N-sided Bézier Surfaces for Computer-Aided Geometric Design

Tamás Várady

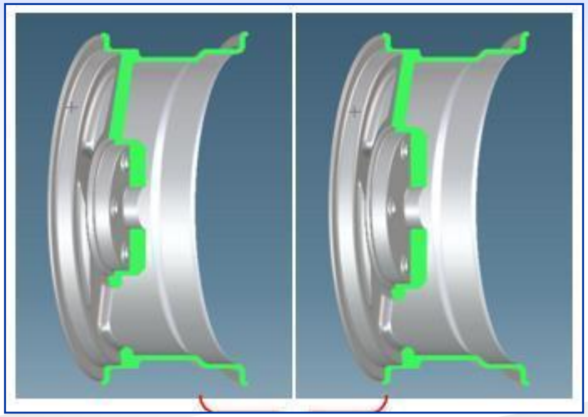
Mathematical Modelling Seminar, 8 Oct 2019

Outline

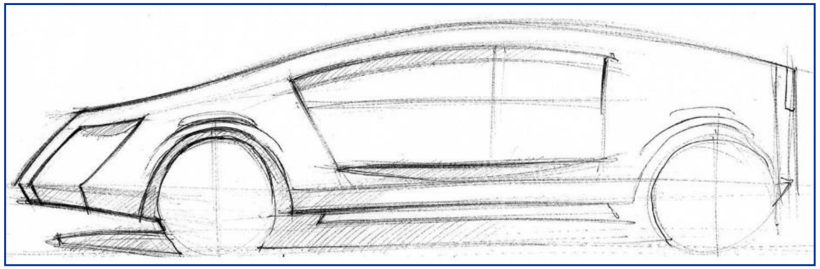
1. Computer-Aided Geometric Design
2. Bézier curves and tensor product Bézier surfaces
3. General topology curve networks
4. The Generalized Bézier patch
    - 3.1. Control-point structure
    - 3.2. Parameterization
    - 3.3. Blending functions and continuity
    - 3.4. Degree elevation
5. Examples

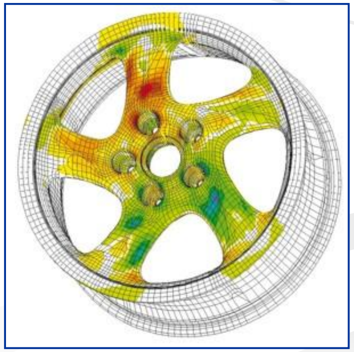
Computer-Aided Geometric Design$_1$

- digital models
- real world objects and virtual reality
- computer algorithms
- wide range of applications



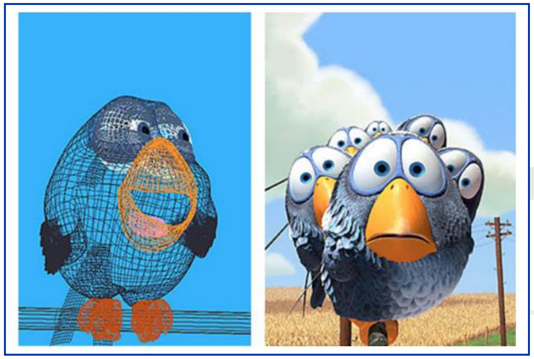
- styling, computer-aided engineering
- reconstruction from measured data
- visualization
- various computations, FE analysis
- manufacturing

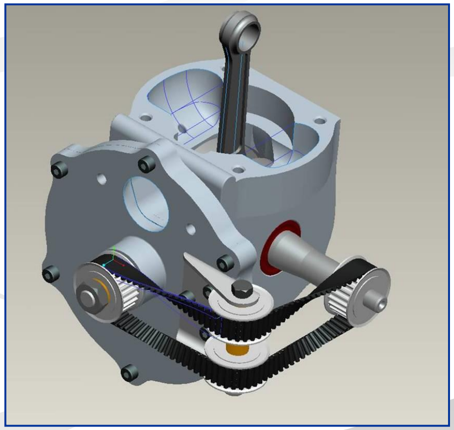


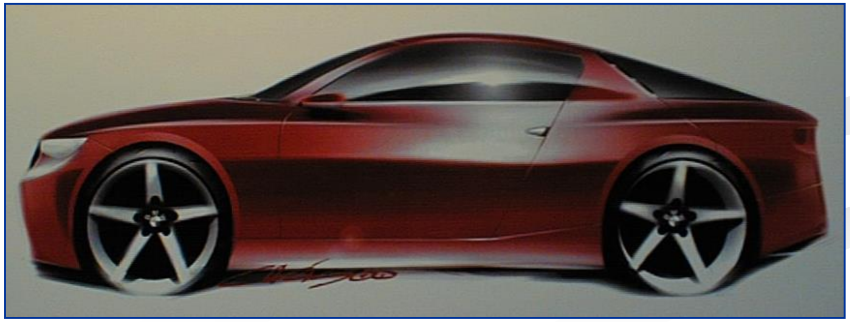


Computer-Aided Geometric Design$_2$

- how to design complex geometric shapes?
- what sort of curve and surface representations are needed?
- how to create, modify and evaluate models?
- what is the meaning of regular and free-form?
- why you sense a 3D curve or glossy surface ”nice” or poor quality?



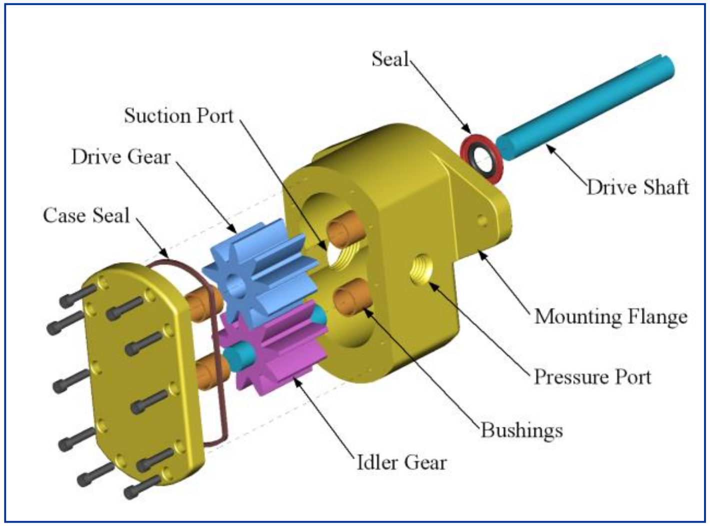


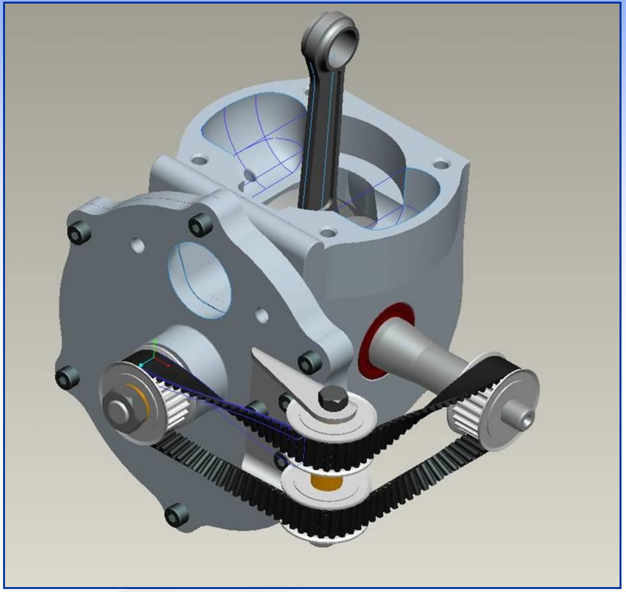


Classifying objects$_1$

Engineering objects

– functional and aesthetic requirements

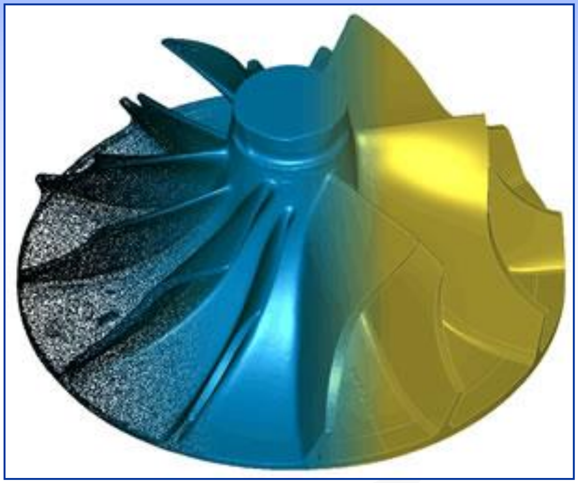




regular surfaces

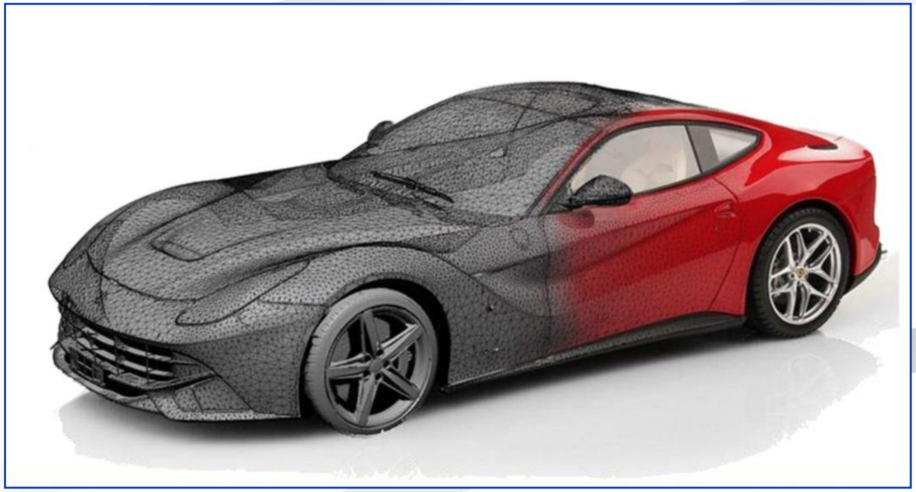
- F(x,y,z) = 0
- implicit equations
- half-spaces

Classifying objects_2



free-form surfaces

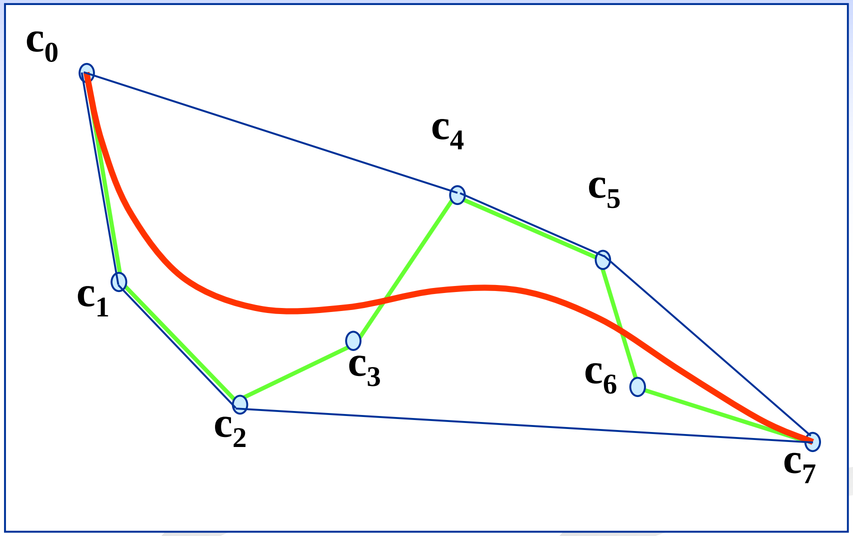
- r(u,v)
- parametric equations
- mapping a 2D domain to 3D



Bézier curves

a degree n Bézier curve:

$$\mathbf{r}(t) = \sum_{k=0}^{n} \mathbf{c}_k B_k^n(t),\ t \in [0,1]$$



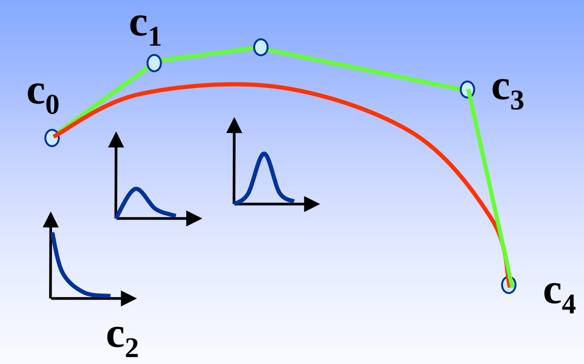
- control polygon: $\mathbf{c}_0, \mathbf{c}_1, \ldots \mathbf{c}_n$
- intuitive design !!!
- endpoint interpolation:

$$\mathbf{r}(0) = \mathbf{c}_0,\ \mathbf{r}(1) = \mathbf{c}_n,$$

$$\dot{\mathbf{r}}(0) = n(\mathbf{c}_1 - \mathbf{c}_0),\ \dot{\mathbf{r}}(1) = -n(\mathbf{c}_n - \mathbf{c}_{n-1})$$

- convex combination of the control points
- affine invariance
- convex hull property
- pseudo local control: editing $\boldsymbol{c}_k$ -- strongest effect at k/n
- line reproduction: if the $\boldsymbol{c}_k$–s are collinear, $\mathbf{r}(t)$ is a line
- variation diminishing property
- stability

Bernstein polynomials



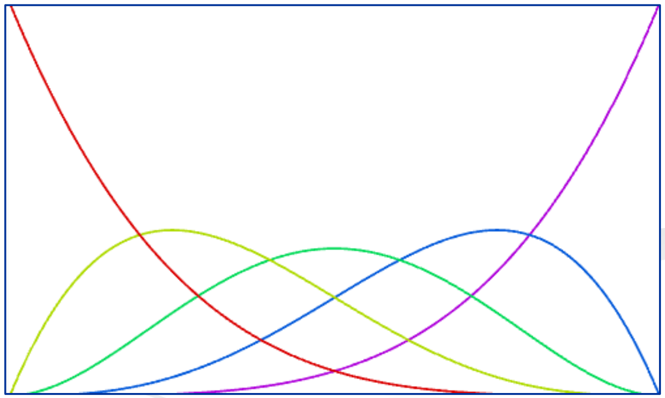
$$((1-t)+t)^n \Rightarrow B_k^n(t) = \binom{n}{k}(1-t)^{n-k}t^k$$

$$\sum_{k=0}^{n} B_k^n(t) = 1, \quad B_k^n(t) \ge 0 \qquad t \in [0,1]$$

$$B_0^n(0) = 1, \quad B_k^n(0) = 0 \quad k \ge 1,$$

$$B_k^n(t) = B_{n-k}^n(1-t),$$

$$B_k^n(t) = t \in [0, \frac{k}{n}] \quad \uparrow, t \in [\frac{k}{n}, 1] \quad \downarrow,$$



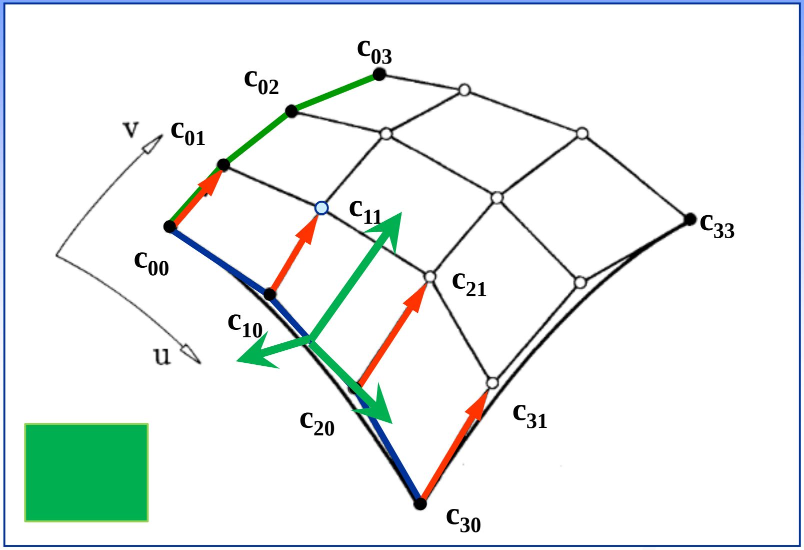
$$n = 1 \rightarrow ((1-t)+t): \quad B_0^1 : (1-t),\ B_1^1 : t$$

$$n = 2 \rightarrow ((1-t)+t)^2: \quad B_0^2 : (1-t)^2,\ B_1^2 : 2(1-t)t,\ B_2^2 : t^2$$

$$n = 3 \rightarrow ((1-t)+t)^3: \quad B_0^3 : (1-t)^3,\ B_1^3 : 3(1-t)^2 t,\ B_2^3 : 3(1-t)t^2,\ B_3^3 : t^3$$

?

Bézier surfaces



- Bézier curve:

$$\mathbf{r}(t) = \sum_{k=0}^{n} \mathbf{c}_k B_k^n(t)$$

- Bézier surface:

$$\mathbf{s}(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} \mathbf{c}_{ij} B_i^n(u) B_j^m(v),$$

$$(u,v) \in [0,1] \times [0,1]$$

$$\mathbf{s}(u,v) = \begin{bmatrix} B_0^n(u) & B_1^n(u) & \dots & B_n^n(u) \end{bmatrix} \begin{bmatrix} \mathbf{c}_{00} & \mathbf{c}_{01} & \dots & \mathbf{c}_{0m} \\ \mathbf{c}_{10} & \mathbf{c}_{11} & \dots & \dots \\ \dots & \dots & \dots & \dots \\ \mathbf{c}_{n0} & \dots & \dots & \mathbf{c}_{nm} \end{bmatrix} \begin{bmatrix} B_0^m(v) \\ B_1^m(v) \\ \dots \\ B_m^m(v) \end{bmatrix}$$

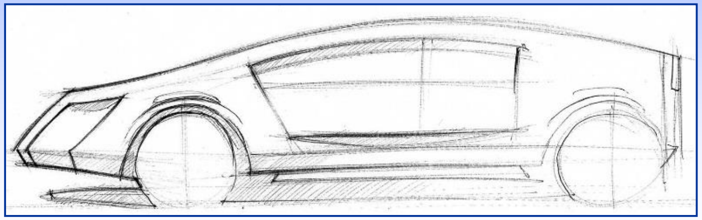
- Boundaries: $\mathbf{s}(u,0) = \sum_{i=0}^{n} \mathbf{c}_{i0} B_i^n(u)$
- Cross-derivatives: $\dot{\mathbf{s}}_v(u,0) = m \sum_{i=0}^{n} (\mathbf{c}_{i1} - \mathbf{c}_{i0}) B_i^n(u)$
- Normal vectors: $\mathbf{N}(u,0) = |\dot{\mathbf{s}}_u(u,0) \times \dot{\mathbf{s}}_v(u,0)|$

?

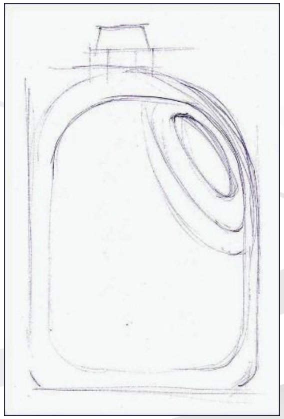
Motivation

- Modeling complex free-form objects
- Aesthetic appearance - crucial
- Quadrilaterals only - not sufficient

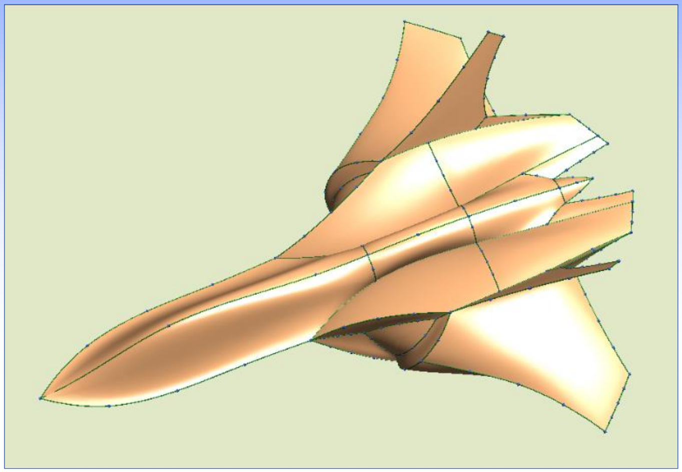
→ general topology surfaces

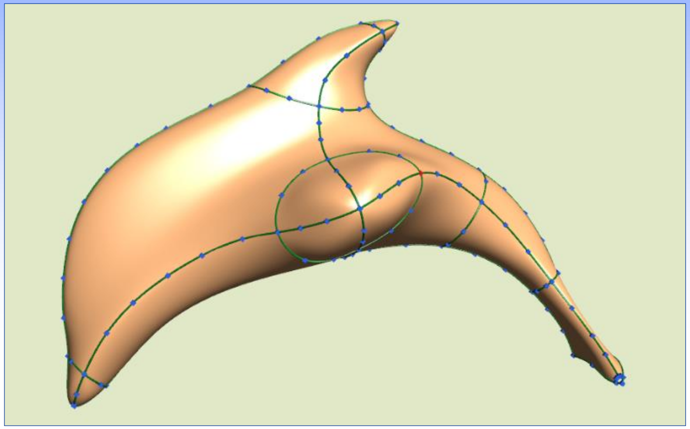


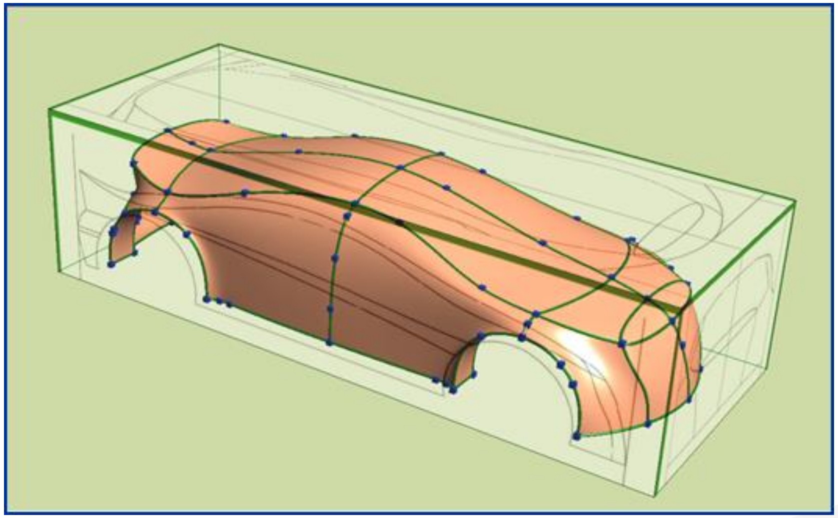


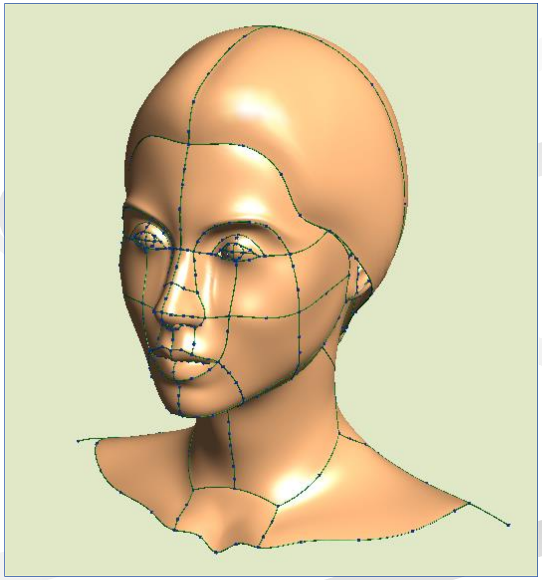



Curve-network based design$_1$



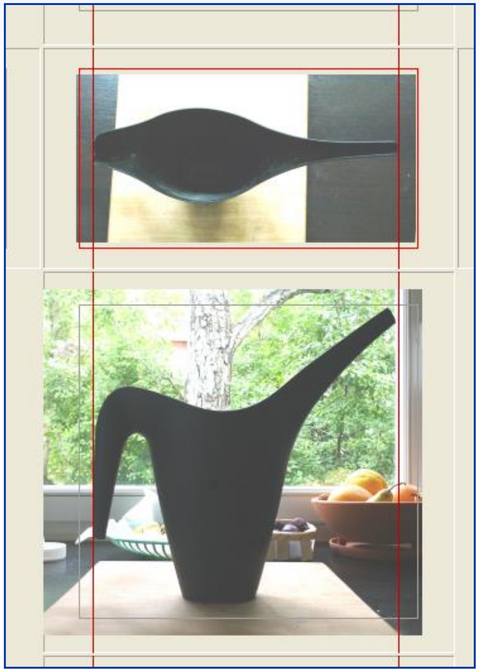


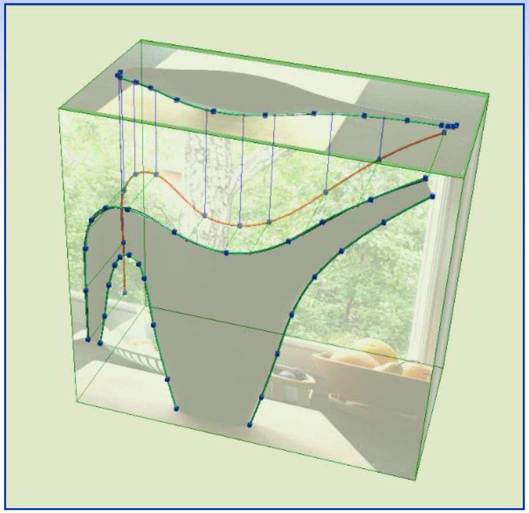


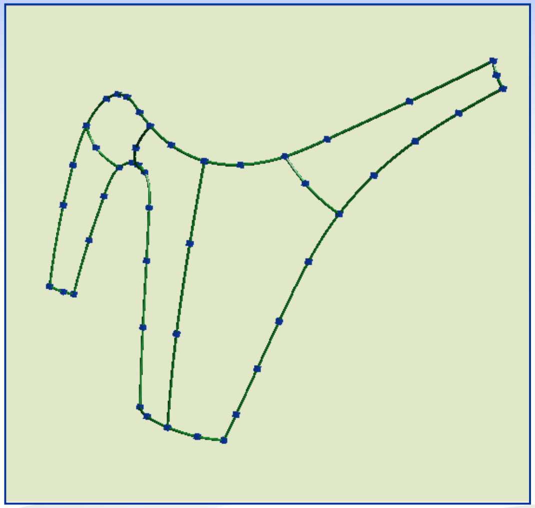


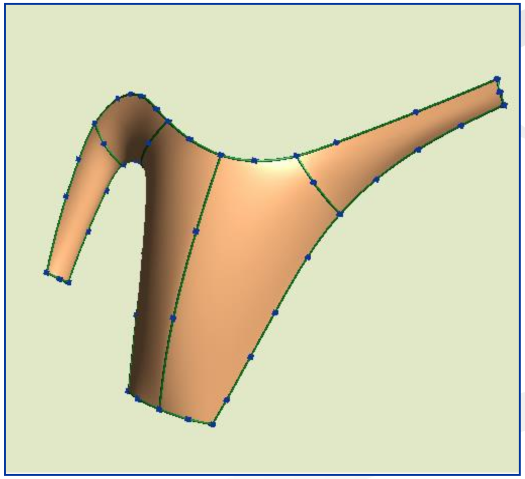
n-sided patches, *n*-valent vertices, T-nodes; smooth connections

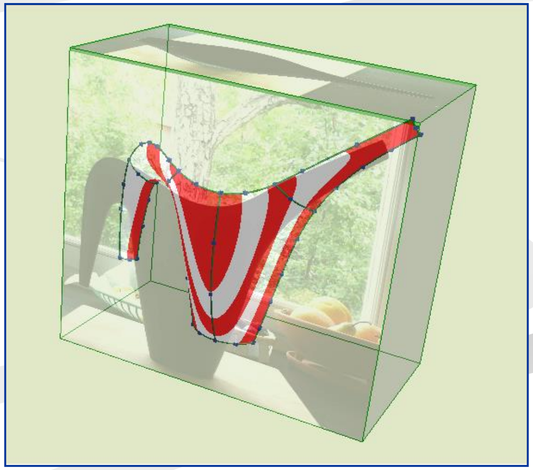
Curve-network based design$_2$



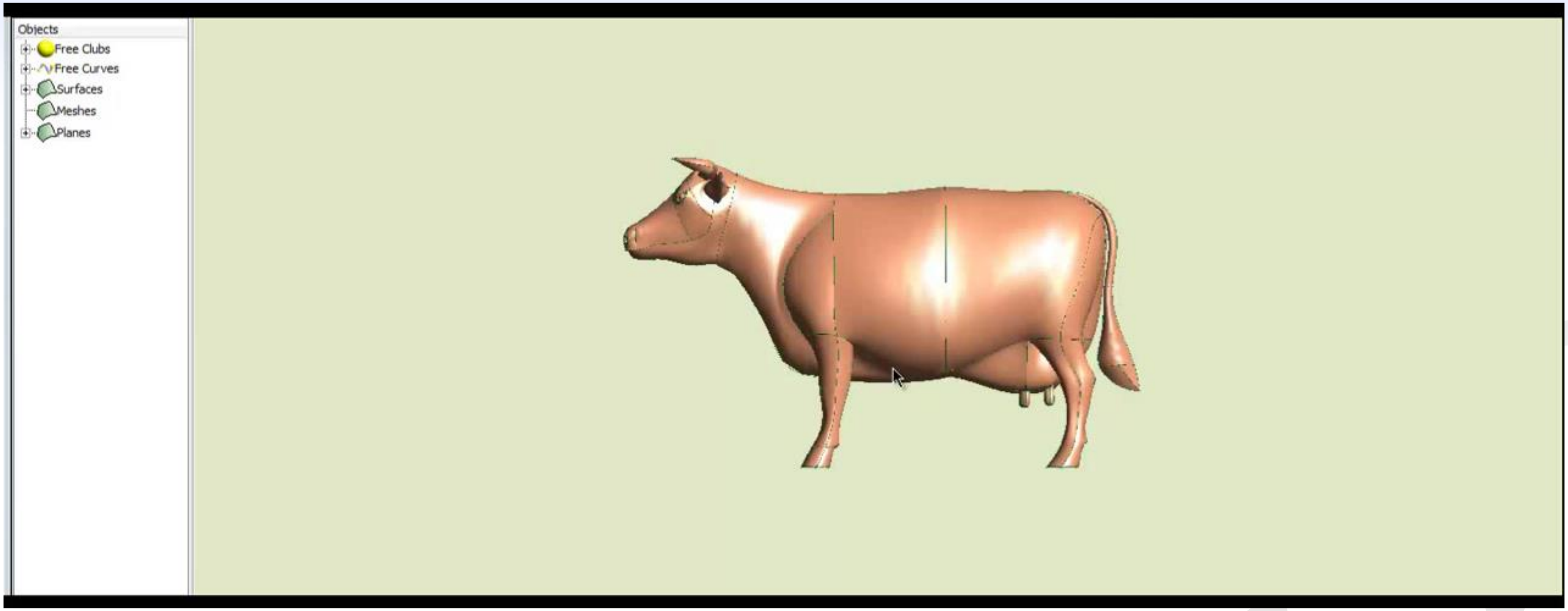




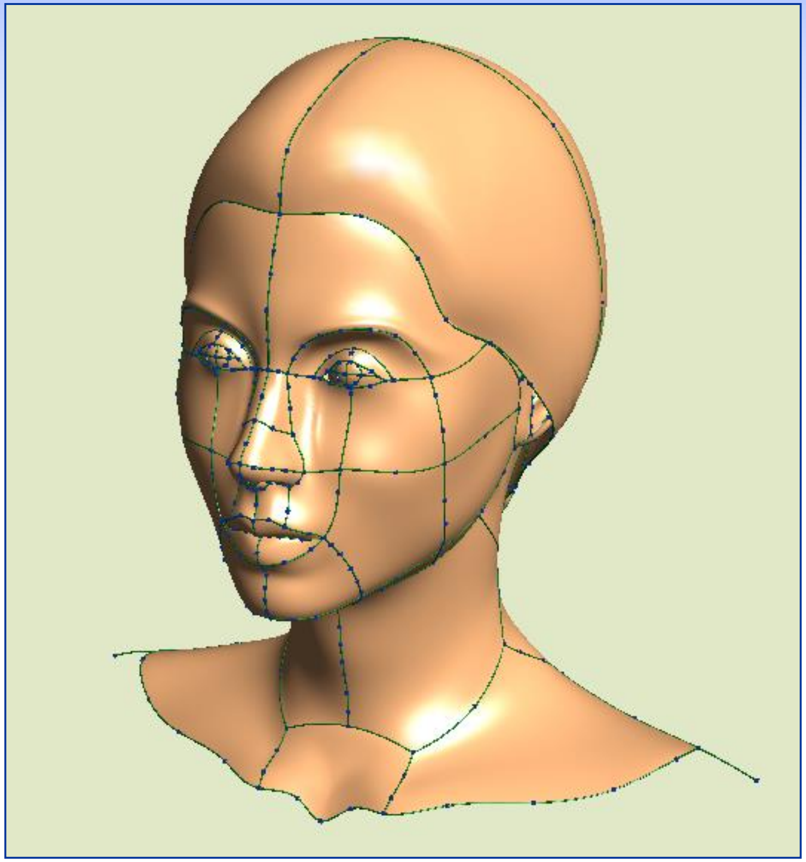




Curve-network based design$_3$

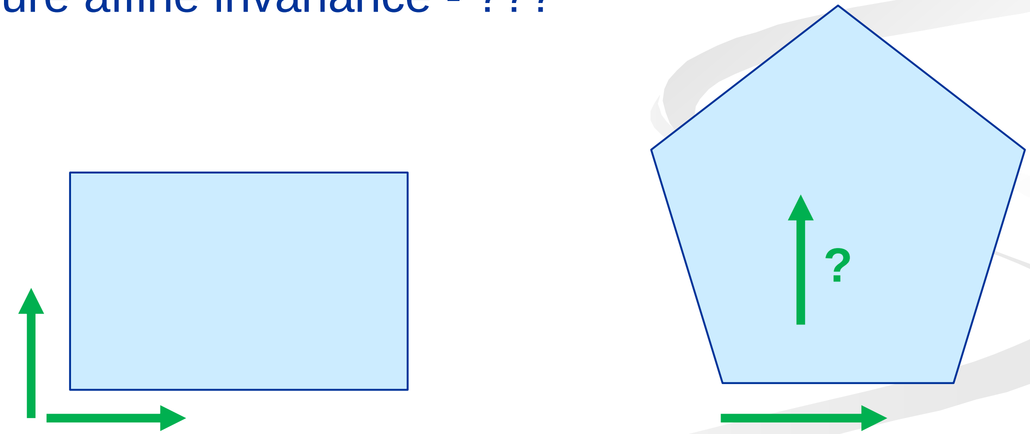


Curve-network based design$_4$



Quadrilateral → N-sided surfaces

- N-sided domain - ???
- spider-net control structure - ???
- new local parameterization - ???
- new basis functions - ???
- ensure interpolation - ???
- ensure affine invariance - ???

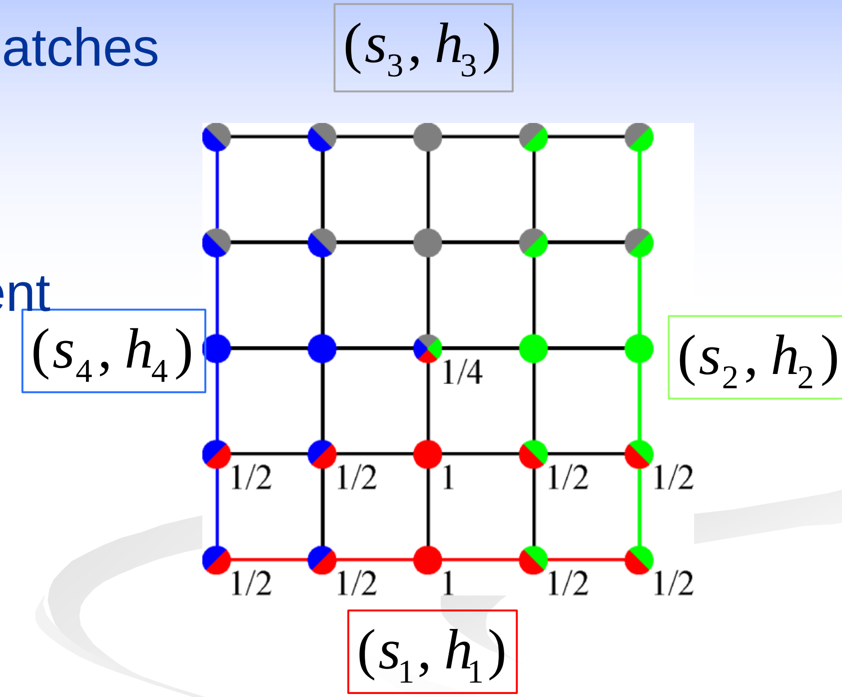


Basic idea

- re-interpret tensor-product Bézier patches

$$\mathbf{S}(u,v) = \sum_{j=0}^{d}\sum_{k=0}^{d} \mathbf{C}_{j,k}^{d} B_{j,k}^{d}(u,v)$$

- a rotationally symmetric arrangement of "half" Bézier ribbons



- introduce local side and distance parameters; circular indexing

$$s_1 = u,\, h_1 = v;\; s_2 = v,\, h_2 = 1-u;\; \ldots$$

- reformulated equation (4 ribbons)

$$\mathbf{S}(u,v) = \sum_{i=1}^{4}\sum_{j=0}^{d}\sum_{k=0}^{l-1} \mathbf{C}_{j,k}^{d,i} \mu_{j,k}^{i} B_{j,k}^{d}(s_i, h_i) + \mathbf{C}_{ll}^{d} \sum_{i=1}^{4} \mu_{l,l}^{i} B_{l,l}^{d}(s_i, h_i) \qquad C_{3,1}^{1} \equiv C_{1,1}^{2}\ !$$

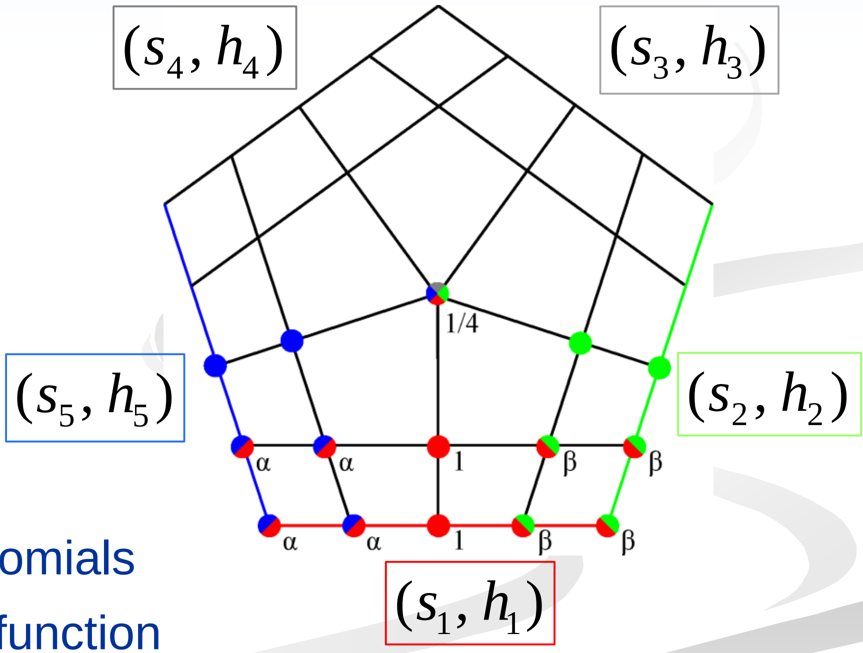
Generalized Bézier patch$_1$

i-th ribbon: $$\mathbf{Rib}(s_i,h_i)=\sum_{j=0}^{d}\sum_{k=0}^{l-1}\mathbf{C}_{j,k}^{d,i}\mu_{j,k}^{i}B_{j,k}^{d}(s_i,h_i)$$

Surface equation: $$\mathbf{S}(u,v)=\sum_{i=1}^{n}\mathbf{Rib}(s_i,h_i)+\mathbf{C}_0B_0(u,v)$$

- n - number of sides
- d - degree
- l - number of layers: $l=(d+1)\div 2;$

  $d=3,4 \rightarrow l=2;\ d=5,6 \rightarrow l=3;\ \ldots.$
- $\mathbf{C}^{d,i}$ - CP-s on the *i*-th side
- μ^i - scalar multipliers (α,β)
- B^d - bi-parametric Bernstein polynomials
- $\mathbf{C}_0$, $B_0(u,v)$ - central CP and blend function

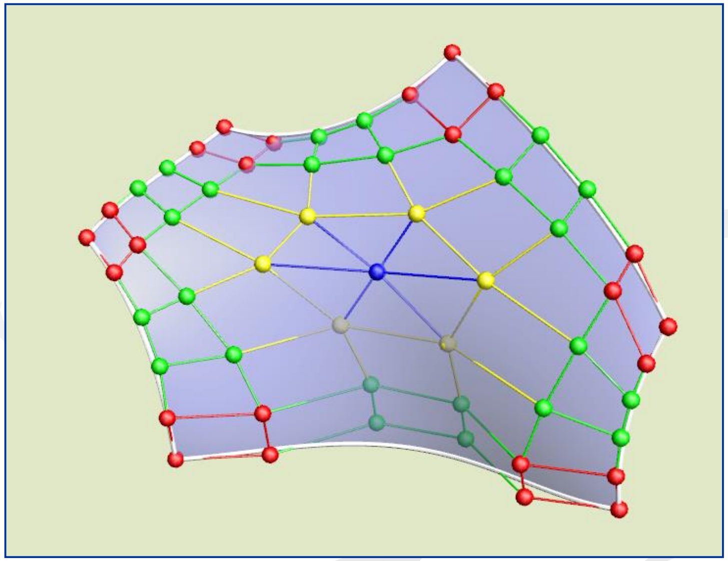


Generalized Bézier patch$_2$

G^1 patch: reproduces boundary positions and cross-derivatives determined by the first two rows of control points (ribbons)

Alternative coloring

- corner CP (red) - shared by sides i and $i+1$
- ribbon CP (green) - exclusive for side i
- interior CP (yellow) - placed by degree elevation
- center CP (blue) - tweaks the interior



(i) create a set of local side-distance parameter pairs, $(u,v,) \rightarrow \{s_i,h_i\}$

(ii) blend the adjacent parameterizations

Generalized barycentric coordinates

- n-sided polygon
- All points in the interior can be defined as a weighted combination of the vertices
- Continuous mapping

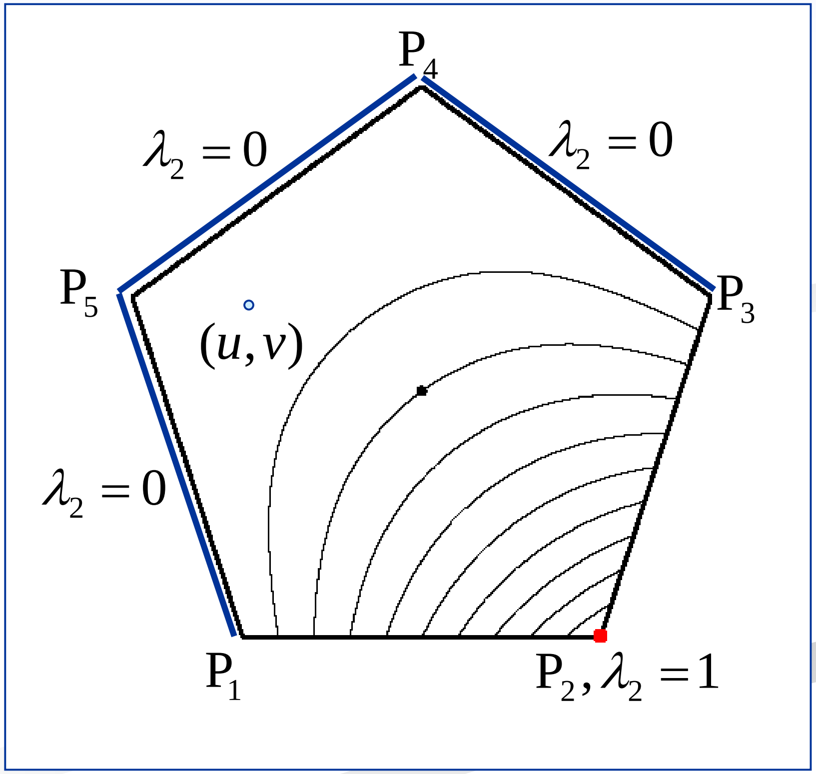
$$(u,v) \Leftrightarrow [\lambda_1,\dots,\lambda_n]$$

$$\lambda_i \geq 0 \quad [\,positivity\,]$$

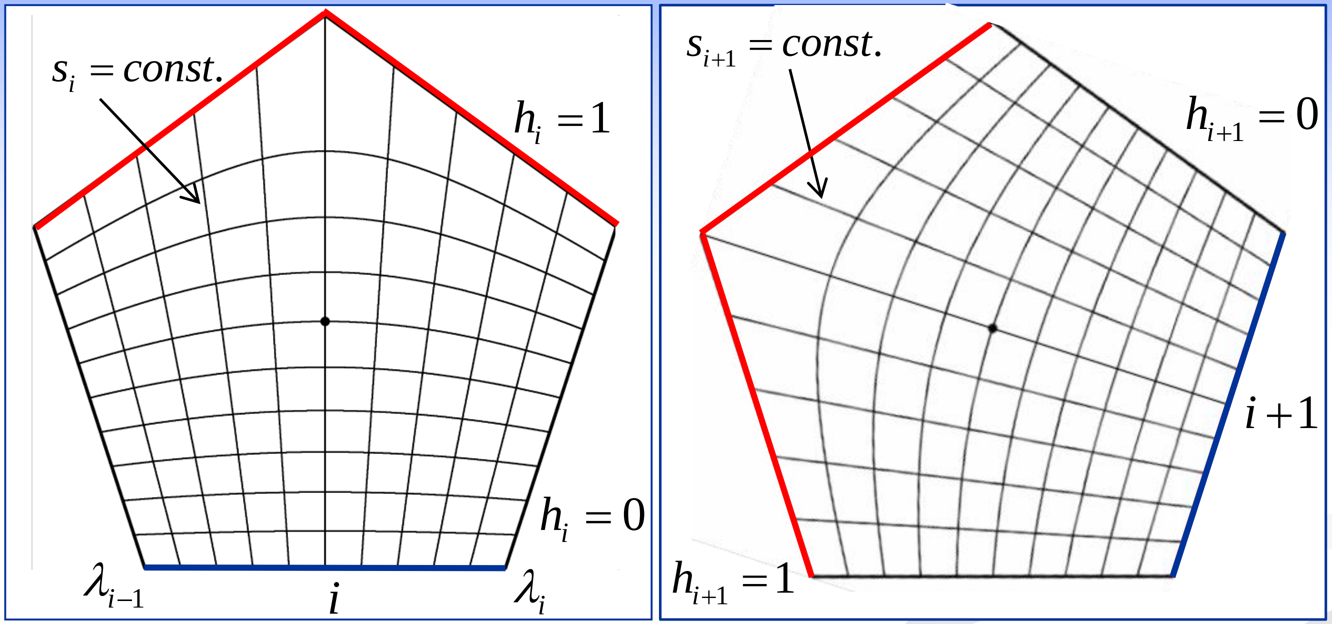
$$\sum_{i=0}^{n} \lambda_i = 1 \quad [\,partition\ \ of\ \ unity]$$

$$(u,v) = \sum_i \lambda_i(u,v)\mathrm{P}_i \quad [\,reproduction\,]$$

$$\lambda_i(\mathrm{P}_j) = \delta_{ij} \quad [\,Lagrange\ property\,]$$



Local side and distance parameters

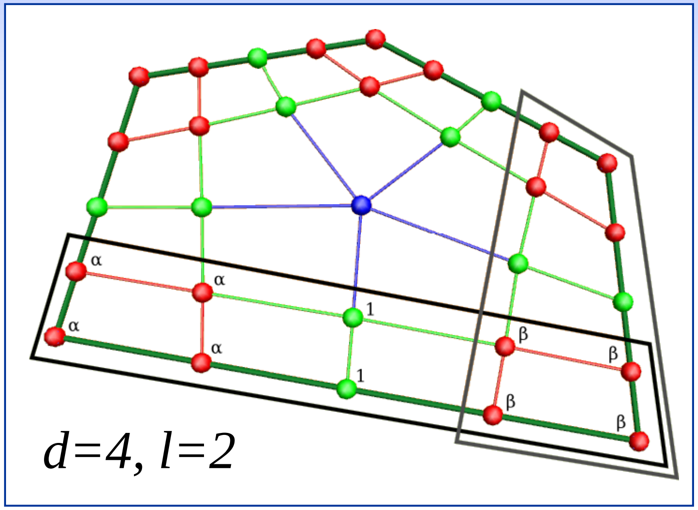


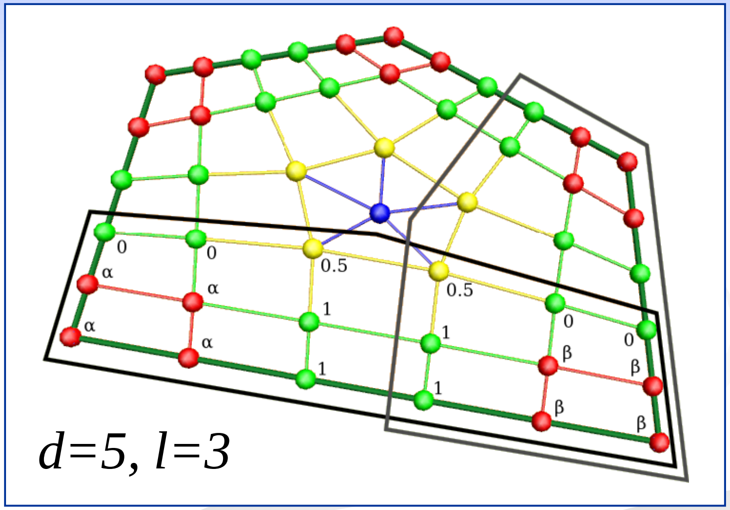
- For each side compute local parameters s_i and h_i from Wachspress coordinates

$$s_i(u,v) = \frac{\lambda_i}{\lambda_{i-1} + \lambda_i}; \quad h_i(u,v) = 1 - \lambda_{i-1} - \lambda_i$$

- s_i is linear on side i, $s_i = const.$ are straight lines
- h_i is zero on side i, $h_i=1$ on sides $j \neq i\text{-}1,i,i+1$

Blend functions by side



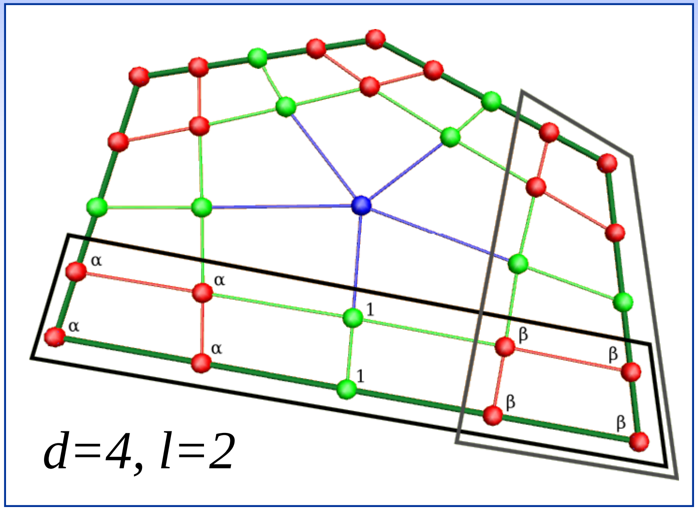


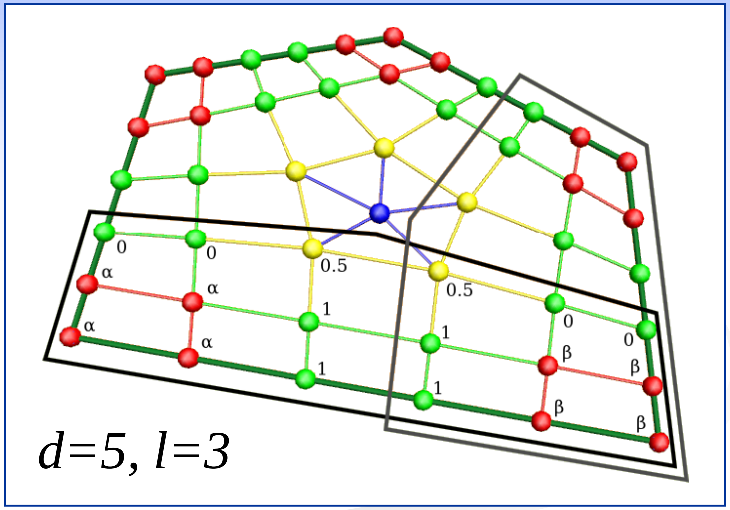
$$\mu^{i}_{j,k}:\{0, 0.5, 1, \alpha_i, \beta_i\};\ \alpha_i = \frac{h_{i-1}}{h_{i-1}+h_i};\ \beta_i = \frac{h_{i+1}}{h_{i+1}+h_i}$$

- green & middle interior CPs multiplied by a constant and a *single* biparametric Bernstein blend function:

$$B^d_{j,k}(s_i, h_i) = B^d_j(s_i)B^d_k(h_i);\quad B^d_j(s) = \binom{d}{j}(1-s)^{d-j}s^j$$

Blend functions at the corner





$$\mu^i_{j,k} : \{0, 0.5, 1, \alpha_i, \beta_i\};\ \alpha_i = \frac{h_{i-1}}{h_{i-1}+h_i};\ \beta_i = \frac{h_{i+1}}{h_{i+1}+h_i}$$

- red corner CPs multiplied by *two* blending functions

$$\beta_i B^d_{j,k}(s_i, h_i) + \alpha_{i+1} B^d_{k,d-j}(s_{i+1}, h_{i+1})$$

- on the i-th boundary $\quad h_i = 0 \rightarrow \beta_i = 1;\ \alpha_{i+1} = \frac{h_i}{h_i + h_{i+1}} = 0$

and the formula degenerates to $B^d_{j,k}(s_i, 0)$

G^1 continuity

- Thus on the *i*-th side

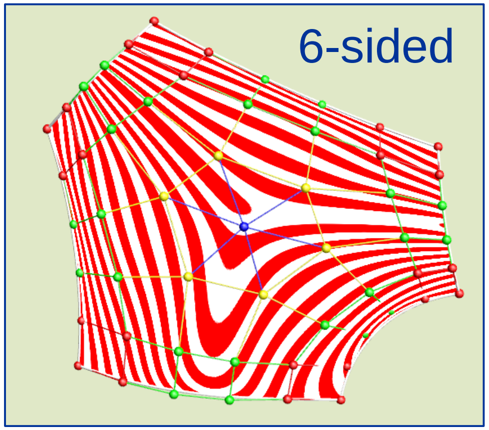
$$\mathbf{S}(u,v) = \sum_{i=1}^{n} \sum_{j=0}^{d} \sum_{k=0}^{l-1} \mathbf{C}_{j,k}^{d,i} \mu_{j,k}^{i} B_{j,k}^{d}(s_i, h_i) + \mathbf{C}_0 B_0(u,v)$$

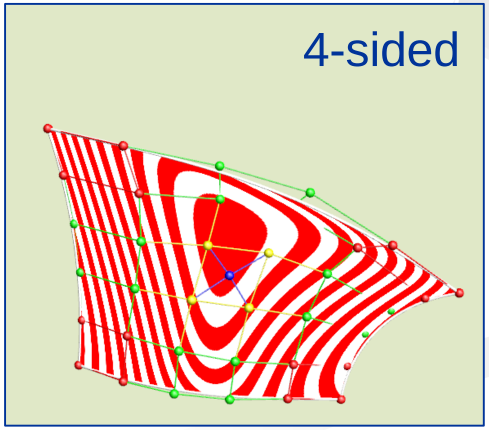
- Bézier boundary reproduced

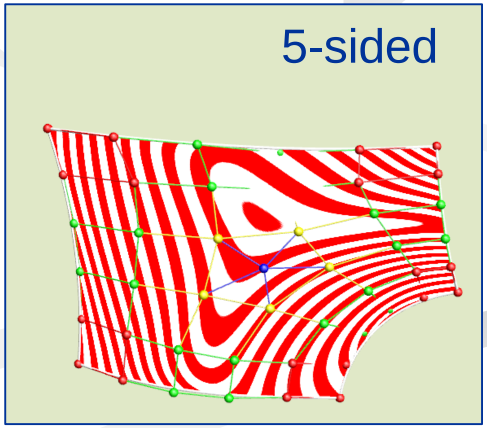
$$\mathbf{S}(u,v)_{|h_i=0} = \sum_{j=0}^{d} \mathbf{C}_{j,0}^{d,i} B_{j,0}^{d}(s_i, 0)$$

- Bézier cross-derivative reproduced

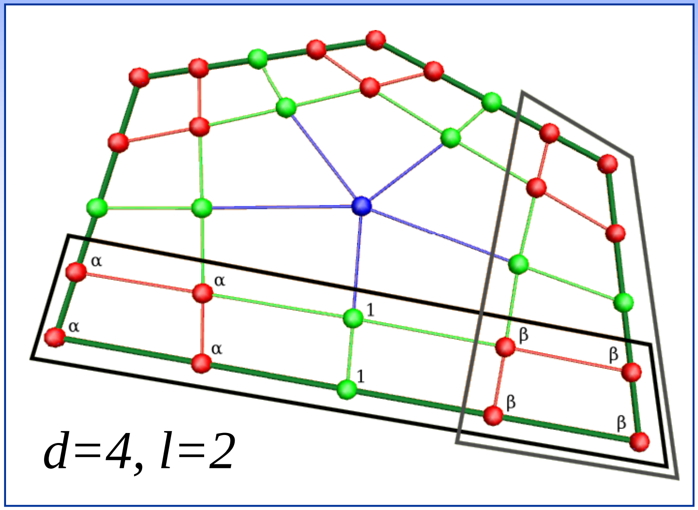
$$\frac{\partial}{\partial h_i} \mathbf{S}(u,v)_{|h_i=0} = d \sum_{j=0}^{d} (\mathbf{C}_{j,1}^{d,i} - \mathbf{C}_{j,0}^{d,i}) B_{j,0}^{d}(s_i, 0)$$

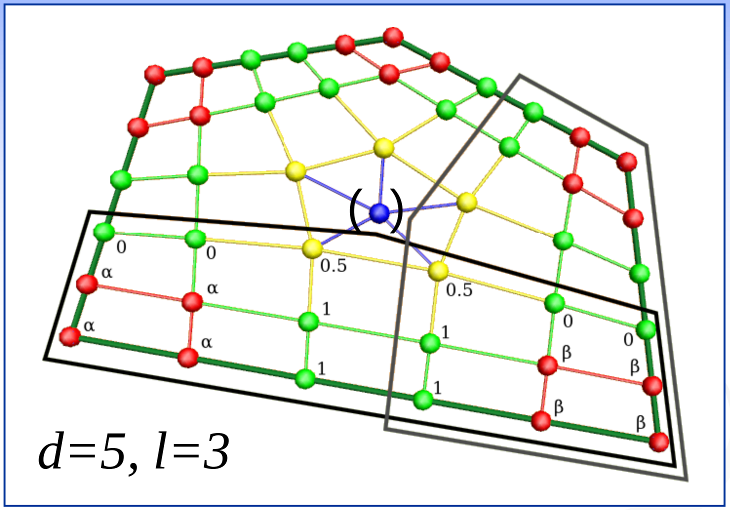






Weight deficiency



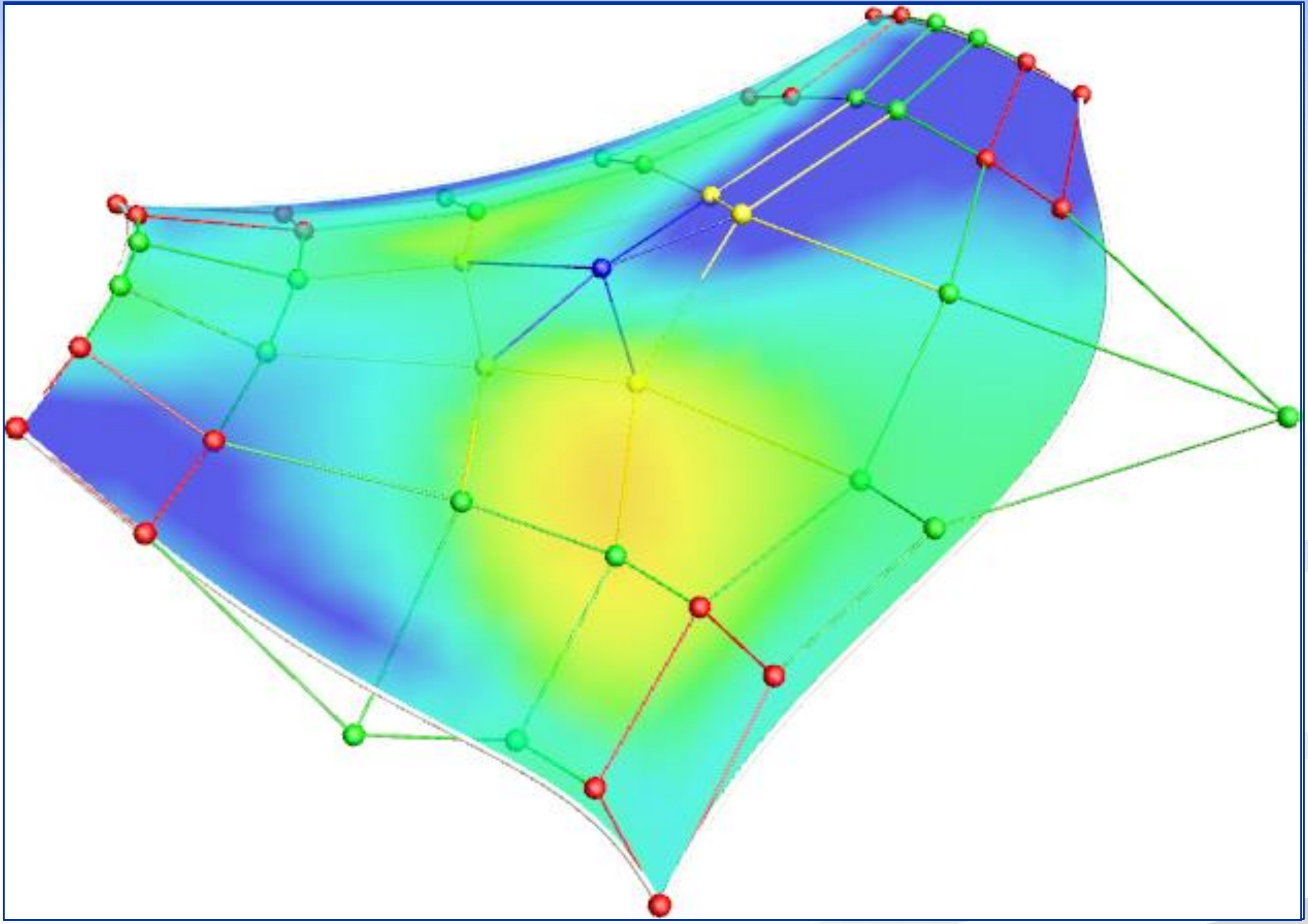


- sum of the blends functions $\sum_{i=1}^{n}\sum_{j=0}^{d}\sum_{k=0}^{l-1} \mu_{j,k}^{i} B_{j,k}^{d}(s_i, h_i) \leq 1$
- for convex combination weight deficiency must be compensated

$$B_0(u,v) = 1 - \sum_{i=1}^{n}\sum_{j=0}^{d}\sum_{k=0}^{l-1} \mu_{j,k}^{i} B_{j,k}^{d}(s_i, h_i)$$

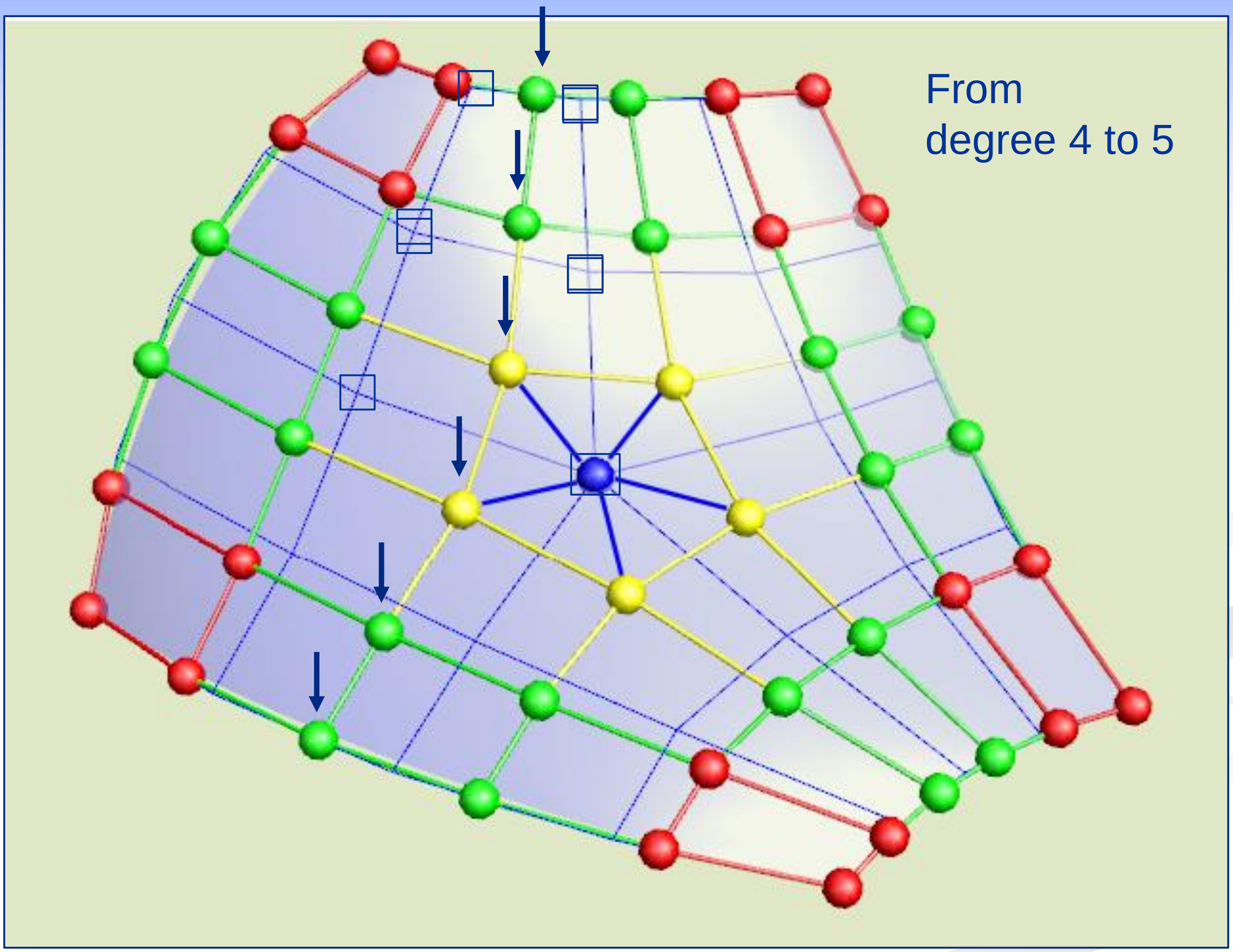
- a central CP $\mathbf{C_0}$ is assigned to the central blend function

Editing GB patches



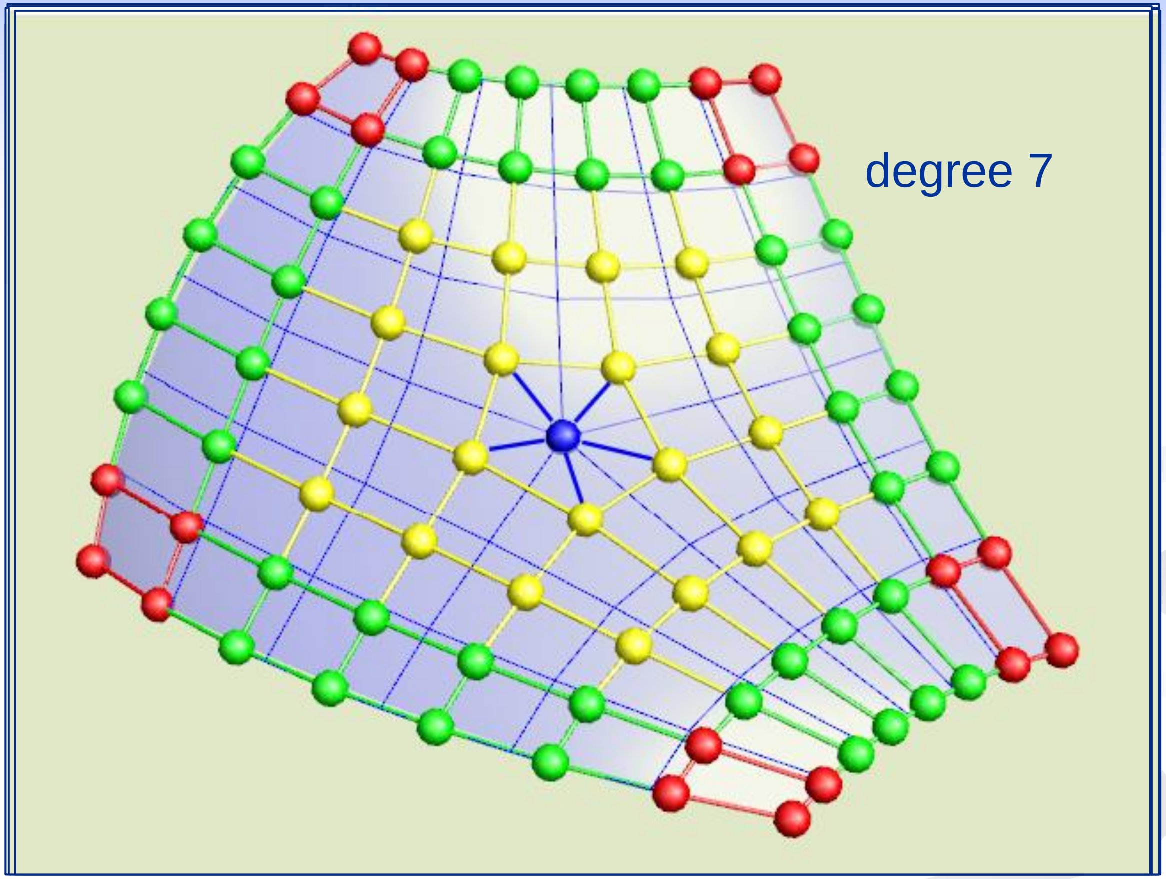
Editing proceeds in the same way as for quadrilateral patches

Degree Elevation for GB patches



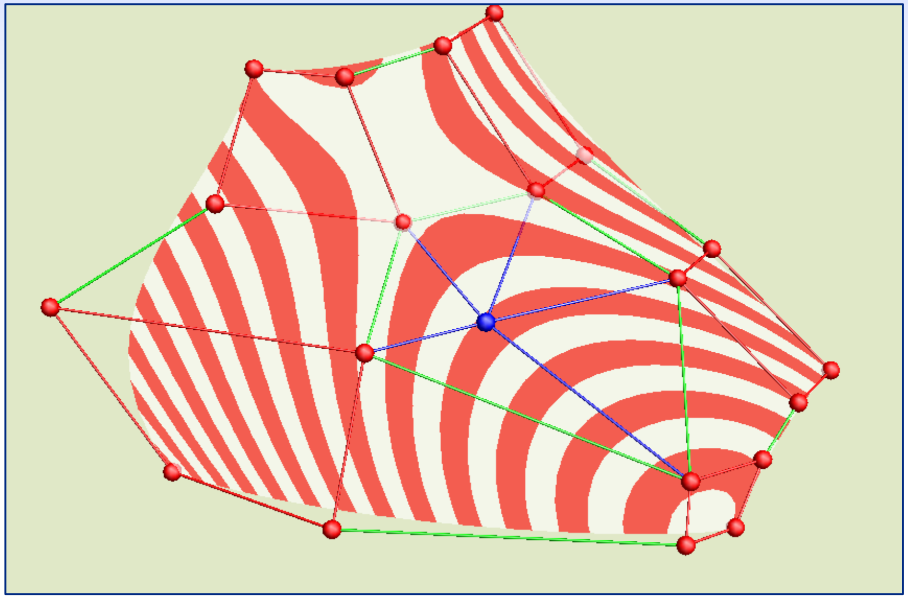
Insert new CPs on the chords of the boundary polygons, and in the quadrilaterals of the control structure

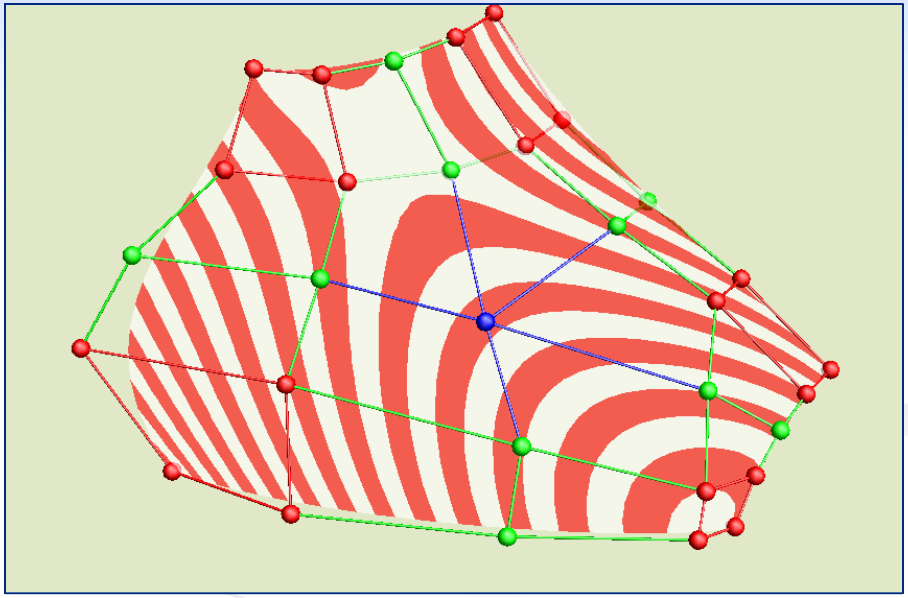
Degree Elevation for GB patches



From degree 4 to degree 10

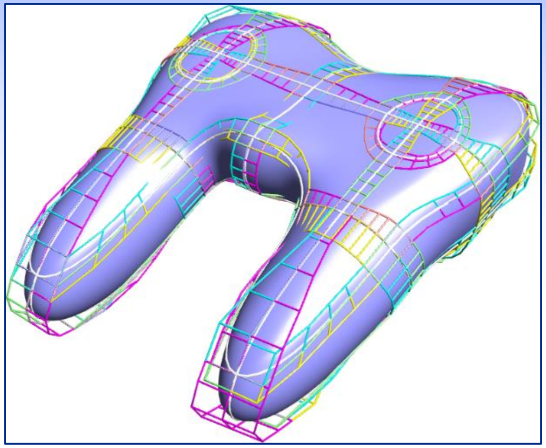
Degree Elevation for GB patches



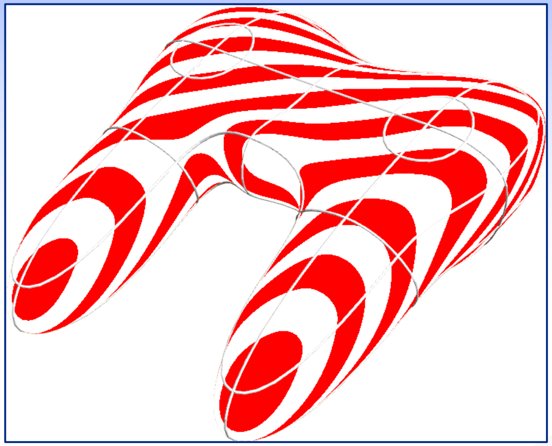


Degree elevation is *accurate* along the boundaries,
but only a *close approximation* in the interior

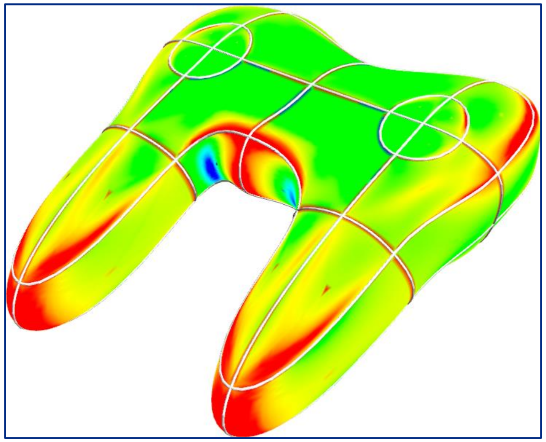
Case Study$_1$ – game pad



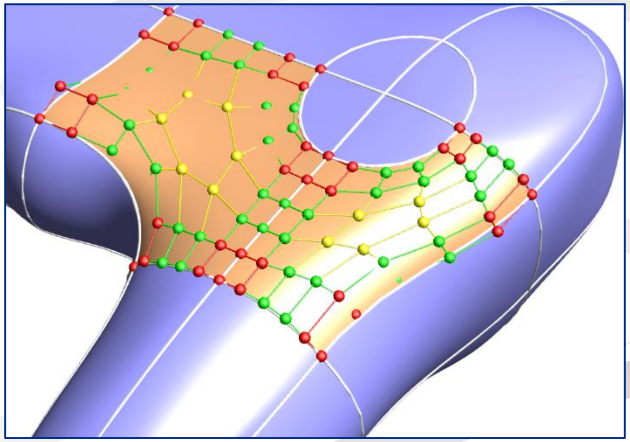
Ribbons



Contouring

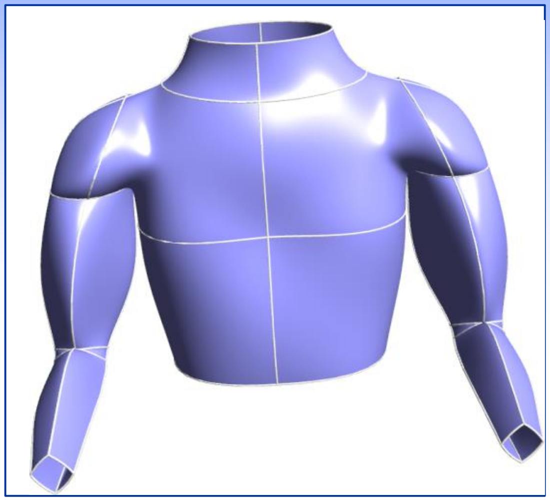


Curvature map

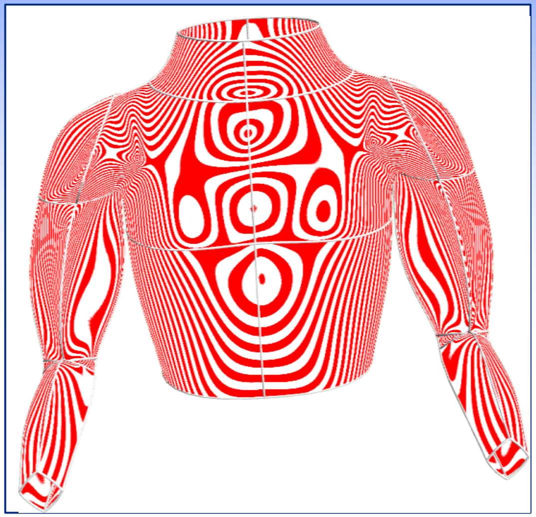


5- and 6-sided GB patches

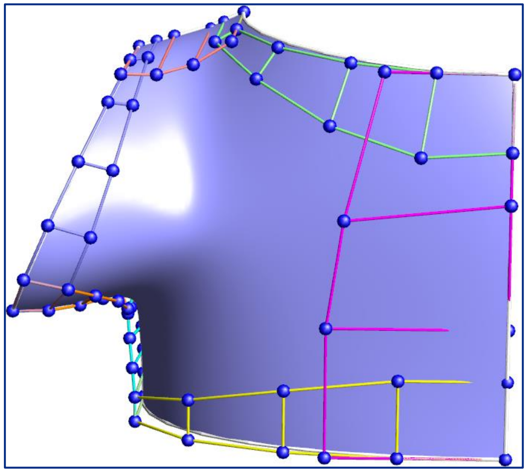
Case Study$_2$ – torso



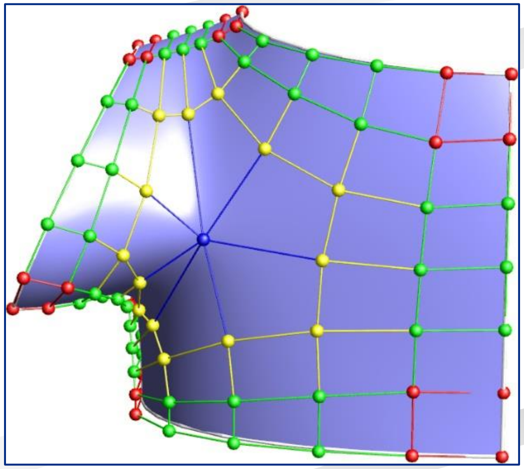
Ribbons



Isophotes

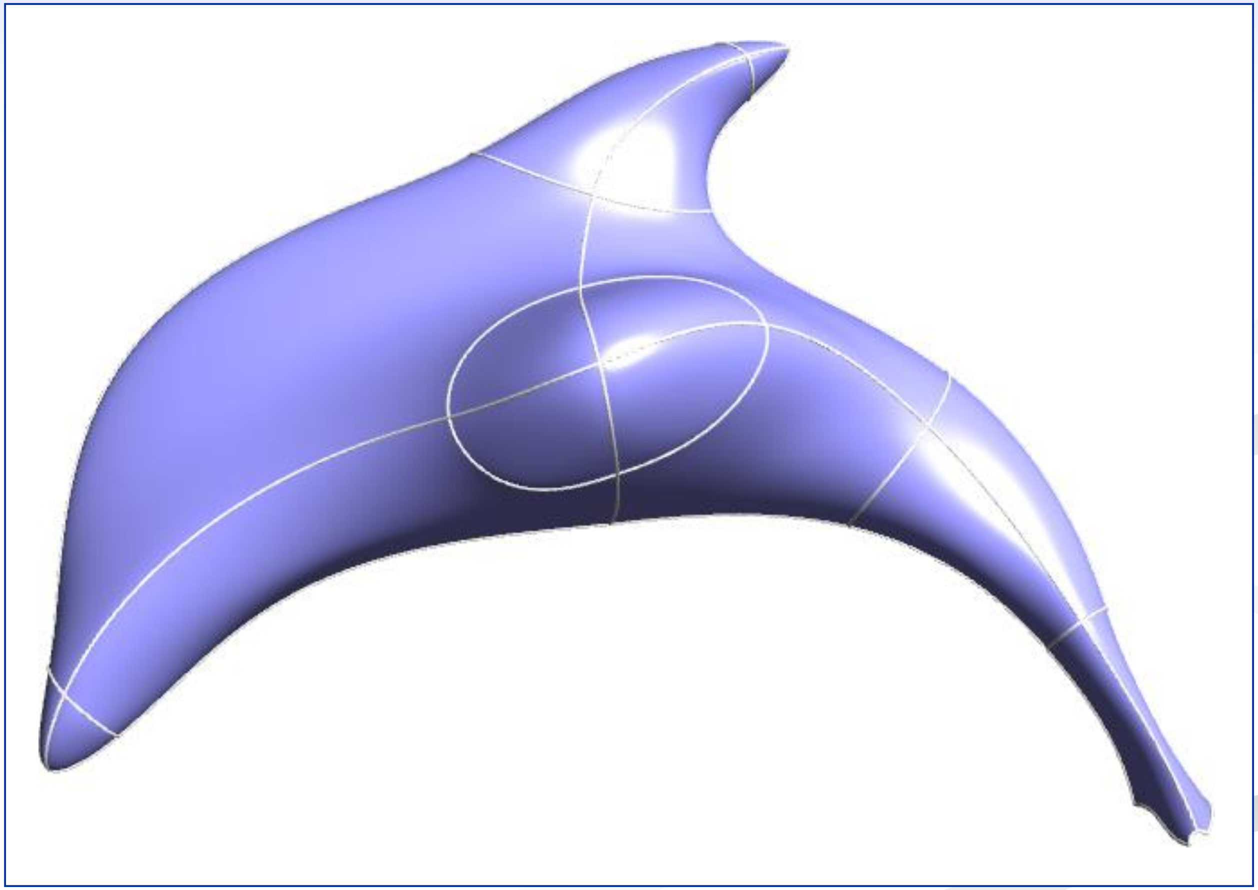


Ribbons with different degrees

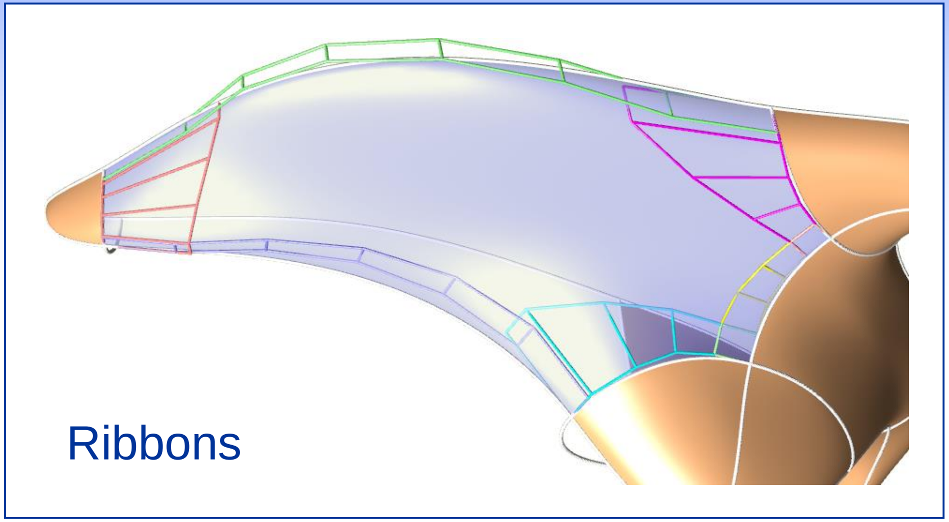


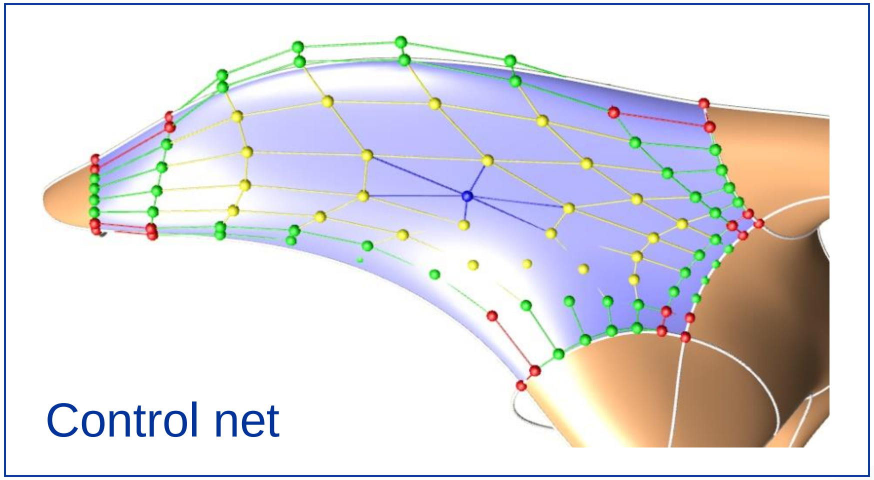
Unified 7-sided GB patch

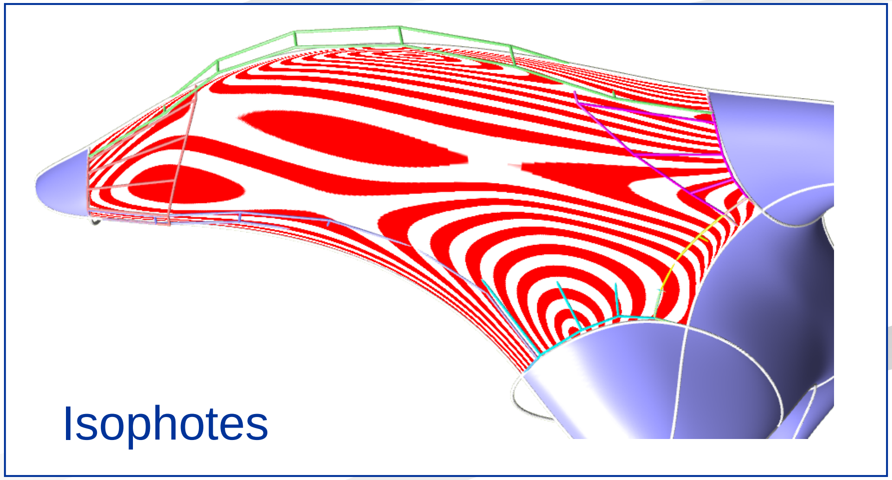
Case Study$_3$ - modeling a dolphin by GB patches



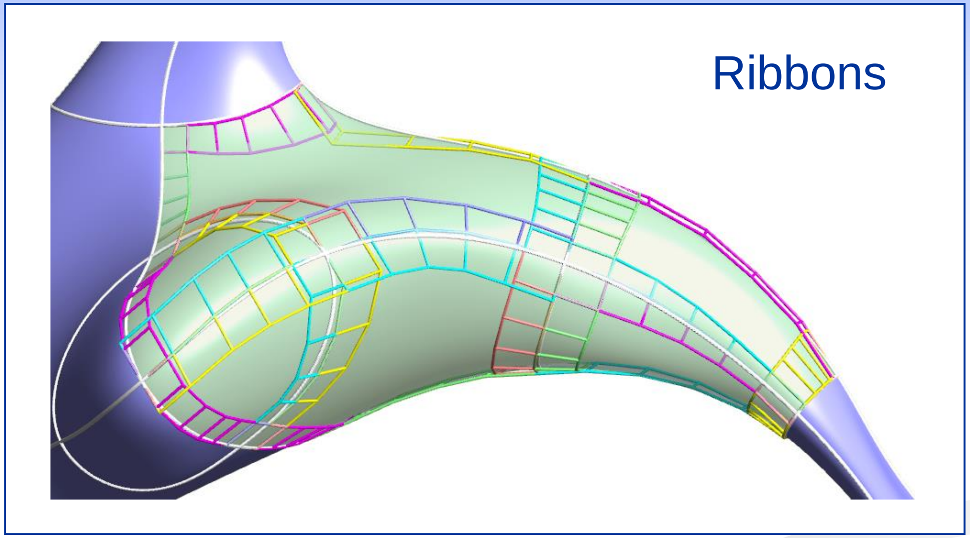
A six-sided patch

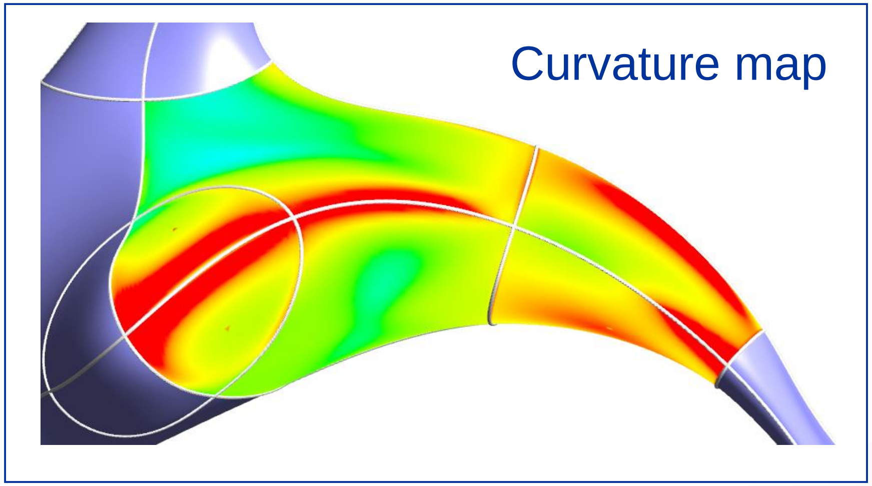


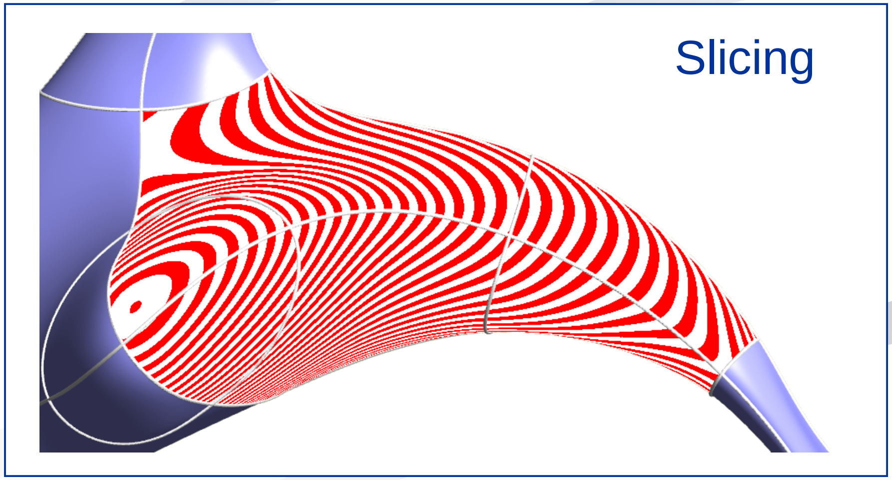




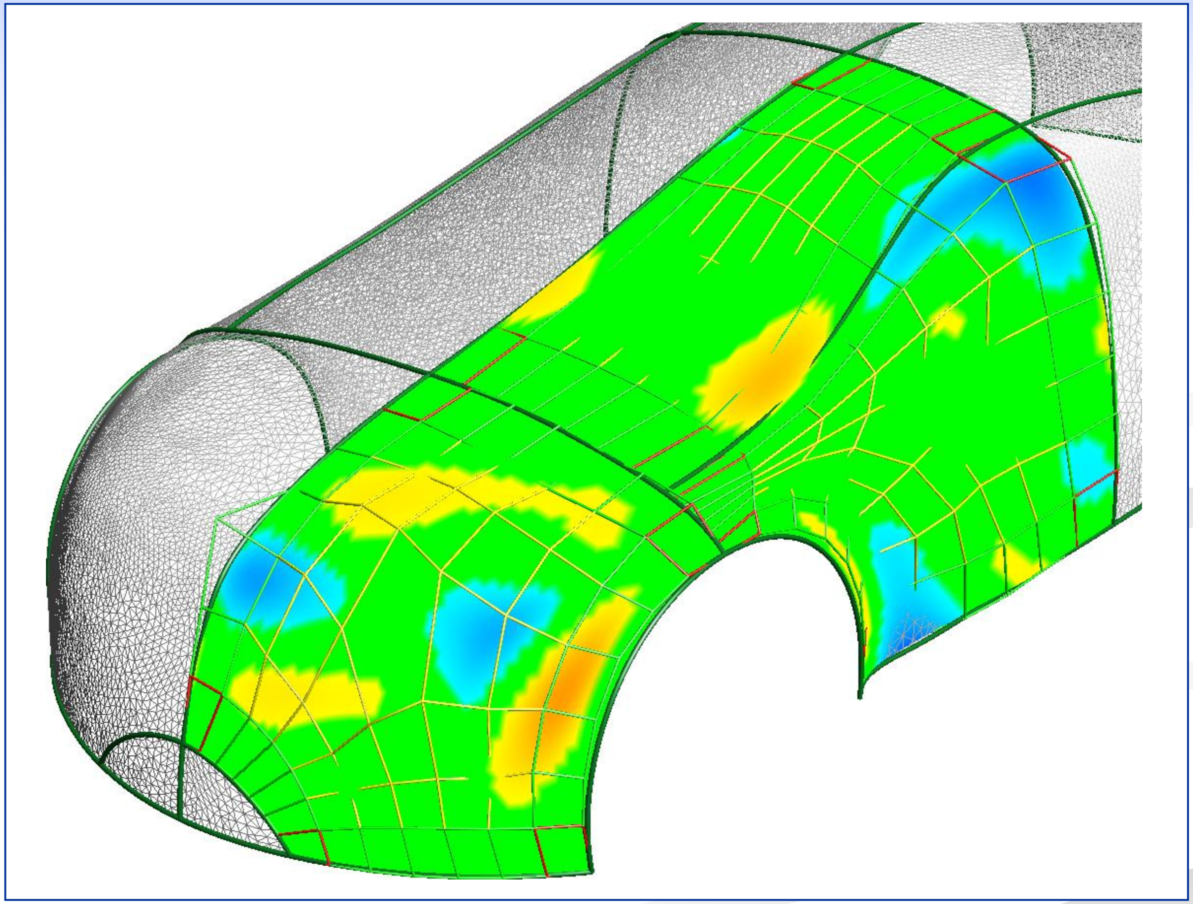
Collection of patches







Case Study$_4$ – GB patches for optimization and approximation



Lsq-fit for the internal control points with some smoothing

Conclusion

Generalized Bézier patches

- natural control structure
- compatible with quadrilateral Bézier patches (G^1, G^2)
- rationally weighted Bernstein functions
- interior control points - automatically generated for further editing and optimization
- merging Bézier ribbons with different degrees