Improving accuracy of high-order discontinuous Galerkin method for time-domain electromagnetics on curvilinear domains

H. Fahs*

INRIA, Nachos team, 2004 Route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex, France

The paper discusses high-order geometrical mapping for handling curvilinear geometries in high accuracy discontinuous Galerkin simulations for time-domain Maxwell problems. The proposed geometrical mapping is based on a quadratic representation of the curved boundary and on the adaptation of the nodal points inside each curved element. With high-order mapping, numerical fluxes along curved boundaries are computed much more accurately due to the accurate representation of the computational domain. Numerical experiments for 2D and 3D propagation problems demonstrate the applicability and benefits of the proposed high-order geometrical mapping for simulations involving curved domains.

1. Introduction

Discontinuous Galerkin (DG) methods are a class of finite element methods (FE) based on completely discontinuous piecewise polynomial spaces for the numerical solution and the test functions. For the same order of accuracy, DG methods require more degrees of freedom than continuous FE methods. To obtain highly accurate and stable DG methods, suitable numerical fluxes need to be designed over elemental interfaces. The construction of such numerical fluxes can be done in many different ways, related closely to the particular equation at hand, and is particularly powerful when one considers nonlinear conservation laws [7, 18, 23]. The discontinuous Galerkin method has become very popular in recent years for solving electromagnetic wave propagation problems [18, 23]. It has several distinct advantages. We refer to the lecture notes [9] and the textbook [24] for details and history of the DG method. In particular, the DG method can easily accommodate a non-conforming locally refined mesh (*h*-refinement) as well as a local definition of the polynomial order (*p*-refinement), or both of them in the context of a *hp*-refinement solution strategy [16]. The DG method is flexible with regards to the choice of the time stepping scheme. One may combine the DG spatial discretization with any global [7, 10, 18] or local [31, 34] explicit time integration scheme, or implicit scheme [6], or even a blending between these two schemes [15, 25] provided that the resulting scheme will be stable.

---

*Email: hassan.fahs@gmail.com

Several authors have investigated the successful application of the DG method to time-domain Maxwell's equations. In particular, the behaviour of the DG method with respect to dissipation and dispersion has been well studied by means of theory and numerical experiments [35]. The influence of h-, p- and hp-refinements on the accuracy of the DG method [7, 16, 23], as well as the discretization error on different kinds of grids [17, 18, 25, 32], were investigated. High-order DG methods in space [10] and in space-time [17] have also been developed. However, all these works do not include the study of the error that is due to the discretization of the geometry (the so-called geometrical error). Actually, the previous works in DG method consists in mapping (under a linear bijective transformation) all elements in the physical domain onto a single reference element for which the local DG matrices are precomputed and stored once and for all. This technique is subject to certain constraints on the geometry of physical elements, e.g., straight-sides for triangles or planar faces for tetrahedra.

When designing a high-order discretization method, the pertinence of an accurate representation of the domain and its boundary has been pointed out by several authors, see [2, 27, 29] among others. In some applications, such as compressible flow and elasticity problems, if a DG method is adopted, an important loss of accuracy is observed when a linear approximation of the boundary is used, see [2, 29]. Bassi and Rebay [2] showed that, in the presence of curved boundaries, a meaningful high-order accurate solution can only be obtained if the corresponding high-order approximation of the geometry is employed (i.e. isoparametric FEs). In fact, it is necessary to take into account the boundary curvature effect in order to have a consistent boundary discretization. Krivodonova and Berger [27] introduced a simple way to approximate curvilinear boundaries in the context of DG methods. Their approach consists in building a continuous representation of the normals, and to use those normals in the Riemann solver computation, while keeping the discontinuous representation in the integration on the edges. This method is efficient and accurate enough in cases when the normals to the physical boundary can be computed explicitly and when the mesh is sufficiently refined. However, oscillations may rapidly appear on very irregular boundaries and coarse meshes. Similar conclusions are derived in [29] for linear elasticity problems: sizable errors are present in the numerical solution when the order of the geometric approximation is lower than the order of functional interpolation, even for geometries as simple as a sphere.

Curved elements have been widely used in the context of FE methods. The technique presented by Zlámal [42, 43] is recognized to be the first FE method considering an exact boundary representation. Triangular elements with one curved edge were introduced, and the isoparametric mapping was modified to map a reference element into the triangular element with an exact boundary description. A similar approach was developed by Scott [36], also using triangular elements with one curved side corresponding to the exact boundary. Alternatives to the standard polynomial approximation of the solution were also proposed within the context of curved FEs with an exact boundary representation, see for instance the rational basis by Wachspress [40]. Nevertheless, all these FE techniques with exact boundary representation were not a practical tool, but a mathematical idealization, due to the impossibility to extend the ideas to 3D domains. *Transfinite elements* by Gordon and Hall [21, 22] represented an inflection point in the development of general procedures to exactly treat curved boundaries. The key idea was to introduce blending functions to define a mapping between a reference square and a subdomain with the boundary given by four parametric curves. The generalization of this technique to simplicial elements can be found in Perronnet [33] and Gatto

and Demkowicz [20]. However, the Gordon-Hall type mappings suffer from two drawbacks which are encountered in 3D problems. First, these mappings are not always bijectives and may lead to singular Jacobian matrices [26, 33, 37]. Second, the expressions of these mappings are very complicated in 3D and may contain non-polynomial functions. Therefore, high-order integration schemes must be used to compute volume and boundary integrals. Nevertheless, high-order quadrature rules for tetrahedra are still suboptimal and computationally expensive, making the assembly a lengthy procedure.

In the present paper, a high-order geometrical mapping combined with high-order discontinuous Galerkin time-domain (DGTD) method is applied to the solution of the Maxwell equations on curvilinear domains. This DGTD method is formulated on unstructured, straight-sided simplicial meshes (triangles in 2D and tetrahedra in 3D) . Within each mesh element, the electromagnetic field components are approximated by an arbitrarily high-order nodal polynomial while time integration is achieved by a fourth-order leap-frog scheme. For elements not intersecting the curved boundary or any curved surface inside the domain, standard interpolation and numerical integration are used. But curved elements are treated through a high-order geometrical mapping which consists of three ingredients: (a) a quadratic representation of the curved boundary, (b) a geometric adaptation of the nodal points inside curved elements, and (c) a proper numerical integration scheme to evaluate the local DG matrices. The present work is an extension to the three-dimensional case of the ones presented in [16, 17] and it is also in some sense complementary to the previous ones in that we consider high-order geometrical techniques for domains with curved boundaries. In particular, we show numerically that the DG method is inaccurate for curved domains, and that a higher-order boundary representation introduces a dramatic improvement in the accuracy of the numerical approximation. The proposed mapping does not require a complex reconstruction and is relatively easy to implement within a three-dimensional DG solver. Moreover, it keeps the degree of the determinant of the Jacobian to a minimum. This reduces the computational cost resulting from the evaluation of the DG matrices. Additionally, the distribution of the nodal points inside each curved elements leads to optimal convergence rates of the DG method. Finally, the proposed DG method is slightly more expensive than those based on affine mapping. However, this extra cost is offset by an important saving in the number of degrees of freedom.

This paper is organized as follows. Section 2 recalls the time-domain Maxwell system and its DG discretization. In this section we also prove the stability of the proposed DG method in the case of conducting material. The high-order geometrical mapping is described in Section 3, with special attention to the interpolation and the numerical integration in elements with curved faces. Section 4 presents numerical examples on curvilinear domains for 2D and 3D propagation problems in homogeneous and heterogeneous media. Finally, Section 5 contains a few concluding remarks and ideas for future works.

2. Discontinuous Galerkin method

We consider the time-domain Maxwell equations in three space dimensions for heterogeneous isotropic linear media. The electric field $\boldsymbol{E}(\boldsymbol{x}, t) = {}^t(E_x, E_y, E_z)$ and the magnetic field $\boldsymbol{H}(\boldsymbol{x}, t) = {}^t(H_x, H_y, H_z)$ satisfy Maxwell's equations

$$\varepsilon \partial_t \boldsymbol{E} = -\sigma \boldsymbol{E} + \nabla \times \boldsymbol{H}, \quad \mu \partial_t \boldsymbol{H} = -\nabla \times \boldsymbol{E}, \tag{1}$$

where $\varepsilon(\boldsymbol{x})$, $\mu(\boldsymbol{x})$ and $\sigma(\boldsymbol{x})$ are respectively the permittivity, the permeability and the conductivity of the medium. These equations are posed on a bounded domain Ω of $\mathbb{R}^3$. Our goal is to solve system (1) in a curvilinear domain Ω with boundary $\partial\Omega = \Gamma_a \cup \Gamma_m$, where we impose the following boundary conditions

$$\boldsymbol{n} \times \boldsymbol{E} = 0 \;\; \text{on } \Gamma_m, \quad \mathcal{L}(\boldsymbol{E}, \boldsymbol{H}) = \mathcal{L}(\boldsymbol{E}_{\text{inc}}, \boldsymbol{H}_{\text{inc}}) \;\; \text{on } \Gamma_a,$$

where Γ_m (resp. Γ_a) is the metallic (resp. absorbing) boundary and $\mathcal{L}(\boldsymbol{E}, \boldsymbol{H}) = \boldsymbol{n} \times \boldsymbol{E} + c\mu\boldsymbol{n} \times (\boldsymbol{n} \times \boldsymbol{H})$. Here $\boldsymbol{n}$ denotes the unit outward normal to $\partial\Omega$, $c = 1/\sqrt{\varepsilon\mu}$ is the speed of propagation and $(\boldsymbol{E}_{\text{inc}}, \boldsymbol{H}_{\text{inc}})$ is a given incident field.

2.1 *Space discretization*

We consider a partition $\mathcal{T}_h$ of $\Omega \subset \mathbb{R}^d$ ($d = 2, 3$ is the spatial dimension) into a set of d-simplices τ_i of size h_i with boundaries $\partial\tau_i$ such that $h = \max_{\tau_i \in \mathcal{T}_h} h_i$. Hence, for two-dimensional problems we shall use triangular elements (2-simplices) while tetrahedra (3-simplices) are employed to fill the computational volume. Within this construction we admit meshes with possibly hanging nodes, i.e., by allowing non-conforming (or irregular) meshes where element vertices can lie in the interior of faces of other elements [16]. For each element τ_i, the parameters ε_i, μ_i and σ_i denotes respectively the local permittivity, permeability and conductivity of the medium, which are assumed constant inside the element τ_i. For two distinct elements τ_i and τ_k in $\mathcal{T}_h$, the intersection $\tau_i \cap \tau_k$ is a convex polyhedron a_{ik} which we will call interface, with unitary normal vector $\boldsymbol{n}_{ik}$, oriented from τ_i towards τ_k. For the boundary interfaces, the index k corresponds to a fictitious element outside the domain. We denote by $\mathcal{V}_i$ the set of indices of the elements which are neighbors of τ_i (having an interface in common). Let τ_r be a fixed master d-simplex element; we assume that each $\tau_i \in \mathcal{T}_h$ is the image, under a bijective mapping Ψ_{τ_i}, of the master element τ_r, that is $\tau_i = \Psi_{\tau_i}(\tau_r)$, $\tau_i \in \mathcal{T}_h$, cf. Section 3. Then, to each $\tau_i \in \mathcal{T}_h$, we assign a non-negative integer p_i that is the local interpolation degree and we collect the p_i and Ψ_{τ_i} in the vectors $\boldsymbol{p} = \{p_i : \tau_i \in \mathcal{T}_h\}$ and $\boldsymbol{\Psi} = \{\Psi_{\tau_i} : \tau_i \in \mathcal{T}_h\}$. In the following, for a given partition $\mathcal{T}_h$ and vectors $\boldsymbol{p}$ and $\boldsymbol{\Psi}$, we seek approximate solutions to Eq. (1) in the finite element space

$$V_{\boldsymbol{p}}(\Omega, \mathcal{T}_h, \boldsymbol{\Psi}) = \{\boldsymbol{u} \in L^2(\Omega)^d : u_{k|\tau_i} \circ \Psi_{\tau_i} \in \mathbb{P}_{p_i}^d(\tau_r), \; k = 1, \dots, d, \; \forall\tau_i \in \mathcal{T}_h\},$$

where $L^2(\Omega)$ is the space of square integrable functions on Ω and $\mathbb{P}_{p_i}^d(\tau_r)$ denotes the space of d-dimensional nodal polynomial functions of degree at most p_i inside the element τ_r. For the numerical experiments described in Section 4, we will only use polynomials up to degree $p_i = 4$. Note that the local polynomial degree p_i as well as the local mapping Ψ_{τ_i} may vary from element to element in the mesh.

Following the discontinuous Galerkin approach, the electric and magnetic fields are approximated inside each finite element τ_i by a linear combination of basis functions $\varphi_{ij}(\boldsymbol{x})$ of degree p_i with support τ_i and with time-dependent coefficient functions $E_{ij}(t)$ and $H_{ij}(t)$ as follows

$$\boldsymbol{E}_i = \sum_{1 \le j \le N_i} E_{ij}(t)\varphi_{ij}(\boldsymbol{x}), \quad \boldsymbol{H}_i = \sum_{1 \le j \le N_i} H_{ij}(t)\varphi_{ij}(\boldsymbol{x}). \tag{2}$$

Here, the index j indicates the j-th basis function and $N_i = N(p_i) = (p_i + d)!/p_i!d!$ denotes the local number of degrees of freedom inside τ_i. We now denote by $\bar{\mathbf{E}}_i$ and $\bar{\mathbf{H}}_i$ respectively the column vectors $(E_{ij})_{1 \le j \le N_i}$ and $(H_{ij})_{1 \le j \le N_i}$. As usual for DG schemes, the Maxwell system (1) is multiplied by a test function $\varphi \in \mathsf{Span}(\varphi_{ij}, 1 \le$

$j \leq N_i$) and integrated over each single element τ_i. After integration by parts, inserting the DG approximation, Eq. (2), and after applying a centered numerical flux in the boundary integrals, the semi-discrete formulation of the scheme (see [17] for more details) in the physical element τ_i reads as

$$
\begin{aligned}
\mathrm{M}_i^\varepsilon \partial_t \bar{\mathbf{E}}_i &= -\mathrm{M}_i^\sigma \bar{\mathbf{E}}_i + \mathrm{K}_i \bar{\mathbf{H}}_i - \sum_{k \in \mathcal{V}_i} \mathrm{S}_{ik} \bar{\mathbf{H}}_k, \\
\mathrm{M}_i^\mu \partial_t \bar{\mathbf{H}}_i &= \qquad\qquad -\mathrm{K}_i \bar{\mathbf{E}}_i + \sum_{k \in \mathcal{V}_i} \mathrm{S}_{ik} \bar{\mathbf{E}}_k,
\end{aligned}
\tag{3}
$$

where the mass matrices M_i^γ (γ stands for ε or, μ or, σ), the stiffness matrix K_i (all of size $N_i \times N_i$) and the $N_i \times N_k$ interface (or flux) matrix S_{ik} are given by

$$
\begin{aligned}
(\mathrm{M}_i^\gamma)_{jl} &= \gamma_i \int_{\tau_i} \varphi_{ij} \cdot \varphi_{il}, \\
(\mathrm{K}_i)_{jl} &= \frac{1}{2} \int_{\tau_i} \varphi_{ij} \cdot \nabla \times \varphi_{il} + \varphi_{il} \cdot \nabla \times \varphi_{ij}, \\
(\mathrm{S}_{ik})_{jl} &= \frac{1}{2} \int_{a_{ik}} \varphi_{ij} \cdot (\varphi_{kl} \times \boldsymbol{n}_{ik}).
\end{aligned}
\tag{4}
$$

2.2 *Time discretization*

The set of local system of ordinary differential equations for each τ_i, Eq. (3), can be formally transformed in a global system. To this end, we suppose that all electric (resp. magnetic) unknowns are gathered in a column vector $\mathbb{E}$ (resp. $\mathbb{H}$) of size $N_g = \sum_{i=1}^{n_t} N_i$ where n_t stands for the number of elements in $\mathcal{T}_h$. Then system (3) can be written as

$$
\begin{aligned}
\mathbb{M}^\varepsilon \partial_t \mathbb{E} &= -\mathbb{M}^\sigma \mathbb{E} + \mathbb{K}\mathbb{H} - \mathbb{A}\mathbb{H} - \mathbb{B}\mathbb{H}, \\
\mathbb{M}^\mu \partial_t \mathbb{H} &= \qquad\quad -\mathbb{K}\mathbb{E} + \mathbb{A}\mathbb{E} - \mathbb{B}\mathbb{E},
\end{aligned}
\tag{5}
$$

where we have used the following definitions and properties

- $\mathbb{M}^\varepsilon, \mathbb{M}^\mu, \mathbb{M}^\sigma$ and $\mathbb{K}$ are $N_g \times N_g$ block diagonal matrices with diagonal blocks equal to $\mathrm{M}_i^\varepsilon, \mathrm{M}_i^\mu, \mathrm{M}_i^\sigma$ and K_i respectively. $\mathbb{M}^\varepsilon, \mathbb{M}^\mu$ and $\mathbb{M}^\sigma$ are symmetric positive definite matrices, and $\mathbb{K}$ is a symmetric matrix.
- $\mathbb{A}$ is also a $N_g \times N_g$ block sparse matrix, whose non-zero blocks are equal to S_{ik} when a_{ik} is an internal interface of $\mathcal{T}_h$. Since $\boldsymbol{n}_{ki} = -\boldsymbol{n}_{ik}$, it can be checked that $(\mathrm{S}_{ik})_{jl} = (\mathrm{S}_{ki})_{lj}$ and then $\mathrm{S}_{ki} = {}^t\mathrm{S}_{ik}$; thus $\mathbb{A}$ is a symmetric matrix.
- $\mathbb{B}$ is a $N_g \times N_g$ block diagonal matrix, whose non-zero blocks are equal to S_{ik} when a_{ik} is a boundary interface of $\mathcal{T}_h$. In that case, $(\mathrm{S}_{ik})_{jl} = -(\mathrm{S}_{ik})_{lj}$; thus $\mathbb{B}$ is a skew-symmetric matrix.

Let $\mathbb{S} = \mathbb{K} - \mathbb{A} - \mathbb{B}$; the system (5) can be rewritten as

$$
\begin{aligned}
\mathbb{M}^\varepsilon \partial_t \mathbb{E} &= -\mathbb{M}^\sigma \mathbb{E} + \mathbb{S}\mathbb{H}, \\
\mathbb{M}^\mu \partial_t \mathbb{H} &= \qquad -{}^t\mathbb{S}\mathbb{E}.
\end{aligned}
\tag{6}
$$

In [17], a fourth-order leap-frog time scheme combined with a DG formulation has been proposed for the Maxwell equations in the case of non-conducting material

($\mathbb{M}^\sigma = 0$). Here we present an extension of this time-stepping scheme to handle conductive materials as well. It is given by

$$
\begin{aligned}
\mathbb{E}^{n+1} &= \Big[\mathbb{I} - \tfrac{\Delta t^3}{24} \mathbb{M}^{-\varepsilon} \mathbb{S} \mathbb{M}^{-\mu}\, {}^t\mathbb{S} \mathbb{M}^{-\varepsilon} \mathbb{M}^\sigma \Big] \mathbb{M}^\alpha \mathbb{E}^n \\
&\quad + \Big[(\mathbb{I} - \mathbb{M}^\alpha) \mathbb{M}^{-\sigma} - \tfrac{\Delta t^3}{24} \mathbb{M}^{-\varepsilon} \mathbb{S} \mathbb{M}^{-\mu}\, {}^t\mathbb{S} \mathbb{M}^{-\varepsilon} \Big] \mathbb{S} \mathbb{H}^{n+\frac{1}{2}}, \\[6pt]
\mathbb{H}^{n+\frac{3}{2}} &= \Big[\mathbb{I} + \tfrac{\Delta t^3}{24} \mathbb{M}^{-\mu}\, {}^t\mathbb{S} \mathbb{M}^{-\varepsilon} \mathbb{M}^\sigma \mathbb{M}^{-\varepsilon} \mathbb{S} \Big] \mathbb{H}^{n+\frac{1}{2}} \\
&\quad + \Delta t \mathbb{M}^{-\mu}\, {}^t\mathbb{S} \Big[-\mathbb{I} + \tfrac{\Delta t^2}{24} \Big(-(\mathbb{M}^{-\varepsilon} \mathbb{M}^\sigma)^2 \\
&\qquad\qquad + \mathbb{M}^{-\varepsilon} \mathbb{S} \mathbb{M}^{-\mu}\, {}^t\mathbb{S} (\mathbb{I} - \tfrac{\Delta t}{2} \mathbb{M}^{-\varepsilon} \mathbb{M}^\sigma) \Big) \Big] \mathbb{E}^{n+1},
\end{aligned}
\tag{7}
$$

where $\mathbb{I}$ is the identity matrix and $\mathbb{M}^\alpha$ is a block diagonal matrix (symmetric positive definite) defined by

$$
\mathbb{M}^\alpha = \exp(-\Delta t \mathbb{M}^{-\varepsilon} \mathbb{M}^\sigma).
$$

The material parameters ε and σ are piecewise constant, then we have that

$$
\mathbb{M}^\varepsilon = \mathbb{M} \mathbb{I}^\varepsilon \quad \text{and} \quad \mathbb{M}^\sigma = \mathbb{M} \mathbb{I}^\sigma,
$$

where $\mathbb{I}^\varepsilon$ and $\mathbb{I}^\sigma$ are block diagonal matrices with diagonal blocks equal to $\varepsilon_i \mathbb{I}$ and $\sigma_i \mathbb{I}$, respectively. Then, the matrix exponential $\mathbb{M}^\alpha$ can be evaluated exactly as

$$
\begin{aligned}
\mathbb{M}^\alpha &= \exp(-\Delta t \mathbb{I}^{-\varepsilon} \mathbb{I}^\sigma) = \exp(-\Delta t \varepsilon_i^{-1} \sigma_i \mathbb{I}) \\
&= \exp(-\Delta t \varepsilon_i^{-1} \sigma_i) \mathbb{I}.
\end{aligned}
$$

In the following, we state a condition on the time step Δt such that the resulting DGTD-$\mathbb{P}_{p_i}$ method, Eq. (7), is stable.

LEMMA 2.1 *For the DGTD-$\mathbb{P}_{p_i}$ method, Eq. (7), the electromagnetic energy*

$$
\mathcal{E}^n = \frac{1}{2} \big({}^t\mathbb{E}^n \mathbb{M}^\varepsilon \mathbb{E}^n + {}^t\mathbb{H}^{n-\frac{1}{2}} \mathbb{M}^\mu \mathbb{H}^{n+\frac{1}{2}} \big),
\tag{8}
$$

is a positive definite quadratic form of all the unknowns provided that

$$
\Delta t \leq \frac{2}{\theta}, \;\; with \;\; \theta = \| \mathbb{M}^{\frac{-\mu}{2}} \mathbb{V} \mathbb{M}^{\frac{-\varepsilon}{2}} \|,
\tag{9}
$$

where $\mathbb{V} = {}^t\mathbb{S} \big[\mathbb{I} - \tfrac{\Delta t^2}{24} \big(-(\mathbb{M}^{-\varepsilon} \mathbb{M}^\sigma)^2 + \mathbb{M}^{-\varepsilon} \mathbb{S} \mathbb{M}^{-\mu}\, {}^t\mathbb{S} (\mathbb{I} - \tfrac{\Delta t}{2} \mathbb{M}^{-\varepsilon} \mathbb{M}^\sigma) \big) \big]$. *Here, $\|.\|$ denotes any matrix norm, and $\mathbb{M}^{\frac{-\gamma}{2}}$ is the inverse square root of $\mathbb{M}^\gamma$.*

Proof Using the second relation of Eq. (7), the energy (8) can be written as

$$
\begin{aligned}
2\mathcal{E}^n &= {}^t\mathbb{E}^n \mathbb{M}^\varepsilon \mathbb{E}^n + {}^t\mathbb{H}^{n-\frac{1}{2}} (\mathbb{M}^\mu + \tfrac{\Delta t^3}{24} {}^t\mathbb{S} \mathbb{M}^{-\varepsilon} \mathbb{M}^\sigma \mathbb{M}^{-\varepsilon} \mathbb{S}) \mathbb{H}^{n-\frac{1}{2}} \\
&\quad - \Delta t\, {}^t\mathbb{H}^{n-\frac{1}{2}} \mathbb{V} \mathbb{E}^n.
\end{aligned}
\tag{10}
$$

The mass matrices $\mathbb{M}^\varepsilon, \mathbb{M}^\mu$ and $\mathbb{M}^\sigma$ are symmetric positive definite and we can construct in a simple way their square root (also symmetric positive definite) denoted

by $\mathbb{M}^{\frac{\varepsilon}{2}}, \mathbb{M}^{\frac{\mu}{2}}$ and $\mathbb{M}^{\frac{\sigma}{2}}$ respectively. We deduce from Eq. (10) that

$$
\begin{aligned}
2\mathcal{E}^n \geq{} & \|\mathbb{M}^{\frac{\varepsilon}{2}}\mathbb{E}^n\|^2 + \|\mathbb{M}^{\frac{\mu}{2}}\mathbb{H}^{n-\frac{1}{2}}\|^2 + \tfrac{\Delta t^3}{24}{}^t\mathbb{H}^{n-\frac{1}{2}}({}^t\mathbb{S}\mathbb{M}^{-\varepsilon}\mathbb{M}^{\sigma}\mathbb{M}^{-\varepsilon}\mathbb{S})\mathbb{H}^{n-\frac{1}{2}} \\
& -\Delta t|\,{}^t\mathbb{H}^{n-\frac{1}{2}}\mathbb{M}^{\frac{\mu}{2}}\mathbb{M}^{\frac{-\mu}{2}}\mathbb{V}\mathbb{M}^{\frac{-\varepsilon}{2}}\mathbb{M}^{\frac{\varepsilon}{2}}\mathbb{E}^n| \\
\geq{} & \|\mathbb{M}^{\frac{\varepsilon}{2}}\mathbb{E}^n\|^2 + \|\mathbb{M}^{\frac{\mu}{2}}\mathbb{H}^{n-\frac{1}{2}}\|^2 + \tfrac{\Delta t^3}{24}{}^t(\mathbb{M}^{-\varepsilon}\mathbb{S}\mathbb{H}^{n-\frac{1}{2}})\mathbb{M}^{\sigma}(\mathbb{M}^{-\varepsilon}\mathbb{S}\mathbb{H}^{n-\frac{1}{2}}) \\
& -\theta\Delta t\|\mathbb{M}^{\frac{\mu}{2}}\mathbb{H}^{n-\frac{1}{2}}\|\|\mathbb{M}^{\frac{\varepsilon}{2}}\mathbb{E}^n\|.
\end{aligned}
$$

The estimate $ab \leq (a^2 + b^2)/2$ leads to

$$
\begin{aligned}
2\mathcal{E}^n \geq{} & \|\mathbb{M}^{\frac{\varepsilon}{2}}\mathbb{E}^n\|^2 + \|\mathbb{M}^{\frac{\mu}{2}}\mathbb{H}^{n-\frac{1}{2}}\|^2 + \tfrac{\Delta t^3}{24}\|\mathbb{M}^{\frac{\sigma}{2}}(\mathbb{M}^{-\varepsilon}\mathbb{S}\mathbb{H}^{n-\frac{1}{2}})\|^2 \\
& -\theta\tfrac{\Delta t}{2}(\|\mathbb{M}^{\frac{\mu}{2}}\mathbb{H}^{n-\frac{1}{2}}\|^2 + \|\mathbb{M}^{\frac{\varepsilon}{2}}\mathbb{E}^n\|^2).
\end{aligned}
$$

We then sum up the lower bounds for the $\mathcal{E}^n$ to obtain

$$
2\mathcal{E}^n \geq (1 - \theta\tfrac{\Delta t}{2})\|\mathbb{M}^{\frac{\varepsilon}{2}}\mathbb{E}^n\|^2 + (1 - \theta\tfrac{\Delta t}{2})\|\mathbb{M}^{\frac{\mu}{2}}\mathbb{H}^{n-\frac{1}{2}}\|^2 + \tfrac{\Delta t^3}{24}\|\mathbb{M}^{\frac{\sigma}{2}}(\mathbb{M}^{-\varepsilon}\mathbb{S}\mathbb{H}^{n-\frac{1}{2}})\|^2.
$$

Then, under the condition in Eq. (9), the energy $\mathcal{E}^n$ is a positive definite quadratic form of the numerical unknowns $\mathbb{E}^n$ and $\mathbb{H}^{n-\frac{1}{2}}$. This concludes the proof. ∎

We note that when $\mathbb{M}^{\sigma} = 0$ we get back the stability condition obtained in [17] for non-conducting materials. The convergence analysis of the DGTD-$\mathbb{P}_{p_i}$ method can be found in [17].

3. Curvilinear tetrahedral elements

For generality we shall limit much of the discussions to the three-dimensional case and regard the two-dimensional problem as a natural simplification. Consider a physical domain $\Omega \subset \mathbb{R}^3$ whose boundary $\partial\Omega$, or a portion of it, is defined by regular parameterized curves. A regular partition of the domain $\bar{\Omega} = \bigcup_{\tau_i \in \mathcal{T}_h} \bar{\tau}_i$ in tetrahedra is assumed such that every interior element τ_i (i.e., an element having at most one vertex on the curved boundary) has only straight edges and planar faces while every curved element has at least one edge on the curved boundary. An element with one edge on the curved boundary has two curved faces defined from the curved edge and an interior tetrahedral vertex. For an element with three edges on the curved boundary, all his faces are curved. A boundary face defined by three curved edges is denoted by Γ_i^F, and a curved face defined from one curved edge is denoted by Γ_i^E, see Figure 1. Each interior element can be defined and treated as standard DG or FE elements, i.e., by using an affine mapping from a reference element τ_r to the physical one. Therefore, in the vast majority of the domain, interpolation and numerical integration are standard. Each curved element is the image of τ_r through high-order geometrical mapping which consists of three ingredients: (a) a quadratic representation of the curved boundary (Section 3.1), (b) a geometric adaptation of the location of points inside a curved element (see the proof of Lemma 3.1), (c) a proper numerical integration scheme to evaluate the matrices in Eq. (4) for each curved element (Section 3.2).
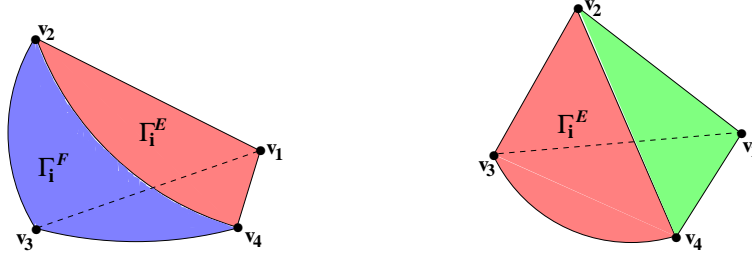
Figure 1. Left: curved tetrahedron τ_i with three curved edges. Right: curved tetrahedron τ_i with one curved edge. A curved face defined from three curved edges is denoted by Γ_i^F while a curved face defined from one curved edge is denoted by Γ_i^E.

3.1   *Quadratic transformation for high-order curved tetrahedra*

Given any curved tetrahedron $\tau_i \in \mathcal{T}_h$ spanned by the four vertices, (v_1, v_2, v_3, v_4), counted counter-clockwise in the Cartesian coordinates $\boldsymbol{x} = (x, y, z)$. We shall assume that τ_i has three curved edges as shown on Figure 2. Let $\tau_r = \{(\xi, \eta, \zeta) \in \mathbb{R}^3 : \xi, \eta, \zeta \geq 0, \xi + \eta + \zeta \leq 1\}$ be the closed straight-sided reference tetrahedron with vertices A_1, A_2, A_3 and A_4 in the $\boldsymbol{\xi} = (\xi, \eta, \zeta)$ coordinates. In τ_r, there exist $M_n = (n+1)(n+2)(n+3)/6$ fundamental nodal points, $A_i, i = 1, \ldots, M_n$, including the four vertices. Let $\{L_i^{(n)}(\boldsymbol{\xi})\}_{i=1}^{M_n}$ be a set of real functions defined in τ_r such that $L_i^{(n)}(A_j) = \delta_{ij}$ (the Kronecker delta) for $i, j = 1, \ldots, M_n$, and $\sum_{i=1}^{M_n} L_i^{(n)} = 1$. Then, the curved tetrahedron τ_i is obtained as the image of τ_r by the mapping

$$\boldsymbol{x} = \Psi_{\tau_i}(\boldsymbol{\xi}) = \sum_{i=1}^{M_n} L_i^{(n)}(\boldsymbol{\xi}) v_i, \tag{11}$$

where v_i, $i \geq 5$ are some nodal points defined in the curved element τ_i; they are enumerated as in Figure 2. Here, n stands for the order of the geometrical mapping or for the order of the curved tetrahedron: $n = 2$ refers to quadratic (or 10-nodes), $n = 3$ refers to cubic (or 20-nodes) and $n = 4$ refers to quartic (or 35-nodes) tetrahedral elements, see Figure 2. For $n = 1$, Eq. (11) becomes a simple affine mapping, see e.g. [17, 18, 23].

To fix the notation within the tetrahedron, let us denote by v_i-v_j the curved edge passing through the points v_i and v_j, and by v_i-v_j-v_k the curved triangular face spanned by the three vertices v_i, v_j and v_k. We shall assume that the nodes along the straight edges in Figure 2 are equally distributed. Now, if we use the standard formula for dividing a line segment in a given ratio, then Eq. (11) reduces to

$$\Psi_{\tau_i}(\boldsymbol{\xi}) = v_1 + (v_2 - v_1)\xi + (v_3 - v_1)\eta + (v_4 - v_1)\zeta$$
$$+ a_{110}^{(n)}\xi\eta + a_{101}^{(n)}\xi\zeta + a_{011}^{(n)}\eta\zeta + \mathcal{H}[n-3]\sum_{3 \leq i+j+k \leq n} a_{ijk}^{(n)}\xi^i\eta^j\zeta^k, \tag{12}$$
$$0 \leq i, j, k \leq n-1, \ (i, j) \neq (0, 0), \ (j, k) \neq (0, 0),$$

where $\mathcal{H}[n-3]$ is the well-known Heaviside step function, and the coefficients $a_{ijk}^{(n)}$ are listed in Appendix A.

The implementation of the mapping in Eq. (12) within a three-dimensional DG solver is hindered by a number of practical difficulties. First, the evaluation of the DG matrices in Eq. (4) is computationally expensive for $n > 2$. For instance, the
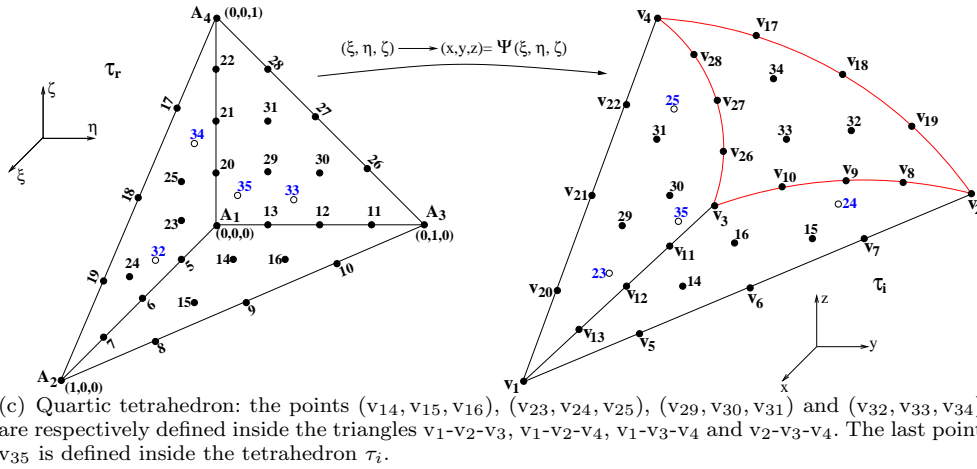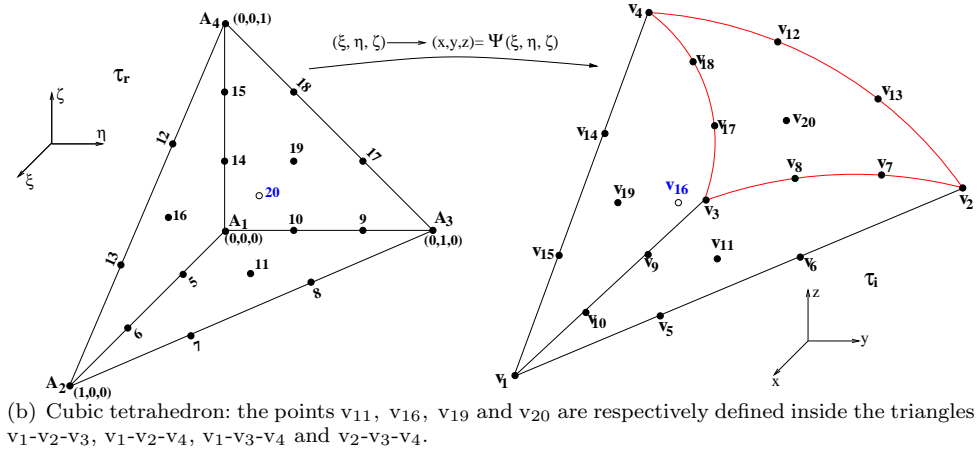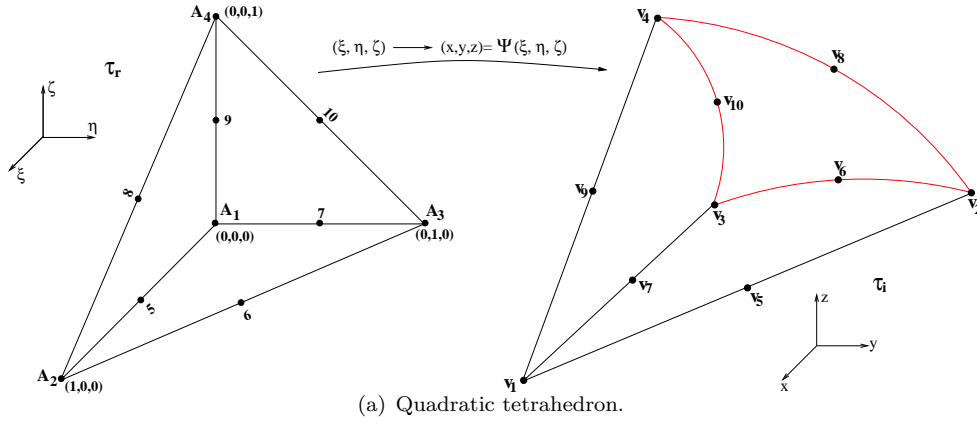
(a) Quadratic tetrahedron.

(b) Cubic tetrahedron: the points v_{11}, v_{16}, v_{19} and v_{20} are respectively defined inside the triangles v_1-v_2-v_3, v_1-v_2-v_4, v_1-v_3-v_4 and v_2-v_3-v_4.

(c) Quartic tetrahedron: the points (v_{14}, v_{15}, v_{16}), (v_{23}, v_{24}, v_{25}), (v_{29}, v_{30}, v_{31}) and (v_{32}, v_{33}, v_{34}) are respectively defined inside the triangles v_1-v_2-v_3, v_1-v_2-v_4, v_1-v_3-v_4 and v_2-v_3-v_4. The last point v_{35} is defined inside the tetrahedron τ_i.

Figure 2. Mapping Ψ_{τ_i} between the master tetrahedron τ_r and the quadratic (a), the cubic (b) and the quartic (c) curved tetrahedron τ_i.

coefficients of the mass matrix on each curved element can be computed as

$$(\mathrm{M}_i^\gamma)_{jl} = \gamma_i \int_{\tau_i} \varphi_{ij}(\boldsymbol{x})\varphi_{il}(\boldsymbol{x})d\boldsymbol{x} = \gamma_i \int_{\tau_r} \varphi_j^\star(\boldsymbol{\xi})\varphi_l^\star(\boldsymbol{\xi})|\mathbf{J}_\Psi(\boldsymbol{\xi})|d\boldsymbol{\xi},$$

where $\{\varphi_j^\star\}_{j=1}^{j=N_i}$ are some basis functions defined on τ_r and $\mathbf{J}_\Psi$ is the Jacobian of the mapping Ψ_{τ_i}. The determinant of the Jacobian, $|\mathbf{J}_\Psi(\boldsymbol{\xi})|$, is a polynomial of degree $3(n-1)$ in $\boldsymbol{\xi}$. For a degree of interpolation p_i, the function to be integrated in local coordinates, $F(\boldsymbol{\xi}) = \varphi_j^\star(\boldsymbol{\xi})\varphi_l^\star(\boldsymbol{\xi})|\mathbf{J}_\Psi(\boldsymbol{\xi})|$, is a polynomial of degree $2p_i + 3(n-1)$. Therefore, the above integral should be computed by a tetrahedral quadrature rule

with degree of exactness $2p_i + 3(n - 1)$ on the reference element τ_r. Nevertheless, high-order ($p_i > 2$ and $n > 2$) quadrature rules over tetrahedra are still suboptimal (see Cools [12]) and computationally expensive, making the assembly a lengthy procedure. The second difficulty is that the optimal convergence of FE methods is provided under some smoothness assumptions on the mapping (12). In particular, the maximum distance between the computational and the exact boundary should be bounded by κh^{p_i}, where κ is a constant and h is the mesh size. Moreover, bounds of the Jacobian of the mapping Ψ_{τ_i} and its first p_i derivatives are also necessary, see Ciarlet and Raviart [8]. In practice, these requirements imply that specific nodal distributions on curved elements are mandatory in order to obtain the optimal rate of convergence, see Lenoir [28]. For instance, with cubic elements, small variations of the interior nodes may lead to suboptimal convergence in the so-called p-version of the FE method [38, 41] (see also [1] and references therein).

In order to overcome these difficulties, we propose to approximate each curved face of the tetrahedron τ_i by a quadratic surface $f(x, y, z) = \sum_{0 \le i+j+k \le 2} p_{ijk} x^i y^j z^k = 0$. This is possible only if we neglect the higher order terms in Eq. (12), i.e. the terms $\sum_{3 \le i+j+k \le n} a^{(n)}_{ijk} \xi^i \eta^j \zeta^k$, leading to a new distribution of the nodal points v_i in the curved tetrahedron. Note that, there is no need to know the equation of the quadratic surface explicitly.

We have the following result.

LEMMA 3.1 *Let τ_i be a curved tetrahedron with three curved edges and three straight edges. Suppose that each curved edge is approximated by a parabolic arc. Then, the mapping Ψ_{τ_i}, Eq. (12), from the master element τ_r onto τ_i can be expressed as*

$$\Psi_{\tau_i}(\boldsymbol{\xi}) = v_1 + e^{(n)}_1 \xi + e^{(n)}_2 \eta + e^{(n)}_3 \zeta + e^{(n)}_4 \xi\eta + e^{(n)}_5 \xi\zeta + e^{(n)}_6 \eta\zeta, \qquad (13)$$

where $e^{(n)}_1 = v_2 - v_1$, $e^{(n)}_2 = v_3 - v_1$, $e^{(n)}_3 = v_4 - v_1$, $\forall\, n = 2, 3, 4$, and,

(i) *Quadratic case (n=2):*
$$\begin{cases} e^{(2)}_4 = 4v_6 - 2v_2 - 2v_3, \\ e^{(2)}_5 = 4v_8 - 2v_2 - 2v_4, \\ e^{(2)}_6 = 4v_{10} - 2v_3 - 2v_4. \end{cases}$$

(ii) *Cubic case (n=3):*
$$\begin{cases} e^{(3)}_4 = \frac{9}{4}[(v_7 + v_8) - (v_2 - v_3)], \\ e^{(3)}_5 = \frac{9}{4}[(v_{12} + v_{13}) - (v_2 - v_4)], \\ e^{(3)}_6 = \frac{9}{4}[(v_{17} + v_{18}) - (v_3 - v_4)]. \end{cases}$$

(iii) *Quartic case (n=4):*
$$\begin{cases} e^{(4)}_4 = \frac{8}{3}[(v_8 + v_{10}) - (v_2 - v_3)], \\ e^{(4)}_5 = \frac{8}{3}[(v_{17} + v_{19}) - (v_2 - v_4)], \\ e^{(4)}_6 = \frac{8}{3}[(v_{26} + v_{28}) - (v_3 - v_4)]. \end{cases}$$

Eq. (13) will be referred to as "the high-order geometrical mapping".

Proof As we have already mentioned, the use of quadratic surface for matching a curved face of a tetrahedron may leads to a new distribution of the nodes inside this tetrahedron. For the implementation of the high-order mapping, one should take into account the specific distribution of the nodes as discussed in the following analysis. This is mandatory to guarantee optimal convergence rates.

The parametric equations of each curved edge in Figure 2 can be obtained by

substituting the coordinates ξ, η and ζ in Eq. (12) such that $\xi + \eta + \zeta = 1$. The coordinates ξ, η and ζ are defined in function of the parameter $0 \leq t \leq 1$ as: $(\xi = t, \eta = 1 - t, \zeta = 0)$ for v_2-v_3, $(\xi = t, \eta = 0, \zeta = 1 - t)$ for v_2-v_4 and $(\xi = 0, \eta = t, \zeta = 1 - t)$ for v_3-v_4. This leads to equations of the form

$$\Psi_{\tau_i}(\boldsymbol{\xi}) = \alpha_0^{(n)} + \alpha_1^{(n)}t + \alpha_2^{(n)}t^2 + \cdots + \alpha_k^{(n)}t^n, \quad n = 2, 3, 4,$$

where $\alpha_k^{(n)}$, $k = 0, 1, \ldots, n$ can be obtained from $a_{ijk}^{(n)}$ values as listed in Appendix A. For $n = 2$, Eq. (13) is a direct consequence of Eq. (12). Let us now analyze the cases $n = 3$ and $n = 4$ in Eq. (12) separately.

Cubic case $(n = 3)$: In this case, the curved edges of the cubic tetrahedron τ_i are spanned respectively by the points $(v_i, i = 2, 3, 7, 8), (v_i, i = 2, 4, 12, 13)$ and $(v_i, i = 3, 4, 17, 18)$. Hence, on the curved edge v_2-v_3 we obtain the following equation

$$\Psi_{\tau_i}(t, 1 - t, 0) = \alpha_0^{(3)} + \alpha_1^{(3)}t + \alpha_2^{(3)}t^2 + \alpha_3^{(3)}t^3. \tag{14}$$

The parametric equation (14) defines a cubic curve passing through the points $(v_i, i = 2, 3, 7, 8)$. Since a cubic curve must possess a double point, which may result in a cusp or a loop in the curve, it is in general undesirable as an approximation to a simple smooth curve [30]. However, for some choice for location of points $(v_i, i = 2, 3, 7, 8)$, the cubic curve in Eq. (14) degenerates to a unique parabola. This can be achieved by setting $\alpha_3^{(3)} = 0$, that is to set $a_{120}^{(3)} - a_{210}^{(3)} = 0$, and this implies

$$v_8 = v_7 - \tfrac{1}{3}(v_2 - v_3).$$

From Eq. (12), we have

$$\Psi_{\tau_i}(\xi, \eta, 0) = v_1 + (v_2 - v_1)\xi + (v_3 - v_1)\eta + a_{110}^{(3)}\xi\eta + (a_{120}^{(3)} + a_{210}^{(3)})(\xi\eta^2 + \xi^2\eta).$$

We choose the point v_{11} inside the curved triangle v_1-v_2-v_3 such that $a_{120}^{(3)} + a_{210}^{(3)} = 0$, we obtain the solution

$$v_{11} = \tfrac{1}{2}(4v_1 + v_2 + v_3 + 3v_7 + 3v_8).$$

The points $(v_i, i = 2, 4, 12, 13)$ and $(v_i, i = 3, 4, 17, 18)$ which interpolate the curved edges v_2-v_4 and v_3-v_4 as well as the points v_{16} and v_{19} inside the curved triangles v_1-v_2-v_4 and v_1-v_3-v_4 are obtained in the same way, yields

$$v_{13} = v_{12} - \tfrac{1}{3}(v_4 - v_2), \quad v_{16} = \tfrac{1}{2}(4v_1 + v_2 + v_4 + 3v_{12} + 3v_{13})$$

$$v_{18} = v_{17} - \tfrac{1}{3}(v_3 - v_4), \quad v_{19} = \tfrac{1}{2}(4v_1 + v_3 + v_4 + 3v_{17} + 3v_{18}).$$

Now, the last point v_{20} can be defined by neglected the high-order terms in the parametric equation of the curved triangle v_2-v_3-v_4, that is to set $a_{111}^{(3)} = 0$ which leads to

$$v_{20} = \tfrac{1}{2}[-2(v_2 + v_3 + v_4) + 3(v_7 + v_8 + v_{12} + v_{13} + v_{17} + v_{18})].$$

Finally, the transformation formulae, Eq. (12), reduces to

$$\Psi_{\tau_i}(\boldsymbol{\xi}) = v_1 + (v_2 - v_1)\xi + (v_3 - v_1)\eta + (v_4 - v_1)\zeta + a_{110}^{(3)}\xi\eta + a_{101}^{(3)}\xi\zeta + a_{011}^{(3)}\eta\zeta,$$

where $a_{110}^{(3)} = \frac{9}{4}(v_7 + v_8 - v_2 - v_3)$, $a_{101}^{(3)} = \frac{9}{4}(v_{12} + v_{13} - v_2 - v_4)$, $a_{011}^{(3)} = \frac{9}{4}(v_{17} + v_{18} - v_3 - v_4)$, and Lemma 3.1 holds for $n = 3$.

Quartic case $(n = 4)$: Here the curved edges of the tetrahedron τ_i are spanned respectively by the points $(v_i, \ i = 2, 3, 8, 9, 10)$, $(v_i, \ i = 2, 4, 17, 18, 19)$ and $(v_i, \ i = 3, 4, 26, 27, 28)$. On the curved edge v_2-v_3 we obtain the following equation

$$\Psi_{\tau_i}(t, 1 - t, 0) = \alpha_0^{(4)} + \alpha_1^{(4)}t + \alpha_2^{(4)}t^2 + \alpha_3^{(4)}t^3 + \alpha_4^{(4)}t^4.$$

Now the choice for the location of points $(v_i, \ i = 2, 3, 8, 9, 10)$ to make the quartic curve to a unique parabola can be achieved by setting $\alpha_3^{(4)} = 0$ and $\alpha_4^{(4)} = 0$ which can be explicitly rewritten as

$$-a_{210}^{(4)} + a_{120}^{(4)} + a_{310}^{(4)} - 2a_{220}^{(4)} + 3a_{130}^{(4)} = 0,$$
$$-a_{130}^{(4)} + a_{220}^{(4)} - a_{310}^{(4)} = 0. \tag{15}$$

Using the explicit relations for the coefficients $a_{ijk}^{(4)}$ as listed in Appendix A, we obtain the following relations

$$v_{10} - v_8 = \tfrac{1}{2}(v_3 - v_2), \quad v_9 = \tfrac{1}{6}(4(v_8 + v_{10}) - (v_2 + v_3)).$$

From Eq. (15), we have

$$a_{130}^{(4)} = \tfrac{1}{2}(a_{220}^{(4)} + a_{210}^{(4)} - a_{120}^{(4)}),$$
$$a_{310}^{(4)} = \tfrac{1}{2}(a_{220}^{(4)} - a_{210}^{(4)} + a_{120}^{(4)}). \tag{16}$$

Substituting the left hand sides of the above equations in the Eq. (12), yields

$$\Psi_{\tau_i}(\xi, \eta, 0) = v_1 + (v_2 - v_1)\xi + (v_3 - v_1)\eta + a_{110}^{(4)}\xi\eta$$

$$+ a_{210}^{(4)}\left(\frac{\xi\eta^3}{2} + \xi^2\eta - \frac{\xi^3\eta}{2}\right) + a_{120}^{(4)}\left(-\frac{\xi\eta^3}{2} + \xi\eta^2 + \frac{\xi^3\eta}{2}\right)$$

$$+ a_{220}^{(4)}\left(\frac{\xi\eta^3}{2} + \xi^2\eta^2 + \frac{\xi^3\eta}{2}\right).$$

We choose the points v_{14}, v_{15} and v_{16} such that $a_{210}^{(4)} = 0$, $a_{120}^{(4)} = 0$ and $a_{220}^{(4)} = 0$. From Eq. (16) we also have $a_{130}^{(4)} = a_{310}^{(4)} = 0$; thus we obtain the following relations

$$v_{14} = \tfrac{1}{12}(6v_1 + v_2 + v_3 + 2v_8 + 2v_{10}),$$
$$v_{14} - v_{15} = \tfrac{1}{12}(3v_1 + v_3 - 4v_8),$$
$$v_{14} - v_{16} = \tfrac{1}{12}(3v_1 + v_2 - 4v_{10}).$$

We proceed in the same way to choose the location points on the curved edges v_2-v_4 and v_3-v_4, as well as the interior points in the triangles v_1-v_2-v_4 and v_1-v_3-v_4.

For v_2-v_4 and v_1-v_2-v_4, we have $a_{i0k}^{(4)} = 0$ for $3 \leq i + k \leq 4$, and

$$v_{17} - v_{19} = \tfrac{1}{2}(v_4 - v_2), \quad v_{18} = \tfrac{1}{6}[4(v_{17} + v_{19}) - (v_2 + v_4)],$$

$$v_{23} = \tfrac{1}{12}(6v_1 + v_2 + v_4 + 2v_{17} + 2v_{19}),$$

$$v_{23} - v_{24} = \tfrac{1}{12}(3v_1 + v_4 - 4v_{19}),$$

$$v_{23} - v_{25} = \tfrac{1}{12}(3v_1 + v_2 - 4v_{17}).$$

For v_3-v_4 and v_1-v_3-v_4, we have $a_{0jk}^{(4)} = 0$ for $3 \leq j + k \leq 4$, and

$$v_{28} - v_{26} = \tfrac{1}{2}(v_4 - v_3), \quad v_{27} = \tfrac{1}{6}[4(v_{26} + v_{28}) - (v_3 + v_4)],$$

$$v_{29} = \tfrac{1}{12}(6v_1 + v_3 + v_4 + 2v_{26} + 2v_{28}),$$

$$v_{29} - v_{30} = \tfrac{1}{12}(3v_1 + v_4 - 4v_{26}),$$

$$v_{29} - v_{31} = \tfrac{1}{12}(3v_1 + v_3 - 4v_{28}).$$

We shall now proceed to determine the interior points v_{32}, v_{33} and v_{34} in the curved triangle v_2-v_3-v_4. Hence, on this curved triangle, we obtain the following equation by substituting $\zeta = 1 - \xi - \eta$ in Eq. (12)

$$\begin{aligned}
\Psi_{\tau_i}(\xi, \eta, 1 - \xi - \eta) = {}& v_4 + (v_2 - v_4 + a_{101}^{(4)})\xi + (v_3 - v_4 + a_{011}^{(4)})\eta \\
& + (a_{111}^{(4)} - a_{101}^{(4)} + a_{110}^{(4)} - a_{011}^{(4)} + a_{112}^{(4)})\xi\eta \\
& + (-a_{111}^{(4)} + a_{121}^{(4)} - 2a_{112}^{(4)})\xi\eta^2 - a_{101}^{(4)}\xi^2 - a_{011}^{(4)}\eta^2 \\
& + (-a_{111}^{(4)} + a_{211}^{(4)} - 2a_{112}^{(4)})\xi^2\eta + (a_{112}^{(4)} - a_{121}^{(4)})\xi\eta^3 \\
& + (-a_{121}^{(4)} + 2a_{112}^{(4)} - a_{211}^{(4)})\eta^2\xi^2 + (a_{112}^{(4)} - a_{211}^{(4)})\xi^3\eta.
\end{aligned} \tag{17}$$

Eq. (17) can be reduced to a quadratic surface by setting

$$a_{112}^{(4)} = a_{121}^{(4)} = a_{211}^{(4)}, \quad a_{112}^{(4)} = -a_{111}^{(4)}, \tag{18}$$

which leads to the following set of linear equations

$$\begin{aligned}
6(v_{32} + v_{33} - 3v_{34}) &= 4v_4 + v_8 + v_{10} - 7v_{17} + v_{19} + v_{26} - 7v_{28}, \\
6(v_{32} - 3v_{33} + v_{34}) &= 4v_3 + v_8 - 7v_{10} + v_{17} + v_{19} - 7v_{26} + v_{28}, \\
6(-3v_{32} + v_{33} + v_{34}) &= 4v_2 - 7v_8 + v_{10} + v_{17} - 7v_{19} + v_{26} + v_{28}.
\end{aligned}$$

The solution of this linear system is

$$\begin{aligned}
v_{32} &= \tfrac{1}{6}(-2v_2 - v_3 - v_4 + 3v_8 + v_{10} + v_{17} + 3v_{19} + v_{26} + v_{28}), \\
v_{33} &= \tfrac{1}{6}(-v_2 - 2v_3 - v_4 + v_8 + 3v_{10} + v_{17} + v_{19} + 3v_{26} + v_{28}), \\
v_{34} &= \tfrac{1}{6}(-v_2 - v_3 - 2v_4 + v_8 + v_{10} + 3v_{17} + v_{19} + v_{26} + 3v_{28}).
\end{aligned}$$

Now, we determine the location of the interior point v_{35} in the curved tetrahedron
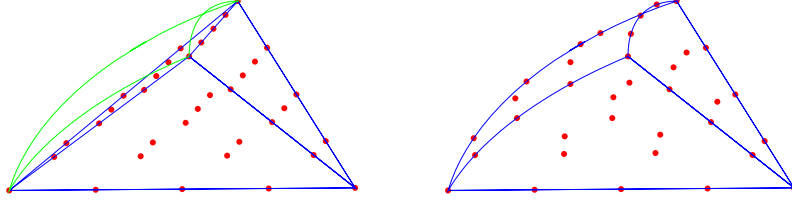
Figure 3. Fourth-order nodal distributions in the quartic straight-sided tetrahedron (left), and adapted to the quartic curved tetrahedron with three curved edges (right).

τ_i. Using Eq. (18) in Eq. (12), yields

$$\Psi_{\tau_i}(\xi, \eta, \zeta) = v_1 + (v_2 - v_1)\xi + (v_3 - v_1)\eta + (v_4 - v_1)\zeta$$
$$+ a_{110}^{(4)}\xi\eta + a_{101}^{(4)}\xi\zeta + a_{011}^{(4)}\eta\zeta + a_{112}^{(4)}(\xi\eta\zeta^2 + \xi\eta^2\zeta + \xi^2\eta\zeta + \xi\eta\zeta).$$

We set $a_{112}^{(4)} = 0$; thus $a_{112}^{(4)} = a_{121}^{(4)} = a_{211}^{(4)} = 0$, we get

$$v_{35} = \tfrac{1}{12}[3v_1 - v_2 - v_3 - v_4 + 2(v_8 + v_{10}) + 2(v_{17} + v_{19}) + 2(v_{26} + v_{28})].$$

Finally, the transformation formulae, Eq. (12), reduces to

$$\Psi_{\tau_i}(\boldsymbol{\xi}) = v_1 + (v_2 - v_1)\xi + (v_3 - v_1)\eta + (v_4 - v_1)\zeta + a_{110}^{(4)}\xi\eta + a_{101}^{(4)}\xi\zeta + a_{011}^{(4)}\eta\zeta,$$

where $a_{110}^{(4)} = \tfrac{8}{3}(v_8 + v_{10} - v_2 - v_3)$, $a_{101}^{(4)} = \tfrac{8}{3}(v_{17} + v_{19} - v_2 - v_4)$, $a_{011}^{(4)} = \tfrac{8}{3}(v_{26} + v_{28} - v_3 - v_4)$, and Lemma 3.1 holds for $n = 4$. ∎

Figure 3 shows an example of fourth-order nodal distributions in the quartic straight-sided tetrahedron and adapted to the quartic curved tetrahedron with three curved edges according to the previous analysis.

Associated with the local mapping, Ψ_{τ_i}, is the Jacobian matrix, $\mathbf{J}_\Psi(\boldsymbol{\xi}) = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\xi}}$, and the determinant

$$|\mathbf{J}_\Psi| = \beta_0 + \beta_1\xi + \beta_2\eta + \beta_3\zeta + \beta_4\xi\eta + \beta_5\xi\zeta + \beta_6\eta\zeta + \beta_7\xi^2 + \beta_8\eta^2 + \beta_9\zeta^2 + \beta_{10}\xi\eta\zeta,$$

which is a third-order polynomial with coefficients $\{\beta_j\}_{j=1}^{j=10} \in \mathbb{R}$.

Remark 1 If τ_i is a curved tetrahedron with one curved edge, v_2-v_3, and five straight edges, then the mapping Ψ_{τ_i}, Eq. (13), is reduced to

$$\Psi_{\tau_i}(\boldsymbol{\xi}) = v_1 + e_1^{(n)}\xi + e_2^{(n)}\eta + e_3^{(n)}\zeta + e_4^{(n)}\xi\eta,$$

where $e_j^{(n)}$ for $j = 1, \ldots, 4$ and $n = 2, 3, 4$ are given in Lemma 3.1. In this case, the determinant of the Jacobian matrix reduces to a first-order polynomial of the form: $|\mathbf{J}_\Psi| = \beta_0 + \beta_1\xi + \beta_2\eta$.

The proposed geometrical mapping has several advantages. First, it is relatively easy to implement within a three-dimensional DG solver. Second, it keeps the degree of the determinant of the Jacobian to a minimum. This will reduce the computational cost resulting from the evaluation of the DG matrices, Eq. (4), cf. Section 3.2. Third, the distribution of the nodal points inside each curved elements ensures the bijectivity of the mapping which leads to optimal convergence rates of the DG method, cf. Section 4.

Moreover, Lemma 3.1 shows that the implicit quadratic surface $f(x, y, z)$ that approximates the original curved surface passes through at most four points of each

original curved edge and so is a reasonable approximation to it. Indeed, we can obtain from Eq. (13) the parametric equations of each of the three curved edges v_2-v_3, v_2-v_4 and v_3-v_4. Taking for instance v_2-v_3, its parametric equations have the following form (the parametric equations of the other two edges have the same form)

$$\mathbf{x}(\xi, 1 - \xi, 0) = s_0^{\mathbf{x}} + s_1^{\mathbf{x}}\xi + s_2^{\mathbf{x}}\xi^2, \qquad \mathbf{x} = x, y, z. \tag{19}$$

If we substitute Eq. (19) in $f(x, y, z)$, then on v_2-v_3 the function f has the form

$$f_0 + f_1\xi + f_2\xi^2 + f_3\xi^3 + f_4\xi^4 = 0. \tag{20}$$

Eq. (20) is a fourth-order polynomial equation in ξ which has at most four roots. Since $f(x, y, z)$ should pass by the end points of v_2-v_3, then $\xi = 0, 1$ are definitely two of the roots of Eq. (20). The other two roots in $0 < \xi < 1$, if they exist, determine two intermediate points v^a and v^b on v_2-v_3. Thus, the parabolic arc that approximate v_2-v_3 can be determined by two intermediate points v^a and v^b and two end points v_2 and v_3. If we have more than two intermediate points on v_2-v_3, then they can all be expressed in terms of v^a and v^b.

3.2 *Numerical integration*

The weak form to be solved requires the evaluation of the matrices in Eq. (4). These matrices involve integrations along element faces and in the element interiors. All integrals in elements not having an edge along the curved boundary are computed using standard procedures, i.e., by using an affine mapping from the master element to the physical element. Since the Jacobian of the affine mapping is constant, the matrices in Eq. (4) can be precomputed and stored for the master element in advance of the main calculation once and for all. Elements with one or three edges on the curved boundary require special attention since the matrices in Eq. (4) should be computed and stored for each curved element. For each curved element τ_i we evaluate the following matrices

$$(\mathrm{M}_i^{\gamma})_{jl} = \gamma_i \int_{\tau_i} \varphi_{ij}(\boldsymbol{x})\varphi_{il}(\boldsymbol{x})d\boldsymbol{x} = \gamma_i \int_{\tau_r} \underbrace{\varphi_j^{\star}(\boldsymbol{\xi})\varphi_l^{\star}(\boldsymbol{\xi})|\mathbf{J}_{\Psi}(\boldsymbol{\xi})|}_{\substack{\text{polynomial of degree} \\ 2p_i + \deg(|\mathbf{J}_{\Psi}|)}} d\boldsymbol{\xi},$$

$$\begin{aligned}
(\mathrm{K}_i)_{jl} &= \frac{1}{2}\int_{\tau_i} [\varphi_{ij}(\boldsymbol{x})\nabla \times \varphi_{il}(\boldsymbol{x}) + \varphi_{il}(\boldsymbol{x})\nabla \times \varphi_{ij}(\boldsymbol{x})]d\boldsymbol{x} \\
&= \frac{1}{2}\int_{\tau_r} [\varphi_j^{\star}(\boldsymbol{\xi})(\mathbf{J}_{\Psi}^{-1})_{jl}(\boldsymbol{\xi})\nabla \times \varphi_l^{\star}(\boldsymbol{\xi}) + \\
&\qquad \varphi_l^{\star}(\boldsymbol{\xi})(\mathbf{J}_{\Psi}^{-1})_{jl}(\boldsymbol{\xi})\nabla \times \varphi_j^{\star}(\boldsymbol{\xi})]|\mathbf{J}_{\Psi}(\boldsymbol{\xi})|d\boldsymbol{\xi} \\
&= \frac{1}{2}\int_{\tau_r} \underbrace{[\varphi_j^{\star}(\boldsymbol{\xi})(\mathbf{j}_{\Psi})_{jl}(\boldsymbol{\xi})\nabla \times \varphi_l^{\star}(\boldsymbol{\xi}) + \varphi_l^{\star}(\boldsymbol{\xi})(\mathbf{j}_{\Psi})_{jl}(\boldsymbol{\xi})\nabla \times \varphi_j^{\star}(\boldsymbol{\xi})]}_{\text{polynomial of degree } (2p_i - 1) + \deg((\mathbf{j}_{\Psi})_{jl})} d\boldsymbol{\xi},
\end{aligned} \tag{21}$$

$$(\mathrm{S}_{ik})_{jl} = \frac{1}{2}\int_{\Gamma^G} \varphi_{ij}(\boldsymbol{x})\varphi_{kl}(\boldsymbol{x})d\boldsymbol{x} = \frac{1}{2}\int_{\Delta_r} \underbrace{\varphi_j^{\star}(\boldsymbol{\xi})\varphi_l^{\star}(\boldsymbol{\xi})|\nabla\mathbf{J}_{\Psi}(\boldsymbol{\xi})|}_{\text{not a polynomial}} d\boldsymbol{\xi},$$

Table 1.   CPU time and the memory overhead per element for computing and storing the matrices in Eq. (21) including the matrix $(\mathrm{M}_i^\gamma)^{-1}$ using the cubature formulae in Eq. (22) at the optimal cubature points.

		2D		3D	
p_i	n	Time (μs)	RAM (KB)	Time (μs)	RAM (KB)
1	2	3.28	0.46	83.3	1.28
2	2	19.7	1.61	387.0	5.99
3	3	75.1	4.25	1736.9	21.55
4	4	216.9	9.25	7133.4	63.25

with the choice of the $\{\varphi_j^\star\}_{j=1}^{j=N_i}$ basis function is flexible and easy to implement, with our preference being the standard Lagrange interpolation functions defined on τ_r. Here, $(\mathbf{J}_\Psi^{-1})_{jl} = (\mathbf{j}_\Psi)_{jl}/|\mathbf{J}_\Psi|$ and $\Phi_\Gamma(\Delta_r) = \Gamma^G$ where Φ_Γ is a parametrization of Γ^G (the index G stands for E or F which corresponds to a curved face with one or three curved edges, see Figure 1). To evaluate these terms we use a cubature formulae which is able to integrate up to 30'th order polynomials in 2D and up to 11'th order polynomials in 3D, on the reference element, to provide a reasonable approximation of the integral of the integrand and the Jacobian, which may be a non-polynomial function. Note that, a cubature is nothing more than a multi-dimensional version of the well-established quadrature formulae. Specifically, a cubature is a set of N_c two- or three-dimensional points $\{\lambda_\ell^c\}_{\ell=1}^{\ell=N_c}$ with N_c associated weights $\{\omega_\ell^c\}_{\ell=1}^{\ell=N_c}$ where the number of points N_c depends on the desired maximum order of polynomial required to be accurately integrated. A survey of cubature formulae can be found in [11]. We evaluate

$$(\mathrm{M}_i^\gamma)_{jl} = \sum_{\ell=1}^{N_c} \omega_\ell^c [\varphi_j^\star \varphi_l^\star |\mathbf{J}_\Psi|]\big|_{\lambda_\ell^c},$$

$$(\mathrm{K}_i)_{jl} = \frac{1}{2} \sum_{\ell=1}^{N_c} \omega_\ell^c [\varphi_j^\star (\mathbf{j}_\Psi)_{jl} \nabla \times \varphi_l^\star + \varphi_l^\star (\mathbf{j}_\Psi)_{jl} \nabla \times \varphi_j^\star]\big|_{\lambda_\ell^c}, \qquad (22)$$

$$(\mathrm{S}_{ik})_{jl} = \frac{1}{2} \sum_{\ell=1}^{N_c} \omega_\ell^c [\varphi_j^\star \varphi_l^\star |\nabla \mathbf{J}_\Psi|]\big|_{\lambda_\ell^c},$$

where each term in the brackets of the right hand sides are evaluated at the cubature points. In our implementation, we use *optimal* cubature points and weights which verify the following criteria: (a) the points λ_ℓ^c are inside τ_r or lie on its boundary; (b) the weights ω_ℓ^c are positive or equal except in a small number of cases where they might be negative; (c) N_c is the minimum number required to achieve the desired accuracy of the cubature rule. The optimal cubature node and weight sets are available for download from [12]. Figure 4 shows a comparison between the number of optimal cubature points and the number of classical Gauss-Legendre (GL) quadrature points, as a function of the polynomial degree. The GL quadrature requires up to $(r+1)^d$ points to integrate exactly a polynomial of degree less or equal to $2r$ over a d-simplex. They can be computed by the so-called Stroud conical rules of d one-dimensional GL quadrature over $[-1,1]$. Table 1 gives the CPU time and the memory overhead per element for computing and storing the matrices in Eq. (21) including the inverse of the mass matrix $(\mathrm{M}_i^\gamma)^{-1}$ for different polynomial degree p_i and different order of the geometrical mapping n. Note that, if we use the GL points to evaluate Eq. (22), the computational cost could be larger (especially in 3D) because of the large number of GL quadrature points.
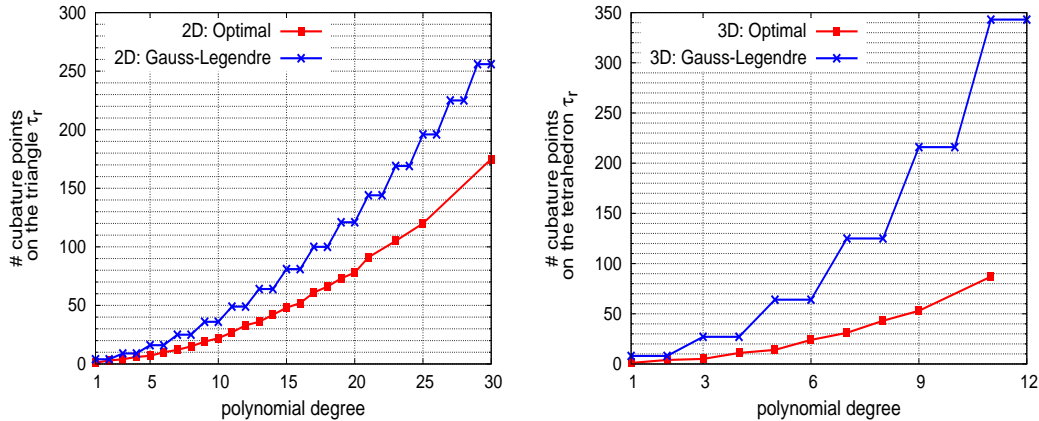
Figure 4. # cubature points versus the polynomial degree. Comparison between the optimal points and the Gauss-Legendre points.

4. Numerical experiments

In this section, we apply the proposed high-order geometrical mapping, Eq. (13), to the simulation of 2D and 3D electromagnetic wave propagation problems. In the following, p_i denotes the local interpolation degree used to approximate the unknowns and n indicates the order of the geometrical mapping. Here $n = 1$ refers to the affine mapping while $n \geq 2$ refers to the high-order geometrical mapping. We recall that the high-order map is only applied on the curved elements while all interior elements are treated using the affine map. In all numerical examples, we present results with polynomial up to degree $p_i = 4$ and we shall compare solutions obtained using the fully affine map (i.e., $n = 1$ for both interior and curved elements) with those obtained using the high-order map (i.e., $n = 1$ for interior elements and $n \geq 2$ for curved elements). The CFL numbers of the DGTD-$\mathbb{P}_{p_i}$ method for $p_i = 1, 2, 3, 4$ are respectively set to 0.95, 0.475, 0.285 and 0.237. For a given p_i, we use the same CFL number independently of n. In the tables of this section, the p-convergence rate $r(p)$ is calculated as $r(p) = -\frac{\log \text{error}(p) - \log \text{error}(p-1)}{\log p - \log(p-1)}$ for $p \geq 2$ where $\text{error}(p)$ denotes the L^2 error when polynomial degree p is used.

4.1 *Two-dimensional examples*

We consider in the following the solution of two different problems which are simple enough that exact solutions exist, yet complex enough not to be trivial. We shall focus the attention on solving the two-dimensional TM-polarized Maxwell's equations in the form

$$\mu_r \partial_t H_x + \partial_y E_z = 0 \ , \quad \mu_r \partial_t H_y - \partial_x E_z = 0 \ , \quad \varepsilon_r \partial_t E_z - \partial_x H_y + \partial_y H_x = 0,$$

subject to boundary conditions between two regions with material parameters, $\varepsilon_r^{(k)}$ and $\mu_r^{(k)}$, for $k = 1, 2$, as $\boldsymbol{n} \times \boldsymbol{H}^{(1)} = \boldsymbol{n} \times \boldsymbol{H}^{(2)}$ and $E_z^{(1)} = E_z^{(2)}$. Here $\boldsymbol{H}^{(k)} = (H_x^{(k)}, H_y^{(k)}, 0)$, and $\boldsymbol{n} = (n_x, n_y, 0)$ represents a unit vector normal to the interface. For the case of a perfectly conducting metallic (PEC) boundary the condition becomes simple as $E_z = 0$.

4.1.1 *Modeling of cylindrical PEC resonators*

We consider a resonator which consists of two concentric PEC cylinders with an electromagnetic wave trapped between the walls. The radii of the cylinders

Table 2.   Characteristics of grids used for the concentric cylinders resonator.

Mesh	M1	M2	M3	M4
# nodes	84	312	1200	4704
# interior elements	120	528	2208	9024
# curved elements	24	48	96	192
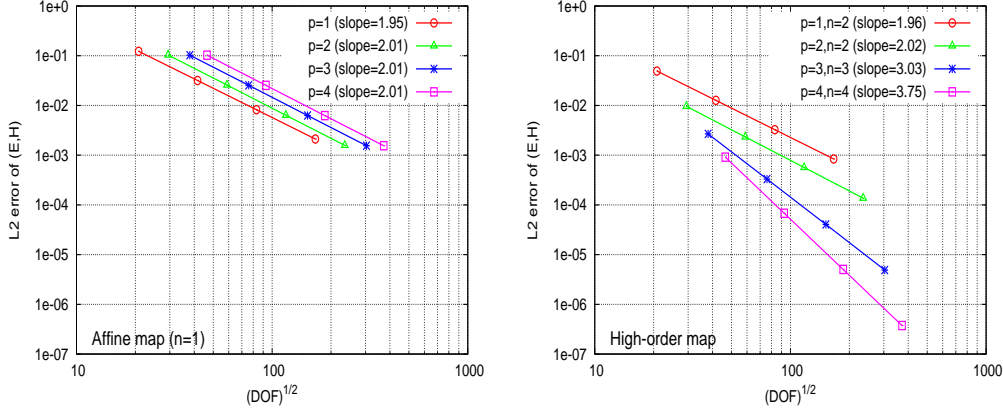
Figure 5.   Concentric cylinders: h-convergence of the DGTD-$\mathbb{P}_p$ method. Errors evaluated after 2 periods.

are $r_1 = 1/6$ m and $r_2 = 1/2$ m. The material is taken to be the vacuum, i.e., $\varepsilon_r = \mu_r = 1$ in normalized units, and the resonant frequency is 0.468 GHz. The exact time-domain solution of this problem is given in [16].

In order to check the accuracy and the convergence properties of the proposed methodology we present computations with uniform degree, i.e., $p_i = p$, $\forall \tau_i \in \mathcal{T}_h$. The various computations have been performed on four successively refined non-uniform grids whose characteristics are summarized in Table 2. Figure 5 shows the h-convergence graphs as a function of the square root of the total number of degrees of freedom (#DOF). The h-convergence rates obtained by the affine map are bounded by 2 $\forall p$, while optimal rates are achieved by the high-order map. Moreover, we observe from Figure 5 that the affine map becomes less efficient in terms of #DOF as the order of approximation p increases. It is clear that the solution accuracy for high degree p is limited by the geometrical error, and that the geometrical error converges at about the same rate as the field error of linear element (i.e., $p = 1$). By comparing all graphs in Figure 5, one can notice that to achieve a given accuracy, the high-order map require less #DOF than the affine map. For instance, for an accuracy of 10^{-3}, the high-order map can save around 80% to 95% of #DOF. Figure 6 shows the temporal behavior of the global L^2 error during 10 periods using a non-uniform mesh (see Figure 7(a)) which consists of 624 nodes, 1056 internal elements, and 96 curved elements (which corresponds to 7 points per wavelength). Table 3 gives the L^2 errors at the final time and the corresponding p-convergence rates. One can observe that the affine map leads to zeroth-order accuracy for $p \geq 2$, while the high-order map achieves exponential convergence. Finally, contour lines of the H_x component for solutions resulting from the affine and the high-order maps are shown on Figure 7 using the DGTD-$\mathbb{P}_2$ method. The affine map leads to large errors that arise in the curved boundaries and pollute the solution inside the domain, which render the use of higher-order DG method useless. Table 3 and Figure 7 confirm that the high-order map strongly reduces the geometrical error and provides more accurate results with errors differing by one to four orders in magnitude.
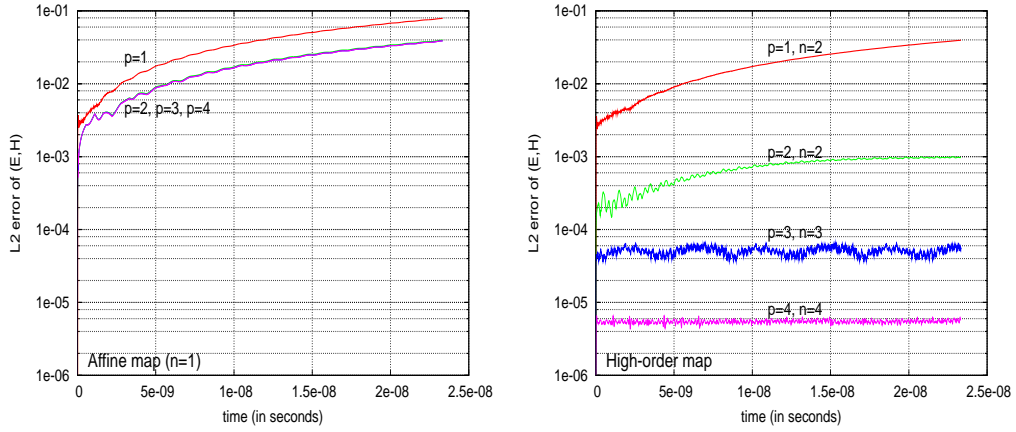
Figure 6. Concentric cylinders: time evolution of the L^2 error during 10 periods.

Table 3.   Concentric cylinders: L^2 errors after 10 periods and convergence rates $r(p)$ for p-refinement.

p	Affine map ($n = 1$)			High-order map ($n \geq 2$)		
	Error	$r(p)$		n	Error	$r(p)$
1	7.88E-02	-		2	3.95E-02	-
2	3.96E-02	0.99		2	9.73E-04	5.34
3	3.87E-02	0.06		3	5.64E-05	7.02
4	3.86E-02	0.01		4	5.77E-06	7.92

(a) Computational mesh.

(b) Exact solution.

(c) Affine map ($n = 1$).

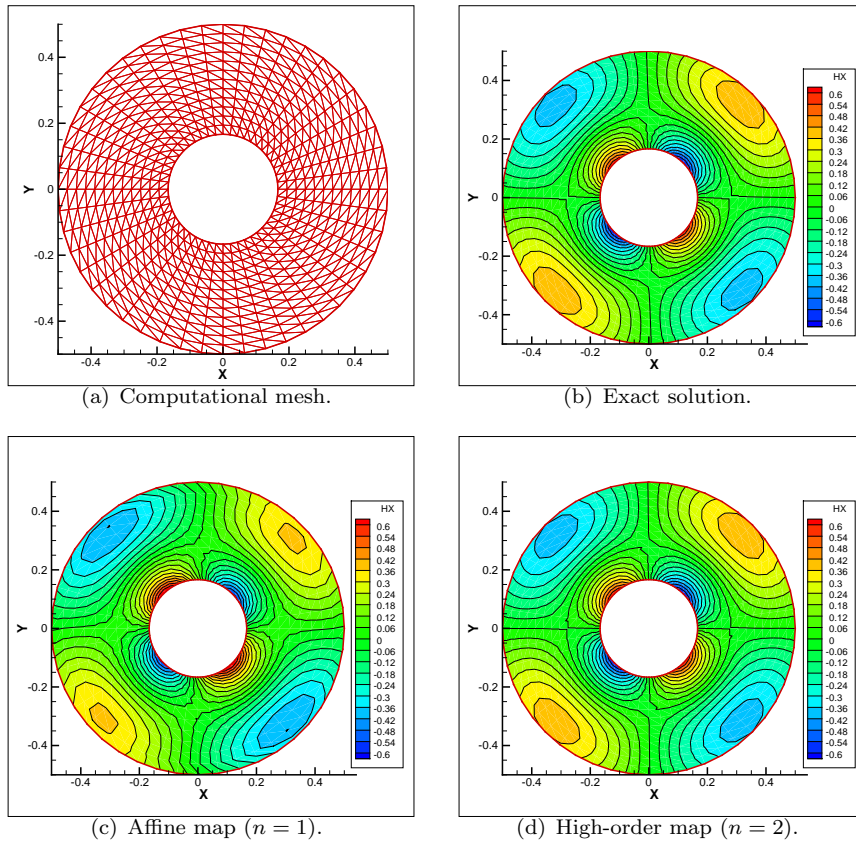(d) High-order map ($n = 2$).

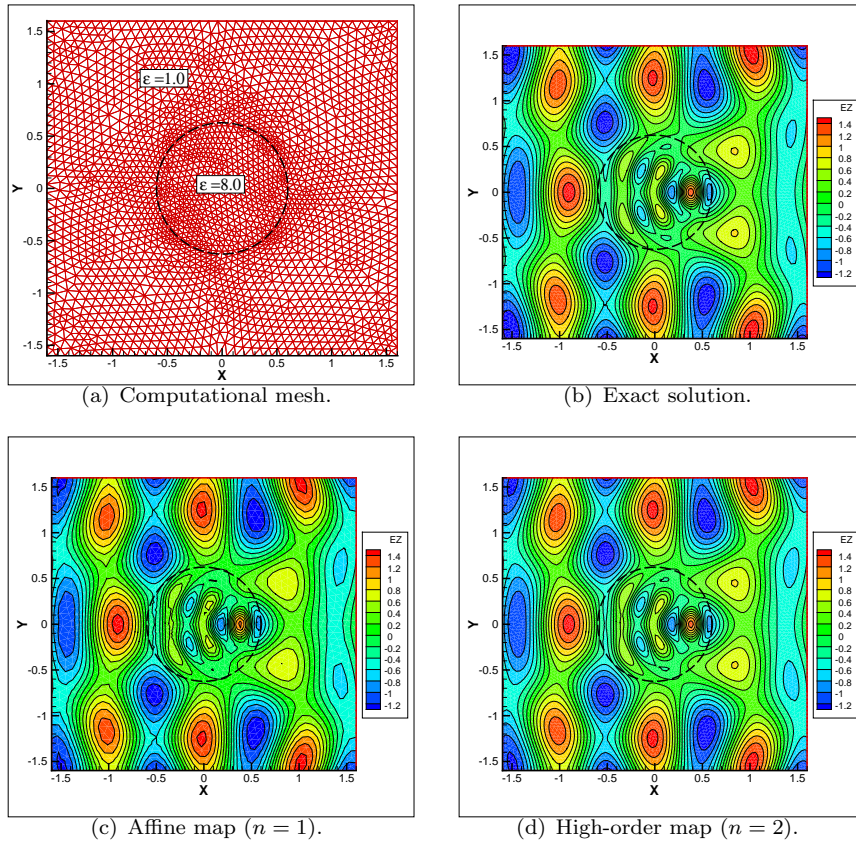Figure 7. Concentric cylinders: contour lines of H_x after 10 periods for $p = 2$.

Table 4. Scattering by a dielectric cylinder: L^2 errors after 10 periods and convergence rates $r(p)$ for p-refinement.

	Affine map ($n = 1$)		High-order map ($n \geq 2$)		
p	Error	$r(p)$	n	Error	$r(p)$
1	8.48E-01	-	2	5.73E-01	-
2	2.17E-01	1.97	2	1.59E-02	5.16
3	7.70E-02	2.55	3	1.09E-03	6.62
4	5.27E-02	1.31	4	1.33E-04	7.31

4.1.2 Plane-wave scattering by a dielectric circular cylinder

As an example of a problem with a material interface, let us consider the scenario shown on Figure 8(a) in which a plane wave with frequency F=300 MHz impinges on a dielectric cylinder, experiencing reflection and refraction at the material interface. The problem is solved in a total field formulation [39]. This test problem has been considered in several works such as [5, 16] where the expression of the analytical solution is detailed. We consider the situation where the material is non-magnetics, and the material exterior to the cylinder is assumed to be vacuum. The internal cylinder has a radius r=0.6 m and bounds a material with relative permittivity ε_r=8.0. The computational domain Ω is bounded by a square of side length $a = 3.2$ m centered at $(0,0)$. A first order Silver-Müller absorbing condition is applied on the boundary of the square. A non-uniform mesh is used which consists of 2714 vertices, 5154 internal elements, and 112 curved elements on the boundary of the cylinder (see Figure 8(a)). The physical simulation time has been set to 10 periods of the incident wave. Similar to the previous case, we give in Table 4 the global L^2 errors as well as the corresponding p-convergence rates at the final time. As for the pure metallic case we see that incorrect treatment of curved material interfaces limits the accuracy of the DG scheme which remains first-order accurate. Contrary to this, the DG method with high-order map achieves exponential convergence and typically yields at least three orders of magnitude improvement in accuracy over the affine map. Contour lines of E_z for the exact solution as well as for solutions resulting from the affine and high-order maps, are illustrated in Figure 8 for $p = 2$. On Figure 9 we compare the x-wise distributions for $y = 0$ of the real part of the discrete Fourier transform of E_z and H_y. The affine map is unable to correctly model the curved material interface and the solutions inside and outside the cylinder are inaccurate, as put in evidence by the geometrical error which develops along the boundary of the cylinder and by the incorrect treatment of the normal. Careful inspection of the computational results in Figures 8-9 confirm this. Similar results has been observed using the hp-like DG method [16], the implicit DG method [6] and the hybrid explicit-implicit DG method [14]. However, solutions obtained by the high-order map are more accurate and the results seem to indicate that a quadratic representation of the boundary is mandatory in order to obtain accurate solutions.

The relevance of this study is two-fold. On one hand it demonstrates the ability of the high-order geometrical mapping to accurately and efficiently model problems with discontinuous coefficients and solutions. Secondly, and perhaps most importantly, it illustrates the inability of the affine map to handle such problems.

(a) Computational mesh.

(b) Exact solution.

(c) Affine map ($n = 1$).

(d) High-order map ($n = 2$).

Figure 8. Scattering by a dielectric cylinder: contour lines of E_z after 10 periods for $p = 2$.
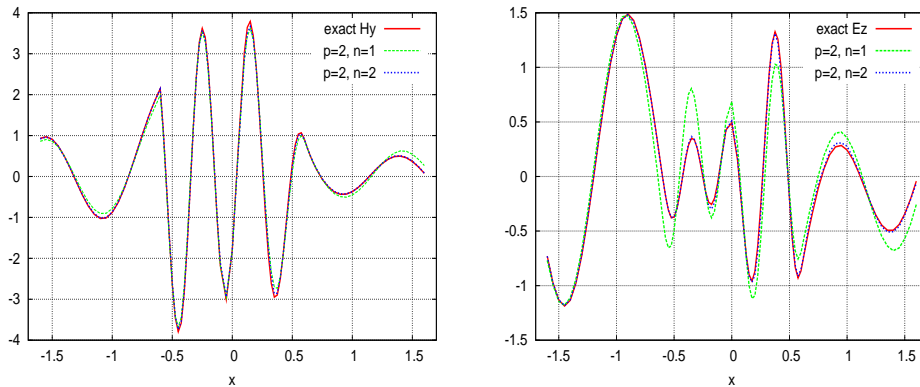
Figure 9. Scattering by a dielectric cylinder: x-wise 1D distribution of the solution for $p = 2$.

4.2 Three-dimensional examples

4.2.1 Eigenmode in a PEC spherical cavity

As a first verification of the general three-dimensional framework, let us consider the propagation of the $(0, 1, 1)$-mode inside a PEC spherical cavity of radius $r = 1$ m. The resonant frequency is 0.131 GHz and the wavelength is 2.29 m. The exact time-domain solution of this problem is given in [18]. All simulations have been carried out for ten periods. Two unstructured tetrahedral meshes have been constructed: a coarse one named MS1, with $2,057$ vertices, $7,840$ internal tetrahedra and $2,400$ curved tetrahedra ($1,120$ of them have only one curved edge), and a finer one named MS2, with $14,993$ vertices, $72,000$ internal tetrahedra and $9,920$ curved tetrahedra ($4,800$ of them have only one curved edge). Here, the finer mesh
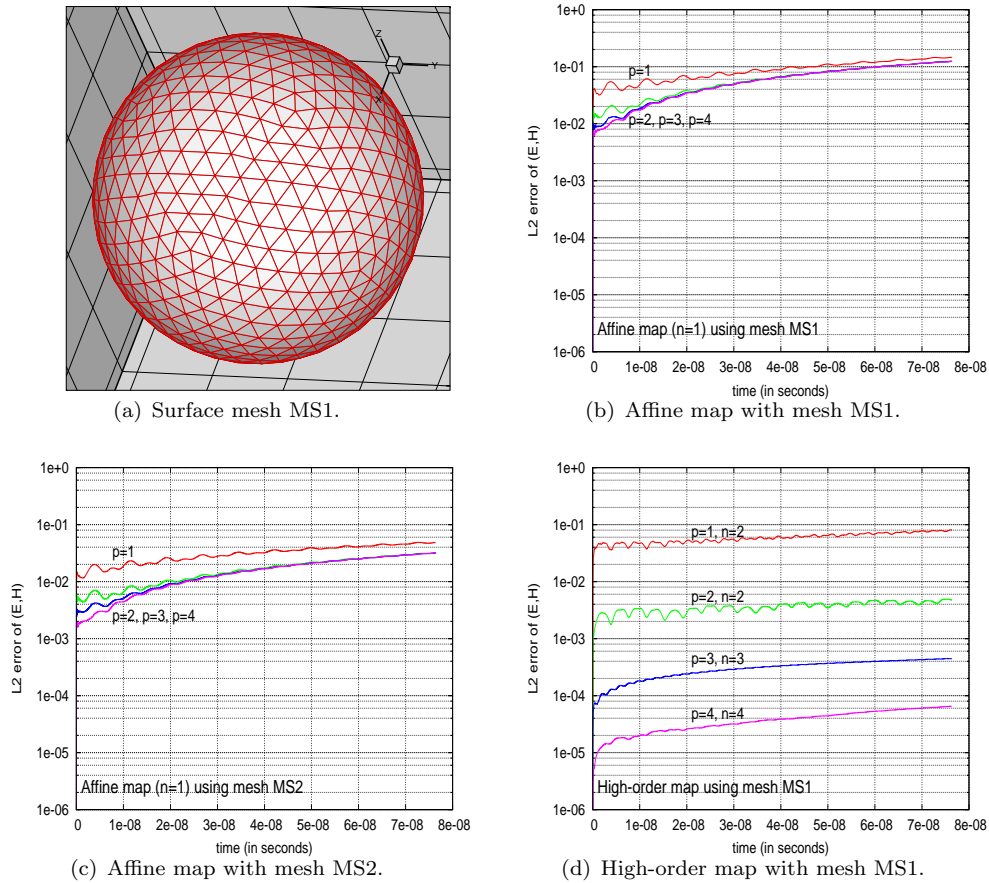
Table 5. Spherical cavity: L^2 errors after 10 periods and convergence rates $r(p)$ for p-refinement.

| | Affine map ($n = 1$) | | | | | High-order map ($n \geq 2$) | | |
| | Mesh MS1 | | Mesh MS2 | | | Mesh MS1 | | |
p	Error	$r(p)$	Error	$r(p)$		n	Error	$r(p)$
1	1.48E-01	-	4.85E-02	-		2	8.07E-02	-
2	1.25E-01	0.24	3.21E-02	0.59		2	4.68E-03	4.11
3	1.25E-01	0.00	3.20E-02	0.00		3	4.45E-04	5.80
4	1.25E-01	0.00	3.20E-02	0.00		4	6.52E-05	6.68

Table 6. Spherical cavity: CPU time to reach 10 periods and #DOF for different p. Here, "RAM" and "Time" are respectively the additional memory overhead and computing time required by the high-order map ($n \geq 2$) for computing and storing the matrices in Eq. (4) for all curved elements.

| | Affine map ($n = 1$) | | | | | High-order map ($n \geq 2$) | | |
| | Mesh MS1 | | Mesh MS2 | | | Mesh MS1 | | |
p	#DOF	CPU time	#DOF	CPU time		n	"RAM"	"Time"
1	40,960	3 min	327,680	44 min		2	3 MB	0.2 s
2	102,400	15 min	819,200	4 h 37 min		2	14 MB	0.9 s
3	204,800	1 h 22 min	1,638,400	26 h 33 min		3	52 MB	4.2 s
4	358,400	4 h 50 min	2,867,200	95 h 20 min		4	152 MB	17.1 s

MS2 has been obtained through a global regular refinement of mesh MS1 (each tetrahedron is divided into 8 tetrahedra [4]). The minimum, maximum and average lengths of the coarser mesh edges are respectively given by $l_{\min}^{\text{MS1}} = 0.1250\,\text{m}$, $l_{\max}^{\text{MS1}} = 0.3703\,\text{m}$ and $l_{\text{avg}}^{\text{MS1}} = 0.1678\,\text{m}$ (which corresponds to 13 points per wavelength) and those of the finer mesh edges are $l_{\min}^{\text{MS2}} = 0.0625\,\text{m}$, $l_{\max}^{\text{MS2}} = 0.2473\,\text{m}$ and $l_{\text{avg}}^{\text{MS2}} = 0.0875\,\text{m}$ (which corresponds to 26 points per wavelength). The surface triangular mesh MS1 of the spherical cavity is shown on Figure 10(a). The simulations discussed here have been performed on Dell Precision M90 workstation equipped with an Intel Core 2 CPU 2.16 GHz processor and 2 GB of RAM memory. We report on results obtained by the DGTD-$\mathbb{P}_p$ method with polynomial interpolation up to degree $p = 4$ using affine map (with meshes MS1 and MS2) and high-order map (with mesh MS1). The overall L^2-error on $(\boldsymbol{E}, \boldsymbol{H})$ are plotted on Figure 10(b)-(d). We give in Table 5 the errors after 10 periods and the corresponding p-convergence rates. We note that the affine map leads to zeroth-order convergence even on the finer mesh while exponential convergence is achieved with the high-order map on the coarser mesh. Moreover, for a given order of interpolation, the errors obtained by the high-order map are between one to four order in magnitude than those obtained by the affine map. Contour lines of the E_x component are shown on Figure 11. The results clearly show that the high-order map produces a smoother solution on the boundary even on the coarser mesh while the mesh refinement considered here is not sufficient to obtain an acceptable solution with the affine map. Finally, performance results for the simulations based on the affine map are summarized in Table 6. In this table, we also give the additional memory overhead and computing time required by the high-order map for computing and storing the matrices in Eq. (4) for all curved elements. The results of Table 6 show that, for the given mesh, the memory overhead associated to the evaluation and the storage of the DG matrices is acceptable while the additional time is negligible. For this problem, the high-order map on the mesh MS1 (for $p = 2$) allows for a reduction in the computing time by a factor of 18.5 and in the #DOF by a factor of 8 compared to those required by the affine map on the mesh MS2 (for $p = 2$). These gains could be larger since the affine map would have required a very fine mesh and a huge computing time to obtain the same kind of accuracy.

(a) Surface mesh MS1.

(b) Affine map with mesh MS1.

(c) Affine map with mesh MS2.

(d) High-order map with mesh MS1.

Figure 10. Spherical cavity: time evolution of the L^2 error for p-refinement.

(a) E_x – affine map – MS1.   (b) E_x – affine map – MS2.   (c) E_x – high-order map – MS1.
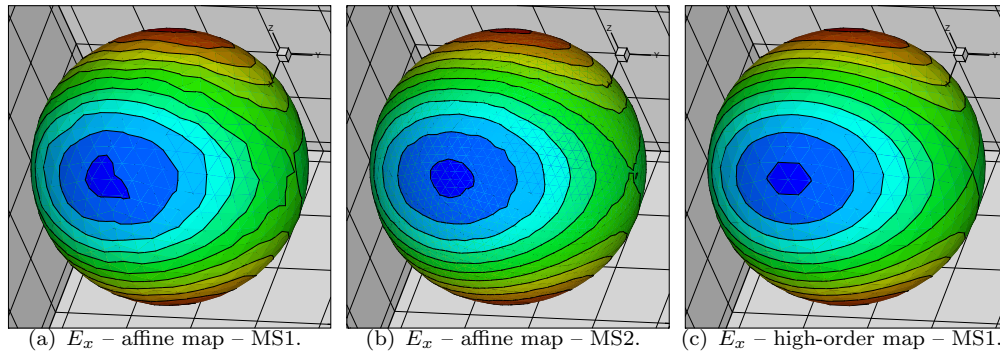
Figure 11. Spherical cavity: contour lines of E_x and E_z after 10 periods for $p = 3$ using affine map ($n=1$) with meshes MS1 and MS2 and high-order map ($n=3$) with mesh MS1.

4.2.2 Multilayered dielectric sphere exposed to a localized source radiation

As a final example of the performance of the three-dimensional framework we shall consider exposure of a multilayered dielectric sphere to a localized source radiation. This problem is based on a realistic application in bio-electromagnetics such as the exposure of human head tissues to electromagnetic fields from mobile phones. The human head in the present example is modeled as a multilayered dielectric sphere of 20 cm diameter and the antenna of the mobile phone is modeled as a dipolar type source with a negligible diameter localized at a distance of 4 mm

away from the head model, as shown in Figure 12, yielding a current of the form

$$\boldsymbol{J}_z^d(\boldsymbol{x}, t) = \delta(\boldsymbol{x} - \boldsymbol{x}_d) f(t), \tag{23}$$

where $f(t)$ is sinusoidally varying temporal signal and $\boldsymbol{x}_d$ is the localization point of the source. This source current is easily introduced and discretized according to the DG formulation discussed in Section 2.1. The dielectric sphere consists of four layers (each of them corresponds to a head tissue) which are brain, cerebro spinal fluid (CSF), skull, and skin. The characteristics of the tissues are summarized in Table 7 where the values of the relative permittivity ε_r, the conductivity σ and the density ρ correspond to a frequency F=1.8 GHz and have been obtained from a special purpose online data base. For all tissues, the relative permeability, μ_r, is set to one, while the variation of ε_r explain how important can it be to take into account accurately heterogeneities in the materials. The physical simulation time has been set to 6 periods of the temporal signal of Eq. (23). A discrete Fourier transform of the components of the electric field is computed during the last period of the simulation. The computational domain is here artificially bounded by a sphere located one wavelength away from the skin. The material between the spherical head and the artificial sphere is assumed to be vacuum, $\varepsilon_r = \mu_r = 1$, which corresponds to a wavelength of $\lambda = 166.7$ mm. The underlying global tetrahedral mesh consists of $20,273$ vertices, $20,477$ internal tetrahedra, $75,186$ curved tetrahedra ($13,869$ of them have only one curved edge), see Figure 13. The minimum, maximum and average lengths of the mesh edges are respectively equal to $l_{\min} = 1.0$ mm, $l_{\max} = 100.0$ mm and $l_{\mathrm{avg}} = 16.2$ mm. We present and discuss numerical results obtained with a uniform approximation degree $p_i = p = 1, 2 \ \forall \tau_i \in \mathcal{T}_h$ as well as with non-uniform degree $p_i = (p_1, p_2)$, where p_1 is the interpolation degree for the approximation of the electromagnetic field in the coarsest elements of the mesh while p_2 is adopted in the smallest elements of the mesh. Typically, in the following, we set $p_1 = 2$ and $p_2 = 1$. The distinction between the coarsest and smallest elements has been done according to the geometrical criterion $c_g(\tau_i) = \min_{j \in \mathcal{V}_i} \frac{V_i V_j}{P_i P_j}$, where V_i and P_i respectively denote the volume and the perimeter of the tetrahedron τ_i. For a threshold $c_g = 4.0\text{E-}07$, only 32% of the mesh elements are treated with the degree p_1.

The quantity of interest involved in the definition of international norms for mobile phones is the Specific Absorption Rate (SAR) which is a measure of the rate at which electric energy is absorbed by the tissues when exposed to a radio-frequency electromagnetic field. The SAR is defined as the power absorbed per mass of tissue and has units of watts per kilogram (W/Kg). It is usually averaged either over the whole body, or over a small sample volume (typically 1g or 10g of tissue). Such SAR calculations are at the basis of numerical dosimetry studies of the exposure of human tissues to microwave radiations from wireless communication systems [3, 19]. These studies are useful for assessing the possible thermal effects (temperature rise in tissues resulting from electric energy dissipation) as well as for compliance testing to regulatory limits. The local SAR at any point inside tissue can be calculated as [13]

$$\mathrm{SAR} = \frac{\sigma}{2\rho} |\boldsymbol{E}|^2 \ (\mathrm{W/Kg}),$$

where σ and ρ represent respectively the specific conductivity and the mass density of the tissue at the point of interest, and $|\boldsymbol{E}|$ refers to the rms electric field value at the same point. Plots of time evolutions of the E_x and E_z components at the same point location near the center of the head are given on Figure 14 (comparison
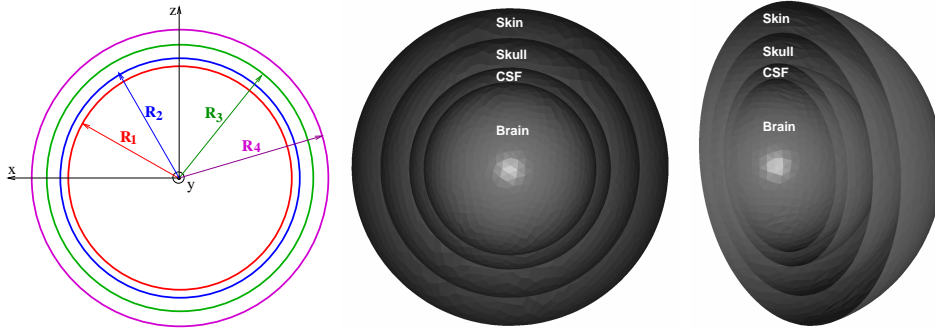
Figure 12. Modeling of the layered sphere with radii (in cm) $R_1 = 9.3$, $R_2 = 9.4$, $R_3 = 9.7$ and $R_4 = 10$.

Table 7. Multilayered dielectric sphere: electromagnetic characteristics of tissues at 1.8 GHz.

Layer	Tissue	ε_r	μ_r	λ (mm)	σ (S/m)	ρ (Kg/m^3)
$R_3 \leq R \leq R_4$	Skin	43.85	1.0	26.73	1.23	1100.0
$R_2 \leq R \leq R_3$	Skull	15.56	1.0	42.25	0.43	1200.0
$R_1 \leq R \leq R_2$	CSF	67.20	1.0	20.33	2.92	1000.0
$0 < R \leq R_1$	Brain	43.55	1.0	25.26	1.15	1050.0

Table 8. Multilayered dielectric sphere: maximum value of the normalized local SAR, CPU time to reach 6 periods and #DOF for different p_i. Here, "RAM" and "Time" are defined in the label of Table 6.

| p_i | #DOF | Affine map ($n = 1$) | | High-order map ($n = 2$) | | |
		Local SAR	CPU time	Local SAR	"RAM"	"Time"
1	382,652	3.151	1 h 10 min	3.278	96 MB	3.5 s
2	956,630	3.221	6 h 46 min	3.325	450 MB	17.6 s
(2,1)	569,492	3.194	2 h 03 min	3.297	202 MB	8.1 s

between the DGTD-$\mathbb{P}_1$, DGTD-$\mathbb{P}_2$ and DGTD-$\mathbb{P}_{(2,1)}$ methods with $n = 1, 2$). Contour lines for $p = 2$ of the local SAR normalized to the maximum value of the local SAR on one hand, and of the local SAR normalized to the total emitted power on the other hand, are shown on Figure 15.

We conclude this numerical study by summarizing in Table 8 the maximum values of the local SAR and the performance results of the calculations reported here. Numerical simulations have been conducted on a workstation equipped with an Intel Xeon 2.33 GHz processor and 32 GB of RAM memory. The results of Figure 14 and Table 8 show that the DGTD-$\mathbb{P}_{(2,1)}$ allows for a reduction in computing time by a factor of 3.3 over the DGTD-$\mathbb{P}_2$ method with a similar accuracy. However, the comparison of accuracy between the affine map and the high-order map is not clear since the patterns of the contour lines (Figure 15) as well as the maximum values of the local SAR seems to be similar. For that, we perform calculation for $p = 1$ and $p = 2$ with $n = 1$ on a finer mesh with $35,490$ vertices and $189,202$ tetrahedra (internal and curved). For this mesh, the maximum local SAR is 3.254 for $p = 1$ (3.263 for $p = 2$) for a total computing time of 9 h 39 min for $p = 1$ (54 h 36 min for $p = 2$). Compared with the cases $p = 1$ and $p = 2$ with $n = 2$ in Table 8, the high-order map allows for a reduction in computing time by a factor of 8.3 for $p = 1$ (8.1 for $p = 2$).

5. Final remarks

High-order geometrical mapping combined with a high-order DG formulation is proposed for the numerical solution of the Maxwell equations. The purpose of this paper has been two-fold. On one hand, we presented a detailed accuracy analysis
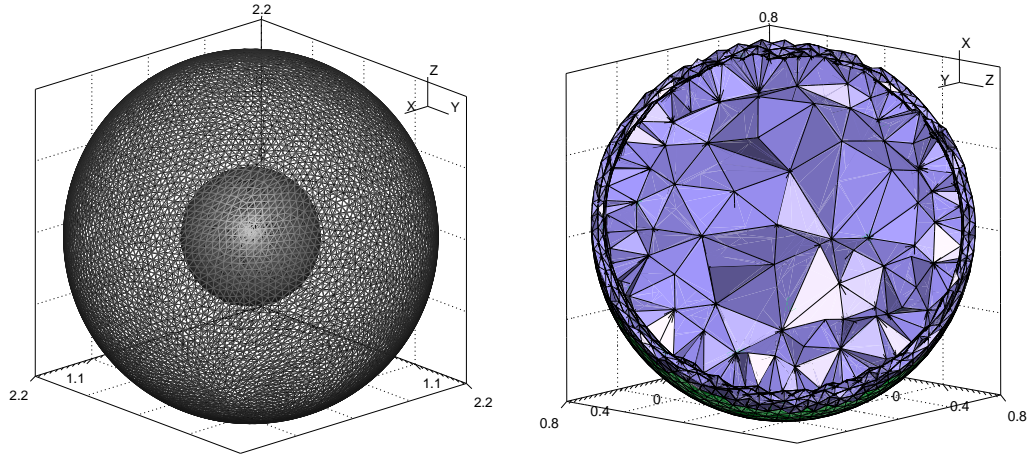
Figure 13.   Left: surface mesh of the layered sphere inside the artificial sphere. Right: volumetric mesh of the layered sphere.
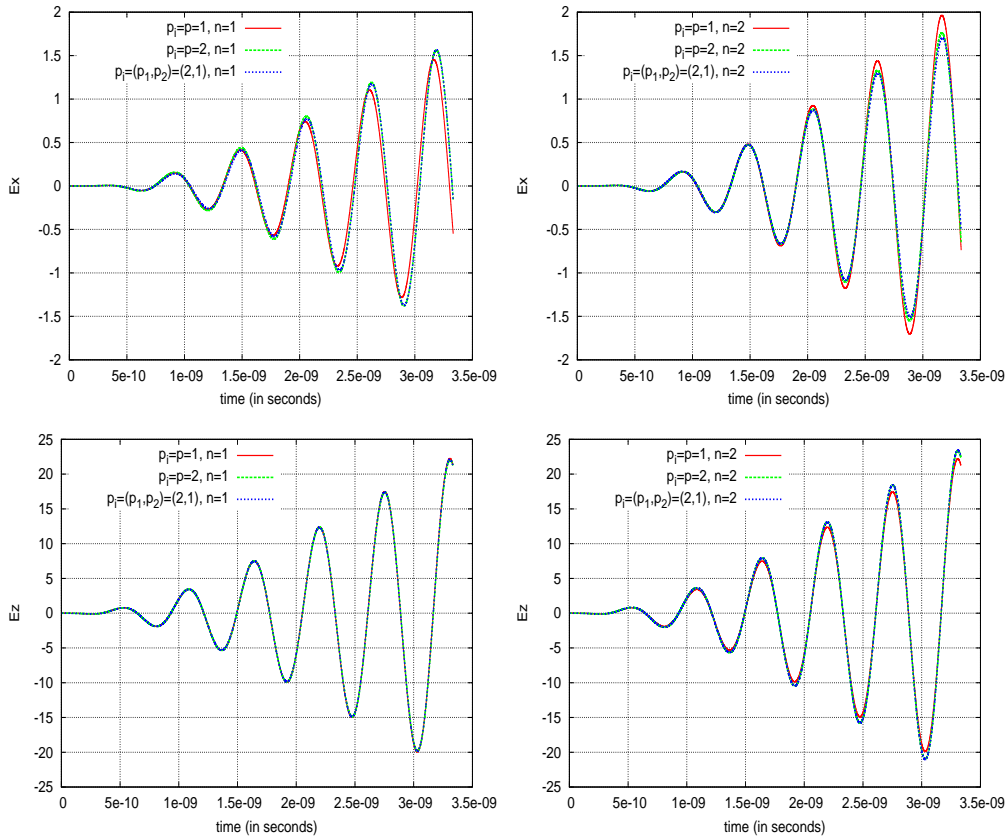

Figure 14. Multilayered dielectric sphere: time evolution of E_x (top) and E_z (bottom).

of the DG method with linear geometric approximation of curved domains. As is well known, and confirmed through the computational examples presented, the use of the affine map leads to large errors that arise in the curved boundaries and pollute the solution inside the domain which render the use of higher-order DG method useless even if the mesh is drastically refined near curved boundaries. The DG method with affine map is at most second-order accurate for h-refinement while under p-refinement it is globally nonconvergent. The situation at material interfaces is even more troubling.
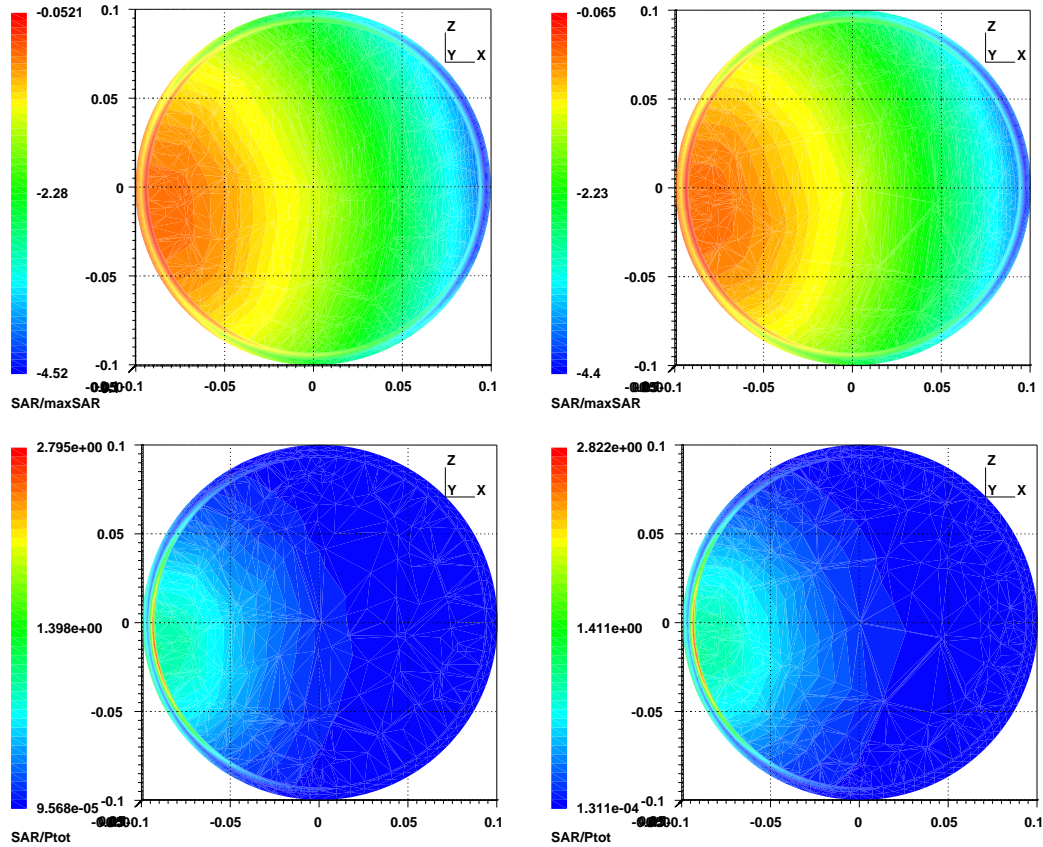
Figure 15.   Multilayered dielectric sphere: contour lines for $p = 2$ of local SAR over maximum local SAR in log scale (top) and the normalized local SAR (bottom) in $y = 0$ cut plane with $n = 1$ (left) and $n = 2$ (right).

The second topic of this paper has been to present a quadratic geometrical mapping for the representation of curved domains and for the proper imposition of the boundary conditions. This high-order geometrical mapping is only applied on element with curved faces and it is based on three ingredients: (a) a quadratic representation of the curved boundary, (b) a geometric adaptation of the location of nodal points inside the curved elements, (c) a proper numerical integration scheme to evaluate the DG matrices. In this case the DG method achieves optimal convergence for h- and p-refinements. With high-order mapping the geometrical error is strongly reduced which yields at least three orders of magnitude improvement over the affine map. The extra cost of the use of the high-order geometrical mapping, due to the numerical integration over elements along the curved boundary, is alleviated by the important saving in the number of degrees of freedom.

While the proposed high-order mapping offers a fairly straightforward way of improving DG methods based on affine map and eliminates the sources of errors for curved domains with known surfaces, it does not take into account geometries with unknown surfaces. The main question is how to choose the nodal points in the curved element. The analysis of Section 3.1 shows that it suffices to choose at least one node on each curved edge to find the remaining nodal points. Encouraged by this fact and by our own initial ideas, we hope to present a high-order geometrical mapping for arbitrarily curvilinear domains in the near future.

Acknowledgements

Appendix A.

A.1 *Quadratic curved tetrahedron*

$$a_{110}^{(2)}=(4v_6-2v_2-2v_3), \quad a_{101}^{(2)}=(4v_8-2v_2-2v_4), \quad a_{011}^{(2)}=(4v_{10}-2v_3-2v_4)$$

A.2 *Cubic curved tetrahedron*

$$a_{111}^{(3)}=9(3v_1+v_2+v_3+v_4-3v_{11}-3v_{16}-3v_{19}+3v_{20}), \quad a_{110}^{(3)}=\tfrac{9}{2}(-2v_1-v_2-v_3-v_7-v_8+6v_{11})$$

$$a_{101}^{(3)}=\tfrac{9}{2}(-2v_1-v_2-v_4-v_{12}-v_{13}+6v_{16}), \quad a_{011}^{(3)}=\tfrac{9}{2}(-2v_1-v_3-v_4-v_{17}-v_{18}+6v_{19})$$

$$a_{012}^{(3)}=\tfrac{9}{2}(2v_1+v_3+3v_{18}-6v_{19}), \quad a_{021}^{(3)}=\tfrac{9}{2}(2v_1+v_4+3v_{17}-6v_{19}), \quad a_{102}^{(3)}=\tfrac{9}{2}(2v_1+v_2+3v_{12}-6v_{16})$$

$$a_{120}^{(3)}=\tfrac{9}{2}(2v_1+v_2+3v_8-6v_{11}), \quad a_{201}^{(3)}=\tfrac{9}{2}(2v_1+v_4+3v_{13}-6v_{16}), \quad a_{210}^{(3)}=\tfrac{9}{2}(2v_1+v_3+3v_7-6v_{11})$$

A.3 *Quartic curved tetrahedron*

$$a_{111}^{(4)}= 32(7v_1+v_2+v_3+v_4-7v_{14}+v_{15}+v_{16}-7v_{23}+v_{24}+v_{25}$$
$$-7v_{29}+v_{30}+v_{31}-v_{32}-v_{33}-v_{34}+8v_{35})$$

$$a_{110}^{(4)}=\tfrac{1}{3}(-96v_1-22v_2-22v_3+16v_8+12v_9+16v_{10}+288v_{14}-96v_{15}-96v_{16})$$

$$a_{101}^{(4)}=\tfrac{1}{3}(-96v_1-22v_2-22v_4+16v_{17}+12v_{18}+16v_{19}+288v_{23}-96v_{24}-96v_{25})$$

$$a_{011}^{(4)}=\tfrac{1}{3}(-96v_1-22v_3-22v_4+16v_{26}+12v_{27}+16v_{28}+288v_{29}-96v_{30}-96v_{31})$$

$$a_{112}^{(4)}=32(-6v_1-v_2-v_3-4v_{25}+8v_{23}+8v_{29}+4v_{14}-4v_{31}+4v_{34}-8v_{35})$$

$$a_{121}^{(4)}=32(-6v_1-v_2-v_4-4v_{16}+8v_{29}+8v_{14}+4v_{23}-4v_{30}+4v_{33}-8v_{35})$$

$$a_{211}^{(4)}=32(-6v_1-v_3-v_4-4v_{15}+8v_{14}+8v_{23}+4v_{29}-4v_{24}+4v_{32}-8v_{35})$$

$$a_{220}^{(4)}=64(-v_1+v_9+4v_{14}-2v_{15}-2v_{16}), \quad a_{202}^{(4)}=64(-v_1+v_{18}+4v_{23}-2v_{24}-2v_{25})$$

$$a_{022}^{(4)}=64(-v_1+v_{27}+4v_{29}-2v_{31}-2v_{30}), \quad a_{013}^{(4)}=\tfrac{32}{3}(-3v_1-v_3+4v_{28}+12v_{29}-12v_{31})$$

$$a_{031}^{(4)}=\tfrac{32}{3}(-3v_1-v_4+4v_{26}+12v_{29}-12v_{30}), \quad a_{103}^{(4)}=\tfrac{32}{3}(-3v_1-v_2+4v_{17}+12v_{23}-12v_{25})$$

$$a_{130}^{(4)}=\tfrac{32}{3}(-3v_1-v_2+4v_{10}+12v_{14}-12v_{16}), \quad a_{301}^{(4)}=\tfrac{32}{3}(-3v_1-v_4+4v_{19}+12v_{23}-12v_{24})$$

$$a_{310}^{(4)}=\tfrac{32}{3}(-3v_1-v_3+4v_8+12v_{14}-12v_{15}), \quad a_{012}^{(4)}=16(4v_1+v_3+10v_{31}-v_{27}+2v_{30}-2v_{28}-14v_{29})$$

$$a_{021}^{(4)}=16(4v_1+v_4+10v_{30}-v_{27}+2v_{31}-2v_{26}-14v_{29}), \quad a_{102}^{(4)}=16(4v_1+v_2+10v_{25}-v_{18}+2v_{24}-2v_{17}-14v_{23})$$

$$a_{120}^{(4)}=16(4v_1+v_2+10v_{16}-v_9+2v_{15}-2v_{10}-14v_{14}), \quad a_{201}^{(4)}=16(4v_1+v_4+10v_{24}-v_{18}+2v_{25}-2v_{19}-14v_{23})$$

$$a_{210}^{(4)}=16(4v_1+v_3+10v_{15}-v_9+2v_{16}-2v_8-14v_{14})$$

References

[1] I. Babuska and B. Guo, *The h-p version of the finite element method for domains with curved boundaries*, SIAM J. Numer. Anal. 25 (1988), pp. 837–861.

[2] F. Bassi and S. Rebay, *High-order accurate discontinuous finite element solution of the 2D Euler equations*, J. Comput. Phys. 138 (1997), pp. 251–285.

[3] P. Bernardi, M. Cavagnaro, S. Pisa, and E. Piuzzi, *Specific absorption rate and temperature increases in the head of a cellular phone user*, IEEE Trans. Microw. Theory Tech. 48 (2000), pp. 1118–1126.

[4] J. Bey, *Tetrahedral grid refinement*, Computing 55 (1995), pp. 355–378.

[5] W. Cai and S. Deng, *An upwinding embedded boundary method for Maxwell's equations in media with material interfaces: 2D case*, J. Comput. Phys. 190 (2003), pp. 159–183.

[6] A. Catella, V. Dolean, and S. Lanteri, *An implicit discontinuous Galerkin time-domain method for two-dimensional electromagnetic wave propagation*, COMPEL 29 (2010), pp. 602–625.

[7] M.H. Chen, B. Cockburn, and F. Reitich, *High-order RKDG methods for computational electromagnetics*, J. Sci. Comput. 22 (2005), pp. 205–226.

[8] P. Ciarlet and P.A. Raviart, *Interpolation theory over curved elements, with applications to finite element methods*, Comput. Methods Appl. Mech. Engrg. 1 (1972), pp. 217–249.

[9] B. Cockburn, G. Karniadakis, and C. Shu (eds.), *Discontinuous Galerkin methods. Theory, computation and applications*, Lecture Notes in Computational Science and Engineering, vol. 11, Springer-Verlag (2000).

[10] G. Cohen, X. Ferrieres, and S. Pernet, *A spatial high-order hexahedral discontinuous Galerkin method to solve Maxwell's equations in time domain*, J. Comput. Phys. 217 (2006), pp. 340–363.

[11] R. Cools, *Monomial cubature rules since "Stroud": a compilation—part 2*, J. Comput. Appl. Math. 112 (1999), pp. 21–27.

[12] ———, *An encyclopaedia of cubature formulas*, J. Complex. 19 (2003), pp. 445–453, software available at http://www.cs.kuleuven.ac.be/~nines/research/ecf/ecf.html.

[13] P. Dimbylow and S. Mann, *SAR calculations in an anatomically realistic model of the head for mobile communication transceivers at 900 MHz and 1.8 GHz*, Phys. Med. Biol. 39 (1994), pp. 1537–1553.

[14] V. Dolean, H. Fahs, L. Fezoui, and S. Lanteri, *Locally implicit discontinuous Galerkin method for time domain electromagnetics*, Tech. Rep. RR-6990, INRIA, 2009, Available at http://hal.inria.fr/inria-00403741.

[15] ———, *Locally implicit discontinuous Galerkin method for time domain electromagnetics*, J. Comput. Phys. 229 (2010), pp. 512–526.

[16] H. Fahs, *Development of a hp-like discontinuous Galerkin time-domain method on non-conforming simplicial meshes for electromagnetic wave propagation*, Int. J. Numer. Anal. Model. 6 (2009), pp. 193–216.

[17] ———, *High-order leap-frog based discontinuous Galerkin method for the time-domain Maxwell equations on non-conforming simplicial meshes*, Numer. Math. Theor. Meth. Appl. 2 (2009), pp. 275–300.

[18] L. Fezoui, S. Lanteri, S. Lohrengel, and S. Piperno, *Convergence and stability of a discontinuous Galerkin time-domain method for the heterogeneous Maxwell equations on unstructured meshes*, ESAIM: Math. Model. Numer. Anal. 39 (2005), pp. 1149–1176.

[19] O. Gandhi, Q.X. Li, and G. Kang, *Temperature rise for the human head for cellular telephones and for peak SARs prescribed in safety guidelines*, IEEE Trans. Microw. Theory Tech. 49 (2001), pp. 1607–1613.

[20] P. Gatto and L. Demkowicz, *Construction of H^1-conforming hierarchical shape functions for elements of all shapes and transfinite interpolation*, Finite Elem. Anal. Des. 46 (2010), pp. 474–486.

[21] W. Gordon and C. Hall, *Construction of curvilinear co-ordinate systems and applications to mesh generation*, Int. J. Numer. Methods Eng. 7 (1973), pp. 461–477.

[22] ———, *Transfinite element methods: Blending-function interpolation over arbitrary curved element domains*, Numer. Math. 21 (1973), pp. 109–129.

[23] J. Hesthaven and T. Warburton, *Nodal high-order methods on unstructured grids. I. Time-domain solution of Maxwell's equations*, J. Comput. Phys. 181 (2002), pp. 186–221.

[24] ———, *Nodal discontinuous Galerkin methods: algorithms, analysis and applications*, Springer Texts in Applied Mathematics, vol. 54, Springer-Verlag (2008).

[25] A. Kanevsky, M. Carpenter, D. Gottlieb, and J. Hesthaven, *Application of implicit-explicit high order Runge-Kutta methods to discontinuous Galerkin schemes*, J. Comput. Phys. 225 (2007), pp. 1753–1781.

[26] G. Karniadakis and S. Sherwin, *Spectral/hp Element Methods for Computational Fluid Dynamics*, 2nd ed., Oxford University Press, Oxford (2005).

[27] L. Krivodonova and M. Berger, *High-order accurate implementation of solid wall boundary conditions in curved geometries*, J. Comput. Phys. 211 (2006), pp. 492–512.

[28] M. Lenoir, *Optimal isoparametric finite elements and error estimates for domains involving curved boundaries*, SIAM J. Numer. Anal. 23 (1986), pp. 562–580.

[29] X. Luo, M. Shephard, and J. Remacle, *The influence of geometric approximation on the accuracy of higher order methods*, Tech. Rep. 2002-1, SCOREC, Rensselaer Polytechnic Institute, 2002.

[30] A. Mitchell, *Basis functions for curved elements in the mathematical theory of finite element*, in *The Mathematics of Finite Elements and Applications II*, J. Whiteman, ed., Academic Press, London, 1976, pp. 43–58.

[31] E. Montseny, S. Pernet, X. Ferrieres, and G. Cohen, *Dissipative terms and local time-stepping improvements in a spatial high order discontinuous Galerkin scheme for the time-domain Maxwell's equations*, J. Comput. Phys. 227 (2008), pp. 6795–6820.

[32] S. Pernet and X. Ferrieres, *HP a-priori error estimates for a non-dissipative spectral discontinuous Galerkin method to solve the Maxwell equations in the time domain*, Math. Comp. 76 (2007), pp. 1801–1832.

[33] A. Perronnet, *Transfinite interpolation on triangle, tetrahedron and pentahedron. application to the generation of meshes and dirichlet's condition*, C. R. Acad. Sci. Paris, Ser. I 326 (1998), pp. 117–122.

[34] S. Piperno, *Symplectic local time stepping in non-dissipative DGTD methods applied to wave propagation problem*, ESAIM: Math. Model. Num. Anal. 40 (2006), pp. 815–841.

[35] D. Sármány, M. Botchev, and J. van der Vegt, *Dispersion and dissipation error in high-order Runge-Kutta discontinuous Galerkin discretisations of the Maxwell equations*, J. Sci. Comput. 33 (2007), pp. 47–74.

[36] L. Scott, *Finite element techniques for curved boundaries*, Ph.D. thesis, Massachusetts Institute of Technology (1973).

[37] S. Sherwin and J. Peiró, *Mesh generation in curvilinear domains using high-order elements*, Int. J. Numer. Methods Eng. 53 (2002), pp. 207–223.

[38] B. Szabó and I. Babuska, *Finite Element Analysis*, 2nd ed., John Wiley & Sons, New York (1991).

[39] A. Taflove, *Computational Electrodynamics: The Finite-Difference Time-Domain Method*, Artech House, Boston, London (1995).

[40] E. Wachspress, *A rational basis for function approximation: II. Curved sides*, J. Inst. Math. Appl. 11 (1973), pp. 83–104.

[41] O. Zienkiewicz, J. Zhu, and N. Gong, *Effective and practical h-p-version adaptive analysis procedures for the finite element method*, Int. J. Numer. Methods Eng. 28 (1989), pp. 879–891.

[42] M. Zlámal, *Curved elements in the finite element method*, SIAM J. Numer. Anal. 10 (1973), pp. 229–240.

[43] ———, *The finite element method in domains with curved boundaries*, Int. J. Numer. Methods Eng. 5 (1973), pp. 367–373.