Algorithm 3.3

Input:
```
τ, tolerance for subdivision control.
S(u,v), an offset surface, possibly self-intersecting.
```

Output:
```
L, a piecewise linear representation of the self
             intersection curves.
```

Algorithm:
```
Q ⇐ S(u,v), a priority queue holding sorted data in z
             according to minimum z of elements.
P ⇐ ∅, a set of all active polygons.
L ⇐ ∅.
While ( Q ≠ ∅ )
begin
  Obj ⇐ first(Q).
  z ⇐ minimumZ(Obj).
  if ( isSurface( Obj ) )
    if ( isFlat( Obj, τ ) )
      Convert to polygons, and for each polygon P_i Do
        L ⇐ L ∪ InterActiveList(P_i, z, P, Q).
    else
      Subdivide into two subsurfaces and insert both to Q.
  else /* Its a polygon */
    L ⇐ L ∪ InterActiveList(Obj, z, P, Q).
end
```

in the parameter space of the surface.

Removal of self-intersections in surface offsets, is not totally solved and should be further investigated. A complete study of the complex topology of the self-intersection curves may provide some leads.

Algorithm 3.3 continued

InterActiveList$(P, z, \mathcal{P}, \mathcal{Q})$
```
begin
  M ⇐ ∅, holding all self-intersections with polygon P.
  if ( minimumZ( P ) > z )
    Insert P to Q.
  else
  begin
    For each polygon P_i in P do
    begin
      if ( maximumZ( P_i ) < z )
        remove( P_i, P ).
      else
        M ⇐ M ∪ IntersectPolyPoly(P, P_i).
    end
    Insert P to P.
  end
  return M.
end
```

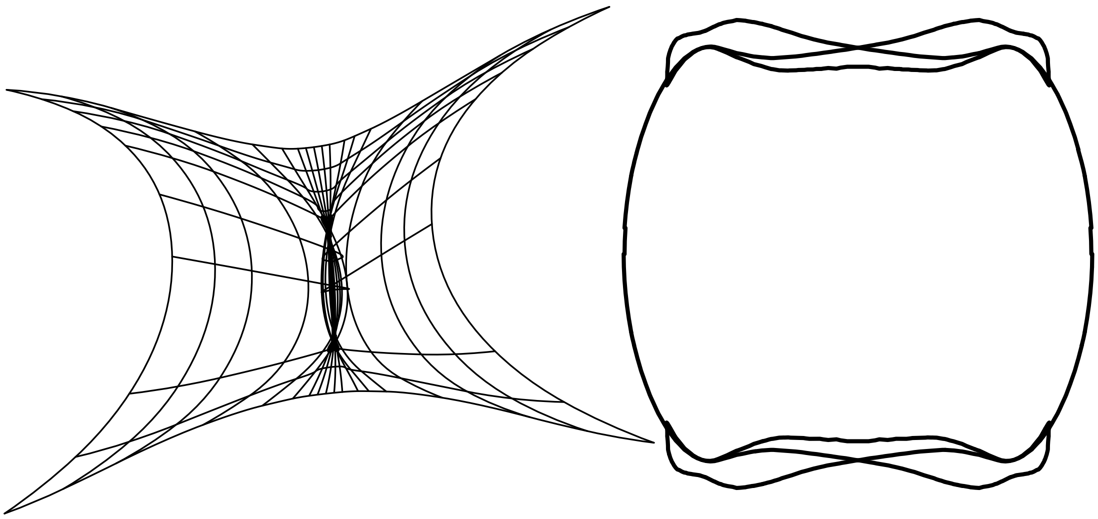

Figure 3.15. Offset surface self-intersection can be topologically complex.

CHAPTER 4

SECOND ORDER SURFACE ANALYSIS

It is our purpose to give a presentation of geometry, as it stands today, in its visual, intuitive aspects. With the aid of visual imagination we can illuminate the manifold facts and problems of geometry, and beyond that, it is possible in many cases to depict the geometric outline of the methods of investigation and proof, without necessarily entering into the details connected with the strict definions of concepts and with the actual calculations.

D. Hilbert, in "Geometry and the Imagination," 1932.

A critical characteristic for many applications in computer graphics and in CAD is the shape of the model's bounding surfaces. Second order surface analysis can be used to understand curvature characteristics, and thus shape, and to improve the implementation, efficiency and effectiveness of manufacturing and analysis processes. Fundamental operations, such as adaptive subdivision and refinement, use shape information to decide where and how many knots to add. Algorithms for the creation of tool paths for NC (Numerically Controlled) code generation for freeform surfaces are usually based on ball end cutters with their spherical centers following an (approximate) offset surface of the original surface. Flat end cutters can remove material faster and have a better finish; however, flat end cutters can be used only with 5 axis milling in convex regions (see Figure 4.1).

Definition 4.1 *A* surface trichotomy *is a partition of a surface into three types of regions: convex, concave and saddle shapes (Figure 4.1).*

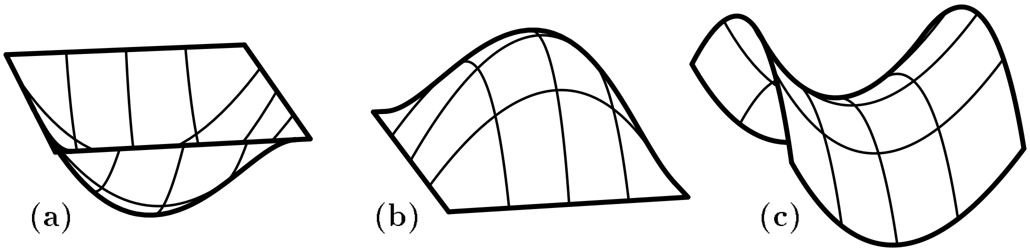

Figure 4.1. Mainly concave (a), convex (b), and saddle (c) regions.

The ability to trichotomize sculptured surfaces into convex, concave or saddle regions (Figure 4.1) is thus essential to the use of flat end cutters in milling freeform surfaces. Also, regions with small curvature can be accurately milled faster with larger ball end cutters. Tool changes should be minimized because the are time consuming operations. Such minimization can be achieved by subdividing the surface into regions with different curvature bounds, each of which can be milled using tools appropriate to that region.

Methods in use do not support the separation of original surfaces into trimmed surfaces each of which with only one of the three characteristics throughout. That is, each trimmed surface is either convex everywhere, concave everywhere, or saddle everywhere. Second order surface properties are usually estimated locally by numerically evaluating them at a grid of points or, in manufacturing, at a finite set of sampled points along a planned milling tool path. Research into the computation of curvature has been done in the context of offset operator approximations with cubic B-spline curves [66] and bicubic patches [29].

There have been attempts [3, 4, 20, 33] to understand and compute second order surface properties as well as twist by evaluation on a predefined grid. The methods use the Gaussian curvature $K(u,v) = \kappa_n^1(u,v)\kappa_n^2(u,v)$ and mean curvature $H(u,v) = \frac{\kappa_n^1(u,v)+\kappa_n^2(u,v)}{2}$, where $\kappa_n^1(u,v)$ and $\kappa_n^2(u,v)$ are the *principal curvatures* at the parameter value (u,v), in an attempt to provide a bound on the surface angularity. However, if the surface is a saddle at (u,v), then κ_n^1 and κ_n^2 have different signs so the magnitude of H is not a useful measure of such a bound. In the extreme condition when the surface is minimal [21], $H \equiv 0$ regardless of the surface angularity. The magnitude of K can also be ineffective. Even if κ_n^1 is large, K may be small because κ_n^2 is small. Therefore, neither K nor H by itself can provide sufficient shape information for subdivision and/or efficient NC applications. This problem has been recognized by some of the authors cited above. These curvature estimation techniques are *local*, because they make use of local surface information only. More surface information might improve an algorithm or change a decision. Local information is inferior to global information in complex settings. Symbolic techniques can be used to help make decisions based upon the entire aspect of a surface rather than a limited number of local samples.

In this Chapter, a hybrid approach using both symbolic and numeric operations for computing curvature properties is developed. We use *property surfaces* (see definition 1.1) whose definitions are derived from different attributes of the original surface, as auxiliary surfaces to help analyze the original surface.

Section 4.1 briefly develops the differential geometry used in the analysis. In section 4.2, we compute second order properties, and use visualization to better understand the shape of a given surface.

4.1 Differential Geometry

Surface curvature is well understood mathematically and the theory behind it is developed in most introductory differential geometry books [21, 48, 63]. The set of analysis equations that are based on the second fundamental form are used extensively in locally evaluating surface curvature. Because these equations are crucial to our discussion, they are briefly stated here.

Let $F(u, v)$ be a $C^{(2)}$ regular parametric surface. Let the *unnormalized normal* to a surface $F(u, v)$, $\hat{n}(u, v)$, be defined as

$$\hat{n}(u,v) = \frac{\partial F}{\partial u} \times \frac{\partial F}{\partial v}, \tag{4.1}$$

and define the surface unit normal, $n(u,v)$, to be

$$n(u,v) = \frac{\frac{\partial F}{\partial u} \times \frac{\partial F}{\partial v}}{\|\frac{\partial F}{\partial u} \times \frac{\partial F}{\partial v}\|}. \tag{4.2}$$

Because $F(u,v)$ is regular, $\|\hat{n}(u,v)\| \neq 0$ and $n(u,v)$ is well defined.

Let $C(t) = F(u(t), v(t))$ be a regular curve on F, that is $\left\|\frac{dC(t)}{dt}\right\| \neq 0$. The rate of change of the arc length of C with respect to its parameter, t, is $\frac{ds}{dt} = \left\|\frac{dC(t)}{dt}\right\|$ where s is arc length. Because $\frac{dC(t)}{dt} = \left(\frac{\partial F}{\partial u}\frac{du}{dt} + \frac{\partial F}{\partial v}\frac{dv}{dt}\right)$, one can show [32, 48, 63] that

$$\left(\frac{ds}{dt}\right)^2 = \begin{bmatrix} \frac{du}{dt} & \frac{dv}{dt} \end{bmatrix} G \begin{bmatrix} \frac{du}{dt} & \frac{dv}{dt} \end{bmatrix}^T = I\left(\frac{du}{dt}, \frac{dv}{dt}\right).$$

I is known as the first fundamental form, with matrix G equal to:

$$G = (g_{ij}) = \begin{bmatrix} \left\langle \frac{\partial F}{\partial u}, \frac{\partial F}{\partial u} \right\rangle & \left\langle \frac{\partial F}{\partial u}, \frac{\partial F}{\partial v} \right\rangle \\ \left\langle \frac{\partial F}{\partial v}, \frac{\partial F}{\partial u} \right\rangle & \left\langle \frac{\partial F}{\partial v}, \frac{\partial F}{\partial v} \right\rangle \end{bmatrix}. \tag{4.3}$$

By considering all such curves, $C(t)$, through a point (u,v) and differentiating twice, one can extract second order properties of the surface F at (u,v). The second order derivatives of $C(t)$ contain terms with $\frac{\partial F}{\partial u}$ and $\frac{\partial F}{\partial v}$ as factors. However, the inner product of these terms with n is always zero because the partials are in the tangent plane of $F(u,v)$. Therefore, $\left\langle n(u,v), \frac{d^2C(t)}{dt^2}\right\rangle$, the component of $\frac{d^2C(t)}{dt^2}$ pointing in the direction perpendicular to the surface is composed of second order derivatives only.

$$\begin{aligned} & \left\langle n(u,v), \frac{d^2C(t)}{dt^2} \right\rangle \\ = & \left\langle n(u,v), \frac{\partial^2 F}{\partial u^2} \right\rangle \left(\frac{du}{dt}\right)^2 + 2\left\langle n(u,v), \frac{\partial^2 F}{\partial u \partial v} \right\rangle \frac{du}{dt}\frac{dv}{dt} + \left\langle n(u,v), \frac{\partial^2 F}{\partial v^2} \right\rangle \left(\frac{dv}{dt}\right)^2 \\ = & \begin{bmatrix} \frac{du}{dt} & \frac{dv}{dt} \end{bmatrix} L \begin{bmatrix} \frac{du}{dt} & \frac{dv}{dt} \end{bmatrix}^T \end{aligned}$$

$$= \quad II\left(\frac{du}{dt},\ \frac{dv}{dt}\right). \tag{4.4}$$

II is known as the second fundamental form, with matrix L equal to:

$$L = (l_{ij}) = \begin{bmatrix} \left\langle n, \frac{\partial^2 F}{\partial u^2} \right\rangle & \left\langle n, \frac{\partial^2 F}{\partial u \partial v} \right\rangle \\ \\ \left\langle n, \frac{\partial^2 F}{\partial u \partial v} \right\rangle & \left\langle n, \frac{\partial^2 F}{\partial v^2} \right\rangle \end{bmatrix}. \tag{4.5}$$

Let $\hat{l}_{ij}$ denote the inner product with the unnormalized normal $\hat{n}(u, v)$. For example, $\hat{l}_{11} = \left\langle \hat{n}, \frac{\partial^2 F}{\partial u^2} \right\rangle$.

The normal curvature on the surface $F(u, v)$ in some tangent direction Δ, where $\Delta = \left\langle \delta, \left(\frac{dF}{du},\ \frac{dF}{dv}\right) \right\rangle$, and $\delta = \left(\frac{du}{dt},\ \frac{dv}{dt}\right)$, is defined [21, 32, 48, 63] as:

$$\kappa_n = \frac{II(\frac{du}{dt}, \frac{dv}{dt})}{I(\frac{du}{dt}, \frac{dv}{dt})} = \frac{\delta L \delta^T}{\delta G \delta^T}. \tag{4.6}$$

The normal curvature depends on the surface tangent direction Δ, and is equal to the curvature of the osculating circle to the intersection curve between $F(u, v)$ and the plane through $n(u, v)$ and Δ at (u, v) (Figure 4.2). The extremal values of the normal curvature serve as bounds on the components of curvature not in the tangent plane.

The normal curvature is an *intrinsic property* [48, 63] of the surface. By differentiating (4.6) with respect to δ, the problem of finding extrema of κ_n is transformed [21, 32, 48, 63] into the problem of solving for the roots of

$$|G|\,\kappa_n^2 + (g_{11}l_{22} + l_{11}g_{22} - 2g_{12}l_{12})\kappa_n + |L| = a\kappa_n^2 + b\kappa_n + c = 0, \tag{4.7}$$

where $|G|$ and $|L|$ denotes the determinants of G and L, respectively.

The Gaussian curvature is a scalar value and is defined as the product of the two roots of (4.7), κ_n^1 and κ_n^2,

$$K = \kappa_n^1 \kappa_n^2 = \frac{|L|}{|G|}. \tag{4.8}$$

The mean curvature is defined as their arithmetic average,

$$H = \frac{\kappa_n^1 + \kappa_n^2}{2} = -\frac{(g_{11}l_{22} + l_{11}g_{22} - 2g_{12}l_{12})}{2\,|G|}. \tag{4.9}$$

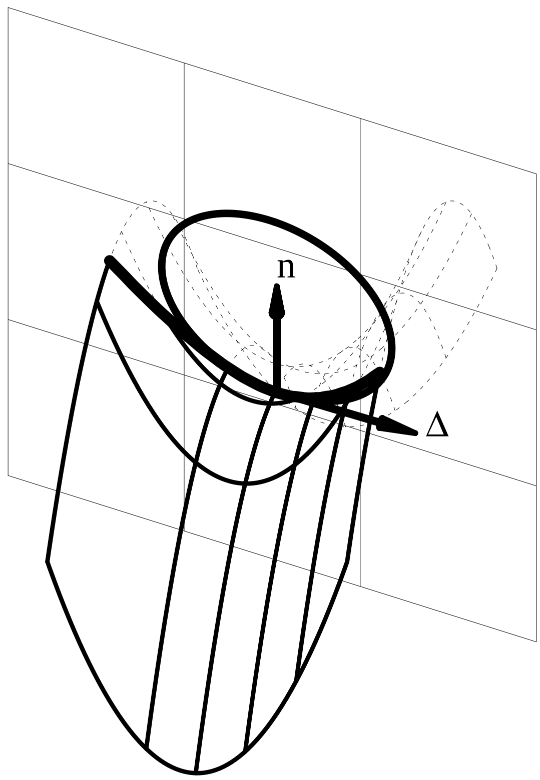

Figure 4.2. Normal curvature κ_n (circle) of $F(u,v)$ at (u,v) in direction Δ.

4.2 The approach

The tools defined in Chapter 2 are used symbolically to compute the second order properties of a given surface as described in Section 4.1. NURBs property surfaces are derived whenever possible so that the method can take advantage of the computational characteristics of NURBs.

4.2.1 Surface Trichotomy

Use of the curvature trichotomy of a surface can result in a more optimal freeform surface milling process. Only convex regions (see Figure 4.1) are millable using flat end cutters and 5 axis milling. Flat end cutters, as opposed to ball end cutters, can mill faster and remove more material per time unit. Furthermore, the surface finish of flat end cutters is usually better. Using the trichotomy operator, convex regions within surfaces can be detected and milled in more efficient way and with a better finish.

The determinant of L, $|L|$, in (4.7) is the key to this second order surface analysis. If $|L| = 0$, one of the normal curvature extrema κ_n^i must be zero. Assuming the surface is curvature continuous, adjacent regions for which κ_n^i has a different sign must be separated by a curve, C_s, for which $|L| = 0$, that is, one of the $\kappa_n^i = 0$. Furthermore, if $|L| > 0$ at some point p on the surface F, the surface is either convex or concave at p, while if $|L| < 0$ the surface locally is a saddle. In order to compute a property surface representing $|L|$ using (4.5), it is necessary to find a square root to compute $n(u,v)$, which cannot be represented, in general, as a polynomial or as a piecewise rational. However, by reordering the operations to use the unnormalized surface normal $\hat{n}(u,v)$ and noting $n(u,v)$ appears twice as a factor in each term of $|L|$, $|L|$ can be represented exactly as a rational function and with no square roots,

$$|L| = \frac{\hat{l}_{11}\hat{l}_{22} - \hat{l}_{21}\hat{l}_{12}}{\|\hat{n}\|^2}. \tag{4.10}$$

This equation is representable as a NURBs using only operations from Chapter 2. $\hat{n}$ is a cross product of two surface partials $\frac{\partial F}{\partial u}$ and $\frac{\partial F}{\partial v}$. The components of L, $\hat{l}_{ij}$, are inner products of $\hat{n}$ with second order partials of F. Because only the zero set is of interest, and F is assumed to be a regular surface, it is necessary to examine only the numerator of (4.10). Once the zero set of $|L|$ has been computed, trimmed surfaces are created, each of which is completely convex, concave or saddle. The sign of $|L|$ at a single point on each trimmed surface is then used to classify the saddle regions while convex and concave regions are distinguished from each other by simply evaluating the sign of $\hat{l}_{11}$, for example, at that single point. Whereas the saddle region is an intrinsic surface characteristic, the convex/concave classification is parameterization dependent. Flipping the u or v (but not both) surface parameterization direction will flip the normal direction $n(u, v)$ and therefore the sign of $\hat{l}_{11}$.

Figures 4.3 through 4.7 show some examples. Figure 4.3 is a biquadratic B-spline surface with three internal knots in each direction (patches of a B-spline surface are counted as how many Bézier patches would result from subdividing the NURBs surface at each interior knot, so this surface yields 16 polynomial patches), while Figure 4.4 is a single biquadratic patch. The bicubic surfaces in Figures 4.5 and 4.6 have two internal knots in each direction, yielding 9 polynomial patches. Figure 4.7 top is a bicubic NURBs surface with a single internal knot in each direction, yielding four Bézier patches. All figures have been colored consistently, with yellow marking the saddle regions, red representing a convex region and green representing a concave region.

The biquadratic surface of Figure 4.3 is not C^2 along each internal knot, and the surface trichotomy is isoparametric along the internal knots lines.

However, in general, this behavior should not be expected, or even anticipated, for biquadratic surfaces, because even a single biquadratic patch may contain both convex and saddle regions simultaneously as shown in Figure 4.4.

The surface in Figure 4.5 uses the same control mesh as the one in Figure 4.3 but is bicubic. Both surfaces in Figure 4.3 and Figure 4.5 use appropriate uniform open end condition knot vectors. A comparison of these two Figures graphically demonstrates the influence of the *order* of the tensor product spline surface on the shape, as shown by comparing the shapes and locations of the convex and concave regions. This phenomenon is somewhat counterintuitive to the common belief that two NURBs surfaces with the same mesh but different order are very similar, except that the one with higher order is a smoother version. The curvature characteristics have actually been changed. Figure 4.3 has one concave region, one convex region and two flat regions, all of which have isoparametric boundaries.

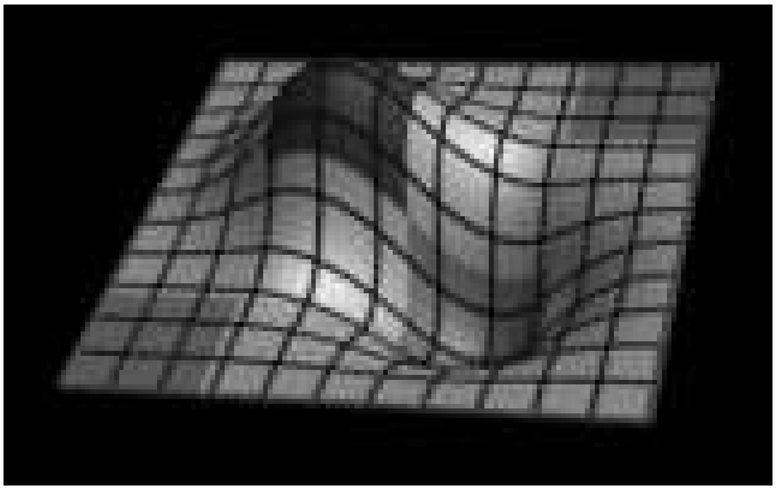

Figure 4.3. Biquadratic surface trichotomy with 16 polynomial patches.

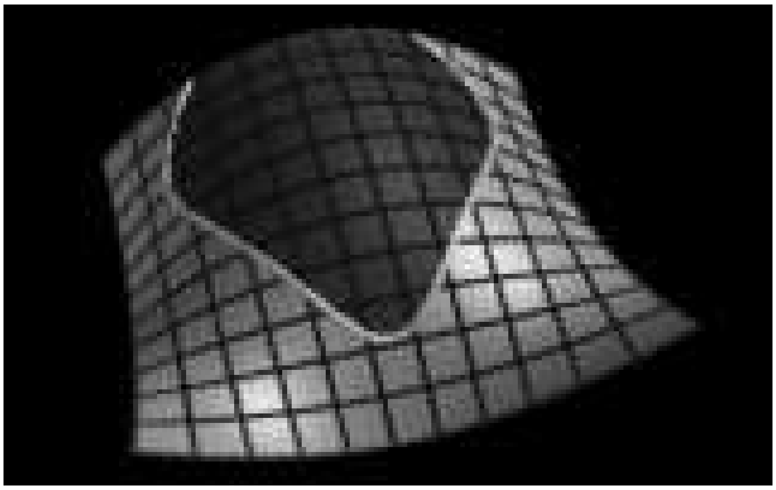

Figure 4.4. Biquadratic polynomial trichotomy.

Figure 4.5, however, has only one concave region and one convex region. The union of the two regions has a figure eight boundary, where convex and concave change at a single point. The curved boundaries of those regions are different from the straight line boundaries in Figure 4.3.

Figure 4.6 shows that the combination of symbolic computation (of $|L|$ as a property surface) with numeric analysis (contouring the property surface) can detect widely separated and isolated regions. In addition, it demonstrates the robustness of this methodology by accurately detecting two very shallow concave

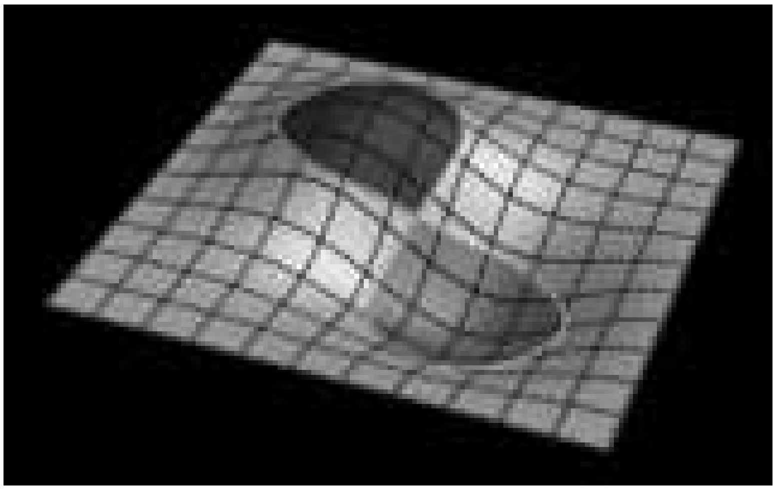

Figure 4.5. Bicubic surface trichotomy: same control mesh as Figure 4.3.

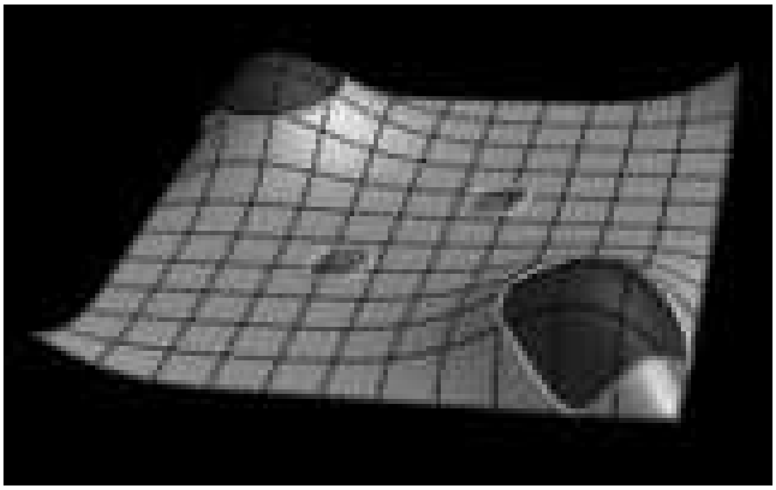

Figure 4.6. Bicubic with isolated convex and concave regions in a saddle region.

regions in the middle of the surface. In Figures 4.5 and 4.7, another ill conditioned case is shown in which several convex and concave regions meet at a single point. Because trimmed surfaces are formed, it was necessary that the boundaries be completely and correctly defined. The points where the three regions meet are correctly detected and determined and the topology of the regions is correctly maintained, which also demonstrates another type of robustness.

To provide a better sense of the process, the bottom of Figure 4.7 also shows the scalar property surface of the determinant of the second fundamental form, $|L|$,

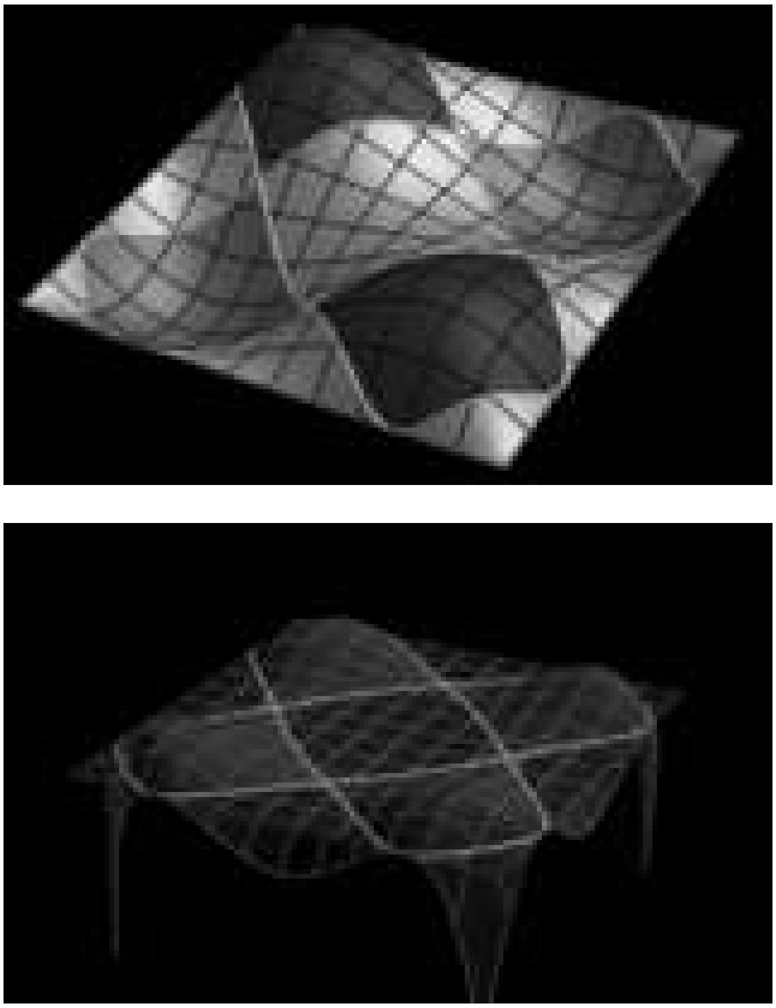

Figure 4.7. Bicubic surface with convex and concave regions meet at a single point (top). The surface second fundamental form property surface and its zero set (bottom).

with its zero set, as a function of u and v.

Figure 4.8 demonstrates this method on a more realistic object. The Utah teapot trichotomy degenerated into a dichotomy because no concave regions exist in the teapot model.

It is interesting to note that a sufficient condition for a surface to be developable [32] is that its Gaussian curvature is always zero: $K(u,v) \equiv 0$. Because $K(u,v) = \frac{|L|}{|G|}$, this condition is equivalent to the condition that $|L| \equiv 0$ for regular surfaces were $|G| \neq 0$. Hereafter, a simple practical test that can answer whether

Figure 4.8. Teapot trichotomy degenerates into a ditochomy (no concave regions).

a surface is developable or not can be derived by symbolically computing $|L|$ and comparing all its coefficients to zero. Figure 4.9 show two developable NURBs surfaces. The top one is ruled surface along an isoparametric direction while the bottom one was bent along *nonisoparametric* direction.

4.2.2 Bounding the Curvature

The extrema of the surface curvature are important for analyzing the curvature of a given surface. Normal curvature extrema occur in the principal directions [32, 48, 63], but the direct application of quadratic equation solution for equation (4.7) would require finding a square root. However, because the surface has been subdivided into convex, concave, and saddle regions, each region carries the following property:

- If the region has a saddle shape, then one of the principal curvatures, κ_n^1, is positive while the other, κ_n^2, is negative.
- If the region is convex both principal curvatures are negative.
- If the region is concave both principal curvatures are positive.

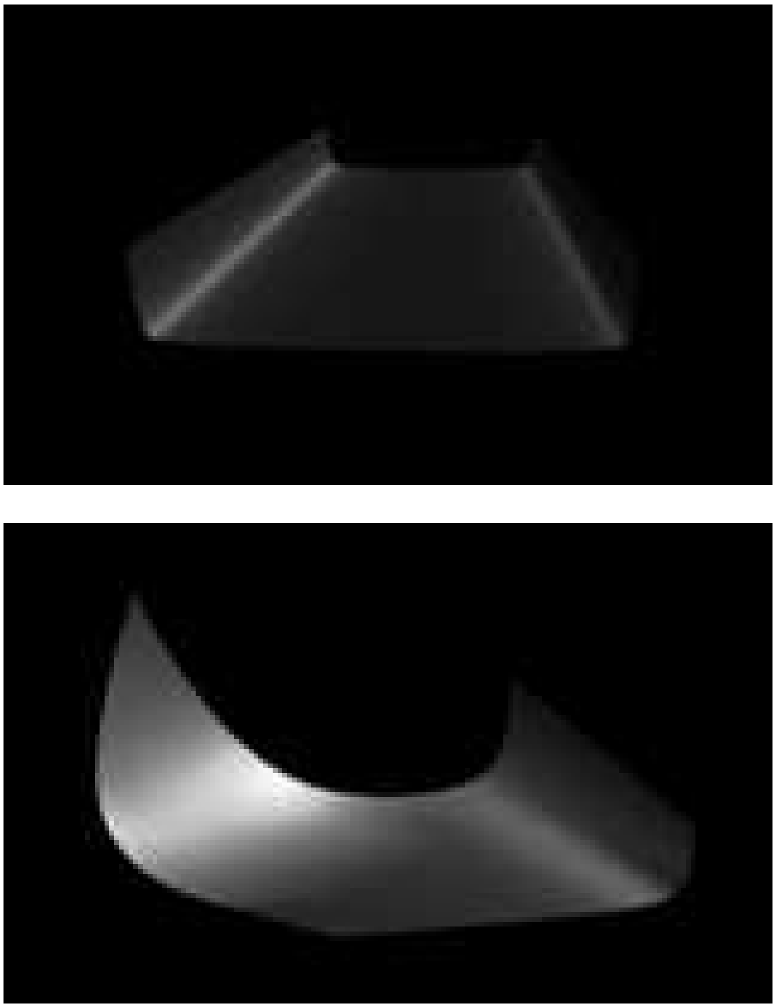

Figure 4.9. Two ruled surface examples.

Using quadratic equation properties for equation (4.7), it can easily be shown that:

$$\begin{aligned}
\psi &= (2H)^2 \\
&= \left(\kappa_n^1 + \kappa_n^2\right)^2 \\
&= \left(-\frac{b}{a}\right)^2 \\
&= \left(-\frac{g_{11}l_{22} + l_{11}g_{22} - 2g_{12}l_{12}}{|G|}\right)^2 \\
&= \frac{\left(g_{11}\hat{l}_{22} + \hat{l}_{11}g_{22} - 2g_{12}\hat{l}_{12}\right)^2}{|G|^2 \, \|\hat{n}\|^2} \qquad (4.11)
\end{aligned}$$

and

$$
\begin{aligned}
\phi &= (\kappa_n^1 - \kappa_n^2)^2 = \frac{b^2 - 4ac}{a^2} \\
&= \frac{(g_{11}l_{22} + l_{11}g_{22} - 2g_{12}l_{12})^2 - 4\,|G|\,|L|}{|G|^2} \\
&= \frac{(g_{11}\hat{l}_{22} + \hat{l}_{11}g_{22} - 2g_{12}\hat{l}_{12})^2 - 4\,|G|\,\left|\hat{L}\right|}{|G|^2\,\|\hat{n}\|^2}.
\end{aligned} \tag{4.12}
$$

Both (4.11) and (4.12) can be represented without square roots and are therefore representable as NURBs using the model and tools defined in Chapter 2.

By using the property surface $\psi(u,v) = (\kappa_n^1(u,v) + \kappa_n^2(u,v))^2$ as a curvature estimate for the convex and concave regions, the computed curvature will be at most twice as large as the real normal curvature in the case where both $\kappa_n^1(u,v)$ and $\kappa_n^2(u,v)$ are equal. Similarly by using $\phi = (\kappa_n^1(u,v) - \kappa_n^2(u,v))^2$ as the curvature estimate for saddle regions one can obtain similar bounds.

$\psi(u,v)$ and $\phi(u,v)$ can be used as curvature estimates for the appropriate trimmed regions and can be contoured to isolate regions with curvature larger than some allowable threshold. Furthermore, one can use $\psi(u,v)$ and $\phi(u,v)$ as pseudo color values to render the input surface $F(u,v)$ according to its curvature and provide visual feedback on which regions are highly curved. In other words, make the color of $F(u,v)$ at the parameter value (u,v) depend on the value of $\psi(u,v)$ in convex and concave regions, and on the value of $\phi(u,v)$ in saddle regions. Using this technique, one can enhance the display of regions with high curvature, low curvature, or within certain bands of curvatures. Figures 4.10 through 4.12 demonstrate this. In Figure 4.10, the surface has been first subdivided into a saddle region (yellow) and a convex region (red). $\psi(u,v)$ has been used as the pseudo color in the convex region of the surface whereas $\phi(u,v)$ has been used for the same purpose in the saddle region, to render the image in Figure 4.12. Figure 4.11 shows $\psi(u,v)$ and $\phi(u,v)$. Not surprisingly, $\psi(u,v)$ is wider in the highly curved convex region because the two principal curvatures cancel each other in $\phi(u,v)$.

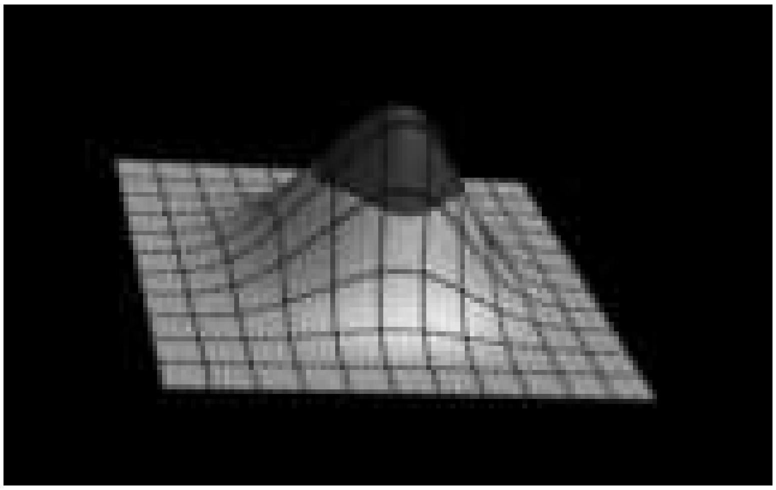

Figure 4.10. Surface dichotomy - saddle and convex regions.

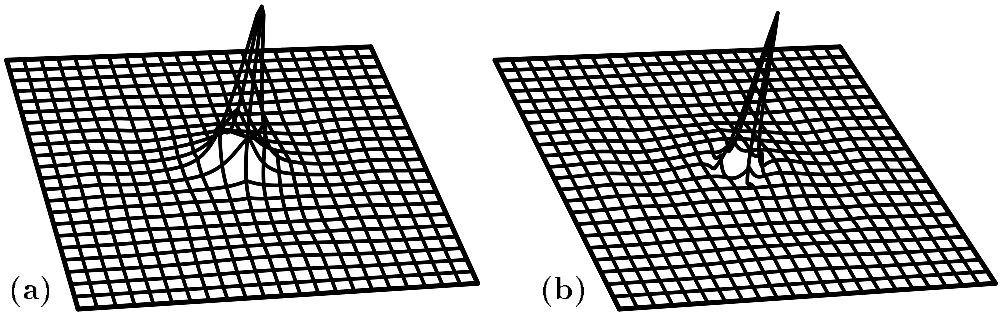

Figure 4.11. $\psi(u,v)$ (a), $\phi(u,v)$ (b), for the surface in Figure 4.10.

A different approach can be used to achieve a better bound. By expanding ϕ,

$$\begin{aligned} \phi &= (\kappa_n^1 - \kappa_n^2)^2 \\ &= \left(\kappa_n^1\right)^2 - 2\kappa_n^1\kappa_n^2 + \left(\kappa_n^2\right)^2. \end{aligned} \tag{4.13}$$

Or

$$\begin{aligned} \xi &= (\kappa_n^1)^2 + (\kappa_n^2)^2 \\ &= \phi + 2\kappa_n^1\kappa_n^2 \\ &= \phi + 2K \\ &= \phi + 2\frac{|L|}{|G|} \end{aligned}$$

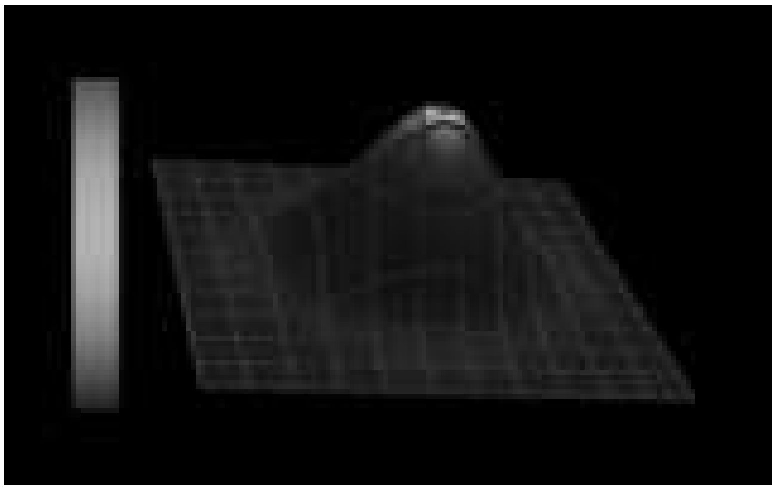

Figure 4.12. Curvature estimate using surface dichotomy, for surface in Figure 4.10.

$$
\begin{aligned}
&= \frac{(g_{11}\hat{l}_{22} + \hat{l}_{11}g_{22} - 2g_{12}\hat{l}_{12})^2 - 4\,|G|\,\left|\hat{L}\right|}{|G|^2\,\|\hat{n}\|^2} + 2\frac{|L|}{|G|} \\
&= \frac{(g_{11}\hat{l}_{22} + \hat{l}_{11}g_{22} - 2g_{12}\hat{l}_{12})^2 - 4\,|G|\,\left|\hat{L}\right|}{|G|^2\,\|\hat{n}\|^2} + 2\frac{\left|\hat{L}\right|}{|G|\,\|\hat{n}\|^2} \\
&= \frac{(g_{11}\hat{l}_{22} + \hat{l}_{11}g_{22} - 2g_{12}\hat{l}_{12})^2 - 2\,|G|\,\left|\hat{L}\right|}{|G|^2\,\|\hat{n}\|^2}. \qquad (4.14)
\end{aligned}
$$

$+\sqrt{\xi}$ is bounded to be at most $\sqrt{2}$ greater than the larger magnitude of the principal curvatures. This worst case occurs when the two principal directions have the same magnitudes. Furthermore, ξ can be represented using the tools described in Chapter 2. Figure 4.13 demonstrates this approach applied to the Utah teapot model. The use of ξ may help to isolate regions with low curvature, which can be milled using larger ball end tools in a more optimal way. Figure 4.14 shows such a surface subdivided in such regions. The curvature bound surface, $\xi(u, v)$, (Figure 4.15) of the surface in Figure 4.14 is being contoured and regions with different curvature bounds are formed. It is clear from Figure 4.14 that the blue regions can be milled using a very large ball end cutter, the green regions with a medium size cutter and only the yellow and red regions, which are less than 5% of the whole surface area, should be milled with a small size tool.

Figure 4.13. Utah teapot curvature estimation.

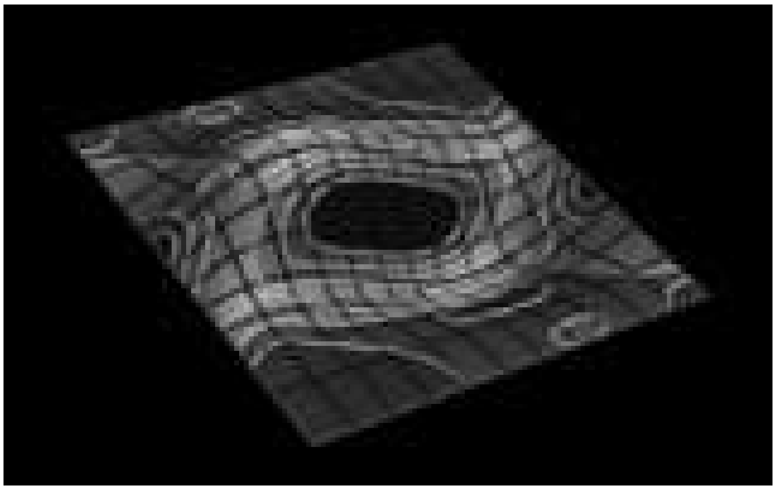

Figure 4.14. The surface is subdivided into regions with different curvature bounds.

4.3 Some Remarks

A method to partition a surface into three disjoint trimmed surfaces (convex, concave, and saddle) and to determine global bounds on surface curvatures, has been presented here which combines symbolic and numeric methods. The hybrid method was found to be robust and fast. The computation involved in the creation of a property surface that is exact to machine accuracy usually takes less than a second for a bicubic Bézier surface on an SGI 240/GTX (25MHz R3000). This symbolic computation has closed forms with complexity directly bounded by the

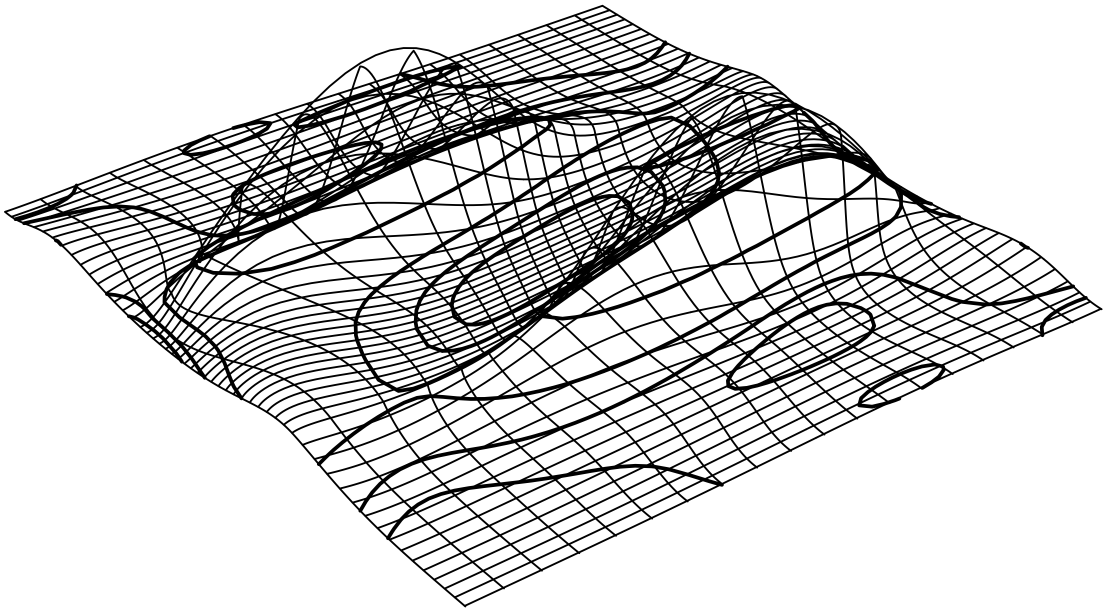

Figure 4.15. Curvature surface bound, ξ, of the surface in Figure 4.14.

surface orders and continuity (knot vectors). Contouring usually takes an order of magnitude longer than that. This numeric process involves high order property surfaces which make subdivision more expensive.

The orders of the resulting property surfaces are high. A second fundamental form determinant property surface for a bicubic B-spline surface has degree 14. The degree of the property surfaces $\psi(u, v)$, $\phi(u, v)$ and $\xi(u, v)$ is even higher, degree 30. However, because the evaluation of Bézier and B-spline representations is robust, the high order does not introduce any numerical problems [31] in evaluation.

Because milling is several magnitudes slower than even the contouring process, and the same toolpath may be used thousands of times, computation time is not a major factor in optimizing the milling process. The ability to isolate regions in a surface with specific curvature bounds makes it possible to mill the surface more optimally by using the largest tool possible for each region.