

# On the Incremental Computation of Argumentation Frameworks

Gianvincenzo Alfano  
supervised by Prof. Sergio Greco

Department of Informatics, Modeling, Electronics and System Engineering (DIMES)  
University of Calabria, Rende (CS), Italy  
{g.alfano, greco}@dimes.unical.it

**Abstract.** We tackle the problem of efficiently recomputing extensions of dynamic Argumentation Frameworks (AFs) under well-known semantics, namely frameworks able to model human ability to argue. We introduce an incremental approach that, given an initial argumentation framework, an initial extension for it, and an update, computes an extension of the updated framework. The proposed approach is then extended to different kinds of argumentation frameworks (i.e., abstract, bipolar, extended, structured), and is able to incorporate existing AF-solvers in order to compute an extension of the updated framework.

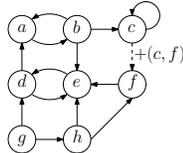
**Keywords:** Argumentation · Knowledge Representation · Incremental Algorithm.

## 1 Introduction

Argumentation has emerged as one of the major fields in Artificial Intelligence [10,17,7]. In particular, abstract argumentation frameworks (AFs) [11] are simple and powerful formalism for modelling disputes between two or more agents. The formal meaning of an AF is given in terms of argumentation semantics, which intuitively tell us the sets of arguments (called *extensions*) that can be exploited to support a point of view in a discussion. Although the idea underlying AFs is very simple and intuitive, most of the argumentation semantics proposed so far suffer from a high computational complexity [12]. Complexity bounds and evaluation algorithms for AFs have been deeply investigated in the literature, as also witnessed by the International Competition on Computational Models of Argumentation (ICCMA, <http://argumentationcompetition.org>) whose aim is encouraging research and development of efficient algorithms for computational models of AFs. However, most research focused on ‘static’ frameworks, whereas in practice AFs are dynamic systems [9,8]. In fact, typically an AF represents a temporary situation, and new arguments and attacks can be added/retracted to take into account new available knowledge.

## 2 Argumentation Frameworks

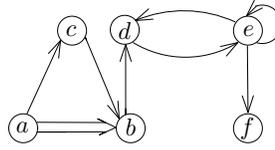
Literature offers several formalism for modeling disputes between two or more agents. Each of them belongs to one of the two major argumentation categories, namely *ab-*



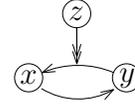
**Fig. 1.** AFs  $\mathcal{A}_0$  and  $\mathcal{A} = + (c, f)(\mathcal{A}_0)$ .



**Fig. 2.** R-AF



**Fig. 3.** BAF for weather exam-  
ple.



**Fig. 4.** EAF for  
weather example.

*struct*, where arguments are abstract entities (i.e., words, phrases, etc.) since we abstain on where it comes from, and *structured*, where the structure describing how an argument is built is also provided in the framework. Here we briefly introduce those frameworks over which we proposed an incremental technique for recomputing extensions after a change operation performed over the framework.

*Abstract Argumentation Frameworks.* An argument is an abstract entity whose role is entirely determined by its relationships with other arguments. We assume the existence of a set  $Arg$  of arguments. An AF is a pair  $\langle A, \Sigma \rangle$ , where  $A \subseteq Arg$  and  $\Sigma \subseteq A \times A$  is a binary relation over  $A$  whose elements are called *attacks*. Thus, an AF is a directed graph where nodes correspond to arguments and edges correspond to attacks. An argumentation semantics specifies the criteria for identifying a set of arguments that can be considered “reasonable” together, called *extension*. Some of the well-known semantics for abstract argumentation are *complete*, *preferred*, *stable*, *grounded* and *ideal*.

*Bipolar Abstract Argumentation Frameworks.* Bipolar argumentation frameworks (BAFs) extend Dung’s argumentation frameworks to explicitly represent the notion of support between arguments, in addition to that of attack. An *abstract bipolar argumentation framework* (BAF for short) [6] is a triple  $\langle A, \Sigma, \Pi \rangle$ , where  $\Pi \subseteq A \times A$  is a binary relation over  $A$  whose elements are called *supports*, and  $\Sigma \cap \Pi = \emptyset$ . Thus, a Dung’s argumentation framework (AF) [11] is a BAF of the form  $\langle A, \Sigma, \emptyset \rangle$ . A BAF can be viewed as a directed graph where each node corresponds to an argument and each edge in the graph corresponds to either an attacks ( $\rightarrow$ ) or a support ( $\Rightarrow$ ). An example is reported on Figure 3.

*Extended Abstract Argumentation Frameworks.* Extended argumentation frameworks (EAFs) extend Dung’s argumentation frameworks (AFs) to represent a kind of defeasible attack in addition to the Dung’s classical notion of attack. Consider the following arguments about weather forecast: argument  $x$  is “The week-end will be dry in Rome since AccuWeather forecasts sunshine”, while argument  $y$  is “The week-end will be wet in Rome since The Weather Channel forecasts rain”. These arguments attack each other, as they claim contradictory conclusions. One could be persuaded that AccuWeather is recognized to be more trustworthy than The Weather Channel, meaning that  $y$  does not successfully attack  $x$ . This kind of situations can be modelled in EAFs that incorporate *second-order attacks* (i.e., attacks to classical Dung’s attacks) whose intuitive meaning is “deactivating” the attacked attack whenever the argument from which the second-order attack originates is accepted. The resulting EAF is shown in Figure 4.

*Defeasible Logic Programming.* DeLP incorporates a defeasible argumentation formalism for the treatment of contradictory knowledge. A *dialectical process* is used for

deciding which information prevails as warranted. This dialectical process involves the construction and evaluation of arguments that either support or interfere with the query under analysis. A DeLP program  $\mathcal{P} = (\Pi, \Delta)$  is a set of *facts*, *strict rules* and *defeasible rules*. Intuitively, facts and strict rules represent non-defeasible information, while defeasible rules represent tentative information, that is, information that can be used if nothing could be posed against it. In a program  $\mathcal{P}$ , we will distinguish the subset  $\Pi$  of facts and strict rules, and the subset  $\Delta$  of defeasible rules.

### 3 Research Focus

Argumentation is an inherently dynamic process which springs accordingly argumentation frameworks that changes over time. Performing an update on an AF  $\mathcal{A}_0$  means modifying it into an AF  $\mathcal{A}$  by adding or retracting arguments or attacks. We first proposed a technique for incrementally computing extensions of dynamic abstract argumentation frameworks (AFs): given an initial extension and a set of updates, our algorithm computes an extension of the updated AF under most popular argumentation semantics [13,15,14,2]. The algorithm consists of the following three main steps: (i) identification of a (smaller) sub-AF, called *reduced AF* (R-AF for short), on the basis of the update and additional information provided by the initial extension (ii) using *any* non-incremental algorithm to compute an extension of the reduced AF; and (iii) getting the final extension by merging a portion of the initial extension with that computed for the reduced AF. To make an example, consider the AF reported on Figure 1, and consider the update operation  $u = +(c, f)$ . The reduced framework (the sub-AF affected by the change) consists of only arguments  $e$  and  $f$ , having as initial grounded extension  $\{f, g\}$ . The reduced AF is reported on Figure 2.

From an experimental analysis, the algorithm performs two orders of magnitude faster than the best solvers computing the semantics from scratch. This allowed us to extend the technique to cope with other different kinds of (abstract) argumentation frameworks. Particularly, in [1] we applied this technique after that a bipolar argumentation framework is mapped into a Dung's equivalent one using additional meta-knowledge. Using the meta-argumentation approach, namely a procedure to convert a framework into a Dung's equivalent one, we envisage the possibility to apply the incremental technique to compute extensions over EAFs [3], where they are firstly mapped into an abstract equivalent one by adding pieces of knowledge (i.e., using meta-arguments and meta-attacks) useful to convert second-order attacks into 'classical' attacks.

Our current research is to identify similar approach for structured argumentation frameworks. A first attempt was proposed in [5] where DeLP formalism is taken into account to model structured information, and updates consisting on adding a defeasible/strict rule. Although the workflow is similar to the algorithm for abstract frameworks, different problems arise when we deal with structured argumentation. One of the major problem is that arguments are not given as input, but they should be build according to the knowledge-base given. For this purpose we envisage the use of an hyper-graph which maps a DeLP-program, and using a particular notion of reachability on hyper-graphs we identify only those portion of the program which needs to be recomputed after that a rule is just added.

## 4 Conclusion and Future Work

Continuing on studying incremental algorithm for structured argumentation frameworks is our primary goal. Particularly, we are currently working on extending our technique for both (strict and defeasible) rule deletion and addition in the DeLP-framework [4]. Additionally, we should also investigate the possibility to find analogous incremental approach to cope with other structured formalisms like *ASPIC*<sup>+</sup> [16]. Another important aspect that reserve more investigation is to extend the technique to enumerate the whole set of extensions when a multiple-status semantics is called over an updated abstract argumentation framework, more precisely, returning the whole set of extensions after that an update is computed.

## References

1. Alfano, G., Greco, S., Parisi, F.: Computing stable and preferred extensions of dynamic bipolar argumentation frameworks. In: Proc. of Workshop on Advances In Argumentation In Artificial Intelligence (*AI*<sup>3</sup>), co-located with AI\*IA. pp. 28–42 (2017)
2. Alfano, G., Greco, S., Parisi, F.: Efficient computation of extensions for dynamic abstract argumentation frameworks: An incremental approach. In: Proc. of IJCAI. pp. 49–55 (2017)
3. Alfano, G., Greco, S., Parisi, F.: Computing extensions of dynamic abstract argumentation frameworks with second-order attacks. In: Proc. of IDEAS. pp. 183–192 (2018)
4. Alfano, G., Greco, S., Parisi, F., Simari, G.I., Simari, G.R.: An incremental approach to structured argumentation over dynamic knowledge bases. In: Proc. of KR (to appear) (2018)
5. Alfano, G., Greco, S., Parisi, F., Simari, G.I., Simari, G.R.: Incremental computation of warranted arguments in dynamic defeasible argumentation: The rule addition case. In: Proc. of ACM Symposium on Applied Computing (SAC). pp. 911–917 (2018)
6. Amgoud, L., Cayrol, C., Lagasquie-Schieux, M.: On the bipolarity in argumentation frameworks. In: Proc. of NMR. pp. 1–9 (2004)
7. Baroni, P., Caminada, M., Giacomin, M.: An introduction to argumentation semantics. The Knowledge Engineering Review **26**(4), 365–410 (2011)
8. Baroni, P., Giacomin, M., Liao, B.: On topology-related properties of abstract argumentation semantics. A correction and extension to dynamics of argumentation systems: A division-based method. Artificial Intelligence **212**, 104–115 (2014)
9. Baumann, R.: Splitting an argumentation framework. In: Proc. of LPNMR. pp. 40–53 (2011)
10. Bench-Capon, T.J.M., Dunne, P.E.: Argumentation in artificial intelligence. Artificial Intelligence **171**(10 - 15), 619 – 641 (2007)
11. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artificial Intellig. **77**(2), 321–358 (1995)
12. Dunne, P.E., Wooldridge, M.: Complexity of abstract argumentation. In: Argumentation in Artificial Intelligence, pp. 85–104 (2009)
13. Greco, S., Parisi, F.: Efficient computation of deterministic extensions for dynamic abstract argumentation frameworks. In: Proc. of ECAI. pp. 1668–1669 (2016)
14. Greco, S., Parisi, F.: Incremental computation of deterministic extensions for dynamic argumentation frameworks. In: Proc. of JELIA. pp. 288–304 (2016)
15. Greco, S., Parisi, F.: Incremental computation of grounded semantics for dynamic abstract argumentation frameworks. In: Proc. of ECAI Workshop COREDEMA. pp. 66–81 (2016)
16. Modgil, S., Prakken, H.: The *ASPIC*<sup>+</sup> framework for structured argumentation: a tutorial. Argument & Computation **5**(1), 31–62 (2014)
17. Rahwan, I., Simari, G.R.: Argumentation in Artificial Intelligence. Springer Publishing Company, Incorporated, 1st edn. (2009)