

---

# Contents

---

<b>List of Figures</b>	<b>xii</b>
<b>List of Tables</b>	<b>xv</b>
<b>1 Introduction</b>	<b>1</b>
1.1 The Evolvable Hardware Principle . . . . .	1
1.2 Challenges of Evolvable Hardware . . . . .	4
1.3 Thesis Contributions . . . . .	8
1.4 Thesis Organization . . . . .	11
<b>2 Evolutionary Algorithms</b>	<b>13</b>
2.1 Historical overview . . . . .	18
2.1.1 Evolutionary Programming . . . . .	18
2.1.2 Evolutionary Strategies . . . . .	19
2.1.3 Genetic Algorithms . . . . .	20
2.1.4 Genetic Programming . . . . .	21
2.2 Multi-objective Evolutionary Algorithms . . . . .	22
2.2.1 Multi-objective Optimization Terminology . . . . .	22
2.2.2 Non-Dominated Sorting Genetic Algorithm II . . . . .	26
2.2.3 Strength Pareto Evolutionary Algorithm 2 . . . . .	27
2.3 Variants of Specialized Evolutionary Algorithms . . . . .	29
2.3.1 Parallel Evolutionary Algorithms . . . . .	29
2.3.2 Hybrid Evolutionary Algorithms . . . . .	30
2.4 Performance Analysis . . . . .	31
2.4.1 The Computational Effort . . . . .	31
2.4.2 Comparing Non-Dominated Sets . . . . .	32
2.5 Chapter Conclusion . . . . .	34
<b>3 Evolvable Hardware</b>	<b>35</b>
3.1 The Basic Idea . . . . .	36
3.2 Digital Logic Evolvable Hardware . . . . .	40

## CONTENTS

---

3.2.1	Reconfigurable Logic Devices . . . . .	40
3.2.2	Programmable logic array Evolvable Hardware . . . . .	52
3.2.3	Unconstrained Evolution . . . . .	53
3.2.4	The POETic Tissue Project . . . . .	58
3.2.5	Evolvable Components . . . . .	59
3.2.6	The Functional Unit Row Architecture . . . . .	65
3.3	Analog Evolvable Hardware . . . . .	68
3.3.1	Heidelberg Field Programmable Transistor Array . . . . .	69
3.3.2	NASA Field Programmable Transistor Array . . . . .	73
3.3.3	Lattice ispPAC . . . . .	76
3.4	Cartesian Genetic Programming . . . . .	77
3.4.1	The Representation Model . . . . .	79
3.4.2	Enzyme Cartesian Genetic Programming . . . . .	86
3.4.3	Recombination for Cartesian Genetic Programming . . . . .	87
3.4.4	Automatic Acquisition and Reuse of Subfunctions in Cartesian Genetic Programming . . . . .	88
3.4.5	Self-modifying Cartesian Genetic Programming . . . . .	90
3.5	Chapter Conclusion . . . . .	91
<b>4</b>	<b>Digital Logic Evolution Framework and Experimentation Toolbox</b> . . . . .	<b>93</b>
4.1	Framework Core Architecture . . . . .	93
4.2	Representation Models and Algorithms . . . . .	98
4.2.1	General Methods . . . . .	99
4.2.2	Methods Specific to a Genotype . . . . .	101
4.3	Experimentation Support and Tools . . . . .	104
4.3.1	Visualization and Analysis . . . . .	104
4.3.2	Command Line Tools and Distributed Simulation . . . . .	107
4.4	Chapter Conclusion . . . . .	108
<b>5</b>	<b>Evolvable Hardware Abstraction Models and Operators</b> . . . . .	<b>109</b>
5.1	Structural Crossover for Cartesian Genetic Programming . . . . .	111
5.1.1	The CGP Crossover Algorithm . . . . .	112
5.1.2	Evaluation . . . . .	114
5.2	Automatic Acquisition of Structural and Age Based Modules . . . . .	117
5.2.1	Age Based Module Creation . . . . .	117
5.2.2	Cone Based Module Creation . . . . .	118
5.2.3	Evaluation . . . . .	118
5.3	Chapter Conclusion . . . . .	120
<b>6</b>	<b>Efficient Multi-Objective Optimization for CGP</b> . . . . .	<b>121</b>
6.1	Selection Schemes for Multi-Objective CGP . . . . .	122
6.1.1	Objective Prioritization . . . . .	123
6.1.2	Evaluation . . . . .	126
6.2	Evolutionary Algorithm Periodization . . . . .	137

6.2.1	The Periodization Model . . . . .	138
6.2.2	Hybrid Evolutionary Strategies . . . . .	139
6.2.3	Evaluation . . . . .	140
6.3	Chapter Conclusion . . . . .	151
<b>7</b>	<b>Evolvable Hardware Applications</b>	<b>153</b>
7.1	ECGP Based Architecture for Electromyographic Signal Classification . . . . .	154
7.1.1	Classification of Electromyographic Signals . . . . .	155
7.1.2	Recording Electromyographic Signals . . . . .	159
7.1.3	Electromyographic Signals Processing and Feature Extraction . . . . .	162
7.1.4	The Embedded Cartesian Genetic Programming Classification Architecture . . . . .	163
7.1.5	Conventional Classifiers . . . . .	166
7.1.6	Evaluation . . . . .	168
7.2	Classifier Adaptation to Resource Fluctuations . . . . .	176
7.2.1	The Reconfigurable Functional Unit Row Architecture	177
7.2.2	Evaluation . . . . .	177
7.2.3	Functional Unit Row Architecture Reconfiguration Schemes . . . . .	182
7.2.4	Evaluation . . . . .	184
7.3	Adaptation of Cache Mappings . . . . .	190
7.3.1	The Concept of EvoCaches . . . . .	190
7.3.2	Algorithms, Representation Models, and Metrics . . . . .	192
7.3.3	Evaluation . . . . .	194
7.4	Chapter Conclusion . . . . .	201
<b>8</b>	<b>Summary and Outlook</b>	<b>203</b>
8.1	Contributions . . . . .	203
8.2	Conclusions and Lessons Learned . . . . .	206
8.3	Future Directions . . . . .	207
	<b>Author's Publications</b>	<b>211</b>
	<b>Bibliography</b>	<b>215</b>