

# Contents

<b>I</b>	<b>Logic</b>	<b>11</b>
<b>1</b>	<b>Abstraction, Refinement, Enrichment</b>	<b>13</b>
1.1	Motivation and History . . . . .	13
1.2	Recapitulation of Kourie's 1989-Paper . . . . .	15
1.2.1	Preliminaries . . . . .	15
1.2.2	Domain of Discourse . . . . .	16
1.2.3	Property Definitions . . . . .	17
1.2.4	Abstraction . . . . .	18
1.2.5	Refinement/Enrichment . . . . .	19
1.2.6	Corollaries . . . . .	20
1.2.7	Base Abstractions . . . . .	21
1.2.8	Intermediate Summary . . . . .	22
1.3	Related Work . . . . .	23
1.3.1	Abstraction . . . . .	23
1.3.2	Refinement . . . . .	26
1.3.3	Enrichment . . . . .	29
1.3.4	Retrenchment . . . . .	30
1.3.5	Evolution . . . . .	33
1.3.6	Intermediate Summary . . . . .	34
1.4	Further Considerations . . . . .	34
1.4.1	Abstraction . . . . .	34
1.4.2	Refinement . . . . .	37
1.4.3	Enrichment . . . . .	37
1.4.4	Example . . . . .	38
1.4.5	Software Product Lines . . . . .	39
1.5	Summary and Outlook . . . . .	40
<b>2</b>	<b>Ceteris Paribus Preferences: Prediction via Abduction</b>	<b>45</b>
2.1	Introduction and Motivation . . . . .	45
2.2	Preliminaries . . . . .	47
2.3	Ceteris Paribus Preferences . . . . .	48
2.4	Preference Prediction . . . . .	50
2.5	Related Work . . . . .	54

2.6	Conclusion and Outlook . . . . .	55
<b>II</b>	<b>Automata</b>	<b>57</b>
<b>3</b>	<b>Minimal Weighted Automata over the Galois Field with Two Elements</b>	<b>59</b>
3.1	Introduction . . . . .	59
3.2	Definitions . . . . .	61
3.3	Main Results . . . . .	68
3.4	Future Work . . . . .	75
<b>4</b>	<b>Symmetric Difference NFA: the State of the Art</b>	<b>77</b>
4.1	Introduction . . . . .	77
4.1.1	Motivation . . . . .	77
4.1.2	History . . . . .	78
4.2	Definitions and Examples . . . . .	79
4.2.1	The Unary Case . . . . .	82
4.3	Descriptive Complexity — Succinctness . . . . .	85
4.4	Ambiguity . . . . .	88
4.5	Future Work . . . . .	92
<b>III</b>	<b>Stringology</b>	<b>93</b>
<b>5</b>	<b>Analyzing Strings with Ordered Lyndon-like Structures</b>	<b>95</b>
5.1	Introduction and Motivation . . . . .	96
5.1.1	Preliminaries . . . . .	97
5.2	Lyndon Words . . . . .	97
5.3	Unique Maximal Factorization Families . . . . .	98
5.3.1	Circ-UMFFs . . . . .	100
5.3.2	Generic Algorithms for Factoring circ-UMFFs . . . . .	102
5.4	Hybrid Lyndon Words . . . . .	103
5.4.1	V-Order and V-Words . . . . .	103
5.4.2	Algorithmics of V-Order . . . . .	107
5.5	Binary Factorizations . . . . .	108
5.5.1	Type-Order and T-Words . . . . .	108
5.5.2	Block-Order and B-Words . . . . .	108
5.6	Conclusion and Outlook . . . . .	109
<b>6</b>	<b>Verifying an Enhanced Cover Array in Linear Time</b>	<b>111</b>
6.1	Introduction and Motivation . . . . .	112
6.1.1	Our Contribution . . . . .	114
6.2	Definitions and Notations . . . . .	115
6.3	Properties of the Enhanced Cover Array . . . . .	116

6.4	Algorithms . . . . .	119
6.4.1	String Inference . . . . .	119
6.4.2	Verification . . . . .	121
6.5	Experimental Results . . . . .	121
6.6	Conclusion and Outlook . . . . .	123
<b>7</b>	<b>Singularities of Extremal Periodic Strings</b>	<b>125</b>
7.1	Introduction . . . . .	125
7.2	Parameterized Approach . . . . .	126
7.3	Basic Properties . . . . .	129
7.4	Computational Substantiation . . . . .	131
7.5	Conclusion . . . . .	132
<b>8</b>	<b>Are Three Squares Impossible?</b>	<b>135</b>
8.1	Introduction and Motivation . . . . .	135
8.2	Terminology and Notation . . . . .	136
8.3	Computing Repetitions and Runs . . . . .	140
8.3.1	Repetitions . . . . .	140
8.3.2	Runs . . . . .	140
8.4	Estimating the Number of Runs . . . . .	141
8.5	Overlapping Squares . . . . .	142
8.5.1	Case 1 . . . . .	143
8.5.2	Case 2 . . . . .	146
8.6	Conclusions and Outlook . . . . .	148
<b>IV</b>	<b>Programming</b>	<b>151</b>
<b>9</b>	<b>A Structured Approach to Concurrency with Zonnon</b>	<b>153</b>
9.1	Introduction . . . . .	153
9.2	Dialogs . . . . .	155
9.3	Activities . . . . .	157
9.3.1	Intrinsic Activities . . . . .	157
9.3.2	Dialog Activities . . . . .	157
9.3.3	Sub-Activities . . . . .	158
9.3.4	Execution Model . . . . .	159
9.3.5	Summary of Concurrency Model . . . . .	159
9.4	The Language at Work . . . . .	160
9.4.1	Frisbee Players . . . . .	161
9.4.2	The Committee Meeting Problem . . . . .	163
9.4.3	Monkey Rocks . . . . .	164
9.4.4	Santa Claus . . . . .	166
9.4.5	Concurrent Quicksort . . . . .	167

9.5	Static Validation . . . . .	168
9.6	Conclusion . . . . .	170
<b>10</b>	<b>Applying and Spicing-Up TABASCO: Taxonomy-based Software and how to increase its Usability</b>	<b>173</b>
10.1	Introduction and Motivation . . . . .	174
10.2	Historical Context . . . . .	175
10.3	TABASCO . . . . .	176
10.4	Applications of TABASCO . . . . .	179
10.5	Spicing it up: Increasing Usability . . . . .	180
10.6	Conclusion and Outlook . . . . .	183
	<b>Bibliography</b>	<b>185</b>