

M. Rusinowitch J. L. Rémy (Eds.)

ccc1-656

# Conditional Term Rewriting Systems

Third International Workshop, CTRS-92  
Pont-à-Mousson, France, July 8-10, 1992  
Proceedings

**Springer-Verlag**

Berlin Heidelberg New York  
London Paris Tokyo  
Hong Kong Barcelona  
Budapest

## Series Editors

Gerhard Goos  
 Universität Karlsruhe  
 Postfach 69 80  
 Vincenz-Priessnitz-Straße 1  
 W-7500 Karlsruhe, FRG

Juris Harmanis  
 Cornell University  
 Department of Computer Science  
 4130 Upson Hall  
 Ithaca, NY 14853, USA

## Volume Editors

Michaël Rusinowitch  
 Jean-Luc Rémy  
 CRIN & INRIA-Lorraine  
 B. P. 239, F-54506 Vandoeuvre-Les-Nancy Cedex, France

CR Subject Classification (1991): F.4.1-2, D.3.1, I.2.3

267

ISBN 3-540-56393-8 Springer-Verlag Berlin Heidelberg New York  
 ISBN 0-387-56393-8 Springer-Verlag New York Berlin Heidelberg

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable for prosecution under the German Copyright Law.

© Springer-Verlag Berlin Heidelberg 1993  
 Printed in Germany

Typesetting: Camera ready by author  
 45/3140-543210 - Printed on acid-free paper

To the memory of Stéphane Kaplan



## Preface

This volume contains the papers presented at the Third International Workshop on Conditional Term Rewriting Systems held on 8-10 July 1992 at Pont-à-Mousson, France.

The first CTRS workshop was held in 1987 at the Université of Paris XI in Orsay and the second one took place in 1990 at Concordia University in Montreal. The proceedings have been published in the Lecture Notes in Computer Science, Springer Verlag, volume 308 and volume 516 respectively. Topics of these workshops include conditional rewriting and its applications to programming languages, specification languages, automated deduction, constrained rewriting, typed rewriting, higher-order rewriting, graph-rewriting.

We would like to thank the *Région Lorraine*, the *Centre de Recherche en Informatique de Nancy* (which is the joint computer science laboratory of the Universities of Nancy and the *Centre National de la Recherche Scientifique*) and the Lorraine research center of the *Institut National de la Recherche en Informatique et Automatique* for their support.

We are also grateful to Christiane Guyot and Véronique Poirel for their help with many tasks during the preparation of the workshop.

September 1992

The Organizers,

Michaël Rusinowitch  
Jean-Luc Rémy



## Table of Contents

July 8

## Typed Systems and Graph Rewriting

Invited Talk: Algebraic Semantics of Rewriting Terms and Types

*Karl Meinke* ..... 1

Context Rewriting

*Stefan Kahrs* ..... 21

Explicit Cyclic Substitutions

*Kristoffer Høgsbro Rose* ..... 36

Simple Type Inference for Term Graph Rewriting Systems

*Richard Banach* ..... 51

## Modularity and Termination

Consistency and Semantics of Equational Definitions over Predefined Algebras

*Valentin M. Antimirov and Anatoly I. Degtyarev* ..... 67

Completeness of Combinations of Conditional Constructor Systems

*Aart Middeldorp* ..... 82

Collapsed Tree Rewriting: Completeness, Confluence, and Modularity

*Detlef Plump* ..... 97

Combinations of Simplifying Conditional Term Rewriting Systems

*Enno Ohlebusch* ..... 113

Sufficient Conditions for Modular Termination of Conditional Term Rewriting Systems

*Bernhard Gramlich* ..... 128Termination of Combined (Rewrite and  $\lambda$ -Calculus) Systems*Carlos Loria-Saenz and Joachim Steinbach* ..... 143

Type Removal in Term Rewriting

*Hans Zanen* ..... 148

Termination of Term Rewriting by Interpretation

*Hans Zanen* ..... 155

Path Orderings for Termination of Associative-Commutative Rewriting

*Nachum Dershowitz and Subrata Mitra* ..... 168

July 9

**Proof Techniques and Extensions of Conditional Rewriting**

Invited Talk: Generic Induction Proofs

*Peter Padawitz* ..... 175

A Constructor-Based Approach for Positive/Negative-Conditional Equational Specifications

*Claus-Peter Wirth and Bernhard Gramlich* ..... 198

Semantics for Positive/Negative Conditional Rewrite Systems

*Klaus Becker* ..... 213

Inductive Theorem Proving by Consistency for First-Order Clauses

*Harald Ganzinger and Jürgen Stuber* ..... 226

Reduction Techniques for First-Order Reasoning

*François Bronsard and Uday S. Reddy* ..... 242**Theorem-Proving and Normal Form Languages**

Invited Talk: Conditional Term-Rewriting and First-Order Theorem Proving

*David Plaisted, Geoffrey Alexander, Heng Chu and Shie-Jue Lee* ..... 257

Decidability of Regularity and Related Properties of Ground Normal Form Languages

*Gregory Kucherov and Mohamed Tajine* ..... 272

Computing Linearizations Using Test-Sets

*Dieter Hofbauer and Maria Huber* ..... 287**Applications of Conditional Rewriting and New Formalisms**

Proving Group Isomorphism Theorems

*Hantao Zhang* ..... 302Semigroups Satisfying  $x^{m+n} = x^n$ *Nachum Dershowitz* ..... 307

Could Orders Be Captured By Term Rewriting Systems ?

*Sergei Vorobyov* ..... 315

A Categorical Formulation for Critical-Pair/Completion Procedures

*Karel Stokkermans* ..... 328

Trace Rewriting Systems

*Yabo Wang and David Lorge Parnas* ..... 343

A Calculus for Conditional Inductive Theorem Proving

*Ulrich Fraus* ..... 357



July 10

# **Contextual Rewriting and Constrained Rewriting**

Implementing Contextual Rewriting

*Hantao Zhang* ..... 363

Confluence of Terminating Membership Conditional TRS

*Junnosuke Yamada* ..... 378

Completeness and Confluence of Order-Sorted Term Rewriting

*Lars With* ..... 393

Completion for Constrained Term Rewriting Systems

*Charles Hoot* ..... 408

Generalized Partial Computation using Disunification to Solve Constraints

*Akihiko Takano* ..... 424

# **Applications to Logic Programming, Normalization Strategies and Unification**

Invited Talk: Decidability of Finiteness Properties

*Leszek Pacholski* ..... 429

Termination Proofs of Well-Moded Logic Programs Via Conditional Rewrite Systems

*Harald Ganzinger and Uwe Waldmann* ..... 430

Logic Programs with Polymorphic Types: A Condition for Static Type Checking

*Staffan Bonnier and Jonas Wallgren* ..... 438

Normalization by Leftmost Innermost Rewriting

*Sergio Antoy* ..... 448

A Strategy to Deal with Divergent Rewrite Systems

*Paola Inverardi and Monica Nesi* ..... 458

A New Approach to General E-Unification Based on Conditional Rewriting Systems

*Bertrand Delsart* ..... 468

An Optimal Narrowing Strategy for General Canonical Systems

*Alexander Bockmayr, Stefan Krischer and Andreas Werner* ..... 483

Set-Of-Support Strategy for Higher-Order Logic

*Wenchang Fang and Jung-Hong Kao* ..... 498



# ALGEBRAIC SEMANTICS OF REWRITING TERMS AND TYPES

Karl Meinke

Department of Mathematics and Computer Science,  
University College Swansea,  
Singleton Park,  
Swansea SA2 8PP,  
Great Britain.  
June 1992

**Abstract** We present a universal algebraic framework for rewriting terms and types over an arbitrary equational specification of types and typed combinators. Equational type specifications and their initial algebra semantics were introduced in Meinke [1991b]. For an arbitrary equational type specification  $(\varepsilon, E)$  we prove that the corresponding rewriting relation  $\xrightarrow{R(\varepsilon, E)^*}$  coincides with the provability relation  $(\varepsilon, E) \vdash$  for the equational calculus of terms and types. Using completeness results for this calculus we deduce that rewriting for ground terms and ground types coincides with calculation in the initial model  $I(\varepsilon, E)$  of the equational type specification.

## 0. INTRODUCTION

The basic theory of universal algebra in higher types was introduced in Poigné [1986], Möller [1987], Möller et al [1988] and Meinke [1990], [1992]. A higher type universal algebra is a many-sorted universal algebra in which the carrier sets have some type theoretic structure. Applications of the theory of higher type universal algebra to programming language definition and hardware verification are discussed in Meinke [1991a] and Meinke and Steggles [1992]. The basic theory of higher type universal algebra concerns only the propositional type constructions  $\times$  and  $\rightarrow$  for product and function spaces. In Meinke [1991b] a more general algebraic theory of abstract type constructions and typed combinator systems was introduced, and an equational specification theory for algebras of types and combinators was presented, including results on initial models and completeness of equation and type assignment calculi. Within this general framework, complex type constructions such as predicative and impredicative  $\Pi$  and  $\Sigma$  types and recursive types may be modelled algebraically. The universality of the theory allows for user defined type disciplines and the specification of abstract data types with a rich type structure.

An equational type specification is a pair  $Spec = ((\Gamma, \varepsilon), (\Sigma, E))$  consisting of a type signature  $\Gamma$ , and equational theory  $\varepsilon$  of types, a combinator signature  $\Sigma$  and an equational theory  $E$  of combinators. The model class  $Mod(Spec)$  forms a category which admits an initial algebra  $I(Spec)$  which we may take as the initial algebra semantics of  $Spec$ . From the viewpoint of computation it is natural to con-

sider term rewriting as a mechanism for computation in the initial model  $I(\text{Spec})$ . In this paper we introduce a theory of term rewriting for equational type specifications. An important new feature of this theory is that both rewriting of types and rewriting of terms are allowed. Given a combinator term  $t$  over a specification  $\text{Spec} = ((T, \varepsilon), (\Sigma, E))$  the oriented equations of  $\varepsilon$  allow us to rewrite type terms in  $t$  while the oriented equations of  $E$  allow us to rewrite combinator terms in  $t$ . Our main result is a Correspondence Theorem for rewriting which establishes that the rewrite relation  $\xrightarrow{R(\varepsilon, E)^*}$  on combinator terms using the oriented equations of  $\varepsilon$  and  $E$  coincides with the provability relation  $(\varepsilon, E) \vdash$  in the equational calculus of terms and types. By a completeness theorem for this calculus it follows that rewriting coincides with calculation in the initial model  $I(\text{Spec})$ .

The structure of this paper is as follows. In section 1 we present essential prerequisites from universal algebra and category theory. In section 2 we review basic definitions and results on the algebraic theory of types and combinators. In section 3 we introduce the fundamental definitions and concepts of rewriting for types and combinators. In section 4 we review the equational calculus of terms and types. Finally in section 5 we prove a Correspondence Theorem for rewriting and equational deduction. We have attempted to make the paper self contained. However the reader may wish to consult Meinke [1991b] and Meinke and Wagner [1992] for more detailed proofs and examples.

## 1. ALGEBRAIC AND CATEGORY THEORETIC PRELIMINARIES

The notation for many-sorted universal algebra that we use is taken from Meinke and Tucker [1992] where all of the basic technical definitions and results on algebra that we require may be found. Let us make our notation explicit.

An  $S$ -sorted signature  $\Sigma$  consists of a non-empty set  $S$ , the elements of which are *sorts*, and an  $S^* \times S$ -indexed family  $(\Sigma_{w,s} \mid w \in S^*, s \in S)$  of sets (where  $S^*$  denotes the set of all words over  $S$ ). For the empty word  $\lambda \in S^*$  and any sort  $s \in S$ , each element  $c \in \Sigma_{\lambda,s}$  is a *constant symbol* of sort  $s$ ; for each non-empty word  $w = s(1) \dots s(n) \in S^+$  and any sort  $s \in S$ , each element  $\sigma \in \Sigma_{w,s}$  is a *function symbol* of *domain type*  $w$ , *codomain type*  $s$  and *arity*  $n$ . Thus we can define  $\Sigma$  to be the pair  $(S, (\Sigma_{w,s} \mid w \in S^*, s \in S))$ .

Let  $\Sigma$  be an  $S$ -sorted signature. An  $S$ -sorted  $\Sigma$  algebra is an ordered pair  $(A, \Sigma^A)$ , consisting of an  $S$ -indexed family  $A = \{A_s \mid s \in S\}$  of carrier sets  $A_s$  and an  $S^* \times S$  indexed family  $\Sigma^A = (\Sigma_{w,s}^A \mid w \in S^*, s \in S)$  of sets of constants and algebraic operations. For each sort  $s \in S$ ,  $\Sigma_{\lambda,s}^A = \{c_A \mid c \in \Sigma_{\lambda,s}\}$ , where  $c_A \in A_s$  is a constant that interprets  $c$ . For each  $w = s(1) \dots s(n) \in S^+$  and each sort  $s \in S$ ,  $\Sigma_{w,s}^A = \{f_A \mid f \in \Sigma_{w,s}\}$ , where  $f_A : A^w \rightarrow A_s$  is an operation with domain  $A^w = A_{s(1)} \times \dots \times A_{s(n)}$  and codomain  $A_s$  which interprets  $f$ . As usual, we let  $A$  denote both a  $\Sigma$  algebra and its  $S$ -indexed family of carrier sets.

We require one essential construction from category theory, the cofibration construction, which we use to structure the model class of an equational type specification.

**1.1. Definition.** Let  $\mathcal{C}$  be a category and  $\text{Cat}$  be a category of categories.