

Lecture Notes in Computer Science

Edited by G. Goos and J. Hartmanis

58

Michael J. O'Donnell

Computing in Systems
Described by Equations



Springer-Verlag
Berlin Heidelberg New York 1977

Editorial Board

P. Brinch Hansen D. Gries C. Moler G. Seegmüller
J. Stoer N. Wirth

Author

Michael J. O'Donnell
Department of Computer Science
Purdue University
West Lafayette, Indiana 47907/USA

AMS Subject Classifications (1970): 02-02, 68-02, 68A05,
CR Subject Classifications (1974): 4.13, 5.21, 5.27

ISBN 3-540-08531-9 Springer-Verlag Berlin Heidelberg New York
ISBN 0-387-08531-9 Springer-Verlag New York Heidelberg Berlin

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically those of translation, re-printing, re-use of illustrations, broadcasting, reproduction by photocopying machine or similar means, and storage in data banks.

Under § 54 of the German Copyright Law where copies are made for other than private use, a fee is payable to the publisher, the amount of the fee to be determined by agreement with the publisher.

© by Springer-Verlag Berlin Heidelberg 1977

Printed in Germany

Printing and binding: Beltz Offsetdruck, Hemsbach/Bergstr.
2144/3140-543210

Dedication

To the Rocky Mountains

Preface

This monograph is an attempt to provide some useful mathematical foundations for the design of interpreters for functional programming languages such as LISP. These languages are often described by sets of equations. One way to attack the problem of interpreting is to study ways of using equations to compute simple output expressions from input expressions. Such computations are studied here in terms of an abstract formalism called a Subtree Replacement System. With the hints given in Example 17 and Appendix A, a good programmer should be able to design efficient implementations of the abstract computations in a pointer manipulation system with garbage collection, and all of the mathematical results are intended to give useful information on the correctness, termination and efficiency properties of such implementations.

The simplest way to intuitively understand the meanings of equations and the relationship of computations to the equations on which they are based is through the traditional logic of equations presented in Chapter II, Sections 1-3. For readers who prefer the more recent logic based on the work of Dana Scott, which involves notions of continuity and fixpoints in partially ordered domains, and for those who wish to compare the Subtree Replacement System approach to the recent literature on recursive equations, special sections marked with a star (*) are provided to sketch the relationship of continuous logic to this monograph. The starred sections are intended only as a brief outline of this relationship, and proofs given in these sections are deliberately sketchy. Readers without a special interest in continuous logic may omit the starred sections with no loss of continuity.

Acknowledgements

Not only did Barry Rosen's thesis serve as the spark which touched off this work, but his thorough critiques of the early drafts were tremendously helpful to me. Thanks are due also to Robert Constable, who supervised the development of the first version into a doctoral dissertation at Cornell University. Gérard Berry and Jean-Jacques Lévy discovered a significant error in the dissertation, which has been corrected in this monograph. The typed copy was produced cheerfully by Teresa Miao and Olivia Stansbury. The research and writing took place at Cornell University and the University of Toronto, and were supported in part by NSF grants DCR 75-09433 and MCS 76-14293, as well as NCR grant A7755.

Table of Contents

Preface	V
Acknowledgements	VII
Table of Definitions	XI
Lemmas	XII
Theorems	XIII
Corollaries	XIV
Counterexamples	XIV
Examples	XIV
Chapter I: Introduction	1
1. Procedural vs. Descriptive Languages	1
2. Equational Logic and Computation	2
Chapter II: Syntax, Semantics and Proof Theory for Equational Logic	4
1. A Formal Syntax for Expressions	4
2. Semantics for Expressions and Equations	8
3. Equational Proofs	11
*4. Continuous Semantics for Equations	13
Chapter III: Subtree Replacement Systems	18
1. A Formalism for Reductions	18
2. Sequences of Reductions--Informal Treatment	19
3. A Formal Notation for Reduction Steps and Sequences	20
4. Residuals	22
Chapter IV: The Confluence Property and the Church-Rosser Property	28
1. Motivation	28
2. Closure	29
3. Proof of the Confluence Property	33
*4. Continuous Semantics and the Confluence Property	38
Chapter V: Termination	40
1. Motivation	40
2. Reducing a Reduction Sequence	40
3. Complete Reduction Sequences	41
*4. Continuous Semantics and Complete Sequences	43
5. Eventually Outermost Reduction Sequences	43
6. The Outer Property	45
7. Proof That Outermost Sequences Terminate	47
*8. Continuous Semantics and Outermost Sequences	51
9. Strictly Innermost Sequences	51

Chapter VI: Costs of Reduction Sequences	54
1. A Cost Measure for Noncopying Sequences	54
2. Commutativity of Residuals	59
3. Strictly Innermost Sequences are Most Costly	60
4. Some Strictly Outermost Sequence is Optimal	61
5. Finding an Optimal Sequence Using a Dominance Ordering	63
Chapter VII: Schematic Definitions of SRSs	66
1. Rule Schemata	66
2. Closure, Commutativity, The Outer Property in Schemata-Generated SRSs	68
3. Weak Reduction in the S-K-I Combinator Calculus	71
Chapter VIII: Recursively Defined Functions	72
1. Recursive Equations Represented by Rule Schemata	72
2. Rule Schemata for Primitive Functions	73
3. Dominance Orderings for Recursive Equations	74
*4. Continuous Semantics for Recursive Equations	76
*5. Continuous Semantics for Primitive Functions	78
Chapter IX: Applications of SRS Theory to LISP and Lucid	80
1. LISP Interpreters	80
2. Interpreting Recursively Defined Functions of S-Expressions	80
3. Interpreting Functions Defined by S-Expressions	82
4. Lucid Interpreters	84
5. A SRS Based Interpreter For Lucid	85
Chapter X: Another Approach to Confluence, Termination, Optimality	89
1. Pseudoresidual Maps	89
2. The General Pseudoresidual Lemma	91
3. Confluence, Termination, Optimality	97
4. β -Reduction in the λ -Calculus	98
Further Research	102
Appendix A: A Problem in Implementing Noncopying Reduction	105
Bibliography	109

Table of Definitions, Lemmas, Theorems,
Corollaries, Counterexamples and Examples

Definitions

Definition 1	Σ -term	4
Definition 2	P and P^*	4
Definition 3	Tree Domain	4
Definition 4	Σ -tree	5
Definition 5	Notational Conventions for Trees	6
Definition 6	Interpretation	8
Definition 7	Valuation	9
Definition 8	Pseudointerpretation	9
Definition 9	Validity	9
Definition 9'	Validity with Variables	10
Definition 10	Proof	11
Definition 11	CPO, chain	13
Definition 12	Constructions of CPOs, Continuity	14
Definition 13	Continuous Interpretation	15
Definition 14	v^A	16
Definition 15	<u>def</u>	17
Definition 16	Subtree Replacement System (SRS)	18
Definition 17	Redex	20
Definition 18	\xrightarrow{x}	20
Definition 19	\xrightarrow{M} Where $\perp M$	21
Definition 20	$\xrightarrow{1}$	22
Definition 21	Reduction Sequence	22
Definition 22	Residual Map (r)	23
Definition 23	Extended Residual Map (\hat{r})	25
Definition 24	Confluence Property	28
Definition 25	Closure	30
Definition 26	Terminating Class of Reduction Sequences	40
Definition 27	Reduction of a Reduction Sequence	41
Definition 28	Complete Reduction Sequences (C)	42
Definition 29	Outermost Redexes (<u>omr</u>)	44
Definition 30	Elimination of Outermost Redexes	44
Definition 31	Eventually Outermost Reduction Sequences (O^e)	45
Definition 32	Outer Property	46
Definition 33	Innermost Redexes (<u>imr</u>)	51
Definition 34	Strictly Innermost Reduction Sequences (I^S)	52
Definition 35	Innermost Preserving Residual Maps	52
Definition 36	Nonterminating Sequences	53

Definition 37	Noncopying Sequences, Cost, Equivalent Address Map	55
Definition 38	Commutativity	59
Definition 39	Strictly Outermost Sequences (O^S)	61
Definition 40	\hat{r}^{-1}	61
Definition 41	Dominance Ordering, d-outer	63
Definition 42	Preorder, Leftmost Outermost Reduction	64
Definition 43	d-Outermost Redexes (omr_d), Eventually d-Outermost Reduction Sequences (O_d^e)	64
Definition 44	Strictly d-Outermost Sequences (O_d^S)	65
Definition 45	Rule Schema	66
Definition 46	Instance of a Rule Schema, Set of Rules Generated by a Rule Schema	66
Definition 47	Map Generated by a Rule Schema	66
Definition 48	Nonoverlapping Set of Rule Schemata	68
Definition 49	Consistent Schemata	69
Definition 50	Outer Set of Rule Schemata	70
Definition 51	Sequential Set of Schemata	75
Definition 19'	\xrightarrow{M} Where $\neg(\text{LM})$	89
Definition 22'	Pseudoresidual Map (r)	89
Definition 23'	Extended Pseudoresidual Map (\hat{r})	90
Definition 25'	Pseudoclosure	91
Definition 38'	Commutativity	91

Lemmas

Lemma 1	Reordering of Substitution in a Proof	12
Lemma 2	Reduction and Proof	12
Lemma 3	Least Upper Bounds of Directed Sets	14
Lemma 4	Least Fixpoints	14
Lemma 5	Independent Reductions Commute	21
Lemma 6	Extended Residuals	25
Lemma 7	Independence of Extended Residuals	27
Lemma 8	Extended Residuals	27
Lemma 9	Extended Residuals of Redexes are Redexes	33
Lemma 10	Weakened Form of Confluence	33
Lemma 11	Weakened Form of Confluence	33
Lemma 12	General Residual Lemma	35
Lemma 13	Convergence of Two Complete Reduction Sequences	42
Lemma 14	Outermost Redexes	46
Lemma 15	Outermost Redexes	46
Lemma 16	Outermost Redexes	47
Lemma 17	Reduction of Eventually Outermost Sequences	48

Lemma 18	Nonterminating Sequences	53
Lemma 19	Noncopying Sequences	59
Lemma 20	Generalized Commutativity	59
Lemma 21	Innermost Reduction of a Noncopying Sequence	60
Lemma 22	Reordering Reduction Steps in a Noncopying Sequence	62
Lemma 23	Nonoverlapping Property	70
Lemma 24	d-Outer Property for Recursive Equations	75
Lemma 25	Commutativity of λ -calculus	99
Lemma 26	d-Outer Property for λ -calculus	101
Lemma 8'	Extended Pseudoresiduals	90
Lemma 9'	Extended Pseudoresiduals of Redexes are Redexes	91
Lemma 10'	Weakened Form of Confluence	92
Lemma 11'	Weakened Form of Confluence	93
Lemma 12'	General Pseudoresidual Lemma	93

Theorems

Theorem 1	Correctness and Completeness	11
Theorem 2	$\hat{V}^A A = \text{def } A$	17
Theorem 3	Equivalence of Confluence and Church-Rosser Properties	28
Theorem 4	Confluence Property Implies Unique Normal Forms	29
Theorem 5	Completeness of Reduction	29
Theorem 6	Comfluence Property for Closed SRSs	38
Theorem 7	Semantics of Reduction	38
Theorem 8	Complete Sequence Theorem	42
Theorem 9	Semantics of Complete Sequences	43
Theorem 10	Eventually Outermost Sequence Termination	50
Theorem 11	Strictly Innermost Sequence Theorem	53
Theorem 12	Cost of Strictly Innermost Sequences	60
Theorem 13	Optimal Sequence Theorem	63
Theorem 14	Eventually d-outermost Sequence Termination	65
Theorem 15	Optimal Sequence Theorem	65
Theorem 16	Residual Map Generated by a Set of Rule Schemata	67
Theorem 17	Closure, Commutativity for Nonoverlapping Sets of Rule Schemata	70
Theorem 18	Outer Set of Schemata Generates Outer SRS	71
Theorem 19	Recursive Equation Theorem	73
Theorem 20	Total Dominance Ordering for Sequential Sets of Schemata	75
Theorem 21	Recursive Equation Semantics	76
Theorem 22	Lambda Calculus Theorem	101

Theorem 6'	Confluence Property	97
Theorem 8'	Complete Sequence Theorem	97
Theorem 11'	Strictly Innermost Theorem	97
Theorem 12'	Cost of Strictly Innermost Sequence	97
Theorem 13'	Optimal Sequence Theorem	97
Theorem 15'	Optimal Sequence Theorem	97

Corollaries

Corollary 1	$\hat{\vee}^A = \text{def } A$	17
Corollary 2	Corollary to General Residual Lemma	38
Corollary 3	Complete Sequence Termination	43
Corollary 4	Correctness of Recursive Computations	77

Counterexamples

Counterexample 1	Eventually Outermost Class Not Terminating	45
Counterexample 2	Strictly Innermost Terminating Sequence	52

Examples

Example 1	Tree	5
Example 2	Set of Σ -trees	7
Example 3	Interpretation	10
Example 4	Proof	13
Example 5	Continuous Interpretation	15
Example 6	Subtree Replacement System (SRS)	18
Example 7	Informal Reduction Sequence	19
Example 8	Redexes	20
Example 9	\xrightarrow{x}	20
Example 10	\xrightarrow{M} Where $\perp M$	21
Example 11	Residual Map (r)	23
Example 12	Extended Residual Map (\hat{r})	25
Example 13	Closure	30
Example 14	Nonclosed SRS	32
Example 15	Normal Form and Infinite Reduction Sequence	40
Example 16	Wastefulness of Complete Reduction Sequence	43
Example 17	Implementing Noncopying Reduction With Pointers	54
Example 18	Noncopying Sequence	56
Example 19	Rules and Map Generated by a Set of Rule Schemata	68
Example 20	Overlapping Schemata	68