

# Table of Contents

## 1. Algorithm Engineering for Parallel Computation

David A. Bader, Bernard M. E. Moret, and Peter Sanders . . . . .	1
1.1 Introduction . . . . .	1
1.2 General Issues . . . . .	3
1.3 Speedup . . . . .	5
1.3.1 Why Speed? . . . . .	5
1.3.2 What is Speed? . . . . .	5
1.3.3 Speedup Anomalies . . . . .	6
1.4 Reliable Measurements . . . . .	7
1.5 Test Instances . . . . .	9
1.6 Presenting Results . . . . .	10
1.7 Machine-Independent Measurements? . . . . .	11
1.8 High-Performance Algorithm Engineering for Shared-Memory Processors . . . . .	12
1.8.1 Algorithms for SMPs . . . . .	12
1.8.2 Leveraging PRAM Algorithms for SMPs . . . . .	13
1.9 Conclusions . . . . .	15
References . . . . .	15
1.A Examples of Algorithm Engineering for Parallel Computation	20

## 2. Visualization in Algorithm Engineering:

### Tools and Techniques

Camil Demetrescu, Irene Finocchi, Giuseppe F. Italiano, and Stefan Näher . . . . .	24
2.1 Introduction . . . . .	24
2.2 Tools for Algorithm Visualization . . . . .	26
2.3 Interesting Events versus State Mapping . . . . .	30
2.4 Visualization in Algorithm Engineering . . . . .	33
2.4.1 Animation Systems and Heuristics: Max Flow . . . . .	33
2.4.2 Animation Systems and Debugging: Spring Embedding	39
2.4.3 Animation Systems and Demos: Geometric Algorithms	41
2.4.4 Animation Systems and Fast Prototyping . . . . .	43
2.5 Conclusions and Further Directions . . . . .	47
References . . . . .	48

**3. Parameterized Complexity: The Main Ideas and Connections to Practical Computing**

Michael R. Fellows ..... 51

    3.1 Introduction ..... 51

    3.2 Parameterized Complexity in a Nutshell ..... 52

        3.2.1 Empirical Motivation:  
            Two Forms of Fixed-Parameter Complexity ..... 52

    3.2.2 The Halting Problem: A Central Reference Point ..... 56

    3.3 Connections to Practical Computing and Heuristics ..... 58

    3.4 A Critical Tool for Evaluating Approximation Algorithms ... 64

    3.5 The Extremal Connection: A General Method Relating *FPT*,  
        Polynomial-Time Approximation, and Pre-Processing Based  
        Heuristics ..... 69

    References ..... 74

**4. A Comparison of Cache Aware and Cache Oblivious Static Search Trees Using Program Instrumentation**

Richard E. Ladner, Ray Fortna, and Bao-Hoang Nguyen ..... 78

    4.1 Introduction ..... 78

    4.2 Organization ..... 80

    4.3 Cache Aware Search ..... 80

    4.4 Cache Oblivious Search ..... 82

    4.5 Program Instrumentation ..... 84

    4.6 Experimental Results ..... 87

    4.7 Conclusion ..... 90

    References ..... 91

**5. Using Finite Experiments to Study Asymptotic Performance**

Catherine McGeoch, Peter Sanders, Rudolf Fleischer, Paul R. Cohen,  
and Doina Precup ..... 93

    5.1 Introduction ..... 93

    5.2 Difficulties with Experimentation ..... 97

    5.3 Promising Examples ..... 99

        5.3.1 Theory with Simplifications:  
            Writing to Parallel Disks ..... 99

        5.3.2 “Heuristic” Deduction: Random Polling ..... 100

        5.3.3 Shellsort ..... 102

        5.3.4 Sharpening a Theory:  
            Randomized Balanced Allocation ..... 103

    5.4 Empirical Curve Bounding Rules ..... 105

        5.4.1 Guess Ratio ..... 107

        5.4.2 Guess Difference ..... 107

5.4.3	The Power Rule . . . . .	108
5.4.4	The BoxCox Rule . . . . .	109
5.4.5	The Difference Rule . . . . .	110
5.4.6	Two Negative Results . . . . .	111
5.5	Experimental Results . . . . .	112
5.5.1	Parameterized Functions . . . . .	112
5.5.2	Algorithmic Data Sets . . . . .	118
5.6	A Hybrid Iterative Refinement Method . . . . .	120
5.6.1	Remark . . . . .	121
5.7	Discussion . . . . .	123
	References . . . . .	124
<b>6.</b>	<b>WWW.BDD-Portal.ORG: An Experimentation Platform for Binary Decision Diagram Algorithms</b>	
	Christoph Meinel, Harald Sack, and Arno Wagner . . . . .	127
6.1	Introduction . . . . .	127
6.1.1	WWW Portal Sites for Research Communities . . . . .	127
6.1.2	Binary Decision Diagrams . . . . .	128
6.2	A Benchmarking Platform for BDDs . . . . .	129
6.2.1	To Publish Code is not Optimal . . . . .	130
6.2.2	What is Really Needed . . . . .	131
6.3	A Web-Based Testbed . . . . .	131
6.3.1	The WWW Interface . . . . .	131
6.3.2	Implementation . . . . .	132
6.3.3	Available BDD Tools . . . . .	132
6.4	Added Value: A BDD Portal Site . . . . .	133
6.4.1	Structure of a Conventional Portal . . . . .	133
6.4.2	Shortcomings of Conventional Portals . . . . .	134
6.4.3	The BDD Portal . . . . .	134
6.5	Online Operation Experiences . . . . .	136
6.6	Related Work . . . . .	136
	References . . . . .	137
<b>7.</b>	<b>Algorithms and Heuristics in VLSI Design</b>	
	Christoph Meinel and Christian Stangier . . . . .	139
7.1	Introduction . . . . .	139
7.2	Preliminaries . . . . .	140
7.2.1	OBDDs – Ordered Binary Decision Diagrams . . . . .	140
7.2.2	Operations on OBDDs . . . . .	141
7.2.3	Influence of the Variable Order on the OBDD Size . . . . .	143
7.2.4	Reachability Analysis . . . . .	144
7.2.5	Image Computation Using AndExist . . . . .	145
7.3	Heuristics for Optimizing OBDD-Size — Variable Reordering . . . . .	147
7.3.1	Sample Reordering Method . . . . .	147

7.3.2	Speeding up Symbolic Model Checking with Sample Sifting .....	149
7.3.3	Experiments .....	151
7.4	Heuristics for Optimizing OBDD Applications – Partitioned Transition Relations .....	152
7.4.1	Common Partitioning Strategy .....	153
7.4.2	RTL Based Partitioning Heuristic .....	154
7.4.3	Experiments .....	156
7.5	Conclusion .....	157
	References .....	160
 <b>8. Reconstructing Optimal Phylogenetic Trees: A Challenge in Experimental Algorithmics</b>		
	Bernard M. E. Moret and Tandy Warnow .....	163
8.1	Introduction .....	163
8.2	Data for Phylogeny Reconstruction .....	165
8.2.1	Phylogenetic Reconstruction Methods .....	166
8.3	Algorithmic and Experimental Challenges .....	167
8.3.1	Designing for Speed .....	167
8.3.2	Designing for Accuracy .....	167
8.3.3	Performance Evaluation .....	168
8.4	An Algorithm Engineering Example: Solving the Breakpoint Phylogeny .....	168
8.4.1	Breakpoint Analysis: Details .....	169
8.4.2	Re-Engineering BPAanalysis for Speed .....	170
8.4.3	A Partial Assessment .....	172
8.5	An Experimental Algorithmics Example: Quartet-Based Methods for DNA Data .....	172
8.5.1	Quartet-Based Methods .....	172
8.5.2	Experimental Design .....	174
8.5.3	Some Experimental Results .....	175
8.6	Observations and Conclusions .....	176
	References .....	178
 <b>9. Presenting Data from Experiments in Algorithmics</b>		
	Peter Sanders .....	181
9.1	Introduction .....	181
9.2	The Process .....	182
9.3	Tables .....	183
9.4	Two-Dimensional Figures .....	184
9.4.1	The <i>x</i> -Axis .....	184
9.4.2	The <i>y</i> -Axis .....	187
9.4.3	Arranging Multiple Curves .....	188
9.4.4	Arranging Instances .....	190

9.4.5	How to Connect Measurements .....	191
9.4.6	Measurement Errors .....	191
9.5	Grids and Ticks .....	192
9.6	Three-Dimensional Figures .....	194
9.7	The Caption .....	194
9.8	A Check List .....	194
	References .....	195
<b>10.</b>	<b>Distributed Algorithm Engineering</b>	
	Paul G. Spirakis and Christos D. Zaroliagis .....	197
10.1	Introduction .....	197
10.2	The Need of a Simulation Environment .....	200
	10.2.1 An Overview of Existing Simulation Environments ...	202
10.3	Asynchrony in Distributed Experiments .....	204
10.4	Difficult Input Instances for Distributed Experiments .....	206
	10.4.1 The Adversarial-Based Approach .....	206
	10.4.2 The Game-Theoretic Approach .....	209
10.5	Mobile Computing .....	212
	10.5.1 Models of Mobile Computing .....	213
	10.5.2 Basic Protocols in the Fixed Backbone Model .....	214
	10.5.3 Basic Protocols in the Ad-Hoc Model .....	218
10.6	Modeling Attacks in Networks:	
	A Useful Interplay between Theory and Practice .....	222
10.7	Conclusion .....	226
	References .....	226
<b>11.</b>	<b>Implementations and Experimental Studies of Dynamic Graph Algorithms</b>	
	Christos D. Zaroliagis .....	229
11.1	Introduction .....	229
11.2	Dynamic Algorithms for Undirected Graphs .....	231
	11.2.1 Dynamic Connectivity .....	231
	11.2.2 Dynamic Minimum Spanning Tree .....	243
11.3	Dynamic Algorithms for Directed Graphs .....	252
	11.3.1 Dynamic Transitive Closure .....	252
	11.3.2 Dynamic Shortest Paths .....	264
11.4	A Software Library for Dynamic Graph Algorithms .....	271
11.5	Conclusions .....	273
	References .....	274
	<b>Author Index .....</b>	<b>279</b>

Experimental Algorithmics

From Algorithm Design to Robust and Efficient Software

Fleischer, R.; Moret, B.; Meineche Schmidt, E. (Eds.)

2002, XVIII, 286 p., Softcover

ISBN: 978-3-540-00346-5