

Contents

List of Figures	xi
List of Tables	xiii
List of Algorithms	xv
List of Listings	xvii
Acronyms	xix
1. Introduction	1
1.1. Motivation	1
1.2. Verification of SystemC Designs	1
1.2.1. Simulation-Based Methods	1
1.2.2. Formal Methods	2
1.3. Thesis Goal: Complete Symbolic Simulation	3
1.4. Thesis Contributions and Organization	4
2. Preliminaries	7
2.1. SystemC	7
2.1.1. Simulation Semantics	8
2.2. SystemC Intermediate Verification Language	9
2.2.1. Example	10
2.3. Symbolic Execution	11
2.4. State Transition System	12
2.4.1. Modeling IVL Programs as STS	13
2.4.2. Example	14
2.4.3. Remarks	15
2.5. Basic Stateful Model Checking Algorithm	16
2.6. Partial Order Reduction	16
2.6.1. Definitions	18
2.6.2. Computing Static Persistent Sets	22
2.6.3. Dynamic Partial Order Reduction	26
3. Static Partial Order Reduction in Stateful Model Checking	33
3.1. Preserving Properties of Interest	33
3.1.1. Deadlocks	33
3.1.2. Ignoring Problem	34
3.1.3. Safety Properties	35
3.2. Static Partial Order Reduced Exploration	37
3.2.1. Preserving Deadlocks	37
3.2.2. Cycle Proviso	37
3.2.3. Preserving Assertion Violations	39

4. Dynamic Partial Order Reduction in Stateful Model Checking	41
4.1. Missing Dependency Problem	42
4.2. Stateful Exploration of Finite Acyclic State Spaces	43
4.3. Supporting Finite Cyclic State Spaces	45
4.3.1. Missing Dependency Problem in Cyclic State Spaces	45
4.3.2. C-SDPOR Algorithm	47
4.3.3. Notes on Correctness	49
4.4. Complete Stateful Exploration	49
5. State Subsumption Reduction	53
5.1. Definitions	54
5.1.1. Execution State	55
5.2. Stateful Model Checking with State Subsumption Reduction	59
5.3. Combination with Partial Order Reduction	60
5.3.1. Integration of SSR into the AVPE Algorithm	61
5.3.2. Integration of SSR into the SDPOR Algorithm	61
5.4. State Matching	63
5.4.1. Matching Concrete State Parts	63
5.4.2. Exact Symbolic Subsumption	64
5.4.3. Relaxing Time Equality Requirements	66
6. Heuristic Symbolic Subsumption	69
6.1. Explicit Structural Matching	70
6.1.1. Basic Method	70
6.1.2. Simplifying Symbolic Expressions	70
6.1.3. Hashing of Symbolic State Parts	76
6.2. Solver-Based Comparison	76
6.2.1. Detecting Equivalent States	76
6.2.2. Detecting State Subsumption	78
6.3. Fresh Symbolic Literal Problem	79
6.4. Extended Explicit Structural Matching	81
6.4.1. Extended Algorithm	81
6.4.2. Examples	83
6.4.3. Comparison with Base Version	83
6.5. Extended Solver-Based Comparison	83
6.5.1. Example: Symmetric Accumulator	86
6.5.2. Generating Consistent Equality Assumptions	88
6.5.3. Breaking Implicit Equality Assumptions	90
6.6. Classification of State Subsumption Algorithms	91
6.7. Further Optimization	92
6.7.1. Garbage Collecting Path Condition Constraints	92
6.7.2. Combining the Explicit and Solver-Based Method	94
7. Experiments	97
7.1. Configuration Overview	97
7.2. Benchmark Overview	99
7.3. Comparing Hashing Methods for Symbolic State Parts	101
7.4. Comparing SMT Solvers	104

7.5. Comparing State Subsumption Matching Methods	105
7.5.1. Discussion of the Observations	105
7.5.2. Result Summary	109
7.6. Comparing POR Implementations	109
7.7. Comparing with Kratos	112
8. Conclusion	117
8.1. Future Work	118
A. Appendix	121
A.1. Generating Consistent Equality Assumptions	121
A.1.1. Definitions	121
A.1.2. First Algorithm	122
A.1.3. Second Algorithm	124
A.2. POR Correctness Proof: Deadlock Preserving	131
A.3. POR Correctness Proof: Assertion Violations Preserving	133
A.3.1. Proof of Theorem A.3.1	134
A.4. SSR without POR Correctness Proof: Assertion Violation Preserving	136
A.5. SSR with POR Correctness Proof: Assertion Violations Preserving	137
A.6. Reduction Function Condition Proofs	139
A.7. Refined Assertion Violation Preserving Exploration	140
A.8. SDPOR Algorithm	143
A.9. Solving the Ignoring Problem implicitly	145
A.9.1. Static Partial Order Reduction	145
A.9.2. Dynamic Partial Order Reduction	147
A.9.3. Discussion	148
A.10. Correctness Proof of the Exact Symbolic Subsumption	148
A.10.1. Preliminary Part	149
A.10.2. Main Part	152
Bibliography	157

Complete Symbolic Simulation of SystemC Models
Efficient Formal Verification of Finite Non-Terminating
Programs

Herd, V.

2016, XIX, 162 p. 26 illus., Softcover

ISBN: 978-3-658-12679-7