EATCS
Monographs on Theoretical Computer Science

Ryszard Janicki Peter E. Lauer

Specification and Analysis of Concurrent Systems

The COSY Approach

With 128 Figures

Springer-Verlag
Berlin Heidelberg New York
London Paris Tokyo
Hong Kong Barcelona
Budapest

Authors

Prof. Dr. Ryszard Janicki
Prof. Dr. Peter E. Lauer

Department of Computer Science and Systems
McMaster University
1280 Main Street West
Hamilton, Ontario L8S 4K1, Canada

Editors

Prof. Dr. Wilfried Brauer
Institut für Informatik, Technische Universität München
Arcisstrasse 21, W-8000 München 2

Prof. Dr. Grzegorz Rozenberg
Institute of Applied Mathematics and Computer Science
University of Leiden, Niels-Bohr-Weg 1, P. O. Box 9512
2300 RA Leiden, The Netherlands

Prof. Dr. Arto Salomaa
The Academy of Finland
Department of Mathematics, University of Turku
SF-20 500 Turku, Finland

This book is the result of the combined efforts of the authors. Their names have been listed in alphabetical order with no implication that one is senior and the other junior.

ISBN-13: 978-3-642-77339-6 e-ISBN-13: 978-3-642-77337-2
DOI: 10.1007/978-3-642-77337-2

Library of Congress Cataloging-in-Publication Data
Janicki, Ryszard. Specification and analysis of concurrent systems: the COSY approach / Ryszard Janicki, Peter E. Lauer. p. cm. - (EATCS monographs on theoretical computer science; v. 26) Includes bibliographical references and index. (U.S.) 1. Parallel processing (Electronic computers) I. Lauer, Peter E., 1934–. II. Title. III. Series. QA76.58.J36 1992 005.1'2–dc20 92-7430

Softcover reprint of the hardcover 1st edition 1992

Typesetting: Camera ready by authors
45/3140 - 5 4 3 2 1 0 - Printed on acid-free paper

Preface

Complex systems of processes which can progress in parallel (or concurrently) and which involve critical periods of activity during which they may interfere with each other in some detrimental manner are becoming increasingly common in practice. Examples of such systems are air, train, or car traffic control, process control (such as a chemical or nuclear plant), communication systems and operating systems, etc. The complexity of the systems and the limitations of human capabilities for making complex decisions have led to attempts to reduce the complexity of the decisions required from the human operator by delegating large amounts of detailed sifting of intermediate information required for making decisions to automatic computers.

Most of the types of system listed above usually require a very high degree of validation before implementation because of the extremely critical nature of the processes being controlled, for example, the various traffic control systems whose malfunctioning could entail not only loss of property but loss of human life as well. The complexity of such validation can be reduced by initially validating the soundness of the intended control strategy independently from implementation details, and only subsequently validating that the implementation of the strategy by a specific computer system preserves the soundness of the strategy.

Concurrent systems are more difficult to specify and analyse than sequential ones, because they require the conceptualization not only of their sequential subsystems, but also of the complex interactions between them. It follows that the programmer's intuition is not enough, being unreliable in cases of high complexity. Here solution of the problem of verification of correct behaviour of the design becomes crucial, and a satisfactory conceptual apparatus for rigorous verification becomes essential.

In such an apparatus, reduction of complexity, by abstracting away from all irrelevant detail specific to some implementation of a concurrent and distributed system strategy, is desirable. But it must be easy to obtain implementations from the abstract specification of a strategy, given enough information about

the synchronization mechanisms of the concrete system on which it is to be implemented. We therefore need a notion of "system" sufficiently abstract to allow analysis only of those aspects of systems arising from their concurrency and yet capable of being readily translated into practical terms.

The COSY (COncurrent SYstem) notation and theory presented in this book constitutes such a conceptual apparatus, and was developed by the authors and others in the last decade as one of a number of mathematical approaches for conceptualizing concurrent and reactive systems.

It is based on the recognition that formal language and automata theory have been very successful in the specification and analysis of sequential and centralized systems, including both software and hardware. Hence, the COSY approach extends the conventional notions of grammar and automaton to collections of "synchronized" grammars and automata, permitting system specification and analysis of "true" concurrency without reduction to non-determinism.

In the book the COSY theory is developed to a great level of detail and this constitutes the first uniform and self-contained presentation of all results about COSY published since 1974, as well as including many new results not published elsewhere.

The COSY theory is used to analyze a sufficient number of typical problems involving concurrency, synchronization and scheduling, to allow the reader to apply the techniques presented to similar problems of interest to him or her. Furthermore, a high-level notation supporting modular development and verification of specifications is presented. A number of computer based analysis systems are also presented briefly and their possible use indicated.

The COSY model is also related to a number of major alternative models of concurrency, particularly Petri Nets, which have since the beginning served as one standard truly concurrent semantics of COSY. Indeed, much of the central part of the chapter on COSY theory is concerned with a detailed study of the relationship between COSY and Petri Nets. Other major alternative models related to COSY are the Communicating Sequential Processes (CSP) of C.A.R. Hoare, and the Calculus of Communicating Systems (CCS) of R. Milner, both of which were developed after COSY had already been introduced.

Finally, although COSY incorporates Path notation, which is due to R. Campbell and A.N. Habermann, it should be understood that COSY Paths are a generalization of the original Paths first introduced by R. Campbell and P.E. Lauer in 1974. Most references to our work in the literature have missed this important point and hence published remarks about COSY paths tend to be wrong since they concern only the original Path notation.

Concerning the structure of the book, it consists of six chapters and eight appendices. The appendices present basic mathematical concepts used throughout the book as well as long, complex and tedious proofs. Chapter 1 provides

detailed motivation, intentions of the book and acknowledgements. Chapter 2 is devoted to the formal theory of (basic) COSY. It contains the majority of formal definitions and results. Its major contents are: two kinds of semantics for COSY, one direct in terms of vector firing sequences; the other indirect in terms of Petri Nets; analysis of adequacy properties, execution strategies, and priorities. Chapter 3 deals with high-level COSY specifications including Process Notation, the Macro Notation and expansion theory. Chapter 4 analyzes 6 different applications of COSY theory. Chapter 5 compares COSY with other approaches including CCS and CSP. The last chapter gives a historical perspective and information about what has been excluded from the book.

R. Janicki
P. E. Lauer

Hamilton, Ontario, Canada, January 1992

Contents