Automatic Visualization of 3D Complexes

Luis A. P. Lozada
Candido F. X. de Mendonça *Jorge Stolfi*

Relatório Técnico IC–99-28

Dezembro de 1999

Automatic Visualization of 3D Complexes

Luis A. P. Lozada* Candido F. X. de Mendonça† Jorge Stolfi‡

Abstract

A three-dimensional complex is a partition of a three-dimensional manifold into simple cells, faces, edges and vertices. We consider here the problem of automatically producing a "nice" geometric representation (in $\mathbb{R}^m$, for $m \geq 3$) of an arbitrary 3D complex, given only its combinatorial description. The geometric realization is chosen by optimizing certain aesthetic criteria, measured by certain "energy functions."

1 Introduction

The visualization of the geometry and topology of objects contained in spaces of dimension greater than 3 always been a challenging problem, since our visual experience is confined to an Euclidean three-dimensional world.

Among these higher-dimensional objects, the 3-dimensional *complexes*, which include the shells of *4D polytopes*, are particularly important, given their many applications – such as mechanics [15] and robotics [10], and their theoretical interest [13]. In those applications, there is often the need to visualize the structure of a 3D complex for which one has only a combinatorial description. To solve that problem, we need a method to automatically produce a geometric representation of the complex that can be easily grasped by the user. The promising results achieved by Rosi et al. [12] for the analogous 2D problem have led us to apply similar techniques to this task.

The structure of this report is as follows. In section 2 we define 3D topological manifolds and complexes. In order to construct the geometric representation, the elements of the given complex must divided into topological tetrahedra, as described in section 3. Section 4 explains how a geometric representation of the manifold, is specified by a mapping of each vertex to a point of $\mathbb{R}^m$, $m \geq 3$.

We choose the position of the vertices (and hence the geometric representation) by optimizing an *energy function* which measures how badly a solution fails to meet certain visual effectiveness criteria. Some energy functions that we have found especially effective are described in section 5; and the minimization methods we used are discussed in section 6. Section 7 discusses the graphics techniques that can be used to visualize the representation (an object in $\mathbb{R}^m$, $m \geq 3$); some examples are shown in section 8. Finally, in section 9 we present our conclusions and suggestions for future work.

*Institute of Computing, University of Campinas, 13081-970 Campinas, SP. Research supported by grant 96-09873-0 from the Foundation for Research Support of the State of São Paulo (FAPESP)

†Department of Informatica, UEM, Paraná

‡grant 301016/92-5(NV) from the Brazilian Council for Scientific and Technological Development (CNPq)

2 Definitions

2.1 Topological Manifold

We assume known the basic concepts of set topology, such as topological space, homeomorphism, closure, interior, etc, as defined in Kosniowski [8]. If X is a subspace of a topological space Y, we denote by $\partial_Y X$ the boundary of X in Y, by $\kappa_Y X$ the closure of X, and by $\iota_Y X$ its interior (we will omit the subscript Y when it is obvious from the context). We also denote B_m the open unit ball of $\mathbb{R}^m$, by κB_m its closure, and by H_m the half-space $\{(x_1, \ldots, x_m) \in \mathbb{R}^m : x_1 \geq 0\}$.

A topological space X is said to be a *d-dimensional manifold* if every point of X has a neighborhood that is homeomorphic to H_d or B_d. The points of X that have B_d neighborhoods constitute the *absolute interior* αX of the manifold; the remainder constitute its *border* denoted by βX. More generally, we call any space homeomorphic to B_d an *open d-ball*.

2.2 Complex

A *d-dimensional complex* is a finite partition $\mathcal{C}$ of a d-dimensional manifold X into open k-balls, for various $k \leq d$, such that: (1) the boundary $\partial_X c$ of every element $c \in \mathcal{C}$ is the union of elements of $\mathcal{C}$; and (2) for each element c of $\mathcal{C}$, there exist a continuous map ψ_c from some closed ball κB_k onto $\kappa_X c$ such that the inverse image by ψ_c^{-1} of any element e of $\mathcal{C}$ contained in the boundary of c is a finite number of disjoint subsets of $\partial_X B_k$, each of them homeomorphic to e. A *k-element* of a complex $\mathcal{C}$ is an element of $\mathcal{C}$ homemorphic to an open k-ball. We call a 0-element a *vertex*, a 1-element an *edge*, a 2-element a *face*, and a 3-element a *cell*. We denote by $\mathbf{V}\mathcal{C}$, $\mathbf{E}\mathcal{C}$, $\mathbf{F}\mathcal{C}$ and $\mathbf{C}\mathcal{C}$ the vertices, edges, faces and cells of a complex $\mathcal{C}$.

Two complexes $\mathcal{C}'$, $\mathcal{C}''$ defined on manifolds X', X'' are said to be *homeomorphic*, or *topologically equivalent*, if there is a homeomorphism ϕ of X' to X'' such that the image of each element of $\mathcal{C}'$ by ϕ is an element of $\mathcal{C}''$. The *topology* of a complex $\mathcal{C}$ is the class of all complexes that are homeomorphic to it.

3 Triangulation of the Complex

Each cell or face of a 3-dimensional complex may have an unbounded number of vertices, and its boundary may have arbitrarily complex topology and connections. Therefore, in order to construct geometric models for such elements, it is almost mandatory to subdivide them into smaller parts of bounded topological complexity. In other words, instead of modeling $\mathcal{C}$, we model some *refinement* (in the set-partition sense) $\mathcal{C}'$ of $\mathcal{C}$. Of course any extra faces, edges and vertices introduced in this subdivision process must be omitted in the final images.

3.1 Geometric Simplices

The *positive convex hull* of a set of points $v_0, \ldots, v_d$ is the set of all points $\alpha_0 v_0 + \cdots + \alpha_d v_d$ where the weights α_i are strictly positive and add to 1.

A *geometric k-simplex* is the positive convex hull of $k+1$ affinely independent points of $\mathbb{R}^m$, $m \geq k$, called its *corners*. Note that a k-simplex Δ is homeomorphic to the open ball B_k. By taking the positive convex hulls of all non-empty subsets of the corners of Δ, we obtain $2^{k+1} - 1$ pairwise disjoint simplices. The $\binom{k+1}{j+1}$ subsets with $j+1$ corners, $j \leq k$ are called the *j-faces of* Δ. The $(k-1)$-faces are called the *facets* of Δ.

3.2 Topological Triangulation

A k-element c of complex $\mathcal{C}$ is said to be a *topological k-simplex* if there is a continuous map ψ from the closure $\kappa\Delta$ of some geometric k-simplex Δ to the closure of c, such that the restriction of ψ to each j-face f of Δ is a homeomorphism of f to a single element of $\mathcal{C}$. A topological k-simplex c of $\mathcal{C}$ is said to be *proper* if ψ can chosen so that it is itself a homeomorphism (i.e. if the images of the j-faces of Δ are pairwise disjoint) as illustrated in figure 1(a); otherwise c is said to be *degenerate*, as illustrated in figures 1(b) and 1(c).
A *d-dimensional triangulation* is a d-dimensional complex, all of whose elements are topological simplices. A triangulation is *proper* if all its simplices are proper.

3.3 Barycentric Subdivision

The (*topological*) *barycentric subdivision* is a standard refinement $\mathcal{C}^\Delta$ of any complex $\mathcal{C}$ into a proper triangulation. The subdivision of each k-element of $\mathcal{C}$ is defined recursively as follows. For $k = 0$, the subdivision is trivial, i.e. every vertex of $\mathcal{C}$ is a vertex of $\mathcal{C}^\Delta$. For $k > 1$, we recursively decompose every j-element of $\mathcal{C}$, for all $0 \leq j < k$. Then, for each k-element c of $\mathcal{C}$, let ψ be a mapping from κB_k to κc that satisfies the conditions stated in section 2.2. Recall that the inverse image by ψ^{-1} of any j-element e of $\mathcal{C}$ that is contained in the boundary of c is a finite number of disjoint subsets of ∂B_k, each of them homeomorphic to e. For each j-element t of $\mathcal{C}^\Delta$ contained in ∂c, and for each component t' of $\psi^{-1}(t)$,

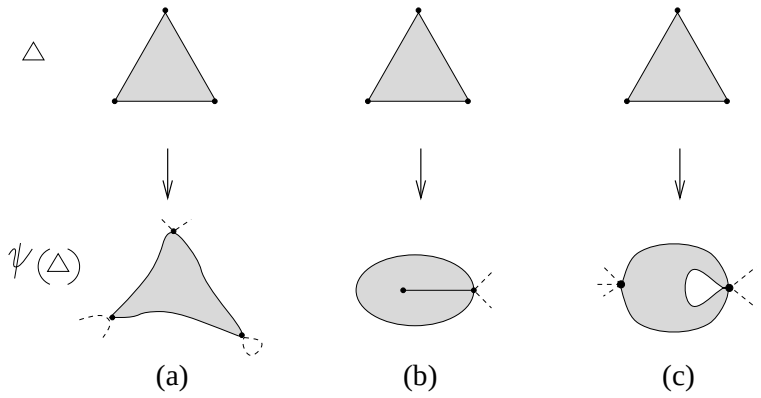


Figure 1: Geometric and Topological 2-simplices: (a) proper, (b) and (c) degenerate.

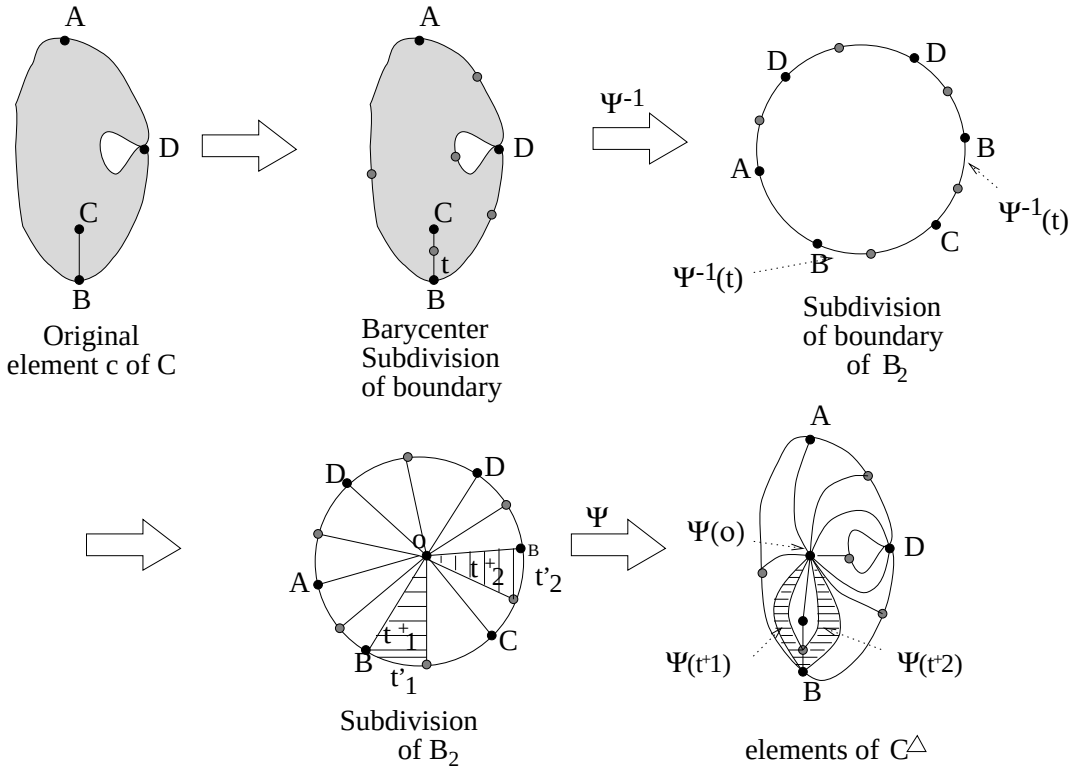


Figure 2: A 2-element (a) and its barycentric subdivision (b).

let t^+ be the union of all open rays that connect the center o of B_k to points in t'. The subdivision of c in $\mathcal{C}^\Delta$ consists of the vertex $\{\psi(o)\}$, and the images of all such sets t^+ by the function ψ. See figure 2.

It can be shown that the barycentric subdivision of a complex $\mathcal{C}$ is essentially unique; i.e. all subdivisions $\mathcal{C}^\Delta$ obtained from a complex $\mathcal{C}$ by different choices of the maps ψ are topologically equivalent.

If $\mathcal{C}$ is a d-dimensional complex, is easy to see that every k-element of $\mathcal{C}^\Delta$ has at most one vertex that belongs to a j-element of $\mathcal{C}$, for each j in $\{0, 1, \ldots, d\}$. It follows that:

Theorem 1 *The barycentric subdivision of a complex $\mathcal{C}$ produces a triangulation $\mathcal{C}^\Delta$ where each k-element has $k+1$ distinct vertices.*

In other words, every element of $\mathcal{C}^\Delta$ is a proper topological simplex and the barycentric subdivision of any complex $\mathcal{C}$ is a *proper* triangulation.

4 Geometric Representation

Having constructed a suitable triangulation $\mathcal{T}$ of the original d-dimensional complex $\mathcal{C}$, we can build a *simplicial representation* of $\mathcal{C}$ by assigning to each (topological) simplex of $\mathcal{T}$ a (geometric) simplex in some space $\mathbb{R}^m$, $m \geq d$, preserving the incidences between elements.

4.1 Configurations

A simplicial representation is completely determined by the coordinates of its vertices $\mathbf{V}\mathcal{T}$. Therefore, we define a *vertex placement* or *configuration* of a triangulation $\mathcal{T}$ as a function $\phi : \mathbf{V}\mathcal{T} \to \mathbb{R}^m$. By extension, for each element e of $\mathcal{T}$ with vertices $\{v_0, \ldots, v_d\}$, we denote by $\phi(e)$ the positive convex hull of the set $\{\phi(v_0), \ldots, \phi(v_d)\}$; and by $\phi(\mathcal{T})$ the collection $\{\phi(e) : e \in \mathcal{T}\}$.

If $m \geq d$, and $\mathcal{T}$ is a proper d-dimensional triangulation, and the vertex positions $\phi(v)$ are in general position, then $\phi(e)$ will be a geometric simplex homeomorphic to e, for every $e \in \mathcal{T}$. Moreover, if elements e_1, e_2 of $\mathcal{T}$ are incident to each other, the same will be true of the corresponding simplices $\phi(e_1), \phi(e_2)$. Therefore, except for possible intersection between its elements, the collection $\phi(\mathcal{T})$ will be a simplicial representation of $\mathcal{C}$, as defined above.

4.2 Geometric Degeneracy

Note that a simplicial representation $\phi(\mathcal{T})$ is useful for visualization only if distinct elements of $\mathcal{T}$ get mapped to distinct geometric simplices. However, if $\mathcal{T}$ contains two distinct j-elements e_1, e_2 that share the same $j+1$ vertices, the simplices $\phi(e_1)$ and $\phi(e_2)$ will *always* coincide, for any vertex placement ϕ.

We call such a pair a *geometric degeneracy* of the triangulation $\mathcal{T}$. It turns out that the barycentric subdivision $\mathcal{C}^\Delta$ of a complex $\mathcal{C}$ may still contain such degeneracies. To resolve them, we need to further subdivide the simplices of $\mathcal{C}^\Delta$. It can be shown that a second barycentric subdivision is enough.

Theorem 2 *Let $\mathcal{C}$ be any 3D complex, and $\mathcal{C}^{\Delta\Delta}$ be the result of applying the barycentric subdivision twice to $\mathcal{C}$. Then any two j-elements of $\mathcal{C}^{\Delta\Delta}$ have at least one vertex not in common.*

Proof. Let be e_1 and e_2 two j-elements of $\mathcal{C}^{\Delta\Delta}$. There are two cases we need to analyze. First, suppose e_1 and e_2 are j-elements contained in two different k-elements f_1, f_2 of $\mathcal{C}^\Delta$ (for any $k \geq j$). By definition, it follows that e_1 has at least one vertex v_1 in the interior of f_1, and similarly e_2 has a vertex v_2 in the interior of f_2. Therefore $v_1 \neq v_2$ and the theorem is proved.

Now suppose that e_1 and e_2 are elements belonging to the same k-element f in $\mathcal{C}^\Delta$. The proof is done by induction in j. For $j = 0$, the theorem is trivial, so let's assume $j > 1$. Since f is a proper topological simplex, when constructing the second barycentric subdivision we can choose the map $\psi : \kappa B_k \to \kappa f$ in such a way that it is a homeomorphism. By definition, e_1 and e_2 are the images under ψ of sets t_1 and t_2, which are the union of rays connecting the center O of B_k to points in t', which are the inverse image of $(j-1)$-elements belonging

to the subdivided boundary of f. By inductive hypothesis any two $(j-1)$-elements of $\mathcal{C}^{\Delta\Delta}$ have at least one vertex not in common, and since ψ is a homeomorphism follows that e_1 and e_2 also have at least one vertex not in common, and so the theorem is proved. □

In light of theorem 2, when $m > d$, almost any assignment ϕ from $\mathbf{V}\mathcal{T}$ to $\mathbb{R}^m$ has the property that $\phi(e) \neq \phi(f)$ wherever $e \neq f$.

4.3 Embedding of a Triangulation

Absence of geometric degeneracies is a necessary but not sufficient condition. A good geometric representation of $\mathcal{C}$ should also be free from spurious self-intersections. In terms of our model, the simplices $\phi(e)$ for all $e \in \mathcal{T}$ should be pairwise disjoint. We call a realization with that property a *simplicial embedding* of $\mathcal{T}$.

Theorem 3 below shows that a simplicial embedding can always be found in a space of sufficiently high dimension:

Theorem 3 *If $\mathcal{T}$ is a d-dimensional triangulation, we can assign to each vertex v of $\mathcal{T}$ a point $\phi(v)$ of $\mathbb{R}^{2d+1}$ in such a way that no two elements of $\phi(\mathcal{T})$ have a point in common.*

This theorem can be proved by assigning to each vertex of $\mathcal{T}$ a distinct point on the *moment curve* $t \to (1, t, t^2, \ldots, t^{2d})$. Specifically, let $v_1, \ldots, v_n$ be the vertices of the triangulation $\mathcal{T}$, and let ϕ be the configuration defined as $\phi(v_i) = (1, i^0, i^1, i^2, \ldots, i^{2d})$. It can be shown that this configuration satisfies the statement of the theorem 3.

This result is only an upper bound, and begs the question of whether simplicial embeddings exist in spaces of dimension lower than $2d+1$. For certain triangulations, $\mathbb{R}^d$ may already suffice. In fact for $d \leq 2$, we can always obtain a simplicial embedding in $\mathbb{R}^{d+1}$ if the complex is orientable, and $\mathbb{R}^{d+2}$ otherwise. We do not know whether these results hold for $d \geq 3$. One must keep in mind, however, that a realization in $\mathbb{R}^4$ with self-intersections may be easier to visualize than an embedding in $\mathbb{R}^7$.

5 Comparing Realizations

We now address the problem of choosing, among all possible configurations $\phi(\mathcal{T})$, one that is particularly "nice" — i.e. that allows the topology of the complex $\mathcal{C}$ to be easily grasped by visual inspection of $\phi(\mathcal{T})$.

5.1 Energy Functions

In order to quantify the intuitive notion of "nice realization" we borrow a concept from the graph-drawing literature, that of an *energy function*—a numerical measure $\mathcal{E}(\phi)$ of some specific visual defects of a particular realization ϕ. An energy function is therefore a map from $(\mathbb{R}^d)^n$ to $\mathbb{R}$, which ideally has its minimum value for configurations ϕ that minimize those particular defects. The energy functions that we have developed try to capture the following criteria:

(A) local flatness of the realization.

(B) homogeneity of size and shape of its elements.

(C) avoidance of self-intersections.

The experience of Rosi[12] for 2D complexes suggests that requirement (B) shoudl be applied to the triangulation $\mathcal{T}$, rather than to the original complex $\mathcal{C}$. The reason being that $\mathcal{C}$ usually gets subdivided more finely in those parts were its topology is more complicated. Therefore, those parts need to be represented at a larger scale in order to make their topological structure more visible.

5.2 Curvature Energy

All visualization mechanisms that we consider require the projection of the chosen realization $\phi(\mathcal{T})$, or a part thereof, onto the ordinary space $\mathbb{R}^3$. In order to minimize the distortion during this step, it is desirable that $\phi(\mathcal{T})$ be as flat as possible, at least in the local sense. In other words, we need to minimize the apparent curvature of $\phi(\mathcal{T})$.

Since each tetrahedron of $\phi(\mathcal{T})$ is contained in an affine subspace of $\mathbb{R}^m$, the curvature of $\phi(\mathcal{T})$ is concentrated on its faces. In general, if t_1, t_2 are two adjacent tetrahedra, their images $\phi(t_1)$ and $\phi(t_2)$ will lie in two distinct 3D spaces U_1, U_2 of $\mathbb{R}^m$, whose intersection is the plane containing the shared face f. To minimize the curvature of $\phi(\mathcal{T})$ on that face, therefore, we need to minimize the angle θ_f between those two subspaces. This requirement is captured by the *curvature energy*, defined as

$$\mathcal{E}^{curv} = \sum_{f \in \mathbf{F}\mathcal{T}} (1 - \cos\theta_f) \tag{1}$$

Note that $1 - \cos\theta_f$ is approximately $\frac{1}{2}\,\theta_f^2$, for small angles. Therefore minimizing $\mathcal{E}^{curv}$ tends to make the angles θ_f as small as possible. The value of $\cos\,\theta_f$ can be computed by the formula $\cos\theta_f = -r_1 \cdot r_2/(|r_1|\,|r_2|)$, where r_i is a vector in U_i, perpendicular to the face f and pointing into the tetrahedron t_i, for $i = 1, 2$.

5.3 Edge Spring Energy

Criterion (B) (element size uniformity) is captured by the *edge spring energy*,

$$\mathcal{E}^{spring} = \sum_{e \in \mathbf{E}\mathcal{T}} K_e \left[\left(\frac{l_e}{L_e}\right)^2 + \left(\frac{L_e}{l_e}\right)^2 - 2 \right] \tag{2}$$

where l_e is the length of edge e in the given configuration, L_e is its "natural" relaxed length (usually set to 1), and K_e is its stiffness (usually 1). The computational cost of evaluating $\mathcal{E}^{spring}$ is linear in the size of $\mathcal{T}$.

Note that each term of formula (2) has minimum value (zero) when l_e is equal to L_e. The value increases approximately like $(l_e/L_e)^2$ for $l_e \gg L_e$, and approximately like $(L_e/l_e)^2$ for $l_e \ll L_e$. Therefore, minimizing $\mathcal{E}^{spring}$ tends to bring all edge lengths close to the natural lengths L_e. This in turn implies other desirable characteristics, such as uniform triangle areas and uniform tetrahedral volumes.

5.4 Long-Range Spring Energy

The energy $\mathcal{E}^{spring}$ keeps adjacent vertices away from each other but does nothing to prevent the configuration from folding over itself.

To address this problem, we can use a more expensive version of the edge spring energy, which was proposed by Kamada for graph drawing on the plane [7]. The idea is to insert a spring between every pair of vertices u, v, adjacent or not:

$$\mathcal{E}^{spring*} = \sum_{u,v \in \mathbf{V}\mathcal{T}, u \neq v} K_{uv} \left[\left(\frac{l_{uv}}{d_{uv}} \right)^2 + \left(\frac{d_{uv}}{l_{uv}} \right)^2 - 2 \right] \tag{3}$$

Here $l_{uv} = |\phi(u) - \phi(v)|$ is the current distance between the vertices u and v; d_{uv} is the graph-theoretical distance between u and v in $\mathcal{T}$; and K_{uv} is the stiffness of the spring (usually set to $1/d_{u,v}^2$).

Note that $\mathcal{E}^{spring*}$ includes all the terms of $\mathcal{E}^{spring}$. This energy is quite effective in preventing intersections between parts of $\phi(\mathcal{T})$ that are far apart in $\mathcal{T}$. However it is fairly expensive, since the number of terms grows quadratically with the size of $\mathcal{T}$.

5.5 Mixed Energy Functions

Since each of the energy functions defined above captures only one aesthetic criterion, its minimum usually fails to meet other important criteria. Therefore one should normally use some combination of all those energies, such as $\mathcal{E} = \alpha_1 \mathcal{E}^{spring} + \alpha_2 \mathcal{E}^{curv} + \cdots$ for some positive weights α_i.

6 Optimization

Having chosen a (mixed) energy function $\mathcal{E}$, the next step is to find a configuration that minimizes it. This is a nontrivial problem, since $\mathcal{E}$ depends on $n\,|\mathbf{V}\mathcal{T}|$ variables, and usually has a large number of local minima.

We use general-purpose non-linear optimization techniques to move from given configuration to a nearby *local* minimum of $\mathcal{E}$. We repeat this search for a specified number of random starting points, and select the local minimum with lowest energy as the answer. Unfortunately, there is no automatic way to tell when the energy is "low enough" to provide a good visualization. So, in practice, we fix a computation budget, and return the best configuration which we can find within its limits.

It should be possible to use combinatorial heuristics, such as simulated annealing, to extend the search beyond the nearest local minimum. However, the experience obtained in the visualization of 2D cellular complexes [12] suggets that such methods will not be efficient enough to be useful.

We use two methods to find a local minimum: *single coordinate* optimization, with periodic diagonal steps [11], and a *gradient descent* method with adaptative stepsize. The single coordinate method consists of optimizing one variable at a time, using Brent's univariate minimization algorithm [11], while all other variables are held fixed after every $n + 1$ such

"axial" steps. We take "diagonal" step along the line connecting the outcomes of the first and the last of the axial steps.

The gradient descent method tries to follow the trajectory $x(t)$ defined by the differential equation $dx/dt = -\nabla f(x(t))$, starting with the initial configuration $x(0)$, until we reach a point where $\nabla f = 0$. Note that this equation forces the configuration to evolve in the direction of greater decrease of energy. We solve this differential equation numerically, using a simple Euler integrator with adaptive stepsize [11].

A drawback of the gradient descent method is the need to compute the gradient of the energy function, which may seem to be excessively costly. However as Baur and Strassen have shown [1], by systematic use of the chain derivation rule we can reduce the cost of computing the gradient of any algebraic or transcendental formula to a small constant times the cost of computing the formula itself.

7 Visualizing 4-Dimensional Objects

In order to visualize the optimized configurations $\phi(\mathcal{T})$ interactively, we have used the *Wire4* program of S. R. Hollasch [6]. *Wire4* first maps the simplicial representation $\phi(\mathcal{T})$ through a parallel or conical (perspective) projection γ from $\mathbb{R}^4$ onto $\mathbb{R}^3$. The resulting 3D object $\gamma(\phi(\mathcal{T}))$ is then rendered as a wireframe model. *Wire4* uses depth cueing to show the distance *in* $\mathbb{R}^4$ from each edge to the 3D space of projection. The 4-dimensional geometry of the object can be grasped by observing the effect of rotations, both in $\mathbb{R}^4$ (before γ) and in $\mathbb{R}^3$ (after γ).

For higher-quality images, we feed the projected simplicial representation $\gamma(\phi(\mathcal{T}))$ to POV-Ray, a freely available ray tracing tool. The elements of $\mathcal{T}$ that are contained in the faces, edges and vertices of $\mathcal{C}$ are modeled by semitrasparent flat triangles, thin cylinders and small spheres, respectively (A further possible improvement would be to use effects such as "fog" and "fire" to make the cells visible too).

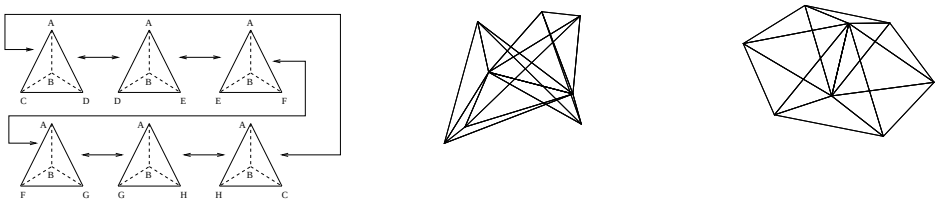
8 Results

We have implemented these ideas in Modula-3 [5]. We used the *facet-edge* data structure of Dobkin and Laszlo [2] to represent the topological structure of 3D complexes and their triangulations. We tested our tool on several 3D complexes, which, once triangulated, ranged in size from a few tens to a few hundred vertices. All tests used $\mathbb{R}^4$ as the modeling space. The initial guess for the optimization was a random configuration, where each vertex coordinate was chosen independently and uniformly in the interval $[-1, +1]$.

The initial complexes were constructed by creating a set of isolated 3-elements with the proper topology, and gluing them in pairs one face at a time.

8.1 Edge-Star Complexes

Figure 3(a) shows a complex consisting of six tetrahedra, glued around a common edge. Each tetrahedron has two unglued faces; the union of those faces constitute the border of the underlying manifold. Figure 3(c) shows a realization of this complex obtained by minimizing the energy function $\mathcal{E}^{spring*}$, limited to 5000 energy evaluations. Note that the final realization is *3-flat*, meaning that it is entirely contained in a single 3D affine subspace of $\mathbb{R}^4$—so it can be projected onto $\mathbb{R}$ on.



(a) (b) (c)

Figure 3: The gluing scheme for a simple complex (a), a random realization (b) and an optimized version of the same (c).

8.2 Hypercube

Figure 4(a) is an incompletely glued model of the 3D complex know as the *hypercube*, with 8 cubic cells. Figures 4(b) and 4(c) show an optimized realization of the hypercube, after barycentric subdivision, rendered with POV-Ray. (The faces, edges and vertices introduced by the subdivision process are not shown.)

The triangular flaps extending out from de edges of model 4(c) are due to the fact that the optimized complex has curved faces in $\mathbb{R}^4$. Figure 4(d) tries to show the analogous effect in the projection of a 2D cube complex with curve faces. Note that the projection of face F extends outside the projection of its boundary a, b, c, d.

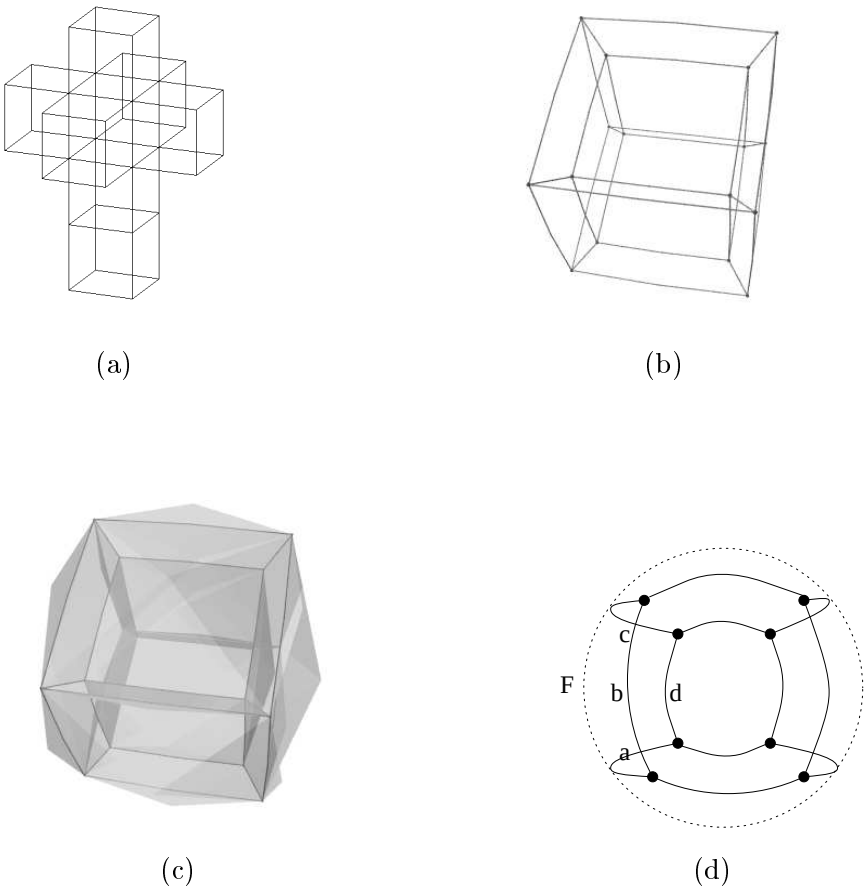
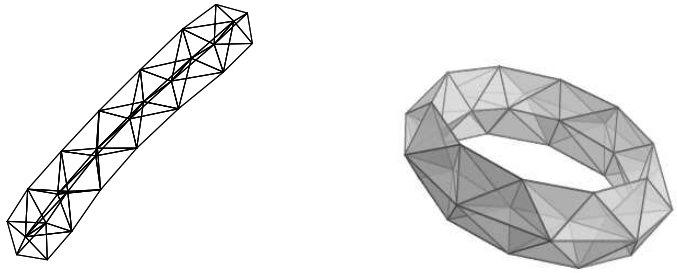


Figure 4: Realization of the hypercube: (a) partially unglued model, (b) wireframe representation, (c) ditto, with partly transparent faces. Figure (d) depicts a cube with curved faces, illustrating the "overflowing" of elements near the edge of the projection.

8.3 Rod and Doughnut

Figure 5 shows how energy optimization distributes any inevitable strain as uniformly as possible. Figure 5(a) shows a triangulation of a rod into 48 tetrahedra. Figure 5(b) shows a triangulation of a doughnut—a 3D solid whose border is a torus obtained by gluing the ends of a similar rod with 60 tetrahedra. In both cases, the optimized configuration is 3-flat, and the tetrahedra are uniform along the length of the object.



(a) (b)

Figure 5: Optimized triangulations of a rod (a) and a doughnut-like solid (b).

9 Conclusions and Future Work

The preceding tests indicate that the automatic visualization of 3D cellular complexes is perfectly feasible, according to methodology presented in this report. These results are encouraging, but there is still a lot of work to be done here. Here are some of the most obvious open problems:

- generate the starting configurations more intelligently; perhaps by the use of specialized heuristics, analogous to those described by Rosi et. al [12].
- understand more about the effects of the various energy functions.
- determine the best way to combine those energy functions. At present we choose the weights α_k by trial and error. Presumably the machine learning approach described by Mendonça and Eades [9] will work here too.
- improve the speed of the optimization procedure, so that the method can be used interactively.
- improve the visualization techniques, e.g. through the use of "insider views" as described by C. Gunn [4].

Acknowledgments. We thank Richard Hollasch for making available his graphical display tool *Wire4*.

References

[1] W. Baur and V. Strassen, The complexity of partial derivatives, *Theoretical Computer Science*, Vol. 22 (1983), 317–330.

[2] D. P. Dobkin and M. J. Laszlo, Primitives for the manipulation of three-dimensional subdivisions. *Algorithmica*, Vol. 4 (1989), 3–32.

[3] T. A. Foley and G. M. Nielson, Practical techniques for producing 3d graphical images, VMEbus Systems, November-December 1987, 65–73.

[4] C. Gunn, Discrete groups and visualization of three-dimensional manifolds. *Computer Graphics Proceedings, Annual Conference Series*, (1993) 255–262.

[5] S. P. Harbison, *Modula-3*. Prentice-Hall (1992).

[6] S. R. Hollasch, *Four-Space Visualization of 4D Objects*. PhD thesis, Arizona State University (August 1991).

[7] T. Kamada and S. Kawai, An algorithm for drawing general undirected graphs. *Information Processing Letters*, Vol. 31 No. 1 (1989), 7–15.

[8] C. Kosniowski, *A First Course in Algebraic Topology*. Cambridge Univ. Press (1980).

[9] C. X. F. Mendonça and P. Eades, Learning aesthetics for visualization. In *Anais do XX Seminário Integrado de Software e Hardware*, Florianópolis, Brazil (1993), 76–88.

[10] A. Paoluzzi, Motion planning + solid modeling = motion modeling. Tech. Rep. 17-89, Dip di Informatica e Sistemistica, Univ. di Roma "La Sapienza," Rome, Italy (1989).

[11] W. H. Press, B. P. Flannery, S. A. Teukolsky and W. T. Vetterling. *Numerical Recipes: The Art of Scientific Computing*. Cambridge Univ. Press (1986).

[12] R. M. Rosi and J. Stolfi, *Automatic visualization of two-dimensional cellular complexes*. Technical Report No IC-96-02, Instituto de Computaçao, Universidade Estadual de Campinas, Brazil (1996).

[13] S. L. Lins, Gems, Computers and Attractors for 3-Manifolds. *Knots and Everything* No. 5, World Scientific (1995).

[14] J. Stillwell, Classical Topology and Combinatorial Group Theory. Springer-Verlag (1993).

[15] W. P. Thurston and J. R. Weeks, The mathematics of three-dimensional manifolds. *Scientific American*, Vol. 251, No. 1 (1984) 94–106.