A Discussion of Challenges in Benchmark Generation for Abstract Argumentation

Isabelle **Kuhlmann**, Matthias **Thimm**

University of Hagen, Germany

Abstract

Abstract argumentation provides a formal framework for modeling and analyzing argumentative reasoning processes. As the field progresses, the need for benchmarks to evaluate and compare different algorithmic approaches becomes increasingly important. However, the process of generating suitable benchmarks for abstract argumentation is not without its challenges. This paper aims to explore the key challenges encountered in benchmark generation for abstract argumentation. In particular, we address the task of skeptical acceptability w.r.t. preferred semantics and describe a benchmark generator designed for this specific problem.

Keywords

Abstract argumentation, benchmarking, graph generator

1. Introduction

As a central aspect of human communication, the concept of argumentation has been adopted in the area of Artificial Intelligence in various forms. The principle of *abstract argumentation* [1], which focuses on the interplay between arguments in order to gain insights and reach conclusions, has become an established mechanism of non-monotonic reasoning. Naturally, an important issue in advancing the research in this field—in particular with regard to algorithmic solutions and applications—is the availability of benchmark data. However, generating suitable benchmarks for abstract argumentation presents several challenges that require careful consideration.

One notable initiative in advancing the evaluation of argumentation systems is the International Competition on Computational Models of Argumentation (ICCMA)[1]. ICCMA serves as a platform for researchers and practitioners to showcase their systems and compare their performance on a common set of benchmarks. While ICCMA has significantly contributed to the evaluation of argumentation systems, the process of generating benchmarks for abstract argumentation remains a perpetual task. For instance, w.r.t. a set of ICCMA'17 benchmarks, it was recently pointed out that a majority of arguments that are skeptically accepted under preferred semantics (a task which is Π_2^P-complete [2]) is also accepted under grounded semantics

CEUR Workshop Proceedings (CEUR-WS.org)

[1]http://argumentationcompetition.org/index.html

(which can be decided in polynomial time) [3]. Thus, in the majority of cases, it is sufficient to solve a less complex problem, which may distort the interpretation of experimental results. In this paper, we revisit this issue to highlight the impact of such properties on practical results. Additionally, we introduce in more detail the *KWT Benchmark Generator* [3], which is designed to circumvent the aforementioned problem. Further challenges arise in numerous respects, including the need for diversity in benchmark scenarios, scalability concerns, or appropriate evaluation metrics. Addressing these challenges is crucial to developing comprehensive benchmarking methodologies that accurately reflect the performance and capabilities of different argumentation systems.

2. Preliminaries

An (abstract) *argumentation framework* (AF) [1] is a pair $F = (\mathsf{Arg}, R)$, with Arg being a set of arguments and $R \subseteq \mathsf{Arg} \times \mathsf{Arg}$ a relation between those arguments. An argument $a \in \mathsf{Arg}$ *attacks* an argument $b \in \mathsf{Arg}$ if $(a, b) \in R$. Moreover, we define the set of arguments attacking a given argument a as $a_F^- = \{b \mid (b, a) \in R\}$, and the set of arguments being attacked by a as $a_F^+ = \{b \mid (a, b) \in R\}$. In the same fashion we define E_F^- and E_F^+ for a set $E \subseteq \mathsf{Arg}$. We call an argument $a \in \mathsf{Arg}$ *defended* by a set of arguments $E \subseteq \mathsf{Arg}$ if every argument $b \in \mathsf{Arg}$ that attacks a is itself attacked by some argument $c \in E$, i.e., if $a_F^- \subseteq E_F^+$.

Further, a set $E \subseteq \mathsf{Arg}$ is *conflict-free* if $E \cap E_F^+ = \emptyset$. If a set $E \subseteq \mathsf{Arg}$ is conflict-free and each $a \in E$ is defended by E, we call E admissible (ad). We call sets of jointly acceptable arguments *extensions*, which can be defined under various semantics. The classical semantics, following the seminal work by Dung [1], are defined as follows:

- A set $E \subseteq \mathsf{Arg}$ is *complete* (co) iff it is admissible, and if E defends $a \in \mathsf{Arg}$ then $a \in E$.
- A set $E \subseteq \mathsf{Arg}$ is *grounded* (gr) iff E is complete and $\subseteq$-minimal.
- A set $E \subseteq \mathsf{Arg}$ is *preferred* (pr) iff E is complete and $\subseteq$-maximal.
- A set $E \subseteq \mathsf{Arg}$ is *stable* (st) iff it is complete and $E \cup E_F^+ = \mathsf{Arg}$.

In addition, we define a set $E \in \mathsf{Arg}$ to be an *ideal* (id) extension [4] if E is admissible, for every preferred extension E', it holds that $E \subseteq E'$, and E is $\subseteq$-maximal with these two properties. Note that the grounded and the ideal extension of an AF are each a uniquely defined, and that the former is always a subset of the latter.

Typical problems in the field of abstract argumentation involve the enumeration of one extension (or all extensions), or deciding whether a given argument is included in one extension (or all extensions) w.r.t. a given semantics. Let $\Sigma = \{\mathsf{co}, \mathsf{gr}, \mathsf{pr}, \mathsf{st}, \mathsf{id}\}$. An argument is *credulously* accepted w.r.t. $\sigma \in \Sigma$ if it is included in at least one σ extension, and it is *skeptically* accepted if it is included in all σ extensions. We denote the computational problem of credulous acceptability regarding a semantics σ as DC_σ, and the problem of skeptical acceptability regarding σ as DS_σ.

3. Challenges in Benchmark Generation

In the following, we provide a brief overview of existing benchmarks (which were used in past editions of ICCMA) and subsequently discuss challenges that arise in the development of new benchmark generation techniques.

3.1. Existing Benchmarks

Existing benchmarks for abstract argumentation can be roughly categorized into the following groups:

- **Random graphs.** Some benchmark instances, e.g., those provided by *AFBenchGen* [5], comprise AFs generated using random graph generation algorithms. A related approach consists of connecting multiple random graphs to model communities of arguments [6].
- **Graphs tailored for argumentation.** Some benchmarks are aimed at specific abstract argumentation problems. Examples are *AdmBuster* [7], which is targeted to the problems DC_{pr} and DC_{gr}, and *SemBuster* [8], which is designed for problems regarding the semi-stable semantics [9]. Further, the *GroundedGenerator* produces AFs with a large grounded extension, the *StableGenerator* produces AFs with many stable extensions, and the *SccGenerator* produces graphs with many strongly-connected components [10]. Although these generators are not necessarily aimed at a specific argumentation problem, they were designed with abstract argumentation as the target application in mind, and allow for investigating certain solver properties (e.g., whether a solver exploits the fact that an argument accepted under grounded semantics—which is computationally easy to obtain—is also accepted under other semantics).
- **Translations from other domains.** AFs can be created by transforming existing problems or data sets from other domains. Examples include benchmarks from planning [11], assumption-based argumentation [12], mass transit data [13], and (inconsistent) knowledge bases expressed in the *Datalog$^{\pm}$* language [14].

3.2. Challenges in the Generation of Novel Benchmarks

When creating a benchmark data set, the overall goal should be to obtain a *diverse* set of argumentation frameworks. Since the term "diversity" allows for multiple perspectives, we discuss some key aspects in the following.

Graph-Theoretical Features   In order to ensure diversity in a graph-theoretical sense, benchmarks for abstract argumentation should encompass a wide range of graph properties and characteristics. This includes various properties, for example the node degree, the occurrence and number of (odd) cycles, variations in connectivity patterns, such as different levels of connectedness, etc. When creating new benchmarks, an analysis regarding such graph properties is valuable in order to check how newly generated AFs differ from existing benchmarks from a graph-theoretical perspective. New graph generators may also offer the possibility of parameterizing a number of graph features (which is already possible, to a degree, with most random graph generators). On the other hand, this may not be applicable in some scenarios (e.g., when dealing with real-world data).

Relation to Real-World Scenarios   Creating benchmarks that reflect real-world argumentation scenarios can be challenging. Abstract argumentation frameworks might abstract from the complexity of real-world arguments and their relationships. Moreover, different domains, such as law, politics, healthcare, or ethics, have unique argumentation characteristics and

requirements. Incorporating domain-specific considerations in benchmark generation allows for more targeted evaluations and comparisons of argumentation systems within their intended domains.

Semantic Aspects   Benchmarks should also be geared towards evaluating solution approaches to the different problems related to abstract argumentation. Some benchmarks are already designed for such purposes (such as *SemBuster*, which is aimed at problems related to the semi-stable semantics), however, there are still numerous problems that have not been specifically addressed yet. As an example, it was recently demonstrated that in most ICCMA'17 instances, a majority of skeptically accepted arguments w.r.t. pr were also included in the grounded and the ideal extension. Since the computational complexity of deciding DS_{pr} is Π_2^P-complete [2], but problems related to id are "only" Θ_2^P-complete, and the grounded extension can be computed in polynomial time. Hence, even though the task of deciding DS_{pr} is computationally complex, it can still be computed relatively efficiently, due to the occurrence of many "easy" cases.

4. KWT Benchmark Generator

In the previous section, we identified a number of challenges that occur in the generation of benchmarks for abstract argumentation. Since it is not reasonable to address all concerns within one graph generator, we focus on a specific semantic aspect as an example, namely the issue that solving the Π_2^P-complete problem DS_{pr} can often be bypassed by checking if the given argument is accepted w.r.t. gr or id. Note that another "easy case" regarding DS_{pr} occurs when arguments are attacked by some admissible set—such arguments are never skeptically accepted w.r.t. pr—and deciding this is a problem in NP. In [3], we briefly introduced a possible solution for this problem. In the following, we provide a more thorough description of our approach.

We developed the *KWT generator*, which takes as parameters

- num_{args}: the total number of arguments,
- num_{pa}: the number of arguments to be skeptically accepted under preferred semantics,
- num_{cred}: the number of arguments to be contained in at least one preferred extension,
- num_{pref}: the number of preferred extensions,
- num_{ideal}: the number of arguments in the ideal extension,

and further parameters that control the probability of attacks between different sets of arguments. More precisely, these parameters set the probabilities of arguments in the ideal extension to be attacked and to attack back, respectively, the probabilities of credulously accepted arguments to be attacked and to attack back, the probabilities of skeptically accepted arguments that are not contained in the ideal extension to be attacked and to attack back, and the probability of further random attacks between unaccepted arguments. Given these parameters, a random AF F is generated as follows:

1. The set Arg of num_{args} arguments is created and arguments are associated to sets S_{pa} (skeptically accepted arguments w.r.t. preferred semantics), S_{ideal} (arguments in the ideal extension), S_{cred} (arguments that are credulously accepted w.r.t. preferred semantics),

S_{unacc} (arguments that are not credulously accepted w.r.t. preferred semantics), such that $S_{ideal} \subseteq S_{pa} \subseteq S_{cred}$, $S_{cred} \cup S_{unacc} = \mathsf{Arg}$, and the corresponding cardinalities are respected. Finally, sets $E_1, \ldots, E_{num_{pref}}$ (the preferred extensions) are created by adding all arguments from S_{pa} and randomly drawn arguments from $S_{cred} \setminus S_{pa}$.

2. For every argument $a \in S_{ideal}$, random attackers from S_{unacc} are sampled. For each of these attackers b, another argument from S_{ideal} is sampled that attacks b. This ensures that the grounded extension will be empty and that the ideal extension is capable of defending itself (thus forming an admissible set).

3. For every argument $a \in S_{pa} \setminus S_{ideal}$, attacks from unaccepted arguments are sampled in a similar way (to ensure an empty grounded extension). Furthermore, every such argument a must be defended by each preferred extension. Thus, for each preferred extension E, some arguments are sampled to defend a.

4. For every preferred extension E and $a \in E \setminus S_{pa}$, attackers for a are sampled from $\mathsf{Arg} \setminus E$ and corresponding defenders are defined within E.

5. Additional random attacks are added between arguments in S_{unacc}.

6. In order to avoid having stable extensions (which may also ease computation of arguments that are skeptically accepted under preferred semantics, since every stable extension is also preferred), we add another self-attacking argument and some attacks between this argument and arguments from S_{unacc}.

Note that due to the random approach of generating an argumentation graph, it may not necessarily be the case that the number of skeptically/credulously accepted arguments (w.r.t. preferred semantics) as well as the number of arguments in the ideal extension exactly match the given parameters. However, our experiments in [3] showed that it is indeed relatively hard to decide skeptical acceptance (w.r.t. preferred semantics) for most arguments in the resulting graph.

The graph generator[2] and an example demonstrating its usage[3] we used can be found online.

5. Conclusion

Throughout this paper we discussed how the generation of new benchmarks for abstract argumentation problems can be challenging from multiple perspectives. Although existing benchmarks for abstract argumentation already provide valuable resources for evaluating and comparing different frameworks and algorithms, they may not adequately capture the challenges and requirements posed by recent developments in the field. Moreover, existing benchmarks may have limitations in terms of the problem space they cover (e.g. concerning characteristics of different graph properties).

Overall, we would like to highlight that new benchmarking techniques should yield AFs that are indeed novel in some regard, i.e., which differ from existing data—for instance, in terms of graph-theoretical properties, by addressing previously little considered semantic aspects, or by incorporating new real-world problems. Combining all of these different facets in one

---

[2]http://tweetyproject.org/r/?r=kwt_gen
[3]http://tweetyproject.org/r/?r=kwt_gen_ex

single generator is presumably rather difficult, however, considering them individually might already lead to new insights. As an example, we presented the KWT generator, which generates particularly challenging AFs for the task of deciding skeptical acceptability w.r.t. preferred semantics.

Acknowledgments

References

[1] P. M. Dung, On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning, Logic Programming and n-Person Games, Artificial Intelligence 77 (1995) 321–358.

[2] P. E. Dunne, T. J. M. Bench-Capon, Coherence in finite argument systems, Artificial Intelligence 141 (2002) 187–203.

[3] I. Kuhlmann, T. Wujek, M. Thimm, On the impact of data selection when applying machine learning in abstract argumentation, in: Computational Models of Argument, IOS Press, 2022, pp. 224–235.

[4] P. M. Dung, P. Mancarella, F. Toni, Computing ideal sceptical argumentation, Artificial Intelligence 171 (2007) 642–674.

[5] F. Cerutti, M. Giacomin, M. Vallati, Generating challenging benchmark afs., COMMA 14 (2014) 457–458.

[6] J.-M. Lagniez, E. Lonca, J.-G. Mailly, J. Rossit, Design and results of iccma 2021, arXiv preprint arXiv:2109.08884 (2021).

[7] M. Caminada, M. Podlaszewski, Admbuster: a benchmark example for (strong) admissibility, 2017. The Second International Competition on Computational Models of Argumentation (ICCMA'17).

[8] M. Caminada, B. Verheij, Sembuster: a benchmark example for semi-stable semantics, 2017.

[9] M. Caminada, W. A. Carnielli, P. E. Dunne, Semi-stable semantics, J. Log. Comput. 22 (2012) 1207–1254.

[10] M. Thimm, S. Villata, The first international competition on computational models of argumentation: Results and analysis, Artificial Intelligence 252 (2017) 267–294.

[11] F. Cerutti, M. Giacomin, M. Vallati, Exploiting planning problems for generating challenging abstract argumentation frameworks, URL: http://argumentationcompetition. org/2017/Planning2AF. pdf (2017).

[12] T. Lehtonen, J. P. Wallner, M. Järvisalo, Assumption-based argumentation translated to argumentation frameworks, URL: http://argumentationcompetition. org/2017/ABA2AF. pdf (2017).

[13] M. Diller, Traffic networks become argumentation frameworks, URL: http://argumentationcompetition. org/2017/Traffic. pdf (2017).

[14] B. Yun, M. Croitoru, Benchmark on logic-based argumentation framework with datalog±, 2019.