Volume 6

Advances in Information Systems Science

Edited by

Julius T. Tou
Center for Information Research
University of Florida
Gainesville, Florida

PLENUM PRESS · NEW YORK—LONDON

The Library of Congress cataloged the first volume of this title as follows:

Advances in information systems science. v. 1–

New York, Plenum Press, 1969–

v. illus. 24 cm.

Editor v. 1– J. T. Tou.

1. Information science—Collections. I. Tou, Tsu-lieh, ed.

Z699.A1A36 029.7 69–12544

Library of Congress [7]

Library of Congress Catalog Card Number 69-12544
ISBN 0-306-39406-5

A Division of Plenum Publishing Corporation
227 West 17th Street, New York, N.Y. 10011

Printed in the United States of America

Advances in Information Systems Science

Volume 6

Contributors

V. E. Kotov	*Computer Center, Siberian Branch* *Academy of Sciences of the USSR* *Novosibirsk, USSR*
A. S. Narin'yani	*Computer Center, Siberian Branch* *Academy of Sciences of the USSR* *Novosibirsk, USSR*
G. Rozenberg	*Institute of Mathematics* *Utrecht University* *Utrecht-DeUithof, The Netherlands* *and* *Department of Mathematics* *University of Antwerp, UIA* *Wilrjk, Belgium*
A. Salomaa	*Department of Computer Science* *University of Aarhus* *Aarhus, Denmark*
Wladyslaw M. Turski	*Institute of Mathematical Machines–MERA* *Warsaw, Poland*

A Continuation Order Plan is available for this series. A continuation order will bring delivery of each new volume immediately upon publication. Volumes are billed only upon actual shipment. For further information please contact the publisher.

Contents of Earlier Volumes

Articles Planned for Future Volumes

David J. Farber (USA)	*Design Principles of Computer Networks*
Thomas Huang (USA)	*Image Enhancement and Recognition by Digital Computer*
Masayuki Inagaki (Japan)	*Diagnosis Techniques and Methodologies in Digital Systems*
Allen Klinger (USA)	*Data Structures for Pattern Recognition*
Ted G. Lewis (USA)	*Hardware, Firmware, Software Technology in Microcomputer Systems*
S. K. Chang and C. N. Liu (USA)	*Design Considerations of Distributed Database Systems*
Peter Wegner (USA)	*Data Structures in Programming Languages*

Preface

Information systems science is rapidly advancing in many directions. Diversified ideas, methodologies, and techniques as well as applications have been conceived and developed. This series intends to discuss some of the recent advances in this field. In this volume, we cover four timely topics which are of great current interest. In each chapter, an attempt is made to familiarize the reader with some basic background information on the advances discussed. The emphasis of this volume is placed upon parallel programming, data structure models in information system design, and the principles of L systems.

One of the effective means to enhance computation performance is parallel information processing on multiprocessor systems. In Chapter 1, V. E. Kotov discusses the practical aspects of parallel programming. He is concerned with the languages and methods of parallel programming, performance analysis, and automatic synthesis of parallel programs. In Chapter 2, A. S. Narin'yani presents the formal theory of parallel computations. These two chapters attempt to correlate and classify various methods in parallel programming, thus providing the reader with a unified approach to this important subject matter.

Data structures play an important role in information system design. Chapter 3 is devoted to this topic. W. M. Turski discusses the basic notion of data morphology and presents several data structure models in information system design. A simple information retrieval scheme is used to illustrate the principles. In Chapter 4, G. Rozenberg and A. Salomaa present a comprehensive survey of the mathematical theory of L systems. This theory was originally developed for providing mathematical models to describe the behavior of simple filamentous organisms. Now, the theory of L systems has been extended and broadened to become a branch of formal language theory. Further development of this theory may advance the design concepts for information systems.

The editor wishes to express sincere thanks to the authors of this volume for their cooperation and for the timely completion of their manuscripts. In

fact, many more of our colleagues contributed to the book than those whose names appear in the contents. Much credit is due our reviewers of the articles who provided invaluable advice and constructive criticism.

Gainesville, Florida
March, 1976

JULIUS T. TOU

Contents

Chapter 3 Data Structure Models in Information System Design

Wladyslaw M. Turski

Chapter 4 The Mathematical Theory of L Systems

G. Rozenberg and A. Salomaa

Chapter 1

THEORY OF PARALLEL PROGRAMMING. I. SURVEY OF PRACTICAL ASPECTS†

V. E. Kotov

Computer Center, Siberian Branch
Academy of Sciences of the USSR
Novosibirsk

1. INTRODUCTION

The search for ways to enhance the performance of computer equipment eventually led, in the 1950s, to the notion of parallel data processing on multiprocessor parallel computer systems. These systems are made up of a number of computing units known as *processors*. Each processor executes, concurrently with the others, some fragment of the computation; the result produced by one processor can be made accessible to the others for further processing. The structure and organization of a multiprocessor computer system can vary: The processors may be of the same type or have different properties; each of the processors may have its own memory and they may exchange data over permanent or dynamically assigned channels, or the processors may all access one central memory through which data are exchanged; supervisory control of the concurrent operation of and exchange of information between processors may be centralized or allocated among the processors; finally, the system may have a hierarchic structure, in which each processor may be organized as a multiprocessor system.

The programming of problems for multiprocessor systems falls under the heading of *parallel programming*, to distinguish it from "conventional" sequential programming for single-processor computer equipment.

The first steps in the investigation of parallel computations, including the comparative analysis of computer system structure and the choice of

† The survey of parallel programming theory is presented as Chapters 1 and 2 of this volume. It was completed in 1972, and the Russian version was published in *Kibernetika*, Nos. 1–3 and 5, 1974.

methods of parallel program synthesis for specific systems, were made in the early 1960s by Gill [46], Estrin and others [38–40], Martin [78], Evreinov and Kosarev [41–43], Conway [24], and others.† The papers published in that period were generally of an ideological character; those years saw the formulation of the main problems and lines of development for parallel programming. During the same period the first multiprocessor systems oriented toward special classes of problems were conceived and elaborated.

In the same period the theory of sequential programming evolved into an independent mathematical discipline, based on the fundamental work of A. A. Lyapunov, Yu. I. Yanov, A. P. Ershov, and J. McCarthy in the fifties. A formal apparatus was created for the investigation of the programs and computations as well as the transformation of programs, and, most important, the methodology was developed for theoretical research on programming on the basis of the results and methods of mathematical logic, algorithm theory, algebra, graph theory, and mathematical linguistics.

In the mid-sixties the theory of parallel computations began developing within the framework of programming theory. This theory is concerned with a broad spectrum of problems that arise in the programming and solution of problems on multiprocessor systems, ranging from the purely combinatorial problems associated with the allocation of tasks among processors to the construction of general mathematical models of parallel computations. It is important to note that parallel programming theory grew up under more favorable conditions than sequential programming theory, since by this time the latter had acquired a wealth of research tools and methods, which were used with great success in the theory of parallel programming. On the other hand, the growth of this theory was held back by the absence of a well-developed parallel computation practice.

Today parallel programming theory boasts a rich bibliography covering a broad sphere of problem areas. On reviewing the papers in this field, one is first struck by the enormous diversity of methods, styles, and approaches to the solution of what are more or less identical problems, as well as the lack of a common conceptual apparatus and a unified philosophy. This situation, of course, is a consequence of the considerable complexity and newness of the subject. At the same time, parallel programming theory is approaching the milestone in its development where, under the influence of the growing demands of practice, the need is mounting for the con-

† The references for both parts of the survey are given at the end of Chapter 2. By no means is the list meant to be an exhaustive bibliography on parallel programming theory; however, it does contain papers that are not mentioned in the text of the survey.

solidation and critical interpretation of the accumulated materials and points of view. There must be a concerted effort to develop adequately general methods and concepts and to define the fundamental problems and avenues for the further development of the theory.

We delineate (admittedly, more or less conditionally) the following four areas in which to group the majority of papers on parallel programming theory:

1. Parallel programming methods and languages. The objective in this group of papers is to formulate practical languages for parallel programming.
2. Scheduling aspects of parallel computations, including determination of their quantitative characteristics (execution waiting times, estimation of the required number of processors, etc.) and the *a priori* and dynamic allocation of program fragments among system processors.
3. The automatic synthesis of parallel programs, including the discovery of internal parallelism inherent in sequential programs and the creation of algorithms for the transformation of sequential programs into parallel programs.
4. General (or formal) parallel computation theory, the scope of which includes formal models of parallel programs and systems with which to investigate the general properties of parallel computations such as equivalence, determinacy, and degree of parallelism.

This chapter is the first part of a critical survey of the current state of parallel programming theory and is concerned with the practical, informal aspects of the theory, namely, the languages and methods of parallel programming, performance analysis, and the automatic synthesis of parallel programs (parallelization). Chapter 2, by A. S. Narin'yani, presents the second part of the survey, dealing with the problems and results of the formal theory of parallel computations. The decision to segregate the survey into two independent parts was motivated by the considerable subject matter and stylistic dissimilarity of papers dealing with the formal theory and papers in the other three groups, as well as the need to "parallelize" the work of the two authors in writing the survey due to the vastness of the material to be covered.

The chief aim of the survey as a whole is to inform the reader of the fundamental problems, results, and methods in parallel programming theory and to try to correlate and classify them from a unified point of view,