

Numerical parameterization of curves and surfaces

Erich Hartmann¹

*Darmstadt University of Technology, Department of Mathematics, Schlossgartenstr. 7,
D-64289 Darmstadt, Germany*

Received January 1999; revised August 1999

Abstract

A method for parameterizing nearly arbitrary implicit plane/space curves and surfaces is introduced. The parameterizations are of class C^{n-1} if the given curves/surfaces are of class C^n . The computation of points and derivatives is performed numerically. These parameterizations can be used for controlled determination of points on curves and surfaces and for the application of developed techniques for parametric curves and surfaces (mesh generation, texture mapping, curve integrals, surface integrals ...) to implicit curves and surfaces. The idea of the normalform of a curve/surface introduced in recent papers makes it possible to apply the numerical parameterization to nearly arbitrary curves and surfaces. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: Parameterization; Implicit curve; Implicit surface; Normalform; Mesh generation; Curvature; Foot point; Intersection curve

1. Introduction

Modern CAD-systems are based on parametrically defined curves and surfaces. The parametric representation has a lot of advantages. For example: It is easy to calculate points or quadrangle meshes which are preferred in finite element methods. Implicit curves and surfaces have, in praxis, an essential disadvantage: It is difficult to calculate points on curves and surfaces in a predictable way. On the other hand there are problems, for example blending curves and surfaces, which have for implicitly defined curves and surfaces simple solutions (cf. (Hartmann, 1998a)). So one is interested in shifting from one representation to the other. The transition from parametric to implicit can be done by using the numerical implicitization introduced in (Hartmann, 1998a). The converse, parameterization of implicit curves and surfaces, is tackled in literature in two

¹ E-mail: ehartmann@mathematik.tu-darmstadt.de.

different ways: (1) approximation by parametric curves and surfaces and (2) exact parameterization. There are several papers proposing algorithms for the *approximation* of algebraic curves and surfaces by parametric curves and surfaces (Waggenpack and Anderson, 1989; Bajaj and Xu, 1997a, 1997b). The papers (Bajaj and Xu, 1994; Sederberg and Nishita, 1991) deal with the approximation of surface/surface intersection curves. Papers which propose *exact* representations of plane curves or intersection curves are mostly restricted to algebraic curves. (Abhyankar and Bajaj, 1989) and (Garrity and Warren, 1989) introduce algorithms for representing intersection curves of algebraic surfaces by a data structure that consists of a plane curve, a linear function mapping points on the space curve to points on the plane curve, and rational functions mapping points on the plane curve to points on the space curve (see also (Berry, 1997)). (Abhyankar and Bajaj, 1988) deal with plane algebraic curves. (Markot and Magedson, 1991) describes a numerical parameterization of the intersection curve of two parametric surfaces. They start with an approximating polygon generated by a surface/surface intersection algorithm. The polygon is parameterized and a procedure using a Newton iteration calculates for a point on the polygon with parameter t a point on the intersection curve. It is shown that this parameterization is piecewise C^2 .

The parameterization of plane curves, surfaces and intersection curves proposed in this paper is applicable to all C^2 -continuous regular curves and surfaces, parametrically or implicitly or even more generally defined. It is based on the idea of the normalform for plane curves and surfaces respectively. The normalform is an extension of the Hessian normalform for lines/planes to curves/surfaces (see definition below). As usually the normalform is not known explicitly the evaluation of the normalform function and its derivatives is done numerically by appropriate foot point algorithms. The described parameterizations of curve arcs and surface patches are of class C^{n-1} if the given curves/surfaces are of class C^n .

In Section 2 the definition of the normalform and basic properties are given. Section 3 describes the numerical parameterization of planar curves, lists an algorithm for its realization and demonstrates the method at an example. Section 4 is dedicated to the surface case. It contains an algorithm for the numerical parameterization of surfaces which calculates surface points, the first derivative vectors of the parameterization and the first and second fundamental forms at a surface point. Section 5 deals with the curvature and the determination of foot points on intersection curves of surfaces. These results are applied to numerical parameterization of intersection curves.

2. The normalform of a curve/surface

Analogously to the Hessian normalform of a line in \mathbb{R}^2 and a plane in \mathbb{R}^3 the normalform of a curve/surface is defined.

Definition. Let $\Gamma(\Phi)$ be a smooth implicit curve (surface) $h = 0$ in \mathbb{R}^2 (\mathbb{R}^3). If the function h is continuously differentiable and $\|\nabla h\| = 1$ on $\Gamma(\Phi)$ and in a vicinity of $\Gamma(\Phi)$ then the equation $h = 0$ is called *normalform* of $\Gamma(\Phi)$ and h the corresponding *normalform function* or (as of its geometrical meaning) *oriented distance function*.

From the theory of the non linear partial differential equation $\|\nabla h\|^2 = 1$ (cf. (Courant and Hilbert, 1962; Weise, 1966)) follows:

Result. *Let $\Gamma(\Phi)$ be a C^2 -continuous curve (surface) in \mathbb{R}^2 (\mathbb{R}^3). Then there exists in a vicinity of $\Gamma(\Phi)$ a unique differentiable function h such that $h = 0$ is the normalform of $\Gamma(\Phi)$.*

If $\Gamma(\Phi)$ is of continuity class C^n then function h is the same.

h and gradient ∇h have the following properties (Hartmann, 1999):

$$h(\mathbf{x} + \delta \nabla h(\mathbf{x})) = h(\mathbf{x}) + \delta, \quad \nabla h(\mathbf{x} + \delta \nabla h(\mathbf{x})) = \nabla h(\mathbf{x}).$$

If $\mathbf{x} \in \Gamma(\Phi)$ then $h(\mathbf{x} + \delta \nabla h(\mathbf{x})) = \delta$ is the (oriented) distance of point $\mathbf{x} + \delta \nabla h(\mathbf{x})$ to the curve (surface) and \mathbf{x} is the foot point of $\mathbf{x} + \delta \nabla h(\mathbf{x})$ on the curve (surface). Equation $h = \delta$ describes the offset curve (surface) of distance δ .

Function h is known explicitly in rare cases only. In general the evaluation of h is done numerically by determining foot points (cf. (Hartmann, 1999)). Once the foot point of a point is known, the first and second derivatives of h can be evaluated using the normal vector and the curvature(s) of the curve (surface) (Hartmann, 1999).

3. Parameterization of planar curves

For the normalform function h of a planar curve we get the special properties (Hartmann, 1999):

- For a point on curve $h = 0$ we have:
 ∇h is eigenvector of the Hessian matrix H_h with eigenvalue 0,
the tangent vector is an eigenvector of H_h with the curvature κ as eigenvalue.
- For a curve point \mathbf{x} and a point $\mathbf{x}_\delta := \mathbf{x} + \delta \nabla h(\mathbf{x})$ the following relation is valid:

$$\kappa(\mathbf{x}_\delta) = \frac{\kappa(\mathbf{x})}{1 + \delta \kappa(\mathbf{x})}.$$

Let $f(x, y) = 0$ be a planar implicit regular (i.e., $\nabla f \neq \mathbf{0}$) curve with normalform $h(x, y) = 0$ and $\Gamma: \mathbf{x} = \gamma(t), t \in [a, b]$ a parametric curve in the vicinity of curve $f = 0$. (For the application below Γ is chosen as a Bézier curve which is a rough approximation of an arc of $f = 0$.) Both curves are supposed to be differentiable enough. Then the parametric curve

$$\Gamma_0: \mathbf{x} = \gamma_0(t) := \gamma(t) - h(\gamma(t))\nabla h(\gamma(t)), \quad t \in [a, b],$$

is an arc of the given curve $f = 0$, because of $h(\gamma - h(\gamma)\nabla h(\gamma)) = h(\gamma) - h(\gamma) = 0$ (see basic properties of the normalform function h above). Differentiating yields

$$\begin{aligned} \dot{\gamma}_0^T &:= \dot{\gamma}^T - (\nabla h^T \nabla h)\dot{\gamma}^T - h H_h \dot{\gamma}^T \\ &= (I - h H_h)(I - \nabla h^T \nabla h)\dot{\gamma}^T, \end{aligned}$$

where \mathbf{x}^T assigns the transpose of the row vector \mathbf{x} , I the 2×2 unit matrix and H_h the Hessian matrix of h . In the equation above h , ∇h and H_h are evaluated at point $\gamma(t)$.

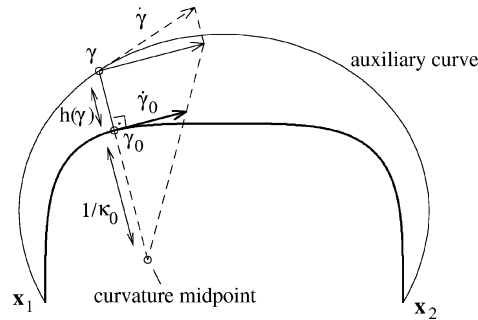


Fig. 1. The relation between the derivatives $\dot{\gamma}$ and $\dot{\gamma}_0$.

Vector $(I - \nabla h^T \nabla h) \dot{\gamma}^T$ is a tangent vector of the curve $h = h(\gamma(t)) = \text{const}$ (offset curve of $h = 0$) and can be expressed by $(\dot{\gamma} \cdot \mathbf{r}) \mathbf{r}$ with $\mathbf{r} := (-h_y, h_x)$. Hence

$$H_h(\gamma)(I - \nabla h(\gamma)^T \nabla h(\gamma)) \dot{\gamma}^T = H_h(\dot{\gamma}^T \mathbf{r}) \mathbf{r}^T = \kappa(\gamma)(\dot{\gamma}^T \mathbf{r}) \mathbf{r}^T.$$

Because $\nabla h(\gamma) = \nabla h(\gamma_0)$ and

$$\kappa(\gamma) = \frac{\kappa(\gamma_0)}{1 + h(\gamma)\kappa(\gamma_0)}$$

(see properties of ∇h above) we get

$$\dot{\gamma}_0 = \frac{(\dot{\gamma} \cdot \mathbf{r}(\gamma_0)) \mathbf{r}(\gamma_0)}{1 + h(\gamma)\kappa(\gamma_0)},$$

γ , $\dot{\gamma}$ and γ_0 evaluated at parameter t . $\gamma_0(t)$ is the foot point of $\gamma(t)$ on curve $h = 0$ and $h(\gamma)$ the oriented distance $\text{sign}(\nabla h(\gamma_0) \cdot (\gamma - \gamma_0)) \|\gamma - \gamma_0\|$ of point $\gamma(t)$ to curve $h = 0$.

Fig. 1 gives a geometric interpretation of the formula for the derivative $\dot{\gamma}_0$.

The *algorithm* for the numerical parameterization of an arc Γ_0 of a planar C^2 -continuous curve (implicit, parametric, ...) is as follows

- (1) Determine start- and endpoint $\mathbf{x}_1, \mathbf{x}_2$ of the arc Γ_0 to be parameterized.
- (2) Choose a Bézier or B-spline curve $\Gamma: \mathbf{x} = \gamma(t), t \in [0, 1]$ that interpolates $\mathbf{x}_1, \mathbf{x}_2$ and is a rough approximation of Γ_0 . (The denominator in the formula below for the tangent vector must be positive! This means curve Γ should be close to Γ_0 where the curvature of Γ_0 is large.)
- (3) Point $\gamma_0(t)$ is the foot point of point $\gamma(t)$ on curve Γ_0 . (For an appropriate algorithm see (Hartmann, 1999).)
- (4) Let \mathbf{n}_0 be the unit normal vector, \mathbf{r}_0 the unit tangent vector and κ_0 the curvature of Γ_0 at point $\gamma_0(t)$. With

$$h(\gamma) = \text{sign}(\mathbf{n}_0 \cdot (\gamma(t) - \gamma_0(t))) \|\gamma(t) - \gamma_0(t)\|$$

we get the *derivative* at point $\gamma_0(t)$:

$$\dot{\gamma}_0 = \frac{(\dot{\gamma} \cdot \mathbf{r}_0) \mathbf{r}_0}{1 + h(\gamma)\kappa_0}.$$

In case of parameterizing an *implicit curve* $f(x, y) = 0$:

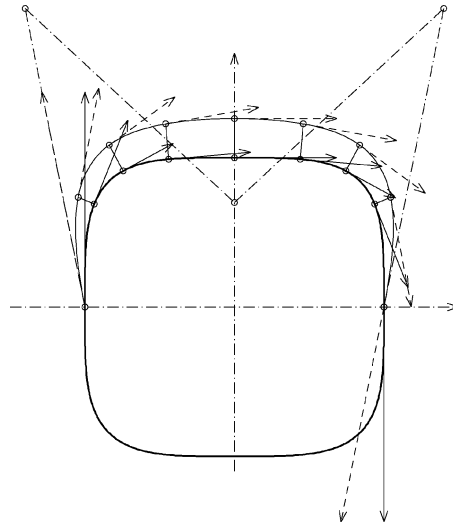


Fig. 2. Parameterization of the upper part of the curve $x^4 + y^4 = 1$.

$$\mathbf{n}_0 := \nabla f / \|\nabla f\|, \quad \mathbf{r}_0 := (-f_y, f_x) / \|\nabla f\|,$$

$$\kappa_0 = \frac{f_y^2 f_{xx} - 2f_x f_y f_{xy} + f_x^2 f_{yy}}{(f_x^2 + f_y^2)^{3/2}}.$$

Remark. As the above result on normalforms the introduced parameterization is of class C^{n-1} in case the auxiliary parametric curve and the given curve are C^n -continuous.

Example. Fig. 2 shows the auxiliary Bézier curve used for a numerical parameterization of the implicit curve $x^4 + y^4 - 1 = 0$ together with points $\gamma(i/n)$, foot points $\gamma_0(i/n)$ and their derivatives $\dot{\gamma}(i/n)$, $\dot{\gamma}_0(i/n)$, $i = 1, \dots, n$, for $n = 8$.

The advantage of the numerical parameterization shows (for example) the determination of the arc length λ of the parameterized arc. The arc length determined

- (a) by evaluation the curve integral $\int_0^1 |\dot{\gamma}_0(t)| dt$ with Romberg quadrature needs 64 intermediate points to get the result $\lambda = 3.50885$,
- (b) by summing the polygon chords needs 825 intermediate points to meet the same accuracy.

4. Parameterization of surfaces

For the normalform function h of a surface we get the special properties (Hartmann, 1999):

- The *normal curvature* for unit tangent direction \mathbf{v} is $\kappa_n = \mathbf{v} H_h \mathbf{v}^T$. (κ_n is the curvature of the surface curve contained in the normal plane determined by point \mathbf{x} , the gradient ∇h and the tangent vector \mathbf{v} .)

- The *eigenvalues* of the Hessian matrix H_h are

$$\lambda_1 = 0 \text{ with eigenvector } \nabla h,$$

$$\lambda_2 = \kappa_{\min}, \lambda_3 = \kappa_{\max} \text{ (main curvatures).}$$

Let $f(\mathbf{x}) = 0$ be an implicit regular (i.e., $\nabla f \neq \mathbf{0}$) surface with normalform $h(\mathbf{x}) = 0$ and $\Phi: \mathbf{x} = \mathbf{S}(u, v)$, $(u, v) \in D \subset \mathbb{R}^2$ a parametric surface in the vicinity of surface $f = 0$. (For the application below Φ is chosen as a tensor product Bézier surface which is a rough approximation of a patch of $f = 0$.) Both surfaces are supposed to be differentiable enough. Then the parametric surface

$$\Phi_0: \mathbf{x} = \mathbf{S}_0(u, v) := \mathbf{S}(u, v) - h(\mathbf{S}(u, v))\nabla h(\mathbf{S}(u, v)), \quad (u, v) \in D,$$

is a patch of the given surface $f = 0$, because of

$$h(\mathbf{S} - h(\mathbf{S})\nabla h(\mathbf{S})) = h(\mathbf{S}) - h(\mathbf{S}) = 0$$

(see basic properties of the normalform function h above). Differentiating yields:

$$\mathbf{S}_{0,u}^T = (I - \nabla h^T \nabla h - h H_h) \mathbf{S}_u^T,$$

$$\mathbf{S}_{0,v}^T = (I - \nabla h^T \nabla h - h H_h) \mathbf{S}_v^T.$$

Function h , ∇h and H_h are evaluated at point $\mathbf{S}(u, v)$.

The *algorithm* for the numerical parameterization of an implicit surface $f(\mathbf{x}) = 0$:

- (1) Choose a tensor product Bézier or B-spline surface $\Phi: \mathbf{x} = \mathbf{S}(u, v)$, $(u, v) \in [0, 1] \times [0, 1]$ which is a rough approximation of the surface patch Φ_0 to be parameterized.
- (2) Point $\mathbf{S}_0(u, v)$ is the foot point of point $\mathbf{S}(u, v)$ on surface $f = 0$. (For an appropriate algorithm see (Hartmann, 1999).)
- (3) If the first derivatives and the fundamental forms are needed then calculate for the normalform function h of the given surface $f = 0$:

$$\nabla h(\mathbf{S}(u, v)) = \nabla f(\mathbf{S}_0(\mathbf{u}, \mathbf{v})) / \|\dots\|$$

and

$$h(\mathbf{S}) = \text{sign}(\nabla h(\mathbf{S}_0) \cdot (\mathbf{S} - \mathbf{S}_0)) \|\mathbf{S} - \mathbf{S}_0\|$$

for parameters (u, v) .

- (4) Determine the Hessian matrix H_h at point $\mathbf{S}(u, v)$ by the method given in (Hartmann, 1999).
- (5) The *derivatives* of $\mathbf{S}_0(u, v)$ are

$$\mathbf{S}_{0,u}^T = (I - \nabla h^T \nabla h - h H_h) \mathbf{S}_u^T,$$

$$\mathbf{S}_{0,v}^T = (I - \nabla h^T \nabla h - h H_h) \mathbf{S}_v^T,$$

h , ∇h and H_h evaluated at point $\mathbf{S}(u, v)$.

- (6) The coefficients of the *first fundamental form* are:

$$E := \mathbf{S}_{0,u}^2, \quad F := \mathbf{S}_{0,u} \cdot \mathbf{S}_{0,v}, \quad G := \mathbf{S}_{0,v}^2.$$

(7) The coefficients of the *second fundamental form*:

- (a) Determine for three directions $\mathbf{v}_i := \xi_i \mathbf{S}_{0,u} + \eta_i \mathbf{S}_{0,v}$, $i = 1, 2, 3$, the normal curvatures of the surface $f(\mathbf{x}) = 0$:

$$\kappa_i = \frac{\mathbf{v}_i H_f \mathbf{v}_i^T}{\|\nabla f\|}.$$

- (b) Using the first fundamental form (from item (6)) and the normal curvatures $\kappa_1, \kappa_2, \kappa_3$ (out of (a)) the coefficients L, M, N of the second fundamental form can be determined from the linear system

$$\kappa_i = \frac{L\xi_i^2 + 2M\xi_i\eta_i + N\eta_i^2}{E\xi_i^2 + 2F\xi_i\eta_i + G\eta_i^2}, \quad i := 1, 2, 3.$$

Remark.

- (a) Because of the result on normalforms in Section 2 the introduced parameterization of a surface is of class C^{n-1} if the auxiliary parametric surface and the given surface are C^n -continuous.
- (b) If there is no demand for continuity (for example, for texture mapping, cf. (Zonenschein et al., 1998)) one may use the more simple procedure `surfacepoint` out of (Hartmann, 1999), Section 5.1.2, instead of the foot point procedure. Usually one recognizes no evident differences.

Example 1. Fig. 3 shows the auxiliary Bézier surface used for a numerical parameterization of a patch of the implicit surface $f = 0$ with

$$f(x, y, z) = (1 - \mu)(x^2 + z^2 - 1)(y^2 + z^2 - 1) - \mu((r^2 - x^2 - y^2 - z^2)(c - z^2))^3$$

for $\mu = 0.002$, $r = 3$, $c = 1.1$ which is a G^2 -blending surface of the cylinders $x^2 + z^2 = 1$ and $y^2 + z^2 = 1$. (For the triangulation of the surface cf. (Hartmann, 1998b).)

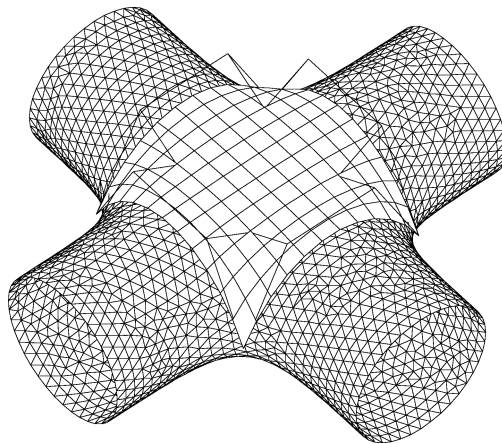


Fig. 3. Bézier surface patch used for parameterization of an implicit surface.

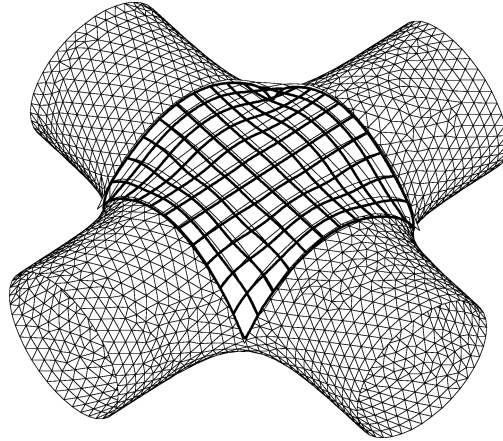


Fig. 4. Bézier surface patch and the parameterized implicit surface patch.

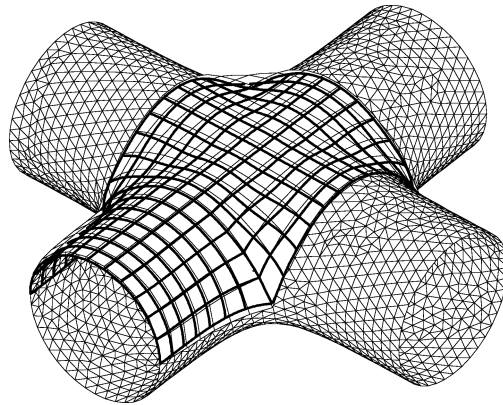


Fig. 5. Extension of the parameterization.

Fig. 4 shows the Bézier surface and the parameterized implicit surface patch.

Fig. 5 shows an extension of the previous parameterized surface patch.

Fig. 6 shows a net of the total parameterization of the implicit blending surface.

Fig. 7 is an example for drawing curves on implicit surfaces using numerical parameterization.

For the sake of evidence the used Bézier surface approximates the implicit surface very rough. This causes the little bit wavy behavior of the parameter curves on the surface. But nevertheless the curves are (numerically) exact on the surface.

The introduced numerical parameterization is applicable to all surfaces for which foot points can be determined. Here is an example of a surface which is defined by normalforms:

Example 2. Given are

- (1) the implicit surface $\Phi_1: (x - 2)^4 + y^4 - r_1^4 = 0, r_1 = 2,$

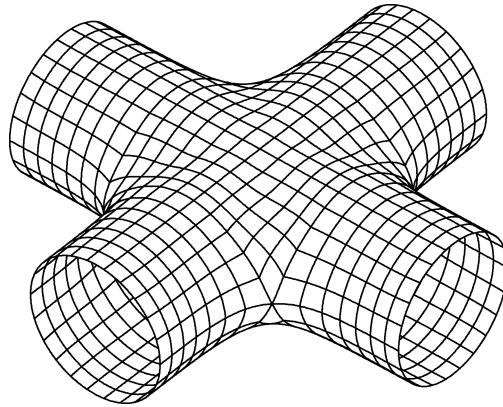


Fig. 6. Net of the total parameterization.

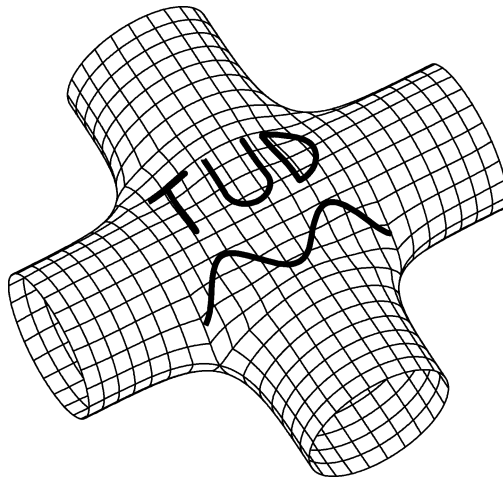


Fig. 7. Curves on an implicit surface.

(2) the parametric surface patch

$$\Phi_2: \mathbf{x} = (10v - 5, 10u - 5, 6(u - u^2 + v - v^2)), \quad 0 \leq u \leq 1, \quad 0 \leq v \leq 0.8,$$

(3) the parametric surface patch

$$\Phi_3: \mathbf{x} = (6(u - u^2 + v - v^2) - 5, 10u - 5, 10v - 5), \\ 0 \leq u \leq 1, \quad 0.5 \leq v \leq 1.$$

Let $h_1(\mathbf{x}) = 0$, $h_2(\mathbf{x}) = 0$, $h_3(\mathbf{x}) = 0$ be the normalforms of these surfaces. The implicit surface

$$\Phi: f(\mathbf{x}) := h_1(\mathbf{x})h_2(\mathbf{x})h_3(\mathbf{x}) = c, \quad c > 0,$$

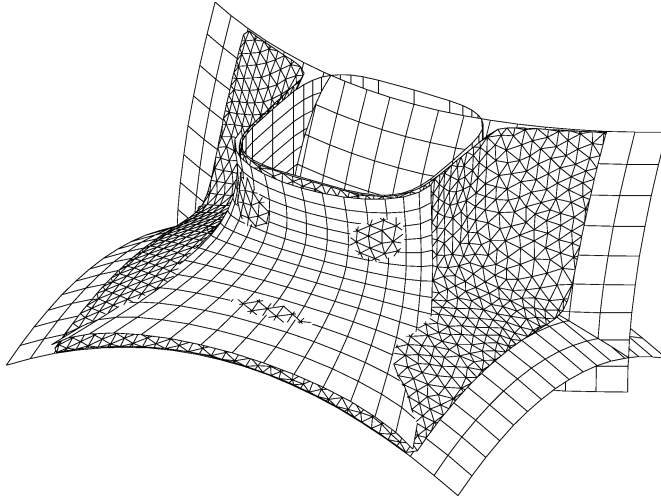


Fig. 8. Approximation of three surfaces and an auxiliary Bézier patch.

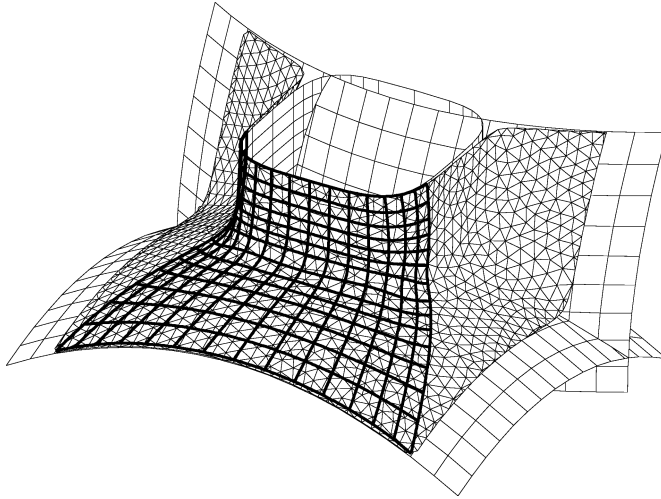


Fig. 9. Parameterization of an approximation of three surfaces.

is a smooth approximation of the set of surfaces Φ_1, Φ_2, Φ_3 . The approximation is independent of their representations.

Fig. 8 shows a triangulation of $f(\mathbf{x}) = c$ for $c = 0.2$ and a Bézier surface which will be used for the parameterization. Fig. 9 shows a net of the parameterization.

5. Parameterization of an intersection curve

For the parameterization of an intersection curve we need a formula for the curvature of such a curve which will be derived first.

5.1. The curvature of an intersection curve

Let Φ_1, Φ_2 be two intersecting C^2 -continuous surfaces with normalforms $h_1 = 0, h_2 = 0$ and let the intersection curve Γ be regular (i.e., $\nabla h_1 \times \nabla h_2 \neq \mathbf{0}$) and represented by the arc length parameterization $\mathbf{x} = \gamma(s)$. Hence $\|\nabla h_1\| = 1, \|\nabla h_2\| = 1$ and $\|\gamma'\| = 1, \gamma' \cdot \gamma'' = 0$.

Differentiating the following equations

$$h_1(\gamma(s)) = 0, \quad h_2(\gamma(s)) = 0$$

yields

$$\nabla h_1 \cdot \gamma' = 0, \quad \nabla h_2 \cdot \gamma' = 0$$

and

$$\nabla h_1 \cdot \gamma'' + \gamma' H_{h_1} \gamma'^T = 0, \quad \nabla h_2 \cdot \gamma'' + \gamma' H_{h_2} \gamma'^T = 0.$$

Because γ' is orthogonal to ∇h_1 and ∇h_2 and $\|\gamma'\| = 1$ we can choose the parameterization of Γ such that

$$\gamma' = \frac{\nabla h_1 \times \nabla h_2}{\|\nabla h_1 \times \nabla h_2\|}.$$

γ'' is a linear combination of ∇h_1 and ∇h_2 . Because the calculation becomes simpler we use another base of the normal plane:

$$\gamma'' = \alpha(\nabla h_1 \times \gamma') + \beta(\nabla h_2 \times \gamma') = (\alpha \nabla h_1 + \beta \nabla h_2) \times \gamma'.$$

Using $\gamma' H_{h_i} \gamma'^T = \kappa_i$ (normal curvature on surface Φ_i in direction γ') we get from the equations $\nabla h_1 \cdot \gamma'' = -\kappa_1, \nabla h_2 \cdot \gamma'' = -\kappa_2$ the result:

$$\gamma'' = \frac{\kappa_2 \nabla h_1 - \kappa_1 \nabla h_2}{\|\nabla h_1 \times \nabla h_2\|} \times \gamma'.$$

This formula can be written independently of the representations of the intersecting surfaces:

Let Φ_1, Φ_2 be two intersecting C^2 -continuous surfaces with unit normals $\mathbf{n}_1, \mathbf{n}_2, \|\mathbf{n}_1 \times \mathbf{n}_2\| \neq 0$, and normal curvatures κ_1, κ_2 in direction $\mathbf{n}_1 \times \mathbf{n}_2$ at a point of the intersection curve Γ then

•

$$\mathbf{t} := \frac{\mathbf{n}_1 \times \mathbf{n}_2}{\|\mathbf{n}_1 \times \mathbf{n}_2\|}$$

is the *tangent unit vector* and

•

$$\mathbf{c} := \frac{(\kappa_2 \mathbf{n}_1 - \kappa_1 \mathbf{n}_2) \times (\mathbf{n}_1 \times \mathbf{n}_2)}{(\mathbf{n}_1 \times \mathbf{n}_2)^2}$$

is the *curvature vector* that means \mathbf{c} describes the change of the tangent unit vector \mathbf{t} and its amount $\kappa = \|\mathbf{c}\|$ is the *curvature*.

5.2. Determining foot points on an intersection curve

Because for any C^2 -continuous surface there exists a normal form it suffice to give a foot point algorithm for implicitly defined surfaces.

We start with two implicit surfaces $\Phi_1: f_1(\mathbf{x}) = 0$, $\Phi_2: f_2(\mathbf{x}) = 0$ and assume that they are regular and sufficiently differentiable for the operations described below. An essential step of the foot point algorithm is the following procedure `curvepoint` which determines for a point \mathbf{p} in the “neighborhood” of the curve a point \mathbf{p}_0 on the intersection curve.

Procedure `curvepoint`:

(C0) Let $f_1(\mathbf{x}) = 0$, $f_2(\mathbf{x}) = 0$ be two intersecting implicit surfaces and \mathbf{p} a point.

(C1) $\mathbf{q}_0 := \mathbf{p}$,

repeat $\mathbf{q}_{k+1} = \mathbf{q}_k + \mathbf{\Delta}_k$, where

$\mathbf{\Delta}_k = \alpha_k \nabla f_1(\mathbf{q}_k) + \beta_k \nabla f_2(\mathbf{q}_k)$ and $\mathbf{\Delta}_k \cdot \nabla f_j(\mathbf{q}_k) = -f_j(\mathbf{q}_k)$, $j = 1, 2$.

(Newton step for $g_j(\alpha, \beta) := f_j(\mathbf{q}_k + \alpha \nabla f_1(\mathbf{q}_k) + \beta \nabla f_2(\mathbf{q}_k))$, $j = 1, 2$.)

until $\|\mathbf{q}_{k+1} - \mathbf{q}_k\|$ is “sufficiently” small.

Curve point: $\mathbf{p}_0 = \mathbf{q}_{k+1}$.

Procedure `footpoint`:

(F0) Let $f_1(\mathbf{x}) = 0$, $f_2(\mathbf{x}) = 0$ be two intersecting implicit surfaces and \mathbf{p} a point.

(F1) $\mathbf{c}_0 = \text{curvepoint}(\mathbf{p})$

(F2) **repeat** $\mathbf{r}_i := \nabla f_1(\mathbf{c}_i) \times \nabla f_2(\mathbf{c}_i)$ (tangent direction),

$$\mathbf{t}_{i+1} = \mathbf{c}_i + \frac{(\mathbf{p} - \mathbf{c}_i) \cdot \mathbf{r}_i}{\mathbf{r}_i^2} \mathbf{r}_i \quad (\text{foot point on tangent line}),$$

$\mathbf{c}_{i+1} = \text{curvepoint}(\mathbf{t}_{i+1})$.

until $\|\mathbf{c}_{i+1} - \mathbf{c}_i\|$ is “sufficiently” small.

Foot point: $\mathbf{p}_0 = \mathbf{c}_{i+1}$.

Remark. In case of problems considering convergence of the foot point algorithm one should improve it by a “parabola step” (see (Hartmann, 1999), Section 5.1.2).

5.3. Parameterization of an intersection curve

Because parametrically defined surfaces can be numerically implicitized (see (Hartmann, 1998a)) we restrict our considerations to intersection curves of implicit surfaces.

Let be

(1) $\Phi_1: f_1(\mathbf{x}) = 0$, $\Phi_2: f_2(\mathbf{x}) = 0$ two intersecting surfaces with $\nabla f_1 \times \nabla f_2 \neq \mathbf{0}$ at any point of intersection curve Σ ,

- (2) $\mathbf{x}_1, \mathbf{x}_2$ start and end point of an arc Γ_0 of the intersection curve Σ ,
- (3) $\Gamma: \mathbf{x} = \gamma(t)$ a curve which interpolates $\mathbf{x}_1, \mathbf{x}_2$ and which is a rough approximation of arc Γ_0 to be parameterized. The parameterization of Γ_0 is denoted by $\gamma_0(t)$.

The calculation of the point $\gamma_0(t)$ on the intersection curve which corresponds to $\gamma(t)$ is done by the foot point algorithm above. The corresponding tangent vector $\dot{\gamma}_0(t)$ is

- (a) in case of $\gamma(t) = \gamma_0(t)$ the projection of $\dot{\gamma}(t)$ onto the unit tangent vector of the intersection curve,
- (b) in case of $\gamma(t) \neq \gamma_0(t)$ determined by projecting the curves Γ and Γ_0 onto the plane containing foot point $\gamma_0(t)$, point $\gamma(t)$ and which is parallel to the tangent at point $\gamma_0(t)$. Then the result for planar curves is applicable.

The *algorithm* for the numerical parameterization of the intersection curve of two implicit surfaces $f_1(\mathbf{x}) = 0, f_2(\mathbf{x}) = 0$ (this algorithm is applicable to all C^2 -continuous surfaces by using their normalforms):

- (1) Determine start- and endpoint $\mathbf{x}_1, \mathbf{x}_2$ of the arc Γ_0 to be parameterized.
- (2) Choose a Bézier or B-spline curve $\Gamma: \mathbf{x} = \gamma(t), t \in [0, 1]$, that interpolates $\mathbf{x}_1, \mathbf{x}_2$ and is a rough approximation of Γ_0 . (Curve Γ should be close to Γ_0 where the curvature of Γ_0 is large.)
- (3) Point $\gamma_0(t)$ is the foot point of point $\gamma(t)$ on the intersection curve $f_1 = 0, f_2 = 0$. (See foot point algorithm above.)
- (4) If the derivative $\dot{\gamma}_0(t)$ is needed:

Let be

$$(a) \quad \mathbf{n}_i := \frac{\nabla f_i}{\|\nabla f_i\|}, \quad i = 1, 2,$$

the unit normals of the surfaces,

$$(b) \quad \mathbf{r} := \frac{\mathbf{n}_1 \times \mathbf{n}_2}{\|\mathbf{n}_1 \times \mathbf{n}_2\|}$$

the unit tangent vector at point $\gamma_0(t)$,

$$(c) \quad \kappa_i := \frac{\mathbf{r} H_{f_i} \mathbf{r}^T}{\|\nabla f_i\|}, \quad i = 1, 2,$$

the normal curvature of surface $f_i = 0$ in direction \mathbf{r} of the intersection curve,

$$(d) \quad \mathbf{c} := \frac{(\kappa_2 \mathbf{n}_1 - \kappa_1 \mathbf{n}_2) \times (\mathbf{n}_1 \times \mathbf{n}_2)}{(\mathbf{n}_1 \times \mathbf{n}_2)^2}$$

the curvature vector of the intersection curve,

$$(e) \quad \mathbf{d} := \gamma(t) - \gamma_0(t).$$

Then

$$\dot{\gamma}_0(t) = \frac{\dot{\gamma}(t) \cdot \mathbf{r}}{1 - \mathbf{d} \cdot \mathbf{c}} \mathbf{r}$$

is the *derivative vector* of the numerical parameterization.

Remark. If we consider a smooth spatial curve as limit of the pencil of corresponding canal surfaces we get from the result on normalforms in Section 2 that the introduced

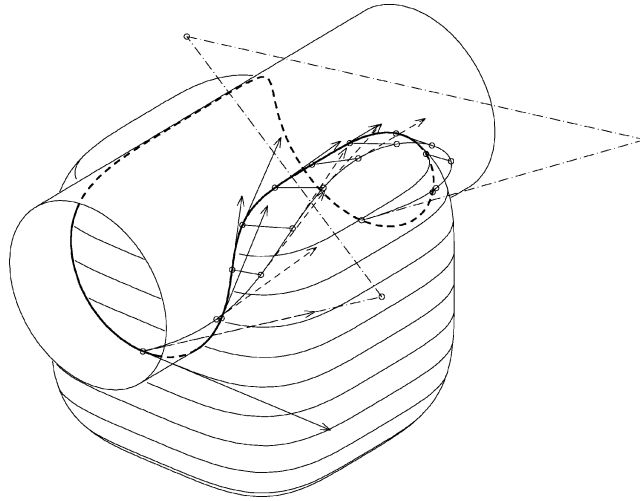


Fig. 10. Parameterization of the intersection curve of two implicit surfaces.

parameterization of an intersection curve is of class C^{n-1} if the auxiliary parametric curve and the given surfaces are C^n -continuous.

Example 1. Fig. 10 shows the numerical parameterization (points and tangent vectors) of the intersection curve of the implicit surfaces

$$f_1(x, y, z) = x^4 + y^4 + z^4 - 1 = 0, \quad f_2(x, y, z) = y^2 - (z - c)^2 - c^2 = 0$$

for $c = 0.6$. The auxiliary parametric curve is a quartic Bézier curve.

The arc length of the parameterized arc is $\lambda = 4.14507$ determined with Romberg quadrature for 128 intermediate points.

The application of the numerical parameterization to the intersection curve of an implicit surface (cylinder) and a surface that is only defined by using normalforms (blend surface of two Bézier surfaces) is shown in the next example.

Example 2. Let Φ_1 and Φ_2 be two tensor product Bézier patches with parametric representations

$$\Phi_1: \mathbf{x} = (10v - 5, 10u - 5, 6(u - u^2 + v - v^2)),$$

$$\Phi_2: \mathbf{x} = (6(u - u^2 + v - v^2), 10u - 5, 10v - 5)$$

and normal forms $h_1 = 0$ and $h_2 = 0$ respectively (cf. (Hartmann, 1999)). The normalforms are not explicitly known but can be evaluated numerically. Using the plane curve

$$k(c, d) = (1 - \mu) \frac{cd}{c_0 d_0} - \mu \left(1 - \frac{c}{c_0} - \frac{d}{d_0} \right)^{n+1} = 0, \quad 0 < \mu < 1, \quad n \geq 0,$$

we get the blending surface $F(\mathbf{x}) := k(h_1(\mathbf{x}), h_2(\mathbf{x})) = 0$ which has G^n -continuous contact to the surfaces Φ_1 and Φ_2 (cf. (Hartmann, 1998a, Hartmann, 1999)).

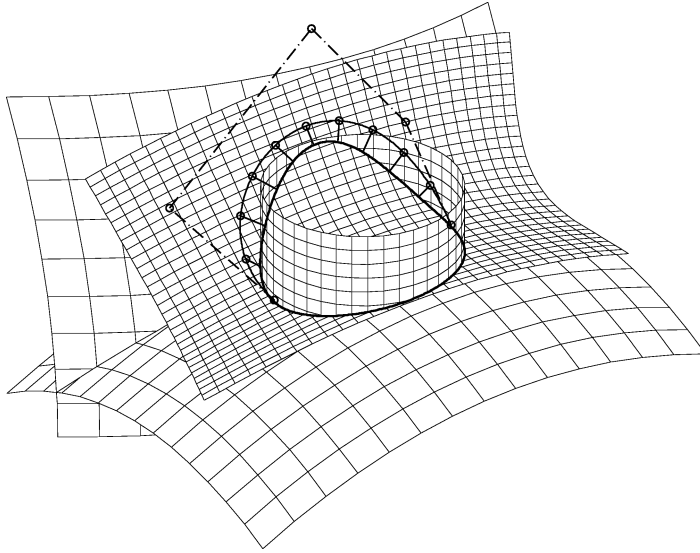


Fig. 11. Parameterization of the intersection curve of a G^2 -blending surface and a cylinder.

Fig. 11 shows a numerical parameterization of the intersection curve of the blending surface with a quadratic cylinder. The auxiliary parametric curve is a Bézier curve of degree 4.

6. Conclusion

The idea of numerical parameterization has been introduced. It allows to handle implicit (plane or space) curves and surfaces or even more general ones as parametric curves and surfaces. The advantages are:

- (1) Predictable curve/surface point calculation.
- (2) Developed techniques for parametric curves and surfaces are applicable to implicit or more general curves and surfaces (for example, (a) mesh generation (b) texture mapping).
- (3) The first derivative vectors make it possible to carry out curve and surface integrals for non parametric curves and surfaces.

But one should notice that numerical parameterization is no algorithm for detecting singularities of curves and surfaces. In order to get a regular parameterization the given curve arc and surface patch, respectively, has to be free of singularities. The user should have a good idea of the curve/surface to be parameterized in advance, perhaps using suitable algorithms for implicit curves (cf. (Hartmann, 1999)), implicit surfaces (cf. (Schmidt, 1993; Hartmann, 1998b)) and intersection curves (cf. (Bajaj et al., 1988; Barnhill and Kersey, 1990; Hartmann, 1998a)) for displaying. The necessary auxiliary parametric curve/surface (rough approximation of the curve/surface to be parameterized) can be designed interactively using Bézier or B-Spline curves/surfaces.

References

- Abhyankar, S.S. and Bajaj, C.L. (1988), Automatic parameterization of rational curves and surfaces III: Algebraic plane curves, *Comput. Aided Geometric Design* 5, 309–321.
- Abhyankar, S.S. and Bajaj, C.L. (1989), Automatic parameterization of rational curves and surfaces IV: Algebraic space curves, *ACM Trans. Graph.* 8, 325–334.
- Bajaj, C.L., Hoffmann, C.M., Lynch, R.E. and Hopcroft, J.E.H. (1988), Tracing surface intersections, *Computer Aided Geometric Design* 5, 285–307.
- Bajaj, C.L. and Xu, G. (1994), NURBS approximation of surface/surface intersection curves, *Advances in Comput. Mathematics* 2, 1–21.
- Bajaj, C.L. and Xu, G. (1997a), Piecewise rational approximations of real algebraic curves, *J. Comput. Math.* 15, 55–71.
- Bajaj, C.L. and Xu, G. (1997b), Spline approximations of real algebraic surfaces, *J. Symbolic Comput.* 23, 315–333.
- Barnhill, R.E. and Kersey, S.N. (1990), A marching method for parametric surface/surface intersection, *Computer Aided Geometric Design* 7, 257–280.
- Berry, T.G. (1997), Parameterization of algebraic space curves, *J. Pure Appl. Algebra* 117, 81–95.
- Courant, R. and Hilbert, D. (1962), *Methods of Mathematical Physics II*, Interscience Publishers, Wiley, New York.
- Gao, X.S. and Chou, S.C. (1992), On the parameterization of algebraic curves, *Appl. Algebra Eng. Commun. Comput.* 3, 27–38.
- Garrity, T. and Warren, J. (1989), On computing the intersection of a pair of algebraic surfaces, *Computer Aided Geometric Design* 6, 137–153.
- Hartmann, E. (1998a), Numerical implicitization for intersection and G^n -continuous blending of surfaces, *Computer Aided Geometric Design* 15, 377–397.
- Hartmann, E. (1998b), A marching method for the triangulation of surfaces, *The Visual Computer* 14, 95–108.
- Hartmann, E. (1998c), The normalform of a planar curve and its application to curve design, in: Daehlen, M., Lyche, T. and Schumaker, L., eds., *Mathematical Methods for Curves and Surfaces II*, Vanderbilt Univ. Press, Nashville.
- Hartmann, E. (1999), On the curvature of curves and surfaces defined by normalforms, *Computer Aided Geometric Design* 16, 355–376.
- Kosters (1991), Curvature-dependent parameterization of curves and surfaces, *Computer-Aided Design* 8, 569–578.
- Lau, T.S., Lo, S.H. and Lee, C.K. (1997), Generation of quadrilateral mesh over analytical curved surfaces, *Finite Elements in Analysis and Design* 27, 251–272.
- Markot, R.P. and Magedson, R.L. (1991), Procedural method for evaluating intersection curves of two parametric surfaces, *Computer-Aided Design* 23, 395–404.
- Schmidt, M. (1993), Cutting cubes—visualizing implicit surfaces by adaptive polygonization, *The Visual Computer* 10, 101–115.
- Sederberg, T.W. and Nishita, T. (1991), Geometric Hermite approximation of surface patch intersection curves, *Computer Aided Geometric Design* 8, 97–114.
- Waggenspack, W.N. and Anderson, D.C. (1989), Piecewise parametric approximations for algebraic curves, *Computer Aided Geometric Design* 6, 33–53.
- Weise, K.H. (1966), *Differentialgleichungen*, Vandenhoeck & Ruprecht, Göttingen.
- Zonenschein, R., Gomes, J., Velho, L. and Figueiredo, L.H. (1998), Controlling texture mapping onto implicit surfaces with particle systems, *Proceedings of Implicit Surfaces '98*, 131–138.