurate resolution of the discontinuous stress state thus demands for a local refinement that follows the moving traction. As depicted exemplarily for different time steps in Figure 22, the used superposition approach allows to refine elements as soon as the load surface Γ_q enters an element

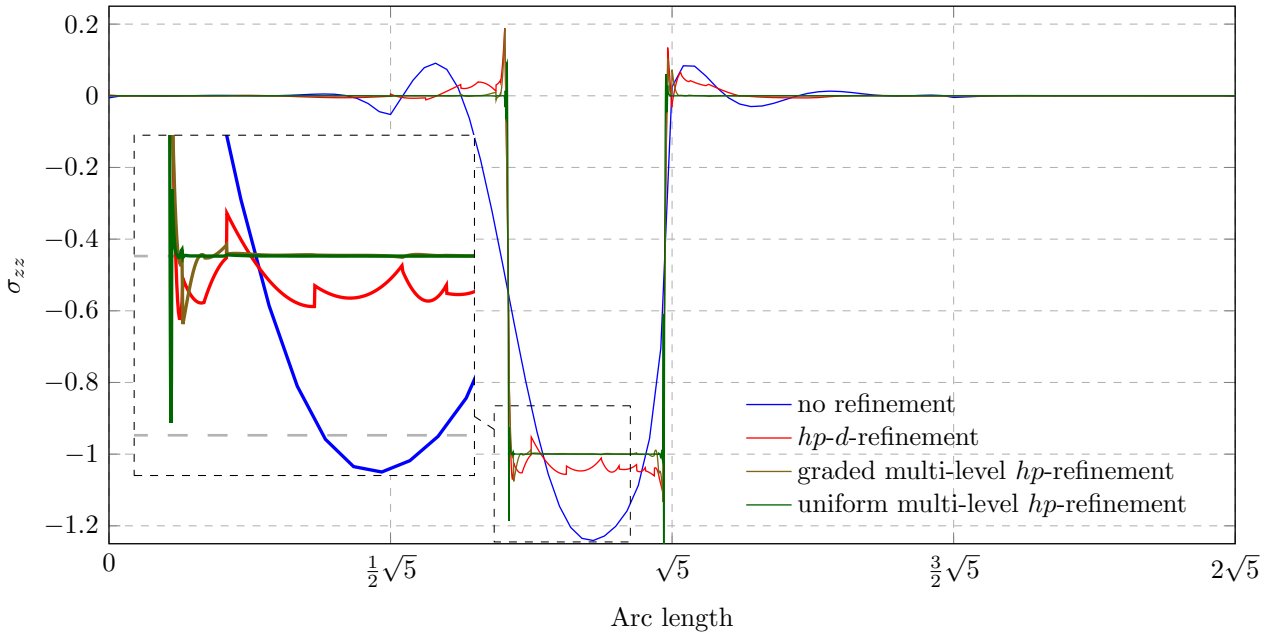
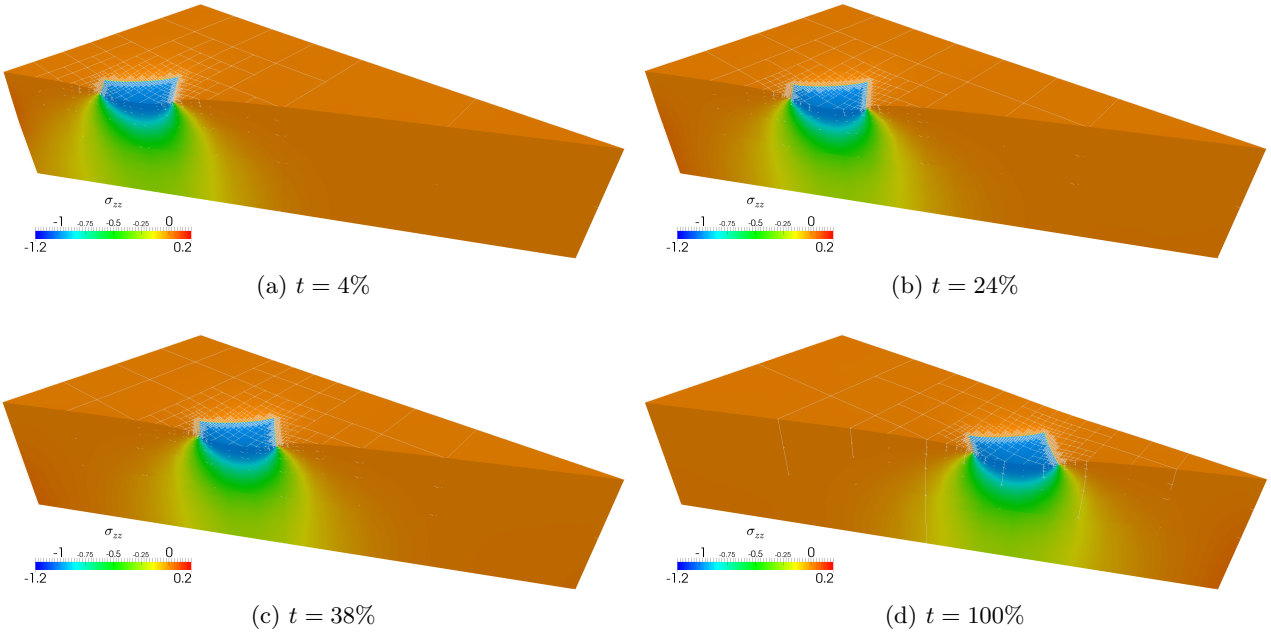


Figure 21: Vertical stress σ_{zz} along top of the block (cut $\overline{AB}$) at $t = 38\%$.



(a) $t = 4\%$ (b) $t = 24\%$

(c) $t = 38\%$ (d) $t = 100\%$

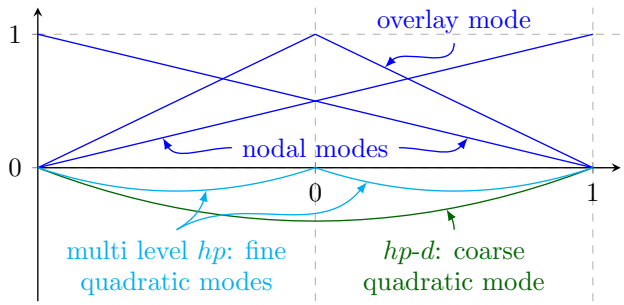
Figure 22: Vertical stress distribution σ_{zz} at different time steps using the uniform multi-level *hp*-refinement strategy ($p = 4$, $k = 6$). For better visualization, the deformation is magnified by a factor of $5 \cdot 10^{10}$.

and to coarsen the discretization again when the singularity has left the element. This continuous change of the discretization allows the refinement zone to follow the moving discontinuity, which is traveling through the domain. Similar to the two the previous examples, the refinement is therefore guided by geometric information.

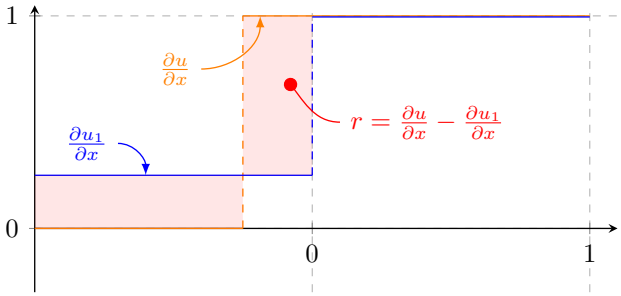
In a first step, the refinement is carried out following the *hp-d* approach (see e.g. [54, 66] and Figure 4a). Here, the base mesh remains unchanged, and the accuracy is increased by superposing six levels of overlay-elements with linear shape functions only. The results depicted in Figure 20b demonstrate a significant improvement in the approximation quality. In particular, the edges of the load surface are resolved more sharply. However, as shown in Figure 21, the oscillations in the numerical stress solution are *not* bounded to the finest element level but still spread over large parts of the domain—although with a reduced amplitude. This remaining approximation error is explained by the simplified example depicted in Figure 23. Here, the standard basis is formed by the two nodal modes and one quadratic mode. Following the *hp-d*-refinement idea, one linear overlay mode is added to increase the approximation accuracy. The final basis is thus composed of a piece-wise linear contribution u_1 and a higher-order contribution u_2:

$$u_{\text{num}} = u_1 + u_2. \tag{33}$$

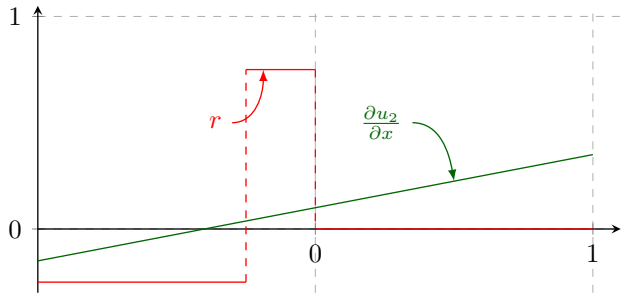
Motivated by the discontinuous stress field considered in the current example, the linear contribution u_1 is used to approximate a function u with a discontinuous derivative in Figure 23b. In the depicted scenario, the additional linear overlay mode breaks the C^∞-continuity of the shape functions in the middle of the coarse base element. Thus, the derivative on the left and the right part of the element



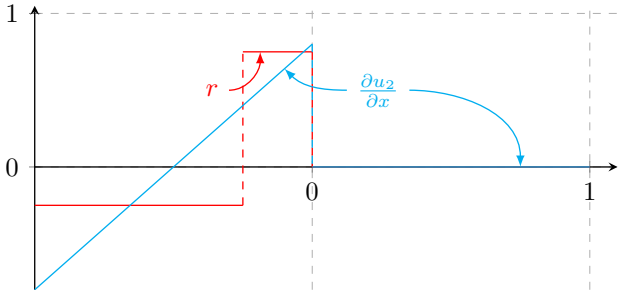
(a) Refined shape functions



(b) Approximation of discontinuous strain with nodal and overlay modes



(c) *hp-d*-approximation of residual with coarse high-order mode: The residual is not captured exactly on neither side of the coarse element



(d) Multi-level *hp*-approximation of residual with fine high-order modes: The residual is captured exactly on the right side of the coarse element, which limits the approximation error to the left side.

Figure 23: Approximation of singular function using the *hp-d*- and the multi-level *hp*-refinement strategy.

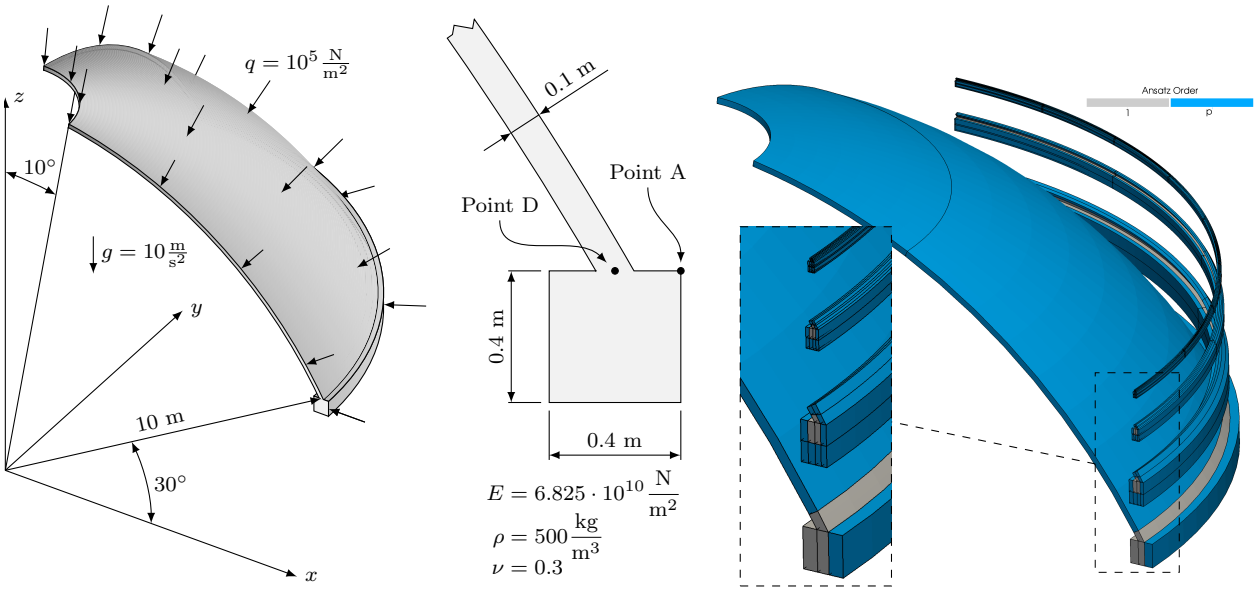
can be approximated independently. For the chosen example, the numerical solution is even exact on the right part of the element. However, the remaining residual in the left part of the element cannot be reduced by the *single* quadratic mode u_2 as this function has support on the original coarse element. Accordingly, any contribution of this global mode will degrade the solution on the right part again as indicated in Figure 23c. This deficiency cannot be compensated by simply increasing the number of refinements since the high-order shape functions remain on the coarse base element. Thus, the *hp-d* refinement only breaks the continuity of the shape functions, while their support remains unchanged. Therefore, the superposition of a linear mode does not correspond to an *h*-refinement of the high-order base element.

This is fundamentally different when using the graded or uniform multi-level *hp*-approach as illustrated in Figure 23d. Since the high-order shape functions are moved from the base to the leaf-elements, their support is minimized. Only this corresponds to a high-order *h*-refinement and yields a substantial increase of the approximation quality as shown in Figures 20c and d. Both versions of the method limit the spread of the oscillations. In particular the uniform multi-level *hp*-approach renders the traction surface extremely sharp, whereas the graded refinement strategy appears to be a good compromise between accuracy and efficiency.

4.4 Multi-level *hp*-refinement for curved geometries

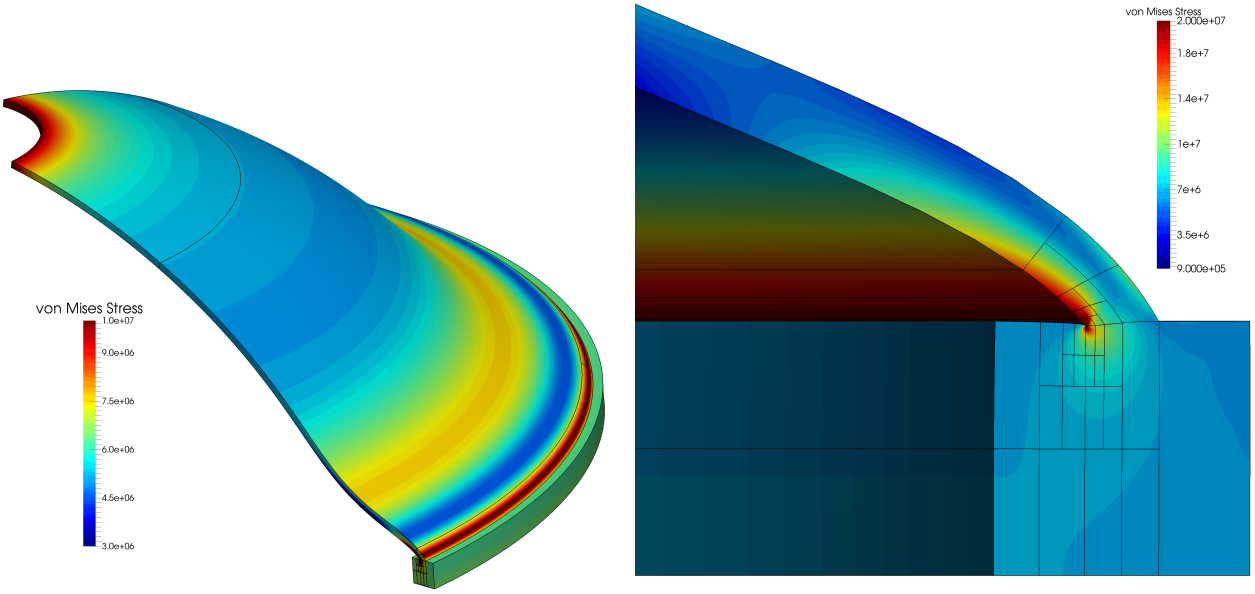
The aim of this last example is to study the applicability of the suggested refinement scheme for more complex examples including curved geometries. To this end, the three-dimensional extension of the Girkmann-problem [83] introduced in [84] is considered. This benchmark was already investigated in the context of shell formulations [85], isogeometric analysis [86–88], T-splines [89], and adaptive volumetric shells [90, 91].

The geometric and mechanical properties are depicted in Figures 24a and b. Due to the symmetry of the problem, only one quarter of the shell is considered. Accordingly, the respective displacements normal to the yz- and zx-planes are constrained to zero. The stiffener is supported at the bottom surface by constraining the vertical displacement u_z to zero. The structure is loaded by self-weight



(a) Problem definition (b) Details of stiffener (c) Refined discretization

Figure 24: Hemispherical shell with stiffener.



(a) Stress distribution on outer shell

(b) Zoom-in on connection between shell and stiffener

Figure 25: von Mises stress in $\mathrm{N/m^2}$ on deformed structure (scaling factor = 500, $p = 10$).

and a pressure q on the outer surface of the shell and the stiffener.

For the numerical approximation of the resulting deformation, the domain is first discretized using six high-order, hexahedral elements depicted in Figure 24c. As shown in [84], such a coarse base mesh is capable of capturing the global deformation state accurately. Due to the applied load, the vertical displacement u_z is constant along the latitude. Hence, the use of linear polynomials in this direction is sufficient. For this reason, the displacement is approximated by a trunk tensor space formed by the following polynomial degree template

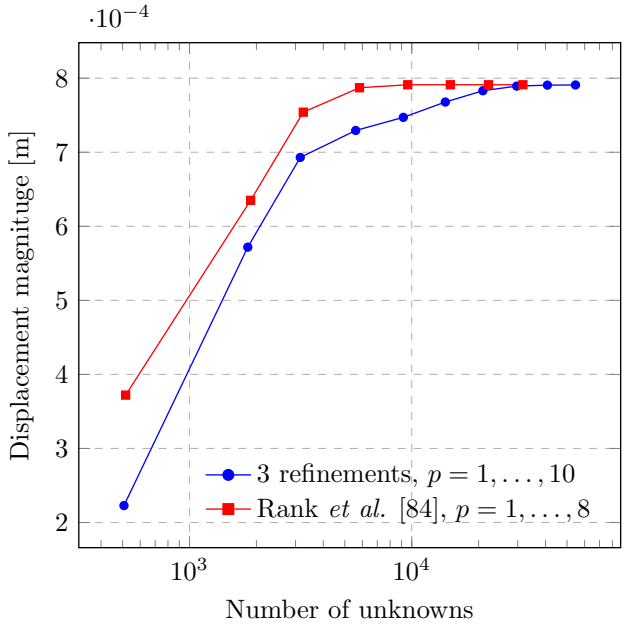
$$\boldsymbol{P} = \begin{array}{c} \\ r \\ s \\ t \end{array} \begin{array}{c} \begin{array}{ccc} u_x & u_y & u_z \end{array} \\ \begin{pmatrix} p & p & p \\ p & p & p \\ p & p & 1 \end{pmatrix} \end{array}. \tag{34}$$

Here, p denotes the polynomial degree used for the remaining solution components. The variables (r, s, t) represent the coordinate directions in the reference space of the element, where t corresponds to the latitude direction.

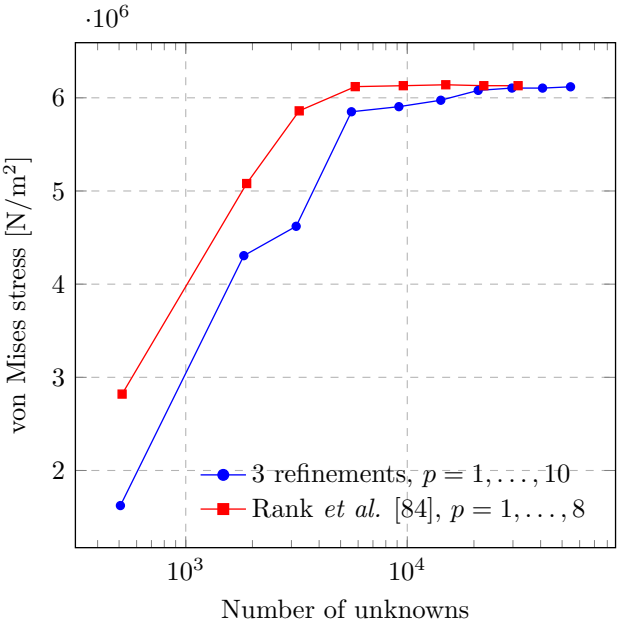
To capture the expected singularity, the base discretization is refined at the connection between the shell and the stiffener by superposing three levels of overlay elements as shown in Figure 24c. As the overlay elements are defined in the index space of their corresponding parent elements, the shape of these fine elements naturally follows the curved geometry.

The approximation obtained with the refined discretization is depicted in Figure 25. The results show that the coarse base mesh efficiently captures the global deformation and the resulting state of stress. At the same time, the local refinement accurately resolves the stress concentration at the re-entrant corner.

To assess the quality of the numerical approximation in more detail, displacements and stresses are recorded at the points labeled A and D in Figure 24b. The results are compared to the reference

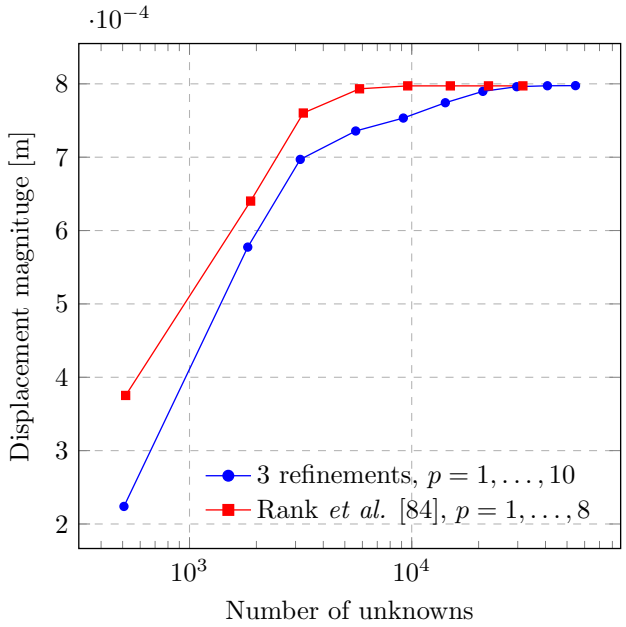


(a) Displacement magnitude at point A

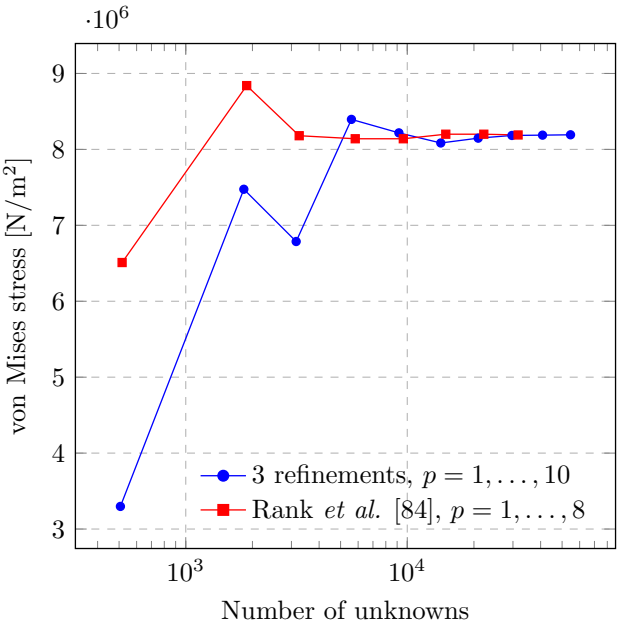


(b) Von Mises stress at point A

Figure 26: Convergence of displacement and stress at point A.

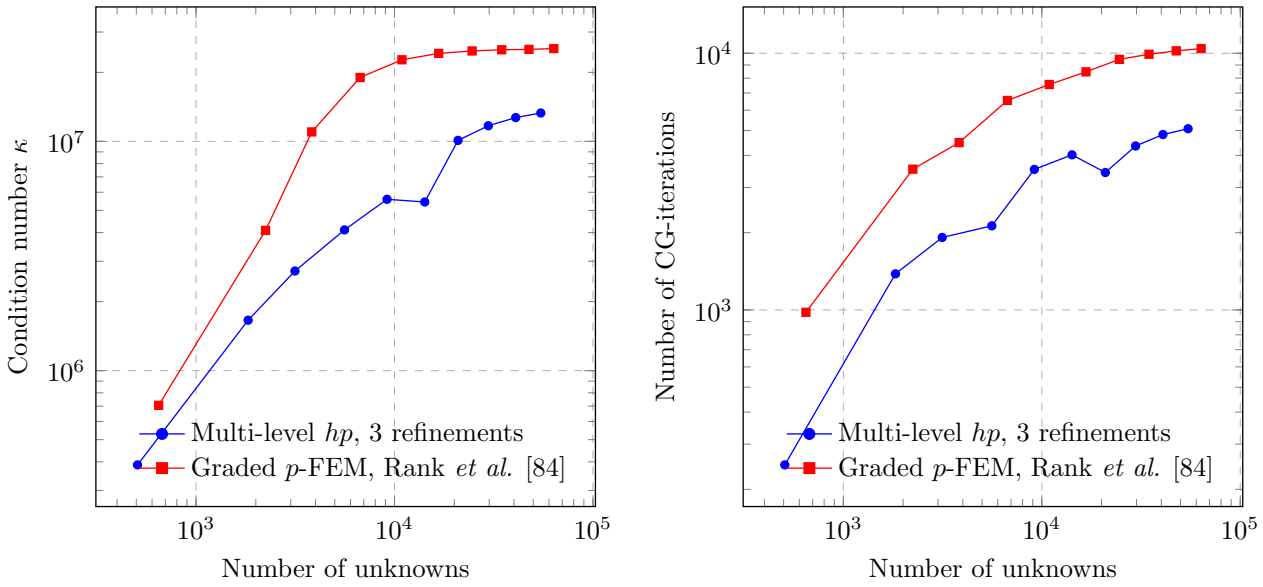


(a) Displacement magnitude at point D



(b) Von Mises stress at point D

Figure 27: Convergence of displacement and stress at point D.



(a) Estimated condition number computed using the scheme suggested in [92] implemented in MATLAB `condest`.

(b) Number of CG-iterations necessary for solving the FE-equation system up to an accuracy of 10^{-15}.

Figure 28: Comparison of computational performance for $p = 1 \dots 10$ after application of a Jacobi pre-conditioner.

values presented in [84] in Figure 26 and 27. The comparisons show an excellent agreement of the convergence limits. This demonstrates that the advantageous approximation properties of the multi-level *hp*-scheme, observed for the simply-shaped benchmarks discussed in the previous sections, also carry over to complex geometries. The solutions obtained by [84] show a slightly faster convergence because Rank *et al.* used a problem-tailored mesh that was graded with a geometric progression of 0.15 towards the re-entrant corners. To achieve a similar performance, the presented scheme would need to be extended to also feature anisotropic refinement mimicking general geometric progressing factors. These extensions are subjects of ongoing research.

The final aspect to be considered is the computational performance of the suggested refinement scheme for this benchmark with a curved geometry. To this end, the condition number of the stiffness matrix obtained by the multi-level *hp*-refinement scheme is compared to the reference obtained by a pure *p*-elevation on the geometrically graded mesh used by Rank *et al.* in Figure 28a. The depicted results demonstrate that the hierarchical overlay approach yields a condition number that is in the same order of magnitude but slightly lower than the classically graded *p*-mesh. As shown in Figure 28b, this advantage is also reflected in the number of conjugate gradient iterations necessary to solve the respective finite element equation systems up to an accuracy of 10^{-15}. These results indicate that the orthogonality properties observed for one-dimensional problem partially carry over to higher-dimensions.

5 Conclusion and outlook

The work at hand presents a three-dimensional *hp*-formulation of the finite element method, which a) circumvents the implementational difficulties caused by hanging nodes and b) yields the same approximation accuracy as conventional *hp*-methods. To this end, the multi-level *hp*-refinement scheme introduced in [63] was extended for three-dimensional applications. Instead of the conventional refine-

by-replacement idea, the current method refines the discretization by superposing the base mesh with finer overlay elements in the domains of interest. This change of paradigm avoids hanging nodes by definition, which significantly eases the implementational complexity of the scheme.

It was shown that the formulation derived for the two-dimensional setting in [63] naturally extends to volumetric meshes without increasing the complexity of the rule set for a linear independent and compatible finite element basis. In this way, a three-dimensional *hp*-formulation of the finite element method was derived in which hanging nodes do not pose a challenge. This allows for a highly flexible discretization kernel featuring arbitrary irregular meshes, which can be refined and coarsened continuously throughout the simulation runtime.

By means of the Fichera corner benchmark, the first example demonstrated that—even in the presence of a singularity—the suggested refinement scheme yields an exponential convergence of the approximation error with respect to both, the number of unknowns and the integration time of the stiffness matrix. In the second example, the refinement scheme resolved the sharp transition of a three-dimensional shock problem with high accuracy. It was shown that the scheme also yields excellent results in case of complex, three-dimensional refinement patterns and significantly increases the exponential rates of convergence. In the third example, the flexibility of the discretization kernel was exploited to resolve a moving singularity caused by a traveling discontinuous load. To this end, the discretization was refined and coarsened continuously to keep the refinement zone local to the current position of the singularity. The presented results demonstrate that this dynamic mesh sharply resolves the discontinuous stress field throughout the simulation runtime and efficiently prevents oscillations of the stress field to spread over large parts of the domain. In a fourth example, it was demonstrated that this high fidelity is not limited to block-like domains by applying the proposed refinement scheme to the hemispherical shell with stiffener benchmark introduced in [84]. It was shown that the refined elements naturally follow the curved geometry of the parent element. The obtained displacement and stress results showed excellent agreement with the reference values provided in the literature, which demonstrates that the advantageous approximation properties of the multi-level *hp*-scheme carry over to curved geometries. Further, the condition numbers obtained by the graded *p*-mesh used in [84] and the hierarchically refined mesh used in this work were compared. The results demonstrate that both schemes yield condition numbers in the same order of magnitude, with the condition number obtained by the hierarchical refinement being a factor two to four lower for the considered polynomial degrees. It was further shown that this advantage is also reflected in the number of conjugate gradient iterations necessary to solve the equation system.

In conclusion, the suggested refine-by-superposition idea is a valuable alternative to conventional refine-by-replacement approaches if the respective application can profit from the high accuracy achievable by *hp*-discretization schemes. Of course, this work does not provide a full-fledged analysis suite but rather gives a proof of concept leaving many points open. In the first place, this includes the optimization of the implementation to increase the computational performance of the discretization kernel. Further, the refinement has to be coupled with a suitable error and smoothness estimator to allow for an automatic adaptive refinement scheme. A third open point is the incorporation of different element types to allow for more than hex-only meshes. Also the types of refinement could be extended to feature also anisotropic refinement and arbitrary ratios for element sub-division. Further, the performance could be improved using *p*-FEM specific optimizations such as sum factorization [93], a suitable parallelization of the assembly procedure, or multi-grid solution techniques that exploit the hierarchical structure of the finite element basis. Another open point is the application of the refinement scheme to more complex physical problems, including but not limited to applications with large strains, contact, and crack-growth or problems in the field of fluid mechanics. Parts of these open points are subjects of ongoing research.

Acknowledgements

The authors gratefully acknowledge the financial support of the German Research Foundation (DFG) under Grant RA 624/22-1 and RA 624/27-1.

References

[1] W. Gui and I. Babuška, "The h, p and h-p versions of the finite element method in 1 dimension Part I: The error analysis of the p-version," *Numerische Mathematik*, vol. 49, no. 6, pp. 577–612, 1986.

[2] W. Gui and I. Babuška, "The h, p and h-p versions of the finite element method in 1 dimension Part II: The error analysis of the h-and h-p versions," *Numerische Mathematik*, vol. 49, no. 6, pp. 613–657, 1986.

[3] W. Gui and I. Babuška, "The h, p and h-p versions of the finite element method in 1 dimension Part III: The Adaptive h-p Version," *Numerische Mathematik*, vol. 49, no. 6, pp. 659–683, 1986.

[4] B. Guo and I. Babuška, "The h-p version of the finite element method Part 1: The basic approximation results," *Computational Mechanics*, vol. 1, no. 1, pp. 21–41, 1986.

[5] B. Guo and I. Babuška, "The h-p version of the finite element method Part 2: General results and applications," *Computational Mechanics*, vol. 1, no. 3, pp. 203–220, 1986.

[6] C. Schwab, *p- and hp-Finite Element Methods: Theory and Applications in Solid and Fluid Mechanics*. Numerical mathematics and scientific computation, Oxford: Oxford University Press, 1998.

[7] L. Demkowicz, *Computing with hp-adaptive finite elements, Vol. 1: One and Two Dimensional Elliptic and Maxwell Problems*. Applied mathematics and nonlinear science series, Boca Raton: Chapman & Hall/CRC, 2007.

[8] P. Šolín, *Higher-order finite element methods*. Studies in advanced mathematics, Boca Raton: Chapman & Hall/CRC, 2004.

[9] S. M. Schnepp and T. Weiland, "Efficient large scale electromagnetic simulations using dynamically adapted meshes with the discontinuous Galerkin method," *Journal of Computational and Applied Mathematics*, vol. 236, no. 18, pp. 4909–4924, 2012.

[10] L. Korous and P. Šolín, "An adaptive hp-DG method with dynamically-changing meshes for non-stationary compressible Euler equations," *Computing*, vol. 95, no. 1, pp. 425–444, 2012.

[11] P. Šolín and L. Korous, "Space-time adaptive hp-FEM for problems with traveling sharp fronts," *Computing*, vol. 95, no. 1, pp. 709–722, 2012.

[12] G. S. Payette, "Spectral/Hp Finite Element Models for Fluids and Structures.," Tech. Rep. SAND2012-7615, Sandia National Laboratories, 2012.

[13] A. Pechstein and J. Schöberl, "Anisotropic mixed finite elements for elasticity," *International Journal for Numerical Methods in Engineering*, vol. 90, no. 2, pp. 196–217, 2012.

[14] C. D. Cantwell, S. Yakovlev, R. M. Kirby, N. S. Peters, and S. J. Sherwin, "High-order spectral/hp element discretisation for reaction–diffusion problems on surfaces: Application to cardiac electrophysiology," *Journal of Computational Physics*, vol. 257, Part A, pp. 813–829, 2014.

[15] C. D. Cantwell, D. Moxey, A. Comerford, A. Bolis, G. Rocco, G. Mengaldo, D. De Grazia, S. Yakovlev, J. E. Lombard, D. Ekelschot, B. Jordi, H. Xu, Y. Mohamied, C. Eskilsson, B. Nelson, P. Vos, C. Biotto, R. M. Kirby, and S. J. Sherwin, "Nektar++: An open-source spectral/hp element framework," *Computer Physics Communications*, vol. 192, pp. 205–219, 2015.

[16] P. Šolín, J. Červený, and I. Doležel, "Arbitrary-level hanging nodes and automatic adaptivity in the hp-FEM," *Mathematics and Computers in Simulation*, vol. 77, no. 1, pp. 117–132, 2008.

[17] P. Šolín, L. Dubcova, and I. Doležel, "Adaptive hp-FEM with arbitrary-level hanging nodes for Maxwell's equations," *Adv. Appl. Math. Mech*, vol. 2, no. 4, pp. 518–532, 2010.

[18] L. Demkowicz, J. T. Oden, W. Rachowicz, and O. Hardy, "Toward a universal h-p adaptive finite element strategy, part 1. Constrained approximation and data structure," *Computer Methods in Applied Mechanics and Engineering*, vol. 77, no. 1–2, pp. 79–112, 1989.

[19] G. Strang, *An analysis of the finite element method.* Englewood Cliffs, N.J: Prentice-Hall, 1973.

[20] T. J. R. Hughes, *The finite element method: linear static and dynamic finite element analysis.* Mineola, NY: Dover Publications, 2000.

[21] K. J. Bathe, *Finite element procedures.* New Jersey: Prentice Hall, 2007.

[22] R. Verfürth, *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques.* Chichester, New York: Teubner Verlag, 1996.

[23] W. F. Mitchell, "A Comparison of Adaptive Refinement Techniques for Elliptic Problems," *ACM Trans. Math. Softw.*, vol. 15, no. 4, pp. 326–347, 1989.

[24] M.-C. Rivara, "Mesh Refinement Processes Based on the Generalized Bisection of Simplices," *SIAM Journal on Numerical Analysis*, vol. 21, no. 3, pp. 604–613, 1984.

[25] M.-C. Rivara, "Design and Data Structure of Fully Adaptive, Multigrid, Finite-element Software," *ACM Trans. Math. Softw.*, vol. 10, no. 3, pp. 242–264, 1984.

[26] M.-C. Rivara, "Algorithms for refining triangular grids suitable for adaptive and multigrid techniques," *International Journal for Numerical Methods in Engineering*, vol. 20, no. 4, pp. 745–756, 1984.

[27] E. G. Sewell, *Automatic Generation of Triangulations for Piecewise Polynomial Approximation.* PhD thesis, Purdue University, 1972.

[28] I. Kossaczký, "A recursive approach to local mesh refinement in two and three dimensions," *Journal of Computational and Applied Mathematics*, vol. 55, no. 3, pp. 275–288, 1994.

[29] R. E. Bank and A. H. Sherman, "A Refinement Algorithm and Dynamic Data Structure for Finite Element Meshes," tech. rep., University of Texas at Austin, Austin, TX, USA, 1980.

[30] R. E. Bank and A. H. Sherman, "An adaptive, multi-level method for elliptic boundary value problems," *Computing*, vol. 26, no. 2, pp. 91–105, 1981.

[31] I. Babuška and A. Aziz, "On the Angle Condition in the Finite Element Method," *SIAM Journal on Numerical Analysis*, vol. 13, no. 2, pp. 214–226, 1976.

[32] R. Schneiders, "Algorithms for quadrilateral and hexahedral mesh generation," *Proceedings of the VKI Lecture Series on Computational Fluid Dynamic, VKI-LS*, vol. 4, 2000.

[33] R. Schneiders and J. Debye, "Refining Quadrilateral and Brick Element Meshes," in *Modeling, Mesh Generation, and Adaptive Numerical Methods for Partial Differential Equations* (I. Babuska, W. D. Henshaw, J. E. Oliger, J. E. Flaherty, J. E. Hopcroft, and T. Tezduyar, eds.), no. 75 in The IMA Volumes in Mathematics and its Applications, pp. 53–65, Springer New York, 1995.

[34] M. Ainsworth and B. Senior, "Aspects of an adaptive hp-finite element method: Adaptive strategy, conforming approximation and efficient solvers," *Computer Methods in Applied Mechanics and Engineering*, vol. 150, no. 1–4, pp. 65–87, 1997.

[35] P. Frauenfelder, *hp-Finite element methods on anisotropically, locally refined meshes in three dimensions with stochastic data.* PhD thesis, Swiss Federal Institute of Technology Zürich, Zürich, 2004.

[36] W. Rachowicz, J. T. Oden, and L. Demkowicz, "Toward a universal h-p adaptive finite element strategy part 3. design of h-p meshes," *Computer Methods in Applied Mechanics and Engineering*, vol. 77, no. 1–2, pp. 181–212, 1989.

[37] L. Demkowicz, K. Gerdes, C. Schwab, A. Bajer, and T. Walsh, "HP90: A general and flexible Fortran 90 hp-FE code," *Computing and Visualization in Science*, vol. 1, no. 3, pp. 145–163, 1998.

[38] L. Demkowicz, A. Bajer, W. Rachowicz, and K. Gerdes, "3D hp-Adaptive Finite Element Package Fortran 90 Implementation (3Dhp90)," TICAM Report 99-29, The University of Texas at Austin, Texas Institute for Computational and Applied Mathematics, 1999.

[39] W. Rachowicz and L. Demkowicz, "An hp-adaptive finite element method for electromagnetics: Part 1: Data structure and constrained approximation," *Computer Methods in Applied Mechanics and Engineering*, vol. 187, no. 1–2, pp. 307–335, 2000.

[40] W. Rachowicz and L. Demkowicz, "An hp-adaptive finite element method for electromagnetics—part II: A 3D implementation," *International journal for numerical methods in engineering*, vol. 53, no. 1, pp. 147–180, 2002.

[41] M. Paszyński and L. Demkowicz, "Parallel, fully automatic hp-adaptive 3D finite element package," *Engineering with Computers*, vol. 22, no. 3-4, pp. 255–276, 2006.

[42] W. Bangerth, R. Hartmann, and G. Kanschat, "deal.II—A general-purpose object-oriented finite element library," *ACM Transactions on Mathematical Software*, vol. 33, no. 4, pp. 1–27, 2007.

[43] A. Schröder, "Constrained Approximation in hp-FEM: Unsymmetric Subdivisions and Multi-Level Hanging Nodes," in *Spectral and High Order Methods for Partial Differential Equations* (J. S. Hesthaven and E. M. Ronquist, eds.), no. 76 in Lecture Notes in Computational Science and Engineering, pp. 317–325, Springer Berlin Heidelberg, 2011.

[44] P. Frauenfelder and C. Lage, "Concepts—An Object-Oriented Software Package for Partial Differential Equations," *ESAIM: Mathematical Modelling and Numerical Analysis*, vol. 36, no. 5, pp. 937–951, 2002.

[45] P. Kus, *Automatic hp-Adaptivity on Meshes with Arbitrary-Level Hanging Nodes in 3D.* PhD thesis, Charles University, Institute of Mathematics, Prague, 2011.

[46] C. D. Mote, "Global-local finite element," *International Journal for Numerical Methods in Engineering*, vol. 3, no. 4, pp. 565–574, 1971.

[47] A. K. Noor, "Global-local methodologies and their application to nonlinear analysis," *Finite Elements in Analysis and Design*, vol. 2, no. 4, pp. 333–346, 1986.

[48] O. C. Zienkiewicz and A. Craig, "Adaptive refinement, error estimates, multigrid solution and hierarchic finite element method concepts," in *Accuracy Estimates and Adaptive Refinements in Finite Element Calculations* (I. Babuska, O. C. Zienkiewicz, J. Gago, and E. R. de Oliviera, eds.), pp. 25–55, New York: Wiley, 1986.

[49] W. F. Mitchell, "Adaptive refinement for arbitrary finite-element spaces with hierarchical bases," *Journal of Computational and Applied Mathematics*, vol. 36, no. 1, pp. 65–78, 1991.

[50] T. Belytschko, J. Fish, and B. E. Engelmann, "A finite element with embedded localization zones," *Computer Methods in Applied Mechanics and Engineering*, vol. 70, no. 1, pp. 59–89, 1988.

[51] J. Fish and T. Belytschko, "Elements with embedded localization zones for large deformation problems," *Computers & Structures*, vol. 30, no. 1–2, pp. 247–256, 1988.

[52] J. Fish and T. Belytschko, "A finite element with a unidirectionally enriched strain field for localization analysis," *Computer Methods in Applied Mechanics and Engineering*, vol. 78, no. 2, pp. 181–200, 1990.

[53] T. Belytschko, J. Fish, and A. Bayliss, "The spectral overlay on finite elements for problems with high gradients," *Computer Methods in Applied Mechanics and Engineering*, vol. 81, no. 1, pp. 71–89, 1990.

[54] E. Rank, "Adaptive remeshing and h-p domain decomposition," *Computer Methods in Applied Mechanics and Engineering*, vol. 101, no. 1–3, pp. 299–313, 1992.

[55] J. Fish, "The s-version of the finite element method," *Computers & Structures*, vol. 43, no. 3, pp. 539–547, 1992.

[56] J. Fish, "Hierarchical modelling of discontinuous fields," *Communications in Applied Numerical Methods*, vol. 8, no. 7, pp. 443–453, 1992.

[57] J. Fish and S. Markolefas, "Adaptive s-method for linear elastostatics," *Computer Methods in Applied Mechanics and Engineering*, vol. 104, no. 3, pp. 363–396, 1993.

[58] J. Fish, S. Markolefas, R. Guttal, and P. Nayak, "On adaptive multilevel superposition of finite element meshes for linear elastostatics," *Applied Numerical Mathematics*, vol. 14, no. 1–3, pp. 135–164, 1994.

[59] P. K. Moore and J. E. Flaherty, "Adaptive local overlapping grid methods for parabolic systems in two space dimensions," *Journal of Computational Physics*, vol. 98, no. 1, pp. 54–63, 1992.

[60] I. Babuška and J. M. Melenk, "The Partition of Unity Method," *International Journal for Numerical Methods in Engineering*, vol. 40, no. 4, pp. 727–758, 1997.

[61] T. Strouboulis, I. Babuška, and K. Copps, "The design and analysis of the Generalized Finite Element Method," *Computer Methods in Applied Mechanics and Engineering*, vol. 181, no. 1–3, pp. 43–69, 2000.

[62] T.-P. Fries and T. Belytschko, "The extended/generalized finite element method: An overview of the method and its applications," *International Journal for Numerical Methods in Engineering*, vol. 84, no. 3, pp. 253–304, 2010.

[63] N. Zander, T. Bog, S. Kollmannsberger, D. Schillinger, and E. Rank, "Multi-level hp-adaptivity: high-order mesh adaptivity without the difficulties of constraining hanging nodes," *Computational Mechanics*, vol. 55, no. 3, pp. 499–517, 2015.

[64] B. A. Szabó and I. Babuska, *Finite element analysis*. New York: John Wiley & Sons, 1991.

[65] B. A. Szabó, A. Düster, and E. Rank, "The p-version of the finite element method," in *Encyclopedia of Computational mechanics* (E. Stein, ed.), Chichester, West Sussex: John Wiley & Sons, 2004.

[66] 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*, vol. 89, no. 9, pp. 1171–1202, 2012.

[67] W. F. Mitchell and M. A. McClain, "A Survey of hp-Adaptive Strategies for Elliptic Partial Differential Equations," in *Recent Advances in Computational and Applied Mathematics* (T. E. Simos, ed.), pp. 227–258, Dordrecht, Netherlands: Springer, 2011.

[68] E. Rank and R. Krause, "A multiscale finite-element method," *Computers & structures*, vol. 64, no. 1, pp. 139–144, 1997.

[69] R. Krause and E. Rank, "Multiscale computations with a combination of the h- and p-versions of the finite-element method," *Computer Methods in Applied Mechanics and Engineering*, vol. 192, no. 35-36, pp. 3959–3983, 2003.

[70] A. Düster, A. Niggl, and E. Rank, "Applying the hp–d version of the FEM to locally enhance dimensionally reduced models," *Computer Methods in Applied Mechanics and Engineering*, vol. 196, no. 37-40, pp. 3524–3533, 2007.

[71] D. Schillinger and E. Rank, "An unfitted hp-adaptive finite element method based on hierarchical B-splines for interface problems of complex geometry," *Computer Methods in Applied Mechanics and Engineering*, vol. 200, no. 47-48, pp. 3358–3380, 2011.

[72] 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*, vol. 249-252, pp. 116–150, 2012.

[73] D. Schillinger, *The p- and B-spline versions of the geometrically nonlinear finite cell method and hierarchical refinement strategies for adaptive isogeometric and embedded domain analysis*. PhD thesis, Technische Universität München, Chair for Computation in Engineering, 2012.

[74] D. Schillinger, J. A. Evans, A. Reali, M. A. Scott, and T. J. R. Hughes, "Isogeometric collocation: Cost comparison with Galerkin methods and extension to adaptive hierarchical NURBS discretizations," *Computer Methods in Applied Mechanics and Engineering*, vol. 267, pp. 170–232, 2013.

[75] L. Beilina, S. Korotov, and M. Krizek, "Nonobtuse Tetrahedral Partitions that Refine Locally Towards Fichera-Like Corners," *Applications of Mathematics*, vol. 50, no. 6, pp. 569–581, 2005.

[76] P. Kus, P. Šolín, and D. Andrs, "Arbitrary-level hanging nodes for adaptive -FEM approximations in 3D," *Journal of Computational and Applied Mathematics*, vol. 270, pp. 121–133, 2014.

[77] D. Schillinger, S. J. Hossain, and T. J. R. Hughes, "Reduced Bézier element quadrature rules for quadratic and cubic splines in isogeometric analysis," *Computer Methods in Applied Mechanics and Engineering*, vol. 277, pp. 1–45, 2014.

[78] I. Babuška and B. Guo, "Approximation properties of the h-p version of the finite element method," *Computer Methods in Applied Mechanics and Engineering*, vol. 133, no. 3-4, pp. 319–346, 1996.

[79] Z. Yosibash, *Singularities in elliptic boundary value problems and elasticity and their connection with failure initiation.* No. 37 in Interdisciplinary applied mathematics, New York, NY: Springer, 2012.

[80] J. R. Shewchuk, "An introduction to the conjugate gradient method without the agonizing pain," tech. rep., Carnegie-Mellon University. Department of Computer Science, 1994.

[81] W. Rachowicz, D. Pardo, and L. Demkowicz, "Fully automatic hp-adaptivity in three dimensions," *Computer Methods in Applied Mechanics and Engineering*, vol. 195, no. 37–40, pp. 4816–4842, 2006.

[82] N. M. Newmark, "A Method of Computation for Structural Dynamics," *Journal of the Engineering Mechanics Division*, vol. 85, no. EM3, pp. 67–94, 1959.

[83] K. Girkmann, *Flächentragwerke.* Wien: Springer, 4 ed., 1956.

[84] E. Rank, A. Düster, V. Nübel, K. Preusch, and O. Bruhns, "High order finite elements for shells," *Computer Methods in Applied Mechanics and Engineering*, vol. 194, no. 21-24, pp. 2494–2512, 2005.

[85] J. Kiendl, Y. Bazilevs, M. C. Hsu, R. Wüchner, and K. U. Bletzinger, "The bending strip method for isogeometric analysis of Kirchhoff–Love shell structures comprised of multiple patches," *Computer Methods in Applied Mechanics and Engineering*, vol. 199, no. 37–40, pp. 2403–2416, 2010.

[86] T. J. R. Hughes, J. A. Cottrell, and Y. Bazilevs, "Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement," *Computer Methods in Applied Mechanics and Engineering*, vol. 194, no. 39–41, pp. 4135–4195, 2005.

[87] J. A. Cottrell, T. J. R. Hughes, and A. Reali, "Studies of refinement and continuity in isogeometric structural analysis," *Computer Methods in Applied Mechanics and Engineering*, vol. 196, no. 41–44, pp. 4160–4183, 2007.

[88] J. Cottrell, T. J. Hughes, and Y. Bazilevs, *Isogeometric Analysis: Toward Integration of CAD and FEA.* New York: Wiley and Sons, 2009.

[89] Y. Bazilevs, V. M. Calo, J. A. Cottrell, J. A. Evans, T. J. R. Hughes, S. Lipton, M. A. Scott, and T. W. Sederberg, "Isogeometric analysis using T-splines," *Computer Methods in Applied Mechanics and Engineering*, vol. 199, no. 5–8, pp. 229–263, 2010.

[90] D. Scholz, S. Kollmannsberger, A. Düster, and E. Rank, "Thin Solids for Fluid-Structure Interaction," in *Fluid-Structure Interaction* (H.-J. Bungartz and M. Schäfer, eds.), no. 53 in Lecture Notes in Computational Science and Engineering, pp. 294–335, Springer Berlin Heidelberg, 2006.

[91] A. Düster, D. Scholz, and E. Rank, "pq-Adaptive solid finite elements for three-dimensional plates and shells," *Computer Methods in Applied Mechanics and Engineering*, vol. 197, no. 1–4, pp. 243–254, 2007.

[92] W. Hager, "Condition Estimates," *SIAM Journal on Scientific and Statistical Computing*, vol. 5, no. 2, pp. 311–316, 1984.

[93] J. M. Melenk, K. Gerdes, and C. Schwab, "Fully discrete hp-finite elements: fast quadrature," *Computer Methods in Applied Mechanics and Engineering*, vol. 190, no. 32–33, pp. 4339–4364, 2001.