ArgSemSAT-1.0: Exploiting SAT Solvers in Abstract Argumentation

Federico Cerutti[1], Mauro Vallati[2], and Massimiliano Giacomin[3]

[1] School of Natural and Computing Science,
University of Aberdeen,
f.cerutti@abdn.ac.uk

[2] School of Computing and Engineering,
University of Huddersfield, m.vallati@hud.ac.uk

[3] Department of Information engineering,
University of Brescia, massimiliano.giacomin@unibs.it

Abstract. In this paper we describe the system ArgSemSAT-1.0 which includes algorithms that efficiently address several decision and enumeration problems — associated to various semantics — in abstract argumentation.

1 Introduction

Dung's abstract argumentation framework is one of the most widely used in computational argumentation by virtue of its simplicity and ability to capture a variety of more specific approaches as special cases. An abstract argumentation framework (AF) consists of a set of arguments and an *attack* relation between them. The concept of *extension* plays a key role in this simple setting: intuitively, it is a set of arguments which can "survive together." Different notions of extensions and of the requirements they should satisfy correspond to alternative *argumentation semantics*. The main computational problems in abstract argumentation are related to extensions and can be partitioned into two classes: *decision* problems and *functional* problems [10].

In this paper we illustrate ArgSemSAT-1.0, a collection of algorithms [6–8] for solving enumeration and sceptical/credulous acceptance problems for grounded, complete, preferred and stable semantics.

2 Background

An argumentation framework [9] consists of a set of arguments[4] and a binary attack relation between them.

Definition 1. *An* argumentation framework *(AF) is a pair $\Gamma = \langle \mathcal{A}, \mathcal{R} \rangle$ where $\mathcal{A}$ is a set of arguments and $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$. We say that* $\boldsymbol{b}$ attacks $\boldsymbol{a}$ *iff $\langle \boldsymbol{b}, \boldsymbol{a} \rangle \in \mathcal{R}$, also denoted as $\boldsymbol{b} \rightarrow \boldsymbol{a}$. The set of attackers of an argument $\boldsymbol{a}$ will be denoted as $\boldsymbol{a}^{-} \triangleq \{\boldsymbol{b} : \boldsymbol{b} \rightarrow \boldsymbol{a}\}$.*

[4] In this paper we consider only *finite* sets of arguments: see [3] for a discussion on infinite sets of arguments.

The basic properties of conflict–freeness, acceptability, and admissibility of a set of arguments are fundamental for the definition of argumentation semantics.

Definition 2. *Given an AF $\Gamma = \langle \mathcal{A}, \mathcal{R} \rangle$:*

- *a set $S \subseteq \mathcal{A}$ is* conflict–free *if $\nexists\, \boldsymbol{a}, \boldsymbol{b} \in S$ s.t. $\boldsymbol{a} \to \boldsymbol{b}$;*
- *an argument $\boldsymbol{a} \in \mathcal{A}$ is* acceptable *with respect to a set $S \subseteq \mathcal{A}$ if $\forall \boldsymbol{b} \in \mathcal{A}$ s.t. $\boldsymbol{b} \to \boldsymbol{a}$, $\exists\, \boldsymbol{c} \in S$ s.t. $\boldsymbol{c} \to \boldsymbol{b}$;*
- *a set $S \subseteq \mathcal{A}$ is* admissible *if S is conflict–free and every element of S is acceptable with respect to S.*

An argumentation semantics σ prescribes for any $AF\ \Gamma$ a set of *extensions*, denoted as $\mathcal{E}_\sigma(\Gamma)$, namely a set of sets of arguments satisfying some conditions dictated by σ.

Definition 3. *Given an AF $\Gamma = \langle \mathcal{A}, \mathcal{R} \rangle$:*

- *a set $S \subseteq \mathcal{A}$ is a* complete extension*, i.e. $S \in \mathcal{E}_{\mathcal{CO}}(\Gamma)$, iff S is admissible and $\forall \boldsymbol{a} \in \mathcal{A}$ s.t. $\boldsymbol{a}$ is acceptable w.r.t. S, $\boldsymbol{a} \in S$;*
- *a set $S \subseteq \mathcal{A}$ is a* preferred extension*, i.e. $S \in \mathcal{E}_{\mathcal{PR}}(\Gamma)$, iff S is a maximal (w.r.t. set inclusion) complete set;*
- *a set $S \subseteq \mathcal{A}$ is the* grounded extension*, i.e. $S \in \mathcal{E}_{\mathcal{GR}}(\Gamma)$, iff S is the minimal (w.r.t. set inclusion) complete set;*
- *a set $S \subseteq \mathcal{A}$ is a* stable extension*, i.e. $S \in \mathcal{E}_{\mathcal{ST}}(\Gamma)$, iff S is a complete set where $\forall \boldsymbol{a} \in \mathcal{A} \setminus S, \exists \boldsymbol{b} \in S$ s.t. $\boldsymbol{b} \to \boldsymbol{a}$.*

Each extension implicitly defines a three-valued *labelling* of arguments (cf. Def. 4). In the light of this correspondence, argumentation semantics can equivalently be defined in terms of labellings rather than of extensions (see [4, 2]). In particular, the notion of *complete labelling* [5, 2] provides an equivalent characterization of complete semantics, in the sense that each complete labelling corresponds to a complete extension and vice versa. Complete labellings can be (redundantly) defined as follows.

Definition 4. *Let $\langle \mathcal{A}, \mathcal{R} \rangle$ be an argumentation framework. A total function $\mathcal{L}ab : \mathcal{A} \mapsto \{\mathtt{in}, \mathtt{out}, \mathtt{undec}\}$ is a* complete labelling *iff it satisfies the following conditions for any $\boldsymbol{a} \in \mathcal{A}$:*

- $\mathcal{L}ab(\boldsymbol{a}) = \mathtt{in} \Leftrightarrow \forall \boldsymbol{b} \in \boldsymbol{a}^- \mathcal{L}ab(\boldsymbol{b}) = \mathtt{out}$;
- $\mathcal{L}ab(\boldsymbol{a}) = \mathtt{out} \Leftrightarrow \exists \boldsymbol{b} \in \boldsymbol{a}^- : \mathcal{L}ab(\boldsymbol{b}) = \mathtt{in}$;
- $\mathcal{L}ab(\boldsymbol{a}) = \mathtt{undec} \Leftrightarrow \forall \boldsymbol{b} \in \boldsymbol{a}^- \mathcal{L}ab(\boldsymbol{b}) \neq \mathtt{in} \wedge \exists \boldsymbol{c} \in \boldsymbol{a}^- : \mathcal{L}ab(\boldsymbol{c}) = \mathtt{undec}$;

It is proved in [4] that:

- preferred extensions are in one-to-one correspondence with those complete labellings maximising the set of arguments labelled `in`;
- the grounded extension is in in one-to-one correspondence with the complete labelling maximising the set of arguments labelled `undec`;
- stable extensions are in one-to-one correspondence with those complete labellings with no argument labelled `undec`.

3 **ArgSemSAT-1.0**

`ArgSemSAT-1.0` is a set of search algorithms in the space of complete extensions to identify also preferred, stable and the grounded extensions (enumeration problems) as well as solving decisions problems associated to those semantics, namely credulous and skeptical acceptance of an argument. `ArgSemSAT-1.0` encodes the constraints corresponding to complete labellings of an AF as a SAT problem and then iteratively producing and solving modified versions of the initial SAT problem according to the needs of the search process. `ArgSemSAT-1.0` has been implemented in C++, and exploits the Glucose SAT solver [1].

For instance, Alg. 1 shows the general idea of the current implementation in `ArgSemSAT-1.0` for enumerating preferred extensions.

Algorithm 1 Enumeration of Preferred Extensions

Input: $\Gamma = \langle \mathcal{A}, \mathcal{R} \rangle$
Output: $E_p \subseteq 2^{\mathcal{A}}$
$E_p := \emptyset$
$cnf := \Pi_\Gamma \wedge \bigvee_{\mathbf{a} \in \mathcal{A}} I_{\phi^{-1}(\mathbf{a})}$
repeat
  $cnfdf := cnf$
  $prefcand := \emptyset$
  repeat
    $lastcompfound := SATSOLV(cnfdf)$
    if $lastcompfound \neq \varepsilon$ **then**
      $emptyundec := UNDECARGS(lastcompfound) = \emptyset$
      $prefcand := lastcompfound$
      for $\mathbf{a} \in INARGS(lastcompfound)$ **do**
        $cnfdf := cnfdf \wedge I_{\phi^{-1}(\mathbf{a})}$
      end for
      $remaining := FALSE$
      for $\mathbf{a} \in OUTARGS(lastcompfound)$ **do**
        $cnfdf := cnfdf \wedge O_{\phi^{-1}(\mathbf{a})}$
        $remaining := remaining \vee I_{\phi^{-1}(\mathbf{a})}$
      end for
      $remaining\_df := FALSE$
      for $\mathbf{a} \in UNDECARGS(lastcompfound)$ **do**
        $remaining := remaining \vee I_{\phi^{-1}(\mathbf{a})}$
        $remaining\_df := remaining\_df \vee I_{\phi^{-1}(\mathbf{a})}$
      end for
      $cnfdf := cnfdf \wedge remaining\_df$
      $cnf := cnf \wedge remaining$
    end if
  until $(lastcompfound = \varepsilon \vee emptyundec = \emptyset)$
  if $prefcand \neq \emptyset$ **then**
    $E_p := E_p \cup \{INARGS(prefcand)\}$
  end if
until $(prefcand = \emptyset \vee prefcand = \mathcal{A})$
if $E_p = \emptyset$ **then**
  $E_p = \{\emptyset\}$
end if
return E_p

In Alg. 1, Π_Γ is a CNF representing the constraints for complete labellings; $\phi^{-1} : \mathcal{A} \mapsto \mathbb{N}$; I_j (resp. O_j and U_j) is a SAT variable identifying the case that the j-th argument is `in` (resp. `out` and `undec`); $SATSOLV$ is a SAT solver which returns a satisfiable assignment of variables or ε if UNSAT; $INARGS$ (reps. $OUTARGS$ and $UNDECARGS$) is a function that takes as input a variable assignment and returns the set of arguments labelled as `in` (resp. `out` and `undec`) in such an assignment.

Acknowledgement

The authors would like to acknowledge the use of the University of Huddersfield Queensgate Grid in carrying out this work.

References

1. Audemard, G., Simon, L.: Lazy clause exchange policy for parallel sat solvers. In: Theory and Applications of Satisfiability Testing–SAT 2014, pp. 197–205 (2014)
2. Baroni, P., Caminada, M., Giacomin, M.: An introduction to argumentation semantics. Knowledge Engineering Review 26(4), 365–410 (2011)
3. Baroni, P., Cerutti, F., Dunne, P.E., Giacomin, M.: Automata for infinite argumentation structures. Artificial Intelligence 203, 104 – 150 (2013)
4. Caminada, M.: On the issue of reinstatement in argumentation. In: Proceedings of JELIA 2006. pp. 111–123 (2006)
5. Caminada, M., Gabbay, D.M.: A logical account of formal argumentation. Studia Logica (Special issue: new ideas in argumentation theory) 93(2–3), 109–145 (2009)
6. Cerutti, F., Dunne, P.E., Giacomin, M., Vallati, M.: A SAT-based Approach for Computing Extensions in Abstract Argumentation. In: Second International Workshop on Theory and Applications of Formal Argumentation (TAFA-13) (2013)
7. Cerutti, F., Dunne, P.E., Giacomin, M., Vallati, M.: Computing Preferred Extensions in Abstract Argumentation: A SAT-Based Approach. In: Black, E., Modgil, S., Oren, N. (eds.) TAFA 2013, Lecture Notes in Computer Science, vol. 8306, pp. 176–193. Springer-Verlag Berlin Heidelberg (2014)
8. Cerutti, F., Giacomin, M., Vallati, M.: ArgSemSAT: Solving Argumentation Problems Using SAT. In: Parsons, S., Oren, N., Reed, C., Cerutti, F. (eds.) 5th Conference on Computational Models of Argument. pp. 455–456 (2014)
9. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming, and n-person games. Artificial Intelligence 77(2), 321–357 (1995)
10. Papadimitriou, C.H., Steiglitz, K.: Combinatorial optimization: algorithms and complexity. Courier Corporation (1998)