Ribbon-based Transfinite Surfaces

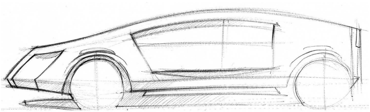
Péter Salvi[†], Tamás Várady[†], Alyn Rockwood[‡]

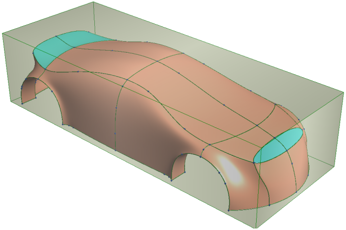
[†]Budapest University of Technology and Economics
[‡]King Abdullah University of Science and Technology

CAGD 31(9), pp. 613–630, 2014.

GMP 2015

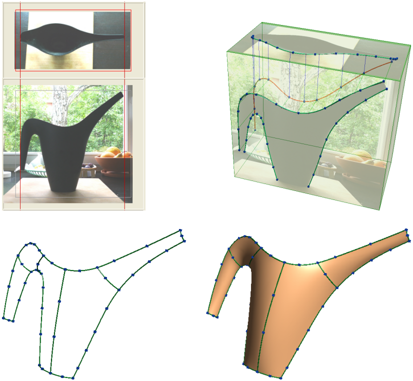
Outline





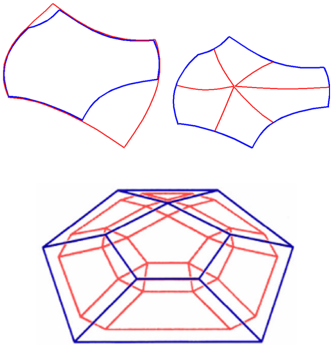
Motivation

- Free-form surface design based on feature curves
- Hand-drawn sketches or images as input
- Tools for 3D curve / cross-derivative generation
- Semi-automatically generated surfaces
- Key issue: *n*-sided surface representation



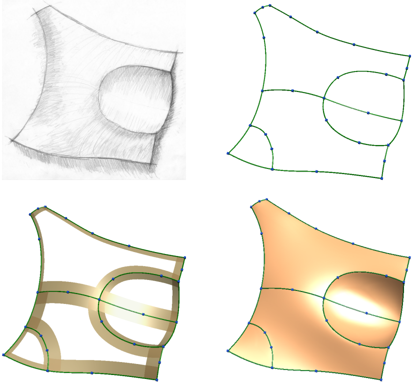
Conventional Surfacing Methods

- Trimming
  - Defining the quadrilateral?
  - Boundary modification?
  - Stitching?
- Quadrilaterals
  - Creating smooth divisions?
  - Modification – effect on the dividing curves?
- Recursive subdivision
  - Initial polyhedra?
  - Cross-derivative constraints?



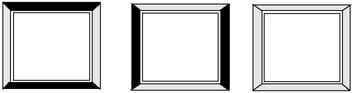
Transfinite Surface Interpolation

- Avoid dealing with control points or polyhedra
- No need for interior data
- Exact boundary interpolation
- Real-time editing of complex free-form models
- Smooth connections
- Previous work:
  - Coons '67
  - Charrot–Gregory '84
  - Kato '91
  - Sabin '96 etc.



C^1 Coons Patch

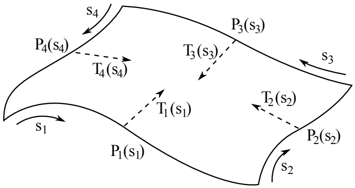
- Boundary curves:
  $S(u,0),\ S(u,1),\ S(0,v),\ S(1,v)$
- Cross-derivatives:
  $S_v(u,0),\ S_v(u,1),\ S_u(0,v),\ S_u(1,v)$
- Hermite blends: $\alpha_0,\ \alpha_1,\ \beta_0,\ \beta_1$



$$
\begin{aligned}
U &= \begin{bmatrix} \alpha_0(u) & \beta_0(u) & \alpha_1(u) & \beta_1(u) \end{bmatrix} \\
V &= \begin{bmatrix} \alpha_0(v) & \beta_0(v) & \alpha_1(v) & \beta_1(v) \end{bmatrix} \\
S^u &= \begin{bmatrix} S(u,0) & S_v(u,0) & S(u,1) & S_v(u,1) \end{bmatrix} \\
S^v &= \begin{bmatrix} S(0,v) & S_u(0,v) & S(1,v) & S_u(1,v) \end{bmatrix} \\
S^{uv} &= \begin{bmatrix} S(0,0) & S_u(0,0) & S(1,0) & S_u(1,0) \\ S_v(0,0) & S_{uv}(0,0) & S_v(1,0) & S_{uv}(1,0) \\ S(0,1) & S_u(0,1) & S(1,1) & S_u(1,1) \\ S_v(0,1) & S_{uv}(0,1) & S_v(1,1) & S_{uv}(1,1) \end{bmatrix} \\
S(u,v) &= V(S^u)^T + S^v U^T - V S^{uv} U^T
\end{aligned}
$$

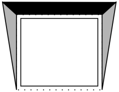
Reformulation

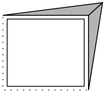
- Positional and tangential constraints: $P_i(s_i)$ and $T_i(s_i)$
- Assume compatible twists: $W_{i,i-1} = T_i'(0) = -T_{i-1}'(1)$



$$S(u,v) = \sum_{i=1}^{4} \begin{bmatrix} \alpha_0(s_{i+1}) \\ \beta_0(s_{i+1}) \end{bmatrix}^T \begin{bmatrix} P_i(s_i) \\ T_i(s_i) \end{bmatrix} -$$

$$\sum_{i=1}^{4} \begin{bmatrix} \alpha_0(s_{i+1}) \\ \beta_0(s_{i+1}) \end{bmatrix}^T \begin{bmatrix} P_i(0) & P_i'(0) \\ T_i(0) & T_i'(0) \end{bmatrix} \begin{bmatrix} \alpha_0(s_i) \\ \beta_0(s_i) \end{bmatrix}$$

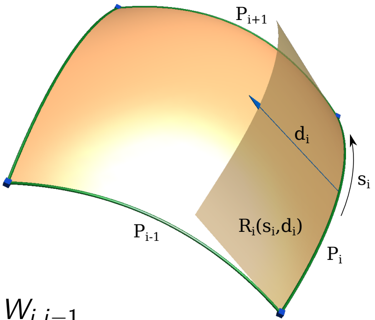




Ribbon-based Coons Patch

- Linear interpolants (ribbons):
  $R_i(s_i, d_i) = P_i(s_i) + \gamma(d_i)T_i(s_i)$
- $\gamma(d_i) = \beta_0(d_i)/\alpha_0(d_i) = \frac{d_i}{2d_i+1}$
- Distance parameter $d_i = s_{i+1}$
- Corner correction patch
  $Q_{i,i-1}(s_i, s_{i-1}) =$

$$P_i(0) + \gamma(1 - s_{i-1})T_i(0) + \gamma(s_i)T_{i-1}(1) + \gamma(s_i)\gamma(1 - s_{i-1})W_{i,i-1}$$

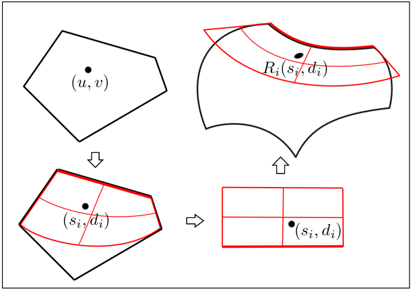


$$S(u, v) = \sum_{i=1}^{4} R_i(s_i, d_i)\alpha_0(d_i) - \sum_{i=1}^{4} Q_{i,i-1}(s_i, s_{i-1})\alpha_0(s_i)\alpha_0(s_{i-1})$$

Transfinite Surface Interpolation

- Input: Hermite data $(P_i,\ T_i)$
- Surface $S(u,v) =$

$$\sum_{i=1}^{n} \mathrm{Interpolant}_i(s_i, d_i) \cdot \mathrm{Blend}_i(d_1, \ldots, d_n)$$

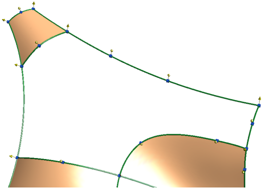


Constituents

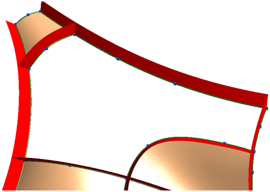
- Ribbons
- Domain polygon
- Parameterization functions
- Blending functions

Ribbon Construction

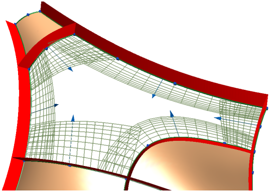
- Given: boundary curves $P_i(s_i)$ and normal vectors at some points



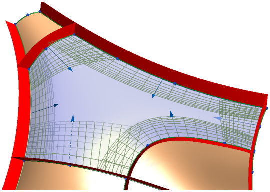
- Continuous normal vector function $N_i(s_i)$ by RMF



- $T_i(s_i) \perp N_i(s_i)$

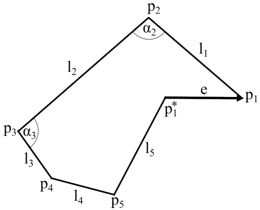


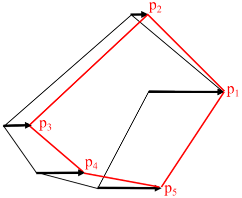
- Resulting surface



Domain Construction

- Regular *n*-sided polygon (good most of the time)
- Domain "similar" to the boundary curves
- Similarity of...
  - Arc lengths
  - Angles
- Measure of similarity:
  - Deviation of arc length / angle ratios
- Use heuristics if measure > threshold





(see Várady et al. '11)

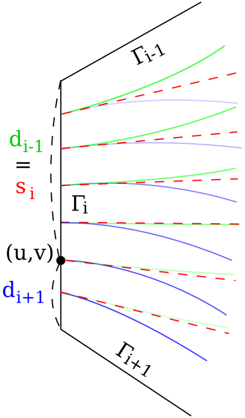
Ribbon Mapping – Parameterization Constraints

- $s_i \in [0, 1]$
- For a point on side i...
  - Simple parameterization:

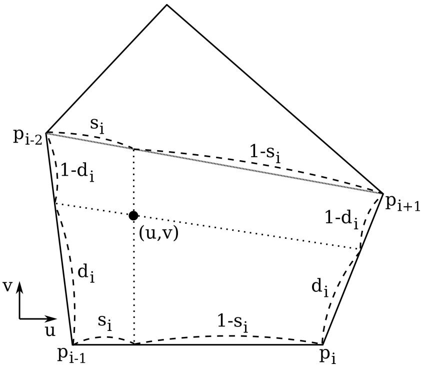
$$d_i = 0$$
$$s_{i-1} = 1 \qquad s_{i+1} = 0$$
$$d_{i-1} = s_i \qquad d_{i+1} = 1 - s_i$$

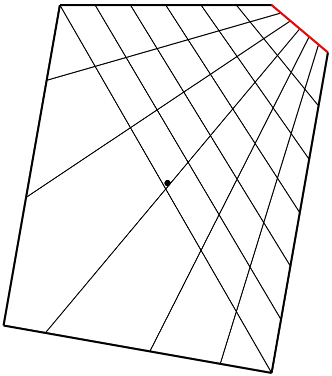
  - Constrained parameterization:

$$\frac{\partial d_{i-1}}{\partial u} = \frac{\partial s_i}{\partial u} \qquad \frac{\partial d_{i-1}}{\partial v} = \frac{\partial s_i}{\partial v}$$
$$\frac{\partial d_{i+1}}{\partial u} = -\frac{\partial s_i}{\partial u} \qquad \frac{\partial d_{i+1}}{\partial v} = -\frac{\partial s_i}{\partial v}$$



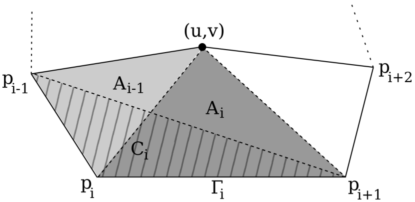
Bilinear Line Sweep (simple)

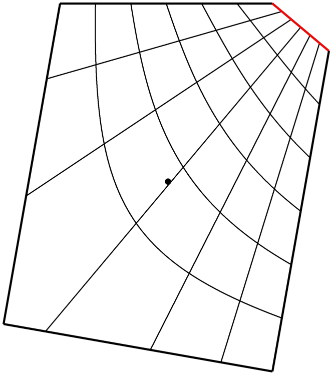




Wachspress Distance Parameters (simple)

- For convex polygons
- $\lambda_i(u,v) = w_i(u,v)/\sum_k w_k(u,v)$
- $w_i(u,v) = C_i/(A_{i-1}(u,v)\cdot A_i(u,v))$
- $\Rightarrow\ d_i(u,v) = 1-\lambda_{i-1}(u,v)-\lambda_i(u,v)$

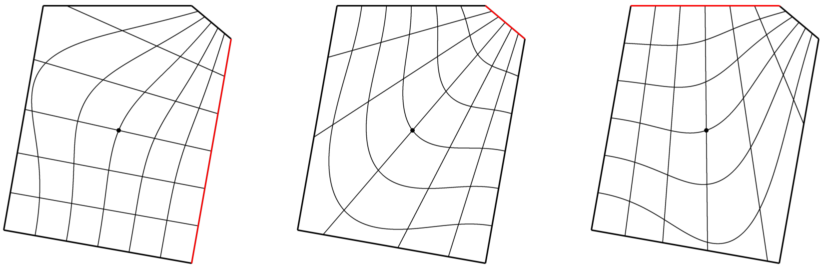




Interconnected Parameterization (constrained)

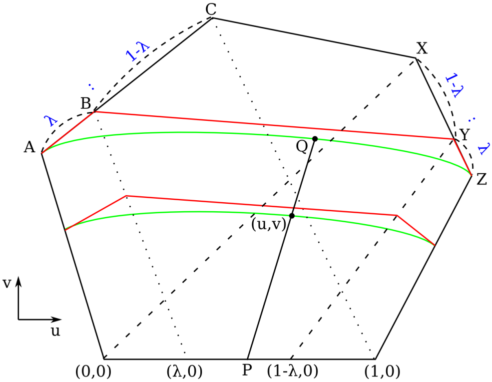
- Let $s_i(u, v)$ be a line sweep (e.g. bilinear)

$$d_i(u, v) = (1 - s_{i-1}(u, v)) \cdot \alpha_0(s_i) + s_{i+1}(u, v) \cdot \alpha_1(s_i)$$



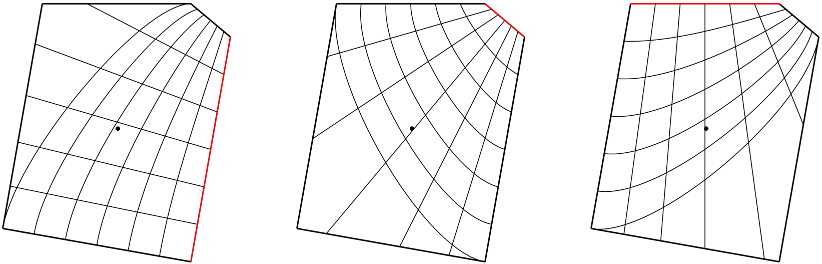
Cubic Parameterization (constrained)

- Based on bilinear
- Constant parameter lines defined by cubic Bézier curves
- λ: fullness parameter
- Leads to a sixth-degree equation
  - Only fourth-degree when $\lambda = \frac{1}{3}$
  - Precomputable



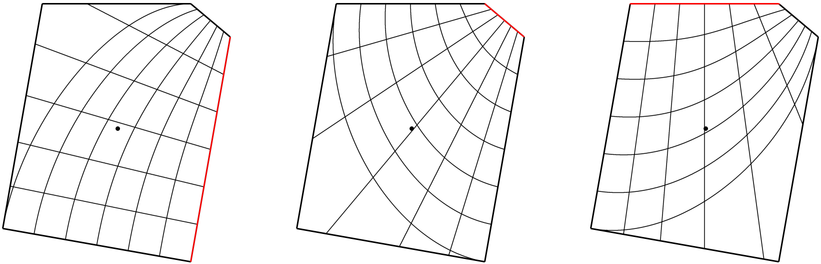
Example

- Using $\lambda = \frac{1}{3}$:



Example

- Using $\lambda = \frac{1}{2}$:



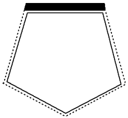
Blending Side Interpolants ⇒ SB Patch

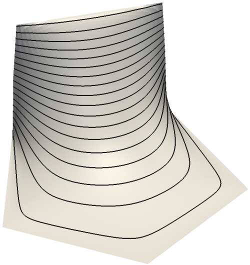
"Side-based" (SB) patch [Kato '91]

$$S^{SB}(u,v) = \sum_{i=1}^{n} R_i(s_i, d_i) \cdot B_i^*(d_1, \dots, d_n)$$

$$B_i^*(d_1, \dots, d_n) = \frac{1/d_i^2}{\sum_j 1/d_j^2} = \frac{\prod_{k \neq i} d_k^2}{\sum_j \prod_{k \neq j} d_k^2}$$

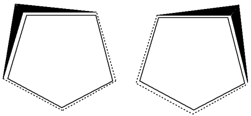
- Blend function singular in the corners





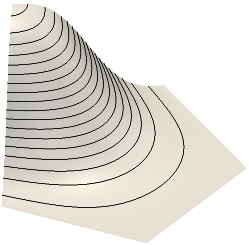
Blending Corner Interpolants ⇒ CB Patch

“Corner-based” (CB) patch
[Charrot–Gregory ’84]



$$S^{CB}(u,v) = \sum_{i=1}^{n} R_{i,i-1}(s_i, s_{i-1}) \cdot B_{i,i-1}(d_1, \dots, d_n)$$

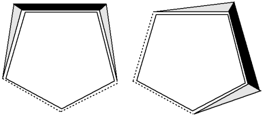
$$B_{i,i-1}(d_1, \dots, d_n) = \frac{\prod_{k \notin \{i,i-1\}} d_k^2}{\sum_j \prod_{k \notin \{j,j-1\}} d_k^2}$$

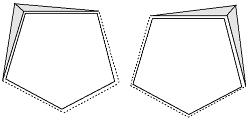


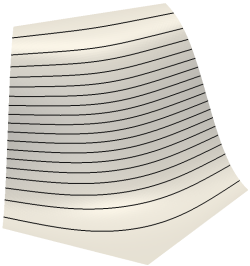
- Corner interpolants:

$$R_{i,i-1}(s_i, s_{i-1}) = R_i(s_i, 1 - s_{i-1}) + R_{i-1}(s_{i-1}, s_i) - Q_{i,i-1}(s_i, s_{i-1})$$

GC Patch – Boolean Sum Construction







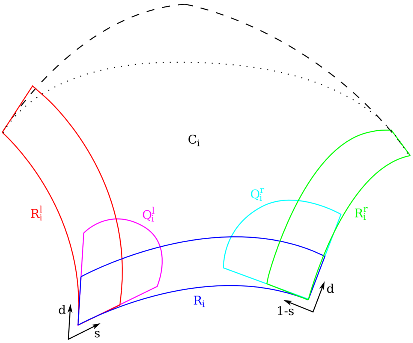
- Same logic as in the reformulated Coons patch
- Side blend: $B_i = B_{i,i-1} + B_{i+1,i}$
- Needs constrained parameterization

$$S(u,v) = \sum_{i=1}^{n} R_i(s_i, d_i) \cdot B_i(d_1, \dots, d_n) - \sum_{i=1}^{n} Q_{i,i-1}(s_i, s_{i-1}) \cdot B_{i,i-1}(d_1, \dots, d_n)$$

CR Patch – Ribbons Interpolating 3 Sides

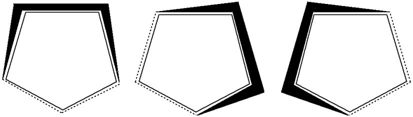
- Curved side interpolants

$$
\begin{aligned}
C_i(s,d) \quad = \quad & \\
& R_i(s,d)\alpha_0(d) + \\
& R_i^l(s,d)\alpha_0(s) + R_i^r(s,d)\alpha_1(s) - \\
& Q_i^l(s,d)\alpha_0(s)\alpha_0(d) - \\
& Q_i^r(s,d)\alpha_1(s)\alpha_0(d) -
\end{aligned}
$$



$$
\begin{aligned}
R_i^l(s,d) &= R_{i-1}(1-d,s) \\
R_i^r(s,d) &= R_{i+1}(d,1-s) \\
Q_i^l(s,d) &= Q_{i,i-1}(s,1-d) \\
Q_i^r(s,d) &= Q_{i+1,i}(d,s)
\end{aligned}
$$

CR Patch – Simpler Equation



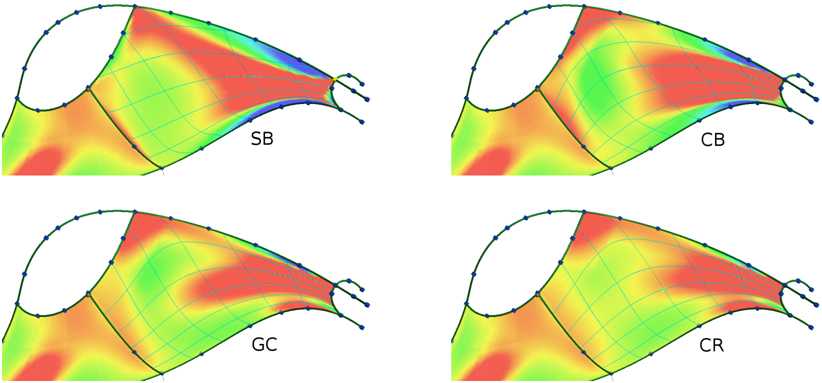
- No need for correction patches:

$$S(u,v) = \frac{1}{2}\sum_{i=1}^{n} C_i(s_i,d_i)B_i(d_1,\dots,d_n)$$

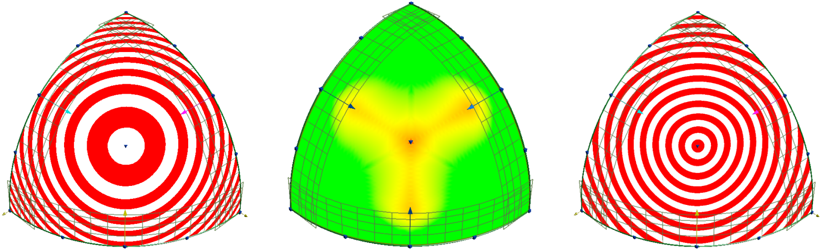
(Correction patches are *inside* curved ribbons)

- Simple parameterization ⇒ reproduces tangent planes
- Constrained parameterization ⇒ reproduces first derivatives

Mean Map Comparison



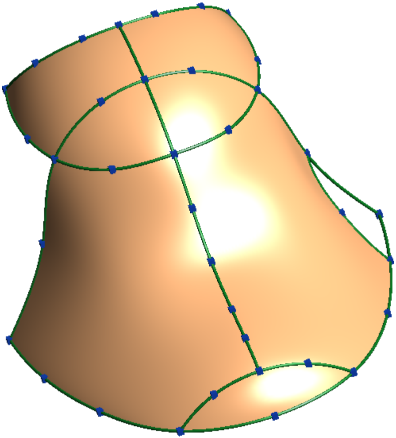
Approximating a Sphere (CR)

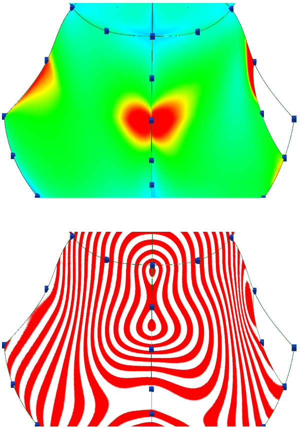


	Min	Max	Average	Std. Deviation
SB	0.9963	1.0098	1.0040	3.41e-3
CB	0.9942	1.0082	0.9990	3.13e-3
GC	0.9960	1.0082	1.0014	3.02e-3
CR	0.9960	1.0057	1.0007	2.77e-3

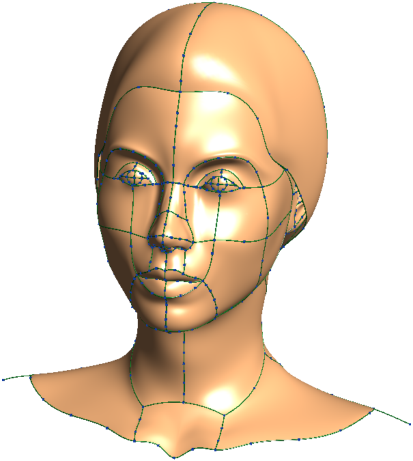
Radii

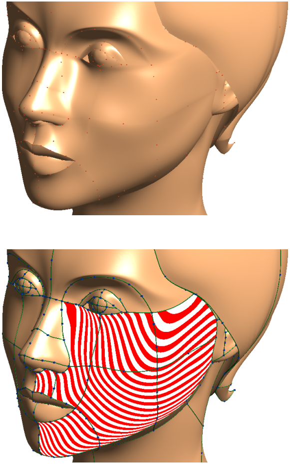
Stability (CR)





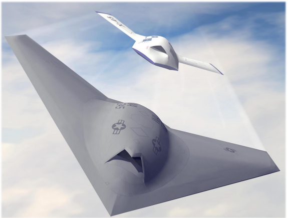
A Complex Model

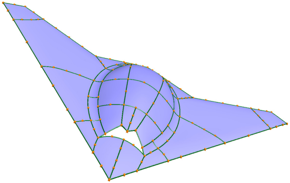




Summary

- Constrained parameterizations
  - Interconnected
  - Cubic
- Coons patch generalization
- Composite ribbon patch
  - Curved ribbons
- Future work
  - G^2 patches (Salvi et al. PG'14)
  - Concave domains
  - Fairing algorithms





Any Questions?

Thank you for your attention.