A collection of over 2600 technical papers (some refereed, some not), covering the full range of computer system design and maintained by the Computer Society as a service to the information processing community.

To submit papers: Send two copies of a paper of interest to the computer field to IEEE Computer Society Publications Office, 5855 Naples Plaza, Suite 301, Long Beach, CA 90803.

Include a 50-100 word abstract, a list of index terms (four maximum), and a cover letter giving permission to enter the paper in the Repository (entry in the Repository does not constitute publication).

All submitted material should be unbound, unstapled, printed one side only in black on 8½" X 11" white paper.

Material not conforming to the above requirements will not be accepted.

To order papers: State the R number, listed before the author’s last name, of each paper you order. Prices are 12 cents per page, plus $1.00 service charge for orders under 50 pages.

Microfiche copies are available for $2.50 for manuscripts under 50 pages, plus $2.50 for each additional 50 pages or fraction thereof.

All Repository items must be prepaid except for companies or institutions with established accounts. A $2.00 invoice charge is added to all non-prepaid orders.

Make your check or money order payable to the IEEE Computer Society.


Some L-stable fourth order explicit one-step numerical integration formulas which require no matrix inversion are proposed to cope effectively with systems of ordinary differential equations with large Lipschitz constants (including those having highly oscillatory solutions). The implicit integration procedure proposed in Fatunla is further developed to handle a larger class of stiff systems as well as those with highly oscillatory solutions. This same pair of nonlinear equations is solved for the stiffness/oscillatory parameters. However, the nonlinear systems are transformed into linear forms and an efficient computational procedure is developed to obtain these parameters. The new schemes compare favorably with the backward differentiation formula (DIFSUB) of Gear and the blended linear multistep methods of Skeel and Kong, and the symmetric multistep methods of Lambert and Watson.

R78-12—Lee, D. T. and C. K. Wong, “Voronoi Diagrams in $L_1$, $(L_{oo})$ Metrics with Two-Dimensional Storage Applications” (22 pp., IBM Thomas J. Watson Research Center, Yorktown Heights, New York)

The problem of scheduling the read/write head movement to handle a batch (fixed number) of I/O requests in a two-dimensional secondary storage device in minimum time is studied. Assumed are two models of storage systems in which the access time of a record, being proportional to the “distance” between the position of the record and that of the read/write head, is measured in terms of $L_1$ and $L_{oo}$ metrics respectively. The scheduling problem, referred to as the open path problem (OPP), is equivalent to finding a shortest Hamiltonian path with a specified end point in a complete graph with $n$ vertices. This paper first shows that there exists a natural isometry between the $L_1$ and $L_{oo}$ metrics. Consequently, the existence of a polynomial time algorithm for the OPP in one metric implies the existence of a polynomial time algorithm for the same problem in the other metric. Based on a result by Garey, Graham, and Johnson, it is easy to show that the OPP in $L_1$ (hence in $L_{oo}$) metric is NP-complete. Thus a heuristic to solve the OPP is presented. It is based on a geometric structure, called Voronoi diagram in $L_1$ metric. An optimal algorithm of time complexity $O(n \log n)$ for constructing the diagram for a set of $n$ points in a plane is described.


This report examines the PLC programming language and basic programming techniques. It is intended for use as a supplemental text in introductory programming courses or for use by advanced undergraduates or graduate students unfamiliar with PLC. Included are descriptions of the PLC program statements, program format, arrays and structures, simple searching and sorting techniques, character string manipulation, procedures, advanced language features and programming techniques, and an explanation of PLC error messages.


This paper considers the bin packing problem with the constraint that the elements are in the plane, and only elements within an oriented unit square can be placed within a single bin. The elements are of given weights, and the bins have unit capacities. The problem is to mini-
mize the number of bins used. Since the problem is obviously NP-complete, no algorithm is likely to solve the problem optimally in better than exponential time.

An obvious suboptimal algorithm is considered, and its worst case behavior is analyzed. It is shown that the algorithm guarantees a solution requiring no more than 3.8 times the minimal number of bins. However, a lower bound of 3.75 in the worst case can also be shown. Then the paper generalizes the problem to arbitrary convex figures, and analyzes a class of algorithms in this case. A generalization to multidimensional "bins" (i.e., the weights of points in the plane are vectors, and the capacities of bins are unit vectors) is also considered.

be allocated dynamically as customers arrive. A heuristic is proposed for this problem and for a simple model of the resulting reference sequence, the authors show that the average distance between consecutive references is asymptotically $\frac{3}{2} n$, where $n$ is the size of the storage. For optimal static placement where one waits for all arrivals before any space allocation, the average distance is shown to be asymptotically $\frac{7}{3} n$. For random placement, the average distance is asymptotically $\frac{3}{2} n$. Thus, the heuristic is asymptotically optimal in a strong sense. For reasonable values of $n$, the authors demonstrate that the heuristic is nearly as good as optimal static placement and much better than random placement.

achieve this only with a sufficient number of ladders and is much more complex to implement. Algorithm 2 obviates the need of merging when two or more ladders are in use. With a minor modification in the system design, and hence Algorithm 2 (Algorithm 2'), the sorting time can also be completely hidden by the load-unload time.

R78-19—Stevenson, David, "A Prospectus for High-Speed Computing" (23 pp., Institute for Advanced Computation, Sunnyvale, California)

The application requirements which justify a large, fast computing facility dictate the characteristics of such a system. After considering the problems and alternatives of different architectures, the authors describe an array processor, designed with the benefit of experience with the Illiac IV.

NOW AVAILABLE

R78-22—IEEE COMPUTER SOCIETY REPOSITORY Author and Subject Index

1974-1975 Edition

Lists over 600 technical papers and documents which may be ordered through the Computer Society Repository Service. Keyword subject file covers the full range of computer technology. A valuable permanent reference tool; 108 pp., $11.00.


The authors propose a layout of access channels in a new kind of magnetic bubble memory called a bubble lattice file. Interchange of bubble columns under certain channels is assumed possible. Then an algorithm for moving a column to any other location using minimal number of column interchanges is proposed. The algorithm and the proof of its optimality involves signed-digit representation of numbers by a mixed radix system. Based on this, the authors next propose a simple algorithm to permute a set of columns. Analysis of this algorithm for both the worst and average case is given. The algorithm is shown to be close to optimal for both cases. Finally, based on an optimization consideration, a geometric layout of access channels is suggested for the best performance of the proposed algorithm.

R78-17—Ellis, Robert A., "On the Interactive Use of a Macroprocessor to Generate Operating System Batch Streams" (9 pp., Washington University, St. Louis, Missouri)

A technique which uses a general-purpose macroprocessor to interactively generate command streams for an operating system is described. This technique, although not without disadvantages, permits the easy implementation of new and useful high-level operating system commands.

R78-18—Chin, F. and K. S. Fok, "Fast Sorting Algorithms on Multiple Shift-Register Loops" (44 pp., University of Alberta, Edmonton, Alberta, Canada)

This paper presents two sorting algorithms on the uniform ladder (a new storage device based on CCD or magnetic bubble implementation) proposed by Tung and Chen. Algorithm 1 enables the sorting process to be completely embedded in the I/O time with a single ladder. In contrast, the previous algorithm can

Order by R-number.
Use the Repository order form on p. 114.
Note price increase to $ .12/page.

R78-21—Babich, Alan F., "Proving the Correctness of Parallel Programs" (56 pp., Burroughs Corporation, Mission Viejo, California)

A general method for proving parallel programs correct is presented. The steps are (1) model the parallel program, (2) prove partial correctness (proper synchronization), and (3) prove the absence of deadlock, livelock, and infinite loops. The parallel program model is based on Keller’s model. The main contributions of the paper are in techniques for proving the absence of deadlock and livelock. A connection is made between Keller’s work and Dijkstra’s work with serial non-deterministic programs. It is shown how a variant function may be used to prove finite termination, even if the variant function is not strictly decreasing, and how finite termination can be used to prove the absence of livelock. Handling of the finite delay assumption is also discussed. The illustrative examples include two which occurred in a commercial environment and a classic synchronization problem solved without the aid of special synchronization primitives.