Chapter 1

Geometric Modeling

This chapter introduces the required background on geometric modeling, focusing on polyhedral representations of solids.

This chapter is organized as follows:

1. *Aims and scope of geometric modeling.* The geometric modeling problem is formalized by distinguishing between three separate levels of modeling: physical objects, mathematical models and representations.

2. *Mathematical models of solids.* Point-set models for representing geometric objects are discussed. Concepts such as two-manifolds and two-manifolds with boundary are introduced.

3. *Representation schemes for solids.* The more popular representation schemes for solids are presented: boundary representations, constructive models and decomposition models.

4. *Photometry representation.* This section discusses the representation of visual information in 3D models due to its impact on appearance-preserving simplification.

5. *Polyhedral representations.* The boundary representation is described in detail, focusing on polyhedral representations. Concepts such as genus and shells are reviewed.

6. *Space decomposition models.* Space decomposition models are introduced, focusing on voxel-based and octree-based models. Surface reconstruction from space decomposition models is also discussed due to its impact on our simplification approach.

1.1 Aims and scope of geometric modeling

Geometric modeling deals with representation and processing of geometric information on 1D, 2D or 3D objects. A rigorous view of modeling is based on distinguishing between three separate levels of modeling [Man88]:

- *physical objects*

  The aim of modeling is to study and argue about some real or imaginary things of our world which are called physical objects.

- *mathematical models*

  A *mathematical model* is a geometric model that has a clear and intuitive mathematical connection with its physical counterpart. Mathematical models are suitable for human reasoning but often inappropriate for computer manipulation.

- *representations*

  A *representation scheme* is a set of rules defining the mapping from a mathematical model to another model suitable to computer manipulation. Such geometric model is called a *representation* and consists of a finite collection of basic elements called *symbols*. The *domain* of a representation scheme is the set of mathematical models that can be modeled with such representation scheme.

Geometric modeling deals with several kinds of objects:

- *curves and polygonal lines*

  Curves and polygonal lines are one-dimensional objects embedded in $\mathbb{R}^2$ or $\mathbb{R}^3$. For example, the idealization of a road in a Geographic Information System (GIS) is a sequence of curved and straight-line segments.

- *surfaces*

  Surfaces are two-dimensional subsets of $\mathbb{R}^3$. For example, the hull of a ship in hydrodynamics applications could be represented with surface models. Surface models give detailed information on the geometry of a curved surface but do not always give sufficient information for determining all the geometric properties of the object potentially bounded by the surface.

- *solids*

  A solid is a three-dimensional object whose interior is considered homogeneous and isotropic. For example, the model of an engine in a CAD system is described as an assembly of solids.

- *volume data*

  Volume data represents spatial properties of heterogeneous, anisotropic 3D objects. For example, density values of a human tissue in medicine applications are described through volume models. Volume models are broadly used in medicine, earth sciences, biochemistry, biology and fluid dynamics.

- *3D models*

  A *3D model* is a model including both geometric information (the geometric model) and non-geometric information such as visual information. For example, a piece of furniture in an interior design application is represented as a 3D model including its geometry and visual properties such as color and texture.

1.2 Mathematical models of solids

As stated in the previous section, mathematical models are suitable for human reasoning but often inappropriate for computer representation. The most common mathematical models are *point-set* models. Point-set models describe geometric objects as a subset of points of $\mathbb{R}^2$ or $\mathbb{R}^3$. For example, a spherical object with radius R centered at the origin can be described as the set of points $\{(x, y, z) \mid x^2 + y^2 + z^2 \leq R^2\}$.

From the point of view of differential geometry, a single parameterized curve embedded in 3D space is a differentiable map $\alpha : I \rightarrow \mathbb{R}^3$ of an open interval $I = (a, b)$ of the real line $\mathbb{R}$ into $\mathbb{R}^3$. Polygonal lines are defined analogously through piece-wise linear functions. The point-set models of these curves are the ranges of the map function.

A *two-manifold with boundary* is a topological space where every point has a neighborhood topologically equivalent to an open disk of the two-dimensional Euclidean space E^2, except points on the edge of an open surface patch. Intuitively speaking, two-manifolds are non-selfintersecting, open surfaces. Two-manifolds with boundary are the most common mathematical models for surfaces not enclosing a volume.

A *two-manifold* is a topological space where every point has a neighborhood topologically equivalent to an open disk of E^2. Intuitively, two-manifolds are non-selfintersecting, closed surfaces. A two-manifold is said to be *realizable* if it encloses a 3D volume [Man88]. Orientable two-manifolds are the most common mathematical models for surfaces enclosing a volume.

From the point of view of point-set models, a *solid* is a bounded, closed subset of E^3.

The *regularization* of a point-set S, $r(S)$, is defined by

$$r(S) = c(i(S)), \tag{1.1}$$

where $c(S)$ and $i(S)$ denote the closure and interior of S, respectively. A set S is said to be *regular* if $r(S) = S$.

A solid is said to be *valid* if it is a bounded, regular set with a two-manifold, orientable boundary [Man88]. Solid modeling deals with *valid* and *complete* geometric representations of solids. By complete we mean that representations must be adequate for answering arbitrary geometric questions about the object [Man88].

1.3 Representation schemes for solids

Several representation schemes have been proposed in the literature for representing solids and surfaces [Man88].

Constructive models represent a point-set as a combination of primitive point-sets. Each primitive is represented as an instance of a primitive type and combination operations are set boolean operations (union, intersection, difference, complement).

Boundary models represent point-sets in terms of their boundary. Objects are represented by dividing their surface into a collection of connected components called *faces*. The division is performed so that each face has a compact mathematical representation, e.g. the face lies on a planar, quadratic or parametric surface. The portion of the underlying surface that forms the face is trimmed out in terms of closed curves lying on such surface. This kind of representation is called *boundary representation* (BRep for short).

Space decomposition models (SDM for short) represent a point-set as the union of disjoint regions of the space called *cells*. Cells that contain a part of the object are usually labeled as *black* and the rest are labeled as *white*. SDM are approximate models, as they cannot represent exactly most solids and surfaces (see [Req80], [Sam90b], [Sam90c]); the accuracy of the SDM depends on the size of the cells.

1.4 Polyhedral models

A *polyhedral model* is a boundary model whose faces are connected subsets of the plane. Each planar face is represented by one or more planar polygons. One polygon defines the outer boundary of the face while the others (called *rings* or *interior loops*) represent interior holes of the faces. Polygons are described by an ordered sequence of straight-line segments called *edges*. Each edge is defined by its two endpoints, called *vertices*.

The number of faces, edges and vertices of a general polyhedral model are related by the Euler-Poincaré formula:

$$v - e + f = 2(s - g) + h, \tag{1.2}$$

where v, e, f, s, g and h are resp. the number of vertices, edges, faces, shells, genus and rings (interior loops in faces). A *shell* is a maximally connected set of faces. The *genus* of the surface, which denotes the number of crossing holes, is defined as half the connectivity number of the surface, i.e., $\frac{h_1}{2}$, where h_1 is the largest number of closed curves that can be drawn on the surface without dividing it into two or more separate components. For instance, cubes and spheres have genus 0 and a torus has genus 1.

A *polygonal mesh* is a polyhedral model whose faces are simply-connected, i.e. they can be represented by a single polygon (see Figure 1.1).

A *triangle mesh* is a polygonal mesh whose faces are triangles (see Figure 1.1). In a two-manifold triangle mesh $2e = 3f$, so the Euler formula can be rewritten:

$$2v - f = 4(s - g). \tag{1.3}$$

1.5 Photometry representation

Realistic visualization of geometric models often requires the representation of *photometry* information such as normal vectors, colors and textures. Photometry study is relevant for geometry simplification since photometry attributes are attached to the surface and therefore simplified 3D models should retain their important photometry features.

Besides the sophisticated photometry of photo-realistic rendering, the most common photometry attributes of 3D models intended for real-time visualization are:

- *color*

  Color is often defined as ambient, diffuse, emission and specular reflection coefficients, whether in a per-object, per-face or per-vertex basis. In the first two cases, the final color is affected by real-time lighting calculations, whereas in per-vertex color models (e.g. on radiosity-lighted models) no lighting computations are required.

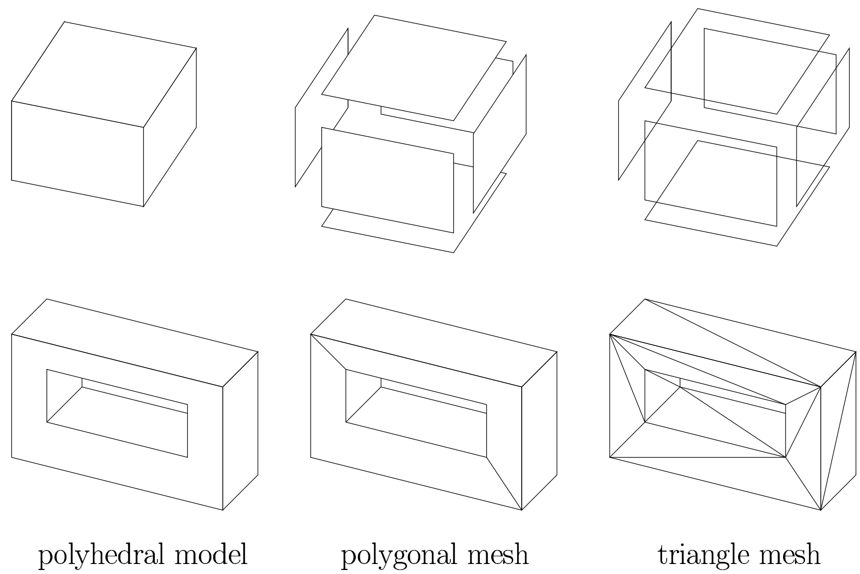

Figure 1.1: *Top*: Geometric entities of a BRep; *Bottom*: polyhedral (*left*), polygonal (*middle*) and triangle mesh (*right*) representations of a simple object.

- *translucency*

  Translucency is usually represented as an extra color component (called *alpha* value) and it comes on a per-object, per-face or per-vertex basis.

- *texture maps*

  Textures maps and some other texture mapping attributes are often defined on a per-object or per-face basis.

- *texture coordinates*

  Explicit texture coordinates are defined at corners or vertices and must be stored along with the geometric model. High-end API's such as OpenGL provide several texture mapping functions for automatic generation of texture coordinates. In this case, the mapping parameters are usually defined at faces.

- *surface normals*

  Real-time visualization systems use per-vertex normals for lighting computations. Unlike per-face normals, which can be easily computed from the geometry, accurate per-vertex normals usually require a little knowledge of the object's shape. For instance, the polyhedral representations of a cylinder and a cylindrical prism can be identical regarding to their geometry but different according to their per-vertex normals.

1.6 Space decomposition models

Space decomposition models (SDM for short) represent a point-set as the union of disjoint regions of the space called *cells* (see [Req80], [Sam90b], [Sam90c]). Cells containing part of the object are labeled as *black* and the rest are labeled as *white*. Unlike BRep models, SDM are suitable for representing volume data. In this case, cells represent regions with homogeneous interior with respect to the studied property.

SDM are approximate models as they cannot represent exactly most solids and surfaces (see [Nav86] for an extended octree representation that can exactly represent polyhedral solids). The accuracy of the SDM representation depends on the size of the cells. Tetrahedra, cubes and boxes are the more relevant cells. SDM play a special role in geometry simplification because solids and surfaces represented by these models can be trivially simplified, by just gluing adjacent cells. Furthermore, SDM provide a simple and stable way of changing topology.

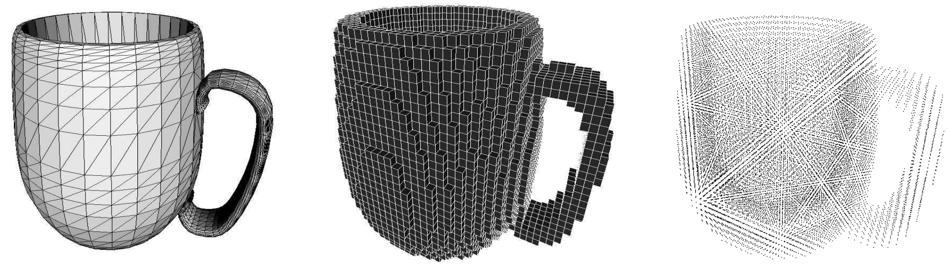

Figure 1.2: Rendering of a mug represented with a triangle mesh (*left*), a voxel decomposition (*middle*) and a digital picture (*right*)

1.6.1 Voxel-based representations

A *voxel decomposition* is a SDM whose cells are equal-sized cubes, called *voxels*, arranged in a regular array. Voxel decompositions are suitable for representing solid objects and volume data. Solid objects are represented by labeling interior voxels as *black* (or '1') and voxels outside the solid as *white* (or '0') (see Figure 1.2). For volume data representation, voxels are labeled according to the value of the property being modeled inside the voxel. In voxel decompositions, the interior of each voxel is considered homogeneous.

A *3D picture* is a set of points arranged in a regular grid defining equal-sized cubic cells. Points of a 3D picture (called *lattice points*) are labeled according to the property being studied. Eight neighbor lattice points define a voxel (see Figure 1.4). Unlike voxel decompositions, 3D pictures deal with voxels with non-homogeneous interior, since the property is only known at the lattice points, which coincide with voxel's vertices (see Figure 1.2). The space of interest can be conveniently scaled so that lattice points have integer coordinates. A *3D digital picture* is a set $B \subset Z^3$. The elements of Z^3 are called points of the picture. The points in B are called the *black points* of the picture; the points in $Z^3 - B$ are called the *white points* of the picture.

Two points in 3D-space are said to be *26-adjacent* if they are distinct and each coordinate of one differs from the corresponding coordinate of the other by at most 1 (Figure 1.3); two points are *18-adjacent* if they are 26-adjacent and differ in at most two of their coordinates; two points are *6-adjacent* if they are 26-adjacent and differ in at most one coordinate. In terms of voxels, 26-adjacent voxels share a face, edge or vertex; 18-adjacent voxels share a face or edge, and 6-adjacent voxels share a face. An *n-neighbor* of a point (resp. voxel) p is a point (resp. voxel) that is n-adjacent to p. A set S of points is *n-connected* if S cannot be partitioned into two sets that are not n-adjacent to each other.

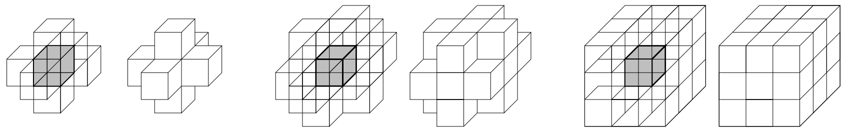

Figure 1.3: A voxel (*shaded*) and its 6,18,26-neighbors

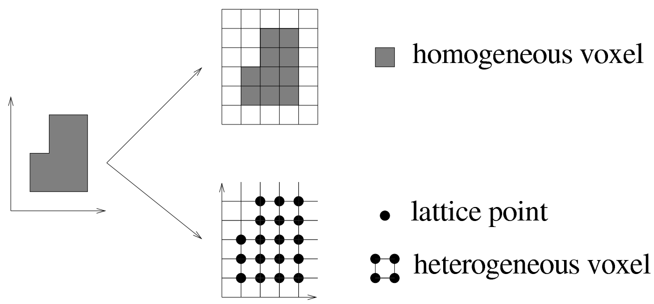

Figure 1.4: Voxel decompositions and 3D pictures

1.6.2 Octree-based representations

The *octree* representation uses a recursive subdivision of a cubic universe into eight octants that are arranged into an 8-ary tree. In the classical octree representation [Sam90a] (CO for short), each node consists of a code (often called *color*) and eight pointers towards eight sons (Figure 1.5). Nodes corresponding to cubic regions completely inside the object are labeled as *black* (B). Nodes corresponding to cubic regions completely outside the object are labeled as *white* (W). White and black nodes are leaves, i.e. they are no further subdivided. Nodes containing a part of the boundary are labeled as *grey* (G) and are recursively subdivided. Leaf grey nodes are called *terminal grey* (TG) nodes.

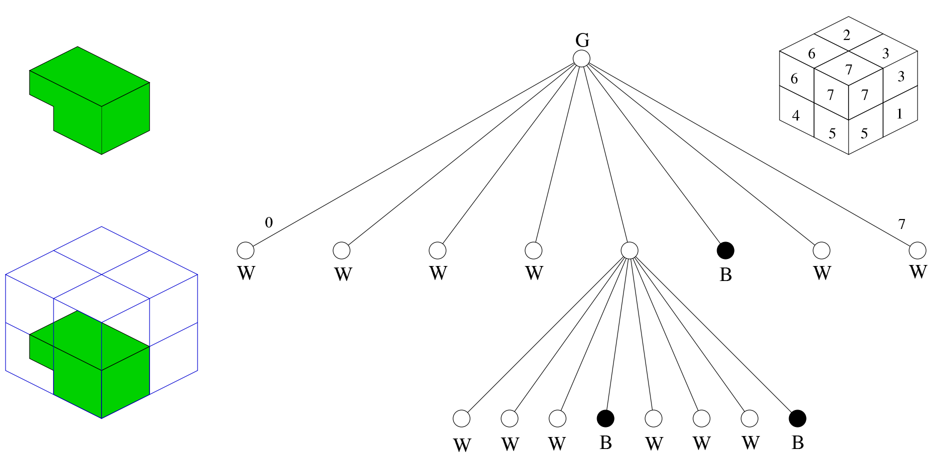

Figure 1.5: Octree representation of a simple object. The cube on the top right corner shows the octant numbering.

The *maximal division classical octree* [BJN+94] (MDCO for short) is an extension of the classical octree scheme where all terminal grey nodes belong to the same level of the octree and hence have the same size (Figure 1.6). Given a solid P and a non-negative integer l, $MDCO(P, l)$ is an octree representation of P, containing W, B, G and TG nodes, where all TG nodes belong to the deepest level l.

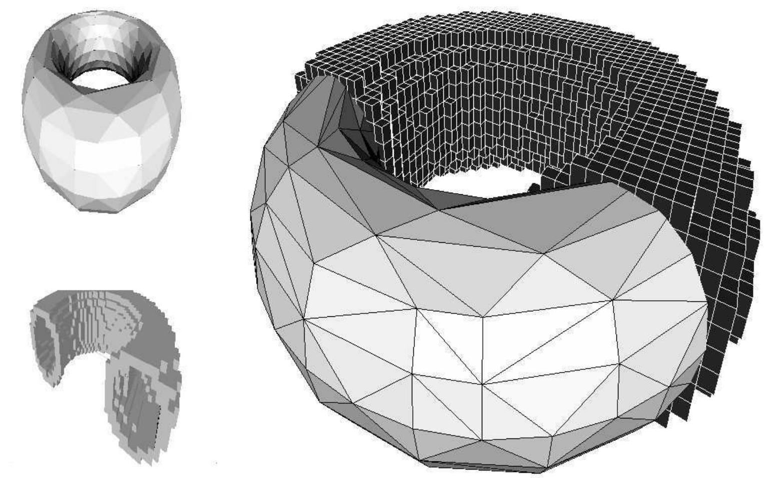

Figure 1.6: MDCO representation of a torus: BRep of a torus (*upper left*), cubic regions corresponding to terminal grey nodes (*lower left*) and merged with the BRep model (*right*).

The boundary of the solid is completely contained in the set of TG nodes. The set of TG nodes can be viewed as a voxelization of the object's boundary with the hierarchical structure of the MDCO on top of it. Unlike voxel-based representations, the adaptive decomposition provided by octree cells allows data compression in homogeneous regions.

1.6.3 Surface reconstruction from space decomposition models

Isosurface extraction deals with generation of isosurfaces from volume data. Isosurfaces approximate the points with a given property value, called *isodensity value*. Isosurface extraction is a powerful tool for analysis and visualization of volume data. *Surface fitting* deals with generation of surfaces approximating a set of points which are known to be on the surface. These points are usually acquired from 3D digitizing techniques. Isosurface extraction, surface fitting and SDM to BRep conversion are globally known as *surface reconstruction*. Since surface reconstruction plays a special role in our simplification strategy, a brief survey on this topic is presented in Section 4.5.1.

1.7 Conclusions

Solid modeling deals with valid and complete representations of solids. From a mathematical point of view, a valid solid is a regular set with a two-manifold boundary. Several representation schemes have been proposed for generating representations of solids suitable for computer manipulation. Boundary models represent point-sets in terms of their boundary. Polyhedral models occupy a prevalent position in interactive computer graphics. Space decomposition models, which provide approximate representations of solids, explicitly represent the interior points through enumeration of occupied cells.