Tetrahedron Topological Characterization with Application in Volumetric Reconstruction

Luis Gustavo Nonato, Antonio Castelo,
José E. P. Pires de Campos, Helton H. Bíscaro and Rosane Minghim

March 27, 2008

Universidade de São Paulo - Brazil
Instituto de Ciências Matemáticas e de Computação
{gnonato,castelo,jeppc,helton,rminghim}@icmc.usp.br

Abstract: One of the challenges of Volume Modelling is the definition of theoretical frameworks to support object manipulation and representation. Opposite to what is seen in the field of surface modelling, few satisfactory tools have been presented that ensure topological robustness to volumetric models. In this paper we offer one such mathematical framework for volumetric model definition based on tetrahedral meshes. From a complete tetrahedron topological characterization, a set of Morse Operators is given, which enable global tolopogical control during tetrahedron addition or removal. A number of applications can be envisaged, from geometrical modelling to volume reconstruction. We show the effectiveness of the tetrahedron characterization framework for volume reconstruction from images, showing that the method is capable of handling certain types of noise topologically without the need for a time consuming preprocessing step, or a post-processing step to detect cavities and holes.

keywords: Morse Operators; Topological Characterization; Tetrahedral Mesh

1 Introduction

Computational studies of topological nature of objects have attracted researchers from several fields of science. The wide interest for such subject is motivated by the large number of computational applications that would benefit from representing topology, such as image processing, solid and volume modelling, visualization, and computer graphics in general [1, 2, 3].

The particular area of Volume Modelling (VM) refers to the body of tools and techniques to handle scientific data in 3D space [2]. Arising from the area of scientific visualization, volume modelling searches for a framework for manipulation of volumetric models and display to support advanced visualization

applications. In VM not much is available to compound such a framework, mainly because most visualizations so far have been realized as a final step in the data understanding process, whilst new applications need to have control over the structures under manipulation in various other stages of the process such as data representation and interaction. For instance, changes made during interaction must be tracked back to previous stages of the visualization pipeline, a task too expensive to perform without proper volumetric representation.

The subject of volume representation bears importance in all other areas of VM. In the current stage of development, it relies in a handful of computational techniques with little mathematical support[2]. In particular, topological aspects of volume models have been largely neglected, regardless of bearing great weight in how VM will develop. Special features of meshes should be under control during simulation, data traversal, mapping, visualization and interaction, since holes, singularities, cavities, borders and other features affect all volume modelling tasks and must be handled with flexibility and robustness, as well as efficiency. In that respect, topological approaches to volume modelling present various advantages over more conventional, analytical or geometrical approaches, in both requirements, i.e., robustness of implementation and control over object features.

This is the case of many applications, and particularly in the case of reconstruction from images. Segmentation processes are slow and hard to complete satisfactorily via conventional techniques, and 3D models are geometrically unstable during construction. In that case topological methods for dealing with objects may contribute towards fast and reliable automatic reconstruction from measuring devices, such as MRI.

In order to analyze the topological properties of objects computationally it is necessary to make use of a description of such object and its topological relationships. For surface models, this is done through Euler operators[4], widely employed to attain consistency and robustness. With the growth of VM, the area is now in need of similar techniques that offer the same advantages for objects defined volumetrically. This makes the cell decomposition issue central to volume modelling.

An specially important kind of volumetric cell decomposition is the tetrahedral decomposition, due to the variety of applications that employ this kind of representation: three-dimensional reconstruction [5], isosurface extraction [6], mesh generation [7, 8] and many others. Although tetrahedral representations have been widely employed, not much has been done to understand their topological properties when compared with voxel representations [9, 10].

These are the main motivations for the work described here, which presents a complete study of tetrahedron characterization, i.e., we analyze and classify the topological changes caused by the insertion and removal of tetrahedra in a tetrahedral object model. Additionally, this work offers a set of Morse operators (TMO's) for tetrahedral meshes that are capable of describing simplicial complexes completely in a similar way to surface Euler operators. They are tools for construction and manipulation of objects described by tetrahedral meshes, maintaining full control of object topology.

We employ these generic tetrahedral operators to the problem of volume reconstruction from images. This gives rise to an algorithm capable of modelling the volume of objects from a set of their cross sections. Apart from thresholding performed on the volume data, this volume reconstruction algorithm requires no further pre-processing step. Objects reconstructed during the process are treated topologically, so that scattered objects (such as those resulting from noise) can be easily detected and eliminated.

This text is organized as follows: section 2 presents a brief description of previous work on topological approaches applied to the various areas related to VM. Basic concepts necessary to understand the notation and nomenclature employed in the remaining of the text is given in section 3. Section 4 presents the topological tetrahedral characterization and the Morse operators associated with it. Section 5 introduces a short discussion of the implementation and complexity of the Morse operators. Section 6 presents the application of the framework in volumetric reconstruction and its results, followed by our conclusions and future work in Section 7.

2 Related Work

In order to treat some of the problems and needs of areas related to volume modelling, researchers have turned to computational topology tools [11, 9], such as the one presented here. The following text identifies needs and efforts involving computational topology associated with VM areas.

In the issue of data organization, concepts of topology have been employed to improve efficiency of the various steps of the visualization process. Virtually all visualization processes handling data large enough not to fit in memory could benefit from proper data indexing via geometrical or topological information of the original data set. For instance, in large data sets, a re-organization of the data using topological indexing may speed up the process of finding groups of cells of interest for a particular visualization procedure, such as checking for surface intersection [10].

Isosurface extraction is another subject that has benefited from the topological approach, in order to solve various problems arising from surface fitting of data. These include shape ambiguity, generation of holes where none existed in the original data, excess of primitives in the surface model, and poor mesh organization leading to inefficient traversal. Several topological tools have been employed to solve problems such as topological consistency [12], mesh simplification [13], and speed up of the surface fitting process [14].

In the subject of volume representation, tetrahedral meshes are produced by cell decomposition processes, which face a number of problems. The quality of the mesh itself and the type of connectivity amongst elements deserve special attention. Additionally, access to mesh elements (mesh representation) leads to difficulties during interaction, for instance achieving real time performance during rebuilding of the mesh submitted to operations such as cuts and slicing. In the cell decomposition context, computational topology has shown to be

an essential mechanism to study various problems of mesh generation [15] and interaction [16].

The potential of topological tools for computational applications has motivated the investigation of important questions related to the computation of homology [11] and characterization of tetrahedra [17]. These cpncepts offer immediate tools to build a VM framework. This is very important when handling irregular objects or phenomena in an unstructured mesh setup. A typical example is tetrahedral mesh compression where topological control is a crucial mechanism to design efficient algorithms [18].

In particular, the work by Saha and Majumder [17] shows a sound mathematical framework to analyze the local characterization of tetrahedra. Their characterization allows the measurement of local topological changes caused by deleting or inserting a tetrahedron. In their work, the type of tetrahedron called a 'simple' tetrahedron is a type of mesh element whose elimination or insertion does not affect the mesh topology locally. For non-simple tetrahedra the tool measures local change in topology, without discussing what happens to the mesh globally.

Global topological changes are important for the detection of volumetric characteristics, solution of some problems such as elimination of noise - reflected as cavities or holes (see section 6) - and measurements of various quantities. Global control is also important for indexing purposes when those characteristics must be accessed frequently. The tools presented here aim at obtaining global control of tetrahedral models.

In order to increase the topological control during object manipulation, some authors have already employed computational tools derived from Morse theory. These tools generally make use of handle attachment theory to update topological data structures that represent two or three-dimensional manifolds [19, 20] and stratified manifolds [21]. Although interesting to deal with manifolds, such approaches do are effective in manipulating three-dimensional simplicial complexes, as accomplished by the tools presented here.

In the present work we propose a set of tools for global characterization of tetrahedra, that can be seen as an extension of some results obtained by Saha and Majumder [17], since it provides a framework to analyze tetrahedral meshes. Using our framework, however, it is possible to tell, once the mesh changes locally, what the effect will be on the whole mesh. Besides that characterization, we provide a set of operators to implement insertion and removal of tetrahedra keeping topological control of the meshes at all times. These operators are called 3D tetrahedral Morse operators, and they provide a topologically consistent framework for many tasks of volume modelling.

3 Basic Concepts

This section introduces the basic concepts and terminology used in the remaining of the text. Definitions and results presented in this and the following sections are restricted to three-dimensional Euclidean space. A reference to

many of the concepts presented in this section is the book by Munkres [22].

A *p-dimensional simplex* or p-simplex in $\mathbb{R}^3$, $0 \leq p \leq 3$, is the convex hull of $p+1$ geometrically independent points in $\mathbb{R}^3$ (a set of points $\{v_0, \ldots, v_p\} \subset \mathbb{R}^3$ is geometrically independent if the vectors $v_1 - v_0, \ldots, v_p - v_0$ are linearly independent). A 0-simplex is called a *vertex*, a 1-simplex is called an *edge*, a 2-simplex is called a *triangle* and a 3-simplex is called a *tetrahedron*. If $\sigma = \{v_0, \ldots, v_p\}$ is a p-simplex, $p = 0, 1, 2, 3$, then any j-simplex, $0 \leq j \leq p$, spanned by a subset S of $\{v_0, \ldots, v_p\}$ (that is, the convex hull of S) is called a *face* of σ. The faces of σ different from σ itself, are called the *proper faces* of σ; their union is called the *boundary* of σ and denoted $Bd(\sigma)$. The interior of a simplex σ is defined by the equation $Int(\sigma) = \sigma - Bd(\sigma)$.

A *simplicial complex* $\mathcal{K}$ is a finite collection of simplices satisfying:

1. If $\sigma \in \mathcal{K}$ then all faces of σ belong to $\mathcal{K}$.

2. If $\sigma, \tau \in \mathcal{K}$ then either $\sigma \cap \tau = \emptyset$ or $\sigma \cap \tau$ is a common face of σ and τ.

The *dimension* of the simplicial complex $\mathcal{K} \neq \emptyset$, denoted $dim(\mathcal{K})$, is the maximum of the dimensions of the simplices in $\mathcal{K}$. A subcollection of $\mathcal{K}$ that is itself a simplicial complex is called a (simplicial) subcomplex of $\mathcal{K}$. The subcomplex of $\mathcal{K}$ constituted by all simplices of $\mathcal{K}$ of dimension at most p, $p = 0, 1, 2$ is called the *p-skeleton* of $\mathcal{K}$. The subset $|K| = \bigcup_{\sigma \in \mathcal{K}} \sigma$ of $\mathbb{R}^3$ is called the *underlying space of* $\mathcal{K}$.

A *regularized simplicial complex* $\mathcal{K}$ is a three-dimensional simplicial complex such that any p-simplex, in $\mathcal{K}$, $p = 0, 1, 2$ is contained in at least one 3-simplex of $\mathcal{K}$.

From now on, $\mathcal{K}$ is always a regularized simplicial complex. A 2-simplex t (triangle) of $\mathcal{K}$ is an *interior* triangle if t is shared by two tetrahedra of $\mathcal{K}$, otherwise, t is a *boundary* triangle. The vertices and edges contained in the boundary triangles are called *boundary vertices* and *boundary edges* of $\mathcal{K}$, respectively. The *boundary* of $\mathcal{K}$ is the subcomplex $\mathcal{S} \subset \mathcal{K}$ constituted by all boundary p-simplex, $p = 0, 1, 2$. The boundary of each connected component of $\mathbb{R}^3 - |\mathcal{K}|$ is called a *boundary component* of $\mathcal{K}$.

The *star* of a simplex $\sigma \in \mathcal{K}$, denoted $st(\sigma)$, is the union of all simplices in $\mathcal{K}$ having σ as a face. The *link* of σ is the union of all simplices of $\mathcal{K}$ lying in $st(\sigma)$ that are disjoint from σ. A simplex $\sigma \in \mathcal{K}$ is *singular* if its link is not homeomorphic to a sphere or to a half-sphere.

Let σ be a p-simplex. Two orderings of its vertex set are equivalent if they differ from each other by an even permutation. If $p > 0$, there are two equivalence classes (and if $p = 0$, just one). Each one of these classes is called an *orientation* of σ. An *oriented simplex* is a simplex σ together with an orientation. The oriented simplex with vertices $v_0, \ldots, v_p$ whose orientation is the equivalence class of the particular ordering $(v_0, \ldots, v_p)$ is denoted by $[v_0, \ldots, v_p]$.

Let $\mathcal{K}$ be a simplicial complex. Let $C_p(\mathcal{K})$ be the free abelian group generated by the p-simplices of $\mathcal{K}$, $p = 0, 1, 2$, each one of them with a fixed orientation. $C_p(\mathcal{K})$ is called the *group of (oriented) p-chains* of $\mathcal{K}$. An element of $C_p(\mathcal{K})$ is

called a *p-chain*. Therefore, any p-chain can be written, in a unique way, as $\sum_{i=1}^{k} n_i\sigma_i$, where k is the number of p-simplices σ_i of $\mathcal{K}$ and, $\forall i \in \{1,\ldots,k\}$, n_i is an integer. If $p > 0, -\sigma_i$ represents σ_i with its opposite orientation. Note that the operation in $C_p(\mathcal{K})$ is given by $(\sum_{i=1}^{k} n_i\sigma_i) + (\sum_{i=1}^{k} m_i\sigma_i) = \sum_{i=1}^{k}(n_i + m_i)\sigma_i$, where $\forall i \in \{1,\ldots,k\}$, m_i and n_i are integers.

We define a homomorphism $\partial_p : C_p(\mathcal{K}) \to C_{p-1}(\mathcal{K})$, called the *boundary operator*, by $\partial_p[v_0,\ldots,v_p] = \sum_{i=0}^{p}(-1)^i[v_0,\ldots,\hat{v}_i,\ldots,v_p]$, where $[v_0,\ldots,v_p]$ is an oriented simplex of $\mathcal{K}$ and the symbol $\hat{v}_i$ means that the vertex v_i is to be deleted from the array. We complete the definition by defining ∂_p to be the trivial homomorphism if $C_p(\mathcal{K})$ or $C_{p-1}(\mathcal{K})$ is the trivial group. It can be shown that ∂_p is well defined, that $\partial_p(-\sigma) = -\partial_p(\sigma)$ for all $\sigma = [v_0,\ldots,v_p]$ and that, $\forall p \in \mathbb{Z}$, $\partial_p \circ \partial_{p+1} = 0$.

The kernel of the boundary operator $\partial_p : C_p(\mathcal{K}) \to C_{p-1}(\mathcal{K})$, denoted $Z_p(\mathcal{K})$, is called the group of *p-cycles* of $\mathcal{K}$. The image of $\partial_{p+1} : C_{p+1}(\mathcal{K}) \to C_p(\mathcal{K})$ is called the group of *p-boundaries* of $\mathcal{K}$ and is denoted $B_p(\mathcal{K})$. Since $\partial_p \circ \partial_{p+1} = 0$, $B_p(\mathcal{K}) \subset Z_p(\mathcal{K})$, therefore we have the quotient group $H_p(\mathcal{K}) = Z_p(\mathcal{K})/B_p(\mathcal{K})$ called the p^{th} *homology group* of $\mathcal{K}$.

The rank of $H_p(\mathcal{K})$, denoted by $\beta_p(\mathcal{K})$, represents the *number of connected components*, when p equals 0; the *number of holes*, when p equals 1; and the *number of cavities*, when p equals 2 in $\mathcal{K}$. The $\beta_p(\mathcal{K})$ are called Betti numbers of the homology group. If n_v, n_e, n_f, and n_t are the number of vertices, edges, triangles, and tetrahedra in $\mathcal{K}$, the *Euler characteristic* of $\mathcal{K}$ is defined by $\chi(\mathcal{K}) = n_v - n_e + n_f - n_t$. It can be shown that the Euler characteristic of $\mathcal{K}$ can also be computed as $\chi(\mathcal{K}) = \beta_0(\mathcal{K}) - \beta_1(\mathcal{K}) + \beta_2(\mathcal{K})$.

A sequence of abelian groups A and homomorphisms ϕ

$$\cdots \to A_1 \xrightarrow{\phi_1} A_2 \xrightarrow{\phi_2} A_3 \to \cdots$$

is said to be exact at A_i if $im(\phi_{i-1}) = ker(\phi_i)$. It is said to be an exact sequence if it is exact in all of its groups.

We finish this section with an important result known as the Mayer-Vietoris Sequence [22].

`Mayer-Vietoris Sequence:` Let $\mathcal{K}$ be a simplicial complex; let $\mathcal{K}_0$ and $\mathcal{K}_1$ be subcomplexes such that $\mathcal{K} = \mathcal{K}_0 \cup \mathcal{K}_1$. Then there is an exact sequence

$$\cdots \to H_p(\mathcal{K}_0 \cap \mathcal{K}_1) \to H_p(\mathcal{K}_0) \oplus H_p(\mathcal{K}_1) \to H_p(\mathcal{K}) \to H_{p-1}(\mathcal{K}_0 \cap \mathcal{K}_1) \to \cdots$$

of abelian groups and homomorphisms, where $\oplus$ is the direct sum of groups.

In the next section the concepts presented here are employed to support the characterization of tetrahedra and the definition of Morse Operators for tetrahedral meshes.

4 Topological Operators for construction and characterization of tetrahedral meshes

In this section we investigate how the insertion or removal of a tetrahedron can change the homology of a simplicial complex $\mathcal{K}$. Based on this discussion we introduce the concept of tetrahedral Morse operators, which provide a robust mechanism to control the homology during the construction of a three-dimensional simplicial complex.

4.1 Characterization of Tetrahedra

Let τ be a new tetrahedron to be added to $\mathcal{K}$. The homological change caused by adding τ in $\mathcal{K}$ is related with the intersection $\tau \cap \mathcal{K}$. A tetrahedron τ is called *adjacent* to $\mathcal{K}$ if $\tau \notin \mathcal{K}$ and $\tau \cap \mathcal{K}$ is either empty or a subcomplex of $\mathcal{K}$ and of τ. We shall say that two simplices τ_1 and τ_2 adjacent to $\mathcal{K}$ are equivalent if $\tau_1 \cap \mathcal{K}$ and $\tau_2 \cap \mathcal{K}$ are homeomorphic, that is, if there is a bijection f between the set of vertices of $\tau_1 \cap \mathcal{K}$ and the set of vertices of $\tau_2 \cap \mathcal{K}$ such that the vertices $v_0, \ldots, v_n$ of $\tau_1 \cap \mathcal{K}$ span a simplex of $\tau_1 \cap \mathcal{K}$ if and only if $f(v_0), \ldots, f(v_n)$ span a simplex of $\tau_2 \cap \mathcal{K}$. Of course this is an equivalence relation.

Lemma (The number of equivalence classes): There are twenty eight equivalence classes for the set of adjacent tetrahedra to a simplex $\mathcal{K}$.

Proof The proof of this lemma can be done by exhaustive enumeration of the cases. In fact, let $\mathcal{L}^{(p)}$ be the p-skeletons, $p = 0, 1, 2$ of the complex $\mathcal{L} = \mathcal{K} \cap \tau$. If $\mathcal{L} = \emptyset$ we have an equivalence class (Figure 1a. Suppose that $\mathcal{L}^{(0)} \neq \emptyset$ and $\mathcal{L}^{(p)} = \emptyset$ for $p = 1, 2$. In this case, the cardinality of $\mathcal{L}^{(0)}$ can be $1, 2, 3$ or 4 and we have four equivalence classes in this case (Figure 1b). If $\mathcal{L}^{(1)} \neq \emptyset$ and $\mathcal{L}^{(2)} = \emptyset$ we can have fourteen equivalence classes as indicated in Figure 1c. Note that the tetrahedra in each dashed box are in classes where the p-skeletons of the intersections have the same cardinality but the intersections are not homeomorphic as simplicial complexes. In the last case, when $\mathcal{L}^{(2)} \neq \emptyset$, we have nine equivalence classes, as represented in Figure 1d).

The equivalence classes described in lemma 4.1 represent the different manners of inserting a tetrahedron in $\mathcal{K}$.

Regarding homological changes, adding tetrahedra from different classes can produce either the same homological change (if any) or homologically distinct complexes. It is also worth noting that the intersection between a tetrahedron τ and $\mathcal{K}$ is not sufficient to decide about the homology of the resulting simplicial complex $\mathcal{K} \cup \tau$. An example of this fact is shown in Figure 2 where the addition of a tetrahedron with three common edges with $\mathcal{K}$ can either generate a cavity (Figure 2a) or close a hole (Figure 2b) of $\mathcal{K}$.

Although the intersection $\tau \cap \mathcal{K}$ does not characterize the homological changes in $\tau \cup \mathcal{K}$, it does determine all of its possibilities. Before showing that, we analise the homology of the intersection $\tau \cap \mathcal{K}$.

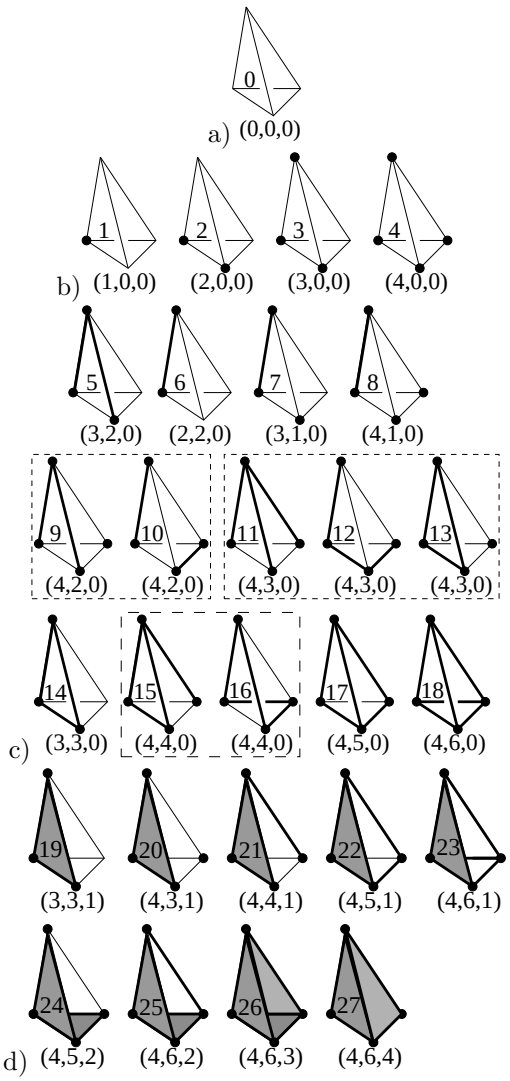


Figure 1: Equivalence classes of the tetrahedra adjacent to $\mathcal{K}$. The three numbers on the bottom of each figure means the number of vertices, edges and triangles in the intersection, respectively.

Lemma (Ranks of homology groups of $\tau \cap \mathcal{K}$): Let τ be a tetrahedron adjacent to a simplicial complex $\mathcal{K}$. Let $\beta_0, \beta_1, \beta_2$ be the ranks of the homology groups of $\tau \cap \mathcal{K}$. If $\beta = (\beta_0(\tau \cap \mathcal{K}), \beta_1(\tau \cap \mathcal{K}), \beta_2(\tau \cap \mathcal{K}))$ is the triple representing the ranks of the homology groups of $\tau \cap \mathcal{K}$ then $\beta \in \{(0,0,0), (1,0,0), (2,0,0), (3,0,0), (4,0,0), (2,1,0), (1,1,0), (1,2,0), (1$

Proof From Figure 1 we derive table 1 that proofs the lema 4.2.

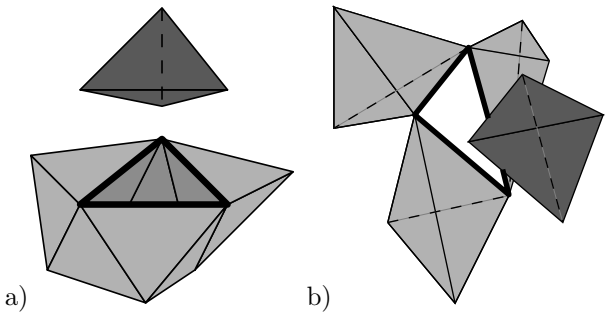


Figure 2: The insertion of the tetrahedron a) generates a cavity; b) closes a hole.

Equivalence Class	$(\beta_0(\tau \cap \mathcal{K}), \beta_1(\tau \cap \mathcal{K}), \beta_2(\tau \cap \mathcal{K}))$
0	$(0,0,0)$
1,5,6,11,12, 19,21,24,26	$(1,0,0)$
2,7,9,10,20	$(2,0,0)$
3,8	$(3,0,0)$
4	$(4,0,0)$
13	$(2,1,0)$
14,15,16,22,25	$(1,1,0)$
17,23	$(1,2,0)$
18	$(1,3,0)$
27	$(1,0,2)$

Table 1: Betti numbers of $\tau \cap \mathcal{K}$. Equivalence Classes in the first column correspond to numbers in Figure 1

Proposition (Tetrahedra topological characterization): Let τ be a tetrahedron adjacent to a simplicial complex $\mathcal{K}$. If $\alpha = (\beta_0(\tau \cap \mathcal{K}), \beta_1(\tau \cap \mathcal{K}), \beta_2(\tau \cap \mathcal{K}))$ and $\nu = (\beta_0(\tau \cup \mathcal{K}) - \beta_0(\mathcal{K}), \beta_1(\tau \cup \mathcal{K}) - \beta_1(\mathcal{K}), \beta_2(\tau \cup \mathcal{K}) - \beta_2(\mathcal{K}))$ then α is related with ν as follows:

1. if $\alpha = (0,0,0)$ then $\nu = (1,0,0)$
2. if $\alpha = (1,0,0)$ then $\nu = (0,0,0)$
3. if $\alpha = (2,0,0)$ then $\nu \in \{(-1,0,0), (0,1,0)\}$
4. if $\alpha = (3,0,0)$ then $\nu \in \{(-2,0,0), (-1,1,0), (0,2,0)\}$
5. if $\alpha = (4,0,0)$ then $\nu \in \{(-3,0,0), (-2,1,0), (-1,2,0), (0,3,0)\}$
6. if $\alpha = (2,1,0)$ then $\nu \in \{(-1,-1,0), (-1,0,1), (0,0,0), (0,1,1)\}$

7. if $\alpha = (1,1,0)$ then $\nu \in \{(0,-1,0),(0,0,1)\}$
8. if $\alpha = (1,2,0)$ then $\nu \in \{(0,-2,0),(0,-1,1),(0,0,2)\}$
9. if $\alpha = (1,3,0)$ then $\nu \in \{(0,-3,0),(0,-2,1),(0,-1,2),(0,0,3)\}$
10. if $\alpha = (1,0,2)$ then $\nu = (0,0,-1)$

Proof Let us consider the Mayer-Vietoris sequence:

$$0 \to H_2(\mathcal{K}\cap\tau) \stackrel{\phi_2}{\to} H_2(\mathcal{K}) \oplus H_2(\tau) \stackrel{\psi_2}{\to} H_2(\mathcal{K}\cup\tau) \stackrel{\Delta_2}{\to}$$

$$\stackrel{\Delta_2}{\to} H_1(\mathcal{K}\cap\tau) \stackrel{\phi_1}{\to} H_1(\mathcal{K}) \oplus H_1(\tau) \stackrel{\psi_1}{\to} H_1(\mathcal{K}\cup\tau) \stackrel{\Delta_1}{\to}$$

$$\stackrel{\Delta_1}{\to} H_0(\mathcal{K}\cap\tau) \stackrel{\phi_0}{\to} H_0(\mathcal{K}) \oplus H_0(\tau) \stackrel{\psi_0}{\to} H_0(\mathcal{K}\cup\tau) \stackrel{\Delta_0}{\to} 0$$

Let us prove item 6 of Proposition 4.1, i.e., suppose that $(\beta_0(\tau\cap\mathcal{K}),\beta_1(\tau\cap\mathcal{K}),\beta_2(\tau\cap\mathcal{K})) = (2,1,0)$. In this particular case, the sequence above can be rewritten as follows:

$$0 \to 0 \stackrel{\phi_2}{\to} H_2(\mathcal{K}) \stackrel{\psi_2}{\to} H_2(\mathcal{K}\cup\tau) \stackrel{\Delta_2}{\to}$$

$$\stackrel{\Delta_2}{\to} \mathbb{Z} \stackrel{\phi_1}{\to} H_1(\mathcal{K}) \stackrel{\psi_1}{\to} H_1(\mathcal{K}\cup\tau) \stackrel{\Delta_1}{\to} \quad (1)$$

$$\stackrel{\Delta_1}{\to} \mathbb{Z}\oplus\mathbb{Z} \stackrel{\phi_0}{\to} H_0(\mathcal{K}) \oplus \mathbb{Z} \stackrel{\psi_0}{\to} H_0(\mathcal{K}\cup\tau) \stackrel{\Delta_0}{\to} 0$$

We have two cases to analyze, either: (a) τ intersects two distinct connected components of $\mathcal{K}$ or (b) τ intersects only one connected component of $\mathcal{K}$.
Case (a): In this case $\beta_0(\mathcal{K}\cup\tau) = \beta_0(\mathcal{K}) - 1$ and we have exact sequences

$$0 \stackrel{\phi_2}{\to} H_2(\mathcal{K}) \stackrel{\psi_2}{\to} H_2(\mathcal{K}\cup\tau) \stackrel{\Delta_2}{\to} im(\Delta_2) \stackrel{\phi_1}{\to} 0$$

and

$$H_2(\mathcal{K}\cup\tau) \stackrel{\Delta_2}{\to} \mathbb{Z} \stackrel{\phi_1}{\to} H_1(K) \stackrel{\psi_1}{\to} H_1(\mathcal{K}\cup\tau) \stackrel{\Delta_1}{\to} \mathbb{Z}\oplus\mathbb{Z} \stackrel{\phi_0}{\to} H_0(\mathcal{K}) \oplus \mathbb{Z}.$$

Either $\Delta_2 = 0$ or $im(\Delta_2) = n\mathbb{Z}$ with $n \neq 0$. If $\Delta_2 = 0$ we have

$$0 \stackrel{\phi_2}{\to} H_2(\mathcal{K}) \stackrel{\psi_2}{\to} H_2(\mathcal{K}\cup\tau) \stackrel{\Delta_2}{\to} 0 \quad (2)$$

and

$$0 \to \mathbb{Z} \stackrel{\phi_1}{\to} H_1(K) \stackrel{\psi_1}{\to} H_1(\mathcal{K}\cup\tau) \stackrel{\Delta_1}{\to} \mathbb{Z}\oplus\mathbb{Z} \stackrel{\phi_0}{\to} H_0(\mathcal{K}) \oplus \mathbb{Z}. \quad (3)$$

From (2) it follows that $\beta_2(\mathcal{K}\cup\tau) = \beta_2(\mathcal{K})$. Additionally, since the generators of each copy of $\mathbb{Z}$ in the domain of ϕ_0 are going to different components in $\mathcal{K}$, ϕ_0 is injective, so $im(\Delta_1) = ker(\phi_0) = \{0\}$ and (3) becomes

$$0 \to \mathbb{Z} \stackrel{\phi_1}{\to} H_1(K) \stackrel{\psi_1}{\to} H_1(\mathcal{K}\cup\tau) \to 0,$$

therefore $\beta_1(\mathcal{K} \cup \tau) = \beta_1(\mathcal{K}) - 1$, thus $\nu = (-1, -1, 0)$. On the other hand, if $im(\Delta_2) = n\mathbb{Z}$ with $n \neq 0$, then we have

$$0 \overset{\phi_2}{\to} H_2(\mathcal{K}) \overset{\psi_2}{\to} H_2(\mathcal{K} \cup \tau) \overset{\Delta_2}{\to} n\mathbb{Z} \to 0 \tag{4}$$

and

$$H_2(\mathcal{K} \cup \tau) \overset{\Delta_2}{\to} \mathbb{Z} \overset{\phi_1}{\to} H_1(K) \overset{\psi_1}{\to} H_1(\mathcal{K} \cup \tau) \overset{\Delta_1}{\to} \mathbb{Z} \oplus \mathbb{Z} \overset{\phi_0}{\to} H_0(\mathcal{K}) \oplus \mathbb{Z}. \tag{5}$$

From (4) it follows that $\beta_2(\mathcal{K} \cup \tau) = \beta_2(\mathcal{K}) + 1$. Additionally, since ϕ_0 is injective, (5) becomes

$$H_2(\mathcal{K} \cup \tau) \overset{\Delta_2}{\to} \mathbb{Z} \overset{\phi_1}{\to} H_1(K) \overset{\psi_1}{\to} H_1(\mathcal{K} \cup \tau) \to 0$$

therefore $\beta_1(\mathcal{K} \cup \tau) = \beta_1(\mathcal{K}) - rank(ker(\psi_1)) = \beta_1(\mathcal{K}) - rank(im(\phi_1))$, but $rank(ker(\phi_1)) = rank(im(\Delta_2)) = 1$, so $rank(im(\phi_1)) = 0$, therefore $\beta_1(\mathcal{K} \cup \tau) = \beta_1(\mathcal{K})$. Thus, $\nu = (-1, 0, 1)$.

Case (b): In this case $\beta_0(\mathcal{K} \cup \tau) = \beta_0(\mathcal{K})$. The computation of $H_2(\mathcal{K} \cup \tau)$ is exactly the same as in case (1), therefore if $\Delta_2 = 0$ then $\beta_2(\mathcal{K} \cup \tau) = \beta_2(\mathcal{K})$ and if $\Delta_2 \neq 0$ then $\beta_2(\mathcal{K} \cup \tau) = \beta_2(\mathcal{K}) + 1$. Regarding $H_1(\mathcal{K} \cup \tau)$ we have that ϕ_0 is no longer injective. Instead, $ker(\phi_0) = \mathbb{Z}$ now. If $\Delta_2 = 0$, we have the exact sequence

$$0 \to \mathbb{Z} \overset{\phi_1}{\to} H_1(K) \overset{\psi_1}{\to} H_1(\mathcal{K} \cup \tau) \overset{\Delta_1}{\to} \mathbb{Z} \to 0$$

therefore $\beta_1(\mathcal{K} \cup \tau) - rank(im(\psi_1)) = 1$, but $rank(im(\psi_1)) = \beta_1(\mathcal{K}) - rank(ker(\psi_1))$ and $rank(ker(\psi_1)) = rank(im(\phi_1)) = 1$, hence $\beta_1(\mathcal{K} \cup \tau) = \beta_1(\mathcal{K})$, thus $\nu = (0, 0, 0)$. Now, if $im(\Delta_2) = n\mathbb{Z}$ with $n \neq 0$, we have

$$H_2(\mathcal{K} \cup \tau) \overset{\Delta_2}{\to} \mathbb{Z} \overset{\phi_1}{\to} H_1(K) \overset{\psi_1}{\to} H_1(\mathcal{K} \cup \tau) \overset{\Delta_1}{\to} \mathbb{Z} \oplus \mathbb{Z} \overset{\phi_0}{\to} H_0(\mathcal{K}) \oplus \mathbb{Z}$$

therefore $\beta_1(\mathcal{K} \cup \tau) - rank(im(\psi_1)) = 1 = rank(im(\Delta_1))$ but $rank(im(\psi_1)) = \beta_1(\mathcal{K}) - rank(ker(\psi_1))$ and $rank(ker(\psi_1)) = rank(im(\phi_1)) = 0$, hence $\beta_1(\mathcal{K} \cup \tau) = \beta_1(\mathcal{K}) + 1$. Thus, $\nu = (0, 1, 1)$.

The same kind of reasoning that we employed to prove item 6, was used to prove the remaining items.

An important result that can be extracted from Proposition 4.1 is:

Corollary (Number of different homological changes): There are twenty five different forms of changing the homology of a simplicial complex $\mathcal{K}$ by attaching tetrahedra.

The proof of Corollary 4.1 is straightforward from the cases shown in Proposition 4.1.

Proposition 4.1 has four important consequences. First, we can group the equivalence classes in Figure 1 in accordance with the possible homological

changes that their tetrahedra can produce in the complex. We arrange the equivalence classes in six categories named: 0-handle, H_0-handle, H_2-handle, H_0H_1-handle, H_1H_2-handle, and $H_0H_1H_2$-handle. Following the numbering presented in Figure 1, each category is constituted as follows:

1. 0-handle= $\{1, 5, 6, 11, 12, 19, 21, 24, 26\}$
2. H_0-handle= $\{0\}$
3. H_2-handle= $\{27\}$
4. H_0H_1-handle= $\{2, 3, 4, 7, 8, 9, 10, 20\}$
5. H_1H_2-handle= $\{14, 15, 16, 17, 18, 22, 23, 25\}$
6. $H_0H_1H_2$-handle= $\{13\}$

If τ is a tetrahedron contained in a class of 0-handles then τ is called a 0-handle tetrahedron; if τ is contained in a class of H_0-handles it is called H_0-handle tetrahedron and so on. The names of the handles are indicative of the types of homological changes, that is, a 0-handle causes no change in the homology, while an H_0-handle causes changes in the number of connected components, an H_2-handle causes changes in the number of cavities, and so forth.

The second consequence of Proposition 4.1 above is that it offers a way of quantifying and describing the nature of the homological change caused by adding a handle into $\mathcal{K}$. This fact leads to the characterization of *simple tetrahedra* according to this theory, i.e., tetrahedra whose insertion (removal) does not change the homologies of a simplicial complex. They are our 0-handles. Therefore, in order to decide if a tetrahedron is simple it is sufficient to analyze its intersection with the simplicial complex to which it will be added. It is worth noting that the characterization of simple tetrahedra given by Proposition 4.1 is more straightforward than that presented in the work by Saha and Majumder [17], once it depicts all possible cases of this kind of tetrahedron.

The third consequence of Proposition 4.1 is another important fact: an $H_0H_1H_2$-handle can be added into a simplicial complex without altering the ranks of the homology groups, i.e., an $H_0H_1H_2$-handle can replace a hole for another one, not modifying the rank of H_1. In that respect, sometimes an $H_0H_1H_2$-handle could be mistaken for a simple tetrahedron. However, this handle does change the homology, although, in some cases, to one with the same rank (see Figure 3).

Finally, Proposition 4.1 is also the basis for table 2 below, where the first column indicates the type of handle (in accordance with Figure 1), the second column displays the group in which the handle is contained and the third column shows the possible changes that such handle can produce in the number of connected components (ν_0), holes (ν_1) and cavities (ν_2) of a simplicial complex $\mathcal{K}$. The additional columns refer to the operators needed to realize the handle.

Table 2 presents all possible changes that the addition of a new tetrahedron can produce in a simplicial complex $\mathcal{K}$. At the time of writing, this is the most

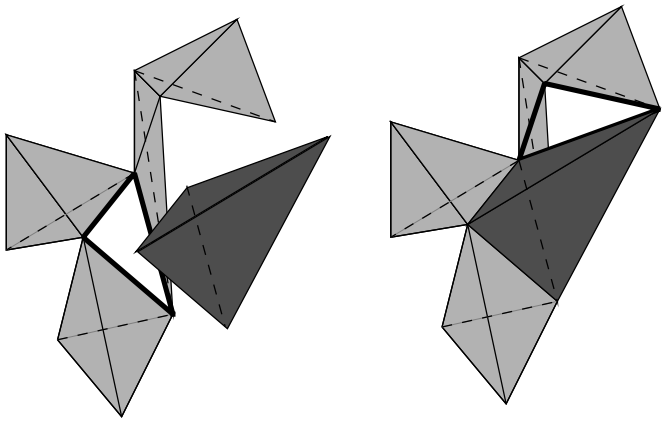


Figure 3: $H_0H_1H_2$-handle changing the homology to one with the same ranks. In this case it filled a hole and opened another one

complete characterization of tetrahedra described in the literature. Under this theory, a set of construction operators (seen in Table 2), and their inverses, can be derived. They are discussed in the next section.

4.2 Tetrahedral Morse Operators

From table 2 above we can define a set of topological operators we shall call *tetrahedral Morse operators* (TMO's). As we are going to show, these Morse operators make it possible to add new tetrahedra into a simplicial complex while keeping control of the number of connected components, of holes, and of cavities. Morse operators enable a more robust handling of the incidence and adjacency relationships in a tetrahedral mesh, considering that all the elements affected by the addition of a new tetrahedron are completely specified for each operator.

Tetrahedral Morse operators are defined in a straightforward manner based on table 2. The operators are grouped into seven different sets according to the homological change they introduce in the simplicial complex.

The seven sets with their respective operators are:

TMO_0 = {MV, ME, M2E, M3E, MF, M2F, M3F, MEF}
TMO_{H_0} = {MC, M2VKC, M3VK2C, M4VK3C, MVEKC, M2VEK2C, MV2EKC, M2EKC, MVFKC}
TMO_{H_1} = {M2VH, M3V2H, M4V3H, MVEH, M2VE2H, MV2EH, M2EH, M3EKH, M4EKH, M5EK2H, M6EK3H, MVFH, M2EFKH, M3EFK2H, ME2FKH, MV3EHKH}
TMO_{H_2} = {M3EP, M4EP, M5E2P, M6E3P, M2EFP, M3EF2P, ME2FP, M4FKP}
$\text{TMO}_{H_0H_1}$ = {M3VHKC, M4VHK2C, M4V2HKC, M2VEHKC, MV3EKCH}
$\text{TMO}_{H_0H_2}$ = {MV3EPKC}
$\text{TMO}_{H_1H_2}$ = {MV3EHP, M5EPKH, M6EPK2H, M6E2PKH, M3EFPKH}

Handle	Equivalence Class	(ν_0, ν_1, ν_2)	Operator
0	H_0-handle	$(1, 0, 0)$	MC
1	0-handle	$(0, 0, 0)$	MV
2	H_0H_1-handle	$(-1, 0, 0)$	M2VKC
		$(0, 1, 0)$	M2VH
3	H_0H_1-handle	$(-2, 0, 0)$	M3VK2C
		$(-1, 1, 0)$	M3VHKC
		$(0, 2, 0)$	M3V2H
4	H_0H_1-handle	$(-3, 0, 0)$	M4VK3C
		$(-2, 1, 0)$	M4VHK2C
		$(-1, 2, 0)$	M4V2HKC
		$(0, 3, 0)$	M4V3H
5	0-handle	$(0, 0, 0)$	M2E
6	0-handle	$(0, 0, 0)$	ME
7	H_0H_1-handle	$(-1, 0, 0)$	MVEKC
		$(0, 1, 0)$	MVEH
8	H_0H_1-handle	$(-2, 0, 0)$	M2VEK2C
		$(-1, 1, 0)$	M2VEHKC
		$(0, 2, 0)$	M2VE2H
9	H_0H_1-handle	$(-1, 0, 0)$	MV2EKC
		$(0, 1, 0)$	MV2EH
10	H_0H_1-handle	$(-1, 0, 0)$	M2EKC
		$(0, 1, 0)$	M2EH
11	0-handle	$(0, 0, 0)$	M3E
12	0-handle	$(0, 0, 0)$	M3E
13	$H_0H_1H_2$-handle	$(-1, -1, 0)$	MV3EKCH
		$(-1, 0, 1)$	MV3EPKC
		$(0, 0, 0)$	MV3EHKH
		$(0, 1, 1)$	MV3EHP
14	H_1H_2-handle	$(0, -1, 0)$	M3EKH
		$(0, 0, 1)$	M3EP
15	H_1H_2-handle	$(0, -1, 0)$	M4EKH
		$(0, 0, 1)$	M4EP
16	H_1H_2-handle	$(0, -1, 0)$	M4EKH
		$(0, 0, 1)$	M4EP
17	H_1H_2-handle	$(0, -2, 0)$	M5EK2H
		$(0, -1, 1)$	M5EPKH
		$(0, 0, 2)$	M5E2P
18	H_1H_2-handle	$(0, -3, 0)$	M6EK3H
		$(0, -2, 1)$	M6EPK2H
		$(0, -1, 2)$	M6E2PKH
		$(0, 0, 3)$	M6E3P
19	0-handle	$(0, 0, 0)$	MF
20	H_0H_1-handle	$(-1, 0, 0)$	MVFKC
		$(0, 1, 0)$	MVFH
21	0-handle	$(0, 0, 0)$	MEF
22	H_1H_2-handle	$(0, -1, 0)$	M2EFKH
		$(0, 0, 1)$	M2EFP
23	H_1H_2-handle	$(0, -2, 0)$	M3EFK2H
		$(0, -1, 1)$	M3EFPKH
		$(0, 0, 2)$	M3EF2P
24	0-handle	$(0, 0, 0)$	M2F
25	H_1H_2-handle	$(0, -1, 0)$	ME2FKH
		$(0, 0, 1)$	ME2FP
26	0-handle	$(0, 0, 0)$	M3F
27	H_2-handle	$(0, 0, -1)$	M4FKP

Table 2: Handles (operators) and their relation with the homology of the simplicial complex.

The types of homological changes caused by these classes of operators are identified by the handles expressed in their names, that is :

TMO_0: Causes no homological changes in the complex.
TMO_{H_0}: Causes change in H_0, that is, operators in this class change the number of connected components.
TMO_{H_1}: Causes change in the number of holes.
TMO_{H_2}: Causes change in the number of cavities.
$\text{TMO}_{H_0H_1}$: Causes change in the number of connected components and in the number of holes.
$\text{TMO}_{H_0H_2}$: Causes change in the number of connected components and in the number of cavities.
$\text{TMO}_{H_1H_2}$: Causes change in the number of holes and cavities.

A number of symbols are used to name the operators, and the convention for naming any particular operator provides a summary description of its action over the simplicial complex.

The seven symbols used for naming the operators are: M - make; K - kill; H - hole; P - cavity (pocket); C - component; V - vertex; E - edge; F - face. Numbers are utilized to describe the number of times that an action is executed in each entity (the number 1 is omitted in order keep the notation clean). An operator is identified by a sequence of symbols representing the actions on elements necessary to complete an operation of insertion of a tetrahedron.

During insertion of a tetrahedron, sometimes the action make (M in the sequence) refers to identification of the elements that already exist in the complex. This happens when the elements that follow are vertices, edges and faces (that is, when V,E or F follows the letter M). Other times M means that the elements in the sequence are actually added or caused in the simplicial complex. This occurs for holes, cavities and components (that is, when the symbols H, P and C follow the letter M). For instance, let's suppose that a tetrahedron is to be inserted in an existing simplicial complex, by gluing it to a single vertex of another tetrahedron already there. In this case, this addition is done through the identification of the vertex already there, to which the new tetrahedron must be glued (MV - make 1 vertex.) Figure 4 shows the action of the MV operator.

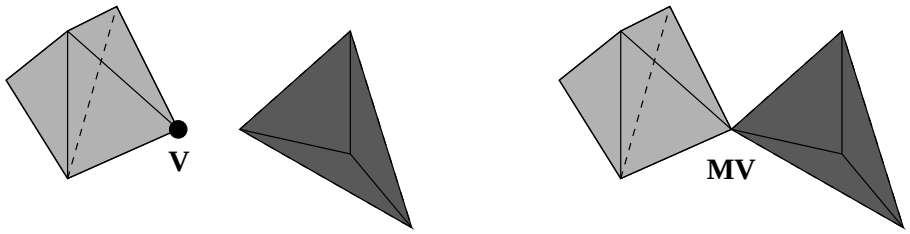


Figure 4: Tetrahedron insertion through the MV operator.

As further examples of the action of operators, let's take the symbol sequence MV2EH naming an operator in TMO_{H_1}. Its name indicates that, to add a tetrahedron, this operator makes (i.e., identifies) one vertex and two edges already in the complex (totalling three vertices), and adds a hole in the process (see Figure 5); the operator MV3EPKC in $\mathrm{TMO}_{H_0H_2}$ indicates that, to add a tetrahedron, it makes (i.e., identifies) one vertex and three edges (totalling four vertices), adds one cavity, and kills a component. Figure 2 given before has shown the execution of tetrahedra insertion through the operators M3EP (Figure 2a) and M3EKH (Figure 2b).

Note that the presence of symbols V and E means that the operator containing such symbols introduces singular vertices and edges into the simplicial complex. From this observation follows the next proposition.

Proposition (The addition of singularities during construction): It is not possible to generate a simplicial complex $\mathcal{K}$ with holes or cavities, by gluing tetrahedra, without adding singular vertices or edges (or both) to $\mathcal{K}$ during the process.

Proof In order to generate a hole in $\mathcal{K}$ it is necessary to add a new tetrahedron into $\mathcal{K}$ through an operator from TMO_{H_1}, $\mathrm{TMO}_{H_0H_1}$, or $\mathrm{TMO}_{H_1H_2}$. For a cavity, it is necessary to make use of a operator from TMO_{H_2}, $\mathrm{TMO}_{H_0H_2}$, or $\mathrm{TMO}_{H_1H_2}$. Since all operators of these classes introduce singular vertices or edges the proposition is proved.

Proposition 4.2 above has an important practical meaning regarding topological data structures, namely, that any data structure dedicated to represent

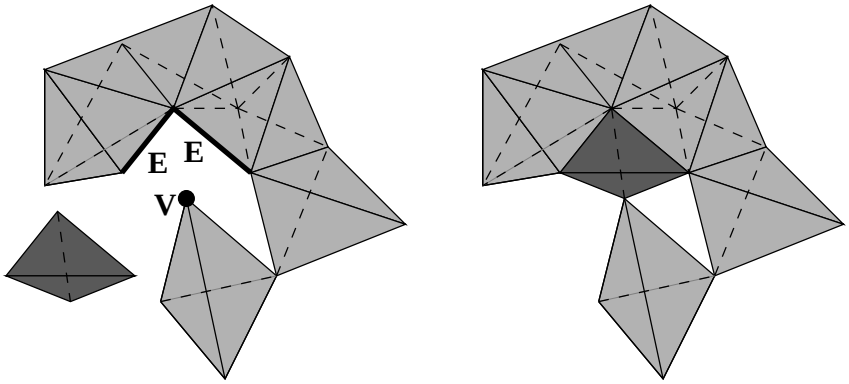


Figure 5: Tetrahedron insertion through MV2EH operator.

regularized simplicial complexes generated by gluing tetrahedra must be able to handle singularities in vertices and edges in order to be effective.

The above operators are called *direct operators*, as they may be used to add tetrahedra to an existing simplical complex. Operators for removing tetrahedra are described in terms of the *inverse operators*, which may be obtained by replacing letters M with K (and vice-versa) in the above description. The inverse operators can also be grouped into the same seven sets according to the homological change they introduce. The homological changes produced are also in the same classes of the direct operators.

We finish this section with a theoretical result about a minimal number of operators to build a tetrahedral mesh.

Proposition (The minimal set of operators): It is possible to build any simplicial complex by gluing tetrahedra using six operators (and their inverses).

Proof *Proof.* Let W be the submodule in $\mathbb{Z}^7$ whose elements (v, e, f, t, c, h, p) satisfy the equation $v - e + f - t - c + h - p = 0$. Note that any simplicial complex $\mathcal{K}$ may be represented as a vector in W and each Morse operator can also be described as a vector in W. For example, the operator MVFH may be represented by vector (0,3,3,1,0,1,0), meaning that it introduces zero new vertices, three new edges, three new faces, one new tetrahedron, zero components, one hole, and zero cavities into $\mathcal{K}$.

Let $x_1 = (1, 3, 3, 1, 0, 0, 0)$, $x_2 = (0, 1, 2, 1, 0, 0, 0)$, $x_3 = (0, 0, 1, 1, 0, 0, 0)$, $x_4 = (0, 3, 3, 1, 0, 1, 0)$, $x_5 = (1, 3, 4, 1, 0, 0, 1)$, and $x_6 = (4, 6, 4, 1, 1, 0, 0)$ be the vector representations of operators MF, M2F, M3F, MVFH, M3EP, and MC, respectively. A straightforward computation shows that $x_i, i = 1, \ldots, 6$ are linearly independent and that they are a basis for W, thus generating any vector contained in W. As any simplicial complex $\mathcal{K}$ is represented by a vector in W, theoretically these six operators constitute a minimal set of operators to generate $\mathcal{K}$.

Next section presents some important issues of implementation of TMOs.

5 Implementation and Computational Complexity

In this section we are concerned with the computational aspects of the Tetrahedral Morse Operators.

One issue that bears importance to all areas of Volume Modelling is that of Data Structure. In the case of our computational representation, one particular data structure, called SHF (Singular Half-Face) was developed. Its description and discussion are given in the next section.

The following section discusses other computational issues such as complexity and storage requirements.

5.1 Data Structure

The Data Structure named *Singular Handle-Face* (SHF) is capable of representing regularized simplicial complexes with singular vertices and edges.

A SHF data structure is organized in terms of seven explicitly represented entities (or nodes) which are:

- **Solid** - Representing each connected component of the regularized simplicial complex.
- **Cells** - Representing the tetrahedra .
- **Vertices** - Representing the vertices.
- **Half-Faces** - Representing the face contained in a cell.
- **Half-Edges** - Representing an edge contained in a Half-Face.
- **Boundary_Components** - Representing each boundary components.
- **Star_Vertex** - Representing each edge incident on a vertex.

Figure 6 shows the hierarchical relationship among the nodes of SHF.

Given a regularized simplicial complex $\mathcal{K}$, the solid, cells, and vertices nodes of SHF are linked lists representing each connected component, the tetrahedra, and the vertices of $\mathcal{K}$ respectively.

The nodes half-faces and half-edges are linked lists that store the simplices of dimension two and one contained in each tetrahedron. In other words, each tetrahedron is a cell containing its lists of half-faces and half-edges. The adjacency relationships among simplices are also stored in the nodes half-faces and half-edges. For example, each half-face "knows" the adjacent half-face in the neighbor tetrahedron, as shown in Figure 7a, and each half-edge knows the adjacent half-edges in its tetrahedron and in the neighbor tetrahedron, as in Figure 7b. If a half-face lies on the boundary of $\mathcal{K}$, its half-edges are employed to give access to the adjacency relationships in the boundary surface, as in a B-Rep representation. Figure 7c presents this schem.

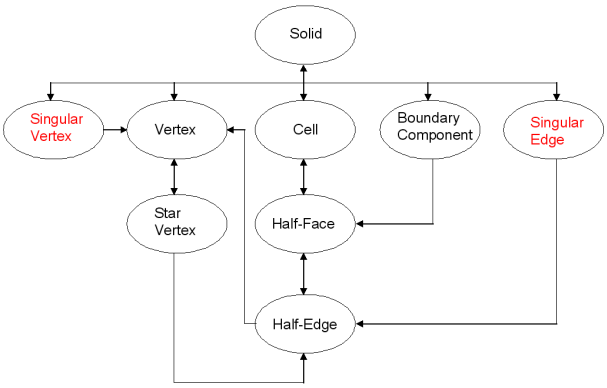


Figure 6: Hierarchical organization of SHF

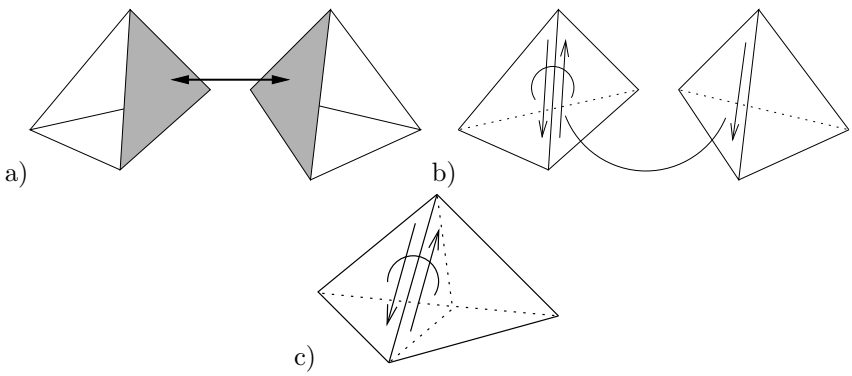


Figure 7: Adjacency relationship through Half-Face and Half-Edge nodes.

The Boundary Components node is a linked list containing a half-face of each boundary component. As the Tetrahedral Morse Operators act on the boundary of $\mathcal{K}$, it is important that SHF offers an efficient mechanism to give access to boundary components.

Each vertex in the SHF contains a Star Vertex node which is a linked list containing a half-edge of each edge incident to the vertex. The Singular Vertex and Singular Edge nodes could be replaced by an algorithm to find the edges but, in order to improve the performance, we represent them explicitly.

The realization of each TMO on a mesh has specific effects on the data structure that were implemented as actual operators on the data structure. This data structure was implemented using object-oriented concepts. The data structure and its application shown in section 6 were also implemented in the context of an open-source visualization system, the VTK (*The Visualization Toolkit*) [23]. The SHF is costly in terms of storage and was implemented this way to speed up testing of the operators. We are studying new data structure schemes so that storage is improved keeping performance at reasonable levels.

It is worth mentioning that explicit representation of boundary surfaces is a requirement that eases the implementation of TMO's, therefore, we should keep such a caracteristic in the future improvements.

5.2 Computational Complexity

Suppose that a new tetrahedron is to be added in a regularized simplicial complex $\mathcal{K}$ represented by the SHF data structure. Also suppose that the position where the new tetrahedron must be added has already been found. As the computational cost to find the correct position of the new tetrahedron is dependent on the application, we do not take it into account. For example, if the tetrahedra of $\mathcal{K}$ are given by vertex coordinates, for each new tetrahedron it is necessary perform a search in the boundary surfaces to find the position where to insert it. However, in some situations, as in the example given in the next section, the position of the tetrahedron is given by the grid location in $\mathbb{R}^3$.

The position of the tetrahedron defines the type of handle that it needs to be added. From the handle we decide which operator must be employed through local and global searches (note from table 2 that a same handle can give rise to different operators). As the SHF data structure maintains the local adjacency and connected components explicitly represented, only local searches are necessary to decide the operators from the 0-handle, H_0-handle, and H_0H_1-handle.

In order to choose the operators from the sets H_1H_2-handle and $H_0H_1H_2$-handle it is necessary make use of global searches, i.e., traverse the boundary surfaces and verify if new cavities are being created. That way, the cost of such operators is proportional to the number of involved boundary faces. They are the most expensive yet the less used of all operators.

6 Volumetric Reconstruction

Aiming at illustrating the applicability of tetrahedra characterization, we present in this section an application of TMO in volumetric reconstruction from a sequence of images.

By using TMO's, the resulting reconstruction algorithm represents a first step towards a one-pass procedure for 3D reconstruction from images, which will overpass the segmentation process. As it is, it empowers thresholding (the only pre-processing step required), by offering a way of defining 3D objects from data sets without the expensive 2D object detection step involved in most reconstruction methods.

Reconstruction starts from bi-dimensional images taken from measurement devices (such as MRI) or other scalar data sets representing consecutive data planes in 3D space. The sequence of images compound a regular 3D grid of cuboid cells, such as illustrated in Figure 8. Each pixel value is stored at a cell vertex. Each cuboid is interpreted as a set of 6 tetrahedra, adjusted to fill out

its volume unit. Tetrahedral adjustment inside a volume unit is illustrated in Figure 9.

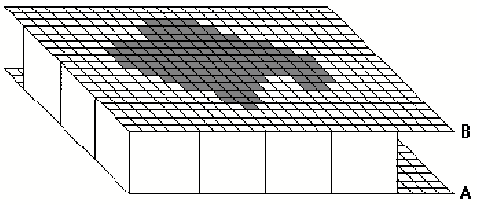


Figure 8: Volume cell organization from images.

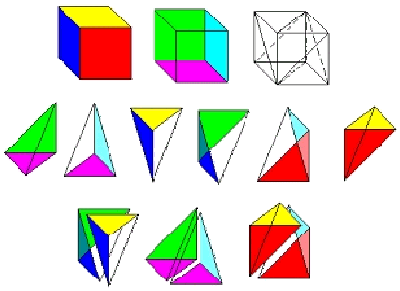


Figure 9: Tetrahedra positioning inside a cell.

Given a range of values indicative of objects of interest inside the regular grid, the reconstruction algorithm checks for the presence of these values at each vertex of the volumetric grid. Tetrahedra inside the volume unit whose vertices checked 'yes' for the presence of values of interest are candidates to be added to the object model. Two possible approaches were taken for the selection of tetrahedra to be included. In one of them all tetrahedra with one 'yes' vertex are added. In the second approach, an average of the vertices values is taken and compared with the values of interest. The average approach produces smoother models.

Figure 10 shows the reconstruction of a cashew nut from MRI images taken 2 mm apart after smoothing. The holes formed in this picture can be seen in the input images presented in Figure 11. The larger 'hole' is actually the nut core, and is of interest for analysis of the object. Apart from these holes, there are many other, very small holes, present in the original images due to noise generated by the data collection process. Those were also handled but are not visible due to their size.

For the reconstruction illustrated in Figure 10, table 3 presents the operators used and their frequency.

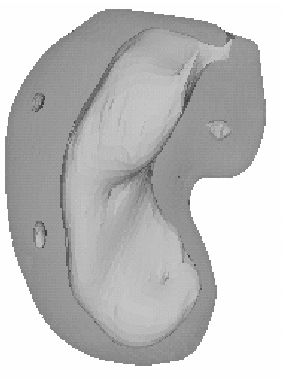


Figure 10: Cashew Nut Reconstruction with four holes.

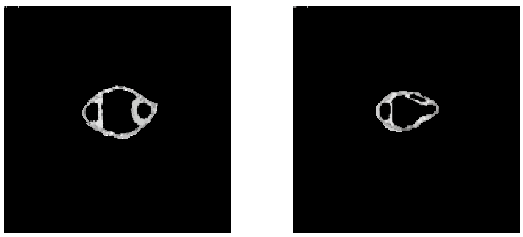
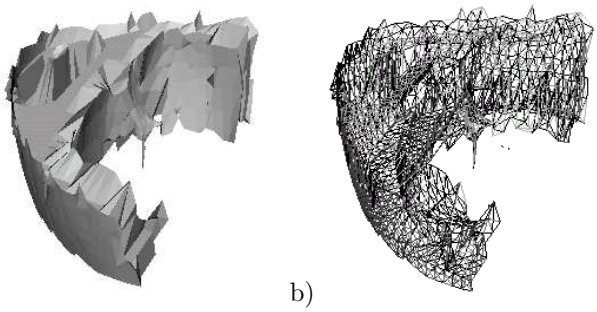


Figure 11: Cashew nut images before reconstruction.

Figure 12 presents a cut of small number of slices from the cashew nut reconstruction, which illustrates the tetrahedral mesh as well as some of the small details generated during the reconstruction process.



a) b)

Figure 12: A slice of the tetrahedral mesh for the cashew nut reconstruction a) surface display; b) wireframe display.

Due to the complete control of the topological structure of the constructed objects exerted by the modelling TMOs, during reconstruction the holes formed are registered so that they can be recovered later for analysis and other purposes.

Operator	Usage	Operator	Usage
MC	22	$M2VEH$	2
MF	8886	$M3F$	16141
MV	99	$M2F$	17295
ME	1179	$M2E$	652
$M2EH$	31	$M3EP$	4
$M3EKH$	2	$M3E$	11
$M3V2H$	5	$M3VHKC$	4
$M2VKC$	7	$MVFH$	4
MEF	9596	$ME2FP$	2
$ME2FKH$	80	$M2EFP$	10
$M2EFKH$	165	$MVFKC$	1
$MVEH$	140	$MVEKC$	5
$M2VH$	56		

Table 3: Operators used to reconstruct the cashew nut and their frequency of use

Even those lost visually due to their small size are registered because of the capabilities of the method, and can be easily recovered due to the organization of the data structure.

A procedure to handle cavities was also developed, that fills out any cavities the user chooses not to consider as part of the object. In Figure 10, for instance, from the four visible cavity components, the larger one is actually the cavity formed by the nut core, while the others are not useful objects. After running the filling out procedure it is possible to eliminate any number of them. Figure 13 shows the same model after elimination of the smaller elements from Figure 10 together with all the other cavities formed due to noise in the original images.

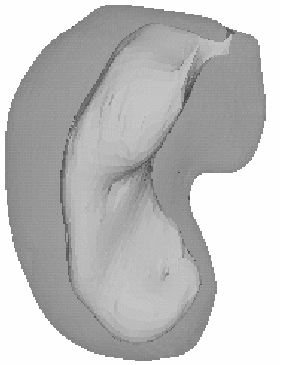


Figure 13: Cashew Nut after elimination of the three smaller holes.

Figures 14 to 16 show another example of the usability of the approach for

volumetric reconstruction from images.

Figures 14a and 14b display the reconstruction of head and brain from a set of 109 MRI images. Figure 14c shows slice 60 of this set. The figures show the surface boundary of the volumetric model of the head, whereas the brain is actually reconstructed as a cavity inside the head.

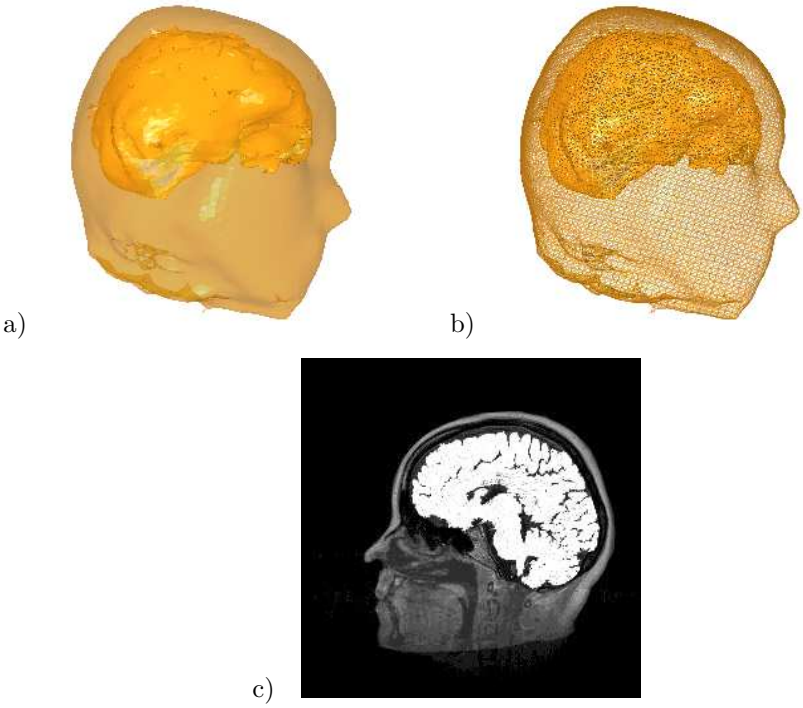


Figure 14: Head Reconstruction from MRI Images. a) Head is reconstructed having the brain as a cavity. b) Wire frame model of the boundaries. c) Sample image.

The initial head reconstruction picked up various noisy elements as cavities (seen in Figure 15a). As each one is detected (and indexed) during reconstruction, the elimination of the unwanted cavities produces a 'clean' model of the brain (Figure 15b).

Another important aspect of this approach is the identification of holes during object construction. They are recognized through the operator used at the time the hole is generated. In model construction activities, locating holes is usually a very costly procedure, demanding extensive search in the mesh, first to calculate the homology (and count the number of holes), then to be able to actually find them for further processing. In the method presented here, however,

a)

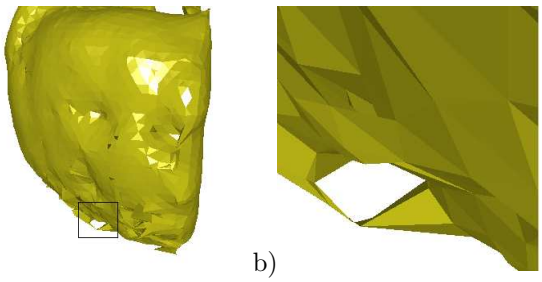
b)

Figure 15: a) Cavities detected during reconstruction of the head of Figure 14. b) Brain model after elimination of all other cavities in the model.

because we know both the homology and the 'moment' the hole is generated, the number of holes is known at the end of the process. Additionally, if these holes are to be handled, it is also possible to tag the use of hole generating operators so that the region of each hole is 'marked', easing further processing. One such hole in the brain model illustrated above can be seen in Figure 16.

7 Discussion and Conclusions

This paper presents a body of techniques to handle Volumetric Modelling via tetrahedral meshes. This is done through the deployment of mathematical tools from the field of topology, with which it is possible to characterize tetrahedral meshes completely. This gives rise to a number of operators capable of creat-



a) b)

Figure 16: a) One of the holes in the brain model detected during reconstruction. b) Scaled view of the hole.

ing and updating any non-manifold regularized simplicial complex, keeping the global topology of the object under control. Any structure created or deleted in the whole mesh due to the addition (or removal) of a tetrahedron is predicted by the operator used during that operation. The type of homological change is also registered by the operator class.

Thus, due to this strict topological control, at any time it is possible to tell how many holes, cavities and connected components there are in an object under construction. With the use of proper data structure (such as the one also presented here) it is also possible to tell where those elements are and handle them adequately (e. g. filling out the undesirable cavities).

Many applications can make use of this framework. In this paper it was given as examples reconstructions of a cashew nut and a human head from planar MRI images. In the case of the cashew nut, the interior of the object must be modelled, once the application needs to simulate forces on the shell during post-harvest processing of the nut. In cases such as this, where the interior of the object is to be modelled in a non-uniform way (for visual or simulation reasons), the tools presented here can support many types of processing that would be difficult (or too slow) otherwise.

For the reconstruction case, we also illustrated that the TMO's lend themselves to modelling the interior of objects via a one-pass procedure, in contrast with other volumetric modelling tools. Here it is not necessary to reconstruct the border first, or to generate the object's contours, and then fill out the internal parts with tetrahedra, to obtain the model of the object's interior. This capability of the reconstruction algorithm is a direct consequence of the topological control exerted by the method, that is, noisy values in the data set are usually detected as topological entities that can be easily labelled and eliminated. This feature is not present in the conventional reconstruction algorithms. This is a step towards 3D reconstruction without segmentation.

The main advantage of the method for modelling is its topological control. Handling topology instead of geometry improves robustness of computational procedures. Additionally, the topological control allows indexing of important features. This effort should support volume modelling of objects through tetra-

hedra in an integrated way, so that objects can be handled in many stage of the visualization process (from simulation to interaction) without the need for change in representation. For instance, tracking changes and their effects during interaction should be directly supported by these tools.

TMO's are being extended to handle voxel meshes, taking as basis a technique developed for 2D digital surfaces by the authors [24].

Acknowledgments

This work was sponsored by FAPESP - State of São Paulo Research Funding Agency - Brazil (proc. 03/02815-0 and 00/03385-0) and CNPq - National Counsel for Research - Brazil (proc. 300531/99-0). Fruit images were provided by Embrapa - Brazilian Agricultural Research Corporation - São Carlos - Brazil. We also acknowledge the work of our student Carlos Andre Sanches de Souza.

References

[1] G. T. Herman. Geometry of digital spaces. *Birkhäuser Boston*, 1998.

[2] G. M. Nielson. Volume Modeling. In: *Volume Graphics*, M. Chen et al. (eds.), Springer, pp. 29-48, 2000.

[3] J. C. Hart. Morse theory for computer graphics. Technical Report EECS-97-002, Washington State University, 1997.

[4] M. Mäntylä. An introduction to solid modeling, *Computer Science Press, 1988.*

[5] L.G. Nonato, R. Minghim, M.C.F. Oliveira, and G. Tavares. A Novel Approach for Delaunay 3D Reconstruction with a Comparative Analysis in the Light of Applications. Computer Graphics Forum, 20:(2), pp. 161-174, 2001.

[6] A. Guezic and R. Hummel. Exploiting triangulated isosurface extraction using tetrahedral decompostion. IEEE Trans. on Visualization and Computer Graphics, 1:(4), pp. 328-342, 1995.

[7] M. Bern and D. Eppstein. Mesh Generation and optimal triangulation. In: Computing in Euclidean Geometry, F. K. Hwang and D.-Z. Du (eds.), World Scientific, pp. 47-123, 1992.

[8] Teng S-H. and Wong C.W. Unstructured Mesh Generation: Theory, Practice, and Perspectives. International Journal of Computational Geometry & Applications, 10(3):227-266, 2000.

[9] S. Fourey and R. Malgouyres. Intersection number and topology preservation within digital surfaces. Theor. Comput. Sci., 283:(1), pp. 109-150, 2002.

[10] J. Cox, D.B. Karron, N. Ferdous. Topological Zone Organization of Scalar Volume Data. Journal of Mathematical Imaging and Vision, 18:(2), pp. 95-117, 2003.

[11] T. K. Dey, H. Edelsbrunner and S. Guha. Computational Topology. Advances in Discrete and Computational Topology, B. Chazelle, J. E. Goodman and R. Pollack (eds.), Contemporary Math., AMS, 1998.

[12] A. Vangelder and J. Wilhelms. Topological considerations in isosurface generation. ACM Transactions on Graphics, 13:(4), pp. 337-375, 1994.

[13] G. M. Treece, R. W. Prager, and , A. H. Gee. Regularised marching tetrahedra: improved iso-surface extraction. Computers & Graphics-UK, 23:(4), pp. 583-598, 1999.

[14] Y. Livnat, H. W. Shen, and C. R. Johnson. A near optimal isosurface extraction algorithm using the span space. IEEE Transactions on Visualization and Gomputer Graphics, 2:(1), pp. 73-84, 1996.

[15] H. Edelsbrunner. Geometry and topology for mesh generation. Cambridge University Press, New York, 2001.

[16] F. Ganovelli, P. Cignoni, C. Montani, and R. Scopigno. A multiresolution model for soft objects supporting interactive cuts and lacerations. Computer Graphics Forum, 19:(3), pp. C271-C281, 2000.

[17] P.K. Saha and D. D. Majumder. Local topological parameters in a tetrahedral representation. Graphical Models and Image Processing, 60, pp. 423-436, 1998.

[18] S. Gumhold, S. Guthe, and W. Strasser. Tetrahedral mesh compression with the cut-border machine. Proceedings of Visualization'99, pp. 51-58, 1999.

[19] A. Castelo, H. Lopes, and G. Tavares. A handlebody representation for surfaces and Morse operators. Proc. Curves and Surf. in Comp. Vis. and Graph. III, SPIE, pp. 270-283, 1992.

[20] H. Lopes and G. Tavares. Structural operators for modeling 3-manifolds. Proceedings of the fourth ACM Sympos. on Solid Modeling and applications, pp. 10-18, 1997.

[21] S. Pesco, G. Tavares and H. Lopes. A stratification approach for modeling two-dimensional cell complexes. Computers & Graphics, 28:(2), pp. 235-247, 2004.

[22] J. R. Munkres. Elements of algebraic topology. Addison Wesley, 1984.

[23] W. Schroeder, K. Martin, and Bill Lorensen. The Visualization Toolkit, 3rd. Edition, Kitware, 2002.

[24] L.G. Nonato, A. Castelo Filho, R. Minghim, and J.E.Batista Neto. Morse Operators for Digital Planar Surfaces and their Application to Image Segmentation. IEEE Transactions on Image Processing, 13:(2), pp. 216-227, 2003.