University of Zagreb
Faculty of Electrical Engineering and Computing

Master Thesis no. 75

Multiresolution analysis of macromolecular structures

Matija Piškorec

July 5, 2010

Contents

List of Figures

Nomenclature

List of used abbreviations and symbols.

Abbrevations:

MA	Multiresolution Analysis
PM	Progressive Mesh
...	

$\alpha_{i,j}$	Angle used to calculate weights for vertex v_i that corresponds to the edge $e_{i,j}$, page 21
ϕ_i^j	Scaling function i of the resolution j, page 11
ψ_i^j	Wavelet function i of the resolution j, page 11
$d_n^{k+1}(v_i^k)$	Detail coefficient for the attribute f_n of the vertex v_i on level k, page 23
f^j	Function of the resolution j, page 11
p_i	Position of vertex v_i, page 21
Rp_i	Relaxed position of vertex v_i, page 21
V^j	Linear function space of resolution j, page 11
$V_1(i)$	1-ring neighbourhood of vertex v_i, page 21
W^j	Orthogonal complement space of resolution j, page 11
$w_{i,j}$	Weight of the curvature relaxation operator for vertex v_i that corresponds to the edge $e_{i,j}$, page 21

Chapter 1

Introduction

Multiresolution analysis and wavelets have gained considerable research attention in the last two and a half decades. The main driving force were probably the promising first applications of the new methods in wide range of science and technology. In the middle of that period it became clear that traditional definition on simple one and two-dimensional domains isn't easily extensible to the more general settings - for example, on the 2-manifolds of arbitrary topological type and their discrete counterparts triangulated meshes that were at that time widely used in computer graphics.

The *second generation wavelets* came at the scene, inspired mostly by the ground breaking work of Lounsbery *et al.* [22, 23] and the invention of the *spherical wavelets* by Schröder and Sweldens [31, 32]. By exploiting the properties of *subdivision surfaces* they allowed the usual signal processing techniques to be employed in the setting of *surfaces* and *functions defined on the surfaces*. The resulting algorithms are fast and efficient, often running in linear time.

One drawback of the above mentioned approaches is that they depend heavily on the *subdivision connectivity* properties of the meshes, which effectively means that all meshes on which they operate need to have *semi-regular* structure obtained by iteratively subdividing some initial base mesh. Unfortunately, this is rarely a case one encounters with the real-world meshes. The problem of reconfiguring the topology of the mesh vertices without influencing (up to a tolerance factor) the geometry of the mesh is called *remeshing* and considerable amount of research is devoted to it. For an excellent survey see [1].

Lounsbery et al. practically started the field by themselves so they relied on their own custom method described by Eck et al. in [12]. The method first parametrize the original model onto a set of base triangular domains and then recursively subdivides the base domain to gain surface with subdivision connectivity. Schröder and Sweldens had somehow easier task because they weren't interested in surfaces themselves but only in the functions defined on one particular surface - the sphere. By recursively subdividing some predetermined base polyhedron mesh (they used icosahedron) and projecting the newly created vertices to the unit radius they obtained the suitable surface for their purposes. Of course, their method can be used for the representation of the more general surfaces, although the remeshing step is unavoidable in that case.

If one doesn't have the subdivision surface to start with, the multiresolution analysis on irregularly sampled point sets is also possible, as demonstrated by Daubechies *et al.* [7], Guskov *et al.* [16] and Valette and Prost [33]. The advantage of those methods is that they keep initial topology of the mesh intact so a completely accurate reconstruction of the initial mesh is possible. This is also the approach taken during research for this thesis.

The purpose of this thesis is to investigate the possibilities of multiresolution analysis and wavelet representation for the analysis of macromolecular structures and *molecular surfaces* in particular.

Chapter 2 gives an overview of the various types of surfaces and surface properties that can be defined given the detailed structural information of the molecule. Chapter 3 explains the theoretical background of the multiresolution analysis that can be applied to both semi-regular and irregular setting. Subdivision surfaces, important in several aspets to both of these settings, are explained in chapter 4. Chapter 5 describes the implementation of the method for irregular multiresolution analysis that was used in this thesis for the analysis of macromolecular surfaces. Finally, chapter 6 demonstrates the results of testing the simple signal processing operations on macromolecular surfaces.

Chapter 2

Macromolecular structures

In the language of protein science the most widely used macromolecular structures are the *primary*, *secondary*, *tertiary* and *quaternary* structure. In the remaining of the thesis we are concerned with macromolecular surfaces that are defined by a finite number of vertices and their connectivity and form a triangulated embedding in 3D space. We will call such representation a *triangulated mesh*.

We will further suppose that molecular surface is topologically equivalent to sphere i.e. has no holes or boundaries, although some of the analysis that will be presented is valid for surfaces that don't posses these properties.

Figure 2.1: Representation of molecular structures. The *stick* representation on the left allows the dihedral angles between individual atoms to be seen. The *cartoon* representation in the middle reveals the details about the secondary structure of the two chains that compose the molecule. Finally, the *surface* representation on the right allows as to view the solvent-accessible surface of the protein.

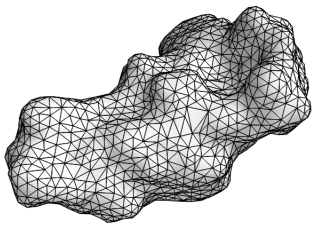

Figure 2.2: Surface of the molecule represented as a triangulated mesh.

2.1 Types of surfaces

One of the most common ways to define the protein surface is to use the envelope obtained by walking a water-radius (1.4 Å) probe along its atoms. Surface defined in such way is called a *solvent accessible surface*. All that is required for a procedure are coordinates of each atom and their radii. This data is easily obtained by parsing through the *PDB* files that store information on each atom in macromolecular complex. The most reliable repository of PDB files is *Worldwide Protein Data Bank* [2].

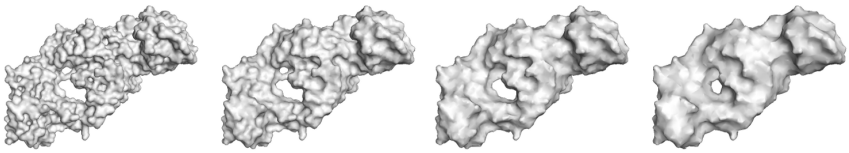

Figure 2.3: Solvent accessible surface for various radius of water-probe. From left to right the radius is 1 Å, 2 Å, 3 Å and 4 Å. Note how increasing the radius of probe corrects small holes and smooths the surface geometry.

Changing the radius of the water-probe affects the geometry of obtained surface - large probes can not fit into the small concave regions of the molecule so they tend to fill small holes and smooth the surface. Usually, a fixed 1.4 Å radius probe is suitable for most applications, although sometimes there are benefits of using probe with *variable* radius. For example, Bhat *et al.* [3] demonstrated that by changing the radius of the probe according to the hydrophobicity of the surface atoms one can easily account for the hydrophobic character of the surface by modifying its geometry.

We used a fixed radius probe with radius chosen appropriately large so that the surface doesn't have small holes. The surface is generated with *PyMOL Molecular Viewer* [9] and exported in wavefront (.obj) format so that it can be easily imported in other applications. A short program written in Python

is used to convert the data into a text format suitable for import by *Matlab* .

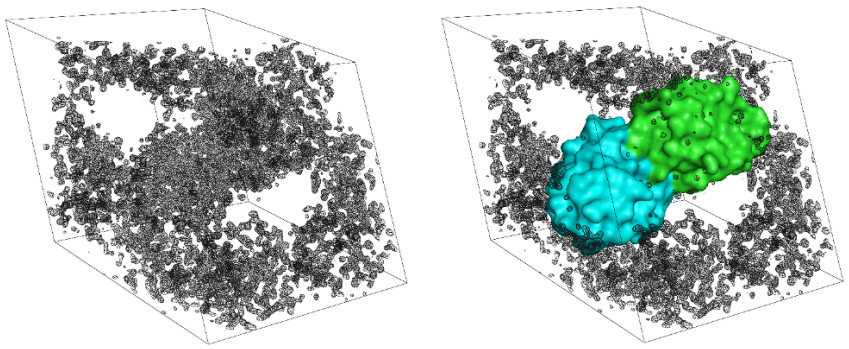

Figure 2.4: Comparison of electron density map and solvent accessible surface of 1ay7 complex. By choosing the appropriate isosurface for the electron density map one can approximate the molecular surface of the protein, though extraction of continuous smooth surface could be non-trivial.

Alternately, one can use the electron density maps or some other volume-based data to gain insight into the overall shape of the surface. As can be seen from the picture 2.4 it can be quite complicated to extract smooth continuous surface from that kind of representation, although recently Giard and MacQ [14] demonstrated fast and parametrizable algorithm that gives good quality meshes by filtering electron density maps.

2.2 Surface properties

In contras to *surfaces* themselves, there is also a need to represent *functions* on the surfaces. In context of macromolecules these functions represent various surface *properties* [1] that are of interest. The most commonly used are *hydrophobicity* and *electrostatic potential*.

2.2.1 Hydrophobicity and electrostatic potential

Both *hydrophobicity* and *electrostatic potential* are defined by the proximity of various molecular properties that aren't tied directly to the surface -

[1] Of course, as there is no way to uniquely define the "surface" of the molecule the notion "surface properties" should also be taken with care. In fact, the definition or even existence of them is not really that relevant for our purposes.

hydrophobicity by the proximity of hydrophobic groups to the molecule surface, *electrostatic potential* by the distribution of charges near the surface. Both are very relevant to the way molecules interact with an environment and each other.

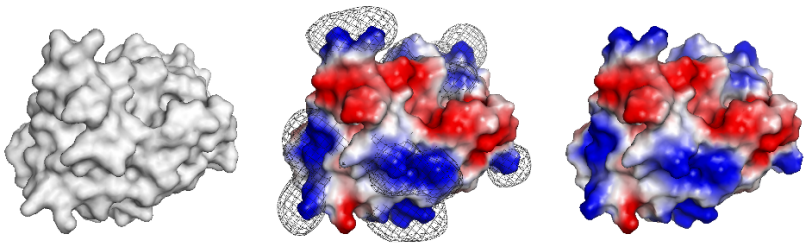

Figure 2.5: Mapping surface electrostatics to 1ppe complex. Left is the original surface, middle shows overlap between one of the isosurfaces of the generated electrostatic map and on the right is the electrostatics mapped onto a surface. This electrostatic potential is calculated in Pymol.

Figure 2.5 shows an example of mapping electrostatic potential to the molecule surface. First, the electrostatic potential map is calculated according to the constituent atoms of the molecule. Then it is visualized by isosurface in wireframe representation and superimposed on the original surface. Note that the isosurface corresponds nicely with the regions of the surface with negative potential (blue regions). Electrostatic potential map and isosurface is calculated in Pymol. Unfortunately, because of the short cut-offs, truncations and lack of solvent "screening" these computed potentials are only qualitatively useful. The further difficulty is that, in contrast to the molecular surface itself, there is no way to export these values out of the Pymol for the use in some other program.

2.2.2 Curvature of the surface

In contrast to hydrophobicity and electrostatic potential, *curvature* of the surface is inherent property of the surface itself. On surfaces defined as continuous 2-manifolds the usual measure of the curvature is the Laplace-Beltrami operator that generalizes the Laplace gradient operator. As our surfaces are defined as triangulated meshes we will need discrete approximation. For the purposes of this thesis we use *discrete mean curvature* as

an approximation. We define it as in [26]:

$$\nabla_p \text{area} M_h = \frac{1}{2} \sum_j (\cot \alpha_j + \cot \beta_j)(p - p_j) \tag{2.1}$$

Angles α_j and β_j are defined as in figure 2.6 (right). Figure also shows representation of discrete mean curvature as a vector over the polyhedral surface (left).

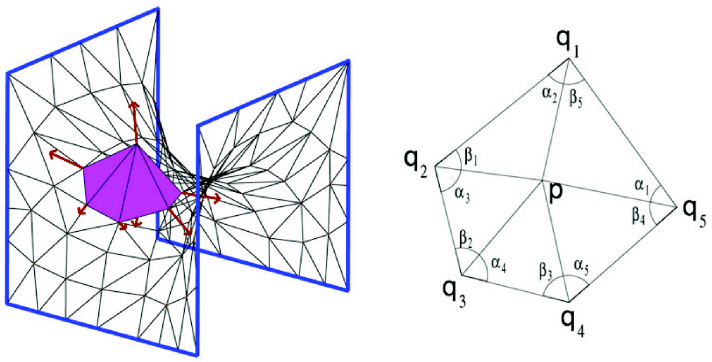

Figure 2.6: Discrete mean curvature is defined as a vector over the polyhedral surface (left). Angles used for computation (right). Figure reproduced from [5].

Figure 2.7 shows the magnitude of discrete mean curvature vector mapped onto a surface of 2wfv protein. Highly detailed parts of the surface are clearly visible on the curvature map and they correspond to the high resolution parts of the *multiresolution representation* of the surface.

2.3 Representing geometry as function

It is probably worth to clarify at this point the difference between *geometry* of triangulated surface and its *topology*. The geometry is defined solely by the position of the vertices that compose the mesh.

On the other hand, the topology is defined by the mutual connections between the vertices. Its easy to imagine a mesh that has all of its vertices positioned at the same location in space (degenerate geometry), but nevertheless having rich topological structure.

This allows us to completely change one of them and still have other preserved. For example, changing the topology of the mesh without influencing (up to an error tolerance) the geometry is called *remeshing*.

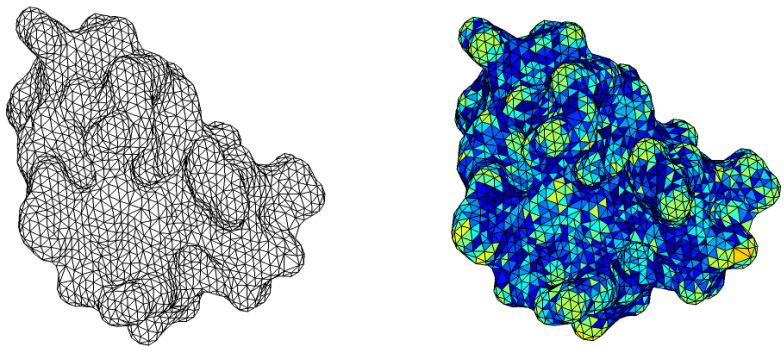

Figure 2.7: Discrete mean curvature plotted on the 2wfv protein surface. Highly detailed parts of the surface are clearly visible on the curvature map.

It also allows as to define geometry as a surface property and to map it to some other surface. Because the points that define the geometry are three-dimensional we have three separate functions $x(u, v)$, $y(u, v)$ and $z(u, v)$ for each coordinate x, y and z. Here u and v are local coordinates defined on the surface on which we parametrized the initial surface. The problem of finding suitable set of (u, v) coordinates for each point on the initial surface is called *parametrization* and it isn't easy in general. For an excellent survey see [13].

Figure 2.8 shows the results of mapping the geometry of random surface to the sphere. Only the $z(u, v)$ part is shown.

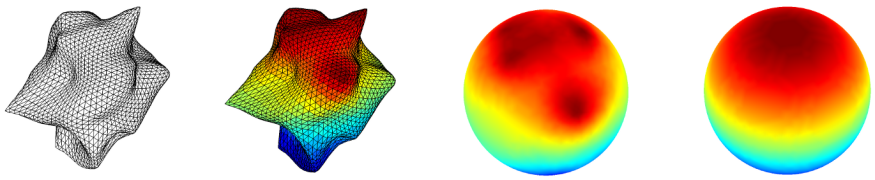

Figure 2.8: Representing geometry as a function on sphere. From left to right: original surface, surface with mapped values for the z coordinate, mapping of the function $z(u, v)$ to the sphere, comparison sphere with values for its z coordinate. Note the extreme peaks on top of the model that extend into the z-direction and that are easily recognizable on the $z(u, v)$ mapped to the sphere. For comparison, the peak on the lower left of the model extends in direction orthogonal to the z-axis and is not recognizable at all on the $z(u, v)$.

Chapter 3

Multiresolution analysis

The basic idea behind multiresolution analysis is to decompose a function into low and high resolution part in such a way that the original function can be reconstructed from them. The high resolution part can be efficiently represented as a linear combination of *wavelet functions* ψ_i^j, so the whole method is often simply referred as the *wavelet transform*. If we further decompose the low resolution part up to some definite (base) level we can discard all of the low resolution parts from higher levels and reconstruct the complete signal just by recursively adding the wavelet functions to the base signal. Because the shape of each wavelet function on different level is known in advance, the whole signal is actually represented with a series of *wavelet coefficients*.

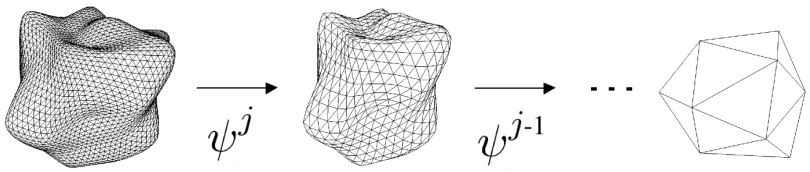

Figure 3.1: Wavelet decomposition of triangulated surface. The surface is decomposed into a sequence of low resolution approximations and a set of wavelet functions ϕ^j. Later, the low resolution approximations can be discarded and the original surface can be reconstructed solely by the lowest resolution mesh and a set of wavelet functions.

There are few applications that can benefit from the decomposition defined above:

Signal compression: Not all wavelet functions are needed for the reconstruction of original signal. Achieving a reasonably good approximation is possible with only a subset of high resolution details, so a signal can be straightforwardly compressed.

Multiresolution editing: By changing the wavelet functions that correspond to the different levels of detail one can edit the original signal at various levels of resolution.

Signal processing: Because high resolution details are often associated with the high frequencies in the original signal, it is possible to achieve approximation of the low-pass, high-pass or band-pass filter by selectively discarding wavelet functions on various levels of resolution.

Optimization: The multiple levels of approximation of original signal can offer a sort of multigrid techniques for optimization.

Traditionally, wavelets are defined on a $\mathbb{R}^1$ or $\mathbb{R}^2$ where all wavelets functions ψ_i^j are simple *translates* and *dilates* of one particular function, the *mother wavelet* ψ. These are the *first generation wavelets*

On the other hand, in this thesis we will be concerned with multiresolution analysis that can be applied in the more general setting of *surfaces* - 2-dimensional manifolds of arbitrary topological type. These are the *second generation wavelets*. Great amount of research in the last two decades is dedicated to the definition of second generation wavelets that have as much good properties as their more traditional counterparts, most important being the *fast (near linear time) algorithms*, *smoothness*, *localization* and ability to characterize various functional spaces of interest.

There are two main types of multiresolution analysis when applied in the context of surfaces: those which operate on *semi-regular* setting and those which operate on *irregular* setting. The first ones rely heavily on the properties of *subdivision surfaces* that will be explained in chapter 4. The second ones don't require semi-regular setting and can work on surfaces composed of triangles with arbitrary connectivity.

In the following sections we will provide theoretical background for the multiresolution analysis on surfaces that applies to both semi-regular and irregular setting. The chapter 5 deals exclusively with multiresolution analysis on irregular setting.

3.1 Theoretical background

Now we will elaborate basic concepts of multiresolution analysis as described in [22, 23]. There are two basic ingredients necessary for the multiresolution analysis: an infinite sequence of nested linear function spaces $V^0 \subset V^1 \subset V^2 \ldots$, and an inner product $\langle f, g \rangle$ defined on a pair of functions $f, g \in V^j$, for some $j < \inf$.

Function spaces V^j represent our signal on various levels of resolution, with $j = 0$ being the level with the lowest resolution. The inner product is necessary to define orthogonal complement spaces W^j as:

$$W^j = \{f \in V^{j+1} | \langle f, g \rangle = 0, g \in V^j\}$$

.

Now we can write any function $f^{j+1} \in V^{j+1}$ as an orthogonal decomposition

$$f^{j+1} = f^j + h^j$$

where $f^j \in V^j$ is our low resolution part and $h^j \in W^j$ is the high resolution part of the original function f^{j+1}.

The bases of V^j are called the *scaling functions* and are usually written as ϕ_i^j, and the bases of W^j are called *wavelets* and are usually written as ψ_i^j. Hence, every function f^j on the resolution level j can be represented as a linear combination of scaling functions ϕ_i^j:

$$f^j = \sum_i v_i^j \phi_i^j \tag{3.1}$$

The original wavelets described by Daubichies [8] and Mallat [24] are *fully orthogonal* meaning that every wavelet is orthogonal to every other wavelet. Full orthogonality is in most cases hard or even impossible to achieve, so it is more convenient to define *semiorthogonal* wavelets that require orthogonality only between wavelets at different levels. However, sometimes its necessary to drop the notion of orthogonality altogether and to require orthogonality just between *some* of the wavelets at different levels. Usually it will be the ones that are spatially close to each other. This least restrictive form of wavelets are called *biorthogonal wavelets*.

3.1.1 Refinement equation

We will now give more precise definition of the nested linear spaces. In particular, we are interested in how, given the highest resolution function

from V^j, to obtain the lower resolution function from V^{j-1}. The equation 3.1 tells us that every function on level j can be written as a linear combination of the scaling functions on the level j. The question is now redefined as: How to obtain the scaling functions from the level $j-1$ given the scaling functions on the level j? It will most certainly be some form of *linear combination*, but unfortunately nothing more can be said for the general case.

When we have *spatially invariant* domain it turns out that simple *translation* and *dilation* of the original scaling function is enough to define the scaling functions on the coarser level. In that case all scaling functions on the coarser level can be obtained by a simple *refinement equation*:

$$\phi(x) = \sum_i p_i \phi(2x - 1) \tag{3.2}$$

Unfortunately, as we are dealing with surfaces whose domain does not satisfy spacial invariance we can not rely on the refinement equation for the construction of scaling function. Instead, we will use properties of *subdivision surfaces* to define refinement on the surface.

Chapter 4

Subdivision surfaces

Subdivision was originaly technique for building smooth functions starting from a coarse description. It is closely connected with the *upsampling* procedure where non-existent signal values are approximated with the help of nearby values. In the context of *surfaces* the theoretical work traces back to 1978 when papers by Doo and Sabin [10, 11] and Catmull and Clark [6] developed subdivision schemes for defining tangent-plane smooth surfaces.

They are important for the multiresolution analysis because they allow the definition of *refinability* in the setting of surfaces. For example, Lounsbery *et al.* [23] demonstrates that if V^j is a matrix whose i-th row consists of the x, y and z coordinates of vertex i then subdivision procedure can be characterized with matrix P^j in order to obtain the subdivided surface V^{j+1}:

$$V^{j+1} = P^j V^j \tag{4.1}$$

As we would expect from a subdivision procedure, matrix P^j depends only on the connectivity of vertices in V^j, not their actual positions. More important, the same matrix P^j can be used to obtain the scaling functions on lower resolution from the scaling functions on the finer resolution:

$$\Phi^j(x) = \Phi^{j+1}(x) P^j \tag{4.2}$$

where $\Phi^j(x)$ defines the row matrix of scaling functions $\phi_i^j(x)$. This equation establishes refinability because it states that each of the scaling function

$\phi_i^j(x)$ can be written as a linear combination of scaling functions ϕ_i^{j+1}.

4.1 Types of subdivision

The subdivision procedures are traditionally divided into two categories: *primal* and *dual*. A primal subdivision procedure uses a refinement procedure to refine the faces of the mesh into a subfaces. The dual subdivision procedures do exactly the same but additionally take the dual [1] of the resulting mesh. In the case of primal subdivision procedures that operate on triangular faces each triangular face is divided into four faces that are defined by the initial vertices and the vertices added in the middle of each edge. This first step is called *splitting* and it is the same in all primal subdivision procedures.

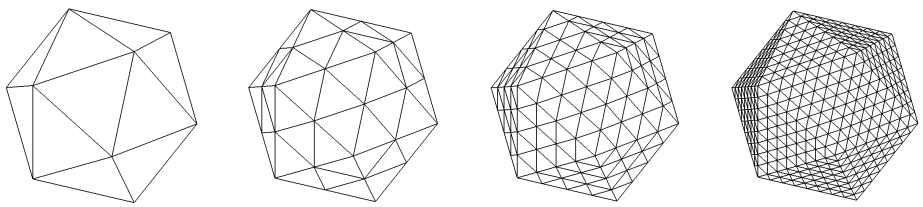

Figure 4.1: Polyhedral subdivision of the icosahedron. The new vertices are simply located at the middle of the edges.

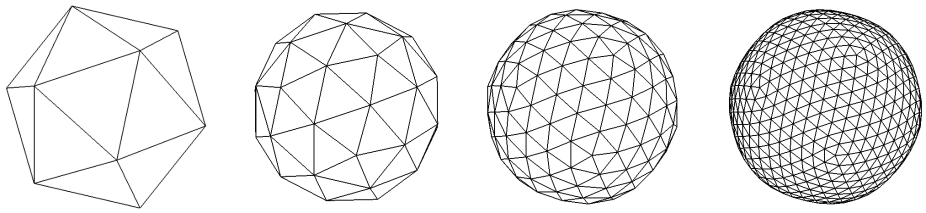

Figure 4.2: Butterfly subdivision of the icosahedron.

The second step is *averaging* and it is here where primal subdivision procedures differ. For example, figures 4.1 and 4.2 show result of subdividing

[1] The dual of the mesh is obtained by positioning the vertices of the dual mesh on the center of each face of the original mesh. Each vertex is then connected with the one that belongs to the neighbouring face in the original mesh.

the initial icosahedron mesh with *polyhedral* and *butterfly* subdivision procedure respectively. Both are examples of *interpolating* subdivision procedures because they don't change the positions of old vertices.

The polyhedral procedure doesn't do any averaging and simply positions new vertices exactly at the middle of each edge. On the other hand, butterfly procedure calculates the position of the new vertex according to the eight-point *stencil* illustrated on figure 4.3.

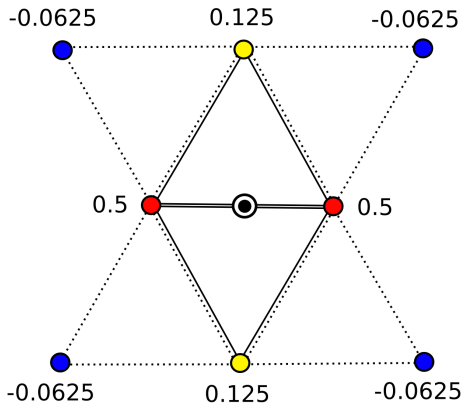

Figure 4.3: Stencil and corresponding weights for the butterfly subdivision procedure.

Polyhedral and butterfly procedure was used by Lounsbery *et al.* [23] in their multiresolution analysis that runs in linear time. They note that using more complicated procedures such as Loop or Catmull and Clark procedures leads to much higher time complexity. Butterfly stencil was used by Schröder and Sweldens [31, 32] for their *spherical wavelets*, although they experimented with other as well.

Alternative way of categorizing subdivision procedures became important after the advent of multiresolution analysis on irregular meshes. According to it, each subdivision procedure can be *uniform*, *semi-uniform* or *non-uniform* depending on the weights used in subdivision procedures. The butterfly and polyhedral procedures mentioned above belong to the semi-uniform category because the weights depend only on the local connectivity of triangulation.

Non-uniform subdivisions are especially important for the multiresolution analysis on irregular meshes because they allow the smooth approximations of any number of irregularly sampled vertices. The weights used in that case depend on the local connectivity *and* geometry of the mesh. The usual subdivision procedure in non-uniform setting begins by adding some number of

vertices and connecting them to the vertices of original mesh. It then calculates their positions according to the connectivity and geometry of nearby vertices. The operators used for calculation of new positions are often called *relaxation operators* because they try to find *smooth* approximation of finite set of vertices. Various relaxation operators can be used for smoothing - for example, Guskov *et al.* in [15, 16] used non-uniform relaxation operator that minimizes *second-differences*. In chapter 5 we will use a similar operator described by Roy *et al.* in [29] to define the non-uniform subdivision on irregular mesh.

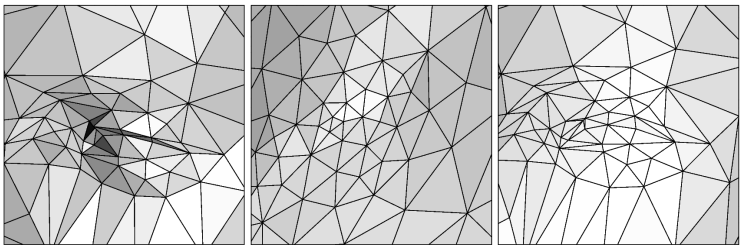

Figure 4.4: Comparison of semi-uniform (middle) and non-uniform (right) smoothing of the original triangular mesh (left). The semi-uniform subdivision tends to average the lengths of the edges distorting the geometry in the process. In contrast, non-uniform smoothing based on relaxation operators smooths only the geometry of the surface while keeping the original triangle shapes. Figure reproduced from [16].

If one wish just to obtain smooth surfaces starting from an irregularly connected mesh, a modified version of traditional butterfly and loop subdivision procedures can be used. For example, Zorin *et al.* in [34] describe one such modification of butterfly subdivision procedure that can be used with good results.

4.2 Regularity of the mesh

Whether or not certain types of multiresolution analysis can be performed depends on the *regularity* of the mesh. All meshes fall in one distinct category: *regular*, *semi-regular* and *irregular* meshes. The figure 4.5 illustrates the three types on 2D plane.

On *regular* meshes *all* vertices have *exactly* six neighbours. These are appropriately called *regular* vertices. Connectivity is implicit while compactness and regularity of data structure improves performance of many algorithms. On *semi-regular* meshes majority of vertices are regular, but there are also some *extraordinary* vertices that have more or less than six neighbours.

Because they are more easily obtained than regular meshes, semi-regular meshes are the most popular choice for doing multiresolution analysis that requires subdivision surface. In that case the lowest resolution (base) mesh consists mostly of extraordinary vertices and his shape usually greatly influences the outcome of the analysis.

Finally, *irregular* meshes consist of vertices of any degree. This is the most common type of regularity and the one usually encountered in meshes that have not been processed by some *remeshing* procedure. The multiresolution analysis is still possible - for example, Daubechies et al. [7], Guskov et al. [16] and Vallete and Prost [33] define multiresolution analysis on irregular point sets.

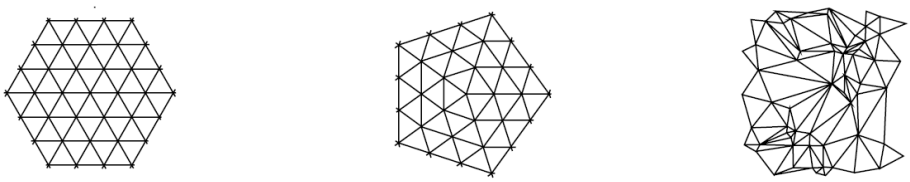

Figure 4.5: Examples of regular, semi-regular and irregular triangle meshes. Figure reproduced from [7].

4.3 Multiresolution editing

In the context of multiresolution analysis editing the subdivision surface at spatial location i on the mesh of resolution level j corresponds to the editing of one particular wavelet function ψ_i^j. This is in contrast to the transformations like *spherical harmonics* that can also represent surfaces but due to their global support are unable to localize particular geometry feature on the surface.

The figure 4.6 illustrates the effect of changing one particular scaling function ϕ_i^j at various resolution level. The overall shape of the scaling function ϕ_i^j on resolution level j is simply the *hat function* centred on vertex i and falling linearly along the incident faces. The fact that neighbouring vertices at all subsequent levels of resolution are affected by a single scaling function at the lower level suggests the possibility of exploiting the spatial and frequency correlation of different parts of the surface. The butterfly subdivision procedure is used for generating the subdivided surface.

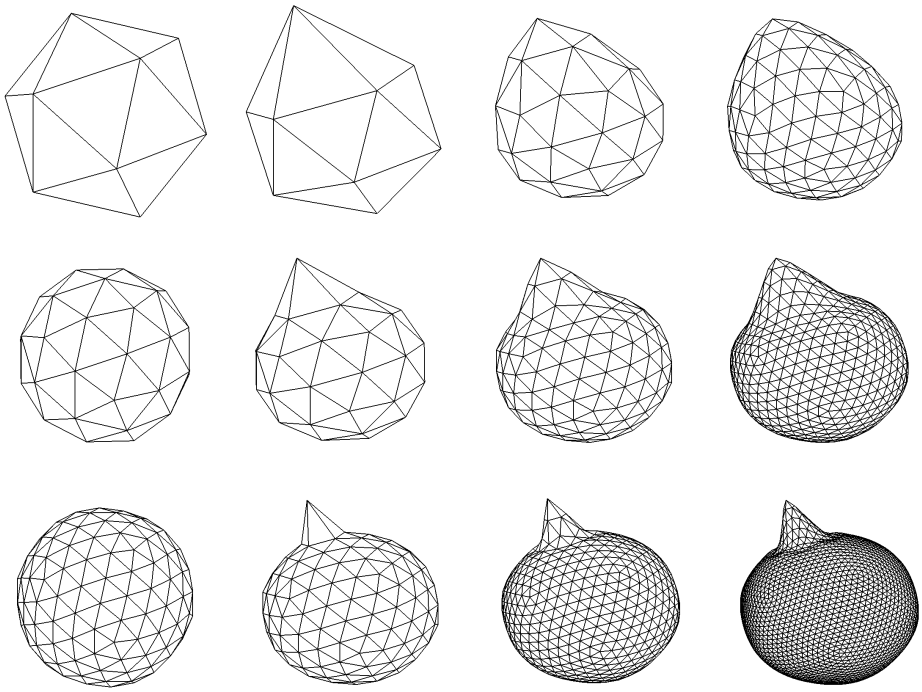

Figure 4.6: Multiresolution editing of single vertex at various resolutions. Note how editing the vertex at low levels of resolution affects the position of neighbouring vertices at all subsequent levels.

Chapter 5

Analysis in irregular setting

The complications with remeshing the original irregular surface into a semi-regular one can be avoided if one chooses to use a multiresolution analysis capable to operate on irregular meshes. One of the main advantages of such approach is that *exact* reconstruction of original mesh is possible. It is also one of the main drawbacks because non-uniform subdivision scheme can not be used as a predictor in compression framework [16].

Because topological information (i.e. connectivity of vertices) of the mesh can not be reconstructed by wavelet coefficients alone a framework for defining the coarser levels of resolution and encoding the decomposition is needed. The coarser levels of resolution for irregular triangulations is easily obtained by various *mesh simplification* algorithms that incrementally decimate the finer mesh. For example, one can use simple *edge*, *half-edge* or *vertex collapse* operations to obtain mesh with fewer faces, edges and vertices.

The decomposition encodings are also numerous to choose from. For example, Bonneau used hierarchical Delaunay triangulation to encode the decomposition in [4], while Roy *et al.* in [29] used Hoppe's Progresive Mesh (PM) framework [19].

Multiresolution analysis in irregular setting started when Bonneau introduced concept of multiresolution analysis over non-nested spaces in [4]. In this thesis we will describe more recent method of Roy *et al.* [29, 30] that is inspired by the method of Guskov *et al.* [16].

5.1 Mesh simplification

Mesh *simplification* [1] techniques consists of various algorithms that transform a given polygonal mesh into another mesh that has fewer faces, edges and vertices. First applications were in the field of computer graphics but it soon became clear that they can also provide framework for doing a multiresolution analysis on irregular surface meshes. An excellent (although little outdated) survey was written by Heckbert *et al.* [18].

For our purposes we will use a sequence of simple *half-edge collapse* operations to obtain the simplified mesh and modified Progresive Mesh (PM) framework from Hoppe [19] to encode our decomposition.

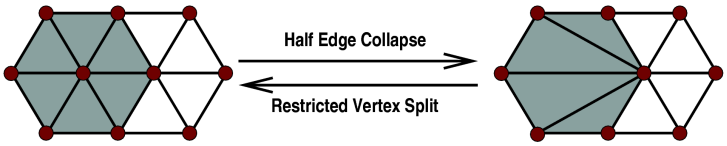

Figure 5.1: Half-edge collapse and its inverse vertex split operation that are used for simplification of original mesh. Figure reproduced from [5].

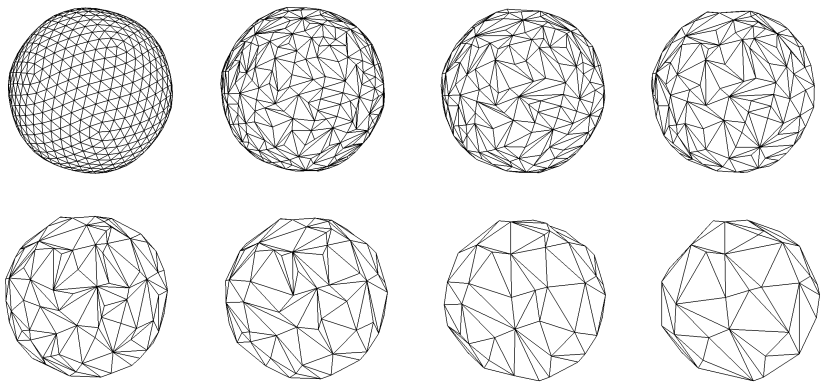

Figure 5.2: Progressive simplification of the subdivided sphere by the sequence of half-edge collapses. Vertices that are not removed do not change positions so this procedure effectively *sub-samples* the mesh.

[1] Mesh *decimation* is also widely used expression throughout the literature.

5.2 Geometric relaxation

Relaxation operators are used to smooth the surface before each step of decomposition. The differences between original and smoothed surface are then stored and used as an equivalent of *wavelet coefficients* in irregular setting. We use the *curvature relaxation* operator introduced by Roy *et al.* in [29]. It is inspired by discrete differential-geometry operators described by Meyer *et al.* in [25]. In contrast to some other discrete operators (e.g. discrete Laplacian that replaces the vertex with the average of its 1-ring neighbours) this operator smooths the geometry and does not affect the triangle shapes much. The relaxed position of Rp_i of vertex v_i is given by:

$$Rp_i = \sum_{j \in V_1(i)} w_{i,j} \quad \text{with} \quad w_{i,j} = \frac{\cot \alpha_{i,j} + \cot \alpha_{i,j}}{\sum_{l \in V_1(i)} (\cot \alpha_{i,j} + \cot \alpha_{i,j})} \tag{5.1}$$

where $V_1(i)$ represents 1-ring neighbourhood of vertex v_i. The weights $w_{i,j}$ minimize the curvature energy of an edge e_i where $\alpha_{i,j}$ and $\alpha_{i,j}$ are angles opposite to the edge e_i (see figure 5.3). Figure 5.4 shows the effect of several curvature relaxations on molecular surface.

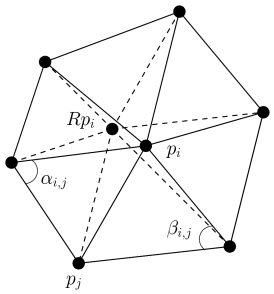

Figure 5.3: Angles used in calculation of curvature relaxation of vertex p_i. Figure reproduced from [29].

What is interesting in this approach is that using the same equation 5.1 we can relax not just position p_i but any other vertex attribute $f_n(v_i)$. In fact, the position p_i is defined as three distinct attributes each corresponding to the x, y and z coordinate of vertex v_i. In this way the multiresolution analysis of surfaces is defined as a special case of multiresolution analysis of functions defined on surfaces allowing us to easily extend the analysis to the various surface properties described in chapter 2.

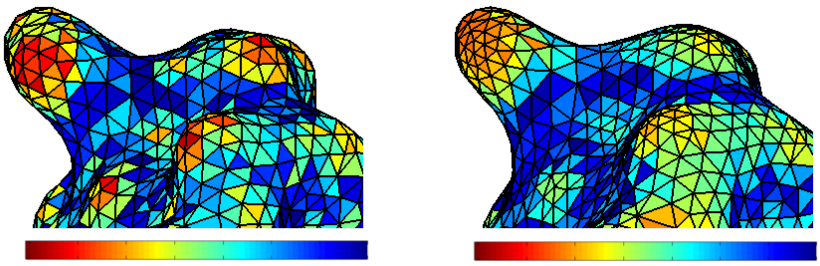

Figure 5.4: Curvature relaxation of molecular surface. On the left is an original surface with values of discrete mean curvature mapped onto each triangle. On the right is the same surface with curvature relaxation operator applied several times. Only the surface geometry is affected while triangle shapes are preserved. Areas with high curvature are coloured red.

5.3 Decomposition

The decomposition decomposes original surface into a sequence of low resolution approximations and a set of detail coefficients. We start from a finest mesh M^0 and compute the sequence of meshes M^k as well as set of differences (details) D^k between successive resolutions. Main steps in analysis are:

Simplification: The simplification downsamples the initial mesh M^k by removing some vertices, giving the approximation M^{k+1} of the fine mesh.

Subdivision: The Subdivision upsamples the initial mesh M^{k+1} by inserting previously removed vertices, giving an estimation S^k of the fine mesh.

Detail computation: The difference between the fine mesh M^k and the subdivided mesh (estimation) S^k gives the detail coefficients D^{k+1} associated with the mesh M^{k+1}. The details are computed for every vertex of the fine mesh M^k.

Decomposition is based on incremental mesh simplification, so to obtain the *invertible* decomposition procedure one needs simplification technique that is also invertible. This is why we choose the half-edge collapse as a basic operation for our simplification procedure. Its corresponding operation in the reconstruction phase is *vertex split* operation where new vertex is added to the triangulated mesh so as to have the same connectivity as before the

removal. It is important to store the adjacency information for each vertex that is removed, as well as the information on parent vertex (i.e. vertex into which we collapse the edge).

Additionally, if one wishes to preserve the orientation of surface throughout the reconstruction the information on parent's *left* and *right* neighbour vertices is also needed. The orientation is needed because we use *dihedral angles* of retriangulated neighbourhoods as a simple guidance for the choice of parent vertex for each vertex to be removed. Other measures can also be used - for example, Roy *et al.* in [29] used Quadric Error Metric.

In addition to the above mentioned *topological* information that is needed to reconstruct the topology of the mesh there is also the *geometric* information that is used for the reconstruction of topology. It consists of already mentioned relaxation weights $w_{i,j}$ that are stored for each edge on each level of decomposition and the *detail coefficients* $d^{k+1}(v_i^k)$ that are defined as a difference between the relaxed and the original vertices:

$$\forall v_i^k \in M^k, \quad d_n^{k+1}(v_i^k) = f_n(v_i^k) - Rf_n(v_i^k) \tag{5.2}$$

5.4 Reconstruction

Using all the above information the original surface can finally be reconstructed, level-by-level, with this formula:

$$f_n(v_i^k) = \sum_{j \in V_1(i)} w_{i,j}^k f_n(v_j^{k+1}) + d_n^{k+1}(v_i^k) \quad \text{with} \quad n \in [1, N] \tag{5.3}$$

New vertices (i.e. those that have been returned to the original mesh at that particular level) need to be reconstructed first, with all other vertices following.

Chapter 6

Results

The solvent-accesible surface of `2wfv` protein obtained with Pymol was chosen for testing the Matlab implementation of irregular multiresolution decomposition and reconstruction. It contains 3654 vertices and 7304 faces so decomposition and reconstruction are finished in a few minutes. This is far too slow in comparison with the implementation of Roy *et al.* [29] - their version runs in order of seconds even for very large meshes of over more than 100000 vertices. The difference in speed is probably due to the implementation in Matlab, rather than in some low-level language like C or C++.

In the remain of this chapter the tests of decomposition and reconstruction are described, as well as implementation of basic signal processing operations on surfaces using irregular multiresolution analysis.

6.1 Decomposition of 2wfv protein surface

Figure 6.1 demonstrates the results of decomposition of protein surface into iteratively coarser approximations. Colour is here to help visualize axis that extends orthogonal to the view. Figure 6.2 shows the details coefficients mapped onto the original surface. Individual bright spots correspond to the vertices that are removed at that level. It can be observed that number of removed vertices at each successive level is decreasing.

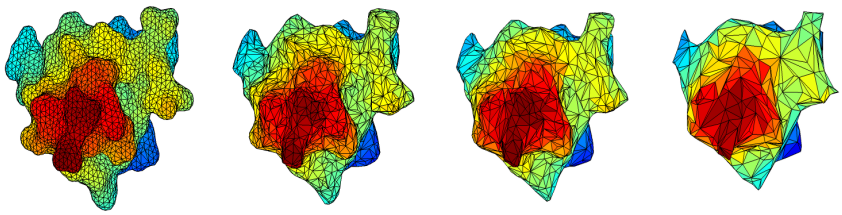

Figure 6.1: `2wfv` protein through various resolutions. From left to right resolutions are 1, 4, 6 and 9.

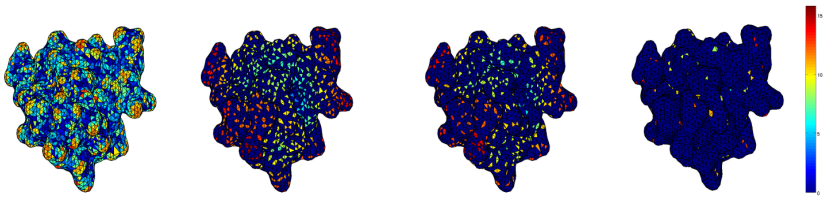

Figure 6.2: Magnitude of details coefficients from various resolutions mapped onto surface. From left to right resolutions are 1, 2, 4 and 10.

6.2 Signal processing on surface

Once the detail coefficients are obtained from multiresolution decomposition one can use them either to completely reconstruct the original mesh or to perform some basic signal processing operations on it. For example, because details from lower levels correspond to the higher level resolution, low-pass filter can be approximated by simply setting all detail coefficients *below* certain level to zero. Figure 6.3 demonstrates this by setting all detail coefficients from levels below 1, 2, 5 and 10 to zero.

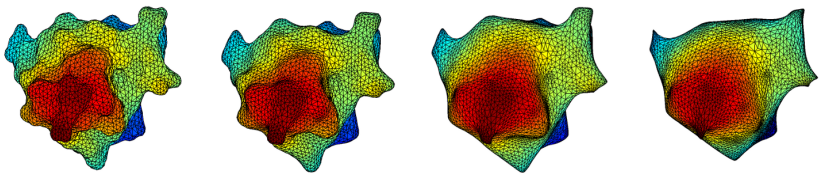

Figure 6.3: Effects of low-pass filter on surface of `2wfv` protein. Low-pass filter is implemented by simply setting all detail coefficients *below* certain level to zero (low details levels correspond to higher resolutions). From left to right levels are 1, 2, 5 and 10.

Analogously, the figure 6.4 demonstrates how the high-pass filter can be approximated by setting all detail coefficients *above* certain level to zero (in this case the levels above 1, 3, 6 and 10).

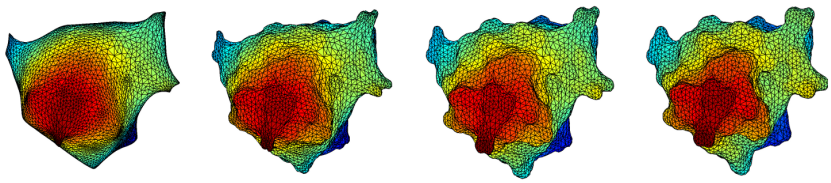

Figure 6.4: Effects of high-pass filter on surface of `2wfv` protein. High-pass filter is implemented by simply setting all detail coefficients *above* certain level to zero (high details levels correspond to lower resolutions). From left to right levels are 1, 3, 6 and 10.

Chapter 7

Conclusion

One of the initial motivation for doing a research in the field of multiresolution analysis was one particular question: Is it possible to define *geometric complementarity* measure for the surfaces in the wavelet domain and what are the computational benefits of using such an approach? Inspiration came from the similar method proposed by Ritchie and Kemp in [27, 28] that uses the combination of spherical harmonics and radial basis functions to find suitable set of solutions to the protein docking problem. The underlying problem with using only the spherical harmonics, as noticed in before cited articles, is that it is computationally difficult to manipulate a pair of surfaces that are defined in different coordinate systems, which is exactly the problem with the wavelet representation as well. Ritchie and Kemp solve the problem by defining the skin of the protein by adding a radial basis functions that extend 2D surface into the 3D space, in that way setting common reference frame for both surfaces and allowing the efficient computation of surface complementarity.

Unfortunately, given the immense scope of research that was published in the last few decades, there was not enough time to get acquainted with the field and in the same time adequately tackle the problem of surface complementarity in wavelet domain. First approach was to investigate the methods of Lounsbery *et al.* [22, 23] and Schröder and Sweldens [31, 32]. Because their methods operate in semi-regular setting, the majority of time was spent in finding the suitable remeshing procedure for the preparation of input mesh to multiresolution analysis. Various approaches were considered: original Eck et al. method [12] for parametrization to the base triangular mesh, MAPS method by Lee et al. [21] that also finds parametrization to the

base triangular weights but avoids expensive calculation of PL embedding harmonic maps, shrink wrapping approach of of Kobbelt et al. [20], Laplace-Beltrami operator approach by Angenent et al. [17] that parametrizes onto a sphere by a conformal (angle-preserving) map. All of them were either too complex for implementation or not well explained in the original articles where only basic sketch of the procedures were given.

For that reason our research interest shifted toward multiresolution analysis that are capable of operating in irregular setting. The approach of Guskov *et al.* [16] was first considered but in the end choice fell on the method of Roy *et al.* [29] because it was somewhat simpler to implement. In addition to its applicability in the irregular setting, the method is also capable of processing surfaces of arbitrary topological type and it trivially generalizes to the processing of functions on the surface. Its multiresolution representation proved sufficient for simple signal processing operations on the original surface such as low-pass and high-pass filtering.

Bibliography

[1] P. Alliez, G. Ucelli, C. Gotsman, and M. Attene. *Shape Analysis and Structuring*, chapter Recent Advances in Remeshing of Surfaces. Springer, 2007.

[2] Berman, Henrick, and Nakamura. Worldwide protein data bank, June 2010. URL http://www.wwpdb.org/docs.html.

[3] S. Bhat and E.O. Purisima. Molecular surface generation using a variable-radius solvent probe. *Proteins: Structure, Function, and Bioinformatics*, 2006.

[4] G.-P. Bonneau. Multiresolution analysis on irregular surface meshes. *IEEE Transactions on Visualization and Computer Graphics*, 1998.

[5] M. Botsch, M. Pauly, L. Kobbelt, P. Alliez, B. Lévy, and S. Bischoff. Geometric modeling based on polygonal meshes. Eurographics 2008 Full-Day Tutorial lecture notes, 2008.

[6] E. Catmull and J. Clark. Recursively generated b-spline surfaces on arbitrary topological meshes. *Computer Aided Design*, 1978.

[7] I. Daubechies, I. Guskov, P. Schröder, and W. Sweldens. Wavelets on irregular point sets. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 1999.

[8] I. Daubichies. Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, 1988.

[9] Warren L. DeLano. Pymol molecular viewer, June 2010. URL http://www.pymol.org/.

[10] D. Doo. *A Recursive Subdivision Algorithm for Fitting Quadratic Surfaces to Irregular Polyhedrons*. Phd thesis, Brunel University, 1978.

[11] D. Doo and M. Sabin. Behavior of recursive division surfaces near extraordinary points. *Computer Aided Design*, 1978.

[12] M. Eck, T. De Rose, T. Duchamp, H. Hoppe, M. Lounsbery, and W. Stuetzle. Multiresolution analysis of arbitrary meshes. *Computer Graphics Proceedings, annual conference series: SIGGRAPH conference proceedings*, 1995.

[13] M.S. Floater and K. Hormann. Surface parametrization: a tutorial and survey. *Advances on Multiresolution in Geometric Modelling*, 2005.

[14] J. Giard and B. MacQ. Molecular surface mesh generation by filtering electron density map. *International Journal of Biomedical Imaging*, 2010.

[15] I. Guskov. Multivariate subdivision schemes and divided differences. Technical report, Department of Mathematics, Princeton University, 1998.

[16] I. Guskov, W. Sweldens, and P. Schröder. Multiresolution signal processing for meshes. *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, 1999.

[17] S. Haker, S. Angenent, A. Tannenbaum, and R. Kikinis. On the laplace-beltrami operator and brain surface flattening. *IEEE Transactions on Medical Imaging*, 1999.

[18] P.S. Heckbert and M. Garland. Survey of polygonal surface simplification algorithms. Technical report, Carnegie Mellon University, 1997.

[19] H. Hoppe. Progressive meshes. *Proceedings of ACM SIGGRAPH*, 1996.

[20] L. P. Kobbelt, J. Vorsatz, U. Labsik, and H. Seidel. A shrink wrapping approach to remeshing polygonal surfaces. *Computer Graphics Forum*, 1999.

[21] A. W. F. Lee, W. Sweldens, P. Schröder, L. Cowsar, and D. Dobkin. Maps: Multiresolution adaptive parameterization of surfaces. *Computer Graphics*, 1998.

[22] M. Lounsbery. *Multiresolution Surfaces of Arbitrary Topological Type*. Phd thesis, University of Washington, January 1994.

[23] M. Lounsbery, T.D. DeRose, and J. Warren. Multiresolution analysis for surfaces of arbitrary topological type. *ACM Transactions on Graphics*, 1997.

[24] S. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1989.

[25] M. Meyer, M. Desbrun, P. Schröder, and A. Barr. Discrete differential-geometry operators for triangulated 2-manifolds. *Proceedings of Visualization and Mathematics*, 2002.

[26] K. Polthier. Computational aspects of discrete minimal surfaces. *Proceedings of the Clay Mathematics Institute 2001 Summer School*, 2001.

[27] D.W. Ritchie. *Parametric Protein Shape Recognition.* Phd thesis, University of Aberdeen, 1998.

[28] D.W. Ritchie and G.J.L. Kemp. Fast computation, rotation, and comparison of low resolution spherical harmonic molecular surfaces. *Journal of Computational Chemistry*, 1999.

[29] M. Roy, S. Foufou, A. Koschan, F. Truchetet, and Abidi M. Multiresolution analysis for irregular meshes. *Proceedings of the SPIE Photonics East, Wavelet Applications in Industrial Processing*, 2003.

[30] M. Roy, S. Foufou, A. Koschan, F. Truchetet, and Abidi M. Multiresolution analysis for meshes with appearance attributes. *Proceedings of the IEEE International Conference on Image Processing (ICIP'05)*, 2005.

[31] P. Schröder and W. Sweldens. Spherical wavelets: Efficiently representing functions on the sphere. *Proceedings of the 22nd Annual ACM Conference on Computer Graphics and Interactive Techniques*, 1995.

[32] P. Schröder and W. Sweldens. Spherical wavelets: Texture processing. *Rendering Techniques*, 1995.

[33] S. Valette and P. Prost. Wavelet-based multiresolution analysis of irregular surface meshes. *IEEE Transactions on Visualization and Computer Graphics*, 2004.

[34] D. Zorin, P. Schröder, and W. Sweldens. Interpolating subdivision for meshes with arbitrary topology. *Proceedings of the 1996 Computer Graphics Conference, SIGGRAPH*, 1996.

Summary

Matija Piškorec: Multiresolution analysis of macromolecular structures

Keywords: multiresolution, wavelets, subdivision, mesh, macromolecules, surfaces

This thesis provides theoretical background on multiresolution analysis and its application in the analysis of macromolecular structures with special emphasis on macromolecular surfaces that can be represented as triangular meshes. Also, possibilities of multiresolution analysis of various physical and chemical properties of those surfaces, and properties that are intrinsic to surfaces themselves (i.e. curvature) are also explored in the context of functions defined on surfaces. The difference between semi-regular and irregular setting is explained, as well as the importance of subdivision surfaces for the multiresolution analysis in semi-regular setting. Further emphasis is put on multiresolution analysis in irregular setting. Program implementation of irregular subdivision is developed in order to perform simple signal processing operations on triangulated meshes such as low-pass and high-pass filtering.