Chapter 13

Surface modeling

Introduction

The aim of this chapter is to review some methods for surface definition. A surface can be defined using various categories of methods, and, in particular, we find parametric, implicit or explicit surfaces.

A parametric surface, given u and v two parameters living in some interval, is the data of a function σ whose values $\sigma(u, v)$ describe the surface. Implicit surfaces are given by a relation like $f(x, y, z) = 0$ while explicit surfaces take the form $z = f(x, y)$.

In the first sections of this chapter we discuss the methods extensively used in practice (for instance, in CAD systems), *i.e.*, parametric surfaces. Most methods developed for surface definition are derived from methods used for curve definition. Thus, the material discussed in Chapter 12 will be reviewed here and extended to the case of surfaces.

★
★ ★

First, we introduce the basic notions related to surface definitions (as we did for curves in Chapter 12) to give a rough idea of the various methods that can be used to define (construct) a surface. It is not our intention to be exhaustive, and we refer the reader to the *ad-hoc* literature. In this respect, references listed in Chapter 12 for curve topics remain relevant since curves and surfaces are strongly related.

Surface definitions can be classified into several categories depending on what the surface looks like or what the geometry of the surface must be. Particular surfaces are briefly discussed including surfaces of *revolution*, *ruled* surfaces and *sweep* surfaces. We then turn to surfaces that can be defined by means of a tensor product and we discuss surfaces related to an interpolation scheme based on a set of points (for instance, Coons patches). Then, we look at tensor product based patches based on a control polyhedron. We briefly examine rational patches before turning to patches that are not included in the above categories. In specific we mention the case of patches with an arbitrary topology or those whose control differs from the classical form. To end, as we did for the curves, we consider the case of *composite* surfaces which are widely used in CAD systems. The initial surface is split into different portions and each of these is defined using one of the surface representation methods. In addition, some specific properties about regularity can be required at the junction of two such portions.

Notations. Parameters used for surface definitions are commonly denoted by u and v (together with w in the case of a triangular patch) unlike that used for curve definitions which was denoted by t in Chapter 12. Nevertheless, in this chapter a curve parameter will be denoted by u (or v) as well since such a curve is now seen as a component of a surface with such parameters. Recall that curves, Γ, are noted as $\gamma(.)$ while surfaces, Σ, will be denoted by $\sigma(.,.)$. While, for instance, $\sigma(u,v)$ is a point on surface Σ, for a given pair (u,v), in some cases, we make no distinction between $\sigma(u,v)$ (where (u,v) varies) and the surface Σ.

13.1 Specific surfaces

Numerous surfaces commonly used in industry have a specific geometric nature. This results from the fact that they must be manufactured. Owing to this, the surfaces must respect certain constraints regarding their manufacture, e.g. have a degree of detail that is compatible with the precision of the tool used.

13.1.1 Surfaces of revolution

Some particular surfaces can be defined by means of the revolution of a two-dimensional entity (point[1], line, planar curve, closed or open polygon) around an axis in space. The position of the line (curve) with respect to the axis leads to various surfaces such as cylinders, truncated cones, solid discs, hyperboloids of one sheet, cones with a cylindrical hole, spheres, ellipsoids, torus, etc.

The entity to be rotated is a function of one parameter, the rotation is determined by another parameter (namely the rotation angle), so a *surface of revolution* is a biparametric function since a point on this surface is specified by two parameters.

Figure 13.1 (left-hand side) demonstrates an example of a surface of revolution. One can see the curve $\gamma(u)$, the rotation axis and, for a given value v of the angle, the point $\sigma(u,v)$ in the surface.

13.1.2 Trimmed surfaces

Given a point $P(u)$ function of a parameter u and a vector $\vec{V}(u)$ such that $\vec{V}(u) \neq 0, \forall u$, then the relationship :

$$\sigma(u,v) = P(u) + v\,\vec{V}(u)\,, \tag{13.1}$$

[1] In this case a curve results from the revolution.

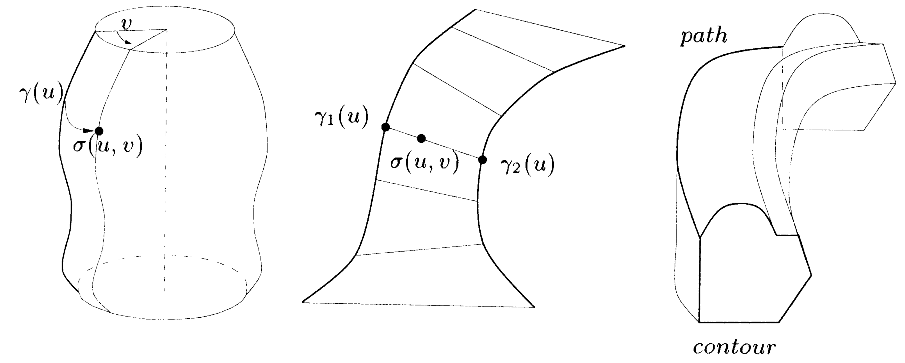

Figure 13.1: *Examples of particular surfaces. Left-hand side, a surface of revolution, middle, a ruled surface and, right-hand side, an extruded surface.*

defines a *ruled* surface.

Given two curves Γ_1 and Γ_2 and the corresponding parameterization, $\gamma_1(u)$ and $\gamma_2(u)$, a surface can be defined as

$$\sigma(u,v) = (1-v)\,\gamma_1(u) + v\,\gamma_2(u)\,.$$

This surface is also a *ruled surface*. In fact, it conforms to the general definition of a ruled surface with $P(u) = \gamma_1(u)$ et $\vec{V}(u) = \gamma_2(u) - \gamma_1(u)$.

In Figure 13.1 (middle) is an example of a ruled surface. One can see, for instance, the two curves $\gamma_1(u)$ and $\gamma_2(u)$ together with the net linking them, *i.e.*, for a given v, the point $\sigma(u,v)$ of the thus defined surface.

13.1.3 Extruded or sweeping surfaces

Given a path and a two-dimensional entity, a surface is obtained by traversing this entity along the given path (a line, a curve). In this way, the defined surface is a *sweeping surface* or an *extruded surface*.

Note that an entity like a line results in a ruled surface as defined in the previous section. A simple example of an extruded surface is given in Figure 13.1 (right-hand side).

13.2 Interpolation-based surfaces

We turn now to arbitrary surfaces which nevertheless are suitable for a tensor product or an interpolation based definition. The easiest way to define a surface is to extend the material used for curve definition (Chapter 12). This leads to quadrilateral patches that can be considered as the *tensor product* of two curve definitions. Based on the way the curves are defined, we obtain various surface definitions. To give a basic feeling of a tensor product based method, we take a rather simple case, namely a bilinear interpolation.

13.2.1 Tensor product based patches (introduction)

We consider a parametric space in $\mathbb{R}^2$ whose parameters u and v live in $[0,1]$. Given four points in $\mathbb{R}^3$, $P_{i,j}$ for $i=0,1$ and $j=0,1$ where $P_{i,j}$ corresponds to $u=i$ and $v=j$, we can define four lines :

$$\text{for } v=0,\ \gamma(u)=(1-u)\,P_{0,0}+u\,P_{1,0}=\sum_{i=0}^{1} B_i^1(u)\,P_{i,0}\,,$$

$$\text{for } v=1,\ \gamma(u)=(1-u)\,P_{0,1}+u\,P_{1,1}=\sum_{i=0}^{1} B_i^1(u)\,P_{i,1}\,,$$

$$\text{for } u=0,\ \gamma(v)=(1-v)\,P_{0,0}+v\,P_{0,1}=\sum_{j=0}^{1} B_j^1(v)\,P_{0,j}\,,$$

$$\text{for } u=1,\ \gamma(v)=(1-v)\,P_{1,0}+v\,P_{1,1}=\sum_{j=0}^{1} B_j^1(v)\,P_{1,j}\,,$$

with $B_0^1(u)=1-u$, $B_1^1(u)=u$ and similar expressions for $B_j^1(v)$ and where the ^{1} in B_j^1 refers to the degree of the function. Combining these curves, we can construct a surface $\sigma(u,v)$ defined by :

$$\sigma(u,v)=\sum_{i=0}^{1}\sum_{j=0}^{1} B_i^1(u)\,B_j^1(v)\,P_{i,j}\,, \qquad (13.2)$$

which is the tensor product based on the above curve definitions. In matrix form, such a curve Γ can be written, for each component, as :

$$\gamma(t)=[\mathcal{U}][\mathcal{M}][\mathcal{P}] \quad \text{or} \quad \gamma(t)=[\mathcal{B}(u)][\mathcal{P}]\,,$$

where $[\mathcal{U}]=[1-u,u]$ is the basis functions of the representation,

$$[\mathcal{M}]=\begin{pmatrix}1&0\\0&1\end{pmatrix}$$

is the coefficient matrix of the function in the above basis (for one of the above curves) and $[\mathcal{P}]={}^t[P_{0,0}\,,\,P_{1,0}]$ is the row of the control points (for the first curve) and, finally, $[\mathcal{B}(u)]=[\mathcal{U}][\mathcal{M}]$.

Similarly, the surface of Equation (13.2) can be seen in matrix form as :

$$\sigma(u,v)=[\mathcal{U}][\mathcal{M}][\mathcal{P}]\,{}^t[\mathcal{M}]\,{}^t[\mathcal{V}] \quad \text{or again} \quad \sigma(u,v)=[\mathcal{B}(u)][\mathcal{P}]\,{}^t[\mathcal{B}(v)]\,, \qquad (13.3)$$

where, now, we have :

$$[\mathcal{P}]=\begin{bmatrix}P_{0,0}&P_{0,1}\\P_{1,0}&P_{1,1}\end{bmatrix},$$

and $[\mathcal{B}(u)]$ is the above matrix form of the (curve) function. As a consequence, any tensor product surface definition can be expressed in a similar matrix form based on the curve matrix form.

Remark 13.1 *In general, the matrix form of a tensor product surface is :*

$$\sigma(u,v)=[\mathcal{B}^n(u)][\mathcal{P}]\,{}^t[\mathcal{B}^m(v)]\,, \qquad (13.4)$$

where the two $[\mathcal{B}]$'s may be different. Such a case is encountered, for instance, when the network of control points includes $n+1$ points in one direction (the u-direction) and $m+1$ points in the other, n and m referring also to the degree of the representation.

Depending on what the B_i^n's are, this kind of method results in various surface definitions. Indeed, both the value of n and the nature of the coefficients in the B_i^n's lead to different definitions. We return, as for the curve case, to interpolation methods where the surface passes through the given (control) points (Lagrange type methods) or passes through the given points while, at the same time, it matches some derivatives at these points (Hermite methods) or again to extrapolation methods such as Bézier, B-spline and many others.

13.2.2 Interpolation-based patches

The "simplest" tensor product that can be used corresponds to the definition of an interpolation method at the curve level. Then, we encounter the two methods previously seen, the Lagrange or the Hermite type interpolation.

13.2.3 Lagrange interpolation

In this case, given an $(n+1)\times(m+1)$ array[2] of data points, $P_{i,j}$, the problem is one of finding the coefficients of the matrices $[\mathcal{B}]$ such that the surface :

$$\sigma(u,v)=[\mathcal{B}^n(u)][\mathcal{P}]\,{}^t[\mathcal{B}^m(v)]$$

passes through the $P_{i,j}$'s. In other words, if $u=u_i$ and $v=v_j$ are the parameters of point $P_{i,j}$, we want to have :

$$P_{i,j}=[\mathcal{B}^n(u_i)][\mathcal{P}]\,{}^t[\mathcal{B}^m(v_j)] \quad \text{for all} \quad i \quad \text{and} \quad j\,.$$

Using as $[\mathcal{B}]$ the Lagrange interpolate of Chapter 12 gives the solution. For example, for $n=m=2$, given 9 control points, the above equation with $[\mathcal{B}^n(u)]=[\mathcal{U}][\mathcal{M}]$ in which :

$$[\mathcal{U}]=[u^2,u,1] \quad \text{and}$$

$$[\mathcal{M}]=\begin{pmatrix}2&-4&2\\-3&4&-1\\1&0&0\end{pmatrix},$$

and similar expressions for $[\mathcal{B}^m(v)]$ defines a Lagrange tensor product resulting in a surface passing through the 9 data points.

[2] where n and m are not necessarily equal.

Hermite interpolation. As for the curve case, a Hermite interpolation involves derivatives of the control points along with these points. The tensor product used to define the surface leads to the same discussion as above. In particular, an equation like Equations (12.3) or (12.4) characterizes the surface through a coefficient matrix based on the basic curves of the tensor product.

For instance, the popular bicubic Hermite patches is one example of such a method. It is defined as :

$$\sigma(u,v) = [\mathcal{B}^3(u)]\,[\mathcal{P}]\,{}^t[\mathcal{B}^3(v)]\,,$$

where $[\mathcal{P}]$ the matrix of the control points contains both the control points and their derivatives. It is written as :

$$[\mathcal{P}] = \begin{bmatrix} P_{0,0} & P_{v,0,0} & P_{v,0,1} & P_{0,1} \\ P_{u,0,0} & P_{uv,0,0} & P_{uv,0,1} & P_{u,0,1} \\ P_{u,1,0} & P_{uv,1,0} & P_{uv,1,1} & P_{u,1,1} \\ P_{1,0} & P_{v,1,0} & P_{v,1,1} & P_{1,1} \end{bmatrix},$$

where, for instance,

$$P_{u,i,j} = \frac{\partial P_{i,j}}{\partial u} \quad \text{and} \quad P_{uv,i,j} = \frac{\partial^2 P_{i,j}}{\partial u \partial v}\,.$$

In this control matrix, the data relative to one control point are grouped into a 2×2 sub-matrix. Now, the coefficient matrix $[\mathcal{B}^3(u)]$ ($[\mathcal{B}^3(v)]$ following the same matrix form) is :

$$[\mathcal{B}^3(u)] = [u^3, u^2, u, 1] \begin{bmatrix} 2 & 1 & 1 & -2 \\ -3 & -2 & -1 & 3 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$$

Remark 13.2 *The above coefficient matrix is the cubic Hermite matrix of Chapter 12 in the case where the control point row is defined as* ${}^t[P_0, \dot P_0, \dot P_1, P_1]$ *instead of* ${}^t[P_0, P_1, \dot P_0, \dot P_1]$ *as we did above. The present notation is motivated by the notations of the* $[\mathcal{P}]$ *matrix for the surface case so as to retrieve the tensor product of the curve definition for the surface case. Also the interpretation of the control points and derivatives as arranged in* $[\mathcal{P}]$ *is then much more intuitive.*

13.2.4 Transfinite interpolation (Coons patches)

The so-called Coons patches are based on transfinite interpolation. Introduced in Chapter 4 as a mesh generation method able to carry out some particular geometries, the transfinite interpolation is now seen as a method for surface definition (construction).

Referring to Chapter 4, we recall the formula of Equation (4.5) :

$$\begin{aligned} F(\xi,\eta) \;=\;& (1-\eta)\,\phi_1(\xi) + \xi\,\phi_2(\eta) + \eta\,\phi_3(\xi) + (1-\xi)\,\phi_4(\eta) \\ & -((1-\xi)(1-\eta)\,a_1 + \xi(1-\eta)\,a_2 + \xi\eta\,a_3 + (1-\xi)\,\eta a_4)\,. \end{aligned}$$

Now, to conform to the current notations, this formula is written as :

$$\begin{aligned} \sigma(u,v) \;=\;& (1-v)\,\phi(u,0) + u\,\phi(1,v) + v\,\phi(u,1) + (1-u)\,\phi(0,v) \\ & -((1-u)(1-v)\,P_{0,0} + u(1-v)\,P_{1,0} + uv\,P_{1,1} + (1-u)\,vP_{0,1})\,. \end{aligned}$$

which can be expressed as :

$$\begin{aligned} \sigma(u,v) \;=\;& [1-u, u]\begin{pmatrix} \phi(0,v) \\ \phi(1,v) \end{pmatrix} + (\phi(u,0), \phi(u,1))\begin{bmatrix} 1-v \\ v \end{bmatrix} \\ & - [1-u,u]\begin{pmatrix} \phi(0,0) & \phi(0,1) \\ \phi(1,0) & \phi(1,1) \end{pmatrix}\begin{bmatrix} 1-v \\ v \end{bmatrix}. \end{aligned} \tag{13.5}$$

The first part of this expression corresponds to a ruled surface in terms of u, the second denotes a ruled surface in terms of v, while the third part is the correction term resulting in the desired properties (see below).

Thus, we have a surface definition which is rather different from the previous ones. The patches that can be dealt with are defined via their boundaries and these boundaries can be seen as a series of four logical sides[3]. Actually, the input data consists of the boundary of the patch, *i.e.*, ϕ. In other words, from a discrete point of view, the control points are located on the four curves defining the patch boundaries. Functions $1-u$ and u are the so-called *blended functions.*

It is easy to check, as in Chapter 4, that the corner identities are satisfied and that the surface interpolates to the four boundary curves. For instance, we have $\sigma(u,0) = \phi(u,0), \sigma(u,1) = \phi(u,1)$ and $\sigma(0,v) = \phi(0,v)$ together with $\sigma(1,v) = \phi(1,v)$.

Changing the blending functions leads to generalizing the Coons patch. If $f_1(u)$ and $f_2(u)$ along with $g_1(v)$ and $g_2(v)$ are two pairs of blending functions, then :

$$\begin{aligned} \sigma(u,v) \;=\;& [f_1(u), f_2(u)]\begin{pmatrix} \phi(0,v) \\ \phi(1,v) \end{pmatrix} + (\phi(u,0), \phi(u,1))\begin{bmatrix} g_1(v) \\ g_2(v) \end{bmatrix} \\ & - [f_1(u), f_2(u)]\begin{pmatrix} \phi(0,0) & \phi(0,1) \\ \phi(1,0) & \phi(1,1) \end{pmatrix}\begin{bmatrix} g_1(v) \\ g_2(v) \end{bmatrix}, \end{aligned} \tag{13.6}$$

is the form of a generalized Coons patch. Note that the blending functions must enjoy certain properties to result in a consistent definition. In practice, the f_i's as well as the g_i's sum to one and continuity must be ensured at the corners, $f_1(0) = g_1(0) = 0$ together with $f_1(1) = g_1(1) = 1$.

An example of blending functions resulting in a powerful surface definition consists of cubic Hermite polynomials. In this case, the surface is defined by :

$$\sigma(u,v) = [H_0^3(u), H_1^3(u), H_2^3(u), H_3^3(u)]\begin{pmatrix} \phi(0,v) \\ \phi_u(0,v) \\ \phi_u(1,v) \\ \phi(1,v) \end{pmatrix}$$

[3] Following the transfinite interpolation for a triangle of Chapter 4, it is possible to develop triangular patches as well.

$$+\quad (\phi(u,0),\phi_v(u,0),\phi_v(u,1),\phi(u,1))\begin{bmatrix} H_0^3(v)\\ H_1^3(v)\\ H_2^3(v)\\ H_3^3(v)\end{bmatrix} \qquad (13.7)$$

$$-\quad [\mathcal{H}^3(u)]\begin{pmatrix} \phi(0,0) & \phi_v(0,0) & \phi_v(0,1) & \phi(0,1)\\ \phi_u(0,0) & \phi_{uv}(0,0) & \phi_{uv}(1,0) & \phi_u(1,0)\\ \phi_u(1,0) & \phi_{uv}(1,0) & \phi_{uv}(1,1) & \phi_u(1,1)\\ \phi(1,0) & \phi_v(1,0) & \phi_v(1,1) & \phi(1,1)\end{pmatrix} {}^t[\mathcal{H}^3(v)],$$

where the H_i^3 are the cubic Hermite polynomials of Section 12.2.5 and $\mathcal{H}^3$ is the corresponding line.

In fact, depending on the input data, using the previous patch definition requires a certain technique. When only the curves ϕ are supplied, quantities involving the necessary derivatives must be approached. When the ϕ are given along with the directional derivatives, the uv-derivatives must be evaluated. Various solutions may then be used leading to various controls of the thus-defined surface.

13.3 Tensor product and control polyhedron

In this section, we discuss the patches that are widely used in CAD software packages. They make it possible to define surfaces with an arbitrary shape (geometry).

13.3.1 Control polyhedron

Let $P_{i,j}$ be a quadrilateral lattice of points. By analogy with the case of curves based on a control polygon, we can introduce, for a given surface Σ, the following definition :

$$\sigma(u,v)=\sum_{i=0}^{m}\sum_{j=0}^{n}\Phi_{i,j}(u,v)P_{i,j}\,. \qquad (13.8)$$

The choice of the basis functions $\Phi_{i,j}$ is left to the user. However, some characteristics are usually imposed in order to relate the surface to its control polyhedron (the $P_{i,j}$'s).

Tensor product. The first assumption is that the surface can be written as a tensor product. In order to ensure this property, the contribution of u and that of v must be separated in the definition of $\Phi_{i,j}(u,v)$. So, we have :

$$\Phi_{i,j}(u,v)=\phi_i(u)\,\psi_j(v)\,.$$

The main interest of such a definition for $\Phi_{i,j}$ is that Equation (13.8) becomes :

$$\sigma(u,v)=\sum_{i=0}^{m}\phi_i(u)\underbrace{\left(\sum_{j=0}^{n}\psi_j(v)P_{i,j}\right)}_{S_i(v)=\gamma_1(v)}=\sum_{i=0}^{m}\phi_i(u)\,S_i(v)\,. \qquad (13.9)$$

Thus, a current point on the surface is defined by the tensor product of two curves based on a definition by a control polygon. The points $S_i(v)$ can be obtained as current points of the curve $\gamma_1(v)$ related to the control polygon $(P_{i,j}(v))_{j\in[0,m]}$ and, these points being calculated, any point $\sigma(u,v)$ is calculated as the current point of a new curve related to the control polygon $(S_i(v))_{i\in[0,n]}$.

Characteristics of $\phi_i(u)$ and $\psi_j(v)$ inherited from the curve definition. To be consistent with the definition of curves, the surfaces based on a control polyhedron generally use the same basis functions for $\phi_i(u)$ and for $\psi_j(v)$ as the curves. Hence, if we use Bernstein polynomials, we define Bézier patches and if we take Splines functions, we get B-Splines type patches. By adding an homogeneous coordinate to each control point of the polyhedron, we find rational patches, *i.e.*, rational Bézier or B-Splines patches, commonly called *NURBS* when the associated sequence of nodes is non-uniform. Therefore, it should be borne in mind that the features of patches inherit the characteristics of the underlying curves.

Cauchy identity. Let us look at the value of $\mathcal{S}=\sum_{i=0}^{m}\sum_{j=0}^{n}\Phi_{i,j}(u,v)$. We have :

$$\mathcal{S}=\sum_{i=0}^{m}\sum_{j=0}^{n}\phi_i(u)\psi_j(v)=\sum_{i=0}^{m}\phi_i(u)\underbrace{\sum_{j=0}^{n}\psi_j(v)}_{\equiv 1}=\underbrace{\sum_{i=0}^{m}\phi_i(u)}_{\equiv 1}\equiv 1\,.$$

Hence, the Cauchy identity is also true in this type of surface definition. Thus, we have :

$$\sigma(u,v)=\frac{\sum_{i=0}^{m}\sum_{j=0}^{n}\Phi_{i,j}(u,v)P_{i,j}}{\sum_{i=0}^{m}\sum_{j=0}^{n}\Phi_{i,j}(u,v)}\,.$$

The point of parameters (u,v) is the barycenter of the system of $(n+1)\times(m+1)$ points $P_{i,j}$ associated with the weights $\Phi_{i,j}(u,v)$.

Remark 13.3 *As for the curves, the Cauchy identity ensures that any linear transformation of the characteristic lattice produces the same transformation onto the surface.*

Positive functions. As the functions $\phi_i(u)$ and $\psi_j(v)$ are positive functions, the functions $\Phi_{i,j}(u,v)$ are positive in the domain of definition of the (u,v)'s.

If we assume the Cauchy identity to be verified, the current point $\sigma(u,v)$ is the barycenter of the points $P_{i,j}$ weighted with positive values. Thus, any convex volume enclosing the set of points $P_{i,j}$ encloses the whole patch.

Remark 13.4 *An application of this feature consists in taking as the bounding box of the patch the box defined via the minima and the maxima of the coordinates of the points of the control polyhedron.*

Relations at the endpoints. Let $[u_{min}, u_{max}] \times [v_{min}, v_{max}]$ be the interval of variation of $u \times v$. If we assume that ϕ_i and ψ_j verify the relation of Section 12.4.2 related to the curves, it is easy to show that :

$$\Phi_{0,0}(u_{min}, v_{min}) = 1 \; ; \; \Phi_{n,0}(u_{max}, v_{min}) = 1 \, ,$$

$$\Phi_{n,m}(u_{max}, v_{max}) = 1 \; ; \; \Phi_{0,m}(u_{min}, v_{max}) = 1 \, .$$

Hence, the four corners of the polyhedron define the four points of the patch.

Boundary curves. Let us place $v = v_{min}$ in Equation (13.9), thus the equation of a curve governed by the parameter u is :

$$\sigma(u, v_{min}) = \sum_{i=0}^{m} \phi_i(u) S_i(v_{min}) = \sum_{i=0}^{m} \phi_i(u) P_{i,0} \, .$$

Hence, the boundaries of the control polyhedron define the control polygons of the curves bounding the patch.

Tangent planes at the corners. As the boundary curves are defined by the boundary of the control polyhedron, we can obtain two particular tangents at any corner by taking the tangents to the endpoints of the boundary curves. These tangents are supported by the segments of the control polyhedron sharing the given corner.

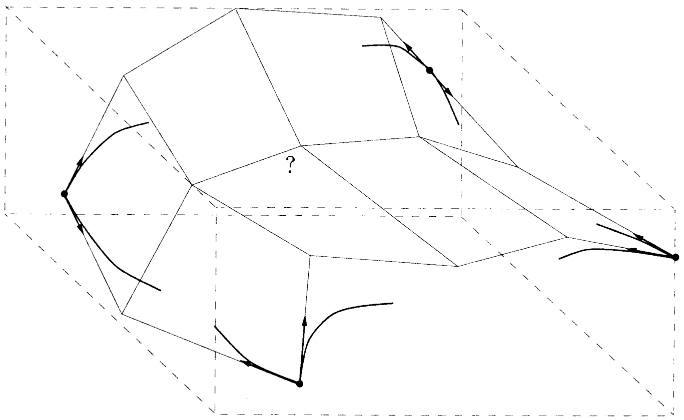

Figure 13.2: *One patch, its control polyhedron and the tangent planes at the corners.*

Figure 13.2 shows a control polyhedron. The corners of this polyhedron are particular points of this patch. The tangent planes at these corners are defined by the segments incident to these points. Within the parametric space, we can only affirm that the surface "is similar to" its control polyhedron and that the patch is enclosed in any box enclosing this control polyhedron.

13.3.2 Bézier quads

Bézier quad patches are constructed using Bernstein polynomials as ϕ_i and ψ_j. As a consequence, we have :

$$\sigma(u, v) = \sum_{i=0}^{n} \sum_{j=0}^{m} B_{i,n}(u) \, B_{j,m}(v) \, P_{i,j} \, , \qquad (13.10)$$

with $B_{i,n}(u)$ (resp. $B_{j,m}(v)$) the classical Bernstein polynomial, *i.e.*, for example,

$$B_{i,n}(u) = C_n^i u^i (1-u)^{n-i} \, .$$

Notice that m can be different to n. The degree of the patch is not necessarily the same in the two directions u and v.

On the other hand , the data of a lattice of control points $P_{i,j}$ whose structure is a quadrilateral grid ($(n+1) \times (m+1)$ control points) makes it possible to define a surface via Equation 13.10.

Properties already seen in Chapter 12 about Bézier curves (and Bernstein polynomials) apply here in Bézier quadrilateral patches.

As for the curves, various recursions allow us to compute different quantities of interest, for instance, for point evaluations, for derivative computations, for degree elevations, and many others. In this way, it will be possible to find the conditions that ensure some degree of continuity when composite surfaces are discussed (see below).

Also the De Casteljau algorithm extends to the surface case when[4] $n = m$. The curve algorithm :

$$D_i^r(t) = (1-t) \, D_i^{r-1}(t) + t \, D_{i+1}^{r-1}(t) \, ,$$

is replaced by :

$$D_{i,j}^{r,r}(u,v) = [1-u, u] \begin{pmatrix} D_{i,j}^{r-1,r-1}(u,v) & D_{i,j+1}^{r-1,r-1}(u,v) \\ D_{i+1,j}^{r-1,r-1}(u,v) & D_{i+1,j+1}^{r-1,r-1}(u,v) \end{pmatrix} \begin{bmatrix} 1-v \\ v \end{bmatrix}$$

for $r = 1, ..., n$ and i, j in $[0, n-r]$. This algorithm is initialized by $D_{i,j}^{0,0} = P_{i,j}$ which is, for consistency, noted by $D_{i,j}^{0,0}(u,v)$ while, obviously u and v do not appear in the D at the initialization step. Then, the surface is equivalently written as :

$$\sigma(u,v) = D_{0,0}^{n,n}(u,v) \, .$$

As previously indicated, a Bézier patch can be formulated in terms of a matrix form. Indeed, we return to Relation 13.3, *i.e.*, :

$$\sigma(u,v) = [\mathcal{B}^n(u)][\mathcal{P}]\,{}^t[\mathcal{B}^m(v)] \quad \text{or} \quad [\mathcal{U}][\mathcal{M}][\mathcal{P}]\,{}^t[\mathcal{N}]\,{}^t[\mathcal{V}]$$

where

$$[\mathcal{U}] = [u^n, u^{n-1}, ..., u^2, u, 1]$$

[4] If $n \neq m$, the De Casteljau algorithm is more subtle.

$$[\mathcal{M}] = [M_{i,j}] \quad \text{with} \quad M_{i,j} = (-1)^{n-i-j}\, \mathcal{C}_n^i\, \mathcal{C}_{n-i}^j$$

and similar expressions for both $[\mathcal{V}]$ and $[\mathcal{N}]$.

Then, for $n = 3$ we return to the matrix of Chapter 12, *i.e.*, :

$$[\mathcal{M}] = \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}.$$

Degree elevation. As for a Bézier curve (see Chapter 12), degree elevation of a Bézier patch is useful for various purposes. In practice, curve degree elevation can be used while the degree is elevated in the u-direction and, this task being completed, it is elevated in the v-direction. Then following the technique for a Bézier curve, we first fix j and m and we seek to obtain a $(n+1, m)$ patch (starting from a (n, m) patch). The surface :

$$\sigma(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} B_{i,n}(u)\, B_{j,m}(v)\, P_{i,j}$$

is seen as :

$$\sigma(u,v) = \sum_{j=0}^{m} \left(\sum_{i=0}^{n} B_{i,n}(u)\, P_{i,j} \right) B_{j,m}(v)$$

and, with

$$Q^*_{i,j} = \frac{i\, P_{i-1,j} + (n+1-i)\, P_{i,j}}{n+1},$$

we obtain :

$$\sigma(u,v) = \sum_{j=0}^{m} \sum_{i=0}^{n+1} B_{i,n+1}(u) B_{j,m}(v)\, Q^*_{i,j},$$

applying the same in terms of m, we obtain the desired result, *i.e.*, :

$$\sigma(u,v) = \sum_{i=0}^{n+1} \sum_{j=0}^{m+1} B_{i,n+1}(u) B_{j,m+1}(v)\, Q_{i,j},$$

where, now,

$$Q_{i,j} = \frac{j\, Q^*_{i,j-1} + (m+1-j)\, Q^*_{i,j}}{m+1},$$

thus, in terms of the $P_{i,j}$'s, we have :

$$Q_{i,j} = \frac{1}{n+1} \frac{1}{m+1} [i, n+1-i] \begin{pmatrix} P_{i-1,j-1} & P_{i,j-1} \\ P_{i-1,j} & P_{i,j} \end{pmatrix} \begin{bmatrix} j \\ m+1-j \end{bmatrix}.$$

13.3.3 B-splines patches

These patches take a form similar to Relation 13.10 in which the polynomials are replaced by the B-spline functions introduced in Chapter 12. Then, the surface is defined by :

$$\sigma(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} N_{i,n}(u)\, N_{j,m}(v)\, P_{i,j}. \qquad (13.11)$$

Note that rational B-splines patches also exist.

13.4 Triangular patches

As mentioned in the introduction, there also exist triangular patches, and we now turn our attention to these.

13.4.1 Tri-parametric forms

Triangular patches are most commonly defined using an expression with three parameters, u, v and w : $\sigma(u,v,w)$. The values (u,v,w) are the barycentric coordinates. Hence, we have the relation :

$$u + v + w = 1 \qquad (u,v,w) \in \mathbb{R}^3_+.$$

Thus, the whole triangle is described by the point P corresponding to the barycentric coordinates :

$$P = u\, P_0 + v\, P_1 + w\, P_2.$$

From this point of view, the function σ can be seen as a deforming function which, applied to a predefined planar triangle (P_0, P_1, P_2), gives a curved triangle.

13.4.2 Bézier triangles

Bézier patches play an important role in surface definition. Two types of Bézier patches are commonly used. As may be imagined, the first type, Bézier triangles, were introduced before the second, namely Bézier quadrilateral patches. For the sake of simplicity, we discussed the quad case before the triangle case since a Bézier quad is a tensor product, thus allowing a "simple" discussion. Now, we consider Bézier triangles.

Given a set of control points that form, in terms of topology, a triangular network, we can define a triangular surface using the relationship :

$$\sigma(u,v,w) = \sum_{i+j+k=n} B^n_{i,j,k}(u,v,w)\, P_{i,j,k}, \qquad (13.12)$$

where

$$B^n_{i,j,k}(u,v,w) = \frac{n!}{i!j!k!} u^i v^j w^k,$$

and u, v and w are the barycentric coordinates. To make the meaning of the above indices clear, we give the following synthetic scheme (where $n = 3$) which corresponds to the arrangement of the control points :

$$\begin{array}{cccc} & & (0,3,0) & \\ & (0,2,1) & (1,2,0) & \\ (0,1,2) & (1,1,1) & (2,1,0) & \\ (0,0,3) & (1,0,2) & (2,0,1) & (3,0,0) \end{array}$$

Also, following the same index arrangement, the Bernstein polynomials, for $n = 3$ are as follows :

$$\begin{array}{cccc} & & v^3 & \\ & 3\,v^2\,w & 3\,u\,v^2 & \\ 3\,v\,w^2 & 6\,u\,v\,w & 3\,u^2\,v & \\ w^3 & 3\,u\,w^2 & 3\,u^2\,w & u^3 \end{array}$$

The Bernstein polynomials of a Bézier triangle are related by the recursion :

$$B^n_{i,j,k}(u,v,w) = uB^{n-1}_{i-1,j,k}(u,v,w) + vB^{n-1}_{i,j-1,k}(u,v,w) + wB^{n-1}_{i,j,k-1}(u,v,w)\,,$$

and, a De Casteljau algorithm can be found, based on this recursion. First, we define :

$$D^0_{i,j,k} = P_{i,j,k}\,,$$

then, for $r \in [1,n]$ with $i+j+k = n-r$, the recursion is :

$$D^r_{i,j,k}(u,v,w) = uD^{r-1}_{i+1,j,k}(u,v,w) + vD^{r-1}_{i,j+1,k}(u,v,w) + wD^{r-1}_{i,j,k+1}(u,v,w)\,,$$

and

$$\sigma(u,v,w) = D^n_{0,0,0}(u,v,w)\,.$$

The surface passes through the three corners, its boundary consists of the three Bézier curves corresponding to $u = 0$, $v = 0$ and $w = 0$. Another interesting property is the following. The tangent planes at the corners are based on the three triangles with such a point as vertex. For example, the tangent plane at corner $P_{n,0,0}$ is the plane of the triangle $P_{n,0,0}, P_{n-1,1,0}, P_{n-1,0,1}$. This is due to the fact that two derivatives are automatically known at the corners. For instance, at corner $P_{n,0,0}$, these two derivatives are the vectors $\overrightarrow{P_{n,0,0}P_{n-1,1,0}}$ and $\overrightarrow{P_{n,0,0}P_{n-1,0,1}}$.

In terms of derivatives, it is easy to find the derivatives of the above Bernstein polynomials. We have,

$$\frac{\partial}{\partial u}B^n_{i,j,k}(u,v,w) = n\,B^{n-1}_{i,j,k}(u,v,w)\,,$$

and similar expressions for the other partials. Now, for a Bézier triangle, the derivatives we are interested in are not the partials but some *directional derivatives*.

Thus, given two points A and B (with barycentric coordinates u_A, v_A, w_A and u_B, v_B, w_B), we consider vector $AB = \overrightarrow{AB}$ whose components $u_{AB} = u_B - u_A$, etc. (which sum to zero). This vector enables us to write the directional derivative of the surface at a given point. We have :

$$\frac{d}{d_{AB}}\sigma(u,v,w) = u_{AB}\frac{\partial}{\partial u}\sigma(u,v,w) + v_{AB}\frac{\partial}{\partial v}\sigma(u,v,w) + w_{AB}\frac{\partial}{\partial w}\sigma(u,v,w)\,,$$

or, by means of the Bernstein polynomials :

$$n \sum_{i+j+k=n-1} B^{n-1}_{i,j,k}(u,v,w)\,(u_{AB}\,P_{i+1,j,k} + v_{AB}\,P_{i,j+1,k} + w_{AB}\,P_{i,j,k+1})\,. \tag{13.13}$$

We can then express the tangent plane anywhere (and not only at the corners) in a triangle in terms of the above definition (a derivative) as well as by means of the De Casteljau algorithm. To this end, Equation (13.13) is split into three parts :

$$\begin{aligned}\frac{d}{d_{AB}}\sigma(u,v,w) = n\Bigg(& u_{AB}\sum_{i+j+k=n-1} B^{n-1}_{i,j,k}(u,v,w)\,P_{i+1,j,k} \\ &+ v_{AB}\sum_{i+j+k=n-1} B^{n-1}_{i,j,k}(u,v,w)\,P_{i,j+1,k} \\ &+ w_{AB}\sum_{i+j+k=n-1} B^{n-1}_{i,j,k}(u,v,w)\,P_{i,j,k+1}\Bigg)\,,\end{aligned} \tag{13.14}$$

then, we have :

$$\frac{d}{d_{AB}}\sigma(u,v,w) = n\,(u_{AB}\,D^{n-1}_{1,0,0} + v_{AB}\,D^{n-1}_{0,1,0} + w_{AB}\,D^{n-1}_{0,0,1})\,,$$

and, as a conclusion, the tangent plane is the plane passing through the points $D^{n-1}_{1,0,0}$, $D^{n-1}_{0,1,0}$ and $D^{n-1}_{0,0,1}$ resulting from the De Casteljau algorithm.

With this material, it will be possible to find the conditions relating to the control points in order to obtain some continuity from patch to patch in the case of a composite surface.

Degree elevation. As Bézier triangles are not tensor product patches, degree elevation of such a patch does not take the same form as for a quad patch. Thus, we explicitly discuss how to elevate the degree of a Bézier triangle. We want to write the surface :

$$\sigma(u,v,w) = \sum_{i+j+k=n} B^n_{i,j,k}(u,v,w)\,P_{i,j,k}\,,$$

in the following form :

$$\sigma(u,v,w) = \sum_{i+j+k=n+1} B^{n+1}_{i,j,k}(u,v,w)\,Q_{i,j,k}\,,$$

and the problem is to find the $Q_{i,j,k}$'s. The solution is then :

$$Q_{i,j,k} = \frac{1}{n+1}\left(i\,P_{i-1,j,k} + j\,P_{i,j-1,k} + k\,P_{i,j,k-1}\right) .$$

Proof. Let us consider the Bernstein polynomial $B^{n+1}_{i+1,j,k}(u,v,w)$. Using the definition of such a polynomial, we have :

$$B^{n+1}_{i+1,j,k}(u,v,w) = \frac{(n+1)!}{(i+1)!j!k!}u^{i+1}v^j w^k \quad \text{then}$$

$$B^{n+1}_{i+1,j,k}(u,v,w) = u\frac{n+1}{i+1}\frac{n!}{i!j!k!}u^i v^j w^k \;=\; u\frac{n+1}{i+1}B^n_{i,j,k}(u,v,w) .$$

Similarly (omitting the parameters),

$$B^{n+1}_{i,j+1,k} = v\frac{n+1}{j+1}B^n_{i,j,k} \quad \text{and} \quad B^{n+1}_{i,j,k+1} = w\frac{n+1}{k+1}B^n_{i,j,k} .$$

Conversely, we have :

$$u\,B^n_{i,j,k} = \frac{i+1}{n+1}B^{n+1}_{i+1,j,k}, \quad v\,B^n_{i,j,k} = \frac{j+1}{n+1}B^{n+1}_{i,j+1,k}, \quad w\,B^n_{i,j,k} = \frac{k+1}{n+1}B^{n+1}_{i,j,k+1},$$

or, after summation,

$$B^n_{i,j,k} = \frac{1}{n+1}\left((i+1)\,B^{n+1}_{i+1,j,k} + (j+1)\,B^{n+1}_{i,j+1,k} + (k+1)\,B^{n+1}_{i,j,k+1}\right) ,$$

since $u+v+w=1$. Now the initial patch definition is replaced by :

$$\sigma(.) = \sum_{i+j+k=n} \frac{1}{n+1}\left((i+1)\,B^{n+1}_{i+1,j,k} + (j+1)\,B^{n+1}_{i,j+1,k} + (k+1)\,B^{n+1}_{i,j,k+1}\right) P_{i,j,k} ,$$

which, in turn, leads to :

$$\sigma(u,v,w) = \sum_{i+j+k=n+1} \frac{1}{n+1}B^{n+1}_{i,j,k}\left(i\,P_{i-1,j,k} + j\,P_{i,j-1,k} + k\,P_{i,j,k-1}\right) ,$$

thus completing the proof. □

13.5 Other types of patches

Here we examine other types of patches, rational patches, rational Bézier patches and patches based on an arbitrary polyhedron.

13.5.1 Rational patches

The same approach as that used to define rational curves (Section 12.7) can be adopted in the case of surfaces. Thus, the construction of rational models comes down to adding an homogeneous coordinate to each point of the control polyhedron. We can thus easily obtain rational Bézier quadrilaterals or triangles, NURBS-type patches, etc. We will briefly describe the case of rational Bézier patches in order to illustrate such patches.

13.5.2 Rational quad Bézier patches

For a quad patch, we have :

$$\sigma(u,v) = \frac{\sum_{i=0}^{n}\sum_{j=0}^{m}\omega_{i,j}\,B_{i,n}(u)\,B_{j,m}(v)\,P_{i,j}}{\sum_{i=0}^{n}\sum_{j=0}^{m}\omega_{i,j}\,B_{i,n}(u)\,B_{j,m}(v)} .$$

13.5.3 Rational Bézier triangles

For a triangular patch, we find :

$$\sigma(u,v,w) = \frac{\sum_{i+j+k=n}\omega_{i,j,k}\,B^n_{i,j,k}(u,v,w)\,P_{i,j,k}}{\sum_{i+j+k=n}\omega_{i,j,k}\,B^n_{i,j,k}(u,v,w)} .$$

Remark 13.5 *Note that rational patches are not tensor product based patches.*

13.5.4 Patches based on an arbitrary polyhedron

Various types of patches exist which have not been discussed so far. We will limit ourselves here to giving a few examples of such patches.

Patches can be defined which are arbitrary polygons. The patches we have discussed so far are quadrilateral (or at least with four sides) or triangular (Bézier triangles). Some geometries are not well determined if we restrict ourselves to these patterns. Thus patches with arbitrary topology can be introduced and methods for handling these cases must be defined.

A popular patch is the so-called Clough-Tocher patch whose name is familiar to finite element people. Indeed, the Clough-Tocher finite element is a triangle with 12 degrees of freedom (Chapter 20). The node value and the two derivatives at the triangle vertices and a normal derivative at the edge midpoints. This finite element is C^1 and its construction is based on the three sub-triangles that are defined using its centroid. In terms of a surface definition, this triangle is seen as a patch with 12 degrees that allows for a C^1 continuity.

Another patch of interest is the Gregory patch . This patch is a triangle and the data consists of the vertices and the normals at these vertices. Also, a patch, proposed by Walton, [Walton,Meek-1996], is constructed with similar data. In a later section, we will return to patches which allow for a G^1 continuity.

We also frequently find patches having a particular shape. In fact, if we take the case of a four-sided patch, it is not always possible to construct a given surface using only this type of patch. Indeed, the number of patches incident to one point is in principle 4. If for any reason, this configuration is not possible, we have to aim at a different number of incident patches. Hence, three- or five-sided patches have been introduced. To reduce the number of sides of a patch, we can degenerate it (*i.e.*, force two vertices to be coincident). On the other hand, increasing the number of sides is more tedious (notice however that the Coons patch makes it possible to envisage this case quite easily).

Finally, notice that restraining the usable domain of a patch makes it possible, while preserving a limited number of patches, to introduce arbitrary boundaries (other than the "natural" borders of the patch) and holes (which could also be obtained by subdividing the initial patch), Figure 13.3.

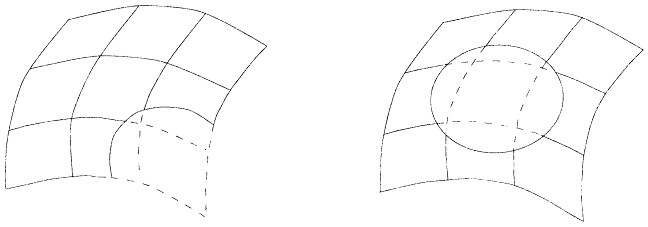

Figure 13.3: *Examples of restricted patches. Left-hand side, the boundary of interest is not the patch boundary. Right-hand side, there is a hole on the surface.*

13.6 Composite surfaces

As for composite curves, composite surfaces is a definition method that allows for a great flexibility in the case of arbitrary surfaces. Figure 13.4 depicts an example of a surface whose definition requires a certain number of patches[5]. Actually, the given surface is split into a series of patches which conform to one of the above patch nature.

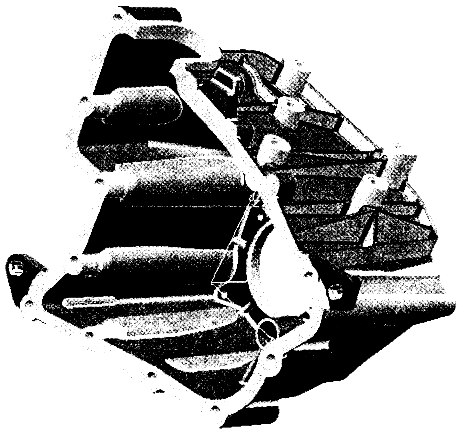

Figure 13.4: *Example of a complex surface using a set of composite patches (view data courtesy by Dassault Systèmes, modeler CATIA).*

Given two (quad) Bézier patches, which are members of a composite surface, the aim is to define the control points in such a way as to insure some degree of

[5]in fact, this model certainly includes also some restricted patches and not only some natural boundaries (Chapter 15).

continuity from one patch to the other, thus resulting in a global continuity when the entire surface is considered.

The key is then to have a continuity at the patch interfaces, *i.e.*, at the curve level. This continuity concerns the adequate derivatives of the curves. First, we need to have the values of the derivatives of a given Bézier patch. By introducing the notation :

$$\Delta_u P_{i,j} = P_{i+1,j} - P_{i,j} \quad \text{and} \quad \Delta_v P_{i,j} = P_{i,j+1} - P_{i,j} ,$$

then, the formula :

$$\Delta^{rs} P_{i,j} = \Delta_u^r(\Delta_v^s P_{i,j}) = \Delta_v^s(\Delta_u^r P_{i,j}) ,$$

allows for the calculation of the derivatives of the surface $\sigma(u,v)$. We have :

$$\frac{\partial^r \partial^s}{\partial u^r \partial v^s}\sigma(u,v) = \frac{n!}{(n-r)!}\frac{m!}{(m-s)!}\sum_{i=0}^{n-r}\sum_{j=0}^{m-s} B_{i,n-r}(u)\, B_{j,m-s}(v)\, \Delta^{rs} P_{i,j} .$$

Now, let us consider two such patches, $\sigma_1(u_1, v_1)$ and $\sigma_2(u_2, v_2)$, with the same degrees[6] n and m and whose interface consists of :

$$\sigma_1(1, v_1) \quad \text{and} \quad \sigma_2(0, v_2) ,$$

the curves corresponding to $u_1 = 1$ and $u_2 = 0$ respectively.

A C^0 continuity from σ_1 and σ_2 leads to having :

$$\sum_{j=0}^{m} B_{j,m}(v_1)\, P_{n,j} = \sum_{j=0}^{m} B_{j,m}(v_2)\, Q_{0,j} \quad \forall v_1 = v_2 \in [0,1]$$

where the $P_{i,j}$ are the control points of σ_1 while the control points of σ_2 are the $Q_{i,j}$'s. Then, in terms of control points, this continuity is achieved if :

$$P_{n,j} = Q_{0,j} \quad \forall j .$$

A C^1 continuity is achieved if the first derivatives match at the interface. Using the above formula about the derivatives with respectively $r = 1, s = 0$ and $r = 0, s = 1$, we have successively :

$$\frac{\partial}{\partial u}\sigma_1(1, v_1) = n \sum_{j=0}^{m} B_{j,m}(v_1)\, (P_{n,j} - P_{n-1,j})$$

$$\frac{\partial}{\partial u}\sigma_2(0, v_2) = n \sum_{j=0}^{m} B_{j,m}(v_2)\, (Q_{1,j} - Q_{0,j})$$

and

$$\frac{\partial}{\partial v}\sigma_1(1, v_1) = m \sum_{j=0}^{m-1} B_{j,m-1}(v_1)\, (P_{n,j+1} - P_{n,j})$$

[6]Actually, the same degree is only necessary at the interface (*i.e.*, for m).

$$\frac{\partial}{\partial v}\sigma_2(0,v_2) = m\sum_{j=0}^{m-1} B_{j,m-1}(v_2)\,(Q_{0,j+1}-Q_{0,j})$$

Then, we need to have : $\frac{\partial}{\partial u}\sigma_1(1,v_1) = \frac{\partial}{\partial u}\sigma_2(0,v_2)$ together with $\frac{\partial}{\partial v}\sigma_1(1,v_1) = \frac{\partial}{\partial v}\sigma_2(0,v_2)$ for $v_1 = v_2 \in [0,1]$. Since a C^0 continuity holds, the v-derivatives match while a condition insuring that the u-derivatives match is $P_{n,j} - P_{n-1,j} = Q_{1,j} - Q_{0,j}$ or, since $Q_{0,j} = P_{n,j}$,

$$P_{n-1,j} + Q_{1,j} = 2\,P_{n,j}\,.$$

In other words, $P_{n,j}$, $P_{n-1,j}$ and $Q_{1,j}$ are aligned[7] and, moreover $P_{n,j}$ must be the midpoint of segment $P_{n-1,j}, Q_{1,j}$.

Similarly a high order continuity can be obtained since additional conditions about the control points are enforced. For instance, a C^2 continuity also leads to having :

$$Q_{2,j} - P_{n-2,j} = 2\,(Q_{1,j} - P_{n-1,j})\,.$$

Remark 13.6 *When the parameters are not in* $[0,1]$, *a variable change must be made to return to the above discussion.*

Since a C-type continuity is very demanding, thus impeding flexibility when defining the control points, the use of a G-type continuity must be discussed. For instance, the G^1 continuity at an interface between two patches means that the two tangent planes are continuous along the interface curve.

The tangent plane for surface $\sigma_1(u_1,v_1)$, along the curve $u_1 = 1$ (we pursue the above example), is a combination like :

$$\alpha_1\frac{\partial}{\partial u}\sigma_1(1,v_1) + \beta_1\frac{\partial}{\partial v}\sigma_1(1,v_1)\,,$$

that for $\sigma_2(u_2,v_2)$, along the curve $u_2 = 0$, is :

$$\alpha_2\frac{\partial}{\partial u}\sigma_2(0,v_2) + \beta_2\frac{\partial}{\partial v}\sigma_2(0,v_2)\,,$$

where the α's and the β's are some non null coefficients (indeed, some functions of the v's). The relation which is needed is then :

$$\alpha_1\frac{\partial}{\partial u}\sigma_1(1,v_1) + \beta_1\frac{\partial}{\partial v}\sigma_1(1,v_1) - \alpha_2\frac{\partial}{\partial u}\sigma_2(0,v_2) - \beta_2\frac{\partial}{\partial v}\sigma_2(0,v_2) = 0\,,$$

thus, due to the C^0 continuity, this relation reduces to :

$$\alpha\frac{\partial}{\partial u}\sigma_1(1,v_1) + \beta\frac{\partial}{\partial v}\sigma_1(1,v_1) + \gamma\frac{\partial}{\partial u}\sigma_2(0,v_2) = 0\,,$$

where $\alpha = \alpha_1, \beta = \beta_1 - \beta_2$ and $\gamma = -\alpha_2$.

[7] Thus, only a G^1 continuity is achieved.

The "simplest" choice is to consider constant values for these functions and to take $\beta = 0$, meaning that $\beta_1 = \beta_2$. Then, the condition for a G^1 continuity reduces to :

$$\alpha\frac{\partial}{\partial u}\sigma_1(1,v_1) + \gamma\frac{\partial}{\partial u}\sigma_2(0,v_2) = 0$$

and, in terms of the control points,

$$\alpha\,n\sum_{j=0}^{m} B_{j,m}(v_1)\,(P_{n,j}-P_{n-1,j}) + \gamma\,n\sum_{j=0}^{m} B_{j,m}(v_2)\,(Q_{1,j}-Q_{0,j})\,,$$

thus, since $v_1 = v_2$, we return to the above condition about the alignment of $P_{n,j}$, $P_{n-1,j}$ and $Q_{1,j}$.

Now, we consider $\beta \neq 0$ and more precisely a function like $\beta = (1-v)\beta + v$. Then, we must have :

$$\alpha\frac{\partial}{\partial u}\sigma_1(1,v_1) + ((1-v)\beta + v)\frac{\partial}{\partial v}\sigma_1(1,v_1) + \gamma\frac{\partial}{\partial u}\sigma_2(0,v_2) = 0\,.$$

In terms of control points, this leads to having :

$$\alpha\,n\sum_{j=0}^{m} B_{j,m}(v_1)\,(P_{n,j}-P_{n-1,j}) +$$

$$((1-v)\beta+v)\,m\sum_{j=0}^{m-1} B_{j,m-1}(v_1)\,(P_{n,j+1}-P_{n,j}) + \gamma\,n\sum_{j=0}^{m} B_{j,m}(v_2)\,(Q_{1,j}-Q_{0,j}) = 0\,,$$

or, equivalently,

$$\alpha\,n\,(P_{n,j}-P_{n-1,j}) + \beta(m-j)\,(P_{n,j+1}-P_{n,j}) + j\,(P_{n,j}-P_{n,j-1}) + \gamma\,n\,(Q_{1,j}-Q_{0,j}) = 0\,.$$

Now, let $\alpha = \gamma$, we have :

$$\alpha\,n\,(Q_{1,j}-P_{n-1,j}) + \beta(m-j)\,(P_{n,j+1}-P_{n,j}) + j\,(P_{n,j}-P_{n,j-1}) = 0\,,$$

and, for $\beta = \frac{j}{m-j}$, this reduces again to :

$$\alpha\,n\,(Q_{1,j}-P_{n-1,j}) + (P_{n,j+1}-P_{n,j-1}) = 0\,,$$

and, finally, a sufficient condition is to have the quad $Q_{1,j}, P_{n,j+1}, P_{n-1,j}, P_{n,j-1}$ planar.

As a conclusion to this discussion, we have exhibited two different conditions for G^1 continuity. Note that other conditions may also be found. Obviously, based on the degree n and m and due to the necessity of considering the continuity for several patches at the same time the construction of G^1 surface is not trivial. Also when the degree of the interface is not the same, this work becomes more complicated (involving, for instance, the degree elevation technique).

13.6.1 Composite Bézier triangles

Let us consider two Bézier triangles of the same degree n, say $\sigma_1(u_1, v_1, w_1)$ and $\sigma_2(u_2, v_2, w_2)$, which share a common interface, for instance for $w_1 = 0$ and $w_2 = 0$. The C^0 continuity holds since $\sigma_1(u_1, v_1, 0) = \sigma_2(u_2, v_2, 0)$. If $P_{i,j,k}$ (resp. $Q_{i,j,k}$) stand for the control points of σ_1 (resp. σ_2), the above condition leads to having :

$$\sum_{i+j+k=n} B^n_{i,j,k}(u_1, v_1, 0)\, P_{i,j,0} = \sum_{i+j+k=n} B^n_{i,j,k}(u_2, v_2, 0)\, Q_{i,j,0}\,,$$

when $u_1 = u_2$ and $v_1 = v_2$. Then, as expected, a C^0 condition is :

$$P_{i,j,0} = Q_{i,j,0} \quad \text{for} \quad i + j = n\,.$$

To discuss high order continuity, we have to return to the directional derivatives of the surface. Let us recall the corresponding formula, for $w = 0$ and patch σ_1 :

$$n \sum_{i+j+k=n-1} B^{n-1}_{i,j,k}(u, v, 0)\, (u_{AB}\, P_{i+1,j,0} + v_{AB}\, P_{i,j+1,0} + w_{AB}\, P_{i,j,1})\,,$$

where AB is the direction of derivation.

Remark 13.7 *It is obvious to see that the directional derivatives along the interface match automatically. Indeed, since $w_{AB} = 0$, the C^0 continuity insures this result.*

Now, for an arbitrary direction AB, if we have :

$$n \sum_{i+j+k=n-1} B^{n-1}_{i,j,k}(u, v, 0)\, (u_{AB}\, P_{i+1,j,0} + v_{AB}\, P_{i,j+1,0} + w_{AB}\, P_{i,j,1})\,,$$

for the AB-derivative of σ_1, then, we have :

$$-n \sum_{i+j+k=n-1} B^{n-1}_{i,j,k}(u, v, 0)\, (u_{AB}\, Q_{i+1,j,0} + v_{AB}\, Q_{i,j+1,0} + w_{AB}\, Q_{i,j,1})\,,$$

for the AB-derivative of σ_2. Thus, the C^1 continuity holds since a relationship like :

$$\alpha\, P_{i+1,j,0} + \beta\, P_{i,j+1,0} + \gamma\, P_{i,j,1} + \gamma\, Q_{i,j,1} = 0 \quad \text{holds}.$$

In other words the two triangles $P_{i+1,j,0}\, P_{i,j+1,0}\, P_{i,j,1}$ and $P_{i+1,j,0}\, P_{i,j+1,0}\, Q_{i,j,1}$ are coplanar and points $P_{i,j,1}$ and $Q_{i,j,1}$ are related one with the other (symmetry).

Note that conditions insuring a G^1 take the same aspect, we have now :

$$\alpha\, P_{i+1,j,0} + \beta\, P_{i,j+1,0} + \gamma\, P_{i,j,1} + \delta\, Q_{i,j,1} = 0\,,$$

where $\delta = k\,\gamma$.

Also, as a conclusion, we return to the final observations about Bézier quads. In particular, a degree 3 patch is too rigid and patches of at least degree 4 must be used.

13.6.2 Other composite surfaces

Patches other than the two above can be considered to develop composite surfaces. For instance, the Gregory and the Walton patches, as briefly introduced, can serve as support for such a construction (see the next section for a more complete discussion).

13.7 Explicit construction of a composite surface

In this section, we consider a quite different approach (as in Section 12.3). The main idea is to use as data a discrete approximation of a given surface. This approximation is indeed a surface mesh composed of triangles. Triangle vertices and normals at these vertices are assumed. Then, the construction is completed triangle by triangle using the available information (singularities, normals at the triangle vertices) is such a way as to obtain a G^1 surface.

The construction is made in two main steps. The first step concerns the construction of a Bézier curve of degree 3 based on the edges of the surface mesh by taking into account the corresponding normals. The second step, using these curve definitions, concerns the construction of the patch.

13.7.1 Constructing the curves boundary of a patch

An edge is characterized by its two endpoints and the two corresponding (surface) normals, thus, 12 degrees of freedom. Hence, a Bézier curve of degree 3 is sought (which has the same number of degrees). Let $\gamma(t)$ be the curve associated with a given edge. We assume that :

$$\gamma(t) = \sum_{i=0}^{3} B_{i,3} P_i\,,$$

where the control points are P_0 the first endpoint of the edge, P_3 its second endpoint and :

$$P_1 = P_0 + V_0 \quad \text{and} \quad P_2 = P_3 - V_2\,,$$

where V_0 and V_2 must be properly defined (after which V_1 will be $V_1 = P_2 - P_1$). Due to the choice of a degree 3 Bézier curve, we have :

$$\gamma'(t) = 3 \sum_{i=0}^{2} B_{i,2} V_i \quad \text{and} \quad \gamma''(t) = 6 \sum_{i=0}^{1} B_{i,1}(V_{i+1} - V_i)\,.$$

Then,

$$\gamma'(t) = 3\,((1-t)^2 V_0 + 2t(1-t)\, V_1 + t^2 V_2) \quad \text{thus} \quad \gamma'(0) = 3\, V_0 \quad \text{and} \quad \gamma'(1) = 3\, V_1\,.$$

$$\gamma''(t) = 6\,((1-t)(V_1 - V_0) + t(V_2 - V_1) \quad \text{thus}$$

$$\gamma''(0) = 6\,(V_1 - V_0) \quad \text{and} \quad \gamma''(1) = 6\,(V_2 - V_1)\,.$$

Now, following Section 11.1.2, the curve normal, $\nu(t)$, at t, is parallel to :

$$\langle\gamma'(t),\gamma'(t)\rangle\gamma''(t) - \langle\gamma'(t),\gamma''(t)\rangle\gamma'(t) .$$

Then, for $t = 0$, we have :

$$\vec{\nu}(0) \quad // \quad \langle\vec{V_0},\vec{V_0}\rangle(\vec{V_1}-\vec{V_0}) - \langle\vec{V_0},(\vec{V_1}-\vec{V_0})\rangle\vec{V_0} \quad i.e.,$$

$$\vec{\nu}(0) \quad // \quad \langle\vec{V_0},\vec{V_0}\rangle\vec{V_1} - \langle\vec{V_0},\vec{V_1}\rangle\vec{V_0} = (\vec{V_0}\wedge\vec{V_1})\wedge\vec{V_0}$$

and, for $t = 1$, a similar result holds :

$$\vec{\nu}(1) \quad // \quad \langle\vec{V_2},\vec{V_2}\rangle\vec{V_1} - \langle\vec{V_2},\vec{V_1}\rangle\vec{V_2} = (\vec{V_2}\wedge\vec{V_1})\wedge\vec{V_2} .$$

We impose the same properties not for the curve normals, $\nu(0)$ and $\nu(1)$, but for the surface normals n_0 and n_1. Indeed, we assume that $\nu(0)$ (resp. $\nu(1)$ matches n_0 (resp. n_1). In other words,

$$n_0 \quad // \quad (V_0\wedge V_1)\wedge V_0 ,$$

$$n_1 \quad // \quad (V_2\wedge V_1)\wedge V_2 ,$$

thus, using the first relation, n_0, V_0 and $V_0\wedge V_1$ form a basis. Since we assume that both n_0 and n_1 are non null, V_0 and V_1 are linearly independent then, n_0, V_0 and V_1 form a basis. As a consequence, three coefficients exist such that :

$$n_1 = \alpha_1\, n_0 + \beta_1\, V_0 + \gamma_1\, V_1 ,$$

and, similarly,

$$n_0 = \alpha_0\, n_1 + \beta_0\, V_1 + \gamma_0\, V_2 .$$

These can be written in terms of P_3P_0 and V_0 and V_2 only since $V_1 = P_0P_3 - V_0 - V_2$.

$$n_1 = \alpha_1\, n_0 + \beta_1\, V_0 + \gamma_1\,(P_0P_3 - V_0 - V_2) ,$$

$$n_0 = \alpha_0\, n_1 + \beta_0\,(P_0P_3 - V_0 - V_2) + \gamma_0\, V_2 ,$$

and, a simple calculation shows that :

$$V_0 = \alpha_1 D + \beta_1 n_0 + \gamma_1 n_1 ,$$

$$V_2 = \alpha_2 D + \beta_2 n_0 + \gamma_2 n_1 ,$$

where D stands for P_0P_3 and the above (new) coefficients must be determined. We assume that $\gamma''(0)$ is parallel to n_0, then

$$\begin{aligned} V_1 - V_0 &= D - V_2 - 2\,V_0 \\ &= (1-\alpha_2-2\alpha_1)\,D + (-\beta_2-2\beta_1)\,n_0 + (-\gamma_2-2\gamma_1)\,n_1 \quad // \quad n_0 \end{aligned}$$

i.e.,

$$1-\alpha_2-2\alpha_1 = 0 \quad \text{and} \quad \gamma_2+2\gamma_1 = 0 .$$

Now, we assume that $\gamma''(1)$ is parallel to n_1, and we have :

$$\begin{aligned} V_2 - V_1 &= 2V_2 + V_0 - D \\ &= (-1+\alpha_1+2\alpha_2)\,D + (2\beta_2+\beta_1)\,n_0 + (2\gamma_2+\gamma_1)\,n_1 \quad // \quad n_1 \end{aligned}$$

i.e.,

$$1-\alpha_1-2\alpha_2 = 0 \quad \text{and} \quad \beta_1+2\beta_2 = 0 .$$

As a first result, we have $\alpha_1 = \alpha_2 = \frac{1}{3}$. Now, we express that :

$$\langle\vec{V_0},\vec{n}_0\rangle = \langle\vec{V_1},\vec{n}_1\rangle = 0 ,$$

then,

$$0 = \frac{1}{3}\langle\vec{D},\vec{n}_0\rangle + \beta_1 + \gamma_1\langle\vec{n}_0,\vec{n}_1\rangle$$

$$0 = \frac{1}{3}\langle\vec{D},\vec{n}_1\rangle + \beta_2\langle\vec{n}_0,\vec{n}_1\rangle + \gamma_2$$

and, replacing β_1 in terms of β_2 and γ_2 in terms of γ_1, these two relations lead to :

$$0 = \frac{1}{3}\langle\vec{D},\vec{n}_0\rangle - 2\beta_2 + \gamma_1\langle\vec{n}_0,\vec{n}_1\rangle \quad \text{and}$$

$$0 = \frac{1}{3}\langle\vec{D},\vec{n}_1\rangle + \beta_2\langle\vec{n}_0,\vec{n}_1\rangle - 2\gamma_1 ,$$

which enable us to find :

$$\beta_2 = \frac{1}{3}\frac{2\langle\vec{D},\vec{n}_0\rangle + \langle\vec{n}_1,\vec{n}_0\rangle\langle\vec{D},\vec{n}_1\rangle}{4-\langle\vec{n}_1,\vec{n}_0\rangle^2} \quad \text{and}$$

$$\gamma_1 = \frac{1}{3}\frac{2\langle\vec{D},\vec{n}_1\rangle + \langle\vec{n}_1,\vec{n}_0\rangle\langle\vec{D},\vec{n}_0\rangle}{4-\langle\vec{n}_1,\vec{n}_0\rangle^2} .$$

Thus, all the coefficients are determined and V_0 and V_2 are known.

$$V_0 = \frac{||D||}{18}\,(6d - 2\rho n_0 + \sigma n_1)$$

$$V_2 = \frac{||D||}{18}\,(6d + \rho n_0 - 2\sigma n_1) ,$$

where $d = \frac{D}{||D||}$ and ρ and σ are respectively

$$\rho = 6\frac{2\langle\vec{d},\vec{n}_0\rangle + \langle\vec{n}_1,\vec{n}_0\rangle\langle\vec{d},\vec{n}_1\rangle}{4-\langle\vec{n}_1,\vec{n}_0\rangle^2} ,$$

$$\sigma = 6\frac{2\langle\vec{d},\vec{n}_1\rangle + \langle\vec{n}_1,\vec{n}_0\rangle\langle\vec{d},\vec{n}_0\rangle}{4-\langle\vec{n}_1,\vec{n}_0\rangle^2} .$$

As a conclusion, the four control points of the Bézier curve corresponding to a given edge, say AB, are determined as :

$$\left\{\begin{array}{lcl} P_0 & = & A \\ P_3 & = & B \\ P_1 & = & P_0 + \dfrac{\|\vec{D}\|}{18}\left(6\vec{d} - 2\rho\vec{n}_0 + \sigma\vec{n}_1\right) \\ P_2 & = & P_3 - \dfrac{\|\vec{D}\|}{18}\left(6\vec{d} + \rho\vec{n}_0 - 2\sigma\vec{n}_1\right) . \end{array}\right. \qquad (13.15)$$

13.7.2 Construction of a patch

Given a triangle (S_1, S_2, S_3) and the three normals $\vec{n}_1, \vec{n}_2$ and $\vec{n}_3$, the above construction results in three Bézier curves of degree 3, one for the edge $[S_1, S_2]$, one for $[S_2, S_3]$ and the last for $[S_3, S_1]$. To create a G^1 patch, we need to use at least degree 4 functions. Thus, we first elevate the degree of the three above curves. Following Chapter 12, the corresponding control points are, for a given curve,

$$L_i = \frac{i\,P_{i-1} + (4-i)\,P_i}{4} ,$$

for $i = 0, 4$. Indeed, we have $L_0 = P_0$, $L_4 = P_3$ while the other L_i's conform to the above formula. By using this formula for the three edges, we obtain the 12 control points that define the patch boundaries.

Hence, we have the 12 patch boundary control points ready. The issue is now how to define the three internal control points in such a way as to ensure G^1 continuity. Formally speaking, the complete control point table associated with a patch of degree 4 is as follows :

$$\begin{array}{c} P_{0,4,0} \\ P_{1,3,0} \quad P_{0,3,1} \\ P_{2,2,0} \quad P_{1,2,1} \quad P_{0,2,2} \\ P_{3,1,0} \quad P_{2,1,1} \quad P_{1,1,2} \quad P_{0,1,3} \\ P_{4,0,0} \quad P_{3,0,1} \quad P_{2,0,2} \quad P_{1,0,3} \quad P_{0,0,4} \end{array}$$

In this control points table, three points are missing. Indeed, in terms of the triangle vertices and edges, we have (with evident notations including these triangle vertices and the $L_{k,i}$'s where $L_{k,i}$ stands for the i^{th} control points of edge k)

$$\begin{array}{c} S_3 \\ L_{2,1} \quad L_{1,3} \\ L_{2,2} \quad ? \quad L_{1,2} \\ L_{2,3} \quad ? \quad ? \quad L_{1,1} \\ S_1 \quad L_{3,1} \quad L_{3,2} \quad L_{3,3} \quad S_2 \end{array}$$

In other words, we have :

$$\begin{array}{c} P_{4,0,0} = S_1 \quad P_{0,4,0} = S_3 \quad P_{0,0,4} = S_2 , \\ P_{0,i,4-i} = L_{1,i} \quad P_{i,4-i,0} = L_{2,i} \quad P_{4-i,0,i} = L_{3,i} . \end{array} \qquad (13.16)$$

And we still need to define the control points inside the patch, *i.e.*, the points $P_{1,1,2}$, $P_{1,2,1}$ and $P_{2,1,1}$ in order to complete the definition of $\sigma(r,s,t)$ which is given by :

$$\sigma(r,s,t) = \sum_{i+j+l=4} B^4_{i,j,l}(r,s,t) P_{i,j,l} . \qquad (13.17)$$

This construction is rather technical and it is completed in several steps. First, we consider the Bézier curves associated with the tangent vectors of the three edges[8] as already introduced :

$$\gamma'_k(t) = 3\sum_{i=0}^{2} B_{i,2} V_{k,i} \quad \text{for} \quad k = 0, 1, 2 ,$$

with $V_{k,i} = L_{k,i+1} - L_{k,i}$. Now, based on the $V_{k,i}$'s, we define :

$$A_{k,0} = n_{k+1} \wedge \frac{V_{k,0}}{\|V_{k,0}\|} \quad \text{and} \quad A_{k,2} = n_{k+2} \wedge \frac{V_{k,2}}{\|V_{k,2}\|} ,$$

as well as :

$$A_{k,1} = \frac{A_{k,0} + A_{k,2}}{\|A_{k,0} + A_{k,2}\|} .$$

Using these coefficients, we construct the quadratic Bézier $h_k(t)$ by :

$$h_k(t) = \sum_{i=0}^{2} A_{k,i} B_{i,2}(t) .$$

This curve allows us to define the tangent along edge k. It can then be shown that the $\gamma'_k(t)$'s and the $h_k(t)$'s allow us to define a suitable continuity between the patches. One constructs the values $D_{k,i}$ for $k = 1, 2, 3$ and $i = 0, 1, 2, 3$.

$$D_{1,i} = P_{1,i,3-i} - \frac{P_{0,i+1,3-i} + P_{0,i,4-i}}{2}$$

$$D_{2,i} = P_{i,3-i,1} - \frac{P_{i+1,3-i,0} + P_{i,4-i,0}}{2}$$

$$D_{3,i} = P_{3-i,1,i} - \frac{P_{3-i,0,i+1} + P_{4-i,0,i}}{2} .$$

Using these $D_{k,i}$'s and the $\gamma'_k(t)''$s, we obtain the $\lambda_{k,i}$'s, for $k = 1, 2, 3$.

$$\lambda_{k,0} = \frac{\langle \vec{D}_{k,0}, \vec{V}_{k,0} \rangle}{\langle \vec{V}_{k,0}, \vec{V}_{k,0} \rangle} ,$$

[8] Index k, from 1 to 3, refers to edge number k. Remember that edge k is opposite vertex k, then, in what follows, as we need to write the normals n_k corresponding to edge k, we will meet, for instance, n_4 which must be identified with n_1, etc.

$$\lambda_{k,1} = \frac{\langle \vec{D}_{k,3}, \vec{V}_{k,2} \rangle}{\langle \vec{V}_{k,2}, \vec{V}_{k,2} \rangle}.$$

Using the $D_{k,i}$'s and the $h_k(t)$'s, we obtain the $\mu_{k,i}$'s, for $k = 1, 2, 3$.

$$\mu_{k,0} = \langle \vec{D}_{k,0}, \vec{A}_{k,0} \rangle$$

$$\mu_{k,1} = \langle \vec{D}_{k,3}, \vec{A}_{k,2} \rangle .$$

From the $\lambda_{k,i}$'s, the $\mu_{k,i}$'s, the $L_{k,i}$'s, the $A_{k,i}$'s and the $V_{k,i}$'s, we deduce the $G_{k,i}$'s, for $k = 1, 2, 3$ and $i = 1, 2$.

$$G_{k,1} = \frac{L_{k,1} + L_{k,2}}{2} + 2\frac{\lambda_{k,0} V_{k,1}}{3} + \frac{\lambda_{k,1} V_{k,0}}{3} + 2\frac{\mu_{k,0} A_{k,1}}{3} + \frac{\mu_{k,1} A_{k,0}}{3}$$

$$G_{k,2} = \frac{L_{k,2} + L_{k,3}}{2} + \frac{\lambda_{k,0} V_{k,2}}{3} + 2\frac{\lambda_{k,1} V_{k,1}}{3} + \frac{\mu_{k,0} A_{k,2}}{3} + 2\frac{\mu_{k,1} A_{k,1}}{3} .$$

In other words, 6 virtual control points have been defined where each pair, for k, corresponds to one edge. These points are then combined and, in this way, the desired control points may be obtained. This gives successively :

$$\begin{aligned} P_{1,1,2} &= \frac{r\,G_{3,2} + s\,G_{1,1}}{r+s}, \\ P_{1,2,1} &= \frac{t\,G_{1,2} + r\,G_{2,1}}{t+r}, \\ P_{2,1,1} &= \frac{s\,G_{2,2} + t\,G_{3,1}}{s+t}. \end{aligned} \tag{13.18}$$

The patch is then well defined via Expression (13.17) and Relationships (13.16) and (13.18). As a conclusion, the resulting surface is G^1 as proved in [Piper-1987].

Remark 13.8 *Note that the three internal control points are combinations of 6 points and depend on the barycentric coordinates. In fact, we encounter a Gregory type patch.*

Remark 13.9 *The case where singularities (such as corners or ridges) exist leads to a more subtle construction where these singularities are fixed.*

A brief conclusion

As for the curves, numerous definitions of surfaces exist, these being of various degrees of difficulty. The context of the application and the desired geometric properties usually make it possible to decide which representation is suitable for the foreseen at hand.

The surface meshing techniques must, as with the curves, as far as possible remain independent of the representation chosen to construct these surfaces.

Moreover, the same phenomenon of perverse numerical effects as for the curves occurs with surfaces. These must be taken into account later.

Chapter 14

Curve meshing

Introduction

Curve meshing is one of the main steps in the meshing process of planes, surfaces and volumes. In fact, most of the automatic mesh generation methods for domains in $\mathbb{R}^2$ or $\mathbb{R}^3$ build the desired covering up from the data of the boundary meshes delimiting the domain considered. In two dimensions, this boundary is naturally formed by a set of curves. The same is true for surfaces. In three dimensions, the boundary of a domain is formed by a set of surfaces whose boundaries again define a set of curves.

As we have already seen, the mesh of a domain is strictly dependent on the mesh of its boundary (Chapters 5 to 7). Thus, the properties of the latter are one of the parameters influencing the quality of the final mesh.

From a topological point of view, a curve is *a priori* a one-dimensional entity. However, when it is a component of a higher-dimensional entity (a planar region, a surface or a boundary of a solid), it must be treated in a multi-dimensional space. Hence, any control at the level of the mesh of a domain of $\mathbb{R}^2$ or $\mathbb{R}^3$, induces a similar control at the level of the curves of this domain. For instance, a size and/or directional specification concerning the mesh elements is translated into a specification onto the curves of the domain. Here, we again encounter a governed meshing problem, which is either isotropic or anisotropic. To a control based on considerations related to the envisaged application (a finite element computation, for instance) is added a control of a purely geometric nature. The desired mesh must be a good approximation, in a sense that we will specify, of the curve geometry.

The curve meshing problem concerns two aspects that must be combined judiciously in order to obtain satisfactory results, a geometrical aspect and an aspect related to the envisaged application. To distinguish clearly between these two cases, we will define the notion of *geometric mesh* and indicate how to construct such a mesh. Then, we will show how to generate a *computational mesh*, which is a mesh that respects the curve geometry while also satisfying specific requirements related to the application.

★
★ ★

This chapter introduces several methods to construct the mesh of a given curve. By mesh, we mean here a piecewise linear approximation of the curve, that is a discretization of the curve with straight segments (*i.e.*, a P^1*-type mesh*). The case of meshes with other kinds of elements (for instance, P^2*-type meshes* with parabola arcs) will be covered in Chapter 22.

For reasons that will appear obvious later (at least we hope so), we first look at how to mesh a (straight) segment. In this particular case, any mesh is geometric (*i.e.*, the geometry is perfectly approached at any point). In other words, there is no geometric-type problem. We will then discuss the case of a parametric curve and that of a curve defined by a discretization. Following this discussion, we will show how to construct a discrete representation of a curve. This being done, a given curve can be replaced by a set of straight segments and the initial meshing technique can then be used, allowing for some slight modifications, to construct the desired mesh. Finally, to conclude this chapter, we will briefly deal with the case of curves in $\mathbb{R}^3$, which are "dangling" or supported by a known surface.

Before going more deeply into the subject, let us mention, perhaps surprisingly, that the (too) rare references on this topic are not prolix, especially concerning the application related to finite elements or numerical computations[1].

14.1 Meshing a segment

In this section, we start with the, *a priori* trivial, problem of meshing a straight segment. Then, we look at the problem of meshing a curve and we show that several similarities exist between these two problems.

14.1.1 Classical segment meshing

Meshing (discretizing) a segment consists in subdividing it into a series of sub-segments of suitable lengths. In practice, these lengths depend on the objectives aimed at and the data (metric specifications) available.

Minimal mesh specifications. In this case, the information provided is relatively simple. The user indicates explicitly what he would like to obtain, for instance :

- a given number of subdivisions (assumed to be of equal size),
- a given length (*i.e.*, a size or a step) for each mesh element,
- an element-size variation along the segment,
- etc.

[1] At least to our knowledge.

According to the given requirements, the aim is to find, depending on the cases, the size and/or the number of sub-segments to construct.

Endpoint size specifications. Here, we assume known, on each segment, information related to the sizes h_1 and h_2 desired at its endpoints. The problem then comes down to using these data in order to deduce a reasonable series of sub-segments (their size and number) so that the first (resp. the last) sub-segment reflects as far as possible the requirement, that it has a size close to h_1 (resp. h_2). Moreover, the intermediate sub-segments must have sizes varying smoothly (monotonically) between the sizes at the endpoints. This variation can be linear, geometric or different depending on the ratio between the two given sizes and the objective sought. Thus, if h_1 and h_2 are close, the type of variation has little influence on the discretization. However, if these two values are very different and if the (classical) length of the segment allows, noticeably different results (in terms of the number of sub-segments and their distribution) can be obtained depending on the particular size variation function chosen (one function emphasizing the small sizes, another having the opposite effect).

General specification map. A more general case is that of a given field of metrics. Such a field can be continuous (analytical case) or discrete (known at certain specific points only). Moreover, the field can be isotropic (it corresponds to sizes) or anisotropic (it specifies sizes in privileged directions). More formally speaking, let us recall that an isotropic field of metrics corresponds to a field of matrices of the form :

$$\mathcal{M} = \frac{1}{\lambda} I_d \,,$$

I_d being the identity matrix and $\lambda = h^2$, where h is the desired size. In two dimensions and in the anisotropic case, the matrices are of the type :

$$\mathcal{M} = \begin{pmatrix} a & b \\ b & c \end{pmatrix} ,$$

which can also be written as :

$$\mathcal{M} = {}^t\mathcal{D} \begin{pmatrix} \frac{1}{h_1^2} & 0 \\ 0 & \frac{1}{h_2^2} \end{pmatrix} \mathcal{D} \,,$$

where $\mathcal{D}$ represents the directions to follow and h_1 (resp. h_2) the sizes in these directions.

If the values h, h_1, h_2 and the directions of $\mathcal{D}$ do not depend on the spatial position, the field is said to be *uniform*. If, on the other hand, these values depend on the position, they induce a *variable* field.

14.1.2 Isotropic governed meshing

We deliberately leave to one side the two first types of specifications (trivial) in order to focus exclusively on the general problem in the isotropic case (the anisotropic case being discussed in the next section).

Let AB be the given segment. As will be seen later, meshing AB consists essentially in computing the length of the segment and, based on this length and according to the given specifications, subdividing AB into a series of sub-segments of suitable lengths.

Length of a segment AB (general expression). Let $\mathcal{M}(t)$ be the value[2] of the metric (matrix) at point M of parameter $t \in [0,1]$ of the parameterized segment AB. Let us recall that the length of AB with respect to the metric $\mathcal{M}$ is obtained using the formula (Chapter 10) :

$$l_{\mathcal{M}}(AB) = \int_0^1 \sqrt{{}^t\overrightarrow{AB}\mathcal{M}(t)\overrightarrow{AB}}\,dt\,. \tag{14.1}$$

The exact calculation of this value is usually impossible or at best very tedious. Therefore, we consider specific situations so as to use approached calculation formulas.

Calculation of the length of a segment AB (case 1). Suppose the values $\mathcal{M}(0)$ at A and $\mathcal{M}(1)$ at B are known. We are looking for a monotonous function $\mathcal{M}(t)$ for each $t \in [0,1]$ smoothly varying between these two values. Depending on the particular form of $\mathcal{M}(t)$, this problem is equivalent to that of finding a function $h(t)$ equal to h_A (resp. h_B) at $t=0$ (resp. $t=1$) and smoothly varying between these two values. Indeed, if $h(t)$ is known, the Integral (14.1) above becomes :

$$l_{\mathcal{M}}(AB) = \int_0^1 \sqrt{{}^t\overrightarrow{AB}\frac{1}{h^2(t)}I_d\overrightarrow{AB}}\,dt \tag{14.2}$$

and the calculation gives :

$$l_{\mathcal{M}}(AB) = \sqrt{{}^t\overrightarrow{AB}\overrightarrow{AB}}\int_0^1 \frac{1}{h(t)}dt = l(AB)\int_0^1 \frac{1}{h(t)}dt\,, \tag{14.3}$$

where $l(AB)$ denotes the classical Euclidean length of AB. The calculation of this quantity is thus based on the choice of a function $h(t)$. The simplest solution consists in choosing a linear interpolation function, by fixing :

$$h(t) = h_A + t\,(h_B - h_A)\,.$$

[2] In the following, we will make the confusion between $\mathcal{M}(t)$ and $\mathcal{M}(M(t))$, as we make the confusion between $h(t)$ and h_M, M being the point of parameter t.

Another linear solution (in $1/h$, this time) corresponds to the choice :

$$\frac{1}{h(t)} = \frac{1}{h_A} + t\left(\frac{1}{h_B} - \frac{1}{h_A}\right).$$

Finally, another example of function h consists in choosing :

$$h(t) = h_A\left(\frac{h_B}{h_A}\right)^t.$$

The choice of a particular interpolation function obviously influences the nature of the resulting distribution.

Let us consider for instance the last choice for h and let us compute the length of the segment AB. According to Formula (14.3), one has to compute the integral :

$$\int_0^1 \frac{1}{h(t)}dt\,,$$

that is :

$$\frac{1}{h_A}\int_0^1 \left(\frac{h_A}{h_B}\right)^t dt = \frac{1}{h_A}\left[\frac{\left(\frac{h_A}{h_B}\right)^t}{\log\left(\frac{h_A}{h_B}\right)}\right]_{t=0}^{t=1} = \frac{h_A - h_B}{h_A\,h_B\,\log\left(\frac{h_A}{h_B}\right)}\,.$$

We find thus, by multiplying the usual length of AB, the length of this segment for the given field of metric :

$$l_{\mathcal{M}}(AB) = l(AB)\,\frac{h_A - h_B}{h_A\,h_B\,\log(\frac{h_A}{h_B})}\,.$$

Remark 14.1 *Notice that if $h_A = h_B$, the previous formula is undetermined. In fact, in this case, we have $h(t) = h_A$ and thus, we find directly that $l_{\mathcal{M}}(AB) = \frac{l(AB)}{h_A}$. Notice also that if h_B is close to h_A, for instance, $h_B = h_A(1+\varepsilon)$ (for ε small), we obtain the desired length using a limited expansion of the general expression. We thus have $l_{\mathcal{M}}(AB) = \frac{2-\varepsilon}{2\,h_A}\,l(AB)$ and for $\varepsilon = 0$ we retrieve the previous value. Nevertheless, in this case, a geometric function $h(t)$ is not strictly required (it varies only a little). Therefore, a linear function is sufficient (we can easily verify that the choice in $1/h$ gives again, at order 2, the same value for $l_{\mathcal{M}(AB)}$).*

The reader can find in [Laug *et al.* 1996] the values corresponding to other choices of h and, in particular, the first two choices mentioned above.

Calculation of the length of a segment AB (case 2). Only the values $\mathcal{M}(0)$ and $\mathcal{M}(1)$ (*i.e.*, h_A and h_B) are known (at A and at B) but, without justifying any longer the particular choice of the interpolation function, we use only these values and we consider a quadrature formula. For example, we write :

$$l_{\mathcal{M}}(AB) = \frac{l(AB)}{2}\left(\frac{1}{h_A} + \frac{1}{h_B}\right).$$

If h does not vary too much between h_A and h_B, we thus obtain an approximated value relatively close to the exact value. Otherwise, the computed value is only an approximation of the result, and may be quite rough. To get a better answer, we shall consider a function $h(t)$ and apply the same quadrature formula, subdividing the segment into several pieces. The additional information required is then the value of h at each new node of the integration.

Calculation of the length of a segment AB (case 3). Here, we know a series of matrices $\mathcal{M}(t_i)$ at different points along the segment. These points M_i correspond to the parameters t_i. We then retrieve the previous discussion by posing :

$$l_{\mathcal{M}}(AB) = \sum_i l_{\mathcal{M}}(M_i M_{i+1}).$$

There are now two possible solutions. Either we use the method presented in case 1 above and we apply it on each segment $M_i M_{i+1}$, or we use the method described in case 2.

Creation of the sub-segments. This stage of generating a mesh of AB is very simple. We compute the length of AB using one of the previously described possibilities. Let $l = l_{\mathcal{M}}(AB)$ be this length. We pick the closest integer value n to l and we subdivide the segment into n sub-segments of length $\frac{l}{n}$ (hence close to 1). Notice that this unit value, depending on the nature of the metric, leads to variable (Euclidean) lengths in each sub-segment (which is, obviously, the desired goal).

14.1.3 Anisotropic meshing

We follow here the same idea as in the isotropic case. At first, we compute the length of the segment in the field of metric, then, depending on the value found, we split the segment into sub-segments of unit length. The computation of the desired length is performed, as previously, using three different solutions, depending on the input data and the fixed choices.

Calculation of the length of a segment AB (case 1). Only the metrics at the endpoints are given. In practice, we have two matrices (non diagonal) and the calculation of the desired length consists of constructing an interpolated metric between the two known values. To this end, we use the method described in

Chapter 10 that allows us to obtain, depending on the choices made, a metric at each point of the segment.

A metric being known everywhere, the length can be obtained using an approximated calculation, for instance, using dichotomy.

Calculation of the length of a segment AB (case 2). This case leads to computing the desired length approximately by numerical integration of the general formula.

Calculation of the length of a segment AB (case 3). We have a series of matrices at different points along the segment. We retrieve the principle described in the isotropic case, applied here following one of the methods seen above.

Creation of sub-segments. The mesh of AB is carried out as in the isotropic case.

Remark 14.2 *Notice the whole discussion corresponds exactly to what was performed in the internal point creation method proposed in Chapter 7. The current mesh edges, used as a geometric support for this construction, are indeed straight segments.*

14.1.4 Examples of straight segment meshes

First, we consider here two examples of isotropic nature. In the first example, the metric is represented in Figure 14.1 where the points Q_i supporting the sizing information are shown. Table 14.1 indicates the isotropic specifications at points Q_i.

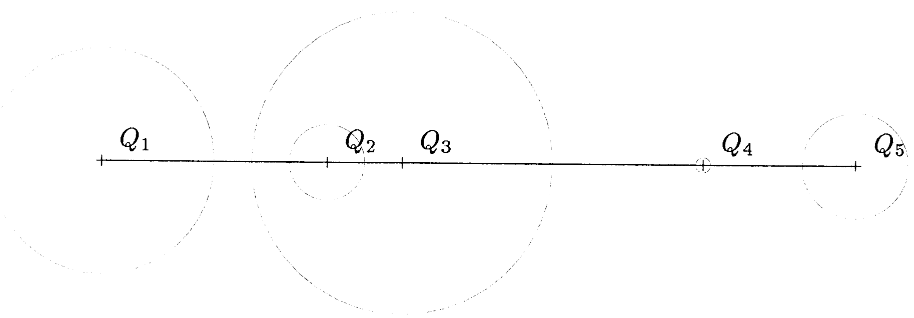

Figure 14.1: *Discrete (isotropic) size specification.*

Figure 14.2 shows the resulting mesh, for a choice corresponding to a linear propagation (the function $h(t)$), for a segment when a discrete size specification is provided. The mesh has 17 elements. Figure 14.2 shows (bottom) the endpoints R_i of the elements. Above, one can see the points Q_i defining the metric. At the top, the function $h(t)$ is represented and, by means of rectangles, its approximation on each element.

Q_i	Q_1	Q_2	Q_3	Q_4	Q_5
s_i	0.0	3.0	4.0	8.0	10.0
h_i	1.5	0.5	2.0	0.1	0.7

Table 14.1: Discrete isotropic specifications at points Q_i.

Figure 14.3 shows the resulting mesh when a geometric propagation is chosen (the function $h(t)$), for the segment when the same discrete size specification is provided (Figure 14.1 and Table 14.1). The mesh now has 26 elements. Notice that the smallest sizes are privileged by this type of distribution function.

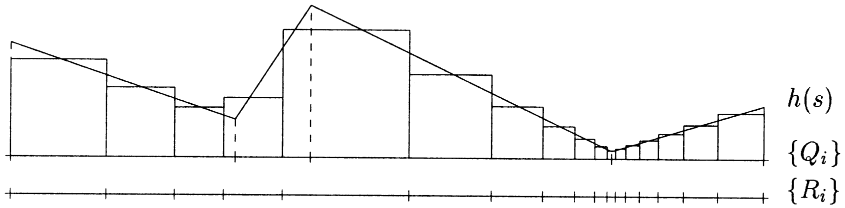

Figure 14.2: *Mesh resulting from a linear interpolation.*

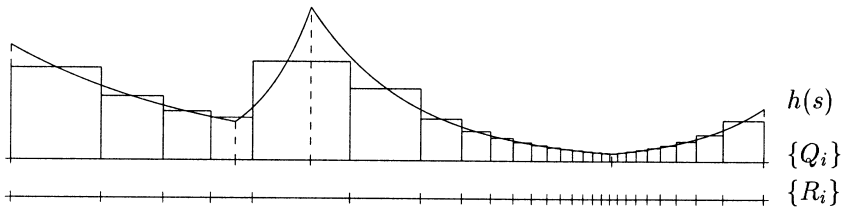

Figure 14.3: *Mesh resulting from a geometric interpolation.*

We now consider two anisotropic examples. The metric specification is also discrete. It corresponds to the data (Table 14.2) of directions and sizes at points Q_i illustrated in Figure 14.4. Two types of propagation function are represented. Figure 14.5 shows the result obtained by fixing a linear function whereas Figure 14.6 corresponds to the case of a geometric function. The figures show the mesh of the segment (bottom) and the evolution of the metrics (*i.e.*, the interpolation between the metrics known at points Q_i) along the segment AB (top).

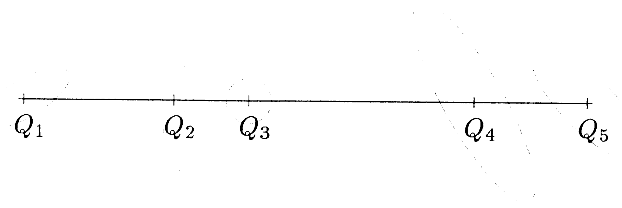

Figure 14.4: *Discrete specifications of directions and sizes.*

Q_i	Q_1	Q_2	Q_3	Q_4	Q_5
s_i	0.0	2.0	3.0	6.0	7.5
θ_i	30°	90°	0°	−60°	−45°
$h_{1,i}$	0.70	1.10	0.30	1.50	1.00
$h_{2,i}$	0.25	0.15	0.30	0.40	0.35

Table 14.2: Discrete anisotropic specifications at points Q_i.

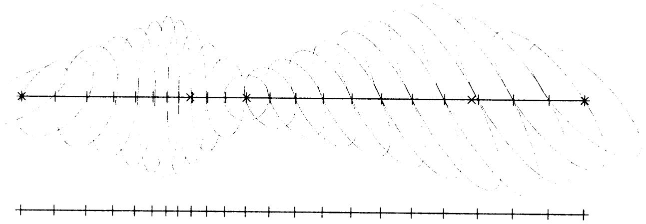

Figure 14.5: *Mesh resulting from a linear interpolation.*

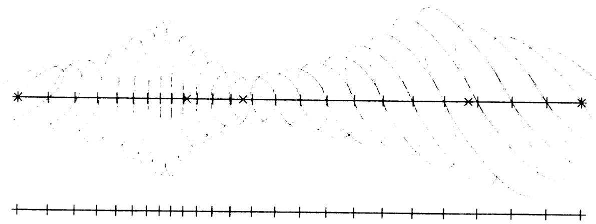

Figure 14.6: *Mesh resulting from a geometric interpolation.*

14.2 Meshing a parametric curve

We are now interested in the case of curves. The principle remains the same. First, we compute the length of the curve for the given field of metrics, then we subdivide this curve into segments, so as to obtain the corresponding unit arc length (*i.e.*, close to 1).

However, unlike the case of a segment, here we must follow the geometry of the curve so as to make sure that the resulting mesh is close to it, in some sense. Therefore, if the given metric is not geometric by nature, it has to be corrected to take this requirement into account. To see this more clearly, we look first at a naive example.

A naive though probably wrong method. As an exercise, we analyze on a simple case of a sinusoid, two examples of meshes having uniform sized edges. Although similar in refinement, it is obvious (in Figure 14.7) that the left-hand side mesh is much better than the right-hand side mesh. The choice of a smaller stepsize would obviously lead to two almost identical meshes. This simple example emphasizes the influence of the point location, given a stepsize. The stepsize influence is very easy to see.

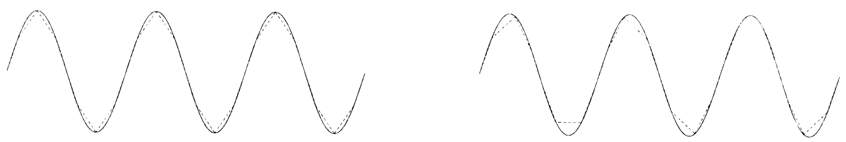

Figure 14.7: *Uniform meshes. Left-hand side, the mesh (dashed lines) "match" the geometry (full lines), right-hand side, the mesh has (roughly) the same stepsize but is "shifted".*

This leads to analyzing how to mesh a curve with respect to its geometry, first without an explicit field, then given such a field.

14.2.1 Geometric mesh

We wish to mesh a curve in such a way that its geometry is respected. The metric to follow is thus strictly related to the geometry (the case where a different metric is specified will be discussed later). The aim of this section is to show how to control the meshing process, that is to define the control metric so as to use it to mesh the given curve effectively.

Desired properties and related problems. The immediate question is how to qualify and quantify the parameters to ensure that the mesh follows the geometry. If h is the length of an element (a segment) and if δ measures the smallest distance between this segment and the curve, then, for a given ε, we want to have :

$$\delta \leq \varepsilon h .$$

This inequality defines a relative control (Figure 14.8) that can be interpreted in a simple way : the curve length (the arc of curve) and that of the corresponding chord (the chord sub-tending this arc) are close. If s is the length of the curve corresponding to the chord of length h, this condition can also be expressed as :

$$|h - s| \leq \varepsilon s \quad \text{or} \quad |h - s| \leq \varepsilon h .$$

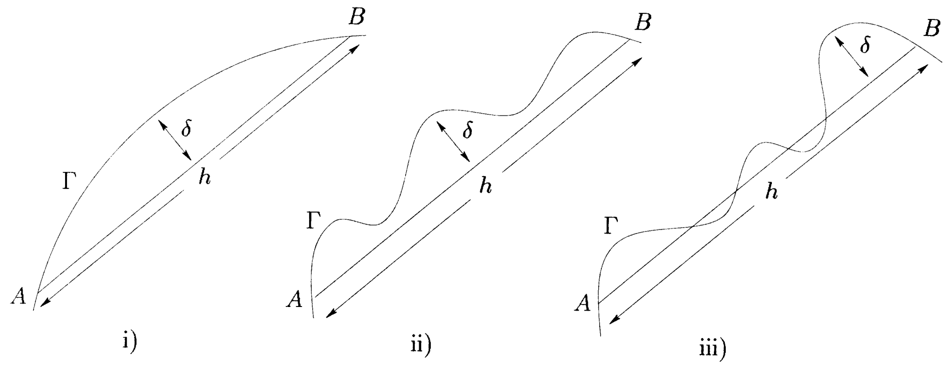

Figure 14.8: *Gap between an arc of the curve* Γ *and the corresponding chord, the chord* AB *of length* h *(i.e., the segment of the mesh supposed to approach this curve).*

The problem is then to find the location of the points along the curve in such a way as to satisfy this property. This problem induces two sub-problems, obviously related one to each other :

- where to locate the points and
- how, given a point, to find the next one.

To illustrate these two questions, let consider the case of a circle. It is obvious that finding a first point M_0 on the circle is easy, as any point will be suitable ! Hence, M_0 being chosen, the second question consists in finding a point M_1, and step by step the series of points M_i, $i = 2, 3, \ldots$ such that the length of the segments $M_i M_{i+1}$ corresponds to the given accuracy. For other types of curves and, especially curves having discontinuities, it is obvious that the points of discontinuity must necessarily be mesh points. Hence, these points being fixed, the second question consists in finding the next points that form closely spaced segments along the curve, with respect to the given accuracy. This being fixed, we assume that the particular points imposed have been identified and we discuss only the case of sufficiently regular curves (without discontinuities) that, in fact, correspond to the different pieces defined between the imposed points.

Back to the local approximation of a curve. As seen in Chapter 11, a limited expansion allows us to study the local behavior of a curve. If Γ is a supposedly sufficiently regular curve described by the function γ and if s denotes the curvilinear abscissa, we can write in the neighborhood of s_0 :

$$\gamma(s) = \gamma(s_0) + \Delta s \gamma'(s_0) + \frac{\Delta s^2}{2}\gamma''(s_0) + \frac{\Delta s^3}{6}\gamma'''(s_0) + \ldots,$$

with $\Delta s = s - s_0$ being sufficiently small to ensure the validity of this development (*i.e.*, the terms corresponding to the ... are negligible). As $\gamma'(s) = \vec{\tau}(s)$ (the tangent), $\gamma''(s) = \frac{\vec{\nu}(s)}{\rho(s)}$ (the ratio between the normal and the radius of curvature) and as $\gamma'''(s) = \frac{-\rho'(s)\,\vec{\nu}(s) - \vec{\tau}(s)}{\rho(s)^2}$, the above expression can also be written as follows :

$$\gamma(s) = \gamma(s_0) + \Delta s\,\vec{\tau} + \frac{\Delta s^2}{2\,\rho(s_0)}\vec{\nu} - \frac{\Delta s^3}{6\,\rho(s_0)^2}(\rho'(s_0)\,\vec{\nu} + \vec{\tau}) + \ldots, \tag{14.4}$$

where $\vec{\tau} = \vec{\tau}(s_0)$ and $\vec{\nu} = \vec{\nu}(s_0)$. Notice that the point P defined by :

$$P = \gamma(s_0) + \Delta s\,\vec{\tau} + \frac{\Delta s^2}{2\,\rho(s_0)}\vec{\nu}$$

i.e., the point of the parabola passing through $\gamma(s_0)$, having a tangent $\vec{\tau}$ at this point and located at the distance Δs from $\gamma(s_0)$, is very close to the *osculating circle* to the curve. Let us consider the point O defined by :

$$O = \gamma(s_0) + \rho(s_0)\,\vec{\nu},$$

then :

$$||\overrightarrow{OP}||^2 = \left\langle \Delta s\,\vec{\tau} + \left(\frac{\Delta s^2}{2\,\rho(s_0)} - \rho(s_0)\right)\vec{\nu},\ \Delta s\,\vec{\tau} + \left(\frac{\Delta s^2}{2\,\rho(s_0)} - \rho(s_0)\right)\vec{\nu}\right\rangle,$$

thus, we have :

$$||\overrightarrow{OP}||^2 = \Delta s^2 + \frac{\Delta s^4}{4\rho(s_0)^2} + \rho(s_0)^2 - \Delta s^2$$

and, therefore :

$$||\overrightarrow{OP}||^2 = \rho(s_0)^2 + \frac{\Delta s^4}{4\rho(s_0)^2} = \rho(s_0)^2\left(1 + \frac{\Delta s^4}{4\,\rho(s_0)^4}\right),$$

thus finally :

$$||\overrightarrow{OP}|| = \rho(s_0)\sqrt{1 + \frac{\Delta s^4}{4\,\rho(s_0)^4}}\,. \tag{14.5}$$

To conclude, when Δs is sufficiently small, the point P above is very close to the circle of center O and of radius $\rho(s_0)$ (*i.e.*, the osculating circle to the curve at M_0). This well-known observation has an important practical consequence. If the point P corresponding to the first three terms of the Equation (14.4) is close to

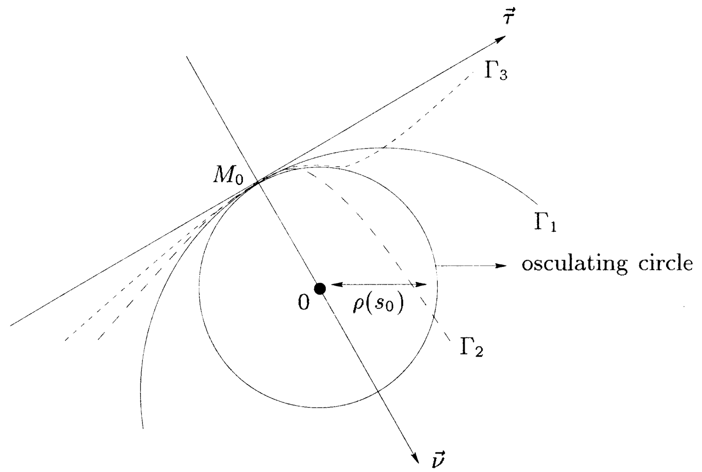

Figure 14.9: *Three types of behavior in a given neighborhood of s_0 (at M_0). We show the curves Γ_1, Γ_2 and Γ_3 passing through M_0, having the same tangent $\vec{\tau}$, the same normal $\vec{\nu}$ and the same osculating circle at M_0. It is obvious that the curve Γ_1 has a behavior such that the analysis at s_0 gives a good indication of what can be expected further, which is not the case for the two other examples.*

the curve, then the point corresponding to the osculating circle (very close to this curve) is very close to this point P. Hence, we can base the analysis on the point of the osculating circle and thus set the desired construction on the corresponding construction on the osculating circle.

The validity of the reasoning is guaranteed as long as the fourth term (and the following ones) of Equation (14.4) is (are) negligible as compared to the previous ones. This leads to a restriction on the stepsize Δs possible (Figure 14.9) depending on the behavior of the curve in a neighborhood of s_0 of size this stepsize Δs. The term we are interested in is the value :

$$\frac{\Delta s^3}{6\,\rho(s_0)^2}(\rho'(s_0)\,\vec{\nu} + \vec{\tau}), \tag{14.6}$$

which, in particular, involves the variation of ρ (*i.e.*, the value ρ'). This term will be negligible if :

$$\frac{\Delta s^3}{6\,\rho(s_0)^2}||\rho'(s_0)\,\vec{\nu} + \vec{\tau}|| \ll \left\|\Delta s\,\vec{\tau} + \frac{\Delta s^2}{2\,\rho(s_0)}\vec{\nu}\right\|.$$

This condition is equivalent to assuming that :

$$\frac{\Delta s^3}{6\,\rho(s_0)^2}||\rho'(s_0)\,\vec{\nu} + \vec{\tau}|| = \varepsilon\left\|\Delta s\,\vec{\tau} + \frac{\Delta s^2}{2\,\rho(s_0)}\vec{\nu}\right\|.$$

choosing a sufficiently small ε. We have thus :

$$\Delta s^2 = 6\,\varepsilon\,\rho(s_0)^2 \sqrt{\frac{1+\dfrac{\Delta s^2}{4\,\rho(s_0)^2}}{1+\rho'(s_0)^2}}\,.$$

This leads to solving an equation (in Δs) of the form :

$$\Delta s^4\,(1+\rho'(s_0)^2) = 6^2\,\varepsilon^2\left(1+\frac{\Delta s^2}{4\,\rho(s_0)^2}\right)\rho(s_0)^4\,,$$

that, posing $\nabla = \Delta s^2$, can be written :

$$(1+\rho'(s_0)^2)\,\nabla^2 - 9\,\varepsilon^2\,\rho(s_0)^2\,\nabla - 36\,\varepsilon^2\,\rho(s_0)^4 = 0\,.$$

This yields :

$$\nabla = \frac{9\varepsilon^2 + 3\,\varepsilon\sqrt{9\varepsilon^2+16\,(1+\rho'(s_0)^2)}}{2\,(1+\rho'(s_0)^2)}\,\rho(s_0)^2 \quad \text{and thus} \quad \Delta s = \sqrt{\nabla}\,.$$

Hence, we have $\Delta s = \alpha\,\rho(s_0)$ and the previous relation leads to choosing as coefficient α a value such that :

$$\alpha \leq \sqrt{\frac{9\varepsilon^2 + 3\,\varepsilon\sqrt{9\varepsilon^2+16\,(1+\rho'(s_0)^2)}}{2\,(1+\rho'(s_0)^2)}} \approx \frac{\sqrt{6\,\varepsilon}}{\sqrt[4]{1+\rho'(s_0)^2}} \tag{14.7}$$

Such a choice ensures that the point (of the parabola) :

$$\gamma(s_0) + \Delta s\,\vec{\tau} + \frac{\Delta s^2}{2\,\rho(s_0)}\,\vec{\nu} = \gamma(s_0) + \alpha\,\rho(s_0)\,\vec{\tau} + \alpha^2\frac{\rho(s_0)}{2}\,\vec{\nu}$$

is close to the curve at the order two, within the following term of Relation (14.4), that is $\alpha^3\dfrac{\rho(s_0)}{6}(\rho'(s_0)\,\vec{\nu}\,+\,\vec{\tau})$ which gives a gap of :

$$\delta = \frac{\sqrt{9\varepsilon^2 + 3\,\varepsilon\sqrt{9\varepsilon^2+16\,(1+\rho'(s_0)^2)}}^{\,3}}{12\sqrt{2}\,(1+\rho'(s_0)^2)}\,\rho(s_0) \approx \frac{\varepsilon\sqrt{6\,\varepsilon}}{\sqrt[4]{1+\rho'(s_0)^2}}\,\rho(s_0)\,. \tag{14.8}$$

As an example, Table 14.3 gives some values of α and of $\dfrac{\delta}{\rho(s_0)}$ according to the given ε and to different values $\rho'(s_0)$. Notice by the way that (and the results in the table confirm it) :

$$\delta \approx \varepsilon\,\alpha\,\rho(s_0)\,.$$

From the previous discussion we deduce that the curve Γ can be analyzed in a neighborhood of $\gamma(s_0)$ of size $\alpha\,\rho(s_0)$, by looking at the parabola corresponding to the first three terms of the limited expansion of γ.

$\rho'(s_0)$	α	α^*	$\delta/\rho(s_0)$
0.0	.245	.2449	.00247
0.5	.232	.231	.00234
1.0	.206	.2059	.00207
$\sqrt{2}$	.186	.186	.00187
$\sqrt{3}$	.173	.173	.00171
2.0	.164	.1638	.00164
3.0	.137	.137	.00138
5.0	.108	.108	.00108
10.0	.07729	.07726	.00077

Table 14.3: Several values of α (and the approached value α^*) determining the stepsize with respect to the gap for $\varepsilon = .01$ and different values of $\rho'(s_0)$.

Let us consider a chord of length $\Delta s = \alpha\,\rho(s_0)$ coming from the point $\gamma(s_0)$ and look at the gap between this chord and the curve. From the previous discussion, we can estimate this gap as the gap between the chord and the parabola. The maximum corresponds to the distance between the midpoint of the parabola (the point P of abscissa $\frac{\alpha}{2}\,\rho(s_0)$) and the point M which is the point of the chord corresponding to the orthogonal projection of P. In the frame $[\gamma(s_0), \vec{\tau}, \vec{\nu}]$, the above parabola is written $y = \dfrac{x^2}{2\,\rho(s_0)}$ and the coordinates of the point P are thus :

$$\frac{\alpha}{2}\,\rho(s_0) \quad \text{and} \quad \frac{\alpha^2}{8}\,\rho(s_0)\,.$$

The normalized equation of the line passing through $\gamma(s_0)$ and $\gamma(\alpha\,\rho(s_0))$ in this frame, is written :

$$\frac{2\,\rho(s_0)\,y}{\sqrt{4\,\rho(s_0)^2+\Delta s^2}} - \frac{\Delta s\,x}{\sqrt{4\,\rho(s_0)^2+\Delta s^2}} = 0\,,$$

or also, with respect to α :

$$\frac{2\,y}{\sqrt{4+\alpha^2}} - \frac{\alpha\,x}{\sqrt{4+\alpha^2}} = 0\,.$$

The distance from this line to the point P, *i.e.*, the distance $||\overrightarrow{PM}||$, if thus of the order of :

$$\delta = \frac{\alpha^2}{4\sqrt{4+\alpha^2}}\,\rho(s_0) \approx \frac{\alpha^2}{8}\,\rho(s_0)\,.$$

Or, depending on ε :

$$\delta = \frac{3\,\varepsilon}{4\sqrt{1+\rho'(s_0)^2}}\,\rho(s_0)\,. \tag{14.9}$$

Before going further, let calculate the quantity $|h-\Delta s|$ where h is the length of the chord sub-tending the arc Δs. We have, using the second order approximation of the curve :

$$h = \sqrt{\alpha^2 + \frac{\alpha^4}{4}}\rho(s_0)\,,$$

hence :

$$|h - \Delta s| = \left|\alpha - \sqrt{\alpha^2 + \frac{\alpha^4}{4}}\right| \rho(s_0)\,,$$

$$|h - \Delta s| = \left|1 - \sqrt{1 + \frac{\alpha^2}{4}}\right| \alpha\,\rho(s_0) \approx \frac{\alpha^2}{8}\Delta s\,,$$

or also :

$$|h - \Delta s| \approx \frac{3\,\varepsilon}{4\sqrt{1+\rho'(s_0)^2}}\Delta s\,. \tag{14.10}$$

This value, measuring the relative gap between the lengths of the arc and that of the curve, will be involved later in the definition of the desired mesh that we propose.

By merging the majorations of Relations (14.8) and (14.9), we find that the gap between the chord and the curve is majorated by :

$$\delta \approx \left(\frac{\varepsilon\sqrt{6\,\varepsilon}}{4\sqrt{1+\rho'(s_0)^2}} + \frac{3\,\varepsilon}{4\sqrt{1+\rho'(s_0)^2}}\right)\rho(s_0)\,.$$

The validity of these above demonstrations is ensured if the terms neglected in Relation (14.4) are in fact negligible. To this end, we are looking for an upper bound and we return to the limited expansion :

$$\gamma(s) = \gamma(s_0) + \Delta s\gamma'(s_0) + \frac{\Delta s^2}{2}\gamma''(s_0) + \frac{\Delta s^3}{6}\gamma'''(s_0) + \ldots\,.$$

But, we know that a value s_j exists between s_0 and s such that :

$$\gamma_j(s) = \gamma_j(s_0) + \Delta s\gamma_j'(s_0) + \frac{\Delta s^2}{2}\gamma_j''(s_0) + \frac{\Delta s^3}{6}\gamma_j'''(s_j)$$

for each component[3] ($j = 1, d$) of γ. If m is an upper bound of $||\gamma'''||$ on the interval $[s_0, s]$ and if we note $r(s)$ the set of terms $\frac{\Delta s^3}{6}\gamma'''(s_0) + \ldots$, then :

$$||r(s)|| \le \frac{\Delta s^3}{6} m\,,$$

hence, setting an accuracy ε' is equivalent to fixing :

$$\frac{\Delta s^3}{6} m = \varepsilon'\,\Delta s\,.$$

[3] Correct for a scalar function, this result is not verified for a vector function. It is true only component by component and s_j depends on the index j of the component.

With the particular form of Δs, we find :

$$\varepsilon = \frac{\sqrt{1+\rho'(s_0)^2}}{\rho(s_0)^2\, m}\varepsilon' \tag{14.11}$$

and, thus, ε must be less than or equal to this value.

Exercise 14.1 *Consider as a curve the quarter of circle defined by :*

$$\gamma(s) = \left(\rho\cos(\frac{s}{\rho})\,,\,\rho\sin(\frac{s}{\rho})\right)\,,$$

find that $m = \frac{1}{\rho^2}$ *and apply Relation (14.11). We then find* $\varepsilon = \varepsilon'$.

From a practical point of view, determining M is not strictly trivial. Therefore, the meshing technique must, somehow, overcome this drawback (see below).

We will then exploit the results obtained to design a meshing method. First, we introduce the notion of a *geometric mesh.*

Geometric mesh of a curve.

Definition 14.1 *A* geometric mesh *of type* P^1*, within a given* ε*, of a curve* Γ *is a piecewise linear discretization of this curve for which the relative gap to the curve at any point is of the order of* ε*. More precisely, if* Δs *is the length of an arc of the curve and if* h *is the length of the corresponding curve, we have (at the limit) when* Δs *tends towards* 0 *:*

$$h = \Delta s$$

or also :

$$\left|\frac{h}{\Delta s} - 1\right| = 0\,,$$

or, for a given ε *:*

$$\left|\frac{h}{\Delta s} - 1\right| \le \varepsilon\,.$$

This last relation can be also expressed as :

$$|h - \Delta s| \le \varepsilon\,\Delta s\,.$$

Remark 14.3 *Notice that other methods can be used to evaluate the gap between the curve and its mesh. Indeed, we can use the value of the surface enclosed between this arc and the corresponding curve rather than computing the difference of length, as above.*

Obviously, a sufficiently (infinitely) fine mesh is necessarily a geometric mesh. However, the mesh we are looking for here must be both geometric and *minimal* (that is having as small a number as possible). For instance, if we subdivide a line segment into 100 segments, we obtain a geometric mesh, whereas the mesh formed by the segment itself is already geometric and (obviously) minimal.

Remark 14.4 *Definition (14.1) is one of the possible definitions. In fact, the notion of a geometric conformity underlying to the notion of a geometric mesh is obviously determined by the application envisaged. In particular, defining the conformity to the geometry by imposing that the discretization be enclosed within a band of a given (small) length corresponds to a different definition, as coherent as the previous one, that leads naturally to a rather different meshing technique.*

If we retain the previous definition, the analysis technique of the local behavior of the curve described previously can serve to construct a geometric mesh. We then replace the parabola approaching the curve by a segment.

The local analysis indicates that, at any point of Γ, the desired length is $\alpha\,\rho(s)$ where s denotes the curvilinear abscissa at this point, α satisfies Condition (14.7) and $\rho(s)$ is the radius of curvature of the curve at s.

In terms of metrics, the Euclidean length $\alpha\,\rho(s)$, can be interpretated as the *unit* length of the metric field defined by the stepsize $\frac{1}{\alpha\,\rho(s)}$, field that can be written in a matrix form as :

$$\mathcal{M}(s) = \frac{1}{\lambda(s)} I_d\,,$$

with $\lambda(s) = \alpha^2\,\rho(s)^2$.

Hence, in terms of metric and unit length, the meshing method consists of constructing a unit mesh for this field of metrics.

Exercise 14.2 *Look at the angular gap between the curve (its tangent) and the mesh resulting from the previous method. Conclusion ?*

Constructing a geometric mesh of a parametric curve consists in applying this principle in its range of validity. Hence, a meshing method consists in :

- identifying the extrema (for the radii of curvature) of the curve as well as the singular points,
- setting these points as mesh vertices,
- subdividing the curve into pieces, each piece being limited by two such points,
- meshing each piece by applying the previous principle.

The meshing of a piece of curve consists then in computing its length for the field of metrics of the radii of curvature, weighted by the coefficients α that have been introduced. We then retrieve exactly the same meshing method as for a straight segment. We search the length l, we round it to the nearest integer n and we split it into segments of length $\frac{l}{n}$. Notice that the lengths computations are performed on the curve (curvilinear abscissa) and that the stepsize of the mesh is the length 1 of the curve. In other words, we identify the length of the curve and that of the chord corresponding to the stepsize.

Remark 14.5 *This meshing process is rather tedious to implement and can only be applied to configurations similar to that represented on the left-hand side of Figure 14.8. The other two situations shown on this figure must be treated in a different manner. Actually, the problem is more one of simplifying the geometry than a problem of conformity (the conformity is true within a strip of given width). In other words, a geometric mesh, according to the previous definition, cannot be reduced to a single segment in these two cases. The corollary is that such a geometric mesh may be relatively large (in terms of the number of elements) in such cases.*

Remark 14.6 *On the contrary, a geometric mesh of a set of straight segments is this set itself and contains very few elements.*

Hints regarding implementation. The implementation of the method can be envisaged in various, more or less complex, forms. The main difficulty lies in calculating the length of the (portion of the) curve with respect to the metric field constructed. In fact, during the processing of a portion of curve, we can use a very fine (uniform) sampling to evaluate the desired length.

This simple method can however turns out to be expensive because capturing the geometry requires *a priori* a very small sampling step (although it may not be strictly required). Hence, other methods to calculate this length can be adopted. We suggest in this regard, the following method :

- we split the chord underlying the curve (we denote d the length of this chord) into n segments with n being relatively small (n depends on the presumed degree of the curve considered). We associate the curve parameter to a parameterization of this chord,
- on each interval, we *randomly* pick a point,
- for each such points, we evaluate the gap to the curve,
- the maximum of these gaps is the gap δ between the curve and the chord,
- if $\delta \leq \varepsilon\, d$, END,
- else, we retain the point of the curve corresponding to the point of the chord where the gap is maximal and we split the curve into two at this point. We replace the curve by the two corresponding chords whose lengths are respectively denoted d_1 and d_2 and we iterate the process on each segment of the subdivision.

Remark 14.7 *The previous test is equivalent to analyzing whether :*

$$\left|\frac{d}{d_1 + d_2} - 1\right| \leq \varepsilon\,.$$

As a result of this algorithm, we have a set of segments $P_i P_{i+1}$ approaching the curve. We still have to determine the length of the curve in the metric specified. For each segment constructed :

- we find the value of the metric at P_i and at P_{i+1},
- we use an interpolation between these two values to determine a metric everywhere,
- we compute the length of the segment in this interpolated metric (or using a quadrature, see the beginning of this chapter),
- if the length is bigger than a given threshold value (for instance one half), we use the metric of the midpoint to refine the length calculation and the process is iterated.

At completion of the procedure, we have segments whose lengths are less than or equal to the given threshold. It is then easy to deduce the desired unit mesh.

Other methods. We propose first a heuristic method.

- if P is a known mesh point, we find the next point P_{+1} that, if it were retained as such, would allow the segment PP_{+1} to be formed,
- then, considering P_{+1}, we travel the curve in the opposite direction to find the corresponding point P_{-1} and
  - if P_{-1} is "before" P, the point P_{+1} is judged correct, we form the segment PP_{+1} and we analyze the corresponding portion of curve to make sure that it stays close (for instance, by dichotomy),
  - else, we set $P_{+1} = P_{-1}$ and we form as segment PP_{+1} the segment PP_{-1} and we analyze the two corresponding portions of curve (by dichotomy).

Another method of the "dichotomy" or "divide and conquer" type (Chapter 2) can be envisaged. We identify the extrema of curvature and the singular points. We join two such consecutive points. The portion of curve limited by two such points is replaced by the corresponding segment. We analyze the gap between this segment and the curve. In the case of an intersection, we split the segment at the intersection points and we apply one of the previously described methods to each sub-segment. If the gap is satisfactory, we have the desired mesh. Otherwise, we apply one of the previous methods. The advantage of this approach is that we very rapidly obtain a situation where the portion of the underlying curve is very close to the segment and, moreover, is very regular.

Finally, another possibility consists in working on a discrete representation of the curve as will be seen below.

Remarks concerning mesh simplification. A point where the radius of curvature ρ is smaller than the given threshold ε (within a coefficient) is not retained as an extremum. This very simple intuitive idea allows us to suppress insignificant details and, hence, to construct simplified geometric meshes, within ε (*i.e.*, such a mesh follows the geometry within ε).

As a particular case and to complete the general discussion, we look at how to mesh a circle while controlling the gap between the circle and the mesh segments.

Meshing a circle. We consider a circle of radius ρ. We construct a chord of length $h = \alpha\,\rho$, the gap δ between this chord and the circle is given by the formula :

$$\delta = \rho\left(1 - \sqrt{1 - \frac{\alpha^2}{4}}\right)$$

and thus, imposing $\frac{\delta}{\rho} < \varepsilon$ is equivalent to fixing :

$$\alpha \leq 2\sqrt{\varepsilon\,(2-\varepsilon)} \qquad (14.12)$$

and, thus, an accuracy of ε (given) imposes that α is bounded in this way.

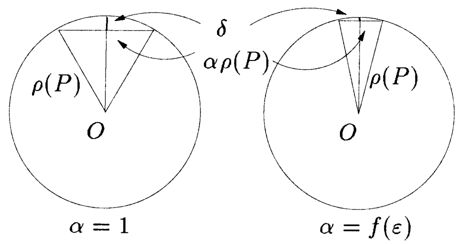

Figure 14.10: *Discretization of a circle into segments of length $\alpha\,\rho$. Left-hand side, we have $\alpha = 1$, right-hand side, α is smaller.*

If we consider segments of sizes $\alpha\,\rho$, so as to satisfy Relation (14.12), we obtain a discretization for which the gap to the circle is controlled, relative to ρ, by a threshold of value $1-\sqrt{1-\frac{\alpha^2}{4}}$, which again gives, as would be expected, according to Relation (14.5), a gap in $\frac{3\,\varepsilon}{4}\rho(s_0)$, hence of the same order (better in fact) than the given threshold ε.

As an example (see the exercise below), the choice $\varepsilon = 0.01$ leads to meshing the circle with 26 elements.

Exercise 14.3 *Verify that this number corresponds to a limited expansion of order 2 with the value $\varepsilon = 0.01$ (Hint : start from Relation (14.6), assume it is equal to $\varepsilon\,\Delta s$, set $\Delta s = \alpha\,\rho(s_0)$, deduce α and find the number of segments of length $\alpha\,\rho(s_0)$ required to cover the perimeter of the circle).*

Exercise 14.4 *Calculate the number of points required if a limited expansion of order 1 is used (Hint : suppose that $\frac{\Delta s^2}{2\,\rho(s_0)} = \varepsilon\,\Delta s$, deduce, if $\Delta s = \alpha\,\rho(s_0)$, a value of α then deduce the number of points required to cover the circle). Deduce that this analysis is not sufficient, too many elements are in fact required to obtain the desired accuracy. Restart by controlling the term in Δs^3 with respect to the term in Δs^2. What is the conclusion ? Notice that in fact the method deduced from these two approaches is unusable in practice. At the same time, notice that we find a justification of the previous method.*

14.2.2 Meshing algorithm without a metric map

As mentioned above, the geometric mesh of the curve is not necessarily suitable for a given calculation. For instance, a portion of curve reduced to a line segment has a geometric mesh identical to the segment, irrespective of its length. Hence, this element size can be very different from the other sizes of the neighboring elements and at least, a control of the size variation from element to element must be performed (see again in Chapter 10 the metric correction procedure).

14.2.3 Meshing algorithm with a metric map

The data is a curve and a field of isotropic or anisotropic metrics. This field, derived from an arbitrary calculation, does not necessarily follow the geometry of the curve. Hence, meshing the curve so as to conform to this field does not usually lead to a geometric approximation of the curve.

The idea then is to combine the given field with the intrinsic geometric field of the curve (*i.e.*, the field of the radii of curvature). The metric to consider is defined as the intersection of two given metrics (Chapter 10).

14.3 Curve meshing using a discrete definition

In this approach, the curve Γ is not known explicitly during the meshing process. It is actually supposed to be known only via a discrete definition (*i.e.*, a mesh reputed sufficiently small to reflect the true geometry). The construction of this mesh, so-called *geometric mesh*, is the responsibility of the user's favorite C.A.D. system.

Each segment of the geometric mesh is replaced by a curve C^1 or G^1 by using the available information (neighbors, singular points, tangents, normals, etc.). The union of these curves forms the *geometric support* that now serves as a geometric definition of the curve Γ. The mesh is then constructed based on this support so as to follow a given metric map if such a map is specified. We then encounter three cases :

- without map specification, we construct a geometric mesh that can be made suitable for numerical computations afterwards via a smoothing of the possible size shocks between two successive elements.
- is a map has been specified, we construct a mesh that follows this map, or
- in the same case, we construct a mesh that both follows this map and which is geometric.

The underlying idea is to get rid of the mathematical form of γ (the function representing Γ) and thus, to propose a method disconnected from the C.A.D. system that generated the curve. Actually, the sole data required is a mesh, possibly a very fine one, which any C.A.D. system is able to provide.

14.3.1 Construction of a definition from discrete data

We follow here the principle given in Chapter 12 where we chose to construct a curve of degree 3 based on each of the segments of the discrete data. If $\gamma(t)$ is the parameterization of this curve, we have :

$$\gamma(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \,, \tag{14.13}$$

and the question is to find the coefficients a_i as well as the interval of variation of the parameter t. The answer is given in Chapter 12 to which the reader is referred. The values of the a_i's and the degree of the curve obtained are given with respect to the known information.

The choice of a different type of curve is, needless to say, another possible solution.

14.3.2 Curve approximation by a polygonal segment

This construction consists in replacing the arcs of the curve by polygonal segments, by controlling the gap between each of these segments and the corresponding arc.

A simple solution consists in distributing regularly (that is uniformly) an arbitrary large number of points along the curve. This solution, which is relatively easy to implement, usually leads to a polygonal support that is much too fine (the number of elements being too great). Another solution consists in finding the points strictly required so that the above gap is bounded. The advantage is then to minimize the size of the resulting discretization while following the geometry accurately, especially in high curvature regions.

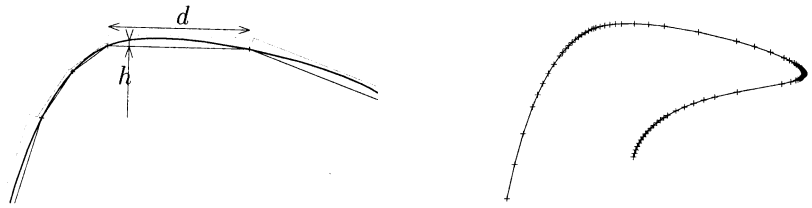

Figure 14.11: *Geometric approximation using a discrete definition formed by a polygonal segment respecting a given tolerance* ε. *Left-hand side,* $\varepsilon = 0.08$, *right-hand side,* $\varepsilon = 0.01$.

Construction of the polygonal segment, method 1. The careful reader has most probably noticed that the desired polygonal segment is nothing other than a geometric mesh according to the previous definition. Hence, any of the previously introduced methods may serve to provide the desired result.

Construction of the polygonal segment, method 2 (bandwidth method). We suggest however a rather different method (which we could have presented at the same time as the other methods).

We consider an arc of the curve and the underlying segment. We denote, Figure 14.11 (left-hand side) :

- h the distance from one point of the segment to the arc,
- d the length of this segment and

we give ε a tolerance threshold. The aim is to ensure $h < \varepsilon\, d$, at any point along the segment. To this end :

- we find the extremum or the extrema in h and we denote such a point E_i,
- if the relation $h < \varepsilon\, d$ is satisfied at this (these) point(s), then the segment is judged correct and is retained in the mesh,
- else, we subdivide this segment at these points and we iterate.

Simple in its principle, this method is nonetheless quite technical (in particular, to find the extrema). Figure 14.11 shows two results, depending on the value of ε considered.

14.3.3 Curve meshing from a discrete definition

The general scheme is always identical. We compute the length of the curve, then we construct the discretization based on this length. The length of the curve is calculated based on the polygonal segment previously constructed. This length is evaluated with respect to the given metric.

Mesh without field specification. Here again, we encounter the previously mentioned problem. The geometric mesh is not *a priori* necessarily correct in view of a numerical computation. The size variation between two adjacent elements may be large and, if this is the case, a control of this variation may be required.

Mesh with respect to a field specification. Once more, we reach the same conclusion as above. The metric to follow does not necessarily conform to the geometry. Depending on whether we have to respect the latter or not, we must perform the intersection between the specified metric and the geometric metric. Notice, in principle, that we do not know explicitly the latter metric as we have only the polygonal segment serving as a geometric support. Therefore, a way of determining the points of minimal radii of curvature and the singular points (discontinuities) is, for instance, to store this information during the construction of the polygonal segment. This being done, we find the usual situation. With a field specification, we mesh the curve by portions, each portion being delimited by two such consecutive points.

14.3.4 Examples of planar curve meshes

We propose, in Figures 14.13, two examples of meshes of a same curve obtained for the specification of the discrete metric field described in Figure 14.12. Left-hand side, the metric specified has been interpolated linearly, right-hand side, the interpolation is geometric. In these figures, we give the interpolated metrics at the various vertices of the resulting mesh. This allows us to verify the coherence of the result with the type of interpolation used.

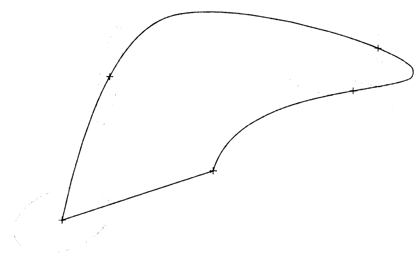

Figure 14.12: *Discrete anisotropic size specification. The information about directions and sizes is known only at specific points.*

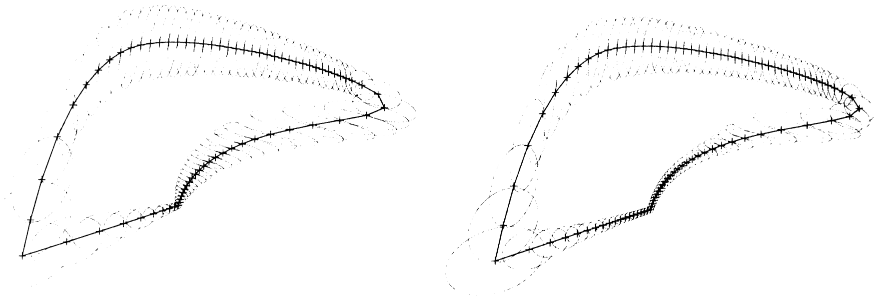

Figure 14.13: *Mesh resulting from a linear interpolation (left-hand side) and a geometric interpolation (right-hand side).*

14.4 Re-meshing algorithm

Curve remeshing is a rather different problem although similar to the previously discussed problem. The applications are manifold. Let us mention, in particular, the possibility of simplifying the mesh, optimizing (for a given criterion) or also adapting the mesh, among the possible applications.

The data is then a mesh and a goal to achieve. The aim is thus to modify the mesh so as to match (or to satisfy) this requirement.

The available mesh modification tools are very simple (see Chapter 18, for instance). We can in fact :

- add points,
- suppress points,
- move points.

The idea is then to use these tools in order to remesh the underlying curve to conform to the fixed objective while preserving, in a certain way, the geometry of this curve. The goal can be expressed by assuming known a metric $\mathcal{M}$ which is continuous or discrete (mesh optimization or adaptation) or a given accuracy (mesh simplification). The immediate question is related to the way in which the geometry of the underlying curve is defined. In fact, the aim is to retrieve (or even to invent) this geometry from the sole data available (the initial mesh).

Discovering the geometry can be achieved in various ways. For instance :

- we identify the presumed singular points (for example, by looking, with respect to a threshold, at the angle variations from element to element),
- we split the initial mesh into pieces where each piece is limited by two such (consecutive) singular points,
- we construct a *geometric support*, for instance a cubic on each element (of each portion). To this end, we can follow the previously described method,
- this geometric support allows us to directly or indirectly (via a piecewise linear approximation with a polygonal segment as already seen) find the geometric features of the curve (radii of curvature, tangents, normals) or, at least, approached values of these quantities.

We then have a definition of the geometry that can be used to drive the operations required to achieve the fixed goal. We then find a situation similar to that related to the meshing of a curve when a metric specification is given or to an objective formulated in a different way.

Remark 14.8 *Numerous and various difficulties can be expected. Let us mention in particular, the problem related to the definition, for a given portion of curve, of a reasonable normal (a tangent) at singular points, assuming the latter have been correctly identified.*

14.5 Curves in $\mathbb{R}^3$

The construction of a mesh for a curve of $\mathbb{R}^3$ may have various forms, depending on whether the curve considered is part of a known surface or not.

14.5.1 Dangling curves

We consider here the curves of $\mathbb{R}^3$ that are not traced on a known surface, that can be used to extract the required information.

Remark 14.9 *Initially, notice that, if we restrict ourselves to two dimensions, the following gives the results previously established.*

We reuse the same reasoning as for the planar curves. The local analysis of the curve corresponds, in the neighborhood of s_0, to the limited expansion :

$$\gamma(s) = \gamma(s_0) + \Delta s\gamma'(s_0) + \frac{\Delta s^2}{2}\gamma''(s_0) + \frac{\Delta s^3}{6}\gamma'''(s_0) + \dots,$$

with $\Delta s = s - s_0$ sufficiently small to ensure the validity of the development. We have still $\gamma'(s) = \vec{\tau}(s)$ and $\gamma''(s) = \dfrac{\vec{\nu}(s)}{\rho(s)}$ while now :

$$\gamma'''(s) = -\frac{1}{\rho(s)^2}\left(\rho'(s)\,\vec{\nu}(s) + \vec{\tau}(s) + \frac{\rho(s)}{R_T(s)}\,\vec{b}(s)\right),$$

where $\vec{b}(s)$ is the binormal vector at s (Chapter 11) and $R_T(s)$ is the radius of torsion of the curve at s. Hence, we have :

$$\gamma(s) = \gamma(s_0) + \Delta s\,\vec{\tau} + \frac{\Delta s^2}{2\,\rho(s_0)}\,\vec{\nu} - \frac{\Delta s^3}{6\,\rho(s_0)^2}(\rho'(s_0)\,\vec{\nu} + \vec{\tau} + \frac{\rho(s_0)}{R_T(s_0)}\,\vec{b}) + \dots, \quad (14.14)$$

where $\vec{\tau} = \vec{\tau}(s_0)$, $\vec{\nu} = \vec{\nu}(s_0)$ and $\vec{b} = \vec{b}(s_0)$. The behavior of the curve if that of the parabola (the first three terms of the development) if :

$$\frac{\Delta s^3}{6\,\rho(s_0)^2}\left(\rho'(s_0)\,\vec{\nu} + \vec{\tau} + \frac{\rho(s_0)}{R_T(s_0)}\,\vec{b}\right), \quad (14.15)$$

which involves the variation of ρ (*i.e.*, the value ρ') and the radius of torsion R_T, is small before the previous terms. If the torsion is null (the radius of torsion is infinite) the problem is then a planar problem and we return to the discussion regarding planar curves. On the other hand, this implies, for a given ε, a condition similar to Condition (14.7), which can be written as :

$$\alpha \approx \frac{\sqrt{6\,\varepsilon}}{\sqrt[4]{1 + \rho'(s_0)^2 + \left(\dfrac{\rho(s_0)}{R_T(s_0)}\right)^2}}. \quad (14.16)$$

The gap to the parabola is then of the following form :

$$\delta \approx \frac{\varepsilon\sqrt{6\,\varepsilon}}{\sqrt[4]{1 + \rho'(s_0)^2 + \left(\dfrac{\rho(s_0)}{T(s_0)}\right)^2}}\,\rho(s_0). \quad (14.17)$$