Volume 7

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 Catalog Card Number 69-12544
ISBN 0-306-39407-3

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 7

Contributors

S. K. Chang	*Department of Information Engineering* *University of Illinois at Chicago Circle* *Chicago, Illinois*
P. A. D. deMaine	*Computer Science Department* *Pennsylvania State University* *University Park, Pennsylvania*
Masayuki Inagaki	*System Development Department* *Computer Engineering Division* *Nippon Electric Co., Ltd.* *Fuchu City, Tokyo, Japan*
Takuo Kitamura	*System Development Department* *Computer Engineering Division* *Nippon Electric Co., Ltd.* *Fuchu City, Tokyo, Japan*
Allen Klinger	*University of California* *Los Angeles, California*
T. G. Lewis	*Computer Science Department* *Oregon State University* *Corvallis, Oregon*
C. N. Liu	*Computer Science Department* *IBM T. J. Watson Research Center* *Yorktown Heights, New York*
Shunji Tashiro	*System Development Department* *Computer Engineering Division* *Nippon Electric Co., Ltd.* *Fuchu City, Tokyo, Japan*
D. E. Whitten	*Technische Universität München* *Organisch-Chemisches Institut* *München, Germany* *Present Address, Bell Telephone Laboratory, Indian Hills* *Napierville, Illinois*

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

E. Gudes, S. Ganesh, and S. Snyder	*A Surface of File Organizations and Performance*
R. C. T. Lee	*Clustering Analysis and Its Applications*
W. M. vanCleemput	*Computer-Aided Digital Systems Design*
Julius T. Tou (USA)	*Computer-Based Intelligent Information System*

Preface

Information systems science is advancing in many directions with rapid strides. Many diversified ideas, methodologies, and techniques have been conceived and developed for improving the design of information systems and for inventing new methods for solving complex information problems. This volume, the seventh of a continuing series on information systems science, covers five timely topics which are in the mainstream of current interest in this growing field. In each chapter, an attempt is made to familiarize the reader with some basic background information on the advances discussed, so that this volume may be used independently or in conjunction with the previous volumes. The emphasis in this volume is centered upon diagnosis for digital systems, distributed information networks, microcomputer technology, and data structures for pattern recognition.

In recent years, digital systems have found widespread applications in on-line real-time processing. Such applications demand high reliability, availability, and serviceability. Reliability may be improved through the use of highly reliable parts. Improvement in integrity may be accompanied by retry operation and redundant configuration. Serviceability may be improved by making use of fault diagnosis techniques. Chapter 1 is devoted to this important subject. Fault diagnosis techniques are developed to improve serviceability and to shorten mean time for repair. Kitamura, Tashiro, and Inagaki discuss many recent methods for fault diagnosis and explain them with illustrative examples.

Sharing and exchange of information and computational resources by a large number of users has been a key concept in present-day information system design. Geographically distributed and physically dissimilar databases may be treated as though they comprise a single, logically uniform database. In Chapter 2 deMaine and Whitten discuss a design for a distributed information network. Their design emphasizes expandability, high speed, efficiency, and economy. The network uses existing computer and communication facilities wherever possible and serves for the acquisition, storage, retrieval, processing, and dissemination of information. A different viewpoint of distributed information processing is presented in Chapter 3.

Chang and Liu investigate several key problems in the design and modeling of distributed information systems. They discuss resource centralization versus resource dispersion, among other interesting topics.

Chapter 4 is devoted to the technology of microcomputer systems. Lewis gives a comprehensive review of hardware, firmware, and software technology, and suggests various applications. Data structures are well-known concepts in computer science that convey the idea of how tables and lists are stored. Now the need for data structures in pattern recognition has arisen. In Chapter 5, Klinger introduces this new concept in pattern recognition.

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 to our reviewers of the articles for their invaluable advice and constructive criticism.

Gainesville, Florida
May, 1978

Julius T. Tou

Contents

Chapter 1 Diagnosis Techniques and Methodologies for Digital Systems

Takuo Kitamura, Shunji Tashiro, and Masayuki Inagaki

Chapter 2 Design for a Distributed Information Network

P. A. D. deMaine and D. E. Whitten

Chapter 4 Hardware, Firmware, Software Technology in Microcomputer Systems

T. G. Lewis

Chapter 5 Data Structures and Pattern Recognition
Allen Klinger

Chapter 1

DIAGNOSIS TECHNIQUES AND METHODOLOGIES FOR DIGITAL SYSTEMS

Takuo Kitamura,
Shunji Tashiro, and
Masayuki Inagaki

System Development Department
Computer Engineering Division
Nippon Electric Co., Ltd.
Fuchu City, Tokyo, Japan

1. INTRODUCTION

Since the main areas of application of digital systems, such as computers, control systems, etc., have been spreading to on-line real-time processing recently, high reliability, availability, and serviceability (RAS) are required. In a big digital system, the complexity of the system increases, but higher serviceability is often required. In that case, it might be impossible to shorten the time required for fault repair by the usual test program and manual diagnosis, because maintenance engineers who repair and diagnose the fault are scarce although many skilled engineers are required.

Generally, RAS can be improved by the following techniques:

1. Reliability improvement through the use of highly reliable parts.
2. Improvement in integrity (the degree to which not all the system functions are lost after a fault occurs) by retry operation and redundant configuration.
3. Serviceability improvement through the fault diagnosis technique.

We will be mainly concerned here with explaining fault diagnosis techniques. The purpose of fault diagnosis technique is to improve serviceability, which results in a shortening of the MTTR (mean time to repair). The systematization of fault diagnosis techniques has not yet been completed, but many researchers have proposed various fault diagnosis methods, which have been described in several papers. In this paper many recent methods are explained as clearly as possible by using examples.

A digital system is divided mainly into logical equipment, memory devices, and electromechanical equipment. Since electromechanical equipment and memory device diagnosis can be performed sufficiently well according to the experience of the maintenance engineer and the usual maintenance manual, these equipment diagnoses are not treated in this paper. Instead, we consider only the diagnosis of faults in logical equipment.

When a fault occurs in logical equipment, the fault diagnosis procedure is basically as follows.

A series of input data is inserted into the equipment under diagnosis and the corresponding output data are observed. The fault location is identified by utilizing the observation results after detecting a fault. The procedure in which the input data series is inserted and the corresponding output data are observed is defined as a "test." The procedure in which the fault location is identified by utilizing the observation results after detecting a fault is defined as "diagnosis." In common usage, a "test" executed during diagnosis may be included in "diagnosis" in the broad meaning. Input data prepared for testing are defined as "test data," and the corresponding output data are defined as "symptom(s)."

The relations between test data, symptom, and fault must be obtained prior to diagnostic execution. The techniques of generating the optimum test data and recognizing the relation between symptom and fault are the basis of fault diagnosis. In this paper, these techniques are explained in Sections 2 and 3. In Section 4, examples of the diagnostic execution method by applying the previously prepared test data to the failed equipment are explained. In Section 5, some future problems in fault diagnosis are described.

2. DIAGNOSTIC DATA GENERATION THEORY

2.1. Digital System and Its Faults

In this section, assumptions and definitions of important terms are described.

Digital system elements are mutually connected with each other for treating discrete finite signals. Popular examples of digital systems are digital computers, electronic switching systems for telephone services, and process control equipment. Logical equipment in a digital system is composed of logical circuits. The analysis and synthesis of logical circuits has been developed as a major switching theory field. Characteristics of a logical circuit containing memory elements and feedback loop differ from those of a logical circuit containing no memory circuit and no feedback loop. The former is called a sequential circuit and the latter is called a combinational circuit. Since a sequential circuit contains at least one memory element, logical circuit symptoms cannot be determined by test data and the specific fault only. Symptoms are also affected by the internal state of the memory elements. For a combinational circuit, a symptom is definitely determined by test data and the fault.

For test and diagnosis, a combinational circuit can be simply treated theoretically. However, it is not too much to say that almost all actual logical equipment contains sequential circuits. Therefore, various methods are necessary for sequential circuit test and diagnosis. The size of a logical circuit, which is the object of test and diagnosis, is classified into large circuits (approximately 10,000 gates), medium circuits (approximately 1,000 gates), and small circuits (approximately 100 gates). Of the various methods described in Section 2.2, some methods have good features but cannot be used for a large circuit. In this case, logical circuits may be converted from sequential circuits to combinational circuits or divided into small circuits by additional circuits with special test and diagnostic function when test and diagnosis is executed. In this paper, fault types are restricted to logical faults. A logical fault is defined as a fault that produces some change in the logical behavior of the circuit.

Since it is very difficult to treat an intermittent fault by the present techniques, only the stuck-type fault is assumed. A stuck-type fault is defined as that which causes an input or output of some gate to be (or appear logically to be) "stuck at one" or "stuck at zero." These names are abbreviated to s-a-1 and s-a-0, respectively, in the sequel. Actual stuck-type faults are classified into stuck-at-0 (s-a-0) fault, stuck-at-1 (s-a-1) fault, short fault, ground fault, and open fault. Strictly speaking, these faults show different characteristics in the hardware system, but these faults can be treated as s-a-0 faults or s-a-1 faults in general consideration. The case of a single fault is assumed basically, in the method of generating tests, but multiple faults can also be treated by some methods.

2.2. Test Data Generation Method

There are two methods of generating test data. In one method, the logical system function is paid attention to. In the other method, the circuit function is paid attention to. When test data are generated from the view-