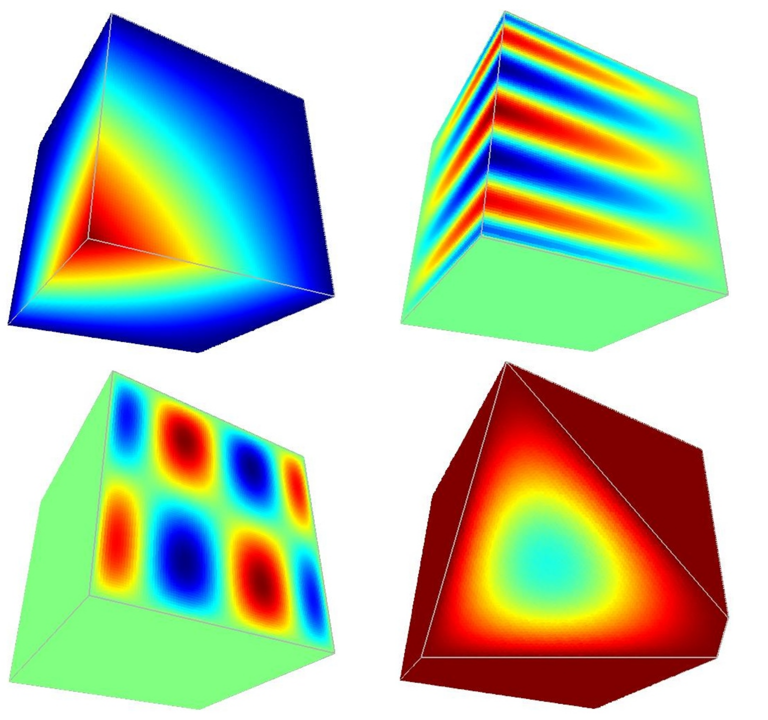
# Adaptive $hp$-FEM for Singular 3D Problems

Pavel Solin[1], Pavel Kus[2], David Andrs[1]
[1]Department of Mathematical Sciences, The University of Texas at El Paso (UTEP)
[2]Academy of Sciences of the Czech Republic, Prague
http://hpfem.math.utep.edu/

## FEM and $hp$-FEM

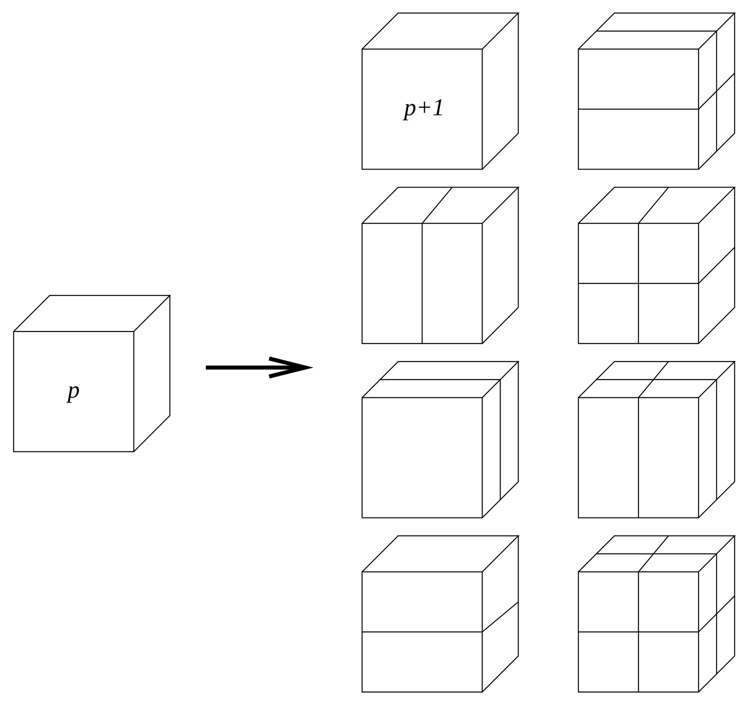
The *Finite element method (FEM)* is the most powerful technique for the numerical solution of partial differential equations (PDEs). Traditional FEM in engineering applications usually employs linear or quadratic elements (this refers to the polynomial degree of approximation in elements). The $hp$-FEM is a modern version of the FEM: It employs elements of variable size $(h)$ and polynomial degree $(p)$ adaptively to maximize the efficiency and accuracy. The efficiency gap between standard FEM and the $hp$-FEM is stunning. Therefore, the $hp$-FEM is becoming increasingly popular despite the fact that it is more difficult to work with than standard FEM. Our group has made significant contributions towards the development of the $hp$-FEM.



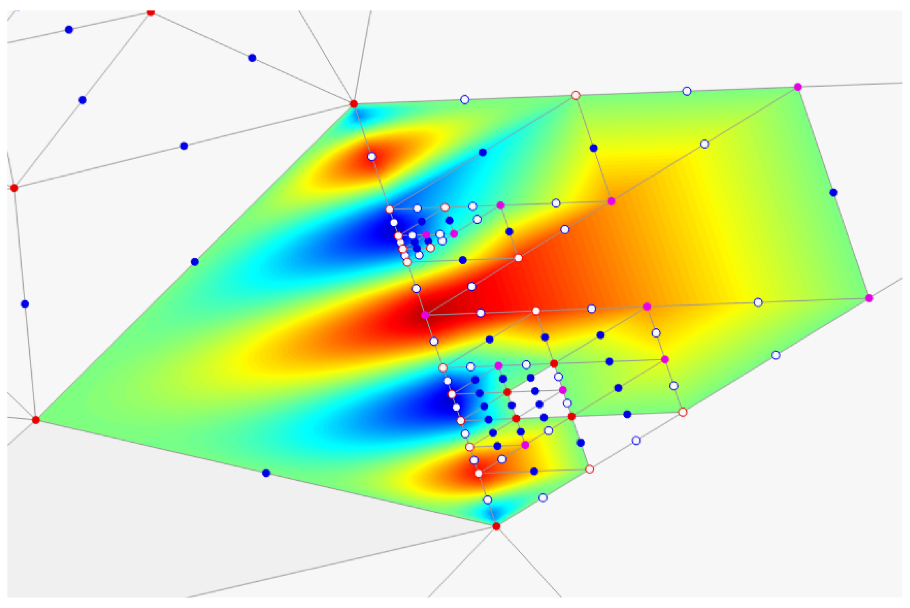
Various types of shape functions employed by the $hp$-FEM: *Vertex function* (top-left), *edge function* (top-right), *face function* (bottom-left), and *bubble function* (bottom-right).

## Automatic Adaptivity

The accuracy of the finite element approximation is not the same all over the computational domain. It is essential to identify the problematic places and refine the finite element mesh there, so that the error is reduced most efficiently. Earlier, the engineer or scientist had to find the problematic spots by himself/herself and regenerate the mesh manually. However, this is difficult or virtually impossible today due to the large complexity of computational models. The computer should identify elements with largest errors by itself and refine the mesh automatically. Notice that the exact solution is not available, and therefore one has to estimate the error using the approximate solution and the underlying PDE only. This is a challenging task which is subject of intense ongoing research.



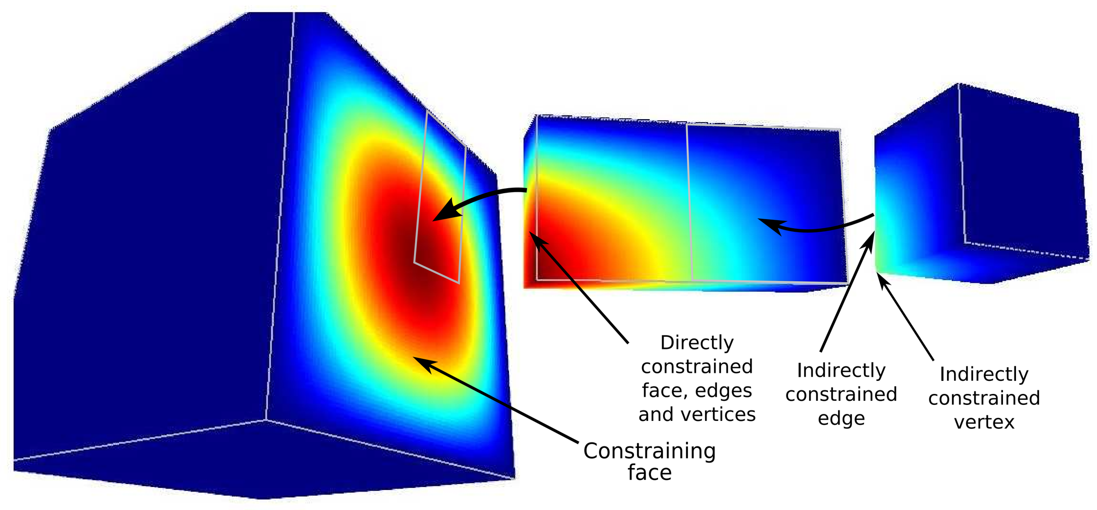

A hexahedral element can be split in multiple ways. The number of these refinement candidates rises even more when variable polynomial degrees in subelements are allowed.



Multiple-level hanging nodes on an edge in 2D.

## Arbitrary-Level Hanging Nodes

When a finite element is refined, some of the faces, edges, and vertices of the subelements do not match the faces, edges, and vertices of adjacent elements. Standard solution to this problem is to refine the adjacent elements as well. However, these *unwanted* or *forced* refinements have severe disadvantages: They do not reduce the error optimally, increase the size of the discrete problem, slow down the convergence, and their recursive nature makes the computer implementation cumbersome. A better solution is to introduce *hanging nodes* – instead of refining adjacent elements, algebraic relations are imposed on the nonmatching faces, edges, and vertices. We designed an original method which does this efficiently, see *P. Solin, J. Cerveny, I. Dolezel: Arbitrary-Level Hanging Nodes and Automatic Adaptivity in the hp-FEM, MATCOM, doi:10.1016/j.matcom.2007.02.011.*



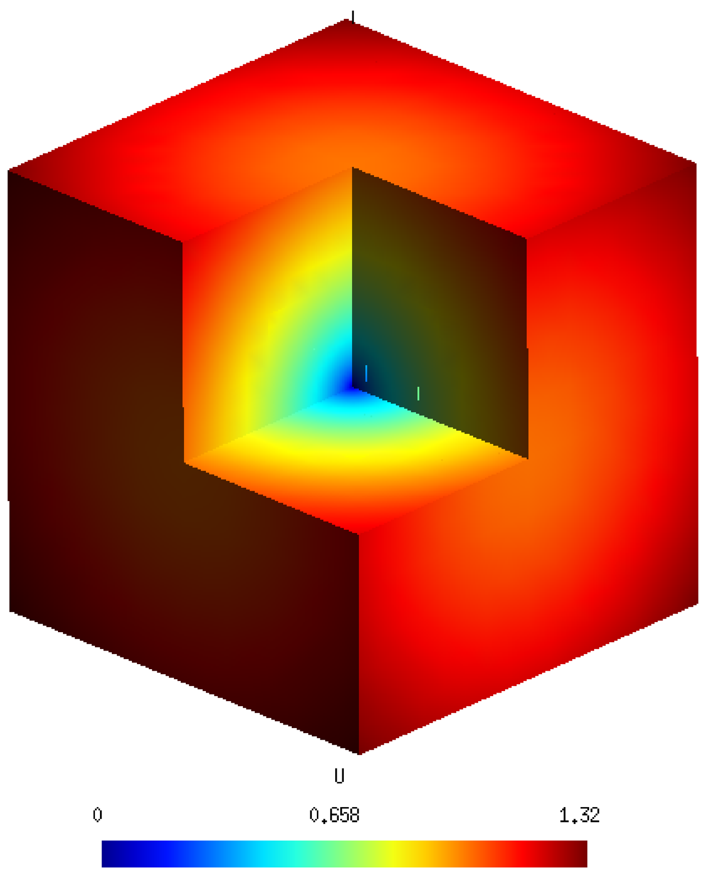

Hanging nodes on a mesh face in 3D.

## Benchmark Problem

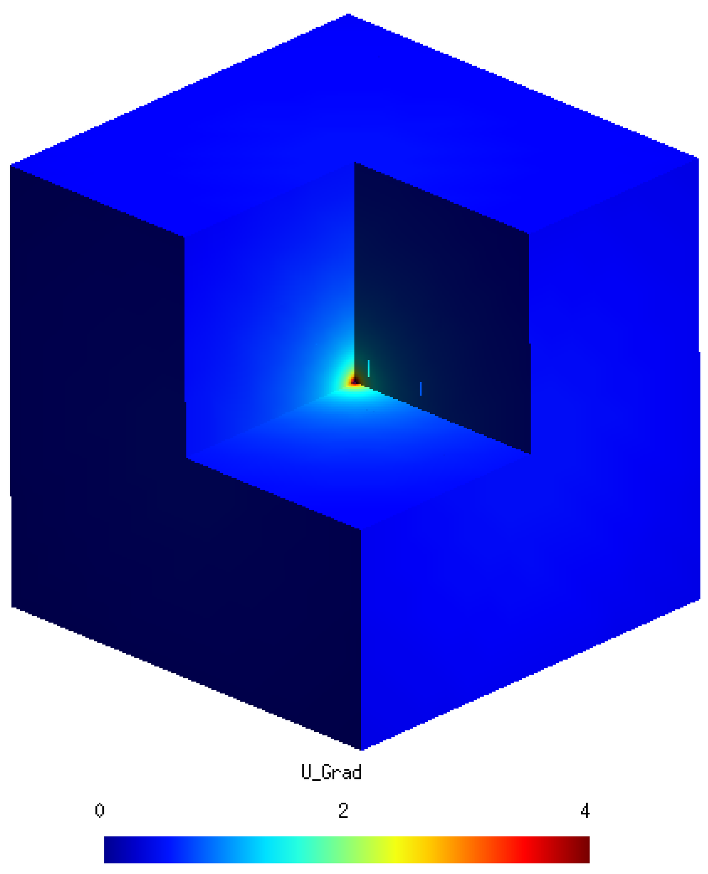
The computational domain is a cube with a missing part – the *Fichera corner*. Solved is the electrostatics equation

$$-\Delta\varphi = f$$

where the boundary conditions are chosen to match an exact solution $\varphi(x_1, x_2, x_3) = (x_1^2 + x_2^2 + x_3^2)^{1/4}$. This solution has singular gradient at the re-entrant corner, and therefore the electrical field strength $\mathbf{E} = -\nabla\varphi$ tends to infinity there. In practical applications, it is extremely important to resolve such singularities accurately. In the figures below, we show the sequence of $hp$-FEM meshes produced by the adaptive algorithm. Polynomial degrees have different colors: dark blue = 1, light blue = 2, green = 3, orange = 4.



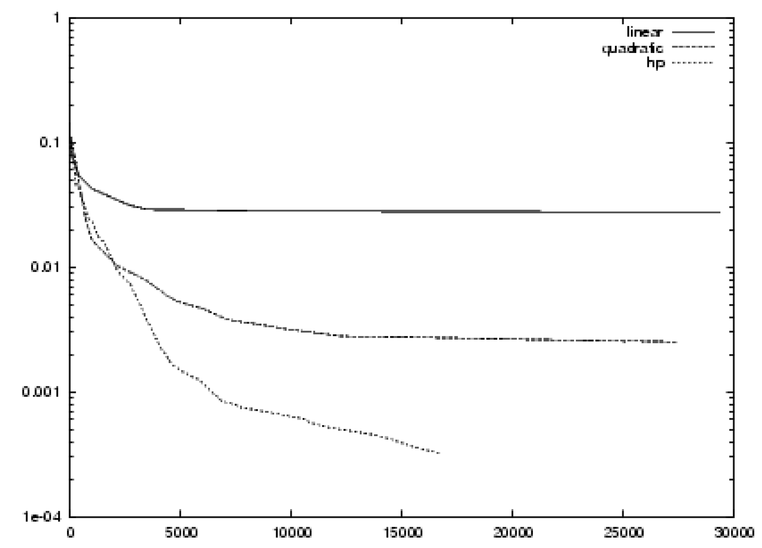

Electric potential $\varphi$.



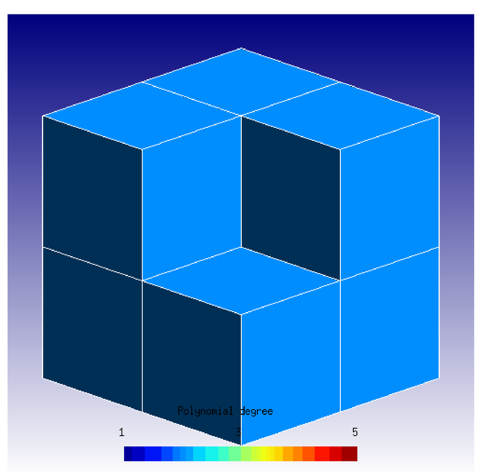

Magnitude of electric field $\mathbf{E} = -\nabla\varphi$.

## Convergence Comparison

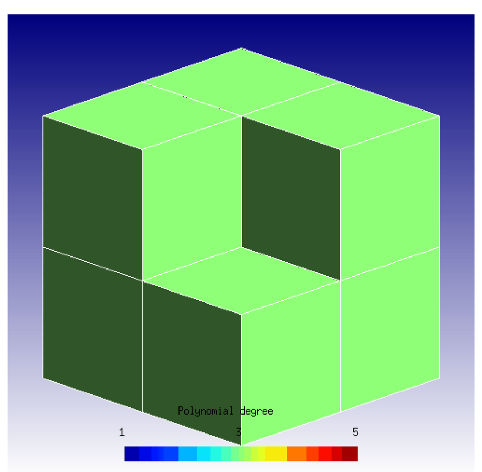
The graphs below show the convergence of adaptive FEM with linear and quadratic elements, and adaptive $hp$-FEM. Linear FEM converges extremely poorly. Quadratic FEM does better, but none of them are close to adaptive $hp$-FEM.



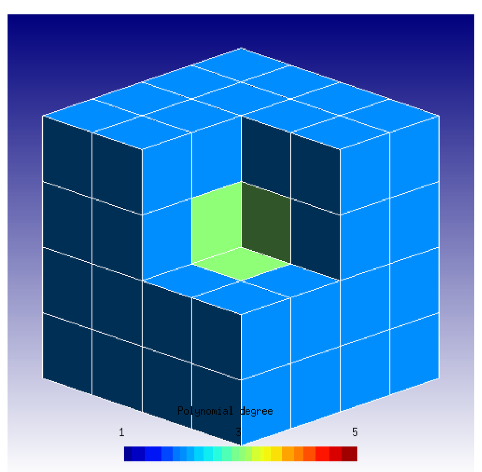

Convergence comparison.



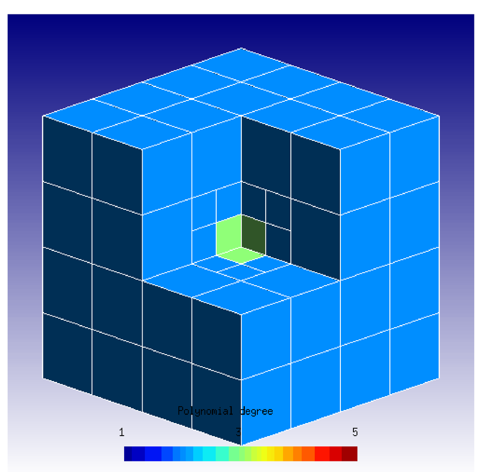
Initial mesh.



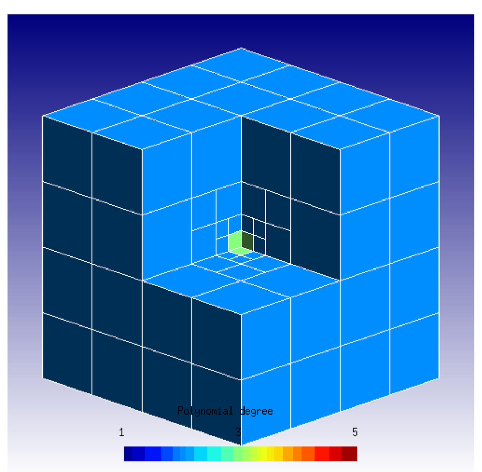
Adaptivity step 2.



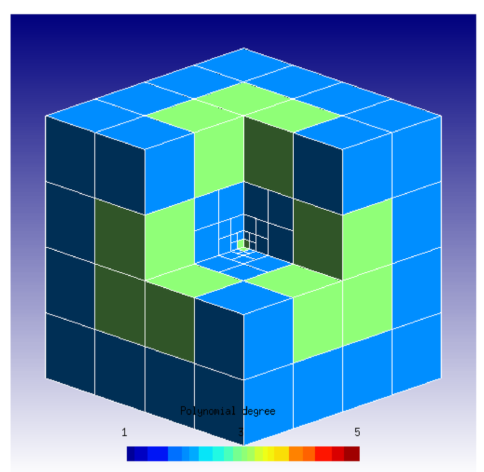
Adaptivity step 4.



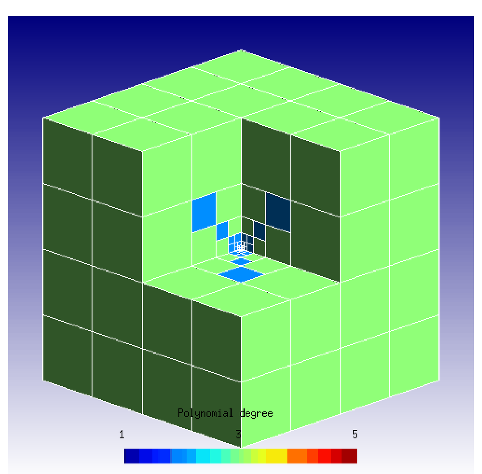
Adaptivity step 6.



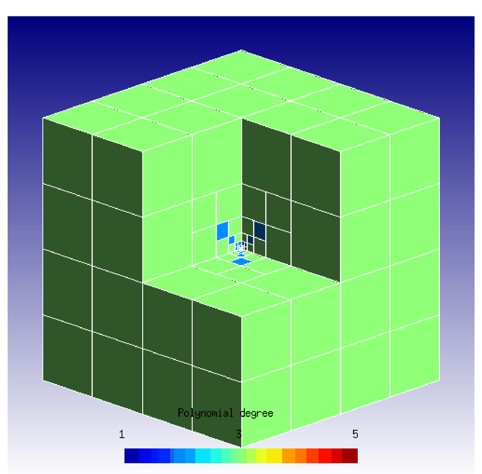
Adaptivity step 8.



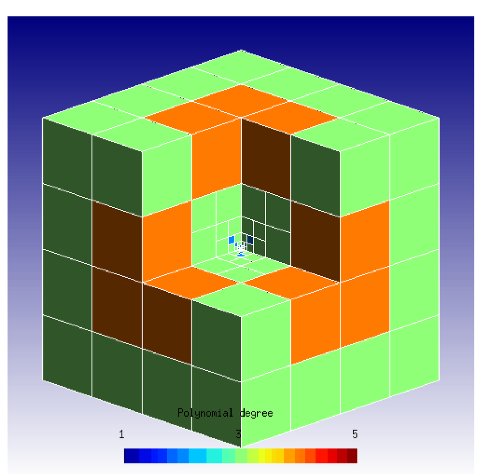
Adaptivity step 10.



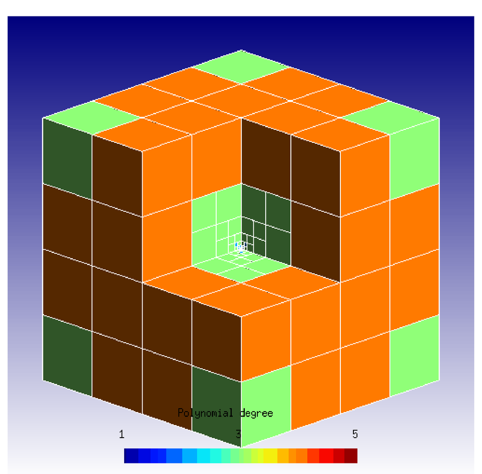
Adaptivity step 12.



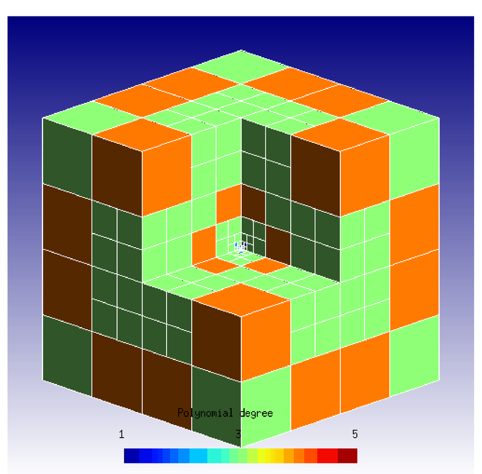
Adaptivity step 14.



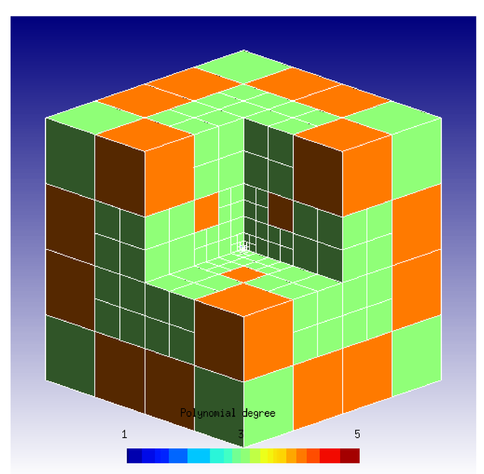
Adaptivity step 16.



Adaptivity step 18.



Adaptivity step 20.



Adaptivity step 22.