

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.S. Calude
S. Even J. Hartmanis J. Hromkovič N. Jones
T. Leighton M. Nivat C. Papadimitriou D. Scott

Leonid Libkin

Elements of Finite Model Theory

With 24 Figures



Springer

Author

Prof. Leonid Libkin
Department of Computer Science
University of Toronto
Toronto ON M5S 3H5
Canada
libkin@cs.toronto.edu
www.cs.toronto.edu/~libkin

Series Editors

Prof. Dr. Wilfried Brauer
Institut für Informatik der TUM
Boltzmannstr. 3, 85748 Garching, Germany
Brauer@informatik.tu-muenchen.de

Prof. Dr. Grzegorz Rozenberg
Leiden Institute of Advanced Computer Science
University of Leiden
Niels Bohrweg 1, 2333 CA Leiden, The Netherlands
rozenber@liacs.nl

Prof. Dr. Arto Salomaa
Turku Centre for Computer Science
Lemminkäisenkatu 14 A, 20520 Turku, Finland
asalomaa@utu.fi

Library of Congress Control Number: 2004105855

ACM Computing Classification (1998): F.4, F.1, H.2, F.3

ISBN 978-3-642-05948-3 ISBN 978-3-662-07003-1 (eBook)
DOI 10.1007/978-3-662-07003-1

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 Berlin Heidelberg GmbH.

Violations are liable for prosecution under the German Copyright Law.

springeronline.com

© Springer-Verlag Berlin Heidelberg 2004

Originally published by Springer-Verlag Berlin Heidelberg New York in 2004

Softcover reprint of the hardcover 1st edition 2004

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 therefore free for general use.

Cover Design: KunkelLopka, Heidelberg

Typesetting: Camera-ready by the author

Printed on acid-free paper 45/3142/GF - 5 4 3 2 1 0

To Helen and Daniel
Алёне и Даниле

Preface

Finite model theory is an area of mathematical logic that grew out of computer science applications.

The main sources of motivational examples for finite model theory are found in database theory, computational complexity, and formal languages, although in recent years connections with other areas, such as formal methods and verification, and artificial intelligence, have been discovered.

The birth of finite model theory is often identified with Trakhtenbrot's result from 1950 stating that validity over finite models is not recursively enumerable; in other words, completeness fails over finite models. The technique of the proof, based on encoding Turing machine computations as finite structures, was reused by Fagin almost a quarter century later to prove his celebrated result that put the equality sign between the class NP and existential second-order logic, thereby providing a machine-independent characterization of an important complexity class. In 1982, Immerman and Vardi showed that over ordered structures, a fixed point extension of first-order logic captures the complexity class PTIME of polynomial time computable properties. Shortly thereafter, logical characterizations of other important complexity classes were obtained. This line of work is often referred to as descriptive complexity.

A different line of finite model theory research is associated with the development of relational databases. By the late 1970s, the relational database model had replaced others, and all the basic query languages for it were essentially first-order predicate calculus or its minor extensions. In 1974, Fagin showed that first-order logic cannot express the transitive closure query over finite relations. In 1979, Aho and Ullman rediscovered this result and brought it to the attention of the computer science community. Following this, Chandra and Harel proposed a fixed-point extension of first-order logic on finite relational structures as a query language capable of expressing queries such as the transitive closure. Logics over finite models have become the standard starting point for developing database query languages, and finite model theory techniques are used for proving results about their expressiveness and complexity.

Yet another line of work on logics over finite models originated with Büchi's work from the early 1960s: he showed that regular languages are precisely those definable in monadic second-order logic over strings. This line of work is the automata-theoretic counterpart of descriptive complexity: instead of logical characterizations of time/space restrictions of Turing machines, one provides such characterizations for weaker devices, such as automata. More recently, connections between database query languages and automata have been explored too, as the field of databases started moving away from relations to more complex data models.

In general, finite model theory studies the behavior of logics on finite structures. The reason this is a separate subject, and not a tiny chapter in classical model theory, is that most standard model-theoretic tools (most notably, compactness) fail over finite models. Over the past 25–30 years, many tools have been developed to study logics over finite structures, and these tools helped answer many questions about complexity theory, databases, formal languages, etc.

This book is an introduction to finite model theory, geared towards theoretical computer scientists. It grew out of my finite model theory course, taught to computer science graduate students at the University of Toronto. While teaching that course, I realized that there is no single source that covers all the main areas of finite model theory, and yet is suitable for computer science students. There are a number of excellent books on the subject. *Finite Model Theory* by Ebbinghaus and Flum was the first standard reference and heavily influenced the development of the field, but it is a book written for mathematicians, not computer scientists. There is also a nice set of notes by Väänänen, available on the web. Immerman's *Descriptive Complexity* deals extensively with complexity-theoretic aspects of finite model theory, but does not address other applications. *Foundations of Databases* by Abiteboul, Hull, and Vianu covers many database applications, and Thomas's chapter "Languages, automata, and logic" in the *Handbook of Formal Languages* describes connections between logic and formal languages. Given the absence of a single source for all the subjects, I decided to write course notes, which eventually became this book.

The reader is assumed to have only the most basic computer science and logic background: some discrete mathematics, theory of computation, complexity, propositional and predicate logic. The book also includes a background chapter, covering logic, computability theory, and computational complexity. In general, the book should be accessible to senior undergraduate students in computer science.

A note on exercises: there are three kinds of these. Some are the usual exercises that the reader should be able to do easily after reading each chapter. If I indicate that an exercise comes from a paper, it means that its level could range from moderately to extremely difficult: depending on the exact level, such an "exercise" could be a question on a take-home exam, or even a course

project, whose main goal is to understand the paper where the result is proven. Such exercises also gave me the opportunity to mention a number of interesting results that otherwise could not have been included in the book. There are also exercises marked with an asterisk: for these, I do not know solutions.

It gives me the great pleasure to thank my colleagues and students for their help. I received many comments from Marcelo Arenas, Pablo Barceló, Michael Benedikt, Ari Brodsky, Anuj Dawar, Ron Fagin, Arthur Fischer, Lauri Hella, Christoph Koch, Janos Makowsky, Frank Neven, Juha Nurmonen, Ben Rossman, Luc Segoufin, Thomas Schwentick, Jan Van den Bussche, Victor Vianu, and Igor Walukiewicz. Ron Fagin, as well as Yuri Gurevich, Alexander Livchak, Michael Taitlin, and Vladimir Sazonov, were also very helpful with historical comments. I taught two courses based on this book, and students in both classes provided very useful feedback; in addition to those I already thanked, I would like to acknowledge Antonina Kolokolova, Shiva Nejati, Ken Pu, Joseph Rideout, Mehrdad Sabetzadeh, Ramona Truta, and Zheng Zhang. Despite their great effort, mistakes undoubtedly remain in the book; if you find one, please let me know. My email is `libkin@cs.toronto.edu`.

Many people in the finite model theory community influenced my view of the field; it is impossible to thank them all, but I want to mention Scott Weinstein, from whom I learned finite model theory, and immediately became fascinated with the subject.

Finally, I thank Ingeborg Mayer, Alfred Hofmann, and Frank Holzwarth at Springer-Verlag for editorial assistance, and Denis Thérien for providing ideal conditions for the final proofreading of the book.

This book is dedicated to my wife, Helen, and my son, Daniel. Daniel was born one week after I finished teaching a finite model theory course in Toronto, and after several sleepless nights I decided that perhaps writing a book is the type of activity that goes well with the lack of sleep. By the time I was writing Chap. 6, Daniel had started sleeping through the night, but at that point it was too late to turn back. And without Helen's help and support I certainly would not have finished this book in only two years.

Toronto, Ontario, Canada
May 2004

Leonid Libkin

Contents

1	Introduction	1
1.1	A Database Example	1
1.2	An Example from Complexity Theory	4
1.3	An Example from Formal Language Theory	6
1.4	An Overview of the Book	8
1.5	Exercises	10
2	Preliminaries	13
2.1	Background from Mathematical Logic	13
2.2	Background from Automata and Computability Theory	17
2.3	Background from Complexity Theory	19
2.4	Bibliographic Notes	21
3	Ehrenfeucht-Fraïssé Games	23
3.1	First Inexpressibility Proofs	23
3.2	Definition and Examples of Ehrenfeucht-Fraïssé Games	26
3.3	Games and the Expressive Power of FO	32
3.4	Rank- k Types	33
3.5	Proof of the Ehrenfeucht-Fraïssé Theorem	35
3.6	More Inexpressibility Results	37
3.7	Bibliographic Notes	40
3.8	Exercises	41
4	Locality and Winning Games	45
4.1	Neighborhoods, Hanf-locality, and Gaifman-locality	45
4.2	Combinatorics of Neighborhoods	49
4.3	Locality of FO	51
4.4	Structures of Small Degree	54
4.5	Locality of FO Revisited	57
4.6	Bibliographic Notes	62
4.7	Exercises	63

5	Ordered Structures	67
5.1	Invariant Queries	67
5.2	The Power of Order-invariant FO	69
5.3	Locality of Order-invariant FO	73
5.4	Bibliographic Notes	83
5.5	Exercises	83
6	Complexity of First-Order Logic	87
6.1	Data, Expression, and Combined Complexity	87
6.2	Circuits and FO Queries	89
6.3	Expressive Power with Arbitrary Predicates	93
6.4	Uniformity and AC^0	95
6.5	Combined Complexity of FO	99
6.6	Parametric Complexity and Locality	99
6.7	Conjunctive Queries	102
6.8	Bibliographic Notes	108
6.9	Exercises	109
7	Monadic Second-Order Logic and Automata	113
7.1	Second-Order Logic and Its Fragments	113
7.2	MSO Games and Types	116
7.3	Existential and Universal MSO on Graphs	119
7.4	MSO on Strings and Regular Languages	124
7.5	FO on Strings and Star-Free Languages	127
7.6	Tree Automata	129
7.7	Complexity of MSO	133
7.8	Bibliographic Notes	136
7.9	Exercises	137
8	Logics with Counting	141
8.1	Counting and Unary Quantifiers	141
8.2	An Infinitary Counting Logic	145
8.3	Games for $\mathcal{L}_{\infty\omega}^*(\mathbf{Cnt})$	151
8.4	Counting and Locality	153
8.5	Complexity of Counting Quantifiers	155
8.6	Aggregate Operators	158
8.7	Bibliographic Notes	161
8.8	Exercises	161
9	Turing Machines and Finite Models	165
9.1	Trakhtenbrot's Theorem and Failure of Completeness	165
9.2	Fagin's Theorem and NP	168
9.3	Bibliographic Notes	174
9.4	Exercises	174

10	Fixed Point Logics and Complexity Classes	177
10.1	Fixed Points of Operators on Sets	178
10.2	Fixed Point Logics	180
10.3	Properties of LFP and IFP	184
10.4	LFP, PFP, and Polynomial Time and Space	192
10.5	DATALOG and LFP	195
10.6	Transitive Closure Logic	199
10.7	A Logic for PTIME?	204
10.8	Bibliographic Notes	206
10.9	Exercises	207
11	Finite Variable Logics	211
11.1	Logics with Finitely Many Variables	211
11.2	Pebble Games	215
11.3	Definability of Types	220
11.4	Ordering of Types	225
11.5	Canonical Structures and the Abiteboul-Vianu Theorem	229
11.6	Bibliographic Notes	232
11.7	Exercises	233
12	Zero-One Laws	235
12.1	Asymptotic Probabilities and Zero-One Laws	235
12.2	Extension Axioms	238
12.3	The Random Graph	241
12.4	Zero-One Law and Second-Order Logic	243
12.5	Almost Everywhere Equivalence of Logics	245
12.6	Bibliographic Notes	246
12.7	Exercises	247
13	Embedded Finite Models	249
13.1	Embedded Finite Models: the Setting	249
13.2	Analyzing Embedded Finite Models	252
13.3	Active-Generic Collapse	256
13.4	Restricted Quantifier Collapse	260
13.5	The Random Graph and Collapse to MSO	265
13.6	An Application: Constraint Databases	267
13.7	Bibliographic Notes	270
13.8	Exercises	271
14	Other Applications of Finite Model Theory	275
14.1	Finite Model Property and Decision Problems	275
14.2	Temporal and Modal Logics	278
14.3	Constraint Satisfaction and Homomorphisms of Finite Models	285
14.4	Bibliographic Notes	288

XIV Contents

References	291
List of Notation	305
Index	307
Name Index	313