CAD Surfaces & solid modeling

CAD Surfaces & solid modeling

- Parametric surfaces
  - (Coons patches) – “like” splines for curves
  - Tensor product surfaces
- Procedural (non parametric) surfaces
  - Subdivision surfaces
- Solid modeling

“Tensor product” surfaces

Tensor product surfaces

- Parametric surfaces as a polar form

$$S(u,v)=\sum_k N_k(u,v)\,P_k$$

  - One shape function per control point
- If N_k is separable : « Tensor product » surface
  - Combination of elementary curves/shape functions independently defined on u and v.

$$S(u,v)=\sum_i \sum_j G_i(u)\,H_j(v)\,P_{ij}$$

    Usually built upon Bézier and B-Splines curves/SFs
  - The "unique" shape function is $N_k(u,v)=G_i(u)\,H_j(v)$

B-Spline surfaces

- B-Splines surfaces uses 1D B-Spline shape fns.
  - Definition as tensor product :

$$S(u,v)=\sum_{i=0}^{n}\sum_{j=0}^{m} N_i^p(u)\,N_j^q(v)\,P_{ij}$$

  - Every variable u and v has a degree (p and q) and a nodal sequence U and V :

$$U=\{\underbrace{0,\cdots,0}_{p+1},u_{p+1},\cdots,u_{r-p-1},\underbrace{1,\cdots,1}_{p+1}\}\qquad (r+1\ \text{nodes})$$

$$V=\{\underbrace{0,\cdots,0}_{q+1},v_{q+1},\cdots,v_{s-q-1},\underbrace{1,\cdots,1}_{q+1}\}\qquad (s+1\ \text{nodes})$$

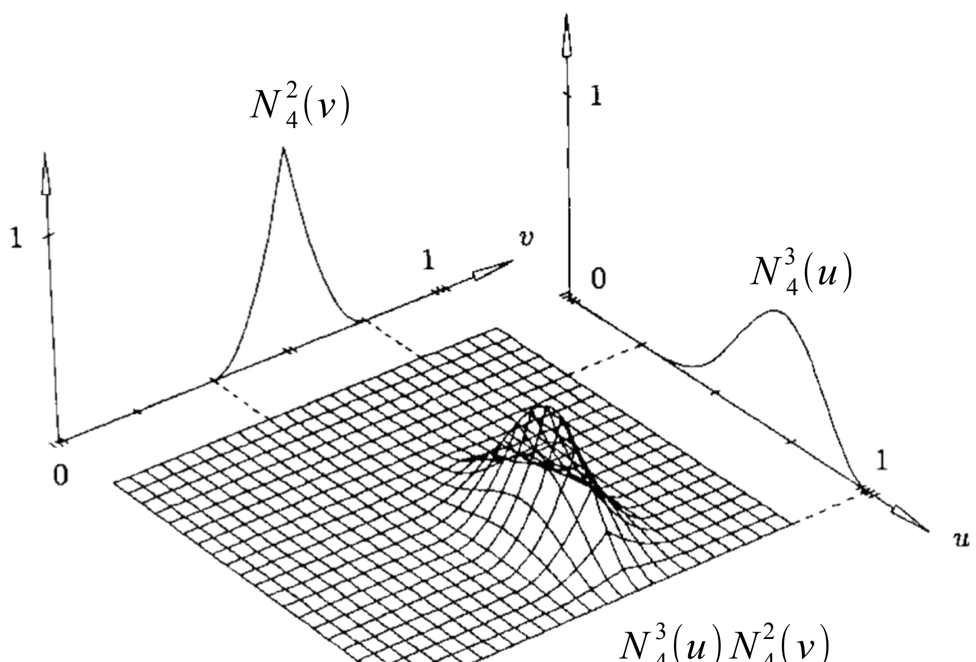
  - The control points forms a regular net P_{ij} ($n+1$ times $m+1$) values.
  - We have the following relations : $r=n+p+1 \quad s=m+q+1$

B-Spline surfaces

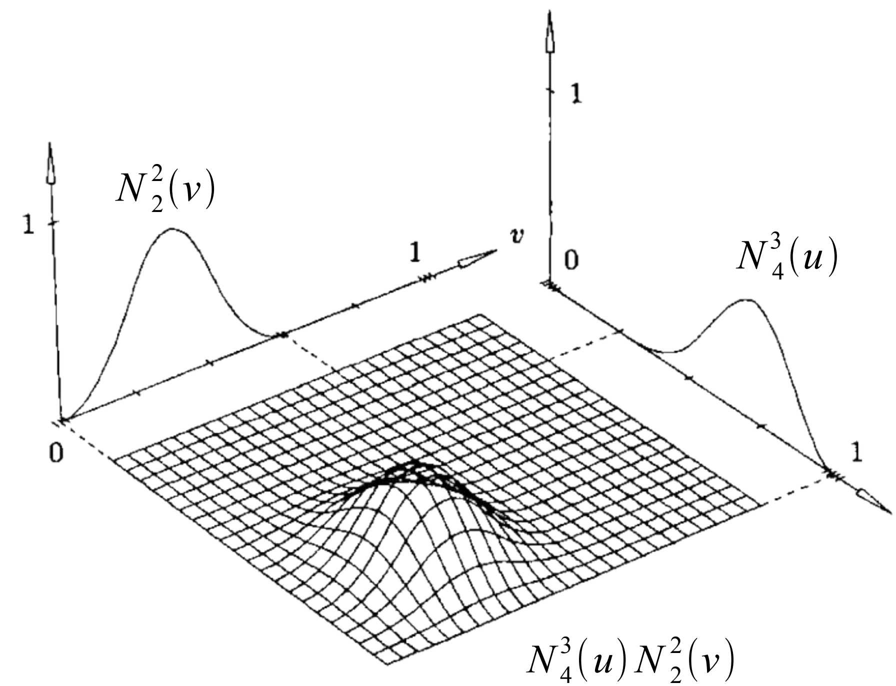
- Example of basis functions

$$U = \{0, 0, 0, 0, 1/4, 1/2, 3/4, 1, 1, 1, 1\} \qquad p = 3$$

$$V = \{0, 0, 0, 1/5, 2/5, 3/5, 3/5, 4/5, 1, 1, 1\} \qquad q = 2$$



Basis function associated to P_{44}



Basis function associated to P_{42}

L. Piegl « The NURBS Book »

- Properties of surface basis functions
  - Extrema
    If $p>0$ and $q>0$, $N_i^p(u)\,N_j^q(v)$ has a unique maximum.
  - Continuity
    Inside rectangles formed by the nodes u_i and v_j, the SF are infinitely differentiable.
    At a node u_i (resp. v_j),$N_i^p(u)\,N_j^q(v)$is (p-k) (resp. (q-k)) times differentiable, k being the node multiplicity u_i (resp. v_j)
    The continuity with respect to u (resp. v) depends solely on the nodal sequence U (resp. V).

B-Spline surfaces

- Properties of surface basis functions
  - A consequence of properties of the 1D shape functions
  - Non-negativity

$$N_i^p(u)\,N_j^q(v)\geq 0\;\forall\, i,j,p,q,u,v$$

  - Partition of unity

$$\sum_{i=0}^{n}\sum_{j=0}^{m} N_i^p(u)\,N_j^q(v)=1\;\forall\,(u,v)\in[u_{min},u_{max}]\times[v_{min},v_{max}]$$

  - Compact support

$$N_i^p(u)\,N_j^q(v)=0 \text{ outside } (u,v)\in[u_i,u_{i+p+1}[\times[v_j,v_{j+q+1}[$$

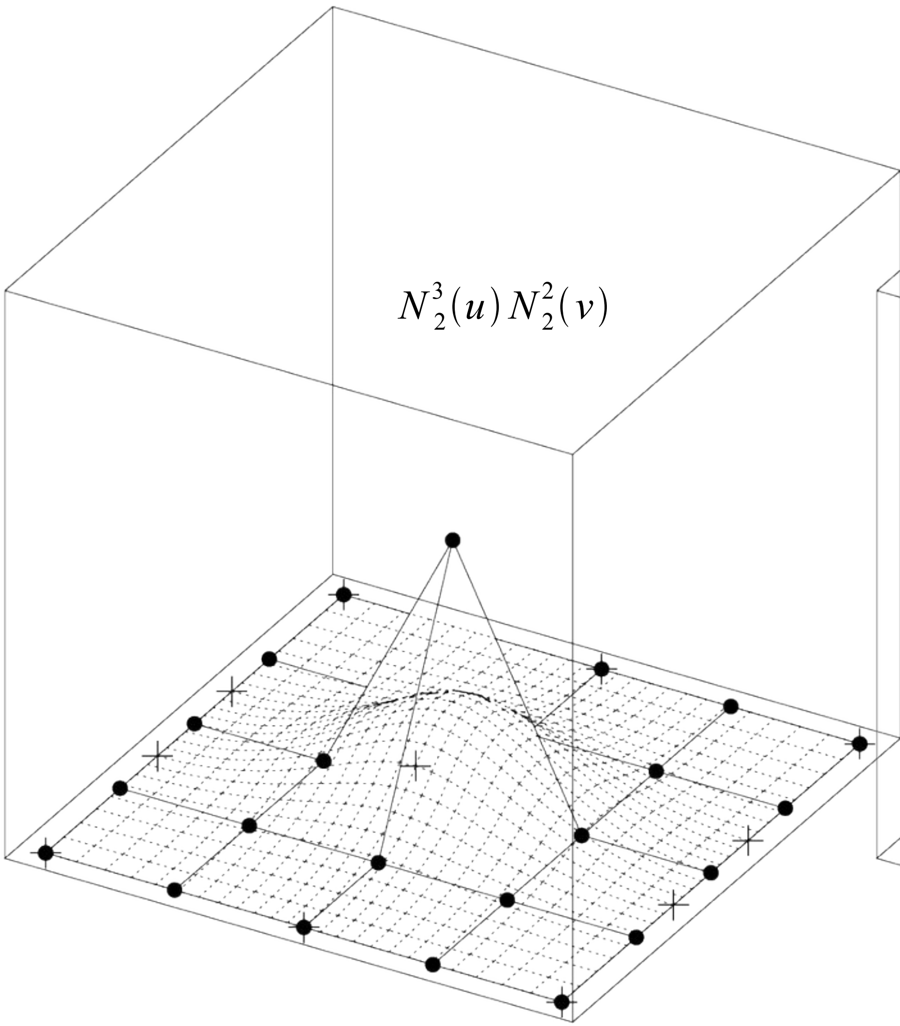
There are at most $(p+1)(q+1)$ non zero SF in a given interval $[u_{i_0},u_{i_0+1}[\times[v_{j_0},v_{j_0+1}[$.

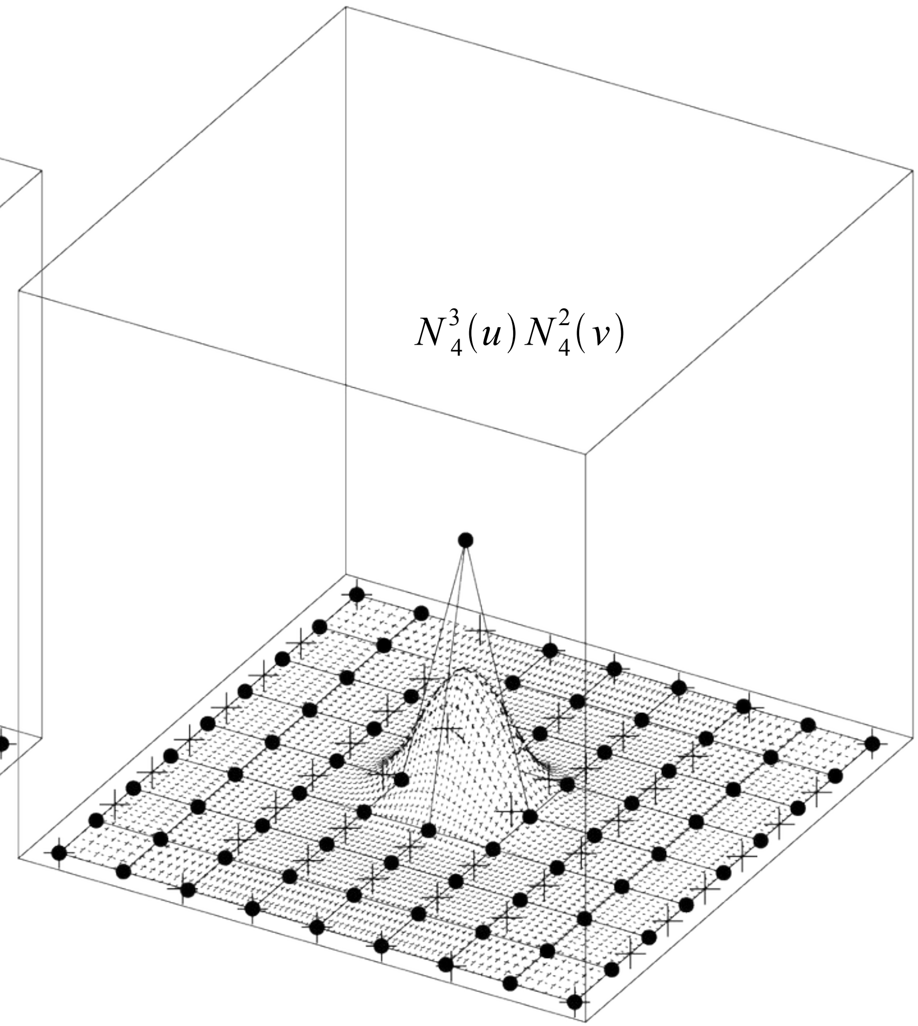
In particular $N_i^p(u)\,N_j^q(v)\neq 0 \quad i_0-p\leq i\leq i_0 \quad j_0-q\leq j\leq j_0$

B-Spline surfaces

- Properties of surface basis functions
  - Extrema
    If $p>0$ and $q>0$, $N_i^p(u)\,N_j^q(v)$ has a unique maximum.
  - Continuity
    Inside rectangles formed by the nodes u_i and v_j, the SF are infinitely differentiable.
    At a node u_i (resp. v_j), $N_i^p(u)\,N_j^q(v)$ is (p-k) (resp. (q-k)) times differentiable, k being the node multiplicity u_i (resp. v_j)
    The continuity with respect to u (resp. v) depends solely on the nodal sequence U (resp. V).

B-Spline surfaces

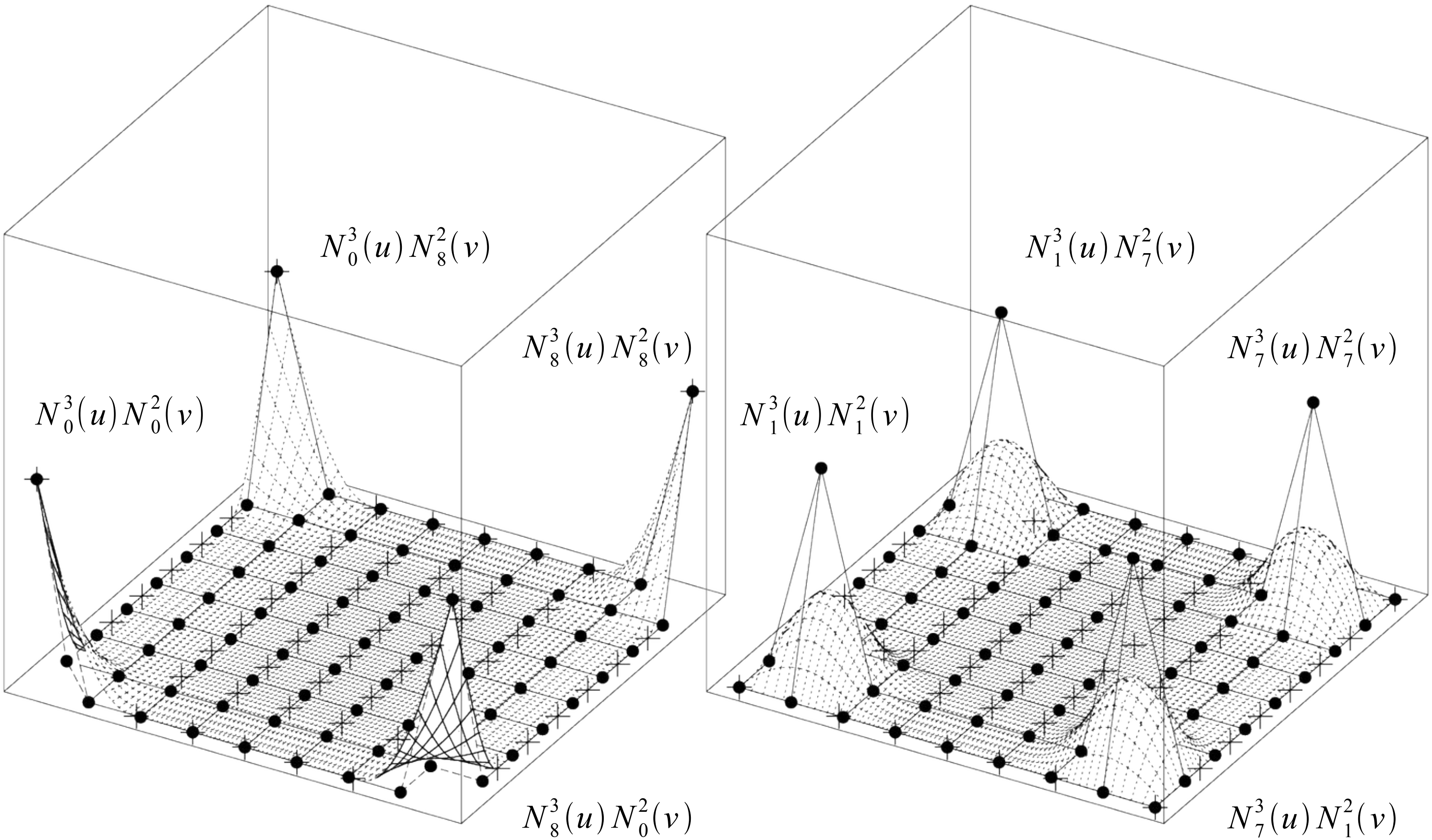




$U=\{0,0,0,0,1,2,2,2,2\}$ $p=3$ $U=\{0,0,0,0,1,2,3,4,5,6,6,6,6\}$

$V=\{0,0,0,1,2,3,3,3\}$ $q=2$ $V=\{0,0,0,1,2,3,4,5,6,7,7,7\}$

B-Spline surfaces



$$U=\{0,0,0,0,1,2,3\ ,4,5,6,6,6,6\} \qquad p=3$$
$$V=\{0,0,0,1,2,3,4,5,6,7,7,7\} \qquad q=2$$

- Computation of a point on the surface

1 – Find the nodal interval in which u is located

$$u \in [u_i, u_{i+1}[$$

2 – Compute the non vanishing 1D shape functions

$$N_{i-p}^{p}(u), \cdots, N_{i}^{p}(u)$$

3 – Find the nodal interval in which v is located

$$v \in [v_j, v_{j+1}[$$

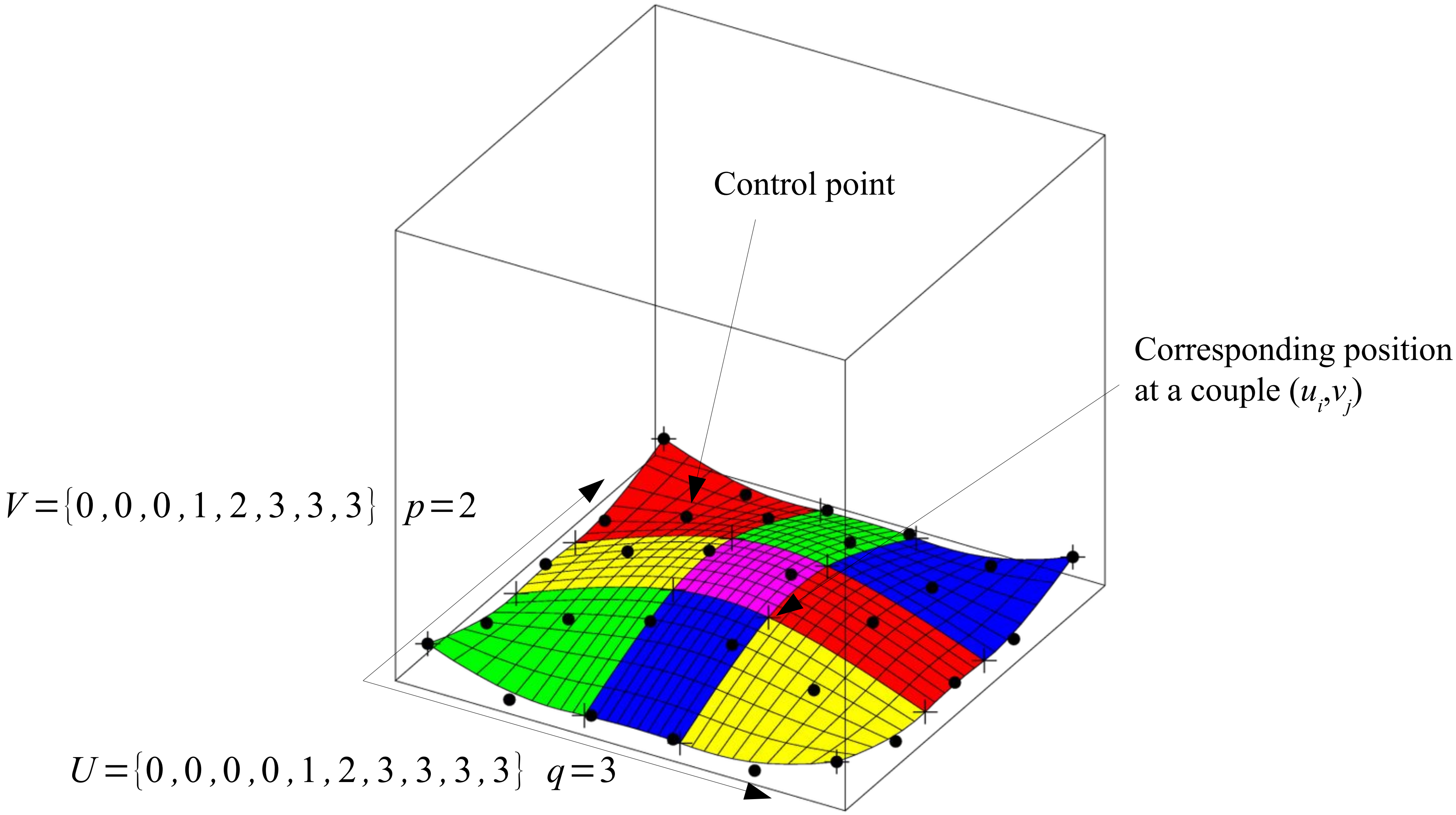
4 – Compute the non vanishing 1D shape functions

$$N_{j-q}^{q}(v), \cdots, N_{j}^{q}(v)$$

5 – Multiply the SFs with the adequate control points

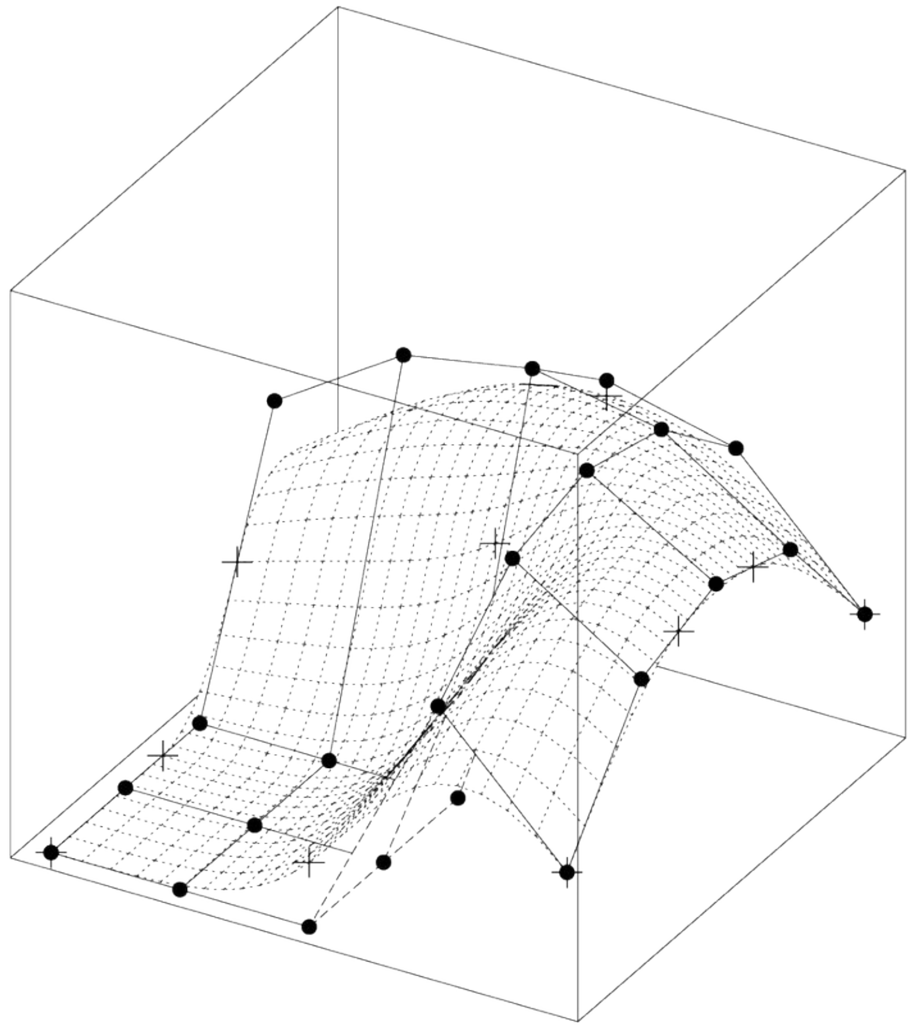
$$S(u, v) = \sum_{k} \sum_{l} N_k^p(u) P_{kl} N_l^q(v) \qquad i-p \le k \le i \quad , \quad j-q \le l \le j$$

B-Spline surfaces



- Each coloured square has an independent polynomial expression

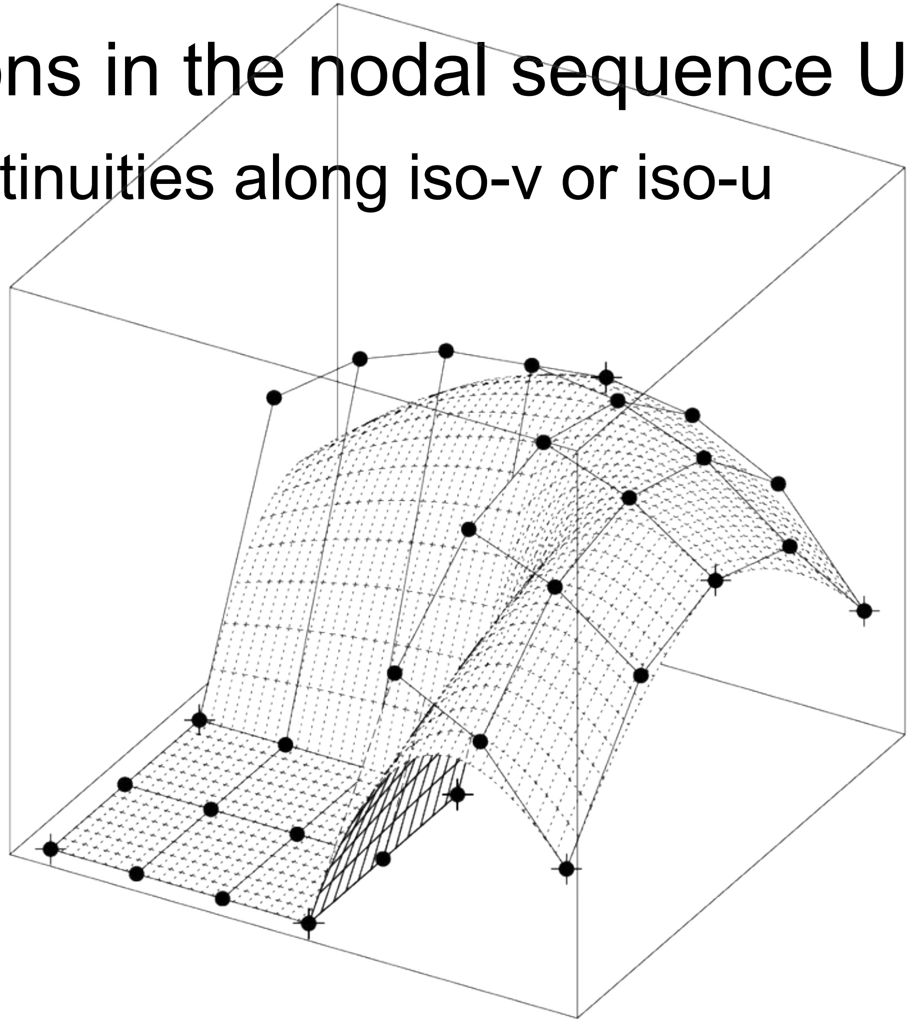
B-Spline surfaces



$$U = \{0,0,0,0,1,2,2,2,2\} \qquad p = 3$$
$$V = \{0,0,0,1,2,3,3,3\} \qquad q = 2$$

B-Spline surfaces

- Repetitions in the nodal sequence U or V
  - Discontinuities along iso-v or iso-u



$$U=\{0,0,0,0,2,2,2,4,4,4,4\} \qquad p=3$$
$$V=\{0,0,0,1.5,1.5,3,3,3\} \qquad q=2$$

- Properties of the B-Spline surface
  - Interpolate the 4 corners if the nodal sequences are of the form

$$U=\{\underbrace{0,\cdots,0}_{p+1},u_{p+1},\cdots,u_{r-p-1},\underbrace{1,\cdots,1}_{p+1}\}$$

$$V=\{\underbrace{0,\cdots,0}_{q+1},v_{q+1},\cdots,v_{s-q-1},\underbrace{1,\cdots,1}_{q+1}\}$$

  - If the nodal sequences correspond to Bézier curves : $U=\{\underbrace{0,\cdots,0}_{p+1},\underbrace{1,\cdots,1}_{p+1}\}$ $V=\{\underbrace{0,\cdots,0}_{q+1},\underbrace{1,\cdots,1}_{q+1}\}$

    then the surface is called a Bézier patch.

- Properties of the B-Spline surface
  - The surface has the property of affine invariance (invariance by translation in particular)
  - The convex hull of the control points contains the surface.
  - In every interval $(u,v)\in[u_{i_0},u_{i_0+1}[\times[v_{j_0},v_{j_0+1}[$, the portion of the surface is in the convex hull of the control points P_{ij} , (i,j) such that $i_0-p\leq i\leq i_0 \quad j_0-q\leq j\leq j_0$
  - Control points may have a local control
  - There is **no** variation diminishing property (on the contrary to B-Spline/Bézier curves)

- Isoparametrics
  - Computation of isoparametrics is easy : Set $u=u_0$

$$C_{u_0}(v)=S(u_{0,}v)=\sum_{i=0}^{n}\sum_{j=0}^{m}N_i^p(u_0)N_j^q(v)P_{ij}$$

$$=\sum_{j=0}^{m}N_j^q(v)\sum_{i=0}^{n}N_i^p(u_0)P_{ij}=\sum_{j=0}^{m}N_j^q(v)Q_j(u_0)$$

$$\text{with } Q_j(u_0)=\sum_{i=0}^{n}N_i^p(u_0)P_{ij}$$

  - same with $v=v_0$

$$C_{v_0}(u)=S(u,v_0)=\sum_{i=0}^{n}N_i^p(u)Q_i(v_0)$$

$$\text{with } Q_i(v_0)=\sum_{j=0}^{m}N_j^q(v_0)P_{ij}$$

- Derivatives of a B-Spline surface
  - We want to compute $\dfrac{\partial^{k+l}}{\partial u^k \partial v^l} S(u,v)$
  - Differentiation of basis functions :

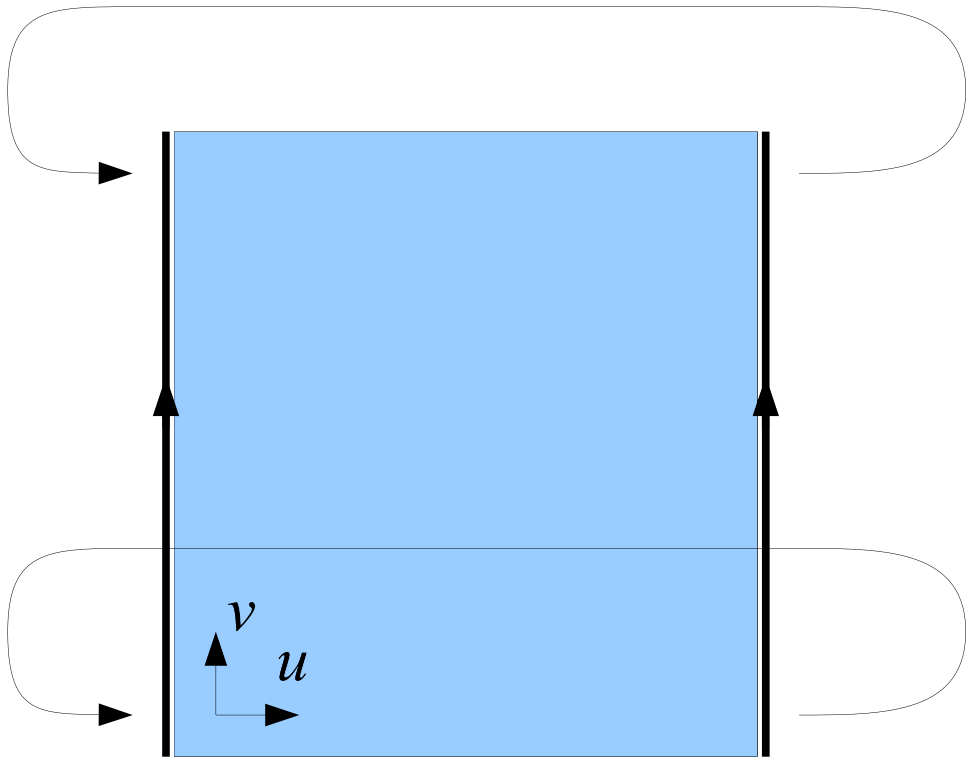
$$\frac{\partial^{k+l}}{\partial u^k \partial v^l} S(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} \frac{\partial^{k+l}}{\partial u^k \partial v^l} N_i^p(u) N_j^q(v) P_{ij}$$

$$= \sum_{i=0}^{n} \sum_{j=0}^{m} \frac{\partial^{k}}{\partial u^k} N_i^p(u) \frac{\partial^{l}}{\partial v^l} N_j^q(v) P_{ij}$$

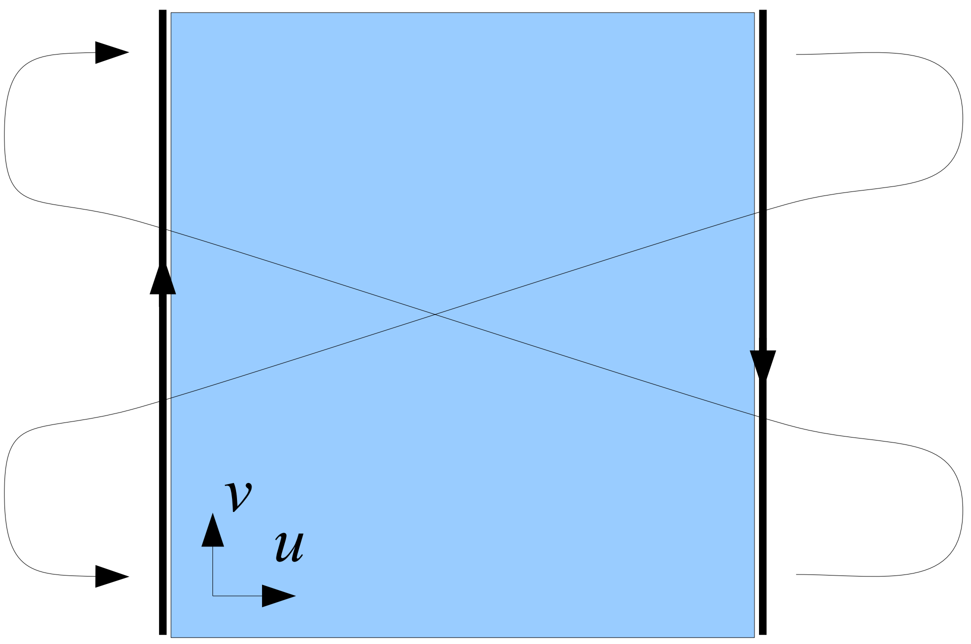
$$= \sum_{i=0}^{n} \sum_{j=0}^{m} N_i^{p(k)}(u) N_j^{q(l)}(v) P_{ij}$$

- Periodic surfaces
  - Like for the curves, possibility to “close” a B-Spline surface by transforming the nodal sequence
    - According to one parameter (u or v)
      Cylindrical surfaces
      - A single periodic nodal sequence
      - Control points on both sides of the seam are doubled
    - According to both parameters (u and v)
      Toroidal surfaces
      - Two periodic nodal sequences
      - Some control points are repeated 4 times !

B-Spline surfaces

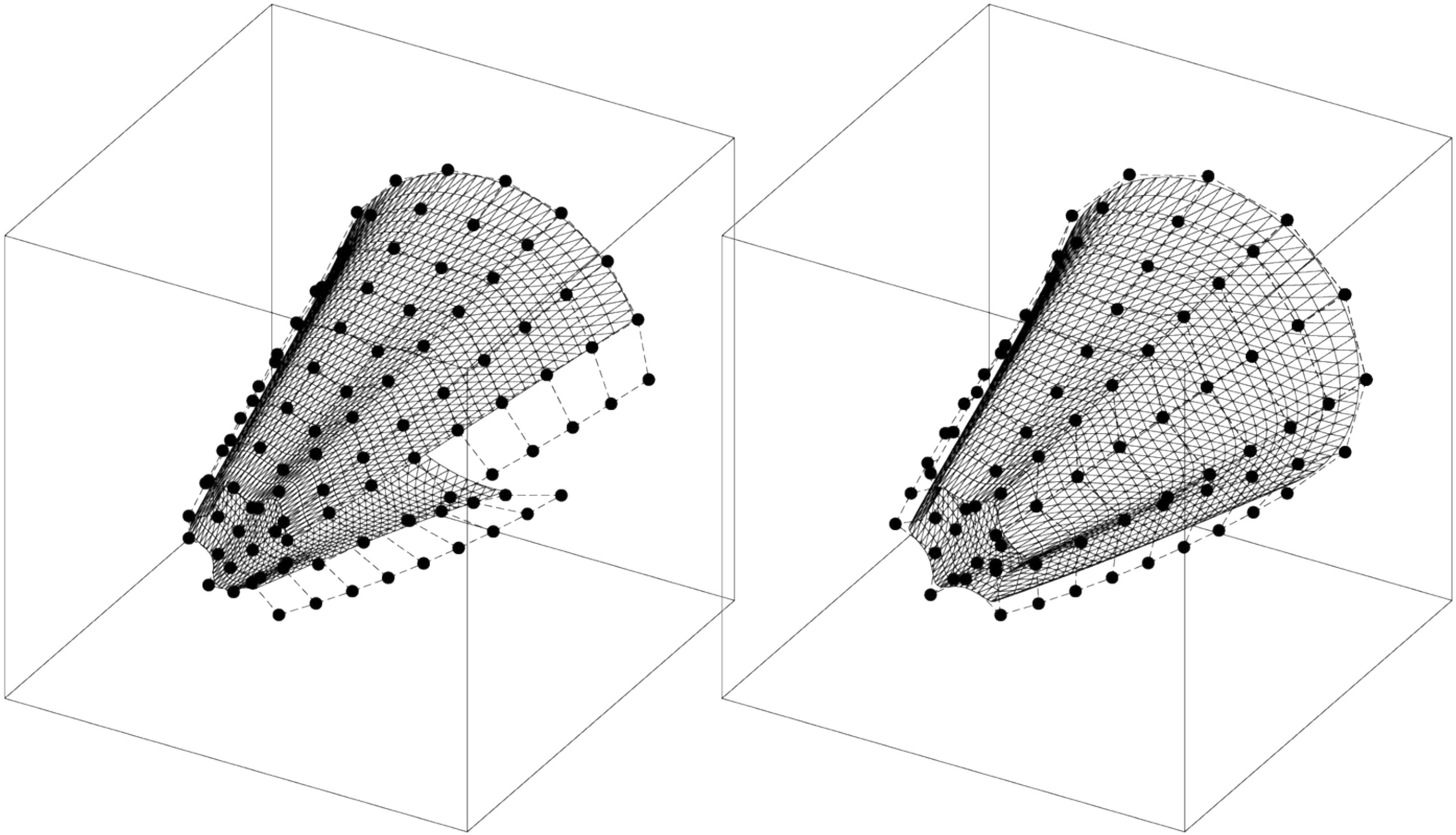


Pipe



Möbius strip

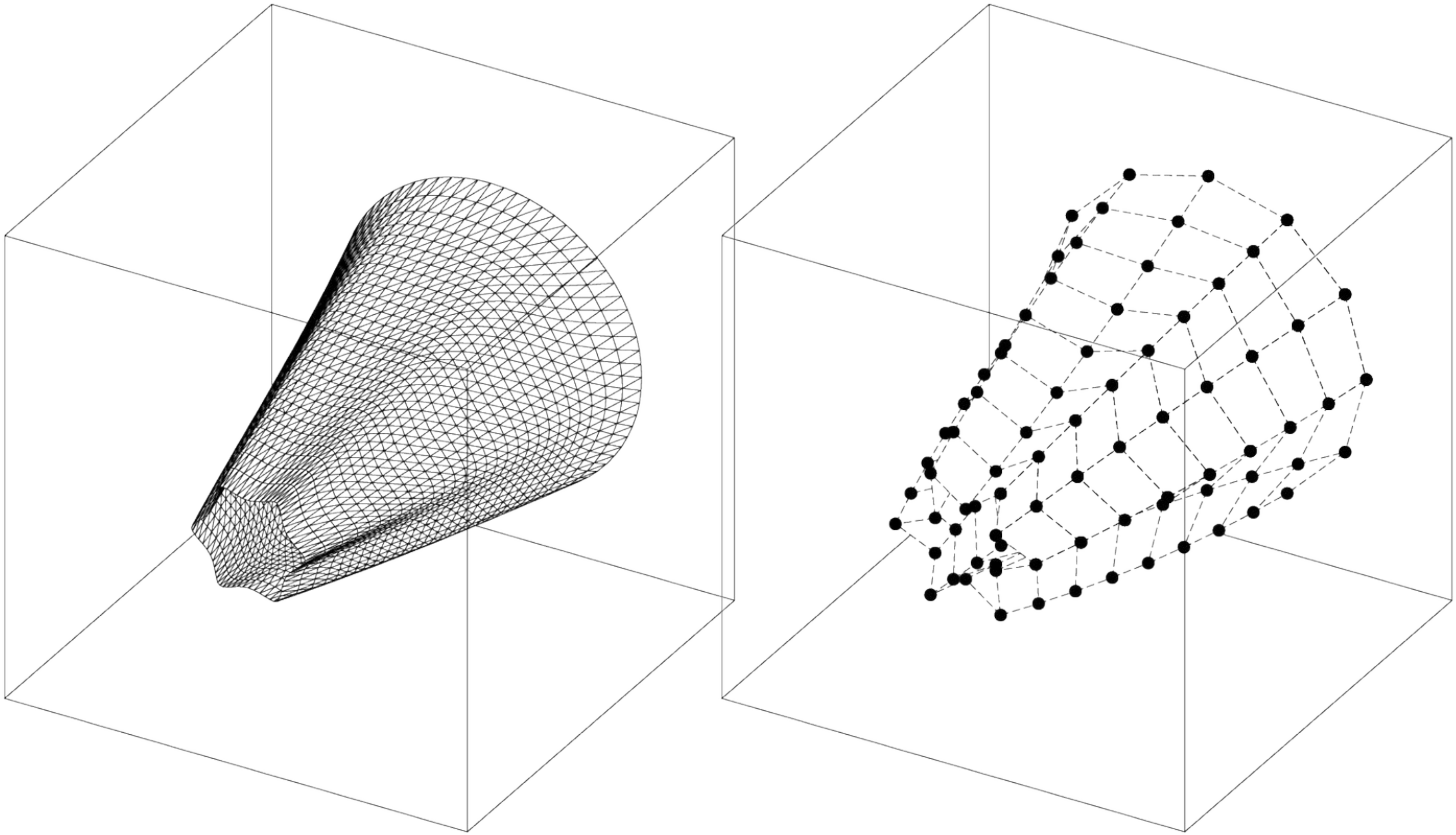
B-Spline surfaces



$$U=\{-3,-2,-1,0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15\} \quad p=3$$
$$V=\{0,0,0,1,2,3,4,5,6,7,7,7\} \quad p=2$$

CAD & Computational Geometry

B-Spline surfaces

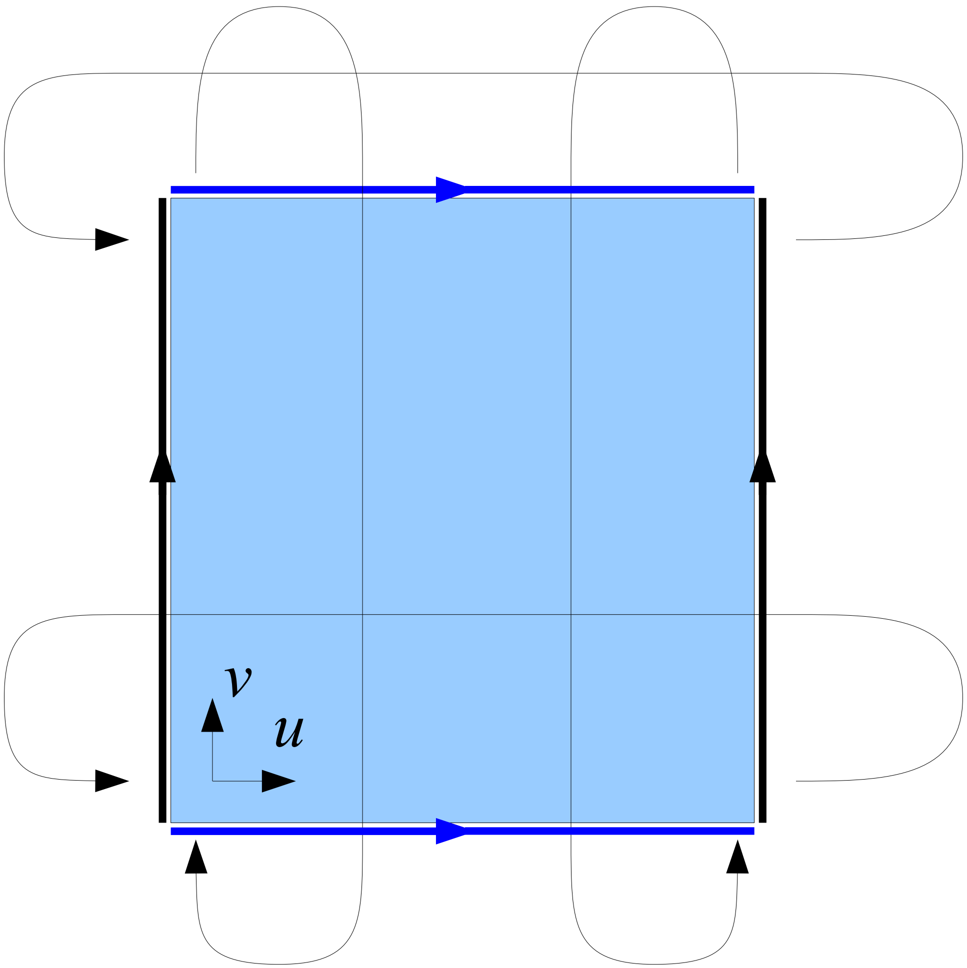


$$U = \{-3, -2, -1, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15\} \quad p = 3$$

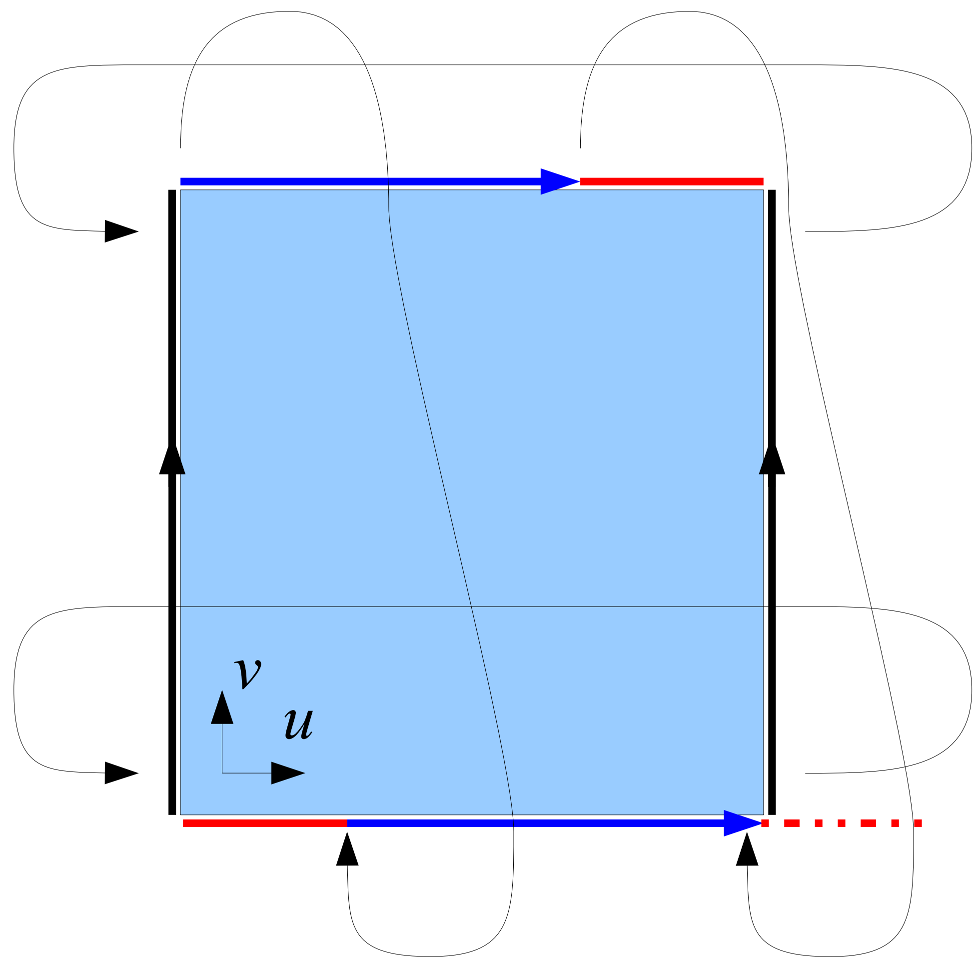
$$V = \{0, 0, 0, 1, 2, 3, 4, 5, 6, 7, 7, 7\} \quad p = 2$$

CAD & Computational Geometry

B-Spline surfaces

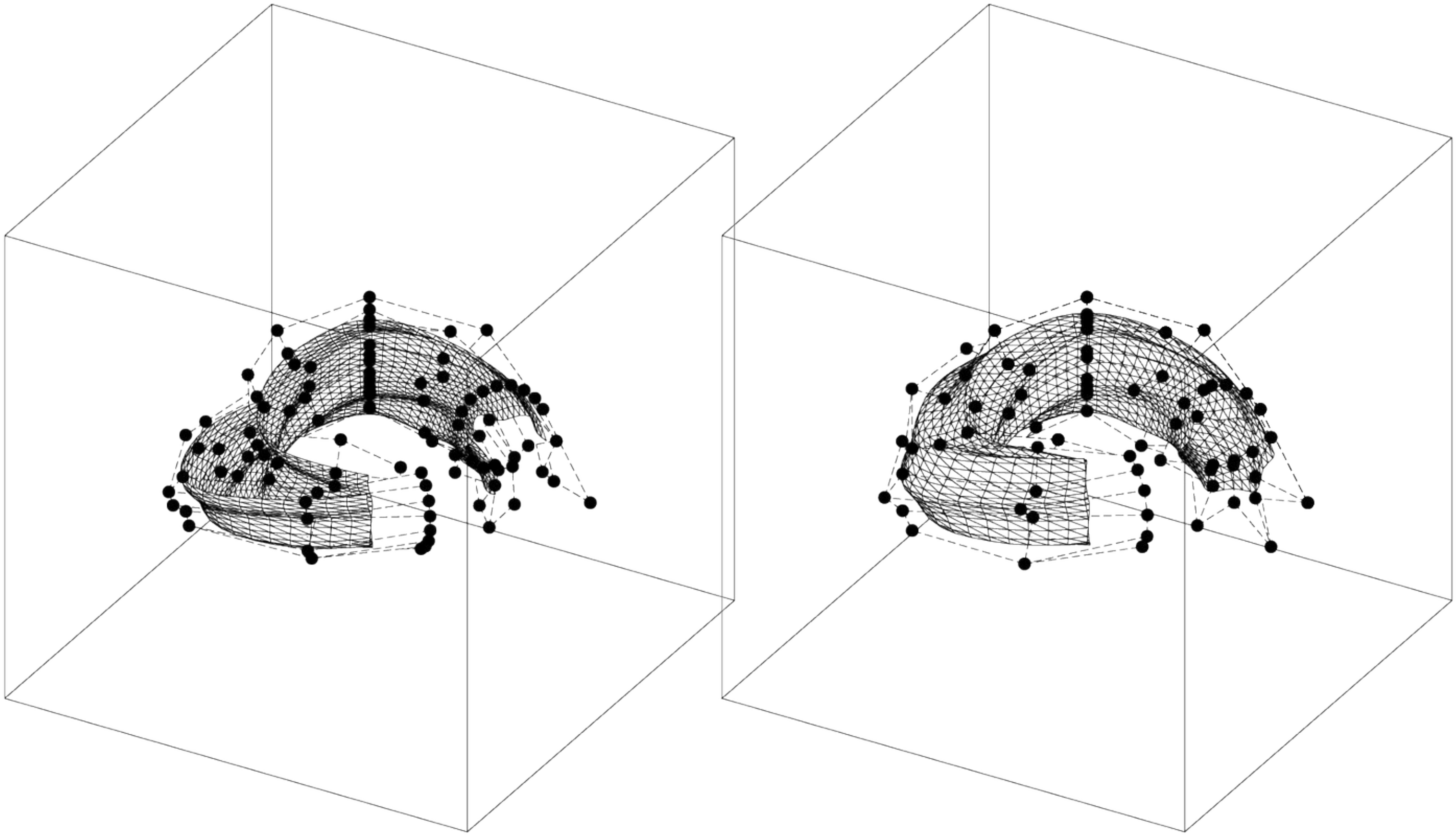


Tore

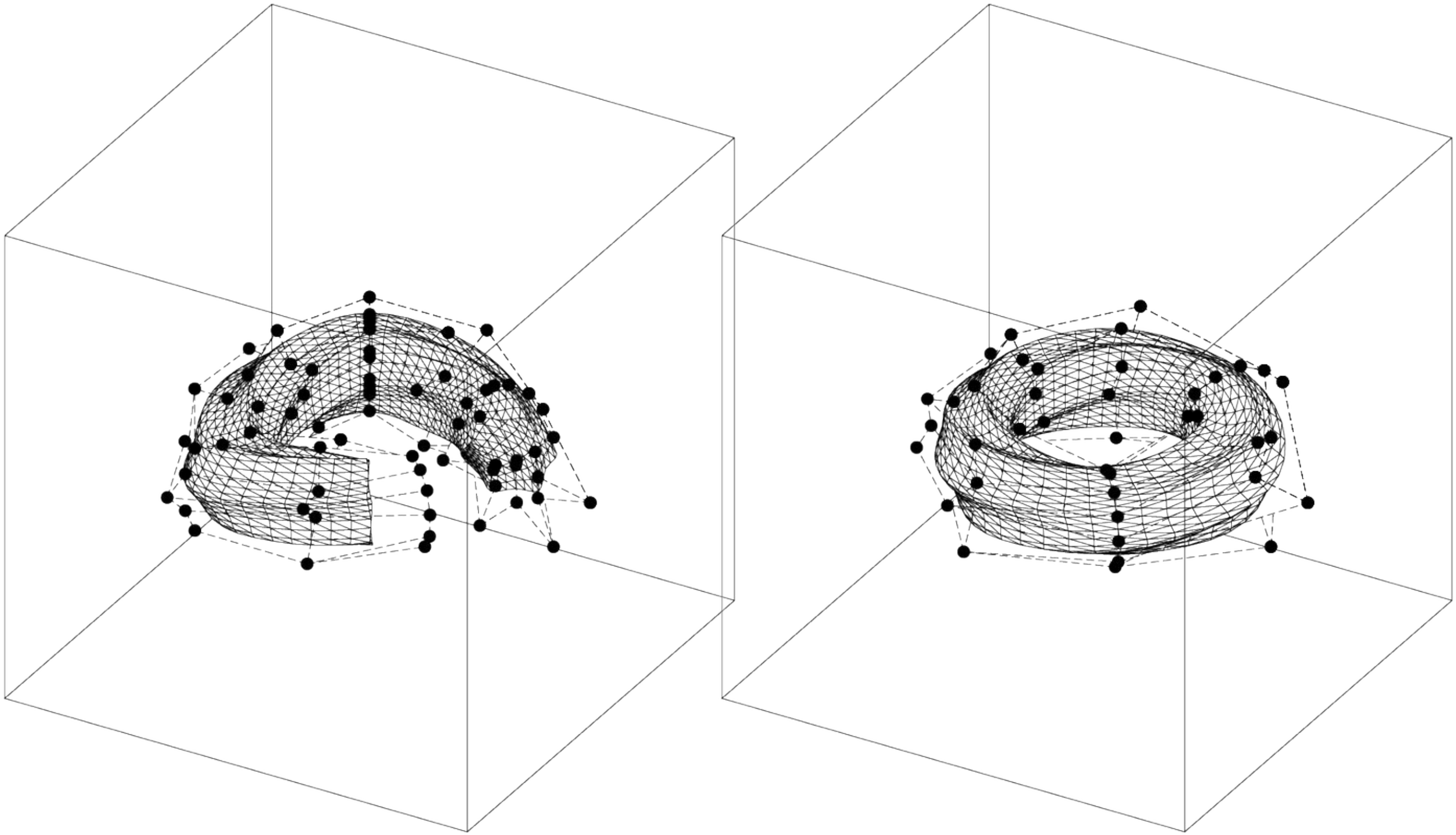


Twisted Tore...

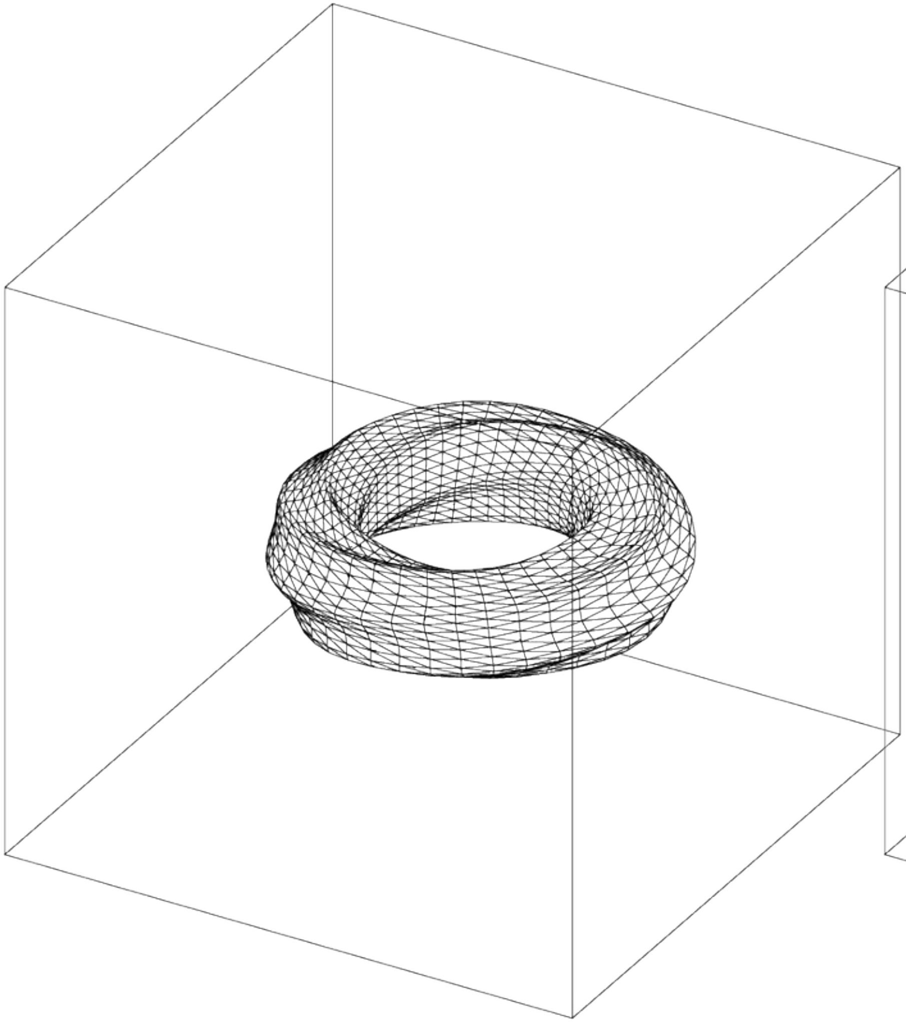
B-Spline surfaces

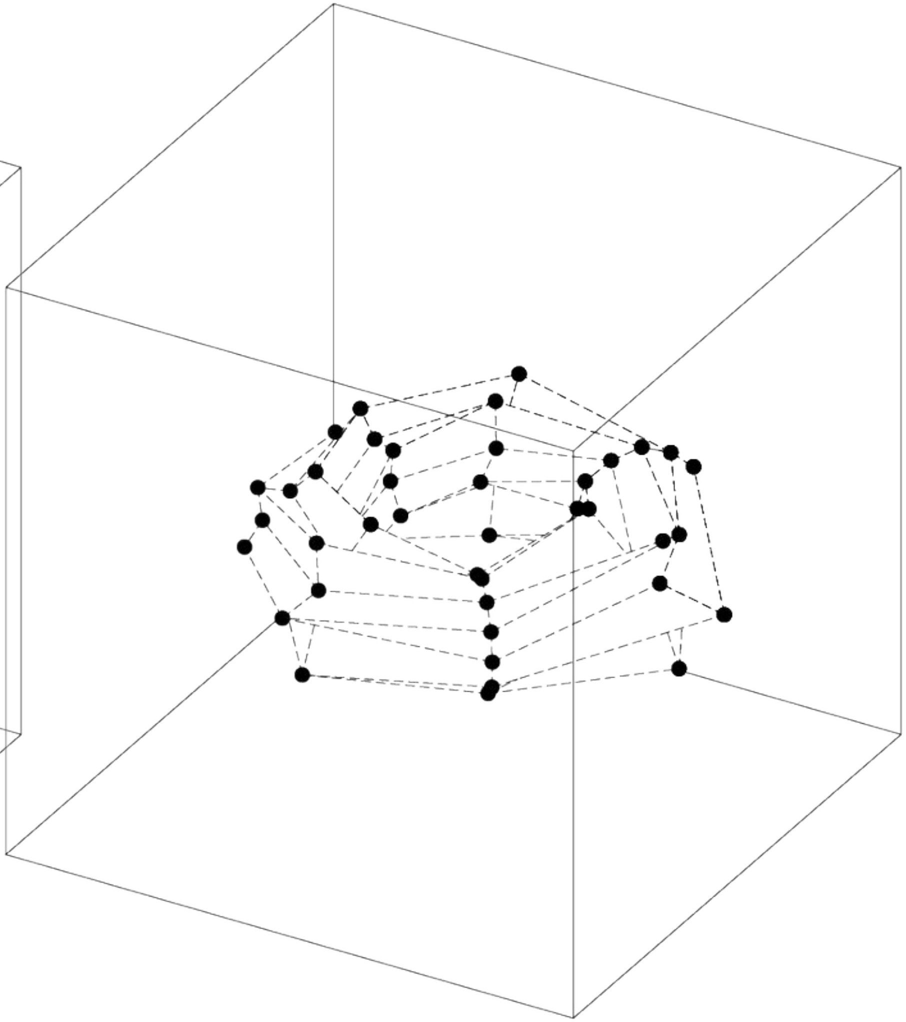


B-Spline surfaces



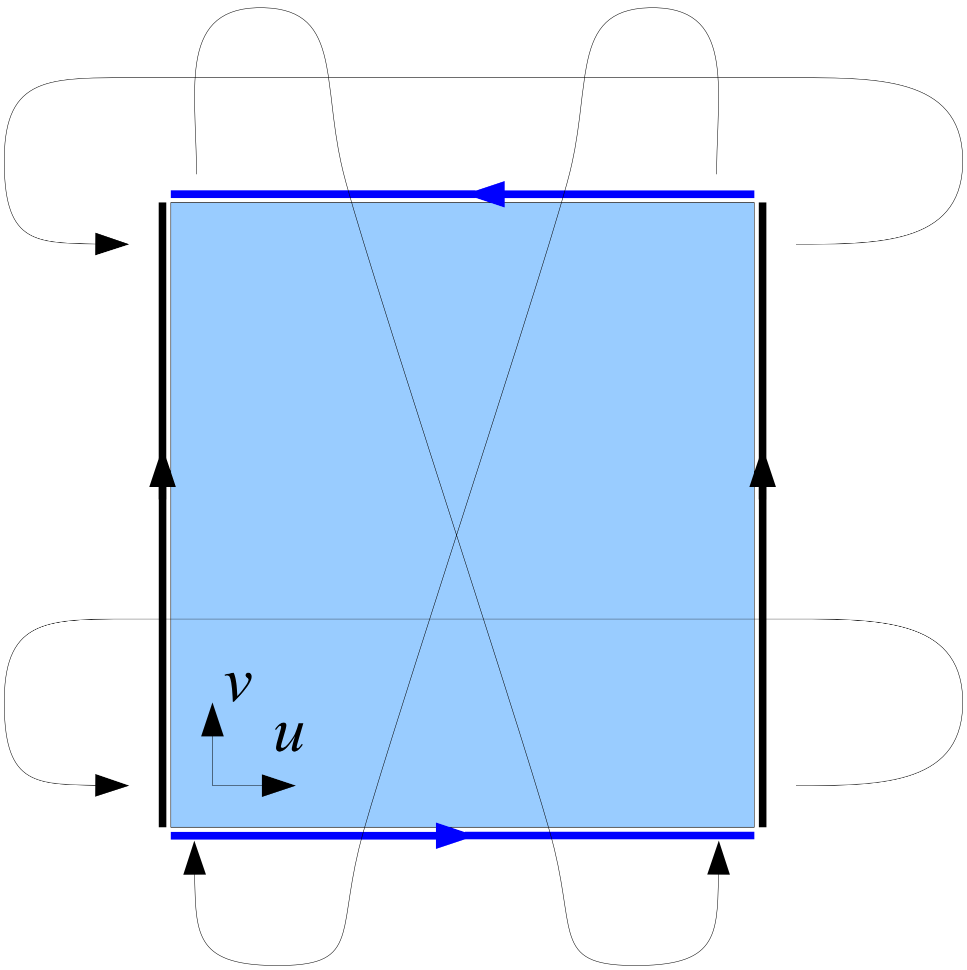
B-Spline surfaces



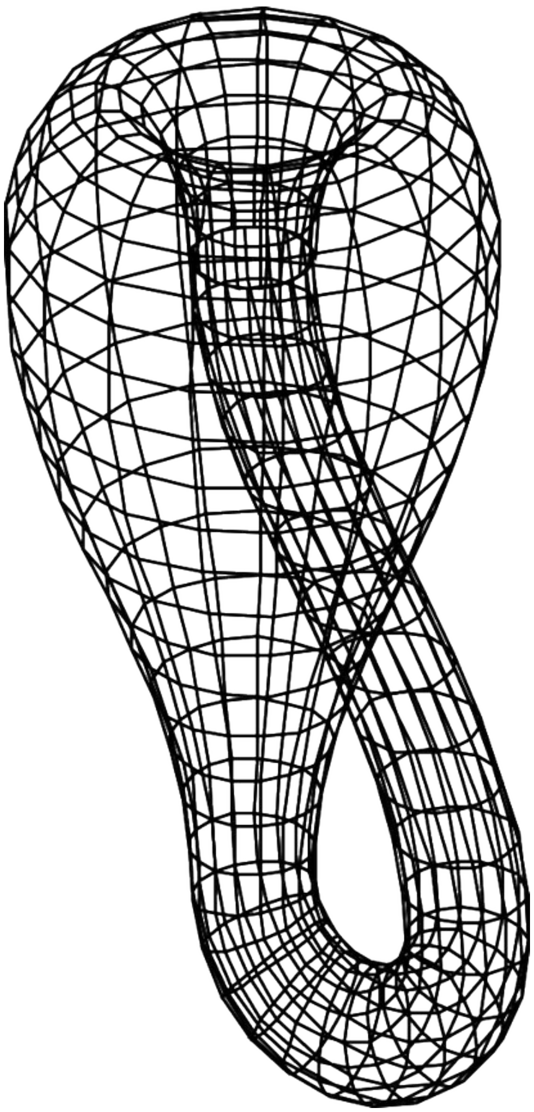


$$U=\{-3,-2,-1,0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15\} \quad p=3$$
$$V=\{-2,-1,0,1,2,3,4,5,6,7,8,9\} \quad p=2$$

B-Spline surfaces



Klein's bottle



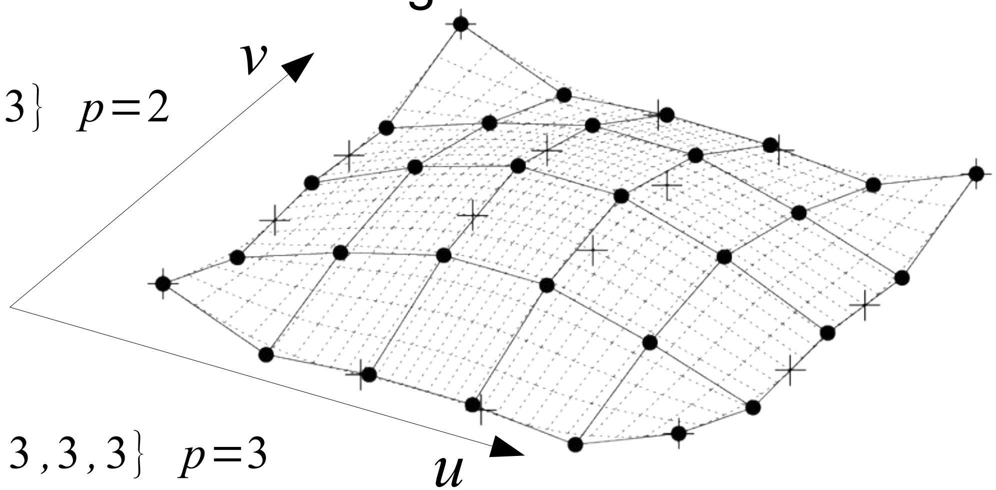
B-Spline surfaces

- Some operations
  - Insertion of nodes
    - Extraction of iso-parametrics
    - Obtaining the position of a point on the surface
    - Subdivision of the surface
    - Transformation into Bézier patches

B-Spline surfaces

- Insertion of nodes
  - We insert nodes In a nodal sequence (U ou V)
  - The new nodal sequence replace the old one
  - The control points are modified
    - If U is modified, every series of control points corresponding to $v=cst$ is independently modified
    - If V is modified, every series of control points corresponding to $u=cst$ is independently modified
    - We use Boehm's algorithm as for curves

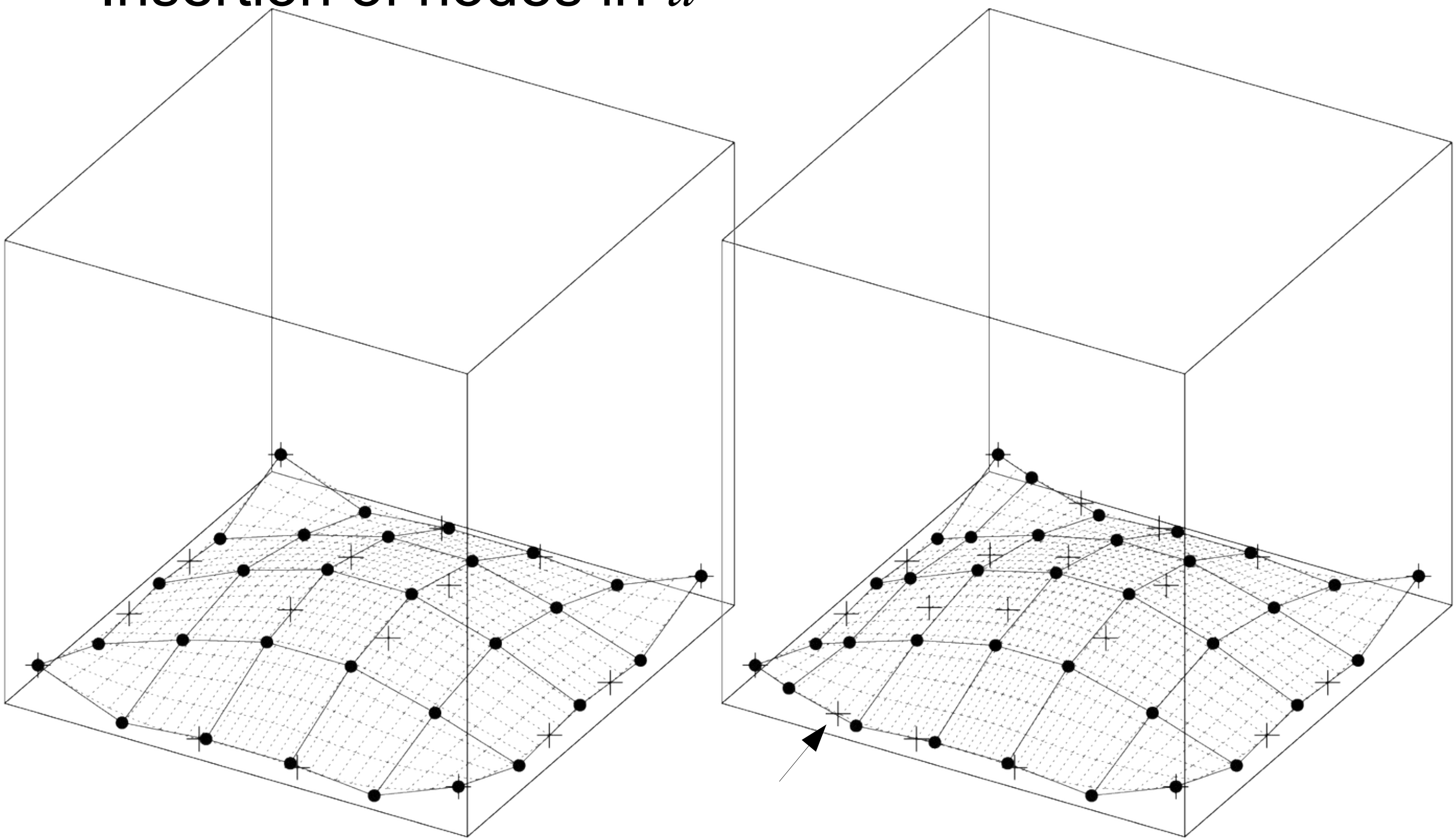
$$V=\{0,0,0,1,2,3,3,3\} \quad p=2$$



$$U=\{0,0,0,0,1,2,3,3,3,3\} \quad p=3$$

B-Spline surfaces

- Insertion of nodes in u



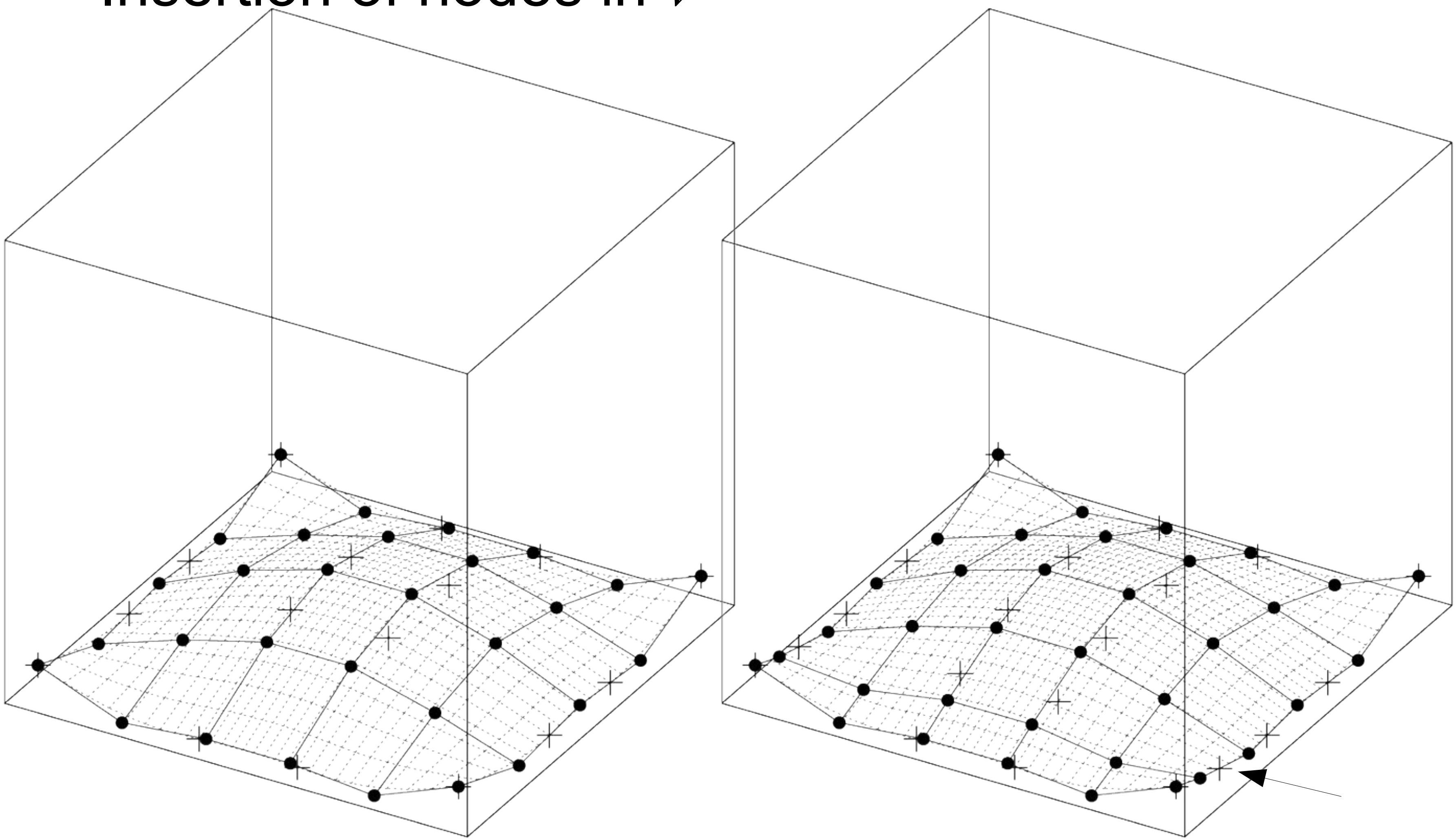
$U = \{0,0,0,0,1,2,3,3,3,3\}$ $p=3$
$V = \{0,0,0,1,2,3,3,3\}$ $q=2$

$U = \{0,0,0,0,0.4,1,2,3,3,3,3\}$ $p=3$
$V = \{0,0,0,1,2,3,3,3\}$ $q=2$

CAD & Computational Geometry

B-Spline surfaces

- Insertion of nodes in v



$U=\{0,0,0,0,1,2,3,3,3,3\}\quad p=3$
$V=\{0,0,0,1,2,3,3,3\}\quad q=2$

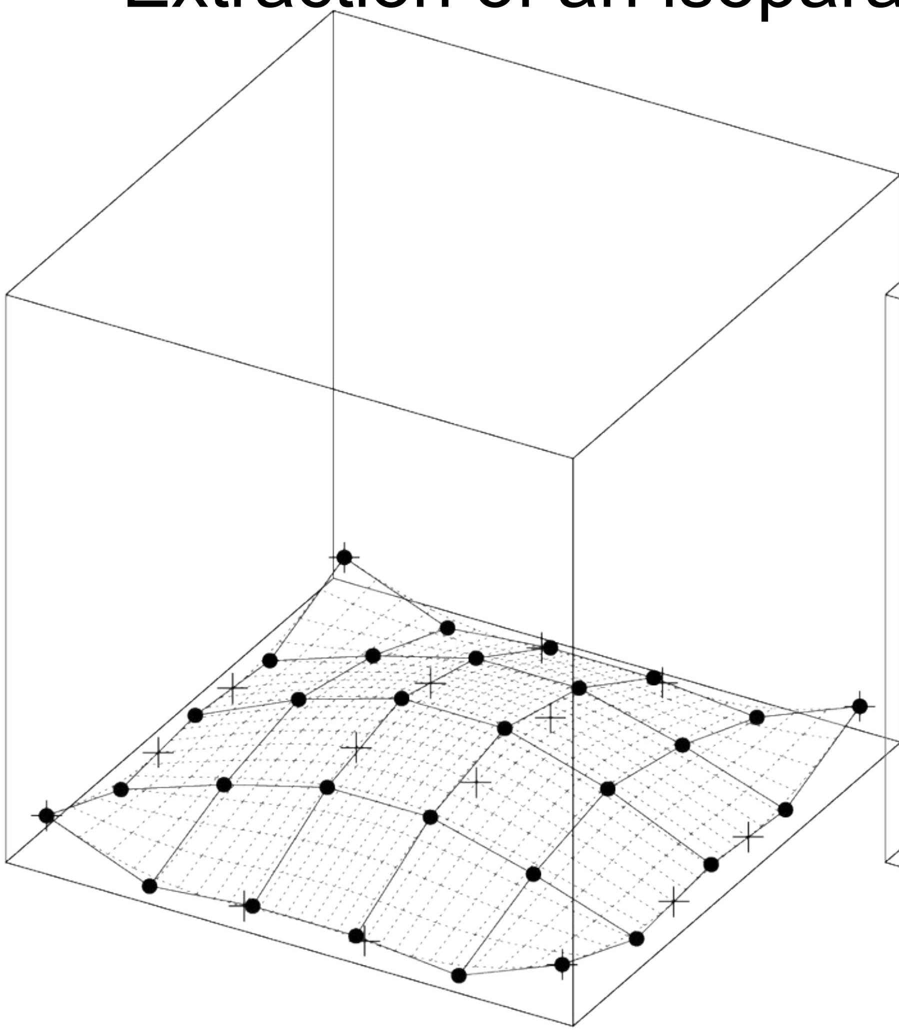
$U=\{0,0,0,0,1,2,3,3,3,3\}\quad p=3$
$V=\{0,0,0,0.4,1,2,3,3,3\}\quad q=2$

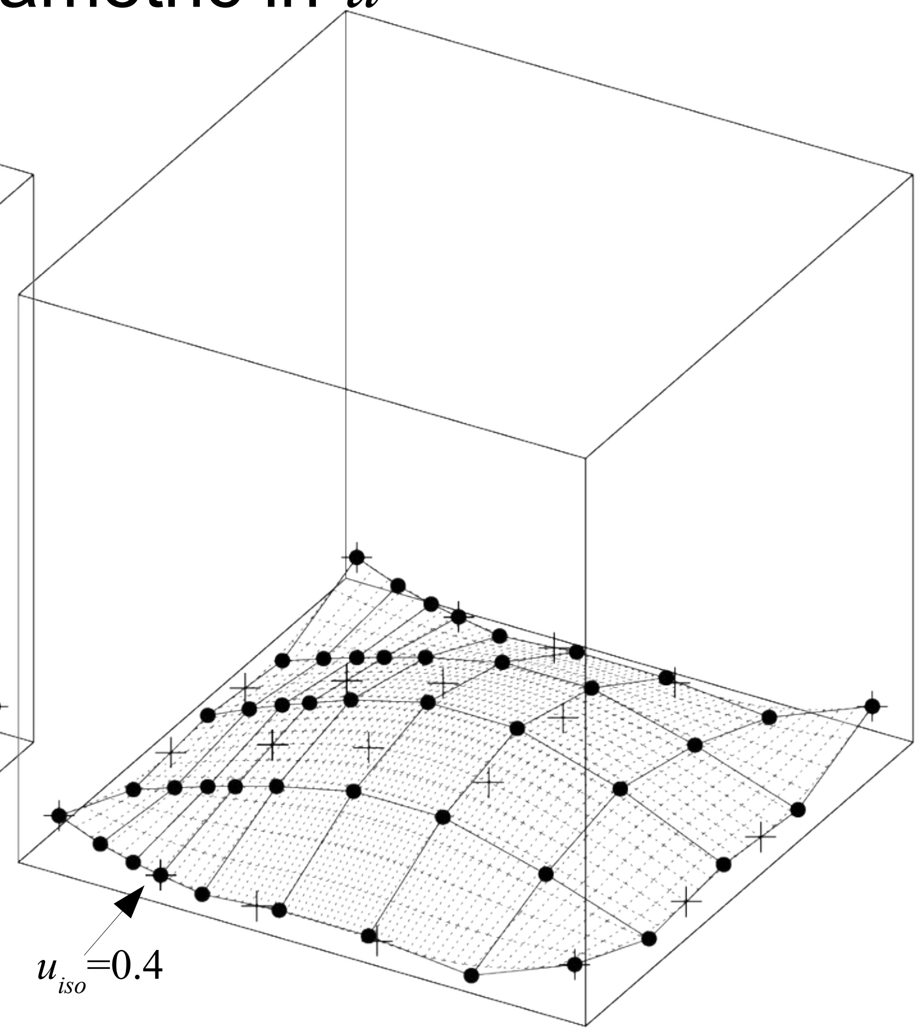
B-Spline surfaces

- Extraction of iso-parametrics using node insertion
  - We must saturate one node in $u=u_{iso}$ (resp. in $v=v_{iso}$).
  - The new control points obtained by Boehm's algorithm do form the control polygon of the iso-parametric curve.
  - The nodal sequence of this curve is V (resp. U).

B-Spline surfaces

- Extraction of an isoparametric in u



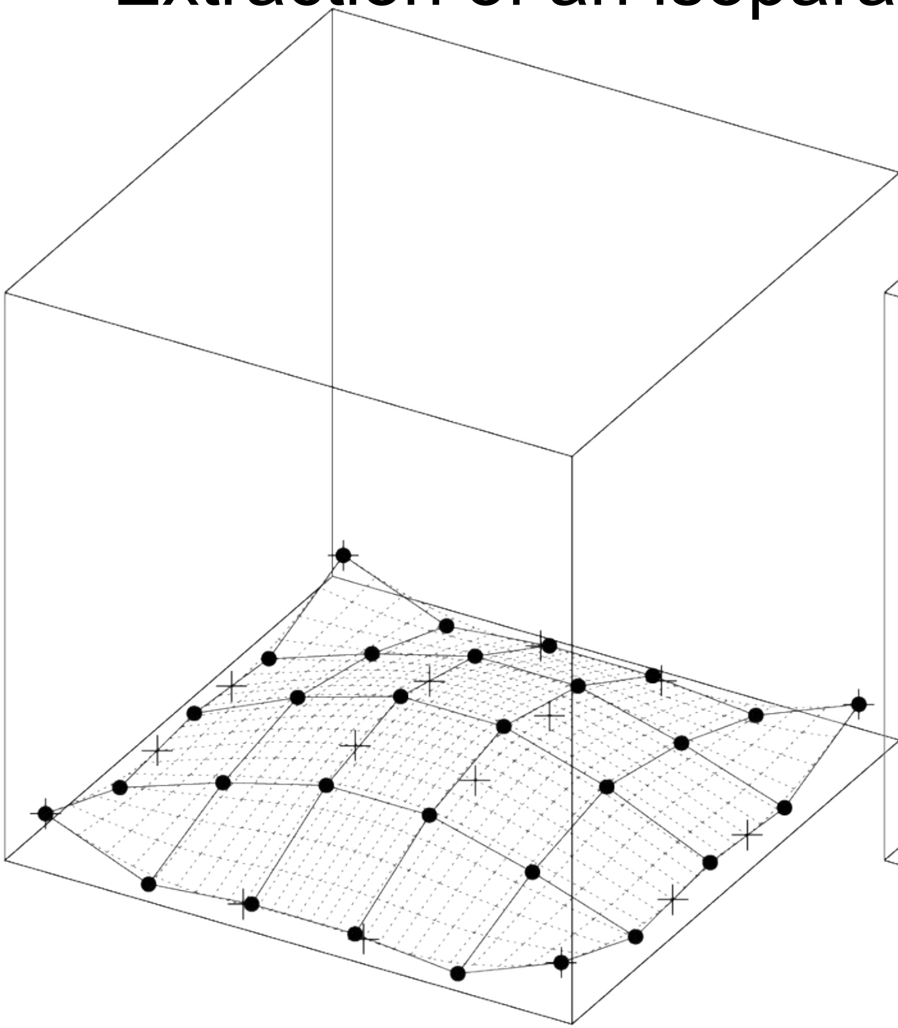


$U=\{0,0,0,0,1,2,3,3,3,3\}\quad p=3$
$V=\{0,0,0,1,2,3,3,3\}\quad q=2$

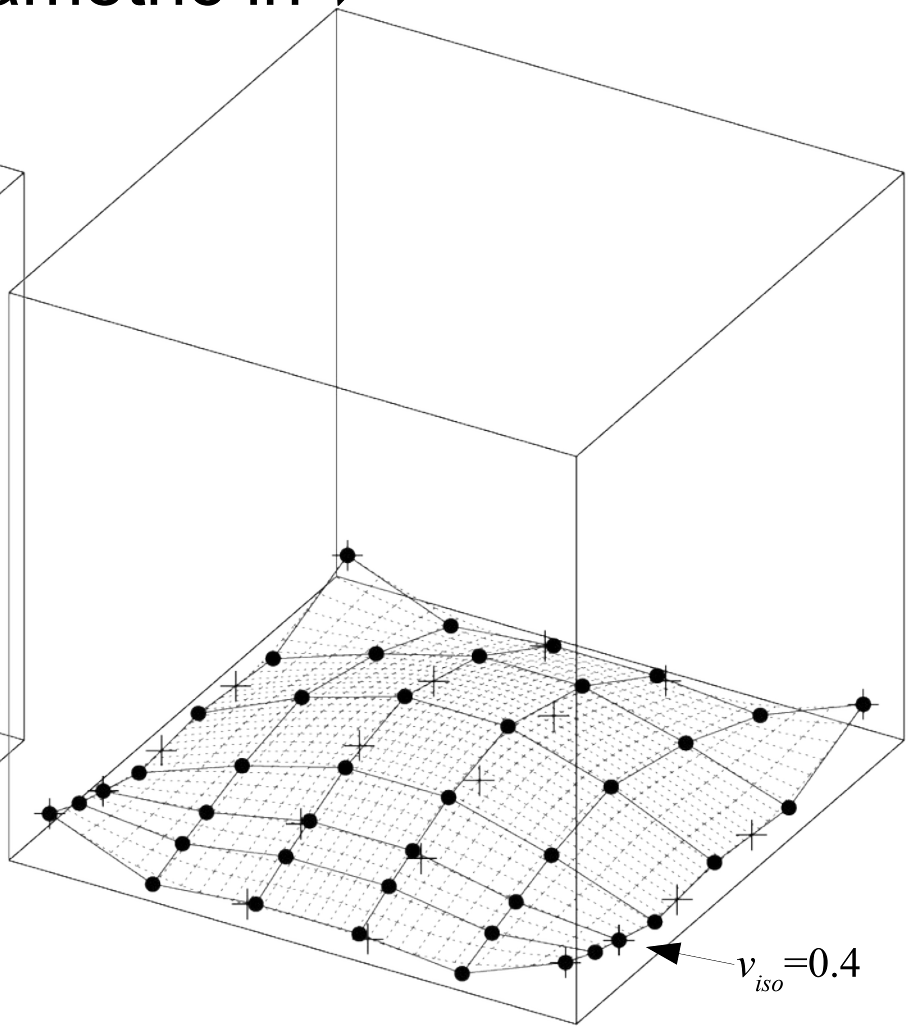
$U=\{0,0,0,0,0.4,0.4,0.4,1,2,3,3,3,3\}\quad p=3$
$V=\{0,0,0,1,2,3,3,3\}\quad q=2$

B-Spline surfaces

- Extraction of an isoparametric in v



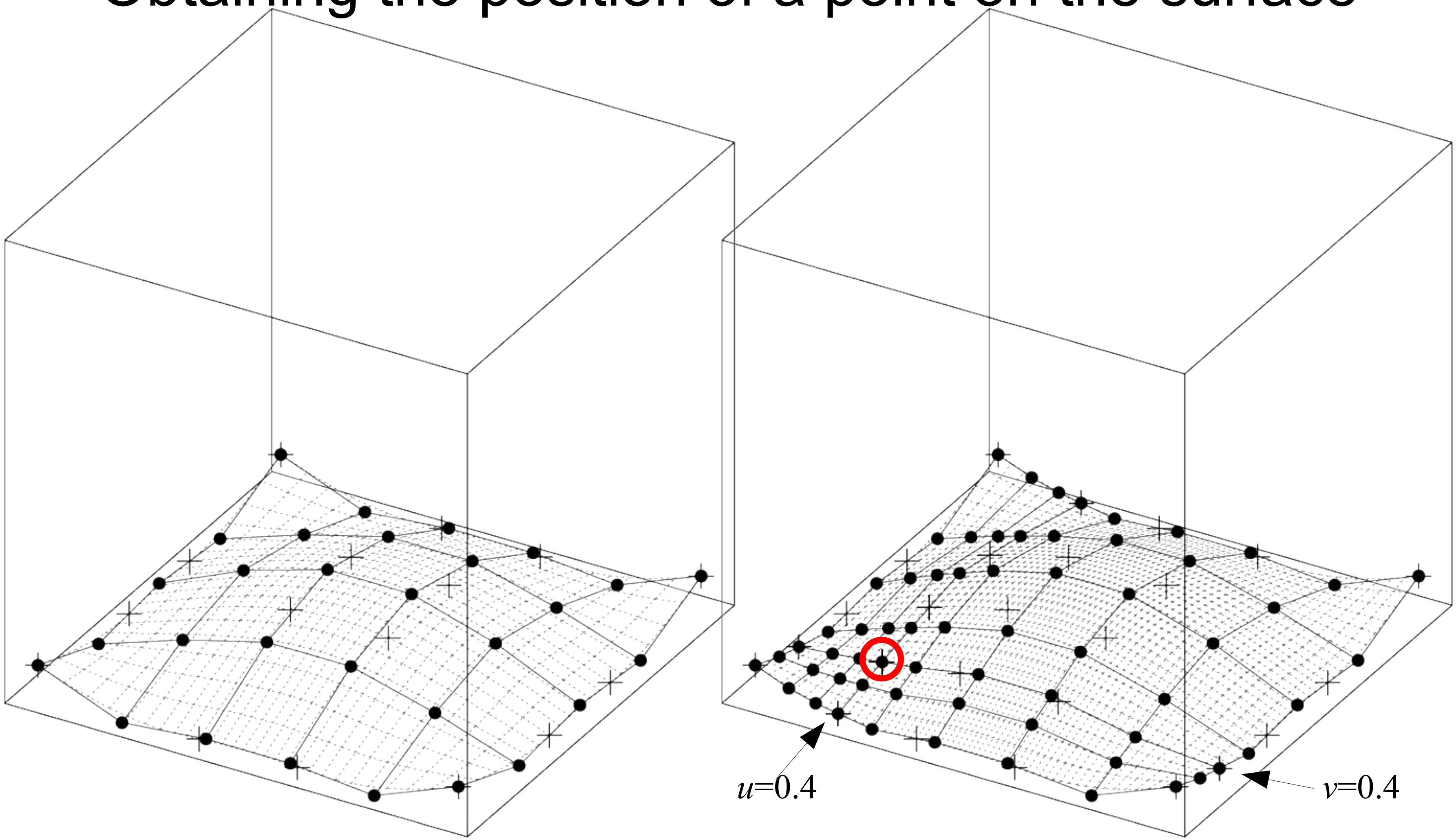
$U=\{0,0,0,0,1,2,3,3,3,3\}\quad p=3$
$V=\{0,0,0,1,2,3,3,3\}\quad q=2$



$U=\{0,0,0,0,1,2,3,3,3,3\}\quad p=3$
$V=\{0,0,0,0.4,0.4,1,2,3,3,3\}\quad q=2$

B-Spline surfaces

- Obtaining the position of a point on the surface

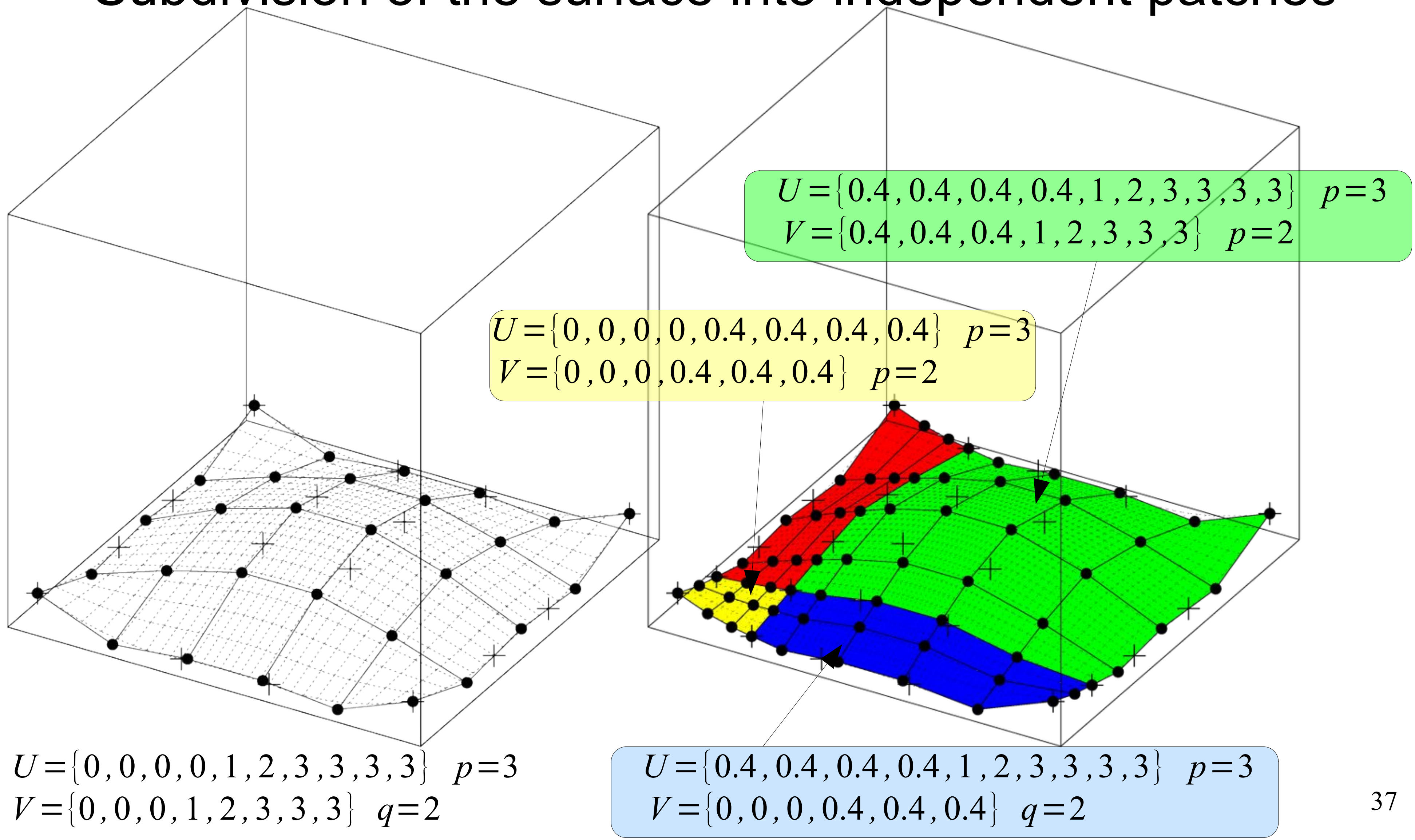


$U=\{0,0,0,0,1,2,3,3,3,3\} \quad p=3$
$V=\{0,0,0,1,2,3,3,3\} \quad q=2$

$U=\{0,0,0,0,0.4,0.4,0.4,1,2,3,3,3,3\} \quad p=3$
$V=\{0,0,0,0.4,0.4,1,2,3,3,3\} \quad q=2$

B-Spline surfaces

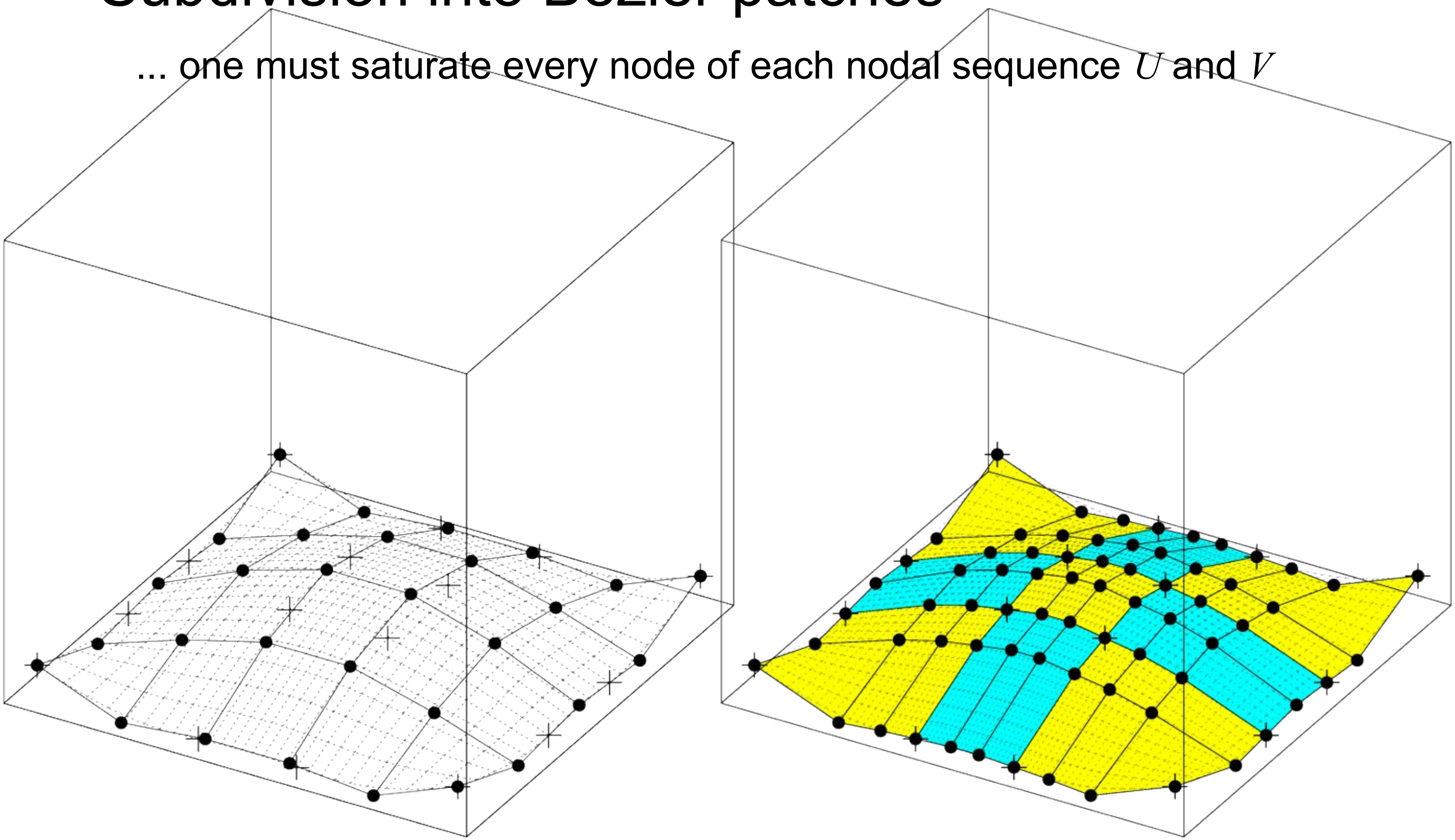
- Subdivision of the surface into independent patches



B-Spline surfaces

- Subdivision into Bézier patches

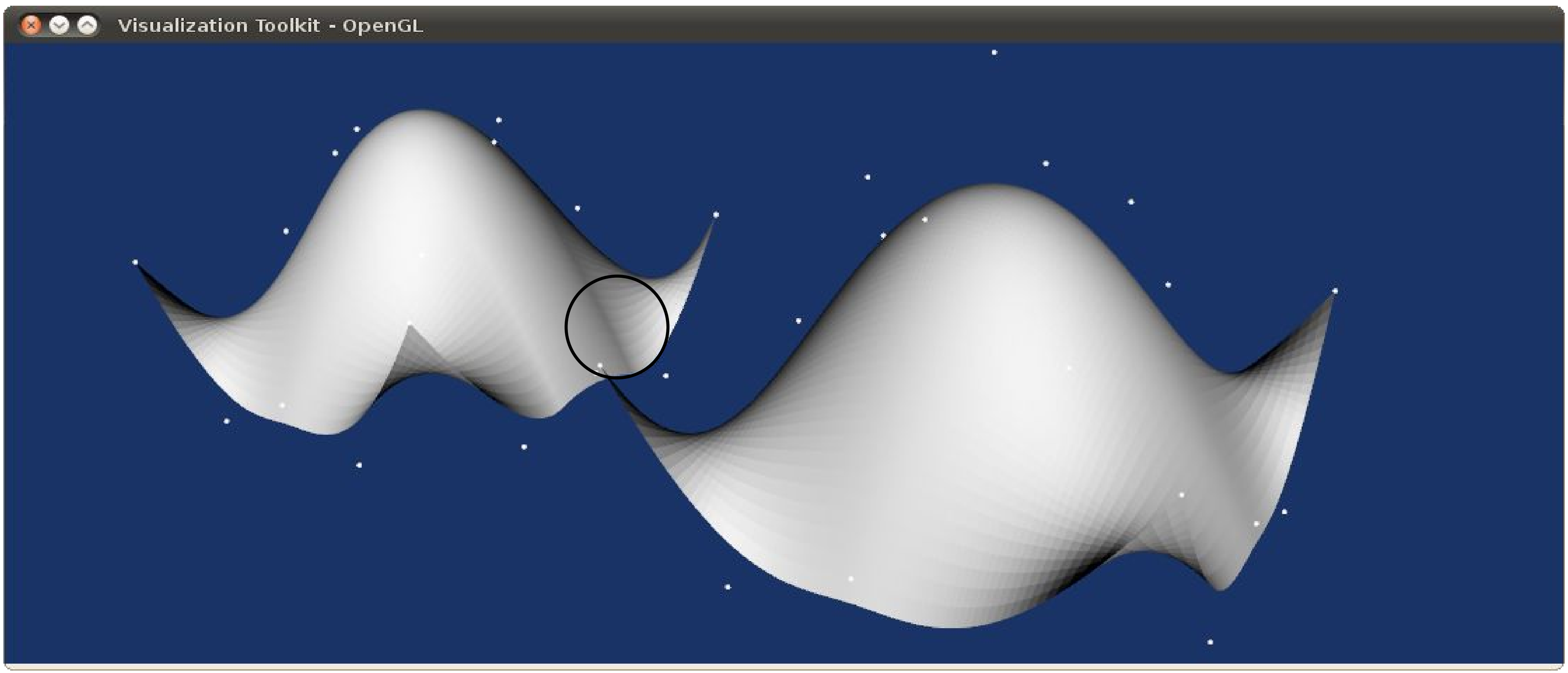
  ... one must saturate every node of each nodal sequence U and V



$U = \{0,0,0,0,1,2,3,3,3,3\}$ $p = 3$
$V = \{0,0,0,1,2,3,3,3\}$ $q = 2$

$U = \{0,0,0,0,1,1,1,2,2,2,3,3,3,3\}$ $p = 3$
$V = \{0,0,0,1,1,2,2,3,3,3\}$ $q = 2$

- Continuity requirements for surfaces
  - C^1 vs C^2 – becomes visible when light interaction comes into play



Subdivision surfaces

- Parametric surfaces: an explicit representation
  - Lightweight
  - Discretization algorithms are non trivial... but it is necessary for display purposes and in computer graphics
  - Generally, these surfaces are used in cases where the geometric accuracy is essential, as in the computation of intersections and other precise geometric primitives
  - **Modeling operators are not trivial**
  - In computer graphics, such accuracy is generally not needed.

- Subdivision surfaces
  - Modelling basis = elementary mesh
  - By successive iterations, this mesh is refined up to the accuracy needed for the application
  - It is more like an algorithmic description vs. an algebraic representation, because the algorithm that is used to subdivide the mesh determines the final shape and the properties of the limiting surface (*ie.* when the number of subdivisions tends to the infinite)
  - Some of these limiting surfaces are equivalent to “regular” parametric surfaces, therefore have the same “accuracy”.

- History
  - 1974 – George Chaikin

    An algorithm for high speed curve generation
  - 1978 – Daniel Doo & Malcolm Sabin

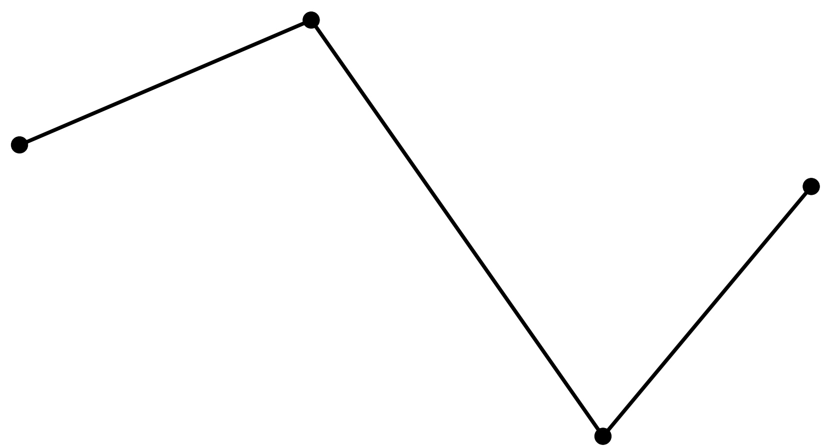
    (D) A subdivision algorithm for smoothing irregularly shaped polyhedrons
    (D&S) Behaviour of recursive division surfaces near extraordinary points.
  - 1978 – Edwin Catmull & Jim Clark

    Recursively generated B-Spline surfaces on arbitrary topological meshes
  - 1987 – Charles Loop

    Smooth subdivision surfaces based on triangles
  - 2000 – Leif Kobbelt

    $\sqrt{3}$ – subdivision (interpolating scheme)

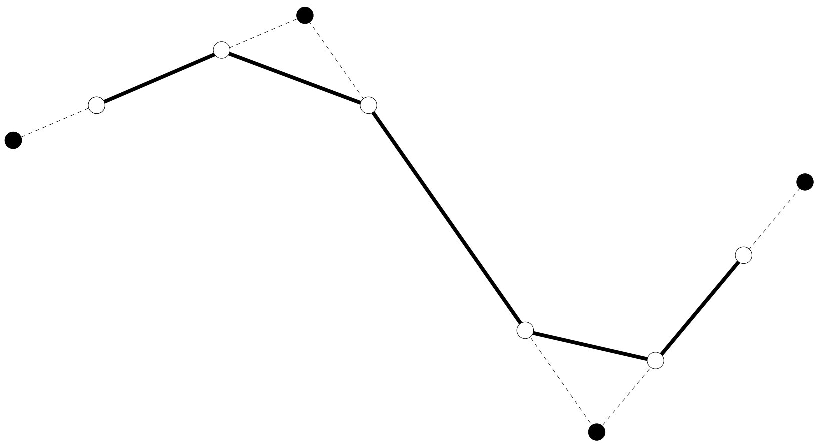
- Chaikin's scheme



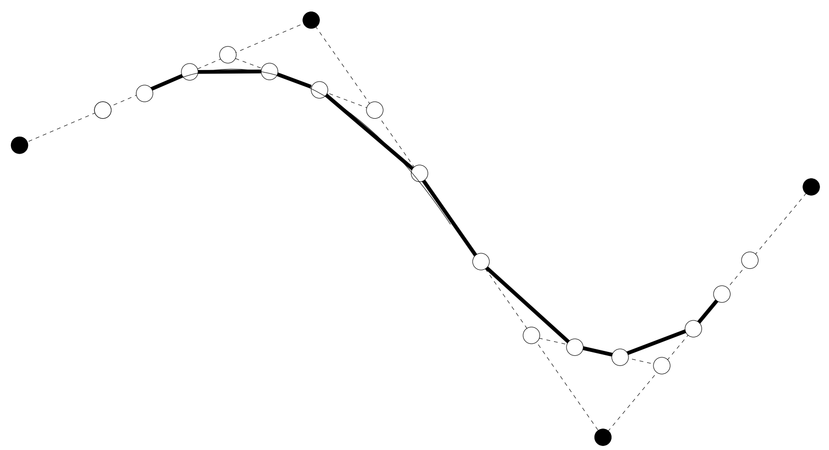
George Chaikin, An algorithm for high speed curve generation, *Computer graphics and Image Processing* **3** (1974) , 346-349

- Chaikin's scheme

  or « Corner-cutting »



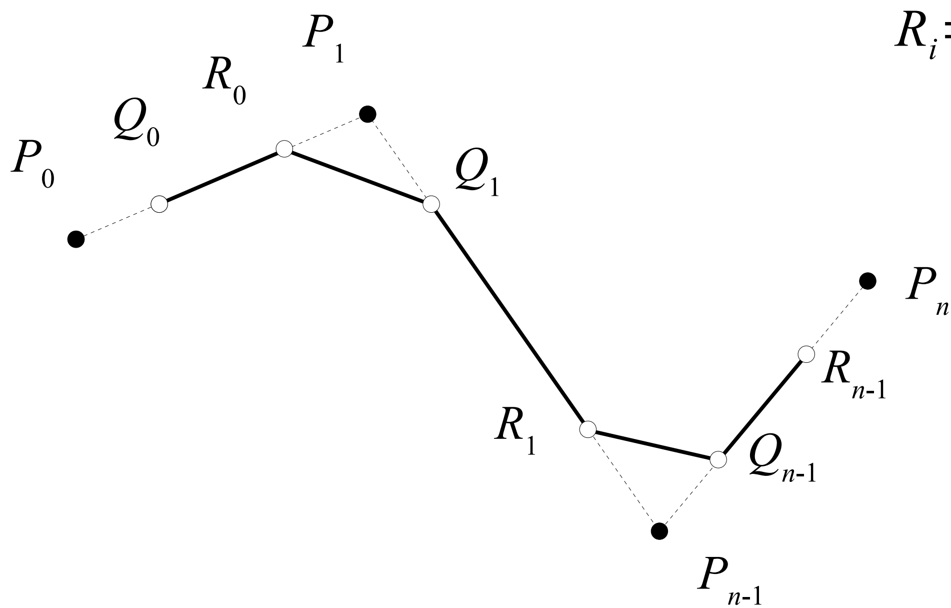
- Chaikin's scheme



Chaikin's idea was simple : repeating the corner cutting, to the limit, one obtains a smooth curve

- Chaikin's scheme
  - Starting from a polygon having n vertices $\{P_0, P_1, \ldots P_{n-1}\}$, one builds the polygon having $2n$ vertices $\{Q_0, R_0, Q_1, R_1, \ldots Q_{n-1}, R_{n-1}\}$. This polygon serves as a basis for the next step of the algorithm : $\{P'_0, P'_1, \ldots P'_{2n-1}\}$
  - The new vertices are :

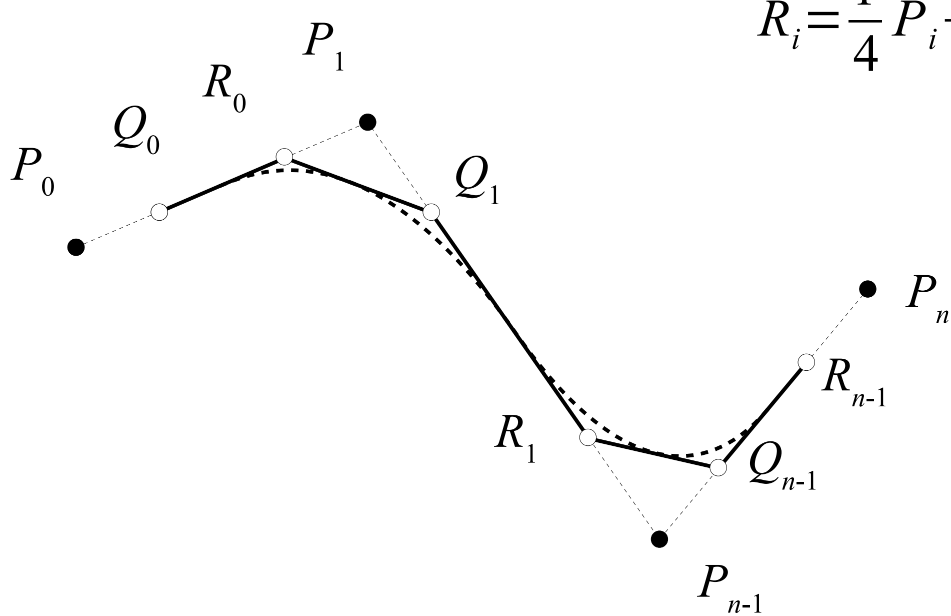
$$Q_i = \frac{3}{4} P_i + \frac{1}{4} P_{i+1}$$
$$R_i = \frac{1}{4} P_i + \frac{3}{4} P_{i+1}$$



- Chaikin's scheme
  - Riesenfeld (1978) has shown that this algorithm leads at the limit to an uniform quadratic B-spline, which exhibits a C^1 continuity.

$$Q_i = \frac{3}{4} P_i + \frac{1}{4} P_{i+1}$$

$$R_i = \frac{1}{4} P_i + \frac{3}{4} P_{i+1}$$



- Demonstration of the equivalence of Chaikin's scheme and uniform quadratic B-Splines
  - The B-Spline curve is defined by : $P(u)=\sum_{i=0}^{n} P_i N_i^2(u)$

$$N_i^2(u)=\frac{u-u_i}{u_{i+2}-u_i} N_i^1(u)+\frac{u_{i+3}-u}{u_{i+3}-u_{i+1}} N_{i+1}^1(u)$$

$$N_i^1(u)=\frac{u-u_i}{u_{i+1}-u_i} N_i^0(u)+\frac{u_{i+2}-u}{u_{i+2}-u_{i+1}} N_{i+1}^0(u)$$

$$N_i^0(u)=\begin{cases}1 & \text{if} \quad u_i \le u < u_{i+1}\\ 0 & \text{otherwise}\end{cases}$$

$$U=\{u_0,\cdots,u_{n+2}\}\ ,\ u_{i+1}-u_i=1\ ,\ i=0\cdots n+1$$

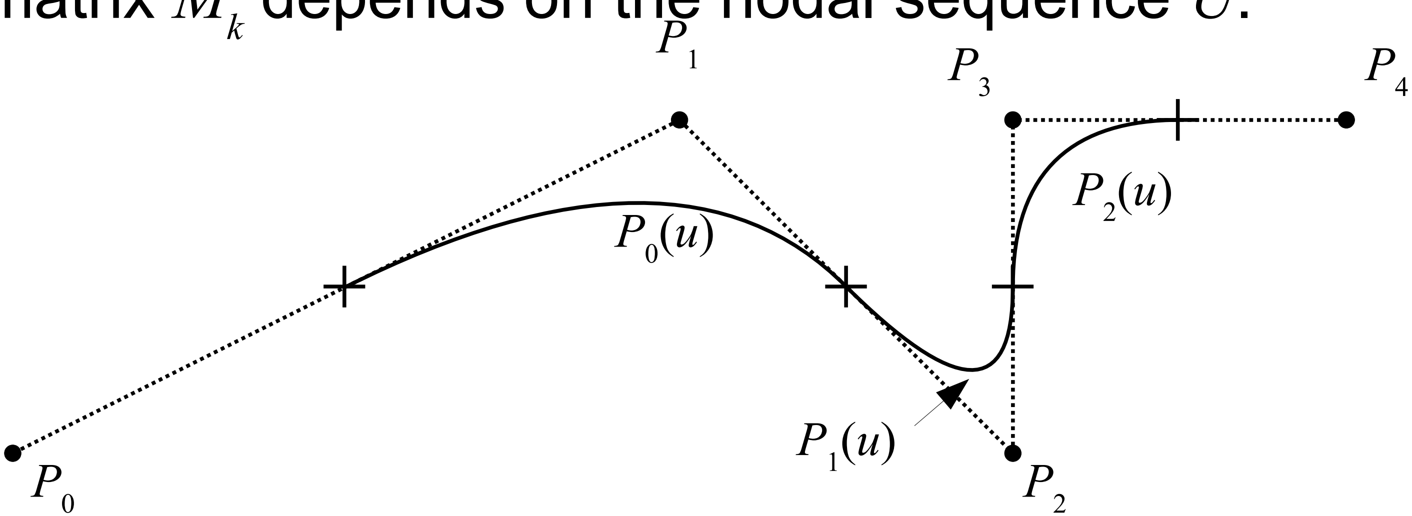
- It can be rewritten as an assembly of curved “parts” :

$$P(u)=\sum_{i=0}^{n} P_i N_i^2(u)=\sum_{k=0}^{n-2} P_k(u)$$

« Portion » of curve

with $P_k(u)=[1 \;\; u \;\; u^2]\cdot M_k \cdot \begin{bmatrix} P_k \\ P_{k+1} \\ P_{k+2} \end{bmatrix}$

- The matrix M_k depends on the nodal sequence U.



- Computation of shape functions of degree $d \leq 2$ for $u_2 = 0 \leq u \leq u_3 = 1$

$$U = \{u_0 = -2, u_1 = -1, u_2 = 0, u_3 = 1, u_4 = 2, u_5 = 3\}$$

$$N_0^0 = 0,\quad N_1^0 = 0,\quad N_2^0 = 1,\quad N_3^0 = 0,\quad N_4^0 = 0$$

$$N_0^1 = 0,\quad N_1^1 = 1-u,\quad N_2^1 = u,\quad N_3^1 = 0$$

$$N_0^2 = \frac{1}{2}(1-2u+u^2)$$

$$N_1^2 = \frac{1}{2}(1+2u-2u^2)$$

$$N_2^2 = \frac{1}{2}(u^2)$$

Therefore, $M_0 = \dfrac{1}{2}\begin{bmatrix} 1 & 1 & 0 \\ -2 & 2 & 0 \\ 1 & -2 & 1 \end{bmatrix}$

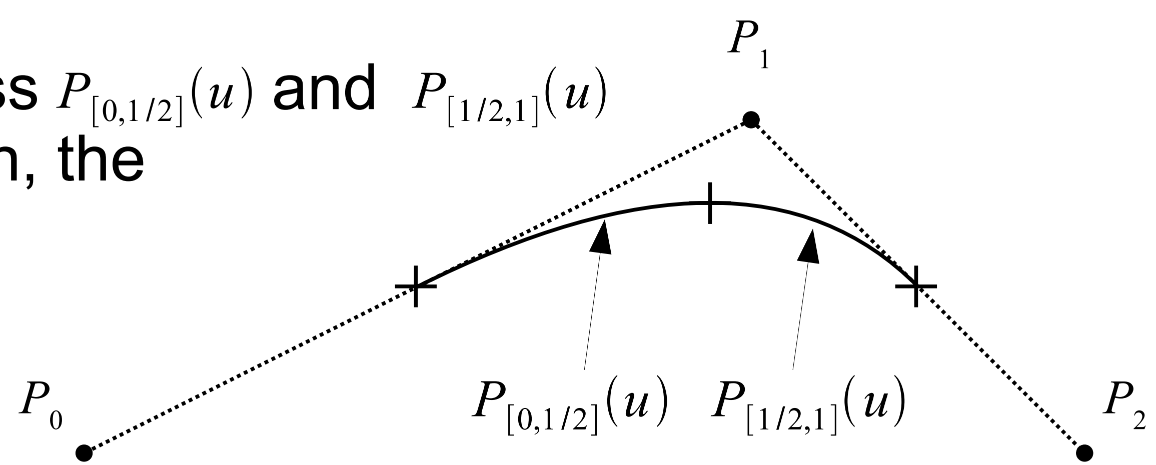
General case :
$$M_k = \frac{1}{u_{k+3}-u_{k+2}} \begin{bmatrix} \dfrac{u_{k+3}^2}{\alpha} & -\dfrac{u_{k+3}u_{k+1}}{\alpha} - \dfrac{u_{k+4}u_{k+2}}{\beta} & \dfrac{u_{k+2}^2}{\beta} \\ -2\dfrac{u_{k+3}}{\alpha} & \dfrac{u_{k+3}+u_{k+1}}{\alpha} + \dfrac{u_{k+4}+u_{k+2}}{\beta} & -2\dfrac{u_{k+2}}{\beta} \\ \dfrac{1}{\alpha} & -\dfrac{1}{\alpha} - \dfrac{1}{\beta} & \dfrac{1}{\beta} \end{bmatrix}$$

with :
$$\alpha = u_{k+3} - u_{k+1}$$
$$\beta = u_{k+4} - u_{k+2}$$

- Binary subdivision of a B-Spline curve for $0 \leq u \leq 1$
  - One has to find the new set of control points for each halves of the curve
  - We set n=2 (number of control points -1)

$$P(u) = [1 \quad u \quad u^2] \cdot M \cdot \begin{bmatrix} P_0 \\ P_1 \\ P_2 \end{bmatrix} \qquad M = \frac{1}{2}\begin{bmatrix} 1 & 1 & 0 \\ -2 & 2 & 0 \\ 1 & -2 & 1 \end{bmatrix}$$

  - One wants to express $P_{[0,1/2]}(u)$ and $P_{[1/2,1]}(u)$ - on each subdivision, the parameter u shall be between 0 and 1.



- Case of $P_{[0,1/2]}(u)$

$$P_{[0,1/2]}(u) = P(u/2) = [1 \;\; u/2 \;\; u^2/4] \cdot M \cdot \begin{bmatrix} P_0 \\ P_1 \\ P_2 \end{bmatrix}$$

$$= [1 \;\; u \;\; u^2] \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1/4 \end{bmatrix} \cdot M \cdot \begin{bmatrix} P_0 \\ P_1 \\ P_2 \end{bmatrix}$$

$$= [1 \;\; u \;\; u^2] \cdot M \cdot M^{-1} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1/4 \end{bmatrix} \cdot M \cdot \begin{bmatrix} P_0 \\ P_1 \\ P_2 \end{bmatrix}$$

$$= [1 \;\; u \;\; u^2] \cdot M \cdot \begin{bmatrix} Q_0 \\ Q_1 \\ Q_2 \end{bmatrix} \quad \text{avec} \quad \begin{bmatrix} Q_0 \\ Q_1 \\ Q_2 \end{bmatrix} = \underbrace{M^{-1} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1/4 \end{bmatrix} \cdot M}_{S_{[0,1/2]}} \cdot \begin{bmatrix} P_0 \\ P_1 \\ P_2 \end{bmatrix}$$

- Case of $P_{[1/2,1]}(u)$

$$P_{[1/2,1]}(u) = P((1+u)/2) = \begin{bmatrix} 1 & (1+u)/2 & (1+u)^2/4 \end{bmatrix} \cdot M \cdot \begin{bmatrix} P_0 \\ P_1 \\ P_2 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & u & u^2 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1/2 & 1/4 \\ 0 & 1/2 & 1/2 \\ 0 & 0 & 1/4 \end{bmatrix} \cdot M \cdot \begin{bmatrix} P_0 \\ P_1 \\ P_2 \end{bmatrix}$$

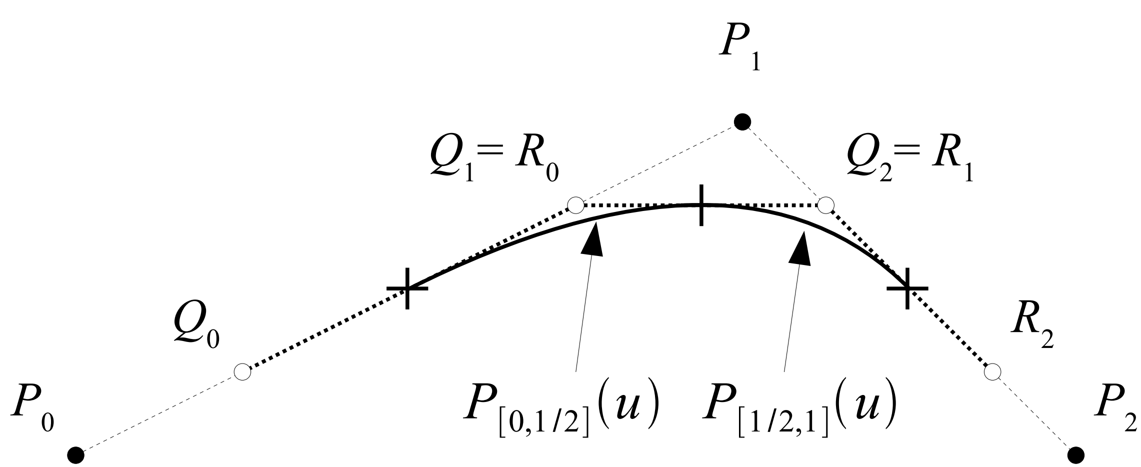
$$= \begin{bmatrix} 1 & u & u^2 \end{bmatrix} \cdot M \cdot M^{-1} \cdot \begin{bmatrix} 1 & 1/2 & 1/4 \\ 0 & 1/2 & 1/2 \\ 0 & 0 & 1/4 \end{bmatrix} \cdot M \cdot \begin{bmatrix} P_0 \\ P_1 \\ P_2 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & u & u^2 \end{bmatrix} \cdot M \cdot \begin{bmatrix} R_0 \\ R_1 \\ R_2 \end{bmatrix} \quad \text{with} \quad \begin{bmatrix} R_0 \\ R_1 \\ R_2 \end{bmatrix} = \underbrace{M^{-1} \cdot \begin{bmatrix} 1 & 1/2 & 1/4 \\ 0 & 1/2 & 1/2 \\ 0 & 0 & 1/4 \end{bmatrix} \cdot M}_{S_{[1/2,1]}} \cdot \begin{bmatrix} P_0 \\ P_1 \\ P_2 \end{bmatrix}$$

- Finally,

$$\begin{bmatrix} Q_0 \\ Q_1 \\ Q_2 \end{bmatrix} = S_{[0,1/2]} \cdot \begin{bmatrix} P_0 \\ P_1 \\ P_2 \end{bmatrix} \qquad S_{[0,1/2]} = M^{-1} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1/4 \end{bmatrix} \cdot M = \frac{1}{4} \begin{bmatrix} 3 & 1 & 0 \\ 1 & 3 & 0 \\ 0 & 3 & 1 \end{bmatrix} \qquad \begin{bmatrix} Q_0 \\ Q_1 \\ Q_2 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 3P_0 + P_1 \\ P_0 + 3P_1 \\ 3P_1 + P_2 \end{bmatrix}$$

$$\begin{bmatrix} R_0 \\ R_1 \\ R_2 \end{bmatrix} = S_{[1/2,1]} \cdot \begin{bmatrix} P_0 \\ P_1 \\ P_2 \end{bmatrix} \qquad S_{[1/2,1]} = M^{-1} \cdot \begin{bmatrix} 1 & 1/2 & 1/4 \\ 0 & 1/2 & 1/2 \\ 0 & 0 & 1/4 \end{bmatrix} \cdot M = \frac{1}{4} \begin{bmatrix} 1 & 3 & 0 \\ 0 & 3 & 1 \\ 0 & 1 & 3 \end{bmatrix} \qquad \begin{bmatrix} R_0 \\ R_1 \\ R_2 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} P_0 + 3P_1 \\ 3P_1 + P_2 \\ P_1 + 3P_2 \end{bmatrix}$$



One finds the same coefficients as in Chaikin's scheme...
(except for the indices)

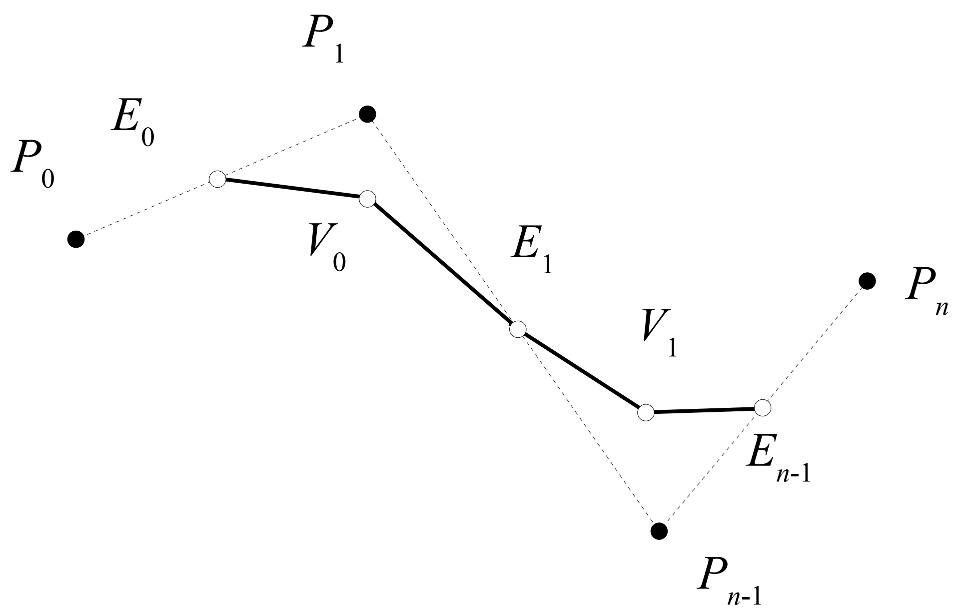
$$Q_i = \frac{3}{4} P_i + \frac{1}{4} P_{i+1}$$

$$R_i = \frac{1}{4} P_i + \frac{3}{4} P_{i+1}$$

- This can be extended to cubic B-Splines
  - C^2 continuity

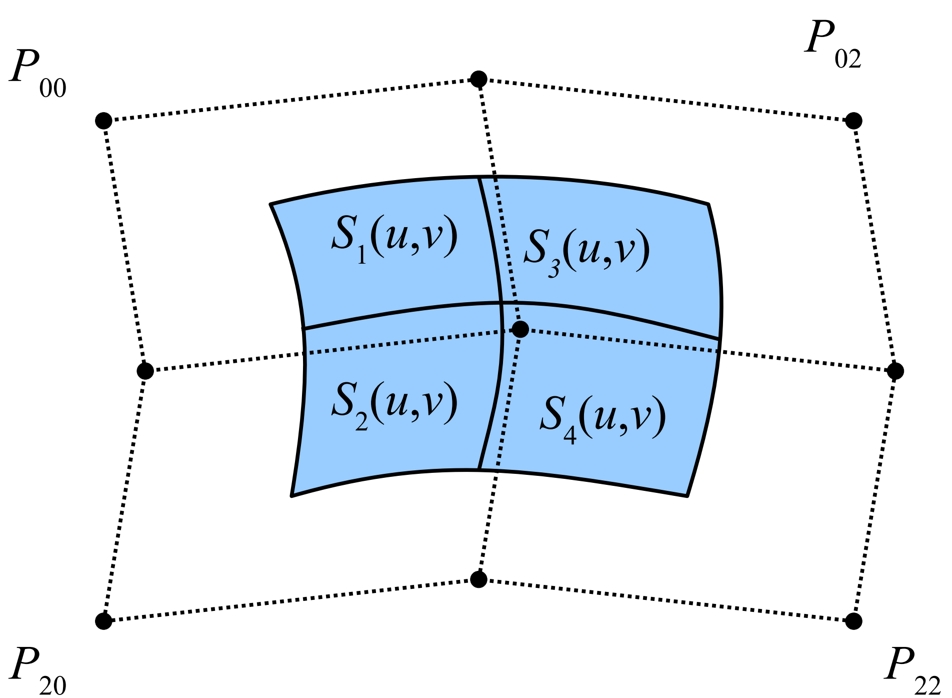
$$E_i = \frac{1}{2} P_i + \frac{1}{2} P_{i+1}$$

$$V_i = \frac{1}{8} P_i + \frac{3}{4} P_{i+1} + \frac{1}{8} P_{i+2}$$



Subdivision surfaces

- Doo-Sabin scheme
  - This is an extension of Chaikin's scheme for a uniform biquadratic B-Spline surface
  - The new mesh is built using the control points resulting from the subdivision of the original patch into 4 new sub-patches.



Subdivision surfaces

- Expression of the bi-quadratic patch as a monomial form for $0 \le u \le 1$ and $0 \le v \le 1$:

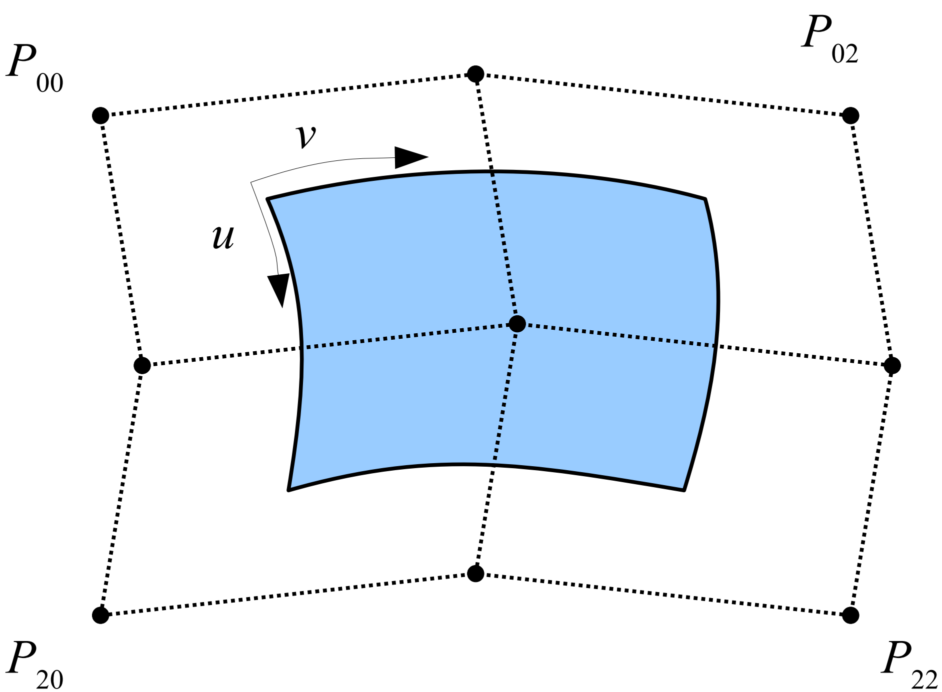
$$S(u,v)=\sum_{i=0}^{2}\sum_{j=0}^{2} N_i^2(u)\,N_j^2(v)\,P_{ij}$$

$$U=V=\{-2,-1,0,1,2,3\}$$

$$N_0^2(t)=\frac{1}{2}(1-2t+t^2)$$

$$N_1^2(t)=\frac{1}{2}(1+2t-2t^2) \qquad \text{(with } t=u \text{ or } v\text{)}$$

$$N_2^2(t)=\frac{1}{2}(t^2)$$



$$S(u,v)=\begin{bmatrix}1 & u & u^2\end{bmatrix}\cdot M\cdot\begin{bmatrix}P_0(v)\\ P_1(v)\\ P_2(v)\end{bmatrix}$$

$$S(u,v)=\begin{bmatrix}1 & u & u^2\end{bmatrix}\cdot M\cdot\begin{bmatrix}P_{00} & P_{01} & P_{02}\\ P_{10} & P_{11} & P_{12}\\ P_{20} & P_{21} & P_{22}\end{bmatrix}\cdot M^{\mathrm{T}}\cdot\begin{bmatrix}1\\ v\\ v^2\end{bmatrix}$$

Again, $M=\frac{1}{2}\begin{bmatrix}1 & 1 & 0\\ -2 & 2 & 0\\ 1 & -2 & 1\end{bmatrix}$

Subdivision surfaces

- Subdivision - patch $S_1(u,v)$

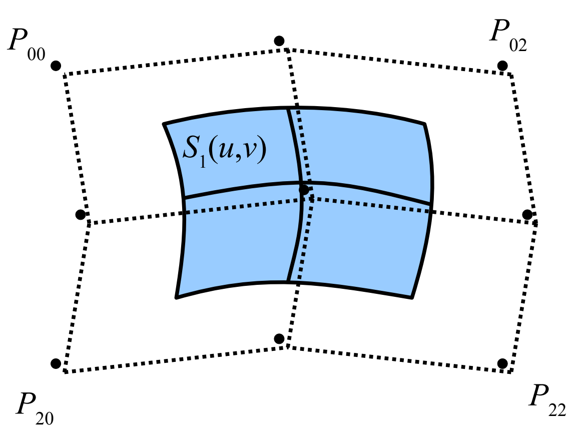
$$S_1(u,v)=S(u/2,v/2)=[1 \;\; u/2 \;\; u^2/4]\cdot M\cdot P\cdot M^{\mathrm{T}}\cdot\begin{bmatrix}1\\ v/2\\ v^2/4\end{bmatrix} \qquad P=\begin{bmatrix}P_{00} & P_{01} & P_{02}\\ P_{10} & P_{11} & P_{12}\\ P_{20} & P_{21} & P_{22}\end{bmatrix}$$

$$S_1(u,v)=S(u/2,v/2)=[1 \;\; u \;\; u^2]\cdot C\cdot M\cdot P\cdot M^{\mathrm{T}}\cdot C^{\mathrm{T}}\begin{bmatrix}1\\ v\\ v^2\end{bmatrix} \qquad C=\begin{bmatrix}1 & 0 & 0\\ 0 & 1/2 & 0\\ 0 & 0 & 1/4\end{bmatrix}$$

$$=[1 \;\; u \;\; u^2]\cdot M\cdot M^{-1}\cdot C\cdot M\cdot P\cdot M^{\mathrm{T}}\cdot C^{\mathrm{T}}\cdot(M^{-1})^{\mathrm{T}}\cdot M^{\mathrm{T}}\cdot\begin{bmatrix}1\\ v\\ v^2\end{bmatrix}$$

$$=[1 \;\; u \;\; u^2]\cdot M\cdot(M^{-1}\cdot C\cdot M)\cdot P\cdot(M^{-1}\cdot C\cdot M)^{\mathrm{T}}\cdot M^{\mathrm{T}}\cdot\begin{bmatrix}1\\ v\\ v^2\end{bmatrix}$$

$$=[1 \;\; u \;\; u^2]\cdot M\cdot P'\cdot M^{\mathrm{T}}\cdot\begin{bmatrix}1\\ v\\ v^2\end{bmatrix} \qquad P'=S\cdot P\cdot S^{\mathrm{T}} \qquad S=M^{-1}\cdot C\cdot M$$

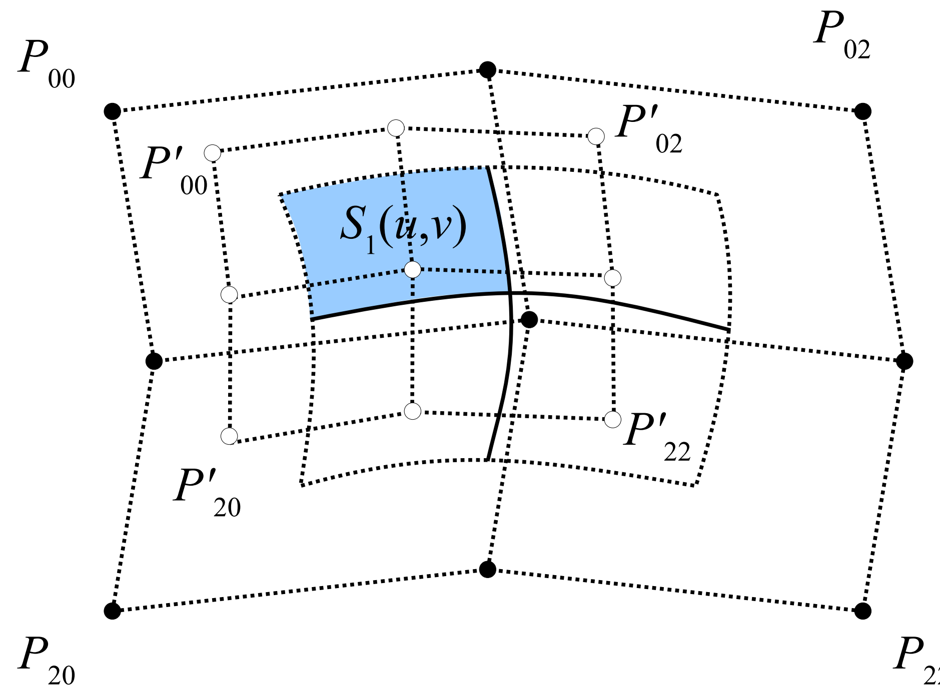


Subdivision surfaces

- Finally,

$$P' = S \cdot P \cdot S^{\mathrm{T}} \qquad S = M^{-1} \cdot C \cdot M = \frac{1}{4} \begin{bmatrix} 3 & 1 & 0 \\ 1 & 3 & 0 \\ 0 & 3 & 1 \end{bmatrix}$$

$$P' = \frac{1}{16} \begin{bmatrix} 3 & 1 & 0 \\ 1 & 3 & 0 \\ 0 & 3 & 1 \end{bmatrix} \cdot \begin{bmatrix} P_{00} & P_{01} & P_{02} \\ P_{10} & P_{11} & P_{12} \\ P_{20} & P_{21} & P_{22} \end{bmatrix} \cdot \begin{bmatrix} 3 & 1 & 0 \\ 1 & 3 & 3 \\ 0 & 0 & 1 \end{bmatrix}$$



$$P' = \frac{1}{16} \begin{bmatrix} 3(3P_{00}+P_{10})+3P_{01}+P_{11} & 3P_{00}+P_{10}+3(3P_{01}+P_{11}) & 3(3P_{01}+P_{11})+3P_{02}+P_{12} \\ 3(P_{00}+3P_{10})+P_{01}+3P_{11} & P_{00}+3P_{10}+3(P_{01}+3P_{11}) & 3(P_{01}+3P_{11})+P_{02}+3P_{12} \\ 3(3P_{10}+P_{20})+3P_{11}+P_{21} & 3P_{10}+P_{20}+3(3P_{11}+P_{21}) & 3(3P_{11}+P_{21})+3P_{12}+P_{22} \end{bmatrix}$$

- Same developments should be done with the 3 other quadrants, and will lead to the same “structure”

- Subdivision - patch $S_2(u,v)$

$$S_2(u,v)=S(u/2,(1+v)/2)=\begin{bmatrix}1 & u/2 & u^2/4\end{bmatrix}\cdot M\cdot P\cdot M^{\mathrm{T}}\cdot\begin{bmatrix}1\\(1+v)/2\\(1+v)^2/4\end{bmatrix}\qquad P=\begin{bmatrix}P_{00} & P_{01} & P_{02}\\P_{10} & P_{11} & P_{12}\\P_{20} & P_{21} & P_{22}\end{bmatrix}$$

$$S_2(u,v)=S(u/2,(1+v)/2)=\begin{bmatrix}1 & u & u^2\end{bmatrix}\cdot C_u\cdot M\cdot P\cdot M^{\mathrm{T}}\cdot C_v^{\mathrm{T}}\begin{bmatrix}1\\v\\v^2\end{bmatrix}$$

$$C_u=\begin{bmatrix}1 & 0 & 0\\0 & 1/2 & 0\\0 & 0 & 1/4\end{bmatrix}\qquad C_v=\begin{bmatrix}1 & 1/2 & 1/4\\0 & 1/2 & 1/2\\0 & 0 & 1/4\end{bmatrix}$$

$$=\begin{bmatrix}1 & u & u^2\end{bmatrix}\cdot M\cdot(M^{-1}\cdot C_u\cdot M)\cdot P\cdot(M^{-1}\cdot C_v\cdot M)^{\mathrm{T}}\cdot M^{\mathrm{T}}\cdot\begin{bmatrix}1\\v\\v^2\end{bmatrix}$$

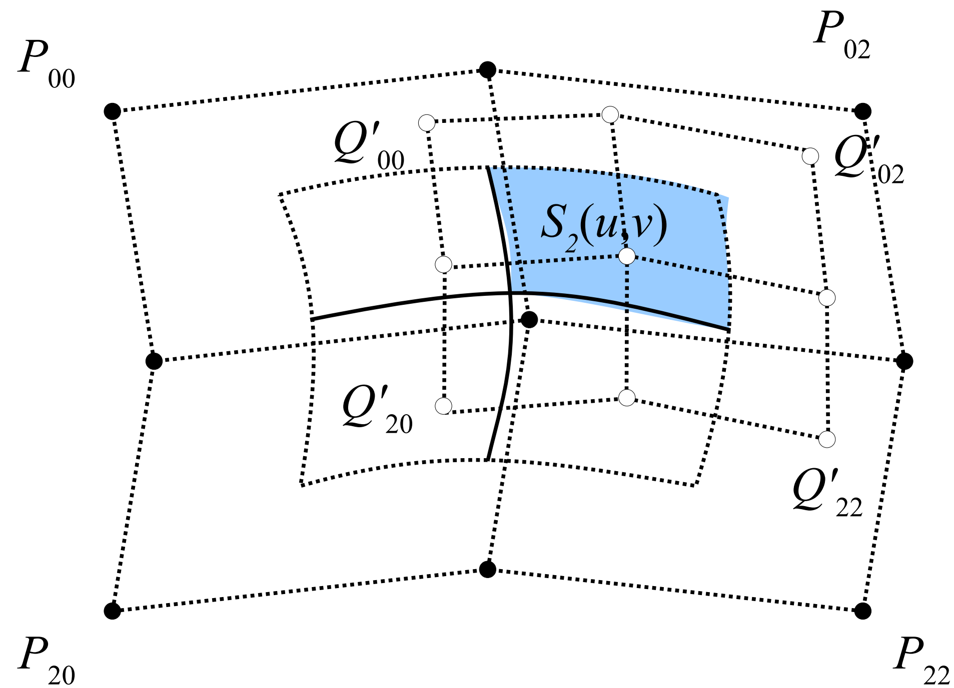
$$=\begin{bmatrix}1 & u & u^2\end{bmatrix}\cdot M\cdot Q'\cdot M^{\mathrm{T}}\cdot\begin{bmatrix}1\\v\\v^2\end{bmatrix}\qquad Q'=S_u\cdot P\cdot S_v^{\mathrm{T}}\qquad \begin{matrix}S_u=M^{-1}\cdot C_u\cdot M\\S_v=M^{-1}\cdot C_v\cdot M\end{matrix}$$

Subdivision surfaces

$$Q' = S_u \cdot P\, S_v^{\mathrm{T}} \qquad S_u = M^{-1} \cdot C_u \cdot M = \frac{1}{4}\begin{bmatrix} 3 & 1 & 0 \\ 1 & 3 & 0 \\ 0 & 3 & 1 \end{bmatrix}$$

$$S_v = M^{-1} \cdot C_v \cdot M = \frac{1}{4}\begin{bmatrix} 1 & 3 & 0 \\ 0 & 3 & 1 \\ 0 & 1 & 3 \end{bmatrix}$$

$$Q' = \frac{1}{16}\begin{bmatrix} 3 & 1 & 0 \\ 1 & 3 & 0 \\ 0 & 3 & 1 \end{bmatrix} \cdot \begin{bmatrix} P_{00} & P_{01} & P_{02} \\ P_{10} & P_{11} & P_{12} \\ P_{20} & P_{21} & P_{22} \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 3 & 3 & 1 \\ 0 & 1 & 3 \end{bmatrix}$$



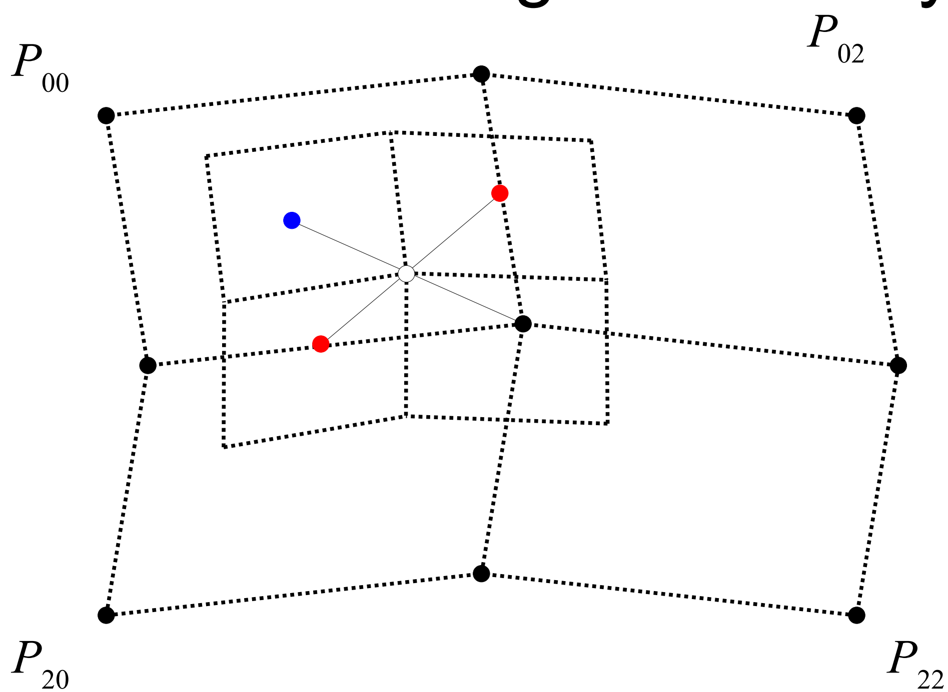
- Some of the points (2) are already computed, e.g.

$$Q'_{00} = 3\left(P_{00} + 3P_{01}\right) + P_{10} + 3P_{11}$$

$$\left(\; = P'_{01} = 3P_{00} + P_{10} + 3\left(3P_{01} + P_{11}\right) \; \right)$$

Subdivision surfaces

- Extension to meshes showing an arbitrary **topology**

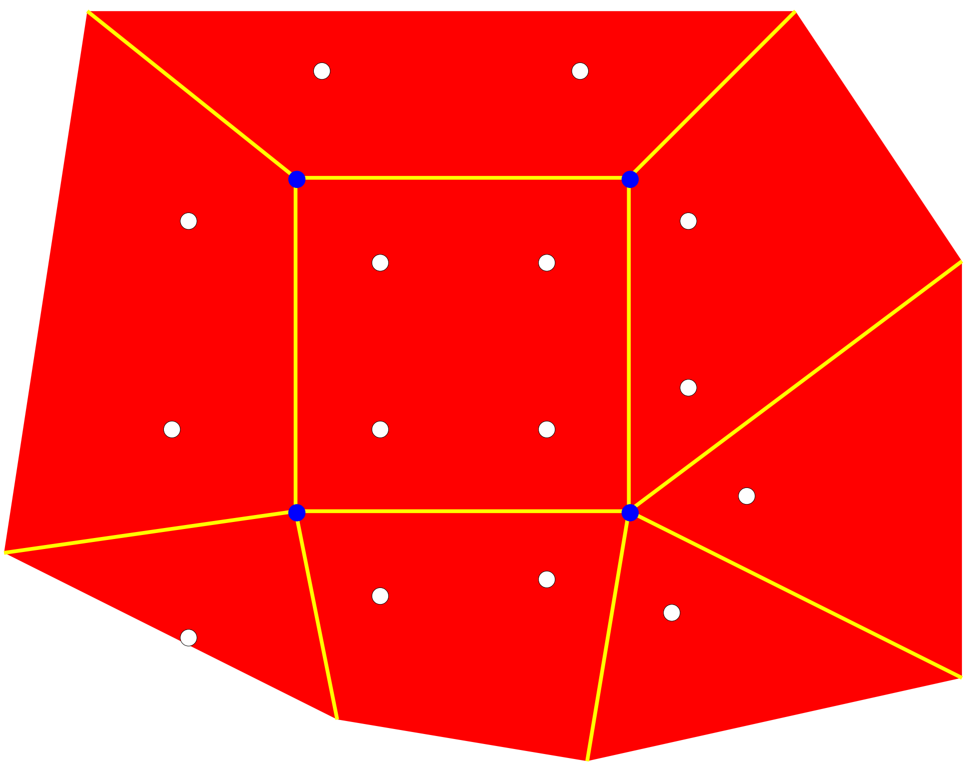


  - The new vertices are obtained as a simple arithmetic mean of 3 categories of vertices :
    - The vertices of the old mesh
    - Vertices on the edges (barycenter of the extremities of the edge)
    - Vertices inside a face (barycenter of the vertices of the face)

Subdivision surfaces

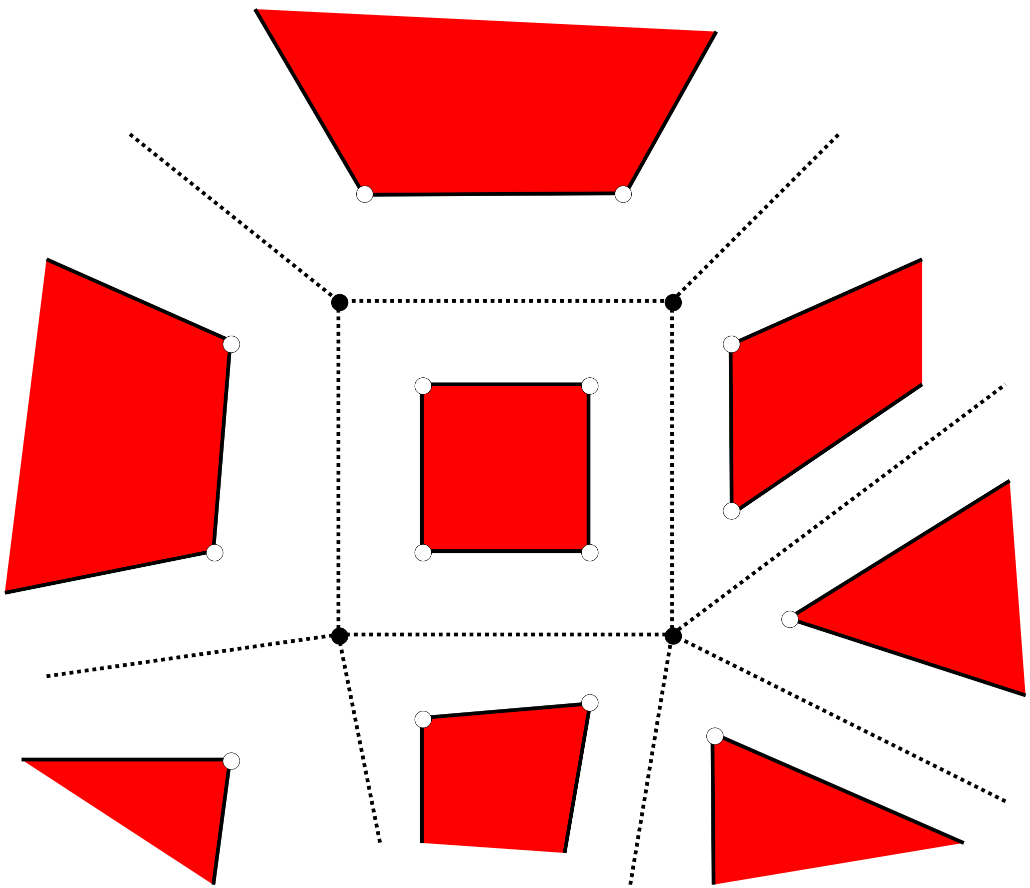
- Extension to meshes showing an arbitrary topology

  1 – Computation of the vertices P' (for each vertex P, compute the mean between P, the vertices on the adjacent faces, and the vertices on adjacent edges)



- Extension to meshes showing an arbitrary topology

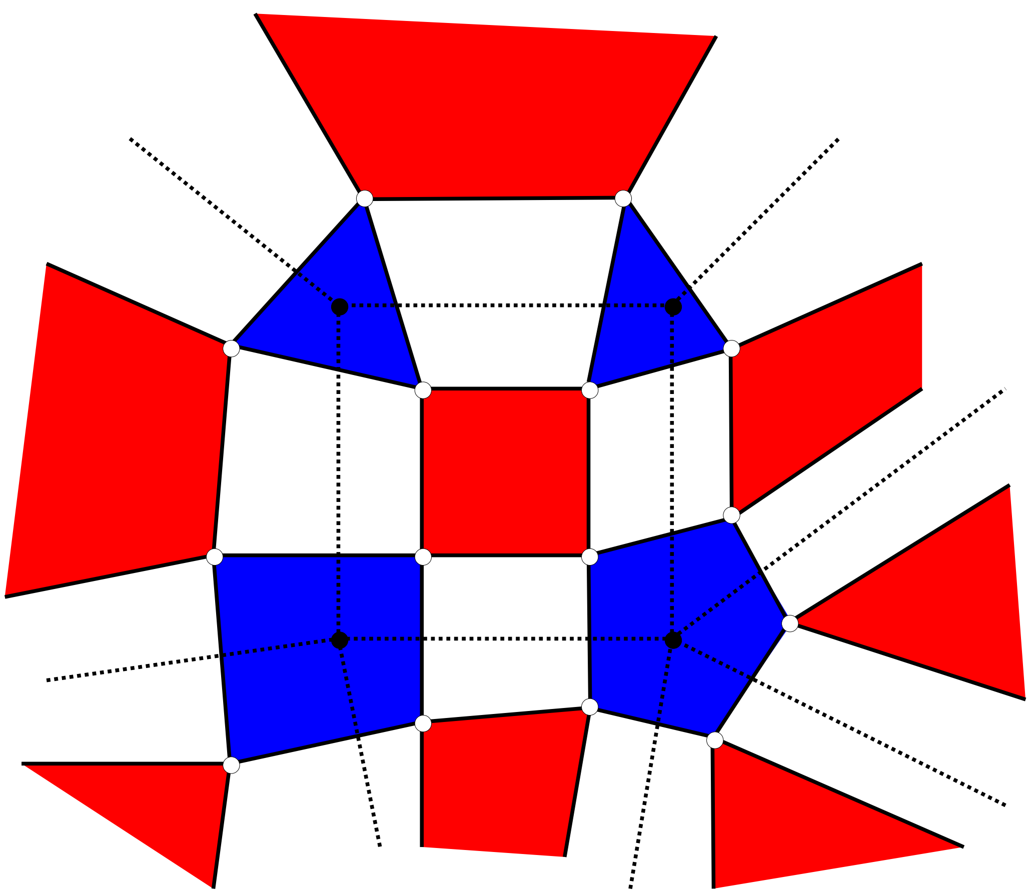
2 – For each face, link the corresponding vertices P'



Subdivision surfaces

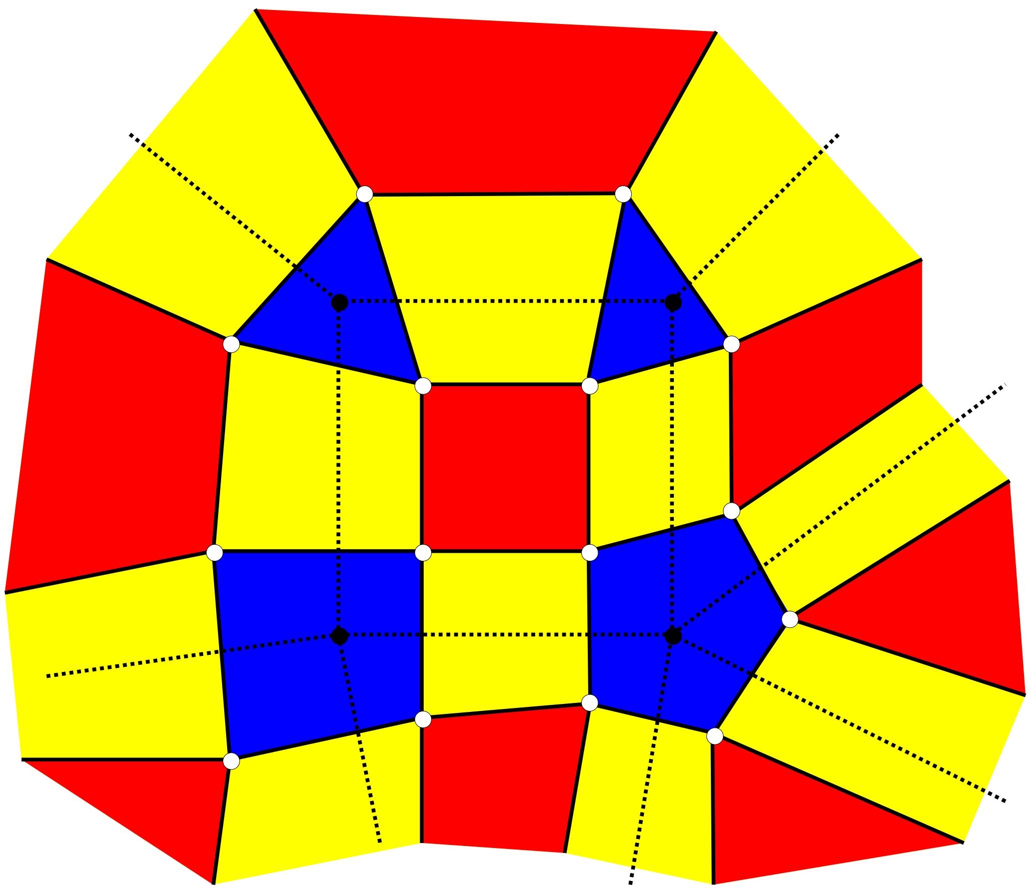
- Extension to meshes showing an arbitrary topology

  3 – For each old vertex, connect the new ones that have been created for each adjacent face to this old vertex.



- Extension to meshes showing an arbitrary topology

  4 – For each old edge, connect the new vertices that have been created for each adjacent face to this old edge.



Subdivision surfaces

- Some points on the mesh and the limiting surface are « extraordinary »
  - These are vertices with a valence (number of incident edges) that is different from 4.

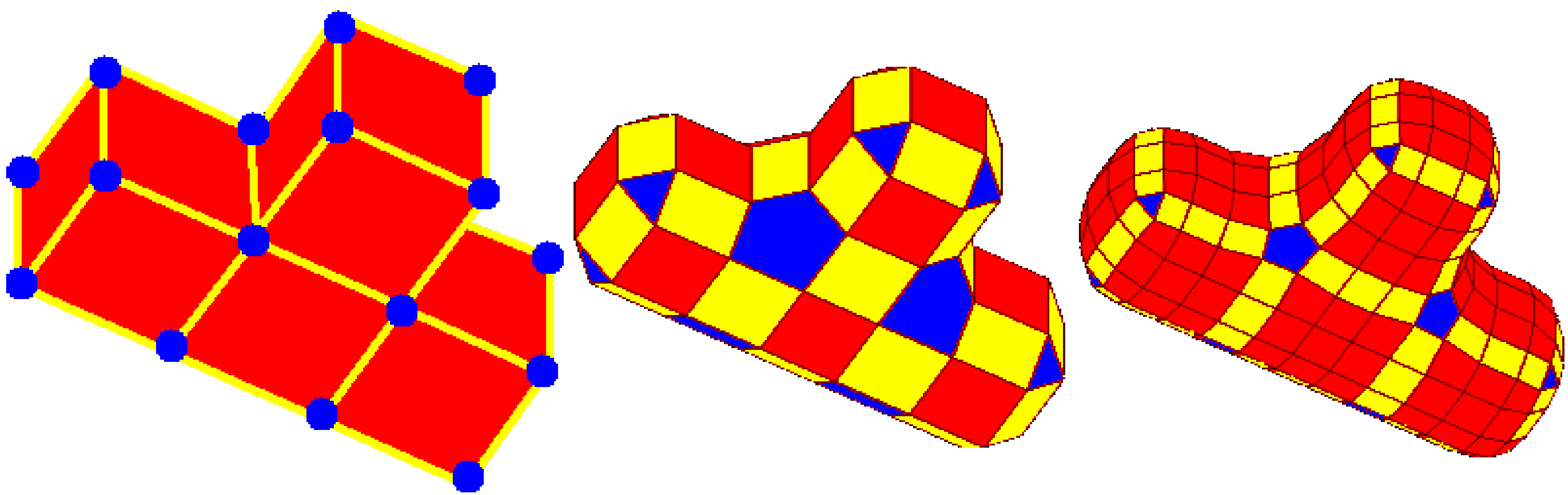
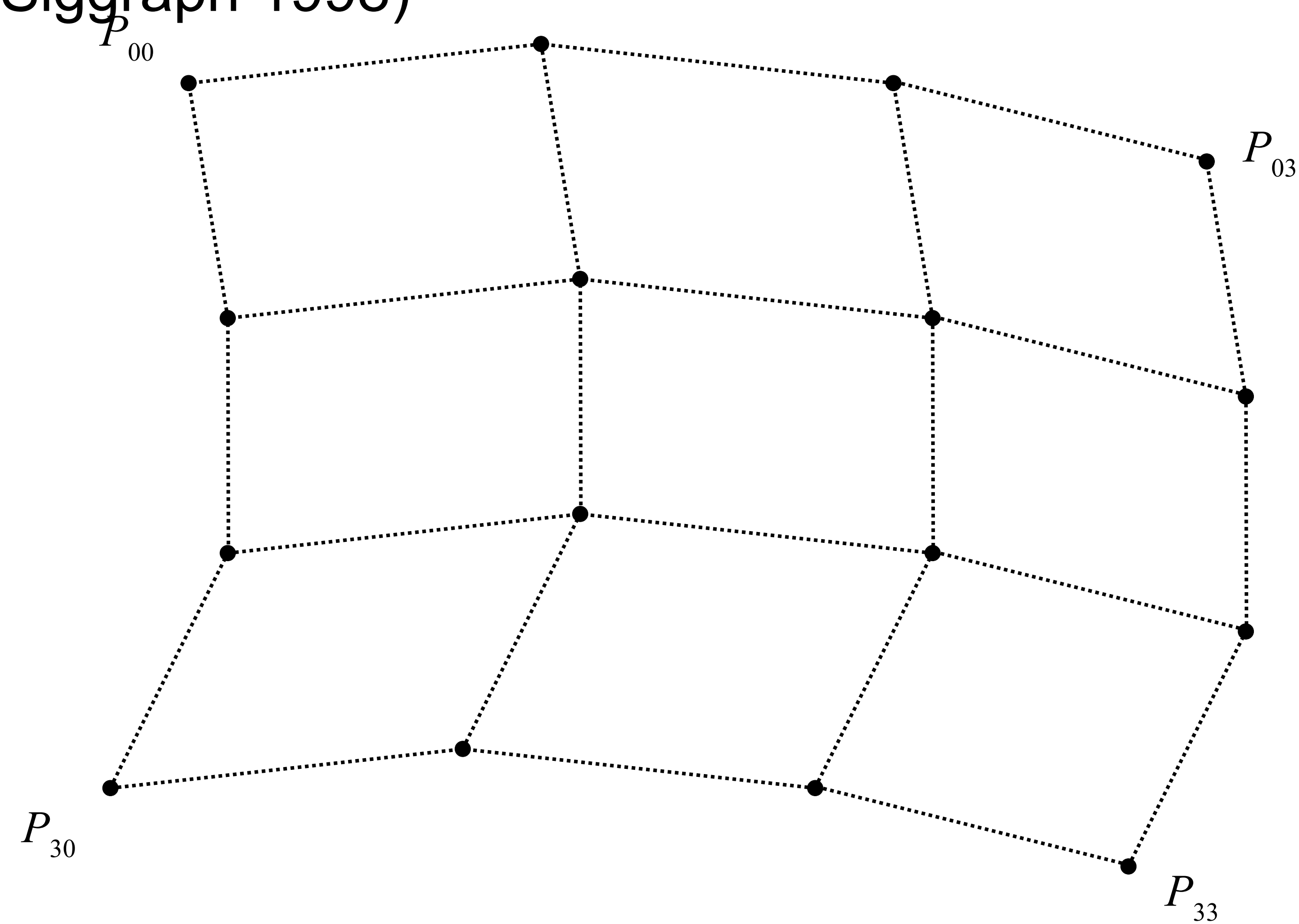


Image : wikipedia

  - Everywhere the continuity of the limiting surface is C^1 ; except at extraordinary points, where it decreases to C^0.

- Catmull-Clark scheme
  - Similar idea for bicubic B-Splines (proof : Jos Stam, Siggraph 1998)



Subdivision surfaces

- Three types of vertices

- « Face » vertices are at the barycentre of the vertices of that face:

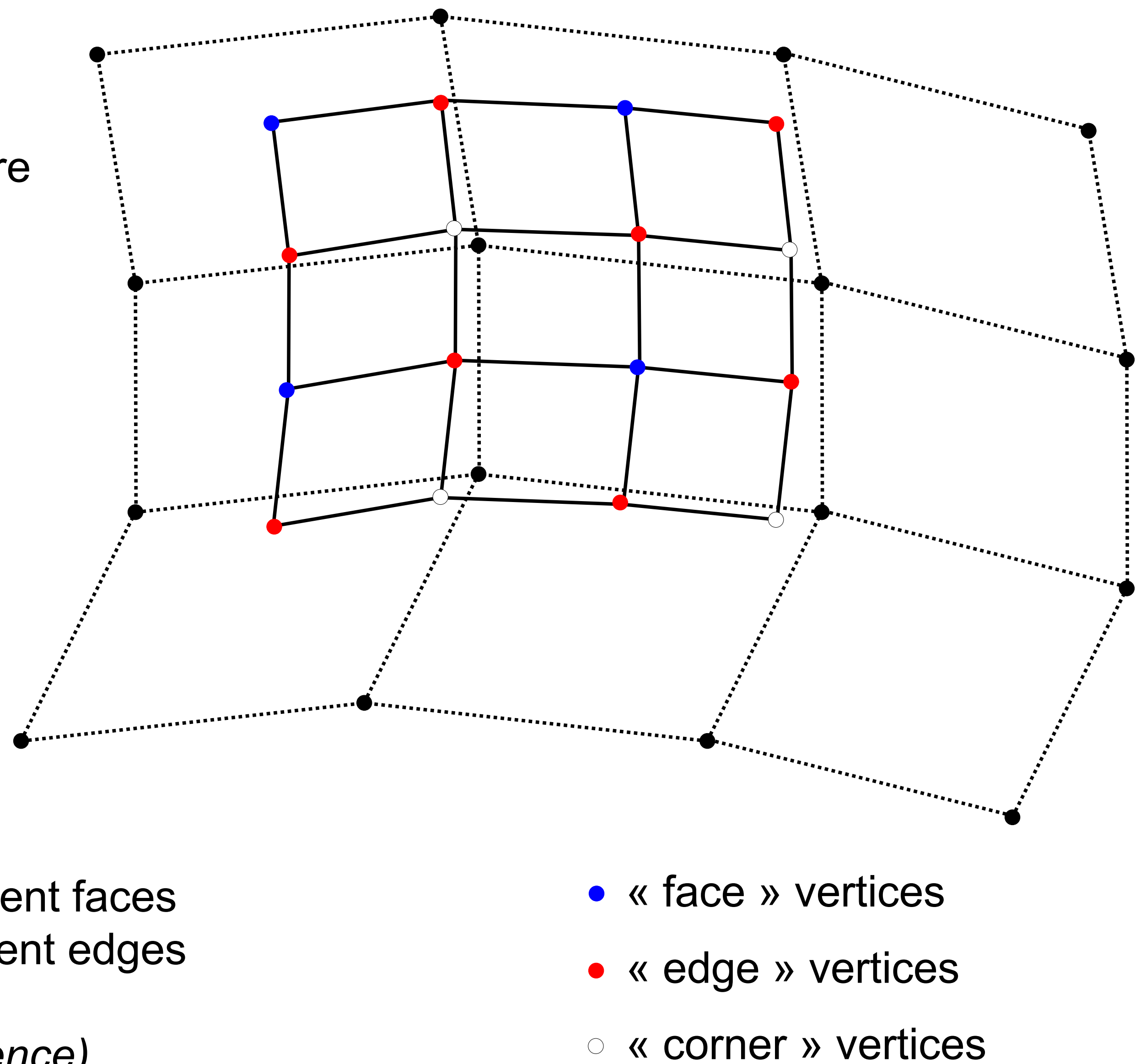
$$P_f = Q$$

- « Edge » vertices are at the barycentre of the extremities of the edge and the two « Face » vertices of the adjacent faces :

$$P_e = \frac{Q+R}{2}$$

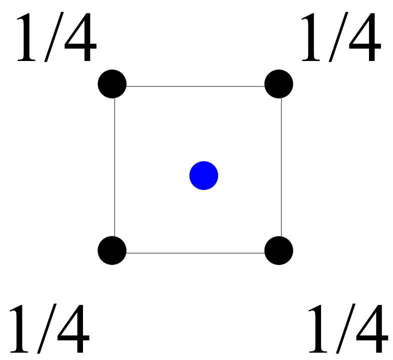
- « Corner » vertices are positioned such that :

$$P_v = \frac{Q+2R+(n-3)S}{n}$$

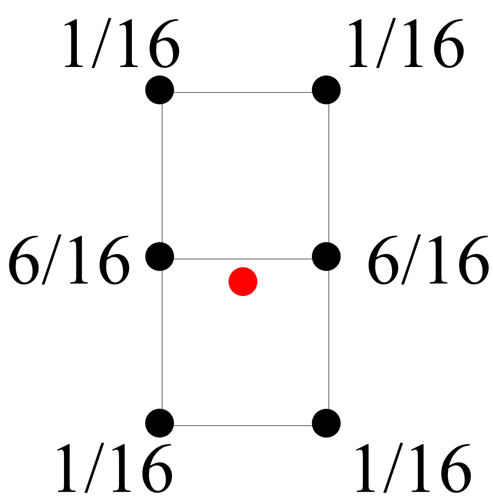


Q = mean of the barycentre of the incident faces
R = mean of the barycentre of the incident edges
S = original vertex
n = number of incident edges to S *(Valence)*

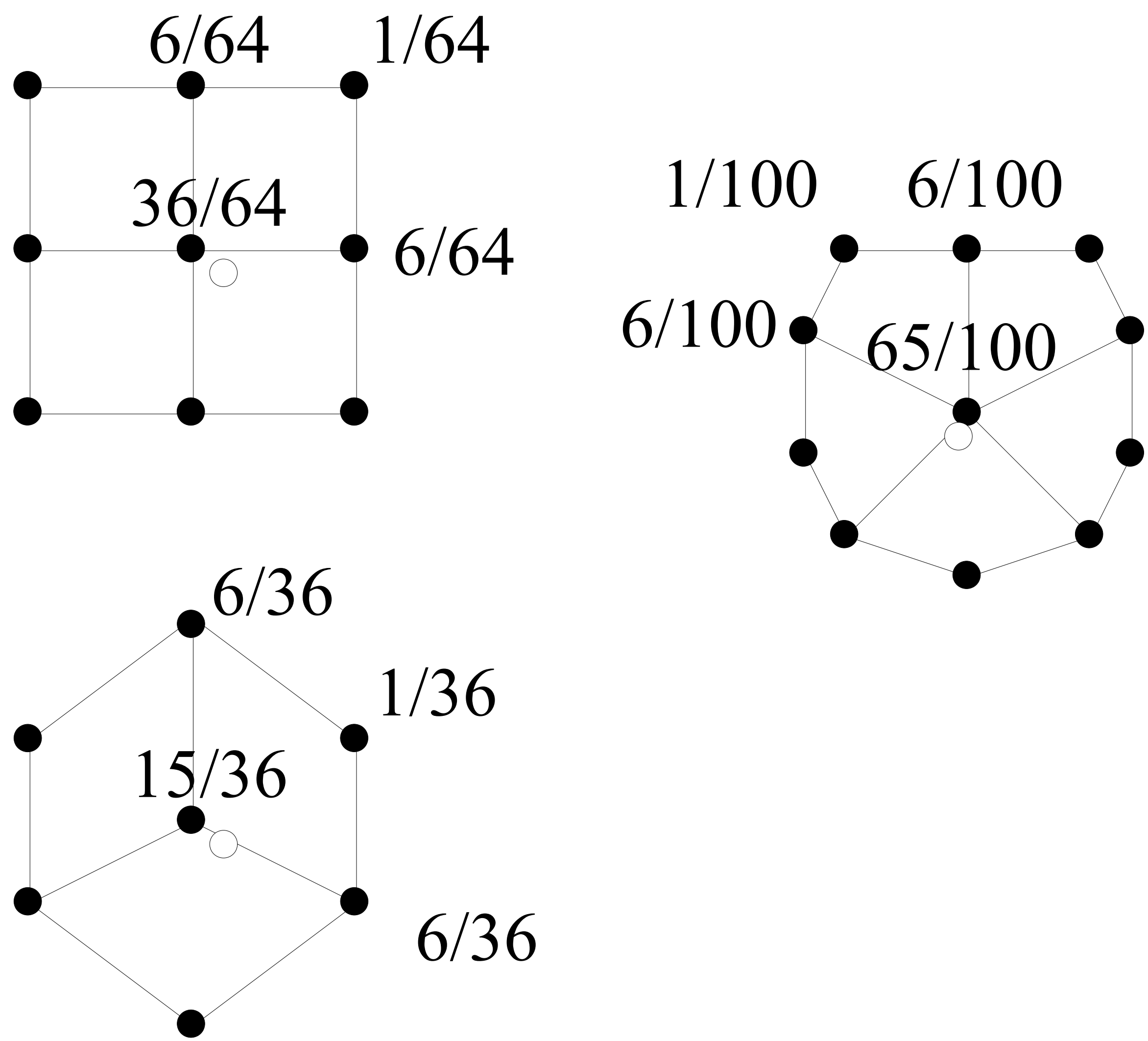
Subdivision surfaces



Face vertex



Edge vertex



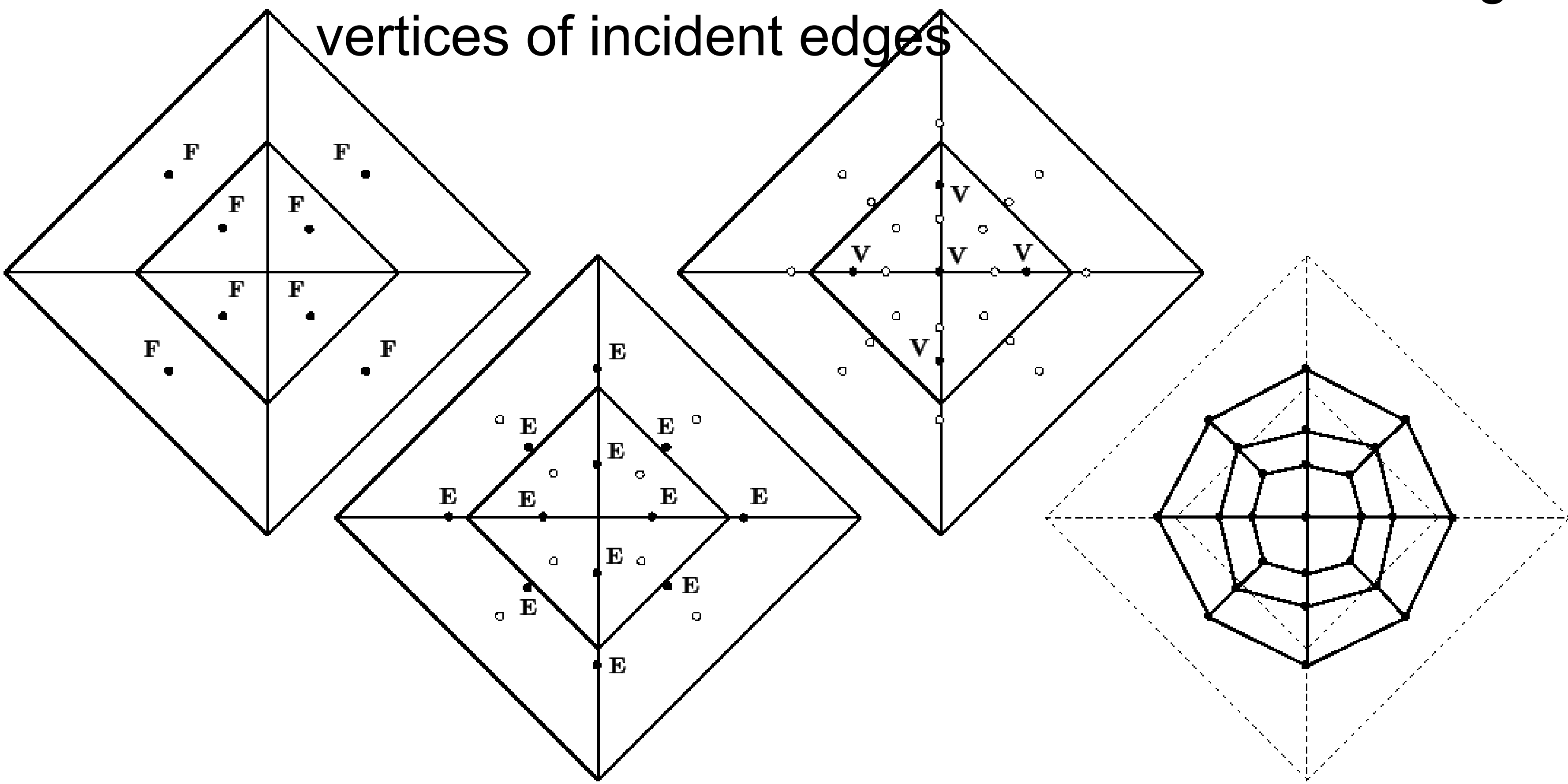
Corner vertex (depends on the valence)

Subdivision surfaces

- Reconnecting the new vertices

  1 – Connect the « face » vertices to the « edge » vertices of neighbouring edges

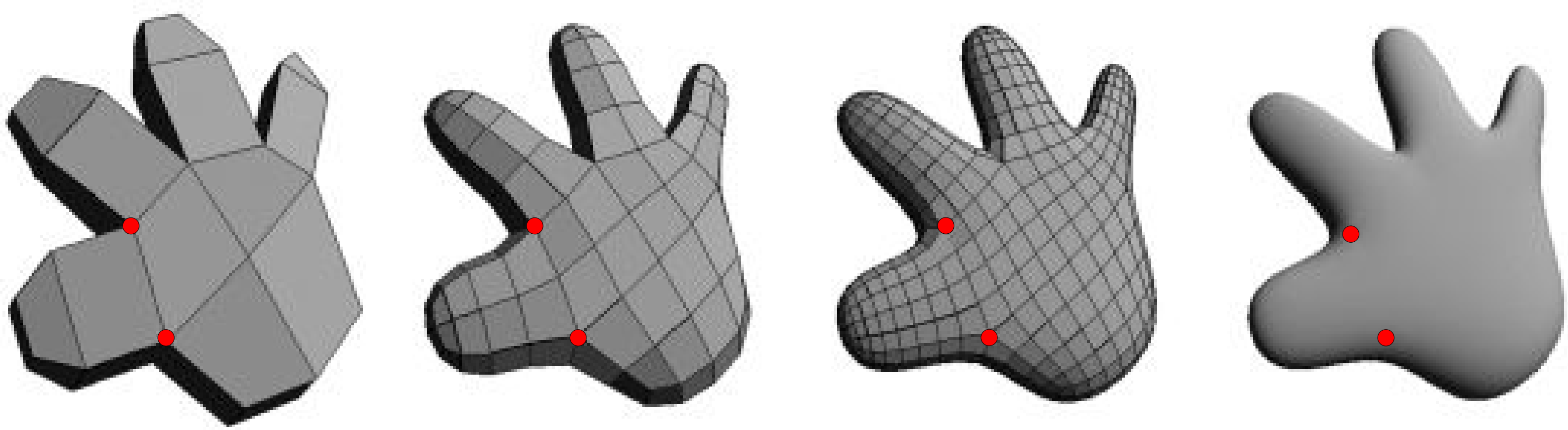
  2 – Connect the « corner » vertices to the « edge » vertices of incident edges



Images : Ken Joy - UCLA

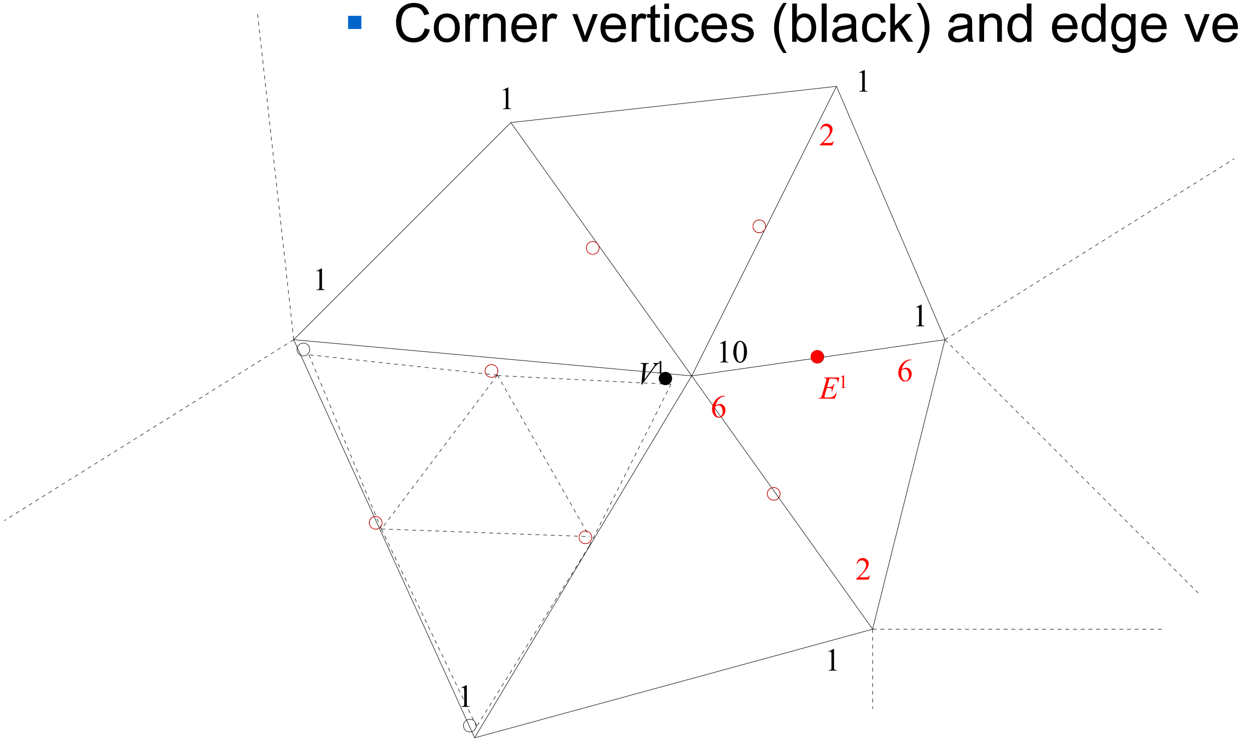
Subdivision surfaces

- As with Doo-Sabin surface, the continuity is degraded for some extraordinary vertices. The bicubic surfaces are therefore C^2 everywhere except at extraordinary points : it is then only C^1.



Images : Imorasa Suzuki

- Loop's scheme
  - Allows to subdivide triangular meshes
  - The limiting surface is C^2 , except at extraordinary vertices of valence <>6 , where it is only C^1.
  - The principle is to subdivide triangles into 4 sub-triangles.
    - Corner vertices (black) and edge vertices (red) are created.



$$V^1 = \frac{10V + Q_1 + Q_2 + Q_3 + Q_4 + Q_5 + Q_6}{16}$$

$$= \frac{5}{8}V + \frac{3}{8}Q$$

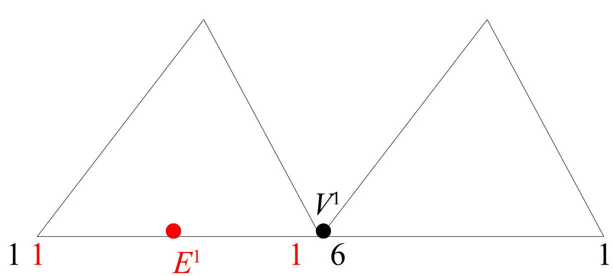
$$E^1 = \frac{6V_1 + 6V_2 + 2F_1 + 2F_2}{16}$$

Subdivision surfaces

- As such, works only for vertices with a valence equal to 6
- It may be extended to other valences, but the formula has to be adapted such that the resulting surface is smooth.
- Let $V^1 = \alpha_n V + (1-\alpha_n) Q$

  with $\alpha_n = \left(\frac{3}{8} + \frac{1}{4}\cos\frac{3\pi}{n}\right)^2 + \frac{3}{8}$

  n is the valence of the original vertex.
- On boundaries : vertices should not move inside the surface, they should rather slide along the boundary. One recovers the classical cubic B-Spline scheme in that case

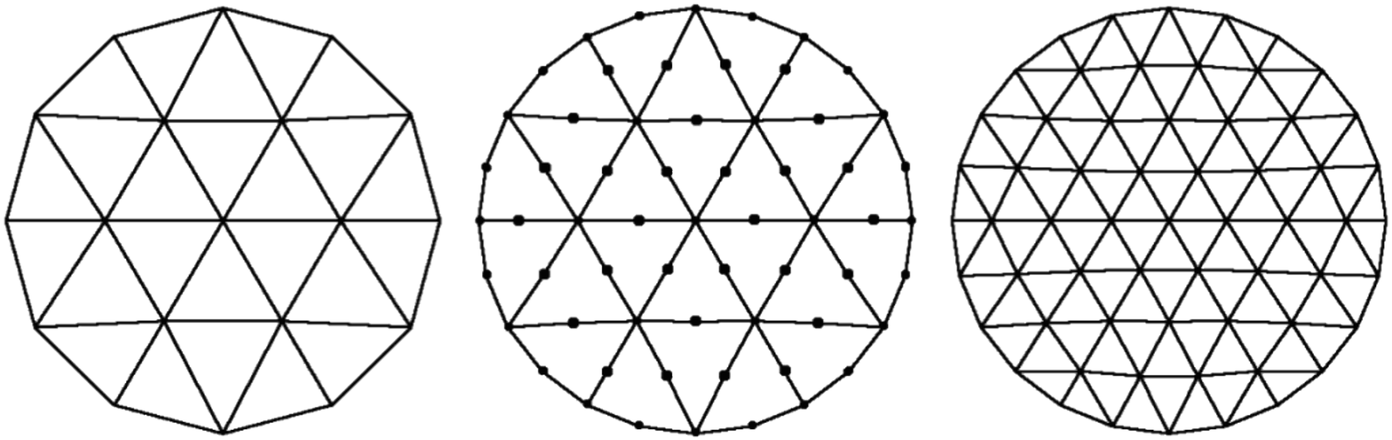


$$E^1 = \frac{V_1 + V_2}{2}$$

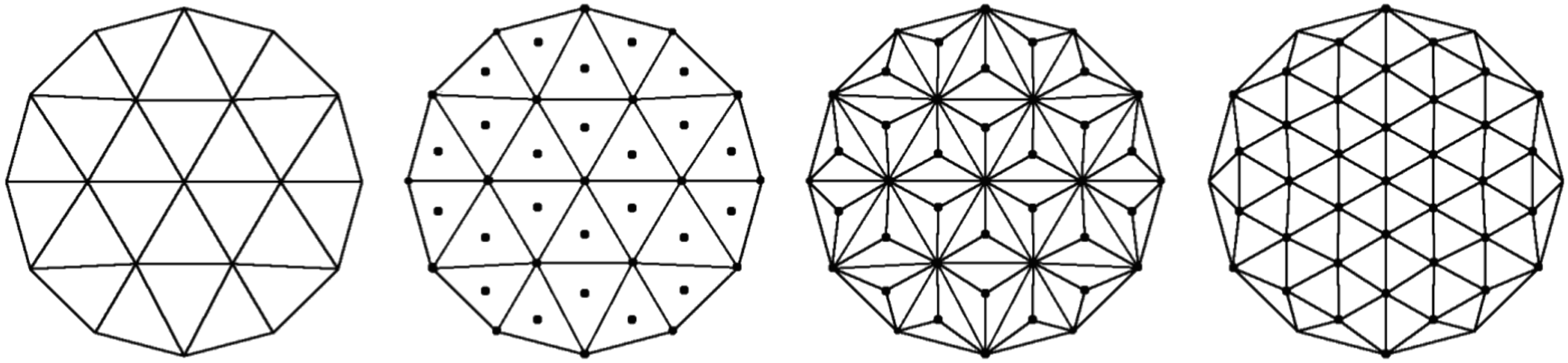
$$V^1 = \frac{6}{8} V + \frac{Q_1^* + Q_2^*}{8}$$

(only if the vertices are neighbors on the boundary)

- Loop's scheme



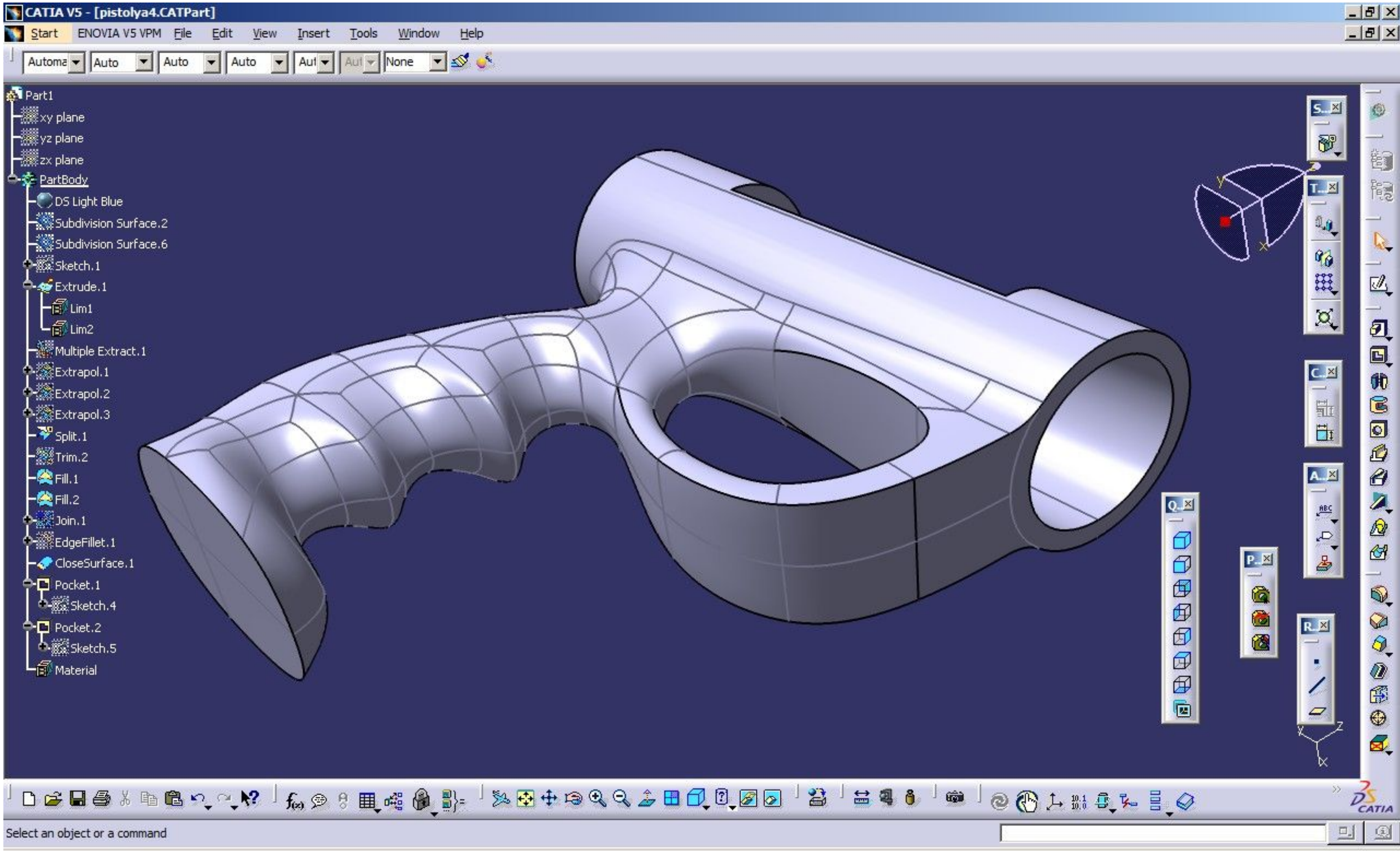
- Kobbelt's $\sqrt{3}$ scheme
  - See paper on the course's website
  - For a similar level of refinement, it generates less triangles than Loop's scheme



Subdivision surfaces

- Subdivision surfaces in CATIA

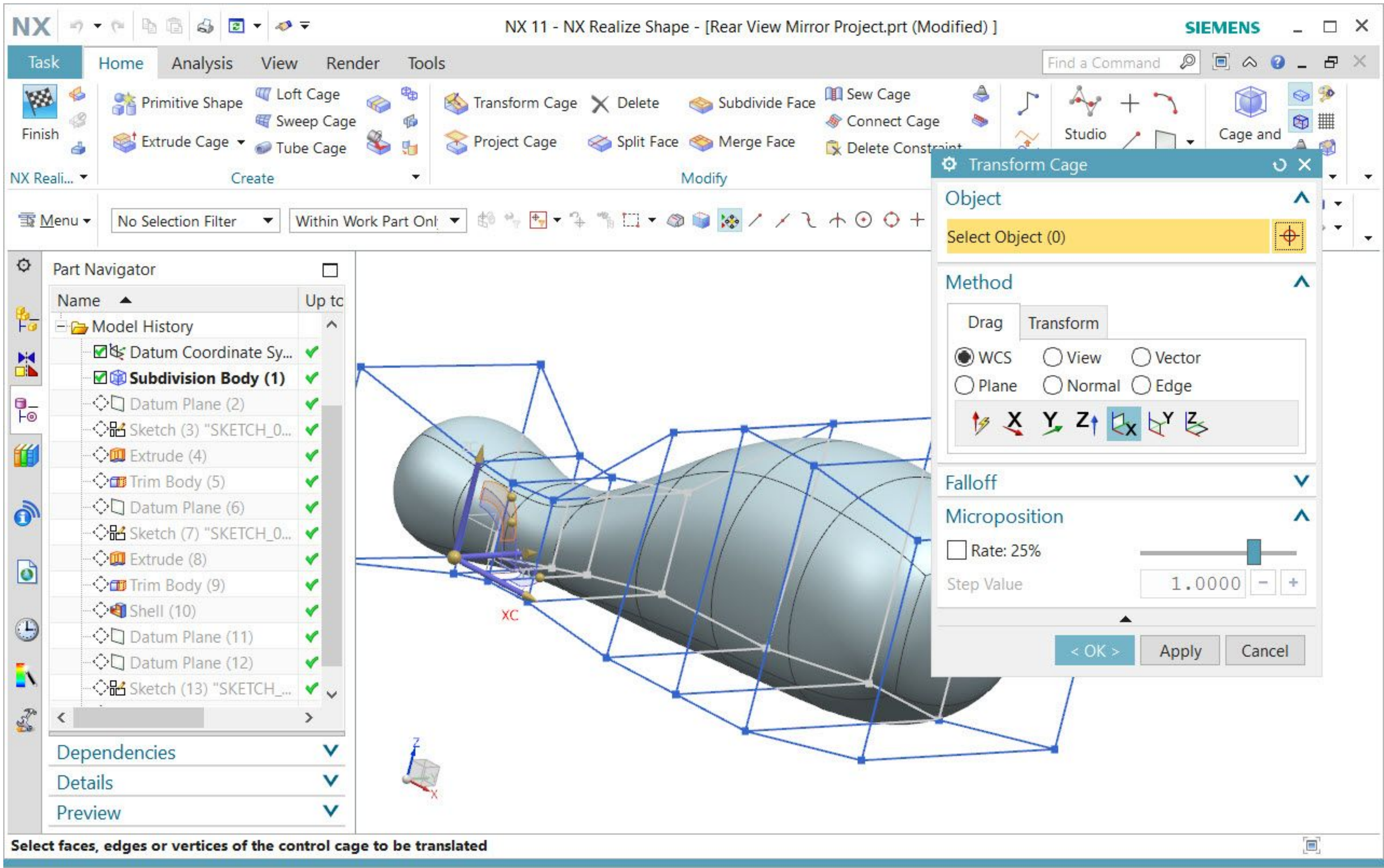
- A very easy-to-use design tool
- As S-s are equivalent to some class of B-Spline surfaces, they retain a good degree of accuracy
- “CATIA Shape” module Imagine Shape (IMA) tool



Subdivision surfaces

- … and in NX

- A very easy-to-use design tool
- As S-s are equivalent to some class of B-Spline surfaces, they retain a good degree of accuracy
- “NX Realize shape” tool



- Zoo of Splines ...

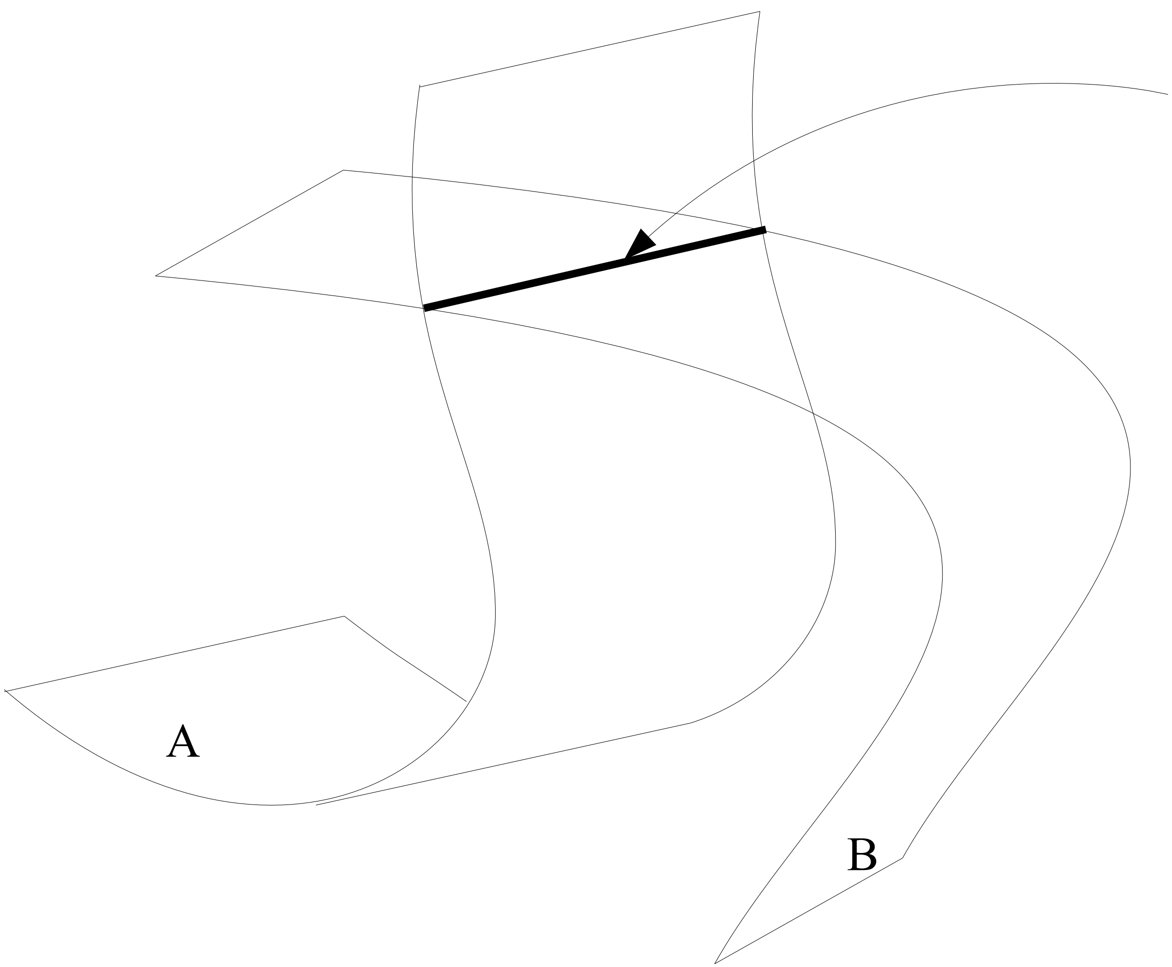
Lg-splines
Analytic splines
Parabolic Arc splines
Beta splines
B-splines
Bernoulli splines
Box splines
Cardinal splines
Circular splines
Complete splines
Nu-splines
Natural splines
L-splines
Whittaker splines

Nonlinear splines
D^m splines
Discrete splines
Euler splines
Exponential splines
Gamma splines
GB-splines
HB-splines
Hyperbolic Splines
Complete Monosplines
Tchebycheffian splines
Tension splines
Trigonometric splines
Bézier splines

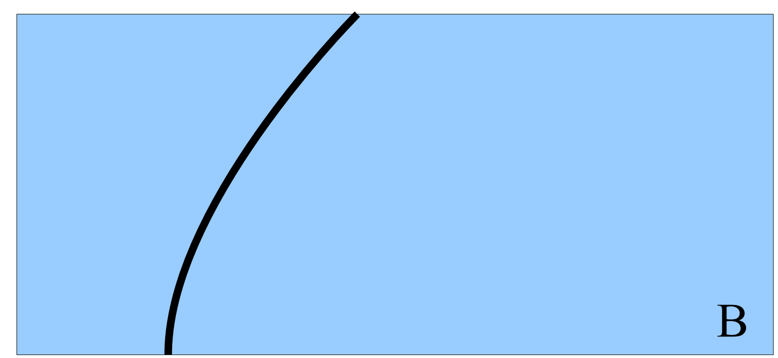
One-sided splines
Parabolic splines
Perfect splines
Periodic splines
Poly-splines
Rational splines
Simplex splines
Spherical splines
Taut splines
Complex splines
Confined splines
Deficient splines
Thin plate splines

Solid models and B-REP

- Classical modelling problem : the intersection



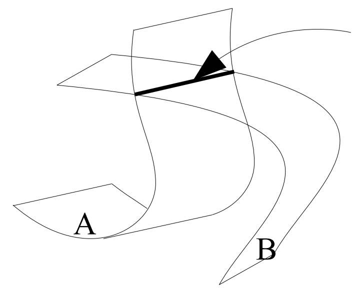
3 independent representations of the intersection :
- a 3D NURBS curve (giving points in the global XYZ coordinate system)
- a 2D NURBS curve in the parametric space of surface A (giving 2D points in the coordinate system of the parametric space of surface A)
- Idem for surface B



Parametric space of surface B

Solid modelling

- Theoretically, these three representations are equivalent ...



3 independent representations of the intersection :
- a 3D NURBS curve
- a 2D NURBS curve (parametric space of surface A)
- a 2D NURBS curve (parametric space of surface B)

- In practice, there are numerical approximations
  - NURBS are finite approximation spaces; therefore approximation/interpolation errors do occur.
  - The use of floating point numbers with a finite binary representation of the mantissa lead to numerical errors
- There is no robust way to ensure, in **a geometrical sense**, that a curve located on surface A is the same as the corresponding curve on surface B, *i.e.* that both surfaces are neighbours, and share the same edge.

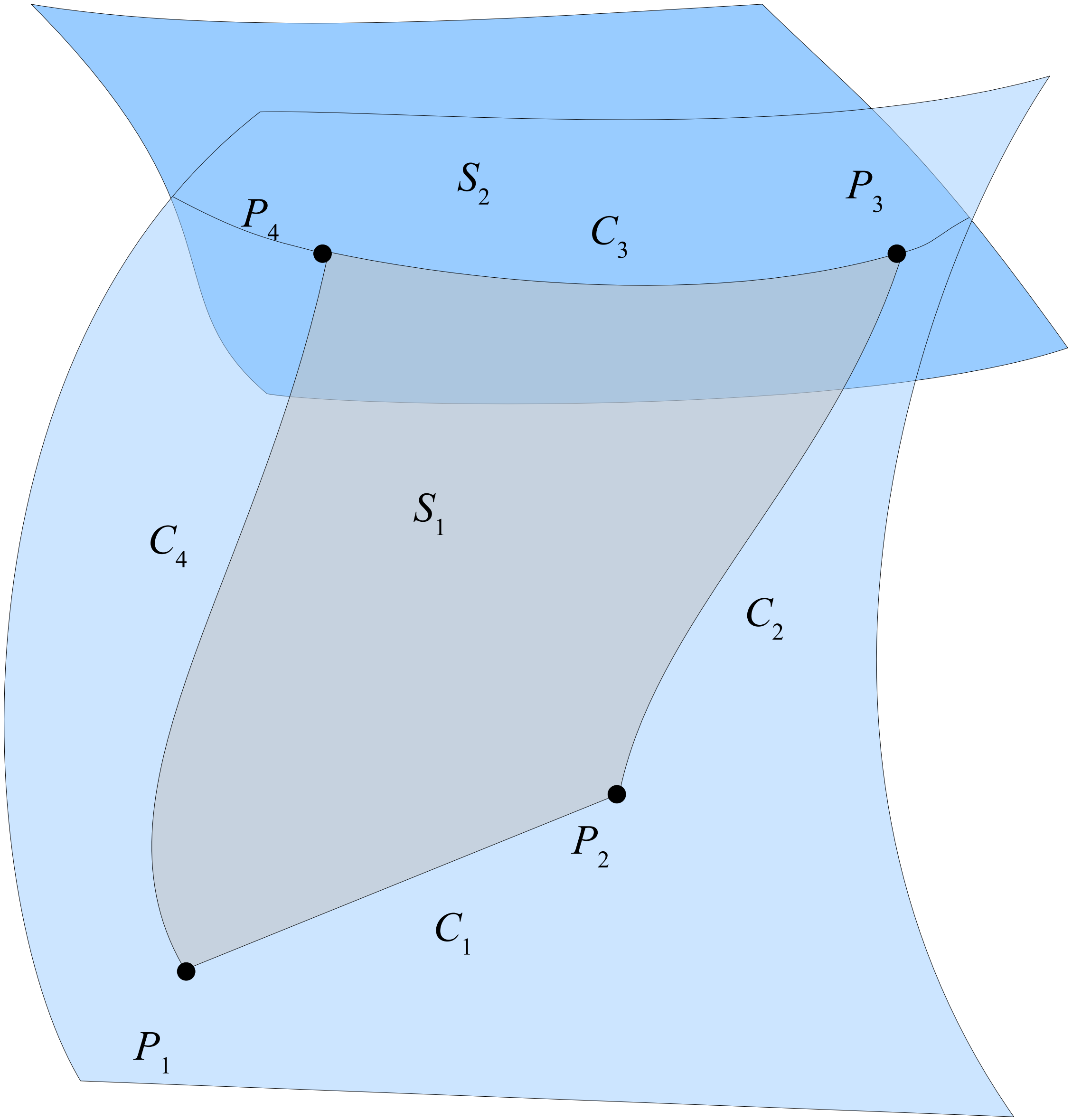
Solid modelling

- Definition of a **topology** : non geometric relations between entities.
- This allows to unify the calculations (of points, normals, etc...) on entities shared (or bounding) other entities (*eg.* an edge shared by surfaces).
- It also allows the explicit definition of volumes – from the surfaces that bound the volume.
- It may also solve the problem of orientation of surfaces

Solid modelling

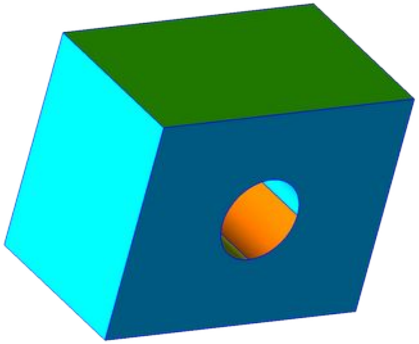
- Topology

B-REP model



Solid modelling

- B-Rep model
  - « Boundary representation »
  - Model based on the representation of surfaces
  - Model of exchange (STEP format) and definition
  - The “natural” set of operators is richer than for CSG
    - Extrusion, chamfer etc ...
  - Does not carry the history of construction of the model (whereas CSG usually does)



Solid modelling

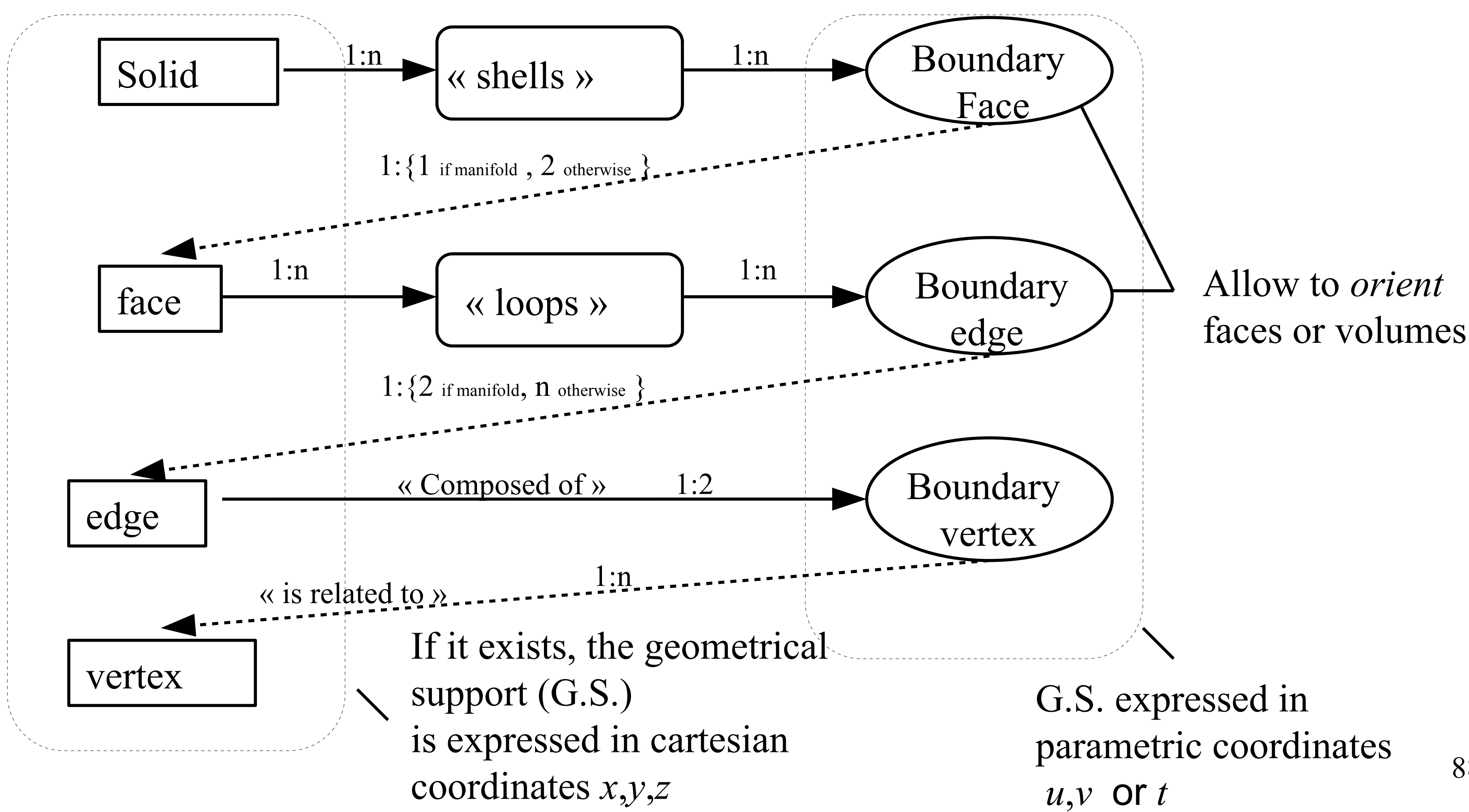
- B-Rep model
  - Consists of two types of information :
    - Geometric

      Geometric information is used for defining the spatial position, the curvatures, etc...

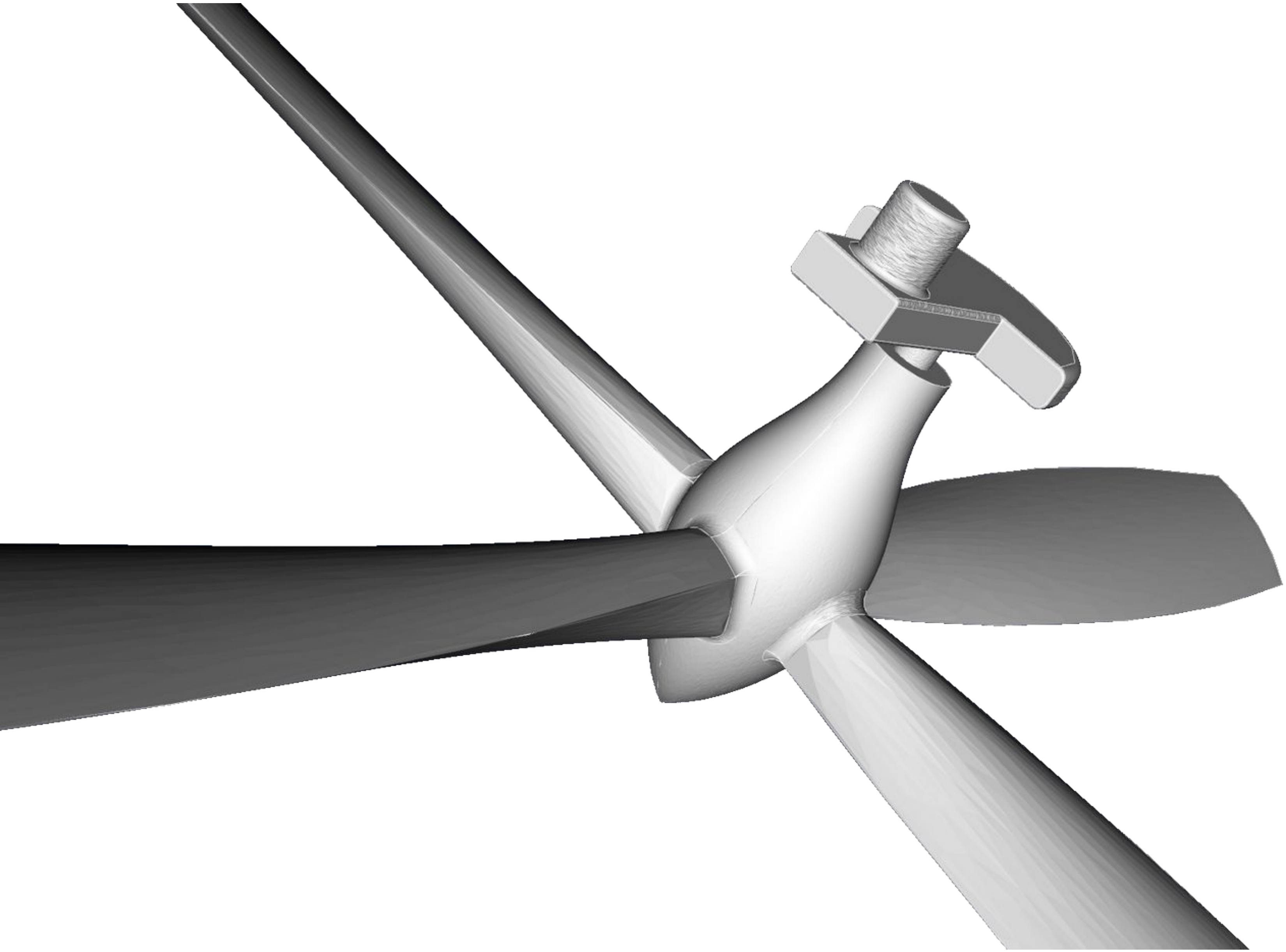
      That's what we have seen until now – NURBS curves and surfaces !
    - Topological

      This allows to make links between geometrical entities.
  - Two types of entities
    - Geometric entities: (volume), surface, curve, point
    - Topological entities : solid, face, edge, vertex
  - A topological entity “lies on” a geometric entity, which is its geometrical support (when existing)

- B-Rep model
  - Complete hierarchical model



Solid modelling



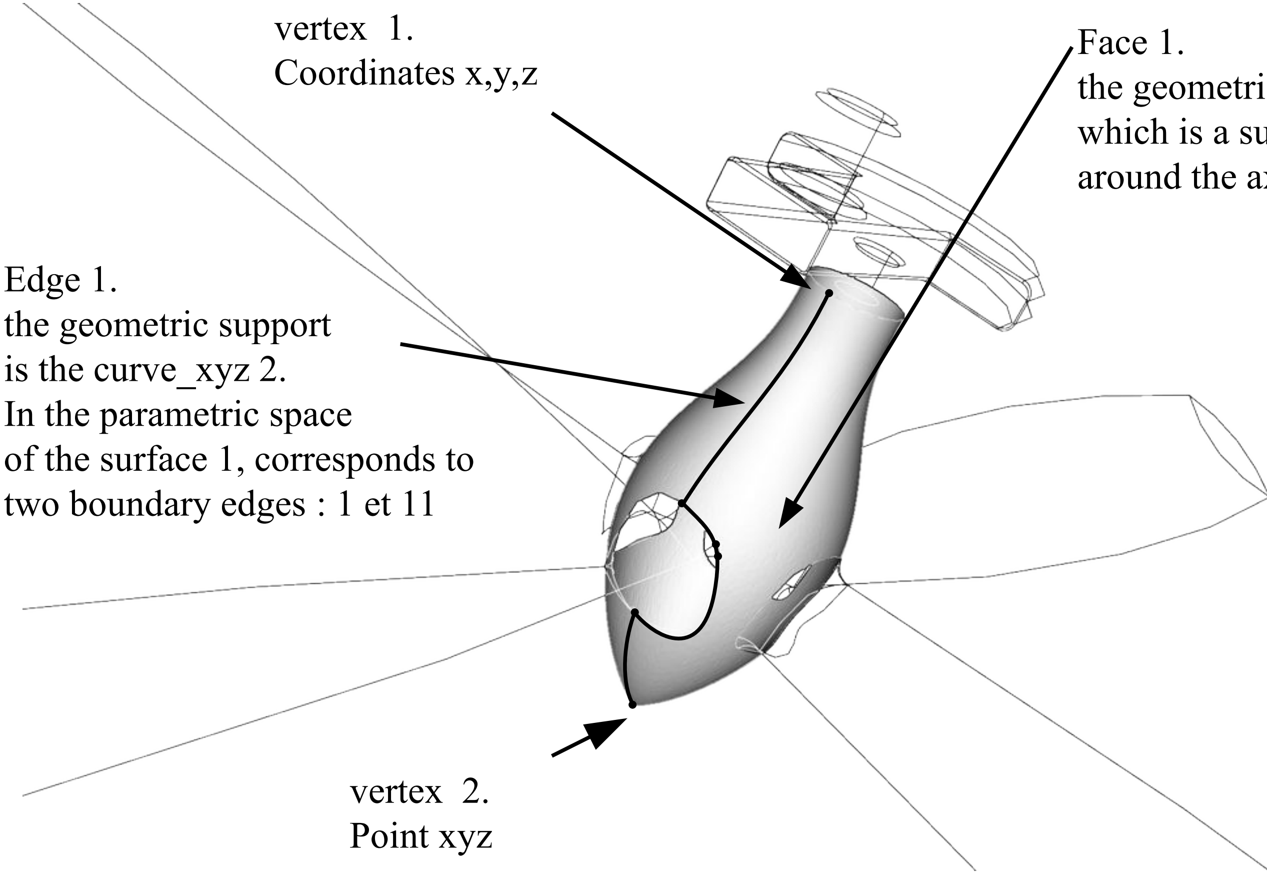
CAD & Computational Geometry

Solid modelling

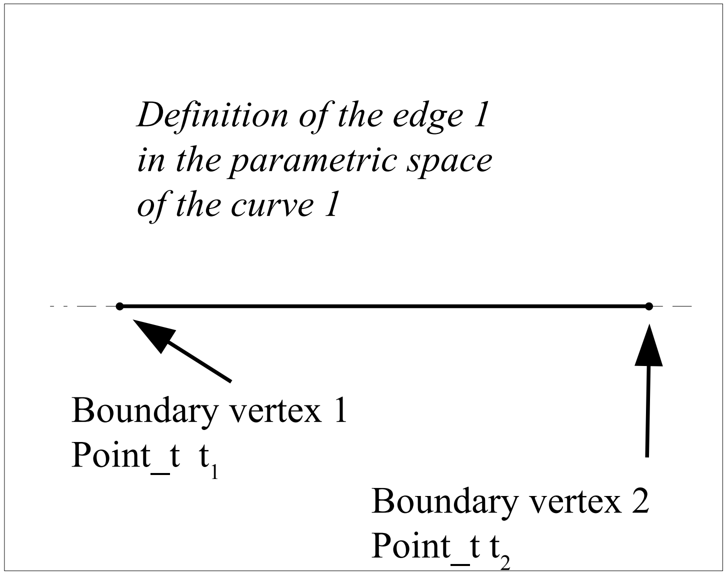
vertex 1.
Coordinates x,y,z

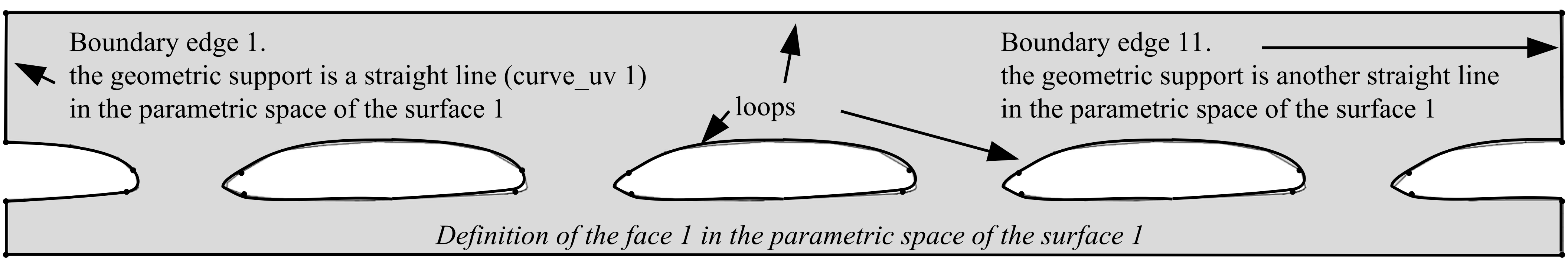
Face 1.
the geometric support is surface 1, which is a surface of revolution around the axis (Oz).

Edge 1.
the geometric support is the curve_xyz 2.
In the parametric space of the surface 1, corresponds to two boundary edges : 1 et 11



vertex 2.
Point xyz

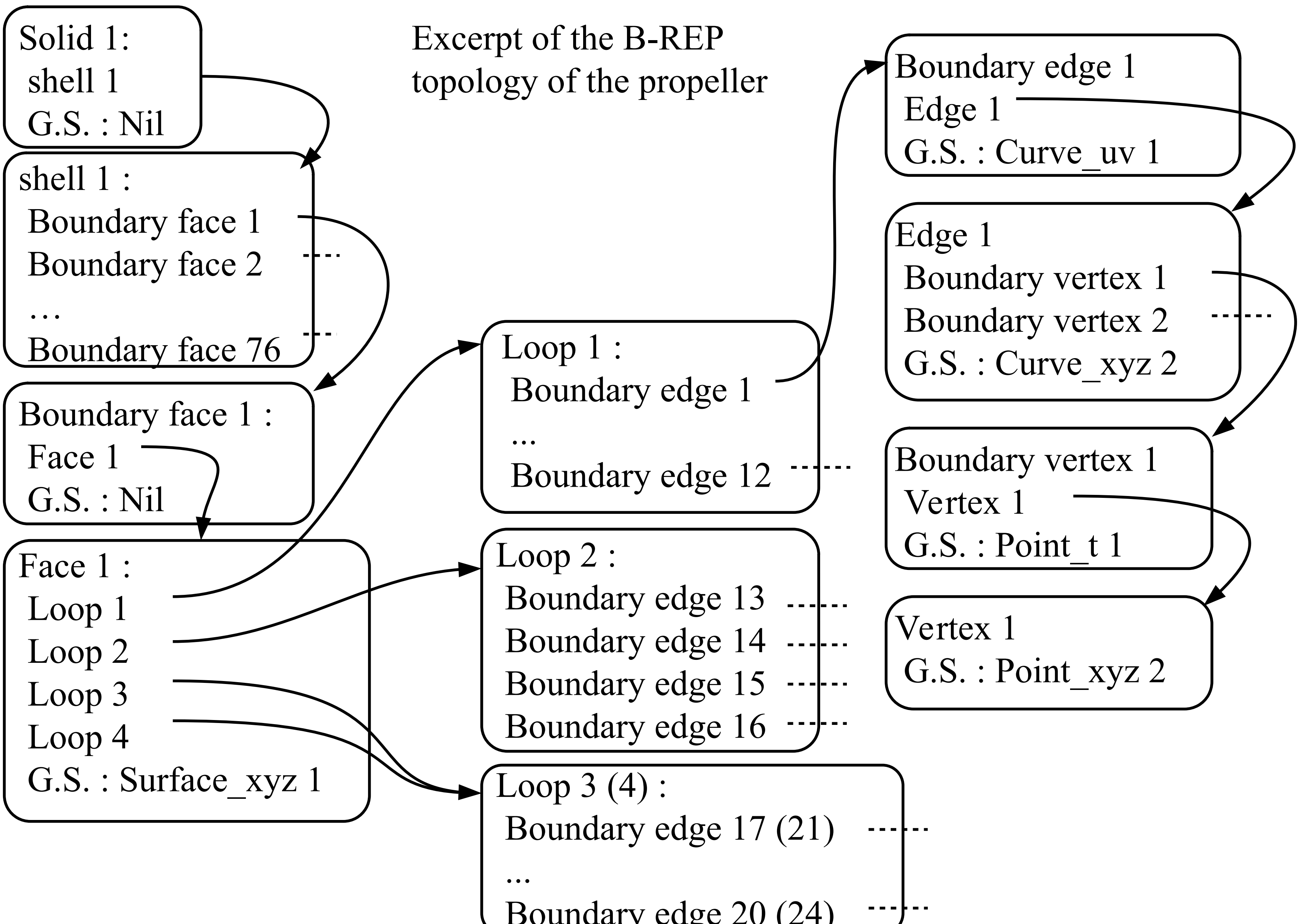




Boundary edge corresponding to a **degenerated edge**
(zero length and identical points of departure and arrival: vertex 2)

Solid modelling

Excerpt of the B-REP topology of the propeller



Solid modeling

Links between the B-REP topology and the actual geometry of the propeller

Boundary edge 1
Edge 1
G.S. : Curve_uv 1

Curve_uv 1 :
Straight line
application (t) → (u,v)

Edge 1
Boundary vertex 1
Boundary vertex 2
G.S. : Curve_xyz 2

Curve_xyz 2 :
NURBS Curve
application (t) → (x,y,z)

Boundary vertex 1
vertex 1
G.S. : Point_t 1

Point_t 1
t = t1

Vertex 1
G.S. : Point_xyz 2

Point_xyz 2
x=x1, y=y1, z=z1

Face 1 :
Loop 1
Loop 2
Loop 3
Loop 4
G.S. : Surface_xyz 1

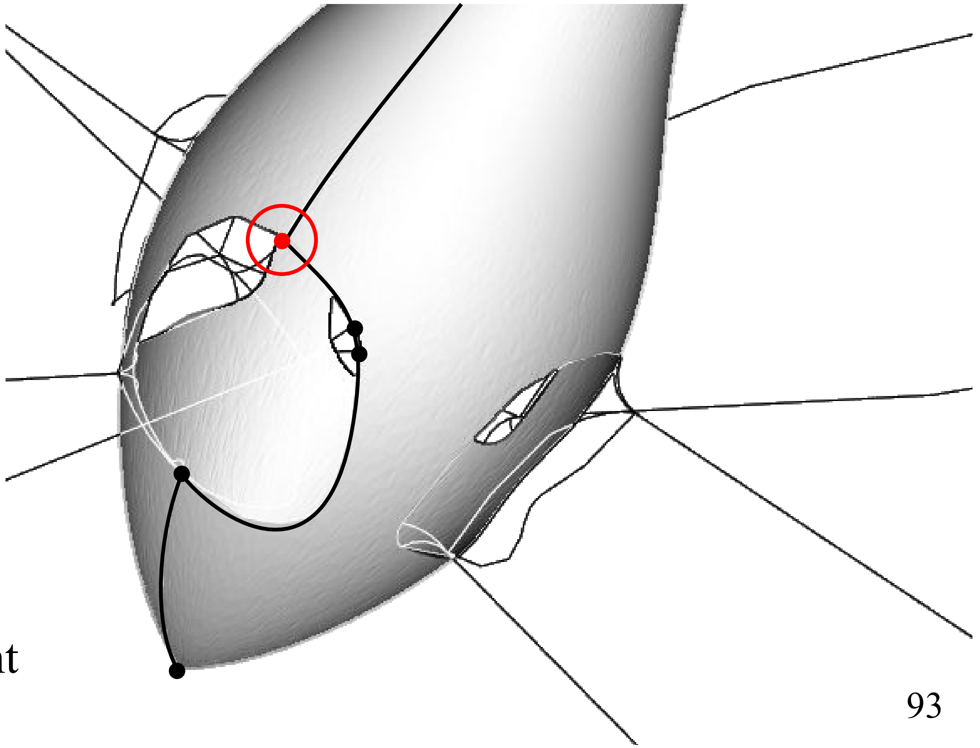
Surface_xyz 1 :
Surface of revolution
NURBS Surface
application (u,v) → (x,y,z)

How to obtain the (x,y,z) coordinates of the encircled point ?

1 – Use the 3D vertex directly (Point_xyz xxx)

2 – Use the boundary vertices for every 3D edge (there are 3 such edges) (Point_t t1,t2,t3)
Then use those (t) to get (x,y,z) by the 3D edges

3 – Use the boundary vertices of the 2D boundary edges in the face (there are 2 faces, so 4 of them), (Point_t t'1,t'2,t'3,t'4). Then use those (t) to obtain coordinates (u,v) in the parametric space of the face, thanks to 2D curves, finally, use those (u,v) to obtain (x,y,z) thanks to the geometry of the face.



So there exists 8 different ways. Nothing indicates that the 8 set of 3D coordinates are exactly equal (there are numerical approximations). Only topology allows us to say that those 8 points are all referring to the same point … at least conceptually.

- B-Rep model
  - Euler characteristic for polyhedra

$$\chi(S) = v - e + f$$

  - Euler – Poincaré formula

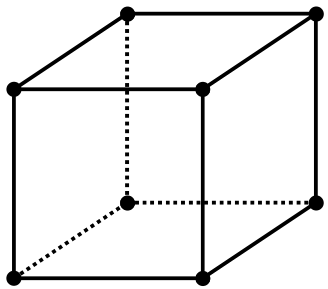
$$\chi(S) = v - e + f - r = 2(s - h)$$

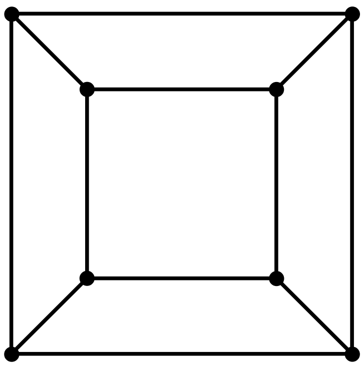
with

v	=	number	of vertices
f	=	"	of faces
e	=	"	of edges
s	=	"	of solids (independent volumes)
h	=	"	of holes – going through (topol. gender)
r	=	"	of internal loops (rings)

Euler's formula

- Euler characteristic



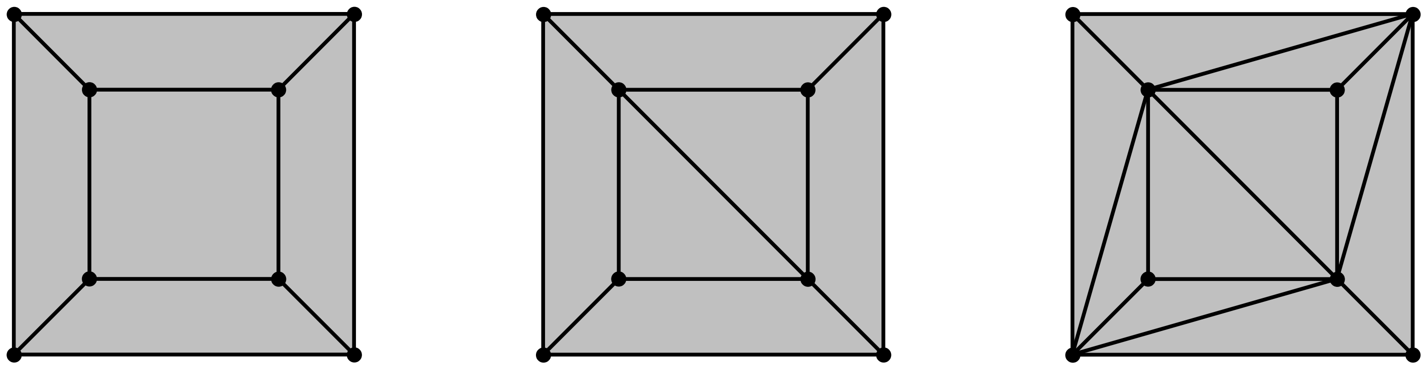


Example : Cube
$v - e + f = \kappa$

Opened and flattened Cube
$v - e + f = \kappa$ -1

Step 0 : we take a face off the polyhedron and flatten it to obtain a plane graph

Euler's formula



$v - e + f = \kappa - 1$

$+1e\,,\ +1f$

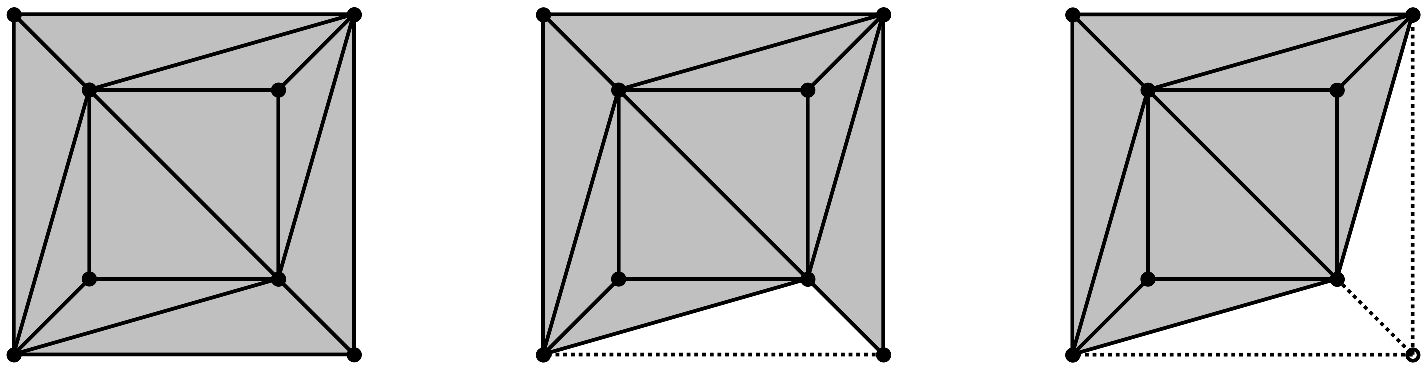
$v - e + f = \kappa - 1$

$+5e\,,\ +5f$

$v - e + f = \kappa - 1$

Step 1 : Repeat the following operation :
For each non triangular face, add one edge linking non related vertices.
Each tome, the number of edges and faces is increased by 1.
This is repeated until no non triangular faces remain.

Euler's formula



$v - e + f = \kappa - 1$

$-1e\ ,\ -1f$

$v - e + f = \kappa - 1$

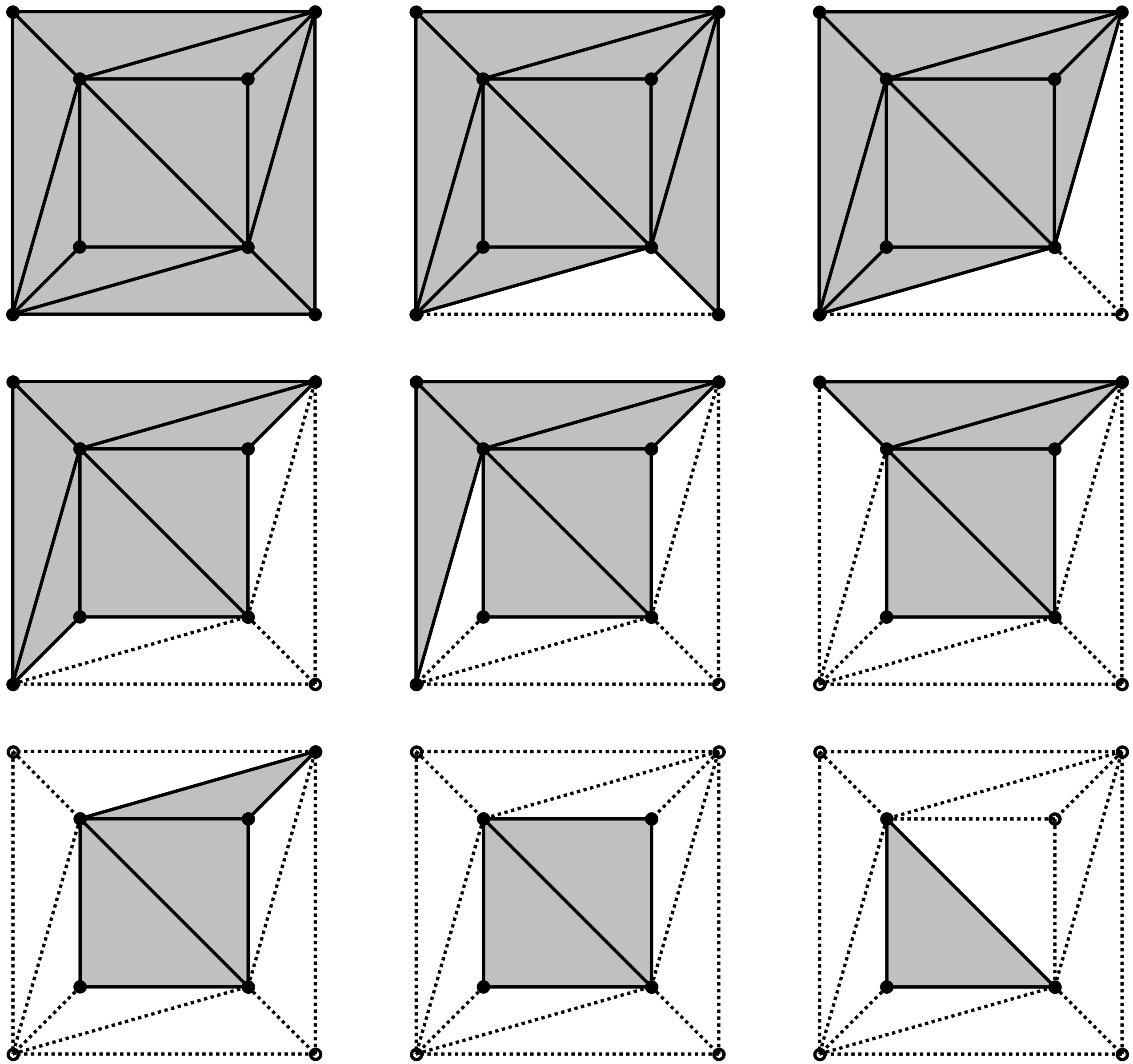
$-2e\ ,\ -1f,\ -1v$

$v - e + f = \kappa - 1$

Step 2 : One alternates between these two operations
- Preferentially, delete triangles that have 2 boundary edges.
 Every time; e decreases by 2 and f and v by 1.
- Then, delete triangles with only one boundary edege.
 Each tile, e and f decrease by 1.

This until only one triangle remain.

Euler's formula



$$v - e + f = \kappa - 1$$

Euler's formula

- Every polygon can be decomposed into triangles

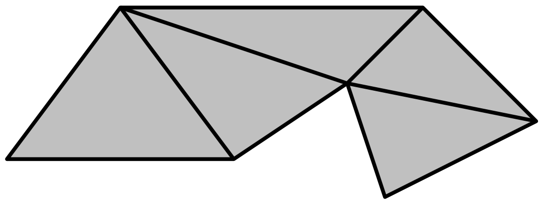
  Therefore, by applying the three operations described in the previous slides, we can transform the planar graph into a triangle without changing Euler's characteristic. The triangle obviously satisfies

  $v - e + f = \kappa - 1$

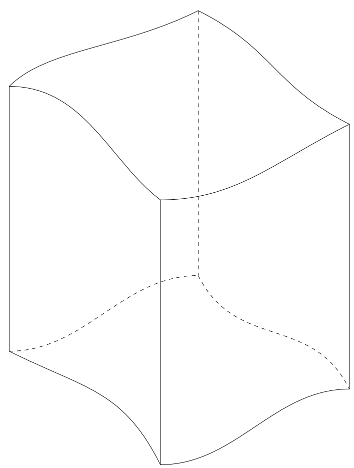
  with $\kappa - 1 = 1$

  As a consequence, the planar graph verifies the formula.
- So the initial polyhedron satisfies :

$$v - e + f = \kappa = 2$$



- Necessity to take “rings” into account - inside faces

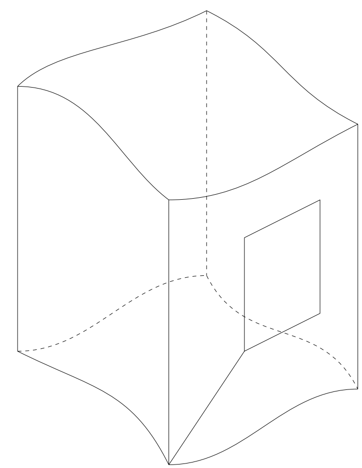


polyhedron

$$\chi(S)=2$$

$$\chi(S)=v-e+f$$

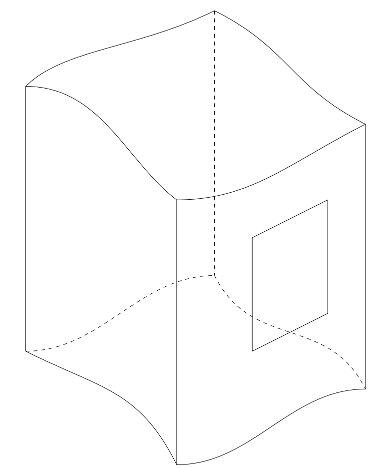
$$\chi(S)=8-12+6=2$$



polyhedron

$$\chi(S)=12-17+7=2$$

OK



Polyhedron with rings

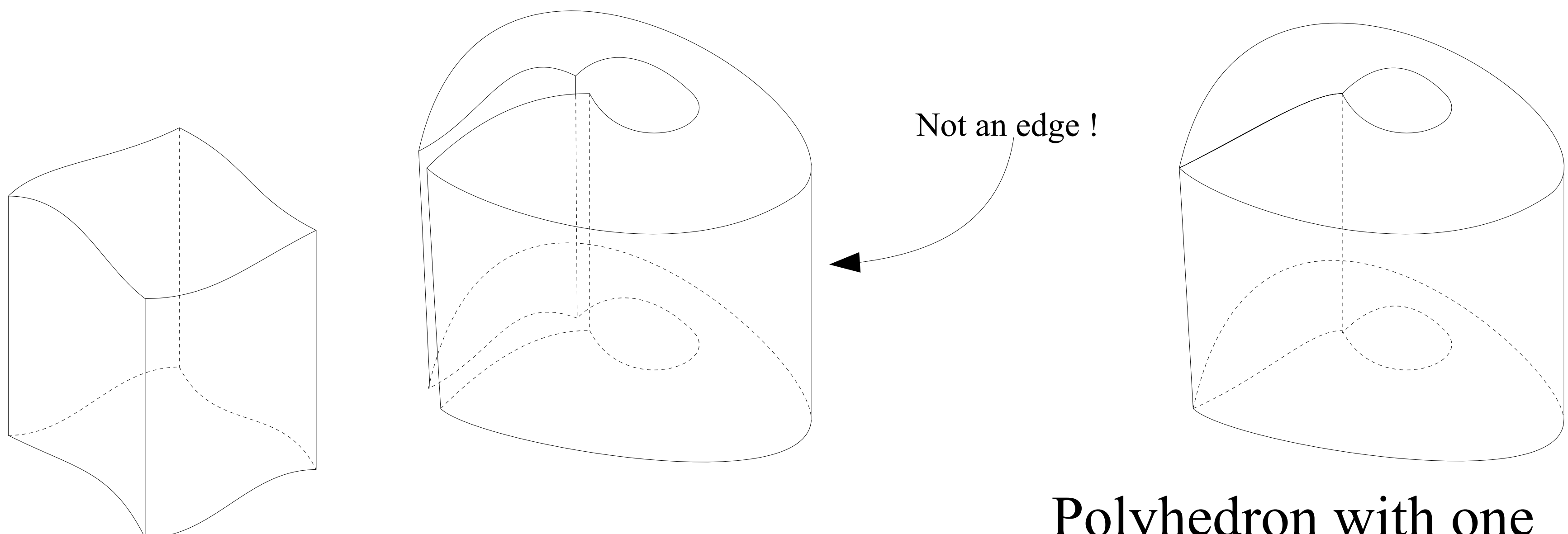
$$\chi(S)=12-16+7\neq 2$$

Not OK !

Contribution of the ring

$$\chi(S)=v-e+f-r=2$$

- Necessity to take “holes” into account



polyhedron

$$\chi(S)=2$$

$$\chi(S)=v-e+f$$

$$\chi(S)=8-12+6=2$$

“Warped”polyhedron

$$\chi(S)=8-12+6=2$$

OK

Polyhedron with one hole, 4 edges less, 2 faces less, 4 vertices less

$$\chi(S)=4-8+4\neq 2$$

Not OK !

Contribution of the hole

$$\chi(S)=v-e+f-r+2h=2$$

Solid modeling

- Every B-rep model is identifiable (topologically) to a « point » in a 6-dimensional vector space.
  - Vector space of coordinates v, e, f, s, h, r.
- Any topologically valid model shall verify the Euler-Poincaré relation
  - This relation defines an « hyperplane » (of dimension 5) in a 6-dimensional space
  - The equation of this hyperplane is :

$$v-e+f-2s+2h-r=0$$

Solid modeling

$$v - e + f - 2s + 2h - r = 0$$

- We can update a valid solid and modify the 6 numbers characterising a model with a transformation that yields a valid solid for which :

$$v + \Delta v - e - \Delta e + f + \Delta f$$
$$-2s - 2\Delta s + 2h + 2\Delta h - r - \Delta r = 0$$

$$\Rightarrow \Delta v - \Delta e + \Delta f - 2\Delta s + 2\Delta h - \Delta r = 0$$

  - In this way, add a vertex ($\Delta v = 1$) must be accompanied, one way or another, by addition of an edge ($\Delta e = 1$) OR of the withdrawal of a face ($\Delta f = -1$), etc...
  - Elementary operations satisfying the Euler-Poincaré relation are called **Euler operators**.
  - They allow staying on the « hyperplane » of validity while changing the topological configuration

Solid modelling

- Euler operators
  - The use of Euler operators guarantees the *topological* validity of the result
  - Here we don't check the *geometric* validity (self-intersections etc...)
  - We identify them under the form : ***MaKb*** where ***M*** = Make ***K*** = Kill and ***a*** and ***b*** are a sequence of entities : vertex, edge, face, solid, hole or ring.
    - In total, there are 99 Euler operators aiming to modify the number of entities by **at most** one unit.
      - These are divided in 49 + 49 inverses, plus the identity operator.
    - Among those 49 operators , we can chose 5 linearly independent operators (the hyperplane has 5 dimensions)
    - Those 5 independent operators form a base for the hyperplane of topologically admissible models

- Example of a set of Euler operators

inverses

MEV, **M**ake an **E**dge and a **V**ertex
MEF, **M**ake an **E**dge and a **F**ace
MEKR, **M**ake an **E**dge and **K**ill a **R**ing
MVFS, **M**ake a **V**extex, a **F**ace and a **S**hell
KFMRH, **K**ill a **F**ace **M**ake a **R**ing and a **H**ole

KEV, **K**ill an **E**dge and a **V**ertex
KEF, **K**ill an **E**dge and a **F**ace
KEMR, **K**ill an **E**dge and **M**ake a **R**ing
KVFS, **K**ill a **V**ertex, a **F**ace and a **S**hell
MFKRH, **M**ake a **F**ace, **K**ill a **R**ing and a **H**ole

- Proof by Mäntylä (1984) that those operators allow to build every valid solid (since they are independent)
- Those operators form a base of the space of valid configurations (the « hyperplane »)

There are three types of operators in this set :

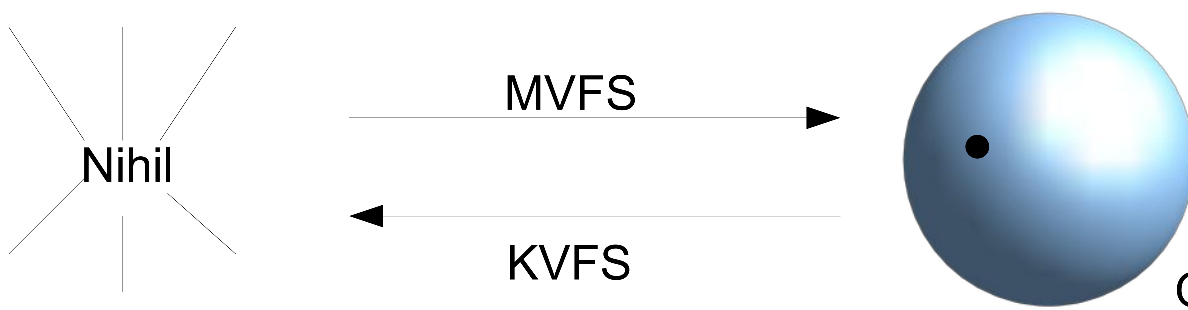
- Skeleton operators **MVFS** and **KVFS**
  - Allow to build/destroy elementary volumes
- Local operators **MEV**, **KEV**, **MEF**, **KEF**, **KEMR**, **MEKR**
  - Allow to modify connectivities for existing volumes
  - Don't modify fundamental topological characteristics of the surfaces - nb of handles/ holes (topological gender) and number of independent volumes
- Global operators **KFMRH** and **MFKRH**
  - Allow to add / remove “handles” (change the topological gender)

- Only the skeleton and global operators do change the topological gender.

- Euler operators
  - « Skeleton » operators

    MVFS; KVFS

    - Allow to « build » an « elementary » volume from void (which is an admissible topological structure) – or destroy it.



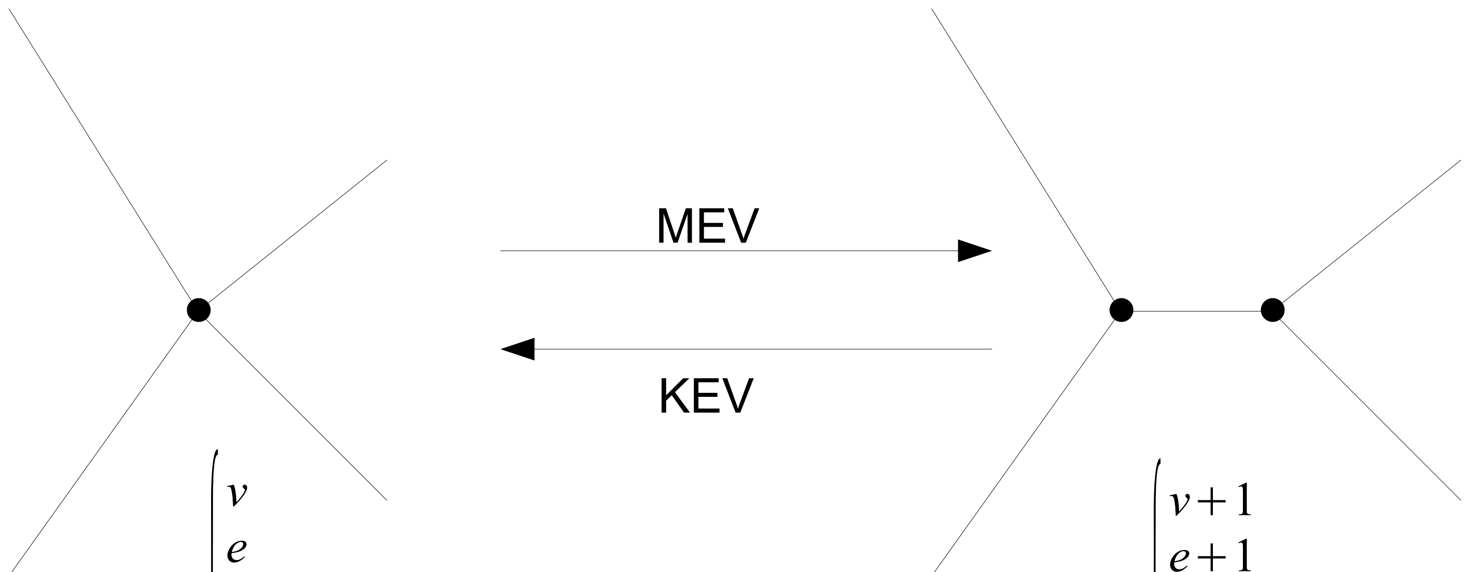
$$\begin{cases} v=0 \\ e=0 \\ f=0 \\ h=0 \\ r=0 \\ s=0 \end{cases}$$

$$\begin{cases} v=1 \\ e=0 \\ f=1 \\ h=0 \\ r=0 \\ s=1 \end{cases}$$

Only one face
 - its boundary is reduced
to a single vertex

- Euler operators
  - Local operators

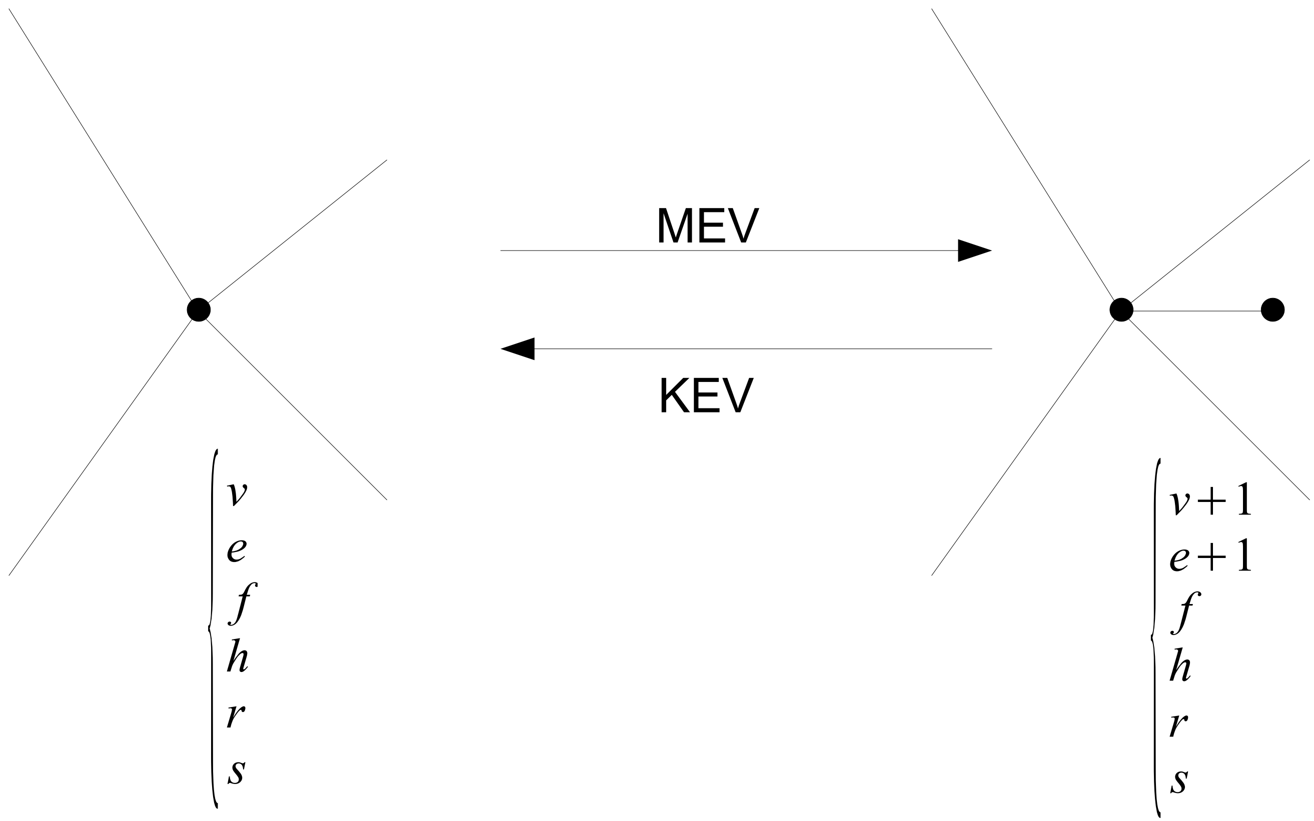
    MEV, **KEV** (case 1)



$$\begin{cases} v \\ e \\ f \\ h \\ r \\ s \end{cases} \qquad \begin{cases} v+1 \\ e+1 \\ f \\ h \\ r \\ s \end{cases}$$

- Euler operators
  - Local operators

    MEV, **KEV** (case 2)



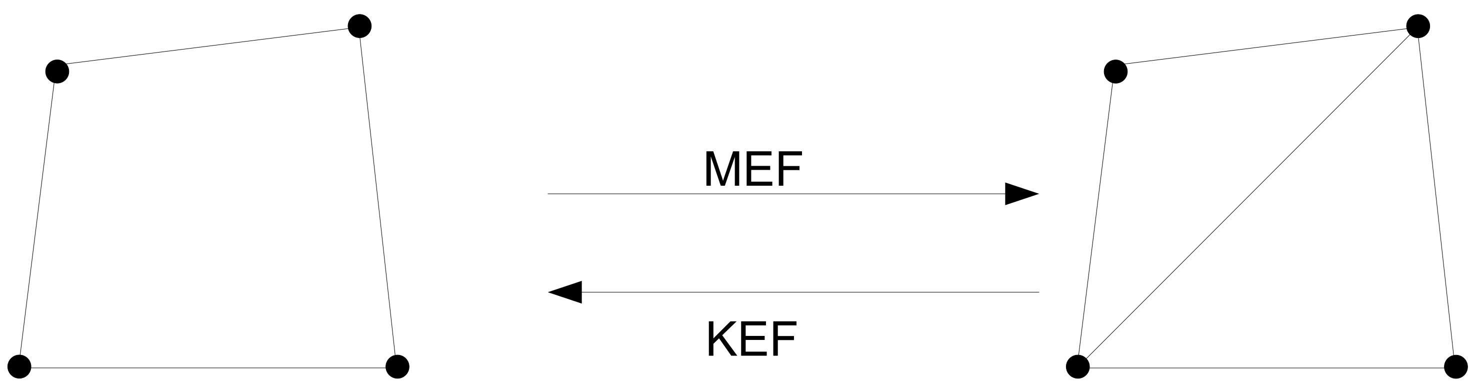
- Euler operators
  - Local operators

    MEV, **KEV** (case 3)

$$\begin{cases} v=1 \\ e=0 \\ f=1 \\ h=0 \\ r=0 \\ s=1 \end{cases} \qquad \begin{cases} v=2 \\ e=1 \\ f=1 \\ h=0 \\ r=0 \\ s=1 \end{cases}$$

- Euler operators
  - Local operators

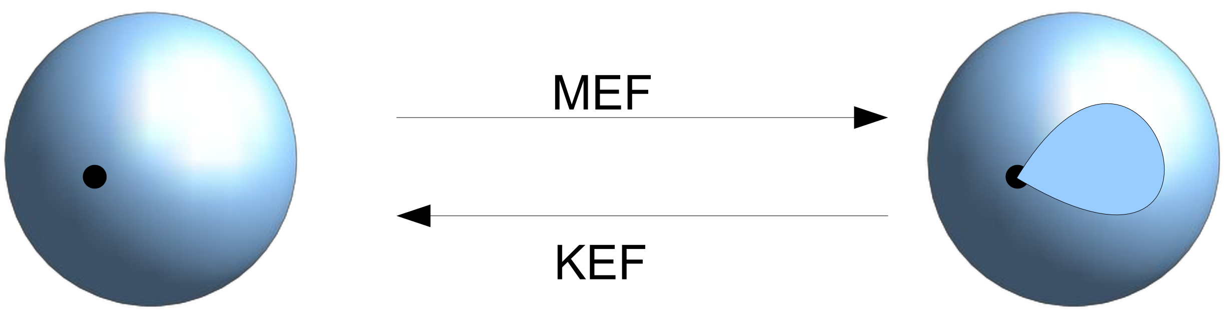
    MEF, **KEF** (case 1)



$$\begin{cases} v \\ e \\ f \\ h \\ r \\ s \end{cases} \qquad \begin{cases} v \\ e+1 \\ f+1 \\ h \\ r \\ s \end{cases}$$

- Euler operators
  - Local operators

    MEF, **KEF** (case 2)

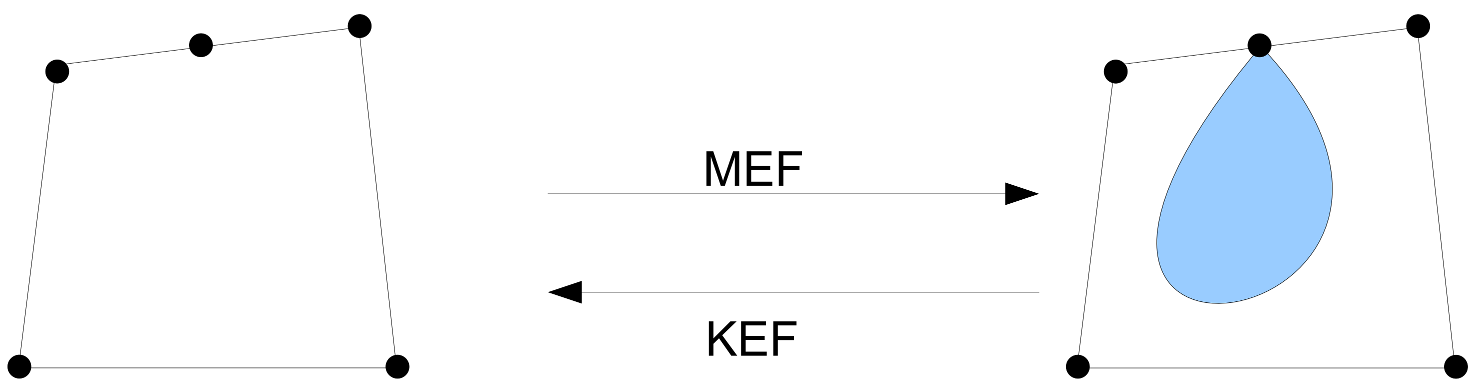


$$\begin{cases} v=1 \\ e=0 \\ f=1 \\ h=0 \\ r=0 \\ s=1 \end{cases}$$

$$\begin{cases} v=1 \\ e=1 \\ f=2 \\ h=0 \\ r=0 \\ s=1 \end{cases}$$

- Euler operators
  - Local operators

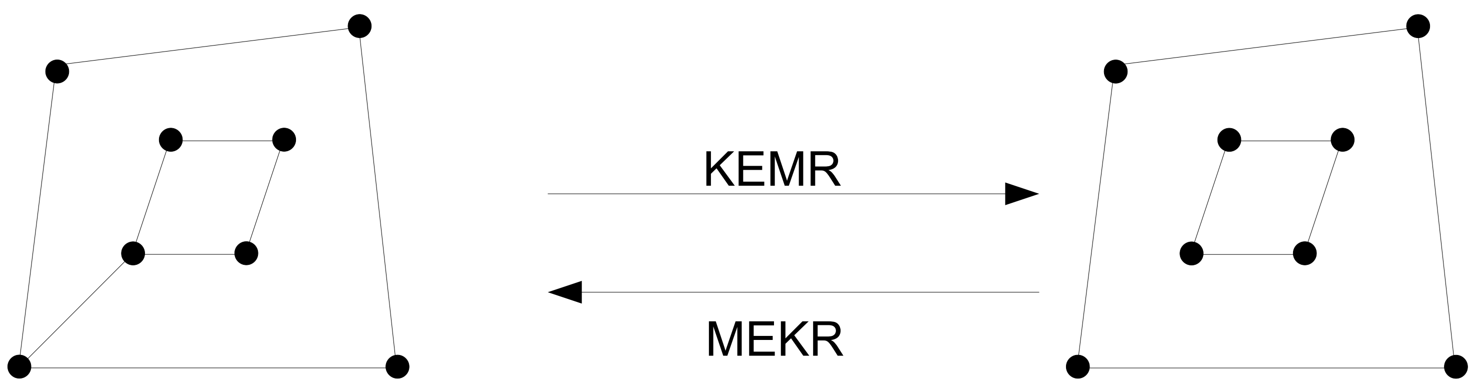
    MEF, **KEF** (case 3)



$$\begin{cases} v \\ e \\ f \\ h \\ r \\ s \end{cases} \qquad \begin{cases} v \\ e+1 \\ f+1 \\ h \\ r \\ s \end{cases}$$

- Euler operators
  - Local operators

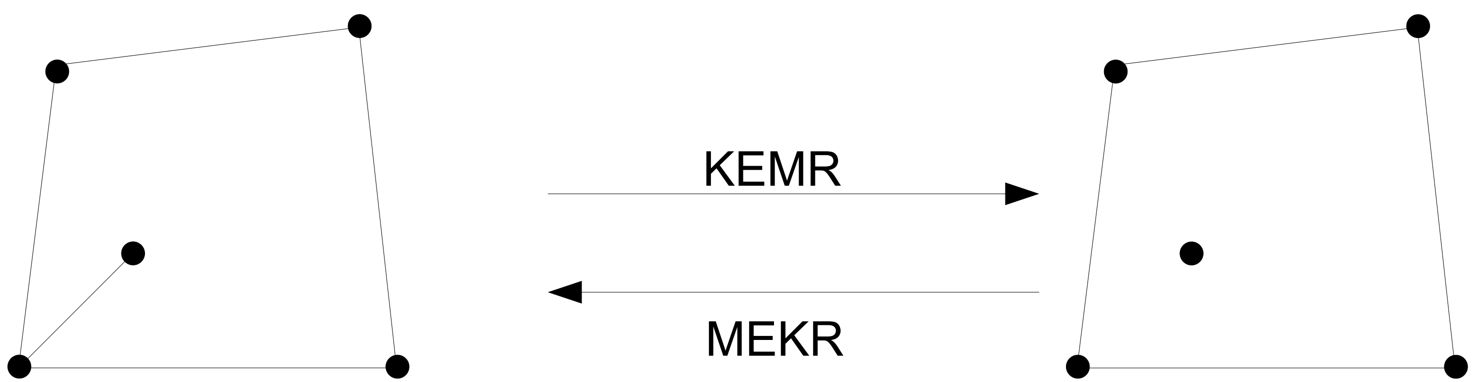
    KEMR, **MEKR** (case 1)



$$\left\{\begin{array}{l} v \\ e \\ f \\ h \\ r \\ s \end{array}\right. \qquad \left\{\begin{array}{l} v \\ e-1 \\ f \\ h \\ r+1 \\ s \end{array}\right.$$

- Euler operators
  - Local operators

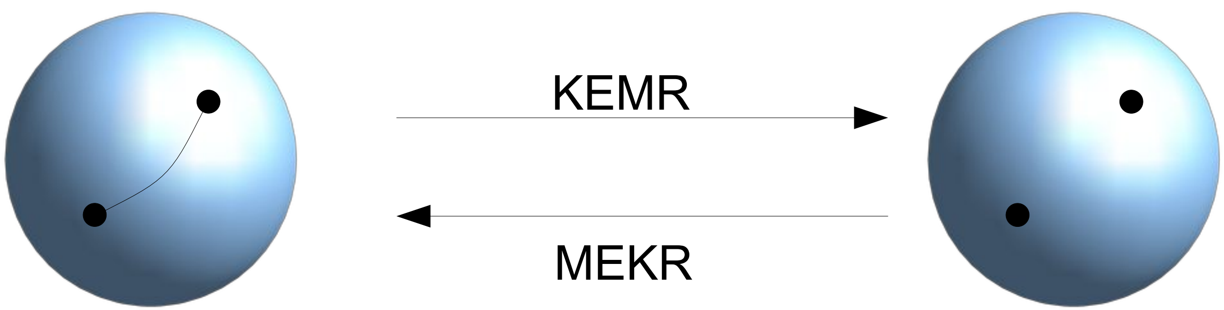
    KEMR, **MEKR** (case 2 : the loop for the internal ring is reduced to a single vertex)



$$\begin{cases} v \\ e \\ f \\ h \\ r \\ s \end{cases} \qquad \begin{cases} v \\ e-1 \\ f \\ h \\ r+1 \\ s \end{cases}$$

- Euler operators
  - Local operators

    KEMR, **MEKR** (case 3 : both loops are reduced to one vertex – one is the ring; the other is the external loop of the face)



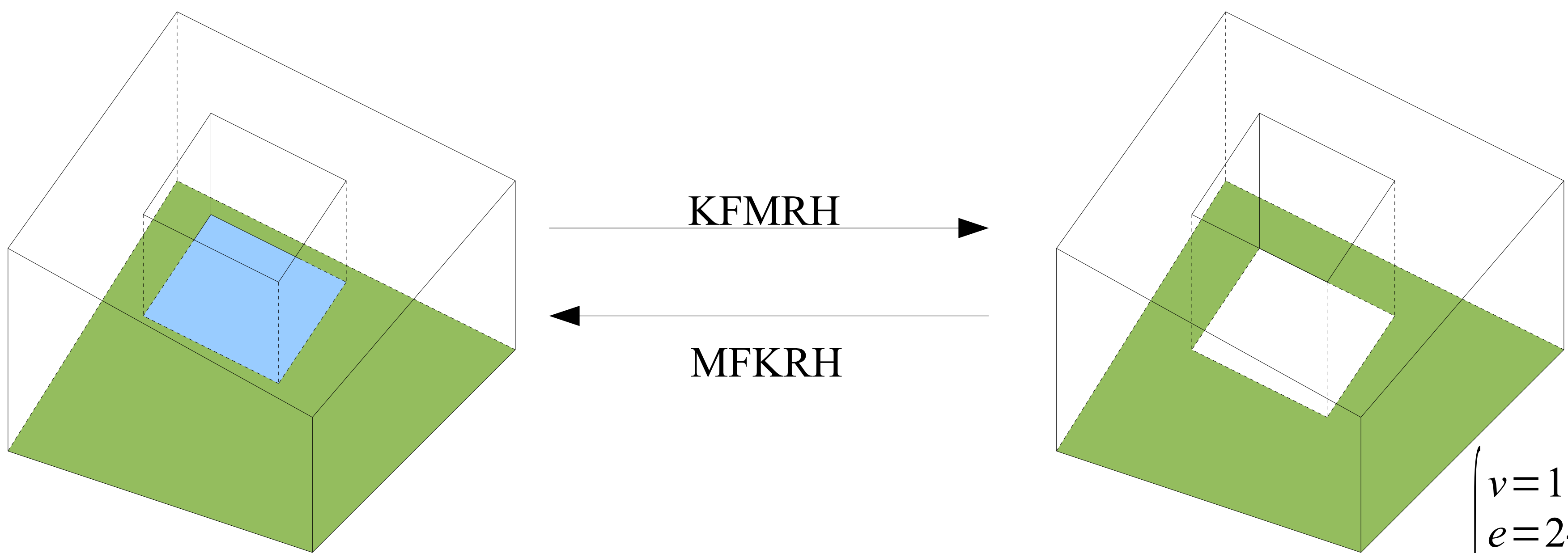
$$\begin{cases} v=2 \\ e=1 \\ f=1 \\ h=0 \\ r=0 \\ s=1 \end{cases} \qquad \begin{cases} v=2 \\ e=0 \\ f=1 \\ h=0 \\ r=1 \\ s=1 \end{cases}$$

CAD & Computational Geometry

Solid modelling

- Euler operators
  - Global operators

    KFMRH, **MFKRH** (case 1 : allow the creation / destruction of holes in a solid)

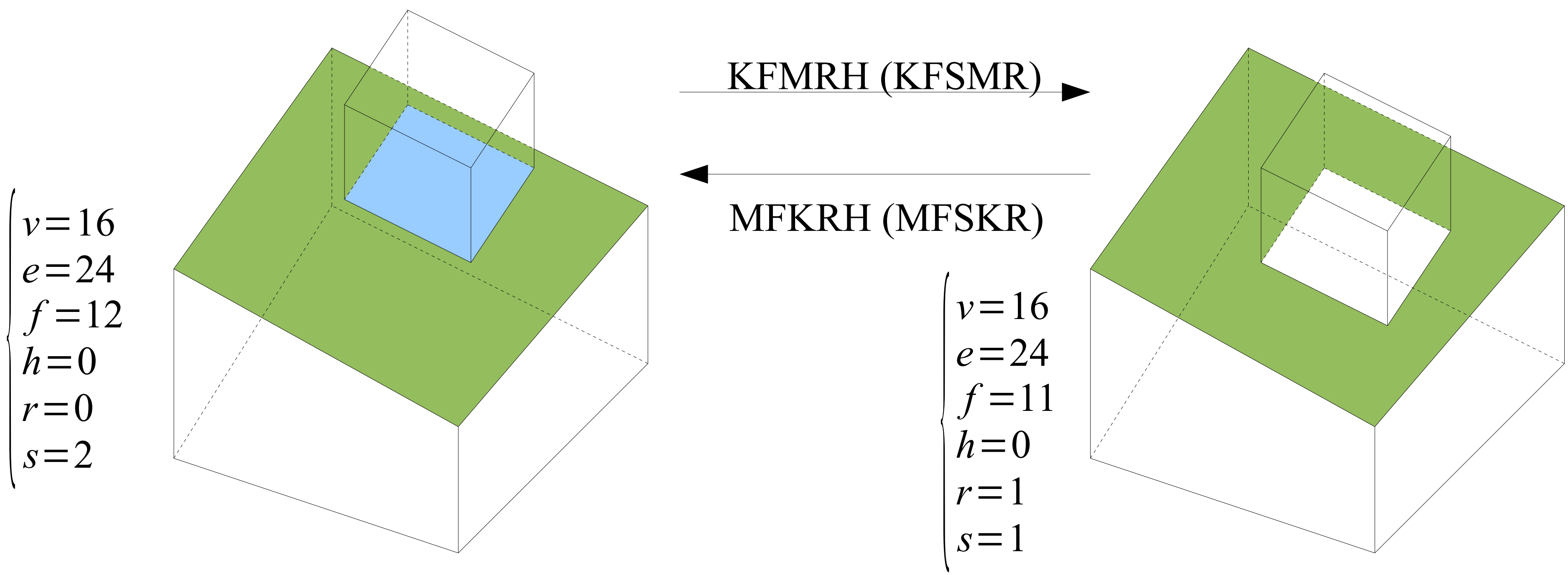


$$\begin{cases} v=16 \\ e=24 \\ f=11 \\ h=0 \\ r=1 \\ s=1 \end{cases}$$

$$\begin{cases} v=16 \\ e=24 \\ f=10 \\ h=1 \\ r=2 \\ s=1 \end{cases}$$

- Euler operators
  - Global operators

    KFMRH, **MFKRH** (case 2) : join two independent solids : here more judiciously called **K**ill **F**ace, **S**olid and **M**ake **R**ing (**KFSMR**)
  - Interpretation of global operators is sometimes confusing



- Example of use of Euler operators

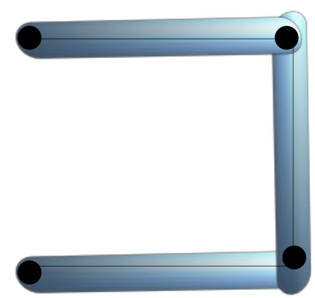
MVFS

$$\begin{cases} v=1 \\ e=0 \\ f=1 \\ h=0 \\ r=0 \\ s=1 \end{cases}$$

MEV

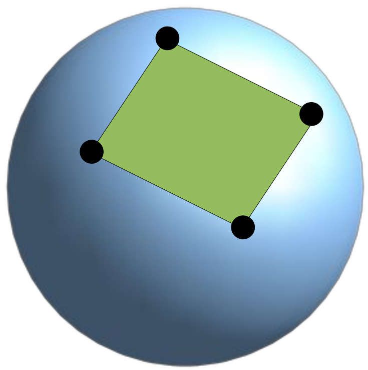
$$\begin{cases} v=2 \\ e=1 \\ f=1 \\ h=0 \\ r=0 \\ s=1 \end{cases}$$

2 x MEV



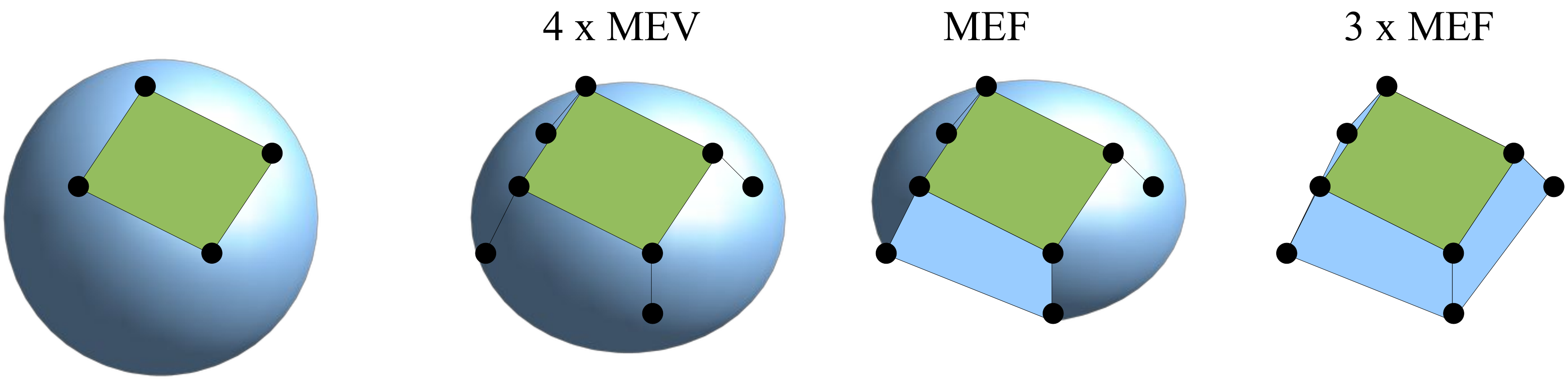
$$\begin{cases} v=4 \\ e=3 \\ f=1 \\ h=0 \\ r=0 \\ s=1 \end{cases}$$

MEF



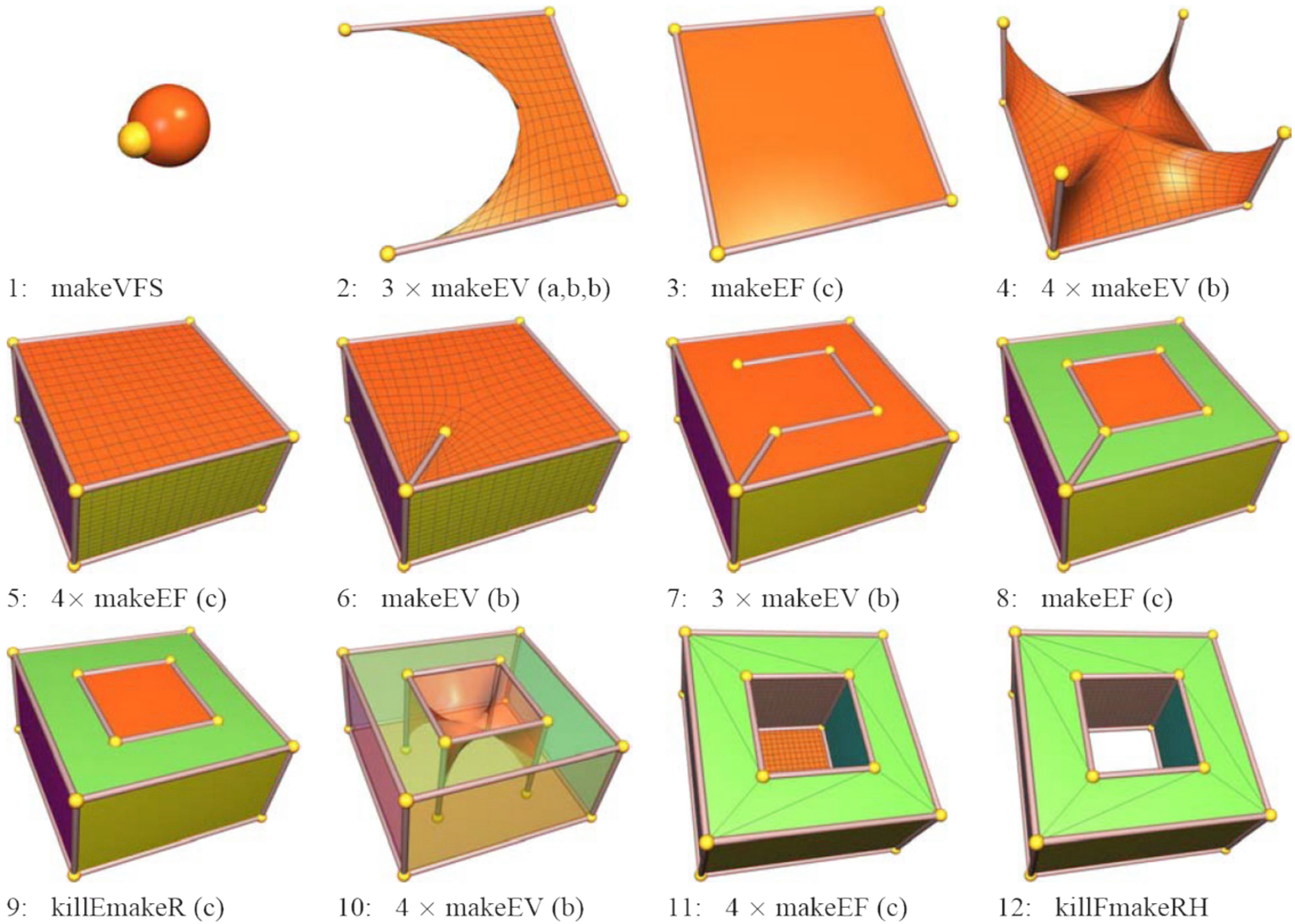
$$\begin{cases} v=4 \\ e=4 \\ f=2 \\ h=0 \\ r=0 \\ s=1 \end{cases}$$

- Example of use of Euler operators



$$\begin{cases} v=4 \\ e=4 \\ f=2 \\ h=0 \\ r=0 \\ s=1 \end{cases} \qquad \begin{cases} v=8 \\ e=8 \\ f=2 \\ h=0 \\ r=0 \\ s=1 \end{cases} \qquad \begin{cases} v=8 \\ e=9 \\ f=3 \\ h=0 \\ r=0 \\ s=1 \end{cases} \qquad \begin{cases} v=8 \\ e=12 \\ f=6 \\ h=0 \\ r=0 \\ s=1 \end{cases}$$

- Example of use of Euler operators



S. Havemann

- Those operators have a vectorial form in the basis of elementary entities

  v e f h r s
  (1, 1, 0, 0, 0, 0) – MEV, Make an Edge and a Vertex
  (0, 1, 1, 0, 0, 0) – MEF, Make a Face and an Edge
  (0,-1, 0, 0, 1, 0) – KEMR, Kill an Edge Make a Ring
  (1, 0, 1, 0, 0, 1) – MVFS, Make a Vertex, a Face and a Solid
  (0, 0,-1, 1, 1, 0) – KFMRH, Kill a Face, Make a Ring and a Hole

  - In order to have a complete basis of the configuration space, a vector orthogonal to the hyperplane of acceptable configurations must be added

  $$v-e+f-2s+2h-r=0$$

  - The coefficients of the equation of hyperplane are precisely the coordinates of the orthogonal vector...

  v e f h r s
  (1,-1, 1, 2,-1,-2) – Euler-Poincaré

Solid modelling

- Any transformation can thus be expressed easily using matrix operations
  - $\mathbf{A}$ is a basis of the topological configurations space
  - The columns of $\mathbf{A}$ are the variation of the number of entities for each operator, and the E-P relation.

Columns corresponding to each of the Euler operators

Column corresponding to Euler-Poincaré's relation

$$\mathbf{A}=\begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & -1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 1 & -1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 2 \\ 0 & 0 & 1 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 & 0 & -2 \end{pmatrix}$$

$$q=\mathbf{A}\cdot p$$

Vector representing the number of times that each operator is applied

Vector representing the number (or the variation of the number) of elementary entities

Solid modelling

- **A** is composed of linearly independent vectors, thus one can get the inverse...

$$q = \mathbf{A} \cdot p$$
$$\mathbf{A}^{-1} \cdot q = \mathbf{A}^{-1} \cdot \mathbf{A} \cdot p$$
$$p = \mathbf{A}^{-1} \cdot q$$

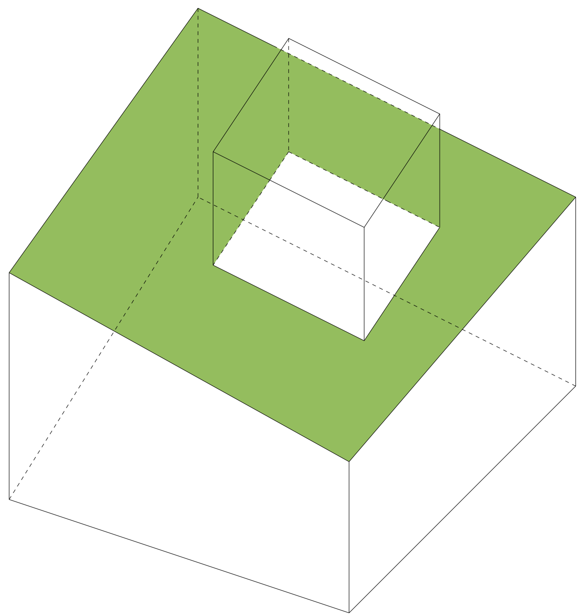
Vector representing the number of times each operator is applied
They are the **Euler Coordinates**

Vector representing the number of elementary entities

$$\mathbf{A}^{-1} = \frac{1}{12} \begin{pmatrix} 9 & 3 & -3 & -6 & 3 & -6 \\ -5 & 5 & 7 & 2 & 5 & -2 \\ 3 & -3 & 3 & -6 & 9 & -6 \\ 2 & -2 & 2 & 4 & -2 & 8 \\ -2 & 2 & -2 & 8 & 2 & 4 \\ 1 & -1 & 1 & 2 & -1 & -2 \end{pmatrix}$$

A Vector that is orthogonal to the « hyperplane »...

- Determination of elementary operations ...



$$q=\begin{cases} v=16 \\ e=24 \\ f=11 \\ h=0 \\ r=1 \\ s=1 \end{cases}=(16,24,11,0,1,1)$$

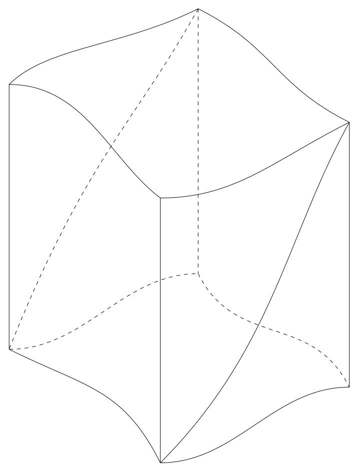
$$p=\mathbf{A}^{-1}\cdot q$$

$$p=(15,10,1,1,0,0)^{\mathbf{T}}$$

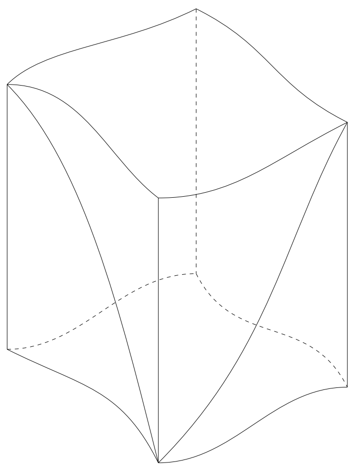
15 x MEV, Make an Edge and a Vertex
10 x MEF, Make a Face and an Edge
1 x MVFS, Make a Vertex, a Face and a Solid
1 x KEMR, Kill an Edge Make a Ring
0 x KFMRH, Kill a Face, Make a Ring and a Hole

The vector q respects the Euler-Poincaré relation

- Are Euler coordinates sufficient to define the topology of a solid ? → No.



1 x MEF, Make an Edge and a Face
1 x KEF, Kill an Edge and a Face



7 x MEV, Make an Edge and a Vertex
7 x MEF, Make an Edge and a Face
1 x MVFS, Make a Vertex, a Face and a Solid
0 x KEMR, Kill an Edge Make a Ring
0 x KFMRH, Kill a Face, Make a Ring and a Hole

7 x MEV
7 x MEF
1 x MVFS
0 x KEMR
0 x KFMRH

Identical Euler coordinates

- Each Euler operator takes a certain number of parameters, in principle the entities to destroy/ or replace, and data necessary to creation of new entities.
- These depend on the structure of data used to represent the B-Rep object
- Application of an Euler operator is not always possible, the entities involved must exist and respect some conditions

  KEF for example may only be applied on an edge separating two distinct faces... if not, one does not remove any face from the model !

- Conditions of application of Euler operators

MEV, Make an Edge and a Vertex	No specific conditions except existence of a solid
KEV, Kill an Edge and a Vertex	The edge has two distinct vertices
MEF, Make an Edge and an Face	Vertices belonging to the same boundary loop of one face
KEF, Kill an Edge and a Face	Distinct faces located on both sides of the edge
MEKR, Make an Edge and Kill a Ring	Vertices belong to distinct boundary loops of the same face
KEMR, Kill an Edge and Make a Ring	Same face located on both sides of the edge, which is not part of a ring
MVFS, Make a Vextex, a Face and a Shell	Empty space
KVFS, Kill a Vertex, a Face and a Shell	The shell (solid) has no edges and has only one vertex (elementary volume)
KFMRH, Kill Face Make a Ring and a Hole	The face cannot hold any ring
MFKRH, Make a Face, Kill a Ring and a Hole	May be only applied to a ring

- Some examples of the application of Euler operators (not shown here)
  - Extrusion of a face
  - Junction of two solids
  - Cutting out a solid by a plane
  - Boolean operations between solids

- Bibliographic note

M. Mäntylä, An Introduction to Solid Modelling, Computer Science Press, 1988

I. Stroud, Solid Modelling and CAD Systems : How to Survive a CAD System, Springer, 2011(available on-line from the university campus)