

## Additional Papers

In the December 1991 issue of *The Computer Journal* a new concept of *Additional Papers* was introduced. Such papers will have been typeset and given numbered pages in the same way as other papers, but only the title, authorship and abstract appear in the *Journal*. To obtain copies of the full paper, readers apply to the editor who will mail copies directly to the reader at cost. Libraries will be able to obtain bound volumes at the year end. Reprints will be supplied to authors. For further information, including price lists and order forms, readers and libraries should apply directly to the Editor-in-Chief at the address given on the inside front cover of the *Journal*.

A further list of additional papers is given below.

### Composite B-tree: an Access Aid for Query Processing and Integrity Enforcement

B. C. DESAI, P. GOYAL AND F. SADRI, *Department of Computer Science, Concordia University, Montreal, Canada*

*Various distributed query-processing algorithms have been presented in the literature. Most of these algorithms are concerned with reducing communication costs. In local area networks, like Ethernet, access and processing costs have been shown to dominate communication costs. With the developments taking place in communication technologies, for example the increasing use of fibre optics, the trend towards faster networks will continue. In this paper we present an access aid, the Bc-tree, that reduces access costs and in some cases also the communication costs. We compare the Bc-tree with the semi-join, and the nested-loop and merge-scan join methods. We also discuss optimal query-processing strategies for simple and general tree queries. The Bc-tree also allows for the enforcement of referential and domain integrity constraints at no additional cost.* (pp. A215–A225)

### A Well-Informed Approach to Distributed Task Assignment

C.-C. HSU, S. D. WANG AND T. S. KUO, *Department of Information Management, National Taiwan Institute of Technology, 43 Keelung Road, Sec. 4, Taipei, Taiwan, 10772 R.O.C.*

*This paper addresses the problem of assigning a partitioned task with precedence constraint to a distributed computing system. A new mathematical model is successfully developed to describe the cost function considering communication overhead and ideal time under a general model of distributed computing systems. The task assignment problem is formulated as a DU-mapping, which maps a directed acyclic task graph onto an undirected system graph. The search for an optimal DU-mapping is then transformed into a state space search problem. Using an approach called critical sink underestimation, we can achieve an optimal DU-mapping with many nodes in the state space tree having been pruned. An alternative approach, critical sink overestimation, is utilised to prune more nodes but a suboptimal DU-mapping is then obtained. Combining the two approaches with a divide-and-conquer method, we can further reduce the complexity of the problem and still attain a satisfactory solution. Results from a wide range of experiments reveal that the proposed approaches perform very well in consequence of their giving close approximation to the actual cost and saving high percentage of node generations.* (pp. A227–A236)

### COPS: A Constraint Programming Approach to Resource-Limited Project Scheduling

D. TAY, *School of Building and Estate Management, Faculty of Architecture and Building, National University of Singapore, 10 Kent Ridge Crescent, Singapore 0511*

*Resource-limited scheduling problems exhibit NP-complete characteristics. The combinatorial nature of such problems renders them insusceptible to resolution by mathematical programming techniques. Scheduling of such projects is traditionally solved through the application of heuristics embedding in procedural languages. Using a new generation Artificial Intelligence (AI) constraint programming language (CONDOR), the author has developed a project scheduler called COPS (CONstraint Programming Scheduler). COPS is used to solve a 46-task multiple-resource project. A variety of unique lead/lag relations are modelled in the project. The prolog syntax of this language enhances the representation of project knowledge. Different scheduling heuristics are implemented with ease and their efficacy compared. Built-in numeric and symbolic constraint manipulation techniques expedite problem solution. In addition, the declarative power of the language makes it easy to prototype very concise AI problem solvers. Formulation of the scheduling problem and results are presented.* (pp. A237–A249)

### A Hash-Based Approach for Computing the Transitive Closure of Database Relations

F. FOTOUHI, A. JOHNSON AND S. P. RANA, *Computer Science Department, Wayne State University, Detroit, Michigan 48202, USA*

*Recursive query processing is one of the key problems in integrating database and artificial intelligence technologies. Among the classes of recursive queries, transitive closure queries are the simplest, but most important. Here, we present an efficient hash-based algorithm for computing the transitive closure of database relations. Hashing is used to reduce the data size dynamically. The original data is partitioned once and the partitioning is maintained throughout the computation. Partitions are used in the computation instead of the entire relation. As the new tuples are generated after each iteration of the algorithm, these are placed in the appropriate partitions. We have shown the performance improvement of the proposed algorithm over the existing methods for a wide range of memory sizes, relation depths and hash selectivity factors.* (pp. A251–A259)

### A New Strategy for Multiple Stacks Manipulation

W. P. YANG, T. C. CHIU, W. C. LEE AND S. S. TUNG, *Department of Computer Engineering, National Chiao-Tung University, Hsinchu, Taiwan, R.O.C.*

*New strategy for the maintenance of multiple stacks in a finite storage area is proposed. The idea is motivated by the graceful manner in which two stacks coexist in a sequential area and grow toward each other. Applying this new strategy to Knuth's and Garwick's algorithms, we design two new algorithms, the Bigroup system and the Multigroup system, respectively. By comparison of the simulation results, it is shown that the Multigroup and the Bigroup algorithms have better performance than Garwick's and Knuth's*

algorithms in almost all cases with respect to the number of movements and runtime. When the ratio of memory size to stack number is greater than 10, Garwick's idea of using re-distribute available memory appears to be better than Knuth's. (pp. A261–A266)

#### An Interleaving Algorithm

S. DVOŘÁK, *Tesla Rožnov, 75661 Rožnov pod Radhoštěm, Czechoslovakia*

*An efficient algorithm for in-situ transformation of sequence  $B_1 B_2 \dots B_N A_1 A_2 \dots A_N$  into  $A_1 B_1 A_2 B_2 \dots A_N B_N$  (interleaving) has been designed. It works in  $O(1)$  space and  $O(N \lg_2 N)$  time. The algorithm has been used in one approach to merging problems.* (pp. A267–A270)

#### Point Visibility of Polygonal Regions

K. DELJOUIE-RAKSHANDEH, *BNR Europe Ltd, London Road, Harlow, Essex CM17 9NA*

*Algorithms for computing the visibility graph of general polygonal regions are presented. The main algorithm computes the visibility polygon of a simple polygonal region containing a query point. An extension to this approach deals with the query point lying outside the region, or the region being multiply-connected. Visibility of a set of disjoint polygons is also considered, where the polygons are enclosed in a rectangular region and transformed to obtain a simple region. The resulting simple region is then examined for visibility.* (pp. A271–A279)

#### Configuration of Objects in Parallel Object-Oriented Computation

J. H. HUR AND K. CHON, *Department of Computer Science, Korea Advanced Institute of Science and Technology, P.O. Box 150, Chongryang, Seoul 130-650, Republic of Korea*

*This paper addresses the configuration of objects in parallel object-oriented computation through a configuration language. In parallel object-oriented computation, objects are active and the object creation is dynamic. These characteristics should be reflected in the configuration language. The 'capsule' concept is introduced as the basic unit of distribution and the resource allocation. 'Capsule' is an engineering concept that encapsulates a set of objects as the unit of distribution and resource allocation. It also functions as the bridge for the granularity gap between the computation model and the underlying system.*

*CornerStone configuration language supports the mapping of a set of objects into a set of capsules. It describes the configuration of objects on to capsules based on directives for the initial object decompositions and the declarative attribute definitions for each type. The type-attribute definitions are analysed by the configuration system to construct the capsules to be created at run time. The CornerStone configuration language is effective in supporting the distribution and the parallelism. The fixed distribution of applications in distributed systems as well as the dynamic assignment of objects in self-growing structures in processor network are described in the configuration language.* (pp. A281–A292)

#### Event-Driven Object-Oriented System Modelling

C.-C. D. POO, *Department of Information Systems and Computer Science, Faculty of Science, National University of Singapore, Kent Ridge Road, Singapore 0511*

*Objects in the real world participate in a set of events; they characterise the problem domain and implicitly define the scope of the problem domain. To achieve an accurate representation of the problem domain in a computer system, the analysis model that we derive should closely reflect what is in the real-world problem domain. As the problem domain is characterised by events and objects that participate in the events, starting the analysis process from the perspective of identifying objects and their participations in the events would thus lead us to a model closer to what is required. The discussion here advocates such an approach. With the use of a real-life library environment, this paper will discuss such an approach.* (pp. A293–A307)

#### A Modified Round Robin Scheduler for Time-Sharing Systems

S. HALDAR AND D. K. SUBRAMANIAN, *Department of Computer Science and Automation, Indian Institute of Science, Bangalore 560012, India*

*A modified round robin (MRR, in short) scheduler, for scheduling of processes in general-purpose time-sharing systems, has been presented in this paper. In general, the MRR scheduler provides some kind of higher priority to I/O bound processes, and at the same time retains the simplicity of conventional round robin (RR, in short) schedulers. This scheme introduces two queues with priorities and dependent time slices with the objective of improving response time and its variance for I/O-bound processes in an environment of mixed workload. The MRR scheduler behaves uniformly and fairly with both I/O-bound and CPU-bound processes in such environments. Also, when the workload consists of wholly CPU-bound processes or wholly I/O-bound processes, the performance of the modified scheduler is similar to that of the RR scheduler. We can easily incorporate the modified scheduler in a system which is currently using a RR scheduler.*

*Detailed simulation studies have been performed to evaluate the performance of the modified scheduler, and compare it with that of the RR scheduler. The simulation results show that the MRR scheduler is, indeed, better than the RR scheduler.* (pp. A309–A315)

#### Hierarchical Cellular Graph Automata as a Novel Architecture for Computer-Supported Cooperative Work

E. D. ADAMIDES, PH. TSALIDES AND A. THANAILAKIS, *Laboratory of Electrical and Electronic Materials Technology, Division of Electronics and Information Systems Technology, Department of Electrical Engineering, School of Engineering, Democritus University of Thrace, 67100, Xanthi, Greece*

*This paper introduces a new parallel cellular automata-based system-architecture model for computer-supported cooperative work applications. Cellular graph automata are used to model the topology and the semantics of a typical group-knowledge work application, whereas hierarchical graph cellular automata are introduced to model the functional and communications structure of typical hierarchically structured organisations' operations, which consist of groups of people and resources cooperating and intercommunicating in the execution of local or organisation-level functions. The application of the proposed formalisation method, as well as of the resulting architecture model, are demonstrated by an example of shared workspace management in a real-time conferencing application.* (pp. A317–A328)