Table of Contents

1. Introduction 1

2. A Hierarchy of Constants 13
2.1 A Taxonomy of Constants 16
2.1.1 Flow Graphs 16
2.1.2 May- and Must-Constants 17
2.1.3 Weakened Constant Detection Problems 19
2.1.4 Classes of Integer Constants 21
2.2 Known Results 22
2.3 New Undecidability Results 24
2.4 New Intractability Results 25
2.5 Summary 29

3. Deciding Constants by Effective Weakest Preconditions . . . 31
3.1 Presburger and Polynomial Constants 32
3.2 Presburger-Constant Detection at a Glance 33
3.3 A Generic Algorithm 37
3.4 Detection of Presburger Constants 40
3.5 A Primer on Computable Ideal Theory 43
3.6 More About $\mathbb{Z}[x_1, \ldots, x_n]$ 45
3.6.1 $\mathbb{Z}[x_1, \ldots, x_n]$ as a Complete Lattice 45
3.6.2 Zeros 45
3.6.3 Substitution 46
3.6.4 Projection 46
3.7 Detection of Polynomial Constants 47
3.8 Conclusion 49

4. Limits of Parallel Flow Analysis 53
4.1 A Motivating Example 55
4.2 Parallel Programs 56
4.3 Interprocedural Copy-Constant Detection 57
4.3.1 Two-Counter Machines 58
4.3.2 Constructing a Program 59
4.3.3 Correctness of the Reduction 62

4.4 Intraprocedural Copy-Constant Detection 62
4.5 Copy-Constant Detection in Loop-Free Programs 66
4.6 Beyond Fork/Join Parallelism 67
4.7 Owicki/Gries-Style Program Proofs 67
4.8 Correctness of the Reduction in Section 4.3 68
4.8.1 Enriching the Program 68
4.8.2 The Proof Outlines 69
4.8.3 Interference Freedom 72
4.9 Correctness of the Reduction in Section 4.4 73
4.9.1 Enriching the Program 73
4.9.2 An Auxiliary Predicate 73
4.9.3 Proof Outline for π_0 74
4.9.4 Proof Outline for $\pi_i(r)$ 75
4.9.5 Proof Outline for *Main* 76
4.9.6 Interference Freedom 77
4.10 Conclusion 78

5. Parallel Flow Graphs 81
5.1 Parallel Flow Graphs 82
5.2 Operational Semantics 84
5.3 Atomic Runs 86
5.4 The Run Sets of Ultimate Interest 87
5.5 The Constraint Systems 88
5.5.1 Same-Level Runs 88
5.5.2 Inverse Same-Level Runs 90
5.5.3 Two Assumptions and a Simple Analysis 91
5.5.4 Reaching Runs 92
5.5.5 Terminating Runs 94
5.5.6 Bridging Runs 94
5.5.7 The General Case 96
5.6 Discussion 98

6. Non-atomic Execution 101
6.1 Modeling Non-atomic Execution by Virtual Variables 103
6.2 A Motivating Example 105
6.3 The Domain of Non-atomic Run Sets 106
6.3.1 Base Statements 107
6.3.2 Sequential Composition 108
6.3.3 Interleaving Operator 108
6.3.4 Pre-operator 109
6.3.5 Post-operator 109
6.4 Conclusion 109

7. Dependence Traces ... 111
7.1 Transparency and Dependences ... 113
7.2 Dependence Traces ... 114
7.3 Implication Order ... 116
7.4 Subsumption Order ... 117
7.5 A Lattice of Antichains ... 118
7.6 Short Dependence Traces ... 121
7.7 The Abstract Domain ... 124
7.8 Pre-operator ... 126
7.9 Post-operator ... 128
7.10 Sequential Composition ... 128
7.11 Interleaving ... 130
7.11.1 Complementary Dependence Traces ... 131
7.11.2 Interleaving Operator ... 132
7.11.3 Soundness Lemmas ... 132
7.11.4 Completeness Lemmas ... 136
7.11.5 Proof of Theorem 7.11.1 ... 139
7.12 Base Edges ... 140
7.13 Running Time ... 141
7.14 Discussion ... 142

8. Detecting Copy Constants and Eliminating Faint Code ... 145
8.1 Copy-Constant Detection ... 146
8.2 Faint-Code Elimination ... 148
8.3 Running Time ... 150
8.4 Conclusion ... 152

9. Complexity in the Non-atomic Scenario ... 153
9.1 The SAT-reduction ... 154
9.2 Towards Stronger Lower Bounds ... 156
9.2.1 Assignment Statements That Propagate Twice ... 157
9.2.2 Propagating Runs of Exponential Length ... 159
9.3 Summary ... 160

10. Conclusion ... 161
10.1 Future Research ... 163

A. A Primer on Constraint-Based Program Analysis ... 165

References ... 173

http://www.springer.com/978-3-540-45385-7

Variations on Constants
Flow Analysis of Sequential and Parallel Programs
Müller-Olm, M.
2006, XIII, 177 p., Softcover
ISBN: 978-3-540-45385-7