Texts in Theoretical Computer Science An EATCS Series

Editors: W. Brauer G. Rozenberg A. Salomaa On behalf of the European Association for Theoretical Computer Science (EATCS)

Advisory Board: G. Ausiello M. Broy C. Calude S. Even J. Hartmanis N. Jones T. Leighton M. Nivat C. Papadimitriou D. Scott

Springer-Verlag Berlin Heidelberg GmbH

Peter Clote • Evangelos Kranakis

Boolean Functions and Computation Models

With 19 Figures



Authors	Series Editors
Prof. Dr. Peter Clote	Prof. Dr. Wilfried Brauer
Boston College	Institut für Informatik
Department of Computer Science	Technische Universität München
and Department of Biology	Arcisstrasse 21, 80333 München, Germany
Fulton Hall 410 B	Brauer@informatik.tu-muenchen.de
140 Commonwealth Avenue Chestnut Hill, MA 02467, USA clote@cs.bc.edu	Prof. Dr. Grzegorz Rozenberg Leiden Institute of Advanced Computer Science University of Leiden
Prof. Dr. Evangelos Kranakis Carleton University	Niels-Bohrweg 1, 2333 CA Leiden, The Netherlands rozenber@liacs.nl
School of Computer Science	Prof. Dr. Arto Salomaa
1125 Colonel By Drive	Turku Centre for Computer Science
Ottawa, Ontario, K1S 5B6, Canada kranakis@scs.carleton.ca	Lemminkäisenkatu 14 A, 20 520 Turku, Finland asalomaa@utu.fi

Library of Congress Cataloging-in-Publication Data

Clote, Peter.

Boolean functions and computation models/Peter Clote, Evangelos Kranakis. p. cm. – (Texts in theoretical computer science) Includes bibliographical references and index.

1. Computational complexity. 2. Algebra, Boolean. I. Kranakis, Evangelos. II. Title. III. Series.

QA267.7 .C58 2001 511.3-dc21

2201031128

ACM Computing Classification (1998): F.1.1, F.4.1, F.1.3

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilm or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable for prosecution under the German Copyright Law.

ISBN 978-3-642-08217-7 ISBN 978-3-662-04943-3 (eBook) DOI 10.1007/978-3-662-04943-3

© Springer-Verlag Berlin Heidelberg 2002 Originally published by Springer-Verlag Berlin Heidelberg New York in 2002.

Softcover reprint of the hardcover 1st edition 2002

The use of general descriptive names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

Cover Design: KünkelLopka, Heidelberg Typesetting: Camera-ready by the authors Printed on acid-free paper SPIN 10074467 45/3142SR – 5 4 3 2 1 0

Dedicated to our parents:

Mary Ann and Paul J. Clote Stamatia and Kostantinos Kranakis

Preface

The foundations of computational complexity theory go back to Alan Turing in the 1930s who was concerned with the existence of automatic procedures deciding the validity of mathematical statements. The first example of such a problem was the undecidability of the Halting Problem which is essentially the question of debugging a computer program: Will a given program eventually halt? Computational complexity today addresses the quantitative aspects of the solutions obtained: Is the problem to be solved tractable? But how does one measure the intractability of computation? Several ideas were proposed: A. Cobham [Cob65] raised the question of what is the right model in order to measure a "computation step", M. Rabin [Rab60] proposed the introduction of axioms that a complexity measure should satisfy, and C. Shannon [Sha49] suggested the boolean circuit that computes a boolean function.

However, an important question remains: What is the nature of computation? In 1957, John von Neumann [vN58] wrote in his notes for the Silliman Lectures concerning the nature of computation and the human brain that

... logics and statistics should be primarily, although not exclusively, viewed as the basic tools of 'information theory'. Also, that body of experience which has grown up around the planning, evaluating, and coding of complicated logical and mathematical automata will be the focus of much of this information theory. The most typical, but not the only, such automata are, of course, the large electronic computing machines.

Let me note, in passing, that it would be very satisfactory if one could talk about a 'theory' of such automata. Regrettably, what at this moment exists — and to what I must appeal — can as yet be described only as an imperfectly articulated and hardly formalized 'body of experience'.

With almost a half century after von Neumann's death, the theory of computation and automata is now a well-developed and sophisticated branch of mathematics and computer science. As he forecasted, the principal tools have proven to come from the fields of mathematical logic, combinatorics, and probability theory.

Using these tools, we have attempted to give a survey of the present state of research in the study of boolean functions, formulas, circuits, and propositional proof systems. All of these subjects are related to the overriding concern of how computation can be modeled, and what limitations and interrelations there are between different computation models.

This text is structured as follows. We begin with methods for the construction of boolean circuits which compute certain arithmetic and combinatorial functions, and investigate upper and lower bounds for circuit families. The techniques used are from combinatorics, probability and finite group theory. We then survey steps taken in a program initiated by S.A. Cook of investigating non-deterministic polynomial time, from a proof-theoretic viewpoint. Specifically, lower bounds are presented for lengths of proofs for families of propositional tautologies, when proven in certain proof systems. Techniques here involve both logic and finite combinatorics and are related to constant depth boolean circuits and to monotone arithmetic circuits.

Outline of the book

A more detailed breakdown of the book is as follows. In Chapter 1, circuits are constructed for data processing (string searching, parsing) and arithmetic (multiplication, division, fast Fourier transform). This material is intended to provide the reader with concrete examples, before initiating a more abstract study of circuit depth and size.

Chapter 2 presents a sampling of techniques to prove size lower bounds for certain restricted classes of circuits – constant depth or monotonic. These include Razborov's elegant constructive proof of the Håstad switching lemma, the Haken–Cook monotonic real circuit lower bound for the *broken moskito screen* problem, Razborov's algebraic approximation method for *majority*, and Smolensky's subsequent generalization to finite fields.

Chapter 3 studies symmetric boolean functions and related invariance groups. A characterization is given of those symmetric functions computable by constant depth polysize circuits. Invariance groups of boolean functions are characterized by a condition concerning orbit structure, and tight upper bounds are given for almost symmetric functions. Applications are given to anonymous networks such as rings and hypercubes. Most of these results are due to P. Clote and E. Kranakis.

Chapter 4 concerns the empirically observed *threshold* phenomenon concerning clause density $r = \frac{m}{n}$, where with high probability random formulas in *k*-CNF form having *m* clauses over *n* variables are satisfiable (unsatisfiable) if *r* is less (greater) than a threshold limit. The results of this chapter include a proof of an analytic upper bound, a result due to M. Kirousis, E. Kranakis and D. Krizanc.

Chapter 5 studies propositional proof systems, which have relevance to complexity theory, since NP = co-NP if and only if there exists a polynomially bounded propositional proof system. In obtaining exponential lower bounds for increasingly stronger proof systems, new techniques have been developed,

such as random restriction, algebraic and bottleneck counting methods – these techniques may ultimately play a role in separating complexity classes, and in any case are of interest in themselves. The proof systems include resolution, cutting planes, threshold logic, Nullstellensatz system, polynomial calculus, constant depth Frege, Frege, extended Frege, and substitution Frege systems.

In Chapter 6 we define various computation models including uniform circuit families, Turing machines and parallel random access machines, and illustrate some features of parallel computation by giving example programs. We then give characterizations of different parallel and sequential complexity classes in terms of function algebras – i.e., as the smallest class of functions containing certain initial functions and closed under certain operations. In the early 1960's, A. Cobham first defined polynomial time P and argued its robustness on the grounds of his machine independent characterization of P via function algebras.

With the development that certain programming languages now admit polymorphism and higher type functionals, using function algebras, complexity theory can now be lifted in a natural manner to higher types, a development which is the focus of Chapter 7. In that chapter, a new yet unpublished characterization of type 2 NC^1 functionals (due to the first author) is given in terms of a natural function algebra and related lambda calculus.

How to use the book

This text is to be of use to students as well as researchers interested in the emerging field of *logical complexity theory* (also called *implicit complexity theory*). The chapters of the book can be read as independent units. However one semester courses can be given as follows:

Semester Course	Chapters
Boolean Functions & Complexity	1,2,3
Proof Systems & Satisfiability	5, 4
Machine Models, Function Algebras & Higher Types	6, 7

At the end of every chapter, there are several exercises: some are simple extensions of results in the book while others constitute the core result of a research article. The various levels of difficulty are indicated with an asterisk placed before more difficult problems, and two asterisks for quite challenging and sometimes open research problems. The reader is invited to attempt them all.

Acknowledgments

Writing this book would have been impossible without the financial support of various research foundations, and without the exchange of ideas from many colleagues and friends.

Peter Clote is indebted to the NSF (National Science Foundation), CNRS (Centre National pour la Recherche Scientifique), Czech Academy of Science and Volkswagen Foundation for financial support of work on this text. In particular, thanks to J.-P. Ressayre for arranging a visit to Université Paris VII, and to D. Thérien for arranging a visit to the Barbados Complexity Theory Workshop, where some of the material from this text was presented. Evangelos Kranakis is indebted to NSERC (Natural Sciences and Engineering Research Council of Canada), and NWO (Netherlands Organization for the Advancement of Research) for partial support during the preparation of the book.

While holding the Gerhard Gentzen Chair of Theoretical Computer Science at the University of Munich, the first author (P. Clote) gave several courses using parts of the current text and would like to thank his students for the feedback. We would like to thank A. Abel, D. Achlioptas, T. Altenkirch, P. Beame, S. Bellantoni, E. Ben-Sasson, S. Buss, N. Danner, M. Hofmann, R. Impagliazzo, J. Johannsen, J. Krajíček, L. M. Kirousis, D. Krizanc, K.-H. Niggl, P. Pudlák, H. Schwichtenberg, Y. Stamatiou, T. Strahm, H. Straubing, G. Takeuti and J. Woelki for comments and suggestions, although of course the authors are solely responsible for any remaining errors. In particular, any omitted or erroneous references are purely unintentional. We are deeply grateful to Sam Buss, Jan Krajíček, Pavel Pudlák, and Gaisi Takeuti, with whom the first author collaborated over a period of years, and who have established many of the deepest results in propositional proof systems, as well as L. M. Kirousis and D. Krizanc with whom the second author has spent many enjoyable discussions.

Finally, we would like to express our deepest appreciation to Dr. Hans Wössner, Executive Editor for Computer Science of Springer-Verlag, who never lost faith in our project.

This book was type set using LATEX with additional macros developed by S.R. Buss for typesetting proof figures.

Peter Clote Evangelos Kranakis

Boston Ottawa July 2002

Contents

1.	Boo	blean Functions and Circuits	1
	1.1	Introduction	1
	1.2	Boolean Functions and Formulas	2
	1.3	Circuit Model	7
	1.4	Basic Functions and Reductions	8
	1.5	Nomenclature	11
	1.6	Parsing Regular and Context-Free Languages	12
	1.7	Circuits for Integer Arithmetic	17
		1.7.1 Circuits for Addition and Multiplication	17
		1.7.2 Division Using Newton Iteration	21
		1.7.3 Division Using Iterated Product	24
	1.8	Synthesis of Circuits	30
		1.8.1 Elementary Methods 3	30
		1.8.2 Shannon's Method	31
		1.8.3 Lupanov's Method	32
		1.8.4 Symmetric Functions	34
	1.9	Reducing the Fan-out	35
	1.10	Relating Formula Size and Depth	39
	1.11	Other Models	45
		1.11.1 Switching Networks	45
		1.11.2 VLSI Circuits	45
		1.11.3 Energy Consumption	45
		1.11.4 Boolean Cellular Automata	46
		1.11.5 Branching Programs	48
		1.11.6 Hopfield Nets	53
		1.11.7 Communication Complexity	54
		1.11.8 Anonymous Networks	54
	1.12	Historical and Bibliographical Remarks	55
	1.13	Exercises	56
2.	Circ	cuit Lower Bounds	61
	2.1	Introduction	61
	2.2	Shannon's Lower Bound	63
	2.3	Nechiporuk's Bound	65

2.4 Monotonic Real Circuits		68
	2.4.1 Broken Mosquito Screen	68
	2.4.2 Monotonic Real Circuits Are Powerful	77
	2.4.3 st-Connectivity	78
2.5	Parity and the Random Restriction Method	90
2.6	Probabilistic Methods	95
	2.6.1 Håstad's Lower Bound for Parity	96
	2.6.2 Depth-k Versus Depth- $(k-1)$	99
	2.6.3 Razborov's Simplification and Decision Trees 1	102
	2.6.4 A Hybrid Switching Lemma and st-Connectivity 1	107
	2.6.5 Hybrid Switching with the Uniform Distribution 1	110
2.7	Algebraic Methods 1	124
	2.7.1 Razborov's Lower Bound for Majority	
	over Boolean Circuits with Parity 1	124
	2.7.2 Smolensky's Lower Bound for MOD_p Versus MOD_q 1	129
2.8	Polynomial Method 1	132
	2.8.1 On the Strength of MOD_m Gates	132
	2.8.2 The MOD_m -Degree of Threshold Functions	135
2.9	Method of Filters 1	137
2.10	Eliminating Majority Gates 1	140
2.11	Circuits for Symmetric Functions 1	141
	2.11.1 Negative Results 1	143
	2.11.2 Positive Results 1	145
2.12	Probabilistic Circuits 1	146
2.13	Historical and Bibliographical Remarks 1	148
2.14	Exercises 1	150
Circ	uit Upper Bounds	55
3.1	Introduction	155
3.2	Definitions and Elementary Properties 1	156
3.3	Pólya's Enumeration Theory 1	162
3.4	Representability of Permutation Groups 1	64
3.5	Algorithm for Representing Cyclic Groups 1	68
3.6	Asymptotics for Invariance Groups 1	172
3.7	Almost Symmetric Languages 1	174
3.8	Symmetry and Complexity 1	178
3.9	Applications to Anonymous Networks 1	84
	3.9.1 Rings 1	185
	3.9.2 Hypercubes 1	185
3.10	Historical and Bibliographical Remarks 1	94
3.11	Exercises 1	94
Ran	domness and Satisfiability2	207
4.1 Introduction		
4.2	Threshold for 2-SAT 2	209
	$\begin{array}{c} 2.4 \\ 2.5 \\ 2.6 \\ 2.7 \\ 2.8 \\ 2.9 \\ 2.10 \\ 2.11 \\ 2.12 \\ 2.13 \\ 2.14 \\ \textbf{Circc} \\ 3.1 \\ 3.2 \\ 3.3 \\ 3.4 \\ 3.5 \\ 3.6 \\ 3.7 \\ 3.8 \\ 3.9 \\ 3.10 \\ 3.11 \\ \textbf{Ram} \\ 4.1 \\ 4.2 \end{array}$	2.41 Monotonic Real Circuits 2.4.1 Broken Mosquito Screen 2.4.2 Monotonic Real Circuits Are Powerful 2.4.3 st-Connectivity 2.5 Parity and the Random Restriction Method 2.6 Probabilistic Methods 2.6.1 Håstad's Lower Bound for Parity 2.6.2 Depth-k Versus Depth-(k - 1) 2.6.3 Razborov's Simplification and Decision Trees 2.6.4 A Hybrid Switching Lemma and st-Connectivity 2.6.5 Hybrid Switching uth the Uniform Distribution 2.6.6 Hybrid Switching with the Uniform Distribution 2.6.7 Razborov's Lower Bound for Majority over Boolean Circuits with Parity 1 2.7.1 Razboroy's Lower Bound for MoD _p Versus MOD _q 2.8 Polynomial Method 1 2.8.1 On the Strength of MOD _m Gates 1 2.8.2 The MOD _m -Degree of Threshold Functions 1 2.10 Eliminating Majority Gates 1 2.11 Negative Results 1 2.11.1 Negative Results 1 2.12 Probabilistic Circuits 1 <td< td=""></td<>

	4.3	Unsat	isfiability Threshold for 3-SAT	. 212
		4.3.1	A General Method and Local Maxima	.213
		4.3.2	Method of Single Flips	.214
		4.3.3	Approximating the Threshold	. 217
		4.3.4	Method of Double Flips	. 217
		4.3.5	Probability Calculations	. 218
	4.4	Satisfi	ability Threshold for 3-SAT	. 224
		4.4.1	Satisfiability Heuristics	. 224
		4.4.2	Threshold	. 226
	4.5 $(2+p)$ -SAT			. 229
		4.5.1	Unsatisfiability Threshold	. 230
		4.5.2	Transition from 2-SAT to 3-SAT	. 232
	4.6	Const	raint Programming	. 235
		4.6.1	Models of CSP	. 236
		4.6.2	A New Model for Random CSP	. 238
		4.6.3	The Method of Local Maxima	. 239
		4.6.4	Threshold for Model E	. 241
	4.7	Histor	rical and Bibliographical Remarks	. 242
	4.8	Exerc	ises	. 243
5	Dro	nositi	anal Proof Systems	947
J .	5 1	Introd	luction	241
	59	Comp	levity of Proofs	249
	53	Gentz	en Sequent Calculus LK	255
	0.0	531	Completeness	257
		5.3.2	Lower Bound for Cut-Free Gentzen	259
		5.3.2	Monotonic Sequent Calculus	267
	54	Resolu	ition	. 268
	0.1	5.4.1	Resolution and the PHP	. 271
		5.4.2	Resolution and Odd-Charged Graphs	. 279
		5.4.3	Schöning's Expander Graphs and Resolution	. 285
		5.4.4	Width-Bounded Resolution Proofs	. 291
		5.4.5	Interpolation and st-Connectivity	. 296
		5.4.6	Phase Transition and Length of Resolution Proofs	. 300
	5.5	Algeb	raic Refutation Systems	. 306
		5.5.1	Nullstellensatz	. 308
		5.5.2	Polynomial Calculus	. 316
		5.5.3	Gaussian Calculus	. 324
		5.5.4	Binomial Calculus	. 326
		5.5.5	Lower Bounds for the Polynomial Calculus	. 332
		5.5.6	Random CNF Formulas and the Polynomial Calculus.	. 337
	5.6	Cuttin	ng Planes CP	. 343
		5.6.1	Completeness of CP	. 345
		5.6.2	Cutting Planes and the PHP	. 348
		5.6.3	Polynomial Equivalence of CP ₂ and CP	. 353

		5.6.4 Normal Form for CP Proofs	355
		5.6.5 Lower Bounds for CP	359
		5.6.6 Threshold Logic PTK	366
	5.7 Frege Systems		
		5.7.1 Bounded Depth Frege Systems	372
		5.7.2 Extended Frege Systems	393
		5.7.3 Frege Systems and the PHP	398
	5.8	Open Problems	403
	5.9	Historical and Bibliographical Remarks	405
	5.10	Exercises	406
6.	Mac	chine Models and Function Algebras	413
	6.1	Introduction	413
	6.2	Machine Models	415
		6.2.1 Turing Machines	415
		6.2.2 Parallel Machine Model	424
		6.2.3 Example Parallel Algorithms	427
		$6.2.4 LogP \text{ Model} \dots \dots$	433
		6.2.5 Circuit Families	434
	6.3	Some Recursion Schemes	437
		6.3.1 An Algebra for the Logtime Hierarchy LH	438
		6.3.2 Bounded Recursion on Notation	450
		6.3.3 Bounded Recursion	458
		6.3.4 Bounded Minimization	465
		6.3.5 Miscellaneous	470
		6.3.6 Safe Recursion	478
	6.4	A Glimpse of Other Work	487
	6.5	Historical and Bibliographical Remarks	488
	6.6	Exercises	489
7.	Higl	her Types	497
	7.1	Introduction	497
	7.2	Type 2 Functionals	497
	7.3	Some Closure Properties of \mathcal{A}_0	502
	7.4	Square-Root and Multiple Recursion	511
	7.5	Parallel Machine Model	527
	7.6	$\lambda\text{-}\mathrm{Calculi}$ for Parallel Computable Higher Type Functionals	554
		7.6.1 Introduction to Higher Types	555
		7.6.2 <i>p</i> -Types	556
		7.6.3 Finite Typed Lambda Calculus	558
	7.7	Historical and Bibliographical Remarks	564
	7.8	Exercises	565
Ref	eren	ces	569
Ind	ex		591