Geometric Modelling Summer 2018

Prof. Dr. Hans Hagen

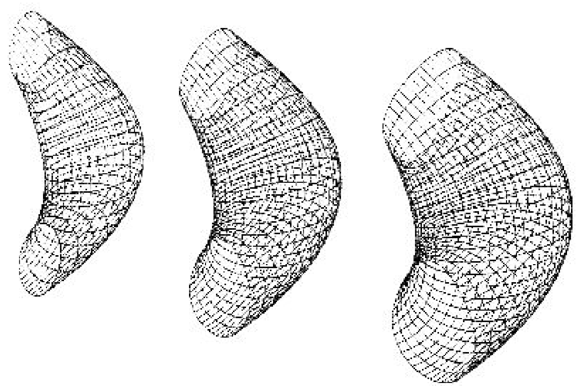
http://hci.uni-kl.de/teaching/geometric-modelling-ss2018

Offset Surfaces

Motivation

Offset-Surfaces occur e.g. in the modelling of surfaces with realistic material thickness. We first present the geometric foundations. Starting from a parameterited surface $X : U \to \mathbb{E}^3$ the **offset surface** $X_d(u, w)$ with distance d along the surface's normal vector $N(u, w)$ is parameterized as follows:

$$X_d(u, w) = X(u, w) + d \cdot N(u, w)$$



Basics

All geometric invariants of the offset-surface can be described with the invariants of the destination surface and the distance d as follows

$$I_d = (1 - K \cdot d^2)I - 2d(1 + H \cdot d)II \quad \text{(first basic form)}$$

where K is the Gaussian curvature and H the mean curvature. I and II are the first respective second fundamentalform.

Basics

The connection of the volume forms is important:

$$det(g_{ij}^d) = (Kd^2 + 2Hd + 1)^2 \cdot det(g_{ij})$$

where one has: $K \cdot d^2 + 2H \cdot d + 1 = (1 + k_{min} \cdot d) \cdot (1 + k_{max} \cdot d)$, k_{min} and k_{max} being the main curvatures of the surface $X(u, w)$.

Basics: Geometric facts

normalvector:	$N_d = \frac{Kd^2+2Hd+1}{\|\|Kd^2+2Hd+1\|\|} \cdot N =: \Gamma(u,w) \cdot N$
second basic form:	$II_d = \Gamma(u,w) \cdot (K \cdot d \cdot I + (1+2Hd)II)$
gausian curvature:	$K_d = \frac{K}{Kd^2+2Hd+1}$
mean curvature:	$H_d = \frac{H+K\cdot d}{\|\|Kd^2+2Hd+1\|\|}$

Basics

These geometric facts provide regularity conditions to offset surfaces and allow some kind of classification of possible ways of degeneration:

edge in offset surface	$k^d_{max} > -d^{-1}$ and $k^d_{min} = -d^{-1}$
peak in the offset surface	$k^d_{max} = -d^{-1}$ and $k^d_{min} < -d^{-1}$
"spherical peak"	$k^d_{max} = -d^{-1} = k^d_{min}$

Offset-Surface-Algorithm

Offset-Surface-Algorithm (Farouki)

1. Approximate or interpolate a set of offset points with bicubic Coons patches.
2. Subdivide into smaller patches if deviation is too large.

Offset-Surface-Algorithm

Like for offset curves, $\mathcal{G}^1$- and $\mathcal{G}^2$-conditions lead to a more optimal procedure. Two surfaces $X(u,w)$ and $Y(u,w)$ touch each other with an order $k(k=1,2)$ if the following two conditions hold

$$: \; k=1 \qquad \begin{bmatrix} Y_u \\ Y_w \end{bmatrix} = \begin{bmatrix} a_{10} & b_{10} \\ a_{01} & b_{01} \end{bmatrix} \cdot \begin{bmatrix} X_u \\ X_w \end{bmatrix}$$

$$k=2 \qquad \begin{bmatrix} Y_{uu} \\ Y_{uw} \\ Y_{ww} \end{bmatrix} = \begin{bmatrix} a_{20} & b_{20} \\ a_{11} & b_{11} \\ a_{02} & b_{02} \end{bmatrix} \cdot \begin{bmatrix} X_u \\ X_w \end{bmatrix} + \begin{bmatrix} a_{10}^2 & 2a_{10}b_{10} & b_{10}^2 \\ a_{10}a_{01} & a_{10}b_{01} + a_{01}b_{10} & b_{10}b_{01} \\ a_{01}^2 & 2a_{01}b_{01} & b_{01}^2 \end{bmatrix} \cdot \begin{bmatrix} X_{uu} \\ X{uw} \\ X{ww} \end{bmatrix}$$

Offset-Surface-Algorithm

Surfaces that touch in first order have share their a tangential plane in every shared point.
Surfaces that touch in second order additionally share a Dupin-Indicatrix in these points.

Offset-Surface Algorithm

Offset-Surface Algorithm (Hoschek)

1. Boundary Curve of an approximation

$$Y(u,w) := \sum_{i=0}^{p} \sum_{k=0}^{q} W_{ik} B_i^p(u) B_k^q(w)$$

of the offset-surface $X_d = X(u,w) + d \cdot N(u,w)$

Offset-Surface Algorithm

Offset-Surface Algorithm (Hoschek)

1. where

$$X(u,w) := \sum_{i=0}^{n}\sum_{k=0}^{m} V_{ik} B_i^n(u) B_k^m(w) :$$

$$\begin{aligned} W_{00} &= V_{00} + d \cdot N(u_0, w_0) \\ W_{0q} &= V_{0m} + d \cdot N(u_0, w_1) \\ W_{p0} &= V_{n0} + d \cdot N(u_1, w_0) \\ W_{pq} &= V_{nm} + d \cdot N(u_1, w_1) \end{aligned}$$

2. Calculate the offset curves of the boundary curve of surface $X(u,w)$ according to the algorithms for cubic resp. quintic offset curves.

Offset-Surface Algorithm

Offset-Surface Algorithm (Hoschek)

3. To calculate the "inner" Bézierpoints W_{11}, W_{12}, W_{21}, W_{22} in the bicubic case, one demands requires the following conditions to hold for the derivatives across the boundary:

$$
\begin{aligned}
Y_u(u_0, w) &= \alpha_1(w) X_u(u_0, w) \\
Y_u(u, w_0) &= \alpha_2(u) X_w(u, w_0) \\
Y_u(u_1, w) &= \alpha_3(w) X_u(u_1, w) \\
Y_u(u, w_1) &= \alpha_4(u) X_w(u, w_1)
\end{aligned}
$$

$$
\begin{aligned}
\text{where } \alpha_1(w) &= \tfrac{3}{n}\lambda_3 B_0^2(w) + w_1 B_1^2(w) + \tfrac{3}{n}\lambda_7 B_2^2(w) \\
\alpha_2(u) &= \tfrac{3}{n}\lambda_1 B_0^2(u) + w_2 B_1^2(u) + \tfrac{3}{n}\lambda_5 B_2^2(u) \\
\alpha_3(w) &= \tfrac{3}{n}\lambda_4 B_0^2(w) + w_3 B_1^2(w) + \tfrac{3}{n}\lambda_8 B_2^2(w) \\
\alpha_4(u) &= \tfrac{3}{n}\lambda_2 B_0^2(u) + w_4 B_1^2(u) + \tfrac{3}{n}\lambda_6 B_2^2(u)
\end{aligned}
$$

The parameters w_i $\quad i = 1, \ldots, 4$ are optimization parameters.

Offset-Surface Algorithm

Offset-Surface Algorithm (Hoschek)

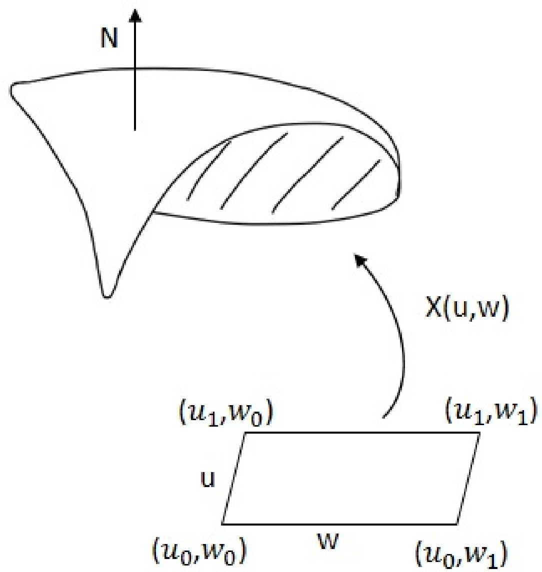
4. Inserting the α_i into the equations for the Y_i, one obtains a linear equation system for the inner control points W_{11}, W_{12}, W_{21} , W_{22} in linear dependence on w_1, w_2, w_3, w_4. The total error D can be minimized by the Least-Square conditions: $\frac{\partial D}{\partial w_i} = 0 \quad i = 1, \ldots, 4$. The resulting linear equation system provides the optimal formparameters w_i in the last step.

For the bicubic case, see [Hoschek, 1989].

Surface Offsetting

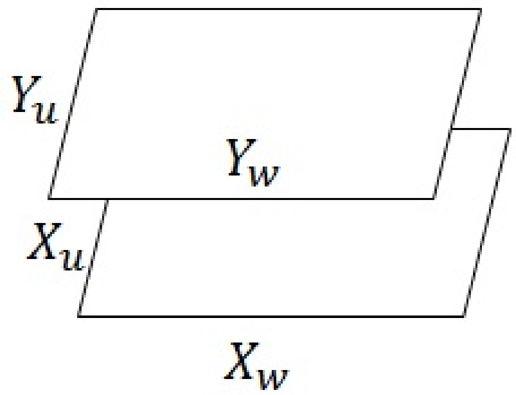
Surface Offsetting (Farin-Hagen-Hansford)

surface	$X(u,w)$	$:=$	$\sum_{p=0}^{n}\sum_{q=0}^{m} b_{pq}B_p^n(u)B_q^m(w)$
offset-surface	$X_d(u,w)$	$:=$	$X(u,w)+d\cdot N(u,w)$
approximation	$Y(u,w)$	$:=$	$\sum_{i=0}^{3}\sum_{j=0}^{3} a_{ij}B_i^3(u)B_j^3(w)$



Surface Offsetting (Farin-Hagen-Hansford)

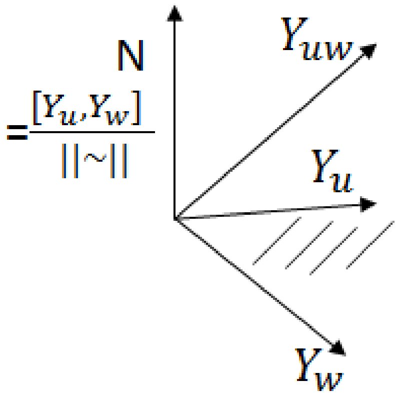
1. condition: **Tangent Planes are parallel**



$$
\begin{aligned}
Y(u_r, w_s) &= X(u_r, w_s) + d \cdot N(u_r, w_s) \\
Y_u(u_r, w_s) &= \lambda_{rs} X_w(u_r, w_s) + \overline{\lambda} X_u(u_r, w_s) \\
Y_w(u_r, w_s) &= \mu_{rs} X_w(u_r, w_s) + \overline{\mu} X_u(u_r, w_s)
\end{aligned}
$$

Surface Offsetting (Farin-Hagen-Hansford)

2. condition: **least square twist fitting**



$$Y_{uw}(u_r, w_s) = d_{rs}N(u_r, w_s) + \beta_{rs}Y_u(u_r, w_s) + \gamma_{rs}Y_w(u_r, w_s)$$

$d_{rs}, \beta_{rs}, \gamma_{rs} \rightarrow$ input for least square fitting.

Surface Offsetting (Farin-Hagen-Hansford)

Algorithm (Farin-Hagen-Hansford)

1. Evaluate the progenitor in a $(k+1) \times (k+1)$ grid within the boundary of the domain. Compute the exact offset at these points and call them E_{ij}.
   Evaluate the progenitor in $k+1$ places along each boundary (excluding the endpoints) and compute the exact offset. The exact offset along the $v = V_0$ edge is called E_{i0}. The other boundaries follows similarly.
2. Compute the exact offset at each corner.
3. Compute the offset approximation Bézier points along each boundary.

Surface Offsetting (Farin-Hagen-Hansford)

Algorithm (Farin-Hagen-Hansford) – cond't

4. Local error along $v = V_0$ edge:

$$\delta_i = E_{i0} - \sum_{j=0}^{3} d_{j0} B_j^3(u_i)$$

.

Surface Offsetting (Farin-Hagen-Hansford)

Algorithm (Farin-Hagen-Hansford) – cond't

4. Global error:

$$\delta = \sum_{i=0}^{k} \delta_i^2 = \sum_{i=0}^{k} [D_{i0} - \lambda_{10} \frac{m}{3} \frac{u_1 - u_0}{v_1 - v_0} (b_{01} - b_{00}) B_1^3(u_i)$$

$$-\overline{\lambda}_{10} \frac{n}{3} (b_{10} - b_{00}) + \lambda_{20} \frac{m}{3} \frac{u_1 - u_0}{v_1 - v_0} (b_{n1} - b_{n0}) B_2^3(u_i)$$

$$-\overline{\lambda}_{20} \frac{n}{3} (b_{n-10}) B_2^3(u_i)]^2$$

where

$D_{i0} = E_{i0} - a_{00} B_0^3(u_i) - a_{00} B_1^3(u_i) - a_{30} B_2^3(u_i) - a_{30} B_3^3(u_i)$.

This implies a 4×4 system of equations:

$\frac{\partial \delta}{\partial \lambda_{10}} = 0; \frac{\partial \delta}{\partial \overline{\lambda}_{10}} = 0; \frac{\partial \delta}{\partial \lambda_{20}} = 0; \frac{\partial \delta}{\partial \overline{\lambda}_{20}} = 0$ for each boundary

Surface Offsetting (Farin-Hagen-Hansford)

Algorithm (Farin-Hagen-Hansford) – cond't

5. The twist at (u_0, v_0) is given by

$$\frac{3}{u_1 - u_0} \cdot \frac{3}{v_1 - v_0}(a_{00} - a_{10} - a_{01} - a_{11})$$

$$= \alpha_{00} N_{00} + \beta_{00} \frac{3}{u_1 - u_0}(b_{10} - b_{00}) + \gamma_{00} \frac{3}{v_1 - v_0}(b_{01} - b_{00})$$

We can solve this for "the twist point" a_{11}. Similar procedure for the twist points a_{12}, a_{21}, a_{22}.

Surface Offsetting (Farin-Hagen-Hansford)

Algorithm (Farin-Hagen-Hansford) – cond't

6. local error:

$$\delta_{rt} = E_{rt} - \sum_{i=0}^{3} \sum_{j=0}^{3} a_{ij} B_i^3(u_r) B_j^3(w_t)$$

.

Surface Offsetting

One could avoid approximations entirely, if a plane curve $X(t) = (x(t), y(t))$ satisfies the following condition:

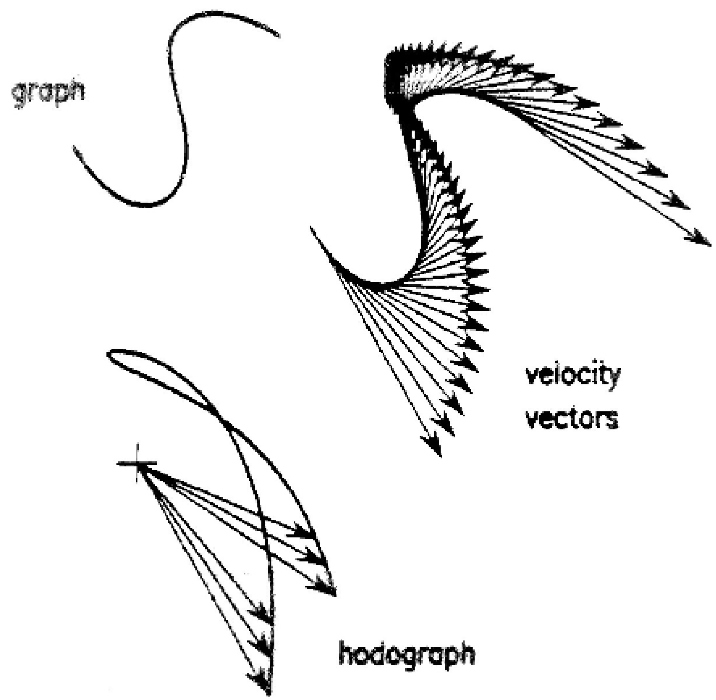
$$\dot{x}^2(t) + \dot{y}^2(t) = \sigma^2(t)$$

with a suitable polynomial $\sigma(t)$.

Surface Offsetting

$$X(t) = (x(t), y(t))$$

is a plane curve with a so-called **Pytagorean Hodograph**, if there exists a polynomial $\sigma(t)$ with $\dot{x}^2(t) + \dot{y}^2(t) = \sigma^2(t)$.



Phytagorean Hodograph

A parameterized curve with Pytagorean Hodograph has:

(a) a polynomial arc length function $s(t)$,

(b) a rational offset curve $X_d(t) := X(t) + d \cdot N(t)$.

In the context of these considerations, one naturally desires to characterize curves of this kind by conditions on the control structures.
For this purpose, a result from number theory is helpful (see [Farouki, 1990] and [Farouki, Sakkalis, 1990]).

Phytagorean Hodograph

The polynomials $a(t)$, $b(t)$, and $c(t)$ then satisfy

$$a^2(t) + b^2(t) = c^2(t)$$

if and only if

$$\begin{aligned} a(t) &= w(t)u^2(t) - v^2(t) \\ b(t) &= 2w(t)u(t)v(t) \\ c(t) &= w(t)u^2(t) + v^2(t) \end{aligned}$$

hold, where coprime polynomials $u(t)$ and $v(t)$.

Phytagorean Hodograph

$$\begin{aligned} w(t) &= 1 \\ u(t) &:= u_0 B_0'(t) + u_1 B_1'(t) \\ v(t) &:= v_0 B_0'(t) + v_1 B_1'(t) \end{aligned}$$

yield the identity:

$$u^2(t) - v^2(t) = (u_0^2 - v_0^2) B_0^2(t) + (u_0 u_1 - v_0 v_1) B_1^2(t)$$

$$+2u(t)v(t) = 2u_0 v_0 B_0^2(t) + (u_0 v_1 + u_1 v_0) B_1^2(t) + 2u_1 v_1 B_2^2(t)$$

Phytagorean Hodograph

The integrations

$$X(t) = \int_0^1 u^2(t) - v^2(t)$$

and

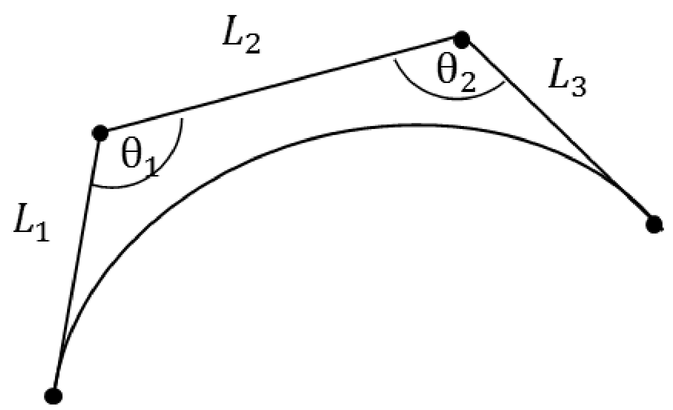
$$Y(t) = \int_0^1 2u(t)v(t)$$

result – combined with comparison of coefficients – in the conditions:

$$\begin{aligned} b_1 &= b_0 + \tfrac{1}{3}(u_0^2 - v_0^2, 2u_0 v_0) \\ b_2 &= b_1 + \tfrac{1}{3}(u_0 u_1 - v_0 v_1, u_0 v_1 + u_1 v_0) \\ b_3 &= b_2 + \tfrac{1}{3}(u_1^2 - v_1^2, 2u_1 v_1) \end{aligned}$$

Phytagorean Hodograph

These have the following geometric interpretation:



$$L_2 = \sqrt{L_1 \cdot L_3} \quad \text{and} \quad \Theta_1 = \Theta_2$$

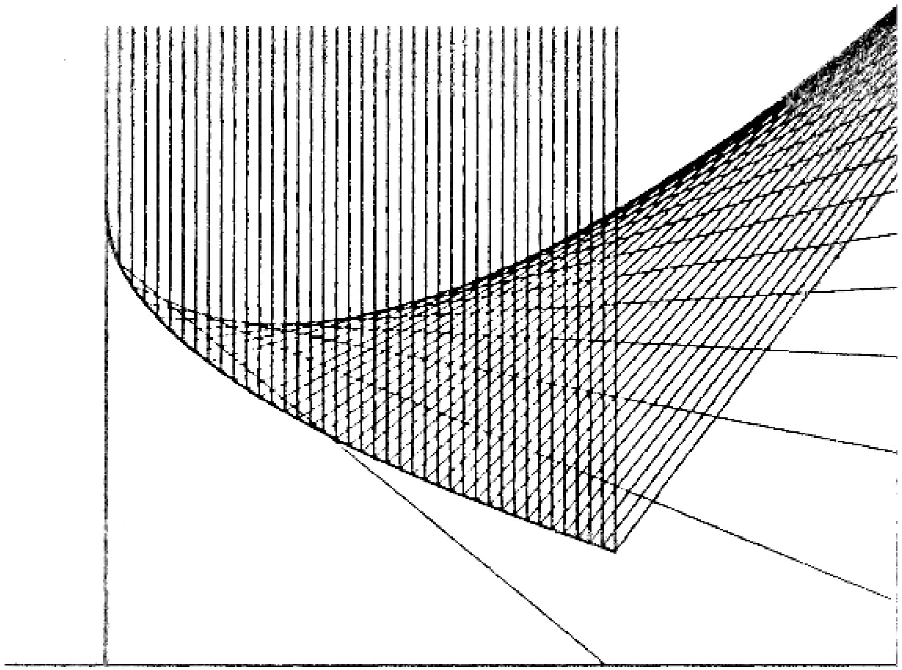
Phytagorean Hodograph

Further considerations (see [Farouki, 1990]) lead to the fact

Theorem

The so called **Tschirnhausen cubic** $T(k) := \begin{pmatrix} \sqrt{3} & (t^2 - 1) \\ t & (t^2 - 1) \end{pmatrix}$ is the only Pythagorean Hodograph cubic.

Phytagorean Hodograph



The so-called **caustic** is the envelope of the reflected rays and identical to the Tschirnhausen cubic.

Phytagorean Hodograph – Remarks

1. A Pythagorean-Hodograph cubic is convex,
2. For curves of higher order, algebraic relations result, that are difficult to interpret geometrically.

How about rational, planar curves?

What does the situation look like for rational planar curves?

Unfortunately, the trivial case of the straight line is the only rational planar Pythagorean Hodrograph curve (see [Farouki, 1991]). For the case of a surface, the following result is known: "All offsets of a non-developable rational ruled surface are rational".