

Answer Set Solving in Practice: Advanced techniques

A Tutorial at IJCAI'15

Martin Gebser, Roland Kaminski, Javier Romero and Torsten Schaub
University of Potsdam, Germany

<http://www.cs.uni-potsdam.de/~torsten/ijcai15tutorial/summary>

Short Description

This full-day tutorial presents a practical introduction to Answer Set Programming (ASP), aiming at using ASP languages and systems for solving application problems. Starting from the essential formal foundations, it introduces ASP's modeling language and methodology, grounding and solving technology, and finally details control techniques needed for embedding ASP in complex software environments.

What makes this tutorial different is its focus on putting ASP at work. This comprises a good understanding of ASP technology and systems as well as basic skills in ASP's declarative modeling, imperative control (via dedicated APIs), as well as their interplay in addressing complex reasoning processes.

Description

The tutorial aims at acquainting the participant with ASP's modeling and solving methodology, enabling her/him to conduct independent problem solving using ASP systems (as standalone systems as well as through dedicated APIs). To this end, the tutorial starts with an introduction to the essential formal concepts of ASP, needed for understanding its semantics and solving technology. In fact, ASP solving rests on two major components: A grounder turning specifications in ASP's modeling language into propositional logic programs and a solver computing a requested number of answer sets of the program. Building on the formal concepts, we provide a characterization of ASP's inference schemes and map them into algorithms relying on advanced Boolean Constraint technology. We illustrate this technology through the award-winning ASP solver `clasp` 3. We then detail ASP's grounding techniques (along with the underlying deductive database concepts) and describe the major algorithms used in the ASP grounder `gringo` 4. This is accompanied with an introduction to the new ASP language standard. The remainder of the tutorial is dedicated to using ASP in conjunction with Python for modeling complex reasoning scenarios. This involves an introduction to the API of `clingo` 4, an ASP system extending `clasp` and `gringo` with control capacities expressible in Python (and Lua). We illustrate this by developing sample gaming agent programs and describe more sophisticated usages in planning, robotics and preference handling. All involved ASP systems are freely available from <http://potassco.sourceforge.net>.

Outline of the Tutorial

- Motivation
- Introduction
- Basic modeling
- Grounding (ASP language standard)
- Solving (multi-shot solving)

- Systems (clingo 4)
- Advanced modeling (Python integration and control)
- Preference handling

The keywords in parentheses aim at underpinning the “Advanced techniques” referred to in the tutorial’s title.

Roughly, the first part of the tutorial provides the theoretical and practical foundations necessary for beginners with ASP, while the second part deals with advanced topics, such as the ones indicated above.

Potential Target Audience

- Graduate students in AI
- Teachers of AI
- Researchers and industrials addressing combinatorial (optimization) problems

Interest to IJCAI Audience

ASP provides a declarative tool for modeling various combinatorial (optimization) problems typical to Knowledge Representation and Reasoning in particular and AI in general. The unique pairing of declarativeness and performance allows for concentrating on an actual problem, rather than a smart way of implementing it. The ASP approach is not only highly suitable for the practitioner solving an AI problem at hand but also for disseminating many basic AI techniques through teaching their (executable) formalization in ASP.

Material

- Lecture slides
- Lecture examples
- Potassco, the Potsdam Answer Set Solving Collection
 - ASP Teaching material
 - ASP Book
 - User’s manual
 - Overview article
- This file as PDF or HTML

Contact

Drop us an email at {gebser, kaminski, javier, torsten}@cs.uni-potsdam.de for any questions.

References

- [1] C. Baral. *Knowledge Representation, Reasoning and Declarative Problem Solving*. Cambridge University Press, 2003.
- [2] G. Brewka, T. Eiter, and M. Truszczyński. Answer set programming at a glance. *Communications of the ACM*, 54(12):92–103, 2011.
- [3] T. Eiter, G. Ianni, and T. Krennwallner. Answer Set Programming: A Primer. In S. Tessaris, E. Franconi, T. Eiter, C. Gutierrez, S. Handschuh, M. Rousset, and R. Schmidt, editors, *Fifth International Reasoning Web Summer School (RW'09)*, volume 5689 of *Lecture Notes in Computer Science*, pages 40–110. Springer-Verlag, 2009.
- [4] M. Gebser, R. Kaminski, B. Kaufmann, M. Ostrowski, T. Schaub, and M. Schneider. Potassco: The Potsdam answer set solving collection. *AI Communications*, 24(2):107–124, 2011.
- [5] M. Gebser, R. Kaminski, B. Kaufmann, and T. Schaub. *Answer Set Solving in Practice*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan and Claypool Publishers, 2012.
- [6] M. Gebser, B. Kaufmann, and T. Schaub. Conflict-driven answer set solving: From theory to practice. *Artificial Intelligence*, 187-188:52–89, 2012.
- [7] M. Gelfond. Answer sets. In V. Lifschitz, F. van Harmelen, and B. Porter, editors, *Handbook of Knowledge Representation*, chapter 7, pages 285–316. Elsevier Science, 2008.
- [8] V. Lifschitz. Foundations of logic programming. In G. Brewka, editor, *Principles of Knowledge Representation*, pages 69–127. CSLI Publications, 1996.