An Extensible Framework for Modeling Simplicial Complexes

David Canino* Leila De Floriani†

Department of Computer Science, Universitá degli Studi di Genova, Genova, Italy

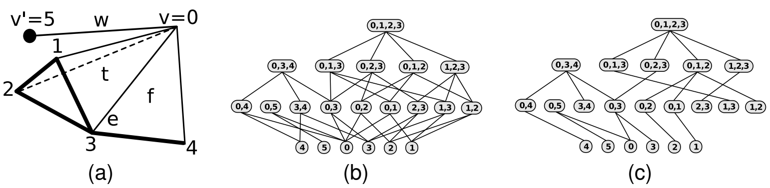

Figure 1: (a) A simplicial 3-complex, and the graph-based representation of the immediate (b) boundary relations $\mathscr{R}_{p,p-1}$ and (c) partial co-boundary relations $\mathscr{R}^*_{p,p+1}$ among its simplices, which are encoded in the *Incidence Simplicial (IS)* data structure [7] . We call them *mangroves*, and they are the basis of our *Mangrove Topological Data Structure (Mangrove TDS)* framework.

Abstract

We introduce the *Mangrove Topological Data Structure (Mangrove TDS)* framework for modeling simplicial complexes. It is based on a graph-based representation of the data structures, called *mangroves*, which ensures an *extensible* representation of a data structure for simplicial complexes. Mangroves can be easily *customized* for any modeling need, including the efficient representation of non-manifold shapes, and of those simplices, not directly encoded in a mangrove, that we call *ghost simplices*. We discuss here the properties of this framework, and current and future developments.

1 Introduction

Simplicial complexes are extensively used to discretize digital shapes in many applications, including computer graphics, solid modeling, finite element analysis and simulation, scientific visualization, and geographic data processing. Simplicial complexes allow modeling non-manifold shapes containing parts of different dimensions, and not necessarily embedded in the 3D Euclidean space. Informally, a *manifold* is a connected subset of the Euclidean space such that every neighborhood of any of its points is homeomorphic to an open ball. Shapes, which do not satisfy this property, are usually called *non-manifold*.

In the literature, a large variety of topological data structures have been proposed for cell and simplicial complexes [6]. Such data structures are formalized through *topological relations*, which capture the connectivity information of the cells (or simplices) in the complex they represent. They are classified on the basis of their *domain* (i.e., manifold or non-manifold) and of the *dimension* for the complex. In particular, there are representations, designed for complexes of a specific dimension, and with a specific embedding space, usually $\mathscr{E}^3$. Finally, there are *incidence-based* data structures, which encode all the cells of the complex and a subset of their incidence relations, and *adjacency-based* data structures, which encode only vertices and top cells, e.g., cells which are not on the boundary of other cells, plus a subset of adjacency relations.

There is the need of a generic framework for the fast design and prototyping of topological data structures in order to perform quantitative comparisons of storage costs and efficiency in answering queries and performing updates, and to be able to simplify the development of any data structure without rewriting it from scratch. According to [13], a framework for the fast prototyping of topological data structures for simplicial or cell complexes should allow replacing the internal representation of the complex so as selecting the most suitable and *efficient* one for a specific task, i.e., it must provide a *flexible* representation. Thus, a topological data structure should be a *dynamic plugin* to be used without modifying the system internally. In addition, it should be as *simple* as possible to use and extend the system with a short learning curve. Another important issue when dealing with representations for non-manifold shapes is the ability to detect non-manifold singularities efficiently. Note that recognizing whether a cell is a non-manifold singularity is decidable only for d-complexes, with $d \leq 6$ [11].

2 The Mangrove TDS Framework

The *Mangrove Topological Data Structure (Mangrove TDS)* framework is a tool for the fast prototyping of topological data structures encoding simplicial complexes. This framework is one of the first tools, which completely satisfy the three design goals, discussed above. The Mangrove TDS framework describes data structures for a simplicial d-complex Σ as a graph-based representation, which we call a *mangrove*. A node n_σ in the mangrove corresponds to a simplex σ in Σ, and an arc $(n_\sigma, n_{\sigma'})$ describes a topological relation between σ and σ'. Thus, a topological data structure is described by a set of nodes (e.g., simplices directly encoded) and by a set of arcs (e.g., topological relations, restricted to these simplices). For each simplex σ, we encode only the endpoints of the arcs outgoing from n_σ. We say that a mangrove is *global* if all the simplices in Σ are directly encoded, otherwise it is *restricted*. Figures 1(b) and 1(c) show, respectively, the graph-based representation of the topological relations in the *Incidence Simplicial (IS)* data structure [7]. The IS data structure encodes, for each p-simplex σ, boundary relation $\mathscr{R}_{p,p-1}(\sigma)$, i.e., $(p-1)$-faces of σ, and partial co-boundary relation $\mathscr{R}^*_{p,p+1}(\sigma)$, i.e., one $(p+1)$-simplex for each connected component in the link of σ.

A mangrove provides a generic representation of any topological data structure without restrictions on the type and dimension of the complex. The key idea of our approach consists of customizing the content of a mangrove in order to encode the specific data structure. In this way, the internal representation of a complex is *extensible*, and can be dynamically replaced in order to choose the most *efficient* one for a specific task. The answers to most of the queries are obtained through breadth-first traversals of the graph describing the data structure. This makes our system *easy* to use and to extend. Currently, we have implemented six data structures [3], among which three are dimension-independent, namely

*e-mail: canino@disi.unige.it

†e-mail: deflo@disi.unige.it

Table 1: Properties of systems for modeling cell and simplicial complexes.

	OpenMesh [1]	*OpenVolumeMesh* [9]	*VCGLib* [14]	*CGAL* [5]	*Mangrove TDS*
Type of Complexes	cell	cell	simplicial	any	simplicial
Dimension of Complexes	2	up to 3	up to 3	any	any
Internal representation	incidence-based	incidence-based	adjacency-based	several	any
Flexible representation	no	no	no	yes (modules)	yes (plugins)
Recognition of non-manifolds	only at vertices	not efficient	complete	complete	efficient

the Incidence Graph [8], the IS data structure and the *Generalized Indexed Data Structure with Adjacencies (IA*)* [4], a compact extension of the *Indexed data structure with Adjacencies (IA)* [6] for non-manifolds. These two latter data structures support the efficient recognition of non-manifold singularities (see [3] for further details). Note that the IS data structure is described by a global mangrove, while the IA^* data structure by a restricted one.

The Mangrove TDS framework provides an implicit representation of simplices not directly encoded in restricted mangroves, which we call the *ghost simplices*. In a restricted mangrove, any p-simplex σ is either a top p-simplex, or the i-th p-face of the k-th m-simplex σ' in Σ, with $p < m$. Thus, a ghost simplex σ is implicitly represented as a tuple (m,k,p,i), which is a *GhostSimplexPointer* reference for σ. This reference is not unique, since several top simplices may be incident at σ. Hence, we improve the expressive power of restricted mangroves, allowing the attachment of attributes to all simplices, not just to vertices and top simplices. Thus, a restricted mangrove becomes basically equivalent to a global one, but with a reduced storage cost. Our tests show that *GhostSimplexPointer* references introduce a large speed-up for retrieving topological relations [3]. To the best of our experience, our framework is the first tool, which provides these references explicitly.

We have implemented topological queries for extracting all the possible topological relations in the complex as well as for detecting non-manifold singularities for all the data structures, mentioned above [3]. The implementation of the common infrastructure of our framework, and of all the data structures is contained in the *Mangrove Topological Data Structure (Mangrove TDS) Library* [2], released under a GPL license.

3 Discussion

As we have seen, a mangrove captures connectivity information provided by topological relations, and can be easily *customized* for any modeling need. To the best of our experience, the Mangrove TDS framework is one of the first tools, which efficiently represent non-manifold shapes, and those simplices, which are not directly encoded in a data structure (*ghost simplices*). Our framework is *extensible*, in the sense that it is possible to add new data structures without messing with the structure of the system.

A few frameworks with these properties have been proposed and implemented in the literature. Most of them exploit a *monolithic* representation of a shape, which cannot be dynamically replaced, unless to completely rewrite it [1, 14, 9], or change the current software module [5]. In [3] we have been shown that the internal representations of some frameworks [1, 9] are equivalent to data structures, like the *Half-edge* [10] and the *Incidence Graph* [8], which do not efficiently support the recognition of non-manifold singularities [6]. In addition, some frameworks [1, 14, 9] can manipulate only 2- and 3-complexes. Table 1 summarizes properties of some of these frameworks. Clearly, our framework offers a fair solution to some of these drawbacks in the existing frameworks.

Our current implementation of the Mangrove TDS Library [2] does not support editing operators on mangroves. We are developing editing operators, like the *Vertex-Pair Collapse (VPC)* [12], to enable topological modifications of simplicial complexes, which allows defining multi-resolution models for non-manifold shapes. We are also planning to develop editing operators, like stellar and bistellar operators, which preserve the homology of the underlying shape (e.g., its Betti numbers). Such operators are useful to improve efficiency of homology computation on simplicial shapes.

In our future work, we plan to extend our framework to deal with structured quadrilateral and hexahedral meshes, due to their increasing relevance in geometry processing, animation, and numerical simulations. The IS and IA* data structures can be easily extended to quad and hexahedral meshes, since boundary relations are constant on such meshes as well, and this assumption is what makes such data structures compact in the case of simplicial complexes. Finally, we are also planning to exploit our framework for applications to shape analysis and reconstruction in high dimensions, due to the compactness of some representations, like the IA^* data structure.

Acknowledgements

This work has been supported by the Italian Ministry of Education and Research under the PRIN 2009 program, and by the National Science Foundation under grant number IIS-1116747.

References

[1] M. Botsch, S. Steinberg, S. Bischoff, and L. Kobbelt. OpenMesh: a Generic and Efficient Polygon Mesh Data Structure. In *Proc. of the OpenSG Symposium*, 2002.

[2] D. Canino.The *Mangrove Topological Data Structure (Mangrove TDS) Library*, 2012. http://mangrovetds.sourceforge.net.

[3] D. Canino. *Tools for Modeling and Analysis of Non-manifold Shapes*. PhD thesis, University of Genova, Italy, 2012.

[4] D. Canino, L. De Floriani, and K. Weiss. IA^*: an Adjacency-based Representation for Non-Manifold Simplicial Shapes in Arbitrary Dimensions. *Computer & Graphics*, 35(3):747–753, 2011.

[5] *Computational Geometry Algorithms Library (CGAL)*, 2011. http://www.cgal.org.

[6] L. De Floriani and A. Hui. Data Structures for Simplicial Complexes: an Analysis and a Comparison. In *Proc. of the Symposium on Geometry Processing*, pages 119–128. ACM Press, 2005.

[7] L. De Floriani, A. Hui, D. Panozzo, and D. Canino. A Dimension-independent Data Structure for Simplicial Complexes. In *Proc. of the International Meshing Roundtable*, pages 403–420. Springer, 2010.

[8] H. Edelsbrunner. *Algorithms in Combinatorial Geometry*. Springer, 1987.

[9] M. Kremer. The *OpenVolumeMesh Library*, 2012. http://www.openvolumemesh.org.

[10] M. Mantyla. *An Introduction to Solid Modeling*. Computer Science Press, 1987.

[11] A. Nabutovsky. Geometry of the Space of Triangulations of a Compact Manifold. *Comm. in Math. Phys.*, 181:303–330, 1996.

[12] J. Popovic and H. Hoppe. Progressive Simplicial Complexes. In *Proc. of the ACM SIGGRAPH*, pages 217–224, 1997.

[13] D. Sieger and M. Botsch. Design, Implementation, and Evaluation of the Surface_Mesh Data Structure. In *Proc. of the International Meshing Roundtable*, pages 533–550. Springer, 2011.

[14] The *Visual Computing Graphics Library (VCGLib)*, 2004. http://vcg.sourceforge.net.