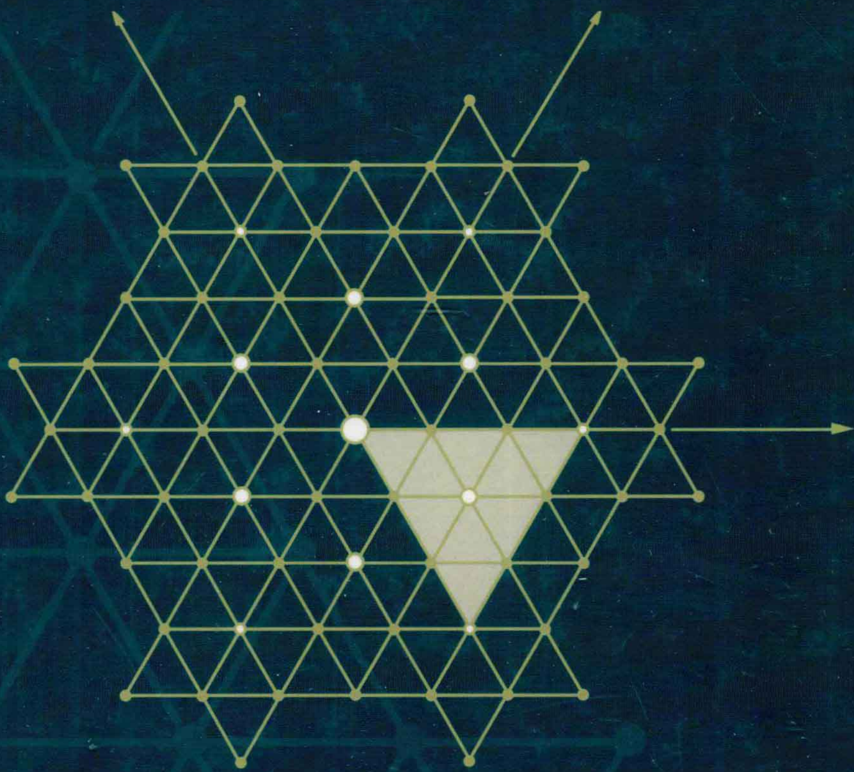
Lars-Erik Andersson · Neil F. Stewart



Introduction to the MATHEMATICS of SUBDIVISION SURFACES

siam.

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

Introduction to the MATHEMATICS of SUBDIVISION SURFACES

To Kristina and Krys

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
Δ_e, Δ, Δ_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^o	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

Contents