CAD & Computational Geometry

Outline

- Interpolation and polynomial approximation
- Interpolation
  - Lagrange
  - Cubic Splines

Approximation

  - Bézier curves
  - B-Splines

B-Splines

- Three useful references :

  R. Bartels, J.C. Beatty, B. A. Barsky, An introduction to Splines for use in Computer Graphics and Geometric Modeling, Morgan Kaufmann Publications,1987

  JC.Léon, Modélisation et construction de surfaces pour la CFAO, Hermes, 1991

  L. Piegl, W. Tiller, The NURBS Book, Second Edition, Springer , 1996

B-Splines

- Isaac J. Schoenberg (1946)
- Carl De Boor (1972-76)
- Maurice G. Cox (1972)
- Richard Riesenfeld (1973)
- Wolfgang Boehm (1980)

B-Splines

- For Bézier curves, the polynomial degree is directly related to the number of control points.
  - The control of the continuity between Bézier curves is not trivial
- B-Splines are a generalization in the sense that the degree doesn't depend on the number of control points
  - One can impose every continuity at any point of the curve (we will see later how to do that)
  - They are polynomial curves, by pieces (Bézier curves have a unique polynomial representation along the interval of definition)
  - They may provide local control
  - The parametrization can be freely chosen (with Bézier, it is fixed , usually $0<u<1$.)

B-Splines

- Basis of Bézier curves :

$$P(u)=\sum_{i=0}^{d} P_i B_i^d(u)$$

  - The support of the basis functions is the interval $[0..1]$
  - Continuity is C_∞ , and between different Bézier curves it is enforced by a wise choice of the P_i's

- B-splines basis

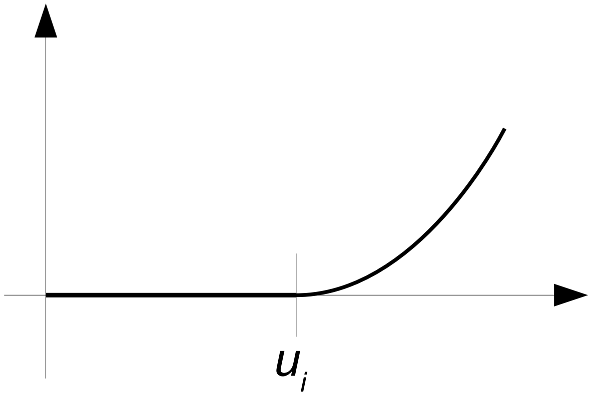
$$P(u)=\sum_{i=0}^{n} P_i N_i^d(u)$$

  - The basis functions N_i^d are piecewise polynomials
  - Have a *compact support* + satisfy partition of the unity
  - The continuity is defined at the basis function's level.

- B-spline basis functions
  - Defined by the nodal sequence and by the polynomials degree of the curve (d)
  - There are $n+1$ such functions, indexed from 0 to n .
- Nodal sequence:
  - It is a series of values u_i (knots) of the parameter u of the curve, not strictly increasing
  - There are $m+1$ such knots, indexed from 0 to m
  - e.g. $U=\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
    $U=\{0, 0, 0, 1, 2, 3, 4, 5, 5, 5\}$
    $U=\{0, 0, 0, 1, 2, 2, 3, 4, 4, 4\}$

- Construction of B-Spline basis functions
  - Truncated Power Function

$$(u-u_i)_+^d = \begin{cases} (u-u_i)^d & \text{if} \quad u \geq u_i \\ 0 & \text{otherwise} \end{cases}$$



  - It is a function of $C^{d\text{-}1}$ continuity

B-Splines

- Divided differences
  - order one (similar to a simple derivative)

$$[u_i, u_{i+1}]_U f(U) = \frac{f(u_{i+1}) - f(u_i)}{u_{i+1} - u_i}$$

U is a hidden parameter (like a variable used to differentiate)

  - order 2 : application of the above formula twice...

$$[u_i, u_{i+1}, u_{i+2}]_U f(U) = \frac{[u_{i+1}, u_{i+2}]_U f(U) - [u_i, u_{i+1}]_U f(U)}{u_{i+2} - u_i}$$

$$[u_i, u_{i+1}, u_{i+2}]_U f(U) = \frac{\dfrac{f(u_{i+2}) - f(u_{i+1})}{u_{i+2} - u_{i+1}} - \dfrac{f(u_{i+1}) - f(u_i)}{u_{i+1} - u_i}}{u_{i+2} - u_i}$$

- At the order k

$$[u_i,\cdots,u_{i+k}]f=\frac{[u_{i+1},\cdots,u_{i+k}]f-[u_i,\cdots,u_{i+k-1}]f}{u_{i+k}-u_i}$$

- One assumed that $u_i\neq u_{i+1}\neq u_{i+2}\cdots$

- Properties (see Bartels, 1987)

1- In the case where $u_i=u_{i+1}=u_{i+2}\cdots$

$$[u_i,\cdots,u_{i+k}]f=\frac{1}{k!}\left.\frac{d^k f}{du^k}\right|_{u=u_i}$$

2- if $u_i\neq u_{i+1}\neq u_{i+2}\cdots$ and $u_i<u_{i+1}<u_{i+2}\cdots$

$$[u_i,\cdots,u_{i+k}]f=\frac{1}{k!}\left.\frac{d^k f}{du^k}\right|_{u=u^*},u_i<u^*<u_{i+k}$$

B-Splines

3- $[u_i, \cdots, u_{i+k}]f$ is symmetric with respect to the knot vector

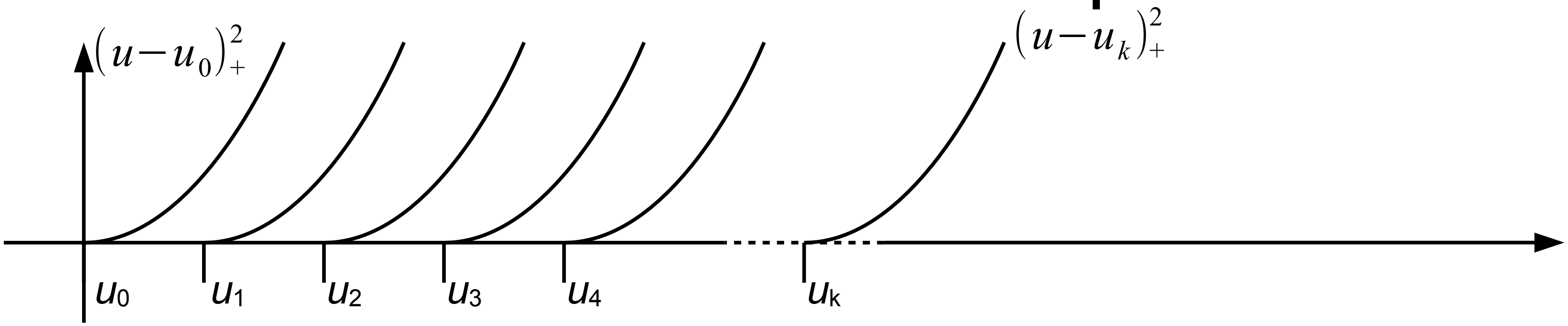
4- If $f(u)$ is a polynomial of degree at the most equal to k , then $[u_i, \cdots, u_{i+k}]f$ is a constant with respect to the u_i.

5- The divided difference of f=$g(u).h(u)$ is :

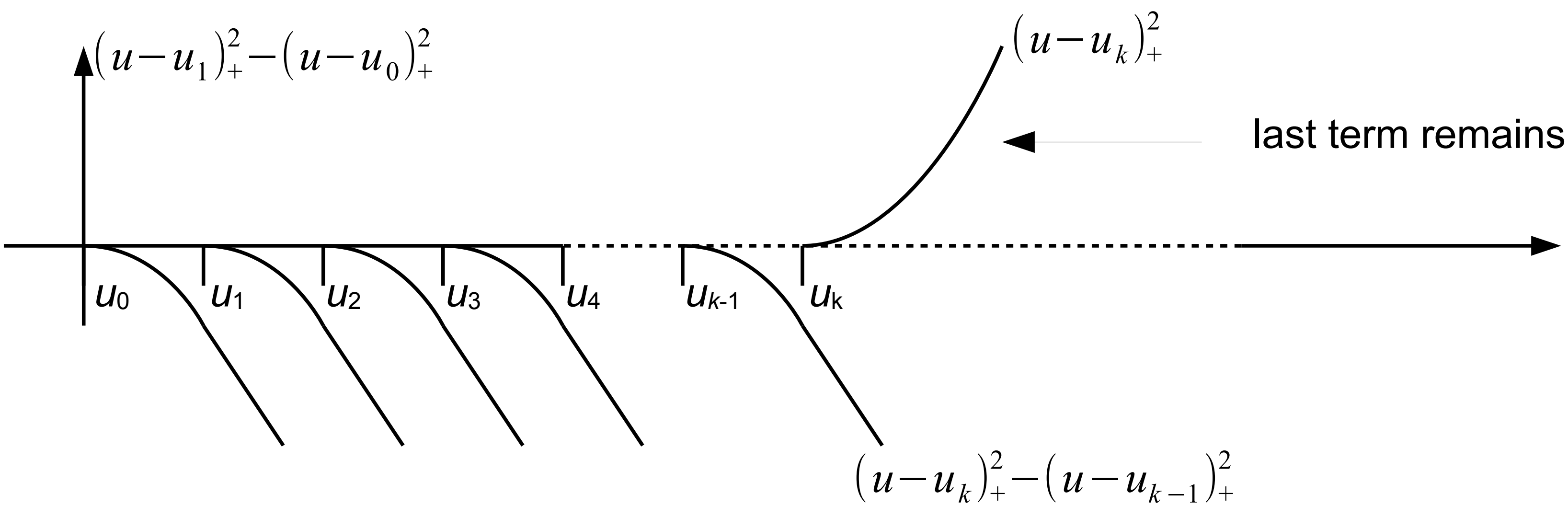
$$[u_i, \cdots, u_{i+k}]f = \sum_{j=i}^{j=i+k} ([u_i, \cdots, u_j]g) \cdot ([u_j, \cdots, u_{i+k}]h)$$

B-Splines

- Divided differences and B-Splines



  - How to cancel quadratic terms ?
    → subtract adjacent truncated power functions.

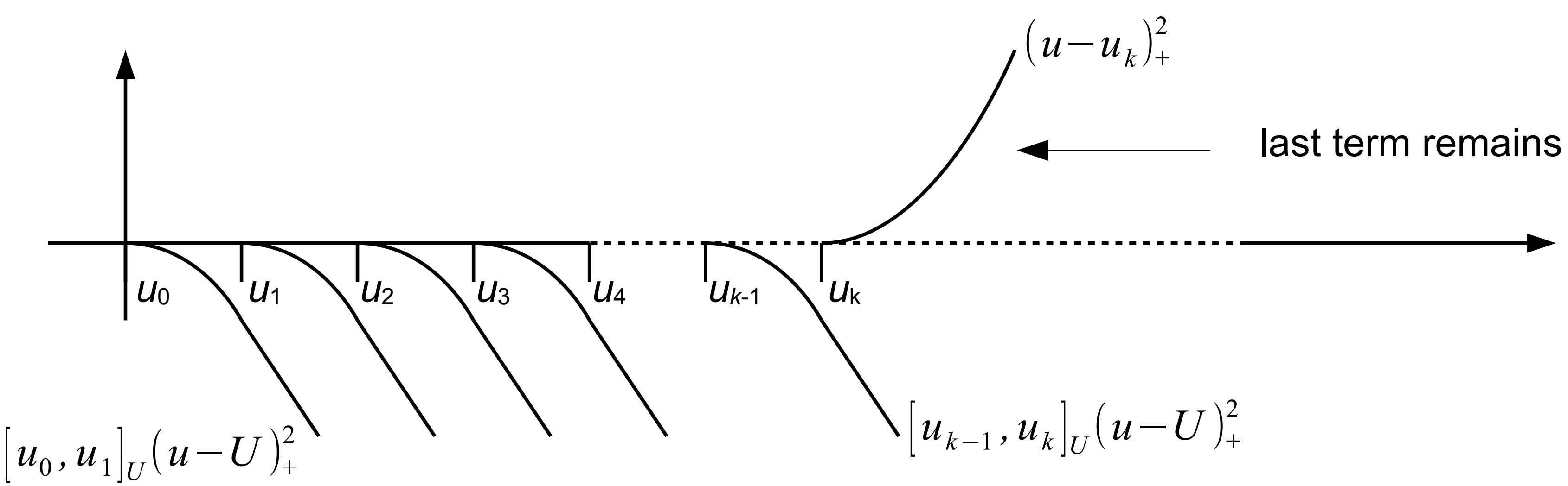


- Problem, lower order terms are dependent on k

$$(u-u_k)_+^2-(u-u_{k-1})_+^2\Big|_{u>u_k}=0\cdot u^2+(u_k-u_{k-1})\cdot u+(u_k-u_{k-1})(u_k+u_{k-1})\cdot 1$$

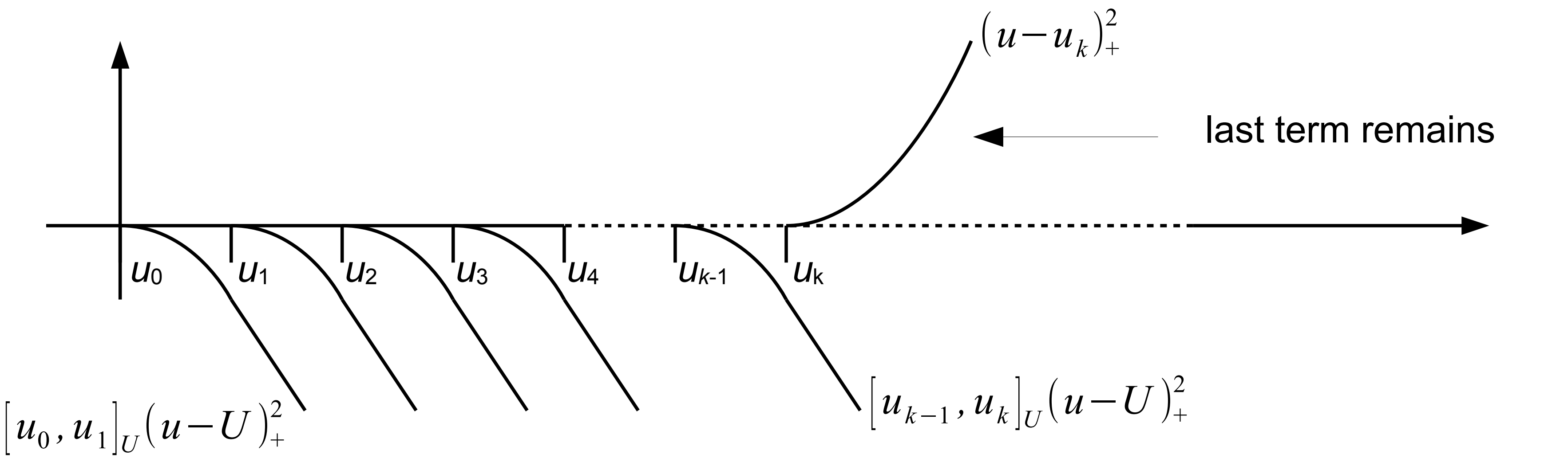
But, dividing by (u_k-u_{k-1}) yields a divided difference :

$$\frac{(u-u_k)_+^2-(u-u_{k-1})_+^2}{u_k-u_{k-1}}=[u_{k-1},u_k]_U(u-U)_+^2$$

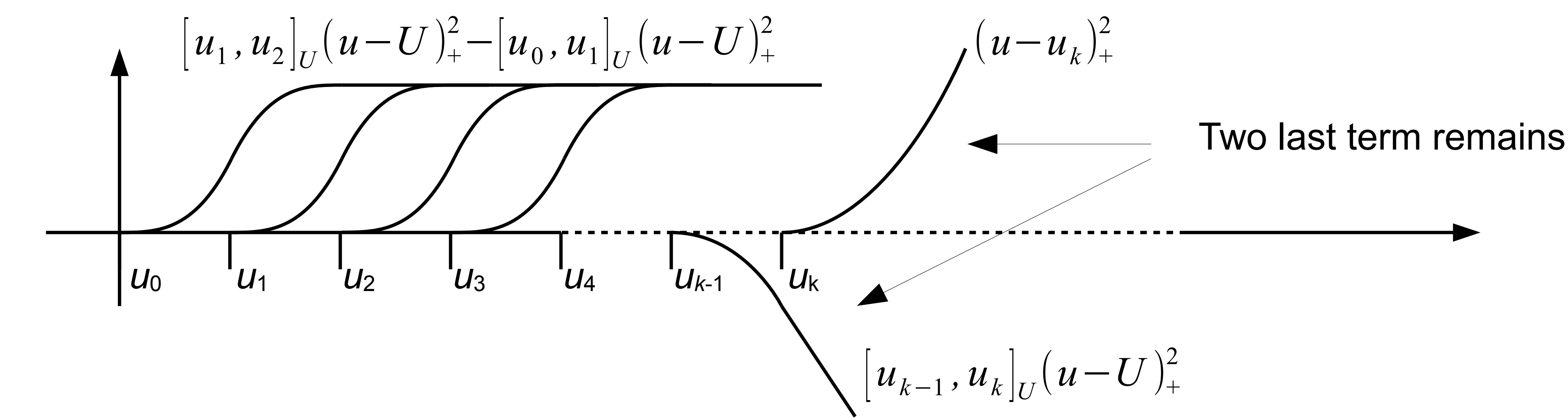


B-Splines

- Now, cancel linear terms …



- Same procedure : subtract adjacent terms.



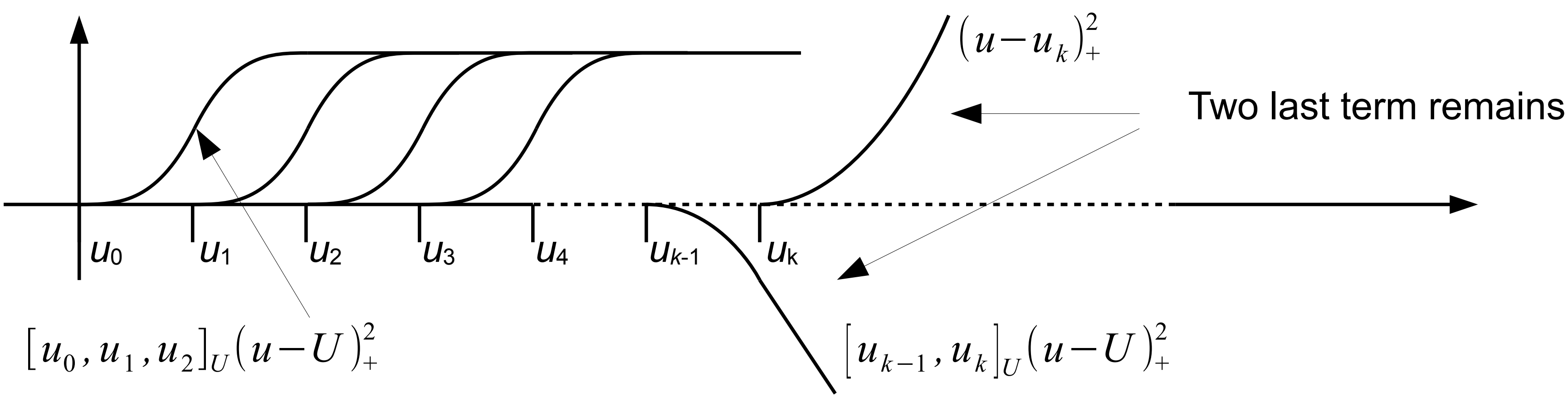
B-Splines

- Again, lower order terms are dependent on k

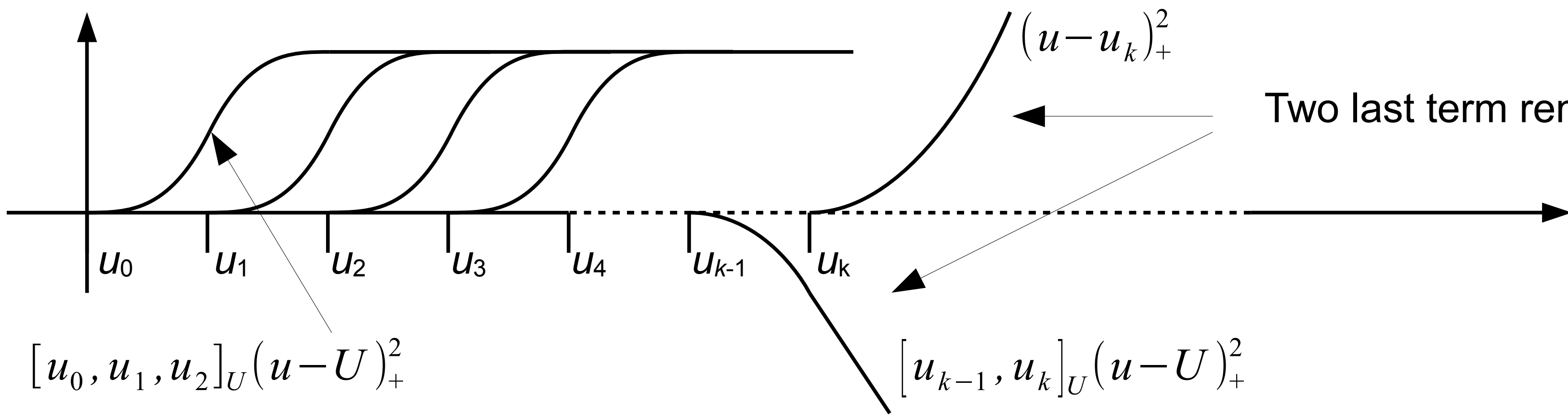
$$\left.[u_{k-1},u_{k-2}]_U(u-U)_+^2-[u_k,u_{k-1}]_U(u-U)_+^2\right|_{u>u_k}$$
$$=0\cdot u+((u_k+u_{k-1})-(u_{k-1}+u_{k-2}))\cdot 1=(u_k-u_{k-2})\cdot 1$$

Dividing by (u_k-u_{k-2}) yields a divided difference again :

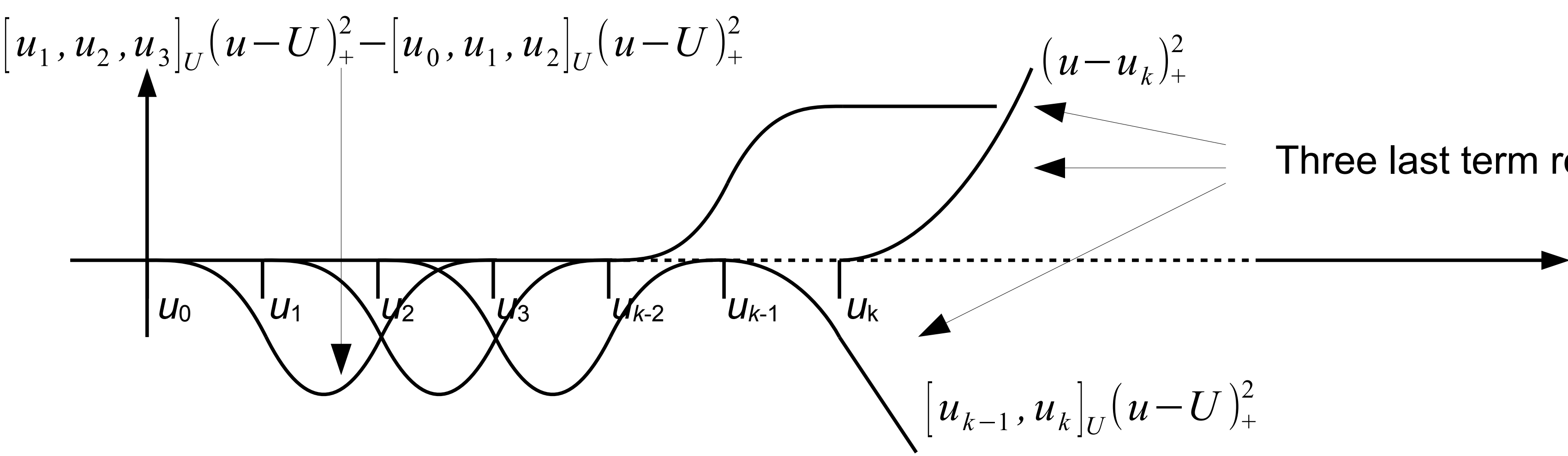
$$\frac{[u_{k-1},u_{k-2}]_U(u-U)_+^2-[u_k,u_{k-1}]_U(u-U)_+^2}{u_k-u_{k-2}}$$
$$=[u_{k-2},u_{k-1},u_k]_U(u-U)_+^2$$



- Now, cancel constant terms …



- Same procedure : subtract adjacent terms.

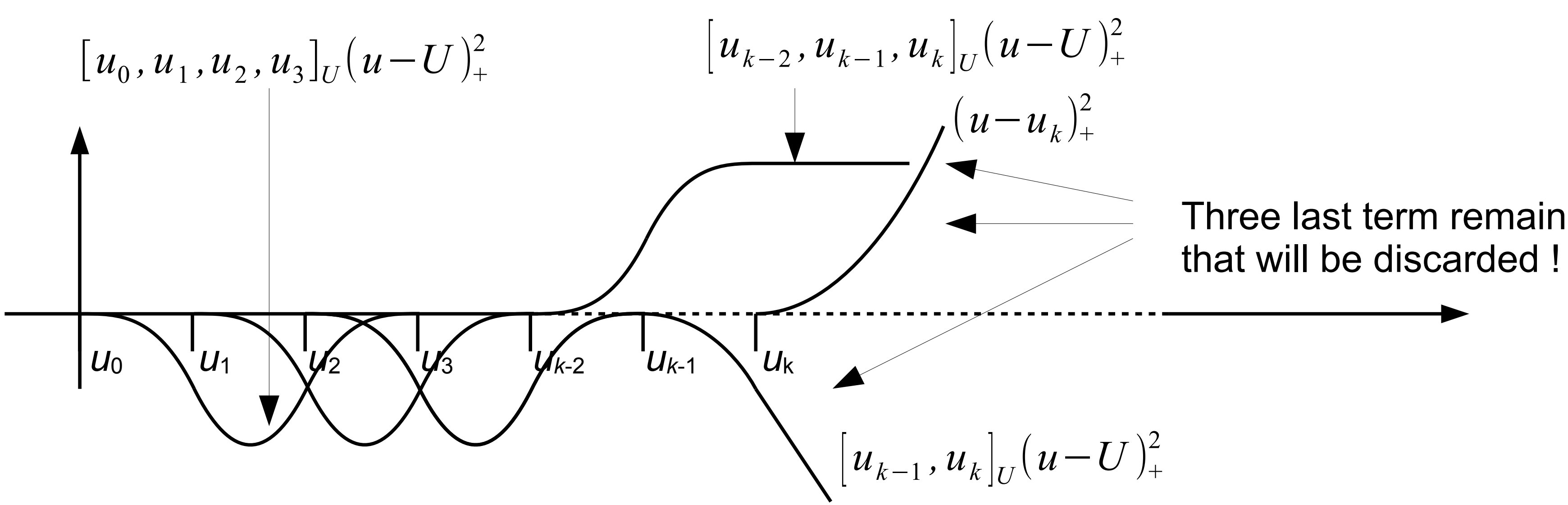


B-Splines

- There are no lower order terms. However we might divide anyway by $\left(u_k - u_{k-3}\right)$ to remain consistent and get an expression as a divided difference again...

$$\frac{\left[u_{k-2}, u_{k-1}, u_k\right]_U \left(u-U\right)_+^2 - \left[u_{k-3}, u_{k-2}, u_{k-1}\right]_U \left(u-U\right)_+^2}{u_k - u_{k-3}}$$

$$= \left[u_{k-3}, u_{k-2}, u_{k-1}, u_k\right]_U \left(u-U\right)_+^2$$



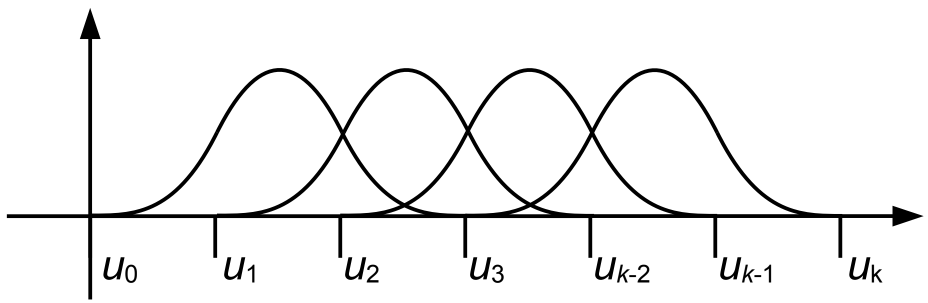
B-Splines

- The sign is alternating with the degree. Shape functions of even degree are negative, while SFs of uneven degree are positive.
- Multiplying by $(-1)^{d+1}$ makes every SF positive.
- To ensure that the SF form a partition of unity , we have to multiply again by $(u_{i+d+1} - u_i)$
- The compact representation of the B-Splines basis functions of degree d with the use of divided differences is therefore :

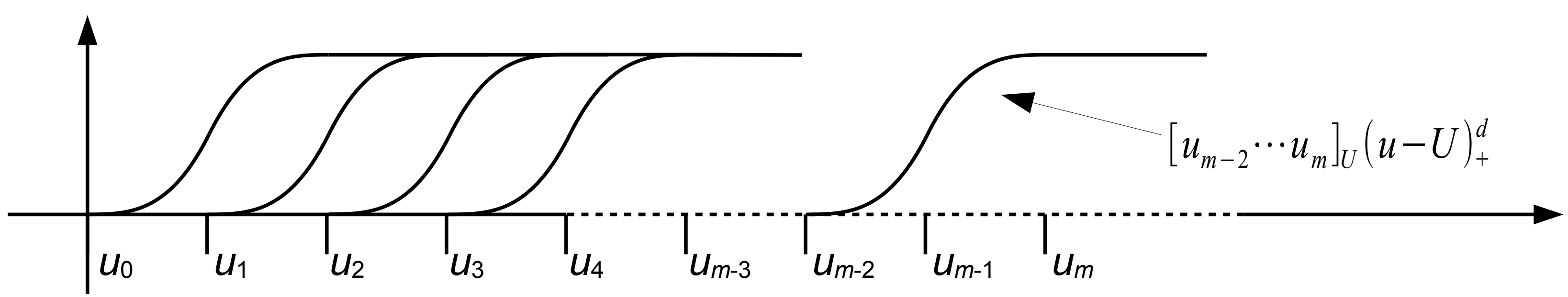
$$N_i^d = (-1)^{d+1} (u_{i+d+1} - u_i) [u_i, \cdots, u_{i+d+1}]_U (u - U)_+^d$$

B-Splines

$$N_i^d = (-1)^{d+1}(u_{i+d+1} - u_i)[u_i, \cdots, u_{i+d+1}]_U (u - U)_+^d$$

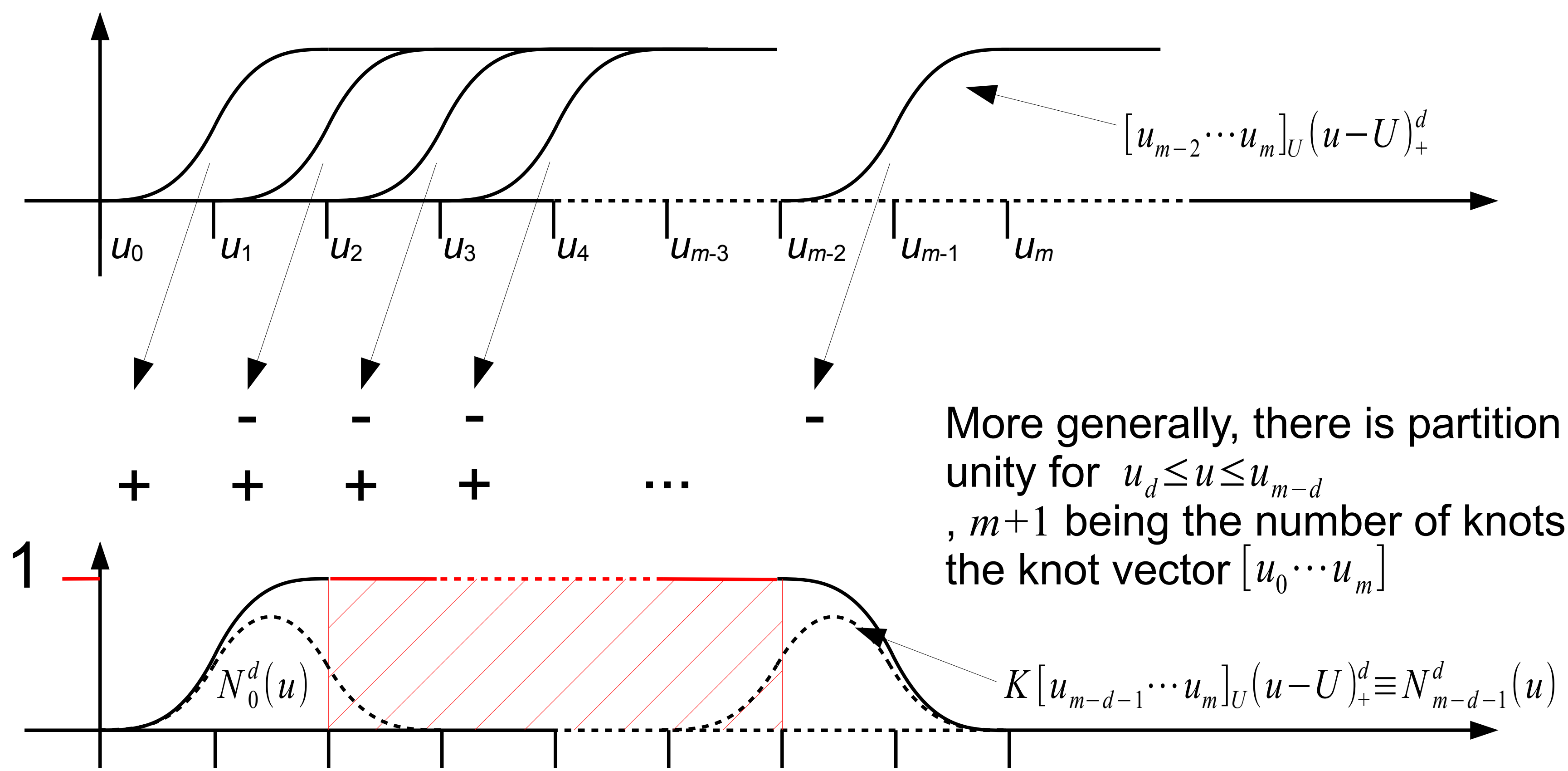


- Proof of the partition of unity : consider the before last operation (the cancellation of constant terms)
  - We subtract consecutive terms to form the final shape functions
  - Partition of unity means the sum of all the final shape functions is equal to 1… that this is indeed the case only on a certain range of u.



$$N_i^d = (-1)^{d+1} (u_{i+d+1} - u_i) [u_i, \cdots, u_{i+d+1}]_U (u-U)_+^d$$

B-Splines



More generally, there is partition of unity for $u_d \le u \le u_{m-d}$, $m+1$ being the number of knots in the knot vector $[u_0 \cdots u_m]$

$$N_i^d = (-1)^{d+1}(u_{i+d+1} - u_i)[u_i, \cdots, u_{i+d+1}]_U (u-U)_+^d$$

- Recursive definition of basis functions
  - Setting $U=\{u_0,\cdots,u_m\}$, $u_i\leq u_{i+1}$, $i=0\cdots m-1$ (nodal sequence)
  - The functions are such as : (recurrence formula of Cox – de Boor)

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

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

  - Where $u_{i+d}-u_i=0$, necessarily $N_i^{d-1}(u)\equiv 0$

    By convention, we set in this case $\frac{0}{0}=0$ when the limit is undefined.

- Example : computation of basis functions of degree $d \leq 2$ for $U = \{u_0 = 0, u_1 = 0, u_2 = 0, u_3 = 1, u_4 = 1, u_5 = 1\}$

$$N_0^1(u) = \frac{u - u_0}{u_1 - u_0} N_0^0(u) + \frac{u_2 - u}{u_2 - u_1} N_1^0(u) \qquad \frac{0}{0} = 0$$

by convention

$$N_0^0 = 0 \qquad N_1^0 = 0 \qquad N_2^0 = \begin{cases} 1 & 0 \leq u < 1 \\ 0 & \text{otherwise} \end{cases} \qquad N_3^0 = 0 \qquad N_4^0 = 0$$

$$N_0^1 = 0 \qquad N_1^1 = \begin{cases} 1 - u & 0 \leq u < 1 \\ 0 & \text{otherwise} \end{cases} \qquad N_2^1 = \begin{cases} u & 0 \leq u < 1 \\ 0 & \text{otherwise} \end{cases} \qquad N_3^1 = 0$$

$$N_0^2 = \begin{cases} (1 - u)^2 & 0 \leq u < 1 \\ 0 & \text{otherwise} \end{cases}$$

$$N_1^2 = \begin{cases} 2u(1 - u) & 0 \leq u < 1 \\ 0 & \text{otherwise} \end{cases}$$

$$N_2^2 = \begin{cases} u^2 & 0 \leq u < 1 \\ 0 & \text{otherwise} \end{cases}$$

Bernstein polynomials of degree 2

B-Splines

- The Bernstein polynomials of degree d are a particular case of the B-splines basis
  - They correspond to a nodal sequence

$$U_B = \{u_0 = 0, \cdots, u_d = 0, u_{d+1} = 1, \cdots, u_{2d+1} = 1\}$$

  - Bézier curves are therefore a particular case of B-splines.
  - It is also possible to transform any B-spline into a sequence of Bézier curves – because the Bernstein polynomials form a complete basis of polynomials of degree d.

- Basis functions and control points
  - In contrary to Bézier curves, the number of control points is not imposed by the degree d
  - Let $m+1$ the number of knots. We have $n+1$ independent basis functions at our hands
    - For every basis function, we associate a control point

$$P(u) = \sum_{i=0}^{n} P_i N_i^d(u)$$

  - The number of control points is fixed by the relation $n+1=m-d$

- Types of nodal sequences...
  - Uniform – The gap is constant

    $$U=\{u_0, u_1, \cdots, u_{m-d-1}\} \quad , \quad u_{i+1}-u_i=k$$

  - Periodic - The gap between the knots at the start of a nodal sequence is identical to the one at the end of the nodal sequence

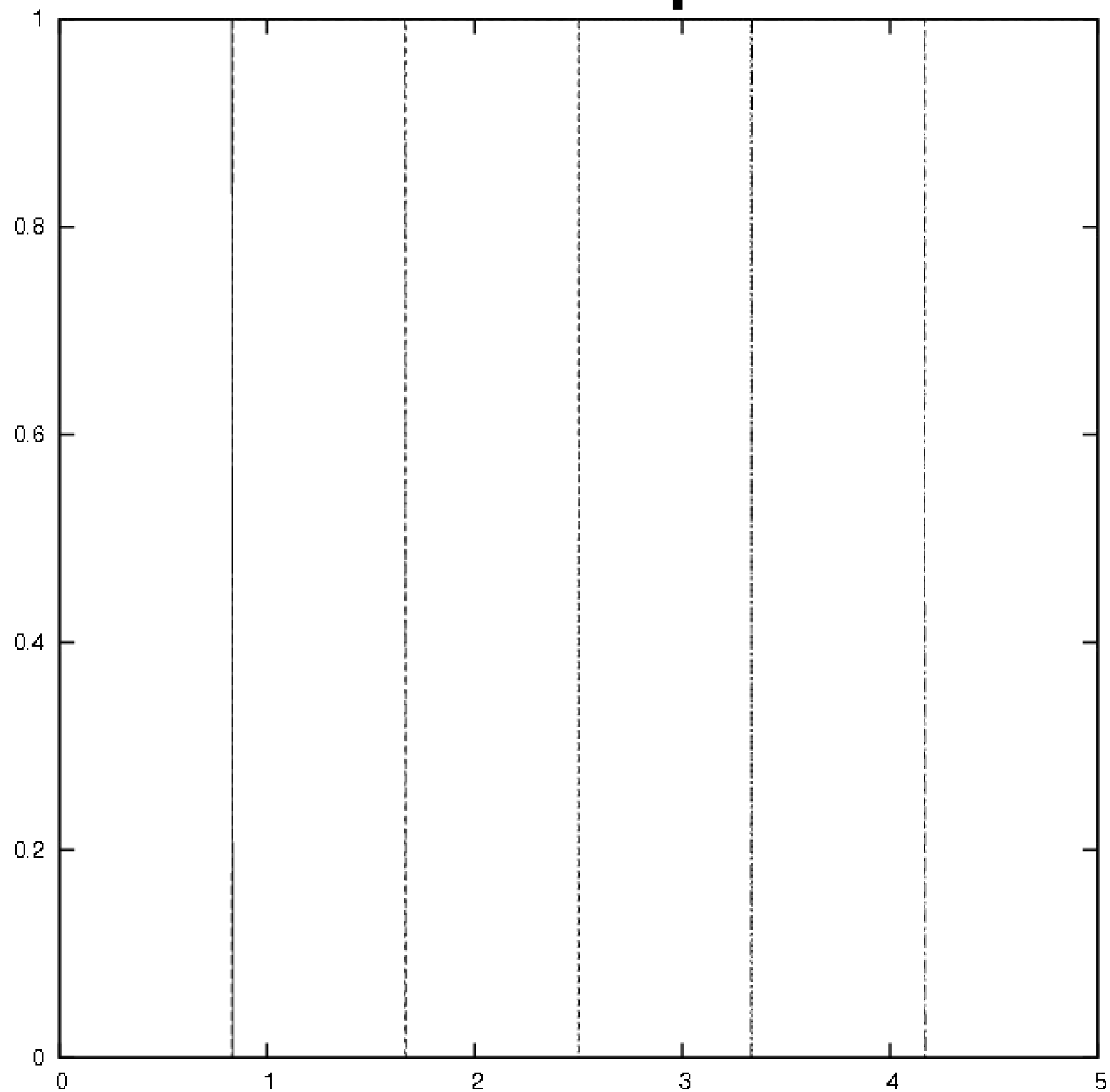
    $$U=\{\underbrace{u_{0,} \cdots, u_d}_{d+1}, u_{d+1}, \cdots, u_{m-d-1}, \underbrace{u'_{0,} \cdots, u'_d}_{d+1}\} \quad , \quad u'_i-u_i=k$$

  - Non uniform, interpolating – first and last CP are interpolated

    $$U=\{\underbrace{a, \cdots, a}_{d+1}, u_{d+1}, \cdots, u_{m-d-1}, \underbrace{b, \cdots, b}_{d+1}\}$$

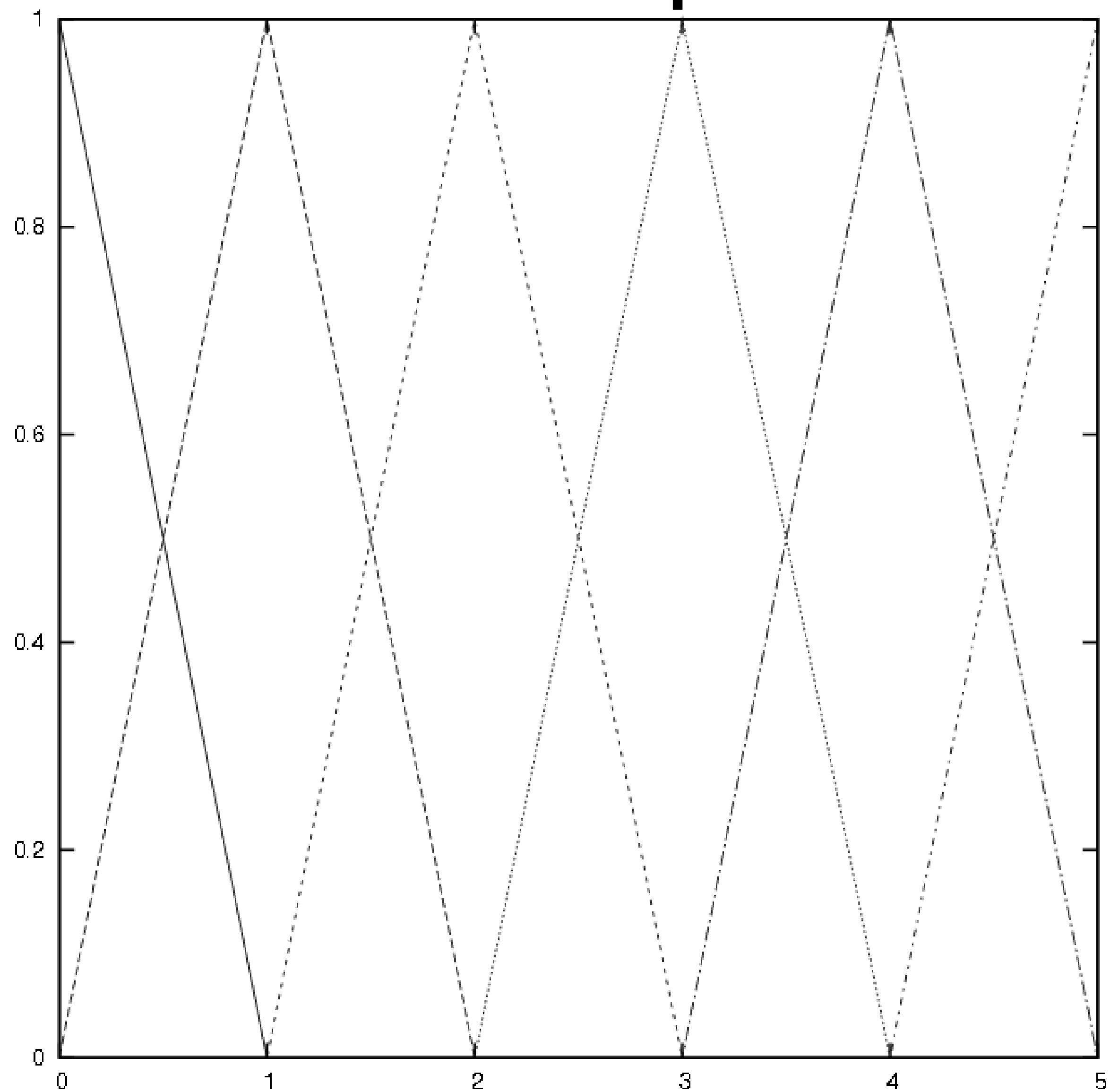
    In the sequel, except where indicated, we consider non uniform nodal sequences interpolating the first and last control points.

B-Splines



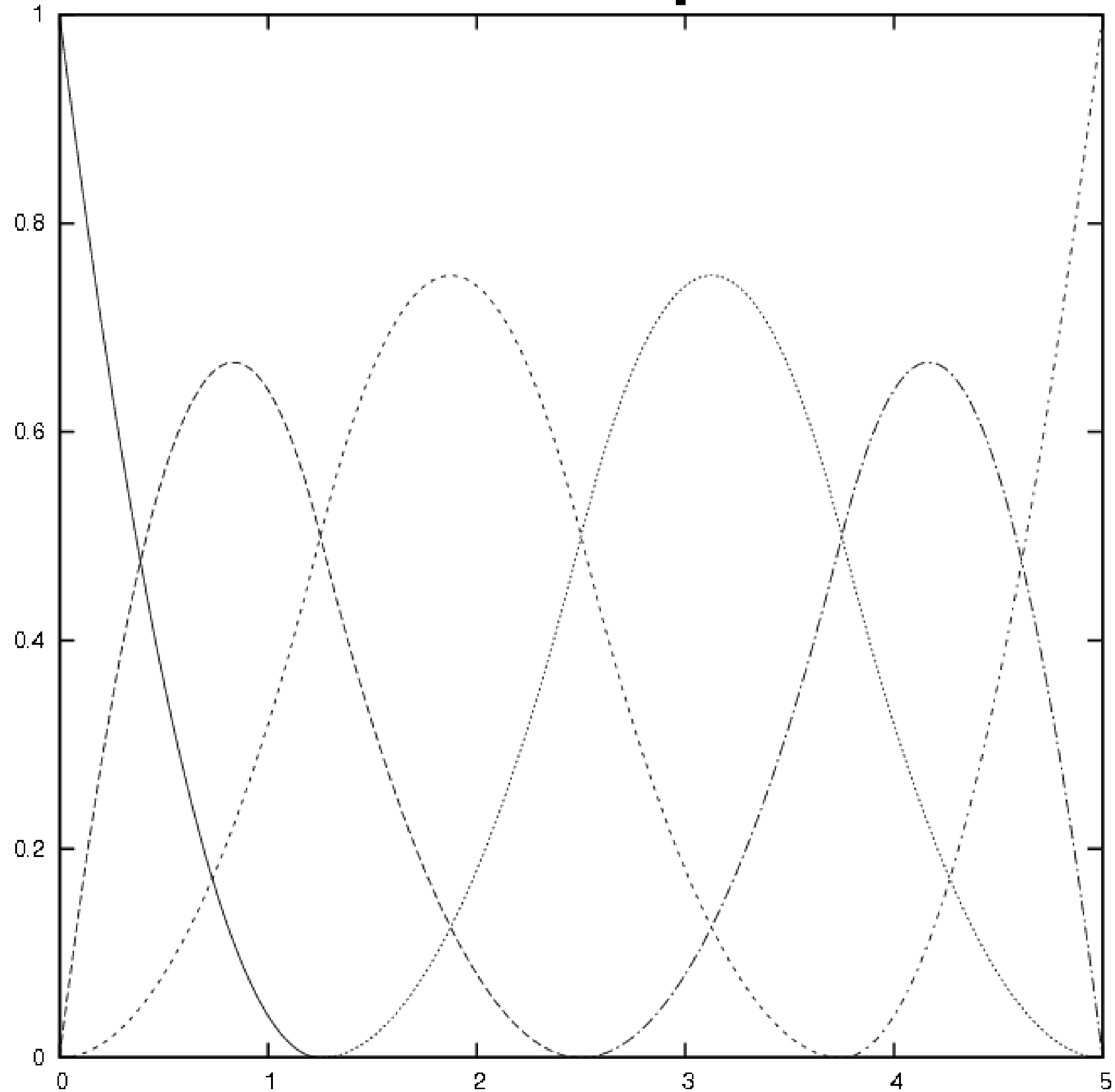
$$U=\left\{0,\frac{5}{6},\frac{10}{6},\frac{15}{6},\frac{20}{6},\frac{25}{6},5\right\}\quad d=0\quad m+1=7\quad n+1=6$$

B-Splines



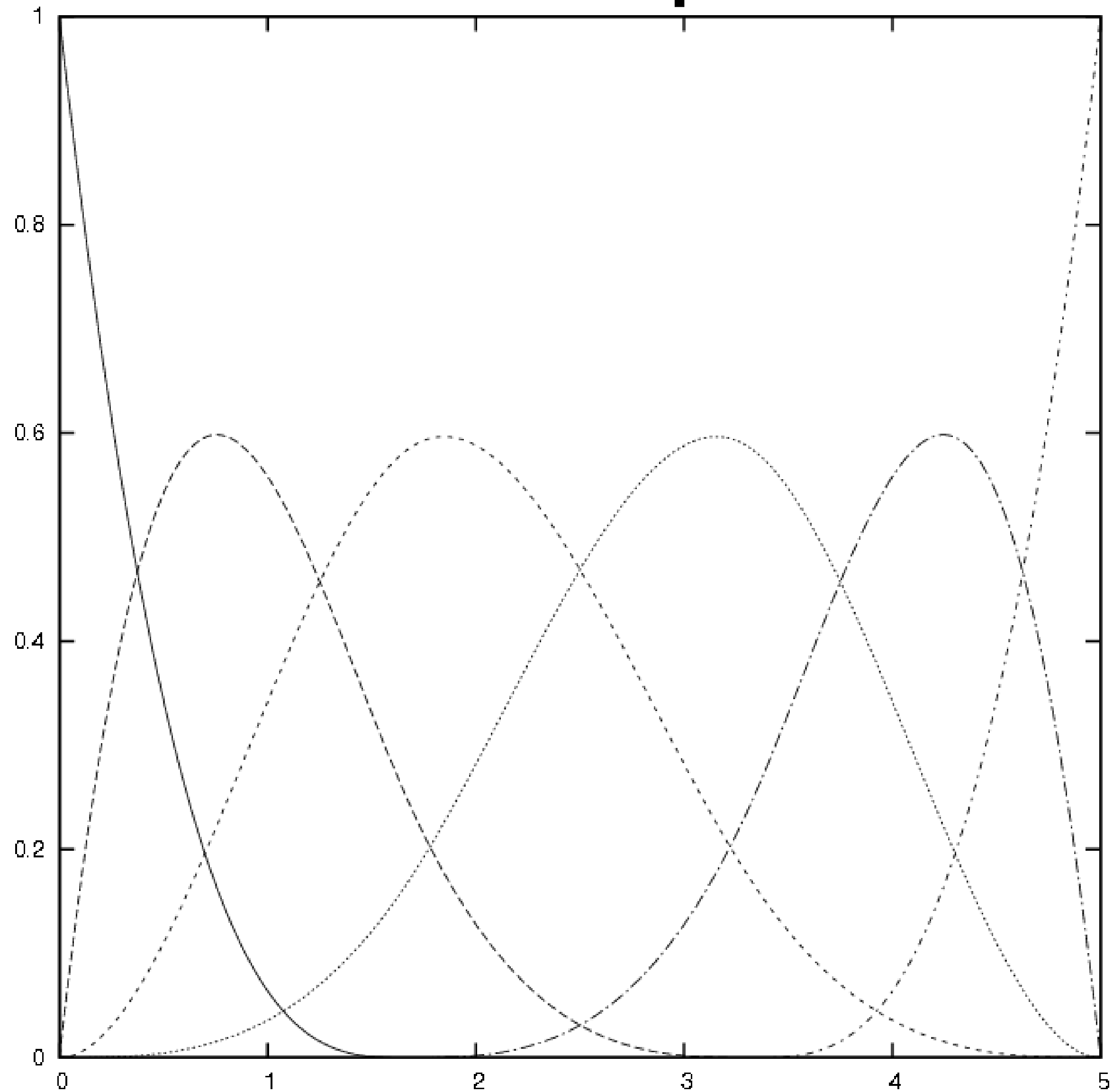
$$U=\{0,0,1,2,3,4,5,5\} \quad d=1 \quad m+1=8 \quad n+1=6$$

B-Splines



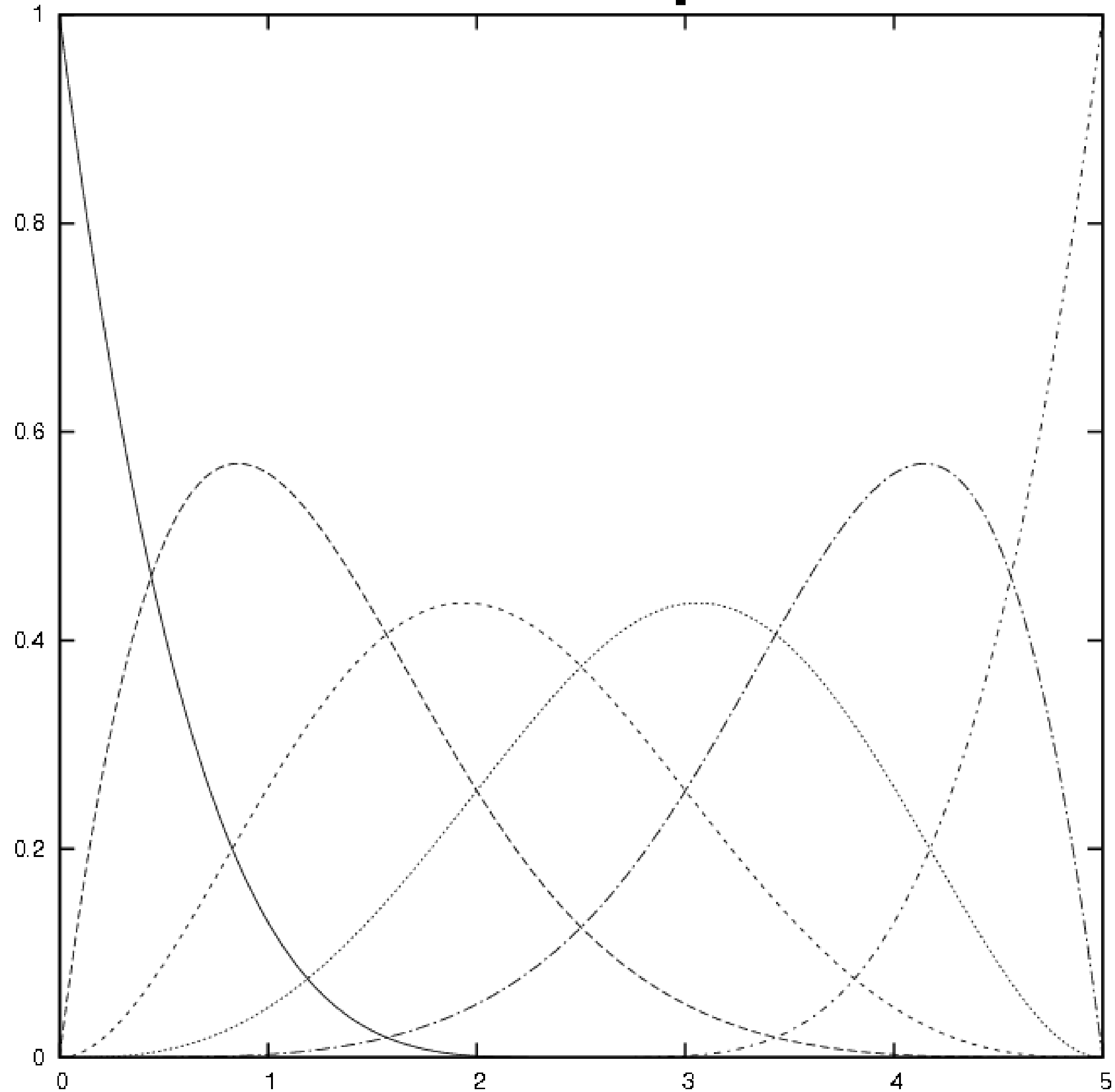
$$U=\left\{0,0,0,\frac{5}{4},\frac{10}{4},\frac{15}{4},5,5,5\right\}\quad d=2\quad m+1=9\quad n+1=6$$

B-Splines



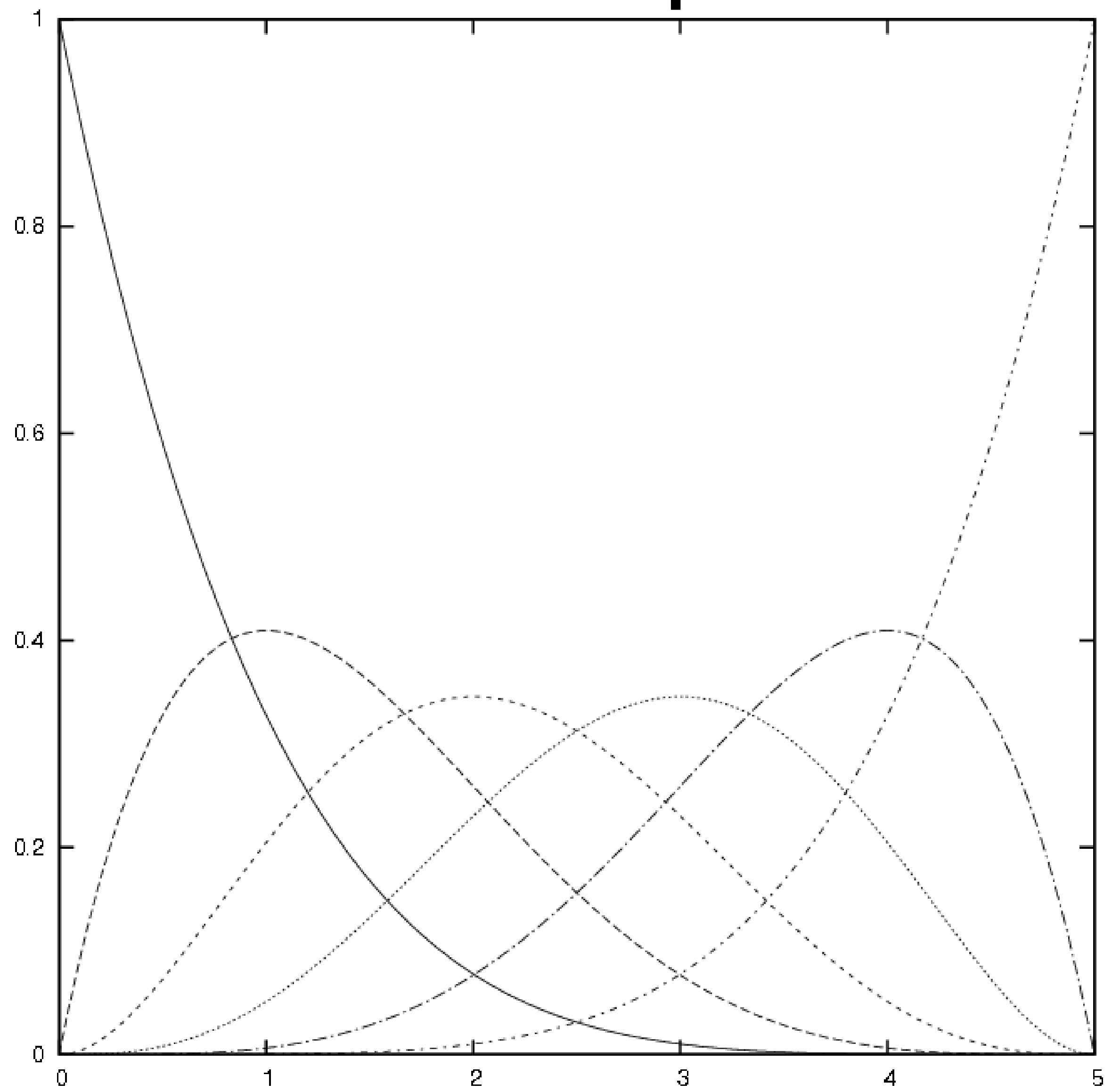
$$U=\{0,0,0,0,\frac{5}{3},\frac{10}{3},5,5,5,5\} \quad d=3 \quad m+1=10 \quad n+1=6$$

B-Splines



$$U=\left\{0,0,0,0,0,\frac{5}{2},5,5,5,5,5\right\} \quad d=4 \quad m+1=11 \quad n+1=6$$

B-Splines



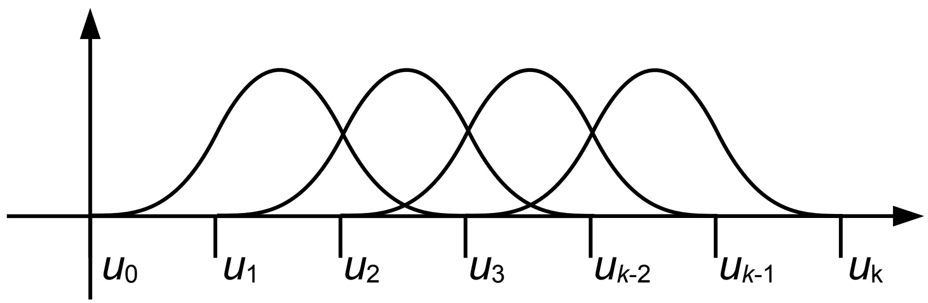
$$U = \{0,0,0,0,0,0,5,5,5,5,5,5\} \quad d=5 \quad m+1=12 \quad n+1=6$$

Bernstein polynomials (with a factor on *u*)

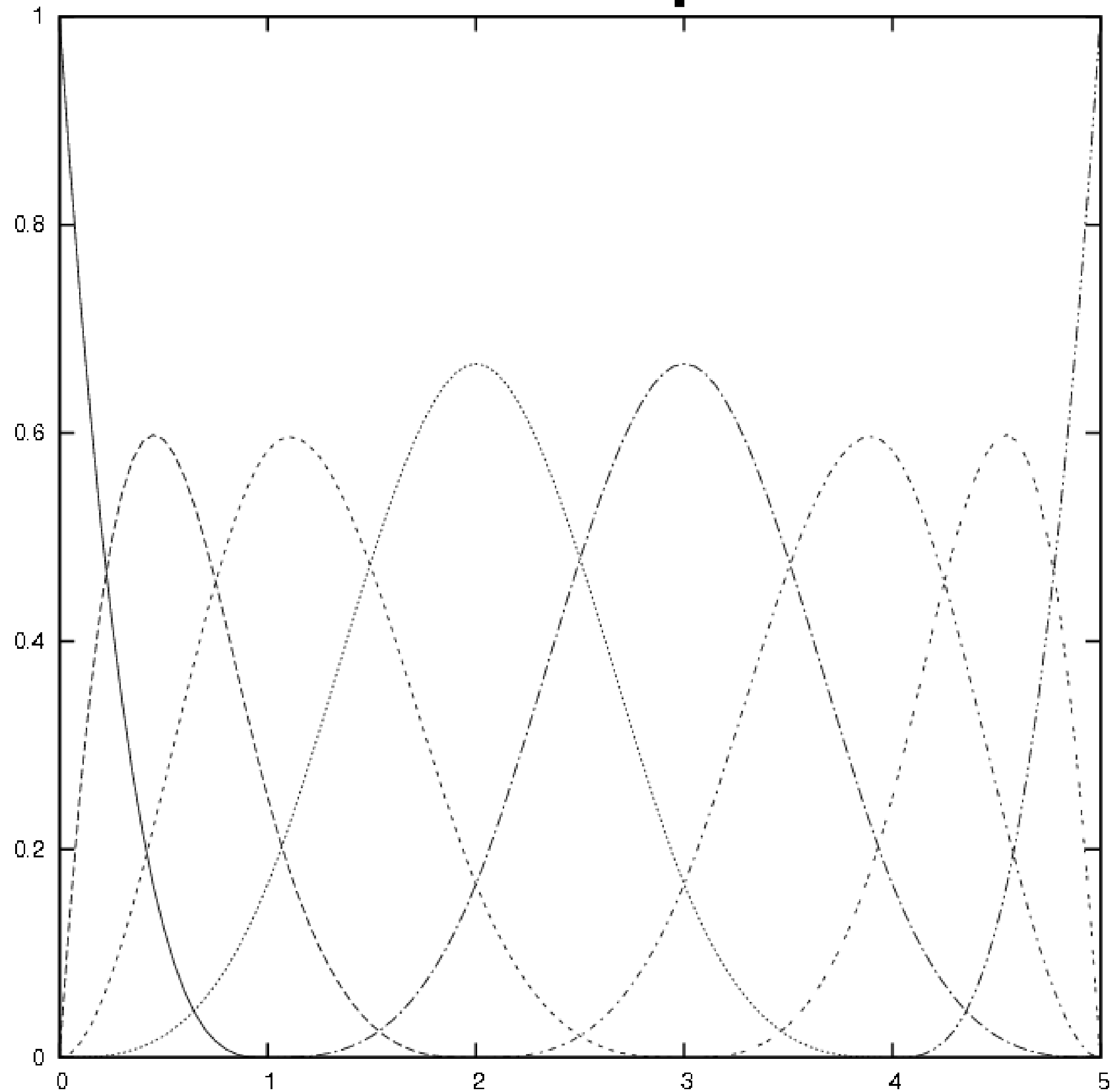
- Properties of B-spline basis functions
  - $N_i^d(u)=0$ outside the interval $[u_i, u_{i+d+1}[$
  - Inside the interval $[u_i, u_{i+1}[$, at most $d+1$ functions $N_*^d(u)$ are non zero : $N_{i-d}^d, \cdots, N_i^d$
  - $N_i^d(u) \geq 0 \quad \forall\, i, d$ and u (always positive)
  - For $u \in [u_i, u_{i+1}[$, $\sum_{j=i-d}^{i} N_j^d(u)=1$ (partition of unity)
  - All derivatives of $N_i^d(u)$ exist inside the interval $[u_i, u_{i+1}[$. At a knot , $N_i^d(u)$ is d-k times differentiable, k being the node multiplicity.
  - Except for d=0, $N_i^d(u)$ reaches exactly one maximum

B-Splines

$$N_i^d = (-1)^{d+1}(u_{i+d+1}-u_i)[u_i,\cdots,u_{i+d+1}]_U(u-U)_+^d$$



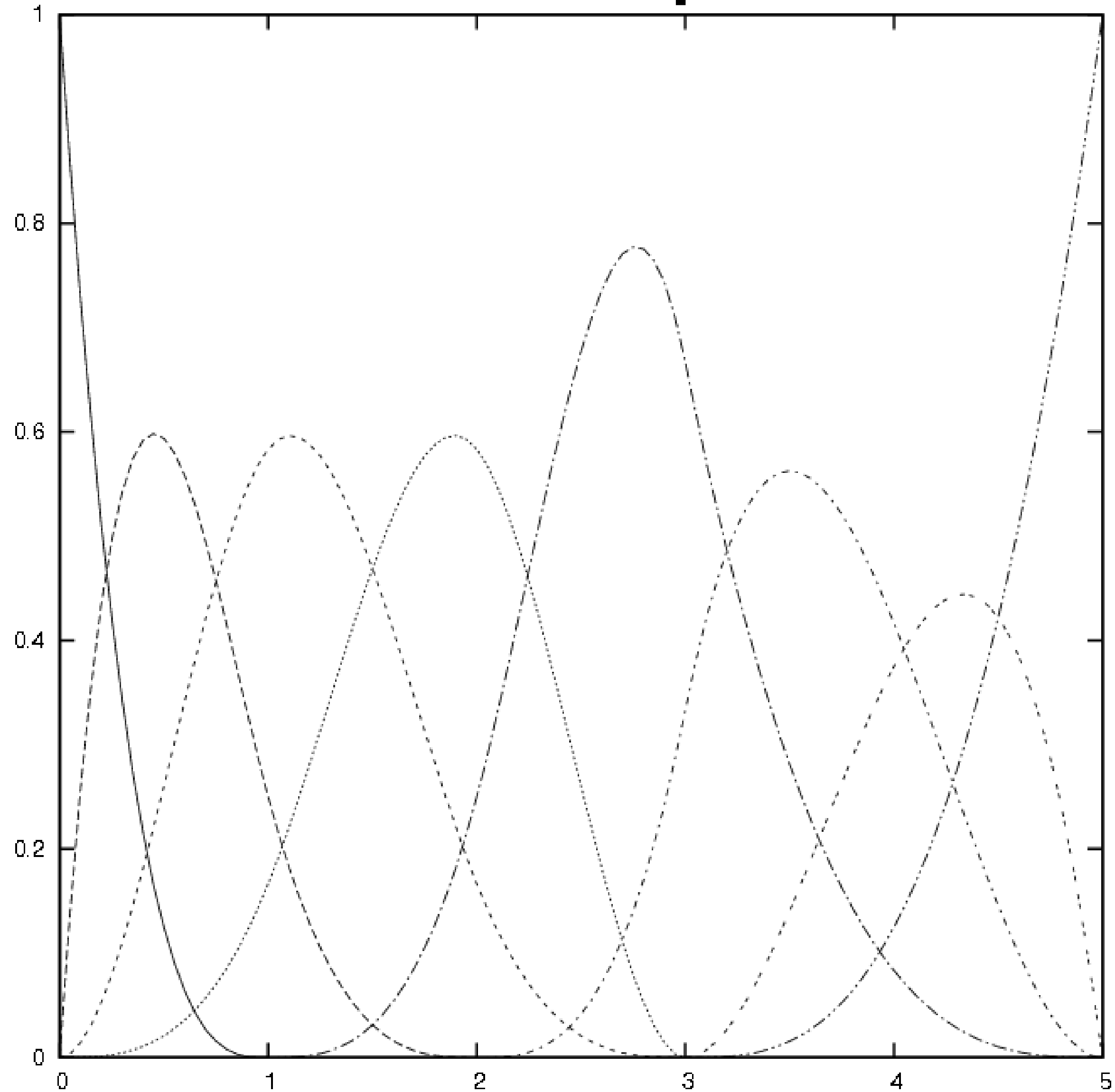
B-Splines



$$U = \{0,0,0,0,1,2,3,4,5,5,5,5\} \quad d=3 \quad m+1=12 \quad n+1=8$$

The knot u=3 is of multiplicity 1

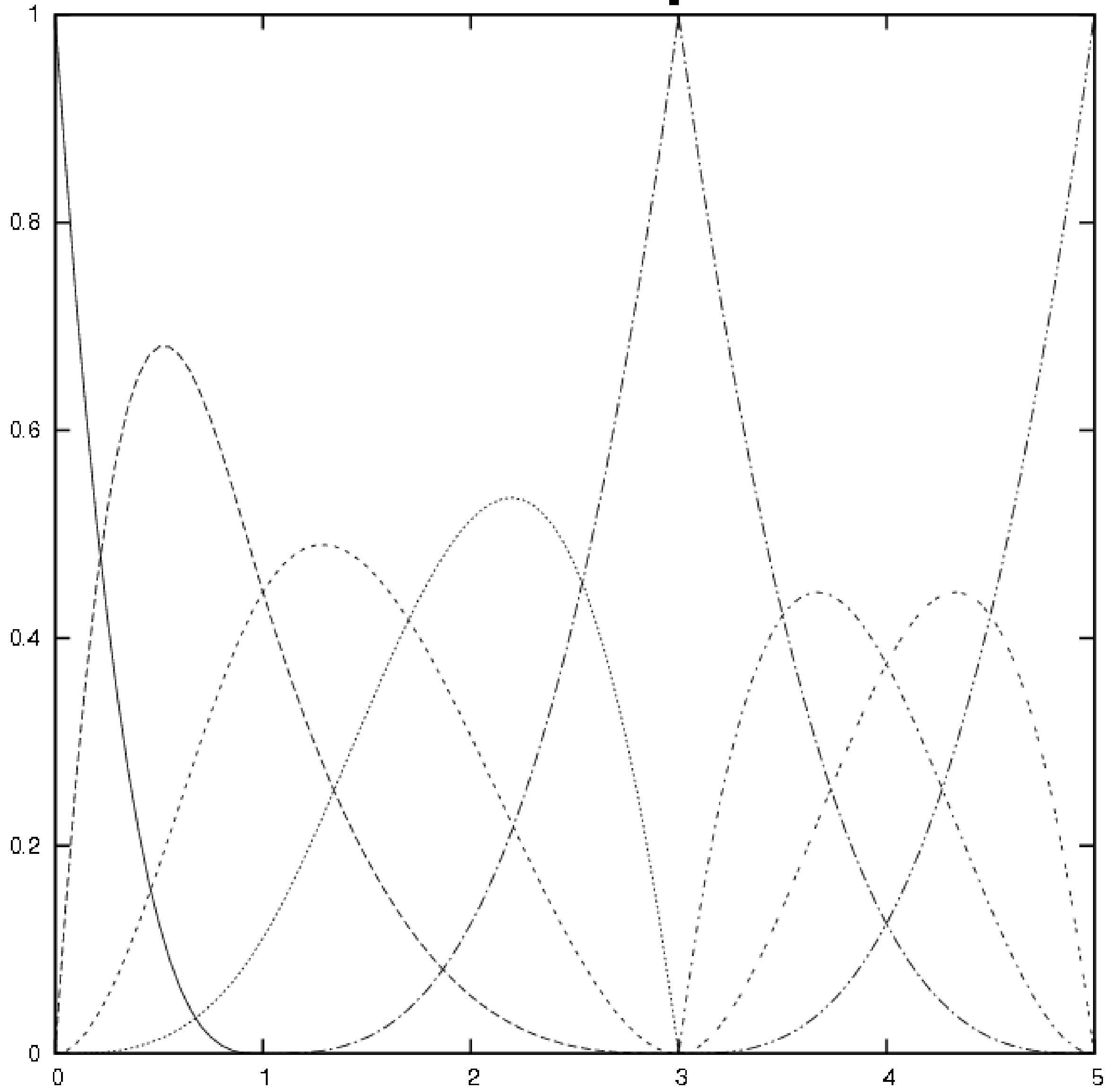
B-Splines



$$U=\{0,0,0,0,1,2,3,3,5,5,5,5\} \quad d=3 \quad m+1=12 \quad n+1=8$$

The node u=3 is of multiplicity 2

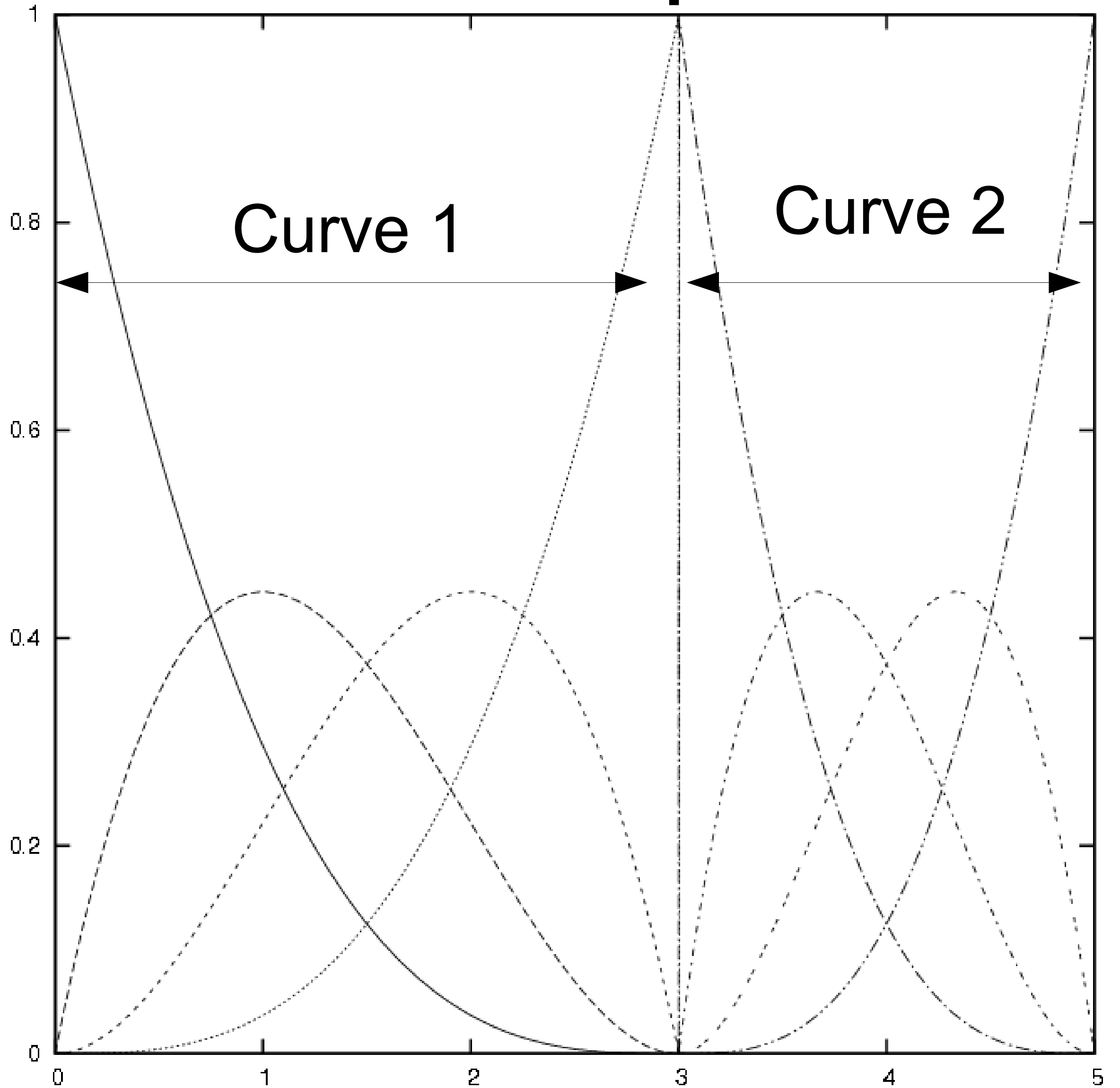
B-Splines



$$U=\{0,0,0,0,1,3,3,3,5,5,5,5\} \quad d=3 \quad m+1=12 \quad n+1=8$$

The node u=3 is of multiplicity 3

B-Splines



$$U = \{0, 0, 0, 0, 3, 3, 3, 3, 5, 5, 5, 5\} \quad d = 3 \quad m+1 = 12 \quad n+1 = 8$$

The node u=3 is of multiplicity 4

- Derivatives of B-spline basis functions
  - Definition by recurrence

$$\frac{d^k N_i^d}{d\,u^k} = N_i^{d,(k)} = d\left(\frac{N_i^{d-1,(k-1)}}{u_{i+d}-u_i} - \frac{N_{i+1}^{d-1,(k-1)}}{u_{i+d+1}-u_{i+1}}\right)$$

  - k should not exceed d : every derivative of higher order vanish.

- The characteristics of basis functions involve that the B-Spline curve

$$P(u)=\sum_{i=0}^{n} P_i N_i^d(u) \qquad U=\{u_0,\cdots,u_m\}\ ,\ u_i \le u_{i+1}\ ,\ i=0\cdots m-1$$

  - interpolates P_0 and P_n , **only if** the nodal sequence admits $d+1$ repetitions at the start and at the end !
  - is invariant by affine transformation ,
  - is contained by the convex hull of the control points (because $P(u)$ is a linear combination of the control points with positive coefficients which sum to one)

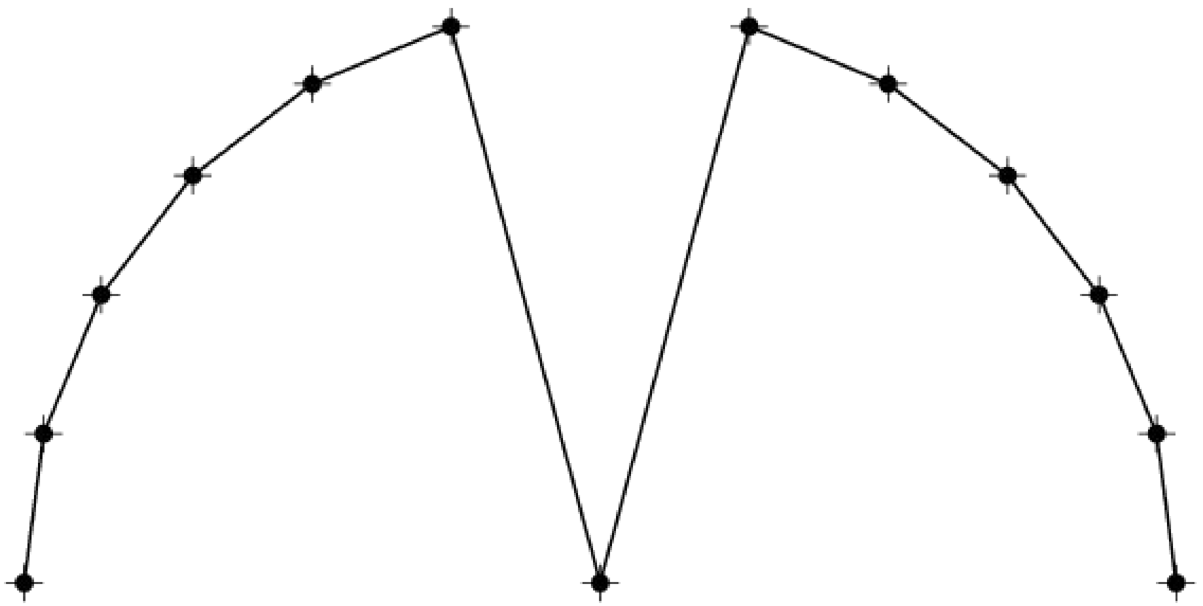
(Following)

- Is *variation diminishing* : The number of inflexion points is lower than the number of wiggles of the characteristic polygon
- Is closed and convex if the characteristic polygon is closed and convex,
- Is of length shorter or equal than that of the control polygon.
- Is invariant by linear transformation of the nodal sequence $u'=au+b$, $a>0$

- Control points, degree and nodal sequence
  - We associate a control point for each basis function N_i^* . We have $n+1$ control points.
  - The degree d is chosen by the user.
  - The nodal sequence (that defines the intervals of the parameter on which the curve has a unique polynomial definition) is then built. We have $m+1=n+d+2$ knots (with $d+1$ repetitions at the start and at the end)
    - there remains $n\text{-}d$ values of the parameter to set (without taking the boundaries into account)

- Geometric examples
  - Constant number of control points
  - We increase the degree
  - Uniform repartition of knots (except at boundaries)
  - For which degree do we have the best approximation of the control points ??

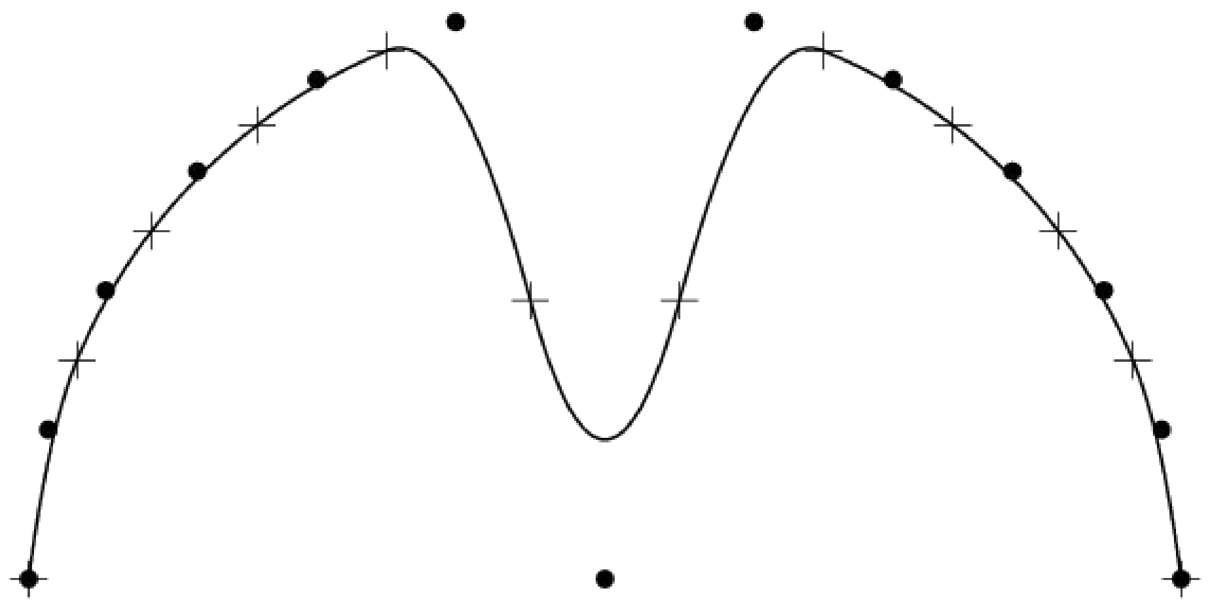
B-Splines



Degree 1

0 0 0.0833333 0.166667 0.25 0.333333 0.416667 0.5 0.583333 0.666667 0.75 0.833333 0.916667 1 1

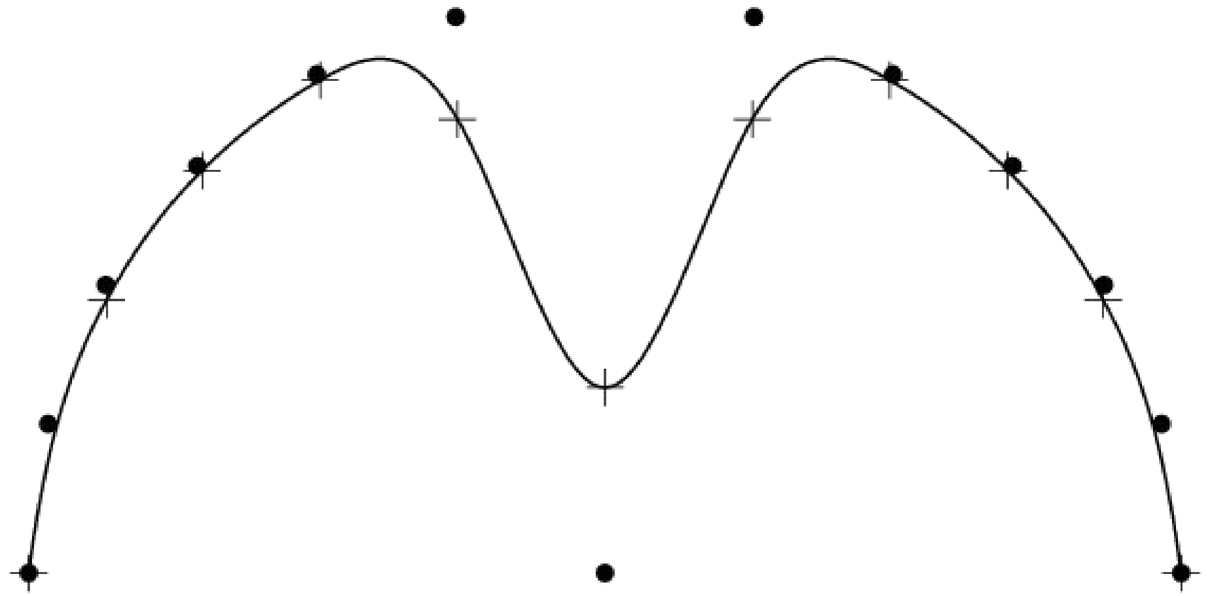
B-Splines



degree 2

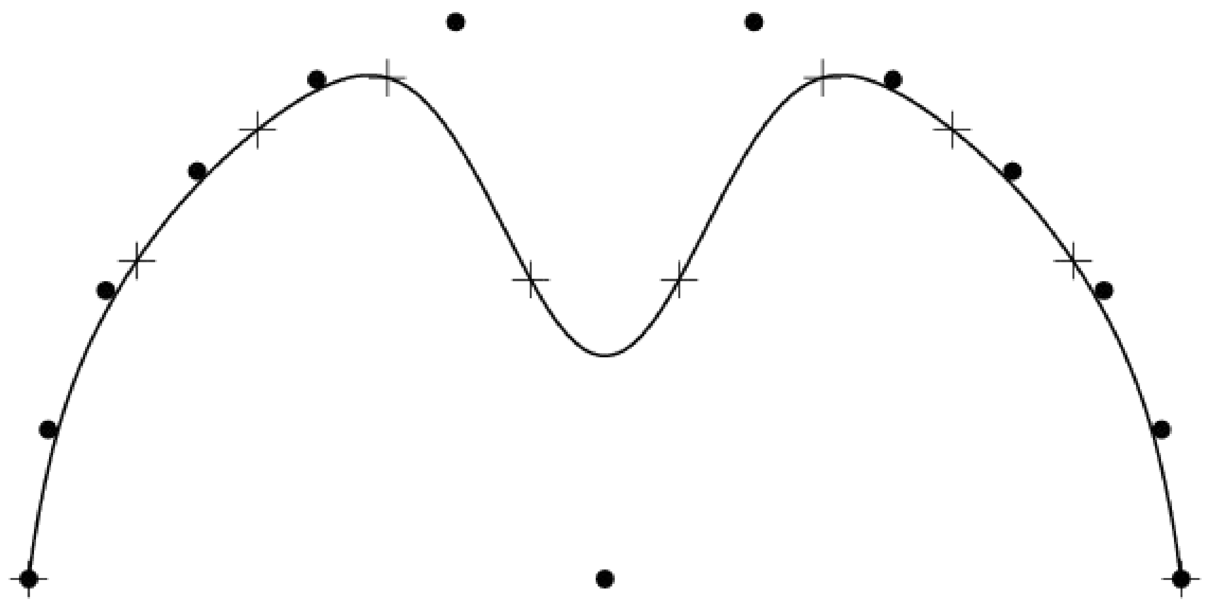
0 0 0 0.0909091 0.181818 0.272727 0.363636 0.454545 0.545455 0.636364 0.727273 0.818182 0.909091 1 1 1

B-Splines



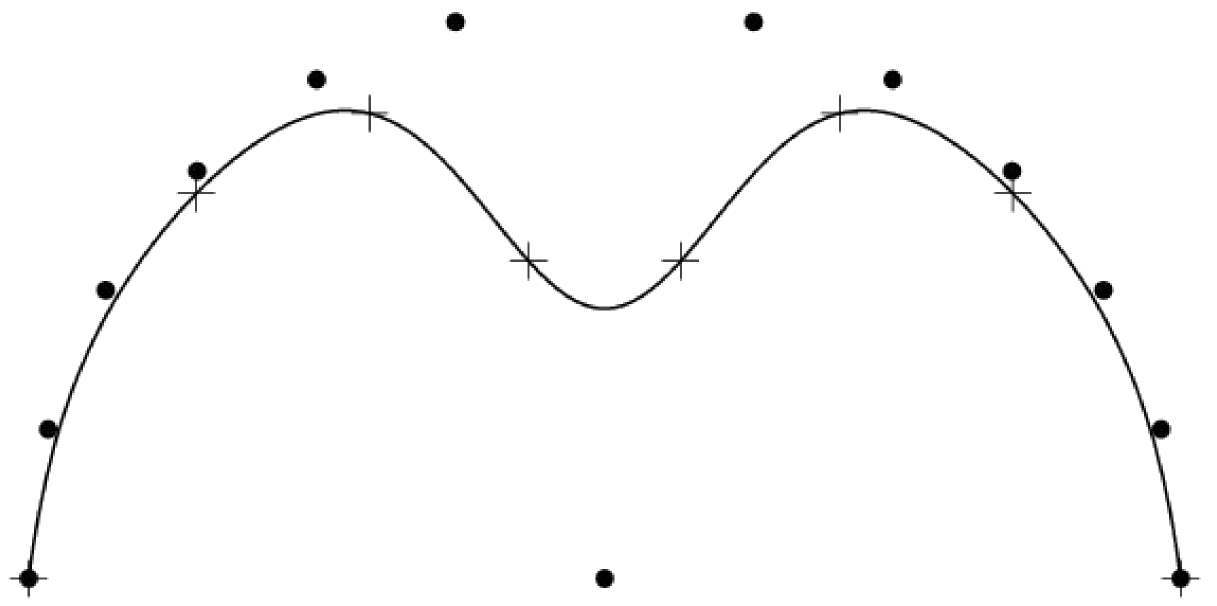
degree 3
0 0 0 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1 1 1 1

B-Splines



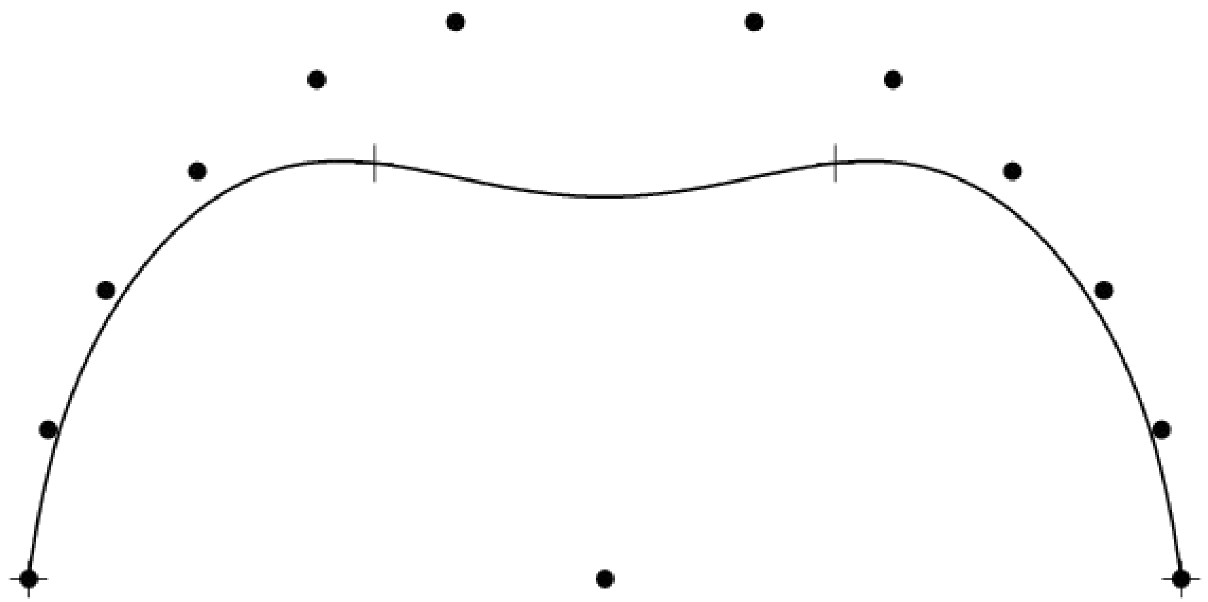
degree 4
0 0 0 0 0 0.111111 0.222222 0.333333 0.444444 0.555556 0.666667 0.777778 0.888889 1 1 1 1 1

B-Splines



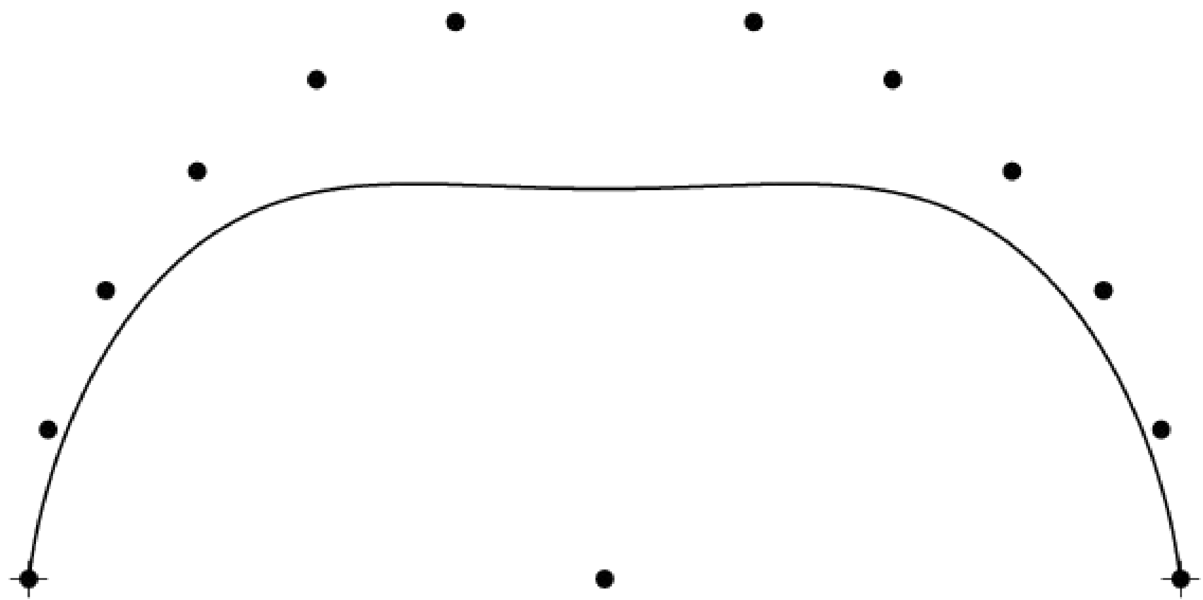
degree 6
0 0 0 0 0 0 0 0.142857 0.285714 0.428571 0.571429 0.714286 0.857143 1 1 1 1 1 1 1

B-Splines



degree 10
0 0 0 0 0 0 0 0 0 0 0 0.333333 0.666667 1 1 1 1 1 1 1 1 1 1 1

B-Splines



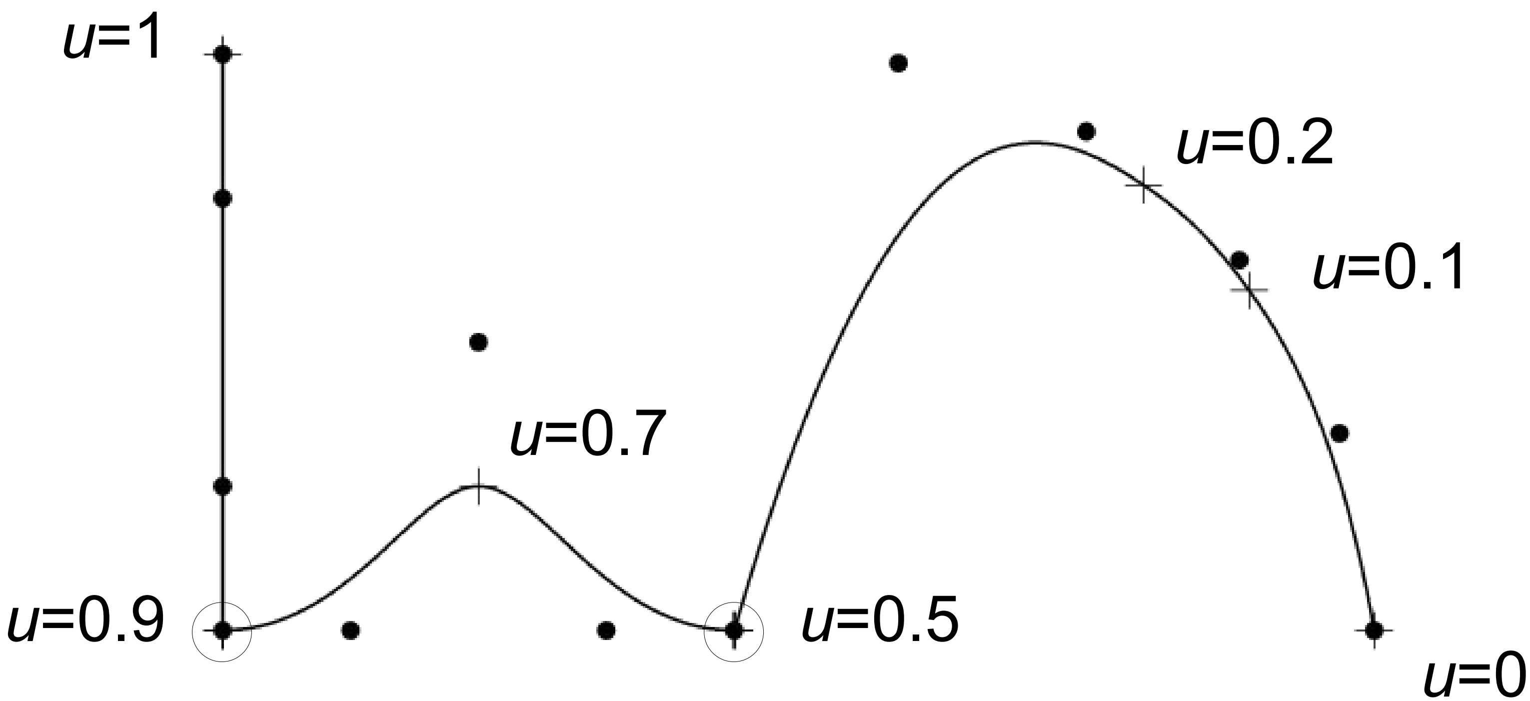
degree 12 (Bézier)

0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1

B-Splines

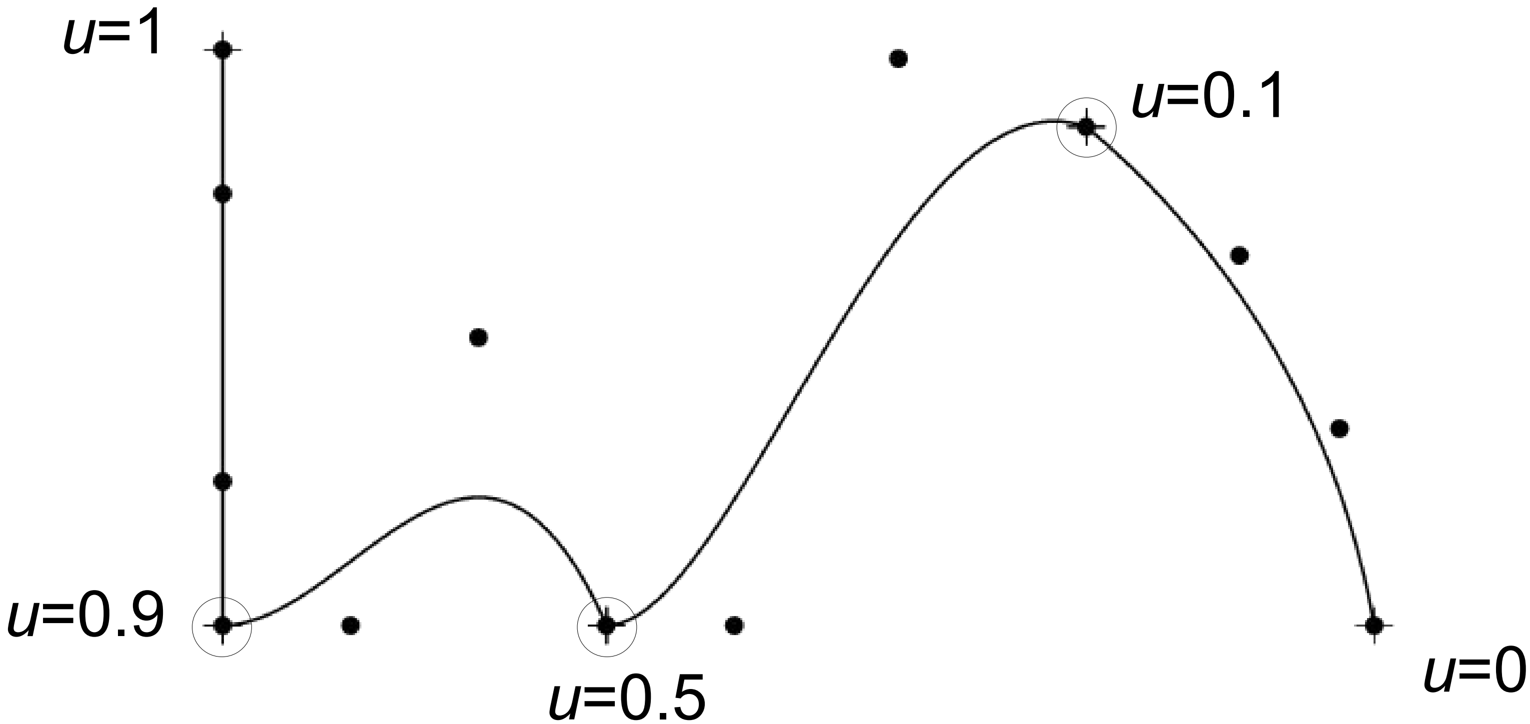
- Impose interpolation points (and C_0 continuity)
  - It is the same as positioning knots of multiplicity d in the nodal sequence
  - One could also repeat d control points...(not shown here)

B-Splines



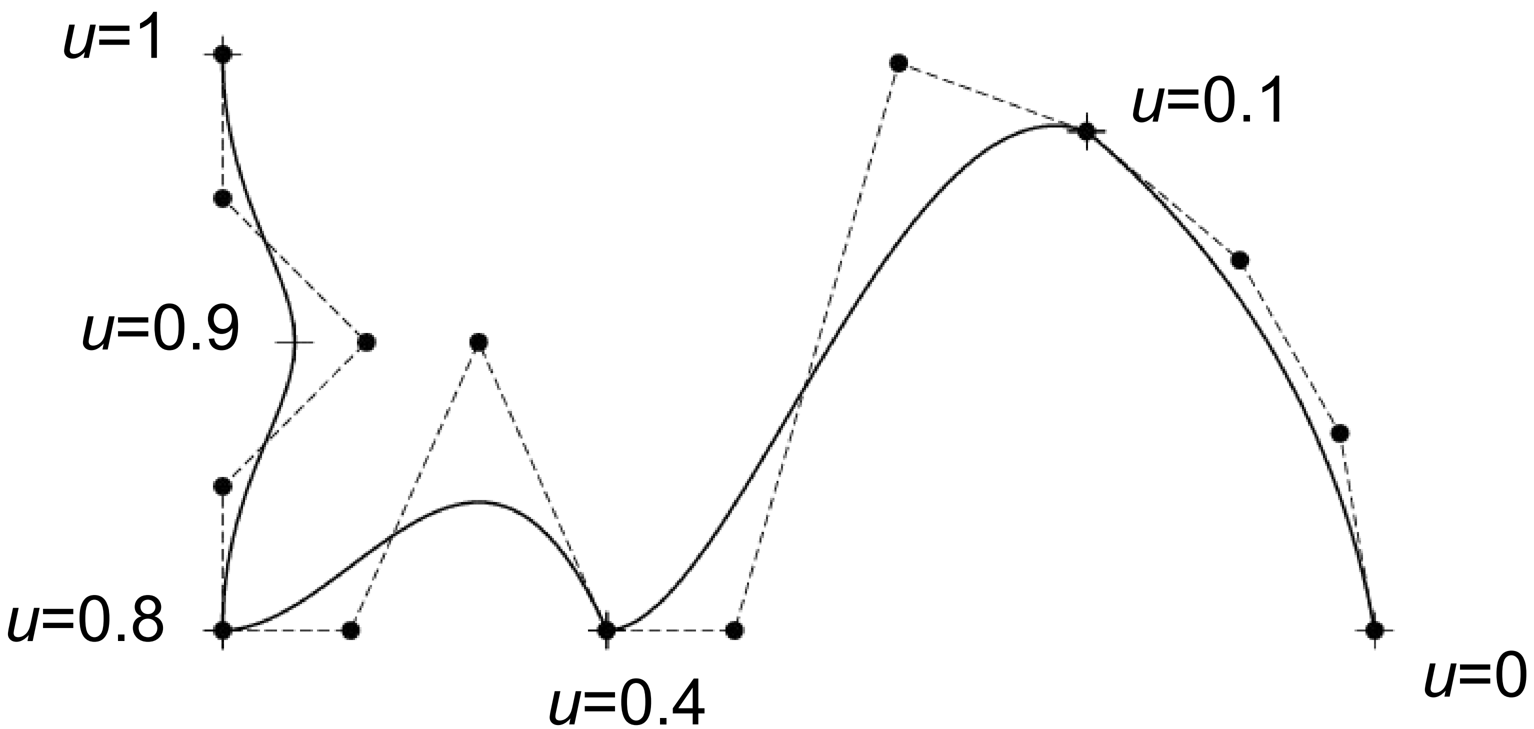
degree 3
0 0 0 0 0.1 0.2 0.5 0.5 0.5 0.7 0.9 0.9 0.9 1 1 1 1

B-Splines



degree 3 (4 Bézier curves of continuity C_0)
0 0 0 0 0.1 0.1 0.1 0.5 0.5 0.5 0.9 0.9 0.9 1 1 1 1

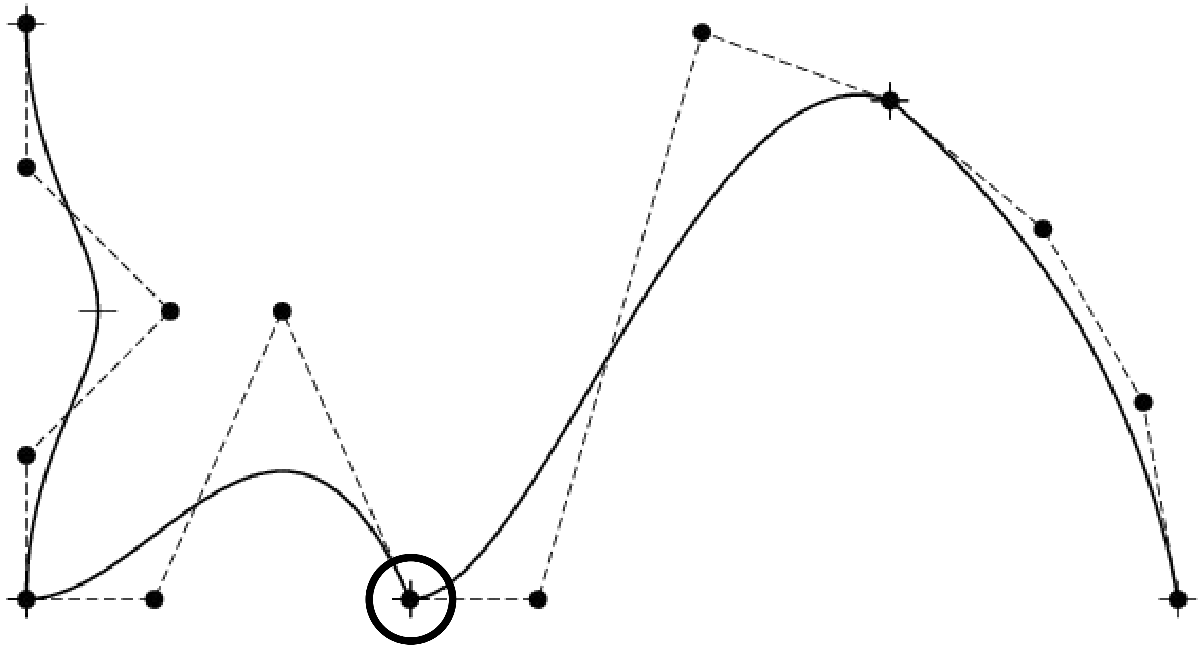
B-Splines



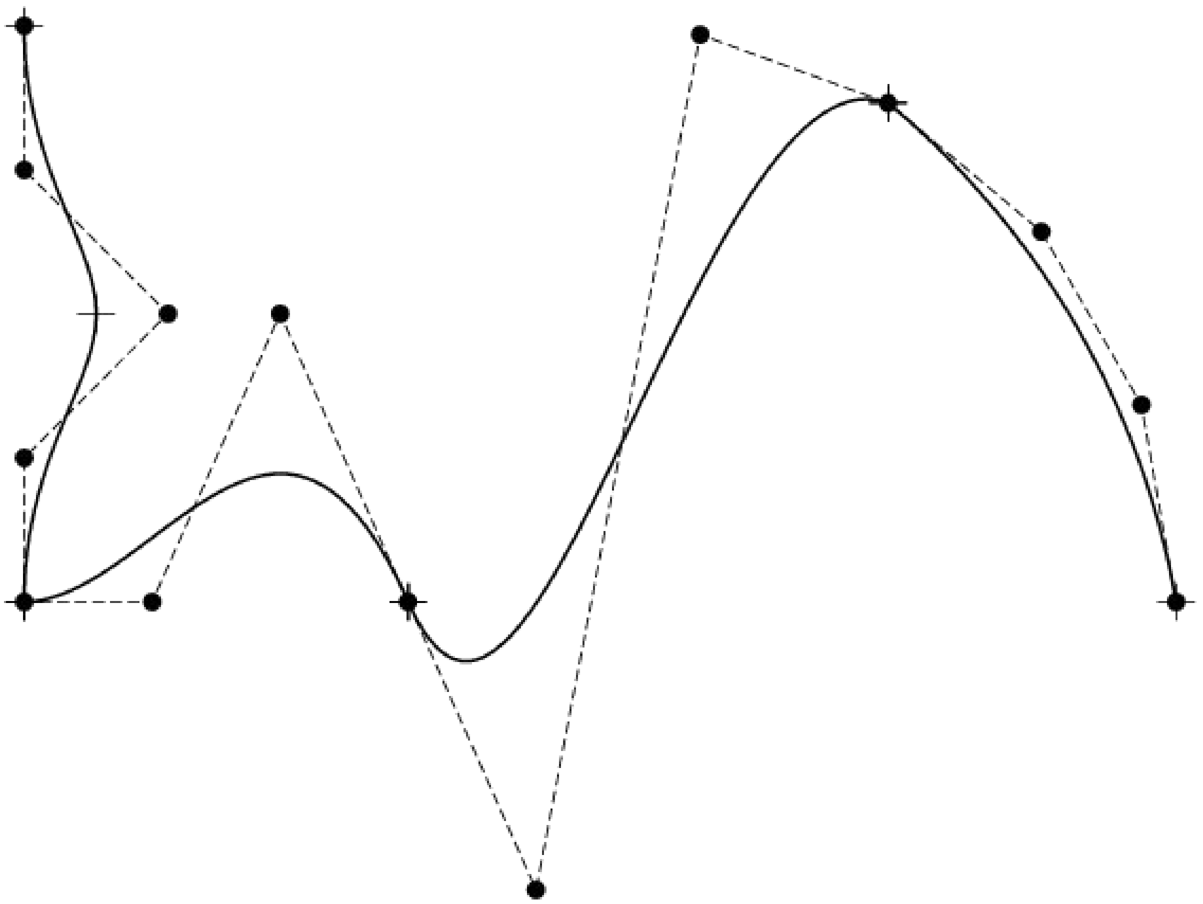
degree 3 (3 Bézier curves of continuity C_0 + 1 bspline deg 3 with 4control pts)
0 0 0 0 0.1 0.1 0.1 0.4 0.4 0.4 0.8 0.8 0.8 0.9 1 1 1 1

B-Splines

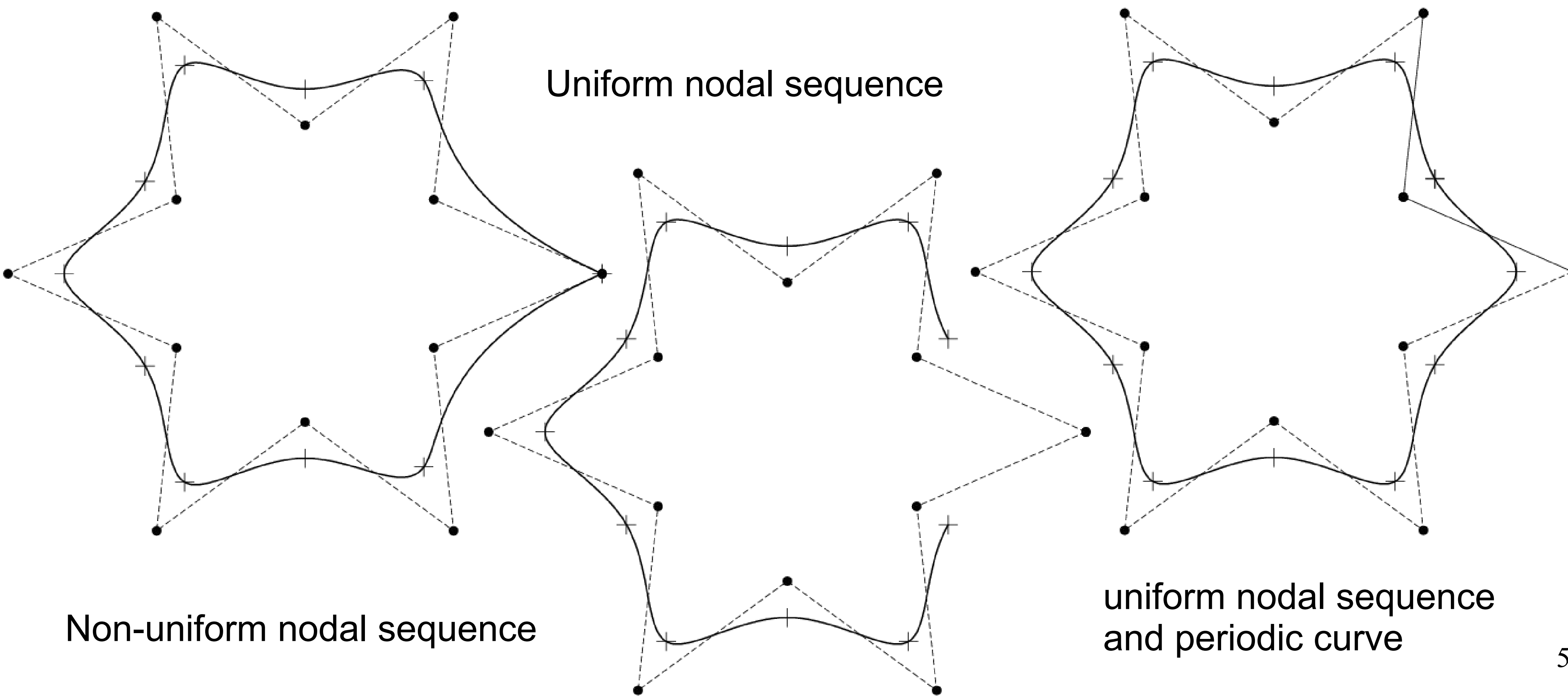
- And if we want to impose interpolation points ***and*** a certain continuity C_k ?
  - Add / align control points in a similar way than in the case of Bézier curves.



B-Splines



- Periodic curves
  - They may be represented by modifying the nodal sequence and by repeating some control points.



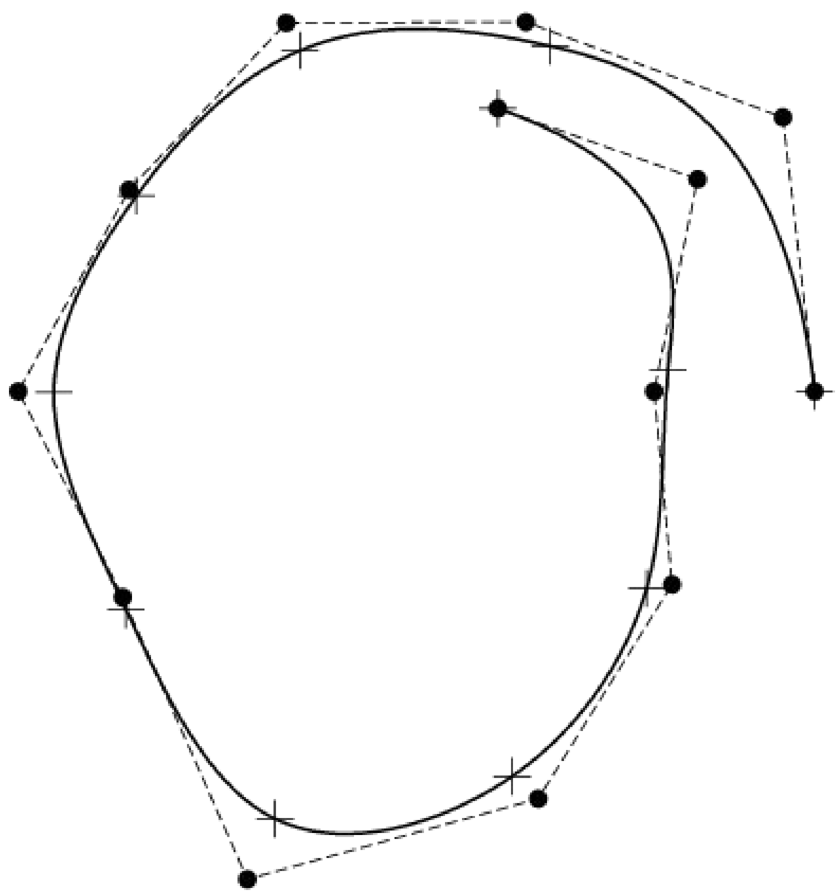
Uniform nodal sequence

Non-uniform nodal sequence

uniform nodal sequence and periodic curve

B-Splines

non uniform nodal sequence interpolating the first and last control points.

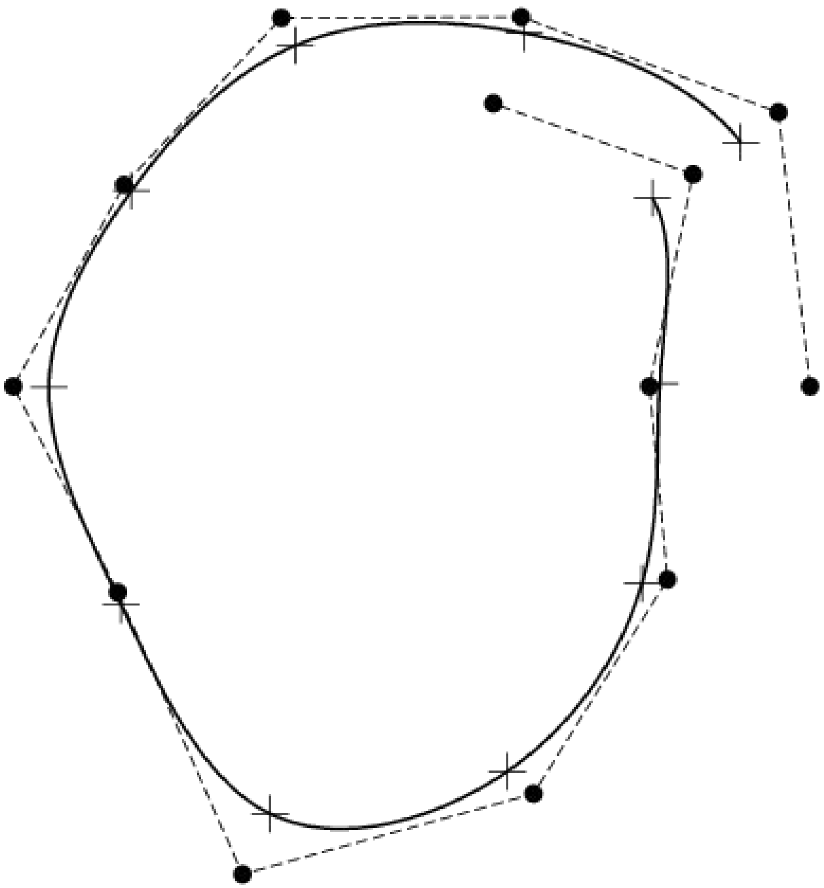


degree 3

0 0 0 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1 1 1 1

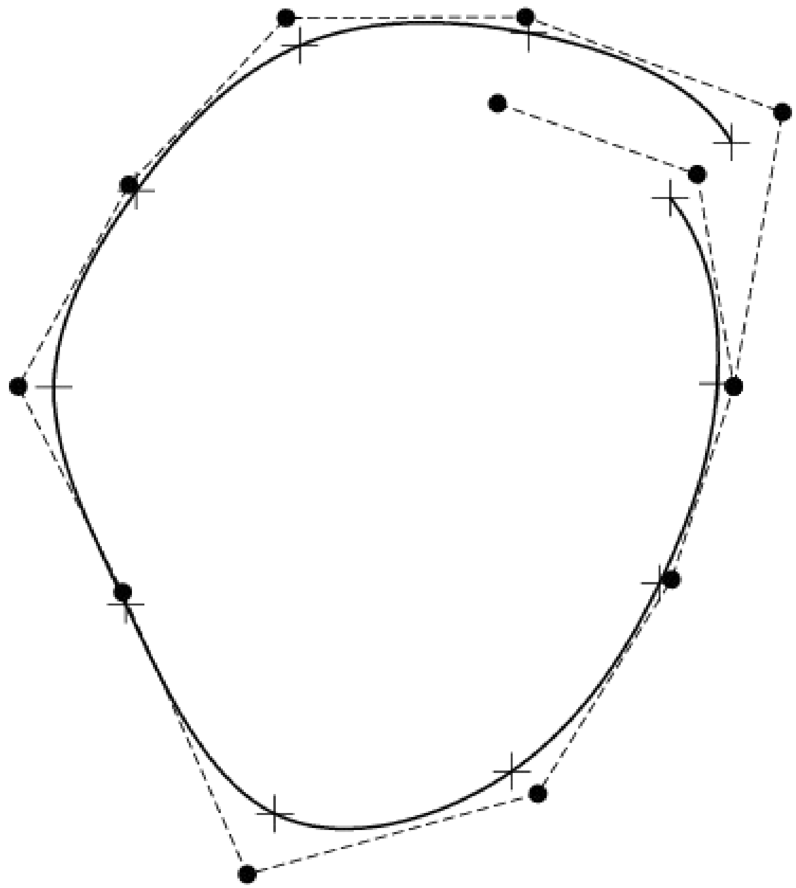
B-Splines

Periodic nodal sequence
(but control points located
inadequately)



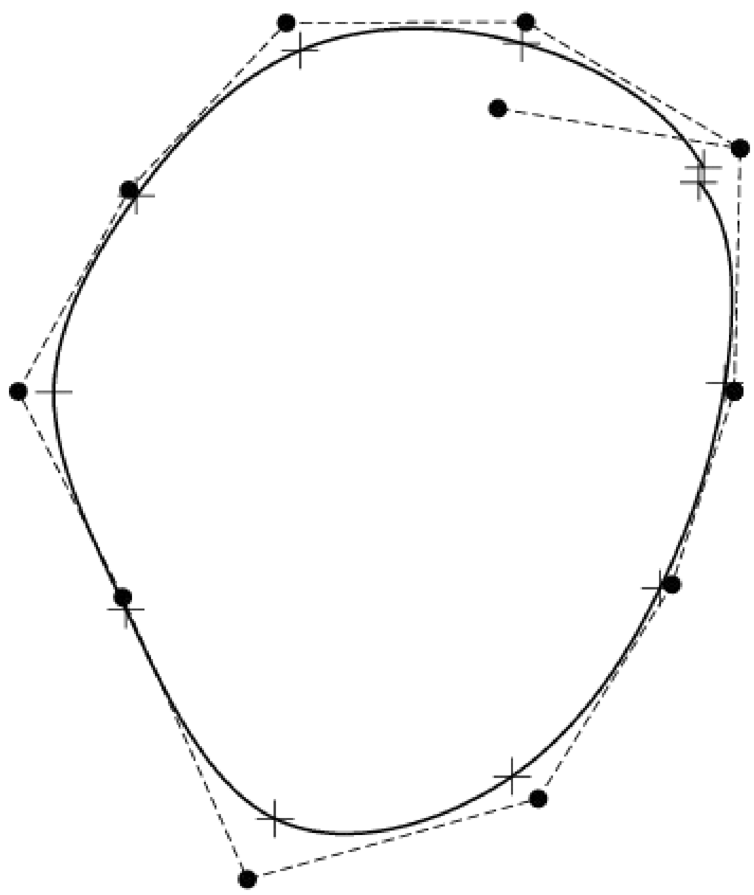
degree 3
-0.3 -0.2 -0.1 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1 1.1 1.2 1.3

B-Splines



degree 3
-0.3 -0.2 -0.1 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1 1.1 1.2 1.3

B-Splines

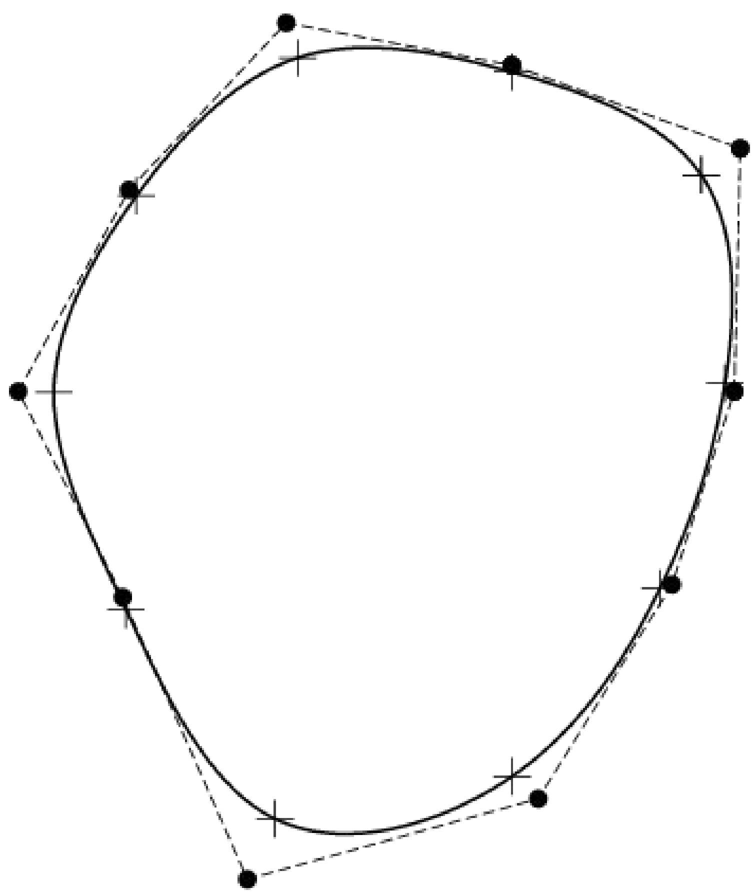


degree 3

-0.3 -0.2 -0.1 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1 1.1 1.2 1.3

B-Splines

Periodic nodal sequence
+ control points placed adequately (repeated)
= periodic curve



degree 3
-0.3 -0.2 -0.1 0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1 1.1 1.2 1.3

B-Splines

- Algorithms for the manipulation of B-Splines curves
  - Boehm's knot insertion algorithm
  - Evaluation of the curve (Cox-de Boor algorithm)
  - Derivatives and hodographs
  - Knot deletion algorithm
  - Restriction/growth of the useful interval of a curve
  - Degree elevation
  - Recursive Subdivision

- Boehm's knot insertion algorithm

The idea is to determine a new control polygon for the same curve after the insertion of one or several knots in the nodal sequence.

The curve is not modified by this change : neither the shape nor the parametrization are affected.

Interest :

- Evaluation of points on the curve
- Subdivision of the curve
- Addition of control points

- Let $P(u)=\sum_{i=0}^{n} P_i N_i^d(u)$ a B-Spline curve built on the nodal sequence : $U=\{u_0,\cdots,u_m\}$
- Let $\bar{u}\in[u_k,u_{k+1}[$ a knot to be inserted
- The new nodal sequence is :

$$\bar{U}=\{\bar{u}_0=u_0,\cdots,\bar{u}_k=u_k,\bar{u}_{k+1}=\bar{u},\cdots,\bar{u}_{m+1}=u_m\}$$

- The new representation of the curve is :

$$P(u)=\sum_{i=0}^{n+1} Q_i \bar{N}_i^d(u)$$

  - The $\bar{N}_i^d(u)$ are the basis functions defined on $\bar{U}$, the Q_i are the $n+2$ new control points.
  - How define the Q_i so that the shape is unchanged ?

- Let's set $P(u)=\sum_{i=0}^{n} P_i N_i^d(u)=\sum_{i=0}^{n+1} Q_i \bar{N}_i^d(u) \quad \forall u$
  - We write the relation for $n+2$ distinct values of u. $(a,b,\ldots)$
  - We obtain a band system, with $3(n+2)$ unknowns (in 3D)

$$\begin{pmatrix} \bar{N}_0^d(a) & \bar{N}_1^d(a) & \cdots \\ \bar{N}_0^d(b) & \bar{N}_1^d(b) & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix} \begin{pmatrix} Q_0 \\ Q_1 \\ \vdots \end{pmatrix} = \begin{pmatrix} \sum_{i=0}^{n} P_i N_i^d(a) \\ \sum_{i=0}^{n} P_i N_i^d(b) \\ \vdots \end{pmatrix}$$

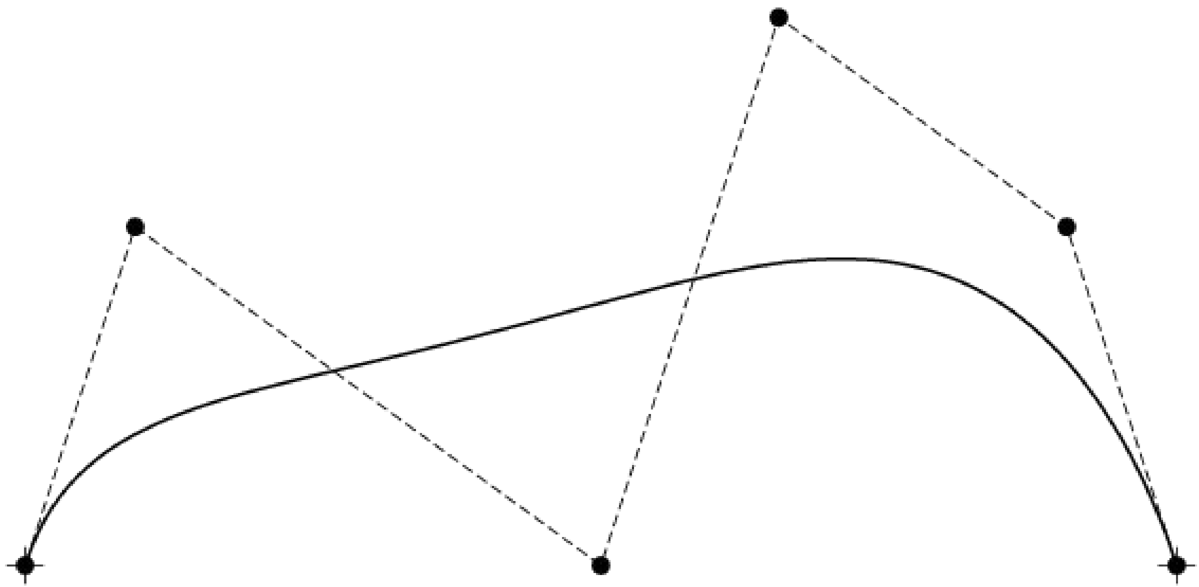
  - It's costly to solve and it assumes that the values $a,b\ldots$. are carefully set to avoid a singular lin. system

B-Splines

In fact, one can use the definition of the shape functions and after (a lot) of algebraic manipulations, one obtains Boehm's knot insertion formula :

$$Q_i = \alpha_i P_i + (1-\alpha_i) P_{i-1} \quad \text{with } \alpha_i = \begin{cases} 1 & i \leq k-d \\ \dfrac{\bar{u}-u_i}{u_{i+d}-u_i} & k-d+1 \leq i \leq k \\ 0 & i \geq k+1 \end{cases}$$

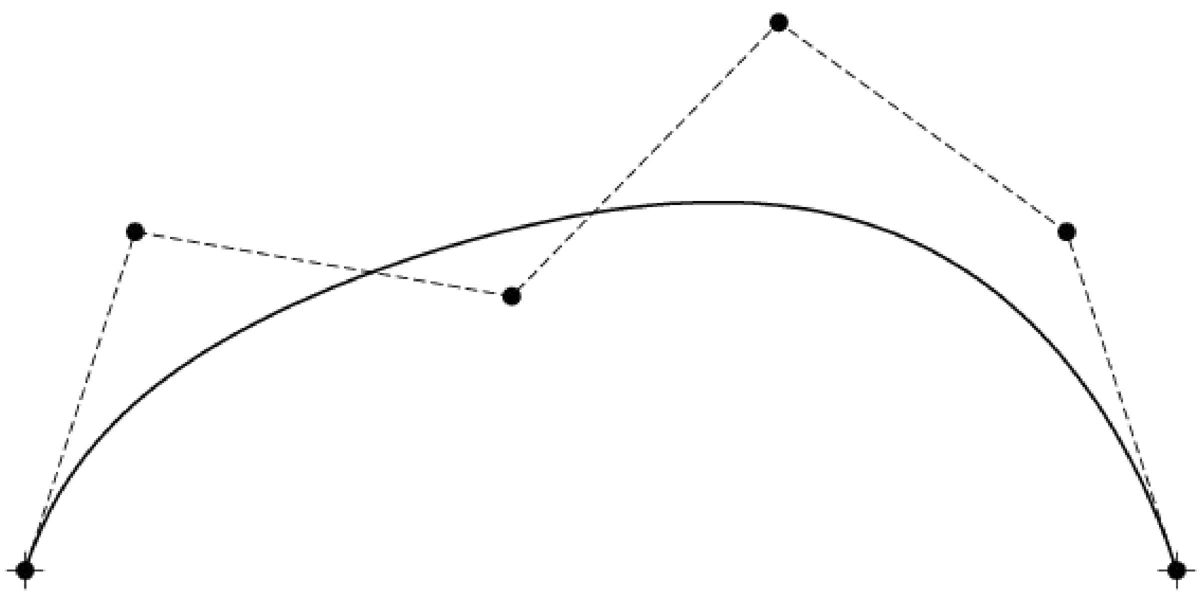
B-Splines



$$U=\{0,0,0,0,0,0,1,1,1,1,1,1\}$$

degree 5

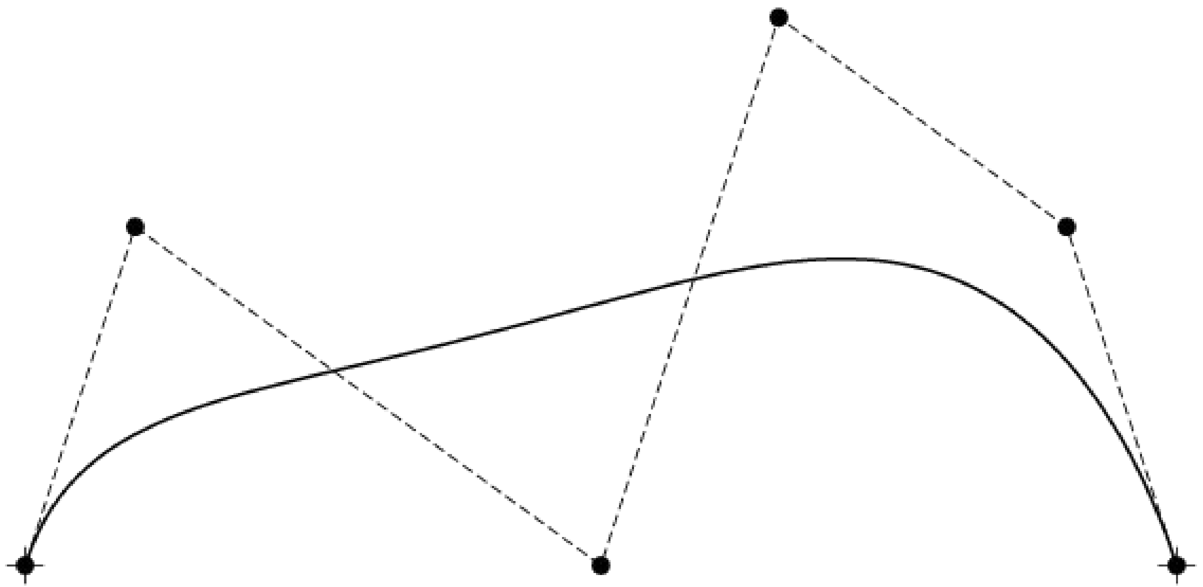
B-Splines



$$U = \{0,0,0,0,0,0,1,1,1,1,1,1\}$$

degree 5

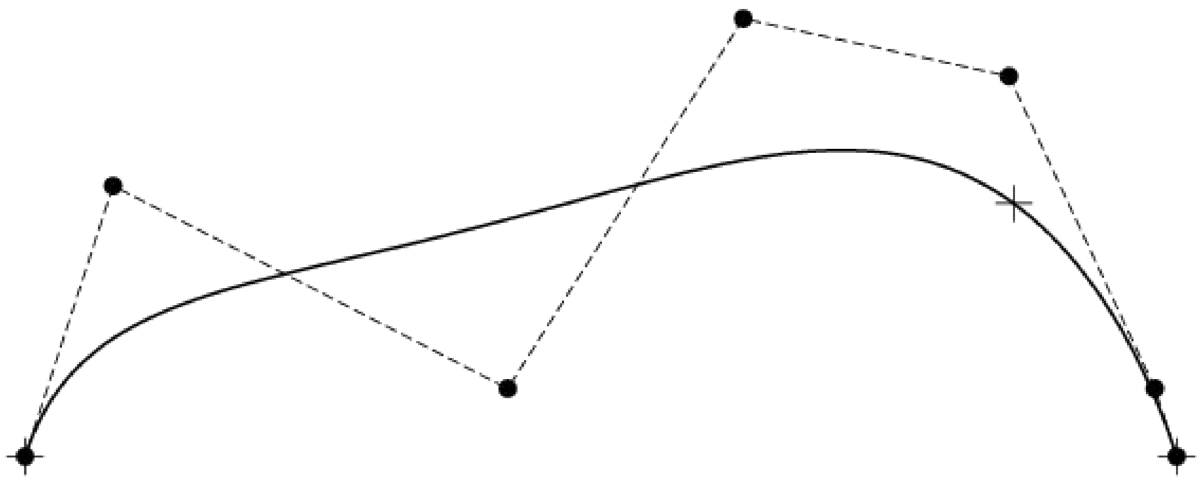
B-Splines



$$U = \{0,0,0,0,0,0,1,1,1,1,1,1\}$$

degree 5

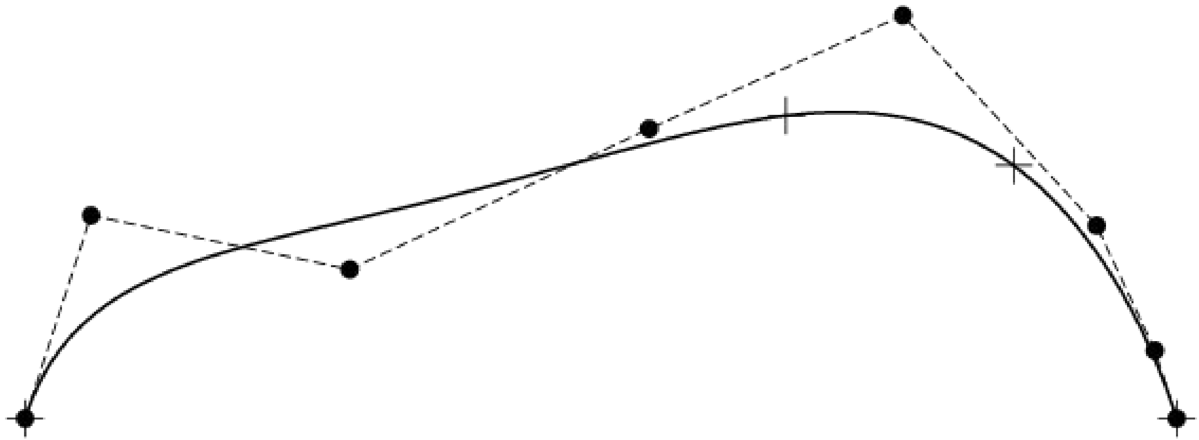
B-Splines



$$U=\{0,0,0,0,0,0,0.2,1,1,1,1,1,1\}$$

degree 5

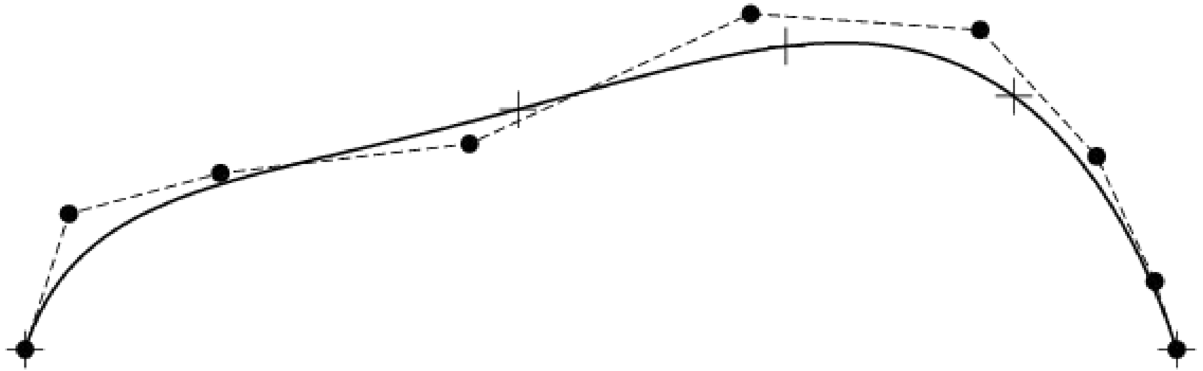
B-Splines



$$U = \{0\,,0\,,0\,,0\,,0\,,0\,,0.2\,,0.4\,,1\,,1\,,1\,,1\,,1\,,1\}$$

degree 5

B-Splines

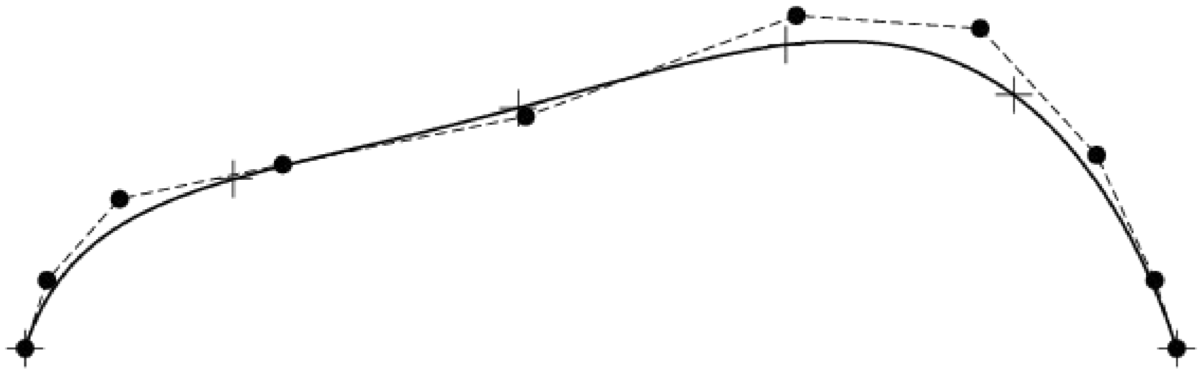


$$U = \{0, 0, 0, 0, 0, 0, 0.2, 0.4, 0.6, 1, 1, 1, 1, 1, 1\}$$

degree 5

CAD & Computational Geometry

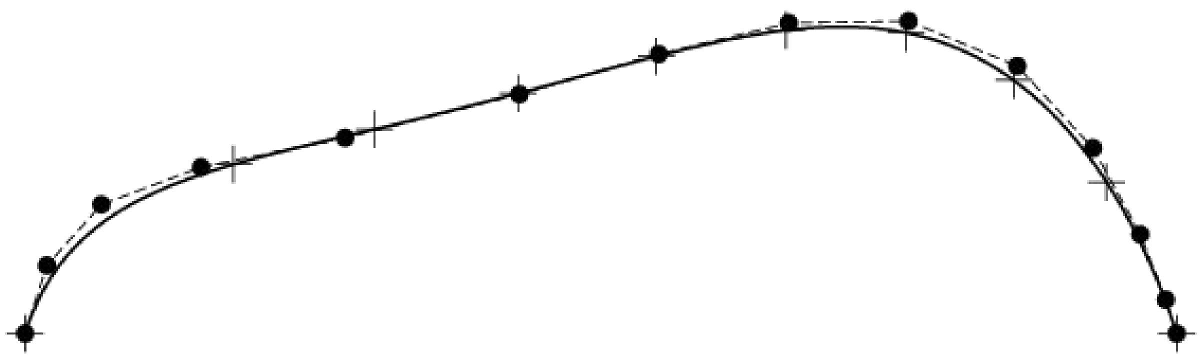
B-Splines



$$U = \{0, 0, 0, 0, 0, 0, 0.2, 0.4, 0.6, 0.8, 1, 1, 1, 1, 1, 1\}$$

degree 5

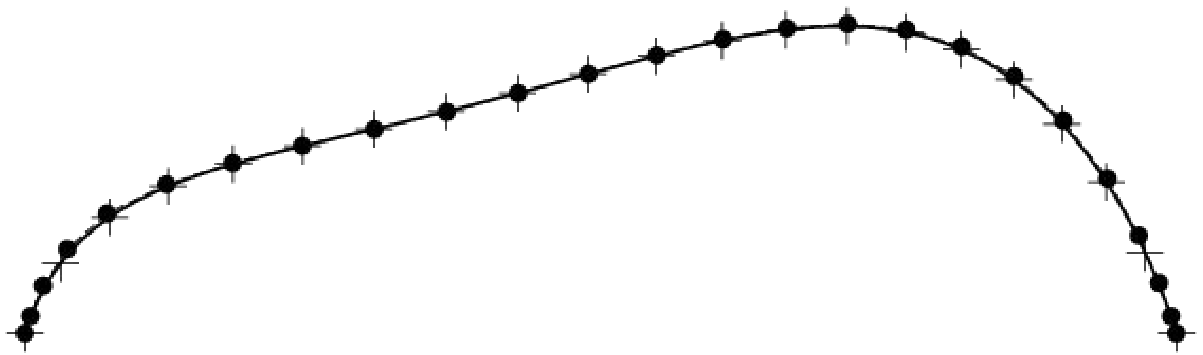
B-Splines



$$U = \{0,0,0,0,0,0,0.1,0.2,\cdots,0.9,1,1,1,1,1,1\}$$

degree 5

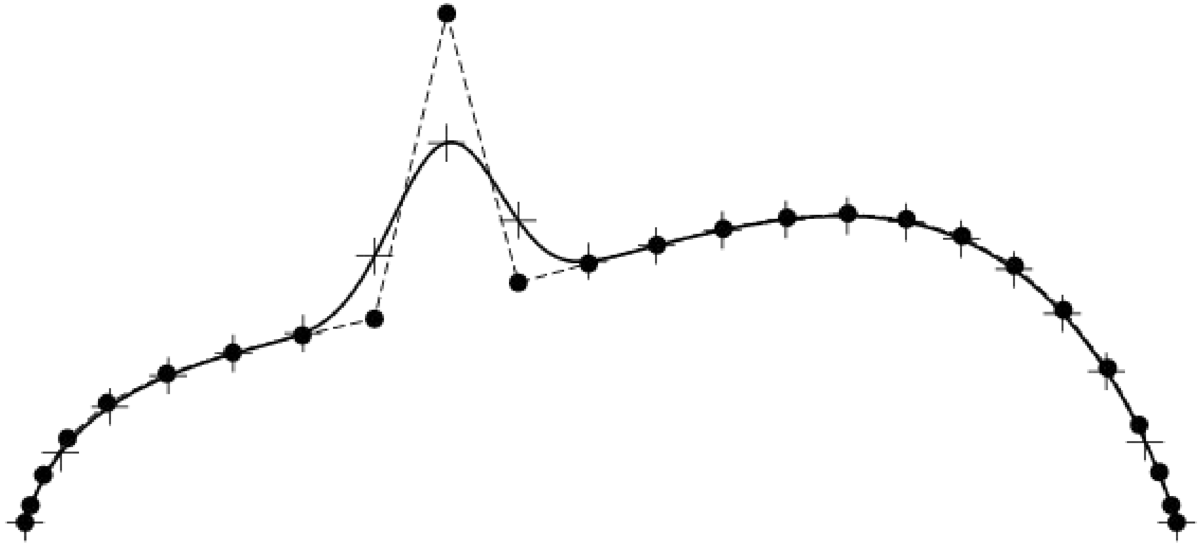
B-Splines



$$U=\{0,0,0,0,0,0,0.05,0.1,\cdots,0.95,1,1,1,1,1,1\}$$

degree 5

B-Splines



$$U = \{0,0,0,0,0,0,0.05,0.1,\cdots,0.95,1,1,1,1,1,1\}$$

degree 5

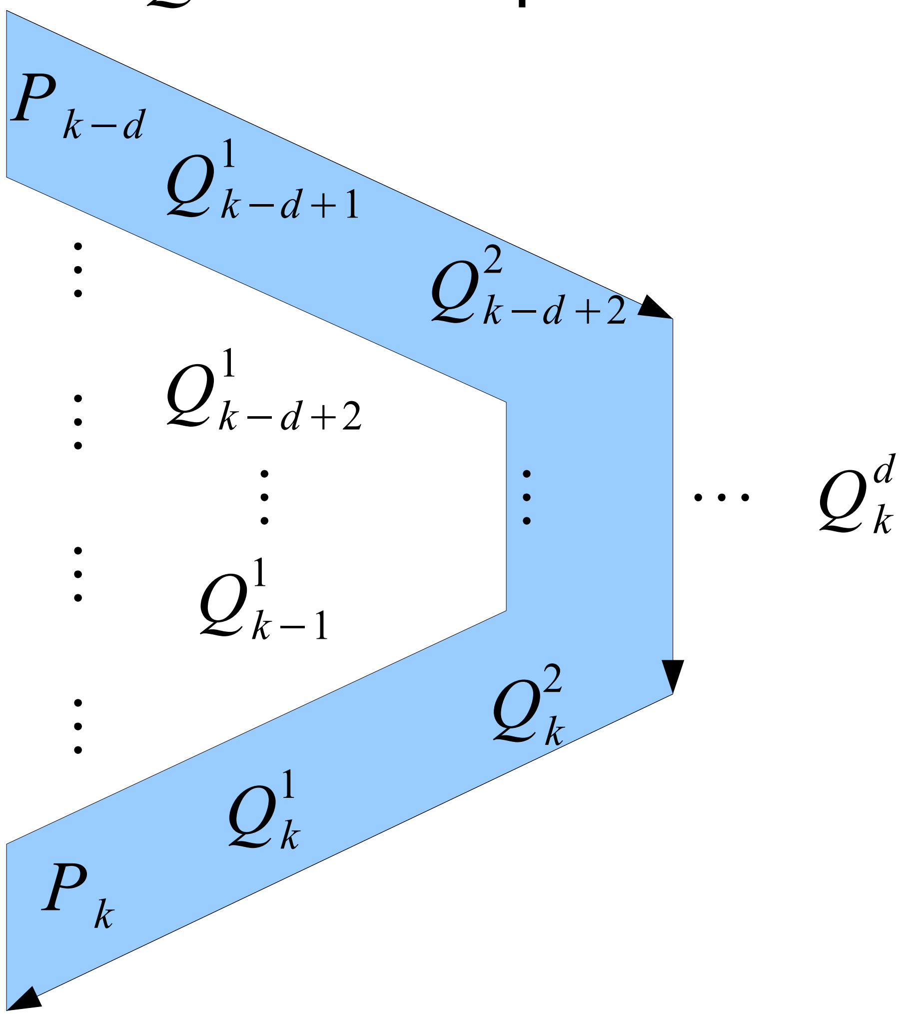
Local control...

- Multiple knot insertions
  - Assume $\bar{u} \in [u_k, u_{k+1}[$ of multiplicity s ($0 \le s < d$). We want to insert it r times with $r + s \le d$.
  - We note Q_i^r the control points of the r-th insertion step
  - We have then :

$$Q_i^r = \alpha_i^r Q_i^{r-1} + (1 - \alpha_i^r) Q_{i-1}^{r-1} \quad \text{with } \alpha_i^r = \begin{cases} 1 & i \le k - d + r - 1 \\ \dfrac{\bar{u} - u_i}{u_{i+d-r+1} - u_i} & k - d + r \le i \le k - s \\ 0 & i \ge k - s + 1 \end{cases}$$

B-Splines

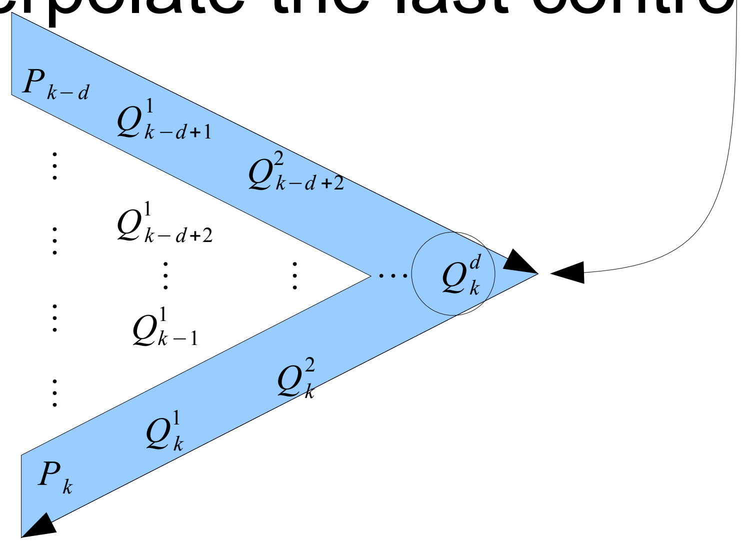
- The Q's can be put in a table:



- The total number of new control points is d-s+r-1 that replace d-s-1.

B-Splines

- The use of the algorithm of node insertion up to multiplicity of $d=r+s$ is such that the curve will interpolate the last control point that is computed.



- Therefore, one can use this algorithm to compute the position of a point of the curve knowing the parameter.
  - It's precisely Cox-de Boor's algorithm. The sequence of points P_i^j is not anything else than the Q_i^j indicated on the graph, cf following

- Case $r+s=d+1$: We carry out the insertion of multiplicity r-1 then we insert one more knot to « cut » the B-spline curve in two independent parts.
- The last control point Q_k^d has to be duplicated.
- Allows to extract a portion of the B-spline.

- There exists an extension of this algorithm in the case of the simultaneous insertion of many knots: it is the somewhat more complex “Oslo” algorithm* , not described here.

* E. Cohen, T. Lyche, R. Riesenfeld “Discrete B-splines and subdivision techniques in computer-aided geometric design and computer graphics”, *Computer Graphics and Image Processing*, **14**(2):87-111, 1980.

- (simplified) Cox-de Boor Algorithm :

Determine the interval of u : $u \in [u_i, u_{i+1}[$

Initialization of P_j^0 $\quad i \in \{d, d+1, \cdots, m-d-1\}$

For k from 1 to d

  For j from i-d+k to i

$$P_j^k = \left(\frac{u - u_j}{u_{j+d+1-k} - u_j} \right) P_j^{k-1} + \left(\frac{u_{j+d+1-k} - u}{u_{j+d+1-k} - u_j} \right) P_{j-1}^{k-1}$$

  Endfor

Endfor

P_i^d is the point that is sought.

- What is its complexity ?
  - quadratic in function of the degree d.

- On a knot of multiplicity s :

Determine the interval of u : $u \in [u_{i-s} = \cdots = u_i, u_{i+1}[$ and $u = u_i$

$i \in \{d, d+1, \cdots, m-d-1\}$

Initialization of P_j^0

For k from 1 to d-s

For j from i-d+k to i-s

$$P_j^k = \left(\frac{u - u_j}{u_{j+d+1-k} - u_j} \right) P_j^{k-1} + \left(\frac{u_{j+d+1-k} - u}{u_{j+d+1-k} - u_j} \right) P_{j-1}^{k-1}$$

Endfor

Endfor

P_{i-s}^{d-s} is the point that is sought.

- Example of computation

$$P_0^0=(0,1)\quad P_1^0=(2,3)\quad P_2^0=(5,4)\quad P_3^0=(7,1)\quad P_4^0=(6,-1)\quad P_5^0=(6,-2)$$

$$U=\{0,0,0,0,1,2,3,3,3,3\}\qquad d=3\qquad u=3/2$$

- Determination of the interval

$$1\le 3/2<2\quad ,u_4=1\ \rightarrow\ i=4$$

$$P_j^k=\left(\frac{u-u_j}{u_{j+d+1-k}-u_j}\right)P_j^{k-1}+\left(\frac{u_{j+d+1-k}-u}{u_{j+d+1-k}-u_j}\right)P_{j-1}^{k-1}$$

- Iteration 1

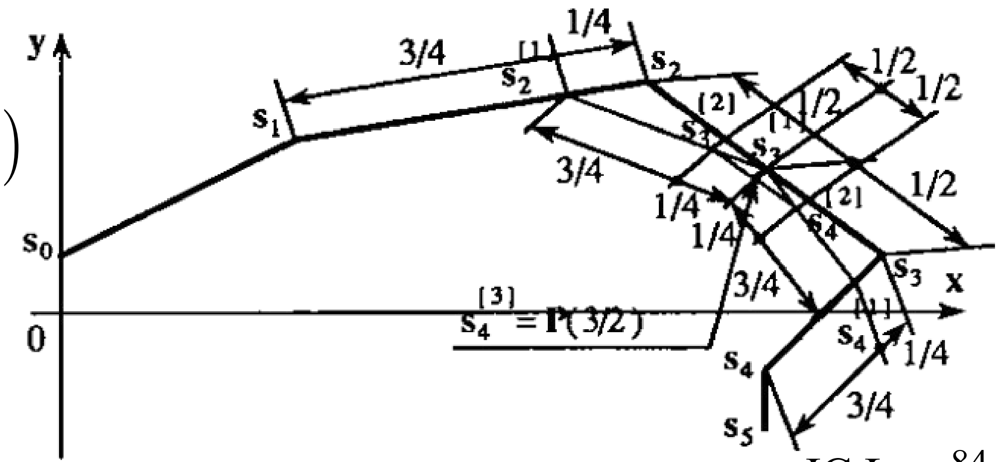
$$P_4^1=(27/4,1/2)\quad P_3^1=(6,5/2)\quad P_2^1=(17/4,15/5)$$

- Iteration 2

$$P_4^2=(99/16,2)\quad P_3^2=(89/16,45/16)$$

- Iteration 3

$$P_4^3=(47/8,77/32)=P(3/2)$$

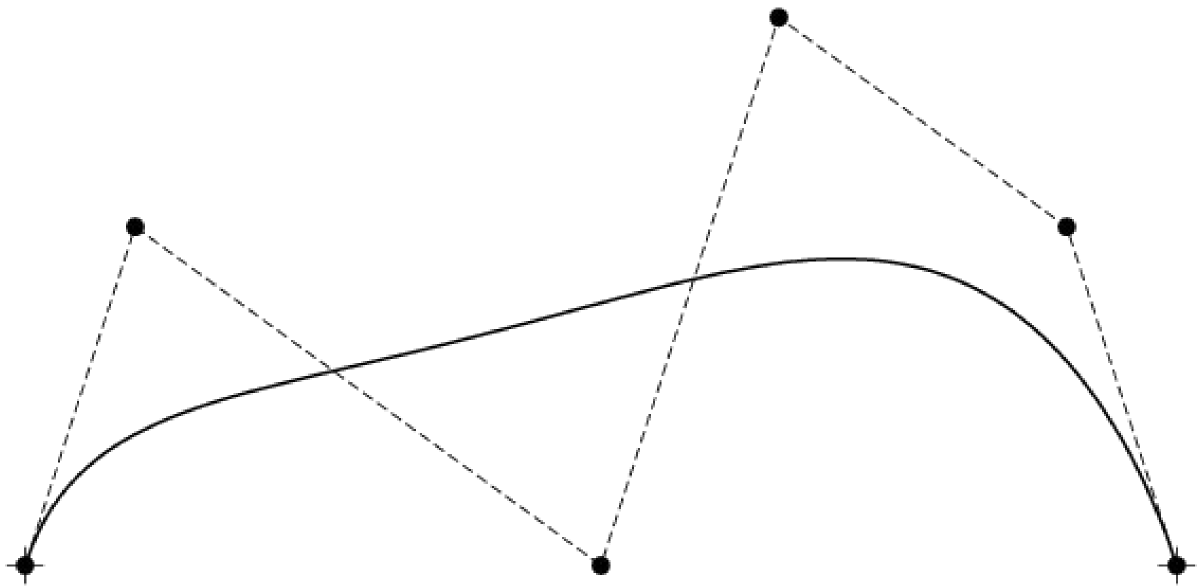


JC Leon

B-Splines

- The algorithm is similar to De Casteljau's algorithm for Bézier curves
  - It is built on a restriction of the set of control points (d+1 points)
  - On this restriction, it is nearly identical, except for the coefficients related to the nodal sequence (which is potentially **non uniform**)
  - The complete algorithm is somewhat longer than this one (possibility to have $0/0$: we set conventionally that $0/0 = 0$!)

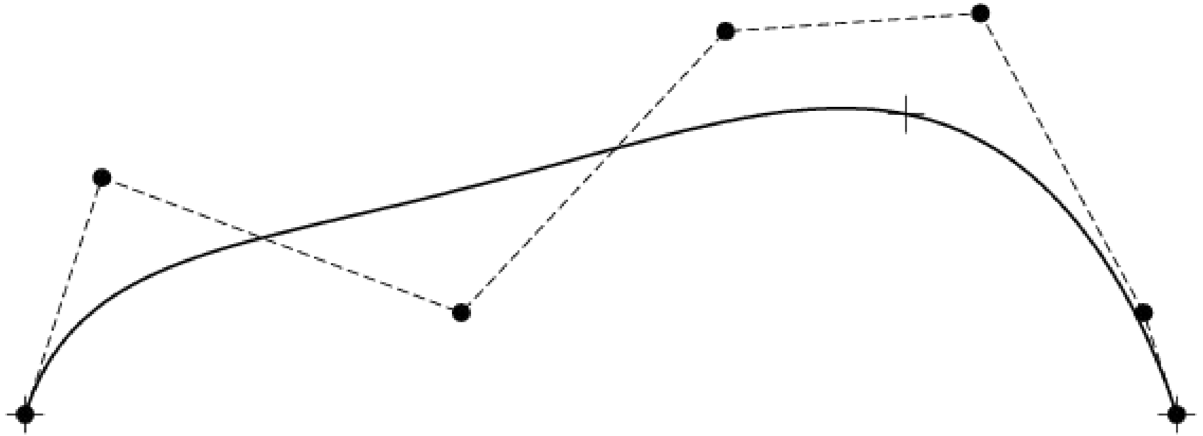
B-Splines



$$U=\{0,0,0,0,0,0,1,1,1,1,1,1\}$$

degree 5

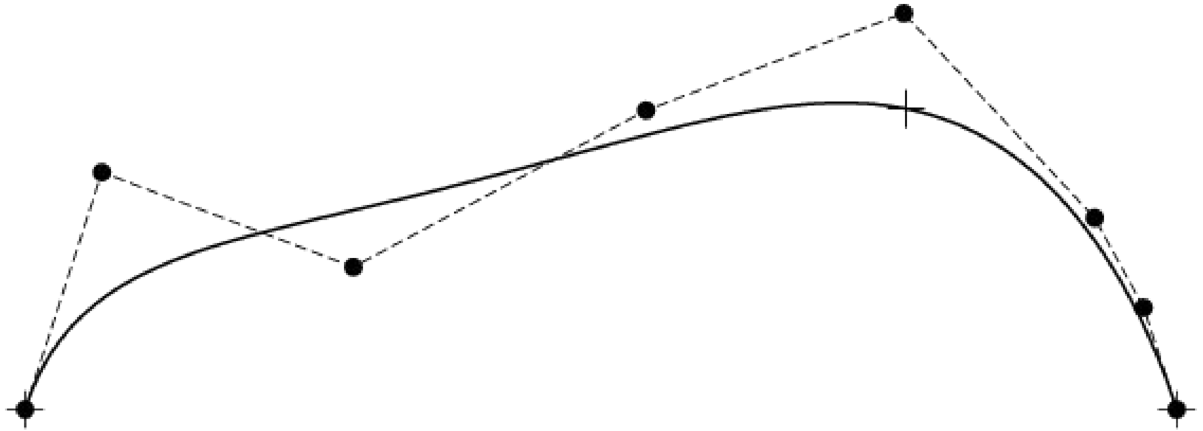
B-Splines



$$U = \{0, 0, 0, 0, 0, 0, 0.3, 1, 1, 1, 1, 1, 1\}$$

degree 5

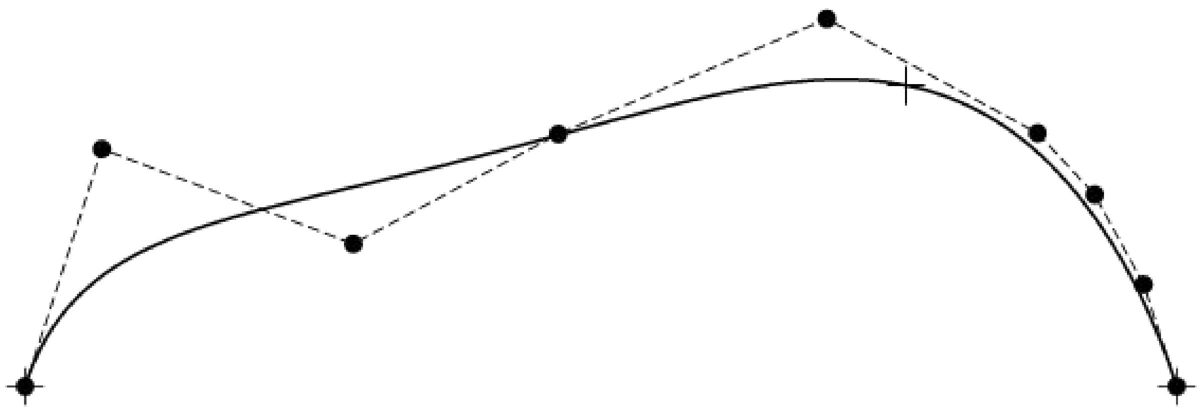
B-Splines



$$U = \{0, 0, 0, 0, 0, 0, 0.3, 0.3, 1, 1, 1, 1, 1, 1\}$$

degree 5

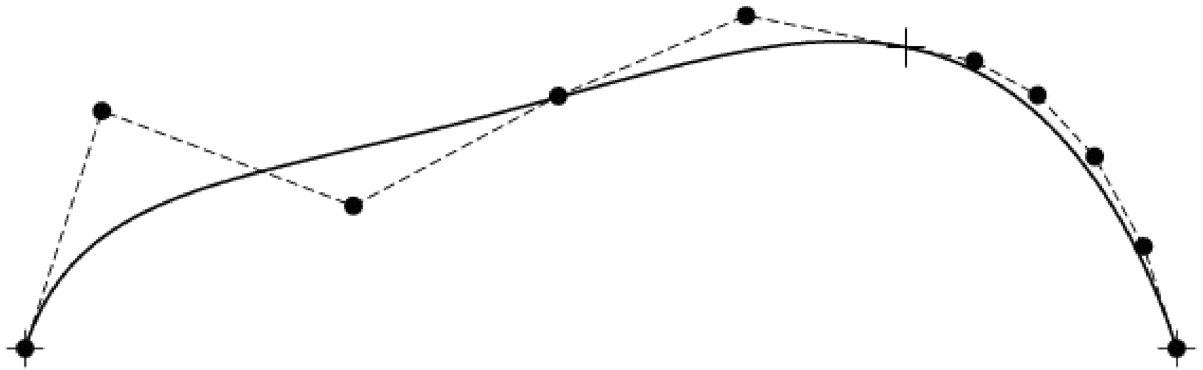
B-Splines



$$U = \{0\,,0\,,0\,,0\,,0\,,0\,,0.3\,,0.3\,,0.3\,,1\,,1\,,1\,,1\,,1\,,1\}$$

degree 5

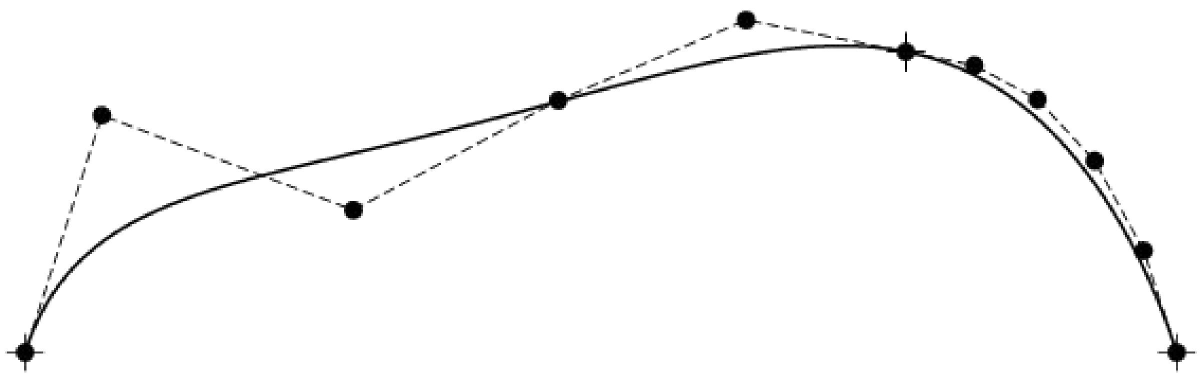
B-Splines



$$U=\{0,0,0,0,0,0,0.3,0.3,0.3,0.3,1,1,1,1,1,1\}$$

degree 5

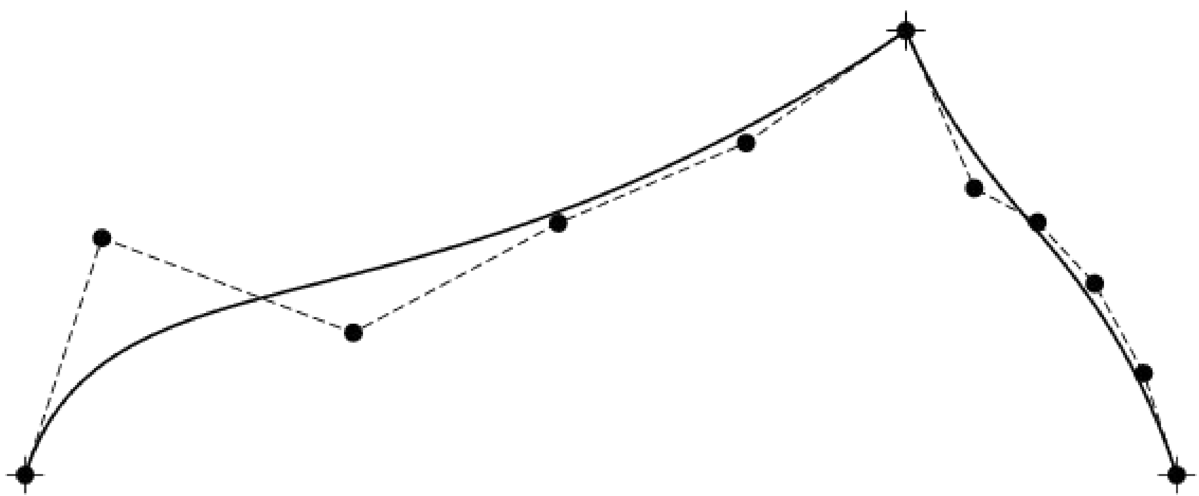
B-Splines



$$U = \{0\,,0\,,0\,,0\,,0\,,0\,,0.3\,,0.3\,,0.3\,,0.3\,,0.3\,,1\,,1\,,1\,,1\,,1\,,1\}$$

degree 5

B-Splines



$$U = \{0\,,0\,,0\,,0\,,0\,,0\,,0.3\,,0.3\,,0.3\,,0.3\,,0.3\,,1\,,1\,,1\,,1\,,1\,,1\}$$

degree 5

- Computation of a point on a B-Spline curve
  - By the use of basis functions

  1 – Find the nodal interval in which u is located

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

  2 – Calculate the non vanishing basis functions

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

  3 – Multiply the values of these basis functions with the right control points

  $$P(u) = \sum_k N_k^d(u) P_k \qquad i-d \leq k \leq i$$

  - By Cox-de Boor's algorithm

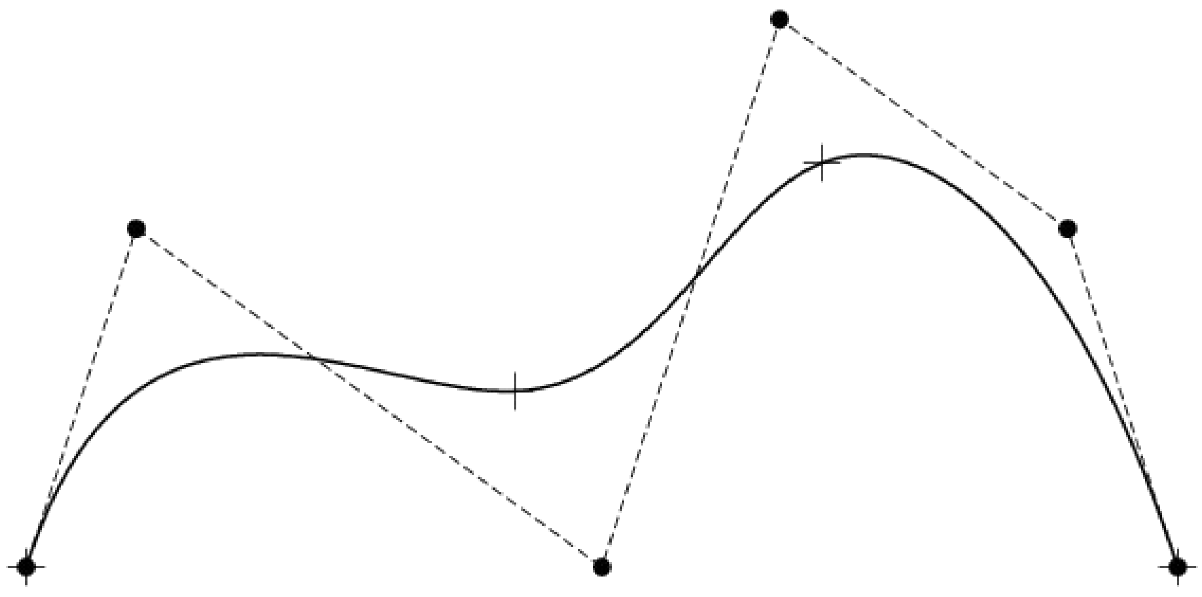
B-Splines

- Transformation of a B-Spline curve into a composite Bézier curve
  - We saturate each distinct knot until its multiplicity is equal to d.
  - This is made with the help of Boehm's algorithm of nodal insertion.
    - The curve is not modified !
  - We obtain a nodal sequence which has the following form :

$$U=\{\underbrace{a,a,a,a}_{d+1\text{ times}},\underbrace{b,b,b}_{d\text{ times}},\underbrace{c,c,c}_{d\text{ times}},\cdots,\underbrace{z,z,z,z}_{d+1\text{ times}}\}$$

    - Each distinct value of u corresponds to one of the points of the curve.

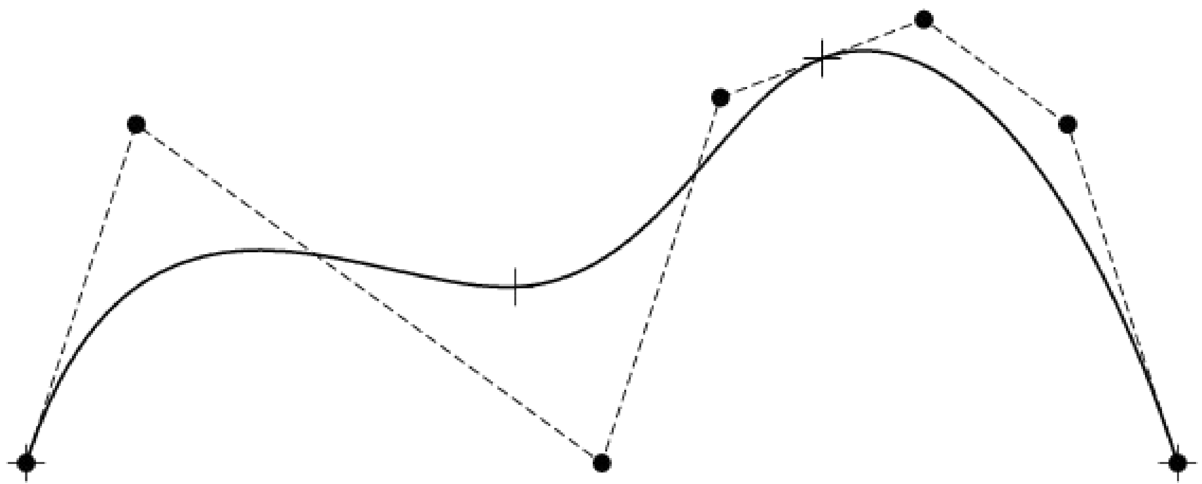
B-Splines



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

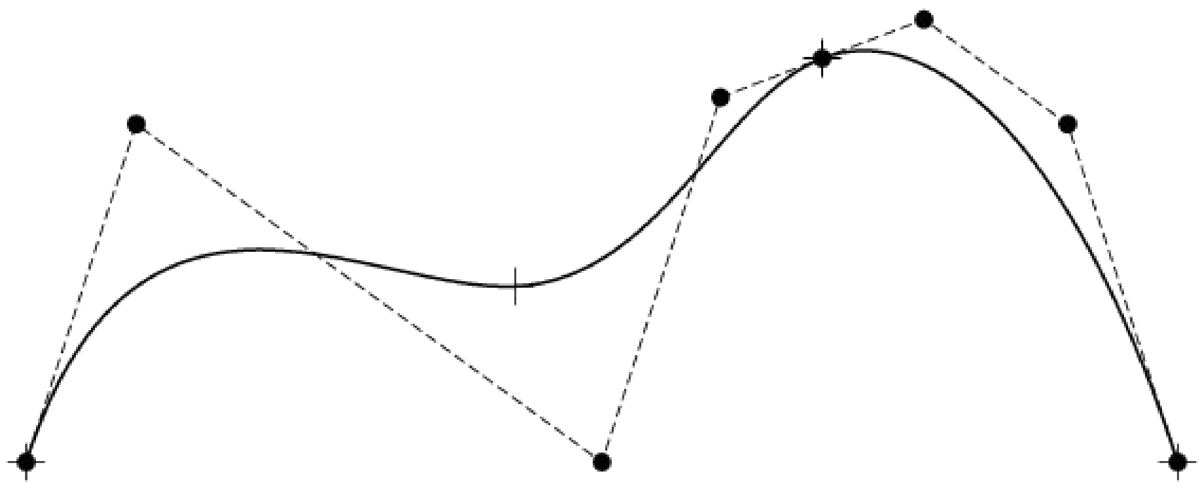
degree 3

B-Splines



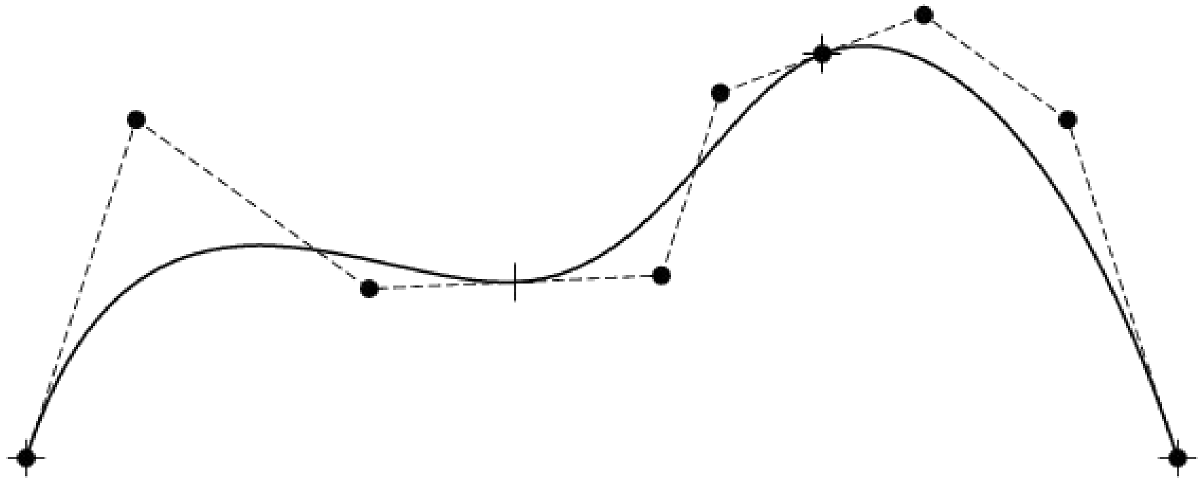
$$U = \{0, 0, 0, 0, 1, 1, 2, 3, 3, 3, 3\}$$

B-Splines



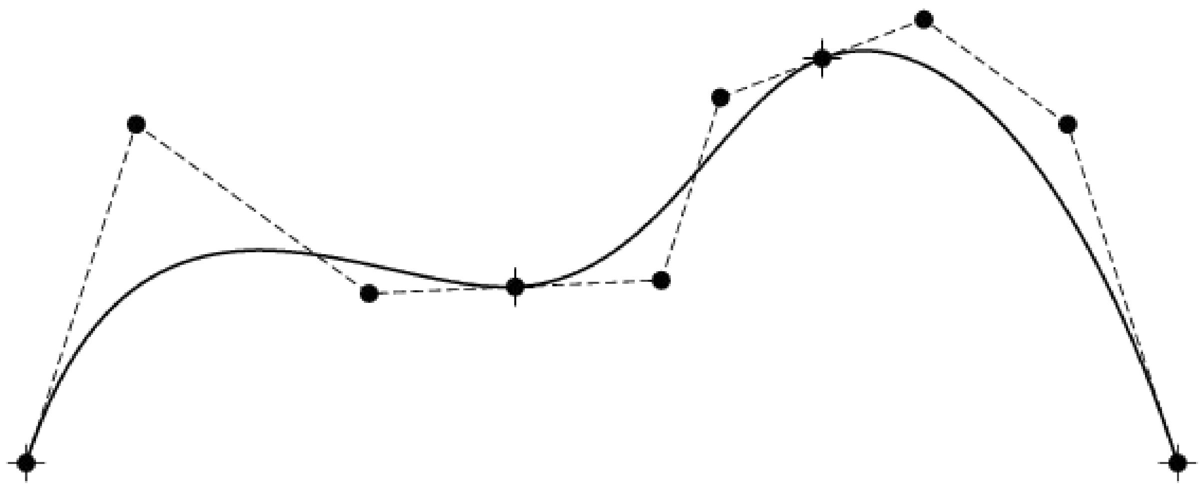
$$U = \{0, 0, 0, 0, 1, 1, 1, 2, 3, 3, 3, 3\}$$

B-Splines



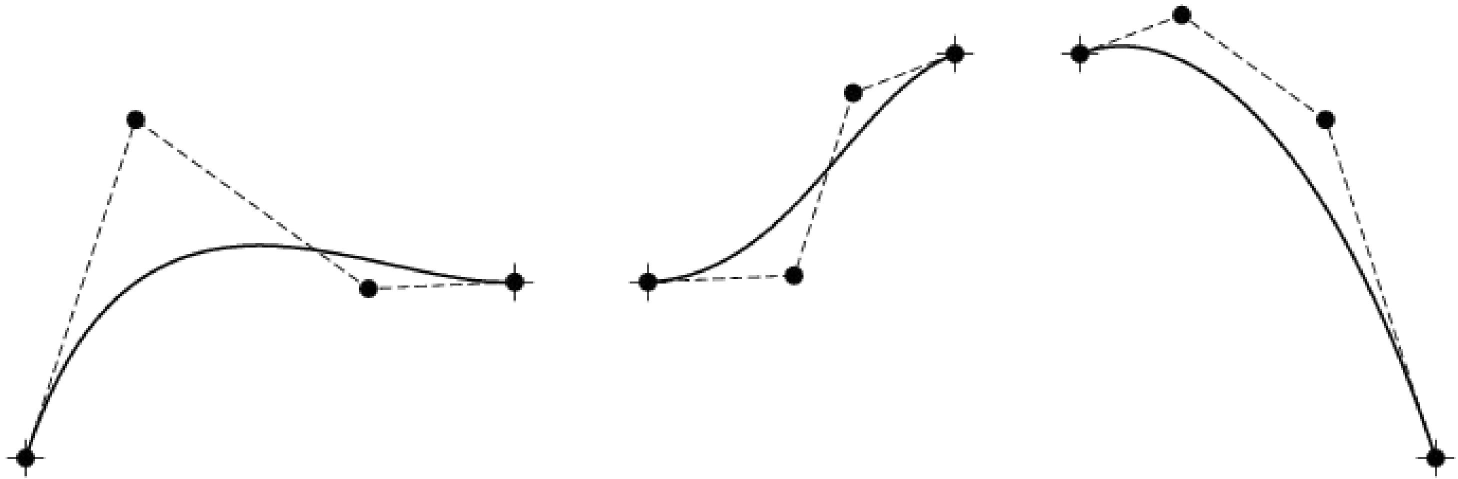
$$U = \{0,0,0,0,1,1,1,2,2,3,3,3,3\}$$

B-Splines



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

B-Splines



Bézier n°3
4 CP
degree 3

Bézier n°2
4 CP
degree 3

Bézier n°1
4 CP
degree 3

- Recursive subdivision
  - As in the case of Bézier curves, one can increase the number of CPs , this time using Boehm's knot insertion algorithm instead of De Casteljau's.
    - The control polygon converges to the curve as one inserts new knots (each time in the middle of the widest knot interval)
  - The usual order of B-Spline curves is small (3 or 4, maybe 5), so savings in computational complexity with respect to a systematic evaluation with Cox-de Boor's algorithm are not so dramatic.

- Some conclusions
  - Flexibility
  - Low order
  - Continuity
  - Periodic curves
  - Conic sections ?