A subdivision approach to splines, exponential splines and beyond

Costanza Conti
University of Florence, Italy

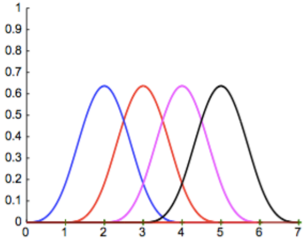
Twenty Years of Biomedical Imaging and Splines
Losanna, Switzerland, March 23, 2018

Outline

The polynomial B-spline basis

Polynomial B-splines are bell shaped compacted supported basis functions for polynomial splines that find application in many different context

- geometric modeling
- computer graphics
- curve/surface fitting
- numerical differentiation/integration
- signal/image processing
- solution of PDE also via IgA
- statistics
- ....



The polynomial cardinal B-spline basis: integer simple knots

The polynomial cardinal B-spline basis: integer simple knots

- recurrence : $\rightarrow B_{h,i}(t) = \frac{t-i}{i+h-1-i} B_{h-1,i}(t) + \frac{i+h-t}{i+h-(i+1)} B_{h-1,i+1}(t), \quad h = 2, ..., N$

The polynomial cardinal B-spline basis: integer simple knots

- recurrence : $\rightarrow$ $B_{h,i}(t) = \frac{t-i}{i+h-1-i} B_{h-1,i}(t) + \frac{i+h-t}{i+h-(i+1)} B_{h-1,i+1}(t), \quad h = 2, ..., N$
- divided difference of truncated power: $\rightarrow$ $B_N = [0, 1, \cdots, N](\cdot)_+^N$;

The polynomial cardinal B-spline basis: integer simple knots

- recurrence : $\rightarrow B_{h,i}(t) = \frac{t-i}{i+h-1-i} B_{h-1,i}(t) + \frac{i+h-t}{i+h-(i+1)} B_{h-1,i+1}(t), \quad h = 2, ..., N$
- divided difference of truncated power: $\rightarrow B_N = [0, 1, \cdots, N](\cdot)_+^N$;
- convolution: $\rightarrow B_h = B_{h-1} * B_0, \; h = 1, \cdots, N$;

The polynomial cardinal B-spline basis: integer simple knots

- recurrence : $\rightarrow B_{h,i}(t) = \frac{t-i}{i+h-1-i} B_{h-1,i}(t) + \frac{i+h-t}{i+h-(i+1)} B_{h-1,i+1}(t), \quad h = 2, ..., N$
- divided difference of truncated power: $\rightarrow B_N = [0, 1, \cdots, N](\cdot)_+^N$;
- convolution: $\rightarrow B_h = B_{h-1} * B_0,\ h = 1, \cdots, N$;
- Green functions of diff. op. : $\rightarrow B_N = \Delta^N (\rho(t))$, with $D^N (\rho(t)) = \delta$;

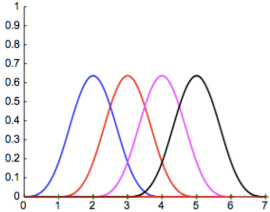
The polynomial cardinal B-spline basis: integer simple knots

- recurrence : $\rightarrow B_{h,i}(t) = \frac{t-i}{i+h-1-i} B_{h-1,i}(t) + \frac{i+h-t}{i+h-(i+1)} B_{h-1,i+1}(t), \quad h = 2, ..., N$
- divided difference of truncated power: $\rightarrow B_N = [0, 1, \cdots, N](\cdot)_+^N$;
- convolution: $\rightarrow B_h = B_{h-1} * B_0, \; h = 1, \cdots, N$;
- Green functions of diff. op. : $\rightarrow B_N = \Delta^N(\rho(t))$, with $D^N(\rho(t)) = \delta$;
- limit of (discrete) iterative schemes: subdivision basic limit functions

The polynomial cardinal B-spline basis: integer simple knots

- recurrence : $\rightarrow B_{h,i}(t) = \frac{t-i}{i+h-1-i} B_{h-1,i}(t) + \frac{i+h-t}{i+h-(i+1)} B_{h-1,i+1}(t), \quad h = 2, ..., N$
- divided difference of truncated power: $\rightarrow B_N = [0, 1, \cdots, N](\cdot)_+^N$;
- convolution: $\rightarrow B_h = B_{h-1} * B_0, \ h = 1, \cdots, N$;
- Green functions of diff. op. : $\rightarrow B_N = \Delta^N (\rho(t))$, with $D^N (\rho(t)) = \delta$;
- limit of (discrete) iterative schemes: subdivision basic limit functions

i support size $N + 1$ and non negative;
ii C^{N-1} global regularity;
iii form a partition of unity $\sum_{i \in \mathbb{Z}} B_{N,i} = 1$;
iv shifted copies of each other $B_{N,i} = B_{N,0}(\cdot - i)$;



Limits of polynomial B-splines

In spite of the nice properties they have, polynomial B-splines

Limits of polynomial B-splines

In spite of the nice properties they have, polynomial B-splines

- their use to model manifolds with arbitrary topology is conceptually very complicated and extremely expensive;

Limits of polynomial B-splines

In spite of the nice properties they have, polynomial B-splines

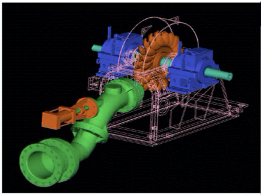
- their use to model manifolds with arbitrary topology is conceptually very complicated and extremely expensive;
- have a low approximation order. A pre-processing of the data is necessary to get higher approximation order;

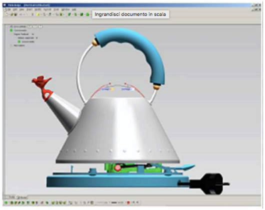
Limits of polynomial B-splines

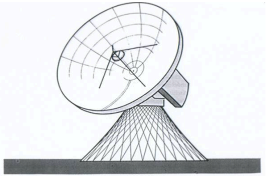
In spite of the nice properties they have, polynomial B-splines

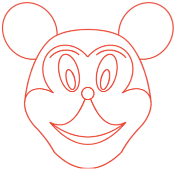
- their use to model manifolds with arbitrary topology is conceptually very complicated and extremely expensive;
- have a low approximation order. A pre-processing of the data is necessary to get higher approximation order;
- are not able reproduce geometries like conic sections which are important e.g. in geometric modeling, biomedical imaging and IgA.

Conics in geometric modeling/CAD

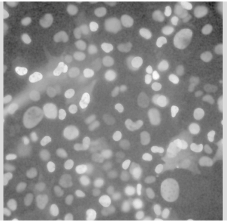


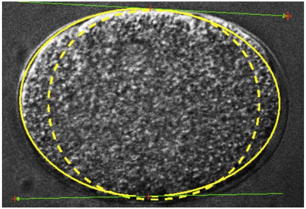






Conics in medical image processing





NURBS: non uniform rational B-splines

In geometric modeling a B-splines generalization yield in the 90s NURBS

$$s(t) = \sum_{i=1}^{n} P_i \, B_{N,i}(t), \qquad s(t) = \frac{\sum_{i=1}^{n} P_i \, B_{N,i}(t) w_i}{\sum_{i=1}^{n} B_{N,i}(t) w_i} = \sum_{i=1}^{n} P_i \, R_{N,i}(t).$$

NURBS: non uniform rational B-splines

In geometric modeling a B-splines generalization yield in the 90s NURBS

$$s(t) = \sum_{i=1}^{n} P_i \, B_{N,i}(t), \qquad s(t) = \frac{\sum_{i=1}^{n} P_i \, B_{N,i}(t) w_i}{\sum_{i=1}^{n} B_{N,i}(t) w_i} = \sum_{i=1}^{n} P_i \, R_{N,i}(t).$$

They are able to exactly reproduce a huge variety of geometric forms but

NURBS: non uniform rational B-splines

In geometric modeling a B-splines generalization yield in the 90s NURBS

$$s(t) = \sum_{i=1}^{n} P_i \, B_{N,i}(t), \qquad s(t) = \frac{\sum_{i=1}^{n} P_i \, B_{N,i}(t) w_i}{\sum_{i=1}^{n} B_{N,i}(t) w_i} = \sum_{i=1}^{n} P_i \, R_{N,i}(t).$$

They are able to exactly reproduce a huge variety of geometric forms but

- need additional parameters or weights which do not have an evident geometric meaning and whose selection is often unclear;

NURBS: non uniform rational B-splines

In geometric modeling a B-splines generalization yield in the 90s NURBS

$$s(t) = \sum_{i=1}^{n} P_i\, B_{N,i}(t), \qquad s(t) = \frac{\sum_{i=1}^{n} P_i\, B_{N,i}(t) w_i}{\sum_{i=1}^{n} B_{N,i}(t) w_i} = \sum_{i=1}^{n} P_i\, R_{N,i}(t).$$

They are able to exactly reproduce a huge variety of geometric forms but

- need additional parameters or weights which do not have an evident geometric meaning and whose selection is often unclear;
- not exact description of transcendental curves like a helix or a cycloid;

NURBS: non uniform rational B-splines

In geometric modeling a B-splines generalization yield in the 90s NURBS

$$s(t) = \sum_{i=1}^{n} P_i \, B_{N,i}(t), \qquad s(t) = \frac{\sum_{i=1}^{n} P_i \, B_{N,i}(t) w_i}{\sum_{i=1}^{n} B_{N,i}(t) w_i} = \sum_{i=1}^{n} P_i \, R_{N,i}(t).$$

They are able to exactly reproduce a huge variety of geometric forms but

- need additional parameters or weights which do not have an evident geometric meaning and whose selection is often unclear;
- not exact description of transcendental curves like a helix or a cycloid;
- their rational nature make them unpleasant with respect to differentiation and integration which are crucial operators in analysis;

An alternative to NURBS: generalized B-splines

To overcome the drawbacks of NURBS an attractive alternative to the rational model is given by the so called generalized B-splines.

An alternative to NURBS: generalized B-splines

To overcome the drawbacks of NURBS an attractive alternative to the rational model is given by the so called generalized B-splines.

- While classical B-splines are piecewise functions with sections in the space of algebraic polynomials, generalized B-splines are "bell-shaped" piecewise functions with sections in more general spaces;

An alternative to NURBS: generalized B-splines

To overcome the drawbacks of NURBS an attractive alternative to the rational model is given by the so called generalized B-splines.

- While classical B-splines are piecewise functions with sections in the space of algebraic polynomials, generalized B-splines are "bell-shaped" piecewise functions with sections in more general spaces;
- With a suitable selection of such spaces (typically including trigonometric or hyperbolic functions), generalized B-splines allow exact representation of polynomial curves, conic sections, helices, cicloids;

An alternative to NURBS: generalized B-splines

To overcome the drawbacks of NURBS an attractive alternative to the rational model is given by the so called generalized B-splines.

- While classical B-splines are piecewise functions with sections in the space of algebraic polynomials, generalized B-splines are "bell-shaped" piecewise functions with sections in more general spaces;
- With a suitable selection of such spaces (typically including trigonometric or hyperbolic functions), generalized B-splines allow exact representation of polynomial curves, conic sections, helices, cicloids;
- They possess all fundamental properties of algebraic B-splines (recurrence construction, minimum support, local linear independence, knot-insertion, degree elevation, which are shared by NURBS) but behave completely similar to B-splines with respect to differentiation and integration.

An alternative to NURBS: generalized B-splines

To overcome the drawbacks of NURBS an attractive alternative to the rational model is given by the so called generalized B-splines.

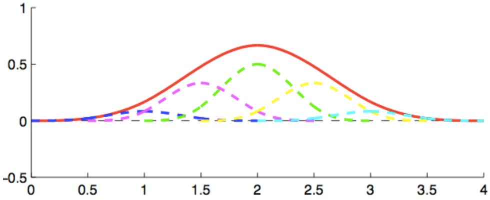
- While classical B-splines are piecewise functions with sections in the space of algebraic polynomials, generalized B-splines are "bell-shaped" piecewise functions with sections in more general spaces;
- With a suitable selection of such spaces (typically including trigonometric or hyperbolic functions), generalized B-splines allow exact representation of polynomial curves, conic sections, helices, cicloids;
- They possess all fundamental properties of algebraic B-splines (recurrence construction, minimum support, local linear independence, knot-insertion, degree elevation, which are shared by NURBS) but behave completely similar to B-splines with respect to differentiation and integration.
- In same instances they are also called Tchebycheffian B-splines

A subdivision approach for cardinal B-splines

Cardinal B-splines: refinement properties

An important property of polynomial cardinal B-splines is their refinability: they can be written as linear combination of shifts of dilates version of themself

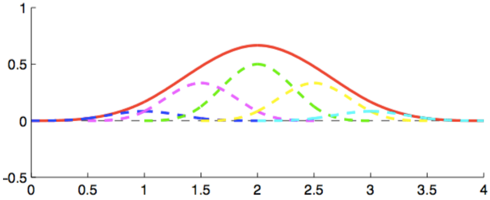
$$B_3(t) = \frac{1}{8}B_3(2t) + \frac{1}{2}B_3(2t-1) + \frac{3}{4}B_3(2t-2) + \frac{1}{2}B_3(2t-3) + \frac{1}{8}B_3(2t-4)$$



Cardinal B-splines: refinement properties

An important property of polynomial cardinal B-splines is their refinability: they can be written as linear combination of shifts of dilates version of themself

$$B_3(t) = \frac{1}{8}B_3(2t) + \frac{1}{2}B_3(2t-1) + \frac{3}{4}B_3(2t-2) + \frac{1}{2}B_3(2t-3) + \frac{1}{8}B_3(2t-4)$$



☞ Coefficients of the cubic B-spline refinement mask: $\frac{1}{8}, \frac{1}{2}, \frac{3}{4}, \frac{1}{2}, \frac{1}{8}$

The binary subdivision scheme for cardinal cubic splines

Using the refinability properties

$$B_3(t) = \sum_{j\in\mathbb{Z}} a_j^3 B_3(2t-j), \quad \text{where} \quad \mathbf{a}^3 = \cdots 0, \frac{1}{8}, \frac{1}{2}, \frac{3}{4}, \frac{1}{2}, \frac{1}{8}, 0, \cdots$$

any cubic polynomial spline can be written as

$$s(t) = \sum_{i\in\mathbb{Z}} P_i B_3(t-i) = \sum_{i\in\mathbb{Z}} P_i \sum_{j\in\mathbb{Z}} a_j^3 B_3(2(t-i)-j), \quad \text{that is}$$

$$s(t) = \sum_{i\in\mathbb{Z}} \left(\sum_{j\in\mathbb{Z}} a_{i-2j}^3 P_j \right) B_3(2t-i) = \sum_{i\in\mathbb{Z}} P_i^{(1)} B_3(2t-i)$$

where

$$P_i^{(1)} = \sum_{j\in\mathbb{Z}} a_{i-2j}^3 P_j, \quad i \in \mathbb{Z}.$$

The binary subdivision scheme for cardinal cubic splines

Iterating we can write any cubic polynomial spline as

$$s(t) = \sum_{i \in \mathbb{Z}} P_i^{(1)} B_3(2t - i) = \sum_{i \in \mathbb{Z}} P_i^{(1)} \sum_{j \in \mathbb{Z}} a_j^3 B_3(2(2t - i) - j),$$

that is

$$s(t) = \sum_{i \in \mathbb{Z}} P_i^{(2)} B_3(4t - i), \qquad \cdots \qquad s(t) = \sum_{i \in \mathbb{Z}} P_i^{(k+1)} B_3(2^k t - i),$$

where

$$P_i^{(k+1)} = \sum_{j \in \mathbb{Z}} a_{i-2j}^3 P_j^{(k)}, \quad i \in \mathbb{Z} \quad \Leftrightarrow \quad \mathbf{P}^{(k+1)} = S_{\mathbf{a}^3} \mathbf{P}^{(k)}, \quad k \geq 0.$$

The binary subdivision scheme for cardinal cubic splines

Iterating we can write any cubic polynomial spline as

$$s(t) = \sum_{i\in\mathbb{Z}} P_i^{(1)} B_3(2t-i) = \sum_{i\in\mathbb{Z}} P_i^{(1)} \sum_{j\in\mathbb{Z}} a_j^3 B_3(2(2t-i)-j),$$

that is

$$s(t) = \sum_{i\in\mathbb{Z}} P_i^{(2)} B_3(4t-i), \qquad \cdots \qquad s(t) = \sum_{i\in\mathbb{Z}} P_i^{(k+1)} B_3(2^k t-i),$$

where

$$P_i^{(k+1)} = \sum_{j\in\mathbb{Z}} a_{i-2j}^3 P_j^{(k)}, \quad i\in\mathbb{Z} \quad \Leftrightarrow \quad \mathbf{P}^{(k+1)} = S_{\mathbf{a}^3}\mathbf{P}^{(k)}, \quad k \geq 0.$$

☞ Since the support of $B_3(2^k\cdot)$ shrink for k large enough the coefficients $\mathbf{P}^{(k)}$ are a good discrete representation of s.

The binary subdivision scheme for degree N splines

More in general, since $B_N(t) = \sum_{i\in\mathbb{Z}} a_j^N B_N(2t-j)$, starting with an initial sequence of points $\mathbf{P}^{(0)}$, the iterative computation of sequence of points

$$\mathbf{P}^{(k+1)} = S_{\mathbf{a}^N}\mathbf{P}^{(k)}, \quad P_i^{(k+1)} = \sum_{j\in\mathbb{Z}} a_{i-2j}^N P_j^{(k)}, \; a_i^N = \frac{1}{2^N}\left(\begin{array}{c} N+1 \\ i \end{array}\right), \; i = 0, \cdots, N$$

defines the so called *subdivision scheme for cardinal degree-N splines* whose limit is a degree N spline associated with the initial control points $\mathbf{P}^{(0)}$ and often denoted as $S_{\mathbf{a}^N}^{\infty}\mathbf{P}^{(0)}$.

The binary subdivision scheme for degree N splines

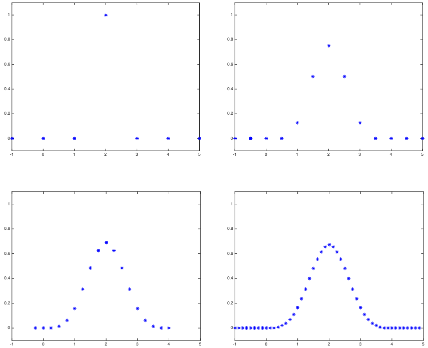
More in general, since $B_N(t) = \sum_{i \in \mathbb{Z}} a_j^N B_N(2t - j)$, starting with an initial sequence of points $\mathbf{P}^{(0)}$, the iterative computation of sequence of points

$$\mathbf{P}^{(k+1)} = S_{\mathbf{a}^N} \mathbf{P}^{(k)}, \quad P_i^{(k+1)} = \sum_{j \in \mathbb{Z}} a_{i-2j}^N P_j^{(k)}, \; a_i^N = \frac{1}{2^N} \left(\begin{array}{c} N+1 \\ i \end{array} \right), \; i = 0, \cdots, N$$

defines the so called *subdivision scheme for cardinal degree-N splines* whose limit is a degree N spline associated with the initial control points $\mathbf{P}^{(0)}$ and often denoted as $S_{\mathbf{a}^N}^{\infty} \mathbf{P}^{(0)}$.

☞ Any degree-N B-splines is the *basic limit function of the corresponding subdivision scheme* when starting with the sequence $\delta = 0,\ 0,\ 1,\ 0,\ 0.$

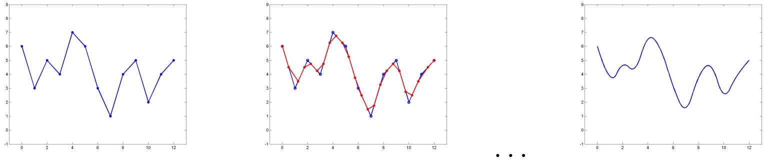
Binary subdivision scheme



Binary subdivision scheme

This subdivision idea allows us to define other type of refinable basis functions not necessarily piecewise polynomial but useful "generalization" of B-splines.

$$S_{\mathbf{a}} \quad \Leftrightarrow \quad \begin{cases} \text{Input} \quad \mathbf{P}^{(0)} \\ \text{For} \quad k = 0, 1, \cdots \\ \qquad\qquad \mathbf{P}^{(k+1)} := S_{\mathbf{a}} \mathbf{P}^{(k)} \end{cases}$$



Binary subdivision scheme

- A subdivision scheme is essentially given by the subdivision coefficients or subdivision mask, say $\mathbf{a} = \{a_i, \in \mathbb{R} \; i \in \mathbb{Z}\}$ (usually finite);

Binary subdivision scheme

- A subdivision scheme is essentially given by the subdivision coefficients or subdivision mask, say $\mathbf{a} = \{a_i, \in \mathbb{R}\ i \in \mathbb{Z}\}$ (usually finite);
- If convergent a sub. defines a compactly supported basic $\phi = S_{\mathbf{a}}^{\infty}\delta$ as limit of the subdivision process starting from $\delta = 0,\ 0,\ 1,\ 0,\ 0;$

Binary subdivision scheme

- A subdivision scheme is essentially given by the subdivision coefficients or subdivision mask, say $\mathbf{a} = \{a_i, \in \mathbb{R}\ i \in \mathbb{Z}\}$ (usually finite);
- If convergent a sub. defines a compactly supported basic $\phi = S_{\mathbf{a}}^{\infty}\delta$ as limit of the subdivision process starting from $\delta = 0,\ 0,\ 1,\ 0,\ 0;$
- Mostly, this function is *not defined analytically*. But:

Binary subdivision scheme

- A subdivision scheme is essentially given by the subdivision coefficients or subdivision mask, say $\mathbf{a} = \{a_i, \in \mathbb{R}\ i \in \mathbb{Z}\}$ (usually finite);
- If convergent a sub. defines a compactly supported basic $\phi = S_{\mathbf{a}}^{\infty}\delta$ as limit of the subdivision process starting from $\delta = 0,\ 0,\ 1,\ 0,\ 0;$
- Mostly, this function is *not defined analytically*. But:
  - $\phi = \sum_{i\in\mathbb{Z}} a_i \phi(2\cdot - i);$
  - partition of unity $\sum_{i\in\mathbb{Z}} \phi(\cdot - i) = 1;$
  - .....

Non-stationary or level dependent subdivision scheme

This subdivision recursion idea can also be implemented by changing the set of coefficients at each level.

$$\{S_{\mathbf{a}^{(k)}},\ k \geq 0\} \quad \Leftrightarrow \quad \begin{cases} \text{Input} \quad \mathbf{P}^{(0)} \\ \text{For} \quad k = 0, 1, \cdots \\ \qquad \mathbf{P}^{(k+1)} := S_{\mathbf{a}^{(k)}} \mathbf{f}^{(k)} \quad \text{level dep. rules} \end{cases}$$

Non-stationary or level dependent subdivision scheme

This subdivision recursion idea can also be implemented by changing the set of coefficients at each level.

$$\{S_{\mathbf{a}^{(k)}},\ k \geq 0\} \quad \Leftrightarrow \quad \begin{cases} \text{Input} \quad \mathbf{P}^{(0)} \\ \text{For} \quad k = 0, 1, \cdots \\ \qquad \mathbf{P}^{(k+1)} := S_{\mathbf{a}^{(k)}} \mathbf{f}^{(k)} \quad \text{level dep. rules} \end{cases}$$

☞ In that case we have a family of basic limit functions

$$\{\phi_m,\ m \geq 0\}, \quad \phi_m = \lim_{k\to\infty} S_{\mathbf{a}^{(k)}} \cdots S_{\mathbf{a}^{(m)}} \delta.$$

Non-stationary or level dependent subdivision scheme

This subdivision recursion idea can also be implemented by changing the set of coefficients at each level.

$$\{S_{\mathbf{a}^{(k)}},\ k \geq 0\} \quad \Leftrightarrow \quad \begin{cases} \text{Input} \quad \mathbf{P}^{(0)} \\ \text{For} \quad k = 0, 1, \cdots \\ \qquad \mathbf{P}^{(k+1)} := S_{\mathbf{a}^{(k)}} \mathbf{f}^{(k)} \end{cases} \quad \text{level dep. rules}$$

☞ In that case we have a family of basic limit functions

$$\{\phi_m,\ m \geq 0\}, \quad \phi_m = \lim_{k \to \infty} S_{\mathbf{a}^{(k)}} \cdots S_{\mathbf{a}^{(m)}} \delta.$$

☞ They still satisfy a refinability property: $\phi_m = \sum_{i \in \mathbb{Z}} a_i^{(m)} \phi_{m+1}(2 \cdot -i).$

The role of symbols in subdivision schemes

How to prove convergence or other properties of a subdivision scheme?

The role of symbols in subdivision schemes

How to prove convergence or other properties of a subdivision scheme?

An advantage of the cardinal framework is that we can make use of standard mathematical tools of signal processing which simplify all formulations and derivations considerably.

The role of symbols in subdivision schemes

How to prove convergence or other properties of a subdivision scheme?

An advantage of the cardinal framework is that we can make use of standard mathematical tools of signal processing which simplify all formulations and derivations considerably.

In the subdivision community a similar notion to the z-trasform is used: the subdivision symbol associated with the subdivision mask .

Subdivision symbol

The symbol of a subdivision mask $\{a_i,\ i \in \mathbb{Z}\}$ is the Laurent polynomial

$$a(z) = \sum_{i \in \mathbb{Z}} a_i\, z^i, \qquad z \in \mathbb{C} \setminus \{0\}.$$

The role of symbols in subdivision schemes

The role of symbols in subdivision schemes

- The mask symbol $a(z)$ fully identifies a stationary subdivision scheme;

The role of symbols in subdivision schemes

- The mask symbol $a(z)$ fully identifies a stationary subdivision scheme;
- In the level dependent case the sequence of symbols $\{a^{(k)}(z),\ k \geq 0\}$ associated with the sequence of masks identifies the subdivision scheme;

The role of symbols in subdivision schemes

- The mask symbol $a(z)$ fully identifies a stationary subdivision scheme;
- In the level dependent case the sequence of symbols $\{a^{(k)}(z),\ k \geq 0\}$ associated with the sequence of masks identifies the subdivision scheme;
- Many of the properties of a subdivision scheme can be easily checked using algebraic conditions on the subdivision symbols;

The role of symbols in subdivision schemes

- The mask symbol $a(z)$ fully identifies a stationary subdivision scheme;
- In the level dependent case the sequence of symbols $\{a^{(k)}(z),\ k \geq 0\}$ associated with the sequence of masks identifies the subdivision scheme;
- Many of the properties of a subdivision scheme can be easily checked using algebraic conditions on the subdivision symbols;
- This is also true for the properties of the basic limit functions

Exponential-polynomial B-splines:
a "perfect" base for the space of exponential-polynomial splines

Cardinal exponential-polynomial B-splines

Exponential-polynomial Splines $\mathcal{S}(EP_\Gamma)$

Exp-polynomial splines are generalized splines whose sections are in EP_Γ

Cardinal exponential-polynomial B-splines

Exponential-polynomial Splines $\mathcal{S}(EP_\Gamma)$

Exp-polynomial splines are generalized splines whose sections are in EP_Γ

Exponential-polynomials

Let $n \in \mathbb{N}$ and let $\Gamma = \{(\theta_1, \tau_1), \ldots, (\theta_n, \tau_n)\}$ with $\theta_i \in \mathbb{R} \cup i\mathbb{R}$, $\theta_i \neq \theta_j$ if $i \neq j$ and $\tau_i \in \mathbb{N},\ i = 1, \cdots, n$. We define the space of exponential polynomials EP_Γ

$$EP_\Gamma = \text{span}\{\ x^{r_i}\ e^{\theta_i x},\ r_i = 0, \cdots, \tau_i - 1,\ i = 1, \cdots, n\}.$$

Cardinal exponential-polynomial B-splines

Exponential-polynomial Splines $\mathcal{S}(EP_\Gamma)$

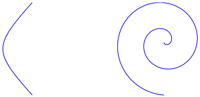
Exp-polynomial splines are generalized splines whose sections are in EP_Γ

Exponential-polynomials

Let $n \in \mathbb{N}$ and let $\Gamma = \{(\theta_1, \tau_1), \ldots, (\theta_n, \tau_n)\}$ with $\theta_i \in \mathbb{R} \cup \mathrm{i}\mathbb{R}$, $\theta_i \neq \theta_j$ if $i \neq j$ and $\tau_i \in \mathbb{N}$, $i = 1, \cdots, n$.
We define the space of exponential polynomials EP_Γ

$$EP_\Gamma = \mathrm{span}\{\, x^{r_i}\, e^{\theta_i x},\ r_i = 0, \cdots, \tau_i - 1,\ i = 1, \cdots, n\}.$$

☞ EP_Γ is a linear space of dim. $N = \sum_{\ell=1}^{n} \tau_\ell$.
It contains polynomials as special case.



Cardinal Exponential B-splines and Green functions

The way exponential B-splines are constructed is simple in the "first order " case $\Gamma = \{(\theta, 1)\}$ by the help of Green functions of the differential operator

Cardinal Exponential B-splines and Green functions

The way exponential B-splines are constructed is simple in the "first order " case $\Gamma = \{(\theta, 1)\}$ by the help of Green functions of the differential operator

- Take the first oder diff. operator $D_\theta = D - \theta\, I$;

Cardinal Exponential B-splines and Green functions

The way exponential B-splines are constructed is simple in the "first order " case $\Gamma = \{(\theta, 1)\}$ by the help of Green functions of the differential operator

- Take the first oder diff. operator $D_\theta = D - \theta\, I$;
- Take its Green function $\rho_\theta = e^{\theta t}\, 1_+(t)$;

Cardinal Exponential B-splines and Green functions

The way exponential B-splines are constructed is simple in the "first order " case $\Gamma = \{(\theta, 1)\}$ by the help of Green functions of the differential operator

- Take the first oder diff. operator $D_\theta = D - \theta\, I$;
- Take its Green function $\rho_\theta = e^{\theta t}\, 1_+(t)$;
- Construct the discrete version of D_θ, Δ_θ, such that $\Delta_\theta f = f - e^\theta f(\cdot - 1)$;

Cardinal Exponential B-splines and Green functions

The way exponential B-splines are constructed is simple in the "first order " case $\Gamma = \{(\theta, 1)\}$ by the help of Green functions of the differential operator

- Take the first oder diff. operator $D_\theta = D - \theta\, I$;
- Take its Green function $\rho_\theta = e^{\theta t}\, 1_+(t)$;
- Construct the discrete version of D_θ, Δ_θ, such that $\Delta_\theta f = f - e^\theta f(\cdot - 1)$;
- Define the "first order" exponential B-spline as $B_\theta(t) = \Delta_\theta\left(\rho_\theta(t)\right)$.

Cardinal Exponential B-splines and Green functions

The way exponential B-splines are constructed is simple in the "first order " case $\Gamma = \{(\theta, 1)\}$ by the help of Green functions of the differential operator

- Take the first oder diff. operator $D_\theta = D - \theta\, I$;
- Take its Green function $\rho_\theta = e^{\theta t}\, 1_+(t)$;
- Construct the discrete version of D_θ, Δ_θ, such that $\Delta_\theta f = f - e^\theta f(\cdot - 1)$;
- Define the "first order" exponential B-spline as $B_\theta(t) = \Delta_\theta\, (\rho_\theta(t))$.

Cardinal exponential B-spline for $\Gamma = \{(\theta_1, \tau_1), \ldots, (\theta_n, \tau_n)\}$ are defined as

Definition of exponential B-splines

$$B_{N,\Gamma} = \underbrace{B_{\theta_1}}_{\tau_1-\text{times}} * \underbrace{B_{\theta_2}}_{\tau_2-\text{times}} * \cdots * \underbrace{B_{\theta_n}}_{\tau_n-\text{times}}, \qquad B_\theta(t) = \rho_\theta(t) - e^\theta \rho_\theta(t-1).$$

Cardinal exponential B-splines and subdivision schemes

Cardinal exponential B-splines can be defined via level dependent subdivision schemes by directly constructing their symbols:

Cardinal exponential B-splines and subdivision schemes

Cardinal exponential B-splines can be defined via level dependent subdivision schemes by directly constructing their symbols:

Symbols of exponential B-splines

For $\Gamma = \{(\theta_1, \tau_1), \ldots, (\theta_n, \tau_n)\}$ the symbols of exponential B-splines of order $N = \sum_{\ell=1}^{n} \tau_\ell$, are given by $B_{N,\Gamma}^{(k)}(z) = F_N^{(k)} \prod_{\ell=1}^{n} (1 + e^{\frac{\theta_\ell}{2^{k+1}}} z)^{\tau_\ell}, \quad k \geq 0.$

Cardinal exponential B-splines and subdivision schemes

Cardinal exponential B-splines can be defined via level dependent subdivision schemes by directly constructing their symbols:

Symbols of exponential B-splines

For $\Gamma = \{(\theta_1, \tau_1), \ldots, (\theta_n, \tau_n)\}$ the symbols of exponential B-splines of order $N = \sum_{\ell=1}^{n} \tau_\ell$, are given by $$B_{N,\Gamma}^{(k)}(z) = F_N^{(k)} \prod_{\ell=1}^{n} (1 + e^{\frac{\theta_\ell}{2^{k+1}}} z)^{\tau_\ell}, \quad k \geq 0.$$

- The subdivision scheme based on $\{B_{N,\Gamma}^{(k)}(z),\ k \geq 0\}$ converges;

Cardinal exponential B-splines and subdivision schemes

Cardinal exponential B-splines can be defined via level dependent subdivision schemes by directly constructing their symbols:

Symbols of exponential B-splines

For $\Gamma = \{(\theta_1, \tau_1), \ldots, (\theta_n, \tau_n)\}$ the symbols of exponential B-splines of order $N = \sum_{\ell=1}^{n} \tau_\ell$, are given by $$B_{N,\Gamma}^{(k)}(z) = F_N^{(k)} \prod_{\ell=1}^{n} (1 + e^{\frac{\theta_\ell}{2^{k+1}}} z)^{\tau_\ell}, \quad k \geq 0.$$

- The subdivision scheme based on $\{B_{N,\Gamma}^{(k)}(z),\ k \geq 0\}$ converges;
- The corresponding basic limit function is $B_{N,\Gamma}$ (a base for $\mathcal{S}(EP_\Gamma)$);

Cardinal exponential B-splines and subdivision schemes

Cardinal exponential B-splines can be defined via level dependent subdivision schemes by directly constructing their symbols:

Symbols of exponential B-splines

For $\Gamma = \{(\theta_1, \tau_1), \ldots, (\theta_n, \tau_n)\}$ the symbols of exponential B-splines of order $N = \sum_{\ell=1}^n \tau_\ell$, are given by $B_{N,\Gamma}^{(k)}(z) = F_N^{(k)} \prod_{\ell=1}^{n} (1 + e^{\frac{\theta_\ell}{2^{k+1}}} z)^{\tau_\ell}, \quad k \geq 0.$

- The subdivision scheme based on $\{B_{N,\Gamma}^{(k)}(z),\ k \geq 0\}$ converges;
- The corresponding basic limit function is $B_{N,\Gamma}$ (a base for $\mathcal{S}(EP_\Gamma)$);
- When $\Gamma = \{(0, N)\}$ the symbols reduce the symbol of polynomial B-spline of order N (degree $N-1$) given by $B_{N-1}(z) = \frac{1}{2^{N-1}}(1+z)^N$.

Exponential B-splines: four dimension ("cubic") case

Support: $[0, 4]$

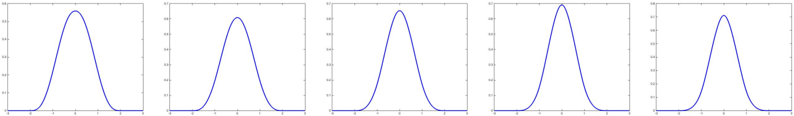
Knot vector: $\{1, 2, 3, \}, \ \mathcal{M} = (1, 1, 1)$

Pieces in the space: $\{1, t, e^{\theta t}, e^{-\theta t}\}, \ \theta \in \mathbb{R} \cup i\mathbb{R}$

$$P_{2i}^{(k+1)} = \frac{1}{4(v^{(k)}+1)} P_{i-1}^{(k)} + \frac{1+2v^{(k)}}{2(v^{(k)}+1)} P_i^{(k)} + \frac{1}{4(v^{(k)}+1)} P_{i+1}^{(k)}$$

$$P_{2i+1}^{(k+1)} = \frac{1}{2} P_i^{(k)} + \frac{1}{2} P_{i+1}^{(k)}$$

For $v^{(k)} = \frac{1}{2}\left(e^{\frac{\theta}{2^{k+1}}} + e^{\frac{-\theta}{2^{k+1}}}\right), \quad v^{(k)} = \sqrt{\frac{1 + v^{(k-1)}}{2}}, \quad k \geq 0, \quad v^{(-1)} > -1$



Basic limit functions for different values of $v^{(-1)} \in \{-0.9, \ -0.5, \ 0.5, \ 0.25, \ 0.45\}$

Cardinal Exponential B-splines

Are B-splines and Exponential B-splines enough?

Beyond B-splines and Exponential B-splines

Beyond B-splines and Exponential B-splines

- They both have low approximation order;

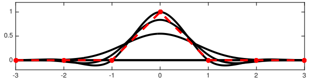
Beyond B-splines and Exponential B-splines

- They both have low approximation order;
- The approximation order of (any) subdivision scheme is strictly connected with its generation/reproduction properties;

Beyond B-splines and Exponential B-splines

- They both have low approximation order;
- The approximation order of (any) subdivision scheme is strictly connected with its generation/reproduction properties;
- By subdivision we can define pseudo-splines (pol. and exp. pol.) that have higher approximation order and generalize B-splines;

- In both cases the first member of the family of pseudo-splines is a B-splines the last an interpolatory basis function
- Pseudo-splines give a wide range of choices balancing approximation order, length of the support and regularity;
- are good candidate for application in different contexts.



Thanks for your attention

Thanks for your attention

Happy birthday Michael!