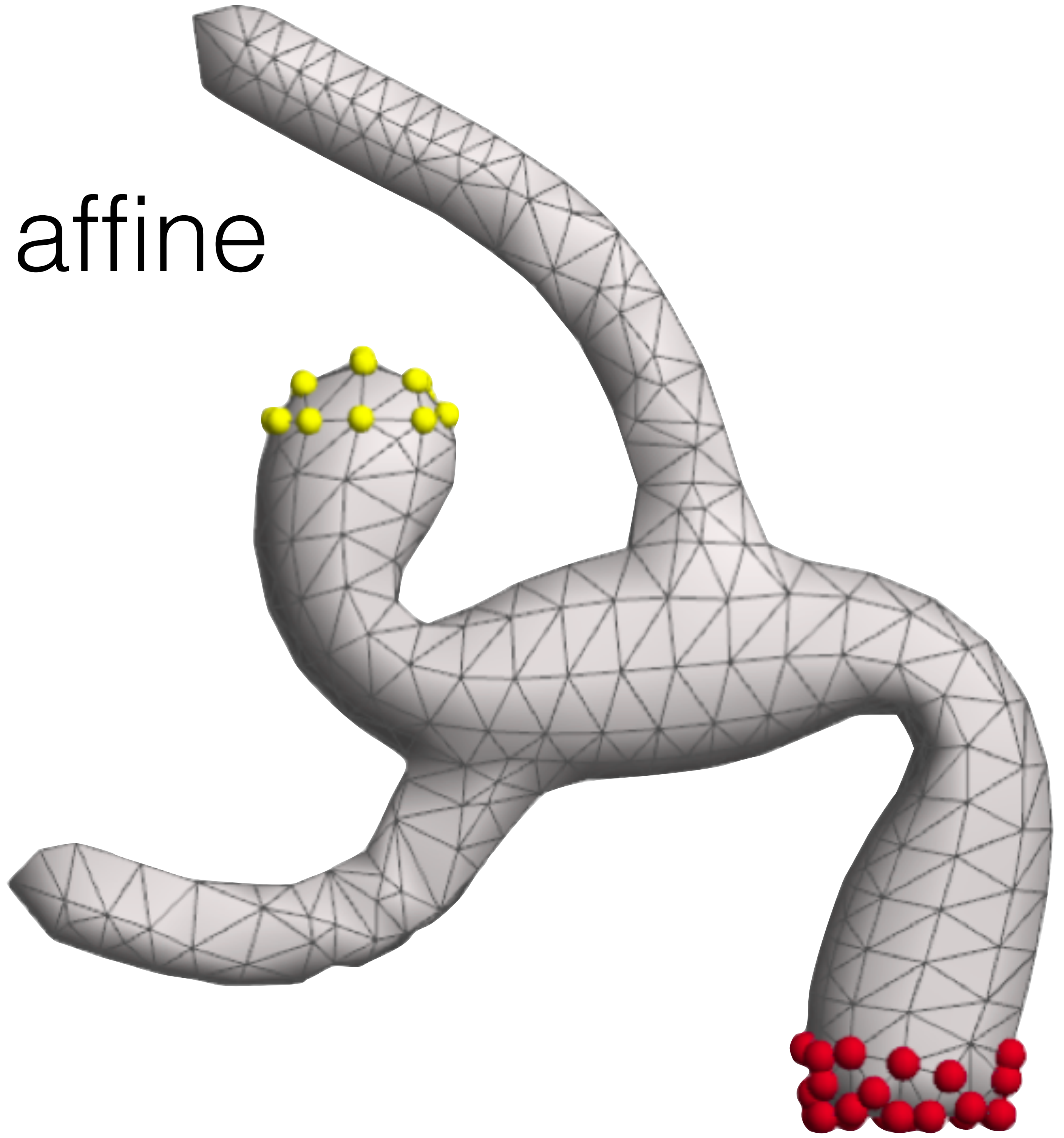
14 - Designing Surfaces

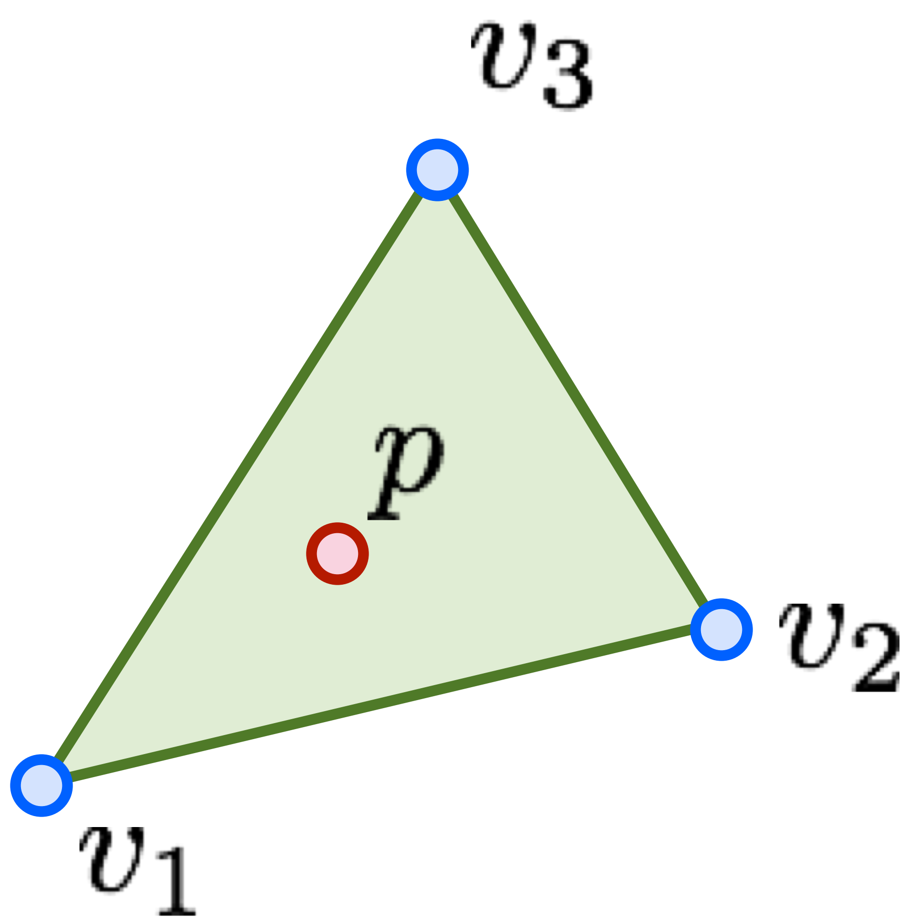
Triangular Surfaces

- A surface can be discretized by a collection of points and triangles
- Each triangle is a subset of a plane
- Every point on the surface can be expressed as an affine combination of three vertices
- The surface is C^0



Barycentric Coordinates

- The points inside a triangle, are usually parametrized using barycentric coordinates **w**:



$$\begin{bmatrix} v_1^x & v_2^x & v_3^x \\ v_1^y & v_2^y & v_3^y \\ 1 & 1 & 1 \end{bmatrix} \bullet \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} = \begin{bmatrix} p_x \\ p_y \\ 1 \end{bmatrix}$$

- The barycentric coordinates naturally defines a **parametrization**

Tensor Surfaces

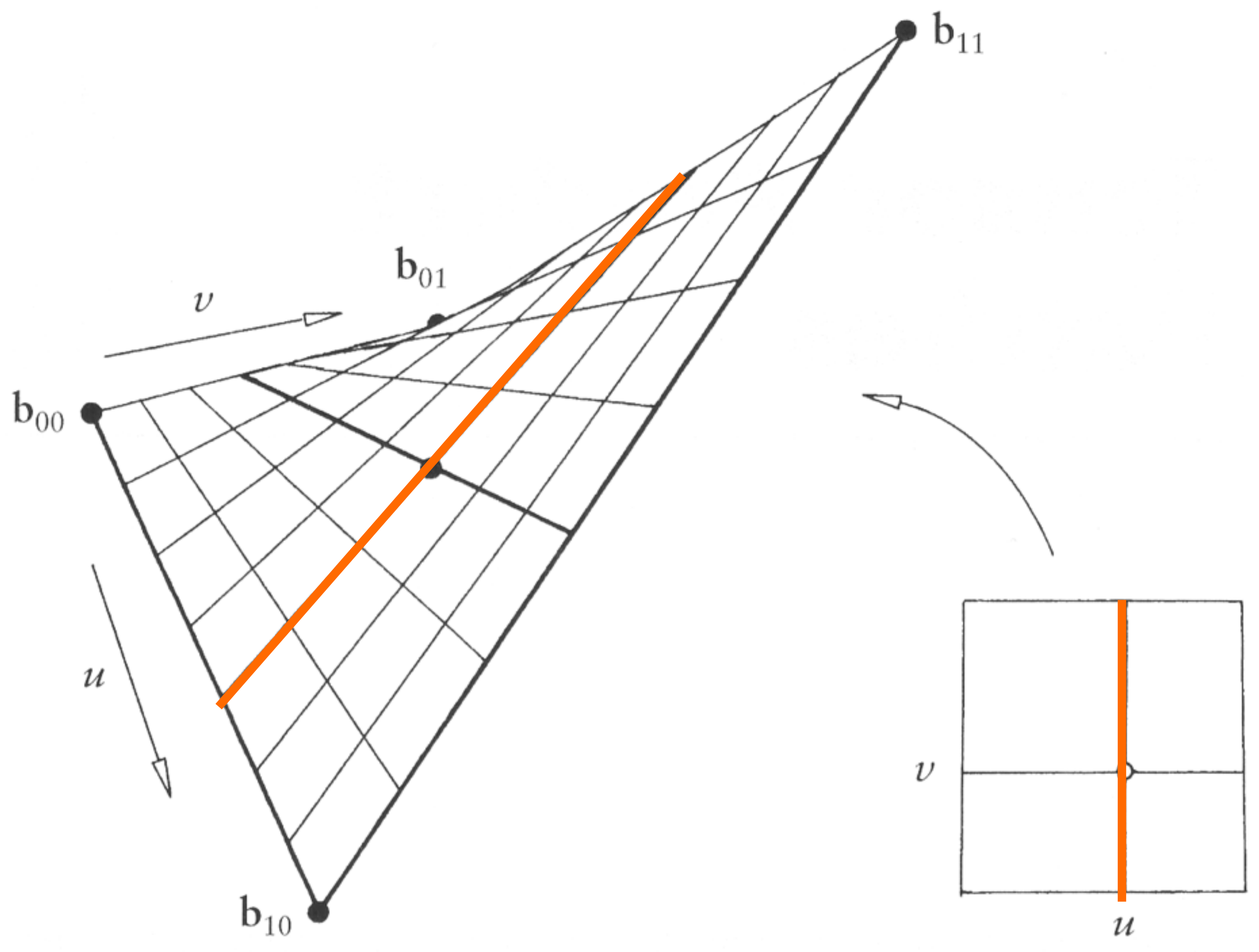
Bilinear Interpolation

- Linear interpolation
  - "Simplest" curve between two points
- Bilinear interpolation
  - "Simplest" surface between four points

$$x(u,v) = \sum_{i=0}^{1}\sum_{j=0}^{1} b_{i,j} B_i^1(u) B_j^1(v)$$

$$= \begin{pmatrix} 1-u & u \end{pmatrix} \begin{pmatrix} b_{00} & b_{01} \\ b_{10} & b_{11} \end{pmatrix} \begin{pmatrix} 1-v \\ v \end{pmatrix}$$

Isoparametric curve

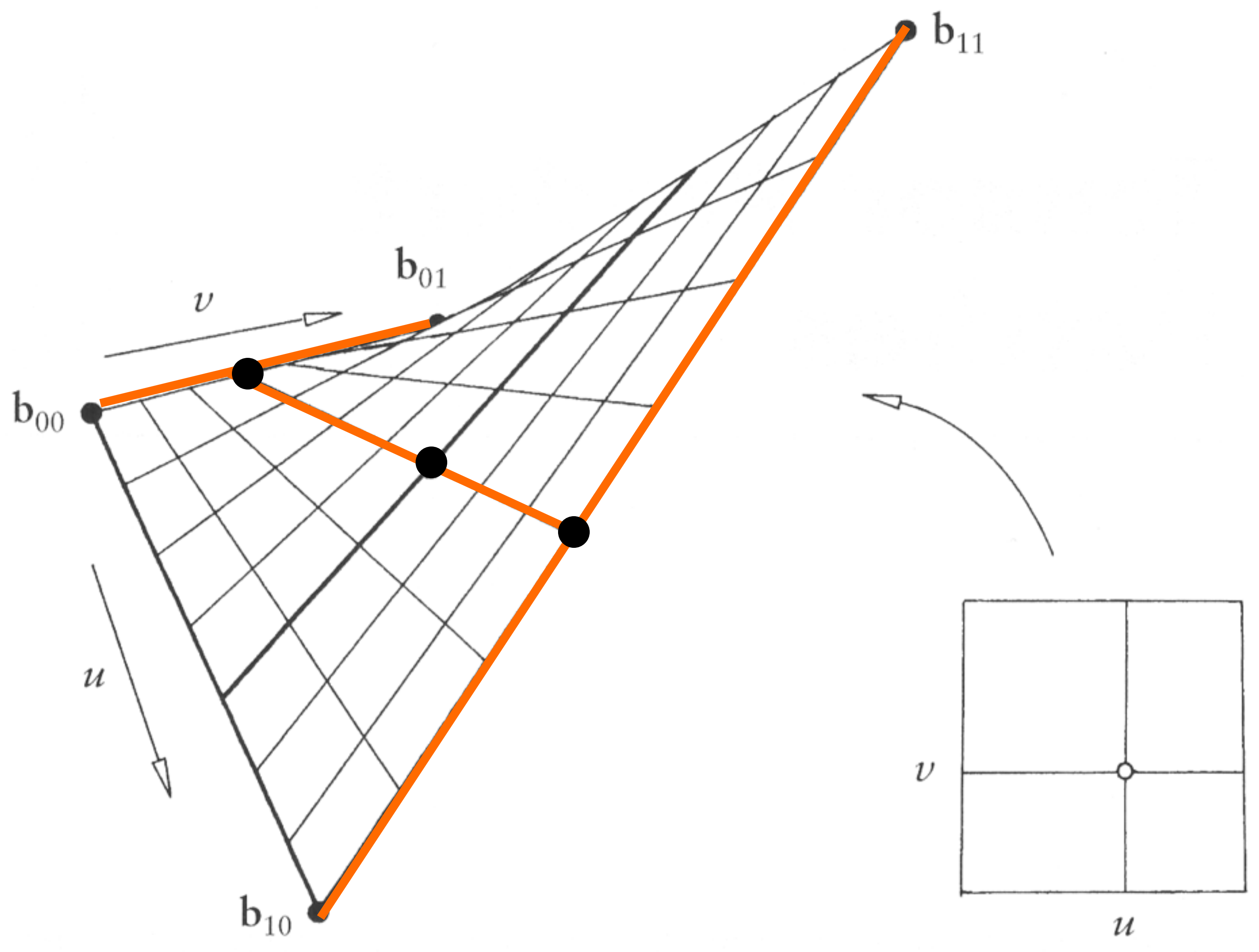


Domain

Bilinear Interpolation

- Linear interpolation
  - "Simplest" curve between two points
- Bilinear interpolation
  - "Simplest" surface between four points

$$b_{00}^{01} = (1-v)b_{00} + vb_{01}$$
$$b_{10}^{01} = (1-v)b_{10} + vb_{11}$$
$$b_{11}^{00} = (1-u)b_{00}^{01} + ub_{10}^{01}$$



Bézier Patches

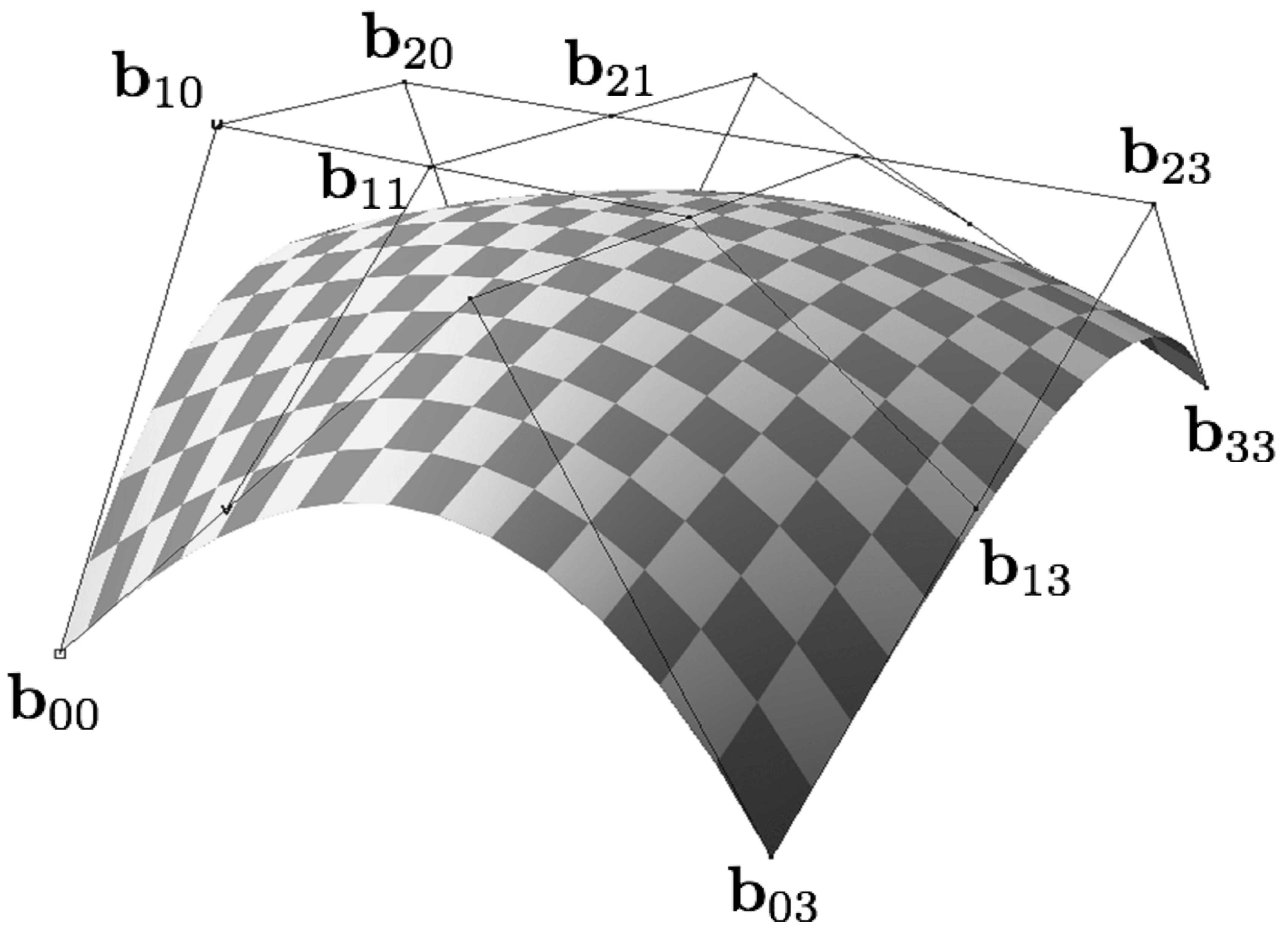
- Build on Bézier curves

$$b^m(u) = \sum_{i=0}^{m} b_i B_i^m(u)$$

- Control points as curves

$$b_i = b_i(v) = \sum_{j=0}^{n} b_{ij} B_j^n(v)$$

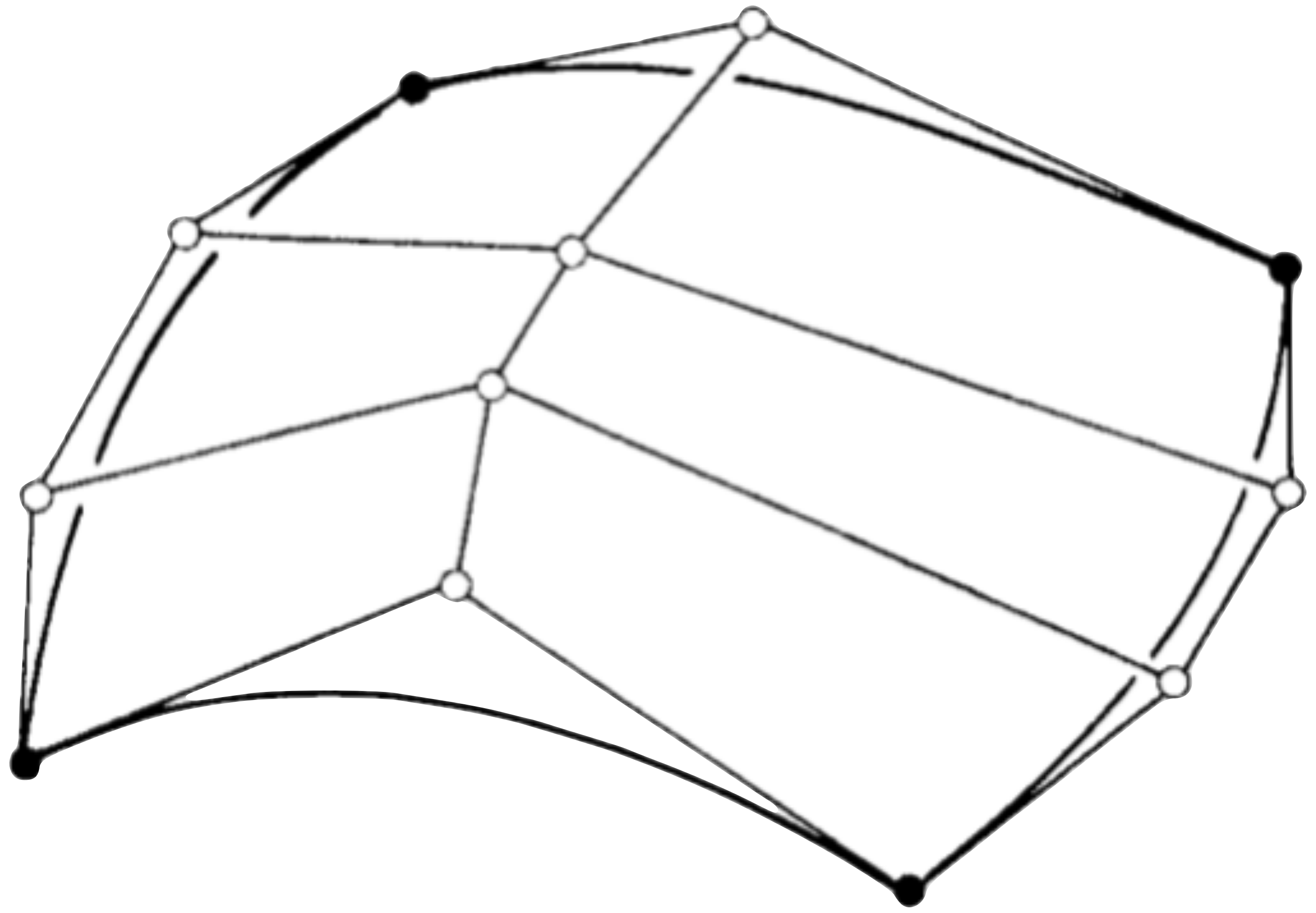
$$b^{mn}(u, v) = \sum_{i=0}^{m} \sum_{j=0}^{n} b_{ij} B_i^m(u) B_j^n(v)$$



- Keep one parameter fixed: iso-parameter curves

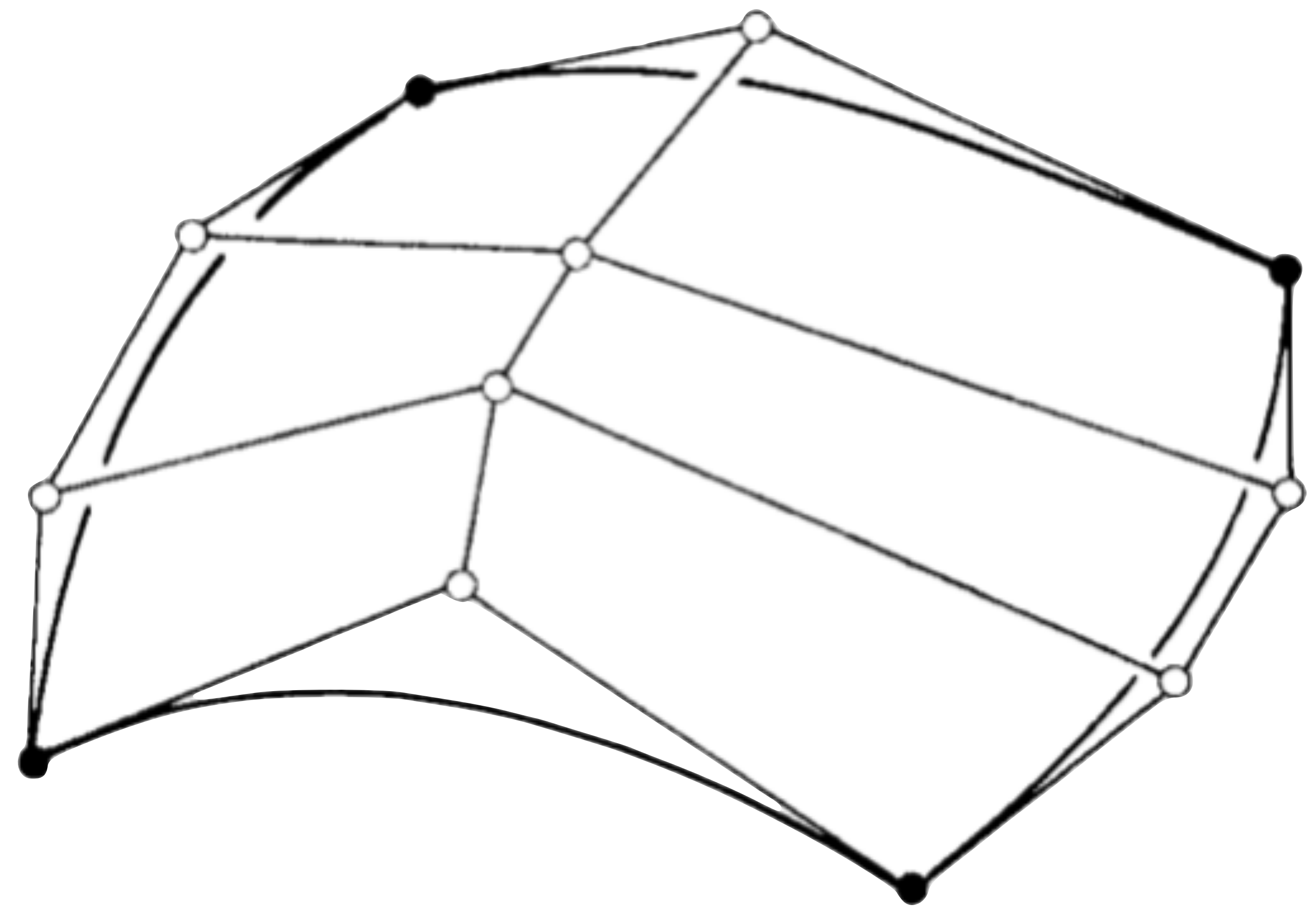
Properties

- Affine invariance



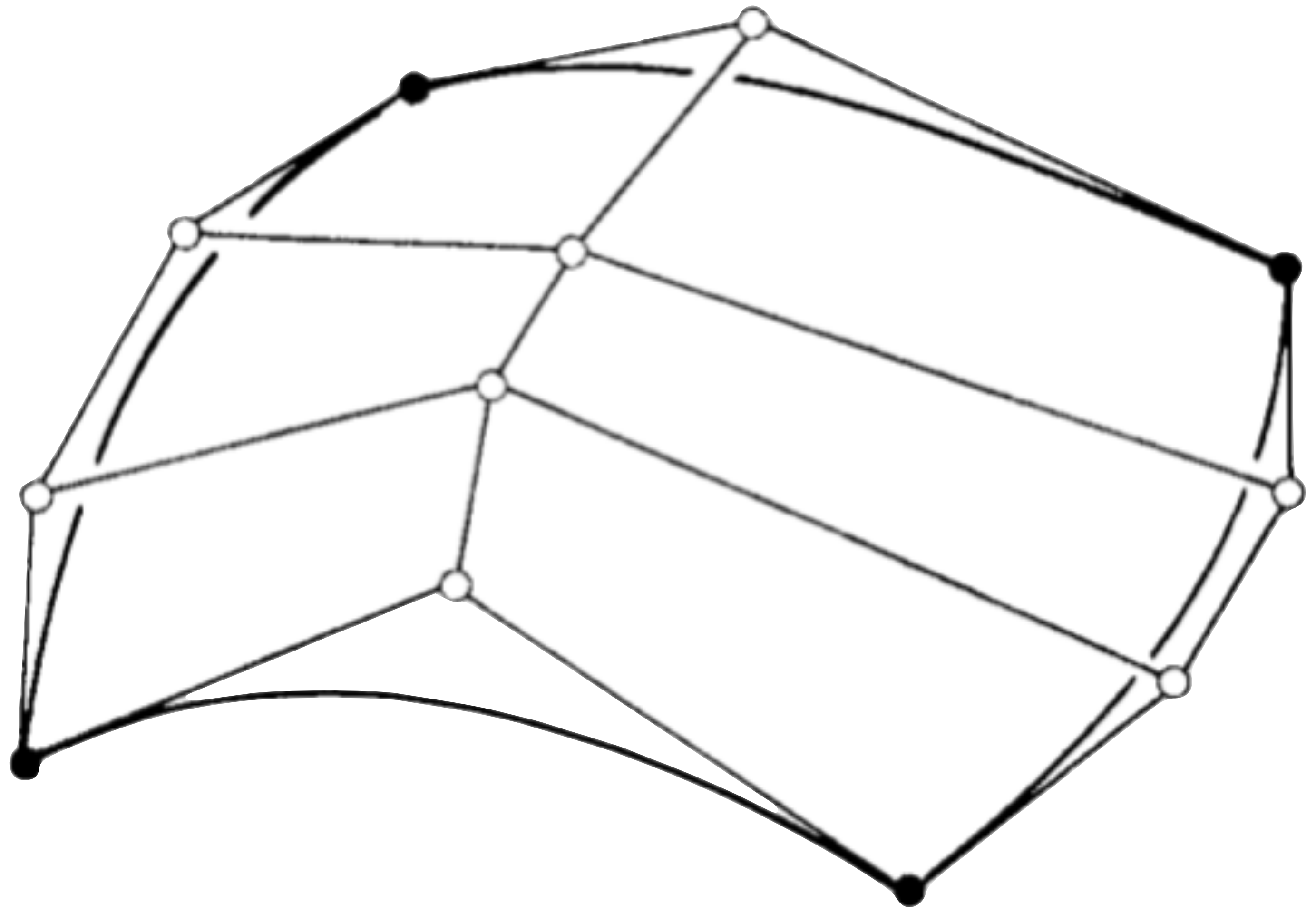
Properties

- Affine invariance
    - Repeated (bi-)linear combinations
- Convex hull property



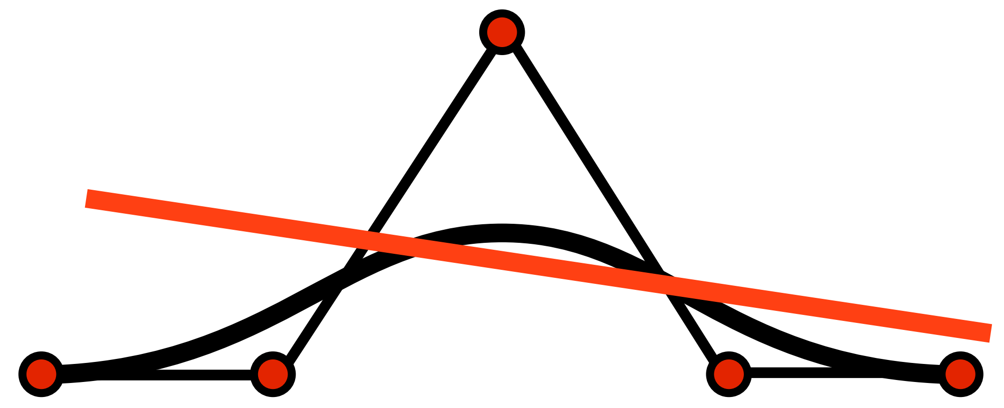
Properties

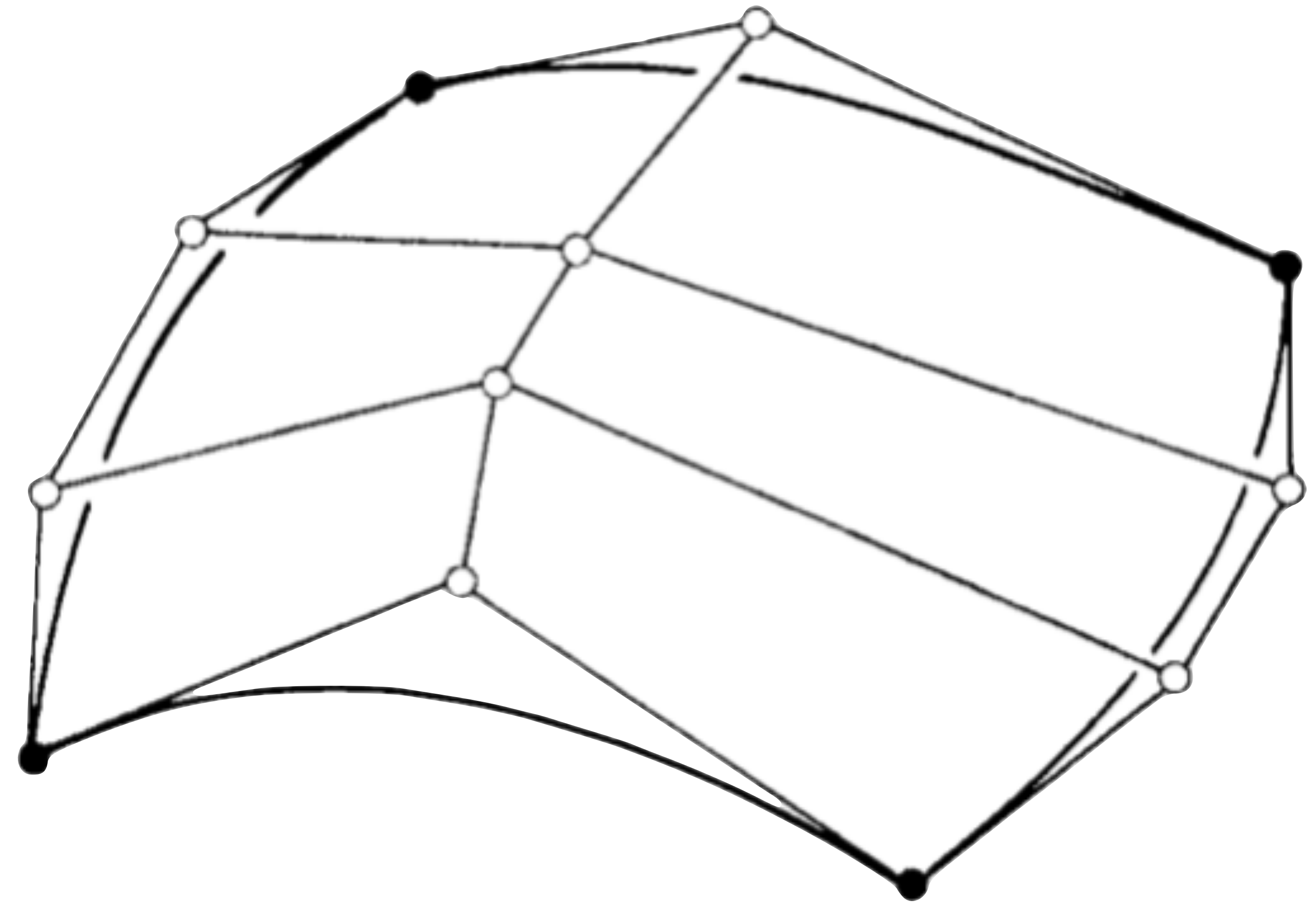
- Affine invariance
  - Repeated (bi-)linear combinations
- Convex hull property
  - Partition of unity and non-negativity
- Polynomial boundary curves and corner interpolation



Properties

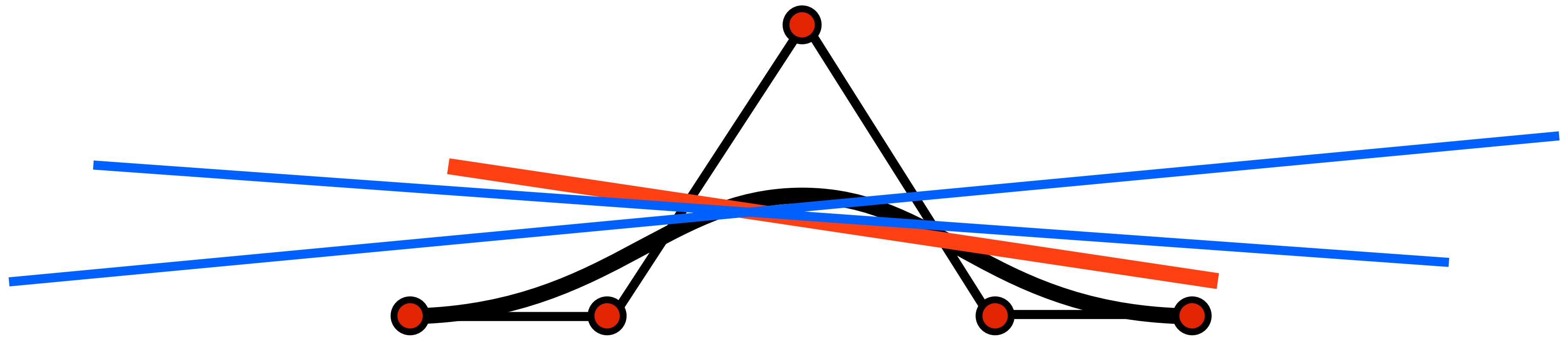
- Affine invariance
  - Repeated (bi-)linear combinations
- Convex hull property
  - Partition of unity and non-negativity
- Polynomial boundary curves and corner interpolation
- Variation diminishing

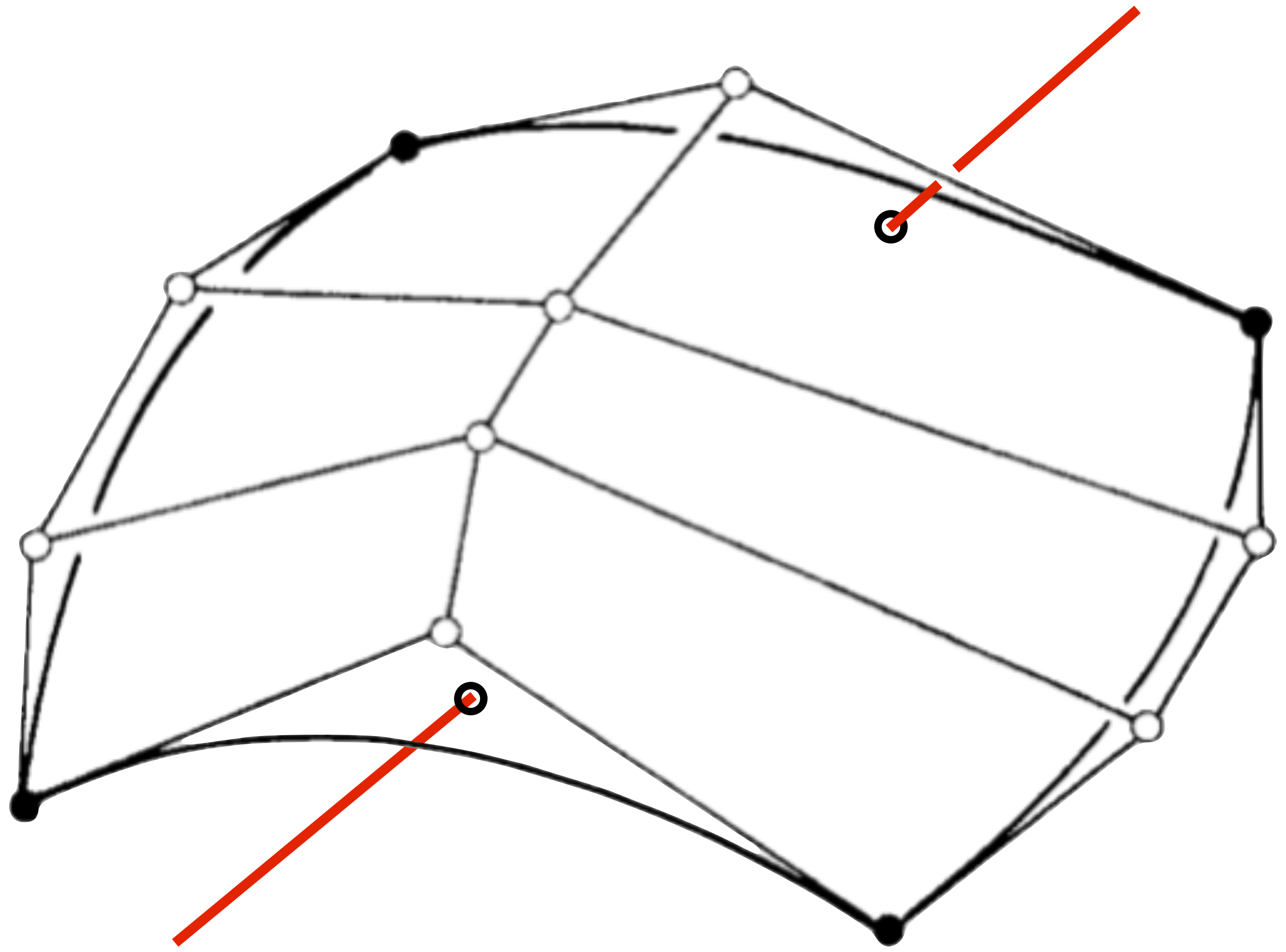




Properties

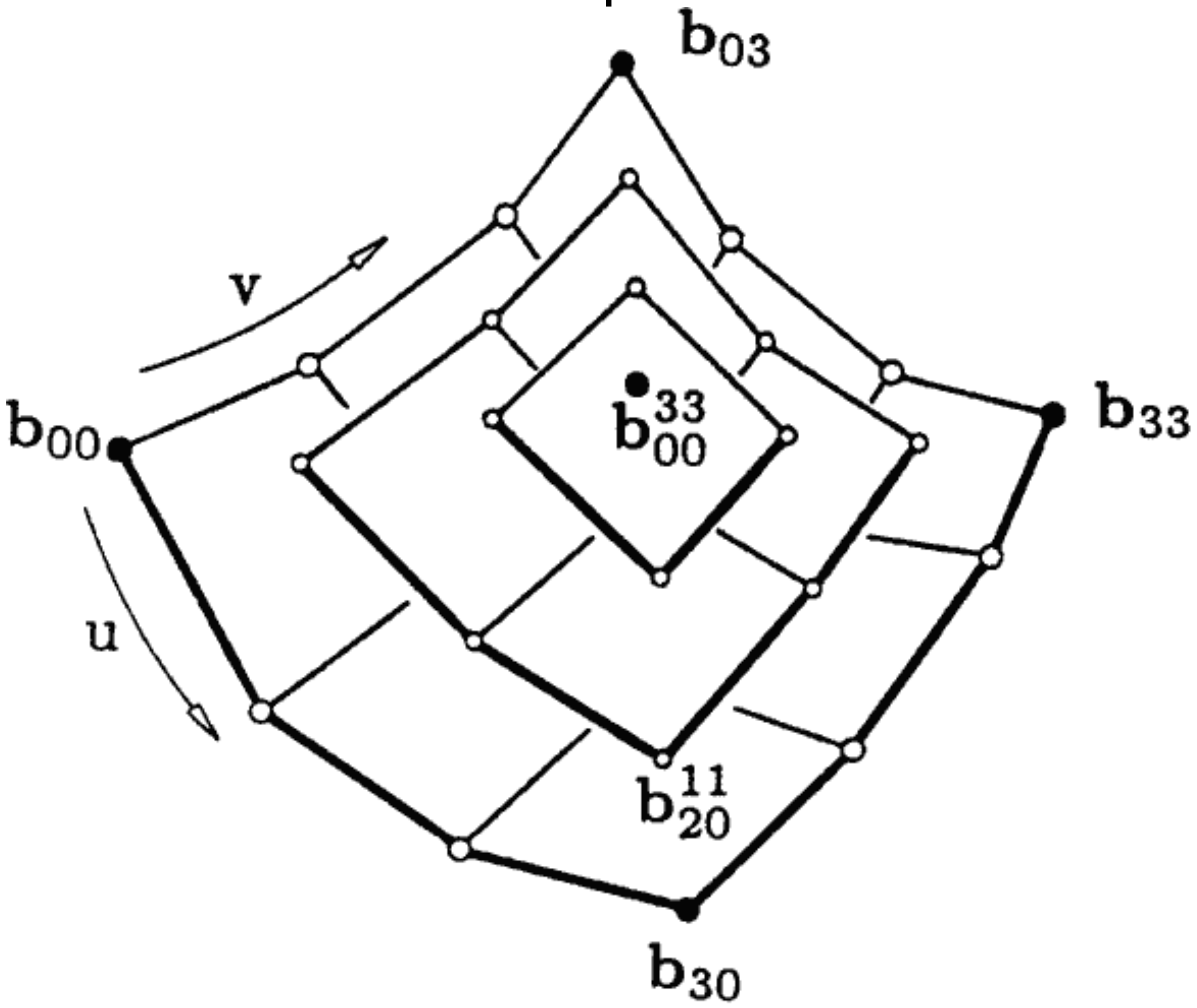
- Affine invariance
  - Repeated (bi-)linear combinations
- Convex hull property
  - Partition of unity and non-negativity
- Polynomial boundary curves and corner interpolation
- ~~Variation diminishing~~

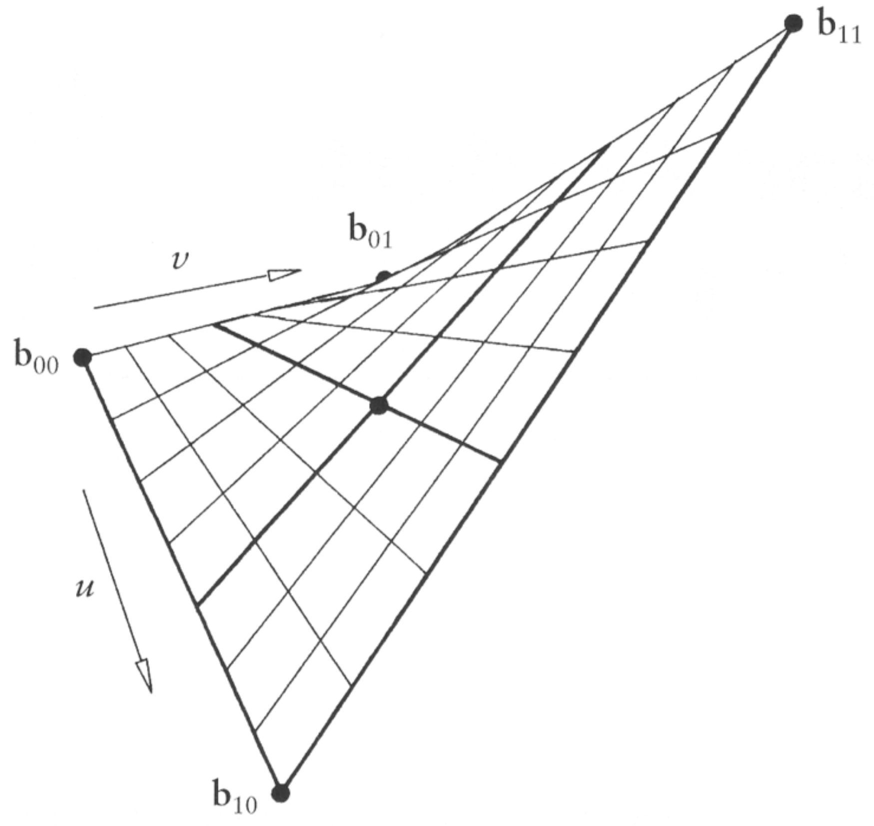




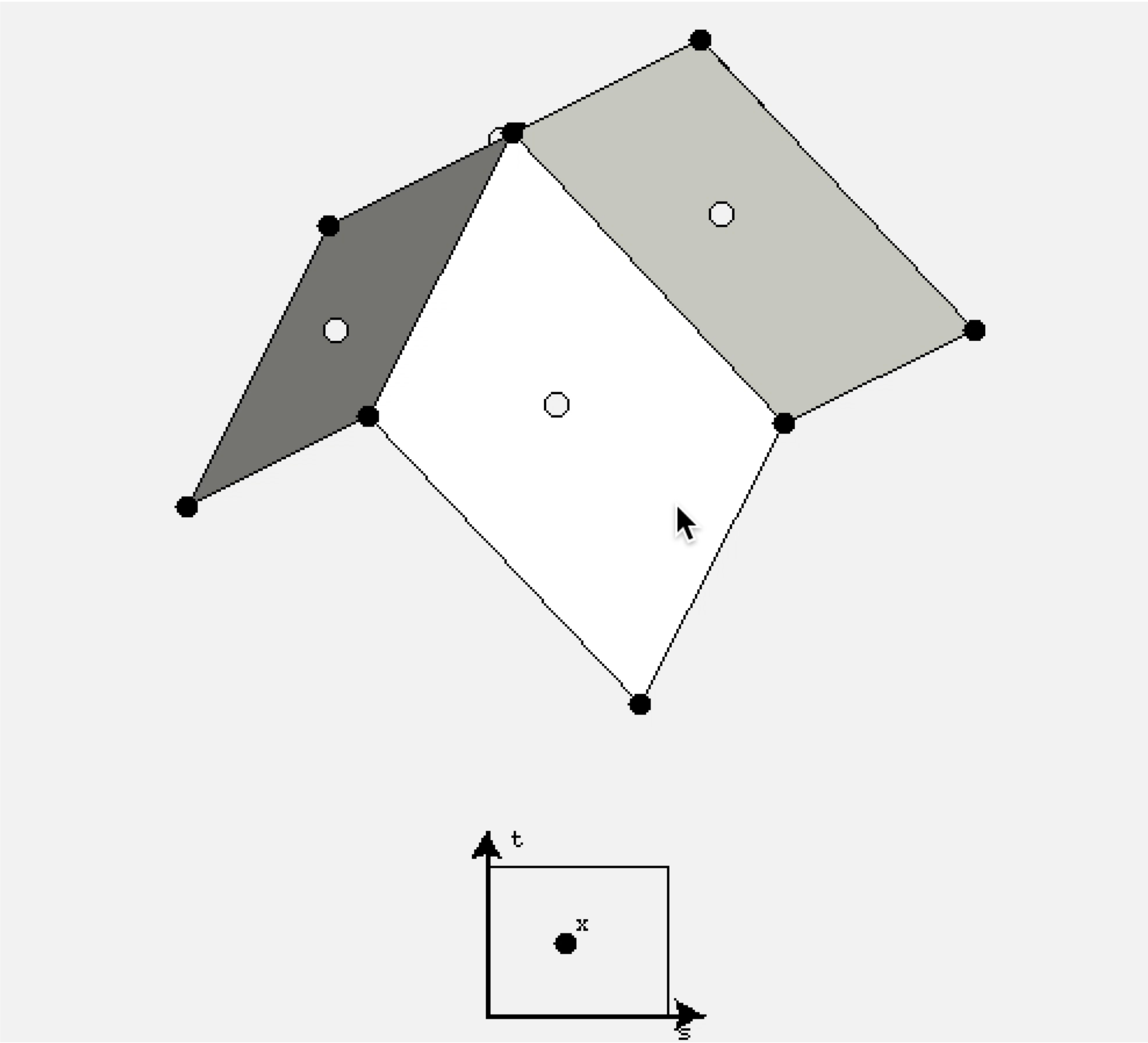
De Casteljau Algorithm

- Bézier curves created by repeated linear interpolation
- Surfaces: repeated bilinear interpolation





Demo



http://cagd-applets.webarchiv.kit.edu/mocca/html/noplugin/inhalt.html

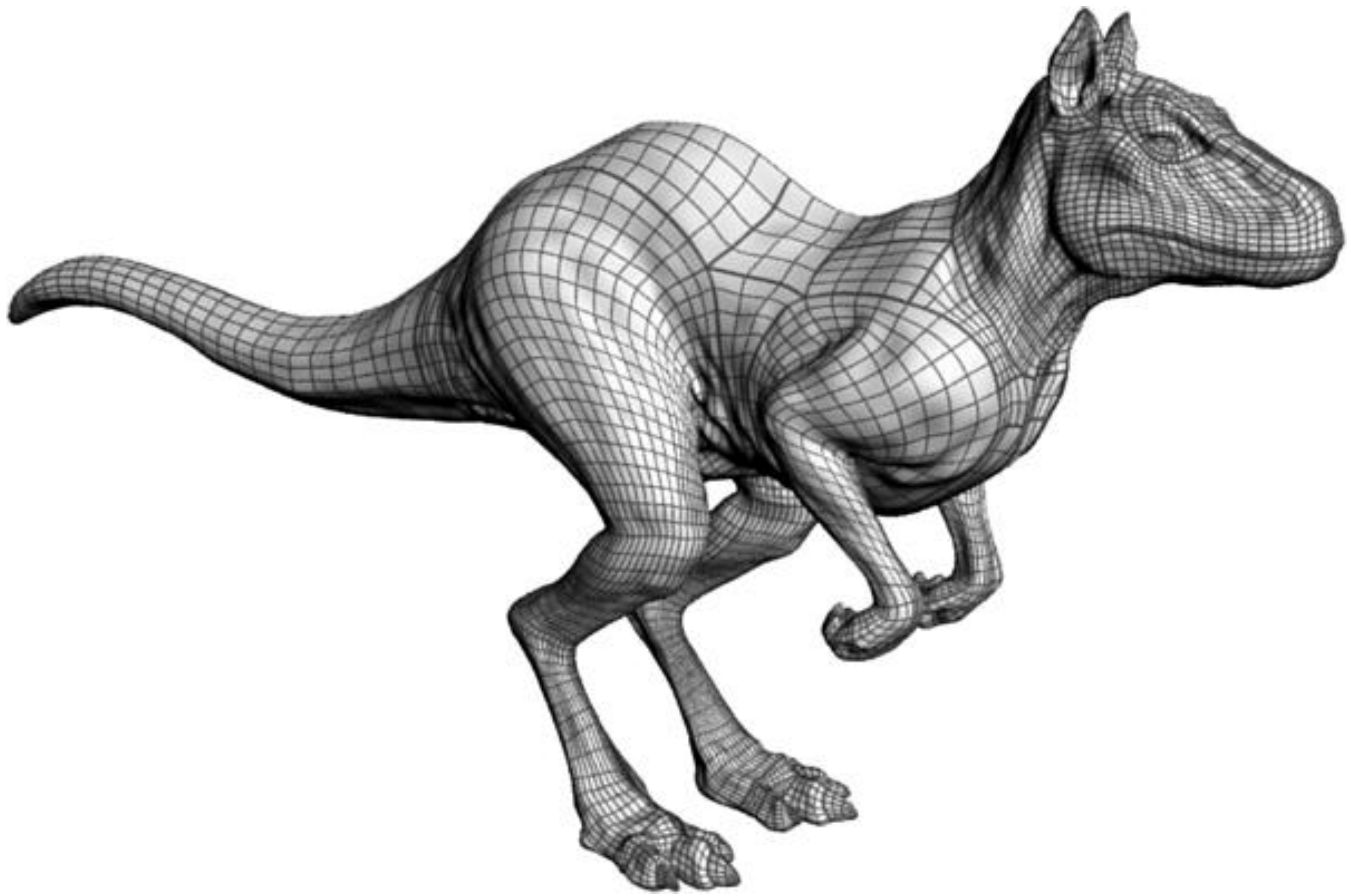
NURBS Surfaces

$$\mathbf{x}(u,v) = \frac{\sum_i \sum_j w_{i,j} \mathbf{d}_{i,j} N_i^m(u) N_j^n(v)}{\sum_i \sum_j w_{i,j} N_i^m(u) N_j^n(v)}$$

- Standard in most advanced modeling systems

NURBS Surfaces

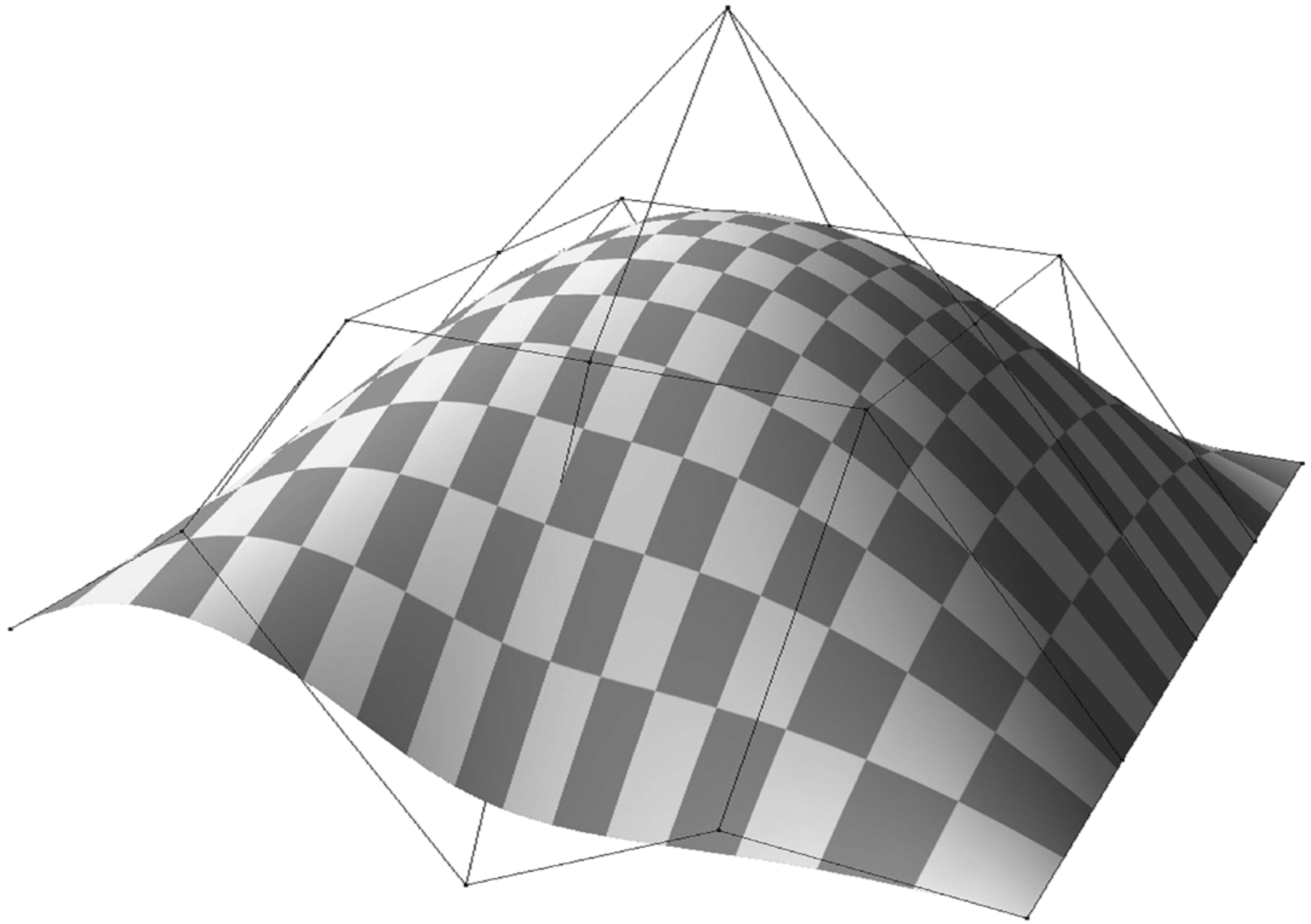
- Standard in most advanced modeling systems



NURBS Surfaces

- Influence of weights

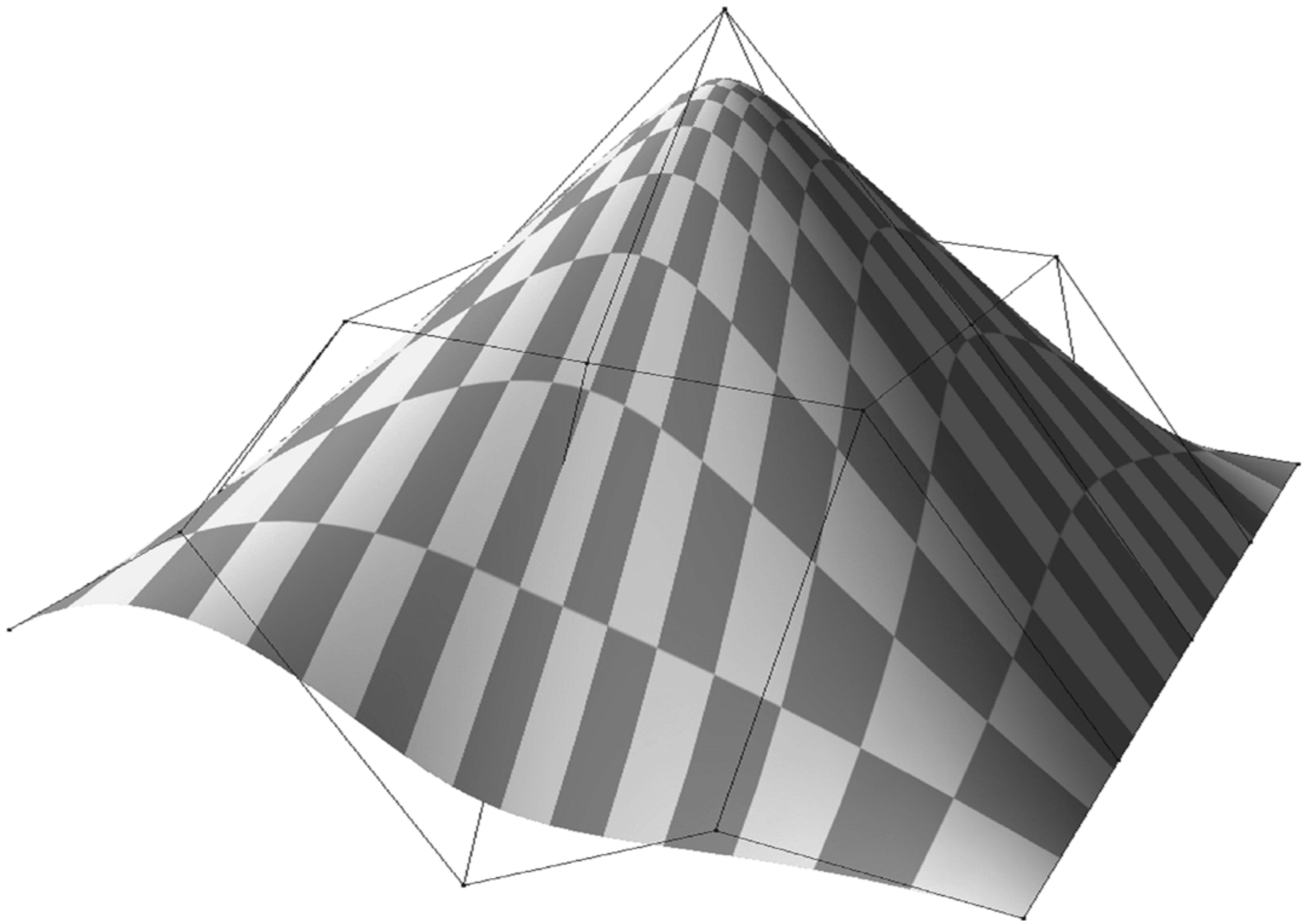
$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$



NURBS Surfaces

- Influence of weights

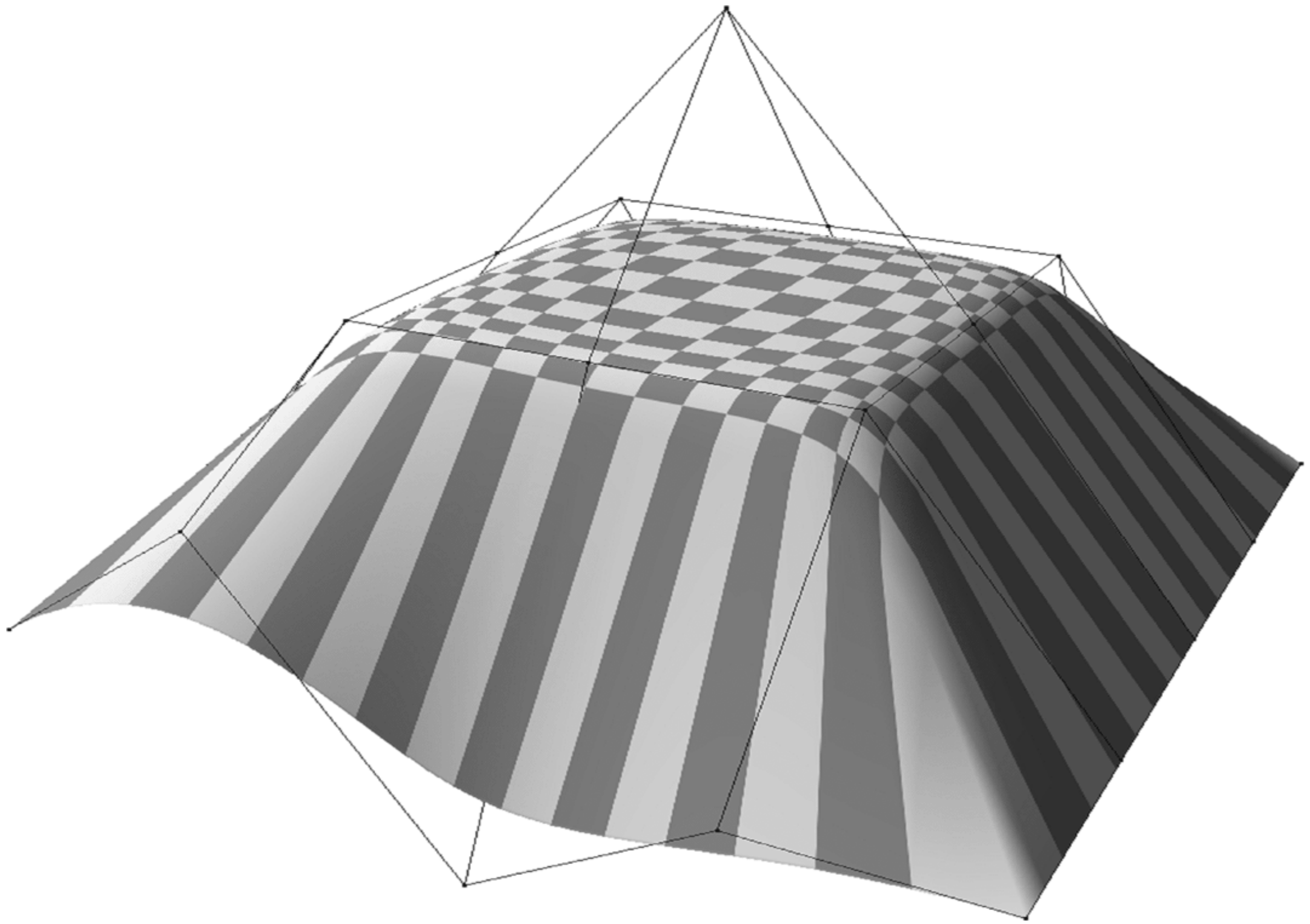
$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 10 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$



NURBS Surfaces

- Influence of weights

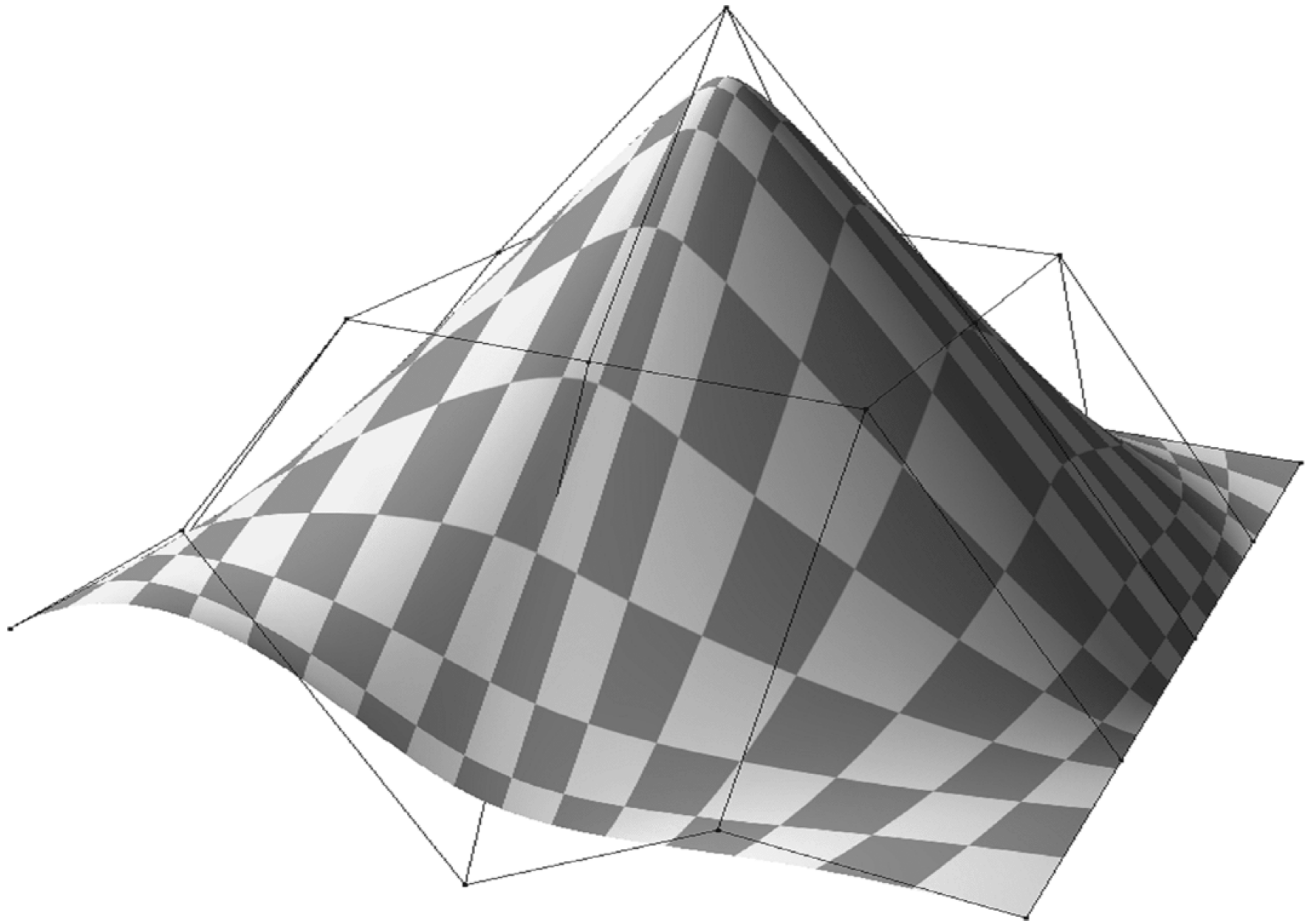
$$
W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 30 & 30 & 30 & 1 \\ 1 & 30 & 1 & 30 & 1 \\ 1 & 30 & 30 & 30 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}
$$



NURBS Surfaces

- Influence of weights

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 0.1 & 0.1 & 0.1 & 1 \\ 1 & 0.1 & 1 & 0.1 & 1 \\ 1 & 0.1 & 0.1 & 0.1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

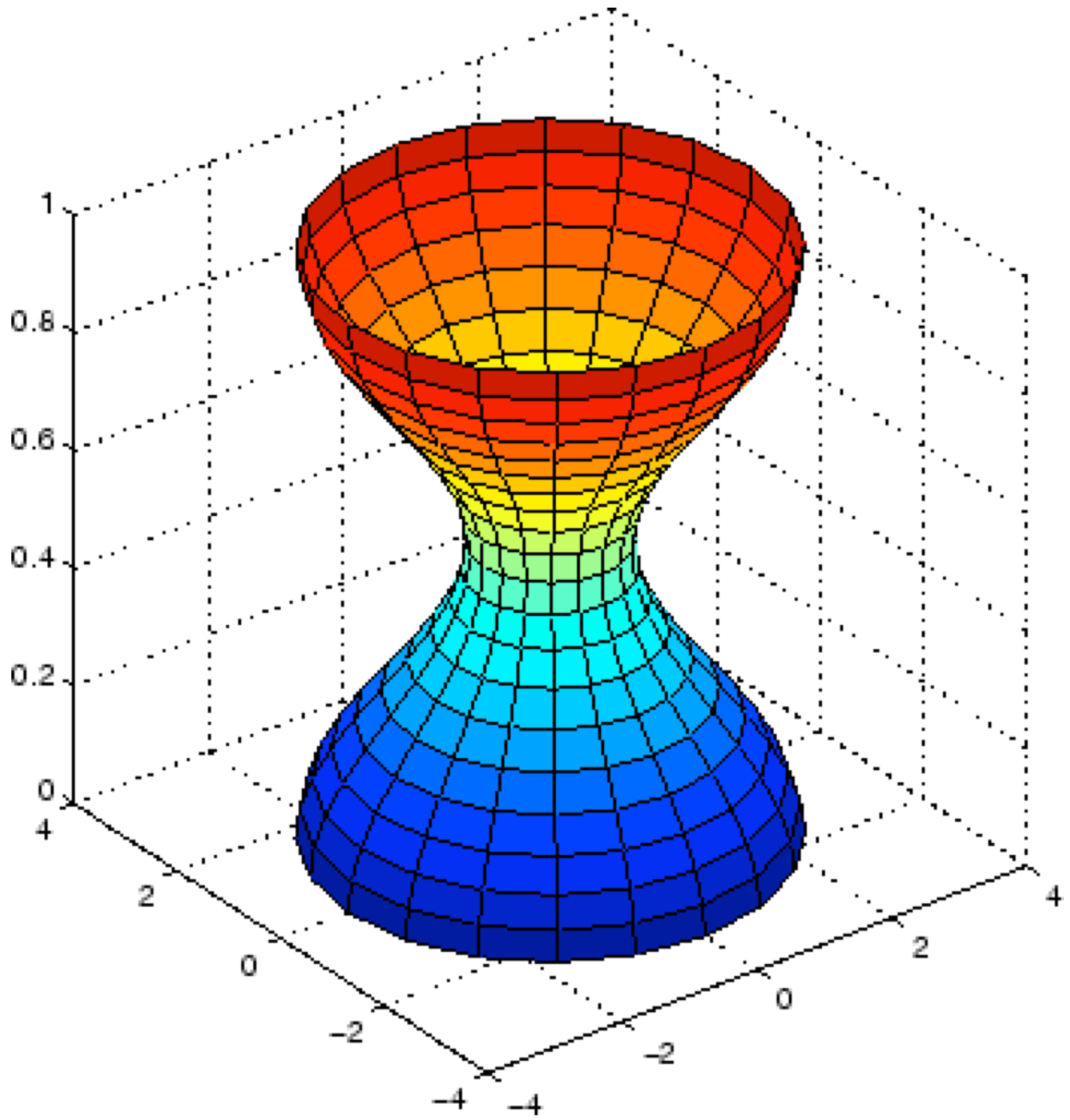


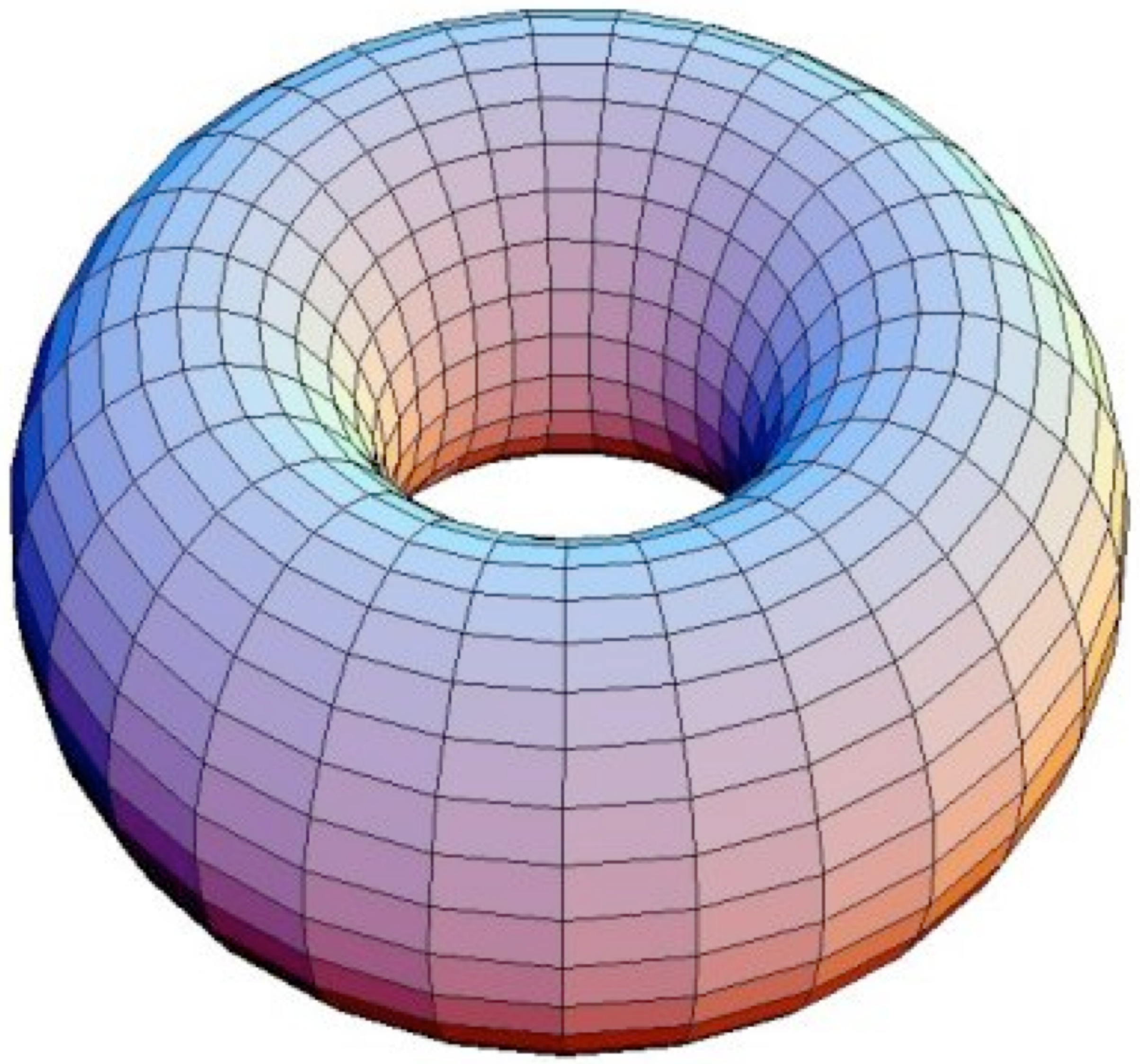
Subdivision Surfaces

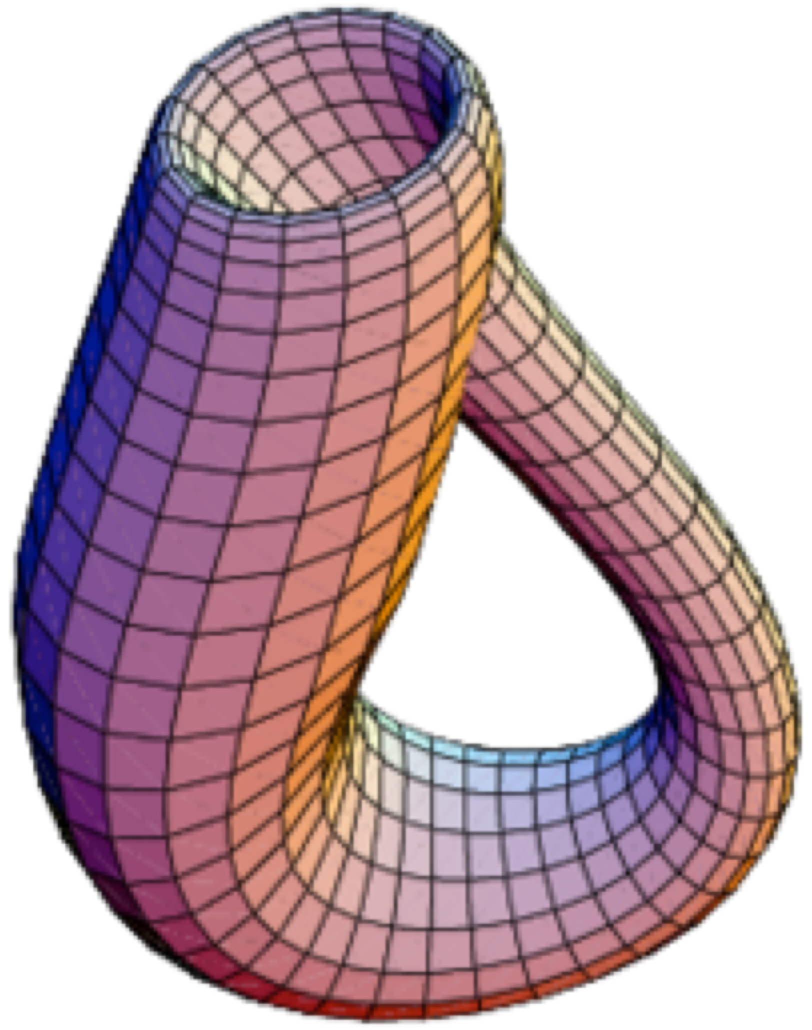
Why?

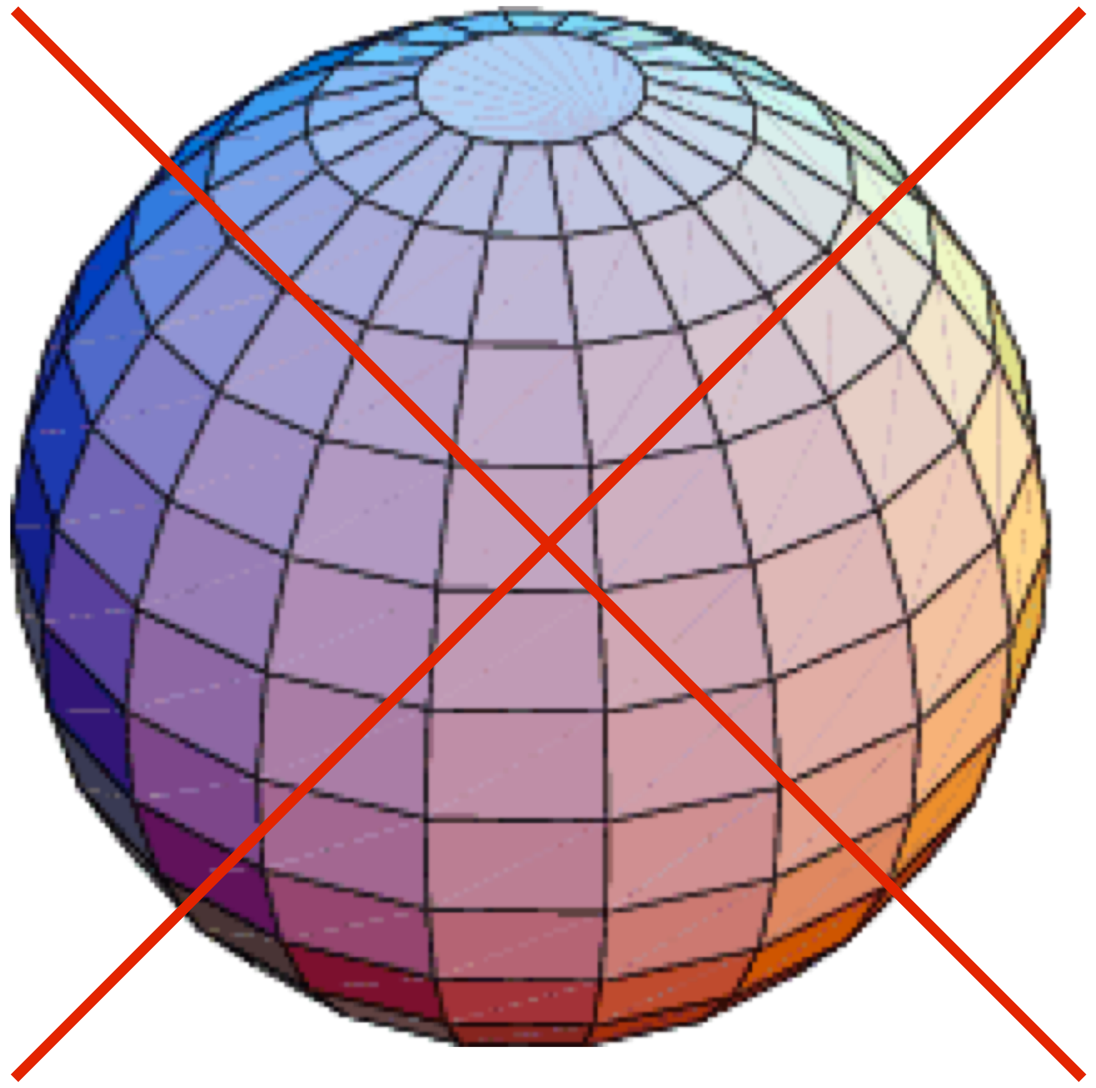
- Tensor product surfaces are defined on “regular grids”

- They cannot be defined on arbitrary meshes

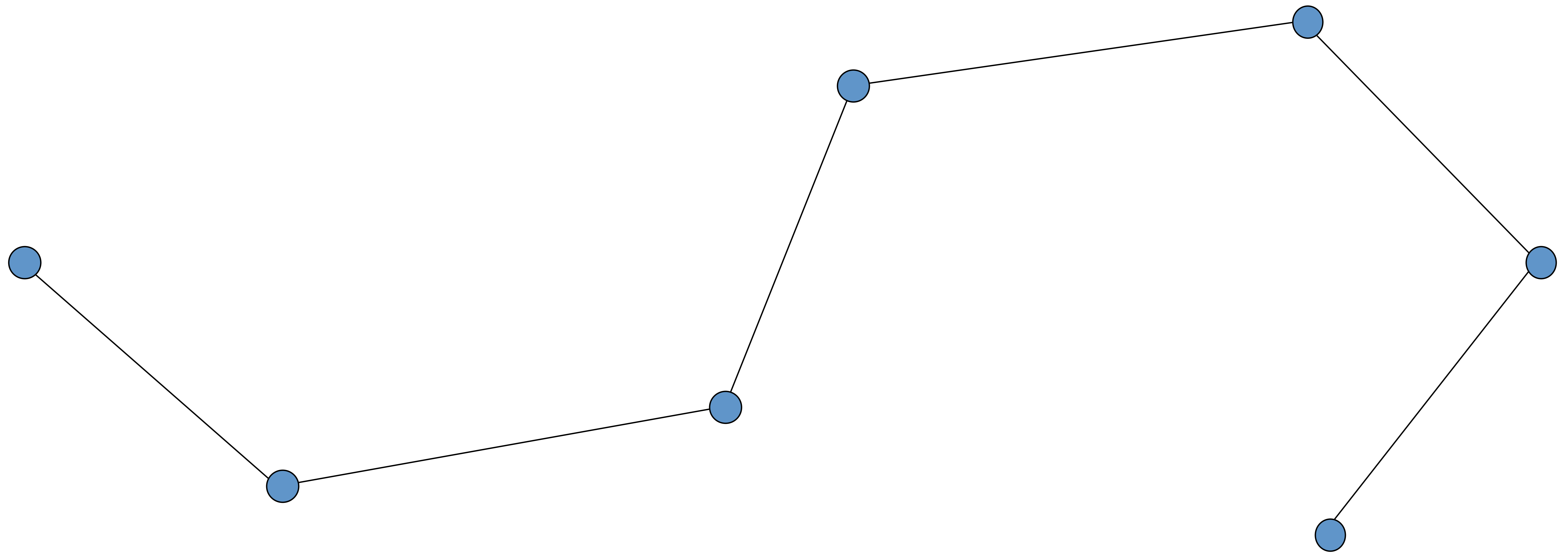


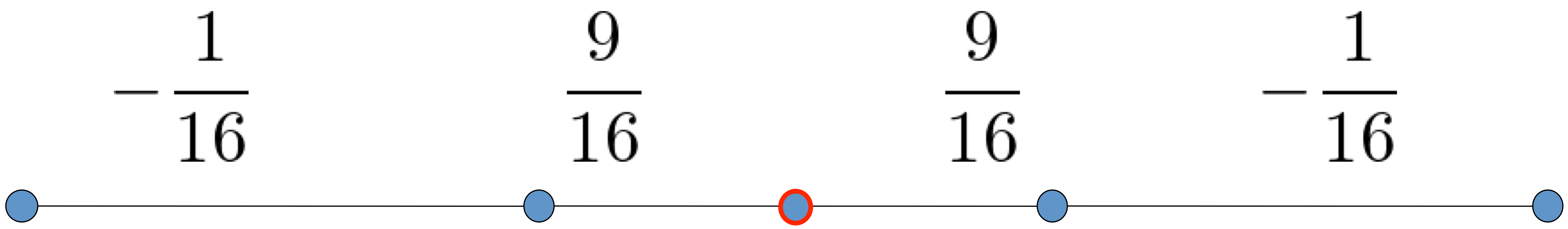




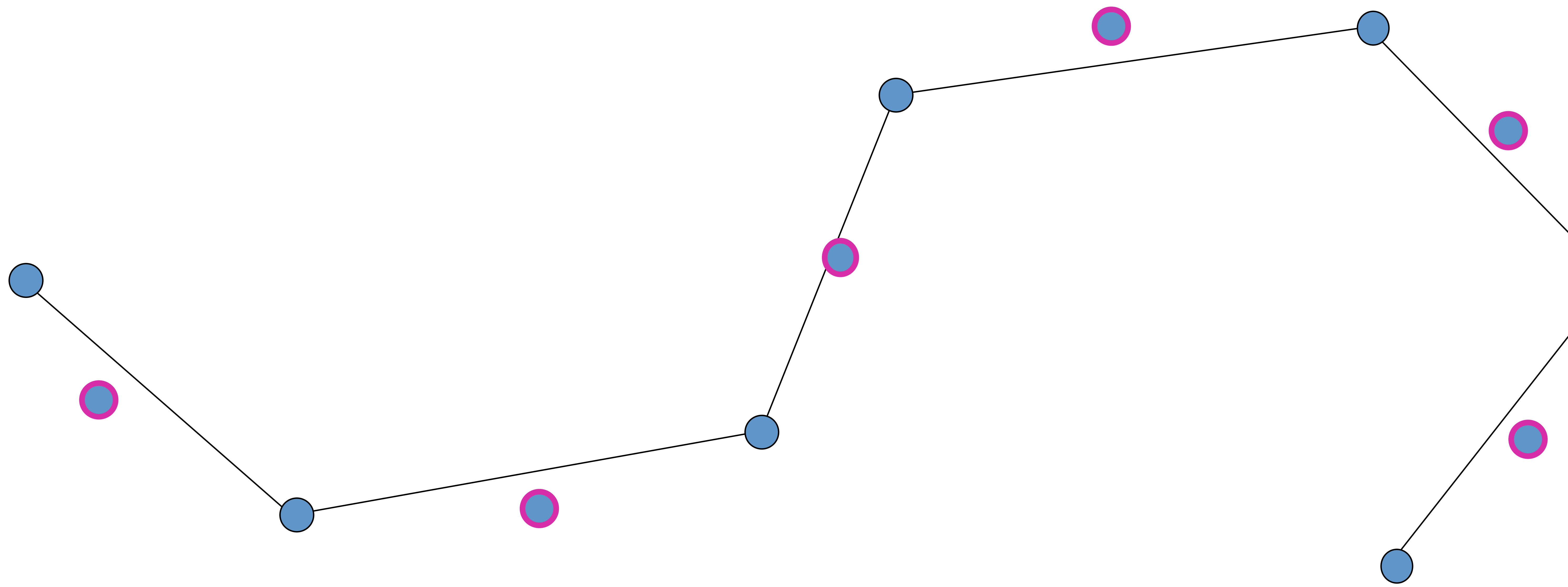


A different way of constructing curves!

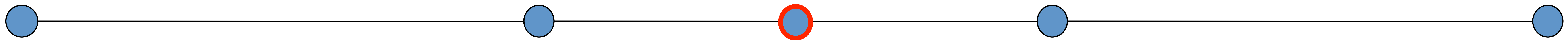




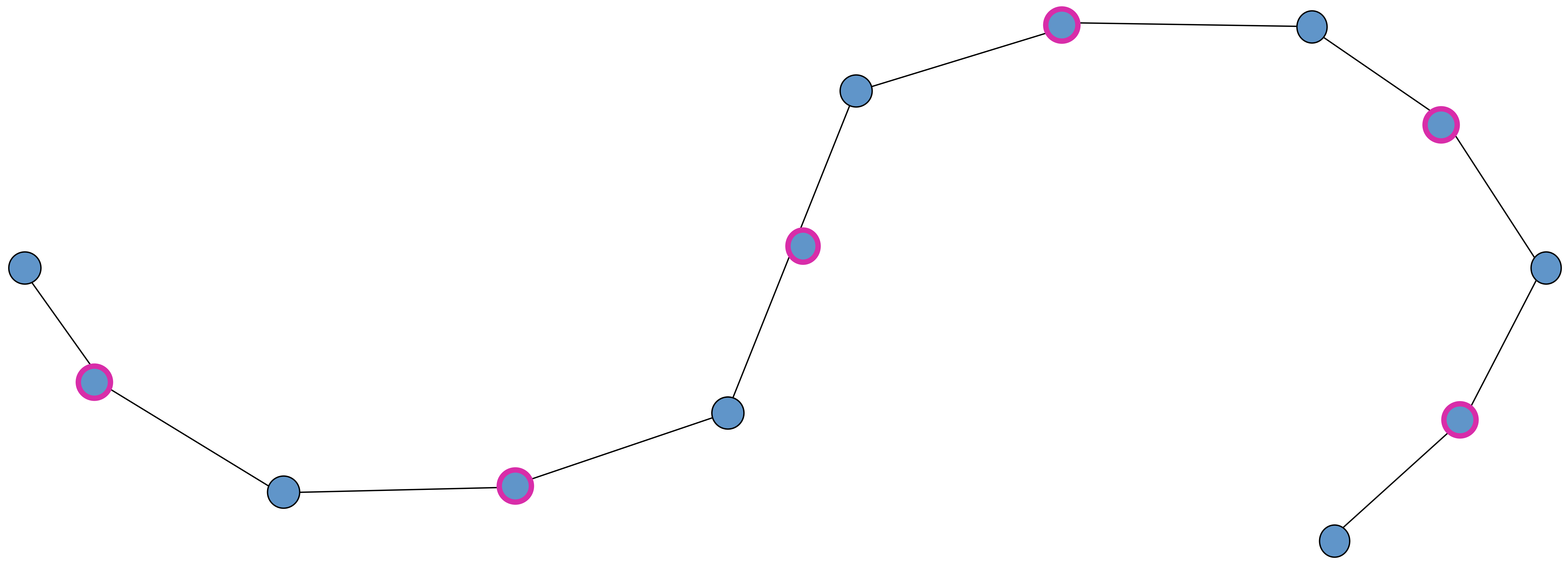
Interpolating: 4 point scheme



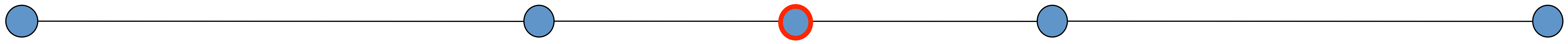
$$-\frac{1}{16} \qquad \frac{9}{16} \qquad \frac{9}{16} \qquad -\frac{1}{16}$$



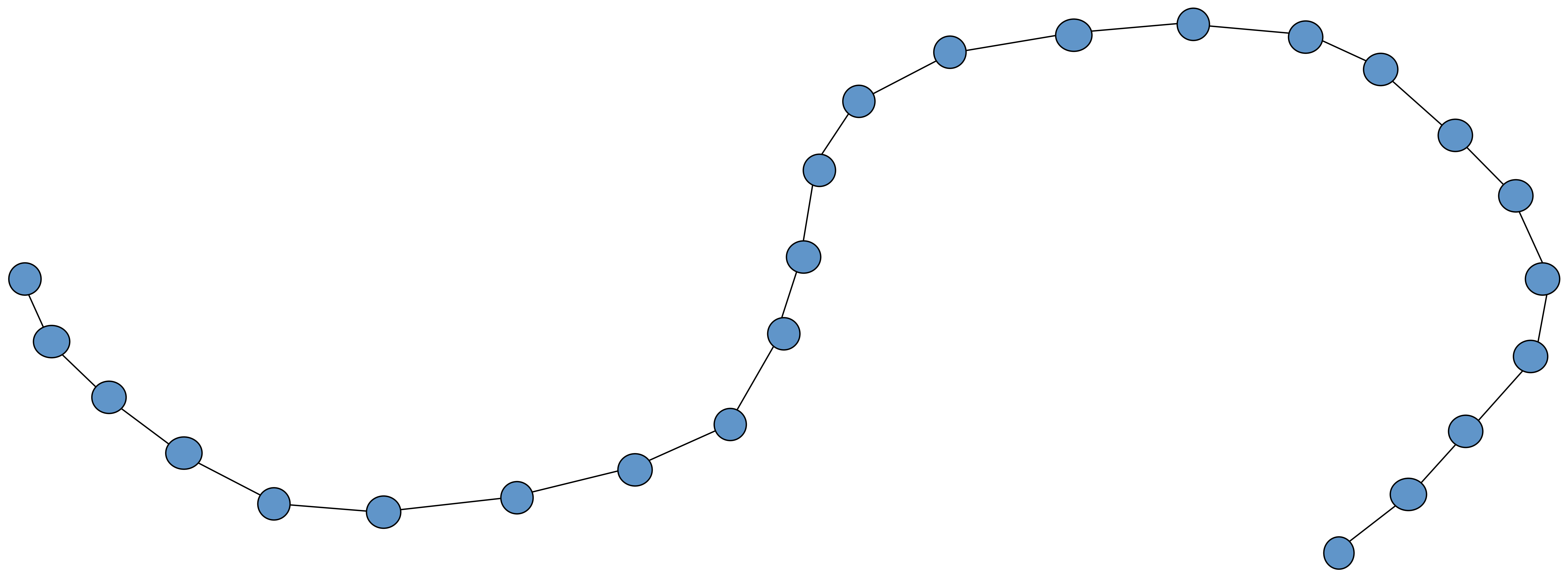
Interpolating: 4 point scheme



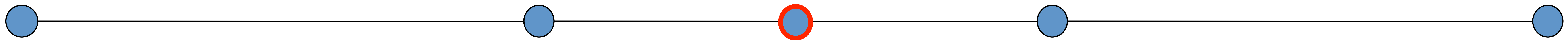
$$-\frac{1}{16} \qquad \frac{9}{16} \qquad \frac{9}{16} \qquad -\frac{1}{16}$$



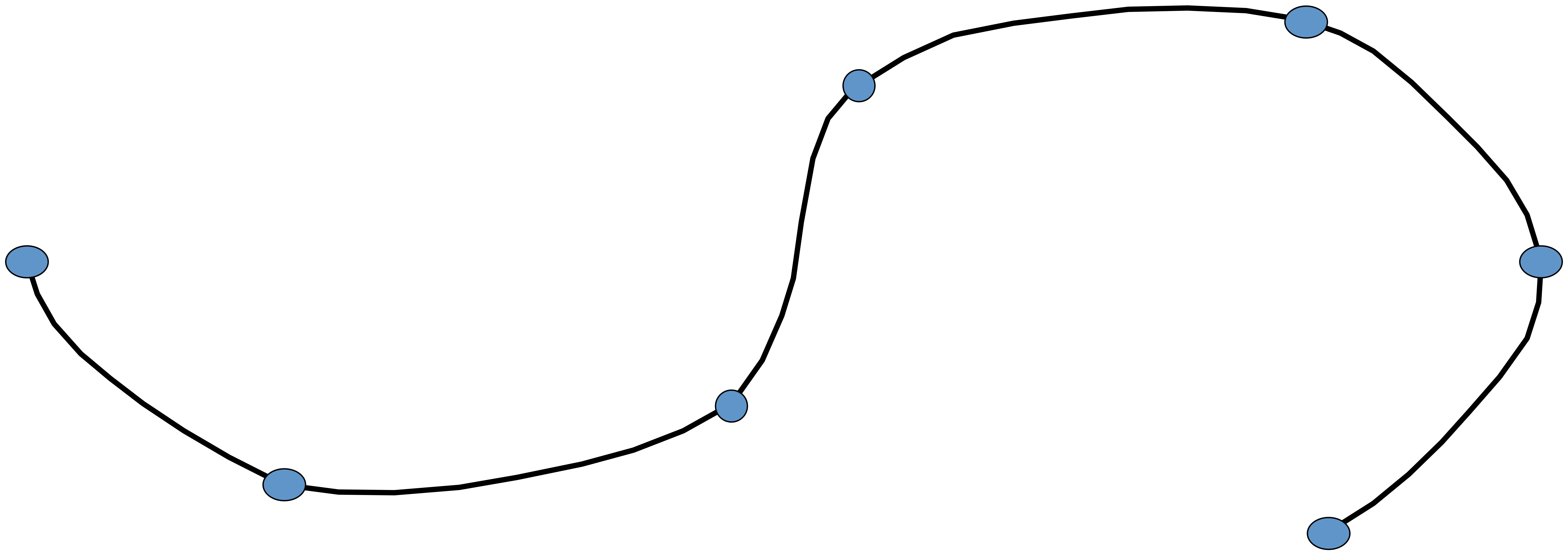
Interpolating: 4 point scheme



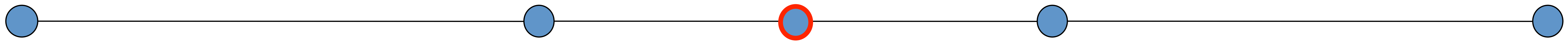
$$-\frac{1}{16} \qquad \frac{9}{16} \qquad \frac{9}{16} \qquad -\frac{1}{16}$$



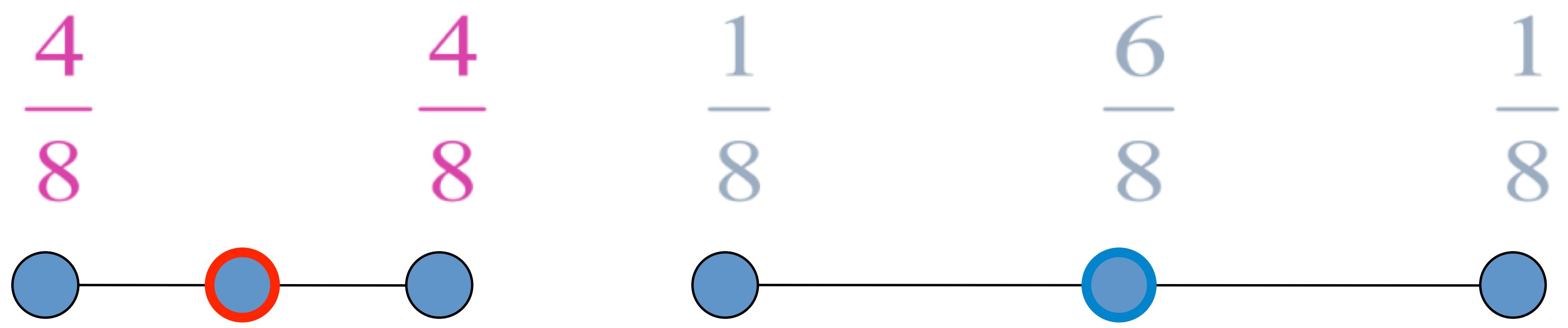
Interpolating: 4 point scheme



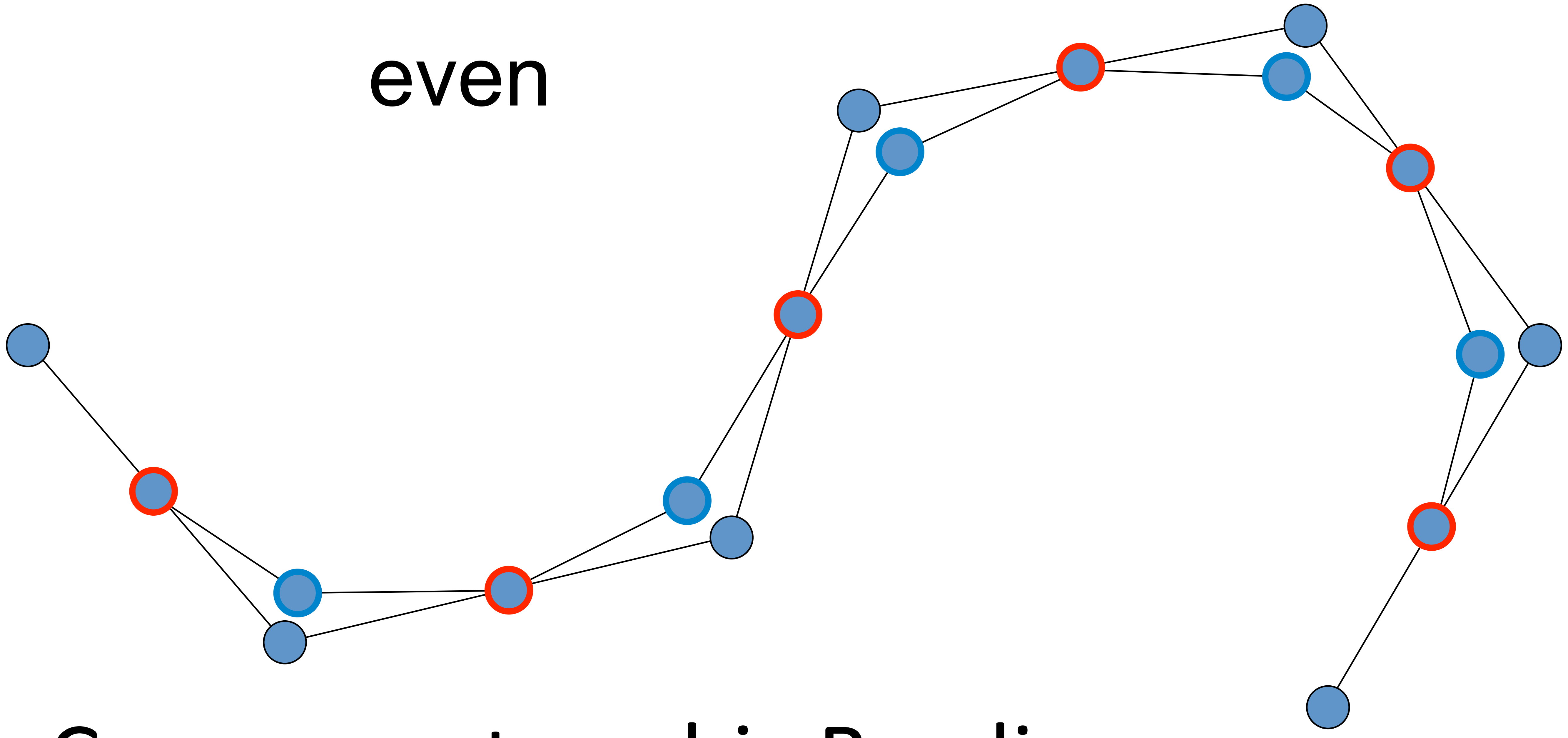
$$-\frac{1}{16} \qquad \frac{9}{16} \qquad \frac{9}{16} \qquad -\frac{1}{16}$$



Catmull-Clark scheme



odd even

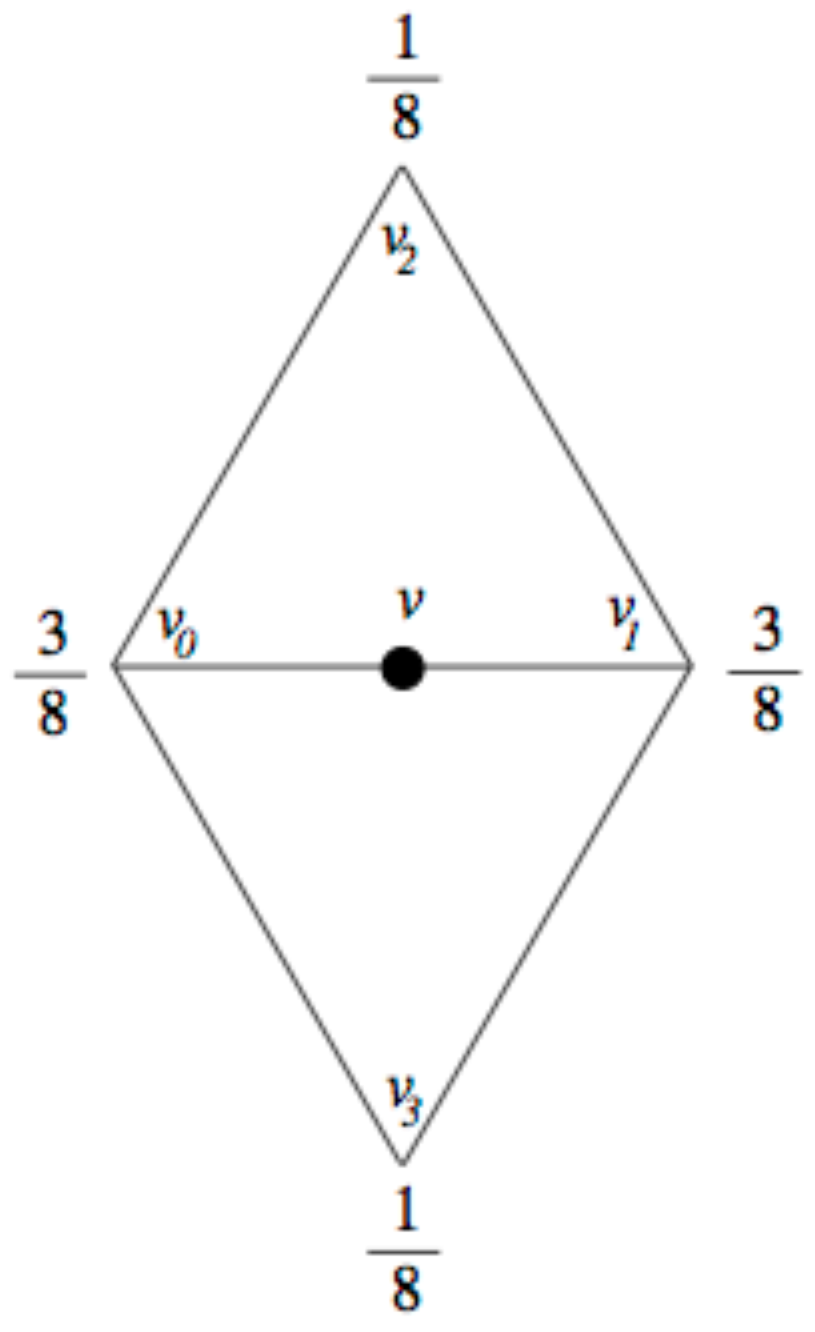


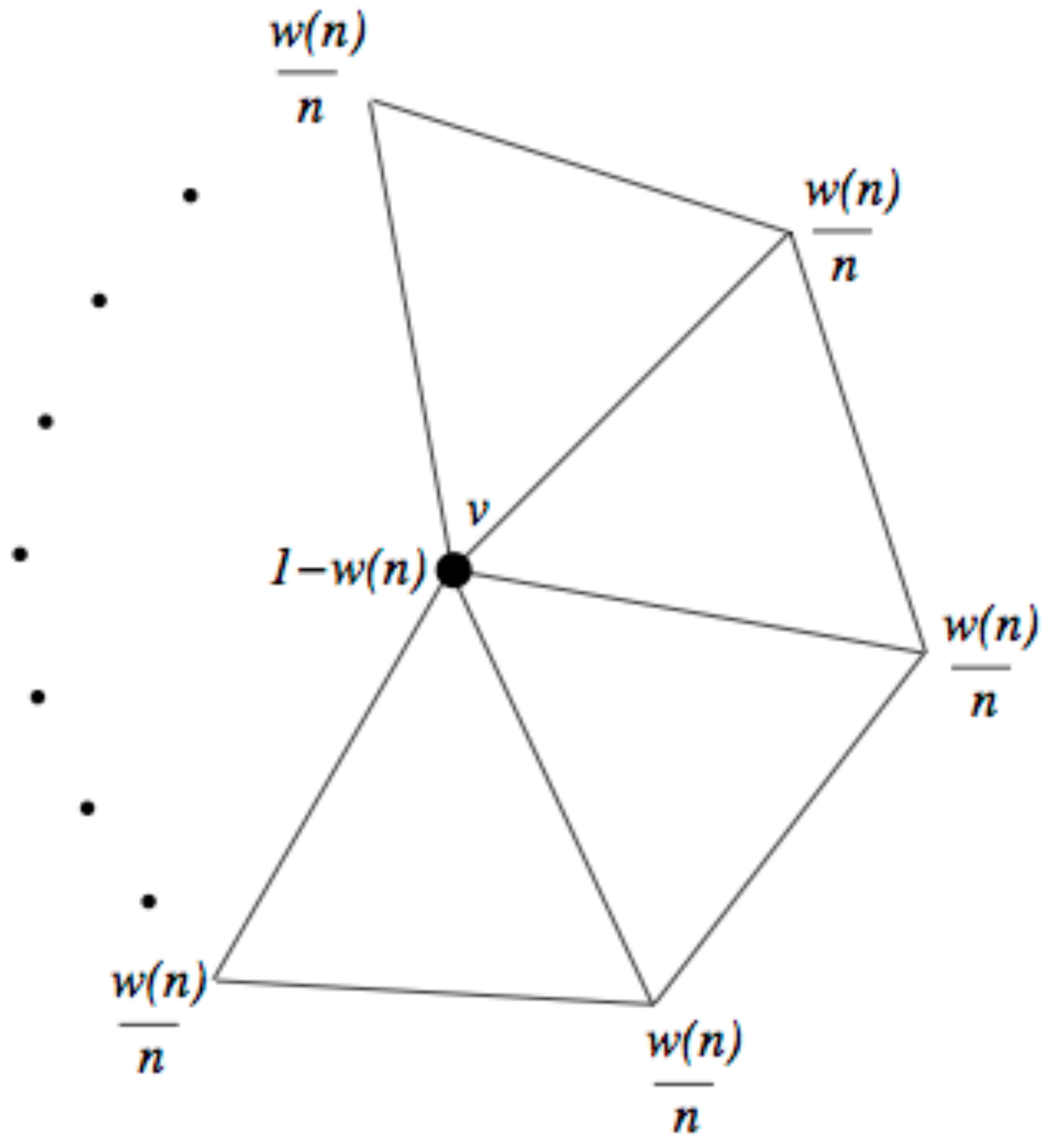
Converges to cubic B-spline

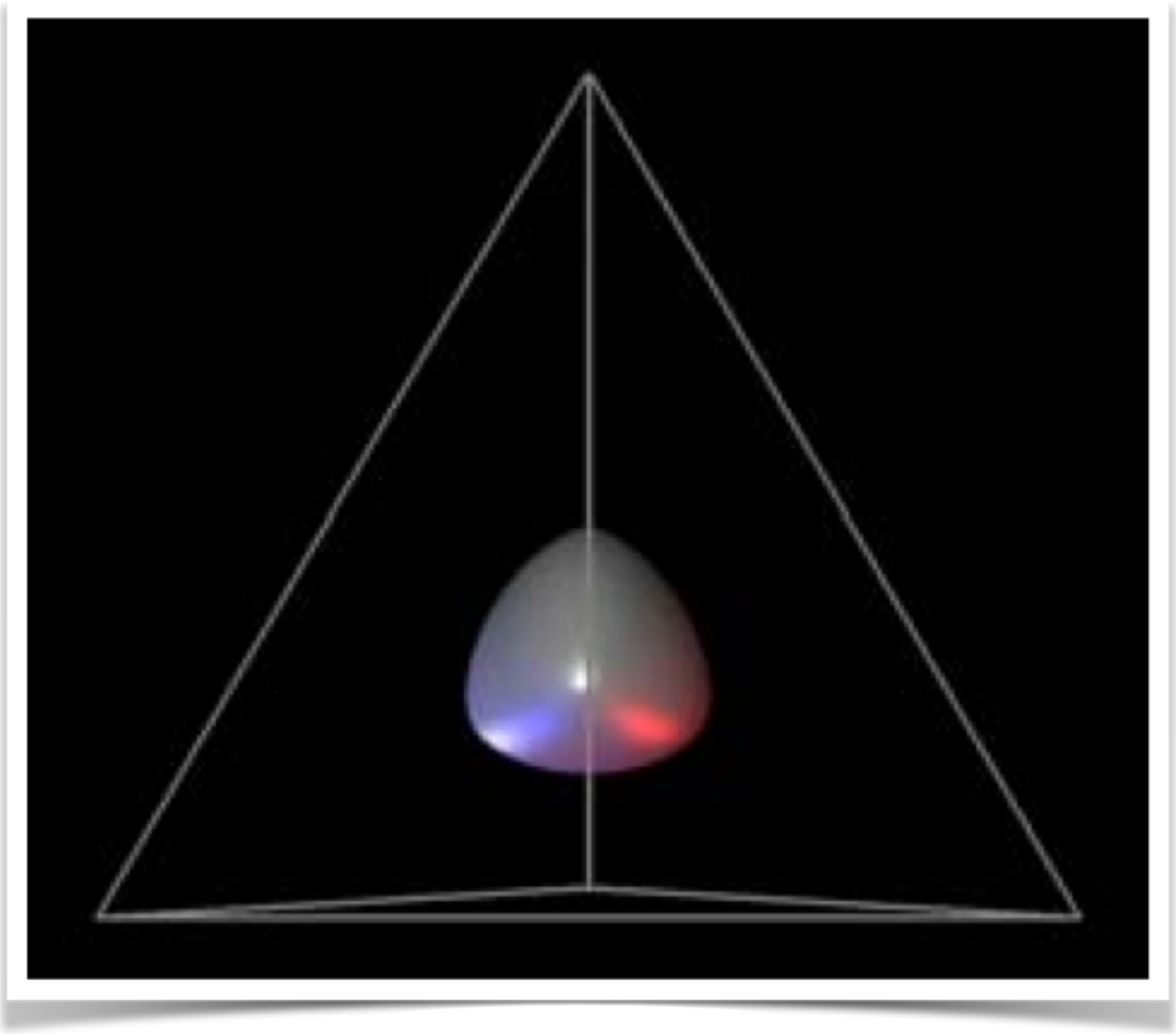
Subdivision Methods

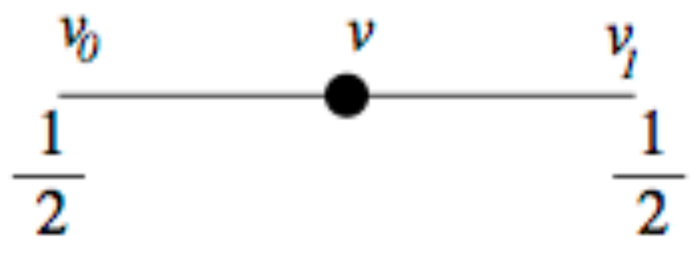
- Principal characteristics:
  - triangular or quadrangular meshes
  - approximating or interpolating
  - smoothness of the limit surface
- We will study 2 of them:
  - Loop subdivision for triangle meshes (Approximating, C^2)
  - Catmull-Clark subdivision for quadrilateral meshes (Approximating, C^2)
- Other famous schemes (see the references for details)
  - Butterfly, Kobbelt, Doo-Sabin, Midedge, Biquartic

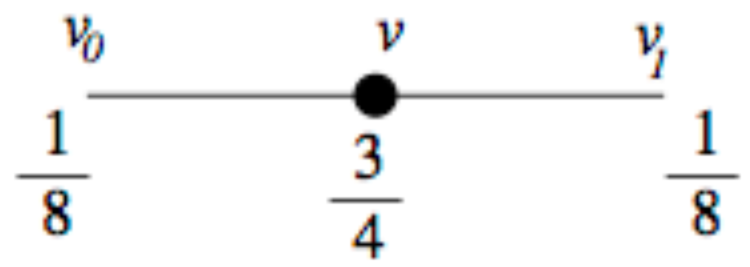
Loop subdivision









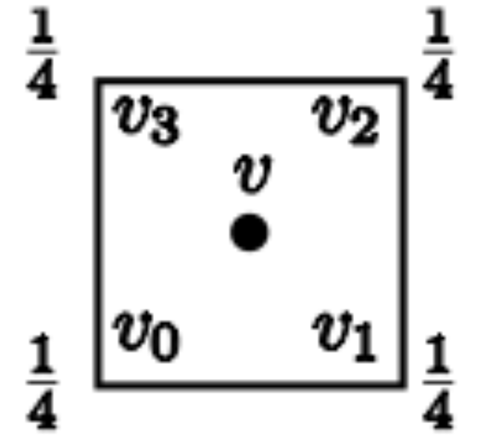


$$w(n) = \frac{5}{8} - \left(\frac{3}{8} + \frac{1}{4} cos \frac{2\pi}{n}\right)^2$$

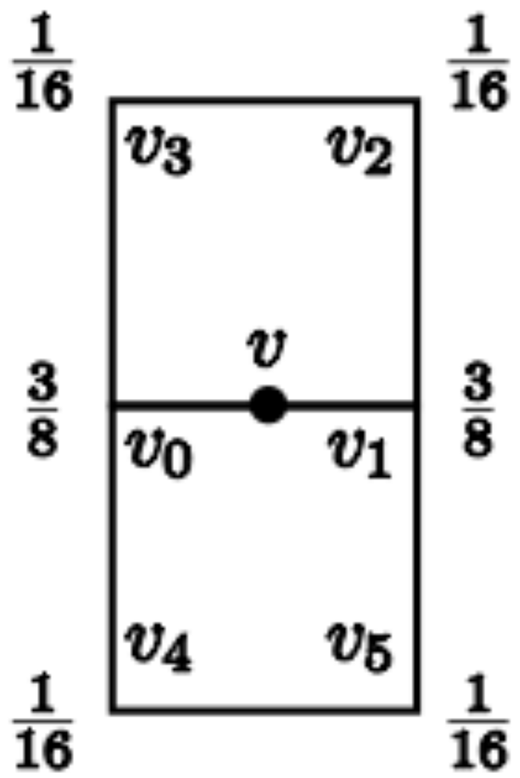
Special stencils

- Special stencils exist that allow to evaluate:
  - **The limit position of a vertex**: that is, in a single step you can compute the position of one vertex after an infinite number of subdivisions
  - **The tangent plane at a vertex**: in this case, two stencils are used and each stencil generates a vector that lies in the tangent plane
- It is also possible to evaluate the surface analytically, in other words it is possible to find the mapping between every point of the control mesh and the **limit surface.** For details see http://www.autodeskresearch.com/publications/loopsubdiv

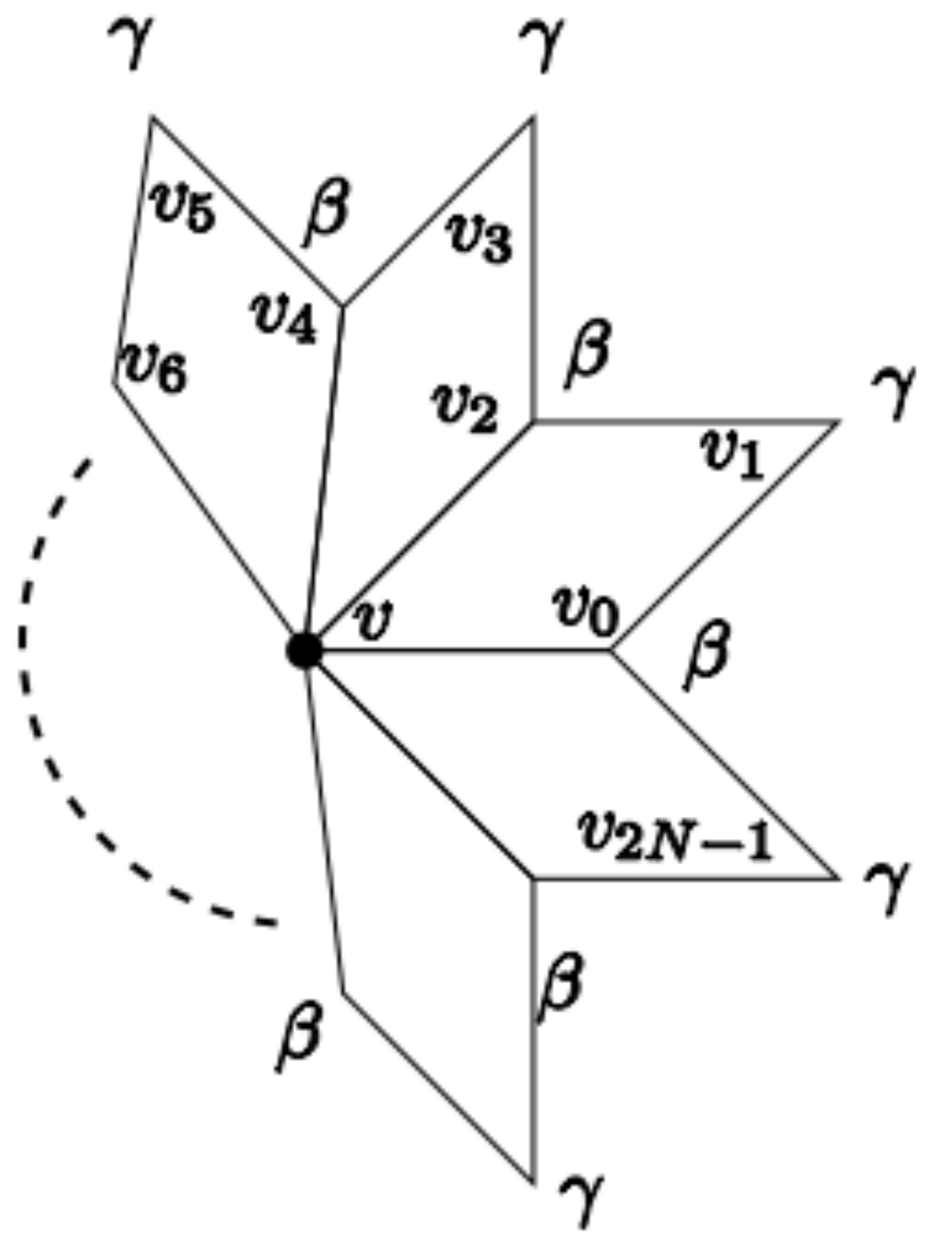
Catmull Clark subdivision



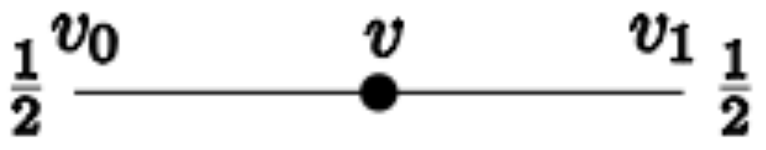
Mask for an odd vertex of type F



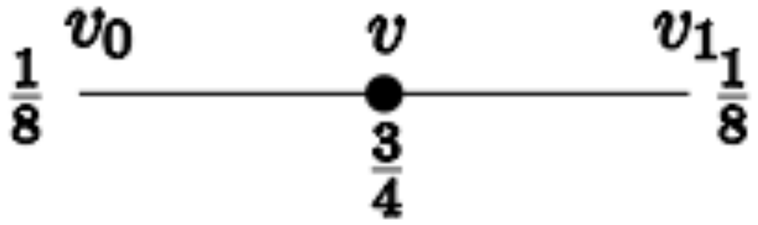
Mask for an odd vertex of type E



Mask for an even vertex



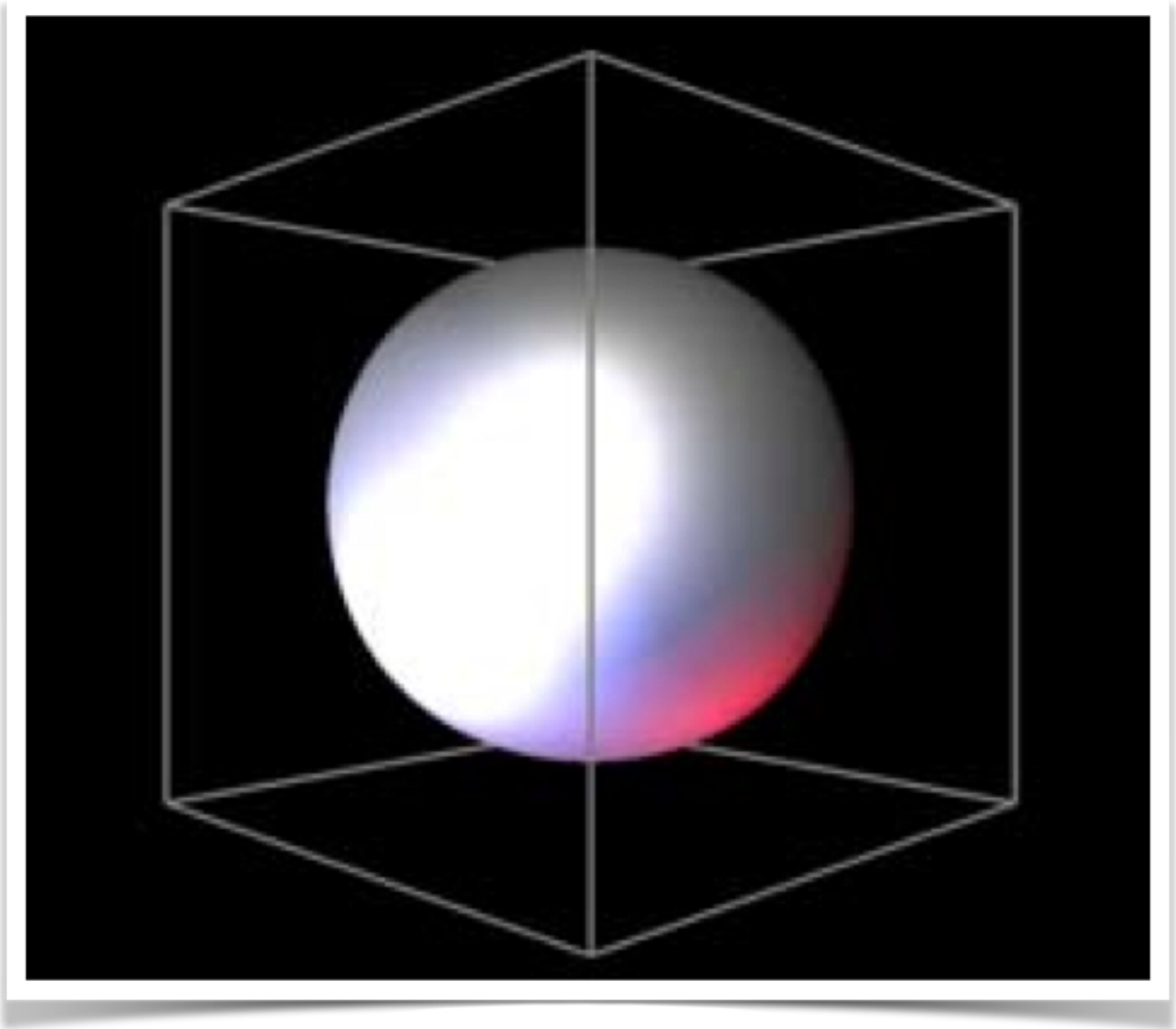
Mask for a boundary odd vertex



Mask for a boundary even vertex

$$\beta = \frac{3}{2N}$$

$$\gamma = \frac{1}{4N}$$

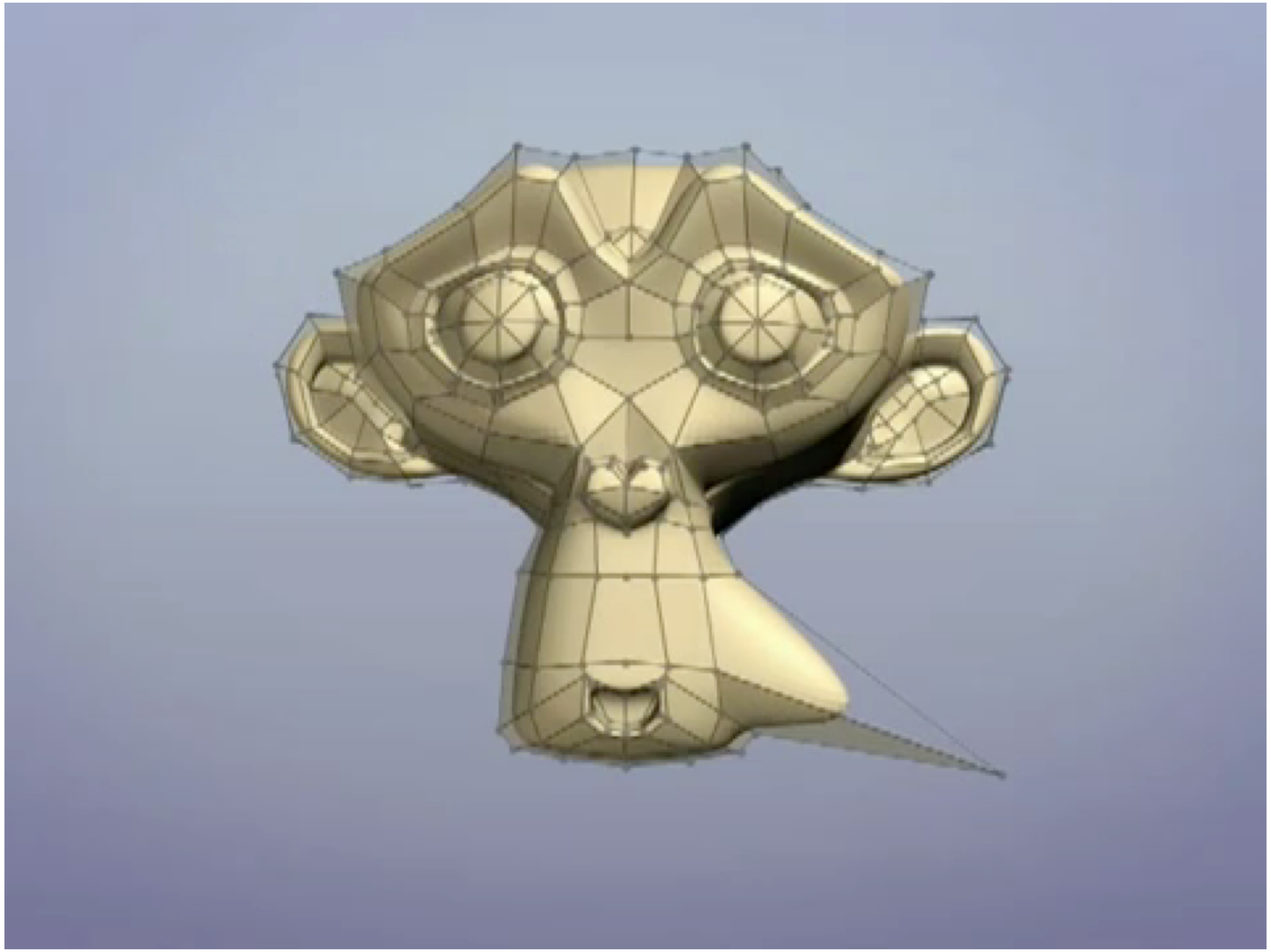


On a regular grid, the Catmull clark subdivision is a collection of bicubic Bézier patches

Catmull-Clark

- It is the standard in the animation industry
- All the major 3D modeling softwares support it
- Similarly to Loop:
  - stencils for the limit surface
  - stencils for tangent plane
  - it can be evaluated analytically

How are they used in practice?



Efficient implementation

- At every subdivision step, we add new vertices and move the existing ones in new positions
- Every position is computed as a weighted average of existing vertices
- This means that the process is linear!
- For a fixed number of levels of subdivision, the vertices of subdivided surface can be computed as:

Subdivided vertices

$$p = Sq$$

Control vertices

- where S is a **sparse** and **fixed** matrix

T-spline

- Extension of splines for non-rectangular grids
- http://www.youtube.com/watch?v=k1ro9S-cAwI

References

Fundamentals of Computer Graphics, Fourth Edition
4th Edition **by Steve Marschner, Peter Shirley**

Chapter 15

Curves and Surfaces for CAGD - Gerald Farin

Subdivision Zoo - Denis Zorin
http://www.cmlab.csie.ntu.edu.tw/~robin/courses/gm/note/subdivision-prn.pdf

Sample Projects

http://kesen.realtimerendering.com/

https://www.crazyengineers.com/threads/computer-graphics-project-ideastopics-for-csit-students.58544/