II.4 Surface Simplification

In applications it is often necessary to simplify the data or its representation. One reason is measurement noise, which we would like to eliminate, another are features, which we look for at various levels of resolution. In this section, we study edge contractions used in simplifying triangulated surface models of solid shapes.

Edge contraction. Suppose K is a triangulation of a 2-manifold without boundary. We recall this means that edges are shared by pairs and vertices by rings of triangles, as depicted in Figure II.9. Let a and b be two vertices and ab the connecting edge in K. By the *contraction* of ab we mean the operation that identifies a with b and removes duplicates from the triangulation. Calling the new vertex c, we get the new triangulation L from K by

- removing ab, abx, and aby;
- substituting c for a and for b wherever they occur in the remaining set of vertices, edges, and triangles;
- removing resulting duplications making sure L is a set.

As a consequence of the operation, there are new incidences between edges and triangles that did not exist in K; see Figure II.9.

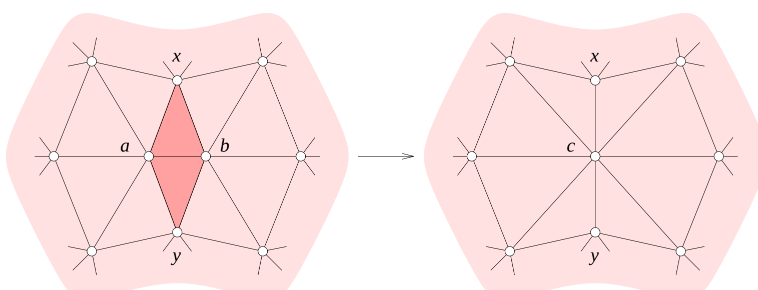

Figure II.9: To contract ab we remove the two dark triangles and repair the hole by gluing their two left edges to their two right edges.

Algorithm. To simplify a triangulation, we iterate the edge contraction operation. In the abstract setting any edge is as good as any other. In a practical situation, we will want to prioritize the edges so that contractions that preserve the shape of the manifold are preferred. To give meaning to this statement, we will define shape to mean the topological type of the surface as well as the geometric form we get when we embed the triangulation in $\mathbb{R}^3$. We will discuss the latter meaning later and for now assume we have a function that assigns to each edge ab a real number $\text{ERROR}(ab)$ assessing the damage the contraction of ab causes to the geometric form. Small non-negative numbers will mean little damage. To write the algorithm, we assume a priority queue stores all edges ordered by the mentioned numerical error assessment. The procedure MINEXTRACT removes the edge with minimum error from the priority queue and returns it. Furthermore, we assume the availability of a boolean test ISSAFE that decides whether or not the contraction of an edge preserves the topological type of the surface.

```
while priority queue is non-empty do
  ab = MinExtract;
  if isSafe(ab) then contract ab endif
endwhile.
```

Some modifications are necessary to recognize edges that no longer belong to the triangulation and to put edges back into the priority queue when they become safe for contraction. Details are omitted. The running time of the algorithm depends on the size of local neighborhoods in the triangulation and on the data structure we maintain to represent it. Under reasonable assumptions the most time-consuming step is the maintenance of the priority queue, which for each step is only logarithmic in the number of edges.

Topological type. We now consider the question whether or not the contraction of an edge preserves the topological type. Define the *link* of an edge ab as the set of vertices that span triangles with ab, and the link of a vertex a as the set of vertices that span edges with a and the set of edges that span triangles with a,

$$\begin{aligned} \mathrm{Lk}\, ab &= \{x \in K \mid abx \in K\}; \\ \mathrm{Lk}\, a &= \{x, xy \in K \mid ax, axy \in K\}. \end{aligned}$$

Since the topological type of K is that of a 2-manifold without boundary, each edge link is a pair of vertices and each vertex link is a closed curve made of edges and vertices in K. Let L be obtained from K by contracting the edge ab. We slightly abuse language by blurring the difference between a triangulation and the topological space it triangulates.

Link Condition Lemma. The triangulations K and L have the same topological type iff $\mathrm{Lk}\, ab = \mathrm{Lk}\, a \cap \mathrm{Lk}\, b$.

In other words, the topological type is preserved iff the links of a and b intersect in exactly two points, namely the vertices x and y in the link of ab, as in Figure II.9.

Proof. We have $\mathrm{Lk}\, ab \subseteq \mathrm{Lk}\, a, \mathrm{Lk}\, b$, by definition. The only possible violation to the link condition is therefore an extra edge or vertex in the intersection of vertex links. If $\mathrm{Lk}\, a$ and $\mathrm{Lk}\, b$ share an edge then the contraction of ab creates a triangle sticking out of the surface, contradicting that L triangulates a 2-manifold. Similarly, if the two vertex links share a vertex $z \notin \mathrm{Lk}\, ab$ then the contraction of ab creates an edge cz that belongs to four triangles, again contradicting that L triangulates a 2-manifold.

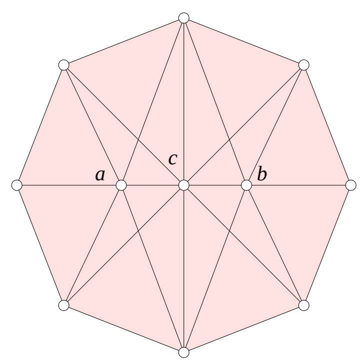

Figure II.10: Mapping the neighborhood of c in L to a triangulated polygon and overlaying it with a similar mapping of the neighborhoods of a and b in K.

To prove the other direction, we draw the link of c in L as a convex polygon in $\mathbb{R}^2$; see Figure II.10. Using Tutte's Theorem from Chapter I, we can decompose the polygon by drawing the triangles incident to c in L. Similarly, we can decompose the polygon by drawing the triangles incident to a or b in K. We superimpose the two triangulations and refine to get a new triangulation, if necessary. The result is mapped back to K and to L, effectively refining the neighborhoods of a and b in K and that of c in L. The link of c and everything outside that link is untouched by the contraction. Hence on an outside the link K and L are the same and inside the link K and L are now isomorphic by refinement. It follows that K and L are isomorphic and therefore have the same topological type. $\sqcup\!\!\!\!\sqcap$

Square distance. To talk about the geometric meaning of shape we now assume that K is embedded in $\mathbb{R}^3$, with straight edges and flat triangles. To develop an error measure we use the planes spanned by the triangles. Letting $u \in \mathbb{S}^2$ be the unit normal of a plane h and $\delta \in \mathbb{R}$ its off-set, we can write h as the set of points $p \in \mathbb{R}^3$ for which $\langle p, u \rangle = -\delta$. Using matrix notation for the scalar product the *signed distance* of a point $x \in \mathbb{R}^3$ from h is

$$d(x,h) \;=\; (x-p)^T \cdot u \;=\; x^T \cdot u + \delta.$$

Defining $\mathbf{x}^T = (x^T, 1)$ and $\mathbf{u}^T = (u^T, \delta)$ we can write this as a four-dimensional scalar product, $\mathbf{x}^T \cdot \mathbf{u}$. We use this to express the sum of square distances from a set of planes in matrix form. Letting H be a finite set of planes, this gives a function $E_H : \mathbb{R}^3 \to \mathbb{R}$ defined by

$$\begin{aligned} E_H(x) &= \sum_{h_i \in H} d^2(x, h_i) \\ &= \sum_{h_i \in H} (\mathbf{x}^T \cdot \mathbf{u}_i)(\mathbf{u}_i^T \cdot \mathbf{x}) \\ &= \mathbf{x}^T \cdot \left(\sum_{h_i \in H} \mathbf{u}_i \cdot \mathbf{u}_i^T \right) \cdot \mathbf{x}. \end{aligned}$$

Hence $E_H(x) = \mathbf{x}^T \cdot \mathbf{Q} \cdot \mathbf{x}$, where

$$\mathbf{Q} \;=\; \sum_{h_i \in H} (\mathbf{u}_i \cdot \mathbf{u}_i^T) \;=\; \begin{bmatrix} A & P & Q & U \\ P & B & R & V \\ Q & R & C & W \\ U & V & W & Z \end{bmatrix}$$

is a symmetric, four-by-four matrix we refer to as the *fundamental quadric* of the map E_H. Writing $x^T = (x_1, x_2, x_3)$ we get

$$\begin{aligned} E_H(x) \;=\; & Ax_1^2 + Bx_2^2 + Cx_3^2 + 2(Px_1x_2 + Qx_1x_3 + Rx_2x_3) \\ & + 2(Ux_1 + Vx_2 + Wx_3) + Z. \end{aligned}$$

We see that E_H is a quadratic map that is non-negative and unbounded. Essential to the efficient implementation of the error assessment is that for disjoint unions, $H = H_1 \cup H_2$, the quadrics can be added giving $\mathbf{Q} = \mathbf{Q}_1 + \mathbf{Q}_2$.

Error assessment. In the application, we are interested in measuring the damage to the geometric form caused by contracting the edge ab to the new vertex c. We think of the operation as a map between vertices, $\varphi : \mathrm{Vert}\, K \to$

$\mathrm{Vert}\,L$, defined by $\varphi(a) = \varphi(b) = c$ and $\varphi(x) = x$ for all $x \neq a, b$. Letting K_0 be the initial triangulation, we obtain L by a sequence of edge contractions giving rise to a composition of vertex maps, which is again a vertex map, $\varphi_0 : \mathrm{Vert}\,K_0 \to \mathrm{Vert}\,L$. The vertices in $V_c = \varphi_0^{-1}(c) \subseteq \mathrm{Vert}\,K_0$ all map to c and we let H be the set of planes spanned by triangles in K_0 incident to at least one vertex in V_c. Finally, we define the *error* of the contraction of ab as the minimum, over all possible placements of c as a point in $\mathbb{R}^3$, of the sum of square distances from the planes,

$$\textsc{Error}(ab) \quad = \quad \min_{c \in \mathbb{R}^3} E_H(c).$$

For generic sets of planes, this minimum is unique and easy to compute. The gradient of $E = E_H$ at a point x is the vector of steepest increase, $\nabla E(x) = (\frac{\partial E}{\partial x_1}(x), \frac{\partial E}{\partial x_2}(x), \frac{\partial E}{\partial x_3}(x))$. It is zero iff x minimizes E. The derivative with respect to x_i can be computed using the multiplication rule,

$$\begin{aligned} \frac{\partial E}{\partial x_i} &= \frac{\partial \mathbf{x}^T}{\partial x_i} \cdot \mathbf{Q} \cdot \mathbf{x} + \mathbf{x}^T \cdot \mathbf{Q} \cdot \frac{\partial \mathbf{x}}{\partial x_i} \\ &= \mathbf{Q}_i^T \cdot \mathbf{x} + \mathbf{x}^T \cdot \mathbf{Q}_i, \end{aligned}$$

where $\mathbf{Q}_i^T$ is the i-th row of $\mathbf{Q}$. The point $c \in \mathbb{R}^3$ that minimizes E can thus be computed by setting $\frac{\partial E}{\partial x_i}$ to zero, for $i = 1, 2, 3$, and solving the resulting system of three linear equations.

It remains to compute the quadric $\mathbf{Q}$. The simplest strategy is to store a quadric $\mathbf{Q}_a$ for every vertex a in K. Initially, in K_0, this is the quadric defined by the triangles sharing a. To evaluate the contraction of the edge ab we use $\mathbf{Q} = \mathbf{Q}_a + \mathbf{Q}_b$, and we store $\mathbf{Q}_c = \mathbf{Q}$ with the new vertex c. Since the sets of planes that contribute to $\mathbf{Q}_a$ and $\mathbf{Q}_b$ are not disjoint, the quadric $\mathbf{Q}_c$ is not exactly what we promised. More precisely, a triangle in K_0 that has i vertices in the preimage V_c of c contributes i times to $\mathbf{Q}_c$. Since the only possibilities are $i = 0, 1, 2, 3$ the difference to the quadric we promised is small and the effect on the computed simplification seems to be insignificant in practice.

Bibliographic notes. The algorithm described in this section is essentially the surface simplification algorithm by Garland and Heckbert [3]. It combines the idea of using edge contraction, which can be found in earlier computer graphics papers, with the particular error measure remembering the original form through accumulated quadrics. The test for maintaining the topological type has been added later and more general versions of the Link Condition Lemma can be found in [1]. A version of the algorithm that maintains the quadrics without double counting using inclusion-exclusion is give in [2].

[1] T. DEY, H. EDELSBRUNNER, S. GUHA AND D. V. NEKHAYEV. Topology preserving edge contraction. *Publ. Inst. Math. (Beograd) (N.S.)* **66** (1999), 23-45.

[2] H. EDELSBRUNNER. *Geometry and Topology for Mesh Generation.* Cambridge Univ. Press, England, 2001.

[3] M. GARLAND AND P. S. HECKBERT. Surface simplification using quadric error metrics. *Computer Graphics*, Proc. SIGGRAPH, 1997, 209–216.