TLP **11** (2–3): 131–133, 2011. © Cambridge University Press 2011
doi:10.1017/S1471068411000020

Introduction to the 24th international conference on logic programming special issue

MARIA GARCIA DE LA BANDA
Faculty of Information Technology Monash University, Melbourne, Victoria, Australia
(*e-mail:* Maria.GarciadelaBanda@monash.edu)

ENRICO PONTELLI
Department of Computer Science New Mexico State University, Las Cruces, New Mexico, USA
(*e-mail:* epontell@cs.nmsu.edu)

submitted 13 June 2010; revised 13 June 2010; accepted 13 June 2010

1 Introduction

The ICLP series of conferences provides a technical forum for presenting and disseminating innovative research in the field of logic programming. The 24th International Conference on Logic Programming took place from December 9–13, 2008 in the city of Udine, Italy. The conference attracted 177 submissions and featured a high-quality program focused on the foundations, developments, and applications of logic programming. Of particular significance was the special session celebrating the 20th anniversary of the seminal paper on the stable model semantics.

The program of the conference provided contributions that were expected to have a long-term impact on the future developments in the field of logic programming. After the conference was held, the Program Committee chairs, with the help of the program committee and the reviewers, embarked on the task of identifying the most significant contributions and inviting their authors to submit extended versions for revision. The outcome of this effort is a collection of papers that includes nine extended contributions of the papers originally published in the conference proceedings (LNCS 5366, Springer Verlag, Berlin, 2008).

This special issue is organized in three parts. The first part of the special issue is a compendium of novel contributions in the area of *Answer Set Programming* (ASP); the second part includes contributions in the more general areas of *theory and implementation of logic programming*; and the third part includes contributions that explore novel *applications* of logic programming technology to real-world domains.

The Answer Set Programming is a fully declarative programming paradigm that combines features from logic programming, constraint programming, and

https://doi.org/10.1017/S1471068411000020 Published online by Cambridge University Press

nonmonotonic reasoning. The ASP paradigm has been the focus of a large amount of research interest and development during the last decade, since the publication of the seminal papers on ASP from Marek, Truszczyński, and Niemelä. Its success has been fueled by the following two key aspects:

(1) The availability of well-engineered and efficient implementations, combining state-of-the-art techniques drawn from a variety of domains and capable of handling large applications.
(2) The development of provably correct knowledge representation modules relevant to different application domains, which enable the use of ASP as a paradigm of choice to support research in such domains.

The papers selected for the first part of this special issue significantly contribute to these two aspects of research.

The first contribution by Lierler offers a general framework for the description and analysis of answer set solvers. In particular, the framework enables the design of new algorithms for implementing the ASP systems and provides a formal model for the comparison of alternative approaches to computing answer sets.

The next two contributions focus on theoretical and practical aspects of ASP. The contribution by Fink provides a general, elegant, and uniform theoretic framework for the study of the many different notions of equivalence that can be characterized under the ASP semantics. Importantly, the results do not have the common finiteness restrictions for non-ground settings and, thus, apply to the practical ASP systems that handle finite non-ground programs over infinite domains. The contribution by Cabalar presents a new language for explicitly dealing with functions in ASP and discusses several modeling issues that are not easily solvable within other ASP modeling paradigms that can also deal with functions. In particular, the new language allows a more natural, compact, and readable representation of functions than the relational approach.

The second part of this special issue includes more general contributions in the areas of theory and implementation of logic programming.

The first two contributions focus on probabilistic extensions of logic programming. The contribution by Kimmig, Demoen, De Raedt, Costa, and Rocha provides exact and approximate inference algorithms that allow the efficient execution of queries in ProbLog, a recent probabilistic extension of Prolog motivated by the mining of large biological networks. The authors discuss the integration of these algorithms into the YAP Prolog system, and evaluate their performance in the context of large networks of biological entities. The contribution by Cohen, Simmons, and Smith provides a natural extension to weighted logic programming, in which weighted logic programs (each performing probabilistic inference over a structure) can be combined to perform inference over two or more structures. The paper shows how this extension provides a general setting for dynamic programming algorithms that process two or more conceptually distinct objects, and how many important dynamic programming algorithms can be derived from it.

https://doi.org/10.1017/S1471068411000020 Published online by Cambridge University Press

The last contribution in this part of the special issue, by Zhang and Yap, focuses on functions in the constraint logic programming setting. In particular, it provides a new method to deal with the functional and bi-functional constraints that commonly arise in constraint logic programming systems. The method, which is based on variable substitution rather than the traditional arc- or path-consistency, is conceptually simpler than previous approaches and can significantly improve the performance of a general constraint solver.

The remaining papers of this special issue present innovative uses of logic programming technology in three different application domains. The contribution of Gebser, Schaub, Thiele, and Veber continues an established line of research dedicated to the use of ASP to model and analyze biological networks, and focuses on identifying inconsistencies in these networks. The contribution of Mileo, Merico, and Bisiani introduces a very innovative use of ASP as the reasoning backend for a sensor-based system for aiding elderly in an independent living environment. The contribution by Boenn, Brain, De Vos, and Ffitch investigates the use of ASP to compose melodic, harmonic, and rhythmic music, diagnose errors in human compositions, and serve as a computer-aided composition tool.

In closing this preface, the editors would like to acknowledge the help and support provided by the ICLP 2008 Program Committee, all the external referees, and the Association for Logic Programming.

https://doi.org/10.1017/S1471068411000020 Published online by Cambridge University Press