Multiwavelet Boundary Element Method for Cavities Admitting Many NURBS Patches

Maharavo Randrianarivony

Simulation Unit, Personal Simulation and Design, Sankt Augustin, Germany
Email: maharavo@gmx.de

Abstract

We consider the modeling and simulation by means of multiwavelets on many patches. Our focus is on molecular surfaces which are represented in the form of Solvent Excluded Surfaces that are featured by smooth blendings between the constituting atoms. The wavelet bases are constructed on the unit square which maps bijectively onto the patches embedded in the space. The cavity which designates the surface bounding a molecular model is acquired from the nuclei coordinates and the Van-der-Waals radii. We use multi-wavelets for which the wavelet basis functions are organized hierarchically on several levels. Our assembly of the linear system is accomplished by using a hierarchical tree which enables the treatment of large molecules admitting thousands of patches. Along with the patch construction, some wavelet simulation outcomes which are applied to realistic patches are reported.

Keywords

Wavelet, Patch, NURBS, Connolly, Modeling

1. Introduction

Boundary Element Method (BEM) has important applications in ion channels, pH computation, membrane simulations and synthetic medicines. Reduction of dimension is the principal advantage of BEM [1] over the traditional FEM (Finite Element Method) [2] [3] [4] [5] as a 3D problem is reduced to an equation located on a 2D-manifold. That enables the use of a 2D-surface embedded in the space instead of a massive 3D-mesh. That is particularly important in the case that one is only interested in the solution on the surface of a given geometry or in the infinite domain beyond the geometry as frequently occurring in quantum simulations [6] [7]. In addition, the convergence is substantially faster because only a small degree of freedom is sufficient to attain a precise approximation. This article treats the modeling and simulation using

the wavelet Galerkin equation which is formulated on patched manifolds. This current implementation has an advantage of functioning in parallel on a multi-processor computer architecture that accelerates the computations considerably. A major drawback of BEM is that the matrix density requires a large memory capacity when the trial functions are standard polynomial bases whose pertaining linear system necessitates a dense linear solver. Wavelets [8] [9] admit a significant advantage over the traditional polynomial bases because the wavelet technique compresses the BEM matrices efficiently [10] [11]. The surface structure which is required by the wavelet-BEM is unfortunately very complicated to construct in contrast to the standard mesh generation [12]. In the current paper, we consider a twofold objective related to modeling and simulation. First, we will describe the molecular surface generation when a set of atoms is provided. The resulting surface structure is suitable for the wavelet-BEM integral equation solver [13]. Afterward, we will apply a BEM simulation on the resulting models. This paper focuses on the practical aspect of the wavelet method for realistic data. The entire molecular surface needs to be decomposed into four-sided patches. One needs a parametrization which maps the unit square to each four-sided patch. The mapping has to be bijective, sufficiently differentiable with bounded Jacobian. Thus, the decomposition becomes conforming as non-matching curvilinear edges are not allowed. Some pointwise agreement for adjacent mappings is therefore fulfilled. That enables the construction of multi-wavelets which are understood in the sense that the wavelet bases are structured on hierarchical levels [14]. Before proceeding further, we survey some related works in the past. The technique which provides a splitting method for CAD surfaces is proposed in [15] where methods for checking regularity of Coons maps is additionally proved. The main task in [16] is the correlation between the transfinite interpolation [17] which resides in an individual patch and the global continuity. While approximations are required to obtain global continuity in [16] for CAD objects, it can be achieved exactly for molecular surfaces in [18]. That is due to the fact that both circular arcs and spherical patches [19] can be exactly recast as NURBS (Non-Uniform Rational B-Spline) [20]. Furthermore, a real chemical simulation by using wavelet BEM is employed in [7] for the quantum computation. A wavelet BEM simulation using domain decomposition techniques was described in [21] which treats the case of ASM (Additive Schwarz Method). It was utilized as an efficient preconditioner for the wavelet single layer potential which is badly conditioned. Recently, we gained experience [3] in elasticity nanosimulation where one has used nanotube fibers immersed in polymer matrices as quantum models. The organization of this current paper is structured as follows. First, it starts by describing the essential steps for constructing the smooth patch decomposition. We proceed afterwards to the presentation of a hierarchical tree method to compute the wavelet integrals when coupled with the Genz-Malik approach [22] [23]. The matrix entries are usually integrals with 4D integrands which are singular. Toward the end, we present some results pertaining to the patch decomposition in which the inputs are either water clusters obtained from former molecular dynamic simulations or quantum

models obtained from PDB (Protein Data Bank) files. In addition, we report on BEM results including the execution runtimes for the different stages in the BEM simulation. Further, we will briefly display the acceleration of the executions when utilizing computers admitting a multi-processor architecture.

2. Integral Equation on Molecular Surfaces

Our modeling and simulation are performed on a cavity [24] which is acquired from the boundary of molecules where each constituting atom is represented as an imaginary sphere whose center m_k corresponds to the nuclear coordinates and whose radius r_k to a scaled Van-der-Waals radius of the atom. That is, by denoting the sphere of center m and radius r by $B(m,r)$, the molecule is represented as the union of N spheres $\mathcal{A} = \bigcup_{k=1}^{N} B(m_k, r_k)$. We need additionally a probe atom which serves as smoothing of the molecular surface. The SES (Surface Excluded Surface) model [25] which is also known as Connolly surface [26] is the surface Γ traced by the probe atom when it is rolled over (see **Figure 1**) the whole surface $\mathcal{S} := \partial \mathcal{A} = \partial \left[\bigcup_{k=1}^{N} B(m_k, r_k) \right]$. The Connolly surface Γ is extracted partly from $\mathcal{S}$ and partly from the blending surfaces traced by the probe atom. The blending surfaces are composed of surfaces of two types. The first type consists of toroidal surfaces while the second one of trimmed spherical surfaces. Our first objective is to decompose the Connolly surface Γ into M four-sided patches [18] admitting (see **Figure 2(a)**) the following properties:
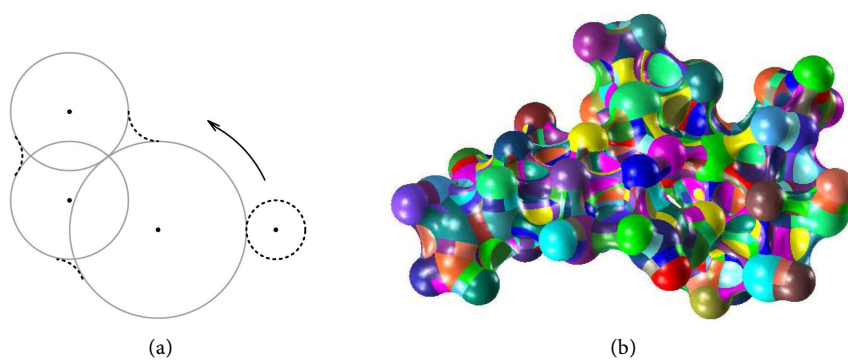

(a)   (b)

Figure 1. (a) Rolling a probe atom on the molecular surface; (b) connolly surface.
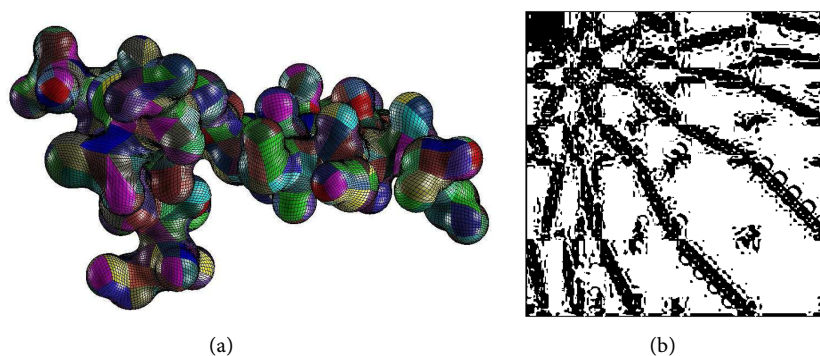

(a)   (b)

Figure 2. (a) Patches for wavelet construction; (b) quasi-sparse matrix.

- We have a covering of the molecular surface by patches $\Gamma = \bigcup_{p=1}^{M} \Gamma_p$,

- Each patch Γ_p where $p = 1, 2, \cdots, M$ is the image by $\gamma_p : \square := [0,1]^2 \to \Gamma_p$ which is described by a bivariate NURBS function that is bijective, sufficiently smooth and admitting bounded Jacobians,

- The intersection of two different patches Γ_p and Γ_q is supposed to be either empty, a common curvilinear edge or a common vertex,

- The patch decomposition has a global continuity: for each pair of patches Γ_p, Γ_q sharing a curvilinear edge, the parametric representation is subject to a matching condition. That is, a bijective affine mapping $\Xi : \square \to \square$ exists such that for all $x = \gamma_p(s)$ on the common curvilinear edge, one has $\gamma_p(s) = (\gamma_q \circ \Xi)(s)$. In other words, the images of the NURBS functions $\boldsymbol{\gamma}_p$ and $\boldsymbol{\gamma}_q$ agree pointwise at common edges after some reorientation,

- The manifold Γ is orientable and the normal vector $n(x)$ is consistently pointing outward for any $x \in \Gamma$.

For each patch Γ_p, the Gram determinant is denoted

$$G_p(t) = G_p(t_1, t_2) := \left\| \frac{\partial \gamma_p}{\partial t_1}(t) \times \frac{\partial \gamma_p}{\partial t_2}(t) \right\| \qquad \forall t = (t_1, t_2) \in \square. \tag{1}$$

The space of square integrable functions is

$$\mathbb{L}^2(\Gamma) := \left\{ f : \Gamma \to \mathbb{R}, \qquad \int_\Gamma |f(x)|^2 \, \mathrm{d}_\Gamma x < \infty \right\} \tag{2}$$

which is equipped with the following scalar product and norm after transformation onto $\square$,

$$
\begin{aligned}
\langle u, v \rangle_{\mathbb{L}^2(\Gamma)} &:= \int_\Gamma u(x) v(x) \mathrm{d}_\Gamma x \\
&= \sum_{p=1}^{M} \int_\square u(\gamma_p(t)) v(\gamma_p(t)) G_p(t) \mathrm{d}t, \qquad \|v\|_{\mathbb{L}^2(\Gamma)} := \left[\langle v, v \rangle_{\mathbb{L}^2(\Gamma)} \right]^{1/2}.
\end{aligned}
\tag{3}
$$

The Sobolev space on Γ for a non-negative integer k is

$$\mathbb{H}^k(\Gamma) := \left\{ f \in \mathbb{L}^2(\Gamma) : \|\partial^\alpha f\|_{\mathbb{L}^2(\Gamma)} < \infty \quad \text{for all} \quad |\alpha| \le k \right\} \tag{4}$$

where the differentiation $\partial^\alpha f$ is interpreted in the sense of distribution [27] such that $\langle \partial^\alpha f, g \rangle_{\mathbb{L}^2(\Gamma)} = (-1)^{|\alpha|} \langle f, \partial^\alpha g \rangle_{\mathbb{L}^2(\Gamma)}$ for all compactly supported smooth functions g. The Sobolev space $\mathbb{H}^k(\Gamma)$ is endowed with the norm

$$\|f\|_{\mathbb{H}^k(\Gamma)}^2 := \sum_{|\alpha| \le k} \|\partial^\alpha f\|_{\mathbb{L}^2(\Gamma)}^2. \tag{5}$$

Concerning a real positive order $p = k + \theta$ such that k is an integer and $\theta \in \,]0,1[$, the Sobolev space $\mathbb{H}^p(\Gamma)$ consists of the functions such that their norms with respect to the next Slobodeckij norm are finite

$$\|f\|_{\mathbb{H}^p(\Gamma)}^2 = \|f\|_{\mathbb{H}^k(\Gamma)}^2 + \sum_{|\alpha|=k} \int_{\Gamma \times \Gamma} \frac{|\partial^\alpha f(x) - \partial^\alpha f(y)|^2}{\|x - y\|^{2+2\theta}} \mathrm{d}_\Gamma x \mathrm{d}_\Gamma y. \tag{6}$$

For negative orders, one employs the dual spaces $\mathbb{H}^{-p}(\Gamma) = \left[\mathbb{H}^p(\Gamma)\right]^*$. By designating the region enclosed within Γ by $\Omega \subset \mathbb{R}^3$, our second objective is to solve the interior problem:

$$\begin{cases} \Delta \mathcal{U}(x) = 0 & \text{for} \quad x \in \Omega \\ \mathcal{U}(x) = g(x) & \text{for} \quad x \in \Gamma = \partial\Omega = \bigcup_{p=1}^{M} \Gamma_p. \end{cases} \tag{7}$$

Introduce the double layer operator

$$(Kv)(x) = \frac{1}{4\pi} \int_\Gamma \frac{\partial}{\partial n(y)} \left[\frac{1}{\|x-y\|}\right] u(y) \, \mathrm{d}_\Gamma y \tag{8}$$

$$= \frac{1}{4\pi} \int_\Gamma \frac{\langle n(y), x-y \rangle}{\|x-y\|^3} u(y) \, \mathrm{d}_\Gamma y. \tag{9}$$

In the general case, we have the following mapping using Sobolev spaces

$$K : \mathbb{H}^{1/2}(\Gamma) \to \mathbb{H}^{1/2}(\Gamma) \tag{10}$$

which is linear and continuous. In the current case, since the entire molecular surface Γ is sufficiently smooth, we have

$$K : \mathbb{H}^{(1/2)+s}(\Gamma) \to \mathbb{H}^{(1/2)+s}(\Gamma) \tag{11}$$

for any real number s according to Theorem 7.1 and Theorem 7.2 of [28]. In particular, we deduce the bounded linearity with respect to square-integrable functions:

$$K : \mathbb{L}^2(\Gamma) \equiv \mathbb{H}^0(\Gamma) \to \mathbb{L}^2(\Gamma) \equiv \mathbb{H}^0(\Gamma). \tag{12}$$

We make now the change of unknown

$$\mathcal{U}(x) = \frac{1}{4\pi} \int_\Gamma \frac{\partial}{\partial n(y)} \left[\frac{1}{\|x-y\|}\right] u(y) \, \mathrm{d}_\Gamma y \tag{13}$$

by using u as the new unknown which is also termed as density function. As a point $x \in \Omega$ approaches $x_0 \in \Gamma = \partial\Omega$, we have [28] [29] [30]

$$\lim_{\substack{x \to x_0 \\ x \in \Omega}} (Ku)(x) = -\frac{1}{2} u(x_0) + (Ku)(x_0) \tag{14}$$

because the manifold Γ contains neither a nonsmooth edge nor a sharp corner. The function $\mathcal{U}$ as mentioned in (13) is harmonic in the sense that $\Delta \mathcal{U} = 0$. Therefore, in the case that u satisfies

$$-\frac{1}{2} u(x_0) + (Ku)(x_0) = g(x_0) \qquad \text{for} \qquad x_0 \in \Gamma, \tag{15}$$

then on account of (14), the function $\mathcal{U} = Ku$ solves the boundary value problem in (7). In operator form, we have the next integral equation after multiplication by (-2):

$$(I - 2K)u = f \qquad \text{where} \qquad f := -2g. \tag{16}$$

By discretizing (16) in some finite dimensional space $\mathbb{V}_h$, one searches for $u_h \in \mathbb{V}_h$ such that for all $v_h \in \mathbb{V}_h$

$$\int_\Gamma u_h(x) v_h(x) \, \mathrm{d}_\Gamma x + \int_\Gamma \int_\Gamma \mathcal{K}(x,y) u_h(x) v_h(x) \, \mathrm{d}_\Gamma x \mathrm{d}_\Gamma y = \int_\Gamma f(x) v_h(x) \, \mathrm{d}_\Gamma x \tag{17}$$

where we use the kernel

$$\mathcal{K}(x,y) := \frac{1}{2\pi} \frac{\langle n(y), y-x \rangle}{\|x-y\|^3}. \tag{18}$$

The subspace $\mathbb{V}_h$ will be piecewise polynomials which could be piecewise constant so that we have in particular $\mathbb{V}_h \subset \mathbb{L}^2(\Gamma)$. Once the solution u to the integral Equation (16) becomes available, the solution $\mathcal{U}$ to the initial boundary value problem (7) is recovered by applying (13). Since both the wavelet basis function and the mappings γ_p are expressed in term of B-spline basis, we shall recall briefly some important properties of a B-spline setting which represents piecewise polynomials. Consider two integers n, k such that $n \geq k \geq 1$. Should the interval $[a,b]$ be the domain of definition of the B-spline, that interval is subdivided by a knot sequence $\zeta = (\zeta_i)_{i=0}^{n+k}$ such that $\zeta_i < \zeta_{i+1}$ for $i = k-1, \cdots, n-1$ and such that the initial and the final entries of the knot sequence are clamped $\zeta_0 = \cdots = \zeta_{k-1} = a$ and $\zeta_n = \cdots = \zeta_{n+k} = b$. One defines the B-splines [31] [32] [33] basis functions as

$$N_i^{\zeta,k}(t) := (\zeta_{i+k} - \zeta_i)[\zeta_i, \cdots, \zeta_{i+k}](\cdot - t)_+^{k-1} \qquad \text{for } i = 0, \cdots, n$$

where one employs the divided difference $[\zeta_i, \zeta_{i+1}, \cdots, \zeta_p] f$ in which one uses the truncated power functions $(\cdot - t)_+^k$ given by $(x-t)_+^k := (x-t)^k$ if $x \geq t$, while it is zero otherwise. The integer k controls the polynomial degree $k-1$ of the B-spline which admits an overall smoothness of $\mathcal{C}^{k-2}$ where the case $k=1$ corresponds to discontinuous piecewise constant functions needed for the Haar wavelet. The integer n controls the number of B-spline functions for which each B-spline basis $N_i^{\zeta,k}$ is supported by $[\zeta_i, \zeta_{i+k}]$. The NURBS patch γ_p admitting the *control points* $d_{i,j} \in \mathbb{R}^3$ and weights $w_{i,j} \in \mathbb{R}^+$ is expressed as

$$\gamma_p(u,v) = \frac{\sum_{i=0}^n \sum_{j=0}^n w_{i,j} d_{i,j} N_i^{\zeta,k}(u) N_j^{\zeta,k}(v)}{\sum_{i=0}^n \sum_{j=0}^m w_{i,j} N_i^{\zeta,k}(u) N_j^{\zeta,k}(v)} \in \mathbb{R}^3 \qquad \forall (u,v) \in \square. \tag{19}$$

As for the construction of the finite dimensional space $\mathbb{V}_h$, since the Gram determinant G_p is bounded

$$0 < c \leq \min_{p=1,\cdots,M} \inf_{t \in \square} G_p(t) < \max_{p=1,\cdots,M} \sup_{t \in \square} G_p(t) \leq C < \infty, \tag{20}$$

the above structure of the surface Γ allows to construct bases on 2D which are carried over the manifold Γ that is embedded in $\mathbb{R}^3$. The Galerkin variational formulation with respect to a finite dimensional space spanned by $(\varphi_\alpha)_{\alpha=1}^m$ uses the approximating functions $u_h(x) = \sum_{\alpha=1}^m u_\alpha \varphi_\alpha(x)$ where $\mathbb{U}^T := [u_1, \cdots, u_m] \in \mathbb{R}^m$ are the BEM-unknowns. The next linear system is eventually obtained

$$(\mathbb{I} + \mathbb{K})\mathbb{U} = \mathbb{S} \tag{21}$$

such that the matrix entries and the right hand side are

$$\mathbb{I}_{(\alpha,\beta)} := \int_\Gamma \varphi_\alpha(x) \varphi_\beta(x) d_\Gamma x = \sum_{p=1}^M \int_{\Gamma_p} \varphi_\alpha(x) \varphi_\beta(x) d_{\Gamma_p} x \tag{22}$$

$$\mathbb{K}_{(\alpha,\beta)} := \sum_{p=1}^{M}\sum_{q=1}^{M}\int_{\Gamma_p}\int_{\Gamma_q}\mathcal{K}(x,y)\varphi_\alpha(x)\varphi_\beta(y)\mathrm{d}_{\Gamma_p}x\mathrm{d}_{\Gamma_q}y \tag{23}$$

$$\mathbb{S}_\alpha := \sum_{p=1}^{M}\int_{\Gamma_p}f(x)\varphi_\alpha(x)\mathrm{d}_{\Gamma_p}x. \tag{24}$$

In general, the linear system (21) is troublesome because the basis functions $(\varphi_\alpha)_{\alpha=1}^{m}$ yield a dense matrix for the operator $\mathbb{K}$. In term of memory, if $(\varphi_\alpha)_{\alpha=1}^{m}$ are standard polynomial basis functions, the matrix $\mathbb{K}$ requires m^2 storage to accommodate all entries. In addition, the determination of a matrix entry of $\mathbb{K}$ calculates an integration in $4D$ where the integrand is highly nonlinear and possibly singular depending on the patch pair $\Gamma_p \times \Gamma_q$. By using tensor product B-spline wavelet basis functions, the matrix $\mathbb{K}$ becomes quasi-sparse. For the instance of high-order odd wavelets [34] [35], we depict on **Figure 2(b)** the quasi-sparse matrix entries for a $2D$ singular operator where very small entries are set to zero.

3. Patch Generation for Wavelet Bases

Let us specify the assumptions concerning the positions of the nuclei m_i and the properties of the radii r_i with respect to the probe radius ρ. For every two arbitrary atoms $B(m_i, r_i)$ and $B(m_j, r_j)$, we assume that one of the following two conditions holds.

(C1) Either the two enlarged spheres $B(m_i, r_i + \rho)$ and $B(m_j, r_j + \rho)$ by the probe radius ρ are completely disjoint such as $\|m_i - m_j\| > r_i + r_j + 2\rho$,

(C2) or we have $D_{ij} := \|m_i - m_j\| \le r_i + r_j + 2\rho$ and additionally

$$2D_{ij}r_i^2 + 4D_{ij}\rho - D_{ij}^2 - (r_i + \rho)^2 + (r_j + \rho)^2 > 0.$$

Those two assumptions exclude the situation where the blending torus which is tangent to both $B(m_i, r_i)$ and $B(m_j, r_j)$ admits a self-intersection. If assumptions (C1) and (C2) are not fulfilled but one still wants to treat the molecule, one carefully inserts additional dummy atoms between every two atoms $B(m_i, r_i)$ and $B(m_j, r_j)$ for which there is a toroidal self-intersection. The sizes and the positions of the dummy atoms should be minimized while keeping the shape of the initial molecule. We will consider the generation of the constituents of the B-Rep (boundary representation) of the SES surfaces. For two spheres $\mathcal{B}_1 := B(m_1, r_1)$ and $\mathcal{B}_2 := B(m_2, r_2)$, we define their power distance as $D(\mathcal{B}_1, \mathcal{B}_2) := \|m_1 - m_2\|^2 - r_1^2 - r_2^2$. The i-th Laguerre cell is composed of points which are closer to $\mathcal{B}_i$ than to any other $\mathcal{B}_j (j \ne i)$ with respect to the power distance: $C_i := \{x \in \mathbb{R}^3 : D(B(x,0), \mathcal{B}_i) \le D(B(x,0), \mathcal{B}_j) \ \forall j \ne i\}$. A Laguerre decomposition of $\mathbb{R}^3$ with respect to the spheres $\mathcal{B}_i$ is $\mathbb{R}^3 = \bigcup_{i=1}^{N}C_i$. If all radii are equal, the Laguerre decomposition coincides with the usual Voronoi decomposition as illustrated in **Figure 3(a)**. For two spheres $\mathcal{B}_i$ and $\mathcal{B}_j$, the radical axis is the set of points which are equidistant to $\mathcal{B}_i$ and $\mathcal{B}_j$ with respect to the power distance. Such a set is a plane given by
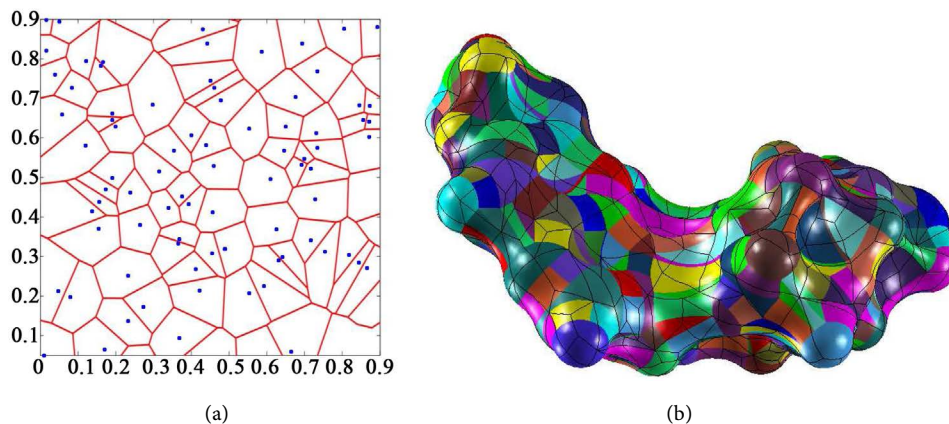
Figure 3. (a) Centered convex decomposition of the space; (b) curvilinear edges traced on the manifold.

$$R\left(\mathcal{B}_i, \mathcal{B}_j\right) = \left\{ x \in \mathbb{R}^3 : 2\left\langle x, m_i - m_j \right\rangle = \left\| m_i \right\|^2 - \left\| m_j \right\|^2 + r_j^2 - r_i^2 \right\}$$

The Laguerre cell is a convex polyhedron which has faces from the radical axes and which could be bounded or unbounded. To obtain the Laguerre decomposition, we consider the uplifting function which associates to $m_i = \left(x_i, y_i, z_i\right) \in \mathbb{R}^3$ the value $\tilde{m}_i := \left(x_i, y_i, z_i, x_i^2 + y_i^2 + z_i^2 + r_i^2\right) \in \mathbb{R}^4$. After generating the convex hull $\mathcal{H}$ of the set of four dimensional points $\{\tilde{m}_i\}$, the projection of the lower face of $\mathcal{H}$ on the space $\mathbb{R}^3$ generates a weighted Delaunay tetrahedral decomposition [36] having apices m_i. Each apex of the Laguerre cell is the orthocenter of a tetrahedron of the weighted Delaunay. The Laguerre decomposition is obtained by the dual of the weighted Delaunay. We describe next the way of obtaining the spherical trimmed surfaces of each atom $\mathcal{B}_m$ by considering its cell C_m. For each neighboring cell C_j of C_m, consider the cell planar face $\mathcal{P}_{mj}$ which separates the spheres $B\left(m_m, r_m\right)$ and $B\left(m_j, r_j\right)$. One generates two offset planes $\boldsymbol{p}_m$ and $\boldsymbol{p}_j$ by orthogonally shifting $\mathcal{P}_{mj}$ by $d_m := D_{mj}\rho/\left(r_m + \rho\right)$ and $d_j := D_{mj}\rho/\left(r_j + \rho\right)$ toward those two spheres respectively. Two circles $\boldsymbol{c}_m$ and $\boldsymbol{c}_j$ are traced on those spheres by $\boldsymbol{p}_m$ and $\boldsymbol{p}_j$. We discard the circular arcs on $\mathcal{B}_m$ which are either included inside an atom other than $\mathcal{B}_m$ or between the planes $\boldsymbol{p}_m$, $\boldsymbol{p}_j$. By organizing the remaining circular arcs, we obtain several closed curves on the sphere $\mathcal{B}_m$. These bounding curves yield several spherical trimmed surfaces on the sphere $\mathcal{B}_m$. Each face $\mathcal{P}_{mj}$ of the Laguerre decomposition corresponds to one torus T_{mj} which is tangent upon $\mathcal{B}_m$ and $\mathcal{B}_j$. To obtain the toroidal surface, we trim off the toroidal part of T_{mj} which is beyond the parallel planes $\boldsymbol{p}_m$, $\boldsymbol{p}_j$. If the rolling probe atom touches at least three atoms, we construct a spherical blend whose boundary is composed of the arcs which are the intersections of the toroidal surface T_{mj} and the probe atom. At this point, we have subsurfaces S_i which are bounded by some circular arcs. Stereographic projections serve as parametrizations of the subsurfaces S_i from some trimmed planar domains.

We describe now the decomposition of a Connolly surface into large four-sided subsurfaces. We use a triangular mesh $\mathcal{M}$ of the whole surface as an intermediate step. We determine first the underlying structure of a coarse quadrangulation $\mathcal{Q}_{\text{coarse}}$. We deduce the nodes, edges and quadrilaterals of $\mathcal{Q}_{\text{coarse}}$ from the mesh. We start from a fine quadrangulation $\mathcal{Q}_{\text{fine}}$ which is obtained by subdividing each triangular element of $\mathcal{M}$ into three quadrilaterals. The resulting quadrangulation $\mathcal{Q}_{\text{fine}}$ is not directly useful for patch representation because it is too fine. As a consequence, we need to coarsen that fine quadrangulation repeatedly in which we initialize $\mathcal{Q}_0 := \mathcal{Q}_{\text{fine}}$. Each coarsening step from $\mathcal{Q}_m$ to $\mathcal{Q}_{m+1}$ consists in amalgamating a few neighboring quadrilaterals in the quadrangulation $\mathcal{Q}_m$ to form a coarser local simplified quadrangulation in $\mathcal{Q}_{m+1}$. It is a long programming to implement many local patterns of simplification and to recursively determine the parts of $\mathcal{Q}_m$ to coarsen so that the final patches have similar surface areas. Since the initial finest quadrangulation $\mathcal{Q}_{\text{fine}}$ is inappropriate to be used as $\mathcal{Q}_{\text{coarse}}$, we impose minimal and maximal numbers of coarse quadrilaterals. For the recursive quadrangulation simplification, we use a coarsening parameter $\alpha \in [0,1]$ which gauges the density of $\mathcal{Q}_{\text{coarse}}$. A unit value of α corresponds to the finest allowed quadrilateral decomposition while a value of α approaching zero corresponds to a very coarse quadrilateral decomposition. After assembling $\mathcal{Q}_{\text{coarse}}$ having straight edges, we need to connect every two nodes P_i and P_j on the endpoints of every edge by a curve on the manifold as illustrated in Figure 3(b). That curve is described by means of a geodesic curve on $\mathcal{M}$ joining P_i to P_j. First, we need a curve which traverses only the nodes and the edges of $\mathcal{M}$. Afterward, we improve that curve by allowing it to traverse the internal parts of the triangles of $\mathcal{M}$. Assembling the complete node-edge graph of the whole mesh $\mathcal{M}$ is very memory consuming. Searching for shortest paths becomes very cumbersome for such a large discrete manifold. Some efficient data structure to accelerate the search is necessary in the implementation. We eventually fit each four-sided submeshes by NURBS patches where each interface curve has a matching parametrization by its incident patches. During the application of those geometric operations, tests must be performed in order to avoid manifold folding and tiny gaps. Many tests related to edge degeneration and angular quality are needed to be applied in practice.

In order to ensure the validity of conditions (C1) and (C2), one adopts the method of GEPOL in which some additional spheres are inserted [37] [38]. Those incorporated extra-spheres are not physically relevant. These are fictive spheres admitting neither atomic masses nor electric charges. The principal objective of the previously mentioned conditions is to avoid the conflicting situation where some toroidal blending surfaces occur when the corresponding two disjoint atoms are distant from one another so that the contact with the probe atom is almost tangential. In such a case, a self-intersecting toroidal blend is formed. In order to remedy that problem, a fictive non-atom centered sphere is inserted between every two atoms where those conditions are violated. It is worth noting anyhow that such a situation occurs only in very rare cases. We would like to explain the similarity and the difference between the above two types of smoothing surfaces which are the toroidal ones and the spherical ones. Both surfaces

serve as blending surfaces between neighboring atoms. The principal difference between them is that the toroidal ones occur when only two atoms are touched by the probe atom, whereas a spherical blend appears when more than two atoms simultaneously have contact with the probe atom as illustrated in **Figure 1(a)** and **Figure 1(b)**. In the latter case, the number of atoms which touch the probe atom is three in most practical cases but that number can theoretically be arbitrarily many depending on the positions and the radii of the constituting atoms.

Now that we have a four-sided decomposition, we want to construct the wavelet spaces on the surface Γ. Since every two incident patches admit pointwise joints, constructing the wavelet bases on the unit square $\square$ is sufficient to construct basis functions on the whole molecular surface. Each $1D$-wavelet basis (see **Figure 4(a)**) will be constructed as a linear combination of B-splines

$$\psi(t) = \sum_{i=0}^{n} q_i N_i^{\zeta,k}(t) \in \mathbb{R} \qquad \text{where} \qquad \forall t \in [0,1]. \tag{25}$$

On level ℓ, we define a knot sequence $\zeta^\ell = \left(\zeta_i^\ell\right) \subset [0,1]$ such that $\zeta_i^\ell = 0$ and $\zeta_{n+1+i}^\ell = 1$ for $i = 0, \cdots, k$ and that the remaining $\zeta_i^\ell \in \,]0,1[$. The internal knots on the next level $(\ell+1)$ are obtained by inserting one new knot inside two consecutive knots on the lower level ℓ. Introduce the B-spline linear space on level ℓ:

$$\mathbb{V}_\ell^k[0,1] := \left\{ N_i^\ell := N_i^{\zeta^\ell,k} \in \mathbb{L}_2[0,1]: \quad i = 0, \cdots, n \right\}. \tag{26}$$

By using the piecewise polynomial property of the B-splines and the inclusion $\zeta^\ell \subset \zeta^{\ell+1}$, the B-spline bases form a nested sequence of subspaces:

$$\mathbb{V}_1^k[0,1] \subset \mathbb{V}_2^k[0,1] \subset \cdots \subset \mathbb{V}_L^k[0,1] \subset \mathbb{L}_2[0,1]. \tag{27}$$

As a consequence, the space $\mathbb{V}_\ell^k[0,1]$ can be expressed as an orthogonal sum

$$\mathbb{V}_\ell^k[0,1] = \mathbb{V}_{\ell-1}^k[0,1] \oplus \mathbb{W}_{\ell-1}^k[0,1] \tag{28}$$

with respect to the $\mathbb{L}_2$-scalar product where $\mathbb{W}_\ell^k[0,1]$ is the wavelet space

$$\mathbb{W}_{\ell-1}^k[0,1] = \text{span}\left\{ \psi_i^{\ell-1} \in \mathbb{V}_\ell^k[0,1], \quad \left\langle \psi_i^{\ell-1}, \phi \right\rangle_{\mathbb{L}^2[0,1]} = 0, \quad \forall \phi \in \mathbb{V}_{\ell-1}^k[0,1] \right\}. \tag{29}$$

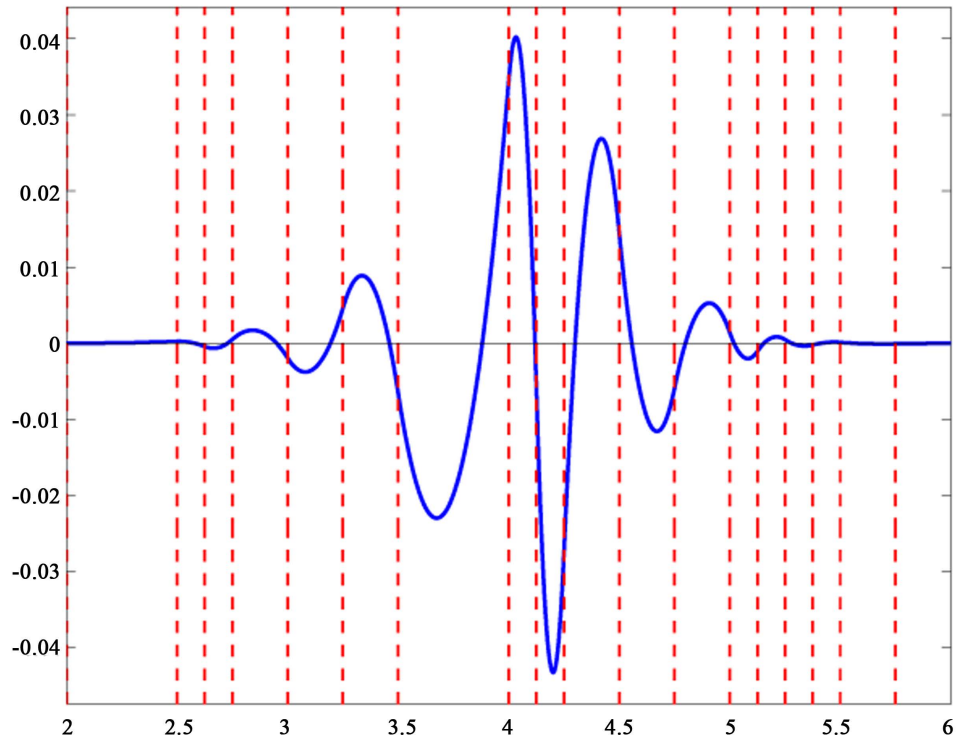By applying the decomposition (28) recursively, one obtains on the maximal level L

$$\mathbb{V}_L^k[0,1] = \mathbb{V}_1^k[0,1] \oplus \left(\bigoplus_{\ell=1}^{L-1} \mathbb{W}_\ell^k[0,1] \right). \tag{30}$$

The $2D$-wavelet spaces (see **Figure 4(b)**) on the unit square $\square$ is defined for any maximal level L as follows.
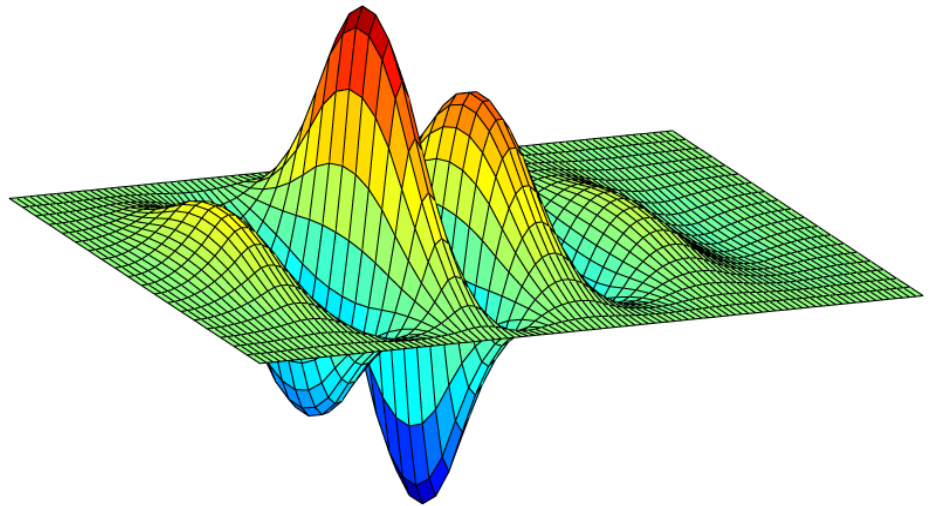
$$\mathbb{V}_L^k(\square) := \mathbb{V}_L^k[0,1] \otimes \mathbb{V}_L^k[0,1] = \mathbb{A}^k(\square) \oplus \mathbb{B}_L^k(\square) \oplus \mathbb{C}_L^k(\square) \oplus \mathbb{D}_L^k(\square) \tag{31}$$

such that

$$\mathbb{A}^k(\square) := \mathbb{V}_1^k[0,1] \otimes \mathbb{V}_1^k[0,1], \qquad \mathbb{B}_L^k(\square) := \bigoplus_{\ell=1}^{L-1}\left(\mathbb{W}_\ell^k[0,1] \otimes \mathbb{V}_1^k[0,1] \right),$$

$$\mathbb{C}_L^k(\square) := \bigoplus_{\ell=1}^{L-1}\left(\mathbb{V}_1^k[0,1] \otimes \mathbb{W}_\ell^k[0,1] \right), \qquad \mathbb{D}_L^k(\square) := \bigoplus_{\ell=1}^{L-1}\bigoplus_{m=1}^{L-1}\left(\mathbb{W}_\ell^k[0,1] \otimes \mathbb{W}_m^k[0,1] \right).$$

Figure 4. (a) Higher-order wavelet on a nonuniform knot sequence, (b) bivariate tensor product basis.

We want now to survey the determination of the wavelet bases ψ_i^{ℓ} of $\mathbb{W}_{\ell}^{k}[0,1]$. The construction of the higher-order wavelets [34] [35] requires the case on the whole infinite real line $\mathbb{R}$ on which one has knot entries on each integer as $\zeta_i := i$. The cardinal B-spline is given by

$$N_i^{\text{CARDINAL}}(x) = \int_0^1 N_{i-1}^{\text{CARDINAL}}(x-t)\,\mathrm{d}t \tag{32}$$

which verifies the two scale relation

$$N_i^{\text{CARDINAL}}(x) = \sum_{j=0}^{i} 2^{-i+1} \binom{j}{i} N_i^{\text{CARDINAL}}(2x - j).$$ (33)

For the polynomial degree k, the corresponding complementary space is spanned by the shifts of the wavelet function

$$\psi_{\text{CARDINAL}}(x) := \frac{1}{2^{k-1}} \sum_{j=0}^{2m-2} (-1)^j N_{2k}^{\text{CARDINAL}}(j+1) \frac{d^k}{dx_k} N_{2k}^{\text{CARDINAL}}(2x - j).$$ (34)

Representing the derivatives $d^k N_{2k}^{\text{CARDINAL}}(2x - j)/dx_k$ in function of B-splines can be used to express the function ψ_{CARDINAL} in term of control points. The cardinal wavelets are orthogonal to B-splines on the infinite real line $\mathbb{R}$ having integers as knot sequence. The cardinal splines serve as construction of internal wavelets on the interval $[0,1]$ by scaling and shifting. Only the odd wavelets for odd polynomial degree k is described because the even case is defined almost similarly with the exception of additional central wavelets. One defines on level ℓ the clamped knot sequence $\zeta_0^\ell = \cdots = \zeta_{k-1}^\ell = 0$, and $\zeta_{n_\ell}^\ell = \cdots = \zeta_{n_\ell+k}^\ell = 1$ where $n_\ell := 2^\ell - 1$. The dependence on ℓ is sometimes dropped to simplify the notations. The $1D$-wavelets use the following internal knot sequence $\zeta^{(\ell)}$

$$\zeta_j^\ell := (\lfloor k/2 \rfloor + j) 2^{-\ell} \in \,]0,1[, \qquad \text{for} \qquad j = k, \cdots, n_\ell$$ (35)

The *internal* wavelet functions are defined by means of scaled and dilated transformations

$$\psi_j^\ell(t) := \psi_{\text{CARDINAL}}(2^\ell t - (j - k + 1)).$$ (36)

Additionally, one needs *boundary* wavelet functions which are of the form

$$\psi_j^\ell(t) = \sum_{p=0}^{k+\lfloor k2 \rfloor - 2} q_{p,\ell} N_p^{\ell+1}(t) + \sum_{p=k+\lfloor k2 \rfloor - 1}^{k+\lfloor k2 \rfloor + 2\ell} q_{p,\ell} N_p^{\ell+1}(t)$$ (37)

where the coefficients $q_{p,\ell}$ for the last summation are the control points of the cardinal wavelet ψ_{CARDINAL}. the control points of the cardinal wavelet ψ_{CARDINAL}. The remaining unknown coefficients $q_{p,\ell}$ on the first summation are obtained by solving

$$\sum_{p=0}^{k+\lfloor k2 \rfloor - 2} \langle N_p^{\ell+1}, N_m^\ell \rangle_{\mathbb{L}_2[0,1]} q_{p,\ell} = - \sum_{p=k+\lfloor k2 \rfloor - 1}^{k+\lfloor k2 \rfloor + 2\ell} q_{p,\ell} \langle N_p^{\ell+1}(t), N_m^\ell \rangle_{\mathbb{L}_2[0,1]},$$ (38)

or $m = 0, \cdots, k + \lfloor k2 \rfloor - 2$. That ensures the $\mathbb{L}_2$-orthogonality

$$\langle \psi_j^\ell, N_m^\ell \rangle_{\mathbb{L}_2[0,1]} = 0, \qquad \forall N_m^\ell \in \mathbb{V}_\ell^k[0,1].$$ (39)

The previous construction creates $2^{\ell-1}$ wavelet basis functions. To complete the construction, the remaining ones are deduced by symmetry such as $\psi_j^\ell(t) := \psi_{2^\ell - j - 1}^\ell(1-t)$ for $t \in [0,1]$, $j = 2^{\ell-1}, \cdots, 2^\ell - 1$. The construction of the Haar wavelets and piecewise linear wavelets is very similar. The former corresponds to piecewise constant polynomials while the latter to piecewise polynomials of degree unity. Since every polynomial of degree $(k-1)$ can be expressed in term of N_i^ℓ, we obtain from the orthogonality (39) the property of vanishing moments:

$$\int_0^1 (t - \tilde{t})^p \psi_j^\ell(t) \mathrm{d}t = 0, \qquad \text{for} \qquad 0 \le p \le k - 1 \tag{40}$$

for any fixed $\tilde{t}$. It is an important property because it yields quasi-sparse structure of the matrix $\mathbb{K}$. By using the constructed wavelet basis, we are interested in the values

$$\mathbb{K}_{(\alpha,\beta)} = \sum_{p=1}^M \sum_{q=1}^M \int_{\square \times \square} \tilde{\mathcal{K}}_{p,q}(u,v) \psi_\alpha(u) \psi_\beta(v) \mathrm{d}u \mathrm{d}v, \tag{41}$$

$$\tilde{\mathcal{K}}_{p,q}(u,v) := \mathcal{K}\big(\gamma_p(u), \gamma_q(v)\big) G_p(u) G_q(v). \tag{42}$$

We will drop the indices of $\tilde{\mathcal{K}}_{p,q}$ since we consider the integrals in the above summation for a fixed pair $\Gamma_p \times \Gamma_q$. By fixing some $(\tilde{\boldsymbol{u}}, \tilde{\boldsymbol{v}}) = \big[(\tilde{u}_1, \tilde{u}_2), (\tilde{v}_1, \tilde{v}_2)\big] \in \square \times \square$ and by considering any $(\boldsymbol{u}, \boldsymbol{v}) = \big[(u_1, u_2), (v_1, v_2)\big] \in \square \times \square$, the Taylor expansion yields

$$\tilde{\mathcal{K}}(\boldsymbol{u}, \boldsymbol{v}) = \sum_{|\alpha| + |\beta| \le k-1} \frac{1}{\boldsymbol{\alpha}!} \frac{1}{\boldsymbol{\beta}!} \frac{\partial^{|\alpha|}}{\partial \boldsymbol{u}^\alpha} \frac{\partial^{|\beta|}}{\partial \boldsymbol{v}^\beta} \tilde{\mathcal{K}}(\tilde{\boldsymbol{u}}, \tilde{\boldsymbol{v}})(\boldsymbol{u} - \tilde{\boldsymbol{u}})^\alpha (\boldsymbol{v} - \tilde{\boldsymbol{v}})^\beta \tag{43}$$

$$+ \sum_{|\alpha| + |\beta| = k} R_{\alpha,\beta}^{p,q}(\boldsymbol{u}, \boldsymbol{v})(\boldsymbol{u} - \tilde{\boldsymbol{u}})^\alpha (\boldsymbol{v} - \tilde{\boldsymbol{v}})^\beta. \tag{44}$$

For the first summation, by multiplication with a tensor product wavelet basis $\psi^p \otimes \psi^q = (\psi_1^p \otimes \psi_2^p) \otimes (\psi_1^q \otimes \psi_2^q)$ and by taking the integration over $\square \times \square$, one obtains for $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)$ and $\boldsymbol{\beta} = (\beta_1, \beta_2)$ where $|\alpha| = |\alpha_1| + |\alpha_2| \le k - 1$ and $|\beta| = |\beta_1| + |\beta_2| \le k - 1$

$$\frac{1}{\boldsymbol{\alpha}!} \frac{1}{\boldsymbol{\beta}!} \frac{\partial^{|\alpha|}}{\partial \boldsymbol{u}^\alpha} \frac{\partial^{|\beta|}}{\partial \boldsymbol{v}^\beta} \tilde{\mathcal{K}}(\tilde{\boldsymbol{u}}, \tilde{\boldsymbol{v}}) \int_0^1 (u_1 - \tilde{u}_1)^{\alpha_1} \psi_1^p(u_1) \mathrm{d}u_1 \cdots \int_0^1 (v_2 - \tilde{v}_2)^{\beta_2} \psi_2^q(v_2) \mathrm{d}v_2 \tag{45}$$

which is zero due to the property of the vanishing moment (40). As for the second summation, introduce $\eta_r := \max_{i=1,2} \big\{ |u_i - \tilde{u}_i| \ \text{where} \ (u_1, u_2) \in \text{Support}(\psi^r) \big\}$ for $r = p, q$. The summation is estimated by $\sum_{|\alpha| + |\beta| = k} |R_{\alpha,\beta}(\boldsymbol{u}, \boldsymbol{v})| \max_{\boldsymbol{u} \in \square} |\psi^p(\boldsymbol{u})| \max_{\boldsymbol{v} \in \square} |\psi^q(\boldsymbol{v})| \eta_p^{|\alpha|} \eta_q^{|\beta|}$. By defining $\eta := \max\{\eta_p, \eta_q\} < 1$, one obtains

$$\eta^k \sum_{|\alpha| + |\beta| = k} \max_{(\boldsymbol{u}, \boldsymbol{v}) \in \square \times \square} |R_{\alpha,\beta}^{p,q}(\boldsymbol{u}, \boldsymbol{v})| \max_{\boldsymbol{u} \in \square} |\psi^p(\boldsymbol{u})| \max_{\boldsymbol{v} \in \square} |\psi^q(\boldsymbol{v})| \tag{46}$$

Due to boundedness (20), one obtains for $\boldsymbol{x} = \gamma_p(\boldsymbol{u}) \in \gamma_p \big[\text{Support}(\psi^p) \big]$ and $\boldsymbol{y} = \gamma_q(\boldsymbol{v}) \in \gamma_q \big[\text{Support}(\psi^q) \big]$ on account of the Calderon-Zygmund inequality in the case $|\alpha| + |\beta| = k$:

$$|R_{\alpha,\beta}^{p,q}(\boldsymbol{u}, \boldsymbol{v})| \le C \left| \frac{\partial^{|\alpha|}}{\partial \boldsymbol{x}^\alpha} \frac{\partial^{|\beta|}}{\partial \boldsymbol{y}^\beta} \mathcal{K}(\boldsymbol{x}, \boldsymbol{y}) \right| \le C \frac{1}{\|\boldsymbol{x} - \boldsymbol{y}\|^{k+1}} \tag{47}$$

which is small if the images $\gamma_p \big[\text{Support}(\psi^p) \big]$ and $\gamma_q \big[\text{Support}(\psi^q) \big]$ are sufficiently distant from one another. Thus, the speed of the decay toward zero depends on the factor η^k using the vanishing moments exponent k and the distances between the basis supports. Hence, the constructed wavelet basis has the advantage that it renders the operator $\mathbb{K}$ quasi-sparse.

4. Wavelet Hierarchy and Integration

This section is occupied by the efficient computation of the matrix entries from (41). Since our method is based on a hierarchical tree on the B-spline knots, we describe next the procedure of inserting new knots into existing ones. The principal objective is to be able to efficiently express a function defined on the coarse knot sequence in term of B-splines on a fine one. Consider two knot sequences $\zeta^\ell = \left(\zeta_0^\ell, \cdots, \zeta_{n_\ell+k}^\ell \right)$ and $\zeta^{\ell+1} = \left(\zeta_0^{\ell+1}, \cdots, \zeta_{n_{\ell+1}+k}^{\ell+1} \right)$ such that $\zeta^\ell \subset \zeta^{\ell+1}$. For both knot sequences, the smoothness index k is conserved intact. One introduces

$$\phi_i^k(y) := (y - a_i)_+^0 \, \Psi_i^k(y), \quad \text{for some} \quad a_i \in \left[\zeta_j^{\ell+1}, \zeta_{j+k}^{\ell+1} \right)$$

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

$$(y - a_i)_+^0 := 0 \quad \text{if} \quad y \le a_i,$$

where $\Psi_i^k(t) := \left(t - \zeta_{i+1}^{\ell+1} \right) \cdots \left(t - \zeta_{i+k-1}^{\ell+1} \right)$. The *discrete B-splines* are defined as

$$\alpha_i^k(r) := \left(\zeta_{i+k}^\ell - \zeta_i^\ell \right) \left[\zeta_i^\ell, \cdots, \zeta_{i+k}^\ell \right] \phi_r^k \tag{48}$$

which enables to express the basis in lower level in term of those in higher level such as

$$N_i^{\zeta^\ell, k}(x) = \sum_{r=0}^m \alpha_i^k(r) N_r^{\zeta^{\ell+1}, k}(x).$$

The De-Boor algorithm for the evaluation of the discrete B-splines $\alpha_i^k(p)$ utilizes the recurrence

$$\alpha_i^1(j) = 1 \quad \text{if} \quad \zeta_j^{\ell+1} \in \left[\zeta_i^\ell, \zeta_{i+j}^\ell \right)$$

$$\alpha_i^1(j) = 0 \quad \text{if} \quad \zeta_j^{\ell+1} \notin \left[\zeta_i^\ell, \zeta_{i+j}^\ell \right)$$

$$\alpha_i^k(j) = \left(\zeta_{j+k-1}^{\ell+1} - \zeta_i^\ell \right) \beta_i^{k-1}(j) + \left(\zeta_{i+k}^\ell - \zeta_{j+k-1}^{\ell+1} \right) \beta_{i+1}^{k-1}(j)$$

in which $\beta_i^k(j) := \alpha_i^k(j) / \left(\zeta_{i+k}^\ell - \zeta_i^\ell \right)$ for $\zeta_{i+k}^\ell > \zeta_i^\ell$ while $\beta_i^k(j) = 0$ otherwise. We will write interchangeably $\zeta_i^\ell \equiv \zeta^\ell[i]$. In addition, for a knot entry ζ_i^ℓ on level ℓ, we denote by $p(i)$ the index on level $\ell+1$ such that

$$\zeta^\ell[i] = \zeta^{\ell+1}\left[p(i) \right]. \tag{49}$$

Since one inserts one $(\ell+1)$-knot entry between two consecutive ℓ-knot entries, one has in particular

$$\zeta^\ell[i+1] = \zeta^{\ell+1}\left[p(i)+2 \right], \quad \text{thus} \quad p(i+1) = p(i)+2. \tag{50}$$

We introduce the the following 4*D-voxels*

$$\square_{[i_1, i_2, i_3, i_4]}^{[\ell_1, \ell_2, \ell_3, \ell_4]} := \left[\zeta_{i_1}^{\ell_1}, \zeta_{i_1+1}^{\ell_1} \right] \times \left[\zeta_{i_2}^{\ell_2}, \zeta_{i_2+1}^{\ell_2} \right] \times \left[\zeta_{i_3}^{\ell_3}, \zeta_{i_3+1}^{\ell_3} \right] \times \left[\zeta_{i_4}^{\ell_4}, \zeta_{i_4+1}^{\ell_4} \right]. \tag{51}$$

We will use the next multi-indices for the wavelet and B-spline bases

$$\psi_{[i_1, i_2, i_3, i_4]}^{[\ell_1, \ell_2, \ell_3, \ell_4]} := \psi_{i_1}^{\ell_1} \otimes \psi_{i_2}^{\ell_2} \otimes \psi_{i_3}^{\ell_3} \otimes \psi_{i_4}^{\ell_4},$$

$$N_{[i_1, i_2, i_3, i_4]}^{[\ell_1, \ell_2, \ell_3, \ell_4]} := N_{i_1}^{\ell_1} \otimes N_{i_2}^{\ell_2} \otimes N_{i_3}^{\ell_3} \otimes N_{i_4}^{\ell_4}. \tag{52}$$

Since the tensor product wavelet bases verify

$$\psi_{[i_1,i_2,i_3,i_4]}^{[\ell_1,\ell_2,\ell_3,\ell_4]} = \sum_{(j_1,j_2,j_3,j_4)} q_{(j_1,j_2,j_3,j_4)} N_{[j_1,j_2,j_3,j_4]}^{[\ell_1,\ell_2,\ell_3,\ell_4]}, \tag{53}$$

it suffices to compute

$$\int_{\square_{[i_1,i_2,i_3,i_4]}^{[\ell_1,\ell_2,\ell_3,\ell_4]}} \tilde{\mathcal{K}}(t) N_{[j_1,j_2,j_3,j_4]}^{[\ell_1,\ell_2,\ell_3,\ell_4]}(t)\mathrm{d}t.$$

For the Haar wavelets where the corresponding B-spline are piecewise constant, we have four recurrences:

$$\int_{\square_{[i_1,i_2,i_3,i_4]}^{[\ell_1,\ell_2,\ell_3,\ell_4]}} \tilde{\mathcal{K}}(t)\mathrm{d}t = \int_{\square_{[p(i_1),i_2,i_3,i_4]}^{[\ell_1+1,\ell_2,\ell_3,\ell_4]}} \tilde{\mathcal{K}}(t)\mathrm{d}t + \int_{\square_{[p(i_1)+1,i_2,i_3,i_4]}^{[\ell_1+1,\ell_2,\ell_3,\ell_4]}} \tilde{\mathcal{K}}(t)\mathrm{d}t. \tag{54}$$

$$= \cdots \tag{55}$$

$$= \int_{\square_{[i_1,i_2,i_3,p(i_4)]}^{[\ell_1,\ell_2,\ell_3,\ell_4+1]}} \tilde{\mathcal{K}}(t)\mathrm{d}t + \int_{\square_{[i_1+1,i_2,i_3,p(i_4)+1]}^{[\ell_1,\ell_2,\ell_3,\ell_4+1]}} \tilde{\mathcal{K}}(t)\mathrm{d}t \tag{56}$$

For wavelets admitting higher polynomial degrees, by using the discrete B-splines one also obtains four recurrences including:

$$\int_{\square_{[i_1,i_2,i_3,i_4]}^{[\ell_1,\ell_2,\ell_3,\ell_4]}} \tilde{\mathcal{K}}(t) N_{[j_1,j_2,j_3,j_4]}^{[\ell_1,\ell_2,\ell_3,\ell_4]}(t)\mathrm{d}t$$

$$= \sum_{r=0}^{m} \alpha_{j_1}^{k}(r) \left\{ \int_{\square_{[p(i_1),i_2,i_3,i_4]}^{[\ell_1+1,\ell_2,\ell_3,\ell_4]}} \tilde{\mathcal{K}}(t) N_{[r,j_2,j_3,j_4]}^{[\ell_1+1,\ell_2,\ell_3,\ell_4]}(t)\mathrm{d}t \right.$$

$$\left. + \int_{\square_{[p(i_1)+1,i_2,i_3,i_4]}^{[\ell_1+1,\ell_2,\ell_3,\ell_4]}} \tilde{\mathcal{K}}(t) N_{[r,j_2,j_3,j_4]}^{[\ell_1+1,\ell_2,\ell_3,\ell_4]}(t)\mathrm{d}t \right\}.$$

Our objective is to avoid computing any involved integrals several times. Therefore, we construct a hierarchical tree whose nodes correspond to the 4D-voxels $\square_{[i_1,i_2,i_3,i_4]}^{[\ell_1,\ell_2,\ell_3,\ell_4]}$. We define the depth of the multilevel $\ell = (\ell_1,\ell_2,\ell_3,\ell_4)$ having 4D-indices to be $\Delta(\ell) = \ell_1 + \ell_2 + \ell_3 + \ell_4$. Each tree-node has offsprings $\square_{[p(i_1),i_2,i_3,i_4]}^{[\ell_1+1,\ell_2,\ell_3,\ell_4]}, \square_{[p(i_1)+1,i_2,i_3,i_4]}^{[\ell_1+1,\ell_2,\ell_3,\ell_4]}, \cdots, \square_{[i_1,i_2,i_3,p(i_4)]}^{[\ell_1,\ell_2,\ell_3,\ell_4+1]}, \square_{[i_1,i_2,i_3,p(i_4)+1]}^{[\ell_1,\ell_2,\ell_3,\ell_4+1]}$. By using the above recurrences, one traverses the hierarchical tree bottom-up in order to obtain the integrals on all relevant 4D-voxels $\square_{[i_1,i_2,i_3,i_4]}^{[\ell_1,\ell_2,\ell_3,\ell_4]}$. Thus, the wavelet integrals are computed hierarchically without doing any repeated computations. That is, one starts from the voxels having the largest depth $\Delta_{\max}(\ell)$. The worst case of the depth of the hierarchical tree is $4L$ depending on the wavelet indices chosen in the quasi-sparse structure. For the current paper, we use the Genz-Malik integration [22] to compute the integrals corresponding to the leaves of the hierarchical tree. In a forthcoming paper, we will present a more efficient method for computing the leaf integrals. The integration on the voxels $\square_{[i_1,i_2,i_3,i_4]}^{[\ell_1,\ell_2,\ell_3,\ell_4]}$ are computed by transforming it onto $[-1,1]^4$. The Genz-Malik method evaluates the integral $\int_{-1}^{1}\int_{-1}^{1}\cdots\int_{-1}^{1} F(x_1, x_2, \cdots, x_D)\mathrm{d}x_1\mathrm{d}x_2\cdots\mathrm{d}x_D$ of a multivariate function $F:[-1,1]^D \to \mathbb{R}$ by utilizing the 7-point rule

$$\mathcal{SP}[F] = w_1 F(0,\cdots,0) + w_2 \sum_{p=\pm 1}\sum_{i=1}^{D}(I_i F)(p\lambda_2) + w_3 \sum_{p=\pm 1}\sum_{i=1}^{D}(I_i F)(p\lambda_3)$$

$$+ w_4 \sum_{p=\pm 1}\sum_{i<j}^{D}(J_{i,j}F)(p\lambda_4) + w_4 \sum_{p=\pm 1}F(p\lambda_5,\cdots,p\lambda_5),$$

$$(I_i F)(p\lambda) := F(0,\cdots,0,p\lambda,0,\cdots,0),$$

$$(J_{i,j}F)(p\lambda) := F(0,\cdots,0,p\lambda,0,\cdots,0,p\lambda,0,\cdots,0).$$

The nonzero entry (*resp.* entries) of $I_i F$ is (*resp.* $J_{ij}F$) at the i-th position (*resp.* i-th and j-th positions). In the $4D$ cases, every application of $\mathcal{SP}[F]$ requires 57 function evaluations. For the determination of the parameters in $\mathcal{SP}$, one solves a nonlinear system [22] involving the functions

$$\mathcal{N}_1(w_1,\cdots,w_5) = w_1 + 2Dw_2 + 2Dw_3 + 2D(D-1)w_4 + 2^D w_5$$

$$\mathcal{N}_2(w_1,\cdots,w_5) = 2\lambda_2^2 w_2 + 2\lambda_3^2 w_3 + 4(D-1)\lambda_4^2 w_4 + 2^D \lambda_5^2 w_5$$

$$\vdots$$

$$\mathcal{N}_7(w_1,\cdots,w_5) = 2^D \lambda_5^6 w_5 + 2^D/27.$$

The solution to that nonlinear system gives exactly $\lambda_2^2 = 9/70$, $\lambda_3^2 = \lambda_4^2 = 9/10$, $\lambda_5^2 = 9/19$ while the weights are $w_1 = 2^D(12824 - 9120D + 400D^2)/19683$, $w_2 = 2^D(980/6561)$, $w_3 = 2^D(1820 - 400D)/19683$, $w_4 = 2^D(200 - 19683)$, $w_5 = 6859/19683$. That integrator is combined with some adaptive subdivision in which D-dimensional hypercubes that produce large integration errors are split into smaller ones. That subdivision starts from $[-1,1]^D$ as the initial hypercube. During the recursive subdivision, the hypercubes to be split have to be identified. In order to evaluate the local error inside a hypercube, an error estimator is achieved by the next 5-point rule

$$\mathcal{FP}[F] = v_1 F(0,\cdots,0) + v_2 \sum_{p=\pm 1}\sum_{i=1}^{D}(I_i F)(p\lambda_2) + v_3 \sum_{p=\pm 1}\sum_{i=1}^{D}(I_i F)(p\lambda_3)$$

$$+ v_4 \sum_{p=\pm 1}\sum_{i<j}^{D}(J_{i,j}F)(p\lambda_4)$$

where the weights are $v_1 = 2^D(729 - 950D + 50D^2)/729$, $v_2 = 2^D(245/486)$, $v_3 = 2^D(265 - 100D)/1458$, $v_4 = 2^D(25/729)$. In order not to evaluate the integrand too many times, the integration points for $\mathcal{FP}$ are taken from the 7-point rule $\mathcal{SP}$. The error estimator is the difference between $\mathcal{SP}$ and $\mathcal{FP}$. As input, one accepts the maximal number of integrand evaluations together with the desired accuracy. We apply the above numerical quadrature to $\tilde{\mathcal{K}}(u,v)$ in which we avoid zero division by replacing the kernel involving $1/\|x - y\|$ by $1/\sqrt{\|x - y\|^2 + \delta}$ for $x \in \Gamma_p$, $y \in \Gamma_q$ where $\delta > 0$ is an adjustable small parameter. A similar approach using $\delta > 0$ has been used in [39] for the purpose of quantum mechanics using high-dimensional approximation.

5. Wavelet Simulation on Molecular Models

We want now to present some results of the former method which was implemented in C functions together with C++ classes. The visualization has been implemented with the help of OpenGL. The computations have been executed on a computer possessing a 4.1 GHz processor and 32 GB RAM. The C-packet cubature [23] serves as computation of the Genz-Malik integrations. We employ different sorts of quantum models including water clusters which are obtained from a former MD (Molecular Dynamics) simulation. A water cluster is generally a collection of many water molecules H_2O which take their positions after a Molecular Dynamic simulation. Each water molecule H_2O is interpreted as one single particle during the Molecular Dynamic steps. When the MD energy becomes stable as the sum of the kinetic energy and the potential energy is supposed to be conserved, the water molecules form a cluster from which a few particles that are located within a prescribed large sphere are extracted at equilibrium state. The hydrogen and oxygen atoms contained in that large sphere constitute the components of the water clusters. As a consequence, the water cluster forms in general a Connolly surface which takes the shape of a large ball. The radius of the given large ball controls the final size of the water cluster. Besides, we use also other molecules which are acquired from PDB files.

We would like to provide some numerical results pertaining to the runtimes of the cavity generation and the quality of the patches. The runtimes depend on several factors. First, it is affected by the number of atoms in the molecule. Second, it varies in relation with the atom distributions. Further, it depends on how coarse the patch decomposition should be. As a first numerical test, we investigate the number of patches in accordance to the size of the molecule. The results of such a test are gathered in **Table 1** where we examine the coarsening Q_{coarse} from Section 3. According to our experience, the interesting practical values of α range between 0.2 and 0.4. A smaller value of α indicates that one has a coarser decomposition which amounts also to a fewer number of fitting tasks. On the other hand, a single large NURBS surface area needs many sampling points which mean also that it takes more time to complete the NURBS determination. Hence, the running time depends not only on the initial mole-

Table 1. Number of patches with respect to the number of atoms and the coarseness factor α.

Molecules	Nb. atoms	Number of patches (Runtime in second)		
		$\alpha = 0.2$	$\alpha = 0.3$	$\alpha = 0.4$
Benzene	12	84 (13.86 s)	97 (14.06 s)	119 (14.60 s)
Cyclohexane	18	99 (15.27 s)	108 (16.27 s)	127 (16.13 s)
LDS	49	286 (50.36 s)	346 (53.73 s)	409 (52.76 s)
Streptomycin	81	399 (68.69 s)	468 (67.78 s)	564 (74.02 s)
Lecithin	128	687 (118.50 s)	821 (120.68 s)	932 (145.11 s)
PDMPG	225	1102 (219.23 s)	1345 (219.12 s)	1556 (226.34 s)
DNA	116	1610 (461.67 s)	1899 (475.33 s)	2232 (487.38 s)

cular size but also on the surface area of the cavity. For example, lecithin has more atoms than DNA but the latter takes almost four times longer than the former.

The purpose of the second test is the investigation of the area $\mu(\Gamma_i)$ of each patch Γ_i. We would like to compare the areas of the patches in two perspectives: in the vicinity of each patch and global comparison. For the vicinity test, let $\mathcal{N}_i$ denote the set of neighboring patches which share at least one corner node with a patch Γ_i. We computed $M(i) := \mu(\Gamma_i) \Big/ \left[\frac{1}{m_i} \sum_{j \in \mathcal{N}_i} \mu(\Gamma_j) \right]$ where $m_i = \text{Card}(\mathcal{N}_i)$. The results for the average values of $M(i)$ are collected in the third column of **Table 2** where we observe that the sizes of the patches vary slowly because $M(i)$ approximates averagely the unity. As for the global comparison, the ideal patch area μ_{ideal} is the area of the whole molecular surface divided by the number of patches. We compute the ratio $R(i) := \min\left\{ \mu(\Gamma_i) / \mu_{\text{ideal}}, \mu_{\text{ideal}} / \mu(\Gamma_i) \right\} \in [0,1]$. The resulting average values of $R(i)$ are exhibited in the last column of **Table 2**. We observe that the sizes of the patches are not too different from the ideal size. In fact, we have in general $0.65 \mu_{\text{ideal}} \leq \mu(\Gamma_i) \leq 1.34 \mu_{\text{ideal}}$. Those patch properties enable the subsequent wavelet bases to be proportionally distributed. An instance of the patch decomposition for the case of a DNA section is depicted on **Figure 5**.
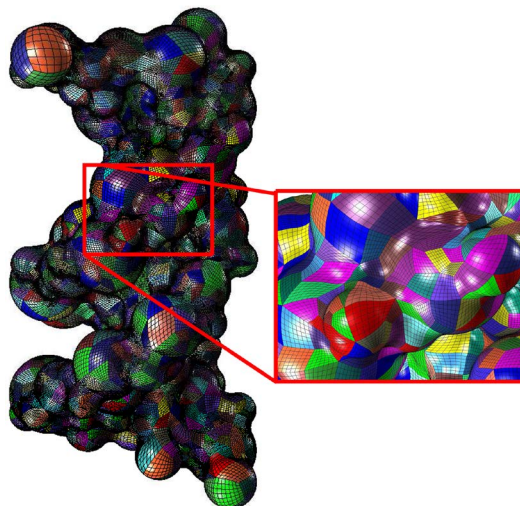

Figure 5. Patch representation of a DNA section with 1905 NURBS.

Table 2. Investigation of patches area.

Molecules	Number of patches	area/(neighb area)	Global ratio
Benzene	12	0.960095	0.706800
Cyclohexane	97	0.959639	0.695809
Tamoxifen	410	1.012924	0.665099
Streptomycin	462	0.959130	0.662762
Lecithin	821	0.971143	0.676250
PDMPG	2175	1.015307	0.688102
DNA	3144	0.950066	0.651584

We focus now on investigating the patch quality which is quantified as follows for a patch Γ_m on level L. For each $i, j = 0, \cdots, 2^L - 1$, we consider the quadrilateral $Q_m(i, j)$ having vertices $a_m(i, j) := \gamma_m(i/2^L, j/2^L)$, $a_m(i+1, j)$, $a_m(i+1, j+1)$, $a_m(i, j+1)$. By inserting a diagonal $\left[a_m(i, j+1), a_m(i+1, j)\right]$ in $Q_m(i, j)$, one defines two triangles T_1 and T_2. Two other triangles T_3 and T_4 are defined by inserting the diagonal $\left[a_m(i, j), a_m(i+1, j+1)\right]$. The quality metric

$\mathcal{L}(Q_m(i, j)) := 0.5\left[\min(\theta(T_1), \theta(T_2)) + \min(\theta(T_3), \theta(T_4))\right]$ is used where $\theta(T_p)$ is the smallest angle in the triangle T_p. The ideal quadrilateral which is a perfect square corresponds to $\mathcal{L}_{\text{ideal}} = 0.25\pi \simeq 0.785398$. In **Table 3**, we gather the results of our test which consists in computing the average values of $\mathcal{L}$ over the whole patches. We obtain a satisfactory quality because the average values of $\mathcal{L}$ do not tend to zero. We would like now to describe some results related to BEM simulation on realistic molecular patches. For the computer implementation, we use only Haar wavelets but the former method can also be applied to higher-order wavelets. We examine first the error reductions in term of the multiscale level for large molecules. The results are collected in **Table 4** which contains both the absolute errors and the relative errors. The

Table 3. Quality of the resulting patches.

Molecules	Number of patches	$\mathcal{L}$-value	
		Average	Ratio with $\mathcal{L}_{\text{ideal}}$
Benzene	133	0.595737	0.758516
Quinine	358	0.554006	0.705383
Borane	812	0.531560	0.676803
Lecithin	821	0.555027	0.706682
Water cluster	1567	0.571700	0.727911
DNA	3348	0.576848	0.734465

Table 4. Error reductions for water cluster and DNA section.

	Solution $\mathcal{U}_1(x)$			Solution $\mathcal{U}_2(x)$		
	Absol. Error	Relat. Error	Reduc.	Absol. Error	Relat. Error	Reduc.
	Water cluster (1109 *patches*):					
Lev 1	2.415E−01	8.322E−03	-	1.186E+00	6.752E−03	-
Lev 2	1.327E−01	4.572E−03	1.820	4.984E−01	2.837E−03	2.211
Lev 3	2.456E−02	8.464E−04	5.404	1.347E−01	7.670E−04	3.698
Lev 4	2.430E−03	8.374E−05	10.10	2.989E−02	1.702E−04	4.506
	DNA section (2119 *patches*):					
Lev 1	2.081E+00	1.309E−01	-	3.853E−01	4.216E−03	-
Lev 2	1.218E+00	7.666E−02	1.707	1.942E−01	2.125E−03	1.984
Lev 3	4.025E−01	2.532E−02	3.027	4.003E−02	4.380E−04	4.851
Lev 4	4.829E−02	3.038E−03	8.334	1.230E−02	1.346E−04	3.254

models consist of a water cluster and a DNA section which admit respectively 1109 patches and 2119 patches. We consider two exact solutions which are respectively $\mathcal{U}_1(x) = .2x^2 - .15y^2 - .05z^2$ and $\mathcal{U}_2(x) = \exp(.5x)\cos(.5y)$ that have vanishing Laplacian. The right hand side $g(x)$ is the restriction of the function $\mathcal{U}$ on the boundary Γ. The error reduction is affected by the exact solutions but in general the errors reduce satisfactorily in function of the wavelet levels. The absolute errors are obtained by comparing with the exact solution at some fixed samples in the interior Ω. A division by the largest value of the exact solution provides the relative error.

We exhibit in **Table 5** the runtimes for the different stages which are required for a complete BEM simulation. These include: (1) the assembly of the identity operator $\mathbb{I}$, (2) the determination of the indices which yield quasi-sparse matrix entries, (3) the computation of the singular integrals in the operator $\mathbb{K}$, (4) the solving of the resulting linear system. In addition, **Table 5** exhibits the required total runtimes for two water clusters which contain respectively 386 patches and 1109 patches on several multiscale levels. It turns out that the construction of the identity operator executes very quickly. The task which mainly dominates the BEM-simulation is the assembly of the singular operator. Although the determination of the quasi-sparse indices and the solving of the linear system take some time, they do not last as long as the singular operator. A GMRes method serves as a solver of the system which is nonsymmetric. Since the time required for the linear solver is very short compared to the assembly of the matrices, it is not yet our priority to improve the linear solver. We examine now the general *linear characteristic* of the relative errors. In **Figure 6**, we display the error evolution where we consider four levels for several molecules and several right hand sides. The curves $e1$ and $e5$ correspond to a propane molecule admitting 75 patches. We use a water cluster admitting 386 patches for $e2$ and $e6$ while another water cluster admitting 1109 patches are used for $e3$ and $e7$. The curve $e7$ corresponds

Table 5. Durations of the individual steps for the water clusters.

|	Identity	Quasisparse	Singularity	Linear oper.	Total	Accuracy
Water cluster (386 *patches*):						
Lev 1	0.28 sc	0.09 sc	15.22 sc	1.44 sc	17.03 sc	1.127E−02
Lev 2	0.32 sc	0.23 sc	27.09 sc	2.20 sc	29.84 sc	1.684E−03
Lev 3	0.45 sc	2.68 sc	32.88 sc	6.39 sc	42.40 sc	4.163E−04
Lev 4	1.34 sc	58.41 sc	8.88 mn	25.93 sc	10.30 mn	9.162E−05
Lev 5	6.84 sc	29.13 mn	144.94 mn	3.29 mn	177.48 mn	3.482E−05
Water cluster (1109 *patches*):						
Lev 1	1.28 sc	0.61 sc	44.56 sc	10.22 sc	56.67 sc	8.322e−03
Lev 2	1.35 sc	1.91 sc	92.24 sc	15.16 sc	110.66 sc	4.572e−03
Lev 3	1.68 sc	24.03 sc	3.79 mn	49.16 sc	5.04 mn	8.464e−04
Lev 4	4.17 sc	9.07 mn	71.23 mn	3.84 mn	84.22 mn	8.374e−05
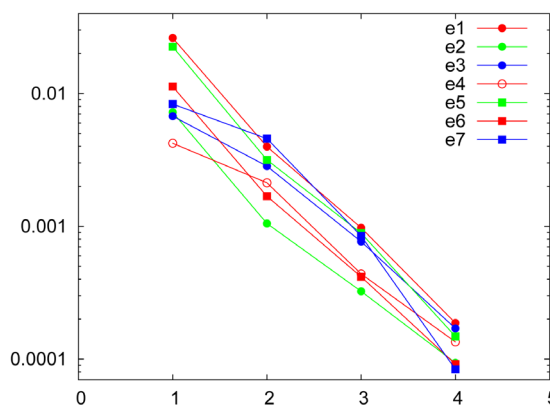
Figure 6. Relative errors for several molecules w.r.t. increasing levels.

to a DNA section admitting 2119 patches. The exact solution $\mathcal{U}_2$ is used for the curves $e1$ till $e4$ while the remaining ones correspond to the solution $\mathcal{U}_1$. The error plots lightly vary in function of the used molecules but in general all the curves exhibit the same slope characteristic. In fact, they decrease linearly in logarithmic scale in function of the BEM levels. The implemented BEM simulation is able to treat large molecules. Our program functions in parallel [40] [41] [42] on computers admitting several cores or multi-processors. The implementation was accomplished by using a message passing technique which was practically carried out by means of MPI (Message Passing Interface). The acceleration of the computation which is also known as speedup is summarized in **Figure 7**. The speedup quantifies the ratio between the execution time on a parallel machine having p processors and the time elapsed on a single processor. The speedup curves indicate that our implementation scales well by increasing the number of processors implying that the load of computing tasks is well balanced among the processors and that the cost of inter-processor communications is not excessive. This consists of the BEM program without the linear solving which takes no more than 5 percent of the entire BEM computation (see **Table 5**) and which functions only sequentially for the time being. All other BEM stages function on a parallel architecture. The speedup curves concern the execution time when the number of the processors is increased for two cases: (a) various molecules possessing different patch counts, (b) increasing wavelet levels. It is beyond the scope of this paper to provide a full description of the parallel implementation which is deferred to a subsequent article. An instance of the computed density function defined on the surface of a DNA section is depicted in **Figure 8** where a triangular mesh serves only as simplification of the graphical presentation.

6. Conclusion

We constructed a surface structure which is suitable for subsequent BEM-simulation. The molecular boundary in form of Connolly surface was decomposed into globally continuous NURBS having similar surface areas. We showed outcomes of BEM simulation on different models possessing many patches. Results pertaining to accuracy
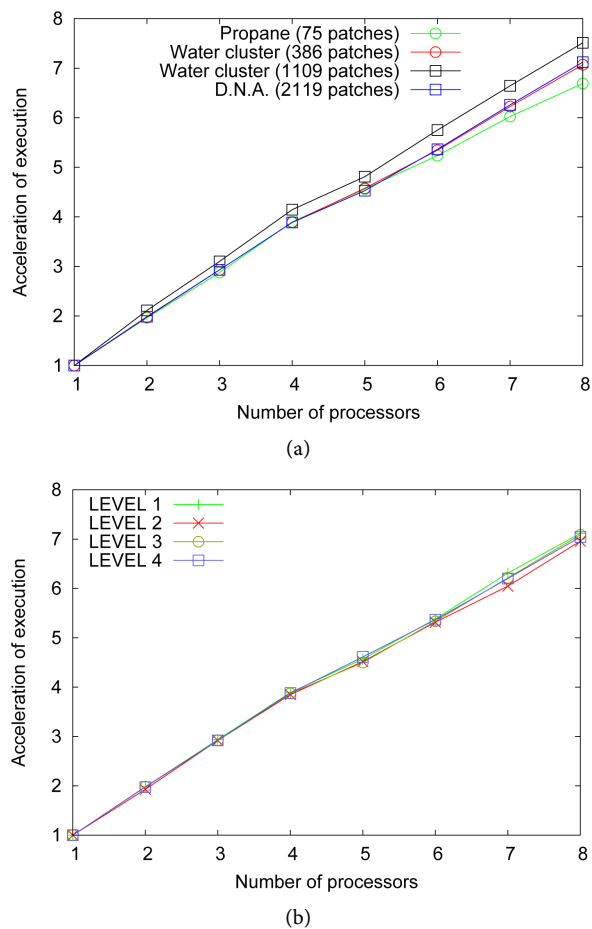
(a)

(b)

Figure 7. Acceleration of the computation in term of the number of processors: (a) Molecules having different patch numbers; (b) several levels.
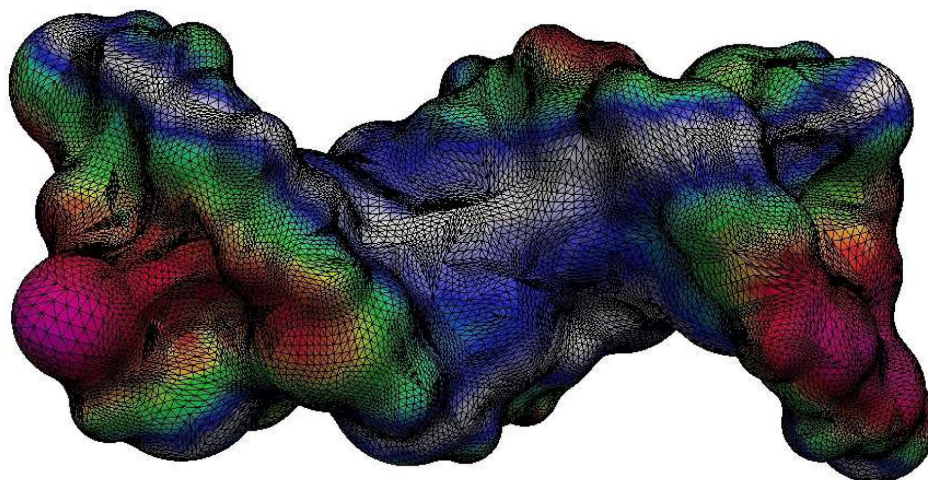

Figure 8. BEM-Density function on the molecular surface of a DNA section.

and runtimes have been reported as well. We treated several molecular models which are acquired from a molecular dynamic simulation or from realistic PDB files.

References

[1] Hsiao, H., Steinbach, O. and Wendland, W. (2000) Domain Decomposition Methods via Boundary Integral Equations. *Journal of Computational and Applied Mathematics*, **125**, 521-537. https://doi.org/10.1016/s0377-0427(00)00488-x

[2] Baker, N., Sept, D., Holst, M. and McCammon, J. (2001) The Adaptive Multilevel Finite Element Solution of the Poisson-Boltzmann Equation on Massively Parallel Computers. *IBM Journal of Research and Development*, **45**, 427-438. https://doi.org/10.1147/rd.453.0427

[3] Diedrich, C., Dijkstra, D., Hamaekers, J., Henniger, B. and Randrianarivony, M. (2015) A Finite Element Study on the Effect of Curvature on the Reinforcement of Matrices by Randomly Distributed and Curved Nanotubes. *Journal of Computational and Theoretical Nanoscience*, **12**, 2108-2116. https://doi.org/10.1166/jctn.2015.3995

[4] Randrianarivony, M. (2004) Anisotropic Finite Elements for the Stokes Problem: A-Posteriori Error Estimator and Adaptive Mesh. *Journal of Computational and Applied Mathematics*, **169**, 255-275. https://doi.org/10.1016/j.cam.2003.12.025

[5] Apel, T. and Randrianarivony, M. (2003) Stability of the Discretizations of the Stokes Problem on Anisotropic Meshes. *Mathematics and Computers in Simulation*. **61**, 437-447. https://doi.org/10.1016/S0378-4754(02)00098-8

[6] Tomasi, J., Mennucci, B. and Cammi, R. (2005) Quantum Mechanical Continuum Solvation Models. *Chemical Reviews*, **105**, 2999-3094. https://doi.org/10.1021/cr9904009

[7] Weijo, V., Randrianarivony, M., Harbrecht, H. and Frediani, L. (2010) Wavelet Formulation of the Polarizable Continuum Model. *Journal of Computational Chemistry*, **31**, 1469-1477.

[8] Cohen, A., Daubechies, I. and Feauveau, J. (1992) Biorthogonal Bases of Compactly Supported Wavelets. *Communications on Pure and Applied Mathematics*, **45**, 485-560. https://doi.org/10.1002/cpa.3160450502

[9] Beylkin, G. (1992) On the Representation of Operators in Bases of Compactly Supported Wavelets. *SIAM Journal on Numerical Analysis*, **29**, 1716-1740. https://doi.org/10.1137/0729097

[10] Lage, C. and Schwab, C. (1999) Wavelet Galerkin Algorithms for Boundary Integral Equations. *SIAM Journal on Scientific Computing*, **20**, 2195-2222. https://doi.org/10.1137/S1064827597329989

[11] Petersdorff, T. and Schwab, C. (1996) Wavelet Approximations for First Kind Boundary Integral Equations on Polygons. *Numerische Mathematik*, **74**, 479-519. https://doi.org/10.1007/s002110050226

[12] Randrianarivony, M. (2008) Harmonic Variation of Edge Size in Meshing CAD Geometries from IGES Format. In: Bubak, M., van Albada, G.D., Dongarra, J. and Sloot, P.M.A., Eds., *Computational Science—ICCS* 2008, Springer, Berlin, 56-65. https://doi.org/10.1007/978-3-540-69387-1_7

[13] Kleemann, B., Rathsfeld, A. and Schneider, R. (1996) Multiscale Methods for Boundary Integral Equations and Their Application to Boundary Value Problems in Scattering Theory and Geodesy. In: Hackbusch, W. and Wittum, G., Eds., *Boundary Elements: Implementation and Analysis of Advanced Algorithms*, Vieweg+Teubner Verlag, Wiesbaden, 1-28. https://doi.org/10.1007/978-3-322-89941-5_1

[14] Harbrecht, H. and Randrianarivony, M. (2010) From Computer Aided Design to Wavelet BEM. *Computing and Visualization in Science*, **13**, 69-82. https://doi.org/10.1007/s00791-009-0129-1

[15] Randrianarivony, M. (2006) Geometric Processing of CAD Data and Meshes as Input of Integral Equation Solvers. Ph.D. Dissertation, Technical University of Chemnitz, Germany.

[16] Randrianarivony, M. (2009) On Global Continuity of Coons Mappings in Patching CAD Surfaces. *Computer-Aided Design*, **41**, 782-791. https://doi.org/10.1016/j.cad.2009.04.012

[17] Randrianarivony, M. (2011) On Transfinite Interpolations with Respect to Convex Domains. *Computer Aided Geometric Design*, **28**, 135-149. https://doi.org/10.1016/j.cagd.2010.10.003

[18] Harbrecht, H. and Randrianarivony, M. (2009) Wavelet BEM on Molecular Surfaces: Parametrization and Implementation. *Computing*, **86**, 1-22. https://doi.org/10.1007/s00607-009-0050-y

[19] Randrianarivony, M. and Brunnett, G. (2008) Preparation of CAD and Molecular Surfaces for Meshfree Solvers. In: Griebel, M. and Schweitzer, M.A., Eds., *Meshfree Methods for Partial Differential Equations IV*, Springer, Berlin, 231-245. https://doi.org/10.1007/978-3-540-79994-8_14

[20] Hoschek, J. and Lasser, D. (1996) Fundamentals of Computer Aided Geometric Design. A. K. Peters, Taylor & Francis, Natick.

[21] Randrianarivony, M. (2016) Domain Decomposition for Wavelet Single Layer on Geometries with Patches. *Applied Mathematics*, **7**, 1798-1823. https://doi.org/10.4236/am.2016.715151

[22] Genz, A. and Malik, A. (1980) An Adaptive Algorithm for Numeric Integration over an N-Dimensional Rectangular Region. *Journal of Computational and Applied Mathematics*, **6**, 295-302. https://doi.org/10.1016/0771-050X(80)90039-X

[23] Johnson, S. (2010). http://ab-initio.mit.edu/wiki/index.php/Cubature

[24] Bajaj, C., Xu, G. and Zhang, Q. (2009) A Fast Variational Method for the Construction of Resolution Adaptive C^2-Smooth Molecular Surfaces. *Computer Methods in Applied Mechanics and Engineering*, **198**, 1684-1690. https://doi.org/10.1016/j.cma.2008.12.042

[25] Lee, B. and Richards, F. (1971) The Interpretation of Protein Structures Estimation of Stratic Accessibility. *Journal of Molecular Biology*, **55**, 379-400. https://doi.org/10.1016/0022-2836(71)90324-X

[26] Connolly, M. (1983) Analytical Molecular Surface Calculation. *Journal of Applied Crystallography*, **16**, 548-558. https://doi.org/10.1107/s0021889883010985

[27] Schwartz, L. (1966) Théorie des Distribution. Hermann, Paris.

[28] McLean, W. (2000) Strongly Elliptic Systems and Boundary Integral Equations. Cambridge University Press, Cambridge.

[29] Atkinson, K. (1997) The Numerical Solution of Integral Equations of the Second Kind. Cambridge University Press, Cambridge. https://doi.org/10.1017/CBO9780511626340

[30] Hsiao, G. and Wendland, W. (2008) Boundary Integral Equations. Springer Verlag, Berlin. https://doi.org/10.1007/978-3-540-68545-6

[31] DeBoor, C. and Fix, G. (1973) Spline Approximation by Quasi-Interpolants. *Journal of Approximation Theory*, **8**, 19-45. https://doi.org/10.1016/0021-9045(73)90029-4

[32] Brunnett, G. (1995) Geometric Design with Trimmed Surfaces. *Computing Supplement*, **10**, 101-115. https://doi.org/10.1007/978-3-7091-7584-2_7

[33] Piegl, L. and Tiller, W. (1995) The NURBS Book. Springer, Berlin. https://doi.org/10.1007/978-3-642-97385-7

[34] Lyche, T. and Mørken, K. (1992) Spline-Wavelets of Minimal Support. In: Braess, D. and Schumaker, L.L., Eds., *Numerical Methods in Approximation Theory*, Birkhäuser, Basel, 177-194. https://doi.org/10.1007/978-3-0348-8619-2_10

[35] Lyche, T., Mørken, K. and Quak, E. (2001) Theory and Algorithms for Non-Uniform Spline Wavelets, Multivariate Approximation and Applications. Cambridge University Press, Cambridge, 152-187.

[36] Gallier, J. (2000) Geometric Methods and Applications for Computer Science and Engineering. In: Antman, S.S. and Greengard, L., Eds., *Texts in Applied Mathematics*, Springer Verlag, New York, 38.

[37] Silla, E., Pascal-Ahuir, J., Tomasi, J. and Bonaccorsi, R. (1987) Electrostatic Interaction of a Solute with a Continuum. Improved Description of the Cavity and of the Surface Cavity Bound Charge Distribution. *Journal of Computational Chemistry*, **8**, 778-787.
https://doi.org/10.1002/jcc.540080605

[38] Klamt, A. (2005) COSMO-RS: From Quantum Chemistry to Fluid Phase Thermodynamics and Drug Design. Elsevier Science, Amsterdam.

[39] Ammar, A. and Chinesta, F. (2008) Circumventing Curse of Dimensionality in the Solution of Highly Multidimensional Models Encountered in Quantum Mechanics Using Meshfree Finite Sums Decompositions. In: Griebel, M. and Schweitzer, M.A., Eds., *Meshfree Methods for Partial Differential Equations IV*, Springer, Berlin, 1-17.
https://doi.org/10.1007/978-3-540-79994-8_1

[40] Rao, A. (2005) MPI-Based Parallel Finite Element Approaches for Implicit Dynamic Analysis Employing Sparse PCG Solvers. *Advances in Engineering Software*, **36**, 181-198.
https://doi.org/10.1016/j.advengsoft.2004.10.004

[41] Zhang, J. (2015) A PETSc-Based Parallel Implementation of Finite Element Method for Elasticity Problems. *Mathematical Problems in Engineering*, **2015**, Article ID: 147286.
https://doi.org/10.1155/2015/147286

[42] Vega-Gisbert, O., Roman, J. and Squyres, J. (2016) Design and Implementation of Java Binding Open MPI. *Parallel Computing*, **59**, 1-20.
https://doi.org/10.1016/j.parco.2016.08.004