doi:10.1093/comjnl/bxl014

Capsule Reviews

FAIROUZ KAMAREDDINE

The Capsule Reviews are intended to provide a short succinct review of each paper in the issue in order to bring the content to a wider readership. The Capsule Reviews were compiled by Fairouz Kamareddine. Professor Kamareddine is an Associate Editor of *The Computer Journal* and is based in the Department of Mathematical and Computer Sciences at Heriot-Watt University, Edinburgh, UK.

Reliability of Data Allocation on a Centralized Service Configuration with Distributed Servers. MEHMET SAVSAR AND FAWAZ AL-ANZI

Most of the earlier studies of data allocation in distributed systems deal with optimum allocation to reduce total cost. The reliability problem in distributed systems has been shown to be NP-hard. Various studies have analysed different aspects of reliability. Most of these studies, however, attempt to approximate the problem by using heuristic rather than exact solutions. This paper aims to develop exact rather than approximate or heuristic solutions. It achieves this goal by analysing the reliability of data allocation on a centralized network service configuration with distributed database servers through the Markov modelling approach. Two models for reliability analysis are introduced. One model is based on the exponential distribution case where time to failure and time to repair are assumed to follow exponential distributions. The other model is based on the Weibull distribution case where the Markov process is used in order to study the probabilities that a database server in one period will continue to function or fail during the next period. Reliability measures are presented for both case models and an extension to the case of three distributed databases is given.

Snap-Stabilizing Depth-First Search on Arbitrary Networks. ALAIN COURNIER, STÉPHANE DEVISMES, FRANCK PETIT, AND VINCENT VILLAIN

A self-stabilizing system is guaranteed to converge to the intended behaviour in finite time. A snap-stabilizing system is a self-stabilizing system which stabilizes in 0 time unit. A number of self-stabilizing algorithms based on depth-first token circulation have been proposed (for arbitrary rooted networks) with a bound on their stabilization time. However, correctness of these self-stabilizing algorithms has only been shown using a (weakly) fair daemon. Furthermore, since self-stabilizing algorithms based on depth-first token circulation for arbitrary networks can be transformed into snap-stabilizing algorithms, we obviously have a number of snap-stabilizing algorithms for arbitrary networks. However, these only work assuming a weakly fair daemon and hence the number of steps cannot be bounded. This paper presents the first snap-stabilizing depth-first search algorithm for arbitrary rooted networks assuming an unfair daemon. This algorithm is proven to be correct and its space and time complexities are studied.

Processing Distance Join Queries with Constraints. APOSTOLOS N. PAPADOPOULOS, ALEXANDROS NANOPOULOS AND YANNIS MANOLOPOULOS

Much research in efficient query processing in spatial and spatiotemporal databases has concentrated on range query, k-nearest neighbor query, spatial join query and k-closest pair query. The latter two types of query processing require much more computation effort and I/O operations than the former two. This paper concentrates on a variation of the so-called k-semi-closest pair query, the k-constrained semi-closest pair query, which imposes spatial constraints between objects, and study efficient query processing algorithms for its computation. These algorithms are extensions and adaptations of existing algorithms such as the naive or multiple nearest-neighbor (MNN) algorithm, the batch nearest-neighbor (BNN) algorithm, and the semi-closest pair (SCP) algorithm. In particular, the proposed probe-and-search (PaS) algorithm which works by means of a sequence of search, probe and batch operations, has reduced CPU cost (due to batching), increased buffer exploitation, low working memory and enforces pruning of the primary tree. The algorithms PaS, BNN, MNN and SCP have been implemented in C++ and a number of experiments conducted. These experiments illustrate that the proposed PaS algorithm has good performance and is expected to outperform other methods.

Medium-Term Scheduler as a solution for the Thrashing Effect. MOSES REUVEN AND YAIR WISEMAN

Multitasking systems execute multiple processes simultaneously. Usually, these processes do not use the whole memory allocated to them. For this reason, many of these processes are stored in virtual memory with only the needed pages stored in the physical memory. Furthermore, the virtual memory scheme is often implemented using a paging concept where a memory page is only loaded into the physical memory when a process requests it. In the absence of a free memory, a page

is swapped from the physical memory of the hard disk. This page swap leads to thrashing, consumes time and slows the system. Various approaches have dealt with this thrashing issue. However, all of these approaches face drawbacks. The present paper suggests a technique that modifies the traditional process scheduling procedure in such a way that fewer pages are swapped in and out. The solution is operating system independent and can be used by any multitasking paging system. In order to avoid thrashing, all the processes are split into several groups (bins) such that the sum of physical memory demands within each group cannot be higher than the physical memory available. To achieve this, a medium-term scheduler which uses the so-called Bin Packing Algorithms is added to the Linux operating system. This medium-term scheduler loads the groups into the ready queue in a Round-Robin manner whereas the Linux scheduler does the usual scheduling but only within the current group. Only at the beginning of each group execution is there intensive swapping overseen by the swapping management. The performance of the kernel has been tested and the results are promising. The medium-term scheduler can drastically reduce the thrashing overhead and can be adapted by many linux machines.

Unconditionally Secure Anonymous Encryption and Group Authentication. GOICHIRO HANAOKA, JUNJI SHIKATA, YUMIKO HANAOKA AND HIDEKI IMAI

In some applications (e.g. in electronic voting), it is important for messages to be exchanged without revealing the identity of some participants. However, if the system is not carefully designed, protecting the identity may be easily compromised. This compromise of identity can happen even in supposedly secure systems. In computationally secure systems, where restrictions are made to take into account the computational difficulties of computing certain problems, the problem of protecting identities can be solved by using public key encryption or group signatures. However, due to progress of technology, the security policy should no longer be based on the difficulty of computationally hard problems. Emphasis must instead be placed on proposing unconditionally secure schemes that provide long-term security and authentication. This paper starts from this point and proposes and studies models, bounds and constructions of novel security primitives with no computational assumptions. The paper starts by defining the so-called 'unconditionally secure asymmetric encryption scheme (USAE)' which is an encryption scheme with unconditional security where a receiver cannot gain any information on any user from an encrypted message. The (tight) lower bounds on required memory sizes for a ciphertext, an encryption key and a decryption key in USAE are given. This is followed by two concrete constructions of USAE which illustrate the advantages of the proposed method with respect to the optimality study of required memory and the flexibility of determining security parameters. These two constructions are compared with respect to these advantages. Then, the USAE method is expanded to deal with multiple receivers giving the so-called MUSAE. A concrete construction of MUSAE is given. Finally, a model, a security definition and a concrete construction of the GA-code (a particular unconditionally secure authentication code with anonymity like group signatures) are given and it is concluded that by combining the USAE with the GA-code, one can obtain without any computational assumptions, a secure communication system which assures confidentiality, authenticity and user's anonymity.

Deterministic Identity-Based Signatures for Partial Aggregation. JAVIER HERRANZ

In traditional public key infrastructures (PKI)-based cryptography, each user generates his own secret and public keys and this is followed by a number of management steps each time the public key needs to be used. This decreases the efficiency of public key systems. On the other hand, in identity-based (ID)-based cryptography, the public key of a user is inferred from his identity. This cryptography style requires less management and, moreover, can be made more efficient by using for example bilinear pairings. In situations where a device must store many signatures, the aggregate signature schemes (used in PKI-based scenarios) are most useful. In such aggregate signature schemes, many signatures on different messages can be aggregated into a single signature which can also be used to verify the correctness of all signatures. This paper proposes to extend this useful concept of aggregate signature schemes to ID-based signature schemes. Unfortunately, this is not possible for existing ID-based signature schemes since the length of the resulting aggregate signatures would be linear in the number of aggregated signatures. For this reason, this paper presents a new ID-based signature scheme suitable for the aggregation of signatures coming from the same signer. This new ID-based signature scheme is deterministic, is shown to be correct and secure, unforgeable under adaptive chosen message attacks and to satisfy good efficiency properties with respect to other ID-based schemes. In addition, this scheme is able to detect possible corruptions of the master entity.

Annotated Unique Input Output sequence generation for conformance testing of FSMs. KARNIG DERDERIAN, ROBERT M. HIERONS, MARK HARMAN and QIANG GUO

Testing can account for up to 50% of the total cost of software development. Usually, implementations of systems specified by finite state machines (FMSs) are tested for conformance to their specifications by applying a sequence of inputs and verifying that the corresponding sequence of outputs is as expected. Since generating complete test suits may not be feasible, generating incomplete test sequences has recently received attention. Test sequences can be generated using different methods. One such method is the Unique Input

Output sequences (UIO) method which has a number of advantages especially in state verification and state transition fault detection, and tend to yield shorter test sequences. These and other advantages motivate an interest in automating the generation of UIOs. This paper describes a method for automatically generating UIO sequences for FSM conformance testing. UIO sequence generation is represented as a search problem and genetic algorithms (GAs) are used to search this space. GAs allow escaping local minima in the search of the global minimum and this is very useful for the search spaces in question. Although GAs have been used previously in the generation of UIOs, this paper improves the earlier approach in many ways and more importantly because it is computationally more efficient and it also generates UIOs for partially (not only fully) specified machines. The experiments conducted in this paper consider real FSMs and randomly generated FSMs and show that the GA presented here outperforms (up to 62% better than) random UIO sequence generation.

Sharbek's algorithm for *t*-ary trees. JAMES F. KORSH

A number of algorithms exist for generating binary or t-ary trees represented by sequences of integers. However, only few of these use linked tree representation and only two generate the next tree from its predecessor. These two are adaptations of Sharbek's algorithm for generating all n nodes ordered trees. This paper generalizes Sharbek's algorithm by providing a new algorithm for generating all n node t-ary trees in linked representation. The paper presents two versions each with a constant average time implementation. The first version generates the next tree from its predecessor given in linked representation and takes average time $O(C\_t)$ per tree generated where C_t increases as t increases. The second version returns additional information about the predecessor beyond just its linked representation and takes average time $O(c\_t)$ per tree generated where c_t converges as t increases.

Improved Algorithms for the *K*-Maximum Subarray Problem. SUNG EUN BAE AND TADAO TAKAOKA

The maximum subarray problem was first used to discuss the efficiency of computer programs. Basically, this problem determines an array portion that sums to the maximum value with respect to all possible array portions within the input array. Obviously, this problem is only interesting if the array contains both negative and positive elements (if the elements are all negative, take the empty array as the solution and if all the elements are all positive, take the entire array as the solution). Solutions to this problem have been studied in the literature for the 1D and 2D cases. In particular, for the 1D case, there is an optimal linear time sequential solution which can be extended for special 2D cases. This paper concentrates on the extension of this problem to the K-maximum sums for both the 1D and 2D cases where the solution is given in sorted order. In the 1D case, the selection of k-largest values from a set of n elements takes $O(n)$ time. The paper gives, for the 1D case, a $O((n+K)\log K)$ time algorithm which produces K-maximum subarrays in sorted order. This solution is more efficient than previous proposals, for the case that $1 \leq t; K \leq t; n(n+1)/2$. This algorithm is extended to the 2D case where various techniques (e.g. sampling, divide-and-conquer, etc.) are discussed and it is shown that the worst-time case is cubic or subcubic if the value of K is relatively small.