Parameterization

Michael S. Floater

November 10, 2011

Triangular meshes are often used to represent surfaces, at least initially, one reason being that meshes are relatively easy to generate from point cloud data. However, we often want a smoother surface representation, and hence the need arises to fit a smooth parametric surface through the vertices of the mesh. This requires making a suitable parameterization. Parameterizations are also useful for texture mapping and other processes in computer graphics. In this lecture we review some parameterization methods.

1 Parameterization of polygons

A standard approach to fitting a smooth parametric curve $\mathbf{c}(t)$ through a sequence of points $\mathbf{x}_i = (x_i, y_i, z_i) \in \mathbb{R}^3$, $i = 0, 1, \ldots, n$, is to start by choosing a *parameterization*, a corresponding increasing sequence of parameter values $t_0 < t_1 < \cdots < t_n$. Then by finding smooth functions $x, y, z : [t_0, t_n] \to \mathbb{R}$ for which $x(t_i) = x_i$, $y(t_i) = y_i$, $z(t_i) = z_i$, an interpolatory curve $\mathbf{c}(t) = (x(t), y(t), z(t))$ results, i.e., a curve $\mathbf{c} : [t_0, t_n] \to \mathbb{R}^3$ such that $\mathbf{c}(t_i) = \mathbf{x}_i$ for each i.

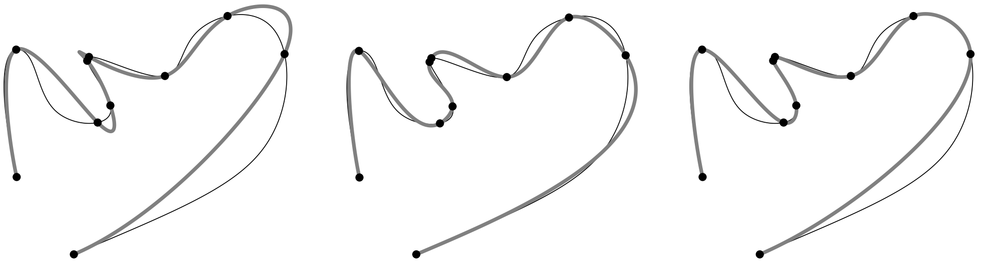
In order to discuss different parameterization methods, we will use the example of C^2 cubic spline interpolation. There is a unique C^2 cubic spline curve $\mathbf{s} : [t_0, t_n] \to \mathbb{R}^d$ such that

$$\mathbf{s}(t_i) = \mathbf{x}_i, \qquad i = 0, 1, \ldots, n,$$

and

$$\mathbf{s}'(t_i) = \mathbf{m}_i, \qquad i = 0, n, \tag{1}$$

for some chosen vectors $\mathbf{m}_0$, $\mathbf{m}_n$. By 'cubic spline curve' we understand that $\mathbf{s}$ is a cubic polynomial curve in each interval $[t_i, t_{i+1}]$, and that $\mathbf{s}$ has C^2 continuity at the break points $t_1, \ldots, t_{n-1}$. The end conditions (1) are sometimes



(a) uniform (b) chordal (c) centripetal

Figure 1: Choice of parameterization for cubic spline interpolation.

known as ‘clamped’ end conditions, and this kind of spline interpolation is sometimes called ‘complete’ spline interpolation.

A typical way to choose parameter values is to set $t_0 = 0$ and recursively let

$$t_{i+1} := t_i + L_i, \tag{2}$$

for some chosen interval lengths $L_0, L_1, \ldots, L_{n-1} > 0$. The simplest choice is the *uniform parameterization* defined by $L_i = 1$, so that the values t_i are uniformly spaced; see Figure (1). But as early as 1967, Ahlberg, Nilson, and Walsh proposed using the *chordal parameterization* in which L_i is taken to be the length of the line segment (‘chord’) $[\mathbf{x}_i, \mathbf{x}_{i+1}]$, i.e.,

$$L_i := \|\mathbf{x}_{i+1} - \mathbf{x}_i\|, \tag{3}$$

where $\| \cdot \|$ is the Euclidean norm in $\mathbb{R}^d$. The motivation behind this is that if the points $\mathbf{x}_i$ are samples from some (typically unknown) curve and if that curve is smooth enough and the points are relatively densely sampled, then the chord length $\|\mathbf{x}_{i+1} - \mathbf{x}_i\|$ is a good approximation to the arc length of the curve between $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$.

In practice, the chordal parameterization often does a better job of avoiding the cusps and self-intersections that sometimes occur with the uniform parameterization when there is a lot of variation in the distances between the points $\mathbf{x}_i$.

Later, it was observed that the uniform and chordal parameterizations are the special cases $\mu = 0$ and $\mu = 1$ of the more general parameterization

$$L_i := \|\mathbf{x}_{i+1} - \mathbf{x}_i\|^{\mu},$$

with $0 \leq \mu \leq 1$ acting as a kind of blending parameter. The choice $\mu = 1/2$ was termed by Lee the *centripetal parameterization* and it has been found that it tends to lead to a spline curve that stays close to the polygon defined by the $\mathbf{x}_i$.

Consider now how we might generalize the incremental method (2) to points arranged in a triangular mesh, with a view to computing parameter points for surface fitting. One approach is to reformulate (2) as a linear system of equations, with t_0 and t_n acting as 'boundary conditions'. Since

$$t_i - t_{i-1} = L_{i-1}, \qquad t_{i+1} - t_i = L_i,$$

for $i = 1, \ldots, n-1$, we have the equations

$$t_i = \left(\frac{L_i}{L_{i-1} + L_i} \right) t_{i-1} + \left(\frac{L_{i-1}}{L_{i-1} + L_i} \right) t_{i+1}, \qquad i = 1, \ldots, n-1.$$

Thus, for general L_i, these equations simply force the parameter value t_i to be some convex combination of the neighbouring values t_{i-1} and t_{i+1}, and therefore, lie between them. We also now see that the chordal parameterization has the 'linear precision' property: if for any i,

$$\mathbf{x}_i = (1-\lambda)\mathbf{x}_{i-1} + \lambda \mathbf{x}_{i+1},$$

for some λ, $0 < \lambda < 1$, which means that $\mathbf{x}_{i-1}$, $\mathbf{x}_i$, and $\mathbf{x}_{i+1}$ lie in a straight line, then

$$t_i = (1-\lambda) t_{i-1} + \lambda t_{i+1}.$$

This is a property that we can mimic in the surface case.

2 Parameterization of triangular meshes

We now describe a general method for constructing a parameterization of triangular mesh in $\mathbb{R}^3$. We denote by $\mathcal{S}$ the set of triangles in the mesh and V its vertices and E its edges. We let $\Omega_{\mathcal{S}} \subset \mathbb{R}^3$ be the union of the triangles in $\mathcal{S}$. Then we define a parameterization of $\mathcal{S}$ as a continuous piecewise linear mapping $\psi : \Omega_{\mathcal{S}} \to \mathbb{R}^2$. Then ψ maps each vertex, edge, and triangle of $\mathcal{S}$ to a corresponding vertex, edge, and triangle in $\mathbb{R}^2$. Such a mapping is completely determined by the points $\psi(\mathbf{v})$, $\mathbf{v} \in V$. Let V_I denote the interior vertices of $\mathcal{S}$ and V_B the boundary ones. The boundary vertices of $\mathcal{S}$ form a polygon $\partial\mathcal{S}$ in $\mathbb{R}^3$ which we call the *boundary polygon* of $\mathcal{S}$. Two distinct

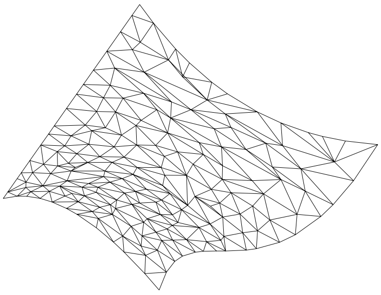


Figure 2: Triangular mesh in $\mathbb{R}^3$

vertices $\mathbf{v}$ and $\mathbf{w}$ in $\mathcal{S}$ are *neighbours* if they are the end points of some edge in $\mathcal{S}$. For each $\mathbf{v} \in V$, let

$$N_{\mathbf{v}} = \{\mathbf{w} \in V : [\mathbf{w}, \mathbf{v}] \in E\},$$

the set of neighbours of $\mathbf{v}$, where $E = E(\mathcal{S})$ is the set of edges in $\mathcal{S}$.

The first step of the method is to choose any points $\psi(\mathbf{v}) \in \mathbb{R}^2$, for $\mathbf{v} \in V_B$, such that the boundary polygon $\partial\mathcal{S}$ of $\mathcal{S}$ is mapped into a simple polygon $\psi(\partial\mathcal{S})$ in the plane. In the second step, for $\mathbf{v} \in V_I$, we choose a set of strictly positive values $\lambda_{\mathbf{vw}}$, for $\mathbf{w} \in N_{\mathbf{v}}$, such that

$$\sum_{\mathbf{w} \in N_{\mathbf{v}}} \lambda_{\mathbf{vw}} = 1. \tag{4}$$

Then we let the points $\psi(\mathbf{v})$ in $\mathbb{R}^2$, for $\mathbf{v} \in V_I$, be the unique solutions of the linear system of equations

$$\psi(\mathbf{v}) = \sum_{\mathbf{w} \in N_{\mathbf{v}}} \lambda_{\mathbf{vw}} \psi(\mathbf{w}), \qquad \mathbf{v} \in V_I. \tag{5}$$

Since these equations force each point $\psi(\mathbf{v})$ to be a convex combination of its neighbouring points $\psi(\mathbf{w})$, we call ψ a *convex combination mapping.* Fig. 2 shows an example of a triangular mesh in $\mathbb{R}^3$. Fig. 3 shows a convex combination mapping of $\mathcal{S}$ into a planar mesh $\mathcal{T}$, whose boundary was chosen to be a rectangle. Fig. 3 also shows a tensor-product spline approximation (in fact a least square approximation) to the vertices of $\mathcal{S}$ based on their parameter points, the vertices of $\mathcal{T}$.

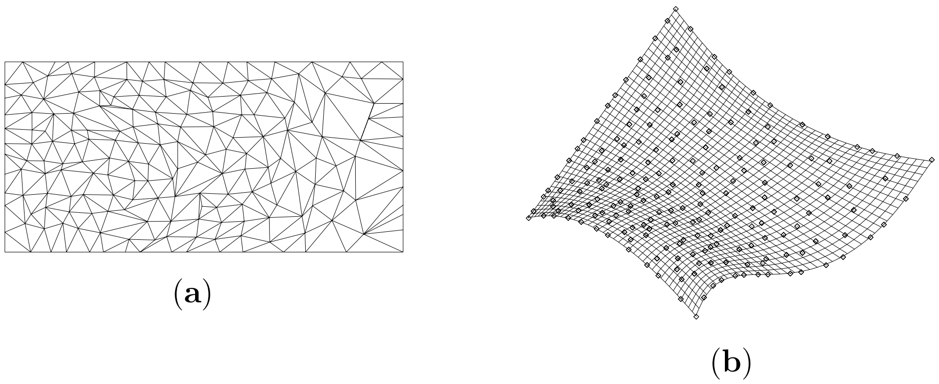


Figure 3: Parameterization (triangular mesh in $\mathbb{R}^2$) (**a**) and resulting tensor-product spline approximation (**b**)

Let us take a closer look at the linear system. We must show that it has a unique solution. To this end, note that it can be rewritten in the form

$$\psi(\mathbf{v}) - \sum_{\mathbf{w} \in N_{\mathbf{v}} \cap V_I} \lambda_{\mathbf{vw}} \psi(\mathbf{w}) = \sum_{\mathbf{w} \in N_{\mathbf{v}} \cap V_B} \lambda_{\mathbf{vw}} \psi(\mathbf{w}), \qquad \mathbf{v} \in V_I. \tag{6}$$

This can be written as the matrix equation

$$A\mathbf{x} = \mathbf{b},$$

where $\mathbf{x} = (\psi(\mathbf{w}))_{\mathbf{w} \in V_I}$ is the column vector of unknowns in some arbitrary ordering, $\mathbf{b}$ is the column vector whose elements are the right hand sides of (6), and the matrix $A = (a_{\mathbf{vw}})_{\mathbf{v},\mathbf{w} \in V_I}$ has dimension $n \times n$, with $n = |V_I|$, and elements

$$a_{\mathbf{vw}} = \begin{cases} 1, & \mathbf{w} = \mathbf{v}, \\ -\lambda_{\mathbf{vw}}, & \mathbf{w} \in N_{\mathbf{v}}, \\ 0, & \text{otherwise}. \end{cases}$$

The existence and uniqueness of the solution to (5) follows from the structure of the matrix A, namely that its off diagonal elements are either zero or negative and each row of A is diagonally dominant. Moreover every row corresponding to a vertex $\mathbf{v} \in V_I$ which has at least one neighbour in V_B is strictly diagonally dominant and every interior vertex can be connected to the boundary by a path of vertices. A standard result in linear algebra shows then that A is non-singular (in fact A is a so-called M-matrix, and

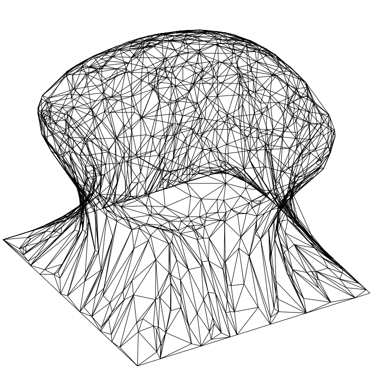


Figure 4: Triangular mesh in $\mathbb{R}^3$

such matrices frequently occur in numerical approximations to elliptic partial differential equations.

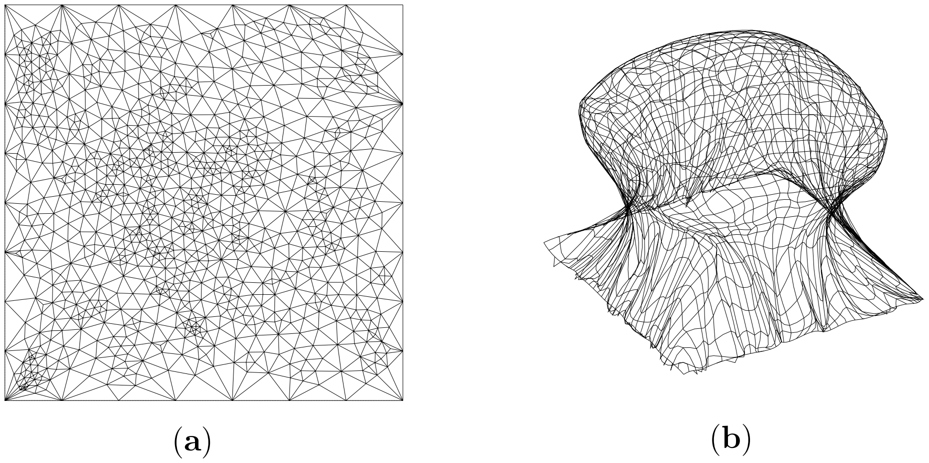
An interesting question is whether ψ is one-to-one. It will not be one-to-one in general but the following result gives a sufficient condition. The proof is beyond the scope of this course. We say that an interior edge of $\mathcal{S}$ is a *dividing edge* of $\mathcal{S}$ if both its end points are boundary vertices of $\mathcal{S}$.

Theorem 1 *If $\psi(\partial\Omega)$ is convex and no dividing edge $[v, w]$ is mapped by ψ into $\partial\Omega$ then ψ is injective.*

3 Choosing the Weights

A simple choice of weights $\lambda_{\mathbf{vw}}$ is to take them to be uniform, i.e., constant for each vertex $\mathbf{v}$, so that $\lambda_{\mathbf{vw}} = 1/d(\mathbf{v})$, $\mathbf{w} \in N_{\mathbf{v}}$ where $d(\mathbf{v})$ is the degree $|N_{\mathbf{v}}|$ of $\mathbf{v}$. Then every interior vertex $\psi(\mathbf{v})$ of the solution to the linear system will be the barycentre of its neighbours. However, numerical examples show that this uniform parameterization usually leads to poor spline surfaces when used for approximation. Look for example at Figures 4 and 5. Figure 4 shows a mesh $\mathcal{S}$ and Figure 5 shows a uniform parameterization $\mathcal{T}$ of $\mathcal{S}$ together with a spline approximation to $\mathcal{S}$ over the mesh $\mathcal{T}$. Clearly the iso-curves are badly behaved.

One reason for the bad behaviour of the surface approximation is that the weights $\lambda_{\mathbf{vw}}$ are independent of the geometry of the vertices $\mathbf{v}$ of $\mathcal{S}$. In practice it better to choose weights that share the linear precision property



(**a**) (**b**)

Figure 5: Uniform parameterization (**a**) and spline approximation (**b**)

of the chordal parameterization for curves. We would like weights $\lambda_{\mathbf{vw}}$ that have the property that when $\mathbf{v}$ and its neighbours lie in a plane, then

$$\mathbf{v} = \sum_{\mathbf{w} \in N_{\mathbf{v}}} \lambda_{\mathbf{vw}} \mathbf{w}. \tag{7}$$

Weights that achieve this and are also positive are the mean value weights:

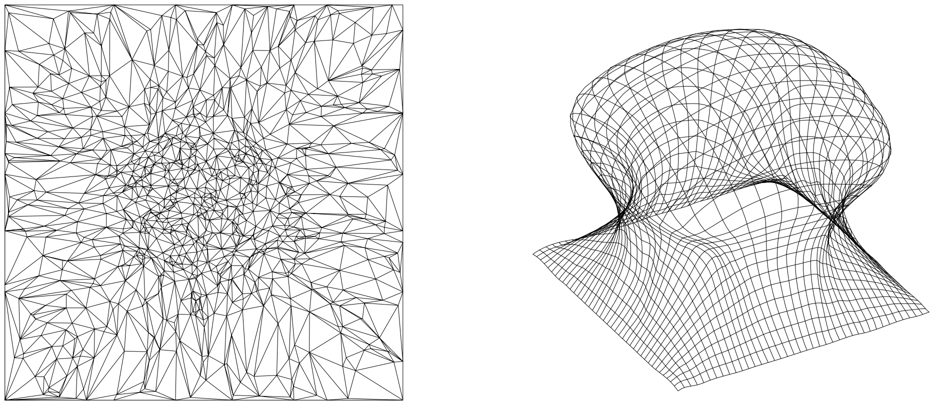
$$\lambda_{\mathbf{vw}} = w_{\mathbf{vw}} / \sum_{\mathbf{u} \in N_{\mathbf{v}}} w_{\mathbf{vu}},$$

where

$$w_{\mathbf{vw}} = \frac{\tan(\alpha/2) + \tan(\beta/2)}{||\mathbf{w} - \mathbf{v}||},$$

and α and β are the angles at the vertex $\mathbf{v}$ of the two triangles adjacent to the edge $[\mathbf{v}, \mathbf{w}]$. Figure 6 shows the result of interpolating $\mathcal{S}$ of Figure 4 with a spline approximation over the mean value parameterization of $\mathcal{S}$. The surface approximation is clearly better than that of Figure 5, using uniform parameterization. The effect of choosing different boundary polygons $\psi(\partial\mathcal{S})$ is shown in Figures 7 and 8. In Fig. 8(**a**) the parameter points of the boundary vertices of the mesh in 7 were distributed by chord length on a rectangle and in Fig. 8(**b**) on a circle.

The mean value parameterization has turned out to be a successful and popular method for parameterizing triangular meshes, both for surface fitting and for texture mapping in computer graphics.



(**a**) (**b**)

Figure 6: Mean value parameterization (**a**) and spline approximation (**b**)

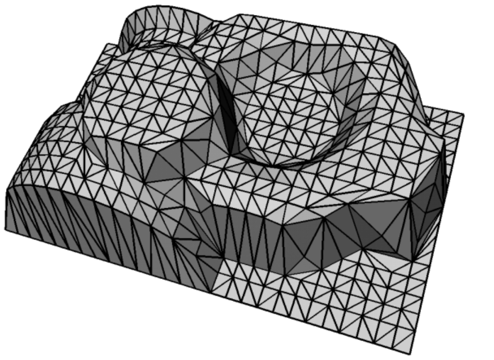
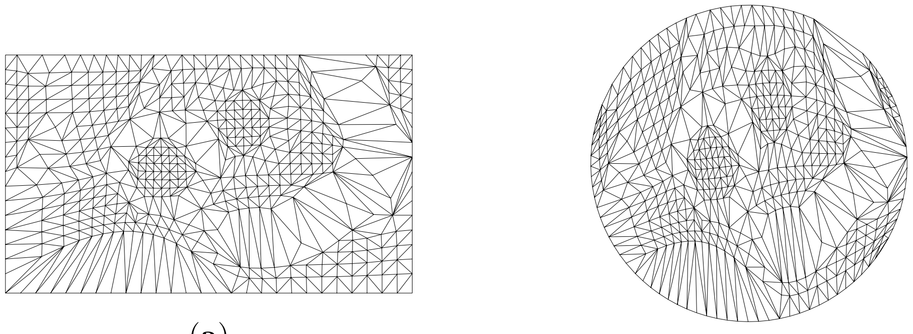


Figure 7: Triangular mesh in $\mathbb{R}^3$



(**a**) (**b**)

Figure 8: Mean value parameterizations of the mesh in Figure 7