Adaptive $hp$-FEM With Arbitrary-Level Hanging Nodes for Time-Harmonic Maxwell’s Equations

Pavel Solin, Lenka Dubcova, Ivo Dolezel
Adaptive $hp$-FEM With Arbitrary-Level Hanging Nodes for Time-Harmonic Maxwell’s Equations

Pavel Solin, Lenka Dubcova, Ivo Dolezel
Pavel Solin, Lenka Dubcova, Ivo Dolezel:
Adaptive hp-FEM With Arbitrary-Level Hanging Nodes
for Time-Harmonic Maxwell’s Equations

Abstract: Great algorithmic difficulty of hp-adaptive algorithms is one of the main obstacles preventing adaptive hp-FEM from being employed widely in realistic engineering computations. In order to reduce their complexity, we present a new technique of arbitrary-level hanging nodes that eliminates forced refinements. By forced refinements we mean refinements which are not based on a large value of an error indicator, but which are needed to keep the mesh sufficiently regular. The algorithmic treatment of forced refinements is highly problematic due to their recursive nature, and obviously, they slow down the performance of automatic adaptivity. We show that in the absence of forced refinements, the complexity of hp-adaptive algorithms drops dramatically while their efficiency improves. This is illustrated on a pair of numerical examples: an inner layer problem with known exact solution, and a problem related to microwave heating.

AMS subject classification: 65N30, 78M10

Keywords: Maxwell’s equations, hp-FEM, arbitrary-order edge elements, automatic adaptivity, constrained approximation, arbitrary-level hanging nodes

Correspondence
solin@utep.edu, ldubcova@miners.utep.edu, dolezel@fel.cvut.cz

Acknowledgment
This work was supported by the Grant Agency of the Czech Republic project 102/07/0496, as well as by the Grant Agency of the Academy of Sciences of the Czech Republic project IAA100760702. The research of L. Dubcova was partly supported by the grant No. 48607 of the Grant Agency of the Charles University in Prague.

The University of Texas at El Paso
Department of Mathematical Sciences
500 West University, El Paso, TX 79968
Email: mathdept@math.utep.edu
URL: http://www.math.utep.edu
Phone: 1.915.747.5761
Fax: 1.915.747.6502
1 Introduction

Nowadays, vector-valued finite elements with continuous tangential components on element interfaces (edge elements) are a standard tool for the solution of Maxwell’s equations in various cavity devices such as waveguides, resonators, microwave ovens, and other models. Edge elements are based on differential forms introduced in late 1950s by H. Whitney [33] (in the context of differential geometry). The first link between the Whitney forms and computational electromagnetics was made in 1984 by P.R. Kotiuga in his thesis [10]. An excellent monograph on this subject is Bossavit’s [4]. The lowest-order edge elements with constant tangential components on element interfaces, called Whitney elements, have been studied by a number of researchers (see, for example, [1, 5, 12, 13, 14, 15, 16, 19]). More recently, the Whitney elements were extended to higher-order edge elements (see [6, 22, 24, 31, 32] and the references therein).

Adaptive higher-order finite element methods (hp-FEM) based on higher-order edge elements [18, 24, 30] belong to the youngest topics in computational electromagnetics. The rapidly growing popularity of these methods is due to their extremely fast, exponential convergence. Mainly for problems containing singularities and internal/boundary layers, the efficiency gap between adaptive hp-FEM and standard adaptive low-order FEM is huge (see, for example, [2, 3, 7, 23, 24, 28, 30]). Despite its advantages, the hp-FEM has not become a widely used computational tool in realistic engineering applications yet. One of the main reasons for this is its high algorithmic complexity and implementation cost.

Therefore, in this paper we present a new automatic hp-adaptivity algorithm that is both substantially simpler and equally or more efficient compared to existing methods such as [7, 18]. The basic idea of the adaptive strategy is that element refinements must not affect adjacent elements in the mesh. Otherwise one has to handle unwanted (forced) refinements that both complicate the algorithm and deteriorate its efficiency. Forced refinements can be encountered in virtually all existing adaptivity algorithms.

The paper is organized as follows: The rest of this introductory section is devoted to the formulation of a model problem. The treatment of hp-FEM approximations with arbitrary-level hanging nodes is discussed in Section 2, automatic hp-adaptivity is described in Section 3, and numerical examples are presented in Section 4. The last Section 5 presents conclusions and an outline of our future work on this subject.

1.1 Time-harmonic Maxwell’s equations

In this section we introduce the time-harmonic Maxwell’s equations for later reference. Consider the problem of solving the normalized time-harmonic Maxwell’s equation [13],

\[ \nabla \times \left( \mu_r^{-1} \nabla \times \mathbf{E} \right) - k^2 \varepsilon_r \mathbf{E} = jk \sqrt{\mu_0} \mathbf{J}_a \]  

(1)
in a bounded domain $\Omega \subset \mathbb{R}^2$ with piecewise-linear boundary. Here $\mu_r = \mu/\mu_0$ is the relative magnetic permeability, $E$ the (complex) phasor of harmonic electric field strength, $k = \omega/c$ the wave number, $j$ the imaginary unit, $\epsilon_r = (\epsilon + j\gamma/\omega)/\epsilon_0$ the (complex) relative electric permittivity, $J_a$ the (complex) phasor of the vector-valued density of conductive currents, $\omega$ the angular frequency and $c$ the speed of light in vacuum. The medium is assumed piecewise-homogeneous for simplicity (i.e., both $\mu_r$ and $\epsilon_r$ are piecewise-constant).

Equation (1) may be equipped with various types of boundary conditions, such as, for example, the perfect conductor boundary condition

$$E \cdot t = 0 \quad \text{on } \partial\Omega,$$

or the impedance condition

$$\nabla \times E - j\lambda E \cdot t = g \cdot t \quad \text{on } \partial\Omega. \quad (3)$$

With (2), the weak formulation of (1) reads: Find $E \in Q$ such that

$$\int_\Omega \mu_r^{-1} (\nabla \times E) \cdot (\nabla \times \overline{F}) \, dx - \int_\Omega k^2 \epsilon_r E \cdot \overline{F} \, dx = \int_\Omega jk\sqrt{\mu_0} J_a \cdot \overline{F} \, dx$$

for all $F \in Q$, where $Q$ is a complex vector space defined as

$$Q = \{ E \in H(\text{curl}, \Omega); \ E \cdot t = 0 \text{ on } \partial\Omega \}.$$  

The symbol $\overline{F}$ stands for the complex-conjugate of $F$. For completeness, let us mention the definitions

$$H(\text{curl}, \Omega) = \{ E \in L^2(\Omega); \ \nabla \times E \in L^2(\Omega) \}$$
and $\nabla \times E = \partial E_2/\partial x_1 - \partial E_1/\partial x_2$.

The domain $\Omega$ is covered with a finite element mesh $\tau_{h,p}$ consisting of non-overlapping convex elements $K_1, K_2, \ldots, K_M$ (in practice triangles or quadrilaterals) equipped with polynomial degrees $1 \leq p_1, p_2, \ldots, p_M$. The finite element subspace of $Q$ has the form

$$Q_{h,p} = \{ E_{h,p} \in Q; \ E_{h,p} \text{ is polynomial of degree } p_i \text{ in } K_i \}. \quad (7)$$

By $N$ we denote the dimension of $Q_{h,p}$ (the number of degrees of freedom in the discrete problem). Recall that functions in the space $Q_{h,p}$ are discontinuous but have continuous tangential components on element interfaces (see, e.g., [4, 13, 20]).

## 2 Arbitrary-level hanging nodes

Adaptive $hp$-FEM with arbitrary-level hanging nodes was first introduced in [21] in the context of continuous higher-order elements for second-order elliptic problems. An extension of this technique to vector-valued $H(\text{curl})$-conforming approximations is nontrivial due to different conformity requirements and a different hierarchic structure of basis functions.
2.1 Arbitrary-level constraints

To begin with, assume a geometrical situation shown in Fig. 1, where a pair of elements $K_2$ and $K_3$ are adjacent to the edge $AB$ of an element $K_1$.

![Figure 1: Example of a mesh with one-level hanging nodes.](image)

For illustration, a quadratic vector-valued edge function associated with the edge $AB$ is shown in Fig. 2.

![Figure 2: Quadratic basis function on the edge $AB$. Left: tangential component, right: normal component.](image)

For the sake of compatibility with the De Rham diagram, all higher-order vector-valued edge functions are defined as gradients of scalar edge functions. This makes them automatically curl-free. The scalar potential of the function from Fig. 2 is shown in Fig. 3.

3
By $E_{K_i}$ let us denote the restriction of the function from Fig. 2 to the element $K_i$, $i = 1, 2, 3$. The constraining function $E_{K_1}$ is a standard edge function. The constrained functions $E_{K_2}$ and $E_{K_3}$ are linear combinations of standard edge functions on the corresponding elements such that

$$E_{K_2} \cdot t_{CB} \equiv E_{K_1} \cdot t_{AB} \quad \text{on the edge } CB$$

and

$$E_{K_3} \cdot t_{AC} \equiv E_{K_1} \cdot t_{AB} \quad \text{on the edge } AC,$$

where $t_{AB}$, $t_{AC}$ and $t_{CB}$ represent unit tangential vectors to the edges $AB$, $AC$ and $CB$, respectively.

In general, let the polynomial degree of the edge $AB$ be some $p_{AB} \geq 0$. Then there are $p_{AB} + 1$ constraining shape functions on $K_1$ associated with the edge $AB$, with polynomial degrees $p = 0, 1, \ldots, p_{AB}$. Interior shape functions (bubble functions) are never constraining nor constrained, and therefore they do not influence the calculation of constraint coefficients. By definition, every constrained edge inherits its orientation and polynomial degree from the constraining one (even if this is in contradiction to the minimum rule [24]). Every edge function on $K_1$ of polynomial degree $0 \leq p \leq p_{AB}$ that is associated with the edge $AB$ constrains $p + 1$ edge functions on the element $K_3$ associated with edge $AC$ and $p + 1$ edge functions on the element $K_2$ associated with edge $CB$.

The case of multiple-level constraints is analogous. Consider, for illustration, a mesh with three-level hanging nodes shown in Fig. 4.

As in the previous case, there are $p_{AB} + 1$ constraining edge functions on $K_1$ associated with the edge $AB$, and an edge function of polynomial degree $p$ constrains $p + 1$ edge func-
Figure 4: Example of a mesh with three-level hanging nodes.

ctions on each of the elements $K_2, K_4, K_5, K_6$ (associated with edges $C_1B, C_2C_1, C_3C_2, AC_3$, respectively). Example of a quadratic basis function associated with the edge $AB$, which consists of a quadratic edge function $E_{K_1}$ on $K_1$ constraining quadratic edge functions on the elements $K_2, K_4, K_5, K_6$, is shown in Fig. 5. 3.

Figure 5: Quadratic basis function on the edge $AB$. Left: tangential component, right: normal component.
Next let us describe in detail how the constraint coefficients are calculated.

### 2.2 Algorithmic treatment

We assume that the reader is familiar with the element-by-element assembling procedure used in higher-order finite element methods [20, 24]. This algorithm requires a unique enumeration of basis functions \( E_1, E_2, \ldots, E_N \) of the finite element space \( Q_{h,p} \) as well as a unique local enumeration of shape functions on the reference domain \( \hat{K} \).

First assume a mesh element \( K_i \) and its unconstrained edge \( e \) of polynomial degree \( p_e \). This element is mapped onto the reference domain \( \hat{K} \) via a reference map \( x_{K_i} : \hat{K} \rightarrow K_i \). Let \( \hat{e} \) be the edge of \( \hat{K} \) such that \( x_{K_i}(\hat{e}) = e \), and by \( \varphi_0^e, \varphi_1^e, \ldots, \varphi_{p_e}^e \) let us denote the edge functions on \( \hat{K} \) associated with the edge \( \hat{e} \). The edge \( e \) is equipped with an edge node

\[
d^e = \{ m_0^e, m_1^e, \ldots, m_{p_e}^e \},
\]

where \( m_j^e \) are indices of the basis functions of the space \( Q_{h,p} \) that are related to the shape functions \( \varphi_0^e, \varphi_1^e, \ldots, \varphi_{p_e}^e \) via the standard transformation relation

\[
\varphi_j^e(\xi) = \left( \frac{Dx_{K_i}}{D\xi} \right)^T \xi E_{m_j^e}(x_{K_i}(\xi)), \quad j = 0, 1, \ldots, p_e.
\]

Next, let \( K_i \) be an element in the mesh whose edge \( e \) is subset of another mesh edge \( AB \) of polynomial degree \( p_e \). In addition to the standard (unconstrained) edge node \( d^e \), we define a constrained edge node

\[
c^e = \{ r, q^e \},
\]

where \( r \) is a reference to the standard node associated with the constraining edge \( AB \) and the index \( q^e \) identifies uniquely the geometrical position of the constrained edge \( e \) within \( AB \). Note that \( AB \) is oriented uniquely through the indices of the vertices \( A \) and \( B \), and \( e \) inherits its orientation. Fig. 6 shows the values of the index \( q^e \) for various geometrical cases.

Finally, assume a basis function \( E_k \) of the space \( Q_{h,p} \), whose tangential component \( E_k \cdot t_{AB} \) on the edge \( AB \) is a polynomial of degree \( p \). Restricted to the edge \( e \subset AB \), the tangential component \( E_k \cdot t_{AB} \) determines the constraint coefficients \( \alpha_0^{e,p}, \alpha_1^{e,p}, \ldots, \alpha_p^{e,p} \) corresponding to the edge functions \( \varphi_0^e, \varphi_1^e, \ldots, \varphi_p^e \) on \( K_i \). The values of these coefficients are obtained by solving a system of \( p + 1 \) linear algebraic equations of the form

\[
\sum_{j=0}^p \alpha_j^{e,p} \varphi_j^e(y_i^p) = \psi_{c,AB}(y_i^p), \quad 0 \leq i \leq p,
\]
where \( y_i^p \in [-1, 1] \) are the \( p + 1 \) Chebyshev points of degree \( p \) on the edge \( e \), and \( \psi_{e,AB} \) is the tangential component \( E_k \cdot t_{AB} \) transformed linearly from \( e \) to \([-1, 1]\). Then,

\[
\varphi^e = \sum_{j=0}^{p} c_{i,j} \phi_j^e
\]

is a new edge function of degree \( p \) on \( \hat{K} \). After transforming \( \varphi^e \) to the element \( K_i \) through (9), its tangential component on the edge \( e \subset AB \) matches exactly the tangential component of \( E_k \cdot t_{AB} \).

The reader does not have to implement these procedures – they are contained in a modular C++ library HERMES which is designed to facilitate the implementation of adaptive \( hp \)-FEM for the solution of nonlinear coupled problems. For more details see the web page http://hpfem.math.utep.edu/hermes.

3 Adaptive \( hp \)-FEM

The major difference between automatic adaptivity in standard \( h \)-FEM and in the \( hp \)-FEM is that in the latter case, a higher-order element can be refined in many different ways. One can either increase its polynomial degree without spatial subdivision (\( p \)-refinement) or the element can be split in space with various distributions of polynomial degrees in the subelements. Fig. 7 illustrates this for a quartic triangular element:

This means that traditional error estimates (that deliver one number per element) do not provide enough information to guide \( hp \)-adaptivity. In order to select an optimal refinement candidate, one needs to use some information about the shape of the error function \( \mathcal{E}_{h,p} = E - E_{h,p} \). In principle, this information could be recovered from suitable estimates of higher derivatives of the solution, but such approach is not very practical and rarely used.
3.1 Error estimation via reference solutions

We prefer to estimate the error by means of the so-called reference solutions [8]. This approach not only provides the desired information about the shape of the error $\mathcal{E}_{h,p}$, but it is very robust in the sense that it can be used for various types of PDEs. A reference solution is an approximation $E_{\text{ref}}$ of the exact solution $E$, that is more accurate than the coarse mesh approximation $E_{h,p}$, and thus the difference $E_{\text{ref}} - E_{h,p}$ provides a meaningful information about the error $\mathcal{E}_{h,p}$. In practice, the reference solution $E_{\text{ref}}$ is sought in an enriched finite element space $Q_{\text{ref}}$ such that we subdivide all elements in the mesh uniformly and increase their polynomial degrees by one, i.e., $Q_{\text{ref}} = Q_{h/2,p+1}$.

3.2 Adaptivity algorithm

With an a-posteriori error estimate of the form

$$\mathcal{E}_{h,p} \approx E_{\text{ref}} - E_{h,p},$$

the outline of our $hp$-adaptivity algorithm is as follows:

1. Assume an initial coarse mesh $\tau_{h,p}$ consisting of (usually) quadratic elements. Besides other technical data, user input includes a prescribed tolerance $TOL > 0$ for the
\( \mathbf{H}(\text{curl}) \) norm of the approximate error function (10) and the number \( D_{\text{DOF}} \) of degrees of freedom to be added in every \( hp \)-adaptivity step.

2. Compute coarse mesh approximation \( \mathbf{E}_{h,p} \in Q_{h,p} \) on \( \tau_{h,p} \).

3. Find reference solution \( \mathbf{E}_{\text{ref}} \in Q_{\text{ref}} \), where \( Q_{\text{ref}} \) is obtained by dividing all elements and increasing the polynomial degrees by one.

4. Construct the approximate error function (10), calculate its \( || \cdot ||_A \) norm \( ERR_i \) on every element \( K_i \) in the mesh, \( i = 1, 2, \ldots, M \). Calculate the global error,

\[
ERR = \sum_{i=1}^{M} ERR_i.
\]

5. If \( ERR \leq TOL \), stop computation and proceed to postprocessing.

6. Sort all elements into a list \( L \) according to their \( ERR_i \) values in decreasing order.

7. While the number of newly added degrees of freedom in this step is less than \( D_{\text{DOF}} \) do:
   (a) Take next element \( K \) from the list \( L \).
   (b) Perform \( hp \)-refinement of \( K \) (to be described in more detail in Paragraph 3.3).

8. Adjust polynomial degrees on unconstrained edges using the so-called minimum rule (every unconstrained edge is assigned the minimum of the polynomial degrees on the pair of adjacent elements).


The following norm is used to guide the adaptive process,

\[
||\mathbf{E}||_A = (\nabla \times \mathbf{E}, \nabla \times \mathbf{E})_\Omega + \kappa^2 (\mathbf{E}, \mathbf{E})_\Omega,
\]

where \( \kappa = \omega \sqrt{\mu_0 \epsilon_0} \) is the wave number in the solved problem. Our experience shows that the adaptive process guided by the standard \( \mathbf{H}(\text{curl}) \) norm does not converge for high frequencies.

Remark 3.1. Note that the refinement of the element \( K \) in step 7(b) cannot cause any recursive refinements in the mesh due to the technique of arbitrary-level hanging nodes.
3.3 Selection of optimal $hp$-refinement

Let $K \in \tau_{h,p}$ be an element of polynomial degree $p_K$ marked for refinement. Without loss of generality, assume that $K$ is a triangle, the procedure for refinement of quadrilateral elements is analogous. We consider the following $N_{ref} = k + (k+1)^4$ refinement options, where $k \geq 0$ is a user input parameter:

1. Increase the polynomial degree of $K$ by $1, 2, \ldots, k$ without spatial subdivision. This yields $k$ refinement candidates.

2. Split $K$ into four similar triangles $K_1, K_2, K_3, K_4$ (as illustrated in Fig. 7). Define $p_0$ to be the integer part of $p_K/2$. For each $K_i$, $1 \leq i \leq 4$ consider $k + 1$ polynomial degrees $p_0 \leq p_i \leq p_0 + k$. This yields additional $(k+1)^4$ refinement candidates. In this case, edges lying on the boundary of $K$ inherit the polynomial degree $p_j$ of the adjacent interior element $K_j$. Polynomial degrees on interior edges are determined using the minimum rule.

For each of these $N_{ref}$ options, we perform a standard $H$(curl)-projection of the reference solution $E_{ref}$ onto the corresponding vector-valued piecewise-polynomial space on the refinement candidate. The candidate with minimum projection error relative to the number of added degrees of freedom is selected.

Remark 3.2. Note that if the technique of arbitrary-level hanging nodes is not in effect, various $hp$-refinements of an element result into different amounts of forced refinements. In particular, $hp$-refinement candidates involving spatial subdivision of $K$ are likely to be more costly than $p$-refinement candidates. It is extremely tedious to take forced refinements into account for the decision about element $hp$-refinement, and usually this is omitted. Then, however, the adaptive algorithm can make wrong decisions.

4 Numerical examples

Let us illustrate the automatic $hp$-adaptivity with arbitrary-level hanging nodes on two numerical examples – an inner layer problem with known exact solution in Paragraph 4.1 and a simplified microwave heating problem in Paragraph 4.2.

4.1 Inner layer problem

We consider a unit square domain $\Omega = (0, 1)^2$ and solve the time-harmonic Maxwell’s equations

$$\nabla \times (\mu_r^{-1} \nabla \times E) - \kappa^2 \epsilon_r E = F \quad \text{in } \Omega,$$

$$\nabla \times (\mu_r^{-1} \nabla \times E) - \kappa^2 \epsilon_r E = F \quad \text{in } \Omega,$$  \hfill (11)
equipped with the impedance boundary condition (3)

\[ \mu_r^{-1} \nabla \times E - j\lambda E \cdot t = g \cdot t \quad \text{on } \partial \Omega, \]

where both the right-hand sides \( F \) and \( g \) are determined by the vector-valued exact solution

\[ E(x_1, x_2) = \left( a(r) \frac{0.25 + x_2}{r}, a(r) \frac{1.25 - x_1}{r} \right). \] (12)

Here, \( r = \sqrt{(1.25 + x_1)^2 + (0.25 - x_2)^2} \) and \( a(r) = \tan(60(r - \pi/3)) + \pi \). We use the parameters \( \mu_r = 1, \epsilon_r = 1, \kappa = 1 \) and \( \lambda = 1 \). The exact solution is shown in Fig. 8.

![Figure 8: Exact solution to the inner layer problem (magnitude and direction of \( E \)).](image)

The \( hp \)-adaptive process starts from a uniform coarse mesh consisting of 4 quadratic elements (2 × 2 division of \( \Omega \)). The mesh after 18 \( hp \)-adaptivity steps, shown in Fig. 9, contains third-level hanging nodes. The numbers inside elements stand for the corresponding polynomial degree. If two numbers are present, then different directional polynomial degrees are used.

The problem was also solved using standard \( h \)-adaptivity with quadratic and cubic elements. The mesh for the quadratic case after 11 refinement steps is shown in Fig. 10.

The efficiency of all three algorithms is compared in Fig. 11. In all cases, the error was measured in the standard \( H(\nabla \times) \)-norm with respect to the exact solution (12). Notice the exponential convergence of the \( hp \)-adaptive algorithm (the convergence curve is roughly a straight line in a decimal-logarithmic scale).
Figure 9: Mesh after 18 $hp$-adaptivity steps, 7171 DOF, relative error 0.09%.

Figure 10: Mesh after 11 $h$-adaptivity steps with quadratic elements, 12720 DOF, relative error 0.30%.

Last we would like to illustrate our experience with using quadrilateral vs. triangular meshes. It seems, for a reason yet unknown, that the convergence on triangular meshes always is worse compared to the quadrilateral ones, as shown in Fig. 13. The $hp$-adaptive
Figure 11: Convergence comparison of $hp$-adaptivity and standard $h$-adaptivity.

process starts from a uniform coarse mesh consisting of 4 quadratic quadrilateral or triangular elements, as shown in Fig. 12.

Figure 12: Initial quadrilateral and triangular meshes.

4.2 Microwave Oven Problem: $TE_{10}$ Mode

This is a simplified model problem of microwave heating, taking place in a square waveguide $\Omega = (-0.125, 0.125)^2$ filled with air, containing a spherical load of radius $r = 0.015625$ m and relative permittivity $\epsilon_r = 5.5$ (permittivity of porcelain). The situation is depicted in Fig. 14.
We solve the normalized time-harmonic Maxwell’s equations
\[ \nabla \times (\mu_r^{-1} \nabla \times E) - \kappa^2 \varepsilon_r E = F, \]
where \( \mu_r = \mu / \mu_0, \kappa = \omega / c, \) and \( \varepsilon_r = \varepsilon / \varepsilon_0 + j \gamma \omega / (\omega \varepsilon_0). \) By \( L = c / f \) we denote the wavelength. The frequency is chosen to be \( f = 1.799 \) GHz, therefore the wavelength \( L = 1/6 \) m and the domain contains one and half of the wave.
In the waveguide, a horizontal wave is generated by time-harmonic current along the edge \( DA \), using the Neumann boundary condition

\[
\mathbf{n} \times (\mu_r^{-1}\nabla \times \mathbf{E}) = -j\omega \mathbf{J}_a.
\]

We use time-harmonic exciting current \( \mathbf{J}_a = 10^{-7} \) A. The rest of the boundary is equipped with perfect conductor boundary conditions \( \mathbf{E} \cdot \mathbf{t} = 0 \). The problem was solved three-times: using adaptive \( hp \)-FEM with arbitrary-level hanging nodes, adaptive \( hp \)-FEM with one-level hanging nodes, and adaptive \( hp \)-FEM with regular meshes. In Figs. 15 – 18 we show the approximate solution corresponding to a relative error of 0.13% and finite element meshes for the three cases.

Figure 15: Approximate solution \( \mathbf{E} \) (relative error 0.1% in the \( H(\text{curl}) \)-norm).
Figure 16: Mesh with arbitrary-level hanging nodes (error 0.1%, 4335 DOF).

Figure 17: Mesh with one-level hanging nodes (error 0.1%, 8438 DOF).
Figure 18: Regular mesh (error 1.27%, 8752 DOF).
Convergence history of the adaptive $hp$-FEM in all three cases is shown in Fig. 19.

Figure 19: Convergence of adaptive $hp$-FEM with arbitrary-level hanging nodes, one-level hanging nodes, and regular meshes.

Finally, Fig. 20 shows again that the convergence on quadrilateral meshes is significantly faster compared to triangular ones. The computations started from the four-element meshes shown in Fig. 12.

Figure 20: Convergence comparison of $hp$-adaptivity on quadrilateral and triangular meshes.
5 Conclusion and Outlook

We presented a new methodology for adaptive $hp$-FEM in the space $\mathbf{H}(\text{curl})$ which is based on arbitrary-level hanging nodes. The main advantage of this technique is that element refinements become completely local. This is important since the refinement of an element never forces neighboring elements to be refined – an unpleasant aspect of adaptivity on regular meshes which is not eliminated even when one-level or two-level hanging nodes are used. Forced refinements have a recursive character, they cannot be taken into account easily in the process of selecting the optimal $hp$-refinement of an element, and their algorithmic treatment is cumbersome. On the other hand, their absence improves the performance of automatic adaptivity and simplifies its algorithmic treatment substantially.

We presented several numerical examples where the technique was illustrated. These examples correspond well to our experience that, typically, one-level hanging nodes are enough for problems with singularities, while multiple-level hanging nodes are needed to approximate efficiently boundary or internal layers.

The effect shown in Figs. 13 and 20 do not seem to be widely known in the computational electromagnetics community yet – for some reason, triangular elements seem to deliver slower convergence compared to quadrilateral ones. This might be related to the fact that triangular meshes typically contain more edges than quadrilateral ones (on the same level of accuracy), but more research needs to be done before this hypothesis can be verified or disproved. In any case, this observation further strengthens the need for hanging nodes in adaptive algorithms.

The development of arbitrary-level hanging nodes for higher-order edge elements of electromagnetics was another step toward our major goal – adaptive multi-mesh $hp$-FEM for coupled problems involving electromagnetic fields, such as microwave heating. The basic idea of adaptive multi-mesh $hp$-FEM is that the physics of coupled problems is not split (like in the case of various operator-splitting methods), but the interpolation error in every physical field or solution component is minimized very efficiently, using an individual mesh equipped with an independent adaptivity algorithm. The arbitrary-level hanging nodes are an essential ingredient for the multi-mesh $hp$-FEM since otherwise, forced refinements occurring on different meshes in the system may collide in inadmissible ways. Recently, we presented a first version of adaptive multi-mesh $hp$-FEM for the coupled problem of linear thermoelasticity [26].

The simultaneous treatment of the electric field and temperature in the sense of the multi-mesh $hp$-FEM requires an ability to deal with standard continuous elements on one mesh and with higher-order edge elements on the other one, with arbitrary-level hanging nodes in both cases. We are now ready to proceed to this coupled problem and hope to report on our progress soon.
References


