Revisiting Approximate Reasoning based on Grounded Semantics

Jérôme Delobelle, Jean-Guy Mailly, Julien Rossit

Revisiting Approximate Reasoning based on Grounded Semantics*

Jérôme Delobelle[1 [0000−0003−1691−4731]], Jean-Guy Mailly[1 [0000−0001−9102−7329]], and Julien Rossit[1 [0000−0001−5275−4435]]

Université Paris Cité, LIPADE, F-75006 Paris, France
{jerome.delobelle,jean-guy.mailly,julien.rossit}@u-paris.fr

Abstract. Efficient computation of hard reasoning tasks is a key issue in abstract argumentation. One recent approach consists in defining approximate algorithms, *i.e.* methods that provide an answer that may not always be correct, but outperforms the exact algorithms regarding the computation runtime. One such approach proposes to use the grounded semantics, which is polynomially computable, as a starting point for determining whether arguments are (credulously or skeptically) accepted with respect to various semantics. In this paper, we push further this idea by defining various approaches to evaluate the acceptability of arguments which are not in the grounded extension, neither attacked by it. We have implemented our approaches, and we describe the result of their empirical evaluation.

Keywords: Abstract argumentation · Approximate reasoning.

Note: This paper has been accepted at ECSQARU 2023. In this version, we fix minor mistakes in some examples. Please refer to the original publication [9] to refer this work.

1 Introduction

Designing algorithms that are computationally effective is an important issue for many fields of Artificial Intelligence. Traditionally, existing approaches addressed for automated reasoning can be divided into two families responding to different philosophies, namely complete or approximate approaches. While the former aim to produce the correct answer however long it takes to produce it, the latter focus on responding as quickly as possible at the expense of the risk of missing it.

Among many symbolic representations offered by the literature of Artificial Intelligence to model a problem, Abstract Argumentation [10] is an intuitive and convenient framework to reason with conflicting pieces of information. Classical semantics for reasoning with an abstract argumentation framework (AF) are based on the notion of extension, *i.e.* sets of collectively acceptable arguments.

* This work benefited from the support of the project AGGREEY ANR-22-CE23-0005 of the French National Research Agency (ANR).

However, a potential drawback of this framework is the high complexity of many classical reasoning tasks for most of the semantics [11]. Among recent approaches for solving these problems, many exact algorithms (which always provide a correct answer, but may require an exponential runtime in the worst case since the problems at hand are theoretically intractable) have been proposed (*e.g.* the SAT-based approaches by [16, 20, 12]). Fewer approximate algorithms have been proposed. We are particularly interested in the Harper++ approach [21], which consists in using the grounded semantics (which is polynomially computable) as a shortcut for determining which arguments are acceptable with respect to various extension-based semantics. Indeed, it is known that any argument belonging to the (unique) grounded extension also belongs to all the extensions for several classical semantics, while an argument which is attacked by the grounded extension cannot belong to any extension (again, for most classical semantics). Thus, the question is how to determine the acceptability of other arguments, those which do not belong to the grounded extension nor are attacked by it. Harper++ proposes to consider that all these arguments should be credulously accepted (*i.e.* belong to at least one extension), and none of them should be skeptically accepted (*i.e.* none of them belongs to any extension). In this paper, we study the question of approximating the acceptability of these arguments, and propose two families of approaches to solve this problem. In the first one, we compare the out-degree and in-degree of arguments (*i.e.* the number of arguments they attack or which attack them), and in the second one we rely on a classical gradual semantics, namely h-categorizer [3]. Each of these families of approaches depends on some parameters. We show that they generalize the Harper++ algorithm in the sense that setting some value to the parameters induces that our approaches have the same behaviour as Harper++. Then, we empirically evaluate the accuracy of our approaches, *i.e.* the ratio of correct answers that we obtain. Finally, we show that our approach is (as expected from an approximate algorithm) significantly faster than a SAT-based exact algorithm.

In Section 2 we recall background notions of abstract argumentation and the Harper++ approach. We describe our new methods for approximating the acceptability of arguments in Section 3. Section 4 presents our experimental evaluation and we analyse the results in Section 5. We describe some related work in Section 6, and Section 7 concludes the paper.

2 Background

2.1 Abstract Argumentation

Let us first recall some basic notions of abstract argumentation [10].

Definition 1. *An argumentation framework (AF) is a directed graph* $\mathcal{F} = \langle \mathcal{A}, \mathcal{R} \rangle$ *where* $\mathcal{A}$ *is a finite and non-empty set of arguments and* $\mathcal{R} \subseteq (\mathcal{A} \times \mathcal{A})$ *is an attack relation over the arguments. We say that* $a \in \mathcal{A}$ *(resp.* $S \subseteq \mathcal{A}$*)* attacks $b \in \mathcal{A}$ *if* $(a, b) \in \mathcal{R}$ *(resp.* $\exists c \in S$ *s.t.* $(c, b) \in \mathcal{R}$*). Then,* $S \subseteq \mathcal{A}$ *defends an argument* a *if* S *attacks all the arguments that attack* a*.*

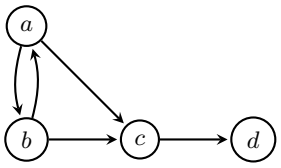
Reasoning with an AF can then be achieved through the notion of extension, *i.e.* sets of arguments which are collectively acceptable. An extension S must satisfy some requirements, namely *conflict-freeness i.e.* no arguments in S attack each other, and *admissibility i.e.* S must be conflict-free and defend all its elements. Conflict-free and admissible sets are denoted by $\mathtt{cf}(\mathcal{F})$ and $\mathtt{ad}(\mathcal{F})$. Then, Dung defines four semantics which induce different sets of extensions. Thus, a set of arguments $S \subseteq \mathcal{A}$ is:

- a *complete extension* iff $S \in \mathtt{ad}(\mathcal{F})$ and S does not defend any argument outside of S;
- a *preferred extension* iff S is a $\subseteq$-maximal admissible set;
- a *stable extension* iff $S \in \mathtt{cf}(\mathcal{F})$ and S attacks every $a \in \mathcal{A} \setminus S$;
- a *grounded extension* iff S is a $\subseteq$-minimal complete extension.

We use $\mathtt{co}(\mathcal{F})$, $\mathtt{pr}(\mathcal{F})$, $\mathtt{stb}(\mathcal{F})$ and $\mathtt{gr}(\mathcal{F})$ for representing these sets of extensions.

It is well known [10] that, for every AF $\mathcal{F}$, $\mathtt{stb}(\mathcal{F}) \subseteq \mathtt{pr}(\mathcal{F}) \subseteq \mathtt{co}(\mathcal{F})$, and $|\mathtt{gr}(\mathcal{F})| = 1$. For these reasons, any argument $a \in E$ (where E is the unique grounded extension) belongs to every complete extension (since E is the unique $\subseteq$-minimal one), and thus to every preferred and stable extension (which are all complete extensions as well). And similarly, any argument b attacked by E is attacked by an argument a which belongs to every stable, preferred or complete extension, so b does not belong to any stable, preferred or complete extension. Finally, given an extension-based semantics σ, an argument is *credulously* (resp. *skeptically*) accepted w.r.t. σ if it belongs to some (resp. each) σ-extension. This is denoted by $a \in \mathtt{cred}_\sigma(\mathcal{F})$ (resp. $a \in \mathtt{skep}_\sigma(\mathcal{F})$).

Example 1. Figure 1 gives an example of AF $\mathcal{F} = \langle \mathcal{A}, \mathcal{R} \rangle$. Its extensions and sets of (credulously and skeptically) accepted arguments for the four extension-based semantics described previously are also given.



σ	$\sigma(\mathcal{F})$	$\mathtt{cred}_\sigma(\mathcal{F})$	$\mathtt{skep}_\sigma(\mathcal{F})$
co	$\{\emptyset, \{a,d\}, \{b,d\}\}$	$\{a,b,d\}$	$\emptyset$
pr	$\{\{a,d\}, \{b,d\}\}$	$\{a,b,d\}$	$\{d\}$
stb	$\{\{a,d\}, \{b,d\}\}$	$\{a,b,d\}$	$\{d\}$
gr	$\{\emptyset\}$	$\emptyset$	$\emptyset$

Fig. 1. An example of AF $\mathcal{F}$ (left) with the extensions and accepted arguments for the four semantics $\sigma \in \{\mathtt{co}, \mathtt{pr}, \mathtt{stb}, \mathtt{gr}\}$ (right).

2.2 Harper++ for Approximate Reasoning

From the previous observation on the relation between the grounded extension and the other semantics, one can notice that using the grounded extension as an

approximation of reasoning with other semantics makes sense, especially since computing the grounded semantics is achieved in polynomial time, while other semantics are generally computationally hard [11]. This has conducted to an empirical evaluation of the similarity between the grounded semantics and the other forms of reasoning in argumentation [7]. A consequence of this work is the development of the approximate argumentation solver Harper++ [21], which works as follows. First consider an AF $\mathcal{F} = \langle \mathcal{A}, \mathcal{R} \rangle$, an argument $a \in \mathcal{A}$, and a reasoning task among DS-σ and DC-σ, corresponding to the skeptical and credulous acceptability, *i.e.* checking whether the argument a belongs to $\texttt{skep}_\sigma(\mathcal{F})$ or $\texttt{cred}_\sigma(\mathcal{F})$. Then, for any semantics, Harper++ computes the grounded extension E of $\mathcal{F}$, and answers:

- YES if a belongs to E,
- NO if a is attacked some $b \in E$,
- otherwise, YES if the problem is DC-σ, and NO if the problem is DS-σ.

This last category of arguments corresponds to the UNDEC arguments with respect to the grounded labelling [5]. We use $\text{IN}(\mathcal{F})$ to denote the set of arguments which belong to the grounded extension of $\mathcal{F}$, $\text{OUT}(\mathcal{F})$ for the arguments attacked by a member of the grounded extension, and $\text{UNDEC}(\mathcal{F})$ for the last category. Formally, the Harper++ approach works as follows:

Definition 2. *Given $\mathcal{F} = \langle \mathcal{A}, \mathcal{R} \rangle$ an AF, $a \in \mathcal{A}$ an argument and $x \in \{\text{DC}, \text{DS}\}$, the function Acc^{++} is defined by:*

$$Acc^{++}(\mathcal{F}, a, x) = \begin{cases} YES & \text{if } a \in \text{IN}(\mathcal{F}) \\ & \text{or } (a \in \text{UNDEC}(\mathcal{F}) \text{ and } x = \text{DC}), \\ NO & \text{otherwise.} \end{cases}$$

Example 2. Continuing the previous example, since the grounded extension is empty, all the arguments belong to $\text{UNDEC}(\mathcal{F})$. So, $Acc^{++}(\mathcal{F}, a_i, \text{DC}) = YES$ and $Acc^{++}(\mathcal{F}, a_i, \text{DS}) = NO$ for all $a_i \in \mathcal{A}$. This means, for instance, that Acc^{++} answers correctly for 3 arguments for the problem DC-`stb` (a, b and d) and for 3 arguments (a, b and c) for the problem DS-`stb`.

3 New Approaches to Acceptability Approximation

A natural question is then whether one can find a better way to approximate the acceptability of arguments with respect to the stable, preferred and complete semantics when they belong to $\text{UNDEC}(\mathcal{F})$. We propose two approaches for addressing this issue, respectively based on a comparison between the out-degree and the in-degree associated with the considered argument on one side, and on an evaluation of its acceptability using a gradual semantics on the other side.

3.1 ARIPOTER-degrees: ARgumentatIon apPrOximaTE Reasoning using In/Out Degrees of Arguments

First, let us define some additional notations. Given an AF $\mathcal{F} = \langle \mathcal{A}, \mathcal{R} \rangle$ and an argument $a \in \mathcal{A}$, define $a^+ = \{b \in \mathcal{A} \mid (a, b) \in \mathcal{R}\}$ and $a^- = \{b \in \mathcal{A} \mid (b, a) \in \mathcal{R}\}$, *i.e.* the set of arguments attacked by a or attacking a, respectively. We call *in-degree* of a the number of attackers of a, *i.e.* $|a^-|$, and *out-degree* of a the number of arguments attacked by a *i.e.* $|a^+|$.

The intuition behind this approach is that an argument which attacks more arguments than the number of its attackers has good chances to defend itself, and then being accepted. Then, given some $k \in \mathbb{R}$, we assume that a is accepted when $|a^+| \geq k \times |a^-|$.

Definition 3. *Given $\mathcal{F} = \langle \mathcal{A}, \mathcal{R} \rangle$ an AF, $a \in \mathcal{A}$ an argument and $k \in \mathbb{R}$, the function $Acc^{Out/In}$ is defined by:*

$$Acc^{Out/In}(\mathcal{F}, a, k) = \begin{cases} YES & \text{if } a \in \text{IN}(\mathcal{F}) \\ & \text{or } (a \in \text{UNDEC}(\mathcal{F}) \text{ and } |a^+| \geq k \times |a^-|), \\ NO & \text{otherwise.} \end{cases}$$

This means that arguments in $\text{IN}(\mathcal{F})$ and $\text{OUT}(\mathcal{F})$ are respectively accepted and rejected, as expected, and UNDEC arguments are considered as accepted iff their out-degree is at least k times higher than their in-degree. We will see in Section 5 that this parameter can be adapted according to the type of AF evaluated and the problem considered.

Example 3. We continue the previous example. Fixing $k = 1$, we observe that $Acc^{Out/In}(\mathcal{F}, a_i, 1) = YES$ for $a_i \in \{a, b\}$ and $Acc^{Out/In}(\mathcal{F}, a_j, 1) = NO$ for $a_j \in \{c, d\}$. So this approach provides a correct answer to *e.g.* DC-`stb` for 3 arguments (a, b and c), and to DS-`stb` for 1 argument (c).

3.2 ARIPOTER-hcat: ARgumentatIon apPrOximaTE Reasoning using the h-Categorizer semantics

Our second approach is to use gradual semantics to assess the acceptability of UNDEC arguments. A gradual semantics [2] is a function mapping each argument in an AF to a number representing its strength, classically in the interval $[0, 1]$. As explained in [1], the acceptability of an argument is, in this case, related to its strength in the sense that only "strong" arguments can be considered accepted. It should also be noted that existing gradual semantics use evaluation criteria that differ from extension-based semantics, such as the quality or quantity of direct attackers, but does not necessarily satisfy the condition of conflict-freeness. However, although these two families of semantics are different, certain aspects of gradual semantics can be used to try to assess whether an argument seems acceptable or not in the context of extension-based semantics. For example, a characteristic shared by most gradual semantics is that the less an argument is attacked, the stronger it is. This suggests that an argument

with a high score is more likely to belong to an extension. The aim of our approach is therefore to accept UNDEC arguments whose score is greater than a given threshold w.r.t. some gradual semantics. Although our approach can be generalised to all gradual semantics, we focus here on the h-categorizer semantics [3]. This gradual semantics uses a function whose purpose is to assign a value which captures the relative strength of an argument taking into account the strength of its attackers which takes into account the strength of their attackers, and so on. Formally, given an AF $\mathcal{F} = \langle \mathcal{A}, \mathcal{R} \rangle$ and $a \in \mathcal{A}$, $\texttt{h-cat}(\mathcal{F}, a) = \frac{1}{1+\sum_{b \in a^-} \texttt{h-cat}(\mathcal{F}, b)}$.

To use this gradual semantics for deciding the acceptability an UNDEC argument a, we consider that a is accepted if $\texttt{h-cat}(\mathcal{F}, a) \geq \tau$ with $\tau \in [0, 1]$.

Definition 4. *Given $\mathcal{F} = \langle \mathcal{A}, \mathcal{R} \rangle$ an AF, $a \in \mathcal{A}$ an argument and $\tau \in [0, 1]$, the function $Acc^{\texttt{h-cat}}$ is defined by:*

$$Acc^{\texttt{h-cat}}(\mathcal{F}, a, \tau) = \begin{cases} YES & \text{if } a \in \text{IN}(\mathcal{F}) \\ & \text{or } (a \in \text{UNDEC}(\mathcal{F}) \text{ and } \texttt{h-cat}(\mathcal{F}, a) \geq \tau), \\ NO & \text{otherwise.} \end{cases}$$

Example 4. Again, continuing the previous example, we see that $\texttt{h-cat}(\mathcal{F}, a) = \texttt{h-cat}(\mathcal{F}, b) \approx 0.62$, $\texttt{h-cat}(\mathcal{F}, c) \approx 0.45$ and $\texttt{h-cat}(\mathcal{F}, d) \approx 0.69$. So, setting $\tau = 0.5$ allows $Acc^{\texttt{h-cat}}$ to give perfect answers to DC-σ queries for the AF $\mathcal{F}$, and $\tau = 0.65$ leads to perfect answers to DS-σ queries for $\mathcal{F}$, for $\sigma \in \{\texttt{stb}, \texttt{pr}\}$.

3.3 Relationships between Approaches

Both our new approaches generalize the Harper++ approach, *i.e.* by choosing a good value for τ or k, our approach recovers the result of Harper++ for the UNDEC arguments (it is obvious that arguments in IN($\mathcal{F}$) or OUT($\mathcal{F}$) are treated equally by Harper++ and our new approaches).

Proposition 1. *For any AF $\mathcal{F} = \langle \mathcal{A}, \mathcal{R} \rangle$ and $a \in \mathcal{A}$, the following hold:*

- $Acc^{++}(\mathcal{F}, a, \text{DC}) = Acc^{Out/In}(\mathcal{F}, a, 0)$;
- $Acc^{++}(\mathcal{F}, a, \text{DC}) = Acc^{\texttt{h-cat}}(\mathcal{F}, a, 0)$.

Proof. Recall that Harper++ considers as accepted any UNDEC argument in the case of credulous acceptability, *i.e.* $Acc^{++}(\mathcal{F}, a, \text{DC}) = YES$ if $a \in \text{UNDEC}(\mathcal{F})$. Obviously, with $k = 0$, the inequality $|a^+| \geq k \times |a^-|$ is satisfied for any argument, which means that $Acc^{Out/In}(\mathcal{F}, a, 0) = YES$ when $a \in \text{UNDEC}(\mathcal{F})$. Similarly, any argument has a h-categorizer value greater than 0, so $Acc^{\texttt{h-cat}}(\mathcal{F}, a, 0) = YES$ when $a \in \text{UNDEC}(\mathcal{F})$. □

Proposition 2. *For any AF $\mathcal{F} = \langle \mathcal{A}, \mathcal{R} \rangle$ and $a \in \mathcal{A}$, the following hold:*

- $Acc^{++}(\mathcal{F}, a, \text{DS}) = Acc^{Out/In}(\mathcal{F}, a, |\mathcal{A}| + 1)$;
- $Acc^{++}(\mathcal{F}, a, \text{DS}) = Acc^{\texttt{h-cat}}(\mathcal{F}, a, 1)$.

Proof. We know that $Acc^{++}(\mathcal{F}, a, \text{DS}) = NO$ when $a \in \text{UNDEC}(\mathcal{F})$.

For the approach based on the out-degree and in-degree, observe that for $a \in \text{UNDEC}(\mathcal{F})$, $|a^+|$ cannot be greater than the number of arguments in the AF, and $|a^-|$ cannot be equal to 0 (because otherwise, a would belong to $\text{IN}(\mathcal{F})$). So, by setting $k = |\mathcal{A}| + 1$ the inequality $|a^+| \geq k \times |a^-|$ can never be true. This means that for any argument $a \in \text{UNDEC}(\mathcal{F})$, $Acc^{Out/In}(\mathcal{F}, a, |\mathcal{A}| + 1) = NO$.

Consider now $Acc^{\texttt{h-cat}}$, *i.e.* the approach based on h-categorizer. We know that only unattacked arguments have a value of h-categorizer equal to 1, and these arguments are in $\text{IN}(\mathcal{F})$. So for any $a \in \text{UNDEC}(\mathcal{F})$, $\texttt{h-cat}(a) < 1$. This means that for any such argument, $Acc^{\texttt{h-cat}}(\mathcal{F}, a, 1) = NO$. □

4 Experimental Settings

We focus on four decision problems: DC-`stb`, DS-`stb`, DC-`pr` and DS-`pr`. Recall that DC-σ is NP-complete for $\sigma \in \{\texttt{stb}, \texttt{pr}\}$, and DS-$\sigma$ is coNP-complete for $\sigma = \texttt{stb}$ and Π_2^P-complete for $\sigma = \texttt{pr}$ [11]. Among Dung's classical semantics, we ignore the grounded semantics since it is polynomially computable and at the base of all the approaches described here. We also ignore the complete semantics because DS-`co` is equivalent to DS-`gr`, and DC-`co` is equivalent to DC-`pr`.

4.1 Benchmarks

Random graphs We consider an experimental setting representing three different models used during the ICCMA competition [13] as a way to generate random argumentation graphs: the Erdös-Rényi model (ER) which generates graphs by randomly selecting attacks between arguments; the Barabási-Albert model (BA) which provides networks, called scale-free networks, with a structure in which some nodes have a huge number of links, but in which nearly all nodes are connected to only a few other nodes; and the Watts-Strogatz model (WS) which produces graphs which have small-world network properties, such as high clustering and short average path lengths.

The generation of these three types of AFs was done by the AFBenchGen2 generator [6]. We generated a total of 9460 AFs almost evenly distributed between the three models (3000 AFs for the WS model and 3230 AFs for the ER and BA model)[1]. For each model, the number of arguments varies among $\texttt{Arg} = \{10, 20, 30, 40, 50, 60, 70, 80, 90, 100\}$. The parameters used to generate graphs are as follows: for ER, 19 instances for each (`nbArg`, `pAtt`) in $\texttt{Arg} \times \{0.15, 0.2, \ldots, 0.95\}$; for BA, 17 instances for each (`nbArg`, `pCyc`) in $\texttt{Arg} \times \{0, 0.05, 0.1, \ldots, 0.9\}$; for WS, 5 instances for each (`nbArg`, `pCyc`, β, $\mathcal{K}$) in $\texttt{Arg} \times \{0.25, 0.5, 0.75\} \times \{0, 0.25, 0.5, 0.75, 1\} \times \{k \in 2\mathbb{N}\ s.t.\ 2 \leq k \leq \texttt{nbArg}-1\}$. We refer the reader to [6] for the meaning of the parameters. For each instance, an argument is randomly chosen to serve as the query for DC and DS problems.

[1] The set of instances can be found at the following address:
`https://cloud.parisdescartes.fr/index.php/s/diZAz5sTzWbNCMt`

In the following, we collectively refer to the group of AFs generated using the Erdös-Rényi model (resp. Barabási-Albert model and Watts-Strogatz model) as `rER` (resp. `rBA` and `rWS`). Finally, the notation `randomAF` refers to the union of these three groups.

Instances from ICCMA 2019/2021 We have also selected the "2019" set of instances from the ICCMA 2021 competition [17]. These instances were sampled from the ICCMA 2019 competition benchmarks in order to provide challenging instances, but not too challenging (in order to avoid a high number of timeouts, which does not help to rank solvers). For our purpose, these instances are also relevant since we want to compare the approximate methods with an exact method, which could often reach the timeout if the instances are too hard. This set of instances is made of 107 AFs, distributed as follows:

- A1 (2), A2 (10), A3 (13), A4 (4), B1 (1), B2 (10), B3 (16), B4 (1), C1 (5), C2 (6), C3 (1), T2 (8), T3 (13), T4 (5) (instances from ICCMA 2017 [13]),
- S (1), M (7) (instances from [23]),
- N (4) (instances from [14]).

The number of arguments in these AFs varies between 102 and 8034 arguments. In the following, we collectively refer to this group of AFs as `iccma19`.

5 Empirical Analysis

A Python implementation of the SAT-based encoding from [16], called Pygarg, was used to obtain the correct answers, allowing us to evaluate the solvers accuracy, *i.e.* the ratio of instances that are correctly solved. We can also compare the runtime of approximate algorithms with the runtime of exact algorithms.

5.1 Solving Time

Table 1 contains the running time of the exact solver Pygarg, our two approximate solvers (ARIPOTER-hcat[2] and ARIPOTER-degrees[3]) and the approximate solver Harper++ on `iccma19`. Note that we have chosen to display only the time taken by the solvers to solve the problem, without including the import time. Indeed, the import time is the same for both exact and approximate approaches, and takes an average of 0.1 seconds for these instances. As expected, we observe that the running time of approximate reasoning is effectively much lower (always under one second) with respect to exact solvers. This clearly justifies the interest of using approximate approaches in practice whenever possible.

5.2 Accuracy

Now we provide the accuracy of our solvers on the benchmarks from Section 4.1.

[2] `https://github.com/jeris90/approximate_hcat`
[3] `https://github.com/jeris90/approximate_inout`

solver	DC-pr	DC-stb	DS-pr	DS-stb
Pygarg	57.0923	24.0770	48.6878	39.8206
ARIPOTER-hcat	0.0148	0.0169	0.0201	0.0168
ARIPOTER-degrees	0.0019	0.0020	0.0019	0.0019
Harper++	0.0019	0.0020	0.0019	0.0019

Table 1. Average running time (in seconds, rounded to 10^{-4}) of Pygarg, ARIPOTER-hcat, ARIPOTER-degrees and Harper++ on `iccma19`. This time includes only the solving of the problem (and not the import of the graph).

Instances `randomAF` - Table 2 shows the accuracy of the different approximate solvers on random instances. Columns represent the decision problems (DC-`pr`, DC-`stb`, DS-`pr`, DS-`stb`), and rows correspond to solvers for each family of instances (`rER`, `rBA`, `rWS`) and for the full set of instances `randomAF`. For each family of instances, the highest accuracy for each problem is bold-faced.
Let us start by focusing on the last line (`randomAF`). We observe that ARIPOTER-hcat reaches the best accuracy for the credulous acceptability problems with around 93% correct answers. It is followed by ARIPOTER-degrees ($\simeq$ 83%) and finally Harper++ ($\simeq$ 42%). For DS-`pr`, ARIPOTER-degrees slightly dominates the other approaches. Note however that the three solvers have an excellent accuracy with more than 97% correct answers. Finally, the accuracy for DS-`stb` is globally lower than the other problems. Indeed, it is better solved by ARIPOTER-degrees with around 78% correct answers. This can be explained by the particularity of the stable semantics whose set of extensions may be empty, which implies that all arguments are skeptically accepted.

We also study the results specifically for each family of instances. Indeed, unlike Harper++, our approaches are parameterised and the optimal choice of parameters may depend on both the topology of the graphs and the problem to be solved. For example, ARIPOTER-degrees and ARIPOTER-hcat obtain the best results (more than 95% correct answers) for all problems except DS-`stb` on `rER` and `rWS` when the parameter values are high. Conversely, the accuracy is highest when the parameter values are minimum for the DS-`stb` problem. For these two families of instances, we can also see that Harper++ has a very low accuracy on all problems except DS-`pr`. Finally, for the instances from `rBA`, it is interesting to note that the results returned by the three approximate solvers match perfectly with the exact solvers. This is explained first by the fact that more instances are directly "solved" by the grounded semantics (1925 against 245 for `rER` and 161 for `rWS`). The second reason comes from the way these AFs are constructed because it allows each argument to appear in at least one extension.

Instances `iccma19` - We have also computed the accuracy of our solvers on the instances `iccma19`. As we do not know the exact structure of the AFs, we have chosen to use the values of k and τ that obtained the best accuracy for

Instances	Solver	DC-pr	DC-stb	DS-pr	DS-stb
rER	Harper++	0.125387	0.121053	0.960991	0.220433
	ARIPOTER-degrees	**0.951084** (8)	**0.951084** (10)	**0.961610** (8)	**0.841796** (0)
	ARIPOTER-hcat	0.950464 (1)	0.950464 (1)	0.960991 (1)	**0.841796** (0.1)
rBA	Harper++	**1.0**	**1.0**	**1.0**	**1.0**
	ARIPOTER-degrees	**1.0** (0)	**1.0** (0)	**1.0** ($\|\mathcal{A}\|$)	**1.0** ($\|\mathcal{A}\|$)
	ARIPOTER-hcat	**1.0** (0)	**1.0** (0)	**1.0** (1)	**1.0** (1)
rWS	Harper++	0.100333	0.098333	**0.977333**	0.18
	ARIPOTER-degrees	**0.953333** ($\|\mathcal{A}\|$)	**0.953333** ($\|\mathcal{A}\|$)	**0.977333** ($\|\mathcal{A}\|$)	**0.863** (0)
	ARIPOTER-hcat	**0.953333** (1)	**0.953333** (1)	**0.977333** (1)	**0.863** (0.1)
randomAF	Harper++	0.416068	0.413953	0.979443	0.473784
	ARIPOTER-degrees	0.830550 (8)	0.830550 (10)	**0.979655** (8)	**0.780233** (0.1)
	ARIPOTER-hcat	**0.932135** (0.5)	**0.930444** (0.5)	0.979443 (1)	0.764799 (0.1)

Table 2. Accuracy comparison of the three approximate solvers for rER, rBA, rWS and randomAF, with the values of k (for ARIPOTER-degrees) or τ (for ARIPOTER-hcat) between brackets.

Instances	Solver	DC-pr	DC-stb	DS-pr	DS-stb
iccma19	Harper++	0.754902	0.757009	**0.971154**	**0.826923**
	ARIPOTER-degrees	**0.794118** ($\|\mathcal{A}\|$)	**0.813084** ($\|\mathcal{A}\|$)	**0.971154** ($\|\mathcal{A}\|$)	0.548077 (0.1)
	ARIPOTER-hcat	**0.794118** (0.5)	**0.813084** (0.5)	**0.971154** (1)	0.538462 (0.1)

Table 3. Accuracy comparison of the three approximate solvers for iccma19, with the values of k (for ARIPOTER-degrees) or τ (for ARIPOTER-hcat) between brackets.

the instances randomAF (see Table 2). Thus, we have $k = |\mathcal{A}|$ and $\tau = 0.5$ for DC-pr; $k = |\mathcal{A}|$ and $\tau = 0.5$ for DC-stb; $k = |\mathcal{A}|$ and $\tau = 1$ for DS-pr; and $k = \tau = 0.1$ for DS-stb. The results can be found in Table 3.

The first observation is that, once again, ARIPOTER-degrees and ARIPOTER-hcat return results that are very similar for the four problems studied. In comparison with previous results, we can see that the accuracy is slightly lower for the DC problems, but is still around 80% of correct answers. However, this decrease is more significant for the DS-stb problem where our solvers obtain an accuracy which is around 0.54. Indeed, contrary to the random instances, most of which have no stable extensions (*i.e.*, all arguments are skeptically accepted), here 94 instances over 107 have at least one extension, so determining skeptical acceptability is not trivial. Finally, it is interesting to note that for the DS-pr problem, the accuracy remains extremely high (around 0.97) and is therefore a serious alternative to the exact approach.

6 Related Work

Besides Harper++, the only approaches in the literature on approximate reasoning in abstract argumentation are based on machine learning approaches. Among them, the solver AFGCN [18] participated to ICCMA 2021. This solver is based on Graph Convolutional Networks. While AFGCN globally performed better

than Harper++ regarding the accuracy of the result, the computation time was much higher (and it is reasonable to assume that our approaches would also outperform AFGCN regarding runtime). The other approaches [15, 19, 8] also use graph neural networks to predict the acceptability of arguments. While these approaches can achieve really high accuracies, they require learning data and time for performing this learning task, which is not the case with Harper++ and our new approaches. An empirical comparison of these techniques with ours is an interesting idea for future work. Finally, in structured argumentation, an approach has been proposed to solve problems in ASPIC+ by using only a subset of the full set of arguments, thus diminishing drastically the runtime while attaining a high accuracy [22]. A direct comparison of our approach with this method is not relevant, since the framework is not the same. However, adapting the idea of argument sampling to abstract argumentation and comparing this method with ours is also an interesting track for future research.

7 Conclusion

We have studied new approaches for approximate reasoning in abstract argumentation, solving credulous and skeptical acceptability problems. We have shown that our two solvers (ARIPOTER-degrees and ARIPOTER-hcat) are competitive, in terms of accuracy, with respect to the state of the art approach which was implemented in the solver Harper++ at the last ICCMA competition. They also clearly outperform a standard SAT-based approach for solving these problems.

We are planning to extend this work in three directions. First, we would like to continue studying how to select the parameters that give the best accuracy in relation to the type of AF, the semantics, and the inference task. Second, it would be interesting to extend our approaches for reasoning with other extension-based semantics (*e.g.* ideal, stage, semi-stable, etc), using other gradual semantics instead of h-categorizer (or an extension of it [4]), and the comparison with the approaches using machine learning. Third, we would like to determine the cases (*e.g.* types of graph and their characteristics such as the number of nodes, the density of the attack relation, etc.) where the use of an approximate solver would become preferable to the use of an exact solver.

References

1. Amgoud, L.: A replication study of semantics in argumentation. In: Kraus, S. (ed.) Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019. pp. 6260–6266. ijcai.org (2019)
2. Baroni, P., Rago, A., Toni, F.: From fine-grained properties to broad principles for gradual argumentation: A principled spectrum. Int. J. Approx. Reason. **105**, 252–286 (2019)
3. Besnard, P., Hunter, A.: A logic-based theory of deductive arguments. Artif. Intell. **128**(1-2), 203–235 (2001)

4. Beuselinck, V., Delobelle, J., Vesic, S.: A principle-based account of self-attacking arguments in gradual semantics. J. Log. Comput. **33**(2), 230–256 (2023)
5. Caminada, M.: On the issue of reinstatement in argumentation. In: Fisher, M., van der Hoek, W., Konev, B., Lisitsa, A. (eds.) Logics in Artificial Intelligence, 10th European Conference, JELIA 2006, Liverpool, UK, September 13-15, 2006, Proceedings. Lecture Notes in Computer Science, vol. 4160, pp. 111–123. Springer (2006)
6. Cerutti, F., Giacomin, M., Vallati, M.: Generating structured argumentation frameworks: AFBenchGen2. In: Baroni, P., Gordon, T.F., Scheffler, T., Stede, M. (eds.) Computational Models of Argument - Proceedings of COMMA 2016, Potsdam, Germany, 12-16 September, 2016. Frontiers in Artificial Intelligence and Applications, vol. 287, pp. 467–468. IOS Press (2016)
7. Cerutti, F., Thimm, M., Vallati, M.: An experimental analysis on the similarity of argumentation semantics. Argument Comput. **11**(3), 269–304 (2020)
8. Craandijk, D., Bex, F.: Deep learning for abstract argumentation semantics. In: Bessiere, C. (ed.) Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020. pp. 1667–1673. ijcai.org (2020)
9. Delobelle, J., Mailly, J.G., Rossit, J.: Revisiting approximate reasoning based on grounded semantics. In: Proceedings of the 17th European Conference Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU 2023) (2023)
10. Dung, P.M.: 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)
11. Dvorák, W., Dunne, P.E.: Computational problems in formal argumentation and their complexity. In: Baroni, P., Gabbay, D., Giacomin, M., van der Torre, L. (eds.) Handbook of Formal Argumentation, pp. 631–688. College Publications (2018)
12. Dvorák, W., Gaggl, S.A., Rapberger, A., Wallner, J.P., Woltran, S.: The ASPARTIX system suite. In: Prakken, H., Bistarelli, S., Santini, F., Taticchi, C. (eds.) Computational Models of Argument - Proceedings of COMMA 2020, Perugia, Italy, September 4-11, 2020. Frontiers in Artificial Intelligence and Applications, vol. 326, pp. 461–462. IOS Press (2020)
13. Gaggl, S.A., Linsbichler, T., Maratea, M., Woltran, S.: Design and results of the second international competition on computational models of argumentation. Artif. Intell. **279** (2020)
14. Gao, Y.: A random model for argumentation framework: Phase transitions, empirical hardness, and heuristics. In: Sierra, C. (ed.) Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017. pp. 503–509. ijcai.org (2017)
15. Kuhlmann, I., Thimm, M.: Using graph convolutional networks for approximate reasoning with abstract argumentation frameworks: A feasibility study. In: Amor, N.B., Quost, B., Theobald, M. (eds.) Scalable Uncertainty Management - 13th International Conference, SUM 2019, Compiègne, France, December 16-18, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11940, pp. 24–37. Springer (2019)
16. Lagniez, J.M., Lonca, E., Mailly, J.G.: CoQuiAAS: A constraint-based quick abstract argumentation solver. In: 27th IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2015, Vietri sul Mare, Italy, November 9-11, 2015. pp. 928–935. IEEE Computer Society (2015)
17. Lagniez, J., Lonca, E., Mailly, J., Rossit, J.: Design and results of ICCMA 2021. CoRR **abs/2109.08884** (2021), `https://arxiv.org/abs/2109.08884`

18. Malmqvist, L.: AFGCN: An approximate abstract argumentation solver. http://argumentationcompetition.org/2021/downloads/afgcn.pdf (2021)
19. Malmqvist, L., Yuan, T., Nightingale, P., Manandhar, S.: Determining the acceptability of abstract arguments with graph convolutional networks. In: Gaggl, S.A., Thimm, M., Vallati, M. (eds.) Proceedings of the Third International Workshop on Systems and Algorithms for Formal Argumentation co-located with the 8th International Conference on Computational Models of Argument (COMMA 2020), September 8, 2020. CEUR Workshop Proceedings, vol. 2672, pp. 47–56. CEUR-WS.org (2020)
20. Niskanen, A., Järvisalo, M.: μ-toksia: An efficient abstract argumentation reasoner. In: Calvanese, D., Erdem, E., Thielscher, M. (eds.) Proceedings of the 17th International Conference on Principles of Knowledge Representation and Reasoning, KR 2020, Rhodes, Greece, September 12-18, 2020. pp. 800–804 (2020)
21. Thimm, M.: Harper++: Using grounded semantics for approximate reasoning in abstract argumentation. http://argumentationcompetition.org/2021/downloads/harper++.pdf (2021)
22. Thimm, M., Rienstra, T.: Approximate reasoning with ASPIC+ by argument sampling. In: Gaggl, S.A., Thimm, M., Vallati, M. (eds.) Proceedings of the Third International Workshop on Systems and Algorithms for Formal Argumentation co-located with the 8th International Conference on Computational Models of Argument (COMMA 2020), September 8, 2020. CEUR Workshop Proceedings, vol. 2672, pp. 22–33. CEUR-WS.org (2020)
23. Yun, B., Vesic, S., Croitoru, M.: Toward a more efficient generation of structured argumentation graphs. In: Modgil, S., Budzynska, K., Lawrence, J. (eds.) Computational Models of Argument - Proceedings of COMMA 2018, Warsaw, Poland, 12-14 September 2018. Frontiers in Artificial Intelligence and Applications, vol. 305, pp. 205–212. IOS Press (2018)