On the CAD-compatible conversion of S-patches

Péter Salvi

Budapest University of Technology and Economics

WAIT 2019

Budapest, January 24th
Outline

1 Introduction
   - Motivation
   - Previous work

2 Simplexes & S-patches
   - Simplexes
   - S-patches

3 Conversion
   - Conversion to quadrilateral S-patch
   - Conversion to tensor product form

4 Conclusion
   - Example
   - Discussion

On the CAD-compatible conversion of S-patches
Multi-sided surfaces in CAD software

- Standard surface representations:
  - Tensor-product Bézier surface
  - Tensor-product B-spline surface
  - Tensor-product NURBS surface
- No standard multi-sided representation
- Conversion to tensor-product patches
  - Trimming
    - Parameterization issues
    - Asymmetric
    - Not watertight
  - Central split
    - Loosely defined dividing curves
    - Only $C^0$ or $G^1$ continuity
Solution

- **Exact** tensor product conversion
- Trimmed rational Bézier surface
  - Only polynomial (Bézier) boundaries
  - Trimming curves ⇒ lines in the domain
- Native $n$-sided representation
  - S-patch
  - Generalization of Bézier curves & triangles
  - Suitable for $G^1$ hole filling [1]

[1] P. Salvi, *$G^1$ hole filling with S-patches made easy.*
In: Proceedings of the 12th Conference of the Hungarian Association for Image Processing and Pattern Recognition, 2019 (accepted).
S-patches & simplexes

- [1989, Loop & DeRose] A multi-sided generalization of Bézier surfaces
  - The original S-patch publication
  - Contains *theoretical results* on the tensor product conversion
  - Missing from the description of the algorithm:
    - Composition of rational Bézier simplexes
    - Blossom of Wachspress coordinates

- [1987, Ramshaw] Blossoming: A connect-the-dots approach to splines

- [1988, DeRose] Composing Bézier simplexes

- [1993, DeRose et al.] Functional composition algorithms via blossoming
Simplexes in $nD$

- $(n + 1)$ points in $nD$
- Let $V_i$ denote these points
- Any $nD$ point is uniquely expressed by the affine combination of $V_i$:

$$ p = \sum_{i=1}^{n} \lambda_i V_i \quad \text{with} \quad \sum_{i=1}^{n} \lambda_i = 1 $$

- $\lambda_i$ are the barycentric coordinates of $p$ relative to the simplex

(images from Wikipedia)
Bézier curve

Let's look at the equation of a Bézier curve:

\[
C(u) = \sum_{i=0}^{d} P_i B_i^d(u)
\]
Let's look at the equation of a Bézier curve:

\[
C(u) = \sum_{i=0}^{d} P_i B_i^d(u) = \sum_{i=0}^{d} P_i \frac{d!}{i!(d-i)!} u^i (1 - u)^{d-i}
\]
Bézier curve

Let's look at the equation of a Bézier curve:

\[ C(u) = \sum_{i=0}^{d} P_i B_i^d(u) = \sum_{i=0}^{d} P_i \frac{d!}{i!(d-i)!} u^i (1 - u)^{d-i} \]

Let \( s = (i, d - i) \) and \( \lambda = (u, 1 - u) \).
Bézier curve

Let's look at the equation of a Bézier curve:

\[ C(u) = \sum_{i=0}^{d} P_i B_i^d(u) = \sum_{i=0}^{d} P_i \frac{d!}{i!(d-i)!} u^i (1 - u)^{d-i} \]

Let \( s = (i, d - i) \) and \( \lambda = (u, 1 - u) \).

Then

\[ C(\lambda) = \sum_{s} P_s \frac{d!}{s_1! s_2!} \lambda_1^{s_1} \lambda_2^{s_2} \]
Bézier triangle

Now let’s look at the equation of a Bézier triangle:

$$T(\lambda) = \sum_s P_s \frac{d!}{s_1!s_2!s_3!} \lambda_1^{s_1} \lambda_2^{s_2} \lambda_3^{s_3} = \sum_s P_s B^d_s(\lambda)$$

- $s = (s_1, s_2, s_3)$ with $s_i \geq 0$ and $s_1 + s_2 + s_3 = d$
- $\lambda = (\lambda_1, \lambda_2, \lambda_3)$ barycentric coordinates of a 2D point relative to the domain triangle (simplex)

Did you know?

This was Paul de Casteljau’s generalization of Bézier curves.
- “Bézier” curves were also his invention
- Tensor product surfaces were invented by Pierre Bézier
- de Casteljau worked at Citroën, while Bézier at Renault
Bézier simplex

- The logical generalization to \((n - 1)\) dimensions:

\[
S(\lambda) = \sum_{s} P_{s} \frac{d!}{\prod_{i=1}^{n} s_i!} \prod_{i=1}^{n} \lambda_{i}^{s_{i}} = \sum_{s} P_{s} B_{s}^{d}(\lambda)
\]

- \(s = (s_1, s_2, \ldots, s_n)\) with \(s_i \geq 0\) and \(\sum_{i=1}^{n} s_i = d\)

- \(\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_n)\) barycentric coordinates of an \((n - 1)D\) point relative to the domain simplex

Note

Bézier simplexes are mappings, not geometric entities!
S-patches as Bézier simplexes

- S-patch equation ($n$ sides, depth $d$):

$$S(\lambda) = \sum_{s} P_s \frac{d!}{\prod_{i=1}^{n} s_i!} \prod_{i=1}^{n} \lambda_i^{s_i} = \sum_{s} P_s B_s^d(\lambda)$$

- Domain for an $n$-sided S-patch:
  - Regular $n$-sided polygon (in 2D)
- Domain for an $(n-1)$-dimensional Bézier simplex:
  - An $(n-1)$-dimensional simplex ($n$ barycentric coordinates)
- Needed:
  - Mapping from an $n$-sided polygon to $n$ barycentric coordinates
  - Generalized barycentric coordinates
    - E.g. Wachspress, mean value, etc.
  - Defines an embedding in the $(n-1)$-dimensional simplex
**Control structure**

- Very complex – many control points, hard to use manually
- Boundary control points define degree $d$ Bézier curves
- Adjacent control points have shifted labels, e.g. $21000 \rightarrow 30000, 11001, 20100, 12000$
Overview

Claim 6.4 in [1989, Loop & DeRose]

For every $m$-sided regular S-patch of depth $d$, there exists an equivalent $n$-sided regular S-patch of depth $d(m - 2)$.

Lemma 6.2 in [1989, Loop & DeRose]

For every 4-sided regular S-patch of depth $d$, there exists an equivalent tensor product Bézier patch of degree $d$.

1. Convert the $n$-sided S-patch of depth $d$
to a quadrilateral S-patch of depth $d(n - 2)$.

2. Convert the quadrilateral S-patch to a tensor product Bézier patch of degree $d(n - 2)$. 

P. Salvi

On the CAD-compatible conversion of S-patches

BME
Conversion to quadrilateral S-patch

Conversion as simplex composition

- Wachspress coordinates on an $n$-sided polygon
  - ... have a Bézier simplex form (denoted by $W_n$)
  - ... are *pseudoaffine* (have an affine left inverse $W_n^{-1}$)
- Mapping from the domain polygon to a 3D point:

  \[ S \circ W_n \]
Conversion as simplex composition

- Wachspress coordinates on an \( n \)-sided polygon
  - \( \ldots \) have a Bézier simplex form (denoted by \( W_n \))
  - \( \ldots \) are pseudoaffine (have an affine left inverse \( W_n^{-1} \))
- Mapping from the domain polygon to a 3D point:

\[
S \circ W_n = S \circ W_n \circ (W_4^{-1} \circ W_4)
\]
Conversion as simplex composition

- Wachspress coordinates on an $n$-sided polygon
  - ... have a Bézier simplex form (denoted by $W_n$)
  - ... are pseudoaffine (have an affine left inverse $W_n^{-1}$)

- Mapping from the domain polygon to a 3D point:

$$S \circ W_n = S \circ W_n \circ (W_4^{-1} \circ W_4) = (S \circ W_n \circ W_4^{-1}) \circ W_4$$
Conversion as simplex composition

- Wachspress coordinates on an $n$-sided polygon
  - ... have a Bézier simplex form (denoted by $W_n$)
  - ... are pseudoaffine (have an affine left inverse $W_n^{-1}$)
- Mapping from the domain polygon to a 3D point:

$$S \circ W_n = S \circ W_n \circ (W_4^{-1} \circ W_4) = (S \circ W_n \circ W_4^{-1}) \circ W_4$$

- The 4-sided formulation is the composition of 3 simplexes:
  - $W_4^{-1}$: defined by the vertices of the rectangular domain
  - $S$: the S-patch (with homogenized control points)
  - $W_n$: ??? [a rational Bézier simplex of degree $n - 2$]
- Composition:
  - Two algorithms (simple vs. efficient) [see the paper]
Determining the control points of $W_n$ – homogenization

$$\lambda_i(p) = \prod_{j \neq i-1,i} D_j(p) / \sum_{k=1}^{n} \prod_{j \neq k-1,k} D_j(p)$$

- $D_j(p)$ is the signed distance of $p$ from the $j$-th side
- Rational expression $\Rightarrow$ homogenized coordinates
  - Use the barycentric coordinates as “normal” coordinates
  - $(x, y, z) \equiv (wx, wy, wz, w(1 - x - y - z))$
- Homogenized form of $W_n$:

$$\left\{ \prod_{j \neq i-1,i} D_j(p) \right\}$$
Determining the control points of $W_n$ – polarization

For any homogeneous polynomial $Q(u)$ of degree $d$, $\exists q$ s.t.

\[
q(u_1, \ldots, u_d) = q(u_{\pi_1}, \ldots, u_{\pi_d}),
\]
\[
q(u_1, \ldots, \alpha u_{k_1} + \beta u_{k_2}, \ldots, u_d) = \alpha q(u_1, \ldots, u_{k_1}, \ldots, u_d) + \beta q(u_1, \ldots, u_{k_2}, \ldots, u_d),
\]
\[
q(u, \ldots, u) = Q(u).
\]

Then $q$ is called the blossom of $Q$.
The control points of its Bézier simplex form are

\[
P_s^Q = q(V_1, \ldots, V_1, V_2, \ldots, V_2, \ldots, V_n, \ldots, V_n),
\]

where $V_i$ are the vertices of the simplex.
Conversion to quadrilateral S-patch

Determining the control points of $W_n$ – blossom

- The blossom of $W_n$ is
  
  $$q(p_1, \ldots, p_{n-2})_i = \frac{1}{(n-2)!} \cdot \sum_{\pi \in \Pi(n-2)} \prod_{k=1}^{n-2} D_j(p_{\pi_k})$$

- $\Pi(n-2)$ is the set of permutations of $\{1, \ldots, n-2\}$
- $k$ runs from 1 to $n-2$ while $j$ from 1 to $n$ skipping $i-1$ and $i$

- With this, the control points can be computed
- Simplex composition gives the quadrilateral S-patch
- Convert to “normal” homogeneous coordinates $(wx, wy, wz, w)$
An 4-sided S-patch of depth $d$ can be represented as

$$\hat{S}(u, v) = \sum_{i=0}^{d} \sum_{j=0}^{d} C_{ij} B_i^d(u) B_j^d(v),$$

where

$$C_{ij} = \sum_{s} \frac{\binom{d}{s}}{\binom{d}{i} \binom{d}{j}} P_s.$$
Converting a 5-sided patch – control net
Converting a 5-sided patch – contours
Converting a 5-sided patch – trimmed tensor product
Converting a 5-sided patch – untrimmed tensor product
Limitations

- **Efficiency**
  - $n = 5, d = 8$ took $> 5$ minutes on a modern machine
    (How long would it have taken in 1989?)
  - Much faster algorithm is developed (see our upcoming paper)
- **3-sided patches**
  - For Bézier triangles, the resulting patch is not rational
  - But there are simple alternative methods, e.g. [1992, Warren]
- **Control net quality**
  - Singularities on a circle around the domain
    - Denominator of Wachspress coordinates vanishes
  - Unstable control points near the corners
- **Conclusion**
  - The algorithm works, but it is not practical
Any questions?

Thank you for your attention.