Technical Communications of the International Conference on Logic Programming, 2010 (Edinburgh), pp. 265–269
http://www.floc-conference.org/ICLP-home.html

TOWARDS A GENERAL ARGUMENTATION SYSTEM BASED ON ANSWER-SET PROGRAMMING

SARAH ALICE GAGGL

Institute of Information Systems 184/2,
Vienna University of Technology,
Favoritenstrasse 9-11,
A-1040 Vienna,
Austria
E-mail address: gaggl@dbai.tuwien.ac.at
URL: http://www.dbai.tuwien.ac.at/staff/gaggl/

Abstract. Within the last years, especially since the work proposed by Dung in 1995, argumentation has emerged as a central issue in Artificial Intelligence. With the so called argumentation frameworks (AFs) it is possible to represent statements (arguments) together with a binary attack relation between them. The conflicts between the statements are solved on a semantical level by selecting acceptable sets of arguments. An increasing amount of data requires an automated computation of such solutions. Logic Programming in particular Answer-Set Programming (ASP) turned out to be adequate to solve problems associated to such AFs. In this work we use ASP to design a sophisticated system for the evaluation of several types of argumentation frameworks.

Introduction and Problem Description

Argumentation systems provide a formal way of dealing with conflicting knowledge. In particular argumentation frameworks (AFs) introduced by Dung [11] in 1995 are used to represent statements together with a relation denoting rebuttals between them, where the internal structure of the statements is of no interest for the evaluation of the framework. Several semantics have been defined to solve the inherent conflicts between the statements by selecting acceptable subsets of them. The most recognized of them are the stable, preferred and grounded semantics. The following example illustrates the definition and graphical representation of an AF.

Example 1. Let the AF $F = (A, R)$ be defined as follows, $A = \{a, b, c, d\}$ is the set of *arguments*, and $R = \{(a, b), (b, c), (b, d), (c, d)\}$ is the *attack relation* between the arguments. Let now $S = \{a, c\}$ be a set of *acceptable* arguments (also called a solution of F wrt a given semantics). Such an AF can be represented as a directed graph as shown in Figure 1.

1998 ACM Subject Classification: D.1.6 Logic Programming, I.2.4 Knowledge Representation Formalisms and Methods.
Key words and phrases: Argumentation, Implementation, Answer-Set Programming.
This work was supported by the Vienna Science and Technology Fund (WWTF) under grant ICT08-028.

Technical Communications of the 26th International Conference on Logic Programming, Edinburgh, July, 2010
Editors: Manuel Hermenegildo, Torsten Schaub
LIPIcs - Leibniz International Proceedings in Informatics. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Germany
Digital Object Identifier: 10.4230/LIPIcs.ICLP.2010.265

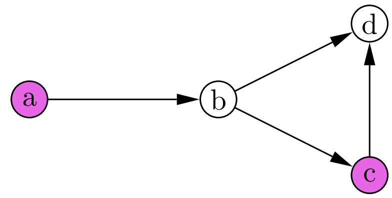

Figure 1: The argumentation framework F from Example 1.

Lets have a closer look why the set S represents a solution for our framework. The argument a is not attacked by any argument, hence it can be clearly viewed as acceptable. The argument b is only attacked by a, and as we accepted a, we can not also accept b, because two arguments attacking each other would not lead to a meaningfull solution. Whereas, we can say the argument c is defended by a against the attack from b and thus can be included into the solution. Finally, the argument d is also defended by a against the attack from b, but it is still attacked by c which already is part of the solution. Hence, d cannot be contained in S.

Within the last years, AFs became a main research area in Artificial Intelligence (AI). Recently two textbooks on argumentation [8, 22] and a special issue on argumentation in AI [7] have been published. Furthermore the Conference on Computational Models of Argument (COMMA) is held every second year.

The increasing interest in this topic resulted in the fact that Dung's approach has been extended and generalized continuously according to specific application scenarios like Multi-Agent Systems and Law Research. On the one hand, various semantics like semi-stable [9] or ideal semantics [12] have been introduced to adjust to the specific scenarios, on the other hand, the framework in itself has been adapted by modifying the notion of rebuttal [1], introducing new relations between the statements [2] or augmenting them with priorities [6].

For small instances it is quite easy to evaluate the frameworks under different semantics, but an increasing amount of data requires a sophisticated system for the evaluation. Argumentation problems are in general intractable, for instance deciding if an argument is contained in some preferred extensions is known to be *NP*-complete. Therefore, developing dedicated algorithms for the different reasoning problems is non-trivial. A promising way to implement such systems is to use a reduction method, where the given problem is translated into another language, for which sophisticated systems already exist. Logic Programming methods, in particular Answer-Set Programming (ASP) [17] turned out to be a promising direction for this aim, since it not only allows for a concise representation of concepts inherent to argumentation semantics, but also offers sophisticated off-the-shelves solvers which can be used as core computation engines (like Smodels, DLV, clasp or GnT [10]).

1. Background and Overview of the Existing Literature

Previous work has demonstrated that Logic Programming is adequate to encode argumentation problems. Dung has already mentioned in [11] the strong relation between argumentation and Logic Programming. Nieves et. al. proposed in [21] an encoding schema to represent AFs as logic programs, and they showed how different semantics for logic programs can be used to compute different forms of extensions using this particular schema. Furthermore, Nieves et. al proposed in [21] an approach to compute preferred extensions by means of logic programs which requires a recompilation of the encoding for each particular AF. Similarly, [24] also provide ASP encodings for different semantics. In contrast to our work, their encodings for complete and stable semantics are based on labelings, whereas for grounded, preferred and semi-stable semantics they use a meta-programming technique applying additional translations for each AF into normal logic programs. One major difference of our system ASPARTIX [15] to this work is that it uses a *fixed* query for all semantics, which requires the actual instance just as an input database. For the concrete queries, we refer to [15] and for the ideal semantics to [16].

2. Goal of the Research

We want to provide a system for argumentation frameworks which is capable to deal with a broad range of argumentation semantics and generalizations of AFs. We turn our attention especially on a user-friendly implementation which does not require any background knowledge on Logic Programming or ASP. Hence, the user just needs to set up the input database, consisting of problem instance, and select the desired evaluation. We believe that this system can be useful for researchers for analysing and comparing argumentation systems, as well as a versatile decision support system. Especially, we will exploit ASP for more advanced problems. On the one hand, we plan to make use of the rich syntax of ASP (e.g., weak constraints, aggregates, weight constraints, etc.) to deal with weights on arguments or attacks [14, 19]; on the other hand, we want to combine our encodings in order to represent reasoning problems where several semantics come into play (e.g. the coherence problem [13] which decides whether for a given AF F every preferred extension of F is also a stable extension of F).

3. Current Status of the Research

In [15] we presented the first version of ASPARTIX, an ASP tool, which makes use of DLV [18]. This system was designed to compute the basic semantics defined by Dung in [11] such as admissible, complete, preferred, grounded and stable semantics. Additionally we provide encodings for semi-stable [9] and ideal semantics [12]. Furthermore, ASPARTIX can be used to evaluate Preference-based Argumentation Frameworks [1], Value-based Argumentation Frameworks [6], and Bipolar Argumentation Frameworks [2]. All necessary programs to run ASPARTIX are available at

`http://www.dbai.tuwien.ac.at/research/project/argumentation/systempage/`

Currently we are focusing on the encodings of the next generation of argumentation semantics and extensions. Recently, we incorporated the SCC-recursive *cf2* semantics [5] into ASPARTIX. Further encodings include the resolution-based semantics due to Baroni and Giacomin [4] as well as some generalizations of AFs like AFs with Recursive Attacks (AFRAs) [3], Extended AF (EAF) [20] and Dynamic AFs (DAFs) [23].

4. Preliminary Results Accomplished

With the system ASPARTIX, we provide ASP encodings for most of the semantics and frameworks proposed so far. As stated in [15], the encodings are adequate from a complexity point of view. One major advantage of ASPARTIX is that it is independent from the concrete AF to process. It serves as an interpreter which takes an AF given as input. Although there is no advantage of the interpreter approach from a theoretical point of view (as long as the reductions are polynomial-time computable), there are several practical ones. The interpreter is easier to understand, easier to debug, and easier to extend.

5. Open Issues and Expected Achievements

Future work includes a comparison between the different ASP solver and systems wrt our encodings. Especially we will perform run-time tests with the grounders Lparse and Gringo and the solvers Smodels, claspD, GnT2 as well as the system DLV [10]. Preliminary tests showed that our system is capable to deal with frameworks of more than 150 arguments.

As another direction of future work, we will offer a web application of ASPARTIX including a graphical representation of the problem instance and the solution. Hence, researchers can use our system without downloading or installation of any program or ASP solver.

References

[1] Leila Amgoud and Claudette Cayrol. A Reasoning Model Based on the Production of Acceptable Arguments. *Ann. Math. Artif. Intell.*, 34(1-3):197–215, 2002.

[2] Leila Amgoud, Claudette Cayrol, Marie-Christine Lagasquie, and Pierre Livet. On Bipolarity in Argumentation Frameworks. *International Journal of Intelligent Systems*, 23:1–32, 2008.

[3] Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Giovanni Guida. Encompassing Attacks to Attacks in Abstract Argumentation frameworks. In Claudio Sossai and Gaetano Chemello, editors, *Proceedings of the 10th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty* (*ECSQARU 2009*), volume 5590 of *Lecture Notes in Computer Science*, pages 83–94, 2009.

[4] Pietro Baroni and Massimiliano Giacomin. Resolution-Based Argumentation Semantics. In Philippe Besnard, Sylvie Doutre, and Anthony Hunter, editors, *Proceedings of the 2nd Conference on Computational Models of Argument* (*COMMA 2008*), volume 172 of *Frontiers in Artificial Intelligence and Applications*, pages 25–36. IOS Press, 2008.

[5] Pietro Baroni, Massimiliano Giacomin, and Giovanni Guida. SCC-Recursiveness: A General Schema for Argumentation Semantics. *Artif. Intell.*, 168(1-2):162–210, 2005.

[6] Trevor J. M. Bench-Capon. Persuasion in Practical Argument Using Value-based Argumentation frameworks. *J. Log. Comput.*, 13(3):429–448, 2003.

[7] Trevor J. M. Bench-Capon and Paul E. Dunne. Special Issue on Argumentation in Artificial Intelligence. *Artificial Intelligence*, 171(10-15):619–641, 2007. Argumentation in Artificial Intelligence.

[8] Philippe Besnard and Anthony Hunter. *Elements of Argumentation*. The MIT Press, 2008.

[9] Martin Caminada. Semi-Stable Semantics. In Paul E. Dunne and Trevor J. M. Bench-Capon, editors, *Proceedings of the 1st Conference on Computational Models of Argument* (*COMMA 2006*), volume 144 of *Frontiers in Artificial Intelligence and Applications*, pages 121–130. IOS Press, 2006.

[10] Marc Denecker, Joost Vennekens, Stephen Bond, Martin Gebser, and Miroslaw Truszczynski. The Second Answer Set Programming Competition. In Esra Erdem, Fangzhen Lin, and Torsten Schaub, editors, *Proceedings of the 10th International Conference on Logic Programming and Nonmonotonic Reasoning,* (*LPNMR 2009*), volume 5753 of *LNCS*, pages 637–654. Springer, 2009.

[11] Phan M. Dung. On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning, Logic Programming and n-Person Games. *Artif. Intell.*, 77(2):321–358, 1995.

[12] Phan M. Dung, Paolo Mancarella, and Francesca Toni. Computing Ideal Sceptical Argumentation. *Artif. Intell.*, 171(10-15):642–674, 2007.

[13] Paul E. Dunne and Trevor J. M. Bench-Capon. Coherence in Finite Argument Systems. *Artif. Intell.*, 141(1/2):187–203, 2002.

[14] Paul E. Dunne, Anthony Hunter, Peter McBurney, Simon Parsons, and Michael Wooldridge. Inconsistency Tolerance in Weighted Argument Systems. In Carles Sierra, Cristiano Castelfranchi, Keith S. Decker, and Jaime Simão Sichman, editors, *Proceedings of the 8th International Joint Conference on Autonomous Agents and Multiagent Systems,* (*AAMAS 2009*), pages 851–858, 2009.

[15] Uwe Egly, Sarah A. Gaggl, and Stefan Woltran. Answer-Set Programming Encodings for Argumentation Frameworks. Accepted for publication in *Argument and Computation*, 2010; a short version appeared as Technical Report, Technische Universität Wien, DBAI-TR-2008-62.

[16] Wolfgang Faber and Stefan Woltran. Manifold Answer-Set Programs for Meta-Reasoning. In *LPNMR*, pages 115–128, 2009.

[17] Michael Gelfond and Vladimir Lifschitz. Classical Negation in Logic Programs and Disjunctive Databases. *New Generation Comput.*, 9(3/4):365–386, 1991.

[18] Nicola Leone, Gerald Pfeifer, Wolfgang Faber, Thomas Eiter, Georg Gottlob, Simona Perri, and Francesco Scarcello. The DLV System for Knowledge Representation and Reasoning. *ACM Trans. Comput. Log.*, 7(3):499–562, 2006.

[19] Diego C. Martínez, Alejandro J. García, and Guillermo R. Simari. An Abstract Argumentation Framework with Varied-strength attacks. In *Proceedings of the 11h International Conference on Principles of Knowledge Representation and Reasoning,* (*KR 2008*), pages 135–144, 2008.

[20] Sanjay Modgil. Reasoning about Preferences in Argumentation Frameworks. *Artif. Intell.*, 173(9-10):901–934, 2009.

[21] Juan C. Nieves, Mauricio Osorio, and Ulises Cortés. Preferred Extensions as Stable Models. *Theory and Practice of Logic Programming*, 8(4):527–543, 2008.

[22] Iyad Rahwan and Guillermo R. Simari. *Argumentation in Artificial Intelligence*. Springer Publishing Company, Incorporated, 2009.

[23] Nicolas D. Rotstein, Martin O. Maguillansky, Alejandro J. Garcia, and Guillermo R. Simari. An Abstract Argumentation Framework for Handling Dynamics. In *Proceedings of the 12th International Workshop on Non-Monotonic Reasoning* (*NMR'08*), pages 131–139, Sydney, Australia, September 2008.

[24] Toshiko Wakaki and Katsumi Nitta. Computing Argumentation Semantics in Answer Set Programming. In Hiromitsu Hattori, Takahiro Kawamura, Tsuyoshi Idé, Makoto Yokoo, and Yohei Murakami, editors, *New Frontiers in Artificial Intelligence* (*JSAI 2008*)*, Conference and Workshops*, volume 5447 of *LNCS*, pages 254–269. Springer, 2008.