

---

# Capsule Reviews

---

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. This issue's Capsule Reviews were compiled by Fairouz Kamarreddine. Professor Kamarreddine is an Associate Editor of *The Computer Journal* and is based in the Department of Computing and Electrical Engineering at Heriot-Watt University, Edinburgh, UK.

---

## **Reliable Omega Interconnected Network for Large-Scale Multiprocessor Systems.** S. BATAINEH AND G. E. QANZU'A

Each network in the class of multistage interconnection networks (MINs) consists of  $L$  inputs and  $L$  outputs and is capable of connecting an arbitrary input terminal to an arbitrary output terminal. All such networks are equivalent (except for the number of input/output terminals) and are called blocking networks as connecting to more than one input and output terminal may block the connection paths.

The omega network belongs to the MINs class and enjoys the features of simple routines, low diameter, good support for broadcasting facilities and the fact that each of its nodes requires a constant degree, independent of the system size. For these reasons, this paper addresses the problem of reliability and fault tolerance, one of the most important problems in real-time applications, within omega. The paper studies an omega-based parallel system with enhanced reliability. The main idea is that when the system is reconfigured to isolate the failed nodes, the full rigid omega topology guarantees the same high performance and the system maintains its full strict structure despite the fault.

The skeleton of the method proposed in the paper is described as follows. First, the authors propose to add extra stages at the end of the network together with extra links between the nodes of the network. Control switches are also added to connect pairs of the original links. These control switches are set to X when the network is fault-free. If a node fails, the reconfiguration procedure determines which of these control switches must change. Each node will also have four additional switches used to bypass a node that fails. In the case of the failure of a node, a reconfiguration procedure is called to replace the faulty node and maintain the omega topology. The reconfiguration and replacement procedures given in the paper are simple and an example is given to illustrate their use.

The resulting omega-based parallel fault-tolerant network presented in this paper can tolerate any single node faults and it is shown to be more reliable than previous reconfigurable designs. The paper discusses possible extensions to accommodate more than one stage of extra nodes for better reliability.

## **Efficient statistical modeling for the compression of tree structured intermediate code.** S. RAI AND P. SHANKAR

It is usually desirable to reduce the time and space it takes

to transmit and to load information. Code compression has been used in various scenarios where reduction in transmission time and memory size is a goal. Although code compression is usually followed by a decompression, the overheads of the decompression are covered by the reduction in transmission time and memory size. Some approaches use syntactic information source models for compressing the source code. This paper is concerned with tree structured intermediate code and proposes a scheme for the compression of such intermediate code consisting of a sequence of trees specified by a regular tree grammar. The paper aims to reduce a syntax conscious compressor directly from specifications of the syntactic structure of intermediate code trees in the form of a regular tree grammar. To achieve this, the authors use an arithmetic coder in conjunction with a model that is automatically generated from the specification of the data set. This scheme applies to any tree source whose trees are constructed by a regular tree grammar. The scheme is shown to perform well under certain conditions and conducted experiments illustrate the benefits of the method. There are many advantages to the method of the paper, including (i) making the probabilistic predictions used in the process easier, (ii) more natural correlation of the semantic attributes of the object with the token representing the object, and (iii) the automatic generation of the model from the regular tree grammar which reduces the user's burden.

## **Parallel Huffman decoding with applications to JPEG files.** S. T. KLEIN AND Y. WISEMAN

The Huffman coding technique is a popular compression technique widely used by either itself or in conjunction with other well known methods like gzip and JPEG. In particular, the widely used standard image compression technique JPEG includes Huffman coding in the last phase of its baseline implementation. This paper starts from parallel static Huffman encoding, where the input text is split among the available processors where each collects statistics about the frequency of occurrence of the various characters. The authors' aim is decompression (and not compression) especially since decompression of short pieces may be done on-line and needs to be fast in order to allow a reasonable response time to a query. The paper studies a method which allows the parallel decoding of a file that has been compressed by a static Huffman code. Although parallel decompression faces the synchronization problem, Huffman codes tend to resynchronize quickly and to recover

in most cases after error decoding. The paper exploits this fact and presents a decompression algorithm for decoding a Huffman encoded file. It gives experimental results relating to the general Huffman encoded files, illustrating the benefits of the method. As the JPEG compression technique includes Huffman coding, the paper deals in detail with JPEG decoding explaining how the general idea of the parallel decoding can be adapted to fit the specific characteristics of a JPEG file.

**Increasing the efficiency of existing sorting algorithms by using randomized wrappers.** A. V. GERBESSIOTIS AND C. J. SINIOLAKIS

Software libraries provide standard generic sorting algorithm implementations that are highly optimized. This paper presents a simple and practical transformation using random sampling and ordered search and converts an implementation of any sorting algorithm  $X$  into a new sorting algorithm  $Y$  of higher efficiency while still using  $X$  for sorting. The transformation assumes the availability of two functions, one for sorting a sequence of keys and one for searching for a key in an ordered sequence of keys. The searching operation is sufficiently general to allow for different implementations. The transformation algorithm derives a randomized sorting algorithm which runs faster than the original sorting algorithm (modulo some probability factor). Theoretical results that show the feasibility of this approach are given and are strengthened by experimental results in the sense that transformations under various sort functions are implemented, allowing the study of the performance of the original and the transformed algorithms. These theoretical and experimental results confirm that the

transformation proposed in this paper does indeed improve the performance of sorting algorithm implementations even if the starting algorithm was itself highly optimized.

**Extended synchronised choice nets.** D. Y. CHAO

A Petri net is a directed bipartite graph which consists of two types of nodes: places that represent conditions and transitions that represent events. Each transition has a certain number of input and output places indicating the preconditions and postconditions of the event. Tokens indicate that a condition holds in a certain place. Patterns of tokens (or markings) represent the status of the system. A first order structure (FOS) contains two directed paths  $H_1$  and  $H_2$  with identical start and end nodes and without paths from  $H_1$  to  $H_2$  or vice versa. Unlike traditional classification by output conditions of places, synchronized choice nets (SNCs) were defined as a new class of nets where all FOSs are symmetric in the sense that both start and end nodes are of the same type (either transitions or places). Symmetry was needed as, otherwise, unboundedness or nonliveness may occur. This paper extends this work by allowing asymmetric FOSs where start and end nodes may have different types. Liveness marking conditions are also generalized from single sequential mutual exclusion models (SMEs) to multiple SMEs connected serially. A subclass of extended SNCs (ESNCs) can be converted to a general Petri net proposed by the authors and called a weighted SNC (WSNC). This paper checks if a WSNC is structural, live and bounded and, if so, derives the necessary and sufficient condition for liveness. Further enhancements of WSNC are given to apply it to flexible manufacturing systems with resource sharing. Further extensions to ESNCs are also discussed.