Logic Programming: Where is it going?
State of the Art and Future Perspectives

Stefania Costantini

In this note we make an attempt to provide an overview of the present state of the field of Logic Programming (LP), and the possible future directions of research, developments and applications. We will list some among the subjects that we deem to be relevant without pretending to relate in this short note on all topics concerning LP. We mention some papers just for the sake of the example, without implying that they are the best or the sole papers treating certain topics.

Technology

The Logic Programming paradigm, meaning both Prolog-based Logic Programming and Answer Set Programming (ASP) is widely adopted in a number of practical and industrial applications. Thus, for making the paradigm efficient and more usable, work is under way for improving efficiency of inference engines, especially via parallel execution. We can mention:

- Several papers about improving ASP solvers (impossible to make a choice about so many of them, appearing in the main Conferences and journals)
- Parallel Execution of the ASP Computation - an Investigation on GPUs *by Andrea Formisano, Flavio Vella, Agostino Dovier and Enrico Pontelli, ICLP 2015*
- On the Implementation of an Or-Parallel Prolog System for Clusters of Multicores, *by João Santos and Ricardo Rocha*. ICLP 2016
- Tabling with Sound Answer Subsumption, *by Alexander Vandenbroucke, Maciej Piróg, Benoit Desouter and Tom Schrijvers*. ICLP 2016
- Lock-free atom garbage collection for multi-threaded Prolog, *by Jan Wielemaker and Keri Harris.* ICLP 2016

Programming

At the same time new programming methodologies are being studied, among which:

- Goal-Directed Answer Set Programming
- Extensions to Constraint Logic Programming

LP as a tool for studying other programming paradigm

Remarkably enough, there has been recent work for exploiting LP for program analysis, transformation and debugging of LP itself and other programming paradigms.

- Proving Correctness of Imperative Programs by Linearizing Constrained Horn Clauses, *by Emanuele De Angelis, Fabio Fioravanti, Alberto Pettorossi and Maurizio Proietti, ICLP 2015*
- Horn Clauses as an Intermediate Representation for Program Analysis and Transformation

Graeme Gange, Jorge A Navas, Peter Schachte, Harald Sondergaard and Peter J. Stuckey ICLP 2015
- Debugging ASP using ILP, b*y Tingting Li, Marina De Vos, Julian Padget, Ken Satoh and Tina Balke, ICLP 2015*
- Precise Complexity Guarantees for Pointer Analysis via Datalog with Extensions, *by K. Tuncay Tekle and Yanhong A. Liu.* ICLP 2016

Extensions

There are many intertesting extensions to the basic LP paradigm that are raising growing attention and expectations, among which the following.

- Probabilistic LP
- Paraconsistent LP
- Causal Reasoning in LP
- Datalog+- and abductive LP for ontological representation and resoning
- Agent-Oriented LP

20 years Test-of-Time Award 2017

In fact, it is not by chance that the following paper has been awarded at ICLP 2017 as a paper successfully passing the test of time, since as mentioned Probabilistic LP is subject of renewed attention.

- `Hybrid Probabilistic Programs', accepted in ICLP 1997, Leuven, Belgium, by Alex Dekhtyar and V.S. Subrahmanian, awarded 2017

One of the authors, namely V.S. Subrahmanian, was among the proposers (in 1992) also of Paraconsistent LP.

Theory

There are many papers on the theory of LP, we mention the following one (best paper at the last ICLP) because it testifies a renewed interest also in Higher-Order Logic Programming, a field that enjoyed a great popularity (together with meta-programming) in the 1990's and was then overlooked for some time.

- The Intricacies of 3-Valued Extensional Semantics for Higher-Order Logic Programs, by Panos Rondogiannis and Ioanna Symeonidou, ICLP 2017 Best Paper.
- Logic Programming and Bisimulation *Agostino Dovier, ICLP 2015.*

Advanced Applications

We cannot overlook the applications of LP to in the fields of Logical Agents and Cognitive Robotics. Significantly enough, Robert A. Kowalski and Keith Clark, who are among the "founding fathers" of LP, are now very active in these fields.

Among the very many other successful applications of LP we mention some papers that should provide an idea about the wide range of application fields, tha range from Bioinformatics to Hybrid Knowledge Bases to Business Processes verification to eHealth, etc.

- The role of SAT, CP, and Logic Programming in Computational Biology, *by Agostino Dovier*, ICLP/SAT/CP 2017
- A Physician Advisory System for Chronic Heart Failure Management Based on Knowledge Patterns, *by Zhuo Chen, Kyle Marple, Elmer Salazar, Gopal Gupta and Lakshman Tamil*. ICLP 2016
- QuLog: A flexibly typed logic based language with function and action rules, by *Keith Clark and Peter Robinson, ICLP 2015*
- Iterated Fixpoint Well-founded Semantics for Hybrid Knowledge Bases*, by Marco Alberti, Evelina Lamma, Fabrizio Riguzzi and Riccardo Zese*. CILC 2017
- Verifying Controllability of Time-Aware Business Processes*, by* Emanuele De Angelis, Fabio Fioravanti, Maria Chiara Meo, Alberto Pettorossi and Maurizio Proietti*.* RuleML+RR 2017
- Improved Answer-Set Programming Encodings for Abstract Argumentation *by Sarah Alice Gaggl, Norbert Manthey, Alessandro Ronca, Johannes Peter Wallner and Stefan Woltran.* TPLP 15, 2015
- Planning as Tabled Logic Programming, *by Neng-Fa Zhou, Roman Bartak and Agostino Dovier, ICLP 2015*

Future Perspectives

Among the many open promising perspectives of LP two lines of research and applications seem to us to be not only promising, but also somehow ineludible. We mention the following two recent interventions.

- The best of both worlds: Machine learning meets logical reasoning, *by Holger H. Hoos*. Invited talk at ICLP 2017
- Programming Machine Ethics, by Luís Moniz Pereira, Ari Saptawijaya, Springer 2016 (plus many related papers by L.M. Pereira and various co-authors)

The need to control Machine Learning (ML) processes by means some kind of logic, and LP is there a perfect candidate, arises related to several aspects, in particular concerning the specification of a logical model of both the data and the learning results. This because, e.g.,

- It is perfectly possible to learn mistakes;
- There is a need for explainability and accountability of learning results, so as to attenuate the problem of the learning process being “Black Box”; explainability is aimed to make AI more acceptable to human users, and also for passing the “competences” learnt by the system to human experts; accountability is an even wider concept, where a human responsible of a process which is demanded at least partly to an AI system must then be able to motivate the decisions taken according to system’s results.
- When learning the functioning of some external system, a logical model of such system allows diagnosis and repair to be performed, that would be impossible without.

- In (very) future perspective: we would not like to leave our descendants hostages to a super-intelligence developed by their ancestors, maybe purely benign, but impossible for them to control and modify; we would like to preserve human competences annd, ultimately, “humanity”, where our descendants should be the masters of their own destiny.

Machine Ethics (ME) has become a pervasive and compelling need in every field of Computer Science but especially in relation to AI in general, and to Agents and ML in particular. In fact, when the life and welfare of humans will be in the custody of autonomous machines, principles must be established about how these machines should behave both in everyday and in exceptional situations, and techniques should be developed to put such principles into play. Logic Programming is the natural paradigm for doing so due to its twofold nature of representation language and of reasoning device. Definition, implementation, verification (not only a-priori but also at run-time), certification and assurance of ethical properties of systems are very important topics. LP and its agent-oriented and higher-order extensions can play an important role here.