

Recent Progress in the Boolean Domain

Recent Progress in the Boolean Domain

Edited by

Bernd Steinbach

CAMBRIDGE
SCHOLARS

P U B L I S H I N G

Recent Progress in the Boolean Domain,
Edited by Bernd Steinbach

This book first published 2014

Cambridge Scholars Publishing

12 Back Chapman Street, Newcastle upon Tyne, NE6 2XX, UK

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

Copyright © 2014 by Bernd Steinbach and contributors

All rights for this book reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of the copyright owner.

ISBN (10): 1-4438-5638-X, ISBN (13): 978-1-4438-5638-6

Contents

LIST OF FIGURES	vii
LIST OF TABLES	viii
PREFACE	ix
FOREWORD	xv
INTRODUCTION	xix
I Exceptionally Complex Boolean Problems	1
1 BOOLEAN RECTANGLE PROBLEM	3
1.1 The Problem to Solve and Its Properties	3
1.1.1 Motivation and Selection of the Problem	3
1.1.2 The Problem in Context of Graph Theory	5
1.1.3 Rectangle-Free Grids	9
1.1.4 Estimation of the Complexity	12
1.2 Search Space Restriction	14
1.2.1 Basic Approach: Complete Evaluation	14
1.2.2 Utilization of Rule Conflicts	20
1.2.3 Evaluation of Ordered Subspaces	24
1.2.4 Restricted Evaluation of Ordered Subspaces	26
1.2.5 Analysis of the Suggested Approaches	28
1.3 The Slot Principle	31
1.3.1 Utilization of Row and Column Permutations	31
1.3.2 The Head of Maximal Grids	34
1.3.3 The Body of Maximal Grids	38
1.3.4 Experimental Results	47

1.4	Restricted Enumeration	51
1.4.1	The Overflow Principle	51
1.4.2	Strategies for Improvements	57
1.4.3	Applying Heuristics	59
1.4.4	Experimental Results	60
1.5	Permutation Classes	63
1.5.1	Potential of Improvements and Obstacles	63
1.5.2	Sequential Evaluation of Permutation Classes	65
1.5.3	Iterative Greedy Approach	66
1.5.4	Unique Representative of a Permutation Class	69
1.5.5	Direct Mapping to Representatives	73
1.5.6	Soft-Computing Results	82
2	FOUR-COLORED RECTANGLE-FREE GRIDS	87
2.1	The Problem to Solve and Its Complexity	87
2.1.1	Extension of the Application Domain	87
2.1.2	The Multiple-Valued Problem	88
2.1.3	Multiple-Valued Model	93
2.1.4	Boolean Model	94
2.1.5	Estimation of the Complexity	95
2.2	Basic Approaches and Results	98
2.2.1	Solving Boolean Equations	98
2.2.2	Utilization of Permutations	99
2.2.3	Exchange of Space and Time	101
2.3	Power and Limits of SAT-Solvers	105
2.3.1	Direct Solutions for Four-Colored Grids	105
2.3.2	Restriction to a Single Color	106
2.4	Cyclic Color Assignments of Four-Colored Grids	110
2.4.1	Sequential Assignment of a Single Color	110
2.4.2	Reusable Assignment for Four Colors	112
2.5	Four-Colored Rectangle-Free Grids of the Size 12×21	121
2.5.1	Basic Consideration	121
2.5.2	Grid Heads of all Four Colors	127
2.5.3	Model Extension for a SAT-Solver	137
2.5.4	Classes of Rectangle-Free Grids $G_{12,21}$	139
3	THEORETICAL AND PRACTICAL CONCEPTS	145
3.1	Perceptions in Learning Boolean Concepts	145
3.1.1	Boolean Concept Learning	145
3.1.2	Complexity of Boolean Concepts	148

3.1.3	Studying the Human Concept Learning	150
3.1.4	New Methodology for Human Learning	152
3.1.5	Research Methodology	157
3.2	Generalized Complexity of \mathcal{ALC} Subsumption	158
3.2.1	Preliminaries	159
3.2.2	Interreducibilities	163
3.2.3	Main Results	165
3.2.4	Discussion of the Very Difficult Problem	169
3.3	Using a Reconfigurable Computer	170
3.3.1	Why Do We Need Algebraic Immunity?	170
3.3.2	Why Do We Need a Reconfigurable Computer?	173
3.3.3	Background and Notation	174
3.3.4	Computation of Algebraic Immunity	177
3.3.5	Results and Comments	181
II Digital Circuits		187
4	DESIGN	189
4.1	Low-Power CMOS Design	189
4.1.1	Power Dissipation	189
4.1.2	Power Consumption Models	191
4.1.3	Power Optimization	199
4.1.4	Low-Power Design Application	206
4.1.5	How Low Can Power Go?	211
4.2	Permuting Variables to Improve Iterative Re-Synthesis	213
4.2.1	Iterative Logic Synthesis	213
4.2.2	Randomness in Logic Synthesis	214
4.2.3	The Influence of the Source File Structure	215
4.2.4	The Proposed Method	220
4.2.5	Experimental Results	223
4.2.6	Convergence Analysis	228
4.2.7	Advantages of Re-Synthesis with Permutations	228
4.3	Beads and Shapes of Decision Diagrams	231
4.3.1	Three Concepts	231
4.3.2	Beads, Functions, and Decision Diagrams	232
4.3.3	Beads and Decision Diagrams	235
4.3.4	Word-Level Decision Diagrams	240
4.3.5	Beads and Classification in Terms of WDDs	242
4.3.6	Approaches for Classification	246

4.4	Polynomial Expansion of Symmetric Functions	247
4.4.1	Polynomials of Boolean Functions	247
4.4.2	Main Definitions	248
4.4.3	Transeunt Triangle Method	250
4.4.4	Matrix Method to Generate $\gamma(F)$ and $\mu(F)$	255
4.4.5	Efficiency of the Matrix Method	262
4.5	Weighted Don't Cares	263
4.5.1	Don't Care Conditions in Logic Synthesis	263
4.5.2	Weighted Don't Cares	264
4.5.3	Application	266
4.5.4	Weighted BOOM: a Synthesis Tool	269
4.5.5	Experimental Results	273
4.5.6	Solutions Count Analysis	275
4.5.7	Future Applications	277
4.6	Assignments of Incompletely Specified Functions	278
4.6.1	Incompletely Specified Boolean Functions	278
4.6.2	Decision Diagrams for ISFs	279
4.6.3	Assignment of Unspecified Values	282
4.6.4	Implementation and Experimental Results	285
4.7	On State Machine Decomposition of Petri Nets	288
4.7.1	Petri Nets as Model of Concurrent Controllers	288
4.7.2	Petri Nets: Main Definitions	289
4.7.3	Conditions of SM-Coverability	291
4.7.4	Calculation of SM-Decompositions	297
4.7.5	Evaluation of the Results	300
5	TEST	303
5.1	Fault Diagnosis with Structurally Synthesized BDDs	303
5.1.1	From Functional BDDs to Structural BDDs	303
5.1.2	Structurally Synthesized BDDs	306
5.1.3	Fault Diagnosis in the Case of Multiple Faults	312
5.1.4	Fault Masking in Digital Circuits	317
5.1.5	Topological View on Fault Masking	321
5.1.6	Test Groups and Hierarchical Fault Diagnosis	327
5.1.7	Experimental Data	329
5.1.8	General Comments About Proposed Methods	331
5.2	Blind Testing of Polynomials by Linear Checks	332
5.2.1	Functional Testing by Linear Checks	332
5.2.2	Walsh Spectrum of Polynomials	335
5.2.3	Spectral Testing of a Polynomial	337

5.2.4 Universal Linear Checks 340
 5.2.5 Complexity of Universal Linear Checks 343

III Towards Future Technologies 347

6 REVERSIBLE AND QUANTUM CIRCUITS 349
 6.1 The Computational Power of the Square Root of NOT . 349
 6.1.1 Reversible Computing \Leftrightarrow Quantum Computing 349
 6.1.2 One-Qubit Circuits 350
 6.1.3 Two-Qubits Circuits 350
 6.1.4 Many-Qubits Circuits 357
 6.1.5 Increased Computational Power 358
 6.2 Toffoli Gates with Multiple Mixed Control Signals . . 359
 6.2.1 On the Evolution of Toffoli Gates 359
 6.2.2 Toffoli Gates with 3 Mixed Control Signals . . 361
 6.2.3 Toffoli Gates with $n > 3$ Mixed Control Inputs 365
 6.3 Reducing the Quantum Cost of Pairs of Toffoli Gates . 369
 6.3.1 Reversible Circuits Synthesis 369
 6.3.2 Background 370
 6.3.3 NCVW Quantum Circuits 372
 6.3.4 Optimal Circuit Synthesis 374
 6.3.5 Experimental Results and Applications 375

BIBLIOGRAPHY 381

LIST OF AUTHORS 413

INDEX OF AUTHORS 419

INDEX 421

List of Figures

1.1	Two bipartite graphs	8
1.2	Grids of two bipartite graphs	10
1.3	Number of grid patterns n_{gp} and all included rectangles for quadratic grids	13
1.4	Creating slots within a grid $G_{4,5}$	33
1.5	Sequence of all maximal grid heads of $G_{4,4}$	38
1.6	Enumeration of the body rows in the grid $G_{6,8}$	42
1.7	Maximal rectangle-free grids $G_{2,2}$ to $G_{10,10}$ and $G_{8,25}$	50
1.8	Paths of patterns taken by the Overflow Algorithm	53
1.9	Absolute error $\Delta(p, m, n)$	56
1.10	Relative runtime improvement	56
1.11	Rectangle-free incrementing a_{i+1} with respect to a_i	57
1.12	Recipes for creating higher bit patterns	58
1.13	Last obtained optimum for $G_{9,8}$ and $G_{9,9}$	59
1.14	Configurations of first three rows	60
1.15	Maximal assignments of the value 1 to the grid $G_{2,2}$	68
1.16	Decimal equivalent of a grid pattern	69
1.17	Maximal representatives of $G_{2,2}$, $G_{3,3}$, and $G_{4,4}$	71
1.18	Maximal representatives of $G_{5,5}$	71
1.19	Maximal representative of $G_{6,6}$	71
1.20	Mapping of grid patterns onto a representative con- trolled by checksums	74
1.21	Mapping of $G_{6,6}$ ordered by unique intervals of rows	75
1.22	Mapping of $G_{6,6}$ ordered by unique intervals of rows and columns	76
1.23	Definition of a new row interval of $G_{6,6}$	77
1.24	Definition of a new column interval of $G_{6,6}$	78
1.25	Mapping of two grid patterns $G_{6,6}$ of the same permu- tation class onto the unique representative	79

2.1	Edge colorings of two complete bipartite graphs $G_{3,4}^1$ and $G_{3,4}^2$ using four colors	90
2.2	Selected four-colored grids $G_{2,2}$	100
2.3	Four-colored rectangle-free grid $G_{15,15}$	107
2.4	Rectangle-free grid $G_{18,18}$ where one fourth of all positions is colored with a single color	111
2.5	Cyclic quadruples in the grid $G_{4,4}$	112
2.6	Cyclic reusable single color solution of the grid $G_{18,18}$	115
2.7	Cyclic reusable coloring of grid $G_{18,18}$	116
2.8	Cyclic four-colored rectangle-free grid $G_{18,18}$	117
2.9	Cyclic quadruples in the grid $G_{5,5}$	119
2.10	Assignment of color 1 tokens to the grid $G_{12,6}$	122
2.11	Grid $G_{12,3}$ of disjoint assignments of color 1 tokens to four rows in three columns	123
2.12	Assignment of 3 color 1 tokens to the body of the grid $G_{12,19}$ using the Latin square 1	125
2.13	All four reduced Latin squares 0, . . . , 3 of the size 4×4	126
2.14	Rectangle-free grid $G_{12,21}$ that contains 63 tokens of one color	127
2.15	Rectangle-free grid $G_{12,6}$ that merges the heads of two colors	128
2.16	Steps to construct a rectangle-free grid head $G_{12,6}$ of all four colors	129
2.17	Alternative assignments to construct a rectangle-free grid $G_{12,6}$ that merges the heads of all four colors	131
2.18	Alternative assignments of color 1 tokens and consecutive tokens of the 3 other colors to the grid $G_{12,6}$	134
2.19	Rectangle-free grid $G_{12,6}$ of all four colors	137
2.20	Four-colored rectangle-free grids $G_{12,21}$	140
2.21	Number of permutation classes of grids $G_{12,21}$ for the grid head of Figure 2.19 (b)	142
2.22	Number of permutation classes of grids $G_{12,21}$ for the grid head of Figure 2.19 (c)	143
3.1	Post's lattice showing the complexity	160
4.1	NMOS transistor with terminal labels, voltages, and currents	191
4.2	CMOS inverter	193
4.3	Power consumption waveforms	199

4.4	Architectural voltage scaling	200
4.5	Logic circuit with two paths	203
4.6	Bus-invert coding	205
4.7	Switching activity dependence on architecture	210
4.8	Distribution of solutions	219
4.9	The iterative re-synthesis	223
4.10	The iterative re-synthesis with random permutations	223
4.11	Area improvements	225
4.12	Delay improvements	226
4.13	Convergence curves for the circuits alu4 and apex2	229
4.14	BDDs for functions f_1 and f_2 in Example 4.8	234
4.15	BDDs for functions f_{AND} , f_{OR} , f_{EXOR} and f_e	237
4.16	BDDs for functions f_1 , f_2 , and f_3	239
4.17	MTBDDs for functions f_1 , f_2 , f_3 , and f_4	241
4.18	Binary matrix $T_{10}(\pi(F))$ with $\pi(F) = (00011110000)$	254
4.19	Matrices D_2 and D_4	256
4.20	Binary matrix $T_{10}(\pi(F))$ with $\pi(F) = (00110000100)$	261
4.21	The Boolean function f of the running example	268
4.22	The implicant generation progress in BOOM	270
4.23	Distribution of different implicants	277
4.24	BDD* to explain the proposed method	280
4.25	Examples of compatible subdiagrams	281
4.26	Examples of conversions of subdiagrams	282
4.27	A very simple Petri net	291
4.28	A Petri net and its concurrency graph	291
4.29	A Petri net and SM-components covering it	292
4.30	A Petri net and its concurrency graph	293
4.31	A Petri net and its concurrency graph	294
4.32	A Petri net, its concurrency graph and SM-components covering the net	295
5.1	Combinational circuit	307
5.2	Structurally Synthesized BDD for a circuit	308
5.3	Topological view on testing of nodes on the SSBDD	311
5.4	Combinational circuit	314
5.5	SSBDDs for diagnostic the experiment $D(T_1, T_2)$	317
5.6	Four faults masking each other in a cycle	318
5.7	Breaking the fault masking cycle	320
5.8	Topological view: test pair – test group	324
5.9	Topological view on the fault masking mechanism	326

5.10	Hierarchical fault diagnosis	327
5.11	The architecture of a WbAH system	334
6.1	Number $g_S(n)$ of different circuits built by a cascade of building blocks	354
6.2	Number $g_S^p(n)$ of different circuits, built by a cascade of n or less building blocks	355
6.3	The Lie group $U(4)$	357
6.4	Unitary matrix, symbol, and quantum realization of the Toffoli gate	360
6.5	Toffoli gates with one negated control input	360
6.6	OR-type Toffoli gate	361
6.7	Analysis of the first element of W and W^{-1}	362
6.8	Abstract representation of a conjunction of three control variables	363
6.9	Extended Toffoli circuit with 4 mixed control units and without ancillary lines	366
6.10	NCV quantum circuit for 3×3 Peres gate	370
6.11	Graphical symbols for NCVW gates	372
6.12	Bloch sphere of quantum states and operations defined by N , V/V^+ and W/W^+ matrices	373
6.13	Identity circuit whose right and left subcircuits are inverse of each other	375
6.14	Reversible 4×4 circuits mapped to optimal NCVW quantum circuits	377
6.15	Optimal 5×5 NCV quantum circuits	378
6.16	Best NCVW and NCV quantum circuits for the pair of 4-bit and 3-bit MCT gates	379
6.17	Realization with reduced quantum cost of a Toffoli gate with 8 control signals	380

List of Tables

1.1	maxrf (m, n) calculated by complete evaluation	18
1.2	Recursive generation of all grids $G_{5,5}$	23
1.3	Iterative generation of grids $G_{5,5}$ and $G_{6,6}$	25
1.4	Restricted iterative generation of grids $G_{6,6}$ and $G_{7,7}$.	29
1.5	Maximal numbers of values 1 in grids $G_{m,n}$	30
1.6	Grid heads of all quadratic grids	39
1.7	Early break of the recursion in Algorithm 1.7	45
1.8	maxrf (m, n) utilizing the slot principle	48
1.9	maxrf (m, n) of grids $G_{m,n}$ calculated by Algorithms 1.5, 1.6, 1.7, and 1.8	49
1.10	Number of assignments modulo permutations	51
1.11	Time estimation to compute maxrf (m, n)	52
1.12	Number of solutions and runtime	60
1.13	Comparison of relative runtime for the grid $G_{10,10}$. .	61
1.14	Runtime by utilizing $z(m, n)$ for estimation	61
1.15	Maximal number of patterns of permutation classes . .	64
1.16	Iterative greedy approach and direct mapping	83
2.1	Unknown four-colorable rectangle-free grids	92
2.2	Encoding of four colors x by two Boolean variables a and b	94
2.3	Solutions of the Boolean equation (2.9)	99
2.4	Selected solutions of the Boolean equation (2.9)	100
2.5	Four-colored rectangle-free grid patterns using Algo- rithm 2.2	104
2.6	Time to find one rectangle-free pattern of quadratic four-colored grids using different SAT-solvers	106
2.7	Knowledge transfer for grids $G_{18,18}$	118
2.8	Knowledge transfer for grids $G_{17,17}$	120

2.9	Alternative assignments in four-token columns of the grid head	133
3.1	All clones and bases relevant for the classification . . .	161
3.2	Functions that annihilate the 3-variable majority function f and their degree	175
3.3	Functions that annihilate the complement of the 3-variable majority function	176
3.4	Boolean functions that annihilate the 3-variable majority function	179
3.5	Comparison of the computation times for enumerating the AI of n -variable functions	182
3.6	Comparing the brute force method with the row echelon method on 4-variable functions	183
3.7	The number of n -variable functions distributed according to algebraic immunity for $2 \leq n \leq 6$	184
3.8	Frequency and resources used to realize the AI computation on the SRC-6's Xilinx XC2VP100 FPGA	185
4.1	Taxonomy of sources of power consumption	198
4.2	The influence of permutation of variables: permuted inputs	218
4.3	The influence of permutation of variables: permuted outputs	220
4.4	The influence of permutation of variables: permuted inputs & outputs	221
4.5	The influence of permutation of variables and nodes: commercial tools	222
4.6	Summary statistics – LUTs	227
4.7	Summary statistics – levels	227
4.8	Sets of beads for functions in Example 4.9	236
4.9	Sets of beads for functions in Example 4.10	237
4.10	Sets of beads for functions in Example 4.11	238
4.11	Sets of integer beads for functions in Example 4.13	241
4.12	LP-representative functions for $n = 3$	243
4.13	Walsh spectra of LP-representative functions	243
4.14	Sets of integer beads for Walsh spectra	244
4.15	Comparison between wBOOM and BOOM	274
4.16	Numbers of solutions	276
4.17	Subfunctions for subdiagrams	281

4.18	Code converters	286
4.19	Randomly generated functions	286
4.20	Benchmark functions	287
4.21	The results of experiments	301
5.1	5-valued algebra for calculating Boolean differentials	314
5.2	Diagnostic process with 5 passed test patterns	315
5.3	Test patterns for selected faults	318
5.4	Test pairs for testing signal paths	319
5.5	Partial test group which detects all four faults	321
5.6	Full test group for testing an SSBDD path	323
5.7	Diagnostic processes for a circuit	328
5.8	Experimental data of generating test groups	330
5.9	Complexity of BCH based linear checks	345
6.1	Number of circuits built from different generator sets	353
6.2	Relationship between 3 input control values and activated/inhibited U -gates	364
6.3	Relationship between 4 input control values and activated/inhibited U -gates	367
6.4	Eight-valued logic for NCVW quantum operations	373
6.5	Database for optimal 4×4 quantum circuits	376
6.6	Database for optimal 5×5 quantum circuits	376

Preface

Boolean logic and algebra are cornerstones of computing and other digital systems, and are thus fundamental to both theory and practice in Computer Science, Engineering, and many other disciplines. Understanding and developing Boolean concepts and techniques are critical in an increasingly digital world. This book presents recent progress through a variety of contributions by thirty-one authors from the international Boolean domain research community.

The first part of this book addresses exceptionally complex Boolean problems. The reader may well ask “What is an exceptionally complex Boolean problem?” The answer is that there are many and they are diverse. Some are theoretical—some are extremely practical. While every problem has its own defining features, they also have many aspects in common, most notably the huge computational challenges they can pose.

The first challenge considered is identified as the Boolean Rectangle Problem. Like many extremely complex Boolean problems, this problem is easy to state, easy to understand, and easy to tell when you have a solution. It is finding a solution that is the challenge.

The discussion of this problem takes the reader through the description of the problem, its analysis, its formulation in the Boolean domain and from there on to several solutions. While the discussion focuses on a particular problem, the reader will gain a very good general understanding of how problems of this nature can be addressed and how they can be cast into the Boolean domain in order to be solved subsequently by sophisticated and powerful tools such as the Boolean minimizer used in this instance. The discussion of the problem continues with a very insightful comparison of exact and heuristic approaches followed by a study of the role of permutation classes in solving such problems.

Building on the above, the discussion continues to show how the techniques developed in the Boolean domain can be extended to the multiple-valued domain. The presentation again focuses on a single problem, Rectangle-free Four-colored Grids, but as before, the presentation provides broad general insights. The reader is encouraged to consider which techniques transfer easily from the Boolean to the multiple-valued domain and where novel ideas must be injected. The discussion is interesting both in terms of how to approach solving the problem at hand and similar problems, and also as an illustration of the use of modern SAT-solvers. Satisfiability (SAT) is a central concept in the theoretical analysis of computation, and it is of great interest and value to see the application of a SAT-solver as a powerful tool for solving an extremely complex Boolean problem. The reader will benefit greatly from this demonstration of another powerful solution technique.

The contribution on Perception in Learning Boolean Concepts approaches the issue of complexity from a very different point of view. Rather than considering complexity in the mathematical or computational sense, the work examines complexity, complexity of Boolean concepts in particular, through consideration of a human concept of learning and understanding. This alternate view provides a very different insight into the understanding of the Boolean domain and will aid readers in broadening their conceptual understanding of the Boolean domain.

The use of logic in computation is a broad area with many diverse approaches and viewpoints. This is demonstrated in the presentation on Generalized Complexity of \mathcal{ALC} Subsumption. The discussion considers a variety of concepts from a rather theoretical point of view but also points to the practical implications of those concepts. It also presents yet another view of an extremely complex Boolean problem in terms of algorithmic constructions for the subsumption problem.

The final presentation on exceptionally complex Boolean problems considers encryption and cryptanalysis. The discussion is of considerable interest due to the obvious practical importance of the problem. Readers, even those familiar with state-of-the-art encryption methods, will benefit from the presentation on the concept of algebraic immunity and its computation. The approach presented is also of con-

siderable interest in its use of a reconfigurable computer. The reader should consider this approach as another tool in the computational toolbox for the Boolean domain.

The second part of this book begins with a discussion of low-power CMOS design. CMOS is currently the dominant technology for digital systems and this contribution is of particular significance given the ever-growing demand for low-power devices. After an overview of CMOS design, a number of techniques for power reduction are described. This discussion ends with the key question, “how low can power go?” – an interesting query on its own and also a perfect lead into the discussion of reversibility in the final part of the book.

Designing and testing of digital devices and systems have for a long time been a major motivation for research in the Boolean domain. The combinational logic design contributions in this book treat a variety of topics: the shape of binary decision diagrams; polynomial expansion of symmetric Boolean functions; and the issue of dealing with the don't-care assignment problem for incompletely specified Boolean functions. The final logic design contribution concerns state machine decomposition of Petri nets. Individually, these contributions provide insight and techniques specific to the particular problem at hand. Collectively they show the breadth of issues still open in this area as well as the connection of theoretical and practical concepts.

Testing is the subject of the next two contributions. The first concerns Boolean fault diagnosis with structurally synthesized BDDs. This contribution shows how binary decision diagrams, which are used in quite different contexts in earlier parts of the book, can be adapted to address a significantly different problem in a unique way. The second testing contribution considers techniques in a built-in self-test. After reviewing spectral techniques for testing, the discussion centers upon testing of polynomials by linear checks. The reader will gain an appreciation of the relationship between the spectral and the Boolean domains and how fairly formal techniques in the first domain are applied to a very practical application in the second.

The final part of this book addresses topics concerning the connection between two important emerging technologies: reversible and quantum logic circuits. A reversible logic circuit is one where there is a

one-to-one correspondence between the input and output patterns, hence the function performed by the circuit is invertible. A major motivation for the study of reversible circuits is that they potentially lead to low power consumption. In addition, the study of reversible circuits has intensified because of the intrinsic connection of quantum computation to the gate model.

The transformations performed by quantum gates are defined by unitary matrices and are thus by definition reversible. Reversible Boolean functions are also central components of many quantum computation algorithms. This part of the book provides novel ideas and is also a very good basis for understanding the challenging problem of synthesizing and optimizing quantum gate realizations of reversible functions.

The part begins with a detailed study of the computational power of a gate referred to as the square root of NOT since two such gates in succession realize the conventional Boolean NOT gate. In addition to describing the computational power of such gates, this contribution provides a very good basis for understanding the connections and differences between reversible Boolean gates and quantum operations.

The Toffoli gate is a key building block in Boolean reversible circuits. The second contribution in this section considers the realization of Toffoli gates using controlled-NOT gates and the square root of NOT as well as the fourth root of NOT gates. The work extends beyond the conventional Toffoli gate to include multiple mixed positive and negative controls, and alternate control functions.

The final contribution in this part concerns the quantum realization of pairs of multi-control Toffoli gates. It builds nicely on the work in the two preceding contributions and provides the reader with valuable insight and techniques for the optimization of quantum gate realizations particularly for reversible logic.

I am confident that this book will provide novel ideas and concepts to researchers and students whether or not they are knowledgeable in modern approaches and recent progress in the Boolean domain. I am also confident that study of the work presented here will lead to further developments in Boolean problem solving and the application

of such techniques in both theory and practice.

The contributions appearing in this book are extended versions of works presented at the International Workshop on Boolean Problems held at the Technische Universität Bergakademie Freiberg, Germany on September 19-21, 2012. The 2012 workshop was the tenth in a series of Boolean Problems Workshops held in Freiberg biennially since 1994. Prof. Bernd Steinbach has organized and hosted the workshop since its inception. The Boolean research community is indebted to him for this long-term contribution and for his efforts in organizing and editing this book.

D. Michael Miller

Department of Computer Science
University of Victoria
Victoria, British Columbia, Canada

Foreword

This book covers several fields in theory and practical applications where Boolean models support these solutions. Boolean variables are the simplest variables of all, because they have the smallest possible range of only two different values. In logic applications these values express the truth values *true* and *false*; in technical applications the values of signals *high* and *low* are described by Boolean variables. For simplification, the numbers 0 and 1 are most commonly used as values of Boolean variables.

The basic knowledge in the Boolean domain goes back to the English mathematician, philosopher and logician George Boole as well the American mathematician, electronic engineer, and cryptographer Claude Shannon. The initiator of the very strong increase of Boolean applications was Conrad Zuse. He recognized the benefit of Boolean values to avoid errors in large technical systems. In this way he was able to build the first computer.

The benefit of Boolean values is not restricted to computers, but can be utilized for all kinds of control systems in a wide range of applications. The invention of the transistor as a very small electronic switch and the integration of a growing number of transistors on a single chip, together with the strong decrease of the cost per transistor was the second important factor for both the substitution of existing systems by electronic ones and the extensive exploitation of new fields of applications. This development is forced by a growing community of scientists and engineers.

As part of this community, I built as an electrician control systems for machine tools, developed programs to solve Boolean equations during my studies, contributed to test software for computers as a graduate engineer, and taught students as an assistant professor for design automation. In 1992, I got a position as a full professor at the

Technische Universität Bergakademie Freiberg. Impressed by both the strong development and the challenges in the Boolean domain, I came to the conclusion that a workshop about *Boolean Problems* could be a valuable meeting point for people from all over the world which are working in different branches of the Boolean domain. Hence, I organized the first workshop in 1994 and, encouraged by the attendees, I continued the organization of the biennial series of such *International Workshops on Boolean Problems* (IWSBP).

The idea for this book goes back to Carol Koulikourdi, Commissioning Editor of Cambridge Scholars Publishing. She asked me one month before the 10th IWSBP whether I would agree to publish this book based on the proceedings of the workshop. I discussed this idea with the attendees of the 10th IWSBP and we commonly decided to prepare this book with extended versions of the best papers of the workshop. The selection of these papers was carried out based on the reviews and the evaluation of the attendees of the 10th International Workshop on Boolean Problems. Hence, there are many people which contributed directly or indirectly to this book.

I would like to thank all of them: starting with the scientists and engineers who have been working hard on Boolean problems and submitted papers about their results to the 10th IWSBP; continuing with the 23 reviewers from eleven countries; the invited speakers Prof. Raimund Ubar from the Tallinn University of Technology, Estonia, and Prof. Vincent Gaudet from the University of Waterloo, Canada; all presenters of the papers; and all attendees for their fruitful discussions the very interesting presentation on all three days of the workshop. Besides the technical program, such an international workshop requires a lot of work to organize all the necessary things. Without the support of Ms. Dr. Galina Rudolf, Ms. Karin Schüttauf, and Ms. Birgit Steffen, I would not have been able to organize this series of workshops. Hence, I would very much like to thank these three ladies for their valuable hard work very much.

Not only the authors of the sections but often larger groups contribute to the presented results. In many cases these peoples are financially supported by grants of many different organizations. Both the authors of the sections of this book and myself thank them for this significant support. The list of these organizations, the numbers of grants, and

the titles of the supported projects is so long that I must forward the interested reader to check this information to the proceedings of the 10th IWSBP [287].

I would like to emphasize that this book is a common work of many authors. Their names are directly associated to each section and additionally summarized in lexicographical order in the section *List of Authors* starting on page 413 and the *Index of Authors* on page 419. Many thanks to all of them for their excellent collaboration and high quality contributions. My special thanks goes to Prof. Michael Miller for his *Preface* which reflects the content of the whole book in a compact and clear manner, Prof. Christian Posthoff and Alison Rigg for corrections of the English text, and Matthias Werner for setting up the L^AT_EX-project for the book and improving the quality of the book using many L^AT_EX-tools.

Finally, I like to thank Ms. Carol Koulikourdi for her idea to prepare this book, the acceptance to prepare this scientific book using L^AT_EX, and for her very kind collaboration. I hope that all readers enjoy reading the book and find helpful suggestions for their own work in the future. It will be my pleasure to talk with many readers at one of the next International Workshops on Boolean Problems or at any other place.

Bernd Steinbach

Department of Computer Science
Technische Universität Bergakademie Freiberg
Freiberg, Saxony, Germany

Introduction

Applications of Boolean variables, Boolean operations, Boolean functions, and Boolean equations are not restricted to computers, but grow in nearly all fields of our daily life. It may be that the digitally controlled alarm-clock wakes us in the morning; we listen to the sounds of digital audio broadcasts during our breakfast; we buy the ticket for the train on a ticket vending machine controlled by Boolean values; we use our smart phones that transmit all information by Boolean values for business; look at the wrist watch which counts and shows the time based on Boolean data so that we do not miss the end of our work; we get the bill for shopping from a pay machine which calculates the sum of the prices and initiates a transmission between our bank account and the bank account of the shop; and in the evening we watch a movie on TV which is also transmitted by Boolean values. Hence, without thinking about the details, our daily life is surrounded with a growing number of devices which utilize Boolean values and operations.

Gordon E. Moore published in 1965 a paper about the trend of components in integrated circuits which double approximately every 12 to 24 months. This observation is known as Moore's Law. Of course, there are many physical limits which take effect against this law, but due to the creativity of scientists and engineers this law is valid in general even now. The exponential increase of the control elements is a strong challenge for all people working in different fields influenced by Boolean values.

The International Workshop on Boolean Problems (IWSBP) is a suitable event where people from all over the world meet each other to report new results, to discuss different Boolean problems, and to exchange new ideas. This book documents selected activities and results of the recent progress in the Boolean domain. All sections are written from authors who presented their new results at the 10th IWSBP in

September 2012 in Freiberg, Germany.

The most general challenge in the Boolean Domain originates from the exponential increase of the complexity of the Boolean systems as stated in Moore's Law. Chapter 1 of this book deals with this problem using a Boolean task of an unlimited complexity. Chapter 2 applies the found methods to a finite, but unbelievable complex multiple-valued problem of more than 10^{195} color patterns. The last open problem in this field was recently solved, too. For completeness, these results are added as Section 2.5 to this book. Basic versions of all other sections of this book are published in the proceedings of the 10th IWSBP [287].

Success in solving special tasks for applications requires a well developed theoretical basis. Chapter 3 of the book contains interesting new results which can be utilized in future applications. The design of digital circuits is the main field where solutions of Boolean problems result in real devices. Seven different topics of this field are presented in Chapter 4. Not all produced devices are free of errors, due to geometrical structures of a few nanometers on the chips. Hence, it is a big challenge to test such circuits which consist of millions of transistors as switching elements in logic gates. Chapter 5 deals with these Boolean problems. Following Moore's Law, in the near future single atoms must be used as logic gates. This very strong change of the basic paradigm from classical logic gates to reversible quantum gates requires comprehensive preparatory work of scientists and engineers. The final chapter, Chapter 6 of this book shows recent results in reversible and quantum computing.

A more detailed overview of the content of this book is given in the excellent preface by Prof. Miller. Hence, it remains to wish the readers in the name of all authors pleasure while reading this book and many new insights which are helpful to solve many future tasks.

Bernd Steinbach

Department of Computer Science
Technische Universität Bergakademie Freiberg
Freiberg, Saxony, Germany