Institut für Numerische Simulation
Rheinische Friedrich-Wilhelms-Universität Bonn

Maharavo Randrianarivony

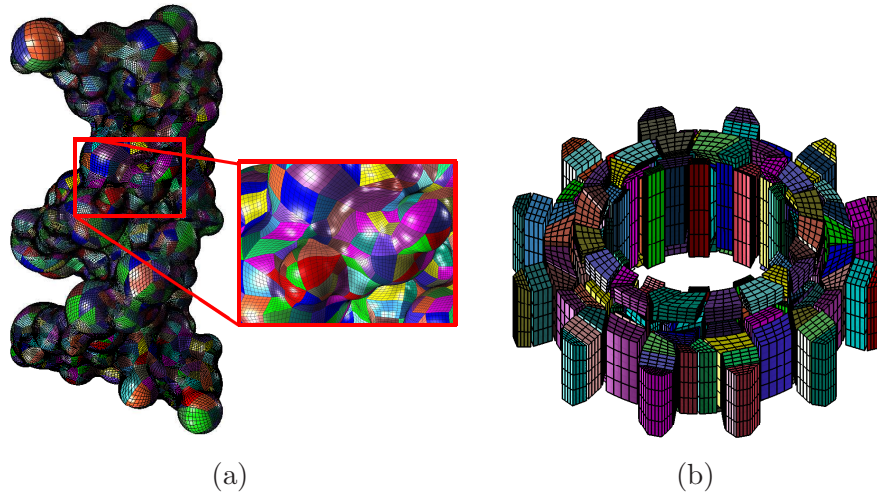# ON A B-SPLINE ADAPTIVE DISCONTINUOUS GALERKIN SIMULATION ON EXACT GEOMETRIES

FIGURE 1. Parametric decompositions: (a) DNA molecule, (b)Hexahedral decomposition

1. INTRODUCTION

B-Spline and NURBS techniques have already been successfully used in Isogeometric Analysis which is a method for directly integrating CAD models in numerical simulations. Our purpose is to improve existing techniques to enhance the efficiency. First, we use local B-Spline subdivisions and knot insertions for the goal of achieving better accuracy in simulations where we concentrate on two and three dimensions. Traditional simulation methods employ fine meshes to represent geometries. Such a representation suffers from a geometric flaw: an unnecessarily large degree of freedom is often required to capture the geometric accuracy. In this paper, we propose a method that can be categorized as a meshless one because we do not consider a fine mesh to represent the initial geometry but rather a set of very coarse parametric spline blocks.

Before presenting our method, a survey of some pertaining works is in order. The initial purpose of Bézier and B-splines entities [3, 10] was to design curves and surfaces especially for car bodies and CAD components. But later they found their use in different disciplines such as molecular modelings and statistical data processing. The desire to apply simulations on curved models is not new. In fact, Höllig and Reif have [16] used the WEB spline to approximate the solution to a PDE. Curved elements motivate equally the isogeometric analysis using NURBS (Non-Uniform Rational B-Spline) described in [17]. As for wavelets, the Wavelet-Galerkin method [14, 15] is able to produce a good accuracy with low computational cost by means of adaptivity. Harbrecht and Randrianarivony [14, 15] have successfully applied Wavelet methods on CAD and molecular models similar to those in Fig. 1(a). As inputs, they accept a CAD

file in an IGES format [22] or a molecular model in PDB format. In the domain of CAD preprocessings, some former works are as follows. For transfinite interpolations [8, 12], Coons patches usually serve as tools to generate the mappings [10] from parametric domains. Brunnett and Randrianarivony have proposed [19] a splitting method for CAD surfaces. They have also invested a lot to implement their methods by using the IGES format [22]. But they did not treat the global continuity of the resulting patches. For molecular surfaces, global continuity can be obtained exactly [14, 15, 20] because all boundary curves are circular arcs which can be easily parametrized. That is not the case for other CAD curves which need more careful treatments. The main task in [21] is the correlation between the Coons patch which resides in an individual patch and the global continuity.

This paper is organized as follows. We start by formulating the problem in Section 2 which contains also the space of approximation and the discontinous Galerkin (DG) scheme using B-splines. We formulate there the treatment of the problem on the parameter domain instead of the physical domain. We will see in Section 3 the establishment of an a-posteriori error estimator [1]. That will be deduced from the de Boor-Fix functional. In Section 4, we describe some outcomes of practical adaptive simulation using our DG method.

2. Discontinuous Galerkin Using B-Splines

2.1. **Parametric Setting.** Our purpose in this document is to solve a Poisson problem having a Dirichlet boundary condition on a multi-dimensional domain $\mathbf{\Omega} \subset \mathbb{R}^d$ where $d = 2, 3$. More precisely, we would like to solve the following problem

$$(2.1) \qquad \Delta u(\mathbf{x}) = f(\mathbf{x}) \qquad \text{for} \qquad \mathbf{x} \in \mathbf{\Omega},$$

$$(2.2) \qquad u(\mathbf{x}) = g(\mathbf{x}) \qquad \text{for} \qquad \mathbf{x} \in \partial\mathbf{\Omega}$$

where the domain $\mathbf{\Omega}$ is supposed to be constituted of a set of very coarse parametric patches as in Fig. 1(b). That is, we suppose that there are mappings $\mathcal{M}_i$ such that

$$(2.3) \qquad \mathbf{\Omega} = \bigcup_{i=1}^N \mathcal{M}_i \Big(\prod_{j=1}^d [a_{i,j}, b_{i,j}] \Big).$$

For the sake of notational convenience, we suppose that we have only a single patch $\mathcal{M} = \mathcal{M}_1$ as illustrated in Fig. 2(b). Later on, the domain $\mathbf{\Omega}$ will be termed *physical domain* while $\mathbf{P} := \prod_{i=1}^d [a_i, b_i]$ *parameter domain*. We suppose that $\mathcal{M}$ is invertible, differentiable and that it admits a regularity condition meaning that the Jacobian matrix $D\mathcal{M}$ has nonvanishing determinant everywhere. In addition, we suppose that
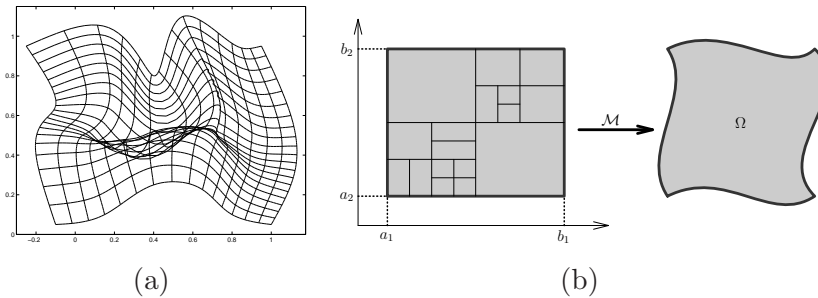
FIGURE 2. (a)Counter example of a regular mapping $\mathcal{M}$, (b) A discretization $\mathcal{T}_h$ on the parameter domain and a mapping onto the physical domain $\boldsymbol{\Omega}$.

there is some positive constant μ such that the linear operators $D\mathcal{M}$ and $D\mathcal{M}^{-1}$ have the next bounds

$$(2.4) \qquad \|D\mathcal{M}\|_* := \sup_{\mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}} \frac{\|D\mathcal{M}\mathbf{x}\|}{\|\mathbf{x}\|} \le \mu, \qquad \text{and} \qquad \|D\mathcal{M}^{-1}\|_* \le \mu.$$

Finally, we suppose that $\mathcal{M}$ is of smoothness $\mathcal{C}^m$ where m is sufficiently large. The mapping $\mathcal{M}$ is in general supposed to be a B-spline or NURBS patch [10]. A counter example of a mapping violating the above condition is observed in Fig. 2(a). An efficient preparation of such mappings is discussed in [18, 19, 21]. For two integers $k \ge 1$ and $n \ge k - 1$, the definition of B-spline basis functions with respect to the knot sequence $\boldsymbol{\zeta} = (\zeta_i)_{i=0}^{n+k}$ uses the divided difference of the truncated power functions $(\cdot - t)_+^k$ given by

$$(2.5) \qquad (x - t)_+^k := \begin{cases} (x - t)^k & \text{if} \quad x \ge t, \\ 0 & \text{if} \quad x < t. \end{cases}$$

More precisely, one has the definition and support property

$$(2.6) \qquad N_i^k(t) = N_i^{n,k,\boldsymbol{\zeta}}(t) := (\zeta_{i+k} - \zeta_i)[\zeta_i, ..., \zeta_{i+k}](\cdot - t)_+^{k-1},$$

$$(2.7) \qquad \text{Supp}(N_i^k) = [\zeta_i, \zeta_{i+k}].$$

To ensure that the B-spline functions are open, we assume that the knot sequence $\boldsymbol{\zeta}$ is *clamped*. That is, the sequence $\zeta_0, ..., \zeta_{n+k}$ is provided as: $\zeta_0 = \cdots = \zeta_{k-1}$ and $\zeta_{n+1} = \cdots = \zeta_{n+k}$. For the multi-variate case $(d = 2, 3)$, let us consider d spline properties $(n_1, k_1, \boldsymbol{\zeta}^1), ..., (n_d, k_d, \boldsymbol{\zeta}^d)$ and let us define

$$(2.8) \qquad \left(N_{i_1}^{n_1,k_1,\boldsymbol{\zeta}^1} \otimes \cdots \otimes N_{i_d}^{n_d,k_d,\boldsymbol{\zeta}^d}\right)(t_1, ..., t_d) := N_{i_1}^{n_1,k_1,\boldsymbol{\zeta}^1}(t_1) \cdots N_{i_d}^{n_d,k_d,\boldsymbol{\zeta}^d}(t_d).$$

We denote the space of d-dimensional splines by

$$(2.9) \qquad \mathbb{S}[n_1, k_1, \boldsymbol{\zeta}^1; \cdots; n_d, k_d, \boldsymbol{\zeta}^d] := \text{span}\left\{N_{i_1}^{n_1,k_1,\boldsymbol{\zeta}^1} \otimes \cdots \otimes N_{i_d}^{n_d,k_d,\boldsymbol{\zeta}^d}\right\}.$$

The efficient de Boor algorithm [3, 10] serves as a method to evaluate a B-spline function without using the above divided differences.

In our present method, we do not use a mesh on the physical domain $\mathbf{\Omega}$. Instead, we have a non-conforming discretization $\mathcal{T}_h$ on the parameter domain as depicted in Fig. 2(b). More precisely, the discretization $\mathcal{T}_h$ is composed of elements $Q_1,...,Q_N$ satisfying the following conditions.

(C1) Each element is a closed domain $Q_i = \prod_{\nu=1}^{d}[r_{i,\nu}, s_{i,\nu}] \subset \mathbf{P}$.

(C2) For two different indices i, j, we have $\mathring{Q}_i \cap \mathring{Q}_j = \emptyset$ where $\mathring{Q}$ designates the topologic interior of Q.

(C3) We have $\mathbf{P} = \cup_{i=1}^{N} Q_i$.

According to those assumptions, the discretization $\mathcal{T}_h$ is allowed to be non-conforming. For an element $Q := \prod_{\nu=1}^{d}[r_\nu, s_\nu]$, we denote $h_\nu(Q) := s_\nu - r_\nu$ so that the element measure is $h(Q) := \prod_{\nu=1}^{d} h_\nu(Q)$. In addition, we assume the shape regularity condition throughout this document. That is, there is a positive constant ρ independent of Q such that

$$(2.10) \qquad\qquad r(Q)/R(Q) \leq \rho \qquad\qquad \forall Q \in \mathcal{T}_h,$$

where $R(Q)$ is the largest circle for $d = 2$ (resp. sphere for $d = 3$) contained in Q while $r(Q)$ is the smallest circle (resp. sphere) including Q. An internal edge (resp. face) is defined to be a nonempty intersection e of two different elements $Q_i \in \mathcal{T}_h$ and $Q_j \in \mathcal{T}_h$ such that e is not a point. Similarly, a boundary edge (resp. face) is a nonempty intersection of an element $Q_i \in \mathcal{T}_h$ and the boundary $\partial \mathbf{P}$ which is not a point. The sets of internal and boundary edges (resp. faces) are denoted by $\mathcal{J}_h$ and $\mathcal{B}_h$ respectively. We will denote by $\mathcal{E}_h = \mathcal{J}_h \cup \mathcal{B}_h$ the set of all edges (resp. faces). Finally, the length (resp. area) of an edge (resp. face) $e \in \mathcal{E}_h$ is denoted by $h(e)$.

For the mesh $\mathcal{T}_h$ on the parameter domain, the approximating space will be

$$(2.11) \qquad \mathbb{V}_h := \left\{ s \in \mathbb{L}_2(\mathbf{P}) : \quad s_{|Q} \in \mathbb{S}\big[n_1, k_1, \boldsymbol{\zeta}^1; \cdots ; n_d, k_d, \boldsymbol{\zeta}^d\big] \qquad \forall Q \in \mathcal{T}_h \right\}$$

where the spline properties depend on Q. That is, $n_i = n_i(Q)$, $k_i = k_i(Q)$ and $\boldsymbol{\zeta}^i = \boldsymbol{\zeta}^i(Q)$ for $i = 1, ..., d$.

2.2. Parametric Discontinuous Galerkin.

The values of a function from $\mathbb{V}_h$ generally do not admit continuities at element interfaces. As a consequence, let us introduce the *jump* value $[\![v]\!]$ and *average* value $\{\!\{v\}\!\}$. For an internal edge $e \in \mathcal{J}_h$ such that $e = \partial Q_1 \cap \partial Q_2$, let $\mathbf{n}_{Q_1}(\mathbf{x})$ and $\mathbf{n}_{Q_2}(\mathbf{x})$ designate the outward normals at $\mathbf{x} \in e$ with respect to $Q_1 \in \mathcal{T}_h$ and $Q_2 \in \mathcal{T}_h$ respectively. For a scalar valued function v, the jump

is defined to be the vector

$$(2.12) \qquad [\![v]\!](\mathbf{x}) := u_{Q_1}(\mathbf{x})\mathbf{n}_{Q_1}(\mathbf{x}) + u_{Q_2}(\mathbf{x})\mathbf{n}_{Q_2}(\mathbf{x}) \qquad \forall \, \mathbf{x} \in e.$$

Additionally, the average is defined as

$$(2.13) \qquad \{\!\{v\}\!\}(\mathbf{x}) := 0.5\big(v_{Q_1}(\mathbf{x}) + v_{Q_2}(\mathbf{x})\big) \qquad \forall \, \mathbf{x} \in e.$$

The jump and average for a boundary edge $e \in \mathcal{E}_h$ are defined similarly where the exterior value is assumed to be zero. Later, the unknown function is approximated by a function in $\mathbb{V}_h$ where the jump values are constrained to be zero with the help of some penalty terms.

The broken Sobolev space with respect to the non-conforming discretization $\mathcal{T}_h$ is denoted by

$$(2.14) \qquad \mathbb{H}^k(\mathcal{T}_h) := \big\{ w \in \mathbb{L}^2(\Omega) : \quad w_{|Q} \in \mathbb{H}^k(Q) \quad \forall \, Q \in \mathcal{T}_h \big\}.$$

We will need also

$$(2.15) \qquad \|w\| := \Big[\sum_{Q \in \mathcal{T}_h} |w|_{1,Q}^2 + h(Q)^2 |w|_{2,Q}^2 + \sum_{e \in \mathcal{E}_h} \frac{1}{h(e)} \big\|[\![w]\!]\big\|_{0,e}^2 \Big]^{1/2}.$$

By using DG variational formulations [2, 4], the initial problem in (2.1) reduces to seek $u_h \in \mathbb{V}_h$ such that

$$(2.16) \qquad \mathcal{B}(u_h, v_h) = \mathcal{L}(v_h) \qquad \forall \, v_h \in \mathbb{V}_h$$

where $\mathcal{B}$ and $\mathcal{L}$ are as follows when expressed in terms of the parameter domain

$$\mathcal{B}(u,v) := \sum_{Q \in \mathcal{T}_h} \int_Q (D\mathcal{M}^T)^{-1} \nabla_{\mathbf{t}} u \cdot (D\mathcal{M}^T)^{-1} \nabla_{\mathbf{t}} v \, \det D\mathcal{M} \, d\mathbf{t}$$

$$- \sum_{e \in \mathcal{E}_h} \int_e \{\!\{(D\mathcal{M}^{-1})(\nabla_{\mathbf{t}} u)\}\!\} \cdot [\![v]\!] \det D\mathcal{M} \, d\mathbf{t}$$

$$- \sum_{e \in \mathcal{E}_h} \int_e [\![u]\!] \cdot \{\!\{(D\mathcal{M}^{-1})(\nabla_{\mathbf{t}} v)\}\!\} \det D\mathcal{M} \, d\mathbf{t} + \frac{1}{\eta} \sum_{e \in \mathcal{E}_h} \int_e [\![u]\!] \cdot [\![v]\!] \det D\mathcal{M} \, d\mathbf{t},$$

$$\mathcal{L}(v) := \sum_{Q \in \mathcal{T}_h} \int_Q f \circ \mathcal{M} \det D\mathcal{M} d\mathbf{t} - \sum_{e \in \mathcal{B}_h} \int_e g \circ \mathcal{M} \Big[\nabla v \cdot \mathbf{n} + \frac{1}{\eta} v \Big] \det D\mathcal{M} d\mathbf{t}$$

in which η is a certain large positive number. The above expressions are similar to the usual DG variational formulation [2, 4, 5] but we use the Jacobians to transform them onto the parameter domain. We will need the operator $\mathcal{A}$ which is defined to be such that

$$(2.17) \qquad \mathcal{B}(u, \phi) = \langle \mathcal{A}u, \phi \rangle \qquad \forall \, \phi \in \mathbb{H}^k(\mathcal{T}_h).$$

Note that although it is possible to reformulate the above form $\mathcal{B}$ on the physical domain $\mathbf{\Omega}$ by using curved elements, we want to avoid that because it is difficult to apply geometric operations such as refinments to curved entities.

Lemma 2.1. *Under the above conditions on the mapping $\mathcal{M}$, we have*

(2.18) $$\frac{1}{\mu} \leq \|D\mathcal{M}\|_* \leq \mu, \qquad \text{and} \qquad \frac{1}{\mu} \leq \|D\mathcal{M}^{-1}\|_* \leq \mu.$$

Thus, the determinant verifies

(2.19) $$0 < C_1 \leq \det(D\mathcal{M}) \leq C_2.$$

Hence, with respect to $\|\cdot\|$, the bilinear form $\mathcal{B}$ admits coercivity

(2.20) $$\|f\|^2 \leq C_{\mu,\mathbf{\Omega}}\, \mathcal{B}(f,f),$$

and boundedness

(2.21) $$\left|\mathcal{B}(f,g)\right| \leq C_{\mu,\mathbf{\Omega}}\|f\|\|g\|$$

Proof. Use the property of $\mathcal{M}$, and proceed as in [2, 4, 5].

$\square$

$\square$

3. Adaptive Simulation

3.1. **Adaptive Refinments and spline operators.** In an adaptive simulation, we deduce a finer discretization $\mathcal{T}_{h+1}$ from a coarse one $\mathcal{T}_h$ by refining some elements $Q \in \mathcal{T}_h$. We consider two kinds of 2D subdivisions. The first one consists in bisecting the rectangle $Q \in \mathcal{T}_h$ by inserting a vertical cut resulting in two sub-rectangles of the same size as shown in Fig. 3(a). The second one does the same but with a horizontal cut as in Fig. 3(b). Note that those two subdivisions could deteriorate the shape regularity (2.10). As a consequence, we choose the subdivision such that the aspect ratios of the resulting rectangles do not exceed the threshold ρ. In fact, it is possible to subdivide in both directions but it is a bit more difficult to apply space hierarchies (see Section 3.3) for that. The generalization to 3D is done in a straightforward manner as illustrated in Fig. 3(c),(d),(e).

In the next description, we show a method to establish if an element $Q \in \mathcal{T}_h$ should be split or not. It is anyhow beyond the scope of this paper to determine which kind of subdivision is optimal. In order to know the elements $Q \in \mathcal{T}_h$ which ought to be subdivided, we need an a-posteriori error indicator. For that purpose, let us first introduce
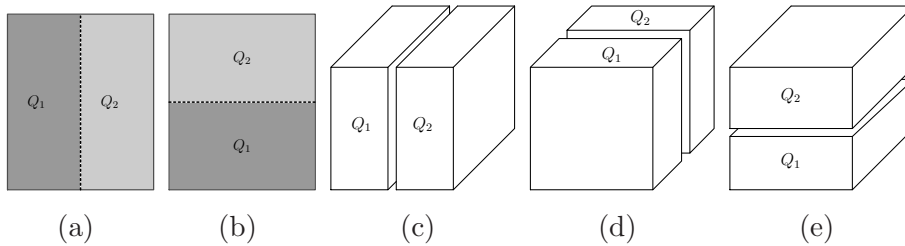
FIGURE 3. (a)Vertical 2D-subdivision (b)Horizontal 2D-subdivision (c)3D-subdivision along x-axis (d)3D-subdivision along y-axis (e)3D-subdivision along z-axis.

some interesting definitions [3, 13] about spline dual functionals where we consider the spline property $(n, k, \boldsymbol{\zeta})$. We define for $i = 0, ..., m$

$$(3.22) \qquad \vartheta_i(t) := \frac{1}{(k-1)!} \prod_{p=1}^{k-1} (t - \zeta_{i+p}).$$

We introduce also the function z_i for $i = 0, ..., m$ as

$$(3.23) \qquad \begin{aligned} z_i(t) &:= 0 & \text{for} & \quad t \leq \zeta_i, \\ z_i(t) &:= \vartheta_i(t) & \text{for} & \quad t \geq \zeta_{i+k}. \end{aligned}$$

Inside the interval $[\zeta_i, \zeta_{i+k}]$, the function z_i is defined to be a smooth function having $\mathcal{C}^{(k-1)}$-joints at ζ_i and ζ_{i+k}:

$$(3.24) \qquad z_i^{(m)}(\zeta_i) = 0, \qquad z_i^{(m)}(\zeta_{i+k}) = \vartheta_i^{(m)}(\zeta_{i+k}) \qquad \forall m = 0, ..., k-1.$$

For that, use for example a higher order Hermite interpolations [11]. For a univariate square integrable function φ in the interval $[\zeta_0, \zeta_{n+k}]$, we define for $i = 0, ..., n$

$$(3.25) \qquad \lambda_i(\varphi) := \int_{\zeta_i}^{\zeta_{i+k}} \varphi(t) \, z_i^{(k)}(t) dt.$$

Under some mild assumption, it can be proved by simple partial integrations that λ_i coincides with the usual de Boor-Fix functional [3, 13]

$$(3.26) \qquad \lambda_i(\varphi) = \sum_{m=1}^{k} (-1)^{k-m} \vartheta_i^{(m-1)}(\tau) \varphi^{(k-m)}(\tau)$$

for some $\tau \in [\zeta_i, \zeta_{i+k}]$. In the present context, (3.25) is more suitable than relation (3.26) that requires point evaluations of φ which are not appropriate if φ is only square integrable. As for the multivariate case $(d = 2, 3)$, let us fix some $\mathbf{n} = (n_1, ..., n_d)$, $\mathbf{k} = (k_1, ..., k_d)$ and $Q = [\zeta_0^1, \zeta_{n_1+k_1}^1] \times \cdots \times [\zeta_0^d, \zeta_{n_d+k_d}^d]$. For $\mathbf{i} = (i_1, ..., i_d)$ where each i_ν belongs to $\{0, 1, ..., n_\nu\}$ in which $\nu = 1, ..., d$, we define for $\varphi \in L^2(Q)$

$$(3.27) \qquad \boldsymbol{\lambda_i}(\varphi) := \int_{\zeta_{i_1}}^{\zeta_{i_1+k_1}} \cdots \int_{\zeta_{i_d}}^{\zeta_{i_d+k_d}} \varphi(t_1, ..., t_d) \, z_{i_1}^{(k_1)}(t_1) \cdots z_{i_d}^{(k_d)}(t_d) \, dt_1 \cdots dt_d.$$

Note that for a tensor product function $\varphi(t_1, ..., t_d) = \prod_{\nu=1}^{d} \varphi_\nu(t_\nu)$, we have $\boldsymbol{\lambda_i}(\varphi) = \prod_{\nu=1}^{d} \lambda_{i_\nu}(\varphi_\nu)$. The above functional admits B-spline duality: for a tensor product B-spline basis function $N_{\mathbf{i}}^{\mathbf{k}} = N_{i_1}^{k_1} \otimes \cdots \otimes N_{i_d}^{k_d}$ we have $\boldsymbol{\lambda_i}(N_{\mathbf{j}}^{\mathbf{k}}) = \delta_{\mathbf{i},\mathbf{j}}$. Finally, for a square integrable function φ in Q, we define the B-spline quasi-interpolant

$$(3.28) \qquad P(\varphi) := \sum_{\mathbf{i} \in I} \boldsymbol{\lambda_i}(\varphi) N_{i_1}^{k_1} \otimes \cdots \otimes N_{i_d}^{k_d}$$

where the sum is over $I := \{0, ..., n_1\} \times \cdots \times \{0, ..., n_d\}$.

3.2. **A-posteriori Error Indicator.** In this section, we discuss about a-posteriori errors where we suppose that the solution u_h of (2.16) with respect to the current discretization $\mathcal{T}_h$ is available.

Theorem 3.1. *For each element $Q \in \mathcal{T}_h$ in which we have the B-spline properties $k_i = k_i(Q)$, $n_i = n_i(Q)$, for $i = 1, ..., d$, we consider the next error indicator*

$$\varepsilon(Q) := \|f \circ \mathcal{M} - \mathcal{A}u_h\|_{0,Q} \left[\prod_{i=1}^{d} \frac{\sqrt{n_i - k_i + 2}}{\sqrt{h_i(Q)}} \right] \left[\sum_{i=1}^{d} \left(\frac{k_i h_i(Q)}{n_i - k_i + 2} \right)^2 \right]^{1+(d/4)}.$$

Under the quasi-uniformity condition

$$(3.29) \qquad \max_{i=k-1,...,n} |\zeta_i - \zeta_{i+1}| \bigg/ \min_{i=k-1,...,n} |\zeta_i - \zeta_{i+1}| \leq \theta < \infty,$$

we have the following reliability relation

$$\|u - u_h\| \leq c(\theta)\varepsilon(\mathcal{T}_h) = c(\theta) \left[\sum_{Q \in \mathcal{T}_h} \varepsilon(Q)^2 \right]^{1/2}.$$

Proof. Denote by $P_{\mathbb{V}_h}$ the projection to $\mathbb{V}_h$ such that inside each $Q \in \mathcal{T}_h$, $P_{\mathbb{V}_h}$ is the quasi-interpolant P_Q as defined in (3.28) with respect to the spline properties $\left(n_1(Q), k_1(Q), \boldsymbol{\zeta}^1(Q)\right), ..., \left(n_d(Q), k_d(Q), \boldsymbol{\zeta}^d(Q)\right)$. From the boundedness (2.21), the coercivity (2.20), the operator $\mathcal{A}$ of relation (2.17) and an orthogonality relation, one obtains

$$
\begin{aligned}
\|u - u_h\| &\leq \mathcal{B}\left(u - u_h, \frac{u - u_h}{\|u - u_h\|}\right) \leq \sup_{\|\phi\|=1} \mathcal{B}(u - u_h, \phi) = \\
&= \sup_{\|\phi\|=1} \mathcal{B}\left(u - u_h, \phi - P_{\mathbb{V}_h}(\phi)\right) \\
&= \sup_{\|\phi\|=1} \langle \mathcal{A}u - \mathcal{A}u_h, \phi - P_{\mathbb{V}_h}(\phi) \rangle = \sup_{\|\phi\|=1} \langle f \circ \mathcal{M} - \mathcal{A}u_h, \phi - P_{\mathbb{V}_h}(\phi) \rangle \\
&= \sup_{\|\phi\|=1} \sum_{Q \in \mathcal{T}_h} \langle f \circ \mathcal{M} - \mathcal{A}u_h, \phi - P_Q(\phi) \rangle_Q \\
&\leq \sum_{Q \in \mathcal{T}_h} \|f \circ \mathcal{M} - \mathcal{A}u_h\|_{0,Q} \sup_{\|\phi\|=1} \|\phi - P_Q(\phi)\|_{0,Q}.
\end{aligned}
$$

Consider now the last supremium within an element $Q \in \mathcal{T}_h$. Consider one spline segment $\Delta(\mathbf{j}) := \prod_{\nu=1}^{d}[\zeta_{j_\nu}^{\nu}, \zeta_{j_\nu+1}^{\nu}] \subset Q$ where $\mathbf{j} = (j_1, ..., j_d)$. Consider also an extension of the spline segment as $\widetilde{\Delta}(\mathbf{j}) := \prod_{\nu=1}^{d}[\zeta_{j_\nu-(k_\nu-1)}^{\nu}, \zeta_{j_\nu+k_\nu}^{\nu}]$. Due to the support property (2.7), for each $\mathbf{t} = (t_1, ..., t_d) \in \Delta(\mathbf{j})$, the basis functions with nonzero coefficients of $P_Q(\phi)(\mathbf{t})$ from (3.28) correspond to $\mathbf{i} = (i_1, ..., i_d)$ such that $i_\nu \in \{j_\nu-(k_\nu-1), ..., j_\nu+k_\nu\}$ for $\nu = 1, ..., d$.

Since the de Boor-Fix quasi-interpolant P_Q keep polynomials unchanged [3], we obtain for every $p \in \mathcal{P}(\widetilde{\Delta}(\mathbf{j}))$ (polynomials on $\widetilde{\Delta}(\mathbf{j})$)

$$(3.30) \qquad \|\phi - P_Q(\phi)\|_{0,\Delta(\mathbf{j})} \leq \sqrt{\mu[\Delta(\mathbf{j})]}\|\phi - P_Q(\phi)\|_{\infty,\Delta(\mathbf{j})}$$

$$(3.31) \qquad = \sqrt{\mu[\Delta(\mathbf{j})]}\|(\phi - p) - P_Q(\phi - p)\|_{\infty,\Delta(\mathbf{j})}$$

$$(3.32) \qquad \leq \sqrt{\mu[\Delta(\mathbf{j})]}(1 + \|P_Q\|)\|\phi - p\|_{\infty,\widetilde{\Delta}(\mathbf{j})}$$

($\mu(X)$ standing for the Lebesgue measure of X). Hence,

$$(3.33) \qquad \|\phi - P_Q(\phi)\|_{0,\Delta(\mathbf{j})} \leq \sqrt{\mu[\Delta(\mathbf{j})]} \inf_{p \in \mathcal{P}(\widetilde{\Delta}(\mathbf{j}))} \|\phi - p\|_{\infty,\widetilde{\Delta}(\mathbf{j})}.$$

In virtue of polynomial approximation of [9], we have

$$(3.34) \qquad \|\phi - p\|_{\infty,\widetilde{\Delta}(\mathbf{j})} \leq C\Big[\operatorname{diam}(\widetilde{\Delta}(\mathbf{j}))\Big]^{2+(d/2)}|\phi|_{2,\widetilde{\Delta}(\mathbf{j})}.$$

As a consequence, one obtains

$$(3.35) \qquad \|\phi - P_Q(\phi)\|_{0,\Delta(\mathbf{j})} \leq \sqrt{\mu[\Delta(\mathbf{j})]}\Big[\operatorname{diam}(\widetilde{\Delta}(\mathbf{j}))\Big]^{2+(d/2)}|\phi|_{2,Q}.$$

Since $\|\phi\| = 1$, we obtain from the definition (2.15) that $h(Q)^2|\phi|_{2,Q}^2 \leq 1$ or equivalently $|\phi|_{2,Q} \leq 1/h(Q)$. As a consequence, we deduce

$$(3.36) \qquad \|\phi - P_Q(\phi)\|_{0,\Delta(\mathbf{j})} \leq \sqrt{\mu[\Delta(\mathbf{j})]}\Big[\operatorname{diam}(\widetilde{\Delta}(\mathbf{j}))\Big]^{2+(d/2)}h(Q)^{-1}.$$

On the other hand, due to the knot quasi-uniformity (3.29), we have

$$(3.37) \qquad \sqrt{\mu[\Delta(\mathbf{j})]} = C(\theta)\prod_{\nu=1}^{d}\left[\frac{h_\nu(Q)}{n_\nu - k_\nu + 2}\right]^{1/2}.$$

We use now the Pythagorean rule and the quasi-uniformity (3.29) again to obtain

$$(3.38) \qquad \operatorname{diam}[\widetilde{\Delta}(\mathbf{j})] = \left[\sum_{\nu=1}^{d}|\zeta_{j_\nu+k_\nu}^{\nu} - \zeta_{j_\nu-k_\nu+1}^{\nu}|^2\right]^{1/2}$$

$$(3.39) \qquad = C(\theta)\left[\sum_{\nu=1}^{d}\left(\frac{k_\nu(Q)h_\nu(Q)}{n_\nu(Q) - k_\nu(Q) + 2}\right)^2\right]^{1/2}.$$

Finally, in order to deduce the theorem, take the sum over $\mathbf{j}$ by noting that there are $\prod_{\nu=1}^{d}\big(n_\nu(Q) - k_\nu(Q) + 2\big)$ spline segments $\Delta(\mathbf{j})$ within each $Q \in \mathcal{T}_h$.
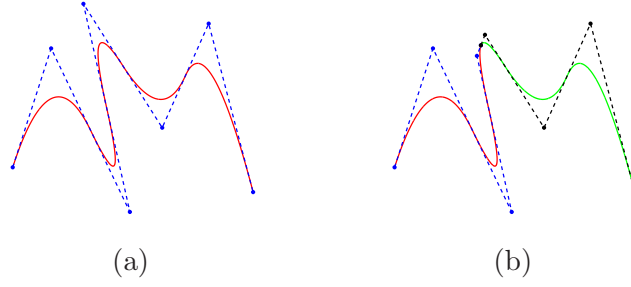
FIGURE 4. (a) Original B-spline (b) Several knot insertions generate two B-splines which have the same parametrization as the original B-spline

☐ ☐

3.3. Cascading from Coarse to Fine Discretizations.

The linear system obtained from (2.16) using the discretization $\mathcal{T}_h$ is solved by a CG (conjugate gradient) with a simple diagonal preconditioner. For the initial guess of the CG, we use *cascading*. That is, the solution from the previous $\mathcal{T}_{h-1}$ is used as a starting value for the CG on $\mathcal{T}_h$. As opposed to the usual Finite Element bases, applying cascading using B-spline basis is not very straightforward. The next description consists of the expression of a function $f \in \mathbb{V}_h$ in terms of the bases of $\mathbb{V}_{h+1} \supset \mathbb{V}_h$ by using *discrete B-splines*.

Consider two knot sequences $\boldsymbol{\zeta} = (\zeta_0, ..., \zeta_{n+k})$ and $\widetilde{\boldsymbol{\zeta}} = (\widetilde{\zeta}_0, ..., \widetilde{\zeta}_{m+k})$ such that $\boldsymbol{\zeta} \subset \widetilde{\boldsymbol{\zeta}}$. We recall [6, 7] the discrete B-splines which enable the expression of a coarse basis $N_i^{k,\boldsymbol{\zeta}}$ as a linear combination of fine bases $N_p^{k,\widetilde{\boldsymbol{\zeta}}}$. Choose $a_i \in [\widetilde{\zeta}_j, \widetilde{\zeta}_{j+k})$ and define

$$(3.40) \qquad \phi_i^k(y) \quad := \quad (y - a_i)_+^0 \, \Psi_i^k(y),$$

$$(3.41) \qquad (y - a_i)_+^0 \quad := \quad 1 \quad \text{if} \quad y > a_i,$$

$$(3.42) \qquad (y - a_i)_+^0 \quad := \quad 0 \quad \text{if} \quad y \leq a_i.$$

where $\Psi_i^k(t) := (t - \widetilde{\zeta}_{i+1}) \cdots (t - \widetilde{\zeta}_{i+k-1})$. One has

$$(3.43) \qquad N_i^{k,\boldsymbol{\zeta}}(x) = \sum_{p=0}^{m} \alpha_i^k(p) N_p^{k,\widetilde{\boldsymbol{\zeta}}}(x)$$

where $\alpha_i^k(p) := (\zeta_{i+k} - \zeta_i)[\zeta_i, \cdots, \zeta_{i+k}]\phi_p^k$.

A fact [6, 7] is that a splitting into two B-splines is equivalent to applying knot insertions several times as shown in Fig. 4. Suppose that a 2D element $Q \in \mathcal{T}_h$ has been bisected vertically as in Fig. 3(a) into Q_1 and Q_2. The spline properties on Q are $(n_1, k_1, \boldsymbol{\zeta}^1(Q))$ and $(n_2, k_2, \boldsymbol{\zeta}^2(Q))$. Let $\widetilde{\boldsymbol{\zeta}}^1(Q)$ be defined by inserting the midpoint $\mu := 0.5(\zeta_0^1 + \zeta_{n_1+k_1}^1)$

in the knot sequence $\boldsymbol{\zeta}^1(Q)$ by k_1 times. We have

$$N_i^{k_1,\boldsymbol{\zeta}^1(Q)} \otimes N_j^{k_2,\boldsymbol{\zeta}^2(Q)} = \sum_{p=0}^{m} \alpha_i^k(p) N_p^{k_1,\widetilde{\boldsymbol{\zeta}}^1(Q)} \otimes N_j^{k_2,\boldsymbol{\zeta}^2(Q)}$$

$$= \sum_{p=0}^{m_1} \alpha_i^k(p) N_p^{k_1,\widetilde{\boldsymbol{\zeta}}^1(Q)} \otimes N_j^{k_2,\boldsymbol{\zeta}^2(Q)} + \sum_{p=m_1}^{n} \alpha_i^k(p) N_p^{k_1,\widetilde{\boldsymbol{\zeta}}^1(Q)} \otimes N_j^{k_2,\boldsymbol{\zeta}^2(Q)}$$

$$= \sum_{p=0}^{m_1} \alpha_i^k(p) N_p^{k_1,\boldsymbol{\zeta}^1(Q_1)} \otimes N_j^{k_2,\boldsymbol{\zeta}^2(Q)} + \sum_{p=0}^{n-m_1} \alpha_i^k(p+m_1) N_p^{k_1,\boldsymbol{\zeta}^1(Q_2)} \otimes N_j^{k_2,\boldsymbol{\zeta}^2(Q)}$$

In the last equality, $\boldsymbol{\zeta}^1(Q_1)$ is the subsequence of $\widetilde{\boldsymbol{\zeta}}^1(Q)$ from the begining until μ while $\boldsymbol{\zeta}^1(Q_2)$ is the remaining subsequence. A similar deduction can be done for the 2D horizontal bisection and for the 3D case.

In practice, the discrete B-splines $\alpha_i^k(p)$ are evaluated by using the recurrence

$$(3.44) \qquad \alpha_i^1(j) \;=\; 1 \qquad \text{if} \quad \widetilde{\zeta}_j \in [\zeta_i, \zeta_{i+j})$$

$$(3.45) \qquad \alpha_i^1(j) \;=\; 0 \qquad \text{if} \quad \widetilde{\zeta}_j \notin [\zeta_i, \zeta_{i+j})$$

$$(3.46) \qquad \alpha_i^k(j) \;=\; (\widetilde{\zeta}_{j+k-1} - \zeta_i)\beta_i^{k-1}(j) + (\zeta_{i+k} - \widetilde{\zeta}_{j+k-1})\beta_{i+1}^{k-1}(j)$$

in which

$$(3.47) \qquad \beta_i^k(j) := \begin{cases} \alpha_i^k(j)/(\zeta_{i+k} - \zeta_i) & \text{for} \quad \zeta_{i+k} > \zeta_i \\ 0 & \text{otherwise.} \end{cases}$$

4. Practical results

In this section, we present some practical results to supplement the previous theory. Let us consider first a 2D adaptive simulation where we consider the next exact solution

$$(4.48) \qquad u(x,y) = \exp\left[-\frac{1}{\alpha}\big((x-a)^2 + (y-b)^2\big)\right]$$

Refin. step	Nb. elements	Ratio estim.	Refin. step	Nb. elements	Ratio estim.
0	9	2.595145	9	34	1.889834
2	12	2.177448	10	37	2.129213
3	16	1.616603	13	46	2.199019
5	23	1.708272	18	61	2.299195
6	25	1.796925	20	67	2.649732

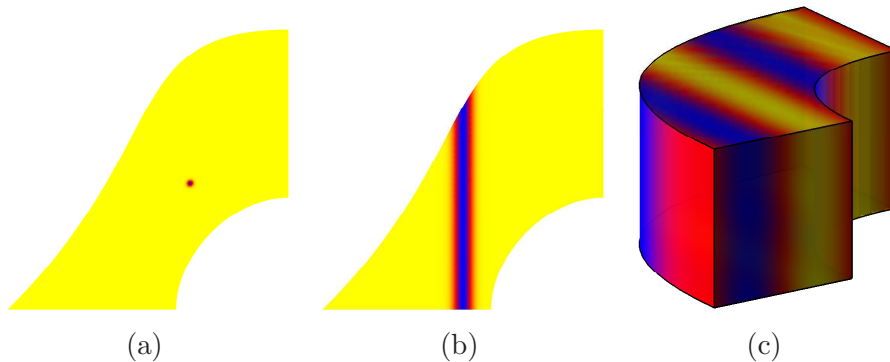TABLE 4.1. Ratio of exact and estimated errors

FIGURE 5. NURBS physical domains and the exact solutions

on the NURBS physical domain shown in Fig. 5(a) . That function takes unit value at $\mathbf{x} = (a, b)$ which is $(-0.35, 0.75)$ in our test. The large dot in mixed blue and red color inside the physical domain indicates the position where the function nearly takes unit values. Note that away from the dot, the function u decays exponentially to zero. The speed of that decay becomes quicker as the value of the parameter α approaches zero. In the depicted case, we have used $\alpha = 1.0e - 04$. In Fig. 6, we collect some adaptive history of a few refinement grids. We start from a very coarse discretization $\mathcal{T}_0$ which is a uniform 3×3 tensor product grid in Fig. 6(a). Then, we refine adaptively according to the a-posteriori error estimator described in Section 3. This example illustrates very clearly the situation where the grid refinement takes place strictly within the domain and therefore there is no need to use a fine mesh at the boundary although we deal with a curved physical domain. A mesh-based approach would necessitate a fine mesh at the boundary because the bounding curves are not straight. Approximating the curved portions of the boundary by PL-curves would require a significant number of points which would substantially increase the degree of freedom.

For the next test, we consider the same NURBS for the physical domain but the exact solution is chosen to be the following

$$(4.49) \qquad u(x, y) = \exp\left[-\frac{1}{\omega}(x + 0.5)^2 \right]$$

This corresponds to an internal layer whose width is specified by ω as shown in Fig. 5(b). As the parameter ω becomes smaller, the layer gets thinner. In our experiment, we chose the parameter value $\omega = 0.001$. As in the former test, we start again from a very coarse tensor product mesh having 3×3 uniform elements. We apply the previously described theory during the adaptive refinements. The results of that process is depicted in Fig. 7 where the a-posteriori error estimator can efficiently detect the position of the internal layer. It is plainly observed that the elements which are far from the layer are very
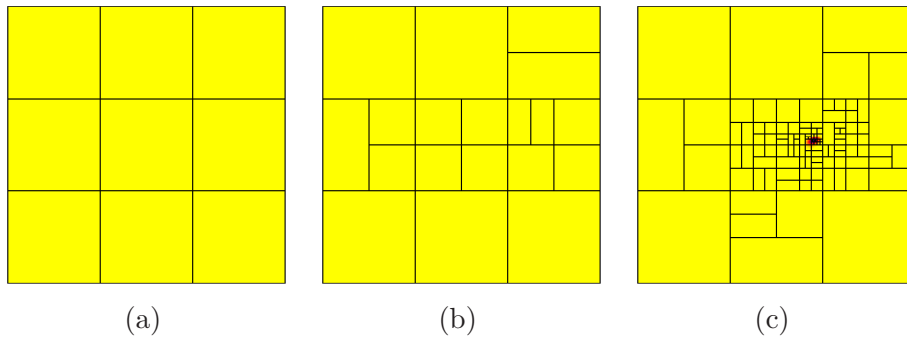
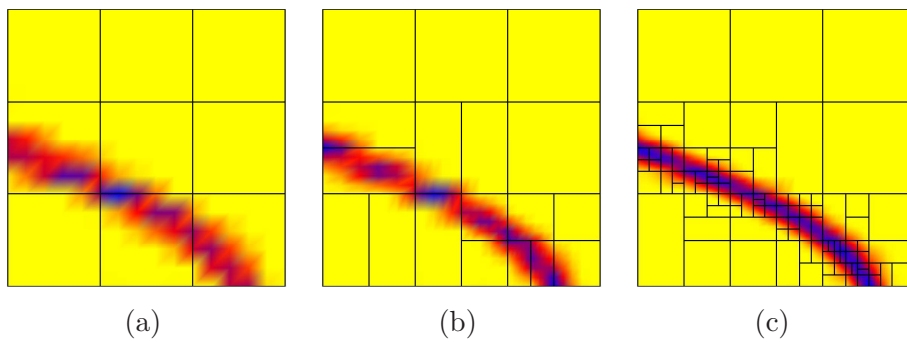FIGURE 6. Adaptive refinement for an internal accumulation.

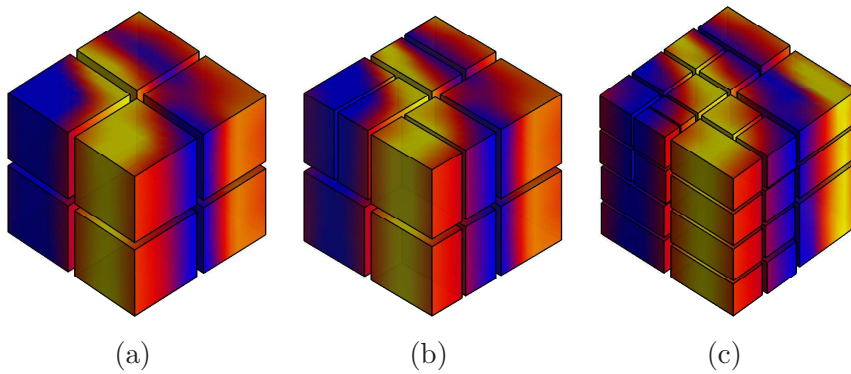FIGURE 7. Local spline adaptivity for a thin internal layer.

FIGURE 8. Adaptive refinement in 3D: the elements are slightly shrunken in order to see the refinements.

coarse. That is for example the case of the top right element which is intact from beginning till the finest discretization shown in Fig. 6(c). In addition, we gather in Table 4.1 the averages of the ratio between the exact error and the a-posteriori error indicator.
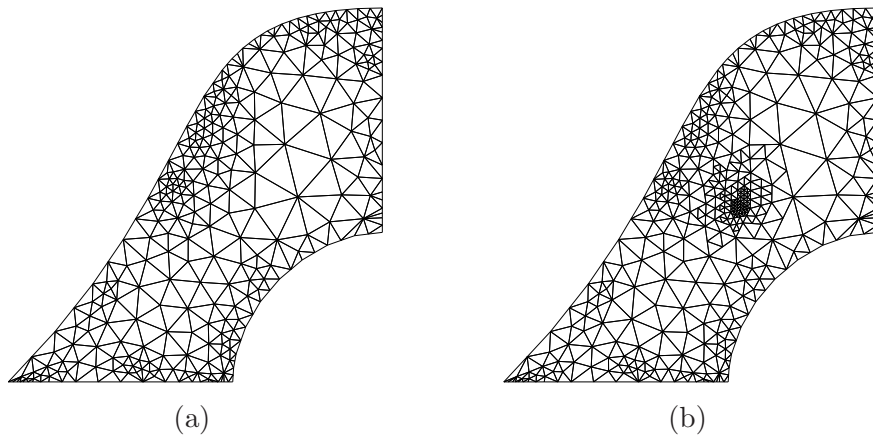
(a)   (b)

FIGURE 9. (a)Initial mesh for the usual DG (b) Adaptive DG-mesh.

In addition to the planar situation, we perform also some practical tests in the 3D case. To that end, we consider the exact solution

$$(4.50) \qquad u(x, y, z) = \sin(\pi x).$$

The 3D NURBS for the physical domain is depicted in Fig. 5(c). The history of some adaptive refinements for this case is summarized in Fig. 8.

REFERENCES

[1] Ainsworth, M.: Review of Adaptive Finite Element Methods for Differential Equations by W. Bangerth and R. Rannacher. SIAM Rev. 46, 354–356 (2004)

[2] Arnold, D., Brezzi, F., Cockburn, B., Marini, L.: Unified Analysis of Discontinuous Galerkin Methods for Elliptic Problems. SIAM J. Numer. Anal. 39, No. 5, 1749–1779 (2002)

[3] De Boor, C., Fix, G.: Spline Approximation by Quasiinterpolants. J. Approx. Theory 8, 19–45 (1973)

[4] Castillo, P., Cockburn, B., Perguis, I., Schötzau, D.: An A-Priori Error Analysis of the Local Discontinuous Galerkin Method for Elliptic Problems. SIAM J. Numer. Anal. 38, No. 5, 1676–1706 (2000)

[5] Cockburn, B., Kanschatr, G., Perguis, I., Schötzau, D.: Superconvergence of the DG Method for Elliptic Problems on Cartesian Grids. SIAM J. Numer. Anal. 39, No. 1, 267–285 (2001)

[6] Cohen, E., Lyche, T., Riesenfeld, R.: Discrete B-Splines and Subdivision Techniques in Computer Aided Geometric Design and Computer Graphics. Comput. Graphics and Image Process. 14, 87–111 (1980)

[7] Cohen, E., Lyche, T., Schumaker, L.: Degree Raising for Splines. J. Approx. Theory 46, 170–181 (1986)

[8] Coons, S.: Surfaces for Computer Aided Design of Space Forms. Technical report, Project MAC, Department of Mechanical Engineering in MIT, Revised to MAC-TR-41 (1967)

[9] Dupont, T., Scott, R.: Polynomial Approximation of Functions in Sobolev Spaces. Math. Comput. 34, No. 150, 441–463 (1980)

[10] Farin, G., Hansford, D.: Discrete Coons Patches. Comput. Aided Geom. Des. 16, No. 7, 691–700 (1999)

[11] Floater, M.: An $\mathcal{O}(h^{2n})$ Hermite Approximation for Conic Sections. Comput. Aided Geom. Design. 14, 135–151 (1997)

[12] Forrest, A.: On Coons and Other Methods for the Representation of Curved Surfaces. Comput. Graph. Img. Process. 1, 341–359 (1972)

[13] Gormaz, R.: A Class of Multivariate de Boor-Fix Formulae. Comput. Aided Geom. Des. 15, 829–842 (1998)

[14] Harbrecht, H., Randrianarivony, M.: From Computer Aided Design to Wavelet BEM. Comput. Vis. Sci. 13, No. 2, 69–82 (2010)

[15] Harbrecht, H., Randrianarivony, M.: Wavelet BEM on Molecular Surfaces: Parametrization and Implementation. Computing 86, 1–22 (2009)

[16] Höllig, K., Reif. U.: Nonuniform WEB-Spline. Comput. Aided Geom. Des. 20, No. 5, 277–294 (2003)

[17] Hughes, T., Cottrell, J., Bazilevs, Y.: Isogeometric Analysis: CAD, Finite Elements, NURBS, Exact Geometry, and Mesh Refinement. Comput. Methods in Appl. Mech. Eng. 194, 4135–4195 (2005)

[18] Randrianarivony, M., Brunnett, G.: Preparation of CAD and Molecular Surfaces for Meshfree Solvers. Lect. Notes Comput. Sci. Eng. 65, 231–245 (2008)

[19] Randrianarivony, M.: Geometric Processing of CAD Data and Meshes as Input of Integral Equation Solvers. PhD thesis, Technische Universität Chemnitz, Germany (2006)

[20] Randrianarivony, M., Brunnett, G.: Molecular Surface Decomposition Using Geometric Techniques. In: Conf. Bildverarbeitung für die Medizine pp. 197–201, Berlin (2008)

[21] Randrianarivony, M.: On Global Continuity of Coons Mappings in Patching CAD Surfaces. Comput.-Aided Design 41, No. 11, 782–791 (2009)

[22] U. S. Product Data Association: Initial Graphics Exchange Specification. IGES 5.3. Trident Research Center, `http://ts.nist.gov/standards/iges`

Maharavo Randrianarivony, Institut für Numerische Simulation, Universität Bonn, Wegelerstr. 6, 53115 Bonn, Germany.

E-mail address: `randrian@ins.uni-bonn.de`