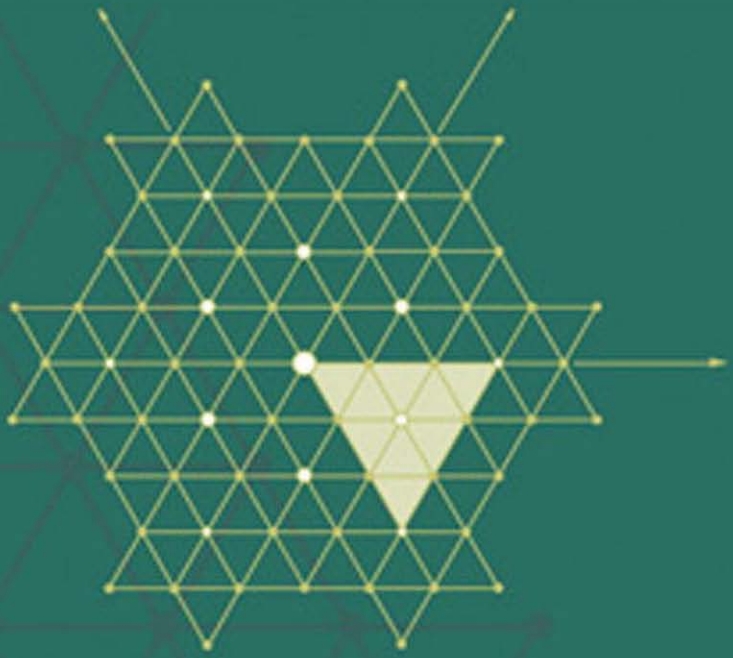
Lars-Erik Andersson · Neil F. Stewart



Introduction to the MATHEMATICS of SUBDIVISION SURFACES

siam

Introduction to the MATHEMATICS of SUBDIVISION SURFACES

Introduction to the MATHEMATICS of SUBDIVISION SURFACES

Lars-Erik Andersson
Linköpings universitet
Linköping, Sweden

Neil F. Stewart
Université de Montréal
Montréal, Canada

siam®
Society for Industrial and Applied Mathematics
Philadelphia

10 9 8 7 6 5 4 3 2 1

Library of Congress Cataloging-in-Publication Data

Andersson, Lars-Erik.
Introduction to the mathematics of subdivision surfaces / Lars-Erik Andersson, Neil F. Stewart.
p. cm.
Includes bibliographical references and index.
ISBN 978-0-898716-97-9
1. Subdivision surfaces (Geometry) I. Stewart, N. F. (Neil Frederick), 1943- II. Title.
QA646.A535 2010
516–dc22 2009047055

To Kristina and Krys

Contents

List of Figures

List of Tables

Preface

Subdivision surfaces were introduced in the Computer-Aided Design (CAD) literature in the late 1970s, and they have since attracted much attention in the fields of computer graphics, solid modelling, and computer-aided geometric design. It is the purpose of this book to introduce the essential mathematics underlying these surfaces, at a level that is accessible both to graduate students in computer science and to researchers and practitioners with a similar or stronger mathematical background.

In terms of mathematical content, the book has two main goals. The first is to provide a unified view of the field. The second is to explain the mathematics carefully, but as simply as possible, so that the reader will be able to easily read the literature.

It is easy to get the impression, from a first encounter with the subdivision literature, that the field consists of a miscellaneous collection of smoothing techniques, some inspired by classical B-spline methods, and others that are completely ad hoc. In particular, even when taxonomies of methods are given, the classifications do not seem to lead to sharp distinctions. For example, methods designed for quadrilateral or triangular meshes can nonetheless be applied to other kinds of meshes, including meshes of opposite or mixed type. Similarly, the distinction between primal and dual methods seems slightly obscure, and in fact this distinction also fails to be perfectly sharp: even if we restrict our attention to the most special classes of methods, they may be of mixed primal-dual type.

In fact, however, there is a great deal of unity and structure to the field. The main idea we use to show this, is to arrange all of the standard subdivision methods in a simple hierarchy based on the class of spline surfaces they generate. The most special methods in this hierarchy are those that generate classical tensor-product uniform B-splines, while the most general methods in the hierarchy correspond to generalized splines, i.e., linear combinations of nodal functions which themselves can be obtained by applying an affine-invariant subdivision procedure to the unit-impulse function. A second idea which shows the unified nature of the field is that a step of the basic subdivision method can be viewed, in the B-spline case, as a series of simple averagings done in alternation between the initial refined mesh for the step and the dual of this mesh. If we decide to alternate back and forth an even number of times at each step, then there is no need to actually construct the dual mesh, and we have what is called a *primal* method. On the other hand, if we decide to compute these averages an odd number of times, then the dual mesh must be constructed in

some way, and we have what is called a *dual* method. This alternating-averaging structure is an important thing to notice. A generalized version of alternating averaging occurs for more general classes of subdivision methods, such as box-spline methods, and even $\sqrt{3}$-subdivision, a non-box-spline method, can be viewed as involving a form of alternate averaging.

The organization of the book is discussed in detail in Section 1.1. One significant aspect of the organization is that Chapter 1 jumps ahead and makes statements, about subdivision methods and surfaces, that are only justified later, in the more orderly mathematical presentation which begins in Chapter 2. One of the reasons for this choice of organization is to make the book more useful as a graduate-level textbook in computer science. In such a situation, the student may already have a great deal of informally obtained information about, say, Catmull–Clark and Loop subdivision and may be interested in seeing a description of these methods without having to first read three or four chapters. Also, Chapter 1 contains basic information that may help the student, or general reader, make sense of what is often left unclear in the literature. For example, as is the case for the implementation of solid-modelling systems, it is important when describing subdivision methods to distinguish between a *logical* mesh and a *polyhedral* mesh (this is done carefully in Chapter 1, but not always in the literature). Similarly, Chapter 1 gives descriptions of various kinds of subdivision matrices that are used in the description and analysis of subdivision procedures (many papers in the literature simply refer to "the" subdivision matrix, which is confusing for the novice, since in fact there are many different varieties of subdivision matrix). Chapter 1 also describes splitting schema, dual meshes, and regular and nonregular meshes, and it presents the hierarchical classification described above. In particular, within this hierarchy, the distinction is made between basic and variant methods, where the latter are designed for use in nonregular meshes.

Early drafts of the book have been used as a reference text in a one-month segment of a graduate course in solid modelling, in the computer science department of the Université de Montréal. This segment includes most of Chapter 1, much of Chapter 2, some of Chapter 3, and some brief remarks on convergence, smoothness, and surface evaluation and estimation (Chapters 5 and 6). This experience led to the conclusion that the material is difficult for beginning graduate students in computer science, but quite accessible to mathematically inclined Ph.D. students. Material from Chapter 7 (shape control) could also be included in such a graduate course, and the Notes might also be useful to the student.

The book should probably be read in the order in which it is written, with the exception of the Appendix and the Notes, which should be consulted as needed. Any material that is already familiar can, obviously, be skimmed, but all chapters depend on the basic information in Chapter 1, and Chapters 2, 3, and 4 are progressively more general. All chapters also rely heavily on Chapter 5, on convergence and smoothness, although these topics are postponed until the basic theory of the first four chapters is in place. Chapters 6 and 7 rely on earlier chapters, and in particular, the last section of the main text, on shape control, makes use of the global subdivision matrices of Chapter 1 and the Generalized-spline subdivision methods of Chapter 4.

The mathematical level required to read the book is that of an advanced graduate student in computer science. It is assumed in particular that the reader has taken courses in Linear Algebra and Advanced Calculus. It is also assumed that the reader is generally familiar with B-splines, at the level it would normally be taught in an undergraduate computer graphics course (based, for example, on [147, Ch. 15] with some supplementary material added on bicubic surface patches, or on [53, Ch. 11]). The presentation of B-spline surfaces is narrowly focused on subdivision surfaces: the reader who wants a thorough understanding of B-splines and Non-Uniform Rational B-Splines (NURBS) should read the books of Cohen, Riesenfeld, and Elber [30], Farin [51], and Piegl and Tiller [127]. We note finally, on the topic of the mathematical level of the book, that it increases quite sharply following Chapter 1.

The reader described in the previous paragraph may from time to time be required to learn techniques not previously seen. A good example is generating functions. It is not possible to read the subdivision literature without knowing something of these: they are used by many authors, because they often lead to simpler derivations. On the other hand, a typical computer science program may not include discussion of this topic, and it may be necessary to consult, for example, Knuth's *The Art of Computer Programming* [72, Sec. 1.2]. Similarly, we make use of the complex Fourier transform (although some of the related derivations are relegated to the Appendix) and the discrete Fourier transform. Many graduate students in computer science know of these techniques (perhaps because of a course in signal processing or in computer vision), but again, a typical computer science program may not include discussion of these topics.

The idea of structuring the field as subclasses of generalized splines came from the understanding gained by reading the work of Peters and Reif, and in particular, by reading a draft of [124]. Similarly, the fundamental nature of the primal-dual alternation in B-spline methods is quite evident in the original Lane–Riesenfeld paper [81], and it is brought out very clearly in the references [101, 151, 177]. On the other hand, the formal structuring of the field as we have done it is new, and our use of centered nodal functions aids considerably in bringing out the essential symmetry of subdivision methods. The presentation of box splines in Chapter 3 is, we believe, made quite accessible by developing it in exact parallel with the development for tensor-product B-splines. Similarly, the later development of subdivision polynomials related to generalized splines is also done in parallel with the more special cases just mentioned, which leads to very natural analyses of the corresponding general methods.

Exercises and projects appear in separate sections at the end of each chapter. Course materials, including solutions to the exercises (and results for a few of the projects) are available to professors using the book as a course text; see www.siam.org/books/ot120 for information. The Notes appear at the end of the book. References to theorems, equations, figures, etc. have an appended subscript giving the page number: for example, $(2.33)_{/65}$ refers to equation (2.33), which appears on page 65, and Figure $2.7_{/68}$ refers to Figure 2.7 on page 68. (This idea, as well as the notation $pQ4$, $pT4$, and $dQ4$ used to identify the standard splitting schema, were also adopted from an early draft of [124].) In the chapters following

Chapter 1, the end of a formal proof is indicated by an open box □, and the end of a remark or an example that has been set off from the main text by a filled box ■. Note that even in the first chapter, which is relatively informal, we occasionally adopt a formal style for definitions, but only when it seems necessary for clarity. Finally, certain remarks are annotated with a star, as in the case of Remark* 1.2.4$_{/11}$. Such remarks, although perhaps important, contain details that need not be thoroughly understood on a first reading. Alternatively, a starred remark may simply mention that the material immediately following can be skimmed on a first reading. Occasionally starred remarks refer forward to results not yet proved.

Many people provided useful comments on the manuscript, at various stages, including P. Beaudoin, S. Bouvier Zappa, F. Duranleau, D. Jiang, V. Lazar, V. Nivoliers, V. Ostromoukhov, J. Peters, P. Poulin, I. Stewart, J. Vaucher, Z. Wu, M. Zidani, and an anonymous referee.

François Duranleau and Di Jiang produced most of the more difficult figures, with help from Wu Zhe. Figure 1.5$_{/5}$ was produced by Wu Zhe using Quasi 4-8 subdivision [161], starting with a model obtained from www.blender.org. All three of these people provided considerable help over a long period.

The members of the team at SIAM were unfailingly friendly, helpful, and efficient. In particular, the authors are very grateful to Elizabeth Greenspan, Sara Murphy, Nancy Griscom, and Lisa Briggeman. They made this a better book.

The second author wishes to thank the Natural Sciences and Engineering Research Council of Canada for its support of his research. He also wishes to express his gratitude to Warren and Enid Damer, without whose inspiration this book would not have been written.

Lars-Erik Andersson
Linköping, Sweden

Neil F. Stewart
Montréal, Canada

July 2009

> It is impossible for an expositor not to write too little for some, and too much for others. He can only judge what is necessary by his own experience; and how long soever he may deliberate, will at last explain many lines which the learned will think impossible to be mistaken, and omit many for which the [uninitiated] will want his help. These are censures merely relative, and must be quietly endured. I have endeavoured to be neither superfluously copious, nor scrupulously reserved, and hope that I have made my author's meaning accessible to many who before were frighted from perusing him, and contributed something to the public by diffusing innocent and rational pleasure.
>
> —*Samuel Johnson*

Notation, Conventions, Abbreviations

Points in $\mathbb{R}^N$ are denoted in ordinary type. For example, a spline surface is denoted by the vector-valued function $x(u, v)$ with values lying in $\mathbb{R}^N$, and similarly for a spline curve $x(t)$. When modelling physical space, the dimension N of the space $\mathbb{R}^N$ is often equal to 3. But N may be arbitrary—the control points of a subdivision mesh may correspond to general attributes. We do not distinguish between N-dimensional Euclidean space (an affine space of points) and the real vector space $\mathbb{R}^N$: points in Euclidean space are viewed as vectors starting at the origin. The value of the function x viewed as a vector in $\mathbb{R}^N$, the associated control points, and certain related coefficients such as c_j are written as row vectors. Other vectors are written as column vectors.

The usual meaning of the principal symbols used is as shown in the following list, but it sometimes happens that a variable with the same or similar name is used locally for some other purpose.

A	a matrix $A_{(N\times N)}$, or a matrix $A_{(2\times k)}$ representing a mapping A
c_j	coefficients in eigenvector expansion
C^k, $C^k(\mathbb{R})$, $C^k(\mathbb{R}^2)$	k times continuously differentiable
C^k	the unit cube in $\mathbb{R}^k$
C, $\bar{C}$, C_k, $\bar{C}_k$, c, c_k	constants
$d = m - 1$	(bi-) degree of (tensor-product) B-spline, m the order of the univariate B-spline
$D = \frac{d}{dt}$, D^k, D_y, D_e $(e \in \mathbb{R}^2)$, ∇	derivative operators
$\Delta_e, \Delta, \Delta_k$	difference operators
$\frac{\partial}{\partial u}$	partial differentiation
∂B	boundary of a subset B of $\mathbb{R}^2$
e	number of edges in a face
$e^m = \{e_1, \ldots, e_m\}$	directions defining a box spline
$e^m_{(i)}$	e^m with e_i deleted
$e^m_{(ij)}$	e^m with e_i and e_j deleted
$\bar{e}/2 = \frac{1}{2}\sum_{i=1}^m e_i$	centre of box-spline coefficient grid
E_i, E_i'	control point (Catmull–Clark)

E_f	the set of edges in face f
f	a face in a logical mesh
$f = f(t)$, $f = f(y)$	a function of the variable t or y
F_i, F_i'	control point (Catmull–Clark)
$F_1, \ldots, F_\alpha$	faces in $\mathbb{R}^2$
$\boldsymbol{F}_1, \ldots, \boldsymbol{F}_\alpha$	faces in manifold $\boldsymbol{M}$
$F(y-l)$	a function in $L^1(\mathbb{R}^2)$
$G(z)$, $G_a(z)$, $G_f(z)$, $G_{f_{ih/2}}(z)$	generating functions
G, G_k^*	coefficient grids (support of subdivision polynomial)
$\mathbb{G}_k$	subset of $\mathbb{R}^2$ (defined by k-ring neighbourhood)
$\mathcal{G}_m$, $\mathcal{G}_m^*$	grids defined by e^m
h	resolution of grid, grid-size
i	a general index, or $\sqrt{-1}$
k, l	general indices (often indexing control points)
$\ell \in \mathbb{Z}$	indexing logical vertices
L	the number of control points in a mesh
$L^1(\mathbb{R}^2)$	the Lebesgue integrable functions on $\mathbb{R}^2$
λ_i	eigenvalues of local subdivision matrix
m	(bi-) order of a (tensor-product) B-spline, or total order of a box spline
M, M', M^*, M^ν, M_{odd}, M_{even}	logical mesh
$\mathcal{M} = (M, p)$	polyhedral mesh
N	dimension of $\mathbb{R}^N$
$N_k^m(h;u)$, $N^1(h;t)$, $N^*(he^k;y)$, $N(y)$	nodal functions
n	valence of a logical vertex
$n(y)$	normal vector depending on parameter y
ν	subdivision iteration index
p_{kl}, p_ℓ, $p^\nu \in \mathbb{R}^N$	control points (row vectors)
$p_{(L\times N)}$	matrix with L rows of control points
$p_{(\omega\times 1)}$	scalar control points (case of an infinite grid)
$p(z)$, $p(h;z)$, $q(z)$	generalized polynomials corresponding to sets of control points
$q_k = \sum_l s_{k-2l} p_l$	control points after subdivision
$\mathbb{R}^N$	real vector space of dimension N
R, Q, S	control points (Catmull–Clark)
S	local subdivision matrix
Σ, Σ^ν	global subdivision matrices
$S(y)$, $\hat{S}(\omega)$	functions used in Fourier analysis
$s(z)$	subdivision polynomial
$(0,1)^t$, S^t	transpose of a matrix
$t_{(1\times N)}$	translation of control sequence

t	independent variable in univariate case: $x = x(t)$
$(u, v)^t$	independent variable in bivariate case: $x = x(y)$, $y = (u, v)^t$
V, V'	control points (Catmull–Clark)
$w \in \mathbb{R}^k$	vector (in the context of box splines)
$w = e^{2\pi i/n}$	nth root of unity
w, w^*, ω	standard parameters in Butterfly, Kobbelt, and Loop methods
ω	cardinality of the natural numbers
ω	variable of Fourier transform
$x = x_{(1\times N)} = x(u, v)$	spline surface
$x = x_{(1\times N)} = x(t)$	spline curve
$y = (u, v)^t$	independent variable in bivariate case
$\mathbb{Z}$	the integers (bi-infinite grid)
$\mathbb{Z}^2 = \mathbb{Z} \times \mathbb{Z} \subset \mathbb{R}^2$	two-dimensional vectors of integers
$\mathbb{Z}_L = \{0, \ldots, L-1\}$	the integers modulo L
z	variable in generating function, translation operator, argument of subdivision polynomial $s(z)$
ζ^ν, κ^ν	control sequences
ξ, η	right and left eigenvectors of local subdivision matrix
ω	see w above
Ω, $\bar{\Omega}$	open subset of $\mathbb{R}^2$ and its closure
$\lvert \cdot \rvert$	Euclidean norm of vector in $\mathbb{R}^2$ or $\mathbb{R}^N$
$\lvert k \rvert = \lvert k_1 \rvert + \lvert k_2 \rvert$	1-norm of $k = (k_1, k_2) \in \mathbb{Z}^2$
$\lfloor a \rfloor$	greatest integer less than or equal to a
$\lceil a \rceil$	smallest integer greater than or equal to a
$\lg(\cdot)$, $\ln(\cdot)$	logarithm base 2, base e
$\eta^* \xi$	η^* denotes transposition and complex conjugation of the complex vector η
$\bar{\eta}_l$	complex conjugate of the component η_l
$\Re$, $\Im$	real and imaginary parts
A^0	interior of the set A
$\mathrm{conv}(\cdot)$	convex hull of a set of points
$\det(\cdot)$	determinant of a matrix
vol_{k-2}	Lebesgue measure in $\mathbb{R}^{k-2}$
$\mathrm{supp}(\cdot)$	the support of a function
$\sim$	equivalence, or asymptotic equality
$:=$	value assignment
$\doteq$	defined to be equal
$\times$	vector cross product
$\otimes$	convolution
$\hat{N}$, $(y^{k-r}F(y))\hat{}$	Fourier transform of N, $y^{k-r}F(y)$

Conventions

- If $z = (z_1, z_2)$, $a = (a_1, a_2)$, then $z^a = z_1^{a_1} z_2^{a_2}$.
- If $p(z) = p(z_1, z_2)$ is a polynomial in two variables, then $p(z^2) = p(z_1^2, z_2^2)$ and $p(z^{1/2^\nu}) = p(z_1^{1/2^\nu}, z_2^{1/2^\nu})$.
- If $p(z) = \sum_a p_a z^a$, $f = f(t)$, then $p(z)f = \sum_a p_a (z^a f)$.
- Let $j = (j_1, j_2)$, $d = (d_1, d_2)$, $z = (z_1, z_2)$, $k = (k_1, k_2)$. Then
  - $0 \le j \le d$ means $0 \le j_1 \le d_1$, $0 \le j_2 \le d_2$;
  - $\partial^j = \partial_1^{j_1} \partial_2^{j_2}$ (partial differentiation);
  - $p^{(k)}(z) = p^{(k_1,k_2)}(z_1, z_2) = \partial_1^{k_1} \partial_2^{k_2} p(z_1, z_2)$.
- The notation $\Pi_d \ni y^k \mapsto y^k + \sum_{0 \le r < k} c_{k,r} y^r \in \Pi_d$ means that each y^k in Π_d is mapped onto the element shown to the right of the symbol $\mapsto$, and this element is also in Π_d.
- The notation $\pi_f : M \supset F \to F \subset \mathbb{R}_f^2$ means that $M \supset F$, $F \subset \mathbb{R}_f^2$, and $\pi_f : F \to F$.

Abbreviations

$dQ4$: dual quadrilateral 4-split
$pQ4$: primal quadrilateral 4-split
$pT4$: primal triangular 4-split
$LR(d)$: the Lane–Riesenfeld algorithm of degree d
$LR(d \times d)$: the Lane–Riesenfeld algorithm of bidegree d
LSS: Linear Subdivision plus Smoothing algorithm
$4pt \times 4pt$: tensor product of the four-point method with itself

Chapter 1
Introduction

Divide et impera.
—Latin maxim

Söndra och härska.
—Swedish translation

A sure axiome, Divide and rule.
—J. Hall, 1605

1.1 A brief overview

Subdivision surfaces were introduced in the field of solid and surface modelling in 1978, with the publication of the papers by Catmull and Clark [24] and by Doo and Sabin [45]. They are now widely used in many application areas, including computer graphics, solid modelling, computer game software, film animation, and others, as an alternative to B-splines and NURBS (Non-Uniform Rational B-Splines) [30, 51, 127].

To illustrate the basic idea of subdivision methods, we give a brief description of Chaikin's algorithm. The algorithm applies to curves, rather than to surfaces, but it is a simple example that illustrates clearly the idea of subdivision. Suppose that we start with a polygonal simple closed curve, made up of four segments, as illustrated by the large square in Figure 1.1$_{/2}$ (left). In the first step of the subdivision process, two intermediate points are introduced on each segment by taking a weighted average of the corner points, using weights $\left(\frac{1}{4}, \frac{3}{4}\right)$ and $\left(\frac{3}{4}, \frac{1}{4}\right)$, respectively. This produces the octagonal polygon (eight segments and eight corners) shown in Figure 1.1$_{/2}$ (middle).

The same procedure is then repeated, using the octagonal polygon as a starting point. In the second subdivision step, two intermediate points are introduced into each of the eight segments, again using the weights $\left(\frac{1}{4}, \frac{3}{4}\right)$ and $\left(\frac{3}{4}, \frac{1}{4}\right)$ for each segment. This produces a polygon with 16 segments, as shown in Figure 1.1$_{/2}$ (right).

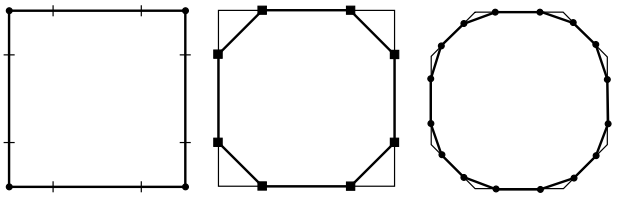


Figure 1.1. *Two steps of Chaikin's algorithm.*

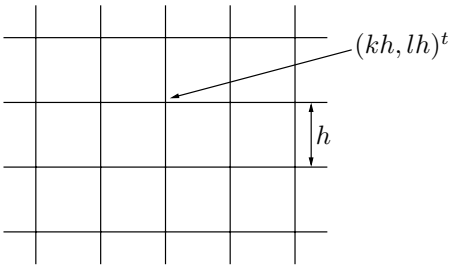


Figure 1.2. *Quadrilateral grid.*

It is intuitively clear that if this process is repeated indefinitely, the polygonal corners computed at each step will converge towards a smooth limit curve.

Chaikin's method will be discussed in more detail later. The basic idea illustrated here generalizes in several different ways to the bivariate case (surfaces), and this is the subject of the book.

In order to discuss the relation between subdivision surfaces and the rest of the field of surface modelling, we compare them to B-spline surfaces. It is assumed that the reader has some familiarity with the latter type of surfaces, but we give here a brief description of a special case, the tensor-product uniform B-splines. These are discussed in much more detail in Section 1.3.1 and Chapter 2, where the notation is carefully defined.

Suppose that a grid has been defined on $\mathbb{R}^2$, with separation h between the grid lines (see Figure 1.2$_{/2}$). We refer to the separation h as the *grid-size* or *resolution of the grid*. A tensor-product uniform B-spline is a parametric surface of the form

$$x(u, v) = \sum_{k \in \mathbb{Z}} \sum_{l \in \mathbb{Z}} p_{k,l} N_k^m(h; u) N_l^m(h; v). \tag{1.1}$$

It is assumed that there is associated with each grid point $(kh, lh)^t$ a control point $p_{k,l}$ lying, say, in $\mathbb{R}^3$. Each of the functions $N_k^m(h; u)$ and $N_l^m(h; v)$ is a function of a single variable u or v, and each has a shape like the profile of a bell. Thus the product $N_k^m(h; u) N_l^m(h; v)$ defines a bell-shaped function, illustrated for $m = 4$ in Figure 1.3$_{/3}$, that determines the influence of the control point $p_{k,l}$ on the final surface $x = x(u, v)$. Such functions are called *basis functions*[1] or *nodal functions*.

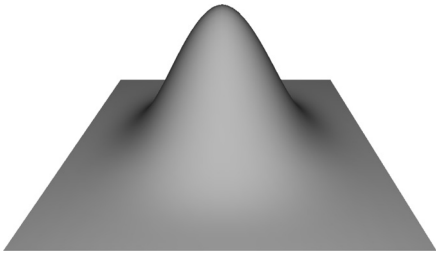


Figure 1.3. *B-spline basis function of order* 4.

Both N_k^m and N_l^m are piecewise polynomials of degree $m-1$, and they are centered around kh and lh, respectively, so that $N_k^m(h;u)N_l^m(h;v)$ is centered around $(kh, lh)^t$ in the parametric domain.

A bounded portion of a tensor-product uniform B-spline surface $x(u,v)$ is illustrated in Figure $1.4_{/4}$, along with a typical control point. The control points $p_{k,l}$ do not usually lie on the surface, but they influence its position: the shape of the surface can be modified by moving the control points. Control points are usually denoted by the letter p, with indexing that varies depending on the context. (An exception occurs in Section 1.3.1 and following, where a different notation for control points is used to describe the Catmull–Clark method.)

If the control points $p_{k,l}$ and their surrounding faces are linked together logically in the way specified by the planar grid, then they form an infinite *polyhedral mesh* in $\mathbb{R}^N$ (in the example above we had $N = 3$). Some remarkable theorems and algorithms have been developed for B-splines which permit their evaluation by means of repeatedly subdividing this mesh. Several of the standard subdivision-surface methods can be viewed as generalizations of these algorithms to the case of meshes (usually finite) having a more general topology. These subdivision methods are applied directly in the polyhedral mesh, but in those parts of the mesh that correspond to a subset of the plane tesselated with a simple quadrilateral grid, the surfaces generated correspond locally to ordinary tensor-product B-splines. This is an important fact.

In fact, subdivision algorithms for B-splines have been developed for cases much more general than the tensor-product uniform case of $(1.1)_{/2}$, namely the cases of nonuniform grids [51] and rational B-splines (NURBS) [127]. (These algorithms include "knot-insertion algorithms" and the well-known "Oslo algorithms" [30, 51, 127].) We do not need to consider these more general B-spline cases, but we do need to use another generalization, in a different direction, of tensor-product uniform B-splines, namely the generalization to box splines.

One of the main advantages of subdivision surfaces, as compared to B-spline and NURBS surfaces, is that the latter must be trimmed and pieced together in order to produce surfaces of general form, since otherwise we would be limited to the bi-infinite sheet or, by identification of points, the cylinder and torus. ("Trimming" simply means that a subset of the domain of the B-spline or NURBS surface is delineated as the part of the surface to be used.) This process of trimming and piecing

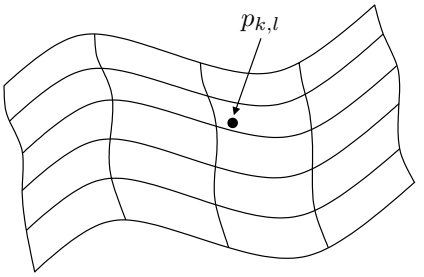


Figure 1.4. *Bounded portion of a B-spline surface.*

together leads to difficult problems related to guaranteeing continuity and consistency [5]. In contrast, subdivision surfaces are intrinsically capable of assuming general form.

Other advantages of subdivision surfaces include scalability and good compatibility with application areas requiring meshes, and in particular, finite-element meshes [6]. On the other hand, combining these advantages with tools comparable to those available in the trimmed-NURBS context [42], and especially, to obtain the advantages of interactive editing [175], is not always simple, and this is one reason why the subdivision-surface literature has grown so quickly over the last two decades.

There are many different ways to categorize subdivision-surface schemes: for example, classification parameters used in [44] include the size of the footprint of the rules defining new points and whether the predominant element in the subdivided mesh should be a triangle or a quadrilateral. (Data defining geometric models are frequently in the form of a triangular mesh, but quadrilaterals are often better than triangles for representing the symmetries of natural and artificial objects, including objects such as arms, legs, and fingers [42].) Other classification criteria that may be useful include whether the methods interpolate the given data or only approximate it, the level and nature of the continuity of the limit surfaces, and whether the control points produced at each step are associated with a refined version of the starting mesh (primal method) or with its dual (dual method). In this book, however, all of the above characteristics are viewed as secondary classification criteria since, as will be seen below, they do not always lead to clear and precise distinctions. The main classification used here is an unambiguous hierarchical classification, based on the mathematical nature of the surface generated.

An example of subdivision in $\mathbb{R}^3$ is shown in Figure $1.5_{/5}$. In this model, the elephant's tusk merges with the trunk.

Historical background

As mentioned above, subdivision surfaces can be viewed as a generalization of B-spline surfaces of the form $(1.1)_{/2}$, and the histories of the two subjects therefore have much in common.

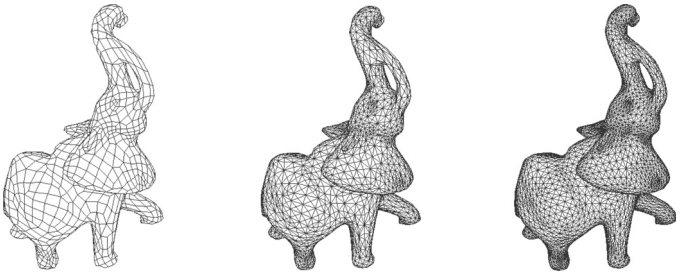


Figure 1.5. *Various stages of Quasi* 4-8 *subdivision.*

A brief history of the beginnings of B-splines is given by Farin [51, Ch. 10], where the earliest reference is to the nineteenth-century work of Lobachevsky, who constructed splines using convolution. A very important modern paper was published in 1946 by Schoenberg [143]. This paper can be viewed as the origin of two distinct and major fields involving the use of splines: geometrical modelling [30, 51, 127] and statistical data smoothing [165].

Subdivision algorithms for curves and surfaces in Bernstein form (Bézier curves and surfaces) were developed starting around 1960 by de Casteljau and Bézier. The recurrence relations for B-splines were discovered by de Boor, Cox, and Mansfield in 1972, and one of the most important algorithms for B-splines, from our point of view, was published by Lane and Riesenfeld in 1980 [81]. This algorithm is studied carefully in Chapter 2. The development of box splines was another advance that is very important for the theory of subdivision surfaces: the standard references are [38, 129], both of which contain bibliographies.

The history of subdivision itself is at least as old as the self-similar subdivision of certain tiles into smaller ones of the same form, which first appeared in medieval architecture beginning in the thirteenth century [93]. It is shown below that current subdivision algorithms are closely related to self-similar subdivision of certain tilings of the plane.

The first modern references to subdivision algorithms are the 1947 and 1956 papers of de Rham [39, 40]. Chaikin's algorithm [26], a special case of a method described in [40], was published in 1974. Subdivision surfaces were introduced into modern solid modelling with the publication in 1978 of papers by Catmull and Clark [24], and Doo and Sabin [45], followed by Loop's paper [91] in 1987. Finally, a recent historical fact is noted in [176, p. 20]: the strong interest in subdivision surfaces that appeared in the 1990s arose out of a desire to circumvent the problem of topology limitations involved with the use of ordinary B-spline surfaces.

Choice of method, including data structures and implementation

We can describe in very general terms some typical criteria that might be used to guide our choice of method. Some of these criteria, such as mesh type, the level and nature of continuity, and whether the surface interpolates given data, have

already been mentioned, and these might be grouped generically under the heading "quality and nature of the generated surface." In addition, there is the cost of evaluating the surface, which might be measured by computation time, by memory utilisation, or by a combination of the two. Yet another criterion is the ability to associate attributes, such as colour and texture, to the faces of the surface in a persistent way.

These criteria will be discussed further, but first we mention, again in very general terms, what is involved in the choice of method. First of all, there is the choice of basic subdivision method (Catmull–Clark, Loop, etc.). Second, there is the choice of data structure: how should the subdivision mesh be represented on the computer? Third, there are choices to be made in the implementation of specific tasks: for example, should the surface be evaluated by repeated subdivision, or rather by an auxiliary method that permits evaluation for specific parameter values?

It is clear, even from these short lists of criteria and choices of method, that there are many complicated trade-offs here, but we do not discuss these. We do of course mention advantages and disadvantages of particular methods from the point of view of implementation, but resolving the various trade-offs involved is beyond the scope of this book. Our goal is limited to providing an understanding of the underlying mathematical structure.

In fact, the situation is much more complicated than the discussion above might suggest. Let us look first at the *criteria* used. If we consider the criterion of cost of evaluating the surface, there is often a time-memory trade-off. In addition, however, we must distinguish between absolute cost and incremental cost: it is sometimes possible to precompute certain quantities once and for all (incurring, usually, an increased cost of memory utilisation), in such a way that the subsequent time cost per evaluation is low. Further, there are other aspects of time and memory costs that are as important as their overall magnitude, such as whether they grow gradually as a function of required precision, or, rather, in sudden jumps. In fact, this is a central issue for subdivision-surface methods.

In addition to all this, related but different criteria, such as the cost of, say, picking a specified point on a subdivision surface, may also be relevant.

The quality of the generated surface is also not an unambiguous criterion. For example, there are many different kinds and levels of continuity that may be specified, as well as certain "fairness" measures that are sometimes very subjective (does the method produce surfaces that are too "pointy," or too "chunky?").

Turning now to the *choice of method*, the situation is again quite complicated. For example, the choice may be heavily influenced by a previous choice of mesh representation in a larger system for which subdivision surfaces are only a part. It is mentioned below that in such circumstances, some methods (the *primal* methods) are more convenient than others (the *dual* methods) when it comes to associating attributes such as colour and texture, in a persistent way, to objects defined by subdivision surfaces. There exist convenient data structures for dual methods, but converting from one representation to another is expensive. Similarly, data structures capable of representing both primal and dual methods tend to be quite elaborate in comparison with the streamlined implementations possible

for a method like Catmull–Clark, which requires only the primal mesh and can be implemented "in-place."

Similar remarks hold for the actual implementation of specific algorithms: different memory-management strategies are possible, and the resolution of the trade-offs involved may depend on whether the implementation is to be on a low-end or high-end system, since these may have different numbers of processors and different memory-cache designs [18]. Resolving the trade-offs is especially difficult in an environment of rapidly evolving graphics cards which permit significant levels of on-card processing.

The complexity does not end with this, however, since partial solutions to the problems mentioned above may introduce new difficulties. For example, in order to reduce both time and memory requirements, it is of interest to use adaptive subdivision (carrying the subdivision to different levels in different parts of the object). This introduces new criteria, namely whether it is possible to vary the level of subdivision in a convenient way, and without visible artifacts in the rendered object.

A good discussion of general approaches to the implementation of important operations in the subdivision-surface context, including the operations of surface display, finding plane sections, and surface-surface intersection, is given in [139].

Plan of the book

Many subdivision surfaces can be viewed as generalizations of B-spline surfaces in the sense that, instead of being limited to surfaces defined in terms of a planar parametric domain, we can represent directly surfaces of more general form, such as an elastic deformation of a sphere, or of a torus (the surface of a doughnut) with several holes. Other subdivision surfaces are generalizations of box splines, which themselves are generalizations (in a different sense) of B-splines. A natural order of mathematical presentation is, therefore, to begin with B-spline surfaces, and later to proceed to the more general cases. This order is the one used in the book, starting in Chapter 2. First, however, we jump ahead a little in the present chapter. The goal in doing this is to give the reader an overview of subdivision methods and the corresponding surfaces they generate. In Section 1.2 we introduce polyhedral meshes and primal and dual subdivision processes, and the idea of a stencil (sometimes referred to in the literature as a subdivision mask). Then, in Section 1.3, we describe several subdivision methods, including those of Catmull–Clark, Doo–Sabin, and Loop, and then we give a classification (or taxonomy) of subdivision methods that shows the mathematical structure of the field. Section 1.4 gives a summary of the relationships among the various kinds of subdivision matrices that appear in the literature. In Section 1.5 we give an example illustrating that a minor modification of a standard method may produce fractal curves. The material in Chapter 1 is presented with little or no mathematical justification.

In Chapter 2, we return to the beginning, introducing those parts of the B-spline theory that are necessary for the later development of subdivision methods. In particular, we discuss generating functions, discrete and continuous convolution,

subdivision polynomials, the Lane–Riesenfeld algorithm, and several important principles related to subdivision. In Chapter 3 this theory is generalized to the box-spline case, and corresponding box-spline methods are described: Loop, Midedge, and 4–8 subdivision. Then, in Chapter 4, we generalize even further, using general subdivision polynomials; this leads to a class of methods wide enough to include all of the commonly used subdivision algorithms, including $\sqrt{3}$-subdivision, the Modified Butterfly method, and the Kobbelt method. We also define a sort of universal set: the class of generalized-spline surfaces, which correspond to the *Generalized-spline subdivision methods*.

Convergence and smoothness are discussed in Chapter 5. Fairly complete results on convergence are given for the box-spline case, and we demonstrate what can be proved, by elementary means, for methods defined by more general subdivision polynomials. The chapter also gives an introduction to the topic of convergence and smoothness in the nonregular case and points the reader to major references, such as [124, 172], which give much more extensive analyses. In Chapter 6, evaluation and estimation of surfaces are discussed. Surface *evaluation* includes evaluation of a subdivision surface as a function of a parameter, and the use of stencils to compute the limiting positions and tangents corresponding to the control points at any subdivision level. Surface *estimation* refers to the calculation of tight bounding envelopes for surface patches, and adaptive subdivision is mentioned as an application. Precision sets and degree of polynomial reproduction are also examined. In Chapter 7 we discuss the question of shape control. This includes the treatment of boundaries and crease edges, interpolation and surface fitting, and methods for multiresolution editing, one of which makes an interesting connection with the theory of wavelets.

Notes on terminology

The classification given in Section 1.3 is intended to describe relationships among *subdivision methods*. The terms *subdivision process* and *subdivision scheme* will be used as synonyms for "subdivision method." The subdivision surface generated by a given subdivision method is of course not the same thing as the method itself, but subdivision methods can be classified according to the type of surfaces they generate. This is the classification approach used here.

Many authors remark informally that some specific subdivision method is a variant of a certain basic subdivision method defined for regular meshes (meshes that correspond to simple tilings of the plane, such as triangular and rectangular tilings). In this book we elevate the terms *basic method* and *variant method* almost to the level of definitions; i.e., we use the adjectives "basic" and "variant" systematically in this context. The basic method applies on regular parts of the mesh, while the variant method is designed to deal with the nonregular case.

The word *grid* is also used systematically to refer to a decomposition of the parametric domain, as in Figure 1.2/2. The corresponding linked collection of control points $p_{k,l}$ in $\mathbb{R}^N$ is referred to as a *polyhedral mesh*, which, it should be noted, is not the same thing as a *logical mesh*. The definitions of logical mesh and polyhedral mesh are given in the next section.

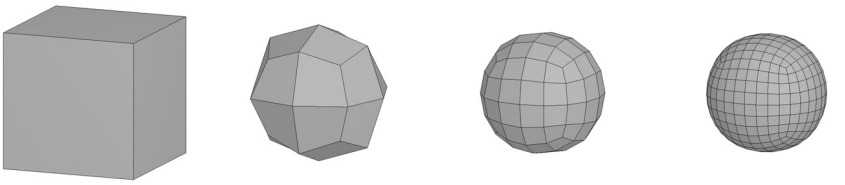


Figure 1.6. *A simple example of Catmull–Clark subdivision.*

1.2 Underlying combinatorial structure

The subdivision rules applied at each step will have the effect, as described below, of producing a *refined mesh*; see Figure 1.6$_{/9}$. (A subdivision step will occasionally be referred to as a *round* of subdivision.) We primarily discuss stationary schemes, which means that the same rules are used repeatedly at each step. The number of steps is arbitrary: the intuitive idea is that after a sufficiently large number of steps, the collection of control points will converge to a smooth limit surface. Note, however, that although convergence proofs make reference to "a sufficiently large number of steps," in practice the number of steps will usually be very small. One reason for this is that memory requirements, and the time required to display the object on the computer screen, increase very rapidly at each step.[2]

The reader might naturally ask where the subdivision rules come from, whether the process converges, and if so, whether it converges to a surface that really is in some sense smooth. Further, if the answers to these questions are satisfactory for the standard methods presented here, one might ask whether the same will be true for some new set of rules that might be proposed. Answers to these and related questions make up the remainder of the book.

1.2.1 Polyhedral meshes

A subdivision method begins with a polyhedral mesh, including the associated control points, and performs a certain number of steps of the subdivision process.

In solid-modelling systems [1, 154], the model is usually divided into two parts. First there is the *logical* or *topological* information, which defines how the various parts (vertices, edges, faces) of an object fit together. Second, there is the *geometric* information, which defines where the various parts are actually situated in Euclidean space. A similar division is appropriate here. We first define a logical mesh and set out the conditions that it must satisfy. This is important for implementation: it determines the conditions that must be satisfied by the data structures used. We then attach geometric information to the logical mesh, to obtain what is called a polyhedral mesh.

The polyhedral meshes used in practice usually correspond to simple well-formed structures[3] in $\mathbb{R}^3$, and we describe below exactly which kinds of well-formed structures are possible. On the other hand, we do not require here that the geometric data should necessarily correspond to physical position in $\mathbb{R}^3$.

A *logical mesh* M is defined by its *vertices* $\ell \in \mathbb{Z}_L = \{0, \ldots, L-1\}$, its *edges* $\{\ell, \ell'\}$, $\ell, \ell' \in \mathbb{Z}_L$, its *faces*, and the topological relationships among all these. A *polyhedral mesh* is denoted $\mathcal{M} = (M, p)$, where M is a logical mesh, and $p_{(L\times N)}$ is a vector of vectors, i.e., a matrix whose rows contain the control points p_ℓ (elements of $\mathbb{R}^N$) associated with the logical vertices ℓ of M. A simple example is a polyhedral mesh corresponding to a sphere, as illustrated in Figure 1.6$_{/9}$. When the word "mesh" is used alone, it will be clear from the context whether "logical" or "polyhedral" is intended.

Before giving the formal definitions, we note that it is convenient to use the single index ℓ, running over the finite set $\mathbb{Z}_L$, to denote a vertex, when discussing the structure of a logical mesh. In other situations, however, the logical vertices are more naturally indexed in other ways, for example, by pairs of integers (k, l) denoting a grid point in the plane, as in $(1.1)_{/2}$. Alternate indexing of this kind is used frequently in the book, especially where infinite logical meshes are used for theoretical analysis, prior to discussion of methods for finite meshes such as those illustrated in Figure 1.6$_{/9}$.

Logical meshes

In the following definitions, we adopt a formal style of presentation, as we will occasionally do even in this first chapter when it seems necessary to make the concepts clear.

Definition 1.2.1. (Face.) *A face f is a finite ordered set of distinct vertices*

$$f = (\ell_0, \ell_1, \ldots, \ell_{e-1}), \quad \ell_i \in \mathbb{Z}_L, i = 0, \ldots, e-1.$$

The face f is also identified with the same set in reverse order, and with sets obtained by cyclic permutation of the elements of either of these.

Definition 1.2.2. (Edge set.) *The edge set of face $f = (\ell_0, \ell_1, \ldots, \ell_{e-1})$ is the collection of unordered pairs*

$$E_f = \{\{\ell_0, \ell_1\}, \{\ell_1, \ell_2\}, \ldots, \{\ell_{e-1}, \ell_0\}\}.$$

The number of edges e may vary depending on the face; we assume $e \geq 3$. Also, to simplify the discussion, we do not permit multiple edges between identical pairs of vertices.

Definition 1.2.3. (Logical mesh.) *A logical mesh is a finite collection of faces and corresponding edge sets.*

As described below, we often deal with logical meshes composed primarily of triangular faces ($e = 3$) or quadrilateral faces ($e = 4$). There is an abuse of terminology here, since there is no geometric information specified, and therefore no suggestion that the "quadrilateral" face is a planar polygon.

An example of a logical mesh is given in Figure 1.7$_{/11}$ (left); it is defined by the six quadrilateral faces $(1, 2, 6, 5)$, $(2, 3, 7, 6)$, $(4, 5, 6, 7)$, $(4, 7, 3, 0)$, $(0, 1, 5, 4)$, and

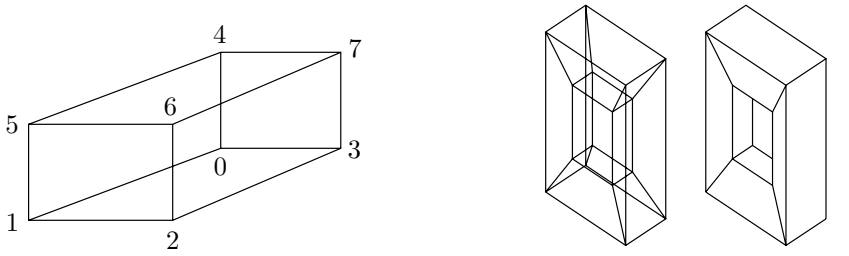


Figure 1.7. *Two examples of logical meshes.*

$(3, 2, 1, 0)$. Another example [98] is given in Figure $1.7_{/11}$ (right). The faces of the two meshes are not indicated in the figure, but these faces (and their edge sets, which implicitly specify the way in which the faces are linked to the vertices and edges) must be specified. For example, in Figure $1.7_{/11}$ (right) the direction of the through hole is ambiguous: it could join the sides as shown in the figure, or it could join the opposite pair of sides; alternatively, the hole could go from top to bottom, or even be absent.

For subdivision, we are interested in logical meshes that satisfy the further condition of *local planarity* [124], [176, Sec. 5.1.1]. Such meshes may be meshes with or without boundary. To illustrate the ideas intuitively, we give some informal examples. If the mesh in Figure $1.7_{/11}$ (left) has the six faces mentioned just above, then it is a locally planar mesh without boundary; the same is true for Figure $1.7_{/11}$ (right) if, for example, the faces are defined so as to create a through hole in some direction. A 4×5 logical mesh, with the form of Figure $1.4_{/4}$ but without control points, would be a locally planar logical mesh with boundary. On the other hand, if two copies of the mesh in Figure $1.7_{/11}$ (left) were combined by identifying the edge $\{4, 7\}$ in one copy with the edge $\{1, 2\}$ in the other copy, then the resulting mesh would not be locally planar.

Remark* 1.2.4. The ideas of the last paragraph are made precise in the two definitions that follow. These definitions can be skimmed on a first reading. ∎

Definition 1.2.5. (Interior edge/vertex, exterior or boundary edge/vertex, boundary, mesh without boundary.) *An edge $\{\ell, \ell'\}$ in the logical mesh is called interior if it belongs to at least two faces; otherwise it is an exterior edge, or boundary edge. The boundary of the logical mesh is the set of all boundary edges. If it is empty, we say that we have a mesh without boundary. An interior vertex is a vertex that is not in a boundary edge; otherwise it is an exterior vertex, or boundary vertex.*

Definition 1.2.6. (Locally planar.) *A logical mesh is called locally planar if each edge belongs to at most two faces and if, for any vertex ℓ, the j faces ϕ_i incident at ℓ can be ordered in such a way that ϕ_i meets ϕ_{i+1} at an edge containing ℓ for $i = 0, 1, \ldots, j-2$.*

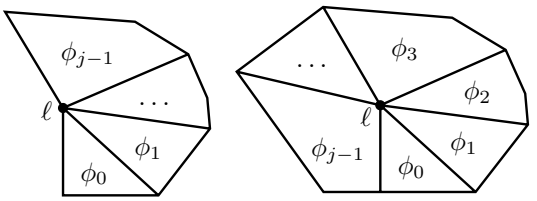


Figure 1.8. *Local planarity: The two possible cases.*

Note that "locally planar" does not mean geometric planarity: no geometric information is present.

In a locally planar mesh, it may happen that ϕ_{j-1} meets ϕ_0 along an edge. If this occurs, then all of the incident edges are interior; otherwise, two of them are exterior. The two cases are illustrated in Figure 1.8$_{/12}$. A locally planar mesh may contain holes (see Exercise 1$_{/47}$).

Another example of a mesh that is not locally planar is a mesh composed of two opposing tetrahedra, joined at a common vertex. Similarly, a non-locally-planar mesh might contain two triangular faces meeting in a single vertex, in the form of a bow tie or butterfly, or several such faces. Other examples are suggested in Exercise 2$_{/48}$.

It is often stated in the subdivision literature that subdivision permits surfaces of "arbitrary topology." This is usually meant to imply that any *locally planar* mesh is permitted, and the topology is therefore not completely arbitrary.[4] We do not often use the hypothesis of local planarity explicitly, but we assume throughout that meshes are locally planar. This is true in particular for the definition of the dual mesh, which is given now.

The concept of the dual of a locally planar mesh M is central to the understanding of subdivision methods. We give the definition[5] first for a locally planar mesh without boundary and then extend the definition to the case of a locally planar mesh with boundary. Let M be a locally planar mesh without boundary. The vertices of the dual of M consist of one point associated with each face of M, as illustrated in Figure 1.9$_{/13}$ for the case of the cube-shaped mesh in Figure 1.7$_{/11}$ (left). The mesh is shown folded into the plane in Figure 1.9$_{/13}$, with the vertices numbered exactly as in Figure 1.7$_{/11}$. In Figure 1.9$_{/13}$, and throughout the book, vertices of the dual mesh are shown as black squares. There is an edge connecting two vertices in the dual of M if the corresponding faces in M are separated by an edge; the edges in the dual are shown in the figure by dashed lines. Finally, there is a face in the dual of M for each vertex in M. The faces of the dual are indicated in Figure 1.9$_{/13}$ by the regions between the dashed lines: in the example shown there are eight, and the original vertex numbers serve to label them.

Now, to define the dual mesh of a locally planar mesh M with boundary, we observe first that such a mesh can be extended to a locally planar mesh without boundary (let us call the extended mesh M^*) by filling in a finite number of holes. To see this, consider any boundary edge in M (see Figure 1.8$_{/12}$, left). This edge

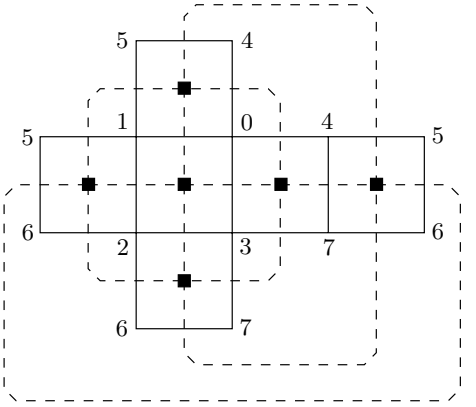


Figure 1.9. *The dual of the cube-shaped locally planar mesh.*

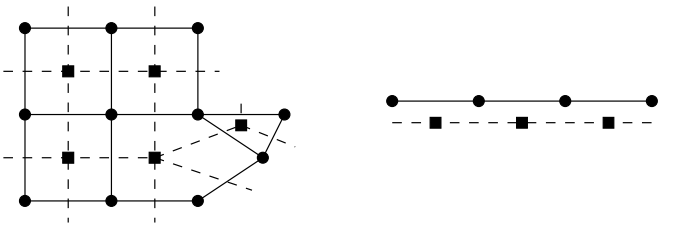


Figure 1.10. *Dual mesh (bivariate and univariate cases).*

must be connected to another boundary edge, which must in turn be connected to another boundary edge. The chain resulting from a continuation of this procedure is closed and simple (no repeated vertices) and has length at least 3, since we have excluded multiple edges between the same pair of vertices. Its length must be finite since M is finite, so that we can add a face to the mesh corresponding to the edges in this chain. Repeating this process, which we have informally described as "filling in a hole in the mesh," until all boundary edges have been exhausted, we obtain a locally planar mesh without boundary. Having denoted this mesh without boundary by M^*, the dual of the original mesh is now defined to be the dual of M^*.

An example corresponding to the case usually illustrated in the literature is shown in Figure $1.10_{/13}$ (left). Here there is only one boundary chain, around the outside of the mesh, and the mesh can be extended to a mesh without boundary by adding a single face corresponding to the exterior region of the figure. The extended mesh M^* in this example has the topological form of a sphere.

As is customary, in Figure $1.10_{/13}$ (left) we have not shown the dual vertex corresponding to the additional face (the external region). The device of adding the additional face, however, must not be omitted, in order to avoid shrinkage of the mesh each time we take the dual of the mesh.

For a locally planar mesh without boundary, the dual of the dual is the original primal mesh. See Exercise $3_{/48}$. The dual mesh can also be illustrated in the one-dimensional case, as in Figure $1.10_{/13}$ (right).

The relevance of the dual mesh is intuitively clear. If we are given a polyhedral mesh, and we form an average of values associated with pairs of vertices, such as those illustrated by black circles in Figure $1.10_{/13}$ (right), it is natural to associate that average value with a newly introduced vertex, shown as a black square. Similarly, if we average the values associated with the corner vertices of the pentagonal face in Figure $1.10_{/13}$ (left), it is natural to associate the average value with a newly introduced vertex associated with the middle of the face. Further, once we have determined the average values for each of the black squares, we may repeat the process, and the new average values are naturally associated with the vertices of the original primal mesh.

In the next section it is observed that many commonly used subdivision methods are either primal methods or dual methods, and the latter make explicit use of the dual mesh. The theoretical importance of the dual mesh is greater than this fact might suggest, however, since it turns out that the underlying mechanism behind many subdivision methods, whether primal or dual, can be viewed as an alternation at each step between a refinement of the initial mesh for the step, and the dual of this mesh (even if it is unnecessary to actually construct the dual mesh in the case of primal methods). This is most easily observed in the Lane–Riesenfeld algorithm, which is the basis for the Doo–Sabin and Catmull–Clark methods, discussed later in this chapter. Furthermore, a generalized version of this alternation mechanism will appear in the discussion of box splines (Chapter 3), and other methods, such as $\sqrt{3}$-subdivision (Chapter 4), can also be viewed as involving alternate averaging.

Polyhedral meshes

We turn now to the control points, introduced above, which form part of the definition of a polyhedral mesh. One (but not the only) possible use of these points is to associate physical position in $\mathbb{R}^3$ with a logical mesh.

Associated with each vertex ℓ is the *control point* $p_\ell \in \mathbb{R}^N$ (see Figure $1.11_{/14}$). These points in $\mathbb{R}^N$ are written as row vectors, and collected together in an $L \times N$

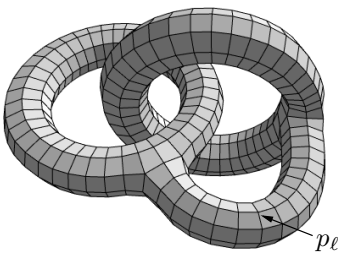


Figure 1.11. *Doughnut with two holes, or coffee cup with intertwined handles.*

matrix, or vector of vectors, denoted p:

$$p = \begin{bmatrix} p_0 \\ \vdots \\ p_{L-1} \end{bmatrix}_{(L \times N)} .$$

Definition 1.2.7. (Polyhedral mesh.) *If M is a logical mesh and p is the associated matrix of control points, the pair $\mathcal{M} = (M, p)$ is called a polyhedral mesh. A polyhedral mesh is called a mesh without boundary if M is a mesh without boundary, and similarly, it is called locally planar if M is locally planar.*

The intuitive idea behind the name "control point" comes from the fact that control points may represent the physical positions of the corners of the mesh: in choosing the initial control points, we have some control over the shape of the limit surface that results from applying the subdivision process. Thus, in Figure $1.6_{/9}$, if the physical positions of the corners of the initial cube were moved, the form of the resulting surface would be changed.

If we continue for a moment to assume that the control points p_ℓ represent physical position, then we can consider the case when the polyhedral mesh $\mathcal{M} = (M, p)$ corresponds to a polyhedron in a bounded subset of $\mathbb{R}^3$. (To establish such a correspondence, a suitable interpretation of a face as a curvilinear surface patch must be introduced, since the control points in a face may not be coplanar.) The nature of such polyhedra is quite well understood, due to the classification theorem for compact surfaces [100, Thm. 7.2]. A mesh without boundary must correspond either to a sphere or to the direct sum[6] of several tori. If we assume that the polyhedron is not self-intersecting, this means that the mesh is like the surface of a doughnut with zero or more holes, where parts of the surface may be intertwined, as illustrated in Figure $1.11_{/14}$. Another metaphor that is often used to describe the most general non-self-intersecting mesh without boundary is a coffee cup with zero or more handles. If the surface is pushed in, or indented, to form a depression, just where p_ℓ is indicated in Figure $1.11_{/14}$, the depression in the surface can be viewed as the basin of a coffee cup; then, the two intertwined circular parts of the surface can be viewed as intertwined handles on the cup. Finally, for the case of meshes with boundary, all such meshes can actually be modelled in $\mathbb{R}^3$, as disks with (perhaps twisted) strips of paper glued to their boundaries [100, Sec. 12].

While it is useful to have some feeling for the level of generality attainable by locally planar surfaces, as just outlined, we make no assumption in this book that the mesh satisfies the conditions mentioned in the previous paragraph. In particular, if the control points correspond to physical position, there is no assumption that $\mathcal{M} = (M, p)$ can somehow be interpreted as a non-self-intersecting polyhedron[7] in $\mathbb{R}^3$. In fact, as we have mentioned, we do not even assume that the control points correspond to physical position: they might instead correspond to colour, texture coordinates, surface normal vectors, or the sharpness of what are called semisharp boundary edges, as in Section 7.1.1. Alternatively, there could be more than one set of control points, some corresponding to physical position and some

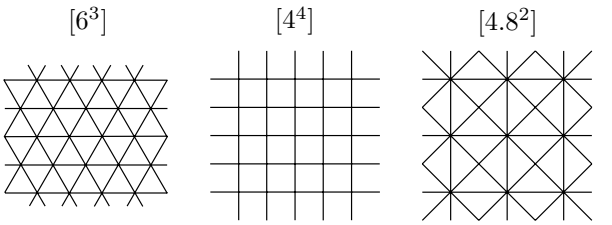


Figure 1.12. *Three regular tilings of the plane.*

not. Verifying that the results of subdivision have a useful interpretation, when the control points do not correspond to physical position, is a topic that merits close attention, but it is not discussed here. We view the control points p_ℓ simply as elements of $\mathbb{R}^N$.

1.2.2 Primal and dual subdivision methods

The conceptual starting point for the standard methods of subdivision can be viewed as a choice of a regular tiling of the plane, along with an associated splitting schema that allows us to subdivide a mesh corresponding to the tiling, into a mesh corresponding to a more refined version of the tiling or its dual. Smoothing rules are then introduced to compute new values of the control points on the refined mesh, and later, the rules are modified to take account of meshes that do not correspond exactly to the regular tiling.

We begin by defining a *regular* mesh. Regularity is defined with respect to one of the regular tilings of the plane. There is a large variety of such tilings [60]. Among the simplest are the eleven isohedral *Laves tilings* [60, pp. 96, 176], and almost all subdivision methods are based on meshes which, although not planar, are closely related to the three tilings shown in Figure 1.12$_{/16}$, namely[8] those denoted $[6^3]$, $[4^4]$, and $[4.8^2]$.

Let M be a locally planar mesh without boundary. We denote the number of edges in a face by e (see Definitions 1.2.1$_{/10}$ and 1.2.2$_{/10}$), and the number of edges incident at a vertex (the *valence*) by n. Then, a *regular triangular mesh* is defined as one for which all faces have $e = 3$ and all interior vertices have valence $n = 6$. Similarly, a *regular quadrilateral mesh* is defined[9] as one for which all faces have $e = 4$ and all interior vertices have valence $n = 4$. A regular *part* of a triangular or quadrilateral mesh is defined similarly, and the definition of a regular 4-8 mesh is given later, in Section 3.7.2.

If a mesh is considered to be triangular, but it is not a regular triangular mesh, then any vertex with $n \neq 6$ is called an *extraordinary vertex*, and any face with $e \neq 3$ is called an *extraordinary face*. Similarly, if a mesh is considered to be a quadrilateral mesh, but it is not a regular quadrilateral mesh, then any vertex with $n \neq 4$ is called an *extraordinary vertex*, and any face with $e \neq 4$ is called an *extraordinary face.*[10] Vertices and faces that are not extraordinary are called *ordinary*. Meshes considered to be triangular are usually made up of mostly triangles, and meshes considered to

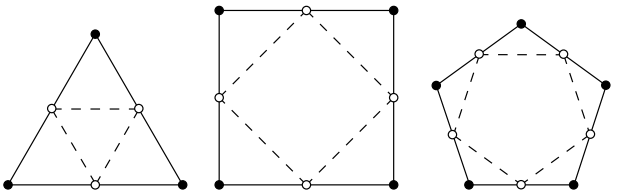


Figure 1.13. *The $pT4$ triangular split.*

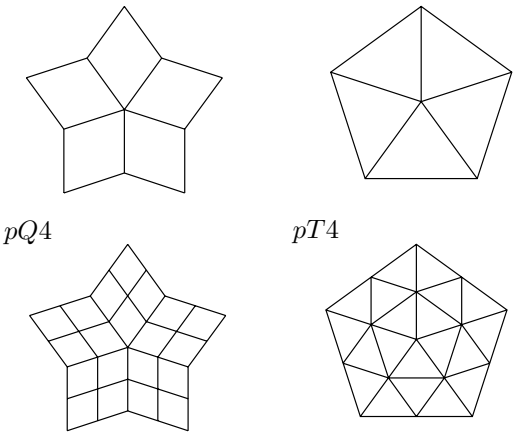


Figure 1.14. *The $pQ4$ and $pT4$ splittings.*

be quadrilateral are usually made up of mostly faces with $e = 4$, but there is no requirement that this always be so.

Associated with the underlying tiling is a splitting schema, which typically satisfies the condition that if the splitting schema is applied to the tiling, viewed as a mesh, then it produces a new version of the tiling, or its dual (which may be the same, as in the case of $[4^4]$).

The *primal* methods use rules that work directly with a refined version of M, defined in terms of its vertices, edges, and faces. One important type of *splitting* used in the context of meshes considered to be triangular proceeds as follows: the edges of each e-gon (a polygon with e edges) are split in two, and joined by new edges, as in Figure $1.13_{/17}$ for $e = 3, 4, 5$. We refer to this schema as *primal triangular 4-split* ($pT4$). With this schema, an e-gon in the original mesh is transformed into e triangles, and one new e-gon, in the refined mesh. In particular, if the original mesh contains only triangles ($e = 3$), then the refined mesh will contain only triangles. This is the usual case; see Figure $1.14_{/17}$ (right), For the $pT4$ schema,

- if there is an extraordinary face (i.e., a nontriangle) in the original mesh, it will remain after each subsequent subdivision step. No new extraordinary faces will be introduced after the first subdivision step.

- if there is an extraordinary vertex in the original mesh, it will remain after each subsequent subdivision step. No new extraordinary vertices will be introduced after the first subdivision step.

In the context of meshes considered to be quadrilateral, the procedure is slightly different. Again, the edges of each e-gon are split in two, but now, an additional vertex is added in the middle of the face. This additional vertex is joined by new edges to each of the new midedge vertices, as illustrated in Figure $1.15_{/18}$ for $e = 3, 4, 5$. As a result, an e-gon in the original mesh is transformed into e quadrilateral faces. Thus, after the first round of subdivision, the mesh will comprise only quadrilateral faces ($e = 4$). This schema is referred to as *primal quadrilateral* 4*-split* ($pQ4$): the case $e = 4$ is shown in Figure $1.14_{/17}$ (left).

Another kind of splitting schema applicable to meshes considered to be quadrilateral leads to the class of *dual* methods. We proceed as in the primal case, except that the procedure is followed by taking the dual of the mesh. Thus, the edges of each e-gon are split in two, an additional vertex is added in the middle of each face, and the additional vertex is joined by new edges to each of the new midedge vertices. So far, this is the same as the $pQ4$ schema. Now, however, there is an additional step: we take the dual of the mesh, as illustrated in Figure $1.16_{/19}$. This new schema[11] is referred to as *dual quadrilateral* 4*-split* ($dQ4$). When it is used, an e-gon is transformed into e quadrilateral faces, and a new vertex is associated with each new quadrilateral face in order to form the dual mesh. After the first round of subdivision, the mesh will contain only vertices with valence $n = 4$.

For the $pQ4$ and $dQ4$ splitting schema, we have the following dual sets of statements related to extraordinary vertices and faces. For the $pQ4$ schema,

- if there is an extraordinary face in the initial mesh, then the $pQ4$ schema will create a corresponding extraordinary vertex in the subdivided mesh, at the first step.

- this new extraordinary vertex, and any other extraordinary vertices that were already in the initial mesh, will remain after each subsequent subdivision step.

- no extraordinary faces will remain in the mesh after the first subdivision step, and no new extraordinary vertices will be introduced after the first subdivision step.

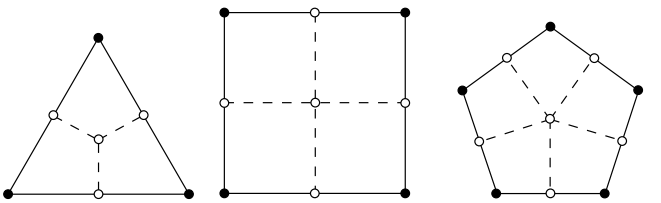


Figure 1.15. *The $pQ4$ quadrilateral split.*

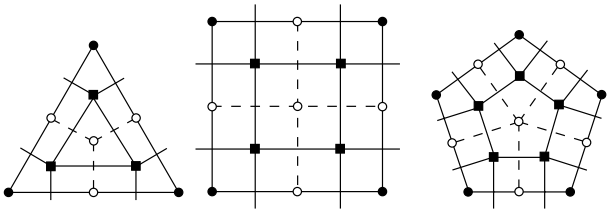


Figure 1.16. *The $dQ4$ dual quadrilateral split.*

Similarly, for the $dQ4$ schema,

- if there is an extraordinary vertex in the initial mesh, then the $dQ4$ schema will create a corresponding extraordinary face in the subdivided mesh, at the first step.
- this new extraordinary face, and any other extraordinary faces that were already in the initial mesh, will remain[12] after each subsequent subdivision step.
- no extraordinary vertices will remain in the mesh after the first subdivision step, and no new extraordinary faces will be introduced after the first subdivision step.

The dual of a triangular mesh that is to be subjected to $pT4$ splitting is a hexagonal mesh, but with a nonhexagonal face corresponding to each extraordinary vertex in the original mesh. A duality, of the sort just described for the $pQ4/dQ4$ case, could be established here, also, but hexagonal meshes are less frequently used in practice,[13] so we do not do this.

Remark 1.2.8. The $dQ4$ splitting procedure involves, in the first step, subdivision followed by taking the dual. The same procedure is followed in the next and subsequent steps, but this does *not* imply that the method alternates between the primal mesh and dual mesh at each step. The first step takes us to the dual of the subdivided primal mesh. The second step takes us not back to the primal, but, rather, to the dual of the *twice* subdivided primal mesh. After ν steps we end up in the dual of the ν-times subdivided primal mesh. Figure $1.17_{/20}$ illustrates both the two-dimensional and one-dimensional cases. The first step takes us to the dual mesh, indicated by solid black lines intersecting black squares (just as in Figure $1.16_{/19}$). At the second step this mesh is subdivided, as shown by the green lines in Figure $1.17_{/20}$ (top), and the dual is taken, producing the points shown by green triangles. These points are in the dual of the twice subdivided primal mesh.

It is shown below that the same remark applies to the process of alternate averaging (at the level of substeps) between the refined initial mesh for each step and the dual of this mesh. This will be described in some detail in Section 1.3.1. In the first step of a dual method we alternate an odd number of times between the subdivided primal and dual meshes. This process is repeated at the second step,

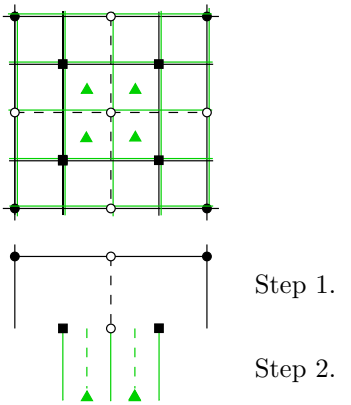


Figure 1.17. *Two steps lead to dual of twice subdivided primal mesh.*

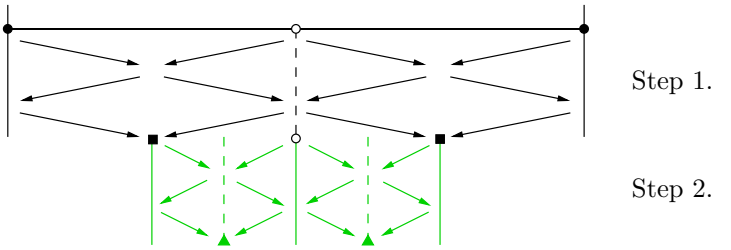


Figure 1.18. *The result of an odd number of alternations at each step.*

but this does not send us back to the primal: it takes us instead to the dual of the twice subdivided primal mesh, as illustrated in the one-dimensional case (supposing three alternations within each step) in Figure $1.18_{/20}$. ∎

It is shown later that there are other, different, splitting schema that can be used in connection with the $[6^3]$ tiling, but they have the same kind of extraordinary vertices (valence $n \neq 6$): an example is $\sqrt{3}$-subdivision. Similarly, the 4-8 subdivision method is defined in terms of a splitting schema that preserves the topology shown in Figure $1.12_{/16}$ (right), and regularity is defined accordingly.

The splitting schema involved is often clear from the context. For example, if the term "nonregular quadrilateral mesh" is used, it is clear that $pQ4$ or $dQ4$ splitting is involved, and that the situation which is considered normal or ordinary is defined by the $[4^4]$ tiling.

1.2.3 Stencils

Suppose now that a splitting schema, such as $pQ4$, $dQ4$, or $pT4$, has been chosen. Given the control points p of the polyhedral mesh $\mathcal{M} = (M, p)$, one step of a

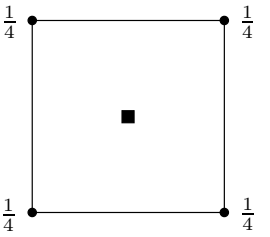


Figure 1.19. *A simple example of a stencil.*

subdivision method involves first the refinement of the logical mesh M, and second the computation of the control points for the refined mesh. The definition of this second computation is referred to as the specification of the smoothing rule for the method. Normally there is a fairly small number of cases for the local vertex-edge-face topology in the refined mesh, and to describe a step in the subdivision method it is sufficient to define how to compute the new control points for each such case. This can be done by specifying a *stencil* for each case. Stencils are primarily used to define or illustrate a subdivision rule. For example, a subdivision rule might introduce a new vertex in the middle of a quadrilateral face with a new associated point p equal to the average of the control points at the four corners, and the stencil would be as illustrated in Figure 1.19$_{/21}$.

Some authors refer to stencils as "subdivision masks," but we follow [25, 44, 124] and others, reserving the word "mask" for the array of nonzero coefficients of the subdivision polynomial introduced in Section 2.2 below.

Calculations defined by subdivision rules will sometimes be described as being done in *Jacobi manner*.[14] This means that the original unmodified data are is somehow kept available, perhaps by making a copy of the data, or by reading the input data, transforming it, and writing the output without overwriting any of the input data. Thus, all calculations of new values are made using the original (nonupdated) version of the data. Computations done in Jacobi manner by making a copy of the data are in contradistinction to *in-place* computations.

1.3 Examples and classification

In this section we present the principal subdivision methods used in practice in the case of meshes without boundary. Auxiliary rules for the case of meshes with boundaries are given in Chapter 7. We begin with a summary description of some of the most commonly discussed algorithms. Following is a classification of subdivision methods, based on a hierarchy of the spline surfaces generated by the various methods. All of the algorithms studied in the book are discussed in terms of this classification.

Following presentation of the main classification, cross-classifications are given, based on criteria such as mesh type, interpolation versus approximation, etc.

1.3.1 The methods of Catmull–Clark, Doo–Sabin, and Loop

The Catmull–Clark [24] and Doo–Sabin [45] methods were among the first surface-subdivision methods introduced, and the Catmull–Clark, along with the Loop [91] method, has become a graphics industry standard. The Doo–Sabin and Catmull–Clark methods provide an example of one approach to generating[15] subdivision algorithms. We start with the class of tensor-product uniform B-splines of order $m = d + 1$, and we observe that there is a method for the evaluation of such functions based on subdivision of the infinite regular quadrilateral grid (see Figure $1.2_{/2}$) in $\mathbb{R}^2$, namely, the Lane–Riesenfeld algorithm. We then observe that the steps of the algorithm depend only on control points available locally. This permits us to change our point of view and apply the subdivision algorithm to regular portions of a *finite* polyhedral mesh and then to propose extensions of the subdivision rules that deal with nonregular portions of the finite mesh. We view this extended method as a *variant* of the *basic* subdivision method: only the variant method is implemented, but it reduces to the basic method in regular parts of the mesh (and this fact is crucial for purposes of analysis).

Two such variants are the Doo–Sabin ($d = 2$) and Catmull–Clark ($d = 3$) methods, but there are others, including the *Repeated Averaging* variant (d arbitrary), which is described first. The Repeated Averaging variant is a very natural and intuitive method, and the Doo–Sabin and Catmull–Clark algorithms can be viewed as minor variations of Repeated Averaging.

Finally, we also describe the Loop method, which is a variant of a method for regular triangular grids.

Tensor-product uniform B-splines and the Lane-Riesenfeld algorithm

Suppose we are given a two-dimensional regular quadrilateral grid with grid-size h, bi-infinite in both dimensions, and with control points $p_{k,l} \in \mathbb{R}^N$ defined on $h\mathbb{Z}^2 = \{(kh, lh)^t : (k, l) \in \mathbb{Z}^2\}$. A *tensor-product uniform B-spline* of bidegree d is the piecewise polynomial

$$x(u, v) = \sum_{(k,l)\in\mathbb{Z}^2} p_{k,l} N_k^m(h; u) N_l^m(h; v), \quad (u, v) \in \mathbb{R}^2,$$

given in $(1.1)_{/2}$. Here, $N_k^m(h; u)$ and $N_l^m(h; v)$ are the centered B-spline basis functions, introduced formally in Chapter 2, and we have used a standard notation for the order m, which is one more than the degree d. Although the summation is over the infinite grid, at a particular parameter value (u, v), only m of the basis functions $N_k^m(h; u)$, and m of the basis functions $N_l^m(h; v)$, are different from zero.

The bivariate Lane–Riesenfeld algorithm for the evaluation of the surface $x(u, v)$ with $m = d + 1$ is denoted $LR(d \times d)$ and consists of a linear subdivision substep followed by $d - 1$ averaging substeps [151]. We consider first the case $d = 2$.

A single step of the $LR(2 \times 2)$ algorithm is illustrated in Figure $1.20_{/23}$. To begin, the control points $p_{k,l}$ are associated with the rectangular grid of points in $h\mathbb{Z}^2$ (Figure $1.20_{/23}$, left). For $d = 2$, each step of the $LR(2 \times 2)$ algorithm proceeds

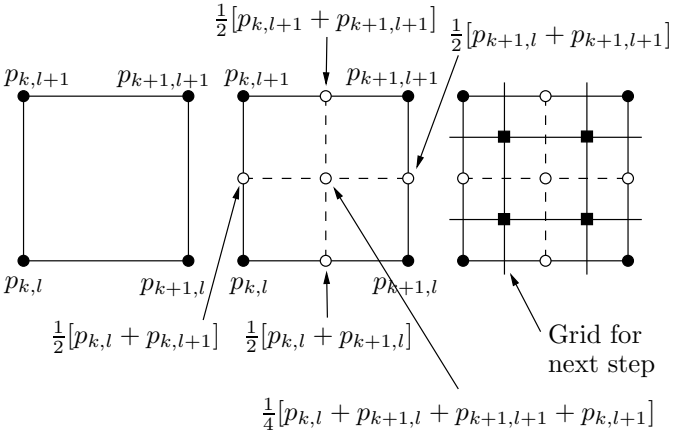


Figure 1.20. *The $LR(2 \times 2)$ algorithm (one step) viewed in parameter space.*

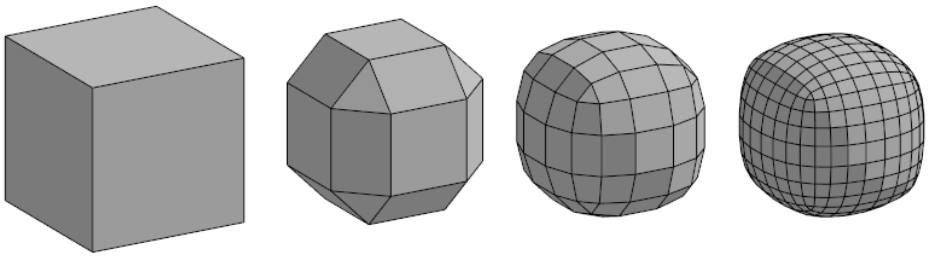


Figure 1.21. *The $LR(2 \times 2)$ algorithm (three steps) viewed in $\mathbb{R}^N$.*

in two substeps. In the linear subdivision substep, the grid is first subdivided so that the grid-size is halved, and new (but temporary) points in the mesh are calculated by averaging along the grid lines and computing the centroid $\frac{1}{4}[p_{k,l} + p_{k+1,l} + p_{k+1,l+1} + p_{k,l+1}]$ of the corner control points (Figure $1.20_{/23}$, middle). Then, a single $(d-1=1)$ averaging substep is executed, producing a new array of points. The vertices of this new array are denoted by black squares in Figure $1.20_{/23}$ (right): the new control point for each black square is calculated as the average of the four neighbouring points computed in the first substep. Thus, for example, the control point for the lower left black square is easily verified to be

$$\frac{9}{16}p_{k,l} + \frac{3}{16}p_{k+1,l} + \frac{3}{16}p_{k,l+1} + \frac{1}{16}p_{k+1,l+1} \; . \tag{1.2}$$

The network of points in $\mathbb{R}^N$, for $N = 3$, after one complete step of the algorithm applied to the unit cube, is shown in the *second* illustration of Figure $1.21_{/23}$.

The single step of the biquadratic Lane–Riesenfeld algorithm $LR(2 \times 2)$, just described, is repeated as often as desired, producing a finer and finer mesh at

each step. Each step consists of a linear subdivision substep and an averaging substep. Figure $1.21_{/23}$ also shows the results of the second and third steps of the algorithm. In the limit, the network of points exactly matches (in a sense to be made precise) the form of $x(u, v)$.

We consider next the case $d = 3$. In this case, in each step, the initial linear subdivision substep is followed by $d - 1 = 2$ averaging substeps. Thus, in addition to the substeps illustrated in Figure $1.20_{/23}$, the bicubic Lane–Riesenfeld algorithm $LR(3 \times 3)$ executes one more averaging substep within each step. To illustrate, before proceeding to the subsequent step, the method averages the values associated with the four black squares in Figure $1.20_{/23}$ (right), and the computed average is associated with the centre node in the subdivided primal mesh.

For larger values of d, $d - 1$ averaging substeps follow the linear subdivision substep within each step. If d is even, the initial mesh for the next step is the dual mesh, as in Figure $1.17_{/20}$ (top), while if d is odd, the initial mesh for the next step is the primal mesh.

Later it will be convenient to divide the linear subdivision substep itself into two smaller substeps, but this is just a difference in terminology.

When $d = 4$, the $LR(d \times d)$ method is called *Biquartic subdivision* [176, p. 82].

The Repeated Averaging variant

It can be shown that the $LR(d \times d)$ algorithm converges to the B-spline surface $x(u, v)$ defined by $m = d + 1$. Furthermore, we see that the computations involved in each step comprise only local averagings, so that we can decide to apply them directly in *finite* regular quadrilateral meshes. In addition, there is a very natural way to define a variant method applicable in regions of the mesh involving extraordinary vertices and faces (such as the mesh illustrated in Figure $1.22_{/24}$). First, in the linear subdivision substep, when computing the control-point value at the new point in the middle of a face, the value $\frac{1}{4}p_{k,l} + \frac{1}{4}p_{k+1,l} + \frac{1}{4}p_{k+1,l+1} + \frac{1}{4}p_{k,l+1}$ is replaced by the centroid, i.e., the average of the e control points around the face, where e is the number of edges (and vertices) in the face. Second, in the averaging substep, the four-point averaging process is replaced by computation of the centroid of the face. The resulting algorithm for meshes without boundary can be described in terms of two procedures [155] named *LinSubd* and *Dual*.

The procedure *LinSubd* takes a locally planar polyhedral mesh $\mathcal{M} = (M, p)$ and subdivides it according to the $pQ4$ schema, assigning new values to p as follows:

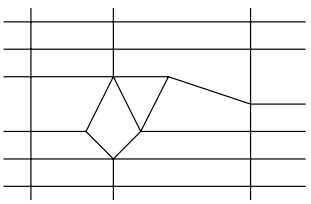


Figure 1.22. *An example of a nonregular mesh.*

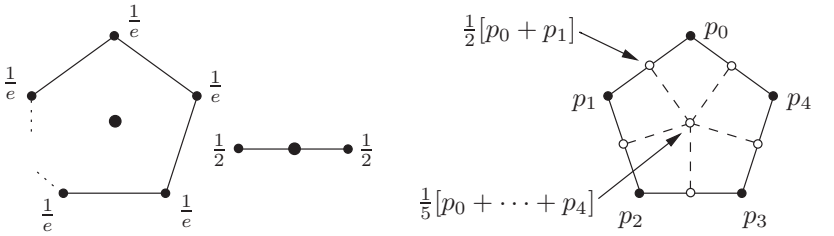


Figure 1.23. *Linear subdivision: Stencils and subdivided face.*

the new control point at a new vertex in the middle of an edge (ℓ, ℓ') is assigned the value $\frac{1}{2}[p_\ell + p_{\ell'}]$, and the new control point in the middle of a face defined by $(\ell_0, \ldots, \ell_{e-1})$ is assigned the value $\frac{1}{e}\sum_{i=0}^{e-1} p_{\ell_i}$. These two rules are illustrated in Figure 1.23$_{/25}$ (left). The values at existing vertices are left undisturbed. Edges are added to connect the new face points to the new edge points, so that the new mesh contains only quadrilateral faces. The linearly subdivided face is illustrated in Figure 1.23$_{/25}$ (right) for the case $e = 5$. The second procedure, called *Dual*, takes a polyhedral mesh $\mathcal{M} = (M, p)$ and replaces it by a polyhedral mesh having as logical mesh the dual of M, with control points defined by the centroids of faces in $\mathcal{M}$. Thus, $Dual(\mathcal{M})$ is a new polyhedral mesh whose vertices have control points equal to the centroids of faces of $\mathcal{M} = (M, p)$, and whose edges join centroids of faces that share a common edge in M [151, p. 387].

The Repeated Averaging algorithm, a variant method that can be used for nonregular meshes, is then defined by the following pseudocode. (Note, however, that in practice the method would not be implemented in this way; see the discussion below.)

Algorithm. Repeated Averaging.

Input: $\mathcal{M}^0$, $d \geq 1$, $\lambda \geq 0$
Output: A mesh subdivided to level λ with a degree-d process

function *RepeatedAveraging*($\mathcal{M}^0$, d, λ)
for $\nu = 1$ **to** λ **do**
  $\mathcal{M}^\nu \leftarrow LinSubd(\mathcal{M}^{\nu-1})$
  for $j = 1$ **to** $d - 1$ **do**
    $\mathcal{M}^\nu \leftarrow Dual(\mathcal{M}^\nu)$
  end
end
return $\mathcal{M}^\lambda$
end *RepeatedAveraging*

The logical mesh associated with odd values of d ($d = 1, 3, 5, \ldots$) is the mesh M_{odd} produced by a single invocation of *LinSubd*, i.e., $\mathcal{M}_{odd} = LinSubd(\mathcal{M}^0)$. The logical mesh M_{even} associated with even values of d ($d = 2, 4, 6, \ldots$) is the dual

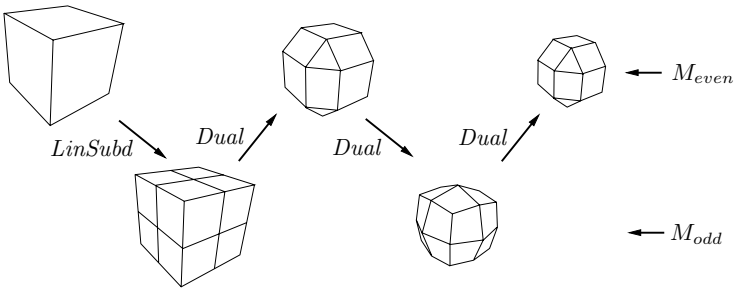


Figure 1.24. *Logical meshes used by RepeatedAveraging.*

of this mesh. The logical meshes M_{odd} and M_{even} can be seen in the two rows in Figure 1.24$_{/26}$ [151, Fig. 3]. For $\nu > 1$, the input polyhedral mesh $\mathcal{M}^0$ is replaced by $\mathcal{M}^{\nu-1}$ from the previous step.

The Repeated Averaging variant of the $LR(d \times d)$ algorithm is called the *Simple* algorithm in [151]; see also [101, 128, 177].

Isolation of extraordinary vertices and faces

In going from a regular planar grid to a nonregular locally planar mesh, there are two kinds of changes involved: we have moved from planar grids to locally planar meshes (such as meshes in the form of a sphere, or a torus with one or more holes), and from regular to nonregular meshes.[16] If an algorithm like the Repeated Averaging variant is applied, however, the basic $LR(d \times d)$ algorithm remains relevant, since large parts of the mesh may be regular. In fact, as described in Section 1.2.2 above, subdivision using the $pQ4$ schema does not introduce new extraordinary vertices after the first step, and subdivision using the $dQ4$ schema does not introduce new extraordinary faces after the first step. This means that as subdivision proceeds, the extraordinary vertices ($pQ4$) or faces ($dQ4$) become (topologically) more and more isolated, and the regular portions of the mesh become larger and larger (see Figure 1.25$_{/27}$, where there are two extraordinary vertices in the $pQ4$ case, and two extraordinary faces in the $dQ4$ case). Thus, over submeshes corresponding to almost all of the subdivided mesh, basic subdivision methods can be analysed in terms of their corresponding classical B-splines. Only extraordinary vertices or faces of the mesh will require a special analysis of the variant method and its associated spline, and the number of such vertices or faces remains fixed as the subdivision proceeds.

The Doo–Sabin variant

The Doo–Sabin variant of the $LR(2 \times 2)$ method is almost identical to the Repeated Averaging variant with $d = 2$. Both methods apply the $dQ4$ schema and assign values to the control points associated with the nodes in the subdivided dual mesh. These values depend only on the control points of the parent face. For the Doo–Sabin variant, if the parent face had e vertices ℓ_j, $j = 0, \ldots, e-1$ (see Figure 1.26$_{/27}$),

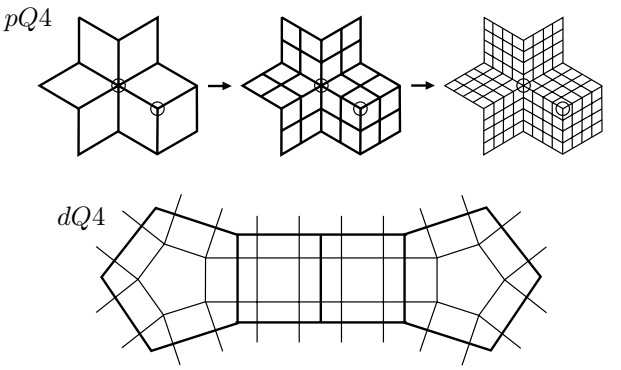


Figure 1.25. *Extraordinary vertices and faces become topologically isolated.*

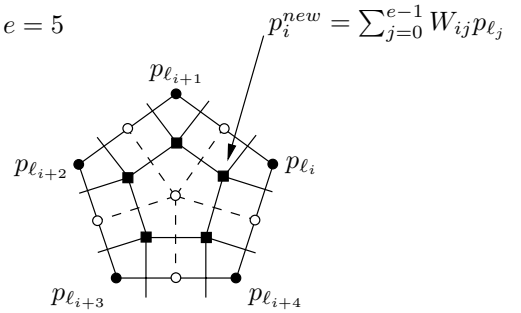


Figure 1.26. *Stencil for Doo–Sabin method.*

then the value p_i^{new} assigned to a new face point, for a face having ℓ_i as one of its vertices, is

$$p_i^{new} = \sum_{j=0}^{e-1} W_{ij} \cdot p_{\ell_j},$$

where[17]

$$W_{ij} = \begin{cases} \frac{e+5}{4e}, & j = i, \\ \frac{3+2\cos(2\pi(i-j)/e)}{4e}, & j \neq i. \end{cases} \tag{1.3}$$

These weights can be compared with those used by the Repeated Averaging variant, which are given in Exercise $5_{/48}$.

The choice of weights in $(1.3)_{/27}$ is related to the spectral properties of a subdivision matrix associated with the method; see Section 5.8. We note here that

$W_{ij} > 0$ and

$$\sum_{j=0}^{e-1} W_{ij} = \frac{e+5}{4e} + \frac{3(e-1) + 2\sum_{k=1}^{e-1} \cos(2\pi k/e)}{4e} = 1 \tag{1.4}$$

(see Exercise $6_{/48}$). The reason for choosing weights satisfying $(1.4)_{/28}$ is that this condition guarantees affine invariance of the method (see Section 1.4.1).

A practical difficulty with dual subdivision methods, such as the Doo–Sabin method, is that it may be difficult to associate attributes, such as colour or texture coordinates, with the dual mesh. The difficulty can be seen in the top centre of Figure $1.24_{/26}$, where the result of the first *Dual* operation is the mesh resulting from a single Doo–Sabin step. If the colours of the three visible faces of the original cube were, say, red, white, and blue, then it would not be clear what colour should be assigned to the triangular face in the dual mesh. This is an example of what in solid modelling is called the "persistent naming problem" [99]. One solution to the problem in the case that concerns us here is the Arbitrary-degree method [155], mentioned in the context of the LSS variant, below.

The Catmull–Clark variant

The Catmull–Clark variant of the $LR(3 \times 3)$ method is almost identical to the Repeated Averaging variant with $d = 3$. Both methods apply the $pQ4$ schema and assign values to the control points in the refined primal mesh.

We begin by examining the Repeated Averaging variant for $d = 3$ more closely. Since the Repeated Averaging variant with $d = 3$ involves two averaging substeps after the linear subdivision substep, there is no need to actually construct the dual mesh: the two averagings can be combined and the values associated directly with the refined primal mesh $\mathcal{M}_{odd}$.

Suppose for simplicity that there are no extraordinary faces, i.e., all faces in the mesh have four edges. (This will always be the case if at least one $pQ4$ step has already been executed.) Departing from our usual notation of p_ℓ for control points, denote the initial control points surrounding a vertex of valence n by $E_0, E_1, \ldots, E_{n-1}, F_0, F_1, \ldots, F_{n-1}$, and the values produced by the Repeated Averaging variant with $d = 3$ by $E'_0, E'_1, \ldots, E'_{n-1}, F'_0, F'_1, \ldots, F'_{n-1}$, as illustrated in Figure $1.27_{/29}$ for $n = 5$ [8, 9]. Linear subdivision produces the values

$$\begin{aligned} E_i^L &= \frac{1}{2}\big(V + E_i\big), \\ & \qquad\qquad i = 0, \ldots, n-1, \\ F_i^L &= \frac{1}{4}\big(V + E_i + F_i + E_{i+1}\big), \end{aligned} \tag{1.5}$$

where the indices are calculated modulo n. Then, it is clear that the result of the two averaging substeps is to replace each of the points E_i^L and F_i^L by a new value obtained by applying the smoothing stencil shown in Figure $1.28_{/29}$ (left) in Jacobi manner in the subdivided mesh, and to replace V by the new value

$$V^{RA} = \frac{1}{n}\sum_{j=0}^{n-1}\left[\frac{1}{4}\big(V + E_j^L + E_{j+1}^L + F_j^L\big)\right].$$

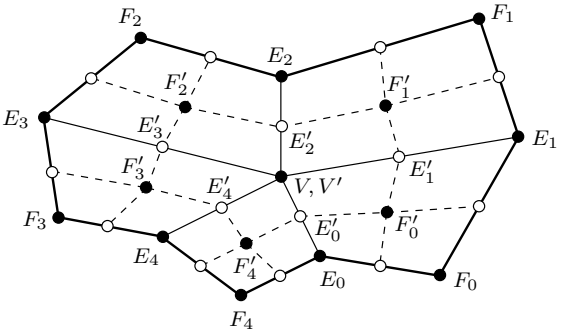


Figure 1.27. *Notation for Catmull–Clark vertices.*

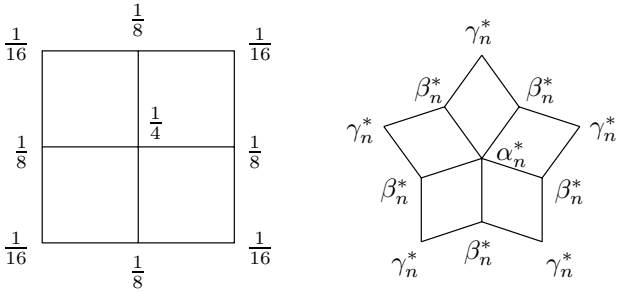


Figure 1.28. *Catmull–Clark smoothing stencil.*

It is easy to show that this smoothing leaves the values F_i^L unchanged, so that

$$F_i' = F_i^L = \frac{1}{4}(V + E_i + F_i + E_{i+1}), \qquad i = 0, \ldots, n-1, \tag{1.6}$$

that it replaces the values E_i^L by

$$E_i' = \frac{3}{8}(V + E_i) + \frac{1}{16}(E_{i-1} + F_{i-1} + F_i + E_{i+1}), \qquad i = 0, \ldots, n-1, \tag{1.7}$$

and it replaces the value of V by

$$V^{RA} = \frac{9}{16}V + \frac{3}{8}\left(\frac{1}{n}\sum_{j=0}^{n-1} E_j\right) + \frac{1}{16}\left(\frac{1}{n}\sum_{j=0}^{n-1} F_j\right). \tag{1.8}$$

Exercise $7_{/48}$ asks for confirmation of this.

The Catmull–Clark method does exactly the same thing as the Repeated Averaging variant with $d = 3$, except that $(1.8)_{/29}$ is replaced by the smoothing stencil shown for $n = 5$ in Figure $1.28_{/29}$ (right), where $\alpha_n^* = (n-3)/n$, $\beta_n^* = 2/n^2$, $\gamma_n^* = 1/n^2$. For $n = 4$ the replacement causes no change, since $\alpha_4^* = 1/4$, $\beta_4^* = 1/8$, and $\gamma_4^* = 1/16$ coincides exactly with the smoothing stencil shown in

Figure $1.28_{/29}$ (left). The values of new face points and new edge points are therefore the same as for the Repeated Averaging method, but the value $(1.8)_{/29}$ is replaced by

$$V' = \alpha_n^* V + \beta_n^* \sum_{j=0}^{n-1} E_j^L + \gamma_n^* \sum_{j=0}^{n-1} F_j^L. \tag{1.9}$$

The Repeated Averaging method corresponds to using a smoothing stencil with α_n^* replaced by $1/4$, β_n^* replaced by $1/(2n)$, and γ_n^* replaced by $1/(4n)$ (again, see Exercise $7_{/48}$). The surfaces produced by this smoothing were rejected in [24] as "too pointy."

Catmull–Clark variant (in-place formulation)

The method as described above involves Jacobi-manner smoothings, including one which is redundant, but an in-place version of the method can also be given. Note first that the expression $(1.9)_{/30}$ for the updated value of V is equivalent to

$$V' = \alpha_n^* S + n\beta_n^* R + n\gamma_n^* Q, \tag{1.10}$$

where

$$\begin{aligned} S &= V, \\ R &= \frac{1}{n} \sum_{j=0}^{n-1} \frac{1}{2}(V + E_j), \\ Q &= \frac{1}{n} \sum_{j=0}^{n-1} F_j'. \end{aligned}$$

That this is equivalent to $(1.9)_{/30}$ can be confirmed by an easy algebraic verification; see Exercise $8_{/49}$.

Now suppose that the current values of V and the E_i, F_i are stored, and that storage has been allocated for each new face point F_i' and each new edge point E_i'. Then, calculating indices modulo n, do the following:

1. Begin the computation of E_i' :
   $E_i' \leftarrow \frac{1}{2}(V + E_i)$.
2. Compute F_i' : $F_i' \leftarrow$ centroid of the face's old vertex points.
   For quadrilateral faces, this means simply:
   $F_i' \leftarrow \frac{1}{2}E_i' + \frac{1}{4}[F_i + E_{i+1}]$.
3. Compute the modified vertex point V' :
   $V' \leftarrow \alpha_n^* V + n\beta_n^* R + n\gamma_n^* Q$.
4. Complete the computation of E_i' :
   $E_i' \leftarrow \frac{1}{2}[E_i' + \frac{1}{2}(F_{i-1}' + F_i')]$.

In substep 3, R is the average of the midpoints of all edges incident on the vertex (the required values are available from substep 1), and Q is the average of the new

face points for faces adjacent to the vertex (the required values are available from substep 2). The final value of E_i' calculated in substep 4 is equal to $1/4(V + E_i) + 1/4(F_{i-1}' + F_i')$, which is the same as the value in $(1.7)_{/29}$.

This in-place version is exactly the original Catmull–Clark formulation, and the notation Q and R used here is the same as that used in [24]. The only difference between the two formulations is the order of computation, which has been rearranged here to obtain an in-place computation.

The in-place formulation of the Catmull–Clark algorithm is equivalent to the one given by $(1.5)_{/28}$, $(1.6)_{/29}$, $(1.7)_{/29}$, and $(1.9)_{/30}$, except that the edge point E_i' calculated at the very first step of the algorithm may be different in the case when there is a nonquadrilateral face in the initial mesh. Note also that auxiliary rules are used for surface boundaries and crease edges; see Section 7.1.1.

The following remark concerning Catmull–Clark formulations may be useful.

Remark 1.3.1. We have chosen to introduce the Catmull–Clark method with a presentation close to the one in [151]. This is by far the simplest way to understand the method: it is just the Repeated Averaging method with a modified choice of weights for extraordinary vertices. Algebraically, this formulation (as well as the in-place formulation and the original Catmull–Clark presentation [24]) can be summarized by $(1.5)_{/28}$, $(1.6)_{/29}$, $(1.7)_{/29}$, and $(1.9)_{/30}$.

On the other hand, there are two other formulations of the Catmull–Clark method that are often used in the literature, and later in the book, both of which express V' in terms of the original control points E_j and F_j, rather than E_j^L and F_j^L. One of these expressions is given in the context of subdivision matrices (see $(1.17)_{/44}$, below). Another is

$$V' = \frac{n-3}{n}V + \frac{2}{n^2}\left[\sum_{j=0}^{n-1}\frac{1}{2}(V + E_j)\right] + \frac{1}{n^2}\left[\sum_{j=0}^{n-1}\frac{1}{4}(V + E_j + F_j + E_{j+1})\right], \quad (1.11)$$

which follows immediately from $(1.6)_{/29}$ and $(1.10)_{/30}$.

A summary of the algebraic equivalence of the various formulations is given in the Appendix (Section A.1). ∎

Linear Subdivision and Smoothing (LSS) variant

Just as pairs of averaging substeps were combined in the case of the Catmull–Clark variant, we can combine pairs of averaging substeps for the Repeated Averaging variant for larger values of d. This gives a variant [151] that we call the Linear Subdivision and Smoothing (LSS) variant.

In the case when d is odd this variant invokes the procedure *LinSubd* once, followed by $(d-1)/2$ smoothings *Smooth*, where *Smooth* is such that $Smooth(\cdot) \equiv Dual(Dual(\cdot))$ for quadrilateral meshes perhaps having extraordinary vertices. In the case when d is even, we must apply *Dual* once to obtain $\mathcal{M}^2 = \mathcal{M}_{even}$. Then, if $d > 2$ ($d = 4, 6, 8, \ldots$), this is followed within each step by $(d-2)/2$ smoothings *Smooth*, each of which again duplicates the effect of two applications of *Dual*. Exercise $9_{/49}$ asks for pseudocode in the two cases.

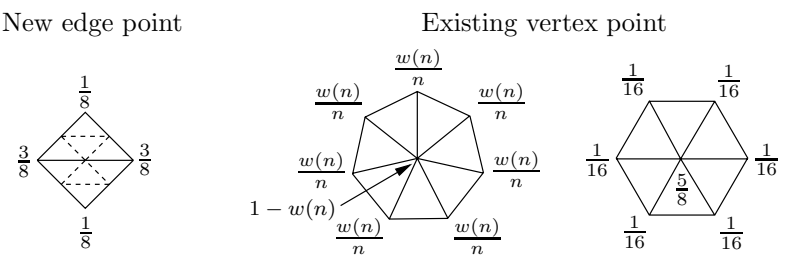


Figure 1.29. *Loop stencils.*

A modification of the LSS variant that permits implementation without explicit use of the dual mesh, for arbitrary values of the degree d, was proposed in [155]. This algorithm solves the problem of persistent assignment of attributes, such as colour, that was mentioned above in the context of the Doo–Sabin variant. It also permits interpolation or "morphing" between surfaces of even and odd degree.

Loop variant

All of the variant methods mentioned so far have been variants of some form of the $LR(d \times d)$ algorithm for uniform tensor-product B-splines defined over a regular quadrilateral grid. The Loop method is a variant of a different basic method.

There is a class of splines defined on $\mathbb{R}^2$ that is more general than the uniform tensor-product B-splines, and methods can be derived from these more general splines in a completely analogous way. One such method uses subdivision to compute a certain box spline called a three-direction quartic box spline. It turns out that this defines a basic method defined on regular triangular meshes in $\mathbb{R}^2$ (see Figure $1.12_{/16}$, left), and the Loop method is a variant of this method for triangular meshes with vertices having valence other than 6. We describe this variant here, leaving aside for now the description of the underlying basic method.

In the Loop method [91], the mesh M is triangular ($e = 3$ for each face). At each subdivision step, the edges of each triangle are split in two to create a new vertex *(new edge point)* in the middle of the edge. These points are then joined, so that four subtriangles are created. The method is therefore based on the $pT4$ schema (Figure $1.14_{/17}$, right). Recall that a vertex is called *extraordinary* in this case if $n \neq 6$.

There are two kinds of vertices: values are assigned to new edge points according to the weights indicated in the stencil in Figure $1.29_{/32}$ (left); thus, values assigned to new edge points depend only on values of the control points at the vertices of the two neighbouring parent triangles. Values are assigned to existing vertex points according to the weights indicated[18] for $n = 7$ and $n = 6$ in the stencil in Figure $1.29_{/32}$ (right). Here,

$$w(n) = \frac{5}{8} - \left(\frac{3}{8} + \frac{1}{4}\cos\left(\frac{2\pi}{n}\right)\right)^2, \tag{1.12}$$

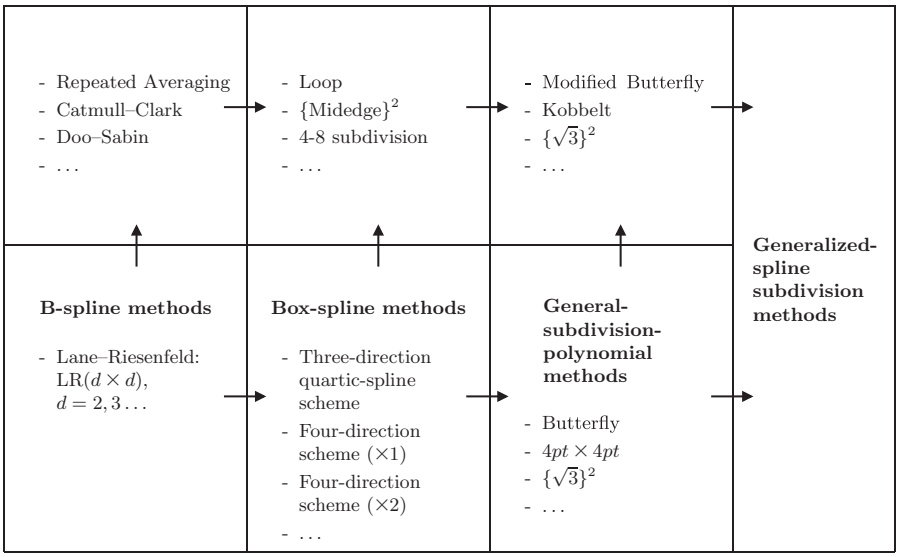


Figure 1.30. *Basic methods (lower row) and variant methods (upper row).*

and when $n = 6$ (an ordinary vertex) we have $w(n)/n = 1/16$. Note that $w(n) > 0$, and that in both cases in Figure 1.29$_{/32}$ the sum of the weights is equal to 1, again to guarantee affine invariance.

1.3.2 A classification of subdivision methods

Despite the large number of mesh topologies, splitting strategies, and averaging rules available, there is a great deal of unity and structure in the class of subdivision methods actually used, and one of our major goals is to show this unity and structure. In this section we give a classification that is used as a reference throughout the book. As elsewhere in this chapter, the concepts and methods described here are presented only in an intuitive way: most of the mathematical definitions and justifications are postponed until later.

The classification: Summary discussion

The entire field of subdivision-surface methods can be arranged in a simple and comprehensive hierarchy defined by the classes of spline functions the methods generate. The structure of the hierarchy is give in Figure 1.30$_{/33}$, which is explained now.

We first give a summary of Figure 1.30$_{/33}$. Each of the seven rectangles in the figure can be viewed as representing either a class of methods, or the class of subdivision surfaces these methods generate. The lower row of the figure corresponds to basic methods, applicable only on regular meshes, while the upper row corresponds to variant methods that apply also to nonregular meshes. Thus, the basic methods

are special cases of the variant methods. Similarly, as we move from left to right in the figure, the level of generality increases: the B-spline methods are a special kind of box-spline methods, the box-spline methods are a special case of the General-subdivision-polynomial methods, and all are special cases of the class we have called the Generalized-spline subdivision methods. Consequently, the arrows in the figure can be interpreted as "is a special case of" if we are thinking of the rectangles as representing classes of methods, or as "is a subset of" if we are thinking of the rectangles as representing the classes of subdivision surfaces the methods generate. We keep the two interpretations in view, since both are convenient, depending on circumstances.

Several particular methods are shown in Figure $1.30_{/33}$. In the lower left-hand rectangle is the Lane–Riesenfeld method of bidegree d, denoted $LR(d \times d)$, a basic method for regular quadrilateral meshes described in Section 1.3.1 above. In the upper left-hand rectangle are variants of this method, suitable for nonregular meshes. In the second column, in the lower row, are listed some common box-spline methods (the notation $\times 1$ and $\times 2$ indicates the multiplicity with which the four directions are included). In the upper row of the second column are the corresponding variant methods for nonregular meshes: for example, the Loop method is a variant of the three-direction quartic-spline scheme. The structure of the third column is similar: the basic methods are indicated in the lower row, and the variant methods[19] for nonregular meshes are shown in the upper row. The methods shown in the third column of the figure lie outside the class of box splines and their variants.

In Figure $1.30_{/33}$, the notation $\{\ldots\}^2$ signifies a method obtained by applying the underlying method (Midedge or $\sqrt{3}$-subdivision) twice in succession.

The class "Generalized-spline subdivision methods," shown on the right in Figure $1.30_{/33}$, is defined in Chapter 4. It serves as a universal set and provides a convenient mathematical framework for the most general case.

Box-spline methods

We turn now to the class shown in the lower row of the *second* column of Figure $1.30_{/33}$, namely the box-spline methods. The *box splines* are a more general class than the uniform B-splines. Tensor-product uniform B-splines can be viewed as the result of a projection of a hyper-rectangle (a box) in a higher-dimensional space, to the plane. By orienting this box in different ways (and by choosing the vectors defining the box in different ways), we obtain a more general class of splines, including many defined on nonrectangular grids in the plane [38, 129]. These are the box splines. Furthermore, there are specific binary subdivision rules associated with this new class of functions, which permit their evaluation using subdivision, in analogy with tensor-product B-splines. One example of a box spline that is not a tensor-product uniform B-spline is the three-direction quartic box spline. It has an associated subdivision process that can be viewed as being defined on a triangular grid. But, as in the case of the tensor-product B-splines, we can change our point of view and consider application of this subdivision process to the regular portions of finite locally planar meshes. This leads to a basic subdivision method for regular triangular meshes.

A variant of the method is again necessary for nonregular meshes, to deal with extraordinary vertices (which, because we are now dealing with $pT4$ splitting of a triangular grid, means $n \neq 6$). As already mentioned, the Loop method is such a variant of the three-direction quartic-spline scheme.

The class of variant box-spline methods also includes other well-known methods. In particular, it includes the "simplest subdivision method" [121], denoted *Midedge*. If this method is applied twice (denoted $\{\text{Midedge}\}^2$), then in a regular quadrilateral mesh the method produces a certain four-direction box spline. Similarly, 4-8 subdivision [161, 162, 164], Quasi 4-8 subdivision [161, 162, 163], and $\sqrt{2}$-subdivision [88] produce a (different) four-direction box spline in a regular quadrilateral mesh.

Generalized-spline subdivision methods

Corresponding to the *fourth* column of Figure $1.30_{/33}$, we introduce a class of generalized splines that imposes only the weakest of conditions on the subdivision process. A generalized spline is a linear combination of nodal functions with compact support produced by applying some affine-invariant subdivision process to a scalar control point ($N = 1$) that has value 1 at one vertex in the mesh, and 0 elsewhere. Such *generalized splines* correspond to the Generalized-spline subdivision methods. They are used, for example, in Section 7.3.1, where multiresolution subdivision surfaces are discussed.

The nodal functions involved in the definition of generalized splines are not necessarily linearly independent, need not be nonnegative, and need not be piecewise polynomials. Their domain of definition is a two-dimensional topological manifold. The manifold and the nodal functions are defined in Chapter 4. In the special case of B-splines the nodal functions are the nonnegative piecewise-polynomial basis functions in $(1.1)_{/2}$.

The requirements for membership in the class of generalized splines are very weak: the class includes the splines generated by any plausibly useful stationary subdivision method. In particular, it includes the class of box splines and their variants, as well as the limit surfaces produced by the following other well-known subdivision methods that are not box-spline methods:

- the Butterfly scheme [49], and the Modified Butterfly scheme [178] which is designed for nonregular triangular meshes;
- the Kobbelt scheme [73] for nonregular quadrilateral meshes, which, in the regular case, is the tensor product with itself of a method called the "four-point scheme for curves" [46, 48] (indicated by $4pt \times 4pt$ in Figure $1.30_{/33}$);
- the $\{\sqrt{3}\}^2$ method [76] for nonregular triangular meshes.

In the regular case, the methods just listed lie in a certain class of basic methods more special than the Generalized-spline subdivision methods, namely the class of *General-subdivision-polynomial* methods (Figure $1.30_{/33}$, *third* column). These methods could also be called *shift-invariant* methods, since the linear combination of nodal functions involves, in this case, shifted versions of a single nodal function.

The Generalized-spline subdivision methods include all of the variant methods shown in Figure 1.30$_{/33}$, upper row. Those in the third column of the figure are discussed in Chapter 4.

Further discussion of classification

The above classification seems to best reflect the mathematical structure of the field. Traditionally, however, subdivision methods have been classified according to other criteria [44, 95], [176, p. 65], even though these criteria do not always lead to sharp distinctions.

One criterion frequently mentioned is whether the method is *interpolating* or *approximating*. A method is interpolating if the initial and all subsequently generated control points remain in the mesh as the subdivision process proceeds (consequently, the initial and all generated points are in the limit surface). Methods that are not interpolating are said to be approximating. In the following two tables, methods from the upper row of Figure 1.30$_{/33}$ are classified as either interpolating or approximating.

Interpolating methods
Modified Butterfly (Section 4.2.3)
Kobbelt method (and four-point method for curves) (Section 4.2.3)
Repeated Averaging with $d = 1$ (Linear Subdivision) (Section 1.3.1)

Approximating methods
Repeated Averaging with $d > 1$ (Section 1.3.1)
Doo–Sabin method (Section 1.3.1)
Catmull–Clark method (Section 1.3.1)
Loop method (Sections 1.3.1 and 3.5.2)
$\{\text{Midedge}\}^2$ (Section 3.7.2)
4-8 subdivision (Section 3.7.2)
The $\{\sqrt{3}\}^2$ method (Sections 4.2.1 and 4.2.3)

Note, however, that approximating methods can also be used to construct surfaces that interpolate the vertices (and normals) of a given control polyhedron [63, 106]. These methods have the advantage that it is not necessary to interpolate every vertex, and their limit surfaces have better curvature properties [115]. See Sections 7.1 and 7.2, and also see [74] on "variational subdivision." Further, there

are also approximating methods called *quasi-interpolation* methods [28]; see Section 6.4. Other interpolating methods, not shown in the table above, are described in [79, 87, 119].

Another criterion that has been used for the classification of methods is the *level of parametric continuity* of the limit surface, away from extraordinary points. On the other hand, other aspects of continuity of limit surfaces are also important, as discussed in Chapter 5.

Here is a summary according to the parametric-continuity classification.

Parametric continuity away from extraordinary points

{Midedge}^{2} method: C^1
Kobbelt method: C^1
Modified Butterfly: C^1
Doo–Sabin method: C^1
Catmull–Clark method: C^2
Repeated Averaging: C^{d-1} (bidegree $d \geq 1$)
Loop method: C^2
The $\{\sqrt{3}\}^2$ method: C^2
Stam's extension of Loop method: $C^{(2m-6)/3}$ (total order $m = 6, 9, 12, \ldots$)
4-8 subdivision: C^4

The *reproduction degree* [28, 67, 86] is another way to classify subdivision methods. The reproduction degree is the largest integer d such that the subdivision scheme reproduces polynomials up to degree d. This means that if the initial control-point data are obtained by uniform sampling of a polynomial of degree d, then the subdivision method converges to that polynomial. The sets of polynomials reproduced are referred to as *precision classes*; thus, a method that reproduces polynomials up to degree 1 is said to have linear precision, up to degree 2 quadratic precision, and up to degree 3 cubic precision.

The degree of polynomial reproduction, or precision, is studied in Section 6.4. To illustrate this method of classification, with a minor caveat all box-spline methods (Figure 1.30$_{/33}$, second column, lower row) have linear precision, and in particular, for $d \geq 1$ the $LR(d)$ algorithm for univariate uniform B-splines has linear precision. Further, the $LR(d \times d)$ algorithm has bilinear precision. The four-point scheme for curves has cubic precision, and the $4pt \times 4pt$ algorithm has bicubic precision.

Yet another criterion used to classify methods is the *type of mesh* on which they are used. Thus, the Loop method and the Modified Butterfly method are normally used with triangular meshes, while the Doo–Sabin, Catmull–Clark, and Kobbelt methods are used with quadrilateral meshes. This is not, however, a reliable way to classify methods. For example, the Repeated Averaging method and the Midedge method are applicable to general meshes, including both triangular and

quadrilateral meshes. Similarly, the Catmull–Clark method converts a triangular mesh into a quadrilateral mesh as part of its first step. Also, some methods explicitly permit the possibility of transition between triangular meshes and quadrilateral meshes [125], or explicitly use both triangular and quadrilateral faces [152], while 4-8 subdivision is based on a completely different tiling of the plane which in a certain sense combines the use of triangular and quadrilateral meshes [164]. Another example is $\sqrt{3}$-subdivision, which subdivides triangular grids [76], but which can also be viewed as a method that makes use of the hexagonal dual. (This is a useful point of view for comparison of the $\sqrt{3}$-subdivision method with box splines; see Section 4.2.2.)

On the other hand, it is possible, as we have seen, to give a rough classification of methods based on the *type of tiling* associated with their splitting schema.

Finally, as mentioned in Section 1.2.2, subdivision methods are sometimes classified as *primal* methods or *dual* methods. This terminology is also useful for rough classification, but there are difficulties associated with it. First, all box-spline subdivision methods (including pure primal methods such as Catmull–Clark) can be viewed as based on alternation between primal grids, dual grids, or semidual grids. Second, if we try to classify methods according to where this alternation terminates, we find that it may terminate in a semidual grid, so that even with this approach we must admit the possibility of methods of mixed type. (This is discussed, in particular, in Example $3.5.8_{/122}$.) Third, even primal methods such as $\sqrt{3}$-subdivision and 4–8 subdivision can be viewed as using the vertices of the dual of the unrefined mesh to conveniently break a single step of the method into two substeps, which is useful for the reduction of the incremental increase in memory requirements.

1.4 Subdivision matrices

There are several kinds of subdivision matrices used in the analysis of subdivision methods, but all varieties have the same general purpose: to specify, using a matrix-vector multiplication,[20] the control points at one step in the process, in terms of the control points at the previous step. When reading the literature, it must be determined which subdivision matrix is being used (often this is not explicitly stated).

One of the most useful kinds of subdivision matrices is the *local* subdivision matrix, which permits analysis of a subdivision surface in a neighbourhood of a single point. Even this matrix, however, comes in several varieties. One of the purposes of this section is to show the relationships among various subdivision matrices that appear in the literature, and it is most convenient to start with the global subdivision matrix.

1.4.1 The global subdivision matrix

We begin with a general form of matrix that specifies how the complete set of control points at one step is transformed into a complete set of control points at the next [92]. Suppose that the mesh is a locally planar mesh without boundary. Given

the L_ν control points $p^\nu = p^\nu_{(L_\nu \times N)}$ at step ν of the subdivision process, the subdivision matrix Σ^ν computes the control points $p^{\nu+1}$ as affine combinations[21] [137] of the control points of p^ν:

$$p^{\nu+1} = \Sigma^\nu p^\nu, \quad \nu = 0, 1, \ldots. \tag{1.13}$$

Since the number of control points increases at each step of the process, the matrix Σ^ν is *not* square: in $(1.13)_{/39}$, Σ^ν is $L_{\nu+1} \times L_\nu$. The constraint that the multiplication should produce *affine combinations* is guaranteed by a hypothesis that the sum of the elements in each row of Σ^ν sums to 1. That this hypothesis guarantees *affine invariance* of the process is easily shown: if $A_{(N\times N)}$ is a matrix, and $t_{(1\times N)} \in \mathbb{R}^N$ is a translation vector, let $\tau_{(L\times N)}$ be a matrix with L identical rows, each equal to t. Then

$$\Sigma^\nu \left(p^\nu A + \tau_{(L_\nu \times N)}\right) = (\Sigma^\nu p^\nu)A + \Sigma^\nu \tau_{(L_\nu \times N)} = p^{\nu+1}A + \tau_{(L_{\nu+1}\times N)};$$

i.e., an affine transformation $p_\ell := p_\ell \cdot A + t$ applied to each control point before a subdivision step will have the same effect as it would if applied after the subdivision step. This is true in particular if $N = 3$ and $A_{(3\times 3)}$ defines a rotation in $\mathbb{R}^3$, so that $p_\ell := p_\ell \cdot A + t$ corresponds to a rigid motion in $\mathbb{R}^3$, and also if A defines a scaling or shear [127, p. 83]. Affine invariance guarantees independence of the coordinate system used [156, p. 146].

In Section 5.1 we show, for methods based on general subdivision polynomials, that affine invariance is a necessary condition for convergence.

Remark 1.4.1. Affine invariance guarantees invariance with respect to translations and rotations. On the other hand, given a process of the form $(1.13)_{/39}$, invariance with respect to translation is sufficient to guarantee affine invariance.

In fact, if a translation $p_\ell := p_\ell + t$ of the control points, with a constant vector t, gives the same translation for the new control points obtained after the subdivision step, then the process is affine invariant. This follows because

$$\Sigma^\nu \left(p^\nu + \tau_{(L_\nu \times N)}\right) = \Sigma^\nu p^\nu + \tau_{(L_{\nu+1}\times N)}$$

implies that

$$\Sigma^\nu \tau_{(L_\nu \times N)} = \tau_{(L_{\nu+1}\times N)},$$

and (omitting the dimensions on the matrices τ)

$$\Sigma^\nu(p^\nu A + \tau) = \Sigma^\nu p^\nu A + \Sigma^\nu \tau = \Sigma^\nu p^\nu A + \tau = p^{\nu+1}A + \tau;$$

i.e., the process is affine invariant. Thus, for linear subdivision, invariance with respect to translation is equivalent to affine invariance. ∎

If the elements of Σ^ν are also nonnegative, then Σ^ν will produce *convex* combinations of the control points at the previous step, and the limiting surface, if it exists, will therefore lie in the convex hull of the original control points.

The condition that the subdivision matrix has row sums equal to one is satisfied for all practical subdivision methods, including the Doo–Sabin method (see $(1.4)_{/28}$), the Catmull–Clark method (see Section 1.4.3), and the Loop method (as indicated following $(1.12)_{/32}$).

The class of processes defined by $(1.13)_{/39}$ is very general, as is shown by the example of fractal generation given in Section 1.5. A process is completely defined by the splitting schema (which specifies how the logical mesh at step $\nu+1$ is obtained from the mesh at step ν), by the global subdivision matrices Σ^ν, and by the initial configuration of the mesh and p^0. Each Σ^ν is independent of the control points p^ν, and so the rules at each step depend only on the logical mesh M, and not on control-point positions. On the other hand, as observed in [176, Sec. 2.3.3], in principle there is no reason why different subdivision rules could not be used in different parts of the mesh, and, since Σ^ν may change with ν, the rules could change as the subdivision progresses. Certain such methods have been proposed [161, 162]. In practice, however, the subdivision processes considered[22] are almost always *stationary*, which means that the subdivision rules, defined in terms of the mesh, remain fixed and depend only on the local topology for all ν greater than or equal to some ν_0 [92, 172]. Thus, for example, the hybrid subdivision of [42] initially uses special rules for an arbitrary, but finite, number of subdivision steps, but subsequently uses a fixed set of rules. Such a method is stationary.

The subdivision matrix in $(1.13)_{/39}$ is unnecessarily cumbersome for many purposes. If, as is usually the case, a small number of rules suffices to define the process locally, then it is better, for purposes of analysis in the neighbourhood of a single point on the surface, to use a local subdivision matrix. We show in Section 1.4.3 how simpler subdivision matrices can be obtained.

1.4.2 Global subdivision matrices for B-spline functions

In order to find more convenient subdivision matrices for the description of surface subdivision, we temporarily leave the case of finite meshes and permit the control points to be defined on a countably infinite domain. At the same time, we leave the case of surfaces and temporarily restrict our attention to scalar functions defined on the real line, or curves having the real line as parametric domain. Thus, in place of the finite vector of control points $p^\nu_{(L_\nu \times N)}$ above, we have instead an infinite vector $p^\nu_{(\omega \times N)}$, where ω is the cardinality of the natural numbers and of $\mathbb{Z}$. We take the underlying grid to be $h\mathbb{Z}$, where h is the grid-size and $\mathbb{Z}$ is the set of (positive and nonpositive) integers. Consequently, the matrix Σ^ν is now doubly infinite in both dimensions. Also, we assume temporarily that $h = 1$.

At first glance, this may seem a surprising way to proceed. As we have just stated, moving to the infinite case has resulted in the replacement of a finite subdivision matrix $\Sigma^\nu_{(L_{\nu+1} \times L_\nu)}$ by a matrix that is doubly infinite in both dimensions. Worse still, the vectors $p^\nu_{(L_\nu \times N)}$ have been replaced by doubly infinite vectors $p^\nu_{(\omega \times N)}$, and it may not be immediately clear how this will conveniently generalize to the case of surfaces, where p^ν is doubly infinite in two dimensions.

It turns out that these worries are unfounded. Initial analysis of the regular case of subdivision methods is, usually, most easily carried out using an infinite

domain. Subsequently, it always transpires that the processes derived are *local* in nature, and this permits us to switch from global subdivision matrices to local subdivision matrices. The crucial fact here is that because of the locality, these local subdivision matrices are finite, even in the surface case.

Let us begin with a very simple case, a scalar B-spline function defined on the real line, with subdivision carried out on a one-dimensional mesh associated with the *dyadic numbers* $2^{-\nu}\mathbb{Z} = \{x : x = 2^{-\nu}y, y \in \mathbb{Z}\}$. In this case, the matrix Σ^ν can be viewed as a $2\omega \times \omega$ matrix. (Mathematically, the cardinality 2ω is the same as ω, but it is informative to write the matrix dimensions in this way, since the notation suggests a doubling of the number of grid points at each step.) It can be shown, for example, that the matrix defining a fourth-order univariate B-spline function is

$$\begin{array}{c} \\ \Sigma \;=\; \frac{1}{8} \end{array} \begin{array}{c} \begin{array}{cccc} 0 & 1 & & \end{array} \\ \begin{array}{cccc} \downarrow & \downarrow & & \end{array} \\ \left[\begin{array}{cccc} 0 & . & . & . \\ 1 & 0 & . & . \\ 4 & 0 & . & . \\ 6 & 1 & 0 & . \\ 4 & 4 & 0 & . \\ 1 & 6 & 1 & 0 \\ 0 & 4 & 4 & 0 \\ 0 & 1 & 6 & 1 \\ . & 0 & 4 & 4 \\ . & 0 & 1 & 6 \\ . & . & 0 & 4 \\ . & . & 0 & 1 \\ . & . & . & 0 \end{array}\right]_{(2\omega\times\omega)} \end{array} \begin{array}{cc} & \\ & \\ \leftarrow & -2 \\ \leftarrow & -1 \\ \leftarrow & 0 \\ & \\ & \\ & \\ & \\ & \\ & \\ & \\ & \\ & \\ & \end{array} \quad . \tag{1.14}$$

Because of the uniformity introduced by the infinite domain, there is no longer any need for the superscript ν on Σ: the matrix does not depend on the iteration number.

The subdivision matrix Σ is often denoted in the literature by S [168, p. 23], [176, p. 28]. It is, however, a global subdivision matrix, so in this book we instead continue to use the symbol Σ. (The local subdivision matrix introduced in Section 1.4.3 below is, confusingly, also denoted by S in the literature. Thus, our choice of Σ here has the additional advantage of avoiding a conflict of notation.)

If $p^\nu_{(\omega\times N)}$ is a one-dimensional array of control points associated with the grid $2^{-\nu}\mathbb{Z}$, then a step in the subdivision process that evaluates the B-spline corresponds to multiplying p^ν on the left by Σ. This is proved in Section 2.2.2, and illustrated for a set of scalar control points ($N = 1$) in Figure 1.31$_{/42}$. In this example the initial vector and the result of one multiplication by Σ are, respectively,

$$\begin{array}{c} \\ p^\nu_{(\omega\times 1)} = \end{array} \begin{array}{c} \begin{array}{cccccc} & & 0 & 1 & & \end{array} \\ \begin{array}{cccccc} & & \downarrow & \downarrow & & \end{array} \\ \left[\begin{array}{cccccc} \ldots & 0 & 1 & 3 & 2 & 0 \end{array}\right. \end{array} \begin{array}{c} \\ \\ \left.\begin{array}{c} \ldots \end{array}\right]^t \end{array}$$

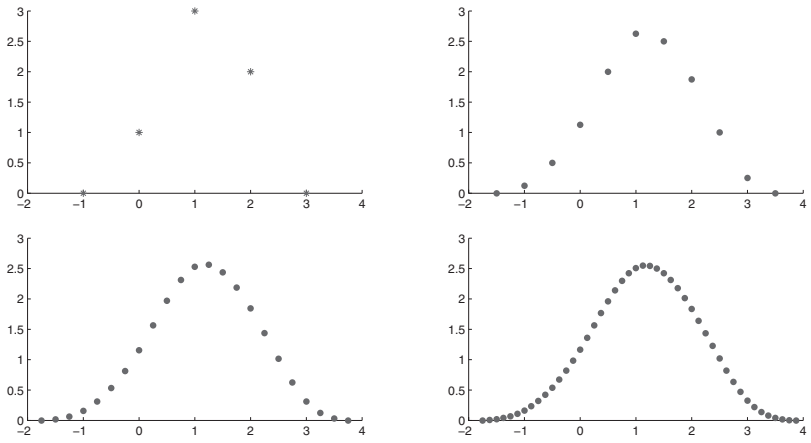


Figure 1.31. *The result of three successive multiplications by* Σ.

and

$$p^{\nu+1}_{(2\omega\times 1)} = \begin{bmatrix} \dots & 0 & \underset{\downarrow}{-2} & \underset{\downarrow}{-1} & \underset{\downarrow}{0} & \underset{\downarrow}{1} & \dots \\ & & \frac{1}{8} & \frac{1}{2} & \frac{9}{8} & 2 & \end{bmatrix}^t .$$

The row indexing in Σ is determined by the use of centered versions of the basis functions: if the index in the result $p^{\nu+1}_{(2\omega\times 1)}$ is divided by 2, we obtain the locations in the original parametric domain, as illustrated in Figure $1.31_{/42}$ (top right). The centered versions of the basis functions are introduced to avoid unwelcome shifting of the resulting spline; this is discussed in detail in Chapter 2. Two further multiplications by Σ are shown in the second row of Figure $1.31_{/42}$.

We have considered an infinite mesh here, because this leads naturally to useful local subdivision matrices for the surface case. We note in passing, however, that we can, in the scalar or curve case, write down the global subdivision matrix Σ^ν corresponding to a finite mesh; see Exercise $10_{/49}$. This is not, however, the most productive avenue to follow in the search for useful subdivision matrices.

1.4.3 Local subdivision matrices

The key to obtaining a concise representation of the information contained in the general matrix Σ^ν, both in the curve and surface cases, is to make use of the fact that in each row there is only a small number of nonzero elements, i.e., to make use of the local nature of the subdivision process.

We again consider, to begin with, the curve case and suppose that we are interested in the behaviour of the curve at a particular point. There is no loss in

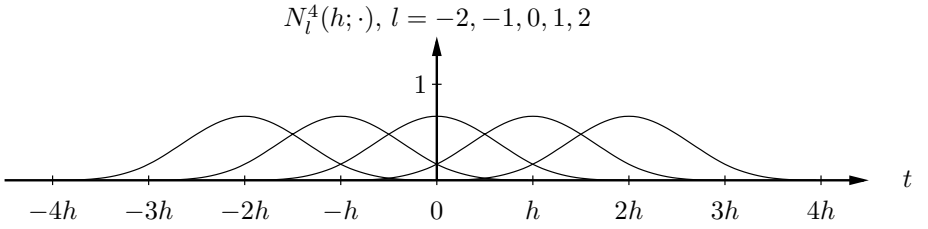


Figure 1.32. *The five order-4 B-spline basis functions nonzero on* $[-h, h]$.

generality in assuming that this point is the origin, $t = 0$ [176, p. 36]. Consider again the fourth-order B-spline curve, which has the form

$$x(t) = \sum_{l=-\infty}^{\infty} p_l N_l^4(h;t), \qquad 0 \le t \le h, \tag{1.15}$$

where the $N_l^4(h;\cdot)$ are centered cubic basis functions shifted by lh, as shown in Figure 1.32/43, and $p_l \in \mathbb{R}^N$. The control points that influence $x(t)$ on $[0, h]$ are p_{-1}, p_0, p_1, and p_2. Similarly, the control points p_{-2}, p_{-1}, p_0, and p_1 are those that influence $x(t)$ on $[-h, 0]$. Thus, to study $x(t)$ in a neighbourhood of the origin, it is sufficient to consider the 5×5 subblock of the bi-infinite matrix Σ, namely

$$\frac{1}{8}\begin{bmatrix} 1 & 6 & 1 & 0 & 0 \\ 0 & 4 & 4 & 0 & 0 \\ 0 & 1 & 6 & 1 & 0 \\ 0 & 0 & 4 & 4 & 0 \\ 0 & 0 & 1 & 6 & 1 \end{bmatrix}. \tag{1.16}$$

We call this matrix a *local subdivision matrix*.

On the other hand, in [156] the middle 3×3 submatrix of the matrix in (1.16)/43 is referred to as the "local subdivision matrix." The reason for this difference in terminology is explained in [176, p. 45], where it is remarked that the 5×5 matrix is needed for analysis, but only the 3×3 matrix is needed for computation of the exact value on the curve, corresponding to the origin (since there are only three nonzero basis functions at the origin); see Sections 2.5.5 and 6.1.1.

Now let us consider subdivision rules for surfaces. These rules too are usually local in nature, and sufficiently uniform that a single subdivision step can be described by a *square* matrix S of small dimension that maps a submesh of M^ν, the locally planar logical mesh at step ν, into a topologically equivalent submesh of the refined mesh $M^{\nu+1}$ at step $\nu + 1$ [76, p. 105].

A common situation is that a k-ring neighbourhood is mapped onto a k-ring neighbourhood in the refined mesh with an identical topological structure. Typically it is sufficient to use a matrix corresponding to $k = 1$ to compute exact

values or tangent vectors (see Section 6.1), but matrices corresponding to a larger number of rings are necessary for the analysis of convergence and smoothness (see Sections 5.5 and 5.6). A precise definition of k-ring neighbourhood is given in Section 5.5.1.

Each row of the matrix S is a rule to compute the control point for a vertex in $M^{\nu+1}$. Each column of the matrix specifies the contribution of the control point associated with a vertex in M^{ν} to the control point associated with the refined mesh $M^{\nu+1}$. Thus, we may relate the local subdivision matrix S to the matrix Σ^{ν}. Suppose that Σ^{ν} (whose dimension $L_{\nu+1} \times L_{\nu}$ varies with ν) defines a stationary process, and that consequently there is a fixed and finite number of rules that may apply at each step, for $\nu \geq \nu_0$. A local subdivision matrix S is a square matrix that defines these rules, and which can be obtained from any Σ^{ν}, $\nu \geq \nu_0$, by means of the following operations: deletion of a certain number of rows of Σ^{ν}, followed by deletion of a certain number of columns of Σ^{ν}, where deleted columns contain only zeroes, and followed possibly by a permutation of rows and columns.

It follows that if the sum of the elements of each row of Σ^{ν} is equal to 1, then, the sum of the elements of each row of S is equal to 1.

A typical example is the Catmull–Clark method, described in Section 1.3.1. After the first step in the $pQ4$ process, all faces in the mesh are quadrilateral, as illustrated in Figure 1.15$_{/18}$. For example, in Figure 1.15$_{/18}$ (centre), the four corner points represent the old vertices, and the five other points represent new vertices in the finer mesh (to be computed).

Suppose that all of the old vertices are ordinary, except possibly for one, denoted by V, which has valence n. It can be seen from Figure 1.27$_{/29}$ (where $n = 5$) that if the mesh is locally planar, there are n old vertices $E_0, \ldots, E_{n-1}$ on an *edge* adjacent to V in M^{ν}, and n old vertices $F_0, \ldots, F_{n-1}$ diagonally across a *face* from V in M^{ν}. Along with V itself, this makes a total of $2n + 1$ old vertices. On the other hand, there are also n newly created edge points $E'_0, \ldots, E'_{n-1}$ adjacent to V in $M^{\nu+1}$, and n newly created face points $F'_0, \ldots, F'_{n-1}$ in faces adjacent to V in $M^{\nu+1}$. Thus, the total number of new vertices, including the vertex V which must be modified, is $2n + 1$. Note that V and $E_i, E'_i, F_i, F'_i, i = 0, \ldots, n-1$, are elements of $\mathbb{R}^N$; they are most conveniently viewed as $(1 \times N)$ row vectors.

The Catmull–Clark rules specify how the $2n + 1$ control points at the new vertices are to be computed in terms of the $2n+1$ control points at the old vertices. In fact, one way to express the Catmull–Clark algorithm, after the first subdivision step, is

$$\begin{aligned} V' &= \frac{4n-7}{4n} V + \frac{3}{2n}\left(\frac{1}{n}\sum_{j=0}^{n-1} E_j\right) + \frac{1}{4n}\left(\frac{1}{n}\sum_{j=0}^{n-1} F_j\right), \\ E'_i &= \frac{3}{8}(V + E_i) + \frac{1}{16}(E_{i-1} + F_{i-1} + F_i + E_{i+1}), \\ & \qquad\qquad i = 0, \ldots, n-1. \\ F'_i &= \frac{1}{4}(V + E_i + F_i + E_{i+1}), \end{aligned} \tag{1.17}$$

Indices here are calculated modulo n. These equations define a $(2n+1)\times(2n+1)$ matrix S having row sums equal to 1:

$$\begin{bmatrix} F'_0 \\ . \\ . \\ . \\ F'_{n-1} \\ E'_0 \\ . \\ . \\ . \\ E'_{n-1} \\ V' \end{bmatrix} = \frac{1}{16} \begin{bmatrix} 4 & & & & & 4 & 4 & & & & 4 \\ & 4 & & & & & 4 & 4 & & & 4 \\ & & \dots & & & & & 4 & \dots & & . \\ & & & 4 & & & & & 4 & 4 & 4 \\ & & & & 4 & 4 & & & & 4 & 4 \\ 1 & & & & 1 & 6 & 1 & & & 1 & 6 \\ 1 & 1 & & & & 1 & 6 & 1 & & & 6 \\ & 1 & \dots & & & & 1 & 6 & \dots & & . \\ & & \dots & 1 & & & & 1 & \dots & 1 & 6 \\ & & & 1 & 1 & 1 & & & 1 & 6 & 6 \\ \frac{4}{n^2} & \frac{4}{n^2} & \dots & \frac{4}{n^2} & \frac{4}{n^2} & \frac{24}{n^2} & \frac{24}{n^2} & \dots & \frac{24}{n^2} & \frac{24}{n^2} & \frac{16n-28}{n} \end{bmatrix} \begin{bmatrix} F_0 \\ . \\ . \\ . \\ F_{n-1} \\ E_0 \\ . \\ . \\ . \\ E_{n-1} \\ V \end{bmatrix},$$

where S is the matrix multiplying $[F_0,\dots,F_{n-1},E_0,\dots,E_{n-1},V]^t$ on the right-hand side of the equation just shown.

The verification that $(1.17)_{/44}$ is the same as the two equivalent formulations of the Catmull–Clark method given in Section 1.3.1, is immediate. The expression for E'_i in $(1.17)_{/44}$ is identical to $(1.7)_{/29}$, and the expression for F'_i in $(1.17)_{/44}$ is the same as that in $(1.5)_{/28}$. Finally, the expression for V' in $(1.11)_{/31}$ is easily verified to be identical to the value of V' in $(1.17)_{/44}$; see Exercise $11_{/49}$ and Section A.1.

The k-ring neighbourhood involved in the subdivision matrix S discussed here has $k=1$: in Figure $1.27_{/29}$, only points from the first ring of vertices surrounding V are used in the definition of S. As already mentioned, larger local subdivision matrices are necessary for the study of smoothness (e.g., matrices of dimension $(12n+1)\times(12n+1)$, or $13n\times 13n$ [124, Example 5.14]). For this reason, the matrix S just introduced is later denoted by S_1, to distinguish it from the larger local subdivision matrices; see Sections 5.4 and 5.6. Similarly, other subdivision matrices are sometimes employed for other purposes. For example, the presentation of Stam's method [150] introduces such matrices (see Section 6.3.2).

Exercise $12_{/49}$ asks for an explicit subdivision matrix S for the Loop scheme, similar to the one defined by $(1.17)_{/44}$ for the Catmull–Clark method.

As mentioned in [124, p. 109], subdivision stencils such as those illustrated in Figure $1.29_{/32}$ are just representations of the rows of the local subdivision matrix.

1.5 Generating fractal-like objects

We conclude this chapter with an example that shows that very simple processes of the sort we have described may lead to fractal-like objects [12, 32, 156], such as continuous curves that are nondifferentiable on a dense set, i.e., nondifferentiable on a set that is dense in the parametric domain. This provides a context for our later study (Chapter 5) of conditions guaranteeing convergence of a subdivision scheme to a smooth limit surface: exactly what are the conditions necessary or sufficient for various levels of smoothness?

For comparison purposes, we begin by presenting one of the earliest and simplest subdivision algorithms invented. This is *Chaikin's algorithm*, which was

introduced in Section 1.1, and which produces continuously differentiable piecewise quadratic curves (see Chapters 3 and 5). Following this, we make a relatively minor change to the algorithm, which results in a new algorithm that generates nowhere-differentiable curves.

Chaikin's algorithm

Chaikin [26] introduced a method for generating a curve defined by the control points $p_l \in \mathbb{R}^N$, $l \in \mathbb{Z}$, with components that are piecewise quadratic functions. We view the points p_l as defined on a grid with resolution h: lh, $l \in \mathbb{Z}$.

Let $p_l^0 \doteq p_l$, $l \in \mathbb{Z}$. Then, subsequent sequences p_l^ν, $\nu \geq 1$, are produced from $p_l^{\nu-1}$ by the following substeps.

1. Linear subdivision:

$$\begin{aligned} \hat{p}_{2l}^\nu &= p_l^{\nu-1}, \\ \hat{p}_{2l+1}^\nu &= \frac{1}{2}\left(p_l^{\nu-1} + p_{l+1}^{\nu-1}\right), \end{aligned} \qquad l \in \mathbb{Z}. \tag{1.18}$$

   These values can be viewed as associated with a grid with resolution $h/2$: letting $i = 2l$ $\hat{p}_i$ takes values at $ih/2$, $i \in \mathbb{Z}$.

2. Averaging:

$$p_i^\nu = \frac{1}{2}\hat{p}_i^\nu + \frac{1}{2}\hat{p}_{i+1}^\nu, \qquad i \in \mathbb{Z}. \tag{1.19}$$

These equations can be rewritten as

$$\begin{aligned} p_{2i}^\nu &= \frac{1}{2}\, p_i^{\nu-1} + \frac{1}{2}\,\frac{1}{2}\left(p_i^{\nu-1} + p_{i+1}^{\nu-1}\right) = \frac{3}{4}p_i^{\nu-1} + \frac{1}{4}p_{i+1}^{\nu-1}, \\ p_{2i+1}^\nu &= \frac{1}{2}\,\frac{1}{2}\left(p_i^{\nu-1} + p_{i+1}^{\nu-1}\right) + \frac{1}{2}p_{i+1}^{\nu-1} = \frac{1}{4}p_i^{\nu-1} + \frac{3}{4}p_{i+1}^{\nu-1}. \end{aligned} \tag{1.20}$$

This process is called "corner cutting" [39, 40], a terminology motivated in Figure 1.33$_{/46}$, where the method is applied to a finite mesh rather than to an

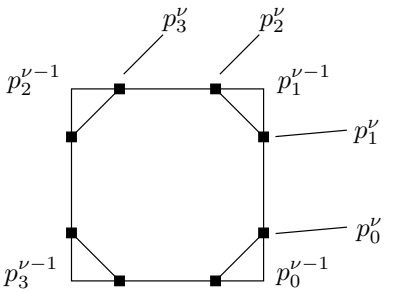


Figure 1.33. *Corner cutting.*

infinite sequence of control points. In the figure, the $p_\ell^{\nu-1}$, $\ell \in \{0,1,2,3\}$, are the four original corner points, and the square boxes show the positions of the new points p_ℓ^ν, $\ell \in \{0,1,\ldots,7\}$. The new points are a weighted average of the neighbouring $p_\ell^{\nu-1}$, with weight 1/4 applied to one neighbour and weight 3/4 applied to the other. Exercise $13_{/49}$ asks that the two steps in the process be expressed as a product of bi-infinite subdivision matrices.

The Chaikin process for each component can be shown to converge to a piecewise quadratic function defined by the initial control points: it is the $LR(2)$ algorithm for curves. The tensor product of the Chaikin method with itself produces, for $d = 2$, the biquadratic Lane–Riesenfeld algorithm $LR(2 \times 2)$ for surfaces.

Generation of a nonsmooth curve

It is shown in [40] that a modification of Chaikin's algorithm, obtained by changing the weights 3/4 and 1/4 in $(1.20)_{/46}$ to 3/5 and 1/5, respectively, produces in the limit a curve that fails to be differentiable on the parametric domain. In fact, the curve contains a dense set of points at which the left and right derivatives exist, but are unequal, and at which the left and right curvatures are both infinite. (Project $2_{/50}$ suggests an implementation permitting a visual comparison of Chaikin's method with a version of the method that uses the modified weights just mentioned.) A similar example, which occurs in the construction of wavelet bases, is shown in [156, Fig. 6.4]. Thus, nonsmooth functions can be produced using subdivision methods almost identical to the simplest methods commonly used in solid modelling. In the context of applications that require smooth surfaces, this provides motivation for finding conditions that guarantee various degrees of smoothness.

1.6 Additional comments

An overview of subdivision surfaces is given in [176], although this reference does not give a systematic mathematical treatment. Mathematically rigorous treatments at a high mathematical level are given in [25, 124] and in the thesis [172]. Further historical information is given in [52, Sec. 1.8]. Another overview of the subject is given in [140], along with [139] on the interrogation of subdivision surfaces. Special journal issues [113, 135] have been devoted to subdivision surfaces, and most of the papers in these issues are cited at various places in this book. An important related document is a series of tutorials on geometric modelling using polygonal meshes [20].

Also, the CGAL library is of interest; see www.cgal.org.

Chapters on subdivision surfaces appear in [30, 158]. Other books on the topic are [141, 168]; the paper [167] is a tutorial. A recent survey paper is [95].

1.7 Exercises

1. Give an example of a polyhedral mesh in $\mathbb{R}^3$ that is a locally planar mesh with boundary, has (geometrically) planar faces, and that has no self-intersections

in $\mathbb{R}^3$, but for which any extension to a locally planar mesh without boundary will introduce a self-intersection.

2. Consider the logical mesh shown in Figure $1.7_{/11}$ (left), with faces as indicated below. State whether the resulting logical mesh is locally planar, and whether it is a mesh with boundary or a mesh without boundary.
   (a) The mesh M defined by the mesh faces $(1, 2, 6, 5)$, $(2, 3, 7, 6)$, $(4, 7, 3, 0)$, and $(0, 1, 5, 4)$.
   (b) The mesh obtained by adding the face $(4, 5, 6, 7)$ to M.
   (c) The mesh obtained by adding the faces $(4, 5, 6, 7)$ and $(3, 2, 1, 0)$ to M.
   (d) The mesh M' obtained by adding the edges $(4, 6)$ and $(0, 2)$, and the faces $(4, 5, 6)$, $(4, 6, 7)$, $(0, 2, 1)$, and $(0, 3, 2)$, to M.
   (e) The mesh obtained by adding the face $(0, 2, 6, 4)$ to M'.

3. Show that for a locally planar mesh without boundary, the dual of the dual is the original primal mesh.

4. The Euler–Poincaré formula for a mesh that is a topological sphere asserts that $F - E + V = 2$, where F is the number of faces, E is the number of edges, and V is the number of vertices in the mesh. (For example, the cube has $F = 6$, $E = 12$, and $V = 8$, and $6 - 12 + 8 = 2$; the tetrahedron has $F = 4$, $E = 6$, and $V = 4$, and $4 - 6 + 4 = 2$.) Use this formula to show that such a mesh can be neither a regular triangular mesh nor a regular quadrilateral mesh.

5. Show that the Repeated Averaging method with $d = 2$ is equivalent to the Doo–Sabin method with the weights W_{ij} of Section 1.3.1 replaced by

$$W_{ij}^R = \begin{cases} \frac{2e+1}{4e}, & j = i, \\ \frac{e+2}{8e}, & j = i \pm 1, \\ \frac{1}{4e}, & \text{otherwise.} \end{cases} \tag{1.21}$$

   Show also that for each i, $\sum_j W_{ij}^R = 1$. (This alternate choice of weights is mentioned by Catmull and Clark [24, p. 354], and is referred to in [177, eq. (1)] as the Catmull–Clark variant of Doo–Sabin subdivision.)

6. Verify that $W_{ij} > 0$, where W_{ij} is defined as in $(1.3)_{/27}$. Also, verify $(1.4)_{/28}$ by showing that $\sum_{k=1}^{e-1} \cos(2\pi k/e) = -1$.

7. Show that in the Repeated Averaging method with $d = 3$, the two averaging steps following linear subdivision leave the face-point values F_i^L produced by linear subdivision unchanged, produce the values in $(1.7)_{/29}$ for new edge-point values, and produce the value V^{RA} in $(1.8)_{/29}$ as new value for an existing vertex.

   Show also that the Repeated Averaging method with $d = 3$ corresponds to using a smoothing stencil with α_n^* replaced by $1/4$, β_n^* replaced by $1/(2n)$, and γ_n^* replaced by $1/(4n)$.

8. Show that $(1.10)_{/30}$ is equivalent to $(1.9)_{/30}$.

9. Write pseudocode for the two procedures

$$\mathit{LSS\_Odd}(\mathcal{M}^0, d, \lambda) \text{ and } \mathit{LSS\_Even}(\mathcal{M}^0, d, \lambda),$$

to replace the procedure $\mathit{RepeatedAveraging}(\mathcal{M}^0, d, \lambda)$ for $d \geq 1$, d odd, $\lambda \geq 0$, and $d \geq 2$, d even, $\lambda \geq 0$, respectively. The procedure $\mathit{Dual}(\cdot)$ should be invoked only once in each subdivision step in *LSS_Even*, and never in *LSS_Odd*. All other invocations of *Dual* should be replaced pairwise by a procedure *Smooth* satisfying $\mathit{Smooth}(\cdot) \equiv \mathit{Dual}(\mathit{Dual}(\cdot))$ for all quadrilateral meshes (perhaps containing extraordinary vertices).

The procedure *Smooth* is understood to be applied in Jacobi manner. Give the stencil that defines its function.

10. Consider a finite circular loop of edges containing 9 edges and vertices. We can apply univariate fourth-order B-spline subdivision to this finite mesh, and after the first step, the circular loop will contain 18 edges and vertices. Write down the global subdivision matrix Σ^ν corresponding to this single subdivision step, $\nu = 1$.

11. Prove that the value of V' given in $(1.17)_{/44}$ is the same as the value given by the two formulations in Section 1.3.1.

12. Consider the effect of one step of the Loop subdivision method, in the neighbourhood of an extraordinary vertex V (i.e., a vertex with valence $n \neq 6$). The control point V is replaced by V', and the value E_i associated with each adjacent vertex is replaced by a new value E_i' corresponding to a new edge point, $i = 0, \ldots, n-1$. Draw the diagram for the Loop method corresponding to Figure $1.27_{/29}$ (case $n = 5$), and write down the $(n+1) \times (n+1)$ local subdivision matrix S corresponding to the one defined for the Catmull–Clark method by $(1.17)_{/44}$. (The k-ring neighbourhood for S again has $k = 1$.)

13. Write the process corresponding to $(1.20)_{/46}$ as the product of two bi-infinite subdivision matrices, one corresponding to the linear subdivision step $(1.18)_{/46}$ and the other corresponding to the averaging step $(1.19)_{/46}$.

1.8 Projects

1. *Generalization of local planarity.*

This is a mathematical project, which requires no implementation.

The definition of local planarity was given in terms of meshes designed for surface subdivision. Analogous definitions in the univariate case, for subdivision in the univariate (function or curve) case, are straightforward. It is also relevant to give such definitions in the case of higher dimensions (see, for example, [7, 97], where subdivision is extended to the three-dimensional case). Give a definition of "locally N-manifold" that is general enough to include

the case of curve, surface, and cellular decomposition. Investigate how this definition fits within the theory of geometric complexes, which are relevant in the context of solid modelling, and homology theory. (An informal and practically oriented description is given in [10] and [11, Chap. 12, 14].)

2. *Modifying the weights of Chaikin's method.*

   Implement Chaikin's method for the case of a univariate function, as well as a modified version of Chaikin's method that uses the weights 3/5 and 2/5 instead of the weights 3/4 and 1/4. Output a graph of the results of the two methods when subdivision is applied to a depth of 10, using the following (l, p_l^0) data:

   $$(0, 0.52), (1, 0.7), (2, 0.3), (3, 0.5), (4, 0.45),$$
   $$(5, 0.55), (6, 0.52), (7, 0.2), (8, 0.8), (9, 0.3), (10, 0.2).$$

   (This data is similar to that in [156, Fig. 6.4].) Visually compare the smoothness of the two results by comparing graphs of the two functions.

Chapter 2

B-Spline Surfaces

With this chapter we begin the more formal mathematical development, starting with a presentation of B-spline surfaces. Our goal is to present the part of the classical B-spline theory needed to formulate subdivision-surface methods. The presentation is more specialised than the usual developments of the classical spline theory [30, 51, 127]: we omit discussion of nonuniform B-splines, and also of the more general Non-Uniform Rational B-splines (NURBS), since these are not needed for the formal description of subdivision-surface methods.[23] In particular, we do not discuss knot-insertion algorithms [30, 36, 51, 127]. Such algorithms are important in the context of the general spline theory, but they are not necessary for our purposes.[24]

We begin, after introducing certain mathematical preliminaries, by presenting definitions and recursion formulas for scalar univariate uniform B-spline functions, and uniform B-spline curves. It may be surprising at first that so much attention is devoted to the univariate case, but the theory developed in this context is almost immediately applicable to tensor-product B-spline surfaces. A result that appears fairly early in the presentation is Proposition $2.2.6_{/64}$, which is the foundation of the Lane–Riesenfeld algorithm [81]. This is perhaps the most important result required for the understanding of the subdivision methods used in computer graphics and solid modelling: it is important to understand its proof, and the subsequent dissection of what is essentially the univariate Lane–Riesenfeld algorithm $LR(d)$. It is Proposition $2.2.6_{/64}$, in combination with Proposition $2.2.3_{/60}$, that shows what is really happening during the execution of a typical binary subdivision method. Furthermore, similar mechanisms come into play for box splines, and even in methods that are not based on box splines, such as $\sqrt{3}$-subdivision [118].

The nodal functions mentioned in Chapter 1 will be referred to as *basis functions* in the B-spline case, since they do in fact form a basis of a linear function space.

The presentation of uniform B-splines given here is unusual, in that it is based on centered basis functions, and corresponding centered versions of the subdivision polynomial, introduced below. The reason for using centered representations is

that they better reflect the essential symmetry of the subdivision processes used in practice. For uniform B-spline curves and surfaces, where the subdivision rule to be applied is the same for all vertices, the method of indexing basis functions is not of fundamental importance, although even in this case the centered version is more natural, as shown in the discussion following $(1)_{/330}$ in note 31 of this chapter. In the case of a mesh that contains extraordinary vertices, however, the subdivision rule to be applied depends on the nature of the particular vertex, and the control point corresponding to this vertex is in a natural way associated with a basis function centered around the vertex and indexed in the same way as the vertex. But doing this for an extraordinary vertex forces us to do the same for the ordinary vertices, i.e., to associate the vertex index with a centered basis function. Further, using centered representations is consistent with what is done for non-box-spline methods, such as the Butterfly, Kobbelt, and $\sqrt{3}$ methods introduced in later chapters.

Following the discussion of uniform B-spline functions and curves, the case of tensor-product surfaces is described. We then discuss application of the methods to finite meshes. Finally, further fundamental results are given for univariate B-splines.

This chapter corresponds to the first column in Figure $1.30_{/33}$. Variants of the tensor-product Lane–Riesenfeld algorithm $LR(d \times d)$, corresponding to the upper row in the first column of Figure $1.30_{/33}$, were described in Section 1.3.1.

2.1 Mathematical preliminaries

We begin by summarizing certain mathematical tools used later in this section and elsewhere in the book.

A convenient technique for the study of sequences of numbers $s_0, s_1, s_2, \ldots$ is the method of *generating functions*. An exposition from the point of view of applications in computer science[25] is given in [72, Sec. 1.2]. Generating functions are widely used in the study of subdivision processes, and we use a generalized version of them extensively.

Given the sequence $s_0, s_1, s_2, \ldots$, the associated generating function is the infinite series

$$G(z) = s_0 + s_1 \cdot z + s_2 \cdot z^2 + \cdots .$$

As observed in [72, p. 82], the advantage of introducing this series is that it represents the entire sequence at once, and if $G(z)$ turns out to be a known function (say, one for which we know the power series expansion), then its coefficients can be found.[26]

Some of the basic properties of generating functions [72, Sec. 1.2.9] are listed below. They follow immediately from the definition of the generating function.

- *Linear transformation.* If $G_a(z)$ is the generating function for $a_0, a_1, a_2, \ldots$, and $G_b(z)$ is the generating function for $b_0, b_1, b_2, \ldots$, then $\alpha G_a(z) + \beta G_b(z)$ is the generating function for the sequence $\alpha a_0 + \beta b_0, \alpha a_1 + \beta b_1, \alpha a_2 + \beta b_2, \ldots$. This can be written as

$$\alpha \sum_{k \geq 0} a_k z^k + \beta \sum_{k \geq 0} b_k z^k = \sum_{k \geq 0} (\alpha a_k + \beta b_k) z^k.$$

- *Shifting.* If $G(z)$ is the generating function for $a_0, a_1, a_2, \ldots$, then $z^i G(z)$ is the generating function for $0, \ldots, 0, a_0, a_1, a_2, \ldots$:

$$z^i \sum_{k \geq 0} a_k z^k = \sum_{k \geq i} a_{k-i} z^k.$$

- *Multiplication.* If $G_a(z)$ is the generating function for $a_0, a_1, a_2, \ldots$, and $G_b(z)$ is the generating function for $b_0, b_1, b_2, \ldots$, then $G_a(z)G_b(z)$ is[27] the generating function for the sequence $\sigma_0, \sigma_1, \ldots$, where

$$\sigma_i = \sum_{0 \leq k \leq i} a_k b_{i-k}.$$

  This last sum is the *discrete convolution* of the two sequences; we also introduce below the convolution of two functions (as opposed to sequences), defined in terms of integrals (as opposed to summation).

There are also properties related to the change of scale $G(cz)$, and to the differentiation and integration of generating functions [72, Sec. 1.2.9].

It is sometimes convenient to use various generalizations of the generating function just introduced, such as versions with negative and noninteger powers of z. One such generalization will be given presently. These more general generating functions are called *generalized polynomials*.

Given a function $f = f(t)$, let the sequence s be defined by

$$s_k = f(t + kh/2), \quad k \in \mathbb{Z},$$

and define the translated function

$$f_{kh/2}(t) = f(t - kh/2), \quad k \in \mathbb{Z}.$$

Then, the doubly infinite generating function for the sequence s specified by f, h, and t is defined as

$$G_f(z) = \sum_{k \in \mathbb{Z}} s_k z^k,$$

and we have

$$\begin{aligned} z^i G_f(z) &= \sum_k f(t + kh/2) \cdot z^{k+i} \\ &= \sum_k f(t + (k-i)h/2) \cdot z^k \\ &= \sum_k f_{ih/2}(t + kh/2) \cdot z^k \\ &= G_{f_{ih/2}}(z), \end{aligned}$$

where $\sum_k = \sum_{k \in \mathbb{Z}}$. Thus, multiplying a certain generating function by z^i can be made to encapsulate the idea of operating on functions by translation.

A translation operator

In the derivation given below, we proceed in a slightly different way, introducing a translation operator directly. It turns out to be convenient, in the context of subdivision, to work with units of $h/2$ rather than units of h, and consequently we introduce the operator z^a which operates on functions by translation in units of $h/2$:

$$z^a f = f_{ah/2}, \tag{2.1}$$

where

$$f_{ah/2}(t) = f(t - ah/2).$$

Next, for a generalized polynomial $p(z) = \sum_k p_k z^k$, we define $p(z)f = \sum_k p_k(z^k f)$, i.e.,

$$p(z)f(t) = \sum_k p_k f(t - kh/2). \tag{2.2}$$

Further, since the operations of taking linear combinations and making translations commute, we have $p(z)(q(z)f) = (p(z)q(z))f$ for all generalized polynomials $p(z)$ and $q(z)$.

As examples, we have

$$z^{-1}f = f(t + h/2),$$

$$z^{1/2}f = f(t - h/4),$$

$$z^{-1/2}f = f(t + h/4).$$

This level of generality is sufficient to permit convenient description of B-spline subdivision.

Convolution of functions

The *convolution* $f \otimes g$ of two functions $f = f(t)$ and $g = g(t)$ is a function of t defined by

$$(f \otimes g)(t) = \int_{-\infty}^{\infty} f(s)g(t - s)ds. \tag{2.3}$$

We can think of $f \otimes g$ as a smoothing of g, using f as a smoothing function: $g(t)$ is replaced by a weighted combination of its neighbouring values $g(t - s)$, where the weight used for each value of s is $f(s)$. A simple change in variable shows that smoothing the function f, using g as a smoothing function, gives the same result:

$$(f \otimes g)(t) = (g \otimes f)(t), \tag{2.4}$$

i.e., convolution is commutative.

From the definition of the translation operator in $(2.1)_{/54}$, or from $(2.2)_{/54}$, it follows that $z^a(z^b f) = z^{a+b}f$, and from $(2.3)_{/54}$ it follows that

$$(z^a f) \otimes (z^b g) = z^{a+b}(f \otimes g). \tag{2.5}$$

Further, if $p(z)$ and $q(z)$ are generalized polynomials, we have from $(2.2)_{/54}$ and $(2.5)_{/54}$ that

$$(p(z)f) \otimes (q(z)g) = (p(z)q(z))(f \otimes g). \tag{2.6}$$

To show this, observe that if $p(z) = \sum_k p_k z^k$ and $q(z) = \sum_l q_l z^l$, then the right-hand side of $(2.6)_{/55}$ is equal to

$$\begin{aligned}\sum_{k,l} p_k q_l z^{k+l}(f \otimes g) &= \sum_{k,l} p_k q_l (z^k f) \otimes (z^l g) \quad (\text{from } (2.5)_{/54}) \\ &= \sum_k p_k (z^k f) \otimes \sum_l q_l (z^l g) \ (\text{from } (2.3)_{/54}) \\ &= (p(z)f) \otimes (q(z)g).\end{aligned}$$

2.2 Univariate uniform B-spline functions

We use a standard approach, based on convolution, to derive uniform B-spline basis functions. These functions are denoted $N_l^m(h;t)$. Such basis functions were used, for example, in the fourth-order B-spline functions that defined the components of the curve $(1.15)_{/43}$.

Following the derivation of the B-spline basis functions, we exhibit certain recursion formulas that they satisfy. These recursion formulas are then extended to recursion formulas for the control points p_l of the B-spline function $\sum_{l\in\mathbb{Z}} p_l N_l^m(h;t)$. Propositions $2.2.3_{/60}$ and $2.2.6_{/64}$, which contain the results just mentioned, are of fundamental importance for the understanding of subdivision methods.

We restrict our attention for now to the univariate case. If the p_l are scalars, then the function $\sum_{l\in\mathbb{Z}} p_l N_l^m(h;t)$ is a scalar univariate function, whereas if $p_l \in \mathbb{R}^N$, it defines a curve in $\mathbb{R}^N$.

2.2.1 Definition of B-spline basis functions using convolution

The following derivation is based on repeated convolution of the pulse function

$$\overset{*}{N}{}^1(h;t) = \begin{cases} 1 & \text{if } 0 < t < h, \\ 1/2 & \text{if } t = 0 \text{ or } t = h, \\ 0 & \text{otherwise} \end{cases} \tag{2.7}$$

with itself, or repeated convolution of a centered version of this function with itself. The centered version of the pulse function is defined by

$$N^1(h;t) = \begin{cases} 1 & \text{if } |t| < h/2, \\ 1/2 & \text{if } |t| = h/2, \\ 0 & \text{otherwise.} \end{cases} \tag{2.8}$$

Centered functions are used as much as possible throughout the book, but on occasion, analyses turn out to be more easily expressed using uncentered versions, and we therefore give the basic definitions in both cases.

From $(2.3)_{/54}$, for any f we have

$$(f \otimes N^1(h;\cdot))(t) = \int_{t-h/2}^{t+h/2} f(s)ds.$$

If we convolve repeatedly by $N^1(h;\cdot)$, then the degree of continuity increases by one at each step: if $f \in C^k$, then

$$(f \otimes N^1(h;\cdot))'(t) = f(t+h/2) - f(t-h/2),$$

which implies $(f \otimes N^1(h;\cdot))' \in C^k$ and $f \otimes N^1(h;\cdot) \in C^{k+1}$. Clearly, a similar statement holds for repeated convolution by $\overset{*}{N}{}^1(h;\cdot)$.

The basis functions $\overset{*}{N}{}^m(h;\cdot)$ are defined inductively, using $(2.7)_{/55}$:

$$\overset{*}{N}{}^k(h;t) = \frac{1}{h}(\overset{*}{N}{}^{k-1}(h;\cdot) \otimes \overset{*}{N}{}^1(h;\cdot))(t), \quad k = 2,\dots,m. \tag{2.9}$$

The centered B-spline basis functions $N^m(h;\cdot)$ are defined similarly, using $(2.8)_{/55}$:

$$N^k(h;t) = \frac{1}{h}(N^{k-1}(h;\cdot) \otimes N^1(h;\cdot))(t), \quad k = 2,\dots,m. \tag{2.10}$$

Consequently, we have

$$\overset{*}{N}{}^m(h;\cdot) = \otimes_{k=1}^m \overset{*}{N}{}^1(h;\cdot)\frac{1}{h^{m-1}}, \ \ m \geq 1,$$

and

$$N^m(h;\cdot) = \otimes_{k=1}^m N^1(h;\cdot)\frac{1}{h^{m-1}}, \ \ m \geq 1. \tag{2.11}$$

To prove $(2.11)_{/56}$, note that from $(2.7)_{/55}$ and $(2.8)_{/55}$ it is clear that $zN^1(h;\cdot) = \overset{*}{N}{}^1(h;\cdot)$. By $(2.5)_{/54}$ and $(2.6)_{/55}$ we conclude that $z^m N(h;\cdot) = \overset{*}{N}{}^m(h;\cdot)$, i.e., that $N^m(h;t-mh/2) = \overset{*}{N}{}^m(h;t)$.

We also introduce shifted versions in both the uncentered and centered cases:

$$\overset{*}{N}{}^m_l(h;t) = \overset{*}{N}{}^m(h;t-lh) = z^{2l}\overset{*}{N}{}^m(h;t) \tag{2.12}$$

and

$$N^m_l(h;t) = N^m(h;t-lh) = z^{2l}N^m(h;t). \tag{2.13}$$

With this the basis functions in the example of $(1.15)_{/43}$ are completely defined.

Remark 2.2.1. The dependence of the nodal functions on the parameter h is given by

$$\overset{*}{N}{}^m(h;t) = \overset{*}{N}{}^m(1;t/h) \text{ and } N^m(h;t) = N^m(1;t/h). \tag{2.14}$$

We give the proof only for the case of centered functions. The result is obvious for $m = 1$. Proceeding by induction, if we assume that, for $k-1 \geq 2$, $N^{k-1}(h;t) =$

$N^{k-1}(1;t/h)$, then it follows by $(2.10)_{/56}$ that

$$\begin{aligned} N^k(1;t/h) &= \int_{\mathbb{R}} N^1(1;t/h-s)N^{k-1}(1;s)\,ds = \{r = sh\} \\ &= \int_{\mathbb{R}} N^1(1;(t-r)/h)N^{k-1}(1;r/h)\,dr/h \\ &= \int_{\mathbb{R}} N^1(h;t-r)N^{k-1}(h;r)\,dr/h = N^k(h;t), \end{aligned}$$

and the proof is complete. ∎

The function $N_l^m(h;\cdot)$ has support[28] $[lh - mh/2, lh + mh/2]$. We note the following series of particular cases, culminating with the example of the basis functions involved in the fourth-order B-spline just mentioned. The function $N_l^1(h;\cdot)$ is a pulse function with support $[(l-1/2)h, (l+1/2)h]$, and it is piecewise constant. The function defined by

$$\begin{aligned} N^2(h;t) &= \frac{1}{h}(N^1(h;\cdot) \otimes N^1(h;\cdot))(t) \\ &= \frac{1}{h}\int_{t-h/2}^{t+h/2} N^1(h;s)ds \end{aligned}$$

is a "hat function" with support $[-h, h]$, and $N_l^2(h;t)$ is also a hat function with support $[(l-1)h, (l+1)h]$: these functions are in C^0. The functions $N_l^1(h;\cdot)$ and $N_l^2(h;\cdot)$ are graphed in Figure $2.1_{/57}$.

Repeating this process, we obtain a function with one further degree of continuity at each step, and which, piecewise, is a polynomial of one higher degree at each step. Consequently, after one more step, the basis function $N_l^3(h;\cdot)$ of order 3 is a piecewise quadratic function in continuity class C^1 (see Figure $2.2_{/58}$, top), with support $[(l-3/2)h, (l+3/2)h]$. After yet another step, the basis function $N_l^4(h;\cdot)$

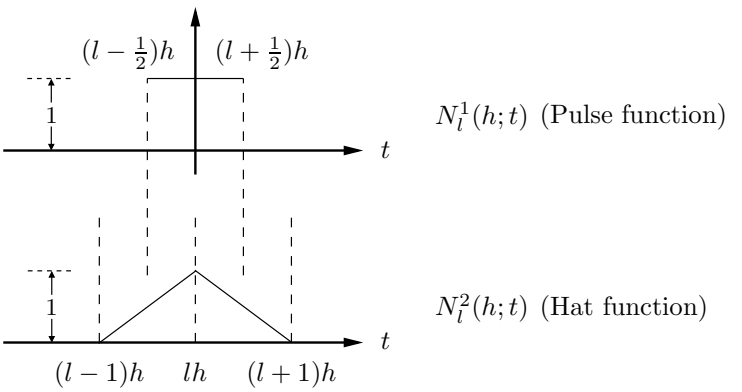


Figure 2.1. *Order-*1 *and order-*2 *centered B-spline basis functions.*

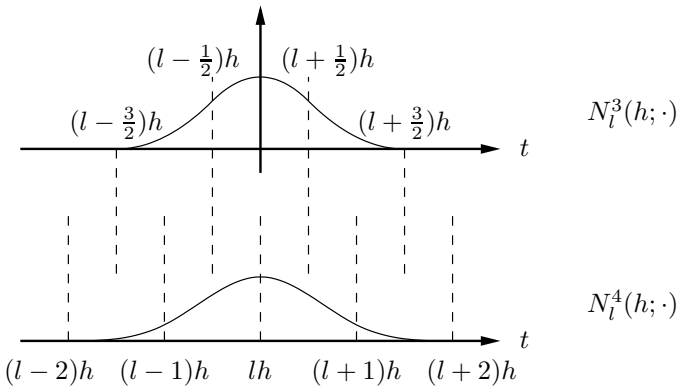


Figure 2.2. *Order-3 and order-4 centered B-spline basis functions.*

of order 4 is a piecewise cubic function in continuity class C^2, supported on the interval $[(l-2)h, (l+2)h]$ (see Figure 1.32$_{/43}$; see also Figure 2.2$_{/58}$, bottom).

The *nodes* of the centered basis function $N^m(h;\cdot)$ are the values of t at which the derivative of order $m-1$ has a jump discontinuity. Referring to Figures 2.1$_{/57}$ and 2.2$_{/58}$, for even values of m the nodes are in $h\mathbb{Z} = \{lh : l \in \mathbb{Z}\}$, while for odd values of m the nodes are in $h\mathbb{Z} + h/2$. The node positions are indicated in the figures by vertical dashed lines. We may summarize by saying that the nodes of $N_l^m(h;t)$ are in $h\mathbb{Z} + (m-2)h/2$. Similarly, the nodes of $N_l^m(h/2;t)$ are in $(h/2)\mathbb{Z} + (m-2)h/4$. See Exercise 1$_{/91}$.

2.2.2 Recursion formulas for uniform B-splines

We first derive recursion formulas for the univariate B-spline basis functions, and then recursion formulas for the B-spline function defined in terms of control points. These lead to a subdivision algorithm (the Lane–Riesenfeld algorithm) that applies to B-spline curves and surfaces. We have already seen, in Section 1.3.1, that this algorithm can be extended to meshes of general topological form; see also Section 2.4.

To emphasize the applications, the exposition in the univariate case is phrased in terms of curves in $\mathbb{R}^N$, rather than a single univariate function. Each component of the curve is a scalar univariate function. More specifically, given a curve represented using uniform B-spline basis functions as

$$x(t) = \sum_{l\in\mathbb{Z}} p_l N_l^m(h;t), \tag{2.15}$$

where the $p_l \in \mathbb{R}^N$ are the control points of the curve, and where $N_l^m(h;t)$ is defined by (2.13)$_{/56}$, we show how to find its representation in a refined grid with resolution $h/2$. Repeating this procedure indefinitely, we get a sequence of points converging to the curve.

Remark 2.2.2. The control points $\{p_l\}_{l\in\mathbb{Z}}$ in representation $(2.15)_{/58}$ are uniquely defined. Equivalently we say that the functions $\{N_l^m(h;\cdot)\}$ are linearly independent, i.e.,

$$\sum_{l\in\mathbb{Z}} p_l N_l^m(h;t) = 0$$

for all t implies that $p_l = 0$ for all l, as proved in Section 2.5.3. Because of this, the nodal functions $N_l^m(h;\cdot)$ are also referred to as basis functions. ∎

Recursion formulas for uniform B-spline basis functions

We first derive recursion formulas for the B-spline basis functions associated with the two one-dimensional grids of resolution h and $h/2$, respectively.

We introduce here a matrix first mentioned in $(1.14)_{/41}$, where it was denoted Σ. This matrix has the following remarkable property: it succinctly describes both the recursion $(2.17)_{/60}$ for uniform B-spline basis functions, and the recursion $(2.28)_{/64}$ for the control points of a B-spline curve, both derived below. The matrix Σ is defined by the requirement that its element in the position (k,l) is equal to s_{k-2l}, where

$$s_i = \frac{1}{2^{m-1}}\binom{m}{i+m/2}, \qquad \begin{array}{ll} i\in\mathbb{Z}, & m \text{ even},\\ i\in\mathbb{Z}+1/2, & m \text{ odd}.\end{array}$$

Then, Σ has the form

$$\begin{array}{rc} & l \\ & \downarrow \\ k = 2l - m/2 & \rightarrow \\ & \\ & \\ & \\ & \\ & \\ & \\ k = 2l & \rightarrow \\ & \\ & \\ & \\ k = 2l + m/2 & \rightarrow \\ & \\ & \\ & \\ & \end{array} \begin{bmatrix} 0 & 0 & . \\ s_{-m/2} & 0 & 0 \\ s_{-m/2+1} & 0 & 0 \\ . & s_{-m/2} & 0 \\ . & s_{-m/2+1} & 0 \\ . & . & s_{-m/2} \\ . & . & s_{-m/2+1} \\ . & s_0 & . \\ s_{m/2-1} & . & \\ s_{m/2} & . & . \\ 0 & s_{m/2-1} & . \\ 0 & s_{m/2} & . \\ 0 & 0 & s_{m/2-1} \\ 0 & 0 & s_{m/2} \\ . & 0 & 0 \end{bmatrix}_{(2\omega\times\omega)} . \tag{2.16}$$

We will show that the coefficients s_i relate the *centered* basis functions at two different levels of subdivision.

Each column of Σ is shifted by two positions relative to an adjacent column. Also, it should be observed that by definition, $s_i = 0$ if $|i| > \frac{m}{2}$, and that, apart

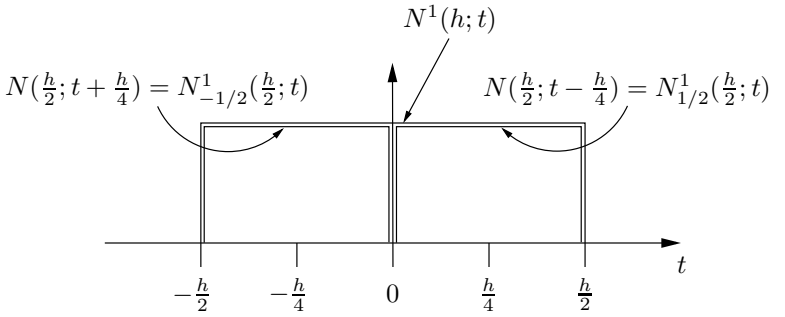


Figure 2.3. *Order-*1 *basis functions: The* 2*-scale relation.*

from the factor $\frac{1}{2^{m-1}}$, the coefficients s_i are just the binomial coefficients (a column of Σ corresponds to a row in Pascal's triangle). The columns of Σ are indexed by $l \in \mathbb{Z}$, but the rows of Σ are indexed by half-integer indices $k \in \mathbb{Z} + \frac{1}{2}$ when m is odd. This is unorthodox, but it presents no difficulty[29] since we always have $i + m/2 \in \mathbb{Z}$, and s_i is therefore well defined. Note that s_i depends on m, $m \geq 1$, and the element in row k of column l is equal to s_i if $k = i + 2l$.

Let

$$\mathcal{N}_h = [\quad \ldots \quad N_l^m(h;t) \quad \ldots \quad]_{(1\times\omega)}$$

be the row vector of basis functions on the grid with resolution h, and let

$$\mathcal{N}_{h/2} = [\quad \ldots \quad N_k^m(h/2;t) \quad \ldots \quad]_{(1\times 2\omega)}$$

be the row vector of basis functions on the grid with resolution $h/2$. Here, $k \in \mathbb{Z} + m/2$, i.e., $k \in \mathbb{Z}$ if m is even, and $k \in \mathbb{Z} + 1/2$ if m is odd.

Proposition 2.2.3. *The row vector of basis functions associated with the grid of resolution* h *satisfies*

$$\mathcal{N}_h = \mathcal{N}_{h/2} \cdot \Sigma, \tag{2.17}$$

i.e.,

$$N_l^m(h;t) = \sum_{k=2l-m/2}^{2l+m/2} s_{k-2l} N_k^m(h/2;t).$$

Proof. By definition $(2.8)_{/55}$ it follows that

$$\begin{aligned} N^1(h;t) &= N^1(h/2;t-h/4) + N^1(h/2;t+h/4) \\ &= N^1_{1/2}(h/2;t) + N^1_{-1/2}(h/2;t). \end{aligned} \tag{2.18}$$

(See Figure $2.3_{/60}$, which is somewhat stylized so that the relevant functions can be distinguished.) This equation illustrates the simplest case of the "2-scale relation" $(2.25)_{/62}$, below, which is equivalent to $(2.24)_{/61}$ in the proof. Equation

$(2.18)_{/60}$ can be rewritten as

$$N^1(h;\cdot) = (z^{1/2} + z^{-1/2})N^1(h/2;\cdot).$$

Applying $(2.6)_{/55}$ $m-1$ times, we obtain

$$[N^1(h;\cdot)]^m = (z^{1/2} + z^{-1/2})^m [N^1(h/2;\cdot)]^m,$$

where the exponent m denotes m-fold convolution. But by $(2.11)_{/56}$ we have

$$\begin{aligned} N^m(h;\cdot) &= \frac{1}{h^{m-1}}[N^1(h;\cdot)]^m = \frac{1}{h^{m-1}}\left(z^{1/2} + z^{-1/2}\right)^m [N^1(h/2;\cdot)]^m \\ &= 2\left(\frac{z^{1/2} + z^{-1/2}}{2}\right)^m \frac{1}{(h/2)^{m-1}}[N^1(h/2;\cdot)]^m \\ &= 2\left(\frac{z^{1/2} + z^{-1/2}}{2}\right)^m N^m(h/2;\cdot). \end{aligned}$$

Thus,

$$N^m(h;t) = 2\left(\frac{z^{1/2} + z^{-1/2}}{2}\right)^m N^m(h/2;t) = s(z)N^m(h/2;t), \tag{2.19}$$

where we have introduced the generalized polynomial

$$s(z) \doteq 2\left(\frac{z^{1/2} + z^{-1/2}}{2}\right)^m = 2z^{-m/2}\left(\frac{1+z}{2}\right)^m = \sum_{i=-m/2}^{m/2} s_i z^i \tag{2.20}$$

with coefficients

$$s_i = \frac{1}{2^{m-1}}\binom{m}{i+m/2}, \tag{2.21}$$

and where the summation is over $i \in \mathbb{Z} + m/2$, i.e., over $i \in \mathbb{Z}$ if m is even and over $i \in \mathbb{Z} + 1/2$ if m is odd. Consequently, $(2.19)_{/61}$ is equivalent to

$$N^m(h;t) = \sum_{i=-m/2}^{m/2} s_i N_i^m(h/2;t), \tag{2.22}$$

and this generalizes to

$$N_l^m(h;t) = \sum_{i=-m/2}^{m/2} s_i N_{i+2l}^m(h/2;t). \tag{2.23}$$

Replacing the summation index by $k = i + 2l$, we have

$$N_l^m(h;t) = \sum_{k=2l-m/2}^{2l+m/2} s_{k-2l} N_k^m(h/2;t), \tag{2.24}$$

completing the proof. □

The 2-scale relation and the subdivision polynomial

The 2-scale relations described in the following remark will be used frequently later.

Remark 2.2.4. The formulas $(2.23)_{/61}$ and $(2.24)_{/61}$ can, with $l = 0$, be rewritten as

$$N^m(h;t) = \sum_k s_k N^m(h;2t-kh) = s(z)N(h;2t) \tag{2.25}$$

with $s(z) = \sum_k s_k z^k$. This follows from $(2.14)_{/56}$ which implies that $N_k^m(h/2;t) = N^m(h/2;t-kh/2) = N^m(h;2t-kh)$. The generalized polynomial $s(z)$ is called the (centered) *subdivision polynomial*.

The relation $(2.25)_{/62}$ expresses a property of the nodal function $N^m(h;t)$ and is called the 2-*scale relation* [86]. This 2-scale relation has been derived above starting with the obvious 2-scale relation $N^1(h,t) = N^1(h;2t-h/2)+N^1(h;2t+h/2) = (z^{-1/2}+z^{1/2})N^1(h;2t)$. Similar 2-scale relations defined by some subdivision polynomial $s(z)$ exist for a much wider class of nodal functions N, including functions defined in the bivariate case. ∎

We have now established the first half of the "remarkable property," mentioned above, of the matrix Σ: it describes the recursion for uniform B-spline basis functions; see $(2.17)_{/60}$. Proposition $2.2.3_{/60}$ is reformulated below, in Proposition $2.2.6_{/64}$, to express the relationship between control points from one step to the next.

The subdivision polynomial $s(z)$ in $(2.19)_{/61}$, $(2.20)_{/61}$, and Remark $2.2.4_{/62}$, as well as generalizations of this polynomial, are used throughout the book. The coefficients of the subdivision polynomial tell us how to express a nodal function in terms of a refined version of the nodal function, as shown in Proposition $2.2.3_{/60}$, and, also, how to find the refined set of control points given the control points p_l, as will be shown in Proposition $2.2.6_{/64}$. Important remarks about the interpretation of the subdivision polynomial appear in the discussion of the *subdivision equation*, which follows the latter proposition.

Generalizations of the polynomial in $(2.20)_{/61}$ are introduced in Chapters 3 and 4, in the context of box-spline and more general subdivision methods.

Example 2.2.5. Recursion for B-spline basis function of order 4.

The functions on the left-hand side of $(2.24)_{/61}$ are illustrated for $m = 4$ in Figure $2.4_{/63}$, where the weights for the five functions $N_{i+2l}^4(h/2;t)$ are, respectively, 1/8, 1/2, 3/4, 1/2, and 1/8. These weights are the coefficients of

$$\begin{aligned} s(z) = 2\left(\frac{z^{1/2}+z^{-1/2}}{2}\right)^4 &= \frac{2}{16}(z^2+4z+6+4z^{-1}+z^{-2}) \\ &= \frac{1}{8}z^2+\frac{1}{2}z+\frac{3}{4}+\frac{1}{2}z^{-1}+\frac{1}{8}z^{-2}. \end{aligned}$$

Figure $2.5_{/63}$ illustrates why a column of Σ is shifted by two positions, relative to an adjacent column, as noted above. In Proposition $2.2.3_{/60}$, increasing l by 1

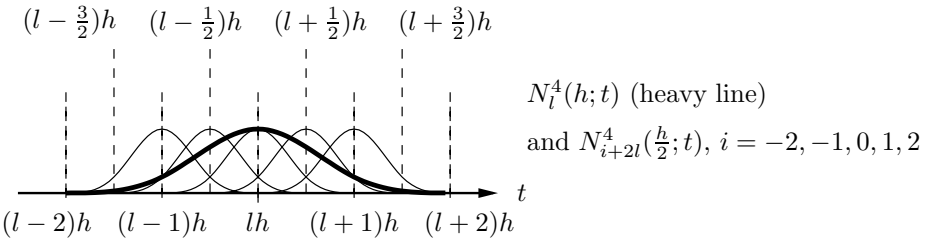


Figure 2.4. *Order-4 basis functions involved in the 2-scale relation.*

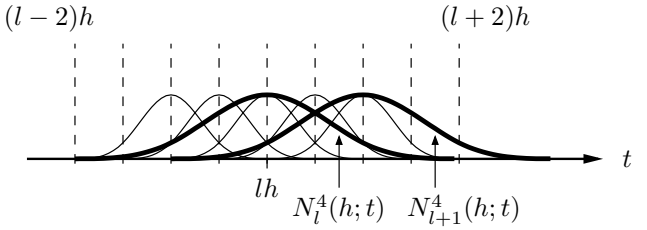


Figure 2.5. *Illustrating the two-position shift in columns of* Σ.

corresponds to looking at the basis function $N_{l+1}^m(h;t)$ instead of $N_l^m(h;t)$; i.e., it corresponds to considering a basis function shifted h units to the right. But it is clear from Figure 2.5$_{/63}$ that the corresponding basis functions on the refined grid are found by looking at the functions two positions to the right. ∎

Exercise 2$_{/91}$ asks that a discussion analogous to that in Example 2.2.5$_{/62}$ be given in the case $m = 3$.

We pointed out in Section 1.4.2 that the coefficients in (2.21)$_{/61}$ are exactly those in the columns of the matrix in (1.14)$_{/41}$, in the case $m = 4$. It is shown next that multiplying the grid-size-h control-point vector on the left by Σ corresponds exactly to one step of the subdivision process, as stated in Section 1.4.2.

Recursion formulas for control points for curves

Assume that we are given the curve

$$x(t) = \sum_{l\in\mathbb{Z}} p_l N_l^m(h;t) \quad (p_l \leftrightarrow \text{grid-size } h). \tag{2.26}$$

We now show how to find a representation

$$x(t) = \sum_{k\in\mathbb{Z}+m/2} q_k N_k^m(h/2;t) \quad (q_k \leftrightarrow \text{grid-size } h/2) \tag{2.27}$$

using spline basis functions defined on a refined grid with grid-size $h/2$. The Lane–Riesenfeld algorithm, which was introduced in a preliminary way in Section 1.3.2,

will accomplish this, and the following proposition is the mathematical basis of the Lane–Riesenfeld algorithm.

Proposition 2.2.6. *Let $p_{(\omega\times N)}$ be the one-dimensional array, viewed as a column vector, of control points p_l associated with the grid of resolution h. Then, there exists another array of control points q_k associated with the grid of resolution $h/2$ for which $(2.27)_{/63}$ is satisfied, and $q_{(2\omega\times N)}$ satisfies*

$$q = \Sigma\, p, \tag{2.28}$$

i.e.,

$$q_k = \sum_{l\in\mathbb{Z}} s_{k-2l} p_l, \quad k \in \mathbb{Z} + m/2. \tag{2.29}$$

Proof. Equations $(2.26)_{/63}$ and $(2.27)_{/63}$ may be rewritten as

$$x(t) = \mathcal{N}_h p = \mathcal{N}_{h/2} q.$$

Inserting $(2.17)_{/60}$, we get

$$\mathcal{N}_{h/2}\Sigma p = \mathcal{N}_{h/2} q \tag{2.30}$$

which is satisfied if

$$q = \Sigma\, p, \tag{2.31}$$

and $(2.31)_{/64}$ can also be written as

$$q_k = \sum_{l\in\mathbb{Z}} s_{k-2l} p_l, \quad k \in \mathbb{Z} + m/2.$$

According to Remark $2.2.2_{/59}$, this representation is unique. □

We now make some further comments on Propositions $2.2.3_{/60}$ and $2.2.6_{/64}$ and their proofs. In the following discussion we use the abbreviations $\sum_l = \sum_{l\in\mathbb{Z}}$ and $\sum_k = \sum_{k\in\mathbb{Z}+m/2}$.

Observe that if we use $(2.1)_{/54}$ and the shift operator introduced above, the representation $(2.26)_{/63}$ can be rewritten as

$$x(t) = \sum_l p_l z^{2l} N^m(h;t) = \left(\sum_l p_l z^{2l}\right) N^m(h;t),$$

and inserting $(2.19)_{/61}$, we have

$$x(t) = \left(\sum_l p_l z^{2l}\right) s(z) N^m(h/2;t).$$

Similarly, $(2.27)_{/63}$ can be rewritten as

$$x(t) = \sum_k q_k z^k N^m(h/2;t) = \left(\sum_k q_k z^k\right) N^m(h/2;t).$$

These representations are equivalent if

$$p(h/2; z) = s(z)\, p(h; z^2), \tag{2.32}$$

i.e., if

$$p(h/2; z) = 2 \left(\frac{z^{1/2} + z^{-1/2}}{2} \right)^m p(h; z^2), \tag{2.33}$$

where we have introduced the notation

$$p(h; z) = \sum_l p_l z^l \quad (p_l \leftrightarrow \text{grid-size } h)$$

and

$$p(h/2; z) = \sum_k q_k z^k \quad (q_k \leftrightarrow \text{grid-size } h/2).$$

Inserting $(2.20)_{/61}$ in $(2.32)_{/65}$, we get

$$\sum_k q_k z^k = \left(\sum_{i=-m/2}^{m/2} s_i z^i \right) \sum_l p_l z^{2l} = \sum_{i=-m/2}^{m/2} \sum_l s_i p_l z^{i+2l} \tag{2.34}$$

or, substituting $k = i + 2l$, i.e., $i = k - 2l$,

$$\sum_k q_k z^k = \sum_{k=-m/2+2l}^{m/2+2l} \left(\sum_l s_{k-2l} p_l \right) z^k,$$

where in the outer summation $k \in \mathbb{Z}$ if m is even and $k \in \mathbb{Z} + 1/2$ if m is odd. For the new control points associated with the refined grid, we therefore take

$$q_k = \sum_l s_{k-2l} p_l, \tag{2.35}$$

where $k \in \mathbb{Z}$ if m is even and $k \in \mathbb{Z} + 1/2$ if m is odd, and we have obtained an alternative derivation of $(2.28)_{/64}$ and $(2.29)_{/64}$ by using generating functions.

As already mentioned, the two propositions above show that the matrix Σ describes both the process of refinement of basis functions (Proposition $2.2.3_{/60}$) and a single step of the subdivision process itself (Proposition $2.2.6_{/64}$).

Finally, notice the discrete convolution in $(2.35)_{/65}$, which corresponds to the multiplication in $(2.32)_{/65}$. This is an example of the "Multiplication" principle mentioned at the beginning of Section 2.1. The presence of the factor of 2 in the subscript $k - 2l$ in $(2.35)_{/65}$ corresponds to the argument z^2 in $p(h; z^2)$ in $(2.32)_{/65}$.

The subdivision equation and the subdivision mask

Equation $(2.29)_{/64}$ is fundamental, and this equation or its generalizations are used in the study of all of the subdivision methods described in the book. We therefore give it a name: the *subdivision equation*.

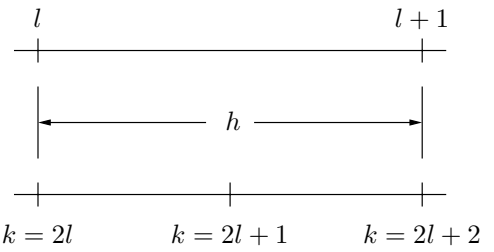


Figure 2.6. *Grid indexing at adjacent levels of resolution (univariate case).*

The subdivision equation gives the control points in the refined mesh, after a single subdivision step, in terms of the control points in the unrefined mesh. This relationship is described in $(2.29)_{/64}$ by means of the coefficients of the subdivision polynomial in the following way. The control points p_l at the beginning of the subdivision step are indexed on the grid of resolution h, whereas following the subdivision step the same grid point is referenced by q_k by doubling the index: $k = 2l$. (This doubled index can be viewed as the index of the grid point on the grid of resolution $h/2$; in addition, there are new grid points k in the refined grid with k odd. See Figure $2.6_{/66}$.) The *subdivision equation* $(2.29)_{/64}$ focuses on the index k (or row k in $(2.16)_{/59}$) and asserts that the control point q_k receives a contribution from each nearby control point p_l. Here, a "contribution from each p_l" corresponds exactly to a contribution from each column of $(2.16)_{/59}$, and "nearby" means that $|k - 2l| \leq m/2$. This last inequality can be rewritten as

$$l \text{ such that } k - 2l \in G \doteq \{-m/2, -m/2 + 1, \ldots, m/2\},$$

which corresponds to the notation that is used often later in the book. The weight assigned to each contribution is s_{k-2l}.

As we saw in Chapter 1, the steps defined by a subdivision matrix such as $(2.16)_{/59}$ are often described by stencils, with one stencil corresponding to each row of the matrix.

The subdivision equation $(2.29)_{/64}$ also gives another interpretation of the subdivision polynomial itself. Looking again at $(2.16)_{/59}$, and focusing now on column l, we see that the coefficients of the subdivision polynomial specify how the weight of p_l is distributed by a subdivision step over the new control points q_k. This set of coefficients is referred to as the *mask* of the subdivision scheme [25, 44, 124].

Example 2.2.7. Subdivision-polynomial description of recursion.

For $m = 2$, $(2.33)_{/65}$ becomes

$$p(h/2; z) = 2\left(\frac{z^{1/2} + z^{-1/2}}{2}\right)^2 p(h; z^2).$$

The product $2p(h;z^2)$ is the representation of a sequence having the doubled control points $2p_l$ assigned to even indices k, and the value 0 assigned to odd indices k:

$$\ldots, 0, 2p_{l-1}, 0, 2p_l, 0, 2p_{l+1}, \ldots.$$

Multiplying by the factor $(\frac{z^{1/2}+z^{-1/2}}{2})$, we obtain the sequence with half-integer indexing (relative to k)

$$\ldots, p_{l-2}, p_{l-1}, p_{l-1}, p_l, p_l, p_{l+1}, p_{l+1}, \ldots,$$

and multiplying by the same factor again gives the sequence with whole-integer indexing (again, relative to k)

$$\ldots, p_{l-2}, (p_{l-2}+p_{l-1})/2, p_{l-1}, (p_{l-1}+p_l)/2, p_l, (p_l+p_{l+1})/2, \ldots. \tag{2.36}$$

Alternatively, we might have said that $p(h;z^2)$ corresponds to the sequence

$$\ldots, 0, p_{l-1}, 0, p_l, 0, p_{l+1}, \ldots,$$

and applying

$$s(z) = 2\left(\frac{z^{1/2}+z^{-1/2}}{2}\right)^2 = \frac{1}{2}z^{-1} + 1 + \frac{1}{2}z$$

directly to this sequence produces immediately the sequence $(2.36)_{/67}$ with whole-integer indexing. The advantage of using the subdivision polynomial is clear: we can manipulate this polynomial algebraically, rather than writing out the various sequences explicitly.

It is useful to make a mental note of the fact that from $(2.1)_{/54}$ integral powers of z, such as the powers -1, 0, and 1 in $s(z) = z^{-1}/2 + z^0 + z^1/2$, refer to steps in the grid of resolution $h/2$, indexed here by k. ∎

2.2.3 The Lane–Riesenfeld algorithm

We now interpret Proposition $2.2.6_{/64}$ as the mathematical foundation for the Lane–Riesenfeld algorithm $LR(d)$. Thus, for now, we discuss only the version of the algorithm that applies to curves, viewing the linearly linked set of control points as a linear mesh.

The Lane–Riesenfeld algorithm: Initial substeps

A single step in the $LR(d)$ algorithm begins by performing substeps that linearly subdivide the mesh. Then, it performs an additional $m-2$ substeps, each involving an averaging operation, with alternation between the primal refined mesh and the dual of the refined mesh. This does not mean that both of these meshes must be constructed for an implementation: we are discussing only the mathematical structure of the method. The process defines a limit curve made up of piecewise polynomials of degree $m-1$, and in particular, if at each step we stop immediately after linear subdivision ($m = 2$), the process defines a piecewise-linear limit surface.

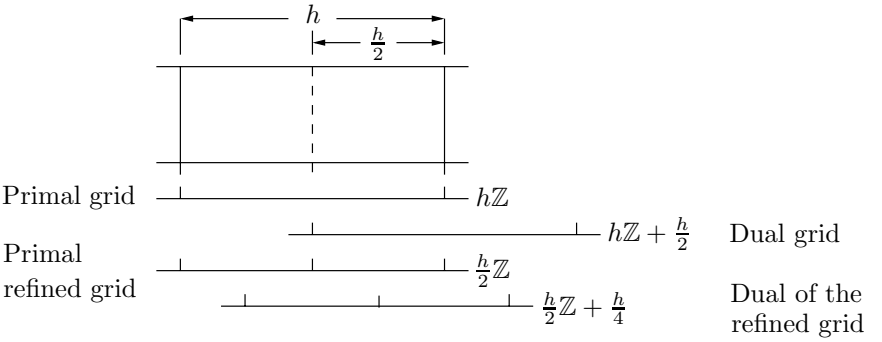


Figure 2.7. *Relevant grids in the univariate case.*

Linear subdivision in the univariate case could also be referred to as the $LR(1)$ algorithm.

In Figure 2.7$_{/68}$ we display the one-dimensional grids involved in the case of curves:

the *primal grid* $h\mathbb{Z}$ (this grid has the nodes lh, $l \in \mathbb{Z}$);

the *dual grid* $h\mathbb{Z} + \frac{h}{2}$ (this grid has the nodes $(l + 1/2)h$, $l \in \mathbb{Z}$);

the *primal refined grid* $\frac{h}{2}\mathbb{Z}$ (this grid has nodes $lh/2$, $l \in \mathbb{Z}$);

the *dual of the refined grid* $\frac{h}{2}\mathbb{Z} + \frac{h}{4}$ (this grid has nodes $(l + 1/2)h/2$, $l \in \mathbb{Z}$).

By definition, the functions $N_l^m(h; t)$ have their supports in the intervals

$$[(l - m/2)h, (l + m/2)h]$$

and their peak values are taken at the nodes lh of the original *primal* grid. Similarly, the functions $N_l^m(h/2; t)$ have their supports in the intervals

$$[(l - m/2)h/2, (l + m/2)h/2]$$

and their peak values are taken at the nodes $lh/2$ of the primal refined grid $\frac{h}{2}\mathbb{Z}$.

Equation (2.33)$_{/65}$ is the central equation related to the $LR(d)$ algorithm. The algorithm begins with the sequence $p(h; z)$ and replaces it by the sequence $2p(h; z^2)$; it then performs $m = d + 1$ subsequent substeps, applying the operator $(z^{1/2} + z^{-1/2})/2$ at each substep k, $k = 1, \ldots, m$. This corresponds to an averaging process at each substep. In terms of the centered subdivision polynomial, this is equivalent to initializing $s(z) \equiv 2$ for $k = 0$, and then multiplying the polynomial by the factor $(z^{1/2} + z^{-1/2})/2$ for each substep, $k = 1, \ldots, m$. We begin by showing how the initial substeps of the algorithm accomplish linear subdivision. This has already been discussed informally in Example 2.2.7$_{/66}$.

The operations in $(2.33)_{/65}$ can be represented as follows. The initial sequence $p(h;z)$ associates control points with a primal grid, producing a linear mesh. In an initial substep 0 we introduce coefficients defined by

$$\begin{aligned} q^0_{2l} &= 2p_l, \\ q^0_{2l+1} &= 0. \end{aligned} \tag{2.37}$$

Then

$$2p(h;z^2) = 2\sum_l p_l z^{2l} = \sum_j q^0_j z^j, \tag{2.38}$$

and this polynomial corresponds to a sequence defined on the refined primal grid, where the control points on the new nodes are set to zero, whereas the values on the original nodes are multiplied by a factor of 2. This initial substep in the process is sometimes referred to [168] as upsampling.[30] The factor of 2 in q^0_{2l} is necessary to initialize the process of applying the operator $(z^{1/2}+z^{-1/2})/2$ at each substep, in a way that is consistent with the fact that $(2.19)_{/61}$ is true for $m=1$.

In substep 1 we form the sequence q^1_i defined for $i \in \mathbb{Z}+1/2$ by

$$q^1_i = (q^0_{i-1/2} + q^0_{i+1/2})/2,$$

where $i = j + \frac{1}{2}$, and the index j on q^0_j is either even ($j=2l$) or odd ($j=2l+1$), $l \in \mathbb{Z}$. This sequence is associated with $\frac{h}{2}\mathbb{Z} + \frac{h}{4}$, the dual of the refined grid, and represents multiplication of $2p(h;z^2)$ by a factor $(z^{1/2}+z^{-1/2})/2$, so that

$$p^1(z) \doteq \sum_{i\in\mathbb{Z}+1/2} q^1_i z^i = \left(\frac{z^{1/2}+z^{-1/2}}{2}\right) 2\,p(h;z^2). \tag{2.39}$$

We have

$$q^1_{j+\frac{1}{2}} = (q^0_j + q^0_{j+1})/2 = \begin{cases} (q^0_{2l} + q^0_{2l+1})/2 = p_l, & j = 2l, \\ (q^0_{2l+1} + q^0_{2l+2})/2 = p_{l+1}, & j = 2l+1 \end{cases} \tag{2.40}$$

(see Figure $2.8_{/70}$).

In substep 2 we define q^2_j for $j \in \mathbb{Z}$ by

$$q^2_j = (q^1_{j-1/2} + q^1_{j+1/2})/2,$$

which gives

$$p^2(z) \doteq \sum_{j\in\mathbb{Z}} q^2_j z^j = \left(\frac{z^{1/2}+z^{-1/2}}{2}\right)^2 2\,p(h;z^2)$$

and

$$q^2_j = \begin{cases} p_l, & j = 2l, \\ (p_l + p_{l+1})/2, & j = 2l+1. \end{cases} \tag{2.41}$$

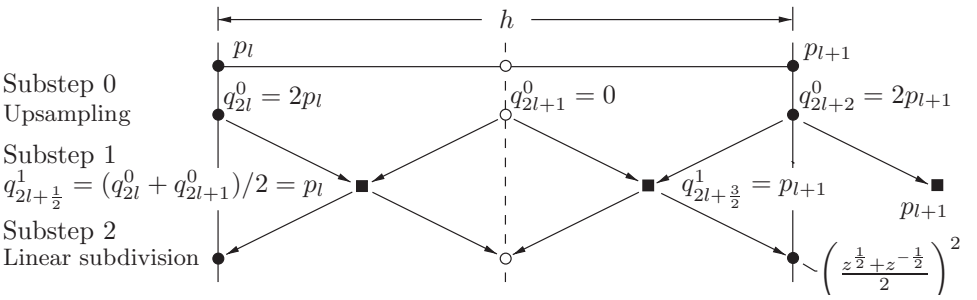


Figure 2.8. *First substeps of the univariate Lane–Riesenfeld algorithm.*

The first substep described here, in (2.40)$_{/69}$, has the effect of transferring a typical control-point value p_{l+1} to two vertices of the dual of the refined grid, one on each side of the vertex in the primal grid (see Figures 2.7$_{/68}$ and 2.8$_{/70}$). This corresponds to a method called "constant subdivision" which is trivial, but which is nonetheless useful for describing other methods. Similarly, in Section 1.5 and (1.2)$_{/23}$, substeps 1 and 2 here were described[31] as a single substep called "linear subdivision." We saw in Example 2.2.7$_{/66}$ that applying

$$s(z) = 2\left(\frac{z^{1/2}+z^{-1/2}}{2}\right)^2 = \frac{1}{2}z^{-1} + 1 + \frac{1}{2}z \tag{2.42}$$

directly to the sequence $\ldots, 0, p_{l-1}, 0, p_l, 0, p_{l+1}, \ldots$ yields the new sequence

$$\ldots, (p_{l-2}+p_{l-1})/2, p_{l-1}, (p_{l-1}+p_l)/2, p_l, (p_l+p_{l+1})/2, p_{l+1}, \ldots.$$

The Lane–Riesenfeld algorithm: Substeps achieving order $m \geq 3$

The process so far is illustrated in Figure 2.8$_{/70}$. These first two substeps have accomplished a linear subdivision of the mesh. If the process is stopped here at each step, it provides a method with $m = 2$, i.e., a process with a piecewise-linear limit function. If an additional substep is introduced at each step ($m = 3$), we obtain the Chaikin method (i.e., the $LR(2)$ algorithm), as we now show.

Each subsequent substep (when $m > 2$) increases the degree of the piecewise polynomials, of which the limit curve is formed, by one, and increases the degree of continuity of the limit curve by one. Repeating the process started above, for $r \geq 1$, we obtain the sequences q_i^{2r+1} defined for $i \in \mathbb{Z} + 1/2$ (i.e., associated with the dual of the refined grid) by

$$q^{2r+1}_{i+1/2} = (q^{2r}_i + q^{2r}_{i+1})/2, \tag{2.43}$$

so that

$$p^{2r+1}(z) = \sum_{i\in\mathbb{Z}+1/2} q_i^{2r+1} z^i = \left(\frac{z^{1/2}+z^{-1/2}}{2}\right)^{2r+1} 2\,p(h; z^2),$$

and the sequences q_j^{2r+2} defined for $j \in \mathbb{Z}$ (i.e., associated with the primal refined grid) by

$$q_j^{2r+2} = (q_{j-1/2}^{2r+1} + q_{j+1/2}^{2r+1})/2, \tag{2.44}$$

so that

$$p^{2r+2}(z) = \sum_{j\in\mathbb{Z}} q_j^{2r+2} z^j = \left(\frac{z^{1/2} + z^{-1/2}}{2}\right)^{2r+2} 2\,p(h; z^2).$$

Thus, we alternate between the primal and the dual of the refined grid until we reach the final substep in $(2.33)_{/65}$.

To transform these expressions into an algorithm for a given value of $m \geq 2$, we see that for each $r = 1, \ldots, \lfloor \frac{m-2}{2} \rfloor$, we must apply $(2.43)_{/70}$ and $(2.44)_{/71}$, followed, if m is odd, by application of $(2.43)_{/70}$ with $r = \frac{m-1}{2}$. Thus, we begin with linear subdivision, and then perform $m-2$ successive substeps, where in each substep we simply average the control-point values obtained in the previous substep. For $m = 3$ this is Chaikin's algorithm, introduced as an example in Section 1.5 ; equation $(2.43)_{/70}$ with $r = 1$ corresponds to $(1.19)_{/46}$. More importantly, for each m, a tensor-product version of the algorithm just described is the basis of the Lane–Riesenfeld algorithm $LR(d \times d)$ for subdivision surfaces (Sections 1.3.1 and 2.3), and of the Repeated Averaging algorithm for subdivision surfaces, described in Section 1.3.1.

Until now, we have been discussing the substeps of one step of the $LR(d)$ algorithm. After ν full steps, the nodes are in $h2^{-\nu}\mathbb{Z}$ if m is even (as in Section 1.4.2), and in $h2^{-\nu}\mathbb{Z} + h2^{-(\nu+1)}$ if m is odd. Exercise $3_{/92}$ asks for the subdivision polynomial corresponding to ν complete steps of the $LR(d)$ algorithm.

2.2.4 Two important principles

We conclude this section with some remarks on the values of $s(-1)$ and $s(1)$, and with the statement of two important principles related to the use of the *unit-impulse* function,[32] i.e., the function defined on the discrete grid of resolution h, with value 1 at the origin and 0 elsewhere. Because these two principles are mentioned so frequently in the book, we give them names: they are the Nodal-Function Computation principle and the Polynomial Coefficient principle. They are introduced here in the context of univariate B-splines, but later in the book they are shown to apply in very general situations in the bivariate case.

Interpretation of the values $s(-1)$ and $s(1)$

According to $(2.20)_{/61}$ we have $s(-1) = 0$ and $s(1) = 2$. These equalities[33] correspond to the statement that in each *column* of Σ, summing separately over row indices k with $\lfloor k \rfloor$ even, and over row indices k with $\lfloor k \rfloor$ odd, must always produce 1. A further interpretation of the two equalities is given in Remark $2.2.8_{/72}$.

It is observed in Section 4.5 that a univariate linear subdivision process defined by a subdivision polynomial $s(z) = \sum_{k\in\mathbb{Z}} s_k z^k$ is affine invariant if and only if $s(-1) = 0$ and $s(1) = 2$. See Theorem $4.5.1_{/172}$.

This equivalence does not depend on s being centered; i.e., there is no requirement that the coefficients of s be centered around the origin.

Remark* 2.2.8. The two equalities just mentioned can be viewed in the univariate B-spline case as guaranteeing an even or equitable distribution of the weight of the control points at each step of the subdivision.

Consider first the case when m is even, so that the rows of Σ are indexed by $k \in \mathbb{Z}$. Then, according to $(2.44)_{/71}$, the values obtained at the end of a complete step are in the primal refined grid $\frac{h}{2}\mathbb{Z}$ (see Figure $2.7_{/68}$). Examining the primal refined grid in Figure $2.7_{/68}$, we see that there are two kinds of nodes in the grid: those corresponding to existing nodes in the primal (nonrefined) grid, and those corresponding to new nodes, situated half way between them. In [75], these are referred to as *even* and *odd* nodes, respectively. Now, the statement that in each column of Σ, summing separately over rows with k even and over rows with k odd, always produces 1, can be interpreted as follows. The total weight assigned to a particular control point p_l doubles as the result of a subdivision step, but not in completely arbitrary fashion: rather, it is distributed evenly between even nodes and odd nodes. Thus, the subdivision step does two things: it distributes the weight of p_l across all even nodes, and in addition, it distributes the weight of p_l across all odd nodes, so that the total weight doubles.

A similarly equitable distribution of weight occurs when m is odd. In this case, according to $(2.43)_{/70}$, the values obtained at the end of a complete step are in the dual of the refined grid, i.e., in $\frac{h}{2}\mathbb{Z} + \frac{h}{4}$ (see Figure $2.7_{/68}$). Examining the dual of the refined grid, we see that again there are two types of nodes in the grid. We might call a node situated $h/4$ units to the left of a node in the primal (nonrefined) grid a "left node," and a node situated $h/4$ units to the right a "right node." Again the total weight assigned to a particular control point p_l doubles as the result of a subdivision step, but it is distributed evenly between left nodes and right nodes. Thus, the subdivision step distributes the weight of p_l across all left nodes, and in addition, it distributes the weight of p_l across all right nodes, so that the total weight doubles. ∎

Nodal-Function Computation principle

Given the definition of $N_l^m(h;t)$ in Section 2.2.1, we can find explicit analytic expressions for this function. The same will be true in the tensor-product and box-spline cases (although the analysis is more complicated). Here, however, we note a useful principle, valid also in more general contexts, that is mentioned often below. If we are given any convergent (in a sense described in Definitions $4.7.1_{/182}$ and $5.1.1_{/193}$) subdivision procedure defining sequences q_k (corresponding to grid-size $h/2$) in terms of sequences p_l (corresponding to grid-size h), as in $(2.26)_{/63}$ and $(2.27)_{/63}$, we can compute an approximation for the nodal function $N_l^m(h;t)$ associated with $l \in \mathbb{Z}$ even if we have no knowledge about its analytic form. This can be done by simply applying the subdivision process, until the approximation is satisfactory, to the scalar control points $p_l = 1$ and $p_i = 0$ if $i \neq l$. The algorithm will converge to $N_l^m(h;t)$, since substituting $p_l = 1$ and $p_i = 0$, $i \neq l$, into $(2.26)_{/63}$

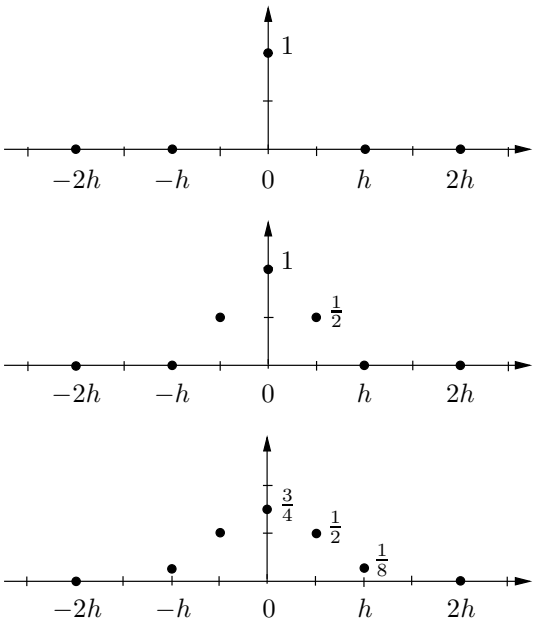


Figure 2.9. *Polynomial Coefficient principle illustrated for $LR(3)$.*

gives $x(t) = N_l^m(h;t)$. In more general situations the basis functions N_l^m are called nodal functions.

Polynomial Coefficient principle

A second useful principle related to the unit-impulse function is the following. Given the subdivision rules, perhaps defined by stencils, that describe how to find control-point values on a grid of resolution $h/2$ given the values at resolution h, it is possible to find the coefficients of the subdivision polynomial by applying the rules to the unit-impulse function. The rules are applied for a single step of the subdivision procedure, in Jacobi manner. It is clear in fact that if we set $p_l = 1$ for $l = 0$, and $p_l = 0$ for $l \neq 0$, then $(2.35)_{/65}$ gives $q_k = s_k$.

We can illustrate this principle with a simple example. Suppose that we are interested in finding the coefficients of the subdivision polynomial for the univariate degree-three Lane–Riesenfeld algorithm $LR(3)$, by using the Polynomial Coefficient principle. We begin by placing a 1 at the origin, and 0 elsewhere, as illustrated in Figure $2.9_{/73}$ (top). Then, according to the rules given in Section 2.2.3, we must linearly subdivide, and then perform $m - 2 = 4 - 2 = 2$ successive averaging substeps. Given the values in Figure $2.9_{/73}$ (top), linear subdivision produces the values shown in Figure $2.9_{/73}$ (middle), and the two subsequent averagings produce the values in Figure $2.9_{/73}$ (bottom), which are exactly those shown in column 0 of Σ in $(1.14)_{/41}$. See also $(2.16)_{/59}$.

We have justified the two principles above by reference to the appropriate mathematical equations, but we also appeal to the reader's intuition about the interpretation of the mathematical objects involved. In the discussion of the subdivision mask, earlier in this section, it was shown that the coefficients of the subdivision polynomial (i.e., the subdivision mask) specify how the weight of p_l is distributed by a subdivision step over the new control points q_k, and this makes the Polynomial Coefficient principle quite obvious at an intuitive level. An intuitive interpretation of the Nodal-Function Computation principle follows immediately: applying the subdivision process not just once, but infinitely often until convergence, we obtain the influence of p_l over a dense set of points in the parametric region surrounding the parameter value $t = hl$ that corresponds to p_l. But, according to $(2.26)_{/63}$, it is exactly $N_l^m(h;t)$ that specifies the influence of p_l on this surrounding part of the parametric domain.

2.3 Tensor-product surfaces

The results in Section 2.2 can be applied immediately to produce tensor-product B-spline surfaces. The curve $(2.15)_{/58}$ given as

$$x(t) = \sum_{l\in\mathbb{Z}} p_l N_l^m(h;t)$$

is replaced by the surface

$$x(u,v) = \sum_{l_1\in\mathbb{Z}} \sum_{l_2\in\mathbb{Z}} p_{l_1,l_2} N_{l_1}^{m_1}(h;u) N_{l_2}^{m_2}(h;v). \tag{2.45}$$

To simplify the exposition, we assume that $m_1 = m_2 = m$, and write

$$x(u,v) = \sum_{l_1} \sum_{l_2} p_{l_1,l_2} N_{l_1}^{m}(h;u) N_{l_2}^{m}(h;v). \tag{2.46}$$

Equation $(2.46)_{/74}$ can be rewritten as

$$\begin{aligned} x(u,v) &= \left(\sum_{l_1} \sum_{l_2} p_{l_1,l_2} z_1^2 z_2^2 \right) N^m(h;u) N^m(h;v) \\ &= \left(\sum_{k_1} \sum_{k_2} q_{k_1,k_2} z_1 z_2 \right) N^m(h/2;u) N^m(h/2;v), \end{aligned}$$

and by $(2.19)_{/61}$ we have

$$\begin{aligned} N^m(h;u) &= s(z_1) N^m(h/2;u), \\ N^m(h;v) &= s(z_2) N^m(h/2;v). \end{aligned}$$

Consequently,

$$\begin{aligned} x(u,v) &= p(z^2) s(z_1) s(z_2) N^m(h/2;u) N^m(h/2;v) \\ &= q(z) N^m(h/2;u) N^m(h/2;v), \end{aligned}$$

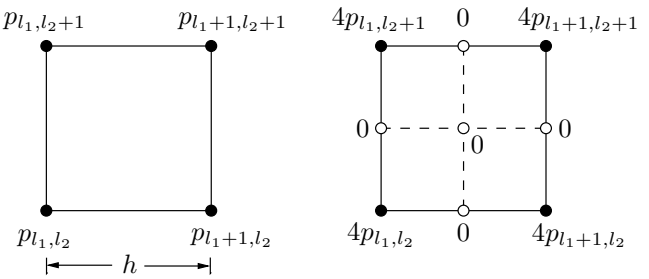


Figure 2.10. *Upsampling in the bivariate case.*

so that

$$q(z) = s(z_1)s(z_2)p(z^2) = \bar{s}(z)p(z^2)$$

and

$$q_k = \sum_{l \in \mathbb{Z}^2} \bar{s}_{k-2l} p_l,$$

where

$$\bar{s}_{k_1,k_2}(z) = s_{k_1}(z_1)s_{k_2}(z_2),$$

i.e.,

$$q_{k_1,k_2} = \sum_{l_1,l_2} s_{k_1-2l_1} s_{k_2-2l_2} p_{l_1,l_2}. \tag{2.47}$$

Thus, we operate separately using the two univariate operators. The bar on $\bar{s}$ is dropped in subsequent chapters.

The tensor-product Lane–Riesenfeld algorithm $LR(d \times d)$ for surfaces begins with control points on the primal grid $h\mathbb{Z}^2$, as shown in Figure 2.10$_{/75}$ (left). Substep 0 (upsampling) produces the values shown in Figure 2.10$_{/75}$ (right), on the two-dimensional primal refined grid. This should be compared with the one-dimensional primal refined grid in Figure 2.7$_{/68}$ and the initial values in (2.37)$_{/69}$.

The next substep (substep 1) assigns values to the two-dimensional dual of the refined grid $\frac{h}{2}\mathbb{Z}^2 + (\frac{h}{4}, \frac{h}{4})$ and represents multiplication of

$$4p(h; z_1^2, z_2^2) = 4 \sum_{l_1} \sum_{l_2} p_{l_1,l_2} z_1^{2l_2} z_2^{2l_2}$$

by a factor of

$$\left(\frac{z_1^{1/2} + z_1^{-1/2}}{2} \right) \left(\frac{z_2^{1/2} + z_2^{-1/2}}{2} \right), \tag{2.48}$$

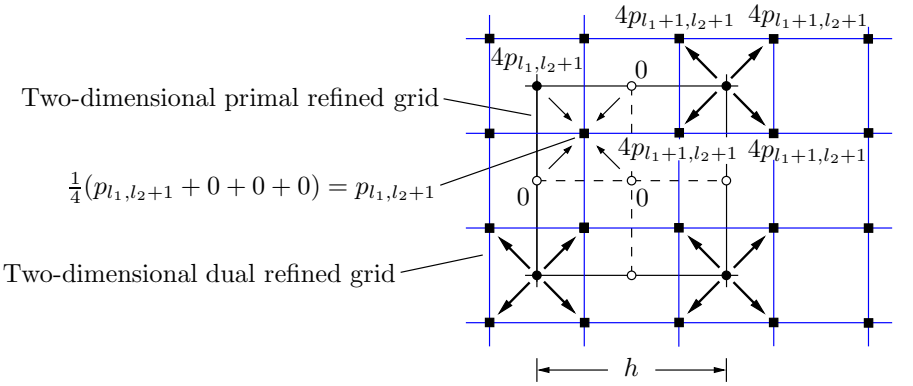


Figure 2.11. *Constant subdivision in the bivariate case.*

which averages the values shown in Figure $2.10_{/75}$ (right), producing the values shown in Figure $2.11_{/76}$. For example, the value p_{l_1,l_2+1} associated with the dual node in the upper left of Figure $2.11_{/76}$ is obtained by computing the average of the four values in the upper left-hand square of the grid in Figure $2.10_{/75}$ (right): $\frac{1}{4}(4p_{l_1,l_2+1}+0+0+0)$. Each of the four terms contributing to the average is indicated by a light arrow in Figure $2.11_{/76}$. This should be compared with the computed result of substep 1 in Figure $2.8_{/70}$.

The method described so far (which we could denote $LR(0\times 0)$) is called *constant* subdivision [177]. Its action is very simple: it forms the dual of the refined grid and copies the value of each original control point on the primal grid to the four associated dual vertices, as illustrated by heavy arrows in Figure $2.11_{/76}$. Computing the average of four terms, in which one term is a certain value multiplied by 4 and the other three are equal to zero, is equivalent to copying the value. The corresponding subdivision polynomial is

$$(z_1^{1/2}+z_1^{-1/2})(z_2^{1/2}+z_2^{-1/2}) = 2\left(\frac{z_1^{1/2}+z_1^{-1/2}}{2}\right)2\left(\frac{z_2^{1/2}+z_2^{-1/2}}{2}\right) \tag{2.49}$$
$$= z_1^{1/2}z_2^{1/2}+z_1^{1/2}z_2^{-1/2}+z_1^{-1/2}z_2^{1/2}+z_1^{-1/2}z_2^{-1/2}$$

(compare $(2.39)_{/69}$).

In analogy with $(2.42)_{/70}$, when $d=1$ we have

$$s(z) = 2\left(\frac{z_1^{1/2}+z_1^{-1/2}}{2}\right)^2 2\left(\frac{z_2^{1/2}+z_2^{-1/2}}{2}\right)^2, \tag{2.50}$$

where $z=(z_1,z_2)$, which corresponds to linear subdivision, $LR(1\times 1)$. Consequently, substep 2 involves multiplying again by the factor in $(2.48)_{/75}$, i.e., averaging the values in the dual grid and placing them in the primal grid, as shown in

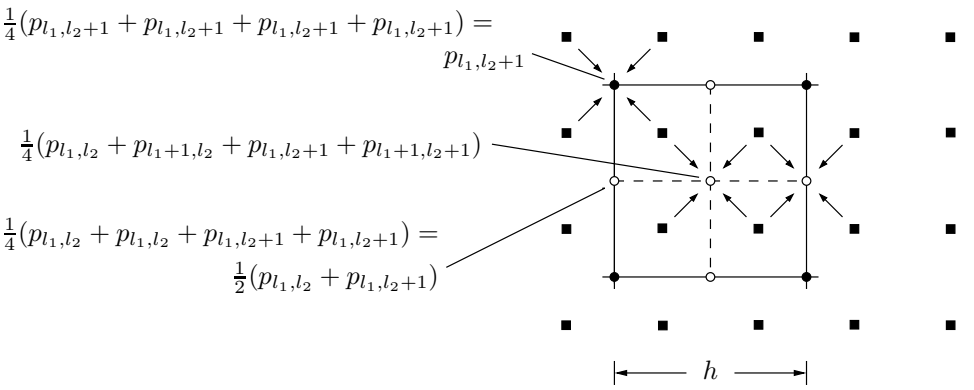


Figure 2.12. *Linear subdivision in the bivariate case.*

Figure $2.12_{/77}$. Thus, in analogy with substep 2 for curves, shown in Figure $2.8_{/70}$, substep 2 of the tensor-product process accomplishes linear subdivision.

Subsequent substeps similarly operate in a way analogous to the process for curves, alternating between the two-dimensional dual refined grid of Figure $2.11_{/76}$ and the two-dimensional primal refined grid. Each substep corresponds to a multiplication that adds another factor of the form $(2.48)_{/75}$ in order to eventually produce the subdivision polynomial

$$4\left(\frac{z_1^{1/2} + z_1^{-1/2}}{2}\right)^m \left(\frac{z_2^{1/2} + z_2^{-1/2}}{2}\right)^m, \tag{2.51}$$

and each substep involves an averaging of the points computed at the previous substep. For practical computation, there is no need to construct the dual mesh unless m is odd: if m is even, a pair of averages can be computed simultaneously and the result stored in the primal refined mesh. The pair of averages used in a double substep corresponds to the factor

$$\begin{aligned}\left(\frac{z_1^{1/2} + z_1^{-1/2}}{2}\right)^2 \left(\frac{z_2^{1/2} + z_2^{-1/2}}{2}\right)^2 = {} & \frac{1}{16}z_1^{-1}z_2 + \frac{1}{8}z_2 + \frac{1}{16}z_1z_2 \\ & + \frac{1}{8}z_1^{-1} + \frac{1}{4} + \frac{1}{8}z_1 \\ & + \frac{1}{16}z_1^{-1}z_2^{-1} + \frac{1}{8}z_2^{-1} + \frac{1}{16}z_1z_2^{-1}\end{aligned} \tag{2.52}$$

which corresponds to a smoothing in the primal mesh using the stencil shown in Figure $1.28_{/29}$ (left). (The right-hand side of $(2.52)_{/77}$ is an ordinary sum: the terms are written on three separate lines for purposes of comparison with the stencil just mentioned.)

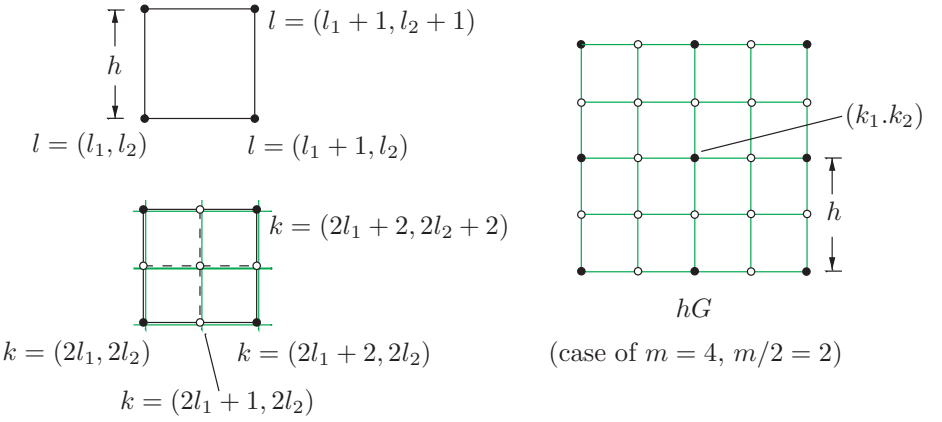


Figure 2.13. *Grid indexing at adjacent levels of subdivision (bivariate case).*

In general, it is necessary to compute $\lfloor m/2 \rfloor$ pairs of averages, in addition to one further simple averaging in the dual mesh in the case when m is odd.

Exercise $4_{/92}$ illustrates the relationship between the curve and tensor-product-surface versions of the Lane–Riesenfeld algorithm.

Writing $(2.47)_{/75}$ with its restricted summation domain, we have

$$q_k = \sum_{l:k-2l\in G} s_{k-2l} p_l, \tag{2.53}$$

where $k = (k_1, k_2)$ and $l = (l_1, l_2)$, and

$$G = \{(l_1, l_2) : -m/2 \le l_1, l_2 \le m/2\}.$$

Like $(2.53)_{/78}$ itself, the associated discussion is entirely analogous to the univariate case: the control points p_l at the beginning of the step are indexed on the grid of resolution h, whereas following the subdivision step, the same grid is referenced by doubling the index: $k = 2l$. See Figure $2.13_{/78}$ (left), which should be compared with Figure $2.6_{/66}$. The subdivision equation $(2.53)_{/78}$ asserts that the control point q_k receives a contribution from each nearby control point p_l, where "nearby" means l such that $k-2l \in G$, i.e., $-m/2 \le k_1-l_1, k_2-l_2 \le m/2$ (see Figure $2.13_{/78}$, right). The coefficients s_{k-2l}, $k - 2l \in G$, define the mask of the subdivision scheme.

The Nodal-Function Computation principle for the approximation of the bivariate nodal function $N^m_{l_1}(h;u)N^m_{l_2}(h;v)$ applies as in the scalar case, by applying the $LR(d,d)$ algorithm to the scalar control points $p_{(l_1,l_2)} = 1$ and $p_{(i_1,i_2)} = 0$ if $(i_1, i_2) \neq (l_1, l_2)$. Similarly, the Polynomial Coefficient Principle applies: we can obtain the subdivision mask s_k by applying a single step of the $LR(d \times d)$ algorithm to the scalar data $p_{(l_1,l_2)} = 1$ for $(l_1, l_2) = (0, 0)$, $p_{(l_1,l_2)} = 0$ for $(l_1, l_2) \neq (0, 0)$. This is the tensor-product version of the process illustrated in Figure $2.9_{/73}$ for $d = 3$.

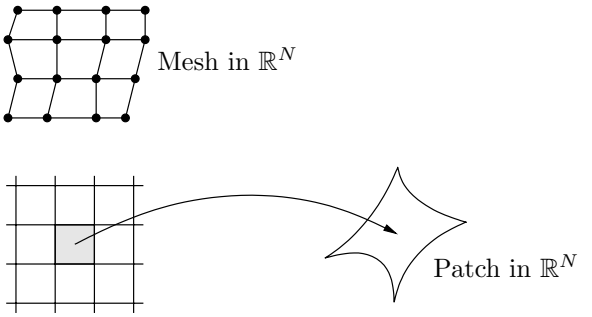


Figure 2.14. *Patch computed by $LR(3 \times 3)$ algorithm.*

2.4 B-spline methods for finite meshes

A basic subdivision method, applicable to *regular* finite quadrilateral meshes, is the ordinary $LR(d \times d)$ algorithm of Section 2.3. In fact, it is a class of methods, parametrized by the degree d, that corresponds to the lower row of column 1 of Figure $1.30_{/33}$, i.e., the case of regular meshes. If we implement a variant method, then we can view the basic $LR(d \times d)$ method as having been implemented on the regular part of the mesh. For example, if the Catmull–Clark method is applied to a polyhedral mesh containing the 16 control points illustrated in Figure $2.14_{/79}$ (top), then part of the resulting surface can be viewed as defined on a quadrilateral grid, as shown in Figure $2.14_{/79}$ (bottom left). Only the 16 control points indicated influence the value of the surface on the central square in the parametric domain (consider the interval $[0, h]$ in Figure $1.32_{/43}$). The image of the central square in the parametric domain can be viewed as defining a surface *patch* corresponding to the central face in the logical mesh defining the original polyhedral mesh. Once convergence has been proved, it follows by definition that each sequence of control points, corresponding to one of the four internal control points in the 16-point mesh, converges to a corner of the patch.

Similar statements can be made for the tensor-product $LR(d \times d)$ algorithm for arbitrary values of d. For example, in Section 1.3.1, variant methods were given that reduce to the basic $LR(d \times d)$ algorithm on regular meshes. Thus, for example, if d is odd and such a variant method is applied to a square polyhedral mesh with $(d+1) \times (d+1)$ vertices, then a patch will be defined that corresponds to the central face in a $(d+1) \times (d+1)$ mesh. Again, this patch can be expressed as a parametric function of the form $(2.46)_{/74}$. The case of d even is considered in Exercise $5_{/92}$.

These remarks can be extended to larger meshes, as illustrated for the case $d = 3$ in Figure $2.15_{/80}$, and the continuity properties on the combined patch will be as specified by the theory in Section 2.5.6. If, however, there are extraordinary vertices nearby, then it will be necessary to use more elaborate parametrizations to describe the extension of the surface. This is discussed in Section 6.3.2.

Variants of the Lane–Riesenfeld algorithm, applicable in the nonregular case, were presented in Section 1.3.1. They included the Repeated Averaging method

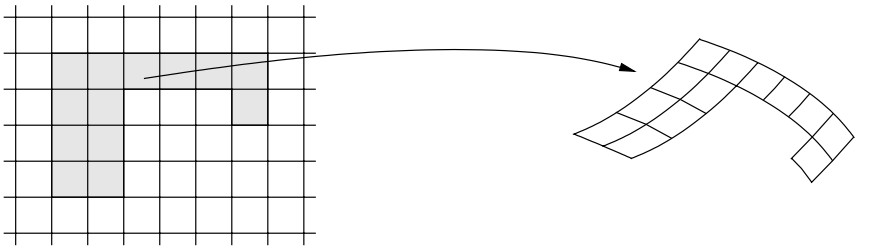


Figure 2.15. *Regular portion of surface computed by* $LR(3 \times 3)$ *algorithm.*

with $d-1$ averagings, the related LSS method (Exercise $9_{/49}$ of Chapter 1), and the Doo–Sabin and Catmull–Clark methods. This corresponds to the upper row in the first column of Figure $1.30_{/33}$.

2.5 Further results for univariate B-splines

We present here some further fundamental results for univariate B-splines. These results either generalize directly, or serve as models for results in more general cases.

2.5.1 Differentiation

For $k \geq 2$ we have, from $(2.9)_{/56}$,

$$\overset{*}{N}{}^k(h;t) = \frac{1}{h}(\overset{*}{N}{}^{k-1}(h;\cdot) \otimes \overset{*}{N}{}^1(h;\cdot))(t) = \frac{1}{h}\int_{t-h}^{t} \overset{*}{N}{}^{k-1}(h;s)\,ds,$$

so that, using the notation $D = \frac{d}{dt}$,

$$D\overset{*}{N}{}^k(h;t) = \frac{1}{h}(\overset{*}{N}{}^{k-1}(h;t) - \overset{*}{N}{}^{k-1}(h;t-h)), \tag{2.54}$$

and, using the translation operator z and $(2.12)_{/56}$,

$$D\overset{*}{N}{}_l^k(h;t) = \frac{1}{h}(1-z^2)\overset{*}{N}{}_l^{k-1}(h;t) = \frac{1}{h}(\overset{*}{N}{}_l^{k-1}(h;t) - \overset{*}{N}{}_{l+1}^{k-1}(h;t)).$$

Similarly, for centered spline functions we obtain

$$DN_l^k(h;t) = \frac{1}{h}(z^{-1}-z)N_l^{k-1}(h;t) = \frac{1}{h}(N_{l-1/2}^{k-1}(h;t) - N_{l+1/2}^{k-1}(h;t)).$$

If we replace k by m and repeat this process, we get the following for higher-order derivatives:

$$D^k\overset{*}{N}{}_l^m(h;t) = \frac{1}{h^k}(1-z^2)^k\overset{*}{N}{}_l^{m-k}(h;t) \tag{2.55}$$

and

$$D^k N_l^m(h;t) = \frac{1}{h^k}(z^{-1}-z)^k N_l^{m-k}(h;t). \tag{2.56}$$

The sizes of the jump discontinuities of $D^{m-1}\overset{*}{N}{}_l^m(h;\cdot)$ are needed later, and they may be computed using $(2.55)_{/80}$. Taking $l=0$ and $k=m-1$ in $(2.55)_{/80}$, expanding the binomial $(1-z^2)^{m-1}$, and using the definition of the translation operator z, we get

$$D^{m-1}\overset{*}{N}{}_0^m(h;t) = \frac{1}{h^{m-1}}\sum_{i=0}^{m-1}\binom{m-1}{i}(-1)^i \overset{*}{N}{}_0^1(h;t-ih). \tag{2.57}$$

Equation $(2.57)_{/81}$ shows that the function $D^{m-1}\overset{*}{N}{}_0^m(h;t)$ has a jump discontinuity

$$\begin{aligned}
&\left[D^{m-1}\overset{*}{N}{}_0^m(h;t)\right]_{t=ih-}^{t=ih+} \\
&= \frac{1}{h^{m-1}}\left[(-1)^i\binom{m-1}{i} - (-1)^{i-1}\binom{m-1}{i-1}\right] &(2.58)\\
&= \frac{1}{h^{m-1}}(-1)^i\binom{m}{i} &(2.59)
\end{aligned}$$

at the point $t=ih$, $i=0,\ldots,m$, where $\binom{m-1}{-1}$ and $\binom{m-1}{m}$ are zero by definition. For translated functions we have similarly

$$\left[D^{m-1}\overset{*}{N}{}_l^m(h;t)\right]_{t=ih-}^{t=ih+} = \frac{1}{h^{m-1}}(-1)^{i-l}\binom{m}{i-l}. \tag{2.60}$$

Thus, for $m\geq 2$, $\overset{*}{N}{}_l^m(h;\cdot)$ is in C^{m-2}, but not in C^{m-1}.

Control points for the derivative of a curve

Assume that we are given a spline curve

$$x(t) = \sum_{l\in\mathbb{Z}} p_l N_l^m(h;t)$$

represented by centered nodal functions and control points $\{p_l\}_{l\in\mathbb{Z}}$.

The tangent curve to this curve is then obtained by operating on the components of the curve with $D=\frac{1}{h}(z^{-1}-z)$, so that

$$Dx(t) = \frac{1}{h}(z^{-1}-z)\sum_{l\in\mathbb{Z}} p_l N_l^{m-1}(h;t) = \frac{1}{h}\sum_{l\in\mathbb{Z}} p_l(N_{l-1/2}^{m-1}(h;t) - N_{l+1/2}^{m-1}(h;t)),$$

i.e.,

$$Dx(t) = \frac{1}{h}\sum_{l\in\mathbb{Z}+1/2}(p_{l+1/2}-p_{l-1/2})N_l^{m-1}(h;t). \tag{2.61}$$

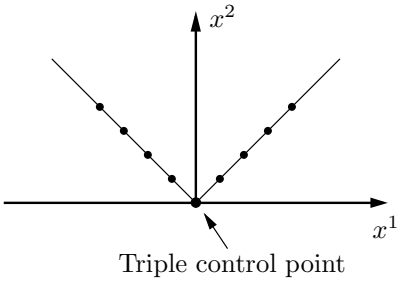


Figure 2.16. *Control points defining a C^2 curve with singularity.*

The control points p_l^d for the tangent curve are thus given by

$$p_l^d = (p_{l+1/2} - p_{l-1/2})/h, \quad l \in \mathbb{Z} + 1/2, \tag{2.62}$$

i.e., by taking centered differences on the original primal grid. The control points are now defined on the dual grid, and the spline functions are of order $m-1$, $m \geq 2$. For $m = 2$ there are jump discontinuities in the derivative, and therefore in the tangent.

Higher-order derivatives are obtained by repeating this process.

Even if we have parametric continuity of the derivatives of each component of the curve, there may nonetheless be singularities in the curve itself, for particular choices of the control points. A similar remark holds for surfaces, and many authors allude to this by saying, for example, that a certain result for subdivision surfaces holds for almost all control points, or that it is assumed that the control points are in general position [132, Thm. 3.5], [144, p. 30], [172, p. 66].

Example 2.5.1. Subdivision producing a C^2 curve with a singularity.

A simple example to illustrate the comments of the previous paragraph can be obtained by arranging the control points uniformly on the curve $(t, |t|)$, with the control point $(0, 0)$ repeated three times (see Figure 2.16$_{/82}$, where the superscripts denote components in $\mathbb{R}^2$).

Let

$$p_l = \begin{cases} (l+1, -(l+1)), & l \leq -1, \\ (0,0), & l = 0, \\ (l-1, l-1), & l \geq 1, \end{cases}$$

and apply the $LR(3)$ method of (1.14)$_{/41}$ with $h = 1$. Because of the repeated control points at the origin, the method converges in each of the first two quadrants to a half line terminating at the origin. For example, from (2.15)$_{/58}$ we have, in the first quadrant,

$$x(t) = \sum_{l=2}^{\infty} p_l N_l^4(1; t),$$

and the curve is equal to $(N_2^4(1;t), N_2^4(1;t))$ on $[0, 1]$. Similarly, the curve is equal to $(-N_{-2}^4(1;t), N_{-2}^4(1;t))$ on $[-1, 0]$, and on the real line it is a C^2 parametrization of $(t, |t|)$, with tangent vector equal to 0 at the origin: $Dx(0) = 0$.

A small perturbation of the control points is sufficient to remove the singularity in the curve. In this case the control points would be said to be in general position.

Exercise $6_{/92}$ asks for the result of applying the subdivision process once to the given control points. ∎

2.5.2 Partition of unity

For univariate uniform B-splines we have the following result concerning partition of unity.

Theorem 2.5.2.

$$\sum_{l\in\mathbb{Z}} N_l^m(h;t) = 1 \tag{2.63}$$

for all t, $m \geq 1$. *Further*

$$\int_{-\infty}^{\infty} N_l^m(h;t)\,dt = h. \tag{2.64}$$

Proof. The equality $(2.63)_{/83}$ is trivially true for $m = 1$, i.e.,

$$\sum_{l\in\mathbb{Z}} N_l^1(h;t) = 1.$$

Convolving repeatedly by $\frac{1}{h}N_0^1(h;\cdot)$ and using that

$$\frac{1}{h}N_0^1(h;\cdot) \otimes 1 = \int_{-\infty}^{\infty} \frac{1}{h}N_0^1(h;t)\,dt = 1,$$

the equality $(2.63)_{/83}$ follows.

Similarly $(2.64)_{/83}$ is trivially valid for $m = 1$. Using that $\int_{\mathbb{R}}(f \otimes g)(t)\,dt = \int_{\mathbb{R}} f(t)\,dt \int_{\mathbb{R}} g(t)\,dt$ and $(2.10)_{/56}$, the equality $(2.64)_{/83}$ follows by induction. □

A slightly more general local result is the following.

Corollary 2.5.3. *Assume that* $(a,b) \subset \mathbb{R}$ *is an open interval and that* $p_l = 1$ *for all* l *such that* $\operatorname{supp}(N_l^m(h;\cdot)) \cap (a,b) \neq \emptyset$. *Then*

$$\sum_{l\in\mathbb{Z}} p_l N_l^m(h;t) = 1 \tag{2.65}$$

for all $t \in (a,b)$.

Proof. The result is trivially valid for $m = 1$. For $m \geq 2$, the sum

$$\sum_{l\in\mathbb{Z}} p_l N_l^m(h;t)$$

does not change its value on the interval (a,b) if all coefficients p_l are taken to be equal to 1. Applying Theorem $2.5.2_{/83}$ completes the proof. □

The above result is actually valid in a much more general setting. In fact, every linear subdivision process that is affine invariant and which converges gives a partition of unity in terms of the nodal functions involved. In this more general context, a nodal function related to the node l is a function defined on the actual manifold by taking $p_l = 1$ at the node l and all other coefficients p_k equal to zero. This is discussed in Remark 4.7.3$_{/183}$.

Remark 2.5.4. It is clear that the statements of Theorem 2.5.2$_{/83}$ and Corollary 2.5.3$_{/83}$ are also true if the centered functions $N_l^m(h;\cdot)$ are subjected to a translation; i.e., the statement does not depend on the fact that the nodal functions are centered. ∎

2.5.3 Linear independence

We have the following theorem on *global* linear independence for univariate uniform B-splines.

Theorem 2.5.5. *If*

$$\sum_{l\in\mathbb{Z}} p_l N_l^m(h;t) = 0$$

for all t, then $p_l = 0$ for all $l \in \mathbb{Z}$.

A more general result on *local* linear independence is the following.

Theorem 2.5.6. *If*

$$\sum_{l\in\mathbb{Z}} p_l N_l^m(h;t) = 0$$

for all $t \in (a,b)$, then $p_l = 0$ for all l such that $\operatorname{supp}\left(N_l^m(h;\cdot)\right) \cap (a,b) \neq \emptyset$.

It is clear that Theorem 2.5.5$_{/84}$ follows from Theorem 2.5.6$_{/84}$ by taking $(a,b) = (-\infty,\infty)$.

Proof. The proof is by induction. The statement of the theorem is trivially true for $m = 1$. Next assume that the statement has been proved for $m = k$, $k \geq 1$. For $m = k+1$ we assume that $\sum_{l\in\mathbb{Z}} p_l N_l^{k+1}(h;t) = 0$ for all $t \in (a,b)$. Differentiating, using (2.62)$_{/82}$ and the induction hypothesis, we conclude that

$$p_{l+1/2} - p_{l-1/2} = 0 \quad \text{for} \quad a/h - k/2 < l < b/h + k/2.$$

It follows that p_l is constant for

$$a/h - (k+1)/2 < l < b/h + (k+1)/2,$$

and using Corollary 2.5.3$_{/83}$, we conclude that $\sum_{l\in\mathbb{Z}} p_l N_l^{k+1}(h;t)$ is equal to this constant on the interval (a,b). By the induction hypothesis, the constant is zero and the proof is complete. □

Remark 2.5.7. It is clear that the statement of Theorem 2.5.6$_{/84}$ is true if the centered functions $N_l^m(h;\cdot)$ are subjected to a translation. ∎

In Section 3.6 we give a more general theorem on linear independence, valid for a class of box splines.

2.5.4 A linear function space

Let us introduce the following linear function space.

Definition 2.5.8. *$\mathcal{B}_m(h\mathbb{Z})$ is the set of functions $x \in C^{m-2}(\mathbb{R})$ such that*

$$x|_{((i-1)h,ih)} \quad \textit{is a polynomial of degree } m-1 \textit{ for every } i \in \mathbb{Z}.$$

It is clear that $\overset{*}{N}{}_l^m(h;\cdot) \in \mathcal{B}_m(h\mathbb{Z})$ for all $l \in \mathbb{Z}$.

The following theorem expresses that the linear span of the set of functions $\{\overset{*}{N}{}_l^m(h;\cdot)\}_{l\in\mathbb{Z}}$ is $\mathcal{B}_m(h\mathbb{Z})$ and that they form a basis for this linear function space.

Theorem 2.5.9. *Every function $x \in \mathcal{B}_m(h\mathbb{Z})$ can be written as*

$$x(t) = \sum_{l\in\mathbb{Z}} p_l \overset{*}{N}{}_l^m(t)$$

with uniquely defined coefficients p_l.

Proof. Let us consider an open interval $((i-1)h, ih)$. We have that

$$\operatorname{supp}(\overset{*}{N}{}_l^m(h;\cdot)) \cap ((i-1)h, ih) \neq \emptyset$$

if and only if $i - m \leq l \leq i-1$, i.e., for exactly m values of the index l. (This can be seen in Figure 1.32$_{/43}$, for example, with $i = 1$, interval $(0, h)$, $m = 4$, and $l = -3, -2, -1, 0$. The index l is shifted by 2 in Figure 1.32$_{/43}$ since the centered functions N_l^m are illustrated there.) By Theorem 2.5.6$_{/84}$ and Remark 2.5.7$_{/85}$, it follows that the set of functions

$$\{\overset{*}{N}{}_l^m(h;\cdot)|_{((i-1)h,ih)}\}_{i-m\leq l\leq i-1}$$

is linearly independent. Since their number is m and their linear span is contained in the space of polynomials on $((i-1)h, ih)$ of degree less than or equal to $m-1$ (which has dimension m), we conclude that they are a basis for this space. This follows from the fact that the number of functions in the power basis is exactly m. It follows next that for every $x \in \mathcal{B}_m(h\mathbb{Z})$, we have

$$x(t)|_{((i-1)h,ih)} = \sum_{i-m\leq l\leq i-1} p_l \overset{*}{N}{}_l^m(h,;t)$$

with uniquely defined coefficients p_l. Using that

$$\overset{*}{N}{}_{m,i}(h;t)|_{(ih,(i+1)h)} = (t/h - i)^{m-1}/(m-1)!$$

and that $D^{m-1}\overset{*}{N}_{m,i}(h; ih_+) = 1/h^{m-1}$, it is clear that we can choose a new coefficient p_i in a unique way so that

$$\left[D^{m-1} \left(x(t) - \sum_{i-m\le l\le i} p_l \overset{*}{N}{}_l^m(h;t) \right) \right]_{t=ih_+} = 0. \tag{2.66}$$

Note also that, since the function $x(t) - \sum_{i-m\le l\le i} p_l \overset{*}{N}{}_l^m(h;t)$ vanishes on the interval $((i-1)h, ih)$ and is in $C^{m-2}(\mathbb{R})$, we have

$$\left[D^{k} \left(x(t) - \sum_{i-m\le l\le i} p_l \overset{*}{N}{}_l^m(h;t) \right) \right]_{t=ih_+} = 0 \quad \text{for} \quad 0 \le k \le m-2. \tag{2.67}$$

By $(2.66)_{/86}$ and $(2.67)_{/86}$ and the fact that $x(t) - \sum_{i-m\le l\le i} p_l \overset{*}{N}{}_l^m(h;t)$ is a polynomial of degree at most $m-1$ on the interval $(ih, (i+1)h)$, we conclude that it vanishes also on that interval.

Repeating this process for an infinite sequence of adjacent intervals, the statement of the theorem follows. □

Corollary 2.5.10. *If $x \in \mathcal{B}_m(h\mathbb{Z})$ is zero on the intervals*

$$(ih, (i+1)h) \quad \textit{and} \quad ((i+k)h, (i+1+k)h), \quad \textit{where} \quad k \le m,$$

then it vanishes also on the intermediate intervals, i.e., $x(t) = 0$ on $(ih, (i+1+k)h)$. It follows that if $x \in \mathcal{B}_m(h\mathbb{Z})$ has its support in an interval, then the interval must have length at least mh.

Proof. Without loss of generality, we may assume that $i = 0$. By Theorem $2.5.9_{/85}$ the function x has the unique representation

$$x(t) = \sum_{l\in\mathbb{Z}} p_l \overset{*}{N}{}_l^m(h;t).$$

Then, by Theorem $2.5.6_{/84}$ we conclude that $p_l = 0$ for all l such that $\operatorname{supp}(\overset{*}{N}{}_l^m(h;\cdot)) \cap (0,h) \neq \emptyset$ and $\operatorname{supp}(\overset{*}{N}{}_l^m(h;\cdot)) \cap (kh, (k+1)h) \neq \emptyset$. This implies that $l = 0$ for $-m+1 \le l \le 0$ and $k+1-m \le l \le k$. If $k - m \le 0$, it follows that $l = 0$ for $-m+1 \le l \le k$, from which we conclude that $x(t) = 0$ for $(0, (k+1)h)$, and the proof is complete. □

2.5.5 Computation of exact values

If we are given control points for a spline curve at some subdivision level, we might want to compute exact positions of some or all points of the curve, or exact tangent vectors (the values of the control points provide only approximations). To obtain simple rules for doing so, the following recursion formula (which is due to de Boor [35] in the more general setting of nonuniform splines), is useful. Note that, in

contrast to the case of Section 2.2.1, the recursion here is in the order m with a fixed mesh.

Lemma 2.5.11. *For all $m \geq 2$ the following recursion formula is valid:*

$$\overset{*}{N}{}_0^m(h;t) = \frac{1}{m-1}((t/h)\overset{*}{N}{}_0^{m-1}(h;t) + (m - t/h)\overset{*}{N}{}_1^{m-1}(h;t)). \tag{2.68}$$

Proof. The functions $\overset{*}{N}{}_l^m(h;t)$ were defined in $(2.7)_{/55}$, $(2.9)_{/56}$, and $(2.12)_{/56}$, and from these definitions the statement of the theorem is true for $m = 2$. The functions are pulse functions when $m = 1$, and hat functions when $m = 2$, as illustrated in the centered case in Figure $2.1_{/57}$.

When $m \geq 3$, using that $\overset{*}{N}{}^m(h;t) = \overset{*}{N}{}^m(1;t/h)$, it is clearly sufficient to carry out the proof for $h = 1$.

By $(2.59)_{/81}$ we have $\overset{*}{N}{}_0^m(1;t) = \overset{*}{N}{}_0^m(t) = t^{m-1}/(m-1)!$ on the interval $(0,1)$ and $\overset{*}{N}{}_{m,1}(t) = (m-t)^{m-1}/(m-1)!$ on the interval $(m-1,m)$. It follows that the right-hand side minus the left-hand side in $(2.68)_{/87}$ vanishes on these intervals. Therefore, if we can prove that the right-hand side is in $C^{m-2}(\mathbb{R})$, then the conclusion of the lemma follows. This is due to the fact that the right-hand side minus the left-hand side would be a function in $\mathcal{B}_m(\mathbb{Z})$ with support on $(1, m-1)$ and therefore, by Corollary $2.5.10_{/86}$, would have to vanish.

Differentiating in the right side of $(2.68)_{/87}$, with $h = 1$, and using the product rule

$$D^{m-2}f(t)g(t) = \sum_{i=1}^{m-2} \binom{m-2}{i} D^i f(t) D^{m-2-i} g(t),$$

we get

$$\begin{aligned}
&D^{m-2}(t\overset{*}{N}{}_0^{m-1}(t) + (m-t)\overset{*}{N}{}_1^{m-1}(t)) \\
&= tD^{m-2}\overset{*}{N}{}_0^{m-1}(t) + (m-t)D^{m-2}\overset{*}{N}{}_1^{m-1}(t) \\
&\quad + (m-2)D^{m-3}\overset{*}{N}{}_0^{m-1}(t) - (m-2)D^{m-3}\overset{*}{N}{}_1^{m-1}(t).
\end{aligned}$$

Here the two last terms are in $C(\mathbb{R})$ and therefore it only remains to prove that the function

$$tD^{m-2}\overset{*}{N}{}_0^{m-1}(t) + (m-t)D^{m-2}\overset{*}{N}{}_1^{m-1}(t)$$

is continuous. Now, using $(2.59)_{/81}$ and $(2.60)_{/81}$ with $h = 1$ and m replaced by $m-1$, we get

$$\begin{aligned}
&\left[tD^{m-2}\overset{*}{N}{}_0^{m-1}(t) + (m-t)D^{m-2}\overset{*}{N}{}_1^{m-1}(t)\right]_{t=i-}^{t=i+} \\
&= i\left[D^{m-2}\overset{*}{N}{}_0^{m-1}(t)\right]_{t=i} + (m-i)\left[D^{m-2}\overset{*}{N}{}_1^{m-1}(t)\right]_{t=i-}^{t=i+} \\
&= (-1)^i\left[i\binom{m-1}{i} - (m-i)\binom{m-1}{i-1}\right] \\
&= (-1)^i\left[i\,\frac{(m-1)\cdots(m-i)}{i\,!} - (m-i)\frac{(m-1)\cdots(m-i+1)}{(i-1)\,!}\right] \\
&= 0,
\end{aligned}$$

and the proof is complete. □

For centered splines we have the following:

$$\begin{aligned} &N_0^m(h;t) \\ &= \frac{1}{m-1}((t/h + m/2)N_{-1/2}^{m-1}(h;t) + (m/2 - t/h)N_{1/2}^{m-1}(h;t)) \end{aligned}$$

or, for translated functions,

$$\begin{aligned} &N_l^m(h;t) \\ &= \frac{1}{m-1}\left\{((t/h - l) + m/2)N_{l-1/2}^{m-1}(h;t) + (m/2 - (t/h - l))N_{l+1/2}^{m-1}(h;t)\right\}. \end{aligned}$$

Rewriting the last expression slightly, we get the following theorem.

Theorem 2.5.12. *For all $m \geq 2$ the following recursion formula is valid:*

$$\begin{aligned} N_l^m(h;t) = &\left(\frac{1}{2} - \frac{l - 1/2}{m-1} + \frac{t}{(m-1)h}\right) N_{l-1/2}^{m-1}(h;t) \\ &+ \left(\frac{1}{2} + \frac{l + 1/2}{m-1} - \frac{t}{(m-1)h}\right) N_{l+1/2}^{m-1}(h;t). \end{aligned} \tag{2.69}$$

The recursion formula $(2.68)_{/87}$ of de Boor can be generalized to box splines. This is shown in Section 6.3.1. As in the univariate case it can be used for exact evaluation of nodal functions.

Computation of exact values, first method (de Boor method)

Now assume that the control points $\{p_l^m\}_{l\in\mathbb{Z}}$ for some spline curve have been computed up to a subdivision level h, so that

$$x(t) = \sum_{l\in\mathbb{Z}} p_l^m N_l^m(h;t).$$

Using $(2.69)_{/88}$ and shifting summation index, we obtain

$$x(t) = \sum_{l\in\mathbb{Z}} p_l^{m-1}(t) N_l^{m-1}(h;t)$$

with functions $p_l^{m-1}(t)$ given by

$$p_l^{m-1}(t) = p_{l+1/2}^m \left(\frac{1}{2} + \frac{(t/h - l)}{m-1}\right) + p_{l-1/2}^m \left(\frac{1}{2} - \frac{(t/h - l)}{m-1}\right), \tag{2.70}$$

where $l \in \mathbb{Z} + 1/2$, i.e., in the dual grid.

We note that the interpolation in $(2.70)_{/88}$ is convex, since $N_l^{m-1}(h;t) = 0$ for

$$\left|\frac{(t/h - l)}{m-1}\right| \geq 1/2.$$

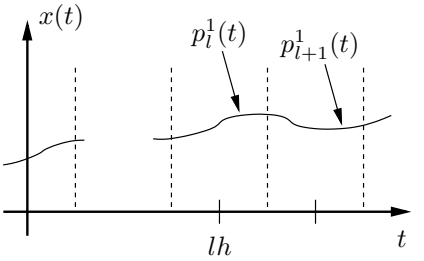


Figure 2.17. *Computation of exact values, first method.*

Interpolating recursively in $(2.70)_{/88}$ we obtain the representation

$$x(t) = \sum_{l\in\mathbb{Z}} p_l^1(t) N_l^1(h;t),$$

where we can read off the value of $x(t)$. Each function $p_l^1(\cdot)$ represents $x(t)$ exactly, on the interval $[lh - h/2, lh + h/2]$, as illustrated in Figure $2.17_{/89}$.

Computation of exact values, second method

Usually it is sufficient to evaluate $x(t)$ at points $t = ih/2$, i.e., for parameter values in the primal or the dual grid. In this case the method described below is easy to use. It is shown in Sections 6.1 and 6.2 that this method is the same as the method described in [156], sometimes referred to as "pushing a control point to its limit position."

For the primal grid we get

$$x(ih) = \sum_{l\in\mathbb{Z}} p_l N_0^m(h; ih - lh) = \sum_{l\in\mathbb{Z}} p_l N^m(1; i - l).$$

This may be rewritten as

$$x(ih) = \sum_{l\in\mathbb{Z}} p_{i-l} N^m(l),$$

where $N^m(t) \doteq N^m(1;t)$ is the spline centered at the origin with $h = 1$. For the dual grid we get similarly

$$x((i + 1/2)h) = \sum_{l\in\mathbb{Z}} p_{i-l} N^m(l + 1/2).$$

Therefore, in order to compute the values $x(ih)$ and $x((i + 1/2)h)$, we have to convolve the sequence $\{p_l\}_{l\in\mathbb{Z}}$ with the sequences $\{N^m(l)\}_{l\in\mathbb{Z}}$ and $\{N^m(l+1/2)\}_{l\in\mathbb{Z}}$, respectively. Now using $(2.69)_{/88}$ with $t = 0$ and l replaced by $-l/2$, we get

$$\begin{aligned} N^m(l/2) = &\left(\frac{1}{2} + \frac{l+1}{2(m-1)}\right) N^{m-1}\left(\frac{l+1}{2}\right) \\ &+ \left(\frac{1}{2} - \frac{l-1}{2(m-1)}\right) N^{m-1}\left(\frac{l-1}{2}\right) \quad . \end{aligned} \tag{2.71}$$

Table 2.1. *Nodal spline values for half-integer arguments.*

$l/2$	0	1/2	1	3/2	2	5/2
$N^2(l/2)$	1	1/2	0	0	0	0
$N^3(l/2)$	3/4	1/2	1/8	0	0	0
$N^4(l/2)$	2/3	23/48	1/6	1/48	0	0
$N^5(l/2)$	115/192	11/24	19/96	1/24	1/384	0

For $m = 2$ we have $N^2(-1/2) = N^2(1/2) = 1/2$, $N^2(0) = 1$, and we have $N^2(i/2) = 0$ for $|i| \geq 2$. Therefore, using $(2.71)_{/89}$ recursively for $m = 3, 4, \ldots$ we can compute the values $N^m(l/2)$ for arbitrary m and l (note the symmetry $N^m(l/2) = N^m(-l/2)$). In Table $2.1_{/90}$ some of these values are listed.

Note also that $\sum_{l\in\mathbb{Z}} N^m(l) = 1$ and $\sum_{l\in\mathbb{Z}} N^m(l + 1/2) = 1$, in accordance with $(2.63)_{/83}$ of Theorem $2.5.2_{/83}$.

Computation of exact tangent vectors

In order to compute exact values for tangent vectors, we simply apply the previous methods to the expression $(2.61)_{/81}$.

2.5.6 Application to the tensor-product-surface case

Most of the results developed in the previous subsections are directly applicable in the tensor-product case. Consider for example the results for differentiation. We first recall the definition of directional derivative.

Definition 2.5.13. *The directional derivative of a function $f : \mathbb{R}^2 \to \mathbb{R}$ in the direction $e \in \mathbb{R}^2$ is given by*

$$D_e f(y) = \lim_{\epsilon\to 0^+} \frac{f(y + \epsilon e) - f(y)}{\epsilon}. \tag{2.72}$$

According to $(2.46)_{/74}$, we have

$$x(u, v) = \sum_k \sum_l p_{k,l} N_k^m(h; u) N_l^m(h; v). \tag{2.73}$$

Now, the results of Section 2.5.1 can be used to compute partial derivatives: for example,

$$\frac{\partial x}{\partial u} = \frac{1}{h} \sum_{k\in\mathbb{Z}+1/2} \sum_{l\in\mathbb{Z}} (p_{k+1/2,l} - p_{k-1/2,l}) N_k^{m-1}(h; u) N_l^m(h; v), \tag{2.74}$$

and similarly for $\frac{\partial x}{\partial v}$. If $m > 2$, then $x \in C^1(\mathbb{R}^2)$, and we can compute arbitrary directional derivatives from

$$D_e x(u, v) = \alpha \frac{\partial x}{\partial u} + \beta \frac{\partial x}{\partial v},$$

where $e = (\alpha, \beta)^t$. This result is generalized to the box-spline case in Section 3.2.

For partition of unity we have

$$\sum_{l_1}\sum_{l_2} N^m_{l_1}(h;u)N^m_{l_2}(h;v) = \left(\sum_{l_1} N^m_{l_1}(h;u)\right)\left(\sum_{l_2} N^m(h;v)\right) = 1.$$

For global linear independence, we have the following. Suppose that

$$\sum_{l_1}\sum_{l_2} p_{l_1,l_2} N^m_{l_1}(h;u)N^m_{l_2}(h;v) = 0$$

for all u, v. Then, keeping v fixed, it follows that

$$\sum_{l_1}\left(\sum_{l_2} p_{l_1,l_2} N^m_{l_2}(h;v)\right) N^m_{l_1}(h;u) = 0,$$

which implies that $\sum_{l_2} p_{l_1,l_2} N^m_{l_2}(h;v) = 0$ for all l_1 (and all v). It follows that $p_{l_1,l_2} = 0$ for all l_1, l_2.

Computation of exact values as in Section 2.5.5 generalizes trivially to the nodal functions $N^m(h;u)N^m(h;v)$.

In fact, many of the results for the univariate case generalize far beyond the case of tensor-product B-splines, and we therefore postpone these generalizations until later sections. In particular, the result in Section 2.5.2 on partition of unity is discussed for box splines in Section 3.6 and more generally in Section 4.7. The extension of the result on linear independence, in Section 2.5.3, is also treated in the more general case of box splines, in Section 3.6. Similarly, the question of computation of exact values, in Section 2.5.5, is treated at quite a high level of generality in Chapter 6.

2.6 Additional comments

A good reference for generating functions is [72, Sec. 1.2]. Convolution and the Fourier transform are discussed in many textbooks: for the level of generality used here and in Section A.3, see the advanced texts [21, 54, 153].

Good references for B-splines and Non-Uniform Rational B-Splines (NURBS) are [30, 51, 127].

2.7 Exercises

1. The nodes of $N^m_l(h;t)$ are in $h\mathbb{Z} + (m-2)h/2$, while the nodes of $N^m_l(h/2;t)$ are in $\frac{h}{2}\mathbb{Z} + (m-2)h/4$. Draw the B-spline nodal functions, with their associated nodes, for the cases $m = 3$ and $m = 4$, and relate the position of the nodes to the grids in Figure 2.7$_{/68}$.

2. Repeat Example 2.2.5$_{/62}$ for the case $m = 3$. In particular, give the illustrations analogous to Figures 2.4$_{/63}$ and 2.5$_{/63}$ in the case $m = 3$.

3. Give the subdivision polynomial $p^\nu(h2^{-\nu}; z)$ corresponding to ν complete steps of the $LR(d)$ algorithm.

4. For $m = 3$ we have, in the notation illustrated in Figure 2.8$_{/70}$,

$$q^3_{2l+\frac{1}{2}} = \frac{1}{2}\left(p_l + \frac{1}{2}(p_l + p_{l+1})\right) = \frac{3}{4}p_l + \frac{1}{4}p_{l+1}$$

and

$$q^3_{2l+\frac{3}{2}} = \frac{1}{2}\left(\frac{1}{2}(p_l + p_{l+1}) + p_{l+1}\right) = \frac{1}{4}p_l + \frac{3}{4}p_{l+1},$$

which corresponds to (1.20)$_{/46}$. It corresponds also to applying the subdivision polynomial $\frac{z^{1/2}+z^{-1/2}}{2}$ to the linearly subdivided curve. Draw three one-dimensional stencils corresponding, respectively, to linear subdivision, to the subsequent smoothing, and to the overall Chaikin process. Do the same for the tensor-product-surface version of the Lane–Riesenfeld algorithm $LR(2\times2)$. The stencils in the latter case are tensor-product versions of the stencils for Chaikin's method, corresponding, for example, to the fact that the factor $\frac{z^{1/2}+z^{-1/2}}{2}$ is replaced by

$$\left(\frac{z_1^{1/2} + z_1^{-1/2}}{2}\right)\left(\frac{z_2^{1/2} + z_2^{-1/2}}{2}\right) = \begin{array}{lll} \frac{1}{4}z_1^{-1/2}z_2^{1/2} & + & \frac{1}{4}z_1^{1/2}z_2^{1/2} \\ + \frac{1}{4}z_1^{-1/2}z_2^{-1/2} & + & \frac{1}{4}z_1^{1/2}z_2^{-1/2} \end{array}$$

(the right-hand side is an ordinary sum with four terms written on two lines).

Repeat the above exercise for $m = 4$, omitting the stencils corresponding to the overall process.

5. In the case $d = 2$, there is a patch that can be associated with the dual node of the face in the middle of the grid in Figure 2.14$_{/79}$ (left). Draw the centered parametric domain corresponding to this patch.

6. State and graph the result of applying the subdivision process $LR(3)$, defined by (1.14)$_{/41}$, to the control points specified in Example 2.5.1$_{/82}$.

Chapter 3

Box-Spline Surfaces

In this chapter we consider a more general class of splines, the box splines, which appear in the second column of Figure 1.30$_{/33}$. This class of splines includes the tensor-product uniform B-splines as a special case, as well as other splines that are not tensor-product B-splines.

We first develop the theory for classical uncentered box-spline surfaces, which have the form

$$x(u,v) = \sum_l p_l N_l^*(he^m; u, v), \quad l = (l_1, l_2) \tag{3.1}$$

(with notation to be explained below). We then introduce centered versions

$$x(u,v) = \sum_l p_l N_l(he^m; u, v), \quad l = (l_1, l_2), \tag{3.2}$$

and subdivision algorithms that can be used in the evaluation of such surfaces. These algorithms are then modified to produce variant methods applicable in finite nonregular meshes without boundary in $\mathbb{R}^N$.

3.1 Notation and definitions

Let $\{f_i\}_{i=1}^m$ be the standard orthonormal basis in $\mathbb{R}^m$, so that $\{f_i\}_{i=1}^k$ spans $\mathbb{R}^k$, which is viewed as a k-dimensional subspace of $\mathbb{R}^m$.

The k-dimensional subsets (unit-cubes) $C^k \subset \mathbb{R}^m$ are defined by

$$C^k = \left\{ w \in \mathbb{R}^k : w = \sum_{i=1}^k c_i f_i,\ 0 \le c_i \le 1, i = 1, \ldots, k \right\} \subset \mathbb{R}^k \subseteq \mathbb{R}^m.$$

Suppose also that we are given a sequence of vectors $e^m = \{e_i\}_{i=1}^m$ in $\mathbb{R}^2$, $0 \neq e_i$, $i = 1, 2, \ldots, m$, with e_1 and e_2 linearly independent and such that all vectors in the sequence lie in an integer grid, i.e., $e_i \in \mathbb{Z}^2 \subset \mathbb{R}^2$, $i = 1, \ldots, m$. (In order to give a simple illustration, in one of the examples below, $\mathbb{R}^2$ is replaced by $\mathbb{R}$, and

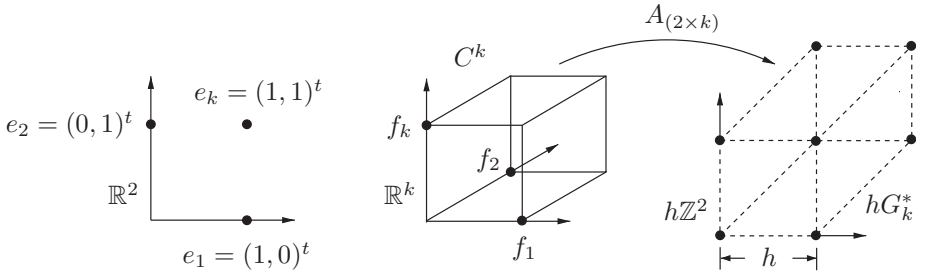


Figure 3.1. *Grid hG_k^* for uncentered box spline ($k = 3$).*

$\mathbb{Z}^2$ by $\mathbb{Z}$.) We allow several of the vectors in the sequence $\{e_i\}_{i=1}^m$ to be equal. Let $e^k = \{e_i\}_{i=1}^k$ denote the subsequence made up of the first k elements of e^m.

Next we define a linear mapping

$$A : \mathbb{R}^m \to \mathbb{R}^2$$

by requiring $Af_i = he_i, i = 1, \ldots, m$. The restriction of A to the subspace $\mathbb{R}^k$ is, with slight abuse of notation, also denoted by A, and the linear mapping is represented by a matrix $A = A_{(2\times k)}$. The parameter h denotes the grid-size.

For $2 < k \leq m$ the set of vertices of the cube C^k is mapped by A onto the set of grid points

$$hG_k^* = \left\{ y \in h\mathbb{Z}^2 : y = h\sum_{i=1}^k \epsilon_i e_i,\ \epsilon_i = 0, 1,\ i = 1, \ldots, k \right\}$$

(see Figure 3.1$_{/94}$), and the whole cube C^k is mapped onto the convex hull of the set hG_k^*:

$$\mathrm{conv}(hG_k^*) = \left\{ y \in \mathbb{R}^2 : y = h\sum_{i=1}^k c_i e_i,\ 0 \leq c_i \leq 1, i = 1, \ldots, k \right\}.$$

In the following definition, the unscaled version of hG_m^*, obtained by taking $h = 1$, is given a special name.

Definition 3.1.1. *The set $G_m^* \subset \mathbb{Z}^2$ defined by $G_m^* = \{y \in h\mathbb{Z}^2 : y = \sum_{i=1}^m \epsilon_i e_i$, $\epsilon_i = 0, 1,\ i = 1, \ldots, m\}$ is called the coefficient grid.*

The explanation for the name is that later in the chapter we introduce a subdivision polynomial for box-spline surfaces, and the coefficients of this polynomial are given on the set defined here as the coefficient grid.

Now, the *box-spline nodal function* $N^*(he^k; \cdot)$ associated with these sets of grid points in $\mathbb{R}^2$ is defined in Definition 3.1.2$_{/95}$. The nodal function depends on the sequence $e^k = \{e_i\}_{i=1}^k$, where $e_i \in \mathbb{Z}^2$, $i = 1, \ldots, k$.

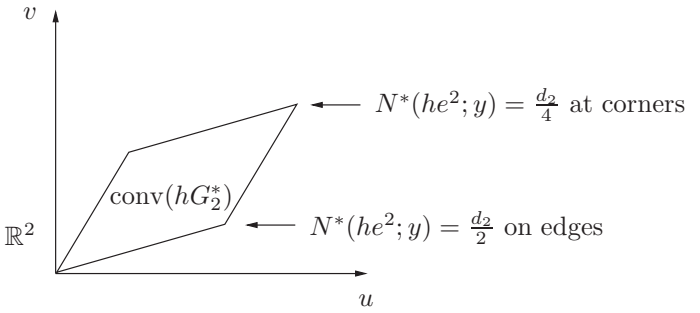


Figure 3.2. *The support of* $N^*(he^2;y)$.

Definition 3.1.2. *If* $2 < k \le m$, *then*

$$N^*(he^k;y) = d_k \mathrm{vol}_{k-2}\left(\{w \in \mathbb{R}^k : Aw = y\} \cap C^k\right). \tag{3.3}$$

If $k = 2$, *then*

$$N^*(he^2;y) = \begin{cases} d_2 & \text{if } y = h(c_1e_1 + c_2e_2) \text{ and } 0 < c_1, c_2 < 1, \\ d_2/2 & \text{if } y = h(c_1e_1 + c_2e_2) \text{ and } c_i = 0 \text{ or } 1 \text{ for exactly one index } i, \\ d_2/4 & \text{if } y = h(c_1e_1 + c_2e_2) \text{ and } c_i = 0 \text{ or } 1 \text{ for both indices } i, \\ 0 & \text{otherwise.} \end{cases} \tag{3.4}$$

Here $\{w \in \mathbb{R}^k : Aw = y\}$ *denotes the* $(k-2)$*-dimensional affine subset in* $\mathbb{R}^k$ *which is the inverse image of the point* $y \in \mathbb{R}^2$, *and* vol_{k-2} *denotes the* $(k-2)$*-dimensional Lebesgue measure. The constants* $d_k = d_k(h)$ *are normalization constants that are chosen so that*

$$\int_{\mathbb{R}^2} N^*(he^k;y)\,dy = h^2. \tag{3.5}$$

The definition in the case $k = 2$ will, after centering, be in accordance with $(2.8)_{/55}$; see Figure $3.2_{/95}$, where $y = (u,v)^t$.

Definition 3.1.3. *The constant* m *is called the (total) order of the box spline.*

The following remark is referred to later.

Remark 3.1.4. We have the following scaling relation:

$$N^*(e^k;y/h) = N^*(he^k;y). \tag{3.6}$$

To see this we note that

$$\begin{aligned} N^*(e^k;y/h) &= d_k(1)\,\mathrm{vol}_{k-2}(\{w : Aw = y/h\} \cap C^k) \\ &= d_k(1)\,\mathrm{vol}_{k-2}(\{w : hAw = y\} \cap C^k), \end{aligned}$$

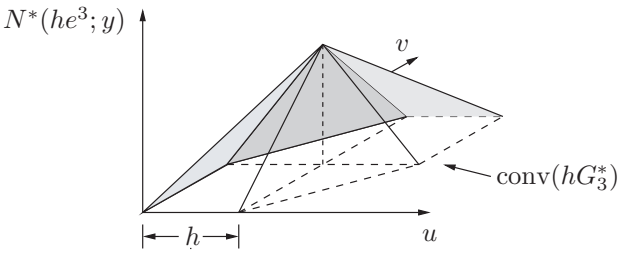


Figure 3.3. *An example of* $N^*(he^3; y)$.

where $Af_i = he_i$, $1 \le i \le k$. It follows from Definition 3.1.2$_{/95}$ that

$$N^*(e^k; y/h) = \frac{d_k(1)}{d_k(h)} N^*(he^k; y).$$

Also, from (3.5)$_{/95}$,

$$1 \int_{\mathbb{R}^2} N^*(e^k; y/h)dy = h^2 \int_{\mathbb{R}^2} N^*(e^k; y)dy = h^2,$$

from which we conclude that $d_k(1)/d_k(h) = 1$, and that (3.6)$_{/95}$ is true.

We also have the scaling relation

$$N^*(he^k; 2y) = N^*(he^k/2; y),$$

since from (3.6)$_{/95}$ we have

$$N^*(he^k; 2y) = N^*(e^k; 2y/h)$$

and

$$N^*(he^k/2; y) = N^*(e^k; 2y/h),$$

which gives the stated equality. ∎

Remark 3.1.5. The support of $N^*(he^k; \cdot)$ is contained in $\mathrm{conv}(hG_k^*)$. This follows directly from Definition 3.1.2$_{/95}$. In fact, the support is exactly the set $\mathrm{conv}(G_k^*)$, as stated in Corollary 3.2.2$_{/98}$. ∎

The graph of the function $N^*(he^k; \cdot)$ corresponding to the example of Figure 3.1$_{/94}$, with $k = 3$, is shown in Figure 3.3$_{/96}$, where again $y = (u, v)^t$. The set $\mathrm{conv}(hG_3^*)$ is also indicated there. We give a geometrical interpretation of (3.3)$_{/95}$, using two particular examples in the next section, but first we prove a basic theorem about $N^*(he^k; \cdot)$. It may be useful to glance ahead at Figures 3.4$_{/103}$ and 3.5$_{/104}$ to get an intuitive feeling for the definition in (3.3)$_{/95}$.

Remark 3.1.6. For the constant d_2 we have

$$d_2 = 1/|\det(e_1, e_2)| = 1/|\det(T)|,$$

where $T = (e_1, e_2)$ denotes the 2×2 matrix with the column vectors e_1 and e_2, since the modulus of the determinant $\det(T)$ is the area of the parallelogram $\mathrm{conv}(G_2^*) \subset \mathbb{R}^2$. ∎

The function $N^*(he^m; \cdot)$ is independent of the order of the vectors in the set $\{e_1, e_2, \ldots, e_m\}$; i.e., any permutation $\{e_{i_1}, e_{i_2}, \ldots, e_{i_m}\}$ will produce the same function $N^*(he^m; \cdot)$, provided that the first vectors e_{i_1} and e_{i_2} are linearly independent. This follows from Definition $3.1.2_{/95}$ for $k = m$. The intermediate functions $N^*(he^k; \cdot)$, $2 \le k < m$, are, however, in general different for this permutation—in fact, the grids hG_k^* themselves may be different. The intermediate functions actually depend on $\{e_1, \ldots, e_k\}$ rather than on hG_k^*; the former determines the latter, but the same grid may be generated by different sets of direction vectors defining different nodal functions.

The discussion in the next three sections is focused on the nodal functions $N^*(he^k; \cdot)$. Only in Section 3.5.2 do we return to the evaluation of the surface $x(u, v)$.

3.2 Properties of box-spline nodal functions

In the following it is shown that the functions $N^*(he^k; \cdot)$ can be obtained through repeated convolution by certain functions which are equal to 1 on intervals with length $h|e_i|$ and with the directions e_i in $\mathbb{R}^2$, in analogy with the presentation in Section 2.2 for uniform one-dimensional B-splines and for uniform two-dimensional tensor-product B-splines. Here, $|\cdot|$ denotes Euclidean distance on $\mathbb{R}^2$.

Construction of box-spline nodal functions by repeated convolution

We have the following theorem for the box-spline nodal functions.

Theorem 3.2.1. *For $2 < k \le m$ we have*

$$N^*(he^k; y) = \frac{1}{h} \int_0^h N^*(he^{k-1}; y - te_k)\, dt. \tag{3.7}$$

Recall that $N^*(he^2; y)$ is defined by $(3.4)_{/95}$.

Before proving the theorem we note that in $(3.7)_{/97}$ the one-dimensional integral is, for a fixed y, taken in the e_k-direction. If we make an orthogonal change of coordinates from $y = (u, v)^t$ in $\mathbb{R}^2$ to new coordinates $y' = (u', v')^t$ so that the e_k-direction coincides with the positive u'-axis, then $(3.7)_{/97}$ takes the form (after substituting $s = t|e_k|$)

$$N^*(he^k; u', v') = \frac{1}{h|e_k|} \int_0^{h|e_k|} N^*(he^{k-1}; u' - s, v')\, ds$$

or, equivalently,

$$N^*(he^k; u', v') = \frac{1}{h|e_k|} \int_{-\infty}^{\infty} N^*(he^{k-1}; u' - s, v') \overset{*}{N}{}^1(h|e_k|; s)\, ds,$$

where, in analogy with the notation of Section 2.2.1,

$$\overset{*}{N}{}^1(h|e_k|;t) = \begin{cases} 1 & \text{if } 0 < t < h|e_k|, \\ 1/2 & \text{if } t = 0 \text{ or } t = h|e_k|, \\ 0 & \text{otherwise.} \end{cases}$$

Consequently, the steps in the iteration suggested by $(3.7)_{/97}$ should be interpreted as a sequence of (one-dimensional) partial convolutions in the directions given by the vectors $e_3, e_4, \ldots, e_m$.

Proof. By Definition $3.1.2_{/95}$ we have for $k > 2$,

$$\begin{aligned} N^*(he^k;y) &= d_k \mathrm{vol}_{k-2}\left(\{w \in \mathbb{R}^k : Aw = y\} \cap C^k\right) \\ &= d_k \mathrm{vol}_{k-2}\left\{c \in C^k : h\sum_{i=1}^{k} c_i e_i = y\right\}. \end{aligned}$$

Since multiple integrals may be evaluated by repeated one-dimensional integration, we have

$$N^*(he^k;y) = d_k b_k \int_0^1 \mathrm{vol}_{k-3}\left\{c \in C^k : h\sum_{i=1}^{k-1} c_i e_i = y - hc_k e_k\right\} dc_k,$$

where b_k is some constant. Using Definition $3.1.2_{/95}$ and substituting $t = hc_k$, we have

$$\begin{aligned} N^*(he^k;y) &= (d_k b_k/d_{k-1}) \int_0^1 N^*(he^{k-1}; y - hc_k e_k)\, dc_k \qquad (3.8) \\ &= (d_k b_k/d_{k-1}) \frac{1}{h} \int_0^h N^*(he^{k-1}; y - te_k)\, dt. \end{aligned}$$

It remains to verify that $d_k b_k/d_{k-1} = 1$. Integrating with respect to y in $(3.8)_{/98}$ we get, by $(3.5)_{/95}$,

$$\begin{aligned} h^2 = \int_{\mathbb{R}^2} N^*(he^k;y)\, dy &= (d_k b_k/d_{k-1}) \int_{\mathbb{R}^2} \int_0^1 N^*(he^{k-1}; y - hc_k e_k)\, dc_k\, dy \\ &= (d_k b_k/d_{k-1}) \int_0^1 \int_{\mathbb{R}^2} N^*(he^{k-1}; y - hc_k e_k)\, dy\, dc_k \\ &= (d_k b_k/d_{k-1}) \int_{\mathbb{R}^2} N^*(he^{k-1};y)\, dy = d_k b_k/d_{k-1} h^2, \end{aligned}$$

and the proof is complete. □

Corollary 3.2.2. *The support of* $N^*(he^k;\cdot)$ *is* $\mathrm{conv}(hG_k^*)$.

Proof. This is trivially true for $k = 2$ and follows for general k by induction from the theorem. □

Derivatives of box-spline nodal functions

Certain of the results of Section 2.5 are now extended to the box-spline case. In particular, we generalize the results of Sections 2.5.1 and 2.5.6 concerning derivatives of nodal functions, and the theorem proved here is used explicitly in Chapter 5. Many of the results of Sections 2.5.2–2.5.5 can also be extended to the box-spline case [38], and even further (see, however, Exercise $4_{/141}$). We do not present such extensions for the specific case of box splines, but certain of the results are discussed later in quite a general context. See, for example, Section 6.1.1, where evaluation stencils are derived.

As in Section 2.5.6, we can use the following formula for the directional derivative $D_e f$, provided that $f \in C^1(\mathbb{R}^2)$:

$$D_e f(y) = e^t \nabla f(y) = \alpha \frac{\partial f}{\partial u} + \beta \frac{\partial f}{\partial v} \tag{3.9}$$

with $e = (\alpha, \beta)^t$ and $y = (u, v)^t$. Similarly, if e_1 and e_2 are two linearly independent vectors in $\mathbb{R}^2$, then any vector $e \in \mathbb{R}^2$ can be written as $e = \gamma e_1 + \delta e_2$, and we get

$$D_e f(y) = (\gamma e_1^t + \delta e_2^t)\nabla f(y) = \gamma D_{e_1} f(y) + \delta D_{e_2} f(y). \tag{3.10}$$

For the Fourier transform we have, from item 2 of Table A.$2_{/313}$, that

$$(D_e f)\hat{}\,(\omega) = i\omega^t e \hat{f}(\omega). \tag{3.11}$$

The following theorem concerns differentiation of box-spline nodal functions in the directions of the vectors in e^m.

Theorem 3.2.3. *For $m > 2$ we have*

$$\begin{aligned} D_{e_m} N^*(he^m; y) &= \frac{1}{h}\big(N^*(he^{m-1}; y) - N^*(he^{m-1}; y - he_m)\big) \\ &= \frac{1 - z^{2e_m}}{h} N^*(he^{m-1}; y), \end{aligned}$$

provided that the vectors e_1 and e_2 are linearly independent.

Proof. By Theorem 3.2.1$_{/97}$ we have

$$\begin{aligned} N^*(he^m; y) &= \frac{1}{h}\int_0^h N^*(he^{m-1}; y - te_m)\, dt \\ &= \frac{1}{h}\int_0^\infty N^*(he^{m-1}; y - te_m)\, dt - \frac{1}{h}\int_h^\infty N^*(he^{m-1}; y - te_m)\, dt \\ &= \frac{1}{h}\int_0^\infty \big(N^*(he^{m-1}; y - te_m) - N^*(he^{m-1}; y - (t+h)e_m)\big)\, dt \\ &= \frac{1}{h}\int_0^\infty g(y - te_m)\, dt, \end{aligned}$$

where $g(y) = N^*(he^{m-1}; y) - N^*(he^{m-1}; y - he_m)$. We then get, for $\epsilon > 0$, that

$$\frac{1}{\epsilon}\big(N^*(he^m; y + \epsilon e_m) - N^*(he^m; y)\big) = \frac{1}{h}\int_0^\infty \frac{g(y - (t-\epsilon)e_m) - g(y - te_m)}{\epsilon}\, dt$$
$$= \frac{1}{h}\int_{-\epsilon}^0 \frac{g(y - te_m)}{\epsilon}\, dt,$$

where the last equality is obtained by making the change of variables $t \leftrightarrow t - \epsilon$ in the first term in the numerator of the integrand. Now, taking limits as $\epsilon \to 0^+$, we get from continuity that

$$D_{e_m} N^*(he^m; y) = \frac{1}{h}\, g(y) = \frac{1}{h}(N^*(he^{m-1}; y) - N^*(he^{m-1}; y - he_m))$$

at all points of continuity for $N^*(he^{m-1}; y)$ and $N^*(he^{m-1}; y - he_m)$. □

The derivative in the theorem is taken in the direction of the last vector e^m. However, since the vectors of e^m can be permuted without changing $N^*(he^m; y)$, we conclude the following. Denote by $e^m_{(i)}$ the sequence obtained by deleting the vector e_i from e^m. Then we have

$$D_{e_i} N^*(he^m; y) = \frac{1}{h}\big(N^*(he^m_{(i)}; y) - N^*(he^m_{(i)}; y - he_i)\big),$$

provided that $e^m_{(i)}$ spans $\mathbb{R}^2$. We conclude also that if both $D_{e_i} N^*(he^m; y)$ and $D_{e_j} N^*(he^m; y)$ can be computed, with e_i and e_j linearly independent, then any directional derivative of $N^*(he^m; y)$ can be computed.

Exercise $1_{/141}$ asks for an explicit justification of the first step in the proof of Theorem $3.2.3_{/99}$.

Example 3.2.4. Tensor-product B-splines.

As a first example, we show that the ordinary tensor-product B-splines of bidegree d can be obtained by taking $d+1$ copies of $(1,0)^t$, and $d+1$ copies of $(0,1)^t$, for the sequence $\{e_i\}_{i=1}^m$, $m = 2(d+1)$, $d \geq 0$. Thus, if

$$\begin{array}{rcl} e_1 &=& (1,0)^t, \\ e_2 &=& (0,1)^t, \end{array} \qquad e_j = \begin{cases} (1,0)^t, & j = 3, \ldots, d+2, \\ (0,1)^t, & j = d+3, \ldots, 2d+2, \end{cases} \tag{3.12}$$

then the matrix T is equal to the 2×2 identity matrix, $\det(T) = 1$, and from $(3.4)_{/95}$ we have, with $y = (u,v)^t$,

$$N^*(he^2; u, v) = \overset{*}{N}{}^1(h; u)\overset{*}{N}{}^1(h; v). \tag{3.13}$$

This means in particular that $N^*(he^2; u, v) = 1$ if $0 < u, v < h$, and $N^*(he^2; u, v) = 0$ if either u or v is not in $[0, h]$.

The grid hG^*_m, with $m = 2(d+1)$, is equal to $\{(j_1 h, j_2 h)^t : 0 \leq j_1, j_2 \leq d+1\}$, and we therefore have $\mathrm{conv}(hG^*_m) = \{y = (u,v)^t : 0 \leq u, v \leq (d+1)h\}$.

We show now that $N^*(he^m; u, v)$, defined recursively by $(3.7)_{/97}$, is exactly equal to an uncentered version of the tensor-product B-spline introduced in Section 2.3. Thus, in the special case considered here, the box-spline nodal functions

reduce to tensor-product uniform B-spline basis functions. Centered box splines are introduced later in this section, and in the special case defined by $(3.12)_{/100}$, they reduce to exactly the basis functions of Section 2.3.

Using $(3.13)_{/100}$ and applying $(3.7)_{/97}$ d times with $e_k = (1,0)^t$, we have

$$N^*(he^{d+2}; u, v) = h^{-d} \left[\otimes_{k=1}^{d+1} \overset{*}{N}{}^1(h; u)\right] \overset{*}{N}{}^1(h; v).$$

Then, applying $(3.7)_{/97}$ d times with $e_k = (0,1)^t$, we obtain for $m = 2(d+1)$

$$N^*(he^m; u, v) = h^{-2d} \left[\otimes_{k=1}^{d+1} \overset{*}{N}{}^1(h; u)\right] \cdot \left[\otimes_{k=1}^{d+1} \overset{*}{N}{}^1(h; v)\right],$$

which is the uncentered tensor-product version of $(2.11)_{/56}$ if $d+1$ is substituted for m in $(2.11)_{/56}$.

Since $e_m = (0,1)^t$, Theorem $3.2.3_{/99}$ gives

$$\begin{aligned}\frac{\partial}{\partial v} N^*(he^m; y) &= \frac{(1 - z_2^2)}{h} N^*(he^{m-1}; y) \\ &= \frac{1}{h} \sum_{k,l \in \mathbb{Z}} (p_{k,l} - p_{k,l-1}) N_k^m(he^m; u) N_l^{m-1}(he^{m-1}; v),\end{aligned}$$

an uncentered derivative with respect to v, similar to $(2.74)_{/90}$. ∎

Example 3.2.5. Univariate B-splines.

The box-spline nodal function can be viewed as the intensity of the shadow of a box in $\mathbb{R}^m$ when projected into $\mathbb{R}^2$. In order to get some geometric intuition for this, we consider an example in the simpler case of a mapping

$$A : \mathbb{R}^m \to \mathbb{R}^1 \tag{3.14}$$

and mimic the development at the beginning of the chapter. In our simple example, the total order m is equal to $d+1$, and we choose $d+1$ identical vectors $1 = e_1 = \cdots = e_{d+1} \in \mathbb{R}^1$, in analogy to what we did in Example $3.2.4_{/100}$, where we had $A : \mathbb{R}^m \to \mathbb{R}^2$. In this new situation, the box spline reduces to an ordinary univariate B-spline, and the order of the box spline is equal to the order of the univariate B-spline.

Consider the case $m = 3$, which, according to what we have just claimed, should reduce to an ordinary univariate B-spline of degree 2. We have

$$f_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad f_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad f_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix},$$

$$A_{(1\times 3)} = h(e_1, e_2, e_3) = h(1,1,1),$$

$$hG_1^* = \{0, h\} \subset h\mathbb{Z},$$

$$\operatorname{conv}(hG_1^*) = [0, h] \subset \mathbb{R},$$

and

$$\overset{*}{N}(he^1;t) = \begin{cases} 1 & \text{if } 0 < t < h\ , \\ 1/2 & \text{if } t = 0 \text{ or } t = h, \\ 0 & \text{otherwise.} \end{cases}$$

Thus,

$$\int_{\mathbb{R}^1} \overset{*}{N}(he^1;t)dt = h,$$

in analogy with $(3.5)_{/95}$. In this simple case, $T_{(1\times 1)}$ has the single element 1, and $\det(T) = 1$.

In analogy with $(3.3)_{/95}$, we have

$$\overset{*}{N}(he^k;t) = d_k \mathrm{vol}_{k-1}\left(\{w \in \mathbb{R}^k : Aw = t\} \cap C^k\right),$$

where $A = A_{(1\times k)}$ contains k elements, each equal to h; t is a scalar; and $d_k = k^{-1/2}$, $1 < k \leq m = 3$. This analytically produces the result

$$\overset{*}{N}(he^k;t) = \frac{1}{h}\int_0^h \overset{*}{N}(he^{k-1};t-s)ds,$$

which is analogous to $(3.7)_{/97}$ and which corresponds exactly to $(2.10)_{/56}$ for univariate B-splines. The constant d_k is chosen so that

$$\int_{\mathbb{R}^1} \overset{*}{N}(he^k;t)dt = h.$$

For $k = 2$ we have

$$A_{(1\times 2)} = h(1,1),$$
$$hG_2^* = \{0, h, 2h\} \subset h\mathbb{Z},$$
$$\mathrm{conv}(hG_2^*) = [0, 2h] \subset \mathbb{R},$$

and

$$\overset{*}{N}(he^2;t) = \frac{1}{\sqrt{2}}\mathrm{vol}_1\left(\{w \in \mathbb{R}^2 : h(1,1)w = t\} \cap C^2\right),$$

where $t \in [0, 2h]$. See Figure $3.4_{/103}$.

Similarly, for $k = 3 = m$,

$$A_{(1\times 3)} = h(1,1,1),$$
$$hG_3^* = \{0, h, 2h, 3h\} \subset h\mathbb{Z},$$
$$\mathrm{conv}(hG_3^*) = [0, 3h] \subset \mathbb{R},$$

and

$$\overset{*}{N}(he^3;t) = \frac{1}{\sqrt{3}}\mathrm{vol}_2\left(\{w \in \mathbb{R}^3 : h(1,1,1)w = t\} \cap C^3\right),$$

where $t \in [0, 3h]$. See Figure $3.5_{/104}$. ∎

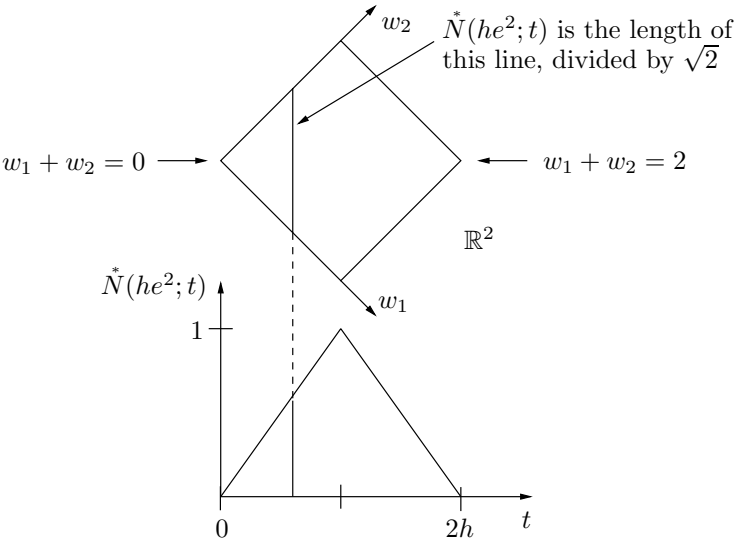


Figure 3.4. *Cross-section of a box in univariate case, $k = 2$.*

Example 3.2.6. Three-direction quartic box spline.

We next consider the box spline that corresponds to the basic method underlying the Loop method (Section 3.7.1). This spline involves three directions, each included with multiplicity two, so that $m = 6$:

$$e_1 = (1,0)^t,\ e_2 = (0,1)^t,\ e_3 = (1,0)^t,\ e_4 = (0,1)^t,\ e_5 = (1,1)^t,\ e_6 = (1,1)^t.$$

The matrix

$$A_{(2\times 6)} = h \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$$

has leading square matrix $hT_{(2\times 2)}$ equal to h times the identity, and $\det(T) = 1$. The grid hG_6^* is illustrated in Figure $3.6_{/104}$ (left), along with sequences of vectors from $\{he_1, \ldots, he_6\}$ that permit each extreme grid point to be reached. It is clear that each of the interior grid points illustrated can also be attained using a subset of the directions. Note also that the sequence of vectors to reach an interior grid point is not necessarily unique. In Figure $3.6_{/104}$ (right), the convex hull of hG_6^* is shown: it is a hexagonal region. By adding diagonal grid lines in the direction $(1,1)^t$, hG_6^* can be viewed as a triangular grid, and this is the view adopted in the Loop method.

The box-spline nodal function $N^*(he^6;\cdot)$, with support $\mathrm{conv}(hG_6^*)$, is shown in Figure $3.7_{/105}$. Theorem $3.2.9_{/107}$ shows that it is a piecewise polynomial of degree $m - 2 = 4$, which gives the spline its name ("quartic box spline"). The pieces on which the function is polynomial are exactly the triangles mentioned in the previous paragraph. Finally, Theorem $3.3.2_{/111}$ shows that it is a C^2 function of u and v. ∎

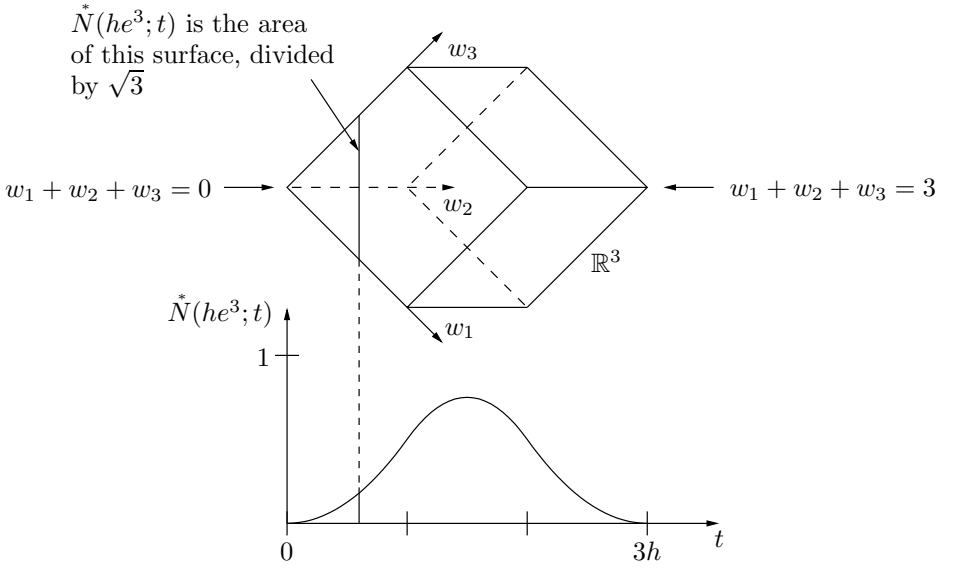


Figure 3.5. *Cross-section of a box in univariate case, $k = 3 = m$.*

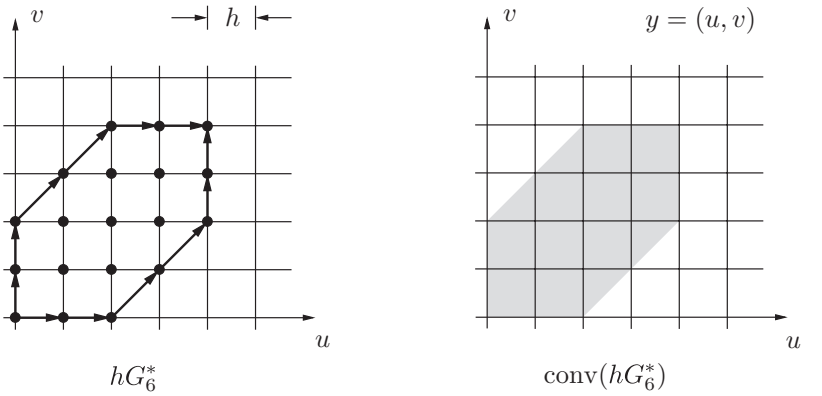


Figure 3.6. *Grid for three-direction quartic box spline.*

Example 3.2.7. Two four-direction box splines.
Our next example involves a choice of four-direction vectors, namely

$$e_1 = (1,0)^t,\ e_2 = (0,1)^t,\ e_3 = (1,1)^t,\ e_4 = (-1,1)^t,$$

which leads to two other well-known methods. If the vectors are chosen with multiplicity one ($m = 4$), the spline is a four-direction quadratic box spline. This box spline corresponds to the basic method underlying the Midedge method (or, more precisely, as shown in Section 3.7.2, the Midedge method applied twice in succession,

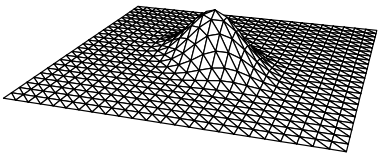


Figure 3.7. *Nodal function for three-direction quartic box spline.*

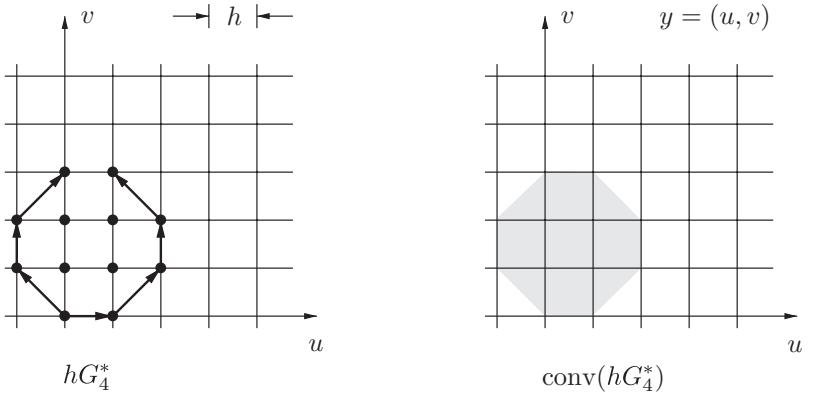


Figure 3.8. *Grid for Zwart–Powell element.*

denoted $\{\text{Midedge}\}^2$). If the vectors are chosen with multiplicity two ($m = 8$), the spline is a four-direction box spline corresponding to the basic method underlying 4-8 subdivision (again, see Section 3.7.2).

We begin with the case when the four vectors are chosen with multiplicity one, so that $m = 4$. The matrix

$$A_{(2\times 4)} = h \begin{bmatrix} 1 & 0 & 1 & -1 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

has leading square matrix $hT_{(2\times 2)}$ equal to h times the identity, and $\det(T) = 1$. The grid hG_4^* is illustrated in Figure $3.8_{/105}$ (left), along with sequences of vectors from $\{he_1, he_2, he_3, he_4\}$ that permit each extreme grid point to be attained. Again, it is clear that each of the interior grid points illustrated can be reached using a subset of the directions. In Figure $3.8_{/105}$ (right), the octagonal region $\mathrm{conv}(hG_4^*)$ is shown. The box-spline nodal function $N^*(he^4; \cdot)$, with support $\mathrm{conv}(hG_4^*)$, is shown in Figure $3.9_{/106}$. Theorem $3.2.9_{/107}$ below shows that $N^*(he^4; \cdot)$ is a piecewise polynomial of degree $m - 2 = 2$, i.e., piecewise quadratic. See Exercise $2_{/141}$. This function is called the *Zwart–Powell element* [38, p. 5]. Theorem $3.3.2_{/111}$ shows that it is a C^1 function of u and v.

Consider now the case when the four vectors are taken with multiplicity two, so that $m = 8$. The grid hG_8^* looks exactly like hG_4^* in Figure $3.8_{/105}$ except that

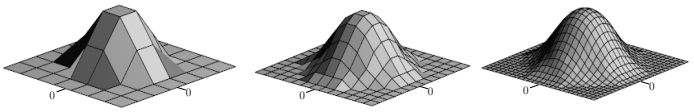


Figure 3.9. *Zwart–Powell element* ($\{Midedge\}^2$ *nodal function, regular case*).

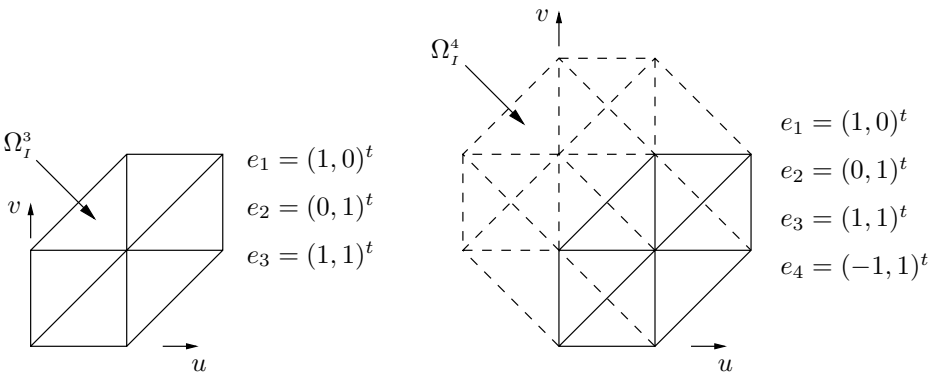


Figure 3.10. *Domains of polynomiality.*

(if h is held constant) the grid is magnified by a factor of two, the centre of the grid moves from $h(1/2, 3/2)^t$ to $h(1,3)^t$, and each arrow is replaced by a pair of arrows. Similarly, $\mathrm{conv}(hG_8^*)$ is the same as $\mathrm{conv}(hG_4^*)$, but magnified by a factor of two. The box-spline nodal function is a piecewise polynomial of degree $m - 2 = 6$, and Theorem $3.3.2_{/111}$ shows that it is a C^4 function of u and v. ∎

We show now that the box-spline nodal functions are piecewise polynomials in two variables.[34] The nature of the pieces on which the function $N^*(he^k; \cdot)$ is a polynomial can be quite complicated. The following example illustrates the conclusions of Theorem $3.2.9_{/107}$ in a simple case. The example is presented before the theorem because it gives strong motivation for the proof of the theorem: each time a new direction is added, the domains of polynomiality are refined.

Example 3.2.8. Domains of polynomiality.
Consider first a three-direction box spline with

$$e_1 = (1,0)^t,\ e_2 = (0,1)^t,\ e_3 = (1,1)^t,$$

where (in contrast to the quartic box spline of Example $3.2.6_{/103}$) the directions are included with a multiplicity of only one. According to Theorem $3.2.9_{/107}$, the domains of polynomiality of the corresponding function $N^*(he^3; \cdot)$ are as shown in Figure $3.10_{/106}$ (left), where the squares have side h. In fact, the function is a piecewise linear function with pieces defined as shown in the left of the figure. If we

now add a fourth direction, so that

$$e_1 = (1,0)^t,\ e_2 = (0,1)^t,\ e_3 = (1,1)^t,\ e_4 = (-1,1)^t,$$

we have the four-direction box spline of Example 3.2.7$_{/104}$, in the case where the directions are chosen with multiplicity one. The corresponding function $N^*(he^4;\cdot)$ has domains of polynomiality as shown in Figure 3.10$_{/106}$ (right): it is a piecewise quadratic. The use of dashed lines in Figure 3.10$_{/106}$ (right) is intended only to bring out the relationship between the two illustrations in the figure. ∎

We now turn to the proof of piecewise polynomiality for box splines. First let us introduce some notation.

We consider the two-dimensional subfaces of the cube C^k which are of the form

$$\left\{ w \in \mathbb{R}^k : w = c_{i_1} f_{i_1} + c_{i_2} f_{i_2} + \sum_i \varepsilon_i f_i \right\},$$

where $c_{i_1}, c_{i_2} \in (0,1)$, $i_1 \neq i_2$, $\varepsilon_i = 0$ or 1, and the last summation is over $1 \leq i \leq k$, $i \neq i_1, i_2$. The images under the mapping A are denoted by

$$F^k_{\mathbf{i},\varepsilon} = \left\{ y \in \mathbb{R}^2 : y = h\left(c_{i_1} e_{i_1} + c_{i_2} e_{i_2} + \sum_i \varepsilon_i e_i \right) \right\} \subset \mathbb{R}^2,$$

where $\mathbf{i} = (i_1, i_2)$ and $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_k)^t$. Next we form intersections

$$\Omega^k_I = \bigcap_{\mathbf{i},\varepsilon \in I} F^k_{\mathbf{i},\varepsilon}$$

which are nonempty, with maximal index set I, i.e., such that $\Omega^k_I \cap F^k_{\mathbf{i},\varepsilon} = \emptyset$ if $\mathbf{i}, \varepsilon \notin I$. The sets Ω^k_I are open, convex, and mutually disjoint, and their boundaries $\partial\Omega^k_I$ consist of finitely many line segments parallel to some vector e_i, $1 \leq i \leq k$. Further, they have the property that

$$\operatorname{supp}(N^*(he^k;\cdot)) = \bigcup_I \bar{\Omega}^k_I,$$

where the union is taken over all index sets I, and where $\bar{\Omega}$ denotes the topological closure of Ω. We now have the following theorem.

Theorem 3.2.9. *For $2 \leq k \leq m$ the functions $N^*(he^k;\cdot)$ are polynomials of degree at most $k-2$ on each subset Ω^k_I.*

Proof. There are two cases. In the first case, not all vectors $e_2, \ldots, e_m$ are parallel. Then, by reordering vectors we may assume that e_2 and e_3 are linearly independent. In the second case we assume that $e_2, \ldots, e_m$ are all parallel.

We carry out the proof in the first case by induction. For $k = 2$ the statement is true, since by definition, $N^*(he^2;\cdot)$ is a piecewise constant function with support

equal to the parallelogram defined by e_1 and e_2. Next assume that the statement is true for some value of $k-1 \geq 2$.

Every set $F^{k-1}_{\mathbf{i},\varepsilon}$ and $F^{k-1}_{\mathbf{i},\varepsilon} + he_k$ belongs to the class of sets of the form $F^k_{\mathbf{i},\varepsilon}$. Moreover, if a set Ω^k_I intersects some set $F^{k-1}_{\mathbf{i},\varepsilon}$ or $F^{k-1}_{\mathbf{i},\varepsilon} + he_k$, then Ω^k_I is a subset of some Ω^{k-1}_J or $\Omega^{k-1}_J + he_k$, respectively. Conversely, if Ω^k_I does not intersect any set $F^{k-1}_{\mathbf{i},\varepsilon}$, then Ω^k_I is disjoint from all sets Ω^{k-1}_J, and if Ω^k_I does not intersect any set $F^{k-1}_{\mathbf{i},\varepsilon} + he_k$, then Ω^k_I is disjoint from all sets $\Omega^{k-1}_J + he_k$.

There are now three possibilities as follows. For some J:

1. $\Omega^k_I \subset \Omega^{k-1}_J \cap (\Omega^{k-1}_J + he_k)$.
2. $\Omega^k_I \subset \Omega^{k-1}_J$ but $\Omega^k_I \cap (\Omega^{k-1}_J + he_k) = \emptyset$.
3. $\Omega^k_I \subset \Omega^{k-1}_J + he_k$ but $\Omega^k_I \cap \Omega^{k-1}_J = \emptyset$.

In the first case, we have for every $y \in \Omega^k_I$ that $y \in \Omega^{k-1}_J$, $y - he_k \in \Omega^{k-1}_J$, and according to Theorem $3.2.3_{/99}$ and the induction hypothesis, that

$$D_{e_k} N^*(he^k; y) = \frac{1}{h}(N^*(he^{k-1}; y) - N^*(he^{k-1}; y - he_k))$$

is a polynomial of degree at most $k-3$. In the second case, we have $y \in \Omega^{k-1}_J$, $y - he_k \notin \operatorname{supp}(N(he^{k-1}; \cdot))$, and that

$$D_{e_k} N^*(he^k; y) = \frac{1}{h} N^*(he^{k-1}; y)$$

is again a polynomial of degree at most $k-3$. In the third case, we have $y \in \Omega^{k-1}_J + he_k$, $y \notin \operatorname{supp}(N^*(he^{k-1}; \cdot))$, and that

$$D_{e_k} N^*(he^k; y) = -\frac{1}{h} N^*(he^{k-1}; y - he_k)$$

is a polynomial of degree at most $k-3$.

Consequently, writing $f(y) = N^*(he^k; y)$, we see that the directional derivative $D_{e_k} f(y)$ is a polynomial of degree at most $k-3$ on each set Ω^k_I. This is not sufficient to conclude that $f(y)$ is a polynomial of degree at most $k-2$ on each Ω^k_I. However, if we assume that not all vectors e_l, $2 \leq l \leq k$, are parallel, then we can reorder the vectors $e_1, e_2, \ldots, e_k$, and by repeating the argument above we may conclude that $D_{e_l} f(y)$ is a polynomial of degree at most $k-3$, with e_l not parallel to e_k (we can take $l = 2$ or 3). Now, since we can write $(1,0)^t = \gamma e_k + \delta e_l$, we get from $(3.10)_{/99}$

$$\frac{\partial f(y)}{\partial u} = (\gamma e_k^t + \delta e_l^t)\nabla f(y) = \gamma D_{e_k} f(y) + \delta D_{e_l} f(y),$$

i.e., $\frac{\partial f}{\partial u}$ is a polynomial of degree at most $k-3$ on each set Ω_I. We also get the same conclusion for $\frac{\partial f}{\partial v}$.

Now

$$\frac{\partial f}{\partial u} = \sum_l a_l u^{l_1} v^{l_2}$$

with $l_1 + l_2 \leq k - 3$ for all indices. This implies that

$$f(u,v) = \sum_k \frac{a_l}{l_1+1} u^{l_1+1} v^{l_2} + g(v), \tag{3.15}$$

where $g(v)$ is some function of v, which is differentiable on Ω_I since f is. If we differentiate with respect to v, we get

$$\frac{\partial f}{\partial v} = \sum_l \frac{l_2 a_l}{l_1+1} u^{l_1+1} v^{l_2-1} + g'(v).$$

Here the left-hand side is a polynomial, and therefore $g'(v)$ must be a polynomial in u and v, i.e., in v of degree at most $k-3$. Therefore, $g(v)$ is a polynomial of degree at most $k-2$, and it follows from $(3.15)_{/109}$ that $f(u,v)$ is a polynomial of degree at most $k-2$ on each set Ω_I.

By induction the statement is valid for all k, $2 \leq k \leq m$.

In the case that $e_2, \ldots, e_m$ are all parallel, we get that $N^*(he^m; y)$ is constant in the e_1-direction and a piecewise polynomial in the direction of e_2. The details are omitted. □

Remark* 3.2.10. We have characterized the open sets Ω_I^k of polynomiality as maximal, nonempty intersections of images under the mapping A of faces of the cube C^k. An alternative way of describing the domains of polynomiality is the following. Consider all the closed edges in the cube C^k. Their number is $m2^{m-1}$ and they have the form

$$\left\{ w \in \mathbb{R}^k : w = c_i f_i + \sum_j \varepsilon_j f_j \right\},$$

where $c_i \in [0,1]$, $\varepsilon_j = 0$ or 1, and where the last summation is over $1 \leq j \leq k$, $j \neq i$. Their images in $\mathbb{R}^2$ under the mapping A are the segments

$$E_{i,\varepsilon} = \left\{ y \in \mathbb{R}^2 : y = h\left(c_i e_i + \sum_j \varepsilon_j e_j \right) \right\}.$$

Now, if we form the closed union $\bigcup_{i,\varepsilon} E_{i,\varepsilon}$ of all these segments and take the complement with respect to $\mathrm{conv}(hG_k^*)$, i.e.,

$$\Omega = \mathrm{conv}(hG_k^*)^0 \setminus \bigcup_{i,\varepsilon} E_{i,\varepsilon},$$

then the open set Ω has the property that $\Omega = \bigcup_I \Omega_I$ and the sets Ω_I above are its connected components (the maximal open connected subsets). We may note that $\bigcup_{i,\varepsilon} E_{i,\varepsilon}$ is the shadow of the edge set of the cube C^k (the image under the mapping A).

Thus, in order to determine explicitly the domains of polynomiality, we must form all the segments $E_{i,\varepsilon}$ in $\mathbb{R}^2$ and identify the disjoint open sets that they bound.

In a general situation the number of segments can be $m2^{m-1}$ and the number of sets Ω_I becomes very large. In practice, however, there are often symmetries which reduce the complexity considerably. ∎

Remark* 3.2.11. In Theorem 3.2.9$_{/107}$ and Remark 3.2.10$_{/109}$ we characterized the domains Ω_I of polynomiality for a fixed nodal function $N^*(he^k; y)$. For a box-spline surface with representation

$$x(y) = \sum_{l \in \mathbb{Z}^2} p_l N(he^k; y - lh),$$

the partition of $\mathbb{R}^2$ into domains of polynomiality is finer, since each point $y \in \mathbb{R}^2$ is affected by several nodal functions $N^*(he^k; y - lh)$. Consequently, we have instead the following partitioning.

Consider the class of all closed segments

$$E_{i,\varepsilon} = \left\{ y \in \mathbb{R}^2 : y = h\left(c_i e_i + \sum_j \varepsilon_j e_j \right) \right\},$$

where, as before, $0 \le c_i \le 1$, but, in contrast to the case in Remark 3.2.10$_{/109}$, $\varepsilon_j \in \mathbb{Z}$, $1 \le j \le k$. Then the components of the open set

$$\mathbb{R}^2 \setminus \bigcup_{i,\varepsilon} E_{i,\varepsilon}$$

are the domains of polynomiality of the surface. The closed set $\bigcup_{i,\varepsilon} E_{i,\varepsilon}$ is the shadow of the edge set of all integer translations of the cube $C^k \subset \mathbb{R}^k$. Thus, in order to determine explicitly the domains of polynomiality, we must form all the segments $E_{i,\varepsilon}$ in $\mathbb{R}^2$ and identify the disjoint open sets which they bound. Also, for the union of all these sets we have

$$\bigcup_{i,\epsilon} E_{i,\epsilon} = h\mathcal{G}_m^* + \bigcup_{j=1}^{m} \{te_j : -\infty < t < \infty\},$$

where $\mathcal{G}_m^* = \{l = \sum_{j=1}^m \epsilon_j e_j, \epsilon_j = 0, 1\}$ is the additive subgroup of $\mathbb{Z}^2$ generated by $\{e_1, e_2, \ldots, e_m\}$. The closed set $\frac{1}{h} \bigcup_{i,\epsilon} E_{i,\epsilon}$ is therefore the union of all straight lines with directions $e_1, e_2, \ldots, e_m$, drawn through all points of $\mathcal{G}_m^*$. The set $\frac{1}{h} \bigcup_{i,\epsilon} E_{i,\epsilon}$ is also the shadow of the edge set of all integer translations of the cube C^k.

As mentioned in Remark 3.5.5$_{/118}$, for all the box splines we consider, $\mathcal{G}_m^* = \mathbb{Z}^2$.

For the case of the three-direction box spline, we get the partition shown in Figure 3.11$_{/111}$, as discussed in the next example. ∎

Example 3.2.12. Domain of polynomiality and associated patch.

Consider the quartic box spline of Example 3.2.6$_{/103}$, where the directions are included with multiplicity two (the spline corresponding to the basic method underlying Loop subdivision). A picture of the domains of polynomiality, for the nodal

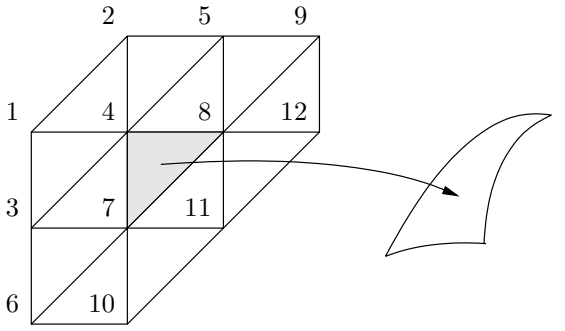


Figure 3.11. *Loop patch.*

functions as well as the box-spline surface, looks exactly like Figure $3.6_{/104}$ (right), but with each one of the shaded $h \times h$ squares divided by a diagonal line having slope $+1$.

Centering the support set shown in Figure $3.6_{/104}$ (right), we see that the only points making a contribution to a single domain of polynomiality are the 12 control points surrounding the triangular domain, and the polynomial surface corresponding to this domain can be viewed as a Loop "patch" (see Figure $3.11_{/111}$). This polynomial function is given explicitly in [149, Appendix A] as a linear combination of polynomials weighted by the 12 control points shown in Figure $3.11_{/111}$. Evaluation of the patch by conversion to Bézier form is described in [29, 71, 80] and [144, p. 21]. ∎

3.3 Continuity properties of box splines

We first define a parameter α that determines the regularity of the box spline.

Definition 3.3.1. *Let α be the smallest integer such that any subsequence of $\{e_i\}_{i=1}^m$ with α elements spans $\mathbb{R}^2$. Equivalently, $\alpha - 1$ is the largest number of vectors e_i that are parallel.*

With this definition, the following regularity result holds.

Theorem 3.3.2. *The box-spline nodal function $N^*(he^m; y)$ defined by the sequence e^m is in $C^{m-1-\alpha}(\mathbb{R}^2)$.*

The proof of this theorem appears in the Appendix (Section A.3). Note that, so far, we have not discussed box-spline subdivision methods at all, and neither here nor in Section A.3 is the convergence of such schemes discussed. For now, we know only that if we define box-spline nodal functions using repeated convolution in the directions e_i, then the regularity properties of Theorem $3.3.2_{/111}$ hold. Convergence of the associated subdivision schemes is discussed in Chapter 5.

Returning to the examples presented above, we now examine the continuity properties of the nodal functions defined. For the tensor-product B-spline case with order $m = 2(d+1)$, discussed in Example 3.2.4$_{/100}$, we have $\alpha = d+2$, and $m-1-\alpha = d-1$, so that the nodal function (in this case a B-spline basis function) has parametric continuity C^{d-1}. This coincides with well-known results for B-splines (see the last table of Section 1.3.2): for the biquadratic Lane–Riesenfeld method $LR(2\times 2)$ (which corresponds to the Doo–Sabin method), the basis function is C^1; for the bicubic Lane–Riesenfeld method $LR(3\times 3)$ (which corresponds to the Catmull–Clark method), the basis function is C^2; more generally, for the Repeated Averaging algorithm with bidegree $d \geq 1$, away from extraordinary vertices, the basis function is C^{d-1}.

For the three-direction quartic box spline, described in Example 3.2.6$_{/103}$, we had $m = 6$, and clearly $\alpha = 3$, so that (since $m-1-\alpha = 2$) the nodal function is C^2, as claimed in the example.

Finally, in Example 3.2.7$_{/104}$, for the four-direction box spline with directions included with multiplicity one, we had $m = 4$, and clearly $\alpha = 2$, so that (since $m-1-\alpha = 1$) the nodal function is C^1, as claimed. Similarly, for the four-direction box spline with directions included with multiplicity two, we had $m = 8$, and clearly $\alpha = 3$, so that (since $m-1-\alpha = 4$) the nodal function is C^4.

These last examples establish three more of the entries in the last table of Section 1.3.2: away from extraordinary points, the Loop method (based on the three-direction quartic box spline) is C^2, the $\{\text{Midedge}\}^2$ method is C^1, and 4-8 subdivision is C^4.

3.4 Box-spline subdivision polynomials

In Section 2.1 we introduced the one-dimensional translation operator z by $z^a f = f_{ah/2}$, i.e., by

$$(z^a f)(t) = f(t - ah/2),$$

where f is a function of one variable and where $a \in \mathbb{R}$. (Usually we have $a \in \mathbb{Z}$ or $a \in \mathbb{Z} + 1/2$.) For functions $f : \mathbb{R}^2 \to \mathbb{R}$ and $a \in \mathbb{Z}^2$ or $a \in \mathbb{Z}^2 + (1/2, 1/2)$, we may in the same way define

$$(z^a f)(y) = f(y - ah/2),$$

where $y = (u, v)^t$, $z = (z_1, z_2)$ and $z^a = z_1^{a_1} z_2^{a_2}$ (a notation somewhat reminiscent of inner-product notation).

In analogy with (2.2)$_{/54}$, for a polynomial $p(z) = \sum_a p_a z^a$ we define

$$p(z)f = \sum_a p_a (z^a f).$$

Now consider the sequence of functions $N^*(he^k; \cdot)$ which was analysed in the previous sections. Since $N^*(he^2; \cdot)$ is constant on the parallelogram

$$\{y \in \mathbb{R}^2 : y = h(c_1 e_1 + c_2 e_2), 0 \leq c_1, c_2 \leq 1\},$$

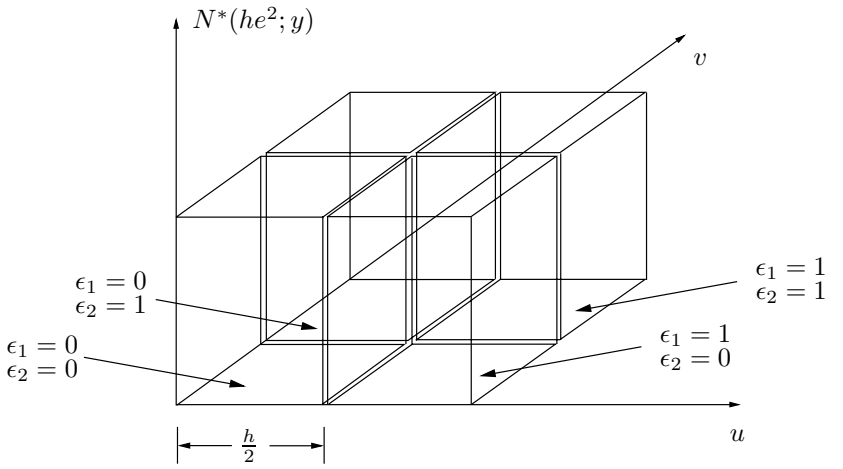


Figure 3.12. *Nodal function with total order* 2.

it is clear that

$$N^*(he^2; y) = \sum_{\substack{\epsilon_1=0,1\\ \epsilon_2=0,1}} N^*(he^2; 2(y - \epsilon_1 e_1 h/2 - \epsilon_2 e_2 h/2))$$
$$= \sum_{\substack{\epsilon_1=0,1\\ \epsilon_2=0,1}} z^{\epsilon_1 e_1} z^{\epsilon_2 e_2} N^*(he^2; 2y) = (1 + z^{e_1})(1 + z^{e_2}) N^*(he^2; 2y), \quad (3.16)$$

i.e., we have the subdivision rule

$$N^*(he^2; y) = (1 + z^{e_1})(1 + z^{e_2}) N^*(he^2; 2y). \quad (3.17)$$

See Figure 3.12$_{/113}$, which illustrates the case $e_1 = (1, 0)^t$, $e_2 = (0, 1)^t$, and compare with Figure 2.3$_{/60}$.

Lemma 3.4.1. *If for some polynomial $p(z)$, the function $N^*(he^{k-1}; y)$, with $k \geq 3$, satisfies the subdivision rule*

$$N^*(he^{k-1}; y) = p(z) N^*(he^{k-1}; 2y),$$

then $N^*(he^k; y)$ satisfies the subdivision rule

$$N^*(he^k; y) = \frac{1}{2}(1 + z^{e_k}) p(z) N^*(he^k; 2y).$$

The polynomial $p(z)$ in the statement of the lemma corresponds, in analogy with (2.23)$_{/61}$, to one of the factors in a subdivision polynomial that is given in the next theorem.

Proof. By the recursion formula (3.7)$_{/97}$ we have

$$N^*(he^k; 2y) = \frac{1}{h} \int_0^h N^*(he^{k-1}; 2y - te_k)\, dt = \frac{1}{h} \int_0^h N^*(he^{k-1}; 2(y - te_k/2))\, dt.$$

Operating with $(1 + z^{e_k})/2$, we get

$$\frac{1}{2}(1 + z^{e_k})N^*(he^k; 2y) = \frac{1}{2}N^*(he^k; 2y) + \frac{1}{2}N^*(he^k; 2(y - he_k/2))$$
$$= \frac{1}{2h}\int_0^h N^*(he^{k-1}; 2(y - te_k/2))\,dt + \frac{1}{2h}\int_0^h N^*(he^{k-1}; 2(y - he_k/2 - te_k/2))\,dt.$$

Substituting $s = t/2$ in the first integral, and $s = t/2 + h/2$ in the second, we get

$$\frac{1}{2}(1 + z^{e_k})N^*(he^k; 2y)$$
$$= \frac{1}{h}\int_0^{h/2} N^*(he^{k-1}; 2(y - se_k))\,ds + \frac{1}{h}\int_{h/2}^h N^*(he^{k-1}; 2(y - se_k))\,ds$$
$$= \frac{1}{h}\int_0^h N^*(he^{k-1}; 2(y - se_k))\,ds.$$

Now using the hypothesis that $p(z)N^*(he^{k-1}; 2y) = N^*(he^{k-1}; y)$, we get after operating with $p(z)$,

$$\frac{1}{2}(1 + z^{e_k})p(z)N^*(he^k; 2y) = \frac{1}{h}\int_0^h p(z)N^*(he^{k-1}; 2(y - se_k))\,ds$$
$$= \frac{1}{h}\int_0^h N^*(he^{k-1}; y - se_k)\,ds = N^*(he^k; y), \tag{3.18}$$

where we have used $(3.7)_{/97}$, and the proof is complete. □

We immediately obtain the following theorem.

Theorem 3.4.2.

$$N^*(he^m; y) = 4\prod_{i=1}^m \left(\frac{1 + z^{e_i}}{2}\right) N^*(he^m; 2y).$$

Recall that the coefficient grid G_m^* was defined in Definition $3.1.1_{/94}$.

Definition 3.4.3. *The polynomial*

$$s^*(z) = 4\prod_{i=1}^m \left(\frac{1 + z^{e_i}}{2}\right) = \sum_{k \in G_m^*} s_k^* z^k \tag{3.19}$$

is called the subdivision polynomial associated with the coefficient grid G_m^*. The second equality defines the notation s_k^* for the coefficients of the polynomial.

(Note that the meaning of the notation k has just changed. So far in this section, as in $(3.18)_{/114}$, it has served as an index for $he^k, k = 2, \ldots, m$. Now that we have derived $N^*(he^m; \cdot)$, however, we no longer need k for this purpose. On the

other hand, in the following pages it will be very convenient to use $k = (k_1, k_2)$ as we have done in $(3.19)_{/114}$, since it facilitates comparison with the previous B-spline development. Also, Definition $3.4.3_{/114}$ should be compared with $(2.20)_{/61}$. In the next section we introduce centered versions of s^* and N^*, which will be denoted without the star superscript.)

We thus have

$$N^*(he^m; y) = s^*(z)N^*(he^m/2; y), \tag{3.20}$$

since, from Remark $3.1.4_{/95}$, $N^*(he^m; 2y) = N^*(he^m/2; y)$. Writing $(3.20)_{/115}$ explicitly,

$$N^*(he^m; y) = \sum_{k \in G_m^*} s_k^* N^*(he^m/2; y - kh/2). \tag{3.21}$$

This should be compared with $(2.22)_{/61}$, although the latter equation corresponds more closely to the centered box-spline nodal functions introduced in the next section.

Example 3.4.4. Box-spline subdivision polynomials.

Referring to Example $3.2.4_{/100}$, we observe that for the tensor-product B-spline we have $d+1$ copies of $e_1 = (1,0)^t$, and $d+1$ copies of $e_2 = (0,1)^t$, which gives, according to $(3.19)_{/114}$ with $m = 2(d+1)$,

$$s^*(z) = 2\left(\frac{1+z_1}{2}\right)^{d+1} 2\left(\frac{1+z_2}{2}\right)^{d+1}.$$

If we multiply by $z_1^{-(d+1)/2} z_2^{-(d+1)/2}$ to allow for the fact that the spline has not yet been centered, we obtain $(2.51)_{/77}$. We return to this example in the next section and show in $(3.46)_{/124}$ that the correction for centering used here is correct.

For the *three-direction* quartic box spline with

$$e_1 = (1,0)^t,\ e_2 = (0,1)^t,\ e_3 = (1,0)^t,\ e_4 = (0,1)^t,\ e_5 = (1,1)^t,\ e_6 = (1,1)^t$$

(Example $3.2.6_{/103}$), we have

$$s^*(z) = \frac{1}{16}(1+z_1)^2(1+z_2)^2(1+z_1z_2)^2, \quad m = 6. \tag{3.22}$$

For the *four-direction* box splines (Example $3.2.7_{/104}$) with

$$e_1 = (1,0)^t,\ e_2 = (0,1)^t,\ e_3 = (1,1)^t,\ e_4 = (-1,1)^t,$$

we have

$$s^*(z) = \frac{1}{4}(1+z_1)(1+z_2)(1+z_1z_2)(1+z_1^{-1}z_2), \quad m = 4, \tag{3.23}$$

if the vectors are included with multiplicity one, and

$$s^*(z) = \frac{1}{64}(1+z_1)^2(1+z_2)^2(1+z_1z_2)^2(1+z_1^{-1}z_2)^2, \quad m = 8, \tag{3.24}$$

if the vectors are included with multiplicity two. ∎

3.5 Centered box-spline subdivision

We now introduce centered nodal functions, centered subdivision polynomials, and centered coefficient grids for box-spline subdivision.

3.5.1 Centered nodal functions and subdivision polynomials

The function $N^*(he^m;\cdot)$ defined in Section 3.1 has its support in $\mathrm{conv}(hG_m^*)$ and its centre at $h\sum_{i=1}^m e_i/2$. We now introduce the centered function $N(he^m;\cdot)$ having its centre at the origin, and then, in Definition 3.5.2$_{/117}$, we introduce the translates $N_l(he^m;\cdot)$.

Definition 3.5.1. *Let $\bar{e}=\sum_{i=1}^m e_i$ so that $\bar{e}/2$ is the centre of the coefficient grid G_m^* and let*

$$N(he^m;y)=N^*(he^m;y+(h/2)\bar{e})=z^{-\bar{e}}N^*(he^m;y),\quad m\geq 2. \tag{3.25}$$

Recursion formula for box-spline nodal functions

It follows from Definition 3.5.1$_{/116}$ that

$$N(he^m/2;y)=N^*(he^m/2;y+(h/4)\bar{e})=z^{-\bar{e}/2}N^*(he^m/2;y).$$

We then get

$$\begin{aligned}N(he^m;y)&=z^{-\bar{e}}N^*(he^m;y)=s^*(z)z^{-\bar{e}}N^*(he^m/2;y)\\&=s^*(z)z^{-\bar{e}}z^{\bar{e}/2}N(he^m/2;y)=z^{-\bar{e}/2}s^*(z)N(he^m/2;y)\end{aligned}$$

or

$$N(he^m;y)=s(z)N(he^m/2;y), \tag{3.26}$$

where $s(z)$ is the generalized polynomial

$$\begin{aligned}s(z)&=z^{-\bar{e}/2}s^*(z)=z^{-\bar{e}/2}4\prod_{i=1}^m\left(\frac{1+z^{e_i}}{2}\right)\\&=4\prod_{i=1}^m\left(\frac{z^{e_i/2}+z^{-e_i/2}}{2}\right)=\sum_{k\in G_m^*}s_k^*z^{k-\bar{e}/2}.\end{aligned} \tag{3.27}$$

Box-spline subdivision polynomial

Introducing the notation

$$G_m=G_m^*-\bar{e}/2 \tag{3.28}$$

for the centered coefficient grid, we have

$$s(z)=\sum_{k\in G_m}s_kz^k \tag{3.29}$$

with

$$s_k = s^*_{k+\bar{e}/2}.$$

This is called the (centered) box-spline subdivision polynomial, and $(2.20)_{/61}$ is a special case: the set G_m corresponds to the set $\{-m/2, -m/2+1, \ldots, m/2\}$ in the univariate B-spline case (see Proposition $2.2.3_{/60}$). We can rewrite $(3.26)_{/116}$ explicitly as

$$N(he^m; y) = \sum_{k \in G_m} s_k N(he^m/2; y - kh/2). \tag{3.30}$$

Definition 3.5.2. *As in Section* 2.2 *we introduce the translated box splines*

$$N_l(he^m; y) = N(he^m; y - lh),$$

where $l \in \mathbb{Z}^2 + \frac{1}{2}(\epsilon_1, \epsilon_2)^t, \epsilon_1, \epsilon_2 \in \{0, 1\}$.

This definition is analogous to $(2.13)_{/56}$. Observe that

$$N_l(he^m; y) = z^{2l} N(he^m; y)$$

and that $(3.26)_{/116}$ generalizes to

$$N_l(he^m; y) = z^{2l} s(z) N(he^m/2; y) = s(z) N_{2l}(he^m/2; y), \tag{3.31}$$

i.e.,

$$N_l(he^m; y) = s(z) N_{2l}(he^m/2; y) = \sum_{i \in G_m} s_i N_{i+2l}(he^m/2; y), \tag{3.32}$$

which should be compared with $(2.23)_{/61}$.

Example 3.5.3. Centered box-spline grids.

We illustrate the above ideas with the example of the three-direction quartic box spline, discussed in Example $3.2.6_{/103}$. In this case, $\bar{e} = \sum_{i=1}^{6} e_i = (4, 4)^t$. The centered grid $G_6 = G_6^* - (2, 2)^t$ is shown in Figure $3.13_{/118}$, which should be compared with Figure $3.6_{/104}$ (left). ∎

3.5.2 Recursion formulas for control points of box-spline surfaces

In analogy with Proposition $2.2.6_{/64}$, we seek recursion formulas for control points of box-spline surfaces. We begin by introducing the grids $\mathcal{G}_m^*$ and $\mathcal{G}_m$. The latter may be equal to $\mathcal{G}_m$, or it may correspond to a dual or semidual grid playing the role of the grid shown in green in Figure $1.17_{/20}$. The exact definitions are given now.

First we introduce the grid $\mathcal{G}_m^*$ by the following definition.

Definition 3.5.4.

$$\mathcal{G}_m^* = \left\{ l \in \mathbb{Z}^2 : l = \sum\nolimits_{i=1}^{m} k_i e_i, k_i \in \mathbb{Z}, 1 \le i \le m \right\} \subset \mathbb{Z}^2. \tag{3.33}$$

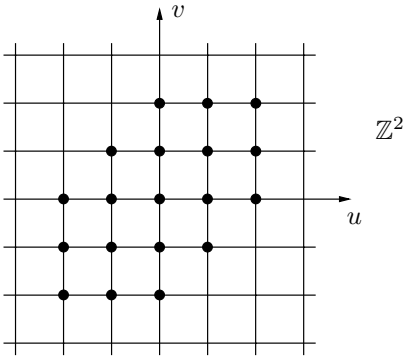


Figure 3.13. *The centered grid* $G_6 = G_6^* - (2,2)^t$.

We also define a centered grid by

$$\mathcal{G}_m = \mathcal{G}_m^* - \bar{e}/2, \tag{3.34}$$

i.e.,

$$\mathcal{G}_m = \{l \in \mathbb{Z}^2/2 : l + \bar{e}/2 \in \mathcal{G}_m^*\}. \tag{3.35}$$

Note that $\mathcal{G}_m \subseteq \mathbb{Z}^2$ *if* $\bar{e}/2 \in \mathbb{Z}^2$, *and that* $\mathcal{G}_m = \mathcal{G}_m^*$ *if* $\bar{e}/2 \in \mathcal{G}_m^*$.

Remark 3.5.5. By definition it is clear that $\mathcal{G}_m^*$ is an additive subgroup of $\mathbb{Z}^2$. Further, it can be shown that Theorem $4.5.1_{/172}$ implies that box-spline methods are affine invariant if and only if $\mathcal{G}_m^* = \mathbb{Z}^2$. Since it will be shown in Chapter 5 that affine invariance is necessary for convergence, in this book we consider only methods for which $\mathcal{G}_m^* = \mathbb{Z}^2$.

The equivalence between affine invariance and $\mathcal{G}_m^* = \mathbb{Z}^2$ follows easily. Theorem $4.5.1_{/172}$ shows that box-spline subdivision methods are affine invariant if and only if their subdivision polynomial $s^*(z_1, z_2)$ (see $(3.19)_{/114}$) satisfies the condition

$$s^*(1,1) = 4, s^*(1,-1) = s^*(-1,1) = s^*(-1,-1) = 0. \tag{3.36}$$

It is straightforward to verify that condition $(3.36)_{/118}$ is satisfied if and only if $\mathcal{G}_m^* = \mathbb{Z}^2$ (see $(3.33)_{/117}$) i.e., if and only if every $l \in \mathbb{Z}^2$ can be written as some integer combination $l = \sum_{i=1}^{m} k_i e_i$ with $k_i \in \mathbb{Z}$. In fact we get the following. Considering $(3.19)_{/114}$ we conclude that, for example, $s^*(-1,1) = 0$ if and only if the factor $1+(-1)^{e_{i1}}(+1)^{e_{i2}} = 1+(-1)^{e_{i1}} = 0$ for at least one i, i.e., e_{i1} must be odd for at least one i. Here, $e_i = (e_{i1}, e_{i2})^t$. Now, any $l = (l_1, l_2) \in \mathbb{Z}^2$ can be written as $(l_1, l_2) = \sum_i k_i(e_{i1}, e_{i2})$, so that $l_1 = \sum_i k_i e_{i1}$ with k_i an integer. Choosing l_1 odd, it is clear that not all e_{i1} can be even. Conversely, if all e_{i1} are even, then l_1 is always even. We argue similarly for $s^*(1,-1)$ and $s^*(-1,-1)$. ∎

Before proceeding we note the following general result for subgroups in $\mathbb{Z}^2$.

Proposition 3.5.6. *For every subgroup $\mathcal{G}^*$ of $\mathbb{Z}^2$, such that $\mathcal{G}^*$ contains at least two linearly independent vectors, there is a set $\{u_1, u_2\} \subset \mathcal{G}^*$ of generators with the property that every element $g \in \mathcal{G}^*$ can be written as*

$$g = k_1 u_1 + k_2 u_2$$

with uniquely defined integers k_1 and k_2. The generators u_1 and u_2 are not uniquely defined.

Proof. Consider the mapping

$$\mathcal{G}^* \times \mathcal{G}^* \ni (a, b) \mapsto \det(a, b) \in \mathbb{Z},$$

and let

$$k_{min} = \min\{\det(a, b) : \ a, b \in \mathcal{G}^*, \det(a, b) > 0\}. \tag{3.37}$$

Now let u_1 and u_2 be vectors in $\mathcal{G}^*$ with $k_{min} = \det(u_1, u_2)$. Let $g \in \mathcal{G}^*$ be arbitrary, and consider the equation

$$k_1 u_1 + k_2 u_2 = g \quad \text{or} \quad (u_1, u_2)(k_1, k_2)^t = g,$$

which, by Cramer's rule [157, p. 233], has the unique solution

$$k_1 = \det(g, u_2)/\det(u_1, u_2), \ k_2 = \det(u_1, g)/\det(u_1, u_2).$$

By the minimality property $(3.37)_{/119}$ it follows that k_1 and k_2 are integers.

In fact, consider the expression

$$\det(g - nu_1, u_2) = \det(g, u_2) - n\det(u_1, u_2) = \det(g, u_2) - nk$$

with $n \in \mathbb{Z}$. The right-hand side is equal to zero for some n; otherwise we could choose n such that

$$0 < \det(g - nu_1, u_2) < k,$$

contradicting the minimality property, and we have shown that k_1 is an integer. For k_2 the argument is similar. □

Remark 3.5.7. We note that $\bar{e}/2 = k_1 u_1/2 + k_2 u_2/2$ for some integers k_i, $i = 1, 2$, which implies that

$$\mathcal{G}_m = \mathcal{G}_m^* + (\epsilon_1 u_1 + \epsilon_2 u_2)/2,$$

where $\epsilon_i = 0$ if k_i is even and $\epsilon_i = 1$ if k_i is odd. Note that $\mathcal{G}_m = \mathcal{G}_m^* + (\epsilon_1 u_1 + \epsilon_2 u_2)/2$ is the dual grid of $\mathcal{G}_m^*$ if $\epsilon_1 = \epsilon_2 = 1$. If $(\epsilon_1, \epsilon_2) = (1, 0)$ or $(\epsilon_1, \epsilon_2) = (0, 1)$, then $\mathcal{G}_m$ is a semidual grid. ■

Control points for the refined mesh

The development now continues in analogy with the B-spline case. We examine here the relationship between the control points in the refined and unrefined meshes. Consider the parametric surface defined on the grid $\mathbb{Z}^2$ by

$$x(y) = x(u,v) = \sum_l p_l N_l(he^m; y), \tag{3.38}$$

where the summation is over $\mathbb{Z}^2$, or in the centered case over $\mathbb{Z}^2 + \bar{e}/2$, and where $p_l \in \mathbb{R}^N$ denote the control points. Using again that

$$N_l(he^m; y) = z^{2l} N(he^m; y),$$

this may be rewritten as

$$x(y) = \left(\sum_l p_l z^{2l} \right) N(he^m; y) = p(h; z^2) N(he^m; y),$$

where we have introduced the generalized polynomial $p(h; z) = \sum_l p_l z^l$.

We now wish to find the new control points q_k for the representation

$$x(y) = x(u,v) = \sum_k q_k N_k(he^m/2; y) = p(h/2; z) N(he^m/2; y) \tag{3.39}$$

on the refined grid, where $q_k \in \mathbb{R}^N$, and

$$p(h/2; z) = \sum_k q_k z^k. \tag{3.40}$$

By $(3.26)_{/116}$ we conclude that

$$x(y) = p(h; z^2) N(he^m; y) = p(h; z^2) s(z) N(he^m/2; y). \tag{3.41}$$

Then $(3.39)_{/120}$ is satisfied if we take

$$p(h/2; z) = s(z) p(h; z^2). \tag{3.42}$$

This may be compared with $(2.32)_{/65}$. Now equations $(3.29)_{/116}$ and $(3.42)_{/120}$ and the definition of $p(h; z)$ give

$$\sum_k q_k z^k = \sum_i s_i z^i \sum_l p_l z^{2l} = \sum_i \sum_l s_i p_l z^{i+2l}. \tag{3.43}$$

Here the coefficients s_i are initially defined for $i \in G_m \subset \mathbb{Z}^2 + \bar{e}/2$. By defining $s_i = 0$ outside G_m, the summation in i can be taken over $\mathbb{Z}^2 + \bar{e}/2$. Now taking $k = i + 2l$, i.e., $i = k - 2l$ in $(3.43)_{/120}$, we have

$$\sum_k q_k z^k = \sum_k \left(\sum_l s_{k-2l} p_l \right) z^k.$$

From this we conclude that

$$q_k = \sum_l s_{k-2l} p_l. \tag{3.44}$$

The summation in l ranges over $\mathbb{Z}^2$ or possibly $\mathbb{Z}^2 + \bar{e}/2$. The terms in the sum of $(3.44)_{/121}$ are nonvanishing only if s_{k-2l} is different from zero, i.e., only if $k - 2l \in G_m \subset \mathbb{Z}^2 + \bar{e}/2$. It follows that $q_k \neq 0$ only if $k \in \mathbb{Z}^2 + \bar{e}/2$. So on the refined level the index set (for k) is always $\mathbb{Z}^2 + \bar{e}/2$, regardless of whether for the initial level the vectors p_l have been chosen with $l \in \mathbb{Z}^2$ or $l \in \mathbb{Z}^2 + \bar{e}/2$ ($2l$ is always in $\mathbb{Z}^2$).

Formula $(3.44)_{/121}$ is the *subdivision equation* for the box-spline case. We note that the choice of the refined polynomial $\sum_k q_k z^k$ is not the only one possible. The choice is unique if an equality

$$\sum_k q_k N(he^m/2; y - kh/2) = \sum_k q'_k N(he^m/2; y - kh/2)$$

always implies that $q_k = q'_k$ for all k, i.e., if the translates of the nodal functions are linearly independent. Linear independence for box-spline nodal functions is treated in Section 3.6. This question of nonuniqueness is, however, no difficulty here. The choice in $(3.44)_{/121}$ will always be used and guarantees satisfactory convergence and smoothness properties for the process, as will be seen later in Chapter 5.

There are essentially four cases for the index k in $(3.44)_{/121}$; see Exercise $3_{/141}$. This limits the number of stencils needed to describe a method. Exercise $4_{/141}$ gives an example of nonuniqueness for the coefficients q_k.

As in the B-spline case, convergence of the q_k has to be proved. This is done in Chapter 5.

Averaging over dual and semidual grids

As for the univariate and tensor-product B-spline cases, the procedure defined by $(3.44)_{/121}$ and $(3.42)_{/120}$ with

$$s(z) = 4 \prod_{i=1}^{m} \left(\frac{z^{e_i/2} + z^{-e_i/2}}{2} \right) \tag{3.45}$$

can be implemented at each step as a sequence of averagings of control points over the refined version of the initial grid and its dual, and this is the generalized version of the alternation mechanism observed for the Lane–Riesenfeld algorithm. The averagings are in the directions of the vectors e_i, $1 \leq i \leq m$, defining the box spline. As before, extraction of a factor of $4p(h; z^2)$ in $(3.42)_{/120}$ is the upsampling. Multiplication by a factor

$$\frac{z^{e_i/2} + z^{-e_i/2}}{2}$$

corresponds to the interpolation or averaging $q^i_j = (q^{i-1}_{j+e_i/2} + q^{i-1}_{j-e_i/2})/2$, where $q^i(z) = \sum_j q^i_j z^j$, $0 \leq i \leq m$, is the sequence of intermediate generalized polynomials.

The final member of this sequence, $q^m(z) = p(h/2; z)$, is the generalized polynomial having control points over a refined grid with resolution $h/2$. After ν complete steps of the process, the control points are in a grid

$$2^{-\nu}\big(h\mathbb{Z}^2 + h(\epsilon_1/2, \epsilon_2/2)^t\big), \quad \epsilon_1, \epsilon_2 \in \{0, 1\}.$$

Example 3.5.8. Dual and semidual grids.

For the tensor-product B-spline of degree $d = 3$, we have $m = 2(d + 1) = 8$, $\bar{e} = (4, 4)^t$, and $\mathcal{G}_8^* = \mathbb{Z}^2$, with $u_1 = (1, 0)^t$ and $u_2 = (0, 1)^t$. Consequently, in Remark 3.5.7$_{/119}$, we have $k_1 = k_2 = 4$ and $\epsilon_1 = \epsilon_2 = 0$. Thus $\mathcal{G}_8 = \mathcal{G}_8^* = \mathbb{Z}^2$.

In contrast, for the tensor-product B-spline of degree $d = 2$, we have $m = 2(d + 1) = 6$, $\bar{e} = (3, 3)^t$, and $\mathcal{G}_6^* = \mathbb{Z}^2$, with $u_1 = (1, 0)^t$ and $u_2 = (0, 1)^t$. Consequently, in Remark 3.5.7$_{/119}$, we have $k_1 = k_2 = 3$ and $\epsilon_1 = \epsilon_2 = 1$. Thus $\mathcal{G}_6 = \mathcal{G}_6^* + (1/2, 1/2)^t = \mathbb{Z}^2 + (1/2, 1/2)^t$.

The tensor product, of a B-spline of even degree and a B-spline of odd degree, would lead to a semidual grid $\mathcal{G}_m$.

Note that in (3.32)$_{/117}$ the index i is in G_m, which means that $i+2l \in \mathcal{G}_m = \mathbb{Z}^2$ if $l \in \mathbb{Z}^2$. If one of the components of $i+2l$ is a half-integer, the nodal function N_{i+2l} in (3.32)$_{/117}$ is shifted by half-integer values in that component, corresponding to a shift of $h/4$. This is analogous to the comment in the solution to Exercise 2$_{/91}$ of Chapter 2 that, if m is odd, the functions on the right-hand side of (2.24)$_{/61}$ have their peak values in $\frac{h}{2}\mathbb{Z} + \frac{h}{4}$. ∎

Nodal-Function Computation principle

We can apply the Nodal-Function Computation principle, given in Section 2.2.4, in the case of box splines. Since we will be able to show that the averaging process we have just described is convergent (in a sense described in Definitions 4.7.1$_{/182}$ and 5.1.1$_{/193}$), we can find an approximation to $N_l(he^m; y)$ by applying the subdivision process to the scalar control points with $p_l = 1$, $l = (l_1, l_2)$, and $p_i = 0$, $i = (i_1, i_2) \neq (l_1, l_2)$. This is because substituting the values just mentioned into (3.38)$_{/120}$ gives $x(y) = N_l(he^m; y)$.

It is worthwhile to confirm for one or two simple examples that this process produces an approximate function with the correct support. For example, if the $LR(3 \times 3)$ algorithm is applied to the unit impulse with 1 at the origin in $h\mathbb{Z}^2$, the first subdivision step is easily seen to produce nonzero values in the domain $\{(u, v) : -h \leq u, v \leq h\}$ (compare with Figure 2.9$_{/73}$). The second subdivision step (with h replaced by $h/2$) then produces nonzero values in the domain $\{(u, v) : -3h/2 \leq u, v \leq 3h/2\}$, and, continuing, the νth subdivision step produces nonzero values in $\{(u, v) : -(2-2^{-\nu+1})h \leq u, v \leq (2-2^{-\nu+1})h\}$. The process thus converges to a function with support $\{(u, v) : -2h \leq u, v \leq 2h\}$, which is consistent with what we know about $\text{conv}(hG_m^*)$ from Example 3.2.4$_{/100}$ and Figure 2.2$_{/58}$ (bottom). Similarly, if Loop subdivision (Figure 1.29$_{/32}$) is applied to the unit impulse with 1 at the origin in $h\mathbb{Z}^2$, in the regular case, then at the νth iteration, nonzero values are obtained in the region

$$\{(u, v)^t : |(1, 0)(u, v)^t|,\ |(0, 1)(u, v)^t|,\ |(-1, 1)(u, v)^t| \leq h(2 - 2^{-\nu+1})\}$$

(see Figure 3.13$_{/118}$). Allowing for centering, the support of the nodal function is thus confirmed to be as given in Example 3.2.6$_{/103}$. See Exercises 5$_{/142}$ and 6$_{/142}$.

From the definition of a box spline, G_m is symmetric in the origin, and from (3.27)$_{/116}$ and (3.29)$_{/116}$ we have

$$4\prod_{i=1}^{m}\left(\frac{z^{e_i/2}+z^{-e_i/2}}{2}\right)=\sum_{j\in G_m} s_j z^j,$$

which implies that the subdivision mask s_j, $j \in G_m$, is symmetric in the origin, i.e., $s_j = s_{-j}$. From the Nodal-Function Computation principle, symmetry of the subdivision mask implies that the corresponding centered box-spline nodal function is also symmetric in the origin: $N(he^m; y) = N(he^m; -y)$.

Polynomial Coefficient principle

This principle, which was introduced in the B-spline case in Section 2.2.4, generalizes immediately to the box-spline case. Putting $p_{(0,0)} = 1$ and $p_l = 0$ for $l \neq (0,0)$ in (3.44)$_{/121}$, and applying one step of the subdivision process, produces $q_k = s_k$. See Exercise 7$_{/142}$.

Interpretation of polynomial factors

Any box spline involving the directions $(1,0)^t$ and $(0,1)^t$ will have the factor

$$\left(\frac{z_1^{1/2}+z_1^{-1/2}}{2}\right)\left(\frac{z_2^{1/2}+z_2^{-1/2}}{2}\right)=\frac{1}{4}(z_1^{1/2}+z_1^{-1/2})(z_2^{1/2}+z_2^{-1/2})$$

in its centered subdivision polynomial. This, along with the upsampling $4p(h; z^2)$, corresponds to constant subdivision (see (2.49)$_{/76}$), and any process for which the subdivision polynomial contains this factor can thus be viewed as having been built on top of constant subdivision. This includes in particular the tensor-product B-splines (Example 3.2.4$_{/100}$), three-direction box splines with vectors included with multiplicity one (Example 3.2.8$_{/106}$) or two (Example 3.2.6$_{/103}$), and four-direction box splines (Example 3.2.7$_{/104}$).

Similarly, any box spline involving the directions $(1,0)^t$ and $(0,1)^t$ with multiplicity two will involve the factor

$$\left(\frac{z_1^{1/2}+z_1^{-1/2}}{2}\right)^2\left(\frac{z_2^{1/2}+z_2^{-1/2}}{2}\right)^2,$$

which, again in conjunction with the upsampling $4p(h; z^2)$, corresponds to linear subdivision: compare (2.42)$_{/70}$ and (2.50)$_{/76}$. Thus, for example, tensor-product

B-splines (Example 3.2.4$_{/100}$) have centered subdivision polynomial

$$\begin{aligned} s(z) &= 4 \prod_{i=1}^{2(d+1)} \left(\frac{z^{e_i/2} + z^{-e_i/2}}{2} \right) \\ &= 4 \left[\left(\frac{z_1^{1/2} + z_1^{-1/2}}{2} \right) \left(\frac{z_2^{1/2} + z_2^{-1/2}}{2} \right) \right]^{d+1} \\ &= 2 \left(\frac{z_1^{1/2} + z_1^{-1/2}}{2} \right)^{d+1} 2 \left(\frac{z_2^{1/2} + z_2^{-1/2}}{2} \right)^{d+1} . \end{aligned} \tag{3.46}$$

Each of these last two factors corresponds exactly to (2.20)$_{/61}$.

Exercise 8$_{/142}$ gives a simple example, related to constant subdivision, which shows that the scaling of the vectors e_i is significant.

The Loop subdivision scheme (regular case)

The centered version of the three-direction quartic box spline (Example 3.5.3$_{/117}$) can be understood as an averaging process built on top of linear subdivision, and it can also be shown to produce Loop subdivision at ordinary vertices, as illustrated in Figure 1.29$_{/32}$. From Definition 3.5.1$_{/116}$ we have that $\bar{e}/2 = (2,2)^t$ (see Figure 3.13$_{/118}$). Consequently, $\mathcal{G}_6^* = \mathcal{G}_6 = \mathbb{Z}^2$ and $z^{-\bar{e}/2} = \frac{1}{z_1^2 z_2^2}$. Thus, it follows from (3.27)$_{/116}$ that

$$\begin{aligned} s(z) &= \frac{1}{16 z_1^2 z_2^2} (1+z_1)^2 (1+z_2)^2 (1+z_1 z_2)^2 \\ &= 2 \left(\frac{z_1^{1/2} + z_1^{-1/2}}{2} \right)^2 2 \left(\frac{z_2^{1/2} + z_2^{-1/2}}{2} \right)^2 \left(\frac{z_1^{1/2} z_2^{1/2} + z_1^{-1/2} z_2^{-1/2}}{2} \right)^2 . \end{aligned} \tag{3.47}$$

Again, the factor

$$2 \left(\frac{z_1^{1/2} + z_1^{-1/2}}{2} \right)^2 2 \left(\frac{z_2^{1/2} + z_2^{-1/2}}{2} \right)^2$$

corresponds to linear subdivision, while the remaining factor corresponds to averaging in the direction $(1,1)^t$ using the weights $\begin{bmatrix} 1/4 & 1/2 & 1/4 \end{bmatrix}$.

Observing (3.47)$_{/124}$, it is natural to redraw Figure 3.13$_{/118}$ as we have done in Figure 3.14$_{/125}$ using an oblique coordinate system. This and similar layouts are used later, for example when discussing $\sqrt{3}$-subdivision. The nodal function would in this case have a different form, due to a linear change of coordinates in the parametric domain. For our purposes here we continue using the rectangular grid.

Introducing diagonal lines in the rectangular grid, as shown in Figure 3.15$_{/125}$, a triangular grid is created, and binary subdivision induces a $pT4$ subdivision of this triangular grid. There are four kinds of vertices resulting from this subdivision: existing triangle vertices (E) and new edge points, which may occur on vertical

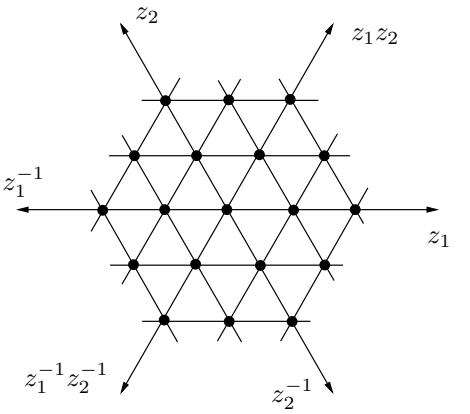


Figure 3.14. *The centered grid in triangular form.*

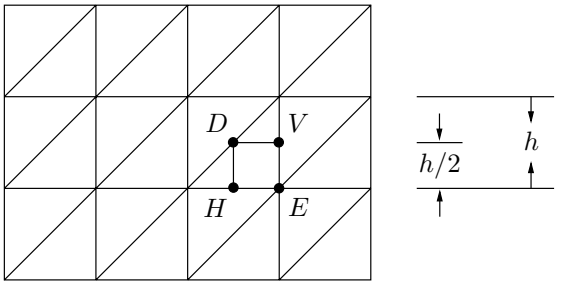


Figure 3.15. *A triangular grid from a rectangular grid.*

lines (V), horizontal lines (H), and diagonal lines (D). It is easily shown that vertices of these last three types all use the same weighted average of the control points at the preceding step if they are viewed in the context of the triangular grid, namely the weighted average illustrated in Figure 1.29$_{/32}$ (left). For a point of type V we have the linearly subdivided mesh shown in Figure 3.16$_{/126}$, and the smoothing that follows linear subdivision produces

$$\begin{aligned} V \leftarrow & \frac{1}{4}\left(p_{k,l} + p_{k+1,l}\right)/2 \\ & + \frac{1}{2}\left(p_{k+1,l} + p_{k+1,l+1}\right)/2 \\ & + \frac{1}{4}\left(p_{k+1,l+1} + p_{k+2,l+1}\right)/2 \\ & = \frac{1}{8}p_{k,l} + \frac{3}{8}p_{k+1,l} + \frac{3}{8}p_{k+1,l+1} + \frac{1}{8}p_{k+2,l+1}, \end{aligned}$$

which corresponds exactly to Figure 1.29$_{/32}$ (left). Averaging in the direction $(1,1)^t$ produces the same result in the other cases; see Exercise 9$_{/142}$.

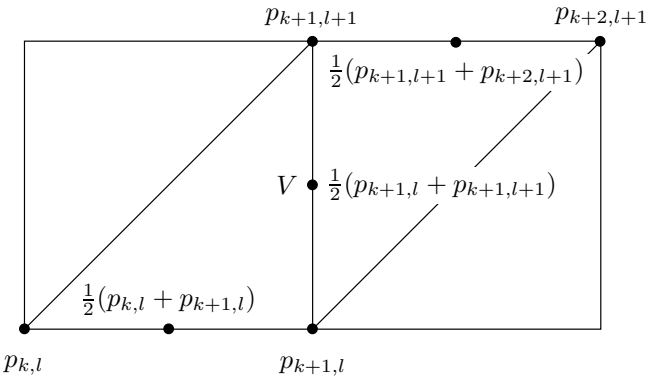


Figure 3.16. *Initial linear subdivision for a type-V vertex.*

There is an easier way to confirm that the subdivision polynomial in $(3.47)_{/124}$ corresponds to Loop subdivision in the regular case: we can apply one step of the Loop subdivision rules to the unit-impulse function in $\mathbb{Z}^2$ and compare what we get with the coefficients of the subdivision polynomial $(3.47)_{/124}$. This is the Polynomial Coefficient principle mentioned above. The procedure will be carried out for the Loop method in Section 3.7.

3.6 Partition of unity and linear independence for box splines

Definition 3.6.1. *Assume that we are given a function $F \in L^1(\mathbb{R}^2)$ with compact support. Then the translates $F(y - lh)$, $l \in \mathbb{Z}^2$, are said to give a partition of unity over $h\mathbb{Z}^2$ if*

$$\sum_{l \in \mathbb{Z}^2} F(y - lh) = 1 \quad \textit{for all } y \in \mathbb{R}^2.$$

They are said to be globally linearly independent if

$$\sum_{l \in \mathbb{Z}^2} p_l F(y - lh) = 0 \quad \textit{for all } y \in \mathbb{R}^2$$

implies that $p_l = 0$ for all $l \in \mathbb{Z}^2$, and globally linearly dependent otherwise. Further they are said to be locally linearly independent if for any open set $\Omega \subset \mathbb{R}^2$

$$\sum_{l \in \mathbb{Z}^2} p_l F(y - lh) = 0 \quad \textit{for all } y \in \Omega$$

implies that $p_l = 0$ for all $l \in \mathbb{Z}^2$ such that $\operatorname{supp}(F(y - lh)) \cap \Omega \neq \emptyset$, and locally linearly dependent otherwise.

In Remark 4.7.3$_{/183}$ it is observed that all subdivision methods which are affine invariant give a partition of unity, and as mentioned in Remark 3.5.5$_{/118}$, we consider only affine-invariant methods.

By definition, local linear independence implies global linear independence. It is a remarkable fact that for box splines, local linear independence is equivalent to global linear independence. This is a consequence of the following theorem, which is the main result of this section.

Theorem 3.6.2. *Assume that the vectors in e^m satisfy the condition*

$$\det(e_i, e_j) = \pm 1 \quad \textit{or} \quad \det(e_i, e_j) = 0 \tag{3.48}$$

for all i, j. Then the translates of $N^*(he^m; y - lh)$ of the nodal functions are locally linearly independent. Conversely, if condition (3.48)$_{/127}$ is not satisfied, then the translates are globally linearly dependent; i.e., there exist control vectors $p_l \in \mathbb{R}^N$, $l \in \mathbb{Z}^2$, not all equal to zero, such that

$$\sum_l p_l N^*(he^m; y - lh) = 0 \quad \textit{for all } y \in \Omega.$$

The requirement (3.48)$_{/127}$ is quite restrictive. Assume that (3.48)$_{/127}$ is satisfied and also, without loss of generality, that $\det(e_1, e_2) = 1$. Then every $a \in \mathbb{Z}^2$ can, in a unique way, be written as $a = k_1 e_1 + k_2 e_2$, where $k_1, k_2 \in \mathbb{Z}$. Then, if for some $i > 2$, $\det(e_i, e_1) = \pm 1$ and $\det(e_i, e_2) = \pm 1$, we get the following. Taking $e_i = k_{i1}e_1 + k_{i2}e_2$, we conclude that $\det(e_i, e_1) = \det(k_{i1}e_1 + k_{i2}e_2, e_1) = k_{i2} \det(e_2, e_1) = -k_{i2} = \pm 1$. Similarly, we have $\det(k_{i1}e_1 + k_{i2}e_2, e_2) = k_{i1}\det(e_1, e_2) = k_{i1} = \pm 1$, and we get that

$$e_i = k_{i1}e_1 + k_{i2}e_2 \quad \text{with} \quad k_{i1} = \pm 1, \quad k_{i2} = \pm 1.$$

Next, if for some $j > 2$ we have $\det(e_j, e_1) = \pm 1$ and $\det(e_j, e_2) = \pm 1$, we have

$$e_j = k_{j1}e_1 + k_{j2}e_2 \quad \text{with} \quad k_{j1} = \pm 1, \quad k_{j2} = \pm 1.$$

Now $\det(e_i, e_j) = \det(k_{i1}e_1 + k_{i2}e_2, k_{j1}e_1 + k_{j2}e_2) = k_{i1}k_{j2} - k_{i2}k_{j1} = \pm 1 \pm 1$. This expression is either ± 2 or 0, but since the values ± 2 are excluded we conclude that $\det(e_i, e_j) = 0$, i.e., $e_i = \pm e_j$.

Moreover, if $e_l = k_{l1}e_1 + k_{l2}e_2$, $l > 2$, satisfies $\det(e_l, e_1) = \pm 1$ and $\det(e_l, e_2) = 0$, we have $-k_{l2} = \pm 1$ and $k_{l1} = 0$. Similarly, if $\det(e_l, e_1) = 0$ and $\det(e_l, e_2) = \pm 1$, we have $k_{l2} = 0$ and $k_{l1} = \pm 1$. The case that $\det(e_l, e_1) = \det(e_l, e_2) = 0$ would give that $k_{l1} = k_{l2} = 0$ and $e_l = 0$ which is not possible. We have shown that there are numbers $\kappa_1, \kappa_2 = \pm 1$ such that every e_i, with $i > 2$, can be written as

$$\begin{aligned} &e_i = \pm(\kappa_1 e_1 + \kappa_2 e_2) \quad \text{or} \\ &e_i = \pm e_1 \quad \text{or} \quad e_i = \pm e_2. \end{aligned} \tag{3.49}$$

Typical box splines satisfying these conditions are

- the bilinear three-direction box spline with

$$e_1 = (1, 0)^t, \quad e_2 = (0, 1), \quad e_3 = (1, 1)^t;$$

- the quartic three-direction box spline with

$$e_1 = e_2 = (1,0)^t, \quad e_3 = e_4 = (0,1), \quad e_5 = e_6 = (1,1)^t;$$

- a tensor-product spline with

$$e_1 = e_2 = e_3 = e_4 = (1,0)^t, \quad e_5 = e_6 = e_7 = e_8 = (0,1)^t.$$

Further we note that for the four-direction box spline with

$$e_1 = (1,0)^t, \quad e_2 = (0,1)^t \quad e_3 = (1,1)^t, \quad e_4 = (-1,1)^t,$$

we have $\det(e_3, e_4) = 2$, and therefore the nodal functions are *linearly dependent*.

The proof of Theorem $3.6.2_{/127}$ can actually be simplified in the sense that when proving linear independence we may replace condition $(3.49)_{/127}$ either by the requirement

$$e_1 = (1,0)^t,\ e_2 = (0,1)^t,\ e_3 = (1,1)^t, \quad \text{and} \quad e_i = e_1, e_2 \text{ or } e_3 \text{ for } i > 3,$$

or by the requirement

$$e_1 = (1,0)^t,\ e_2 = (0,1)^t, \quad \text{and} \quad e_i = e_1 \text{ or } e_2 \text{ for } i > 3.$$

This can be understood in the following way.

Suppose that $h = 1$. We first note that replacing some vector $e_i \in e^m$ with $-e_i$ corresponds to a translation of the nodal function and does not affect the linear independence. In fact if $f^m = \{e_1, \ldots, e_{i-1}, -e_i, e_{i+1}, \ldots, e_m\}$, then

$$N(f^m; y) = N(e^m; y + e_i).$$

Also $N(e^m; y)$ is unchanged if the vectors of e^m are permutated. Therefore, if there is at least one vector $e_i \in e^m$ such that $e_i \neq \pm e_1$ or $\pm e_2$, then we can take the three first to be e_1, e_2, e_3, with $e_3 = e_1 \pm e_2$, and $e_i = +e_1$, $+e_2$ or $+e_3$ for $i > 3$. Further, in the case that $e_3 = e_1 - e_2$, we may reorder them as $e_1 - e_2, e_2, e_1$. Then we can rename them as $e_1 := e_1 - e_2$, $e_3 := e_1$, so that we have

$$e_1, e_2, e_3, \text{with } e_3 = e_1 + e_2, \text{ and } e_i = +e_1 \text{ or } +e_2 \text{ for } i > 3. \tag{3.50}$$

We conclude that when proving that condition $(3.48)_{/127}$ implies linear independence, it suffices to consider the case $(3.50)_{/128}$. Similarly, if for $i > 2$ we have $e_i = \pm e_1$ or $\pm e_2$, then we need only consider the case that $e_i = +e_1$ or $+e_2$.

We also note the following. Suppose that we have the case $(3.50)_{/128}$. Then the linear mapping

$$A : \mathbb{R}^2 \to \mathbb{R}^2$$

defined by $Ae_1 = (1,0)^t$ and $Ae_2 = (0,1)^t$ is a bijection when restricted to $\mathbb{Z}^2$. The sequence of vectors $f^m = Ae^m$ then generates a three-direction box spline of degree m with the standard directions $(1,0)^t$, $(0,1)^t$, and $(1,1)^t$. Further

$$N(e^m; y) = N(f^m; Ay) \quad \text{and} \quad N(e^m; A^{-1}y) = N(f^m; y),$$

and therefore we get

$$\sum_{l\in\mathbb{Z}^2} p_l N(e^m; y-l) = \sum_{l\in\mathbb{Z}^2} p_l N(f^m; Ay-Al) = \sum_{l'\in\mathbb{Z}^2} p_l N(f^m; y'-l'),$$

where $l' = Al$ and $y' = Ay$. This shows that the translates $N(e^m; y-l)$ and $N(f^m; y-l)$ are linearly independent simultaneously.

We now give the proof of Theorem 3.6.2$_{/127}$, but only in the case when $\Omega = \mathbb{R}^2$.

Proof. It is enough to consider the case of scalar control points.

First assume that (3.48)$_{/127}$ is not satisfied, i.e., $\det(e_i, e_j) \neq \pm 1$, or 0 for some $e_i, e_j \in e^m$. We will show that for some nonvanishing set $\{p_l \in \mathbb{R}^N\}_{l\in\mathbb{Z}^2}$ of control points we have

$$\sum_{l\in\mathbb{Z}^2} p_l N^*(e^m; y-l) = 0 \quad \text{for all} \quad y \in \mathbb{R}^2. \tag{3.51}$$

By reordering the vectors in e^m we may assume that $i = 1$ and $j = 2$. Now the additive subgroup

$$\mathcal{G}_2^* = \{k_1 e_1 + k_2 e_2 : k_1, k_2 \in \mathbb{Z}\} \subset \mathbb{Z}^2$$

(see(3.33)$_{/117}$) does not coincide with $\mathbb{Z}^2$, i.e., there exists an $l_0 \in \mathbb{Z}^2 \setminus \mathcal{G}_2^*$. It also follows that the sets $\mathcal{G}_2^*$ and $l_0 + \mathcal{G}_2^*$ are disjoint. We now define $p_l = 1$ if $l \in \mathcal{G}_2^*$, $p_l = -1$ if $l \in l_0 + \mathcal{G}_2^*$, and $p_l = 0$ otherwise. Observing that $N^*(e^2; y)$ is the characteristic function of the parallelogram spanned by e_1 and e_2, we conclude that

$$\sum_{l\in\mathcal{G}_2^*} N^*(e^2; y-l) = 1 \quad \text{for all} \quad y \in \mathbb{R}^2.$$

Similarly, it follows that $\sum_{l\in l_0+\mathcal{G}_2^*} N^*(e^2; y-l) = 1$ for all $y \in \mathbb{R}^2$, and hence (3.51)$_{/129}$ is satisfied, with e^m replaced by e^2. Using (3.7)$_{/97}$ repeatedly we conclude that (3.51)$_{/129}$ is true.

Next assume that (3.48)$_{/127}$ is satisfied. We will provide the proof of global linear independence only (i.e., the case $\Omega = \mathbb{R}^2$). For the more general result on local linear independence, see [38].

By the preceding discussion it suffices to consider the cases

$$e_1 = (1,0)^t,\ e_2 = (0,1)^t,\ e_3 = (1,1)^t,\ e_i = e_1, e_2 \text{ or } e_3 \text{ for } 4 \le i \le m \tag{3.52}$$

and

$$e_1 = (1,0)^t,\ e_2 = (0,1)^t,\ e_i = e_1 \text{ or } e_2 \text{ for } 3 \le i \le m. \tag{3.53}$$

We first carry out the proof when we have (3.52)$_{/129}$, or (3.53)$_{/129}$ with not all e_i equal for $3 \le i \le m$.

If $x(y) = \sum_{l\in\mathbb{Z}^2} p_l N^*(e^m; y-l)$, we conclude from Theorem 3.2.3$_{/99}$ that

$$\begin{aligned} D_{e_i} x(y) &= \sum_{l\in\mathbb{Z}^2} p_l (N^*(e^m_{(i)}; y-l) - N^*(e^m_{(i)}; y-l-e_i)) \\ &= \sum_{l\in\mathbb{Z}^2} (p_l - p_{l-e_i}) N^*(e^m_{(i)}; y-l), \end{aligned} \tag{3.54}$$

where $e^m_{(i)}$ denotes the sequence obtained from e^m by deleting e_i. We now argue by induction. For $m = 2$ the function $N^*(e^2; y)$ is the characteristic function for the unit square $[0,1] \times [0,1]$, and therefore $x(y) = p_l N^*(e^2; y - l) = p_l$ if $(u, v) \in (l_1, l_2) + (0,1) \times (0,1)$ (an open square). This shows that the statement above is true for $m = 2$. Next assume that it has been proved for some value $m = k - 1 \geq 2$. For $m = k$ we then obtain the following. If

$$x(y) = \sum_{l \in \mathbb{Z}^2} p_l N^*(e^k; y - l) = 0 \quad \text{for all} \quad y \in \mathbb{R}^2,$$

then, by $(3.54)_{/129}$ we have

$$D_{e_i} x(y) = \sum_{l \in \mathbb{Z}^2} (p_l - p_{l-e_i}) N^*(e^k_{(i)}; y - l) = 0 \quad \text{for all} \quad y \in \mathbb{R}^2$$

if $i = 1, 2$. By the induction hypothesis we conclude that $p_l - p_{l-e_i} = 0$ for all $l \in \mathbb{Z}^2$. It follows that $p_l = p_{l-(1,0)^t} = p_{l-(0,1)^t} = p$ for all l, where p is a constant. Therefore,

$$x(y) = \sum_{l \in \mathbb{Z}^2} p_l N^*(e^k; y - l) = p \sum_{l \in \mathbb{Z}^2} N^*(e^k; y - l) = p = 0,$$

i.e., $p_l = 0$ for all $l \in \mathbb{Z}^2$. By induction the argument is complete (for $\Omega = \mathbb{R}^2$).

The remaining case is that $e_1 = (1,0)^t$, $e_2 = (0,1)^t$, and either $e_i = e_1$ for $3 \leq i \leq m$ or $e_i = e_2$ for $3 \leq i \leq m$. Without loss of generality, we assume $e_i = e_2$. Then $N^*(e^m; y) = \overset{*}{N}{}^1(u) \overset{*}{N}{}^{m-1}(v)$, and

$$\sum_{l \in \mathbb{Z}^2} p_l N^*(e^m; y - l) = \sum_{l_1} \sum_{l_2} p_{l_1, l_2} \overset{*}{N}{}^1(u - l_1) \overset{*}{N}{}^{m-1}(v - l_2) = 0$$

implies that, for $u \in (0,1) + l_1$, with u and l_1 fixed,

$$\sum_{l} p_l N^*(e^m; y - l) = \sum_{l_1} \sum_{l_2} p_{l_1, l_2} \overset{*}{N}{}^{m-1}(v - l_2) = 0$$

for all v. By Theorem $2.5.5_{/84}$, $p_{l_1, l_2} = 0$ for all l_2. Since l_1 was arbitrary we conclude that $p_l = 0$ for all l, and the proof is complete (for $\Omega = \mathbb{R}^2$). □

3.7 Box-spline methods and variants for finite meshes

We have presented several standard box splines as examples: Example $3.2.4_{/100}$ showed that the tensor-product B-splines are a special kind of box splines, while Example $3.2.6_{/103}$ introduced the three-direction quartic box spline, and Example $3.2.7_{/104}$ introduced two four-direction box splines. We now show how the corresponding methods can be adapted for use in finite polyhedral meshes without boundary, both in the regular and nonregular case. This corresponds to the second column in Figure $1.30_{/33}$.

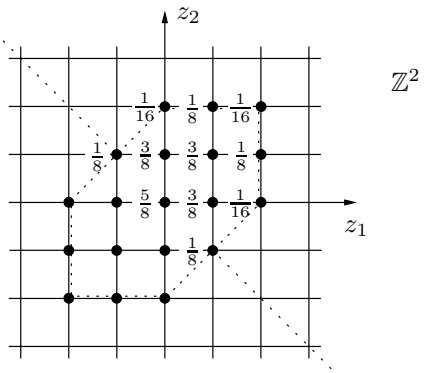


Figure 3.17. *Application of Loop method to unit-impulse function.*

3.7.1 The Loop method and its extension to higher orders

In Section 3.5 we mentioned that an easy way to verify that the Loop subdivision rules of Figure $1.29_{/32}$ give, in the regular case, the subdivision polynomial $(3.47)_{/124}$ corresponding to the three-direction quartic box spline, is to use the Polynomial Coefficient principle. If we substitute the value 1 for p_l when $l = (0, 0)$ in $(3.44)_{/121}$, and 0 for all other points $l \in \mathbb{Z}^2$, we obtain simply $q_k = s_k, k \in \mathcal{G}_m$. Thus, if we apply the Loop subdivision method to the unit-impulse function, for one step, we get the coefficients of the centered subdivision polynomial. Applying the Loop rules in Figure $1.29_{/32}$ to the unit-impulse function produces the values shown in Figure $3.17_{/131}$. (See Exercise $10_{/142}$. Values below the diagonal axis in the figure are given by reflections in the axis; other values not shown are equal to zero.) But from $(3.47)_{/124}$ we find

$$\begin{aligned} s(z) &= \frac{1}{16}(z_1 + 2 + z_1^{-1})(z_2 + 2 + z_2^{-1})(z_1 z_2 + 2 + z_1^{-1} z_2^{-1}) \\ &= \frac{5}{8} + \frac{3}{8}z_1 + \frac{3}{8}z_1^{-1} + \frac{1}{16}z_1^2 + \frac{1}{16}z_1^{-2} + \frac{1}{8}z_1^{-1}z_2 + \cdots , \end{aligned}$$

i.e., a polynomial with coefficients as shown in Figure $3.17_{/131}$.

In the nonregular case, the weights for an existing vertex, with $n \neq 6$, were given in Figure $1.29_{/32}$: a weight of $1 - w(n)$ is assigned to the vertex, and a weight of $w(n)/n$ is assigned to each of its neighbours. The formula for $w(n)$ is given in $(1.12)_{/32}$; it is chosen with an eye to the smoothness of the surface at the extraordinary point. This issue is discussed briefly in Section 5.8. The splitting schema is the $pT4$ split, which was described in Section 1.3.1.

We consider now an extension of the Loop method, in the nonregular case, that provides surfaces of higher continuity away from extraordinary points. This is a class of methods (parametrized by the total order $m = 6, 9, 12, \ldots$) that can be placed in the second column of Figure $1.30_{/33}$: the methods are analogous to the *RepeatedAveraging* algorithm. Recall that if m is the total order of a tensor-product B-spline, viewed as a box spline as in Example $3.4.4_{/115}$, the *RepeatedAveraging*

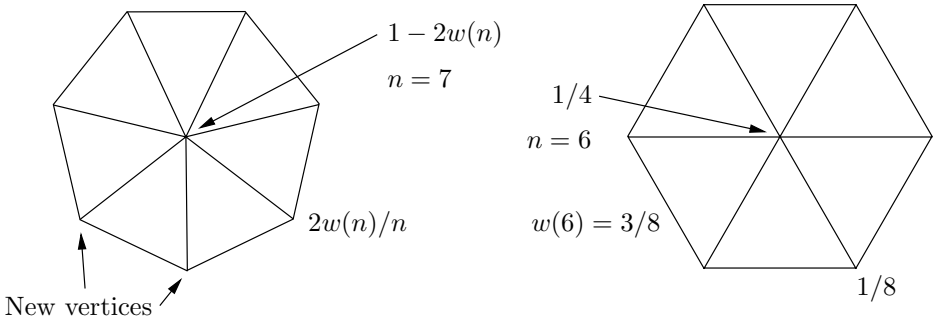


Figure 3.18. *Smoothing stencils for extended Loop method.*

algorithm generalized the $LR(3\times3)$ or Catmull–Clark process to the case of bidegree d, $d > 3$. (Catmull–Clark is the case $d = 3$, and $m = 2d + 2$.) The algorithm began with Linear Subdivision and performed $d - 1 = m/2 - 2$ subsequent averagings to obtain in the regular case a surface of bidegree d. This same idea was used in [151] to generalize the Loop method to the case of total order $m = 3j + 3, j = 1, 2, \ldots$. The algorithm begins by placing new vertices in the middle of each logical edge of the triangular mesh and assigning a control point to these new vertices that is equal to the average of the control points at each end of the edge. Then the algorithm applies $m/3 - 1$ smoothings using the stencil shown in Figure $3.18_{/132}$ (left), where $w(n)$ is defined as in $(1.12)_{/32}$. It is easy to check that if the process is initialized by assigning midpoint values to the control points, as described above, followed by a single smoothing ($m = 6$, $m/3 - 1 = 1$ smoothing) using the stencil shown in Figure $3.18_{/132}$ (left), then the Loop weights of Figure $1.29_{/32}$ are obtained for new edge points and for existing vertex points. See Exercise $11_{/142}$.

We now consider larger values of m. The class of methods just described, involving j smoothings and with total order $m = 3j + 3$, $j = 1, 2, \ldots$, in the regular case, can be included in column 2 of Figure $1.30_{/33}$. The Loop method is the case $j = 1$, $m = 6$.

This extension of the Loop method [151] was also listed in the table on page 37, where the continuity properties of the limit surface, away from extraordinary vertices, were given. To explain these properties, we examine the subdivision polynomial in the regular case. It is shown that the method involves inclusion of the three directions of the three-direction box spline $j + 1$ times, $j \geq 2$, rather than only twice, so that the exponent 2 appearing on each of the three right-hand factors of $(3.47)_{/124}$ is replaced by $j + 1$.

The initial assignment of new control points at edge midpoints corresponds to the polynomial

$$\begin{aligned} s_I(z) &= \frac{1}{2z_1z_2}(1 + z_1)(1 + z_2)(1 + z_1z_2) \\ &= 1 + \frac{1}{2}(z_1 + z_2 + z_1^{-1} + z_2^{-1} + z_1z_2 + z_1^{-1}z_2^{-1}), \end{aligned} \tag{3.55}$$

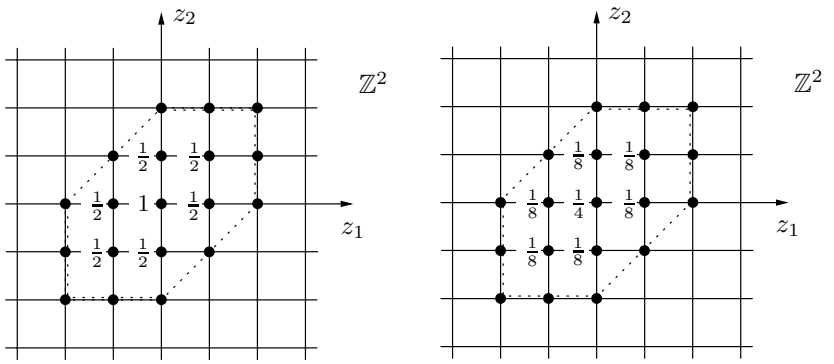


Figure 3.19. *Subdivision mask for extended Loop method.*

where the subscript $_I$ is intended to suggest "initial." (If we apply the initial midpoint averaging to the unit-impulse function, it produces the mask shown in Figure 3.19$_{/133}$ (left), and these are exactly the coefficients of the subdivision polynomial displayed in (3.55)$_{/132}$.) At each subsequent smoothing step we apply the stencil in Figure 3.18$_{/132}$ (right), which when applied to the unit-impulse function produces the mask shown in Figure 3.19$_{/133}$ (right). This corresponds to multiplying by the polynomial

$$s_S(z) = \frac{1}{8z_1z_2}(1+z_1)(1+z_2)(1+z_1z_2)$$

at each smoothing (the subscript S is intended to suggest "smoothing"). The product

$$s(z) = s_I(z)\,[s_S(z)]^j = \frac{4(1+z_1)^{j+1}(1+z_2)^{j+1}(1+z_1z_2)^{j+1}}{(8z_1z_2)^{j+1}} \tag{3.56}$$

is the subdivision polynomial corresponding to the complete process: we do smoothings for $j = 1, \ldots, m/3-1$.

Referring to Theorem 3.3.2$_{/111}$, the value of α there is equal to $m/3+1$, and the surface therefore has parametric continuity $(2m-6)/3$, $m = 6, 9, \ldots$. We might also envisage adding directions one at a time, rather than in groups of three as happens using the stencil in Figure 3.19$_{/133}$ (right). If this were done, the level of parametric continuity would be $\lfloor (2m-6)/3 \rfloor$, $m = 6, 7, 8, 9, \ldots$. See Exercises 12$_{/143}$ and 13$_{/143}$.

3.7.2 The Midedge and 4-8 subdivision methods

In this section we describe two other important box-spline methods, which are based on two different versions of four-direction box splines. Like the Loop method, these methods are listed in the second column of Figure 1.30$_{/33}$.

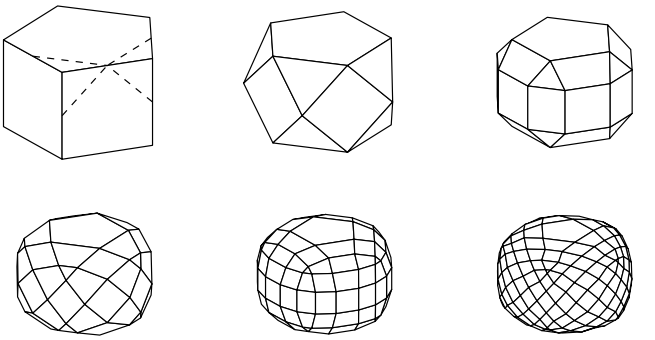


Figure 3.20. *An example of Midedge subdivision.*

The Midedge method

We begin by discussing the Midedge method [121] which, when applied twice in succession on a regular mesh, is equivalent to the four-direction quadratic box spline of Example 3.2.7$_{/104}$ and (3.23)$_{/115}$. Just as the Loop method reduced, on regular triangular meshes, to the three-direction quartic spline, the method $\{\text{Midedge}\}^2$ reduces, on regular quadrilateral meshes, to the four-direction quadratic box spline.

The Midedge method is defined as follows. Let (ℓ, ℓ') be an edge in the mesh; then, there are exactly four edges that share both a vertex and a face with this edge. The Midedge method joins the midpoints of each of these four edges to the midpoint of (ℓ, ℓ'), computes the control point at each new vertex as the average of the points at adjacent vertices, and discards the old mesh. Thus, after one step, the valence of all mesh points is four; see Figure 3.20$_{/134}$, which is similar to [121, Fig. 1].

The Midedge method, as just defined, is applicable to arbitrary locally planar meshes. In contrast to other methods, such as the Loop method, the Midedge method can be expressed without reference to the number of edges associated with a vertex or face.

We define extraordinary faces and vertices as they were defined for the $dQ4$ splitting, i.e., faces or vertices with other than four incident edges. In fact, we can view the $\{\text{Midedge}\}^2$ process as one that uses the $dQ4$ schema, although the method was not originally presented in this way. Referring to the outermost polygon in Figure 3.21$_{/135}$ (a pentagon is used to illustrate, i.e., $e = 5$), $dQ4$ begins with linear subdivision (dashed lines), followed by taking the dual (the vertices of the dual mesh are indicated by the black squares, as in Figure 1.16$_{/19}$, right). This can be compared with the results of applying $\{\text{Midedge}\}^2$: the smallest pentagon in Figure 3.21$_{/135}$ corresponds both to the edges in the mesh produced by $\{\text{Midedge}\}^2$ and to the edges of the dual mesh produced by $dQ4$.

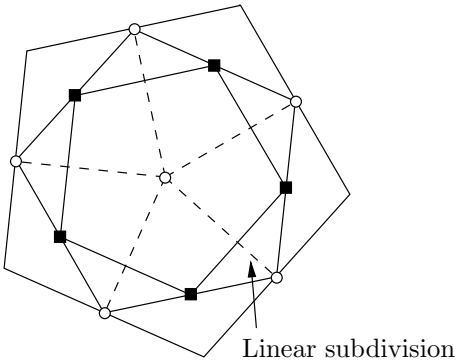


Figure 3.21. *Splitting for two steps of the Midedge method is dQ4.*

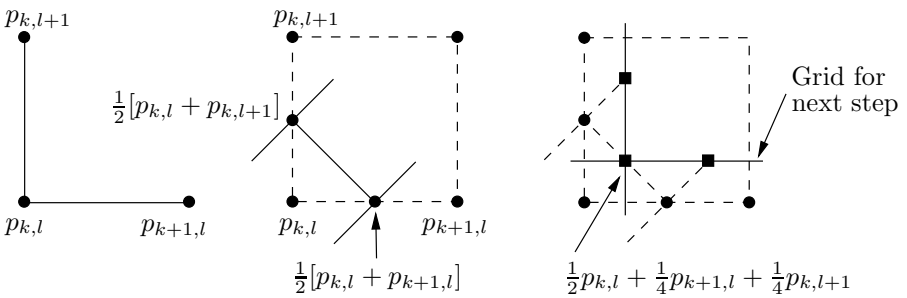


Figure 3.22. *Two steps of the Midedge method.*

Consequently, in terms of the topology of the subdivided mesh, the $\{\text{Midedge}\}^2$ method behaves in exactly the same way as a method based on $dQ4$ splitting, as described in Section 1.2.2.

For a regular face, as illustrated in Figure $3.22_{/135}$ (left), we obtain the new points illustrated in Figure $3.22_{/135}$ (middle) after one application of the Midedge method (the discarded mesh is shown with a dashed line). Continuing, the second application of the Midedge method is illustrated in Figure $3.22_{/135}$ (right), along with the values of the control points associated with the new vertices. We can view these new vertices, illustrated by black squares in the figure, as corresponding to nodes in the dual mesh, and the new control-point value of

$$\frac{1}{2}p_{k,l} + \frac{1}{4}p_{k+1,l} + \frac{1}{4}p_{k,l+1} \tag{3.57}$$

is exactly that of the four-direction quadratic box spline of Example $3.2.7_{/104}$. This can be seen in the following way. From $(3.23)_{/115}$ and $(3.27)_{/116}$ we have, for the

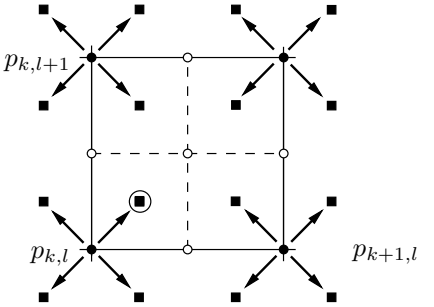


Figure 3.23. *Constant subdivision in the bivariate case.*

four-direction quadratic box spline,

$$\begin{aligned} s(z) &= z^{-\bar{e}/2} 4 \prod_{i=1}^{4} \left(\frac{1+z^{e_i}}{2} \right) \\ &= \frac{1}{4} z_1^{-1/2} z_2^{-3/2} (1+z_1)(1+z_2)(1+z_1 z_2)(1+z_1^{-1} z_2) \\ &= (z_1^{1/2} + z_1^{-1/2})(z_2^{1/2} + z_2^{-1/2}) \frac{1}{4} (z_2^{-1} + z_1 + z_1^{-1} + z_2). \end{aligned} \tag{3.58}$$

The first two factors here correspond to constant subdivision (see $(2.49)_{/76}$ and Figure $3.23_{/136}$, where the grid shown corresponds to the primal refined grid, shown in black, in Figure $2.11_{/76}$). The remaining factor $\frac{1}{4}(z_2^{-1}+z_1+z_1^{-1}+z_2)$ corresponds, for the encircled node in Figure $3.23_{/136}$, to 1/4 of the sum of the values at its North, South, East, and West neighbours in the dual mesh: $\frac{1}{4}(p_{k,l+1}+p_{k,l}+p_{k+1,l}+p_{k,l})$. This last expression coincides with $(3.57)_{/135}$.

With convergence established, the discussion following Theorem $3.3.2_{/111}$ shows that the $\{\text{Midedge}\}^2$ method produces C^1 surfaces on regular parts of the mesh. See also Exercise $2_{/141}$.

It is interesting to compare Figure $3.22_{/135}$ with Figure $1.20_{/23}$, which illustrated the $LR(2 \times 2)$ algorithm. The similarity is only at the level of the logical mesh; the computed control-point values are different.

The rate of convergence of the Midedge method is uneven. A Modified Midedge method for nonregular meshes was also introduced in [121, p. 426], with the goal of improving the rate of convergence, and this modified method could be added to the variant methods in the upper row of Figure $1.30_{/33}$ (second column). The modified method uses modified weights for faces having a number of edges other than four and therefore does, in contrast to the Midedge method, make reference to the number of edges associated with an extraordinary face.

4-8 subdivision

We now describe the 4-8 subdivision method [164]. In the regular case (this will be defined presently), it is equivalent to the four-direction box spline of

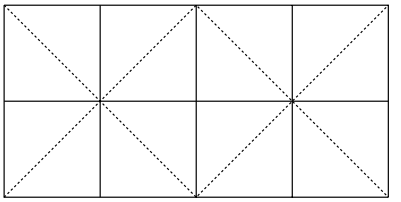


Figure 3.24. *A quadrilateral mesh with auxiliary diagonal edges.*

Example $3.4.4_{/115}$ when the four directions are chosen with multiplicity two. Since we have $\bar{e}/2 = (1,3)^t$, $z^{-\bar{e}/2} = \frac{1}{z_1 z_2^3}$, it follows from $(3.23)_{/115}$ that

$$s(z) = \frac{1}{64 z_1 z_2^3}(1+z_1)^2(1+z_2)^2(1+z_1 z_2)^2(1+z_1^{-1}z_2)^2 \tag{3.59}$$
$$= 2\left(\frac{z_1^{1/2}+z_1^{-1/2}}{2}\right)^2 2\left(\frac{z_2^{1/2}+z_2^{-1/2}}{2}\right)^2 \left(\frac{z_1^{1/2}z_2^{1/2}+z_1^{-1/2}z_2^{-1/2}}{2}\right)^2$$
$$\cdot \left(\frac{z_1^{-1/2}z_2^{1/2}+z_1^{1/2}z_2^{-1/2}}{2}\right)^2 .$$

It can be seen that this four-direction box spline corresponds to linear subdivision, $LR(1\times 1)$, but with additional averagings in the directions $(1,1)^t$ and $(-1,1)^t$, using the weights [1/4 1/2 1/4] for both averagings.

Other variants of the four-direction box spline, similar to 4-8 subdivision, are Quasi 4-8 subdivision [161, 162, 163] (a method that is nonstationary in the nonregular case) and $\sqrt{2}$-subdivision [88].

We first examine the 4-8 subdivision method without reference to the four-direction box spline. The basic method can be related to a mesh with topology corresponding to the $[4.8^2]$ tiling of [60], which was illustrated in Figure $1.12_{/16}$ (right). Rotating that tiling by 45 degrees, we obtain Figure $3.24_{/137}$, which corresponds to [164, Fig. 1.a]. For the purpose of integrating the method within the box-spline theory, however, we view the edges defining the squares in Figure $3.24_{/137}$ as the principal edges, and the diagonal edges as auxiliary. Thus, the mesh is viewed as a quadrilateral mesh, with *regular* vertices having valence 4.

In a general mesh, a quadrilateral face that has been split into two triangles is called a *basic* block. Each of the eight quadrilateral faces in Figure $3.24_{/137}$ is therefore a basic block, but note that if one of the eight diagonal edges were flipped, or rotated 90 degrees, the block would remain a basic block. Similarly, five basic blocks are shown in Figure $3.25_{/138}$ (left), with one of them shown lightly shaded. The common edge of the two triangles making up a basic block is called an *interior edge*, and the other two edges of each triangle are called *exterior edges*.

One step of the 4-8 subdivision method is made up of a series of two substeps which are loosely referred to as "bisection" substeps. To apply bisection we need a *triangulated quadrilateral mesh*, i.e., a locally planar mesh formed of basic blocks.

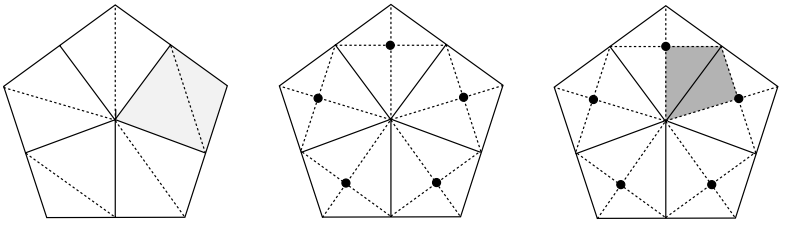


Figure 3.25. *First bisection substep.*

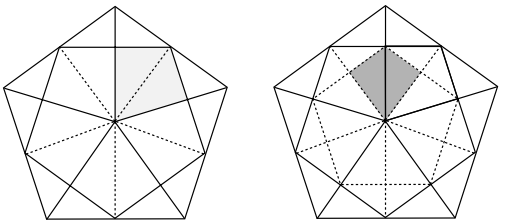


Figure 3.26. *Second bisection substep.*

Methods for transforming a locally planar mesh into a triangulated quadrilateral mesh are discussed below, at the end of the description of the method.

A bisection substep proceeds as follows. A new vertex is introduced in the middle of every internal edge of every basic block in the logical mesh (see Figure $3.25_{/138}$, middle), and it is assigned a new control point (see "Face rule," below). The adjoining triangular faces in the logical mesh are then split into two subfaces by linking the new vertex on the internal edge to the opposite vertex of each adjoining triangle. The dashed lines in Figure $3.25_{/138}$ (middle) show all edges, including those that were present due to the existence of the original basic block, and those that were created as a result of the bisection. In Figure $3.25_{/138}$ (right) a typical new basic block (for the next substep) is shown with dark shading, along with the newly introduced vertices shown by black circles. The new basic blocks are defined by two adjacent triangles separated by an edge that was *exterior* in the first substep.

To complete the first substep, the values of control points associated with existing vertices are revised (see "Vertex rule," below).

The second substep performs another bisection with the new basic blocks. To illustrate the second step, we first replace the dashed lines by solid lines, and the solid lines by dashed lines, so that we are in the same situation that obtained at the beginning of the first substep, as illustrated in Figure $3.26_{/138}$ (left). We then apply the second-substep bisection to obtain Figure $3.26_{/138}$ (right). The new vertex is introduced with valence 4 on an edge that was external in the first substep. This completes one full step of 4-8 subdivision. A new basic block at the end of the second substep is shown with dark shading.

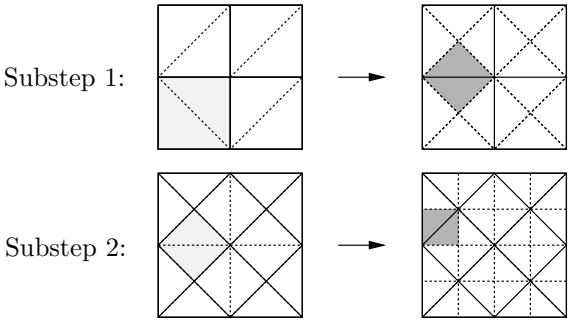


Figure 3.27. *Two bisections in the regular case.*

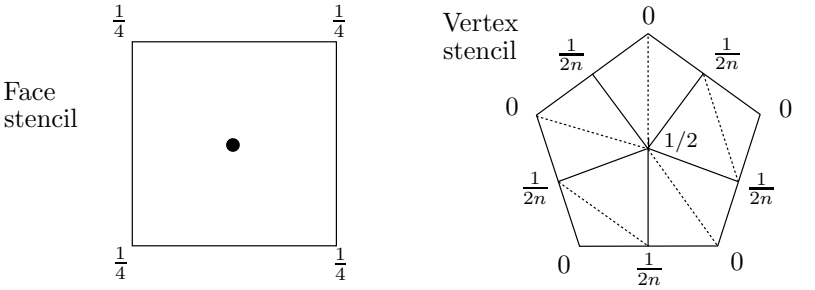


Figure 3.28. *Stencils for the* 4-8 *subdivision method.*

The corresponding illustrations are given for the regular case in Figure 3.27$_{/139}$, where the top row shows the bisection of the first substep, and the lower row shows the bisection of the second substep.

The new control points are computed in each substep according to the following rules.

- **Face rule**: the new control point associated with a vertex inserted as a result of a single bisection of a basic block is computed as the centroid of that block.
- **Vertex rule**: the revised control point associated with an existing vertex ℓ is computed as the average of the old value of the control point and the average of the vertices sharing an exterior edge with ℓ.

These rules are presented as stencils in Figure 3.28$_{/139}$, where the vertex has valence $n = 5$. The weights given for the vertex stencils are specified at vertices incident to an external edge, and they are valid independently of the orientation of the internal edges of basic blocks.

We can show that the given subdivision rules correspond in the regular case to the subdivision polynomial $(3.59)_{/137}$ by using the Polynomial Coefficient principle; i.e., we place the value 1 at the origin of the grid $\mathbb{Z}^2$, and 0 elsewhere, and apply the subdivision process. The first substep produces the values shown in

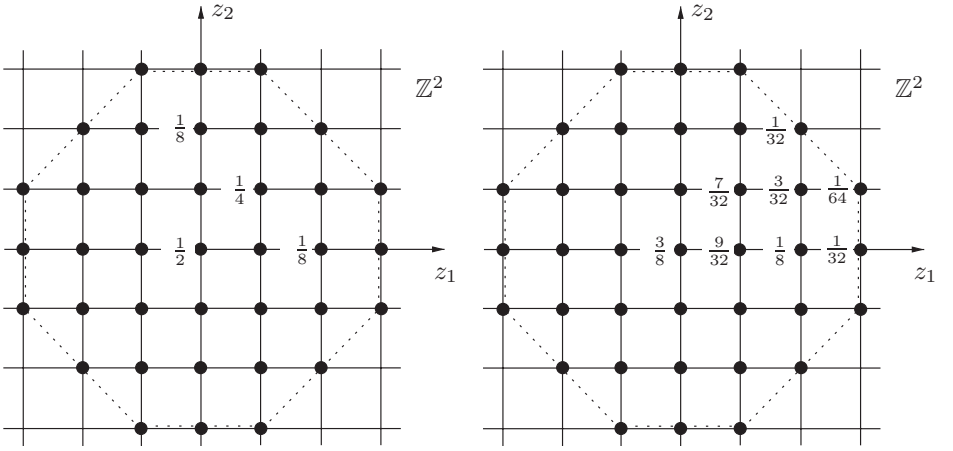


Figure 3.29. *Subdivision mask for the* 4-8 *subdivision method.*

Figure $3.29_{/140}$ (left), where other nonzero values are given by reflection in the vertical axis followed by reflection in the horizontal axis. The second substep produces the values in Figure $3.29_{/140}$ (right); again, nonzero values not shown are defined by symmetry: reflection in the diagonal, reflection in the vertical axis, and reflection in the horizontal axis. See Exercise $14_{/143}$.

It can be verified that these values are exactly the coefficients of the subdivision polynomial $s(z)$ given by $(3.59)_{/137}$, and that the process is equivalent to linear subdivision followed by the averagings described following $(3.59)_{/137}$. See Exercise $15_{/143}$.

The Midedge and 4-8 subdivision methods are sometimes referred to as $\sqrt{2}$ methods [44], since in the regular case the edge length of the grid is reduced by a factor of $\sqrt{2}$ at each substep. Similarly, it is shown later that in the regular case, the edge length of the grid is reduced by a factor of $\sqrt{3}$ at each of the two substeps of the $\{\sqrt{3}\,\}^2$ method.

Again, once convergence has been established, the discussion following Theorem $3.3.2_{/111}$ shows that the 4-8 subdivision method produces C^4 surfaces on regular parts of the mesh.

An important advantage of the 4-8 subdivision method is that the process may be stopped after the first substep within a step, and the number of faces, and therefore the amount of memory used, increases by only a factor of two after each substep. Also, it is shown in [164] how to subdivide adaptively using 4-8 subdivision, i.e., how to subdivide to different depths in different parts of the mesh.

Finally, methods for obtaining an initial triangulated quadrilateral mesh are discussed in [164]. For example, if we are given a quadrilateral mesh, it is sufficient to split each face into two triangles. If we are given a general locally planar mesh, we can begin with one Catmull–Clark step ($pQ4$ splitting) to obtain a quadrilateral mesh, and the faces can be split as before. Other methods for obtaining triangulated quadrilateral meshes are given in [164].

3.8 Additional comments

The standard references for box splines are [38], [129]. See also [37].

In the case when nonregular points occur, the questions of linear dependence and independence are harder to resolve. These questions are settled for the Catmull–Clark and Loop methods in [126].

3.9 Exercises

1. In the proof of Theorem $3.2.3_{/99}$, we used that $\int_0^h \phi(t)dt = \int_0^\infty \phi(t)dt - \int_h^\infty \phi(t)dt$. Justify the use of this equality in the proof just mentioned.

2. Consider the four-direction quadratic box spline, with directions included with multiplicity one. This is the method on which the $\{\text{Midedge}\}^2$ method is based, and the corresponding nodal function is the Zwart–Powell element. Use Theorem $3.2.9_{/107}$ to show that the nodal functions associated with the method are piecewise polynomials of degree at most 2, and show also that the polynomial is of degree exactly 2 on each piece of the domain. (This justifies the name "quadratic.")

3. We have seen, in examining the Catmull–Clark and other methods, that the subdivision rules describing a method can be formulated in more than one way, and these formulations may appear to involve a different number of rules for the same method. Fundamentally, however, the number of rules to be specified is closely related to the parity of the indices $\lfloor k \rfloor$ and $(\lfloor k_1 \rfloor, \lfloor k_2 \rfloor)$ in the subdivision equations $(2.35)_{/65}$ and $(3.44)_{/121}$, respectively. What is the number of rules that must be specified in the univariate and bivariate cases, to describe a box-spline method in the regular case, if symmetric cases are not combined? Relate this to the rules for the Catmull–Clark and Loop methods.

4. Consider the subdivision method defined by the subdivision polynomial $s(z) = z^{-3/2}(1+z)(1+z^2)/2$, which has nodal functions defined by $N_l(h;t) = N(h;t-lh)$, where $N_l(h;t) = N^1(h;t) \otimes N^1(2h;t)$ and $N^1(h;t)$ is defined as in $(2.8)_{/55}$. Show that

$$N_l(h;t) = \begin{cases} \frac{1}{2h}(t-lh) + \frac{3}{4}, & -3h/2 \le t - lh < -h/2, \\ \frac{1}{2}, & -h/2 \le t - lh < h/2, \\ -\frac{1}{2h}(t-lh) + \frac{3}{4}, & h/2 \le t - lh < 3h/2, \\ 0, & \text{otherwise}, \end{cases}$$

and consequently $\sum_l (-1)^l N_l(h;t) = 0$. There is therefore a dependence among the nodal functions, and the coefficients in $(3.44)_{/121}$ are not necessarily unique. The same example is valid in the surface case: consider the tensor product of the above method with itself.

In contrast, the univariate B-spline nodal functions for $m = 2$, corresponding to the subdivision polynomial $s(z) = z^{-1}(1 + z)(1 + z)/2$, are linearly independent. Verify this fact.

5. Carry out the first subdivision step of the $LR(3 \times 3)$ algorithm with the unit-impulse function as initial data. Consider also the effects of subsequent subdivision steps (it is not necessary to carry them out explicitly) and conclude that the support of $N(he^8; y)$ in this case is indeed $\{(u, v) : -2h \le u, v \le 2h\}$, as claimed in the discussion on the Nodal-Function computation principle that followed Example 3.5.8$_{/122}$. Similarly, carry out the first subdivision step of the method corresponding to the three-direction quartic box-spline, again with the method applied to the unit-impulse function, and conclude that in the regular case, the support of the nodal function for Loop subdivision is as shown in Figure 3.6$_{/104}$.

6. Consider again the Zwart–Powell element, discussed in Exercise 2$_{/141}$. Carry out the first subdivision step, and perhaps part of the second step, in order to determine the support of the nodal function (as in Exercise 5$_{/142}$). See Figure 3.9$_{/106}$.

7. Apply the $LR(d \times d)$ algorithm for one subdivision step, for the cases $d = 2$ and $d = 3$, and show that it produces the coefficients of the centered subdivision polynomial, as guaranteed by the Polynomial Coefficient principle.

8. Compare the two box-spline methods defined, respectively, with the vectors $(1, 0)^t$, $(0, 1)^t$, $(1, 1)^t$ and the vectors $(1, 0)^t$, $(0, 1)^t$, $(2, 2)^t$. Show that each method is related to constant subdivision combined with an averaging, but that the two methods are different.

9. In the regular case the Loop method can be viewed as linear subdivision followed by an averaging in the direction $(1, 1)^t$ using weights $[1/4 \quad 1/2 \quad 1/4]$. This was shown in the text in the case of vertices of type V in Figure 3.15$_{/125}$, by applying the averaging to the values shown in Figure 3.16$_{/126}$. The statement is also true for vertices of type H, by symmetry. Show that the statement is also true for vertices of type D and E in Figure 3.15$_{/125}$.

10. Show that applying the Loop subdivision rules of Figure 1.29$_{/32}$ to the unit-impulse function produces the coefficients 5/8, 1/16, 1/16, 1/16 shown in Figure 3.17$_{/131}$. Then, do the same for the coefficients 3/8, 3/8, 3/8, 1/8, 1/8 in the nonnegative first quadrant in Figure 3.17$_{/131}$.

11. Consider the extension of the Loop method to the case of total order $m = 3j + 3$, $j = 1, 2, \ldots$, described in Section 3.7.1. Show that if new vertices are introduced in each logical edge of a triangular mesh, with control points equal to the average of the control points at each end of the edge, and if the stencil of Figure 3.18$_{/132}$ is then applied once ($j = 1$), then the resulting method is identical to the Loop method, even if there are extraordinary vertices.

12. Show that the value of α in Theorem $3.3.2_{/111}$ is $m/3+1$ for the extended Loop method, and that the level of parametric continuity is therefore $(2m-6)/3$, as asserted in the text.

13. Extending Exercise $12_{/143}$, show that if direction vectors in the extended Loop method are added one at a time, rather than in groups of three, then the level of parametric continuity is $\lfloor (2m-6)/3 \rfloor$.

14. Show that in the regular case the Face rule and Vertex rule for 4-8 subdivision, when applied to the unit-impulse function, produce at the end of the first substep the values shown in Figure $3.29_{/140}$ (left). Also, verify the following three typical cases for the results of the second substep (values shown in Figure $3.29_{/140}$, right): the coefficient $9/32$ at $(1,0)$, the coefficient $3/8$ at the origin, and the coefficient $7/32$ at $(1,1)$.

    Suggestion. Consider first the basic block with vertices $(0,0)$, $(2,0)$, $(2,2)$, and $(0,2)$ for the first substep, and the basic block with vertices $(0,0)$, $(1,-1)$, $(2,0)$, and $(1,1)$ for the second substep.

15. Verify that applying linear subdivision, followed by averagings in the directions $(1,1)^t$ and $(-1,1)^t$ with weights $[1/4, 1/2, 1/4]$, produces the coefficients shown in Figure $3.29_{/140}$ (right). (In view of Exercise $14_{/143}$, this shows that in the regular case the 4-8 subdivision rules are equivalent to the process of linear subdivision plus the two averagings. In Section 4.2.2, it is shown that this equivalence is a consequence of the commutativity of the factors in the subdivision polynomial.) Show also that the coefficients in Figure $3.29_{/140}$ (right) are in fact the coefficients of the subdivision polynomial $(3.59)_{/137}$.

Chapter 4

Generalized-Spline Surfaces

In this chapter we consider the most general class of methods shown in Figure $1.30_{/33}$, namely the Generalized-spline subdivision methods, indicated in the fourth column of the figure.

We begin in the first section by introducing a subset of this class, the General-subdivision-polynomial methods (Figure $1.30_{/33}$, lower row of third column). This subset is general enough to include the regular case of all of the major methods mentioned in the literature, including the Modified Butterfly, Kobbelt, and $\{\sqrt{3}\}^2$ methods. The derivation and analyses are done by means of general subdivision polynomials, and the corresponding subdivision methods are basic methods.

On the other hand, the class of Generalized-spline subdivision methods is even more general. It could in principle contain basic methods that are not based on general subdivision polynomials, and it includes in addition all of the variant methods in the upper row of Figure $1.30_{/33}$. We also study the three specific methods mentioned above in the nonregular case.

Following this, we return to the regular case and extend certain ideas, discussed in previous chapters, to the more general context of this section. These ideas include the subdivision equation, the Nodal-Function Computation principle, and the Polynomial Coefficient principle. Using Fourier analysis we investigate the existence and construction of a nodal function corresponding to a given subdivision polynomial. We also describe how the support of a nodal function is related to the coefficient set of the subdivision polynomial and give conditions on the subdivision polynomial that characterize affine invariance.

Finally, this chapter introduces a certain two-dimensional manifold associated with a given locally planar logical mesh. This manifold can be viewed as a global parametric domain for the subdivision surface, and we use it to define the generalized nodal splines (Figure $1.30_{/33}$, fourth column) by means of the Nodal-Function Computation principle.

4.1 General subdivision polynomials

In Section 2.2 we derived recursion formulas for control points that were based on the 2-scale relation

$$N^m(h;t) = \sum_k s_k N^m(h/2; t - kh/2)$$

as in formula $(2.24)_{/61}$ with $l = 0$. This may also be rewritten as

$$N^m(h;t) = \sum_k s_k N^m(h; 2t - kh) = \left(\sum_{k \in G} s_k z^k \right) N^m(h; 2t),$$

i.e.,

$$N^m(h;t) = s(z) N^m(h; 2t),$$

where

$$s(z) = 2\Big(\frac{z^{1/2} + z^{-1/2}}{2}\Big)^m. \tag{4.1}$$

The corresponding formulas for control points were, in terms of generating functions,

$$p(h/2; z) = s(z) p(h; z^2), \tag{4.2}$$

and explicitly in terms of coefficients,

$$q_k = \sum_{l \in \mathbb{Z}} s_{k-2l} p_l, \tag{4.3}$$

where $p(h/2; z) = \sum_k q_k z^k$ and $p(h; z) = \sum_l p_l z^l$. This was generalized to the case of box splines in $(3.44)_{/121}$.

It is now quite natural to consider more general subdivision polynomials $s(z) = \sum_{k \in G} s_k z^k$ than those given in $(3.29)_{/116}$. In order to carry out the analysis we must, for a given subdivision polynomial $s(z)$, find some function $N(t)$ such that

$$N(t) = \sum_{k \in G} s_k z^k N(2t) = \sum_{k \in G} s_k N(2(t - kh/2)) = \sum_{k \in G} s_k N(2t - kh), \tag{4.4}$$

where h is an initial mesh size and G denotes a finite subset of $\mathbb{Z}$ or $\mathbb{Z} + 1/2$. Then the subdivision procedure is used to produce a parametric curve

$$x(t) = \sum_{l \in \mathbb{Z}} p_l N(t - lh) = \sum_{l \in \mathbb{Z}} p_l N_l(t). \tag{4.5}$$

For a given subdivision polynomial the crucial problem is first to find (if possible) a function $N(t)$ satisfying the 2-scale relation $(4.4)_{/146}$, then to show that the subdivision procedure converges to the curve in $(4.5)_{/146}$, and finally to analyse the

regularity of this curve. The analysis, in the general case, is quite difficult. A rigorous convergence and smoothness analysis for ordinary B-spline curves, as well as for those given by more general subdivision polynomials, is postponed until Chapter 5. In this section we only describe some more formal, although illuminating, aspects of the analysis.

Exactly the same formalism and analysis may be used in the bivariate case when we wish to construct parametric surfaces of the form

$$x(y) = \sum_{l \in \mathbb{Z}^2} p_l N(y - lh). \tag{4.6}$$

Here $N(y) = N(u, v)$ is a function of $(u, v)^t \in \mathbb{R}^2$ satisfying a 2-scale relation (compare with $(4.4)_{/146}$):

$$N(y) = \sum_{k \in G} s_k z^k N(2y) = \sum_{k \in G} s_k N(2(y - kh/2)) = \sum_{k \in G} s_k N(2y - kh), \tag{4.7}$$

i.e.,

$$N(y) = s(z)N(2y),$$

where we have introduced the subdivision polynomial

$$s(z) = \sum_{k \in G} s_k z^k \tag{4.8}$$

and where, as usual, we have the compact notation $z = (z_1, z_2)$, $k = (k_1, k_2)$, and $z^k = z_1^{k_1} z_2^{k_2}$. The coefficient grid G is a finite subset of $\mathbb{Z}^2 + (\epsilon_1/2, \epsilon_2/2)^t$ with $\epsilon_1, \epsilon_2 \in \{0, 1\}$. In all our examples G is a centrally symmetric set, which leads to centered subdivision polynomials, but this hypothesis is not necessary for purposes of the analysis.

Consider the case $\epsilon_1 = \epsilon_2 = 0$. If such a function $N(y)$ satisfying $(4.7)_{/147}$ can be found, then, as in the univariate case, we must have

$$x(y) = \left(\sum_{l \in \mathbb{Z}^2} p_l z^{2l} \right) N(y) = \left(\sum_{l \in \mathbb{Z}^2} p_l z^{2l} \right) s(z) N(2y) = \left(\sum_{k \in \mathbb{Z}^2} q_k z^k \right) N(2y), \tag{4.9}$$

which is satisfied if

$$\sum_{k \in \mathbb{Z}^2} q_k z^k = s(z) \sum_{l \in \mathbb{Z}^2} p_l z^{2l}$$

or, with the previous notation,

$$p(h/2; z) = s(z) p(h; z^2). \tag{4.10}$$

As in the univariate and the box-spline cases, this gives the fundamental recursion formula

$$q_k = \sum_{l \in \mathbb{Z}^2} s_{k-2l} p_l, \quad k \in \mathbb{Z}^2 \tag{4.11}$$

for the refinement of sets of control vectors, and as before, this formula is referred to as the *subdivision equation*. It is shown below that if the subdivision process converges, then the function $N(y)$ must satisfy the condition that $\int_{\mathbb{R}^2} N(y)dy = h^2$.

The set G is defined as the set of nonzero coefficients of the subdivision polynomial, and it gives the indices for which p_l influences the control point q_k. In fact, one of the first things we do, following the description of certain particular General-subdivision-polynomial methods in the next section, is to generalize the subdivision equations $(2.29)_{/64}$ and $(3.44)_{/121}$. It is instructive to look again at column l of the matrix Σ in $(2.16)_{/59}$, in relation to the equation $q_k = \sum_{l\in\mathbb{Z}} s_{k-2l}p_l$ in $(2.29)_{/64}$. Together these show that first, the influence of p_l on control points q_k in the refined mesh is restricted to values of k such that $-m/2 \le k - 2l \le m/2$, and second, that the additive contribution of p_l to the value of the control point q_k is s_{k-2l}. In the more general context of this section, the set G corresponds exactly to the set $\{-m/2, -m/2+1, \ldots, m/2\}$ in the scalar uniform B-spline case, since $(4.11)_{/147}$ above generalizes $(2.29)_{/64}$ to

$$q_k = \sum_{k-2l\in G} s_{k-2l}p_l.$$

Further, it follows that the Polynomial Coefficient principle of Section 2.2.4 generalizes to the case of general subdivision polynomials. As in the B-spline case, if only the subdivision process is given, the subdivision polynomial, and the set G, can be found by applying a single step of the process to the unit-impulse function.

To simplify the notation, we present the analysis only in the case $\epsilon_1 = \epsilon_2 = 0$, since the analysis is identical for the other cases. Also, we have dropped the parameter h when referring to $N(h;\cdot)$. We often do this, from now on, when h is not explicitly required.

Bisection is not the only possibility for defining general subdivision polynomials: another is trisection. For example, the $\{\sqrt{3}\,\}^2$ method is based on trisection, and the necessary modifications in the definitions to cover this case are straightforward. The variable z in the corresponding general subdivision polynomial corresponds to translations of $h/3$ rather than $h/2$. The polynomial is given in Section 4.2.1.

Later in the chapter, we give conditions on the subdivision polynomial that characterize affine invariance, and the latter is shown in Chapter 5 to be necessary for convergence. Further, the Nodal-Function Computation principle of Section 2.2.4 can be applied in the case of general subdivision polynomials. Thus, if the subdivision process is convergent, then $(4.6)_{/147}$ implies that $N(y)$ can be computed by applying the process to the unit-impulse function. In fact, the Nodal-Function Computation principle can be applied even in the case of nonregular meshes; see Section 4.7. In Section 4.3 we investigate in a formal way, using Fourier analysis, how for a given subdivision polynomial $s(z)$ a nodal function satisfying the 2-scale relation $(4.7)_{/147}$ can be constructed. In Section 4.4 we show how to determine the support of the nodal function. It is also possible to relate the support of the nodal function to the coefficient set G.

We first present the examples mentioned above.

4.2 General-subdivision-polynomial methods and their variants

In this section we give examples of subdivision methods which correspond to general subdivision polynomials, but which are not box-spline methods. We call these methods *General-subdivision-polynomial* methods; see Figure $1.30_{/33}$, lower row of the third column. They are also called *shift-invariant* methods.

4.2.1 Examples of non-box-spline schemes: $4pt \times 4pt$, Butterfly, $\sqrt{3}$-subdivision (regular case)

We first observe that the box splines are indeed a special case, i.e., they can be formulated in terms of general subdivision polynomials, as a special case. The subdivision polynomial in $(3.29)_{/116}$ corresponds to $(4.8)_{/147}$, with $G = G_m$, and $(3.38)_{/120}$ corresponds to $(4.6)_{/147}$. Looking ahead, the 2-scale relation $(3.30)_{/117}$, corresponds to $(4.33)_{/167}$, and the statement about the support of the nodal function, in Remark $3.1.5_{/96}$, corresponds to Theorem $4.4.1_{/170}$.

The following examples, however, are not box splines.

The four-point subdivision scheme

We return temporarily to the univariate case and consider an important example. In the description here, we suppose that the subdivision polynomial has been given.

The four-point subdivision scheme as described in [46, 48] is an interpolating subdivision process for curves, having the centered subdivision polynomial

$$s(z) = \Big(\frac{z^{1/2} + z^{-1/2}}{2}\Big)^4 (-z + 4 - z^{-1}) \tag{4.12}$$

$$= (-z^{-3} + 9z^{-1} + 16 + 9z - z^3)/16. \tag{4.13}$$

In $(4.13)_{/149}$ all terms, except the middle one, are of odd degree. Now, the subdivision formula

$$p(h/2; z) = s(z)p(h; z^2)$$

produces the coefficients $\{q_k\}_{k\in\mathbb{Z}}$ in

$$p(h/2; z) = \sum_k q_k z^k$$

given by

$$q_k = \sum_{l\in\mathbb{Z}} s_{k-2l} p_l.$$

Since from $(4.13)_{/149}$ we have

$$s_{2l} = \begin{cases} 1 & \text{if } l = 0, \\ 0 & \text{otherwise,} \end{cases}$$

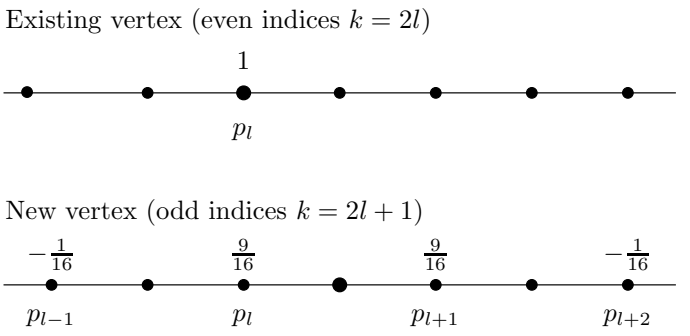


Figure 4.1. *Stencils for the four-point method.*

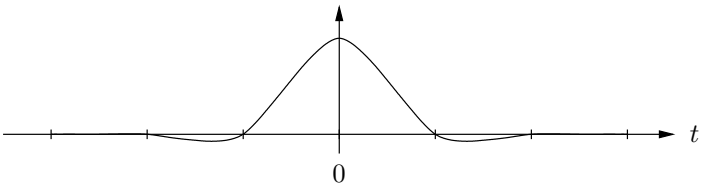


Figure 4.2. *Nodal function for the four-point scheme.*

it follows that for even indices $k = 2l$ we get $q_{2l} = p_l$, i.e., the old control points are unchanged and the process is in fact interpolating. For odd indices, we have

$$q_{2l+1} = -\frac{1}{16}p_{l-1} + \frac{9}{16}p_l + \frac{9}{16}p_{l+1} - \frac{1}{16}p_{l+2}.$$

The two cases are illustrated by the stencils in Figure $4.1_{/150}$. A subdivision matrix, analogous to the matrix Σ given in $(2.16)_{/59}$ for the B-spline case, can be determined from these expressions; see Exercise $1_{/186}$.

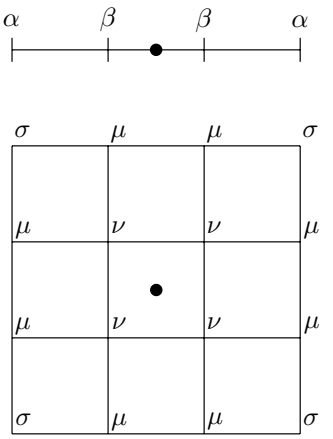
The set $G \subset \mathbb{Z}$ is $\{-3, -1, 0, 1, 3\}$, and the support of the nodal basis function is the interval $h\,\mathrm{conv}(G) = [-3h, 3h]$; see Section 4.4.

The nodal function for the four-point scheme is shown in Figure $4.2_{/150}$. It was obtained by applying the subdivision process to a scalar control sequence with value 1 at $l = 0$, and 0 elsewhere. This is the Nodal-Function Computation principle: substituting the values just mentioned into $(4.5)_{/146}$ gives $x(t) = N_0(t)$. The nodal function is not a piecewise polynomial; see Exercise $2_{/186}$.

A more general form of the four-point subdivision scheme [49] introduces a tension parameter $w > 0$ and has centered subdivision polynomial

$$s(z) = -wz^{-3} + (1/2 + w)z^{-1} + 1 + (1/2 + w)z - wz^3. \tag{4.14}$$

In $(4.13)_{/149}$ we have $w = 1/16$.



Four-point scheme

$\alpha = -w \qquad \beta = (1/2 + w)$

$4pt \times 4pt$ scheme

$\sigma = \alpha^2 = w^2$

$\mu = \alpha\beta = -w(1/2 + w)$

$\nu = \beta^2 = (1/2 + w)^2$

Figure 4.3. *Stencil for the $4pt \times 4pt$ scheme (new vertices).*

The $4pt \times 4pt$ subdivision scheme

We denote the tensor product of the four-point scheme with itself by $4pt \times 4pt$ (Figure $1.30_{/33}$, third column). This is an interpolating subdivision scheme that uses a $pQ4$ splitting schema and applies to a regular quadrilateral grid. The generalization of the resulting surface-subdivision scheme to nonregular meshes is called Kobbelt's method, which is discussed in Section 4.2.3.

It is clear from $(4.14)_{/150}$ that the $4pt \times 4pt$ scheme is not a box-spline method, since from $(3.19)_{/114}$, the coefficients of box-spline subdivision polynomials are nonnegative.

The weights for the general four-point scheme [49], and the tensor-product $4pt \times 4pt$ scheme, are shown in Figure $4.3_{/151}$. (Note that the w of [48, 49] is denoted $\omega/16$ in [73]; also, a new variable w has been introduced in [73] which is equal, in the regular case, to 8 times the w used in this book and in [48, 49].)

The coefficients of the univariate four-point rule are shown in Figure $4.3_{/151}$ (top), and the coefficients for the $4pt \times 4pt$ method are shown in Figure $4.3_{/151}$ (bottom).

The Butterfly subdivision scheme (regular case)

In the description of this method, we suppose that the stencil describing the subdivision is given.

The Butterfly subdivision scheme is described in [49]. It applies to triangular meshes and uses the $pT4$ splitting schema. Here we consider its application to a regular mesh. Given the triangular mesh at some subdivision level, new vertices are introduced on each edge, and the new vertices corresponding to a given triangle are connected (dashed lines in Figure $4.4_{/152}$). The control point for this "new edge point" is defined by

$$q = \frac{1}{2}(p_1 + p_2) + 2w(p_3 + p_4) - w(p_5 + p_6 + p_7 + p_8) \tag{4.15}$$

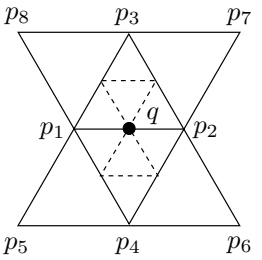


Figure 4.4. *Notation for the Butterfly scheme.*

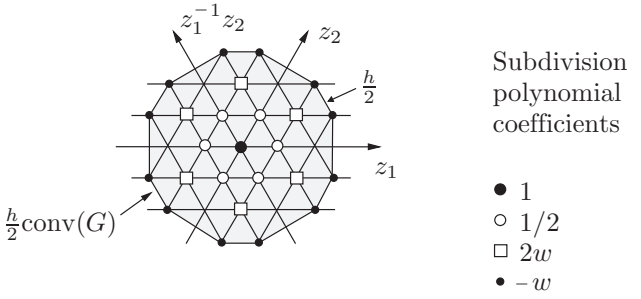


Figure 4.5. *Subdivision mask for the Butterfly method.*

with notation according to Figure 4.4$_{/152}$. The stencil for the new vertices is shown in Figure 4.6$_{/154}$ (left). The old vertices retain their control points, and therefore the Butterfly scheme is interpolating. Using this last fact, and (4.15)$_{/151}$, we may construct the subdivision polynomial for the Butterfly scheme. To do this, we use a layout similar to the one shown in Figure 3.14$_{/125}$: we associate z_1 and z_2 with a nonorthogonal coordinate system having axes separated by an angle of $\pi/3$ rather than $\pi/2$, as shown in Figure 4.5$_{/152}$. This gives a triangular grid defined by the six axes z_1, z_2, $z_1^{-1}z_2$, z_1^{-1}, z_2^{-1}, $z_1z_2^{-1}$. The grid shown in the figure is $\frac{h}{2}G$, and the equilateral triangles shown have sides of length $h/2$. The set $\frac{h}{2}\text{conv}(G)$ is also indicated. The nodal basis function $N(u,v)$ has its support in $h\,\text{conv}(G)$; see Section 4.4, below.

The stencil shown in Figure 4.4$_{/152}$ focuses on the points p_l contributing to the value of a typical point q in the refined mesh, but the subdivision mask in Figure 4.5$_{/152}$ is centered on a typical control point p_l from the nonsubdivided mesh. According to the Polynomial Coefficient principle, mentioned in Section 4.1, the coefficients s_{k-2l} of the subdivision polynomial quantify the influence of such a point p_l on the neighbouring control points in the refined mesh, and Figure 4.5$_{/152}$ shows this influence. The subdivision mask can be found, as usual, by applying a single step of the subdivision rules illustrated in Figure 4.4$_{/152}$ to the unit-impulse function in $\mathbb{Z}^2$.

Giving more detail, the points in the set G for which p_l will make a contribution are

$$G = (0,0) \cup G^1 \cup G^2 \cup G^3$$

with

$$\begin{aligned}
G^1 &= \{(1,0),(0,1),(-1,1),(-1,0),(0,-1),(1,-1)\},\\
G^2 &= \{(1,1),(-1,2),(-2,1),(-1,-1),(1,-2),(2,-1)\},\\
G^3 &= \{(2,1),(1,2),(-1,3),(-2,3),(-3,2),(-3,1),\\
&\qquad (-2,-1),(-1,-2),(1,-3),(2,-3),(3,-2),(3,-1)\}.
\end{aligned}$$

Thus,

$$s(z) = \sum_{k \in G} s_k z^k,$$

where $s_{(0,0)} = 1$, $s_k = 1/2$ if $k \in G^1$, $s_k = 2w$ if $k \in G^2$, and $s_k = -w$ if $k \in G^3$. This gives

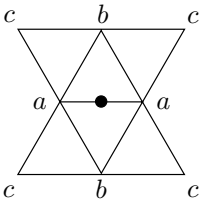
$$\begin{aligned}
s(z) = s(z_1, z_2) = 1 &+ \frac{1}{2}(z_1 + z_1^{-1} + z_2 + z_2^{-1} + z_1^{-1}z_2 + z_1 z_2^{-1})\\
&+ 2w(z_1 z_2 + z_1^{-1} z_2^{-1} + z_1^{-1} z_2^2 \; + \; z_1 z_2^{-2} + z_1^{-2} z_2 \; + z_1^2 z_2^{-1})\\
&- w(z_1^2 z_2 + z_1^{-2} z_2^{-1} + z_1 z_2^2 + z_1^{-1} z_2^{-2} + z_1^{-1} z_2^3 + z_1^1 z_2^{-3}\\
&\qquad + z_1^{-2} z_2^3 + z_1^2 z_2^{-3} + z_1^{-3} z_2^2 + z_1^3 z_2^{-2} + z_1^{-3} z_2 + z_1^3 z_2^{-1}). \quad (4.16)
\end{aligned}$$

The Butterfly method is not a box-spline subdivision scheme. This again follows from the fact that box-spline nodal functions are by definition nonnegative: for the Butterfly scheme the tension parameter w is positive, and therefore the nodal function $N(y)$ has the property that $N(y) = -w < 0$ if $y \in hG^3/2$. Here we have used the interpolation property of the scheme.

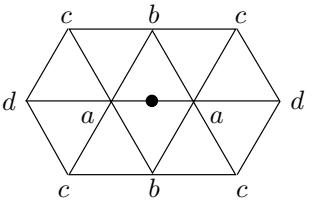
Another version of the basic Butterfly scheme uses a larger 10-point stencil for the regular case: stencils for the two versions are shown side by side in Figure 4.6$_{/154}$. The 10-point stencil with $w^* = 0$ gives the 8-point stencil with $w = 1/16$ [178, Sec. 3.3]. Also, there is a variant of the Butterfly scheme, called the Modified Butterfly Scheme [178], which is suitable for nonregular meshes, and which is discussed in Section 4.2.3. Although the 8-point Butterfly method is applicable in the nonregular case, the Modified Butterfly method was introduced in order to obtain satisfactory smoothness properties in the nonregular case; see Section 5.8 and [172, Sec. 6.4.1].

The $\sqrt{3}$-subdivision scheme (regular case)

The $\sqrt{3}$-subdivision method [76] applies to triangular meshes, which need not be regular (the nonregular case is discussed in Section 4.2.3). The method uses a splitting schema that is different from the $pT4$ schema used by the Loop and Butterfly



8-point stencil
$a = 1/2$
$b = 2w$
$c = -w$



10-point stencil
$a = 1/2 - w^*$
$b = 1/8 + 2w^*$
$c = -1/16 - w^*$
$d = w^*$

Figure 4.6. *Butterfly method:* 8*-point and* 10*-point stencils.*

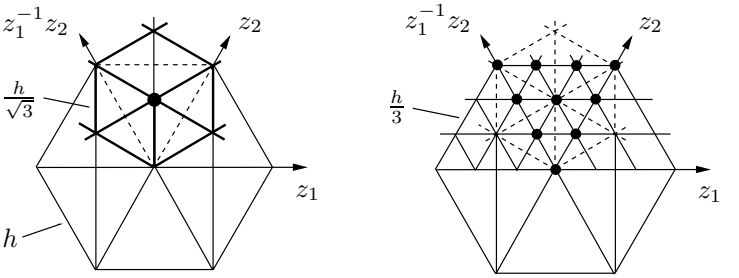


Figure 4.7. *The* $\sqrt{3}$ *splitting: Two substeps.*

methods. The $pT4$ schema adds a new vertex in each edge of each triangle and performs a 4-1 split. In contrast, in a single subdivision step, the splitting schema used by the $\sqrt{3}$ method inserts a new vertex in the middle of each triangle of the mesh, joins them to the original vertices, and then flips the original edges to obtain the subdivided mesh. The underlying tiling implicit in the splitting schema is, as it was for the $pT4$ splitting, the $[6^3]$ tiling of Figure $1.12_{/16}$. Consequently, ordinary vertices again have $n = 6$, and ordinary faces have $e = 3$.

The process is illustrated in Figure $4.7_{/154}$ (left), in which six equilateral triangles are shown, each with side h. The new vertex inserted in the top centre triangle is indicated by a black dot. This new vertex and the new vertices inserted in adjacent triangles are joined to the original vertices, as illustrated by the heavy solid lines. The edges of the original triangle that disappear, due to the edge flip, have been downgraded to dashed lines, while the new edges resulting from the flip are shown as heavy solid lines. The resulting triangles have been rotated by 30 degrees (this is made more explicit below) and have edge length equal to $h/\sqrt{3}$.

Applying this process twice (which corresponds to the method we have denoted $\{\sqrt{3}\,\}^2$), produces a subdivided mesh with nine triangles for each original

triangle, and with two new vertices introduced for each original edge, as illustrated in Figure $4.7_{/154}$ (right). For reasons that will become apparent in Section 4.2.2, where we compare this method with the box-spline method underlying 4-8 subdivision, we prefer to view the $\{\sqrt{3}\}^2$ method as the fundamental method, so that each of the two applications of the $\sqrt{3}$ process is a substep of the method. The initial edges at the beginning of the second substep are shown in Figure $4.7_{/154}$ (right) as dashed lines, since, as in the first substep, they will presently disappear, due to the edge flip. In fact, in the second substep, new vertices are inserted in the middle of the triangles having edge length $h/\sqrt{3}$, the new vertices are joined to the original vertices of these triangles, and the edges are flipped; the flipping process causes the dashed lines to be replaced by solid lines that cross the dashed lines at right angles, as illustrated in Figure $4.7_{/154}$ (right). The newly formed triangles have sides of length $(h/\sqrt{3})/\sqrt{3} = h/3$, so that the edges which had length h, at the start of the complete two substep process, have now been trisected.

The $\{\sqrt{3}\}^2$ method involves a form of primal-dual alternation [118]. In fact, the nodes introduced in the first substep, such as the one indicated by the black dot in Figure $4.7_{/154}$ (left), correspond to nodes in the hexagonal dual of the primal triangular graph we started with, at the beginning of the first substep, whereas the nodes introduced in the second substep are in a subdivided version of the primal triangular graph.

In each of the two substeps, the smoothing rule for $\sqrt{3}$-subdivision specified in [76] assigns a value, for a *new* vertex introduced in the middle of a face, that is equal to the centroid of the vertices of the face. Further, if $p_{\ell_0}, \ldots, p_{\ell_5}$ are the immediate neighbours of an *existing* regular vertex ℓ with control point p_ℓ, then the modified value of the control point is

$$\frac{2}{3}p_\ell + \frac{1}{3}\left(\frac{1}{6}\sum_{i=0}^{5} p_{\ell_i}\right) = \frac{2}{3}p_\ell + \frac{1}{18}\sum_{i=0}^{5} p_{\ell_i}. \tag{4.17}$$

As mentioned in Section 4.1, the Polynomial Coefficient principle can be applied for methods based on general subdivision polynomials, and this includes general subdivision polynomials based on trisection. If we apply this principle to the $\{\sqrt{3}\}^2$ method, putting a value of 1 at the origin results in the values shown to the right of the vertices in Figure $4.8_{/156}$. In order to not clutter the figure, the values resulting from the first $\sqrt{3}$ substep are not explicitly shown: they are 2/3 at the origin (large white circular symbol), 1/3 at the vertices in the middle of each of the six original triangles (medium-size white circles), and 1/18 at the outer vertices of the six original triangles (small white circles). A typical original triangle with side h is shown shaded. Given the values from the first $\sqrt{3}$ substep, the second $\sqrt{3}$ substep is applied to produce the values shown in Figure $4.8_{/156}$. (Values not shown are defined by symmetry: first by reflection in a line drawn vertically from the origin in Figure $4.8_{/156}$, and then by hexagonal symmetry.)

There are several things worth mentioning about the values shown in Figure $4.8_{/156}$. First, the sum of the values of all coefficients is equal to 9: in each of the two substeps, the total weight increases by a factor of $2/3+6(1/3)+6(1/18) = 3$. See Exercise $3_{/186}$. Second, according to the Nodal-Function Computation principle,

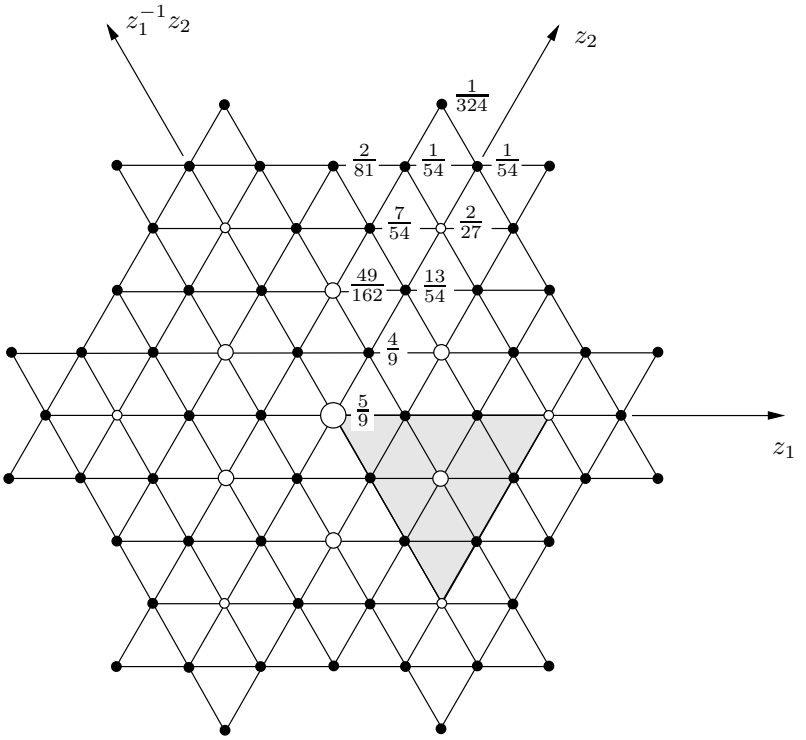


Figure 4.8. *Subdivision mask for the* $\{\sqrt{3}\}^2$ *method.*

the coefficients can be viewed as a first approximation to the nodal function in $(4.4)_{/146}$. Third, the fact that the new coefficients in Figure $4.8_{/156}$ contain negative powers of integers other than 2 shows that $\{\sqrt{3}\}^2$ is not a box-spline method in the sense of Section 3.5.1, since this would be inconsistent with $(3.27)_{/116}$. On the other hand, it is shown in Section 4.2.2 that the $\{\sqrt{3}\}^2$ method is strongly analogous to 4-8 subdivision, which is a box-spline method.

It is interesting, also, that by affine invariance, the total contribution to a single point, upon applying a weight of 1 to *all* of the vertices of the original triangles having edge length h, must be 1. This can be seen, for example, for the origin, which receives a contribution of $2/27$ from each of its neighbours in the six original triangles, plus a contribution of $5/9$ from itself, for a total of $6(2/27) + 5/9 = 1$. The other types of points are similar. See Exercise $4_{/186}$.

We may view the $\{\sqrt{3}\}^2$ method as defined on a regular triangular grid where the grid points are generated by the vectors $h(1,0)$, $h(0,1)$, $h(-1,1)$, $h(-1,0)$, $h(0,-1)$, $h(1,-1)$, which are represented by the monomials z_1, z_2, $z_1^{-1}z_2$, z_1^{-1}, z_2^{-1}, $z_1z_2^{-1}$, respectively, so that the grid points are exactly $h\mathbb{Z}^2$. If we consider an oblique coordinate system where the z_1- and z_2-axes are separated by an angle of $\pi/3$, then all triangles are equilateral, as illustrated in Figure $4.8_{/156}$.

The sequence of control points at the subdivision level h are represented by the generalized polynomial

$$p(h;z) = \sum_{k\in\mathbb{Z}^2} p^h_{k_1,k_2} z_1^{k_1} z_2^{k_2}.$$

After the two substeps of the $\{\sqrt{3}\}^2$ method, the sequence of control points at level $h/3$ is represented by $p(h/3;z)$ given by

$$p(h/3;z) = s(z_1,z_2)p(h;z^3), \tag{4.18}$$

where $p(h;z^3)$ is obtained from $p(h;z)$ by replacing z_1 by z_1^3 and z_2 by z_2^3, and the subdivision polynomial $s(z_1,z_2)$ is given by

$$s(z_1,z_2) = s_P(z_1z_2, z_1^{-1}z_2^2)s_P(z_1^3,z_2^3) \tag{4.19}$$

with

$$\begin{aligned} s_P(z_1,z_2) &= \frac{2}{3} \\ &+ \frac{1}{3}(z_1^{1/3}z_2^{1/3} + z_1^{-1/3}z_2^{2/3} + z_1^{-2/3}z_2^{1/3} + z_1^{-1/3}z_2^{-1/3} + z_1^{1/3}z_2^{-2/3} + z_1^{2/3}z_2^{-1/3}) \\ &+ \frac{1}{18}(z_1 + z_2 + z_1^{-1}z_2 + z_1^{-1} + z_2^{-1} + z_1z_2^{-1}). \end{aligned} \tag{4.20}$$

The subscript P in s_P is intended to suggest "partial."

Multiplying $p(h;z_1^3,z_2^3)$ by the second factor $s_P(z_1^3,z_2^3)$ corresponds to the first substep, and multiplying by the first factor $s_P(z_1z_2, z_1^{-1}z_2^2)$ corresponds to the second substep. The second factor in $(4.19)_{/157}$ is, explicitly,

$$\begin{aligned} s_P(z_1^3,z_2^3) = \frac{2}{3} &+ \frac{1}{3}(z_1z_2 + z_1^{-1}z_2^2 + z_1^{-2}z_2 + z_1^{-1}z_2^{-1} + z_1z_2^{-2} + z_1^2z_2^{-1}) \\ &+ \frac{1}{18}(z_1^3 + z_2^3 + z_1^{-3}z_2^3 + z_1^{-3} + z_2^{-3} + z_1^3z_2^{-3}); \end{aligned}$$

the first factor in $(4.19)_{/157}$ is

$$\begin{aligned} s_P(z_1z_2, z_1^{-1}z_2^2) = \frac{2}{3} &+ \frac{1}{3}(z_1 + z_2 + z_1^{-1}z_2 + z_1^{-1} + z_2^{-1} + z_1z_2^{-1}) \\ &+ \frac{1}{18}(z_1z_2 + z_1^{-1}z_2^2 + z_1^{-2}z_2 + z_1^{-1}z_2^{-1} + z_1z_2^{-2} + z_1^2z_2^{-1}). \end{aligned}$$

Multiplying the two polynomials just given, as prescribed by $(4.19)_{/157}$, gives a polynomial with exactly the coefficients shown in Figure $4.8_{/156}$. This can be verified by symbolic computation, or by means of the method summarized in the solution to Exercise $15_{/143}$ at the end of Chapter 3.

It was stated in Section 4.1 that the theory presented later in this section holds for trisection, as well as for bisection, provided that appropriate modifications

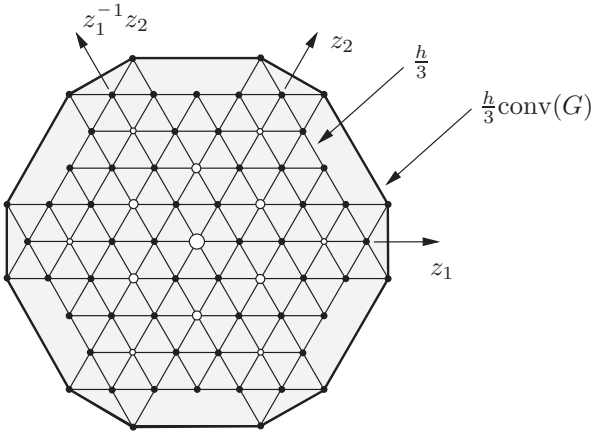


Figure 4.9. *The set* $\frac{h}{3}\text{conv}(G)$.

are made. Thus, the variable z in the subdivision polynomial corresponds to translations of $h/3$ rather than $h/2$, and the side of an equilateral triangle in Figure 4.8$_{/156}$ is $h/3$, as opposed to $h/2$ in Figure 4.5$_{/152}$. Carrying the comparison between the two figures a little further, in Figure 4.5$_{/152}$ the set $\frac{h}{2}\text{conv}(G)$ is illustrated, and in Figure 4.9$_{/158}$, in place of that set, we show the set $\frac{h}{3}\text{conv}(G)$, which corresponds to the $\{\sqrt{3}\,\}^2$ method.

Remark* 4.2.1. Theorem 4.4.1$_{/170}$ states that the support of the nodal function for a method like the Butterfly method is contained in a set twice as large as $\frac{h}{2}\text{conv}(G)$, i.e., the set $h\,\text{conv}(G) = 2\left[\frac{h}{2}\text{conv}(G)\right]$. The support of the nodal function in the case of the $\{\sqrt{3}\,\}^2$ method is contained in $\frac{3}{2}\left[\frac{h}{3}\text{conv}(G)\right]$, i.e., a set $\frac{3}{2}$ as large as the one shown in Figure 4.9$_{/158}$. The modifications to Theorem 4.4.1$_{/170}$ that explain the new constants are easy to describe: in (4.43)$_{/168}$ and (4.56)$_{/170}$, each appearance of the constant 4 must be replaced by a 9, and each 2 by a 3. This means that the conclusion of Theorem 4.4.1$_{/170}$ becomes $\text{supp}\,(N) \subseteq \frac{h}{2}\,\text{conv}(G)$, since in (4.57)$_{/171}$, $h/2$ is replaced by $h/3$ and

$$\sum_{j=0}^{k} 2^{-j}\text{conv}(G) = 2(1-2^{-k-1})\text{conv}(G)$$

is replaced by

$$\sum_{j=0}^{k} 3^{-j}\text{conv}(G) = \frac{3}{2}(1-3^{-k-1})\text{conv}(G).$$

Thus the product $\frac{h}{2}2 = h$ is replaced by $\frac{h}{3}\frac{3}{2} = \frac{h}{2}$, and

$$\text{supp}\,(N) \subseteq \frac{h}{2}\text{conv}(G) = \frac{3}{2}\left[\frac{h}{3}\text{conv}(G)\right].$$

In fact, equality holds, although we do not prove it. ∎

4.2.2 Comparison of 4-8 and $\sqrt{3}$-subdivision (regular case)

In this section we compare the order-8 four-direction box-spline method, defined in Examples $3.2.7_{/104}$ and $3.4.4_{/115}$ and by $(3.42)_{/120}$, with the regular version of the $\{\sqrt{3}\,\}^2$ method, just discussed. The first of these two methods is a box-spline method, but the second is not, as can be seen from the fact that in the $\{\sqrt{3}\,\}^2$ case we are led to subdivision polynomials that do not have the form $(3.27)_{/116}$. We show nonetheless that the two methods are essentially the same, with the first applied to regular quadrilateral meshes and the second applied to regular triangular meshes.

Exercises $14_{/143}$ and $15_{/143}$ in Chapter 3 confirmed that the 4-8 subdivision method, defined in terms of the rules given in Section 3.7.2, is equivalent in the regular case to the order-8 four-direction box-spline method. For brevity, in this section we omit the words "four-direction."

Aside from the different types of mesh, the essential difference between the order-8 box-spline method and the $\{\sqrt{3}\,\}^2$ method is a different choice of weights for an existing vertex, relative to the weights of its neighbours, when updating the existing vertex. This is a fairly minor difference, something like the different choice of weights mentioned in Section A.1 for the Catmull–Clark method.

For the order-8 box-spline method we have, repeating $(3.59)_{/137}$, that

$$\begin{aligned} s(z) &= \frac{1}{64 z_1 z_2^3}(1+z_1)^2(1+z_2)^2(1+z_1 z_2)^2(1+z_1^{-1}z_2)^2 \qquad (4.21)\\ &= 2\left(\frac{z_1^{1/2}+z_1^{-1/2}}{2}\right)^2\left(\frac{z_2^{1/2}+z_2^{-1/2}}{2}\right)^2 2\left(\frac{z_1^{1/2}z_2^{1/2}+z_1^{-1/2}z_2^{-1/2}}{2}\right)^2 \\ &\qquad \cdot\left(\frac{z_1^{-1/2}z_2^{1/2}+z_1^{1/2}z_2^{-1/2}}{2}\right)^2. \end{aligned}$$

This means that the sequence of control points at level $h/2$ is represented by

$$p(h/2;z) = s(z_1,z_2)p(h;z^2),$$

where $p(h/2;z)$ represents the sequence of control points at subdivision level h. The subdivision polynomial $s(z_1,z_2)$ can be written as

$$s(z_1,z_2) = s_P(z_1z_2, z_1^{-1}z_2)s_P(z_1^2,z_2^2), \qquad (4.22)$$

where

$$\begin{aligned} s_P(z_1,z_2) &= 1/2 + (1/4)(z_1^{1/2}z_2^{1/2} + z_1^{-1/2}z_2^{1/2} + z_1^{1/2}z_2^{-1/2} + z_1^{-1/2}z_2^{-1/2}) \\ &\quad +(1/8)(z_1+z_2+z_1^{-1}+z_2^{-1}). \qquad (4.23) \end{aligned}$$

As in $(4.19)_{/157}$, the subscript P is intended to suggest "partial": multiplying $p(h;z^2)$ by the second factor

$$s_P(z_1^2,z_2^2) = 2\left(\frac{z_1^{1/2}z_2^{1/2}+z_1^{-1/2}z_2^{-1/2}}{2}\right)^2\left(\frac{z_1^{-1/2}z_2^{1/2}+z_1^{1/2}z_2^{-1/2}}{2}\right)^2$$

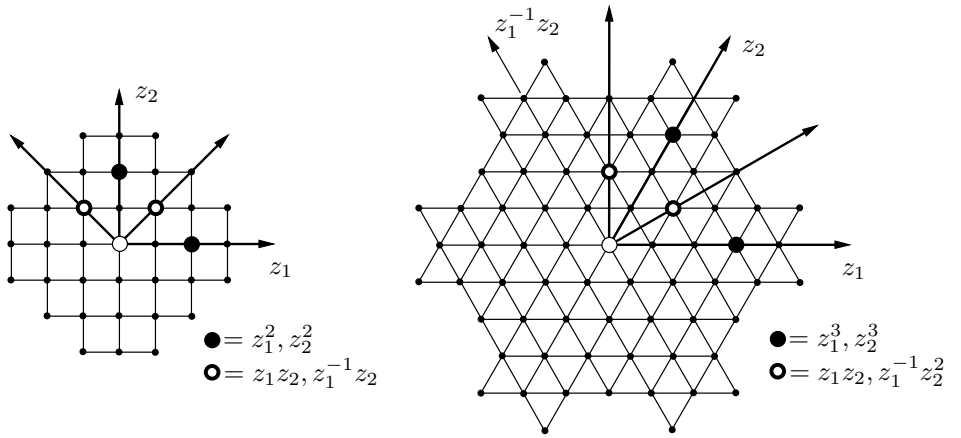


Figure 4.10. *Rotations corresponding to a single substep.*

represents the first substep of the method, and multiplying by the first factor

$$s_P(z_1 z_2, z_1^{-1} z_2) = 2 \left(\frac{z_1^{1/2} + z_1^{-1/2}}{2} \right)^2 \left(\frac{z_2^{1/2} + z_2^{-1/2}}{2} \right)^2$$

represents the second substep. See Section 3.7.2.

It can be seen from $(4.19)_{/157}$ and $(4.22)_{/159}$ that for both the order-8 box-spline method and the $\{\sqrt{3}\}^2$ method, the algorithm replaces the initial tiling with a new tiling that has been rotated and given a change in scale (see Figure $4.10_{/160}$, where the black circles correspond to the coordinates $(1, 0)$ and $(0, 1)$ in the initial coordinate systems, and the black circles with small white dots correspond to the coordinates $(1, 0)$ and $(0, 1)$ in the rotated coordinate systems). The new control points are assigned to the vertices in the new tiling.

In the case of the order-8 box-spline method, the rotation corresponding to a single substep is $\pi/4$, with a change in scale of $1/\sqrt{2}$. In the case of a single substep of $\sqrt{3}$-subdivision, the rotation is $\pi/6$, with a change in scale of $1/\sqrt{3}$. In both cases the new tiling contains the set of vertices belonging to the initial tiling as a subset. In both cases, application of two substeps of the process gives us back the initial tiling, either bisected or trisected.

Each substep of the order-8 box-spline method is, except for a change in the normalization factor[35] a four-direction box-spline subdivision, with directions $(1, 1)$, $(1, 1)^t$, $(-1, 1)^t$, $(-1, 1)^t$ and $(1, 0)^t$, $(1, 0)^t$, $(0, 1)^t$, $(0, 1)^t$, respectively. Each box spline is a bilinear tensor-product spline (linear subdivision): the second factor in $(4.21)_{/159}$ (corresponding to the first substep in the process) is a linear subdivision taken along axes given by the vectors $(1, 1)$ and $(-1, 1)$.

Let us now compare the effect on the control points of s_P, in the $\{\sqrt{3}\}^2$ case, with the effect of s_P on the control points in the order-8 box-spline case. In both cases the new value assigned to a *new* vertex, introduced in the middle of a face, is equal to the centroid of the face.

In the case of *existing* vertices, for $\sqrt{3}$-subdivision, s_P assigns a weight of $2/3$ to the existing vertex, and a weight of $1/3$ to the average of its neighbouring vertices. For the order-8 box-spline method, s_P assigns a weight of $1/2$ to the existing vertex, and a weight of $1/2$ to the average of its neighbouring vertices. This difference between the two methods is relatively minor, since these choices are somewhat arbitrary: the only requirement is that affine invariance be maintained. This difference persists in the nonregular case, which is described in the next section.

In the case of $\sqrt{3}$-subdivision, the assignment of new vertices by s_P produces vertices with double the weight of existing vertices, while reassignment of weights to existing vertices maintains the weight of existing vertices: in total therefore the weight is tripled. In the case of the order-8 box-spline method, assignment of new vertices by s_P produces vertices with weight equal to that of existing vertices, while reassignment of weights to existing vertices maintains the weight of existing vertices: in total the weight is doubled. If we apply these facts to the two substeps, the total overall increase in weight is 9 for the $\{\sqrt{3}\}^2$ method and 4 for the order-8 box-spline method.

4.2.3 Some Generalized-spline methods: $\sqrt{3}$-subdivision, Modified Butterfly, and Kobbelt

In this section, we discuss the three subdivision methods listed in the heading. They are applicable in the general setting of locally planar meshes without boundary in $\mathbb{R}^3$ (Figure $1.30_{/33}$, third column). The regular versions of the methods were introduced in Section 4.2.1.

The $\sqrt{3}$-subdivision scheme (nonregular case)

The $\sqrt{3}$-subdivision method [76] applies to triangular meshes. As described in Section 4.2.1, it uses a splitting schema that is different from $pT4$ splitting: it inserts a new vertex in the middle of each triangle of the mesh, joins them to the original vertices, and then flips the original edges to obtain the subdivided mesh. This procedure can be applied to triangular meshes with nonregular vertices ($n \neq 6$) and can even be extended to the case of nonregular faces ($e \neq 3$). The cases of $n = 5$ and $e = 4$ are illustrated in Figure $4.11_{/162}$, where a pair of $\sqrt{3}$ steps has been applied in each case. As can be seen from the figure, the first pair of subdivision steps may create a new extraordinary vertex, but subsequent steps will introduce no further extraordinary vertices and no extraordinary faces. The schema is analogous to $pT4$ in this respect.

The smoothing rule for $\sqrt{3}$-subdivision specified in [76] assigns a value, for a new vertex introduced in the middle of a triangle of the original mesh, that is equal to the centroid of the vertices of the triangle. (In the case of an extraordinary face, we can simply take the centroid of the vertices of the face.) After the first step, all faces will be triangles, and the expression for a newly introduced face point is

$$p^{new} = \frac{1}{3}(p' + p'' + p'''),$$

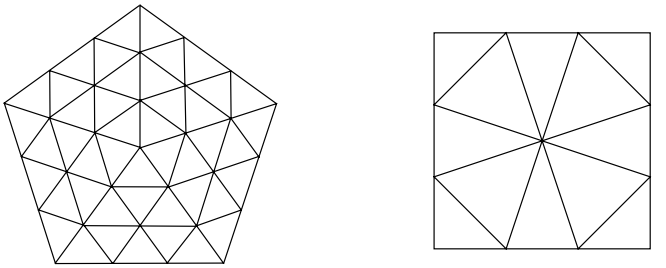


Figure 4.11. *Initial extraordinary vertex and initial extraordinary face.*

where p', p'', and p''' are the control points at the triangle vertices (these are denoted p_i, p_j, and p_k in [76, eqn. (1)]). As for the modified value of a control point p_ℓ corresponding to an existing vertex of valence n, let $p_{\ell_0}, \ldots, p_{\ell_{n-1}}$ be the directly adjacent neighbours in the unrefined mesh. Then, the value of the modified control point is

$$(1 - \alpha_n)p_\ell + \alpha_n \frac{1}{n} \sum_{i=0}^{n-1} p_{\ell_i}, \tag{4.24}$$

where

$$\alpha_n = \frac{4 - 2\cos(2\pi/n)}{9}.$$

Note that $\alpha_6 = 1/3$, so that in the regular case, $(4.24)_{/162}$ becomes

$$\frac{2}{3}p_\ell + \frac{1}{3}\left(\frac{1}{6}\sum_{i=0}^{5} p_{\ell_i}\right),$$

as in $(4.17)_{/155}$.

Like the 4-8 subdivision method, $\sqrt{3}$ subdivision has advantages in the context of adaptive subdivision; see [76, Sec. 4].

Modified Butterfly method

The Butterfly subdivision method was discussed in Section 4.2.1. Two versions were described, one using an 8-point stencil and another using a 10-point stencil; see Figure $4.6_{/154}$. When the parameter w^* is assigned the value 0 in the 10-point scheme, it reduces to the 8-point scheme with $w = 1/16$.

The Butterfly subdivision scheme can be applied in triangular meshes with extraordinary vertices ($n \neq 6$), since it is an interpolating method that uses no vertex stencil. It was noticed, however, that the method may produce cusps at nonregular vertices; see [179, Fig. 1] for an example, and also see the references in Section 5.8. As a result of this observation the Modified Butterfly method was introduced in [178, 179] (the latter reference provides more detail). This method uses modified edge-point rules for edges adjacent to extraordinary vertices.

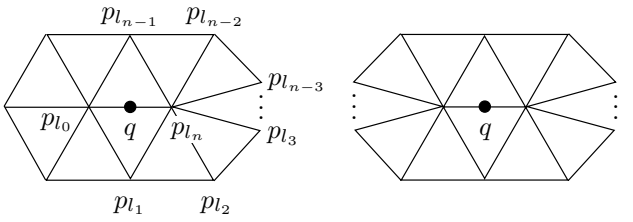


Figure 4.12. *Stencils for Modified Butterfly method.*

The modified edge-point rules are given by the stencils shown in Figure 4.12$_{/163}$ for an extraordinary vertex p_{ℓ_n} of valence n. (The notation in Figure 4.12$_{/163}$ has been chosen to be consistent with that of Figure 4.4$_{/152}$ and the rest of the book. It is *not* consistent with [178, 179]: in particular, in those papers p_{ℓ_n} is denoted by q, and the point that we have denoted q is not given a name.) The left illustration in Figure 4.12$_{/163}$ shows the case where only one neighbour of q is extraordinary, while the right illustration shows the case where both neighbours are extraordinary.

The method used in [178, 179] in the case of two ordinary neighbours is the Butterfly scheme of Section 4.2.1, with $w^* = 0$, as illustrated in Figure 4.6$_{/154}$. (In [178, 179], w^* is denoted by w: we have added a star to avoid the conflict in notation with the parameter w in the basic Butterfly method.) The point q is therefore calculated according to (4.15)$_{/151}$.

In the case of one extraordinary neighbour, with valence $n \neq 6$ (Figure 4.12$_{/163}$, left), the point q is calculated according to

$$q = \frac{3}{4} p_{\ell_n} + \sum_{j=0}^{n-1} s_j p_{\ell_j},$$

where the coefficients s_j are defined as follows. For $n \geq 5$, i.e., for $n = 5, 7, 8, 9, \ldots$,

$$s_j = \frac{1}{n}\big(1/4 + \cos(2\pi j/n) + 1/2\cos(4\pi j/n)\big), \quad j = 0, \ldots, n-1. \tag{4.25}$$

(This rule does not, in the case $n = 6$, reduce to the rule used in the case of two ordinary neighbours.) For $n = 3$, $s_0 = 5/12$ and $s_j = -1/12$ for $j = 1, 2$, while for $n = 4$, $s_0 = 3/8$, $s_2 = -1/8$, and $s_j = 0$ for $j = 1, 3$. In all cases ($n = 3, 4, 5, 7, 8, 9, \ldots$), $\sum_{j=0}^{n-1} s_j = 1/4$, as required for affine invariance.

In the case of two extraordinary neighbours (Figure 4.12$_{/163}$, right), the coefficients are computed according to (4.25)$_{/163}$ for each of the two neighbours, and the average of the values for the two neighbours is used.

The explanation for the choice of weights s_j is given in [178, 179].

Kobbelt method

The Kobbelt method [73], like the Butterfly method, is an interpolating method, but in contrast to the Butterfly method, it is based on the $pQ4$ splitting schema. In the

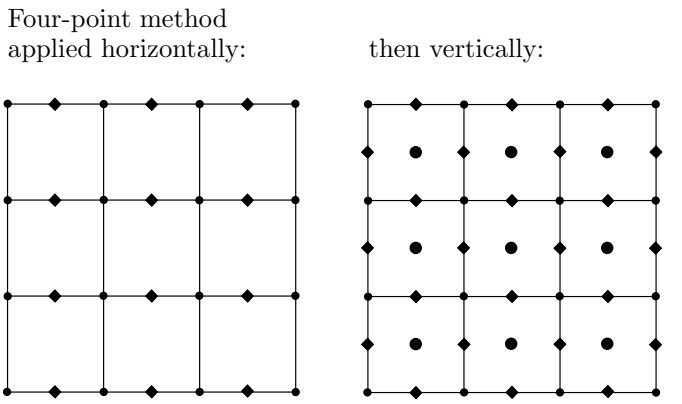


Figure 4.13. *Two substeps of Kobbelt method (regular case).*

regular case the method reduces to the $4pt \times 4pt$ scheme, which is the tensor product with itself of the four-point subdivision method that was discussed in Section 4.2.1.

We assume that the mesh is quadrilateral, possibly with extraordinary vertices ($n \neq 4$). To achieve this, it may be necessary to use one step of, say, the Catmull–Clark method, in order to eliminate extraordinary faces. With this assumption, there remains only the question of how to compute new edge points, and new face points, when there is an extraordinary vertex. No new vertex points are computed, since the method is interpolating.

The calculation of new face points in the regular case can be viewed as made up of two substeps. In the first substep, the four-point subdivision method (Figure $4.3_{/151}$, top) is applied in the horizontal direction to produce edge points in the middle of each horizontal segment in Figure $4.3_{/151}$ (bottom). This is shown in Figure $4.13_{/164}$ (left), where the edge points are shown with diamond-shaped symbols. Then, the four-point subdivision method is applied in the vertical direction to produce edge points, in the middle of each vertical segment, and to produce new face points. The result of this second substep is illustrated in Figure $4.13_{/164}$ (right). Because of the symmetry in the coefficients in Figure $4.3_{/151}$, it does not matter in which order the two substeps are performed.

If we apply the four-point subdivision method in a mesh with an extraordinary vertex, new edge points and new face points can be calculated in the same way, except for edges and faces adjacent to the extraordinary vertex. This is illustrated in Figure $4.14_{/165}$: if the edge points adjacent to p that have been indicated by open circles are computed using the four-point method, the value of a face point, such as the one denoted y_i in the figure, depends on whether we first compute horizontally or vertically. It is not immediately clear, therefore, how the new edge points and face points indicated by open circles in Figure $4.14_{/165}$ should be defined.

The first requirement introduced in [73], in order to fix the values of the edge points and face points, is to require that the evaluation of a face point, such as y_i in Figure $4.14_{/165}$, should not depend on whether the univariate four-point subdivision

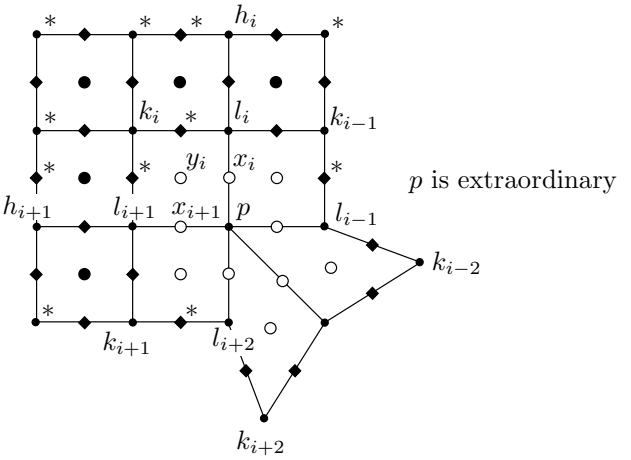


Figure 4.14. *Notation for Kobbelt method.*

rule is applied in the vertical direction or in the horizontal direction. If notation is introduced for each of the points indicated by a $*$ in the figure, it is easy to verify that the requirement just mentioned leads to the following compatibility condition on the x_i's:

$$\begin{aligned} x_{i+1} - x_i &= \beta(l_{i+1} - l_i) + \alpha(h_{i+1} - h_i) + \alpha(l_{i-1} - l_{i+2}) + \frac{\alpha^2}{\beta}(k_{i-2} - k_{i+2}) \\ &= w(h_i - h_{i+1}) + \frac{2w^2}{1+2w}(k_{i-2} - k_{i+2}) \\ &\quad + w(l_{i+2} - l_{i-1}) + \frac{1}{2}(1+2w)(l_{i+1} - l_i), \end{aligned} \tag{4.26}$$

where w is the coefficient in Figure 4.3$_{/151}$ (note the discussion on page 151 concerning the use of ω and w in [73]); the other notation is defined in Figure 4.14$_{/165}$.

The other requirement introduced to fix the values of the edge points and face points is that the centroid of the edge points $\frac{1}{n}\sum_{i=0}^{n-1} x_i$ should give the correct value in the regular case. This value is easily verified to be

$$\frac{1}{2}(1+2w)p + \frac{1}{2n}\sum_{i=0}^{n-1} l_i - \frac{w}{n}\sum_{i=0}^{n-1} h_i \quad \text{(here, } n = 4\text{)}. \tag{4.27}$$

Defining $\Delta x_i = x_{i+1} - x_i$, $i = 0, \ldots, n-1$ (with indices calculated modulo n), it is easy to verify the following identity, using telescoping sums:

$$\frac{1}{n}\sum_{i=0}^{n-1} x_i = x_j + \frac{1}{n}\sum_{i=0}^{n-2}(n-1-i)\Delta x_{i+j}, \quad j = 0, \ldots, n-1. \tag{4.28}$$

If we define the virtual point

$$v_j = \begin{cases} \left(\frac{4}{n}\sum_{i=0}^{n-1} l_i\right) - (l_{j-1} + l_j + l_{j+1}) \\ \quad + \frac{2w}{1+2w}\left(k_{j-2} + k_{j-1} + k_j + k_{j+1} - \frac{4}{n}\sum_{i=0}^{n-1} k_i\right), & n \neq 4, \\ \\ l_{j+2}, & n = 4, \end{cases}$$

and insert $(4.26)_{/165}$ in $(4.28)_{/165}$, we obtain

$$x_j = -wh_j + (1/2 + w)l_j + (1/2 + w)p - wv_j,$$

and when $n = 4$, the centroid of the x_i's is equal to the value given in $(4.27)_{/165}$, as required. (When $n = 3$, we set $l_{i-1} = l_{i+2} = k_{i-2} = k_{i+2}$.) Thus, the edge points x_j can be computed by applying the univariate four-point rule to the points h_j, l_j, p, and v_j, $j = 0, \ldots, n-1$. The face points y_j can then be computed, and it does not matter whether we compute horizontally (using x_i) or vertically (using x_{i+1}).

4.3 Fourier analysis of nodal functions

We now return to the regular case and derive some of the generalizations mentioned at the end of Section 4.1.

We carry out the analysis in the bivariate case. A summary of the necessary changes, to do the analysis in the univariate case, is also given.

Section A.2 provides some background for this section, on Fourier analysis and the use of the delta function. Also, Exercise $5_{/187}$ is intended to provide more detail and intuition for certain of the steps in the analysis given here. There is a slight conflict in notation between the function $S(y)$, to be introduced now, and the local subdivision matrix S, but this should cause no confusion.

The delta function $\delta(y)$ is defined by the requirement [54] that it have its support in the origin, where it is assumed to be "infinite" in such a way that for every continuous function $f : \mathbb{R}^2 \to \mathbb{R}$ we have

$$\int_{\mathbb{R}^2} f(z)\delta(y - z)\, dz = f(y) \tag{4.29}$$

if $z \in \mathbb{R}^2$. Further, for convolution with the delta function we have the following formal relations:

$$f(y) \otimes \delta(y) = f(y),$$
$$f(y - z) \otimes \delta(y) = f(y) \otimes \delta(y - z) = f(y - z).$$

These definitions are consistent with $(4.29)_{/166}$ if we define

$$(f \otimes g)(y) = (f(y) \otimes g(y))(y) = \int_{\mathbb{R}^2} f(z)g(y - z)dz \tag{4.30}$$

in analogy with $(2.3)_{/54}$. For convolution we also have the formulas

$$\widehat{f(ay)}(\omega) = \frac{1}{a^2}\hat{f}(\omega/a) \tag{4.31}$$

if $a \neq 0$ and

$$\widehat{f \otimes g}(\omega) = \hat{f}(\omega)\hat{g}(\omega); \tag{4.32}$$

see the Appendix, Sections A.2 and A.3.

In analogy to $(4.4)_{/146}$, we seek a 2-scale relation

$$N(y) = \sum_{k \in G} s_k N(2y - kh) \tag{4.33}$$

which can be rewritten as

$$N(y) = \sum_{k \in G} s_k (\delta(y - kh/2) \otimes N(2y))(y) \tag{4.34}$$

or

$$N(y) = (S(y) \otimes N(2y))(y) \tag{4.35}$$

if we introduce the function

$$S(y) \doteq \sum_{k \in G} s_k \delta(y - kh/2). \tag{4.36}$$

Next, taking Fourier transforms (again, see Sections A.2 and A.3) we get from $(A.15)_{/312}$ that

$$\widehat{\delta(y - hk)}(\omega) = \exp(-i\omega^t kh)$$

and

$$\hat{S}(\omega) = \sum_{k \in G} s_k \exp(-i\omega^t kh/2). \tag{4.37}$$

Using $(4.31)_{/166}$ and $(4.32)_{/167}$, $(4.35)_{/167}$ can be rewritten as

$$\hat{N}(\omega) = \frac{1}{4}\hat{S}(\omega)\hat{N}(\omega/2). \tag{4.38}$$

In order to determine if there exists some function $N(y)$ satisfying $(4.35)_{/167}$, we pick an initial function $N^0(y)$ that is continuous, with compact support and such that $\int_{\mathbb{R}^2} N^0(y)\, dy = h^2$, but otherwise arbitrary. We then compute recursively,

$$N^{\nu+1}(y) = S(y) \otimes N^\nu(2y), \quad \nu = 0, 1, 2, \ldots. \tag{4.39}$$

Again using the Fourier transform, $(4.39)_{/167}$ may be rewritten as

$$\hat{N}^{\nu+1}(\omega) = \frac{1}{4}\hat{S}(\omega)\hat{N}^\nu(\omega/2), \quad \nu = 0, 1, 2, \ldots. \tag{4.40}$$

The functions in $(4.39)_{/167}$ are well defined, since N^0 is continuous.

Now, using $(4.40)_{/167}$ recursively we obtain

$$\hat{N}^{\nu+1}(\omega) = \frac{1}{4}\hat{S}(\omega)\frac{1}{4}\hat{S}(\omega/2)\frac{1}{4}\hat{S}(\omega/4)\cdots\frac{1}{4}\hat{S}(\omega/2^\nu)\hat{N}^0(\omega/2^{\nu+1}), \quad \nu = 0, 1, 2, \ldots, \tag{4.41}$$

i.e.,

$$\hat{N}^{\nu+1}(\omega) = \hat{S}(\omega)\hat{S}(\omega/2)\hat{S}(\omega/2^2)\cdots\hat{S}(\omega/2^{\nu})\hat{N}^0(\omega/2^{\nu+1})4^{-\nu-1},$$

or

$$\hat{N}^{\nu+1}(\omega) = \prod_{j=0}^{\nu}\Big(\frac{\hat{S}(\omega/2^j)}{4}\Big)\hat{N}^0(\omega/2^{\nu+1}). \tag{4.42}$$

Applying the inverse Fourier transform, $(4.41)_{/167}$ becomes

$$N^{\nu+1}(y) = S(y)\otimes 4S(2y)\otimes\cdots\otimes 4^{\nu}S(2^{\nu}y)\otimes N^0(2^{\nu+1}y), \quad \nu = 0,1,2,\ldots. \tag{4.43}$$

Taking $\omega = 0$ in $(4.42)_{/168}$ we get

$$\hat{N}^{\nu+1}(0) = (\hat{S}(0)/4)^{\nu+1}\hat{N}^0(0), \quad \nu = 0,1,2,\ldots. \tag{4.44}$$

Since $\hat{N}^0(\omega) = \int_{\mathbb{R}^2} N^0(y)\exp(-i\omega^t y)\,dy$, taking $\omega = 0$ gives

$$\hat{N}^0(0) = \int_{\mathbb{R}^2} N^0(y)\,dy = h^2,$$

so that

$$\hat{N}^{\nu+1}(0) = (\hat{S}(0)/4)^{\nu+1}h^2, \quad \nu = 0,1,2,\ldots. \tag{4.45}$$

Note here that from $(4.37)_{/167}$ we have $\hat{S}(0) = \sum_{k\in G} s_k = s(1,1)$.

The function N^0 is continuous with compact support, and it follows that its Fourier transform is continuous. Therefore, considering the equality $(4.42)_{/168}$ we see, since $\hat{N}^0(\omega/2^{k+1}) \to \hat{N}^0(0) = h^2$ as $k\to\infty$, that we have a pointwise limit

$$\hat{N}(\omega) \doteq \lim_{k\to\infty}\hat{N}^{k+1}(\omega) = h^2\prod_{j=0}^{\infty}\frac{\hat{S}(\omega/2^j)}{4}, \tag{4.46}$$

provided that the product in the right-hand side of $(4.46)_{/168}$ converges for each ω. It follows from $(4.46)_{/168}$ that the limit function $\hat{N}(\omega)$ satisfies

$$\hat{N}(\omega) = \frac{\hat{S}(\omega)}{4}\hat{N}(\omega/2) \tag{4.47}$$

or, equivalently, from the convolution theorem and the inverse Fourier transform,

$$N(y) = S(y)\otimes N(2y) = \sum_{k\in G} s_k N(2y-kh), \tag{4.48}$$

as was our goal in $(4.33)_{/167}$.

Note finally that the limit $(4.46)_{/168}$ is independent of $N^0(y)$, which justifies our choice, for this function, of an arbitrary continuous function with compact support.

An exactly parallel development is possible for the univariate case. Corresponding to $(4.31)_{/166}$ we have

$$\widehat{f(ay)}(\omega) = \frac{1}{|a|}\hat{f}(\omega/a) \tag{4.49}$$

if $a \neq 0$, and corresponding to $(4.39)_{/167}$ and $(4.40)_{/167}$ we have

$$N^{\nu+1}(t) = S(t) \otimes N^{\nu}(2t), \quad \nu = 0, 1, 2, \ldots, \tag{4.50}$$

and

$$\hat{N}^{\nu+1}(\omega) = \frac{1}{2}\hat{S}(\omega)\hat{N}^{\nu}(\omega/2), \quad \nu = 0, 1, 2, \ldots, \tag{4.51}$$

and corresponding to $(4.43)_{/168}$ we have

$$N^{\nu+1}(t) = \otimes_{j=0}^{\nu} 2^j S(2^j t) \otimes N^0(2^{\nu+1}t), \quad \nu = 0, 1, 2, \ldots.$$

Equation $(4.37)_{/167}$ also holds in the univariate case, while in $(4.38)_{/167}$, $(4.42)_{/168}$, $(4.46)_{/168}$, and $(4.47)_{/168}$, the factor of 4 in the denominator is replaced in each case by a factor of 2. Finally, in the univariate case, $(4.45)_{/168}$ is replaced by

$$\hat{N}^{\nu+1}(0) = (\hat{S}(0)/2)^{\nu+1}h, \quad \nu = 0, 1, 2, \ldots, \tag{4.52}$$

and $\hat{S}(0) = \sum_{k\in G} s_k = s(1)$.

We next observe, from $(4.46)_{/168}$, that the condition

$$\hat{S}(0) = s(1, 1) = 4 \tag{4.53}$$

is necessary in order that the infinite product in $(4.46)_{/168}$ should converge to a function which is neither identically zero nor identically equal to ∞. (As can be seen from $(A.15)_{/312}$ and $(A.16)_{/313}$, such a function would correspond to a function N that is itself identically zero or infinite, and therefore is useless as a subdivision polynomial.) Similarly, we conclude that in the univariate case,

$$\hat{S}(0) = s(1) = 2 \tag{4.54}$$

is a necessary condition for convergence of the corresponding product to something other than 0 or ∞. These conditions are described more fully in Theorems $4.5.1_{/172}$ and $5.1.3_{/193}$.

Example 4.3.1. Fourier transform for the four-point scheme.

For the four-point subdivision scheme, from $(4.12)_{/149}$ and $(4.37)_{/167}$ we conclude that

$$\frac{1}{2}\hat{S}(\omega) = \cos^4(\omega h/4)(2 - \cos(\omega h/2)). \tag{4.55}$$

Note that $(4.54)_{/169}$ is satisfied. Also, from $(4.13)_{/149}$,

$$S(t) = \frac{1}{16}(-\delta(t + 3h/2) + 9\delta(t + h/2) + 16\delta(t) + 9\delta(t - h/2) - \delta(t - 3h/2)).$$

This should be compared with Figure $4.1_{/150}$. ∎

The preceding analysis is formal in the sense that we have given no conditions on the coefficients of the polynomial $s(z)$ guaranteeing the appropriate convergence of the product $\prod_{j=0}^{\infty}\left(\frac{\hat{S}(\omega/2^j)}{4}\right)$ and reasonable regularity properties for the corresponding function $N(y)$. This is, with the exception of the B-spline and box-spline cases, a difficult task. These questions of convergence and regularity are discussed in Chapter 5.

Note also that the convergence properties of the product in $(4.46)_{/168}$ are directly related to the convergence properties of the subdivision algorithm, when applied to the control point sequence $\{p_l\}_{l\in\mathbb{Z}^2}$ in $(4.6)_{/147}$.

If we have convergence to the curve in $(4.5)_{/146}$ or the surface in $(4.6)_{/147}$, then we can apply the Nodal-Function Computation principle and the Polynomial Coefficient principle.

4.4 Support of nodal functions generated by subdivision polynomials

Assume that the set G is given together with the subdivision polynomial $s(z) = \sum_{k\in G} s_k z^k$. Let $\mathrm{conv}(G)$ denote the convex hull of G.

We then have the following result concerning the support of the function $N(y)$ defined by the subdivision process in the preceding section (provided that N is a well-defined function).

Theorem 4.4.1. $\mathrm{supp}\,(N) \subseteq h\,\mathrm{conv}\,(G)$.

Proof. Consider the repeated convolution

$$\pi_k(y) = S(y) \otimes 4S(2y) \otimes 4^2 S(2^2 y) \otimes \cdots \otimes 4^k S(2^k y) \tag{4.56}$$

appearing in $(4.43)_{/168}$. We need the following well-known lemma about the support of convolutions (which is valid also for delta functions). In the formulation, the sum $A + B$ of two subsets in $\mathbb{R}^2$ is the Minkowski sum defined as

$$A + B \doteq \{a + b : a \in A, b \in B\}.$$

Similarly, if $\alpha \in \mathbb{R}$,

$$\alpha A \doteq \{\alpha a : a \in A\}.$$

Lemma 4.4.2.

$$\mathrm{supp}\,(f \otimes g) \subseteq \mathrm{supp}\,(f) + \mathrm{supp}\,(g).$$

Proof. We carry out the proof only for the case when f and g are continuous functions.[36] We observe that

$$f \otimes g(y) \neq 0$$

implies that for at least one $z \in \operatorname{supp}(g)$ we have $y - z \in \operatorname{supp}(f)$, i.e.,

$$f \otimes g(y) \neq 0 \Longrightarrow y \in \operatorname{supp}(g) + \operatorname{supp}(f),$$

which completes the proof of the lemma. □

Next, using that $\operatorname{supp}(4^j S(2^j y)) = 2^{-j}\operatorname{supp}(S(y))$, we conclude from the lemma that

$$\operatorname{supp}(\pi_k(y)) \subseteq \sum_{j=0}^{k} 2^{-j}\operatorname{supp}(S(y)).$$

Further from

$$S(y) = \sum_{i \in G} s_i \delta(y - ih/2),$$

it follows that $\operatorname{supp}(S(y)) \subseteq (h/2)\operatorname{conv}(G)$ and that

$$\operatorname{supp}(\pi_k(y)) \subseteq (h/2)\sum_{j=0}^{k} 2^{-j}\operatorname{conv}(G). \tag{4.57}$$

Next, we claim that if $A \subseteq \mathbb{R}^2$ is a convex set and if $\alpha > 0$ and $\beta > 0$, then

$$\alpha A + \beta A = (\alpha + \beta)A.$$

In fact any element in $\alpha A + \beta A$ can be written as

$$\alpha a_1 + \beta a_2 = (\alpha + \beta)\left(\frac{\alpha}{\alpha + \beta}a_1 + \frac{\beta}{\alpha + \beta}a_2\right)$$

with a_1 and $a_2 \in A$. Then the convex combination $\frac{\alpha}{\alpha+\beta}a_1 + \frac{\beta}{\alpha+\beta}a_2$ is in A and the claim is proved, since the opposite inclusion is trivial. It now follows that

$$\sum_{j=0}^{k} 2^{-j}\operatorname{conv}(G) = \left(\sum_{j=0}^{k} 2^{-j}\right)\operatorname{conv}(G) = (2 - 2^{-k})\operatorname{conv}(G)$$

and that

$$\operatorname{supp}(\pi_k(y)) \subseteq h(1 - 2^{-k-1})\operatorname{conv}(G) \subset h\operatorname{conv}(G).$$

Considering again $(4.43)_{/168}$, we conclude that

$$\begin{aligned}\operatorname{supp}(\pi_k(y)) &\subseteq h(1 - 2^{-k-1})\operatorname{conv}(G) + \operatorname{supp}(N^0(2^{k+1}y)) \\ &= h(1 - 2^{-k-1})\operatorname{conv}(G) + 2^{-k-1}\operatorname{supp}(N^0(y)).\end{aligned}$$

As $k \to \infty$, the last set shrinks to the origin and we obtain in the limit

$$\operatorname{supp}(N(y)) \subseteq h\operatorname{conv}(G),$$

which completes the proof. □

Generally, the support of N is equal to $h\ \mathrm{conv}(G)$, but we do not give a formal proof of this. Equality holds, however, for all of the standard subdivision methods discussed in the literature.

4.5 Affine invariance for subdivision defined by a subdivision polynomial

We observed in the previous section that $(4.53)_{/169}$ and $(4.54)_{/169}$ are necessary conditions for convergence of the subdivision process in the bivariate and univariate case, respectively. In this section, we formulate additional conditions on the subdivision polynomial being equivalent to affine invariance of a subdivision process based on bisection.

Theorem 4.5.1. *If, in the bivariate case, the subdivision process is defined by some polynomial $s(z) = s(z_1, z_2)$, then it is affine invariant if and only if*

$$s(1,-1) = s(-1,1) = s(-1,-1) = 0 \quad \textit{and} \quad s(1,1) = 4. \tag{4.58}$$

Similarly, in the univariate case with a subdivision polynomial $s(z)$, affine invariance is equivalent to the conditions

$$s(-1) = 0 \quad \textit{and} \quad s(1) = 2.$$

This should be compared with the discussion in Section 2.2.4.

Proof. We give the proof for the bivariate case only. Exercise $6_{/187}$ asks for a proof in the univariate case.

A necessary and sufficient condition for affine invariance is that a shift $p_l := p_l + t$ for the control points, with a constant vector t, should give the same shift for the new control points q_k on the refined grid; see Remark $1.4.1_{/39}$. Using $(4.11)_{/147}$, this is equivalent to the statement that

$$\sum_l s_{k-2l} = 1 \quad \text{for all} \quad k \in \mathbb{Z}^2 \tag{4.59}$$

(consider the vector $t \in \mathbb{R}^N$ with each component equal to 1). Taking $k = (0,0)$, $k = (1,0)$, $k = (0,1)$, and $k = (1,1)$, respectively, we find that the equalities in $(4.59)_{/172}$ are equivalent to the following:

$$\begin{aligned}
s_{ee} \doteq q_{(0,0)} &= \sum_{l \in \mathbb{Z}^2} s_{2l} = 1, \\
s_{oe} \doteq q_{(1,0)} &= \sum_{l \in \mathbb{Z}^2} s_{(1,0)+2l} = 1, \\
s_{eo} \doteq q_{(0,1)} &= \sum_{l \in \mathbb{Z}^2} s_{(0,1)+2l} = 1, \\
s_{oo} \doteq q_{(1,1)} &= \sum_{l \in \mathbb{Z}^2} s_{(1,1)+2l} = 1.
\end{aligned} \tag{4.60}$$

In the notation for these sums "o" stands for odd and "e" for even subindices. We also have

$$\begin{aligned}
s(1,-1) &= \sum_k s_{(k_1,k_2)}(-1)^{k_2} = s_{ee} + s_{oe} - s_{eo} - s_{oo},\\
s(-1,1) &= \sum_k s_{(k_1,k_2)}(-1)^{k_1} = s_{ee} - s_{oe} + s_{eo} - s_{oo},\\
s(-1,-1) &= \sum_k s_{(k_1,k_2)}(-1)^{k_1}(-1)^{k_2} = s_{ee} - s_{oe} - s_{eo} + s_{oo},\\
s(1,1) &= \sum_k s_{(k_1,k_2)} = s_{ee} + s_{oe} + s_{eo} + s_{oo},
\end{aligned} \tag{4.61}$$

which gives the equivalence between $(4.58)_{/172}$ and $(4.60)_{/172}$. □

Exercise $7_{/187}$ suggests a specific application of the theorem for 4-8 subdivision. This is relevant in the context of the comparison in Section 4.2.2.

Condition $(4.58)_{/172}$ is much easier to check than condition $(4.60)_{/172}$, but the latter gives an intuitive interpretation of these two equivalent conditions, in analogy with Remark* $2.2.8_{/72}$. This is examined in Exercise $8_{/187}$, and Exercise $9_{/187}$ asks how Exercise $8_{/187}$ should be modified in order to deal with a method based on trisection, rather than bisection.

4.6 A two-dimensional manifold serving as parametric domain

To support the definition of subdivision surfaces in a general context, we define here a *two-dimensional manifold* $\boldsymbol{M}$ associated with a given locally planar logical mesh M. The manifold $\boldsymbol{M}$ is a topological space such that each point p in $\boldsymbol{M}$ has a neighbourhood that is homeomorphic to either the unit disk in $\mathbb{R}^2$ (p is an interior point) or to a unit half disk in $\mathbb{R}^2$ (p is a boundary point), with p corresponding to the origin in $\mathbb{R}^2$. This manifold can be viewed as a parametric domain for the subdivision surface defined by a subdivision method.

Remark* 4.6.1. The details in this section can be skimmed at first reading, although this may mean that the important discussion at the beginning of Section 4.7, and the important Example $4.7.5_{/184}$, are understood only intuitively. ■

4.6.1 The two-dimensional manifold

The mathematical machinery introduced in order to define the manifold $\boldsymbol{M}$ appears somewhat complicated, but the idea is very simple. Consider first the simple case where the logical mesh can be represented in $\mathbb{R}^3$ in such a way that the logical faces are associated with planar subsets of $\mathbb{R}^3$, as in Figure $1.7_{/11}$ (left). In such a case we have at our disposal a two-manifold (a two-sphere in the case of Figure $1.7_{/11}$) that could be conveniently viewed as a parametric domain for the subdivision surface, and we would like to have such a manifold for arbitrary locally planar logical meshes.

There are, however, some technical problems in doing this. In particular, if we try to use the control points of the corresponding polyhedral mesh $\mathcal{M}$ to construct this representation, as in [18, Sec. 2.3] for example, there remains the problem of what to do if the control points in each face are not coplanar. One possibility [172, Ch. 2] is to identify the polyhedral mesh with a simplicial complex, introducing auxiliary vertices and tagged edges to deal with the case of nontriangular faces. Although this approach presents no theoretical difficulty, it is often more convenient to define the manifold $\boldsymbol{M}$ in terms of faces residing in disjoint spaces. This leads to a simple explicit[37] representation of the parametric domain that is well defined even for nontriangular faces. A similar approach is used in [124], although the focus there is on quadrilateral faces; see Example 4.7.5$_{/184}$, below.

Our definition of the parametric domain introduces the ideas of *chart* and *atlas*. A chart is a local parametric representation of the underlying manifold $\boldsymbol{M}$, and these charts are used very often in convergence and regularity analysis, and in other applications, even if the word "chart" is not explicitly mentioned. A typical example is Stam's method for the parametric evaluation of surfaces near extraordinary vertices: for example, the illustration in Figure 6.3$_{/264}$ is a chart. Detailed definitions are given below.

Our goal here is limited to introducing a parametric domain that is sufficient and convenient for the mathematical analysis of subdivision surfaces. The ideas presented here are, however, relevant for other questions, such as the choice of convenient parametric domains for texture mapping and rendering in computer graphics. Surveys of applications in graphics are given in [59, 146].

First, let us introduce the notation $\mathcal{F}$ for the set of all faces in the logical mesh M. For each $f \in \mathcal{F}$ we consider a copy $\mathbb{R}^2_f$ of the space $\mathbb{R}^2$. This means that for different faces f and g, the Euclidean spaces $\mathbb{R}^2_f$ and $\mathbb{R}^2_g$ are different.

Now, given a logical face $f = (\ell_0, \ell_1, \ldots, \ell_{e-1})$ in $\mathcal{F}$, we first define points $[\ell_j]_f \in \mathbb{R}^2_f$, called vertices, by

$$[\ell_j]_f = r(\cos(2\pi j/e), \sin(2\pi j/e)),\ j = 0, \ldots, e-1, \tag{4.62}$$

where the radius $r = 1/(2\sin(\pi/e))$ is chosen so that the distance between two consecutive vertices $[\ell_j]_f$ and $[\ell_{j+1}]_f$ is equal to 1. See Exercise 10$_{/187}$. Here and in the following definitions, indices are calculated modulo e.

Next, we define the face $F \subset \mathbb{R}^2_f$ by

$$F = \operatorname{conv}\{[\ell_0]_f, [\ell_1]_f, \ldots, [\ell_j]_f, \ldots, [\ell_{e-1}]_f\} \tag{4.63}$$

and the edge $[\ell_j, \ell_{j+1}]_f \subset F$ by

$$[\ell_j, \ell_{j+1}]_f = \operatorname{conv}\{[\ell_j]_f, [\ell_{j+1}]_f\},\ \ j = 0, \ldots, e-1. \tag{4.64}$$

Thus the face F corresponds to the logical face f, the edge $[\ell_j, \ell_{j+1}]_f$ to the logical edge $\{\ell_j, \ell_{j+1}\} \in E_f$, and $[l_j]_f$ to the logical vertex l_j. Each logical face f is associated with the regular closed e-gon defined by (4.62)$_{/174}$ and (4.63)$_{/174}$, having edges and vertices given by (4.64)$_{/174}$ and (4.62)$_{/174}$. See Figure 4.15$_{/175}$.

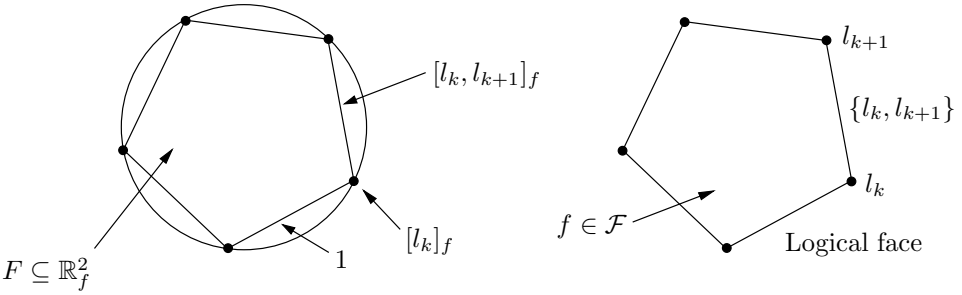


Figure 4.15. *The face represented in $\mathbb{R}^2$, and the corresponding logical face.*

Remark 4.6.2. We denote faces associated with logical faces $f, g, \ldots$ by the corresponding uppercase letters $F, G, \ldots$. If the logical faces $f_0, \ldots, f_\alpha, \ldots, f_{m-1}$ are enumerated by α, the associated faces are denoted by $F_0, \ldots, F_\alpha, \ldots, F_{m-1}$. ∎

We now let the set $\boldsymbol{M}'$ be defined by

$$\boldsymbol{M}' = \bigcup_{f_\alpha \in \mathcal{F}} F_\alpha$$

with the union taken over all faces f_α. Expressing this differently, we can say that $\boldsymbol{M}'$ is the *disjoint union* of all the regular e-gons F. The union is disjoint because each face F_α lies in a different copy of $\mathbb{R}_f$.

Finally, two edges in different faces F and G in $\boldsymbol{M}'$ will be *identified* if they correspond to the same logical edge $\{\ell, \ell'\}$, and in this case the two points

$$x_f = (1-u)[\ell]_f + u[\ell']_f \quad \text{and} \quad y_g = (1-u)[\ell]_g + u[\ell']_g$$

are considered to be identical, or equivalent, for every $u \in [0, 1]$.

Definition 4.6.3. *The set obtained from $\boldsymbol{M}'$ by identifying points on edges in this way is denoted by $\boldsymbol{M}$. Further, we introduce the notation $\boldsymbol{F}$ for the subset of $\boldsymbol{M}$ corresponding to points in the face $F \subset \mathbb{R}^2_f$. The point in $\boldsymbol{M}$ corresponding to $[\ell]_f \equiv [\ell]_g$ is denoted $[\ell]$, and the point in $\boldsymbol{M}$ corresponding to $[\ell']_f \equiv [\ell']_g$ is denoted $[\ell']$. Similarly, the edge corresponding to $[\ell, \ell']_f \equiv [\ell, \ell']_g$ is written $[\ell, \ell']$.*

This definition expresses informally that the set $\boldsymbol{M}$ is obtained by gluing together all the disjoint faces $F, G, \ldots$ along their common logical edges. A more rigorous mathematical definition of $\boldsymbol{M}$ can be given using equivalence classes.

The sets $\boldsymbol{F}$ and $\boldsymbol{G}$ corresponding, respectively, to F and G, sharing the same logical edge $\{\ell, \ell'\}$, are not disjoint. The correspondence between $\boldsymbol{F}$ and F is defined now.

Definition 4.6.4. *The one-to-one mapping*

$$\pi_f : \boldsymbol{M} \supset \boldsymbol{F} \to F \subset \mathbb{R}^2_f$$

is defined by letting the point $\boldsymbol{x} \in \boldsymbol{F}$ be mapped onto the corresponding point x in $F \subset \mathbb{R}^2_f$. If x is an interior point of F, then x and $\boldsymbol{x}$ are identical. In the case that x is contained in some edge in F, then $\boldsymbol{x}$ can be considered as a set of equivalent points from edges of different faces.

Note that if we did not make the distinction between the face $F \subset \mathbb{R}^2_f$ and the face $\boldsymbol{F}$ as a subface of $\boldsymbol{M}$, then π_f would be the identity mapping.

Example 4.6.5. The correspondence between subfaces in $\mathbb{R}^2$ and subfaces of $\boldsymbol{M}$.

A simple example is shown in Figure 4.16$_{/177}$, in a case where the manifold $\boldsymbol{M}$ can be represented as a cube in $\mathbb{R}^3$ (although we emphasize again that there is no requirement that there should exist such a representation). The set of logical vertices is

$$\mathbb{Z}_8 = \{0, 1, 2, \ldots, 7\}$$

and the logical faces are denoted by

$$\begin{aligned} f_0 &= \{0, 1, 2, 3\}, \\ f_1 &= \{0, 3, 4, 7\}, \\ f_2 &= \{0, 1, 6, 7\}, \\ f_3 &= \{2, 3, 4, 5\}, \\ f_4 &= \{1, 2, 5, 6\}, \\ f_5 &= \{4, 5, 6, 7\}. \end{aligned}$$

The six faces $F_0, \ldots, F_5$ are shown in the figure, and for three of these, namely F_0, F_1, and F_2, the details of the vertices $[\ell]$ and certain of the edges $[\ell, \ell']$ are also shown. Each of these six faces lies in a separate copy of $\mathbb{R}^2$. Finally, the figure shows π_{f_0}, which takes $\boldsymbol{F}_0$ onto F_0.

This example will be discussed further, below. ∎

4.6.2 A topology on the manifold

We now define a topology on the abstract point set $\boldsymbol{M}$ defined above.

First, if $\boldsymbol{x}_0 \in \boldsymbol{F}$, and $x_0 = \pi_f(\boldsymbol{x}_0) \in \mathbb{R}^2_f$, then we define the ϵ-neighbourhood of $x_0 \in F$ as

$$\{x \in F : |x - x_0| < \epsilon\} \subset F.$$

Here we also make the following restrictions on the size of ϵ.

1. If x_0 is an interior point of F, then ϵ must be so small that
$$\{x \in \mathbb{R}^2_f : |x - x_0| < \epsilon\} \subset F.$$
2. If x_0 is an interior point $x_0 = (1-t)[\ell]_f + t[\ell']_f$, $0 < t < 1$, of some edge $[\ell, \ell']_f \subset F$, then ϵ must be so small that $\{x \in \mathbb{R}^2_f : |x - x_0| < \epsilon\}$ does not intersect any other edge of F, and in particular it contains no vertices of F.

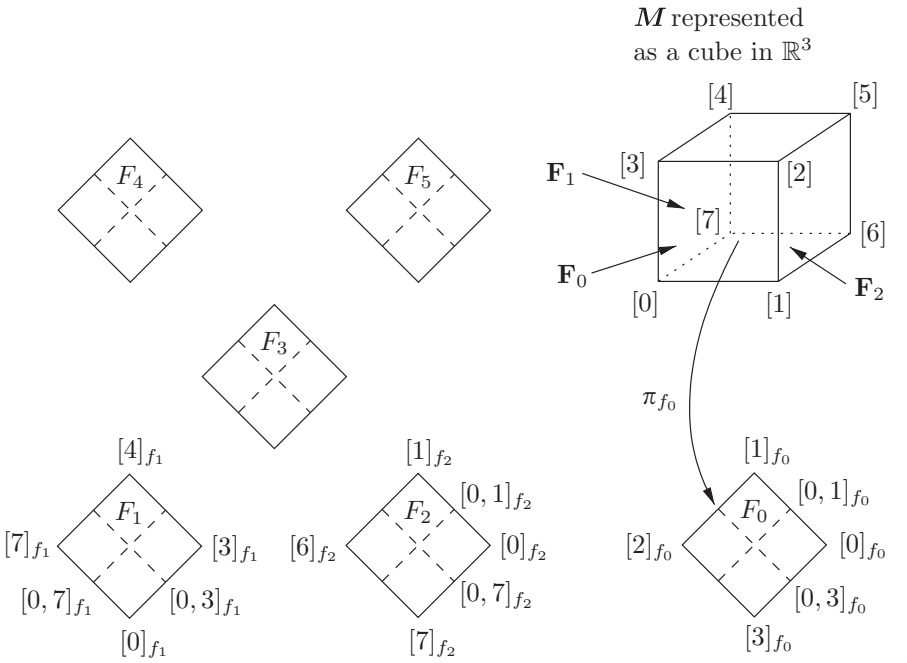


Figure 4.16. *Faces $F_\alpha \subset \mathbb{R}^2_\alpha$, $\alpha = 0, \ldots, 5$, and manifold $\boldsymbol{M}$.*

3. If x_0 is a vertex $[\ell]_f$, then we require that $\epsilon < 1$, so that the ϵ-neighbourhood does not contain any other vertex of F.

   The corresponding ϵ-neighbourhood $B_f(\boldsymbol{x}_0, \epsilon)$ in $\boldsymbol{F}$ is then defined by

$$B_f(\boldsymbol{x}_0, \epsilon) = \pi_f^{-1}(\{x \in F : |x - x_0| < \epsilon\}).$$

Now, if $\boldsymbol{x}_0$ is any point in $\boldsymbol{M}$, then it is contained in at most finitely many faces $\boldsymbol{F}_\alpha$, $0 \le \alpha \le m-1$, corresponding to the logical faces f_α, $0 \le \alpha \le m-1$. We then define the ϵ-neighbourhood $B(\boldsymbol{x}_0, \epsilon) \subset \boldsymbol{M}$ by

$$B(\boldsymbol{x}_0, \epsilon) = \bigcup_{\alpha=0}^{m-1} B_{f_\alpha}(\boldsymbol{x}_0, \epsilon). \tag{4.65}$$

Here ϵ must satisfy all the restrictions originating from the separate faces $\boldsymbol{F}_\alpha$.

We now have the following definition of open and closed sets, which defines the topology on $\boldsymbol{M}$.

Definition 4.6.6. *A subset $\boldsymbol{A} \subset \boldsymbol{M}$ is said to be open if for every $\boldsymbol{x}_0 \in \boldsymbol{A}$ there exists an ϵ-neighbourhood $B(\boldsymbol{x}_0, \epsilon) \subset \boldsymbol{A}$. The subset $\boldsymbol{A}$ is said to be closed if the complement $\boldsymbol{M} \setminus \boldsymbol{A}$ is open.*

We also give the following definition.

Definition 4.6.7. *If $\boldsymbol{A} \subset \boldsymbol{M}$ is a given set, then $\boldsymbol{x}_0 \in \boldsymbol{A}$ is said to be an interior point if there exists an ϵ-neighbourhood $B(\boldsymbol{x}_0, \epsilon) \subset \boldsymbol{A}$. The set of all interior points in $\boldsymbol{A}$ is denoted by $\boldsymbol{A}^0$.*

We note that the set $\boldsymbol{A}^0$ is open, and that $\boldsymbol{A}$ is open if and only if $\boldsymbol{A}^0 = \boldsymbol{A}$.

4.6.3 Local homeomorphisms

For every point $\boldsymbol{x}_0 \in \boldsymbol{M}$ there exists an open subset U containing $\boldsymbol{x}_0$ and a bi-continuous mapping

$$X : U \to \mathbb{R}^2, \tag{4.66}$$

i.e., a mapping that is bijective and with domain homeomorphic to its range $X(U)$. Further, we will show that if we are given any two such open subsets U and V with the corresponding mappings

$$X : U \to \mathbb{R}^2$$

and

$$Y : V \to \mathbb{R}^2,$$

then, if $U \cap V \neq \emptyset$, the mapping

$$Y \circ X^{-1} : X(U \cap V) \to Y(U \cap V) \tag{4.67}$$

and its inverse

$$X \circ Y^{-1} : Y(U \cap V) \to X(U \cap V) \tag{4.68}$$

are continuous and, in fact, piecewise-affine functions.

One says that the point $\boldsymbol{x}_0$, the open set U, and the mapping X define a coordinate patch around $\boldsymbol{x}_0$.

Such a pair (X, U) is also called a *chart* on $\boldsymbol{M}$, and a family of charts with the properties that the charts cover $\boldsymbol{M}$ (every point in $\boldsymbol{M}$ appears in the domain of at least one chart in the family) and are compatible (any two charts (X, U) and (Y, V) in the family satisfy $(4.67)_{/178}$ and $(4.68)_{/178}$) is called an *atlas* on $\boldsymbol{M}$. See, for example, [31] for more details on differential manifolds.

4.6.4 Construction of the local homeomorphisms

It is natural [172] in the context of subdivision surfaces to define a chart for each face, edge, and vertex in the logical mesh. We distinguish the following cases:

a. chart corresponding to the logical face f;

b_i. chart corresponding to an interior edge of some face $\boldsymbol{F}$ (i.e., the corresponding logical edge $\{\ell, \ell'\}$ is an interior edge);

b_e. chart corresponding to an exterior edge of some face $\boldsymbol{F}$ (i.e., the corresponding logical edge $\{\ell, \ell'\}$ is an exterior edge);

c_i. chart corresponding to an interior vertex (i.e., the corresponding logical vertex ℓ is interior);

c_e. chart corresponding to an exterior vertex (i.e., the corresponding logical vertex ℓ is exterior).

In the cases a and b_e, we take $U = \boldsymbol{F}^0$ and $X = \pi_f$. In the case b_i, $[\ell, \ell']$ is contained in exactly two faces F and G, since the logical mesh is locally planar. Then let

$$\mathcal{A} : \mathbb{R}^2_f \to \mathbb{R}^2_g$$

be a rotation followed by a translation ($\mathcal{A}$ is therefore an affine mapping), mapping the set $F \subseteq \mathbb{R}^2_f$ onto a set which is such that $\mathcal{A}(F) \cap G = [\ell, \ell']_g \subseteq G$. Take $U = (\boldsymbol{F} \cup \boldsymbol{G})^0$ and let X be defined by

$$X(\boldsymbol{x}) = \mathcal{A} \circ \pi_f(\boldsymbol{x}) \quad \text{if} \quad \boldsymbol{x} \in \boldsymbol{F} \quad \text{and} \quad X(\boldsymbol{x}) = \pi_g(\boldsymbol{x}) \quad \text{if} \quad \boldsymbol{x} \in \boldsymbol{G}.$$

The definition is consistent, since $\mathcal{A} \circ \pi_f(\boldsymbol{x}) = \pi_g(\boldsymbol{x})$. We note that

$$X(U) = (\mathcal{A}(F) \cup G)^0 \subset \mathbb{R}^2.$$

In Figure 4.16$_{/177}$, for example, the faces F_1 and F_2 in the lower left corner of the figure could play the roles of F and G, respectively, and in this case the mapping $\mathcal{A}$ would carry F_1 into the version of $\mathbb{R}^2$ corresponding to F_2.

In case c_i, the vertex $[\ell]$ belongs to the faces

$$\boldsymbol{F}_0, \boldsymbol{F}_1, \ldots, \boldsymbol{F}_k, \ldots, \boldsymbol{F}_{n-1}, \quad n \geq 3,$$

corresponding to the logical faces $f_0, f_1, \ldots, f_{n-1}$, and to the edges $[\ell, \ell_k]$, $0 \leq k \leq n-1$, enumerated in such a way that

$$[\ell, \ell_k] \cup [\ell, \ell_{k+1}] \subset \boldsymbol{F}_k \quad \text{for} \quad 0 \leq k \leq n-2, \quad \text{and} \quad [\ell, \ell_{n-1}] \cup [\ell, \ell_0] \subset \boldsymbol{F}_{n-1}.$$

We then define affine mappings $\mathcal{A}_k : \mathbb{R}^2_{f_k} \to \mathbb{R}^2$ such that

$$\mathcal{A}_k([\ell]_{f_k}) = (0,0)^t \quad \text{and} \quad \mathcal{A}_k([\ell_k]_{f_k}) = (\cos(2\pi k/n), \sin(2\pi k/n))^t \quad \text{for} \quad 0 \leq k \leq n-1$$

and so that

$$\mathcal{A}_k(F_k) \subset \{(r\cos\varphi, r\sin\varphi)^t : r \geq 0, \varphi \in [k2\pi/n, (k+1)2\pi/n]\} \quad \text{for} \quad 0 \leq k \leq n-1. \tag{4.69}$$

Then take

$$U = \left(\bigcup_{k=0}^{n-1} \boldsymbol{F}_k \right)^0$$

and define the mapping $X : U \to \mathbb{R}^2$ by

$$X(\boldsymbol{x}) = \mathcal{A}_k \circ \pi_{f_k}(\boldsymbol{x}) \quad \text{if} \quad \boldsymbol{x} \in \boldsymbol{F}_k.$$

The definition is consistent since $\mathcal{A}_k \circ \pi_{f_k}(\boldsymbol{x}) = \mathcal{A}_{k'} \circ \pi_{f_{k'}}(\boldsymbol{x})$ if $\boldsymbol{x} \in \boldsymbol{F}_k \cap \boldsymbol{F}_{k'}$.

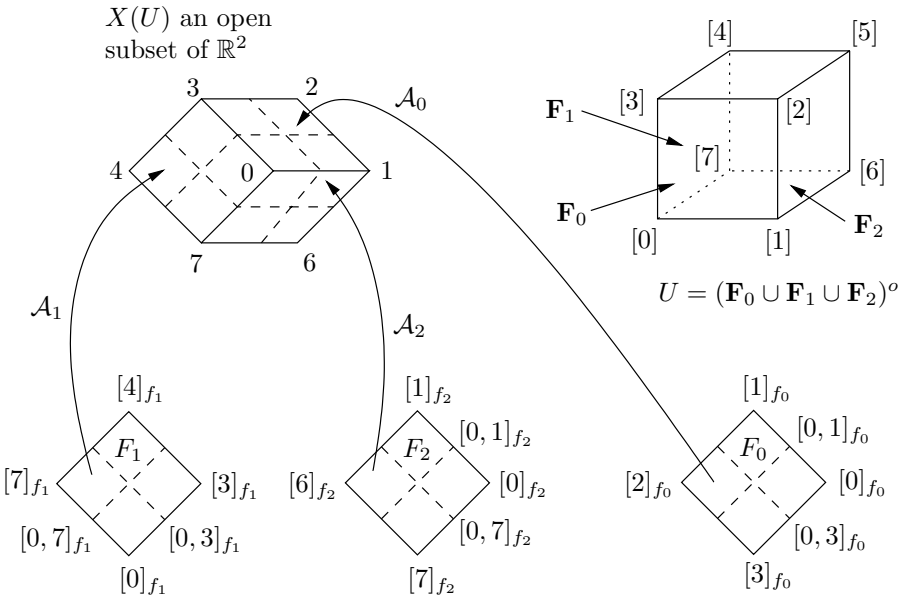


Figure 4.17. *Chart corresponding to an internal vertex.*

We note that

$$X(U) = \left(\bigcup_{k=0}^{n-1} \mathcal{A}_k(F_k) \right)^0 . \tag{4.70}$$

Example 4.6.8. The case of an internal vertex.

Example 4.6.5$_{/176}$ can be extended to illustrate the case c_i, just discussed. Consider the vertex $[\ell] = [0]$ in Figure 4.17$_{/180}$, The edges $[0,1]_{f_0} \subset F_0$ and $[0,1]_{f_2} \subset F_2$ are identified and considered to be equal in $\boldsymbol{M}$. In the same way the edges $[0,3]_{f_1} \subset F_1$ and $[0,3]_{f_0} \subset F_0$ are identified, as well as $[0,7]_{f_1} \subset F_1$ and $[0,7]_{f_2} \subset F_2$. Here we have $n = 3$, $\ell_0 = 1$, $\ell_1 = 3$, $\ell_2 = 7$ and $[0,1] \cup [0,3] \subset F_0$, $[0,3] \cup [0,7] \subset F_1$, and $[0,7] \cup [0,1] \subset F_2$.

The set $X(U) \subset \mathbb{R}^2$ is also shown. ∎

In case c_e, the vertex $[\ell]$ is an exterior vertex belonging to the faces

$$\boldsymbol{F}_0, \boldsymbol{F}_1, \ldots, \boldsymbol{F}_k, \ldots, \boldsymbol{F}_{n-2}, \quad n \geq 2,$$

and to the edges $[\ell, \ell_k]$, $0 \leq k \leq n-1$, enumerated in such a way that

$$[\ell, \ell_k] \cup [\ell, \ell_{k+1}] \subset \boldsymbol{F}_k \quad \text{for} \quad 0 \leq k \leq n-2,$$

and with $[\ell, \ell_0]$ and $[\ell, \ell_{n-1}]$ being exterior edges. Then define affine mappings $\mathcal{A}_k : \mathbb{R}^2_{f_k} \to \mathbb{R}^2$ such that

$$\mathcal{A}_k([\ell]_{f_k}) = (0,0)^t \quad \text{and} \quad \mathcal{A}_k([\ell_k]_{f_k}) = (\cos(\pi k/(n-1)), \sin(\pi k/(n-1)))^t$$
$$\text{for} \quad 0 \leq k \leq n-2$$

and so that

$$\mathcal{A}_k(F_k) \subset \{(r\cos\varphi, r\sin\varphi)^t : r \geq 0, \varphi \in [k\pi/(n-1), (k+1)\pi/(n-1)]\}$$
$$\text{for } \ 0 \leq k \leq n-2. \tag{4.71}$$

The set U is defined by

$$U = \left(\bigcup_{k=0}^{n-2} \boldsymbol{F}_k\right)^0$$

and the mapping $X : U :\to \mathbb{R}^2$ by

$$X(x) = \mathcal{A}_k \circ \pi_{f_k}(x) \ \text{ if } \ x \in \boldsymbol{F}_k.$$

The definition is again consistent since $\mathcal{A}_k \circ \pi_{f_k}(\boldsymbol{x}) = \mathcal{A}_j \circ \pi_{f_j}(\boldsymbol{x})$ if $\boldsymbol{x} \in \boldsymbol{F}_k \cap \boldsymbol{F}_j$, and again

$$X(U) = \left(\bigcup_{k=0}^{n-2} \mathcal{A}_k(F_k)\right)^0. \tag{4.72}$$

4.7 Generalized splines and Generalized-spline subdivision methods

In the case of box-spline surfaces (or uniform B-spline curves), the underlying logical mesh is a doubly infinite rectangular mesh (or a doubly infinite linear mesh), and as we have seen, the resulting surface (curve) is in a natural way associated with a parametrical representation over $\mathbb{R}^2$ (or $\mathbb{R}$) as in $(3.2)_{/93}$, $(2.46)_{/74}$, and $(2.26)_{/63}$. Moreover, referring to $(3.38)_{/120}$ above, the control vector p_l in the representation

$$x(y) = \sum_l p_l N_l(he^m; y) \tag{4.73}$$

is the coefficient for a function $N_l(he^m; y)$ with its support centered at the point $l \in \mathbb{Z}^2 \subset \mathbb{R}^2$ in the parameter space. (In fact, in $(4.73)_{/181}$, the index l varies over $\mathbb{Z}^2$, or $\mathbb{Z}^2 + \bar{e}/2$, as described in Remark $3.5.7_{/119}$ and $(3.38)_{/120}$, but for simplicity we assume here that l varies over $\mathbb{Z}^2$.) This function is defined by the subdivision process by choosing a sequence of scalar control points $\{p_i\}_{i\in\mathbb{Z}^2} \subset \mathbb{R}$ so that $p_l = 1$ and $p_i = 0$ for $i \neq l$. This is the Nodal-Function Computation principle.

In a more general case when the logical mesh does not have the simple structure above (in particular, the mesh may be a finite mesh without boundary, and containing nonregular points), we can still define a parametric representation similar to that in $(4.73)_{/181}$. We assume that the logical mesh M is locally planar and define the two-dimensional manifold $\boldsymbol{M}$ as in Section 4.6. We then obtain an analogous representation

$$x(y) = \sum_\ell p_\ell N_\ell(y), \tag{4.74}$$

where ℓ ranges over all logical vertices in the mesh and $y \in \boldsymbol{M}$. The functions $N_l(y)$ are defined on $\boldsymbol{M}$ with their supports located in some neighbourhood of the corresponding point $[\ell] \in \boldsymbol{M}$.

We now describe how to obtain the representation $(4.74)_{/181}$ and the nodal functions $N_\ell(y)$. Generally, a subdivision process can be described in the following way. In each step of the process the old logical mesh is replaced by a new one according to some specific rule (which for nonstationary processes may be different from step to step). This is done by introducing new vertices on old edges and/or faces, possibly deleting some of the old vertices, and forming new faces by connecting new vertices (obeying the restriction that the new mesh must be locally planar) in a prescribed manner. The control vectors corresponding to the old logical vertices are also replaced according to some specific rule, and in the end the resulting surface in $\mathbb{R}^N$ is obtained by taking limits of the sequence of control vectors.

In step ν of the subdivision process, let us denote the logical mesh by M^ν, its set of logical vertices by $\mathbb{Z}_{L_\nu}$, and the set of control vectors by

$$p^\nu = p^\nu_{(L_\nu \times N)}. \tag{4.75}$$

We also assume that to each control vector p^ν_ℓ we have associated a corresponding point $y_{\ell,\nu} \in \boldsymbol{M}$, and that the sequence of point sets

$$Y_\nu = \{y_{\ell,\nu} : \ell \in \mathbb{Z}_{L_\nu}\} \subset \boldsymbol{M}$$

becomes dense in $\boldsymbol{M}$ as $\nu \to \infty$, in the following sense: for every $y \in \boldsymbol{M}$ there exists a sequence $\{\ell_\nu\}_{\nu=1}^\infty$ such that

$$y_{\ell_\nu,\nu} \to y \quad \text{as} \quad \nu \to \infty.$$

There are several different concepts of convergence for subdivision methods. From a practical point of view, we restrict our attention to *local uniform convergence*, defined in the following way.

Definition 4.7.1. *The subdivision process is said to be locally uniform convergent if for every initial sequence* $\{p^0_\ell\}_{\ell \in \mathbb{Z}_{L_0}} \subset \mathbb{R}^N$ *there exists a continuous function* $x : \boldsymbol{M} \to \mathbb{R}^N$ *such that for every compact subset* $A \subset \boldsymbol{M}$*, we have*

$$\max_{y_{\ell,\nu} \in A} |x(y_{\ell,\nu}) - p^\nu_\ell| \to 0$$

as $\nu \to \infty$.

We note that the limiting function is unique.

Now, if for the initial set of control vectors p^1, we choose the set of scalars

$$p^1_i = \begin{cases} 1 & \text{if } i = \ell, \\ 0 & \text{if } i \neq \ell, \end{cases}$$

then we obtain the functions $N_\ell(y)$ defined on $\boldsymbol{M}$. Thus, in this most general case, we actually use the Nodal-Function Computation principle to *define* the nodal functions. We call a convergent subdivision process defining the nodal functions in this way a *Generalized-spline subdivision method*.

The previous description of how to define the nodal functions N_ℓ does not tell us how to choose the points in $\boldsymbol{M}$ corresponding to the vertices of the logical mesh (or equivalently, to the control vectors) in order to satisfy the assumptions of the previous three paragraphs. This is due to the generality of the definitions. In all subdivision processes in the literature, however, the way to construct the nodal functions and their parametrizations comes quite naturally, as illustrated by the example of Catmull–Clark in Section 5.5.

Definition 4.7.2. *A generalized spline is a linear combination of nodal functions with compact support produced by applying some locally uniformly convergent affine-invariant subdivision process to a scalar control point ($N = 1$) that has value* 1 *at one vertex in the mesh, and* 0 *elsewhere.*

In all the methods that we consider, the nodal functions have compact support,[38] and therefore the Nodal-Function Computation principle gives a (globally) uniformly convergent method, i.e.,

$$\max\{|x(y_{\ell,\nu}) - p_\ell^\nu| : y_{\ell,\nu} \in \boldsymbol{M}\} \to 0$$

as $\nu \to \infty$.

In the following remark, we observe that the nodal functions on which generalized splines are based have the property of *partition of unity*.

Remark 4.7.3. For the sequence of control vectors p^ν, $\nu \geq 1$, we have

$$p^{\nu+1} = \Sigma^\nu p^\nu, \quad \nu = 1, \ldots,$$

with Σ^ν denoting the global subdivision matrix in $(1.13)_{/39}$. In Section 1.4.1, we noted that affine invariance is equivalent to the property that all the row sums of the matrices Σ^ν are equal to one. Then, taking an initial set

$$p_\ell^1 \subset \mathbb{R}^1$$

with $p_\ell^1 = 1$ for all $\ell \in \mathbb{Z}_{L_1}$, it follows that $p_\ell^\nu = 1$ for all $\nu > 1$ and all $\ell \in \mathbb{Z}_{L_\nu}$. We conclude that for the limiting function, we have

$$x(y) = \sum_\ell N_\ell(y) = 1$$

for all $y \in \mathbf{M}$. Consequently, the nodal functions N_ℓ give a partition of unity on the manifold $\mathbf{M}$, provided that the subdivision process is affine invariant. ∎

Remark 4.7.4. Although the surface in $\mathbb{R}^N$ is uniquely defined by the subdivision process, the nodal functions N_ℓ described above are not. In fact they depend on how we choose to do the subdivision of the two-dimensional manifold $\boldsymbol{M}$ corresponding to the subdivision of the logical meshes, i.e., how we assign points $y_{\ell,\nu} \in \boldsymbol{M}$ to points $p_\ell^\nu \in \mathbb{R}^N$. It is also clear that, if we are given one set $\{N_\ell(y)\}$ of nodal functions and if

$$h : \boldsymbol{M} \to \boldsymbol{M}$$

is a homeomorphism, then $\{N_\ell(h(y))\}_{\ell \in \mathbb{Z}_L}$ is another set of nodal functions. ∎

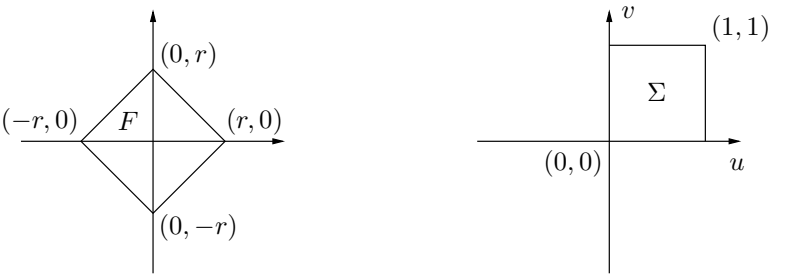


Figure 4.18. *Comparison of parametric domains.*

If in Example 4.6.8$_{/180}$ we assign the control values $p_0 = 1$ and $p_\ell = 0$, for $\ell = 1, 2, \ldots 7$, and apply, for example, Catmull–Clark subdivision, we obtain in the limit the nodal function $N_0(y)$ defined for $y \in \boldsymbol{M}$. In Figure 4.17$_{/180}$ we have also indicated the first step in the splitting of the logical mesh and the corresponding splitting in the parameter domain $X(U)$. It also follows, by the properties of the Catmull–Clark method, that the support of the nodal function $N_0(y)$ is all of $\boldsymbol{M}$, and similarly for every nodal function.

Example 4.7.5. A parametrization of Peters–Reif.

The above ideas can be illustrated by a construction introduced in [124, Chap. 3], which leads to a parametrization involving an *oriented* manifold $\boldsymbol{M}$.

Suppose that all faces in the logical mesh have $e = 4$. (Such an assumption is appropriate, for example, in the case of analysis of a $pQ4$ method, assuming that at least one subdivision step has been executed.) In this case the points $[\ell_j]_f$ are, from (4.62)$_{/174}$, equal to $(r, 0)$, $(0, r)$, $(-r, 0)$, $(0, -r)$, where $r = \sqrt{2}/2$, as illustrated in Figure 4.18$_{/184}$ (left). A typical face $F \subset \mathbb{R}^2_f$ is therefore defined by $\{x \in \mathbb{R}^2 : (\pm 1, \pm 1)x \leq r\}$, and a typical edge by the line segment with endpoints $re^{i\pi j/2}$ and $re^{i\pi(j+1)/2}$, $j = 0, 1, 2, 3$, where $\mathbb{R}^2_f$ is viewed as the complex plane. Indices are calculated modulo 4 throughout this example.

We begin by describing the differences between our parametrization and that used in [124]. First, in the construction of [124], the faces have $e = 4$, since the analysis there focuses on this case. Second, there is the trivial difference that the faces F in [124] are defined as lying in the standard position shown in Figure 4.18$_{/184}$ (right): it is denoted $\Sigma = [0, 1]^2$. The two versions of the face can be obtained from one another by a rigid motion comprising (plus or minus) a rotation of $\pi/4$ and a translation of $(1/2, 1/2)$. The version illustrated in Figure 4.18$_{/184}$ (right) is convenient if only the case $e = 4$ is to be considered. We have used Figure 4.18$_{/184}$ (left) in order to easily describe cases with $e \neq 4$, and we continue to use it in this presentation of the construction of [124], making the necessary straightforward modifications.

For $j = 0, 1, 2, 3$ define

$$\begin{aligned} \kappa_j &= re^{i\pi j/2}, \\ \epsilon_j(u) &= (1-u)\kappa_j + u\kappa_{j+1}, \quad 0 \leq u \leq 1. \end{aligned} \tag{4.76}$$

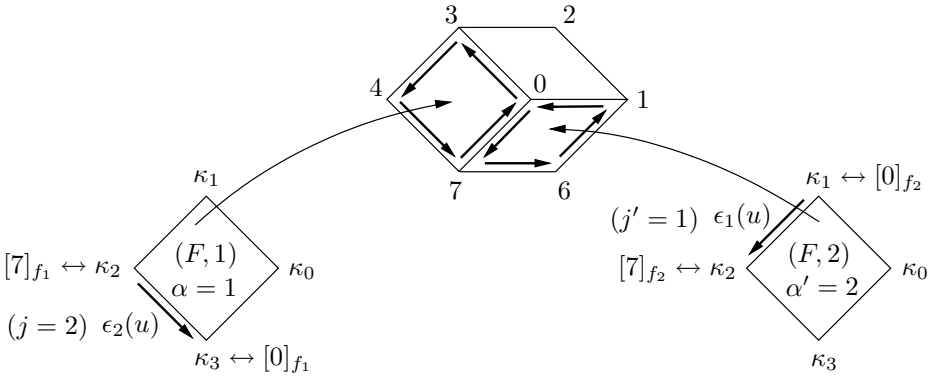


Figure 4.19. *Parametrization near an extraordinary vertex.*

Recall that we have indexed the faces in the logical mesh by α. Peters and Reif [124] call the pair (F, α) a *cell*, and the union $\cup_\alpha (F, \alpha)$ a *spline domain*. They then "stamp" a patch layout on the spline domain by identifying points on edges according to

$$(\epsilon_j(u), \alpha) \sim (\epsilon_{j'}(1-u), \alpha'), \quad 0 \le u \le 1. \tag{4.77}$$

This pointwise identification of edges induces an equivalence relation on the set $\{\kappa_0, \kappa_1, \kappa_2, \kappa_3\} \times \{\alpha\}$ of corners,

$$(\epsilon_j, \alpha) \sim (\epsilon_{j'}, \alpha') \Rightarrow (\kappa_j, \alpha) \sim (\kappa_{j'+1}, \alpha').$$

Each equivalence class is called a *knot*, and pairs of related edges are called *knot lines*.

To illustrate, the construction and linking of the faces in the manifold corresponding to $(0, 7, 6, 1)$ and $(7, 0, 3, 4)$ in the logical mesh illustrated in Figure $4.17_{/180}$ can be accomplished as shown in Figure $4.19_{/185}$. Note that we have had to change the parametrization of F_2 from what was given in Figure $4.17_{/180}$, by reversing the orientation, since we are now required to glue faces together in a way that produces oriented faces. The notation $\leftrightarrow$ is intended only to show the link with our previous presentation.

The equivalences defining knots are

$$[7]_{f_1} \quad \leftrightarrow \quad (\kappa_2, 1) \quad \sim \quad (\kappa_2, 2) \quad \leftrightarrow \quad [7]_{f_2},$$

$$[0]_{f_1} \quad \leftrightarrow \quad (\kappa_3, 1) \quad \sim \quad (\kappa_1, 2) \quad \leftrightarrow \quad [0]_{f_2}$$

and the knot line is defined by

$$(\epsilon_2, 1) \sim (\epsilon_1, 2)$$

with the pointwise correspondence

$$(\epsilon_2(u), 1) \sim (\epsilon_1(1-u), 2), \quad 0 \le u \le 1.$$

The subscripts on the corners κ come from the choices $j = 2$ and $j' = 1$, required for the correspondence $(4.77)_{/185}$: we have

$$\begin{aligned}(\epsilon_j(u), \alpha) &= ((1-u)\kappa_j + u\kappa_{j+1}, \alpha) \\ &= ((1-u)\kappa_2 + u\kappa_3, 1) \\ &\leftrightarrow (1-u)[7]_{f_1} + u[0]_{f_1},\end{aligned}$$

$$\begin{aligned}(\epsilon_{j'}(1-u), \alpha') &= (u\kappa_{j'} + (1-u)\kappa_{j'+1}, \alpha') \\ &= ((1-u)\kappa_2 + u\kappa_1, 2) \\ &\leftrightarrow (1-u)[7]_{f_2} + u[0]_{f_2},\end{aligned}$$

and so the identification required by $(4.77)_{/185}$ has been achieved.

The topology on the manifold can now be introduced in exact analogy with the presentation given above, beginning in Section 4.6.2. ∎

4.8 Additional comments

General subdivision polynomials are discussed in the monograph [25]. For example, the necessary conditions for convergence, discussed in Theorem $4.5.1_{/172}$, can be found in [25, Sec. 2.1]. References for the particular methods discussed in Section 4.2 were given in the text.

For the use of Fourier analysis in the context of subdivision, see [25]. For a more general reference on Fourier analysis, see [153].

The two-dimensional manifold introduced to serve as a parametric domain relies on standard methods in differential geometry [43, 31]. A somewhat different choice of parametric domain was made in [124], as suggested by Example $4.7.5_{/184}$.

The ideas of charts and atlases have recently found wide use in computer graphics in the context of reparametrization for texture mapping [22, 83, 142].

4.9 Exercises

1. Give the global subdivision matrix for the univariate four-point scheme. (The columns have coefficients corresponding to $q_k = \sum_{l\in\mathbb{Z}} s_{k-2}p_l$, and the matrix is analogous to the matrix Σ given for the B-spline case in $(2.16)_{/59}$.)

2. The nodal function for the four-point scheme is not piecewise polynomial. Prove the weaker result that for any $a > 0$, the nodal function has an infinite number of sign changes in the interval $(a, 3h)$, which means that it cannot be piecewise polynomial on this interval.

3. Show that the sum of the coefficients in Figure $4.8_{/156}$ ($\sqrt{3}$-subdivision) is equal to 9. (Compare this exercise with Exercise $7_{/187}$ below.)

4. Show that the points corresponding to z_2, and to $z_1^{-1}z_2^2$, in Figure $4.8_{/156}$, receive a total contribution of 1 when a unit impulse is applied to all of the vertices of the original triangles having edge length h.

5. Consider the derivation in Section 4.3. The following are intended to provide motivation and further explanation of the steps in the derivation.

   (a) Explain in detail how $(4.34)_{/167}$ follows from the equations preceding it.

   (b) Explain what purpose is served by introducing the function S in $(4.36)_{/167}$. Also, in the lines following $(4.34)_{/167}$ we see the main advantage of introducing the Fourier transform: explain what it is.

   (c) The right-hand side of $(4.41)_{/167}$ is a product with several factors, and the right-hand side of $(4.43)_{/168}$ is a multiple convolution involving what can also be referred to loosely as "factors." Explain what the first factor in $(4.41)_{/167}$ and in $(4.43)_{/168}$ corresponds to; similarly, explain what the second factor corresponds to. Finally, relate the overall convolution in $(4.43)_{/168}$ to control vectors.

6. Prove Theorem $4.5.1_{/172}$ in the univariate case.

7. Use Theorem $4.5.1_{/172}$ (4-8 subdivision) to show that the sum of the values of the coefficients shown in Figure $3.29_{/140}$ should be equal to 4. Verify that this is in fact true. (Compare this exercise with Exercise $3_{/186}$ above.)

8. Illustrate the mask for the $LR(3 \times 3)$ subdivision method, i.e., draw the grid illustrating the set G (the points where the coefficients of the subdivision polynomial are not zero), along with the coefficients themselves. Also, draw four corresponding grids for which the sum of the coefficients, according to $(4.60)_{/172}$, should be 1. Finally, verify that these sums are in fact equal to 1 for the $LR(3 \times 3)$ method. Thus, affine invariance of the subdivision process places some constraints on how a subdivision method distributes the weight of a given input control point: the weight must be distributed evenly among the four classes of grid point indicated.

9. Make a conjecture about how $(4.60)_{/172}$ should be modified in the case of a subdivision method based on trisection, rather than bisection. Then (omitting cases once the principle has become clear), repeat Exercise $8_{/187}$ for the $\{\sqrt{3}\}^2$ method, which is based on trisection.

10. Show that if r is chosen to be $1/(2\sin(\pi/e))$ in $(4.62)_{/174}$, then the distance between the two consecutive vertices $[\ell_k]_f$ and $[\ell_{k+1}]_f$ is equal to 1.

Chapter 5

Convergence and Smoothness

The question of convergence has been left open until now, even when the discussion depended on the convergence of the method involved. For example, the Nodal-Function Computation principle depended on the fact that the method under discussion was convergent. This chapter discusses both convergence and smoothness, and for the latter topic, we consider parametric continuity and tangent-plane (or normal) continuity. Other kinds of smoothness are also relevant, and for these we give references to the literature.

We begin with some preliminary results concerning the convergence of subdivision methods defined by subdivision polynomials in one or two variables. First, Theorem 5.1.3$_{/193}$ gives three necessary conditions for convergence of a subdivision process. Following this is the statement of Theorem 5.1.4$_{/195}$, which shows that if two subdivision polynomials defining convergent processes are given, then their product defines a new convergent process, and the nodal function for the new process is the convolution of those for the factors. This result is proved in the Appendix.

We then consider specific classes of methods. For box-spline schemes we prove very precise and general results on convergence, but results on convergence for General-subdivision-polynomial schemes are more difficult to obtain. In the latter case we give some results based on elementary analyses and, for results depending on more elaborate methods of analysis, we give references to the literature.

In the nonregular case a fairly detailed exposition is given, using the Catmull–Clark method as an example. This analysis shows the main ideas related to convergence and smoothness in the nonregular case. Spectral analysis of a subdivision matrix corresponding to a 2-ring neighbourhood of the nonregular point is used to show convergence, while spectral analysis of a matrix corresponding to a larger neighbourhood is used to establish regularity of a map called the characteristic map. Then, injectivity of the characteristic map is used to show single sheetedness, i.e., to show that the process converges to a well-defined surface. The chapter concludes with a section giving suggestions for further reading on the extensive subject of the relationship between surface shape and the spectral properties of subdivision matrices.

5.1 Preliminary results for the regular case

We first present certain general results concerning the convergence of subdivision methods defined by subdivision polynomials. Thus, we restrict our attention for now to the regular case (Figure $1.30_{/33}$, lower row). We analyse only the bivariate case: corresponding results for the univariate case can be obtained by obvious modifications. It is sufficient to study the case of convergence of methods applied to the unit-impulse function, towards a nodal function $N(y)$, since the results will then apply immediately to any finite linear combination of such functions defining $x(y)$.

In Section 4.1 we introduced a general subdivision polynomial $\sum_{k\in G} s_k z^k$ and asked (see $(4.33)_{/167}$ in Section 4.3) if there exists a continuous function $N(y)$, with compact support, satisfying

$$N(y) = \sum_{k\in G} s_k N(2y - kh) = \sum_{k\in G} s_k N(2(y - kh/2)), \tag{5.1}$$

or equivalently

$$N(y) = \left(\sum_{k\in G} s_k z^k\right) N(2y) = s(z)N(2y), \tag{5.2}$$

where, as usual, z denotes translation by $h/2$. As in Section 4.1, the finite set G is a subset of $\mathbb{Z} + (\epsilon_1/2, \epsilon_2/2)$, $\epsilon_i \in \{0, 1\}, i = 1, 2$, but the presentation is phrased in terms of the case $\epsilon_1 = \epsilon_2 = 0$. There is no assumption that s or G is symmetric in the origin.

It is convenient to extend the coefficients s_k to the index set $\mathbb{Z}^2$ by defining $s_k = 0$ if $k \in \mathbb{Z}^2 \setminus G$. Then we may write, for example,

$$N(y) = \sum_{k\in\mathbb{Z}^2} s_k N(2y - kh).$$

We also require that N should be normalized so that $\int_{\mathbb{R}^2} N(y)\, dy = h^2$ in the bivariate case. In Section 4.3 it was shown that $(5.1)_{/190}$ can be rewritten as

$$N(y) = (S(y) \otimes N(2y))(y),$$

where $S(y) = \sum_{k\in G} s_k \delta(y - kh/2)$ and δ denotes the delta function, and that its Fourier transform can be expressed as

$$\hat{N}(\omega) = \frac{\hat{S}(\omega)}{4} \hat{N}(\omega/2), \tag{5.3}$$

where $\hat{S}(\omega) = \sum_{k\in G} s_k \exp(-i\omega^t kh/2)$. We also showed, in Section 4.4, that

$$\operatorname{supp}(N) \subseteq \operatorname{conv}(hG).$$

If such a nodal function exists, then it must be unique. In fact, using $(5.3)_{/190}$ we get

$$\hat{N}(\omega) = \hat{N}(\omega/2^\nu) \prod_{j=0}^{\nu-1} \frac{\hat{S}(\omega/2^j)}{4},$$

and since $\hat{N}(\omega/2^\nu) \to \hat{N}(0) = \int_{\mathbb{R}^2} N(y)\,dy$ for every ω, we have

$$\hat{N}(\omega) = \hat{N}(0) \prod_{j=0}^{\infty} \frac{\hat{S}(\omega/2^j)}{4}.$$

See $(4.46)_{/168}$. This implies that the Fourier transform $\hat{N}(\omega)$ is unique, and hence $N(y)$ is unique. In the following, we use the notation $N^s(h;y)$ for such a nodal function when we wish to emphasize the dependence on the particular polynomial $s(z)$ and the parameter h, and otherwise $N(h;y)$, $N^s(y)$, or $N(y)$. It is shown in Theorem $5.1.3_{/193}$ that if the subdivision process is convergent, it is necessary that $\hat{N}(0) = h^2$, as we have required.

Now consider a generalized-subdivision-polynomial surface given on its initial grid as

$$x(y) = \sum_{l\in\mathbb{Z}^2} p_l^0 N_l(h;y) = \sum_{l\in\mathbb{Z}^2} p_l^0 N_l(h;y-lh) = p^0(z^2)N(h;y), \tag{5.4}$$

where we have introduced the generating function $p^0(z)$, related to the initial grid $h\mathbb{Z}^2$, defined by

$$p^0(z) = \sum_{l\in\mathbb{Z}^2} p_l^0 z^l.$$

The control vectors p_l^0, $l \in \mathbb{Z}^2$, here are associated with parameter values $y = lh$. We also remark that if a bivariate polynomial $p(z) = p(z_1, z_2)$ is given, then $p(z^2)$ denotes the polynomial $p(z_1^2, z_2^2)$. Similarly, below we introduce fractional powers such as $p(z^{1/2^\nu}) = p(z_1^{1/2^\nu}, z_2^{1/2^\nu})$.

On the refined grid $h\mathbb{Z}^2/2$ we then have the representation

$$x(y) = \sum_{l} p_l^1 N_l(h;2(y-lh/2)) = \sum_{l} p_l^1 z^l N(h;2y) = p^1(z)N(h;2y),$$

where

$$p^1(z) = \sum_{k\in\mathbb{Z}^2} p_k^1 z^k$$

and where the control vectors p_l^1, $l \in \mathbb{Z}^2$, are related to parameter values $y = lh/2$. Inserting $(5.2)_{/190}$ we get

$$p^1(z) = s(z)p^0(z^2).$$

The displayed equations just given correspond to $(4.9)_{/147}$ and $(4.10)_{/147}$, which used a notation that emphasized the link with the previously developed B-spline and box-spline cases. The notation here is slightly different and is chosen to facilitate study of the result of repeating the process many times. Thus, repeating the procedure recursively we get

$$p^{\nu+1}(z) = s(z)p^\nu(z^2) \tag{5.5}$$

for $\nu = 0, 1, \ldots$, i.e.,

$$p^{\nu+1}(z^{1/2^\nu}) = s(z^{1/2^\nu})s(z^{1/2^{\nu-1}})s(z^{1/2^{\nu-2}})\cdots s(z^{1/2})s(z)p^0(z^2), \tag{5.6}$$

where the refined generating function $p^\nu(z) = \sum_{l\in\mathbb{Z}^2} p_l^\nu z^l$ is related to the grid $h\mathbb{Z}^2/2^\nu$. The index in the coefficient p_l^ν is related to the position $lh/2^\nu \in \mathbb{R}^2$ in the parameter domain and we have

$$\begin{aligned} x(y) &= \sum_{l\in\mathbb{Z}^2} p_l^\nu N_l(h; 2^\nu(y - lh/2^\nu)) \\ &= \left(\sum_{l\in\mathbb{Z}^2} p_l^\nu z^{l/2^{\nu-1}}\right) N(h; 2^\nu y) \\ &= p^\nu(z^{1/2^{\nu-1}}) N(h; 2^\nu y). \end{aligned}$$

Using the recursion relation $(5.5)_{/191}$, we get

$$p_k^{\nu+1} = \sum_{l\in\mathbb{Z}^2} s_{k-2l} p_l^\nu \tag{5.7}$$

for $\nu = 0, 1, 2, \ldots$, as in $(4.11)_{/147}$. If we now take $p^0(z) \equiv 1$, i.e., $p_0^0 = 1$ and $p_l^0 = 0$ if $l \neq 0$, then

$$x(y) = N(h; y)$$

in $(5.4)_{/191}$, and from $(5.6)_{/191}$ we conclude that

$$p^{\nu+1}(z^{1/2^\nu}) = p^\nu(z^{1/2^\nu}) s(z), \tag{5.8}$$

or equivalently

$$p^{\nu+1}(z) = p^\nu(z) s(z^{2^\nu}). \tag{5.9}$$

(Exercise $1_{/245}$ asks that this, and two of the other steps in the derivation being presented here, be explained in slightly more detail.) This gives the following recursion formula:

$$p_k^{\nu+1} = \sum_{i\in G} s_i p_{k-i2^\nu}^\nu. \tag{5.10}$$

To obtain the coefficients in $p^\nu(z^{1/2^{\nu-1}})$ for $\nu \geq 1$ we write

$$\begin{aligned} p^\nu(z^{1/2^{\nu-1}}) &= \prod_{j=0}^{\nu-1} s(z^{1/2^j}) \\ &= \prod_{j=0}^{\nu-1} \left(\sum_{i_j\in\mathbb{Z}^2} s_{i_j} z^{i_j/2^j}\right) \\ &= \sum s_{i_0} s_{i_1} s_{i_2} \cdots s_{i_{\nu-1}} z^{i_0 + i_1/2 + i_2/4 + \cdots + i_{\nu-1}/2^{\nu-1}} \\ &= \sum_{i\in\mathbb{Z}^2} p_i^\nu z^{i/2^{\nu-1}}, \end{aligned} \tag{5.11}$$

where

$$p_i^\nu = \sum_{i/2^{\nu-1} = i_0 + i_1/2 + \cdots + i_{\nu-1}/2^{\nu-1}} s_{i_0} s_{i_1} s_{i_2} \cdots s_{i_{\nu-1}}. \tag{5.12}$$

In $(5.12)_{/192}$ the summation is, for a fixed index $i \in \mathbb{Z}^2$, over all combinations of indices $i_0, i_1, \ldots, i_{\nu-1}$ in $\mathbb{Z}^2$ such that $i/2^{\nu-1} = i_0 + i_1/2 + \cdots + i_{\nu-1}/2^{\nu-1}$.

We now turn to the question of convergence.

Definition 5.1.1. *We say that the subdivision process with $p^0(z) \equiv 1$ converges uniformly towards a continuous limit function, which we denote by $N^s(y)$, if*

$$\max_{k\in\mathbb{Z}^2} |N^s(kh/2^\nu) - p_k^\nu| \doteq \epsilon_\nu^s \to 0 \tag{5.13}$$

as $\nu \to \infty$. This is equivalent to the statement that the function sequence $P_\nu(y)$, defined by the requirement that $P_\nu(y)$ is continuous everywhere, that $P_\nu(kh/2^\nu) = p_k^\nu$ for all $k \in \mathbb{Z}$, and that it is bilinear on all squares $2^{-\nu}h\{y \in \mathbb{R}^2 : k_1 \le u \le k_1 + 1, k_2 \le v \le k_2 + 1\}$, should converge uniformly, i.e.,

$$P_\nu(y) \to N^s(y) \quad \textit{uniformly as} \quad \nu \to \infty.$$

This definition should be compared with Definition $4.7.1_{/182}$, which is applicable in a more general situation.

Remark 5.1.2. In $(5.12)_{/192}$ the sum is over $i \in (2^\nu \mathrm{conv}(G)) \cap \mathbb{Z}^2$ in the bivariate case and $i \in (2^\nu \mathrm{conv}(G)) \cap \mathbb{Z}$ in the univariate case. This means that the number of nonzero coefficients p_i^ν is limited by $C4^\nu$ in the bivariate case and by $C2^\nu$ in the univariate case, where C is some constant.

From Theorem $4.4.1_{/170}$, the function N^s in $(5.13)_{/193}$ has compact support, and since it is explicitly assumed to be continuous, it must also be bounded. ∎

Before formulating the following theorem we recall from Theorem $4.5.1_{/172}$ that, in the bivariate case, the subdivision process defined by a polynomial $s(z) = s(z_1, z_2)$ is affine invariant if and only if

$$s(1,-1) = s(-1,1) = s(-1,-1) = 0 \quad \text{and} \quad s(1,1) = 4,$$

or, equivalently, if and only if $\sum_l s_{k-2l} = 1$ for all $k \in \mathbb{Z}^2$. In the univariate case the condition is $s(-1) = 0$ and $s(1) = 2$.

Theorem 5.1.3. *Assume that the General-subdivision-polynomial subdivision process converges in the sense of Definition $5.1.1_{/193}$ to a function that is not identically zero. Then*

- *the process is affine invariant,*
- *the limit function $N(y)$ satisfies the 2-scale relation*

$$N(y) = \sum_{k\in G} s_k N(2y - kh) = s(z)N(2y),$$

- *the limit function $N(y)$ satisfies*

$$N(h; y) = N(1; y/h) \tag{5.14}$$

and

$$\int_{\mathbb{R}^2} N(h;y)\,dy = h^2. \tag{5.15}$$

Proof. Adding and subtracting terms in $(5.7)_{/192}$, we get

$$\begin{aligned} &p_k^{\nu+1} - N(kh/2^{\nu+1}) \\ &= \sum_{l\in\mathbb{Z}^2} s_{k-2l}(p_l^\nu - N(lh/2^\nu)) + \sum_{l\in\mathbb{Z}^2} s_{k-2l}(N(lh/2^\nu) - N(kh/2^{\nu+1})) \\ &\quad + N(kh/2^{\nu+1})\left(\sum_{l\in\mathbb{Z}^2} s_{k-2l} - 1\right). \end{aligned}$$

By $(5.13)_{/193}$ and the fact that G is a finite set, we conclude that the left-hand side and the first term on the right-hand side tend to zero as $\nu\to\infty$. Since $k-2l\in G$, we have that $G/2^{\nu+1}\ni k/2^{\nu+1}-l/2^\nu\to 0$ as $\nu\to\infty$, and therefore also the second term on the right-hand side tends to zero. Consequently, $N(kh/2^{\nu+1})\left(\sum_{l\in\mathbb{Z}^2} s_{k-2l}-1\right)\to 0$ as $|k|$ and ν tend to ∞. Let $k^{ij}=(i,j)^t$, where $i,j\in\{0,1\}$. For fixed (i,j), every $k\in\mathbb{Z}^2$ can be written as $k=k^{ij}+2r$ for some $r\in\mathbb{Z}^2$. Further, it is clear that $\sum_{l\in\mathbb{Z}^2} s_{k-2l}=\sum_{l\in\mathbb{Z}^2} s_{k^{ij}-2l}$. Now choose $y\in\mathbb{R}^2$ such that $N(y)\neq 0$ and a sequence $\{k_\nu\}_{\nu=0}^\infty\subset\mathbb{Z}^2$ such that $k_\nu=k^{ij}+2r_\nu$ and $k_\nu/2^{\nu+1}\to y$ as $\nu\to\infty$. Then

$$N(k_\nu h/2^{\nu+1})\left(\sum_{l\in\mathbb{Z}^2} s_{k^{ij}-2l}-1\right)\to N(y)\left(\sum_{l\in\mathbb{Z}^2} s_{k^{ij}-2l}-1\right)=0,$$

and we conclude that $\sum_{l\in\mathbb{Z}^2} s_{k^{ij}-2l}-1=0$ for each of the four vectors k^{ij}, and the first statement of the theorem is proved.

For the proof of the second statement we use $(5.10)_{/192}$. Adding and subtracting terms, we have

$$\begin{aligned} &(p_k^{\nu+1} - N(kh/2^{\nu+1})) + N(kh/2^{\nu+1}) \\ &= \sum_{i\in G} s_i(p_{k-i2^\nu}^\nu - N(kh/2^\nu - ih)) + \sum_{i\in G} s_i N(kh/2^\nu - ih). \end{aligned}$$

Using $(5.13)_{/193}$ we conclude that

$$\left|N(kh/2^{\nu+1}) - \sum_{i\in G} s_i(p_{k-i2^\nu}^\nu - N(kh/2^\nu - ih))\right| \le 2\epsilon_\nu$$

for all k and i. Now for given $y\in\mathbb{R}^2$ choose a sequence $\{l_\nu\}_{\nu=0}^\infty\subset\mathbb{Z}^2$ such that $l_\nu h/2^{\nu+1}\to y$ as $\nu\to\infty$. Taking limits, we obtain the second statement of the theorem.

Next, we observe that the subdivision procedure defines the coefficients p_k^ν recursively by $p_k^{\nu+1}=\sum_{l\in\mathbb{Z}^2} s_{k-2l}p_l^\nu$ with $p_0^0=1$ and $p_l^0=0$ if $l\neq 0$. This means

that the values p_k^ν are related to the parameter value $kh/2^\nu$, $k \in \mathbb{Z}^2$, independently of the value of h. From this we conclude that $(5.14)_{/193}$ is valid and that it suffices to prove $(5.15)_{/194}$ for $h = 1$.

We have

$$\sum_{k\in\mathbb{Z}^2} N(k/2^\nu)4^{-\nu} \to \int_{\mathbb{R}^2} N(y)\,dy$$

as $\nu \to \infty$ since the integrand in the right-hand side is continuous with compact support and the left-hand side is its Riemann sum with the grid-size $2^{-\nu}$. Moreover, from $(5.13)_{/193}$ it then follows that

$$\sum_{k\in\mathbb{Z}^2} 4^{-\nu} p_k^\nu \to \int_{\mathbb{R}^2} N(y)\,dy.$$

By the recursion relation $(5.7)_{/192}$ we have

$$p_k^\nu = \sum_{l:k-2l\in G} s_{k-2l} p_l^{\nu-1} \quad \text{for} \quad k \in \mathbb{Z}^2.$$

Summing over k and using that $\sum_{k\in\mathbb{Z}^2} s_{k-2l} = \sum_{k\in\mathbb{Z}^2} s_k = 4$, we conclude that

$$\sum_{k\in\mathbb{Z}^2} p_k^\nu = 4 \sum_{k\in\mathbb{Z}^2} p_k^{\nu-1} = \cdots = \cdots = 4^\nu \sum_{k\in\mathbb{Z}^2} p_k^0 = 4^\nu$$

for all ν, and it follows that

$$1 = \sum_{k\in\mathbb{Z}^2} 4^{-\nu} p_k^\nu \to \int_{\mathbb{R}^2} N(y)\,dy;$$

i.e., the constant sequence with all terms equal to 1 converges to the integral on the right. The proof is now complete. □

We now state a theorem concerning the convergence of subdivision procedures defined by subdivision polynomials that are the product of subdivision polynomials corresponding to other convergent procedures. The theorem is proved in the Appendix (Section A.4).

Theorem 5.1.4. *Assume that we are given two subdivision polynomials $s(z)$ and $w(z)$ defining convergent subdivision processes in the sense of Definition* $5.1.1_{/193}$. *Then in the bivariate case the polynomial $\psi(z) = s(z)w(z)/4$ also defines a convergent process producing the continuous nodal function*

$$N^\psi(h;y) = \frac{1}{h^2} N^s(h;y) \otimes N^w(h;y). \tag{5.16}$$

In the univariate case the same conclusion is valid with $\psi(z) = s(z)w(z)/2$ and with the factor $\frac{1}{h}$ in the right-hand sides of $(5.16)_{/195}$ *and* $(A.35)_{/321}$. *Further, the convergence for the subdivision associated with the polynomial ψ is uniform in the sense of Definition* $5.1.1_{/193}$.

Theorem 5.1.4$_{/195}$ is interesting, for example, in connection with box splines. Thus, in the context of the four-direction box splines discussed in Example 3.2.7$_{/104}$, the theorem states that the Zwart–Powell element (the nodal function generated by the $\{\text{Midedge}\}^2$ method in the regular case), convolved with itself, is the nodal function generated by 4-8 subdivision in the regular case; see Section 3.7.2. In fact, from (3.58)$_{/136}$ we have for the $\{\text{Midedge}\}^2$ method the centered subdivision polynomial

$$s_{ME}(z) \doteq \frac{1}{4} z_1^{-1/2} z_2^{-3/2} (1+z_1)(1+z_2)(1+z_1 z_2)(1+z_1^{-1} z_2), \tag{5.17}$$

and according to (3.59)$_{/137}$, $s_{ME}(z) s_{ME}(z)/4$ is the centered subdivision polynomial corresponding to the 4-8 subdivision method, i.e.,

$$s_{4\text{-}8}(z) \doteq \frac{1}{64 z_1 z_2^3} (1+z_1)^2 (1+z_2)^2 (1+z_1 z_2)^2 (1+z_1^{-1} z_2)^2, \tag{5.18}$$

which provides a confirmation of Theorem 5.1.4$_{/195}$.

It should be noted, however, that not every factorization of a subdivision polynomial gives a useful result. For example, from (4.22)$_{/159}$ we have that

$$s_{4\text{-}8}(z) = s_P(z_1 z_2, z_1^{-1} z_2) s_P(z_1^2, z_2^2), \tag{5.19}$$

where

$$\begin{aligned} s_P(z_1, z_2) = 1/2 &+ (1/4)(z_1^{1/2} z_2^{1/2} + z_1^{-1/2} z_2^{1/2} + z_1^{1/2} z_2^{-1/2} + z_1^{-1/2} z_2^{-1/2}) \\ &+ (1/8)(z_1 + z_2 + z_1^{-1} + z_2^{-1}). \end{aligned}$$

The method defined by twice the second factor on the right in (5.19)$_{/196}$, with subdivision polynomial $s(z_1, z_2)$ defined by

$$2 s_P(z_1^2, z_2^2) = 1 + \frac{1}{2}(z_1 z_2 + z_1^{-1} z_2 + z_1 z_2^{-1} + z_1^{-1} z_2^{-1}) + \frac{1}{4}(z_1^2 + z_2^2 + z_1^{-2} + z_2^{-2}),$$

is not affine invariant, and therefore not convergent in the sense of Definition 5.1.1$_{/193}$. (It must be used in alternation with the first factor on the right in (5.19)$_{/196}$ to produce a convergent method.) That the method is not affine invariant follows from Theorem 4.5.1$_{/172}$. In fact, $s(1,1) = 4$, and $s(1,-1) = s(-1,1) = 0$, but $s(-1,-1) = 4 \neq 0$. On the other hand, the method defined by twice the *first* factor on the right in (5.19)$_{/196}$, i.e., the method with subdivision polynomial $s(z_1, z_2)$ defined by

$$2 s_P(z_1 z_2, z_1^{-1} z_2) = 1 + \frac{1}{2}(z_2 + z_1^{-1} + z_1 + z_2^{-1}) + \frac{1}{4}(z_1 z_2 + z_1^{-1} z_2 + z_1^{-1} z_2^{-1} + z_1 z_2^{-1}),$$

happens to be affine invariant: $s(1,1) = 4$ and $s(1,-1) = s(-1,1) = s(-1,-1) = 0$.

Similar negative remarks hold for the factorization (4.19)$_{/157}$ of the subdivision polynomial for the $\{\sqrt{3}\}^2$ method. The method defined by three times the second

factor on the right in $(4.19)_{/157}$, with subdivision polynomial $s(z_1, z_2)$ defined by

$$3s_P(z_1^3, z_2^3) = 2 + (z_1 z_2 + z_1^{-1} z_2^2 + z_1^{-2} z_2 + z_1^{-1} z_2^{-1} + z_1 z_2^{-2} + z_1^2 z_2^{-1}) + \frac{1}{6}(z_1^3 + z_2^3 + z_1^{-3} z_2^3 + z_1^{-3} + z_2^{-3} + z_1^3 z_2^3),$$

where s_P is defined by $(4.20)_{/157}$, is not affine invariant. Thus, although $s(1,1) = 9$ as required, none of the equations $\sum_{l \in \mathbb{Z}^2} s_{k-3l} = 1$ (mentioned in the solution to Exercise $9_{/187}$ in Chapter 4) is satisfied, $k \in \{(k_1, k_2) : 0 \le k_1, k_2 \le 2\}$. Furthermore, the method defined by three times the first factor on the right in $(4.19)_{/157}$ fails to be affine invariant; see Exercise $2_{/245}$ at the end of this chapter.

5.2 Convergence of box-spline subdivision processes

In this section, we give two general theorems on linear and quadratic convergence for box-spline nodal functions. As in Section 5.1, here we are concerned only with the regular case (Figure $1.30_{/33}$, lower row, first and second columns).

Assume that we are given a box-spline subdivision polynomial

$$s^*(z) = 4 \prod_{i=1}^{m} \Big(\frac{1 + z^{e_i}}{2} \Big) = \sum_{k \in G_m^*} s_k^* z^k \tag{5.20}$$

as in $(3.19)_{/114}$. We consider the case that the parameter value α in Theorem $3.3.2_{/111}$ is such that $m - 1 - \alpha \ge 0$, i.e., the box-spline nodal function $N^*(he^m; y)$ is at least continuous. Since we must have $\alpha \ge 2$, this means that m is at least 3. That $N^*(he^m; y)$ is piecewise polynomial and continuous implies that all first derivatives are continuous over all subdomains of polynomiality and that they have only jump discontinuities. It also means, since $m - 2 \ge \alpha - 1$ (the largest number of parallel vectors in e^m), that the subset $\{e_1, e_2, \ldots, e_m\} \setminus \{e_k\}$ obtained by omitting one of the vectors in e^m spans $\mathbb{R}^2$ and defines an $(m-1)$-order box-spline nodal function which is piecewise continuous with at most jump discontinuities.

Before proceeding we introduce some notation. The notation

$$\Delta_e \zeta_k^\nu \doteq \zeta_k^\nu - \zeta_{k-e}^\nu$$

denotes a first backward difference, while the notations

$$\Delta_{e_i} \Delta_{e_j} \zeta_k^\nu = \zeta_k^\nu - \zeta_{k-e_j}^\nu - \zeta_{k-e_i}^\nu + \zeta_{k-e_j-e_i}^\nu$$

and

$$\Delta_{e_i}^2 \zeta_k^\nu = \zeta_k^\nu - 2\zeta_{k-e_i}^\nu + \zeta_{k-2e_i}^\nu$$

denote second differences. Also let $e_{(i)}^m$ denote the sequence of vectors obtained from $e^m = \{e_1, e_2, \ldots, e_m\}$ by omitting e_i, and let $e_{(ij)}^m$ denote that obtained by omitting both e_i and e_j. Then, $N^*(he_{(i)}^m; y)$ and $N^*(he_{(ij)}^m; y)$ are the corresponding nodal functions, and the refined subdivision polynomials for $N^*(he^m; y)$, $N^*(he_{(i)}^m; y)$, and

$N^*(he^m_{(ij)};y)$ are denoted by, respectively,

$$\zeta^\nu(z^{1/2^{\nu-1}}) = \sum_{k\in\mathbb{Z}^2} \zeta^\nu_k z^{k/2^{\nu-1}},$$

$$p^\nu(z^{1/2^{\nu-1}}) = \sum_{k\in\mathbb{Z}^2} p^\nu_k z^{k/2^{\nu-1}},$$

and

$$\kappa^\nu(z^{1/2^{\nu-1}}) = \sum_{k\in\mathbb{Z}^2} \kappa^\nu_k z^{k/2^{\nu-1}}.$$

The polynomials p^ν and let κ^ν depend on $e^m_{(i)}$ and $e^m_{(ij)}$, respectively, but this is suppressed in the notation. Also, these polynomials correspond to the uncentered subdivision polynomial s^*, but to reduce the notational complexity, we have not added the symbol $*$ to the coefficients ζ^ν_k, p^ν_k, and κ^ν_k. This creates no problem, except in the case of ζ^ν_k: in Theorem 5.2.7$_{/205}$, we need to introduce the corresponding constants for the centered box splines, and these constants would normally be denoted ζ^ν_k, so that a mental change of notation is necessary immediately preceding Theorem 5.2.7$_{/205}$.

5.2.1 Linear convergence

We first formulate a lemma.

Lemma 5.2.1. *If $N^*(he^m;y)$ is continuous, then*

$$|\Delta_{e_j}\zeta^\nu_k| = |\zeta^\nu_k - \zeta^\nu_{k-e_j}| \le h/2^\nu$$

for all $k\in\mathbb{Z}^2$ and all j, $j=1,2,\dots,m$.

Proof. It suffices to carry out the proof for $j=m$. Using Theorem 3.2.3$_{/99}$ we have

$$D_{e_m}N^*(he^m;y) = \frac{1-z^{2e_m}}{h}N^*(he^{m-1};y) \tag{5.21}$$

with z^{2e_m} denoting translation by the vector he_m. Now, let us introduce the notation

$$p^\nu(z^{1/2^{\nu-1}}) = \prod_{i=0}^{\nu-1} s^*_{m-1}(z^{1/2^i}) = \sum_{\mu\in\mathbb{Z}^2} p^\nu_\mu z^{\mu/2^{\nu-1}}$$

for the refined subdivision polynomial corresponding to the box-spline subdivision polynomial

$$s^*_{m-1}(z) = 4\prod_{i=1}^{m-1}\Big(\frac{1+z^{e_i}}{2}\Big) = \sum_{k\in G_{m-1}} s^*_k z^k.$$

Then we have

$$N^*(he^m;y) = \zeta^\nu(z^{1/2^{\nu-1}})N^*(he^m/2^\nu;y)$$
$$= \left(\sum_{k\in\mathbb{Z}^2} \zeta_k^\nu z^{k/2^{\nu-1}}\right) N^*(he^m/2^\nu;y) \tag{5.22}$$

and

$$N^*(he^{m-1};y) = p^\nu(z^{1/2^{\nu-1}})N^*(he^{m-1}/2^\nu;y)$$
$$= \left(\sum_{k\in\mathbb{Z}^2} p_k^\nu z^{k/2^{\nu-1}}\right) N^*(he^{m-1}/2^\nu;y). \tag{5.23}$$

Now, using $(5.22)_{/199}$ and the fact that differentiation commutes with translation, we have

$$D_{e_m}N^*(he^m;y) = \zeta^\nu(z^{1/2^{\nu-1}})D_{e_m}N^*(he^m/2^\nu;y). \tag{5.24}$$

Inserting $(5.21)_{/198}$ with he^m replaced by $he^m/2^\nu$, this becomes

$$D_{e_m}N^*(he^m;y) = \zeta^\nu(z^{1/2^{\nu-1}})\, 2^\nu \frac{1-z^{e_m/2^{\nu-1}}}{h} N^*(he^{m-1}/2^\nu;y).$$

Again, using $(5.21)_{/198}$ and $(5.23)_{/199}$ we have

$$\frac{1-z^{2e_m}}{h} p^\nu(z^{1/2^{\nu-1}})N^*(he^{m-1}/2^\nu;y)$$
$$= 2^\nu \frac{1-z^{e_m/2^{\nu-1}}}{h} \zeta^\nu(z^{1/2^{\nu-1}})N^*(he^{m-1}/2^\nu;y), \tag{5.25}$$

i.e.,

$$2^\nu \frac{1-z^{e_m/2^{\nu-1}}}{h} \zeta^\nu(z^{1/2^{\nu-1}}) = \frac{1-z^{2e_m}}{h} p^\nu(z^{1/2^{\nu-1}}).$$

For the coefficients of these polynomials we then get

$$2^\nu \frac{1-z^{e_m/2^{\nu-1}}}{h} \left(\sum_{\mu\in\mathbb{Z}^2} \zeta_\mu^\nu z^{\mu/2^{\nu-1}}\right) = \frac{1-z^{2e_m}}{h} \left(\sum_{k\in\mathbb{Z}^2} p_k^\nu z^{k/2^{\nu-1}}\right) \tag{5.26}$$

and, identifying powers of z,

$$\zeta_k^\nu - \zeta_{k-e_m}^\nu = \frac{h}{2^\nu}(p_k^\nu - p_{k-2^\nu e_m}^\nu) \tag{5.27}$$

for all $k \in \mathbb{Z}^2$. Now by $(5.7)_{/192}$,

$$p_k^\nu = \sum_{l\in\mathbb{Z}^2} s_{k-2l}^* p_l^{\nu-1}$$

and similarly

$$\begin{aligned} p^{\nu}_{k-2^{\nu}e_m} &= \sum_{l\in\mathbb{Z}^2} s^{*}_{k-2(l+2^{\nu-1}e_m)} p^{\nu-1}_{l} \\ &= \sum_{l\in\mathbb{Z}^2} s^{*}_{k-2l} p^{\nu-1}_{l-2^{\nu-1}e_m}. \end{aligned}$$

This gives

$$p^{\nu}_{k} - p^{\nu}_{k-2^{\nu}e_m} = \sum_{l\in\mathbb{Z}^2} s^{*}_{k-2l}(p^{\nu-1}_{l} - p^{\nu-1}_{l-2^{\nu-1}e_m}).$$

Using that $s^{*}_{k-2l} \geq 0$ for all $k, l \in \mathbb{Z}^2$ and that $\sum_{l\in\mathbb{Z}^2} s^{*}_{k-2l} = 1$, we conclude that

$$\max_{k\in\mathbb{Z}^2} |p^{\nu}_{k} - p^{\nu}_{k-2^{\nu}e_m}| \leq \max_{l\in\mathbb{Z}^2} |p^{\nu-1}_{l} - p^{\nu-1}_{l-2^{\nu-1}e_m}|.$$

By recursion we have

$$\max_{k\in\mathbb{Z}^2} |p^{\nu}_{k} - p^{\nu}_{k-2^{\nu}e_m}| \leq \max_{k\in\mathbb{Z}^2} |p^{0}_{k} - p^{0}_{k-e_m}| = 1$$

and by $(5.27)_{/199}$ we have

$$\max_{k\in\mathbb{Z}^2} |\zeta^{\nu}_{k} - \zeta^{\nu}_{k-e_m}| \leq h/2^{\nu},$$

which completes the proof of the lemma. □

By Remark $3.5.5_{/118}$ the vectors $e_1, e_2, \ldots, e_m$ generate $\mathbb{Z}^2$. This implies that given any $k \in \mathbb{Z}^2$ there exist $k_i \in \mathbb{Z}$, $1 \leq i \leq m$, such that

$$k = \sum_{i=1}^{m} k_i e_i$$

or equivalently,

$$k = \sum_{j=1}^{M} u_j,$$

where $\pm u_j \in e^m$ and M is the number of elements in the sum.

Now, let us introduce constants C_k, C, and $\bar{C}_i$ defined by the set e^m in the following way.

Definition 5.2.2.

$$C_k = \min\left\{ \sum_{i=1}^{m} |k_i| : k = \sum_{i=1}^{m} k_i e_i, k_i \in \mathbb{Z} \right\}.$$

$$C = \max\left\{ C_k : k \in \mathbb{Z}^2 \cap (\mathrm{conv}(G^{*}_{m}))^{0} \right\}.$$

$$\bar{C}_i = \min\left\{ \sum_{j} |k_j| : e_i = \sum_{1\leq j\leq m, j\neq i} k_j e_j \right\}.$$

We then have the following lemma.

Lemma 5.2.3. *For all $l \in \mathbb{Z}^2$ and all $k \in \mathbb{Z}^2$ we have*

$$|\Delta_k \zeta_l^\nu| = |\zeta_l^\nu - \zeta_{l-k}^\nu| \leq C_k h/2^\nu. \tag{5.28}$$

Proof. We first note that

$$\Delta_k \zeta_l^\nu = \sum_{j=1}^{M} \Delta_{u_j} \zeta_{l-l_j}^\nu, \tag{5.29}$$

where $l_1 = 0$ and

$$l_j = \sum_{1 \leq r < j} u_r \quad \text{for} \quad 1 < j. \tag{5.30}$$

We conclude by the triangle inequality and Lemma 5.2.1$_{/198}$ that

$$|\Delta_k \zeta_l^\nu| \leq Mh/2^\nu,$$

i.e., by Definition 5.2.2$_{/200}$ that

$$|\Delta_k \zeta_l^\nu| \leq C_k h/2^\nu,$$

and the proof is complete. □

Theorem 5.2.4. *If the nodal function $N^*(he^m; y)$ is continuous, then the subdivision process, defined by $s^*(z)$ in (5.20)$_{/197}$ and with initial control points $p_l = 1$ if $l = 0$ and $p_l = 0$ otherwise, converges uniformly towards $N^*(he^m; y)$. Moreover, the convergence is linear in the sense that*

$$|\zeta_k^\nu - N^*(he^m; kh/2^\nu)| \leq Ch/2^\nu \tag{5.31}$$

as $\nu \to \infty$, where C is the constant in Definition 5.2.2$_{/200}$.

Proof. Since

$$N^*(he^m; y) = \sum_{k \in \mathbb{Z}^2} \zeta_k^\nu N^*(he^m/2^\nu; y - hk/2^\nu)$$

and

$$\sum_{k \in \mathbb{Z}^2} N^*(he^m/2^\nu; y - hk/2^\nu) = 1,$$

we get

$$\begin{aligned} \zeta_l^\nu - N^*(he^m; lh/2^\nu) &= \sum_{k \in \mathbb{Z}^2} (\zeta_l^\nu - \zeta_k^\nu) N^*(he^m/2^\nu; (l-k)h/2^\nu) \\ &= \sum_{k \in \mathbb{Z}^2} (\zeta_l^\nu - \zeta_k^\nu) N^*(he^m; l-k) = \sum_{k \in \mathbb{Z}^2} (\zeta_l^\nu - \zeta_{l-k}^\nu) N^*(he^m; k). \end{aligned}$$

In the last term the summation over k can be restricted to $(\mathrm{conv}\,(G_m^*))^0$, and we conclude by Lemma 5.2.3$_{/201}$ that

$$|\zeta_l^\nu - N^*(he^m; lh/2^\nu)| \le Ch2^{-\nu} \sum_{k\in\mathbb{Z}^2} N^*(he^m; k) = Ch2^{-\nu},$$

which completes the proof. □

5.2.2 Quadratic convergence

To establish quadratic convergence we need a lemma analogous to Lemma 5.2.1$_{/198}$.

Lemma 5.2.5. *If $N^*(he^m; y)$ is continuously differentiable, then*

$$\left|\frac{1}{2}\Delta_{e_i}\Delta_{e_j}\zeta_k^\nu\right| \le h^2/4^\nu \tag{5.32}$$

for all $k \in \mathbb{Z}^2$ and all $i, j \in \{1, 2, \ldots, m\}$ with $i \ne j$. If, in addition, e_i may be written as

$$e_i = \sum_{j:j\ne i} k_j e_j, \tag{5.33}$$

then

$$\left|\frac{1}{2}\Delta_{e_i}^2\zeta_k^\nu\right| \le \bar{C}_i h^2/4^\nu, \tag{5.34}$$

where $\bar{C}_i$ is as in Definition 5.2.2$_{/200}$.

If e_i appears at least twice in the sequence e^m, then by (5.32)$_{/202}$ the inequality (5.34)$_{/202}$ is valid with $\bar{C}_i = 1$.

Proof. $N^*(he^m; y)$ is in $C^1(\mathbb{R}^2)$, and therefore $m - 1 - \alpha \ge 1$ where α is the parameter value in Theorem 3.3.2$_{/111}$. Since $\alpha \ge 2$ we have $m \ge 4$. That $N^*(he^m; y)$ is piecewise polynomial and in $C^1(\mathbb{R}^2)$ implies that all second derivatives are continuous over all subdomains of polynomiality and that they have at most jump discontinuities. The subset $\{e_1, e_2, \ldots, e_m\} \setminus \{e_i, e_j\}$ obtained by omitting any two vectors e_i and e_j spans $\mathbb{R}^2$ and defines an $(m-2)$-order box-spline nodal function with at most jump discontinuities.

If $e_i \ne e_j$, we conclude by Theorem 3.2.3$_{/99}$ that

$$D_{e_i}D_{e_j}N^*(he^m; y) = \frac{1 - z^{2e_i}}{h}\frac{1 - z^{2e_j}}{h}N^*(he^m_{(ij)}; y).$$

We obtain then, similarly to Lemma 5.2.1$_{/198}$,

$$\begin{aligned} &4^\nu \frac{1 - z^{e_i/2^{\nu-1}}}{h}\frac{1 - z^{e_j/2^{\nu-1}}}{h}\left(\sum_{\mu\in\mathbb{Z}^2}\zeta_\mu^\nu z^{\mu/2^{\nu-1}}\right) \\ &= \frac{1 - z^{2e_i}}{h}\frac{1 - z^{2e_j}}{h}\left(\sum_{k\in\mathbb{Z}^2}\kappa_k^\nu z^{k/2^{\nu-1}}\right). \end{aligned} \tag{5.35}$$

For the coefficients this gives, after identifying powers of z,

$$\Delta_{e_i}\Delta_{e_j}\zeta_k^\nu = \frac{h^2}{4^\nu}\Delta_{2^\nu e_i}\Delta_{2^\nu e_j}\kappa_k^\nu.$$

In the case that e_i appears at least twice in the sequence e^m, we have in the same way

$$\Delta_{e_i}^2\zeta_k^\nu = \frac{h^2}{4^\nu}\Delta_{2^\nu e_i}^2\kappa_k^\nu.$$

Similarly to the proof of Lemma 5.2.1$_{/198}$, we now get that

$$\max_{k\in\mathbb{Z}^2}|\Delta_{2^\nu e_i}\Delta_{2^\nu e_j}\kappa_k^\nu| \le \max_{k\in\mathbb{Z}^2}|\Delta_{2^{\nu-1} e_i}\Delta_{2^{\nu-1} e_j}\kappa_k^{\nu-1}| \le \cdots \le \max_{k\in\mathbb{Z}^2}|\Delta_{e_i}\Delta_{e_j}\kappa_k^0| \le 2$$

and we conclude that

$$\left|\frac{1}{2}\Delta_{e_i}\Delta_{e_j}\zeta_k^\nu\right| \le h^2/4^\nu \tag{5.36}$$

and, if e_i appears at least twice in the sequence e^m, that

$$\left|\frac{1}{2}\Delta_{e_i}^2\zeta_k^\nu\right| \le h^2/4^\nu.$$

In this case, (5.34)$_{/202}$ holds with $\bar{C}_i = 1$.

For the case that e_i appears only once in the sequence e^m we argue as follows. Equation (5.33)$_{/202}$ can be rewritten as

$$e_i = \sum_{j=1}^{M} u_j,$$

where $\pm u_j \in e_{(i)}^m$ and M is the number of terms in the sum. We then have

$$\Delta_{e_i}\zeta_k^\nu = \sum_{j=1}^{M}\Delta_{u_j}\zeta_{l-l_j}^\nu, \tag{5.37}$$

where $l_1 = 0$ and

$$l_j = \sum_{1\le r<j} u_r \quad \text{for} \quad 1 < j.$$

Operating with Δ_{e_i} in (5.37)$_{/203}$ we get

$$\Delta_{e_i}^2\zeta_k^\nu = \sum_{j=1}^{M}\Delta_{e_i}\Delta_{u_j}\zeta_{l-l_i-l_j}^\nu,$$

and since $\Delta_{u_j} = \Delta_{\pm e_j}$, we conclude from (5.36)$_{/203}$ that

$$\left|\frac{1}{2}\Delta_{e_i}^2\zeta_k^\nu\right| \le \bar{C}_i h^2/4^\nu$$

with $\bar{C}_i = M$, which completes the proof. □

Lemma 5.2.6. *If $N^*(he^m; y)$ is continuously differentiable and every $e_i \in e^m$ may be written as an integer combination*

$$e_i = \sum_{1 \le j \le m, j \ne i} k_j e_j, \quad k_j \in \mathbb{Z},$$

then for every $k \in \mathbb{Z}^2$ we have

$$\frac{1}{2}|\Delta_k^2 \zeta_l^\nu| \le C^* h^2 / 4^\nu, \tag{5.38}$$

where

$$C^* = C_k(C_k - 1) + C_k \max_i \bar{C}_i \, .$$

If in particular $k \in \mathbb{Z}^2 \cap (\mathrm{conv}(G_m^*))^0$, then $C^* = C(C-1) + C \max_i \bar{C}_i$.

Proof. Let $k = \sum_{j=1}^M u_j$ with $\pm u_j \in e^m$ and $C_k = M$. Then

$$\Delta_k \zeta_l^\nu = \sum_{1 \le j \le M} \Delta_{u_j} \zeta_{l - l_j}^\nu$$

with $l_1 = 0$ and $l_j = \sum_{1 \le r < j} u_r$ for $j > 1$. Operating with Δ_k on both sides we get

$$\Delta_k^2 \zeta_l^\nu = \sum_{1 \le i,j \le M} \Delta_{u_i} \Delta_{u_j} \zeta_{l - l_i - l_j}^\nu$$

and by Lemma 5.2.5$_{/202}$ and the triangle inequality, we conclude that

$$\frac{1}{2}|\Delta_k^2 \zeta_l^\nu| \le \frac{1}{2} \sum_{i \ne j} |\Delta_{u_i} \Delta_{u_j} \zeta_{l - l_i - l_j}^\nu| + \frac{1}{2} \sum_i |\Delta_{u_i}^2 \zeta_{l - 2l_i}^\nu| \le \Big(M(M-1) + \sum_{i=1}^M \bar{C}_i \Big) h^2 / 4^\nu$$

from which (5.38)$_{/204}$ follows. □

Before formulating a result on quadratic convergence we recall the definition of centered box splines:

$$N(he^m; y) = N^*(he^m; y + h\bar{e}/2),$$

where $\bar{e} = \sum_{i=1}^m e_i$. The centered box splines (see Section 3.5.2) have the symmetry property

$$N(he^m; y) = N(he^m; -y).$$

We now make a change of notation, letting ζ_k^ν denote the corresponding coefficients:

$$\zeta_k^\nu := \zeta_{k + (2^\nu - 1)\bar{e}/2}^\nu.$$

We then have the following theorem on quadratic convergence.

Theorem 5.2.7. *Assume that $N(he^m; y)$ is continuously differentiable and that every $e_i \in e^m$ may be written as an integer combination*

$$e_i = \sum_{j \neq i} k_j e_j.$$

Then

$$|\zeta_l^\nu - N(he^m; lh/2^\nu)| \leq C^* h^2/4^\nu,$$

where $C^* = C(C-1) + C \max_i \bar{C}_i$.

Proof. Similarly to the proof of Theorem 5.2.4$_{/201}$, we get

$$\zeta_l^\nu - N(he^m; lh/2^\nu) = \sum_{k \in \mathbb{Z}^2} (\zeta_l^\nu - \zeta_k^\nu) N(he^m/2^\nu; (l-k)h/2^\nu)$$

$$= \sum_{k \in \mathbb{Z}^2} (\zeta_l^\nu - \zeta_k^\nu) N(he^m; l-k) = \sum_{k \in \mathbb{Z}^2} (\zeta_l^\nu - \zeta_{l-k}^\nu) N(he^m; k),$$

where the last summation can be taken over $\mathbb{Z}^2 \cap (\mathrm{conv}(G_m^*))^0$. Changing summation index k to $-k$ in the last sum and using that $N(he^m; -k) = N(he^m; k)$, we get

$$\zeta_l^\nu - N(he^m; lh/2^\nu) = \sum_{k \in \mathbb{Z}^2} (\zeta_l^\nu - \zeta_{l+k}^\nu) N(he^m; k),$$

and taking sums,

$$\zeta_l^\nu - N(he^m; lh/2^\nu) = \frac{1}{2} \sum_{k \in \mathbb{Z}^2} (2\zeta_l^\nu - \zeta_{l+k}^\nu - \zeta_{l-k}^\nu) N(he^m; k), \tag{5.39}$$

i.e.,

$$\zeta_l^\nu - N(he^m; lh/2^\nu) = -\frac{1}{2} \sum_{k \in \mathbb{Z}^2} \Delta_k^2 \zeta_{l+k}^\nu N(he^m; k).$$

In the last term the summation over k can again be restricted to $(\mathrm{conv}\,(G_m))^0$, and we conclude by Lemma 5.2.6$_{/204}$ and the triangle inequality that

$$|\zeta_l^\nu - N(he^m; lh/2^\nu)| \leq C^* h^2/4^\nu \sum_k N(he^m; k) = C^* h^2/4^\nu,$$

and the proof is complete. □

Example 5.2.8. The three-direction quartic box spline.

Taking $e^6 = \{(1,0)^t, (-1,0)^t, (0,1)^t, (0,-1)^t, (1,1)^t, (-1,-1)^t\}$, we get the set G_6^* as in Figure 3.19$_{/133}$, and the following for the constants in Definition 5.2.2$_{/200}$:

$$C_k = |k_1| + |k_2| \text{ for any } k \in \mathbb{Z}^2,$$

$$C = \max\{C_k : k \in \mathbb{Z}^2 \cap (\mathrm{conv}(G_6^*))^0\} = 2,$$

$$\bar{C}_i = 2 \text{ for all vectors } e_i \in e^6.$$

Consequently, $C^* = C(C-1) + C \max_i \bar{C}_i = 2 + 2 \cdot 2 = 6$. ∎

The next proposition shows that for most subdivision methods defined by subdivision polynomials, quadratic convergence is the best one can expect. This proposition applies beyond the box-spline case.

Proposition 5.2.9. *Assume that we are given a subdivision polynomial defining a convergent subdivision procedure and a nodal function $N(h;y)$ having the following properties:*

(i) $N(h;y) = N(h;-y)$.

(ii) $N(h;y) \geq 0$ *everywhere.*

(iii) *There is at least one point $y_0 \in \operatorname{supp}(N(h;\cdot))$ such that $N(h;y_0) > 0$, $N(h;\cdot)$ is twice continuously differentiable in a neighbourhood of y_0, and the Hessian H of N has the property that $H(y_0)$ is either positive or negative definite.*

Then an inequality of the form

$$|p_l^\nu - N(h;lh/2^\nu)| \leq \epsilon_\nu h^2/4^\nu,$$

valid for all l and with $\epsilon_\nu \to 0$ as $\nu \to \infty$, is impossible, unless $p_l^\nu = N(h;hl/2^\nu)$ for all l and ν (interpolating subdivision). Here we have used the notation

$$p^\nu(z^{1/2^{\nu-1}}) = \sum_{k\in\mathbb{Z}^2} p_k^\nu z^{k/2^{\nu-1}}$$

for the refined subdivision polynomials for $N(h;y)$, so that $|p_l^\nu - N(h;hl/2^\nu)| \to 0$ as $\nu \to \infty$.

Proof. Arguing as in the proof of Theorem 5.2.7$_{/205}$ we get, similarly to (5.39)$_{/205}$,

$$p_l^\nu - N(h;lh/2^\nu) = \frac{1}{2}\sum_{k\in\mathbb{Z}^2}(2p_l^\nu - p_{l+k}^\nu - p_{l-k}^\nu)N(1;k), \tag{5.40}$$

and indices other than $k = 0$ enter into this sum, since the subdivision is assumed not to be interpolating. Inserting

$$p_l^\nu = N(h;lh/2^\nu) + \delta_{\nu,l} \tag{5.41}$$

in (5.40)$_{/206}$, where $|\delta_{\nu,l}| \leq \epsilon_\nu h^2/4^\nu$, we have

$$\begin{aligned}\delta_{\nu,l} = {} & \frac{1}{2}\sum_{k\in\mathbb{Z}^2}\big(2N(h;lh/2^\nu) - N(h;(l+k)h/2^\nu) - N(h;(l-k)h/2^\nu)\big)N(1;k)\\ & + \frac{1}{2}\sum_{k\in\mathbb{Z}^2}(2\delta_{\nu,l} - \delta_{\nu,l+k} - \delta_{\nu,l-k})N(1;k).\end{aligned}$$

Now, by Taylor's theorem we get

$$\begin{aligned}& 2N(h;lh/2^\nu) - N(h;(l+k)h/2^\nu) - N(h;(l-k)h/2^\nu)\\ & = -\frac{h^2}{4^\nu}k^t H(\xi)k,\end{aligned}$$

where H denotes the Hessian of $N(h;y)$ and $\xi = \xi(l,k,\nu)$ is some point on the segment $\{y \in \mathbb{R}^2 : y = (l+tk)h/2^\nu, -1 < t < 1\}$. Consequently,

$$\begin{aligned} &-\frac{h^2}{4^\nu}\frac{1}{2}\sum_{k\in\mathbb{Z}^2} k^t H(\xi(l,k,\nu))k\, N(1;k) \\ &= \delta_{\nu,l} - \frac{1}{2}\sum_{k\in\mathbb{Z}^2}(2\delta_{\nu,l} - \delta_{\nu,l+k} - \delta_{\nu,l-k})N(1;k). \end{aligned} \tag{5.42}$$

Now, since in (5.42)$_{/207}$, k is in the bounded set $\operatorname{supp}(N(1,\cdot))$, and since $H(y_0)$ is definite, we may choose sequences ν, l in such a way that $\xi(l,k,\nu) \to y_0$ and so that $|k^t H(\xi(l,k,\nu))k| \geq c$ for some constant $c > 0$ if $k \neq 0$. Using that $|\delta_{\nu,l}| \leq \epsilon_\nu \frac{h^2}{4^\nu}$, $N(1;k) \geq 0$, $\sum_{k\in\mathbb{Z}^2} N(1;k) = 1$, and the triangle inequality, we get

$$c\frac{h^2}{4^\nu} \leq \epsilon_\nu \frac{h^2}{4^\nu} + \frac{1}{2}4\epsilon_\nu \frac{h^2}{4^\nu} = 3\epsilon_\nu \frac{h^2}{4^\nu},$$

i.e.,

$$0 < c \leq 3\epsilon_\nu \to 0,$$

which is a contradiction. □

It was mentioned, immediately before the statement of Theorem 5.2.7$_{/205}$, that the hypothesis (i) of Proposition 5.2.9$_{/206}$ is satisfied for any centered box spline. In fact, the Nodal-Function Computation principle mentioned in Section 4.1 ensures more generally that if the subdivision mask is symmetric in the origin, then the nodal function is symmetric.

A result corresponding to Theorem 5.2.9$_{/206}$ is valid in the univariate case. In this case the Hessian of $N(h;y)$ is replaced by the second derivative $N''(t)$ which is assumed to be continuous and different from zero on some interval.

Corollary 5.2.10. *For all univariate nodal spline functions $N^m(h;t)$ with $m \geq 3$ we have quadratic convergence, but not better.*

Remark 5.2.11. For all interpolating subdivision methods with continuous nodal functions, the convergence is perfect in the sense that $p_l^\nu - N(h;lh/2^\nu) = 0$ for all l and ν. ■

5.3 Convergence and smoothness for general subdivision polynomials

It was possible, in Sections 5.2 and 3.3, respectively, to give a very complete analysis of the convergence and regularity properties of box-spline subdivision methods. In contrast, a full analysis for General-subdivision-polynomial methods, which correspond to the lower row of column 3 of Figure 1.30$_{/33}$, is much more difficult. In this section we present some useful sufficient conditions for convergence and smoothness using elementary methods, and the results here probably represent the most we can

do by using such methods. In Section 5.9 we give references to the literature, where stronger results have been obtained using more elaborate analyses.

5.3.1 Convergence analysis

As always, it is sufficient to carry out the convergence analysis for an initial sequence $\{p_l^0\}_{l\in\mathbb{Z}^2}$ with $p_0^0 = 1$ and $p_l^0 = 0$ for $l \neq 0$, where the limiting function, if it exists, is the nodal function N.

In the ν th step we have a refined sequence p_l^ν, $l \in \mathbb{Z}^2$, defining the polynomials $p^\nu(z) = \sum_l p_l^\nu z^l$ recursively by

$$p^{\nu+1}(z) = s(z)p^\nu(z^2),$$

(see $(5.5)_{/191}$) so that

$$p_k^{\nu+1} = \sum_{l\in\mathbb{Z}^2} s_{k-2l} p_l^\nu$$

and

$$p^{\nu+1}(z^{1/2^\nu}) = s(z^{1/2^\nu})s(z^{1/2^{\nu-1}})\cdots s(z).$$

Now, we introduce the piecewise bilinear and continuous function P^ν defined by

$$P^\nu(y) = \sum_{l\in\mathbb{Z}^2} p^\nu_{l_1,l_2} N^2(h; u - l_1 h)N^2(h; v - l_2 h), \tag{5.43}$$

where $N^2(h;\cdot)$ is the first-degree univariate spline function with support in $[-h, h]$ and with $N^2(h; 0) = 1$. Thus, P^ν is the piecewise bilinear function which interpolates the values p_l^ν in the points $y = hl/2^\nu$. For the univariate case,

$$P^\nu(t) = \sum_{l\in\mathbb{Z}} p_l^\nu N^2(h; t - lh). \tag{5.44}$$

In order to simplify notation we choose $h = 1$ in the following analysis. We also use that $N^2(h2^{-\nu}; t - lh2^{-\nu}) = N^2(1; 2^\nu t - l) := N^2(2^\nu t - l)$.

We have the following lemma.

Lemma 5.3.1. *Assume that for some constants $\gamma < 1$ and c_1 we have*

$$|p_l^\nu - p^\nu_{l-e}| < c_1\gamma^\nu \tag{5.45}$$

for all ν, where in the bivariate case $e = (1, 0)^t$ and $e = (0, 1)^t$, and in the univariate case $e = 1$. Then the subdivision process converges uniformly towards a continuous limit function $N(y)$ and

$$|N(y) - P^\nu(y)| \le cc_1\gamma^\nu/(1 - \gamma) \tag{5.46}$$

for some constant c depending only on the polynomial $s(z)$.

More generally, in the bivariate case the following is true. Let e_1 and $e_2 \in \mathbb{Z}^2$ be given and assume that they generate $\mathbb{Z}^2$, i.e., that every $k \in \mathbb{Z}^2$ can be written as $k = n_1 e_1 + n_2 e_2$ for some integers n_1 and n_2. Also assume that the inequality $(5.45)_{/208}$ is valid for e_1 and e_2. Then we obtain the same conclusion.

Proof. For simplicity we carry out the proof in the univariate case only. The generalization to the bivariate case is straightforward. We have

$$\begin{aligned} P^{\nu+1}(t) &= \sum_{l\in\mathbb{Z}} p_l^{\nu+1} N^2(2^{\nu+1}t - l) \\ &= \sum_{l\in\mathbb{Z}} p_{2l}^{\nu+1} N^2(2^{\nu+1}t - 2l) + \sum_{l\in\mathbb{Z}} p_{2l-1}^{\nu+1} N^2(2^{\nu+1}t - 2l + 1). \end{aligned} \tag{5.47}$$

Further, using $(2.25)_{/62}$, with $s(z) = \left(\frac{z^{1/2}+z^{-1/2}}{2}\right)^2 2 = z/2 + 1 + z^{-1}/2$, we have the 2-scale relation

$$N^2(t) = N^2(2t+1)/2 + N^2(2t) + N^2(2t-1)/2$$

and therefore,

$$N^2(2^\nu t - l) = N^2(2^{\nu+1}t - 2l + 1)/2 + N^2(2^{\nu+1}t - 2l) + N^2(2^{\nu+1}t - 2l - 1)/2.$$

Inserting this into $(5.44)_{/208}$ we get

$$\begin{aligned} P^\nu(t) &= \sum_{l\in\mathbb{Z}} p_l^\nu \big(N^2(2^{\nu+1}t - 2l + 1)/2 + N^2(2^{\nu+1}t - 2l) + N^2(2^{\nu+1}t - 2l - 1)/2\big) \\ &= \sum_{l\in\mathbb{Z}} p_l^\nu N^2(2^{\nu+1}t - 2l) + \sum_{l\in\mathbb{Z}} ((p_l^\nu + p_{l-1}^\nu)/2) N^2(2^{\nu+1}t - 2l + 1). \end{aligned}$$

Consequently, using $(5.47)_{/209}$,

$$\begin{aligned} P^{\nu+1}(t) - P^\nu(t) = &\sum_{l\in\mathbb{Z}} (p_{2l}^{\nu+1} - p_l^\nu) N^2(2^{\nu+1}t - 2l) \\ &+ \sum_{l\in\mathbb{Z}} (p_{2l-1}^{\nu+1} - (p_l^\nu + p_{l-1}^\nu)/2) N^2(2^{\nu+1}t - 2l + 1). \end{aligned} \tag{5.48}$$

Now, since $\sum_{l\in\mathbb{Z}} s_{k-2l} = 1$ for all k, we have

$$p_{2k}^{\nu+1} - p_k^\nu = \sum_{l\in\mathbb{Z}} s_{2k-2l}(p_l^\nu - p_k^\nu) \tag{5.49}$$

and

$$p_{2k-1}^{\nu+1} - (p_k^\nu + p_{k-1}^\nu)/2 = \sum_{l\in\mathbb{Z}} s_{2k-1-2l}(p_l^\nu - (p_k^\nu + p_{k-1}^\nu)/2). \tag{5.50}$$

The values $k-l$ in $(5.49)_{/209}$ and $(5.50)_{/209}$ range over a bounded set, defined by the condition that $2k-2l$ and $2k-1-2l$ should be coefficients of $s(z)$. Therefore,

$$|p_l^\nu - p_k^\nu| \le \sum_{i\in\mathbb{Z}} |p_{k+i+1}^\nu - p_{k+i}^\nu| \le cc_1\gamma^\nu$$

and

$$\begin{aligned} |p_l^\nu - (p_k^\nu + p_{k-1}^\nu)/2| &= |(p_l^\nu - p_k^\nu)/2 + (p_l^\nu - p_{k-1}^\nu)/2| \\ &\le \sum_{i\in\mathbb{Z}} |p_{k+i+1}^\nu - p_{k+i}^\nu| \le cc_1\gamma^\nu \end{aligned}$$

for some constant c depending only on the support of s. Using the triangle inequality in $(5.48)_{/209}$, it then follows that

$$|P^{\nu+1}(t) - P^\nu(t)| \le cc_1\gamma^\nu.$$

Consequently,

$$\sum_{\nu=0}^{\infty} |P^{\nu+1}(t) - P^\nu(t)| < cc_1 \sum_{\nu=0}^{\infty} \gamma^\nu = cc_1/(1-\gamma)$$

and we conclude that the partial sums

$$P^\nu(t) = \sum_{k=0}^{\nu-1} (P^{k+1}(t) - P^k(t))$$

converge uniformly on $\mathbb{R}$ towards a continuous limit function $N(t)$. Further, we have

$$|N(t) - P^\nu(t)| \le \sum_{k=\nu}^{\infty} |P^{k+1}(t) - P^k(t)| \le cc_1 \sum_{k=\nu}^{\infty} \gamma^k = cc_1\gamma^\nu/(1-\gamma),$$

which completes the proof. □

In the univariate case the conditions $s(1) = 2$ and $s(-1) = 0$ are necessary for convergence. Consequently, for a convergent process, $s(z)$ can be factorized as $s(z) = q(z)(1+z)$ with $q(z) = \sum_k q_k z^k$ and $q(1) = \sum_k q_k = 1$. We then have the following theorem for the univariate case.

Theorem 5.3.2. *Let $s(z)$ be factorized as $s(z) = (1+z)q(z)$. Assume that the coefficients of q satisfy the inequality*

$$\sum_{l\in\mathbb{Z}} |q_{k-2l}| \le \gamma < 1$$

for all k. Then, for some constant c_1 depending only on the support of q, we have $|p_k^\nu - p_{k-1}^\nu| \le c_1\gamma^\nu$ for all k and ν, and the subdivision converges uniformly with a convergence rate given by $(5.46)_{/208}$.

More generally, the following is valid. Assume that for some positive integer r the coefficients of the polynomial

$$\tilde{q}(z) = q(z)q(z^2)q(z^4)\cdots q(z^{2^{r-1}}) = \sum_{k\in\mathbb{Z}} \tilde{q}_k z^k$$

satisfy the inequality

$$\sum_{l\in\mathbb{Z}} |\tilde{q}_{k-2^r l}| \le \gamma < 1 \tag{5.51}$$

for all k. Then we have the same conclusion, with γ replaced by $\gamma^{1/2^{r-1}}$.

Proof. We first deal with the case $r = 1$. We have

$$p^{\nu+1}(z) = q(z)(1+z)p^\nu(z^2)$$

and therefore

$$(1-z)p^{\nu+1}(z) = q(z)(1-z^2)p^\nu(z^2).$$

Since $(1-z)p^{\nu+1}(z) = \sum_{l\in\mathbb{Z}}(p_l^{\nu+1} - p_{l-1}^{\nu+1})z^l$ and $(1-z^2)p^\nu(z^2) = \sum_{l\in\mathbb{Z}}(p_l^\nu - p_{l-1}^\nu)z^{2l}$, we conclude that

$$p_k^{\nu+1} - p_{k-1}^{\nu+1} = \sum_{l\in\mathbb{Z}} q_{k-2l}(p_l^\nu - p_{l-1}^\nu).$$

Then, if $|p_l^\nu - p_{l-1}^\nu| < c_1\gamma^\nu$, it follows that

$$|p_l^{\nu+1} - p_{l-1}^{\nu+1}| < c_1\gamma^{\nu+1},$$

and by induction, $|p_l^\nu - p_{l-1}^\nu| < c_1\gamma^\nu$ is true for all ν. By Lemma 5.3.1$_{/208}$ and the estimate (5.46)$_{/208}$ the first statement follows.

Next we turn to the second statement. To simplify, we carry out the proof only for the case $r = 2$, i.e., for $\tilde{q}(z) = q(z)q(z^2)$. The general case is very similar. We have

$$p^{\nu+2}(z) = (1+z)q(z)(1+z^2)q(z^2)p^\nu(z^4)$$

and, multiplying by $1-z$,

$$(1-z)p^{\nu+2}(z) = (1-z^4)q(z)q(z^2)p^\nu(z^4) = \tilde{q}(z)(1-z^4)p^\nu(z^4).$$

We conclude that

$$p_k^{\nu+2} - p_{k-1}^{\nu+2} = \sum_{l\in\mathbb{Z}} \tilde{q}_{k-4l}(p_l^\nu - p_{l-1}^\nu).$$

Then it follows that

$$\max_k |p_k^{2\nu} - p_{k-1}^{2\nu}| \le \gamma \max_l |p_l^{2\nu-2} - p_{l-1}^{2\nu-2}| \le \cdots \le \gamma^\nu \max_l |p_l^0 - p_{l-1}^0|$$

and

$$\max_k |p_k^{2\nu+1} - p_{k-1}^{2\nu+1}| \le \gamma \max_l |p_l^{2\nu-1} - p_{l-1}^{2\nu-1}| \le \cdots \le \gamma^\nu \max_l |p_l^1 - p_{l-1}^1|,$$

i.e., that

$$\max_k |p_k^\nu - p_{k-1}^\nu| \le c_1 (\gamma^{1/2})^\nu,$$

which completes the proof. □

For the analysis of convergence in the bivariate case, the following lemma is useful.

Lemma 5.3.3. *Let the polynomial $s(z)$ be factorized as*

$$s(z) = (1 + z^e) q(z),$$

where $e \in \mathbb{Z}^2$ and $q(1,1) = 2$. Assume that the coefficients of q satisfy the inequality

$$\sum_{l \in \mathbb{Z}^2} |q_{k-2l}| \le \gamma < 1. \tag{5.52}$$

Then there exists a constant c_1 depending only on the support of q such that

$$|p_k^\nu - p_{k-e}^\nu| \le c_1 \gamma^\nu \tag{5.53}$$

for all k and ν.

More generally the following is valid. Assume that for some integer r the coefficients of the polynomial

$$\tilde{q}(z) = q(z) q(z^2) q(z^4) \cdots q(z^{2^{r-1}}) = \sum_{k \in \mathbb{Z}^2} \tilde{q}_k z^k$$

satisfy the inequality

$$\sum_{l \in \mathbb{Z}^2} |\tilde{q}_{k-2^r l}| \le \gamma < 1. \tag{5.54}$$

Then we have the same conclusion with γ replaced by $\gamma^{1/2^{r-1}}$.

The proof is very similar to the proof of Theorem $5.3.2_{/210}$ and is omitted.

Theorem 5.3.4. *Let $s(z) = (1 + z^{e_1}) q_1(z) = (1 + z^{e_2}) q_2(z)$, where e_1 and $e_2 \in \mathbb{Z}^2$ generate $\mathbb{Z}^2$. Assume that both factors q_1 and q_2 satisfy an estimate $(5.52)_{/212}$, or, more generally, that the corresponding polynomials $\tilde{q}_1$ and $\tilde{q}_2$ satisfy an estimate $(5.54)_{/212}$. Then the process is convergent with a convergence rate given by $(5.46)_{/208}$, or the same rate with γ replaced by $\gamma^{1/2^{r-1}}$.*

Proof. By Lemma $5.3.3_{/212}$ we have

$$|p_k^\nu - p_{k-e_i}^\nu| \le c_1' (\gamma^{1/2^{r-1}})^\nu \tag{5.55}$$

for $i = 1, 2$ and some constant c_1'. Then, using that

$$(1,0) = n_{11}e_1 + n_{12}e_2,$$
$$(0,1) = n_{21}e_1 + n_{22}e_2$$

with n_{ij} integers, we conclude that

$$|p_k^\nu - p_{k-e}^\nu| \le c_1(\gamma^{1/2^{r-1}})^\nu \tag{5.56}$$

for $e = (1,0)$ and $e = (0,1)$ with some constant c_1 depending on c_1' and the numbers n_{ij}. The convergence now follows by Lemma 5.3.1$_{/208}$. □

For tensor-product subdivision methods we have the following result.

Theorem 5.3.5. *Assume that we are given two univariate subdivision polynomials $s_1(z)$ and $s_2(z)$ defining uniformly convergent processes with nodal functions $N_1(t)$, and $N_2(t)$, respectively. Then the subdivision process defined by the bivariate polynomial $s_1(z_1)s_2(z_2)$ defines a convergent process with nodal function $N(u,v) = N_1(u)N_2(v)$.*

Exercise 3$_{/245}$ asks for a proof of this theorem.

By Remark 5.2.11$_{/207}$, convergence of the generalized four-point method, and the Butterfly method, is assured for all parameter values $k/2^k \in \mathbb{Z}^2/2^\nu$, $\nu \ge 0$. But in order to prove that we have convergence to a continuous limit function, which is nontrivial, we need to use the results developed in this section, above. This is done in the next two examples.

Example 5.3.6. Convergence for the generalized four-point method.

The generalized four-point method has the subdivision polynomial given by (4.14)$_{/150}$, i.e.,

$$s(z) = -wz^{-3} + (1/2 + w)z^{-1} + 1 + (1/2 + w)z - wz^3,$$

where $w = 1/16$ for the ordinary method, i.e., $s(z) = (-z^{-3} + 9z^{-1} + 16 + 9z - z^3)/16$. It is easy to verify that

$$s(z) = (-wz^5 + wz^4 + z^3/2 + z^2/2 + wz - w)(z+1)/z^3,$$

i.e., that $s(z) = (1+z)q(z)$ with

$$q(z) = (-wz^5 + wz^4 + z^3/2 + z^2/2 + wz - w)/z^3.$$

We now have

$$\sum_l |q_{-2l}| = \sum_l |q_{1-2l}| = |w| + 1/2 + |w| = 2|w| + 1/2,$$

and therefore we have uniform convergence if $2|w| + 1/2 = \gamma < 1$, i.e., if $|w| < 1/4$. For $w = 1/16$ we have $\gamma = 5/8$. The convergence rate is given by the inequality $|N(t) - P^\nu(t)| \le c(1/2 + 2|w|)^\nu$, where c is some constant. ∎

Example 5.3.7. Convergence for the Butterfly method.

The Butterfly method, presented in Section 4.2.1, has subdivision polynomial given by $(4.16)_{/153}$, i.e.,

$$\begin{aligned} s(z) = s(z_1, z_2) = 1 + \frac{1}{2}(&z_1 + z_1^{-1} + z_2 + z_2^{-1} + z_1^{-1}z_2 + z_1 z_2^{-1}) \\ &+ 2w(z_1 z_2 + z_1^{-1}z_2^{-1} + z_1^{-1}z_2^2 + z_1 z_2^{-2} + z_1^{-2}z_2 + z_1^2 z_2^{-1}) \\ &- w(z_1^2 z_2 + z_1^{-2}z_2^{-1} + z_1 z_2^2 + z_1^{-1}z_2^{-2} + z_1^{-1}z_2^3 + z_1^1 z_2^{-3} \\ &\quad + z_1^{-2}z_2^3 + z_1^2 z_2^{-3} + z_1^{-3}z_2^2 + z_1^3 z_2^{-2} + z_1^{-3}z_2 + z_1^3 z_2^{-1}). \end{aligned} \tag{5.57}$$

and can be factorized as

$$\begin{aligned} s(z) = (1 + z_1)(&-wz_1^{-2}z_2^3 - wz_1^{-3}z_2^2 + wz_1^{-2}z_2^2 + wz_1^{-1}z_2^2 - wz_2^2 \\ &- wz_1^{-3}z_2 + 3wz_1^{-2}z_2 + (1/2 - 3w)z_1^{-1}z_2 + 3wz_2 - wz_1 z_2 \\ &+ z_1^{-1}/2 + 1/2 - wz_1^{-2}z_2^{-1} + 3wz_1^{-1}z_2^{-1} + (1/2 - 3w)z_1^{-1} \\ &+ 3wz_1^{-1}z_2 - wz_1^{-1}z_2^2 - wz_1^{-1}z_2^{-2} + wz_2^{-2} + wz_1 z_2^{-2} \\ &- wz_1^2 z_2^{-2} - wz_1 z_2^{-3}), \end{aligned} \tag{5.58}$$

$$\begin{aligned} s(z) = (1 + z_1)(1 + z_2)(&-wz_1^{-2}z_2^2 - wz_1^{-3}z_2 + 2wz_1^{-2}z_2 + wz_1^{-1}z_2 - wz_2 \\ &+ wz_1^{-2} + (1/2 - 4w)z_1^{-1} + 4w - wz_1 - wz_1^{-2}z_2^{-1} \\ &+ 4wz_1^{-1}z_2^{-1} + (1/2 - 4w)z_2^{-1} + wz_1^1 z_2^{-1} - wz_1^{-1}z_2^{-2} \\ &+ wz_2^{-2} + 2wz_1 z_2^{-2} - wz_1^2 z_2^{-2} - wz_1^{-3}z_2), \end{aligned} \tag{5.59}$$

and

$$\begin{aligned} s(z) = (z_1^{1/2} + z_1^{-1/2})&(z_2^{1/2} + z_2^{-1/2})(z_1^{1/2}z_2^{-1/2} + z_1^{-1/2}z_2^{1/2}) \\ &\cdot\big((1/2 - 6w) + 2w(z_1 + z_1^{-1}) + 2w(z_2 + z_2^{-1}) + 2w(z_1 z_2^{-1} + z_1^{-1}z_2) \\ &\quad - w(z_1 z_2 + z_1^{-1}z_2^{-1}) - w(z_1^{-2}z_2 + z_1^2 z_2^{-1}) - w(z_1^{-1}z_2^2 + z_1 z_2^{-2})\big). \end{aligned} \tag{5.60}$$

The factorizations in $(5.59)_{/214}$ and $(5.60)_{/214}$ are not needed until the next section.

By $(5.58)_{/214}$ we have $s(z) = (1 + z_1)q(z)$ (defining $q(z)$), and it is straightforward to verify that the coefficients of $q(z) = \sum_k q_k z^k$ satisfy

$$\max_k \sum_{l \in \mathbb{Z}^2} |q_{k-2l}| = \max\{1/2 + 6|w|, |1/2 - 3w| + 9|w|\} = \gamma(w).$$

We find that $\gamma(w) < 1$ if and only if $-1/24 < w < 1/12$. In this case

$$\gamma(w) = \begin{cases} 1/2 + 6w & \text{if} \quad w \geq 0, \\ 1/2 - 12w & \text{if} \quad w < 0. \end{cases}$$

Consequently, by Lemma $5.3.3_{/212}$, $\max_k |p_k^\nu - p_{k-e}^\nu| \leq c\gamma^\nu$ for some constant c, where $e = (1, 0)$. Due to the symmetry we get the same estimate for $s(z) = (1 + z_2)q(z)$ and $e = (0, 1)$. By Theorem $5.3.4_{/212}$ we have convergence with rate given by $(5.46)_{/208}$. ∎

5.3.2 Smoothness

The nodal function $N(y)$ constructed in the previous section is continuous if the convergence criteria are valid. In order to establish higher-order regularity, techniques similar to those in the previous section may be used. We first give a theorem for the univariate case.

Theorem 5.3.8. *Assume that*

$$s(z) = \left(\frac{1+z}{2}\right)^k (1+z)q(z)$$

with k a positive integer. Also assume that q satisfies the conditions of Theorem 5.3.2$_{/210}$. Then the subdivision process is convergent and $N \in C^k(\mathbb{R})$.

Proof. We give the proof for the case that $k = 1$ and the parameter r in Theorem 5.3.2$_{/210}$ is equal to 1. Consider the differences

$$\left\{\frac{p_l^\nu - p_{l-1}^\nu}{2^\nu}\right\}_{\nu=0}^{\infty} = \left\{\frac{\Delta p_l^\nu}{2^\nu}\right\}_{\nu=0}^{\infty},$$

where Δ denotes a backward difference. Since

$$p^{\nu+1}(z) = \frac{(1+z)^2}{2} q(z) p^\nu(z^2),$$

we have

$$(1-z)^2 p^{\nu+1}(z) = \frac{1}{2} q(z)(1-z^2)^2 p^\nu(z^2).$$

Now, if $q(z) = \sum_{i\in\mathbb{Z}} q_i z^i$, it follows that

$$\Delta^2 p_i^{\nu+1} = \frac{1}{2} \sum_{l\in\mathbb{Z}} q_{i-2l} \Delta^2 p_l^\nu,$$

i.e.,

$$2^{\nu+1}(\Delta p_i^{\nu+1} - \Delta p_{i-1}^{\nu+1}) = \sum_{l\in\mathbb{Z}} q_{i-2l} 2^\nu (\Delta p_l^\nu - \Delta p_{l-1}^\nu).$$

Then, if

$$2^\nu |\Delta p_l^\nu - \Delta p_{l-1}^\nu| \le c\gamma^\nu,$$

it follows (using the hypothesis of Theorem 5.3.2$_{/210}$ with $r = 1$) that

$$2^{\nu+1} |\Delta p_l^{\nu+1} - \Delta p_{l-1}^{\nu+1}| \le \left(\sum_{l\in\mathbb{Z}} |q_{i-2l}|\right) c\gamma^\nu \le c\gamma^{\nu+1},$$

and by induction that

$$2^\nu|\Delta p_l^\nu - \Delta p_{l-1}^\nu| \le c\gamma^\nu$$

for all ν.

It now follows from Lemma 5.3.1$_{/208}$ that for some continuous function $N_1(t)$,

$$|N_1(t) - 2^\nu \Delta p_l^\nu| \to 0$$

uniformly as $|t - l/2^\nu| \to 0$. Introducing the notation

$$p^\nu(t) = p_l^\nu N^2(2^\nu t - l),$$

we have, using $(2.61)_{/81}$ with $h = 2^{-\nu}$, that

$$(p^\nu)'(t) = \sum_{l\in\mathbb{Z}+1/2} (p_{l+1/2}^\nu - p_{l-1/2}^\nu)2^\nu N^1(2^\nu t - l)$$

and we conclude that $(p^\nu)'(t) - N_1(t) \to 0$ uniformly as $\nu \to \infty$. Therefore, $p^\nu(t) - p^\nu(0) - \int_0^t N_1(s)\,ds \to 0$ uniformly, i.e.,

$$p^\nu(t) \to p^\nu(0) + \int_0^t N_1(s)\,ds = N(t),$$

which proves that $N \in C^1(\mathbb{R})$. □

Example 5.3.9. Smoothness for the generalized four-point method.

For the polynomial $s(z) = wz^{-3} + (1/2 + w)z^{-1} + 1 + (1/2 + w)z - wz^3$, we have the factorization

$$s(z) = 2(-wz^4 + 2wz^3 + (1/2 - 2w) + 2wz - w)(1 + z)^2/2 = \frac{1+z}{2}(1 + z)q(z)$$

with $q(z) = 2(-wz^4 + 2wz^3 + (1/2 - 2w) + 2wz - w) = \sum_{k\in\mathbb{Z}} q_k z^k$. In order to apply Theorem 5.3.2$_{/210}$, we calculate

$$\sum_{l\in\mathbb{Z}} |q_{1-2l}| = 2(2|w| + 2|w|) = 8|w|$$

and

$$\sum_{l\in\mathbb{Z}} |q_{-2l}| = 2(|w| + |1/2 - 2w| + |w|) = 4|w| + 2|1/2 - 2w|.$$

It is easy to see that $4|w| + 2|1/2 - 2w| \ge 1$ for all w, which means that the simplest smoothness criterion ($r = 1$) of Theorems 5.3.8$_{/215}$ and 5.3.2$_{/210}$ fails.

We then turn to the criterion of Theorems 5.3.8$_{/215}$ and 5.3.2$_{/210}$ with the parameter $r = 2$. We compute the polynomial $q(z)q(z^2)$, and after simple (although slightly tedious) calculations we get

$$\begin{aligned}\tilde{q}(z) &= q(z)q(z^2)\\ &= 4(w^2z^{12} - 2w^2z^{11} - (w/2)z^{10} + 2w^2z^9 + (1/2 - w)wz^8 + wz^7\\ &\qquad + (1/4 - 2w)z^6 + wz^5 + (1/2 - w)wz^4 + 2w^2z^3 - (w/2)z^2 - 2w^2z + w^2).\end{aligned}$$

Now

$$\sum_l |\tilde{q}_{-4l}| = 4(w^2 + 2|(1/2 - w)w| + w^2) = 8w^2 + 8|(1/2 - w)w|,$$
$$\sum_l |\tilde{q}_{1-4l}| = 4(2w^2 + |w| + 2w^2) = 16w^2 + 4|w|,$$
$$\sum_l |\tilde{q}_{2-4l}| = 4(|w|/2 + |1/4 - 2w| + |w|/2) = 4|w| + |1 - 8w|,$$
$$\sum_l |\tilde{q}_{3-4l}| = 4(2w^2 + |w| + 2w^2) = 16w^2 + 4|w|.$$

We see that $w = 1/16$ permits $\gamma = 3/4$ in $(5.54)_{/212}$ with $r = 2$, and therefore the nodal function is in $C^1(\mathbb{R})$ (see [176, p. 35]).

A closer analysis gives that

- $\gamma \geq 1$ for $w \leq 0$, and the smoothness criterion fails;
- $\gamma = 1 - 4w < 1$ for $0 < w \leq (\sqrt{2} - 1)/2$, and we have $N \in C^1(\mathbb{R})$;
- $\gamma = 16w^2 + 4w < 1$ for $(\sqrt{2} - 1)/2 \leq w < (\sqrt{5} - 1)/2$, and we have $N \in C^1(\mathbb{R})$;
- $\gamma \geq 1$ for $w \geq (\sqrt{5} - 1)/2$, and the smoothness criterion fails.

The smallest value for $\gamma(w)$ is obtained for $w = (\sqrt{2} - 1)/2$ which gives the value $\gamma = 2 - \sqrt{2}$. ∎

For the bivariate case it is more difficult to formulate general results. However, Lemma $5.3.3_{/212}$ can be used as a tool, as demonstrated in the following smoothness analysis of the Butterfly method.

Example 5.3.10. Smoothness for the Butterfly method.

By $(5.60)_{/214}$ the nodal function $N(u, v)$ for the Butterfly method is associated with a polynomial

$$s(z) = (z_1^{1/2} + z_1^{-1/2})(z_2^{1/2} + z_2^{-1/2})(z_1^{1/2} z_2^{-1/2} + z_1^{-1/2} z_2^{1/2})q(z)$$

with $q(1) = 1/2$. Now, let $N_1(u, v)$ and $N_2(u, v)$ denote the nodal functions for the subdivision methods associated with the polynomial subfactors

$$s_1(z) \doteq (z_2^{1/2} + z_2^{-1/2})(z_1^{1/2} z_2^{-1/2} + z_1^{-1/2} z_2^{1/2})2q(z) = (z_2^{1/2} + z_2^{-1/2})q^1(z)$$

and

$$s_2(z) \doteq (z_1^{1/2} + z_1^{-1/2})(z_1^{1/2} z_2^{-1/2} + z_1^{-1/2} z_2^{1/2})2q(z) = (z_1^{1/2} + z_1^{-1/2})q^2(z),$$

respectively, so that $s(z) = (z_1^{1/2} + z_1^{-1/2})s_1(z)$ and $s(z) = (z_2^{1/2} + z_2^{-1/2})s_2(z)$. Here, $q^1(z) = q^2(z) = (z_1^{1/2} z_2^{-1/2} + z_1^{-1/2} z_2^{1/2})2q(z)$. For the Fourier transforms of

the nodal functions, we then have

$$\hat{N}(\omega_1,\omega_2)=\frac{\sin(\omega_1/2)}{\omega_1/2}\hat{N}_1(\omega_1,\omega_2)$$

and

$$\hat{N}(\omega_1,\omega_2)=\frac{\sin(\omega_2/2)}{\omega_2/2}\hat{N}_2(\omega_1,\omega_2).$$

We have the following lemma.

Lemma 5.3.11. *If the functions N_1 and N_2 are continuous, then $N\in C^1(\mathbb{R}^2)$.*

Proof. Let

$$F(u,v)=\int_{-1/2}^{1/2}N_1(u-t,v)\,dt.$$

Taking the Fourier transform we get

$$\begin{aligned}\hat{F}(\omega_1,\omega_2)&=\int_{-1/2}^{1/2}e^{-i\omega_1 t}\hat{N}_1(\omega_1,\omega_2)\,dt\\&=\frac{e^{i\omega_1/2}-e^{-i\omega_1/2}}{i\omega_1}\hat{N}_1(\omega_1,\omega_2)=\frac{\sin(\omega_1/2)}{\omega_1/2}\hat{N}_1(\omega_1,\omega_2)=\hat{N}(\omega_1,\omega_2).\end{aligned}$$

Consequently, $i\omega_1\hat{F}(\omega_1,\omega_2)=(e^{i\omega_1/2}-e^{-i\omega_1/2})\hat{N}_1(\omega_1,\omega_2)$, and it follows (see item 2 in Table A.1$_{/311}$) that

$$\partial_u F(u,v)=N_1(u+1/2,v)-N_1(u-1/2,v)$$

is continuous. In the same way it follows that

$$\partial_v F(u,v)=N_2(u,v+1/2)-N_2(u,v-1/2)$$

is continuous, and the proof is complete. □

By this lemma we conclude that in order to prove that $N\in C^1(\mathbb{R}^2)$, it suffices to prove that the subdivision processes associated with the polynomials

$$s_1(z)=(z_2^{1/2}+z_2^{-1/2})q^1(z)$$

and

$$s_2(z)=(z_1^{1/2}+z_1^{-1/2})q^2(z)$$

are uniformly convergent towards continuous limit functions. For this we use Lemma 5.3.3$_{/212}$.

By (5.59)$_{/214}$ we have $s_1(z)=(1+z_2)q^*(z)$ where

$$\begin{aligned}q^*(z)&=(z_1^{1/2}z_2^{-1/2}+z_1^{-1/2}z_2^{1/2})2q(z)\\&=2(-wz_1^{-2}z_2^2-\cdots-wz_1^{-3}z_2)\end{aligned}$$

is equal to twice the last factor in $(5.59)_{/214}$. We now apply Lemma $5.3.3_{/212}$ with $r = 1$ and calculate $\max_k \sum_{l\in\mathbb{Z}^2} |q^*_{k-2l}|$. We see then that

$$\sum_{l\in\mathbb{Z}^2} |q^*_{-2l}| = 16|w|$$

and

$$\sum_{l\in\mathbb{Z}^2} |q^*_{(-1,0)-2l}| = |1-8w| + 8|w| \geq 1 \quad \text{for all } w.$$

Therefore, as for the generalized four-point method, the simplest version of the smoothness criterion of Lemma $5.3.3_{/212}$ fails.

We then take the parameter $r = 2$ and compute

$$\gamma(w) = \max_k \sum_{l\in\mathbb{Z}^2} |\tilde{q}^*_{k-4l}|$$

for the polynomial $\tilde{q}^*(z) = q^*(z)q^*(z^2)$, which can be done using symbolic computation. The result is the following (the details are left to Project $1_{/246}$). If

$$w \in W = (0, (\sqrt{17}-1)/2), \tag{5.61}$$

then we have $|\gamma(w)| < 1$ and consequently

$$|p^\nu_k - p^\nu_{k-(0,1)}| \leq c_1(\gamma^{1/2})^\nu$$

for all k and ν, where p^ν_k are the elements of the refined control sequence for the nodal function $N_1(y)$. Moreover, explicit expressions for $\gamma(w)$ when $w \in W$ can be obtained. Again, the calculations are long and tedious and are best performed using symbolic computation. For $w = 1/16$ we get $\gamma = 7/8$.

Next, $s_1(z)$ may also be written as $s_1(z) = (1 + z_1 z_2^{-1})q^{**}(z)$ with $q^{**}(z) = (z_2^{1/2} + z_2^{-1/2})2q(z)$. We apply Lemma $5.3.3_{/212}$ for q^{**}. By the symmetry of the problem, the values of $\gamma(w)$ will be the same as for q^*. It follows that

$$|p^\nu_k - p^\nu_{k-(1,-1)}| \leq c_1(\gamma^{1/2})^\nu.$$

Then, if we apply Lemma $5.3.1_{/208}$ with $e_1 = (0,1)$ and $e_2 = (1,-1)$, we conclude that the subdivision process associated with the polynomial $s_1(z)$ is convergent and that the nodal function $N_1(u,v)$ is continuous.

Due to the symmetry of the problem, the analysis for s_2 gives the same result as for s_1. We conclude by Lemma $5.3.11_{/218}$ that $N \in C^1(\mathbb{R}^2)$ if $w \in W$. ∎

5.4 General comments on the nonregular case

The first three sections of this chapter discussed the convergence and smoothness of basic methods, i.e., methods corresponding to the lower row of Figure $1.30_{/33}$. To illustrate, the Catmull–Clark method reduces to the $LR(3\times 3)$ method when all faces are quadrilateral with valence 4. The results of Section 5.2 apply, so we have

quadratic convergence, and we know from Theorem $3.3.2_{/111}$ that the parametric continuity of the limit function is C^2. Similarly, the 4-8 subdivision method reduces to the box-spline method with subdivision polynomial given in $(3.59)_{/137}$ in the regular case (which is illustrated in Figure $3.24_{/137}$). Again we have quadratic convergence, and Theorem $3.3.2_{/111}$ guarantees C^4 parametric continuity for the limit functions. Finally, the Kobbelt method provides an example from the third column of Figure $1.30_{/33}$: this method reduces to the $4pt \times 4pt$ scheme in the case of a quadrilateral mesh with all vertices of valence 4, and as discussed in Example $5.3.9_{/216}$ the method converges to a function with C^1 parametric continuity.

In the three sections following this one, we turn our attention to the variant methods used for nonregular meshes. This corresponds to the upper row of Figure $1.30_{/33}$. In the nonregular case, the usual parametrization of the surface necessarily has a singularity (the parametric expression for the surface normal vanishes, and the expressions for the tangent vectors are linearly dependent), and special methods of analysis must be used even to show that standard subdivision methods produce locally well-defined surfaces in the neighbourhood of a nonregular point. Similarly, for convergence and smoothness, the methods of analysis are quite different from those used in the regular case and depend on the properties of certain local subdivision matrices.

In Section 1.4.3 we said that several different versions of the local subdivision matrix S are used in the literature. The focus in that section was on local subdivision matrices defined in terms of a k-ring of vertices with $k = 1$ (for example, the $(2n+1) \times (2n+1)$ matrix in the formula following $(1.17)_{/44}$ for the Catmull–Clark method, the middle 3×3 submatrix of the matrix given in $(1.16)_{/43}$ for the $LR(3)$ method, and the $(n+1) \times (n+1)$ matrix given in the solution to Exercise $12_{/49}$ in Chapter 1 for the Loop method). In this chapter, however, we also use local subdivision matrices corresponding to larger values of k. The choice of local subdivision matrix is discussed in detail, below. For example, in the case of Catmull–Clark, the matrix corresponding to a 1-ring is by itself only useful as a computational formula implementing the subdivision process. To establish convergence and smoothness, we need the matrix corresponding to a 2-ring, and to prove single sheetedness we need the matrix corresponding to a 3-ring.

To give some intuition for the relevance of the eigenstructure of a local subdivision matrix S, we give a short discussion involving some simplifying assumptions.

Suppose that 1 is a strictly dominant eigenvalue of S, and suppose also that S is not defective, i.e., there is a nonsingular matrix $X_{(K\times K)}$ with columns equal to the eigenvectors ξ^j of S, $j = 1, \ldots, K$, so that $SX = XD$, where $D_{(K\times K)}$ is a diagonal matrix with the eigenvalues of S on its diagonal. We then have $X^{-1}S = DX^{-1}$: the rows $(\eta^i)^t$ of X^{-1} are the left eigenvectors of S, $i = 1, \ldots, K$, and they satisfy $(\eta^i)^t\xi^i = 1$, $(\eta^i)^t\xi^j = 0, j \neq i$.

We denote the K control points to be transformed by S in a single subdivision step by $p_1, \ldots, p_K$, $p_i \in \mathbb{R}^N$, $i = 1, \ldots, K$, written as usual as row vectors, and we write

$$p = \begin{bmatrix} p_1 \\ \vdots \\ p_K \end{bmatrix}_{(K\times N)},$$

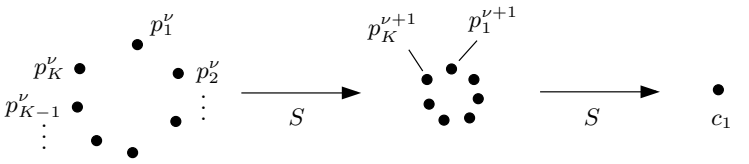


Figure 5.1. *Intuitive illustration of the relevance of eigenanalysis.*

possibly with a superscript ν indicating the iteration number. Then, a single subdivision step can be written as

$$p^{\nu+1} = Sp^\nu, \quad \nu = 0, 1, \ldots,$$

where p^0 is the initial control point p (see, for example, $(1.17)_{/44}$). Now, p can be expressed in terms of the basis of eigenvectors as

$$p = \xi^1 c_1 + \xi^2 c_2 + \cdots + \xi^K c_K, \tag{5.62}$$

where p is $(K \times N)$, ξ^j is $(K \times 1)$, and each c_j is a $(1 \times N)$ row vector of coefficients, $j = 1, \cdots, K$. If the subdivision process is applied ν times, we have

$$S^\nu p = \lambda_1^\nu \xi^1 c_1 + \cdots + \lambda_K^\nu \xi^K c_K,$$

and since $\lambda_1 = 1$ is a *strictly* dominant eigenvalue, the limit is

$$\lim_{\nu \to \infty} S^\nu p = \xi^1 c_1.$$

This follows since all but the first term eventually become negligible relative to the others.[39] If the subdivision process is affine invariant, the row sums of S are equal to 1, and the eigenvector ξ^1 has all components equal to 1. Thus, each row of $\xi^1 c_1$ is equal to c_1, and all K points $p_i \in \mathbb{R}^N$ converge to c_1. The geometric intuition is given in Figure $5.1_{/221}$, where the points $p_1^\nu, p_2^\nu, \ldots, p_K^\nu$ form a cluster of some size. At the next step, the points are normally clustered more closely, and in the limit all K sequences of points converge to $c_1 \in \mathbb{R}^N$.

If it is assumed further that the eigenvalues λ_2 and λ_3 are real, and that $1 > \lambda_2 = \lambda_3 > |\lambda_k|$, $k = 4, \ldots, K$, and if $c_2 \times c_3 \neq 0$, then the vectors c_2 and c_3 can be interpreted as linearly independent tangent vectors of the surface at the point c_1. In fact, as $\nu \to \infty$, the subdivision process approximates c_1 by

$$\xi^1 c_1 + \lambda_2^\nu (\xi^2 c_2 + \xi^3 c_3),$$

and the second term can be viewed as a first-order error term in the plane tangent to the surface.

In Section 5.5 the *convergence* of methods in the nonregular case is discussed, and this is followed by an analysis of *smoothness* in Section 5.6. In the nonregular case, however, even if we have guarantees of convergence and smoothness we are still not quite done, and it is worth presenting an example immediately of the sort

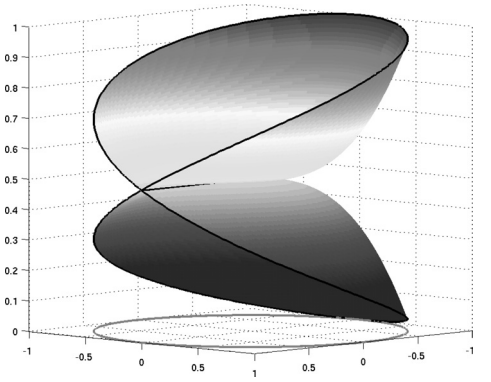


Figure 5.2. *A surface that is not single sheeted.*

of situation we wish to avoid. The danger is that the surface might be regular at a point, in the sense that it is well defined with a continuous normal vector, but it might not be *single sheeted*. An example similar to one given in [4] is shown in Figure $5.2_{/222}$: the surface is defined by $\left(r\cos 2\theta, r\sin 2\theta, \frac{1}{2}(1+r^3\sin\theta)\right)^t$ for $0 < r < 1$, $0 < \theta < 2\pi$, where $r = (u^2+v^2)^{1/2}$, $\theta = \tan^{-1}(v/u)$ if $u > 0$, $\theta = \tan^{-1}(v/u) + \pi$ if $u < 0$, and on the boundaries of the domain defined by $0 < r < 1$ and $0 < \theta < 2\pi$ by continuous extension. Here, the components $(r\cos 2\theta, r\sin 2\theta)^t$ correspond to the horizontal plane, and $\frac{1}{2}(1+r^3\sin\theta)$ to the vertical axis in the figure. Note that when we go once around the circle in the parametric domain defined by $r = 1$ and $0 \le \theta < 2\pi$, the projection of the surface in the horizontal plane goes twice around the circle indicated in Figure $5.2_{/222}$. In the limit, as r approaches 0 from the right, the surface has unit normal $(0,0,1)^t$ (see Exercise $4_{/245}$), but however small a neighbourhood of $(0,0,1)^t$, we consider there is always a point, in the neighbourhood and on the surface, which is the image of two different points in parameter space (r,θ). In fact, the distinct points $(r,0)$ and (r,π) are both mapped onto the same point $(r,0,1/2)^t$ on the surface. This rules out the possibility of a homeomorphism between an open neighbourhood of $r = 0$, in the parametric domain, and the part of the surface which is its image.

The convergence and smoothness analysis in a neighbourhood of a nonregular vertex follows the same pattern for most methods. In subsequent sections we illustrate the analyses with the example of the Catmull–Clark method in the neighbourhood of a nonregular point, and in the figures we show the case $n = 3$. A more general presentation is given in [124]: that presentation also focuses to some extent on a small number of example methods, including Catmull–Clark, but the presentation is generic and makes it clear for an arbitrary method exactly which properties of which subdivision matrices must be verified. Similarly, [172] gives general conditions for convergence that apply to *classes* of methods.

In the presentation we attempt to clarify certain aspects of the analysis that are sometimes left unclear in the literature. These aspects are summarized in a note.[40]

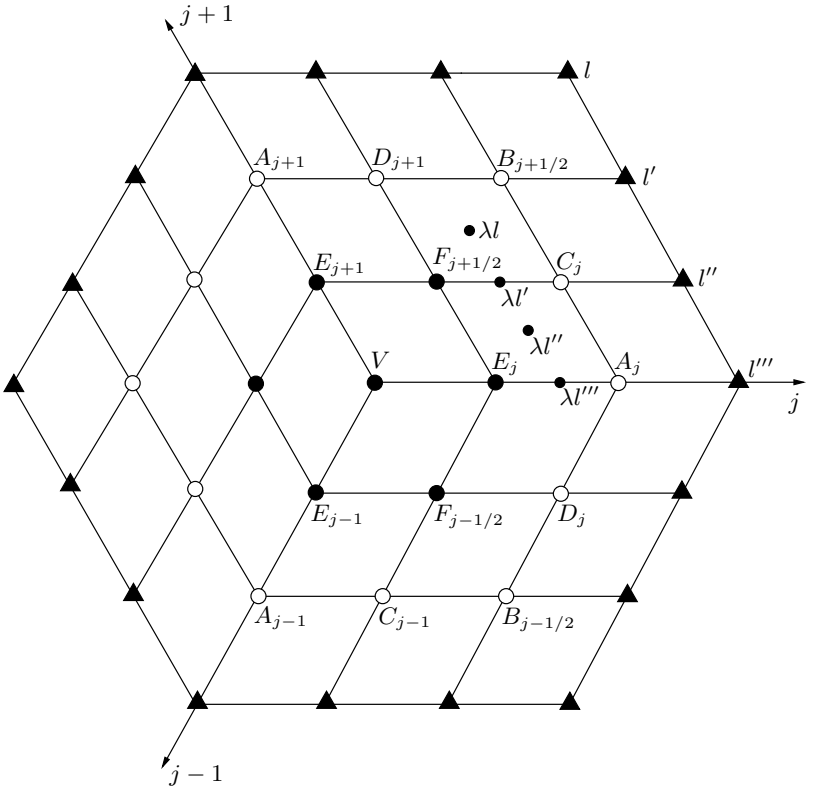


Figure 5.3. *Grid points around a nonregular vertex ($n = 3$).*

5.5 Convergence for the nonregular case (example of Catmull–Clark)

We consider the Catmull–Clark method in the neighbourhood of a point of valence n, which is assumed to be the origin in parameter space. See Figure 5.3$_{/223}$, where a set of initial grid points in $\mathbb{R}^2$ is shown. The notation for individual grid points, the significance of the different symbols used for different grid points, and the relevance of the points with labels involving l, l', l'', and l''' are explained below. The grid can be extended to an infinite grid $\mathbb{G}$ as will be described presently.

5.5.1 The parametric domain

We use the notation $e_j = (\cos(2\pi j/n), \sin(2\pi j/n))^t$, $j = 0, 1, \ldots, n-1$, for the unit vectors defining the directions of the grid, shown in Figure 5.3$_{/223}$ for the case $n = 3$. We have $e_j = e_{j+n}$ for all j. Then the set $\mathbb{G}_k$ defined as

$$\mathbb{G}_k = \Big\{ l \in \mathbb{R}^2 : l = \epsilon_j e_j + \epsilon_{j+1} e_{j+1}, \quad \text{with} \quad \epsilon_j = 0, 1, \ldots, k, \quad j = 0, 1, \ldots, n-1 \}$$

is called a k-ring neighbourhood around the origin [76]. In Figure $5.3_{/223}$, the grid points in $\mathbb{G}_1$ are indicated by black circles, the points in $\mathbb{G}_2$ are indicated by black circles and white circles, and the points in $\mathbb{G}_3$ make up all the points in the figure (the points in $\mathbb{G}_3 \setminus \mathbb{G}_2$ are indicated by black triangles). A 1-ring contains $2n+1$ grid points, and a k-ring contains $nk(k+1)+1$ grid points, Exercise $5_{/245}$ asks for verification of the number of grid points and for verification of certain other facts.

The grids denoted by $\mathbb{G}_k$ are similar to those introduced in Definition $3.1.1_{/94}$ and $(3.28)_{/116}$, although the directions e_j are enumerated in a different way. For example, in the case of the Catmull–Clark method and a regular vertex (see Example $3.2.4_{/100}$), we have $G_8 = \mathbb{G}_2$.

Let $\mathbb{G} = \bigcup_{k=1}^{\infty} \mathbb{G}_k$ be the infinite extension of an initial grid, and define the closed set

$$\bar{\mathbb{G}}_k = \Big\{ y \in \mathbb{R}^2 : y = \epsilon_j e_j + \epsilon_{j+1} e_{j+1} \quad \text{with} \quad 0 \le \epsilon_j \le k, \quad j = 0, 1, \ldots, n-1 \Big\}.$$

If we are given an initial set $\{p_l^0\}_{l \in \mathbb{G}}$ of control vectors, then the Catmull–Clark method defines a parametric surface

$$x(y) = \sum_{l \in \mathbb{G}} p_l^0 N_l(y), \tag{5.63}$$

where $N_l(y)$ denotes the nodal basis function centered around the point l, which is defined by choosing $p_k^0 = 1$ if $k = l$ and $p_k^0 = 0$ if $k \neq l$. The parametric domain $(\bar{G}_k)^0$ is the image $X(U)$ corresponding to some chart (X, U), $U \subseteq \boldsymbol{M}$, as defined in Section 4.6. The mapping X is piecewise affine in the sense that X^{-1} maps each parallelogram spanned by $\{ke_j, ke_{j+1}\}$ onto a part of a square face $\boldsymbol{F}_\alpha \subset \boldsymbol{M}$.

From the theory for the regular case, it follows that we have local uniform convergence towards a continuous function $x(y)$ outside the origin, in the following sense. For every $\delta > 0$ and every bounded set $A \subset \mathbb{R}^2$ such that $A \cap \{y : |y| < \delta\} = \emptyset$, we have

$$\max_{l/2^\nu \in A} |x(l/2^\nu) - p_l^\nu| \to 0 \quad \text{as} \quad \nu \to \infty.$$

After a piecewise-affine reparametrization, the nodal functions $N_l(y)$ are C^2 outside the origin.

In Section 5.5.3, we prove that in fact the process converges uniformly towards a continuous function $x(y)$ in some closed sphere $\{y : |y| \le \delta\}$. We also prove that the process defines a well-defined surface in a neighbourhood of the origin, and that we have tangent-plane continuity at the origin. This is done in Sections 5.6 and 5.7. First, however, we need some spectral information about certain local subdivision matrices.

5.5.2 Spectral analysis

The parametric surface $(5.63)_{/224}$ is uniquely determined for $y \in \bar{\mathbb{G}}_1 \setminus \{0\}$ by the vectors $\{p_l^0\}_{l \in \mathbb{G}_2}$, i.e.,

$$x(y) = \sum_{l \in \mathbb{G}_2} p_l^0 N_l(y).$$

More generally, the parametric surface is determined for $y \in \bar{\mathbb{G}}_{k-1} \setminus \{0\}$ by the vectors $\{p_l^0\}_{l\in\mathbb{G}_k}$.

If we perform the first step of the Catmull–Clark process, the infinite sequences $\{p_l^0\}_{l\in\mathbb{G}}$ are mapped linearly onto infinite sequences $\{p_l^1\}_{l\in\mathbb{G}}$. Also, the finite-dimensional vectors $p^{0,k} = \{p_l^0\}_{l\in\mathbb{G}_k}$ are mapped linearly onto vectors $p^{1,k} = \{p_l^1\}_{l\in\mathbb{G}_k}$ by some particular local subdivision matrix S_k (depending on the particular ordering of the indices $l \in \mathbb{G}_k$ which has been chosen). The dimension of S_k is $(nk(k+1)+1) \times (nk(k+1)+1)$, i.e., S_1 has dimension $(2n+1)\times(2n+1)$, S_2 has dimension $(6n+1)\times(6n+1)$, and S_3 has dimension $(12n+1)\times(12n+1)$. (In [124, p. 97] it is found convenient to replace S_3 by a $13n \times 13n$ matrix involving some redundancy.)

The matrices S_k are determined by the rules $(1.17)_{/44}$. Defining

$$p^{\nu,k} = \{p_l^\nu\}_{l\in\mathbb{G}_k}, \tag{5.64}$$

where p_l^ν is related to the parameter value $l/2^\nu \in \mathbb{R}^2$, we have

$$p^{\nu,k} = S_k^\nu p^{0,k}. \tag{5.65}$$

Since the subdivision procedure is affine invariant, the row sums of the matrices S_k are equal to 1, and $\lambda_1 = 1$ is an eigenvalue corresponding to the eigenvector $\xi^1 = (1,1,\ldots,1)^t$.

Also, we have the following block structure for the matrices S_k:

$$S_{k+1} = \begin{pmatrix} \sigma_k' & \sigma_k'' \\ 0 & S_k \end{pmatrix}. \tag{5.66}$$

This is true since the control vectors $\{p_l^{\nu-1}\}_{l\in\mathbb{G}_{k+1}\setminus\mathbb{G}_k}$ do not affect the new control vectors $\{p_l^\nu\}_{l\in\mathbb{G}_k}$. Therefore, if we denote

$$p^{\nu,k+1} = \begin{pmatrix} \rho^{\nu,k} \\ p^{\nu,k} \end{pmatrix},$$

where

$$\rho^{\nu,k} = \{p_l^\nu\}_{l\in\mathbb{G}_{k+1}\setminus\mathbb{G}_k},$$

then

$$\begin{pmatrix} \rho^{\nu+1,k} \\ p^{\nu+1,k} \end{pmatrix} = \begin{pmatrix} \sigma_k' & \sigma_k'' \\ 0 & S_k \end{pmatrix} \begin{pmatrix} \rho^{\nu,k} \\ p^{\nu,k} \end{pmatrix}, \tag{5.67}$$

i.e.,

$$\begin{aligned} \rho^{\nu+1,k} &= \sigma_k'\rho^{\nu,k} + \sigma_k''p^{\nu,k}, \\ p^{\nu+1,k} &= S_k p^{\nu,k}. \end{aligned} \tag{5.68}$$

Further, if $k \geq 2$, then $\sigma_k' = 0$, i.e.,

$$S_{k+1} = \begin{pmatrix} 0 & \sigma_k'' \\ 0 & S_k \end{pmatrix}. \tag{5.69}$$

This follows from the fact that for $k \geq 2$ the control vectors $\{p_l^{\nu-1}\}_{l \in \mathbb{G}_{k+1} \setminus \mathbb{G}_k}$ do not affect the new control vectors $\{p_l^{\nu}\}_{l \in \mathbb{G}_{k+1}}$.

From $(5.68)_{/225}$ we may conclude that the set of eigenvalues for S_{k+1} is the union of the set of eigenvalues for S_k and σ'_k. This follows from the fact, shown below, that S_k and σ'_k have no common eigenvalues. Thus, if λ is an eigenvalue of S_k with eigenvector p^k, so that $S_k p^k = \lambda p^k$, then λ is not an eigenvalue of σ'_k, and the corresponding eigenvector $p^{k+1} = (\rho^k, p^k)^t$ for S_{k+1} can be obtained by solving for ρ^k in the equation

$$(\sigma'_k - \lambda I)\rho^k = -\sigma''_k p^k. \tag{5.70}$$

Similarly, if ρ^k is an eigenvector of σ'_k with eigenvalue μ, so that

$$\sigma'_k \rho^k = \mu \rho^k, \tag{5.71}$$

then $(\rho^k, 0)^t$ is the corresponding eigenvector of S_{k+1}.

Since $\sigma'_k = 0$ for $k \geq 2$, it also follows that, for $k \geq 3$, the set of eigenvalues of S_k consists of the eigenvalues of S_2 (all of them simple) and the multiple eigenvalue $\lambda = 0$. The eigenspace of $\lambda = 0$ consists of the vectors $\{p_l\}_{l \in \mathbb{G}_k}$ having $p_l = 0$ if $l \in \mathbb{G}_2$.

Lemma 5.5.1. *The eigenvalues λ_i of the matrices S_1, S_2, and S_3 satisfy the condition*

$$1 = \lambda_1 > \lambda_2 = \lambda_3 > |\lambda_i| \quad \textit{for} \quad i \geq 3, \tag{5.72}$$

where $\lambda_2 = \lambda_3$ are equal for all three matrices. Moreover, all the eigenvalues are real, and explicit expressions for them can be found for all valences n.

The proof does not lend itself to our usual format, since it is quite long and is best divided into prominently labelled parts. It is therefore presented over the next few pages as ordinary text.

Although the matrices S_k are of rather high dimension, the spectral analysis is considerably simplified by the fact that they can be given a block-diagonal structure by using discrete Fourier series [117]. See Section A.2.4.

The matrix S_1 and its eigenstructure

We first investigate the matrix S_1. By $(1.17)_{/44}$ we have

$$\begin{aligned} V' &= \frac{4n-7}{4n} V + \frac{3}{2n}\left(\frac{1}{n}\sum_{i=0}^{n-1} E_i\right) + \frac{1}{4n}\left(\frac{1}{n}\sum_{i=0}^{n-1} F_{i+1/2}\right), \\ E'_j &= \frac{3}{8}(V + E_j) + \frac{1}{16}(E_{j-1} + F_{j-1/2} + F_{j+1/2} + E_{j+1}), \\ F'_{j+1/2} &= \frac{1}{4}(V + E_j + F_{j+1/2} + E_{j+1}), \end{aligned} \tag{5.73}$$

$j = 0, \ldots, n-1$, where the notation has been slightly changed: the subscript on F has been shifted by $1/2$, since this leads to real coefficients in the system $(5.76)_{/227}$, below. The grid points related to these values are illustrated in Figure $5.3_{/223}$.

According to the theory of discrete Fourier transforms, any n-periodic discrete sequence $\{E_j\}_{j=0}^{n-1}$ can be represented as

$$E_j = \sum_{r=0}^{n-1} e_r w^{jr},$$

where $w = e^{i2\pi/n}$ and

$$e_r = \frac{1}{n} \sum_{j=0}^{n-1} E_j w^{-jr}.$$

Similarly, the sequence $\{F_{j+1/2}\}_{j=0}^{n-1}$ can be written as

$$F_{j+1/2} = \sum_{r=0}^{n-1} f_r w^{(j+1/2)r} = \sum_{r=0}^{n-1} f_r w^{r/2} w^{jr}.$$

Inserting into $(5.73)_{/226}$, we get

$$\begin{aligned} V' &= (1-7/4n)V + (3/2n^2) \sum_{j,r} e_r w^{rj} + \frac{1}{4n^2} \left(\sum_{j,r} f_r w^{r/2} w^{rj} \right) \\ &= (1-7/4n)V + (3/2n)e_0 + (1/4n)f_0, \qquad (5.74) \\ E'_j &= 3V/8 + \sum_r \left(3/8 + (w^r + w^{-r})/16\right) e_r w^{rj} \\ &\quad + \sum_r (w^{r/2} + w^{-r/2}) f_r w^{rj}/16, \\ F'_{j+1/2} &= V/4 + \sum_r \left(e_r(w^{r/2} + w^{-r/2}) + f_r\right) w^{r(j+1/2)}/4. \end{aligned}$$

In $(5.74)_{/227}$ we have used that $\sum_{j=0}^{n-1} w^{rj} = n$ if $r = 0$ and $\sum_{j=0}^{n-1} w^{rj} = 0$ if $1 \le r \le n-1$.

Writing $E'_j = \sum_{r=0}^{n-1} e'_r w^{jr}$ and $F'_{j+1/2} = \sum_{r=0}^{n-1} f'_r w^{(j+1/2)r} = \sum_{r=0}^{n-1} f_r w^{r/2} w^{jr}$, we get, after identifying coefficients for each r, the following. For $r = 0$,

$$\begin{aligned} V' &= (1-7/4n)V + (3/2n)e_0 + (1/4n)f_0, \\ e'_0 &= 3V/8 + e_0/2 + f_0/8, \\ f'_0 &= V/4 + e_0/2 + f_0/4, \qquad (5.75) \end{aligned}$$

and for $1 \le r \le n-1$, using that $(w^r + w^{-r})/2 = \cos(2\pi r/n)$ and $(w^{r/2} + w^{-r/2})/2 = \cos(\pi r/n)$,

$$\begin{aligned} e'_r &= (3/8 + \cos(2\pi r/n)/8)e_r + \cos(\pi r/n) f_r/8, \\ f'_r &= \cos(\pi r/n) e_r/2 + f_r/4. \qquad (5.76) \end{aligned}$$

Now, λ is an eigenvalue of S_1 if and only if

$$V' = \lambda V, \quad E'_j = \lambda E_j, \quad F'_{j+1/2} = \lambda F_{j+1/2}$$

for some V, E_j, and $F_{j+1/2}$, $j = 0, \ldots, n-1$, not all equal to 0, i.e., if and only if either

$$\begin{aligned} &(1 - 7/4n - \lambda)V + (3/2n)e_0 + (1/4n)f_0 = 0, \\ &3V/8 + (1/2 - \lambda)e_0 + f_0/8 = 0, \\ &V/4 + e_0/2 + (1/4 - \lambda)f_0 = 0 \end{aligned} \tag{5.77}$$

with V, e_0, and F_0 not all 0, and $e_r = f_r = 0$ for $1 \le r \le n-1$; or, if for some r, $1 \le r \le n-1$,

$$\begin{aligned} &\left(\frac{3}{8} + \frac{1}{8}\cos(2\pi r/n) - \lambda\right) e_r + \frac{1}{8}\cos(\pi r/n) f_r = 0, \\ &\frac{1}{2}\cos(\pi r/n) e_r + \left(\frac{1}{4} - \lambda\right) f_r = 0 \end{aligned} \tag{5.78}$$

with e_r and f_r not both 0, and $V = e_0 = f_0 = 0$ and $e_s = f_s = 0$ for $1 \le s \le n-1$, $s \ne r$.

Computing the determinant for the system $(5.77)_{/228}$, we get

$$\begin{vmatrix} 1 - 7/4n - \lambda & 3/2n & 1/4n \\ 3/8 & 1/2 - \lambda & 1/8 \\ 1/4 & 1/2 & 1/4 - \lambda \end{vmatrix} = (1-\lambda)(\lambda^2 - (3/4 - 7/4n)\lambda + (1 - 3/n)/16) = 0,$$

and solving for λ,

$$\begin{aligned} &\lambda_1 = 1, \\ &\lambda = \left(3 - 7/n \pm \sqrt{(3 - 7/n)^2 - 4(1 - 3/n)}\,\right)/8. \end{aligned} \tag{5.79}$$

For $n = 3$ we obtain, in the latter expression, $\lambda = 0$ and $\lambda = 1/6$. For $\lambda_1 = 1$ we have, as mentioned earlier, the eigenvector defined by $V = 1$, $E_j = 1$, and $F_{j+1/2} = 1$ for $0 \le j \le n-1$. Similarly, computing the eigenvalues for the systems $(5.78)_{/228}$, we have

$$\begin{vmatrix} \frac{3}{8} + \frac{1}{8}\cos(2\pi r/n) - \lambda & \frac{1}{8}\cos(\pi r/n) \\ \frac{1}{2}\cos(\pi r/n) & \frac{1}{4} - \lambda \end{vmatrix} = \lambda^2 - \frac{1}{4}(2 + \cos^2(\pi r/n))\lambda + \frac{1}{16} = 0 \tag{5.80}$$

and the eigenvalues

$$\lambda = \left(5 + \cos(2\pi r/n) \pm [(1 + \cos(2\pi r/n))(9 + \cos(2\pi r/n))]^{1/2}\right)/16. \tag{5.81}$$

The largest of these eigenvalues are obtained with the plus sign and $r = 1$ or $r = n-1$, which gives a double eigenvalue

$$\lambda_{2,3} = \left(5 + \cos(2\pi/n) + [(1 + \cos(2\pi/n))(9 + \cos(2\pi/n))]^{1/2}\right)/16. \tag{5.82}$$

It is straightforward to verify that $\lambda_{2,3} < 1/2$ for $n = 3$ and $\lambda_{2,3} > 1/2$ for $n > 4$. For the regular case, $n = 4$, we have $\lambda_{2,3} = 1/2$. See Exercise $6_{/245}$.

Equations $(5.77)_{/228}$ and $(5.79)_{/228}$ give three eigenvalues and linearly independent eigenvectors defined by

$$V, \quad E_j = e_0, \quad \text{and} \quad F_{j+1/2} = f_0, \quad j = 0, 1, \ldots, n-1, \tag{5.83}$$

with V, e_0, and f_0 being the solutions of $(5.77)_{/228}$. Similarly, $(5.78)_{/228}$ and $(5.81)_{/228}$ give (taking the solutions e_r and f_r of $(5.78)_{/228}$ to be real with $e_r = e_{n-r}$ and $f_r = f_{n-r}$) $n-1$ different real eigenvalues and $2(n-1)$ (two for each eigenvalue) linearly independent eigenvectors defined by

$$V = 0, \quad E_j = e_r w^{jr}, \quad \text{and} \quad F_{j+1/2} = f_r w^{(j+1/2)r}, \quad j = 0, 1, \ldots, n-1.$$

Let

$$\begin{aligned} \xi &= (\ldots, F_{j+1/2}, \ldots \mid \ \ldots, E_j, \ldots \mid 0)^t \\ &= (\ldots, f_r\, \omega^{(j+1/2)r}, \ldots \mid \ \ldots, e_r \omega^{jr}, \ldots \mid 0)^t. \end{aligned} \tag{5.84}$$

Real-valued eigenvectors are obtained by taking real and imaginary parts, so that

$$E_j = e_r \cos(2\pi jr/n), \quad F_{j+1/2} = f_r \cos(2\pi(j+1/2)r/n)$$

and

$$E_j = e_r \sin(2\pi jr/n), \quad F_{j+1/2} = f_r \sin(2\pi(j+1/2)r/n),$$

respectively. Thus,

$$\begin{aligned} \xi^2 &= \Re\xi \\ &= f_r(\ldots, \cos(2\pi(j+1/2)r/n), \ldots \mid 0, \ldots, 0 \mid 0)^t, \\ &\quad + e_r(0, \ldots, 0 \mid \ \ldots, \cos(2\pi jr/n), \ldots \mid 0)^t, \\ \xi^3 &= \Im\xi \\ &= f_r(\ldots, \sin(2\pi(j+1/2)r/n), \ldots \mid 0, \ldots, 0 \mid 0)^t \\ &\quad + e_r(0, \ldots, 0 \mid \ \ldots, \sin(2\pi jr/n), \ldots \mid 0)^t. \end{aligned} \tag{5.85}$$

To summarize, we have, for the local subdivision matrix S_1, $3 + 2(n-1) = 2n+1$ linearly independent eigenvectors, spanning $\mathbb{R}^{2n+1}$.

The corresponding eigenvectors obtained by $(5.78)_{/228}$ are given by, for example, $e_1 = 1$, $f_1 = \cos(\pi/n)/2(\lambda - 1/4) = 2\cos(\pi/n)/(4\lambda - 1)$. Using $(5.81)_{/228}$, we get after some calculation

$$e_1 = 1, \quad f_1 = \sqrt{4 + \cos^2(\pi/n)} - \cos \pi/n.$$

Considering the subdominant double eigenvalue $\lambda_{2,3}$, where $r = 1$ or $r = n-1$, we note the following:

$$\begin{aligned} &\text{for} \quad n = 3, \quad f_1/e_1 = (\sqrt{17} - 1)/2 > \sqrt{2}, \\ &\text{for} \quad n = 4, \quad f_1/e_1 = \sqrt{2}, \quad \text{and} \\ &\text{for} \quad n > 4, \quad f_1/e_1 < \sqrt{2}. \end{aligned} \tag{5.86}$$

All of the eigenvalues of S_1 are given by $(5.79)_{/228}$ and $(5.80)_{/228}$, and all of the eigenvectors except $(1, \ldots, 1)^t$ are given by $(5.83)_{/229}$ and $(5.84)_{/229}$.

The matrices S_2 and S_3 and their eigenstructures

Having completed the spectral analysis of the matrix S_1, we consider the matrices S_2 and S_3. The analysis involves a change of basis, using the discrete Fourier transform, which leads to modified matrices S_{k+1} in $(5.66)_{/225}$. However, these matrices have the same block structure, and the same eigenvalues, as the original matrices: only the eigenvectors change due to the change in basis.

In Figure $5.3_{/223}$ the grid points related to $\mathbb{G}_2$ are depicted, together with the corresponding values V, E_j, $F_{j+1/2}$, A_j, $B_{j+1/2}$, C_j, and D_j for the control points. In addition to $(5.73)_{/226}$ we now have by $(1.17)_{/44}$

$$\begin{aligned}
A'_j &= (1-7/16)E_j + (F_{j-1/2} + F_{j+1/2} + V + A_j)3/32 \\
&\quad + (D_j + C_j + E_{j-1} + E_{j+1})/64, \\
B'_{j+1/2} &= (1-7/16)F_{j+1/2} + (E_j + C_j + D_{j+1} + E_{i+1})3/32 \\
&\quad + (A_j + B_{j+1/2} + A_{j+1} + V)/64, \\
C'_j &= (3/8)F_{j+1/2} + (3/8)E_j + (E_{j+1} + C_j + V + A_j)/16, \\
D'_{j+1} &= (3/8)F_{j+1/2} + (3/8)E_{j+1} + (E_j + D_{j+1} + V + A_{j+1})/16.
\end{aligned} \tag{5.87}$$

Next, inserting the expansions

$$\begin{aligned}
A_j &= \sum_{r=0}^{n-1} a_r w^{jr}, \quad B_{j+1/2} = \sum_{r=0}^{n-1} b_r w^{(j+1/2)r}, \\
C_j &= \sum_{r=0}^{n-1} c_r w^{jr}, \quad D_j = \sum_{r=0}^{n-1} d_r w^{jr}
\end{aligned}$$

and the corresponding expansions for the primed variables into $(5.87)_{/230}$ we get, after identifying coefficients, the following. For $r = 0$, we have $(5.75)_{/227}$ and

$$\begin{aligned}
a'_0 &= (3/32)a_0 + (c_0 + d_0)/64 + (19/32)e_0 + (3/16)f_0 + (3/32)V, \\
b'_0 &= (1/32)a_0 + b_0/64 + (3/32)(c_0 + d_0) + (3/16)e_0 + (9/16)f_0 + V/64, \\
c'_0 &= a_0/16 + c_0/16 + (7/16)e_0 + (3/8)f_0 + V/16, \\
d'_0 &= a_0/16 + d_0/16 + (7/16)e_0 + (3/8)f_0 + V/16.
\end{aligned} \tag{5.88}$$

Similarly, for $1 \le r \le n-1$ we have $(5.76)_{/227}$ and

$$\begin{aligned}
a'_r &= (3/32)a_r + (c_r + d_r)/64 + (9/16 + (w^r + w^{-r})/64)e_r \\
&\quad + (3/32)(w^{r/2} + w^{-r/2})f_r, \\
b'_r &= (w^{r/2} + w^{-r/2})a_r/64 + b_r/64 + (3/32)(w^{-r/2}c_r + w^{r/2}d_r) \\
&\quad + (3/32)(w^{r/2} + w^{-r/2})e_r + (9/16)f_r, \\
c'_r &= a_r/16 + c_r/16 + (3/8 + w^r/16)e_r + (3/8)w^{r/2}f_r, \\
d'_r &= a_r/16 + d_r/16 + (3/8 + w^{-r}/16)e_r + (3/8)w^{-r/2}f_r.
\end{aligned} \tag{5.89}$$

(The notation c_r used here conflicts slightly with the notation introduced in $(5.62)_{/221}$, which is used again below. But we will use the notation c_r for Fourier coefficients only in the next four paragraphs.)

In the formula $(5.66)_{/225}$, with $k = 1$, the submatrix σ_1' represents the mapping of a_r, b_r, c_r, d_r onto a_r', b_r', c_r', d_r' when $e_r = f_r = V = 0$ for $r = 0, \dots, n-1$. See Exercise $7_{/245}$. The eigenvalues of σ_1' are therefore obtained from the following equations for $0 \le r \le n-1$:

$$\begin{vmatrix} 3/32-\lambda & 0 & 1/64 & 1/64 \\ \cos(\pi r/n)/32 & 1/64-\lambda & (3/32)w^{-r/2} & (3/32)w^{r/2} \\ 1/16 & 0 & 1/16-\lambda & 0 \\ 1/16 & 0 & 0 & 1/16-\lambda \end{vmatrix} = 0.$$

After straightforward calculations we find that, independently of n and r, the determinant has the zeroes $1/16$ (double), $1/64$, and $(5 \pm \sqrt{21})/32$. This shows that the subdominant eigenvalues $\lambda_{2,3}$ for the matrix S_2 are equal to the subdominant eigenvalues for S_1 (the smallest subdominant eigenvalue of S_1 occurs for $n = 3$, when $\lambda_{2,3} = (9 + \sqrt{17})/32 > \max\{1/16, 1/64, (5 \pm \sqrt{21})/32\}$). Thus, the subdominant eigenvalues of S_2 are given by the formula $(5.82)_{/228}$. Also, these calculations show that σ_1' and S_1 have no common eigenvalues, which means that $(5.70)_{/226}$ can be solved.

It can be verified in a similar way that the subdominant eigenvalues $\lambda_{2,3}$ for the matrix S_3 are also equal to the subdominant eigenvalues for S_1.

In order to obtain the eigenvalues and eigenvectors for S_2, we proceed as follows. The eigenvalues and eigenvectors for S_1 have already been found above in $(5.77)_{/228}$ and $(5.78)_{/228}$ and are inserted into $(5.88)_{/230}$ and $(5.89)_{/230}$. We then get

$$\begin{aligned} &(3/32-\lambda)a_r + (c_r+d_r)/64 + (9/16 + (w^r + w^{-r})/64)e_r \\ &+ (3/32)(w^{r/2} + w^{-r/2})f_r = 0, \\ &(w^{r/2} + w^{-r/2})a_r/64 + (1/64-\lambda)b_r + (3/32)(w^{-r/2}c_r + w^{r/2}d_r), \\ &+ (3/32)(w^{r/2} + w^{-r/2})e_r + (9/16)f_r = 0, \\ &a_r/16 + (1/16-\lambda)c_r + (3/8 + w^r/16)e_r + (3/8)w^{r/2}f_r = 0, \\ &a_r/16 + (1/16-\lambda)d_r + (3/8 + w^{-r}/16)e_r + (3/8)w^{-r/2}f_r = 0. \end{aligned} \tag{5.90}$$

Solving for a_r, b_r, c_r, and d_r, we obtain eigenvectors for the matrix S_2. This step corresponds to solving $(5.70)_{/226}$ with $k = 1$ and the right-hand side known. The remaining eigenvalues and eigenvectors can be found by taking $e_r = 0$ and $f_r = 0$ in $(5.90)_{/231}$ and solving the eigenvalue problem. This step corresponds to solving $(5.71)_{/226}$ for ρ_1 and μ.

Finally, again using the block structure in $(5.66)_{/225}$, the eigenvalues and eigenvectors for S_3 can be calculated in a similar way as above. In this case the matrix σ_2' is the zero matrix and $(5.70)_{/226}$ takes the form

$$\lambda\rho^2 = \sigma_2'' p^2.$$

The eigenvectors corresponding to the eigenvalues already known for S_2 may, however, be found in a simpler way, namely by interpolation using $(5.73)_{/226}$. This is illustrated by Figure $5.3_{/223}$, where the eigenvector components V, E_j, $F_{j+1/2}$, C_j, and D_{j+1} have been shown. The value indicated by l in Figure $5.3_{/223}$, which is a component in the corresponding eigenvector for S_3, can be obtained from

$$\lambda l = (F_{j+1/2} + C_j + B_{j+1/2} + D_{j+1})/4.$$

This results from the fact that in Figure $5.3_{/223}$, λl is equal to the value in a new node in the middle of an old face, so that the expression just given follows by direct substitution in the third line of $(5.73)_{/226}$. Similar expressions for the other components of the eigenvector when $l \in \mathbb{G}_3 \setminus \mathbb{G}_2$ can also be given:

$$\lambda l' = \frac{3}{8}(F_{j+1/2} + C_j) + \frac{1}{16}(E_j + A_j + B_{j+1/2} + D_{j+1}),$$

$$\lambda l'' = \frac{1}{4}(F_{j+1/2} + E_j + A_j + C_j),$$

$$\lambda l''' = \frac{3}{8}(E_j + A_j) + \frac{1}{16}(F_{j-1/2} + D_j + C_j + F_{j+1/2}).$$

We conclude that all eigenvectors and eigenvalues of S_3 can be given explicitly, by elementary means, and this completes our outline of the proof of Lemma $5.5.1_{/226}$.

Except for the dominant eigenvalue $\lambda_1 = 1$, the calculation of the eigenvectors is tedious and the resulting expressions are complicated. We therefore do not give the complete result, but instead restrict ourselves to describing only the main structure of the analysis.

5.5.3 Convergence

We have the following theorem for convergence at a nonregular point.

Theorem 5.5.2. *For all choices of an initial sequence p_l^0 of control vectors, the resulting surface representation*

$$x(y) = \sum_l p_l^0 N_l(y)$$

is continuous at the origin. Moreover, the subdivision process has local uniform convergence, in the following sense (see Definition $4.7.1_{/182}$). For every compact (i.e., bounded and closed) set $A \subset \mathbb{R}^2$ and every $\varepsilon > 0$ there exists a ν_0 such that

$$\nu \geq \nu_0 \text{ and } l/2^\nu \in A \text{ implies } |x(l/2^\nu) - p_l^\nu| < \varepsilon.$$

Proof. We already know from the results of Section 5.2 that $x(y)$ is continuous outside the origin, so we only have to verify continuity at the origin. From the analysis in the previous section the eigenvectors $\{\xi^j\}_{j=1}^{6n+1}$ of S_2 form a basis for $\mathbb{R}^{6n+1}$ with corresponding eigenvalues $\{\lambda_j\}_{j=1}^{6n+1}$. We then have $S_2\xi^j = \lambda_j\xi^j$ and we

can expand the vector $p^{0,2}$ (see $(5.64)_{/225}$) as

$$p^{0,2} = \sum_{j=1}^{6n+1} \xi^j c_j.$$

Since $p^{0,2}$ has coordinates which themselves are vectors in $\mathbb{R}^N$, the coefficients c_j are also vectors in $\mathbb{R}^N$. We now get, since $\lambda_1 = 1$ and $|\lambda_j| < 1$ for $2 \le j \le 6n+1$,

$$p^{\nu,2} = S_2^\nu p^{0,2} = \sum_j \lambda_j^\nu \xi^j c_j \to \xi^1 c_1 \quad \text{as} \quad \nu \to \infty,$$

and consequently

$$p_l^{\nu,2} \to \xi_l^1 c_1 = c_1$$

for all $l \in \mathbb{G}_2$ as $\nu \to \infty$.

Now let $\varepsilon > 0$ be given and choose ν such that $|p_l^{\nu,2} - c_1| < \varepsilon$ for all $l \in \mathbb{G}_2$. Then choose $\delta > 0$ such that $\{y \in \mathbb{R}^2 : |y| < \delta\} \subset \bar{\mathbb{G}}_1/2^\nu$. Then, if $0 < |y| < \delta$,

$$x(y) = \sum_{l \in \mathbb{G}} p_l^\nu N_l^\nu(y) = \sum_{l \in \mathbb{G}_2} p_l^{\nu,2} N_l^\nu(y),$$

where N_l^ν denotes the nodal basis function at the ν th step of the process, centered around the point $l/2^\nu$ in parameter space. Using that $N_l^\nu(y) \ge 0$ for all l and that $\sum_{l \in \mathbb{G}_2} N_l^\nu(y) = 1$, we get

$$x(y) - c_1 = \sum_{l \in \mathbb{G}_2} (p_l^{\nu,2} - c_1) N_l^\nu(y)$$

and

$$|x(y) - c_1| \le \sum_{l \in \mathbb{G}_2} |p_l^{\nu,2} - c_1)| N_l^\nu(y) \le \varepsilon \sum_{l \in \mathbb{G}_2} N_l^\nu(y) = \varepsilon.$$

We have shown that for an arbitrary $\varepsilon > 0$ there exists a $\delta > 0$ such that

$$0 < |y| < \delta \Longrightarrow |x(y) - c_1| < \varepsilon.$$

This proves that $x(y)$ is continuous at the origin if we define $x(0) = c_1$. We now prove the local uniform convergence of the process. Let $\varepsilon > 0$ be given. Choose ν_1 such that

$$\nu \ge \nu_1 \Longrightarrow |p_l^{\nu,2} - c_1| < \varepsilon/2,$$

and then choose $\delta > 0$ such that $\{y : |y| < \delta\} \subset \bar{\mathbb{G}}_1/2^{\nu_1}$. Next choose $\nu_0 \ge \nu_1$ such that

$$\left.\begin{array}{l} |l/2^\nu| \ge \delta,\ l/2^\nu \in A \\ \nu \ge \nu_0 \end{array}\right\} \Longrightarrow |x(l/2^\nu) - p_l^\nu| < \varepsilon/2.$$

This is possible, since we know that the process has local uniform convergence outside the set $\{y : |y| < \delta\}$.

We next observe that all control vectors p_l^ν for $\nu \geq \nu_1$ and $|l/2^\nu| < \delta$ are convex combinations of $p_l^{\nu_1}$, $l \in \mathbb{G}_2$. This follows from the subdivision rules $(5.73)_{/226}$ where new values are convex combinations of old ones. Therefore,

$$\left.\begin{array}{l} |p_l^{\nu_1} - c_1| < \varepsilon/2 \\ l \in \mathbb{G}_2 \end{array}\right\} \Longrightarrow |p_l^\nu - c_1| < \varepsilon/2 \ \text{ for all } l \text{ with } |l/2^\nu| < \delta \text{ and all } \nu \geq \nu_1.$$

We conclude that

$$|x(l/2^\nu) - p_l^\nu| = |(x(l/2^\nu) - c_1) - (p_l^\nu - c_1)| \leq \varepsilon/2 + \varepsilon/2 = \varepsilon$$

whenever $|l/2^\nu| < \delta$ and $\nu \geq \nu_0$. This completes the proof. □

Note that the preceding proof makes use of spectral structure of the matrix S_2 only, and no further information regarding S_3 is needed.

A similar convergence result can be proved in a much more general setting, namely for any subdivision method where the restriction to the regular case is known to converge uniformly, where the values in $\bar{\mathbb{G}}_1 \setminus \{0\}$ are determined by the control vectors $\{p_l^\nu\}$ with indices in the k-ring $\mathbb{G}_k$ and where the eigenvalues satisfy $1 = \lambda_1 > |\lambda_j|$, $j = 2, 3, \ldots$. The particular valence n and the value of k determine the size of the relevant local subdivision matrix.

We also make the following remark. Assume that we are given a subdivision method that is a variant of some basic method, as in Figure $1.30_{/33}$, and that it is in addition affine invariant and stationary. Then it can be shown that it is necessary for convergence that $\lambda_1 = 1$ is an eigenvalue of algebraic multiplicity one for all local subdivision matrices (with the eigenvector $(1, 1, \ldots, 1)^t$) and that $|\lambda_j| < 1$ for all other eigenvalues. See Exercise $8_{/245}$.

5.6 Smoothness analysis for the Catmull–Clark scheme

First we introduce the concept of tangent-plane continuity for a parametric surface. This property was first studied in the context of subdivision surfaces in [8, 9]. Assume that we have a locally uniformly convergent subdivision method with nodal functions that are continuous at the nonregular point and at least C^1 outside. As before we let the nonregular point be the origin.

Definition 5.6.1. *The subdivision surface is said to be tangent-plane continuous at the origin if for the normal vector $n(y)$ we have $n(y) \neq 0$ for $0 < |y| \leq \delta$, for some $\delta > 0$, and if the limit*

$$\lim_{y \to 0} n(y)/|n(y)| = \hat{n}(0)$$

exists.

Tangent-plane continuity does not necessarily mean that the surface is well behaved in a neighbourhood of the origin. We also need conditions which guarantee

that the mapping

$$\mathbb{R}^2 \ni y \mapsto x(y) \in \mathbb{R}^3$$

is one-to-one for $0 < |y| \leq \delta$; otherwise the phenomenon of multiple sheetedness might appear. For example, the surface in Figure $5.2_{/222}$ is tangent-plane continuous. Conditions for single sheetedness are discussed below in Section 5.7.

For the analysis of tangent-plane continuity for the Catmull–Clark method we need certain spectral properties of the matrix S_3. In particular, we require explicit expressions for two linearly independent eigenvectors $\xi^2 = \{\xi^2_l\}_{l\in\mathbb{G}_3}$ and $\xi^3 = \{\xi^3_l\}_{l\in\mathbb{G}_3}$ for the subdominant double eigenvalue $\lambda_2 = \lambda_3 < \lambda_1 = 1$. We have the following result.

Theorem 5.6.2. *For almost all choices of an initial sequence $p^0_{l\in\mathbb{G}}$ (in a sense to be specified below in the proof) we have tangent-plane continuity at the origin.*

Proof. A given initial subcontrol vector $p^{0,3} \in \mathbb{R}^{N(12n+1)}$ can be expanded as

$$p^{0,3} = \sum_{j=1}^{12n+1} \xi^j c_j, \tag{5.91}$$

where $S_3\xi^j = \lambda_j \xi^j$. We now assume that the vector-valued coefficients c_2 and c_3 are linearly independent. This is the exact meaning of "for almost all choices of an initial sequence" in the statement of the theorem. It is straightforward to prove that the exceptional set $M \subset \mathbb{R}^{N(12n+1)}$, defined by the condition that the coefficients c_2 and c_3 in the expansion $(5.91)_{/235}$ are linearly dependent, has the following property. If $p^{0,3} \in M$ with $c_2 \neq 0$ and $c_3 \neq 0$, then there exists a neighbourhood of $p^{0,3}$ in M which is a nonlinear manifold of dimension $N(12n+1)$. In particular, this implies that the Lebesgue measure of $M \subset \mathbb{R}^{N(12n+1)}$ is zero.

In order to prove tangent-plane continuity we must show the following. There exists a unit vector $\hat{n}(0)$ such that, for any given $\varepsilon > 0$, there exists a $\delta > 0$ such that

$$0 < |y| < \delta \Longrightarrow \left| \hat{n}(0) - \frac{n(y)}{|n(y)|} \right| < \varepsilon.$$

As shown below, if $N = 3$, the limiting normal vector is $\hat{n} = \pm(c_2 \times c_3)/|c_2 \times c_3|$. Here $\times$ denotes the vector cross product in $\mathbb{R}^3$.

After the ν th step of the subdivision, we have

$$x(y) = \sum_{l\in\mathbb{G}_3} p^{\nu,3}_l N^\nu_l(y) \tag{5.92}$$

if $y \in 2^{-\nu}\bar{\mathbb{G}}_2$, and in particular if $y \in 2^{-\nu}(\bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1)$. Next,

$$\begin{aligned} p^{\nu,3} = S^\nu_3 p^{0,3} &= \sum_{j=1}^{12n+1} \lambda^\nu_j \xi^j c_j \\ &= \xi^1 c_1 + \lambda^\nu_2 \left\{ \xi^2 c_2 + \xi^3 c_3 + \sum_{j=4}^{12n+1} (\lambda_j/\lambda_2)^\nu \xi^j c_j \right\}. \end{aligned}$$

Inserting into $(5.92)_{/235}$ we get

$$x(y) = \sum_{l\in\mathbb{G}_3} p_l^{\nu,3} N_l^\nu(y) = c_1 \sum_{l\in\mathbb{G}_3} \xi_l^1 N_l^\nu(y) + \lambda_2^\nu \left\{ c_2 \sum_{l\in\mathbb{G}_3} \xi_l^2 N_l^\nu(y) + c_3 \sum_{l\in\mathbb{G}_3} \xi_l^3 N_l^\nu(y) + \sum_{j=4}^{12n+1} (\lambda_j/\lambda_2)^\nu c_j \sum_{l\in\mathbb{G}_3} \xi_l^j N_l^\nu(y) \right\}.$$

Here the first term in the right-hand side simplifies to the constant c_1 since $\xi_l^1 = 1$ and $\sum_{l\in\mathbb{G}_3} N_l^\nu(y) = 1$.

Using that $N_l^\nu(y) = N_l(2^\nu y) = N_l^0(2^\nu y)$, where $N_l = N_l^0$ denotes a nodal function on the initial grid $\mathbb{G}$, after changing scale we get

$$\begin{aligned} x(2^{-\nu}y) &= \sum_{l\in\mathbb{G}_3} p_l^{\nu,3} N_l(y) = c_1 \\ &\quad + \lambda_2^\nu \left\{ c_2 \sum_{l\in\mathbb{G}_3} \xi_l^2 N_l(y) + c_3 \sum_{l\in\mathbb{G}_3} \xi_l^3 N_l(y) + \sum_{j=4}^{12n+1} (\lambda_j/\lambda_2)^\nu c_j \sum_{l\in\mathbb{G}_3} \xi_l^j N_l(y) \right\} \\ &= c_1 + \lambda_2^\nu \left\{ X(y) + \alpha_\nu(y) \right\}, \end{aligned} \tag{5.93}$$

where the part

$$X(y) = c_2 \sum_{l\in\mathbb{G}_3} \xi_l^2 N_l(y) + c_3 \sum_{l\in\mathbb{G}_3} \xi_l^3 N_l(y)$$

of this expression[41] defines a surface patch in $\mathbb{R}^N$ for $y \in \bar{\mathbb{G}}_2$.

(If the origin is a regular point with valence 4, then ξ_l^2 and ξ_l^3 are linear functions of l, while $\sum_{l\in\mathbb{G}_3} \xi_l^2 N_l(y)$ and $\sum_{l\in\mathbb{G}_3} \xi_l^3 N_l(y)$ are linear functions of y interpolating the functions ξ_l^2 and ξ_l^3, i.e., $\sum_{l\in\mathbb{G}_3} \xi_l^2 N_l(k) = \xi_k^2$ and $\sum_{l\in\mathbb{G}_3} \xi_l^3 N_l(k) = \xi_k^3$ for all $k \in \mathbb{Z}^2$. Consequently, in this case $X(y)$ is a linear function of $y \in \mathbb{R}^2$ with values in the plane spanned by c_2 and c_3. Properties such as these, of polynomial reproduction, are investigated in Section 6.4.)

Continuing the proof, we now formulate the following lemma.

Lemma 5.6.3. *Let $N = 3$. Then the surface patch defined by*

$$\bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1 \ni y = (u, v) \mapsto X(u, v) \in \mathbb{R}^N$$

has a normal of the form $\frac{\partial X}{\partial u} \times \frac{\partial X}{\partial v} = n_0(y)(c_2 \times c_3)$ with $n_0(y)$ continuous and either $n_0(y) \geq c > 0$ or $n_0(y) \leq -c < 0$ for some constant c. If $N = 2$, the conclusion is instead the following. Let $X = (X_1, X_2) \in \mathbb{R}^2$. Then

$$\pm \left(\frac{\partial X_1}{\partial u} \frac{\partial X_2}{\partial v} - \frac{\partial X_1}{\partial v} \frac{\partial X_2}{\partial u} \right) \geq c > 0$$

for some constant c and for some choice of sign (the same choice for all y).

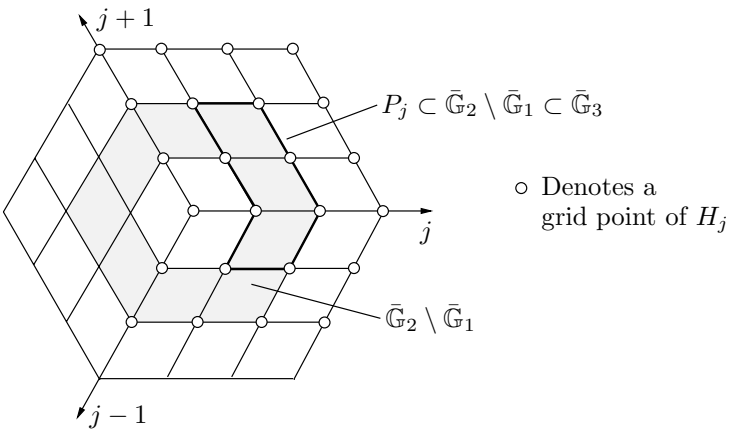


Figure 5.4. *The subpatch P_j.*

Proof. We outline the main ideas of a proof without giving all the details, and for $N = 3$ only. Consider the subpatch given by the subset $P_j \subset \bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1$ defined by

$$P_j = \{y \in \bar{\mathbb{G}}_3 : 0 \le \varepsilon_{j-1} \le 1,\ 1 \le \varepsilon_j \le 2,\ 0 \le \varepsilon_{j+1} \le 2,\ \varepsilon_k = 0,\ \text{if } |k-j| > 1\}$$

in parameter space; see Figure 5.4$_{/237}$. Due to the symmetry, it suffices to prove the statement about $n_0(y)$ when $y \in P_j$ for a single value of j. We have

$$X(y) = c_1 \sum_{l \in H_j} \xi_l^2 N_l(y) + c_3 \sum_{l \in H_j} \xi_l^3 N_l(y),$$

where

$$H_j = \{l \in \mathbb{G}_3 : \varepsilon_{j-1} = 0,1,2,\ \varepsilon_j = 0,1,2,3,\ \varepsilon_{j+1} = 0,1,2,3\}.$$

After a piecewise-affine transformation $\mathcal{A}$ of parameter space defined by letting $\mathcal{A}(e_j) = (1,0)^t$, $\mathcal{A}(e_{j+1}) = (0,1)^t$, and $\mathcal{A}(e_{j-1}) = -(1,0)^t$, the sets P_j and H_j are mapped onto sets

$$\mathcal{A}(P_j) = P' = \{y = (u,v) \in \mathbb{R}^2 : -1 \le u \le 2,\ 1 \le v \le 2\}$$

and

$$\mathcal{A}(H_j) = H' = \{l = (l_1, l_2) \in \mathbb{Z}^2 : -2 \le l_1 \le 3,\ 0 \le l_2 \le 3\};$$

see Figure 5.5$_{/238}$.

If $l' = \mathcal{A}(l) \in H'$, we define $\xi_{l'}^2$ by $\xi_{l'}^2 \doteq \xi_l^2$ and similarly for $\xi_{l'}^3$. Then in the new local coordinates $y = (u,v) \in P'$, we have the following representation of the surface:

$$X(u,v) = c_2 \sum_{l \in H'} \xi_l^2 N_{l_1}^3(u) N_{l_2}^3(v) + c_3 \sum_{l \in H'} \xi_l^3 N_{l_1}^3(u) N_{l_2}^3(v), \tag{5.94}$$

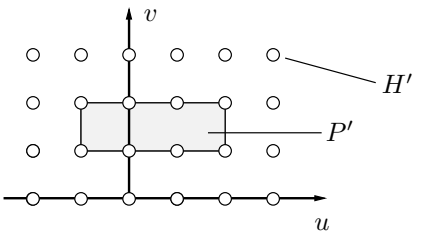


Figure 5.5. *Parameter space after piecewise-affine transformation.*

where $N^3(u)$ and $N^3(v)$ are third-degree univariate nodal spline functions centered at the origin. The expression in $(5.94)_{/237}$ is a uniform third-degree tensor-product representation of the surface patch.

Since the coefficients ξ_l^2 and ξ_l^3 in $(5.94)_{/237}$ are known, the normal $\frac{\partial X}{\partial u} \times \frac{\partial X}{\partial v}$ can be explicitly calculated. We get, using $(2.61)_{/81}$ and $(2.62)_{/82}$,

$$\frac{\partial X(u,v)}{\partial u} = c_2 \sum_l \Delta_1 \xi_l^2 N_{l_1}^2(u) N_{l_2}^3(v) + c_3 \sum_l \Delta_1 \xi_l^3 N_{l_1}^2(u) N_{l_2}^3(v),$$

where $\Delta_1 \xi_l^2 = \xi_{l+(1/2,0)}^2 - \xi_{l-(1/2,0)}^2$ and the summation is over those l for which $l \pm (1/2, 0) \in H'$. Similarly, we get

$$\frac{\partial X(u,v)}{\partial v} = c_2 \sum_l \Delta_2 \xi_l^2 N_{l_1}^3(u) N_{l_2}^2(v) + c_3 \sum_l \Delta_2 \xi_l^3 N_{l_1}^3(u) N_{l_2}^2(v),$$

where $\Delta_2 \xi_l^2 = \xi_{l+(0,1/2)}^2 - \xi_{l-(0,1/2)}^2$ is a central-difference operator and the summation is over those l for which $l \pm (0, 1/2) \in H'$. Now, using that $c_2 \times c_2 = c_3 \times c_3 = 0$ and $c_3 \times c_2 = -c_2 \times c_3$ we get

$$\begin{aligned} &\frac{\partial X(u,v)}{\partial u} \times \frac{\partial X(u,v)}{\partial v} \\ &= (c_2 \times c_3) \sum_{l,k} (\Delta_1 \xi_l^2 \Delta_2 \xi_k^3 - \Delta_1 \xi_l^3 \Delta_2 \xi_k^2) N_{l_1}^2(u) N_{k_1}^3(u) N_{l_2}^3(v) N_{k_2}^2(v). \end{aligned} \tag{5.95}$$

Since the nodal functions in the sum are nonnegative and since

$$\sum_{k,l} N_{l_1}^2(u) N_{k_1}^3(u) N_{l_2}^3(v) N_{k_2}^2(v) = 1,$$

the right-hand side of $(5.95)_{/238}$ is a convex combination of the real numbers $\Delta_1 \xi_l^2 \Delta_1 \xi_k^3 - \Delta_1 \xi_l^3 \Delta_2 \xi_k^2$. In order to prove the statement of the lemma, it suffices to show that either $\Delta_1 \xi_l^2 \Delta_2 \xi_k^3 - \Delta_1 \xi_l^3 \Delta_2 \xi_k^2 \geq c > 0$ for some c or that $\Delta_1 \xi_l^2 \Delta_2 \xi_k^3 - \Delta_1 \xi_l^3 \Delta_2 \xi_k^2 \leq -c < 0$. In fact, after fairly tedious calculations this can be verified, but we omit these calculations. Again, for a regular point, the expressions $\Delta_1 \xi_l^2 \Delta_2 \xi_k^3 - \Delta_1 \xi_l^3 \Delta_2 \xi_k^2$ are constant, i.e., independent of k and l, and the factor $n_0(y)$ is constant.

This completes the proof of the lemma. □

Now returning to the expression $(5.93)_{/236}$ with y replaced by $2^\nu y$, we have

$$x(y) = c_1 + \lambda_2^\nu \left\{ X(2^\nu y) + \sum_{j=4}^{12n+1} (\lambda_j/\lambda_2)^\nu c_j \sum_{l\in\mathbb{G}_3} \xi_l^j N_l(2^\nu y) \right\} \tag{5.96}$$

for $y \in 2^{-\nu}\bar{\mathbb{G}}_2$. Differentiating we get

$$\begin{aligned}\frac{\partial x(u,v)}{\partial u} &= (2\lambda_2)^\nu \left\{ \left(\frac{\partial X}{\partial u}\right)(2^\nu y) + \sum_{j=4}^{12n+1} (\lambda_j/\lambda_2)^\nu c_j \sum_{l\in\mathbb{G}_3} \xi_l^j \left(\frac{\partial N_l}{\partial u}\right)(2^\nu y) \right\} \\ &= (2\lambda_2)^\nu \left\{ \left(\frac{\partial X}{\partial u}\right)(2^\nu y) + \gamma_\nu(2^\nu y) \right\}\end{aligned}$$

(defining γ_ν), where, if $y \in 2^{-\nu}(\bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1)$, then $|\gamma_\nu| < C(|\lambda_4|/\lambda_2)^\nu$ for some constant C. We also have a similar expression for $\frac{\partial x(u,v)}{\partial v}$ and we obtain, for the surface normal,

$$\begin{aligned} n(y) = \frac{\partial x}{\partial u} \times \frac{\partial x}{\partial v} &= (2\lambda_2)^{2\nu} \left\{ \left(\frac{\partial X}{\partial u}\right)(2^\nu y) \times \left(\frac{\partial X}{\partial v}\right)(2^\nu y) + \delta_\nu(2^\nu y) \right\} \\ &= (2\lambda_2)^{2\nu} \Big\{ n_0(2^\nu y)(c_2 \times c_3) + \delta_\nu(2^\nu y) \Big\}, \end{aligned} \tag{5.97}$$

where $|\delta_\nu| < C(|\lambda_4|/\lambda_2)^\nu$ for some constant C, provided that $y \in 2^{-\nu}(\bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1)$. Without loss of generality, we may assume that $n_0(y) > 0$, so that $n_0(2^\nu y) \geq c > 0$ if $y \in 2^{-\nu}(\bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1)$. Then, if $\varepsilon > 0$ is given, we can choose ν_0 so large that

$$\left| \frac{n(y)}{|n(y)|} - \frac{c_2 \times c_3}{|c_2 \times c_3|} \right| < \varepsilon$$

if $\nu \geq \nu_0$ and $y \in 2^{-\nu}(\bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1)$.

Next choose $\delta > 0$ so that $\{y : |y| < \delta\} \subset 2^{-\nu_0}\bar{\mathbb{G}}_2$. Then, if y is given with $0 < |y| < \delta$, there exists a unique $\nu \geq \nu_0$ such that $y \in 2^{-\nu}(\bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1)$. Consequently,

$$0 < |y| < \delta \Longrightarrow \left| \frac{n(y)}{|n(y)|} - \frac{c_2 \times c_3}{|c_2 \times c_3|} \right| < \varepsilon.$$

This completes the proof of the theorem. □

5.7 Conditions for single sheetedness

In order to prove that the subdivision method produces a locally well-defined surface in a neighbourhood $U \ni y_0$ of a point y_0 in parameter space, one must verify that the mapping

$$U \ni y \mapsto x(y) \in \mathbb{R}^N$$

is one-to-one. This is normally done by showing that the functional matrix $\left[\frac{\partial x}{\partial u}, \frac{\partial x}{\partial v}\right]$ is continuous and of full rank at y_0 or, equivalently, that the normal

vector $n(y) = \frac{\partial x}{\partial u} \times \frac{\partial x}{\partial v}$ is continuous and different from the null vector at y_0. At points in parameter space corresponding to a regular vertex this is true for almost all choices of initial control vector sequences. The exceptional sequences are in a lower-dimensional submanifold of the space of all possible choices, similarly to the situation described in the proof of Theorem $5.6.2_{/235}$. Having established that the mapping is one-to-one, it also follows that after introducing a new local Euclidean coordinate system with the x_3-axis parallel to the normal $n(y_0)$ and the origin at $x(y_0)$, the surface has the equation $x_3 = f(x_1, x_2)$, where f is C^1-continuous in a neighbourhood of the origin.

In the nonregular case this approach is not possible, even if we have been able to establish tangent-plane continuity. This follows from $(5.97)_{/239}$ where $y \in 2^{-\nu}(\bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1)$. If we let $\nu \to \infty$, we see that $\lim_{y\to 0} |n(y)| = 0$ or $\lim_{y\to 0} |n(y)| = \infty$ unless $\lambda_2 = 1/2$, in which case we have a regular point.

Instead we will rely on some results of functional analysis proved by using the concept of topological degree, or winding number. The results needed will be described below, but not their full proofs. The underlying theory of topological degree and winding number is extensively analysed in [41, 70]. The necessary tools for a proof of Theorem $5.7.1_{/241}$ below can also be found in [4] and in [124, Sec. 2.3].

Let the open sets Ω_j, $j = 0, 1, \ldots, n-1$, be defined by

$$\Omega_j = \{y : y = \varepsilon_j e_j + \varepsilon_{j+1} e_{j+1}, 0 < \varepsilon_j, \varepsilon_{j+1} < 2\},$$

so that

$$\bar{\mathbb{G}}_2 = \bigcup_{j=0}^{n-1} \bar{\Omega}_j,$$

where $\bar{\Omega}_j$ denotes the topological closure of Ω_j. Then the vector-valued function $x(y)$ is piecewise C^1-continuous over this partitioning of $\bar{\mathbb{G}}_2$ in the sense that the restrictions $x(y)|_{\Omega_j}$, $\frac{\partial x}{\partial u}|_{\Omega_j}$, and $\frac{\partial x}{\partial v}|_{\Omega_j}$ are continuous and can be extended to continuous functions on $\bar{\Omega}_j$ for all j (the extensions need not coincide on intersections $\bar{\Omega}_j \cap \bar{\Omega}_{j-1}$ of adjacent subpatches). Further, we can perform a piecewise-affine change of variables on the union $\bar{\Omega}_j \cup \bar{\Omega}_{j-1}$ so that $x(y)$ becomes C^1-continuous in the interior of this union.

We begin by giving some more detailed information about the asymptotic behaviour of $n(y)$. From $(5.97)_{/239}$ it follows that the component of $n(y)$ which is parallel to $c_2 \times c_3$ is the dominant one as $\nu \to \infty$. We may estimate the order of magnitude for this component when $|y| \to 0$ in the following way. If $y \in \bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1$, then $|y| \sim 2^{-\nu}$, i.e., $\nu \sim -\lg y$ and

$$(2\lambda_2)^{2\nu} = 2^{2\nu \lg(2\lambda_2)} \sim 2^{-2 \lg y \lg(2\lambda_2)} = |y|^{\gamma},$$

where $\gamma = -2\lg(2\lambda_2)$, and $\lg(\cdot)$ denotes the logarithm base 2. For $n = 3$, $2\lambda_2 > 1$ and $\gamma < 0$. For $n > 4$, $2\lambda_2 < 1$ and $\gamma > 0$. For a regular point with $n = 4$ we have $2\lambda_2 = 1$ and $\gamma = 0$. So, for $n = 3$, $|n(y)| \to \infty$ and for $n > 4$, $|n(y)| \to 0$ as $|y| \to 0$.

Next, we introduce a new Euclidean coordinate system $x_1, \ldots, x_N$ in $\mathbb{R}^N$ with the origin at the point $x(0) = c_1$ and, if $N = 3$, with the x_3-axis parallel to the

vector $c_2 \times c_3$. In these new coordinates the surface has the representation (if $N = 3$)

$$\mathbb{R}^2 \ni y \mapsto (x_1(y), x_2(y), x_3(y)) \in \mathbb{R}^3.$$

The $c_2 \times c_3$ component of the surface normal is now given by the functional determinant

$$n_3(y) = \frac{d(x_1(u,v), x_2(u,v))}{d(u,v)} = \begin{vmatrix} \frac{\partial x_1}{\partial u} & \frac{\partial x_1}{\partial v} \\ \frac{\partial x_2}{\partial u} & \frac{\partial x_2}{\partial v} \end{vmatrix}.$$

Consider the mapping

$$\mathbb{R}^2 \ni y = (u,v) \mapsto (x_1(u,v), x_2(u,v)) = x^p(y) \in \mathbb{R}^2,$$

where the notation x^p is intended to suggest "projection" into the (x_1, x_2)-plane. It has been proved above that if ν is large enough, then there exist positive constants c and C such that

$$c|y|^\gamma < \frac{d(x_1, x_2)}{d(u,v)} < C|y|^\gamma$$

for all $y \in 2^{-\nu}\bar{\mathbb{G}}_2 \setminus \{0\}$.

We now formulate the mentioned topological uniqueness theorem needed for establishing single sheetedness. Let ∂ denote the boundary of a subset in $\mathbb{R}^2$.

Theorem 5.7.1. *Let $x : \bar{\mathbb{G}}_2 \to \mathbb{R}^2$ be a mapping which is piecewise C^1-continuous over the partition $\bar{\mathbb{G}}_2 = \bigcup_{j=1}^n \bar{\Omega}_j$ as described above. Assume also that*

$$\frac{d(x_1, x_2)}{d(u,v)}\Big|_{\bar{\Omega}_j \setminus \{0\}} > 0 \quad \textit{for all } j.$$

Then the mapping is one-to-one if and only if the restriction $x|_{\partial\bar{\mathbb{G}}_2}$ to the boundary is one-to-one.

Next, we give the main theorem guaranteeing single sheetedness (almost everywhere) around a nonregular point, for the Catmull–Clark scheme.

Theorem 5.7.2. *If the vector $c_2 \times c_3$ is different from the null vector, then there exists a $\delta > 0$ such that the mapping $\mathbb{R}^2 \ni y \mapsto x(y) = (x^p(y), x_3(y)) \in R^3$ is one-to-one for $|y| < \delta$. Moreover, the surface has a representation $x_3 = f(x_1, x_2)$ in a neighbourhood of the origin, with f a C^1-continuous function of (x_1, x_2).*

Proof. By Theorem 5.7.1$_{/241}$ it suffices to prove that, for some ν the mapping $x|_{2^{-\nu}\partial\bar{\mathbb{G}}_2}$ is one-to-one. Having done this we choose $\delta > 0$ such that $\{y : |y| < \delta\} \subset 2^{-\nu}\partial\bar{\mathbb{G}}_2$.

By (5.93)$_{/236}$ we have (since $c_1 = 0$ is now the origin)

$$x^p(2^{-\nu}y) = \lambda_2^\nu \{X(y) + \alpha_\nu^p(y)\}$$

for all $y \in \bar{\mathbb{G}}_2$, where $\alpha_\nu^p = (\alpha_{\nu,1}, \alpha_{\nu,2})$ is the (x_1, x_2)-component of the vector α_ν. On $\partial\bar{\mathbb{G}}_2$, the functions X and α_ν^p represent third-degree spline curves in $\mathbb{R}^2$ with control polygons defined by ξ_l^i, $l \in \mathbb{G}_3 \setminus \{0\}$ and $i = 2, 3$ for X and $4 \le i \le n$ for α_ν.

We have the following lemma.

Lemma 5.7.3. *If $X|_{\partial\bar{\mathbb{G}}_2}$ is one-to-one and if the derivative of $X|_{\partial\bar{\mathbb{G}}_2}$ along $\partial\bar{\mathbb{G}}_2$ does not vanish at any point, then, if ν is chosen large enough, $(X + \alpha_\nu^p)|_{\partial\bar{\mathbb{G}}_2}$ is one-to-one.*

Proof. Let

$$\epsilon(\delta) = \min\{|X(y_1) - X(y_2)| : y_1, y_2 \in \partial\bar{\mathbb{G}}_2, |y_1 - y_2| \ge \delta\}.$$

Since X is continuous and $\partial\bar{\mathbb{G}}_2$ is compact, $\epsilon(\delta) > 0$ for all $\delta > 0$. Now $X|_{\partial\bar{\mathbb{G}}_2}$ is a piecewise third-degree polynomial curve along $\partial\bar{\mathbb{G}}_2$, and by assumption the derivative along $\partial\bar{\mathbb{G}}_2$ does not vanish anywhere. Therefore we have, for some $\delta > 0$ and $K_1 > 0$, that

$$y_1, y_2 \in \partial\bar{\mathbb{G}}_2, \quad |y_1 - y_2| \le \delta \Longrightarrow |X(y_1) - X(y_2)| \ge K_1|y_1 - y_2|.$$

Further, let $C_1 = \max\{|y_1 - y_2| : y_1, y_2 \in \partial\bar{\mathbb{G}}_2\}$. Then

$$\begin{aligned} &y_1, y_2 \in \partial\bar{\mathbb{G}}_2, \quad \delta \le |y_1 - y_2| \le C_1 \\ &\Longrightarrow |X(y_1) - X(y_2)| \ge \varepsilon(\delta) = (\varepsilon(\delta)/C_1)C_1 \ge (\varepsilon(\delta)/C_1)|y_1 - y_2|. \end{aligned}$$

Consequently, with $c = \min\{\varepsilon/C_1, K_1\}$ we have $|X(y_1) - X(y_2)| \ge c|y_1 - y_2|$. Similarly, $\alpha_\nu^p|_{\partial\bar{\mathbb{G}}_2}$ is a piecewise third-degree polynomial curve. The spline coefficients of this curve are majorized by a factor $(|\lambda_4|/\lambda_2)^{\nu_0}$ (see $(5.96)_{/239}$) and therefore we have

$$|\alpha_\nu^p(y_1) - \alpha_\nu^p(y_2)| \le C(|\lambda_4|/\lambda_2)^\nu |y_1 - y_2|$$

for some constant C. It follows that

$$|X(y_1) + \alpha_{\nu_0}^p(y_1) - (X(y_2) + \alpha_{\nu_0}^p(y_2))| \ge (c - C(|\lambda_4|/\lambda_2)^{\nu_0})|y_1 - y_2|.$$

If we choose ν_0 so large that the constant in the right-hand side is positive, then $(X + \alpha_{\nu_0})|_{\partial\bar{\mathbb{G}}_2}$ is one-to-one. □

In order to complete the proof of Theorem $5.7.2_{/241}$, it suffices to show that $X|_{\partial\bar{\mathbb{G}}_2}$ is one-to-one and has nonvanishing derivative everywhere. This requires a detailed analysis of this piecewise third-degree spline curve in terms of the coefficients ξ_l^2 and ξ_l^3, $l \in \mathbb{G}_3 \setminus \{0\}$, which is omitted. We refer instead to Chapters 5 and 6 of [124].

We have proved that the mapping

$$2^{-\nu}\bar{\mathbb{G}}_2 \ni y = (u, v) \mapsto x^p(y) = (x_1(u, v), x_2(u, v)) \in \mathbb{R}^2$$

is one-to-one if ν is large enough. It follows that the inverse mapping

$$(u, v) = (u(x_1, x_2), v(x_1, x_2))$$

is uniquely defined on the image of $2^{-\nu}\bar{\mathbb{G}}_2$. Therefore, the component $x_3(u, v)$ is a well-defined function

$$x_3 = f(x_2, x_3)$$

in a neighbourhood of the origin. The surface unit normal is given by

$$\hat{n}(y) = (-f_{x_1}, -f_{x_2}, 1)/\sqrt{1 + f_{x_1}^2 + f_{x_2}^2}$$

in these coordinates. From the fact that $n(y)/|n(y)|$ is continuous in a neighbourhood of the origin, it follows that f_{x_1} and f_{x_2} are continuous functions of (x_1, x_2) in a neighbourhood of the origin and that $f_{x_1}(x_1, x_2)$ and $f_{x_2}(x_1, x_2)$ tend to 0 as $(x_1, x_2) \to (0, 0)$.

The mapping

$$\bar{\mathbb{G}}_2 \ni y = (u, v) \mapsto X(u, v) \in \mathbb{R}^2$$

is called the *characteristic map* of the subdivision method. The analysis above shows that the invertibility of the characteristic map guarantees single sheetedness and C^1-continuity of the resulting surface for almost all choices of initial control sequences. □

5.8 Further reading on convergence and smoothness

The topic of convergence and smoothness of subdivision methods has been a subject of intense research effort in recent years. The sections above gave the basic ideas on the subject, including many facts that are often left unstated in the literature. On the other hand, these sections gave *only* the basic ideas. Important references for further reading include the thesis of Zorin [172], as well as related papers such as [173, 174], and the book of Peters and Reif [124].

In this book we have adopted a point of view that is standard in computer science for the study of subdivision surfaces: subdivision is viewed as a set of operations applied to a two-dimensional cellular complex, which is essentially a graph structure with the additional specification of which edges are connected to which faces [10, 11]. In [124], however, there is a shift in point of view: subdivision is viewed as generating a sequence of nested annular rings of surfaces, whose union forms a generalized spline [124, Sec. 3.1, Sec. 4.4]. This change in point of view permits application in a natural way of many concepts from differential geometry and leads to an elegant presentation of the main known results about convergence and smoothness of subdivision methods.

Shape and the spectral properties of subdivision matrices

It has been understood since the earliest modern papers on subdivision that the distribution of the eigenvalues of local subdivision matrices, as determined by the

choice of weights in stencils for nonregular points, is important in determining the shape and smoothness properties of the limit surfaces. This is particularly evident in the paper of Doo and Sabin [45], and in the choice of weights using empirical criteria in the paper of Catmull and Clark [24]. Understanding of these issues increased significantly over the three decades following these papers. Here we give a brief summary, based on the two principal references mentioned above.

It was shown earlier in the chapter, for the example of the Catmull–Clark method, that C^1-continuity of the limit surface can be obtained by establishing regularity conditions (Lemma $5.6.3_{/236}$) and invertibility conditions (Lemma $5.7.3_{/242}$) on the characteristic map. In [124, 172], these questions are studied in a general context. What are necessary and sufficient conditions, for various *classes* of methods, in order to achieve C^1-continuity? What are the conditions for C^2-continuity (curvature)? What is the general relationship between the spectral properties of the subdivision matrices and the shape of the limit surfaces?

In Chapters 4 and 5 of [172] algorithms are given to test whether regularity and injectivity of the characteristic map are satisfied for certain general classes of methods, and a study of specific schemes, including the Loop, Butterfly, Catmull–Clark, and Doo–Sabin methods, is given in [172, Ch. 6]. In particular, it is shown that the Butterfly method satisfies necessary conditions for C^1-continuity only for the particular valences $n = 4, 5$, and 7, due to the clustering of eigenvalues for large valences. Similarly, it is shown that although the standard Loop method is formally C^1, in practice it does not always produce visibly smooth surfaces, again due to clustering of eigenvalues. Alternative weights for the Loop methods are proposed [172, Sec. 6.6] in order to alleviate this problem. Similarly, the alternative Modified Butterfly Method (see Section 4.2.3) is studied in [172, Sec. 6.4], and it is shown how the enlarged stencil of Figure $4.6_{/154}$ permits the avoidance of clustered eigenvalues and a proof of C^1-continuity. (It is also shown that enlargement of the Butterfly-scheme stencil is essential to achieve such results.)

In the book of Peters and Reif [124] a general presentation is given, for convergence and smoothness, using the methods and terminology of differential geometry. In Chapter 6 of [124] the results are applied to certain standard methods, including Catmull–Clark, Doo–Sabin, Midedge (called "Simplest Subdivision" in [124]), and variants of these methods. The necessary and sufficient conditions that we have illustrated in the nonregular case are presented in a general setting, and it is shown, for example, that a double subdominant eigenvalue is neither necessary nor sufficient for tangent-plane continuity.

In [124, Ch. 7], different levels of what might be called geometric continuity, parametrized by $r \geq 0$ for certain classes of surfaces C^k_r, are introduced. The case $r = 1$ corresponds to tangent-plane continuity combined with single sheetedness. Note that the subscript $r = 1$ here corresponds to the superscript 1 in C^1, when [172] refers to C^1-continuity. The case $r = 2$ corresponds to what is called *curvature continuity*. It is pointed out, however, that even this level of smoothness is not necessarily sufficient to provide satisfactory surfaces. Other characteristics, related to principal curvatures, convexity, ellipticity, hyperbolicity, and generalizations of these concepts, are studied in the context of the spectral properties of subdivision matrices. See also [123]. The relationship between second-order continuity and

quadratic precision, in the nonregular case, is also examined in [124]. Quadratic precision in the regular case is discussed in Section 6.4.

5.9 Additional comments

A fundamental reference related to Sections 5.1–5.3 above is the monograph [25]. In particular, [25] gives stronger results than those of Section 5.3, using more elaborate methods of analysis than those used in this book. See also the overview given by Dyn in [124, Sec. 1.7].

Important early papers in the nonregular case were [8, 9], the paper [132] which introduced the idea of the characteristic map, and the papers [91, 144] on the Loop method. See also the more recent papers [57, 159].

A summary of the references [124, 172] was given in Section 5.8, immediately preceding this one. Both of these references contain bibliographies. In particular, bibliographic summaries are given at the end of each chapter of [124], most of which pertain to the general questions of convergence and smoothness in the nonregular case. See also the papers [122, 173, 174], mentioned above.

Estimates of the minimal degree necessary for C^k continuity were given in [130, 133].

5.10 Exercises

1. Explain why $(5.6)_{/191}$ implies $(5.8)_{/192}$ if $p^0(z) \equiv 1$. Also, explain in detail why $(5.10)_{/192}$ and $(5.11)_{/192}$ follow from the statements preceding them.

2. Show that the trisection method defined by three times the first factor in $(4.19)_{/157}$ is not affine invariant.

3. Give the proof of Theorem $5.3.5_{/213}$.

4. Show that in the limit as $r \to 0^+$, the unit normal of the surface in Figure $5.2_{/222}$ is $[0, 0, 1]$.

5. Using the inclusion-exclusion principle, show that a k-ring neighbourhood contains $nk(k+1)+1$ grid points. Also, show that the dimension of σ'_k in $(5.66)_{/225}$ is $2n(k+1) \times 2n(k+1)$, and the dimension of σ''_k is $2n(k+1) \times nk(k+1)+1$.

6. Verify that $\lambda_{2,3}$, given by $(5.82)_{/228}$, satisfies $\lambda_{2,3} < 1/2$ for $n = 3$, $\lambda_{2,3} = 1/2$ for $n = 4$, and $\lambda_{2,3} > 1/2$ for $n > 4$.

7. Explain to which part of $(5.67)_{/225}$ (with $k = 1$) equation $(5.73)_{/226}$ corresponds, and to which part of $(5.67)_{/225}$ equations $(5.88)_{/230}$ and $(5.89)_{/230}$ correspond. Also, state to which variables in $(5.67)_{/225}$ the primed variables in $(5.73)_{/226}$, $(5.88)_{/230}$, and $(5.76)_{/227}$ correspond, and to which variables the unprimed variables correspond.

8. Prove the remark in the last paragraph of Section 5.5.

5.11 Project

1. *Algebraic verifications.*

   Using some symbolic-computation software, confirm the interval in $(5.61)_{/219}$. Similarly, many of the algebraic derivations in Section 5.5 were obtained using such software: verify the results for which details were not given.

Chapter 6

Evaluation and Estimation of Surfaces

In previous chapters we have usually assumed that subdivision surfaces are to be evaluated using the subdivision process. An alternative approach, however, is exact evaluation, i.e., the use of piecewise explicit formulas to evaluate the surface. For example, in the regular tensor-product B-spline case, we might consider doing the evaluation of $x(u, v)$ for some range of values of the parameters (u, v) by using a specialization of the algorithm of [127, Sec. 3.4], which simply evaluates the linear combination $(2.46)_{/74}$ using the explicit definitions of the B-spline nodal functions. (We do not need the full generality of the algorithm in [127], which applies in the more general case of nonuniform B-splines.) Another possibility is the use of the methods in Section 2.5.5, generalized to the tensor-product case, again to evaluate $x(u, v)$ for some range of parameter values.

In this chapter we examine how exact evaluation of the surface $x(u, v)$ can be done in cases more general than regular tensor-product B-splines.

Related to exact evaluation of the surface for arbitrary values of the parameters is the device of "pushing points to their limit." This refers to the following. Suppose that at some level of subdivision we have produced a refined sequence of control vectors approximating the exact surface at the corresponding points in parameter space. We then want to find the exact surface points for some, or all, of these parameter values and the corresponding exact tangent vectors to the surface. The process involves[42] the use of *evaluation stencils* and *tangent stencils*.

Following the discussion of these stencils, we describe methods of de Boor and Stam, for basic and variant methods, respectively, to evaluate the nodal functions exactly for any parameter value.

Also included in this chapter is a quite detailed study of precision sets, and the degree of polynomial reproduction, in both the univariate and bivariate cases. We also discuss the Wu–Peters method for finding tight bounding envelopes for the surface [75, 120, 169, 170]. The goal in this case is to find regions that tightly enclose the surface, given the control points. The chapter concludes with a brief discussion of adaptive subdivision, an application that requires good estimates of the error in surface patches.

6.1 Evaluation and tangent stencils for nonregular points

6.1.1 Evaluation stencils

Assume that we are given a subdivision method that is a variant of some basic method, as in Figure $1.30_{/33}$, and that it is stationary and affine invariant. We also assume that the method is convergent in the sense described in Theorem $5.5.2_{/232}$. The nonregular point is situated at the origin and the same notation is used as in Sections 5.5 and 5.6. In particular, see Figures $5.3_{/223}$ and $5.4_{/237}$, and equation $(5.63)_{/224}$: N_l is the nodal function associated with $l \in \mathbb{G}$ by application of the Nodal-Function Computation principle, and $l = 0$ corresponds to the origin. To simplify notation we take the parameter $h = 1$.

Assume that κ is the smallest number such that the set $\bar{\mathbb{G}}_\kappa$ corresponding to the κ-ring $\mathbb{G}_\kappa$ satisfies $\bar{\mathbb{G}}_\kappa \supseteq \operatorname{supp}(N_0)$, for any valence (including the case that the origin might be regular). Also let S_κ denote the corresponding local subdivision matrix. Then after the νth subdivision step we have, if $2^\nu y \in \bar{\mathbb{G}}_1$, the surface representation,

$$x(y) = \sum_{l \in \mathbb{G}_\kappa} p_l^\nu N_l(2^\nu y),$$

i.e.,

$$x(2^{-\nu} y) = \sum_{l \in \mathbb{G}_\kappa} p_l^\nu N_l(y) = \sum_{l \in \mathbb{G}_\kappa} p_l^{\nu,\kappa} N_l(y) \tag{6.1}$$

if $y \in \bar{\mathbb{G}}_1$. Here $p^{\nu,\kappa}$ is defined by $(5.64)_{/225}$. We now expand $p^{0,\kappa}$ as

$$p^{0,\kappa} = \sum_j \xi^j c_j,$$

where ξ_j are the, possibly generalized, eigenvectors of S_κ [56]. The vectors c_j are given by

$$c_j = (\eta^j)^* p^{0,\kappa} = \sum_{l \in \mathbb{G}_\kappa} \bar{\eta}_l^j p_l^{0,\kappa},$$

where the set of column vectors $\{\eta^j\}_j$ is the dual of the basis $\{\xi_j\}_j$, with the property that they are also, possibly generalized, eigenvectors of the transposed matrix S_κ^t, with the same eigenvalues.[43] The $*$-notation means transposition and complex conjugation and $\bar{\eta}_l^j$ denotes the complex conjugate of η_l^j. Then

$$p^{\nu,\kappa} = S_\kappa^\nu p^{0,\kappa} = \sum_j \lambda_j^\nu \xi^j c_j,$$

where λ_j are the eigenvalues of S_κ. If we insert into $(6.1)_{/248}$ and use that $\xi_l^1 = 1$ for all l, we get

$$\begin{aligned} x(2^{-\nu} y) &= \sum_{j \geq 1} c_j \lambda_j^\nu \sum_{l \in \mathbb{G}_\kappa} \xi_l^j N_l(y) = c_1 \sum_{l \in \mathbb{G}_\kappa} N_l(y) + \sum_{j \geq 2} c_j \lambda_j^\nu \sum_{l \in \mathbb{G}_\kappa} \xi_l^j N_l(y) \\ &= c_1 + \sum_{j \geq 2} c_j \lambda_j^\nu \sum_{l \in \mathbb{G}_\kappa} \xi_l^j N_l(y). \end{aligned} \tag{6.2}$$

Theorem 6.1.1. *For all $j > 1$ we have $|\lambda_j| < 1$.*

Proof. Suppose that $\lambda_1 = 1$ is of algebraic multiplicity at least 2 with a generalized eigenvector ξ^2 such that

$$S_\kappa \xi^1 = \xi^1 \quad \text{and} \quad S_\kappa \xi^2 = \xi^2 + \xi^1.$$

It follows that $S_\kappa^\nu \xi^2 = \nu\xi^2 + \xi^1$. Then, if we choose the initial sequence of control vectors such that $p^{0,\kappa} = \xi^2 c_2$ for some nonzero $c_2 \in \mathbb{R}^N$, we get $p^{\nu,\kappa} = (\nu\xi^2 + \xi^1)c_2$. Consequently, if $l \in \mathbb{G}_1$, we have

$$x(l/2^\nu) - p_l^\nu = x(l/2^\nu) - (\nu\xi_l^2 + \xi_l^1)c_2.$$

Here $\xi_l^2 \neq 0$ for at least one $l \in \mathbb{G}_1$, and letting $\nu \to \infty$ the first term in the right-hand side converges, whereas the second tends to infinity, which contradicts the convergence.

Similarly, we can exclude the possibility that $|\lambda_2| = 1$ with an eigenvector ξ^2 (possibly complex) linearly independent of ξ^1 and the possibility that $|\lambda_j| > 1$ for some j. The arguments are omitted. □

If we now let $\nu \to \infty$ in $(6.2)_{/248}$ and use that $|\lambda_j| < 1$ for $j > 1$, we have $x(0) = c_1$ and

$$c_1 = (\eta^1)^* p^{0,\kappa} = \sum_{l\in\mathbb{G}_\kappa} \bar\eta_l^1 p_l^0 = \sum_{l\in\mathbb{G}_\kappa} \bar\eta_l^1 p_l^\nu$$

for any fixed ν. The last equality follows since $(\eta^j)^* p^{\nu,\kappa} = \lambda_j^\nu c_j$. We have defined η^1 as the eigenvector of the real matrix S_κ^t with eigenvalue $\lambda_1 = 1$, and η^1 is therefore a real vector, possibly multiplied by a complex constant. From the requirement $(\eta^1)^*\xi^1 = \sum_l \bar\eta_l^1 = 1$, it follows that η^1 must be a real vector and the complex conjugation can be dropped, i.e.,

$$x(0) = c_1 = (\eta^1)^t p^{0,\kappa} = \sum_{l\in\mathbb{G}_\kappa} \eta_l^1 p_l^0 = \sum_{l\in\mathbb{G}_\kappa} \eta_l^1 p_l^\nu. \tag{6.3}$$

Similarly, it is clear that if λ_j is real, then the (possibly generalized) eigenvectors ξ^j and η^j can be chosen to be real. Now, by $(6.1)_{/248}$ with $y = 0$ we have

$$x(0) = \sum_{l\in\mathbb{G}_\kappa} p_l^\nu N_l(0) = \sum_{l\in\mathbb{G}_k} p_l^\nu N_0(-l).$$

We conclude that $N_l(0) = \eta_l^1$ for all $l \in \mathbb{G}_\kappa$. Since $N_l(0) = N_0(-l) = 0$ for $l \in \mathbb{G}_\kappa \setminus \mathbb{G}_{\kappa-1}$, the same is true for η_l^1 and we may write

$$\eta^1 = \begin{pmatrix} 0 \\ \eta^{1,\kappa-1} \end{pmatrix}.$$

Next, considering the structure of S_κ we have (compare with $(5.67)_{/225}$)

$$\begin{pmatrix} (\sigma'_{\kappa-1})^t & 0 \\ (\sigma''_{\kappa-1})^t & S_{\kappa-1}^t \end{pmatrix} \begin{pmatrix} 0 \\ \eta^{1,\kappa-1} \end{pmatrix} = \begin{pmatrix} 0 \\ S_{\kappa-1}^t \eta^{1,\kappa-1} \end{pmatrix} = \begin{pmatrix} 0 \\ \eta^{1,\kappa-1} \end{pmatrix},$$

and we conclude that $\eta^{1,\kappa-1}$ is the dominant eigenvector of $S^t_{\kappa-1}$ with $\lambda_1 = 1$ the corresponding simple eigenvalue. Thus, the evaluation stencil at the possibly nonregular point (the origin) is

$$\{\eta^1_l\}_{l\in\mathbb{G}_{\kappa-1}} = \{N_l(0)\}_{l\in\mathbb{G}_{\kappa-1}},$$

where η^1 is the unique dominating eigenvector of the local subdivision matrix $S^t_{\kappa-1}$.

For the case where we wish to evaluate $(6.1)_{/248}$ at some other $y \in \mathbb{G}_\kappa \setminus \{0\}$ we use the same procedure, but the matrix $S_{\kappa-1}$ may be different.

In the following example we derive evaluation stencils for the Catmull–Clark method. Evaluation stencils for the Loop method are discussed as part of Example $6.1.4_{/255}$, and stencils for the Doo–Sabin method are discussed in Section 7.2.1.

Example 6.1.2. The evaluation stencil for the Catmull–Clark method.

We use the notation of Figure $5.3_{/223}$. Here $\bar{\mathbb{G}}_2 \supset \operatorname{supp}(N_0)$ and $N_0(l) \neq 0$ only if $l \in \mathbb{G}_2$ $(\kappa = 2)$. Therefore, in order to determine the evaluation stencil around the origin, we must find the dominant eigenvector η^1 of S^t_1. We recall that S_1 is given by $(5.73)_{/226}$ and that the eigenvectors are calculated using $(5.83)_{/229}$ and $(5.84)_{/229}$. Note that the eigenspaces corresponding to different values of r, $0 \le r \le n-1$, are orthogonal (they are three dimensional if $r = 0$ and two dimensional if $0 < r \le n-1$).

Thus the eigenvector problem $S_1\xi^j = \lambda\xi^j$ when λ is given by $(5.79)_{/228}$ reduces to the eigenvector problem in $(5.77)_{/228}$, i.e.,

$$A\begin{pmatrix} f_0 \\ e_0 \\ V \end{pmatrix} = \lambda\begin{pmatrix} f_0 \\ e_0 \\ V \end{pmatrix}, \tag{6.4}$$

where

$$A = \begin{pmatrix} 1/4 & 1/2 & 1/4 \\ 1/8 & 1/2 & 3/8 \\ 1/4n & 3/2n & 1-7/4n \end{pmatrix}.$$

Introducing the notation

$$\xi = (F_{1/2}, F_{3/2}, \ldots, F_{n-1/2} \mid E_0, E_1, \ldots, E_{n-1} \mid V)^t,$$

for vectors in $\mathbb{R}^{2n+1}$ we recall from Section 5.5.2 that the eigenvectors of S_1 have the form

$$\xi^j = (f_0, f_0, \ldots, f_0 \mid e_0, e_0, \ldots, e_0 \mid V)^t$$

if λ_j are given by $(5.79)_{/228}$ and

$$\xi^j = (f_r w^{r/2}, \ldots, f_r w^{r(i+1/2)}, \ldots, f_r w^{r(n-1/2)} \mid e_r, e_r w^r, \ldots, e_r w^{ri}, \ldots, e_r w^{r(n-1)} \mid 0)^t,$$

$i = 0, \ldots, n-1$, with e_r and f_r solving $(5.78)_{/228}$, for other eigenvalues.

If we then solve the following eigenvector problem (with $\lambda = 1$) for the transposed matrix:

$$A^t \begin{pmatrix} nf'_0 \\ ne'_0 \\ V' \end{pmatrix} = \lambda \begin{pmatrix} nf'_0 \\ ne'_0 \\ V' \end{pmatrix}, \tag{6.5}$$

we have the orthogonality relations

$$nf_0 f'_0 + ne_0 e'_0 + VV' = 0 \tag{6.6}$$

if $(f_0, e_0, V)^t$ solves $(6.4)_{/250}$ with λ given by $(5.79)_{/228}$ and $\lambda \neq 1$. Now $(6.6)_{/251}$ expresses that

$$\eta = (f'_0, f'_0, \ldots, f'_0 \mid e'_0, e'_0, \ldots, e'_0 \mid V')^t$$

has the property that $(\eta^1)^t \xi^j = 0$ if ξ^j is the eigenvector corresponding to λ_j given by $(5.79)_{/228}$ and $\lambda \neq 1$. It also follows that $(\eta)^t \xi^j = 0$ for all λ_j, $j \neq 1$. Since the vector η is orthogonal to all vectors ξ^j, $j > 1$, it is a multiple of η^1.

Next, solving $(6.5)_{/251}$ with $\lambda = 1$, we get the solution

$$\eta = V'(1/n^2, \ldots, 1/n^2 \mid 4/n^2, \ldots, 4/n^2 \mid 1)^t,$$

and inserting into the normalization condition $(\eta)^t \xi^1 = 1$, we get

$$V'(1/n + 4/n + 1) = 1,$$

i.e., that $V' = n/(n+5)$ and

$$\eta = \eta^1 = \frac{n}{n+5}(1/n^2, \ldots, 1/n^2 \mid 4/n^2, \ldots, 4/n^2 \mid 1)^t. \tag{6.7}$$

For $n = 3$ we have

$$\eta^1 = (1/24, 1/24, 1/24 \mid 1/6, 1/6, 1/6 \mid 3/8)^t \tag{6.8}$$

and for $n = 4$ (the regular case),

$$\eta^1 = (1/36, 1/36, 1/36, 1/36 \mid 1/9, 1/9, 1/9, 1/9 \mid 4/9)^t.$$

Stencils such as these are usually illustrated by diagrams showing the weights to be assigned to points around a central point, as shown in Figure $6.1_{/252}$ for the cases $n = 3$ and $n = 4$.

For the regular case, $n = 4$, we have a tensor-product bicubic spline and in this case the evaluation stencil can be derived in a simpler way by means of a univariate subdivision polynomial. This is demonstrated below, in Example $6.2.1_{/257}$. ∎

6.1.2 Tangent stencils

Again we assume that we are given a subdivision method that is a variant of some basic method, is stationary and affine invariant, and is locally uniformly convergent outside the origin, as described in Theorem $5.5.2_{/232}$. With notation as in the

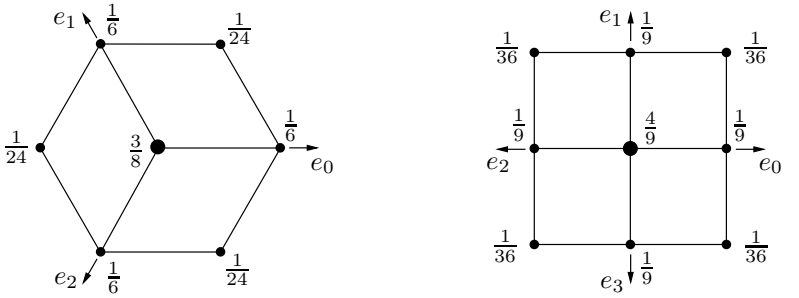


Figure 6.1. *Evaluation stencils ($n = 3, 4$) for the Catmull–Clark method.*

previous section, we assume in addition that the local subdivision matrix $S_{\kappa+1}$ has a double subdominant real eigenvalue $\lambda_{2,3}$ with two linearly independent eigenvectors ξ^j, $j = 2, 3$, and that $\lambda_1 > |\lambda_{2,3}| > |\lambda_j|$ for $j > 3$. We also assume that the characteristic map, with c_2 and c_3 linearly independent,

$$y = (u, v) \mapsto X(y) = c_2 \sum_{l \in \mathbb{G}_{\kappa+1}} \xi_l^2 N_l(y) + c_3 \sum_{l \in \mathbb{G}_{\kappa+1}} \xi_l^3 N_l(y),$$

is such that for its normal

$$n(y) = \frac{\partial X}{\partial u} \times \frac{\partial X}{\partial v} = n_0(y) c_2 \times c_3$$

we have $n_0(y) \geq c > 0$ for some constant c if $y \in \bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1$. Then, it follows in the same way as in Lemma 5.6.3$_{/236}$, which dealt with the Catmull–Clark method, that we have tangent-plane continuity in the origin and that two linearly independent tangent vectors there are

$$c_j = (\eta^j)^t p^{\nu,\kappa+1} = \sum_{l \in \mathbb{G}_{\kappa+1}} \eta_l^j p_l^\nu, \quad j = 2, 3. \tag{6.9}$$

The normal vector is given by $c_2 \times c_3$. Thus, the tangent stencils are given by the subdominant eigenvectors

$$\{\eta_l^j\}_{l \in \mathbb{G}_{\kappa+1}}, \quad j = 2, 3,$$

of the transposed local matrix $S_{\kappa+1}^t$. In the case that the subdominant eigenvalue $\lambda_{2,3}$ is also a double eigenvalue of $S_{\kappa-1}$, which is true for the Catmull–Clark method (with $\kappa = 2$), we have, for $j = 2, 3$, that $\eta_l^j = 0$ if $l \in \mathbb{G}_{\kappa+1} \setminus \mathbb{G}_{\kappa-1}$, and that the values η_l^j for $l \in \mathbb{G}_{\kappa-1}$ are equal to the values of $\eta_l^{j,\kappa-1}$ if we let $\eta^{j,\kappa-1}$ denote the corresponding eigenvector of $S_{\kappa-1}$. In fact, by (5.66)$_{/225}$ we have

$$S_\kappa^t = \begin{pmatrix} (\sigma'_{\kappa-1})^t & 0 \\ (\sigma''_{\kappa-1})^t & S_{\kappa-1}^t \end{pmatrix},$$

and therefore $S_{\kappa-1}^t \eta^{j,\kappa-1} = \lambda_j \eta^{j,\kappa-1}$ implies that

$$S_\kappa^t \begin{pmatrix} 0 \\ \eta^{j,\kappa-1} \end{pmatrix} = \begin{pmatrix} (\sigma'_{\kappa-1})^t & 0 \\ (\sigma''_{\kappa-1})^t & S_{\kappa-1}^t \end{pmatrix} \begin{pmatrix} 0 \\ \eta^{j,\kappa-1} \end{pmatrix} = \lambda_j \begin{pmatrix} 0 \\ \eta^{j,\kappa-1} \end{pmatrix}.$$

Similarly, it follows that

$$S_{\kappa+1}^t \begin{pmatrix} 0 \\ 0 \\ \eta^{j,\kappa-1} \end{pmatrix} = \lambda_j \begin{pmatrix} 0 \\ 0 \\ \eta^{j,\kappa-1} \end{pmatrix},$$

and since by assumption the eigenspace of $S_{\kappa+1}$ is two dimensional, it follows that

$$\eta^{j,\kappa+1} = \begin{pmatrix} 0 \\ 0 \\ \eta^{j,\kappa-1} \end{pmatrix},$$

where the size of the matrices denoted by 0 varies; see Section 5.5.2. Consequently, in this case the tangent stencils are given by the subdominant eigenvectors

$$\{\eta_l^j\}_{l \in \mathbb{G}_{\kappa-1}}, \quad j = 2, 3,$$

of the transposed local subdivision matrix $S_{\kappa-1}^t$.

Tangent stencils for the Catmull–Clark and Loop methods are discussed in the following two examples. For the Doo–Sabin method, see Section 7.2.1.

Example 6.1.3. Tangent stencils for the Catmull–Clark method.

In Example 6.1.2$_{/250}$ we noted that the eigenspaces of S_1 corresponding to different values of r were orthogonal. The eigenvectors ξ^j, $j = 2, 3$, corresponding to the subdominant eigenvalue $\lambda_{2,3}$ can now be found by solving the eigenvalue problems (5.78)$_{/228}$, for $r = 1$ and $r = n - 1$ and with $\lambda_{2,3}$ given by (5.81)$_{/228}$ with the plus sign. We have

$$A \begin{pmatrix} f_r \\ e_r \end{pmatrix} = \lambda_{2,3} \begin{pmatrix} f_r \\ e_r \end{pmatrix}, \tag{6.10}$$

where

$$A = \begin{pmatrix} 1/4 & 1/2\cos(\pi r/n) \\ (1/8)\cos(\pi r/n) & 3/8 + (1/8)\cos(2\pi r/n) \end{pmatrix}.$$

Consequently, from (5.84)$_{/229}$, (5.85)$_{/229}$, and (5.86)$_{/229}$, taking $r = 1$ and $e_1 = 1$, we have, when $n = 3$,

$$\xi^2 = \Re\xi = f_1(1/2, -1, 1/2 \mid 0, 0, 0 \mid 0)^t, +(0, 0, 0 \mid 1, -1/2, -1/2 \mid 0)^t,$$
$$\xi^3 = \Im\xi = f_1(\sqrt{3}/2, 0, -\sqrt{3}/2 \mid 0, 0, 0 \mid 0)^t + (0, 0, 0 \mid 0, \sqrt{3}/2, -\sqrt{3}/2 \mid 0)^t,$$

where $f_1 = (\sqrt{17} - 1)/2$.

Eigenvectors η of S_1^t are then found by solving

$$A^t \begin{pmatrix} f'_r \\ e'_r \end{pmatrix} = \lambda_{2,3} \begin{pmatrix} f'_r \\ e'_r \end{pmatrix} \tag{6.11}$$

and forming the eigenvector

$$\begin{aligned}\eta = e'_r(0,0,\ldots,0 \mid 1, w^r, \ldots, w^{jr}, \ldots, w^{(n-1)r} \mid 0)^t \\ + f'_r w^{r/2}(1, w^r, w^{2r}, \ldots, w^{jr}, \ldots, w^{(n-1)r} \mid 0,\ldots,0 \mid 0)^t,\end{aligned}$$

where $j = 0,\ldots,n-1$ and $w = \exp(2\pi i/n)$. A linearly independent eigenvector for the same eigenvalue is obtained by taking the complex conjugate η^*.

If $r < n$, straightforward calculations give that

$$f'_r = \frac{1}{4}(\sqrt{4 + \cos^2(\pi r/n)} - \cos(\pi r/n))e'_r.$$

Taking $r = 1$ and $e'_1 = 1$ and using that $w^n = 1$, we get the eigenvectors

$$\begin{aligned}\eta^2 = (0,0,\ldots,0 \mid 1, w, \ldots, w^j, \ldots, w^{-1} \mid 0)^t \\ + f'_1(w^{1/2}, w^{3/2}, \ldots, w^{j+1/2}, \ldots, w^{-1/2} \mid 0,0,\ldots,0 \mid 0)^t\end{aligned}$$

and

$$\begin{aligned}\eta^3 = (0,0,\ldots,0 \mid 1, w^{-1}, \ldots, w^{-j}, \ldots, w^1 \mid 0)^t \\ + f'_1(w^{-1/2}, w^{-3/2}, \ldots, w^{-(j+1/2)}, \ldots, w^{1/2} \mid 0,\ldots,0 \mid 0)^t,\end{aligned}$$

where

$$f'_1 = \frac{1}{4}(\sqrt{4 + \cos^2(\pi/n)} - \cos(\pi/n)).$$

Taking real and imaginary parts, we obtain the real eigenvectors

$$\begin{aligned}\eta^2 = (0,0,\ldots,0 \mid 1, \cos(2\pi/n), \ldots, \cos(2\pi j/n), \ldots, \cos(2\pi/n) \mid 0)^t \\ + f'_1(\cos(\pi/n), \cos(3\pi/n), \ldots, \cos((j+1/2)2\pi/n), \ldots, \cos(\pi/n) \mid 0,\ldots,0 \mid 0)^t\end{aligned}$$

and

$$\begin{aligned}\eta^3 = (0,0,\ldots,0 \mid 0, \sin(2\pi/n), \ldots, \sin(2\pi j/n), \ldots, -\sin(2\pi/n) \mid 0)^t \\ + f'_1(\sin(\pi/n), \sin(3\pi/n), \ldots, \sin((j+1/2)2\pi/n), \ldots, -\sin(\pi/n) \mid 0,\ldots,0 \mid 0)^t.\end{aligned}$$

These are the vectors that serve as tangent stencils for the Catmull–Clark method.

If we take $n = 3$, we get the linearly independent tangent stencils

$$\eta^2 = (0,0,0 \mid 1, -1/2, -1/2 \mid 0)^t + f'_1(1/2, -1, 1/2 \mid 0,0,0 \mid 0)^t$$

and

$$\eta^3 = (0,0,0 \mid 0, \sqrt{3}/2, -\sqrt{3}/2 \mid 0)^t + f'_1(\sqrt{3}/2, 0, -\sqrt{3}/2 \mid 0,0,0 \mid 0)^t,$$

where $f'_1 = (\sqrt{17} - 1)/8$. The stencil is illustrated for $n = 3$ in Figure 6.2$_{/255}$. For $n = 4$ we have $f'_1 = \sqrt{2}/4$ and

$$\begin{aligned}\eta^2 = (1/4, -1/4, -1/4, 1/4 \mid 1, 0, -1, 0 \mid 0)^t, \\ \eta^3 = (1/4, 1/4, -1/4, -1/4 \mid 0, 1, 0, -1 \mid 0)^t.\end{aligned}$$

Since this is the regular case with a bicubic tensor-product spline surface, we can also obtain these stencils by using (2.74)$_{/90}$.

Using the expressions that we have just derived for $n = 3$, for example, it is easy to verify, using (6.8)$_{/251}$, that $\eta^1\xi^2 = \eta^2\xi^1 = 0$. ∎

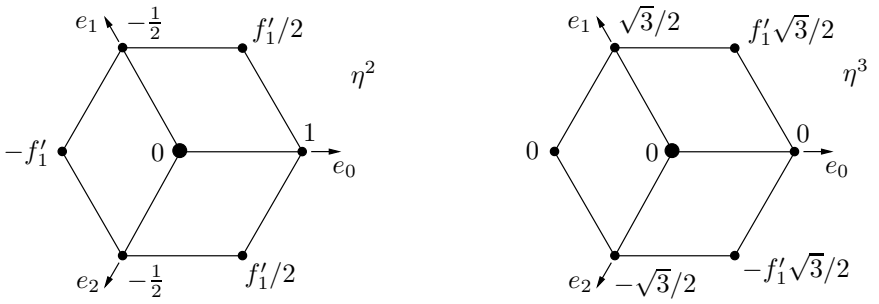


Figure 6.2. *Tangent stencils ($n = 3$) for the Catmull–Clark method.*

Example 6.1.4. Stencils for the Loop method.

We do not give a detailed analysis in this case, but state only the results, with references.

Exercise $12_{/49}$ asked for the $(n+1)\times(n+1)$ subdivision matrix corresponding to a 1-ring neighbourhood for the Loop method. The required matrix is given in the solutions document, on the book's Web site www.siam.org/books/ot120. We also give it here, in concise form. The leading $n \times n$ submatrix of $(S_1)_{(n+1)\times(n+1)}$ is band diagonal, with 3's on the diagonal and 1's on the two adjacent subdiagonals. The remainder of the matrix S_1 is defined as follows:

$$\begin{array}{lcl} (S_1)_{n+1,j} & = & 8w(n)/n, \quad j = 1, \ldots, n, \\ (S_1)_{i,n+1} & = & 3, \quad i = 1, \ldots, n, \\ (S_1)_{n+1,n+1} & = & 8(1 - w(n)). \end{array}$$

It is easy to verify that the left eigenvector corresponding to the dominant eigenvalue $\lambda_1 = 1$ is

$$\big(1, \ldots, 1 \mid (3/8)n/w(n)\big)^t$$

and this is the evaluation stencil for the Loop method (corresponding to η^1 in Example $6.1.2_{/250}$ for the Catmull–Clark method).

Similarly, it can be shown that there are two linearly independent eigenvectors corresponding to the subdominant eigenvalue $\lambda_{2,3} = (3 + 2\cos(2\pi/n))/8$, and they are $(c_1, \ldots, c_n \mid 0)$ and $(s_1, \ldots, s_n \mid 0)$, where $c_j = \cos(\frac{2\pi(j-1)}{n})$ and $s_j = \sin(\frac{2\pi(j-1)}{n})$. These vectors serve therefore as tangent stencils for the Loop method. The details can be found, for example, in [144, pp. 15, 55]. (The notation b in [144] corresponds to $8w(n)$ in our notation.) See also [172, Fig. 2.8, p. 31] and [66, Fig. 2]. In the latter reference, a different pair of linearly independent eigenvectors is given, but they span the same two-dimensional subspace. ∎

6.2 Evaluation and tangent stencils for subdivision-polynomial methods

For subdivision methods given by some subdivision polynomial, the computation of evaluation and tangent stencils is done exactly as in the previous section, and since

all points are regular, the stencils are the same at all control points. Sometimes, however, we might want more detailed information about the basis nodal functions.

Assume that we have a subdivision process given by some subdivision polynomial $s(z) = \sum_k s_k z^k$, as in Figure 1.30$_{/33}$ (third column, lower row). The shift-invariant nodal functions at the ν th step are denoted by $N(h2^{-\nu}; y - lh2^{-\nu})$. Due to the scaling properties we have $N(h2^{-\nu}; y - lh2^{-\nu}) = N(h; 2^\nu y - lh)$, and to simplify notation we can, without loss of generality, take $h = 1$ and $N(1; y) := N(y)$; see also Section 5.3.1.

After the ν th step of the subdivision we have the representation

$$x(y) = \sum_l p_l^\nu N(2^\nu y - l). \tag{6.12}$$

Here the sum may be over $l \in \mathbb{Z}^2$ or possibly over $l \in \mathbb{Z}^2 + \bar{e}/2$ (with notation as in Definition 3.5.1$_{/116}$) with $\bar{e} = (\epsilon_1, \epsilon_2)$ and $\epsilon_j \in \{0, 1\}$, $j = 1, 2$. In a centered version the vectors p_l^ν of the refined control vector sequence are related to a parameter value $l/2^\nu \in \mathbb{R}^2$, and if the process converges, then p_l^ν is an approximation of $x(l/2^\nu)$. We now wish to calculate the exact value of $x(l/2^\nu)$ at all or some of the grid points in $\mathbb{Z}^2/2^\nu$.

6.2.1 Evaluation of nodal functions at grid points

We assume that the subdivision process, with initial control sequence $p_l^0 = 1$ if $l = 0$ and $p_l^0 = 0$ if $l \neq 0$, converges uniformly towards a continuous function $N(y)$ not identically zero; compare Definition 5.1.1$_{/193}$ and Theorem 5.1.3$_{/193}$. According to this theorem the method must be affine invariant and, by Theorem 6.1.1$_{/249}$, $\lambda_1 = 1$ is an eigenvalue of multiplicity one of all local subdivision matrices, and the corresponding eigenvector is $(1, 1, \ldots, 1)^t$. All other eigenvalues have modulus strictly less than one.

Taking $y = k/2^\nu$ in (6.12)$_{/256}$ we get

$$x(k/2^\nu) = \sum_l p_l^\nu N(k - l) = \sum_l p_{k-l}^\nu N(l), \tag{6.13}$$

and it follows that the coefficients of the evaluation stencil are the values $N(l)$, $l \in \mathbb{Z}^2$, of the nodal basis function. With the previous notation we have $\eta_l^1 = N(-l)$. Letting $G = \{k : s_k \neq 0\}$ we have (see Section 4.4) $\operatorname{supp}(N) = \operatorname{conv}(G)$, and due to the continuity of N, $N(l) \neq 0$ only when l is in the interior of $\operatorname{conv}(G)$.

In order to find the values $N(k)$ for $k \in (\operatorname{conv}(G))^\circ \cap \mathbb{Z}^2$ (without loss of generality we can assume that $\bar{e} = (0, 0)$), we use the 2-scale relation from Theorem 5.1.3$_{/193}$,

$$N(y) = \sum_{k \in G} s_k N(2y - k). \tag{6.14}$$

Taking $y = l \in (\operatorname{conv}(G))^\circ$ we get, after a shift of summation variables,

$$N(l) = \sum_{k \in G} s_k N(2l - k) = \sum_k s_{2l-k} N(k), \tag{6.15}$$

where the last sum extends over $k \in \mathbb{Z}^2 \cap (\mathrm{conv}(G))^0$. Since $N(y) = 0$ outside $(\mathrm{conv}(G))^0$, we see that $(6.15)_{/256}$ is valid for all $l \in \mathbb{Z}^2$ with the summation taken over all $k \in \mathbb{Z}^2$. If we introduce the vector $\{N(l)\}_{l\in(\mathrm{conv}(G))^0}$, then $(6.15)_{/256}$ is equivalent to the eigenvalue problem $S^t_{\kappa-1}\eta^1 = \eta^1$ described in Section 6.1.1. Therefore, the eigenvector problem $(6.15)_{/256}$ has a one-dimensional solution space. The values $\{N(l)\}_{l\in(\mathrm{conv}(G))^0}$ are uniquely determined by the normalization condition

$$\sum_k N(k) = 1.$$

Finally we may observe that, once the values $N(k)$, $k \in (\mathrm{conv}(G))^0 \cap \mathbb{Z}^2$, have been computed, the values $N(k/2^\nu)$ can be found for all k and ν by the recursion formulas

$$N(l + \bar{e}/2) = \sum_k s_k N(2l + \bar{e} - k)$$

and

$$N(l + \bar{e}/2)/2^\nu) = \sum_k s_k N((2l + \bar{e})/2^\nu - k),$$

where $\bar{e} = (\epsilon_1, \epsilon_2)$ with $\epsilon_j = 0, 1$ and where, by the latter formula, N is determined on $\mathbb{Z}^2/2^{\nu+1}$ from its values on $\mathbb{Z}^2/2^\nu$.

Example 6.2.1. Bicubic tensor-product splines

For bicubic tensor-product splines we have the nodal functions $N(u, v) = N^4(u)N^4(v)$ with evaluation stencil $\{N(l_1, l_2) = N^4(l_1)N^4(l_2)\}_{l\in(\mathrm{conv}(G))^0}$, where $G = \{l : -2 \le l_i \le 2, i = 1, 2\}$ contains the index set of the bivariate subdivision polynomial. In the notation of Example $3.2.4_{/100}$, G is the set G_8. The set $\mathbb{G}_2$ is the smallest ring with convex hull containing $\mathrm{conv}(G)$, i.e., $\bar{\mathbb{G}}_2 \supseteq \mathrm{conv}(G)$, and $(\mathrm{conv}(G))^0 \cap \mathbb{Z}^2$ is equal to $\mathbb{G}_1 = \{l : -1 \le l_i \le 1, i = 1, 2\}$. We also observe that it suffices to solve the eigenvalue problem above for the univariate case (we have presented all analyses for the bivariate case, but the modifications needed for the univariate case are obvious). For univariate cubic splines we have the subdivision polynomial $s(z) = (z^{1/2} + z^{-1/2})^4/8 = (z^2 + 4z + 6 + 4z^{-1} + z^{-2})/8$, $G = \{-2, -1, 0, 1, 2\} \subset \mathbb{Z}$, $\mathbb{G}_1 = \{-1, 0, 1\}$, and the subdivision matrix S_1 given by

$$S_1 = \begin{pmatrix} 1/2 & 1/2 & 0 \\ 1/8 & 3/4 & 1/8 \\ 0 & 1/2 & 1/2 \end{pmatrix}.$$

It is straightforward to verify that the eigenvalues of S_1 are $1, 1/2$, and $1/4$. The eigenvector of S_1 corresponding to $\lambda_1 = 1$ is $\xi^1 = (1, 1, 1)^t$. We also get that the eigenvector η^1 of S^t_1 corresponding to $\lambda_1 = 1$ is $\eta^1 = (1/6, 2/3, 1/6)^t$ normalized so that $(\eta^1)^t\xi^1 = 1$. This is in accordance with Table $2.1_{/90}$, where the same stencil has been calculated by another method. (In Table $2.1_{/90}$, we are concerned with the row corresponding to $N^4(l/2)$ for integral values of $l/2$, since this corresponds to the primal grid.) The evaluation stencil for the cubic tensor-product spline (regular Catmull–Clark) is therefore the same as Figure $6.1_{/252}$ (right), so that we also have agreement with Example $6.1.2_{/250}$. ∎

6.2.2 Evaluation of derivatives of the nodal functions

We assume that the subdivisional polynomial is such that the nodal functions are C^1-continuous everywhere. Then, for any initial sequence of control vectors, we have an expansion of the form $(5.93)_{/236}$, i.e.,

$$x(2^{-\nu}y) = c_1 + \sum_j c_j \lambda_j^\nu \sum_l \xi_l^j N_l(y),$$

where ξ^j are, possibly generalized, eigenvectors. Differentiating, we get

$$\partial_u x(2^{-\nu}y) = 2^{-\nu}\left(\frac{\partial x}{\partial u}\right)(2^{-\nu}y) = \sum_{j\geq 2} c_j \lambda_j^\nu \sum_l \xi_l^j \partial_u N_l(y),$$

i.e.,

$$\left(\frac{\partial x}{\partial u}\right)(2^{-\nu}y) = \sum_{j\geq 2} c_j (2\lambda_j)^\nu \sum_l \xi_l^j \partial_u N_l(y) \tag{6.16}$$

and similarly for the v-derivative.

Now if, for example, $2\lambda_2 > 1$, then one could choose an initial sequence such that $p_l^0 = \xi_l^2$, and then the derivatives would tend to ∞ as $\nu \to \infty$, contradicting the assumption that the process is C^1. Similarly, one can exclude that all but one of the eigenvalues have modulus less than $1/2$, or that $\lambda = 1/2$ has higher multiplicity than two.

Consequently, for all local subdivision matrices S, we have a double subdominant eigenvalue $\lambda_{2,3} = 1/2$ with two linearly independent eigenvectors ξ^2 and ξ^3.

Next, differentiating in $(6.14)_{/256}$ we get

$$\partial_u N(y) = \sum_{k\in G} 2s_k \left(\frac{\partial N}{\partial u}\right)(2y-k).$$

Again taking $y = l \in \mathrm{conv}(G)^0$ and shifting variables, we get

$$\partial_u N(l) = \sum_{k\in G} 2s_k \left(\frac{\partial N}{\partial u}\right)(2l-k) = \sum_k 2s_{2l-k}\partial_u N(k). \tag{6.17}$$

It is now clear that $(6.17)_{/258}$ is equivalent to the eigenvalue problems $S_{\kappa-1}^t \eta^j = \lambda_{2,3}\eta^j$, $j = 2, 3$, in Section 6.1.2 with $\lambda_{2,3} = 1/2$, where κ is defined as in Section 6.1.1.

If we take $\{\partial_u N(k)\}$ and $\{\partial_v N(k)\}$ to be row vectors with $k \in \mathrm{conv}(G)^0$ and some ordering of the indices, we see that these vectors can be found by solving the eigenvector problem $(6.17)_{/258}$. However, since the solution space of $(6.17)_{/258}$ is two dimensional, we must use additional information if we wish to determine the vectors $\{\partial_u N(k)\}$ and $\{\partial_v N(k)\}$ completely. This additional information can most easily be found if, for example, we assume that the subdivision has generation degree equal to one, a concept which is analysed below in Section 6.4 (no methods used in practice fail to have this property). Thus, in addition to the assumption

in the previous section, we assume that the nodal function $N(y)$ is in $C^1(\mathbb{R}^2)$ and that it has generation degree equal to one, i.e.,

$$\sum_l lN(y-l) = y + a \tag{6.18}$$

with a a constant vector. This means that if we sample the control vectors from a linear function, then the subdivision produces the same linear function, modulo a constant, in the limit.

Using $(6.18)_{/259}$ we have, taking the first component $\sum_l l_1 N(u-l_1, v-l_2) = u + a_1$ and differentiating,

$$\sum_l l_1 \partial_u N(u-l_1, v-l_2) = 1 \quad \text{and} \quad \sum_l l_1 \partial_v N(u-l_1, v-l_2) = 0.$$

Similarly, we get

$$\sum_l l_2 \partial_u N(u-l_1, v-l_2) = 0 \quad \text{and} \quad \sum_l l_2 \partial_v N(u-l_1, v-l_2) = 1.$$

Using that these relations are valid for $(u, v) \in (\mathrm{conv}(G))^0 \cap \mathbb{Z}^2$, we get that $\{\partial_u N(k)\}$ and $\{\partial_v N(k)\}$ are uniquely defined.

We might note that even if we do not make the a priori assumption of generation degree one, it is always possible to extract additional information, sufficient to determine $\{\partial_u N(k)\}$ and $\{\partial_v N(k)\}$, from $(6.16)_{/258}$.

Evaluation of derivatives for box-spline nodal functions

For the evaluation of derivatives of box-spline nodal functions at integer points, it is also convenient to use the differentiability properties given in Theorem $3.2.3_{/99}$. If the subdivision polynomial is given by

$$s(z) = 4 \prod_{i=1}^{m} \frac{z^{e_i/2} + z^{-e_i/2}}{2},$$

then the directional derivative along the direction e_m can be computed using

$$\begin{aligned} D_{e_m} N(he^m; y) &= \frac{1}{h}(N(he^{m-1}; y + he_m/2) - N(he^{m-1}; y - he_m/2)) \\ &= \frac{z^{-e_m} - z^{e_m}}{h} N(he^{m-1}; y); \end{aligned}$$

see Theorem $3.2.3_{/99}$ (which is formulated for the uncentered case). Then, if

$$x(y) = \sum_l p_l^\nu N(he^m; y - lh) = \left(\sum_l p_l^\nu z^{2l} \right) N(he^m; y),$$

we get

$$D_{e_m}x(y) = \frac{z^{-e_m} - z^{e_m}}{h}\left(\sum_l p_l^\nu z^{2l}\right) N(he^{m-1}; y)$$
$$= \left(\sum_l \frac{p^\nu_{l+e_m/2} - p^\nu_{l-e_m/2}}{h} z^{2l}\right) N(he^{m-1}; y).$$

Taking $h = 2^{-\nu}$ and $y = k/2^\nu$, we have

$$D_{e_m}x(k/2^\nu) = \sum 2^\nu (p^\nu_{l+e_m/2} - p^\nu_{l-e_m/2}) N(e^{m-1}; k-l).$$

We conclude that the tangential directions $D_{e_m}x(k/2^\nu)$ can be computed by first taking the differences $2^\nu(p^\nu_{l+e_m/2} - p^\nu_{l-e_m/2})$ in the νth control sequence and then convolving with the stencil $\{N(e^{m-1}; l)\}$.

Finally, we remark that the vectors $D_{e_m}x(k/2^\nu)$ can be computed in the way described above for every subdivision method with a subdivision polynomial containing the factor $z^{e_m/2} + z^{-e_m/2}$, so that $s(z) = (z^{e_m/2} + z^{-e_m/2})q(z)$ with $q(z)$ a polynomial.

6.3 Exact parametric evaluation

In some situations one is interested in obtaining exact expressions for the nodal functions. For box-spline nodal functions we know from Theorem 3.2.9$_{/107}$ that the nodal function $N(he^m; y)$ is piecewise polynomial and that the domains of polynomiality are obtained as intersections of shadows of faces in the cube C^m. See Sections 3.1 and 3.2. In Section 6.3.1, we give a brief description of a method of de Boor which can be used to determine the piecewise polynomials for an arbitrary box-spline nodal function.

For variants of box-spline methods, it was believed for a long time that exact evaluation around a nonregular point was not possible. Stam disproved this in [149, 150] and described, for the Catmull–Clark and the Loop methods, how to evaluate the surface as a piecewise polynomial arbitrarily close to the nonregular point. This is presented in Section 6.3.2.

6.3.1 A method of de Boor

As mentioned earlier, the recursion formula (2.68)$_{/87}$ in Lemma 2.5.11$_{/87}$ for univariate splines can be used to determine recursively the piecewise polynomials defining $N^m(h; t)$. The formula (2.68)$_{/87}$ is a special case of a formula given by de Boor [35]. This formula can also be generalized to arbitrary box splines [38, p. 17] and used recursively to obtain explicit expressions for the piecewise polynomials defined on the subdomains of polynomiality. The computations can, however, be complicated and tedious. The procedure is illustrated in the case of the Zwart–Powell box spline in [38, p. 17].

de Boor's formula in the bivariate case

In the bivariate case, $(2.68)_{/87}$ generalizes to

$$(m-2)N^*(he^m;y) = \sum_{j=1}^{m} \alpha_j N^*(he^m_{(j)};y) + \sum_{j=1}^{m}(1-\alpha_j)N^*(he^m_{(j)};y-he_j), \quad (6.19)$$

where, as in Section 5.2, $e^m_{(j)}$ denotes $\{e_1,e_2,\dots,e_m\}\setminus\{e_j\}$, and $N^*(he^m_{(j)};y)$ the corresponding nodal function. Further, it is assumed that $y = h\sum_j \alpha_j e_j$. The coefficients α_j are in general not uniquely defined by this requirement, but if, for example, we assume that all α_j except two, say α_1 and α_2, are fixed, then α_1 and α_2 are coordinates for y with respect to the basis $\{e_1,e_2\}$. Consequently, as in $(2.68)_{/87}$, the coefficients in the right-hand side are polynomials of degree less than or equal to one. The coefficients α_j and $1-\alpha_j$, $j=1,2$, are of degree one, and the others are constants. Thus, both $(2.68)_{/87}$ and $(6.19)_{/261}$ express a box-spline nodal function of order m as a linear combination of nodal functions or order $m-1$ with coefficients that are polynomials of degree at most one. The formula $(2.68)_{/87}$ is, however, a unique representation of $N_0^m(h;t)$ of this kind, whereas for the representation $(6.19)_{/261}$ there are many degrees of freedom. We also note that the constant $m-1$ in the left-hand side of $(2.68)_{/87}$ is replaced by $m-2$ in the bivariate case.

In order to emphasize the analogy between $(2.68)_{/87}$ and $(6.19)_{/261}$, we take, in the univariate case, $e_1 = e_2 = \cdots = e_m = 1$. In this case, $N^*(he^m;y)$ corresponds to $N_0^m(h;t)$ and $N^*(he^m_{(j)};y)$ to $N_0^{m-1}(h;t)$. Then, if we let $t = \alpha_1 h$ and $\alpha_2 = \alpha_3 = \cdots = \alpha_m = 0$ and insert these values in $(6.19)_{/261}$ (with $m-2$ replaced by $m-1$), we obtain

$$\begin{aligned}(m-1)N_0^m(h;t) &= \alpha_1 N_0^{m-1}(h;t) + (1-\alpha_1)N_0^{m-1}(h;t-h)\\ &\quad + \sum_{j=2}^{m}(1-\alpha_j)N_0^{m-1}(h;t-h)\\ &= \alpha_1 N_0^{m-1}(h;t) + (1-\alpha_1)N_0^{m-1}(h;t-h)\\ &\quad + \sum_{j=2}^{m}(1-0)N_0^{m-1}(h;t-h)\\ &= (t/h)N_0^{m-1}(h;t) + (m-t/h)N_0^{m-1}(h;t-h),\end{aligned}$$

which is the formula $(2.68)_{/87}$.

A recursion analogous to $(2.70)_{/88}$ can also be found. See, however, [37] concerning actual implementation.

Proof of de Boor's formula

To simplify the presentation we give the proof for $h=1$.

Proof. Taking the directional derivative of $N^*(e^m;y)$ in the y-direction, we get

$$\begin{aligned}D_y N^*(e^m;y) &= \lim_{\varepsilon\to 0}(N^*(e^m;y+\varepsilon y) - N^*(e^m;y))/\varepsilon\\ &= y^t\nabla N^*(e^m;y) = u\partial_u N^*(e^m;y) + v\partial_v N^*(e^m;y).\end{aligned}$$

Since $y = \sum_{j=1}^{m} \alpha_j e_j$, we also have

$$u\partial_u N^*(e^m;y) + v\partial_v N^*(e^m;y) = \left(\sum_{j=1}^{m} \alpha_j e_j\right)^t \nabla N^*(e^m;y) = \sum_{j=1}^{m} \alpha_j e_j^t \nabla N^*(e^m;y)$$
$$= \sum_{j=1}^{m} \alpha_j D_{e_j} N^*(e^m;y) = \sum_{j=1}^{m} \alpha_j (N^*(e^m_{(j)};y) - N^*(e^m_{(j)};y-e_j)).$$

Next, taking the Fourier transform and using items 2, 5, and 9 of Table A.2$_{/313}$, we have

$$(i\partial_{\omega_1}(i\omega_1) + i\partial_{\omega_2}(i\omega_2))\widehat{N^*}(e^m,\omega) = \sum_{j=1}^{m} \alpha_j \widehat{N^*}(e^m_{(j)};\omega)(1-\exp(-i\omega_j^t e_j)),$$

and if we expand the operator $i\partial_{\omega_1}(i\omega_1) + i\partial_{\omega_2}(i\omega_2)$, we get

$$(-2 - \omega_1\partial_{\omega_1} - \omega_2\partial_{\omega_2})\widehat{N^*}(e^m,\omega) = \sum_{j=1}^{m} \alpha_j \widehat{N^*}(e^m_{(j)};\omega)(1-\exp(-i\omega_j^t e_j)). \quad (6.20)$$

Now, by (A.26)$_{/316}$,

$$\widehat{N^*}(e^m;\omega) = \prod_{j=1}^{m} \frac{1-\exp(-i\omega^t e_j)}{i\omega^t e_j}$$

and

$$\partial_{\omega_1}\widehat{N^*}(e^m;\omega) = \sum_{j=1}^{m} \partial_{\omega_1}\left(\frac{1-\exp(-i\omega^t e_j)}{i\omega^t e_j}\right) \prod_{1\le r\le m, r\ne j} \frac{1-\exp(-i\omega^t e_r)}{i\omega^t e_r}$$
$$= \sum_{j=1}^{m} \partial_{\omega_1}\left(\frac{1-\exp(-i\omega^t e_j)}{i\omega^t e_j}\right) \widehat{N^*}(e^m_{(j)};\omega). \quad (6.21)$$

Next,

$$\partial_{\omega_1}\left(\frac{1-\exp(-i\omega^t e_j)}{i\omega^t e_j}\right) = \frac{i\exp(-i\omega^t e_j)}{i\omega^t e_j} e_{j1} - \frac{1-\exp(-i\omega^t e_j)}{i(\omega^t e_j)^2} e_{j1},$$

where $e_j = (e_{j1}, e_{j2})^t$, and multiplying (6.21)$_{/262}$ by ω_1, we have

$$\omega_1\partial_{\omega_1}\widehat{N^*}(e^m;\omega) = \sum_{j=1}^{m} \frac{\exp(-i\omega^t e_j)}{\omega^t e_j} \omega_1 e_{j1} \widehat{N^*}(e^m_{(j)};\omega)$$
$$- \sum_{j=1}^{m} \frac{1-\exp(-i\omega^t e_j)}{i(\omega^t e_j)^2} \omega_1 e_{j1} \widehat{N^*}(e^m_{(j)};\omega)$$
$$= \sum_{j=1}^{m} \frac{\exp(-i\omega^t e_j)}{\omega^t e_j} \omega_1 e_{j1} \widehat{N^*}(e^m_{(j)};\omega) - \sum_{j=1}^{m} \frac{1}{\omega^t e_j} \omega_1 e_{j1} \widehat{N^*}(e^m;\omega).$$

Using the similar expression for $\omega_2\partial_{\omega_2}\widehat{N^*}(e^m;\omega)$, we now get

$$(-2-\omega_1\partial_{\omega_1}-\omega_2\partial_{\omega_2})\widehat{N^*}(e^m;\omega)=(m-2)\widehat{N^*}(e^m;\omega)-\sum_{j=1}^{m}\exp(-i\omega^t e_j)\widehat{N^*}(e^m_{(j)};\omega),$$

and using $(6.20)_{/262}$,

$$(m-2)\widehat{N^*}(e^m;\omega)-\sum_{j=1}^{m}\exp(-i\omega^t e_j)\widehat{N^*}(e^m_{(j)};\omega)=\sum_{j=1}^{m}\alpha_j\widehat{N^*}(e^m_{(j)};\omega)(1-\exp(-i\omega_j^t e_j)).$$

Rearranging the terms, we obtain

$$(m-2)\widehat{N^*}(e^m;\omega)=\sum_{j=1}^{m}\alpha_j\widehat{N^*}(e^m_{(j)};\omega)+\sum_{j=1}^{m}(1-\alpha_j)\exp(-\omega^t e_j)\widehat{N^*}(e^m_{(j)};\omega),$$

and taking the inverse Fourier transform we have $(6.19)_{/261}$. □

6.3.2 Stam's method

As mentioned in the introduction to this section, Stam [149, 150] has shown how the surfaces generated by methods like Catmull–Clark and Loop can be evaluated in closed form in a neighbourhood of a nonregular point. The idea behind the method is fairly simply explained, and it is illustrated here for the Catmull–Clark scheme [150]. The method can be implemented in conjunction with any subdivision scheme for which we have explicit expressions for the nodal functions in the regular case (including therefore 4-8 subdivision [164, Sec. 1]).

Let the origin be a nonregular point with valence $n \neq 4$ (see Figure $6.3_{/264}$, where $n=3$). Here the nodes in $\mathbb{G}_3$ are depicted, the set $\bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1$ is lightly shaded, and certain subsets of the nodes, to be defined below, are marked with circles and squares. The sets $\bar{\mathbb{G}}_1$, $\bar{\mathbb{G}}_2$, and $\mathbb{G}_3$ are those defined in Section 5.5.1. We also introduce subsets $\Omega_{j+k/3} \subset \bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1$ of the parametric domain, with $0 \leq j \leq n-1$ and $0 \leq k \leq 2$, defined by

$$\begin{aligned}
\Omega_j &= \{y : y = \gamma_j e_j + \gamma_{j+1}e_{j+1}, 1 < \gamma_j \leq 2, 0 \leq \gamma_{j+1} \leq 1\},\\
\Omega_{j+1/3} &= \{y : y = \gamma_j e_j + \gamma_{j+1}e_{j+1}, 1 < \gamma_j \leq 2, 1 \leq \gamma_{j+1} \leq 2\},\\
\Omega_{j+2/3} &= \{y : y = \gamma_j e_j + \gamma_{j+1}e_{j+1}, 0 \leq \gamma_j < 1, 1 < \gamma_{j+1} \leq 2\},
\end{aligned}$$

and

$$\mathcal{Y}_j = \{y : y = \gamma_j e_j + \gamma_{j+1}e_{j+1}, 0 < \gamma_j \leq 1, 0 \leq \gamma_{j+1} \leq 1\}.$$

These subsets can be viewed as defining patches in $\mathbb{R}^N$.

Now, given an initial sequence of control vectors $\{p_l^0\}$, the surface for $y \in \bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1$ is given by

$$x(y) = \sum_{l \in \mathbb{G}_3} p_l^{3,0} N_l(y),$$

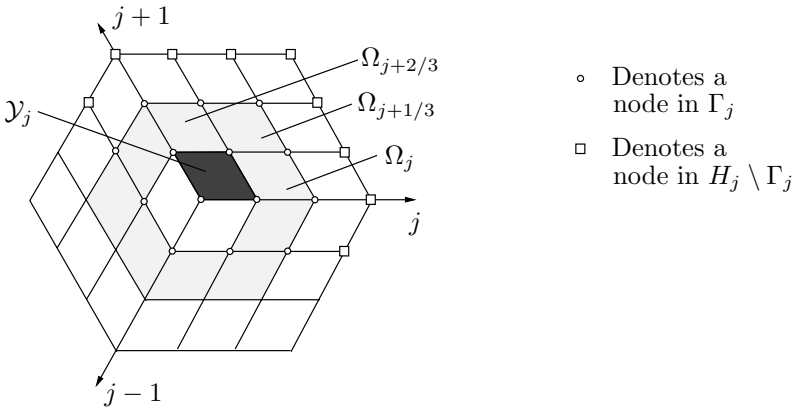


Figure 6.3. *Control points for Stam's method.*

where $p_l^{3,0} = p_l^0$, $l \in \mathbb{G}_3$, and where the restriction of the functions N_l to the subdomains $\Omega_{j+k/3}$, $0 \le k \le 2$, $0 \le j \le n-1$, are tensor-product bicubic polynomials which are known and can be obtained explicitly from $(2.68)_{/87}$. After the ν th subdivision step we have, from $(5.92)_{/235}$,

$$x(y) = \sum_{l \in \mathbb{G}_3} p_l^{3,\nu} N_l(2^\nu y)$$

if $2^\nu y \in \bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1$, i.e.,

$$x(2^{-\nu} y) = \sum_{l \in \mathbb{G}_3} p_l^{3,\nu} N_l(y) \tag{6.22}$$

if $y \in \bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1$. Now, if $\{\xi^j\}_{1 \le j \le 12n+1}$ denote the eigenvectors of the local subdivision matrix S_3 and $\{\eta^j\}_{1 \le j \le 12n+1}$ those of S_3^t, we may expand $p^{0,3}$ as

$$p^{0,3} = \sum_{j=1}^{12n+1} \xi^j c_j \tag{6.23}$$

with $c_j = (\eta^j)^t p^{0,3}$. Then we get

$$p^{\nu,3} = \sum_{j=1}^{12n+1} \lambda_j^\nu \xi^j c_j, \tag{6.24}$$

and inserting in $(6.22)_{/264}$, we get

$$x(2^{-\nu} y) = \sum_{l \in \mathbb{G}_3} \left(\sum_{j=1}^{12n+1} \lambda_j^\nu c_j \xi_l^\nu \right) N_l(y),$$

which gives the polynomial representation when $2^{-\nu} y \in 2^{-\nu}(\bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1)$.

In the above analysis we used the entire $(12n+1)\times(12n+1)$ matrix S_3, and finding the expansions $(6.23)_{/264}$ and $(6.24)_{/264}$ above is equivalent to diagonalizing the matrix S_3 as

$$S_3 = XDX^{-1}$$

with $D = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_{12n+1})$, $X = (\xi^1, \xi^2, \ldots, \xi^{12n+1})$, and the inverse of X given by $X^{-1} = (\eta^1, \eta^2, \ldots, \eta^{12n+1})^t$. The vector $p^{\nu,3}$ is computed as

$$p^{\nu,3} = XD^{\nu}X^{-1}p^{0,3}.$$

Due to the particular structure of the subdivision matrices S_2 and S_3, the diagonalization of S_3 can be considerably simplified and the analysis need only involve the initial vector $p^{0,2}$. By $(5.66)_{/225}$ we have

$$S_{k+1} = \begin{pmatrix} \sigma'_k & \sigma''_k \\ 0 & S_k \end{pmatrix}$$

for $k \geq 1$. Further, as pointed out in Section 5.5.2, the control vectors $p_l^{3,\nu}$ with $l \in \mathbb{G}_3 \setminus \mathbb{G}_2$ do not affect the vectors $p_l^{3,\nu+1}$ with $l \in \mathbb{G}_3$, and therefore $\sigma'_2 = 0$, so that

$$S_3 = \begin{pmatrix} 0 & \sigma''_2 \\ 0 & S_2 \end{pmatrix}.$$

This gives

$$S_3^{\nu} = \begin{pmatrix} 0 & \sigma''_2 S_2^{\nu-1} \\ 0 & S_2^{\nu} \end{pmatrix},$$

and, with $p^{\nu,3} = \binom{\rho^{\nu,2}}{p^{\nu,2}}$,

$$p^{\nu,3} = S_3^{\nu}p^{0,3} = \begin{pmatrix} 0 & \sigma''_2 S_2^{\nu-1} \\ 0 & S_2^{\nu} \end{pmatrix} \begin{pmatrix} \rho^{0,2} \\ p^{0,2} \end{pmatrix} = \begin{pmatrix} \sigma''_2 S_2^{\nu-1} \\ S_2^{\nu} \end{pmatrix} p^{0,2} = \begin{pmatrix} \sigma''_2 \\ S_2 \end{pmatrix} p^{\nu-1,2}$$

and $p^{\nu,2} = S_2^{\nu}p^{0,2}$, $\rho^{\nu,2} = \sigma''_2 S_2^{\nu-1}p^{0,2} = \sigma''_2 p^{\nu-1,2}$.

It follows that, in order to evaluate the vector $p^{\nu,3}$ in $(6.24)_{/264}$, it suffices to diagonalize the $(6n+1)\times(6n+1)$ matrix S_2 and compute

$$\begin{aligned} p^{\nu-1,2} &= S_2^{\nu-1}p^{0,2}, \\ p^{\nu,3} &= \begin{pmatrix} \sigma''_2 \\ S_2 \end{pmatrix} p^{\nu-1,2}. \end{aligned}$$

As shown in [150] the complexity of the calculations can be further reduced by considering smaller local subdivision matrices which are submatrices of S_2.

Assume that, for a fixed j, we wish to find a polynomial representation on the subpatch $\mathcal{Y}_j \subset \bar{\mathbb{G}}_1 \setminus \{0\}$. Let Γ_j be the set of nodes l such that $\mathrm{supp}\,(N_l(y)) \cap \mathcal{Y}_j \neq \emptyset$. The nodes in Γ_j are denoted by circles in Figure $6.3_{/264}$. Also, let H_j be the set of nodes such that $\mathrm{supp}\,(N_l(y)) \cap (2\mathcal{Y}_j) \neq \emptyset$. The cardinality of Γ_j is $2n+8$ and the cardinality of H_j is $2n+17$.

Next introduce vectors

$$\pi^{\nu,3} = \{p_l^\nu\}_{l\in H_j}, \quad \pi^{\nu,2} = \{p_l^\nu\}_{l\in\Gamma_j}, \quad \text{and} \quad \varrho^{\nu,2} = \{p_l^\nu\}_{l\in H_j\setminus\Gamma_j},$$

so that

$$\pi^{\nu,3} = \begin{pmatrix} \varrho^{\nu,2} \\ \pi^{\nu,2} \end{pmatrix}.$$

Now,

$$l \in H_j \Leftrightarrow \operatorname{supp}(N_l(2^\nu y)) \cap (2^{-\nu+1}\mathcal{Y}_j) \neq \emptyset.$$

Therefore, if $y \in \mathcal{Y}_j$ and $2^\nu y \in \bar{\mathbb{G}}_2 \setminus \bar{\mathbb{G}}_1$, we have

$$x(2^{-\nu}y) = \sum_{l\in H_j} p_l^\nu N_l(y) = \sum_{l\in H_j} \pi_l^{\nu,3} N_l(y),$$

so that in order to evaluate $x(y)$ in $\mathcal{Y}_j$ we need to compute the vectors $\pi^{\nu,3}$.

Using that the control vectors p_l^ν, $l \in H_j$, are affected only by $p_l^{\nu-1}$, $l \in \Gamma_j$, it follows, similarly to the analysis above, that

$$\pi^{\nu,3} = S^\nu \pi^{0,3} = \begin{pmatrix} 0 & \sigma A^{\nu-1} \\ 0 & A^\nu \end{pmatrix} \begin{pmatrix} \varrho^{0,2} \\ \pi^{0,2} \end{pmatrix} = \begin{pmatrix} \sigma A^{\nu-1} \\ A^\nu \end{pmatrix} \pi^{0,2} = \begin{pmatrix} \sigma \\ A \end{pmatrix} \pi^{\nu-1,2}$$

and

$$\begin{aligned} \pi^{\nu,2} &= A^\nu \pi^{0,2}, \\ \varrho^{\nu,2} &= \sigma A^{\nu-1} \rho^{0,2} = \sigma \pi^{\nu-1,2}. \end{aligned}$$

Here S is a submatrix of S_3, A is a submatrix of S_2 of order $(2n+8)\times(2n+8)$, and σ is a submatrix of σ_2'' of order $9\times(2n+8)$. We can now diagonalize explicitly the lower-order matrix A and compute the vector $\pi^{\nu,3}$.

Peters and Reif [124, p. 81 (note 6)] have remarked that it is not necessarily a good idea to carry out the diagonalization of the matrices A above. In fact, for the Catmull–Clark method it is numerically more efficient to compute the vector $\pi^{\nu,3}$ by simple repeated multiplication by A unless ν is very large, e.g., $\nu > 16$. This is due to the fact that the matrix A is relatively sparse.

6.4 Precision sets and polynomial reproduction

In addition to bounds given in terms of global constants, such as those in the convergence proofs in Section 5.2, other descriptions of the proximity of the control points and the limit surface are of interest. In this section and the next we consider two such descriptions. We begin in this section with a discussion of the concept of *precision* of the subdivision process, and related concepts such as the degree of polynomial reproduction.

6.4.1 Conditions for given reproduction and generation degree

Two quantities that categorize the behaviour of a (stationary uniform) subdivision scheme, when the control points lie on a polynomial, are the reproduction degree and the generation degree, which were presented in the following way in the univariate case in [67, Sec. 5]. Let $d \geq 1$ be the maximal degree for which the following hold:

1. For all polynomials up to degree d, if the control points lie on the polynomial, then the limit curve is the same polynomial. In this case d is called the *reproduction degree*.

2. For all polynomials up to degree d, if the control points lie on the polynomial, then the limit curve is a polynomial, of the same degree and with the same leading term. In this case d is called the *generation degree*.

The reproduction degree is a measure of the precision of the subdivision method: if the initial control-point data is obtained by uniform sampling of a polynomial of degree less than or equal to the reproduction degree, then the subdivision method converges to that polynomial. As mentioned in Section 1.3, the sets of polynomials reproduced are referred to as *precision classes*. For example, a method that reproduces polynomials up to degree 1 is said to have linear precision, up to degree 2 quadratic precision, and up to degree 3 cubic precision. Such methods are also called *quasi-interpolation* methods [28].

The generation degree is clearly at least as large as the reproduction degree. If the generation degree is equal to d, then the subdivision method reproduces polynomials of degree up to and including d, provided that we subject the initial data to a preprocessing step involving the solution of an upper-triangular linear system of equations. This is discussed below.

In this section we focus on the bivariate case. For consistency with the literature, results are stated and proved for $h = 1$. We also use the following notation for bivariate derivatives and multi-indices in $\mathbb{Z}^2$. The notation ∂_i denotes the partial derivative $\partial/\partial\omega_i$, $i = 1, 2$. If $k = (k_1, k_2)$, then ∂^k denotes $\partial_1^{k_1}\partial_2^{k_2}$. If $k \in \mathbb{Z}^2$, then $|k| = |k_1| + |k_2|$, and $k \geq 0$ means that $k_i \geq 0$, $i = 1, 2$, while $k > 0$ means $k \geq 0$ and in addition $k_1 + k_2 > 0$. Further $r \leq k$ and $r < k$ mean, respectively, that $k - r \geq 0$ and $k - r > 0$. If $z = (z_1, z_2)$ and $l = (l_1, l_2) \in \mathbb{Z}^2$, then $z^k = z_1^{k_1} z_2^{k_2}$ and $l^k = l_1^{k_1} l_2^{k_2}$. Further for binomial coefficients we use the convention that $\binom{k}{r} = \binom{k_1}{r_1}\binom{k_2}{r_2}$. Then if $a, b \in \mathbb{Z}^2$, we have

$$(a + b)^k = (a_1 + b_1)^{k_1}(a_2 + b_2)^{k_2} = \sum_{0 \leq r \leq k} \binom{k}{r} a^{k-r} b^r.$$

We also use the notation

$$\Pi_d \quad \text{and} \quad \Pi_{(d_1,d_2)}$$

for, respectively,

- the linear space consisting of all polynomials in $y = (u, v)$ of degree less than or equal to d,

- the linear space consisting of polynomials of bidegree less than or equal to (d_1, d_2), i.e., of degree less than or equal to d_1 in u and less than or equal to d_2 in v.

Let $s(z)$ be a subdivision polynomial defining a convergent subdivision process in the sense of Definition 5.1.1$_{/193}$. Then for a given sequence of control vectors

$$\{p_l\}_{l\in\mathbb{Z}^2} \subset \mathbb{R}^N, \tag{6.25}$$

the subdivision process defines the parametric surface

$$\mathbb{R}^2 \ni y \mapsto \sum_{l\in\mathbb{Z}^2} p_l N(y-l) \in \mathbb{R}^N,$$

where $N(y)$ denotes the continuous nodal function obtained by choosing $p_l = 1$ for $l = (0,0)$ and $p_l = 0$ for $l \neq (0,0)$.

For the bivariate case we use the following definitions for reproduction degree or bidegree, and generation degree or bidegree.

Definition 6.4.1. *The subdivision process is said to have*

- *reproduction degree d (respectively, bidegree (d_1, d_2)) if*

$$\sum_{l\in\mathbb{Z}^2} p(l)N(y-l) = p(y)$$

for all $p \in \Pi_d$ (respectively, $p \in \Pi_{(d_1,d_2)}$);

- *generation degree d (respectively, bidegree (d_1, d_2)) if for every monomial $y^k \in \Pi_d$ (respectively, $y^k \in \Pi_{(d_1,d_2)}$), we have*

$$\sum_{l\in\mathbb{Z}^2} l^k N(y-l) = y^k + \sum_{0\le r<k} c_{k,r} y^r \tag{6.26}$$

with $c_{k,r}$ constants.

Thus, for example, the reproduction degree is d, if whenever the control vectors are samples in $\mathbb{Z}^2$ of some polynomial function of degree less than or equal to d, then the subdivision process reproduces exactly the same polynomial.

From the definition, it follows fairly easily that if the generation degree is d, then for every $p \in \Pi_d$ there exists exactly one polynomial $q \in \Pi_d$ such that

$$\sum_{l\in\mathbb{Z}^2} q(l)N(y-l) = p(y),$$

and similarly for the bidegree case. In fact we have the following proposition and its bidegree counterpart.

Proposition 6.4.2. *Assume that the generation degree is d. Then* $(6.26)_{/268}$ *defines a mapping of monomials* $\Pi_d \ni y^k \mapsto y^k + \sum_{0\le r<k} c_{k,r} y^r \in \Pi_d$ *and, extending linearly, a linear mapping* $\mathcal{S} : \Pi_d \to \Pi_d$. *This mapping* $\mathcal{S}$ *is one-to-one, and since the dimension of* Π_d *is finite, it is onto and invertible.*

Proof. Since $\mathcal{S}$ is linear, it fails to be one-to-one only if there exists a nonzero polynomial such that $\mathcal{S}(p) = 0$. Now if $p \in \Pi_d$ and $p \neq 0$, then the terms in p with maximal degree remain unchanged in $\mathcal{S}(p)$, and therefore $\mathcal{S}(p) \neq 0$. □

Therefore every polynomial p can be produced, or *generated*, by the subdivision process by choosing the control vectors as samples of some polynomial with the same leading terms. This is the reason for the terminology.

We now want to find conditions on $s(z)$ and the nodal function N giving a certain reproduction or generation degree. In Chapter 2 we introduced the concept of partition of unity for a nodal function $N(y)$, meaning that

$$\sum_{l\in\mathbb{Z}^2} N(y-l) = 1.$$

For the analysis of reproduction and generation degree for arbitrary subdivision processes defined by subdivision polynomials, we need to analyse partition of unity, and generalizations thereof, for more general functions $F \in L^1(\mathbb{R}^2)$ having compact support.

We now have the following theorem, where equalities involving F are only claimed to hold almost everywhere.

Theorem 6.4.3. *We have partition of unity, i.e.,*

$$\sum_{l\in\mathbb{Z}^2} F(y-l) = 1$$

if and only if

$$\hat{F}(0) = 1, \quad \textit{and} \quad \hat{F}(2\pi l) = 0 \quad \textit{for all} \ \ l \neq 0. \tag{6.27}$$

Next, let $y^k = u^{k_1} v^{k_2}$ *be a monomial. Then*

$$\sum_{l\in\mathbb{Z}^2} l^k F(y-l) = y^k$$

if and only if F *satisfies condition* $(6.27)_{/269}$ *and in addition satisfies*

$$\partial^r \hat{F}(2\pi l) = 0 \quad \textit{for} \quad l \neq 0 \quad \textit{and} \quad 0 < r \le k, \tag{6.28}$$

$$\partial^r \hat{F}(0) = 0 \qquad \textit{for} \qquad 0 < r \le k. \tag{6.29}$$

Further,

$$\sum_{l\in\mathbb{Z}^2} l^k F(y-l) = y^k + \sum_{0\le r<k} c_r y^r,$$

with c_r *constants, if and only if* F *satisfies conditions* $(6.27)_{/269}$ *and* $(6.28)_{/269}$.

Proof. In order to prove the first statement in the theorem, we note that the function $g(y) = \sum_l F(y-l)$ is doubly periodic in $y = (u, v)$ with periods 1. Therefore (see Section A.2.4 in the Appendix) it can be expanded in a two-dimensional Fourier series

$$g(y) = \sum_{k \in \mathbb{Z}^2} c_k \exp(i2\pi k^t y),$$

where $c_k = \int_D g(y) \exp(-i2\pi k^t y)\, dy$ and $D = (0,1) \times (0,1)$. We have $g(y) = 1$ if and only if $c_0 = 1$ and $c_k = 0$ for $k \neq 0$. We now get

$$c_k = \sum_l \int_D F(y-l) \exp(-i2\pi k^t y)\, dy.$$

Since

$$\begin{aligned}\int_D F(y-l) \exp(-i2\pi k^t y)\, dy &= \exp(-i2\pi k^t l) \int_{D-l} F(y) \exp(-i2\pi k^t y)\, dy \\ &= \int_{D-l} F(y) \exp(-i2\pi k^t y)\, dy,\end{aligned}$$

we get, after summing over l,

$$c_k = \int_{\mathbb{R}^2} F(y) \exp(-i2\pi k^t y)\, dy = \hat{F}(2\pi k),$$

which shows that the first statement in the theorem is equivalent to $(6.27)_{/269}$.

Now let $y^k = u^{k_1} v^{k_2}$ be a monomial. To prove the second statement of the theorem we let $g(y) = \sum_l l^k F(y-l)$. Using the binomial theorem we expand l^k as

$$l^k = ((l-y)+y)^k = \sum_{0 \leq r \leq k} \binom{k}{r} (l-y)^{k-r} y^r.$$

Consequently,

$$g(y) = \sum_{0 \leq r \leq k} y^r \sum_l \binom{k}{r} (l-y)^{k-r} F(y-l) = \sum_{0 \leq r \leq k} y^r g_r(y)$$

with

$$g_r(y) = \sum_l \binom{k}{r} (l-y)^{k-r} F(y-l). \tag{6.30}$$

It is clear that the functions $g_r(y)$ are doubly periodic with periods 1. It also follows that $g(y) = y^k$ if and only if $g_r(y) =$ for $0 \leq r < k$ and $g_k(y) = 1$. Arguing as in the previous step, it follows that $g_k(y) = \sum_l F(y-l) = 1$ if and only if condition $(6.27)_{/269}$ is satisfied. Also, $g_r(y) = 0$ for $0 \leq r < k$ if and only if all its Fourier coefficients vanish, i.e., if and only if

$$(y^{k-r} F(y))\hat{}(2\pi l) = (-i\partial)^{k-r} \hat{F}(2\pi l) = 0 \quad \text{for } 0 \leq r < k \quad \text{and for all} \quad l \in \mathbb{Z}^2,$$

where $(y^{k-r}F(y))\hat{}$ denotes the Fourier transform of $y^{k-r}F(y)$. The displayed condition is equivalent to conditions $(6.28)_{/269}$ and $(6.29)_{/269}$.

To prove the third statement of the theorem we note that the functions $g_r(y)$, $0 \le r < k$, are constant exactly when $\hat{g}_r(2\pi l) = 0$ for $l \neq 0$, i.e., when

$$(y^{k-r}F(y))\hat{}(2\pi l) = (-i\partial)^{k-r}\hat{F}(2\pi l) = 0 \quad \text{for all} \quad l \in \mathbb{Z}^2 \setminus \{(0,0)\},$$

i.e., when condition $(6.28)_{/269}$ is valid.

This completes the proof. □

Since the subdivision process was assumed to be convergent, we have by Theorems $4.5.1_{/172}$ and $5.1.3_{/193}$ that

$$s(1,1) = 4, \quad s(1,-1) = s(-1,1) = s(-1,-1) = 0. \tag{6.31}$$

With $\hat{S}(\omega) = \sum_k s_k \exp(-i\omega^t k/2)$ denoting the Fourier transform in $(4.37)_{/167}$ we conclude that this is equivalent to

$$\hat{S}(2\pi,0) = \sum_k s_k(-1)^{k_1} = s(-1,1), \tag{6.32}$$

$$\hat{S}(0,2\pi) = \sum_k s_k(-1)^{k_2} = s(1,-1), \tag{6.33}$$

$$\hat{S}(2\pi,2\pi) = \sum_k s_k(-1)^{k_1}(-1)^{k_2} = s(-1,-1), \tag{6.34}$$

$$\hat{S}(0,0) = \sum_k s_k = s(1,1). \tag{6.35}$$

Using the fact that $\hat{S}(\omega)$ is doubly periodic with periods 4π we conclude that $(6.31)_{/271}$ is also equivalent to the condition that

$$\hat{S}(0) = 4 \text{ and } \hat{S}(2\pi l) = 0 \text{ for all } l \text{ with } l_1 \text{ or } l_2 \text{ odd}. \tag{6.36}$$

Further, from $(4.47)_{/168}$ it follows that any zero of $\hat{S}$ is a zero of $\hat{N}$. We claim that $\hat{N}(2\pi l) = 0$ for all $l \neq 0$. By $(6.36)_{/271}$ this is clear if l_1 or l_2 is odd. If both l_1 and l_2 are even, then for some $r > 0$, $(2^{-r}l_1, 2^{-r}l_2) \in \mathbb{Z}^2$ with $2^{-r}l_1$ or $2^{-r}l_2$ odd, and we conclude from $(4.46)_{/168}$ that $\hat{N}(2\pi l) = 0$. Consequently, for a convergent process, $(A.15)_{/312}$ and $(5.15)_{/194}$ in Theorem $5.1.3_{/193}$ imply that

$$\hat{N}(0) = \int_{\mathbb{R}^2} N(y)dy = 1,$$

and $\hat{N}$ must satisfy conditions $(6.27)_{/269}$, i.e.,

$$\hat{N}(0) = 1, \quad \hat{N}(2\pi l) = 0 \quad \text{for} \quad l \neq (0,0). \tag{6.37}$$

Now let us introduce the conditions A_d, $A_{(d_1,d_2)}$, B_d, and $B_{(d_1,d_2)}$ for a function $\hat{F}$ (below, $\hat{F}$ is typically $\hat{N}$ or $\hat{S}$). These conditions, which are additional to the basic

conditions $(6.31)_{/271}$–$(6.37)_{/271}$ guaranteed by convergence, permit characterization of various levels of precision, and this is the main goal of this section.

$$A_d\ (A_{(d_1,d_2)}): \quad \partial^r \hat{F}(2\pi l) = 0 \quad \text{for} \quad l \neq 0, \quad \text{and} \quad 0 < r, \quad |r| \leq d \quad (r \leq (d_1, d_2)),$$

$$B_d\ (B_{(d_1,d_2)}): \quad \partial^r \hat{F}(0) = 0 \quad \text{for} \quad 0 < r, \quad |r| \leq d \quad (r \leq (d_1, d_2)).$$

When we do not need to specify the particular degree or bidegree, we use the simpler notation A and B for these conditions.

We immediately get the following theorems.

Theorem 6.4.4. *The process has reproduction degree d if and only if the function $\hat{N}$ satisfies conditions A_d and B_d. It has reproduction bidegree (d_1, d_2) if and only if $\hat{N}$ satisfies conditions $A_{(d_1,d_2)}$ and $B_{(d_1,d_2)}$.*

Theorem 6.4.5. *The process has generation degree d if and only if $\hat{N}$ satisfies condition A_d. It has generation bidegree (d_1, d_2) if and only if $\hat{N}$ satisfies condition $A_{(d_1,d_2)}$.*

We now formulate some results that are slightly weaker than Theorems $6.4.4_{/272}$ and $6.4.5_{/272}$, involving only the function $\hat{S}(\omega)$. These results are useful when no explicit expression for $\hat{N}(\omega)$ is known.

By $(4.47)_{/168}$, i.e.,

$$\hat{N}(\omega) = \frac{\hat{S}(\omega)}{4}\hat{N}(\omega/2),$$

we get

$$\partial_1 \hat{N}(\omega) = \frac{\partial_1 \hat{S}(\omega)}{4}\hat{N}(\omega/2) + \frac{\hat{S}(\omega)}{4}\frac{1}{2}\partial_1 \hat{N}(\omega)|_{(\omega/2)}$$

and taking $\omega = 0$, we get

$$\partial_1 \hat{N}(0) = \frac{\partial_1 \hat{S}(0)}{4} + \frac{1}{2}\partial_1 \hat{N}(0).$$

So $\partial_1 \hat{N}(0) = 0$ if and only if $\partial_1 \hat{S}(0) = 0$. By repeated differentiation and induction one can prove that *$\hat{N}$ satisfies condition B if and only if $\hat{S}$ does.*

Next, for a convergent process we have $\hat{S}(2\pi l) = 0$ for all $l \in \mathbb{Z}^2$ with l_1 or l_2 odd. By repeated differentiation we conclude that if $\hat{S}$ satisfies the condition

$$A'_d\ (A'_{(d_1,d_2)}): \quad \partial^r \hat{S}(2\pi l) = 0 \text{ if } l_1 \text{ or } l_2 \text{ is odd and } 0 < r, |r| \leq d \ (r \leq d_1, d_2),$$

then $\hat{N}$ satisfies condition A_d $(A_{(d_1,d_2)})$. This gives the following corollaries.

Corollary 6.4.6. *The process has reproduction degree d if the function $\hat{S}$ satisfies conditions A'_d and B_d. It has reproduction bidegree (d_1, d_2) if $\hat{S}$ satisfies conditions $A'_{(d_1,d_2)}$ and $B_{(d_1,d_2)}$.*

Corollary 6.4.7. *The process has generation degree d if $\hat{S}$ satisfies condition A'_d. It has generation bidegree (d_1, d_2) if $\hat{S}$ satisfies condition $A'_{(d_1,d_2)}$.*

Because the corollaries give only sufficient conditions, they are weaker than Theorems 6.4.4$_{/272}$ and 6.4.5$_{/272}$. Exercise 1$_{/284}$ gives an example to illustrate this.

Before proceeding we reformulate conditions A, A', and B on $\hat{N}$ and $\hat{S}$. By Table A.2$_{/313}$, item 9 we have that

$$(i\partial)^k \hat{N}(\omega) = \int_{\mathbb{R}^2} N(y) y^k \exp(-i\omega^t y)\, dy.$$

Consequently,

$$\partial^k \hat{N}(0) = 0 \quad \text{if and only if} \quad \int_{\mathbb{R}^2} N(y) y^k\, dy = 0.$$

Similarly, since, from (4.37)$_{/167}$, $\hat{S}(\omega) = \sum_l s_l \exp(-i\omega^t l/2)$ we get

$$(i\partial)^k \hat{S}(\omega) = \sum_l s_l (l/2)^k \exp(-i\omega^t l/2) \tag{6.38}$$

and, using that

$$(z\partial)^k s(z) = \sum_l s_l l^k z^k \quad \text{and} \quad (z\partial)^k s(1,1) = \sum_l s_l l^k,$$

we get

$$\partial^k \hat{S}(0) = 0 \quad \text{if and only if} \quad (z\partial)^k s(1,1) = \sum_l s_l l^k = 0.$$

Thus, condition B on $\hat{N}$ may be reformulated as

B: $\int_{\mathbb{R}^2} N(y) y^k\, dy = 0$,

i.e., as moment conditions on N, and condition B on $\hat{S}$ as

B: $\sum_l s_l l^k = 0$,

i.e., as moment conditions on s. In the same way, if we take $\omega = 2\pi\nu$ in (6.38)$_{/273}$, we get

$$(i\partial)^k \hat{S}(2\pi\nu) = \sum_l s_l (l/2)^k \exp(-i\pi\nu^t l) = \frac{1}{2^{k_1+k_2}} \sum_l s_l l^k (-1)^{\nu_1 l_1} (-1)^{\nu_2 l_2},$$

and it follows that

$$(i\partial)^k \hat{S}(2\pi, 0) = \sum_l s_l (l/2)^k (-1)^{l_1} = (z\partial/2)^k s(-1,1), \tag{6.39}$$

$$(i\partial)^k \hat{S}(0, 2\pi) = \sum_l s_l (l/2)^k (-1)^{l_2} = (z\partial/2)^k s(1,-1), \tag{6.40}$$

$$(i\partial)^k \hat{S}(2\pi, 2\pi) = \sum_l s_l (l/2)^k (-1)^{l_1} (-1)^{l_2} = (z\partial/2)^k s(-1,-1), \tag{6.41}$$

$$(i\partial)^k \hat{S}(0, 0) = \sum_l s_l (l/2)^k = (z\partial/2)^k s(1,1). \tag{6.42}$$

This may be compared with (6.31)$_{/271}$–(6.35)$_{/271}$.

Further we note that, when $z \neq (0,0)$, $(z\partial)^k s(z) = 0$ for all $k \in \mathcal{I}_d$ ($k \in \mathcal{I}_{(d_1,d_2)}$) if and only if $\partial^k s(z) = 0$ for all $k \in \mathcal{I}_d$ ($k \in \mathcal{I}_{(d_1,d_2)}$). Consequently, for s and $\hat{S}$ conditions A' and B have the equivalent forms

A: $$(i\partial)^k \hat{S}(2\pi,0) = 0, \quad \partial^k s(-1,1) = 0, \quad (z\partial)^k s(-1,1) = 0,$$
$$(i\partial)^k \hat{S}(0,2\pi) = 0, \quad \partial^k s(1,-1) = 0, \quad (z\partial)^k s(1,-1) = 0,$$
$$(i\partial)^k \hat{S}(2\pi,2\pi) = 0, \quad \partial^k s(-1,-1) = 0, \quad (z\partial)^k s(-1,-1) = 0,$$

and

B: $$(i\partial)^k \hat{S}(0,0) = 0, \quad \partial^k s(1,1) = 0, \quad (z\partial)^k s(1,1) = 0.$$

We now look at some applications and examples.

6.4.2 Polynomial precision for box splines

Consider a general box spline with the (centered) subdivision polynomial

$$s(z) = 4 \prod_{i=1}^{m} \frac{z^{e_i/2} + z^{-e_i/2}}{2}.$$

From $(4.37)_{/167}$ the function $\hat{S}(\omega)$ is obtained by replacing z^l by $\exp(-i\omega^t l/2)$, and from Table $A.2_{/313}$, item 15, the function $\hat{N}(\omega)$ is given by

$$\hat{N}(\omega) = \prod_{i=1}^{m} \frac{\sin(\omega^t e_i/2)}{\omega^t e_i/2}. \tag{6.43}$$

(Exercise $2_{/285}$ asks for verification of item 15 of the table.) We now have $s(z) = \sum_l s_l z^l$ with $s_l \geq 0$, $s(1,1) = \sum_l s_l = 4$, and, because of the symmetry, $s_l = s_{-l}$. Further, from $(6.43)_{/274}$, $\hat{N}(\omega) = \hat{N}(-\omega)$ and therefore $N(y) = N(-y)$. Then obviously $\int_{\mathbb{R}^2} N(y) y^k \, dy = 0$ for $k = (1,0)$ and $k = (0,1)$. Consequently, condition B_1 is satisfied. On the other hand, by $(3.3)_{/95}$ we have $N(y) \geq 0$. Provided that $\alpha \leq m-1$, N is continuous and does not vanish. (Here α denotes the parameter of Definition $3.3.1_{/111}$, with the property that any subsequence of $e^m = \{e_i\}_{i=1}^m$ with α elements must span $\mathbb{R}^2$.) It follows that $\int_{\mathbb{R}^2} N(y) y^k \, dy > 0$ for $k = (2,0)$ and $k = (0,2)$ and that N does not satisfy condition B_d for any set $\mathcal{I}_d$ with $d \geq 2$. Therefore, by Theorem $6.4.6_{/272}$, *a box spline can have at most bilinear precision.* More precisely, we have the following result.

Theorem 6.4.8. *If $\alpha \leq m-1$, then the box-spline subdivision process has reproduction degree equal to* 1.

Proof. We have shown that condition B_1 is satisfied. We now show that under the hypotheses of the theorem, condition A_1 is also satisfied. Differentiating $(6.43)_{/274}$, we get

$$\partial_1 \hat{N}(\omega) = \sum_{j=1}^{m} \left(\prod_{1 \leq i \leq m, i \neq j} \frac{\sin(\omega^t e_i/2)}{\omega^t e_i/2} \right) \partial_1 \frac{\sin(\omega^t e_j/2)}{\omega^t e_j/2},$$

and taking $\omega = 2\pi l$,

$$\partial_1 \hat{N}(2\pi l) = \sum_{j=1}^{m} \left(\prod_{1\le i\le m, i\neq j} \frac{\sin(\pi l^t e_i)}{\pi l^t e_i} \right) \partial_1 \left(\frac{\sin(\omega^t e_j/2)}{\omega^t e_j/2} \right)\Big|_{\omega=2\pi l}.$$

If $l = 0$, then all second factors in this sum are equal to zero, and consequently $\partial_1 \hat{N}(0) = 0$. If $l \neq 0$, then by assumption for each j the vectors e_i, $1 \le i \le m$, $i \neq j$, span $\mathbb{R}^2$ and therefore there exists at least one vector e_i, $i \neq j$, such that $l^t e_i \neq 0$, i.e., so that

$$\prod_{1\le i\le m, i\neq j} \frac{\sin(\pi l^t e_i)}{\pi l^t e_i} = 0.$$

Consequently, $\partial_1 \hat{N}(2l\pi) = 0$ for all l. That $\partial_2 \hat{N}(2l\pi) = 0$ follows in the same way and we have verified conditions $(6.37)_{/271}$ and A_1. □

As in the preceding proof, it can be proved that condition $A_{m-\alpha}$ is satisfied, and therefore we also have the following result for the generation degree of a box spline.

Theorem 6.4.9. *The generation degree of a box-spline subdivision process is at least equal to* $m - \alpha$.

We recall now that, by Theorem $3.3.2_{/111}$, N is in $C^{m-\alpha-1}(\mathbb{R}^2)$.

Example 6.4.10. Zwart–Powell (Midedge) method.

Take $e_1 = (1,0)^t$, $e_2 = (0,1)^t$, $e_3 = (1,1)^t$, $e_4 = (-1,1)^t$. Then the centered subdivision polynomial is

$$s(z_1, z_2) = \frac{1}{z_1 z_2}(1+z_1)(1+z_2)(1+z_1 z_2)(1+z_1^{-1} z_2);$$

see $(3.58)_{/136}$. We have $\alpha = 2$ and $m = 4$, so by Theorem $6.4.8_{/274}$ the reproduction degree is equal to 1. Further,

$$\hat{N}(\omega_1, \omega_2) = \frac{\sin(\omega_1/2)}{\omega_1/2} \frac{\sin(\omega_2/2)}{\omega_2/2} \frac{\sin((\omega_1+\omega_2)/2)}{(\omega_1+\omega_2)/2} \frac{\sin((\omega_2-\omega_1)/2)}{(\omega_2-\omega_1)/2}.$$

Using that $\sin t/t = 1 - t^2/6 + \cdots$, we get the following power series expansion:

$$\begin{aligned}
&\hat{N}(\omega_1, \omega_2) \\
&= \Big(1 - \frac{\omega_1^2}{24} + \cdots\Big)\Big(1 - \frac{\omega_2^2}{24} + \cdots\Big)\Big(1 - \frac{(\omega_1+\omega_2)^2}{24} + \cdots\Big)\Big(1 - \frac{(\omega_1-\omega_2)^2}{24} + \cdots\Big) \\
&= 1 - \omega_1^2/8 - \omega_2^2/8 + \cdots,
\end{aligned}$$

where the dots denote terms of degree 4 or higher. Since terms $\omega_1\omega_2$ are missing in the expansion, we conclude that

$$\partial_1 \partial_2 \hat{N}(0,0) = 0,$$

i.e., condition $B_{(1,1)}$ is satisfied. Condition $A_{(1,1)}$ is also satisfied, since condition A_2 is satisfied. Therefore we have in addition bilinear precision, i.e., bilinear polynomials are reproduced. Since $m - \alpha = 2$, Theorem 6.4.9$_{/275}$ implies that the generation degree is at least 2. ∎

Further examples are considered in the exercises. Exercise 3$_{/285}$ discusses bivariate subdivision polynomials arising from tensor products, Exercise 4$_{/285}$ discusses the Loop method, and Exercise 5$_{/285}$ discusses constant subdivision. Exercise 6$_{/285}$ examines another box-spline method.

6.4.3 Polynomial precision for non-box-splines

The example to be presented now illustrates by means of the univariate four-point scheme that, if we are given a box spline, then it can be modified, by attaching some factor, so that for the new polynomial the reproduction degree becomes equal to the generation degree. This is interesting because box splines cannot produce more than bilinear precision, and if we want more, we must do something else. This example shows one way to do so.

Increasing precision by attaching a factor to the subdivision polynomial

Consider third-degree univariate splines with the subdivision polynomial

$$s(z) = 2\Big(\frac{z^{1/2} + z^{-1/2}}{2}\Big)^4 = (z^2 + 4z + 6 + 4z^{-1} + z^{-2})/8.$$

It follows by substitution into the middle expression (see (4.37)$_{/167}$) that $\hat{S}(\omega) = s(-\exp(i\omega/2)) = 2\cos^4(\omega/4)$. By Corollaries 6.4.6$_{/272}$ and 6.4.7$_{/272}$, the reproduction degree is 1 and the generation degree is 3. Now consider a polynomial

$$q(z) = a + b(z + z^{-1}) \quad \text{with} \quad q(1) = 1, \quad \text{i.e.,} \quad a + 2b = 1,$$

so that

$$\hat{Q}(\omega) = q(e^{i\omega/2}) = a + 2b\cos(\omega/2) \quad \text{and} \quad \hat{Q}(0) = 1.$$

Also, define the products

$$s_4(z) = s(z)q(z) \quad \text{and} \quad \hat{S}_4(\omega) = \hat{S}(\omega)\hat{Q}(\omega).$$

(The reason for the choice of subscript in s_4 will become clear shortly.)

Since $\hat{S}/2$ satisfies the conditions $\hat{S}(0) = 1$, $\hat{S}(2\pi l) = 0$ if l is odd, B_1, and A'_3, it is easy to see that $\hat{S}_4/2$ does too. We now try to choose the coefficients in q such that in addition $\hat{S}_4(\omega)$ satisfies B_3. Since $\hat{S}_4$ is an even function, all derivatives of odd order are zero at the origin, and we need only verify that $\partial^2\hat{S}_4(0) = 0$. We get

$$\partial^2\hat{S}_4(0) = \partial^2\hat{S}(0)\hat{Q}(0) + \partial^2\hat{Q}(0)\hat{S}(0) = 0,$$

and using that $\hat{S}(0) = 2$ and $\hat{Q}(0) = 1$, we get

$$\partial^2\hat{S}(0) + 2\partial^2\hat{Q}(0) = 0.$$

Since $\partial^2\hat{S}(0)$ is known, this uniquely determines $\partial^2\hat{Q}(0)$. Differentiating $\hat{S}(\omega)$ (or expanding it in power series), we get $\partial^2\hat{S}(0) = -1/2$, i.e., $\partial^2\hat{Q}(0) = 1/4$. Differentiating $\hat{Q}(\omega) = a + 2b\cos(\omega/2)$, we get

$$\partial^2\hat{Q}(0) = -b/2 = 1/4, \quad \text{i.e.,} \quad b = -1/2,$$

and, since $a + 2b = 1$, $a = 2$. Consequently,

$$\hat{Q}(\omega) = 2 - \cos(\omega/2) \quad \text{and} \quad q(z) = -z/2 + 2 - z^{-1}/2.$$

We have determined the coefficients so that the reproduction degree is equal to 3 for the process defined by

$$s_4(z) = (z^2+4z+6+4z^{-1}+z^{-2})(4-z-z^{-1})/16 = (-z^3+9z+16+9z^{-1}-z^{-3})/16.$$

This is the well-known four-point subdivision polynomial which is interpolating, since all terms with even powers vanish, except the constant term. It does not follow from our analysis that the process is convergent, but this can be proved. It follows that the four-point subdivision process has the property that, if we start with a control sequence sampled from some cubic polynomial, exactly the same polynomial curve is reproduced, and the refined sequences of control vectors all lie on the same curve, i.e., the four-point method has cubic precision. According to Exercise $3_{/285}$, it also follows that the Kobbelt method has bicubic precision in the regular case.

In [67] the univariate polynomials

$$s_m(z) = 2\Big(\frac{z^{1/2} + z^{-1/2}}{2}\Big)^m K_m(z),$$

where $K_m(z) = (-mz + (8 + 2m) - mz^{-1})/8$, are studied and it is shown that the precision is cubic for $m \geq 4$ and quadratic for $m = 3$. This can be derived in the same way as in the example above, which is exactly the case $m = 4$. Calculations similar to those above give the values $a = (8 + 2m)/8$ and $b = -m/8$. See Exercise $7_{/285}$.

We state the following theorem, valid also in the bivariate case.

Theorem 6.4.11. *Assume that we are given a subdivision polynomial $\psi(z)$ with reproduction and generation degree d_r and d_g, respectively, where $d_g > d_r$. Also assume that the Fourier transform $\hat{\Psi}(\omega)$ corresponding to $\psi(z)$ satisfies conditions A_{d_g} and B_{d_r}. Then there exists a polynomial $q(z)$ such that the product $s(z) = \psi(z)q(z)$ defines a subdivision process which, if it is convergent, has generation and reproduction degree equal to d_g.*

The proof, which parallels the analysis of the previous example, is omitted.

Increasing precision by preconvolution

We now describe a method to increase polynomial precision which differs from the one just presented. Assume that we have a process, defined by a subdivision polynomial, having generation degree d_g. Then according to Proposition 6.4.2$_{/269}$ we can, for a given polynomial $p \in \Pi_{d_g}$, find a polynomial $p' \in \Pi_{d_g}$ such that the sequence $p'(l)$ generates the original polynomial $p(y)$. Here we show a little more, namely that the inverse image p' can be determined by a simple convolution with a suitably chosen finite mask $\{q_l\}_{l\in\mathbb{Z}^2}$. Thus, instead of changing the subdivision polynomial, we only preprocess the initial sequence of control vectors by convolving it with a given mask. This is less time consuming than changing the subdivision polynomial since only the first step in the process is changed.

Assume that we are given a subdivision polynomial with generation degree d_g and reproduction degree $d_r < d_g$. Also assume that an initial sequence of control vectors $\{p_l\}_{\mathbb{Z}^2}$ is sampled uniformly in $\mathbb{Z}^2$ from some polynomial $p \in \Pi_{d_g}$, i.e., $p_l = p(l)$. We show that if we preconvolve $\{p_l\}_{l\in\mathbb{Z}^2}$ by some suitably chosen finite sequence $\{q_k\}_{k\in\mathbb{Z}^2}$, i.e., if we replace[44] the sequence $\{p_l\}_{l\in\mathbb{Z}^2}$ by $\{p'_l\}_{l\in\mathbb{Z}^2}$ where

$$p'_l = \sum_k q_k p_{l-k}, \tag{6.44}$$

then the subdivision applied to the new sequence will produce the polynomial $p(y)$ in the limit.

We note that the new control points are sampled from the polynomial $p'(y) = \sum_k q_k p(y-k)$ which is of degree less than or equal to d_g. Next, we use that the limiting surface has the parametric representation

$$\sum_l p_l N(y-l) = \sum_l p_l z^{2l} N(y) = p_g(z^2)N(y),$$

where we have introduced the generating function $p_g(z) = \sum_l p_l z^l$ for the initial sequence of control vectors. The limiting surface given by the modified sequence is similarly

$$\sum_l p'_l N(y-l) = \sum_l p'_l z^{2l} N(y) = p'_g(z^2)N(y),$$

where $p'_g(z) = \sum_l p'_l z^l$. Now (6.44)$_{/278}$ is equivalent to $p'_g(z) = q(z)p_g(z)$, and the parametric surface $y \mapsto \sum_l p'_l N(y-l)$ is given by

$$p'_g(z^2)N(y) = p_g(z^2)q(z^2)N(y) = p_g(z^2)\left(\sum_k q_k N(y-k)\right).$$

Thus, the preconvolution is equivalent to replacing $N(y)$ by $\sum_k q_k N(y-k)$, or equivalently $\hat{N}(\omega)$ by

$$\sum_k q_k \exp(-i\omega^t k)\hat{N}(\omega) = \hat{Q}(2\omega)\hat{N}(\omega),$$

where $\hat{Q}(\omega) = \sum_k q_k \exp(-i\omega^t k/2)$. See Table A.1$_{/311}$, item 5.

By Theorem 6.4.4$_{/272}$, it suffices to prove that $\hat{Q}(2\omega)\hat{N}(\omega)$ satisfies conditions (6.37)$_{/271}$, A_{d_g}, and B_{d_g}.

By assumption $\hat{N}(\omega)$ satisfies conditions (6.37)$_{/271}$ and A_{d_g}. If $\hat{Q}(0) = 1$, it follows immediately that $\hat{Q}(2\omega)\hat{N}(\omega)$ also satisfies (6.37)$_{/271}$ and A_{d_g}. Now, the condition that $\hat{Q}(2\omega)\hat{N}(\omega)$ should satisfy B_{d_g} gives conditions on the derivatives of $\hat{Q}$ at the origin, and a trigonometric polynomial solution can always be found. The details are quite similar to those in the analysis of the four-point scheme, above, and are omitted. We also note that a similar result is valid for the bidegree case when (d_{g_1}, d_{g_2}) and (d_{r_1}, d_{r_2}) are the generation and reproduction bidegrees and $d_{g_1} + d_{g_2} > d_{r_1} + d_{r_2}$.

Example 6.4.12. Preconvolution using a mask.

Consider again univariate cubic splines, with

$$\hat{N}(\omega) = \left(\frac{\sin(\omega/2)}{\omega/2}\right)^4.$$

We again want to find a polynomial $q(z) = a + b(z + z^{-1})$ such that $\hat{Q}(\omega) = q(e^{i\omega/2}) = a + 2b\cos(\omega/2)$ satisfies $\hat{Q}(0) = 1$ and condition B_2:

$$\partial^2(\hat{Q}(2\omega)\hat{N}(\omega)|_{\omega=0} = 0. \tag{6.45}$$

(Condition A_2 is satisfied, since $\partial^2\hat{N}(2\pi l) = 0$ for $l \neq 0$ and $0 \leq r \leq 2$. Also, since $\hat{Q}(\omega)$ and $\hat{N}(\omega)$ are even functions, the derivatives vanish at the origin, which means that if we find the polynomial $q(z)$ we seek, we will have cubic precision.)

Using that $\frac{\sin t}{t} = 1 - t^2/6 + \cdots$, we get

$$\begin{aligned}
\hat{Q}(2\omega)\hat{N}(\omega) &= (a + 2b\cos\omega)\left(\frac{\sin(\omega/2)}{\omega/2}\right)^4 \\
&= \left(a + 2b - (2b)\frac{\omega^2}{2} + \cdots\right)\left(1 - \frac{(\omega/2)^2}{6} + \cdots\right)^4 \\
&= (1 - b\omega^2 + \cdots)(1 - \omega^2/24 + \cdots)^4 \\
&= (1 - b\omega^2 + \cdots)(1 - \omega^2/6 + \cdots) \\
&= 1 - \omega^2(b + 1/6) + \cdots.
\end{aligned}$$

Consequently, (6.45)$_{/279}$ is satisfied if and only if $b = -1/6$, $a = 4/3$, i.e., $q(z) = (8 - z - z^{-1})/6$. This gives the convolution mask

$$[-1/6,\ \ 4/3,\ \ -1/6],$$

which produces cubic precision. Thus, if we apply the mask to the sequence $\{l^2\}_{l\in\mathbb{Z}}$, we get

$$(4/3)l^2 - (1/6)\{(l+1)^2 + (l-1)^2\} = l^2 - 1/3,$$

so

$$\sum_{l\in\mathbb{Z}}(l^2 - 1/3)N^4(t-l) = t^2.$$

Similarly, if the mask is applied to the sequence $\{l^3\}_{l\in\mathbb{Z}}$, we get

$$(4/3)l^3 - (1/6)\{(l+1)^3 + (l-1)^3\} = l^3 - l,$$

and so

$$\sum_{l\in\mathbb{Z}}(l^3 - l)N^4(t-l) = t^3.$$

This example is taken from [67]. ∎

6.5 Bounding envelopes for patches

In this section we describe a method of Wu and Peters [169, 170] for finding tight bounding envelopes for surface patches. This method can be viewed as an extension and improvement of an earlier method proposed by Kobbelt [75]. Like the method of [18], it uses pretabulated basis functions, although for a different purpose. Also, it makes use of an extension, to the nonregular case, of the concept of linear precision. The method is described in the context of Loop subdivision, but the same approach can be used for other subdivision methods.

The bounding envelopes discussed here have application in collision detection, and also for adaptive subdivision, which is discussed in Section 6.6.

According to $(4.74)_{/181}$, the spline surface can be expressed as

$$x(y) = \sum_{\ell} p_\ell N_\ell(y), \tag{6.46}$$

where ℓ ranges over all vertices in the logical mesh, and y is in some appropriately defined two-dimensional manifold $\boldsymbol{M}$. The functions $N_\ell(y)$ are defined on $\boldsymbol{M}$ with their supports located in some neighbourhood of the corresponding point $[\ell] \in \boldsymbol{M}$.

In the case of the Loop subdivision method, in a neighbourhood of an isolated extraordinary vertex of valence n the manifold $\boldsymbol{M}$ can be defined locally to be a portion of the plane, laid out according to the illustrations in Figure $6.4_{/281}$. (The vertex labelled [0] is the vertex of valence n, and the direct neighbours of this vertex form a regular unit n-gon; the vertices [1] and [2] are the other vertices of a typical triangle incident at the extraordinary vertex. The case $n = 5 < 6$ is illustrated in Figure $6.4_{/281}$ (left), and the case $n = 7 > 6$ is illustrated in Figure $6.4_{/281}$ (right). The regular case $n = 6$ is also permitted. More details on the geometric layout are given below.) There are $n + 6$ vertices indicated in the examples in the figure, and these are exactly the vertices for which the support of the centered basis function $N_\ell(y)$ extends to the centre triangle [0]-[1]-[2], which we denote by T. The sum in $(6.46)_{/280}$ can therefore be viewed as running over $n + 6$ vertices and defining a patch corresponding to the triangle T. In the case $n = 6$, this corresponds to the

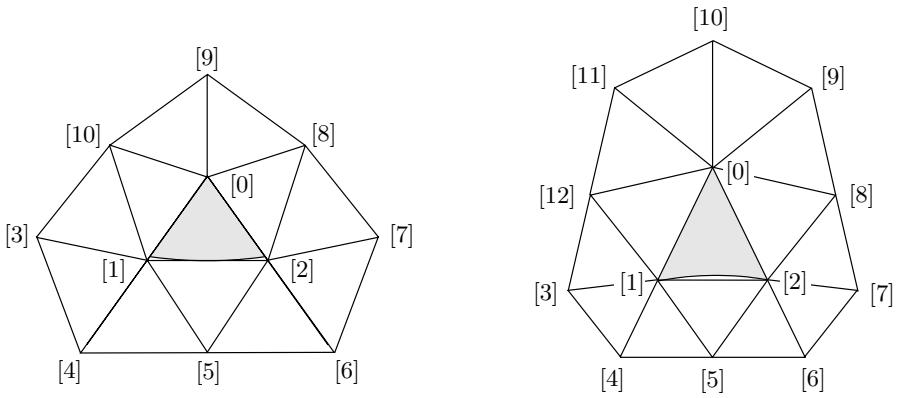


Figure 6.4. *Geometric layout of parametric domain.*

patch illustrated in Figure $3.11_{/111}$. To simplify the presentation, we refer to the logical vertices $[\ell]$ embedded in the plane by the integers $0, 1, 2, \ldots, n+5$. The $n+6$ nodes in parameter space are denoted by $y_i = (u_i, v_i)$ for $i = 0, 1, \ldots, n+5$, the nodal functions centered around y_i are denoted by N_i, and the control vectors by p_i, so that we have the parametric surface representation

$$x(y) = x(u, v) = \sum_{i=0}^{n+5} p_i N_i(u, v) = \sum_{i=0}^{n+5} p_i N_i(y)$$

when $y = (u, v)$ is in the triangle T.

6.5.1 Modified nodal functions

In an infinite regular grid with equilateral triangles and all nodes of valence 6, the nodal functions N_i have the property that they reproduce linear functions, which means that

$$l(y) = \sum_i l(y_i) N_i(y) \tag{6.47}$$

for all linear functions $l : \mathbb{R}^2 \to \mathbb{R}$ (see Section 6.4). However, for a nonregular finite grid, the equality $(6.47)_{/281}$ is no longer valid. To apply the techniques of Wu–Peters we need to introduce auxiliary nodal functions $b_i(u, v)$ having such a linear precision property. To simplify the presentation, we make the temporary assumption that the initial grid is extended to an infinite grid in $\mathbb{R}^2$ and that we wish to define the functions b_i so that

$$l(y) = \sum_i l(y_i) b_i(y) \tag{6.48}$$

for all linear functions $l : \mathbb{R}^2 \to \mathbb{R}$.

The functions b_i are defined in the following way. For a fixed i, $0 \le i \le n+5$, we consider the parametric surface

$$\mathbb{R}^2 \ni y' \mapsto \sum_{j \neq i} (y_j, 0) N_j(y') + (y_i, 1) N_i(y') \in \mathbb{R}^3, \tag{6.49}$$

which is the result of applying Loop subdivision to a set of control vectors $(u_j, v_j, x_j) \in \mathbb{R}^3$, $x_i = 1$, and $x_j = 0$ if $j \neq i$. This representation may be rewritten as

$$\mathbb{R}^2 \ni y' \mapsto (\Pi(y'), N_i(y')) \in \mathbb{R}^3, \tag{6.50}$$

where $\Pi(y') = \sum y_j N_j(y')$. Now, if we assume that the location of the nodal points have been chosen so that the mapping $\Pi : \mathbb{R}^2 \to \mathbb{R}^2$ is invertible (this assumption is relaxed below), we may define

$$b_i(y) = N_i(\Pi^{-1}(y)), \quad y = \Pi y',$$

so that the surface in $(6.50)_{/282}$ has the representation $w = b_i(u, v)$ in an orthogonal (u, v, w)-coordinate system, or more compactly, $w = b_i(y)$. We also have $N_i(y') = b_i(\Pi(y'))$. Further, any surface with a representation $x(y') = \sum_i p_i N_i(y')$ is the same surface as that represented by $x(y) = \sum_i p_i b_i(y)$.

For the functions b_i, we now have the crucial linear-precision property $(6.48)_{/281}$. Assume that we start with an initial set of control vectors $(y_j, c_j) = (u_j, v_j, c_j) = p_j^0 \in \mathbb{R}^3$, all lying in some plane with the equation $w = l(y)$, so that $c_j = l(y_j)$. After one subdivision we obtain a refined set of control vectors p_j^ν which, by the affine invariance, must be located in that same plane. (See Exercise $8_{/285}$.) The same is true at all refinement levels and therefore the limiting surface patch lies in the same plane. Using the definition of the functions b_i and linear superposition, we conclude that in an orthogonal (y, w)-coordinate system, the equation of that plane is $w = \sum_i c_i b_i(y)$ and, since $c_i = l(y_i)$, that $l(y) = \sum_i l(y_i) b_i(y)$.

Remark 6.5.1. All the nodal functions $N_i : \boldsymbol{M} \to \mathbb{R}$ are well-defined functions on the two-dimensional manifold $\boldsymbol{M}$, whereas the functions b_i are not. These functions depend on the location of the nodes in parameter space $\mathbb{R}^2$: if we choose different locations, the functions b_i will change. ∎

6.5.2 Bounding linear functions

For each coordinate of $x = x(y') = \sum_i p_i N_i(y')$, two linear functions are computed defining a lower and an upper bound over the triangle T. For component j of x we write

$$\lambda^j(y') \le x^j(y') \le \Lambda^j(y') \quad \text{if} \quad y' \in T.$$

The error in the component is $e^j = \Lambda^j - \lambda^j$, and the total error for the subdivision patch (used, for example, to guide an adaptive subdivision process) is the maximal value of the norm of the vector $e \in \mathbb{R}^N$. Since the analysis is the same for each

component, we drop the superscript j and denote a typical component by $x = x(y')$. We then have, for $y' \in T$,

$$x(y') = \sum_{i=0}^{n+5} c_i N_i(y') = \sum_{i=0}^{n+5} c_i N_i(\Pi^{-1}(y)) = \sum_{i=0}^{n+5} c_i b_i(y), \tag{6.51}$$

where c_i is the corresponding scalar component of p_i and $y' = \Pi^{-1}(y)$.

If we let $l(y)$ be the linear function that interpolates the three points (y_0, c_0), (y_1, c_1), and (y_2, c_2), then we can rewrite $(6.51)_{/283}$ as

$$x(y) = l(y) + \sum_{i=3}^{n+5} d_i b_i(y) \quad \text{for} \quad y \in \Pi(T),$$

where $d_i = c_i - l(y_i)$. We then have the upper and lower bounds

$$\lambda(y) = l(y) + \sum_{i=3}^{n+5} (\max\{d_i, 0\} b_i^+(y) + \min\{d_i, 0\} b^-(y)),$$
$$\Lambda(y) = l(y) + \sum_{i=3}^{n+5} (\max\{d_i, 0\} b_i^-(y) + \min\{d_i, 0\} b^+(y)),$$

where b_i^+ and b_i^- are precomputed linear upper and lower bounds for the basis functions $b_i(y)$. These bounds must be valid when $y = \Pi(y')$ and $y' \in T$, i.e., for $y \in \Pi(T)$. Therefore, we must require that the domain of Π contain the triangle T, that $\Pi|_T$ is one-to-one, and that $T \subset \Pi(T)$, or equivalently, $\Pi^{-1}(T) \subset T$.

These properties can be verified provided that the locations in the parameter space of the points y_j, $j = 0, 1, \ldots, n+5$, are carefully chosen. Some details are given in [169, 170]. The set $\Pi^{-1}(T) \subset T$ is shaded in Figure $6.4_{/281}$ and is denoted by Ω_n in [169, 170]. We also note that the subpatch corresponding to the triangle T is defined completely by the expression

$$x(y') = \sum_{j=0}^{n+5} c_j N_j(y') = l(y) + \sum_{j=3}^{n+5} d_j b_j(y)$$

for $y' \in T$ and $y = \Pi y' \in \Pi(T)$. Therefore, only the restrictions $N_j|_T$ and $b_j|_{\Omega_n}$, $j = 3, 4, \ldots, n+5$, are needed when computing $\lambda(y)$ and $\Lambda(y)$, and there is no need to introduce the infinite grid above, nor to define the functions b_i everywhere.

If the locations of the points y_j, $j = 0, 1, \ldots, n+5$, are chosen so that $T \not\subset \Pi(T)$, then the functions b_i to be used are not well defined in Ω_n using only information from nodes $0, 1, \ldots, n+5$ and additional information must be supplied. One may, for example, extend the domains of b_j by introducing additional nodal points in parameter space. See [170, Sec. 3].

The details of the geometric form of the parametric grids are given in [169, 170] and in the statement of Exercise $9_{/286}$. These include the definition of the constant k_n, which is the length of the segment y_1-y_4. The exercise asks for a proof of the fact that for $n \geq 6$, Loop subdivision maps y_i into y_i, $i = 0, 1, 2$.

6.6 Adaptive subdivision

One application of the bounding envelopes discussed in Section 6.5 is *adaptive subdivision*. As observed in Section 1.1, time and memory requirements for subdivision methods increase exponentially if subdivision is applied uniformly across the mesh, and it is therefore of interest to vary the depth of subdivision depending on the accuracy of the surface approximation in different parts of the mesh.

To exploit this idea, several problems must be solved. First of all, faces from different levels of refinement have to be joined together in a consistent way, to obtain a conformal mesh. For example, if the splitting of one face involves an edge which is shared with another face that is not split, then a crack may appear between the two corresponding limit patches. This is a well-known problem in the general context of meshing, and it is also mentioned often in the specific context of subdivision-surface methods. For example, the advantages of 4-8 subdivision and $\sqrt{3}$-subdivision in this respect are discussed in [164, Sec. 2.2.3] and [76, Sec. 4], respectively, and the problem is discussed in the context of the bounds in Section 6.5, above, in [170, Sec. 4]. See also [20, 180]. Another problem is that, unless the method is interpolating, the geometric location of a control point in the refined mesh is not well defined if faces from different refinement levels share the vertex [76, Sec. 4].

A remaining problem is the elaboration of measures to use as a stopping criterion, when subdividing a certain part of the mesh. Various measures of error, and various flatness criteria, have been proposed for different methods [27, 68, 69, 94, 103, 104, 105, 133, 134]. The bounds in Section 6.5 have been used to control adaptive subdivision for the Loop method (see [120] and [170, Sec.4]).

6.7 Additional comments

The use of evaluation and tangent stencils is a very widely used technique. It is often referred to using the terminology "evaluation and tangent masks," or "pushing points to the limit."

The method of de Boor [35], [38, p. 17], described in Section 6.3.1, has been rediscovered in more recent papers. Stam's method [150] is important in graphics applications. See [116] for the use of these and other methods on graphics cards.

The methods related to polynomial precision are widely known in the wavelet community, but less well described in the subdivision-surface literature. Some recent papers in the latter category are [28, 67, 86].

The question of estimating the appropriate depth (number of iterations) for subdivision has recently received much attention, with many heuristic methods (based, for example, on "flatness" criteria) suggested, as mentioned in Section 6.6.

6.8 Exercises

1. Consider the univariate subdivision process with centered subdivision polynomial
$$s(z) = 2\left(\frac{z^{-1/2} + z^{1/2}}{2}\right)\left(\frac{z^{-1} + z}{2}\right).$$

Show that for this method we have $\hat{S}(\omega) = 2\cos(\omega/4)\cos(\omega/2)$ and $\hat{N}(\omega) = f(\omega/2)f(\omega)$, where $f(\omega) = \sin(\omega)/\omega$, and that $\hat{S}$ satisfies condition B_1, which means that $\hat{N}$ does too. Show also that $\hat{N}$ satisfies condition A_1, so that by Theorem 6.4.4$_{/272}$ the reproduction degree is equal to 1.

Show, however, that $\hat{S}$ does not satisfy A_1, which means that Corollary 6.4.6$_{/272}$ misses the fact that the method has linear precision.

2. Using (A.26)$_{/316}$ and item 5 of Table A.2$_{/313}$, verify item 15 of the table.

3. Suppose that the nodal functions for two univariate subdivision processes satisfy conditions A_{d_1} and A_{d_2}, respectively. Show that the nodal function for the corresponding bivariate tensor-product process satisfies condition $A_{(d_1,d_2)}$. Similarly, show that the corresponding statement for condition B is also true. (This is relevant in the context of Theorems 6.4.4$_{/272}$ and 6.4.5$_{/272}$ for methods such as tensor-product B-splines and the Kobbelt method.)

4. Using Theorem 6.4.4$_{/272}$ show that the Loop method does not have bilinear precision.

5. (a) Constant subdivision clearly does not have linear precision: such a method cannot reproduce all linear functions and cannot have reproduction degree equal to 1. State why this does not contradict the statement of Theorem 6.4.8$_{/274}$, even though constant subdivision is a box-spline subdivision process. (Similarly, this fact does not contradict the statement of Corollary 6.4.6$_{/272}$ even though condition B_1 is satisfied, since the other hypothesis of the corollary, condition A_1, is not satisfied.)

   (b) Show that condition B_1 is satisfied, that condition A_1 is not satisfied, and show exactly where the proof of Theorem 6.4.8$_{/274}$ fails in the case of constant subdivision.

6. Consider the box-spline method defined by $e_1 = (1,0)^t$, $e_2 = (0,1)^t$, $e_3 = (1,1)^t$, with centered subdivision polynomial $\frac{1}{2z_1z_2}(1+z_1)(1+z_2)(1+z_1z_2)$. Show, using the results of this chapter, that the method has linear precision, but does not have bilinear precision.

7. Consider the subdivision polynomial

$$s_m(z) = 2\left(\frac{z^{1/2} + z^{-1/2}}{2}\right)^m q(z)$$

   where $q(z) = a + b(z + z^{-1})$. Proceeding as in Section 6.4.3 in the book, show that the choice $a = (8+2m)/8$ and $b = -m/8$ gives quadratic precision for $m = 3$ and cubic precision for $m \geq 4$.

8. Show that if an affine-invariant subdivision method is applied to the vertices (u_i, v_i) of the parametric plane, embedded in $\mathbb{R}^2$, at the same time as it is

applied to the control points p_i, as described in Section 6.5, then linearity is preserved even in the case of nonregular meshes. That is, if at step ν of the process we have that $p_i^\nu = \alpha u_i^\nu + \beta v_i^\nu + \gamma$ for each control point i, then at step $\nu + 1$ we will have $p_k^{\nu+1} = \alpha u_k^{\nu+1} + \beta v_k^{\nu+1} + \gamma$ for each control point k in the refined mesh.

9. The layout of the points in Figure 6.4$_{/281}$ is as follows [169, 170]:

   (a) Set $y_0 = 0$, the origin of the (u, v) plane.

   (b) Choose the direct neighbours y_i of y_0 to form a regular unit n-gon.

   (c) Extend the edge y_0-y_1 and y_0-y_2 by k_n to get y_4 and y_6.

   (d) Choose y_5 to be the average of y_4 and y_6.

   (e) Choose y_3 and y_7 to be the reflection of y_5 across y_0-y_4 and y_0-y_6, respectively.

   The constant k_n is defined by

$$k_n = \begin{cases} -4(c^2-2)/(1+2c^2) - 1, & n \geq 6, \\ -6(2c^2-7)/(15+2c^2) - 1, & n < 6, \end{cases}$$

   where $c = \cos \pi/n$.

   Show that in the case $n \geq 6$, y_i is mapped by Loop subdivision into y_i, $i = 0, 1, 2$.

   Similarly, for $3 \leq n \leq 5$, let y_m be the edge point between y_1 and y_2 computed after one step of Loop subdivision: $y_m = \frac{3}{8}(y_1 + y_2) + \frac{1}{8}(y_0 + y_5)$. Show that the limit position of y_m is $\frac{1}{2}(y_1 + y_2)$.

6.9 Projects

1. *Implementation of Stam's method.*

   Obtain the papers [149, 150] and implement the method of exact evaluation of Catmull–Clark or Loop surfaces.

2. *Implementation of the method of Wu–Peters.*

   Implement the method described in Section 6.5 and compare it with the strategy of omitting the linear function $l(u, v)$, i.e., taking $l(u, v) \equiv 0$.

Chapter 7

Shape Control

There has been considerable research over the last two decades on methods for shape control, in order to design tools more powerful than simple trial-and-error interaction with the user. Of fundamental importance is the treatment of boundaries, and the definition of crease and sharp edges on the boundary or in the interior of the surface. These topics are discussed in the first two sections of this chapter. Primal methods have achieved more prominence than dual methods in this context, but some of the earliest work [106] was related to the Doo–Sabin method, a dual method. We discuss the primal case first, and then we give an example for the dual case.

Another important problem is surface fitting (again, both in the interior and boundary cases), including the problem of interpolation of points and curves subject to various continuity requirements. There is a large number of cases to deal with: an overview is given in the papers [114, 115]. Here we only give some examples to illustrate the main ideas.

The final section of the chapter gives further references for surface fitting, and references for further reading in areas related to shape control that are not covered in detail in the book, including the topic of free-form editing [175].

A major approach to free-form editing is multiresolution editing and, although we do not present a complete description, we give some detail on this topic. One way to obtain multiresolution is wavelet decomposition, and this provides an interesting link to the theory of wavelets [33, 156]. We also mention two other methods for multiresolution editing, one based on a smoothing approach, and another which is similar but outside the field of subdivision surfaces.

7.1 Shape control for primal methods

We first discuss methods to model sharp features such as crease edges, darts, corners, and boundary edges, and then we consider the question of interpolation.

7.1.1 Surface boundaries and sharp edges

An important early paper [66] on this topic was written by eight researchers at the University of Washington, and the work was extended in [144]. For convenience we refer to these two papers as the *Seattle work*: this work describes the basic ideas necessary to deal with sharp features in the case of Loop subdivision. We present these ideas, and similar ideas for the Catmull–Clark method, followed in Section 7.1.2 by modifications that rectify deficiencies in the original methods.

We begin by defining the terms introduced above and by describing a preliminary set of crease-edge rules for the two primal methods most commonly used in practice, the Loop and Catmull–Clark methods. For the Loop method we may use the rules described in the Seattle work. For the Catmull–Clark method, a natural approach to crease and boundary edges is simply to apply the degree-three curve algorithm $LR(3)$ along the boundary or crease, as outlined for example in [95]. These rules for Catmull–Clark can also be viewed as deriving from the Seattle work [42, Appendix A]. We call these two sets of rules "preliminary" because modified rules were later proposed in [15], to deal with certain shortcomings of the original rules. The modified rules are presented later in the section.

The first step is to identify a subset of the edges $\{(\ell, \ell')\}$ in the logical mesh M as *tagged edges*. (This identification is done interactively by the user of the system.) Tagged edges are also called *sharp edges*, or *crease edges*, in the literature: the three terms are synonymous, but from now on, we always use the term *sharp edge*.[45] All boundary edges, in a mesh with boundary, are sharp. In addition, other edges may be sharp, at the discretion of the user. Edges that are not sharp are called *smooth*.

In addition to *sharp edges*, certain vertices in the mesh are tagged as non-smooth. The set of tagged vertices is completely determined by the set of sharp edges, but with some schemes, the user may also be able to use interactive tagging to change the type of a tagged vertex (for example, tagging a vertex with only two incident sharp edges as a corner, when normally three would be required).

Sharp edges for Loop subdivision

The Loop subdivision method produces tangent-plane-continuous surfaces [91] (see Definition $5.6.1_{/234}$), but in the Seattle work, modified subdivision rules were introduced that relaxed the requirement of tangent-plane continuity across sharp edges, while maintaining a well-defined tangent plane on each side of the crease. Similar considerations apply to corners and darts, which are special types of vertices that are defined now.

As mentioned, first a subset of edges in the logical mesh is tagged as sharp. This subset must include all boundary edges in a mesh with boundary. Then, vertices in the logical mesh are classified [66] as being of one of five types, including two types of a crease vertex.

- *smooth* vertex (0 incident sharp edges);
- *dart* vertex (1 incident sharp edge);

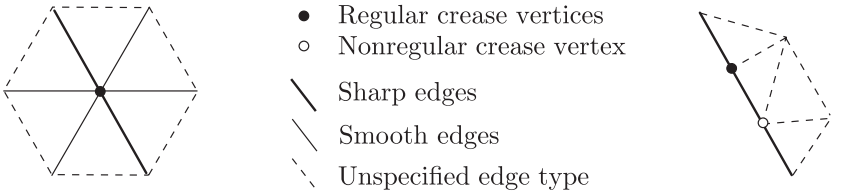


Figure 7.1. *Regular and nonregular crease vertices (Loop method).*

Table 7.1. *Type of sharp-edge stencil, given the types of incident vertices.*

Vertex types ↓ →	dart	regular crease	nonregular crease	corner
dart	1	1	1	1
regular crease	1	2	3	3
nonregular crease	1	3	2	2
corner	1	3	2	2

- *crease* vertex (2 incident sharp edges);
  - *regular crease* vertex;
  - *nonregular crease* vertex;
- *corner* vertex (3 or more incident sharp edges).

The nonsmooth vertices are said to be *tagged*; the tagged vertices, and their types, are determined by the set of sharp edges. In the list just given, crease vertices are split into two subcategories, regular and nonregular. An interior crease vertex is defined as *regular* if it has valence 6 with exactly two smooth edges on each side of the sharp edge; a boundary crease vertex is *regular* if it has valence 4. All other crease vertices, whether interior or boundary, are *nonregular*. Regular and nonregular crease vertices are illustrated in Figure $7.1_{/289}$. It follows from the definition of a smooth vertex that both vertices ℓ and ℓ' incident at a sharp edge (ℓ, ℓ') are nonsmooth vertices.

The stencil shown in Figure $1.29_{/32}$ (left) is modified, for sharp edges, as illustrated in Figure $7.2_{/290}$. The first stencil (Edge stencil 1) shown in Figure $7.2_{/290}$ is for the case of a sharp edge with one or both incident vertices a dart vertex; in this case the stencil used is the same as the one for a smooth edge. Edge stencil 2 and Edge stencil 3 treat the other cases of incident nonsmooth vertices. The stencils used are defined in Table $7.1_{/289}$. In Figure $7.2_{/290}$ and subsequent figures, we do not distinguish carefully between smooth edges and edges of unspecified type.

The vertex stencil shown in Figure $1.29_{/32}$ (right) is also modified, as illustrated in Figure $7.3_{/290}$, where heavy lines again denote sharp edges. The first stencil shown in Figure $7.3_{/290}$ is the case of a dart vertex, where the stencil is the same as the one for a smooth vertex. The second and third stencils treat the other cases of tagged vertices. Continuity of the resulting surfaces is discussed in [144]. See also [172, Sec. 6.7].

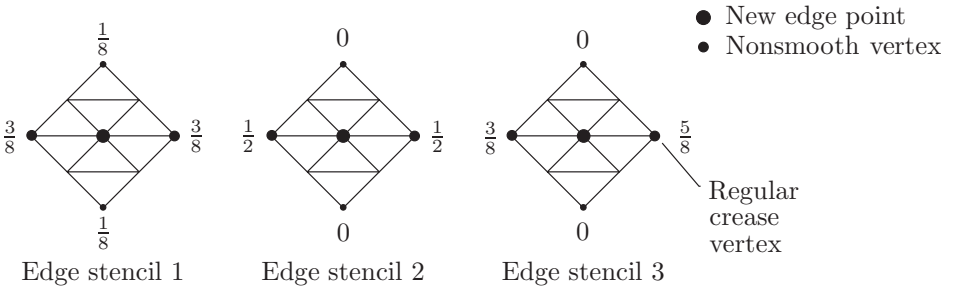


Figure 7.2. *Stencils for sharp edges (Loop method).*

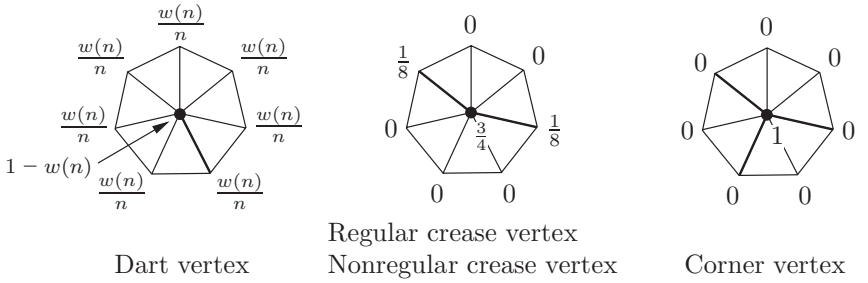


Figure 7.3. *Stencils for nonsmooth vertices (Loop method).*

Evaluation and tangent stencils (see Section 6.1) were also given in [66] for crease vertices.

The vertex rules in Figure 7.3$_{/290}$ are designed so that sharp features converge to uniform B-splines except near nonregular crease and corner vertices. The zeroes in these stencils completely decouple the behaviour of the surface on the two sides of a sharp edge. On the other hand, note that the behaviour of the surface at the boundary depends on the number of internal points incident to a boundary vertex (Figure 7.1$_{/289}$, right), and clearly this may lead to cracks if we attempt, after subdivision, to join two surfaces that originally had a common boundary [15]. Further, when we refer to the "side" of a sharp edge, we mean topologically (within the topological mesh), but this does not decouple the two *geometric* sides: consequently, undesirable folding may occur. These two shortcomings are remedied in Section 7.1.2. (Exercise 1$_{/304}$ gives a simple example to illustrate the possibility of a surface folding back on itself if the preliminary Loop rules are used. It is worth looking at this exercise before reading Section 7.1.2.)

Sharp edges for Catmull–Clark subdivision

The idea behind the rules for sharp edges for the Catmull–Clark method [42, Appendix A], [95] is to use the $LR(3)$ algorithm for curves along sharp edges. This includes, as always, all boundary edges.

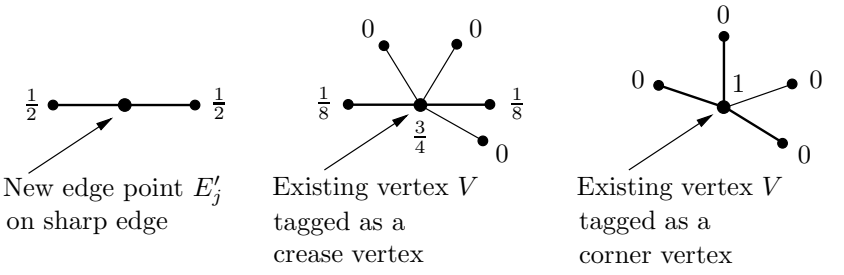


Figure 7.4. *Modified stencils for certain tagged vertices (Catmull–Clark method).*

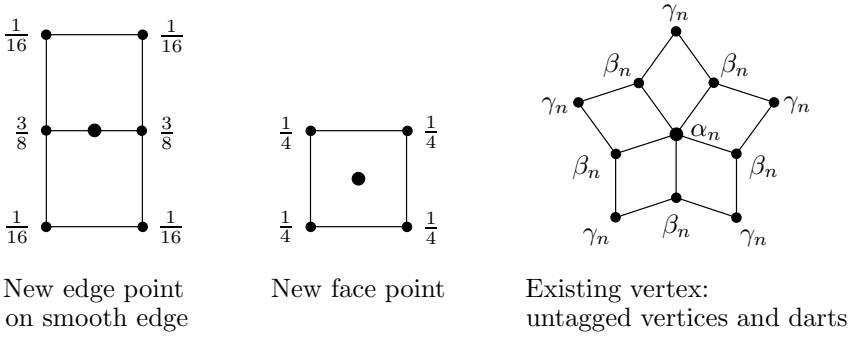


Figure 7.5. *Standard stencils for Catmull–Clark.*

Recall from Figure $1.27_{/29}$ that the Catmull–Clark method produces, at each step, new edge points E'_i and new face points F'_i, $i = 0, \ldots, n-1$, as well as a modified value V' of a control point V. From $(1.16)_{/43}$ the use of $LR(3)$ implies that the stencils shown in Figure $7.4_{/291}$ (left and middle) should be used along sharp edges, which are shown by heavy lines. In addition, an existing vertex may be of type corner, in which case the control point is interpolated (Figure $7.4_{/291}$, right). Apart from these cases, the standard Catmull–Clark rules are used.

In [42] the Catmull–Clark rules are presented using the formulation of $(A.8)_{/309}$. Sometimes, however, these rules are presented using the Ball–Storry formulation given in $(A.5)_{/308}$ and $(A.6)_{/309}$ (see [95]). In this case, the stencils for the standard rules are shown in Figure $7.5_{/291}$. These stencils should be compared with those shown in Figure $1.28_{/29}$. Here the constants are $\alpha_n = (4n-7)/(4n)$, $\beta_n = 3/(2n)$, and $\gamma_n = 1/(4n)$, as in $(A.6)_{/309}$, rather than α_n^*, β_n^* and γ_n^*, as in Figure $1.28_{/29}$, and the points to which they are applied are the E_i and F_i of Figure $1.27_{/29}$, rather than the E_i^L and F_i^L of $(1.5)_{/28}$.

Similarly to the presentation above for the Loop method, the method may be summarized as follows. For Catmull–Clark the vertices are classified as being of one of four types:

- *smooth* vertex (0 incident sharp edges);

- *dart* vertex (1 incident sharp edge);
- *crease* vertex (2 incident sharp edges);
- *corner* vertex (3 or more incident sharp edges).

Again, the nonsmooth vertices are said to be *tagged*, and the type of each vertex is completely determined by the set of sharp edges. In contrast to the Seattle work, there is no distinction made between regular and nonregular crease vertices. Operations involving vertices that are not on sharp edges, or on an existing vertex that is the endpoint of a lone sharp edge (i.e., a dart vertex), use the standard Catmull–Clark stencils shown in Figure $7.5_{/291}$. Operations involving an existing vertex with two or more incident sharp edges use the modified stencils of Figure $7.4_{/291}$. Creation of a new edge point on a sharp edge uses the modified stencil of Figure $7.4_{/291}$ (left).

Semisharp edges

The rules just described for sharp edges for the Catmull–Clark method were extended to the case of semisharp edges in [42]. This is important in practice since real-world surfaces can seldom be modelled as infinitely sharp: edges are usually rounded to some extent. Introduction of semisharp edges therefore introduces a higher level of realism, or permits the same level of realism with a smaller number of faces in the mesh [42, Fig. 10].

The methods used in [42] are conceptually simple. Sharpness of an edge is parametrized by a variable $s \in [0, \infty]$, where $s = 0$ is a smooth edge, and $s = \infty$ is an infinitely sharp edge. Integer values of s are obtained by using *hybrid subdivision*: the rules for sharp edges discussed previously in this section are applied for the first s subdivision steps, $\nu = 1, \ldots, s$, after which the standard (smooth) Catmull–Clark edge rules are used. This can be described intuitively by saying that the surfaces are sharp at coarser scales, but smooth at finer scales.

To obtain sharpness s for noninteger values of s, two sets of vertices along a soft crease (a sequence of semisharp edges) are computed. The first set of vertices is obtained by $\lfloor s \rfloor$ applications of the rules for sharp edges, followed by one application of the smooth edge rules. The second set of vertices is obtained by $\lceil s \rceil$ applications of the rules for sharp edges. The final values of the vertices are obtained by linear interpolation between the pairs of vertices so obtained.

A further generalization of this heuristic is suggested in [42, Appendix B], for the case when the crease is made up of a sequence of edges, each with a different value of s. In this case the values of s themselves are smoothed as subdivision proceeds, using Chaikin's method $LR(2)$.

7.1.2 The Biermann–Levin–Zorin rules for sharp edges

The rules for the Loop and Catmull–Clark methods, described in the previous section, were modified by Biermann, Levin, and Zorin in [15]. The purpose of these

modified rules is to prevent the possibility of folding of the surface, in the case of concave corners, and to permit the introduction of flatness and normal control. The modified rules are the basis, in particular, for a feature-editing method [17] mentioned in Section 7.3.1.

We mention three ways in which the modified rules are different from those presented above. The first is minor: in [15], a tagged vertex with only two incident sharp edges may be designated by the user to be a corner. This has the consequence that the type of a tagged vertex is no longer completely determined by the set of sharp edges, but this is no inconvenience. A second difference is that in the Loop case, the distinction between the two subtypes (namely, regular and nonregular) of crease vertex is abandoned. Consequently, there are only three possible types of (nonsmooth) vertices, as in the Catmull–Clark case in Section 7.1.1. The reason for this modification is that making the distinction between regular and nonregular crease vertices leads to subdivision rules for boundary edges that depend on the neighbouring topology. This means that two separate meshes, originally with common boundary, may no longer have a common boundary after subdivision. In [15], in contrast to the use of both stencils 2 and 3 (Figure 7.2$_{/290}$ and Table 7.1$_{/289}$), new vertices on sharp edges are always introduced as the average of two adjacent vertices.

The third difference between the preliminary rules of the previous section and the rules of [15] is that the edge rule is modified for a smooth edge adjacent to a tagged vertex. To do this, the concept of *sectors*, surrounding a tagged vertex, is introduced. At a tagged vertex, the sharp edges meeting at the vertex separate the ring of triangles around the vertex into sectors. Each of these sectors can be tagged as *convex* or *concave*, at the discretion of the user, subject to the restriction that concave sectors must consist of at least two faces.

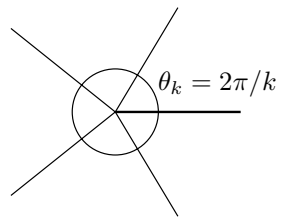
Now, given a logical vertex ℓ, two adjacent sharp edges in the logical mesh are separated by a certain angle α, $0 \le \alpha \le \pi$, which can be obtained by examining the geometric information associated with the sharp edges. If there are k faces within a given sector, the angle θ_k is defined as follows. If ℓ is

- a dart vertex, then $\theta_k = 2\pi/k$, $k \ge 1$;
- a crease vertex, then $\theta_k = \pi/k$, $k \ge 1$;
- a concave corner vertex, then $\theta_k = (2\pi - \alpha)/k$, $k \ge 2$;
- a convex corner vertex, then $\theta_k = \alpha/k$, $k \ge 1$

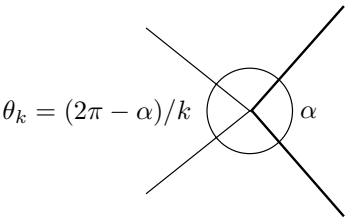
(see Figure 7.6$_{/294}$). Then, the standard rules of Figure 1.29$_{/32}$ (left) and Figure 7.5$_{/291}$ (left) are modified as shown in Figure 7.7$_{/294}$, where γ_k is defined to be

$$\gamma_k = \frac{1}{2} - \frac{1}{4}\cos\theta_k \quad \text{(Loop)},$$

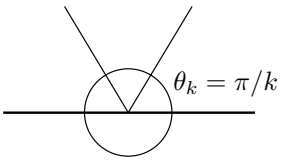
$$\gamma_k = \frac{3}{8} - \frac{1}{4}\cos\theta_k \quad \text{(Catmull–Clark)}.$$



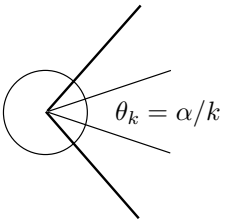
dart vertex ($k = 5$)



concave corner vertex ($k = 3$)

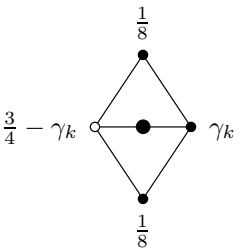


crease vertex ($k = 3$)



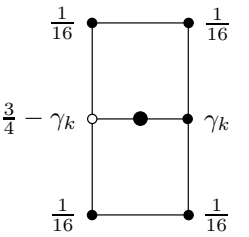
convex corner vertex ($k = 3$)

Figure 7.6. *Definition of the angle θ_k.*



Loop method

$\circ\ \ell$
(tagged vertex)



Catmull-Clark method

Figure 7.7. *Modified stencils for new edge points.*

Flatness and normal control are discussed in [15, Sec. 6.3]. An example showing how folding can occur if only the standard rules of Section 7.1.1 are used is given in Exercise $1_{/304}$.

7.1.3 Interpolation of position and normal direction

In [63], the evaluation and tangent stencils are used to generate a control mesh that interpolates some or all of the control points of a given input polyhedral mesh, and, possibly, to constrain the surface to have a specified normal at certain points. The method is described in the context of Catmull–Clark surfaces, expressed as

in $(A.8)_{/309}$ and $(A.9)_{/309}$ by

$$F'_j = F^L_j = \frac{1}{4}[V + E_j + F_j + E_{j+1}],\ j = 0, \ldots, n-1,$$
$$E'_j = \frac{1}{4}[V + E_j + F^L_{j-1} + F^L_j],\ j = 0, \ldots, n-1,$$
$$V' = \frac{n-2}{n}V + \frac{1}{n^2}\sum_{j=0}^{n-1} E_j + \frac{1}{n^2}\sum_{j=0}^{n-1} F^L_j,$$

where V is a control point at a vertex of valence n.

The new control mesh is taken to have the same topology as the input mesh, i.e., the same number and connectivity of vertices, faces, and edges. Position constraints are defined by $(6.3)_{/249}$, where η^1 is given by $(6.7)_{/251}$. This leads to a global system of linear constraints, where $x(0)$ in $(6.3)_{/249}$ corresponds to the control-point data to be interpolated (the given mesh), and p^0_l in $(6.3)_{/249}$ corresponds to the control mesh being generated. If p^0_l is written as a $(2n+1) \times N$ matrix

$$(F_0, \ldots, F_{n-1}, E_0, \ldots, E_{n-1}, V)^t, \tag{7.1}$$

then the constraint $(6.3)_{/249}$ requires that the inner product of η^1 with each column j of the displayed matrix in $(7.1)_{/295}$, $1 \le j \le N$, should be equal to the jth component of the value to be interpolated at vertex l. If $N = 3$, the number of unknowns and the number of constraints are each equal to 3 times the number of control points in the input polyhedral mesh. Since the matrix of constraints may be singular, a least-squares solution was used in [63].

To impose a given normal direction at specific point on the surface, corresponding to the limiting value of a control point, it is sufficient to require that c_2 and c_3, given by $(6.9)_{/252}$, are orthogonal to the given normal vector. This corresponds to 2 linear constraints, so that the total number of constraints is equal to 3 times the number of positional constraints plus 2 times the number of normal constraints. In [63], interpolation of normal directions was put aside, and the position-interpolation problem was solved with the incorporation in addition of a "fairness norm" to reduce the amount of undulation in the interpolating surface.

The interpolation guaranteed here occurs only in the limit, and the nonsingularity of the coefficients matrix is not guaranteed [115, Sec. 6].

The problem of interpolation of boundary curves is discussed in the context of a dual method in the next section. Further references on surface fitting are given in Section 7.3.

7.2 Shape control for dual methods

One of the first discussions of interpolation for subdivision surfaces was in a paper of Nasri [106]. This paper and subsequent papers [107, 108, 109, 110, 111] by the same author were focused on the Doo–Sabin method, a dual method. Intuitively, one might expect that boundary control would be more difficult for dual methods than for primal methods, because dual methods abandon the initial polyhedron

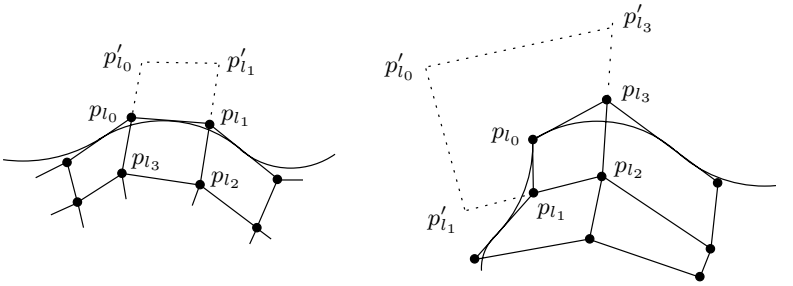


Figure 7.8. *Polyhedron modification for four-sided faces.*

boundary at the first subdivision step. At least in the case of the Doo–Sabin method, however, it is possible to devise interpolation methods in the dual case, since the quadratic nature of the limit surface in the regular case leads to simple linear constraints defining interpolation.

The papers [114, 115] give a taxonomy of interpolation constraints for shape control in a general context, but with some emphasis on the case of dual methods.

7.2.1 Control-point modification

One approach to designing interpolation methods in the dual case involves only the modification of control points.

We saw in Chapter 1 that methods based on $dQ4$ splitting cause an extraordinary vertex to be transformed into an extraordinary face at the first subdivision step, and that subsequently no extraordinary faces other than those originally in the mesh are created. Here we suppose for simplicity that the original mesh had no extraordinary vertices, so that the only extraordinary faces in the subdivided mesh are those in the original mesh.

An important fact about the Doo–Sabin method is that the limiting surface contains the centroids $\frac{1}{e}\sum_{j=0}^{e-1} p_{l_j}$ of all faces at all refinement levels. Also, given some face with control points p_{l_j}, $j = 0, \ldots, e-1$, at some refinement level, local subdivision of that face produces a unique sequence of faces, with the same number of edges, and where the control points of the faces converge to the centroid of the original face. To prove this, note that from $(1.3)_{/27}$, the local subdivision matrix for the Doo–Sabin method is the $e \times e$ symmetric matrix with elements W_{ij}, $0 \le i, j \le e-1$. From the symmetry of the matrix, $\sum_{i=0}^{e-1} W_{ij} = 1$ (compare $(1.4)_{/28}$), and a normalized eigenvector is $\eta = (1/e, \ldots, 1/e)^t$. This vector defines the evaluation stencil for any control point in the face. Thus, each point p_{l_i} begins a sequence that converges to $\frac{1}{e}\sum_{j=0}^{e-1} p_{l_j}$.

Nasri [106] used this fact to modify the control points of a given polyhedral mesh $\mathcal{M}$ so that the Doo–Sabin method converges to prescribed values along boundary edges and at corners. Consider, for example, a four-sided face p_{l_0}-p_{l_1}-p_{l_2}-p_{l_3}, as illustrated in Figure $7.8_{/296}$ (see [106, Fig. 4]). In the case when the face has two nonboundary vertices, as in Figure $7.8_{/296}$ (left), the control points p_{l_2} and p_{l_3} can

be reflected in p_{l_1} and p_{l_0} to give p'_{l_1} and p'_{l_0}, respectively, so that the centroid of the polyhedron p'_{l_0}-p'_{l_1}-p_{l_2}-p_{l_3} is equal to $\frac{1}{2}(p_{l_0} + p_{l_1})$. The boundary of the limiting surface is a quadratic B-spline curve with p_{l_0} and p_{l_1} in the control polyhedron, and $\frac{1}{2}(p_{l_0} + p_{l_1})$ is a point on this curve. Similarly, in the case when the face has only one nonboundary vertex, and the middle boundary vertex has no internal link, as in Figure 7.8$_{/296}$ (right), then we can take

$$\begin{aligned} p'_{l_1} &= 2p_{l_1} - p_{l_2}, \\ p'_{l_3} &= 2p_{l_3} - p_{l_2}, \\ p'_{l_0} &= 4p_{l_0} - 2p_{l_1} - 2p_{l_3} + p_{l_2} \end{aligned}$$

to obtain a modified face p'_{l_0}-p'_{l_1}-p_{l_2}-p'_{l_3} with centroid p_{l_0}. In the limit surface, two separate boundary curves interpolate p_{l_0}, and they are tangential to p_{l_0}-p_{l_3} and p_{l_0}-p_{l_1}, respectively.

In [106], extensions of these methods are proposed for the case of extraordinary faces, and for the solution of generally stated interpolation problems, including (in our terminology) the following: "Given a polyhedral mesh $\mathcal{M}$, construct a new mesh, based on the same logical mesh, that interpolates (in the limit under Doo–Sabin subdivision) some or all of the geometric vertices of $\mathcal{M}$." Both the cases with and without interpolation along the boundary are considered. The solutions to these problems involve large but sparse systems of linear equations.

It is clear, also, that by using the tangent stencils for the Doo–Sabin method, it is possible to devise algorithms that adjust the initial control points of a polyhedral mesh in order to match given surface normal vectors. Tangent stencils for the Doo–Sabin method can be obtained by finding eigenvectors corresponding to the eigenvalues $\lambda_2 = \lambda_3 = 1/2$. Two such eigenvectors can be obtained by taking the real and imaginary parts of the complex eigenvector $(1, w, w^2, \ldots, w^{e-1})^t$, where $w = e^{2\pi i/e}$, since this eigenvector corresponds to the eigenvalue $1/2$. Exercise 2$_{/304}$ asks for the verification. See also [108].

7.2.2 Other approaches

Other methods have also been developed to achieve interpolation in the case of dual methods. Thus, in [109, 110] it is shown how to modify the *topology* of $\mathcal{M}$ in order to construct strips of panels interpolating predefined curves in the limit. For example, in [109] it is shown how to use this type of construction to interpolate predefined closed quadratic curves.

Methods involving modification of the subdivision process itself (as was done in the primal case) have also been devised. A classification of the various approaches is given in [115].

7.3 Further reading on shape control

We conclude this chapter (and the main text of the book) with some brief descriptions of other important methods for shape control and give some references for further reading.

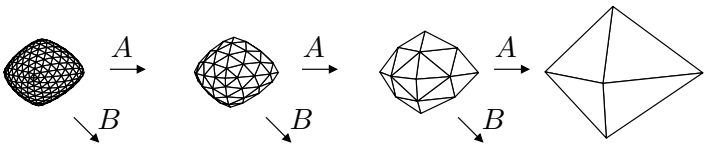


Figure 7.9. *A subdivision surface at four levels of resolution.*

We begin, in Sections 7.3.1 and 7.3.2, with a summary of an important paradigm for shape control, multiresolution editing. Then, in Section 7.3.3, we give references to work on other aspects of shape control, including further references for subdivision-surface fitting (already discussed to some extent in Sections 7.1 and 7.2), variational design, and Boolean operations on subdivision-surface solids.

Only the methods of Section 7.3.1, on subdivision-based multiresolution editing, are treated in any detail. In particular, we give a fair amount of detail for the wavelet-decomposition method, in order to show the interesting connection with wavelets. It turns out, however, that due to the computational cost of the associated algorithms, this method is of less practical importance than the smoothing approach presented immediately afterward.

7.3.1 Subdivision-based multiresolution editing

Subdivision-based multiresolution methods can be used for mesh compression, level-of-detail display, progressive transmission, and multiresolution editing, among other applications [62, 92]. Here we focus on multiresolution editing, where two principal methods have been proposed. The goal of multiresolution editing, as the name suggests, is to permit editing at any level of resolution, and to have the changes appropriately reflected in models corresponding to other levels of resolution.

The first of the two methods [92] is based on a wavelet decomposition of the mesh in the spatial domain, while the second [131, 180] can be viewed as imitating the wavelet-decomposition process. In both cases the reconstruction step that takes us to higher levels of resolution can be viewed as a combination of a subdivision step and a correction step [19, Sec. 4.2].

The wavelet-decomposition method

Our presentation of this method follows [92] quite closely, although we change the notation to correspond to that used in this book. A similar presentation can be found in [156, Ch. 10]. See also [50, Appendix A].

Consider the triangular mesh shown in Figure 7.9$_{/298}$ (left). The figure is similar to [92, Fig. 1]. This surface can be viewed as a parametric function on the sphere, and parametrized over any surface homeomorphic to the sphere. The function is decomposed into a low-resolution part (second illustration in Figure 7.9$_{/298}$) and a

detail part defined in terms of *wavelet coefficients*. The wavelet coefficients are specified as a function of the control points of the original triangular mesh, Figure 7.9$_{/298}$ (left), as described below. Similarly, the control points of the low-resolution part are specified as a function of the control points of the original triangular mesh; in fact, they are computed as certain weighted averages of these control points. These weighted averages can be viewed as defining a low-pass filter, denoted A, while the wavelet coefficients can be viewed as defining a high-pass filter, denoted B. This decomposition (or *analysis*) process can be used to decompose the surface into an even lower-resolution version, with corresponding wavelet coefficients, as illustrated in Figure 7.9$_{/298}$ (middle and right).

Now, moving instead from right to left in Figure 7.9$_{/298}$, the original triangular mesh can be recovered from the lowest-resolution version and the wavelet coefficients corresponding to all levels. This reconstruction (or *synthesis*) process involves subdividing each coarse-level triangle by means of, say, a $pT4$-type subdivision process, followed by a perturbation of the control points using the wavelet coefficients.

If an edit involving modifications of control points is made at some level of resolution, it will be reflected in objects of lower or higher resolution, and these modified objects can be found by analysis or synthesis.

In [92], the analysis and synthesis processes were defined in terms of a spatial-domain wavelet formulation. In general, such a formulation involves an infinite chain of linear function spaces

$$V^0 \subset V^1 \subset \cdots$$

and an inner product $\langle f,\ g\rangle$ defined for any pair of functions $f, g \in V^j$, $0 \leq j$. We think of V^j as containing the functions of resolution j, with detail increasing as j increases. The inner product defines the orthogonal complement spaces

$$W^j = \{f \in V^{j+1} : \langle f,\, g\rangle = 0,\, g \in V^j\},$$

and $f^{j+1} \in V^{j+1}$ can be written uniquely as an orthogonal decomposition $f^{j+1} = f^j + h^j$, where $f^j \in V^j$ and $h^j \in W^j$. The function f^j corresponds to the low-resolution part, and h^j corresponds to the detail part. It should be noted, however, that in [92] the orthogonality requirements are dropped.

Let the lowest-resolution subdivision surface (Figure 7.9$_{/298}$, right) be denoted by the polyhedral mesh $\mathcal{M}^0$. The final surface is parametrized over $\mathcal{M}^0$, which therefore plays the role of the manifold $\boldsymbol{M}$ in Section 4.6, above. (Since in the present case the manifold is assumed to be a well-defined subset of $\mathbb{R}^3$, there is no need to introduce charts.) Now, given a polyhedral mesh $\mathcal{M}^{\nu-1}$, a subdivision step can be viewed as composed of two substeps. The first is a splitting step, which uses $pT4$ splitting to produce an auxiliary mesh $\hat{\mathcal{M}}^\nu$ with the same logical mesh as $\mathcal{M}^\nu$, i.e., with four subfaces for each face in $\mathcal{M}^{\nu-1}$, but with the new control points defined to be the midpoints of edges in $\mathcal{M}^{\nu-1}$ (Figure 7.10$_{/300}$, middle). The second is an averaging step, which uses the subdivision rules to produce the polyhedral mesh $\mathcal{M}^\nu$ (Figure 7.10$_{/300}$, right). The subdivision process can now be

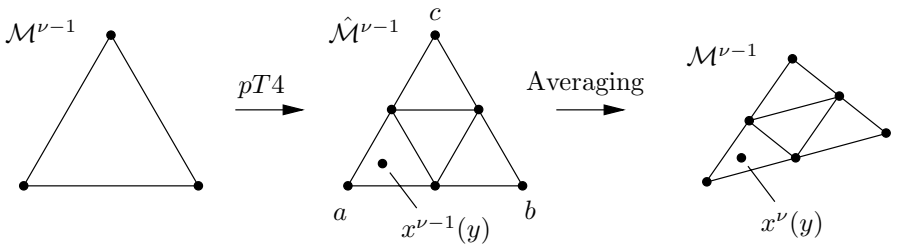


Figure 7.10. *A subdivision step viewed as two substeps.*

used to define, for each j, a collection of functions $N_\ell^j(y)$, $y \in \mathcal{M}^0$, satisfying

$$x(y) = \sum_\ell p_\ell^j N_\ell^j(y), \tag{7.2}$$

as in $(4.74)_{/181}$. Here, the index j corresponds to the level of resolution; in particular, $j = 0$ corresponds to the lowest-resolution surface $\mathcal{M}^0$, and the summation is over the control points in $\mathcal{M}^j$. (Equation $(7.2)_{/300}$ corresponds to [92, eq. (4)].) Each function N_ℓ^j lies in the span of the functions with superscript $j+1$. Note that, as in the case of $(4.74)_{/181}$, in the case of a nonregular mesh, the functions N_ℓ^{j+1} are not necessarily related to the functions N_ℓ^j by a 2-scale relation.

The functions N_ℓ^j are obtained in the following way. The subdivision process, from $(1.13)_{/39}$, is defined by

$$p^{j+1} = \Sigma^j p^j, \quad j = 0, 1, \ldots$$

(see $(1.13)_{/39}$ in Chapter 1; this corresponds to [92, eq. (1)]). Now, define $x^\nu(y)$ as follows. First, $x^0(y) = y$, $y \in \mathcal{M}^0$. Then, if $x^{\nu-1}(y)$ lies in the triangle $(\hat{p}_a^\nu, \hat{p}_b^\nu, \hat{p}_c^\nu)$ of $\hat{\mathcal{M}}^\nu$ with barycentric coordinates (α, β, γ), the function $x^\nu(y)$ is defined as

$$x^\nu(y) = \alpha p_a^\nu + \beta p_b^\nu + \gamma p_c^\nu,$$

where $(p_a^\nu, p_b^\nu, p_c^\nu)$ is the triangle of $\mathcal{M}^\nu$ corresponding to $(\hat{p}_a^\nu, \hat{p}_b^\nu, \hat{p}_c^\nu)$ in $\hat{\mathcal{M}}^\nu$. See Figure $7.10_{/300}$.

It is then shown in [92, Lemma 4.2.1] that $x^\nu(y)$ can be written as

$$x^\nu(y) = \sum_\ell p_\ell^j N_\ell^{\nu \leftarrow j}(y),$$

where

$$N^{\nu \leftarrow j}(y) = b^\nu(y) \Sigma^{\nu-1} \Sigma^{\nu-2} \cdots \Sigma^j.$$

Here, b^ν is a $(1 \times L_\nu)$ row vector with zeroes everywhere except at the indices corresponding to vertices a, b, and c in Figure $7.10_{/300}$ (middle). Recall that Σ^j is $(L_{j+1} \times L_j)$, p^j is $(L_j \times N)$, and $x^\nu(y)$ is $(1 \times N)$.

Next, taking $N_\ell^j(y) = \lim_{\nu\to\infty} N_\ell^{\nu\leftarrow j}(y)$, Theorem 4.2.1 in [92] gives

$$x(y) = \sum_\ell p_\ell^j N_\ell^j(y)$$

for meshes of arbitrary topology, as desired. The row vector composed of the nodal functions $N_\ell^j(y)$ can be obtained by multiplying the corresponding row vector of functions $N^{j+1}(y)$ on the right by Σ^j. This permits the definition of a chain of nested linear spaces $V^j(\mathcal{M}^0)$ defined by the span of the functions $N_\ell^j(y)$.

To obtain a computational algorithm, the functions $N_\ell^j(y)$ are separated into two groups corresponding to old and new vertices, and wavelets are defined [92, eqs. (17)–(18)] as the difference between $N_\ell^{j+1}(y)$ and a certain linear combination of the functions $N_\ell^j(y)$. The wavelets are chosen so that the number of nonzero coefficients in the linear combination is a fixed constant, so that the analysis and synthesis processes (and in particular the calculation of wavelet coefficients) can be realized by inverting a sparse matrix [92, Sec. 6.3]. Unfortunately, for methods like the Loop method, the inversion of the sparse matrix leads to nonlinear computation times for each step in the analysis process. This is the main drawback of the wavelet-based approach, relative to the more heuristic approach described next.

A smoothing approach to subdivision-based multiresolution models

A simpler and somewhat heuristic multiresolution scheme can be obtained by replacing the analysis step in the wavelet approach by a smoothing procedure [180]. Then, further heuristics, obtained by ignoring details below a certain threshold (intuitively, in flat regions of the mesh, there are no significant details), can be introduced to obtain interactive speeds. To make these heuristics work, the detail vectors are defined in a local coordinate frame, as suggested in a different hierarchical-editing context in earlier work by Forsey and Bartels [55]. See also [131].

In the following presentation, we continue to use the notation of this book. Also, we continue to use the global subdivision matrix Σ^j to represent subdivision of the surface defined by the control points p^j, although these matrices are replaced by a single linear operator S in [180]. The more explicit notation makes clear how many control points are being transformed and produced at each stage: we again recall that Σ^j is $(L_{j+1} \times L_j)$.

As in the wavelet-decomposition method, the object to be edited is defined at several levels of resolution, which are represented here by the polyhedral meshes $\mathcal{M}^j$, $j = 0, \ldots, k$. We can again make use of Figure 7.9$_{/298}$, where $k = 3$, and the polyhedral meshes illustrated are now named, in order, $\mathcal{M}^3$, $\mathcal{M}^2$, $\mathcal{M}^1$, and $\mathcal{M}^0$.

The meshes in the sequence $\{\mathcal{M}^j\}_{k\geq j\geq 0}$ are very closely linked. We begin with the topology of the meshes, defined by the associated logical meshes $\{M^j\}_{k\geq j\geq 0}$. The coarsest-level mesh is a well-formed mesh (we restrict our attention to the case of triangulated locally planar meshes). Further, the logical mesh M^j is obtained from the mesh M^{j-1} by $pT4$ subdivision, $j = 1, \ldots, k$. A mesh $\mathcal{M}^k$ with topology obtained in this way is said to have *subdivision connectivity*, or is said to be a *semiregular* mesh. As discussed below, the requirement of subdivision

connectivity is sometimes cited as a drawback of subdivision-based multiresolution methods.

The control vectors corresponding to the meshes $\{\mathcal{M}^j\}_{k\geq j\geq 0}$ are also closely related. These control sequences are denoted here by q^j to distinguish them from the sequences p^j in $(4.75)_{/182}$, where p^{j+1} is obtained exactly from p^j by subdivision. Here we have an initial sequence of control vectors in $\mathbb{R}^N$, namely

$$q^0 = q^0_{(L_0\times N)},$$

which corresponds to the coarsest-level representation. The finer levels of resolution are represented by the control vectors

$$q^j = q^j_{(L_j\times N)},$$

but these vectors are not related exactly by subdivision. Rather, they are related to each other by a smoothing operator $H^j = H^j_{(L_{j-1}\times L_j)}$:

$$H^{j+1}q^{j+1} = q^j, \quad j = k-1,\ldots,0. \tag{7.3}$$

Thus, the operators H^j completely define the analysis process: given q^{j+1}, we can compute q^j.

The synthesis process is defined by subdivision plus correction. To simplify the discussion, we temporarily omit the orthogonal matrix F^j which defines the change to the local coordinate frame, mentioned above. At the end of the discussion, however, the equations defining the synthesis will be displayed with F^j included.

To obtain q^{j+1} from q^j (synthesis) we must also have available *detail vectors* d^j, $j = 1,\ldots,k$, where d^j is an additive correction to be applied following subdivision at level j. They are defined by

$$d^{j+1} = q^{j+1} - \Sigma^j H^{j+1}q^{j+1}, \quad j = 0,\ldots,k-1.$$

The synthesis step is in turn defined by

$$q^{j+1} = \Sigma^j q^j + d^{j+1}, \quad j = 0,\ldots,k-1,$$

which is consistent with $(7.3)_{/302}$. The analysis and synthesis processes are illustrated in Figure $7.11_{/303}$.

An orthogonal matrix F^j is introduced at each step in order to permit the transformation into a local system of coordinates, as mentioned above. Tangent stencils (Section 6.1.2) are used to find this local coordinate system. The equations for the detail vectors, and for synthesis, become

$$d^{j+1} = (F^{j+1})^t(q^{j+1} - \Sigma^j H^{j+1}q^{j+1}), \quad j = 0,\ldots,k-1,$$

and

$$q^{j+1} = \Sigma^j q^j + F^{j+1}d^{j+1}, \quad j = 0,\ldots,k-1,$$

respectively.

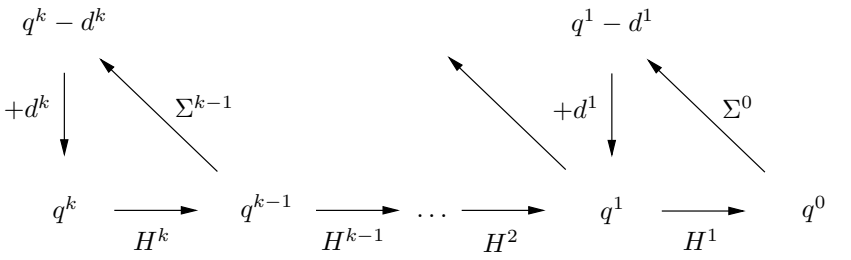


Figure 7.11. *Analysis and synthesis processes.*

When an edit level, say level j, is chosen by the user, the surface is represented at this level as an approximation defined by q^j and the finer-level details $d^{j+1}, \ldots, d^k$. If the control points q^j are modified by the edit, the system can use the synthesis algorithm to render the modified object, with the finer-level details added in unchanged. At the end of a sequence of edits at level j, the analysis algorithm must be used to update the values of $q^{j-1}, \ldots, q^0$ and $d^j, \ldots, d^1$.

Storage of all of the meshes $\mathcal{M}_k, \mathcal{M}_{k-1}, \ldots, \mathcal{M}_0$ increases memory requirements, beyond those for $\mathcal{M}_k$ alone, by only a third; see Exercise $3_{/305}$. As already mentioned, heuristics are introduced in [180] in order to achieve computation times suitable for interactive use. The heuristics are based on lazy evaluation and the use of thresholds for the detail vectors.

Subdivision-based multiresolution editing has been integrated into more general feature-editing systems that include the possibility of editing sharp features using the methods of Section 7.1.2 [13, 16, 17].

7.3.2 Mesh-decimation multiresolution editing

It was noted in the previous section that subdivision-based multiresolution methods depend on the fact that the meshes $\mathcal{M}^j$ in the hierarchy, $k \geq j \geq 0$, have subdivision connectivity. Since in practice that finest-level mesh may have been obtained directly as a triangular mesh, it may well happen that the topology of the mesh $\mathcal{M}^k$ does not satisfy this requirement. Remeshing algorithms have therefore been devised to resolve this problem. In particular, [50] gives a remeshing algorithm that produces a new polyhedral mesh that approximates $\mathcal{M}^k$ to any desired precision, and such that the logical mesh M^k has subdivision connectivity relative to a simple base mesh found by Delaunay triangulation. See also [61, 82].

The requirement of subdivision connectivity is still sometimes seen as a drawback, however, and analogous algorithms based on mesh decimation have been proposed as alternatives [78]. Like subdivision-based multiresolution, these algorithms have application beyond editing, such as those described at the beginning of Section 7.3. Good discussions of the trade-offs between the subdivision-based approach and the alternatives can be found in [77, 175].

7.3.3 Other aspects of subdivision-surface shape control

Surface fitting can be viewed as one aspect of free-form editing [175, Sec. 3.1], and it has been discussed to some extent in the first two sections of the chapter. In particular, the overview [115] was mentioned. One way of dealing with the problem is described in [84, 85, 90], where surfaces are trimmed to an externally specified boundary curve $c : [0, 1] \mapsto \mathbb{R}^3$. Thus, a surface with boundary is represented by a Loop-subdivision geometric mesh, in conjunction with parametric curves defining the boundary. In [90] this approach is called *Combined Loop subdivision*. See also [89]. Whether it is reasonable to assume that patch-boundary trimming curves are available depends on the application context. One application area where they are certainly assumed available is in solid-modelling systems [1, 154]. This approach provides one way to represent the boundaries of what are called *nonmanifold objects*[46] in solid modelling [65, p. 61], [136].

A good survey for shape control in the context of subdivision surfaces is [175]. One important topic is variational design, which attempts to optimize fairness of the surface; see for example [74]. Other topics include the problem of specifying a region of interest when making edits, in order to avoid modification of parts of the surface that have already been carefully defined, and the problem of topological modifications (as opposed to modifications of the control points). More generally, the problem of computing Boolean operations on solids defined by subdivision surfaces is of great interest [139]; see in particular [14]. Many of the problems mentioned here, and the proposed solutions, take us outside the realm of affine-invariant subdivision.

7.4 Additional comments

The practically oriented literature dealing with shape control is very large, and we have cited a relatively small number of papers. Good starting points for further reading are the surveys already mentioned [115, 175].

7.5 Exercises

1. Show that for the concave corner illustrated in Figure 7.12$_{/305}$, the first step of Loop subdivision, using the edge rules suggested in Section 7.1.1, causes the triangle $(-1, 1, 0)^t$—$(-1, -1, 0)^t$—$(0, 0, 0)^t$ to flip upside down, so that folding of the surface occurs. (The example uses triangles with large aspect ratios so that the folding will appear immediately.)

2. The Doo–Sabin local subdivision matrix has elements W_{kj}, $0 \le k, j \le e - 1$, as defined in (1.3)$_{/27}$. Show that $(1, w, w^2, \ldots, w^{e-1})$, where $w = e^{2\pi i/e}$, is a complex left eigenvector of this matrix, with corresponding eigenvalue $\lambda = 1/2$. (Since the matrix is symmetric, the transpose of the vector is also a right eigenvector.) Taking real and imaginary parts gives two real eigenvectors of the subdivision matrix. These real eigenvectors can be used as tangent

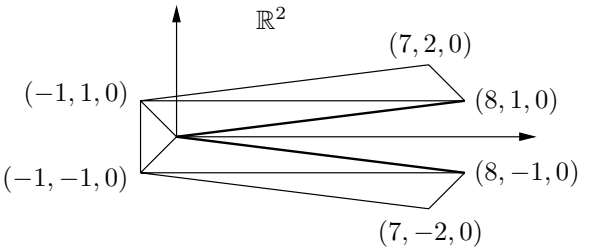


Figure 7.12. *Simple example of folding.*

stencils for the Doo–Sabin method, and these stencils can in turn be used to define equations prescribing normal vectors on the surface.

3. Show that the memory requirements for a multiresolution method, like the one described in Section 7.3.1, are only about one third above those for the storage of the highest-resolution mesh.

7.6 Projects

1. *Implementation of soft creases.*

   Implement a user interface that permits input of wire-frame objects, with interactive tagging of edges with sharpness $s \in (0, \infty]$, and in association with this, implement also the Catmull–Clark method with semisharp edges described in [42, Appendix B].

2. *Empirical verification of folding.*

   Implement a simplified version of Loop subdivision for coplanar control points in the neighbourhood of a concave corner such as that illustrated in Figure 7.12$_{/305}$ (see also Figure 7.6$_{/294}$, upper right).

   Confirm empirically that folding eventually occurs if $\alpha < \pi$, independently of the triangle layout near the corner.

Appendix

The first section of this appendix summarizes equivalent formulations of the Catmull–Clark method. Later sections give details concerning the Fourier transform, the Fourier series, and certain proofs that were omitted in the main text.

A.1 Equivalence of Catmull–Clark rules

There are three major formulations of the Catmull–Clark method. Here we show their algebraic equivalence.

Catmull–Clark/Repeated Averaging formulation

In the original form [24] of the method, the new vertex point was defined as

$$\left(\frac{n-3}{n}\right) S + \left(\frac{2}{n}\right) R + \left(\frac{1}{n}\right) Q.$$

In the notation of Section 1.3.1,

$$\begin{aligned} S &= V, \\ R &= \frac{1}{n} \sum_{j=0}^{n-1} \frac{1}{2}(V + E_j), \\ Q &= \frac{1}{n} \sum_{j=0}^{n-1} F'_j, \end{aligned}$$

and it was shown in that section that, except for variable names and the order of computation, the original form is equivalent to the "in-place formulation" described there.

We write the new vertex point in an inner-product–like notation:

$$\left[\begin{array}{ccc} \frac{n-3}{n} & \frac{2}{n} & \frac{1}{n} \end{array} \right] \cdot \left[\begin{array}{ccc} S & R & Q \end{array} \right], \tag{A.1}$$

$$\left[\begin{array}{ccc} \frac{n-3}{n} & \frac{2}{n} & \frac{1}{n} \end{array} \right] \cdot \left[\begin{array}{ccc} V & \frac{1}{n} \sum_{j=0}^{n-1} \frac{1}{2}(V + E_j) & \frac{1}{n} \sum_{j=0}^{n-1} F'_j \end{array} \right], \tag{A.2}$$

or

$$\left[\begin{array}{ccc} \alpha_n^* & \beta_n^* & \gamma_n^* \end{array}\right] \cdot \left[\begin{array}{ccc} V & \frac{1}{n}\sum_{j=0}^{n-1}\frac{1}{2}(V+E_j) & \frac{1}{n}\sum_{j=0}^{n-1}F_j' \end{array}\right], \tag{A.3}$$

remembering that each component of the 3-tuple on the right in $(A.2)_{/307}$ is a vector in $\mathbb{R}^N$. Here,

$$\alpha_n^* = \frac{n-3}{n}, \quad \beta_n^* = \frac{2}{n}, \quad \gamma_n^* = \frac{1}{n}.$$

Except for the range of the subscript on the variables, the notation E_j, F_j, E_j', and F_j' is that of [8, 9]. We use it in this section to describe the various formulations.

In [24, p. 353], the alternate possibility of using the weights corresponding to $n = 4$ for every vertex, independently of its valence, is mentioned. This would mean computing the new vertex point from

$$\left[\begin{array}{ccc} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{array}\right] \cdot \left[\begin{array}{ccc} S & R & Q \end{array}\right]. \tag{A.4}$$

This choice of weights was rejected as unsatisfactory because it produced objects that were too "pointy." See the second part of Exercise $7_{/48}$ in Chapter 1.

The modified face points are given by

$$F_i' = F_i^L = \frac{1}{4}(V + E_i + F_i + E_{i+1}), \qquad i = 0, \ldots, n-1,$$

and the modified edge points by

$$E_i' = \frac{3}{8}(V + E_i) + \frac{1}{16}(E_{i-1} + F_{i-1} + F_i + E_{i+1}), \qquad i = 0, \ldots, n-1,$$

as in $(1.6)_{/29}$ and $(1.7)_{/29}$.

It was also shown in Section 1.3.1 that the Repeated Averaging formulation of Catmull–Clark is equivalent to the original Catmull–Clark formulation [24].

Ball–Storry formulation

It is often convenient to express the new control-point values as a linear combination of the averages of the original points E_j and F_j, $j = 0, \ldots, n-1$. For example, this was done in $(1.17)_{/44}$, in the discussion of subdivision matrices, and in Figure $7.5_{/291}$, in the discussion of sharp edges. The new values F_i' and E_i' are expressed as before, but $(A.2)_{/307}$ is now written as

$$\left[\begin{array}{ccc} \frac{4n-7}{4n} & \frac{3}{2n} & \frac{1}{4n} \end{array}\right] \cdot \left[\begin{array}{ccc} V & \frac{1}{n}\sum_{j=0}^{n-1}E_j & \frac{1}{n}\sum_{j=0}^{n-1}F_j \end{array}\right], \tag{A.5}$$

the formulation given by Ball and Storry [8, 9]. It was shown in Section 1.4.3 that $(A.2)_{/307}$ and $(A.5)_{/308}$ are equivalent: this is a straightforward algebraic verification. See Exercise $11_{/49}$ in Chapter 1.

In [8, 9] the definitions

$$\alpha_n = \frac{4n-7}{4n}, \quad \beta_n = \frac{3}{2n}, \quad \gamma_n = \frac{1}{4n}$$

are made, so that the Ball–Storry formulation for the evaluation of the new vertex point V' can also be written as

$$\left[\begin{array}{ccc} \alpha_n & \beta_n & \gamma_n \end{array}\right] \cdot \left[\begin{array}{cc} V & \frac{1}{n}\sum_{j=0}^{n-1} E_j \quad \frac{1}{n}\sum_{j=0}^{n-1} F_j \end{array}\right]. \tag{A.6}$$

A similar notation is used in [124, Chap. 6].

The alternate weights corresponding to $n = 4$ result in the evaluation rule

$$\left[\begin{array}{ccc} \frac{9}{16} & \frac{3}{8} & \frac{1}{16} \end{array}\right] \cdot \left[\begin{array}{cc} V & \frac{1}{n}\sum_{j=0}^{n-1} E_j \quad \frac{1}{n}\sum_{j=0}^{n-1} F_j \end{array}\right], \tag{A.7}$$

as mentioned in [8]. This is equivalent to $(A.4)_{/308}$.

The coefficients α_n^*, β_n^*, and γ_n^* were denoted by α_n, β_n, and γ_n in [151]. This notation is inconsistent with the earlier and frequently used notation of [8, 9]. To remove the inconsistency, we have renamed the coefficients in [151] as α_n^*, β_n^*, and γ_n^*, i.e., we have added a star to the names of the coefficients in [151].

Cohen–Riesenfeld–Elber formulation

Sometimes, as in [30, 42], $(A.2)_{/307}$ is rewritten as

$$\left[\alpha_n - \gamma_n \quad \beta_n - 2\gamma_n \quad 4\gamma_n\right] \cdot \left[V \quad \frac{1}{n}\sum_{j=0}^{n-1} E_j \quad \frac{1}{n}\sum_{j=0}^{n-1} \frac{1}{4}(V + E_j + F_j + E_{j+1})\right],$$

an equivalence that is easily verified algebraically. This reduces to

$$\left[\begin{array}{ccc} \frac{n-2}{n} & \frac{1}{n} & \frac{1}{n} \end{array}\right] \cdot \left[\begin{array}{cc} V & \frac{1}{n}\sum_{j=0}^{n-1} E_j \quad \frac{1}{n}\sum_{j=0}^{n-1} \frac{1}{4}(V + E_j + F_j + E_{j+1}) \end{array}\right], \tag{A.8}$$

as stated in [30, eq. (20.12)], and in the case $n = 4$ to

$$\left[\begin{array}{ccc} \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \end{array}\right] \cdot \left[\begin{array}{cc} V & \frac{1}{n}\sum_{j=0}^{n-1} E_j \quad \frac{1}{n}\sum_{j=0}^{n-1} \frac{1}{4}(V + E_j + F_j + E_{j+1}) \end{array}\right],$$

as mentioned in [30, Sec. 20.1.3]. This last is equivalent to $(A.4)_{/308}$. The 3-tuple on the left is different from that in $(A.4)_{/308}$ (there is no typographical error here). The 3-tuple on the right is denoted $[\,C \quad \mathcal{V} \quad \boldsymbol{p}\,]$ in [30, eq. (20.12)].

The face-point and edge-point updates in [30, eq. (20.12)] are also consistent, with the following correspondences: $p_i \leftrightarrow F_i^L$ and $v_i \leftrightarrow E_i'$. In fact, the overall formulation can be summarized as

$$F_i' = F_i^L = \frac{1}{4}[V + E_i + F_i + E_{i+1}], \qquad i = 0, \ldots, n-1,$$

$$E_i' = \frac{1}{4}[V + E_i + F_{i-1}^L + F_i^L], \qquad i = 0, \ldots, n-1, \tag{A.9}$$

and $(A.8)_{/309}$, above, and these correspond exactly to [30, eq. (20.12)]. Note that $(A.9)_{/309}$ follows from $(1.5)_{/28}$ and $(1.7)_{/29}$.

A.2 The complex Fourier transforms and series

In this section we define the complex Fourier transform in one and two dimensions and summarize its most important properties [21, 47, 54, 153]. We also summarize discrete Fourier series [117].

A.2.1 The Fourier transform: Univariate case

If $f : \mathbb{R} \to \mathbb{R}$, or alternatively $f : \mathbb{R} \to \mathbb{C}$, is a real- or complex-valued function and if $f \in L^1(\mathbb{R})$, i.e., $\int_{\mathbb{R}} |f(t)|\, dt < \infty$, then we have the following definition.

Definition A.2.1.

$$\hat{f}(\omega) = \int_{\mathbb{R}} f(t) e^{-i\omega t}\, dt. \tag{A.10}$$

Although in this book we are usually concerned with real-valued functions, it is most convenient to present the theory using complex-valued functions.

Inversion formula: If, in addition, f is piecewise continuously differentiable for $-\infty < t < \infty$, then

$$\frac{1}{2}\left(f(t_+) + f(t_-)\right) = \frac{1}{2\pi} \lim_{A\to\infty} \int_{-A}^{A} \hat{f}(\omega) e^{i\omega t}\, d\omega.$$

If $\hat{f} \in L^1(\mathbb{R})$, then $f(t)$ is continuous for all t and we have

$$f(t) = \frac{1}{2\pi} \int_{\mathbb{R}} \hat{f}(\omega) e^{i\omega t}\, d\omega.$$

The following is often useful.

Parseval's formula: If $f \in L^2(\mathbb{R})$, then $\hat{f} \in L^2(\mathbb{R})$ and

$$\int_{\mathbb{R}} |f(t)|^2\, dt = \frac{1}{2\pi} \int_{\mathbb{R}} |\hat{f}(\omega)|^2\, d\omega,$$

where $L^2(\mathbb{R})$ denotes the class of functions such that $\int_{\mathbb{R}} |f(t)|^2\, dt < \infty$, and where $\hat{f}(\omega)$ is defined for $f \in L^2(\mathbb{R})$ as

$$\hat{f}(\omega) = \lim_{A\to\infty} \int_{-A}^{A} f(t) e^{-i\omega t}\, dt.$$

Here, the limit in the right-hand side has to be interpreted in the sense of mean-square convergence [47]. More generally, if f and $g \in L^2(\mathbb{R})$, then

$$\int_{-\infty}^{\infty} f(t)\overline{g(t)}\, dt = \frac{1}{2\pi} \int_{\mathbb{R}} \hat{f}(\omega)\overline{\hat{g}(\omega)}\, d\omega,$$

where the bar denotes complex conjugation.

Table A.1. *Table of Fourier transforms.*

1.	$f(t) = \dfrac{1}{2\pi}\displaystyle\int_{\mathbb{R}} \hat{f}(\omega) e^{i\omega t}\, d\omega$	$\hat{f}(\omega) = \int_{\mathbb{R}} f(t) e^{-i\omega t}\, dt$
2.	$f'(t)$	$i\omega \hat{f}(\omega)$
3.	$f^{(k)}(t) = D^k f(t)$	$(i\omega)^k \hat{f}(\omega)$
4.	$f(ax)$, $a \neq 0$	$\dfrac{1}{\lvert a\rvert} \hat{f}\,(\omega/a)$
5.	$f(t-h)$	$e^{-ih\omega} \hat{f}(\omega)$
6.	$(f \otimes g)(t)$	$\hat{f}(\omega)\hat{g}(\omega)$
7.	$e^{iht} f(t)$	$\hat{f}(\omega - h)$
8.	$\displaystyle\int_{-\infty}^{t} f(s)\, ds$	$\hat{f}(\omega)/i\omega$, if $\hat{f}(0) = \displaystyle\int_{\mathbb{R}} f(t)\, dt = 0$
9.	$tf(t)$	$i\dfrac{d\hat{f}(\omega)}{d\omega}$
10.	e^{-at^2}, $a > 0$	$\dfrac{\pi}{a} e^{-\omega^2/4a}$
11.	$e^{-a\lvert t\rvert}$	$\dfrac{2a}{a^2 + \omega^2}$
12.	$te^{-a\lvert t\rvert}$, $a > 0$	$\dfrac{-4ia\omega}{(\omega^2 + a^2)^2}$
13.	$N^1(h;t)$	$\dfrac{2\sin(h\omega/2)}{\omega}$
14.	$N^2(h;t)$	$\dfrac{4\sin^2(h\omega/2)}{h\omega^2}$
15.	$N^m(h;t)$	$\left(\dfrac{2\sin(h\omega/2)}{\omega}\right)^m \dfrac{1}{h^{m-1}}$
16.	$\delta(t-h)$	$e^{-ih\omega}$
17.	$\delta(t-h) + \delta(t+h)$	$2\cos h\omega$
18.	$\delta(t-h) - \delta(t+h)$	$-2i\sin h\omega$

Table A.1$_{/311}$ gives, in the right column, the Fourier transforms of the corresponding functions listed in the left column. All the formulas in this table can be verified easily from the definition of the Fourier transform. Some of the transforms listed are not explicitly used in this book but are included in the table because for some reason it seems natural to do so; for example, item 7 is dual to item 5.

The Fourier transform is closely related to the concept of periodic Fourier series, which can be described as follows. Consider a real- or complex-valued function f in $L^1(\mathbb{R})$. Let f_T denote the truncated function which is equal to $f(t)$ for $|t| < T/2$ and to zero for $|t| \geq T/2$. Then there exists, for $|t| < T/2$, a series expansion

$$f_T(t) = \sum_{n \in \mathbb{Z}} c_n e^{int2\pi/T}, \tag{A.11}$$

where the coefficients c_n are given by the formula

$$c_n = \frac{1}{T} \int_{-T/2}^{T/2} f_T(t) e^{-int2\pi/T}\, dt. \tag{A.12}$$

The convergence properties for the Fourier series are similar to those for the Fourier transform. Using $(A.12)_{/312}$ the formula $(A.11)_{/312}$ may, for $|t| < T/2$, be rewritten as

$$f_T(t) = \sum_{n=-\infty}^{\infty} \hat{f}_T(n2\pi/T) e^{in2\pi t/T} \frac{1}{T}, \tag{A.13}$$

where $\hat{f}_T(\omega)$ denotes the complex Fourier transform of f_T.

We now observe that the formula $(A.13)_{/312}$ is a Riemann sum for the inversion formula

$$f_T(t) = \frac{1}{2\pi} \int_{\mathbb{R}} \hat{f}_T(\omega) e^{i\omega t}\, d\omega$$

sampled at the points $\omega = n2\pi/T$, $n \in \mathbb{Z}$. If $\hat{f} \in L^1(\mathbb{R})$ and if f is piecewise continuously differentiable, one can prove that the right-hand side of $(A.13)_{/312}$ converges to the function $f(t)$ as $T \to \infty$. Therefore, the Fourier transform is a generalization of the concept of periodic Fourier series to functions f that are not periodic.

A.2.2 The Fourier transform: Bivariate case

The Fourier transform has a straightforward generalization to functions of several variables. For simplicity we restrict our attention to the case of two variables. Let $f(y) = f(y_1, y_2)$ be a real- or complex-valued function in $L^1(\mathbb{R}^2)$, i.e., in the class of functions such that $\int_{\mathbb{R}^2} |f(y_1, y_2)|\, dy_1 dy_2 < \infty$. (Note that often in the main text we have used the notation (u, v) in place of (y_1, y_2).)

We have the following definition.

Definition A.2.2.

$$\hat{f}(\omega_1, \omega_2) = \int_{\mathbb{R}^2} f(y_1, y_2) e^{-i\omega_1 y_1} e^{-i\omega_2 y_2}\, dy_1\, dy_2. \tag{A.14}$$

Using a more compact notation, this may be rewritten as

$$\hat{f}(\omega) = \int_{\mathbb{R}^2} f(y) \exp(-i\omega^t y)\, dy, \tag{A.15}$$

Table A.2. *Modified table of Fourier transforms (bivariate case).*

1.	$f(y) = \frac{1}{(2\pi)^2}\int_{\mathbb{R}^2} \hat{f}(\omega)e^{i\omega^t y}\,d\omega$	$\hat{f}(\omega) = \int_{\mathbb{R}^2} f(y)e^{-i\omega^t y}\,dy$
2.	$\frac{\partial f}{\partial y_k}$	$i\omega_k \hat{f}(\omega)$
3.	$\partial^j f(y)$	$(i\omega)^j \hat{f}(\omega)$
4.	$f(ay), \quad a \neq 0$	$\frac{1}{a^2}\hat{f}\,(\omega/a)$
5.	$f(y-h), \quad h^t = (h_1, h_2)$	$e^{-ih^t\omega}\hat{f}(\omega)$
6.	$(f \otimes g)(y)$	$\hat{f}(\omega)\hat{g}(\omega)$
7.	$e^{ih^t y}f(y), \quad h^t = (h_1, h_2)$	$\hat{f}(\omega - h)$
9.	$y_j f(y)$	$i\frac{\partial \hat{f}(\omega)}{\partial \omega_j}$
15.	$N(he^m; y),\ h \in \mathbb{R}$	$h^2 \prod_{i=1}^{m} \frac{\sin(h\omega^t e_i/2)}{h\omega^t e_i/2}$
16.	$\delta(y-h), \quad h^t = (h_1, h_2)$	$e^{-ih^t\omega}$
17.	$\delta(y-h) + \delta(y+h)$	$2\cos h^t\omega$
18.	$\delta(y-h) - \delta(y+h)$	$-2i\sin h^t\omega$

where t denotes transposition, $\omega = (\omega_1, \omega_2)^t$, $y = (y_1, y_2)^t = (u, v)^t$, and $\omega^t y = \sum_{i=1}^{2} \omega_i y_i$. Under certain regularity conditions the following inversion formula is valid:

$$f(y) = \frac{1}{(2\pi)^2}\int_{\mathbb{R}^2} \hat{f}(\omega)\exp(i\omega^t y)\,d\omega. \tag{A.16}$$

Parseval's formula (bivariate case): If $f \in L^2(\mathbb{R}^2)$, then $\hat{f} \in L^2(\mathbb{R}^2)$ and

$$\int_{\mathbb{R}} |f(y)|^2\,dy = \frac{1}{(2\pi)^2}\int_{\mathbb{R}} |\hat{f}(\omega)|^2\,d\omega.$$

More generally, if f and $g \in L^2(\mathbb{R}^2)$, then

$$\int_{\mathbb{R}^2} f(y)\overline{g(y)}\,dy = \frac{1}{(2\pi)^2}\int_{\mathbb{R}} \hat{f}(\omega)\overline{\hat{g}(\omega)}\,d\omega.$$

For the bivariate case we have a modified set of transforms, as shown in Table A.2$_{/313}$. In item 3 of that table, the left-hand entry $\partial^j f(y)$, with $j = (j_1, j_2)$, denotes the partial derivative

$$\frac{\partial^{j_1+j_2} f}{\partial y_1^{j_1}\partial y_2^{j_2}},$$

and the right-hand entry denotes

$$(i\omega)^j = i^{j_1+j_2}\omega_1^{j_1}\omega_2^{j_2}.$$

In items 5, 16, 17, and 18 of the table, the translate is $h^t = (h_1, h_2)$. Item 6 is referred to as the *convolution theorem*. In item 15, h is a scalar. Exercise $2_{/285}$ of Chapter 6 asked for a verification of item 15.

A.2.3 Complex Fourier series

Under certain regularity conditions on the T-periodic function $F(t)$, i.e., $F(t) = F(t+T)$ for all $t \in \mathbb{R}$, we have the following complex Fourier series expansion:

$$F(t) = \sum_{k=-\infty}^{\infty} c_k e^{2\pi i k t/T},$$

where the coefficients c_k are determined by

$$c_k = \frac{1}{T}\int_0^T F(t) e^{-2\pi i k t/T}.$$

We also have Parseval's relation,

$$\frac{1}{T}\int_0^T |F(t)|^2 dt = \sum_{k=-\infty}^{\infty} |c_k|^2.$$

In the bivariate case, if $F(y)$ is doubly periodic with period T, i.e., if $F(u,v) = F(u+T,v) = F(u,v+T)$ for all $(u,v)^t \in \mathbb{R}^2$, then we have

$$\begin{aligned} F(y) = F(u,v) &= \sum_{k\in\mathbb{Z}^2} c_k e^{2\pi i k^t y/T} dy \\ &= \sum_{(k_1,k_2)\in\mathbb{Z}^2} c_{k_1,k_2} e^{2\pi i (k_1 u + k_2 v)/T} du dv, \end{aligned}$$

where

$$\begin{aligned} c_k = c_{(k_1,k_2)} &= \frac{1}{T^2}\int_0^T\int_0^T F(y) e^{-2\pi i k^t y/T} dy \\ &= \frac{1}{T^2}\int_0^T\int_0^T F(u,v) e^{-2\pi i (k_1 u + k_2 v)/T} du dv. \end{aligned}$$

Again, we have Parseval's relation,

$$\frac{1}{T^2}\int_0^T\int_0^T |F(y)|^2 dy = \sum_{k\in\mathbb{Z}^2} |c_k|^2.$$

A.2.4 Discrete complex Fourier series

Let $\{F_j\}_{j=0}^{n-1} \subset \mathbb{C}$ be a finite sequence of complex numbers (for convenience, the sequence is sometimes extended periodically so that $F_{j+n} = F_j$ for all n). Let

$w = e^{2\pi i/n}$. Then we have the following expansion of $\{F_j\}_{j=0}^{n-1}$ and its inverse:

$$F_j = \sum_{r=0}^{n-1} f_r w^{jr}, \tag{A.17}$$

$$f_r = \frac{1}{n} \sum_{j=0}^{n-1} F_j w^{-jr}. \tag{A.18}$$

Proof. Let us define $\{f_r\}_{r=0}^{n-1}$ by $(A.18)_{/315}$. Then

$$\sum_{r=0}^{n-1} f_r w^{jr} = \sum_{r=0}^{n-1} \left(\frac{1}{n} \sum_{k=0}^{n-1} F_k w^{-kr} \right) w^{jr} = \frac{1}{n} \sum_{k=0}^{n-1} \left(\sum_{r=0}^{n-1} w^{r(j-k)} \right) F_k. \tag{A.19}$$

Now, if $j - k = l \in \mathbb{Z}$, and if $-(n-1) \le l \le n-1$, $l \neq 0$, then

$$\sum_{r=0}^{n-1} w^{rl} = \frac{1 - w^{nl}}{1 - w^l} = 0.$$

If $l = j - k = 0$, then $\sum_{r=0}^{n-1} w^{rl} = n$. It follows that $\sum_{r=0}^{n-1} f_r w^{rj} = F_j$.

In the same way we can show that $(A.17)_{/315}$ implies $(A.18)_{/315}$. Consequently, any sequence $\{F_j\}_{j=0}^{n-1}$ can be expanded as in $(A.17)_{/315}$ in a unique way, with coefficients determined by $(A.18)_{/315}$. □

A.3 Regularity for box-spline nodal functions

The regularity of box-spline nodal functions is determined by the parameter α in Definition $3.3.1_{/111}$ and Theorem $3.3.2_{/111}$. In this section we prove this theorem, beginning in Section A.3.1 with a derivation of the Fourier transform of box-spline nodal functions.

A.3.1 Fourier transforms of box-spline nodal functions

As described in Section A.2.2, the two-dimensional Fourier transform is defined for $f \in L^1(\mathbb{R}^2)$ as

$$\hat{f}(\omega) = \int_{\mathbb{R}^2} f(y) \exp(-i\omega^t y)\, dy, \tag{A.20}$$

where t denotes transposition, $\omega = (\omega_1, \omega_2)^t$, $y = (y_1, y_2)^t = (u, v)^t$, and $\omega^t y = \sum_{i=1}^{2} \omega_i y_i$. If $\hat{f} \in L^1(\mathbb{R}^2)$, we have the following inversion formula:

$$f(y) = \frac{1}{(2\pi)^2} \int_{\mathbb{R}^2} \hat{f}(\omega) \exp(i\omega^t y)\, d\omega.$$

Introducing the convolution $f \otimes g$ defined by

$$(f \otimes g)(w) = \int_{\mathbb{R}^2} f(w - s) g(s)\, ds,$$

we have the following formula (the convolution theorem):

$$(f \otimes g)\hat{}(\omega) = \hat{f}(\omega)\hat{g}(\omega). \tag{A.21}$$

Further, we may note that

$$\hat{f}(0) = \int_{\mathbb{R}^2} f(y)\, dy.$$

For the functions $N^1(h;t)$ defined in $(2.8)_{/55}$ we have, from item 13 of Table $A.1_{/311}$, the (one-dimensional) Fourier transforms

$$\widehat{N^1}(h;\omega) = \frac{\sin(h\omega/2)}{\omega/2}, \quad \omega \in \mathbb{R}. \tag{A.22}$$

Also, from $(A.21)_{/316}$ we have

$$\widehat{N^m}(h;\omega) = \frac{1}{h^{m-1}} \left(2\, \frac{\sin(h\omega/2)}{\omega} \right)^m = \left(\frac{\sin(h\omega/2)}{h\omega/2} \right)^m \cdot h. \tag{A.23}$$

For the shifted functions $N^1(h;t-h/2)$ and $N^m(h;t-\frac{m}{2}h)$, we have the Fourier transforms

$$\frac{1-\exp(-ih\omega)}{i\omega} \tag{A.24}$$

and

$$\frac{1}{h^{m-1}} \left(\frac{1-\exp(-ih\omega)}{i\omega} \right)^m, \tag{A.25}$$

respectively. These last can be verified by multiplying $(A.22)_{/316}$ by $\exp(-ih\omega/2)$. See Exercises $1_{/323}$ and $2_{/323}$ in Section A.5.

We now have the following theorem.

Theorem A.3.1. *For $2 \le k \le m$ we have*

$$\widehat{N^*}(he^k;\omega) = \frac{1}{h^{k-2}} \prod_{j=1}^{k} \frac{1-\exp(-ihe_j^t\omega)}{i(e_j^t\omega)}. \tag{A.26}$$

Proof. We first calculate the Fourier transform $\widehat{N^*}(he^2;\omega)$.

Using that $y = hc_1e_1 + hc_2e_2$, we get, with

$$T = (e_1, e_2) \quad \text{and} \quad T^t = \begin{pmatrix} e_1^t \\ e_2^t \end{pmatrix},$$

that $y = hTc$.

Further, introducing the characteristic function $\chi(c) = \chi(c_1, c_2)$ defined by

$$\chi(c) = \chi(c_1, c_2) = \begin{cases} 1 & \text{if } 0 \le c_1, c_2 \le 1, \\ 0 & \text{otherwise,} \end{cases}$$

we have

$$N^*(he^2; y) = N^*(he^2; hTc) = d_2\chi(c).$$

We recall that by Remark 3.1.6$_{/96}$,

$$d_2\,|\det(T)| = 1.$$

Next, it is straightforward to verify that

$$\hat{\chi}(\omega) = \frac{1-\exp(-i\omega_1)}{i\omega_1}\,\frac{1-\exp(-i\omega_2)}{i\omega_2}.$$

Further,

$$\begin{aligned}
\widehat{N^*}(he^2;\omega) &= \int_{\mathbb{R}^2} N^*(he^2; y)\exp(-i\omega^t y)\,dy \\
&= |\det(hT)| \int_{\mathbb{R}^2} N^*(he^2; hTc)\exp(-i\omega^t hTc)\,dc \\
&= h^2|\det(T)|d_2 \int_{\mathbb{R}^2} \chi(c)\exp(-ih(T^t\omega)^t c)\,dc \\
&= h^2\hat{\chi}(hT^t\omega) = h^2\hat{\chi}(he_1^t\omega, he_2^t\omega),
\end{aligned}$$

and therefore

$$\widehat{N^*}(he^2;\omega) = \frac{1-\exp(-ihe_1^t\omega)}{ie_1^t\omega}\,\frac{1-\exp(-ihe_2^t\omega)}{ie_2^t\omega}.$$

Next, using the relation (3.7)$_{/97}$ we get

$$\begin{aligned}
\widehat{N^*}(he^k;\omega) &= \int_{\mathbb{R}^2} N^*(he^k; y)\exp(-i\omega^t y)\,dy \\
&= \frac{1}{h}\int_{\mathbb{R}^2}\int_0^h N^*(he^{k-1}; y-te_k)\exp(-i\omega^t y)\,dt\,dy \\
&= \frac{1}{h}\int_0^h \int_{\mathbb{R}^2} N^*(he^{k-1}; y-te_k)\exp(-i\omega^t(y-te_k))\exp((-i\omega^t e_k)t)\,dy\,dt \\
&= \int_{\mathbb{R}^2} N^*(he^{k-1}; y)\exp(-i\omega^t y)\,dy\,\frac{1}{h}\int_0^h \exp((-i\omega^t e_k)t)\,dt \\
&= \widehat{N^*}(he^{k-1};\omega)\frac{1-\exp(-i\omega^t e_k h)}{ih\omega^t e_k}.
\end{aligned}$$

By induction, the proof is complete. □

A.3.2 Proof of Theorem 3.3.2

We can now prove Theorem 3.3.2$_{/111}$.

Proof. By (A.26)$_{/316}$ in Theorem A.3.1$_{/316}$, we conclude that there exists a constant C such that

$$|\widehat{N^*}(he^m;\omega)| \le C\prod_{j=1}^{m}\frac{1}{1+|e_j^t\omega|}.$$

We introduce polar coordinates ρ and φ in the ω-plane by

$$\begin{cases} \omega_1 = \rho\cos\varphi, \\ \omega_2 = \rho\sin\varphi. \end{cases}$$

Next, for the vectors $\{e_j\}_{j=1}^m$ we consider the polar representation

$$e_j^t = |e_j|(-\sin\varphi_j, \cos\varphi_j),$$

so that the vector $(\cos\varphi_j, \sin\varphi_j)^t$ is orthogonal to e_j. Then

$$\begin{aligned} e_j^t\omega &= |e_j|\rho(-\sin\varphi_j, \cos\varphi_j)(\cos\varphi, \sin\varphi)^t \\ &= |e_j|\rho(-\cos\varphi\sin\varphi_j + \sin\varphi\cos\varphi_j) = |e_j|\rho\sin(\varphi-\varphi_j). \end{aligned}$$

We conclude that for some C,

$$|\widehat{N^*}(he^m;\omega)| \le C\prod_{j=1}^{m}\frac{1}{1+\rho|\sin(\varphi-\varphi_j)|}\ .$$

Now, with the notation $y=(u,v)^t$, $\beta=(\beta_1,\beta_2)^t$, and

$$D^\beta = \frac{\partial^{\beta_1}}{\partial u^{\beta_1}}\frac{\partial^{\beta_2}}{\partial v^{\beta_2}},$$

we have

$$(D^\beta N^*(he^m;y))\hat{}\,(\omega) = (-i\omega_1)^{\beta_1}(-i\omega_2)^{\beta_2}\widehat{N^*}(he^m;\omega),$$

and by Fourier's inversion formula, we have

$$D^\beta N^*(he^m;y) = \frac{1}{(2\pi)^2}\int_{\mathbb{R}^2} e^{i\omega^t y}(-i\omega_1)^{\beta_1}(-i\omega_2)^{\beta_2}\widehat{N^*}(he^m;\omega)\,d\omega.$$

Thus, with the notation

$$F(\omega) = \frac{1}{(2\pi)^2}(-i\omega_1)^{\beta_1}(-i\omega_2)^{\beta_2}\widehat{N^*}(he^m;\omega)\,d\omega,$$

we have

$$D^\beta N^*(he^m;y) = \int_{\mathbb{R}^2} e^{i\omega^t y}F(\omega)\,d\omega, \tag{A.27}$$

provided the conditions for use of Fourier's inversion formula are satisfied. By well-known results from integration theory [153], the function $D^\beta N^*(he^m; y)$ on the left-hand side of $(A.27)_{/318}$ is continuous if

$$\int_{\mathbb{R}^2} |F(\omega)|\, d\omega < \infty. \tag{A.28}$$

Now

$$|F(\omega)| \le C\rho^{|\beta|} \prod_{j=1}^{m} \frac{1}{1+\rho|\sin(\varphi-\varphi_j)|}\ ,$$

where $|\beta| = \beta_1 + \beta_2$. Let us introduce the notation

$$G(\rho,\varphi) = \rho^{|\beta|} \prod_{j=1}^{m} \frac{1}{1+\rho|\sin(\varphi-\varphi_j)|}\ .$$

Now, if we can prove that

$$\int_0^\infty \int_0^{2\pi} G(\rho,\varphi)\rho\, d\varphi d\rho < \infty$$

for $|\beta| \le m - \alpha - 1$, then the inequality $(A.28)_{/319}$ follows and the proof will be complete.

We first carry out the integration in the angular variable φ. To begin, we consider the following subintervals of $(0, 2\pi)$:

$$I_j = (\varphi_j - \epsilon, \varphi_j + \epsilon).$$

If e_j and e_k are parallel, then $I_j = I_k$. However, ϵ may be chosen so small that $I_j \cap I_k = \emptyset$ if e_j is not parallel to e_k. The complement of these intervals is denoted by I so that

$$I = (0, 2\pi) \Big\backslash \bigcup_{j=1}^{m} I_j.$$

Now, we observe that if $\varphi \in I_j$, then for some constant C,

$$G(\rho,\varphi) \le C\, \frac{\rho^{|\beta|}}{1+\rho^{m-(\alpha-1)}}\, \frac{1}{1+\rho|\sin(\varphi-\varphi_j)|}.$$

Further, if $\varphi \in I$, then

$$G(\rho,\varphi) \le C\, \frac{\rho^{|\beta|}}{1+\rho^m}\ . \tag{A.29}$$

Now

$$\int_{I_j} G(\rho,\varphi)\, d\varphi \le C\, \frac{\rho^\beta}{1+\rho^{m-(\alpha-1)}} \int_{\varphi_j-\epsilon}^{\varphi_j+\epsilon} \frac{1}{1+\rho|\sin(\varphi-\varphi_j)|}\, d\varphi.$$

Performing the substitution $t = \sin(\varphi - \varphi_j)$ in the last integral, we get

$$\int_{\varphi_j-\epsilon}^{\varphi_j+\epsilon} \frac{1}{1+\rho|\sin(\varphi-\varphi_j)|}\,d\varphi = 2\int_0^{\sin\epsilon} \frac{1}{1+\rho t}\,\frac{dt}{\sqrt{1-t^2}}$$
$$\leq \frac{2}{\cos\epsilon}\int_0^{\sin\epsilon} \frac{1}{1+\rho t}\,dt = \frac{2}{\cos\epsilon}\,\frac{1}{\rho}\ln(1+\rho\sin\epsilon).$$

We conclude that

$$\int_{I_j} G(\rho,\varphi)\,d\varphi \leq C\,\frac{\rho^{|\beta|-1}}{1+\rho^{m-(\alpha-1)}}\,\ln(1+\rho\sin\epsilon).$$

On the set I the sharper inequality (A.29)$_{/319}$ is valid. It follows that for some constant C,

$$\int_0^\infty\int_0^{2\pi} G(\rho,\varphi)\rho\,d\varphi d\rho \leq C\int_0^\infty \frac{\rho^{|\beta|-1}}{1+\rho^{m-(\alpha-1)}}\,\ln(1+\rho\sin\epsilon)\,\rho\,d\rho.$$

Now, the integral in the right-hand side of this equation is finite if $m-(\alpha-1)-|\beta| \geq 2$, i.e., if $|\beta| \leq m-\alpha-1$, and the proof is complete. □

A.4 Products of convergent subdivision polynomials

In Section 5.1, Theorem 5.1.4$_{/195}$ was stated without proof. The proof is given here.

We begin by formulating a purely algebraic lemma concerning products of subdivision polynomials.

Lemma A.4.1. *Let $s(z)$ and $w(z)$ be bivariate polynomials defining affine-invariant subdivision procedures. Then the product polynomial $\psi(z) = s(z)w(z)/4$ also defines an affine-invariant subdivision procedure. Moreover, if $s(z)$ and $w(z)$ have the refined subdivision polynomials*

$$p^\nu(z^{1/2^{\nu-1}}) = \prod_{j=0}^{\nu-1} s(z^{1/2^j}) = \sum_{i\in\mathbb{Z}^2} p_i^\nu z^{i/2^{\nu-1}} \tag{A.30}$$

and

$$\kappa^\nu(z^{1/2^{\nu-1}}) = \prod_{j=0}^{\nu-1} w(z^{1/2^j}) = \sum_{i\in\mathbb{Z}^2} \kappa_i^\nu z^{i/2^{\nu-1}}, \tag{A.31}$$

respectively, then the product polynomial $\psi(z) = s(z)w(z)/4$ has the refined subdivision polynomial

$$\zeta^\nu(z^{1/2^{\nu-1}}) = \prod_{j=0}^{\nu-1} \psi(z^{1/2^j}) = \sum_{\mu\in\mathbb{Z}^2} \zeta_\mu^\nu z^{\mu/2^{\nu-1}}, \tag{A.32}$$

where

$$\zeta_\mu^\nu = 4^{-\nu} \sum_{i \in \mathbb{Z}^2} p_i^\nu \kappa_{\mu-i}^\nu. \tag{A.33}$$

In the case that $s(z)$ and $w(z)$ are univariate polynomials, the same conclusion is valid if the factor $4^{-\nu}$ in (A.33)$_{/321}$ *is replaced by $2^{-\nu}$ and all summations are over $\mathbb{Z}$ instead of $\mathbb{Z}^2$.*

Proof. We carry out the proof for the bivariate case.

By Theorem 4.5.1$_{/172}$, it is clear that the product $\psi(z) = w(z)s(z)/4$ defines an affine-invariant process.

Now the product $\psi(z) = w(z)s(z)/4$ gives the generating function

$$\begin{aligned}
\zeta^\nu(z^{1/2^{\nu-1}}) &= 4^{-\nu} w(z^{1/2^{\nu-1}}) w(z^{1/2^{\nu-2}}) \cdots w(z^{1/2}) w(z) \\
&\qquad \cdot s(z^{1/2^{\nu-1}}) s(z^{1/2^{\nu-2}}) \cdots s(z^{1/2}) s(z) \\
&= 4^{-\nu} \sum_{i \in \mathbb{Z}^2} \kappa_i^\nu z^{i/2^{\nu-1}} \sum_{j \in \mathbb{Z}^2} p_j^\nu z^{j/2^{\nu-1}} \\
&= 4^{-\nu} \sum_{i,j \in \mathbb{Z}^2} \kappa_i^\nu p_j^\nu z^{(i+j)/2^{\nu-1}} \\
&= 4^{-\nu} \sum_{\mu \in \mathbb{Z}^2} \left(\sum_{i \in \mathbb{Z}^2} \kappa_i^\nu p_{\mu-i}^\nu \right) z^{\mu/2^{\nu-1}}.
\end{aligned} \tag{A.34}$$

We conclude that the coefficients in (A.32)$_{/320}$ are given by

$$\zeta_\mu^\nu = 4^{-\nu} \sum_{i \in \mathbb{Z}^2} \kappa_i^\nu p_{\mu-i}^\nu,$$

and the proof is complete. □

Lemma A.4.1$_{/320}$ expresses that the product of two subdivision polynomials has, for nodal functions, refined generating polynomials whose coefficient sequence is the *discrete convolution* of those for the factors. The proof above relies on the fact that multiplying generalized polynomials corresponds to convolving their coefficient sequences.

The following theorem restates Theorem 5.1.4$_{/195}$.

Theorem A.4.2. *Assume that we are given two subdivision polynomials $s(z)$ and $w(z)$ defining convergent subdivision processes in the sense of Definition* 5.1.1$_{/193}$. *Then in the bivariate case, the polynomial $\psi(z) = s(z)w(z)/4$ also defines a convergent process producing the continuous nodal function*

$$N^\psi(h; y) = \frac{1}{h^2} N^s(h; y) \otimes N^w(h; y). \tag{A.35}$$

In the univariate case, the same conclusion is valid with $\psi(z) = s(z)w(z)/2$ and with the factor $\frac{1}{h}$ on the right side of (A.35)$_{/321}$. *Further, the convergence for the subdivision associated with the polynomial ψ is uniform in the sense of Definition* 5.1.1$_{/193}$.

Proof. We carry out the proof for the bivariate case and for $h = 1$. We have to prove that

$$\max_{\mu\in\mathbb{Z}^2} |\zeta_\mu^\nu - N^s \otimes N^w(\mu/2^\nu)| \doteq \epsilon_\nu^\psi \to 0 \quad \text{as} \quad \nu \to \infty,$$

where $N^s \otimes N^w(\mu/2^\nu) = \int_{\mathbb{R}^2} N^s(y)N^w(\mu/2^\nu - y)\,dy$, given that $\epsilon_\nu^s \to 0$ and $\epsilon_\nu^w \to 0$.

First, we present some preliminaries. We note that the number of points in $\mathrm{supp}(N^s) \cap (\mathbb{Z}^2/2^\nu)$ is bounded by $C4^\nu$ for some constant C. Further, the function $N^s(y)N^w(x-y) \doteq F(x,y)$ is continuous as a function of $(x,y) \in \mathbb{R}^2 \times \mathbb{R}^2$ and has compact support. Therefore, if we let

$$\delta(\epsilon) \doteq \max\{|F(x,y) - F(x,y')| : |y - y'| \le \epsilon, x \in \mathbb{R}^2\}, \tag{A.36}$$

it follows by the uniform continuity that $\delta(\epsilon) \to 0$ as $\nu \to \infty$.

By the assumptions $|p_k^\nu - N^s(k/2^\nu)| \le \epsilon^s$ and $|\kappa_k^\nu - N^w(k/2^\nu)| \le \epsilon^s$, it also follows that

$$|p_k^\nu| \le \|N^s\|_\infty + \epsilon_\nu^s \quad \text{and} \quad |\kappa_k^\nu| \le \|N^w\|_\infty + \epsilon_\nu^s. \tag{A.37}$$

Here $\|N^s\|_\infty = \max_y |N^s(y)|$ and $\|N^s\|_\infty = \max_y |N^s(y)|$.

We also introduce some further notation. The rectangles D and D_k are defined by

$$D = \{(u,v) \in \mathbb{R}^2 : |u|, |v| \le 1/2\} \quad \text{and} \quad D_k = k/2^\nu + 2^{-\nu}D$$

and $\mathrm{vol}_2(D_k) = 4^{-\nu}$, where $\mathrm{vol}_2(A)$ is the area of a subset $A \subset \mathbb{R}^2$.

Now

$$\begin{aligned}
&\zeta_\mu^\nu - N^s \otimes N^w(\mu/2^\nu) \\
&\quad = \zeta_\mu^\nu - \int_{\mathbb{R}^2} N^s(x)N^w(\mu/2^\nu - x)\,dx \\
&\quad = \zeta_\mu^\nu - \sum_{k\in\mathbb{Z}^2} \int_{D_k} N^s(x)N^w(\mu/2^\nu - x)\,dx \\
&\quad = \zeta_\mu^\nu - \sum_{k\in\mathbb{Z}^2} \int_{D_k} (N^s(x)N^w(\mu/2^\nu - x) - N^s(k/2^\nu)N^w((\mu-k)/2^\nu))dx \\
&\qquad - \sum_{k\in\mathbb{Z}^2} N^s(k/2^\nu)N^w((\mu-k)/2^\nu).
\end{aligned}$$

The last term is equal to

$$\begin{aligned}
&- \sum_{k\in\mathbb{Z}^2} (N^s(k/2^\nu) - p_k^\nu)N^w((\mu-k)/2^\nu)4^{-\nu} \\
&- \sum_{k\in\mathbb{Z}^2} p_k^\nu(N^w((\mu-k)/2^\nu) - \kappa_{\mu-k}^\nu)4^{-\nu} - \sum_{k\in\mathbb{Z}^2} p_k^\nu\kappa_{\mu-k}^\nu 4^{-\nu}.
\end{aligned}$$

By Lemma A.4.1$_{/320}$ we get

$$\begin{aligned}
&\zeta_\mu^\nu - N^s \otimes N^w(\mu/2^\nu) \\
&= \sum_{k\in\mathbb{Z}^2} \int_{D_k} (F(x,\mu/2^\nu) - F(k/2^\nu))\,dx \\
&\quad - \sum_{k\in\mathbb{Z}^2} (N^s(k/2^\nu) - p_k^\nu) N^w((\mu-k)/2^\nu) 4^{-\nu} \\
&\quad - \sum_{k\in\mathbb{Z}^2} p_k^\nu (N^w((\mu-k)/2^\nu) - \kappa_{\mu-k}^\nu) 4^{-\nu}.
\end{aligned}$$

Using the triangle inequality we get

$$\begin{aligned}
&|\zeta_{\mu/2^\nu}^\nu - N^s \otimes N^w(\mu/2^\nu)| \le \sum_{k\in\mathrm{supp}(N^s)\cap(\mathbb{Z}^2/2^\nu)} \delta(2^{-\nu})4^{-\nu} \\
&\quad + \sum_{k\in\mathrm{supp}(N^s)\cap(\mathbb{Z}^2/2^\nu)} \epsilon_\nu^s \|N^w\|_\infty 4^{-\nu} + \sum_{k\in\mathrm{supp}(N^s)\cap(\mathbb{Z}^2/2^\nu)} \epsilon_\nu^w (\|N^s\|_\infty + \epsilon_\nu^s) 4^{-\nu} \\
&\le C\delta(2^{-\nu}) + \epsilon_\nu^s \|N^w\|_\infty + \epsilon_\nu^w \|N^s\|_\infty + \epsilon_\nu^w \epsilon_\nu^s) \to 0 \quad \text{as} \quad \nu \to \infty.
\end{aligned}$$

This completes the proof. □

A.5 Exercises

1. Verify items 5, 8, and 13 in Table A.1$_{/311}$.
2. Verify (A.24)$_{/316}$ and (A.25)$_{/316}$.

Notes

Chapter 1

1. The terminology "nodal functions" is used, for example, in [124]. In [176, Sec. 2.4.1], these functions are called "fundamental solutions." They are sometimes inaccurately called "basis functions" even in cases where they do not form a basis.

2. An informal estimate, in the context of graphics applications, might be obtained as follows. Suppose that the characters in a scene require the storage of 1,000 triangles. This is a very modest number in the context of current practice (2009). If this data is to be treated (such "treatment" may be quite elaborate on a modern graphics card) at 60 frames per second, as is not unusual in animated graphics, then this corresponds to 60,000 triangles per second.

The number of triangles increases by a factor of 4^ν if subdivision is carried out using a standard quadrilateral splitting technique, where ν is the number of subdivision iterations. Thus, if $\nu = 4$, such an application would require the storage of the information related to 256,000 triangles, and (if subdivision is to be done uniformly over the whole object, and on the fly) treatment by the graphics card of 15 million triangles per second. If we suppose that this is at the limit of the capability of a given graphics card, even quadrupling the card performance would permit only an increase from $\nu = 4$ to $\nu = 5$. Considerations such as these help to explain the interest in adaptive techniques, where subdivision is not applied uniformly over the whole mesh.

On the other hand, another reason for the small number of iterations in practice is that even four subdivision steps may be more than enough for many graphics applications. Except for animated-film applications, and cinematic sequences within computer games, even fewer iterations may be sufficient. This is especially true in contexts (such as character animation) where *perfectly* smooth surfaces are not the goal.

3. The practical question of ensuring that polyhedral meshes are well formed is not a simple one. For example, if it is desired to ensure that the logical structure

corresponds to a two-manifold without boundary embedded in ordinary Euclidean three-dimensional space, as is often required in solid-modelling systems [98], then the usual tools used in such systems can be used [2, 3, 65, 98]. Among these tools are the so-called Euler operators, designed to ensure that a generalized version of the Euler–Poincaré formula remains true with each modification of the logical structure [98, p. 139]. Even the question of ensuring that just the *logical* data structure is valid is a large topic, and crucially important for practical implementations. We content ourselves here with the remark that a logical mesh is a collection of records, stored in the computer and linked together by the mechanism of a data structure. At the level of the logical data structure, a programmer has great freedom, and there is nothing to prevent the creation of a logical data structure that corresponds, say, to the surface referred to as a "Klein bottle" [3, p. 351], [98, p. 43]. Such nonorientable surfaces cannot form the boundary of an object realizable in three-dimensional space. See Exercise $1_{/47}$.

4. See, however, [168, Sec. 7.2.1] and [171], where *nonmanifold* subdivision is described, and [97], where subdivision is extended to three-dimensional deformation lattices of arbitrary topology.

Also, in practical implementations, it is important that the program react to non-locally-planar cases in a stable way, even if the result of the computation has not been mathematically characterized a priori. Examples of such degeneracy are two opposing tetrahedra (mentioned in Section 1.2.1), cubes sharing an edge, 2-gons and 1-gons in the logical mesh, or two or more triangles in the mesh that are disjoint except at a single shared vertex.

The term "arbitrary topology" is sometimes taken to mean also that vertices of arbitrary valence are permitted [176, p. 20].

5. The dual mesh is a generalization of the concept of *dual graph* [138, p. 509]. The definition depends crucially on the fact that the *faces* of the mesh, and their topological relationship with the edges and vertices of the mesh, are specified. It is quite possible, for example, that the same abstract graph can be drawn in two different ways in the plane, and that its dual is not uniquely specified. Similarly, it is possible that the same abstract mesh, having no "dangling edges" but viewed only as a graph made up of vertices and edges, can be drawn on a sphere in two different ways, so that the dual mesh is not uniquely specified [64, p. 114].

Definitions of the dual mesh given in the subdivision-surface literature often apply, strictly speaking, only to meshes without boundary.

The "dual mesh" used in [177] would be described, in our (and the usual) terminology, in terms of the dual of a *linearly subdivided* version of the original mesh.

6. The *direct sum* is also sometimes called the *connected sum*. The connected sum of two surfaces is formed "by cutting a small circular hole in each surface, and then gluing the two surfaces together along the boundaries of the holes" [100, p. 9].

7. Note, however, that any polyhedral mesh can be embedded without self-intersections in $\mathbb{R}^4$.

8. The notation used to name regular tilings, such as those illustrated in Figure $1.12_{/16}$, is explained in [60, p. 59].

9. This terminology corresponds therefore to the standard terminology for tilings of the plane [60, p. 95]. A vertex of a tiling of the plane is called regular if the angle between each consecutive pair of edges incident at the vertex is $2\pi/n$. The triangular tiling $[6^3]$ of the plane using equilateral triangles has $n = 6$ and angle $\pi/3 = 2\pi/6$, and the tiling $[4^4]$ of the plane using squares has $n = 4$ and angle $\pi/2 = 2\pi/4$.

10. The terminology "extraordinary face" is employed less frequently in the literature than "extraordinary vertex," but it is natural to use the former. There is a clear duality between *extraordinary vertices* and *extraordinary faces*. Some authors refer to "irregular faces," which is quite consistent provided "irregular vertices" is used in place of "extraordinary vertices." In [106] extraordinary faces are called "anomalous regions."

11. Dual subdivision can also be described in terms of "vertex splitting" (see for example [176, Ch. 4], [177]).

12. Strictly speaking, an extraordinary face or vertex will not remain in the mesh; rather, it will be replaced by successor faces or vertices, similarly extraordinary, as the subdivision proceeds.

13. It is shown in Section 4.2.1, however, that the $\sqrt{3}$-subdivision method can be interpreted as operating alternately in triangular and hexagonal dual meshes. See also [7].

14. This terminology is used, for example, in [151]. It derives from the Jacobi method [160] for iteratively solving linear equations; this method may be contrasted with the Gauss–Seidel method [160], which uses, at each substep of the iteration, the most recent updated values of solution components.

15. We do not claim that our description is historically accurate, i.e., that the Doo–Sabin and Catmull–Clark algorithms were discovered in this way. In fact, these algorithms were published before the paper of Lane–Riesenfeld [81]. The description in terms of the generation of subdivision algorithms is, however, a convenient way to look at things.

16. These aspects are not completely independent: for example, a triangular or quadrilateral mesh in the form of a sphere in $\mathbb{R}^3$ *must* contain extraordinary vertices. See Exercise $4_{/48}$. Meshes defining other kinds of surfaces may be regular, however. For example, the following types of meshes *may* be regular: any finite mesh without boundary in $\mathbb{R}^3$ other than the sphere; meshes without boundary, such as the Klein bottle, embedded in $\mathbb{R}^N$ for $N > 3$ (recall that meshes do not necessarily represent physical position); meshes with boundary in $\mathbb{R}^3$; and meshes corresponding to infinite grids, such as a tiling of the plane.

17. In [45], p_i^{new} is denoted a_i, and p_{ℓ_j} is denoted A_j.

18. Alternative weights are sometimes used [15, Sec. 6.2], [96, 168].

19. The $\{\sqrt{3}\}^2$ method indicated in the third column of Figure $1.30_{/33}$ appears in both the upper and lower rows: it is applicable in a triangular mesh with vertices of arbitrary valence $n \geq 3$ (upper row of the table), and in the special case of a regular mesh having vertices with valence $n = 6$ (lower row of the table), the method can be viewed as a General-subdivision-polynomial method based on trisection (see Section 4.2.1).

The Butterfly method could similarly be shown in both the upper and lower rows, since it is also applicable in the nonregular case. In Figure $1.30_{/33}$, however, it has been shown only in the lower row of the third column. The reason is that, as mentioned in Section 4.2.3, the method may produce undesirable surfaces at nonregular vertices, and it is therefore replaced by the Modified Butterfly method in the nonregular case.

20. The vector is a vector of row vectors in $\mathbb{R}^N$, so that N separate ordinary matrix-vector multiplications are involved.

21. Although we do not study this possibility here, there may be situations where it is useful to replace affine combinations by alternative interpolation operations. For example, two distinct points $p_0, p_1 \in \mathbb{R}^3$, both at distance 1 from the origin, can be viewed as describing a circular arc of radius 1 lying in the plane defined by the three points p_0, p_1, and the origin. Let Ω be the angle subtended by that arc: $\cos\Omega = p_0 p_1$. Then spherical linear interpolation *(slerp)* can be used to replace a linear interpolation $(1-t)p_0 + tp_1$:

$$slerp(p_0, p_1; t) = \frac{\sin((1-t)\Omega)}{\sin\Omega} p_0 + \frac{\sin(t\Omega)}{\sin\Omega} p_1,$$

which is a vector of length 1 making the angle $t\Omega$ with p_0 and the angle $(1-t)\Omega$ with p_1. This idea is based on quaternions [148].

Similarly, the process of normal specification, in the context of joining piecewise-smooth subdivision surfaces [13, p. 20], [15, Sec. 6], involves replacement of scalar subdivision coefficients by matrices.

22. Nonstationary processes of various kinds have, however, been studied, in particular in the context of exponential B-splines [168, Sec. 4.3] and schemes capable of representing general surfaces of revolution [101], [168, Sec. 7.2.3]. For these methods, the subdivision rules depend on the iteration index ν. Another example of nonstationary subdivision is Quasi 4-8 subdivision [162]. The subdivision rules for this method are "geometry sensitive" in the nonregular case, i.e., the rules depend on metric qualities of the control points at a given step.

Chapter 2

23. It may be, however, that the nonuniform case of the classical spline theory will lead to useful methods for feature insertion in the subdivision-surface case [23, 34, 58, 145, 166]. Also, NURBS have been adapted to the subdivision-surface context [102, 145] to produce rational subdivision surfaces. This permits representation of spheres, surfaces of revolution, and similar surfaces. This approach has been criticized, however, because of the nonuniform nature of the subdivision [168, p. 212], and because of inconvenient complexity [15]. There are alternative ways [101] to obtain objects such as surfaces of revolution without using nonuniform methods. See also [112].

24. Most of the subdivision methods discussed in the text can, however, be viewed as performing knot insertion at the midpoints of uniform parametric intervals.

25. The generating-function technique is used in computer science for the analysis of algorithms [72, Sec. 1.2.10], the enumeration of binary trees [72, Sec. 2.3.4.4], and other applications.

26. One manifestation of this principle, in this book, is the following. Subdivision often involves repeated averaging, and the binomial coefficients and the binomial theorem often give relevant information about the corresponding generalized polynomial.

27. To prove the multiplication property, multiply the two polynomials and compare coefficients.

28. The support of a function is the closure of the subset of the domain of the function on which the function is nonzero. It is denoted supp. If the support of a function is compact, it is said to have *compact support*. In the case when the domain of the function is in $\mathbb{R}^2$, the support is compact if and only if it is bounded.

29. Methods with m odd are examples of dual methods, of which the most important are the Chaikin method, its generalization to nonregular surface meshes (the Doo–Sabin method), and a box-spline method that we denote $\{\text{Midedge}\}^2$.

30. Thus, for example, at the beginning of [34] it is stated that "...the main idea behind subdivision is to iterate upsampling and local averaging to build complex geometric shapes."

31. The values given in $(2.41)_{/69}$ are evident from Figure $2.8_{/70}$, but they can also be derived explicitly. First, j is either even ($j = 2l$) or odd ($j = 2l + 1$), $l \in \mathbb{Z}$. If $j = 2l$, we can write $j - \frac{1}{2} = [2(l-1)+1] + \frac{1}{2}$, and using $(2.40)_{/69}$ with l replaced by $l - 1$, $q^1_{j-\frac{1}{2}} = p_{(l-1)+1} = p_l$. Since $q^1_{j+\frac{1}{2}}$ is also equal to p_l, we have $q^2_j = p_l$. On the other hand, if $j = 2l + 1$, we can write $j - \frac{1}{2} = 2l + \frac{1}{2}$ and $j + \frac{1}{2} = 2l + 1 + \frac{1}{2}$,

from which it follows that $q^1_{j-\frac{1}{2}} = p_l$ and $q^1_{j+\frac{1}{2}} = p_{l+1}$, so that $q^2_j = (p_l + p_{l+1})/2$. These values are associated with $\frac{h}{2}\mathbb{Z}$.

Note that the multiplicative factor corresponding to substeps 1 and 2 is

$$\left(\frac{z^{1/2} + z^{-1/2}}{2}\right)^2 = \frac{1}{z} \cdot \frac{1}{2}(1+z) \cdot \frac{1}{2}(1+z). \tag{1}$$

Each of the factors $\frac{1}{2}(1+z)$ corresponds to an averaging, and the factor $\frac{1}{z}$ corresponds to a shift of one index value in the refined grid. In the case of substeps 1 and 2, the first averaging results in the cancellation of the factor of 2 introduced by upsampling, by averaging $2p_l$ with 0 to produce the control points $\ldots, p_l, p_{l+1}, \ldots$; the second factor $\frac{1}{2}(1+z)$ corresponds to linear subdivision, producing the averages $\frac{1}{2}(p_l + p_{l+1})$ at new nodes in the refined grid.

Equation $(1)_{/330}$ illustrates the advantage of having used centered basis functions. The right-hand side of $(1)_{/330}$ corresponds to a process that remains in the primal grid. The factors $\frac{1}{2}(1+z)$ correspond to averaging to the left (staying in the primal grid): at each substep, q_j is replaced by $(q_{j-1} + q_j)/2$ for all indices j. The shift of index corresponding to the factor $\frac{1}{z}$ is necessary to correct the effect of doing two such averagings "to the left." In contrast, the left-hand side of $(1)_{/330}$ corresponds to two averagings, where the result of the first averaging is placed in the dual grid, and the result of the second averaging is placed in the dual of the dual grid, i.e., the primal grid. This corresponds to linear subdivision, as illustrated in Figure $2.8_{/70}$.

The operations corresponding to the right-hand side of $(1)_{/330}$ can be described in more detail. The process corresponding to $\frac{1}{2}(1+z)$ corresponds to replacing q_j by $(q_{j-1} + q_j)/2$ for all indices, always remaining in the primal refined grid. This operation is performed twice on the sequence

$$q^0_{2l-2} = 2p_{l-1}, \;\; q^0_{2l-1} = 0, \;\; q^0_{2l} = 2p_l, \;\; q^0_{2l+1} = 0, \;\; q^0_{2l+2} = 2p_{l+1},$$

producing after the first substep the sequence

$$q^1_{2l-2} = p_{l-1}, \;\; q^1_{2l-1} = p_{l-1}, \;\; q^1_{2l} = p_l, \;\; q^1_{2l+1} = p_l, \;\; q^1_{2l+2} = p_{l+1}$$

(this corresponds to "constant subdivision"), and after the second substep,

$$q^2_{2l-2} = (p_{l-2} + p_{l-1})/2, \;\; q^2_{2l-1} = p_{l-1}, \;\; q^2_{2l} = (p_{l-1} + p_l)/2,$$

$$q^2_{2l+1} = p_l, \;\; q^2_{2l+2} = (p_l + p_{l+1})/2$$

(this corresponds to "linear subdivision"). Finally, multiplying by $\frac{1}{z}$ gives the sequence

$$q^2_{2l-2} \leftarrow q^2_{2l-1} = p_{l-1}, \;\; q^2_{2l-1} \leftarrow (p_{l-1} + p_l)/2, \;\; q^2_{2l} \leftarrow p_l,$$

$$q^2_{2l+1} \leftarrow (p_l + p_{l+1})/2, \;\; q^2_{2l+2} \leftarrow p_{l+1},$$

which corresponds to linear subdivision.

It is not incorrect to describe the process in this way, but it is more natural to think of the averages as being placed in the dual grid, and, at the next substep,

in the dual of the dual grid, etc. Thus, in actual implementations, if the number of substeps is odd, we have a dual subdivision method, with the revised control points at the end of a full step stored in a mesh that is dual to the mesh existing at the beginning of the full step.

Note that the shifting of indices, associated with the use of uncentered nodal functions, gets worse as the number of substeps increases.

32. What we have called the "unit-impulse function" is also referred to as the "Dirac polygon" [75].

33. The following technical remark, concerning the presence of the factor $z^{-m/2}$ in $(2.20)_{/61}$, may be useful in the context of the evaluation of $s(-1)$ and $s(1)$.

The factor $z^{-m/2}$ was introduced in $(2.20)_{/61}$ by the centering of the subdivision polynomial. This and similar factors, introduced later, do not lead to any mathematical difficulty, when we evaluate $s(z)$, since we never have to consider values near $z = 0$. We may define $z = re^{i\alpha}$, $-\pi/2 < \alpha < 3\pi/2$, so that $(-1)^{1/2} = i$, and $(-1)^{k/2} = i^k$ for all $k \in \mathbb{Z}$. With this definition, -1 is well defined as a zero of $z^{-1/2}((1+z)/2) = r^{-1/2}e^{-i\alpha/2}((1+z)/2)$. We may also note for later use that $z^{1/2}$ is infinitely differentiable away from the branch point $z = 0$. In the case discussed here, this means that -1 is an m-fold zero of $z^{-m/2}((1+z)/2)^m$, and therefore all derivatives of order less than or equal to $m-1$ vanish at $z = -1$.

Chapter 3

34. It will be observed from the proof of Theorem $3.2.9_{/107}$ that when we talk about functions that are piecewise polynomials, we ignore their values on the boundaries of the pieces. In fact, since they are polynomials, the function values and derivatives are well defined as we approach the boundary, although these limits can be different if we approach the boundary from different subdomains. The overall regularity of the box-spline nodal functions is discussed in Sections 3.3 and A.3.

Chapter 4

35. Such a change in normalization factor will always be necessary if we decompose methods in this way.

A box spline with a subdivision polynomial that is the product of two subdivision polynomials, each of which corresponds to a box spline, can be viewed as corresponding to consecutive applications of the two constituent box splines. The normalization for achieving affine invariance, however, is done once at the end, and the normalization factor is not just the product of the normalization factors for the two constituent box splines applied separately.

A transparent example to illustrate this can be described in terms of Figure $2.8_{/70}$. We may start with the sequence $\ldots, 0, p_{\ell-1}, 0, p_\ell, 0, p_{\ell+1}, \ldots$ corresponding to $p(h; z^2)$ and apply linear subdivision, represented by the subdivision polynomial $s(z) = 2\big(\frac{z^{\frac{1}{2}}+z^{-\frac{1}{2}}}{2}\big)^2$, as in $(2.42)_{/70}$. Note that $s(1) = 2$, as

required for affine invariance: the total weight of points in the refined grid has doubled.

On the other hand, we could define another box spline (let us call it double dual averaging), which places a value in the refined dual mesh by taking *twice* the average of the adjacent control points. The subdivision polynomial corresponding to double dual averaging is $s_d(z) = 2\big(\frac{z^{\frac{1}{2}}+z^{-\frac{1}{2}}}{2}\big)$, and it is affine invariant since $s_d(1) = 2$. If we apply double dual averaging once to the sequence corresponding to $p(h; z^2)$, we obtain constant subdivision, as illustrated by the black squares in Figure 2.8$_{/70}$: the boxes preceding and following the grid point with value p_ℓ are both assigned the value $2(0 + p_\ell)/2 = 2(p_\ell + 0)/2 = p_\ell$.

Now, having defined the methods *linear subdivision* and *double dual averaging*, we may decide to view the former as two consecutive applications of double dual averaging (constant subdivision followed by an averaging; see Figure 2.8$_{/70}$). We must, however, renormalize by dividing by two: simply taking twice the average at both substeps would give a value two times too large.

Expressed in terms of the subdivision polynomials, $s(z) = \frac{1}{2}s_d(z)s_d(z)$. We may view linear subdivision as two applications of double dual averaging if we wish, but a renormalization is necessary to get affine invariance.

36. The lemma is true for arbitrary distributions with compact support. In the theorem, we actually need the result for arbitrary linear combinations of delta functions, in order to apply the result to $S = S(y)$, but we give a proof only for continuous functions.

37. A parametric domain can also be described in general terms by appealing to a theorem which states that any polyhedron can be embedded in $\mathbb{R}^4$ [176, p. 52]. Again, however, it is more convenient for our purposes to have an explicit representation of the parametric domain.

38. A subset of a topological space is compact if every open covering of the set has a finite subcovering.

If the space is $\mathbb{R}^N$, then compactness is equivalent to boundedness and closedness. If we consider the two-dimensional manifold $\boldsymbol{M}$ in Section 4.6, a set is compact if and only if it is closed and a subset of a finite union of faces $\boldsymbol{F}_\alpha$.

Chapter 5

39. The principle here is also the basic principle of the power method for finding the eigenvectors of a matrix.

40. One aspect of the analysis that is sometimes left unclear is the exact choice of parametric domain. In our presentation we make the link with the definitions of charts and atlases given in Section 4.6. This is similar to the parametric domain used in [124] (see Example 4.7.5$_{/184}$). An alternative approach is used in [172, 173, 174].

A second aspect of the analysis that we attempt to clarify is the relationship amongst the eigenstructures of various subdivision matrices relevant for one

specific method. We remarked early in the book that the literature often refers to "the" subdivision matrix, without specifying which of several possibilities has been chosen. The confusion is accentuated by the fact that spectral properties of the subdivision matrix may be described, still without specifying which matrix is involved. The explanation of this riddle is that the eigenstructures of the relevant choices of subdivision matrix are related, and the crucial properties needed for analysis are common to them all. Schweitzer [144] provides a clarification similar to ours, in the context of the Loop method.

A third aspect of the analysis that should be clear from our presentation is the use of the discrete Fourier transform in the spectral analysis of local subdivision matrices.

Finally, we have tried to show clearly which tools are necessary to accomplish which tasks. In particular, spectral analysis of subdivision matrices, corresponding to a certain neighbourhood of the nonregular point, is used to show simple *convergence*. Spectral analysis of subdivision matrices corresponding to a larger neighbourhood is used to show *smoothness*. Injectivity of the characteristic map is used to guarantee that the limit surface is *single sheeted*.

41. In the literature, the functions $f^j(y) = \sum_{l\in\mathbb{G}_3} \xi_l^j N_l(y)$ are called *eigenbasis functions* [172]. They are the result of applying the subdivision process to an initial sequence of control points being equal to the eigenvector coordinates ξ_l^j, $l \in \mathbb{G}_3$. By $(5.93)_{/236}$, it is clear that they satisfy the scaling relation $f^j(y/2) = \lambda_j f^j(y)$. See [172, p. 51] and also [150] (discussed in Section 6.3.2).

Chapter 6

42. Authors who use "mask" where we have used "stencil" (see Section 1.2.3) refer, of course, to evaluation stencils and tangent stencils as "evaluation masks" and "tangent masks," respectively.

43. If $\{\xi^j\}_j$ is a basis of generalized eigenvectors [56] of a real matrix S, and $\{\eta^j\}_j$ is the dual basis, then the vectors η^j are generalized eigenvectors of the transposed matrix S^t.

More precisely, let $\{\xi^j\}_{j\in I}$ be a basis of generalized eigenvectors and $\{\eta^j\}_{j\in I}$ the dual basis defined by the condition that (η^i, ξ^j) is equal to 1 if $i = j$, and 0 otherwise. Here we use the notation $(\eta, \xi) = \eta^*\xi$ for the possibly complex inner product.

For a fixed eigenvalue λ, we consider a cycle $C(\lambda) = \{\xi^1, \xi^2, \ldots, \xi^m\}$ of generalized eigenvectors (for convenience we assume that they are enumerated from 1 to m) so that $\xi^i = (S-\lambda I)^{i-1}\xi^1 \neq 0, 1 \leq i \leq m$, and $(S-\lambda I)\xi^m = (S-\lambda I)^{m+1}\xi^1 = 0$.

We claim that the vectors $\{\eta^1, \eta^2, \ldots, \eta^m\}$ in the dual basis have the property

$$\eta^{m-i} = (S^t - \bar{\lambda} I)^i \eta^m \neq 0, \qquad 0 \leq i \leq m-1,$$

where $\bar{\lambda}$ is the complex conjugate of λ, and

$$(S^t - \bar{\lambda} I)\eta^1 = (S^t - \bar{\lambda} I)^m \eta^m = 0.$$

To prove these statements, it suffices to prove
(i) $((S^t - \bar{\lambda}I)^i \eta^m, \xi_j) = 1$ if $m = i + j$ and 0 otherwise, for $1 \le i, j \le m$, and
(ii) $((S^t - \bar{\lambda}I)^i \eta^m, \xi^j) = 0$ for $j > m$ and $1 \le i \le m$.

Now

$$\begin{aligned}((S^t - \bar{\lambda}I)^i \eta^m, \xi^j) &= (\eta^m, (S - \lambda I)^i \xi^j) \\ &= (\eta^m, (S - \lambda I)^{i+j-1} \xi^1) = (\eta^m, \xi^{i+j}),\end{aligned}$$

which is equal to 1 if $m = i + j$, and 0 otherwise, and (i) is proved. Next,

$$((S^t - \bar{\lambda}I)^i \eta^m, \xi^j) = (\eta^m, (S - \lambda I)^i \xi^j).$$

Now, the linear subspace spanned by each Jordan cycle is invariant under multiplication by S. Consequently, if $\xi^j \in C(\mu)$ with $C(\mu) \ne C(\lambda)$, then $(S - \lambda I)^i \xi^j \in C(\mu)$ and it follows that $(\eta^m, (S - \lambda I)^i \xi^j) = 0$, and (ii) is proved.

Conditions (i) and (ii) now imply that $((S^t - \bar{\lambda}I)^i \eta^m - \eta^{m-i}, \xi^j) = 0$ for all j, i.e.,

$$(S^t - \bar{\lambda}I)^i \eta^m - \eta^{m-i} = 0, \qquad 1 \le i \le m - 1.$$

For $i = m$ we get $((S^t - \bar{\lambda}I)^m \eta^m, \xi^j) = 0$ for all j, i.e.,

$$(S^t - \bar{\lambda}I)^m \eta^m = (S^t - \bar{\lambda}I)\eta^1 = 0.$$

We conclude that the dual basis $\{\eta^j\}_{j \in I}$ consists of generalized eigenvectors of S^t. Thus, for each cycle $C(\lambda) = \{\xi^1, \ldots, \xi^m\}$, there is a corresponding cycle $C(\bar{\lambda}) = \{\eta^1, \ldots, \eta^m\}$ for the transposed matrix, with the order of the cycle reversed and the eigenvalue replaced by its conjugate.

44. The dimension of the linear system to be solved here is $d + 1$ in the univariate case, $d(d + 1)/2$ in the bivariate case, and $(d + 1)^2$ in the case of bidegree d.

Chapter 7

45. The use of the words "crease," "tagged," "sharp," and "smooth" is fairly consistent in the literature, but not perfectly so, and there is sometimes the possibility of confusion. Within this book, we have tried to use terminology that is both consistent and very close to the language used in the various papers cited.

As stated in the text, the terms "crease edge," "tagged edge," and "sharp edge" are synonymous, in this book and in most of the literature. We have mostly avoided the use of "crease edge," however, because of the danger that the adjectives "crease" and "tagged" themselves might be considered synonyms. They are not: a "crease vertex" is only one among several kinds of "tagged vertex." Thus, we have used "sharp edge" or "tagged edge" throughout the technical description, rather than "crease edge."

46. Nonmanifold objects include, for example, an object formed by the union of two tetrahedra sharing only a single edge, or sharing only a single point. The boundary

of a nonmanifold object can be rigorously defined using homology theory [10, 11], [136, Appendix B]: in the language of this theory, the boundary must be made up of faces forming a 2-cycle. In practical terms, this means that the number of faces (or patches) sharing a single edge must be even. The special case of a "manifold object" corresponds to the situation when a single edge is always shared by two incident faces.

Bibliography

[1] ACIS, *The ACIS 3D Toolkit Guide*, Spatial Technology, 1999.

[2] M. K. Agoston, *Computer Graphics and Geometric Modeling: Implementation and Algorithms*, Springer, Berlin, New York, 2005.

[3] ——, *Computer Graphics and Geometric Modeling: Mathematics*, Springer, Berlin, New York, 2005.

[4] L.-E. Andersson, N. F. Stewart, and M. Zidani, *Conditions for use of a non-selfintersection conjecture*, Computer Aided Geometric Design, 23 (2006), pp. 599–611.

[5] ——, *Error analysis for operations in solid modeling in the presence of uncertainty*, SIAM Journal on Scientific Computing, 29 (2007), pp. 811–826.

[6] I. Babuška and T. Strouboulis, *The Finite Element Method and Its Reliability*, Clarendon Press, Oxford, 2001.

[7] C. Bajaj, S. Schaefer, J. Warren, and G. Xu, *A subdivision scheme for hexahedral meshes*, The Visual Computer, 18 (2002), pp. 343–356.

[8] A. A. Ball and D. J. T. Storry, *A matrix approach to the analysis of recursively generated B-spline surfaces*, Computer-Aided Design, 18 (1986), pp. 437–442.

[9] ——, *Conditions for tangent plane continuity over recursively generated B-spline surfaces*, ACM Transactions on Graphics, 7 (1988), pp. 83–102.

[10] P. Bamberg and S. Sternberg, *A Course in Mathematics for Students of Physics* 1, Cambridge University Press, Oxford, 1992.

[11] ——, *A Course in Mathematics for Students of Physics* 2, Cambridge University Press, Oxford, 1992.

[12] M. Barnsley, *Superfractals*, Cambridge University Press, Cambridge, UK, 2006.

[13] H. Biermann, *Representing and Modifying Complex Surfaces*, Ph.D. thesis, Department of Computer Science, New York University, New York, NY, 2002.

[14] H. Biermann, D. Kristjansson, and D. Zorin, *Approximate Boolean operations on free-form solids*, in Proceedings of SIGGRAPH '01, 2001, pp. 185–194.

[15] H. Biermann, A. Levin, and D. Zorin, *Piecewise smooth subdivision surfaces with normal control*, in Proceedings of SIGGRAPH '00, 2000, pp. 113–120.

[16] H. Biermann, I. M. Martin, F. Bernardini, and D. Zorin, *Cut-and-paste editing of multiresolution surfaces*, ACM Transactions on Graphics, 21 (2002), pp. 312–321.

[17] H. Biermann, I. M. Martin, D. Zorin, and F. Bernardini, *Sharp features on multiresolution subdivision surfaces*, Graphical Models, 64 (2002), pp. 61–77.

[18] J. Bolz and P. Schröder, *Rapid evaluation of Catmull–Clark surfaces*, in Proceeding of the Web 3D Symposium (*Web3d'02*), 2002, pp. 11–17.

[19] G.-P. Bonneau, G. Elber, S. Hahmann, and B. Sauvage, *Multiresolution analysis*, in Shape Analysis and Structuring, L. De Floriani and M. Spagnuolo, eds., Springer, Berlin, New York, 2008, pp. 83–114.

[20] M. Botsch, M. Pauly, L. Kobbelt, P. Alliez, B. Lévy, S. Bischoff, and C. Rössl, *Geometric modeling based on polygonal meshes*, in Eurographics 2008 Tutorial, 2008.

[21] J. W. Brown and R. V. Churchill, *Fourier Series and Boundary Value Problems*, Sixth ed., McGraw–Hill, New York, 2000.

[22] N. A. Carr, J. Hoberock, K. Crane, and J. C. Hart, *Rectangular multi-chart geometry images*, in Proceedings of the Fourth Eurographics Symposium on Geometry Processing (SGP '06), 2006, pp. 181–190.

[23] T. J. Cashman, N. A. Dodgson, and M. A. Sabin, *Non-uniform B-spline subdivision using refine and smooth*, in Mathematics of Surfaces XII, M. Sabin and J. Winkler, eds., Lecture Notes in Computer Science 4647 Springer, Berlin, New York, 2007, pp. 121–137.

[24] E. Catmull and J. Clark, *Recursively generated B-spline surfaces on arbitrary topological meshes*, Computer-Aided Design, 10 (1978), pp. 350–355.

[25] A. S. Cavaretta, W. Dahmen, and C. A. Micchelli, *Stationary subdivision*, Memoirs of the American Mathematical Society, 93 (1991), pp. 1–186.

[26] G. M. Chaikin, *An algorithm for high speed curve generation*, Computer Graphics and Image Processing, 3 (1974), pp. 346–349.

[27] F. Cheng and J. Yong, *Subdivision depth computation for Catmull–Clark subdivision surfaces*, Computer-Aided Design and Applications, 3 (2006), pp. 485–494.

[28] S. W. Choi, B.-G. Lee, Y. J. Lee, and J. Yoon, *Stationary subdivision schemes reproducing polynomials*, Computer Aided Geometric Design, 23 (2006), pp. 351–360.

[29] C. K. Chui and M.-J. Lai, *Algorithms for generating B-nets and graphically displaying spline surfaces on three- and four-directional meshes*, Computer Aided Geometric Design, 8 (1991), pp. 479–493.

[30] E. Cohen, R. F. Riesenfeld, and G. Elber, *Geometric Modeling with Splines*, A K Peters, Wellesey, MA, 2001.

[31] W. W. Curtis and F. R. Miller, *Differentiable Manifolds and Theoretical Physics*, Academic Press, New York, 1985.

[32] I. Daubechies, *Orthonormal bases of compactly supported wavelets*, Communications on Pure and Applied Mathematics, 41 (1988), pp. 909–996.

[33] ——, *Ten Lectures on Wavelets*, CBMS-NSF Regional Conference Series in Applied Mathematics 61, SIAM, Philadelphia, 1992.

[34] I. Daubechies, I. Guskov, and W. Sweldens, *Regularity of irregular subdivision*, Constructive Approximation, 15 (1999), pp. 381–426.

[35] C. de Boor, *On calculating with B-splines*, Journal of Approximation Theory, 6 (1972), pp. 50–62.

[36] ——, *Spline basics*, in Handbook of Computer Aided Geometric Design, G. Farin, J. Hoschek, and M. S. Kim, eds., North–Holland, Amsterdam, 2002, pp. 141–164.

[37] ——, *On the evaluation of box splines*, Numerical Algorithms, 5 (1993), pp. 5–23.

[38] C. de Boor, K. Höllig, and S. Riemenschneider, *Box Splines*, Applied Mathematical Sciences 98, Springer, Berlin, New York, 1993.

[39] G. de Rham, *Un peu de mathématiques à propos d'une courbe plane*, Revue de Mathématiques Élémentaires II, 5 (1947). Oeuvres Mathématiques (Collected Works). L'Enseignement Mathématique, Université de Genève, 1981, pp. 678–689.

[40] ——, *Sur une courbe plane*, J. de Mathématiques Pures et Appliquées, 35 (1956), pp. 24–42. Oeuvres Mathématiques (Collected Works). L'Enseignement Mathématique, Université de Genève, 1981, pp. 696–713.

[41] K. Deimling, *Non-linear Functional Analysis*, Springer, Berlin, New York, 1984.

[42] T. DeRose, M. Kass, and T. Truong, *Subdivision surfaces in character animation*, in Proceedings of SIGGRAPH '98, 1998, pp. 85–94.

[43] M. do Carmo, *Differential Geometry of Curves and Surfaces*, Prentice–Hall, Englewood Cliffs, NJ, 1976.

[44] N. A. Dodgson, *An heuristic analysis of the classification of bivariate subdivision schemes*, in Mathematics of Surfaces XI, R. Martin, H. Bez, and M. Sabin, eds., Lecture Notes in Computer Science 3604, Springer, Berlin, New York, 2005, pp. 161–183.

[45] D. Doo and M. Sabin, *Behaviour of recursive subdivision surfaces near extraordinary points*, Computer-Aided Design, 10 (1978), pp. 356–360.

[46] S. Dubuc, *Interpolation through an iterative scheme*, Journal of Mathematical Analysis and Applications, 114 (1986), pp. 185–204.

[47] H. Dym and H. P. McKean, *Fourier Series and Integrals*, Academic Press, New York, 1972.

[48] N. Dyn, J. Gregory, and D. Levin, *A 4-point interpolatory subdivision scheme for curve design*, Computer Aided Geometric Design, 4 (1987), pp. 257–268.

[49] ——, *A butterfly subdivision scheme for surface interpolation with tension control*, ACM Transactions on Graphics, 9 (1990), pp. 160–169.

[50] M. Eck, T. DeRose, T. Duchamp, H. Hoppe, M. Lounsbery, and W. Stuetzle, *Multiresolution analysis of arbitrary meshes*, in Proceedings of SIGGRAPH '95, 1995, pp. 173–182.

[51] G. Farin, *Curves and Surfaces for CAGD*, Third ed., Academic Press, New York, 1993.

[52] ——, *A history of curves and surfaces in CAGD*, in Handbook of Computer Aided Geometric Design, G. Farin, J. Hoschek, and M. S. Kim, eds., North–Holland, Amsterdam, 2002, pp. 1–22.

[53] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes, *Computer Graphics, Principles and Practice*, Second ed., Addison–Wesley, Reading, MA, 1996.

[54] G. B. Folland, *Real Analysis, Modern Techniques and Their Application*, Second ed., Wiley, New York, 1999.

[55] D. Forsey and R. Bartels, *Hierarchical B-spline refinement*, Computer Graphics, 22 (1988), pp. 205–212.

[56] S. H. Friedberg, A. J. Insell, and L. E. Spence, *Linear Algebra*, Fourth ed., Prentice–Hall, Upper Saddle River, NJ, 2003.

[57] I. Ginkel and G. Umlauf, *Analyzing a generalized Loop subdivision scheme*, Computing, 79 (2007), pp. 353–363.

[58] R. Goldman and S. Schaefer, *Non-uniform subdivision for B-splines of arbitrary degree*, Computer Aided Geometric Design, 26 (2009), pp. 75–81.

[59] C. Grimm and D. Zorin, *Surface modeling and parameterization with manifolds*, in ACM SIGGRAPH '05 Course Notes, 2005.

[60] B. Grünbaum and G. C. Shephard, *Tilings and Patterns*, W. H. Freeman, New York, 1987.

[61] I. Guskov, K. Vidimče, W. Sweldens, and P. Schröder, *Normal meshes*, in Proceedings of SIGGRAPH '00, 2000, pp. 95–101.

[62] S. Hahmann and G. Elber, *Constrained multiresolution geometric modeling*, in Advances in Multiresolution Geometric Modeling, N. A. Dodgson, M. S. Floater, and M. Sabin, eds., Springer, New York, 2005, pp. 119–142.

[63] M. Halstead, M. Kass, and T. DeRose, *Efficient, fair interpolation using Catmull–Clark surfaces*, in Proceedings of SIGGRAPH '93, 1993, pp. 35–44.

[64] F. Harary, *Graph Theory*, Addison–Wesley, Reading, MA, 1972.

[65] C. M. Hoffmann, *Geometric and Solid Modeling*, Morgan Kaufmann, San Francisco, 1989.

[66] H. Hoppe, T. DeRose, T. Duchamp, M. Halstead, H. Jin, J. McDonald, J. Schweitzer, and W. Stuetzle, *Piecewise smooth surface reconstruction*, in Proceedings of SIGGRAPH '94, 1994, pp. 295–302.

[67] K. Hormann and M. Sabin, *A family of subdivision schemes with cubic precision*, Computer Aided Geometric Design, 25 (2008), pp. 41–52.

[68] Z. Huang and G. Wang, *Distance between a Catmull–Clark subdivision surface and its limit mesh*, in Proceedings of ACM Solid and Physical Modeling SPM '07, 2007, pp. 233–240.

[69] ——, *Improved error estimate for extraordinary Catmull–Clark subdivision surface patches*, The Visual Computer, 23 (2007), pp. 1005–1014.

[70] O. Kavian, *Introduction à la Théorie des Points Critiques*, Mathématiques et Applications, Springer, New York, 1993.

[71] M. Kim and J. Peters, *Fast and stable evaluation of box-splines via Bézier form*, Tech. Report REP-2007-422, Department of Computer and Information Science, University of Florida, Gainesville, FL, 2007.

[72] D. Knuth, *The Art of Computer Programming, Volume* I: *Fundamental Algorithms*, Third ed., Addison–Wesley, Reading, MA, 1997.

[73] L. Kobbelt, *Interpolatory subdivision on open quadrilateral nets with arbitrary topology*, in Proceedings of Eurographics '96, 1996, pp. 409–420.

[74] L. Kobbelt, *A variational approach to subdivision*, Computer Aided Geometric Design, 13 (1996), pp. 743–761.

[75] ——, *Tight bounding volumes for subdivision surfaces*, in Proceedings of Pacific Graphics '98, 1998, pp. 17–26.

[76] ——, $\sqrt{3}$*-subdivision*, in Proceedings of SIGGRAPH '00, 2000, pp. 103–112.

[77] ——, *Multiresolution techniques*, in Handbook of Computer Aided Geometric Design, E. Farin, J. Hoschek, and M. S. Kim, eds., North–Holland, Amsterdam, 2002, pp. 343–361.

[78] L. Kobbelt, S. Campagna, J. Vorsatz, and H.-P. Seidel, *Interactive multi-resolution modeling on arbitrary meshes*, in Proceedings of SIGGRAPH '98, 1998, pp. 105–114.

[79] U. Labsik and G. Greiner, *Interpolatory* $\sqrt{3}$*-subdivision*, in Proceedings of Eurographics '00, 2000, pp. 131–138.

[80] M.-J. Lai, *Fortran subroutines for B-nets of box splines on three- and four-directional meshes*, Numerical Algorithms, 2 (1992), pp. 33–38.

[81] J. M. Lane and R. F. Riesenfeld, *A theoretical development for the computer generation and display of piecewise polynomial surfaces*, IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-2 (1980), pp. 35–46.

[82] A. W. F. Lee, W. Sweldens, P. Schröder, L. Cowsar, and D. Dobkin, *MAPS: Multiresolution Adaptive Parameterization of Surfaces*, in Proceedings of SIGGRAPH '98, 1998, pp. 95–104.

[83] B. Levery, S. Petitjean, N. Ray, and J. Maillot, *Least squares conformal maps for automatic texture atlas generation*, ACM Transactions on Graphics, 21 (2002), pp. 362–371.

[84] A. Levin, *Combined subdivision schemes for the design of surfaces satisfying boundary conditions*, Computer Aided Geometric Design, 16 (1999), pp. 345–354.

[85] ——, *Interpolating nets of curves by smooth subdivision surfaces*, in Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, 1999, pp. 57–64.

[86] ——, *Polynomial generation and quasi-interpolation in stationary non-uniform subdivision*, Computer Aided Geometric Design, 20 (2003), pp. 41–60.

[87] G. Li, W. Ma, and H. Bao, *Interpolatory* $\sqrt{2}$*-subdivision surfaces*, in Proceedings of IEEE Geometric Modeling and Processing GMP '04, 2004, pp. 1–10.

[88] G. Li, W. Ma, and H. Bao, $\sqrt{2}$ *subdivision for quadrilateral meshes*, The Visual Computer, 20 (2004), pp. 180–198.

[89] N. Litke, A. Levin, and P. Schröder, *Fitting subdivision surfaces*, in Proceedings of the IEEE Visualization Conference, 2001, pp. 319–324.

[90] ——, *Trimming for subdivision surfaces*, Computer Aided Geometric Design, 18 (2001), pp. 463–481.

[91] C. T. Loop, *Smooth Subdivision Surfaces Based on Triangles*, Master's thesis, University of Utah, Salt Lake City, UT, 1987.

[92] M. Lounsbery, T. DeRose, and J. Warren, *Multiresolution analysis for surfaces of arbitrary topological type*, ACM Transactions on Graphics, 16 (1997), pp. 34–73.

[93] P. J. Lu and P. J. Steinhardt, *Decagonal and quasi-crystalline tilings in medieval islamic architecture*, Science, 315 (2007), pp. 1106–1110.

[94] D. Lutterkort and J. Peters, *Tight linear envelopes for splines*, Numerische Mathematik, 89 (2001), pp. 735–748.

[95] W. Ma, *Subdivision surfaces for CAD—An overview*, Computer-Aided Design, 37 (2005), pp. 693–709.

[96] W. Ma, X. Ma, S.-K. Tso, and Z. Pan, *A direct approach for subdivision surface fitting from a dense triangle mesh*, Computer-Aided Design, 36 (2004), pp. 525–536.

[97] R. MacCracken and K. I. Joy, *Free-form deformations with lattices of arbitrary topology*, in Proceedings of SIGGRAPH '96, 1996, pp. 181–188.

[98] M. Mäntylä, *Solid Modeling*, Computer Science Press, New York, 1988.

[99] D. Marcheix and G. Pierra, *A survey of the persistent naming problem*, in Proceedings of ACM Solid Modeling '02, 2002, pp. 13–22.

[100] W. S. Massey, *Algebraic Topology: An Introduction*, Springer, Berlin, New York, 1967.

[101] G. Morin, J. Warren, and H. Weimer, *A subdivision scheme for surfaces of revolution*, Computer Aided Geometric Design, 18 (2001), pp. 483–502.

[102] K. Müller, L. Reusche, and D. Fellner, *Extended subdivision surfaces: Building a bridge between NURBS and Catmull–Clark surfaces*, ACM Transactions on Graphics, 25 (2006), pp. 268–292.

[103] G. Mustafa, F. Chen, and J. Deng, *Estimating error bounds for binary subdivision curves/surfaces*, Journal of Computational and Applied Mathematics, 193 (2006), pp. 596–613.

[104] G. Mustafa and J. Deng, *Estimating error bounds for ternary subdivision curves/surfaces*, Journal of Computational Mathematics, 25 (2007), pp. 473–484.

[105] D. Nairn, J. Peters, and D. Lutterkort, *Sharp, quantitative bounds on the distance between a polynomial piece and its Bézier control polygon*, Computer Aided Geometric Design, 16 (1999), pp. 613–631.

[106] A. H. Nasri, *Polyhedral subdivision methods for free-form surfaces*, ACM Transactions on Graphics, 6 (1987), pp. 29–73.

[107] ——, *Boundary corner control in recursive subdivision surfaces*, Computer-Aided Design, 23 (1991), pp. 405–410.

[108] ——, *Surface interpolation on irregular networks with normal conditions*, Computer Aided Geometric Design, 8 (1991), pp. 89–96.

[109] ——, *Curve interpolation in recursively generated B-spline surfaces over arbitrary topology*, Computer Aided Geometric Design, 14 (1997), pp. 13–30.

[110] ——, *Interpolation of open B-spline curves by recursive subdivision surfaces*, in Mathematics of Surfaces VII, T. Goodman and R. Martin, eds., Institute of Mathematics and its Applications, Information Geometers, 1997, pp. 173–188.

[111] ——, *Interpolating meshes of boundary intersecting curves by subdivision surfaces*, The Visual Computer, 16 (2000), pp. 3–14.

[112] A. H. Nasri and G. Farin, *A subdivision algorithm for generating rational curves*, Journal of Graphics Tools, 6 (2001), pp. 35–47.

[113] A. H. Nasri and M. Sabin, *Editorial, Special issue on subdivision*, The Visual Computer, 18 (2002), p. 285.

[114] ——, *Taxonomy of interpolation constraints on recursive subdivision curves*, The Visual Computer, 18 (2002), pp. 259–272.

[115] ——, *Taxonomy of interpolation constraints on recursive subdivision surfaces*, The Visual Computer, 18 (2002), pp. 382–403.

[116] T. Ni and I. Castaño, *Efficient substitutes for subdivision surfaces*, in ACM SIGGRAPH '09 Course Notes, 2009.

[117] A. Oppenheim, R. W. Schafer, and J. R. Buck, *Discrete Time Signal Processing*, Prentice–Hall, Upper Saddle River, NJ, 1999.

[118] P. Oswald and P. Schröder, *Composite primal/dual $\sqrt{3}$-subdivision schemes*, Computer Aided Geometric Design, 20 (2003), pp. 135–164.

[119] J. Peters, *C^1-surface splines*, SIAM Journal on Numerical Analysis, 32 (1995), pp. 645–666.

[120] J. Peters, *Surface envelopes*, Tech. Report CISE-TR-01-003, Dept. Computer & Information Science, University of Florida, Gainesville, FL, 2001.

[121] J. Peters and U. Reif, *The simplest subdivision scheme for smoothing polyhedra*, ACM Transactions on Graphics, 16 (1997), pp. 420–431.

[122] ——, *Analysis of algorithms generalizing B-spline subdivision*, SIAM Journal on Numerical Analysis, 35 (1998), pp. 728–748.

[123] ——, *Shape characterization of subdivision surfaces—Basic principles*, Computer Aided Geometric Design, 21 (2004), pp. 585–599.

[124] ——, *Subdivision Surfaces*, Springer, Berlin, New York, 2008.

[125] J. Peters and L.-J. Shiue, *Combining* 4- *and* 3-*direction subdivision*, ACM Transactions on Graphics, 23 (2004), pp. 980–1003.

[126] J. Peters and X. Wu, *On the local linear independence of generalized subdivision functions*, SIAM Journal on Numerical Analysis, 44 (2006), pp. 2389–2407.

[127] L. Piegl and W. Tiller, *The NURBS Book*, Second ed., Springer, Berlin, New York, 1997.

[128] H. Prautzsch, *Smoothness of subdivision surfaces at extraordinary points*, Advances in Computational Mathematics, 9 (1998), pp. 377–389.

[129] H. Prautzsch and W. Boehm, *Box splines*, in Handbook of Computer Aided Geometric Design, G. Farin, J. Hoschek, and M. S. Kim, eds., North–Holland, Amsterdam, 2002, pp. 255–282.

[130] H. Prautzsch and U. Reif, *Degree estimates for C^k-piecewise polynomial subdivision surfaces*, Advances in Computational Mathematics, 10 (1999), pp. 209–217.

[131] K. Pulli and M. Lounsbery, *Hierarchical editing and rendering of subdivision surfaces*, Tech. Report UW-CSE-97-04-07, Department of Computer Science and Engineering, University of Washington, Seattle, 2007.

[132] U. Reif, *A unified approach to subdivision algorithms near extraordinary vertices*, Computer Aided Geometric Design, 12 (1995), pp. 153–174.

[133] ——, *A degree estimate for subdivision surfaces of high regularity*, Proceedings of the American Mathematical Society, 124 (1996), pp. 2167–2174.

[134] ——, *Best bounds on the approximation of polynomials and splines by their control structure*, Computer Aided Geometric Design, 17 (2000), pp. 579–589.

[135] U. Reif and L. Kobbelt, *Preface, Special issue on subdivision algorithms*, Computer Aided Geometric Design, 18 (2001), p. 381.

[136] A. A. G. Requicha, *Mathematical models of rigid solid objects*, Tech. Report TM-28, Production Automation Project, University of Rochester, Rochester, NY, 1977.

[137] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

[138] K. H. Rosen, *Discrete Mathematics and Its Applications*, Third ed., McGraw–Hill, New York, 1995.

[139] M. Sabin, *Interrogation of subdivision surfaces*, in Handbook of Computer Aided Geometric Design, G. Farin, J. Hoschek, and M. S. Kim, eds., North–Holland, Amsterdam, 2002, pp. 309–326.

[140] M. Sabin, *Subdivision surfaces*, in Handbook of Computer Aided Geometric Design, G. Farin, J. Hoschek, and M. S. Kim, eds., North–Holland, Amsterdam, 2002.

[141] D. Salomon, *Curves and Surfaces for Computer Graphics*, Springer, New York, 2006.

[142] P. V. Sander, Z. J. Wood, S. J. Gortler, J. Snyder, and H. Hoppe, *Multi-chart geometry images*, in Proceedings of the 2003 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing SGP '03, 2003, pp. 146–155.

[143] I. Schoenberg, *Contributions to the problem of approximation of equidistant data by analytic functions*, Quarterly of Applied Mathematics, 4 (1946), pp. 45–99.

[144] J. E. Schweitzer, *Analysis and Application of Subdivision Surfaces*, Ph.D. thesis, Department of Computer Science and Engineering, University of Washington, Seattle, 1996. Technical Report UW-CSE-96-08-02.

[145] T. W. Sederberg, D. Sewell, M. Sabin, and J. Zheng, *Non-uniform recursive subdivision surfaces*, in Proceedings of SIGGRAPH '98, 1998, pp. 387–394.

[146] A. Sheffer, E. Praun, and K. Rose, *Mesh parameterization methods and their applications*, Foundations and Trends in Computer Graphics and Vision, 2 (2006), pp. 105–171.

[147] P. Shirley and S. Marschner, *Fundamentals of Computer Graphics*, Third ed., A K Peters, Natick, MA, 2009.

[148] K. Shoemake, *Animating rotation with quaternion curves*, in Proceedings of SIGGRAPH '85, 1985, pp. 245–254.

[149] J. Stam, *Evaluation of Loop subdivision surfaces*, in CD-ROM-Proceedings of SIGGRAPH '98, 1998.

[150] J. Stam, *Exact evaluation of Catmull–Clark subdivision surfaces at arbitrary parameter values*, in Proceedings of SIGGRAPH '98, 1998, pp. 395–404.

[151] ——, *On subdivision schemes generalizing uniform B-spline surfaces of high degree*, Computer Aided Geometric Design, 18 (2001), pp. 383–396.

[152] J. Stam and C. Loop, *Quad/triangle subdivision*, in Proceedings of Eurographics '03, 2003, pp. 1–7.

[153] E. M. Stein and G. Weiss, *Introduction to Fourier Analysis on Euclidean Spaces*, Princeton University Press, Princeton, NJ, 1971.

[154] STEP, *International Standard, ISO* 10303–42. *Industrial automation systems and integration—Product data representation and exchange—Part* 42, International Organization for Standardization (ISO), 1997. Reference Number ISO 10303-42 1994(E). First Edition 1994-12-15.

[155] I. F. Stewart and A. R. Foisy, *Arbitrary-degree subdivision with creases and attributes*, Journal of Graphics Tools, 9 (2005), pp. 3–17.

[156] E. J. Stollnitz, T. D. DeRose, and D. H. Salesin, *Wavelets for Computer Graphics*, Morgan Kaufmann, San Francisco, 1996.

[157] G. Strang, *Linear Algebra and Its Applications*, Third ed., Harcourt Brace Jovanovich, New York, 1988.

[158] T. Theoharis, G. Papaioannou, N. Platis, and N. M. Patrikalakis, *Graphics & Visualization, Principles and Algorithms*, A K Peters, Wellesley, MA, 2008.

[159] G. Umlauf, *Analyzing the characteristic map of triangular subdivision surfaces*, Constructive Approximation, 16 (2000), pp. 145–155.

[160] R. S. Varga, *Matrix Iterative Analysis*, Prentice–Hall, Englewood Cliffs, NJ, 1962.

[161] L. Velho, *Quasi* 4-8 *subdivision*, Computer Aided Geometric Design, 18 (2001), pp. 345–357.

[162] L. Velho and J. Gomes, *Quasi* 4-8 *subdivision surfaces*. Manuscript, Instituto de Matemática Pura e Applicada, IMPA, Rio de Janiero, Brazil.

[163] ——, *Variable resolution* 4-*k meshes*, in Proceedings of XIII Brazilian Symposium on Computer Graphics and Image Processing, 2000, pp. 123–129.

[164] L. Velho and D. Zorin, 4-8 *subdivision*, Computer Aided Geometric Design, (2001), pp. 397–427.

[165] G. Wahba, *Spline Models for Observational Data*, CBMS-NFS Regional Conference Series 59, SIAM, Philadelphia, 1990.

[166] J. Warren, *Binary subdivision schemes for functions over irregular knot sequences*, in Mathematical Methods in CAGD III, M. Daehlen, T. Lyche, and L. L. Schumaker, eds., Academic Press, New York, 1995, pp. 1–3.

[167] J. Warren and S. Schaefer, *A factored approach to subdivision surfaces*, IEEE Computer Graphics and Applications, 74 (2004), pp. 74–81.

[168] J. Warren and H. Weimer, *Subdivision Methods for Geometric Design*, Morgan Kaufmann, San Francisco, 2002.

[169] X. Wu and J. Peters, *Interference detection for subdivision surfaces*, in Proceedings of Eurographics '04, 2004, pp. 577–585.

[170] X. Wu and J. Peters, *An accurate error measure for adaptive subdivision surfaces*, in Proceedings of the International Convention on Shapes and Solids SMI-05, 2005, pp. 51–57.

[171] L. Ying and D. Zorin, *Nonmanifold subdivision*, in Proceedings of IEEE Visualization, 2001, pp. 325–332.

[172] D. Zorin, *Stationary Subdivision and Multiresolution Surface Representations*, Ph.D. thesis, California Institute of Technology, Pasadena, CA, 1998.

[173] ——, *A method for analysis of C^1-continuity of subdivision surfaces*, SIAM Journal on Numerical Analysis, 37 (2000), pp. 1677–1708.

[174] ——, *Smoothness of stationary subdivision on irregular meshes*, Constructive Approximation, 16 (2000), pp. 359–397.

[175] ——, *Modeling with multiresolution subdivision surfaces*, in ACM SIGGRAPH '06 Course Notes, 2006, pp. 30–50.

[176] D. Zorin and P. Schröder, *Subdivision for modeling and animation*, in ACM SIGGRAPH '00 Course Notes, 2000.

[177] ——, *A unified framework for primal/dual quadrilateral subdivision schemes*, Computer Aided Geometric Design, 18 (2001), pp. 429–454.

[178] D. Zorin, P. Schröder, and W. Sweldens, *Interpolating subdivision for meshes with arbitrary topology*, in Proceedings of SIGGRAPH '96, 1996, pp. 189–192.

[179] ——, *Interpolating subdivision for meshes with arbitrary topology*, Tech. Report CS-TR-96-06, Computer Science Department, California Institute of Technology, Pasadena, CA, 1996.

[180] ——, *Interactive multiresolution mesh editing*, in Proceedings of SIGGRAPH '97, 1997, pp. 259–268.

Index

Boldface references indicate the pages that should be consulted first in cases where there are several page references for a single entry. These are the pages that give the main idea, or the definition, or the most important fact to be understood about the index entry. If a reference appears alone, however, it is set in ordinary type, even if it is important.