# Curves and Surfaces in Computer Graphics

I.Kolingerová

# Contents

- 1. Types of description
- 2. Approximating curves
- 3. Bezier curves
- 4. B-spline curves
- 5. NURBs curves
- 6. Approximating surfaces

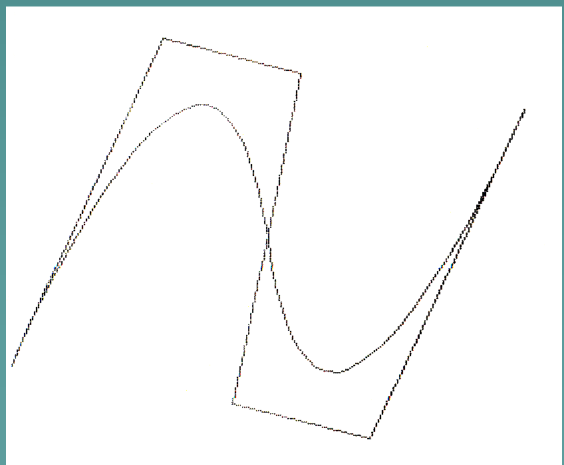
Figures: namely J.Žára a kol.: Moderní počítačová grafika, 2.vyd., Computer Press, 2004, and private if not stated otherwise

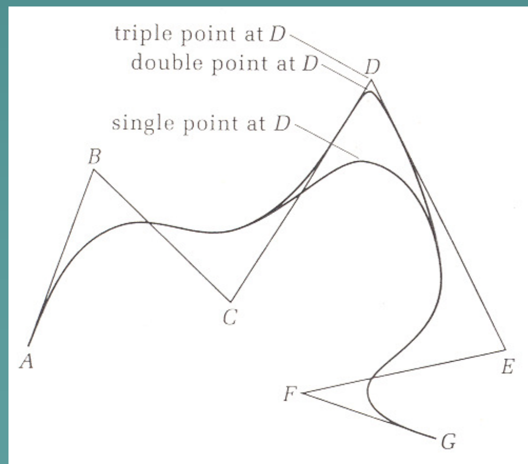
# 1. Types of description

- Analytical - explicit, implicit or parametric description
- Interpolating – given points are interpolated by the curve
- Approximating – given points are approximated by the curve, some points lie on the curve; most often in CG

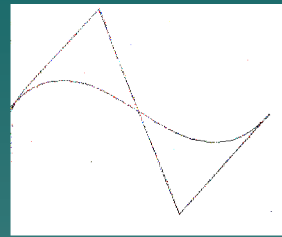
# 2. Approximating curves

- Given by control points, the points can also be multiple (it decreases the order of continuity)
- Polynomial equations => numerically instable in case of a high order => cubical curves are most often

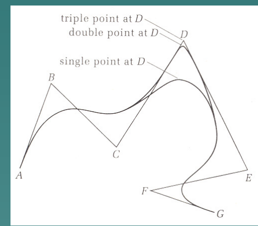




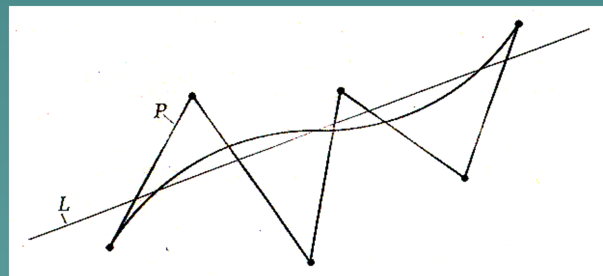
# Requirements



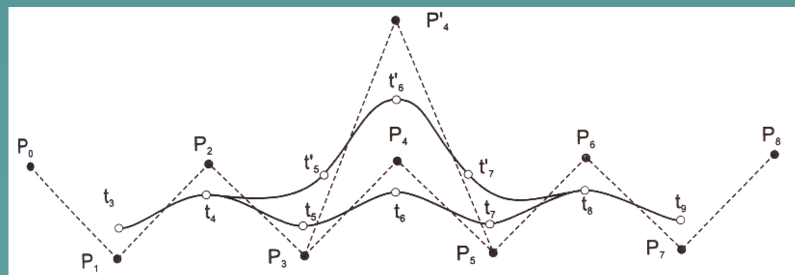
- Possibility to influence the shape by control points
- Possibility to use multiple control points
- Independence of shape of the choice of coordinate system




- Control as much local as possible
- Keeping the required order of continuity
  (also when connecting more curves/surfaces)
- Shape derived from the control points placement, variation diminishing
- Symmetry
- Computation simplicity
- Nice look - subjective
- Generality




The method usually does not meet all.



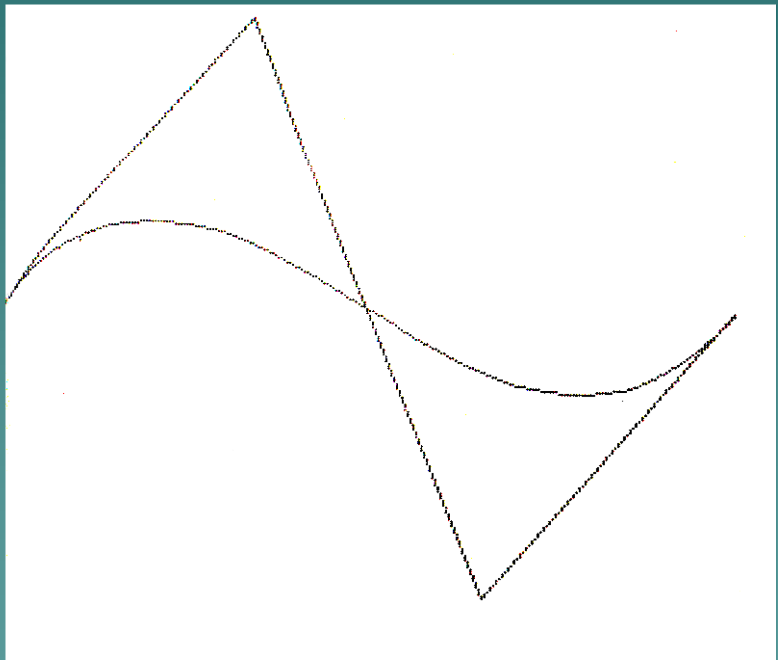
# 3. Bezier curves

- Since ~1960, French automobile industry
- The curves of degree *n* given by *n*+1 points

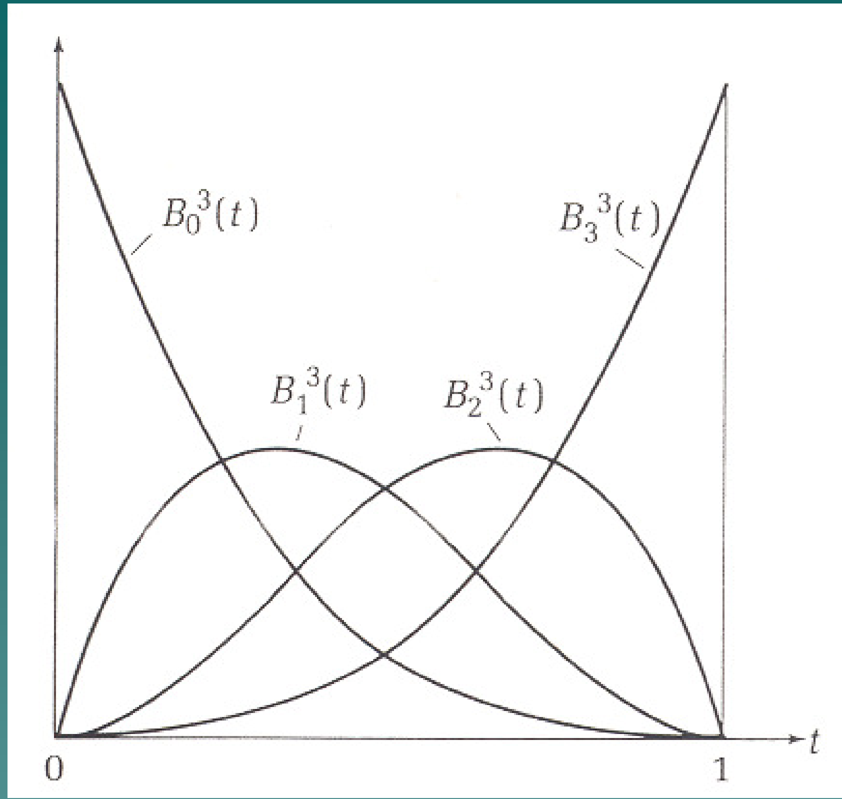
$$P(t) = \sum_{i=0}^{n} P_i B_i^n(t), t \in <0;1>$$

- where *t* - parameter, $P_i$ – control points, $B_i^n$ - Bernstein polynoms

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}, i = 0,1,\dots,n,$$
$$\binom{n}{0} = \binom{0}{0} = 1$$

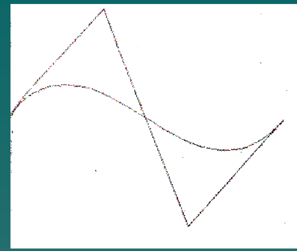


Example of Bezier curve of degree 3
(a cubics)

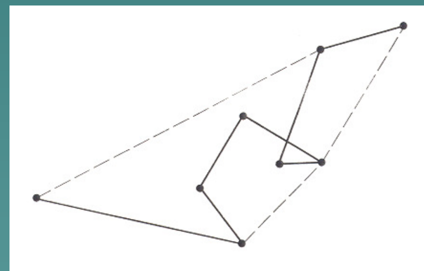



Bernstein polynoms of degree 3

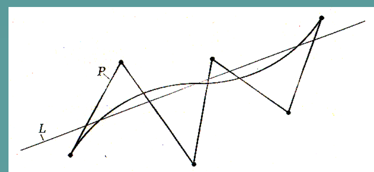
# Bezier curve properties



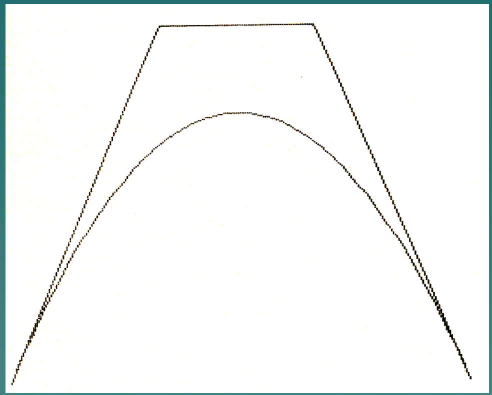
- It starts and ends in the outer control points
- A control point influences the whole curve
- Affine invariance, invariance to an affine transformation of the parameter

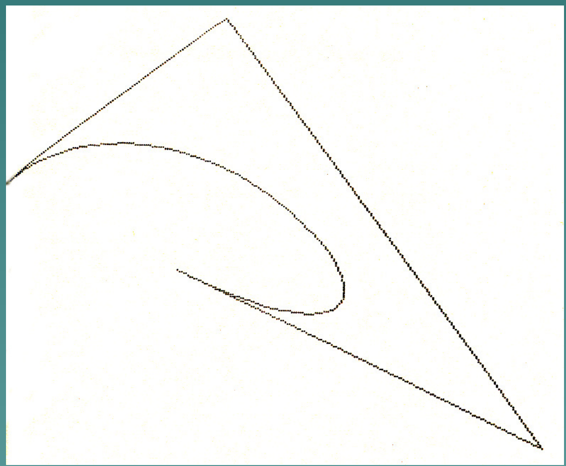


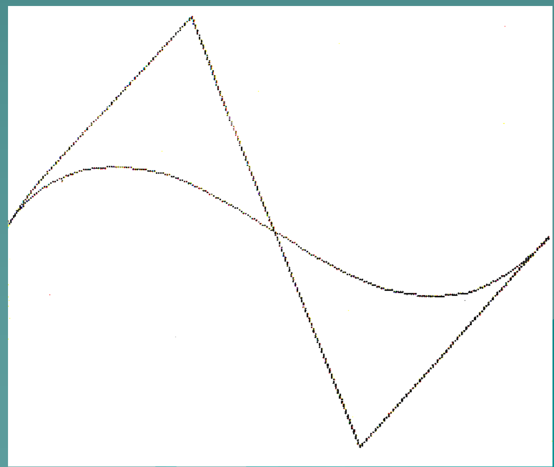
- Convex hull property
- Variation diminishing of c. points
- If more c. points than $n+1$, then either a grow of the curve degree or a need to connect more curves



# Examples of Bezier curves







Bezier cubics with their control polygons

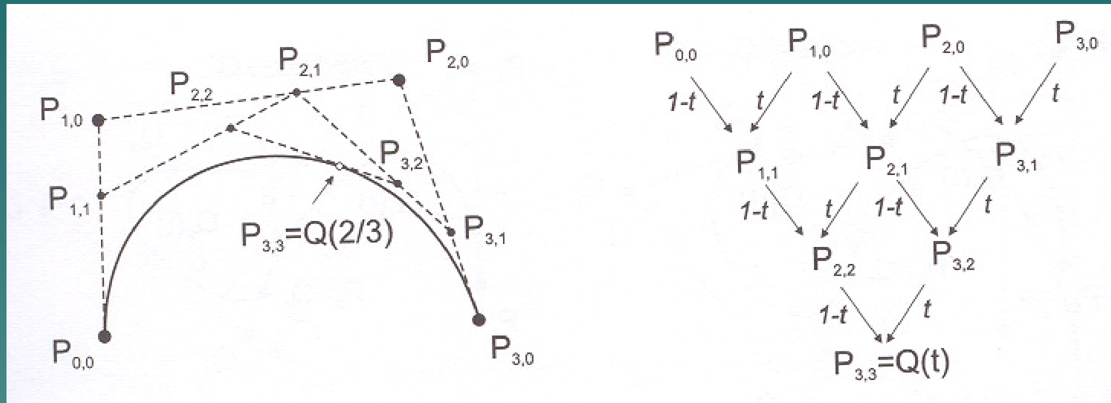
# Computation

- Algorithm de Casteljau – a point *P(t)* obtained by repeated subdivision of line segments of the control polygon in the rate t

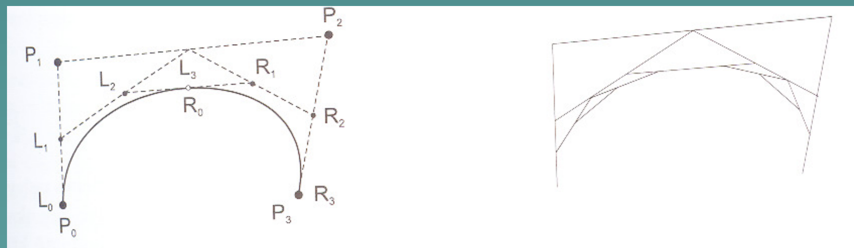
$$P_j^i = (1-t)P_{j-1}^{i-1} + tP_j^{i-1},$$
$$i = 1,2,\ldots,n,\ j = i, i+1,\ldots,n$$

- $P_n^n$ is a point of the curve

# Algorithm de Casteljau




Computation of a point in t=2/3 and the computation scheme




Subdivision of the control polygon into 2 parts,
converges into the Bezier curve

# 4. B-spline curves

- Not necessary to increase the curve degree with a higher number of control points

$$P(t) = \sum_{i=0}^{n} P_i N_i^k(t), t \in < 0; t_n >$$

$$N_i^1(t) = \begin{cases} 1 \text{ for } x_i \leq t < x_{i+1} \\ 0 \text{ in other cases} \end{cases}$$

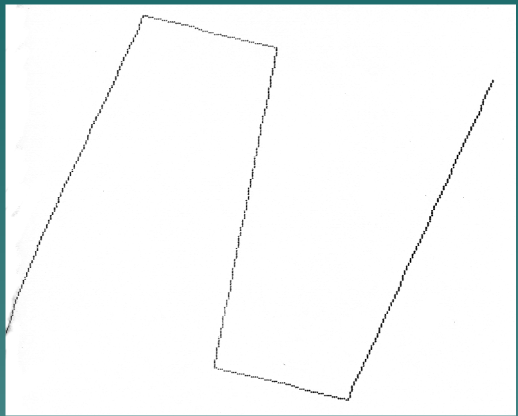
$$N_i^k(t) = \frac{(t - x_i) N_i^{k-1}(t)}{x_{i+k-1} - x_i} + \frac{(x_{i+k} - t) N_{i+1}^{k-1}(t)}{x_{i+k} - x_{i+1}}$$

$$\frac{0}{0} = 0$$

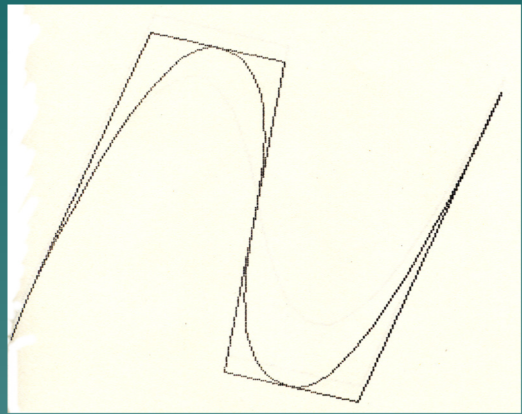
where

- $t$ - parameter,
- $P_i$ – control points,
- $N_i^k$ – basis functions,
- $k$ – curve degree
- $n+1$ – number of control points
- $x_i$ – knot vector

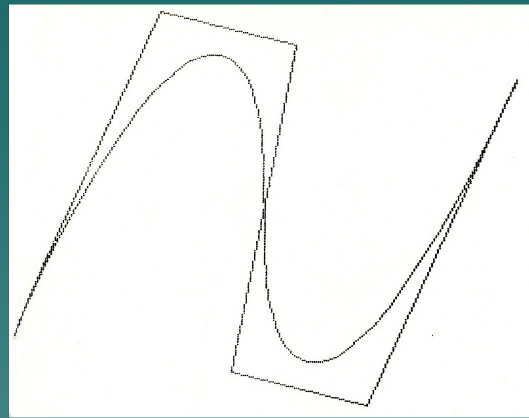
# Examples of B-spline curves



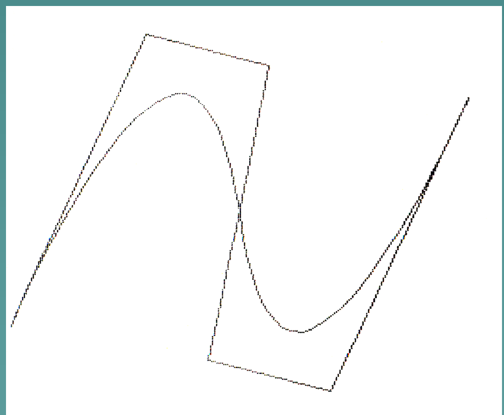
Degree 1



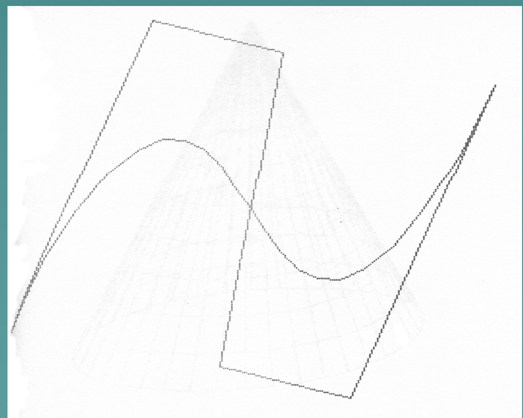
Degree 2



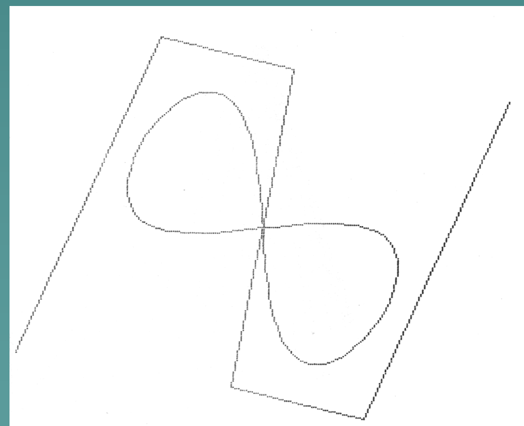
Degree 3



Degree 4



Degree 5



Degree 5, periodic

# B-spline curve properties

- It starts and ends in the outer control points
- A control point influences only max. *k+1* intervals
- Affine invariance, invariance to an affine parameter transformation
- Convex hull property
- Variation diminishing property
- More control points than $n$+1 implies neither the necessity of order growth, nor of connection of more curves

# Choice of knot vector

- For non-periodic B-spline modelling open curves: $k+n+1$ values

  $x_i=0$ for $i = 0,1,\ldots,k-1$

  $x_i=i-k+1$ for $k \leq i \leq n$

  $x_i=n-k+1$ for $i = n+1,n+2,\ldots,n+k$

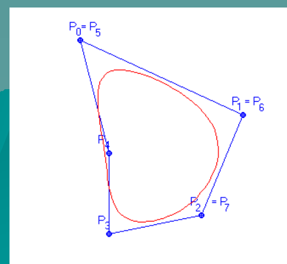
- We obtain $n-k+1$ segments

- For periodic B-spline modelling closed curves:

  $x_i=i,\ i=0,1,\ldots,n+1$

  For this choice we have to reduce the basis function to:

$$N_i^k(t) = N_0^k((t-i+n+1)\bmod(n+1))$$



# Examples 1

- B-spline given by 4 points, degree 3, open:

  we get 1 cubic segment

  knot vector is (0,0,0,1,1,1,1)

  the identical curve with the Bezier cubical curve

- B-spline given by 5 points, degree 3, open:

  we get 2 cubic segments

  knot vector is (0,0,0,1,2,2,2,2)

# Examples 2

- B-spline given by 4 points, degree 3, closed:
  we get 4 cubic segments
  knot vector is (0,1,2,3,4)
  the curve does not start and end in the given points

- B-spline given by 5 points, degree 3, closed:
  we get 5 cubic segments
  knot vector is (0,1,2,3,4,5)
  the curve does not start and end in the given points

# Coons B-spline curve

Uniform cubic B-spline curve, does not start and end in the outer control points

$$P_j(t) = \frac{1}{6}\sum_{i=0}^{3} P_{j+i} C_i(t-j), t \in < j; j+1 >, j = 0,\ldots,n-3$$

$$C_0(s) = -s^3 + 3s^2 - 3s + 1 = (1-s)^3$$

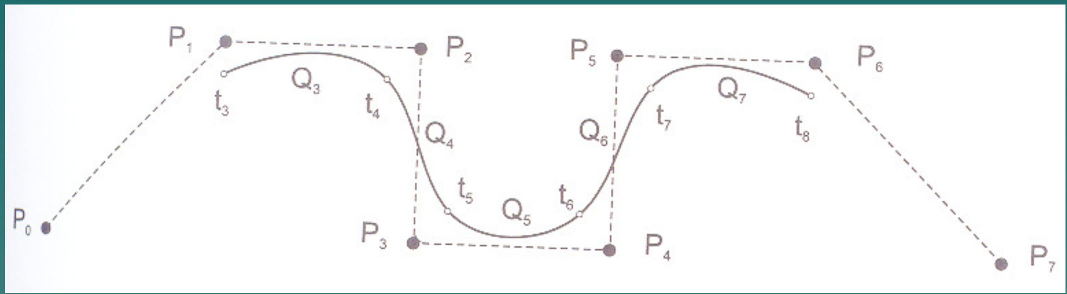
$$C_1(s) = 3s^3 - 6s^2 + 4$$

$$C_2(s) = -3s^3 + 3s^2 + 3s + 1$$

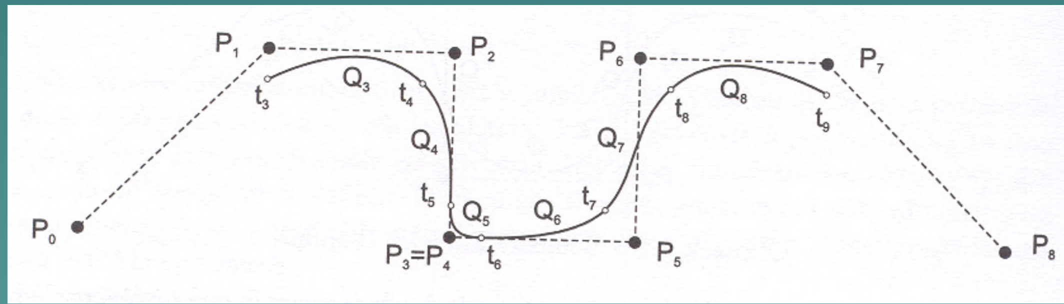
$$C_3(s) = s^3$$

- Advantage: an easier computation than B-spline, easy connection of curves, a change of a control point influences max. 4 segments

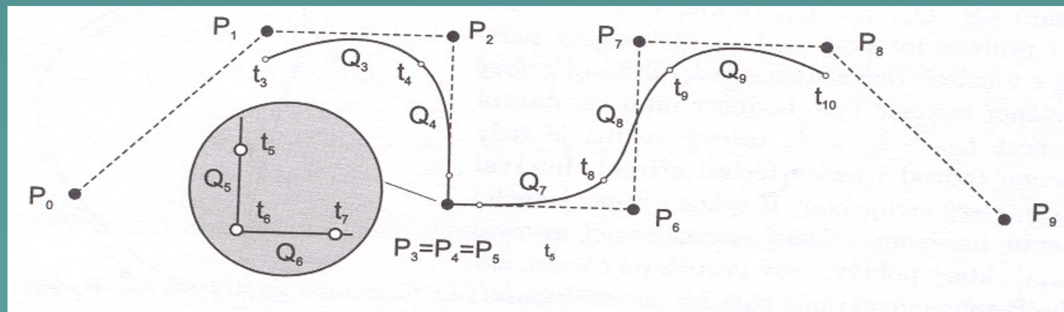
# Examples – Coons cubic B-spline



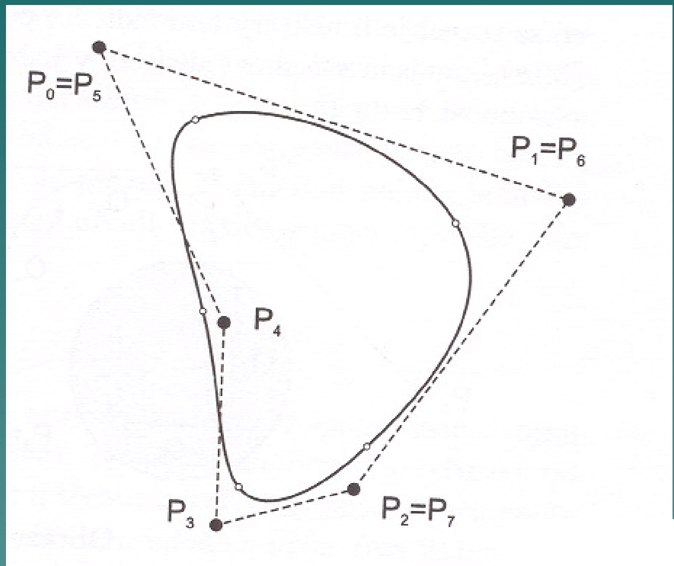

Double control point



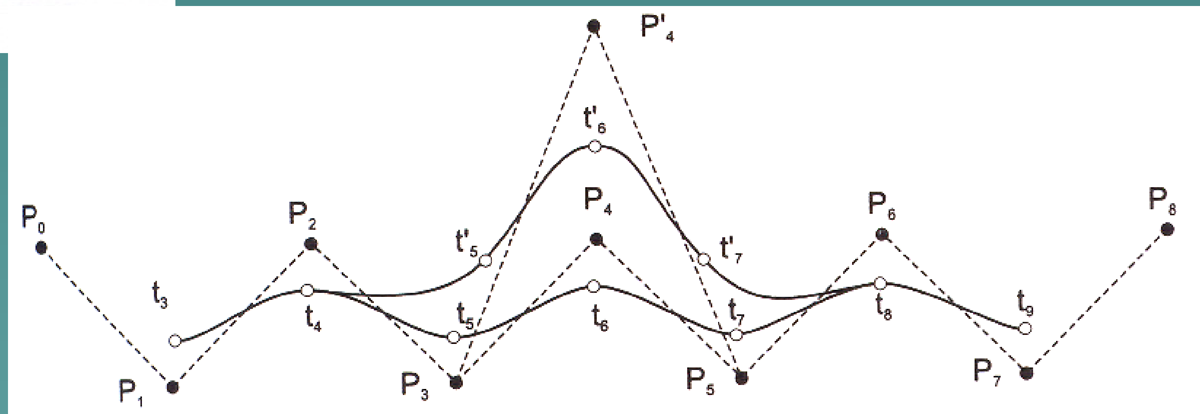

Triple control point



# Examples - Coons cubic B-spline




Closed spline




Locality of shape change when a control point is moved

# 5. NURBs curves

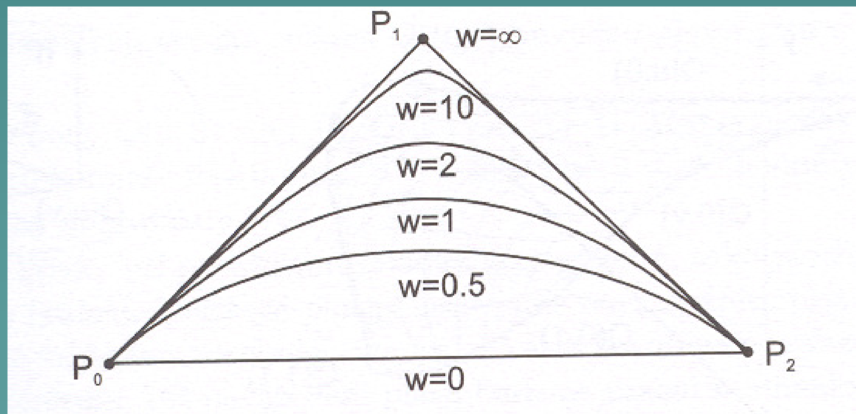
- Non-uniform rational B-spline curves
- Non-uniformity – relates to the parameter, rationality – to the point weights and to the division by the sum of weights

$$P(t) = \frac{\sum_{i=0}^{n} w_i P_i N_i^k(t)}{\sum_{i=0}^{n} w_i N_i^k(t)}$$

$w_i$ – weights of control points, usually non-negative

# NURBs curve properties

- As B-spline, but it can also describe cone sections (e.g., a circle) and has wider possibilities of control
- For unitary weights identical with a B-spline curve, for *n*+1 points and degree *n* with Bezier

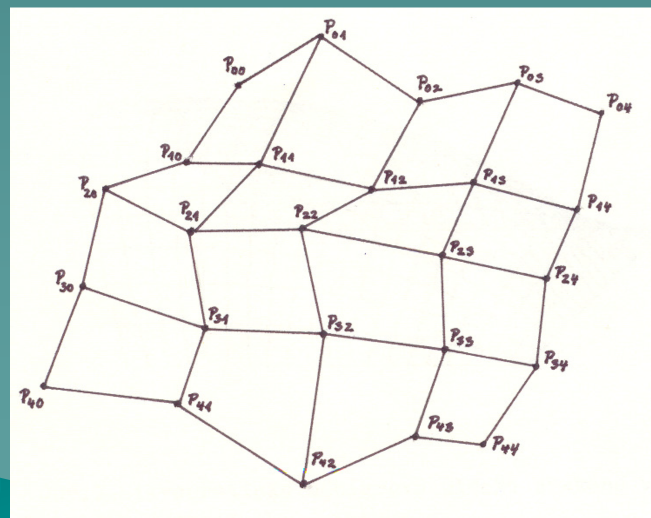


# Computation

- Cox-de Boor algorithm – a generalization of de Casteljau, a repeated subdivision of control polygon in the parts which influence the current segment

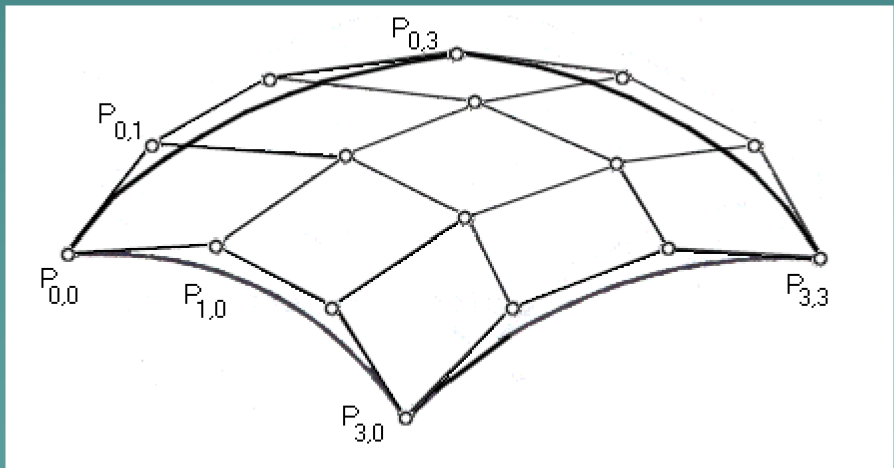
# 6. Approximating surfaces

- Given by a matrix of control points (they form a net, a map of the surface – a control polyhedron)
- Equations are a combination of control points and pairs of basis functions for 2 parameters
- Properties correspond to the curves from which they are built



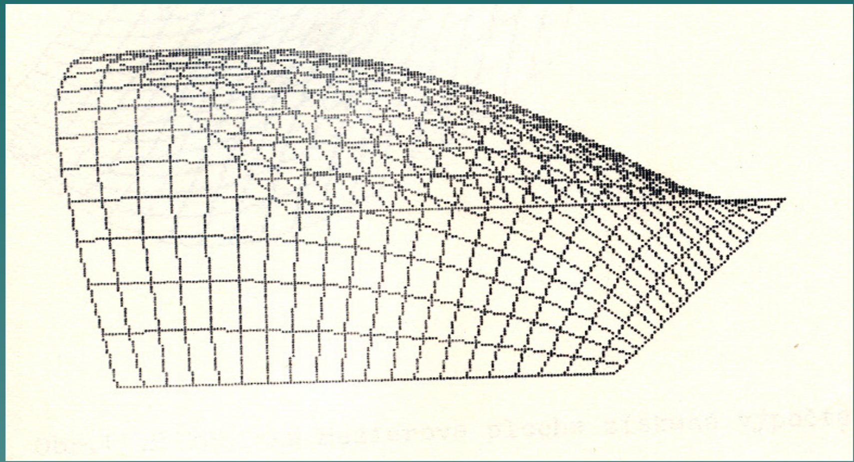
# Bezier surfaces

$$P(u,v) = \sum_{i=0}^{m} \sum_{j=0}^{n} B_i^m(u) P_{i,j} B_i^n(v), u, v \in <0;1>$$

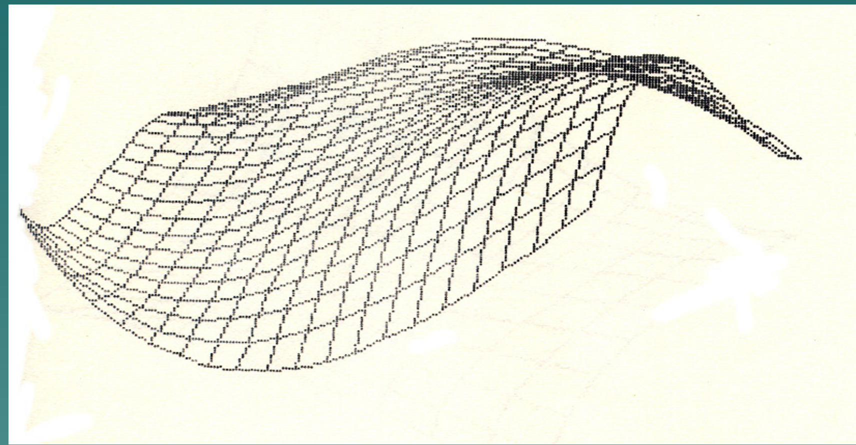



Bezier bicubic patch

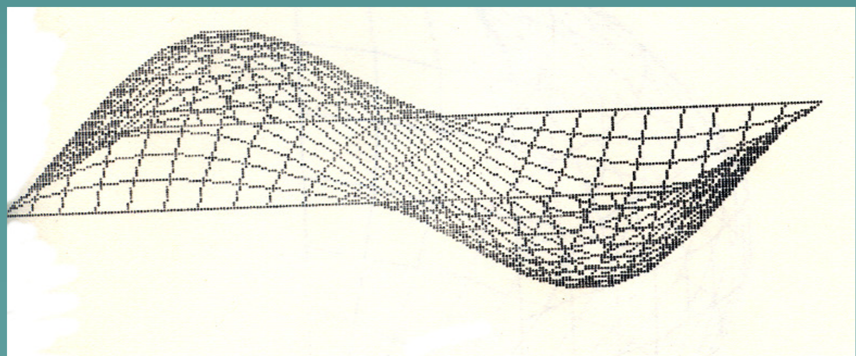
# Bezier surfaces examples



Biquadratic

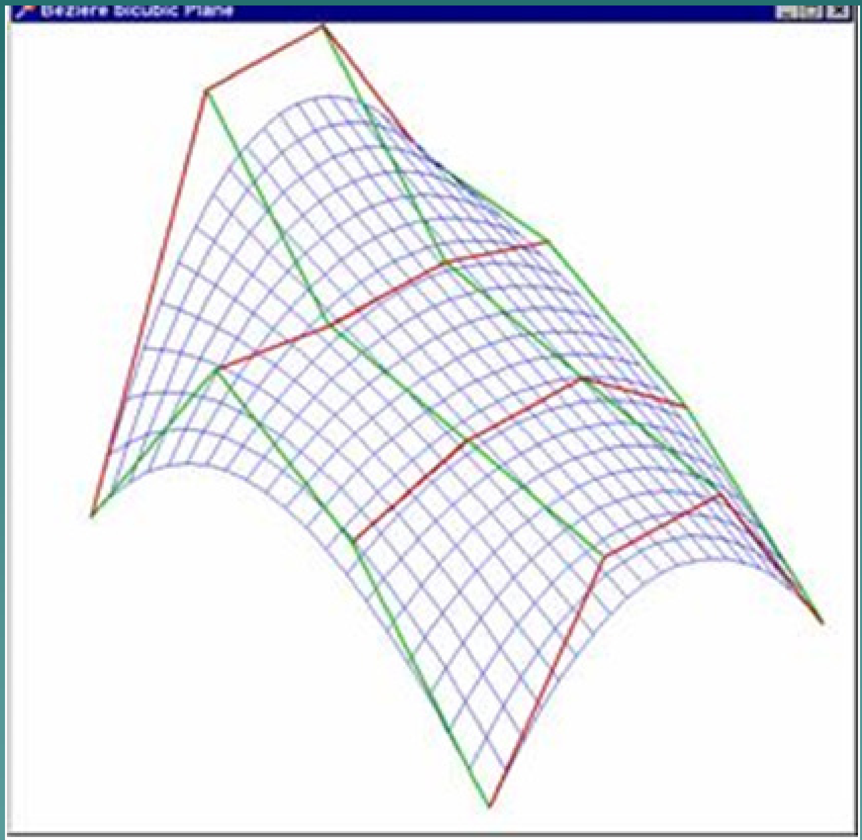


Bicubic



Degree 3 and 5

# Bezier surfaces examples



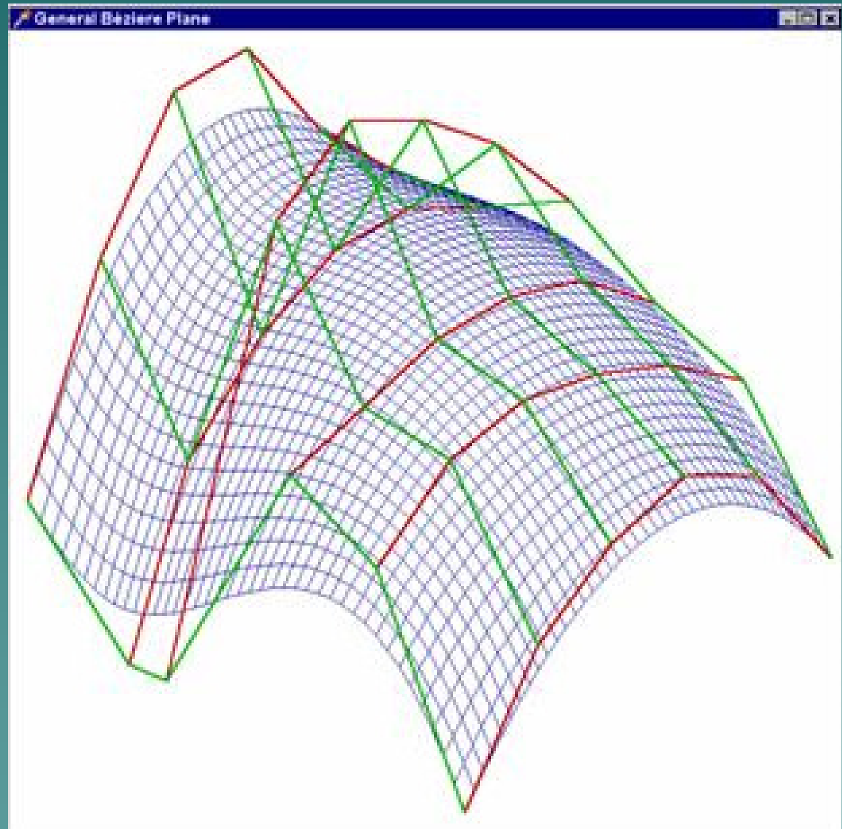



Fig.: http://mathonline.fme.vutbr.cz/pg/Algoritmy/
07_TECHNICKE_PLOCHY.htm

# B-spline surfaces

$$P(u,v) = \sum_{i=0}^{m}\sum_{j=0}^{n} N_i^k(u) P_{i,j} N_j^l(v), u,v \in <0;t_n>$$

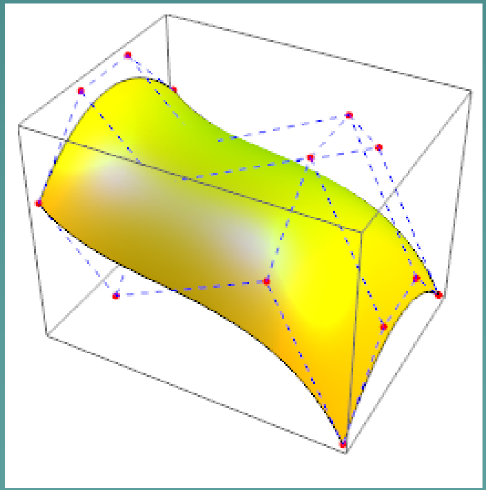
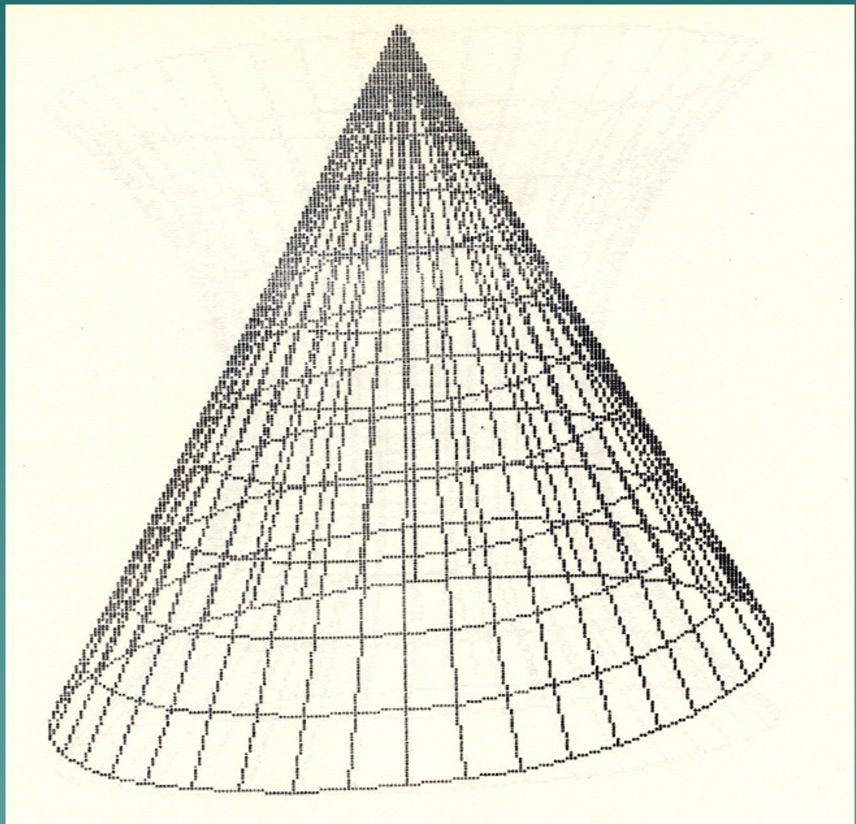
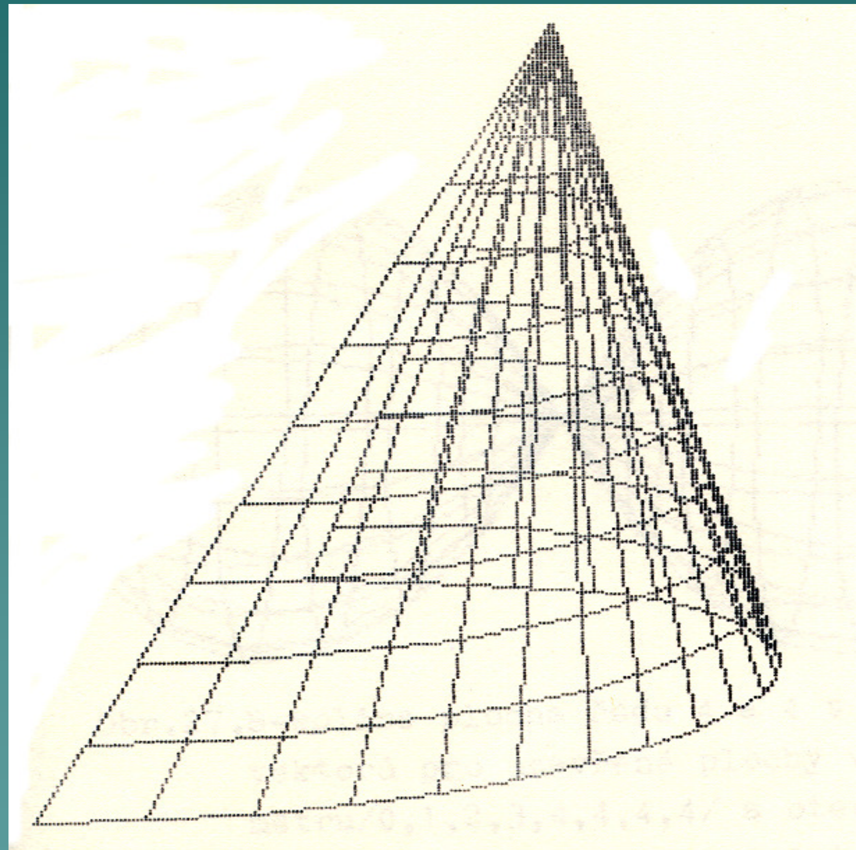


Fig.: http://blog.wolfram.com/wp-content/uploads/2009/01/splines_in28.png

# B-spline surfaces examples

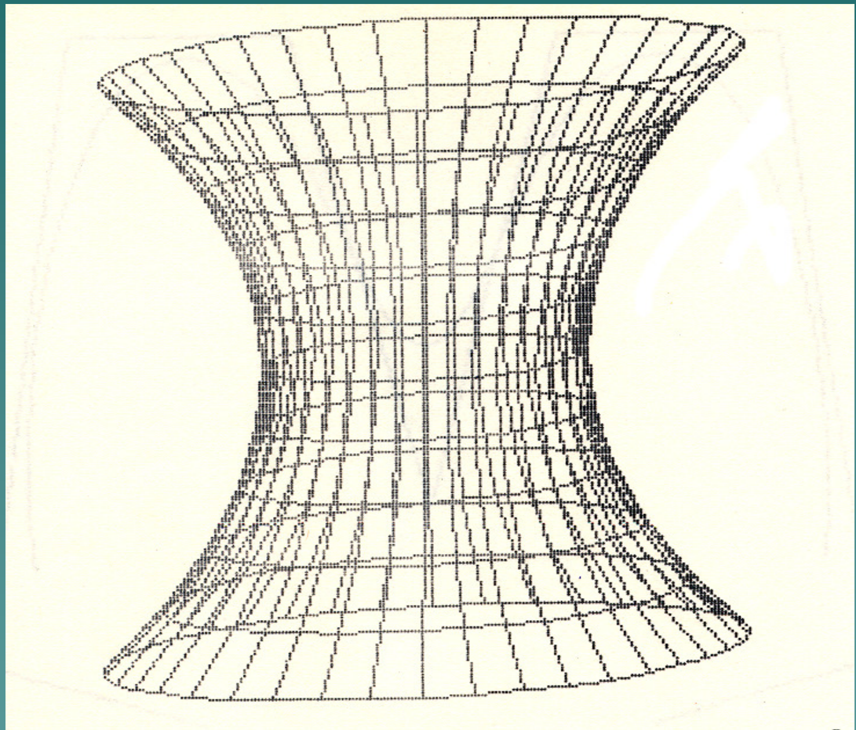


Degrees 1 (periodic) and
3 (non-periodic)

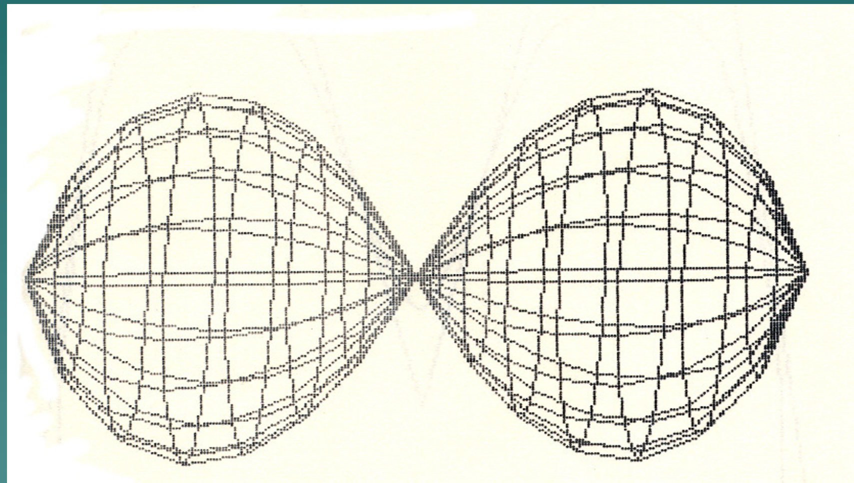


Degrees 1 and 3 (2x non-periodic)

# B-spline surfaces examples

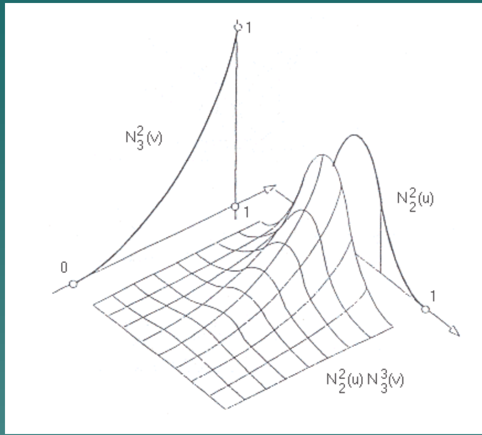


Degrees 2 (non-periodic) and
3 (periodic)



Degrees 3 (periodic) and
3 (non-periodic)

# B-spline surfaces examples




Patch $N^2{}_2(u)N^3{}_3(v)$

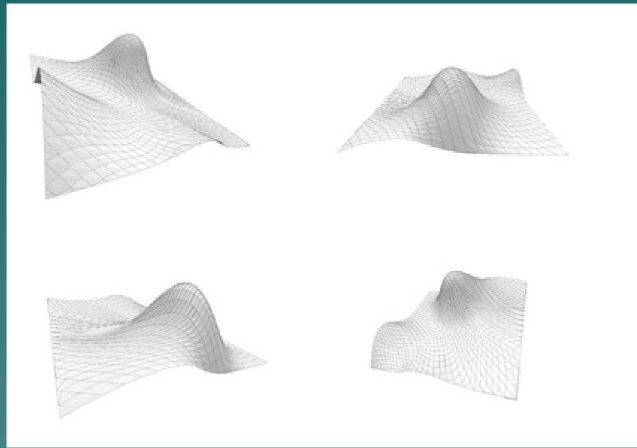


Fig.: http://martha.pnoe.net/blog/wp-content/uploads/2006/10/surfaces.jpg

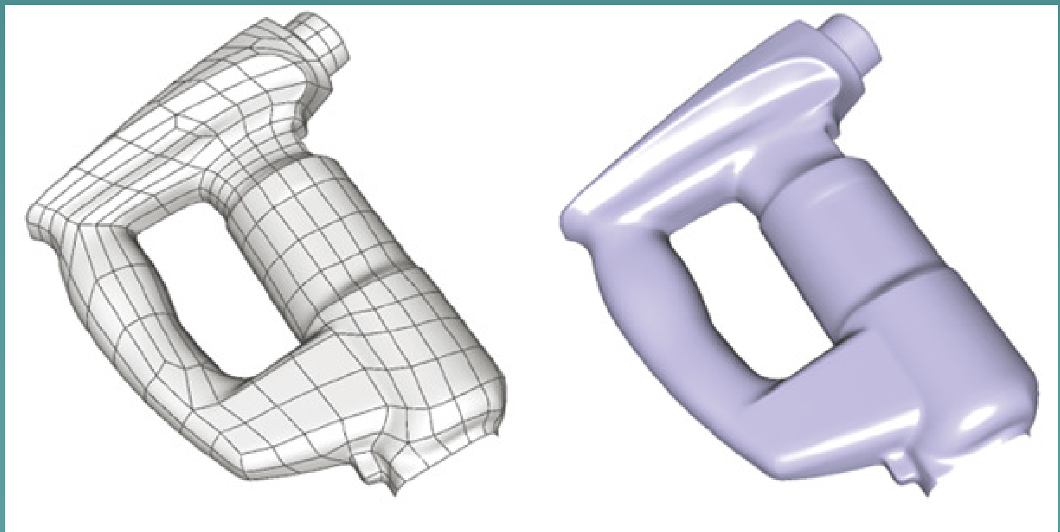


Fig.: http://graphics.c.u-tokyo.ac.jp/~kanai/archives/uploads/2008/04/spray.png

# NURBs surfaces

$$P(u,v) = \frac{\sum_{i=0}^{m}\sum_{j=0}^{n} w_{i,j} P_{i,j} N_i^k(u) N_j^l(v)}{\sum_{i=0}^{m}\sum_{j=0}^{n} w_{i,j} N_i^k(u) N_j^l(v)}$$

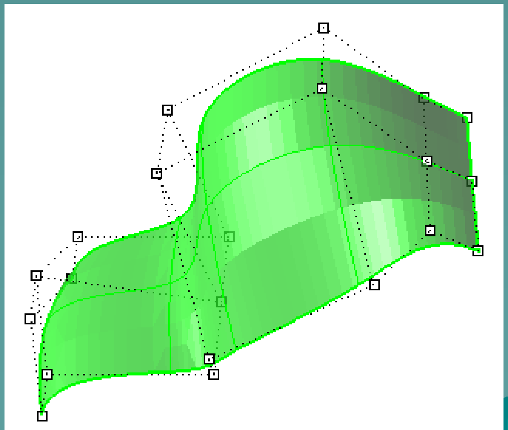
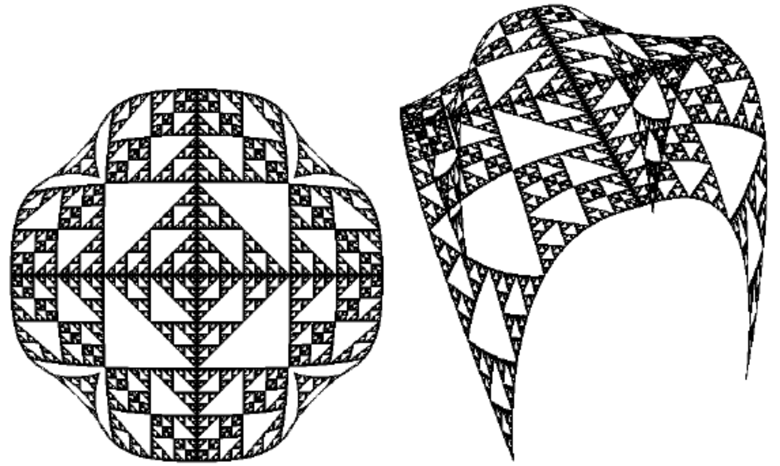
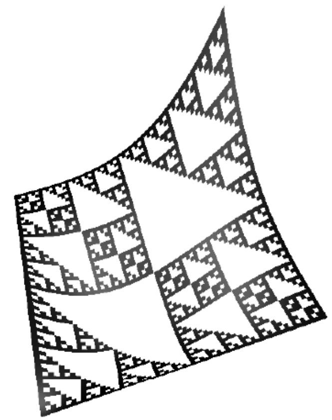


Fig.: H.Novotná:
Modelování v prostoru, 2009

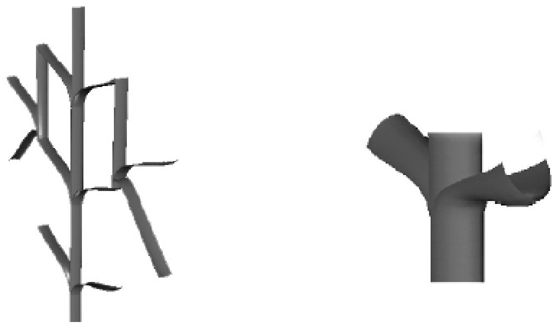
Something else to show how beautiful tool the surfaces can be:
Approximating surfaces, combined with fractals



A top and perspective views of a fractal bicubic surface generated by an L-system



The double Sierpinski triangle



Branching structures as the result of tensor product surface rewriting

Kolingerová I., Märtz P., Beneš B.: Tensor Product Surfaces as Rewriting Process, Proceedings of the Spring Conference on Computer Graphics, Častá-Papiernička, Slovakia, 2006, pp.107-112