

Contents

1 Introduction	1
Sunil P. Khatri and Kanupriya Gulati	
1.1 Logic Decomposition	2
1.2 Boolean Satisfiability	3
1.3 Boolean Matching	4
1.4 Logic Optimization	4
1.5 Applications to Specialized Design Scenarios	5
References	6

Part I Logic Decomposition

2 Logic Synthesis by Signal-Driven Decomposition	9
Anna Bernasconi, Valentina Ciriani, Gabriella Trucco, and Tiziano Villa	
2.1 Introduction	9
2.2 Decomposition Methods	11
2.3 P-Circuits	17
2.3.1 Synthesis Algorithms	19
2.4 Multivariable Decomposition	21
2.5 Experimental Results	24
2.6 Conclusion	28
References	28
3 Sequential Logic Synthesis Using Symbolic Bi-decomposition	31
Victor N. Kravets and Alan Mishchenko	
3.1 Introduction and Motivation	31
3.2 Preliminary Constructs	33
3.2.1 “Less-Than-or-Equal” Relation	33
3.2.2 Parameterized Abstraction	34
3.3 Bi-decomposition of Incompletely Specified Functions	35
3.3.1 OR Decomposition	35
3.3.2 XOR Decomposition	36

3.4	Parameterized Decomposition	37
3.4.1	OR Parameterization	37
3.4.2	XOR Parameterization	38
3.5	Implementation Details of Sequential Synthesis	39
3.5.1	Extraction of Incompletely Specified Logic	39
3.5.2	Exploring Decomposition Choices	40
3.5.3	Synthesis Algorithm	41
3.6	Experimental Evaluation	42
3.7	Conclusions and Future Work	44
	References	45
4	Boolean Factoring and Decomposition of Logic Networks	47
	Robert Brayton, Alan Mishchenko, and Satrajit Chatterjee	
4.1	Introduction	47
4.2	Background	48
4.3	General Non-disjoint Decompositions	50
4.4	Rewriting K -LUT networks	53
4.4.1	Global View	53
4.4.2	Cut Computation	54
4.4.3	Cuts with a DSD Structure	56
4.4.4	Cut Weight	56
4.4.5	Decomposition and Network Update	57
4.4.6	Finding the Maximum Support-Reducing Decomposition	58
4.4.7	Additional Details	60
4.4.7.1	Using Timing Information to Filter Candidate Bound Sets	60
4.4.7.2	Restricting Bound Sets for Balanced Decompositions	60
4.4.7.3	Opportunistic MUX-Decomposition	60
4.5	Comparison with Boolean Matching	61
4.6	Experimental Results	62
4.7	Conclusions and Future Work	64
	References	65
5	Ashenhurst Decomposition Using SAT and Interpolation	67
	Hsuan-Po Lin, Jie-Hong Roland Jiang, and Ruei-Rung Lee	
5.1	Introduction	67
5.2	Previous Work	69
5.3	Preliminaries	69
5.3.1	Functional Decomposition	70
5.3.2	Functional Dependency	71
5.3.3	Propositional Satisfiability and Interpolation	71
5.3.3.1	Refutation Proof and Craig Interpolation	71

- 5.3.3.2 Circuit-to-CNF Conversion 72
- 5.4 Main Algorithms 72
 - 5.4.1 Single-Output Ashenhurst Decomposition 72
 - 5.4.1.1 Decomposition with Known Variable Partition 72
 - 5.4.1.2 Decomposition with Unknown Variable Partition 75
 - 5.4.2 Multiple-Output Ashenhurst Decomposition 79
 - 5.4.3 Beyond Ashenhurst Decomposition 80
- 5.5 Experimental Results 80
- 5.6 Chapter Summary 84
- References 84

6 Bi-decomposition Using SAT and Interpolation 87

Ruei-Rung Lee, Jie-Hong Roland Jiang, and Wei-Lun Hung

- 6.1 Introduction 87
- 6.2 Previous Work 88
- 6.3 Preliminaries 89
 - 6.3.1 Bi-Decomposition 89
 - 6.3.2 Propositional Satisfiability 90
 - 6.3.2.1 Refutation Proof and Craig Interpolation 90
 - 6.3.3 Circuit to CNF Conversion 91
- 6.4 Our Approach 91
 - 6.4.1 OR Bi-decomposition 91
 - 6.4.1.1 Decomposition of Completely Specified Functions 91
 - 6.4.1.2 Decomposition of Incompletely Specified Functions 97
 - 6.4.2 AND Bi-decomposition 97
 - 6.4.3 XOR Bi-decomposition 98
 - 6.4.3.1 Decomposition of Completely Specified Functions 98
 - 6.4.4 Implementation Issues 101
- 6.5 Experimental Results 101
- 6.6 Summary 103
- References 104

Part II Boolean Satisfiability

7 Boundary Points and Resolution 109

Eugene Goldberg and Panagiotis Manolios

- 7.1 Introduction 109
- 7.2 Basic Definitions 111
- 7.3 Properties 112

7.3.1	Basic Propositions	112
7.3.2	Elimination of Boundary Points by Adding Resolvents . . .	113
7.3.3	Boundary Points and Redundant Formulas	115
7.4	Resolution Proofs and Boundary Points	115
7.4.1	Resolution Proof as Boundary Point Elimination	116
7.4.2	SMR Metric and Proof Quality	116
7.5	Equivalence Checking Formulas	117
7.5.1	Building Equivalence Checking Formulas	118
7.5.2	Short Proofs for Equivalence Checking Formulas	119
7.6	Experimental Results	120
7.7	Some Background	122
7.8	Completeness of Resolution Restricted to Boundary Point Elimination	123
7.8.1	Cut Boundary Points	123
7.8.2	The Completeness Result	124
7.8.3	Boundary Points as Complexity Measure	125
7.9	Conclusions and Directions for Future Research	126
	References	126
8	SAT Sweeping with Local Observability Don't-Cares	129
	Qi Zhu, Nathan B. Kitchen, Andreas Kuehlmann, and Alberto Sangiovanni-Vincentelli	
8.1	Introduction	129
8.2	Previous Work	130
8.3	Preliminaries	131
8.3.1	AND-INVERTER Graphs	131
8.3.2	SAT Sweeping	132
8.4	SAT Sweeping with Observability Don't Cares	134
8.4.1	Motivating Example	134
8.4.2	Observability Don't Cares	134
8.4.3	Algorithm	137
8.4.4	Implementation	139
8.4.5	Applications	141
8.5	Results	142
8.6	Conclusions	146
	References	147
9	A Fast Approximation Algorithm for MIN-ONE SAT and Its Application on MAX-SAT Solving	149
	Lei Fang and Michael S. Hsiao	
9.1	Introduction	149
9.2	Preliminaries	151
9.3	Our Approach	153
9.3.1	RelaxSAT	153
9.3.2	Relaxation Heuristic	155

- 9.3.3 Discussion on Computation Complexity 156
- 9.4 Experimental Results 156
- 9.5 Application Discussion: A RelaxSAT-Based MAX-SAT Solver ... 161
 - 9.5.1 The New MAX-SAT Solver: RMAXSAT 163
 - 9.5.2 Evaluation of MAX-SAT Solver 165
- 9.6 Conclusions and Future Works 168
- References 169

10 Algorithms for Maximum Satisfiability Using Unsatisfiable Cores ... 171

Joao Marques-Sila and Jordi Planes

- 10.1 Introduction 171
- 10.2 Background 172
 - 10.2.1 The MaxSAT Problem 172
 - 10.2.2 Solving MaxSAT with PBO 173
 - 10.2.3 Relating MaxSAT with Unsatisfiable Cores 173
- 10.3 A New MaxSAT Algorithm 174
 - 10.3.1 Overview 175
 - 10.3.2 The Algorithm 175
 - 10.3.3 A Complete Example 176
- 10.4 Experimental Results 178
- 10.5 Related Work 180
- 10.6 Conclusions 180
- References 181

Part III Boolean Matching

11 Simulation and SAT-Based Boolean Matching for Large Boolean Networks 185

Kuo-Hua Wang, Chung-Ming Chan, and Jung-Chang Liu

- 11.1 Introduction 185
- 11.2 Background 186
 - 11.2.1 Boolean Matching 186
 - 11.2.2 Boolean Satisfiability 187
 - 11.2.3 And-Inverter Graph 187
- 11.3 Detection of Functional Property Using S&S Approach 188
- 11.4 Definitions and Notations 189
- 11.5 Simulation Approach for Distinguishing Inputs 190
 - 11.5.1 Type-1 191
 - 11.5.2 Type-2 192
 - 11.5.3 Type-3 192
- 11.6 S&S-Based Boolean Matching Algorithm 194
 - 11.6.1 Our Matching Algorithm 194
 - 11.6.2 Recursive-Matching Algorithm 194

- 11.6.3 Implementation Issues 196
 - 11.6.3.1 Control of Random Vector Generation 196
 - 11.6.3.2 Reduction of Simulation Time 196
 - 11.6.3.3 Analysis of Space Complexity and Runtime . . . 196
- 11.7 Experimental Results 197
- 11.8 Chapter Summary 200
- References 200

- 12 Logic Difference Optimization for Incremental Synthesis 203**
 - Smita Krishnaswamy, Haoxing Ren, Nilesh Modi, and Ruchir Puri
 - 12.1 Introduction and Background 203
 - 12.2 Previous Work 205
 - 12.3 DeltaSyn 206
 - 12.3.1 Phase I: Equivalence-Based Reduction 207
 - 12.3.2 Phase II: Matching-Based Reduction 209
 - 12.3.2.1 Subcircuit Enumeration 210
 - 12.3.2.2 Subcircuit Matching 213
 - 12.3.2.3 Subcircuit Covering 217
 - 12.3.3 Phase III: Functional Hashing-Based Reduction 218
 - 12.4 Empirical Validation 220
 - 12.5 Chapter Summary 224
 - References 224

- 13 Large-Scale Boolean Matching 227**
 - Hadi Katebi and Igor Markov
 - 13.1 Introduction 227
 - 13.2 Background and Previous Work 229
 - 13.2.1 Definitions and Notation 230
 - 13.2.2 And-Inverter Graphs (AIGs) 230
 - 13.2.3 Boolean Satisfiability and Equivalence Checking 231
 - 13.2.4 Previous Work 231
 - 13.3 Signature-Based Matching Techniques 232
 - 13.3.1 Computing I/O Support Variables 232
 - 13.3.2 Initial refinement of I/O clusters 233
 - 13.3.3 Refining Outputs by Minterm Count 234
 - 13.3.4 Refining I/O by Unateness 234
 - 13.3.5 Scalable I/O Refinement by Dependency Analysis 235
 - 13.3.6 Scalable I/O Refinement by Random Simulation 235
 - 13.3.6.1 Simulation Type 1 236
 - 13.3.6.2 Simulation Type 2 236
 - 13.3.6.3 Simulation Type 3 237
 - 13.4 SAT-Based Search 237
 - 13.4.1 SAT-Based Input Matching 238

- 13.4.2 Pruning Invalid Input Matches by SAT Counterexamples 239
- 13.4.3 SAT-Based Output Matching 240
- 13.4.4 Pruning Invalid Output Matches by SAT Counterexamples 241
- 13.4.5 Pruning Invalid I/O Matches Using Support Signatures . . . 241
- 13.4.6 Pruning Invalid Input Matches Using Symmetries 241
- 13.4.7 A Heuristic for Matching Candidates 242
- 13.5 Empirical Validation 242
- 13.6 Chapter Summary 246
- References 246

Part IV Logic Optimization

14 Algebraic Techniques to Enhance Common Sub-expression Extraction for Polynomial System Synthesis 251

Sivaram Gopalakrishnan and Priyank Kalla

- 14.1 Introduction 251
 - 14.1.1 Motivation 252
 - 14.1.2 Contributions 253
 - 14.1.3 Paper Organization 253
- 14.2 Previous Work 254
 - 14.2.1 Kernel/Co-kernel Extraction 254
- 14.3 Preliminary Concepts 255
 - 14.3.1 Polynomial Functions and Their Canonical Representations 255
 - 14.3.2 Factorization 257
- 14.4 Optimization Methods 257
 - 14.4.1 Common Coef cient Extraction 258
 - 14.4.2 Common Cube Extraction 259
 - 14.4.3 Algebraic Division 260
- 14.5 Integrated Approach 261
- 14.6 Experiments 264
- 14.7 Conclusions 265
- References 265

15 Automated Logic Restructuring with *a*SPFDs 267

Yu-Shen Yang, Subarna Sinha, Andreas Veneris, Robert Brayton, and Duncan Smith

- 15.1 Introduction 267
- 15.2 Background 269
 - 15.2.1 Prior Work on Logic Restructuring 269
 - 15.2.2 Sets of Pairs of Functions to Be Distinguished 269
- 15.3 Approximating SPFDs 270
 - 15.3.1 Computing *a*SPFDs for Combinational Circuits 271

15.3.2	Computing <i>a</i> SPFDs for Sequential Circuits	273
15.3.3	Optimizing <i>a</i> SPFDs with Don't Cares	274
15.3.3.1	Conflicts in Multiple Expected Traces	275
15.4	Logic Transformations with <i>a</i> SPFDs	277
15.4.1	SAT-Based Searching Algorithm	278
15.4.2	Greedy Searching Algorithm	279
15.5	Experimental Results	280
15.5.1	Logic Restructuring of Combinational Designs	280
15.5.2	Logic Restructuring of Sequential Designs	283
15.6	Summary	285
	References	285
16	Extracting Functions from Boolean Relations Using SAT and Interpolation	287
	Jie-Hong Roland Jiang, Hsuan-Po Lin, and Wei-Lun Hung	
16.1	Introduction	287
16.2	Previous Work	290
16.3	Preliminaries	290
16.3.1	Boolean Relation	290
16.3.2	Satisfiability and Interpolation	291
16.4	Our Approach	292
16.4.1	Single-Output Relation	292
16.4.1.1	Total Relation	292
16.4.1.2	Partial Relation	293
16.4.2	Multiple Output Relation	294
16.4.2.1	Determinization via Expansion Reduction	294
16.4.2.2	Determinization via Substitution Reduction	295
16.4.3	Deterministic Relation	296
16.4.4	Function Simplification	297
16.4.4.1	Support Minimization	297
16.4.4.2	Determinization Scheduling	298
16.5	Experimental Results	298
16.6	Chapter Summary	305
	References	306
17	A Robust Window-Based Multi-node Minimization Technique Using Boolean Relations	309
	Jeff L. Cobb, Kanupriya Gulati, and Sunil P. Khatri	
17.1	Introduction	309
17.2	Problem Definition	311
17.3	Previous Work	312
17.4	Preliminaries and Definitions	314
17.4.1	BREL Boolean Relation Minimizer	316

- 17.5 Approach 317
 - 17.5.1 Algorithm Details 318
 - 17.5.1.1 Selecting Node Pairs 318
 - 17.5.1.2 Building the Subnetwork 320
 - 17.5.1.3 Computing the Boolean Relation \mathcal{R}^Y 321
 - 17.5.1.4 Quantification Scheduling 322
 - 17.5.1.5 Endgame 324
- 17.6 Experimental Results 324
 - 17.6.1 Preprocessing Steps 325
 - 17.6.2 Parameter Selection 325
 - 17.6.2.1 Selecting α 325
 - 17.6.2.2 Selecting k_1 and k_2 327
 - 17.6.2.3 Selecting *thresh* 327
 - 17.6.3 Comparison of the Proposed Technique with *mfsw* 328
 - 17.6.4 Additional Experiments 330
 - 17.6.4.1 Running *relation* After *mfsw* 330
 - 17.6.4.2 Running *relation* Twice 331
 - 17.6.4.3 Minimizing Single Nodes 331
 - 17.6.4.4 Effects of Early Quantification 331
- 17.7 Chapter Summary 332
- References 333

Part V Applications to Specialized Design Scenarios

- 18 Synthesizing Combinational Logic to Generate Probabilities:
Theories and Algorithms 337**

Weikang Qian, Marc D. Riedel, Kia Bazargan, and David J. Lilja

 - 18.1 Introduction and Background 337
 - 18.2 Related Work 341
 - 18.3 Sets with Two Elements that Can Generate Arbitrary Decimal Probabilities 341
 - 18.3.1 Generating Decimal Probabilities from the Input Probability Set $S = \{0.4, 0.5\}$ 341
 - 18.3.2 Generating Decimal Probabilities from the Input Probability Set $S = \{0.5, 0.8\}$ 345
 - 18.4 Sets with a Single Element that Can Generate Arbitrary Decimal Probabilities 348
 - 18.5 Implementation 351
 - 18.6 Empirical Validation 355
 - 18.7 Chapter Summary 356
 - References 357

19 Probabilistic Error Propagation in a Logic Circuit Using the Boolean Difference Calculus 359

Nasir Mohyuddin, Ehsan Pakbaznia, and Massoud Pedram

19.1 Introduction 359

19.2 Error Propagation Using Boolean Difference Calculus 361

 19.2.1 Partial Boolean Difference 361

 19.2.2 Total Boolean Difference 362

 19.2.3 Signal and Error Probabilities 363

19.3 Proposed Error Propagation Model 364

 19.3.1 Gate Error Model 364

 19.3.2 Error Propagation in 2-to-1 Mux Using BDEC 367

 19.3.3 Circuit Error Model 369

19.4 Practical Considerations 370

 19.4.1 Output Error Expression 370

 19.4.2 Reconvergent Fanout 371

19.5 Simulation Results 373

19.6 Extensions to BDEC 377

 19.6.1 Soft Error Rate (SER) Estimation Using BDEC 377

 19.6.2 BDEC for Asymmetric Erroneous Transition Probabilities 379

 19.6.3 BDEC Applied to Emerging Nanotechnologies 379

19.7 Conclusions 379

References 380

20 Digital Logic Using Non-DC Signals 383

Kalyana C. Bollapalli, Sunil P. Khatri, and Laszlo B. Kish

20.1 Introduction 383

20.2 Previous Work 386

20.3 Our Approach 387

 20.3.1 Standing Wave Oscillator 387

 20.3.2 A Basic Gate 389

 20.3.2.1 Multiplier 389

 20.3.2.2 Low-Pass Filter 391

 20.3.2.3 Output Stage 391

 20.3.2.4 Complex Gates 392

 20.3.3 Interconnects 392

20.4 Experimental Results 393

 20.4.1 Sinusoid Generator 393

 20.4.2 Gate Optimization 395

 20.4.3 Gate Operation 397

20.5 Conclusions 399

References 400

21 Improvements of Pausible Clocking Scheme for High-Throughput and High-Reliability GALS Systems Design 401

Xin Fan, Milos Krstić, and Eckhard Grass

21.1 Introduction 401

21.2 Analysis of Pausible Clocking Scheme 402

 21.2.1 Local Clock Generators 402

 21.2.2 Clock Acknowledge Latency 403

 21.2.3 Throughput Reduction 404

 21.2.3.1 Demand-Output (D-OUT) Port to Poll-Input (P-IN) Port Channel 404

 21.2.3.2 Other Point-to-Point Channels 406

 21.2.3.3 Further Discussion on Throughput Reduction 406

 21.2.4 Synchronization Failures 407

 21.2.4.1 $\Delta_{LCIkRx} < T_{LCIkRx}$ 407

 21.2.4.2 $\Delta_{LCIkRx} \geq T_{LCIkRx}$ 408

21.3 Optimization of Pausible Clocking Scheme 409

 21.3.1 Optimized Local Clock Generator 409

 21.3.2 Optimized Input Port 410

 21.3.2.1 Double Latching Mechanism 410

 21.3.2.2 Optimized Input Port Controller 411

21.4 Experimental Results 412

 21.4.1 Input Wrapper Simulation 412

 21.4.2 Point-to-Point Communication 415

21.5 Conclusions 415

References 416

Subject Index 419



<http://www.springer.com/978-1-4419-7517-1>

Advanced Techniques in Logic Synthesis, Optimizations
and Applications

Gulati, K. (Ed.)

2011, XXI, 423 p., Hardcover

ISBN: 978-1-4419-7517-1