described in other papers and will give the reader details of the implementation on the Cyber 205, though that machine is no longer being produced.

ELI TURKEL

School of Mathematical Sciences
Tel Aviv University
Ramat-Aviv 69978, Israel

12[68T05, 65K10, 92-04].—EMILE AARTS & JAN KORST, *Simulated Annealing and Boltzmann Machines: A Stochastic Approach to Combinatorial Optimization and Neural Computing*, Wiley Interscience, New York, 1989, xii+272 pp., 24 cm. Price $49.95.

This book is an excellent introduction for mathematicians and physicists to the subjects of simulated annealing and Boltzmann machines. Furthermore, the discussion of Boltzmann machines provides a rigorous foundation with which to penetrate the very trendy subjects of "neural computing" and "neural networks". The book is divided into two sections, the first concentrating on the simulated annealing algorithm, and the second on aspects of Boltzmann machines, especially those pertaining to parallel and neural computation.

The authors motivate the simulated annealing algorithm as a method for solving problems of combinatorial optimization. These problems are generally considered to be very hard to solve, and in particular, all of the examples of combinatorial optimization problems in the book are from the class of NP-complete problems. The simulated annealing algorithm is then presented using the conceptual analogy of the algorithm to metallurgical annealing. The presentation is very general and only requires a minimization problem with a well-defined objective function, C, over a finite and discrete solution space that has a neighborhood structure. Within this mathematical framework the simulated annealing algorithm consists of proposing a neighboring configuration. The proposal is accepted if it either decreases the objective function, or, when the proposed configuration increases the objective function, a uniformly distributed random number chosen in $[0, 1]$ is greater than the value of $e^{-\Delta C/c}$. This is essentially the well-known Metropolis algorithm, where the constant c in the Boltzmann factor is the simulated annealing analog of temperature. This procedure, augmented with a sequence of c values going to zero, making certain that the algorithm reaches the equivalent of thermal equilibrium at every value of c, constitutes the simulated annealing algorithm. The decreasing sequence of c values is called a cooling schedule.

The discussion of the simulated annealing algorithm then continues with practical considerations, implementations of the algorithm for the NP-complete examples, analytic results, and numerical examples. There seems to be an extensive body of results concerning the global asymptotic convergence properties of the algorithm, and two very important results are presented in great detail. The first is an asymptotic result with the assumption of thermal equilibrium at each value of c. The second, more impressive result, shows that the global

asymptotic convergence of the algorithm with a given cooling schedule is possible with only a finite number of iterations at each value of c. This result is based on the analysis of the simulated annealing algorithm as a finite-state Markov chain, and requires only elementary results from the theory of finite-state Markov processes.

The extensive discussion of the simulated annealing algorithm serves as strong motivation for the second section, a discussion of Boltzmann machines. A Boltzmann machine is an interconnected network of elements whose state is either 0 or 1. These binary units are bidirectionally connected with strengths that can take arbitrary positive or negative values. Implicit in a set of connection strengths is the consensus function of the Boltzmann machine which is the sum of the product of the connection strengths and the states of the interconnected units. The computational task of the Boltzmann machine is to maximize its consensus function. This is accomplished with an algorithm analogous to simulated annealing. A unit is chosen for a proposed change in state. This change is accepted if either the consensus function increases or if $1/(1 + e^{-\Delta C/c})$ is smaller than a chosen uniformly distributed random number in the interval $[0, 1]$. Decreasing values of c are then used to "cool" the Boltzmann machine into a near optimal configuration.

This definition of Boltzmann machines shows the clear analogy with the simulated annealing algorithm, and so the homologous asymptotic convergence results for Boltzmann machines that are presented next are predictable. One can augment the definition of the Boltzmann machine to allow the choice of proposed units for transition to be done in a concurrent manner for implementation on a parallel device. However, asymptotic convergence for these parallel Boltzmann machines is still an open problem. Implementational and numerical aspects of Boltzmann machines for the solution of the examples from combinatorial optimization are then presented, concluding the discussion of optimization.

The parallel implementation of the Boltzmann machines leads very naturally into the subject of neural computing. First, the problem of classification for Boltzmann machines is addressed. Here a simple example of the classification of digits in a digital display is carefully presented. Its implementation is with a Boltzmann machine with two layers of units, an input and an output layer. The famous problem of classification for the exclusive-or function is then shown to confound a simple two-layered Boltzmann machine, motivating the consideration of hidden units. With hidden units come the ambiguities in the assignment of connection strengths in Boltzmann machines. This leads to the consideration of learning algorithms for iteratively determining connection strengths that will result in a consensus function with local maxima which correspond to the desired classification groupings.

A very elegant theorem for learning in a Boltzmann machine is then described. Given a current equilibrium distribution of local maxima, q', and a desired equilibrium, q, the divergence function $D(q|q')$ is defined. Min-

imization of this function in the space of connection strength is shown to be equivalent to determining the optimal set of connection strengths for the desired equilibrium distribution. In the special case where the Boltzmann machine has no hidden units, it is proven that $D(q|q')$ is a strictly convex function with a single local minimum. This implies that a steepest descent approach to the minimization of the divergence function is guaranteed to converge. If a Boltzmann machine does have hidden units, $D(q|q')$ is no longer guaranteed to be convex, and heuristic approaches to its minimization are presented.

All in all, the presentation of the material in this book is very balanced. Rigorous results are presented, and an indication of what the authors believe to be the important open problems in the field are included. The Boltzmann machine serves as a fairly rigorous intellectual springboard into the much less rigorous field of neural networks and neural computing. For myself, I found this book an intellectually comforting introduction to this seemingly chaotic new discipline, which clearly marks out the firm ground and the quicksand.

MICHAEL MASCAGNI

Mathematical Research Branch
National Institute of Diabetes, Digestive, and Kidney Diseases
National Institute of Health
Bethesda, Maryland 20892

13[65-01, 65Fxx, 65Kxx].—PHILIPPE G. CIARLET, *Introduction to Numerical Linear Algebra and Optimisation*, Cambridge University Press, Cambridge, 1989, xiv+436 pp., 22 $\frac{1}{2}$ cm. Price $29.95.

This is what appears to be a straight translation of the French original, entitled "Introduction à l'analyse numérique matricielle et à l'optimisation", except that the exercises, which originally were published separately, are now incorporated in the same volume at the end of each subsection. For a review of the original text, see [1].

W. G.

1. V. Thomée, Review 5, Math. Comp. **42** (1984), 713–714.

14[65-00, 65-01, 65-04, 41-00, 41-01, 33-00].—B. A. POPOV & G. S. TESLER, *Computation of Functions on Electronic Computers–Handbook* (in Russian), Naukova Dumka, Kiev, 1984, 599 pp., 21 cm. Price 1 Ruble, 90 Kopecks.

For the user of modern computers or calculators of all sizes, the computation of values of elementary functions—and even of some special functions—has become a simple and common task. This fact, however, should not make us forget that a good deal of mathematics has had to be developed over the last few decades in order to establish the methods which ensure that these computations can be performed in a fast and accurate manner. Several handbooks have