x_s not rigidly defined, but to have them in a set of sub-intervals $I_s \in [-1, +1]$ (disjoint or not) $s \to 0, 1, \ldots, q$, where they can be freely pre-setted. In other words we suppose we have some degree of freedom in the choice of the individual position of each sampling point though such freedom is limited by the size of the assigned sub-interval.

Under these initial assumptions it may be possible to find a relatively simple characteristic-function that preserves continuity in the higher order derivatives everywhere and at the same time has its zeros in the prescribed sub-intervals $I_{s\to 0,1,\ldots,q}$.

We leave for another paper the detailed study of such characteristic-functions, but we would like to give an example here because of its potential interest:

$$\psi(x) = \omega(\mathrm{e}, x) \sin\left(\frac{\pi}{2} x\right) \quad (6.1)$$

$$\omega(\mathrm{e}, x) = \left(\mathrm{e}^2 + (1 - \mathrm{e}^2) \sin^2\left(\frac{\pi}{2} x\right)\right)^{-\frac{1}{2}} \quad (6.2)$$

The 'eccentricity' parameter e is chosen depending on the set of prescribed sub-intervals I_s.

Acknowledgements

I would like to thank Dr. J. C. P. Miller and Professor B. Conolly for their constructive criticisms (that I have received on many occasions) and Dr. J. K. M. Moody for his opportune suggestions that have greatly improved the presentation of this paper.

References

Davis, P. J. (1963). *Interpolation and approximation*, Blaisdell, New York.
Forsythe, G. E. (1957). Generation and use of Orthogonal Polynomials for Data-Fitting with a Digital Computer, *J. Soc. Indust. and Appl. Mathematics*, 5, pp. 74-88.
Hastings, C. (1955). *Approximations for Digital Computers*, Princeton Univ. Press, Princeton.
Lanczos, C. (1961). *Linear Differential Operators*, van Nostrand, London.
Mineur, H. (1952). *Techniques de Calcul Numerique*, Beranger, Paris.
Oliveira-Pinto, F. (1972). *Numerical Approximation over Sets of Discrete Data Points*, Ph.D. dissertation, Univ. of Cambridge, England.
Rice, J. R. (1964). *Approximation of Functions*, Vol. I, Addison-Wesley, Reading, Mass.
Szegö, G. (1959). *Orthogonal Polynomials*, American Mathematical Soc., New York.

Book reviews

Logic Design Algorithms, by D. Zissos, 1972; 458 pages. (*Oxford University Press*, £9·00)

It seems a long time, though it was only about ten years ago in fact, that Logic Design, as a theoretical study, was a preoccupation of rather a small number of people in this country. Boolean Algebra or Combinational Logic were then being taught in a mere handful of courses and the small group of its devotees were small voices in a large and hostile wilderness; for many people were designing logic networks and designing them pretty well without benefit of Boolean Algebra or academic training and were prepared to say so. Theoretical works on Logic Design and concepts of Sequential Logic imported from America were the mainstay of teachers and academics in the field.

About this time an ebullient and enterprising Greek exploded in our midst, then a teacher of Logic Design in Liverpool. He was so heretical as to say that he didn't believe in reading all the published works that we were studying; this only led to following old ideas and rediscovering cul-de-sacs. It would not be untrue to say that many of us regarded Zissos' ideas with scepticism, if not suspicion. The problems of Logic Design, approached through mathematical techniques seemed pretty formidable—which is not surprising, in retrospect, since Boolean Algebra and actual practical circuit design are quite far removed from each other. Zissos was, and is, a pragmatist. He did not allow himself to be hidebound by algebra but aimed his researches at including the real properties of logic circuits in his design methods; where there was no provision in the algebra he did without it and this must be the key to his success. This book on Logic Design Algorithms is essentially a practical document making it possible, even easy, to design complicated logic networks taking into account fan-in, circuit delays and multi-level solutions. It is thorough and the techniques or algorithms are painstakingly explained and profusely illustrated with examples. Moreover, in spite of his contentions about reading published works the techniques used and the references given show that Zissos has left little unexplored in his search for truth.

Conventional works on Logic Design have hitherto restricted their approach to two-level or three-level implementations such as can be coped with by simple Boolean Algebra. Considerations of fan-in have not been formally introduced in the basic design stages but have been left largely to the intuitive skill of the designer at the implementation stage. Formal methods of design in NAND or NOR logic have been scanty and ineffective or else formidably complicated—anyone who has read Gimpel's excellent paper on the minimising of TANT networks must subscribe to this! But Zissos has picked the bones of this problem, too, and his algorithms do work.

The second half of the book deals with Sequential Logic and again the methods put forward do produce answers. Here Zissos has reached the very proper conclusion that the best, virtually the only sure way, of commencing the design of a sequential circuit is with a state graph; he extends this concept by introducing the idea of 'Internal' and 'External' state graphs and separating them as steps in the method. From then on the algorithms follow the same pattern as those for combinational circuits.

The book is heavy reading and expensive; most people will need to familiarise themselves with Zissos terminology, e.g. 'signal substitution' and 'optional products'. But it is worth reading and the effort is well rewarded. In addition to ample examples, there are included graded test papers with solutions and explanations. This is a very complete book on how to design logic circuits and should, properly, find a wide market.

B. S. Walker (Reading)

ALGOL 60 Programming, by R. F. Shepherd, 1972; 169 pages. (*McGraw-Hill*, £2·50, paperback)

When an experienced programmer is working in an unfamiliar language, and turns to a textbook for guidance, he usually finds that the particular point troubling him is glossed over or ignored altogether, since the author was writing for the beginner. This book, however, does deal with the finer points, the detailed ramifications, as well as the basics of ALGOL 60 programming. The author makes the interesting point that most of us learned grammars of languages and the concepts of formal syntax at school; why not, therefore, make use of these skills when teaching a programming language? I would quarrel with this if applied to a beginner in programming; I learned French by means of syntactic synthesis and analysis, with the result that my conversational French is pitiful, certificates and prizes notwithstanding. But once one has learned to converse with a computer, a deeper understanding of the language structure is highly desirable; while there are a few misprints and places which I would have handled differently, this book can be thoroughly recommended, for the experienced programmer wishing to add ALGOL 60 to his repertoire, or as a companion reference work in parallel with an introductory text of the conventional kind.

B. L. Meek (London)