REVIEWS AND DESCRIPTIONS OF TABLES AND BOOKS

The numbers in brackets are assigned according to the revised indexing system printed in Volume 28, Number 128, October 1974, pages 1191–1194.

1[2.05.6].—CARL DE BOOR, *A Practical Guide to Splines*, Springer-Verlag, New York, 1978, xxiv + 392 pp., 23 cm. Price $14.80.

Up to now, there has been no comprehensive source of sound, practical information on how to use splines. Many scientists and engineers have recognized the advantages of using splines and piecewise polynomials and then end up with awkward, unstable or inefficient implementations. This book is truly a "practical guide to splines" and the long missing comprehensive reference work for those who use splines. A strong point of the book is that all of the basic algorithms are presented in Fortran so that the accuracy or efficiency of the calculations need not be ruined by sloppy programming. The book is not, however, just a cookbook for using these algorithms. The underlying theory is carefully and concisely developed and even the expert will find new insights and understanding of piecewise polynomials. This book is essential reading for anyone involved in spline calculations.

The first two chapters present background information about polynomial approximation and interpolation. The shortcomings of classical polynomial approximation are presented. The next two chapters present piecewise linear and cubic polynomial interpolation and form a basis for understanding the power and subtleties of piecewise polynomials. Chapter 5 presents the basic theory of approximation for splines, and this is followed by a chapter on parabolic splines which illustrates how to handle the slightly more difficult situation of even degree splines. At the end of these six chapters the reader will have learned a lot about splines and their potential, but he will not yet know how to use them effectively.

The next three chapters address the central problem for applications: the representation of piecewise polynomials. This problem is neglected by theoreticians because they do not intend to use splines, and it is overlooked by most practitioners because they do not appreciate the possibilities and pitfalls inherent in the choice of a basis. Unfortunately, there is no representation which is uniformly superior; there are times when the splines should be represented redundantly by a collection of polynomials and associated domains. The B-spline representation is better, often essential, for most calculations of splines and piecewise polynomials. These topics are presented carefully with illuminating examples.

B-splines are somewhat mysterious at first, and it is not easy to see how to manipulate them effectively. This question is addressed in Chapter 10, and three key algorithms are presented for their stable and efficient manipulation. It is shown how B-splines are equally effective for general piecewise polynomials as for splines.

0025-5718/80/0000-0024/$02.50

Chapter 11 presents the *B*-spline series and establishes the properties (well-conditioned, variation diminishing, etc.) of this series which make splines with this basis so effective in practice. The next chapter further explores the approximation theoretic properties of splines and introduces the important topic of knot placement. This line of investigation is continued in Chapter 13 where interpolation is studied in more depth. By this point all the machinery has been established for all the basic operations with splines.

The next three chapters present the more specialized topics of data smoothing, spline collocation for ordinary differential equations and special splines (taut, periodic and cardinal). Each of these chapters is a thorough development of the problem area complete with Fortran programs. The topics are developed carefully to illustrate how to accomplish things with the previously developed machinery, and they further illustrate the strengths and weaknesses of various ways to apply splines. These chapters contain a lot of original and important new material.

The final chapter introduces the important problem of surface approximation. Only tensor product methods are considered, but these are very important in applications. A very clever scheme is presented to take the tensor product of *Fortran programs* and thus easily extend 1-dimensional algorithms to several dimensions.

The book closes with a short discussion of "things not covered". We can hope that some day de Boor will cover these other important topics with the same elegant and penetrating manner that he used to introduce us to the use of splines.

J. R.

2[2.35].—CLAUDE BREZINSKI, *Algorithmes d'Accélération de la Convergence Etude Numérique*, Editions TechniP, 27 Rue Ginoux, 75735 Paris, France, 1978, xi + 392 pp., 24 cm. Price 195 French francs (≈ U. S. $45.00).

The author's intentions are announced at the beginning of the preface: "This book is addressed to engineers, researchers and students who need to use methods for accelerating convergence in the course of their work; it has, moreover, been written at the suggestion of a number of them. This is then an essentially practical book which presents the algorithms and their applications as well as the corresponding computer programs. . ."

In my opinion the presentation is very clear and easy to understand so that the book can serve as a textbook or a handbook for anyone in the intended audience for whom the French language is not a serious barrier. I recommend it highly.

The publisher has paid considerable attention to a pleasing and uncrowded page layout and the typing is nearly perfect—I noted only a very few errors and they were trivial. Physically, the book is a high quality paperback; its price reflects the current high costs of publishing, especially for a limited audience.

Five basic acceleration algorithms are presented. They are (1) Wynn's epsilon algorithm based on Shanks' transforms, (2) Wynn's rho algorithm based on Thiele's reciprocal differences, (3) Brezinski's theta algorithm which provides a link between the epsilon and rho algorithms, (4) Overholt's algorithm, and (5) Richardson's algorithm.