Subdivision Schemes for Geometric Modelling

Kai Hormann

University of Lugano

Outline

- Sep 5 – Subdivision as a linear process
  - basic concepts, notation, subdivision matrix
- ***Sep 6 – The Laurent polynomial formalism***
  - ***algebraic approach, polynomial reproduction***
- Sep 7 – Smoothness analysis
  - Hölder regularity of limit by spectral radius method
- Sep 8 – Subdivision surfaces
  - overview of most important schemes & properties

Schemes considered so far

- polygon subdivision
  - even stencil [1] odd stencil [1, 1] / 2
  - interpolatory C^0 limit curve (piecewise linear)
- cubic B-spline subdivision
  - even stencil [1, 6, 1] / 8 odd stencil [1, 1] / 2
  - approximating C^2 limit curve (piecewise cubic)
- 4-point scheme
  - even stencil [1] odd stencil [–1, 9, 9, –1] / 16
  - interpolatory C^1 limit curve (non-polynomial)

Primal and dual schemes

- primal schemes
  - even and odd stencil are both symmetric
  - even stencil: modify old points
  - odd stencil: insert new points (between old points)
  - point $\rightarrow$ point, edge $\rightarrow$ point
- dual schemes
  - insert two new points (between old points)
  - discard old points
  - point $\rightarrow$ edge, edge $\rightarrow$ edge

Chaikin's corner cutting

Example

$$p_{2i}^{j+1} = \tfrac{3}{4}\, p_i^j + \tfrac{1}{4}\, p_{i+1}^j, \qquad p_{2i+1}^{j+1} = \tfrac{1}{4}\, p_i^j + \tfrac{3}{4}\, p_{i+1}^j$$

- even stencil $[\underline{3}, 1]/4$ odd stencil $[\underline{1}, 3]/4$
- invariant neighbourhood size: 2
- local subdivision matrix $S = \begin{pmatrix} \frac{3}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{3}{4} \end{pmatrix}$
- eigenvalues: 1, ½
- limit stencil $[\underline{1}, 1]/2$

- note: even/odd stencil are symmetric to each other

The subdivision mask

- refinement rules
  - even stencil $[\ldots, \alpha_2, \alpha_1, \alpha_0, \alpha_{-1}, \alpha_{-2}, \ldots]$
  - odd stencil $[\ldots, \beta_2, \beta_1, \beta_0, \beta_{-1}, \beta_{-2}, \ldots]$
  - rules $p_{2i}^{j+1} = \sum_k \alpha_k p_{i-k}^j, \qquad p_{2i+1}^{j+1} = \sum_k \beta_k p_{i-k}^j$
  - combine stencils into ***subdivision mask***

$$\boldsymbol{a} = [\ldots, \alpha_2, \beta_1, \alpha_1, \beta_0, \alpha_0, \beta_{-1}, \alpha_{-1}, \beta_{-2}, \alpha_{-2}, \ldots]$$
$$= [\ldots, a_4, a_3, a_2, a_1, a_0, a_{-1}, a_{-2}, a_{-3}, a_{-4}, \ldots]$$

  - one single refinement rule $p_i^{j+1} = \sum_k a_{i-2k} p_k^j$

Masks of the schemes so far

- polygon subdivision

$$a = [1, \underline{2}, 1]/2$$

- Chaikin's corner cutting

$$a = [1, \underline{3}, 3, 1]/4$$

- cubic B-spline scheme

$$a = [1, 4, \underline{6}, 4, 1]/8$$

- 4-point scheme

$$a = [-1, 0, 9, \underline{16}, 9, 0, -1]/16$$

Laurent polynomials

- **Definition**

  given a sequence $\boldsymbol{c} = \{c_i : i \in \mathbb{Z}\}$, we call

  $$c(z) = \sum_{i \in \mathbb{Z}} c_i z^i$$

  the *z-transform* of $\boldsymbol{c}$

- if $\boldsymbol{c}$ is finitely supported, then $c(z)$ is a ***Laurent polynomial***
- for a subdivision scheme with mask $\boldsymbol{a}$, we call the Laurent polynomial $a(z)$ the *symbol* of the scheme

Symbols of the schemes so far

- polygon subdivision

  $\boldsymbol{a} = [1, \underline{2}, 1]/2$ $\qquad a(z) = \frac{1}{2z}(1+z)^2$

- Chaikin's corner cutting

  $\boldsymbol{a} = [1, \underline{3}, 3, 1]/4$ $\qquad a(z) = \frac{1}{4z^2}(1+z)^3$

- cubic B-spline scheme

  $\boldsymbol{a} = [1, 4, \underline{6}, 4, 1]/8$ $\qquad a(z) = \frac{1}{8z^2}(1+z)^4$

- 4-point scheme

  $\boldsymbol{a} = [-1, 0, 9, \underline{16}, 9, 0, -1]/16$

  $$a(z) = \frac{-1+4z-z^2}{16z^3}(1+z)^4$$

Symbols and convergence

- *necessary condition* for convergence
  - coefficients of even/odd stencil sum to 1

$$\sum_{i\in\mathbb{Z}} \alpha_i = \sum_{i\in\mathbb{Z}} a_{2i} = 1, \qquad \sum_{i\in\mathbb{Z}} \beta_i = \sum_{i\in\mathbb{Z}} a_{2i+1} = 1$$

  - equivalent to

$$a(-1) = 0, \qquad a(1) = 2$$

  - implies

$$a(z) = (1+z)b(z), \qquad b(1) = 1$$

Subdivision in terms of symbols

- consider the z-transform $p^j(z)$ of the data $\{p_i^j\}$ at level j, then the refinement rule

$$p_i^{j+1} = \sum_k a_{i-2k} p_k^j$$

can be written as

$$p^{j+1}(z) = a(z) p^j(z^2)$$

- very neat and compact way of writing the rule
  - note: we are not interested in the polynomials as such, but rather in their coefficients

Subdivision of differences

- let Δ denote the (finite) ***difference operator*** on sequences

$$\Delta \boldsymbol{c} = \{c_i - c_{i-1} : i \in \mathbb{Z}\}$$

- if $a(z) = (1+z)\, b(z)$ is the symbol of a convergent subdivision scheme, then $b(z)$ is the symbol of the scheme for the differences

$$\Delta p_i^{j+1} = \sum_k b_{i-2k} \Delta p_k^j$$

Subdivision of differences

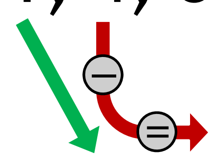
- **Example**
  - cubic B-spline scheme

$$a(z) = \frac{1}{8z^2}(1+z)^4 = (1+z)b(z), \qquad b(z) = \frac{1}{8z^2}(1+z)^3$$

  - corresponding scheme for the differences
    - mask $b = [1,3,3,1]/8$
    - local subdivision matrix $S = \begin{pmatrix} \frac{3}{8} & \frac{1}{8} \\ \frac{1}{8} & \frac{3}{8} \end{pmatrix}$
    - eigenvalues: ½, ¼
    - maps differences (edge vectors) to 0
    - can we conclude that the scheme converges ?

Symbols and masks

- multiplying a symbol by $(1+z)$
  - write down the mask [1, 2, 3, –1]
  - write it again, shifted to the left [1, 2, 3, –1]
  - add both rows [1, 3, 5, 2, –1]
- dividing a symbol by $(1+z)$
  - check, if sum of odd/even coefficients is the same
  - write down the mask [1, 4, 6, 4, 1]/8
  - copy first coefficient, then take differences



[1, 3, 3, 1]/8

Polynomial sequences

- **Definition**

  a sequence $\boldsymbol{c} = \{c_i : i \in \mathbb{Z}\}$ is called ***polynomial of degree d***, if there exists some polynomial π of degree d such that $c_i = \pi(i)$ for all $i \in \mathbb{Z}$

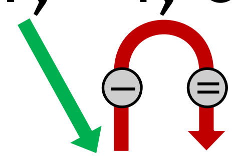
- **Examples**
  - (…, 3, 3, 3, 3, …) is of degree 0
  - (…, –2, 1, 4, 7, …) is of degree 1

- if $\boldsymbol{c}$ is polynomial of degree d, then

$$\Delta^{d+1}\boldsymbol{c} = \mathbf{0} \quad \Leftrightarrow \quad (1-z)^{d+1}c(z) = 0$$

Symbols and masks

- multiplying a symbol by $(1-z)$
  - write down the mask [1, –2, 3, –1]
  - write it negated, shifted to left [–1, 2, –3, 1]
  - add both rows [–1, 3, –5, 4, –1]
- dividing a symbol by $(1-z)$
  - check, if coefficients add to zero
  - write down the mask [1, –4, 6, –4, 1]/8
  - copy first coefficient negated, then take differences



[–1, 3, –3, 1]/8

Polynomial generation

- suppose the initial data $\boldsymbol{p}^0$ is polynomial of degree d and the symbol of the scheme is

  $$a(z) = (1+z)^{d+1} b(z) \qquad (\star)$$

  then the refined data $\boldsymbol{p}^j$ at any level j is polynomial of degree d, and so is the limit curve
- in fact, condition $(\star)$ is necessary and sufficient for the scheme being able to ***generate polynomials of degree d***

Polynomial generation

- **Example**
  - cubic B-spline scheme $\boldsymbol{a} = [\,1, 4, 6, 4, 1\,]/8$
  - symbol $a(z) = \frac{1}{8z^2}(1+z)^4$
  - generates polynomials up to degree 3
  - initial data $\boldsymbol{p}^0 = (\ldots, 9, 4, 1, 0, 1, 4, 9, \ldots)$
  - refined data $\boldsymbol{p}^1 = (\ldots, 10, 5, 2, 1, 2, 5, 10, \ldots)/4$

    $\boldsymbol{p}^2 = (\ldots, 14, 9, 6, 5, 6, 9, 14, \ldots)/16$
  - limit curve is a quadratic polynomial, but not the one from which the initial data was sampled
  - no polynomial reproduction

The functional setting

- initial data $f^0 = (f_i^0)_{i\in\mathbb{Z}}$
- mask $a = (a_i)_{i\in\mathbb{Z}}$
- refinement rule

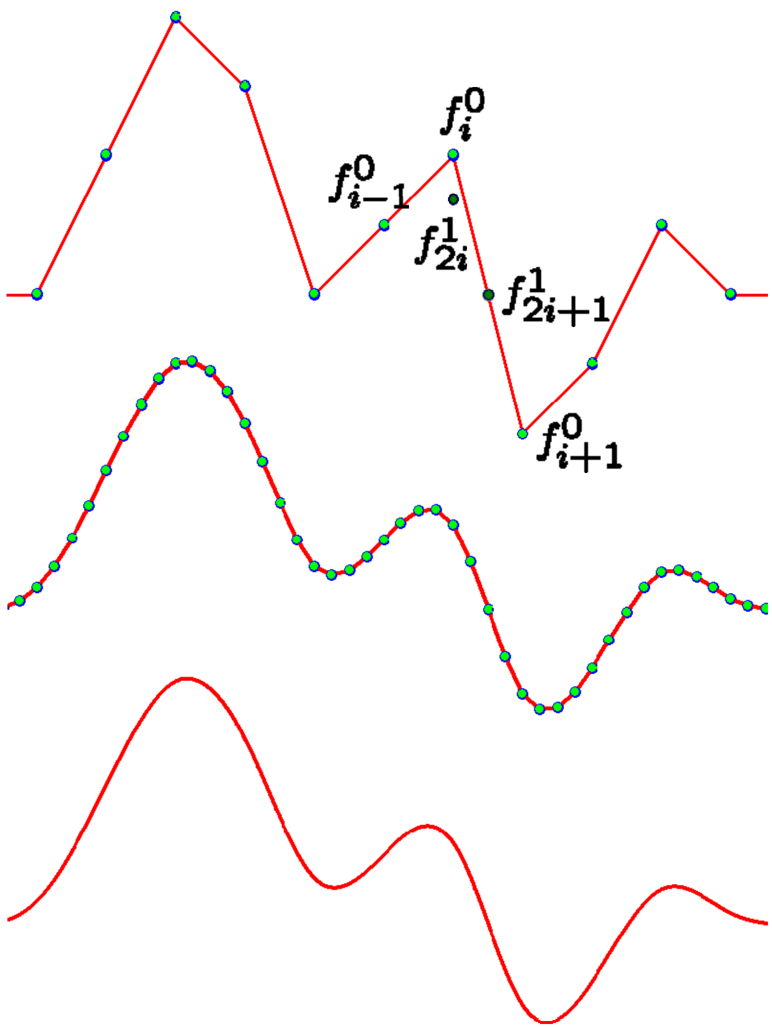
$$f_i^{j+1} = \sum_k a_{i-2k} f_k^j$$

- parameter values $(t_i^j)_{i\in\mathbb{Z},j\in\mathbb{N}}$
- piecewise linear functions

$$F^j \text{ with } F^j(t_i^j) = f_i^j$$

- limit function

$$S_a^\infty f^0 = \lim_{j\to\infty} F^j$$



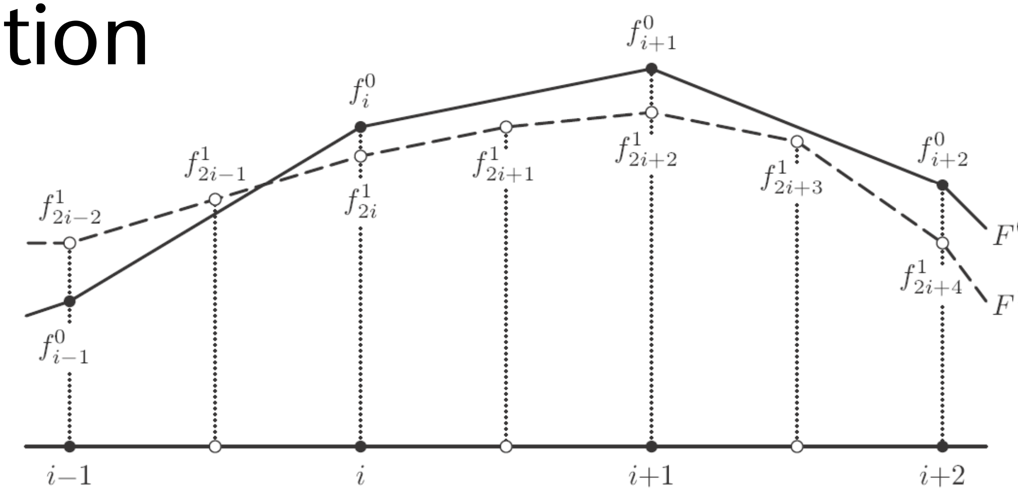
Parameterizations

- ***primal*** **parameterization**

$$t_i^j = i/2^j$$

  - for ***primal*** schemes with ***odd*** symmetry

$$a_{-i} = a_i$$

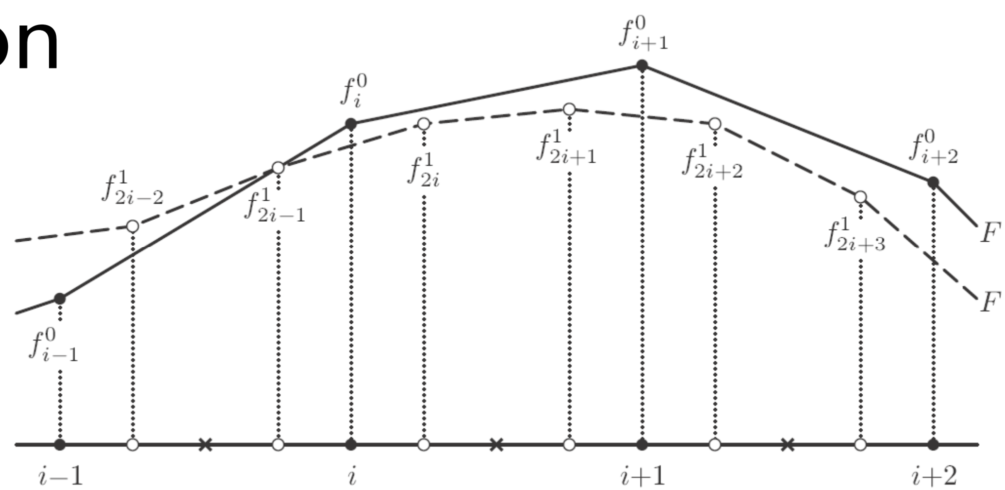


- ***dual*** **parameterization**

$$t_i^j = \frac{1}{2} + (i - \frac{1}{2})/2^j$$

  - for ***dual*** schemes with ***even*** symmetry

$$a_{-i} = a_{i-1}$$



Linear reproduction

Example

- Chaikin's corner cutting $\boldsymbol{a} = [\,1, \underline{3}, 3, 1\,]/4$
- initial data $f_i^0 = \pi(t_i^0)$, for $\pi(x) = x$
- piecewise linear functions F^j with $F^j(t_i^j) = f_i^j$
- does the scheme reproduce π, i.e. $\lim_{j\to\infty} F^j = \pi$?
- primal parameterization

$$t_i^j = i/2^j \qquad \Rightarrow \qquad \Big(\lim_{j\to\infty} F^j\Big)(x) = x + \tfrac{1}{2}$$

- dual parameterization

$$t_i^j = \tfrac{1}{2} + (i - \tfrac{1}{2})/2^j \qquad \Rightarrow \qquad \Big(\lim_{j\to\infty} F^j\Big)(x) = x$$

Linear reproduction

- for any scheme that ***generates*** linear functions
  - with symbol $a(z) = (1+z)^2 b(z)$
  - let $\tau = a'(1)/2$
  - attach data f_i^j to parameter $t_i^j = -\tau + (i+\tau)/2^j$
  - then the scheme also ***reproduces*** linear functions
- **Examples**
  - cubic B-spline scheme $a'(1) = 0 \;\Rightarrow\; t_i^j = i/2^j$
  - Chaikin's corner cutting

$$a'(1) = -1 \;\Rightarrow\; t_i^j = \tfrac{1}{2} + (i - \tfrac{1}{2})/2^j$$

Polynomial reproduction

- reproduction of degree d requires generation of degree d
- generation of degree d is equivalent to
  - $a(z) = (1+z)^{d+1}b(z) \quad \Leftrightarrow \quad$ zero of order $d+1$ at $z=-1$
- **Theorem**

  the subdivision scheme with symbol $a(z)$ reproduces polynomials of degree d w.r.t. the parameterization with $\tau = a'(1)/2$, if and only if

$$a^{(k)}(-1) = 0, \quad a^{(k)}(1) = 2\prod_{j=0}^{k-1}(\tau - j), \quad k = 0,\dots,d$$

Polynomial reproduction

- **Examples**
  - polygon subdivision $\boldsymbol{a} = [1, \underline{2}, 1]/2$
    - linear reproduction w.r.t. primal parameterization
  - Chaikin's corner cutting $\boldsymbol{a} = [1, \underline{3}, 3, 1]/4$
    - linear reproduction w.r.t. dual parameterization
  - general primal 3-point $\boldsymbol{a} = [w, ½, \underline{1-2w}, ½, w]$
    - linear reproduction w.r.t. primal parameterization
  - an unsymmetric scheme $\boldsymbol{a} = [-1, 0, 6, \underline{8}, 3]/8$
    - quadratic reproduction w.r.t. primal parameterization
    - note: this is an interpolating scheme!

Approximation order

- a scheme that reproduces polynomials of degree d has ***approximation order*** d+1
  - given a sufficiently smooth function F
  - take the initial data $f_i^0 = F(ih)$
  - then
$$\|F - S_{\boldsymbol{a}}^{\infty}\boldsymbol{f}^0\| \leq Ch^{d+1}$$
  - the constant C does not depend on h

Summary

- combining even/odd stencils into the mask
  - one common subdivision rule
- use z-transform to turn mask into the symbol
  - formally, the symbol is a Laurent polynomial
  - transform data in the same way
  - yields an algebraic way to describe subdivison
  - necessary convergence condition: $a(-1)=0$, $a(1)=2$
- hands-on rules for multiplying and dividing masks by $(1+z)$ and by $(1-z)$

Summary

- polynomial generation of degree d

$$a(z) = (1+z)^{d+1} b(z)$$

$$\Leftrightarrow \quad a^{(k)}(-1) = 0 \quad \text{for} \quad k = 0, \dots, d$$

- polynomial reproduction of degree d
  - requires polynomial generation of degree d
  - depends on the correct parameterization

$$t_i^j = -\tau + (i + \tau)/2^j \quad \text{with} \quad \tau = a'(1)/2$$

  - correct values of the d derivatives of a at $z=1$

$$a^{(k)}(1) = 2 \prod_{j=0}^{k-1} (\tau - j) \quad \text{for} \quad k = 0, \dots, d$$