A Course in Topology-based Geometric Modeling

Pascal Lienhardt, **Laurent Fuchs** and **Yves Bertrand**
IRCOM-SIC,
Poitiers, France
{lienhardt, fuchs, bertrand}@sic.sp2mi.univ-poitiers.fr

Abstract

The content of a course in Topology-based Geometric Modeling is presented. The key ideas are based upon simplicial notions issued from combinatorial topology. The well adequacy of these notions to geometric patches (such as Bézier patches) is pointed out. Topology and geometric notions are simultaneously presented in order to show their complementarity.
Keywords: *Geometric modeling, combinatorial topology, subdivisions, geometric patches.*

1. Introduction

The goals and the content of a course in Topology-based Geometric Modeling are described in this paper. Since several years, it is addressed to Computer Science students (Master Degree, ninth and tenth semester) who have acquired basic knowledge in computer graphics, but who did not thoroughly approach Geometric Modeling. Other courses specialized in other fields of Geometric Modeling can supplement this course, e.g. C.S.G. and implicit surfaces.

The purpose is the acquisition of solid basic concepts: so, we insist here about the fundamental concepts in mathematics and computer science, and mainly about :

- (combinatorial) topological notions related to subdivisions of geometric spaces,
- parametric curves and surfaces, and more generally curved shapes,
- the links that can be made between these two notions, especially combinatorial ones.

The design of data structures is considered as well as that of basic construction operations and algorithms for computing topological properties.

The course contains three main parts :

- simplicial notions, mainly simplicial sets and triangular patches ;
- simploidal notions, where cells are cartesian products of simplices ;
- cellular notions, where cellular structures are deduced from “numbered” simplicial sets and embedding is based upon parametric trimmed patches.

We insist here about the complementarity between the topological aspects (subdivision, etc) and the embedding ones (curves and surfaces). Highlighting the common combinatorial concepts facilitates this. Hence a global and homogeneous outline of the field is given.

2. Course content

2.1. Introduction (2h)

Basic concepts are introduced: subdivision of geometric space (i.e. partition into cells of various dimensions on which boundary relations are defined), various "classes" of subdivisions (simplicial or cellular ones, complexes, manifolds, ...), homeomorphism, etc.

This initiation is carried out by the means of a constructive approach of the classification of surface subdivisions into topological surfaces, as it can be found in [8]: the basic objects are polygons (topological discs) and surface subdivisions are built by identification of edges. The classification of surfaces (and the related characteristics: number of boundaries, orientability factor, genus) is approached by distinguishing the various cases of identification (the proof that any subdivision belongs to a unique topological surface is not tackled).

This part can by supplemented by a practical session during which the students, provided with paper, scissors and tape, build and classify paper surfaces.

2.2. Simplicial : Fundamentals (6h).

A first part shows how to deduce combinatorial notions from geometrical ones. We define successively:

- simplicial complexes, defined as convex hulls of linearly independent points,

- abstract simplicial complexes, where a simplex is a set of abstract vertices ; the *geometric realization* of an abstract simplicial complex is a simplicial complex (vertices are associated with points, ...),
- semi-simplicial sets [12], where a simplex is an abstract object and boundary operators associate its boundary simplices with the simplex (cf. the following definition and Figure 1). The geometric realization of a semi-simplicial set is a CW-complex.

Definition [12] **:** A d-dimensional *semi-simplicial set* K, is a union $K = \bigcup_{l=0}^{d} K_l$ of sets K_l of abstract simplices of dimension $l \le d$, together with boundary operators ∂_i such that:

$$\partial_i : K_l \to K_{l-1} \text{ with } l \ge 1 \text{ and } 0 \le i \le l,$$
$$\sigma\partial_i\partial_k = \sigma\partial_k\partial_{i-1} \text{ for } i > k \text{ and } \sigma \in K.$$

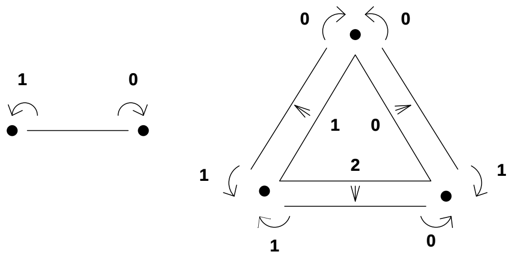

Figure 1. 1- and 2-dimensional simplices, and their boundaries.

Basic additional notions and construction operations are defined (e.g. geometric realization, isomorphism, simplex creation, identification) and traversal algorithms are described in order to compute topological properties (e.g. connectivity, boundary, star).

In a second part, the embedding notion is introduced in order to define the object shape. First, we show that a subclass of semi-simplicial sets is equivalent to abstract simplicial complexes ; so, they can be embedded as simplicial complexes, by associating points with 0-simplices.

Embedding is then generalized by taking curved shapes into account (i.e. curves, triangular surfaces, tetrahedral volumes, etc). These shapes are introduced by the "Bézier simplex" notion and described using the de Casteljau algorithm in any dimension. Bernstein polynomials are defined. Joining and continuity problems are taken into consideration. Figure 2 gives us an example of a semi-simplicial set.

Combinatorial links between an abstract simplex and a Bezier simplex are pointed out. Mainly it is possible to extract a semi-simplicial set structure from the control point structure (cf. Figure 3) : it is thus possible to associate subsets of control points with abstract simplices, and this can be used in order to design data structures.

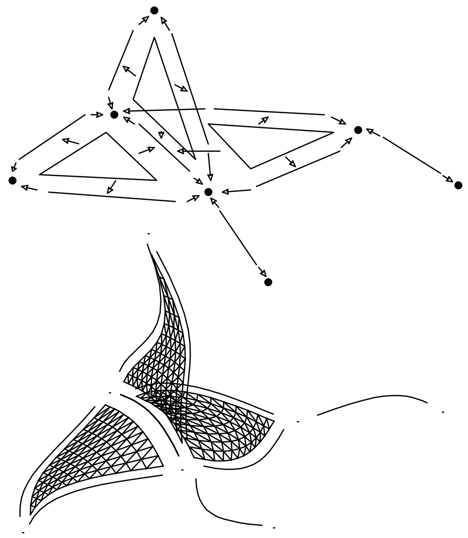

Figure 2. Example of semi-simplicial set, up the topological structure, down the embedding.

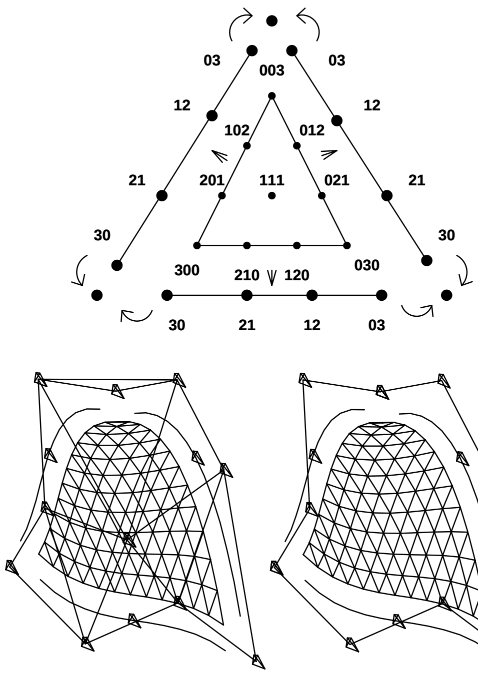

Figure 3. Semi-simplicial set structure and control point structure.

The last part is devoted to geometric operations (e.g. splitting, extrusion and cone). They are compositions of operations acting on semi-simplicial sets and on Bézier simplices.

Implementation is also addressed from abstract data types to concrete data structures using records and pointers. A basic concrete data structure can be directly deduced from the combinatorial definition of n-dimensional semi-simplicial sets.

```
struct cpl {
        (I+1)_SIMPLEX * s ;
        int b_op ;
        struct cpl * next, * prev ;
} * CPL ;

struct i_simplex {
        struct simplex * delta[i+1] ;
        CPL ante ;
} I_SIMPLEX ;
```

The record (written in pseudo-code) `I_IMPLEX` represents an i-simplex : the first field (`delta`) implements the boundary operators ; the second field implements the "inverse" operators : a doubly-linked list contains the simplices which have the considered simplex as image by a boundary operator.

From an abstract data type point of view, the simplicial sets are defined by the operations working on the data structure. Basic operations such as

```
set_border :
  nat simplex simplex simplexSet -> simplexSet

border : simplex nat -> simplex
```

which respectively define boundary operators and apply boundary operator to a simplex are defined by equations. Then fundamental properties are expressed, e.g.

```
border(k, border(i, s)) =
   border(i-1, border(k, s)) if i > k
```

Similar implementation mechanisms are used for the other simploidal and cellular notions presented in the next sections.

2.3. Simplicial (4h).

This part has three complementary goals:

- Extend semi-simplicial sets by defining simplicial sets [12] ; thus cartesian product can be defined. This operation is a generalization of the extrusion one; it is the combinatorial basis for the computation of Minkowski sums. Moreover, notions presented in section 2.4 are based upon cartesian product.
- Define basic mechanisms which make it possible to deduce optimized structures for some classes of simplicial sets (e.g. homogeneous, regular, quasi-manifolds), and related operations (i.e. constrained operations that are internal to sub-classes). Using these mechanisms, relations can be established with other structures defined in literature, and with cellular structures (see section 2.5) [4, 9, 11].
- Notions related to curved shapes are abstracted and generalized by introducing the general simplicial algorithms [6, 7]. This provides a uniform approach for the parametric curves, surfaces, volumes, etc.

2.4. Simploidal (4h).

Previous simplicial notions are generalized by defining structures that can represent subdivisions in which cells are cartesian products of simplices.

First, cubic sets are defined [13] : a cube is the cartesian product of 1-dimensional simplices.

> **Definition:** A d-dimensional *cubic set* K is the union $K = \bigcup_{l=0}^{d} K_l$ of sets K_l, which elements are abstract cubes of dimension $l \leq d$, together with boundary operators ∂_j^i such that:
>
> $$\partial_j^i : K_l \to K_{l-1} \text{ with } l \geq 1,\ j = 0{,}1 \text{ and } i \leq l,$$
> $$\sigma\partial_j^i\partial_l^k = \sigma\partial_l^k\partial_j^{i-1} \text{ for } i > k \text{ and } \sigma \in K.$$

Combinatorial links between Bézier patches (tensor product of Bézier curves) and abstract cubes are pointed out, leading to the embedding of cubic sets using cubic shapes.

Abstract simplices and abstract cubes are generalized by abstract simploids (the corresponding geometric notion is the simploid notion [3]). They are cartesian products of abstract simplices. An abstract simploid is fully defined by the k-tuple $(d_1, \ldots, d_k)$ of the dimensions of the k simplices involved in the product (cf. Figure 4).

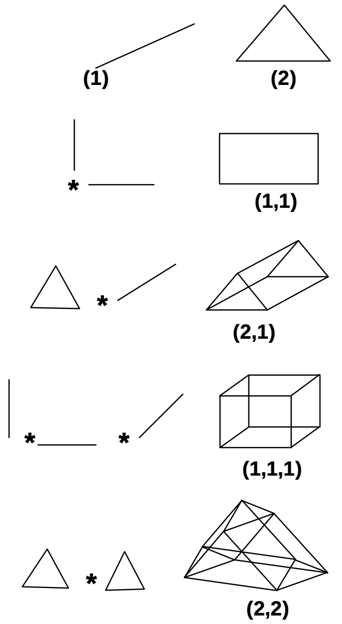

Figure 4. Example of simploids. The cartesian product of at least two simplices is graphically represented.

Then, simploidal sets are defined by (cf. Figure 5) :

Definition [7] **:** A d-dimensional simploidal set K is the union $K=\bigcup_{l=0}^{d} K_l$ of sets K_l, of abstract simploids of dimension $l \le d$, together with boundary operators ∂^i_j such that[1]:

$$(\ldots,d_i,\ldots)\partial^i_j = \begin{cases}(\ldots,d_i-1,\ldots) & \text{if } d_i>1\\ (\ldots,\hat{d}_i,\ldots) & \text{else}\end{cases}$$

$$(\ldots,d_i,\ldots)\partial^i_k\partial^i_l = (\ldots,d_i,\ldots)\partial^i_l\partial^i_{k-1} \text{ with } k>l \text{ and } d_i>1$$

$$(\ldots,d_i,\ldots,d_j,\ldots)\partial^j_k\partial^i_l = \begin{cases}(\ldots,d_i,\ldots,d_j,\ldots)\partial^i_l\partial^j_k & \text{if } d_i>1\\ (\ldots,d_i,\ldots,d_j,\ldots)\partial^i_l\partial^{j-1}_k & \text{else}\end{cases}$$

Products of general simplicial algorithms [6], generalizing cubic shapes, are also defined, and used for embedding simploidal sets. Once again, the complementarity of the embedding and the topological aspects are pointed out and used. For instance for the design of data structures, we also explain how common combinatorial notions simplify the implementation. Simplicial operations (and implementation aspects) are extended to the simploidal framework. Lastly, open research problems are highlighted.

[1] The hat notation means that the element is removed from the tuple.

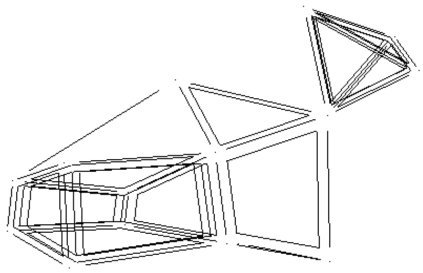

Figure 5. Example of simploidal set.

2.5. Cellular Structures (8h)

A first part shows how to deduce a first cellular structure from a subclass of simplicial sets [11]. Simplicial sets are structured into cells by associating numbers with 0-simplices (this is quite close to the classical notion of barycentric triangulation : numbers associated with the vertices of an *i*-simplex are {0, 1, ..., i}). Then, we define cellular quasi-manifolds as a sub-class of such "numbered" simplicial sets, and the equivalent notion of combinatorial generalized map is deduced by applying conversion mechanisms presented in section 2.2 (cf. Figure 6).

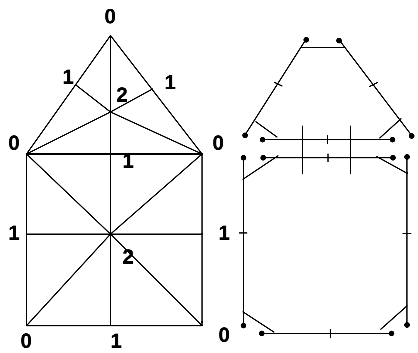

Figure 6. Numbered simplicial set and generalized map. A dart corresponds to a 2-dimensional simplex {0,1,2}. Application α_i joins simplices which share a 1-dimensional simplex {0,1,2}-{i}.

Definition: A d-dimensional *generalized map* (or G-map) is a set D of *darts* together with applications (cf. Figures 7,8,9) :

$$\alpha_k : D \to D \text{ with } 0 \le k \le d,$$
$$b\alpha_k^2 = b \quad \text{with } b \in D,$$
$$b\alpha_k\alpha_l = b\alpha_l\alpha_k \text{ with } k+1<l \le d.$$

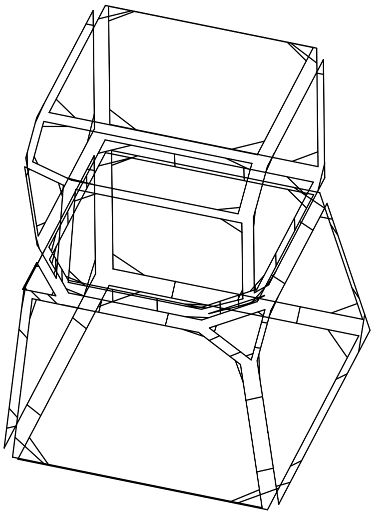

Figure 7. Example of a generalized map.

Construction operations are defined for handling generalized maps, and also algorithms for computing topological properties [1, 11] (e.g. connectivity, orientability; characteristics can be computed on 2-dimensional generalized maps providing their classification in the meaning of section 2.1).

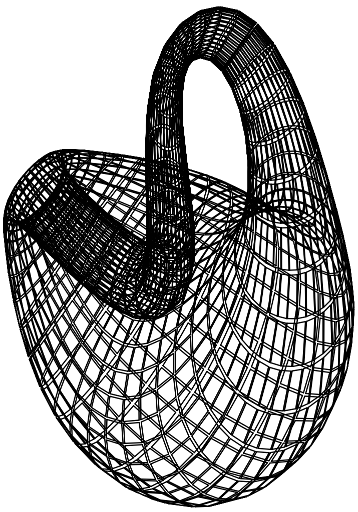

Figure 8. A subdivision of a Klein bottle whose topology is represented by a G-map ; its topological characteristics can be computed by simple traversal operations (i.e. number of boundaries = 0, orientability factor = 2, genus = 0).

A second part shows how to deduce cellular structures for other classes of "cellular objects" [4] (from combinatorial map chains for representing cellular complexes, to oriented combinatorial maps for representing oriented quasi-manifolds without boundaries). These conversions are obtained mainly by applying mechanisms presented in section 2.2. Relations with equivalent structures are also highlighted (e.g. winged-edge, quad-edge, radial-edge, facet-edge, cell-tuple structures, incidence graphs, etc. See [10] for instance).

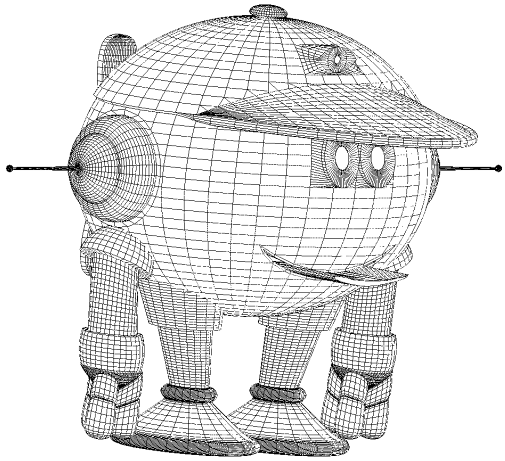

Figure 9. Complex objects can be modeled with G-maps

The last part is devoted to embedding. First linear embedding is described. After, parametric trimmed shapes are introduced. They generalize the notion of trimmed patches [5]. Hence, the link between the topological structure and embedding becomes more sophisticated: the space of parameters must be topologically structured.

Again, several open research problems are pointed out.

3. Remarks and Conclusion

Our experience shows that a practical work is useful : all notions can be perceived as theoretical ones and intrinsically interesting ; a practical work, for instance the conception of the kernel of a geometric modeler, shows the usefulness of these notions for the design of data structures and algorithms.

Note that sections 2.1, 2.2 and/or 2.5 can be extracted and taught in other curricula. Parts 2.1 and 2.2 have been taught to undergraduate Computer Science students : so, they can experiment simplicial structures which are graph generalizations. It is also possible to study only cellular structures, by removing sections 2.2, 2.3, 2.4. It is thus necessary to give an intuitive introduction, e.g. "how to deduce a generalized map from a cellular manifold?" (for instance using barycentric triangulation, splitting cell boundaries and/or cell-tuples [2]). But some notions can not be completely explained (e.g. definition of cartesian product of cellular structures). At last, the course can be improved by studying other important notions, e.g. algorithms for computing homology groups, according to the background of students and the pedagogical context.

To sum up, this course is based upon simplicial notions (topology: simplicial sets; embedding: curved shapes) and two mechanisms:

- cartesian product providing spaces with simploidal cells,
- numbering 0-simplices, providing cellular structures embedded with parametric trimmed shapes.

This approach presents mathematical and computational aspects. Fundamentals are explained. Indications are given about higher-level aspects, some of them still being subjects of active research.

Topology and embedding are simultaneously presented in order to show their complementarity and to highlight their common combinatorial notions, which are important for the conception of data structures and algorithms.

All topological notions are based upon simplicial ones issued from combinatorial topology. This discussion thread is useful for the study of deduced structures, and simplifies the explanations about topological and embedding concepts, operations and algorithms. Moreover, this approach leads coherently to many other works in geometric modeling, and mainly in Boundary Representation.

References

[1] Y. Bertrand, J.-F. Dufourd, J. Françon and P. Lienhardt. Algebraic Specification and Development In Geometric Modeling. *Conf. Tapsoft'93*, LNCS n°668, pages 75-89, Springer-Verlag, April 1993.

[2] E. Brisson. Representing Geometric Structures in d-dimensions : Topology and Order. *Discrete and Computational Geometry*, 9, pages 387-426, 1993.

[3] W. Dahmen and C. A. Micchelli. On the Linear independence of Multi-Variate B-splines I. Triangulation of Simploids. *SIAM J. Numer. Anal.* 19, pages 993-1012, 1982.

[4] H. Elter and P. Lienhardt. Cellular Complexes as Structured Semi-simplicial sets. *International Journal of Shape Modeling*, 1(2) pages 191-217, 1994.

[5] G. Farin. *Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide.* Academic Press, San Diego, 1988.

[6] L. Fuchs. Spécification formelle des modèles de courbes et de surfaces de formes libres. PhD Thesis, Université Louis Pasteur, Strasbourg, France, 1997.

[7] L. Fuchs and P. Lienhardt, Topological Structures and Free-Form Spaces. *Journées de Géométrie Algorithmique*, Barcelone, Spain, September 1997.

[8] H.-B. Griffiths. *Surfaces.* Cambridge University Press, 2nd edition, 1981.

[9] V. Lang and P. Lienhardt. Geometric Modeling with Simplicial Sets. *Proc. of C.G.I.'96*, Pohang, Korea, 1996.

[10] P. Lienhardt. Topological Models for Boundary Representation : a Comparison with N-Dimensional Generalized Maps. *Computer-Aided Design*, 23(1), pages 59-82, 1991.

[11] P. Lienhardt, N-Dimensional Generalized Combinatorial Maps and Cellular Quasi-Manifolds. *International Journal of Computational Geometry and Applications*, 4(3), pages 275-324, 1994.

[12] J. P. May. *Simplicial Objects in Algebraic Topology.* Van Nostrand, Princeton, 1967.

[13] J. P. Serre. Homologie singulière des espaces fibrés. *Ann. of Mathematics*, 54, pages 425-505, 1951.