Geometric Modeling with a Multiresolution Representation

Richard Hammersley
Schlumberger
8311 North RR 620
Austin TX, 78726
+1 512 331 3731
hammersl@slb.com

Hong-Qian (Karen) Lu
Schlumberger
8311 North RR 620
Austin TX, 78726
+1 512 331 3350
karenlu@slb.com

Steven Assa
Schlumberger
Maddingley Road
Cambridge
+44 1223 325440
sassa@slb.com

1. ABSTRACT

Many disciplines can profitably use large high-resolution geometric models whose computational requirements exceed current computer hardware capabilities. This paper presents a new multiresolution surface representation and its application for building scalable geometric models in geoscience. We build models from surfaces at full resolution and construct topologically correct decimated models. The technology is embedded into an application-neutral 3D geoscience geometric modeling framework.

1.1 Keywords

Multiresolution, surface representation, geometric modeling, decimation.

2. INTRODUCTION

Geoscience applications need to build and share large high resolution geometric models across widely varying computer hardware. This paper presents a new approach, the Scalable Interactive geometric Modeling Architecture (SIGMA) which extends current geometric modeling technology to solve many of the problems in building and sharing large geometric models for geoscience.

Two major issues for interactively building large geometric models are memory usage and rendering performance. There is a rich literature on surface decimation [4,5] and one solution is to construct the geometric model from decimated surfaces. However, the information lost in decimation cannot be recovered from the model.

SIGMA is a scalable solution that uses a new multiresolution surface representation to build geometric models. The model is built at full resolution to ensure no loss of information. Furthermore, SIGMA can generate many topologically consistent decimated models. Multiresolution surface representations have been a hot research topic in recent years [7], but the research has been confined to visualization.

The SIGMA surface representation is a multiresolution hierarchy based on a regular subdivision. The hierarchy is implemented as a quadtree, and is used for intersection computations and multi-resolution decimation. This paper focuses on the surface representation, geometric modeling, and model decimation. SIGMA visualization is presented in another paper [8].

2.1 Background

A geoscience geometry model is built from surfaces that represent the discontinuities of material properties in the subsurface. Figure 1 illustrates some subsurface structures.

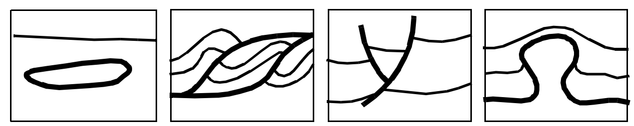


Figure 1. Cross-sections of 3D subsurface structures

We construct subsurface geometry models using Irregular Space Partitioning (ISP) [2]. ISP is a sequence of subdivides by surfaces and volumes on a volume of interest which is similar to Constructive Solid Geometry [10]. See Figure 2

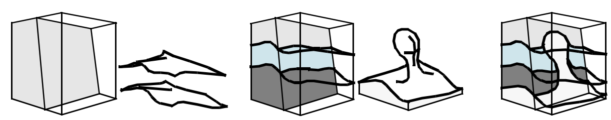


Figure 2. Irregular Space Partitioning.

This paper assumes a piece-wise linear geometric modeling kernel. We use SHAPES from XOX [15].

3. PSEUDO-MANIFOLDS

A *pseudomanifold* [1], *M*, is a simplicial complex such that

1. *M* is homogeneously *n*-dimensional. That is, every simplex of *M* is a face of a *n*-simplex of *M*.
2. Every *(n*-1*)*-simplex is a face of at most two *n*-simplices.
3. If σ and σ' are two distinct *n*-simplices of *M*, then there exists a sequence $\sigma_1,...,\sigma_k$ of *n*-simplices in M, such that $\sigma_1 = \sigma$, $\sigma_k = \sigma'$ and σ_i meets σ_{i+1} in a *(n-1)*-face for $1 \le i < k$.

A *triangulated surface* or *triangle mesh* is a two-dimensional pseudo-manifold. The boundary of a simplicial complex *K* is denoted by ∂K.

4. CELL COMPLEXES AND CLASSIFICATION

Our boundary representation is based on the notions of CW-complexes [9], except the cells are relaxed to be pseudo-manifolds as opposed to disks [3, 6]. We call these boundary representations *subdivided complexes* denoted by *(X,C)* where *X* is a topological space and *C* is the collection of cells. The boundary representation is similar to Selective Geometric Complexes [11]. We define the operation of classification for ISP. The previous works [3, 6, 14] also define classification, but their interest is with the representation of cellular topology.

Let *(X,C)* and *(Y,D)* be two subdivided complexes. A *boolean classification* of *(X,C)* and *(Y,D)* is a third subdivided space *(Z,E)*, such that,

1. $Z = X \cup Y$
2. For each pair of cells $c_\alpha \in C$ and $d_\beta \in D$ there are

disjoint collections of cells in $E = \{e_\zeta\}$, such that,

$c_\alpha \backslash d_\beta = \bigcup e_\gamma$, $d_\beta \backslash c_\alpha = \bigcup e_\delta$ and $c_\alpha \cap d_\beta = \bigcup e_\varepsilon$. Classification must commute with the boundary operator, that is $\partial_Z(c_\alpha \cup d_\beta) = \partial_X c_\alpha \cup \partial_Y d_\beta$ as cells in *E*.

5. ISOMORPHISM OF MODELS

Let *(X,C)* and *(Y,D)* be two subdivided complexes. A continuous map $f{:}X \to Y$ is a *cellular* map if *f* induces a map from C to *D*. If the cellular map is a homeomorphism which induces an isomorphism on the cells *C* and *D*, then *(X,C)* is *isomorphic* to *(Y,D).*

5.1 Cracking

Cracking is often seen as a new hole where there was no hole before. Figure 3 shows an example of cracking.

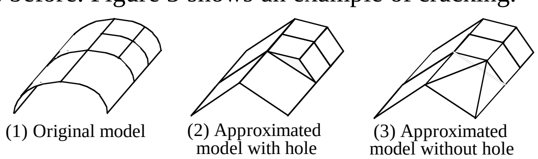


Figure 3. Cracking in a model.

Let *(X,C)* and *(Y,D)* be two subdivided complexes. Then *(Y,D)* is a *crack-free* representation of *(X,C)* if there is a cellular map $f{:}X \to Y$ which has a left inverse $g{:}Y \to X$, that is $gf = Id$.

5.2 Bubbling

Bubbling is new intersections where there were none previously. In Figure 4 shows an example of bubbling.

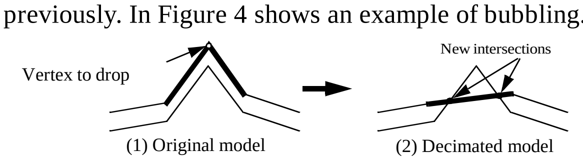


Figure 4. Bubbling in a model.

Let *(X,C)* and *(Y,D)* be two subdivided complexes. Then *(Y,D)* is a *bubble-free* representation of *(X,C)* if there is a cellular map $g{:}Y \to X$ which has a left inverse $f{:}X \to Y$ that is $fg = Id$.

6. AN HIERARCHICAL SURFACE REPRESENTATION

SIGMA has a new hierarchical surface representation based on a quadtree. A quadtree [12] was chosen because of its geometrical relationship to sub-sampling in grids. SIGMA supports ISP through the boolean classification operator. SIGMA also supports the construction of crack-free and bubble-free decimated models.

6.1 Definitions

The number of elements (cardinality) in a collection *C* will be denoted by Card(*C*). Every node of a quadtree can be assigned a unique key, Key(*N*). The quadtree node of a key *K* is Node(*K*). In a fixed quadtree $\mathrm{Node}(\mathrm{Key}(N)) = N$ and $\mathrm{Key}(\mathrm{Node}(K)) = K$. The depth of the key *K* is Depth(*K*). and the root key has depth 0. The ancestor key at depth *i* of the key *K* is $\mathrm{Ancestor}_i(K)$. $\mathrm{Ancestor}_i(K)$ is defined for $i \le \mathrm{Depth}(K)$ with $\mathrm{Ancestor}_{\mathrm{Depth}(K)}(K) = K$. $\mathrm{Ancestors}_i(C)$ are the ancestor keys of the keys *C*.

A collection of nodes, *C*, of the tree *T* is a *node front* if every leaf node of T has at most one ancestor in *C*. A node front is a *complete node front* if every leaf node of *T* has exactly one ancestor in *C.* A node front *D* is *finer* than the node front *C* of the tree *T*, (*C* is *coarser* than *D*) if every node in a node in *C* is an ancestor of a node in *D.*

6.2 Subdivision

For each surface in the model a quadtree is built. Triangles are assigned to unique tree leaf nodes. For a structured grid this is done by tiling the parameter space of the grid with quadtree leaf nodes and assigning a grid cell's triangles to its nearest quadtree leaf node. Each tree node is assigned the triangles of its descendants. Hence, a complete node front defines a partitioning of the triangles. The tree leaf node which contains the triangle *S* is Leaf(*S*).

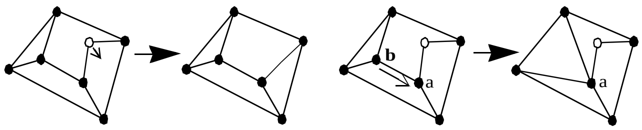
6.3 Vertex Descriptor

Each triangle has been assigned to a leaf node in the tree and the vertices of the triangle are also assigned to the same leaf node. The *vertex descriptor* for the vertex, *v*, is the list of leaf keys of the triangles connected to the vertex, *v*, $\mathrm{Keys}(v) = \bigcup_{S \in \mathrm{Simps}(v)} \mathrm{Key}(\mathrm{Leaf}(S))$ where Simps(*v*) are the triangles having *v* as a vertex.

6.4 Boundaries of Tree Nodes

A tree node comprises a collection of triangles originating from a pseudo-manifold and has a well-defined boundary. Let *C* be a complete node front of the tree. The *boundaries of C* are the boundaries of all the nodes of *C.* The boundaries of the complete node front form a graph.

In Figure 5a the unshaded vertex can be dropped without changing the topology of the graph. In Figure 5b when the vertex b is collapsed to the vertex a, the valence of vertex a changes from three to four and the topology is modified.



(a) Dropping a vertex does not change the graph topology

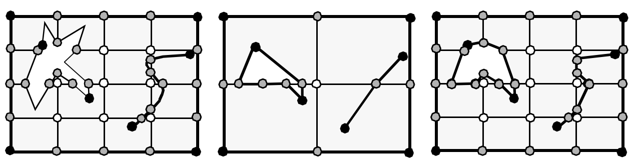
(b) Dropping a vertex changes the graph topology

Figure 5. Dropping a vertex may change graph topology.

It can be seen the only vertices which can be dropped and still preserve the topology of the graph are the vertices of valence two. This can be made precise by introducing the notion of homeomorphic graphs [1].

6.5 Critical Vertices

Let $\dim(v)$ be the lowest dimension of the cells that a vertex v lies in. Let $k_i(v) = \mathrm{Card}(\mathrm{Ancestors}_i(\mathrm{Keys}(v)))$ be the number of ancestor keys at depth i of the vertex v. Then the vertex v is *critical* at level i, if $k_i(v) > \dim(v)$. The *depth* of the vertex v is the smallest i for which the vertex is critical. A vertex is assigned to all the nodes for which it is critical. The critical vertices for a complete node front, C, of the tree T are all those vertices which have been assigned to the nodes of C. Figure 6 shows how critical vertices approximate a surface.



Original surface

Decimated surface with level 1 nodes

Decimated surface with level 2 nodes

● 0-cell vertex ◉ 1-cell vertex ○ Internal vertex

Figure 6. Critical vertices at depths 1 and 2

Lemma: The critical vertices of a complete node front, C, contain the vertices which cannot be removed without changing the topology of the graph of boundaries of C.

Lemma: Suppose a complete node front D is finer than the complete node front C. Then the critical vertices of C are a subset of the critical vertices of D.

Lemma: Let nodes N and M share a critical manifold vertex, v. Let P and Q be the pseudomanifold subsets of N and M respectively which contain v. Then P and Q are connected by the pseudomanifold relationship.

6.6 The Topology of Tree Nodes

If a tree node includes any triangles that are on the boundary of the underlying mesh, then this node is called a *macro-node*. Any tree node which is not a macro-node is required to be homeomorphic to a 2-disk, in particular the node is connected and simply-connected (does not have any holes).

Lemma: Let C be a complete node front. Then every node in C has at least one critical vertex.

7. MODEL CONSTRUCTION

As described in Section 2 an ISP model is constructed incrementally by inserting new surfaces or new volumes into an existing model. When a new surface is introduced into the model it is necessary to recompute the topology of the model, which is called classification. Classification requires identifying a new decomposition of the model as pseudo-manifolds. This section describes the SIGMA part of the classification algorithm and the decimation.

Geometric modeling algorithms break into two fundamental parts, geometry calculations and topological calculations. The geometry calculations are performed by SHAPES [15]. This paper addresses some of the issues of the topological calculations. We assume the geometric modeling kernel can compute the topology of a collection of simplices.

7.1 Pairwise Intersection of Surfaces

The classification algorithm first identifies all intersecting simplices. The SIGMA tree structure aids in this calculation by storing a min-max box (or similar data-structure, such as, a convex hull), at each node of the tree large enough to contain all the triangles assigned to this node.

Given two SIGMA trees pairwise intersection of min-max boxes using hierarchical traversal will identify all intersecting leaf nodes. This process identifies a complete node front from each tree.

The collection of triangles from the intersecting leaf nodes of a surface forms a *swath*. The swaths are passed to the geometric modeling kernel which intersects all the triangles passed to it. This is sufficient to compute the topology of the swath.

7.2 Migration

Migration is the process by which the new pseudomanifold regions are computed during classification. Migration splits the original tree into several trees, one tree for each component. Migration is implemented as a flood fill operation in a complete node front seeded from the intersecting triangles.

Let C be a complete node front which contains all the leaf nodes involved in the intersection. Suppose, furthermore every leaf node containing a non-manifold vertex is in C and all other nodes in C are connected.

Migration proceeds as follows. Triangles which are involved in the intersection are retriangulated to respect the new intersection curve; each new triangle is assigned to the leaf node of the original triangle. Using the pseudo-manifold structure of a leaf node, we migrate from the interior edges of the split triangles to the remaining triangles that were not split in a split leaf node. This results in a new collection of interior edges which lie on the boundary of nodes in the complete node front which were not split.

Collect all the vertices from a collection of interior edges from one of the swath components. Next collect the vertex descriptors from these vertices. From these vertex descriptors identify the coarsest non-migrated node in the complete node front *C* which has this vertex as a critical vertex. If there are no nodes left in the complete node front then the migration has been completed. If any split nodes have any of the migrated vertices as critical vertices which were not in the original collection of split nodes for this collection of interior edges, connect the two swaths of triangles.

After the migration has been completed the critical vertices of the tree nodes are recomputed. The min-max boxes of the quadtree nodes which were split can also be recomputed.

7.3 Decimation

Constructing a decimated model which is isomorphic to the original model is of interest to both visualization and simulation. This section describes an algorithm to construct a crack-free and bubble-free decimated model for SIGMA.

By analogy with the quadtree a binary tree is constructed for each 1-cell. A complete node front of the binary tree defines a crack-free decimation of the 1-cell by taking line segments joining the critical vertices. Many 2-cells may share the same 1-cell as a boundary and must use the same decimated version of this 1-cell to prevent cracking. In the parameter space of each 2-cell the pre-image of the decimated 1-cell may have bubbles, which must be removed.

For each 2-cell, choose a complete node front from the quadtree. Collect the critical vertices from the node front.

In the parameter space of a 2-cell, build a list of parameter values and edges from the critical vertices and the edges from the decimated bounding 1-cells. Run a tessellator (for example, constrained Delaunay [13]) which respects the imposed edges from the 1-cells in the surface and the critical vertices from the chosen quadtree nodes. Build up a tessellation of the surface in three-dimensional space by evaluating the parameter values for the surface and using these as corner points for the triangles computed by the tessellator in parameter space.

This algorithm prevents cracking in the model. To prevent bubbling rerun the classification algorithm and ensure no additional topology has been introduced. If new topology has been introduced then refine the model in these areas. Since the full-resolution model has no bubbling this process must terminate.

8. CONCLUSIONS

This paper presents a new multiresolution representation which can be used for scalable geometric modeling in geoscience. Geometric models are built from surfaces at full resolution and we construct topologically correct decimated models from a full resolution model. The new representation has been integrated with a commercial geometric modeling kernel. The integration defines a clear separation of topology computation from geometric computation. We have embedded the technology presented in this paper into a 3D geoscience geometric modeling application framework that supports many applications.

9. Reference

[1] M.K. Agoston, *Algebraic Topology, A First Course*, Pure and Applied Mathematics, Marcel Dekker 1976.

[2] S. Assa, G. Celniker, and C. Ramshorn, “Feature-Based Geometric Modeling for Geoscience”, *GOCAD ENSG Conference, 3D Modeling of Natural Objects: A Challenge for the 2000’s*, June 1998.

[3] E. Brisson, “Representing Geometric Structures in d Dimensions; Topology and Order,” *Proc. 5th ACM Symposium on Computational Geometry*, June, 1989, pages 187-196.

[4] J. Cohen, A. Varshney, D. Manocha, G. Turk, H. Weber, P. Agarwal, F. Brooks, and W. Wright, “Simplification Envelopes,” *SIGGRAPH 96 Proceedings*, pages 119-128.

[5] P. Heckbert and M. Garland, “Survey of Polygonal Simplification Algorithms,” *CMU Technical Report* CMU-CS-95-194.

[6] P. Lienhardt, “N-Dimensional Generalized Combinatorial Maps and Cellular Quasi-Manifolds,” *International Journal of Computational Geometry and Applications*, Vol. 4, no 3, 1994, pages 275-324.

[7] M. Lounsberry, T. DeRose, J. Warren, “Multiresolution Analysis for Surfaces of Arbitrary Topological Type,” *ACM Transactions on Graphics*, Vol. 16, No. 1, Jan 1997, pp. 34-73.

[8] H.Q. Lu and R. Hammersley, “Scalable Visualization for Interactive Geometric Modeling in Geoscience,” Submitted to VIS ‘99.

[9] J.W. Milnor and J.D. Stasheff, “Characteristic Classes,” *Annals of Mathematics Studies 76*, Princeton University Press, 1974.

[10] M.E. Mortenson, *Geometric Modeling*, John Wiley & Sons, Inc. 1985.

[11] J.R. Rossignac, M.A. O’Connor, SGC: A dimension-independent model for pointsets with internal structures and incomplete boundaries. In *Geometric Modeling for Product Engineering*, Eds. M. Wosny, J. Turner, K. Preiss, North-Holland, pp. 145-180, 1989.

[12] H. Samet, *The Design and Analysis of Spatial Data Structure,*. Addison-Wesley Series in Computer Science 1990. ISBN 0-201-50255-0.

[13] J. Shewchuk, “Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator,” *1st ACM Workshop on Applied Computational Geometry*, pages. 124-133.

[14] Tilove, R. B. and A. A. Requicha. Closure of Boolean operations on geometric entities. *Computer Aided Design*, 12(5):219-20, Sept. 1980.

[15] *XOX SHAPES Programmers Guide*. XOX Corporation, 1994, URL=http://www.xox.com.