Algorithmic Geometry WS 2017/2018

Prof. Dr. Hans Hagen
Benjamin Karer M.Sc.

http://gfx.uni-kl.de/~alggeom

Gordon-Coons Patches

Gordon-Coons Patches

Problem

The elementary Lagrange- and Newton interpolation schemes for surfaces do have the same severe disadvantages as in the curve case. So we have to come up with different ideas.

General Idea

Input: position and tangent information along and across a topological four sided patch
Key idea: blend the information through to fill out the patch.

Gordon-Coons Patches

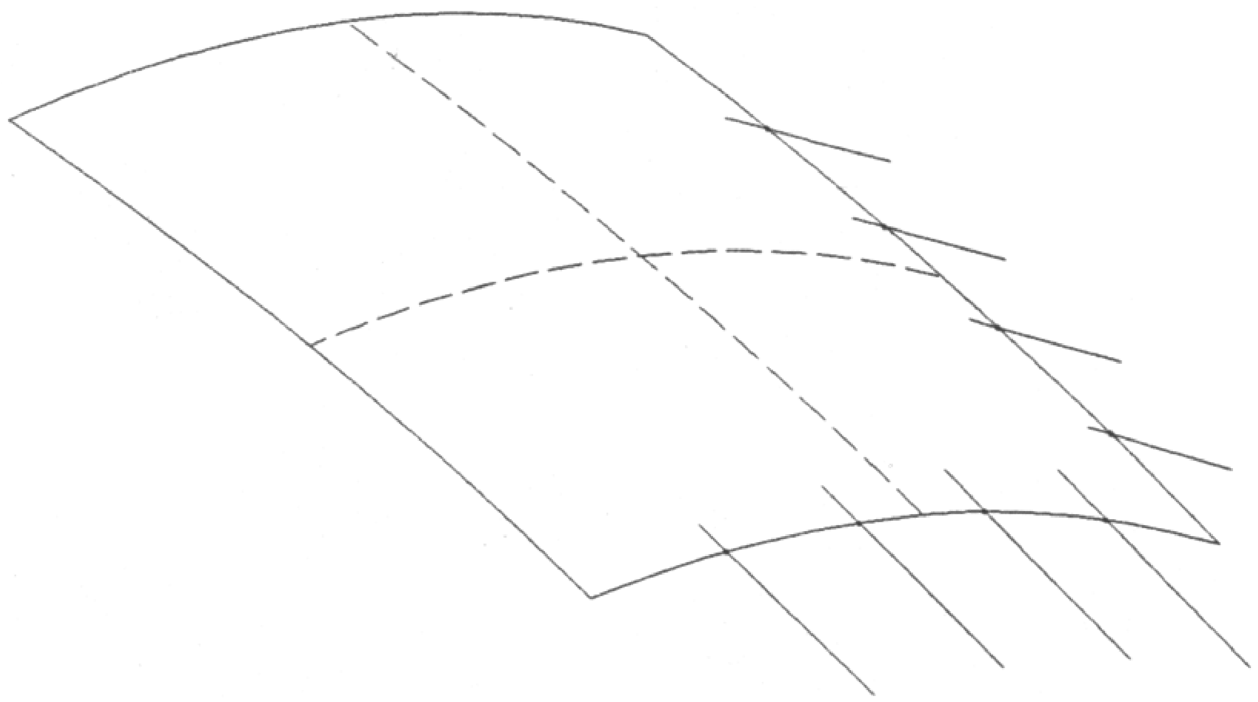

Figure: Example patch

Construction principle:

We start with a linear interpolation over $[0, 1]$,

$$f(x) := (1-x)\cdot f(0) + x\cdot f(1),$$

and generalize this to fit in a ruled surface, generated by a one-parameter family of straight lines

$$X(u,x) = (1-x)\cdot X(u,0) + x\cdot X(u,1) \qquad (1)$$

Gordon-Coons Patches

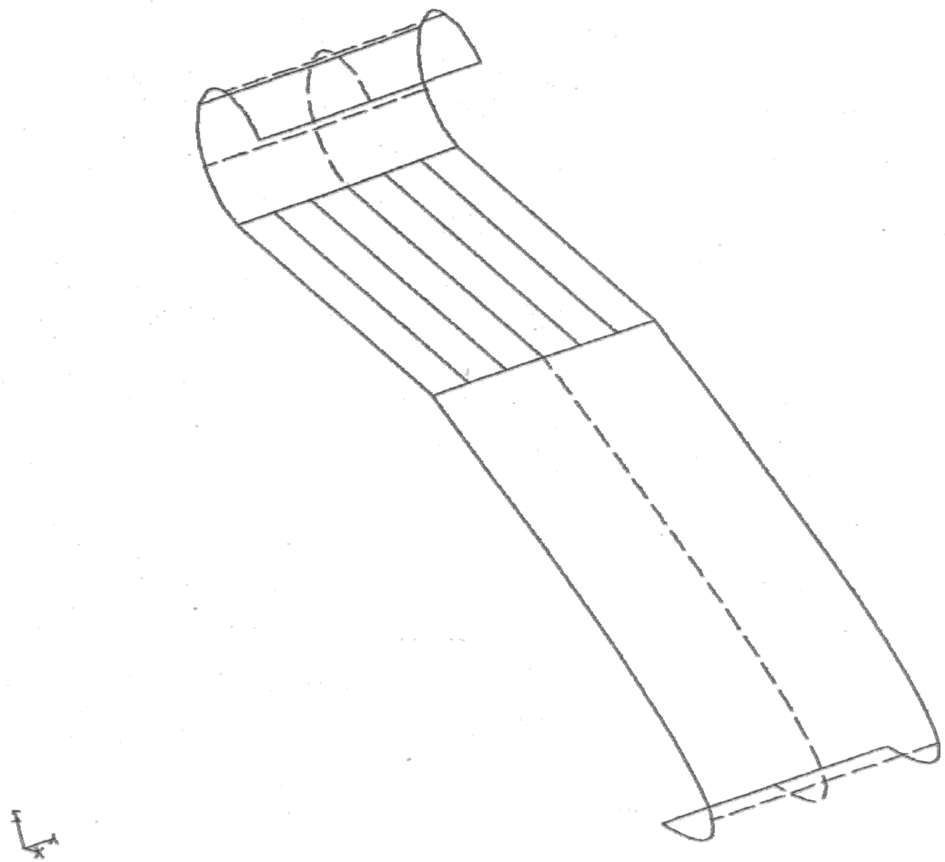

Figure: linearly connected surface patches

Gordon-Coons Patches

The next step is to replace this linear interpolation by a bilinear interpolation over the unit square:

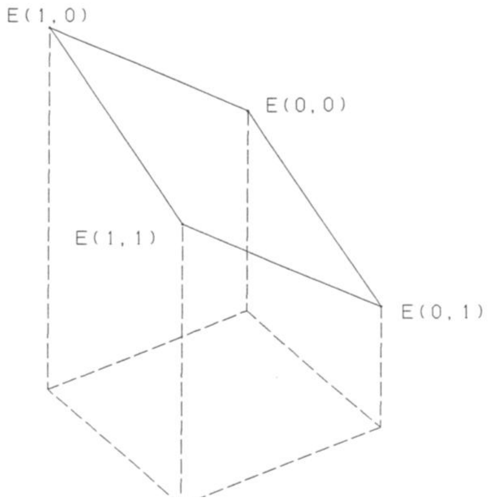

Figure: interpolation on $[0,1]^2$

$$
\begin{aligned}
P_1 E &:= (1-w)\cdot E(0,0) + w \cdot E(1,0) \\
P_2 E &:= (1-w)\cdot E(0,1) + w \cdot E(1,1) \\
QE &:= (1-u)\cdot P_1 E + u \cdot P_2 E
\end{aligned}
$$

Gordon-Coons Patches

We have to upgrade the projectors to upgrade the quality of the patch. Projectors are linear idempotent operators.
An operator T is a mapping $T : X \rightarrow Y$ (X is a linear space and Y a subspace of X), with

$$T(\alpha x_1 + \beta x_2) = \alpha T(x_1) + \beta T(x_2) \quad (\alpha, \beta \in \mathbb{R}).$$

A linear operator is idempotent iff $T \circ T(x) = T(x)$
Considering curved boundary curves, we cannot just sum up the operators:

Gordon-Coons Patches

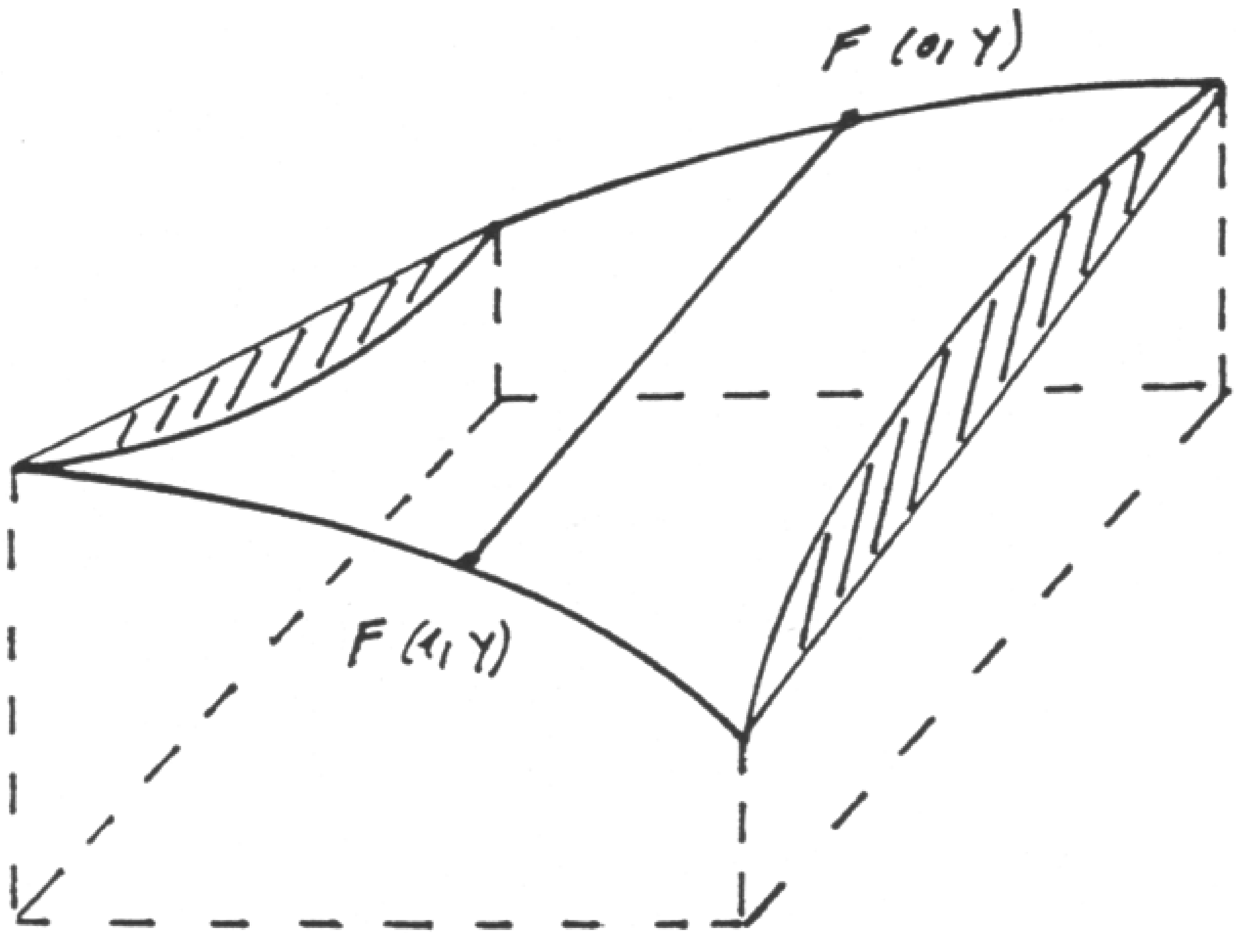

Figure: Error when interpolating curved surfaces linearly

Gordon-Coons Patches

Starting with one projector:

$$P_1 F := (1-u) \cdot F(0,w) + u \cdot F(1,w)$$

we have to "apply" the second projector to the "error" $F - P_1 F$:

$$P_2(F - P_1 F) := (1-w)(F - P_1 F)(u,0) + w(F - P_1 F)(u,1)$$

As a result we get the **Gordon scheme** :

$$QF := P_1 F + P_2 F - P_1 P_2 F = P_1 \oplus P_2(F) \qquad (2)$$

Applying this principle to the bilinear case we get the so called **bilinear Coons Patch**:

$$\begin{aligned} X(u,w) &:= (1-u)F(0,w) + uF(1,w) + (1-w)F(u,0) + wF(u,1) \\ &\quad -(1-u)\left((1-w)F(0,0) + wF(0,1)\right) \\ &\quad -u\left((1-w)F(1,0) + wF(1,1)\right) \\ &= -\begin{bmatrix} -1 \\ 1-u \\ u \end{bmatrix}^T \begin{bmatrix} 0 & F(u,0) & F(u,1) \\ F(0,w) & F(0,0) & F(0,1) \\ F(1,w) & F(1,0) & F(1,1) \end{bmatrix} \begin{bmatrix} -1 \\ 1-w \\ w \end{bmatrix} \qquad (3) \end{aligned}$$

Gordon-Coons Patches

Generalize the blending functions and the "information along and across the boundary":

$$P_1 X \quad := \quad \sum_{i=0}^{1} H_i(u) X(i, w) + \overline{H}_i(u) X_u(i, w)$$

$$P_2 X \quad := \quad \sum_{j=0}^{1} H_j(w) X(u, j) + \overline{H}_j(w) X_w(u, j)$$

$$X_u := \frac{\partial X}{\partial u} \quad \text{and } X_w = \frac{\partial X}{\partial w},$$

where $\left\{ H_0,\ H_1,\ \overline{H}_0,\ \overline{H}_1 \right\}$ are the cubic Hermite basis functions.

Gordon-Coons Patches

This boolean sum in matrix form:

$$X(u,w) = \begin{bmatrix} H_0(u) \\ H_1(u) \\ \overline{H}_0(u) \\ \overline{H}_1(u) \end{bmatrix}^T \begin{bmatrix} X(0,w) \\ X(1,w) \\ X_u(0,w) \\ X_u(1,w) \end{bmatrix} + \begin{bmatrix} H_0(w) \\ H_1(w) \\ \overline{H}_0(w) \\ \overline{H}_1(w) \end{bmatrix}^T \begin{bmatrix} X(u,0) \\ X(u,1) \\ X_w(u,0) \\ X_w(u,1) \end{bmatrix}$$
$$- \begin{bmatrix} H_0(u) \\ H_1(u) \\ \overline{H}_0(u) \\ \overline{H}_1(u) \end{bmatrix}^T \begin{bmatrix} X(0,0) & X(0,1) & X_w(0,0) & X_w(0,1) \\ X(1,0) & X(1,1) & X_w(1,0) & X_w(1,1) \\ X_u(0,0) & X_u(0,1) & X_{uw}(0,0) & X_{uw}(0,1) \\ X_u(1,0) & X_u(1,1) & X_{uw}(1,0) & X_{uw}(1,1) \end{bmatrix} \begin{bmatrix} H_0(w) \\ H_1(w) \\ \overline{H}_0(w) \\ \overline{H}_1(w) \end{bmatrix} \quad (4)$$

Gordon-Coons Patches

Constraints for the blending functions:

$$H_i(j) = \delta_i^j; \quad H_i'(j) = 0; \quad \overline{H}_i(j) = 0; \quad \overline{H}_i'(j) = \delta_i^j$$

General principle to create blending functions:

$$\left\{H_0(s), H_1(s), \overline{H}_0(s), \overline{H}_1(s)\right\} = \left\{s^3, s^2, s^1, s^0\right\} \cdot M, \quad s \in [0, 1]$$

Gordon-Coons Patches

$$\begin{bmatrix} H_0(0) & H_1(0) & \overline{H}_0(0) & \overline{H}_1(1) \\ H_0(1) & H_1(1) & \overline{H}_0(1) & \overline{H}_1(1) \\ H_0'(0) & H_1'(0) & \overline{H}_0'(0) & \overline{H}_1'(1) \\ H_0'(1) & H_1'(1) & \overline{H}_0'(1) & \overline{H}_1'(1) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix}}_{\mathcal{M}} \cdot M$$

$$\mathcal{M}^{-1} = M = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Gordon-Coons Patches

This leads to the cubic Hermite basis functions:

$$\begin{aligned} H_0(s) &= 2s^3 - 3s^2 + 1 & H_1(s) &= -2s^3 + 3s^2 \\ \overline{H}_0(s) &= s^3 - 2s^2 + s & \overline{H}_1(s) &= s^3 - s^2 \end{aligned}$$

We can rewrite the Coons patch in a more compact form:

$$\begin{bmatrix} -1 \\ H_0(u) \\ H_1(u) \\ \overline{H}_0(u) \\ \overline{H}_1(u) \end{bmatrix}^T \cdot \begin{bmatrix} 0 & X(u,0) & X(u,1) & X_w(u,0) & X_w(u,1) \\ X(0,w) & X(0,0) & X(0,1) & X_w(0,0) & X_w(0,1) \\ X(1,w) & X(1,0) & X(1,1) & X_w(1,0) & X_w(1,1) \\ X_u(0,w) & X_u(0,0) & X_u(0,1) & X_{uw}(0,0) & X_{uw}(0,1) \\ X_u(1,w) & X_u(1,0) & X_u(1,1) & X_{uw}(1,0) & X_{uw}(1,1) \end{bmatrix} \cdot \begin{bmatrix} -1 \\ H_0(w) \\ H_1(w) \\ \overline{H}_0(w) \\ \overline{H}_1(w) \end{bmatrix} \quad (5)$$

Gordon-Coons Patches

The representation of the boundary curve is still free! Assuming we have only position- and tangent information in some 3D–points (we use these points as vertices). It makes sense to “blend along” the boundary the same way we blend through the patch:

$$\begin{aligned}
X(u,j) &:= \sum_{i=0}^{1} H_i(u)X(i,j) + \overline{H}_i(u)X_u(i,j) \\
X(i,w) &:= \sum_{j=0}^{1} H_j(w)X(i,j) + \overline{H}_j(w)X_w(i,j) \\
X_w(u,j) &:= \sum_{i=0}^{1} H_i(u)X_w(i,j) + \overline{H}_i(u)X_{uw}(i,j) \\
X_u(i,w) &:= \sum_{j=0}^{1} H_j(w)X_u(i,j) + \overline{H}_j(u)X_{uw}(i,j) \qquad (6)
\end{aligned}$$

In this case the Coons patch looks like this:

$$\begin{bmatrix} H_0(u) \\ H_1(u) \\ \overline{H}_0(u) \\ \overline{H}_1(u) \end{bmatrix} \cdot \begin{bmatrix} X(0,0) & X(0,1) & X_w(0,0) & X_w(0,1) \\ X(1,0) & X(1,1) & X_w(1,0) & X_w(1,1) \\ X_u(0,0) & X_u(0,1) & X_{uw}(0,0) & X_{uw}(0,1) \\ X_u(1,0) & X_u(1,1) & X_{uw}(1,0) & X_{uw}(1,1) \end{bmatrix} \cdot \begin{bmatrix} H_0(w) \\ H_1(w) \\ \overline{H}_0(w) \\ \overline{H}_1(w) \end{bmatrix} \tag{7}$$

Short form: $H(u) \cdot C \cdot H(w)$. The matrix C contains only discrete values in the 4 corner points.

Data exchange format:

$$X(u,w) = U^T \cdot M \cdot C \cdot M^T \cdot W$$

$$\text{with: } U := \begin{bmatrix} u^3 \\ u^2 \\ u^1 \\ u^0 \end{bmatrix}, \quad W := \begin{bmatrix} w^3 \\ w^2 \\ w^1 \\ w^0 \end{bmatrix}, \quad M := \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Twist problem:

- twist input problem
- twist compatibility problem: $X_{uw} \neq X_{wu}$

Gregory Square

$$\text{replace the twist-part } \begin{pmatrix} X_{uw}(0,0) & X_{uw}(0,1) \\ X_{uw}(1,0) & X_{uw}(1,1) \end{pmatrix} \text{ with}$$

$$\begin{pmatrix} \frac{uX_{wu}(0,0)+wX_{uw}(0,0)}{u+w} & \frac{-uX_{wu}(0,1)+(w-1)X_{uw}(0,1)}{w-u-1} \\ \frac{(1-u)X_{wu}(1,0)+wX_{uw}(1,0)}{1-u+w} & \frac{(u-1)X_{wu}(1,1)+(w-1)X_{uw}(1,1)}{u-1+w-1} \end{pmatrix} \tag{8}$$

In the case of compatible twist ($X_{uw} = X_{wu}$) the Gregory Square has the usual form:

$$\begin{pmatrix} X_{uw}(0,0) & X_{uw}(0,1) \\ X_{wu}(1,0) & X_{wu}(1,1) \end{pmatrix} = \begin{pmatrix} X_{uw}(0,0) & X_{uw}(0,1) \\ X_{wu}(1,0) & X_{wu}(1,1) \end{pmatrix}$$

Gordon-Coons Patches

We get the rational correcting terms out of:

$$(P_i \oplus P_j)F + (P_j \oplus P_k)R \qquad \text{and}$$
$$(P_i \oplus P_j)F + (P_k \oplus P_j)R \qquad (i, j, k \text{ pairwise different})$$
$$\text{with } R := F - (P_i \oplus P_j)F \qquad \text{interpolate } F \in C'(\partial T).$$

The compatibility problem can also be solved by a modification of the Boolean sum:

$$P \oplus Q := P + Q - \alpha(u, w)P \cdot Q - \beta(u, w)Q \cdot P \qquad (9)$$
$$\text{with } \alpha, \beta \geq 0, \quad \alpha + \beta = 1 \quad \alpha(i, w) = 1,$$
$$\alpha(u, j) = 0, \quad \beta(i, w) = 0, \quad \beta(u, j) = 1$$

Gordon-Coons Patches

In this case, the twist part has the following form:

$$\begin{pmatrix} \alpha X_{uw}(0,0) + \beta X_{wu}(0,0) & \alpha X_{uw}(0,1) + \beta X_{wu}(0,1) \\ \alpha X_{uw}(1,0) + \beta X_{wu}(1,0) & \alpha X_{uw}(1,1) + \beta X_{wu}(1,1) \end{pmatrix}$$

with

$$\alpha(u,w) := \frac{w(1-w)}{u(1-u) + w(1-w)}, \quad \beta(u,w) := \alpha(w,u).$$

Generalizing this method to a C^2-biquintic Coons patch requires the quintic Hermite operators:

$$Q_1X := \sum_{i=0}^{1} H_i(u)X(i,w) + \overline{H}_i(u)X_u(i,w) + \overline{\overline{H}}_i(u)X_{uu}(i,w)$$

$$Q_2X := \sum_{j=0}^{1} H_j(w)X(u,j) + \overline{H}_j(w)X_w(u,j) + \overline{\overline{H}}_j(w)X_{ww}(u,j)$$

Gordon-Coons Patches

The blending functions read:

$$\begin{aligned} H_0(s) &= (1-s)^3(1+3s+6s^2) \\ H_1(s) &= s^3(10-15s+6s^2) \\ H_0'(s) &= s(1-s)^3(1+3s) \\ H_1'(s) &= s^3(1-s)(3s-4) \\ H_0''(s) &= \frac{1}{2}s^2(1-s)^3 \\ H_1''(s) &= \frac{1}{2}s^3(1-s)^2 \end{aligned}$$

$\Rightarrow$ compatibility–corrected biquintic Coons patch

Gordon-Coons Patches

$$
\begin{aligned}
X(u,w) :=& \begin{bmatrix} H_0(u) \\ H_1(u) \\ \overline{H}_0(u) \\ \overline{H}_1(u) \\ \overline{\overline{H}}_0(u) \\ \overline{\overline{H}}_1(u) \end{bmatrix}^T \cdot \begin{bmatrix} X(0,w) \\ X(1,w) \\ X_u(0,w) \\ X_u(1,w) \\ X_{uu}(0,w) \\ X_{uu}(1,w) \end{bmatrix} + \begin{bmatrix} H_0(w) \\ H_1(w) \\ \overline{H}_0(w) \\ \overline{H}_1(w) \\ \overline{\overline{H}}_0(w) \\ \overline{\overline{H}}_1(w) \end{bmatrix} \cdot \begin{bmatrix} X(u,0) \\ X(u,1) \\ X_w(u,0) \\ X_w(u,1) \\ X_{ww}(u,0) \\ X_{ww}(u,1) \end{bmatrix} \\
&- \begin{bmatrix} H_0(u) \\ H_1(u) \\ \overline{H}_0(u) \\ \overline{H}_1(u) \\ \overline{\overline{H}}_0(u) \\ \overline{\overline{H}}_1(u) \end{bmatrix}^T \begin{bmatrix} X(0,0) & X(0,1) & X_w(0,0) & X_w(0,1) & X_{ww}(0,0) & X_{ww}(0,1) \\ X(1,0) & X(1,1) & X_w(1,0) & X_w(1,1) & X_{ww}(1,0) & X_{ww}(1,1) \\ X_u(0,0) & X_u(0,1) & & & & \\ X_u(1,0) & X_u(1,1) & & T_{11} & & T_{12} \\ X_{uu}(0,0) & X_{uu}(0,1) & & T_{21} & & T_{22} \\ X_{uu}(1,0) & X_{uu}(1,1) & & & & \end{bmatrix} \begin{bmatrix} H_0(w) \\ H_1(w) \\ \overline{H}_0(w) \\ \overline{H}_1(w) \\ \overline{\overline{H}}_0(w) \\ \overline{\overline{H}}_1(w) \end{bmatrix}
\end{aligned} \tag{10}
$$

Gordon-Coons Patches

$$T_{11} = \begin{pmatrix} \frac{u^2 X_{uw}(0,0)+w^2 X_{wu}(0,0)}{u^2+w^2} & \frac{u^2 X_{uw}(1,0)+(1-w)^2 X_{wu}(0,1)}{u^2+(1-w)^2} \\ \frac{(1-u)^2 X_{uw}(1,0)+w^2 X_{wu}(1,0)}{(1-u)^2+w^2} & \frac{(1-u)^2 X_{uw}(1,1)+(1-w)^2 X_{wu}(1,1)}{(1-u)^2+(1-w)^2} \end{pmatrix}$$

$$T_{12} = \begin{pmatrix} \frac{u^2 X_{uww}(0,0)+w^2 X_{wwu}(0,0)}{u^2+w^2} & \frac{u^2 X_{uww}(1,0)+(1-w)^2 X_{wwu}(0,1)}{u^2+(1-w)^2} \\ \frac{(1-u)^2 X_{uww}(1,0)+w^2 X_{wwu}(1,0)}{(1-u)^2+w^2} & \frac{(1-u)^2 X_{uww}(1,1)+(1-w)^2 X_{wwu}(1,1)}{(1-u)^2+(1-w)^2} \end{pmatrix}$$

$$T_{21} = \begin{pmatrix} \frac{u^2 X_{uuw}(0,0)+w^2 X_{wuu}(0,0)}{u^2+w^2} & \frac{u^2 X_{uuw}(1,0)+(1-w)^2 X_{wuu}(0,1)}{u^2+(1-w)^2} \\ \frac{(1-u)^2 X_{uuw}(1,0)+w^2 X_{wuu}(1,0)}{(1-u)^2+w^2} & \frac{(1-u)^2 X_{uuw}(1,1)+(1-w)^2 X_{wuu}(1,1)}{(1-u)^2+(1-w)^2} \end{pmatrix}$$

$$T_{22} = \begin{pmatrix} \frac{u^2 X_{uuww}(0,0)+w^2 X_{wwuu}(0,0)}{u^2+w^2} & \frac{u^2 X_{uuww}(1,0)+(1-w)^2 X_{wwuu}(0,1)}{u^2+(1-w)^2} \\ \frac{(1-u)^2 X_{uuww}(1,0)+w^2 X_{wwuu}(1,0)}{(1-u)^2+w^2} & \frac{(1-u)^2 X_{uuww}(1,1)+(1-w)^2 X_{wwuu}(1,1)}{(1-u)^2+(1-w)^2} \end{pmatrix}$$

Gordon-Coons Patches

Instead of the Gregory Square we can also use Nielson's Boolean sum extension with:

$$\alpha(u,w) := \frac{w^2(1-w)^2}{u^2(1-u)^2 + w^2(1-w)^2}, \quad \beta(u,w) := \alpha(w,u)$$

The twist part is then:

$$\begin{aligned}
T_{11} &: \quad \alpha X_{uw}(i,j) + \beta X_{wu}(i,j) := \tilde{X}_{uw}(i,j) \\
T_{12} &: \quad \alpha X_{uww}(i,j) + \beta X_{wwu}(i,j) \\
T_{21} &: \quad \alpha X_{uuw}(i,j) + \beta X_{wuu}(i,j) \\
T_{22} &: \quad \alpha X_{uuww}(i,j) + \beta X_{wwuu}(i,j),
\end{aligned}$$

where $(i,j) \in \{(0,0),(1,0),(0,1),(1,1)\}$.
Assuming smooth patches without dramatic changes in the derivatives of curvature values, we can set X_{uuw}, X_{uww} and X_{uuww} to zero.

Gordon-Coons Patches

modified biquintic Coons patch:

$$\begin{bmatrix} -1 \\ H_0(u) \\ H_1(u) \\ \overline{H}_0(u) \\ \overline{H}_1(u) \\ \overline{\overline{H}}_0(u) \\ \overline{\overline{H}}_1(u) \end{bmatrix}^T \cdot X \cdot \begin{bmatrix} -1 \\ H_0(w) \\ H_1(w) \\ \overline{H}_0(w) \\ \overline{H}_1(w) \\ \overline{\overline{H}}_0(w) \\ \overline{\overline{H}}_1(w) \end{bmatrix}, \text{ with}$$

$$X = \begin{bmatrix} 0 & X(u,0) & X(u,1) & X_w(u,0) & X_w(u,1) & X_{ww}(u,0) & X_{ww}(u,1) \\ X(0,w) & X(0,0) & X(0,1) & X_w(0,0) & X_w(0,1) & X_{ww}(0,0) & X_{ww}(0,1) \\ X(1,w) & X(1,0) & X(1,1) & X_w(1,0) & X_w(1,1) & X_{ww}(1,0) & X_{ww}(1,1) \\ X_u(0,w) & X_u(0,0) & X_u(0,1) & \tilde{X}_{uw}(0,0) & \tilde{X}_{uw}(0,1) & 0 & 0 \\ X_u(1,w) & X_u(1,0) & X_u(1,1) & \tilde{X}_{uw}(1,0) & \tilde{X}_{uw}(1,1) & 0 & 0 \\ X_{uu}(0,w) & X_{uu}(0,0) & X_{uu}(0,1) & 0 & 0 & 0 & 0 \\ X_{uu}(1,w) & X_{uu}(1,0) & X_{uu}(1,1) & 0 & 0 & 0 & 0 \end{bmatrix}$$

Incorporating a Hermite interpolation along and across the boundaries we get the **geometric Coons patch**:

$$\begin{bmatrix} H_0(u) \\ H_1(u) \\ \overline{H}_0(u) \\ \overline{H}_1(u) \\ \overline{\overline{H}}_0(u) \\ \overline{\overline{H}}_1(u) \end{bmatrix}^T \cdot \begin{bmatrix} X(0,0) & X(0,1) & X_w(0,0) & X_w(0,1) & X_{ww}(0,0) & X_{ww}(0,1) \\ X(1,0) & X(1,1) & X_w(1,0) & X_w(1,1) & X_{ww}(1,0) & X_{ww}(1,1) \\ X_u(0,0) & X_u(0,1) & \tilde{X}_{uw}(0,0) & \tilde{X}_{uw}(0,1) & 0 & 0 \\ X_u(1,0) & X_u(1,1) & \tilde{X}_{uw}(1,0) & \tilde{X}_{uw}(1,1) & 0 & 0 \\ X_{uu}(0,0) & X_{uu}(0,1) & 0 & 0 & 0 & 0 \\ X_{uu}(1,0) & X_{uu}(1,1) & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} H_0(w) \\ H_1(w) \\ \overline{H}_0(w) \\ \overline{H}_1(w) \\ \overline{\overline{H}}_0(w) \\ \overline{\overline{H}}_1(w) \end{bmatrix} \tag{11}$$

Data exchange format:

$$X(u,w) = U^T \cdot M \cdot C \cdot M^T \cdot W$$

Matrix M maps the monome basis $\{s^5, s^4, s^3, s^2, s, 1\}$ into the Hermite basis $\{H_0(s), H_1(s), \overline{H}_0(s), \overline{H}_1(s)\overline{\overline{H}}_0(s), \overline{\overline{H}}_1(s)\}$:

$$M := \begin{bmatrix} -6 & 6 & -3 & -3 & -0.5 & 0.5 \\ 15 & -15 & 8 & 7 & 1.5 & -1 \\ -10 & 10 & 6 & -4 & -1.5 & 0.5 \\ 0 & 0 & 0 & 0 & 0.5 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Twist-Input

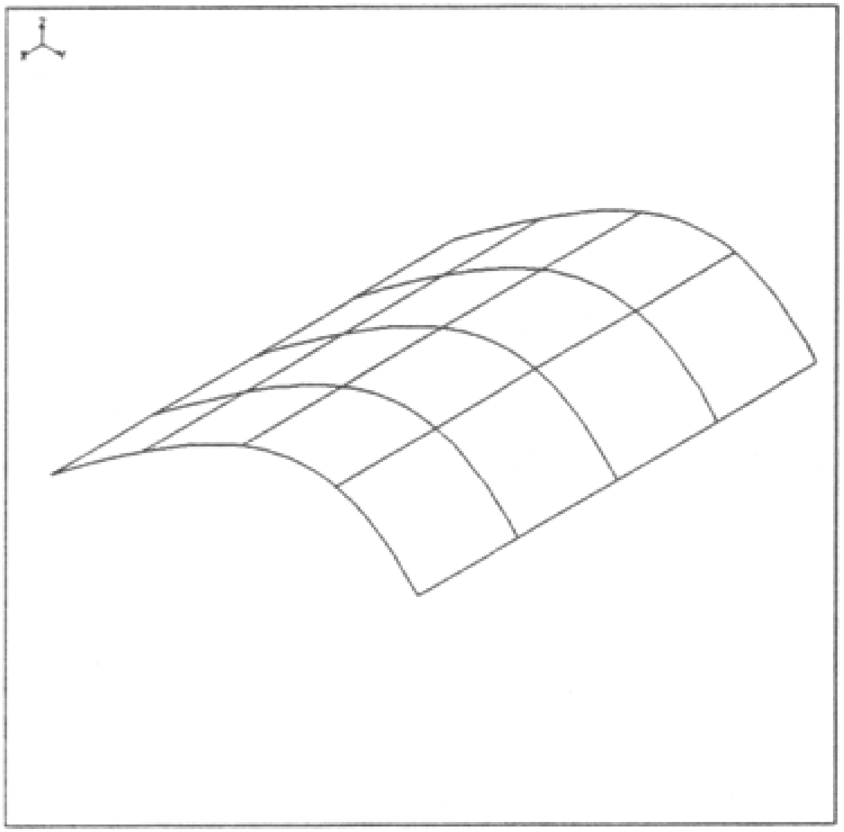

Figure: **Zero twist** : this is only fine for sweep surfaces along straight lines (ruled sweeping).

Adini twist

Calculate the bilinear (!) Coons patch through the four boundary curves. Use the twist vectors of this patch as twist input.
The Adini twists of a ruled sweep surface are zero.

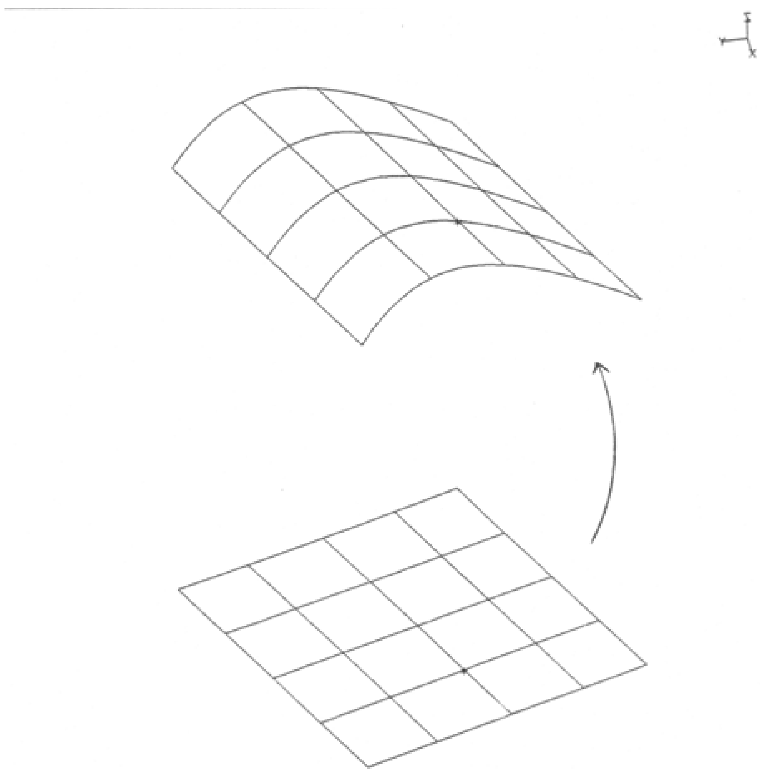

Figure:

Bessel twist

Calculate the biquadratic interpolant $X(u,w) := \sum_{p,q=0}^{2} C_{pp} u^p w^q$ for the 9 points $X(u_{i+r}, w_{j+s}),\ \ i,j \in \{-1,0,1\}$. Use the twists at (u_r, w_s) of this patch.
The Bessel twist is the bilinear interpolant of the Adini twists of the four bilinear Coons patches of the 9 points!

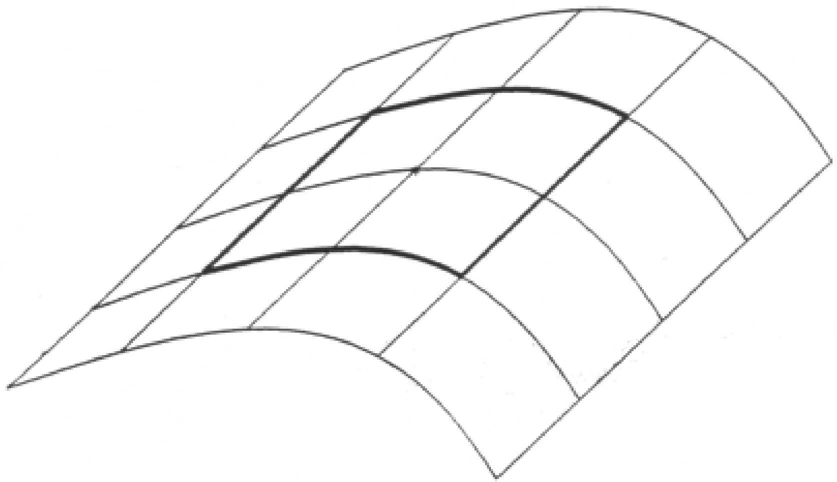

Figure: 1 Bessel twist: 4 Adini twists

Hagen twist

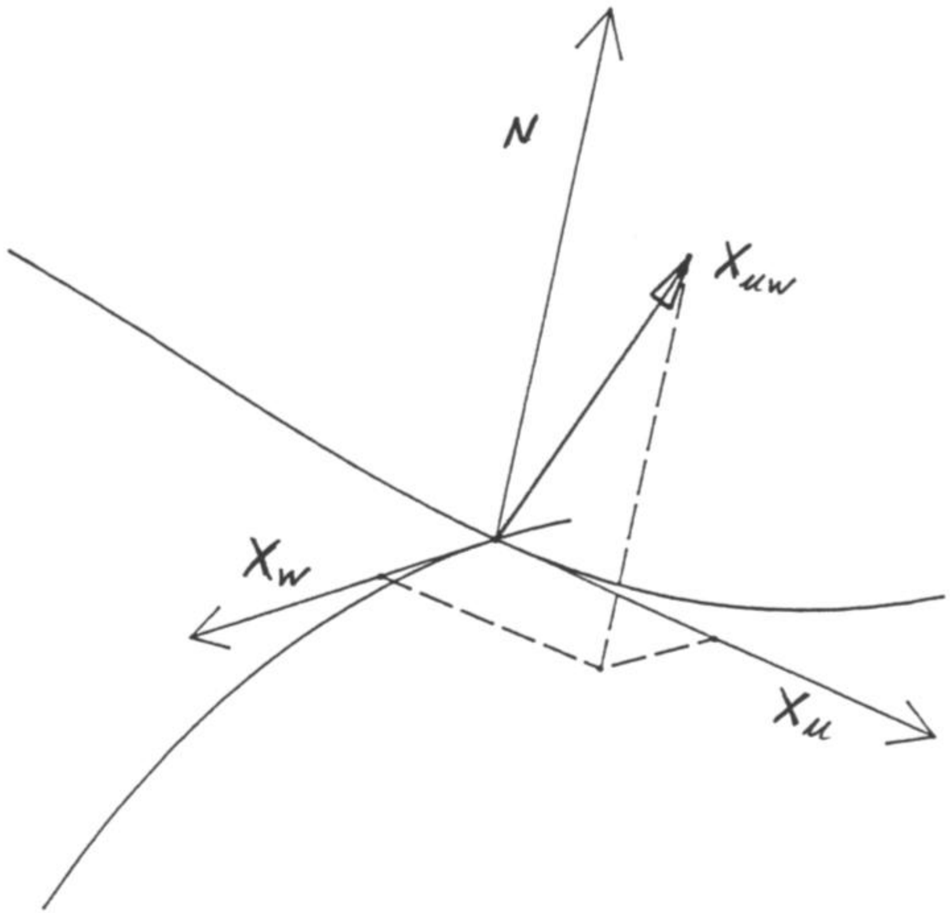

Figure: Idea of the Hagen twist

Gordon-Coons Patches

Twist vector in the Gauss frame:

$$
\begin{aligned}
X_{uw} &= < N, X_{uw} > \cdot N + < X_u, X_{uw} > \cdot X_u + < X_w, X_{uw} > \cdot X_w \\
&= h_{12} \cdot N + \Gamma'_{12} \cdot X_u + \Gamma^2_{12} \cdot X_w
\end{aligned}
$$

The Christoffel symbols Γ^k_{ij} are completely determined by the first fundamental form. The second fundamental form "takes care" of the curvature situation.

$h_{12} =< N, X_{uw} >$ is a component function of the second fundamental form.

Energy method

The functional $\int_S (k_1^2 + k_2^2) dS$ (k_1 and k_2 are the maximal and minimal normal section curvature) is a standard smoothing criterion for surfaces in CAD/CAM. This functional "describes" the bending energy in a small plate s which has no inner deformation.

$$
\begin{aligned}
G &:= \int_S (k_1^2 + k_2^2)\,\mathrm{d}S \\
&= \int_a \frac{(g_{11}h_{22} - 2g_{12}h_{12} + g_{22}h_{11})^2 - 2gh}{g^2} \sqrt{g}\,\mathrm{d}u\mathrm{d}w \\
&= \int\int F(u, w, h_{12}(u, w))\,\mathrm{d}u\mathrm{d}w
\end{aligned}
$$

The Euler equation:

$$\frac{\partial F}{\partial h_{12}} = 2h_{12}(2g - 4g_{12}^2) - 4g_{12}(g_{11}h_{22} + g_{22}h_{11}) = 0$$

gives a necessary condition for the energy minimum:

$$h_{12}(u, w) = \frac{g_{12}(gh_{22} + g_{22}h_{11})}{g + 2g_{12}^2} = 2g_{12} \cdot H$$

(H is the mean curvature of the surface.) In the case of an orthogonal network ($g_{12} = 0$), we get $h_{12} = 0$. This means the orthogonal parameter lines are lines of curvature!

Gordon Surfaces

The Gordon scheme can also be used to solve the network–interpolation problem.

Network Interpolation Problem:

Input: Two sets of blending functions $\{\Phi_i(s)\}_{i=0}^{M}$ and $\{\Psi_j(k)\}_{j=0}^{N}$,

with $\Phi_i(s) = \begin{cases} 0 & s \neq i \\ 1 & s = i \end{cases}$ and $\Psi_j(k) = \begin{cases} 0 & k \neq j \\ 1 & k = j \end{cases}$.

Two sets of vector-valued functions $\{g_i(t)\}_{i=0}^{M}$, $\{f_j(s)\}_{j=0}^{N}$, with $g_i(t_i) = f_j(s_i) \forall i, j$.

Gordon-Coons Patches

Solution to the network interpolation problem:

$$X(s,t) := \sum_{i=0}^{M} g_i(t)\Phi_i(s) + \sum_{J=0}^{N} f_j(s)\Psi_j(t) - \sum_{i=0}^{M}\sum_{j=0}^{N} X_{ij}\Phi_i(s)\Psi_j(t)$$

with $X(s_k, t) = g_k(t),\ k = 0, 1, \dots, M,$

$X(s, t_l) = f_l(s),\ l = 0, 1, \dots, N$

and $X_{ij} = X(s_i, t_j).$

remarks

This is a global scheme if we use blending functions without local support.
Interpolating tangent- and curvature values, we run into twist problems again.

Gordon-Coons Patches

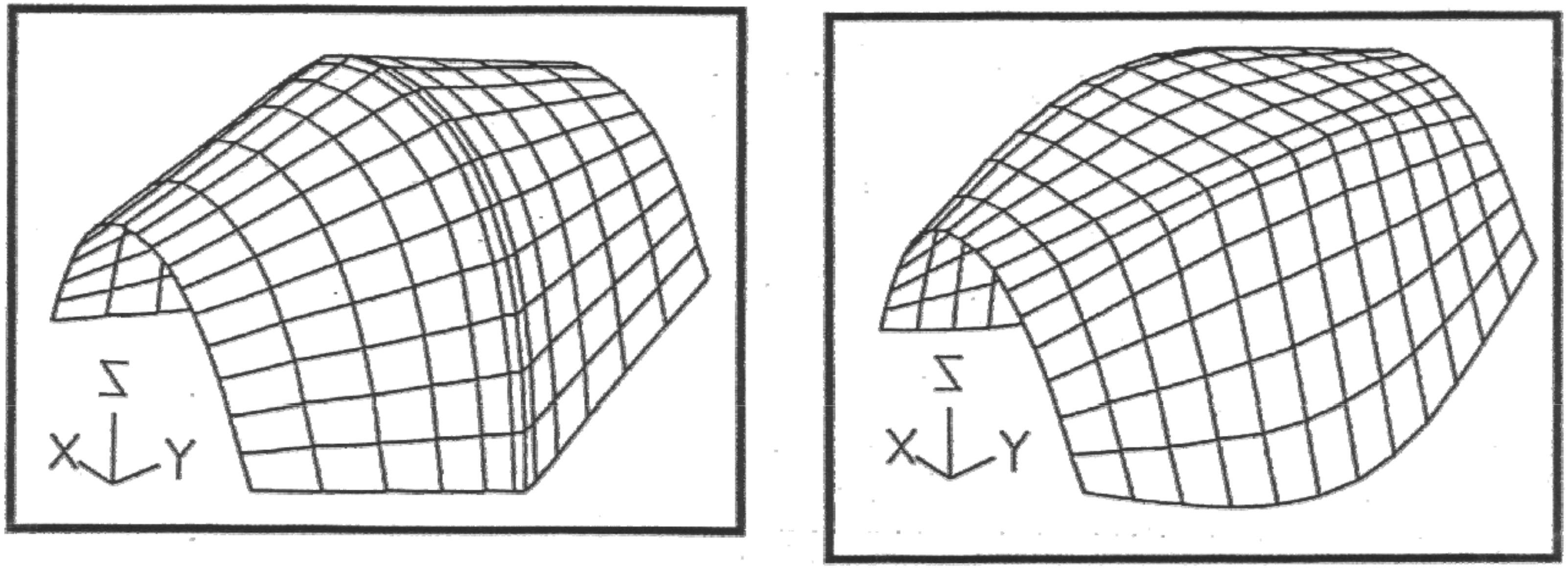

Figure: The effects of applying tension across (left) in comparison to along a curve

Gordon-Coons Patches

The transformation between Coons and Bézier patches can be done by matrix operations:

$$
\begin{aligned}
X(u,w) &= u^T \cdot M \cdot C \cdot M^T \cdot w \\
X(u,w) &= u^T \cdot T \cdot B \cdot T^T \cdot w
\end{aligned}
$$

$$
\begin{aligned}
X_{uw}(u_k, w_l) &= \alpha\beta(b_{00} - b_{10} - b_{01} + b_{11}) =: BP(u_k, w_l) \\
X_{uw}(u_{k+1}, w_l) &= \alpha\beta(b_{20} - b_{30} - b_{21} + b_{31}) =: BP(u_{k+1}, w_l) \\
X_{uw}(u_k, w_{l+1}) &= \alpha\beta(b_{02} - b_{12} - b_{03} + b_{11}) =: BP(u_k, w_{l+1}) \\
X_{uw}(u_{k+1}, w_{l+1}) &= \alpha\beta(b_{22} - b_{32} - b_{23} + b_{33}) =: BP(u_{k+1}, w_{l+1})
\end{aligned}
$$

with $\alpha = 3/\Delta u_k$, $\beta = 3/\Delta w_l$, $\Delta u_k = n_{k+1} - u_k$, $\Delta w_l = w_{l+1} - w_l$

Gordon-Coons Patches

$$\begin{bmatrix} b_{03} & b_{13} & b_{23} & b_{33} \\ b_{02} & b_{12} & b_{22} & b_{32} \\ b_{01} & b_{11} & b_{21} & b_{31} \\ b_{00} & b_{10} & b_{20} & b_{30} \end{bmatrix}$$

$$\updownarrow$$

$$\begin{bmatrix} b_{00} & b_{03} & \beta(b_{01} - b_{00}) & \beta(b_{03} - b_{02}) \\ b_{30} & b_{33} & \beta(b_{31} - b_{30}) & \beta(b_{33} - b_{32}) \\ \alpha(b_{10} - b_{00}) & \alpha(b_{03} - b_{13}) & BP(u_k, w_l) & BP(u_{k+1}, w_l) \\ \alpha(b_{30} - b_{20}) & \alpha(b_{33} - b_{23}) & BP(u_k, w_{l+1}) & BP(u_{k+1}, w_{l+1}) \end{bmatrix}$$

Gordon-Coons Patches

Bilinear Coons patch:

$$X(u,w) = \frac{u_{l+1}-u}{\Delta u_k} X(u_k, w) - \frac{u_k - u}{\Delta u_k} X(u_{k+1}, w)$$

$$+ \frac{w_{l+1}-w}{\Delta w} X(u, w_l) - \frac{w_l - w}{\Delta w_l} X(u, w_{l+1})$$

$$+ \frac{(u_k - u)(w_{l+1} - w)}{\Delta u_k \Delta w_l} X(u_{k+1}, w_l) - \frac{(u_{k+1} - u)(w_{l+1} - w)}{\Delta u_k \Delta w_l} X(u_k, w_l)$$

$$+ \frac{(u_{k+1} - u)(w_l - w)}{\Delta u_k \Delta w_l} X(u_{k+1}, w_{l+1}) - \frac{(u_k - u)(w_l - w)}{\Delta u_k \Delta w_l} X(u_{k+1}, w_{l+1}).$$

Calculate the twist vectors in the corner points:

Adini twists

$$A_{uw}(u_k, w_l) = \frac{1}{\Delta w_l}(X_u(u_k, w_{l+1}) - X_u(u_k, w_l)) + \frac{1}{\Delta u_k}(X_w(u_{k+1}, w_l) - X_w(u_k, w_l)) - A_{kl}$$

$$A_{uw}(u_k, w_{l+1}) = \frac{1}{\Delta w_l}(X_u(u_k, w_{l+1}) - X_u(u_k, w_l)) + \frac{1}{\Delta u_k}(X_w(u_{k+1}, w_{l+1}) - X_w(u_k, w_{l+1})) - A_{kl}$$

$$A_{uw}(u_{k+1}, w_l) = \frac{1}{\Delta w_l}(X_u(u_{k+1}, w_{l+1}) - X_u(u_{k+1}, w_l)) + \frac{1}{\Delta u_k}(X_w(u_{k+1}, w_l) - X_w(u_k, w_l)) - A_{kl}$$

$$A_{uw}(u_{k+1}, w_{l+1} = \frac{1}{\Delta w_l}(X_u(u_{k+1}, w_{l+1}) - X_u(u_{k+1}, w_l)) \frac{1}{\Delta u_k}(X_w(u_{k+1}, w_{l+1}) - X_w(u_k, w_{l+1})) - A_{kl}$$

with $A_{kl} := \frac{1}{\Delta_{uk}\Delta_{wl}}(X(u_k, w_l) - X(u_{k+1}, w_l) - X(u_k, w_{l+1}) + X(u_{k+1}, w_{l+1}))$.

Gordon-Coons Patches

Coons patches $\Leftrightarrow$ Bézier patches:

$$
\begin{aligned}
b_{00} &= X(u_j, w_j) \\
b_{01} &= X(u_i, w_j) + \tfrac{\Delta u_i}{3} X_u(u_i, w_j) \\
b_{10} &= X(u_i, w_j) + \tfrac{\Delta w_j}{3} X_w(u_i, w_j) \\
b_{11} &= -b_{00} + b_{10} + b_{01} + \tfrac{\Delta u_i \Delta w_j}{9} X_{uw}(u_i, w_j)
\end{aligned}
$$

The other 12 Bézier points we get out of similar equations. Only the 4 “inner” Bézier points b_{11}, b_{21}, b_{12} and b_{22} are “touched by” the twist vectors. This is fine for smoothing, since the 12 outer Bézier points influence the boundary curves and we don’t want to change those.

Gordon-Coons Patches

As twist input we can use the Adini twists or ano other twists.

$$\begin{aligned}
b_{11} &= -b_{00} + b_{10} + b_{01} + \frac{\Delta_{uk}\Delta_{wl}}{9} A_{uw}(u_k, w_l) \\
b_{11} &= -b_{03} + b_{02} + b_{13} + \frac{\Delta_{uk}\Delta_{wl}}{9} A_{uw}(u_k, w_{l+1}) \\
b_{21} &= -b_{30} + b_{20} + b_{31} + \frac{\Delta_{uk}\Delta_{wl}}{9} A_{uw}(u_k, w_l) \\
b_{22} &= -b_{33} + b_{32} + b_{23} + \frac{\Delta_{uk}\Delta_{wl}}{9} A_{uw}(u_k, w_{l+1})
\end{aligned}$$

An often used method is the stiffness degree concept (Hagen):

$$X_{uw} = \alpha Ad + \beta h_{12}^{\text{opt.}} \cdot N$$

with $\alpha + \beta = 1$, $\alpha > 0$, $\beta \geq 0$. β/α is called stiffness degree.