Agostino Dovier   Enrico Pontelli (Eds.)

A 25-Year Perspective on Logic Programming

Achievements of the Italian Association for Logic Programming, GULP

Springer

Volume Editors

Agostino Dovier
Università di Udine
Dip. di Matematica e Informatica
Via delle Scienze 206, 33100 Udine, Italy
E-mail: dovier@dimi.uniud.it

Enrico Pontelli
New Mexico State University
Department of Computer Science
P.O. Box 30001, MSC CS, Las Cruces, NM 88003, USA
E-mail: epontell@cs.nmsu.edu

Library of Congress Control Number: 2010929758

CR Subject Classification (1998): D.1.6, F.4.1, F.3, F.4, I.2.3, I.2

LNCS Sublibrary: SL 2 – Programming and Software Engineering

ISSN 0302-9743
ISBN-10 3-642-14308-3 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-14308-3 Springer Berlin Heidelberg New York

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper 06/3180

Foreword

This book celebrates the 25th anniversary of GULP—the Italian Association for Logic Programming. Authored by Italian researchers at the leading edge of their fields, it presents an up-to-date survey of a broad collection of topics in logic programming, making it a useful reference for both researchers and students.

During its 25-year existence, GULP has organised a wide range of national and international activities, including both conferences and summer schools. It has been especially active in supporting and encouraging young researchers, by providing scholarships for GULP events and awarding distinguished dissertations.

We in the international logic programming community look upon GULP with a combination of envy, admiration and gratitude. We are pleased to attend its conferences and summer schools, where we can learn about scientific advances, catch up with old friends and meet young students. It is an honour for me to acknowledge our appreciation to GULP for its outstanding contributions to our field and to express our best wishes for its continuing prosperity in the future.

March 2010

Robert Kowalski
Imperial College London

Preface

On June 18, 1985, a group of pioneering researchers, including representatives from industry, national research labs, and academia, attended the constituent assembly of the *Group of researchers and Users of Logic Programming (GULP)* association. That was the starting point of a long adventure in science, that we are still experiencing 25 years later.[1] This volume celebrates this important event.

What about the editors of this volume? On that date, one of us was completing his secondary school studies, the other his mandatory military service. But only one year later, the two of us met in the introductory class of the computer science program at the University of Udine, and that was the beginning of a logic programming experience that spans the majority of our careers in academia.

With excitement, humbleness, and profound honor, and after a formal nomination as editors from the GULP assembly (during the CILC'08 meeting), we embarked upon the mission of developing this volume. The purpose of this effort is to celebrate an important milestone in the world of logic programming, the 25th anniversary of GULP. GULP is the oldest formal logic programming association (the international Association for Logic Programming, for example, was established in 1986), and, over the last 25 years, GULP has promoted research activities whose results and directions are at the core of the whole logic programming world.

Summarizing 25 years of research on logic programming in a single volume is a daunting and perhaps impossible task. We were forced to make difficult decisions in selecting the topical areas of logic programming to be analyzed in the various chapters; this task was particularly complex, due to the diversity of the research initiatives in logic programming that have developed over the years in Italy. In the end, we decided to concentrate on those areas that, historically, have been at the core of logic programming research in Italy; we wish to apologize to those researchers whose areas have been excluded from this volume.

Each chapter of this volume has been co-authored by several researchers. In particular, we have attempted to create a balance between historical developments and current state of the art by pairing, in each chapter, younger researchers with more established leaders in the field (but we will not explicitly identify who is who...). The response from the logic programming community to our invitations to author chapters was overwhelmingly positive; 35 researchers enthusiastically accepted to participate in this initiative. The effort resulted in 14 chapters, each providing a fresh and useful overview of a different area of logic programming. Thanks to the hard work of the authors, each chapter represents

[1] As a remark, Italy was the reigning football world champion in 1985, as they are today.

a great analysis of a specific research field, providing both historical perspectives as well as a precise discussion of the current state of the art. The authors also provide an interesting view of how the contributions of Italian researchers have shaped the field of logic programming over the years.

This volume represents the logical continuation of the volume edited by Maria I. Sessa in 1995, celebrating the 10th anniversary of GULP. While several of the chapters address analogous topics (i.e., theoretical foundations, program transformations, non-monotonic reasoning, constraint logic programming, concurrent logic programming, program verification), other chapters have either been replaced, due to the lack of intense research (e.g., metalogic programming) or expanded into more detailed chapters, to reflect the changes in directions within the field. For instance, the stable models chapter has evolved into the more mature answer set programming chapter, and the chapter on applications to software engineering has evolved into a wider scope applications chapter. We added other new chapters that represent very active fields, like databases and web, agents and multi-agent systems, two chapters on extensions of logic programming (functional logic programming and higher order programming), and a seminal paper on research in automated theorem proving.

The organization of this volume follows a structure that highlights what we perceived to be the historical dependencies among the various areas. These dependencies are summarized in the graph in Fig.1.

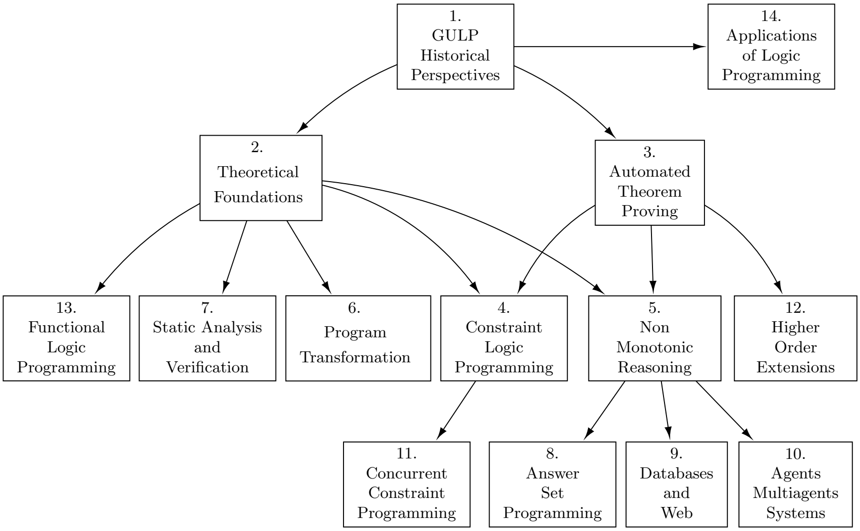
1. GULP Historical Perspectives
14. Applications of Logic Programming
2. Theoretical Foundations
3. Automated Theorem Proving
13. Functional Logic Programming
7. Static Analysis and Verification
6. Program Transformation
4. Constraint Logic Programming
5. Non Monotonic Reasoning
12. Higher Order Extensions
11. Concurrent Constraint Programming
8. Answer Set Programming
9. Databases and Web
10. Agents Multiagents Systems

Fig. 1. The GULP tree and book structure

The volume opens with an historical perspective of the first 25 years of the association, written by the current GULP president, Gianfranco Rossi. Gianfranco has witnessed the evolution of GULP since its inception, and he reports a detailed history of the GULP association in his chapter. He also provides a personal view of the directions to be followed by GULP to avoid past mistakes and expand the success of logic programming in Italy, especially in terms of impact on the industrial world.

The roots of logic programming research in Italy can be traced back to the research efforts in the areas of automated theorem proving and theoretical computer science (e.g., programming languages semantics). Indeed, looking back at the areas of the various contributions reported in the first volumes of the proceedings of GULP, one can note how semantical foundations and theorem proving are at the backbone of many of the reported contributions. The editors decided to open the research overview of this volume with two chapters dedicated to these two foundational areas. These are identified at the top of the graph and reported in Chaps. 2 and 3.

A reason for the great initial success of logic programming was undoubtedly the elegance of its semantics. The set of ground atoms that can be inferred from a program P, using SLD resolution (operational semantics), can be proved to be equivalent to the minimum Herbrand model of P (logical semantics) and, in turn, equivalent to the least fixpoint of a continuous operator dependent on P (declarative semantics). Chapter 2, developed by Annalisa Bossi and Chiara Meo, gives an overview of the original roots of research in the theoretical foundations of logic programming. Work in this area was spearheaded by the group of Giorgio Levi (first president of GULP) and his colleagues in Pisa and Torino, and was instrumental in placing Italian logic programming research on the international map.

The clear ties of logic programming, since its inception, with mathematical logic and theorem proving, have provided ample opportunities for research in automated theorem, laying the foundations to the growth of logic programming. In particular, the completeness proof of SLD resolution as an inference method for first-order theories given as sets of definite clauses, and the Turing completeness of this fragment of first-order logic, are probably the two fundamental contributions of automated reasoning that allowed Kowalski to write the seminal contribution *Predicate Logic as Programming Language*. Chapter 3, developed by Andrea Formisano and Eugenio G. Omodeo revisits the original work in the area of theorem proving, highlighting the ties to logic programming.

A combination of the studies in theorem proving (based on theory-based resolution), and on a generalization of the semantics of logic programming (to the case of non-Herbrand domains) offers the foundations on which the area of constraint logic programming developed. Constraint logic programming enabled the first step towards enhancing the declarative nature of logic programming, often lost in the use of Prolog, and at the same time gaining a level of efficiency required by industrial-strength applications. The combination of declarativeness of logic programming and of efficiency of solvers in suitable theories allows one to solve efficiently real-life problems without the need of writing low-level code. A nice survey of this area is presented in Chap. 4, developed by Marco Gavanelli and Francesca Rossi.

The field of constraint logic programming, thanks also to the intense work conducted in the context of the Fifth Generation Computer Systems project, has evolved to create a revolutionary paradigm that combines logic programming, constraint programming, and concurrency; the challenging issues of concurrent constraint programming are reviewed in Chap. 11, written by Maurizio Gabbrielli, Catuscia Palamidessi, and Frank Valencia.

The original developments on the semantics of logic programming quickly moved towards the investigation of variants of the logic programming paradigm where the traditional elegant properties of logic programming semantics (e.g., uniqueness of the least Herbrand model) fail. This is particularly true in the case of extensions of logic programming developed to handle non-monotonicity, which are vital to the task of knowledge representation and commonsense reasoning. This volume dedicates two related chapters to the investigation of these aspects. The first is Chap. 5, by Laura Giordano and Francesca Toni, which explores the role of logic programming in the area of non-monotonic reasoning and knowledge representation. While techniques for non-monotonic reasoning moved originally in different directions, in recent years the field has witnessed a convergence of effort towards the use of stable model semantics proposed by Gelfond and Lifschitz. The embedding of stable model semantics in a concrete programming paradigm, originated from the concurrent work of Marek, Truszczyński, and Niemelä, led to what is now known as answer set programming. The field is now at the core of logic programming, thanks also to the development of highly competitive solvers. This area is surveyed in Chap. 8, by Piero Bonatti, Francesco Calimeri, Nicola Leone, and Francesco Ricca.

The field of logic-based intelligent agents also traces back its foundations to the area of logic programming and non-monotonic reasoning; this field has matured over the years and Italian research in this domain has gained reputation within the larger umbrella of the international artificial intelligence community. Chapter 10, by Matteo Baldoni, Andrea Omicini, Cristina Baroglio, Viviana Mascardi, and Paolo Torroni, provides an exciting review of work on logic-based methodologies for intelligent agents and multi-agent systems.

The foundations of non-monotonic reasoning have also been deeply tied with two other areas that have witnessed intense research with the support of GULP—databases and intelligent agents. The field of databases has been present at GULP since its beginning; work in the area of deductive databases has offered significant contributions not only to the database community, but also to the development of the foundations of modern logic programming (e.g., the work on DATALOG$^{\neg}$ contributed to answer set programming). Chapter 9, by Francesca Lisi and Sergio Greco, provides an overview of logic programming work in the area of databases and the web.

The work on semantics of logic programming has traditionally provided the foundations for enhancing the understanding of programs; this is essential in order to develop techniques for program transformation, to gain efficiency, and program verification, to guarantee correctness. These two aspects are analyzed in Chap. 6, by Alberto Pettorossi, Maurizio Proietti, and Valerio Senni—which covers the area of program transformations—and Chap. 7, by Giogio Delzanno, Roberto Giacobazzi, and Francesco Ranzato—which provides an overview of research in the areas of static analysis, abstract interpretation, and program verification.

From an automated reasoning point of view, traditional logic programming is just one particular instance of automated deduction with a given first-order language (with definite clauses) and with a particular proof engine (SLD resolution). One can enlarge this schema in several directions, for instance working on the proof structure (e.g., uniform proofs) or admitting higher-order predicates. This area is surveyed in Chap. 12, by Alberto Momigliano and Mario Ornaghi.

The overall area of declarative approaches to programming includes several other paradigms beyond logic programming. In particular, functional programming provides a number of features that are absent in logic programming and that are convenient in many programming tasks. The area of functional logic programming investigates attempts to combine logic programming and functional programming within a single paradigm, which provides the benefits of both logic programming (e.g., search, non-determinism) and functional programming (e.g., higher order constructs). The role of Italian research in functional logic programming has been predominant since its inception, and it is summarized in Chap. 13, by Maria Alpuente, Demis Ballis, and Moreno Falaschi.

Last but not least, Chap. 14, by Alessandro Dal Palù and Paolo Torroni, reviews in detail the main applications of logic programming developed in Italy and/or by Italian researchers in the last 25 years. This chapter represents an ideal closure to this volume—there is wide agreement that the continued success of the field of logic programming vitally depends on investigating the use of logic programming technology to solve concrete real-world problems. The chapter nicely illustrates successful work done and potential directions for future developments.

In closing this introduction, the editors would like to take the opportunity to extend their heartfelt thanks to a number of people who made this effort possible:

- The authors of the chapters, who have tirelessly worked on creating comprehensive overviews of research directions that have developed over 25 years of logic programming (most of them also acted as reviewers of other chapters):

  Maria Alpuente
  Matteo Baldoni
  Demis Ballis
  Cristina Baroglio
  Piero Bonatti
  Annalisa Bossi
  Francesco Calimeri
  Alessandro Dal Palù
  Giogio Delzanno
  Moreno Falaschi
  Andrea Formisano
  Maurizio Gabbrielli
  Marco Gavanelli
  Roberto Giacobazzi
  Laura Giordano
  Sergio Greco
  Nicola Leone
  Francesca Lisi
  Viviana Mascardi
  Chiara Meo
  Alberto Momigliano
  Andrea Omicini
  Eugenio G. Omodeo
  Mario Ornaghi
  Catuscia Palamidessi
  Alberto Pettorossi
  Maurizio Proietti

Francesco Ranzato
Francesco Ricca
Francesca Rossi
Gianfranco Rossi
Valerio Senni
Francesca Toni
Paolo Torroni
Frank Valencia

- The additional reviewers who provided insightful comments on the various chapters under very strict time constraints:

  Sergio Antoy
  Nicoletta Cocco
  Sandro Etalle
  Camillo Fiorentini
  Roberta Gori
  Evelina Lamma
  Michela Milano
  Alessandra Mileo
  Angelo Montanari
  Carla Piazza
  Germán Vidal
  Alicia Villanueva García

- The Italian logic programming community and GULP, who have created in Italy a nurturing environment for countless young researchers to embrace and appreciate the beauty of logic programming. Personally, the editors would not have been coordinating this volume without the friendship and advice of Gianfranco Rossi and Eugenio Omodeo, who originally introduced us to logic programming.
- Bob Kowalski for his foreword and for what has done and is still doing for the logic programming community.

Finally, we would like to send a “thank you” to all those we love in this and other worlds.

March 2010

Agostino Dovier
Enrico Pontelli

Table of Contents

Twenty-Five Years of Logic Programming in Italy